USE OF A NEURAL NETWORK FOR THE PREDICTION OF DISRUPTIONS ON ASDEX UPGRADE

Ch. Tichmann, G. Pautasso, J.C. Fuchs, K. Lackner, V. Mertens, F.C. Morabito [a], W. Schneider, ASDEX Upgrade Team, Max-Planck-Institut für Plasmaphysik, EURATOM Association, [a] DIMET, Faculty of Engineering, University of Reggio Calabria, Italy

1. Introduction

Most of the plasma disruptions in ASDEX Upgrade happen in a plasma parameter regime (poor L-mode confinement) which is far away from the desired operational space (H-mode, high beta). In addition, disruptions are usually announced by well identified precursors (detachment, MARFE, growth and locking of resistive tearing modes) which can be detected by the available diagnostics. In this work we make use of the experience gathered in several years of operation and disruption studies to train a neural network to recognize a forthcoming disruption.

2. Artificial Neural Network

An Artificial Neural Network implements a non-linear function mapping one multidimensional space, $\{\vec{x}\}$, into another one, $\{\vec{z}\}$; for an overview on the subject see, for example, Ref. [1]. This function has a predefined structure but contains several parameters which are going to be determined during the training. The training consists of the evaluation of the parameters which minimize the difference between the target output, $\vec{t}$, and the network output, $\vec{z}$.

Among several possible structures of the network, we use a, so called, feed-forward multilayer (two layers, in this work) perceptron. This kind of network is known to approximate arbitrarily well any continuous multi-dimensional mapping [1]. The hth-component of the vector output ($h = 1, nz$), can be written as :

$$z_h = \mathcal{F}(\sum_{i=1}^{ny} WY_{hi}\, y_i) \qquad\qquad y_i = \mathcal{F}(\sum_{j=1}^{nx} WX_{ij}\, x_j) \tag{1}$$

where y_i is the ith-component of the output of the first layer; nx, ny and nz are the dimension of the input vector, the number of the hidden neurons and the dimension of the network output respectively. $\mathcal{F}(a)$ is a non-linear function; in this work we chose it to be the sigmoidal function: $(1+exp(-a))^{-1}$. In each layer, the input variable to the specific layer is transformed first linearly, by means of a matrix (WX and WY respectively for the first and second layer) and then by a non-linear function.

The values of the $(nx * ny + ny * nz)$ unknown elements of the matrixes WX and WY are found by minimizing an error function of the type:

$$E = 0.5 * \sum_{k=1}^{N} [\vec{z}(\vec{x}_{(k)}, WX, WY) - \vec{t}_{(k)}]^2 \tag{2}$$

where the sum is extended to the whole training set. A slow but reliable method to minimize the above equation is known as back-propagation and consists of evaluating the derivatives of E with respect to the elements of the WX and WY matrixes and correct the unknown parameters using gradient descendent in the following way:

$$WX_{ij}^{(n+1)} - WX_{ij}^{(n)} = -\delta \frac{\partial E}{\partial WX_{ij}} \tag{3}$$

where δ is an appropriate learning rate parameter and (n) is the iteration number.

3. Input Data and Output

For the training the network needs a large database of input vectors and associated outputs. In this work the input consists of the time histories of several plasma parameters describing the onset, the presence and the evolution of MARFE, tearing modes and plasma regime preceding the disruption. The output of the network was chosen to be the time interval up to the disruption; this makes the output a variable easy to interpret and a flexible trigger, which can be used to avoid or mitigate a disruption.

The choice of input variables is the result of a compromise between the physics and the availability of the data in real-time, since the network will be used for the real-time control of the discharge. A preliminary set of input variables was successively reduced during the training by eliminating the variables which did not significantly contribute to the output. The magnitude of the derivative of the output with respect to the input variables summed over the whole training set, i.e. $\sum_{k=1}^{N} \partial z_{(k)}/\partial x_i$ [2], is a statistical measure of the sensitivity of the output to a given input parameter (our output has dimension 1).

The plasma parameters used to prepare the input of the network presented in this work are: the magnetic field, the safety factor, the plasma thermal energy, the energy confinement time, the plasma inductivity, the plasma density, the input power, a few of the divertor bolometer channels, a bolometer channel trough the plasma center, the radiated energy, two H_α signals from the divertor, the locked-mode signal and their time derivatives. Several of these parameter were properly normalized (for example, the density was divided by the Greenwald limit).

The database used to train and validate the network consists of shots in the range 10000 - 10800 which ended with the disruption of a lower-single-null plasma in flat-top. Some shots were neglected owing to incomplete or incorrect measurements. Disruptions caused by injection of killer pellets or following a VDE were also excluded. Only the pre-disruption phase of these shots was selected to be part of the input for the network. The pre-disruption phase was defined as the L-mode phase following the H-mode phase before the disruption or as the phase starting just before a MARFE and ending with the disruption, for a plasma which has been for longer than 1 s in L-mode. The diagnostic data were further normalized to the [0,1] interval, time averaged over 25 ms and feed to the network every 2.5 ms. The database of 16398 data points from 106 shots was then used for training.

4. Training

The training of the network was carried out on the parallel computer Cray T3E and with programs written by the first author in Fortran 90. The training consisted in minimizing the function:

$$E = \sqrt{N^{-1} \sum_{k=1}^{N} [1 - \Delta t_{NN(k)}/\Delta t_{(k)}]^2} \qquad (4)$$

where Δt and Δt_{NN} are the measured and the predicted time to disruption respectively.

The input data were distributed to 16 processors and Eq.(4) was minimized iteratively (i.e., N=16) by changing the weights according to Eq. (3). The database of disruptive shots was divided in a training set (71 shots and 11991 data points) and a validation set (35 shots and 4407 points). Several networks with different number of hidden neurons and different number of input data were trained and tested. The training lasted typically 1000-4000 cycles (in one cycle the weights have been updated according to Eq. (3) on the basis of the whole training set). The training was stopped after checking by visual inspection that the function given by Eq. (4) and calculated for the validation set had reached a minimum or a stationary value. The network with the smallest RMS errors on the validation data (60 %; see Fig. 1) and on the whole database of disruptive shots (55 %) was reached with 7 hidden neurons and 26 input parameter (11 of which are time derivatives); this optimized network was then tested as disruption predictor.

5. Network Predictions

The performance of the network was tested off-line in the flat-top phase of 476 shots without disruption (in the shot range 10000-10800) and 53 disruptive shots (in the range 10000-11300), which were not included in the training.

We have to say at this point, that the performance of a neural network outside of its training space is not reliable. Since our network was trained with only pre-disruption plasmas we need to pre-process the input data for the feed-forward prediction of the time-to-disruption in order to find out if it belongs to the training space. This is done presently in an elementary way. The ranges of the variables of the training set were divided into a number of intervals and the training set was therefore divided into a number of cells; each of the training patterns corresponds to a point in one of the cells; an input is classified as *known* and feed to the network if it falls into an occupied cell; an input classified as *unknown* is going to be ignored: it probably represents a plasma far away from disruptions or a plasma close to a new kind of disruption.

The disruption alarm was defined as $\Delta t_{NN} < -50$ ms for 7.5 ms. The prediction of the disruption was defined to be correct if the disruption alarm was activated in the time interval $[t_{disr} - 400ms, t_{disr} - 5ms]$; a disruption alarm in shots without a disruption or more than 400 ms before a disruption was considered a false alarm; a disruption was not recognized when the disruption alarm was activated too late or not at all.

The results of the analysis were as follows:

- 83 % of the disruptions were recognized; the network did not recognize the disruption in 9 of the 53 disruptive shots (17 %);

- the network produced at least one false alarm in 9 of the 476 shots (2 %);

The statistical distribution of the real time interval to disruption at the time of the disruption alarm is shown in Fig. 2. An example of the network prediction is shown in Fig. 3.

The false alarms were caused by pre-disruption phases from which the plasma recovered thanks to an action taken by the feed-back system (increasing the heating power or closing a gas valve). The disruptions, which were not recognized, had all been poorly represented in the training database; they had the following causes:

- 6 disruptions were caused by strong impurity puffing in non-disruptive shots and in experiments aimed to reduce the disruption loads (mitigated disruptions);

- 1 disruption followed impurity accumulation;

- 2 disruptions followed mode locking in low density plasmas.

6. Conclusions

The present work shows that a disruption recognition system based on a neural network is feasible and relatively reliable (more than 80 % predicted events and a few % false predictions) and it encourages further development of such systems and their application. The feed-forward calculation of the output of a trained network is in fact simple and fast, making it suitable for real-time applications.

References

[1] C.M. Bishop, *Neural Networks for Pattern Recognition,* Clarendon Press Oxford

[2] D. Wroblewki et al., Nuclear Fusion **37** (1997) 725

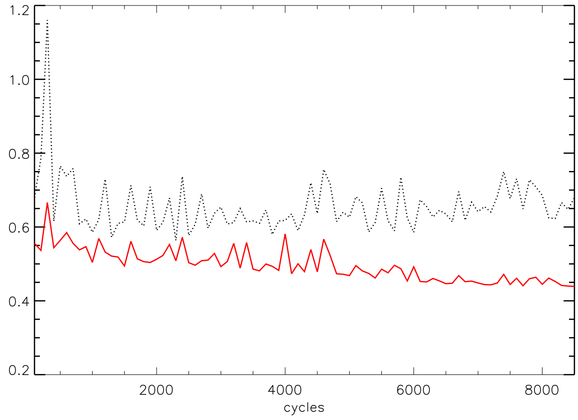

Figure 1. - RMS error of the network calculated using the training set (continuous line) and validation set (dotted line) as function of the training cycle.

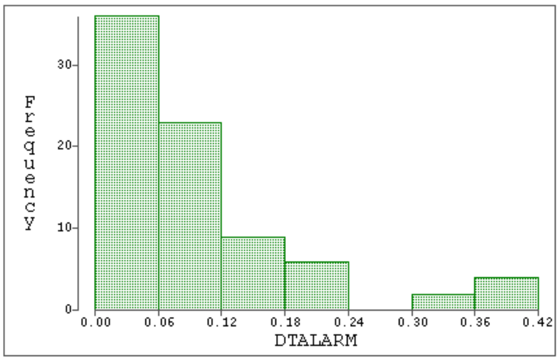

Figure 2. - Statistical distribution of the real time interval to disruption at the time of the disruption alarm.

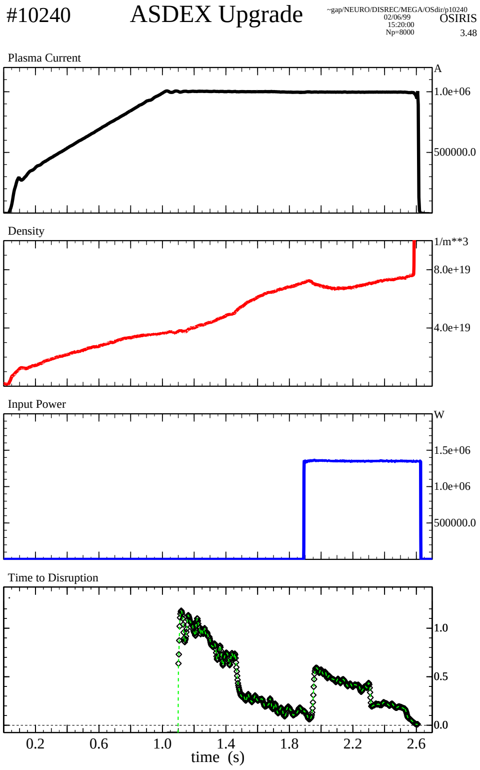

Figure 3. - Prediction of the network for a plasma which disrupted because of high electron and impurity densities.