Chrome Layer Thickness Modelling in a Hard Chromium Plating Process Using a Hybrid PSO/ RBF–SVM–Based Model

Paulino José García Nieto[1]*, Esperanza García-Gonzalo[1], Fernando Sánchez Lasheras[1], Antonio Bernardo Sánchez[2]

[1] Department of Mathematics, Faculty of Sciences, University of Oviedo, 33007 Oviedo (Spain)
[2] Department of Mining Technology, Topography and Structures, University of León, 24071 León (Spain)

unir
LA UNIVERSIDAD
EN INTERNET

Abstract

The purpose of chromium plating is the creation of a hard and wear-resistant layer of chromium over a metallic surface. The principal feature of chromium plating is its endurance in the face of the wear and corrosion. This industrial process has a vast range of applications in many different areas. In the performance of this process, some difficulties can be found. Some of the most common are melt deposition, milky white chromium deposition, rough or sandy chromium deposition and lack of toughness of the layer or wear and lack of thickness of the layer deposited. This study builds a novel nonparametric method relied on the statistical machine learning that employs a hybrid support vector machines (SVMs) model for the hard chromium layer thickness forecast. The SVM hyperparameters optimization was made with the help of the Particle Swarm Optimizer (PSO). The outcomes indicate that PSO/SVM–based model together with radial basis function (RBF) kernel has permitted to foretell the thickness of the chromium layer created in this industrial process satisfactorily. Thus, two kinds of outcomes have been obtained: firstly, this model permits to determine the ranking of relevance of the seven independent input variables investigated in this industrial process. Finally, the high achievement and lack of complexity of the model indicate that the PSO/SVM method is very interesting compared to other conventional foretelling techniques, since a coefficient of determination of 0.9952 is acquired.

I. Introduction

HARD chromium is an electroplating process that has been employed for many years. Its purpose is the creation of an extremely hard surface over a metal part [1].

According to the thickness of the chromium layer, there are two kinds of chromium plating processes. The decorative plating which makes use of layer thickness of up to 1.5 μm and the hard chromium plating whose thickness goes from 2.5 to 500 μm. Not only the thickness is the difference of decorative and hard chromium plating processes, but also that in the case of hard chromium plating the layer is applied over the metal part while in the case of decorative plating a coat of nickel or copper nickel alloy is employed.

In general, the hard chromium plating process requires of some initial baths that clear the piece and of an electropolishing phase. Afterwards the chrome plating operation is performed. The bath in which this operation is performed contains chromic and sulphuric acid in proportions of 100 to 1 respectively [2]. A chemical equilibrium is maintained of chromic and sulphuric acid that have a reversible chemical reaction, forming chrome metal, water and oxygen. In this electrolytic process the anode employed has not to be replaced as it is solid.

The presence of sulphuric acid is compulsory, as it is essential for the reaction and therefore, the coating, to take place but it must be controlled in order to maintain among certain limits [3].

There is a vast range of composition proportions of the bath. For example, the weight ratio of chromic acid to sulphate goes from 75 to 1 to 120 to 1 with an optimum value about 90 to 1.

The disadvantages of low chromic acid have been reported in the literature [4] and they mainly consist of the difficulties to obtain the required thickness for the layer, the high sensitivity to different bath concentration and the need of a higher voltage. Despite these, such kind of baths are employed as they work faster. When high concentrations are employed, the resulting bath is less affected by bath contamination and, also, it can operate with lower voltages.

Hard chromium plating process is mainly applied in those parts that require of a high resistance to wear. The hardness of chromium plating is between 800 and 1000 HV. Also, it has a low friction coefficient. For example, it is of 0.16 for chromium to steel while for steel to steel it is of 0.30 [5]. One of the main fields where this kind of plating is useful is in automotive industry, where these covers are employed to fight against wear. Also, in iron and steel industry hard chromium plating is a process employed for plating rolling mill cylinders.

* Corresponding author.

E-mail address: pjgarcia@uniovi.es

In the case of tools, the hard chromium plating recommendation is linked to the achievement of a higher wear, friction and tear resistance [6]. This treatment is also of interest for the corrosion reduction. It means than any tool with a chrome layer from 2.5 to 13 μm will have a longer life. Hard chromium plating is also employed for the rework of parts out of tolerance. For example, engine components like shafts and cylinders.

Also, it is remarkable that those rolling mills that are coated experiment a great increase in their average life [3]. From a mechanical point of view, the properties of electrodeposited chromium are like those of metal chromium. The most important of these are its great hardness, low friction coefficient, good corrosion resistance, low thermal expansion, and high conductivity.

For a good result of a chromium plating process, it is not only required the control of the plating parameters but also having the required kind of base material. A cleaning with electropolishing before the plating is in many cases required. The base material employed must be able to withstand the external forces applied without hardly any deformation. That is that the properties of the base material must be like those of the applied chromium layer. Also, the thickness of the chromium layer should be reduced to a minimum in those areas of the part that would suffer from high deformation values.
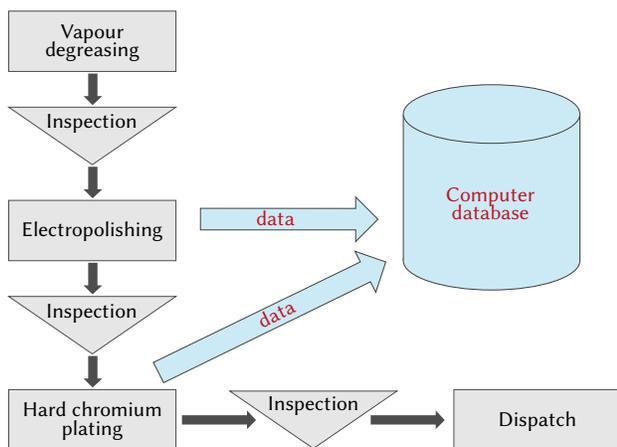

Fig. 1. The industrial process of hard chromium plating.

Fig. 1 shows the flowchart of the industrial hard chromium process. In a first stage, parts go through a vapour degreasing operation that cleans the part. It is followed by an electropolishing phase that prepares the part surface to receive the chromium layer. The next step is the proper chromium operation. This entire process requires of some intermediate inspection points and generates data that can be analyzed in order to adjust and improve the whole process.

The aim of this research is to assess the goodness of the support vector machines (SVMs) regression model [8]–[12] to predict the chromium layer thickness considering different variables from the industrial process. The purpose of the model developed in this research can overcome some problems found in industry and give to the researchers a more in-depth knowledge of the industrial process as a whole.

SVM are a kind of machine learning models that have shown a great performance for both regression and classification tasks [8]–[12]. In the case of the present research SVM models were trained to predict the thickness of the chromium plating layer. SVM models has proved to be successful in other disciplines such as the prediction of biological parameters in environmental problems like forest modeling [13], solar power generation prediction [14], [15] and air and water quality estimation [16]–[19].

Particle swarm optimization (PSO) technique is a metaheuristic evolutionary global method able to solve multidimensional optimization problems that makes use of continuous variables. It is a bio-inspired algorithm relied on swarm intelligence [20]–[25].

In summary, the main objective of the present study was to develop a hybrid algorithm with PSO optimizing SVM parameters (PSO/SVM–based model) to predict the thickness of the hard chromium layer from the seven physical-chemical input parameters from the hard chromium plating process.

II. Materials and Methods

A. The Chromium Plating Process and Experimental Datasets

The chromium plating process requires of different stages in order to achieve the required chromium layer. There are four main process stages: vapor degreasing, cleaning, electropolishing and hard chrome process.

Usually the degreasing in vapor phase is performed in a stainless steel tank. This process takes about 10 minutes as such time is enough in order to reach the temperature required. It can be verified that the required temperature has been reached because the steam does not condense on the lower surfaces of the pieces. The next step consists on a parts cleaning in order to remove all the residual traces of dirt and surface impurities, this operation has an important manual component.

The next stage is the Electropolishing: The electropolishing operation consists of the following phases, first of all, pieces are cleaned with acetone. Afterwards, laded in the frames where the electropolishing process is performed. For each electropolished load, the system records the minimum, maximum and average temperature values. The electropolishing time depends on the amount of material to be removed.

After the electropolishing the hard chromium plating stage of the process is performed. Before the chrome operation, pieces are cleaned with a pressurized water gun. Next, each piece is blown with pressurized air to remove the remains of water. The chromating of the pieces is carried out by immersion in a bath with a temperature between 53 and 56°C. Once the chromium plating process has finished, pieces are removed from the frame, washed with water and introduced into an alkaline bath. The minimum time in this bath will be of three minutes. After this time, they are extracted from the bath and washed by hand with water.

The experimental dataset was collected using a database from the hard chromium plating of the factory belonging to the company Tecnocrom Industrial Ltd (Madrid, Spain). Eight process variables are collected from the hard chrome plating process including the dependent variable (thickness of the hard chrome layer). These seven input variables are briefly described in Table I. The total number of data used in this study is 37,360. The output variable (or dependent variable) considered in this study was the hard chromium plating thickness (in microns).

In this research relationship between the thickness of the hard chromium layer (dependent variable) in microns and some physical-chemical variables (independent variables) has been studied. The variables employed are listed below:

- Iron content of the electropolishing bath (mg/L): this variable measures the iron content in the electropolishing bath expressed in mg/L.

- Electropolishing time (minutes): this variable represents the amount of time, expressed in seconds that the part is submerged in the electropolishing bath.

- Electropolishing bath temperature (ºC): this variable expresses the electropolishing bath temperature in Celsius degrees. It is well-know that when the higher the temperature value, the lower the process performance.
- Layer thickness removed by electropolishing (mm): the electropolishing process suposses the removal of a certain amount of material. This variable measures the amount of material removed.
- Chromic acid content (g/L): chromic acid content in the hard chromium bath expressed in g/L.
- Hard chrome plating time (minutes): this variable measures the time of the hard chromium plating operation.
- Hard chrome plating temperature (ºC): this variable expresses the chrome plating bath temperature in Celsius degrees. As in the case of the electropolishing and in this case, due to the electrolytical process, the bath temperature increases. The bath temperature influences in the operation performance.

TABLE I. Set of Operation Physical input Variables Used in This Study and their Names Along With their Mean and Standard Deviation

Input variables	Variable name	Mean	Stand. dev.
Iron content of the electropolishing bath (mg/l)	iron_ep	0.12	0.09
Electropolishing time (minutes)	ep_time	4.50	0.91
Electropolishing bath temperature (ºC)	ep_temp	53.60	4.62
Layer thickness removed by electropolishing (µm)	thick_electropol	305.9	34.46
Chromic acid content (g/l)	chrom_acid	240.6	8.71
Hard chromium process time (minutes)	hard_chrom_time	30.7	3.33
Hard chrome bath temperature (ºC)	hard_chrom_temp	53.9	0.47

Fig. 2 shows an overview of the eight sample variables over time. The highly nonlinear behaviour of the output variables can be clearly appreciated.

B. Computational Procedures

The first step in this study is the determination of the correlation matrix of all the process variables (see Fig. 3). This matrix indicates that there is hardly any correlation among the independent variables and some correlation of the dependent variable thickness of the hard chrome layer (thickness_chrom) with input variables layer thickness removed by electropolishing (thick_electropol) and hard chromium process time (hard_chrom_time). This outcome stands out the trouble that we can encounter in acquiring foretold models using linear regression methods. As a consequence, it is required to deal with this complicated issue employing more developed nonlinear procedures relied on statistical machine learning such as the support vector machines (SVMs) together with the particle swarm optimizer (PSO) giving place to the PSO/SVM–relied model used in this study.
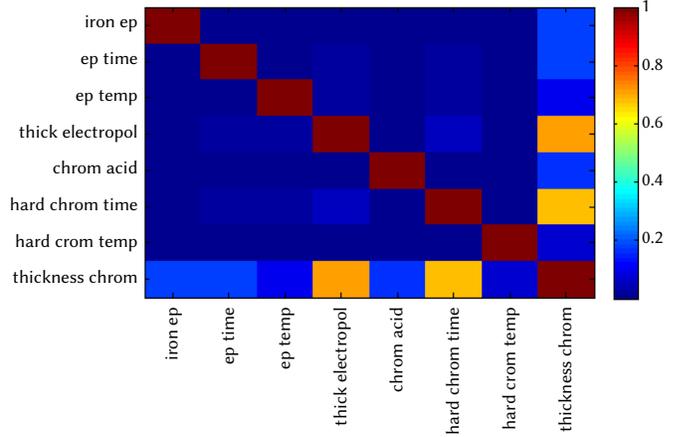

Fig. 3. Correlation matrix.

1. Support Vector Machines For Regression (SVR)

Initially, support vector machines (SVMs) arose to treat binary classification problems. Under these circumstances, it was quickly observed that the basic rules that support them could be used to tackle other kind of problems such as the regression problems. Certainly, in place of trying to sort new unnoticed variables $\mathbf{x}'$ into one of two classes $y' = \pm 1$, we are going now to foretell a real-valued output for y' from the training data $\{\mathbf{x}_i, y_i\}$, so that $i = 1, 2, ..., L$ with $y_i \in \Re$, $\mathbf{x}_i \in \Re^D$ [26]–[29] so that:

$$y_i = \mathbf{w} \cdot \mathbf{x}_i + b \tag{1}$$

where:

- $\mathbf{w}$ is director vector of the hyperplane (normal to this hyperplane);
- $\dfrac{b}{\|\mathbf{w}\|}$ is the perpendicular (normal) distance from the hyperplane to the coordinates origin.

The support vector regression (SVR) approach employs a more
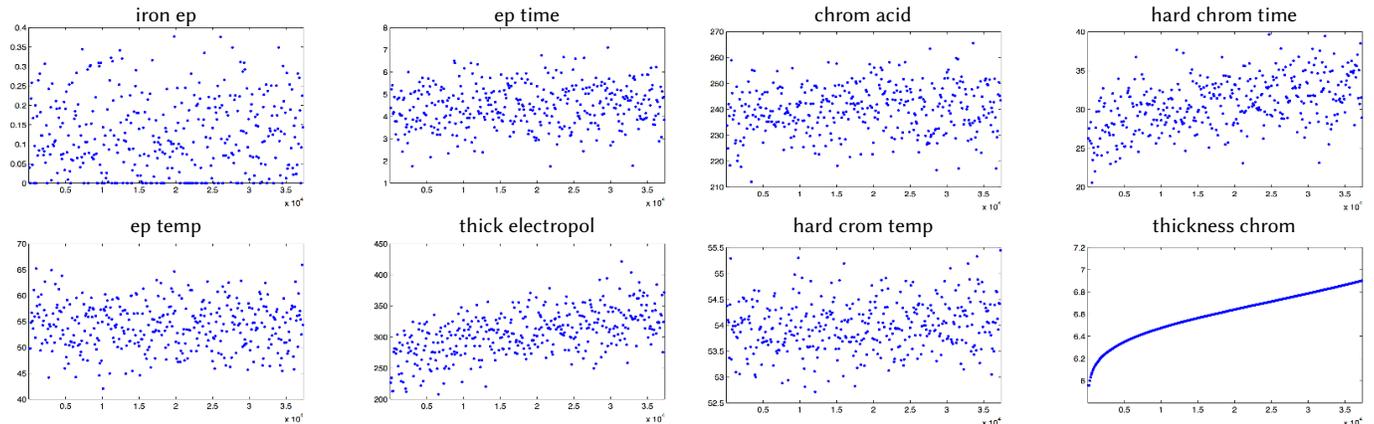

Fig. 2. Graphical overview of the process variables: variable value versus observation number.

cumbersome penalty function than in case of classification, not permitting a penalization if the foretold value y_i is less than a distance ε away from the observed value t_i, that is to say, if $|t_i - y_i|$ is fulfilled. Alluding to Fig. 4, the region enclosed by $y_i \pm \varepsilon$ for all i is termed an ε – insensitive tube. The another change of the penalization function is that output variables falling out of the tube receive one of two slack variable penalizations relying on whether they stay above (ξ^+) or below (ξ^-) the tube (so that $\xi_i^+ > 0$, $\xi_i^- > 0\ \forall i$) [12], [13], [30], [31]:

$$t_i \le y_i + \varepsilon + \xi_i^+ \tag{2}$$
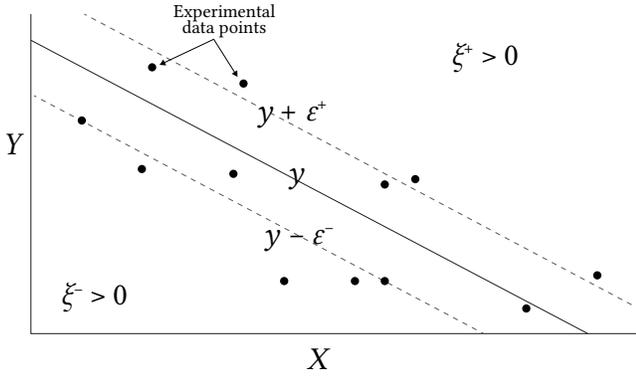
$$t_i \ge y_i - \varepsilon - \xi_i^- \tag{3}$$

Fig. 4. Regression with ε – insensitive tube.

The error function for SVR approach is given by [9], [30]:

$$C \sum_{i=1}^{L} \left(\xi_i^+ + \xi_i^- \right) + \frac{\|w\|^2}{2} \tag{4}$$

In brief, the *primal problem* in the event of regression is described as [12] , [31], [32]:

$$\min \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^{L} \left(\xi_i^+ + \xi_i^- \right)$$

$$\text{s.t.} \begin{cases} t_i - \left(\mathbf{w} \cdot \mathbf{x}_i + b \right) - \varepsilon - \xi_i^+ \le 0 \\ t_i - \left(\mathbf{w} \cdot \mathbf{x}_i + b \right) + \varepsilon + \xi_i^- \ge 0 \\ \xi_i^+ , \xi_i^- \ge 0, \forall i = 1, 2, ..., n \end{cases} \tag{5}$$

Next, we minimize the error function supplied by Eq. (4) along with the constraints $\xi_i^+ \ge 0$, $\xi_i^- \ge 0\ \forall i$ in addition to inequalities (2) and (3), which is expressed compactly by the Eq. (5). For the purpose of building the dual problem, the following steps are demanded [9], [12], [18], [30], [31]:

- Step 1: Lagrange function construction

To this end, we use the following Lagrange multipliers $\alpha_i^+ \ge 0, \alpha_i^- \ge 0, \mu_i^+ \ge 0, \mu_i^- \ge 0\ \forall i$:

$$L_P \left(\mathbf{w}, b, \xi^+, \xi^-, \alpha^+, \alpha^-, \mu^+, \mu^- \right) = \frac{1}{2} \|\mathbf{w}\|^2$$

$$+ C \sum_{i=1}^{L} \left(\xi_i^+ + \xi_i^- \right) - \sum_{i=1}^{L} \left(\mu_i^+ \xi_i^+ + \mu_i^- \xi_i^- \right)$$

$$- \sum_{i=1}^{L} \alpha_i^+ \left[\varepsilon + \xi_i^+ + \left(\mathbf{w} \cdot \mathbf{x}_i + b \right) - t_i \right] - \sum_{i=1}^{L} \alpha_i^- \left[\varepsilon + \xi_i^- - \left(\mathbf{w} \cdot \mathbf{x}_i + b \right) + t_i \right] \tag{6}$$

- Step 2: Implementation of Karush–Kuhn–Tucker (KKT) optimality conditions

Carrying out the differentiation of L_P with respect to $\mathbf{w}$, b, ξ_i^+ and ξ_i^-, and matching the derivatives to zero, it turns out that:

$$\frac{\partial L_P}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^{L} \left(\alpha_i^+ - \alpha_i^- \right) \mathbf{x}_i \tag{7}$$

$$\frac{\partial L_P}{\partial b} = 0 \Rightarrow \sum_{i=1}^{L} \left(\alpha_i^+ - \alpha_i^- \right) = 0 \tag{8}$$

$$\frac{\partial L_P}{\partial \xi_i^+} = 0 \Rightarrow C = \alpha_i^+ + \mu_i^+ \Leftrightarrow \mu_i^+ = C - \alpha_i^+ \tag{9}$$

$$\frac{\partial L_P}{\partial \xi_i^-} = 0 \Rightarrow C = \alpha_i^- + \mu_i^- \Leftrightarrow \mu_i^- = C - \alpha_i^- \tag{10}$$

$$\alpha_i^+ \left[\varepsilon + \xi_i^+ + \left(\mathbf{w} \cdot \mathbf{x}_i + b \right) - t_i \right] = 0 \tag{11}$$

$$\alpha_i^- \left[\varepsilon + \xi_i^- - \left(\mathbf{w} \cdot \mathbf{x}_i + b \right) + t_i \right] = 0 \tag{12}$$

- Step 3: Establishment of the relations among the primal problem's variables ($\mathbf{w}$, b, ξ^+, ξ^-) and those of the dual problem($\boldsymbol{\alpha}^+$, $\boldsymbol{\alpha}^-$, $\boldsymbol{\mu}^+$, $\boldsymbol{\mu}^-$). For this, we make use of Eq. (7).

- Step 4: From the result obtained in previous step, it is possible to remove the primary variables of the Lagrangian function so that:

$$L_D \left(\alpha^+, \alpha^- \right) = \sum_{i=1}^{L} \left(\alpha_i^+ - \alpha_i^- \right) t_i - \varepsilon \sum_{i=1}^{L} \left(\alpha_i^+ - \alpha_i^- \right)$$

$$- \frac{1}{2} \sum_{i=1}^{L} \sum_{j=1}^{L} \left(\alpha_i^+ - \alpha_i^- \right) \left(\alpha_j^+ - \alpha_j^- \right) \mathbf{x}_i \cdot \mathbf{x}_j \tag{13}$$

Using $\mu_i^+ \ge 0$ and $\mu_i^- \ge 0$ together with Eqs. (9) and (10) means that $\alpha_i^+ \le C$ and $\alpha_i^- \le C$. Finally, the mathematical formulation of the dual problem can be expressed as [9], [12], [18], [30], [31]:

$$\max_{\boldsymbol{\alpha}^+, \boldsymbol{\alpha}^-} \left[\sum_{i=1}^{L} \left(\alpha_i^+ - \alpha_i^- \right) t_i - \varepsilon \sum_{i=1}^{L} \left(\alpha_i^+ - \alpha_i^- \right) - \frac{1}{2} \sum_{i=1}^{L} \sum_{j=1}^{L} \left(\alpha_i^+ - \alpha_i^- \right) \left(\alpha_j^+ - \alpha_j^- \right) \mathbf{x}_i \cdot \mathbf{x}_j \right]$$

$$\text{s.t.} \begin{cases} \sum_{i=1}^{L} \left(\alpha_i^+ - \alpha_i^- \right) = 0 \\ 0 \le \alpha_i^+ \le C, i = 1, ..., n \\ 0 \le \alpha_i^- \le C, i = 1, ..., n \end{cases} \tag{14}$$

Replacing Eq. (7) into Eq. (1), new forecasts y' can be calculated employing [12], [30]:

$$y' = \sum_{i=1}^{L} \left(\alpha_i^+ - \alpha_i^- \right) \mathbf{x}_i \cdot \mathbf{x}' + b \tag{15}$$

An ensemble S of support vectors $\mathbf{x}_S$ can be determined encountering the indices i such that $0 < \alpha < C$ and $\xi_i^+ = 0$ (or $\xi_i^- = 0$). This operation gives place to [12], [30]:

$$b = t_S - \varepsilon - \sum_{m=1}^{L} \left(\alpha_m^+ - \alpha_m^- \right) \mathbf{x}_m \cdot \mathbf{x}_S \tag{16}$$

Following the methodology, we average over all indices i in S so that b is defined correctly by [9], [12], [17]:

$$b = \frac{1}{N_S} \sum_{s \in S} \left[t_S - \varepsilon - \sum_{m=1}^{L} \left(\alpha_m^+ - \alpha_m^- \right) \mathbf{x}_m \cdot \mathbf{x}_S \right] \tag{17}$$

Additionally, some investigations have shown that some regression problems are not linearly solved in the space of the inputs $\mathbf{x}$, but are solvable in a higher-dimensional space called the *feature space* by using a suitable mapping $\mathbf{x} \rightarrow \Phi (\mathbf{x})$. This indicates that if the functions can be brought into a space of higher dimensionality by some potentially nonlinear mapping $\mathbf{x} \rightarrow \Phi (\mathbf{x})$, it is only necessary

to determine the scalar products of the inputs mapped in the feature space without having to explicitly calculate $\Phi(\mathbf{x})$. For the purpose of using an SVM approach to solve a regression problem with nonlinear separable data, firstly it is necessary to select a kernel together with its relevant parameters so that we can map the nonlinear separable data into a *feature space* where they are linearly separable (*kernel trick*).

Hence, the first step is the selection of a suitable kernel that gives place to a mapping $\mathbf{x} \rightarrow \Phi(\mathbf{x})$. Next, it is mandatory to follow the steps indicated below [9], [12], [30]:

- Selection of the appropriate values for the hyperparameters C and ε, that is to say, the weight of misclassifications and the size of the insensitive loss region.

- Determination *of* $\boldsymbol{\alpha}^+$ and $\boldsymbol{\alpha}^-$ by means of:

$$\max_{\boldsymbol{\alpha}^+,\boldsymbol{\alpha}^-} \left[\sum_{i=1}^{L}\left(\alpha_i^+ - \alpha_i^-\right)t_i - \varepsilon \sum_{i=1}^{L}\left(\alpha_i^+ - \alpha_i^-\right) \right.$$
$$\left. -\frac{1}{2}\sum_{i=1}^{L}\sum_{j=1}^{L}\left(\alpha_i^+ - \alpha_i^-\right)\left(\alpha_j^+ - \alpha_j^-\right)k\left(\mathbf{x}_i,\mathbf{x}_j\right) \right]$$
$$\text{s.t.} \begin{cases} \sum_{i=1}^{L}\left(\alpha_i^+ - \alpha_i^-\right) = 0 \\ 0 \le \alpha_i^+ \le C, i=1,...,n \\ 0 \le \alpha_i^- \le C, i=1,...,n \end{cases}$$
(18)

such that $k(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i)\cdot\Phi(\mathbf{x}_j)$ is termed the kernel function. Moreover, the kernel function must be symmetric and semidefinite positive according to the Moore–Aronszajn theorem. Likewise, the solution of the dual problem (15) is carried out employing a quadratic programming (QP) solver.

- Calculation of $\mathbf{w} = \sum_{i=1}^{L}\left(\alpha_i^+ - \alpha_i^-\right)\Phi(\mathbf{x}_i)$.
- Determination of the ensemble of *Support Vectors S* encountering the indices i such that $0 < \alpha \le C$ and $\xi_i^+ = 0$ (or $\xi_i^- = 0$).
- Calculation of b:

$$b = \frac{1}{N_S}\sum_{s\in S}\left[t_s - \varepsilon - \sum_{m=1}^{L}\left(\alpha_m^+ - \alpha_m^-\right)k\left(\mathbf{x}_m,\mathbf{x}_s\right) \right]$$
(19)

- All novel point $\mathbf{x}'$ is encountering by means of the expression:

$$y' = \sum_{i=1}^{L}\left(\alpha_i^+ - \alpha_i^-\right)\Phi(\mathbf{x}_i)\cdot\Phi(\mathbf{x}') + b$$
$$= \sum_{i=1}^{L}\left(\alpha_i^+ - \alpha_i^-\right)k\left(\mathbf{x}_i,\mathbf{x}'\right) + b$$
(20)

Several usual functions used as kernels in the scientific literature [10], [12], [30]–[32] are formulated as:

- Radial basis function termed RBF kernel:

$$k\left(\mathbf{x}_i,\mathbf{x}_j\right) = e^{-\sigma\|\mathbf{x}_i-\mathbf{x}_j\|^2}$$
(21)

- Polynomial kernel:

$$k\left(\mathbf{x}_i,\mathbf{x}_j\right) = \left(\sigma\mathbf{x}_i\cdot\mathbf{x}_j + a\right)^b$$
(22)

- Sigmoid kernel:

$$k\left(\mathbf{x}_i,\mathbf{x}_j\right) = \tanh\left(\sigma\mathbf{x}_i\cdot\mathbf{x}_j + a\right)$$
(23)

so that a, b and σ are hyperparameters indicating the kernel's functioning.

Moreover, representative parameters of the SVM approach can be summarized as [12], [30], [32]:

- Regularization constant (C): also term *cost function*. This factor defines the trade-off between the margin (i.e., model flatness) and the relevance of the slack variables (related to the training error). Furthermore, this constant C must be chosen a priori, being a parameter of the machine learning.

- ε parameter: this factor controls the width of the error margin allowed. The second term of the objective function (see Eqs. (4) and (5)) is known as empirical error determined by means of the $\varepsilon-$insensitive loss function, which indicates that it does not disregard errors below ε (i.e. to a distance ε of the true value).

- a, b and σ: these factors determine the expression of the different kernels in the subsequent model.

Therefore, it is appropriate to utilize some mathematical technique that determines the previous hyperparameters accurately enough. These parameters determine the ensemble of support vectors and their influence in the subsequent regression model, that is to say, the precision and robustness of the last model. Moreover, the particle swarm optimizer (PSO) explained in more detail below was employed [20], [33] with triumph in this work.

2. The Particle Swarm Optimization (PSO) Algorithm

Particle swarm optimizer (PSO) [20]–[22], [33] is a heuristic kind of optimizer aimed at finding global minimums or maximums. Its operation is motivated by the habits of groups of birds or ensembles of fish in which, the movement of each individual (direction, speed, acceleration, etc.), is the result of combining the individual decisions of each with the behaviour of the rest.

Although there are variations, in general terms, the structure of a PSO algorithm to optimize (maximize or minimize) a function with one or multiple variables follows the following steps:

1. Create an initial swarm of n random particles. Each particle consists of 4 elements: a position that represents a certain combination of variable values, the value of the goal function at the position where the particle is located, a velocity that indicates how and where the particle moves, and a record of the best position in which the particle has been so far.

2. Evaluate each particle with the objective function.

3. Update the position and speed of each particle. This is the part that gives the algorithm the ability to optimize. This process is described in detail in the step *Move the particle* below.

4. If a stopping criterion is not fulfilled, return to step number 2.

In the following steps, each stage of the process is implemented to finally combine them all into a single function:

- *Create a particle*

Each particle is determined by a position, speed and value that vary as the particle moves. In addition, it also stores the best position in which the particle has been so far. When a new particle is created, only information about its position and velocity (normally initiated as zero) is available and the rest of the values are not known until the particle is evaluated.

- *Evaluate the particle*

Evaluating a particle consists in calculating the value of the goal function in the position that the particle occupies in that moment. The position with the best value in which the particle has been so far is stocked up on. In order to identify if a novel location is better than the previous ones, it is necessary to know if it is a problem of minimization or maximization.

- *Move the particle*

Moving a particle means updating its speed and position. This step is the most important since it gives the algorithm the ability to optimize.

The speed of each particle in the ensemble is upgraded employing the next equation:

$$v_i\left(t+1\right) = \omega v_i\left(t\right) + c_1 r_1 \left[\hat{x}_i\left(t\right) - x_i\left(t\right)\right]$$
$$+ c_2 r_2 \left[g\left(t\right) - x_i\left(t\right)\right] \tag{24}$$

where:

- $v_i(t+1)$: particle velocity i at time $t+1$, that is, the new velocity.
- $v_i(t)$: particle velocity i at time t, that is, the current velocity.
- ω: inertia coefficient, which reduces or increases at the speed of the particle.
- c_1: cognitive coefficient.
- r_1: vector of random values between 0 and 1 of length equal to that of the velocity vector.
- $\hat{x}_i(t)$: best position in which the particle i has been so far.
- $x_i(t)$: position of the particle i at time t.
- c_2: social coefficient.
- r_2: vector of random values between 0 and 1 of length equal to that of the velocity vector.
- $g(t)$: position of the whole swarm at time t, that is, the best overall value.

To understand how this equation relates to the motion of the particle, it is useful to differentiate three parts:

- $\omega v_i(t)$ is the inertia constituent, in charge of remaining the particle advancing in the direction in which it has been doing so far. The recommended value of the inertia constituent ω is usually ranging from 0.8 to 1.2. If $\omega < 1$, the particle slows down as the iterations progress. This means less exploration but a faster convergence towards the optimum. If $\omega > 1$, the particle is accelerating, allowing more areas of the function space to be explored, but this makes convergence difficult.
- $c_1 r_1 [\hat{x}_i(t) - x_i(t)]$ is the cognitive constituent, responsible for the particle tendency to advance towards the position where it has obtained better results so far. The cognitive coefficient c_1 is usually bounded in the range [0, 2], where 2 is the recommended value. r_1 is a vector of random values between 0 and 1 (one value for each dimension) that provides some stochastic behavior to the movement of the particles, thus improving the ability to escape from local minima.
- $c_2 r_2 [g(t) - x_i(t)]$ is the social constituent, responsible for the particle tending to advance towards the best position found by the flock up until now. It can be interpreted as *collective knowledge*. The value of the social coefficient c_2 is usually limited in the range [0, 2], where 2 is the recommended value. r_2 is a vector of random values between 0 and 1 (one value for each dimension) that provides some stochastic behavior to the movement of the particles, thus improving the ability to escape from local minima.

The relative magnitude between the cognitive component and the social component allows regulating the exploratory behaviour of the algorithm. The higher the value of c_1 with respect to c_2, the greater independence of movement each particle has, which allows greater exploration but slower convergence. On the contrary, the higher the value of c_2 with respect to c_1, the more obliged the particles are to advance towards the most excellent zone found up until now, which reduces exploration but accelerates convergence.

Once the new velocity is calculated, the position of the particle can be updated with the expression:

$$x_i\left(t+1\right) = x_i\left(t\right) + v_i\left(t+1\right) \tag{25}$$

Finally, the parameters ω, c_1 and c_2 must be selected and control the behaviour and efficacy of the PSO method. Moreover, the Standard PSO 2011 [34] has been employed in this work. It considers some betterments from the point of view of its implementation [20]–[22], [33], [34] and besides the PSO parameters are calculated according to the next expressions:

$$\omega = \frac{1}{2\ln 2} \text{ and } \varphi_p = \varphi_g = 0.5 + \ln 2 \tag{26}$$

3. Accuracy of this Approach

Seven input variables already described previously in subsection 2.1 were employed in this study to build this novel PSO/SVM–relied method applying a regularized method [8]. As it is also known, the hard chrome layer thickness (HCLT) is the dependent variable that we want to foretell. In order to forecast HCLT from the seven remaining input variables with sufficient security, it is mandatory to select the best model fitted to the observed dataset. Although several possible statistics can be used to ascertain the goodness–of–fit, the rule used in this study was the coefficient of determination R^2 [8], [35]. The coefficient of determination is a statistic used in the context of a statistical model to foretell future results or to test a hypothesis. Next, we will call the observed values t_i versus the values predicted by the model y_i. Then, it is possible to define the following sums of squares given by [35]:

- $SS_{tot} = \sum\limits_{i=1}^{n}\left(t_i - \bar{t}\right)^2$: is the overall sum of squares, proportional to the sample variance.
- $SS_{reg} = \sum\limits_{i=1}^{n}\left(y_i - \bar{t}\right)^2$: is the regression sum of squares, also termed the explained sum of squares.
- $SS_{err} = \sum\limits_{i=1}^{n}\left(t_i - y_i\right)^2$: is the residual sum of squares.

where $\bar{t}$ is the mean of the n observed data:

$$\bar{t} = \frac{1}{n}\sum\limits_{i=1}^{n} t_i \tag{27}$$

Considering the former sums, the coefficient of determination is given by the following equation:

$$R^2 \equiv 1 - \frac{SS_{err}}{SS_{tot}} \tag{28}$$

The closer the R^2 statistic is to the value 1.0, the smaller the difference between the observed and predicted data.

Two additional criteria considered in this study were the root mean square error (RMSE) and mean absolute error (MAE) [35]. These statistics are also used frequently to evaluate the forecasting capability of a mathematical model. Indeed, the root mean square error (RMSE) and mean absolute error (MAE) are given by the expressions [8], [35]:

$$RMSE = \sqrt{\frac{\sum\limits_{i=1}^{n}\left(t_i - y_i\right)^2}{n}} \tag{29}$$

$$MAE = \frac{\sum\limits_{i=1}^{n}\left|t_i - y_i\right|}{n} \tag{30}$$

If the root mean square error (RMSE) has a value of zero, it means that there is no difference between the predicted and observed data. Mean Absolute Error (MAE) is the average vertical distance between each point and the identity line. MAE is also the average horizontal distance between each point and the identity line. MAE has a clear interpretation as the average absolute difference between t_i and y_i.

III. Results and Discussion

The hybrid model built here used as input variables seven distinct input operator parameters. All these variables were previously given in Table I. The hard chrome layer thickness (HCLT) is taken as output variable of this suggested new hybrid PSO/SVM–relied model. The forecast carried out from the seven self-reliant variables [8], [26] was appropriate as it was already exposed before, the SVM approximation is affected by the assortment of the SVM hyperparameters such as the regularization constant C, the width ε of the insensitive tube, and the parameters a, b and σ which condition the shape of the distinct kernels in the ultimate model. Usually, the traditional way of performing hyperparameter optimization in most computational codes has been *grid search*, or a *parameter sweep*, which is simply an exhaustive searching through a manually specified subset of the hyperparameter space of a learning algorithm. Indeed, the grid search is a *brute force* method and, as such, almost any optimization method improves its efficiency [10], [12], [30], [32].

The determination of the SVM optimal parameters was carried out assisted by the metaheuristic optimizer termed as *particle swarm optimizer* (PSO) [20]–[22], [34] which showed a good performance. In this sense, we have built a novel hybrid model called as PSO/SVM–relied model, which utilizes as output (dependent) variable HCLT in the hard chromium plating process [4].

For the purpose of studying variables effect, the coefficient of determination was calculated. The flowchart indicating the way of implementation of the present novel model is made visible in Fig. 5.
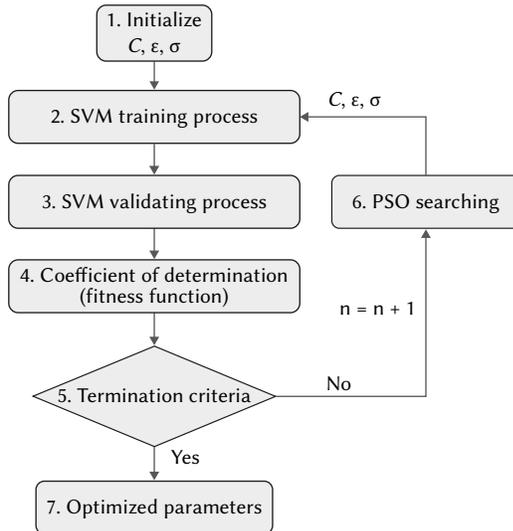

Fig. 5. Flowchart of the novel hybrid PSO/SVM–relied model.

Next, Fig. 6 makes visible those terms that make up of the best-fitted PSO/SVM–relied model for prediction of the hard chrome layer thickness (HCLT).

This graphical chart permits us to know the relevance of the distinct input variables entailed in the model. From the previous chart, Fig. 6 (a) shows the hard chrome layer thickness (Y–axis) as a function of hard chromium process time (X–axis), keeping as constant the six

remaining input variables. Similarly, Fig. 6 (d) shows the hard chrome layer thickness as a function of hard chromium process time and the thickness of the layer removed by electropolishing, keeping constant the five remaining input variables.
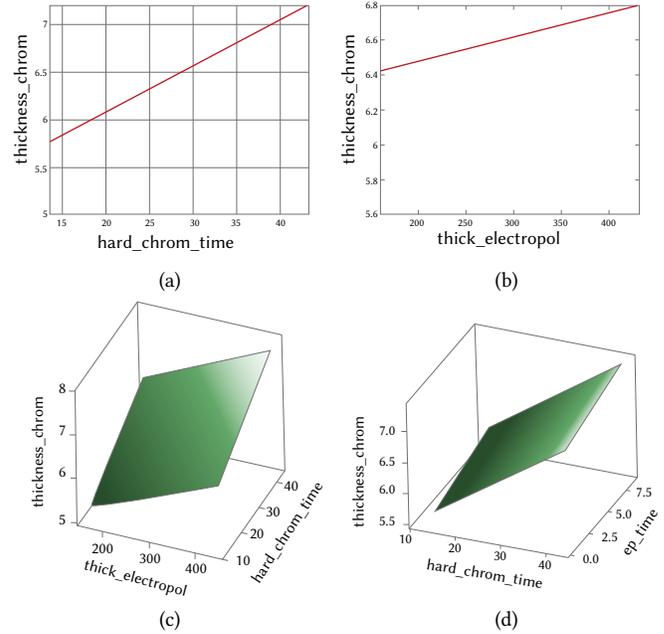

Fig. 6. Graphical chart of the terms that make up the hybrid PSO/SVM model for the hard chrome layer thickness (HCLT): (a) hard chromium process time first-order term; (b) layer thickness removed by electropolishing first-order term; (c) electropolishing time first-order term; (d) hard chromium process time and layer thickness removed by electropolishing second-order term; (e) hard chromium process time and electropolishing time second-order term; and (f) layer thickness removed by electropolishing and electropolishing time second-order term.

The value of R^2 was calculated assisted by the K–fold cross-validating procedure [36]. The mean of the term R^2 values obtained is taken into account to be the cross-validating R^2. The LIBSVM library of the statistical software R [37], along with the PSO technique utilizing the standard PSO 2011 form [34], were employed to build the ultimate regression model. Table II indicates the ranges of the space of solutions utilized for the metaheuristic PSO optimizer.

TABLE II. Intervals of the Solutions Space for Each of the RBF–SVM Parameters in the PSO Tuning Procedure

SVR parameters	Lower limit	Upper limit
C	10^{-4}	10^1
ε	10^{-6}	10^0
σ	10^{-4}	10^0

Hence, the PSO algorithm was employed to find the optimal SVM parameters utilizing the cross-validation error in each interaction. The particles $\mathbf{x}_i$ are represented by vectors that include the parameters to tune: for example, $\mathbf{x}_i = (C_i, \varepsilon_i, \sigma_i)$ for the RBF kernel employed here. Moreover, we have employed 20 particles in this work. Next, we initialized them randomly in the first iteration. Then, the particles for the next iterations were calculated according to the PSO algorithm. The value of the objective function for the particles was calculated in each stage. In this sense, the objective function value was computed as the cross-validation coefficient of determination for each particle. When the stopping criteria were fulfilled, the global best $\mathbf{x}_i$ included the optimized parameters. Additionally, the search space is three-dimensional and Table III points out the optimal parameters of the

best-fitted PSO/SVM–relied model with RBF kernel encountered with the PSO optimizer.

TABLE III. Optimal Parameters of the Best-fitted RBF–SVM Model Encountered With the PSO Optimizer

RBF/SVR hyperparameters	Optimal values
C	0.99900×10^1
ε	2.972757×10^{-3}
σ	6.396450×10^{-1}

Moreover, Table IV shows the determination and correlation coefficients for the novel PSO/SVM–relied approach with RBF kernel adjusted for the hard chrome layer thickness.

TABLE IV. Coefficient of Determination (R^2), Correlation Coefficient (R), Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) for the Novel PSO/SVM–relied Model With RBF Kernel Adjusted for the Hard Chrome Layer Thickness

Model	R^2/ R	RMSE	MAE
PSO/RBF–SVM	0.9952/0.9975	0.01571	0.006050

Taking into account the outcomes accomplished, the SVM technique together with the PSO optimizer is able to build a novel model with a high efficiency for the forecasting of the hard chrome layer thickness. The R^2 of the fitted SVM model turned out to be 0.9952 and a correlation coefficient of 0.9975. A computer with a CPU Intel Core i7-4770 @ 3.40 GHz with eight cores and 15.5 GB RAM memory was used, taking 333 seconds (approximately 6 min) to obtain the hard chrome layer thickness (HCLT) model.

Table V indicates the weights of the seven input variables entailed in the best-fitted PSO/SVM–relied model with RBF kernel for the hard chrome layer thickness (HCLT) forecast. These weights in absolute value indicate the relevance of the distinct independent input variables within the model. Indeed, the greater the weight in absolute value, the more significant is the variable. According to these rules, hard chromium process time is the most significant input variable in the HCLT forecast. The next input variable in relevance is the layer thickness removed by electropolishing. Similarly, the electropolishing time and chromic acid content are the third and fourth most significant variables in the HCLT forecast, respectively. Then, the iron content of the electropolishing bath holds the fifth place in relevance and, finally the two last places are occupied by electropolishing bath temperature and hard chrome bath temperature (see Table V and Fig. 7).

TABLE V. Relevance Ranking for the input Variables Entailed in the Best-fitted PSO/SVM–relied Model With RBF Kernel for the Hard Chrome Layer Thickness (HCLT) forecast

Input variable	Weight
hard_chrom_time	0.9683
thick_electropol	0.9129
ep_time	0.2493
chrom_acid	0.2327
iron_ep	0.2025
hard_chrom_temp	0.1246
ep_temp	0.1220

Furthermore, the model demonstrated its ability to forecast the HCLT. Fig. 8 makes a comparison between the observed and predicted values of the HCLT by employing this novel hybrid PSO/SVM–relied model. Therefore, a convenient way to overcome the difficulties due to the problem complexity as a result of nonlinearities is the use of this PSO/SVM–relied model.
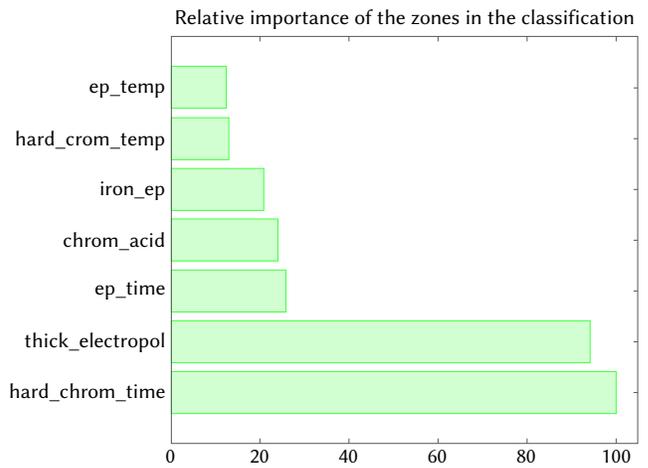

Fig. 7. Relative relevance of the input operation variables to forecast the hard chrome layer thickness (HCLT).
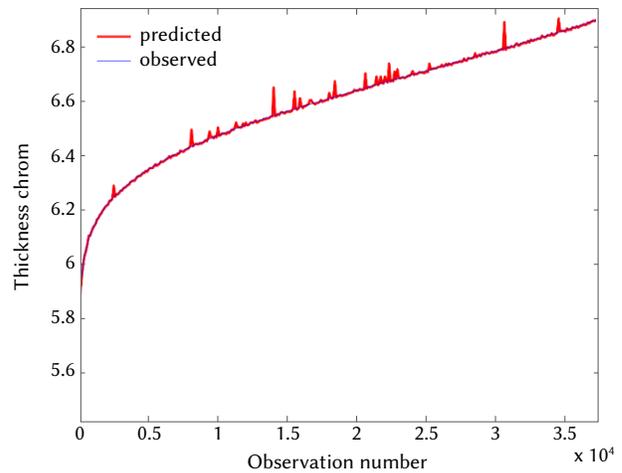

Fig. 8. Comparison between the HCTL values observed and predicted by employing the PSO/SVM–relied model (R^2 = 0.9952).

In summary, this novel hybrid model exposed here permits to assess the HCTL conforming to the actual experimental values observed employing the PSO/SVM–relied model with enough precision and success. Therefore, it is very appropriate the use of a RBF–SVM model with a PSO–relied optimizer for the purpose of accomplishing the best effective approximation in this regression problem. Obviously, these results have concordance again with the statistical criterion based on 'the goodness-of-fit' (R^2), since the PSO/SVM–relied model with RBF kernel achieves an outstanding fitting.

IV. Conclusions

Relied on the former results, several core discoveries of this study can be drawn and indicated as follows:

- A PSO/SVM–relied model is an accurate tool in order to predict the hard chrome layer thickness (HCLT).

- In this study, we have predicted the HCLT from the measured seven independent variables. This type of data-driven models are serviceable for industry for the purpose of decreasing costs in the setup of new industrial processes.

- The predicted HCLT values match with the observed ones since this novel PSO/SVM model gives place to a high coefficient of determination equal to 0.9952.

- The assessment of the input variables relevance in this industrial process was possible. Indeed, the hard chromium process time and the variable layer thickness removed by electropolishing are the most significant variables.

- The PSO/SVM–relied model used the LIBSVM library of the free software environment for statistical computing R [37] in combination with the PSO optimizer employing the standard PSO 2011 form [34].

- According to the results obtained it has been tested that PSO/SVM–relied model with RBF kernel improves the generalization capability reached employing only the standard support vector machines (SVMs) for regression.

To sum up the current research, it can be affirmed that this new approximation can be put into practice in other similar plating industrial processes. From our point of view and to conclude, this effective PSO/SVM–relied approach with RBF kernel is a convenient and pragmatic solution to the problem of the forecasting of the hard chrome layer thickness obtained from the hard-chromium plating industrial process.

Acknowledgment

References

[1] C. G. Fink, U.S. Patent 1,581,188, April 20, 1926.

[2] J. K. Dennis, T. E. Such, *The nickel and chromium plating*, New York, USA: Woodhead Publishing, 1994.

[3] F. Loffler, "Methods to investigate mechanical properties of coatings," *Thin Solid Films*, vol. 339, no. (1–2), pp. 181–186, 1999.

[4] R. K. Guffie, *The handbook of chromium plating*, New York, USA: Gardner Publications Ltd, 1986.

[5] J. H. Lindsay, "Decorative and hard chromium plating," *Plating and Surface Finishing*, vol. 90, pp. 22–24, 2003.

[6] M. Schlesinger, M. Paunovich, *Modern electroplating*, New York, USA: Wiley-Interscience, 2000.

[7] B. Irving, R. Knight, R. W. Smith, "The HVOF process - The hottest topic in the thermal spray industry," *Welding Journal*, vol. 72, no. 7, pp. 25–30, 1993.

[8] M. Kuhn, K. Johnson, *Applied predictive modeling*, New York, USA: Springer, 2013.

[9] B. Schölkopf, A. J. Smola, R. Williamson, P. Bartlett, "New support vector algorithms," *Neural Computation.*, vol. 12, no. 5, pp. 1207–1245, 2000.

[10] T. Hansen, C. J. Wang, "Support vector based battery state of charge estimator," *Journal of Power Sources*, vol. 141, pp. 351–358, 2005.

[11] X. Li, D. Lord, Y. Zhang, Y. Xie, "Predicting motor vehicle crashes using Support Vector Machine models," *Accident Analysis and Prevention*, vol. 40, pp. 1611–1618, 2008.

[12] I. Steinwart, A. Christmann, *Support vector machines*, New York, USA: Springer, 2008.

[13] C. Ordóñez, F. Sánchez Lasheras, J. Roca-Pardiñas, F. J. de Cos Juez, F. J., "A hybrid ARIMA–SVM model for the study of the remaining useful life of aircraft engines". *Journal of Computational and Applied Mathematics*, vol. 346, pp. 184–191, 2019.

[14] V. H. Quej, J. Almorox, J. A. Arnaldo, L. Saito, "ANFIS, SVM and ANN soft–computing techniques to estimate daily global solar radiation in a warm sub-humid environment," *Journal of Atmospherical and Solar-Terrestrial Physics*, vol. 155, pp. 62–70, 2017.

[15] F. J. de Cos Juez, F. Sánchez Lasheras, N. Roqueñí, J. Osborn, "An ANN-Based Smart Tomographic Reconstructor in a Dynamic Environment", *Sensors*, vol. 12, no. 7, pp. 8895–8911.

[16] A. Suárez Sánchez, P. J. García Nieto, P. Riesgo Fernández, J. J. del Coz Díaz, F. J. Iglesias–Rodríguez, "Application of an SVM-based regression model to the air quality study at local scale in the Avilés urban area (Spain)," *Mathematical and Computer Modelling*, vol. 54, no. (5–6), pp. 1453–1466, 2011.

[17] P. J. García Nieto, E. F. Combarro, J. J. del Coz Díaz, E. Montañés, "A SVM-based regression model to study the air quality at local scale in Oviedo urban area (Northern Spain): A case study," *Applied Mathematics and Computation*, vol. 219, no. 17, pp. 8923–8937, 2013.

[18] P. J. García Nieto, J. R. Alonso Fernández, F. Sánchez Lasheras, F. J. de Cos Juez, C. Díaz Muñiz, "A new improved study of cyanotoxins presence from experimental cyanobacteria concentrations in the Trasona reservoir (Northern Spain) using the MARS technique", *Sciences of the Total Environment*, 430, pp. 88–92, 2012.

[19] X. Xu, Y. Liu, S. Liu, J. Li, G. Guo, K. Smith, "Real-time detection of potable-reclaimed water pipe cross-connection events by conventional water quality sensors using machine learning methods," *Journal of Environmental Management*, vol. 238, pp. 201–209, 2019.

[20] R. C. Eberhart, Y. Shi, J. Kennedy, *Swarm intelligence*, San Francisco, USA: Morgan Kaufmann, 2001.

[21] M. Clerc, *Particle swarm optimization*, London, UK: Wiley-ISTE, 2006.

[22] K. E. Parsopoulos, M. N. Vrahatis, *Particle swarm optimization and intelligence: advances and applications*, Hershey, USA: IGI Global, 2010.

[23] B. K. Panigrahi, Y. Shi, M.–H. Lim, *Handbook of swarm intelligence: concepts, principles and applications*, Berlin: Springer, 2011.

[24] D. Simon, *Evolutionary optimization algorithms*, New York, USA: Wiley, 2013.

[25] X.–S. Yang, Z. Cui, R. Xiao, A. H. Gandomi, M. Karamanoglu, *Swarm intelligence and bio-inspired computation: theory and applications*, London, UK: Elsevier, 2013.

[26] V. Vapnik, *Statistical learning theory*, New York, USA: Wiley-Interscience, 1998.

[27] M. Pal, A. Goel, "Estimation of discharge and end depth in trapezoidal channel by support vector machines," *Water Resources Management*, vol. 21, no. 10, pp. 1763–1780, 2007.

[28] J.–L. Chen, G.–S. Li, S.–J. Wu, "Assessing the potential of support vector machine for estimating daily solar radiation using sunshine duration," *Energy Conversion and Management*, vol. 75, pp. 311–318, 2013.

[29] M. Abbaszadeh, A. Hezarkhani, S. Soltani–Mohammadi, "Proposing drilling locations based on the 3D modeling results of fluid inclusion data using the support vector regression method," *Journal of Geochemical Exploration*, vol. 165, pp. 23–34, 2016.

[30] N. Cristianini, J. Shawe–Taylor, *An introduction to support vector machines and other kernel-based learning methods*, Cambridge, UK: Cambridge University Press, 2000.

[31] T. Gu, W. Lu, X. Bao, N. Chen, "Using support vector regression for the prediction of the band gap and melting point of binary and ternary compound semiconductors," *Solid State Sciences*, vol. 8, pp. 129–136, 2006.

[32] J. Shawe–Taylor, N. Cristianini, *Kernel methods for pattern analysis*, Cambridge, UK: Cambridge University Press, 2004.

[33] J. Kennedy, R. Eberhart, "Particle Swarm Optimization," in *Proceedings of the Fourth IEEE International Conference on Neural Networks, Perth, Australia, IEEE Publishing*, 1995, vol. 4, pp. 1942–1948.

[34] M. Clerc, "Standard particle swarm optimisation: from 2006 to 2011," Technical Report, Accessed: Nov. 5, 2015. [Online]. Available: http://clerc.maurice.free.fr/pso/SPSO_descriptions.pdf/.

[35] D. Freedman, R. Pisani, R. Purves, *Statistics*, New York, USA: WW Norton & Company, 2007.

[36] R. Picard, D. Cook, "Cross-validation of regression models," *Journal of the American Statistical Association*, vol. 79, no. 387, pp. 575–583, 1984.

[37] C.–C. Chang, C.–J. Lin, "LIBSVM: a library for support vector machines," ACM Transactions on Intelligent Systems and Technology, vol. 2, pp. 1–27, 2011.

Paulino José García Nieto

Paulino J. García Nieto was born in Oviedo, Spain, in1965. He received the Bachelor's degree in mining engineering with specialization in fuels and energy and the Master's and Ph.D. degrees in mining engineering from the University of Oviedo, in 1989, 1990, and 1994, respectively. Since 1996, he has been with the Department of Mathematics, University of Oviedo, where he is currently a Full Professor. He was involved in research on numerical simulation, as well as the application of finite-element methods in numerous physical and engineering problems. He is currently involved in the application of statistical learning and data mining to several biological, electrical, and physical data sets to obtain predictive models.

Esperanza García-Gonzalo

Esperanza García Gonzalo was born in Soria, Spain, in 1964. She received the M.Sc. degree in mining engineering in 1990 and the Ph.D. degree in mathematics from Oviedo University, Asturias, Spain, in 2010. She is an associate professor at the Mathematics Department (Oviedo University. Her research interests include global stochastic optimization methods, evolutionary algorithms, particle swarm optimization and machine learning.

Fernando Sánchez Lasheras

Fernando Sánchez Lasheras was born in Oviedo, Spain, in 1975. He received the M.Sc. and Ph.D. degrees in industrial engineering from the University of Oviedo, in 2000 and 2008, respectively. In 2017, he joined the Department of Mathematics of Oviedo University. His current research interests include applied mathematics, machine learning and artificial intelligence, with about a hundred of technical publications.

Antonio Bernardo Sánchez

Antonio Bernardo Sánchez was born in Oviedo, Spain, in 1961. He received the M.Sc. degree in mining engineering in 1988 and the Ph.D. from Oviedo University, Asturias, Spain, in 1994. Since 1990 to 2014, he was an associate professor at the Construction and Manufacturing Engineering Department (Oviedo University). Since 2014, he is an associate professor at the Department of Mining Technology, Topography and Structures, University of León, (Spain). His research interests include energy storage, mining technology, materials and the application of numerical simulation and data mining in several engineering problems.