Deep Learning-Based Fault Detection and Isolation in Solar Plants for Highly Dynamic Days

Sara Ruiz-Moreno
Dept. de Ingeniería de Sistemas y Automática
University of Seville
Seville, Spain
srmoreno@us.es

Antonio J. Gallego
Dept. de Ingeniería de Sistemas y Automática
University of Seville
Seville, Spain
agallego2@us.es

Adolfo J. Sanchez
Dept. of Mechanical, Biomedical and Manufacturing Engineering
Munster Technological University
Cork, Ireland
adolfo.j.sanchez@ieee.org

Eduardo F. Camacho
Dept. de Ingeniería de Sistemas y Automática
University of Seville
Seville, Spain
efcamacho@us.es

***Abstract*—Solar plants are exposed to numerous agents that degrade and damage their components. Due to their large size and constant operation, it is not easy to access them constantly to analyze possible failures on-site. It is, therefore, necessary to use techniques that automatically detect faults. In addition, it is crucial to detect the fault and know its location to deal with it as quickly and effectively as possible. This work applies a fault detection and isolation method to parabolic trough collector plants. A characteristic of solar plants is that they are highly dependent on the sun and the existence of clouds throughout the day, so it is not easy to achieve methods that work well when disturbances are too variable and difficult to predict. This work proposes dynamic artificial neural networks (ANNs) that take into account past information and are not so sensitive to the variations of the plant at each moment. With this, three types of failures are distinguished: failures in the optical efficiency of the mirrors, flow rate, and thermal losses in the pipes. Different ANNs have been proposed and compared with a simple feedforward ANN, obtaining an accuracy of 73.35%.**

***Index Terms*—artificial intelligence, artificial neural networks, fault detection and diagnosis, solar energy, parabolic-trough collectors**

I. Introduction

With the advancement of industry and technology, energy demand has risen dramatically and is expected to continue to increase drastically in the coming years [1]. This augments the consumption of the planet's resources and emissions into the atmosphere. That is why now the reduction of greenhouse emissions is a priority [2] and the use of renewable energy sources is necessary for the planet, and sustainable development [3].

The renewable energy sources being harvested nowadays are wind, hydraulic and solar. Specifically, solar energy is increasingly attractive [4], as it is the most abundant. This work focuses on the use of solar thermal energy, in particular on parabolic-trough collectors (PTC), which consist of parabolic mirrors that concentrate the solar irradiance onto a tube containing a fluid to produce steam and drive a turbine generator [5].

Industrial plants are increasingly equipped with sensors and actuators, among many other components, which increases the risk of failure. Therefore, fault detection and diagnosis (FDD) techniques are of vital importance. Fault detection consists of warning of the existence of a fault, usually through the use of residuals, and triggering an alarm. Diagnosis provides more information about the fault and is broken up into isolation (determines its location and type) and identification (determines its magnitude). Fault detection and isolation (FDI) is a subcategory of FDD techniques.

Many FDD techniques exist [6], [7]. Quantitative model-based methods are founded on mathematical functional relationships through e.g., observers [8], or parity equations [9]. Qualitative model-based methods express the model in terms of qualitative functions with e.g., fuzzy logic [10]. Process history-based methods use large amounts of historical data to obtain knowledge from experience, generally using machine learning [11]–[13].

In the field of solar thermal energy [7], and especially in PTCS, where the faults are strongly coupled, this field has not been explored in depth. Some works address the detection phase [14], and others make a broad diagnosis, considering the PTC as a single subsystem [15]. This paper is the continuation of a previous work [16] that proposed a methodology based on artificial neural networks (ANNs) and two decoupling stages by analyzing the rise time and the effect of defocusing.

Renewable energies depend strongly on weather conditions and are difficult to predict. Dealing with the unreliability of the sources is a challenging factor for the system safety and management [17]. The solar radiation that reaches the PTCs depends strongly on the amount, sizes, and speed of clouds [2],

This research has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (OCONTSOLAR, grant agreement No 789051) and the Spanish Ministry of Science, Innovation, and Universities (Grant FPU20/01958).

which hinders the application of FDD techniques. Moreover, the rise time analysis and residuals lose effectiveness due to the strong transients that lead to false positives. The previous work assumed clear days to isolate the different faults; however, this work eliminates this assumption, for which it proposes the use of dynamic neural networks that take into account the temporal evolution of the system.

The main contribution of this work is the inclusion of highly complex irradiance days in the FDD system design for PTCs by using ANNs that look into the past. For this purpose, it uses real irradiance profiles that reflect the appearance of different types of clouds in the plant. Taking into account real radiation is a challenge since the behavior of a system changes plenty depending on the radiation profile, and it is not easy to generalize. When considering real different clouds, transients are introduced into the system and both radiation and flow rate change. This strongly modifies the dynamics of the plant and makes it very complex and difficult to manage. Therefore, neural networks that take into account past information are necessary to deal with this strong dynamic.

This paper is organized as follows. Section 2 provides a description of the system, including the model and the controller. Next, section 3 details the proposed methodology, introducing ANNs and the evaluation metrics. The simulation results are shown in section 4, and, finally, some discussion and conclusions are extracted in sections 5 and 6, respectively.

II. System description

This section aims to provide a brief description of the system. In this work, the simulations were carried out on the ACUREX plant [5], wich was located at the Plataforma Solar de Almería. It is conformed by 10 loops of 4 sngle-axis PTCs with 12 modules. The loops are 172 m long, with an active part (the one that receives solar radiation) of 142 m and a passive part of 30 m., and the fluid employed is Therminol 55 thermal oil. In this work, one loop of collectors is considered, see figure 1, as the proposed methodology can be applied to each loop independently.

Fig. 1. ACUREX collector loops.

TABLE I
Parameters and variables description.

Symbol	Description	Units
A	Cross-sectional Area	m^2
$\rho(t,T)$	Density	kg/m^3
$C(t,T)$	Specific heat capacity	J/(kg °C)
A_f	Transversal area of the interior pipe	m^2
G	Collector aperture	m
L	Pipe length	m
$T_a(t)$	Ambient temperature	°C
$T(t,x)$	Temperature	°C
$q(t)$	Flow rate	l/s
$I(t)$	Direct solar irradiance	W/m^2
K_{opt}	Optical efficiency	–
$n_o(t)$	Geometric efficiency	–
$H_l(t,T)$	Thermal loss coefficient	W/(m^2 °C)
$H_t(t,T)$	Metal-fluid heat transmission coefficient	W/(m^2 °C)
t	Time	s

A. Distributed parameter model

The distributed parameter model [18], [19] is used for simulating the plant. It describes the system with energy balances on the metal and the fluid circulating through the pipes. The model is computed by equations 1 and 2 with the notation given by table I and the subscripts m and f referring to metal and fluid, respectively. Multiplicative faults are considered and modelled as parameters $\alpha_{K_{opt}} < 1$, $\alpha_q \neq 1$ and $\alpha_{Hl} > 1$. The non-faulty situation is represented with values of 1 in these parameters. For simulating the system, the loop is divided into 1 m long segments and a simulation time of 0.25 s is used. The reader may refer to the previous paper [16] for more details on the model.

$$\begin{aligned}\rho_m C_m A_m \frac{\partial T_m}{\partial t} = \alpha_{K_{opt}} I K_{opt} n_o G + \\ -H_l G(T_m - T_a) - L H_t (T_m - T_f)\end{aligned} \tag{1}$$

$$\rho_f C_f A_f \frac{\partial T_f}{\partial t} + \alpha_q \rho_f C_f q \frac{\partial T_f}{\partial x} = L H_t (T_m - T_f) \tag{2}$$

The specific heat capacity C_f and the fluid density ρ_f are computed by equations 3 y 4.

$$C_f = 1820 - 3.478 T_f \tag{3}$$

$$\rho_f = 903 - 0.672 T_f \tag{4}$$

Equations 5 and 6 give the metal-fluid transmission coefficient H_t and the thermal losses H_l coefficient.

$$\begin{aligned}H_t = q^{0.8}(2.17 \cdot 10^6 - 5.01 \cdot 10^4 T_f + 4.53 \cdot 10^2 T_f^2 + \\ -1.64 T_f^3 + 2.1 \cdot 10^{-3} T_f^4)\end{aligned} \tag{5}$$

$$H_l = 0.00249\,(T_f - T_a) - 0.06133 \tag{6}$$

The geometric efficiency n_o, also known as $cos(\theta)$, is computed by using the position of the mirror normal vector relative to the radiation beam vector and depends on the declination, latitude, hourly angle, solar hour, Julianne day and collector dimensions [20], [21].

B. Lumped parameter model flow control

A series feedforward controller is used for temperature reference tracking by manipulating the flow rate $q(t)$. The control signal is obtained from the lumped parameter model, which approximates the behavior of the system using the variation of the internal energy of the fluid. The selected sample time is 39 s [22], [23] and the control signal is given by equation 7, where $P_{cp} = \rho_m C_m$, T_{ref} is the reference temperature and T_{mean} is the mean value of the inlet and outlet temperatures T_{in} and T_{out}.

$$q(t) = \frac{n_o K_{opt} SI - H_l A(T_{mean} - T_a)}{P_{cp}(T_{ref} - T_{in})} \tag{7}$$

III. Proposed methodology

The method used is based on dynamic neural networks that classify the data into four groups: no fault, fault in K_{opt}, fault in q and fault in H_l. This section describes the general idea of artificial neural networks and the evaluation metrics used in this work.

A. Artificial neural network

An artificial neural network (ANN) is an approximator of nonlinear functions formed by the combination of linear regressors placed at the neurons, also called nodes. The origin of neural networks takes back to the first half of the 20th century [24] and now, with the development of technologies and computing, they are being applied to many different fields [25].

The multilayer perceptron is one of the most common ANNs, in which the neurons are arranged in layers of three types: an input layer, an output layer, and hidden layers that transform the data. The structure (i.e., the number of nodes and layers) and some neural network parameters are selected in an iterative process. An initial architecture is chosen, the ANN is trained, then evaluated, and the whole process is repeated until the desired results are obtained.

Each neuron solves a linear regression problem, and the weights are obtained in the training phase of the ANN, where the output error is used as cost function. The backpropagation algorithm [26], [27] allows the ANN to compute the partial derivatives for adapting the weights iteratively, according to the direction of greatest decrease of the cost function. The scaled conjugate gradient algorithm [28] is a backpropagation method that uses conjugate directions for faster convergence.

The output of each neuron represents an activation state using activation functions. This work uses hyperbolic tangent sigmoid functions for all layers, except for the last one, which contains a softmax function that transforms the data into the range $[0, 1]$. The ANN has one output for each class. Values near 1 correspond to higher possibilities of being the correct class. When the ANNs are dynamic, they take into account the variation of the variables with time. In this case, the inputs contain delays to take account of the dynamic behavior of the system.

For training the ANNs, many simulations are made to obtain a representative dataset. Then, it is scaled in the range $[-1, +1]$, randomized (conserving the selected time windows), and divided into three subsets: a training set for adjusting the parameters, a validation set for validating online and readjusting some parameters, and a test set for estimating the behavior of the ANN with new data.

B. Evaluation metrics

To validate the behaviour of the neural networks, the following metrics are used in this work. The accuracy is used either in the training process or in the simulations. The F1-score is used in the simulations.

- Accuracy: the rate of true positives (TP) over all instances: true positives, false positives (FP), true negatives, and false negatives (FN).

$$Acc = \frac{TP + TN}{TP + FP + TN + FN} \tag{8}$$

- Precision: the rate of correct TP over all positive-assigned instances.

$$Pre = \frac{TP}{TP + FP} \tag{9}$$

- Recall: the rate of correct TP over all positive instances.

$$Rec = \frac{TP}{TP + FN} \tag{10}$$

- F1-score: the harmonic mean of recall and precision.

$$F1 = 2 \cdot \frac{Rec \cdot Pre}{Rec + Pre} \tag{11}$$

IV. Simulation Results

The results obtained from different simulations are presented in this section. Different ANNs were trained, and the most promising ones were selected to perform batches of simulations to compare their performance.

For all the experiments, the distributed parameter model divides the collector loop into 172 segments of 1 m, and it is solved with an integration time of 0.25 s. The controller sample time is 39 s [23].

Ten real irradiance profiles were obtained. The seven of them that are shown in figure 2 were used for creating the dataset, and the remaining three were used for validation purposes. The dataset was created from simulations corresponding to 1259256 instances (13462 hours) of one-day simulations eliminating data out of range 10:00-17:59 hours, with reference temperatures between 220^oC and 300^oC, control signals obtained from the controller and with constant values and fault values of $\alpha_{K_{opt}}$ between 0.1 and 0.9, α_q between ± 0.1 l/s and ± 0.5 l/s and α_{H_l} between 1.1 and 2.

One of the most important steps when training neural networks is selecting the set of variables that represent the data. In this case, the outputs are four classes: no-fault, K_{opt} fault, q fault and H_l fault. The inputs are the same that use the baseline static ANN. For selecting the evolution with time,

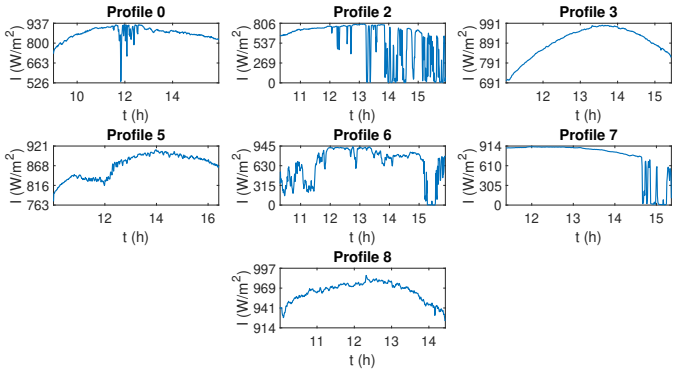

Fig. 2. Real irradiance profiles used for training.

different tests were carried out. Taking the example of variable z, $z(k-i)$ is the value from i instants before, $\bar{z}(k-j:k-i)$ is the mean value between instants $k-i$ and $k-j$ and $\bar{z}^w(k-j:k-i)$ is the weighted mean between instants $k-i$ and $k-j$. Note that here q is not the actual value but the one assumed from the sensor measurements since the fault is unknown. The most relevant sets of inputs are listed below:

- Static ANN (v16): $X = [T_{in}(k), T_{out}(k), dT_{out}(k)/dT, T_a(k), I(k), q(k), n_o(k)]$
- ANNs v20: $X = [T_{in}(k), T_{out}(k), T_{out}(k-1), T_{out}(k-15), T_{out}(k-30), T_a(k), I(k), q(k), q(k-1), q(k-15), n_o(k)]$
- ANNs v23: $X = [T_{in}(k), T_{out}(k), \bar{T}_{out}(k-5:k-1), \bar{T}_{out}(k-15:k-6), \bar{T}_{out}(k-30:k-16), T_a(k), I(k), q(k), \bar{q}(k-3:k-1), \bar{q}(k-10:k-4), n_o(k)]$
- ANNs v24: $X = [T_{in}(k), T_{out}(k), \bar{T}_{out}(k-5:k-1), \bar{T}_{out}(k-15:k-6), \bar{T}_{out}(k-30:k-16), T_a(k), I(k), \bar{I}(k-5:k-1), \bar{I}(k-20:k-6), q(k), \bar{q}(k-3:k-1), \bar{q}(k-10:k-4), n_o(k)]$
- ANNs v25: $X = [T_{in}(k), \bar{T}_{in}(k-5:k-1), \bar{T}_{in}(k-20:k-6), T_{out}(k), \bar{T}_{out}(k-5:k-1), \bar{T}_{out}(k-15:k-6), \bar{T}_{out}(k-30:k-16), T_a(k), I(k), \bar{I}(k-5:k-1), \bar{I}(k-20:k-6), q(k), \bar{q}(k-3:k-1), \bar{q}(k-10:k-4), n_o(k)]$
- ANNs v26: $X = [T_{in}(k), T_{out}(k), \bar{T}^w_{out}(k-20:k-1), T_a(k), I(k), q(k), \bar{q}^w(k-15:k-1), n_o(k)]$
- ANNs v28: $X = [\bar{T}_{in}(k-5:k-1), \bar{T}_{in}(k-20:k-6), \bar{T}_{out}(k-5:k-1), \bar{T}_{out}(k-15:k-6), \bar{T}_{out}(k-30:k-16), \bar{T}_a(k-5:k-1), \bar{I}(k-5:k-1), \bar{I}(k-20:k-6), \bar{q}(k-3:k-1), \bar{q}(k-10:k-4), n_o(k)]$
- ANNs v29: $X = [T_{in}(k), \bar{T}_{in}(k-5:k-1), \bar{T}_{in}(k-20:k-6), \bar{T}_{in}(k-20:k-1), T_{out}(k), \bar{T}_{out}(k-5:k-1), \bar{T}_{out}(k-15:k-6), \bar{T}_{out}(k-30:k-16), \bar{T}_{out}(k-30:k-1), T_a(k), I(k), \bar{I}(k-5:k-1), \bar{I}(k-20:k-6), \bar{I}(k-20:k-1), q(k), \bar{q}(k-3:k-1), \bar{q}(k-10:k-4), \bar{q}(k-10:k-1), n_o(k)]$
- ANNs v30: $X = [T_{in}(k), \bar{T}_{in}(k-5:k-1), \bar{T}_{in}(k-20:k-6), \bar{T}_{in}(k-30:k-21), T_{out}(k), \bar{T}_{out}(k-5:k-1), \bar{T}_{out}(k-15:k-6), \bar{T}_{out}(k-30:k-16), \bar{T}_{out}(k-40:k-31), T_a(k), I(k), \bar{I}(k-5:k-1), \bar{I}(k-20:k-6), \bar{I}(k-30:k-21), q(k), \bar{q}(k-3:k-1), \bar{q}(k-10:k-4), \bar{q}(k-20:k-11), n_o(k)]$

TABLE II
ACCURACIES OF THE SELECTED NEURAL NETWORKS IN THE DATASET.

ANN	Acc train (%)	Acc valid. (%)	Acc test (%)
400-200 v16	95.0	94.9	94.8
200-100-50 v20	88.5	88.1	88.1
400-200 v23	89.6	88.9	89.1
49-32-28-12 v24	81.7	81.9	81.7
400-200 v25	91.1	90.8	90.7
200-100-50 v26	89.0	88.5	88.7
400-200 v28	89.8	89.5	89.4
200-100-50 v29	90.6	90.3	90.4
200-100 v30	92.4	92.2	92.0

TABLE III
ACCURACIES AND F1-SCORES FROM SIMULATION.

ANN	F1-score (%)				Acc (%)
	Faultless	K_{opt} fault	q fault	H_l fault	
400-200 v16	48.82	55.18	47.34	47.58	49.57
200-100-50 v20	33.23	54.77	44.16	31.41	42.27
400-200 v23	43.07	66.40	46.64	21.64	46.61
49-32-28-12 v24	52.17	60.65	47.06	50.86	52.57
400-200 v25	**88.21**	**82.56**	**65.53**	**55.17**	**73.35**
200-100-50 v26	18.01	49.25	42.48	13.46	36.11
400-200 v28	62.74	80.63	59.50	54.55	64.15
200-100-50 v29	74.51	80.08	61.03	56.79	66.75
200-100 v30	45.41	52.03	49.31	43.27	46.79

For each one of the ANN versions, different architectures were evaluated. The dataset was randomized, and the repeated instances were removed, remaining 1192296 data. The ANNs were trained by splitting the dataset into three subsets: training set (75%), validation set (10%), and test set (15%). Table II gathers the best architectures for each version in each subset, where, e.g., 400-400 indicates that the ANN has two hidden layers with 400 nodes in the first one and 200 in the second one. The first one (400-200 v16) is the baseline, which was trained using a different dataset with constant irradiances instead of real profiles. This table shows that the results worsen when considering real variable radiation. The best ANNs are from versions 25, 28, 29, and 30, the ones that contain mean values of T_{in}.

Figures 3 and 4 show the results with a fault of 0.8 in the optical efficiency occurring at 14:00 with irradiance profile 1. This is a clear day in which the ANN was perfectly able to detect the fault before 15:00.

In 5 and 6, profile 4 was used, which is an example of highly dynamic days with clouds passing by all day long. In this case, even the control is challenging, judging from the temperature tracking, and there are nonzero values of the output corresponding at flow rate and thermal loss from 11:30. A fault of -0,25 l/s in the flow rate was added at 13:00, and it was correctly alarmed around 13:15.

The last experiment is shown in figures 7 and 8 and corresponds to a day with moderate irradiance and a big cloud passing by. A 1.2 fault in thermal loss coefficient was added at 12:00 and alarmed before 13:00.

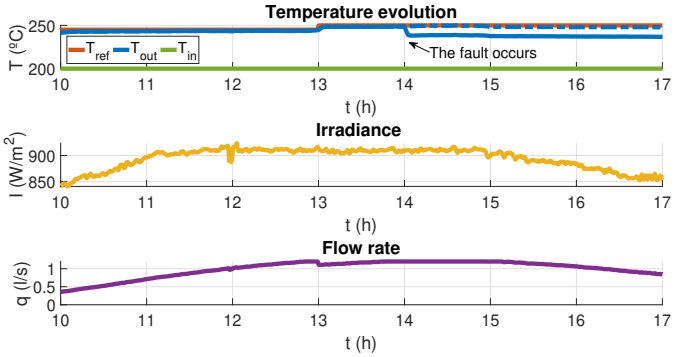

Fig. 3. Temperatures, irradiance and flow rate evolution from the first experiment with test profile 1 and a fault of 0.8 in the efficiency of the collectors after hour 14:00. The discontinuous line shows the non-faulty outlet temperature.

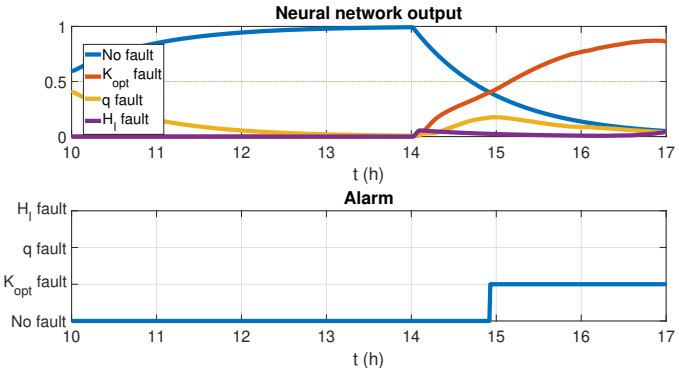

Fig. 4. Neural network output from the first experiment with test profile 1 and a fault of 0.8 in the efficiency of the collectors after hour 14:00.

V. Discussion

In this work, different neural networks were trained and tested, and different types of inputs were analyzed. The results reveal that the neural networks with a higher number of delays in the inputs and with delays in either the outlet or inlet temperatures perform better than the rest of the ANNs applied. Although the baseline ANN from the previous article presented higher accuracy with clear days, the study proves that its behavior worsens when considering real cloudy days. This work demonstrates the need to consider the dynamics of the system and include past data.

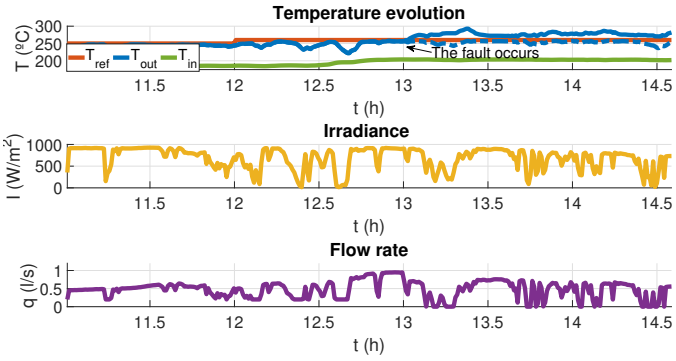

Fig. 5. Temperatures, irradiance and flow rate evolution from the second experiment with test profile 4 and a fault of -0.25 l/s in the flow rate after hour 13:00. The discontinuous line shows the non-faulty outlet temperature.

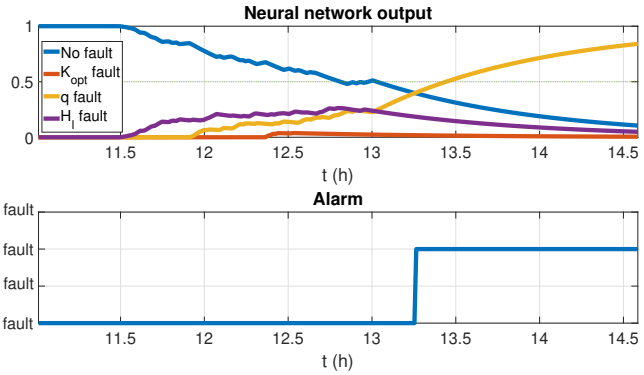

Fig. 6. Neural network output from the second experiment with test profile 4 and a fault of -0.25 l/s in the flow rate after hour 13:00.

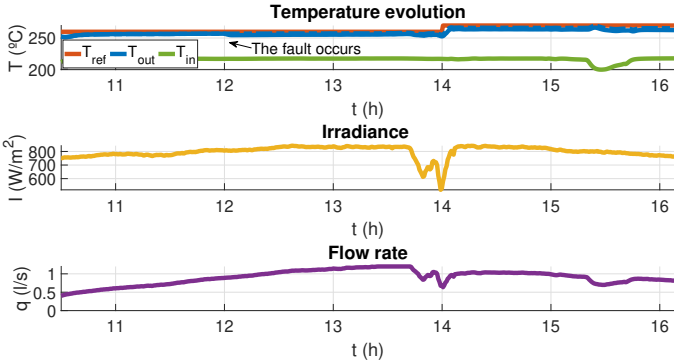

Fig. 7. Temperatures, irradiance and flow rate evolution from the third experiment with test profile 9 and a fault of 1.2 in the thermal loses after hour 12:00. The discontinuous line shows the non-faulty outlet temperature.

VI. Conclusions

This paper addresses the problem of FDI in PTC plants by using artificial neural networks with knowledge of present and past information. The proposed method can detect and

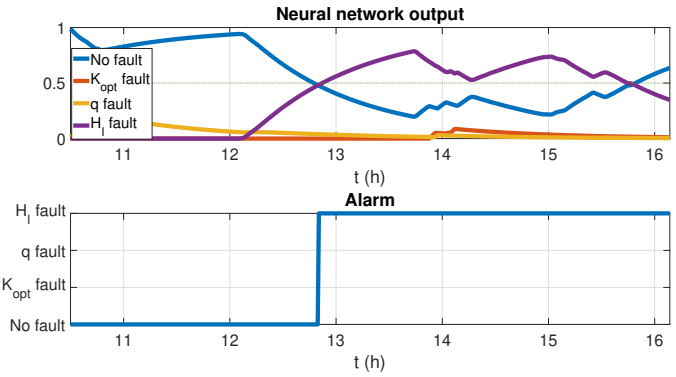

Fig. 8. Neural network output from the third experiment with test profile 9 and a fault of 1.2 in the thermal loses after hour 12:00.

isolate three types of faults in the collector area of the field. In addition, the use of past data from the plant enables the FDI system to detect faults even on highly variable days with extreme cloud conditions.

The method was validated by batches of simulations on the ACUREX plant. The proposed ANN outperforms the baseline ANN from previous work, obtaining an accuracy of 73.35% and an F1-score of 88.21% for the non-faulty case. As the baseline ANN showed better results for clear synthetic days, future development will be to study their performances in different situations and access possible combinations of these methods to reach a trade-off. In addition, different techniques will be examined and compared, and the probability of failure will be analyzed.

REFERENCES

[1] K. Solangi, M. Islam, R. Saidur, N. Rahim, and H. Fayaz, "A review on global solar energy policy," *Renewable and sustainable energy reviews*, vol. 15, no. 4, pp. 2149–2163, 2011.

[2] J. M. Aguilar-López, R. A. García, A. J. Sánchez, A. J. Gallego, and E. F. Camacho, "Mobile sensor for clouds shadow detection and direct normal irradiance estimation," *Solar Energy*, 2022.

[3] J. Jacob, A. Pandey, N. Abd Rahim, J. Selvaraj, M. Samykano, R. Saidur, and V. Tyagi, "Concentrated photovoltaic thermal (cpvt) systems: Recent advancements in clean energy applications, thermal management and storage," *Journal of Energy Storage*, p. 103369, 2021.

[4] E. Rakhshani, K. Rouzbehi, A. J. Sánchez, A. C. Tobar, and E. Pouresmaeil, "Integration of large scale pv-based generation into power systems: A survey," *Energies*, vol. 12, no. 8, p. 1425, 2019.

[5] E. F. Camacho, M. Berenguel, F. R. Rubio, and D. Martinez, *Control of solar energy systems*. Springer-Verlag, Jan 2012.

[6] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, "A review of process fault detection and diagnosis: Part i: Quantitative model-based methods," *Computers & Chemical Engineering*, vol. 27, no. 3, pp. 293 – 311, 2003.

[7] G. Faure, M. Vallée, C. Paulus, and T. Tran, "Fault detection and diagnosis for large solar thermal systems: A review of fault types and applicable methods," *Solar Energy*, vol. 197, pp. 472 – 484, 2020.

[8] E. Bernardi and E. J. Adam, "Fault-tolerant predictive control based on linear parameter varying scheme for industrial processes," *Journal of the Taiwan Institute of Chemical Engineers*, vol. 129, pp. 1–14, 2021.

[9] J. Marquez, A. Zafra-Cabeza, C. Bordons, and M. A. Ridao, "A fault detection and reconfiguration approach for mpc-based energy management in an experimental microgrid," *Control Engineering Practice*, vol. 107, p. 104695, 2021.

[10] X. Xie, B. Zang, D. Yue, and M. Chadli, "Robust fault estimation of discrete-time nonlinear plants via a comprehensive partition-based switching scheme," *International Journal of Robust and Nonlinear Control*, vol. 30, no. 16, pp. 6518–6534, 2020.

[11] F. Qu, J. Liu, X. Liu, and L. Jiang, "A multi-fault detection method with improved triplet loss based on hard sample mining," *IEEE Transactions on Sustainable Energy*, vol. 12, no. 1, pp. 127–137, 2021.

[12] M. Hussain, M. Dhimish, S. Titarenko, and P. Mather, "Artificial neural network based photovoltaic fault detection algorithm integrating two bi-directional input parameters," *Renewable Energy*, vol. 155, pp. 1272–1292, 2020.

[13] S. Fadhel, C. Delpha, D. Diallo, I. Bahri, A. Migan, M. Trabelsi, and M. F. Mimouni, "Pv shading fault detection and classification based on i-v curve using principal component analysis: Application to isolated pv system," *Solar Energy*, vol. 179, pp. 1 – 10, 2019.

[14] C. Correa-Jullian, J. M. Cardemil, E. López Droguett, and M. Behzad, "Assessment of deep learning techniques for prognosis of solar thermal systems," *Renewable Energy*, vol. 145, pp. 2178 – 2191, 2020.

[15] T. Zahra, L. M. Mourad, and A. H. Ahmed, "Robust fuzzy sliding mode observer for faults detection in solar power plant application." *Instrumentation, Mesures, Métrologies*, vol. 19, no. 4, 2020.

[16] S. Ruiz-Moreno, A. J. Sanchez, A. J. Gallego, and E. F. Camacho, "A deep learning-based strategy for fault detection and isolation in parabolic-trough collectors," *Renewable Energy*, vol. 186, pp. 691–703, 2022.

[17] A. F. Branco, M. M. Morato, G. Andrade, and J. E. Normey-Rico, "Tools for the control of modern solar-thermal heating plants," in *Proceedings of the 14th Brazilian Symposium of Inteligent Automation (SBAI), Ouro Preto, Brazil, Oct. 27-30, 2019, Vol. 1*. SBA, 2019, p. 104234.

[18] R. Carmona, "Analisis, modelado y control de un campo de colectores solares distribuidos con sistema de seguimiento en un eje," *Universidad de Sevilla, Seville*, 1985.

[19] E. F. Camacho, M. Berenguel, and F. R. Rubio, *Advanced control of solar plants*. Springer, Dec 1997.

[20] D. Y. Goswami, F. Kreith, and J. F. Kreider, *Principles of solar engineering*. CRC Press, 2000.

[21] A. J. Gallego Len, L. J. Yebra, E. Fernández Camacho, and A. J. Sánchez, "Mathematical modeling of the parabolic trough collector field of the tcp-100 research plant," 2018.

[22] A. J. Sánchez, A. J. Gallego, J. M. Escaño, and E. F. Camacho, "Temperature homogenization of a solar trough field for performance improvement," *Solar Energy*, vol. 165, pp. 1 – 9, 2018.

[23] S. J. Navas, F. R. Rubio, P. Ollero, and J. M. Lemos, "Optimal control applied to distributed solar collector fields with partial radiation," *Solar Energy*, vol. 159, pp. 811 – 819, 2018.

[24] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115–133, 1943.

[25] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. Mohamed, and H. Arshad, "State-of-the-art in artificial neural network applications: A survey," *Heliyon*, vol. 4, no. 11, p. e00938, 2018.

[26] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[27] T. P. Lillicrap, A. Santoro, L. Marris, C. J. Akerman, and G. Hinton, "Backpropagation and the brain," *Nature Reviews Neuroscience*, vol. 21, no. 6, pp. 335–346, 2020.

[28] M. F. Møller, "A scaled conjugate gradient algorithm for fast supervised learning," *Neural Networks*, vol. 6, no. 4, pp. 525–533, 1993.