# NEURAL NETWORK BASED FAULT DETECTION IN HYDROPONICS

**K. P. Ferentinos[1], L. D. Albright[1], B. Selman[2]**

*[1]Department of Agricultural and Biological Engineering*
*[2]Department of Computer Science*
*Cornell University, Ithaca, NY 14853, U.S.A.*

Abstract: A fault detection model for hydroponic systems, based on the feedforward Neural Network methodology was developed. Three kinds of faults were considered: mechanical, sensor and biological faults. In this paper, a preliminary detection system is presented, which generally detects the existence of any faulty situations. In the developed network, only the two first kinds of faults were considered. Biological faults, because of their particularities, were treated separately and some of their characteristics are presented at the end. *Copyright © 2001 IFAC*

Keywords: Neural Networks, Fault Detection, Hydroponics, Backpropagation Algorithms.

## 1. INTRODUCTION

The goal of every greenhouse facility is to maximize quantity as well as quality of production. This maximization is achieved using automated systems in order to control the environment inside the greenhouse, so optimal conditions for the specific cultivated plant are approximated. In addition, hydroponic cultivation gives the opportunity to control root environment precisely and thus to have more extensive plant production system control.

Loss of control over a greenhouse is a phenomenon usually having negative effects on the production and, consequently, the profit of the facility. Certain failures (faults) are easily noticed, but others are quite difficult to detect. Of course, a feedback-controlled greenhouse may be able to maintain desired conditions even when some parts of the control mechanism are out of order. However, especially when faults concern the plant, the effects may be disastrous for the entire production. Thus, detection and diagnosis of possible faults becomes very important. In particular, that which is the most important is fast detection of incipient faults. That is, detection at the earliest possible stage of slowly developing faults, as well as the quick identification of the problem. In order to have quick detection, an on-line identification system is necessary.

A feedforward Neural Network (NN) based fault detection system was developed. The main area of concentration was deep-trough hydroponic systems. This becomes easier in hydroponics, for several environmental variables of the plant can readily be monitored. The cultivated plant was lettuce (*Lactuca sativa*, cv. Vivaldi). The main variables of the root environment that can be monitored and can give an image of the real situation of the system are the nutrient solution's pH, Electrical Conductivity (EC), dissolved oxygen (DO) as well as its temperature and in addition, the transpiration rate of the plants.

## 2. MATERIALS AND METHODS

Behaviors of normal and malfunctioning systems may differ completely and those differences may be detected by measuring the environmental variables mentioned before. The main advantage of the use of Neural Networks is that the exact relation between the values of those environmental variables and the situation of the system (normal of faulty), is not needed. This means that it is not required to have an accurate model of the hydroponic system, from which several residuals or fault signatures would be extracted and used for the final fault detection. Neural Networks have been proved capable of identifying faults in several complex biological processes (Parlos *et al.*, 1994; Sorsa *et al.*, 1991; Venkatasubramanian and Chan, 1989; Watanabe *et al.*, 1989) in which, neither analytical models nor intermediate residual calculations were used.

Lettuce plants were cultivated in a continuous production deep-trough hydroponic system. They were transplanted into the system after growing from seed in a growth chamber for 11 days. Every two days, seven new plants were transplanted to the system, while seven plants of the age of 28 days (from seeding) were harvested. Consequently, the three ponds of the system had constantly the same number of plants of the same age. This made the system somewhat stable or, more precisely, periodically stable, with a period of two days. The important advantage of this method, except for having a quasi-stable system, is that a continuous production of plants was achieved (harvest every two days), which resembles real-life hydroponic production systems more closely than other techniques previously used in neural network modeling of hydroponics (Ferentinos, 1999).

Desired values of the environmental parameters during the operation of the hydroponic system were an air temperature of 24° C during the day and 19° C during the night, relative humidity from 30% to 70% and a light integral of 17 $moles \cdot m^{-2} \cdot d^{1}$. For the nutrient solution, the pH set point was 5.8, the EC was maintained between 1150 and 1250 $microS \cdot cm^{-1}$ and the DO was maintained between 6 and 7.5 *mg/L*.

### *2.1 Normal and Faulty Situations*

The procedure of training the NN requires an accurate definition of "normal operation", defined in our case as unstressed plants in a system that is in control. We need not express this normal situation by means of specific values of the environmental variables, because neural networks do not need such a representation in order to learn the pattern. Thus, we need to know only when the system and, in extension, the plants are in conditions considered to be normal by the producers, and also to know which training data sets correspond to those normal conditions. The values of the measured variables mentioned before were considered to be the normal values for the system and the plants were considered to be normal as long as they appeared healthy. We also need to define the "faulty operation" and to categorize this kind of operation into different types of faulty operations, one for each different kind of fault. In order to take data sets for each kind of fault, we have to impose those faults and take the corresponding measurements of the microenvironment variables.

Because the NN was trained off-line (meaning that the data sets were first collected and then used for training), there was no way of mistakenly having unhealthy plants in data sets of "normal operation". The "faulty operation" consisted of three different kinds of faults:

- Faults in actuators of the hydroponic system,
- Faults in sensors of the hydroponic system and
- Faults in the plants themselves.

More specifically, the faults considered, by category, are the following:

Mechanical faults. These are failures in mechanical parts of the hydroponic system, such as: a) pH control pump is out of order, or b) circulation pump is out of order.
Sensor faults. The ones considered were: a) pH sensor failure and b) EC sensor failure.
Biological faults. These are problems in the cultivated plants themselves and are divided into: a) root area faults and b) shoot area faults.

For the first two categories of faulty operations, real data exist, as a lot of sensor and actuator failures were encountered during the set-up of the system. In addition, several faults were especially imposed in order to train the NN model and investigate its inherent fault detection capabilities. For the third category however, faults were imposed directly on the plants. To cause or to try to imitate the effects of a possible fault in the root zone, the plants were removed from the ponds and the root were exposed to air for intervals of five minutes. In the case of imitating a modest fault in the shoot area, leaves were disturbed (mechanically) for intervals of five minutes and slightly damaged in doing so. Finally, to imitate more permanent damage to the plants, several experiments were performed by cutting several leaves of each plant or by covering each plant with transparent plastic bags.

### *2.2 Neural Network Fault Detection Model*

The feedforward methodology of neural networks was used. The inputs of the NN were the environmental parameters (air temperature, relative humidity and light intensity), the measurable variables of the microenvironment of the plant (pH, EC, DO, nutrient solution temperature and transpiration rate of the plants) and the control signals of the pH and the DO control schemes (amounts of acid and oxygen added, respectively). Each output of the NN corresponded to a specific fault and there was also one output that corresponded to normal operation.

Several different architectures of one-hidden-layer and two-hidden-layer networks were tested, with two different activation functions (logistic and hyperbolic tangent). A new methodology for optimal network design and parameterization, based on Genetic Algorithms, was developed, but its results are incomplete, so only results of the conventional approach are presented. The training methodology was the Backpropagation Training Algorithm (Rumelhart *et al.*, 1986). Four different multidimensional minimization algorithms (steepest descent, conjugate gradient, quasi-Newton and Levenberg-Marquardt algorithm) were tested. An on-line adjustable learning rate performed better than a constant one. In the steepest descent and the conjugate gradient algorithms the Hessian was used at every iteration to solve for the "best" learning rate. For the other two algorithms, the "best" learning rate was calculated with an approximate line search using a cubic interpolation. The final NN fault detection system was tested using new data and its generalization capabilities were explored.

In addition to the network inputs listed before, one-step and two-step histories of the pH, EC and DO variables were included. That is, for each of these variables, three inputs existed: one for time $t$ (current time), one for time $t$-1 (previous time step) and one for time $t$-2 (two time steps before). Thus, the network had 15 inputs. The time step was 20 minutes.

The final neural network is going to have one output for the normal operation and an output for each of the faults considered. However, the amount of data collected so far is not yet sufficient to train such a network and in addition to test its performance. Therefore, a simpler neural network was trained and tested as a preliminary step. This NN had all the inputs presented before except for the transpiration rate, but only one output, a binary output having the value zero corresponding to normal operation and one corresponding to faulty operation.

## 3. RESULTS

In this paper, a preliminary investigation of the performance of the general normal/faulty operation detection neural network is presented. All mechanical and sensor faults of the systems were treated as a general "faulty situation". Biological faults showed no correlation with any measured variable except for the transpiration rate. On the other hand, transpiration rate was not affected by the first two fault categories. Therefore, biological faults were not considered in this preliminary model and in addition, transpiration rate was not used as an input to the network. The following section describes the training and evaluation of the preliminary NN fault detection system, while the next section analyzes some first results of some imposed biological faults.

### *3.1 Neural Network Performance*

The NN model was trained with experimental data collected from all three ponds of the system, for both normal and faulty situations. Approximately 15 (non continuous) days of data of both normal and faulty operations formed the final training set. The time step of these data was 20 minutes. Thus, the NN training was based on 1050 entries for each input.

The training process included two basic parts. The first part, the preliminary training process, determined the best combination of network architecture and training algorithm. This was achieved by training several candidate network topologies (both 1-HL and 2-HL networks) with all four training algorithms and comparing the results. The second part of the training process, which was the *basic training process*, focused on training the best combination of architecture and algorithm. Based on the preliminary training, the network architecture/algorithm combination that gave the best results was the 2-HL NN with 9 nodes in the first hidden layer and 9 nodes in the second hidden layer, trained with the quasi-Newton algorithm.

The basic training process had a goal to further train the selected NN with the best possible algorithm for this system and architecture, which was proven to be the quasi-Newton algorithm. Many different random initial network parameters were tested in that training. Also, several values of the coefficient of the penalty term for the regularization ($\lambda$), varying by the order of 5, were tried. Both logistic and hyperbolic tangent activation functions (functions of hidden nodes) were tested. The results of these tests showed that the value of $\lambda$ that leads to the minimum sum squared error (SSE) was $\lambda$= 0.05 and that the network with logistic activation functions performed better than the one with hyperbolic tangent activation functions. Thus, the final NN model consisted of a network with 15 inputs, two hidden layers with 9 nodes each that have logistic activation functions, and one output. The preferred training algorithm was the backpropagation training algorithm using the quasi-Newton multidimensional minimization algorithm with parameter $\lambda$=0.05.

Testing consisted of presenting new data to the trained NN model and exploring its generalization capabilities. The main goal in a fault detection system is not only detection of the existence of a fault, but also its rapid detection. Especially, when we deal with incipient faults, the time factor becomes more important because these faults are more difficult to detect as they begin. Six testing data sets were presented to the NN fault detection system. Each set starts with data of normal operation and some specific fault is imposed at some known time, except for the last data set that contains only normal data.

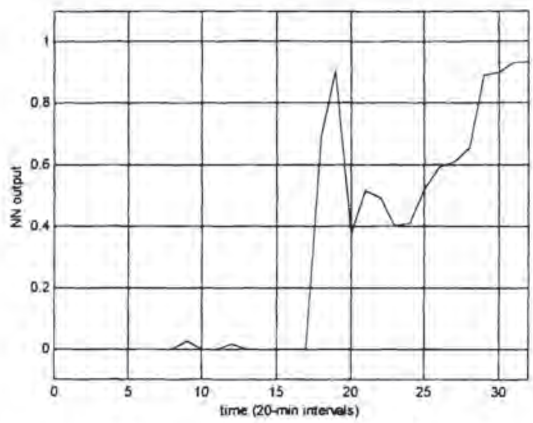


Fig. 1. Output of the FDNN for testing data set 1 (fault imposed at the 16th interval).

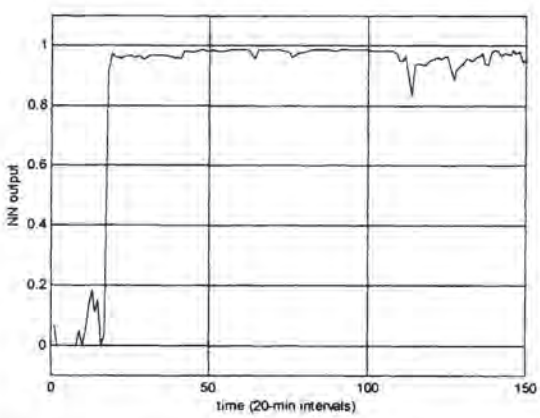


Fig. 2. Output of the FDNN for testing data set 2 (fault imposed at the 18th interval).

The output of the NN was considered to represent a faulty operation if it has a value greater than 0.5, while for values smaller than 0.5 a normal operation was assumed. Fig. 1 shows the NN output on the first data set. The "pH control pump out of order" fault was introduced at the 16th interval. The fault was detected by the network within only two step intervals (point 18), that is a period of 40 minutes. After that the network gives a steady and strong indication that the operation is not normal, with values very close to 1.

Fig. 2 shows the network output for the second testing set, which contains a "circulation pump out of order" fault. The data again start with normal operation and the fault is introduced in the 18th interval. It is clear here that this kind of fault is detected very fast. Even from the 18th data point, the output of the network is 0.69, which is considered as a weak fault indication. The next data point gives a value of 0.91 that strongly indicates the existence of faulty operation. A disadvantage here can be considered the fact that the output drops below 0.5 (normal operation) 40 minutes after the occurrence of the fault and stays in that area for an hour. After that period it returns to the faulty indication. This is not something important, if we consider the fact that the indication of some fault is already given at a very early stage and also that the kinds of faults considered are supposed to be irreversible without the interaction of a human factor with the system.

In fig. 3, the network response in the third testing set is presented. This set contains the same fault type as the previous one that is now introduced in the 9th data point. This time it takes 20 minutes to the network to indicate a possible fault (value 0.64) while in a total of one-hour period after the introduction of the fault, the output becomes high enough (0.82) to strongly indicate the existence of faulty operation.

The fourth testing data set has 167 20-minute intervals and the "failure in pH sensor" fault was introduced in the 16th point. The output of the NN model is presented in fig. 4. Similarly to the previous case, it takes one time step (20 minutes) for the network to indicate a possible fault with an output of 0.60, while at the next 20-minute step the output becomes 0.91 and stays in that area. The rather periodical fluctuations of the output are caused by the nature of the sensor fault. This kind of fault was reproduced by adding a periodically changing noise to the readings of the pH sensor. The form of noise is

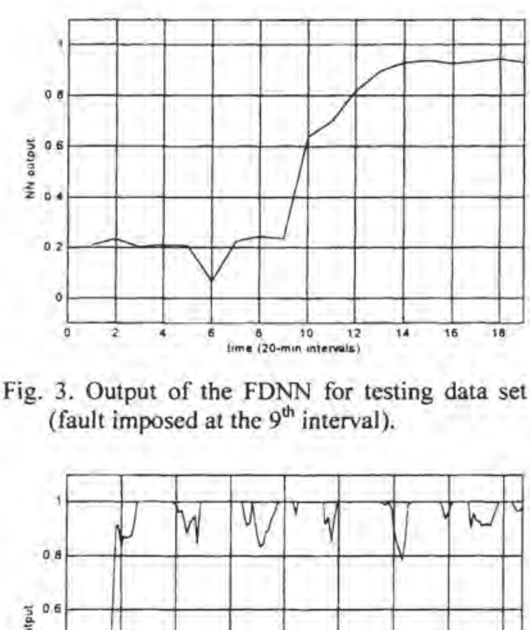


Fig. 3. Output of the FDNN for testing data set 3 (fault imposed at the 9th interval).

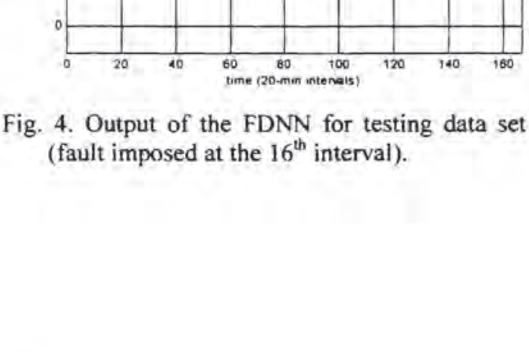


Fig. 4. Output of the FDNN for testing data set 4 (fault imposed at the 16th interval).

a sine function. Thus, in the points where the noise becomes small, the fault confidence of the network decreases. However, this does not cause the network to exit the "faulty situation" area.

The fifth testing data set contains the implementation of the "failure in EC sensor" fault. The set has 69 20-minute intervals and the fault was introduced in the $16^{th}$ interval. As can be seen in fig. 5, the fault is detected 4 hours after its beginning. Moreover, several hours later, when the noise of the sensor failure becomes small, the network indicates normal operation for that period. It seems that this specific fault causes some problems for the detection process, probably because no information about the control signal for the EC is present; EC was controlled manually in the hydroponic system.

Finally, the last testing data set contains only normal operation data and it was used to check the network ability to recognize continuously varying normal behavior. The output of the network is shown in fig. 6. The graph shows a period of almost one whole day and the output is always below 0.4, indicating normal operation during the entire period.

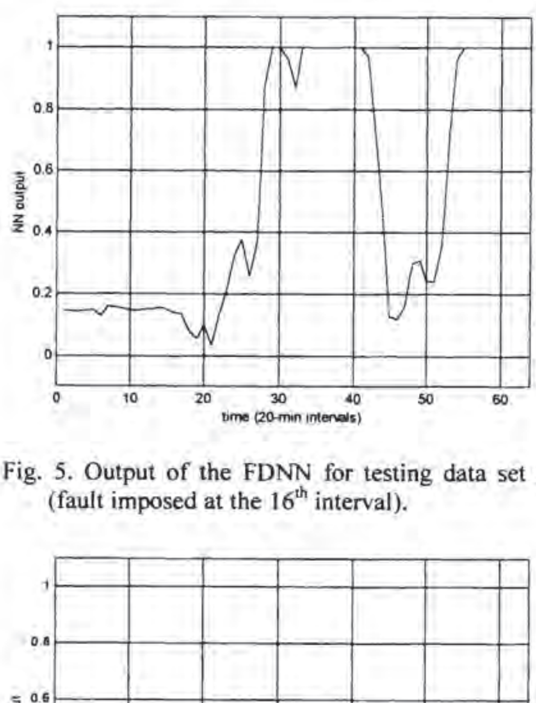


Fig. 5. Output of the FDNN for testing data set 5 (fault imposed at the $16^{th}$ interval).

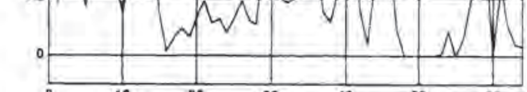


Fig. 6. Output of the FDNN for testing data set 6 (normal operation).

*3.2 Biological Faults*

Several experiments consisted of imposing different faults on the plants in order to examine their effect on the monitored variables of the nutrient solution of the system. Four different series of experiments were performed. In the first, most of the largest plants were removed from the pond for five minutes. In this way, possible problems in the root zone of the plants were imitated. In the second series, several leaves of the largest plants were removed. This action imposed permanent damage on the plants and imitated the effects of major problems in the shoot zone of the plants. A similar but less influential series of experiments was the one in which leaves of the plants were disturbed for intervals of five minutes and slightly damaged. These experiments imitated shoot problems less important than the ones imitated in the previous series of experiments. Finally, in the fourth series, the largest plants (of ages of 23, 25 and 27 days) were covered with transparent plastic bags. This imposed a temporary fault that imitated minor problems in the shoot zone.

Effects of these biological faults, unfortunately, were not significant enough to be used in a fault detection scheme. The pH and the electrical conductivity appeared not to be affected at all by the faults. The transpiration, a variable known to be drastically affected by the condition of the plants, was so highly correlated with the environmental conditions of the greenhouse (temperature, light intensity and relative humidity) that effects of plant damage, even when seemingly severe, were not noticeable in most of the cases. Even in the experiments in which some leaves of the plants were cut and where one would expect major impacts in the transpiration ratio, the effects were "hidden" by the high correlation of the transpiration with the environmental parameters, especially temperature and light intensity. In fig. 7, the differences between the cumulated evapotranspiration rates between two of the tanks of the hydroponic system are shown. Every set of points represents periods of two days, between transplanting. At those points, the cumulated differences were reinitialized. At around time interval No. 2000, half of the leaves of the largest plants of tank 1 were removed, while nothing changed in tank 2. One would expect that the transpiration would be reduced in tank 1, thus the difference between transpiration rates of tank 2 and tank 1 would increase. As can be seen in the graph, this is clearly not the case. This can be explained if the additional transpiration from the cuts of the removed leaves is taken into account. Thus, no indication of the fault appeared in this case. The differences between tanks even when normal conditions exist in both tanks, is caused by differences in the air movement above the tanks, which lead to difference transpiration rates. Similar results were obtained by the imposed faults of disturbing the leaves or removing the largest plants from the tanks for periods of five minutes.

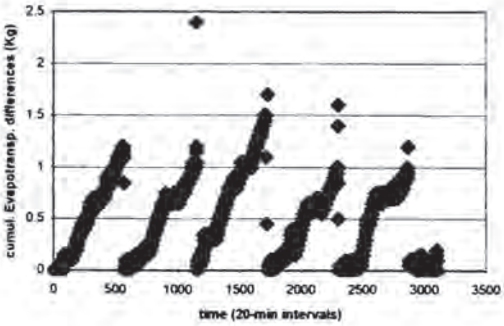


Fig. 7. Cumulative evapotranspiration differences between two tanks of the hydroponic system ("cutting the leaves" fault imposed at the 2000$^{th}$ interval).

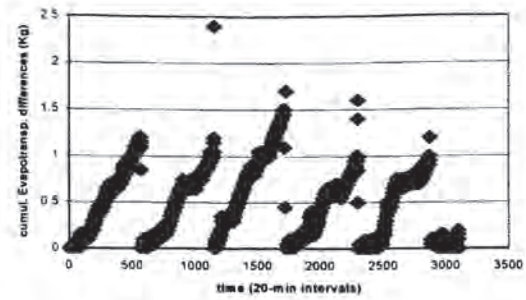


Fig. 8. Cumulative evapotranspiration differences between two tanks of the hydroponic system ("covering the plants" fault imposed at the 1000$^{th}$ interval ).

The only biological fault that gave an indication of its existence in the rate of transpiration was the fourth type, during which the largest plants were covered with transparent plastic bags. As shown in fig. 8, significant increase in the difference of cumulated transpiration rates between tank 2 and tank 1 occurred after the introduction of the fault in tank 1 (at time interval No. 1000). It is clear that during transpiration rate was reduced in tank 1 (the difference from transpiration of normal plants increased) during the six following to the introduction of the fault days. During these days, covered plants existed in the tank. After that period, all covered plants had been harvested, thus transpiration rate in tank 1 returned back to normal (last two sets of points in the graph).

More data of this type of biological faults have to be collected so that a detection system can be developed. Because of the large amount of history data of the transpiration of the system that is needed by such a detection scheme, this model is going to be separate from the NN model the detects the other kinds of faults (mechanical and sensor faults).

## 4. CONCLUSIONS

The methodology of constructing a neural network based fault detection system in hydroponics was developed. Some preliminary results are presented. The results reflect a simplification of the more general NN model in that it has only one output and tries to classify specific data into either normal or faulty operation. These testing results indicate that this simplified network is capable of detecting the faulty situation is very short time in most cases. The rapidity of detection suggests time steps smaller than the 20-minute time step used here. The results show that the NN has useful generalization capabilities. A next step is to develop a more detailed fault detection system that has one output for each specific fault.

## ACKNOWLEDGEMENTS

This research was supported by BARD Research Project No. IS-2680-96.

## REFERENCES:

Ferentinos, K.P. (1999). *Artificial Neural Network Modeling of pH and Electrical Conductivity in Hydroponic Systems.* MS Thesis, Cornell University, Ithaca, NY.

Parlos, A. G., J. Muthusami, and A. F. Atiya. (1994). Incipient Fault Detection and Identification in Process Systems Using Accelerated Neural Network Learning. *Nuclear Technology*, **vol. 105**, pp. 145-161.

Rumelhart, D. E., G. E. Hinton, and R. J. Williams. (1986). Learning Representations by Back-Propagating Errors. *Nature*, **vol. 323**, pp. 533-536.

Sorsa, T., H. N. Koivo, and H. Koivisto. (1991). Neural Networks in Process Fault Diagnosis. *IEEE Transactions on Systems, Man, and Cybernetics*, **vol. 21**, No. 4, pp. 815-825.

Venkatasubramanian, V. and K. Chan. (1989). A Neural Network Methodology for Process Fault Diagnosis. *AIChE Journal*, **vol. 35**, pp. 1993-2002.

Watanabe, K., I. Matsuura, M. Abe, M. Kubota, and D. M. Himmelblau. (1989). Incipient Fault Diagnosis of Chemical Processes via Artificial Neural Networks. *AIChE Journal*, **vol. 35**, pp. 1803-1812.