

# Burn conditions stabilization with artificial neural networks of subignited thermonuclear reactors with scaling law uncertainties

Javier E Vitela and Julio J Martinell

Instituto de Ciencias Nucleares, Universidad Nacional Autónoma de México, 04510 México D.F., México

Received 23 February 2000, in final form 15 June 2000

## Abstract

In this work it is demonstrated that robust burn control in long-pulse operations of subignited thermonuclear reactors can be achieved with radial basis neural networks (RBNNs) composed of Gaussian nodes in the hidden layer and sigmoidal units in the output layer. The results reported here correspond to a volume-averaged zero-dimensional nonlinear model of a subignited fusion reactor with design parameters corresponding to those of the ITER-EDA group. The control actions are implemented through the concurrent modulation of the D–T refuelling rate, a neutral  $^4\text{He}$  beam and an auxiliary heating power, constrained to lie below maximum allowable levels.

It is shown that the resulting network provides feedback stabilization over a wide range of energy confinement times for plasma density and temperature excursions significantly far from their nominal operating values. The results show that the RBNN feedback-controlled nonlinear system is stable regardless of any particular scaling law, as long as the confinement time lies within the scope of the training region. In addition, it also shows robustness with respect to noise in the energy confinement time value fed into the controller during simulated transients of a thermonuclear system using a particular ELMy scaling law, as well as with respect to the thermalization time of the alpha particles produced by fusion.

## 1. Introduction

In an earlier work [1] the authors demonstrated the feasibility of using feedforward artificial neural networks to stabilize the burn conditions of a thermonuclear system represented by a nonlinear zero-dimensional model, where the reactor parameters corresponded to those of the ITER-CDA tokamak. The resulting neural network was capable of stabilizing the nonlinear thermonuclear reactor model at nearly ignited conditions for density and temperature excursions significantly far from the nominal operating point and, in addition, it showed robustness with respect to the thermalization time of the alpha particles produced by the fusion

reactions. In that paper, the energy and particle transport were taken into account through the ITER90H-P scaling law.

In general, particle and energy transport losses are extremely difficult to model and predict because they are driven by highly nonlinear turbulent processes. Thus, for design studies of magnetic fusion machines these losses are modelled through scaling laws extrapolated using a database gathered from past and current experimental devices. As a consequence the confinement time predicted by these scaling laws may suffer from significant uncertainties. The purpose of this work is to demonstrate that a thermonuclear reactor subject to scaling law uncertainties can be stabilized using artificial neural networks. Furthermore, we do not constrain the system to a specific scaling law embedded in the dynamical equations; instead we generalize the previous work training the neural network to stabilize the system at fixed plasma parameters independently of the scaling law of the device. In this work we use radial basis neural networks (RBNNs) [2] instead of the standard feedforward neural networks.

Several related works using traditional control design techniques were pointed out in [1]. More recently, some authors have proposed a set of diagnostics for burn control and studied the effect of diagnostic failure in the ignited operation of tokamaks using standard constant gain controllers [3]. In [3] the need to impose limits in the D–T refuelling rate was stressed in order to avoid oscillations in the plasma parameters.

The reactor model we use here is an extension of the volume-averaged zero-dimensional nonlinear model used previously [1, 4]. It is composed of a quasineutral plasma system consisting of a 50:50 D:T fuel of density  $n_{DT}$ , fully ionized  $^4\text{He}$  ions with density  $n_\alpha$ , a small fraction of high- $Z$  impurities (density  $n_I$ ) and free electrons with density  $n_e$ , all having Maxwellian distributions sharing the same temperature at all times. The energy and particle transport losses are accounted for through the energy confinement time  $\tau_E$ , as well as by the D–T and the helium ash confinement times,  $\tau_p$  and  $\tau_\alpha$ , respectively. Similarly to other studies, this model assumes that the high- $Z$  impurities density  $n_I$  and its charge  $Z_I$  remain constant at all times [5]. Since it has been shown that synchrotron radiation is negligible at the operating temperatures of the current device designs, bremsstrahlung is the only radiation loss mechanism included [6].

The plasma heating is considered to take place mainly through the thermalization of the alpha particles produced by the fusion reactions and an auxiliary heating power which is provided, for instance, by the appropriate electromagnetic waves; for completeness ohmic heating is also included. Although instantaneous thermalization of the alpha particles produced by the D–T fusion reactions is assumed, the effect of finite thermalization times on the stability properties of the joint neural network–thermonuclear system is also studied.

On the other hand, artificial intelligence techniques in the form of neural networks has been an active field of pure and applied research since the early 1980s. In particular multilayer feedforward neural networks with sigmoidal activation units found immediate applications in pattern recognition tasks, and due to their exceptional nonlinear mapping capabilities these networks also originated a great deal of interest in nonlinear dynamical system identification and control [7, 8]. However, these networks show several drawbacks: a low speed of convergence in their training procedures in spite of the different acceleration techniques available; the burden of determining the topology of the hidden layers; and the presence of multiple local minima, which affects the effectiveness of any neural network learning scheme. Among alternative types of artificial neural networks is the radial basis network. In general the convergence speed of this network is higher than the standard feedforward multilayered neural network and under some conditions in the learning pattern's environment, the cost function is local minima free [9].

Hence RBNNs will be considered here to provide feedback stabilization under sub-ignited burn conditions of a thermonuclear reactor using the parameters of the EDA-ITER tokamak design group [10]. Furthermore, as mentioned above, instead of restraining us to a specific scaling law for the energy confinement time, we consider the energy confinement time as an input parameter to the RBNN and train the neural network to provide feedback stabilization for a wide range of energy confinement times. The control actions include the concurrent modulation of an auxiliary heating power source, the D–T refuelling rate and the injection of a  $^4\text{He}$  neutral beam, to which maximum and minimum levels have been imposed; similarly to previous works it is assumed that no delay occurs in the refuelling and in the auxiliary heating power action on the plasma.

Due to the high computing costs involved, the RBNN was trained using a parallel training code developed using MPI, a portable standard message passing environment, whose structure is similar to that reported in [11] but modified to work with RBNNs instead of the standard feedforward multilayer neural networks.

The rest of the paper is organized as follows. In section 2 the mathematical model of the thermonuclear reactor used in this work is established, the nominal operating conditions are obtained and the behaviours of the control variables as functions of the energy confinement time when the system is in a steady state are determined. In section 3 the RBNN used in this work is discussed and the training strategy is sketched out. In section 4 the results from the training process are shown. In section 5 results are shown concerning the robustness tests when the thermonuclear system is assumed to follow an ELM scaling law previously unseen by the network, and where the effects of the thermalization time of the alpha particles as well as the noise in the energy confinement time measurements are considered. Finally, section 6 contains some concluding remarks concerning this work. The appendix contains the parallel training strategy as well as the dynamic backpropagation algorithm used to develop the RBNN controller.

## 2. Thermonuclear reactor model

The tokamak fusion reactor model used here is a quasineutral zero-dimensional plasma system composed of a 50:50 D–T mixture, fully ionized helium ions, a small fraction of high-Z impurities and electrons. All particles in the system are taken to be at the same temperature, and the alpha particles produced by the fusion reactions are assumed to be instantaneously thermalized. As mentioned previously bremsstrahlung is the only radiation loss mechanism considered and the transport losses are taken into account through the energy confinement time  $\tau_E$ , as well as by the D–T and the alpha particles confinement times,  $\tau_p$  and  $\tau_\alpha$  respectively. With these assumptions the following set of coupled nonlinear differential equations account for the balance of the D–T fuel, helium ash and the thermal energy densities, respectively:

$$\frac{d}{dt}n_{\text{DT}} = S_f - 2\left(\frac{n_{\text{DT}}}{2}\right)^2 \langle \sigma v \rangle - \frac{n_{\text{DT}}}{\tau_p} \quad (1)$$

$$\frac{d}{dt}n_\alpha = S_\alpha + \left(\frac{n_{\text{DT}}}{2}\right)^2 \langle \sigma v \rangle - \frac{n_\alpha}{\tau_\alpha} \quad (2)$$

$$\begin{aligned} \frac{d}{dt} \left[ \frac{3}{2}(n_e + n_{\text{DT}} + n_\alpha + n_I)T \right] = & P_{\text{aux}} + Q_\alpha \left(\frac{n_{\text{DT}}}{2}\right)^2 \langle \sigma v \rangle + \eta j^2 - A_B Z_{\text{eff}} n_e^2 T^{1/2} \\ & - \frac{3}{2}(n_e + n_{\text{DT}} + n_\alpha + n_I) \frac{T}{\tau_E}. \end{aligned} \quad (3)$$

Here it is assumed that the particle density,  $n_I$ , as well as the charge,  $Z_I$ , of the impurities remain constant at all times. In the above equations  $Q_\alpha = 3.5$  MeV is the energy carried by the fusion alpha particles,  $\langle\sigma v\rangle$  is the D–T reactivity,  $A_B$  is the coefficient of the bremsstrahlung radiation losses [12] and  $\eta$  is the neoclassical parallel resistivity accounting for the ohmic heating by the plasma current [13]. The control actions are represented by  $S_f$  the refuelling rate,  $S_\alpha$  the neutral  $^4\text{He}$  injection rate and  $P_{\text{aux}}$  the injection rate of the auxiliary heating power density.

Using the quasineutrality condition  $n_e = n_{\text{DT}} + 2n_\alpha + Z_I n_I$ , the above equations can be transformed into the following coupled set of equations for the electron density  $n_e$ , the relative fraction of helium ash  $f_\alpha = n_\alpha/n_e$  and the plasma temperature:

$$\frac{dn_e}{dt} = S_f - \left( \frac{2f_\alpha}{\tau_\alpha} + \frac{1-2f_\alpha}{\tau_p} \right) n_e + \frac{Z_I n_I}{\tau_p} + 2S_\alpha \quad (4)$$

$$\begin{aligned} \frac{df_\alpha}{dt} = & \frac{1}{4} n_e \left( 1 - 2f_\alpha - \frac{Z_I n_I}{n_e} \right)^2 \langle\sigma v\rangle - \frac{f_\alpha Z_I n_I}{n_e \tau_p} - \frac{f_\alpha}{n_e} S_f + f_\alpha (1 - 2f_\alpha) \left( \frac{1}{\tau_p} - \frac{1}{\tau_\alpha} \right) \\ & + S_\alpha \frac{1}{n_e} (1 - 2f_\alpha) \end{aligned} \quad (5)$$

$$\begin{aligned} \frac{dT}{dt} = & \left( \frac{1}{6} Q_\alpha + \frac{1}{4} T \right) n_e \langle\sigma v\rangle \frac{(1 - 2f_\alpha - Z_I n_I/n_e)^2}{2 - f_\alpha - (Z_I - 1)n_I/n_e} + 2 \frac{(1 - 2f_\alpha - Z_I n_I/n_e)}{2 - f_\alpha - (Z_I - 1)n_I/n_e} \frac{T}{\tau_p} \\ & - \frac{2}{3} A_B \frac{(1 + 2f_\alpha + Z_I(Z_I - 1)n_I/n_e)}{2 - f_\alpha - (Z_I - 1)n_I/n_e} n_e T^{1/2} - \frac{T}{\tau_E} \\ & + \frac{3f_\alpha}{2 - f_\alpha - (Z_I - 1)n_I/n_e} \frac{T}{\tau_\alpha} + \frac{2}{3} A_h \frac{(1 + 2f_\alpha + Z_I(Z_I - 1)n_I/n_e)^{0.5}}{n_e (2 - f_\alpha - (Z_I - 1)n_I/n_e) T^{3/2}} \\ & \times \frac{1 + 1.198(1 + 2f_\alpha + Z_I(Z_I - 1)n_I/n_e)^{0.5} + 0.222(1 + 2f_\alpha + Z_I(Z_I - 1)n_I/n_e)}{1 + 2.966(1 + 2f_\alpha + Z_I(Z_I - 1)n_I/n_e)^{0.5} + 0.75(1 + 2f_\alpha + Z_I(Z_I - 1)n_I/n_e)} \\ & - \frac{T}{n_e (2 - f_\alpha - (Z_I - 1)n_I/n_e)} (2S_f + 3S_\alpha) + \frac{2}{3} \frac{P_{\text{aux}}}{n_e (2 - f_\alpha - (Z_I - 1)n_I/n_e)}. \end{aligned} \quad (6)$$

In the above equations the parallel neoclassical resistivity was explicitly written in terms of a coefficient  $A_h$ , the temperature and particle densities [13]. For the purpose of this work it will be assumed that  $\tau_p = 3\tau_E$  and  $\tau_\alpha = 5.5\tau_E$ , and the analytical expression used for the D–T reactivity  $\langle\sigma v\rangle$  is due to Hively [14].

The nominal operating state was determined as the ignited steady-state condition, i.e.  $P_{\text{aux}} = 0$  and  $S_\alpha = 0$ , corresponding to the EDA-ITER design parameters assuming an ELM-free energy confinement scaling law [15]:

$$\tau_E = 0.031 I^{0.95} B^{0.25} P^{-0.67} n^{0.35} R^{1.92} \epsilon^{0.08} \kappa^{0.63} M^{0.42}. \quad (7)$$

Imposing the fact that the desired steady-state operating value of the electron density is  $n_0 = 1.0 \times 10^{20} \text{ m}^{-3}$ , the ignited condition turns out to be obtained with  $T_0 = 12$  keV,  $f_0 = 0.09$  and a D–T refuelling rate of  $S_0 = 3.58 \times 10^{18} \text{ m}^{-3} \text{ s}^{-1}$ , where (7) yields an energy confinement time value of  $\tau_E = 7.65$  s; in the above calculations it was assumed that the alpha heating efficiency is 96% and the high- $Z$  impurity density was taken as  $n_I = 7.0 \times 10^{17} \text{ m}^{-3}$  with a charge  $Z_I = 14.7$ . The above values of the plasma parameters, i.e. temperature, electron density and fraction of helium ash will constitute the nominal operating point for the sub-ignited tokamak reactors we are concerned with in this work. With these values we define the following normalized state variables

$$z_1 \equiv n_e/n_0 \quad z_2 \equiv f_\alpha/f_0 \quad \text{and} \quad z_3 \equiv T/T_0 \quad (8)$$

and transform equations (4)–(6) into

$$\frac{d}{dt}z_1 = \hat{S}_f + 2f_0\hat{S}_\alpha - \left( \frac{2f_0}{\tau_\alpha}z_2 + \frac{1-2f_0z_2}{\tau_p} \right)z_1 + \frac{k_2}{\tau_p} \quad (9)$$

$$\begin{aligned} \frac{d}{dt}z_2 = & \frac{1}{4} \frac{n_0}{f_0} (1 - 2f_0z_2 - k_2/z_1)^2 z_1 \langle \sigma v \rangle + z_2 (1 - 2f_0z_2) \left[ \frac{1}{\tau_p} - \frac{1}{\tau_\alpha} \right] - \frac{k_2}{z_1} \frac{z_2}{\tau_p} - \hat{S}_f \frac{z_2}{z_1} \\ & + \frac{1 - 2f_0z_2}{z_1} \hat{S}_\alpha \end{aligned} \quad (10)$$

$$\begin{aligned} \frac{d}{dt}z_3 = & \left[ \frac{1}{6} \frac{Q_\alpha}{T_0} + \frac{1}{4} z_3 \right] \frac{(1 - 2f_0z_2 - k_2/z_1)^2}{2 - f_0z_2 - k_3/z_1} n_0 \langle \sigma v \rangle z_1 \\ & + \frac{2}{3} A_h \frac{T_0^{-5/2}}{n_0 (2 - f_0z_2 - k_3/z_1)} \left( \frac{z_3}{z_1} \right)^{-3/2} (1 + 2f_0z_2 + k_1/z_1)^{1/2} \\ & \times \left[ \frac{1.0 + 1.198(1 + 2f_0z_2 + k_1/z_1)^{1/2} + 0.222(1 + 2f_0z_2 + k_1/z_1)}{1.0 + 2.966(1 + 2f_0z_2 + k_1/z_1)^{1/2} + 0.75(1 + 2f_0z_2 + k_1/z_1)} \right] \\ & - \frac{2}{3} A_B n_0 \left( \frac{1 + 2f_0z_2 + k_1/z_1}{2 - f_0z_2 - k_3/z_1} \right) T_0^{-1/2} z_1^{1/2} z_3^{1/2} - \frac{1}{\tau_E} z_3 \\ & + 2 \left( \frac{1 - 2f_0z_2 - k_2/z_1}{2 - f_0z_2 - k_3/z_1} \right) \frac{z_3}{\tau_p} + \frac{3f_0z_2}{2 - f_0z_2 - k_3/z_1} \frac{z_3}{\tau_\alpha} \\ & + \frac{1}{z_1 (2 - f_0z_2 - k_3/z_1)} \hat{P}_{\text{aux}} - \frac{z_3}{z_1 (2 - f_0z_2 - k_3/z_1)} [2\hat{S}_f + 3f_0\hat{S}_\alpha] \end{aligned} \quad (11)$$

which are the coupled set of dynamical equations for the normalized state variables. In the above set of equations we used

$$\hat{S}_f \equiv S_f/n_0 \quad \hat{S}_\alpha \equiv S_\alpha/f_0n_0 \quad \text{and} \quad \hat{P}_{\text{aux}} \equiv 2P_{\text{aux}}/3n_0T_0 \quad (12)$$

as the normalized D–T refuelling rate, normalized source of neutral helium atoms and auxiliary heating power density, respectively. In addition we defined

$$\begin{aligned} k_1 & \equiv Z_1(Z_1 - 1)n_1/n_0 \\ k_2 & \equiv Z_1n_1/n_0 \end{aligned}$$

and

$$k_3 \equiv (Z_1 - 1)n_1/n_0. \quad (13)$$

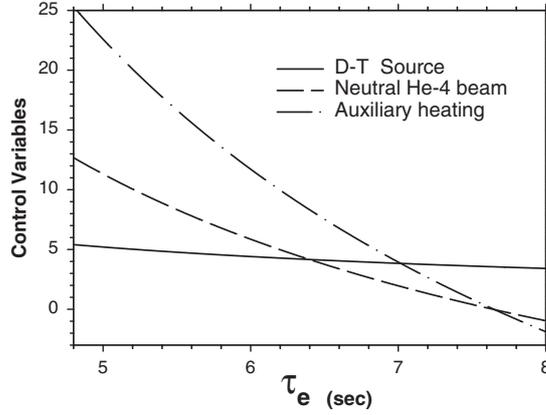
Equations (1)–(3) can be solved for the control variables that yield the dynamical system in a steady state at the above nominal operating values when a constant value of the energy confinement time  $\tau_E$  in seconds is given, to obtain

$$\begin{aligned} S_f & = (0.428 + 23.89/\tau_E) \times 10^{18} \text{ m}^{-3} \text{ s}^{-1} \\ S_\alpha & = (-0.214 + 1.636/\tau_E) \times 10^{18} \text{ m}^{-3} \text{ s}^{-1} \end{aligned}$$

and

$$P_{\text{aux}} = (-0.427 + 3.264/\tau_E) \times 10^{21} \text{ keV m}^{-3} \text{ s}^{-1}. \quad (14)$$

In figure 1, the above expressions are plotted as functions of  $\tau_E$ . It is observed that all control variables decrease monotonically with  $\tau_E$ . This is so because it is assumed that, regardless of the particular confinement time value, the steady state must always be achieved with the same plasma parameters, i.e. the same particle densities and plasma temperature; hence, all non-transport processes such as the fusion reactions rate and the radiation losses remain unchanged, while the different transport losses, which depend on the value of  $\tau_E$ ,



**Figure 1.** Values of the refuelling rate  $S_f \times 10^{-18}$ , the neutral  $^4\text{He}$  injection rate  $S_\alpha \times 10^{-16}$  and the auxiliary heating power density  $P_{\text{aux}} \times 10^{-19}$  required for steady-state operation at the same plasma parameters as functions of the energy confinement time of the device.

must then be compensated for externally if the steady state is to persist. Thus, if transport losses increase (smaller confinement times) then the D–T refuelling rate, the auxiliary heating power and the neutral  $^4\text{He}$  injection rate have all to be increased in order to maintain the same steady-state operating conditions. It is observed in figure 1 that when  $\tau_E$  increases the D–T refuelling rate decreases at a slower pace than the auxiliary heating and the neutral  $^4\text{He}$  beam, which vanish simultaneously when the energy confinement time reaches the ignition value, i.e. around 7.65 s. Beyond this value  $S_\alpha$  and  $P_{\text{aux}}$  become negative, a non-feasible situation, since for the plasma parameters required, no steady state can be achieved with values of  $\tau_E$  larger than that at which ignition is obtained. It should be pointed out that if we had chosen not to keep the plasma parameters constant, but some other set of quantities instead, the behaviour of the control variables for different values of  $\tau_E$  would be different.

Since the actual control actions are always constrained between a maximum and a minimum value, in the model described in equations (8)–(11) we shall impose

$$0 \leq \hat{S}_f \leq \hat{S}_f^{\max} \quad 0 \leq \hat{S}_\alpha \leq \hat{S}_\alpha^{\max} \quad 0 \leq \hat{P}_{\text{aux}} \leq \hat{P}_{\text{aux}}^{\max}. \quad (15)$$

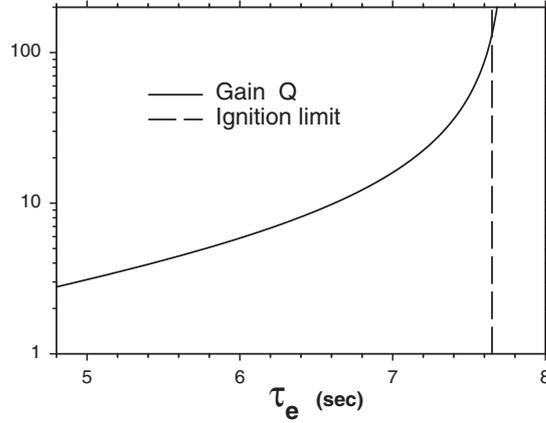
These limits should contain the required values for steady-state operation given in equation (14).

On the other hand, a measure of how near a system is to the true ignition conditions is given by the gain factor  $Q_G$ , defined as the ratio of the rate at which energy is deposited in the plasma by the alpha particles produced by the fusion reactions to the total auxiliary heating power, i.e.

$$\begin{aligned} Q_G &= P_\alpha / (P_{\text{aux}} + P_{\text{ohmic}}) \\ &= Q_\alpha (n_{\text{DT}}/2)^2 \langle \sigma v \rangle / (P_{\text{aux}} + \eta j^2). \end{aligned} \quad (16)$$

In order to be economically viable, a gain factor of  $Q_G > 25$  is desired in a practical thermonuclear reactor [16]. In figure 2 we show the behaviour of  $Q_G$  under steady-state conditions as a function of the energy confinement time for the system described by equations (8)–(14). As it can be concluded from this figure, for the cases we are considering, having  $5.0 \text{ s} \leq \tau_E \leq 6.5 \text{ s}$ , the magnitude of  $Q_G$  is below this value of 25; although, as we will show in section 6 the RBNN controller can handle some cases outside this range.

In order to test for the robustness of the controlled system with respect to the thermalization time of the alpha particles, which is a measure of the average time an alpha particle produced



**Figure 2.** Behaviour of  $Q_G$ , the gain value, under steady-state conditions as a function of the energy confinement time.

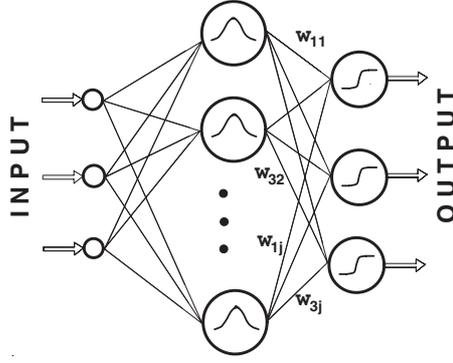
by fusion requires to deposit all its energy in the plasma, an analytical expression for this quantity is needed, which can be used during the simulated transients. In a previous work [1] we approximated the thermalization time as the time it takes for an alpha particle produced by the D–T fusion reactions to reach the critical energy (which was defined such that above it the particle loses its energy mainly to the electrons of the plasma during the slowing-down process) plus the time it takes for the electrons to share this energy with the plasma ions through an energy equipartition process. In that approach we neglected the energy deposited in the plasma by the alpha particles until the latter period of time elapsed. In this work we make a more realistic assumption to this physical phenomena, although still a rough approximation, by assuming that the time dependence of the energy deposition rate during the thermalization of the alpha particles can be divided in two parts: the time interval,  $t_{\text{delay}}$ , needed by the particles slowing down to reach the critical energy [1, 17], during which it will be assumed that no energy is deposited into the plasma,

$$t_{\text{delay}} = 0.1386 \frac{z_3^{3/2}}{z_1} \ln \left( 0.5 + \frac{8.3181 z_3^{-3/2}}{1 - 0.091(1 - z_2) + 0.1047(1 - 1/z_1)} \right) \quad (17)$$

and an interval of time in which the fusion energy is assumed to be deposited uniformly and will be taken to be equal to the characteristic energy equipartition time between the electrons and ions in the plasma [18, 19]. That is

$$t_{\text{spread}} = \frac{0.535}{1 - 0.0386(1 - z_2) + 0.1108(1 - 1/z_1)} \frac{z_3^{3/2}}{z_1} \quad (18)$$

with  $t_{\text{delay}}$  and  $t_{\text{spread}}$  in seconds, and where  $z_1$ ,  $z_2$  and  $z_3$  are the normalized plasma parameters defined in (8). At the nominal operating conditions of this work the critical energy is 0.54 MeV; thus the alpha particles deposit almost 85% of their original energy directly to the plasma electrons during the thermalization time. Under these conditions (17) yields a value of 0.3 s and (18) a value of 0.54 s, yielding an effective thermalization time of approximately 0.84 s. In the above expressions we have included the contribution of the electrons, D–T and helium ash in the thermalization process, but the contribution of the high- $Z$  impurities is neglected.



**Figure 3.** Structure of the RBN used in this work, showing the Gaussian nodes in the hidden layer and the sigmoids in the output units.

### 3. RBNNs and the parallel training strategy

Similarly to the standard multilayered perceptrons, the radial basis networks have a feedforward topological structure and in this work we will be concerned with RBNNs composed of Gaussian nodes in the hidden layer and sigmoidal units in the output (see figure 3). In general, an RBNN is composed of three layers: the input layer ( $l = 1$ ), the hidden layer ( $l = 2$ ) and the output layer ( $l = 3$ ); the activation of the  $j$ th node in the  $l$ th layer will be denoted by  $\mathcal{O}_j^{(l)}$  and the number of nodes in each layer by  $J(l)$ .

Since communication exists exclusively between units in adjacent layers, the output of the RBNN is computed by propagating forward the activation of the nodes in the previous layer of neurons. The nodes in the hidden and output layers are nonlinear functions which map the multidimensional input taken from the previous layer into a one-dimensional output. Similarly to the standard feedforward artificial neural network, the role of the input nodes is to pass unaltered the values received in their inputs to the Gaussian nodes in the hidden layer, whose activation levels are then determined by

$$\mathcal{O}_j^{(2)} = \exp\left(-\sum_{i=1}^{J(1)} (\mathcal{O}_i^{(1)} - C_{ji})^2 / 2\sigma_{ji}^2\right) \quad \text{for } j = 1, \dots, J(2). \quad (19)$$

Here the set  $\{C_{ji}, i = 1, \dots, J(1)\}$  defines the location of the centre of the  $j$ th Gaussian node in the feature or input space, and  $\sigma_{ji}$  is the width of the Gaussian function in the  $i$ th feature direction. The activation of the output nodes is given by

$$\mathcal{O}_j^{(3)} = f(\text{input}_j^{(3)}) \quad \forall j = 1, \dots, J(3) \quad (20)$$

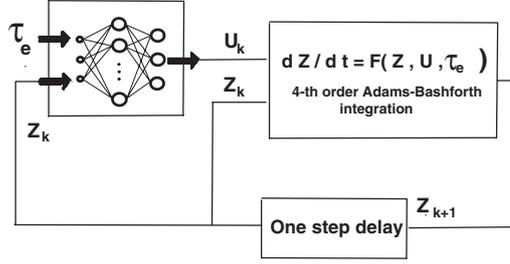
where

$$\text{input}_j^{(3)} = \sum_{i=1}^{J(2)} \omega_{ji} \mathcal{O}_i^{(2)} + \theta_j$$

are the net input to the different output nodes and  $f(\cdot)$  is the sigmoidal function

$$f(\text{input}) = 1/(1 + e^{-\text{input}}).$$

Here  $\omega_{ji}$  is the weight connecting the  $i$ th node in the hidden layer with the  $j$ th node in the output layer and  $\theta_j$  denotes the threshold of the  $j$ th output node. Using sigmoids, the activation level of the output nodes is thus bounded between zero and one. The activation level of each particular hidden node is determined not only by the value of the input but also by the location



**Figure 4.** RBNN–dynamical system feedback configuration used for the stabilization of the burn conditions.

of the centre of the Gaussian function in the feature space as well as by the width of the receptive field determined by the values  $\sigma_{ji}$ .

In general, both the centres and the widths of each Gaussian unit, as well as the weights connecting them to the output layer, can be treated as parameters to be determined during the learning stage. However, in many cases, where the inputs of the set of examples are known to be uniformly distributed within some region of the feature hyperspace, the centres of the Gaussian nodes can be fixed *a priori*, and if, in addition, the input variables are properly normalized then the values of the width  $\sigma_{ji}$  of the Gaussian nodes can all be assigned the same value  $\sigma$ . The resulting network is called by some authors the standard or regular RBNN [20].

The activation of any particular Gaussian node depends on the distance between the centre of the unit and the input pattern, which originated the name of these networks; thus for every input pattern, each one of the hidden units shows an activation level representing a certain degree of matching with the corresponding unit centre.

In order to train the neural network to stabilize the burn conditions of a thermonuclear reactor with energy confinement time varying within certain range of values, the components of the input vector to the RBNN are the normalized values of the electron density, the relative fraction of helium ash, the plasma temperature at a given time step and the corresponding value of the confinement time. The elements of the output vector, on the other hand, are associated with the normalized D–T refuelling rate, neutral helium injection rate and the auxiliary power heating, equation (12), through

$$\hat{S}_f = \frac{S_0}{n_0} k_1 u_1 \quad \hat{S}_\alpha = k_2 (2u_2 - 1)^2 \quad \text{and} \quad \hat{P}_{\text{aux}} = k_3 (2u_3 - 1)^2 \quad (21)$$

where  $u_1$ ,  $u_2$  and  $u_3$  are the activation levels of the sigmoidal output nodes of the neural network, which are bounded between zero and one. In this work we chose  $k_1 = 4.0$ ,  $k_2 = 0.1 \text{ s}^{-1}$  and  $k_3 = 0.2 \text{ s}^{-1}$ , defining the maximum values that the refuelling rate, the neutral helium injection and the auxiliary power heating can take, respectively. Thus, from (12) and (15) we obtain

$$0 \leq S_f \leq 4S_0 \quad 0 \leq S_\alpha \leq 0.1 \times f_0 n_0 \text{ s}^{-1} \quad 0 \leq P_{\text{aux}} \leq 0.2 \times 1.5 n_0 T_0 \text{ s}^{-1}. \quad (22)$$

In order to stabilize the system around a given state, the neural network must provide appropriate values for the control variables, according to the current state of the system. Thus by considering the neural network–dynamical system joint configuration as a single unit, see figure 4, a set of input–output teaching patterns for training can be generated.  $\mathcal{E}_m$  denotes the error between the target state  $z_i$  and the actual final state  $z_{mk_f}$  of the trajectory reached, after  $k_f$  time steps, by the joint neural network–dynamical system configuration given the initial condition  $m$ .

Thus for a particular set of weights specifying the neural network, and a subset of  $M$  initial conditions that the system can take, the total error  $\mathcal{E}$  is defined as the sum of these individual errors after the generation of the set of  $M$  trajectories, i.e.

$$\mathcal{E} = \sum_{m=1}^M \mathcal{E}_m = \frac{1}{2} \sum_{m=1}^M |z_{mk_f} - z_t|^2. \quad (23)$$

A successful training consists of determining a set of weights  $w$  connecting the Gaussian nodes in the hidden layer with the sigmoidal output units, for which the total error  $\mathcal{E}$  is a global minimum. For this purpose a dynamic backpropagation algorithm [21, 22] is used to calculate the gradient of the error  $\mathcal{E}$  in weight space,  $\nabla \mathcal{E}$ ; and with this information we iteratively reduce the above error by updating the weights using a conjugate gradients method [23, 24].

In the appendix we present the details of the training strategy and the dynamic backpropagation used.

#### 4. Training results

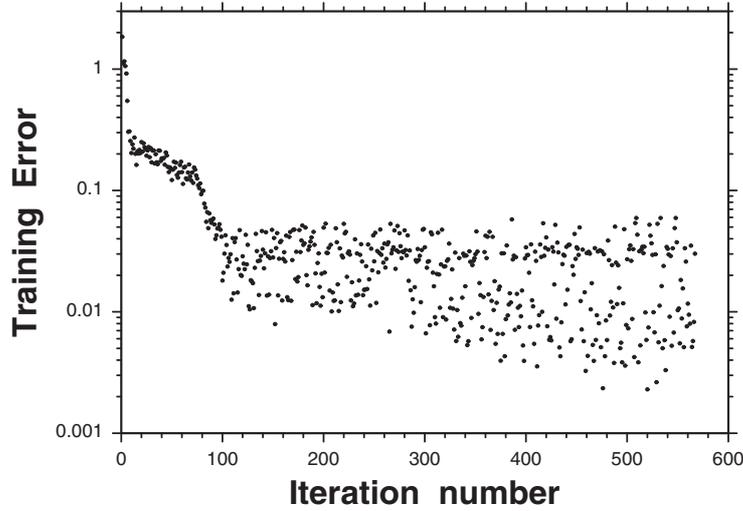
It is our purpose to train a RBNN to provide feedback stabilization for devices with energy confinement times ranging between 5.0 s and 6.5 s. Thus, as shown in figure 4, in addition to the current state of the system we must also provide the value of the energy confinement time as an input to the network. In order to be able to use a regular RBNN it is necessary to properly normalize the energy confinement time; this normalization, together with the region of the phase space that is desired to cover, determine the region of the hyperspace over which the centres of the Gaussian units in the hidden layer must be uniformly distributed. In this work the centres of the radial nodes were located such that their coordinates in the phase space cover uniformly a range of values between 0.8 and 1.2 along each input axis; using four nodes along each axis, the coordinates of the hidden units in the normalized phase space were obtained from all possible combinations of the values 0.8, 0.933, 1.067 and 1.2. Thus, a total of 256 Gaussian units were used in the hidden layer. The width of all the Gaussian units, equation (19), was chosen as twice the squared distance between closest-neighbour centres, i.e.  $\sigma^2 = 0.035\,56$ .

The input parameter associated with the energy confinement time, which we will label  $z_4$ , also needs to be restricted to take values within this range, and the corresponding confinement time will be obtained using the following transformation:

$$1/\tau_E = 0.292\,31 - 0.115\,38z_4 \quad (24)$$

which yields  $\tau_E = 6.5$  for  $z_4 = 1.2$  and  $\tau_E = 5.0$  when  $z_4 = 0.8$ . The reason why this particular transformation was chosen is that it is  $1/\tau_E$  and not  $\tau_E$ , the factor that always appears in the dynamical equations.

In the training algorithm described in the appendix it is required to divide a region of the phase space into a certain number of hypercells. For the purpose of this work the cells were constructed as follows: each of the  $z_1$ ,  $z_2$  and  $z_3$  axes, corresponding to the normalized densities and temperature, were divided into three sections with upper and lower boundaries at 1.08 and 0.9 respectively, while the region of the fourth input to the network,  $z_4$ , associated with the energy confinement time through equation (24) above, was divided into four sections between 0.8 and 1.2. This procedure yields 108 hypercells, from which, in step 3 of the algorithm, the initial states are randomly chosen from within each of these cells. It follows that three initial states with density and temperature perturbations, in the range within 10% below and 8% above the nominal values, were chosen for every one of the four values of the energy confinement time randomly chosen within the range 5.0–6.5 s. Following step 4 of the algorithm, the state of the system was then allowed to evolve over time, according to



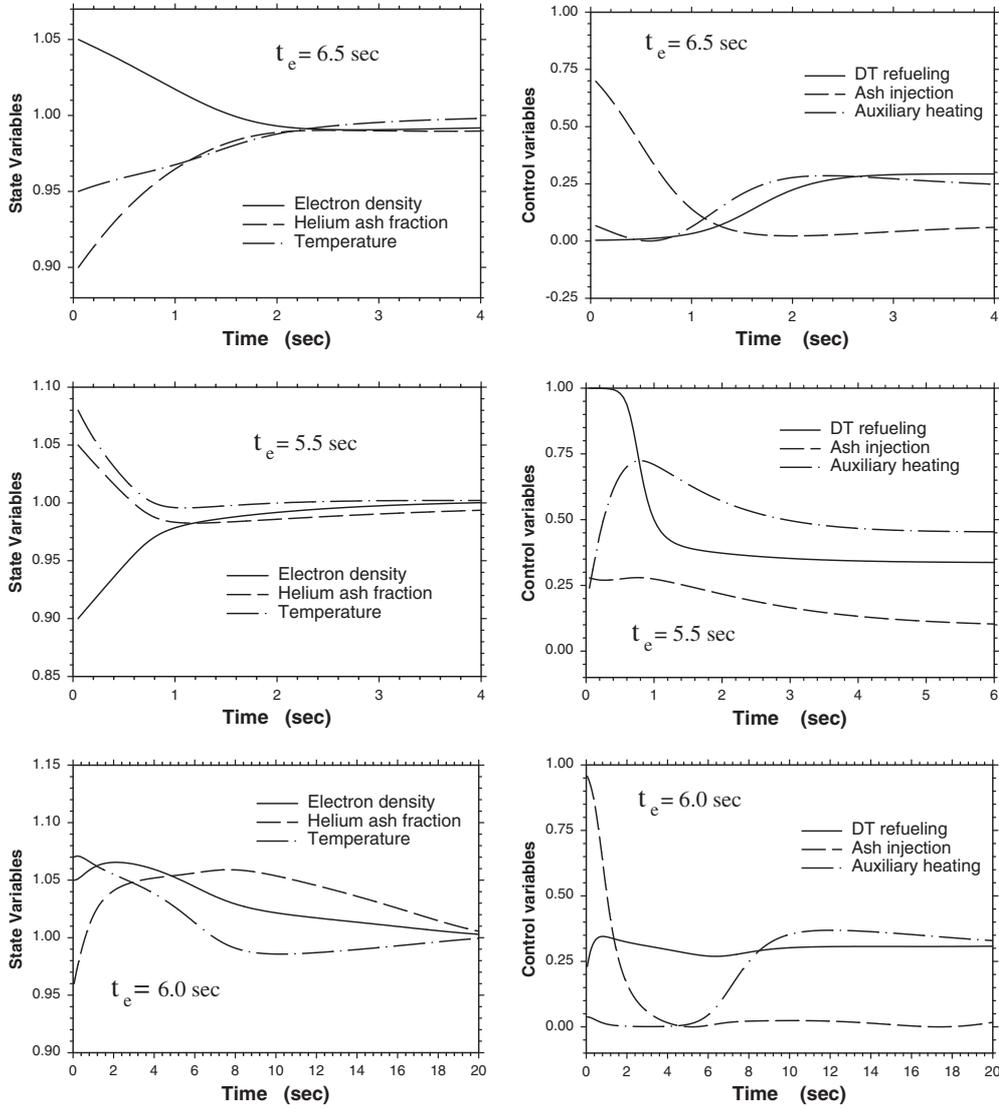
**Figure 5.** Training error as defined in equation (23) as a function of the iteration number, obtained with the training algorithm described in section 4.

the configuration shown in figure 4, using a fourth-order Adams–Bashfort integration scheme [25] with time steps of 0.05 s, while keeping constant the values of  $\tau_E$  during the entire simulated transient. In step 5, during the calculation of the components of  $\nabla_{\omega}\mathcal{E}$  using the dynamic back propagation algorithm in equations (A1)–(A4), it is required to evaluate terms of the type  $\partial z_i(k+1)/\partial u_j(k)$  and  $\partial z_i(k+1)/\partial z_j(k)$ , and we do this in an approximate fashion using a simple Euler discretization:  $z(k+1) = z(k) + f(z(k), u(k))\Delta t$ , of the dynamical equations. The maximum number of time steps allowed in each trajectory varied according to the iteration number of the algorithm: it was set to a maximum of 130 time steps during the first 10 iterations, to 150 steps during the next 40 iterations and to 170 time steps for the next 50; then 200 time steps were allowed up to the 300th iteration; followed by 230 steps up to iteration 500; afterwards a maximum of 250 time steps was imposed. The simulation transients were forced to stop when either the maximum number of time steps allowed was reached or any of the state variables deviated by more than 15% from its nominal operating value. In figure 5 we show the training error  $\mathcal{E}$ , in equation (23), as function of the iteration number of the algorithm.

We illustrate the stabilization capabilities of the resulting RBNN with three transient examples.

*Case a.* In this example the normalized initial electron density, plasma temperature and helium ash fraction are chosen to be 1.05, 0.95 and 0.90, respectively. In figure 6 (top row) we show the behaviour of the state and control variables when the energy confinement time of the thermonuclear system is 6.5 s, and it is kept constant for the entire length of the simulation. It is observed that the RBNN is able to control the perturbation, returning the state of the system close to its nominal operating point within 4 s into the transient.

*Case b.* In this case, at the start of the transient, the electron density, the plasma temperature and helium ash fraction are 10% below, and 8% and 5% above their nominal values, respectively; while the energy confinement time is equal to 5.5 s and remains constant during the entire duration of the transient. In figure 6 (middle row) we illustrate the time behaviour of this transient. It is observed that the network is able to suppresses these perturbations within 4 s into the transient.



**Figure 6.** Behaviour of the normalized state variables (left-hand column) and the normalized control variables  $S_f/(4S_0)$ ,  $S_\alpha/(0.1 \times f_0 n_0)$  and  $P_{aux}/(0.2 \times 1.5 n_0 T_0)$  (right-hand column) as functions of time for different initial conditions in the plasma parameters and different energy confinement times. The top row contains the results for the initial values  $n_e = 1.05 \times n_0$ ,  $f_\alpha = 0.95 \times f_0$  and  $T = 0.9 \times T_0$ , and an energy confinement time of 6.0 s; the middle row shows the cases of an energy confinement time of 5.5 s for the initial conditions  $n_e = 0.9 \times n_0$ ,  $f_\alpha = 1.05 \times f_0$  and  $T = 1.08 \times T_0$ . The bottom row shows a transient with initial conditions of  $n_e = 1.05 \times n_0$ ,  $f_\alpha = 0.96 \times f_0$  and  $T = 1.07 \times T_0$ , and an energy confinement time value of 6.0 s.

*Case c.* Finally, in this example we chose an initial perturbation of 1.05, 0.96 and 1.07 for the normalized electron density, helium ash fraction and plasma temperature, respectively. In figure 6 (bottom row) we show the behaviour of the state and the control variables as time evolves, when the energy confinement time is 6.0 s. It is observed that the RBNN suppresses these perturbations within 20 s into the transient.

In general, large simultaneous perturbations above the nominal values in the electron density and plasma temperature are more difficult to control because of the greater amount of fusion reactions taking place in the system, which makes it necessary for the system to first cool down, reducing the fusion heating power; in addition the larger confinement time of the helium ash contributes significantly to the longer time required to suppress these perturbations.

## 5. Robustness tests

In the previous section we presented some examples of the time behaviour of transients with constant  $\tau_E$ , obtained with the RBNN after the training session. In what follows we show that the training strategy resulted in a network with robust stabilization capabilities, independent of the device's scaling law.

It is expected that in practical fusion power plants the global energy confinement time  $\tau_E$  can be measured or estimated on-line from [3]

$$\tau_E = \frac{3}{2} \int_{V_{\text{core}}} d^3r \frac{nT}{P_{\text{net}} - \dot{W}_{\text{th}}} \quad (25)$$

where  $P_{\text{net}}$  is the total net heating power and  $\dot{W}_{\text{th}}$  is the rate of change of the total thermal energy contained in the plasma. Here in order to illustrate the capability of the network to adapt to different confinement time behaviours, an 'on-line' estimation will be simulated with the instantaneous values of  $\tau_E$  obtained from

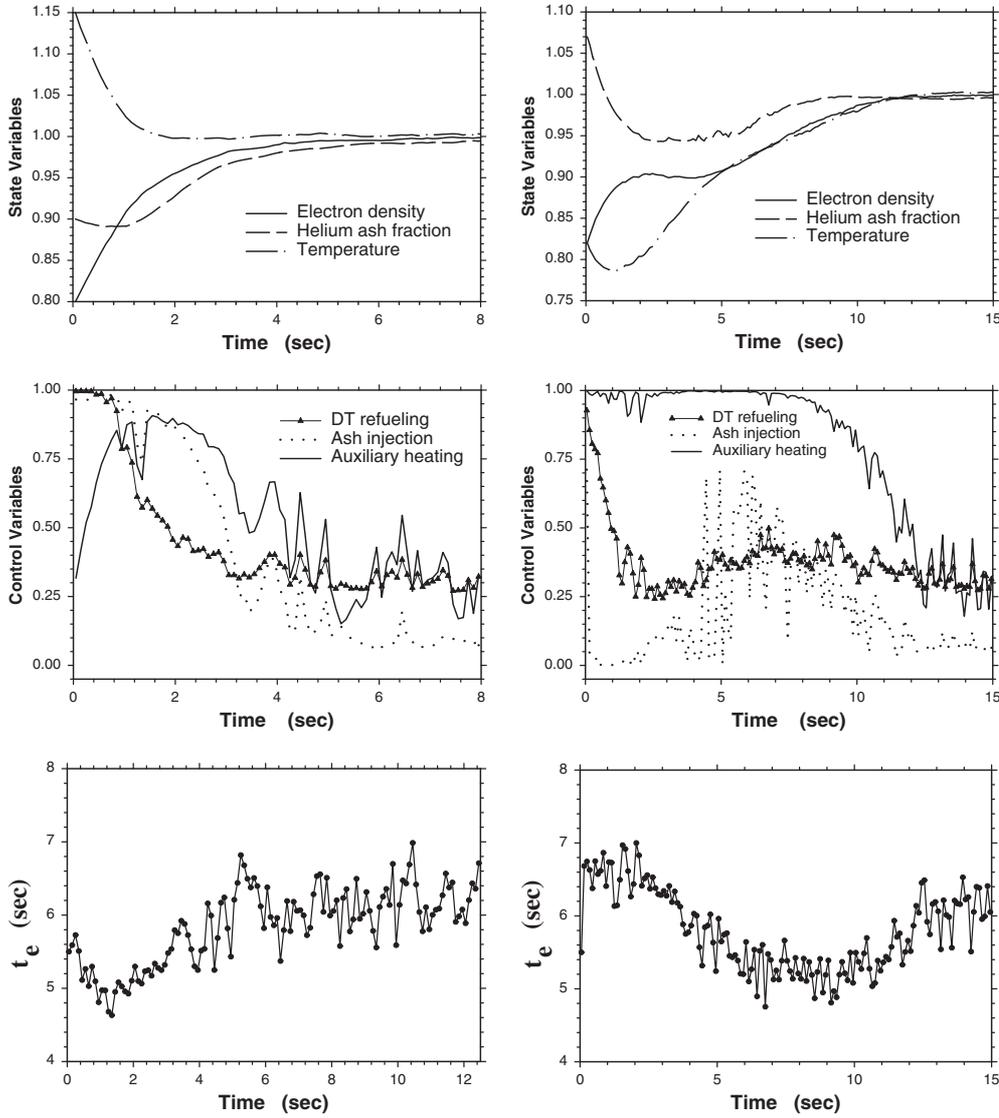
$$\tau_{\text{elmy}} = 0.029 I^{0.90} B^{0.20} P_{\text{net}}^{-0.66} n^{0.40} R^{2.03} \epsilon^{0.19} k^{0.92} M^{0.2} \quad (26)$$

a scaling law associated with an ELMy operating mode [15]; where denoting by  $V_{\text{core}}$  the volume of the plasma, then  $P_{\text{net}} = V_{\text{core}}(P_{\alpha} + P_{\text{ohm}} + P_{\text{aux}} - P_{\text{rad}})$ . Under the desired nominal operating conditions of the system in this work, this expression yields a nominal energy confinement time  $\tau_E \approx 6.17$  s, with a gain value  $Q_G$  of approximately seven as can be deduced from figure 2.

Due to the range of energy confinement times used in the training the RBNN is likely to fail to return the system to normal operating conditions for perturbations where the energy confinement time attains values far above the range used for training. Thus, in the results presented in this section we avoid this unwanted effect by limiting the upper value of the  $\tau_E$  fed into the RBNN to a maximum value of 7 s; nevertheless, the value of  $\tau_E$  obtained from (26) is not modified when used in the dynamical equations.

In order to represent a noisy  $\tau_E$  measurement environment or an uncertain scaling law, we show two simulated transients in which the thermonuclear reactor follows the scaling law in (26), but where the actual value of  $\tau_E$  which is put into the RBNN is obtained at each time step from a Gaussian stochastic process with a mean value given by the instantaneous value of  $\tau_{\text{elmy}}$ , equation (26), and with a standard deviation of  $0.04 \times \tau_{\text{elmy}}$ . Preliminary partial results of this test were reported in [26].

*Case d.* For this transient we choose the following initial conditions  $n_e = 0.8 \times n_0$ ; for the ash fraction  $f_{\alpha} = 0.9 \times f_0$ , corresponding to a helium ash density of 28% below its nominal value, and a plasma temperature  $T = 1.15 \times T_0$ . Figure 7 (left column) shows the time behaviour of the normalized state and control variables for this transient. In figure 7 (bottom figure) the ELMy confinement time  $\tau_E$  as function of time is shown, which has been corrupted with Gaussian noise as explained above. Apart from the Gaussian noise added, the behaviour of  $\tau_E$  can be explained using the scaling law in (26). It has, initially, a small value because the electron density lies below its nominal value and the contribution of both the alpha particle heating and the auxiliary heating driven by the controller yield a large net



**Figure 7.** Two transients of the RBNN–dynamical system configuration shown in figure 4, for a fusion device obeying the ELMy scaling law in equation (26). The energy confinement time fed into the RBNN is corrupted with Gaussian noise as discussed in the text. The behaviour of the normalized state variables (top row), the normalized control variables  $S_f/(4S_0)$ ,  $S_\alpha/(0.1 \times f_0 n_0)$ , and  $P_{aux}/(0.2 \times 1.5n_0 T_0)$  (middle row) and the noisy confinement time fed into the RBNN (bottom row), are shown as functions of time for the following initial conditions: in the left-hand column we have case d  $n_e = 0.8 \times n_0$ ,  $f_\alpha = 0.9 \times f_0$  and  $T = 1.15 \times T_0$ ; in the right-hand column we have the case e discussed in the text, whose initial conditions are  $n_e = 0.82 \times n_0$ ,  $f_\alpha = 1.07 \times f_0$  and  $T = 0.82 \times T_0$ .

heating power. After approximately 4 s the density has increased and the temperature has decreased to almost their nominal values; this, together with the fact that the auxiliary heating has diminished, as observed in figure 7, makes the net heating power into the plasma decrease and the confinement time to eventually increase around its nominal value. We note that in spite

of the noise, the control actions produced by the network are able to return the thermonuclear system to its nominal operating point within 8 s into the transient.

*Case e.* In this case the electron density and the plasma temperature are both 18% below their nominal operating values, while the helium ash fraction is 7% above its nominal value; corresponding, in this case, to an helium ash density  $n_\alpha \approx 12\%$  below its nominal operating value. Figure 7 (right column) shows the resulting time behaviour of the electron density, the temperature and the ash fraction as a consequence of the control actions of the D–T refuelling rate, the neutral  $^4\text{He}$  injection beam and the auxiliary heating. In this figure the confinement time as obtained from (26) corrupted with Gaussian noise is also shown. It is observed that the network suppresses these perturbations within 15 s into the transient. Again the behaviour of  $\tau_E$  shown in this figure for this case is explained as follows. Since, initially, both the electron density and the plasma temperature are well below their nominal values a low net heating power into the plasma is produced in spite of the upper limit value attained by the auxiliary heating driven by the controller; this produces a large value of  $\tau_E$  during the first couple of seconds into the transient. Afterwards the density increases and although the temperature is still low, together with the auxiliary heating, the net heating power increases enough and hence the confinement time decreases. Eventually the plasma temperature increases and the auxiliary heating decreases, the confinement time then recovers to within a region around its nominal value.

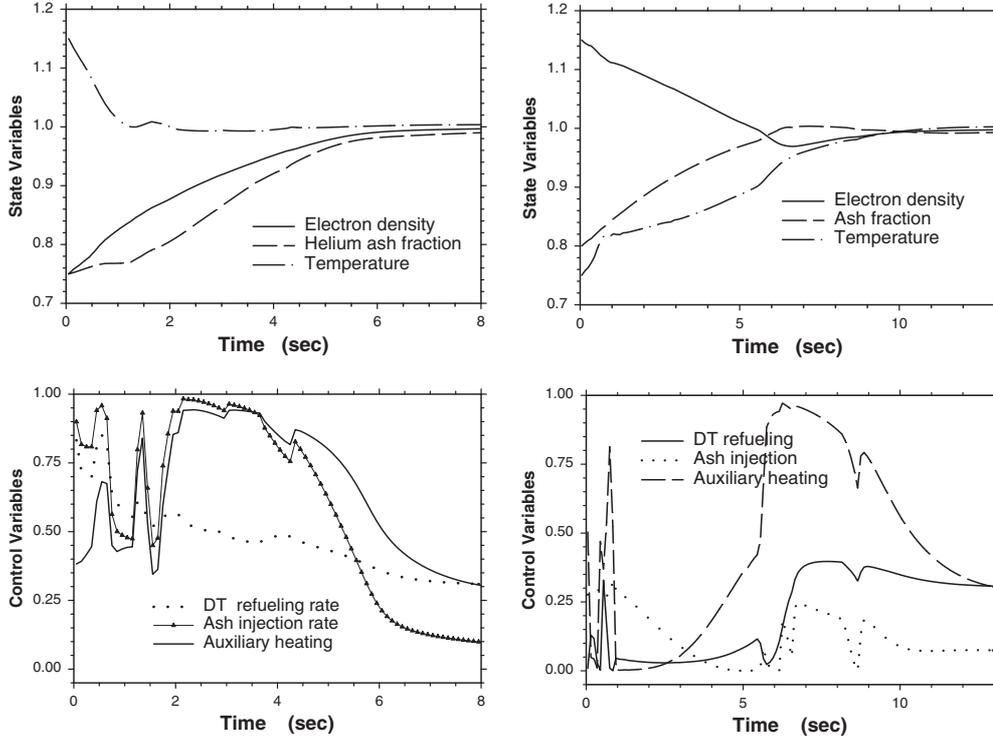
In what follows we show the effect of the thermalization process of the alpha particles produced by the fusion reactions on the behaviour of the joint RBNN–dynamical system configuration. We shall recall that the training of the RBNN, discussed in sections 3 and 4, was performed assuming instantaneous thermalization of the alpha particles. Following the discussion of section 2, we test for robustness with regard to this process by approximating the energy deposition rate in the plasma by an alpha particle produced by a fusion reaction at, for instance, time  $t = 0$ , as

$$\mathcal{P}_\alpha(t) = \frac{Q_\alpha}{t_{\text{spread}}} [u(t - t_{\text{delay}}) - u(t - t_{\text{delay}} - t_{\text{spread}})] \quad (27)$$

where  $u(x)$  is the Heaviside unit step function,  $Q_\alpha$  is the total energy carried by the alpha particle and  $t_{\text{delay}}$  and  $t_{\text{spread}}$  are the delay and spread time intervals defined in equations (17) and (18), which depend on the current plasma parameters.

With this assumption, we show, next, two typical transient simulations using dynamical equations which include the alpha particle thermalization and where the noise has been excluded in order not to obscure this effect; the fusion reactor is assumed again to follow the ELM scaling law in equation (26). It is necessary to point out here that by including the thermalization time delay, the dynamical system acquires a memory, i.e. the behaviour of the system defined by equations (9)–(11) is no longer determined by the current values of the state variables,  $z_1$ ,  $z_2$  and  $z_3$ , alone, but also by their past values which determine the actual energy deposited into the plasma at the current time. Hence, in the simulated transients shown below the energy deposited by the alphas into the plasma will not be taken correctly during the initial stages of the transients, but it is eventually self-adjusted after few thermalization time periods into the transient.

*Case f.* In this transient example the following initial conditions were chosen,  $n_e = 0.75 \times n_0$ ,  $f_\alpha = 0.75 \times f_0$  and  $T = 1.15 \times T_0$ . In figure 8 (left column), the time behaviour of the normalized state and control variables provided by the RBNN are shown, where it is observed that in spite of some initial hesitant behaviour of the control actions, due to the reasons explained above, the network was able to suppress these perturbations within 7 s into the transient.

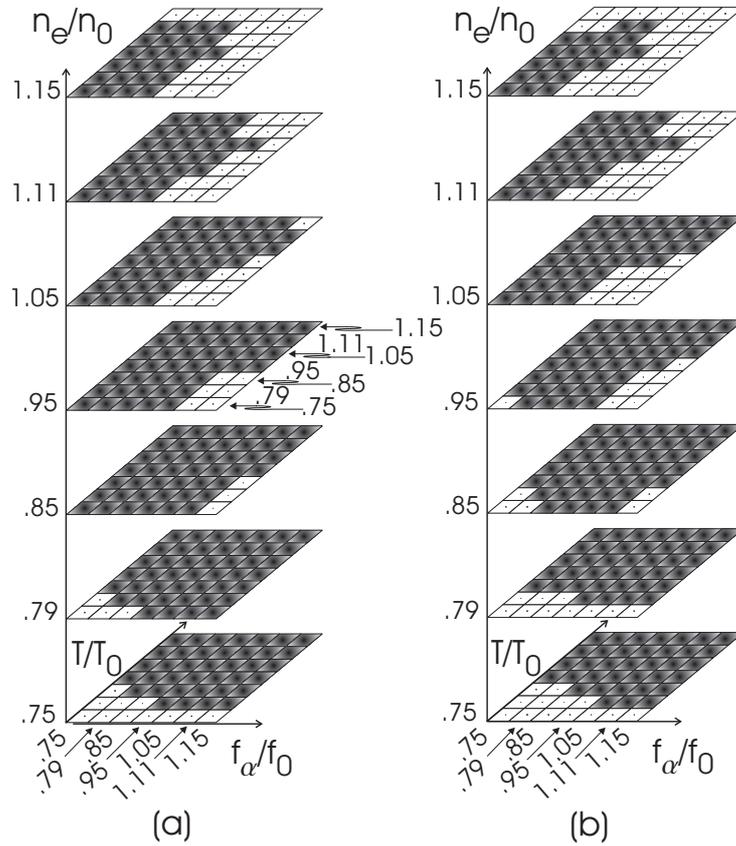


**Figure 8.** Transient examples corresponding to a fusion device obeying the ELMy scaling law in (26) in which the thermalization time of the alpha particles is taken into account. The behaviour of the normalized state variables (top row), and the normalized control variables  $S_f/(4S_0)$ ,  $S_\alpha/(0.1 \times f_0 n_0)$  and  $P_{aux}/(0.2 \times 1.5n_0 T_0)$  (bottom row), are shown as functions of time for the following initial conditions: in the left-hand column we have case  $f$   $n_e = 0.75 \times n_0$ ,  $f_\alpha = 0.75 \times f_0$  and  $T = 1.15 \times T_0$ . The right-hand column contains the case  $g$  discussed in the text, whose initial conditions are  $n_e = 1.15 \times n_0$ ,  $f_\alpha = 0.8 \times f_0$  and  $T = 0.75 \times T_0$ .

*Case g.* In this case the electron density, the ash fraction and the plasma temperature were assigned, respectively, the following initial values  $n_e = 1.15 \times n_0$ ,  $f_\alpha = 0.8 \times f_0$  and  $T = 0.75 \times T_0$ . The state and control values as functions of time resulting from the RBNN control actions are shown in figure 8 (right column). As in the previous case the network was also able to suppress the perturbations within 12 s into the transient. Similarly to case  $f$ , the control actions during the earliest stage of the transient are a little erratic due to an initially incorrect account in the energy deposition of the alpha particles during the simulated transient as explained above, which is corrected however, after a short period of time.

It should be mentioned here that in some simulations high-frequency oscillations in the energy confinement time, which die out after a few seconds, are observed during the evolution of the transients; these fluctuations are also reflected, although to a smaller scale, in the plasma parameters. Similar oscillations, which are produced by periodic fluctuations in the net energy deposited into the plasma, have been observed previously [1].

In figure 9 we show schematically the region of stability in the normalized phase space  $(n_e/n_0, f_\alpha/f_0, T/T_0)$  of the RBNN–fusion device joint configuration (figure 4) with a scaling law given in equation (26). Each of the two diagrams in this figure was generated using 343 simulated transients. The shaded squares indicate the initial deviations in the plasma



**Figure 9.** Schematic diagrams showing the regions where the RBNN–fusion reactor is stable (shaded squares) and unstable (white squares) comprehending perturbations between  $-25\%$  and  $+15\%$ , around the nominal values of the plasma parameters. (a) shows the results when the alpha particles’ thermalization times are taken into account and the cases where Gaussian noise is added to  $\tau_E$ , are shown in (b). In both cases the reactor device is assumed to follow the ELMy scaling law in (26).

parameters for which the RBNN controller is able to drive the fusion system back to its nominal operating point, while the white squares indicate those for which it failed to suppress the perturbation. Figure 9(a) corresponds to the cases where the alpha particle’s thermalization time was included in the simulated transients. On the other hand, figure 9(b) is associated with the cases where the current value of  $\tau_E$  fed into the RBNN controller is corrupted with Gaussian noise, as in the cases d and e discussed above. As we can see, the stability regions in both cases are larger than the original region used for training the RBNN. It is observed that the system fails to suppress the perturbations in the plasma parameters in those regions of phase space for which  $\tau_E$  is significantly larger than the values used for training. In general we can say that the time required to suppress a perturbation is larger when, at the onset of the transient, the electron density and the plasma temperature are both above their values at the nominal operating point. In contrast, the times required are shorter when their initial values are both below this point, although for large departures the RBNN also fails to suppress the perturbations.

## 6. Summary and conclusions

In this work it is shown that a RBNN composed of Gaussian activation units in the hidden layer and sigmoidal nodes in the output layer can be used successfully to stabilize a thermonuclear reactor under subignited burn conditions for a wide range of energy confinement times. A zero-dimensional model was used to represent the evolution of the electron density, the helium ash and the plasma temperature. The radial basis network was trained to stabilize the dynamical system for a range of energy confinement times; thus each transient evolved over time keeping constant the energy confinement time during the entire duration of the transient. The control actions included the concurrent modulation of the D–T refuelling rate, a neutral  $^4\text{He}$  beam and an auxiliary heating power, which were constrained to lie within allowable levels.

After training, the resulting network showed that it was able to stabilize the system at fixed plasma parameters, independently of the scaling law of the device, suppressing plasma density and temperature excursions significantly far from their nominal values.

Numerical examples were used to test for robustness of the resulting RBNN under uncertain or noisy environments, showing its capability to adapt to new operating conditions by controlling perturbations in the plasma parameters when the thermonuclear system follows a given ELM scaling law, feeding the instantaneous value of the energy confinement time corrupted with Gaussian noise into the RBNN. In addition, using analytical expressions to estimate the thermalization time of the alpha particles produced by the fusion reactions, the network proved also to be robust with respect to this process. In both cases the region of stability was roughly estimated using simulated transients.

Further research is being undertaken to develop a neural network with load-following capabilities, as well as to allow for the ions and electrons to have different temperatures.

## Acknowledgments

The authors gratefully acknowledge partial financial support from the CONACyT 34911-A, DGAPA-UNAM IN117498, CONACyT 27974-E and CRAY-UNAM SC-010399 projects.

## Appendix

The neural network training was performed using a parallel code [11] developed using MPI, a recently created portable message passing environment standard for parallel programming and which was modified to work with radial basis networks. A parallel computation in MPI lies within the distributed memory SPMD framework (Single Program Multiple Data) and consist of a set of processors concurrently running copies of a single program written in a standard language, such as F77 or C, enhanced by library calls to MPI functions for sending and receiving messages (data) between processes as well as for task synchronization [27, 28].

The parallelization strategy used the fact that the gradient of the total error,  $\nabla\mathcal{E}$ , is the sum of independent gradients of the individual errors  $\mathcal{E}_m$  of each of the trajectories,

$$\nabla\mathcal{E} = \sum_{m=1}^M \nabla\mathcal{E}_m. \quad (\text{A1})$$

Thus, assuming that we have a set of  $P$  processors, labelled  $p = 0, 1, \dots, P - 1$ , which are available to contribute to the calculation of  $\nabla\mathcal{E}$ , the following parallel algorithm can be devised.

- Step 1. Task assigned to processor  $p = 0$ . Randomly select the initial values of the set of weights that specify the neural network, and use MPI library calls to broadcast these values to all the other  $P - 1$  processors.
- Step 2. Task assigned to processor  $p = 0$ . Divide a properly chosen region of the phase space into a number  $M$  of hypercells.
- Step 3. Task assigned to processor  $p = 0$ . Select  $M$  initial states by randomly sampling each one of the cells in the extended phase space. Use MPI library calls to broadcast these states to all the other  $P - 1$  processors.
- Step 4. Assign to each processor a subset of the  $M$  initial states following a load balance strategy. Starting from each one of these initial states we allow the RBNN-system configuration to evolve over time until either the state of the systems moves out of a predetermined region or a pre-fixed maximum number of time steps is reached.
- Step 5. Using dynamic backpropagation each of the  $P$  processors calculate and store the gradient of the individual error  $\mathcal{E}_m$  associated with the subset of the  $M$  trajectories that were assigned to it.
- Step 6. A test for convergence in each trajectory of their corresponding subsets is performed by each processor and the results are sent, using MPI calls, to processor  $p = 0$ , which determines whether or not global convergence has been achieved. If the global convergence criteria is satisfied one should stop; otherwise proceed to the next step.
- Step 7. Using the MPI global sum operation, the partial gradients of the errors,  $\nabla \mathcal{E}_m$ , are shared by all  $P$  processors; each processor then proceeds to determine, by adding these components, the gradient of the total error  $\nabla \mathcal{E} = \sum_{m=1}^M \nabla \mathcal{E}_m$ .
- Step 8. All processors individually use  $\nabla \mathcal{E}$  to determine the new conjugate gradient's direction, and cooperate in a way similar to step 4 in the search for the minimum along this direction.
- Step 9. Updating of the weights is performed in each processor.
- Step 10. Repeat steps 3–9 until the training of the neural network is successfully completed, i.e. when the entire set of  $M$  trajectories, each of which starts from a different cell, reaches the target  $z_t$  within the error range  $\epsilon$ .

Summarizing, the above algorithm involves three phases. In the first phase the joint neural network–dynamical system configuration evolves over time from an initial state  $z(1)$ , until a final state  $z(N)$ ; this final state is then compared with its target value, resulting in an error signal for each of the state variables of the system. In the second phase these error signals are propagated recursively backward in time from the final time step  $N$  to the initial time to calculate the components of  $\nabla \mathcal{E}$ . In the last phase, the adjustments needed in the weights composing the neural network are determined with the method of conjugate gradients. The use of conjugate gradients has also the advantage of preventing the occurrence of the phenomenon of premature saturation in the sigmoidal activation units [29].

The components of the gradient of the error  $\mathcal{E}_m$  with respect to the weights of the neural network, associated with the trajectory generated from the  $m$ th cell, i.e.  $\partial^\dagger \mathcal{E}_m / \partial \omega_{pq}$ , are obtained, using the concept of ordered partial derivatives [1, 21], from the following expression:

$$\frac{\partial^\dagger \mathcal{E}_m}{\partial \omega_{pq}} = \sum_{n=1}^{N-1} \sum_{j=1}^{J(3)} \frac{\partial^\dagger \mathcal{E}_m}{\partial \mathcal{O}_j^{(3)}(N-n)} \frac{\partial \mathcal{O}_j^{(3)}(N-n)}{\partial \omega_{pq}} \quad (\text{A2})$$

the second factor on the right-hand side of the above equation is obtained for a radial basis network as

$$\frac{\partial \mathcal{O}_j^{(3)}}{\partial \omega_{pq}} = \begin{cases} 0 & \text{if } p \neq j \\ \mathcal{O}_j^{(3)}(1 - \mathcal{O}_j^{(3)})\mathcal{O}_q^{(2)} & \text{for } p = j \end{cases} \quad (\text{A3})$$

where  $\mathcal{O}_q^{(2)}$  is the activation level of the  $q$ th Gaussian node in the hidden layer and  $\mathcal{O}_j^{(3)}$  is the activation of the  $j$ th sigmoidal node in the output layer; the first factor is obtained using the following recursive set of equations:

$$\frac{\partial^\dagger \mathcal{E}_m}{\partial \mathcal{O}_j^{(3)}(N-n)} = \sum_{i=1}^{J(1)} \frac{\partial^\dagger \mathcal{E}_m}{\partial z_i(N-n+1)} \frac{\partial z_i(N-n+1)}{\partial \mathcal{O}_j^{(3)}(N-n)} \quad \text{for } n = 1, \dots, N-1 \quad (\text{A4})$$

with

$$\begin{aligned} \frac{\partial^\dagger \mathcal{E}_m}{\partial z_i(N-n+1)} &= \sum_{j=1}^{J(3)} \frac{\partial^\dagger \mathcal{E}_m}{\partial \mathcal{O}_j^{(3)}(N-n+1)} \frac{\partial \mathcal{O}_j^{(3)}(N-n+1)}{\partial z_i(N-n+1)} \\ &+ \sum_{k=1}^{J(1)} \frac{\partial^\dagger \mathcal{E}_m}{\partial z_k(N-n+2)} \frac{\partial z_k(N-n+2)}{\partial z_i(N-n+1)} \quad \text{for } n = 2, 3, \dots, N-1 \end{aligned} \quad (\text{A5})$$

with the following initialization:

$$\frac{\partial^\dagger \mathcal{E}_m}{\partial z_i(N)} = \frac{\partial \mathcal{E}_m}{\partial z_i(N)}. \quad (\text{A6})$$

In the above equations the symbol  $\dagger$  is used to denote an ordered partial derivative;  $\mathcal{O}_j^{(3)}(N-n)$ , the  $j$ th output of the neural network, is the value of the corresponding control variable at time step  $N-n$ , and  $z_i(N-n+1)$  corresponds to the value of the  $i$ th state variable at time step  $N-n+1$ .

## References

- [1] Vitela J E and Martinell J J 1998 Stabilization of burn conditions in a thermonuclear reactor using artificial neural networks *Plasma Phys. Control. Fusion* **40** 295–318
- [2] Poggio T and Girosi F 1990 Networks for approximation and learning *Proc. IEEE* **78** 1481
- [3] Mitarai O and Muraoka K 1999 A proposed set of diagnostics for core ignition burn control in a tokamak reactor *Nucl. Fusion* **39** 725  
Mitarai O and Muraoka K 1999 Analyses of diagnostic failure effect and fail safe ignited operation in a tokamak fusion reactor *Fusion Technol.* **36** 194
- [4] Hui W, Bamieh B A and Miley G H 1994 Robust burn control of a fusion reactor by modulation of the refueling rate *Fusion Technol.* **25** 318  
Hui W, Bamieh B A and Miley G H 1994 Robust burn control of a fusion reactor with modulation of refueling rate *Fusion Technol.* **26** 1151
- [5] Mitarai O and Muraoka K 1998 Ignition analyses with ITER89P and ITER93HP scalings for burn control and diagnostics in ITER-ID *Plasma Phys. Control. Fusion* **40** 1349–72
- [6] Taylor R J, Fried B D and Morales G J 1990 Burn threshold for fusion plasmas with helium accumulation *Internal Report* UCLA PPG 1290, Institute of Plasma and Fusion Research, Universidad de California, Los Angeles, CA
- [7] Vemuri V R (ed) 1992 *Artificial Neural Networks: Concepts and Control Applications* (Los Alamitos, CA: IEEE Computer Society Press)
- [8] Levin A U and Narendra K S 1993 Control of nonlinear dynamical systems using neural networks: controllability and stabilization *IEEE Trans. Neural Networks* **4**
- [9] Bianchini M, Frasconi P and Gori M 1995 Learning without local minima in radial basis function networks *IEEE Trans. Neural Networks* **6** 749
- [10] ITER Engineering Design Activity Group 1998 webpage <http://www.iteru.de>
- [11] Vitela J E, Hanebutte U R and Gordillo J L Performance analysis of a parallel neural network training code for control of dynamical systems *Int. J. Comput. Res.* at press
- [12] Glasstone S and Lovberg R H 1975 *Controlled Thermonuclear Reactions* (New York: Krieger)
- [13] Hirshman S P and Sigmar D J 1981 Neoclassical transport of impurities in tokamak plasmas *Nucl. Fusion* **21** 1079
- [14] Hively L M 1977 Convenient computational forms for Maxwellian reactivities *Nucl. Fusion* **17** 873

- [15] ITER Confinement Database and Modelling Working Group 1997 Energy confinement scaling and the extrapolation to ITER *Plasma Phys. Control. Fusion* **39** B115–27
- [16] Cohn D R 1994 High gain requirements and high field tokamak experiments *J. Fusion Energy* **13** 271
- [17] Vitela J E and Coronado M 1990 Nonequilibrium fluctuations of a charged Brownian particle in a plasma *Phys. Rev. A* **42** 2354
- [18] Trubnikov B A 1965 *Review of Plasma Physics* vol 1, ed M A Leontovich (New York: Consultants Bureau)
- [19] Spitzer L Jr 1962 *Physics of Fully Ionized Gases* 2nd edn (New York: Wiley)
- [20] Hunt K J and Sbarbaro D 1991 Neural networks for nonlinear internal model control *IEEE Proc. D* **138** 431
- [21] Werbos P 1990 Backpropagation through time: what it does and how to do it *Proc. IEEE* **78** 1550
- [22] Piché S W 1994 Steepest descent algorithms for neural network controllers and filters *IEEE Trans. Neural Networks* **5** 198
- [23] Fletcher R and Reeves C M 1964 Function minimization by conjugate gradients *Comput. J.* **7** 149
- [24] Reifman J and Vitela J E 1994 Accelerating learning of neural networks with conjugate gradients for nuclear power plant applications *Nucl. Technol.* **106** 225
- [25] Golub G and Ortega J M 1993 *Scientific computing: an introduction with parallel computing* (San Diego, CA: Academic)
- [26] Vitela J E and Martinell J J 2000 Robust stabilization of burn conditions in subignited fusion reactors using artificial neural networks *Proc. 10th Inter. Toki Conf. (Toki City, Japan, January 18–21, 2000)* *J. Plasmas Fusion Res.* to be published
- [27] Foster I 1994 *Designing and Building Parallel Programs* (Reading, MA: Addison-Wesley)
- [28] The MPI Committee 1994 The message passing interface (MPI) standard, webpage <http://www.mcs.anl.gov/mpi/index.html>
- [29] Vitela J E and Reifman J 1997 Premature saturation in backpropagation networks: mechanism and necessary conditions *Neural Networks* **16** 721