Dynamic Model for the pH in a Raceway Reactor using Deep Learning techniques

Pablo Otálora[1], José Luis Guzmán[1], Manuel Berenguel[1], Francisco Gabriel Acién[2]

[1]Department of Informatics, University of Almería, Ctra. Sacramento s/n, ceiA3, CIESOL, 04120 Almería, España,
p.otalora@ual.es, joseluis.guzman@ual.es, beren@ual.es,
[2]Department of Chemical Engineering, University of Almería, Ctra. Sacramento s/n, ceiA3, CIESOL, 04120 Almería, España,
facien@ual.es

Abstract. This paper presents a black-box dynamic model for microalgae production in raceway reactors. The black-box model, developed using Deep Learning techniques, allows the estimation of the pH in a 100 m^2 raceway reactor. The model has been created using only and exclusively data, what gives a high ease of use. The results obtained verify the effectiveness of this type of techniques for the modelling of complex dynamic processes. The model was validated for different weather conditions obtaining satisfactory results. Thus, the obtained model is fairly useful for simulation purposes or for the implementation of model-based control techniques.

Keywords: deep learning, neural network, microalgae production, raceway reactor

1 Introduction

Microalgae production is a process with an increasing interest due to the high variety of its applications. Examples can be found in derived products for cosmetics, animal food or human nutrition. Moreover, the production process is useful for wastewater treatment, eliminating pollutants such as phosphorus or nitrogen, or to mitigate CO_2 emissions from other industrial facilities. Typically, microalgae cultivation can be accomplished in two different ways: in tubular photobioreactors and in open reactors or "raceways". The first ones take place in an environment where microalgae conditions are strongly controlled [4], while the second ones are carried out in large open ponds. This last type of photobioreactor, despite being susceptible to external contaminants and incapable of controlling their temperature, has the advantage of being less expensive and more easily scalable, making them the most commonly used at commercial scale. However, conventional raceway reactors are unable to maximize biomass production capacity mainly because of inadequate fluid-dynamic and mass transfer capacity. Thus, an optimization of the process is required to reduce costs as much as

possible and to increase the production, what is closely related to reach optimal values for pH, dissolved oxygen, temperature, light integration, and CO_2 injections [3]. For this purpose, the development of models and the implementation of advanced control strategies is essential.

In literature, a lot of effort has been made to develop nonlinear models to describe the dynamics of the microalgae system variables [3, 5, 6]. These models are extremely relevant, as they are a key element to optimize the system design and its operation mode. Most of the available nonlinear models are based on first principles balances, which are very useful tools for the process understanding. However, these models have a high complexity and are subject to parameter uncertainties, since many of the biological parameters are very complicated to be perfectly calibrated all the time.

In the last decades, because of the increasing computer capacity, Machine Learning, and more specifically, Deep Learning or Neural Networks, are becoming more relevant in the development of models for different fields [1]. These algorithms are capable of developing a model based solely on the data, without any physical meaning and without being explicitly programmed for it [9]. So, this paper deals with the development of a "black box" model for the pH of a raceway reactor making use of this type of techniques [17]. The core idea consists in obtaining a robust dynamic model that is easily updatable, based only on data, and well adapted to any circumstance that may take place in the system. As described above, the microalgae production process depends on solar irradiance and many other variables, such pH, dissolved oxygen, or medium temperature. Since the light requirements and temperature cannot be manipulated during normal operation, the pH and DO are the typical variables to be controlled and kept close to given optimal values. Among all the variables, the pH is the most important one in the process [13, 15, 16]. For that reason, the model presented in this paper is focused on the pH estimation based on the rest of the variables, which are assumed to be measured in the system.

2 Materials and Methods

2.1 Microorganism and culture medium

The microalgae strain modelled in this work was *Scenedemus almeriensis* (CCAP 276/24). This strain is resistant to temperatures up to 45°C and pH values up to 10, although the optimal values for its growth are 35°C and a pH of 8. The medium used in the experiments was Arnon, prepared by fertilizers instead of pure chemicals.

2.2 Raceway reactor

All the data used were taken from the raceway reactor located at the Research Centre "Las Palmerillas" (36º 48' N-2º 43' W), property of the Cajamar Foundation (Almería, Spain), in the year 2016 [5]. The reactor, as shown in Fig. 1,

is composed of two 50m long channels connected at their ends by 180º bends. It also has a $0.59m^3$ pit 1m away from the curve of one of the two channels, where air or CO_2 is injected through a diffuser in order to control the variables of interest (dissolved oxygen and pH). The liquid is propelled by a wheel with 8 blades of 1.2 m in diameter, driven by a speed-controllable electric motor. The reactor can be divided into 3 main parts depending on these elements: the paddle wheel, the pit and the channels. Each of these points has a different pH and dissolved oxygen value, which are measured separately.

Fig. 1. Real view of the raceway photobioreactor.

2.3 Variables of interest

In the performed tests, the following variables were measured with a sampling period of one minute:

- Dissolved oxygen, pH, and medium temperature.
- Medium level.
- Air, CO_2 and medium flow rates being injected.
- Solar radiation and ambient temperature.

Thus, all the previous variables have been used to develop the proposed black-box model in order to estimate the pH variable.

2.4 Deep Learning for dynamic modelling

Computational learning algorithms are able to "learn" from data to obtain models with different purposes without being explicitly programmed for it [1]. The use of one or another type of algorithms will depend on the problem and the available data.

Neural networks are within this set of algorithms [7]. These are intended to emulate the functioning of biological neurons through a structure formed by layers of nodes and connections between them, predominantly in parallel. Each node or neuron has a series of inputs and a dependent output whose relationship is expressed in the following equation[10]:

$$y = \phi\left(\sum_k W_k x_k + b\right) \tag{1}$$

where y symbolizes the output of the node, x_k the value of each input k of the node, W_k is the weight of each input k, b is the bias of the node, and ϕ is its activation function. Thus, if the values of W_k and b are known for all the nodes, as well as their activation function, it is possible to obtain the network output for any combination of inputs.

The objective of the learning algorithm is therefore to determine the W_k and b values for each neuron that minimize the difference between the predicted and actual process output. The process to solve this problem can be divided into two main stages: data processing and network training.

Data processing. In any computational learning algorithm, the quality of the resulting model is directly proportional to the quality of the data. The more data we have and the higher the quality, the better predictions we can expect. Likewise, if the data are erratic or insufficient, it will be impossible to obtain reasonable results no matter how much the network is trained. Thus, this stage is critically important, besides taking up most of the time of network development.

Usually, the raw data records available for any problem are not suitable to be directly assumed by the network, either due to sensor noise, wrong samples, or data gaps. Therefore, it is necessary to standardize the data and mitigate any irregularities in them. Some techniques used for this purpose are data interpolation, filtering or directly removing too poor sets. How this work has been done in this paper will be described later on.

Once the data have been treated, it will be separated into two sets: one set destined to train the network, which will cover around 70% of the total data, and a second set whose purpose is to test the network trained by the first set, in order to ensure that the model is not exclusively focused on memorizing the training data, but also has the ability to generalize to other different situations.

Network training. The training of the network will be carried out once the processed data are available. For this purpose, it is necessary to define a series of elements and parameters that will shape it. Firstly, it is essential to determine a proper network architecture, that is, the layers that will constitute the network. This is frequently a not deterministic process, as several iterations are necessary to compare between them and to select the appropriate layers. A higher number of layers will give us a more complex network that can better adapt to different behaviours, but it is also possible to produce overfitting, which is the over memorization of training data.

Besides the number of layers, it is necessary to determine their type, as well as their number of nodes. This depends on the kind of problem that is being faced: regression or classification, involvement of temporary component, the need to avoid overfitting etc. The architecture developed for this paper and its justification will be explained later on.

Furthermore, there are certain parameters that will affect the learning process, regardless of the network architecture. These are the number of epochs and the batch size. One epoch equals one network pass through the entire data set, while the batch size defines the number of samples the network passes through before its parameters are adjusted. Therefore, if there is a set of 100 samples and it is trained during 500 epochs with a batch size of 20, the network will recalculate its parameters 5 times per epoch and 2500 times overall. These parameters are fundamental not only for the time it takes to train the net, but also to improve learning and prevent overfitting. After the architecture and parameters of the network have been selected, the next step is the training stage. It is important to subsequently validate the network obtained using the test set in order to achieve the most accurate and robust prediction possible.

3 Results

3.1 Model Development

The proposed model is a black-box model, which is not intended to demonstrate or represent the physical interactions between the variables, but rather to obtain the correlation between each of the system variables and the pH at the desired point (at the end of the channel). Since it is a dynamic system, time plays a crucial role and the model must reflect this issue. For this purpose, a LSTM (Long Short Term Memory) layer was selected [14], which stores the 'network state" at each instant. So, the model output at a given time does not only depend on the inputs at that time, but also on the network state. The data processing, model development and model validation have been done in the MATLAB environment and using of the Deep Learning Toolbox [8, 10].

Due to the use of this type of layer, it is necessary to guarantee the continuity of the data. Therefore, the data processing stage begins by discarding the data sets of days in which a large amount of data is missing. If the data gaps last only a few moments, it is not necessary to discard the day, but if the gap is long enough to make interpolation illogical, the dataset must be deleted. In the case of small data gaps, the interpolation of each sample instant shall be made between the nearest previous and next instants whose measurement is correct.

To improve the network training and performance, the mean and standard deviation of each of the variables shall be calculated to normalize them according to the following expression:

$$X_n = \frac{X - \mu}{\sigma} \tag{2}$$

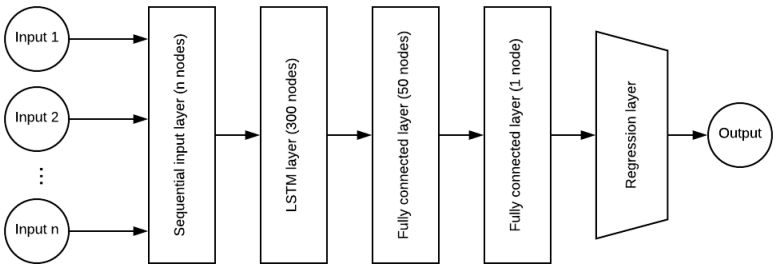

Fig. 2. Diagram of the selected network architecture.

where X_n symbolizes the already normalized variable, μ is the arithmetic mean, σ the standard deviation, and X the raw variable.

A data set of 105 days was originally used to develop the proposed model. After the data processing stage and analysis, 27 days were discarded because of errors and gaps in the measurements. Thus, a data set of 78 days was finally considered, where 60 of them were randomly selected to constitute the training set, and the remaining 18 for validation purposes. As previously mentioned, the core of the proposed network is the LSTM layer. Thus, the first layer set will be a sequential input layer, having as many nodes as input variables are used to make the predictions. The purpose of this layer is to serve as the data entry. The second layer will be the LSTM. A sequential output mode will be configured for it, as well as a number of nodes between 200 and 300, This parameter was selected after several tests with different values, picking the one that had the best performance. After that, a fully connected layer of 50 nodes will be established, which will work as an intermediate layer to provide more depth to the network. Following this, the use of a dropout layer is optional. The utility of this layer is to ignore a percentage of the data in each iteration. In normal circumstances, this is not positive for the network, but in case of overfitting, it is very helpful. Finally, another fully connected layer will be introduced with a single node for the output, and a last regression layer. In figure 2 a diagram of the selected architecture for this work is shown, where all the stages described above are summarized. The number of nodes in each layer has been selected after several iterations, in order to make it as low as possible to accelerate the training of the network, but large enough so that it doesn't deteriorate the prediction.

Regarding the learning parameters, different numbers of epochs have been tested. It has been demonstrated that a higher number of epochs leads to better results, and that overfitting does not take place with less than 3000 epochs. Since above a certain number of epochs the difference is not significant, the final value given to this parameter has been 2000 epochs. Besides, as there are 60 days available for training, the selected batch size has been 20, so that three

iterations are carried out in each period and no data is left out. The optimizer 'Adam' was used, with an initial learning ratio of 0.01.

3.2 Model Results

In this section, validation results of the proposed model are presented. The performance of the network has been evaluated for two different purposes: one-step prediction and multi-step prediction. In the first case, the network is used as a regression model, where the network state is updated with the real value of the pH variable at each instant time. Thus, only predictions for one step ahead are performed. On the other hand, in the multi-step prediction case, the network is used as an independent model, where the network state is updated by using the own model predictions. For both cases, the Root Median Square Error (RMSE) metric was used to analyze the goodness of fit of the model:

$$RMSE = \sqrt{\mu((Y_{real} - Y_{pred})^2)} \tag{3}$$

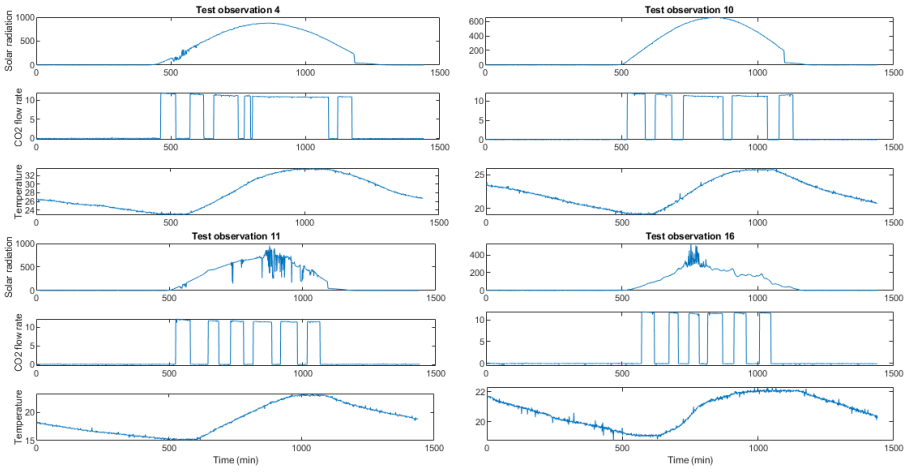

Fig. 3. Radiation and CO2 flow rate profiles for four representative days.

In this paper, four different days have been randomly selected to show the model results. Figure 3 shows some of the variables for these days, which are used as inputs for the proposed model. Notice that days with many different input profiles are considered for the validation process. First of all, the performance of the network will be checked by performing the one-step prediction. The test data of all the variables are available for each instant and the aim is to predict the pH value in the following instant. In this test, the results are really promising

with RMSE value of 0.1082. Figure 4 shows the obtained results for the data set presented in figure 3, where it is observed that the model behaves really well for all the data sets. This solution for one-step prediction can be very useful for fault detection techniques or real time estimation. However, for simulation or control purposes the prediction horizon of a single sample instant is insufficient.

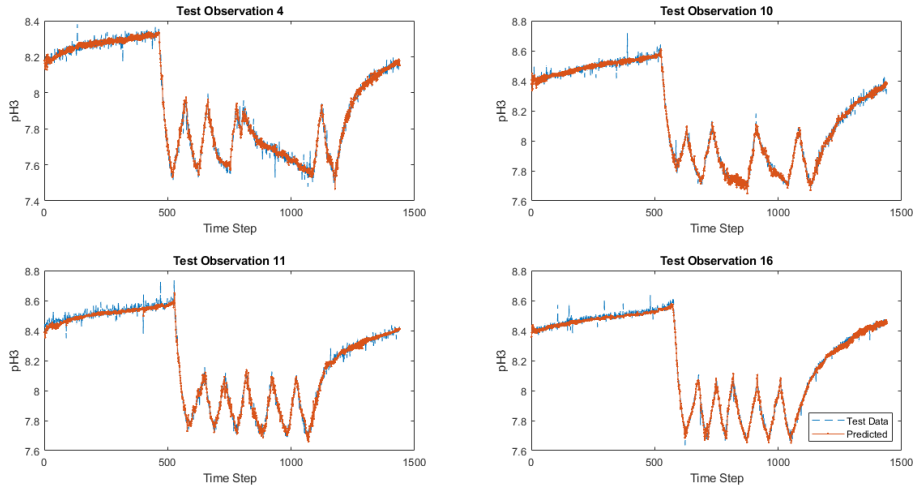

Fig. 4. Experimental and predicted data of pH for four representative days with a single-step prediction.

For this reason, the multi-step prediction was subsequently implemented. The basis of this test is the same as the previous one, with the difference that for the prediction of an instant $k+1$ the pH value at the instant k will not be the real one, but the one predicted at the previous instant. This makes it possible to predict the pH value indefinitely. However, there are some limitations to be considered. Notice that the resulting network has a certain error in the predictions such as observed in the one-step prediction case. This means that, since future predictions depend on values with a slight error, that error is fed back, and therefore increases proportionally to the number of sample instants that are intended to be known.

Therefore, the prediction was made for a whole day, which corresponds to 1440 samples. Despite the aforementioned, the results obtained are highly satisfactory, as can be seen in the figure 5. As expected, the prediction error is increased, reaching a value of 0.3720, but even so the network is able to predict a complete day with considerable accuracy, especially in the instants when the control was taking place during the real experiments. This performance is mainly related to the use of dissolved oxygen as an input to the network, since this variable is highly correlated to the pH. Facing future works, the aim is to

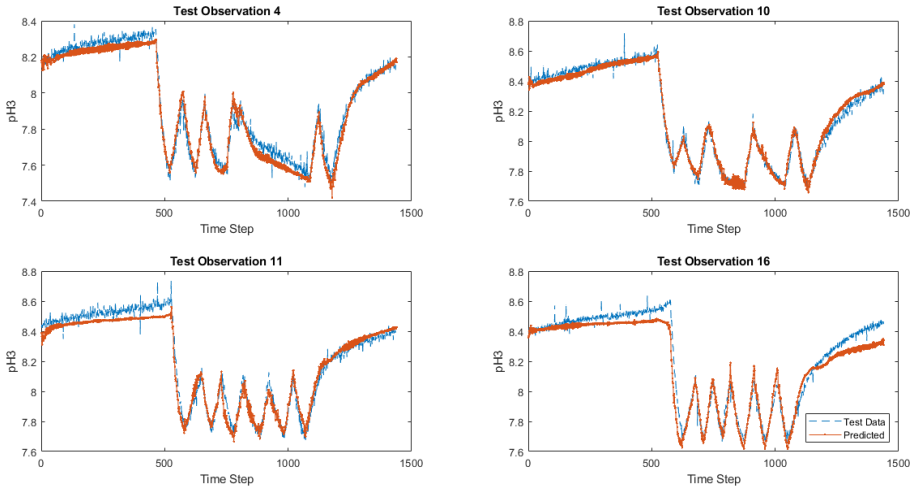

Fig. 5. Experimental and predicted data of pH for four representative days with a multi-step prediction.

develop a network with a similar structure capable of predicting both variables, in order to achieve a complete simulation of the system. Notice that even in time lapses when a small offset appear, the dynamics of the system are well represented during all the daily operation.

4 Conclusions

In this paper, a dynamic model based on neural networks has been developed and validated for a raceway photobioreactor. The model allows the prediction of the pH value in the channels with a time horizon up to one day based only on measurable data inputs. The methodology followed in the article grants the possibility of obtaining similar models for the prediction of other variables of interest, such as dissolved oxygen or biomass concentration. This type of models opens a wide variety of possibilities for future works in the field of photobioreactor modelling and control, providing highly reliable non-linear models that are easily updateable and whose calibration is fully automated, depending only on measurable data.

5 Acknowledgements

This work has been partially funded by the following projects: DPI2017 84259-C2-1-R (financed by the Spanish Ministry of Science and Innovation and EU-ERDF funds) and the European Union's Horizon 2020 Research and Innovation Program under Grant Agreement No. 727874 SABANA.

References

1. M. Amini and S. Chang. A review of machine learning approaches for high dimensional process monitoring. In *IISE Annual Conference and Expo 2018*, pages 390–395, Orlando, USA, 2018.
2. E. A. del Rio-Chanona, J. L. Wagner, H. Ali, F. Fiorelli, D. Zhang, and K. Hellgardt. Deep learning-based surrogate modeling and optimization for microalgal biofuel production and photobioreactor design. *AIChE Journal*, 65(3):915–923, 2019.
3. I. Fernández, F. G. Acién, M. Berenguel, and J. L. Guzmán. First principles model of a tubular photobioreactor for microalgal production. *Industrial and Engineering Chemistry Research*, 53(27):11121–11136, 2014.
4. I. Fernández, F. G. Acién, M. Berenguel, J. L. Guzmán, G. A. Andrade, and D. J. Pagano. A lumped parameter chemical-physical model for tubular photobioreactors. *Chemical Engineering Science*, 112:116–129, 2014.
5. I. Fernández, F. G. Acién, J. L. Guzmán, M. Berenguel, and J. L. Mendoza. Dynamic model of an industrial raceway reactor for microalgae production. *Algal Research*, 17:67–78, 2016.
6. F. García-Mañas, J. L. Guzmán, M. Berenguel, and F. G. Acién. Biomass estimation of an industrial raceway photobioreactor using an extended Kalman filter and a dynamic model for microalgae production. *Algal Research*, 37(June 2018):103–114, 2019.
7. A. Gupta. Introduction to deep learning: Part 1. *Chemical Engineering Progress*, 114(6):22–29, 2018.
8. M. Hudson, B. Martin, T. Hagan, and H. B. Demuth. Deep Learning Toolbox$^{\text{TM}}$ User's Guide. 1992.
9. B. S. Kim, B. G. Kang, S. H. Choi, and T. G. Kim. Data modeling versus simulation modeling in the big data era: Case study of a greenhouse control system. *Simulation*, 93(7):579–594, 2017.
10. P. Kim. *MATLAB Deep Learning*. 2017.
11. Mathworks. Deep learning in MATLAB. *The MathWorks, Inc.*, pages https://www.mathworks.com/help/nnet/deep–learning–, 2018.
12. Mathworks. Practical Deep Learning Examples with MATLAB. page 33, 2018.
13. A. Pawlowski, J. L. Guzmán, M. Berenguel, and F. G. Acién. Control system for pH in raceway photobioreactors based on wiener models. *IFAC-PapersOnLine*, 52(1):928–933, 2019.
14. S. S. Pon Kumar, A. Tulsyan, B. Gopaluni, and P. Loewen. A Deep Learning Architecture for Predictive Control. *IFAC-PapersOnLine*, 51(18):512–517, 2018.
15. E. Posadas, M. d. M. Morales, C. Gomez, F. G. Acién, and R. Muñoz. Influence of pH and CO2 source on the performance of microalgae-based secondary domestic wastewater treatment in outdoors pilot raceways. *Chemical Engineering Journal*, 265:239–248, 2015.
16. Z. Wu, Y. Zhu, W. Huang, C. Zhang, T. Li, Y. Zhang, and A. Li. Evaluation of flocculation induced by pH increase for harvesting microalgae and reuse of flocculated medium. *Bioresource Technology*, 110:496–502, 2012.
17. S. Zhang and O. R. Zaiane. Comparing Deep Reinforcement Learning and Evolutionary Methods in Continuous Control. In *NIPS 2017 Deep Reinforcement Learning Symposium*, Long Beach, USA, 2017.