

Modeling and Control of a Re-entry Heat Transfer Problem Using Neural Networks

K. A. Grantham, R. Padhi, S. N. Balakrishnan and D. C. Look, Jr.

Abstract— A nonlinear optimal re-entry temperature control problem is solved using single network adaptive critic (SNAC) technique. The nonlinear model developed and used accounts for conduction, convection and radiation at high temperature, represents the dynamics of heat transfer in a cooling fin for an object re-entering the earth's atmosphere. Simulation results demonstrate that the control synthesis technique presented is very effective in obtaining a desired temperature profile over a wide envelope of initial temperature distribution.

I. INTRODUCTION

In this paper, the problem under consideration is a high temperature heat-transfer application, which may represent a cooling fin for an object on re-entry into the Earth's atmosphere. The system dynamics is governed by a nonlinear partial differential equation, and hence, it is a distributed parameter system (DPS). Analysis and control design for such systems are often more complex as compared to lumped parameter systems (which are governed by a set of ordinary differential equations).

An engineering approach to deal with DPS problems is to have a finite dimensional lumped parameter approximation of the system using a set of orthogonal basis functions in a Galerkin projection. To obtain a low-order lumped parameter approximation, the basis functions are designed from a set of snap-shot (representative) solutions, following a technique called proper orthogonal decomposition (POD). Out of the numerous works published in literature on this topic and its use in control system design, we cite [1], [3] [8]-[10] for reference.

Even though POD technique has been used in practice for control design of nonlinear DPS systems, an important issue that remains open is the use of a 'good controller' for simulating the system to collect the snap-shot solutions,

which are used to design the basis functions. Quite often an open-loop controller is used for this purpose. In this work, however, a state feedback controller designed based on the philosophy of dynamic inversion [4] was incorporated to generate the snap-shot solutions. Since this controller is stabilizing, the snap-shots so obtained are more likely to be encountered with the application of the optimal controller.

Many difficult real-life control design problems can be formulated in the framework of optimal control theory. The dynamic programming formulation offers the most comprehensive solution approach to nonlinear optimal control in a state feedback form [2]. However, solving the associated Hamilton-Jacobi-Bellman (HJB) equation demands a very large (rather infeasible) number of computations and storage space requirement. An innovative idea was proposed by Werbos [11] to get around this numerical complexity through a dual neural network approach called adaptive critic (AC). In one version of the AC approach, one network (called the action network) represents the mapping between the state and control variables while a second network (called the critic network) represents the mapping between the state and costate variables. Optimal solution is reached after the two networks are iteratively trained. This process overcomes the computational complexity of the dynamic programming approach.

Recently, a significant improvement to the adaptive critic architecture has been proposed by Padhi et al. [7], [8]. It is named as single network adaptive critic (SNAC) because it uses only the critic network instead of the action-critic dual network set up in typical adaptive critic architecture. SNAC is applicable to a large class of problems for which the optimal control (stationary) equation is explicitly solvable for control in terms of state and costate variables. This leads to significant computational savings besides eliminating the approximation error due to action networks.

In this paper, the SNAC technique is used to design a state feedback optimal controller for the problem under consideration. It has been demonstrated from the numerical simulation studies that the technique solves for the optimal controller for a large number of initial conditions at a time. Note that the solution can be implemented in real-time as the on-line computation involves only using the neural networks.

K. A. Grantham, Graduate Student, was with the Department of Mechanical and Aerospace Engineering, Univ. of Missouri – Rolla, MO.

Radhakant Padhi, Asst. Professor, is with the Department of Aerospace Engineering, Indian Institute of Science, Bangalore, 560012, India (phone: +91-80-2293-2756, e-mail: padhi@aero.iisc.ernet.in).

S. N. Balakrishnan, Professor, is with the Department of Mechanical and Aerospace Engineering, University of Missouri – Rolla, MO, USA (phone: 573-341-4675, e-mail: bala@umr.edu).

D. C. Look Jr., Emer. Professor, is with the Department of Mechanical and Aerospace Eng., University of Missouri – Rolla, MO.

II. MATHEMATICAL MODEL FOR THE PROBLEM

The problem under consideration is a high temperature heat-transfer application, which may represent a cooling fin for an object on re-entry into the Earth's atmosphere. Figure 1 depicts the pertinent geometry.

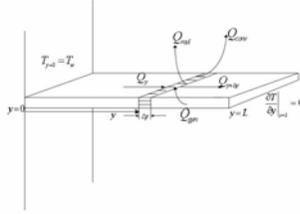


Figure 1: Pictorial representation of the problem

For such a problem, using the law of conservation of energy [6] in an infinitesimal volume at a distance y having length Δy , we get

$$Q_y + Q_{gen} = Q_{y+\Delta y} + Q_{conv} + Q_{rad} + Q_{chg} \quad (1)$$

where,

$$Q_y = -kA(\partial T / \partial y) \quad (2a)$$

$$Q_{gen} = S A \Delta y \quad (2b)$$

$$Q_{conv} = h P \Delta y (T - T_{\infty_1}) \quad (2c)$$

$$Q_{rad} = \epsilon \sigma P \Delta y (T^4 - T_{\infty_2}^4) \quad (2d)$$

$$Q_{chg} = \rho C A \Delta y (\partial T / \partial t) \quad (2e)$$

In (1)–(2a-e), $T(t, y)$ and $S(t, y)$ represents the temperature and $S(t, y)$ is the rate of heat generation per unit volume, which acts as the control variable for this problem. The meanings of various parameters and their numerical values used and are given in Table 1.

Table 1: Parameter Definitions and Numerical Values

Parameter	Definition	Numerical value
k	Thermal conductivity	$19 W / (m^{\circ}C)$
A	Cross sectional area	$2' \times 3''$
P	Perimeter	$4'6''$
h	Convective heat transfer coefficient	$20 W / (m^{\circ}C)$
T_{∞_1}	Temperature of the medium in the immediate surrounding of the surface	$100^{\circ}C$
T_{∞_2}	Temperature at a far away place in the direction normal to the surface	$-40^{\circ}C$
ϵ	Emissivity of the material	0.965
σ	Stefan-Boltzmann constant	$5.669 \times 10^{-8} W / m^2 K^4$
ρ	Density of the material	$7865 kg / m^3$
C	Specific heat of the material	$0.46 kJ / (kg^{\circ}C)$

Substituting the expressions of (2) in (1) and carrying out the necessary algebra, it leads to

$$\frac{\partial T}{\partial t} = \frac{k}{\rho C} \left(\frac{\partial^2 T}{\partial y^2} \right) - \frac{P}{A \rho C} \left[h(T - T_{\infty_1}) + \epsilon \sigma (T^4 - T_{\infty_2}^4) \right] + \left(\frac{1}{\rho C} \right) S \quad (3)$$

For convenience, the following parameters are defined $\alpha_1 \triangleq (k / \rho C)$, $\alpha_2 \triangleq -(Ph) / (A \rho C)$, $\alpha_3 \triangleq -(P \epsilon \sigma) / (A \rho C)$ and $\beta \triangleq 1 / (\rho C)$. Substitution of these parameters in (3) leads to

$$\frac{\partial T}{\partial t} = \alpha_1 \left(\frac{\partial^2 T}{\partial y^2} \right) + \alpha_2 (T - T_{\infty_1}) + \alpha_3 (T^4 - T_{\infty_2}^4) + \beta S \quad (4)$$

Boundary conditions considered for (4) are as follows:

$$\left. \frac{\partial T}{\partial y} \right|_{y=0} = c, \quad \left. \frac{\partial T}{\partial y} \right|_{y=L} = 0 \quad (5)$$

where the value of c will be dictated by the temperature profile $T(t, y)$ at $y=0$. In our numerical implementation, c was to be time varying, the value of which was dictated by the instantaneous value of the slope of $T(t, y)$ at $y=0$.

An insulated boundary condition at the tip is considered with the assumption that either there is some physical insulation at the tip or the heat loss at the tip is negligible (due to its low surface area). Note that a one-dimensional approximation is assumed in the temperature dynamics, assuming that a steady-state is obtained in the other spatial dimensions instantaneously.

III. CONTROL DESIGN: MATHEMATICAL FORMULATION

The goal is to reach a desired temperature profile in the fin; i.e. $T(y) \rightarrow T^*(y)$ as $t \rightarrow \infty$, where $T^*(y)$ is a desired final temperature distribution which is assumed to be a constant profile along the fin.

A. Feedforward Controller

The steady state control solution S^* is obtained by substituting T^* in place of T in (4) and imposing the steady-state condition ($\partial T^* / \partial t = 0$). This leads to

$$S^*(y) = -\frac{1}{\beta} \left[\alpha_1 \left(\frac{\partial^2 T^*}{\partial y^2} \right) + \alpha_2 (T^* - T_{\infty_1}) + \alpha_3 (T^{*4} - T_{\infty_2}^4) \right] \quad (6)$$

This S^* acts as a feedforward controller for this problem. A feedback controller (to be discussed next) is added to it to obtain the total controller.

B. Feedback Controller: Optimal Control Formulation

Due to the availability of the desired steady state temperature profile $T^*(y)$ and associated feedforward controller $S^*(y)$, we split $T(t, y)$ and $S(t, y)$ as follows

$$T(t, y) = T^*(y) + x(t, y) \quad (7a)$$

$$S(t, y) = S^*(y) + u(t, y) \quad (7b)$$

where $x(t, y)$ and $u(t, y)$ are the deviations in temperature (state) and control from their respective steady state values. Next, the deviation dynamics are developed by substituting

these expressions in (5) and carrying out the necessary algebra. This leads to

$$\frac{\partial x}{\partial t} = \alpha_1 (\partial^2 x / \partial y^2) + \alpha_2 x + \alpha_3 f(x) + \beta u \quad (8)$$

where $f(x) \triangleq (4T^3 x + 6T^2 x^2 + 4T x^3 + x^4)$. The associated boundary conditions are obtained by substituting (7a-b) in (5), which are as follows

$$\partial x / \partial y|_{y=0} = c - \partial T^* / \partial y|_{y=0}, \quad \partial x / \partial y|_{y=L} = -\partial T^* / \partial y|_{y=L} \quad (9)$$

The purpose of the feedback controller $u(t, y)$ is to make sure that $x(t, y) \rightarrow 0$ as $t \rightarrow \infty$. Note that in the process it is also desired that $u(t, y) \rightarrow 0$ as $x(t, y) \rightarrow 0$. These goals are achieved by minimizing the following cost function

$$J = \frac{1}{2} \int_0^L \int_0^T (q x^2 + r u^2) dy dt \quad (10)$$

where $q \geq 0$ and $r > 0$ are the weights for state and control respectively. Although the mathematical requirement for obtaining a solution for the controller is $q \geq 0$, in this application we will consider $q > 0$ in order to ensure the cancellation of the state deviations. Equations (8)-(10) results in consistent optimal control formulation. In the following subsections details about the technique used for solving this optimal control problem are addressed.

C. Initial Condition Generation

In this subsection a procedure is given to generate a large number of possible initial conditions. Note that to account for reality, the initial conditions should be continuous and smooth profiles along the spatial domain. To achieve this objective, the initial conditions for the problem $x(y)_{mi}$ is expressed as a Fourier series as

$$x(y)_{mi} = a_0 + \sum_{n=1}^N a_n \cos(2n\pi y/L) + \sum_{n=1}^N b_n \sin(2n\pi y/L) \quad (11)$$

In (11) this equation a_0 , a_n , and b_n are the Fourier series constants (to be determined), N is the number of terms used in the series and L is the length of the fin. The Fourier series constants are chosen by imposing constraints on the norms of the initial condition and its first and second derivatives (with respect to the spatial variable) as follows

$$\|x(y)_{mi}\|^2 \leq k_1, \quad \|x'(y)_{mi}\|^2 \leq k_2, \quad \|x''(y)_{mi}\|^2 \leq k_3 \quad (12)$$

The constants $k_1, k_2, k_3 > 0$ are judiciously selected so that it allows sufficient flexibility to generate a large number of smooth profiles and yet does not lead to too much (unrealistic) waviness in the profiles. The set containing all such initial conditions define our domain of interest.

Note that once the constants $k_1, k_2, k_3 > 0$ are selected and N is fixed, the process of finding $x(y)_{mi}$ include finding the coefficients a_0, a_i, b_i , ($i=1, \dots, N$) satisfying (12) and then substituting those in (11). We omit the details both for brevity as well as for lack of space.

D. Proper Orthogonal Decomposition: A Brief Review

Let $\{x_i(y) : 1 \leq i \leq N, y \in \Omega\}$ be a set of N snapshot solutions (observations) of a physical process over the domain Ω at arbitrary instants of time. The goal of the POD technique is to design a coherent structure that has the largest mean square projection on the snapshots. Consequently, functions φ (the basis functions) are designed which most resemble $\{x_i(y)\}_{i=1}^N$, by maximizing the figure of merit given by

$$I = \frac{1}{N} \sum_{i=1}^N |\langle x_i, \varphi \rangle|^2 / \langle \varphi, \varphi \rangle \quad (13)$$

As a standard notation, the L^2 inner product is defined as $\langle \varphi, \psi \rangle = \int_{\Omega} \varphi \psi dy$. It has been shown in the literature that when the number of degrees of freedom required to describe x_i is larger than the number of snapshots N (always true for infinite dimensional systems), it is sufficient to express the basis functions as linear combinations of the snapshots, i.e.

$$\Phi = \sum_{i=1}^N w_i x_i \quad (14)$$

The coefficients w_i are to be determined such that Φ maximizes the figure of merit in (13). The steps involved in this are as follows:

- Construct an eigenvalue problem

$$C W = \lambda W, \quad C = [c_{ij}]_{N \times N} \quad (15)$$

$$c_{ij} = \frac{1}{N} \int_0^L x_i(y) x_j(y) dy$$

- Obtain N eigenvalues and corresponding eigenvectors of the C matrix. Sort the eigenvalues in descending order $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N \geq 0$. Let the corresponding eigenvectors be

$$W^1 = [w_1^1 \dots w_N^1]^T, \dots, W^N = [w_1^N \dots w_N^N]^T \quad (16)$$

Note that the eigenvectors are orthogonal to each other.

- Normalize the eigenvectors to satisfy

$$\langle W^i, W^i \rangle = (W^i)^T W^i = 1 / (N \lambda_i) \quad (17)$$

This will ensure that the POD basis functions are orthonormal.

- Cut-off the eigenspectrum ‘‘judiciously’’, so that the truncated system with $\tilde{N} \leq N$ eigenvalues will satisfy

$$\sum_{j=1}^{\tilde{N}} \lambda_j \approx \sum_{j=1}^N \lambda_j. \text{ Usually, it turns out that } \tilde{N} \ll N.$$

- Finally, construct the \tilde{N} basis functions as

$$\varphi_1(y) = \sum_{i=1}^{\tilde{N}} w_i^1 U_i, \dots, \varphi_{\tilde{N}}(y) = \sum_{i=1}^{\tilde{N}} w_i^{\tilde{N}} U_i \quad (18)$$

An interested reader can refer to [9] for more details about this procedure.

E. Analogous Lumped Parameter System Representation

After obtaining the basis functions, following the principle of 'separation of variables', $x(t,y)$ and $u(t,y)$ are expanded as follows

$$x(t,y) \approx \sum_{j=1}^{\tilde{N}} \hat{x}_j(t) \Phi_j(y) \quad (19a)$$

$$u(t,y) \approx \sum_{j=1}^{\tilde{N}} \hat{u}_j(t) \Phi_j(y) \quad (19b)$$

One can notice that both $x(t,y)$ and $u(t,y)$ have the same basis functions. This is because the final aim is to design a *state feedback controller*, where the controller is a function of the state variables. Essentially, the basis functions for the state are assumed to be capable of spanning the controller as well. Substituting these expansions of state and control variables in (8), we obtain

$$\sum_{j=1}^{\tilde{N}} \hat{x}_j \varphi_j = \alpha_1 \sum_{j=1}^{\tilde{N}} \hat{x}_j \varphi_j' + \alpha_2 \sum_{j=1}^{\tilde{N}} \hat{x}_j \varphi_j + \alpha_3 f(x) + \beta \sum_{j=1}^{\tilde{N}} \hat{u}_j \varphi_j \quad (20)$$

Next, taking the Galerkin projection of (20) on the basis function φ_i (i.e. taking the inner product with respect to φ_i), and using the fact that the basis functions are orthonormal, we obtain

$$\hat{\dot{x}}_i = \alpha_1 \sum_{j=1}^{\tilde{N}} \langle \varphi_i, \varphi_j \rangle \hat{x}_j + \alpha_2 \hat{x}_i + \alpha_3 \langle f(x), \varphi_i \rangle + \beta \hat{u}_i \quad (21)$$

Repeating this exercise for $i=1, \dots, \tilde{N}$ leads to a low-order analogous lumped-parameter system representation.

$$\dot{\hat{X}} = A\hat{X} + \hat{f}(\hat{X}) + B\hat{U} \quad (22a)$$

where $\hat{X} \triangleq [\hat{x}_1, \dots, \hat{x}_{\tilde{N}}]^T$ and $\hat{U} \triangleq [\hat{u}_1, \dots, \hat{u}_{\tilde{N}}]^T$. Other terms are defined as follows

$$\begin{aligned} A &\triangleq \alpha_1 [a_{ij}] + \alpha_2 I_{\tilde{N}}, \quad B \triangleq \beta I_{\tilde{N}} \\ a_{ij} &\triangleq \langle \varphi_i, \varphi_j' \rangle = \int_0^L \varphi_i \varphi_j' dy = [\varphi_i, \varphi_j']_{y=0}^{y=L} - \int_0^L \varphi_i' \varphi_j dy \\ \hat{f}_i(\hat{X}) &\triangleq \alpha_3 \langle f(x), \varphi_i \rangle = \int_0^L f(x) \varphi_i dy \end{aligned} \quad (22b)$$

Note that the spatial boundary conditions for the problem usually get absorbed while evaluating the integrals. Next, in the cost function (10) we observe that

$$\int_0^L q x^2 dy = q \langle x, x \rangle = q \left\langle \left(\sum_{i=1}^{\tilde{N}} \hat{x}_i \Phi_i \right), \left(\sum_{j=1}^{\tilde{N}} \hat{x}_j \Phi_j \right) \right\rangle = \hat{X}^T Q \hat{X} \quad (24)$$

where $Q = q I_{\tilde{N}}$. Similarly we can write

$$\int_0^L r u^2 dy = r \langle u, u \rangle = r \left\langle \left(\sum_{i=1}^{\tilde{N}} \hat{u}_i \Phi_i \right), \left(\sum_{j=1}^{\tilde{N}} \hat{u}_j \Phi_j \right) \right\rangle = \hat{U}^T R \hat{U} \quad (25)$$

where $R = r I_{\tilde{N}}$.

Thus the cost function in (10) can be written as

$$J = \frac{1}{2} \int_0^{\infty} (\hat{X}^T Q \hat{X} + \hat{U}^T R \hat{U}) dt \quad (26)$$

Equations (22a-b) and (26) define an analogous optimal control problem in the lumped parameter framework, which is solved using the SNAC technique (discussed next).

IV. CONTROL DESIGN: SINGLE NEURAL ADAPTIVE CRITIC (SNAC) SYNTHESIS

The focus of this work is on the large class of problems where the control variable \hat{U}_k is explicitly solvable in terms of state variable \hat{X}_k and costate variable λ_{k+1} . A set of neural networks is used to solve the optimal control problem contained in (22a-b) and (26), together with appropriate boundary conditions. This control synthesis is obtained through a set of *critic networks* (retaining the terminology of the *adaptive critic* method [7], [11]).

A. Necessary Conditions of Optimality

The neural network technique requires a discrete time version of these equations. To arrive at such discrete equations it is well-known that the state equation (22a-b) develops forward, while the costate equation develops backward in time. Hence, at any instant of time t_k one can write the discrete-time versions of state and costate equations as

$$\hat{X}_{k+1} = \hat{F}_d(\hat{X}_k, \hat{U}_k) \quad (27)$$

$$\lambda_k = \hat{G}_d(\hat{X}_k, \hat{U}_k, \lambda_{k+1}) \quad (28)$$

where \hat{F}_d and \hat{G}_d are the resulting algebraic functions of their arguments. The details of this process have been omitted for brevity. The discrete version of the optimal control equation can be written as

$$\hat{U}_k = -R^{-1} B^T \lambda_{k+1} \quad (29)$$

Note that λ_{k+1} is substituted for $\lambda(t)$ to make the equation compatible with the discrete optimal control theory [2].

B. State Generation for Neural Network Training

Note that the lumped parameter states can be computed from $x(y)$ in (19a) as follows

$$\hat{x}_j = \langle x(y), \varphi_j(y) \rangle \quad (30)$$

Hence, in principle one can use the method adopted for generation of the initial conditions (see Section III.C) to generate a number of such profiles and then use (30) for $j=1, \dots, \tilde{N}$ to come up with appropriate values for \hat{X}_k . However, an alternative (easier) method of generating this lumped parameter state vector for training the networks was followed. All the snapshots were used and the minimum and maximum values for the individual elements of \hat{X}_k were fixed. Let \hat{X}_{\max} and \hat{X}_{\min} denote the vectors of maximum and minimum values for the elements of \hat{X}_k respectively. Then fixing a positive constant $0 \leq C_i \leq 1$, the idea is to select $\hat{X}_k \in C_i [\hat{X}_{\min}, \hat{X}_{\max}]$. Let $S_i = \{ \hat{X}_k : \hat{X}_k \in C_i [\hat{X}_{\min}, \hat{X}_{\max}] \}$. One can notice that for $C_1 \subseteq C_2 \subseteq \dots$, $S_1 \subseteq S_2 \subseteq \dots$. Thus, for some $i=I$, $C_i=1$ and S_i will include the domain of interest for initial conditions. Hence, to begin the synthesis procedure, a small value for the constant C_1 was chosen and the networks

were trained for the states, randomly generated within S_i . The constant C_i was increased this way until the set S_i included the domain of interest for the initial conditions. In this paper, $C_1 = 0.05$, $C_i = C_1 + 0.05(i-1)$ for $i = 2, 3, \dots$ and continued until $i = I$, where $C_I = 1$.

C. Selection of Network Structure and Initial Weights

In this work the network was split internally into \tilde{N} sub-networks, assuming one network (rather a sub-network) for each channel of the costate vector. The input to each sub-network, however, is the entire state vector X_k . Since $\tilde{N} = 5$ (for this problem), five feedforward $\pi_{5,5,1}$ neural networks were used. A $\pi_{5,5,1}$ neural network means five neurons in the input layer, five neurons in the hidden layer and one neuron in the output layer. For activation functions, *tangent sigmoid* function was used for all the hidden layers and a *linear* function was used for the output layer. No optimization was carried out for the ‘best’ neural architecture. Numerical results (see Section V), however, indicate that the choice was appropriate to solve the problem that we were interested in.

Next, appropriate values of initial weights of the networks need to be chosen. For this purpose, first one can notice that if the system is square (i.e. dimensions of \hat{U}_k and λ_{k+1} are same, which is the case for this problem), then from (29) one can write

$$\lambda_{k+1} = -[B^T]^{-1} R \hat{U}_k \quad (31)$$

Hence, if one can come up with some stabilizing control solution for \hat{U}_k (by any method), then (31) can be used to compute the corresponding λ_{k+1} , ‘assuming’ that control solution to be optimal. Then the relationship between the corresponding \hat{X}_k and λ_{k+1} can be used to train (rather pre-train) the networks, before starting the SNAC training process. This is what has been done in this paper, using dynamic inverse technique [4] to generate \hat{U}_k .

D. Training of Neural Networks

The non-optimal neural network (but initialized with respect to a stabilizing solution, as discussed in Section IV.C) is subjected to the optimal control equations and trained appropriately. The training process with the target quantity is presented in Figure 2. Note that Figure 2 is self-explanatory to outline the steps involved in the neural network training process. We omit the details for lack of space (see [7], [8] for details).

The Levenberg-Marquardt method [5] was used in the back propagation technique for training. 2000 input-output data points for each S_i were chosen. After training the networks with 2000 data points for 25 epochs, the networks were checked for convergence with another 2000 different data points. If the convergence condition was met, the

networks were trained again with a different set of 2000 data points in S_{i+1} and so on. Otherwise, the training process was repeated with different sets of data points within S_i .

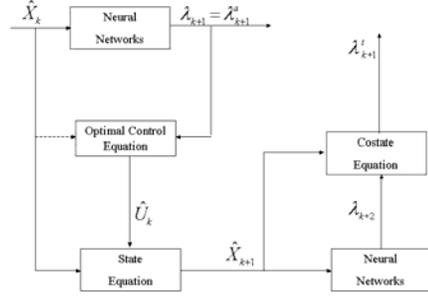


Figure 2: Schematic for Single Neural Network Adaptive Critic

E. Online Computation of the Control Solution

After the controller synthesis, the following steps are used for computing the controller online:

- Measure the state variables at time t_k across the spatial dimension. Find the error of the state with respect to the desired steady-state profile using (7a)
- Compute \hat{X}_k using the basis functions.
- Using \hat{X}_k in the neural network, compute \hat{U}_k .
- Get the desired (spatially-continuous) control $u(t_k, y)$ from \hat{U}_k , using the basis functions, from (19b).
- Get $S(t_k, y)$ from $u(t_k, y)$ using (7b)

V. NUMERICAL RESULTS

Two representative simulations are presented in this section. Figures 3 demonstrates this property for a cooling fin with an initial temperature profile (assumed be an exponentially decaying function from root to tip). It was verified that the desired temperature profiles and the system’s final temperature profiles are almost identical after the 120 second simulations. The associated control history is as shown in Figure 4. The relatively large magnitude of the controller is due to the short simulation time of 120 seconds (chosen to be small due to re-entry time limitation). In other applications where a longer simulation time is permissible, the magnitude of the controller will be less.

As pointed out in Section I, the SNAC technique retains all the advantages of adaptive critic technique and is capable of finding the solutions for a large number of initial conditions. To demonstrate this, simulations have been carried out for a large number of ‘random’ initial conditions, generated following the approach presented in Section III.C. Figure 5 shows the result from such an initial condition. Once again this figure demonstrates that the goal is achieved without any problem, if the associated control (Figure 6) is applied.

VI. CONCLUSIONS

A nonlinear optimal re-entry temperature control problem is solved. All forms of heat transfer (namely conduction, convection and radiation) were accounted for, which results in a nonlinear distributed parameter model. This nonlinear model was used without any approximation (like linearization) to come up with an optimal controller using the single network adaptive critic (SNAC) technique. The simulation results demonstrate that by using the proposed technique, one can track a desired temperature profile, starting from any initial condition in the domain of interest.

ACKNOWLEDGMENT

This research was supported by NSF-USA grant 0201076 and 0324428

REFERENCES

- [1] Annaswamy A., Choi J. J. and Sahoo D., Active Closed-loop Control of Supersonic Impinging jet Flows Using POD Models, *Proceedings of the 41st IEEE Conf on Decision and Control, Las Vegas, NV, 2002*, pp. 3294-3299.
- [2] Bryson A. E. and Ho Y. C., *Applied Optimal Control*, London: Taylor and Francis, 1975
- [3] Burns J. and King B. B., A Reduced Basis Approach to the Design of Low-order Feedback Controllers for Nonlinear Continuous Systems, *Journal of Vibration and Control*, Vol. 4, No. 3, 1998, pp. 297-323.
- [4] Enns D., Bugajski D., Hendrick R. and Stein G., Dynamic Inversion: An Evolving Methodology for Flight Control Design, *International Journal of Control*, Vol.59, No.1,1994, pp.71-91.
- [5] Hagan M. T., Demuth H. B. and Beale M., *Neural Network Design*, PWS Publishing Company, 1996.
- [6] Miller A. F., *Basic Heat and Mass Transfer*, Richard D. Irwin Inc., Concord, MA, 1995.
- [7] Padhi R., Unnikrishnan N. and Balakrishnan S. N., Optimal Control Synthesis of a Class of Nonlinear Systems Using Single Network Adaptive Critics, *American Control Conference*, 2004.
- [8] Padhi R. and Balakrishnan S. N., Proper Orthogonal Decomposition Based Optimal Neurocontrol Synthesis of a Chemical Reactor Process Using Approximate Dynamic Programming, *Neural Networks*, Vol. 16, No. 5-6, 2003, 719-728.
- [9] Ravindran S. S., Proper Orthogonal Decomposition in Optimal Control of Fluids, *NASA/TM-1999-209113*.
- [10] Singh S. N., Myatt J. H. and Addington G. A., Adaptive Feedback Linearizing Control of Proper Orthogonal Decomposition Nonlinear Flow Models, *Proceedings of the American Control Conference*, 2001, pp. 1533-1538.
- [11] Werbos, P. J., Approximate dynamic programming for real-time control and neural modeling, in White D. A., and Sofge D. A (Eds.), *Handbook of Intelligent Control*, Multiscience Press, 1992.

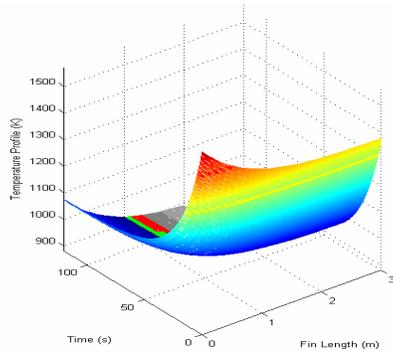


Figure 3: Temperature Time History

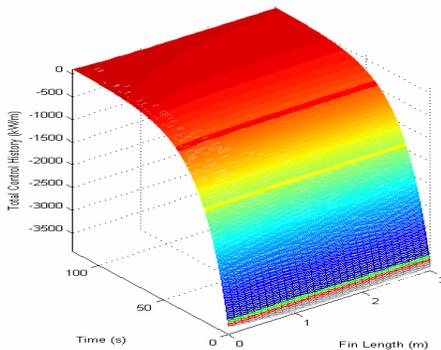


Figure 4: Control Time History

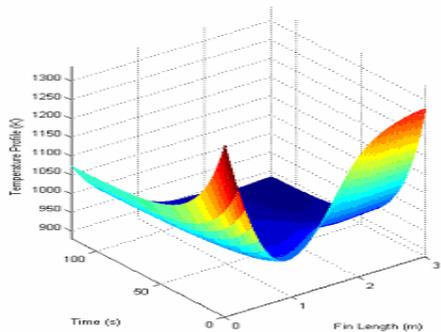


Figure 5: Temperature Time History

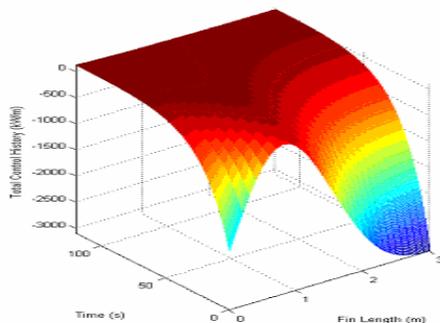


Figure 6: Control Time History