01 Jan 2006

Optimal Management of Beaver Population using a Reduced-Order Distributed Parameter Model and Single Network Adaptive Critics

Radhakant Padhi

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

Recommended Citation

R. Padhi and S. N. Balakrishnan, "Optimal Management of Beaver Population using a Reduced-Order Distributed Parameter Model and Single Network Adaptive Critics," *IEEE Transactions on Control Systems Technology*, Institute of Electrical and Electronics Engineers (IEEE), Jan 2006.
The definitive version is available at https://doi.org/10.1109/TCST.2006.876633

Optimal Management of Beaver Population Using a Reduced-Order Distributed Parameter Model and Single Network Adaptive Critics

Radhakant Padhi, *Member, IEEE,* and S. N. Balakrishnan

***Abstract*—Beavers are often found to be in conflict with human interests by creating nuisances like building dams on flowing water (leading to flooding), blocking irrigation canals, cutting down timbers, etc. At the same time they contribute to raising water tables, increased vegetation, etc. Consequently, maintaining an optimal beaver population is beneficial. Because of their diffusion externality (due to migratory nature), strategies based on lumped parameter models are often ineffective. Using a distributed parameter model for beaver population that accounts for their spatial and temporal behavior, an optimal control (trapping) strategy is presented in this paper that leads to a desired distribution of the animal density in a region in the long run. The optimal control solution presented, imbeds the solution for a large number of initial conditions (i.e., it has a feedback form), which is otherwise nontrivial to obtain. The solution obtained can be used in real-time by a nonexpert in control theory since it involves only using the neural networks trained offline. Proper orthogonal decomposition-based basis function design followed by their use in a Galerkin projection has been incorporated in the solution process as a model reduction technique. Optimal solutions are obtained through a "single network adaptive critic" (SNAC) neural-network architecture.**

***Index Terms*—Beaver population control, distributed parameter control, proper orthogonal decomposition, single network adaptive critic (SNAC), wildlife management.**

I. Introduction

BEAVERS are a small mammal species and have a strong tendency to create nuisances, mainly by building dams on the flowing water, thereby creating flooding in the low land areas, roads, crop lands, etc. However, the same activities sometimes lead to desirable consequences as well—like increased vegetation, increased water table, etc. Because of this conflicting situation, an optimal management strategy is needed to control their population in any particular area. In the northern United States, beavers are usually desired [25], whereas in the southern United States they are known for their nuisance activities [5]. Their nuisance activities have also been studied beyond the United States [13]. This fascinating species still keeps on attracting researchers to study about their population and impact [19], [20].

Managing the beaver population is difficult due to their migratory nature. Some of the reasons for this are better availability of food and water resources, the desire of two-year olds to set up new colonies, etc. [5]. Because of this migratory nature, any controlling (harvesting) technique specific to the property of a particular land owner should also account for the beaver population dynamics in the surroundings, as well as the need of the neighboring land owners. In mathematical terms, this introduces the need for a distributed model to design an appropriate management strategy. Awareness of the need for a proactive beaver management policy is slowly increasing [19] and, hence, implementing such a strategy may be less difficult as people in a region may cooperate with the decision of the wildlife manager more willingly. With the assumption that the neighboring land owners have a common goal, a distributed parameter model has been proposed in [5]. An optimal harvesting strategy using this model has also been proposed [5], [17].

In contrast to the lumped parameter systems, distributed parameter systems (DPS) are governed by a set of partial differential equations (PDEs) and are also known as infinite-dimensional systems because of the presence of infinite system modes. Control of distributed parameter systems has been studied both from mathematical, as well as the engineering point of view. An interesting historical perspective of the control of such systems can be found in [16]. There exists an infinite amount of dimensional operator theory-based methods for the control of distributed parameter systems. While there are many advantages, these operator theory-based approaches are mainly limited to linear systems [11] and some limited classes of nonlinear problems like spatially invariant systems [2]. Moreover, for implementation purpose the infinite-dimensional control solution needs to be approximated (e.g., truncating an infinite series, reducing the size of feedback gain matrix, etc.) and, hence, it is not completely free from approximation errors.

It should be noted, however, that while dealing with infinite dimensional systems, approximations need to be introduced at some point or another. An alternate way of dealing with infinite dimensional systems is to have a finite dimensional approximation of the system using a set of orthogonal basis functions via Galerkin projection [14]. Even though such approximation processes obviously introduce errors at the first place (in strict mathematical sense), usually these are adequate to address

Manuscript received December 22, 2003; revised July 28, 2005. Manuscript received in final form February 28, 2006. Recommended by Associate Editor F. Ghorbel. This work was supported by the National Science Foundation under Grant 0201076 and Grant 0324428.

R. Padhi is with the Department of Aerospace Engineering, Indian Institute of Science, 560 012 Bangalore, India (e-mail: padhi@aero.iisc.ernet.in).

S. N. Balakrishnan is with the Department of Mechanical and Aerospace Engineering, University of Missouri-Rolla, Rolla, MO 65409 USA (e-mail: bala@umr.edu).

Digital Object Identifier 10.1109/TCST.2006.876633

practical engineering problems. However, the methodology of Galerkin projection normally leads to high-order lumped system representations to adequately represent the properties of the original system, if arbitrary orthogonal functions (e.g., Legendre polynomials, Chebyshev polynomials, Fourier functions, etc.) are used as the basis functions [23]. For this reason, attention is being increasingly focused in the recent literature on the technique of proper orthogonal decomposition (POD). In this technique, a set of "problem oriented" orthogonal functions is designed to approximately span the solution space of the original system of PDEs. This is done through the "snap shot solutions," which by definition are representative solutions of the system at arbitrary instants of time. Using these basis functions in a Galerkin procedure a very low-order lumped parameter model (i.e., a set of ordinary differential equations) is created that is usually good enough for control design. For linear systems, it has been proved that such an approach of designing the POD-based basis function leads to the optimal representation of the PDE system in the sense that it captures the maximum energy of the system with the least number of basis functions as compared to any other set of orthogonal set of basis functions [14]. For nonlinear systems, even though such useful theorems do not exist, the basic underlying idea still holds. Out of numerous studies published on this topic, we cite [3], [7], [14], [22], and [24] for reference with respect to the application of the technique in various linear and nonlinear distributed parameter problems.

The main goal of this research is to come up with an "optimal" beaver harvesting scheme in the entire region of interest. The obvious choice in this regard is to use optimal control theory. In fact, many difficult real-life lumped parameter problems can be formulated in the framework of optimal control theory [6], [18]. An interested reader can refer to [8] for the application of optimal control techniques for bioeconomic problems. It is well-known that the dynamic programming formulation offers the most comprehensive solution approach to nonlinear optimal control in a state feedback form [6]; a feedback control is desirable because of its beneficial properties like robustness with respect to noise and modeling uncertainties. However, solving the associated Hamilton–Jacobi–Bellman (HJB) equation demands a very large (rather infeasible) amount of computations and storage space dedicated for this purpose. An innovative idea was proposed in [26] to get around this numerical complexity by using an "approximate dynamic programming" (ADP) formulation. The solution to the ADP formulation is obtained through a dual neural-network approach called adaptive critic (AC). In one version of the AC approach, called the dual heuristic programming (DHP), 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. The optimal solution is reached after the two networks iteratively train each other successfully. This DHP process, aided by the nonlinear function approximation capabilities of neural networks, overcomes the computational complexity that had been the bottleneck of the dynamic programming approach. Another important advantage of this method is that this solution can be implemented online, since the control computation requires only a few multiplications of the network weights that are trained offline.

One of the many benefits of a neural-network approach for control design, which has extensively been used in the control of lumped parameter systems [15], is its ability to control nonlinear plants. This paper uses a variant (rather a significant improvement) of the AC architecture and is named "single network adaptive critics" (SNAC). The SNAC architecture retains all the powerful features of the AC technique. However, there is no need of "action" networks and, hence, there is no requirement of iterative training loops between the "action" and "critic" networks, which are necessary in a typical AC procedure. Consequently, it leads to a considerable amount of computational savings besides eliminating the error associated with the additional neural-network approximations.

In this paper, our goal is to design an optimal harvesting strategy so as to achieve a desired beaver population distribution throughout a region under consideration over a long run. Even though many population control techniques are available, like relocating the food resources, poisoning the food resources, etc., in this work the methods of control are trapping and subsequent killing (for pelt value) or relocation in case of a negative control demand elsewhere in the region. The solution uses the ideas of model reduction with proper orthogonal decomposition (POD) technique-based basis functions and the philosophy of SNAC-based optimal control design. The effectiveness of the resulting control is tested from numerous simulations, even though a set of representative results are presented because of space limitations.

Besides solving the application problem under consideration, the content in this paper can be interpreted to have two main developmental contributions: 1) it uses a newly developed SNAC technique, which has significant computational savings as compared to the DHP/AC technique currently used by many neural network researchers and 2) the technique presented also serves as a computational tool to synthesize a feedback form of optimal control solutions for distributed parameter systems, in general. To the best of our knowledge, this is probably the first paper in which a neural-network-based controller has been designed for a two-dimensional (2-D) distributed parameter system. Besides the advantages, the approach followed in this paper differs from that presented in [5] and [17] in the sense that, it is based on a "tracking" formulation and any desired profile can be achieved, subjected to the limitations in control actuation. The numerical results show that the magnitude of error in tracking the required beaver density profile after five years is much smaller than the errors after ten years in the results presented in [5], [17].

The rest of the paper is organized as follows. In Section II, the distributed parameter model for beaver population is presented and the control objective is defined. The steps in the reduced-order model development using POD basis functions are presented in Section III. In Section IV, the SNAC technique is discussed in detail. The numerical results are presented and conclusions are drawn in Sections V and VI, respectively.

II. Beaver Population Model and Controller Objective

A. Beaver Population Model

Assuming the beaver population distribution to be continuous in a territory, the following distributed parameter model has been developed in the literature for beaver population density [5], [17]:

$$\begin{aligned} \frac{\partial Z}{\partial t} &= \alpha\nabla^2 Z + (aZ - bZ^2) - PZ \quad \text{in } \Omega \times (0, t_f) \\ Z(y_1, y_2, t) &= c \quad \text{on } \partial\Omega \times (0, t_f) \\ Z(y_1, y_2, 0) &= Z_0(y_1, y_2) \quad \text{in } \Omega \text{ at } t = 0 \end{aligned} \tag{1}$$

where Z is the beaver population density in heads per square miles (hd/mi^2) and P is the portion of Z to be trapped per year (yr^{-1}), which acts as a control variable. α, a, and b are growth parameters of the model (their meanings and values are in Table I). Note that the term $(aZ - bZ^2)$ represents the density-dependent annual biological productivity of beavers in the absence of dispersion. Assuming that the spatial domain $\Omega \in \mathbb{R}^2$ and it is a rectangle, $\Delta \triangleq \{y = (y_1, y_2) : y_1 \in [0, L_1], y_2 \in [0, L_2]\}$, where L_1 and L_2 are the lengths of its sides. $\partial\Omega$ represents the boundary of Ω and time $t \in (0, t_f)$ and $Z_0(y_1, y_2)$ represents the initial density distribution. Based on a study for the state of New York by its Department of Environmental Conservation, the parameters of the model obtained in [5] and [17] are as given in Table I.

TABLE I
Parameters of the Beaver Population Model

Symbol	Meaning	Units	Value
a	Maximum rate of net recruitment	yr^{-1}	0.335
b	Density dependence of beaver stock	$mi^2\, hds^{-1}\, yr^{-1}$	0.2066315
α	Diffusion coefficient	$mi^2\ yr^{-1}$	725.27

The assumption $c \geq 0$ to be a constant describes the boundary condition. Note that the condition $Z(y_1, y_2, t) = c$ on $\partial\Omega \times (0, t_f)$ in (1) differs from the model in [5] and [17] in the sense that they have assumed $c = 0$. This modification was introduced both to avoid a numerical problem in the control computation (see Section II-B). It also makes better physical sense to have a nonzero boundary condition as the number of beavers is not expected to be exactly zero on the boundary. In our numerical simulations $c = 1$.

It is clear from (1) that the growth (or decay) of the population density is a dynamic process that depends on the parameters of the model. Some of the reasons for the diffusion term in the model include migration of two-year-olds to set up new colonies, migration of the entire colonies for better food availability, etc. Similarly, some of the main reasons for the decay terms include their natural demise, being eaten by predators, diseases due to contaminated water (their habitat is always close to water resources), etc. Even though many types of control techniques are available (e.g., relocating their food resource, poisoning their food habitat, introducing predators, trapping by humans, etc.), the application of a particular technique is dictated by the nature of the problem for a particular region. The model in (1) assumes that there is a collective goal for all of the land owners (which includes the person in charge of the public land) of a particular region of interest and that the owners are willing to act collectively for their mutual benefit.

B. Controller Objective

The objective of this controller in this study is to trap the beavers throughout the territory in an optimal way that leads to a desired distribution $Z^*(y)$ in the long run. This leads to an optimal controller formulation in an infinite-time framework.

1) Choice of the Desired Distribution: The territory considered in this paper is a forest land and the desired distributions wanted by a wildlife manager $Z^*(y)$ are restricted to satisfy the following conditions:

1) $Z^* \geq 0$ in Ω;
2) $Z^* = c$ in $\partial\Omega$;
3) Z^* is continuous and smooth (i.e., $\nabla^2 Z^*$ continuous);
4) $(\nabla^2 Z^*/Z^*)$ is finite whenever $Z^* = 0$.

Condition 1 is assumed because of the fact that $Z^* < 0$ has no physical meaning. Condition 2 is imposed to reflect the problem dynamics. Note that with this condition Z^* satisfies the boundary condition for Z in (1). The necessity of conditions 3 and 4 will be clear from the discussions later in this section [see (3)].

Without loss of generality, it is assumed that $Z^* = c = 1$ throughout Ω as the desired profile. It satisfies all of the conditions mentioned above. However, note that the underlying technique is not constrained by this assumption. If some other profile is desired (satisfying the above condition), the technique presented in this study can achieve that also. The goal of this study is the same as in [5] and [17], in the sense that their studies seeked to achieve a desired profile $Z^* = c = 0$ throughout Ω (note that they have assumed $Z^* = c$ on $\partial\Omega$ as well). In such a case, the tracking problem reduces to a regulator problem, and hence, there is no need of the feedforward controller (condition 4 mentioned above is not needed in that case). However, the technique presented here is capable of solving tracking problems as well, and, hence, is capable of achieving a wider set of objectives as compared to the one presented in [5] and [17].

2) Feedforward Controller: Let P^* be the associated control with Z^* so that Z^* remains at steady state. Then from (1), it is clear that Z^* and P^* should satisfy the following equation:

$$\alpha\nabla^2 Z^* + Z^*(a - bZ^* - P^*) = 0 \tag{2}$$

which leads to

$$P^* = (a - bZ^*) + \alpha\left(\frac{\nabla^2 Z^*}{Z^*}\right). \tag{3}$$

The value of P^*, as obtained from (3), is a feedforward controller that acts in conjunction with the optimal feedback controller to be developed so that a steady-state condition will be maintained after $Z \rightarrow Z^*$. Note that the conditions 3 and 4 imposed on Z^* make P^* well-behaved.

Note that in general, (3) poses a restrictive constraint on the choice of the desired profiles. Unless the desired profiles have sufficiently small curvature, the term $\alpha(\nabla^2 Z^*/Z^*)$ becomes numerically high (mainly because of the high value of α). This leads to a high value of P^*, which is not allowed. Due to the way control is defined, one must guarantee that $P \le 1$ everywhere in the domain and for all time. This implies that we must also have $P^* \le 1$, since in the steady state essentially $P = P^*$. Note that because of our selection of a constant desired profile $Z^* = c = 1$ throughout Ω (including $\partial\Omega$), we have $\nabla^2 Z^* = 0$ the above problem does not arise. It will be clear from the results in Section V that the constraints $P \le 1$ and $P^* \le 1$ are satisfied without any problem.

3) Deviation Dynamics and Cost Function Selection: With the availability of the desired final values for state Z^* and control P^*, $Z \triangleq Z^* + x$ and $P \triangleq P^* + u$, where x and u are deviations in state and control, respectively. Then it follows from (1) that

$$\begin{aligned} \frac{\partial x}{\partial t} &= \alpha\nabla^2 x + (a - P^* - 2bZ^* - bx)x - (Z^* + x)u \\ x(y_1, y_2, t) &= 0 \text{ on } \partial\Omega \\ x(y_1, y_2, 0) &= Z_0(y_1, y_2) - Z^*(y_1, y_2) \text{ on } \Omega. \end{aligned} \tag{4}$$

The goal of the controller design now is to cancel the deviation terms x and u throughout the domain. This can be achieved by finding a controller that minimizes

$$J = \frac{1}{2}\int_0^{t_f\to\infty}\int_0^{L_1}\int_0^{L_2}(qx^2 + ru^2)dy_2 dy_1 dt \tag{5}$$

where $q \ge 0$ and $r > 0$ are the weights on state and control, respectively. The selection of the cost function as in (5) along with the nonlinear dynamics of the problem leads us to a nonlinear quadratic regulator problem, for which closed-form solution for control does not exist (to the best knowledge of the authors). Hence, an adaptive critic neural-network-based numerical approach is used to find solutions in a domain of interest.

Note that ideally, one should also impose a constraint $\int_\Omega P\ dy \ge 0$, to address an implementation constraint as beavers cannot be created at will. However, such a restriction is not used with the assumption that should such a case arise, the beavers can be bought in from a different territory. Even though this assumption is made for mathematical tractability, this condition was not violated in any of our numerous simulation studies.

III. Reduced-Order Model Development

Distributed parameters are otherwise known as "infinite dimensional systems" because of the presence of an infinite number of system modes. Because it is impossible to deal with an infinite dimensional system (either for control computation or its implementation), some approximation needs to be introduced while dealing with such systems. In this research, "proper orthogonal decomposition" based basis function design is used followed by their use in a Galerkin projection, which leads to a very low-dimensional representation of the system with a good level of accuracy [3], [7], [14], [22], [24]. This powerful technique has become widely popular in recent years. In this section, key steps of this procedure are outlined.

A. Basis Function Design Based on Proper Orthogonal Decomposition (POD)

Proper orthogonal decomposition (POD) is a technique of finding an optimal set of basis functions, which spans an ensemble of data optimally in an average sense. Let $\{U_i(y) : 1 \le i \le N,\ y \in \Omega\}$ be an ensemble of data, consisting of set of N *snapshot* solutions (observations), of some physical process over the domain Ω at arbitrary instants of time. The process of getting the snapshot solutions for the beaver control problem will be discussed later in Section III-D. The goal of the POD technique is to design a coherent structure that has the largest mean square projection on the snapshots. In other words, it is an effort to find all such possible basis functions Φ, each of which provides a local maximum for the following figure of merit

$$I = \frac{1}{N\langle\Phi,\Phi\rangle}\sum_{i=1}^{N}|\langle U_i,\Phi\rangle|^2. \tag{6}$$

The standard L_2 inner product is used in (6), which is defined as $\langle\Phi_1,\Phi_2\rangle = \int_\Omega \Phi_1\Phi_2 dy$. The idea then is to seek a basis function Φ as a linear combination of snapshots of the form

$$\Phi = \sum_{i=1}^{N} w_i U_i \tag{7}$$

where the coefficients w_i are to be determined such that Φ maximizes the cost function in (6). However, we notice that

$$\begin{aligned} &\frac{1}{N}\sum_{i=1}^{N}|\langle U_i,\Phi\rangle|^2 \\ &= \frac{1}{N}\sum_{i=1}^{N}\int_\Omega\int_\Omega U_i(y)\Phi(y)U_i(y')\Phi(y')dydy' \\ &= \int_\Omega\left(\frac{1}{N}\sum_{i=1}^{N}\int_\Omega U_i(y)U_i(y')\Phi(y')dy'\right).\Phi(y)dy \\ &= \langle K\Phi,\Phi\rangle \end{aligned} \tag{8}$$

where the operator K is defined as

$$K\Phi \equiv \frac{1}{N}\sum_{i=1}^{N}\int_\Omega U_i(y)U_i(y')\Phi(y')dy'. \tag{9}$$

Defining $\sigma \triangleq \langle K\Phi,\Phi\rangle/\langle\Phi,\Phi\rangle$, it can be seen from (6) and (8) that the objective now is to find a function Φ^* that maximizes σ. Following the principle of calculus of variations it can be shown that this problem leads to $K\Phi^* = \sigma\Phi^*$, subjected to $\|\Phi^*\| = 1$. Substituting for Φ and $K\Phi$ from their definitions in (7) and (9), after some algebra it can be shown that $CW = \sigma W$, where $C = [c_{ij}]$, $c_{ij} = (1/N)\int_\Omega U_i(y)U_j(y)dy$ and $W = [w_1,\ldots,\ w_N]^T$. An interested reader can refer to [14] and [22] for more mathematical details. This is now a standard matrix eigenvalue and eigenvector problem to find W. It is clear that $c_{ij} = c_{ji}$, i.e., C matrix is symmetric. Furthermore, from the definition of σ, observe that $\sigma \ge 0$. So C matrix has N *nonnegative real* eigenvalues and N orthogonal eigenvectors. The eigenvectors can be sorted in a descending order as $\sigma_1 \ge \sigma_2 \ge,\ldots,\ge \sigma_N \ge 0$. Let the corresponding eigenvectors be $W^1 = [w_1^1, \ldots, w_N^1]^T, \ldots, W^N = [w_1^N, \ldots, w_N^N]^T$. The N basis functions can be written as

$$\Phi_1 = \sum_{i=1}^{N} w_i^1 U_i(y),$$
$$\vdots$$
$$\Phi_N = \sum_{i=1}^{N} w_i^N U_i(y). \tag{10}$$

The $\|\Phi\| = 1$ condition is met when W^js are normalized by forcing $\langle W^j, W^j \rangle = 1/(N\sigma_j)$ with the inner product being taken in a standard l_2 (vector) notation. The eigenspectrum can then be truncated judiciously such that $\sum_{j=1}^{\tilde{N}} \sigma_j \approx \sum_{j=1}^{N} \sigma_j$, where the truncated system has $\tilde{N} \leq N$ eigenvalues and eigenvectors. In that case, the $\tilde{N}$ orthonormal eigenfunctions approximately span the solution space. Usually it turns out that $\tilde{N} \ll N$. An important property of the POD basis functions is that they are optimal in the sense that for a given number of modes n, the projection on the subspace spanned by leading n POD basis functions contains the greatest possible energy on an average sense [14], [22].

B. Reduced Order Model: Use of Galerkin Projection

After obtaining the basis functions, x and u are expanded as follows:

$$x(t, y) = \sum_{j=1}^{\tilde{N}} \hat{x}_j(t)\Phi_j(y_1, y_2)$$
$$u(t, y) = \sum_{j=1}^{\tilde{N}} \hat{u}_j(t)\Phi_j(y_1, y_2). \tag{11}$$

Before we proceed further, the following needs to be pointed out.

- The snapshots are assumed to be spread out symmetric about zero (since the formulation is based on deviation dynamics).
- The same basic functions are used in the expansions of x and u.
- Once the functional relationship between $\hat{x}_j$ and $\hat{u}_j, \forall j = 1, 2, \ldots, \tilde{N}$, are captured, the desired feedback controller can be easily computed.
- Because all Φ_js are orthonormal, one can also notice that $\langle x, \Phi_i \rangle = \hat{x}_i(t)$. This fact is required for the neural-network synthesis.
- The principle of Galerkin projection [14] is used after substituting (11) into (4) to obtain the following reduced-order finite-dimensional model for the deviation dynamics:

$$\dot{\hat{X}} = \hat{A}(\hat{X}) \cdot \hat{X} + \hat{B}(\hat{X}) \cdot \hat{U} \tag{12}$$

where $\hat{X} \triangleq [\hat{x}_1 \ldots \hat{x}_{\tilde{N}}]^T$, $\hat{U} \triangleq [\hat{u}_1 \ldots \hat{u}_{\tilde{N}}]^T$. $\hat{A}(\hat{X})$ and $\hat{B}(\hat{X})$ are defined as

$$\hat{A}(\hat{X}) \triangleq \hat{A}_1 + b\hat{A}_2(\hat{X})$$
$$\hat{B}(\hat{X}) \triangleq \hat{B}_1 + \hat{A}_2(\hat{X}) \tag{13a}$$

where for $n, j = 1, \ldots, \tilde{N}$, the matrices $\hat{A}_1$, $\hat{B}_1$, and $\hat{A}_2(\hat{X})$ are defined as follows:

$$\hat{A}_{1_{nj}} \triangleq -\int_0^{L_1}\int_0^{L_2} [\nabla\Phi_n \cdot \nabla\Phi_j + (2bZ^* + P^*)\Phi_n\Phi_j]\, dy_2 dy_1 + a\delta_{nj}$$
$$\hat{B}_{1_{nj}} \triangleq -\int_0^{L_1}\int_0^{L_2} Z^*\Phi_n\Phi_j dy_2 dy_1 \tag{13b}$$
$$\hat{A}_{2_{nj}} \triangleq -\int_0^{L_1}\int_0^{L_2} x\Phi_j\Phi_j dy_2 dy_1.$$

In (13b), δ_{nj} is the "delta function," which means $\delta_{nj} = 1$, only if $n = j$. Otherwise, $\delta_{nj} = 0$. Note that even though (12) is simply written in "linear-looking" form. It is still a nonlinear equation and no approximation (like linearization) has been used here.

Next, substituting for x and u from (11) in the cost function expression (5), we get

$$J = \frac{1}{2}\int_0^{\infty} (\hat{X}^T\hat{Q}\hat{X} + \hat{U}^T\hat{R}\hat{U})dt \tag{14}$$

where $\hat{Q} = qI_{\tilde{N}}$ and $\hat{R} = rI_{\tilde{N}}$.

Equations (12)–(14) represent a nonlinear quadratic regulator optimal control problem. This problem is solved using the SNAC technique (see Section IV).

C. Domain of Interest and Selection of Initial Conditions

It is well known that the Fourier series is a universal function approximator. So, any possible initial condition can be written in the following form:

$$x(0, y_1, y_2) = \sum_{j=1}^{N_1}\sum_{k=1}^{N_2} b_{jk}\Psi_{jk} = \sum_{j=1}^{N_1}\sum_{k=1}^{N_2} b_{jk}\left[\frac{2}{\sqrt{L_1L_2}} Sin\left(\frac{j\pi y_1}{L_1}\right) Sin\left(\frac{k\pi y_2}{L_2}\right)\right] \tag{15}$$

where $\Psi_{jk} = (2/\sqrt{L_1L_2})\mathrm{Sin}(j\pi y_1/L_1)\mathrm{Sin}(k\pi y_2/L_2)$ is a Fourier basis function for mode-jk. Typically, N_1, N_2 are used-defined sufficiently large positive integers. Values of $N_1 = 4$ and $N_2 = 5$ are used to produce $N_1N_2 = 20$ possible modes. Any other higher values for N_1, N_2 should also work fine. Note that $\langle\Psi_{jk}, \Psi_{lm}\rangle = \delta_{jk}\delta_{lm}$, where δ_{jk}, δ_{lm} are *delta-functions*.

With the selection of the Fourier expansion for x as in (15), the following equations can be derived (after some algebra):

$$\langle x, x\rangle = \sum_{j=1}^{N_1}\sum_{k=1}^{N_2} b_{jk}^2 \tag{16}$$

$$\|x'\|^2 \triangleq \int_0^{L_1}\int_0^{L_2} (\nabla x \cdot \nabla x) dy_2 dy_1 = \sum_{j=1}^{N_1}\sum_{k=1}^{N_2} b_{jk}^2 \left[(j\pi/L_1)^2 + (k\pi/L_2)^2\right] \tag{17}$$

$$\langle \nabla^2 x, \nabla^2 x\rangle = \sum_{j=1}^{N_1}\sum_{k=1}^{N_2} b_{jk}^2 \left[(j\pi/L_1)^4 + (k\pi/L_2)^4\right]. \tag{18}$$

Next, an "envelope profile" was selected, which has the following form:

$$x_{\text{env}}(y_1, y_2) = A_{\text{env}} \text{Sin}(\pi y_1/L_1) \text{Sin}(\pi y_2/L_2). \tag{19}$$

With the selection of this expression for x_{env}, after some algebra, the following can be derived:

$$\|x_{\text{env}}\|^2 = \langle x_{\text{env}}, x_{\text{env}}\rangle = A_{\text{env}}^2 (L_1 L_2/4) \tag{20}$$

$$\|x'_{\text{env}}\|^2 \triangleq \int_0^{L_1}\int_0^{L_2} (\nabla x_{\text{env}} \cdot \nabla x_{\text{env}}) dy_2 dy_1 = A_{\text{env}}^2 \pi^2 \left(1/L_1^2 + 1/L_2^2\right)(L_1 L_2/4) \tag{21}$$

$$\|\nabla^2 x_{\text{env}}\|^2 = \langle \nabla^2 x_{\text{env}}, \nabla^2 x_{\text{env}}\rangle = A_{\text{env}}^2 \pi^4 \left(1/L_1^4 + 1/L_2^4\right)(L_1 L_2/4). \tag{22}$$

Now, the following set is defined as the domain of interest:

$$S = \Big\{ x_0(y_1, y_2) : \|x_0\|^2 \le \|x_{\text{env}}\|^2, \|x'_0\|^2 \le k_1 \|x'_{\text{env}}\|^2, \|\nabla^2 x_0\|^2 \le k_2 \|\nabla^2 x_{\text{env}}\|^2 \Big\} \tag{23}$$

where k_1, k_2 are positive constants. In the numerical experiments conducted, we selected $A_{\text{env}} = 1$, $k_1 = 2$, and $k_2 = 4$. Any profile in set S can be deemed as a feasible initial condition for our problem. By using (15)–(22), an algorithm to generate the initial conditions that satisfy the inequality constraints in (23) was generated. The details of this algorithm are omitted here for brevity.

D. Snapshot Solution Generation

Snapshot solutions, which should be representative state solutions under the application of "a reasonably good control," are the key to the success in finding a proper reduced-order model. It is a common practice to assume an open-loop control for this purpose [22]. However, a feedback solution will be better for our application, since that is what is aimed in this paper. The following procedure is used to generate snapshots.

The spatial domain $\Omega \backslash \partial\Omega$ is discretized denoting $m_1 = 1, \ldots, M_1$ as the node points along y_1 and $m_2 = 1, \ldots, M_2$ as the node points along y_2. Then, for $m_1 = 2, \ldots, (M_1 - 1)$ and $m_2 = 2, \ldots, (M_2 - 1)$, the following ordinary differential equations can be written [12]:

$$\begin{aligned} \dot{x}_{m_1,m_2} = \alpha &\left[\frac{1}{\Delta y_1^2}\left(x_{m_{1+1},m_2} - 2x_{m_1,m_2} + x_{m_{1-1},m_2}\right) + \frac{1}{\Delta y_2^2}\left(x_{m_1,m_{2+1}} - 2x_{m_1,m_2} + x_{m_1,m_{2-1}}\right)\right] \\ &+ \left(a - 2bZ_{m_1,m_2}^* - P_{m_1,m_2}^*\right) x_{m_1,m_2} \\ &- Z_{m_1,m_2}^* U_{m_1,m_2} - bx_{m_1,m_2}^* \\ &- x_{m_1,m_2} U_{m_1,m_2}. \end{aligned} \tag{24}$$

It is also observed that $x_{m_1,m_2} = 0$ for either $m_1 = 1, M_1$, or $m_2 = 1, M_2$ for all time t (because of the boundary conditions). By defining

$$X \triangleq \left[[x_{2,2} \ldots x_{M_1-1,2}]^T \cdots [x_{2,M_2-1} \ldots x_{M_1-1,M_2-1}]^T \right]$$

and

$$U \triangleq \left[[u_{2,2} \ldots u_{M_1-1,2}]^T \cdots [u_{2,M_2-1} \ldots u_{M_1-1,M_2-1}]^T \right]$$

the following finite-dimensional approximated system dynamics can be written as:

$$\dot{X} = A(X) \cdot X + B(X) \cdot U \tag{25}$$

where the matrices $A(X), B(X)$ are appropriately defined (we have omitted the detailed expressions for brevity). Next, the cost function was also approximated using this discretized system in the form

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

where $Q = (1/2) q (\Delta y_1 \Delta y_2) I_{(M_1-2)(M_2-2)}$ and $R = (1/2) r (\Delta y_1 \Delta y_2) I_{(M_1-2)(M_2-2)}$. Even though the dimensions are high, nevertheless, (25)–(26) form an analogous nonlinear quadratic regulator optimal control problem in a finite-dimensional framework.

Note that the system dynamics in (25) naturally comes out to be in the state dependent coefficient (SDC) form and the cost function in (26) is quadratic. In addition, it is an infinite time formulation. This is a classic case for applying the recently-developed state dependent Riccati equation (SDRE) technique. The essential idea in this technique is to treat the problem as a linear quadratic regulator (LQR) problem and solve the associated algebraic Riccati equation [6], [18] at each instant of time, to find out a time-varying Riccati matrix $P(t)$. Then, gain matrix is computed as $K(t) = R^{-1} B^T(X) \cdot P(t)$ and the control solution is given by $U(t) = -K(t)X$. An interested reader can see [9], [10], and [21] for more details about the SDRE technique. A key point to note, however, is that this technique can only lead to a "suboptimal" controller. This happens mainly because even though the state and optimal control equations are satisfied for all time, the associated costate equation is satisfied only asymptotically [9], [21]. On the other hand, the SNAC technique used here leads to obtaining an "optimal" controller, since all three equations are satisfied for all time.

We have used this SDRE technique to synthesize a suboptimal feedback nonlinear controller for the purpose of collecting snapshot solutions. Taking an arbitrary initial condition (as outlined in Section III-C), the system is simulated for $t_f = 3$ yr. Once the solution is obtained for a particular initial condition, representative solutions are collected at arbitrary instants of time and assumed as representative snapshot solutions. In this research, the problem was solved using a gradient technique for 50 different initial conditions and 20 random solutions were collected from each of them, thereby collecting 1000 snapshots in total. The POD basis functions are designed with these snapshot solutions are subsequently used in a Galerkin projection to obtain a reduced-order representation of the model and cost function (see Sections III-A and III-B).

IV. Single Network Adaptive Critics (SNAC)

In this section, a neural-network structure is proposed for solving the optimal control problem. This control synthesis is essentially obtained through, what is called a set of *critic networks*. This is to retain the terminology of the *adaptive critic* methodology outlined earlier in [1] and [26]. However, the need of the *action networks* in the adaptive critic framework is eliminated. Hence this methodology can be considered as a good improvement of the adaptive critic technique and it is called the *single network adaptive critic (SNAC)*.

A. Optimality Conditions

The necessary conditions of optimality for a lumped system driven by the system dynamics in (12) and cost function in (14), are well known [6], [18]. First, a *Hamiltonian* variable is defined as

$$H = \frac{1}{2}(\hat{X}^T\hat{Q}\hat{X} + \hat{U}^T\hat{R}\hat{U}) + \lambda^T\left[\hat{A}(\hat{X})\cdot\hat{X} + \hat{B}(\hat{X})\cdot\hat{U}\right] \tag{27}$$

where λ is the Lagrange multiplier variable. Then the optimal control equation is given by

$$\frac{\partial H}{\partial \hat{U}} = 0 \tag{28}$$

which leads to

$$\hat{U} = -\hat{R}^{-1}\hat{B}^T(\hat{X})\cdot\lambda. \tag{29}$$

The costate equation is given by

$$\dot{\lambda} = -\left(\frac{\partial H}{\partial \hat{X}}\right) = \quad -\left(\hat{Q}\hat{X} + \hat{A}^T(\hat{X})\cdot\lambda\right) \tag{30}$$

(12), (29), and (30) need to be solved simultaneously, along with the boundary conditions for optimal control with $\hat{X}(0)$ is known and $\lambda(t_f \to \infty) = 0$.

It should be noted, however, that the adaptive critic methodology using a set of neural networks needs a discrete version of the problem. At any instant of time k, the discrete versions of state and costate equations are

$$\hat{X}_{k+1} = \hat{F}_d(\hat{X}_k, \hat{U}_k) \tag{31}$$

$$\lambda_k = \hat{G}_d(\hat{X}_k, \hat{U}_k, \lambda_{k+1}) \tag{32}$$

where $\hat{F}_d$ and $\hat{G}_d$ are the resulting algebraic functions of their arguments. From (39), the optimal control equation can be written as

$$\hat{U}_k = \xi(\hat{X}_k, \lambda_{k+1}) \tag{33}$$

where ξ is the resulting *explicit* algebraic function of its arguments. A fourth-order Runge–Kutta method [12], with constant step size in time $\Delta t = 1$ week $= 0.0192$ yr, was used in this study.

B. Neural Network Synthesis Process

1) State Generation for Training: In the controller synthesis process, a set of states for training needs to be chosen so that its elements approximately cover the states that are possible initial conditions in the *domain of interest*. S should also contain all possible states of the controlled system as they evolve from different initial conditions. Once the snapshot solutions are generated and POD basis functions are designed, the minimum and maximum values for the individual elements of $\hat{X}_k$ can be fixed as follows.

Let $\hat{X}_{\max}$ denote the vector of maximum values for $\hat{X}_k$ and $\hat{X}_{\min}$ the vector for minimum values. Then, fixing a positive constant $0 \le c_i \le 1$, the states $\hat{X}_k \in c_i[\hat{X}_{\min}, \hat{X}_{\max}]$ are selected. Let $S_i = \{\hat{X}_k : \hat{X}_k \in c_i[\hat{X}_{\min}, \hat{X}_{\max}]\}$. Then, for $c_1 \le c_2 \le c_3 \le \ldots, S_1 \subseteq S_2 \subseteq S_3 \subseteq \ldots$. Hence, for some $i = I$, $c_I = 1$ and S_I will include the domain of interest for initial conditions. At the beginning a small value for the constant c_1 is fixed and the networks is trained with these states, randomly generated within S_1. Once the critic networks converge for this set, a value of c_2 close to c_1 is picked and the network training is continued for the profiles within S_2 and so on, until the set S_i includes domain of interest for the initial conditions. In this work, the values of the constants are updated progressively from $c_1 = 0.05, c_i = c_1 + 0.05\,(i-1)$ for $i = 2, 3, \ldots$, until $i = I$, where $c_I = 1$.

2) Training Procedure: Since neural networks are capable of doing universal function approximations [4], the idea here is to capture the optimal relationship between the state $\hat{X}_k$ and co-state λ_{k+1} using a neural network. For faster training, however, we have synthesized separate networks for each element of the vector λ_{k+1}. Since $\hat{U}_k$ is an explicit function of $\hat{X}_k$ and λ_{k+1}, optimal control $\hat{U}_k$ can be calculated from (33).

The SNAC training algorithm is described in the following steps [Fig. 1].

1) Fix c_i and generate S_i.
2) For each element $\hat{X}_k$ of S_i follow the steps below.
   - Input $\hat{X}_k$ to the network(s) to get λ_{k+1}. Denote the actual output as λ^a_{k+1}.
   - Calculate $\hat{U}_k$, knowing $\hat{X}_k$ and λ_{k+1}, from *optimal control equation* (33).
   - Get $\hat{X}_{k+1}$ from the *state equation* (31), using $\hat{X}_k$ and $\hat{U}_k$.
   - Input $\hat{X}_{k+1}$ to the network(s) to get λ_{k+2}.
   - Calculate λ_{k+1}, form the *costate equation* (32). Let us denote this as λ^t_{k+1}.
3) Train the network(s), with all $\hat{X}_k$ as input and all corresponding λ^t_{k+1} as output.

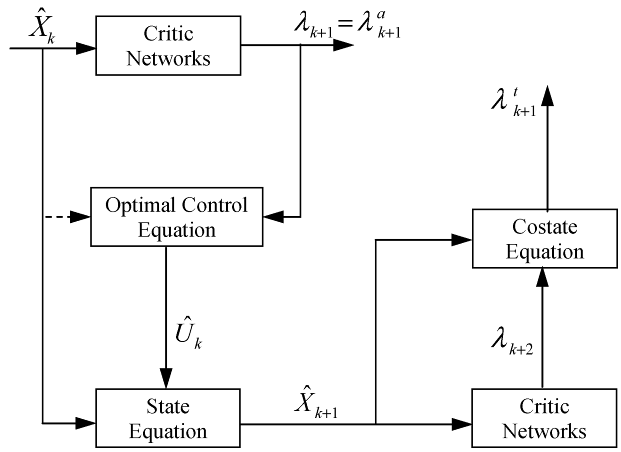


Fig. 1. Schematic of simplified adaptive critic neural-network synthesis.

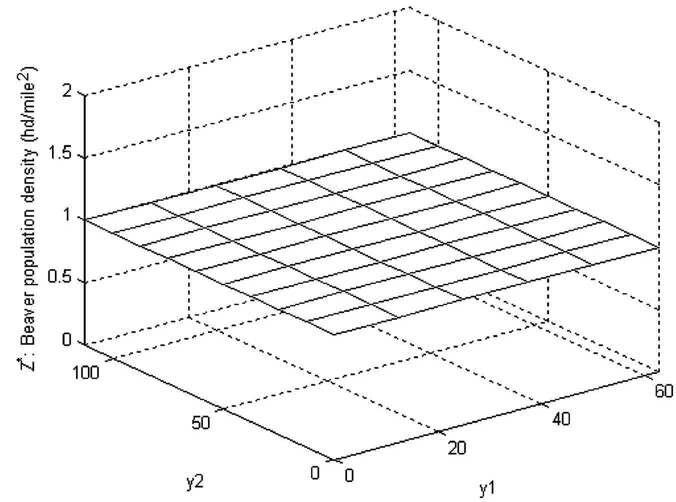


Fig. 2. Required steady-state population density distribution.

4) If proper convergence is achieved, stop and revert to step 1 with S_{i+1}. If not, go to step 1 and retrain the networks with a new S_i.

In order to minimize the chance of getting trapped in a local minimum, a *batch training* is used wherein the network is trained for all of the elements of S_i together.

3) Convergence Condition: At each c_i it should be assured that the proper convergence in training is reached. For this purpose, generate a set S_i^c of profiles in exactly the same manner used to generate S_i.

- Fix a tolerance value tol (we have fixed $tol = 0.05$).
- Using the state profiles from S_i^c, generate the target outputs as described in Section IV-B. Let the outputs be $\lambda_1^{t_i}, \ldots, \lambda_{\tilde{N}}^{t_i}$.
- Generate the actual output from the networks, by using the state profiles from S_k^c. Say the values of the outputs are $\lambda_1^{a_i}, \ldots, \lambda_{\tilde{N}}^{a_i}$.
- Check whether simultaneously $\|\lambda_j^{t_i} - \lambda_j^{a_i}\|_2 / \|\lambda_j^{t_i}\|_2 < tol, \forall\ j = 1, 2, \ldots, \tilde{N}$. If yes, then the networks have converged.

Note that after successful training of the networks offline, the control can be implemented online, since it can then be computed by simply using the networks.

4) Neural Network Structure and Initialization: Since $\tilde{N} = 4$ for the beaver problem, four critic networks are used with X_k as inputs and a component of the vector λ_{k+1} as the output. For the wildlife management problem, the network architecture is $\pi_{4,6,6,1}$, where $\pi_{4,6,6,1}$ means that there are four neurons in the input layer, six neurons in the first hidden layer, six neurons in the second hidden layer, and one neuron in the output layer. For activation functions, a *tangent sigmoid* function for the input and hidden layers and a *linear* function for the output layer are used. Simulation results indicate the network choices were adequate.

Note that the reduced-order dynamics (12)–(13) is in the control-affine SDC form and the associated cost function (14) is also quadratic. This facilitates the use of the SDRE technique to this reduced-order problem as well [9], [10], [21]. Hence, the randomly chosen weights of the networks were first trained with respect to this solution (we call this process "pretraining"),

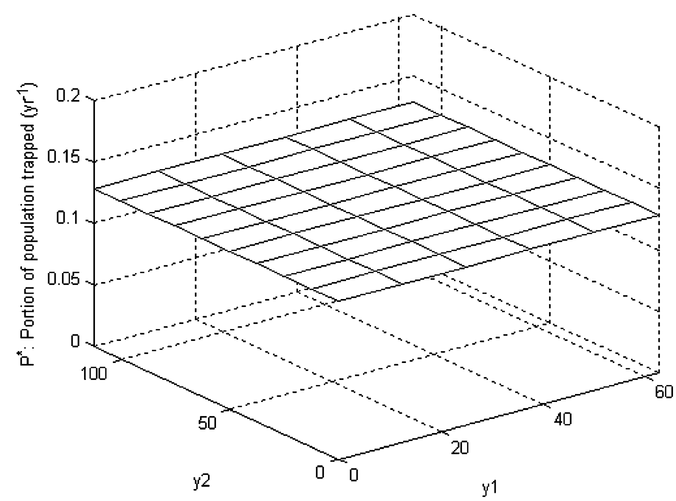


Fig. 3. Required steady-state trapping rate for population density distribution in Fig. 2.

before starting the SNAC training outlined in Section IV-B. Further details of this process are omitted for brevity.

V. Numerical Results

A. Selection of Numerical Values

The values of parameters used in the numerical experiments in this study are the same as used in [5] and [17] wherever relevant. A spatial domain having $L_1 = 62.75$ mi and $L_2 = 112.95$ mi was selected and the grid parameters are $\Delta y_1 = \Delta y_2 = 12.55$ mi. The time step $\Delta t = (7/365)$ yr (one week) was chosen, which means that the control solution (rate of beavers to be harvested) is updated every week. For the cost function weights $q = r = 1$ was chosen.

B. Analysis of Results

The main goal for this problem was to drive the actual beaver population density distribution Z to a required beaver population density distribution Z^*. According to the formulation in this

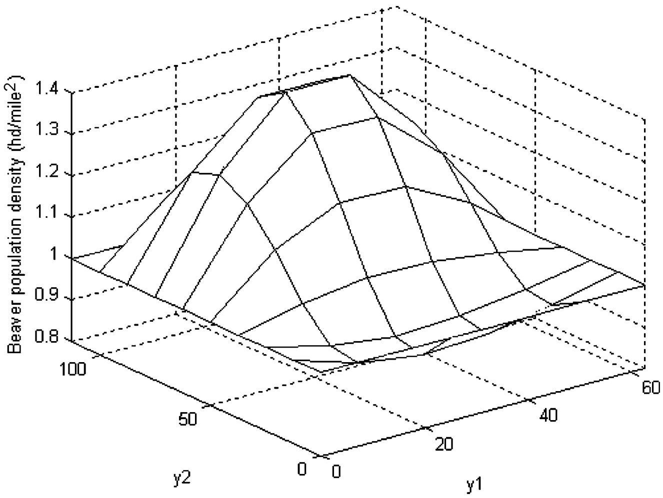


Fig. 4. Initial condition for the population density distribution (state).

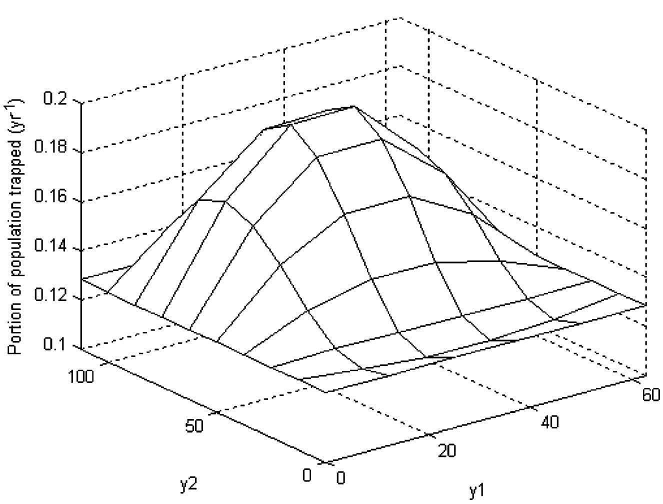


Fig. 5. Trapping rate (control) distribution for the initial condition as in Fig. 4.

study, the distribution of the control variable P, which is the portion of Z to be trapped per year (see Section II), should approach a steady-state distribution for any initial condition in the domain of interest for which the networks have been trained. This has been verified from a large number of simulation studies. However, since it is impossible to all the simulation results (due to space constraints), the results for one representative case is presented in Figs. 2–17. For the rest of the discussion in this section, we call P as the "trapping rate" for convenience.

Figs. 2–3 depict the steady-state beaver population density distribution and the associated steady-state trapping rate, respectively. In other words, starting from any initial condition that has been accounted for training the networks, the population density distribution and the trapping rate distribution should converge to these distributions with time.

The randomly chosen initial condition for the population density distribution (at time $t = 0$) for this simulation and the computed trapping rate distribution are as shown in Figs. 4 and 5, respectively.

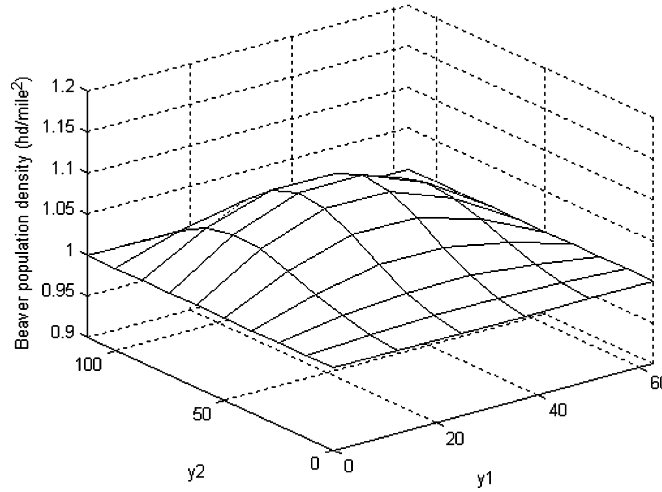


Fig. 6. Population density distribution at $t = 6$ mo.

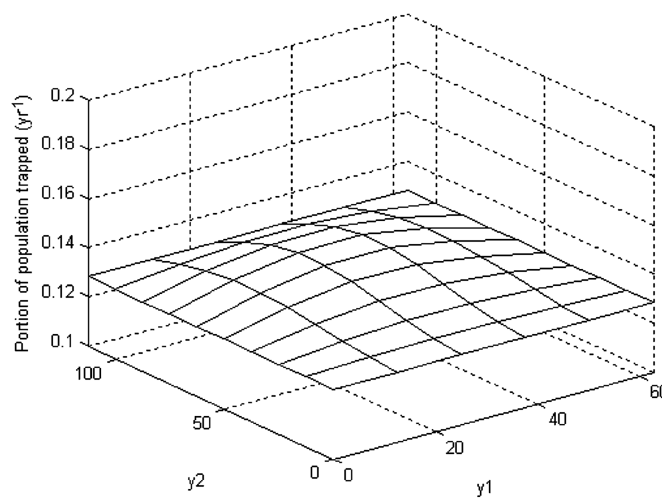


Fig. 7. Trapping rate distribution at $t = 6$ mo.

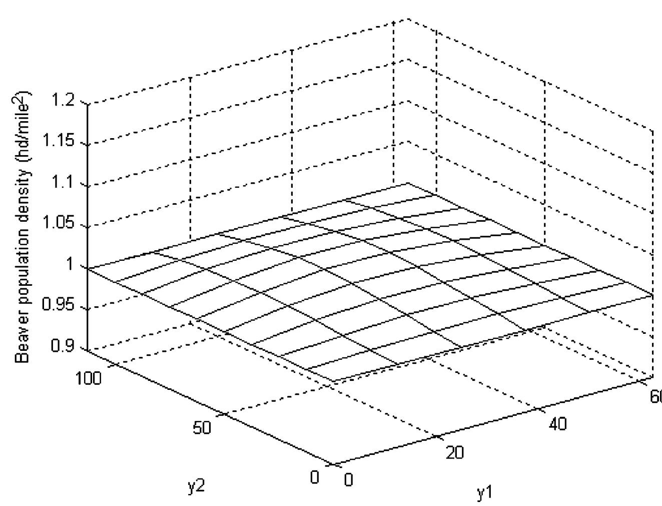


Fig. 8. Population density distribution at $t = 1$ yr.

Since it is impossible to show a 3-D surface plot as it evolves with time, distributions for population density (state) and trapping rate (control) are presented at different time instants. At

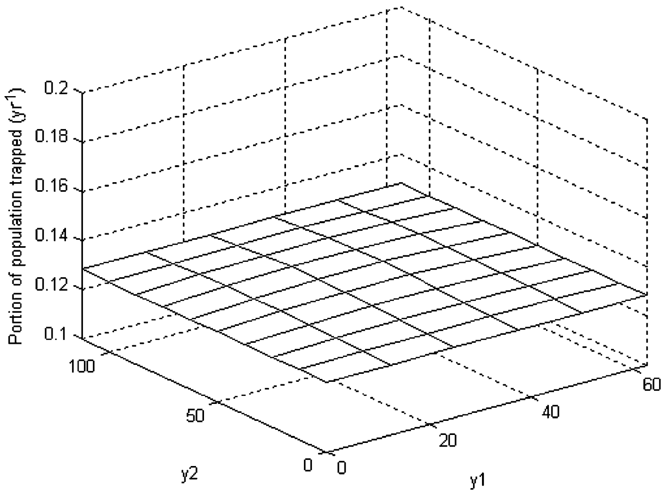


Fig. 9. Trapping rate distribution at $t = 1$ yr.

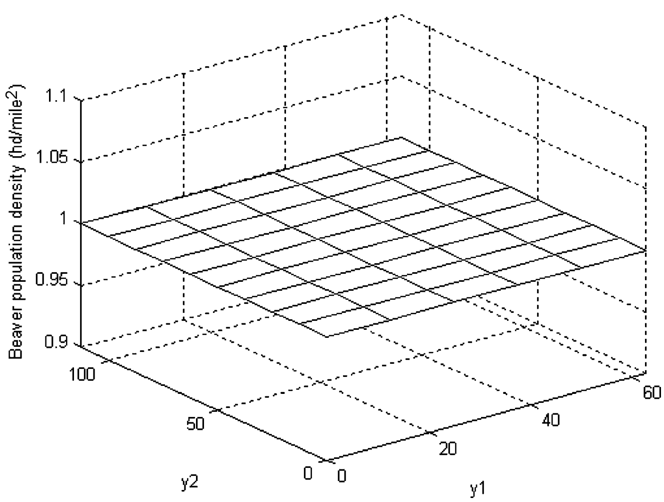


Fig. 10. Population density distribution at $t = 5$ yr.

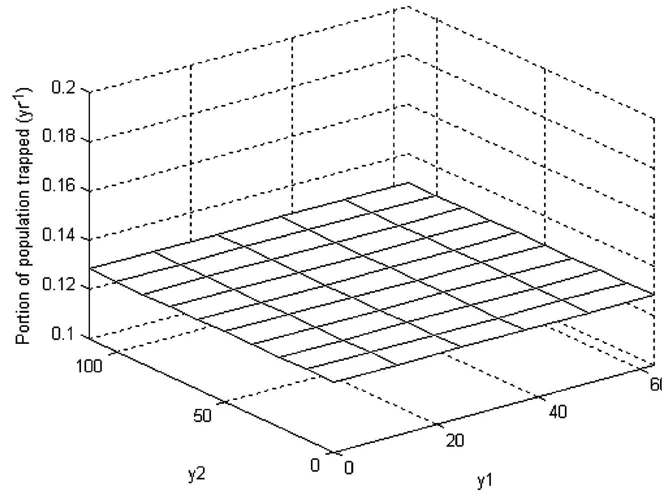


Fig. 11. Trapping rate distribution at $t = 5$ yr.

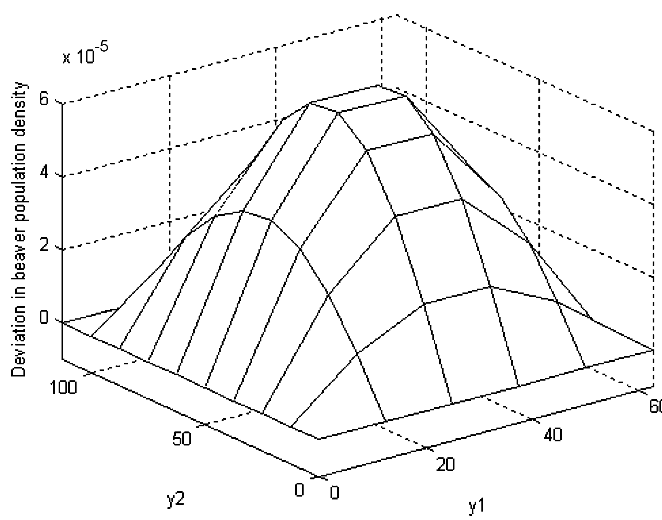


Fig. 12. Error between actual population density distribution and desired population density distribution at $t = 5$ yr.

$t = 6$ mo, the state and control are as in Figs. 6 and 7, respectively. At $t = 1$ yr, the state and control are as in Figs. 8 and 9, respectively. From these figures, it is clear that both the state and control are developing toward their respective desired steady-state profiles. In other words, the beaver population density distribution, as well as the trapping rate distribution, develops toward their respective desired steady-state distributions, respectively.

Note that the problem in this study has been formulated in an infinite time framework and the simulation can essentially be continued for as long as one wishes. Simulations in this paper were carried out for $t = 5$ yr. It was observed in the simulations that the steady-state distributions for beaver population density and their trapping rate were reached in approximately two years. The population density distribution and trapping rate distribution at $t = 5$ yr are shown in Figs. 10 and 11, respectively.

It is evident that the Figs. 10 and 11 are quite close to the respective desired steady-state distributions (compared with Figs. 2 and 3, respectively). However, to show even better accuracy that can be reached, the error between the actual population density distribution and its desired steady-state population density distribution, as well as the actual trapping rate distribution and its desired steady-state trapping rate distribution at $t = 5$ yr were plotted. These are shown in Figs. 12 and 13, respectively.

It is quite clear from Figs. 12 and 13 that the order of magnitude of the error of population density (state) is five-orders of magnitude smaller than its desired steady-state distribution. Similarly, the order of magnitude of the error in the trapping rate (control) is three-orders of magnitude smaller than the desired steady-state trapping rate distribution. These trends show that the controller performance is excellent in driving the state to the desired target.

To further illustrate the way the state and control develop toward their steady state, the time histories of the lumped parameter states and controls for the state and control deviations (see Section III-B) are shown in Figs. 14 and 15, respectively. From these figures, it is clear that the state and control converge to

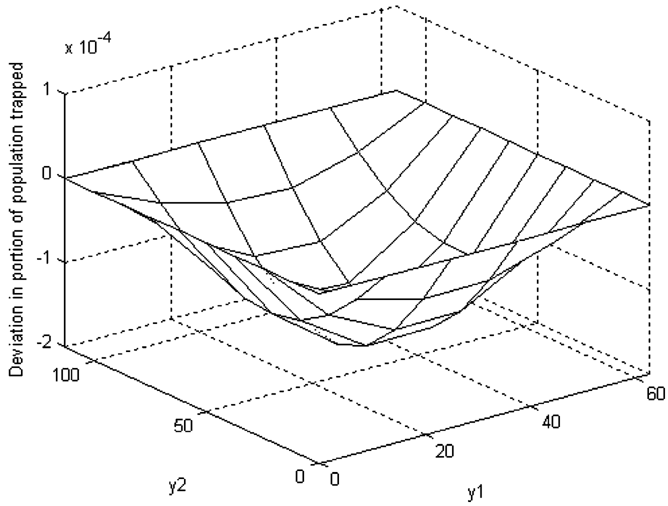


Fig. 13. Error between actual trapping rate distribution and desired trapping rate distribution at $t = 5$ yr.

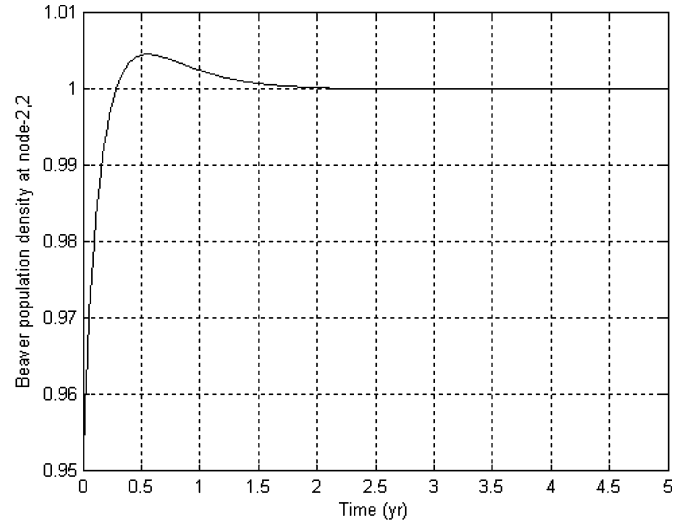


Fig. 16. Population density evolution at node-2,2.

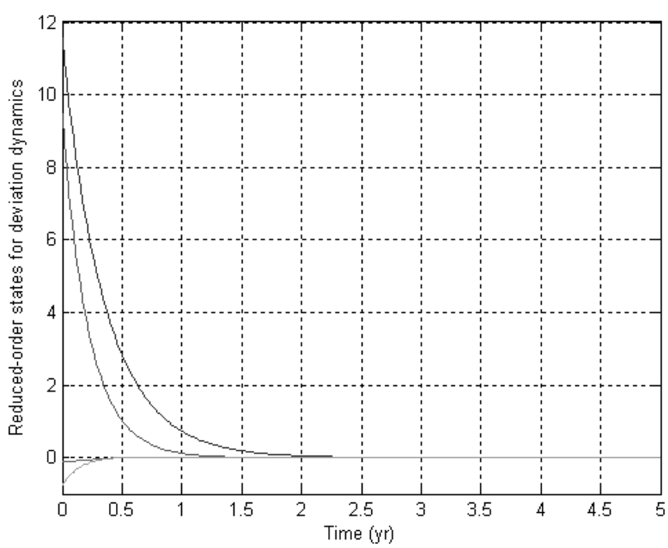


Fig. 14. Lumped parameter state histories for the state deviation.

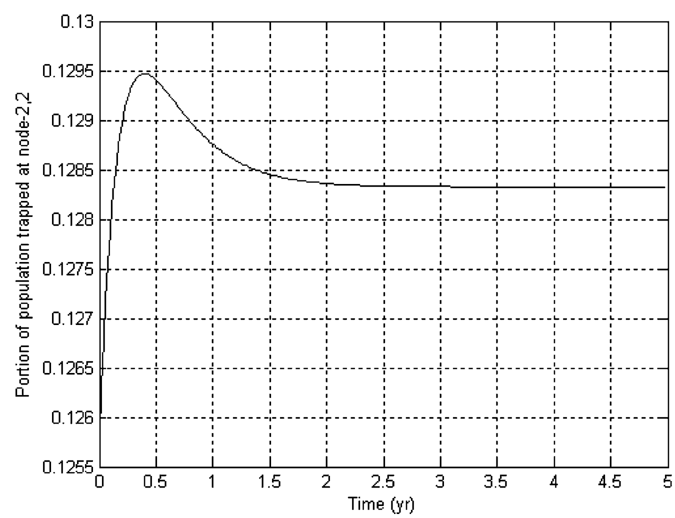


Fig. 17. Trapping rate evolution at node-2,2.

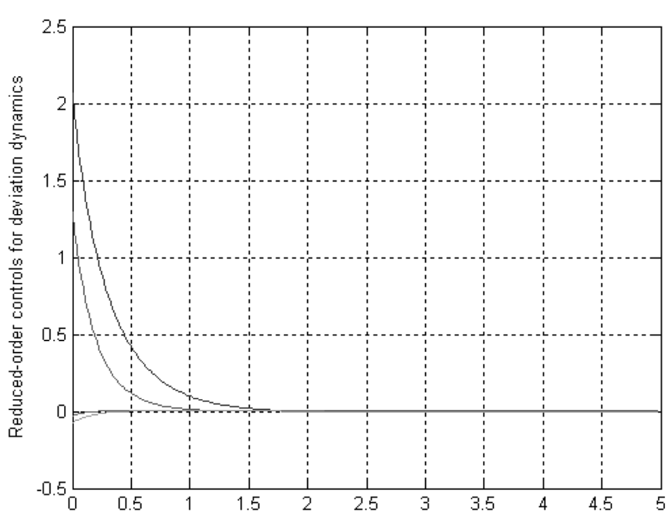


Fig. 15. Lumped parameter control histories for the control deviation.

their respective targets in about two years and stay there afterwards. Such phenomena were observed in all the simulations.

Next, to see the time history at a specific location in the domain clearly, we arbitrarily chose node-22 and plotted the beaver population density and trapping rate at that location in Figs. 16 and 17, respectively. From these plots, it is obvious how the state and control evolve toward their respective steady-state values. The over-shooting is minimal, indicating that the choice of the cost function weight parameters $q = r = 1$ was appropriate. A similar trend was observed at all node points. Note that since large oscillations of both the population density, as well as the trapping rate are not observed, the management policy proposed here is more likely to be accepted by the farmers.

Note that the SDRE technique, which was used both to generate the snap-shot solutions, as well as for "pretraining" the networks, leads to a locally stabilizing suboptimal controller [9], [10], [21]. Hence, the pretrained networks also lead to a suboptimal controller. To demonstrate how the SNAC technique

TABLE II
COST COMPARISON BETWEEN TRAINED AND PRETRAINED NETWORKS

Case	$J_{trained}$	$J_{pre-trained}$
1	6.4883	10.4644
2	13.4060	23.4481
3	21.7651	41.2040
4	31.1715	61.7291
5	42.8081	88.2159

leads to an optimal controller, we have carried out a cost comparison study. For this, we took some random initial condition and simulated the evolution for $t_f = 5$ yr (by that time the steady-state conditions have already been achieved) by both using the pretrained networks (by SDRE technique), as well by the fully-trained networks (by the SNAC technique). Some representative results are as in Table II.

From the numbers in Table II, it is quite obvious that in all cases the pretrained networks lead to cost function values that are approximately double the values of the cost function, if the SNAC-trained networks are used. Hence, we conclude that the pretrained networks are very much "suboptimal," which are driven more toward their optimal values by the SNAC technique.

VI. CONCLUSION

An optimal control theory-based solution has been proposed as a harvesting strategy of nuisance creating beavers. A distributed parameter model for beaver population dynamics has been presented and a proper orthogonal decomposition-based basis function design followed by their use in a Galerkin projection has been incorporated in the process as a model reduction technique. Using the reduced model in a single network adaptive critic scheme, the optimal control is made available in a feedback form.

The optimal harvesting technique presented for managing the beaver population in a desired territory leads to a desired distribution of the animal density in the long run. It does not lead to the complete elimination of the species. Besides, the control computed can be used online by a nonexpert in control theory since it essentially involves only using the neural networks (which are trained offline). Because of these reasons, the proposed technique can be used as a great tool by a wildlife manager.

ACKNOWLEDGMENT

The authors would like to thank Dr. M. G. Bhat, Associate Professor, Department of Environmental Studies, Florida International University, Miami, for their helpful and productive e-mail communications. The authors would also like to thank the anonymous reviewers for their constructive criticisms and useful suggestions, which led to a substantial improvement of this paper.

REFERENCES

[1] S. N. Balakrishnan and V. Biega, "Adaptive-critic based neural networks for aircraft optimal control," *J. Guid. Control Dyn.*, vol. 19, no. 4, pp. 893–898, Jul./Aug. 1996.

[2] B. Bameih, "The structure of optimal controllers of spatially-invariant distributed parameter systems," in *Proc. Conf. Dec. Contr.*, 1997, pp. 1056–1061.

[3] H. T. Banks, R. C. H. Rosario, and R. C. Smith, "Reduced-order model feedback control design: Numerical implementation in a thin shell model," *IEEE Trans. Autom. Contr.*, vol. 45, no. 7, pp. 1312–1324, Jul. 2000.

[4] A. R. Barron, "Universal approximation bounds for superposition of a sigmoidal function," *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 930–945, May 1993.

[5] M. G. Bhat, "Controlling wildlife damage by diffusing beaver population: a bioeconomic application of the distributed parameter control model," Ph.D. dissertation, Dept. Agricultural Economics Rural Soc., Univ. Tennessee, Knoxville, TN, 1991.

[6] A. E. Bryson and Y. C. Ho, *Applied Optimal Control*. London, U. K.: Taylor & Francis, 1975.

[7] J. Burns and B. B. King, "A reduced basis approach to the design of low-order feedback controllers for nonlinear continuous systems," *J. Vibration Contr.*, vol. 4, pp. 297–323, 1998.

[8] C. W. Clark, *Mathematical Bioeconomics: The Optimal Management of Renewable Resources*, 2nd ed. New York: Wiley, 1990.

[9] J. R. Cloutier, "State-dependent Riccati equation techniques: An overview," in *Proc. Amer. Contr. Conf.*, 1997, pp. 932–936.

[10] J. R. Cloutier and D. T. Stansbery, "The capabilities and art of state-dependent Riccati equation-based design," in *Proc. Amer. Contr. Conf.*, 2002, pp. 86–91.

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

[12] S. K. Gupta, *Numerical Methods for Engineers*. New York: Wiley, 1995.

[13] S. Harkonen, "Forest damage caused by the Canadian beaver (castor canadensis) in South Savo, Finland," *Silva Fennica*, vol. 33, no. 4, pp. 247–259, 1999.

[14] P. Holmes, J. L. Lumley, and G. Berkooz, *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge, U.K.: Cambridge Univ. Press, 1996, pp. 87–154.

[15] K. J. Hunt, "Neural networks for control systems—A survey," *Automatica*, vol. 28, no. 6, pp. 1083–1112, Nov. 1992.

[16] I. Lasiecka, "Control of systems governed by partial differential equations: A historical perspective," in *Proc. 34th Conf. Dec. Contr.*, 1995, pp. 2792–2796.

[17] S. M. Lenhart and M. G. Bhat, "Application of distributed parameter control model in wildlife damage control management," *Math. Models Methods Appl. Sci.*, vol. 2, no. 4, pp. 423–439, 1992.

[18] F. Lewis and V. L. Syrmos, *Optimal Control*, 2nd ed. New York: Wiley, 1995.

[19] M. C. McKinstry and S. H. Anderson, "Attitude of private and public managers in Wyoming, USA, toward beaver," *Environ. Manage.*, vol. 23, no. 1, pp. 95–101, 1999.

[20] S. T. McTaggart and T. A. Nelson, "Composition and demographics of beaver (castor canadensis) colonies in central Illinois," *Amer. Midland Naturalist*, vol. 150, no. 1, pp. 139–150, 2003.

[21] C. P. Mracek and J. R. Cloutier, "Control designs for the nonlinear benchmark problem via the state-dependent Riccati equation method," *Int. J. Robust Nonlinear Contr.*, vol. 8, no. 4-5, pp. 401–433, 1998.

[22] S. S. Ravindran, "A reduced-order approach for optimal control of fluids using proper orthogonal decomposition," *Int. J. Num. Methods Fluids*, vol. 34, no. 5, pp. 425–448, 2000.

[23] S. Sadek and M. A. Bokhari, "Optimal control of a parabolic distributed parameter system via orthogonal polynomials," *Opt. Contr. Appl. Methods*, vol. 19, no. 3, pp. 205–213, 1998.

[24] S. N. Singh, J. H. Myatt, and G. A. Addington, "Adaptive feedback linearizing control of proper orthogonal decomposition nonlinear flow models," in *Proc. Amer. Contr. Conf.*, 2001, pp. 1533–1538.

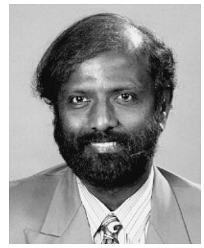
[25] J. P. Wright, C. G. Jones, and A. S. Flecker, "An ecosystem engineer, the beaver, increases species richness at the landscape scale," *Oecologia*, vol. 132, no. 1, pp. 96–101, Jun. 2002.

[26] P. J. Werbos, "Approximate dynamic programming for real-time control and neural modeling," in *Handbook of Intelligent Control*, D. A. White and D. A. Sofge, Eds: Multiscience Press, 1992.

Radhakant Padhi (M'05) received the B.S.Eng. degree in mechanical engineering from the University College of Engineering-Burla, Orissa, India, in 1994, the M.S.Eng. degree in aerospace engineering from the Indian Institute of Science, Bangalore, India, in 1996, and the Ph.D. degree in aerospace engineering from the University of Missouri-Rolla, Rolla, MO, in 2001.

From April 1996 to July 1997, he worked as a Scientist in the Defense Research and Development Organization, Hyderabad, India. In 2001, he worked as a Postdoctoral Fellow at the University of Missouri-Rolla for two years. In January 2004, he began working as an Assistant Professor in the Department of Aerospace Engineering, Indian Institute of Science. His research interests include the broad area of systems theory and applications. Specifically, he is interested in nonlinear control design and analysis, optimization and optimal control, distributed parameter systems, neural networks, control and guidance of aerospace vehicles, and application of control theory to biomedical systems.

Dr. Padhi was a recipient of the Best Student Paper Award (third place) in the 2001 IEEE Joint Conference on Control Applications and International Symposium on Intelligent Control, Mexico City. He also received the Best Application Paper Award in XXIX National Systems Conference, in 2005, IIT-Bombay, India. His biography is listed in *Who's Who in America,* 2004.



S. N. Balakrishnan received the Ph.D. degree in aerospace engineering from the University of Texas, Austin, in 1983.

In 1985, he joined the University of Missouri-Rolla, Rolla, MO. He is currently a Professor in the Department of Mechanical and Aerospace Engineering. His nonteaching experience includes work as a Lead Engineer in the Space Shuttle program, Fellow, Center for Space Research at the University of Texas at Austin, and Summer faculty Fellow at the Air Force Research Laboratory, Eglin, FL. His research interests include the areas of System Theory and Applications. His current research uses neural networks and classical methods in the identification and robust control of missiles, airplanes, rockets, and other "interesting" systems. He has developed a suboptimal closed form nonlinear control method called theta-D and used it to solve engineering problems in many disciplines that includes spacecraft formation control. His work has been sponsored by the National Science Foundation, the Air Force, the Naval Surface Warfare Center, the Army Space and Missile Defense Command, and NASA.

Dr. Balakrishnan is a member of Sigma Gamma Tau and an Associate Fellow of the American Institute of Aeronautics and Astronautics.