International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 10 Issue: 03 | Mar 2023 www.irjet.net p-ISSN: 2395-0072

Development of an Explainable Model for a Gluconic Acid Bioreactor and Profit Maximization through Grey Wolf Optimizer Trained Artificial Neural Network Technique

Sucharita Pal[1], Sandip Kumar Lahiri[1*]

[1]Chemical Engineering Department, National Institute of Technology Durgapur, Durgapur – 713209

Abstract - *The current study focuses on building a model of a laboratory scale bioreactor using a grey wolf optimizer trained ANN approach, then optimizing it for profit maximization. The glucose to gluconic acid bioprocess was employed as a case study. The fermenter is a multiphase enzymatic bioreactor and developing a viable first principle-based model is difficult due to its complexity; on the other hand, data-driven models lack explanability. As a result, in this study, a general methodology was developed in which a data-driven technique such as the grey wolf optimizer trained ANN technique was used as a modelling tool, and the model was then post-processed to increase the model's explainability. The model was chosen for its ability to accurately describe the underlying physics of the system. It was subjected to optimization after the establishment of an acceptable model. The goal of this study was to maximize gluconic acid yield, which has a substantial impact on the process' profitability. Using an evolutionary algorithm on the created model, an ideal solution was determined.*

Key Words: Grey wolf optimizer trained ANN technique; Bioprocess; Modelling; Optimization; Genetic algorithm

1.INTRODUCTION

The business environment has altered drastically in the previous decade as a result of severe worldwide competition. Because of globalisation, chemical process industries are seeing their profit margins erode as a result of more harsh competition in a volatile market. The only way to properly mitigate these challenges is to improve process efficiency through yield maximisation through technological innovation. Chemical companies all around the world are seeking for new and innovative ways to save costs and boost revenues. The application of artificial intelligence-based techniques to extract value from huge amounts of experimental data through data mining and knowledge discovery is one of the most intriguing creative tactics to investigate.

Chemical bioreactors are catching the interest of scientists who are looking for new methods to make money. The reactor is the only major piece of machinery that adds real value to raw materials by turning them into finished goods. In this aspect, reactor optimization has a tremendous potential impact on overall profitability [1]. As a first step in optimizing the bioreactor, modelling complex systems of chemical processes is critical. In biochemical reactions, complex reaction kinetics and thermodynamics were involved. Building a credible phenomenological model for bioreactors is a time-consuming and difficult task that requires a detailed grasp of heterogeneous catalytic behaviour such as mass diffusion and catalyst deactivation, among other things. Furthermore, in laboratories, various parameters such as agitation speed, temperature, negative influence of toxins contained in incoming gas, diffusional coefficients, and others affect reaction rate, yield, and selectivity, the processes of which are unknown. As a result, the reactor's optimization is hampered by a lack of understanding of chemical reaction dynamics. For safety and dependability concerns, most chemical bioreactors remain a black box, and scientists do not tamper with them. This causes bioreactors to operate inefficiently, which has a considerable influence on the profitability. As a result, in the chemical sector, reactors are viewed as an unknown territory. A minor increase in catalyst selectivity and reaction yield, on the other hand, has a major impact on raw material consumption and overall profitability in large-scale operations. [1].

To derive the kinetic equation of a given biochemical reaction, laboratory scale kinetic analysis is typically performed in a very ideal and controlled context. The applicability of such an approach for kinetic modelling of bioreactors is questionable due to the presence of poisons and inert in input gases, and the varying heat and mass transfer environment. Furthermore, chemical engineers are unwilling to commit adequate time and effort to a full examination of these complex reaction mechanisms due to the restricted market window for chemical goods. As a result, an alternative straightforward method is to develop approximated reactor models for these complicated reaction systems using a data-driven effective computational strategy, which can then be used to optimize the reactor and increase profit.

The key challenge is figuring out how to leverage this plethora of data to produce more money, because most chemical laboratories collect and store vast volumes of reactor input and output operational data. Data is the new oil, and data-driven modelling tools such as artificial neural networks (ANN) and support vector machines (SVM) are the

new IC engines of our day. In the last decade, ANN and SVR have become very popular black box modelling methodologies, with a number of applications developed for biochemical reactors. Engineers, on the other hand, reject ANN and SVM models because they are difficult to understand and provide a black box model. The ANN model offers no insight into the underlying physics of chemical reactors.

Despite its high prediction capacity, the model suffers from explainability limits because it produces a black box type equation consisting of a complex sigmoidal function with several tuning elements known as weights and biases. To acquire better understanding and profit, process engineers prefer intelligible equations in differential/ algebraic form that relate output variables to input features. A closed form equation that can describe the effect of key process parameters on the output variable is preferred in the created model. SVM and ANN both have the difficulty of providing closed form explainable equations that are portable and simple to implement in a DCS system. Gaining insights and obtaining a closed form explainable model equation is crucial for engineers to accept the model's use in real life

Despite the ANN's outstanding prediction skills, there are few uses of this technology in chemical reaction engineering work in biochemical reactors. An attempt was made in this work to use ANN modelling on a gluconic acid bioreactor. One of the main objectives of this study is to convert the ANN model into an explainable closed form equation, which will give crucial information about the reactor's phenomenology.

The second purpose of this project is to train the ANN model using freshly found nature-inspired grey wolf optimization methodologies. Learning is a vital part of every neural network, and it has piqued the interest of many researchers. In most applications, the traditional [7] or upgraded [8–10] Back-Propagation (BP) algorithm is employed to train feedforward neural networks (FNNs). The BP method is a gradient-based algorithm with some flaws, including delayed convergence [11] and a proclivity to stay stuck in local minima [12]. During the learning process of FNNs, the goal is to identify the best combination of connection weights and biases to achieve the least amount of error. FNNs, on the other hand, frequently converge on points that are the best answer locally but not worldwide. In other words, rather than the global minimum, learning methods lead FNNs to local minima. According to [13], the BP algorithm's convergence is strongly dependent on the weights, biases, and parameters' initial values. Learning rate and momentum are two of these characteristics. A popular approach in the literature is to enhance the problems of BP-based learning algorithms by using unique heuristic optimization methods or evolutionary algorithms.

Some of the heuristic optimization approaches used to train FNNs include Simulated Annealing (SA) [13,14], Genetic Algorithms (GAs) [15], Particle Swarm Optimization (PSO) algorithms [16–20], Magnetic Optimization Algorithm (MOA) [21], and Differential Evolution (DE) [22]. Some algorithms, such as SA and GA, can reduce the chance of local minima trapping, but they still have poor convergence rates, according to [11].

Despite the widespread usage of meta-heuristics in ANN learning, none of them has done well in all applications. Existing metaheuristics also have a number of flaws [23-27], such as slow convergence speed, trapping in local minima, long computational time, tuning many parameters, and a difficult encoding scheme. As a result, it appears that boosting the efficiency of ANN learning in various domains requires either improving the performance of existing meta-heuristics or proposing new ones. [18] presented GWO, a new stochastic and metaheuristic optimization technique. The efficacy of the GWO approach for training FNNs is studied in this work.

Because of its financial importance in global markets, the current study chose to model a gluconic acid bioreactor. Gluconic acid is used as the metal supplement of calcium, iron, etc. in pharmaceuticals and as an acidulent in the food industry. It also finds applications as a biodegradable chelating agent, filler, metal cleaner, dye stabilizer, and in the textile industry for removing instructions. A reliable first principle-based model is rarely accessible in the literature due to a lack of understanding and complexity of multiphase enzymatic reactions occurring in gluconic acid bioreactors. Data driven modelling is a potential alternative strategy due to the vast amount of bioreactor operating parameter data available after multiple runs. The current project aims to make use of a huge amount of process data to create a framework for converting the data's information into profit.

The study's next goal is to use the developed model to increase the gluconic acid factory's profit. This is performed by optimizing the input process parameters utilising a model-based, nature-inspired metaheuristic optimization method in order to enhance gluconic acid yield (i.e., reactor performance). The GA is used to improve the input space of the bioreactor's ANN model in order to give pareto optimum solutions that achieve the objective in the most efficient manner possible.

2. Case study of gluconic acid bioreactor

2.1. Background

Because of the lack of understanding and complexity of multiphase catalytic processes in lab scale batch reactors, trustworthy first principle-based models are hard to come by. Data driven modelling is a feasible alternative strategy due to the vast amount of reactor operating parameter data available from bioreactors. The current project aims to make use of a huge amount of process data to create a framework for converting the data's information into profit.

2.2. Reactions:

Commercially, gluconic acid is produced primarily using two biological methods, but more expensive chemical and electrochemical ways are also available. The most prevalent biochemical approaches are freecell fermentation and immobilised enzyme-based glucose bioconversion (glucose oxidase, GOD, of Aspergillus niger and Gluconobacter). The GOD converts glucose to glucono-d-lactone, which is then hydrolyzed by lactonase to gluconic acid. Producing gluconic acid with immobilised enzymes is a costly and time-consuming technique due to obstacles in the immobilisation and separation phases; extra difficulties develop as a result of denaturization of the enzymes. During free-cell fermentation, mycelia are exposed to a variety of mass and heat-transfer stresses. Mechanical agitation aids in the removal of these restrictions, but it creates a turbulent flow that can result in cell disintegration, cell fracture or surface erosion, and pellet breaking. As a result, there may be a sudden or gradual decline in cellular activity. On the other hand, fermentation of gluconic acid by cells immobilised on a support matrix under submerged conditions is a cost-effective and efficient method.

Gluconic Acid Yield= (Moles of gluconic acid produced)/(Moles of glucose consumed) (1)

2.3. Process flow diagram

A new batch fermentation procedure for producing gluconic acid from glucose has recently been devised, with A. niger immobilised on a cellulosic fabric support matrix generating higher yields. The enhanced overall productivity of this technique is mostly due to the increased interaction between dissolved oxygen and the fungal mycelia. To optimise the aforementioned reaction, a continuous substrate dripping mechanism (see Figure 1) is used instead of the mechanical agitation used in free-cell fermentation. As a result, the bioreactor's yield determines the total profitability.

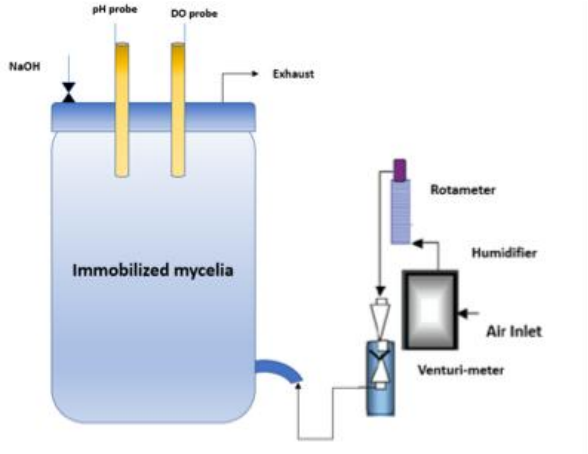

FIGURE 1 Experimental setup of the bioprocess taken from [33]

2.4. Production Objectives

The purpose of this research is to develop a mathematical model of the new glucose to gluconic acid batch fermentation process and to discover the best process conditions for higher gluconic acid yield. The fermenter model was built using experimental data that took into account the impacts of substrate (glucose), biomass, and dissolved oxygen levels. Figure 1 depicts the complex reaction and mass transfer pathways involved in the glucose to gluconic acid bioconversion using A. niger immobilised on cellulose microfibrils. Because the physicochemical events that drive bioconversion, as well as the kinetic and transport mechanisms that follow them, are poorly understood.

3. Background materials

This part discusses the fundamental background materials required for a complete understanding of the proposed method. The Multi-layer perceptron (MLP) as a Feed-forward neural network (FNN) is introduced first, followed by GWO approaches, which are then compared to the suggested technique for MLP learning in bioreactors modelling. MLP network is followed by a brief discussion of the hybrid ANN and GWO training method.

The feed-forward neural network (FFNN) is one of the most common ANNs and receives a lot of academic attention because of its ability to map any function to an unlimited degree of precision. The multi-layer perceptron has been employed in a variety of sectors, including finance, medicine, engineering, geology, physics, and biology. Nonlinear process modelling, fault diagnostics, and process control are all common applications in the field of chemical engineering.

As shown in Figure 2, MLP contains one input layer that receives external inputs, one or more hidden layers, and one output layer that displays the results. All levels, with the exception of the input layer, are made up of processing nodes and activation functions. The input layer provides data, and the network nodes perform calculations in successive layers until each output node receives an output value.

3.1. Training of ANN by GWO

There are three techniques to train FNNs with a heuristic algorithm in general. The ideal mix of weights and biases for a FNN with the least amount of error is first discovered using heuristic techniques. Second, in order to find the appropriate structure for a FNN in a specific situation, heuristic approaches are applied. Finally, an evolutionary algorithm can be used to tune the parameters of a gradient-based learning system, such as the learning rate and momentum.

In the first scenario, the structure is fixed before training FNNs. The goal of a training method is to find a good value for all connection weights and biases in order to lower the FNNs' total error. In the second instance, the FNN structures are different. A training technique is applied to a FNN to determine the right structure for a given problem. Change the structure of the FNN by manipulating the connections between neurons, the number of hidden layers, and the number of hidden nodes in each layer.

The first technique in this study is used to apply GWO to a FNN; these operations are referred to as hybrid FNN-GWO. The FNN's structure is fixed, and the GWO method chooses a set of weights and biases that gives the FNN the least amount of error. In order to design FNN-GWO, the following fundamental features must be defined. In FNN-GWO, build a fitness function based on the FNN's mistake before evaluating agents' fitness. Second, an encoding strategy for the FNN-GWO agents should be devised to encode the FNN's weights and biases. These elements are described in further depth below.

3.1.1. Fitness function

The fitness function used in this article [10] has the following formula:

In Figure 2, which has two layers, the number of input nodes is equal to n, the number of hidden nodes is equal to h, and the number of output nodes is equal to m. (one input, one hidden, and one output layer). At the end of each learning period, the output of each hidden node is calculated as follows:

$$f(s_j) = 1/\left(1 + \exp\left(-\left(\sum_{i=1}^{n} w_{ij}.x_i - \theta_j\right)\right)\right), \text{j} = 1, 2, \ldots, \text{h} \quad (2)$$

In $\sum_{i=1}^{n} w_{ij}.x_i - \theta_j$, n is the number of the input nodes, w_{ij} is the connection weight from the i^{th} node in the input layer to the j^{th} node in the hidden layer, θ_j is the bias (threshold) of the j^{th} hidden node, and x_i is the i^{th} input.

After calculating outputs of the hidden nodes, the final output can be defined as follows:

$$o_k = \sum_{i=1}^{n} w_{kj}.f(s_j) - \theta_k, \quad k = 1,2, \ldots, m, \quad (3)$$

where w_{kj} is the connection weight from the j^{th} hidden node to the kth output node and θ_k is the bias (threshold) of the k^{th} output node.

Finally, the learning error E (fitness function) is calculated as follows:

$$E_k = \sum_{i=1}^{m} \left(o_i^k - d_i^k\right)^2 \quad (4)$$

$$E = \sum_{k=1}^{q} \frac{E_k}{q} \quad (5)$$

where q is the number of training samples, d_i^k is the desired output of the i^{th} input unit when the k^{th} training sample is used, and o_i^k is the actual output of the i^{th} input unit when the kth training sample is used. Therefore, the fitness function of the i^{th} training sample can be defined as follows:

$$\text{Fitness } (X_i) = E(X_i) \quad (6)$$

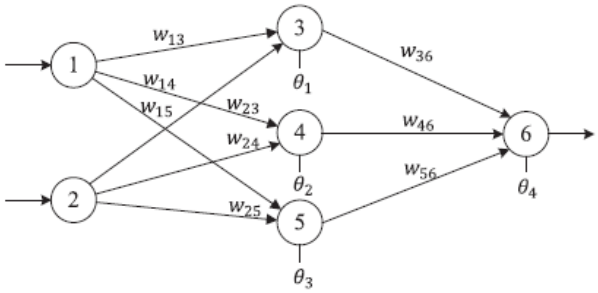

FIGURE 2: FNN with a 2-3-1 structure

3.1.2 Encoding strategy

Following the specification of the FNN-GWO fitness function, the next step is to select an encoding technique for each FNN-GWO agent to encode the FNN's weights and biases. There are three approaches to encode and express the weights and biases of FNNs for each agent in evolutionary algorithms, according to [10]. These are the methods of vector, matrix, and binary encoding. Each agent is encoded as a vector in vector encoding. During training, each agent reflects all of the FNN's weights and biases. Each agent is encoded as a matrix in matrix encoding. Because we're interested in training FNNs, we've employed the matrix encoding technique in this piece. An example of this encoding strategy for the FNN of Figure 2 is provided as follows:

$$\text{particle}(:,:,\text{i}) = [W_1, B_1, W_2', B_2], \quad (7)$$

$$W_1 = \begin{bmatrix} w_{13} & w_{23} \\ w_{14} & w_{24} \\ w_{15} & w_{25} \end{bmatrix}, B_1 = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}, W_2' = \begin{bmatrix} w_{36} \\ w_{46} \\ w_{56} \end{bmatrix}, B_2 = [\theta_4] \quad (8)$$

where W_1 is the hidden layer weight matrix, B_1 is the hidden layer bias matrix, W_2 is the output layer weight matrix, W_2' is the transpose of W_2, and B_2 is the hidden layer bias matrix.

3.2. Grey Wolf Optimization (GWO): at a glance

GWO, a new stochastic and metaheuristic optimization technique, was introduced by [28]. GWO's purpose is to mimic the cooperative hunting behaviour of grey wolves in the wild. The grey wolf pack's hierarchical organisation and predation behaviour are promoted by this bionic optimization approach, in which the wolves take prey by surrounding, haunting, and attacking it under the command of the top grey wolf [31]. The top three grey wolves were the target of this large-scale search methodology, but there was no way to eliminate them. The optimization technique differs from others in terms of modelling. As shown in Figure 3, it produces a strict hierarchical pyramid.

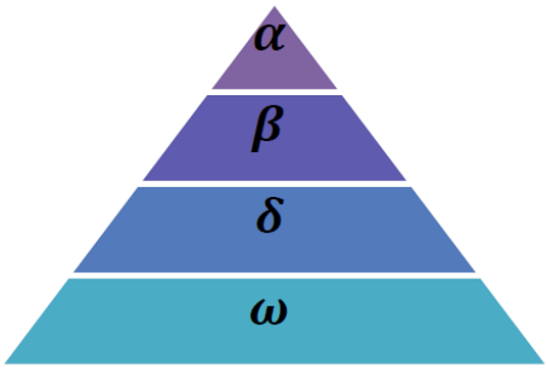

FIGURE 3: Hierarchy of grey wolf (dominance decreases from top down)

The typical size of a group is 5-12 persons. The layer, which consists of a male and female leader, is the most powerful and capable member of the team when it comes to making decisions on predation and other activities. The second and third layers of the hierarchy, respectively, are in charge of aiding group organisations in their behaviour. The bottom of the pyramid, often known as, is home to the majority of the world's population. They are largely responsible for satisfying the entire pack by maintaining the dominance hierarchy, regulating the population's internal connections, and caring for the young [28]. GWO mathematical modelling:

The main key point of the GWO model is the social hierarchy, encircling, hunting, attacking and searching prey.

A. Social hierarchy

In the model of GWO, α is considered as fittest solution. β and δ are considered as second and third best solution, respectively. The rest of the solutions are assumed to be ω.

B. Encircling

At first, the location of the prey is determined and during the hunting process grey wolves encircled the prey. The following equations are proposed for mathematically modelling.

$$\overrightarrow{D_p} = \left|\vec{C}.\overrightarrow{X_p}(t) - \vec{X}(t)\right| \quad (9)$$

$$\vec{X}(t+1) = \overrightarrow{X_p}(t) - \vec{A}.\left(\overrightarrow{D_p}\right) \quad (10)$$

Where t is the number of current iterations, $\vec{X}(t)$ is the position vector of one grey wolf, $\vec{X}(t+1)$ is the next position vector it arrives, $\overrightarrow{X_p}(t)$ is the position vector of the prey, $\vec{A}$ and $\vec{C}$ are coefficient vectors which are evaluated as follows:

$$\vec{A} = 2\vec{a}.\overrightarrow{r_1} - \vec{a} \quad (11)$$

$$\vec{C} = 2.\overrightarrow{r_2} \quad (12)$$

r_1 and r_2 are random vectors in [0,1], a is decreasing value during the iteration in [0,2], typically $a = 2 - 2t/I$ (I is the maximum number of iterations).

In this concept, grey wolves move around the best solution in hyper-cubes within an n dimensional space and able to detect the position of the prey and encircle it.

C. Hunting

Grey wolves have the ability to hunt prey with the guidance of alpha after encircling the prey. The beta and delta also take part in the hunting procedure on occasion. The first three best solutions in the mathematical stimulation of hunting behaviour update the position of other search agents (including the omegas). In this approach, the following equations are proposed.

$$\overrightarrow{D_\alpha} = \left|\overrightarrow{C_1}.\overrightarrow{X_\alpha} - \vec{X}(t)\right|;\ \overrightarrow{D_\beta} = \left|\overrightarrow{C_2}.\overrightarrow{X_\beta} - \vec{X}(t)\right|;\ \overrightarrow{D_\gamma} = \left|\overrightarrow{C_3}.\overrightarrow{X_\gamma} - \vec{X}(t)\right| \quad (13)$$

$$\overrightarrow{X_1} = \overrightarrow{X_\alpha}(t) - A_1\left(\overrightarrow{D_\alpha}\right);\ \overrightarrow{X_2} = \overrightarrow{X_\beta}(t) - A_2.\left(\overrightarrow{D_\beta}\right);\ \overrightarrow{X_3} = \overrightarrow{X_\gamma}(t) - A_3.\left(\overrightarrow{D_\gamma}\right) \quad (14)$$

$$\overrightarrow{X_p}(t+1) = \frac{X_1 + X_2 + X_3}{3} \quad (15)$$

D. Attacking prey (exploitation)

Grey wolves complete the hunting phase and ready to capture the prey. For the purpose of mathematically modelling, the value of $\vec{a}$ gradually decrease. Therefore, the fluctuation rate of $\vec{A}$ is also decreased by $\vec{a}$. $\vec{A} \epsilon [-2a, 2a]$ where a is decremented from 2 to 0 over the course of iterations. When the random values of $\vec{A} \epsilon [-1,1]$, the next position of search agent can be in any position between its current position and the position of the prey. When $|A| \geq 1$, the grey wolves would diverge from the prey to achieve global search and when $|A| \leq 1$, the grey wolves would converge towards the prey and complete it.

E. Searching prey (exploration)

GWO algorithm uses an efficient exploration methodology by allowing its search agents to update their position on the basis of alpha, beta, delta and attacks towards the prey. This mechanism creates a good diversity in the problem search space. Grey wolves are stay away from each other for global search of prey and close to each other for attacking the prey. For $\vec{A}$, values between 1 and -1 are taken. $\vec{C}$ vector is also favoured the exploration technique and random values $[0,2]$ are used. After generating the random population, alpha, beta and delta determined the position of the best prey. For selection of exploration and exploitation, the value of $\vec{a}$ is decreased from 2 to 0 respectively.

The GWO algorithm terminates when the criterion is satisfied. This metaheuristic approach is applied in various real-world problems because of its efficient and simple performance ability by tuning the fewest operators [28-32].

3.3. Hybrid learning of GWO and FNN network

In this section, GWO is used to learn FNN. A hybrid learning of GWO and FNN networks (GWO-FNN) is utilised to increase the network's accuracy. The algorithm will simultaneously determine the set of weights and their related accuracy by training the network. The FNN network's network weights and biases can be expressed as a D-dimensional vector. The vector for FNN is defined by Equation 16. To optimize the FNN weights using GWO methods, each particle's dimension is regarded as a vector D.

D = (Input × Hidden) + (Hidden × Output) +Hidden bias + Output bias, (16)

where Input, Hidden and Output are referred the number of inputs, hidden and output neurons of FNN network respectively. The number of biases in the hidden and output layers is also known as Hidden bias and Output bias. A dataset is collected, normalized, and read to begin the GWO-FNN. Following that, the appropriate number of inputs, output, and hidden neurons are specified to establish the particle dimension as Equation (5). The population is initialized, and the training error is determined as a fitness function following FNN training. Every particle (wolf) modifies its velocity and position based on training error. The new places represent the FNN network's new weights, which are supposed to minimize the fitness function. The fitness function is computed based on test set error. These steps will go on until meeting stop conditions.

4. MATHEMATICAL MODELLING OF BIOPROCESS

4.1. Selection of input and output variables for modelling

Because reactor yield has such a large impact on overall profitability, it is kept as an output variable. All reactor operational factors that could affect yield are stored as a "wish list" of input variables. Initially, all bioreactor experimental data was acquired. Following that, all of the input factors that could affect the output variable were recorded after speaking with a technical specialist. After that, a cross-correlation analysis was carried out. This method was used to determine the correlation coefficients of each input variable with the output variable, as well as the inter input cross-correlation coefficients.

The following criteria are used to shortlist the input variables.

(i) For a particular input variable, there should be high cross-correlation coefficient with output variable.

(ii) The values of cross-correlation coefficients of inter input variables should be low.

(iii) The input set of variables were kept as minimum as possible to avoid complexity of the model.

Based on the above criteria 3 input variables are finally shortlisted and tabulated in Table 1.

TABLE 1 Input Output variables for model building and their range

Variables used in modeling	Data Range
Input Variables	
Glucose concentration, g/L (x_1)	100.0–180.0
Biomass concentration, g/L (x_2)	1.00–3.00
Dissolved oxygen concentration, mg/L (x_3)	10.0–60.0
Output Variables	
Gluconic Acid Yield, % (Y_1)	5.9–94.58

4.2. Data collection, Data cleaning and removal of outliers

The GWO-FNN based model for the glucose to gluconic acid bioprocess was created using experimental input-output data from the fermenter. In this study, the gluconic acid-producing strain Aspergillus niger NCIM 545 was employed. The spore germination media, growth medium, and cellulosic fibre support have all been well reported previously.

The quality of data used to build a data-driven model is widely accepted as determining the model's quality. Because noisy and erroneous data can have a major impact on model performance, data quality is an important consideration when employing data driven modelling. Due to the high number of process data, this research developed an automated data cleaning technique that eliminates the need for human intervention. In this work, the data was pre-processed using multivariate Principal Component Analysis (PCA). An automated MATLAB-based system was built to build a multivariate statistical vector called t-squared from the operating dataset. The corresponding rows in the t-squared vector with values over the 95th percentile was therefore considered outliers and were eliminated from the dataset.

4.3. Modelling through FNN-GWO algorithm

An ANN-based model was built using the cleansed data. The dataset, which included eight input variables and two output variables (Table 1), was divided into a training set (80% of the total data) and a test set (the remaining 20%). (20 percent of the total data). The training data set was used to build the model by maximising the fitness value, and the test data set was used to cross-validate the result. Cross-main validation's purpose is to improve the model's generalizability.

The FNN-GWO-based model was developed using MATLAB 2019a code. The mean-squared error (MSE) between actual and projected outputs was employed as the fitness function in this study, and the programme was run in such a way that the MSE value was minimized. Due to the stochastic nature of the ANN, the software was run 100 times to create the model.

4.4. Optimization through Genetic Algorithm

The model is then utilized to optimize the bioreactor process parameters once a trustworthy and accurate bioreactor model has been built. The goal is to find the best process conditions for maximal bioreactor profitability. In other words, the optimization algorithm should aim for the highest possible reactor yield.

A multi-objective genetic algorithm is utilised in this study to establish a balance between two opposing goals. The genetic algorithm (GA) has shown to be a powerful optimization tool that has been applied to a wide range of technical and medical applications.

For implementation of GA algorithm, an objective function was developed which is as follows (Equation 17):

$$F_1(x) = 1/Y_1(x) \quad (17)$$

where $Y_1(x)$ is the function of the model corresponding to gluconic acid yield. Therefore, in GA $F_1(x)$ has to be minimized in order to maximize the yield of the bioprocess.

5. Results and discussions

5.1. Performance of FNN-GWO model

The major purpose of this study is to create an accurate, simple, portable, and easy-to-understand closed model equation for a bioreactor.

The values of ANN parameters required for modelling were determined using a trial-and-error approach and a literature review. In the current work, the number of nodes in the hidden layer fluctuates consistently from 5 to 25, and the FNN GWO approach is used to find the appropriate weights and bias that produce the lowest MSE between actual and anticipated output each time.

Shortlisting the models: To choose a valid model from a pool of likely candidates or representative model equations with varying degrees of complexity and accuracy, the following criteria were used:

(i) Simplicity: The model should be as straightforward as possible. Model complexity was determined by the number of nodes in the hidden layer.
(ii) Prediction accuracy: The gap between expected and actual yield.
(iii) The fundamental physics of the process should be captured in the model equation. To put it another way, model equations should include a physical understanding of the system under investigation rather than just a prediction relationship. This is an important consideration for building realistic reactor models.

TABLE 2 Rules to select best model from experimental observations.

Sl. no	Parameters changed keeping all other parameters constant	What happen to glucon yield?
1	If glucose concentration increase	Increases
2	If biomass concentration increase	Slightly decreases
3	If dissolved oxygen (DO) concentration increase	Increases

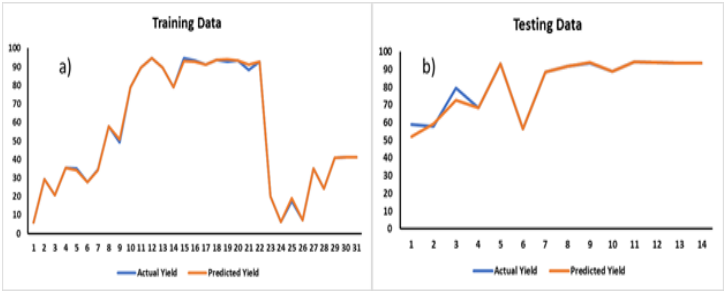

FIGURE 4: Actual vs. predicted plots of (a) gluconic acid yield with training data (b) gluconic acid yield with testing data

The ANN-acquired equations were subjected to the aforementioned assessment to see if the produced equation agrees with the experimental data. The models that were created were thoroughly tested. Assume that a ten-test data set was created with all variables set to their 50th percentile value, except for oxygen inlet concentration, which was changed at 10-point intervals from its minimum to maximum value. After that, a graph of glucose concentration vs. gluconic acid yield was made. Using these charts, Table 2 observation number 1 was confirmed. Only one model equation for gluconic acid yield was chosen from the shortlisted model equations because they are exceedingly precise, obey the Table 2 observations, and capture the internal physics of the reactor, such equation is considered the representative model equation for gluconic acid yield.

The corresponding Coefficient of Determination (R^2) and Average Percentage Error (APE) of the above model for training and test data have been mentioned in Table 3.

TABLE 3. Performance of ANN model

	Training		Testing	
Model	**R^2**	**APE**	**R^2**	**APE**
Gluconic acid yield model	0.99	0.42	0.99	0.42

With high R^2 and low APE for the gluconic acid yield (Table 3), it can be concluded that the predicted output values are comparable to the actual output values and that the models developed are reliable, fairly accurate, and capture the inherent physics of the bioreactor. The model's generalizability and accurate learning on nonlinear input and output relationships are further indicated by the strong R^2 value on unseen test data and low APE.

Figure 4 depicts the models' prediction performance on training and testing data. The fact that the real and forecast curves almost coincide implies that the model has strong prediction accuracy.

From Table 3 and Figure 4 it is concluded that developed model is highly accurate and reliable as it also performs well with unseen test data.

Generation of explainable model equations: Though ANN generates a closed form of equation which has very high predictive capability, the developed equation is large and complex and sometimes difficult to directly interpret. In present study, a methodology is developed to enhance the interpretability of the developed equations. Figure 5 summarizes the developed methodology. Those figures are generated by changing one variable at a time from its minimum to maximum value (10 steps) while keeping all other 2 input variables at their 50-percentile value. Yield equations developed by ANN is used to predict the yield value in each case of these simulated test data. After plotting was done, a trendline was drawn through each data whose equation and R^2 value is shown in figure. Based on visual inspection and R^2 value trend line curve was selected (like straight line, or polynomial with degree 2 or 3 or more) so that generated trendline almost matches with the data. As seen from the figure 10, the developed trend lines are very decisive and monotonically increasing and decreasing. As mentioned earlier, they all match the actual observations and obey the Table 2. In short, developed models captures the nonlinear relationship between yield and reactor operating parameters. These trend lines can be used by scientists to get the insights on how a particular input parameter affects the gluconic acid yield. For example, from figure 5, it is quantitively clear that increasing glucose and DO concentration, actually enhance the yield whereas increasing biomass reduces the yield.

The yield decreases linearly with biomass concentration with negative slope of 12.172, whereas the relation of yield with glucose and DO concentration are nonlinear and represented by second and third order polynomial, respectively. Now these trend line equations are used to develop the following explainable equation (Equation 18).

$$Yield = -0.0053\left(x_1^2 - x_{1,avg}^2\right) + 2.1499\left(x_1 - x_{2,avg}\right) - 12.172\left(x_2 - x_{3,avg}\right) - 0.0012\left(x_3^3 - x_{3,avg}^3\right) + 0.1136\left(x_3^2 - x_{3,avg}^2\right) - 1.3496\left(x_3 - x_{3,avg}\right) + 72.71 \quad (18)$$

where x_1, x_2, x_3 are the actual value of the 3 input variables and $x_{1,avg}$, $x_{2,avg}$, $x_{3,avg}$ are the average (50 percentile) value of input variables, respectively.

Each term in the equation 18 represents the change in yield if a particular parameter deviates from its average value. For example, the term, $\left(x_2 - x_{2,avg}\right)$ represents the deviation of biomass concentration from its average value and when it multiplied by co-efficient -12.172, represents the yield penalty (or gain) due to biomass. In this way, all 3 parameters contribution is calculated in equation 18 and it is added with 72.71% (average yield) to get the actual yield.

Main advantage of this equation (equation 18) over ANN model is that this equation is interpretable and easily

explainable to engineers. Equation is simple and contains terms or parametric co-efficient which throw lights relative importance of each parameter on the overall yield if they deviate from this base value. Also, it indicates whether the effect of each parameter is linear or non-linear.

Equation 18 is then used to predict yield of experimental data and predicted, and actual yield is compared. Prediction error is 0.42% and R^2 is 0.99. This low value of prediction error and high value of R^2 signifies that developed equation (equation 18) is highly accurate and reliable.

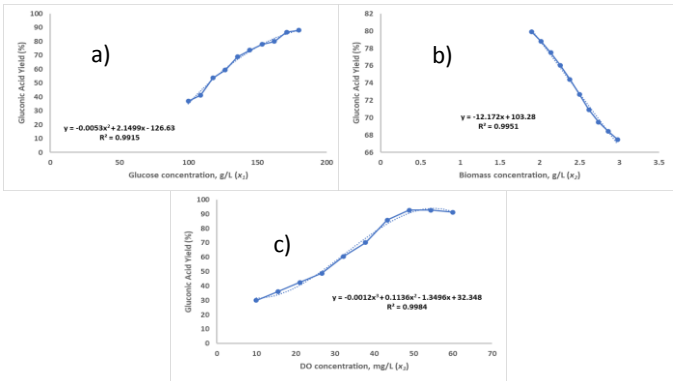

FIGURE 5: Influence of each variable on gluconic acid yield

5.2. Optimization

5.2.1. Optimization through GA

Once the reliable models were successfully developed, the models were subjected to optimization. Purpose of optimization is to find the optimum value of rector operating parameters to achieve maximum gluconic acid yield. One of the critical tasks for optimization of any process is fixing of search space at which the optimal process conditions are to be found out. Therefore, before running the optimization a lower bound and upper bound of the process variables were fixed in consultation with scientists. The lower bounds (LB) and upper bounds (UB) considered in this case have been depicted in Table 4.

Table 4. Lower bounds and upper bounds for optimization

	x_1	x_2	x_3
LB	100	1	10
UB	180	3	60

With the help of GA tool in MATLAB, the optimum experimental conditions were found out which gives the gluconic acid yield of 99.59% (Table 5). The main advantage of such a study is that it gives the experimental engineers a strategy to run the reactor in optimum condition in real-time. Due to unavailability of an explainable model, since scientists have no idea about optimal solution, experimentalists try to optimize the process heuristically based on their experience and knowledge. The only action scientist must do is to run the GA with proper bound in real-time, and GA will provide a set of optimum operating conditions that the scientist needs to set in the experiment.

Table 5. GA optimal solution

x1	x2	x3	Yield
161.80	1.00	58.64	99.59

6. Conclusion

From the existing operating data, this study uses Artificial Neural Networking to construct an accurate model of a gluconic acid bioreactor. An ANN creates a closed model equation that is portable and may be used in a control system. The true value of this research is that it has produced an explainable model equation that is very accurate and provides insights into the process. The produced model equations are based on the underlying physics of the process and are in line with the observations and experiences of the experimentalist. After that, the developed model equations are used to construct optimum solution that optimize gluconic acid yield and thus ensure profit maximization.

References

1. Lahiri, Sandip K. Profit Maximization Techniques for Operating Chemical Plants. John Wiley & Sons, 2020.
2. Lakshminarayanan, S., Fujii, H., Grosman, B., Dassau, E., Lewin, D.R., 2000. New product design via analysis of historical databases. Comput. Chem. Eng. 24, 671–676. https://doi.org/10.1016/S0098-1354(00)00406-3
3. Grosman, B., Lewin, D.R., 2002. Automated nonlinear model predictive control using genetic programming. Comput. Chem. Eng. 26, 631–640. https://doi.org/10.1016/S0098-1354(01)00780-3
4. Searson, D., Willis, M., Montague, G., 2007. Co-evolution of non-linear PLS model components. J. Chemom. 21, 592–603. https://doi.org/10.1002/cem.1084
5. Barati, R., Neyshabouri, S.A.A.S., Ahmadi, G., 2014. Development of empirical models with high accuracy for estimation of drag coefficient of flow around a smooth sphere: An evolutionary approach. Powder Technol. 257, 11–19. https://doi.org/10.1016/j.powtec.2014.02.045
6. Floares, Alexandru G., and Irina Luludachi. "Inferring transcription networks from data." Springer Handbook

of Bio-/Neuroinformatics. Springer, Berlin, Heidelberg, 2014. 311-326.
7. Horne, B.G., 1993. Progress in supervised neural networks. Signal Process. Mag. IEEE 10, 8–39.
8. Hagan, M.T., Menhaj, M.B., 1994. Training Feedforward Networks with the Marquardt Algorithm. IEEE Trans. Neural Networks 5, 989–993. https://doi.org/10.1109/72.329697
9. Adeli, H., Hung, S.L., 1994. An Adaptive Conjugate Gradient Learning Algorithm. Appl. Math. Comput. 62, 81–102.
10. Zhang, N., 2009. An online gradient method with momentum for two-layer feedforward neural networks. Appl. Math. Comput. 212, 488–498. https://doi.org/10.1016/j.amc.2009.02.038
11. Zhang, J.R., Zhang, J., Lok, T.M., Lyu, M.R., 2007. A hybrid particle swarm optimization-back-propagation algorithm for feedforward neural network training. Appl. Math. Comput. 185, 1026–1037. https://doi.org/10.1016/j.amc.2006.07.025
12. Gori, M., Tesi, A., 1992. On the problem of local minima in backpropagation. IEEE Trans. Pattern Anal. Mach. Intell. https://doi.org/10.1109/34.107014
13. Shaw, D., Kinsner, W., 1996. Chaotic simulated annealing in multilayer feedforward networks. Can. Conf. Electr. Comput. Eng. 1, 265–269. https://doi.org/10.1109/ccece.1996.548088
14. Koh, C.S., Hahn, S.Y., 1994. Detection of Magnetic Body using Artificial Neural Network with Modified Simulated Annealing. IEEE Trans. Magn. 30, 3644–3647. https://doi.org/10.1109/20.312730
15. Montana, D.J., Davis, L., 1989. Training Feedforward Neural Networks Using Genetic Algorithms. Proc. 11th Int. Jt. Conf. Artif. Intell. - Vol. 1 89, 762–767.
16. Kiranyaz, S., Ince, T., Yildirim, A., Gabbouj, M., 2009. Evolutionary artificial neural networks by multi-dimensional particle swarm optimization. Neural Networks 22, 1448–1462. https://doi.org/10.1016/j.neunet.2009.05.013
17. Settles, M., Rylander, B., 2002. Neural network learning using particle swarm optimizers. Adv. Inf. Sci. Soft Comput. 224–226.
18. Zhang, C., Li, Y., Shao, H., 2000. A new evolved artificial neural network and its application. Proc. World Congr. Intell. Control Autom. 2, 1065–1068. https://doi.org/10.1109/wcica.2000.863401
19. van den Bergh, F., Engelbrecht, A.P., 2000. Cooperative Learning in Neural Networks using Particle Swarm Optimizers. South African Comput. J. 26, 84–90.
20. Zhang, C., Shao, H., Li, Y., 2000. Particle swarm optimization for evolving artificial neural network. Proc. IEEE Int. Conf. Syst. Man Cybern. 4, 2487–2490. https://doi.org/10.1109/icsmc.2000.884366
21. Mirjalili, S., Sadiq, A.S., 2011. Magnetic Optimization Algorithm for training Multi Layer Perceptron. 2011 IEEE 3rd Int. Conf. Commun. Softw. Networks, ICCSN 2011 42–46. https://doi.org/10.1109/ICCSN.2011.6014845
22. Si, T., Hazra, S., Jana, N.D., 2012. Artificial neural network training using differential evolutionary Algorithm for classification. Adv. Intell. Soft Comput. 132 AISC, 769–778. https://doi.org/10.1007/978-3-642-27443-5_88
23. Leung, Y., Gao, Y., Xu, Z. Ben, 1997. Degree of population diversity - A perspective on premature convergence in genetic algorithms and its Markov chain analysis. IEEE Trans. Neural Networks 8, 1165–1176. https://doi.org/10.1109/72.623217
24. Hrstka, O., Kučerová, A., 2004. Improvements of real coded genetic algorithms based on differential operators preventing premature convergence. Adv. Eng. Softw. 35, 237–246. https://doi.org/10.1016/S0965-9978(03)00113-3
25. Liang, J.J., Qin, A.K., Suganthan, P.N., Baskar, S., 2006. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. IEEE Trans. Evol. Comput. 10, 281–295. https://doi.org/10.1109/TEVC.2005.857610
26. Gao, W. feng, Liu, S. yang, Huang, L. ling, 2012. Particle swarm optimization with chaotic opposition-based population initialization and stochastic search technique. Commun. Nonlinear Sci. Numer. Simul. 17, 4316–4327. https://doi.org/10.1016/j.cnsns.2012.03.015
27. Moslemipour, G., Lee, T.S., Rilling, D., 2012. A review of intelligent approaches for designing dynamic and robust layouts in flexible manufacturing systems. Int. J. Adv. Manuf. Technol. 60, 11–27. https://doi.org/10.1007/s00170-011-3614-x
28. Mirjalili, S., Mirjalili, S.M., Lewis, A., 2014. Grey Wolf Optimizer. Adv. Eng. Softw. 69, 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007
29. Faris, H., Aljarah, I., Al-Betar, M.A., Mirjalili, S., 2018. Grey wolf optimizer: a review of recent variants and applications. Neural Comput. Appl. 30, 413–435. https://doi.org/10.1007/s00521-017-3272-5

30. Saremi, S., Mirjalili, S.Z., Mirjalili, S.M., 2015. Evolutionary population dynamics and grey wolf optimizer. Neural Comput. Appl. 26, 1257–1263. https://doi.org/10.1007/s00521-014-1806-7
31. Nadimi-Shahraki, M.H., Taghian, S., Mirjalili, S., 2021. An improved grey wolf optimizer for solving engineering problems. Expert Syst. Appl. 166, 113917. https://doi.org/10.1016/j.eswa.2020.113917
32. Mirjalili, S., Saremi, S., Mirjalili, S.M., Coelho, L.D.S., 2016. Multi-objective grey wolf optimizer: A novel algorithm for multi-criterion optimization. Expert Syst. Appl. 47, 106–119. https://doi.org/10.1016/j.eswa.2015.10.039
33. Singh Cheema, J.J., Sankpal, N. V., Tambe, S.S., Kulkarni, B.D., 2002. Genetic programming assisted stochastic optimization strategies for optimization of glucose to gluconic acid fermentation. Biotechnol. Prog. 18, 1356–1365. https://doi.org/10.1021/bp015509s
34. Pal, S., & Lahiri, S. K. (2022). Grey wolf optimizer trained ANN technique for development of explainable model of commercial ethylene oxide reactor and multi-objective optimization to maximize profit.
35. Pal, S., Chowdhury, S., Hens, A., & Lahiri, S. K. (2022). Artificial intelligence based modelling and multi-objective optimization of vinyl chloride monomer (VCM) plant to strike a balance between profit, energy utilization and environmental degradation. Journal of the Indian Chemical Society, 99(1), 100287.
36.

APPENDIX:

Experimental Data Utilized for Building GP Based Model taken from [33]

batch no	glucose concn (x1) (g/L)	biomass concn (x2) (g/L)	DO (x3) (mg/L)	gluconic acid yield (Y_1) (%)
1	100	1	10	5.9
2	150	2	10	29.42
3	120	2	15	20.76
4	150	2.5	15	35.51
5	150	3	15	35.16
6	120	2	25	27.77
7	120	2	30	34.48
8	150	2	30	57.86
9	150	3	25	49.32
10	150	2	40	78.99
11	150	2	45	89.48
12	150	2	50	94.5
13	180	2	50	89.63
14	150	3	40	79.05
15	150	2.5	50	94.58
16	150	2.5	55	93.41
17	150	2.5	60	91.26
18	160	2.5	60	93.67
19	175	3	55	92.69
20	160	3	60	93.3
21	180	3	60	88.13
22	150	3	60	92.7
23	100	3	60	20.04
24	100	2	10	6.13