JOURNAL OF PROPULSION AND POWER
Vol. 21, No. 5, September–October 2005

Performance Optimization of a Magnetohydrodynamic Generator at the Scramjet Inlet

Nilesh V. Kulkarni*
QSS Group, Inc., Moffett Field, California 94035
and
Minh Q. Phan†
Dartmouth College, Hanover, New Hampshire 03755

The performance optimization problem of a magnetohydrodynamic (MHD) generator at the inlet of a scramjet engine is studied. The generator performance is characterized in terms of both the net energy extracted from the flow and the flow characteristics through the channel before it enters the combustion chamber of the scramjet engine. The analysis assumes a steady-state one-dimensional flow in which the position along the channel is treated as an independent coordinate. The performance optimization problem can then be handled by optimal control theory, in which position now plays the role of time as in a standard control problem. In this work, an optimal neural-networks-based controller design technique developed in our previous work is used. The technique is well suited for this application, because the MHD system is highly nonlinear. Furthermore, the technique is data-based, in that experimental data obtained from the system can be used to design the optimal controller without requiring an explicit model of the system.

Nomenclature

A_f	=	cross-sectional area of the channel, m^2
B_f	=	externally applied magnetic field, T
$\boldsymbol{b}_{\mathrm{NN1}}, \boldsymbol{b}_{\mathrm{NN2}}$	=	neural network biases for the first and second layers, respectively
C_v	=	specific heat of the fluid at constant volume, J/kg · K
e	=	electron charge, C
$\boldsymbol{f}_{\mathrm{NN1}}, \boldsymbol{f}_{\mathrm{NN2}}$	=	neuron activation function for the first and second layers, respectively
$\boldsymbol{g}$	=	feedback function that computes the control given the state of the system (controller)
J	=	cost function
j_{bf}	=	electron beam current, A
k	=	discretized index of position
k_f	=	load factor
L	=	Langragian, current time cost function
L_{bf}	=	effective length of ionization, m
M_{fe}	=	prescribed channel exit Mach number
N_f	=	number density of the fluid particles, m^{-3}
n_{ef}	=	number density of electrons in the fluid, m^{-3}
P_f	=	static pressure of the fluid, N/m^2
p_1, p_2	=	end position cost weighting terms
$\boldsymbol{Q}, \boldsymbol{R}$	=	state and control weighting matrices in the cost function
$Q_{\beta f}$	=	energy deposited by the electron beams, J/m^3
q_1, q_2, q_3	=	integral cost weighting terms
R_m	=	Rydberg molecular constant, J/kg · K
r_1	=	control cost weighting term
S_f	=	entropy of the fluid, J/kg · K
T_f	=	temperature of the fluid, K
T_{fe}	=	prescribed channel exit temperature, K
t	=	time, s
t_f	=	final time, s
$\boldsymbol{u}$	=	control input to the system
$\boldsymbol{u}_{\mathrm{NN}}$	=	inputs to the neural network
v_f	=	velocity of the fluid, m/s
$\boldsymbol{W}_{1,\mathrm{inp}}$	=	neural network weights for the layer connecting the inputs to the first (hidden) layer
$\boldsymbol{W}_{2,1}$	=	neural network weights connecting the first layer to the second (output) layer
$\boldsymbol{w}$	=	fluid state vector
x	=	position along the channel, m
$\boldsymbol{x}$	=	state of the system
x_f	=	position coordinate for the channel exit, m
$\boldsymbol{x}_{\mathrm{NN1}}, \boldsymbol{x}_{\mathrm{NN2}}$	=	inputs to the first and second layers of the neural network, respectively
Y_i	=	energy cost of ionization per fluid molecule, eV
$\boldsymbol{y}_{\mathrm{NN1}}, \boldsymbol{y}_{\mathrm{NN2}}$	=	neural network outputs from the first and second layers, respectively
Z_f	=	width of the channel, m
β_f	=	electron–ion combination rate, m^3/s
γ_m	=	ratio of the specific heats of the fluid
$\varepsilon_{\mathrm{bf}}$	=	electron beam energy, eV
ε_f	=	internal energy of the fluid, J/kg
μ_f	=	mobility of the electrons in the fluid, m^2/V · s
ρ_f	=	density of the fluid, kg/m^3
σ_f	=	conductivity of the fluid, $(\Omega \cdot \mathrm{m})^{-1}$
ϕ	=	end time cost function

Presented as Paper 2002-5121 at the AIAA/AAAF 11th International Spaceplanes and Hypersonic Technologies Conference, Orleans, France, 29 September–4 October 2002; received 8 May 2003; revision received 18 January 2005; accepted for publication 6 February 2005.

*Research Engineer, QSS Group, Inc.; Ph.D. Candidate, Department of Mechanical and Aerospace Engineering, Princeton University; nilesh@email.arc.nasa.gov. Member AIAA.

†Associate Professor, Thayer School of Engineering; Minh.Q.Phan@dartmouth.edu. Senior Member AIAA.

I. Introduction

POSSIBLE application of magnetohydrodynamics (MHD) in hypersonic systems has generated a lot of interest in recent years. Several applications[1–8] have been suggested that use MHD as an integral part of flight systems. These range from using MHD for the modification of the external flowfield ahead of the flight vehicle to MHD power generation and the MHD energy bypass engine (AJAX) concept. MHD concepts have been used before in different engineering applications. They have been used for ground-based electrical power generation since the 1950s.[9] In the case of the MHD energy bypass engine, the application relies on an external nonequilibrium source of ionization due to the low static

temperature of these hypersonic flows. Several nonequilibrium ionization methods have been analyzed, and it has been shown that high-energy electron beams present themselves as the most efficient ionizers.[10–12] The electron-beam current profile can be varied along the hypersonic channel to optimize the performance of the MHD device. The electron-beam current can therefore be treated as a control input variable for the MHD systems, and the resultant optimization problem can be handled by optimal control theory.

In this work we present the control implementation for the hypersonic channel that acts as an MHD generator at the inlet of the energy bypass engine. The performance of the MHD channel is optimized using the neural-networks-based control algorithm developed in previous work.[13,14] The application of this approach is appropriate because the MHD system is nonlinear, as will be described in the next section. At the same time, because the approach is data-based, experimentally obtained data from the MHD channel can be directly used for designing an optimal controller without the availability of an explicit analytical model in standard form. A detailed discussion on the field of optimal control can be found in the literature.[15]

The detailed analysis of the MHD system would typically consist of solving three-dimensional, time-dependent MHD equations with the electron beam current as an input to this system. For the present work, we focus on steady-state behavior, with dependence on the x coordinate alone, along the length of the channel. The system of partial differential equations reduces to ordinary differential equations with the position along the length of the channel as an independent coordinate. The optimal control problem then corresponds to finding the electron-beam current settings along the channel that optimize a prescribed performance measure for any given set of inlet conditions.

Section II gives the details of the inlet channel and the MHD flow equations. Section III discusses how optimal control theory can be applied to the present problem. Section IV summarizes the data-based optimal controller design technique developed in previous experiments.[13,14] Finally, Section V presents the results of the optimization for different performance measures.

II. System Description

The system under consideration is a hypersonic channel at the inlet of the MHD energy bypass engine. The freestream flow is assumed to be deflected by the nose of the vehicle before entering the inlet by an angle of 10 deg. The flow (in the x direction) enters the channel, where it is subjected to the electron beams that ionize the flow. As shown in Fig. 1, the electron beams are introduced in the same direction (z direction) as the applied magnetic field. As the ionized flow passes across the magnetic field lines, an emf is generated (in the y direction) perpendicular to both the flow direction and the applied magnetic field direction. This induced emf gives rise to an induced current, as depicted in Fig. 1.

The induced current can drive an external load or can be used to accelerate the flow in the nozzle in a reverse mechanism as outlined for the energy bypass engine concept.[2,3] In this process, the flow experiences the Faraday force that decelerates the flow, thereby converting the kinetic energy of the flow to electrical energy. Thus, the channel is assumed to be an ideal Faraday generator.[9] The geometry of the hypersonic inlet channel in this study is chosen to be consistent with the one used in Refs. 10–12.

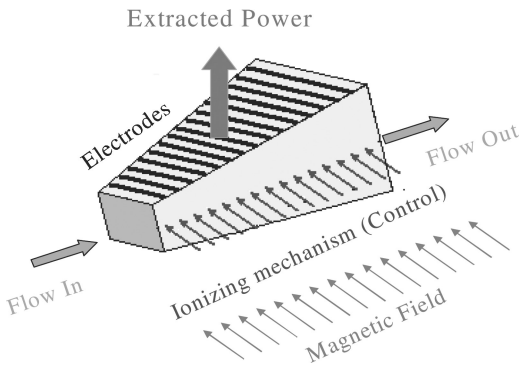

Fig. 1 Schematic of the MHD power generator (adapted from Ref. 10).

A. Specifications for the MHD Channel

The inlet area is 0.0225 m^2 (15×15 cm) and the exit area is 0.1225 m^2 (35×35 cm). The width of the channel is assumed to vary linearly along the length of the channel. The applied external magnetic field is 5 T and the length of the channel is 1 m. The freestream conditions are as follows: altitude is 30 km ($\pm 10\%$) and mach number is 8 ($\pm 10\%$).

A 10% window is provided for the freestream conditions so that the controller can be designed for a range of freestream conditions in this window. It is important to point out that after design the same controller will give the electron-beam current control profile for any Mach number and altitude in this range.

B. Governing Equations

Neglecting detailed chemical kinetics and assuming steady-state conditions, the governing equations for the one-dimensional flow are given as follows. These correspond to the Euler equations with the added MHD terms.

Continuity:

$$\frac{\mathrm{d}(\rho_f v_f A_f)}{\mathrm{d}x} = 0 \tag{1}$$

which states that the mass flow through the channel remains constant.

Force:

$$\rho_f v_f \frac{\mathrm{d}v_f}{\mathrm{d}x} + \frac{\mathrm{d}P_f}{\mathrm{d}x} = -(1 - k_f)\sigma_f v_f B_f^2 \tag{2}$$

where the term on the right-hand side corresponds to the Faraday force. The load factor k_f is a measure of the power extracted by the external load. For the present work, k_f is chosen to be 0.5. The conductivity of the fluid is given by

$$\sigma_f = e\mu_f n_{\mathrm{ef}} \tag{3}$$

The mobility of the electrons, μ_f, is given by

$$\mu_f = (44/P_f)(T_f/300) \tag{4}$$

where the pressure P_f is given in torr.

Energy:

$$\rho_f v_f \frac{\mathrm{d}\left[\gamma_m \varepsilon_f + \left(v_f^2/2\right)\right]}{\mathrm{d}x} = -k_f(1 - k_f)\sigma_f v_f^2 B_f^2 + Q_{\beta\mathrm{f}} \tag{5}$$

which states that the energy change in the system is due to the energy extraction from the flow given by the first term on the right-hand side, and the energy addition is due to the applied electron-beam current given by the second term on the right-hand side. The ratio of the specific heats of the fluid, γ_m, is chosen to be 1.4, and the internal energy of the fluid, ε_f, is given by

$$\varepsilon_f = R_m T_f/(\gamma_m - 1) \tag{6}$$

where the value of the molecular constant, R_m, is 287.12 J/kg $\cdot$ K. The energy deposited by the electron beam, $Q_{\beta\mathrm{f}}$, is given by

$$Q_{\beta\mathrm{f}} = 2 j_{\mathrm{bf}} \varepsilon_{\mathrm{bf}}/Z_f \tag{7}$$

The electron beam is assumed to be impinging from both sides ofthe channel. Hence, we observe the factor of 2 in Eq. (7).

The effective length of ionization, L_{bf}, due to this electron beam is given by[16]

$$L_{\mathrm{bf}}(\varepsilon_{\mathrm{bf}}, N_f) = 1.1 \times 10^{21}\left(\varepsilon_{\mathrm{bf}}^{1.7}/N_f\right) \tag{8}$$

where the electron-beam energy $\varepsilon_{\mathrm{bf}}$ is measured in kiloelectron volts.

For the channel, the effective length of ionization for obtaining almost uniform ionization is assumed to be

$$L_{\mathrm{bf}} = \tfrac{2}{3} Z_f \tag{9}$$

Using Eqs. (8) and (9), the electron-beam energy $\varepsilon_{\mathrm{bf}}$ at a particular location x can be calculated for uniform ionization. This value is used in Eq. (7).

Electron continuity:

The equation for the number density of the electrons that corresponds to its continuity equation is given by

$$\frac{\mathrm{d}(n_{\mathrm{ef}} v_f)}{\mathrm{d}x} = \frac{2 j_{\mathrm{bf}} \varepsilon_{\mathrm{bf}}}{e Y_i Z_f} - \beta_f n_{\mathrm{ef}}^2 \tag{10}$$

Equation (10) states that the electron flux changes with the addition of electrons due to the ionization by the applied electron beam and their depletion due to ion–electron recombination; $\varepsilon_{\mathrm{bf}}$ is measured in electron volts in this equation. Here it is implicitly assumed that the electron and ion number densities are equal under the assumption of plasma quasineutrality. The energy cost of ionization per fluid molecule, Y_i, is 34 eV. The rate constant β_f is given by

$$\beta_f = 2.1 \times 10^{-13}\ \mathrm{m^{-6}s^{-1}} \tag{11}$$

Finally, we have the equation of state:

$$P_f = \rho_f R_m T_f \tag{12}$$

which can be used to substitute temperature in terms of pressure and density in the preceding equations. Using this substitution, Eqs. (1), (2), (5), and (10) can be written together as

$$\begin{bmatrix} v_f A_f & \rho_f A_f & 0 & 0 \\ 0 & \rho_f v_f & 1 & 0 \\ \dfrac{-\gamma_m v_f P_f}{(\gamma_m - 1)} & \dfrac{1}{2}\rho_f v_f^2 & \dfrac{\gamma_m v_f}{(\gamma_m - 1)} & 0 \\ 0 & n_{\mathrm{ef}} & 0 & v_f \end{bmatrix} \begin{bmatrix} \dfrac{\mathrm{d}\rho_f}{\mathrm{d}x} \\ \dfrac{\mathrm{d}v_f}{\mathrm{d}x} \\ \dfrac{\mathrm{d}P_f}{\mathrm{d}x} \\ \dfrac{\mathrm{d}n_{\mathrm{ef}}}{\mathrm{d}x} \end{bmatrix} = \begin{bmatrix} -\rho_f v_f \dfrac{\mathrm{d}A_f}{\mathrm{d}x} \\ -(1 - k_f)\sigma_f v_f B_f^2 \\ -k_f(1 - k_f)\sigma_f v_f^2 B_f^2 + Q_{\beta f} \\ \dfrac{2 j_{\mathrm{bf}} \varepsilon_{\mathrm{bf}}}{e Y_i Z_f} - \beta_f n_{\mathrm{ef}}^2 \end{bmatrix} \tag{13}$$

Because the first matrix on the left-hand side is invertible for nonzero fluid velocity, the preceding equation can be written in a more concise form:

$$\frac{\mathrm{d}\boldsymbol{w}}{\mathrm{d}x} = f[\boldsymbol{w}(x), \boldsymbol{u}(x), x] \tag{14}$$

where $\boldsymbol{w}(x)$ corresponds to the state vector given by

$$\boldsymbol{w} = \begin{bmatrix} \rho_f \\ v_f \\ P_f \\ n_{\mathrm{ef}} \end{bmatrix} \tag{15}$$

and where $\boldsymbol{u}(x)$ corresponds to the control, which in this case is the electron-beam current j_{bf}, and x corresponds to the position variable that acts as the independent variable for this system. Thus, Eq. (14) governs the behavior of this MHD system.

III. Optimal Control Formulation

As outlined in the Introduction, the optimization of the MHD channel performance can be treated as an optimal control problem. Therefore, before the control implementation is described it is important to draw parallels between the standard time-based optimal control problem and the present problem.

For a time-dependent optimal control problem, we are given a system with dynamics

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = f[\boldsymbol{x}(t), \boldsymbol{u}(t), t] \tag{16}$$

where $\boldsymbol{x}$ refers to the state of the system [equivalent to $\boldsymbol{w}$ in Eq. (14)], which describes all the variables of the system at a given time. Note that $\boldsymbol{x}$ denotes the state variable, compared to the variable x, which denotes the position variable along the length of the channel. The problem of optimal control then corresponds to finding a control law

$$\boldsymbol{u}(t) = \boldsymbol{g}[\boldsymbol{x}(t), t] \tag{17}$$

that minimizes a prescribed cost function:

$$J = \phi[\boldsymbol{x}(t_f), t_f] + \int_0^{t_f} L[\boldsymbol{x}(t), \boldsymbol{u}(t), t]\,\mathrm{d}t \tag{18}$$

The function $\phi[\boldsymbol{x}(t_f), t_f]$ corresponds to the terminal time cost and $L[\boldsymbol{x}(t), \boldsymbol{u}(t), t]\,\mathrm{d}t$ is the incremental time cost.

In a supersonic flow, any perturbations or inputs given to the flow are felt only downstream of that point. The x coordinate along the flow therefore behaves like the time coordinate in the sense that any event occurring at time t affects the system only for times greater than t. Therefore, the x coordinate can be thought of as being equivalent to the time coordinate when we look at the simplified ordinary differential equation system corresponding to the steady-state one-dimensional flow.

In terms of this "x–t equivalence," we can note the equivalence between Eqs. (16) and (14) that describes the dynamics of the system. Therefore, we can apply standard optimal control theory, that has been extensively developed, with time as the independent coordinate to the MHD channel that has the position as the independent coordinate. We can prescribe the cost function to be optimized for the MHD channel in terms of the terminal and incremental costs equivalent to Eq. (18). The details of the definition of the performance measure are reserved for Sec. IV. Having defined an appropriate performance measure, we can then find an optimal controller equivalent to Eq. (17) to optimize the performance measure:

$$\boldsymbol{u}(x) = g[\boldsymbol{w}(x), x] \tag{19}$$

IV. Data-Based Optimal Control Design

The electron-beam current, as a function of position along the channel, acts as the control to the system. The electron-beam current allows for the number density of the electrons to be varied, which in turn affects the conductivity of the flow. The flow conductivity determines how much Faraday force is felt by the flow and also how much energy is extracted from the flow.

In this section, we briefly describe the control approach developed in previous work[13,14] and how the approach is applicable for the problem currently under consideration. The control architecture developed in previous work[13,14] uses neural networks to parameterize the control architecture.

A. Neural Networks

Neural networks are traditionally brain-inspired connectionist models that consist of many similar linear and nonlinear computational elements connected in patterns. Some of the interesting features of artificial neural networks are that they are able to learn, adapt, and generalize. Having learned to model a system, neural networks can be used to provide sensitivities of the system outputs with respect to the system inputs. This proves to be very useful information for the training of our control architecture.

Figure 2 shows a two-layer feedforward neural network. The first layer (or the hidden layer) consists of neurons that have sigmoid activation functions. The second layer (or the output layer) consists of neurons that have linear activation functions. A two-layer feedforward network with enough neurons in the hidden layer can approximate a nonlinear function to any degree of arbitrary accuracy.[17] The

network equations are as follows:

$$\boldsymbol{x}_{\mathrm{NN1}} = \boldsymbol{W}_{1,\mathrm{inp}}\boldsymbol{u}_{\mathrm{NN}} + \boldsymbol{b}_{\mathrm{NN1}}, \qquad \boldsymbol{y}_{\mathrm{NN1}} = \boldsymbol{f}_{\mathrm{NN1}}(\boldsymbol{x}_{\mathrm{NN1}})$$

$$\boldsymbol{x}_{\mathrm{NN2}} = \boldsymbol{W}_{2,1}\boldsymbol{y}_{\mathrm{NN1}} + \boldsymbol{b}_{\mathrm{NN2}}, \qquad \boldsymbol{y}_{\mathrm{NN2}} = \boldsymbol{f}_{\mathrm{NN2}}(\boldsymbol{x}_{\mathrm{NN2}}) \quad (20)$$

The activation functions $\boldsymbol{f}_{\mathrm{NN1}}$ and $\boldsymbol{f}_{\mathrm{NN2}}$ are

$$\boldsymbol{f}_{\mathrm{NN1}}(\boldsymbol{x}_{\mathrm{NN1}}) = 2\Big/\left(1 + e^{-2\boldsymbol{x}_{\mathrm{NN1}}}\right) - 1, \qquad \boldsymbol{f}_{\mathrm{NN2}}(\boldsymbol{x}_{\mathrm{NN2}}) = \boldsymbol{x}_{\mathrm{NN2}} \quad (21)$$

Reference 18 provides a comprehensive overview of the field of neural networks. Inspired by adaptive critic designs,[19] the design developed in previous work[13,14] is a predictive control scheme, which consists of two groups of such networks. The first group is used to model the cost function and the second group is used to model the controller.

B. Neural Network Cost Function Approximator

It can be noted from the definition of the cost function given by Eq. (18) that J can be estimated given the state of the system, $\boldsymbol{w}$, at the inlet of the channel, and the control $\boldsymbol{u}$ given to the system from the inlet to the final position x_f because the states of the system from the inlet through x_f are functions of these. The states of the system from the inlet to the channel exit need to be predicted, given the state at the inlet and the control profile along the channel. These estimated states along the channel, along with the control profile, then provide the cost function using an equation similar to Eq. (18). Figure 3 gives an example of the cost function estimator that is constructed using a group of neural networks known as subnets. Each subnet consists of a neural network as depicted in Fig. 2. Note that the index k now refers to the index of position, not time. The subnets correspond to multi-step-ahead predictors as shown in Fig. 3.

For the MHD channel, a single-step-ahead subnet (subnet 1) can correspond to behavior of the flow between one electron-beam window or a group of electron-beam windows that input the same electron-beam current. The physical picture corresponding to subnet 1 is shown in Fig. 4.

For the MHD system under consideration, the system dynamics depicted in Fig. 4 differs at every location in the channel. In other words, given certain flow variables at a particular location and given an applied electron-beam current, the output variables at the end of the section differ depending on the location of the application. This position dependency arises because the channel width varies with x and this has an effect on the flow dynamics. This can be inferred from Eqs. (7), (9), and (10), where Z_f, the width of the channel, enters these equations. The equivalence with our corresponding time-based picture lies in the system being *time-varying* vs *time-invariant*. Designing the subnets to have the x coordinate as another input along with the state and control elements can capture this effect. Therefore, the subnet depicting the system dynamics in Fig. 4 is modified to include position dependency, as shown in Fig. 5.

Higher-order subnets can be similarly designed using neural networks. Subnet m corresponds to an m-step-ahead model of the system that takes the state of the system at the location x, and m values of electron-beam currents ahead of this location, to generate the state of the system at the location $(x + m\Delta x)$. Figure 6 depicts the input and output variables for subnet m.

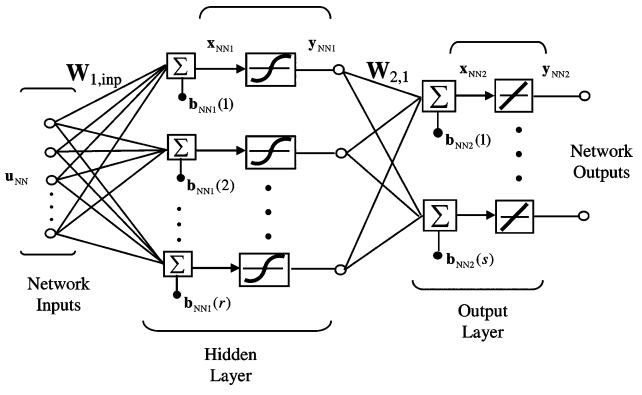

Fig. 2 A two-layer feedforward neural network.

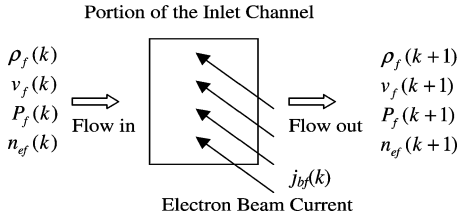

Fig. 4 Physical picture describing subnet 1.

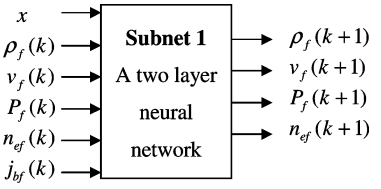

Fig. 5 Inputs and outputs of subnet 1.

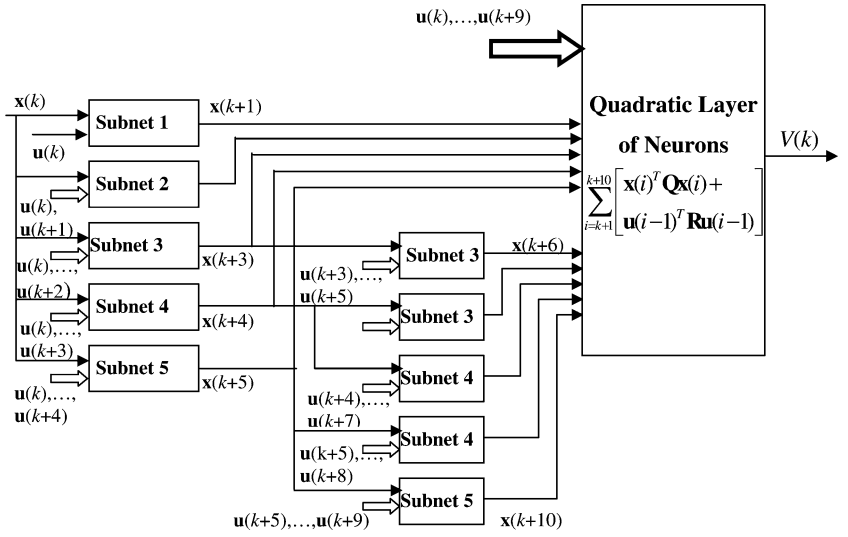

Fig. 3 Cascading individual multi-step-ahead subnets to compute $V(k)$.

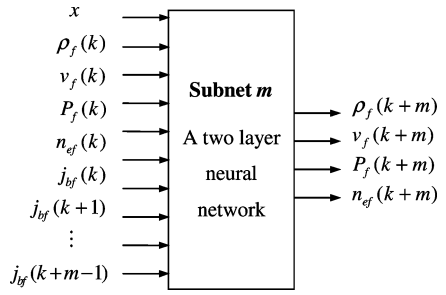

Fig. 6 Inputs and outputs of subnet *m*.

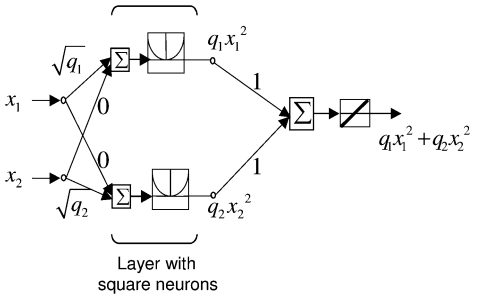

Fig. 7 Simple example of building the chosen functional form of the cost function.

After designing these subnets, they need to be trained to model the corresponding system dynamics using the available data. To do that, randomized inlet conditions are assumed in terms of the flight altitude and Mach number around the design condition. The electron-beam current values input through the individual windows are also randomized. This helps in covering the input space of the neural networks. With these random inlet conditions and electron-beam current values, Eq. (13) is integrated using the fourth-order Runge–Kutta scheme. The training data are created in this manner in a simulation study; in an actual system, these data can be obtained directly from experiments. This is one of the advantages of a data-based approach, where measured data instead of an analytical model of the system are used to derive the optimal controller. The data-based approach avoids errors present in an analytical model of the system, which could be a cause of concern in a model-based approach.

Because higher-order subnets are more difficult to train than the lower-order ones due to increasing input dimension, it is expected that the subnets can be reliably trained only up to a certain order. At this point, the trained subnets can be cascaded to generate the subnets of orders higher than those that are already trained, as shown in Fig. 3. The outputs of all these subnets are then hard-coded in terms of a neural network to produce the cost function network as depicted in Fig. 3. The cost function for this example is chosen to be a quadratic function of the states of the system and the applied control:

$$J = \int_0^{x_f} [\boldsymbol{w}(x)^T \boldsymbol{Q}\boldsymbol{w}(x) + \boldsymbol{u}(x)^T \boldsymbol{R}\boldsymbol{u}(x)]\,\mathrm{d}x \tag{22}$$

However, any other form of cost function can be chosen for the MHD channel based on the prescribed performance measure to be optimized. Figure 7 is a simple example of how a representative function can be hard-coded in terms of a neural network, given such subnet outputs. This representative network uses neurons with quadratic activation functions to generate the required result with proper weighting terms. Any algebraic equation can thus be coded in terms of such a network.

The cost function definition also involves the values of the state variables at the end position. Therefore, the trained subnets described earlier need to be cascaded together to be able to capture the end position system dynamics based on the inlet conditions and the electron-beam current profile along the entire length of the channel.

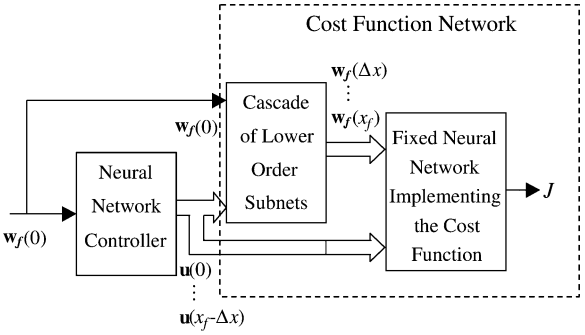

Fig. 8 Overall control architecture.

C. Neural Network Controller

After designing and training the cost function network, another neural network is attached before this network (Fig. 8). This second network is trained to become the optimal controller. The neural network controller takes the state of the system at the inlet as its input and produces electron-beam settings along the channel. For a randomized set of the state variables at the channel inlet, this controller can be trained to minimize the cost function over the entire length of the channel. (Refer to Ref. 14 for details of the minimization procedure.) This implementation therefore provides us with an open-loop controller that would prescribe the electron-beam current profile for a set of inlet conditions.

Figure 8 shows the entire architecture that shows the neural network controller in relation to the cost function approximator. As shown in the figure, the neural network controller takes the state of the system corresponding to the inlet flow variables, $\boldsymbol{w}(0)$, and produces the control corresponding to the electron-beam profile along the channel, $\boldsymbol{u}(0)$ through $\boldsymbol{u}(x_f - \Delta x)$. The details of the choice of the performance measure are discussed in the next section. The controller part of the network is trained to minimize the cost function for a range of initial states of the system, which correspond to a range of the MHD channel inlet conditions.

D. Performance Measure

To optimize the electron-beam current along the channel, a suitable performance measure similar to Eq. (18) needs to be defined. Some of the considerations for this definition are as follows:

1) We would like to maximize the net energy extracted from the system, which corresponds to the difference between the energy extracted from the flow and the energy spent on the electron-beam ionization.

2) A performance measure incorporating maximization of the extracted energy would typically lead to pressure profiles with an adverse pressure gradient. Although the simulation of boundary layers and their effects is beyond the scope of the present work, adverse pressure gradients would lead to boundary-layer separation. Therefore, the pressure profile should be such that it has as little adverse pressure gradient as possible.

3) The applied electron-beam current profile needs to be such that the flow conditions at the end of the inlet come close to certain prescribed values of temperature, pressure, and Mach number.

4) The entropy increase between the ends of the channel should be minimized.

5) While trying to achieve the aforementioned objectives, we would like to minimize the net usage of the electron-beam current.

One candidate cost function to be minimized that implements the aforementioned requirements is

$$J = p_1[T_f(x_f) - T_{\mathrm{fe}}]^2 + p_2[M_f(x_f) - M_{\mathrm{fe}}]^2 + \int_0^{x_f} \left\{ \begin{array}{c} \dfrac{q_1}{\rho_f v_f A_f}\left[Q_{\beta f} A_f - k_f(1-k_f)\sigma_f v_f^2 B_f^2 A_f\right] \\ + q_2 h(P_f) + q_3 \Delta S_f^2 + r_1 j_{\mathrm{bf}}^2 \end{array} \right\} \mathrm{d}x \tag{23}$$

It is instructive to take a term-by-term look at this cost function. The terms on the first line correspond to the end position cost. Minimizing the first and the second term corresponds to ensuring the exit temperature and Mach number are close to their prescribed values T_{fe} and M_{fe}, respectively. The scalars p_1 and p_2 indicate relative weighting between these two terms. The term on the second line corresponds to the incremental cost function. The first element of the integrand maximizes the difference between the power spent to ionize the flow and the power extracted from the flow by minimizing the minus of that difference. This term is weighted by a positive number, q_1. The second element of the integrand is a function (h) of the static pressure. This function is chosen such that it minimizes any significant pressure rise in the channel that corresponds to an adverse pressure gradient. This term is weighted by the positive number q_2. The next term in the integrand minimizes the entropy rise in the channel. The entropy rise between positions x and $(x + \Delta x)$ is given by

$$\Delta S_f = C_v \log\left[\frac{P_f(x+\Delta x)}{\rho_f(x+\Delta x)^{\gamma_m}}\right] - C_v \log\left[\frac{P_f(x)}{\rho_f(x)^{\gamma_m}}\right] \tag{24}$$

Finally, the last term in the integrand directly penalizes any excessive use of the electron-beam current. This is weighted by the positive number r_1. An appropriate choice of the weighting elements sets the relative importance of the different terms in the cost function. Having chosen such a representative cost function, we can hard-code it in terms of a neural network, as illustrated in Fig. 7 for a simple quadratic cost function.

V. Results

This section presents the results of the implementation of the optimal control algorithm on the MHD channel. The electron-beams are assumed to be applied through windows that are 0.5 cm wide and are placed next to each other continuously with no spacing between them. For our channel length of 1 m, this corresponds to 200 electron beams. We simplify the control implementation by grouping the windows into groups of four. Each window of a group has the same electron-beam current setting. For a given set of inlet conditions, the control algorithm therefore specifies 50 values of electron-beam current along the channel, one value for a group of four windows.

As explained in the preceding section, there are three steps in the implementation of the control algorithm: 1) training the single- and multi-step-ahead predictor subnets from the available system data, 2) assembling the subnets along with hard-coding the cost function layer with a particular choice of the weighting parameters to get the cumulative cost network, and 3) training the controller neural network to minimize the output of this cumulative cost network. Once trained, the controller will produce optimal electron-beam settings along the entire channel for any inlet condition in the prescribed range.

A. Subnet Training

The subnets are trained with data obtained by integrating Eq. (13) and using the fourth-order Runge–Kutta solver. For creating a rich training set, the solver was given random inlet conditions of altitude and Mach number, and random electron-beam current values for the 50 groups of electron-beam windows. The altitude and Mach number were chosen from a uniform distribution with a range of 10% around the design altitude of 30 km and design Mach number of 8. The electron-beam windows were assumed to allow a maximum current of 50 A/m^2. The electron-beam current was therefore chosen with a uniform distribution between 0 and 50. Having collected the simulation data, we divided it into input/output training sets for the subnets. Subnet 1, for example, gets the flow variables at the beginning of a particular group of four electron-beam windows; the location (x) of the first window of the group and the electron-beam current applied over that group as its inputs are shown in Fig. 4. Recall that through each window of the group is an electron beam and the same setting is used for each beam of the group. The outputs of subnet 1 are the flow variables at the end of those four windows.

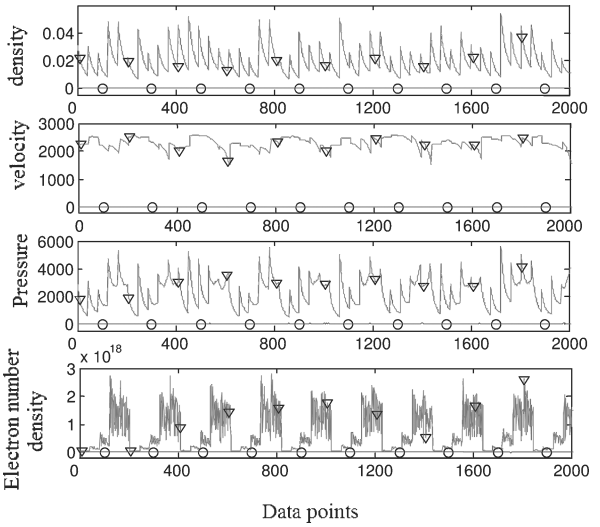

Fig. 9 Testing the prediction accuracy of subnet 10 (∇, subnet output; ○, difference between the subnet output and the exact value from simulation; units as given in the Nomenclature).

Similarly, the outputs of subnet m are the flow variables at the end of m groups. The input/output training data for subnet 1 are therefore created by going over the entire length of the channel and collecting data depicting the system behavior among four electron-beam windows. As the subnet order increases, the corresponding neural network receives more inputs. We found that it was computationally feasible to train subnets directly up to order 10. Beyond this point, the subnets should be stacked (cascaded) to generate prediction beyond 10 groups. This strategy will be discussed in more detail in the next subsection.

Figure 9 shows the training results for subnet 10. The errors between the subnet output and the exact value from the simulation lie in the range of 0.1 to 0.4%. Similar error levels are achieved for all other subnets from order 1 to 10. The neural networks are trained using the Levenberg–Marquardt algorithm[20] available in the MATLAB® neural network toolbox.

B. Building the Cost Function Approximator

Having trained the subnets, we arrange them together to produce the cumulative cost function approximator, which is given from Eq. (23) as

$$J = p_1[T_f(x_f) - T_{\text{fe}}]^2 + p_2[M_f(x_f) - M_{\text{fe}}]^2 + \sum_{i=1}^{50} \times \left\{ \begin{aligned} &\frac{q_1}{\rho_f(i)v_f(i)A_f(i)}[Q_{\beta f}(i)A_f(i) \\ &\quad - k_f(1-k_f)\sigma_f(i)v_f(i)^2 B_f^2 A_f(i)] \\ &\quad + q_2(i)P_f(i) + q_3[S_f(i) - S_f(i-1)]^2 + r_1 j_{\text{bf}}(i-1)^2 \end{aligned} \right\} \Delta x \tag{25}$$

where the integration is replaced by a summation. The cost function involves the flow variables from the beginning to the end of the channel; however, the subnets are trained up to an order of 10, which corresponds to a prediction for one-fifth of the section. For a given electron-beam profile, the flow variables in the later part of the channel are predicted by stacking these 10 subnets five times. For example, Fig. 3 illustrates the stacking of 5 trained subnets two times to get an equivalent subnet of order 10. Thus, 5 sets of 10 subnets provide the flow variables at 50 locations from the beginning to the end of the channel. These values, along with the 50 electron-beam current values, are given to a fixed layer of neurons that are hard-coded to calculate the cumulative cost function given by Eq. (25).

Figure 10 compares the output of the cumulative cost function approximator with the actual cumulative cost computed using

Table 1 Weighting parameters for the comparison of the true and the predicted values of the cumulative cost function

Parameter	p_1	p_2	q_1	q_3	r_1
Value	0.0001	100	0.0001	0	0.005

Table 2 Weighting parameters that maximize the power output and penalize excess use of the electron-beam current

Parameter	p_1	p_2	q_1	q_3	r_1
Value	0	0	0.0001	0	0.005

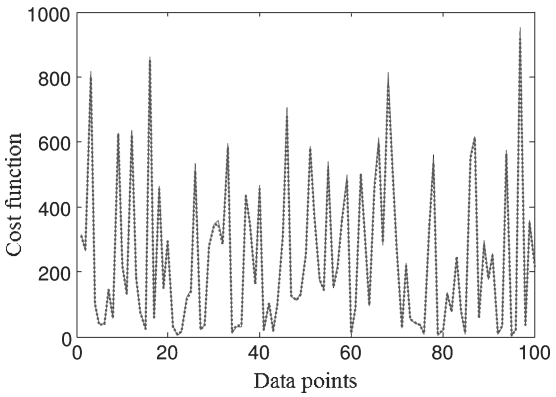

Fig. 10 Testing the CGA neural network (——, CGA output; · · · exact value from simulation).

simulation. We can see that the groups of subnets arranged together produce excellent approximation of the cumulative cost. The values of the weighting parameters are given in Table 1.

The functional form of q_2 as seen in Table 1 is chosen as

$$q_2(x) = 0, \qquad 0 < x < 0.9$$

$$q_2(x) = 200x^4, \qquad 0.9 < x < 1 \tag{26}$$

The details for this choice of $q_2(x)$ follow in a later section.

C. Controller Training

Here we examine the optimal controllers for three different cases corresponding to giving priorities to three different terms in Eq. (25). The objective is to show the ability of the optimal controller to meet a particular performance criterion.

Case 1

In this case we look at maximizing the power output from the system while avoiding excessive use of the electron-beam current. The power output from the system corresponds to the first term in the summation term of Eq. (25) with weighting q_1. The final term in the summation term penalizes excess usage of the electron-beam current with the use of a nonzero value of r_1. The weighting parameters for this cost function are therefore chosen as given in Table 2.

Figure 11 shows that the cumulative cost is minimized as the neural network controller is trained for a range of inlet conditions. The controller weights are updated using the resilient backpropagation algorithm[21] provided in the MATLAB neural network toolbox. The details of the controller training that minimize the cost function are illustrated in Ref. 14.

The results obtained for this choice of parameters for the optimal neural network controller are compared with two different control profiles: one corresponds to using the maximum value of 50 A/m^2 throughout the length of the channel and the second corresponds to using a random electron-beam profile between 0 and 50 A/m^2. Figure 12 shows the plots of the system states $\boldsymbol{w}(x)$ vs x. Figure 13 shows the Mach number profile and the electron-beam current profile.

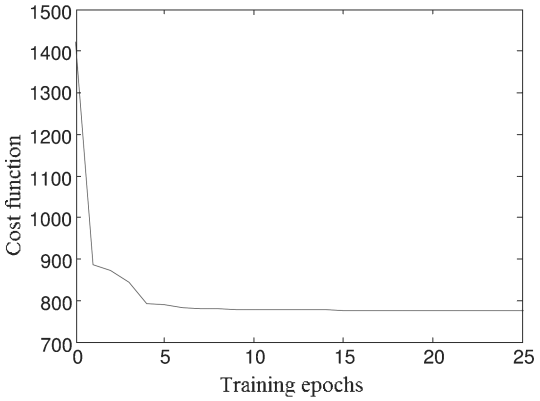

Fig. 11 Training the optimal controller to minimize the cost function.

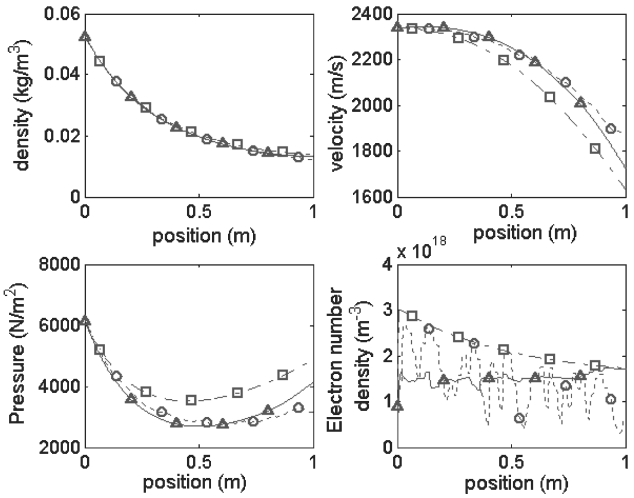

Fig. 12 System states for different electron-beam current profiles (□, constant electron-beam current of 50 A/m^2; ○, random profile; △, neural network controller).

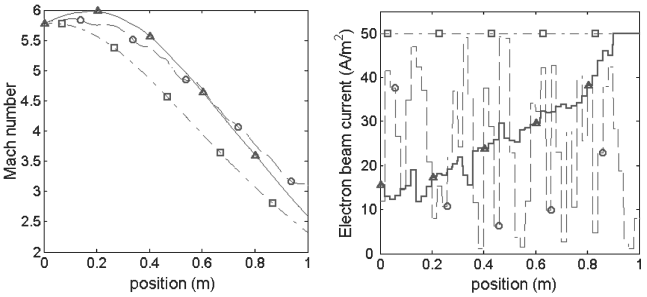

Fig. 13 Mach-number output and the electron-beam current profile (□, constant electron-beam current (50 A/m^2); ○, random profile; △, neural network controller).

From Table 3, it can be observed that a constant current profile of 50 A/m^2 extracts the highest power but consumes the most power. The random current profile uses the least power but also delivers the least power. The optimal electron-beam current profile balances both extremes. The amount of power extracted is close to that extracted by the constant current profile, but the amount of power used is significantly less.

Case 2

In the preceding case it was observed that, although good results can be obtained in terms of the power output, the resulting pressure profile could lead to some unwanted effects such as boundary-layer separation. In this case, therefore, we weight the pressure term in Eq. (25) to tackle the adverse pressure gradient. The functional form

Table 3 Power input/output for the three control profiles

Profile	Power spent, kW	Power extracted, MW	Net power extracted, MW
Constant current (50 A/m^2)	300	1.918	1.618
Random current	121	1.381	1.260
Optimal	174	1.717	1.544

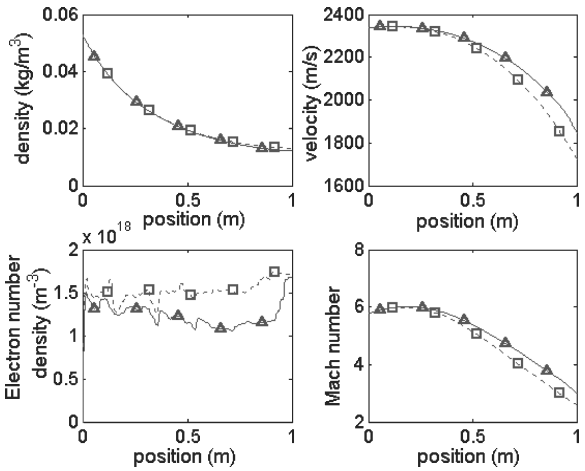

Fig. 14 System states with and without pressure weighting (□, without pressure weighting; △, with pressure weighting).

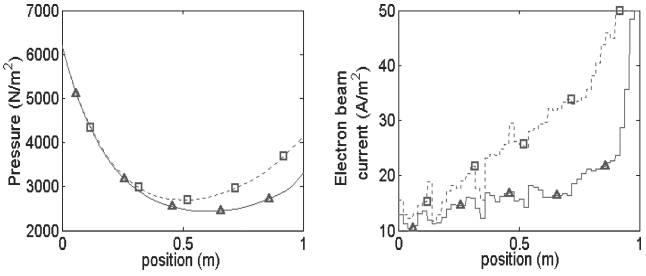

Fig. 15 Resulting pressure distribution and the electron-beam current profile along the channel (□, without pressure weighting; △, with pressure weighting).

of q_2 is chosen as given in Eq. (26). Higher pressure at the latter portion of the channel is penalized by a weighting function that penalizes the pressure profile for the last 10% of the channel length as given in Eq. (26). This choice follows from the observation of the adverse pressure gradient toward the end of the channel. Figure 14 shows the system outputs with and without the pressure weighting term in the cost function. Figure 15 shows the resulting pressure profile and the corresponding electron-beam current profile. Observe that the pressure profile without the pressure weighting term has a steep adverse pressure gradient, which is now reduced by the addition of the pressure weighting term in the cost function.

Case 3

We now illustrate the ability of the control algorithm to bring the flow variables to certain prescribed values at the end of the channel. This provision is made available through the first two terms in Eq. (25). Our simulation study indicates that a random electron-beam current profile typically results in a Mach number of between 3 and 3.5 at the end of the channel. In this illustration, an exit Mach number of 4.5 is prescribed. The objective of the control system is to bring the exit Mach number to this value for different inlet conditions. The weighting parameters are given in Table 4. Note the high value of p_2, corresponding to large weighting on any difference in the exit Mach number from the prescribed value.

Table 4 Weighting parameters for getting a prescribed exit Mach number

Parameter	p_1	p_2	q_1	q_2	q_3	r_1
Value	0	100	10^{-6}	0	0	0

Table 5 Exit Mach number for different initial freestream conditions

Freestream altitude, km	Freestream Mach number	Exit Mach number	Legend in Fig. 16
30	8	4.41	△
31.5	7.6	4.58	□
28.5	8.4	4.52	○
31.5	8.4	4.51	▽
28.5	7.6	4.51	∗

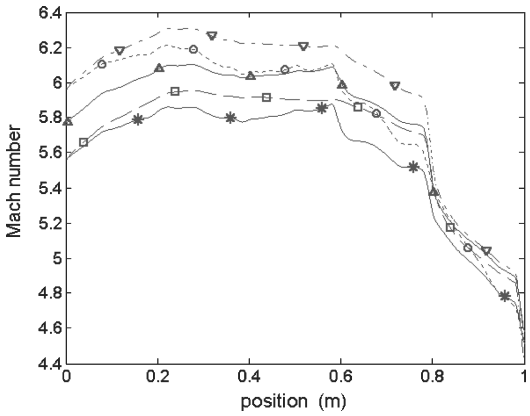

Fig. 16 Mach-number profiles for different freestream conditions (symbols described in Table 5).

Table 5 gives the details of these different initial conditions shown in the plots and the corresponding exit Mach number for each of them. With all these different initial conditions, we see in Fig. 16 that the Mach number gets very close to the value of 4.5.

Here we have considered that we have a single sensor that gives us the state information at the inlet using that which the controller gives the electron-beam current profile along the entire channel. In our subsequent work,[22] we assume multiple internal sensors and design controllers that use information from all these sensors to provide a feedback controller that can handle actuator failures. We also consider adapting the controller for looking at modeling errors.[23]

VI. Conclusions

In this work, we analyzed the performance optimization of an MHD generator at the inlet of a scramjet engine as an optimal control problem using a data-based optimal controller design algorithm. It was shown that the neural-network-based algorithm was successfully able to optimize the system performance, taking into account many important factors such as energy extraction, exit Mach number and temperature, entropy rise along the channel, adverse pressure gradient, energy spent for electron-beam ionization. After a particular cost function is specified, the controller can be trained. Once trained, it can be used directly to specify how the electron-beam current profile along the channel should be adjusted to maximize a user-defined performance measure for different inlet conditions. In this study, the optimal controller is only a function of the inlet conditions. This is perhaps the most practical situation because the strategy does not require sensors along the channel. In our subsequent work, we consider cases where internal sensors are available and how they can be used to further improve the system performance along with adapting the controller for modeling errors.

Acknowledgments

This research is supported by a grant from the National Science Foundation through ANSER Corporation. The authors acknowledge the support of Richard Miles and his MHD research group at Princeton University. In particular, we are grateful to Mikhail Shneyder for his assistance in the modeling of the MHD channel.

References

[1]Chase, R. L., Mehta, U. B., Bogdanoff, D. W., Park, C., Lawrence, S., Aftosmis, M., Macheret, S. O., and Shneyder, M. N., "Comments on a MHD-Bypass Spaceliner Performance," AIAA Paper 99-4965, Nov. 1999.

[2]Fraishtadt, V. L., Kuranov, A. L., and Sheikin, E. G., "Use of MHD Systems in Hypersonic Aircraft," *Technical Physics*, Vol. 43, No. 11, 1998, p. 1309.

[3]Gurijanov, E. P., and Harsha, P. T., "AJAX: New Directions in Hypersonic Technology," *7th AIAA International Spaceplanes and Hypersonic Technologies Conference*, Norfolk, VA, Nov. 1996; also AIAA Paper 96-4609, 1996.

[4]Bityurin, V. A., Lineberry, J. T., Potebnia, V. G., Alferov, V. I., Kuranov, A. L., and Sheikin, E. G., "Assessment of Hypersonic MHD Concepts," *28th AIAA Plasmadynamics and Lasers Conference*, Atlanta, 23–25 June 1997; also AIAA Paper 97-2393, 1997.

[5]Bityurin, V. A., Klimov, A. I., Leonov, S. B., Bocharov, A. N., and Lineberry, J. T., "Assessment of a Concept of Advanced Flow/Flight Control for Hypersonic Flights in Atmosphere," *3rd AIAA Weakly Ionized Gases Workshop*, Norfolk, VA, 28 June–1 July 1999; also AIAA Paper 99-4820, 1999.

[6]Brichkin, D. I., Kuranov, A. L., and Sheikin, E. G., "MHD Technology for Scramjet Control," *8th AIAA International Spaceplanes and Hypersonic Technologies Conference*, Norfolk, VA, 27–30 April 1998; also AIAA Paper 98-1642, 1998.

[7]Kuranov, A. L., and Sheikin, E. G., "The Potential of MHD Control for Improving Scramjet Performance," *30th AIAA Plasmadynamic and Lasers Conference*, Norfolk, VA, 28 June–1 July 1999; also AIAA Paper 99-3535, 1999.

[8]Vatazhin, A., Kopchenov, V., and Gouskov, O., "Some Estimations of Possibility of Use of MHD Control for Hypersonic Flow Deceleration," *3rd AIAA Weakly Ionized Gases Workshop*, Norfolk, VA, 28 June–1 July 1999; also AIAA Paper 99-4972, 1999.

[9]Rosa, R. J., *Magnetohydrodynamic Energy Conversion*, Hemisphere, Washington, DC, 1967.

[10]Macheret, S. O., Shneyder, M. N., and Miles, R. B., "Magnetohydrodynamic Power Extraction from Cold Hypersonic Airflows with External Ionizers," *Journal of Propulsion and Power*, Vol. 18, No. 2, 2002, pp. 424–443.

[11]Macheret, S. O., Shneyder, M. N., and Miles, R. B., "Electron Beam Genrated Plasmas in Hypersonic Magnetohydrodynamic Channels," *AIAA Journal*, Vol. 39, No. 6, 2001, pp. 1127–1138.

[12]Macheret, S. O., Shneyder, M. N., and Miles, R. B., "Potential Performance of Supersonic MHD Power Generators," AIAA Paper 2001-0795, Jan. 2001.

[13]Kulkarni, N. V., and Phan, M. Q., "Data-Based Cost-to-Go Design for Optimal Control," AIAA Paper 2002-4668, Aug. 2002.

[14]Kulkarni, N. V., and Phan, M. Q., "A Neural Networks Based Design of Optimal Controllers for Nonlinear Systems," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 5, 2004, pp. 745–751.

[15]Stengel, R. F., *Optimal Control and Estimation*, Dover, New York, 1994.

[16]Shneyder, M. N., Macheret, S. O., and Miles, R. B., "Nonequilibrium Magnetohydrodynamic Control of Scramjet Inlets," *33rd Plasmadynamics and Lasers Conference*, Maui, Hawaii, 20–23 May 2003; also AIAA Paper 2002-2251, 2002.

[17]Cybenko, G., "Approximation by Superposition of Sigmoidal Function," *Mathematics of Control, Signals, and Systems*, Vol. 2, No. 4, 1989, pp. 303–314.

[18]Haykin, S., *Neural Networks: A Comprehensive Foundation*, Prentice–Hall, Upper Saddle River, NJ, 1999.

[19]Werbos, P. J., "Approximate Dynamic Programming for Real Time Control and Neuro-modeling," *Handbook of Intelligent Control*, edited by D. A. White and D. A. Sofge, Van Nostrand Reinhold, New York, 1992.

[20]Hagan, M. T., and Menhaj, M. B., "Training Feedforward Networks with the Marquardt Algorithm," *IEEE Transactions on Neural Networks*, Vol. 5, No. 6, 1994, pp. 989–993.

[21]Reidmiller, M., and Braun, H., "A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm," *Proceedings of the IEEE International Conference on Neural Networks (ICNN)*, CA, 1993, pp. 586–591.

[22]Kulkarni, N. V., and Phan, M. Q., "Optimal Feedback Control of the Magneto-Hydrodynamic Generator for a Hypersonic Vehicle," AIAA Paper 2003-5497, Aug. 2003.

[23]Kulkarni, N. V., and Phan, M. Q., "Data-Based Adaptive Predictive Control with Application to In-Flight MHD Power Generation," AIAA Paper 2004-6221, Sept. 2004.