

**UNIVERSIDAD AUTONOMA DE NUEVO LEON**  
**FACULTAD DE INGENIERIA MECANICA Y ELECTRICA**  
**SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO**



**“OPTIMIZATION IN PREPARATION PROCESS OF  $V_2O_5$  USING SYMBOLIC  
REGRESSION  $\alpha$ - $\beta$ ”**

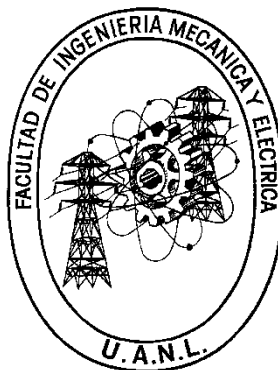
**By:**  
**M. ENG. GUILLERMO GONZALEZ CAMPOS**

**IN ORDER TO OBTAIN THE DEGREE OF:**  
**DOCTORADO EN INGENIERIA CON ORIENTACION EN TECNOLOGIAS DE  
LA INFORMACION**

**SAN NICOLÁS DE LOS GARZA, NUEVO LEÓN**

**January 2019**

**UNIVERSIDAD AUTONOMA DE NUEVO LEON**  
**FACULTAD DE INGENIERIA MECANICA Y ELECTRICA**  
**SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO**



**“OPTIMIZATION IN PREPARATION PROCESS OF  $V_2O_5$  USING SYMBOLIC  
REGRESSION  $\alpha$ - $\beta$ ”**

**By:**  
**M. ENG. GUILLERMO GONZALEZ CAMPOS**

**IN ORDER TO OBTAIN THE DEGREE OF:**  
**DOCTORADO EN INGENIERIA CON ORIENTACION EN TECNOLOGIAS DE  
LA INFORMACION**

**SAN NICOLÁS DE LOS GARZA, NUEVO LEÓN**

**January 2019**

**UNIVERSIDAD AUTONOMA DE NUEVO LEON**  
**FACULTAD DE INGENIERIA MECANICA Y ELECTRICA**  
**SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO**

Los miembros del Comité de Tesis recomendamos que la Tesis "**OPTIMIZATION IN PREPARATION PROCESS OF  $V_2O_5$  USING SYMBOLIC REGRESSION  $\alpha$ - $\beta$** " realizada por el alumno(a) **GUILLERMO GONZALEZ CAMPOS**, con número de matrícula 1306412 sea aceptada para su defensa como opción al grado de "DOCTORADO EN INGENIERIA CON ORIENTACION EN TECNOLOGIAS DE LA INFORMACION"

El Comité de Tesis



Dr. Luis Martin Torres Treviño  
Director



Dr. Cesar Guerra Torres  
Revisor



Dr. Elias Gabriel Carrum Siller  
Revisor

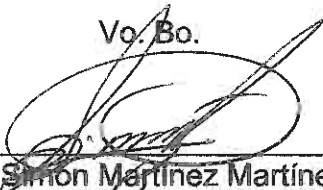


Dr. Rolando Javier Praga Alejo  
Revisor



Dra. Leticia Amalia Neira Tovar  
Revisor

Vo. Bo.



Dr. Simon Martinez Martinez  
Subdirector de Estudios de Posgrado



San Nicolás de los Garza, Nuevo León, January 2019

---

# CONTENTS

<b>Abstract</b>	1
 <b>Chapter 1.</b>	
<b>Introduction</b>	3
1.1. Problem description	3
1.2. State of the art	4
1.3. Justification	6
1.4. Hipotesis	7
1.5. General objective	7
 <b>Chapter 2.</b>	
<b>Background</b>	8
2.1. Introduction to the industrial processes	8
2.2. Introduction to genetic programming and symbolic regression	9
2.3. Introduction to artificial neural networks	9
2.4. Introduction to linear regression	10
2.5. Introduction to symbolic regression alpha-beta	10
2.6. Introduction to evaluation metrics	13
2.7. Proposed method for applying symbolic regression $\alpha$ - $\beta$	14
 <b>Chapter 3.</b>	
<b>Experimental description</b>	17
3.1. Description of process	17
3.1.2. $V_2O_5$ compound	18
3.1.2. $V_2O_5$ preparation and data acquisition	18

---

3.2.	Characteristics of evolutionary algorithm	21
3.3.	Other approaches to compare	23
<b>Chapter 4.</b>		
<b>Results</b>		25
4.1.	Modelling with symbolic regression	25
4.2.	Comparisons	28
4.3	Response surfaces	29
<b>Chapter 5.</b>		
<b>Conclusions</b>		32
<b>References</b>		34
<b>List of Figures</b>		39
<b>List of Tables</b>		40
<b>Appendix</b>		
Appendix A		41
Appendix B		46
Appendix C		79

---

# ABSTRACT

Guillermo González Campos

Candidate for the Degree of Doctor in Engineering with orientation in Information Technologies

Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

Title of study:

Optimization in preparation process of  $V_2O_5$  using symbolic regression  $\alpha$ - $\beta$

In this work a symbolic regression algorithm was used for modeling the preparation process of the compound  $V_2O_5$ . This algorithm was used with the proposal method on this work, were 8 steps are proposed to have better and faster results when symbolic regression is applied for industrial processes. The preparation of  $V_2O_5$  consists in modifying the solvent of the medium and calcination temperature to obtain a compound with different physical properties. A mathematical model of four input variables and one output response was generated. This compound is widely used in catalysis and photocatalysis to solve environmental problems, such as water and air purification. The aim of creating a model was to obtain the best combination of preparation variables that result in a

---

compound with the ideal properties, which promote a better performance in catalysis and photocatalysis. A design of experiments was made to obtain a data-driven model. Initial population, selection and iterations were considered to enhance the results when symbolic regression is applied. Genetic programming, artificial neural networks and linear regression were compared with symbolic regression in order to know which technique has better efficiency to generate models. Some metrics related with quality and deliveries of the model were proposed to compare the results. The model generated with symbolic regression showed better results in comparison with other techniques used, due to the metrics used describes less error and better performance to predict new data. Responses surfaces were made using the generated SR model and were compared showing different perspective in order to optimize the quantity of materials used for the preparation process of  $V_2O_5$ .

# 1.INTRODUCTION

## 1.1 PROBLEM DESCRIPTION

One of several approaches of researching in computer science is to get better and faster ways to solve problems. In some fields of study different from computer science there is an opportunity area where new techniques can be applied and studied. In computer science, artificial intelligence (AI) techniques are being used to help the humanity for solving problems in daily life, industry and academic research. AI has many techniques and methods with different strengths and weaknesses that can be adapted depending of the application. One of the methods to solve problems using AI is the Artificial Neural Network (ANN). For example, ANN is common used to solve problems of classification, pattern recognition and clustering, among others.

Genetic Programming is other technique in AI that use genetic or evolutionary algorithm to perform a predefined task, this programming evolves as biological systems and its evolutionary theories, where genes are changing during the time. There is an application in genetic programming (GP) called symbolic



regression (SR) that has been used for modeling on industrial and dynamic process [1, 2].

Models are useful when the process of interest needs to be improved. Mathematical models are the most common used, however there are other forms to generate models with computer science techniques. ANN can be used for optimizing [3]; however, they are black boxes where an explicit formulation about the correlation between variables and effects on output response is not evident [4]. In chemical processes for academic research, data acquisition of experiments is necessary to repeat reactions procedures. The repetitions are expensive and time spent in the process is very important even on industry. The main motivations for creating models in chemical process are to reduce costs and time spent. In this paper, a model using symbolic regression is proposed to obtain the best combination of preparation variables of  $V_2O_5$  that result in a compound with the ideal properties that promote a better performance in catalysis and photocatalysis.

## 1.2 STATE OF THE ART

A recently study [5] investigate the evidence for SR using GP being an effective method to prediction and estimation in software engineering. They used 23 primary studies from 1995 to 2008, the results show that SR using GP has been applied in three domains within software engineering predictive modeling; software quality classification, software cost/effort/size estimation and software fault prediction/software reliability growth modeling. Smits et al [6], use SR to obtain the maximum scalability to architectures with a very large number of processors in a process of a distillation tower with 23 inputs and 5000

records. In other work of Smits et al [7] give an overview of the importance of variable selection to build robust models from industrial datasets.

Castillo et al [8] uses SR and a design of experiments to obtain the maximum data utilization when extrapolation is necessary. In combination with Pareto front Castillo et al [9], also uses SR bases on design of experiments and industrial data. Kotanchek et al [10] summarize their experience in industrial application of genetic programming to empirical modeling and transfer key learnings with respect to real-world application. Oliveria et al [11] change the basic behavior of the method of SR adding some concepts of evolution strategies (ES) obtaining excellent results. Dervis et al [12] made a work where a set of SR problems were solved using artificial bee colony programming and then their performance was compared with the very well known method evolving computer programs, GP. Dabhi et al [13] explored the suitability of ANN and SR to solve empirical modeling problems and conclude that SR can deal efficiently with these problems.

Cai et al [14] describe a methodology that uses SR to extract correlations from heat transfer measurements by searching the form of correlation equation and the constants in it that enable the closest fit to experimental data. Zhu et al [15] present a method for multivariable SR modeling and predicting, based on gene expression programming, furthermore they give an example to explain this technique and experiment results show that the model set up is better than statistical linear regression techniques. Davidson et al [16] describes a new method for creating polynomial regression models and is compared with stepwise regression and SR using three example problems, this new method includes some changes on the basic genetic programming algorithm first proposed by Koza [17]. Finally Barmpalexis et al [18] use SR via GP in the

optimization of a pharmaceutical zeroorder release matrix table, and its predictive performance was compared with ANN models.

Going deeper in SR, more industrial processes have been described with SR models, in 2013 a model for cutting machining processes using SR was proposed [19]. This model can be used to establish the machining parameters to obtain the desired roughness. Recently, a model with SR was used for setting of machining parameters and tool selection for a cutting process. The model improves the quality of process and increases its performance, which results good for the company [20].

### **1.3 JUSTIFICATION**

The benefit of working with new techniques for novel applications on industry and science is that the generated knowledge will help other researchers to improve their processes of investigation. SR as novel technique still have a wide field of study and experimentation on different applications. On this work the process of synthesis of the photocatalyst  $V_2O_5$  has been chosen to optimize and improve its process of synthesis. A method is proposed to apply SR on this chemical process where some variables are involved to get best results.

## **1.4 HIPOTHESIS**

The process of synthesis of the photocatalyst  $V_2O_5$  can be optimized using on technique of artificial intelligence called symbolic regression alpha – beta. The way to optimize the process is proposing a method to apply the SR with some steps clearly defined.

## **1.5 GENERAL OBJECTIVE**

Propose a method to apply symbolic regression and optimize the process synthesis of photocatalyst  $V_2O_5$ .

## **2.BACKGROUND**

### **2.1 INTRODUCTION TO THE INDUSTRIAL PROCESSES**

Some centuries ago when industrial revolution started, new machines have been created to improve the productive processes using different energy sources. Processes like developing textiles, vapor machines for transportation were improved during industrial revolution. The main objective for improving the processes is to produce more with less materials, energy or resources. Production methodologies and new materials have been used to enhance the results, but this challenge every time has been more difficult.

Thanks to the creation of new materials as semiconductors, the integrated circuits have been developed and computers have been invented. New fields of study arrived thanks at the computers, math problems are solved faster than decades ago and computer science starts rising to create techniques for problem solving. Actually, techniques of computer science are used to improve industrial processes.

## **2.2 INTRODUCTION TO GENETIC PROGRAMING AND SYMBOLIC REGRESSION**

Genetic Programming (GP) is an extension of Evolutionary Algorithms (EA) presented by Koza in 1990 [17]. GP is a representation of a data tree structure, where nodes show a function that has arithmetic and logic operations, and leafs represent variables and constants. When a GP is running, a function is created with the nodes and leafs. This function is evaluated for each generation and genetic operators such as crossover and mutation are used to improve the results.

## **2.3 INTRODUCTION TO ARTIFICIAL NEURAL NETWORKS**

Other novel technic on artificial intelligence area are the artificial neural network (ANN) which is a technique inspired by biological neuron processing. It has a wide application on several sciences for time series forecasting, pattern recognition and process control, but the most used application are for classification and regression. Training of the neural networks is sensitive to the number of neurons in the hidden layer. A better performance of the neural network in fitting data can be reached when is involved a high number of neurons. However too many neurons in the hidden layer may result in the over fitting. The neural network has some desirable characteristics like robustness and precision in the approximation nevertheless neural networks are black boxes due to there is not a mathematical equation which explains the model.

## 2.4 INTRODUCTION TO LINEAR REGRESSION

Linear regression is statistical technique used for discovering the relation between one or more variables, mainly is applied in engineering and science. For example, some industrial processes have one variable that is dependent from another one, and its relationship is described in a linear equation or linear regression model. If we have data table with 2 variables, where “y” increase its value while “x” increases, this behavior can be expressed in a mathematical linear model. This model also can be used to predict new values inside the range of data table. The simple linear regression model is explained in equation 1:

$$Y = \beta_0 + \beta_1 x + \epsilon \quad (1)$$

Where Y is a linear function of x, and  $\epsilon$  is the random error term. This technique is useful when the variables and its behavior are strongly linked in a linear way. However, sometime the processes in real life are not lineal and this technique will have some restrictions. So, it is required to use other approaches, such as symbolic regression  $\alpha$ - $\beta$ , which is proposed in this work.

## 2.5 INTRODUCTION TO SYMBOLIC REGRESSION $\alpha$ - $\beta$

The core of this work is based in symbolic regression  $\alpha$ - $\beta$  approach [4]. Where mathematical equations are represented by the combination of  $\alpha$  and  $\beta$  operators. An  $\alpha$  operators is a function which needs one argument and applies

---

one mathematical operation, 13 operations are shown in table 1. These operations are chosen as  $\alpha$  operators.

An  $\alpha$  operator uses two real number parameters called  $k_1$  and  $k_2$ . Also, an integer that describes the mathematical operation is used. The  $\alpha$  operator is shown in equation 2:

$$Opr_{\alpha}(x, k_1, k_2) = \alpha(k_1 * x + k_2) \quad (2)$$

where  $x$  is an input variable and  $\alpha$  is an operation. Depending of the  $\alpha$  operator selected, a specific mathematical operation that requires only one argument is executed; For example, if  $\alpha = 1$  then the operation made is  $(k_1 * x + k_2)$ . The  $\alpha$  operator is an integer number and its value determinate a specific mathematical operation described in Table 1. The  $\beta$  operators are described as a function that require two arguments and makes the four basic arithmetic operations  $\beta = c$  so a  $\beta$  operator equal to 1 imply the plus operator or  $\beta(a, b) = a + b$ , and  $\beta(a, b) = a/b$  if  $\beta = 4$ .



Table 1.  $\alpha$  Operators parameters and mathematical function related

$\alpha$ Operator	Mathematical operation
1	$(k_1x + k_2)$
2	$(k_1x + k_2)^2$
3	$(k_1x + k_2)^3$
4	$(k_1x + k_2)^{-1}$
5	$(k_1x + k_2)^{-2}$
6	$(k_1x + k_2)^{-3}$
7	$(k_1x + k_2)^{1/2}$
8	$(k_1x + k_2)^{1/3}$
9	$\exp(k_1x + k_2)$
10	$\log(k_1x + k_2)$
11	$\sin(k_1x + k_2)$
12	$\cos(k_1x + k_2)$
13	$\tan(k_1x + k_2)$

By means of  $\alpha$ - $\beta$  operators several configurations can be established. A basic configuration can be defined when an  $\alpha$  operator is assigned per input variable then an  $\beta$  operator is used to connect two  $\alpha$  operators (3). Usually, a simple configuration in majority of the cases is enough for the regression.

$$y = \beta_{n-1}(\dots \beta_2(\beta_1(\alpha_1), \alpha_2(x_2)), \dots, \alpha_n(x_n)) \quad (3)$$

The representation required is a real vector with  $n$  element where  $n$  is equal to the number of  $\alpha$  operators and  $k$  parameters plus  $\beta$  operators. Using one  $\alpha$  operator per variable and connect them by  $\beta$  operators, the number of parameters is given by the number of  $\alpha$  operators, the number of  $\beta$  operators and  $k$  parameters (two per  $\alpha$  operator). In a basic structure is  $\alpha + \beta + 2 * \alpha$ , because  $\beta = \alpha - 1$ , and  $\alpha$  = number of variables ( $N_v$ ) then the number of parameters is  $N_v + (N_v - 1) + 2 * N_v$ . A normalized real vector can be used to represent operators

and  $k$  parameters, but  $\alpha$  and  $\beta$  operators are integers, so is required the following formulation to get its value:

$$\alpha = \lceil V(i) * 13 + 0.5 \rceil \quad (4)$$

$$\beta = \lceil V(i) * 4 + 0.5 \rceil \quad (5)$$

where  $\lceil . \rceil$  is the ceiling function. There are 13  $\alpha$  operators defined in Table 1, and 4  $\beta$  operators (basic algebraic operations). In this work, Evonorm [21, 22] is used to solve the problem of selection of the suitable parameters ( $k$ 's), and integers to define  $\alpha$  and  $\beta$  operations.

## 2.6 INTRODUCTION TO THE EVALUATION METRICS

The statistics metrics proposed in this work are: (i) mean square error (MSE), (ii) prediction error sum of squares (PRESS) and (iii)  $R^2_{pred}$ . This metrics are used in order to validate a regression model. For this reason, is necessary to try it with new experimental data to determine how well is the performance of the model in practice [23]. The simplest measure is the residual calculated as the difference ( $e(i)$ ) between new observations made by the response of the process ( $y(i)$ ) and predicted response generated by the regression model made ( $\hat{y}(i)$ ), (Eq. 6).

$$e(i) = y(i) - \hat{y}(i) \quad (6)$$

PRESS is a measure of how well a model works to predict new data. Usually, a small value of PRESS is desirable (Eq. 7). In this case, PRESS is obtained using cross validation.

$$PRESS = \sum_{i=1}^n (y(i) - \hat{y}(i))^2 \quad (7)$$

The percentage of variability  $R^2_{pred}$  is a measurement for indicating the efficiency of the model to predict new observations. A value close to one is desirable on this indicator (Eq. 8).

$$R^2_{pred} = 1 - \frac{\sum_{i=1}^n (y(i) - \hat{y}(i))^2}{y'y - (\sum_{i=1}^n y(i))^2} \quad (8)$$

MSE calculates the average of squared errors between new observations and experimental data.

An additional metric is proposed to measure the time of calculations for the model; the reference to evaluate how fast it works generating models is central processing unit (CPU) time of the computer running the algorithm.

## 2.7 PROPOSED METHOD FOR APPLYING SYMBOLIC REGRESSION $\alpha$ - $\beta$

In this work, a method for using symbolic regression in an industrial process is proposed in order to optimize it. The method proposed is divided in 7 steps to obtain the best information quality and results.

Steps:

1. **Identify Variables.** On this step is necessary to identify the variables of the process; input variables and response. The input variables change the result of response when we change its values. The response variable is the desired output that determines how well our process was performed.

- 2. Data acquisition.** For the step 2, we need to get some data from the process, in order to feed the symbolic regression algorithm and get a good training. The data can be obtained from historic results and it is recommended to have data with good and bad results, but with all the information complete from all variables. If it is not possible to get many results from the process, a Design Of Experiments (DOE) can be mode. With a DOE, fewer experiments are needed to obtain data values.
- 3. Pre processing.** Once a data table is obtained, this information should be used for the symbolic regression. Nevertheless some pre processing is needed to make the calculations faster. The input and response data will be normalized from 0 to 1. It is proposed to use 80% of data to generate models with SR and the 20% left will be used for testing. This testing will show results about how well the model generated is good for predicting.
- 4. Define architecture for modeling.** As is mentioned before, some  $\alpha$  operators are used for each variable in symbolic regression  $\alpha$ - $\beta$ . For this method is proposed to use 3  $\alpha$  operators for each variable. This number can change, however if there are more  $\alpha$  operators, the complexity of model also changes. The  $\beta$  remains as an operator that makes the four basic operations.

5. **Set characteristics of evolutionary algorithm.** When we have the data and the architecture of SR algorithm, now the evolutionary algorithm needs to be set in order to have the best population in less time and with few resources.
6. **Results validation.** To evaluate the best performance, four indicators are selected, the same as mentioned before: Mean Square Error (MSE),  $R^2$ Pred, PRESS and CPU time.
7. **Analysis of mathematical model.** When the SR is executed, a model is generated. This mathematical model is the combination of the 4 basic operations and the 13 operations defined in previously in the algorithm. This equation now can be analyzed, for example to find which is the strongest variable that is affecting the response. In this method, this step is recommended in order to find the variables that we can optimize with other statistical techniques.
8. **Optimize model and usage.** Using the model to predict new data is the last step to optimize the process. With the mathematical model it is not necessary a sophisticated software to forecasting. Response surface can be made and other types of plots.

These 8 steps are for the proposal method to apply symbolic regression  $\alpha$ - $\beta$  for industrial process, in order to obtain an optimization of it.

### 3. EXPERIMENTAL DESCRIPTION

### 3.1 DESCRIPTION OF PROCESS

The synthesis process of  $V_2O_5$  consists in 4 inputs variables and 1 output response. This output response is the desired variable to optimize, modifying the inputs. The figure # shows an iconic model of the process.

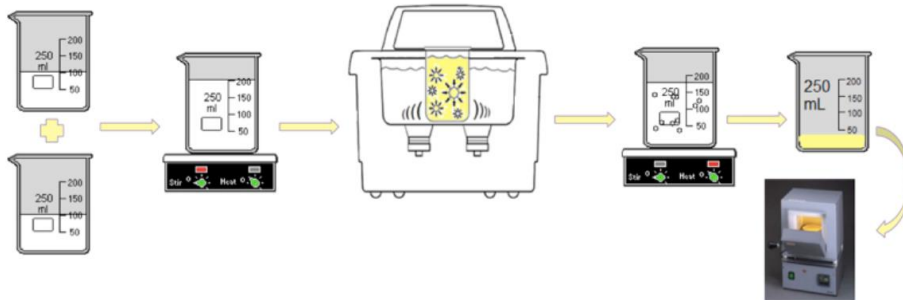


Figure 1. Synthesis process of  $V_2O_5$

This process is interesting to create a model, because even though there is an explanation of how to synthesize this compound, some of the variables are not probed which of them have a high impact in the results. Make trial and fail samples

to find the best result of the compound can be expensive and a waste of time. On industry and research, this samples can be predicted using computer science techniques. For that reason, this chemical process is useful to try how effective is symbolic regression applied for the process. The intention is clear, and is trying to get better results using fewer resources, if with this model generated by SR, the process can be optimized to use less heat treatment, water, or other element, then goal will be reached. Is feasible to obtain a mathematical model with SR, instead of a black box.

### 3.1.1 $V_2O_5$ compound

The  $V_2O_5$  compound has been widely studied for the reduction of  $NO_x$  gases by selective catalytic reduction in presence of ammonia (SCR- $NH_3$ ) [24,25,26]. In this reaction, one important parameter is the number of reaction sites, which is related with its surface area. In addition, the acid sites present in the  $V_2O_5$  surface promotes a better efficiency of the reduction of  $NO_x$  to  $N_2$  and  $O_2$  gases. On the other hand, in the area of photocatalysis the oxide  $V_2O_5$  had been propose as photocatalyst to carry out the removal of organic compounds from industrial waste, i.e., petrochemical and textile [27,28]. In these reactions, the surface area plays an important role to carry out the adsorption of the pollutants in the media to start to decompose them in carbon dioxide and water. For this purpose, we propose a method to prepare  $V_2O_5$  by modifying different experimental conditions that promotes the development of high surface area values.

### 3.1.2 $V_2O_5$ preparation and data acquisition

The  $V_2O_5$  samples were prepared by precipitation method. The chemical materials involved in the process were deionized water, ethylene glycol ( $HOCH_2CH_2OH$ ) (Aldrich, 99%), and ammonium vanadate ( $NH_4VO_3$ ) (Aldrich, 99%). For this purpose, 0.0054 mole of  $NH_4VO_3$  was dissolved in 50 mL of

---

distillated water or ethylene glycol under vigorous stirring for 30 minutes. The solution was exposed to ultrasound irradiation (40 kHz, 70W) under ambient air at 60°C for different time intervals (0-120 minutes). Once the time has lapsed, the resulting mixture was heated at 100°C to promote the slow evaporation of the solvent. The resulted powders were calcined at 400 and 500°C for 24 h to obtain polycrystalline powders.

According to the method proposed in this work, the **Step 1 (Identify variables)** can be done. The input variables are identified, they are:

1. Quantity of H<sub>2</sub>O in mL.
2. Quantity of EG in mL
3. Ultrasonic treatment time in minutes
4. Temperature in Celsius degrees

The response output is:

1. Surface area in m<sup>2</sup>g<sup>-1</sup>

Once the variables are identified, the **Step 2 (Data acquisition)** from the method was started. For this chemical process, the variables were modified according with a design of experiments (DOE) to prepare V<sub>2</sub>O<sub>5</sub> at different conditions. As a result, 18 samples were prepared modifying 4 inputs variables and measuring one response to evaluate its physical properties. The experimental data set is shown in the Table 2.



Table 2. Experimental data set of  $V_2O_5$ 

Sample	H <sub>2</sub> O (mL)	EG (mL)	Ultrasonic time (min)	T (°C)	Surface area (m <sup>2</sup> g <sup>-1</sup> )
1	0	50	0	400	7.7419
2	0	50	60	400	12.1860
3	0	50	120	400	10.3980
4	0	50	0	500	4.1292
5	0	50	60	500	1.9645
6	0	50	120	500	2.3174
7	25	25	0	400	7.4281
8	25	25	60	400	8.0970
9	25	25	120	400	9.7480
10	25	25	0	500	3.9813
11	25	25	60	500	4.2494
12	25	25	120	500	4.5984
13	50	0	0	400	2.9952
14	50	0	60	400	4.2209
15	50	0	120	400	4.8132
16	50	0	0	500	2.9073
17	50	0	60	500	4.6259
18	50	0	120	500	3.8645

### 3.2 CHARACTERISTICS OF EVOLUTIONARY ALGORITHM

For this work Evonorm is used as the evolutionary algorithm [21,22]. Some of the parameters to adjust that will affect directly to response variable are population, individuals selected and iterations.

The table 3 shows the values selected to adjust in the algorithm, and with this data, a design of experiments (DOE) were used.

Table 3. Level of DOE

Population	Selection	Iterations
100	10	150
200	20	500
	30	

Twelve groups were needed to know its best performance, table 4 shows the groups.

Table 4. Groups of DOE

Group	Population	Selection	Iterations
1	100	10	150
2	100	10	500
3	100	20	150
4	100	20	500
5	100	30	150
6	100	30	500
7	200	10	150
8	200	10	500
9	200	20	150
10	200	20	500
11	200	30	150
12	200	30	500

Finally, on appendix A is shown the information obtained from running the algorithm for 17 hours on 120 runs during different days. The equipment used to run the algorithm was a MacBook Air Laptop with 4GB RAM and a processor of 1.5 Ghz Core i5.

With this information, on table 5 now we can observe that the best group with better performance indicators is the group 7, however its time performance is not the best but is under the mean (425 seconds). Meanwhile the best group with the faster time is group 5, but its performance indicator to predict new data is very poor.

Table 5. Results of means of each group of DOE

Group	MSE	R2pred	PRESS	CPU time (Seconds)
1	0.004216	0.9798	0.4216	161.3561
2	0.00378	0.9795	0.378	729.3228
3	0.004385	0.9782	0.4385	129.8918
4	0.004174	0.9796	0.4174	463.435
5	0.004443	0.9794	0.4443	120.6474
6	0.00404	0.98	0.404	387.8231
7	0.0032499	0.984	0.32499	220.2551
8	0.00427	0.9795	0.427	721.9884
9	0.003831	0.982	0.3831	231.5504
10	0.0041221	0.9784	0.41221	704.9504
11	0.003916	0.9789	0.3916	265.6669
12	0.003608	0.9811	0.3608	968.1012

With these results the proposal to use in the architecture for evolutionary algorithm for symbolic regression alpha-beta is using population with 200, the individual selected with 10 and running for 150 iterations.

### 3.3 OTHER APPROACHES TO COMPARE

When the data is acquired, before to use it with SR will be helpful to compare it with other tools. This comparison is to know which techniques can explain the process as well thru a model, it is proposed to use genetic programming, artificial neural network and linear regression. If a comparison is made using the results with other techniques, will have a general view of how well it performs the SR

algorithm for this process. Some industrial process can be explained with simpler solutions or not.

Genetic programming uses the following operations  $\{+, -, *, /, \exp, \log\}$  for all nodes for 300 generations, considering 100 individuals, a simply crossover with a probability of 0.9 and a simple mutation with a probability of 0.05. This parameters for GP were found the best solution using the key performance indicators of PRESS, MSE and  $R^2$ Pred. An 80% of experimental data is used for model building and 20% for test validation.

For linear regression is executed under the same conditions and evaluated with the same statistical metrics that are used in this work.

With artificial neural network approach, a perceptron-multilayer neural network with back propagation rule was used, with 8 neurons on middle layer and a constant learning parameter 0.25 and a moment of 0.5 during 800 epochs. These configuration parameters of ANN were the best combination found to get better results using the same statistical metrics that are used with the other techniques.

## 4.RESULTS

### 4.1 MODELLING WITH SYMBOLIC REGRESSION

With the data obtained in Step 2, the **Step 3 (Pre processing)** was made when the algorithm was executed, all the data was normalized from 0 to 1 and, it was defined 80% of data used for generate models and 20% to test them.

To **define the architecture for modeling in Step 4**, the SR algorithm was defined with 3  $\alpha$  operators.

The **Step 5 (Set characteristics of evolutionary algorithm)**, was defined as the method proposal in subchapter 3.2, with a population of 200, 10 individual selected per iteration, during 150 iterations.

The model generated with SR for the oxide  $V_2O_5$  with the best parameter configuration is show in equation 9 (coefficients of equation are normalized):

$$f(x_1x_2x_3) = (((((\tan(0.1990473x_1 + 0.9119780) + \log(0.0415200x_2 + 0.9910278)) + (0.997222x_3 + 0.4656022)^{-2}/(0.6793249x_1 + 0.3253794)^{1/2} -$$

$$\frac{\sin(0.1825961x_2 + 0.9986060)}{(0.3713540x_3 + 0.7209171)} - (0.0123117x_1 + 0.9517829)^3 * (0.0858313x_2 + 0.3856453) + \frac{\sin(0.7869533x_3 + 0.7936947)}{(0.8517493 + 0.9853957)^3} \quad (9)$$

The results with the model and the data set are compared in a plot, the figure 2 shows the comparison between them.

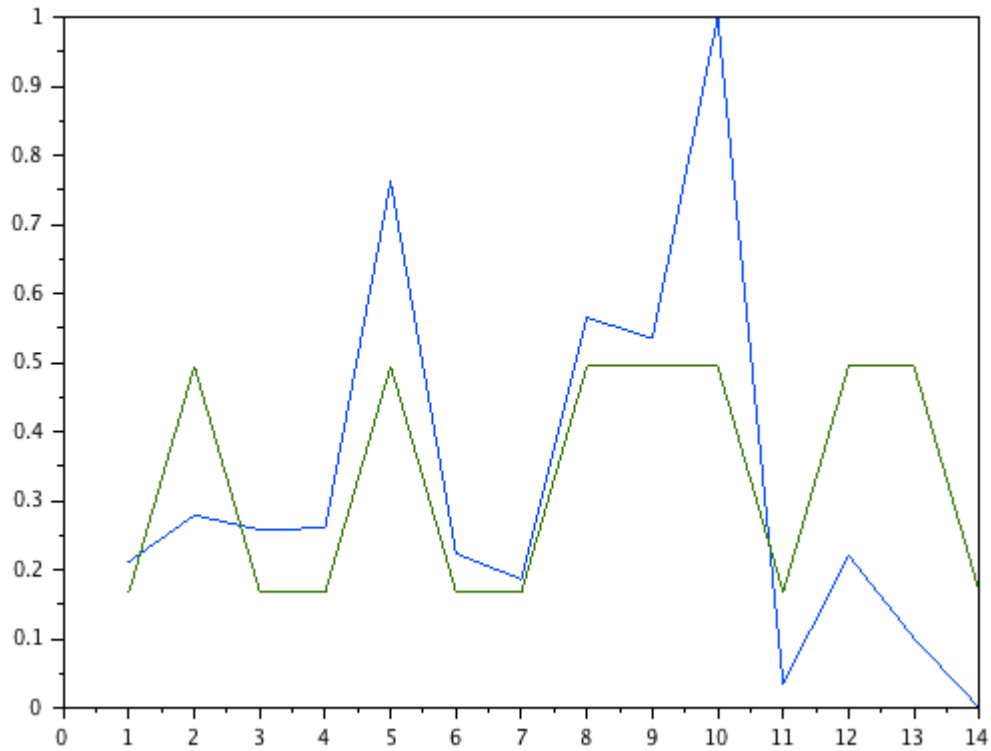


Fig.2 . Comparison data between model results

Also, the evaluation (Figure 3) fitness graph is showed to explain how was its behavior during time on 150 iterations.

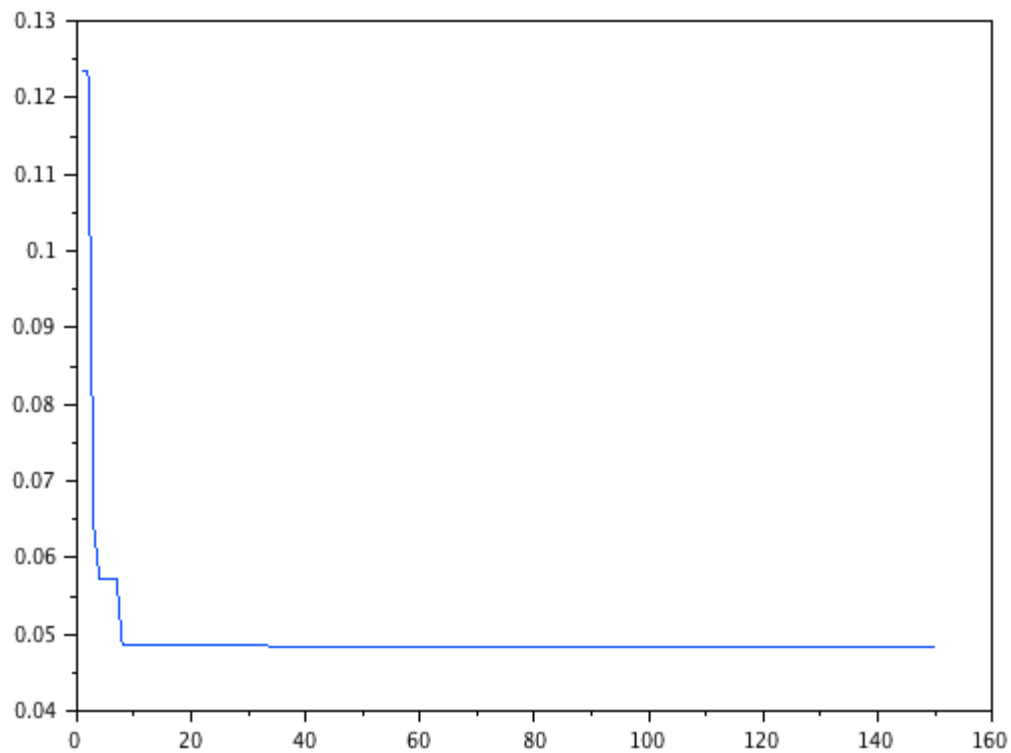


Fig. 3. Evaluation plot

Once the step 5 was performed, the **Step 6 (Results validation)**, was executed, using the performance indicators. The statistics metrics results for this equation are MSE equal to 0.0017962, PRESS equal to 0.1796221 and  $R^2_{pred}$  equal to 0.99125554.2



## 4.2 COMPARISONS

The same results validation using the performance indicators, were used for the other techniques. A resume of performance of the techniques to generate the best model for the preparation process oxide  $V_2O_5$  is shown on the table 6.

Table 6. Statistical metrics results of the best model found of  $V_2O_5$  for linear regression, genetic programming and symbolic regression alpha-beta

Technique	MSE	PRESS	$R^2_{pred}$	CPU time
Linear regresssion	0.146478636	14.6478636	0.364175752	0.009
Genetic programming	0.080161008	8.01610081	0.608180149	488.122
Artificial neural network	0.010416923	1.041692271	0.935792749	18.423
Symbolic regression alpha-beta	0.0017962	0.1796221	0.9912555	125.581

Considering results shown in table 6 the ideal criteria low error, high  $R^2_{pred}$  and low PRESS can be taken here. The best technique for predicting new data according to this table is SR. The technique with less error according to this table is SR, the results show that the best statistic metric values belong to SR model. Performance on CPU time is different in each case and it is expected to be like this, due the fact that SR runs for 150 generations, GP for 300 and ANN for 800 generations. Using artificial neural networks could be a good option, but an explicit correlation between variables and effects on output response is not evident in other words ANN are black boxes.

---

The **Step 7 (Analysis of mathematical model)** was performed. The “y” is surface area, H<sub>2</sub>O is  $x_1$ , ethylenglycol is  $x_2$ , ultrasound irradiation is  $x_3$  and heat treatment temperature is  $x_4$ . Symbolic regression eliminates the factor  $x_4$ , this mean that heat treatment temperature is irrelevant for the response according to this model generated.

It can be see from equation 9 that the time of ultrasound exposure of the reactive mixture is a very important factor, which can be related to the acoustic cavitation that promotes extreme conditions inside the collapsing bubble with hot spots of 5000 K, pressures of 1000 bar (Luévano-Hipólito et al 2014).

### 4.3 RESPONSE SURFACES

Finally for the proposal method to apply SR for a industrial process, the **Step 8 (Optimize model and usage)**, was made. In order to optimize the process, the model generated with SR is used to create response surfaces. For the response surfaces, the variable  $x_4$  was fixed as a static value of 400 and  $x_1$  was fixed to 0, 25 and 50. The variables  $x_2$  and  $x_3$  were calculated with the equation 9 with different values.

For  $x_2$  the values starts from 0 to 50 with steps of 1. For  $x_3$  the values starts from 0 to 120 with steps of 1. In x axis is the variable  $x_3$ , in the y axis is the variable  $x_2$  and z axis is the response.

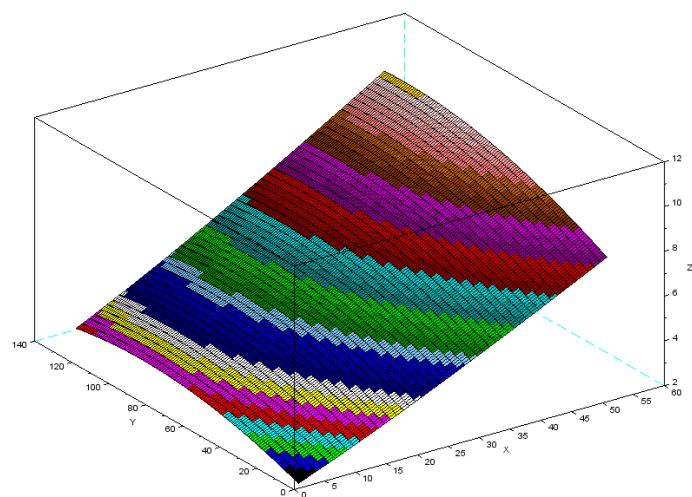


Fig.4 Response surface with  $x_1$  at 0

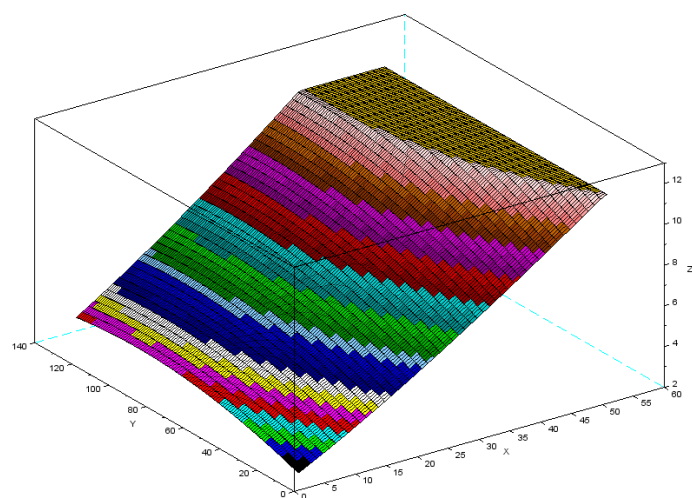


Fig.5 Response surface with  $x_1$  at 25

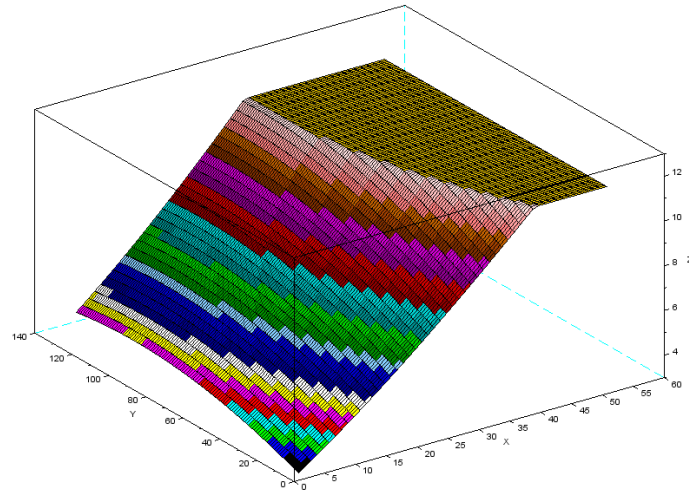


Fig.6 Response surface with  $x_1$  at 50

With the response surfaces we can compare each other. It is observed that in figure 6 with  $x_1 = 50$ , is easier to reach greater numbers for the response variable even if values of  $x_2$  and  $x_3$  are no high, which means for the interested in preparation process of  $V_2O_5$  that using water instead of other materials can get similar results that using more energy or more expensive materials.

## 5.CONCLUSIONS

On this work, a method to apply a symbolic regression alpha-beta algorithm to industrial process was proposed. The method consists in 8 steps to obtain the better and faster results using the algorithm. To apply the method, a mathematical model was generated for the preparation process of  $V_2O_5$ . This model was generated from experimental data obtained in a chemistry laboratory. The compound  $V_2O_5$  has 18 samples on data set. In this case, 4 variables were the input variables and 1 output response. In order to compare the efficiency of the method proposed to apply SR, a comparison was made using similar approaches like linear regression, artificial neural network and genetic programming, the performance of each model was evaluated using statistical metrics and CPU time running the algorithm. To enhance the performance of SR algorithm a DOE was made to get the best configuration parameters, these configuration parameters were set for the proposal method. Finally, the results showed that symbolic regression model have better results on the statistical metrics than other techniques, nevertheless the CPU time was not the best enough, due the calculations that are needed to be performed during execution of SR algorithm.

Finally, the last step of the method is use the data for optimizing and response surfaces were made using the model generated by SR and were compared each other. The comparison of response surfaces showed different perspective for the

---

input variables and to optimize the quantity of materials used for the preparation of  $V_2O_5$ .

Symbolic regression can be used in other chemical process to optimize the process methods; however, for future work other output values on experimental data set of this compound could be used to generate new models in order to have the ideal properties that promote a better performance in catalysis and photocatalysis.

For future work, there are many possibilities to improve the algorithm according to the process of apply. For example, the type of operators can be changed, with other ones that describe better the process. The use of hierarchies to employ equations that represent hidden abstractions. And other type of process can be optimized with symbolic regression alpha-beta to prove that the algorithm works in a broader context and is robust enough.

---

## REFERENCES

- [1] D. R. L. Benyamin Grosman, "Automated nonlinear model predictive control using genetic programming," vol. 26, no. 4–5, pp. 631-640, (2002).
- [2] J. A. a. a. F. S. János Madár, "Genetic Programming for the Identification of Nonlinear Input–Output Models," *Industrial & Engineering Chemistry Research*, vol. 44, no. 9, pp. 3178-3186, (2005).
- [3] G. González-Campos, E. Luévano-Hipólito, L. M. Torres-Treviño and A. Martinez De La Cruz, "Artificial Neural Network for Optimization of a Synthesis Process of  $\gamma$ -Bi<sub>2</sub>MoO<sub>6</sub> Using Surface Response Methodology," *Advances in Computational Intelligence*, vol. 2, pp. 200-210, (2013).
- [4] L. M. Torres-Treviño, "Identification and prediction using symbolic regression alpha-beta: preliminary results," *Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation*, pp. 1367-1372, (2014).
- [5] W. Afzal and R. Torkar, "On the application of genetic programming for software engineering predictive modeling : A systematic review," *Expert Systems with Applications*, vol. 38, no. 9, p. 11984–11997, (2011).
- [6] Guido F. Smits, Ekaterina Vladislavleva and Mark E. Kotanchek, "Scalable Symbolic Regression by Continuous Evolution with Very Small Populations,"

---

*Genetic Programming Theory and Practice VIII. Genetic and Evolutionary Computation*, vol. 8, (2011).

[7] Guido Smits, Arthur Kordon, Katherine Vladislavleva, Elsa Jordaan, Mark Kotanchek.: Variable Selection in Industrial Datasets Using Pareto Genetic Programming. *Genetic Programming Theory and Practice III* pp 79-92, Volume 9, (2006)

[8] Flor Castillo, Kenric Marshall, James Green, Arthur Kordon.: A Methodology for Combining Symbolic Regression and Design of Experiments to Improve Empirical Model Building. *GECCO 2003, LNCS 2724*, pp. 1975–1985, (2003)

[9] Flor Castillo, Arthur Kordon, Guido Smits.: Robust Pareto Front Genetic Programming Parameter Selection Based on Design of Experiments and Industrial Data. *Genetic Programming Theory and Practice IV, Genetic and Evolutionary Computation*, pp 149-166 (2007)

[10] Mark Kotanchek, Guido Smits, Arthur Kordon.: Industrial Strength Genetic Programming. *Genetic Programming Theory and Practice, Genetic Programming Series Volume 6*, pp 239-255 (2003)

[11] Eduardo Oliveira Costa, Aurora Pozo.: A New Approach to Genetic Programming based on Evolution Strategies. *Systems, Man and Cybernetics*, 2006. SMC '06. IEEE International Conference on , vol.6, pp.4832,4837, 8-11 Oct. (2006)

[12] Dervis Karaboga, Celal Ozturk, Nurhan Karaboga, Beyza Gorkemli.: Artificial bee colony programming for symbolic regression. *Information Sciences* 209, 1–15 (2012)



- 
- [13] Vipul K. Dabhi, Sanjay K. Vij.: Empirical Modeling Using Symbolic Regression via Postfix Genetic Programming. Image Information Processing (ICIIP), 2011 International Conference on , vol., no., pp.1,6, 3-5 Nov. (2011)
- [14] Weihua Cai, Arturo Pacheco-Vega, Mihir Sen, K.T. Yang.: Heat transfer correlations by symbolic regression. International Journal of Heat and Mass Transfer 49, 4352–4359 (2006)
- [15] Ming-fang Zhu, Jian-bin Zhang , Yan-ling Ren, Yu Pan, Guang-ping Zhu.: Multivariable Symbolic Regression Based on Gene Expression Programming. iscid, vol. 2, pp.298-301, (2011)
- [16] J.W. Davidson, D.A. Savic , G.A. Walters.: Symbolic and numerical regression: experiments and applications. Information Sciences 150, 95–117 (2003)
- [17] John R. Koza.: Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, Cambridge, MA, USA, (1992)
- [18] P. Barmpalexis , K. Kachrimanis, A. Tsakonas, E. Georgarakis.: Symbolic regression via genetic programming in the optimization of a controlled release pharmaceutical formulation. Chemometrics and Intelligent Laboratory Systems 107, 75–82 (2011)
- [19] Luis M. Torres-Treviño, Indira G. Escamilla-Salazar, Bernardo González-Ortíz, Rolando Praga-Alejo.: An expert system for setting parameters in machining processes Expert Systems with Applications 40 (2013) 6877–6884

---

[20] Torres-Treviño, L.; Escamilla, I.; Gonzalez, B.; Praga-Alejo, R.; Pérez-Villanueva, P: Modeling cutting machining process using symbolic regression alpha -beta. *he International Journal of Advanced Manufacturing Technology* (2012)

[21] Torres-T, L.: Evonorm, a new evolutionary algorithm to continuous optimization. In Workshop on optimization by building and using probabilistic models (OBUPM) genetic and evolutionary computation conference (GECCO) (2006)

[22] Torres-Trevino, L.: Evonorm: easy and effective implementation of estimation of distribution algorithms. *Journal of Research in Computing Science* 23, 75–83 (2006)

[23] Douglas, C.: Montgomery introduction to linear regression analysis. John Wiley and Sons. (2007)

[24] Dong, GJ, Zhang YF, Yuan ZHAO, Yang BAI. Effect of the pH value of precursor solution on the catalytic performance of V<sub>2</sub>O<sub>5</sub>-WO<sub>3</sub>/TiO<sub>2</sub> in the low temperature NH<sub>3</sub>-SCR of NO<sub>x</sub>. *Journal of Fuel Chemistry and Technology*, 42(12): 1455-1463. doi: 10.1016/S1872-5813(15)60003-2. (2014)

[25] Wang J, Yan Z, Liu L, Chen Y, Zhang Z, Wang X. In situ DRIFTS investigation on the SCR of NO with NH<sub>3</sub> over V<sub>2</sub>O<sub>5</sub> catalyst supported by activated semi-coke. *Applied Surface Science*, 313: 660-669. doi: 10.1016/j.apsusc.2014.06.043. (2014)

---

[26] Soyer S, Uzun A, Senkan S, Onal I. A quantum chemical study of nitric oxide reduction by ammonia (SCR reaction) on  $V_2O_5$  catalyst surface. *Catalysis today*, 118(3): 268-278. doi: 10.1016/j.cattod.2006.07.033. (2006)

[27] Aslam M, Ismail IM, Salah, N, Chandrasekaran S, Qamar MT, Hameed A. Evaluation of sunlight induced structural changes and their effect on the photocatalytic activity of  $V_2O_5$  for the degradation of phenols. *Journal of hazardous materials*, 286: 127-135. doi: 10.1016/j.solidstatesciences.2007.12.026. (2015)

[28] Fei HL, Zhou HJ, Wang JG, Sun PC, Ding DT, Chen T H. Synthesis of hollow  $V_2O_5$  microspheres and application to photocatalysis. *Solid State Sciences*, 10(10), 1276-1284. doi: 10.1016/j.solidstatesciences.2007.12.026. (2008)

---

## LIST OF FIGURES

**Figure 1.** Synthesis process of  $V_2O_5$

**Figure 2.** Comparison data between model results

**Figure 3.** Evaluation plot

**Figure 4.** Response surface with  $x_1$  at 0

**Figure 5.** Response surface with  $x_1$  at 25

**Figure 6** Response surface with  $x_1$  at 50

---

## LIST OF TABLES

**Table 1.**  $\alpha$  Operators parameters and mathematical function related

**Table 2.** Experimental data set of  $V_2O_5$

**Table 3.** Level of DOE

**Table 4.** Groups of DOE

**Table 5.** Results of means of each group of DOE

**Table 6.** Statistical metrics results of the best model found of  $V_2O_5$  for linear regression, genetic programming and symbolic regression alpha-beta

## APPENDIX A

Run	Training Data	Population	Selection	Iterations	# Alphas	# Betas	MSE	R2pred	PRESS	CPU time
1	0.8	100	10	150	13	4	0.00385	0.982	0.385	138.471
2	0.8	100	10	150	13	4	0.00254	0.987	0.254	181.123
3	0.8	100	10	150	13	4	0.00204	0.99	0.204	201.273
4	0.8	100	10	150	13	4	0.00255	0.988	0.255	172.567
5	0.8	100	10	150	13	4	0.00324	0.986	0.324	168.481
6	0.8	100	10	150	13	4	0.00703	0.969	0.703	124.611
7	0.8	100	10	150	13	4	0.00721	0.954	0.721	196.366
8	0.8	100	10	150	13	4	0.00328	0.986	0.328	135.784
9	0.8	100	10	150	13	4	0.00391	0.983	0.391	176.306
10	0.8	100	10	150	13	4	0.00651	0.973	0.651	118.579
Mean							0.004216	0.9798	0.4216	161.3561
11	0.8	100	10	500	13	4	0.00511	0.972	0.511	621.6
12	0.8	100	10	500	13	4	0.00722	0.966	0.722	698.72
13	0.8	100	10	500	13	4	0.00604	0.967	0.604	737.551
14	0.8	100	10	500	13	4	0.00153	0.992	0.153	678.582
15	0.8	100	10	500	13	4	0.00481	0.971	0.481	631.451
16	0.8	100	10	500	13	4	0.00261	0.982	0.261	779.718
17	0.8	100	10	500	13	4	0.00448	0.977	0.448	823.382
18	0.8	100	10	500	13	4	0.00117	0.995	0.117	684.265
19	0.8	100	10	500	13	4	0.00187	0.988	0.187	738.284
20	0.8	100	10	500	13	4	0.00296	0.985	0.296	899.675
Mean							0.00378	0.9795	0.378	729.3228

# APPENDIX

21	0.8	100	20	150	13	4	0.00524	0.977	0.524	200.414
22	0.8	100	20	150	13	4	0.00175	0.992	0.175	152.674
23	0.8	100	20	150	13	4	0.00503	0.972	0.503	159.257
24	0.8	100	20	150	13	4	0.00603	0.963	0.603	145.805
25	0.8	100	20	150	13	4	0.00447	0.981	0.447	115.013
26	0.8	100	20	150	13	4	0.00538	0.974	0.538	113.446
27	0.8	100	20	150	13	4	0.00339	0.984	0.339	114.552
28	0.8	100	20	150	13	4	0.00491	0.974	0.491	85.074
29	0.8	100	20	150	13	4	0.00291	0.986	0.291	122.585
30	0.8	100	20	150	13	4	0.00474	0.979	0.474	90.098
Mean							0.00438 5	0.9782	0.4385	129.8918
31	0.8	100	20	500	13	4	0.00373	0.983	0.373	453.444
32	0.8	100	20	500	13	4	0.00214	0.99	0.214	515.934
33	0.8	100	20	500	13	4	0.00419	0.98	0.419	525.564
34	0.8	100	20	500	13	4	0.00223	0.986	0.223	528.94
35	0.8	100	20	500	13	4	0.0041	0.982	0.41	270.103
36	0.8	100	20	500	13	4	0.0043	0.979	0.43	269.703
37	0.8	100	20	500	13	4	0.00708	0.971	0.708	471.594
38	0.8	100	20	500	13	4	0.00609	0.968	0.609	510.56
39	0.8	100	20	500	13	4	0.00378	0.978	0.378	594.858
40	0.8	100	20	500	13	4	0.0041	0.979	0.41	493.65
Mean							0.00417 4	0.9796	0.4174	463.435
41	0.8	100	30	150	13	4	0.0052	0.977	0.52	123.298
42	0.8	100	30	150	13	4	0.00631	0.967	0.631	88.327
43	0.8	100	30	150	13	4	0.0022	0.99	0.22	129.288
44	0.8	100	30	150	13	4	0.00498	0.975	0.498	87.693
45	0.8	100	30	150	13	4	0.00539	0.976	0.539	136.345
46	0.8	100	30	150	13	4	0.00345	0.986	0.345	126.708
47	0.8	100	30	150	13	4	0.00246	0.988	0.246	149.533
48	0.8	100	30	150	13	4	0.0069	0.969	0.69	147.329

## APPENDIX

49	0.8	100	30	150	13	4	0.00361	0.984	0.361	111.741
50	0.8	100	30	150	13	4	0.00393	0.982	0.393	106.212
Mean							0.00444 3	0.9794	0.4443	120.6474
51	0.8	100	30	500	13	4	0.0019	0.991	0.19	335.092
52	0.8	100	30	500	13	4	0.00398	0.978	0.398	437.247
53	0.8	100	30	500	13	4	0.0043	0.985	0.43	362.92
54	0.8	100	30	500	13	4	0.0069	0.965	0.69	491.435
55	0.8	100	30	500	13	4	0.00385	0.98	0.385	445.225
56	0.8	100	30	500	13	4	0.00489	0.973	0.489	391.251
57	0.8	100	30	500	13	4	0.00166	0.991	0.166	392.84
58	0.8	100	30	500	13	4	0.00214	0.989	0.214	353.521
59	0.8	100	30	500	13	4	0.00529	0.974	0.529	328.528
60	0.8	100	30	500	13	4	0.00549	0.974	0.549	340.172
Mean							0.00404	0.98	0.404	387.8231
61	0.8	200	10	150	13	4	0.00421	0.973	0.421	211.956
62	0.8	200	10	150	13	4	0.00303	0.987	0.303	223.955
63	0.8	200	10	150	13	4	0.00172	0.99	0.172	242.569
64	0.8	200	10	150	13	4	0.00241	0.989	0.241	213.18
65	0.8	200	10	150	13	4	0.00292	0.987	0.292	206.297
66	0.8	200	10	150	13	4	0.00697	0.971	0.697	224.216
67	0.8	200	10	150	13	4	0.0043	0.977	0.43	223.994
68	0.8	200	10	150	13	4	0.0034	0.983	0.34	218.608
69	0.8	200	10	150	13	4	0.00255	0.988	0.255	224.101
70	0.8	200	10	150	13	4	0.00098 9	0.995	0.0989	213.675
Mean							0.00325	0.984	0.325	220.2551
71	0.8	200	10	500	13	4	0.00548	0.973	0.548	685.716
72	0.8	200	10	500	13	4	0.00521	0.974	0.521	669.948
73	0.8	200	10	500	13	4	0.00537	0.971	0.537	708.825
74	0.8	200	10	500	13	4	0.00466	0.974	0.466	719.38
75	0.8	200	10	500	13	4	0.00491	0.979	0.491	666.221



## APPENDIX

76	0.8	200	10	500	13	4	0.00124	0.994	0.124	874.499
77	0.8	200	10	500	13	4	0.00402	0.978	0.402	673.456
78	0.8	200	10	500	13	4	0.00605	0.976	0.605	731.105
79	0.8	200	10	500	13	4	0.00201	0.991	0.201	707.201
80	0.8	200	10	500	13	4	0.00375	0.985	0.375	783.533
Mean							0.00427	0.9795	0.427	721.9884
81	0.8	200	20	150	13	4	0.00331	0.983	0.331	227.205
82	0.8	200	20	150	13	4	0.00357	0.983	0.357	411.805
83	0.8	200	20	150	13	4	0.00383	0.983	0.383	220.228
84	0.8	200	20	150	13	4	0.00429	0.974	0.429	208.118
85	0.8	200	20	150	13	4	0.00435	0.981	0.435	205.104
86	0.8	200	20	150	13	4	0.00328	0.986	0.328	203.19
87	0.8	200	20	150	13	4	0.00423	0.983	0.423	216.121
88	0.8	200	20	150	13	4	0.00255	0.986	0.255	215.087
89	0.8	200	20	150	13	4	0.00388	0.982	0.388	204.132
90	0.8	200	20	150	13	4	0.00502	0.979	0.502	204.514
Mean							0.00383 1	0.982	0.3831	231.5504
91	0.8	200	20	500	13	4	0.00531	0.974	0.531	656.586
92	0.8	200	20	500	13	4	0.00499	0.975	0.499	698.608
93	0.8	200	20	500	13	4	0.00468	0.976	0.468	661.698
94	0.8	200	20	500	13	4	0.00197	0.989	0.197	673.456
95	0.8	200	20	500	13	4	0.00089 1	0.996	0.0891	649.955
96	0.8	200	20	500	13	4	0.00447	0.977	0.447	1021.038
97	0.8	200	20	500	13	4	0.00336	0.981	0.336	699.231
98	0.8	200	20	500	13	4	0.00521	0.975	0.521	664.877
99	0.8	200	20	500	13	4	0.00545	0.969	0.545	660.23
100	0.8	200	20	500	13	4	0.00489	0.972	0.489	663.825
Mean							0.00412 2	0.9784	0.4122	704.9504
101	0.8	200	30	150	13	4	0.00399	0.978	0.399	224.421

# APPENDIX

102	0.8	200	30	150	13	4	0.00283	0.985	0.283	226.597
103	0.8	200	30	150	13	4	0.00438	0.97	0.438	311.782
104	0.8	200	30	150	13	4	0.00402	0.98	0.402	467.748
105	0.8	200	30	150	13	4	0.0045	0.97	0.45	238.065
106	0.8	200	30	150	13	4	0.00428	0.982	0.428	233.233
107	0.8	200	30	150	13	4	0.00503	0.976	0.503	244.913
108	0.8	200	30	150	13	4	0.00325	0.982	0.325	163.482
109	0.8	200	30	150	13	4	0.00429	0.98	0.429	274.477
110	0.8	200	30	150	13	4	0.00259	0.986	0.259	271.951
Mean							0.003916	0.9789	0.3916	265.6669
111	0.8	200	30	500	13	4	0.00216	0.989	0.216	1023.71
112	0.8	200	30	500	13	4	0.00366	0.982	0.366	905.585
113	0.8	200	30	500	13	4	0.00387	0.979	0.387	1035.032
114	0.8	200	30	500	13	4	0.00201	0.99	0.201	958.163
115	0.8	200	30	500	13	4	0.00403	0.977	0.403	991.27
116	0.8	200	30	500	13	4	0.00344	0.979	0.344	962.446
117	0.8	200	30	500	13	4	0.00252	0.989	0.252	1017.397
118	0.8	200	30	500	13	4	0.00458	0.979	0.458	1116.642
119	0.8	200	30	500	13	4	0.0037	0.982	0.37	747.471
120	0.8	200	30	500	13	4	0.00611	0.965	0.611	923.296
Mean							0.003608	0.9811	0.3608	968.1012

## APPENDIX B

Coded symbolic regression alpha-beta //Desnormalization

//Interval [0, 1] -> [LMIN, LMAX] function **a=desnormalization(Vn, LMIN, LMAX)** **a=Vn\*LMAX+LMIN\*(1-Vn);** if **a>LMAX**

**a=LMAX;**

end

if **a<LMIN**

**a=LMIN;**

end endfunction //Normalization. Interval [LMIN, LMAX] -> [0, 1] function

**a=normalization(Van, LMIN, LMAX)** **a=(Van-LMIN)/((LMAX-LMIN)+0.000001);** if **a>1**

**a=1;**

end

if **a<0**

**a=0;**

end endfunction function [**posmax, valmax**]=maxp(V) temp=size(V);

NTB=max(temp); **posmax=0;**

**valmax=0;**

for b=1:NTB

if **V(b)>valmax**

**valmax=V(b);**

**posmax=b;**

end

---

```

end endfunction //Algoritmya Evolutionary computation 2004
//Proyect of simplification of algorithms
//Autor: Luis Torres T.
//All rights reserved
//May 2004
//Evolution Strategies and Genetic Algorithms
//Maxp function [posmin, valmin]=minp(V) temp=size(V);
NTB=max(temp); posmin=0;
valmin=1000000000;
for b=1:NTB
    if V(b)<valmin
        valmin=V(b);
        posmin=b;
    end
end endfunction function [M]=shaking(M) [NTPat NTCol]=size(M);
//Shaking the information for i=1:10*NTPat
    pos1 = round(rand()*NTPat+0.5);
    pos2 = round(rand()*NTPat+0.5);

    temp=M(pos1,:);
    M(pos1,:)=M(pos2,:);
    M(pos2,:)=temp;
end

endfunction    function    [DataTrain,    DataVal1,    DataVal2,
MRange]=GenTrainVal(DataExp, percent) //Generation of training and
validation databases [NTPat NTCol]=size(DataExp);
//Shaking the information for i=1:10*NTPat
    pos1 = round(rand()*NTPat+0.5);
    pos2 = round(rand()*NTPat+0.5);

```

---

```

temp=DataExp(pos1,:);
DataExp(pos1,:)=DataExp(pos2,:);
DataExp(pos2,:)=temp;
end //Normalization of information //Normalization of the data;
[NTD NTCols]=size(DataExp); //columns of the data
[NTR2,NTC2]=size(DataExp);
DataN=zeros(NTR2,NTC2);
//Matriz to save every range of the matrix
MRRange=zeros(NTCols,2);//1-Lmin, 2-Lmax
for col=1:NTCols
    Lmax = max(DataExp(:,col)); //+max(DataExp(:,col))*0.1;
    Lmin = min(DataExp(:,col)); //-min(DataExp(:,col))*0.1;
    MRRange(col,1)=Lmin; MRRange(col,2)=Lmax;
    DataN(:,col)=(DataExp(:,col)-Lmin)./(Lmax-Lmin);
end //Generation of Training data base
//percent=0.8;
posel=round(percent*NTPat);
DataTrain = DataN(1:posel,:); //Generation of validation Data Base
//Direct experimental data
DataVal1 = DataN(posel:NTPat,:); //Random NTPat data
DataVal2=zeros(NTPat,NTCols);
for d=1:100
    pos=round(rand()*NTPat+0.5);
    DataVal2(d,:) = DataN(pos,:);
end endfunction    function r=OprAlpha(alphao, k1, k2, x) r=0;
if alphao==1
    r=(k1*x+k2);
end if alphao==2
    r=(k1*x+k2)^2;
end if alphao==3
    r=(k1*x+k2)^3;

```

```

end if alphao==4
     $r = (k_1 * x + k_2 + 0.00000001)^{-1}$ ;
end if alphao==5
     $r = (k_1 * x + k_2 + 0.00000001)^{-2}$ ;
end if alphao==6
     $r = (k_1 * x + k_2 + 0.00000001)^{-3}$ ;
end if alphao==7
     $r = (k_1 * x + k_2)^{1/2}$ ;
end if alphao==8
     $r = (k_1 * x + k_2)^{1/3}$ ;
end if alphao==9
     $r = \exp(k_1 * x + k_2)$ ;
end if alphao==10
     $r = \log(k_1 * x + k_2 + 0.0000000000000001)$ ;
end if alphao==11
     $r = \sin(k_1 * x + k_2)$ ;
end if alphao==12
     $r = \cos(k_1 * x + k_2)$ ;
end if alphao==13
     $r = \sin(k_1 * x + k_2) / (\cos(k_1 * x + k_2) + 0.0000000001)$ ;
end if alphao<1 | alphao>13
     $r = 0$ ;
end endfunction function  $r = \text{OprAlphaV2}(\text{alphao}, k_1, k_2, x)$   $r = 0$ ;
if alphao==1
     $r = (k_1 * x + k_2)$ ;
end if alphao==2
     $r = (k_1 * x + k_2)^2$ ;
end if alphao==3
     $r = (k_1 * x + k_2)^{1/2}$ ;
end if alphao==4
     $r = (k_1 * x + k_2 + 0.00000001)^{-1}$ ;

```

---

```

end if alphao==5
    r=exp(k1*x+k2);
end if alphao==6
    r=log(k1*x+k2+0.0000000000000001);
end if alphao==7
    r=(k1*x+k2+0.00000001)^(-2);
end if alphao==8
    r=(k1*x+k2)^3;
end if alphao==9
    r=(k1*x+k2+0.00000001)^(-3);
end if alphao==10
    r=(k1*x+k2)^(1/3);
end if alphao==11
    r=sin(k1*x+k2);
end if alphao==12
    r=cos(k1*x+k2);
end if alphao==13
    r=sin(k1*x+k2)/(cos(k1*x+k2)+0.0000000001);
end if alphao<1 | alphao>13
    r=0;
end endfunction function r=OprBeta(betao, x1, x2) r=0; if betao==1
    r=(x1+x2);
end if betao==2
    r=(x1-x2);
end if betao==3
    r=(x1*x2);
end if betao==4
    r=(x1/(x2+0.0000000001));
end endfunction //int2bin function [B]=Int2Bin(I, ne)

cc=1;

```

---

```

B=zeros(1,ne);
out=0;
R=I;
while out==0

    R=R/2;
    n=R-floor(R);
    if n>0 then
        B(cc)=1; cc=cc+1;
    else
        B(cc)=0;cc=cc+1;
    end

    R=floor(R);
    if R<=0 then
        out=1;
    end
end

```

```

end

```

```

endfunction //*****
//***** Experimental Data *****
//*****
//+++++
+++++ function [err, Rep]=evalIndiN(Cx, K, Opa, Opb,
DataTrain) [NTRows, NTCols]=size(DataTrain); //Normalization Rep=[];
//Non-codifications
//Two beta levels

```



---

k11 = **K**(1);  
k21 = **K**(2);  
k12 = **K**(3);  
k22 = **K**(4);  
k13 = **K**(5);  
k23 = **K**(6);  
k14 = **K**(7);  
k24 = **K**(8);  
k15 = **K**(9);  
k25 = **K**(10);  
k16 = **K**(11);  
k26 = **K**(12);  
k17 = **K**(13);  
k27 = **K**(14);  
k18 = **K**(15);  
k28 = **K**(16);  
k19 = **K**(17);  
k29 = **K**(18);  
k1A = **K**(19);  
k2A = **K**(20);  
k1B = **K**(21);  
k2B = **K**(22);  
k1C = **K**(23);  
k2C = **K**(24);

alphao1 = **Opa**(1);  
alphao2 = **Opa**(2);  
alphao3 = **Opa**(3);  
alphao4 = **Opa**(4);  
alphao5 = **Opa**(5);

---

```

alphao6 = Opa(6);
alphao7 = Opa(7);
alphao8 = Opa(8);
alphao9 = Opa(9);
alphao10 = Opa(10);
alphao11 = Opa(11);
alphao12 = Opa(12);

```

```

betao1= Opb(1);
betao2= Opb(2);
betao3= Opb(3);
betao4= Opb(4);
betao5= Opb(5);
betao6= Opb(6);
betao7= Opb(7);
betao8= Opb(8);
betao9= Opb(9);
betao10= Opb(10);
betao11= Opb(11);

```

```

//2^(4*3)-1
// l=round(Cx*63); two variable * 3

```

```

l=round(Cx*4095);

```

```

errsum=0;

```

```

for row=1:NTRows

```

```

x1 = DataTrain(row,1);
x2 = DataTrain(row,2);
x3 = DataTrain(row,3);
x4 = DataTrain(row,4);

```

```
[B]=Int2Bin(I,12); //6 for two variables, 12 for four variables
```

```

// r1 = B(1)*OprAlpha(alphao1,k11,k21,x1);
// r2 = B(2)*OprAlpha(alphao2,k12,k22,x1);
// r3 = B(3)*OprAlpha(alphao3,k13,k23,x1);
// r4 = B(4)*OprAlpha(alphao4,k14,k24,x2);
// r5 = B(5)*OprAlpha(alphao5,k15,k25,x2);
// r6 = B(6)*OprAlpha(alphao6,k16,k26,x2);
// r7 = B(7)*OprAlpha(alphao7,k17,k27,x3);
// r8 = B(8)*OprAlpha(alphao8,k18,k28,x3);
// r9 = B(9)*OprAlpha(alphao9,k19,k29,x3);
// r10 = B(10)*OprAlpha(alphao10,k1A,k2A,x4);
// r11 = B(11)*OprAlpha(alphao11,k1B,k2B,x4);
// r12 = B(12)*OprAlpha(alphao12,k1C,k2C,x4);

```

```

r1 = B(1)*OprAlphaV2(alphao1,k11,k21,x1);
r2 = B(2)*OprAlphaV2(alphao2,k12,k22,x1);
r3 = B(3)*OprAlphaV2(alphao3,k13,k23,x1);
r4 = B(4)*OprAlphaV2(alphao4,k14,k24,x2);
r5 = B(5)*OprAlphaV2(alphao5,k15,k25,x2);
r6 = B(6)*OprAlphaV2(alphao6,k16,k26,x2);
r7 = B(7)*OprAlphaV2(alphao7,k17,k27,x3);
r8 = B(8)*OprAlphaV2(alphao8,k18,k28,x3);
r9 = B(9)*OprAlphaV2(alphao9,k19,k29,x3);
r10 = B(10)*OprAlphaV2(alphao10,k1A,k2A,x4);

```

---

```

r11 = B(11)*OprAlphaV2(alphao11,k1B,k2B,x4);
r12 = B(12)*OprAlphaV2(alphao12,k1C,k2C,x4);

```

```

y1 = OprBeta(betao1,r1,r2);
y2 = OprBeta(betao2,y1,r3);
y3 = OprBeta(betao3,y2,r4);
y4 = OprBeta(betao4,y3,r5);
y5 = OprBeta(betao5,y4,r6);
y6 = OprBeta(betao6,y5,r7);
y7 = OprBeta(betao7,y6,r8);
y8 = OprBeta(betao8,y7,r9);
y9 = OprBeta(betao9,y8,r10);
y10 = OprBeta(betao10,y9,r11);
y = OprBeta(betao11,y10,r12);

```

```

//Desnormalization
yd = DataTrain(row,5);
aux=[yd y];
Rep=[Rep; aux];
errsum = (y - yd)^2 + errsum; //Error calculation

```

```

end

```

```

err=(errsum)/NTD;

```

```

//sqrt(sum((Rep(:,1)-Rep(:,2))^2)) ; //Other way to calc error...

```

```

endfunction
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
//Adjustment in the range
//Copyrights LMTT092010 function [P]=adjust(P, M)
[NTI NTPr] = size(P);
for k=1:NTI
    for pr=1:NTPr

        //Checks the inferior range
        if P(k,pr)<M(pr,1)
            P(k,pr)=M(pr,1);
        end
        //Checks the superior range
        if P(k,pr)>M(pr,2)
            P(k,pr)=M(pr,2);
        end

    end
end endfunction //Normalization. Interval [LMIN, LMAX] -> [0, 1] function
a=normalization(Van, LMIN, LMAX) a=(Van-LMIN)/((LMAX-
LMIN)+0.000001); if a>1
    a=1;
end
if a<0
    a=0;
end endfunction //Desnormalization
//Interval [0, 1] -> [LMIN, LMAX] function a=desnormalization(Vn, LMIN,
LMAX) a=Vn*LMAX+LMIN*(1-Vn); if a>LMAX
    a=LMAX;
end

```

---

```

if a<LMIN
    a=LMIN;
end endfunction //Algoritmya Evolutionary computation 2004
//Proyect of simplification of algorithms
//Autor: Luis Torres T.
//All rights reserved
//May 2004
//Evolution Strategies and Genetic Algorithms
//Maxp function [posmax, valmax]=maxp(V) temp=size(V);
NTB=max(temp); posmax=0;
valmax=0;
for b=1:NTB
    if V(b)>valmax
        valmax=V(b);
        posmax=b;
    end
end endfunction //Algoritmya Evolutionary computation 2004
//Proyect of simplification of algorithms
//Autor: Luis Torres T.
//All rights reserved
//May 2004
//Evolution Strategies and Genetic Algorithms
//Maxp function [posmin, valmin]=minp(V) temp=size(V);
NTB=max(temp); posmin=0;
valmin=1000000000;
for b=1:NTB
    if V(b)<valmin
        valmin=V(b);
        posmin=b;
    end
end endfunction function [Meann, Stdn]=CalculationEvonorm(PS)

```

---

```

//Calculation of the EvoLogNorm
//D is a matrix of NTISxNTPr

[NTIS NTPr]=size(PS);
Meann=zeros(1,NTPr);
Stdn=zeros(1,NTPr);
D=zeros(1,NTIS);
NTD=NTIS;
for pr=1:NTPr
    D=PS(:,pr);
    Meann(pr) = sum(D)/NTD;

    Stdn(pr) = sqrt(sum((D - Meann(pr)).^2)/NTD);

end

endfunction //A proposal for a new evolutionary algorithm
// Evola heuristics
//Generation of a population function [P]=GenEvonorm(Meann, Stdn,
Imax, NTI)

NTPr=max(size(Meann));

P=zeros(NTI,NTPr); for k=1:NTI
    for pr=1:NTPr
        Nc = sum(rand(1,12))-6; //Estimation of the normal random variable
        if rand()>0.5
            P(k,pr) = Meann(pr)+(Stdn(pr)+0.00000052)*Nc;
        else
            P(k,pr) = Imax(pr)+(Stdn(pr)+0.00000052)*Nc;
        end
    end
end

```

---

---

```

    end
end

endfunction function [PS]=selectiondet(P, FE, NTIS) for k=1:NTIS
    [pos,val]=maxp(FE);
    PS(k,:)=P(pos,:);
    FE(pos)=-10000000;
end

endfunction //*****
//***** C A S O I *****
//***** function [Sol, err,
Report, Rep, msres, R2pred, PRESS]=SRABcorrosionNMQ(NTI, NTIS,
NTGen)

    //v205 //(h20)    (eg)  (t US) Temp (°C)    area superficial
ExpData=[0  50  0  400  7.7419
0  50  60  400  12.1860
0  50  120  400  10.3980
0  50  0  500  4.1292
0  50  60  500  1.9645
0  50  120  500  2.3174
25  25  0  400  7.4281
25  25  60  400  8.0970
25  25  120  400  9.7480
25  25  0  500  3.9813
25  25  60  500  4.2494
25  25  120  500  4.5984
50  0  0  400  2.9952
50  0  60  400  4.2209
50  0  120  400  4.8132

```



---

```

50 0 0 500 2.9073
50 0 60 500 4.6259
50 0 120 500 3.8645
]; //Normalization [DataTrain,DataVal1,DataVal2,MRange] =
GenTrainVal(ExpData,0.8); Table=DataTrain; [NTD NTCol]=size(Table);
//Pk changes, POa, POB stay without change //Evonorm structures for k
adjustements

//V2
NTPr=48;//six k parameters, three alpha operators and two beta
operators P=zeros(NTI,NTPr);

Report=[];

//Margin per parameter
MR=zeros(NTPr,2);
//Constanst limits

MR(:,1)=0; //minimum
MR(:,2)=1; //maximum

//Generate a new population
//V2
for k=1:NTI
    for pr=1:NTPr
        P(k,pr)=desnormalization(rand(),MR(pr,1),MR(pr,2));
    end
end

//Auxiliar variables
miny=1000000000000000000;

```

---

---

```

maxy=-10000000000000000;

FE=zeros(1,NTI); //Evaluation per individual
lmax=P(1,:); //Best individual found


//Principal cycle begin here

for cycle=1:NTGen

    // Evaluation
    for k=1:NTI    //Decoding
        Cx=P(k,1);
        Kp=P(k,2:25);
        Opa=round(13*P(k,26:37)+0.5);
        Opb=round(4*P(k,38:48)+0.5);

        [err, Rep] = evalIndiN(Cx,Kp,Opa,Opb,DataTrain);
        y=err;

        if y > maxy
            maxy=y;
        end    if y<miny
            miny=y;

            //lmax=Kp;

            //V2
            lmax=P(k,:);
            Sol=[Kp Opa Opb];
        end
    end
end

```

---

```

//Minimization
aux = (sum(Opa)/(13*12) + sum(Opb)/(4*11))/2;
FE(k) = 0.5*(1 - normalization(y, miny, maxy)) + 0.5*(1-aux);
end //of k

//Selection

[PS]=selectiondet(P,FE,NTIS);

//Generation

//Normalization of PS-PSN
// PSN=PS;
// for k=1:NTIS
//   for pr=1:NTPr
//     PSN(k,pr)=normalization(PS(k,pr),MR(pr,1),MR(pr,2));
//   end
// end

//Estimation of parameters
[Meann,StdN]=CalculationEvonorm(PS);

//Using the heuristics
[P]=GenEvonorm(Meann,StdN,lmax,NTI);

//Desnormalization of PN->P
//   for k=1:NTIS
//     for pr=1:NTPr

```

---

```
// P(k,pr)=desnormalization(PN(k,pr),MR(pr,1),MR(pr,2));
// end
// end
```

```
Report=[Report miny];
```

```
//Adjust for corresponding limits
```

```
[P]=adjust(P,MR);
```

```
end
```

```
Table=DataVal2;
```

```
[NTD NTCol]=size(Table); Cx=Imax(1);
```

```
Kp=Imax(2:25);
```

```
Opa=round(13*Imax(26:37)+0.5);
```

```
Opb=round(4*Imax(38:48)+0.5);
```

```
I=round(Cx*4095);
```

```
[B]=Int2Bin(I,12);
```

```
Sol=[B Kp Opa Opb];
```

```
[err, Rep] = evalIndiN(Cx,Kp,Opa,Opb,DataTrain);
```

```
//PRESS and R2pred considering DataVal2
```

```
Y=DataVal2(:,5);
```

```
//Studentized
```

```
//Calculate H???
```

```
//msres=sum(VErr.^2)/(NTD-4);
```

```
//ds=VErr/sqrt(msres(1-Hii));
```

---

```

//PRESS and R2 prediction
VErr=Rep(:,1)-Rep(:,2);
PRESS=sum(VErr.^2);
SST=Y'*Y-sum(Y.^2)/NTD;
R2pred=1-PRESS/SST;
//Standardized

msres=sum(VErr.^2)/(NTD);
ds=VErr/sqrt(msres); //ds with high value, a potential outlier (non tipic
value)                                     endfunction
//XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
//Decode equations function [R]=ExtractAlpha(opalpha, k1, k2, varx)

ks=string(k1);

r=strcat(['',ks]);
ks=string(k2);

xs='';

if varx==1 then
    xs='x1';
end

if varx==2 then

```

---

---

```
    xs='x2';  
end
```

```
if varx==3 then  
    xs='x3';  
end
```

```
if varx==4 then  
    xs='x1';  
end
```

```
if varx==5 then  
    xs='x2';  
end
```

```
if varx==6 then  
    xs='x3';  
end
```

```
if varx==7 then  
    xs='x1';  
end
```

```
if varx==8 then  
    xs='x2';  
end
```

```
if varx==9 then  
    xs='x3';  
end
```

---

```
if opalpha==1 then
    R=strcat(['(,r,xs,+',ks,')']);
end

if opalpha==2 then
    R=strcat(['exp(',r,xs,+',ks,')']);
end

if opalpha==3 then
    R=strcat(['(,r,xs,+',ks,')^2']);

end

if opalpha==4 then
    R=strcat(['log(',r,xs,+',ks,')']);

end

if opalpha==5 then
    R=strcat(['(,r,xs,+',ks,')^(-1)']);
end

if opalpha==6 then
    R=strcat(['(,r,xs,+',ks,')^(-2)']);
end

if opalpha==7 then
    R= strcat(['(,r,xs,+',ks,')^(1/2)']);
end
```

```
if opalpha==8 then
    R=strcat(['(,r,xs,+',ks,')^3']);
end

if opalpha==9 then
    R=strcat(['(,r,xs,+',ks,')^(-3)']);

end

if opalpha==10 then
    R=strcat(['(,r,xs,+',ks,')^(1/3)']);
end

if opalpha==11 then
    R=strcat(['sin(',r,xs,+',ks,')']);
end

if opalpha==12 then
    R=strcat(['cos(',r,xs,+',ks,')']);
end

if opalpha==13 then
    R=strcat(['tan(',r,xs,+',ks,')']);
end

if opalpha<1 | opalpha>13
    R='0';
end
```



```
endfunction
function [R]=ExtractAlphasim(opalpha, k1, k2, varx)

r="";

xs="";

if varx<=3 then
    xs='x1';
    r='k11';ks='k21';
end

if varx>3 & varx<=6 then
    xs='x2';
    r='k12';ks='k22';
end

if varx>6 & varx<=9 then
    xs='x3';
    r='k13';ks='k23';
end

if varx>9 & varx<=12 then
    xs='x4';
    r='k14';ks='k24';
```

---

end

if **varx**>13 & **varx**<=15 then

    xs='x5';

    r='k15';ks='k25';

end

if **opalpha**==1 then

**R**=strcat(['(',r,xs,'+',ks,')']);

end

if **opalpha**==2 then

**R**=strcat(['(',r,xs,'+',ks,')^2']);

end

if **opalpha**==3 then

**R**=strcat(['(',r,xs,'+',ks,')^3']);

end

if **opalpha**==4 then

**R**=strcat(['(',r,xs,'+',ks,')^(-1)']);

end

if **opalpha**==5 then

**R**=strcat(['(',r,xs,'+',ks,')^(-2)']);

end

if **opalpha**==6 then

**R**=strcat(['(',r,xs,'+',ks,')^(-3)']);

---

end

if **opalpha**==7 then

**R**=strcat(['(',r,xs,'+',ks,')^(1/2)']);

end

if **opalpha**==8 then

**R**=strcat(['(',r,xs,'+',ks,')^(1/3)']);

end

if **opalpha**==9 then

**R**=strcat(['exp(',r,xs,'+',ks,')']);

end

if **opalpha**==10 then

**R**=strcat(['log(',r,xs,'+',ks,')']);

end

if **opalpha**==11 then

**R**=strcat(['sin(',r,xs,'+',ks,')']);

end

if **opalpha**==12 then

**R**=strcat(['cos(',r,xs,'+',ks,')']);

end

if **opalpha**==13 then

**R**=strcat(['tan(',r,xs,'+',ks,')']);

end

---

```

    if opalpha<1 | opalpha>13
        R='0';
    end

```

```

endfunction function [EquationR]=DecodeEqu(Sol)

```

```

    B=Sol(1:12);
    K=Sol(13:36);
    opA = Sol(37:48);
    opB = Sol(49:59);

```

```

    TB=max(size(B));
    cbeta=1;
    R="";
    for b=1:TB

```

```

        opalpha=opA(b);
        k1=K(2*b-1);
        k2=K(2*b);
        if B(b)==1 then

```

```

            [rn]=ExtractAlpha(opalpha,k1,k2,b);

```

```

        else
            rn='0';

```

```

        end

```

---

```
if b>1 then
    if opB(cbeta)==1 then
        betas='+';
    end

    if opB(cbeta)==2 then
        betas='-';
    end

    if opB(cbeta)==3 then
        betas='*';
    end

    if opB(cbeta)==4 then
        betas='/';
    end

    if opB(cbeta)>4 then
        betas='?';
    end

    cbeta=cbeta+1;

    R=strcat(['(',R,betas,rn,')']);

    // R=strcat([R,betas,rn]);
```

---

---

```

        else

            R=rn;

        end

    end

    EquationR=strcat(['y(x1,x2,x3,x4,x5)=' ,R]);

endfunction function [EquationR]=DecodeEquisim(Sol)

    B=Sol(1:15);
    K=Sol(16:45);
    opA = Sol(46:60);
    opB = Sol(61:74);

    TB=max(size(B));
    cbeta=1;
    R="";
    for b=1:TB

        opalpha=opA(b);
        k1=K(2*b-1);
        k2=K(2*b);
        if B(b)==1 then

```

---

```
[rn]=ExtractAlphasim(opalpha,k1,k2,b);
```

```
else
```

```
rn='0';
```

```
end
```

```
if b>1 then
```

```
    if opB(cbeta)==1 then
```

```
        betas='+';
```

```
    end
```

```
    if opB(cbeta)==2 then
```

```
        betas='-';
```

```
    end
```

```
    if opB(cbeta)==3 then
```

```
        betas='*';
```

```
    end
```

```
    if opB(cbeta)==4 then
```

```
        betas='/';
```

```
    end
```

```
    if opB(cbeta)>4 then
```

```
        betas='?';
```

```
    end
```

---

```

        cbeta=cbeta+1;

        R=strcat([' ',R,betas,rn,']);
        // pause

        //R=strcat([R,betas,rn]);

    else

        R=rn;

    end

end

EquationR=strcat(['y(x1,x2,x3,x4,x5)=' ,R]);

endfunction function [Table, ReportS, ReportY]=ExperTable()

Table=[];
ReportS=[];
ReportY=[];
for exper=1:2

tic();[Sol,err,Report,Rep,msres,R2pred,PRESS]=SRABcorrosionNMO(20
0,10,150);a=toc();
    aux=[msres R2pred PRESS a];
    Table=[Table; aux];

```

---



---

```

    ReportS=[ReportS;Sol];
    ReportY=[ReportY Rep];
end

endfunction function [Ip, J, Fo, s, Rep, Interv]=IntervalPred(X0)
ExpData=[0 50 0 400 7.7419
0 50 60 400 12.1860
0 50 120 400 10.3980
0 50 0 500 4.1292
0 50 60 500 1.9645
0 50 120 500 2.3174
25 25 0 400 7.4281
25 25 60 400 8.0970
25 25 120 400 9.7480
25 25 0 500 3.9813
25 25 60 500 4.2494
25 25 120 500 4.5984
50 0 0 400 2.9952
50 0 60 400 4.2209
50 0 120 400 4.8132
50 0 0 500 2.9073
50 0 60 500 4.6259
50 0 120 500 3.8645
]; //Normalization
[DataTrain,DataVal1,DataVal2,MRange] = GenTrainVal(ExpData,0.8);
[NTD NTCol]=size(DataTrain); J=zeros(NTD,4);
Rep=[];
Ip=[0 0 0];
suma=0; for d=1:NTD
    x1=DataTrain(d,1);
    x2=DataTrain(d,2);

```

---

---

```

x3=DataTrain(d,3);
k13=0.2663285;
k23=-0.6372761;
k14=-0.5851344;
k24=0.5818615;
    //Original function
y=(k13*x3+k23)^2+(k14*x2+k24);

aux=[DataTrain(d,4) y];
Rep=[Rep; aux];

//Jacobian

suma = suma + (DataTrain(d,4)-y)^2;

Fd=[2*(k13*x3+k23)*x3 2*(k13*x3+k23) x2 1];

J(d,:) = Fd;
end s=sqrt(suma/(NTD-4)); //four parameters k's //Xo normalization
x1n = normalization(X(1),MRange(1,1),MRange(1,2));
x2n = normalization(X(2),MRange(2,1),MRange(2,2));
x3n = normalization(X(3),MRange(3,1),MRange(3,2)); //Original function
with Xo
yo=(k13*x3n+k23)^2+(k14*x2n+k24);
yr = desnormalization(yo,MRange(4,1),MRange(4,2));

//Fo evaluated with Xo
Fo = [2*(k13*x3n+k23)*x3n 2*(k13*x3n+k23) x2n 1];

```

---

---

```

tstud=2.776;
Interv=tstud*s*sqrt(1+Fo*inv(J'*J)*Fo);

//Desnormalization of intervals
intervr = desnormalization(Interv,MRange(4,1),MRange(4,2));

a=intervr-yr;
b=intervr+yr

lp=[ a    yr  b]; endfunction          function [TableSRAB,
SolSR]=experimentsSR()

TableSRAB=[];SolSR=[];
for exper=1:10

[Sol,err,Report,Rep,msres,R2pred,PRESS]=SRABcorrosionNMQ(500,50,
200);

    aux=[msres R2pred PRESS];
    TableSRAB=[TableSRAB; aux];
    SolSR=[SolSR; Sol];
end
endfunction

```

## APPENDIX C

Scientific production

# Modeling Synthesis Processes of Photocatalysts Using Symbolic Regression $\alpha$ - $\beta$

G. González-Campos, L.M. Torres-Treviño, E. Luévano-Hipólito,  
and A. Martínez-de la Cruz

Facultad de Ingeniería Mecánica y Eléctrica  
Universidad Autónoma de Nuevo León  
San Nicolás de los Garza, México

guillermogc53@gmail.com, edithlue@gmail.com, luis.torres.ciuidit@gmail.com, azael70@gmail.com

**Abstract**— Symbolic regression is an application of genetic programming and is used for modeling different dynamic processes. Industrial processes problems have been solved using this technique. In this work a symbolic regression algorithm is used for modeling the synthesis process of the oxides  $\text{Bi}_2\text{MoO}_6$  and  $\text{V}_2\text{O}_5$  in order to provide a model. These oxides are used on heterogeneous photocatalysis. Genetic programming, artificial neural network and linear regression are compared with symbolic regression models using statistics metrics to evaluate them.

**Keywords**— Symbolic regression; genetic programming; photocatalysis; industrial modeling.

## I. INTRODUCTION

Modeling industrial and dynamic processes is an opportunity area where new techniques are being studied. There is a new application in genetic programming (GP) called symbolic regression (SR) and is used for modeling on industrial processes. There are several forms to generate models; mathematical models are not black boxes [1]. Artificial neural networks (ANN) could be a good model for optimizing [2]; however, they are black boxes where an explicit formulation about the correlation between variables and effects on output response is not evident [1]. In chemical processes, Repetitions of reactions on chemical experiments are expensive and spent time is very important, even on industry or academic research. This is the importance of creating models on chemical processes. In this paper symbolic regression models are proposed for two different compounds used on heterogeneous photocatalysis;  $\text{Bi}_2\text{MoO}_6$  [3] and  $\text{V}_2\text{O}_5$ .

## II. RELATED WORKS

Recently some authors have been worked with SR; Guido F. Smits et al use SR with the aim of obtain maximum scalability to architectures with a very large number of processors in a process of a distillation tower with 23 inputs and 5000 records [4]. They present a GP variant; Pareto GP, which exploits the Pareto front to dramatically speed the

symbolic regression solution evolution as well as explicitly exploit the complexity-performance trade-off [5]. Furthermore, in other work they give an overview of the importance of variable selection to build robust models from industrial datasets [6]. Flor Castillo et al, use SR and a design of experiments (DOE) to obtain the maximum data utilization when extrapolation is necessary [7] and also use SR based on Pareto front [8]. An important work of Mark Kotanchek et al, summarize their experience in industrial application of genetic programming to empirical modeling and transfer key learnings with respect to real-world application.[9]. Eduardo Oliveria et al, changes the basic behavior of the method of SR adding some concepts of evolution strategies (ES) obtaining good results. [10].

Other important work was made by Birkan Can et al, where they made a comparative analysis of GP and ANN for meta modeling of discrete-event simulation models, [11]. Xiong Shengwu et al, apply GP to SR problem and propose a new GP representation and algorithm that can be applied to both continuous function's regression and discontinuous function's regression [12]. Luis M. Torres-Treviño et al, proposed an hybrid system for setting machining parameters from experimental data using SR alpha-beta to build mathematical models [1]. Dervis Karaboga et al made a work where a set of SR benchmark problems were solved using artificial bee colony programming and then its performance was compared with the very well known method evolving computer programs, GP [13]. Vipul K. Dabhi et al, explored the suitability of ANN and SR to solve empirical modeling problems and conclude that SR can deal efficiently with these problems [14].

Weihua Cai et al describe a methodology that uses SR to extract correlations from heat transfer measurements by searching for both the form of the correlation equation and the constants in it that enable the closest fit to experimental data [15]. T.L. Lew et al, extend the class of possible models

considerably by carrying out a general SR using GP approach [16]. Ming-fang Zhu et al, present a method for multivariable SR modeling and predicting, based on gene expression programming, a recently proposed evolutionary computation technique, furthermore they give an example to explain this technique. Experiment results show that the model set up is better than statistical linear regression techniques [17].

J.W. Davidson et al describes a new method for creating polynomial regression models and is compared with stepwise regression and SR using three example problems[18], this new method includes some changes on the basic genetic programming algorithm first proposed by Koza [19]. P. Bampalexis et al use SR via GP in the optimization of a pharmaceutical zerorder release matrix table, and its predictive performance was compared with ANN models [20].

Finally Wasif Afzal et al, investigate the evidence for SR using GP being an effective method to prediction and estimation in software engineering, when with regression/machine learning models. They used 23 primary studies from 1995 to 2008, the results show that SR using GP has been applied in three domains within software engineering predictive modeling; software quality classification, software cost/effort/size estimation and software fault prediction/software reliability growth modeling [21].

### III. HETEROGENEOUS PHOTOCATALYSIS

During the last decade the heterogeneous photocatalysis has been positioned as an effective technology to solve environmental problems [22, 23]. The chemical reactions involved in heterogeneous photocatalysis are highly attractive because they take place at ambient conditions (T=298K, P=1bar) and the photocatalyst can be used almost indefinitely. For these reasons heterogeneous photocatalysis has been applied to the treatment of wastewaters and indoor purification. Titanium dioxide (TiO<sub>2</sub>) is one of the most important semiconductor photocatalyst due to its high photocatalytic activity under UV radiation, low cost, and stability to corrosion processes of its anatase polymorph. However, its relative wide energy band gap of 3.2 eV limits further applications of the material in the visible-light region. In the search of semiconductor materials with photocatalytic activity under visible-light irradiation, important efforts have been carried out since the last decade. For example, the TiO<sub>2</sub> anatase polymorph has been doped with some metals and no metals in order to increase its absorption in the visible range [24]. In another approach, several authors have proposed alternative oxides than traditional TiO<sub>2</sub> with high photocatalytic activity under visible-light irradiation such as: V<sub>2</sub>O<sub>5</sub>, WO<sub>3</sub>, In<sub>1-x</sub>NixTaO<sub>4</sub>, CaIn<sub>2</sub>O<sub>6</sub>, InVO<sub>4</sub>, BiVO<sub>4</sub> and Bi<sub>2</sub>MoO<sub>6</sub> [25,26,27,28,29,30,31]. Specifically bismuth molybdate has an Aurivillius type structure that has atoms of molybdenum in octahedral positions which is interesting from the point of view in photocatalysis. On the other hand, V<sub>2</sub>O<sub>5</sub> is an efficient catalyst due to its strong acidity, high thermal stability and low oxidation potential compared with other catalysts [32]. The efficiency of synthesis

process depend of several parameters such as crystallinity, surface area and thermal treatment. These parameters can be modified depending on the experimental conditions of synthesis to obtain the semiconductor oxide. In this work the oxides V<sub>2</sub>O<sub>5</sub> and Bi<sub>2</sub>MoO<sub>6</sub> were selected for modelling its synthesis process using SR.

### IV. METHODOLOGY: MODEL GENERATION BY SYMBOLIC REGRESSION ALPHA-BETA

In this work a symbolic regression  $\alpha$ - $\beta$  approach is used [33]. A mathematical equation is represented by the combination of  $\alpha$  and  $\beta$  operators. An  $\alpha$  operators is defined as a function that requires only one argument and applies only one mathematical operation. Considering a review of several mathematical models of real processes, 13 operations are chosen as  $\alpha$  operators (see Table 1).

An  $\alpha$  operator uses two real number parameters called  $k_1$  and  $k_2$  and an integer that describes the mathematical operation. The  $\alpha$  operator is defined as follows:

$$Opt_{\alpha}(x, k_1, k_2) = \alpha(k_1 * x + k_2) \quad (1)$$

where  $x$  is an input variable and  $\alpha$  is an operation. Depending of the  $\alpha$  operator selected, a specific mathematical operation that requires only one argument is executed; e.g., if  $\alpha = 1$  then the operation made is  $(k_1 * x + k_2)$ , if  $\alpha = 13$  then the operation made is  $\tan(k_1 * x + k_2)$ . The  $\alpha$  operator is an integer number and its value determinate a specific mathematical operation described in Table 1. A  $\beta$  operator is defined as a function that require two arguments and makes the four basic arithmetic operations  $\beta = c$  so a  $\beta$  operator equal to 1 imply the plus operator or  $\beta(a, b) = a + b$ , and  $\beta(a, b) = a/b$  if  $\beta = 4$ .

**Table 1.**  $\alpha$  Operators parameters and its related mathematical function

$\alpha$ Operator	Mathematical operation
1	$(k_1 x + k_2)$
2	$(k_1 x + k_2)^2$
3	$(k_1 x + k_2)^3$
4	$(k_1 x + k_2)^{-1}$
5	$(k_1 x + k_2)^{-2}$
6	$(k_1 x + k_2)^{-3}$
7	$(k_1 x + k_2)^{1/2}$
8	$(k_1 x + k_2)^{1/3}$
9	$\exp(k_1 x + k_2)$
10	$\log(k_1 x + k_2)$
11	$\sin(k_1 x + k_2)$
12	$\cos(k_1 x + k_2)$
13	$\tan(k_1 x + k_2)$

### A. Representation of operators

By means of  $\alpha$ - $\beta$  operators several configurations can be established. A basic configuration can be defined when an  $\alpha$  operator is assigned per input variable then an  $\beta$  operator is used to connect two  $\alpha$  operators (2). Usually, a simple configuration in majority of the cases is enough for the regression.

$$y = \beta_{n-1}(\dots \beta_2(\beta_1(a_1), a_2(x_2)), \dots a_n(x_n)) \quad (2)$$

The representation required is a real vector with  $n$  element where  $n$  is equal to the number of  $\alpha$  operators and  $k$  parameters plus  $\beta$  operators. Using one  $\alpha$  operator per variable and connect them by  $\beta$  operators, the number of parameters is given by the number of  $\alpha$  operators, the number of  $\beta$  operators and  $k$  parameters (two per  $\alpha$  operator). In a basic structure is  $\alpha + \beta + 2 * \alpha$ , because  $\beta = \alpha - 1$ , and  $\alpha =$  number of variables ( $N_v$ ) then the number of parameters is  $N_v + (N_v - 1) + 2 * N_v$ . A normalized real vector can be used to represent operators and  $k$  parameters, but  $\alpha$  and  $\beta$  operators are integers, so is required the following formulation to get its value:

$$\alpha = \lceil V(i) * 13 + 0.5 \rceil \quad (3)$$

$$\beta = \lceil V(i) * 4 + 0.5 \rceil \quad (4)$$

where  $\lceil \cdot \rceil$  is the ceiling function. There are 13  $\alpha$  operators defined in Table 1 and 4  $\beta$  operators (basic algebraic operations). Consider the following example: the vector of parameters is  $V = [0.854 \ 0.124 \ 0.456 \ 0.232 \ 0.987 \ 0.654 \ 0.0234]$  for two variables, so, two  $\alpha$  operators, one  $\beta$  operator, and four  $k$  parameters are represented. Decoding is as follows: first,  $\alpha$  and  $\beta$  operators are decoded using the first elements of the vector, then the  $k$  parameters are represented on the rest of the elements.

$$\begin{aligned} \alpha_1 &= \lceil V(1) * 13 + 0.5 \rceil = \lceil (0.854 * 13 + 0.5) \rceil = 12 \text{ this represents a} \\ &\quad \text{cos function,} \\ \alpha_2 &= \lceil V(2) * 13 + 0.5 \rceil = \lceil (0.124 * 13 + 0.5) \rceil = 3 \text{ this represents a} \\ &\quad \text{cubic exponential,} \\ \beta_1 &= \lceil V(3) * 4 + 0.5 \rceil = \lceil (0.456 * 4 + 0.5) \rceil = 3 \text{ this represents a} \\ &\quad \text{multiplication,} \\ k_{11} &= V(4) = 0.232, \\ k_{21} &= V(5) = 0.987, \\ k_{12} &= V(6) = 0.654, \\ k_{22} &= V(7) = 0.0234. \end{aligned}$$

The formulation represented is:

$$y = \cos(k_{11}x_1 + k_{12}) * (k_{21}x_2 + k_{22})^3 \quad (5)$$

In this work, Evonorm is used to solve the problem of selection, the suitable parameters ( $k$ 's), and integers to define  $\alpha$  and  $\beta$  operations. During the last decade the heterogeneous photocatalysis has been positioned.

### B. Evolutionary algorithm Evonorm

Evonorm is an easy way to implement an estimation of distribution algorithm [34, 35]. As an evolutionary algorithm selection of new individuals and the generation of a new population is used; however, the crossover and mutation mechanism is substituted by an estimation of parameters of a normal distribution function. The following steps are used in Evonorm:

1. Evaluation of a population P.
2. Deterministic selection of individuals from P to P S.
3. Generation of a new population using P S

A population P is a matrix of size  $I_p$  (total of individuals) and  $D_p$  (total of decision variables). A solution is a set of decision variables, and this set is represented as a real vector. Every row of the population P represents a set of decision variables. The selection mechanism is deterministic because the fittest individuals are selected. Usually the number of selected individuals is lower than the number of the original population, usually a 20 or 10 % of the total population. A random variable with normal distribution is estimated per decision variable, so a marginal distribution function is used. Two parameters are estimated, the mean and the standard deviation, that are determined using the values of the selected individuals. The population of selected individuals is a matrix  $P_s$  of size  $I_s$  (total of individuals selected) and  $D_s$ . The Eqs. 6 and 7 are used to calculate the mean and standard deviation, considering every vector of population  $P_s$ .

$$\mu_{pr} = \sum_{k=1}^{I_s} (P_{s_{pr,k}}) / I_s \quad (6)$$

$$\sigma_{pr} = \sqrt{\left( \sum_{k=1}^{I_s} (P_{s_{pr,k}} - \mu_{pr})^2 \right) / I_s} \quad (7)$$

where  $pr = 1, \dots, D_s$ .

A new population is generated using the estimated normal random variables. This is a stochastic process; however, a heuristic is used to maintain equilibrium between exploration and exploitation, so new solutions can be found not necessarily near of the mean calculated. The best solution found  $I_x$  at the moment is involved in the generation, so, in the 50 % of the times, the mean is used in the calculations and, on the other 50 % of the time, the best solution found  $I_x$  is used as a mean as is shown in the following equation:

$$P_{i,pr} = \begin{cases} N(\mu_{pr}, \sigma_{pr}) & U() > 0.5 \\ N(I_{x_{pr}}, \sigma_{pr}) & \text{otherwise} \end{cases} \quad (8)$$



The random variable  $U()$  has a uniform distribution function;  $N()$  is a random variable with a normal distribution function.

### C. Residual analysis

One effective way to validate a regression model is to collect new experimental data to determine how well the model performs in practice [36]. The most simple measure is the residual calculated as the difference ( $e(i)$ ) between new observations made by the response of the process ( $y(i)$ ) and predicted response generated by the regression model made ( $\hat{y}(i)$ ), (Eq. 9).

$$e(i) = y(i) - \hat{y}(i) \quad (9)$$

The prediction error sum of squares (PRESS) is a measure of how well a model works to predict new data. Usually, a small value of PRESS is desirable (Eq. 10). In this case, the PRESS is obtained using cross validation.

$$PRESS = \sum_{i=1}^n (y(i) - \hat{y}(i))^2 \quad (10)$$

The percentage of variability  $R^2_{pred}$  is a measurement for indicating the efficiency of the model to predict new observations. A value near one is desirable on this indicator (Eq. 11).

$$R^2_{pred} = 1 - \frac{\sum_{i=1}^n (y(i) - \hat{y}(i))^2}{y'y - (\sum_{i=1}^n y(i))^2} \quad (11)$$

## V. A CASE OF APPLICATION: SYNTHESIS PROCESS OF PHOTOCATALYSTS $\text{Bi}_2\text{MoO}_6$ AND $\text{V}_2\text{O}_5$

An experimental design was made considering two different oxides for photocatalysis:  $\text{Bi}_2\text{MoO}_6$  and  $\text{V}_2\text{O}_5$ . First  $\text{Bi}_2\text{MoO}_6$  material was synthesized by co-precipitation assisted with ultrasonic radiation. For this purpose two aqueous solutions were prepared. In the first one, 9.1508 g of  $\text{Bi}(\text{NO}_3)_3 \cdot 5\text{H}_2\text{O}$  [Aldrich, 99.99%] were dissolved in 100 mL of diluted  $\text{HNO}_3$ . The second one was prepared by dissolving 1.7359 g of  $(\text{NH}_4)_6\text{Mo}_7\text{O}_{24} \cdot 4\text{H}_2\text{O}$  [Aldrich, 99.99%] in 100 mL of distilled water. The bismuth nitrate solution was added dropwise to the molybdate solution with a vigorous stirring. When the mix was reached, the pH of the solution was adjusted at 3 by using a 2M  $\text{NH}_4\text{OH}$  solution. Afterward, the solution was exposed to ultrasonic radiation in a water bath at 60°C (70W, 42 kHz). The resulting yellow suspension was maintained at 100°C to promote a slow evaporation of the solvent. The yellow powder obtained after this thermal treatment was used as precursor of  $\text{Bi}_2\text{MoO}_6$ . A slow thermal treatment of 10°C/min in air at 300, 350, 400, 450 and 500°C was employed to obtain polycrystalline powders of  $\text{Bi}_2\text{MoO}_6$ .

The second oxide is  $\text{V}_2\text{O}_5$  which was prepared by precipitation method using ethylenglycol as stabilizer. In a typical synthesis, 0.0054 mole of ammonium metavanadate ( $\text{NH}_4\text{VO}_3$ ) (Aldrich, 99%) were dissolved in 50 mL of distilled water or ethylenglycol ( $\text{HOCH}_2\text{CH}_2\text{OH}$ ) (Aldrich, 99%) under vigorous stirring. Subsequently, the solution was exposed to ultrasound irradiation (70W, 42 kHz) under ambient air in a water bath at 60°C for time intervals from 0 to 120 minutes. Once elapsed time, resulting mixture was heated at 100°C in a hot plate to promote the evaporation of the solvent. The resulted powders were heated at 400 and 500°C for 24 h to obtain polycrystalline powders.

### A. Model generation and comparison using genetic programming artificial neural network and linear regression

Other similar approaches can be used to generate mathematical models like genetic programming, artificial neural network and linear regression. For both oxides genetic programming uses the following operations  $\{+, -, *, /, \exp, \log\}$  for all nodes for 300 generations, considering 100 individuals. A simply crossover with a probability of 0.9 and a simple mutation with a probability of 0.05 is used. Statistical metrics as mean square error (MSE), PRESS, and  $R^2_{pred}$  are calculated. An 80% of experimental data is used for model building and 20% for test validation. Linear regression is executed under the same conditions. With artificial neural network approach, a multiperceptron neural network with back propagation rule was used, with 8 neurons on middle layer and a constant learning parameter 0.25 and a moment of 0.5 during 800 epochs.

In Symbolic regression operators and parameters are set by Evonorm, for the oxide  $\text{V}_2\text{O}_5$  a population of 200 individuals, 20 are selected for generating new population during 200 generations. For  $\text{Bi}_2\text{MoO}_6$  a population of 100 individuals, 10 are selected during 300 generations. A resume of the performance of the algorithm is shown on the table 2 and 3. Central processing unit (CPU) time is calculated using a laptop with dual-core 1.3Ghz with 4GB RAM.

**Table 2.** Statistical metrics results of best model found of  $\text{Bi}_2\text{MoO}_6$  using linear regression, genetic programming and symbolic regression.

Approach	MSE	PRESS	$R^2_{pred}$	CPU time (Seconds)
Linear regression	0.040092316	4.009231632	0.695858415	0.014
Genetic programming	0.048135072	4.8135072	0.486767808	152.025
Artificial neural network	0.032906447	3.290644684	0.866807996	11.414
Symbolic regression alpha-beta	0.037891119	3.789111887	0.702546929	133.385



**Table 3.** Statistical metrics results of best model found of  $V_2O_5$  using linear regression, genetic programming and symbolic regression.

Approach	MSE	PRESS	$R^2_{pred}$	CPU time (seconds)
Linear regression	0.146478636	14.6478636	0.364175752	0.009
Genetic programming	0.080161008	8.01610081	0.608180149	488.122
Artificial neural network	0.010416923	1.041692271	0.935792749	18.423
Symbolic regression alpha-beta	0.003638107	0.363810729	0.980505595	246.525

## VI. RESULTS

The models generated are normalized. The model generated with SR for the oxide  $V_2O_5$  is shown in equation 12:

$$f(x_1, x_2, x_3, x_4) = 0.01834868x_1^2 - 0.000848388x_2^2 + 0.015535327x_3 - 0.02876764x_4 + 0.1202515x_1 - 1.322645399 \quad (12)$$

where  $y$  is surface area,  $H_2O$  is  $x_1$ , ethylenglycol is  $x_2$ , ultrasound irradiation is  $x_3$  and heat treatment temperature is  $x_4$ . In this model the factor  $x_4$  is not considered, because there heat treatment temperature is irrelevant for the response of surface area.

The model generated with SR for the oxide  $Bi_2MoO_6$  is shown in equation 13:

$$f(x_1, x_2, x_3) = \frac{\exp(-0.1101033x_1 - 1)}{x_1^2 + 1.6175924x_1 + 0.654151293 + \exp(-0.1078465x_2 - 0.6371079) - (-0.1060289x_3 + 0.1046486)^{1/2}} \quad (13)$$

where  $y$  is half life time, ultrasound irradiation is  $x_1$ , heat treatment time is  $x_2$  and temperature of heat treatment is  $x_3$ . In this model the heat treatment time is irrelevant for the response of half life time.

Considering results shown in tables 2 and 3, criteria of low complexity, low error, high  $R^2_{pred}$  and low PRESS can be taken here. Performance on CPU time is best on linear regression, however its other statistical metrics are poor compared with symbolic regression. Genetic programming CPU time is higher compared with SR in both cases. Artificial neural network works well and better in model for  $Bi_2MoO_6$ , using ANN is a good option, but an explicit correlation between variables and effects on output response is not evident. The results on the model of oxide  $V_2O_5$  show a better performance compared with  $Bi_2MoO_6$  model. In this case to have more samples on dataset is recommended for better results on statistical metrics.

## VII. CONCLUSIONS AND FUTURE WORK

A symbolic regression modeling is proposed for different synthesis process of 2 oxides used on photocatalysis. A set of models are generated from experimental data. The oxides  $Bi_2MoO_6$  have 12 samples and  $V_2O_5$  have 18 samples. A comparison using similar approaches like artificial neural network, linear regression and genetic programming is made. The performance of each model is evaluated using statistical metrics and CPU time running the algorithm. The results show that symbolic regression model have good results compared with other techniques, especially when there is more samples on data set. A low performance model could be generated when there are few samples of the process. In this examples, few data are introduced, so models of low performance are generated; in spite of, results are superior than linear regression. Symbolic regression could be used in other chemical process to optimize their methods; nevertheless for future work other output values on experimental data set could be used to generate new models.

## REFERENCES

- [1] Luis M. Torres-Treviño, Indira G. Escamilla-Salazar, Bernardo González-Ortiz, Rolando Praga-Alejo.: An expert system for setting parameters in machining processes. Expert Systems with Applications 40, 6877–6884 (2013)
- [2] Guillermo González-Campos, Edith Luévano-Hipólito, Luis Martín Torres-Treviño, Azael Martínez-De La Cruz.: Artificial neural network for optimization of a synthesis process of  $\gamma$ - $Bi_2MoO_6$  using surface response methodology. MICAI 2012, Part II, LNAI 7630, pp. 200–210, (2013)
- [3] E. Luévano-Hipólito, A. Martínez-de la Cruz, E. López Cuéllar.: Synthesis, characterization, and photocatalytic properties of  $\gamma$ - $Bi_2MoO_6$  prepared by co-precipitation assisted with ultrasound irradiation. Journal of the Taiwan Institute of Chemical Engineers, (2014)
- [4] Guido F. Smits, Ekaterina Vladislavleva, Mark E. Kotanchek.: Scalable symbolic regression by continuous evolution with very small populations. Genetic Programming Theory and Practice VIII, Chapter 9 (2011)
- [5] Guido F. Smits, Mark Kotanchek.: Pareto-front exploitation in symbolic regression. Genetic programming theory and practice II, Chapter 17 (2005)
- [6] Guido Smits, Arthur Kordon, Katherine Vladislavleva, Elsa Jordaán, Mark Kotanchek.: Variable Selection in Industrial Datasets Using Pareto Genetic Programming. Genetic Programming Theory and Practice III pp 79-92, Volume 9, (2006)
- [7] Flor Castillo, Kenric Marshall, James Green, Arthur Kordon.: A Methodology for Combining Symbolic Regression and Design of Experiments to Improve Empirical Model Building. GECCO 2003, LNCS 2724, pp. 1975–1985, (2003)
- [8] Flor Castillo, Arthur Kordon, Guido Smits.: Robust Pareto Front Genetic Programming Parameter Selection Based on Design of Experiments and Industrial Data. Genetic Programming Theory and Practice IV, Genetic and Evolutionary Computation, pp 149-166 (2007)

- [9] Mark Kotanchek, Guido Smits, Arthur Kordon.: Industrial Strength Genetic Programming. Genetic Programming Theory and Practice, Genetic Programming Series Volume 6, pp 239-255 (2003)
- [10] Eduardo Oliveira Costa, Aurora Pozo.: A New Approach to Genetic Programming based on Evolution Strategies. Systems, Man and Cybernetics, 2006. SMC '06. IEEE International Conference on , vol.6, pp.4832-4837, 8-11 Oct. (2006)
- [11] Birkan Can, Cathal Heavy.: Comparison of experimental designs for simulation-based symbolic regression of manufacturing systems. Computers & Industrial Engineering 61, 447-462 (2011)
- [12] Xiong Shengwu, Wang Weiwu, Li Feng.: A New Genetic Programming Approach in Symbolic Regression. Tools with Artificial Intelligence, 2003. Proceedings. 15th IEEE International Conference on pp.161,165, 3-5 Nov. (2003)
- [13] Dervis Karaboga, Celal Ozturk, Nurhan Karaboga, Beyza Gorkemli.: Artificial bee colony programming for symbolic regression. Information Sciences 209, 1-15 (2012)
- [14] Vipul K. Dabhi, Sanjay K. Vij.: Empirical Modeling Using Symbolic Regression via Postfix Genetic Programming. Image Information Processing (ICIIP), 2011 International Conference on , vol., no., pp.1,6, 3-5 Nov. (2011)
- [15] Weihua Cai, Arturo Pacheco-Vega, Mihir Sen, K.T. Yang.: Heat transfer correlations by symbolic regression. International Journal of Heat and Mass Transfer 49, 4352-4359 (2006)
- [16] T.L. Lewa, A.B. Spencer, F. Scarpaa, K. Wordena, A. Rutherfordb, F. Hemez.: identification of response surface models using genetic programming. Mechanical Systems and Signal Processing 20, 1819-1831 (2006)
- [17] Ming-fang Zhu, Jian-bin Zhang , Yan-ling Ren, Yu Pan, Guang-ping Zhu.: Multivariable Symbolic Regression Based on Gene Expression Programming. iscid, vol. 2, pp.298-301, (2011)
- [18] J.W. Davidson, D.A. Savic , G.A. Walters.: Symbolic and numerical regression: experiments and applications. Information Sciences 150, 95-117 (2003)
- [19] John R. Koza.: Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, Cambridge, MA, USA, (1992)
- [20] P. Barmapalexis , K. Kachrimanis, A. Tsakonas, E. Georgarakis.: Symbolic regression via genetic programming in the optimization of a controlled release pharmaceutical formulation. Chemometrics and Intelligent Laboratory Systems 107, 75-82 (2011)
- [21] Wasif Afzal, Richard Torkar.: On the application of genetic programming for software engineering predictive modeling: A systematic review. Expert Systems with Applications 38, 11984-11997 (2011)
- [22] Chong, M.N., Jin, B., Chow, C.W.K., Saint, C.: Recent developments in photocatalytic water treatment technology: a review. Water Research 44, 2997 (2010)
- [23] Laufs, S., Burgeth, G., Duttlinger, W., Kurtenbach, R., Maban, M., Thomas, C., Wiesen, P., Kleffmann, J.: Conversion of nitrogen oxides on commercial photocatalytic dispersion paints. Atmospheric Environment (2010)
- [24] Yang, S., Gao, L.: New method to prepare nitrogen-doped titanium dioxide and its photocatalytic activities irradiated by visible light. Journal of the American Ceramic Society 87, 1803-1805 (2004)
- [25] E. Luévano-Hipólito, A. Martínez-de la Cruz, Q.L. Yu, H.J.H. Brouwers.: Precipitation synthesis of WO<sub>3</sub> for NO<sub>x</sub> removal using PEG as template. Ceramics International 40, 12123-12128 (2014)
- [26] Jain, R., Mathur, M., Sikarwar, S., Mittal, A.: Removal of the Hazardous Dye Rhodamine B through Photocatalytic and Adsorption Treatments. Journal of Environmental Management 85(4), 956-964 (2007)
- [27] Zou, Z.G., Ye, J.H., Sayama, K., Arakawa, H.: Direct Splitting of Water under VisibleLight Irradiation with an Oxide Semiconductor Photocatalyst. Nature 414, 625-627 (2001)
- [28] Tang, J.W., Zou, Z.G., Ye, J.H.: Effects of Substituting Sr<sup>2+</sup> and Ba<sup>2+</sup> for Ca<sup>2+</sup> on the Structural Properties and Photocatalytic Behaviors of CaIn<sub>2</sub>O<sub>4</sub>. Chemistry of Materials 16(9), 1644-1649 (2004)
- [29] Zou, Z., Ye, J., Sayama, K., Arakawa, H.: Photocatalytic and Photophysical Properties of a Novel Series of Solid Photocatalysts, Bi<sub>2</sub>MnNbO<sub>7</sub> (M=Al<sup>3+</sup>, Ga<sup>3+</sup> and In<sup>3+</sup>). Chemical Physics Letters 333(1-2), 57-62 (2001)
- [30] Kudo, A., Omori, K., Kato, H.: A Novel Aqueous Process for Preparation of Crystal Form Controlled and Highly Crystalline BiVO<sub>4</sub> Powder from Layered Vanadates at Room Temperature and Its Photocatalytic and Photophysical Properties. Journal American Chemistry Society 121(49), 11459-11467 (1999)
- [31] Kato, H., Hori, M., Kato, R., Shimodaira, Y., Kudo, A.: Construction of ZScheme-Type Heterogeneous Photocatalysis Systems for Water Splitting into H<sub>2</sub> and O<sub>2</sub> under Visible Light Irradiation. Chemistry Letters 33(10), 1348-1349 (2004)
- [32] S. S. R. Putluru, A. D. Jensen, A. Riisager, R. Fehrmann, Heteropoly.: Acid promoted V<sub>2</sub>O<sub>5</sub>/TiO<sub>2</sub> catalysts for NO abatement with ammonia in alkali containing flue gases. Catalysis Science and Technology 1, 631-637 (2011)
- [33] Torres-Treviño, L. M.: Identification and prediction using symbolic regression alpha-beta: preliminary results. Proceedings of the 2014 conference companion on genetic and evolutionary computation companion (GECCO), 1367-1372 (2014)
- [34] Torres-T, L.: Evonom, a new evolutionary algorithm to continuous optimization. In Workshop on optimization by building and using probabilistic models (OBUPM) genetic and evolutionary computation conference (GECCO) (2006)
- [35] Torres-Trevino, L.: Evonom: easy and effective implementation of estimation of distribution algorithms. Journal of Research in Computing Science 23, 75-83 (2006)
- [36] Douglas, C.: Montgomery introduction to linear regression analysis. John Wiley and Sons. (2007)