ECR identification by neural networks

An alternative ECR recognition procedure, based on neural networks (NN) algorithms, is under development by Vlad Constantinescu, who contributed to ECSTRA as young scientist. Vlad completed a Master program at Politehnica University of Bucharest, Faculty of Automatic Control and Computers, with a thesis based on this work (*Constantinescu*, *2009*), and later on presented preliminary results at the 9th International School for Space Simulations, at the 19th Cluster workshop, and at a Summer School on Neural Networks in Classification, Regression and Data Mining (*Constantinescu and Marghitu*, *2009*, *2010a*, *2010b*). The collaboration with Dr. Simon Wing, from Johns Hopkins University / Applied Physics Laboratory (JHU/APL), initiated after the 19th Cluster workshop, was essential for the progress of the NN work. A visit to JHU/APL in the spring of 2011 contributed to speed up this progress. In the summer of 2011 Vlad was included in the Young Scientist program of ISSI and received ISSI support to attend the second meeting of the POLARIS project.

Neural networks provide a powerful pattern recognition tool, that can be used to search large amounts of data for certain types of 'events' — in our case for ECR events — once the NN is properly trained. Using NN with Cluster data raised, at the beginning, two problems: first, the initial training set, consisting of the 43 events in the manual database, appeared to be too limited, and second, the data used for training could not be explored later, in a consistent manner, for the presence of (additional) ECRs. In order to overcome these problems we tried to use synthetic data for training and testing the NN. Initial results obtained in the detection of real and synthetic ECR events are presented in Figure 1, where the synthetic ECR events have random amplitude, duration, and sign.

Initially, the ECR events were identified by processing data intervals of 100 elements in one step, with a NN configuration holding 100 input and 100 output neurons. This structure proved to be much too complex, with unsatisfactory generalization capabilities and requiring very long training times. Following the suggestion of Dr. Simon Wing, the NN structure was changed by implementing a sliding window algorithm. The input layer was reduced to less than 30 neurons (together with the data intervals) and the output layer to just 1 neuron, with the effect of a significant decrease in the NN complexity and the training time — from a few days to a few tens of minutes. Another important change was in the selection of the training set. Instead of arbitrary time intervals, with the CLR or CGR events typically much shorter than the surrounding non-events, the time intervals in the new training set consisted of events and non-events of comparable duration (like in Figure 2, left). In addition, the events were evenly distributed in amplitude, duration, and sign, in order to avoid a biased training of the NN. As a result of these changes, the detection accuracy was considerably improved (Figure 2, right).

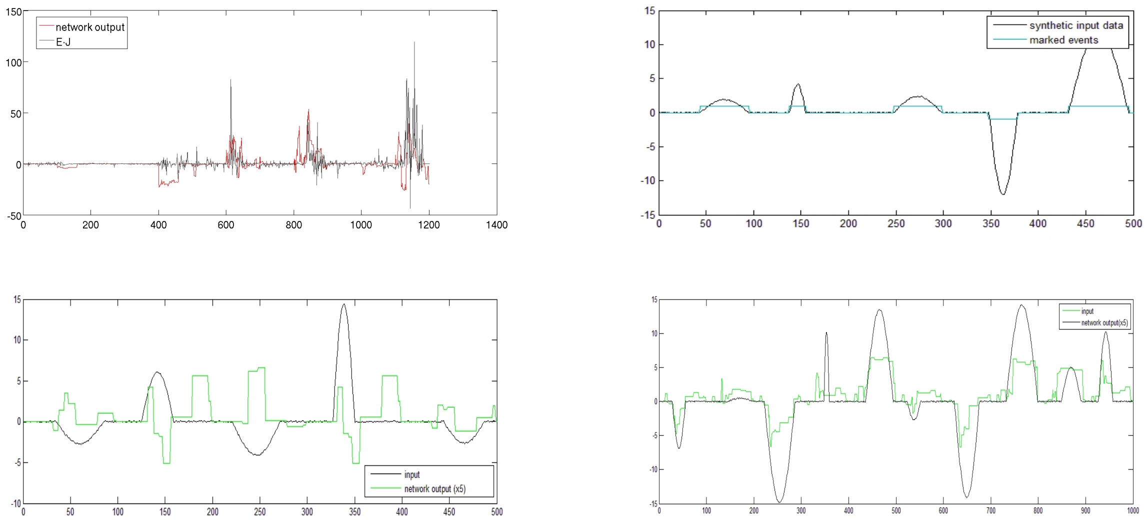

Figure 1: NN detection of real ECR events after training on Cluster data (top left), NN training on synthetic data (top right), and NN detection of synthetic ECR events (bottom left and right). The detection of synthetic ECR events is considerably improved in the bottom right panel, where the training set was larger, although the the NN had a smaller number of neurons.

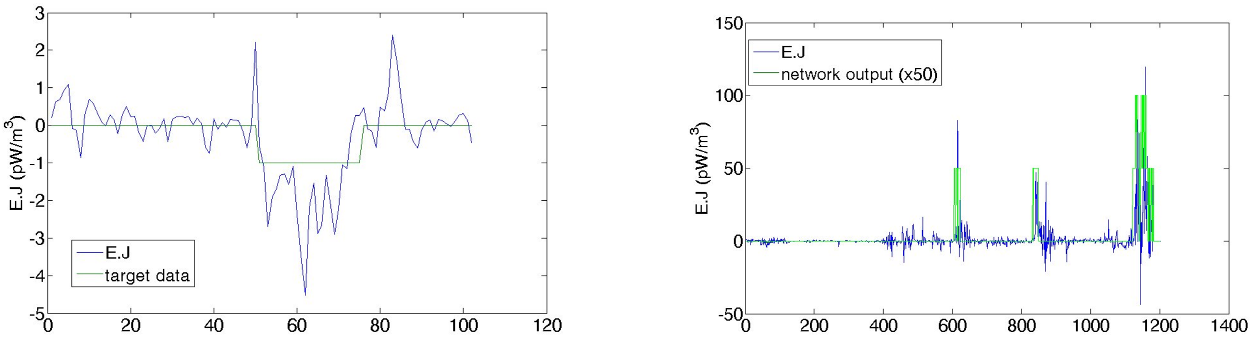

Figure 2: *Left:* CGR event from July 31, 2001, 21:31–22:12 UT, presented to the NN in the training data set. The duration of the event (-1 in the green line) is comparable (even if shorter) with the duration of the non-event (0 in the green line). *Right:* Test run output, using Cluster data from August 17, 2001, 9:00–17:00 UT. The NN identifies correctly the strongest events. The time resolution of the data points is 24 s in both plots.

One difficulty encountered in the tests with ECR data was the unstable NN behavior, with fast growth of the weights (defining the connections between neurons) and error (i.e. the difference between the actual response of the NN and the target response) to numerical overflow. In order to check the software implementation, the NN was used with the radar data set from *Wing et al., 2003 (Radio Science, 38(4), 1063, doi:10.1029/2003RS002869)*. A key outcome of this check was the observation that one of the parameters used in training the NN, the learning rate, had a strong influence on the convergence of the training and on the speed of this convergence. When the learning rate was too large, the training process could indeed get unstable. Two more important features proved to be the stop condition and the intrinsic variability of the network. Thus, the stop condition had to be formulated in terms of relative decrease in the error, as opposed to a fixed number of iterations. The intrinsic variability of the network was related to the randomly selected initial weights and the tests on radar data indicated that the performance could vary by more than 10 % for a given network configuration. In practice, the network was trained several times, and the best instance selected for further operation.

By properly adjusting the learning rate, the size of the sliding window, and the stop condition for the training process, it was possible to duplicate the results of *Wing et al., 2003* and to obtain encouraging results with ECR synthetic data (Figure 3, left). Subsequently, the training set based on Cluster observations was increased, by using 3/4 of the 555 ECR events identified in the plasma sheet data from 2001, 2002, and 2004, and the trained NN was tested on the remaining 1/4 ECR events. The result of this test (Figure 3, right) indicates a fairly good match between the NN output and the ECR events identified before, suggesting that NNs may soon provide an efficient mean to explore Cluster data (as well as data from other missions).

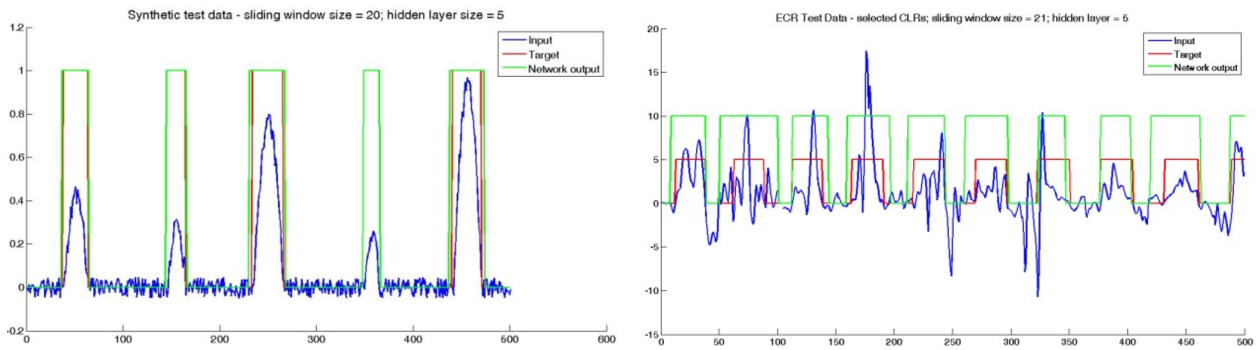

Figure 3: *Left:* NN detection of synthetic ECR events, after properly adjusting the learning rate, the size of the sliding window, and the stop condition. The network output (green) is almost identical to the desired target response (red). *Right:* NN detection of real data ECR events, identified previously by a semi-automated procedure developed at Umeå University. The plot, showing a selection of a few events, uses the same color code as in the left panel.