MATCH
Communications in Mathematical and in Computer Chemistry

MATCH Commun. Math. Comput. Chem. **83** (2020) 295-327

ISSN 0340 - 6253

Determination of Activation Thermodynamic Parameters by Means of the Application of the Neural Networks–AGDC Hybrid Algorithm

J. L. González-Hernández, M. Mar Canedo*, Sonsoles Encinar

Department of Physical Chemistry, Faculty of Chemistry, University of Salamanca, E-37008 Salamanca, Spain

(Received May 15, 2019)

Abstract

In this work the hybrid algorithm, ANN-AGDC, is used for the determination of activation thermodynamic parameters (ATPs) of diverse chemical reactions by means of the treatment of non-isothermal experimental kinetic data.

The hybrid algorithm consists of two different algorithms that are sequentially applied, the artificial neural networks (ANN) method and the algorithm of mathematical optimization AGDC. The ANN method is used to determine the ATPs in the chemical reactions and the obtained values are utilized as initial estimations for the second stage that consist of a mathematical optimization process by the AGDC algorithm. The applicability of the algorithm for the determination of activation thermodynamic parameters has been proven by the treatment of synthetic data generated for different reaction mechanisms which have been provided with an error in the order of the experimental. The treatment of synthetic data has allowed to successfully obtain the ATPs thus, in this work we apply the hybrid algorithm for the treatment of non-isothermal experimental kinetic data from the isomerization reaction of the steroid 5-cholesten-3-ona to 4-cholesten-3-ona catalyzed by sodium ethoxide and from the kinetics for the breakdown of the trinuclear chromium acetate cluster with a series of monoprotic and diprotic amino acid ligands. In both cases the ATPs involved in the Arrhenius and Eyring equations have been directly determined from the non-isothermal experimental kinetic data: activation energy (E), pre-exponential factor (A), activation enthalpy ($\Delta H^{\neq}$) and the activation entropy ($\Delta S^{\neq}$). The application of hybrid algorithm has a great advantage because it is not necessary to give out the initial ATPs estimations, since the ANN method, first stage of the method, does not need these values unlike the mathematical optimization methods.

(*) ***Author for correspondence***
e-mail: mcanedo@usal.es

1 Introduction

The determination of the activation thermodynamic parameters (ATPs) of a chemical reaction can be carried out either from isothermal kinetic data (isothermal method) or non-isothermal kinetic data (non-isothermal method).

The isothermal method first determines the values of the rate constants *k(T)* from isothermal experimental kinetic data obtained in kinetic experiments at different temperatures. From the *k(T)* values the ATPs are determined by linearization of the Arrhenius and Eyring equations [1]. The values of the pre-exponential factor (*A*) and the activation energy (*E*) are determined for each step of the reaction mechanism from each individual value of *k(T)* from Arrhenius equation and the activation enthalpy ($\Delta H^{\neq}$) and the activation entropy ($\Delta S^{\neq}$) from the Eyring's equation.

In the case of a system formed by *"r"* chemical reactions with rate constants $k_r(T)$, the system of differential equations is established (ODE) and solved to obtain the explicit mathematical expressions of the concentrations of all the species involved in the kinetic system as a function of $k_r(T)$ and time, *t* [2]. The values of $k_r(T)$ can be calculated by fitting the isothermal kinetic data obtained at different temperatures to these explicit functions and the ATP's values are determined from the pairs of values k_r/T. Sometimes, the ODE system doesn't provide exact mathematical solutions, so the $k_r(T)$ cannot be directly obtained and, therefore, the values of the ATPs cannot be determined. Approximate methods can be used to determine the rate constants but in this case the values of the ATPs obtained are not exact. In addition, this methodology requires a large amount of experimental work at the laboratory which is needed to determine a reduced collective pair of values k_r/T.

The non-isothermal method does not need to previously determine the values of the rate constants, but it is necessary to obtain experimental data from a single non-isothermal kinetic experience where the temperature changes over time in the course of the reaction, which implies introducing one more variable (temperature). This method provides a more numerous set of data and more accurate final results.

The determination of the parameters involved in mathematical functions is usually carried out by means of mathematical optimization methods applying numeric second order gradient algorithms [3]. The computational methods found in the bibliography that use different optimization algorithms, need initial estimates of the parameters to be determined. Sometimes the order of magnitude of the parameters is unknown, which causes many problems. In many cases, only if the initial estimates are very close to the real values of the parameters, the optimization process is carried out correctly. If the initial estimates are far from the real values the process may become divergent, leading the optimization process to fail. Taking into account the importance of the initial estimates, it is convenient to use a method that provides values which approximate to the global minimum and then use such results as a starting point to apply a robust gradient method that will guarantee the success of the mathematical optimization of parameters.

In previous works we have designed a new hybrid algorithm (HA) that has been applied to different reaction mechanisms for the determination of different parameters. A hybrid algorithm consists of two different algorithms that solve the same problem, in this case it is formed by the artificial neural network method (ANN) and the AGDC optimization algorithm.

The ANN method is a versatile "*soft modeling*" method that can be applied in diverse fields with acceptable results [4-5]. This method has been used in chemical kinetics for quantitative purposes for the treatment of kinetic data from simple reactions [6-8]. The ANN methodology provides acceptable results in the case of determining the individual rate constants. ANN does not need initial estimates of the parameters to determine, through a *training* process and subsequent *prediction*, it determines the values of the parameters. These values have been used as the initial estimates of a robust and efficient numeric second order gradient algorithm (AGDC) [9-13], which is able to reach the desired global minimum to guarantee the success of the final optimization of the parameters. The AGDC mathematical optimization algorithm is a symbolic second-order gradient method that performs a rigorous analysis and control of the movement vector and each of its terms. This algorithm has been used successfully for the determination of kinetic, analytical and thermodynamic parameters in different reaction mechanisms [9-13].

The hybrid algorithm (HA) used in this work consists of a combination of ANN and AGDC methods and their applicability has been verified by the treatment of non-isothermal kinetic data synthetically generated for different reaction mechanisms [14-15]. In the first work, the HA is applied for the determination of activation energy and the pre-exponential factor in a simple chemical reaction [14]. Subsequently, it was used for the determination of ATPs in a system of consecutive reactions [15], in this case the number of parameters to be determined is increased considerably since the activation energy, the pre-exponential factor is determined in addition to the enthalpy and entropy of activation of both steps.

This hybrid algorithm has been applied to determine the individual rate constants that correspond to three different reaction models in which several different species, the reactions between the species and the rate constants are involved [16]. The methodology has the capacity of discrimination between the different models that are theoretically applicable to the chemical reaction.

Just like we mentioned, we have verified the hybrid algorithm applicability for the determination of the ATPs through the treatment of kinetic data generated synthetically. In this work, the HA is applied for the treatment of non-isothermal experimental kinetic data from two reactions: the isomerization reaction of 5-cholesten-3-ona to 4-cholesten-3-ona catalyzed by sodium ethoxide [17-18] and the reaction based on the rupture of the trinuclear chromium acetate cluster with a series of monoprotic and diprotic amino acids that act as a ligand in an aqueous medium [19-20]. In both cases, the activation energy values, pre-exponential factor, enthalpy and entropy of activation of the different stages that constitute the mechanisms of these chemical reactions are determined.

The application of the hybrid algorithm has a series of advantages; it is not necessary to previously determine the rate constants, it is not necessary provide estimated values for the parameters to be determined and it is possible to determine a high number of parameters. The treatment requires to acquire the kinetic data from non-isothermal kinetic experiments, imposing a controlled variation of temperature along the reaction kinetic.

2 Theoretical aspects

2.1 Chemical kinetics aspects

In a chemical system formed by n_r chemical elementary reactions where n_s chemical species are involved, the *r-th* reaction can be expressed with the generic equation [21]:

$$0 = \sum_{j=1}^{n_s} \nu_{j,r} B_j \qquad (1)$$

where: B_j= chemical species involved in the system of reactions; $r = (1, \ldots, n_r)$ number of chemical reactions; $j = (1, \ldots, n_s)$ number of chemical species; k_r= kinetic rate constant of the *r-th* reaction and $\nu_{j,r}$= stoichiometric coefficient of the species B_j in the *r-th* reaction ($\nu_{j,r} < 0$ when B_j plays only the role of reactant in the *r-th* reaction and $\nu_{j,r} > 0$ when B_j plays only the role of product in the *r-th* reaction).

The concentration of each species involved in the mechanism of the reaction varies with respect to the time *i*, this variation is given by a general differential equation:

$$\frac{d[B_j]}{dt} = \sum_{r=1}^{n_r} k_r \nu_{j,r} \prod_{l=1}^{n_s} [B_l]^{|\nu_{l,r}|} \qquad (2)$$

where: k_r=rate constant of the reaction *r*; $\nu_{l,r}$=stoichiometric coefficient of the species B_l in the reaction *r* and $[B_l]$=concentration of the species that act as reagents in the reaction *r* ($\nu_{l,r}$< 0).

The different species involved in the mechanism of the reaction provide a differential equation and therefore we have a system of differential equations (ODEs) whose resolution provides the concentration of each of species with respect to time ($[B_j]_{t_i}$).

The chemical reactions that have been studied in the present work can be expressed schematically according to the following models:

Model I $\mathbf{B_1 \xrightarrow{k_{12}} B_2}$

In a non-isothermal process, the rate constants of each of the stages that constitutes a mechanism ($k_r(T)$) are functions of the temperature and consequently of time, therefore; the variation of the reactant concentration is given by:

$$-\frac{d[B_1]}{dt} = k_{12}(T)[B_1] \tag{3}$$

Separating variables and integrating the first member of the equation, we have:

$$-\ln\left(\frac{[B_1]}{[B_1]_0}\right) = \int_0^t k_{12}(T)dt \tag{4}$$

where: $[B_1]_0$ = initial concentration of B_1, $[B_1]$ = concentration at time *t* and $k_{12}(T)$ = rate constant, function of the temperature *T* in non-isothermal conditions.

Considering the remaining molar fraction of the reactant B_1 (α_1) given by equation 5 and according to the Arrhenius equation, we obtain equation 6:

$$\alpha_1 = \frac{[B_1]}{[B_1]_0} \tag{5}$$

$$-ln\,\alpha_1 = \int_0^t A_{12}e^{-\frac{E_{12}}{RT}}dt \tag{6}$$

where E_{12} is the activation energy and A_{12} is the pre-exponential factor. This procedure allows the determination of the activation parameters E_{12} and A_{12}.

In non-isothermal conditions, the second member of equation 6 cannot be integrated since there are two dependent variables (*T=f(t)*). It is very important to establish the variation of the temperature with the time because the mathematical method of resolution of the differential equation 6 will be different. Temperature *vs* time function (*T vs t*) must correspond to a continuous, monotonous increasing function and responding to a profile variation according to adequate temporary dependence of temperature, since it is necessary to monitor it in the laboratory. We consider two possibilities for the function *T=f(t)* in order to choose the variation temperature *vs* time:

a) Hyperbolic variation, in this case the inverse function of temperature corresponds to the hyperbolic branch:

$$\frac{1}{T} = \frac{1}{T_0} - mt \tag{7}$$

Then substituting in the equation 6 we have:

$$-\,ln\,\alpha_1 = \int_0^t A_{12} e^{-\frac{E_{12}}{R}\left(\frac{1}{T_0}-mt\right)}\,dt \tag{8}$$

Solving this integral, the exact mathematical solution is obtained:

$$ln\,\alpha_1 = -\,\frac{A_{12}R}{mE_{12}} e^{-\frac{E_{12}}{RT_0}}\left(e^{\frac{mE_{12}}{R}t} - 1\right) \tag{9}$$

Representing the equation explicitly will remain as:

$$\alpha_1 = e^{-\frac{A_{12}R}{mE_{12}} e^{-\frac{E_{12}}{RT_0}}\left(e^{\frac{mE_{12}}{R}t} - 1\right)} \tag{10}$$

The explicit function of α_1 is not simple and it is necessary the application of a method of sufficiently robust treatment for the determination of the thermodynamic parameters E_{12} and A_{12} [14]. In addition, it is necessary to consider the great difference in the magnitude order of both parameters that complicates extraordinarily the success in the application of the method of treatment. On the other hand the experimental points (*T/t*) must satisfy this hyperbolic function and therefore, it is necessary to reproduce the profile of the curve of the function (*T/t*) with the points obtained in the laboratory.

b) Polynomial variation of *n-th* degree:

$$T = \sum_{j=0}^{j=n} a_j t^j \tag{11}$$

In this case substituting in the equation 6 we have:

$$-\ln\alpha_1 = \int_0^t A_{12}\, e^{{-E_{12}}/{R\left(\sum_{j=0}^{j=n} a_j t^j\right)}} dt \tag{12}$$

The differential equation 12 has no exact mathematical solution, so it is necessary to resort to other treatments such as numerical integration or numerical resolution of differential equation 3 expressed in terms of Arrhenius' equation:

$$d[B_1] = -[B_1]A_{12}e^{-E_{12}/_{RT}}dt \tag{13}$$

If we consider that the variation of k_{12} with T is given by Eyring's equation and we substitute that variation on equations 3 and 4, we can determine the enthalpy and entropy of activation ($\Delta H_{12}^{\neq}$ and $\Delta S_{12}^{\neq}$):

$$d[B_1] = -[B_1]\left(\frac{k_B}{h}\right)Te^{\frac{-\Delta H_{12}^{\neq}}{RT}}e^{\frac{\Delta S_{12}^{\neq}}{R}}\,dt \tag{14}$$

$$-\ln\alpha_1 = \int_0^t \left(\frac{k_B}{h}\right)Te^{\frac{-\Delta H_{12}^{\neq}}{RT}}e^{\frac{\Delta S_{12}^{\neq}}{R}}\,dt \tag{15}$$

The integral's solution depends on the explicit function $T = f\ (t)$, in this case we exclusively consider the polynomic function of *n* degree (equation 11) since in the temperature inverse function's case (equation 7) the integral does not possess an exact mathematical solution. The treatment to be performed is identical in the Arrhenius' equation case considering two possibilities: the numerical integration of the equation using formulas of appropriate quadrature or by performing the numerical resolution of the differential equation 15 by means of the application of suitable methods for the treatment of stiff systems.

Model II $\mathbf{B_1 \xrightarrow{k_{12}} B_2 \xrightarrow{k_{23}} B_3}$

The system of ordinary differential equations for this mechanism can be expressed as:

$$\begin{aligned}\frac{d[B_1]}{dt} &= -k_{12}(T)[B_1]\\ \frac{d[B_2]}{dt} &= k_{12}(T)[B_1] - k_{23}(T)[B_2]\\ \frac{d[B_3]}{dt} &= k_{23}(T)[B_2]\end{aligned} \tag{16}$$

The first stage of *model II* corresponds to *model I*, so the treatment for the determination of the activation thermodynamic parameters (E_{12}, A_{12}, $\Delta H_{12}^{\neq}$ and $\Delta S_{12}^{\neq}$) which is similar to the one seen for *model I*. The resolution of the differential equations 16 in isothermal conditions, provide the following equations:

$$[B_1] = [B_1]_0e^{-k_{12}(T)t} \tag{17}$$

$$[B_2] = \frac{[B_1]_0 k_{12}}{k_{23} - k_{12}} \left(e^{-k_{12}(T)t} - e^{-k_{23}(T)t}\right) \tag{18}$$

$$[B_3] = [B_1]_0 \left(1 - \frac{k_{23}}{k_{23} - k_{12}} e^{-k_{12}(T)t} + \frac{k_{12}}{k_{23} - k_{12}} e^{-k_{23}(T)t}\right) \tag{19}$$

In non-isothermal conditions, if we introduce the dependence of $k_{12}(T)$ and $k_{23}(T)$ in equations 16 as a function of the parameters of the Arrhenius and Eyring equations, we obtain a system of differential equations that lacks an exact mathematical solution. Therefore there are non-explicit equations of $\alpha_1(t_i)$, $\alpha_2(t_i)$ and $\alpha_3(t_i)$ that depend on the activation thermodynamic parameters and time. The solutions of the differential equations 16 are exclusively numerical, which means that we will only obtain discreet values of $\alpha_1(t_i)$, $\alpha_2(t_i)$ and $\alpha_3(t_i)$ for each value of time. For this reason, it is necessary to determine $k_{12}(T_i)$ and $k_{23}(T_i)$ for each value of T_i, substituting the values of t_i in the following equations:

$$k_{12}(T_i) = A_{12}\, e^{{}^{-E_{12}}\!/_{R \sum_{j=0}^{j=n} a_j t_i^j}} = \left(\frac{k_B}{h}\right) \sum_{j=0}^{j=n} a_j t_i^j \, e^{\frac{-\Delta H^{\neq}_{12}}{R} \sum_{j=0}^{j=n} a_j t_i^j e^{\frac{\Delta S^{\neq}_{12}}{R}}} \tag{20}$$

$$k_{23}(T_i) = A_{23}\, e^{{}^{-E_{23}}\!/_{R \sum_{j=0}^{j=n} a_j t_i^j}} = \left(\frac{k_B}{h}\right) \sum_{j=0}^{j=n} a_j t_i^j \, e^{\frac{-\Delta H^{\neq}_{23}}{R} \sum_{j=0}^{j=n} a_j t_i^j e^{\frac{\Delta S^{\neq}_{23}}{R}}} \tag{21}$$

The values of A_{12}, E_{12}, $\Delta H^{\neq}_{12}$ and $\Delta S^{\neq}_{12}$ involved in the equation 20 can be previously determined and should be constant in the subsequent determination and optimization processes of A_{23}, E_{23}, $\Delta H^{\neq}_{23}$ and $\Delta S^{\neq}_{23}$ through the application of the HA to the equation 21. At last, the numerical values of $\alpha_2(t_i)$ and $\alpha_3(t_i)$ will be determined from equations 22 and 23:

$$\alpha_2(t_i) = \frac{k_{12}(T_i)}{k_{23}(T_i) - k_{12}(T_i)} \left(e^{-k_{12}(T_i)t_i} - e^{-k_{23}(T_i)t_i}\right) \tag{22}$$

$$\alpha_3(t_i) = \left(1 - \frac{k_{23}(T_i)}{k_{23}(T_i) - k_{12}(T_i)} e^{-k_{12}(T_i)t_i} + \frac{k_{12}(T_i)}{k_{23}(T_i) - k_{12}(T_i)} e^{-k_{23}(T_i)t_i}\right) \tag{23}$$

These two pairs sets of kinetic values $[\alpha_2(t_i)/t_i]$ and $[\alpha_3(t_i)/t_i]$ constitute the data for the application of the hybrid algorithm (ANN-AGDC), which will allow us to simultaneously establish the values of the eight activation thermodynamics parameters (A_{12}, E_{12}, $\Delta H_{12}^{\neq}$, $\Delta S_{12,}^{\neq}$ A_{23}, E_{23}, $\Delta H_{23}^{\neq}$ and $\Delta S_{23}^{\neq}$).

2.2 Hybrid Algorithm (ANN-AGDC)

In this paper we utilize a hybrid algorithm (HA) designed in our laboratory to determine the activation thermodynamics parameters (ATPs) in two reaction mechanisms. A hybrid algorithm is one in which two or more different algorithms are combined in order to solve the same mathematical problem. The HA utilized in this paper [14-16], utilizes two methods based on different mathematical principles and it is applied sequentially in two steps. First the methodology *soft-modelling* based on artificial neural networks (ANN) and later the AGDC mathematical optimization algorithm is applied. The ANN methodology has a great advantage, since it is not necessary to provide initial estimates of the parameters to determine. The results obtained with ANN (output matrix), are the initial estimates used to start the optimization process with the AGDC optimization algorithm, continuing with the second step in the treatment.

2.2.1 Artificial Neural Networks (ANN)

The artificial neural networks consisting of a large number of simple processors with many interconnections. An ANN is constituted by a set of activation units called nodes or artificial neurons that are connected to each other through a network and are structured in layers [22].

There are a wide variety of ANNs, a multilayer neural network is an oriented graph in which the nodes represent a set of processing units (neurons) and the connections represent the information flow channels. Basically, they consist of inputs which are multiplied by weights which are computed by a mathematical function which determines the activation of the neuron. Another function (which may be the identity) computes the output of the artificial neuron. The neurons sum their inputs and since the input neurons have only one input, their output will be the input they received multiplied by a weight. By adjusting the weights of an artificial neuron using the appropriate algorithm, the desired output of the network can be obtained for specific inputs. This process of adjusting the weights is called *learning* or *training*.

When MATLAB [24] is applied, the multilayer neural network uses sets of input data and parameters (called *targets)* distributed in 2 *input* matrices. The elements of the *target* matrix are sets of parameters (*n_p*) that give rise to the data contained in the *input data* matrix, where one row contains a single curve and all the curves thus obtained (n_c) are grouped in an *input data* matrix. In our case, the *input data* matrix contained the kinetic data of all curves expressed in $\alpha_i(t_i)$ (remaining molar fraction) and the *target* matrix ($n_c x n_p$) contained the set of ATPs.

The elements of the *input data* matrices provide the information to the neurons that form the first layer of a multilayer neural network and are transmitted from the i-*th* neuron of a layer to the j-*th* neuron of the subsequent one, with a weight w_{ji}. The *bias* is summed with the weighted inputs of the neurons and passed through the transfer function to generate the output of the neurons. The next layer is the *hidden* one, the weighted inputs of each of its neurons coming from the previous layer are summed with each other and added to a *bias*. Then, the result is transformed by means of a mathematical function and an output called *activation of the neuron* is obtained, which is transferred to the neurons in the next layer after another weighing step. In the last layer (*output*) the output parameters values are calculated by means of a suitable transformation function. This process is called *training* or *learning* of the neural network and it is an iterative method where after each iteration (*epochs*), the calculated values of the parameters are grouped in the *output* matrix (b_{ij}^{output}) and they are compared with those of the corresponding curve in the *target* matrix (b_{ij}^{target}).

The value of the mean squared error (*MSE*), expressed in absolute value, is calculated according to the expression:

$$MSE = \left(\frac{\sum_{i=1}^{n_p} \sum_{j=1}^{n_c} (b_{ij}^{output} - b_{ij}^{target})^2}{n_p . n_c}\right)^{½} \qquad (24)$$

where n_c is the number of curves, n_p is the number of parameters and $n_c x n_p$ are the dimensions of both matrices (*output* matrix and *target* matrix). During the process of *training* the values of *weights* and *bias* are modified by means of suitable mathematical optimization algorithms in order to minimize the calculated values of *MSE* in each *epoch*. The *back-propagation* algorithm was used in the present work [22]. The iterative process finishes when the minimum value of *MSE* is reached, after which the *training* process can be considered to be completed.

The neural network has to be designed for each specific system which means search for the optimal architecture, that is, the optimal number of layers (structure) and the number of nodes in each layer (configuration).

The steps to follow in order to establish the optimal architecture of the neural network are the following:

a) Design of a network:
   i) Establish the number of layers *n* ((*n-1*) *hidden* + 1 *output*).

ii) Establish the number of neurons from the *hidden* layers.

b) *Training* of the ANN, process during which the value of *MSE* is minimized.

When the *training* has finished the errors of *training*, *validation* and final *testing* can be seen and a series of parameters (general *MSE*, regression between the *outputs* and *targets* values, number of iterations (*epochs*), *performance*, the value of the gradient vector of the Levenberg-Marquardt method (value of μ), number of *validation* checks…etc) that will allow us to deduct if the ANN *training* has been successfully carried out or not. The result is acceptable if the following requirements are met: The mean square error (*MSE*) is significantly small, the *validation* and *testing* errors have similar characteristics, *Overtraining* does not happen and there is a satisfactory evaluation of the ANN's response, producing a linear regression between the corresponding network *outputs* and the *targets*.

The analysis of all of these factors allows to decide if the *training* has been successfully performed. If that is the case, the network is ready to carry out *prediction* tasks and in the opposite case, we can improve the results taking into account the following considerations: Reestablish the initial ANN and train it again, *retrain*, Increase the number of neurons of the *hidden* layers (different configuration), increase the number of *hidden* layers, increase the number of *training* vectors, modifying the experiences percentage for *training*, *testing* and *validation* or use a more robust optimization algorithm that minimizes the differences between the values of the *outputs* and *targets* matrices.

When the processes of *validation* and *testing* reach satisfactory results the neural network *training* is completed. These are two control and verification processes of the iterative minimization method between the elements of the *output* and *target* matrices. Among the different curves comprising the *input* matrix, a random choice is made of a percentage of the total, established previously (5%,10%...), which gives rise to a *sub-matrix* of *input* curves that are subjected to iterative optimization until a minimum *MSE* value is reached. It is thus possible to verify the validity of the *training* process by ensuring that it is convergent, that it has an appropriate termination, and that there has not been any *overfitting*, since any possible *overtraining* has been taken into account. *Validation* is completed when in a given number (≥6) of consecutive *epochs* the *MSE* remains constant or shows a slight tendency to increase. The *testing* process is similar, except that the control of the process is performed by controlling the computation time instead of the number of *epochs*.

Once the optimal network has been obtained (network *training* is completed), the process of *prediction* is carried out. This process consists in the determination of the unknown parameters from a set of experimental data after the application of the optimal and trained neural network. In the case of the *prediction* process, the elements of the *target* matrix are unknown and only the *input data* matrix is provided to the neural network. In our case, the elements of the *input data* matrix in the process of *prediction* are experimental kinetic values of remaining molar fraction of the reactant ($\alpha_i(t_i)$), acquired from a system of reactions developed at the laboratory.

2.2.2 AGDC Algorithm

The values of the activation thermodynamic parameters (ATPs) determined by ANN (*outputs*) are used as initial estimates for the application of the following method that forms the hybrid algorithm. This second method is the AGDC [9-13], a mathematical optimization algorithm that allows the determination of different parameters by means of a second-order gradient method that minimizes the numerical function (SQD) given by:

$$\mathrm{SQD}(\mathbf{X}) = \sum_{i=1}^{N_d} \left(\left(\alpha_j(t_i) \right)_C - \left(\alpha_j(t_i) \right)_E \right)^2 \qquad (25)$$

$$\alpha_j(t_i) = \left[B_j\right]_i / [B_1]_0 \qquad (26)$$

where **X** is the vector that contains the parameters to be optimized according to the reaction mechanism studied (E_{12} and A_{12}, $\Delta S_{12}^{\neq}$ and $\Delta H_{12}^{\neq}$, E_{23} and A_{23} or $\Delta S_{23}^{\neq}$ and $\Delta H_{23}^{\neq}$).

Initially the algorithm AGDC uses, as the movement vector, the one indicated by the Gauss-Newton method [9-10]:

$$\boldsymbol{p}^{(m)} = -\boldsymbol{g}^{(m)} \left[\boldsymbol{H}^{(m)}\right]^{-1} \qquad (27)$$

where $\mathbf{p}^{(m)}$ is the movement vector, $\mathbf{g}^{(m)}$ is the gradient vector and $[\mathbf{H}^{(m)}]^{-1}$ the inverse of the Hessian matrix of the iteration *m*, whose terms are derived from the function to be minimized (SQD) with respect to each of the parameters to be determined (**X**).

The residuals are given by:

$$RES_i = \left(\alpha_j\right)_C - \left(\alpha_j\right)_E \qquad (28)$$

then $\mathbf{g}^{(m)}$ and $\mathbf{H}^{(m)}$ are given by:

$$\mathbf{g} = 2\begin{bmatrix} \sum_{i=1}^{N_d} RES_i \frac{\partial(\alpha_j)_C}{\partial X_1} \\ \sum_{i=1}^{N_d} RES_i \frac{\partial(\alpha_j)_C}{\partial X_2} \end{bmatrix} \tag{29}$$

$$\mathbf{H} = 2\begin{bmatrix} \sum_{i=1}^{N_d} \left(\frac{\partial(\alpha_j)_C}{\partial X_1}\right)^2 & \sum_{i=1}^{N_d} \left(\frac{\partial(\alpha_j)_C}{\partial X_1}\right)\left(\frac{\partial(\alpha_j)_C}{\partial X_2}\right) \\ \sum_{i=1}^{N_d} \left(\frac{\partial(\alpha_j)_C}{\partial X_2}\right)\left(\frac{\partial(\alpha_j)_C}{\partial X_1}\right) & \sum_{i=1}^{N_d} \left(\frac{\partial(\alpha_j)_C}{\partial X_2}\right)^2 \end{bmatrix} \tag{30}$$

The numerical derivatives are calculated by the method of central differences [23] and the inverse of the Hessian matrix is computed by means of the MATLAB application that computes the inverse of a square matrix using LU factorization [24]. The AGDC algorithm performs a control of the movement vector and carries out appropriate modifications if errors are detected, which guarantees a successful optimization. Once the optimization process has been completed, the errors of the optimized parameters are determined [25] and an exhaustive analysis of the residuals is carried out, which allows verifying the goodness of fit [26].

The application of the AGDC algorithm for the determination of the ATPs in the mechanisms considered, can be written schematically as follows:

Iteration m → Input data: $\mathbf{X}^{(m)}$, $SQD^{(m)}$

1. Calculate the vector of movement.

1.1. Compute partial numerical derivatives of $(\alpha_j)_C$ with respect to the parameters to be determined $\boldsymbol{X_p}$ [23], $\left(\partial(\alpha_j)_C/\partial X_p\right)^{(m)}$

1.2. Compute Gradient vector and Hessian Matrix ($\mathbf{g}^{(m)}$ and $\mathbf{H}^{(m)}$) (29), (30).

1.3. Compute $(\mathbf{H}^{(m)})^{-1}$ [24].

1.4. Calculate the components of the vector of movement $\mathbf{p}^{(m)}$ (27).

2. Control and correction of the direction of the vector of movement $\mathbf{p}^{(m)}$.

2.1. If $\mathbf{H}^{(m)}$ is singular, $\mathbf{p}^{(m)} = -\mathbf{g}^{(m)}$, go to 3.

2.2. If $\mathbf{p}^{(m)}\,\mathbf{g}^{(m)} < \varepsilon$ (ε = scalar close to zero), $\mathbf{p}^{(m)} = -\mathbf{g}^{(m)}$ and go to 3.

2.3. If $\mathbf{p}^{(m)}\,\mathbf{g}^{(m)} > 0$, $\mathbf{p}^{(m)} = -\mathbf{p}^{(m)}$.

3. Control the length of the vector of movement $\mathbf{p}^{(m)}$.

3.1. Compute the scalar ($\lambda^{(m)}$) by the method of Hartley [27].

3.2. $\boldsymbol{X}^{(m+1)} = \boldsymbol{X}^{(m)} + \lambda^{(m)}\boldsymbol{p}^{(m)}$.

3.3. Determinate the $SQD^{(m+1)}$ function (25).

3.4. If the Goldstein-Armijo criterium [28] is satisfied go to 4.

3.5. $\lambda^{(m)} = \lambda^{(m)}/2$ go to 3.2.

4. Calculate:

$$CON = \left|\frac{SQD^{(m+1)} - SQD^{(m)}}{SQD^{(m)}}\right| \tag{31}$$

5. If convergence is not attained (CON > CC), set m = m + 1 and go to 1.

6. $\mathbf{X}^{(m+1)}$ = Optimized Parameters.

7. END optimization.

The procedure followed to carry out the optimization of the activation parameters is schematically expressed as follows:

0. Input data: Experimental data of $(\alpha_j)_E/t_i$, $[B_j]_0$, Convergence Criteria (CC), Initial estimates of the unknown parameters $\mathbf{X}^{(0)}$: $E_{12}^{(0)}$, $A_{12}^{(0)}$, $\Delta S_{12}^{\neq(0)}$, $\Delta H_{12}^{\neq(0)}$, $E_{23}^{(0)}$, $A_{23}^{(0)}$, $\Delta S_{23}^{\neq\,(0)}$, $\Delta H_{23}^{\neq\,(0)}$ (Values of *outputs* from ANN application).

I. Optimization of E_{12}, A_{12}, $\Delta S_{12}^{\neq}$ and $\Delta H_{12}^{\neq}$

Ia. $\mathbf{X} = [E_{12}, A_{12}]$.

m=0. $\mathbf{X}^{(0)} = [E_{12}^{(0)}, A_{12}^{(0)}]$.

Ia1. Calculate $(\alpha_1^{(0)})_C$ (26).

i) T =f (t) inverse hyperbolic (7) → Mathematical exact solution (10).

ii) T =f(t) polynomial of n-th degree (11) → Numerical integration of (12) or numerical solution of the rate differential equation (13).

Ia2. Calculate $SQD^{(0)}$ (25).

Ia3. AGDC Algorithm

Ia4. Optimized parameters: $\mathbf{X}^* = [E_{12}, A_{12}]$ → Calculate the errors of E_{12}, A_{12}.

Ib. $\mathbf{X} = [\Delta S_{12}^{\neq}, \Delta H_{12}^{\neq}]$.

m=0. $\mathbf{X}^{(0)} = [\Delta S_{12}^{\neq(0)}, \Delta H_{12}^{\neq(0)}]$.

Ib1. Calculate $(\alpha_1^{(0)})_C$ (26): T =f (t) Polynomial of n-th degree (11) → Numerical solution of the rate differential equation (14) or Numerical integration (15).

Ib2. Calculate $SQD^{(0)}$ (25).

Ib3. AGDC Algorithm

Ib4. Optimized parameters: **X*** = [$\Delta S_{12}^{\neq}$, $\Delta H_{12}^{\neq}$]→ Calculate the errors of $\Delta S_{12}^{\neq}$, $\Delta H_{12}^{\neq}$.

II. Optimization of E_{23}, A_{23}, $\Delta S_{23}^{\neq}$ and $\Delta H_{23}^{\neq}$

IIa. X = [E_{23}, A_{23}].

m = 0. $\mathbf{X}^{(0)}$ =[$E_{23}^{(0)}$, $A_{23}^{(0)}$].

IIa1. Calculate $(\alpha_2(t_i)^{(0)})_C$, $(\alpha_3(t_i)^{(0)})_C$ (26).

i) Calculate T_i (11).

ii) Calculate $k_{12}(T_i) = A_{12}e^{-E_{12}/R\sum_{j=0}^{j=n} a_j t_i^j}$.

iii) Calculate $k_{23}(T_i) = A_{23}^{(0)}e^{-E_{23}^{(0)}/R\sum_{j=0}^{j=n} a_j t_i^j}$.

iv) Numerical solution of the rate differential equations (16).

IIa2. Calculate $SQD^{(0)}$ (25).

IIa3. AGDC Algorithm

IIa4. Optimized parameters: **X*** = [E_{23}, A_{23}] → Calculate the errors of E_{23}, A_{23}.

IIb. X = [$\Delta S_{23}^{\neq}$, $\Delta H_{23}^{\neq}$].

m = 0. $\mathbf{X}^{(0)}$ = [$\Delta S_{23}^{\neq (0)}$, $\Delta H_{23}^{\neq (0)}$].

IIb1. Calculate $(\alpha_2(t_i)^{(0)})_C$, $(\alpha_3(t_i)^{(0)})_C$ (26).

i) Calculate T_i (11).

ii) Calculate $k_{12}(T_i) = (k_B/h)\sum_{j=0}^{j=n} a_j\, t_i^j\; e^{-\Delta H_{12}^{\neq}/R\sum_{j=0}^{j=n} a_j t_i^j} e^{\Delta S_{12}^{\neq}/R}$.

iii) Calculate $k_{23}(T_i) = (k_B/h)\sum_{j=0}^{j=n} a_j\, t_i^j\; e^{-\Delta H_{23}^{\neq(0)}/R\sum_{j=0}^{j=n} a_j t_i^j} e^{\Delta S_{23}^{\neq(0)}/R}$.

iv) Numerical solution of the rate differential equations (16).

IIb2. Calculate $SQD^{(0)}$ (25).

IIb3. AGDC Algorithm

IIb4. Optimized parameters: **X*** = [$\Delta S_{23}^{\neq}$, $\Delta H_{23}^{\neq}$]→ Calculate the errors of $\Delta S_{23}^{\neq}$, $\Delta H_{23}^{\neq}$.

III. Statistical analysis of residuals.

3 Computational aspects

The hybrid algorithm used in this work combines two different methods (ANN and AGDC), the computational treatment of these methods is carried out through the application of mathematical software MATLAB©. The computational treatment of ANN has been performed by means of the software of MATLAB *"Neural Networks Toolbox"* [24].We design and perform specific computational executable programs (##.m type) in the MATLAB environment using "M" language to obtain synthetic data with which to carry out the training of the network.

The second method, the mathematical optimization, was carried out with the AGDC algorithm by means of the computer program KINNOISOT(AGDC) [14-15] written in MATLAB language for the treatment of non-isothermal kinetics. The program is formed by a main program and a series of subprograms, in which the different treatments and calculations necessary for the optimization process of AGDC are carried out. It has been structured in the following parts:

1. ***Main program*** **KINNOISOT(AGDC)**

Performs the following functions:

1.1. Select the parameters to be optimized.

1.2. Select the method to calculate $(\alpha_j)_C$.

1.3. Input data: Estimates of the unknown parameters (values of *outputs* from ANN application), Experimental data of $(\alpha_j)_E/t_i$**,** Initial concentrations $[B_j]_0$, Parameter values whose value is known, Convergence criteria CC,..

1.4. Output data: Optimized Parameters E_{12}, A_{12}, $\Delta S_{12}^{\neq}$, $\Delta H_{12}^{\neq}$, E_{23}, A_{23}, $\Delta S_{23}^{\neq}$, $\Delta H_{23}^{\neq}$,

Errors of the parameters and statistical analysis.

2. ***Functions:***

2.1. fAGDC, function that performs the optimization process by applying the AGDC algorithm.

2.2. fEXACT, function that calculates $[B_1]_i$ and $(\alpha_1)_C$ from the exact mathematical solution.

2.3. fNUMINT1 function that generates and integrates numerically the equation 12 by means of the methods provided by MATLAB: *quad, quadl, quadv* or *quadgk* [8].

2.4. fNUMINT2 function that generates and integrates numerically the equation 15 by means of the methods provided by MATLAB: *quad, quadl, quadv* or *quadgk* [8].

2.5. fDIFEQ, establishes the set of differential rate equations.

2.6. fNUMSOL, solves the set of differential rate equations by means of the methods provided by MATLAB: *ode45, ode23, ode113* or *ode15s* [24].

2.7. fDERIV, calculates the numerical partial derivatives of the $(\alpha_j)_C$ with respect to the parameters to be optimized by means of the central differences method [23].

2.8. fINVER, calculates the determinant and performs the inversion of the Hessian matrix by means the MATLAB function *inv* [24].

2.9. fESTAD, function that determines the errors of the parameters and performs statistical analysis of the residuals [25-26].

4 Results and discussion

The applicability and efficiency of the hybrid algorithm (HA) ANN-AGDC, has been proved by means of the treatment of non-isothermal kinetic data generated (synthetic) for different reaction mechanisms [14-15]. The application of the HA to the synthetic data generated for these mechanisms, allowed to determine successfully the activation thermodynamic parameters.

In this work we have applied the HA for the treatment of experimental kinetic data (non-isothermal) from two chemical reactions which generally respond to the following reaction mechanisms.

4.1 *Model I* $\boldsymbol{B_1 \overset{k_{12}}{\longrightarrow} B_2}$

The first chemical reaction studied responds in a general way to *model I*, it is an isomerization reaction of the steroid 5-cholesten-3-ona (5CHOL) to 4-cholesten-3-ona (4-CHOL) catalyzed by sodium ethoxide in ethanol absolute medium. According to the experimental evidences [17-18], a reaction mechanism constituted by two reversible steps can be proposed:

$$5-CHOL+EtO^{-}\underset{k_{21}}{\overset{k_{12}}{\rightleftarrows}}3{,}5-E^{-}+EtOH\underset{k_{32}}{\overset{k_{23}}{\rightleftarrows}}4-CHOL+EtO^{-}$$

The intermediate specie 3,5-dienolate $(3{,}5-E^{-})$ is considered highly reactive so the global isomerization reaction can be written as:

$$5-CHOL\overset{k_{13}}{\longrightarrow}4-CHOL$$

We apply the HA to study the reaction in the direct path, meaning, when we start with 5-cholesten-3-ona and sodium ethoxide and we obtain 4-cholesten-3-ona. The non-isothermal kinetic data obtained at diverse temperatures (from 288.15K to 308.15K) will be correctly correlated through the appropriate equations derives from the Arrhenius and Eyring theories, which will allow us to determine the activation thermodynamic parameters.

4.1.1 Experimental design (ED)

The hybrid algorithm consists of two methods, ANN and AGDC, so first it is necessary to perform the experimental design that allows to generate a set of kinetic curves to carry out the ANN *training*. The suitable ED is of the *Central Star Composite Experimental Desing* (CSCED) type [14-15], whose factors are formed by a set of parameters E_{13}/A_{13} or $\Delta H_{13}^{\neq}/\Delta S_{13}^{\neq}$ and the responses are data of the base curve of *inputs* (α_1/t).

The experimental values for the activation thermodynamic parameters taken from the bibliography [27-28], corresponding to the central points from which the data will be generated to carry out the training process (*inputs* for *training*), are the following: E_{13}= 6.28.10^4 Jmol^{-1}, A_{13}=1.96.10^{10} min^{-1}, $\Delta H_{13}^{\neq}$= 6.03.10^4 Jmol^{-1} and $\Delta S_{13}^{\neq}$= -90.5 JK^{-1}mol^{-1}. Figure 1 shows the ED obtained for E_{13} and A_{13} and figure 2 corresponds to $\Delta H_{13}^{\neq}$ and $\Delta S_{13}^{\neq}$ (*targets* for *training*).

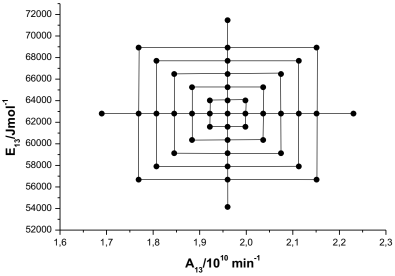


Figure 1. Experimental Design (CSCED) constituted by 2 factors (A_{13} and E_{23}) and 45 points distributed in 9 levels (points from 1-37) and 4 sub-levels (points from 38-45) that participate in the base curve of the *inputs* matrix used for the neural network *training* process.

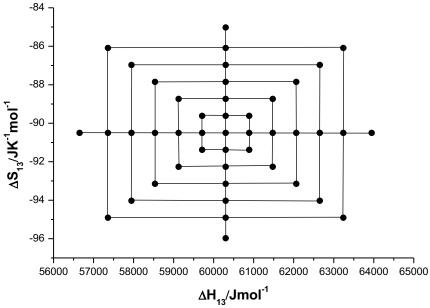


Figure 2. Experimental Design (CSCED) constituted by two *factors* ($\Delta H_{13}^{\neq}/\Delta S_{13}^{\neq}$) and 45 points distributed in 9 levels (points from 1-37) and 4 sub-levels (points from 38-45) that use as a base curve the *input* matrix used for the neural network *training* process.

4.1.2 Determination of E_{13} and A_{13}

4.1.2.1 *Training* of the ANN and *Prediction* of E_{13} and A_{13} through ANN

From the sets of E_{13} and A_{13} obtained in the corresponding ED (*targets* for *training*), we have generated a set of 45 non-isothermal kinetic curves α_1/t. These kinetic curves are formed by the remaining molar fraction of the reactant B_1 (α_1/t) considering an hyperbolic variation of temperature (equation 7). This data forms the matrix of *inputs* for *training* and in conjunction with the matrix of *targets* for *training* are supplied in the neural network to proceed with the *training*.

The ANN *training* process has the objective of finding an optimal network (optimal architecture) to carry out the *prediction* process of E_{13} and A_{13} for experimental kinetic experiences in which their value is unknown. We have carried out the *training* process for a great set of neural networks considering different possible configurations for the architectures with 2 and 3 *hidden* layers. In the first 5 columns of table 1, the most relevant results obtained in the *training* process of the ANN (made with the 45 curves obtained from the ED) and the subsequent *prediction* for the case of the kinetic data corresponding to the central point are shown. The experiences have been performed considering the same curve percentage relation for *training*, *validation* and *testing* in all cases (80/10/10). In the table 1, the values of *outputs*, E_{13}/A_{13} and the corresponding deviations (*"Des. %"*) [14-15] obtained after the application of different ANN can be seen. The optimal neural network is one that consists of three layers (structure) and with one configuration (number of nodes in each layer) of 10/7/10 even though the obtained values indicate that it is necessary to improve the results by means of a posterior optimization carried out in the second step of the hybrid algorithm (AGDC).

Table 1. Results of the *training* and *prediction* processes of the neural network constituted by an architecture with 2 and 3 *hidden* layers and different configurations, when A_{13} and E_{13} are computed.

ANN					**HA (ANN - AGDC)**			
Config.	**A_{13} /10^{10} min^{-1}**	**E_{13} /10^4 Jmol^{-1}**	**Des. % (A_{13})**	**Des. % (E_{13})**	**A_{13}/10^{10} min^{-1}**	**E_{13} /10^4 Jmol^{-1}**	**Des. % (A_{13})**	**Des. % (E_{13})**
10/14	1.9299	6.0621	1.5351	3.4686	1.9326	6.2762	1.3986	$6.0026.10^{-2}$
8/10/10	1.9201	6.4747	2.0318	-3.1006	1.9437	6.2750	$8.3007.10^{-1}$	$7.9505.10^{-2}$
10/7/10	**1.9444**	**5.7361**	**$7.9334.10^{-1}$**	**8.6607**	**1.9577**	**6.2781**	**$1.2007.10^{-1}$**	**$3.0918.10^{-2}$**
10/10/11	1.9449	6.5542	$7.6889.10^{-1}$	-4.3663	1.9463	6.5410	$6.9693.10^{-1}$	-4.1568
10/10/12	2.1382	5.7873	-9.0962	7.8449	1.9673	6.3012	$3.7003.10^{-1}$	$-3.3752.10^{-1}$
10/10/14	1.9279	6.2998	1.6375	$-3.1628.10^{-1}$	1.9566	6.2760	$1.7006.10^{-1}$	$6.4006.10^{-2}$

The joint representation of the *outputs/targets* regressions of A_{13} and E_{13} (figure 3) shows that they are widely separated within the graph, as a result of which they present very different orders of magnitude ($E_{13} \approx 10^4$ and $A_{13} \approx 10^{10}$). A poor linearity is observed in the case of the A_{13} deviations and in the E_{13} case, all the values overlap, and they produce an unique point. This is due to the different magnitudes in the scales of both parameters, which covers the true dispersion effects making the parameters to set a good regression line, considering this graph seemingly acceptable.

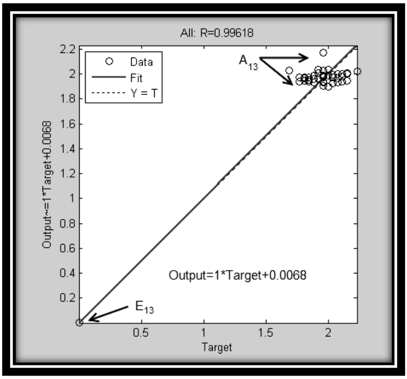


Figure 3. Linear regression of the matrix of *outputs vs targets*, when both parameters (E_{13}/A_{13}) are represented together during the ANN *training* process with an architecture of 3 *hidden* layers (25, 3, 2).

The *outputs/*targets regressions separately show the great dispersion that exists in both parameters (figures 4.a and 4.b), so that the results obtained with the application only of ANN are unacceptable, therefore it is necessary to use an alternative algorithm, in our case AGDC, to obtain the satisfactory A_{13} and E_{13} final values.

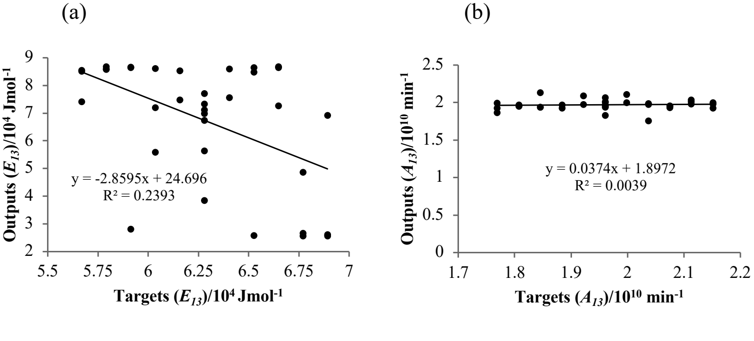


Figure 4.a and 4.b. Linear regression of the matrix of *outputs vs targets,* exclusively in the case of E_{13} (a) and A_{13} (b) for the ANN *training* process with an architecture of 3 *hidden* layers (25, 3, 2).

4.1.2.2 Optimization of E_{13} and A_{13} through AGDC

The values of E_{13} and A_{13} obtained in the *prediction* process after the application of ANN have been used as initial estimations in the optimization process that was carried out through the application of the program **KINNOISOT(AGDC)**. The second part of table 1 shows the values of the optimized parameters after the application of AGDC and the deviations of the real values. The application of the HA presumes an improvement of the results comparing the values of the deviations of both parameters with the ones calculated from the ANN application.

4.1.3 Determination of $\Delta H_{13}^{\neq}$ and $\Delta S_{13}^{\neq}$

4.1.3.1 *Training* of the ANN and *Prediction* of $\Delta H_{13}^{\neq}$ and $\Delta S_{13}^{\neq}$ through ANN

From the set of values $\Delta H_{13}^{\neq}$ and $\Delta S_{13}^{\neq}$ obtained from the corresponding ED (figure 2) (*targets* for *training*) we have generated the non-isothermal kinetic curves (equation 17) (*inputs* for *training*) and the process of ANN *training* has been executed with the objective of finding the best one to carry out the process of *prediction* of $\Delta H_{13}^{\neq}$ and $\Delta S_{13}^{\neq}$.

Table 2 shows the most relevant results obtained in the *training* process and the subsequent *prediction* of $\Delta H_{13}^{\neq}$ and ΔS_{13} (corresponding to the central point of the 45 curves of ED), obtained for different configurations for the architectures with 2 and 3 *hidden* layers. In the first part of table 2 we show the *outputs* values, $\Delta H_{13}^{\neq}/\Delta S_{13}^{\neq}$ and the appropriate deviations (*"Des. %"*), the optimal network in this case consists of 3 *hidden* layers and the configuration 12/10/10. The obtained values for the errors of the parameters indicate that it is necessary to improve the results through the application of the AGDC.

As in the previous case, the joint representation of the regressions *outputs/targets* of $\Delta H_{13}^{\neq}$ and $\Delta S_{13}^{\neq}$ (figure 5) are extremely separated within the graph since their magnitude orders are very different ($\Delta H_{13}^{\neq}\approx 10^4$ and $\Delta S_{13}^{\neq}\approx 10^1$) and because they are represented at the same scale, the points that appear in the graph form two groups separated by a great distance. Moreover, all the conforming values to $\Delta S_{13}^{\neq}$ overlap in the same point. This situation masks the real dispersion effects, which caused the regression line to be correct when both ATPs are represented.

The individual representation of the regression *outputs/targets* of $\Delta H_{13}^{\neq}$ and $\Delta S_{13}^{\neq}$ (figures 6.a and 6.b) show great dispersion in both parameters which causes the results obtained through the exclusive application of ANN to be unacceptable and the necessity to use a different algorithm, AGDC in our case.

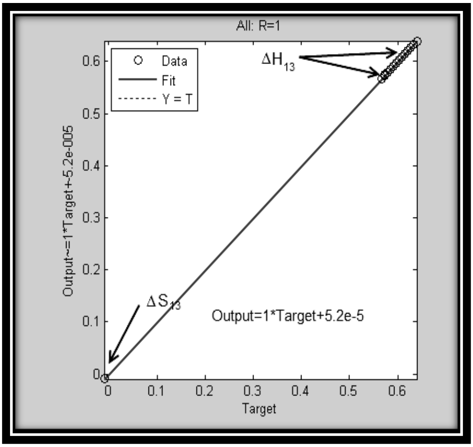
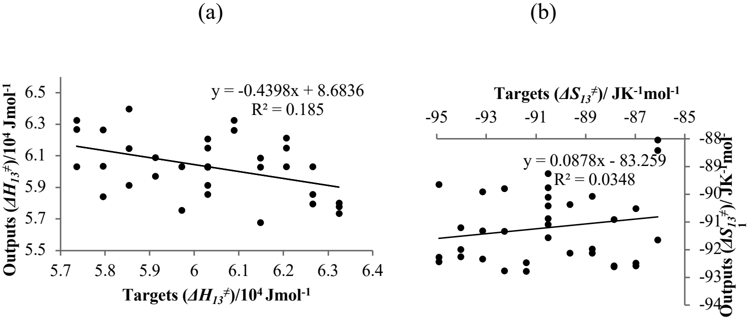


Figure 5. Linear regression of the matrix of *outputs vs targets*, when both parameters ($\Delta S_{13}^{\neq}/\Delta H_{13}^{\neq}$) are represented together during the ANN *training* process with an architecture of 3 *hidden* layers (25, 3, 2).



Figures 6.a and 6.b. Linear regression of the matrix of *outputs vs targets*, exclusively in the case of $\Delta H_{13}^{\neq}$ (a) and $\Delta S_{13}^{\neq}$ (b) the ANN t*raining* process with an architecture of 3 *hidden* layers (25, 3, 2).

Table 2. Results of the *training* and *prediction* processes of the neural network constituted by an architecture with 2 and 3 *hidden* layers and different configurations, when $\Delta S_{13}^{\neq}$ and $\Delta H_{13}^{\neq}$ are computed.

ANN					**(ANN**	**-**	**AGDC)**	**HA**
Config.	**$\Delta H_{13}^{\neq}/10^4$ Jmol^{-1}**	**$\Delta S_{13}^{\neq}/$ JK^{-1}mol^{-1}**	**Des. % ($\Delta H_{13}^{\neq}$)**	**Des. % ($\Delta S_{13}^{\neq}$)**	**$\Delta H_{13}^{\neq}/10^4$ Jmol^{-1}**	**$\Delta S_{13}^{\neq}/$ JK^{-1}mol^{-1}**	**Des. % ($\Delta H_{13}^{\neq}$)**	**Des. % ($\Delta S_{13}^{\neq}$)**
8/10	6.0308	-90.049	$-1.3808.10^{-2}$	$4.9839.10^{-1}$	6.0318	-90.443	$-2.9118.10^{-2}$	$6.2643.10^{-2}$
9/10	6.0371	-91.375	$-1.1871.10^{-1}$	$-9.6635.10^{-1}$	6.0318	-90.399	$-4.6118.10^{-2}$	$1.1219.10^{-1}$
10/8	6.0318	-94.818	$-2.9918.10^{-2}$	-4.7711	6.0392	-90.128	$-1.5242.10^{-1}$	$4.1118.10^{-1}$
10/9	6.0436	-87.700	$-2.2646.10^{-1}$	3.0939	6.0178	-90.907	$2.0278.10^{-1}$	$-4.4977.10^{-1}$
10/10	6.0322	-96.858	$-3.7422.10^{-2}$	-7.0258	5.9727	-92.583	$9.5007.10^{-1}$	-2.3011
10/11	6.0375	-90.570	$-1.2585.10^{-1}$	$-7.7870.10^{-2}$	6.0327	-90.409	$-4.5427.10^{-2}$	$1.0079.10^{-1}$
10/12	6.0270	-80.201	$4.8370.10^{-2}$	11.380	6.0283	-90.365	$2.8983.10^{-2}$	$1.4905.10^{-1}$
11/10	6.0310	-91.517	$-1.7310.10^{-2}$	-1.1239	6.0271	-90.592	$4.8071.10^{-2}$	$-1.0122.10^{-1}$
12/10	6.0036	-89.777	$4.3616.10^{-1}$	$7.9897.10^{-1}$	6.0275	-90.578	$4.0875.10^{-2}$	$-8.6178.10^{-2}$
8/10/10	6.0307	-91.991	$-1.2407.10^{-2}$	-1.6477	6.0291	-90.529	$1.5591.10^{-2}$	$-3.2229.10^{-2}$
9/10/10	6.0360	-94.019	$-1.0050.10^{-1}$	-3.8886	6.0161	-90.896	$2.3041.10^{-1}$	$-4.3756.10^{-1}$
10/8/10	6.0271	-91.831	$4.7171.10^{-2}$	-1.4708	6.0314	-90.451	$-2.2814.10^{-2}$	$5.4051.10^{-2}$
10/9/10	6.0362	-95.052	$-1.0362.10^{-1}$	-5.0296	5.9959	-91.521	$5.6519.10^{-1}$	-1.1282
10/10/8	6.0345	-86.522	$-7.5845.10^{-2}$	4.3954	6.0425	-90.095	2.0665.10-1	$4.5145.10^{-1}$
10/10/9	6.0412	-91.534	$-1.8722.10^{-1}$	-1.1423	6.0352	-90.312	8.6752.10-2	$2.0742.10^{-1}$
10/10/10	6.0336	-92.451	$-6.1136.10^{-2}$	-2.1556	6.0345	-90.333	$-7.4245.10^{-2}$	$1.8513.10^{-1}$
10/10/11	6.0411	-93.390	$-1.8491.10^{-1}$	-3.1934	6.0286	-90.495	$2.4086.10^{-2}$	$5.7495.10^{-3}$
10/10/12	6.0320	-89.701	$-3.4120.10^{-2}$	$8.8271.10^{-1}$	6.0335	-90.386	$-5.7935.10^{-2}$	$1.2636.10^{-1}$
10/11/10	5.9612	-96.321	1.1403	-6.4318	6.0440	-90.066	$-2.3260.10^{-1}$	$4.4926.10^{-1}$
10/12/10	6.0367	-91.272	$-1.1147.10^{-1}$	$-8.5312.10^{-1}$	6.0330	-90.391	$-5.0530.10^{-2}$	$1.2081.10^{-1}$
11/10/10	6.0204	-90.563	0.15834	$-6.9063.10^{-2}$	6.0245	-90.692	$9.4045.10^{-2}$	$-2.1232.10^{-1}$
12/10/10	**6.0294**	**-89.790**	**0.0097294**	**$7.8450.10^{-1}$**	**6.0308**	**-90.476**	**$1.3808.10^{-2}$**	**$2.6776.10^{-2}$**

4.1.3.2 Optimization of $\Delta S_{13}^{\neq}$ and $\Delta H_{13}^{\neq}$ through AGDC

In the second step, the **KINNOISOT(AGDC)** program is applied to optimize the values of $\Delta H_{13}^{\neq}$ and $\Delta S_{13}^{\neq}$ using as initial estimations the predicted values through ANN. The values of $\Delta H_{13}^{\neq}$ and $\Delta S_{13}^{\neq}$ optimized with the AGDC algorithm and the deviations of the real values are shown in the second part of table 2, the values of the deviations of $\Delta H_{13}^{\neq}$ and $\Delta S_{13}^{\neq}$ show abundant improvement.

Lastly, the HA is applied to determine the values of A_{13}, E_{13}, $\Delta H_{13}^{\neq}$ and $\Delta S_{13}^{\neq}$ corresponding to non-isothermal experimental kinetic experiences. First, the parameter values are determined using for each case the optimal neural network and then, the results obtained by means of the *prediction* with ANN, are submitted to a later refining process or improvement by the application of the AGDC algorithm. The optimization process of makes up the end of the HA that provides us with the unknown values of the parameters corresponding to the experimental kinetics performed in the laboratory. Table 3 shows the results obtained in the determination of A_{13}, E_{13}, $\Delta H_{13}^{\neq}$ and $\Delta S_{13}^{\neq}$: values obtained with the Arrhenius and Eyring equations (columns 1 and 2), values obtained after the application of HA (columns 3 and 4) and deviation of each parameter.

Table 3. Results of the application of the hybrid algorithm (ANN-AGDC) for the *prediction* of:
a) A_{13} and E_{13}, 2 experiences (architecture of the neural network (25, 3, 2) with 3 *hidden* layers and configuration 10/7/10).
b) $\Delta S_{13}^{\neq}$ and $\Delta H_{13}^{\neq}$, 7 experiences (architecture of the neural network (25, 3, 2) with 3 *hidden* layers and configuration 12/10/10).

		HA	**(ANN**	**-**	**AGDC)**
$(A_{13})_{\text{Arrhenius}}$/ 10^{10} min^{-1}	**$(E_{13})_{\text{Arrhenius}}$/ 10^{4}Jmol^{-1}**	**$(A_{13})_{\text{HA}}$/ 10^{10} min^{-1}**	**$(E_{13})_{\text{HA}}$/ 10^{4} Jmol^{-1}**	**% Des. (A_{13})**	**% Des. (E_{13})**
1.9589	6.2924	1.9590	6.2799	$5.2809.10^{-2}$	$2.0997.10^{-3}$
1.9252	6.0405	1.9554	6.0345	-1.5652	$9.4054.10^{-2}$
$(\Delta H_{13}^{\neq})_{Eyring}$/ 10^{4} Jmol^{-1}	**$(\Delta S_{13}^{\neq})_{Eyring}$ / JK^{-1}mol^{-1}**	**$(\Delta H_{13}^{\neq})_{\text{HA}}$/ 10^{4} Jmol^{-1}**	**$(\Delta S_{13}^{\neq})_{\text{HA}}$/ JK^{-1}mol^{-1}**	**% Des. ($\Delta H_{13}^{\neq}$)**	**% Des. ($\Delta S_{13}^{\neq}$)**
5.8539	-92.254	4.0214	-93.151	$1.2214.10^{-3}$	$-4.2151.10^{-3}$
6.2106	-92.616	4.1366	-91.473	$1.5856.10^{-1}$	$-3.5703.10^{-1}$
5.9122	-91.320	3.9553	-92.272	$2.2553.10^{-3}$	$-7.4272.10^{-3}$
6.1491	-92.334	3.9843	-92.254	$-5.4843.10^{-3}$	$1.1854.10^{-2}$
6.0852	-90.069	3.9403	-89.712	$4.6303.10^{-2}$	$-1.0512.10^{-1}$
6.0891	-91.342	3.8591	-91.370	$-6.9591.10^{-3}$	$1.4370.10^{-2}$
5.7530	-90.371	4.0373	-90.504	$1.9373.10^{-3}$	$-4.7504.10^{-3}$

4.2 Model II $\mathbf{B_1 \xrightarrow{k_{12}} B_2 \xrightarrow{k_{23}} B_3}$

The rupture reaction of the trinuclear chromium acetate cluster with a series of monoprotic and diprotic amino acids that act as a ligand in an aqueous medium [19-20], considering reaction conditions of pseudo-first order, can be assimilated to a mechanism formed by 2 consecutive steps, (*model II*), where B_1 is supposed to be an ionic pair, B_2 the intermediate specie and B_3 the product.

The reaction has been studied experimentally for pH values between 3.5-5.5, in a temperature range of 318.15-333.15K and an initial reactant concentration of $3.00.10^{-4}$ mol dm^{-3}. Its evolution has been followed spectrophotometrically and the values of the rate constants corresponding to the first and second steps of the reaction are of the order: $k_{12} \approx 10^{-4}$ s^{-1} and $k_{23} \approx 10^{-7}$ s^{-1} [19-20].

We have applied in this work the HA for the determination of the ATPs from the non-isothermal experimental kinetic data from this chemical reaction. The determination of the ATPs corresponding to the first step of the mechanism (E_{12}, A_{12}, $\Delta S_{12}^{\neq}$ and $\Delta H_{12}^{\neq}$) is performed as in the case of the previous mechanism (*model I*); thus, we will focus in the determination of the ATPs corresponding to the second step of the mechanism (E_{23}, A_{23}, $\Delta S_{23}^{\neq}$ and $\Delta H_{23}^{\neq}$).

4.2.1 Experimental design (ED)

The experimental designs used are of the *Central Star Composite Experimental Design* (CSCED) type and for its implementation we have taken experimental values obtained from bibliographic data [19-20]. In the case that the *factors* implicated are E_{23} and A_{23}, the *responses* are the non-isothermal kinetic data from the base curve of *inputs* (α_2/t). The values of the central point, acquired from the bibliography are: E_{23} = 12.6x10^4 Jmol^{-1} and A_{23}=1.81x10^{14} s^{-1}. If the *factors* are formed by $\Delta H_{23}^{\neq}/\Delta S_{23}^{\neq}$, the *responses* are the data from the base curve of *inputs* (α_2/t and α_3/t) and the values of the central point, acquired from the bibliography, are the following: $\Delta H_{23}^{\neq}$ = 123000 Jmol^{-1} and $\Delta S_{23}^{\neq}$ = 19 JK^{-1}mol^{-1}. The corresponding EDs are shown in figures 7 and 8.

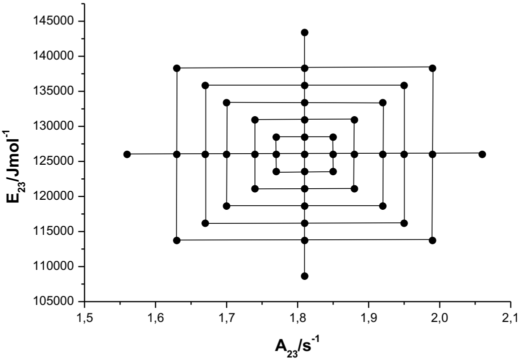


Figure 7. Experimental Design (CSCED) formed by 2 *factors* (A_{23}/E_{23}) and 45 points distributed in 9 levels (points 1-37) and 4 sub-levels (points 38-45) using as a base curve the matrix of *input* used for the network *training* process.

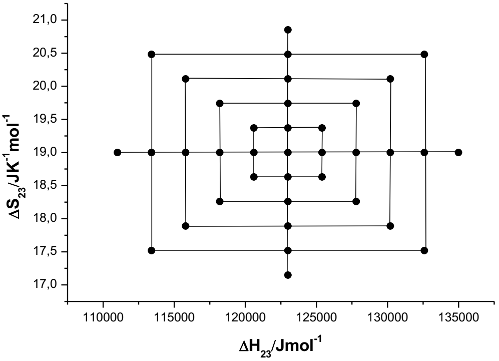


Figure 8. Experimental Design (CSCED) formed by 2 *factors* ($\Delta H_{23}^{\neq}/\Delta S_{23}^{\neq}$) and 37 points distributed in 9 levels and 3 sub-levels using as a base curve the matrix of *input* used for the network *training* process.

4.2.2 Determination of E_{23} and A_{23}

4.2.2.1. *Training* of the ANN and *prediction* of E_{23} and A_{23} through ANN

The *prediction* of E_{23} and A_{23} through ANN is carried out following the same steps as in the previous case (section 4.1.2.1):

- Generation of 45 non-isothermal kinetic curves (*inputs* for *training*) from the pairs of data E_{23}/A_{23} corresponding to ED (*targets* for *training*).
- *Training* of the artificial neural network with the objective of finding the most adequate network that allow us to determine E_{23} and A_{23}.
- *Prediction* with the optimal ANN.

The joint and separated representation of the *outputs/targets* regression of E_{23}/A_{23} show the same behavior that in the previous cases (sections 4.1.2.1 and 4.1.2.2) which marks the results as unacceptable through the unique application of ANN and it shows the necessity to use a different algorithm, in our case AGDC to complete the HA.

In the first part of table 4, the results obtained from the *training* process are shown and the subsequent *prediction* for the most meaningful configurations corresponding to the central point of the 45 curves of ED. In this case, the suitable neural network consists of 3 layers and one configuration 10/10/14 with a percentage of curves for the *training*, *validation* and *testing*, (80/10/10) (table 4). The pair of values of *outputs*, E_{23}/A_{23} and the corresponding deviations (*"Des. %"*) indicate that it is necessary to improve the result by applying the AGDC algorithm (second step of HA).

Table 4. Results of the *training* and *prediction* processes of the neural network constituted by an architecture with 2 and 3 *hidden* layers and different configurations, when A_{13} and E_{13} are computed.

ANN					**HA (ANN - AGDC)**			
Config.	**A_{23} /10^{14} s^{-1}**	**E_{23} /10^{5} Jmol^{-1}**	**Des. % (A_{23})**	**Des. % (E_{23})**	**A_{23} /10^{14} s^{-1}**	**E_{23} /10^{5} Jmol^{-1}**	**Des. % (A_{23})**	**Des. % (E_{23})**
7/10	1.8151	1.1930	4.9871.10^{-1}	5.3163	1.8131	1.1930	-2.7951.10^{-1}	5.3162
10/8	1.8142	1.3272	6.0382.10^{-1}	-5.3406	1.8142	1.3273	-2.3482.10^{-1}	-5.3401
10/10	1.8119	1.4265	7.4679.10^{-1}	-13.217	1.8128	1.4265	-1.0248.10^{-1}	-13.217
10/11	1.8137	1.1913	-1.0254	5.4514	1.8147	1.1913	-2.0487.10^{-1}	5.4514
10/12	1.8113	1.1657	4.9593.10^{-1}	7.4846	1.8113	1.1657	-7.0313.10^{-2}	7.4846
10/14	1.8088	1.3846	6.1918.10^{-1}	-9.8910	1.8098	1.3846	6.9498.10^{-2}	-9.8910
10/20	1.8115	1.2861	-5.7425.10^{-1}	-2.0719	1.8125	1.2861	-8.4025.10^{-2}	-2.0719
13/10	1.7950	1.1106	-1.3458	11.853	1.7950	1.1106	8.2860.10^{-1}	11.853
14/10	1.8085	1.1178	4.8265.10^{-1}	11.289	1.8085	1.1178	8.3585.10^{-2}	11.289
20/10	1.7933	1.3579	-9.8133.10^{-1}	-7.7730	1.7933	1.3579	9.2453.10^{-1}	-7.7730
10/10/14	**1.8138**	**1.2714**	**-2.1188.10^{-1}**	**-9.0834.10^{-1}**	**1.8148**	**1.2714**	**-2.1188.10^{-1}**	**-9.0834.10^{-1}**

4.2.2.2 Optimization of A_{23} and E_{23} through AGDC

Starting with the *outputs* values, E_{23}/A_{23} obtained after the application of ANN, they are optimized through the application of the program **KINNOISOT(AGDC)**. The second part of table 4 shows the values for the optimized parameters and the deviations from the real values. Comparing the deviation values of both parameters through the HA with the calculated from the application of ANN, the improvement of results is confirmed.

Lastly, the hybrid algorithm was applied to determine the values of A_{23} and E_{23} corresponding to a series of non-isothermal experimental kinetic experiences. Table 5 contains the results obtained after the application of the complete HA that gives us the unknown values of the parameters.

Table 5. Results of the application of the hybrid algorithm (ANN-AGDC) for the *prediction* of A_{13} and E_{13}, 8 experiences (architecture of the neural network (9.00.10^5, 3, 2) with 3 *hidden* layers and configuration 10/10/14).

		HA			
(A_{13})Arrhenius/ 10^{14} s^{-1}	**(E_{13}) Arrhenius/ 10^5Jmol^{-1}**	**(A_{23})HA/ 10^{14} min^{-1}**	**(E_{23})HA/ 10^5 Jmol^{-1}**	**% Des. (A_{23})**	**% Des. (E_{23})**
1.8199	1.5299	1.8199	1.5299	5.2112	-28.964
1.8173	1.3696	1.8173	1.3696	-6.5722	-2.6932
1.8033	1.4829	1.8033	1.4829	4.0785	-22.472
1.8040	1.3524	1.8040	1.3524	-3.6787	-3.3073
1.8142	1.4870	1.8142	1.4870	1.9364	-20.369
1.8049	1.3160	1.8049	1.3160	-1.9707	-2.4489
1.8040	1.3509	1.8040	1.3509	3.3280.10^{-1}	-3.1919
1.8138	1.2714	1.8138	1.2714	-2.1188.10^{-1}	-9.0834.10^{-1}

4.2.3 Determination of $\Delta S_{23}^{\neq}$ and $\Delta H_{23}^{\neq}$

4.2.3.1 *Training* of the ANN and *prediction* of of $\Delta S_{23}^{\neq}$ and $\Delta H_{23}^{\neq}$ through ANN

The process followed is analogous to the one previously followed:

- With the corresponding ED (*targets* for *training*), 37 kinetic curves (*inputs* for *training*) were generated considering a variation polynomic of the temperature.
- The process of *training* of ANN was done varying the number of nodes and the configuration of the systematically *hidden layers* with the goal of finding the most adequate network that allow us to determine $\Delta H_{23}^{\neq}$ and $\Delta S_{23}^{\neq}$. In this case the optimal architecture of the network is simpler, one *hidden* layer and the optimal number of nodes is 6 (table 6), since carrying out the process of *training* with more complex structures and a higher number of nodes, the value of *MSE* ($\approx 10^{-8}$) is not significantly modified.
- *Prediction* with the optimal ANN.

The joint representation of the *outputs/targets* regressions of $\Delta H_{23}^{\neq}$ and $\Delta S_{23}^{\neq}$ (figure 9), are highly separated because they present different magnitude orders ($\Delta H_{23}^{\neq} \approx 10^5$ and $\Delta S_{23}^{\neq} \approx 10^1$). In the case of the $\Delta H_{23}^{\neq}$ deviations, apparently exists a good linearity, which we check when we perform the individual regression (figure 10.a). In the case of the $\Delta S_{23}^{\neq}$ deviations, figure 9 shows the overlap of all the dispersion values giving an unique point as a consequence of the different magnitude of the scales of both parameters. Representing separately the *outputs/targets* regressions of $\Delta S_{23}^{\neq}$, we note a great dispersion for $\Delta S_{23}^{\neq}$ (figure 10.b) which makes the results to be unacceptable through the unique application of ANN, so that it is necessary to use another algorithm.

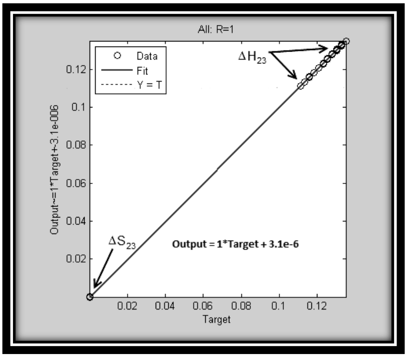
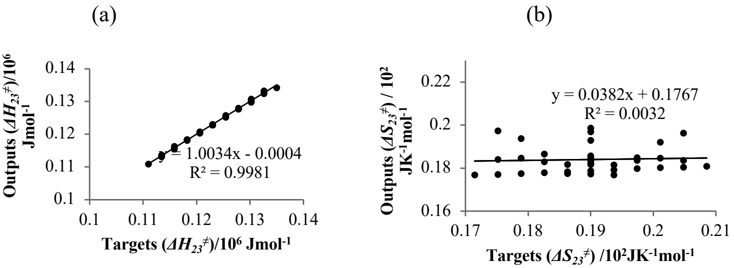


Figure 9. Linear regression of the *output vs targets* matrices, when both parameters ($\Delta S_{23}^{\neq}/\Delta H_{23}^{\neq}$) are jointly represented during the ANN *training* process with an architecture of 1 *hidden layer* (9.00.10^5, 1, 2).



Figures 10.a and 10.b. Linear regression of the *output vs targets* matrices, exclusively in the case of $\Delta H_{23}^{\neq}$ (a) and $\Delta S_{23}^{\neq}$ (b) during the ANN *training* process with an architecture of 1 *hidden layer* (9.00.10^5, 1, 2).

Table 6. Results of the *training* and *prediction* processes of the neural network constituted by an architecture with 1 *hidden* layer and different configurations, when $\Delta S_{13}^{\neq}$ and $\Delta H_{13}^{\neq}$ are computed.

ANN					**HA (ANN - AGDC)**			
Config.	**$\Delta H_{23}^{\neq}/10^5$ Jmol^{-1}**	**$\Delta S_{23}^{\neq}/$ JK^{-1}mol^{-1}**	**Des. % ($\Delta H_{23}^{\neq}$)**	**Des. % ($\Delta S_{23}^{\neq}$)**	**$\Delta H_{23}^{\neq}/10^5$ Jmol^{-1}**	**$\Delta S_{23}^{\neq}/$ JK^{-1}mol^{-1}**	**Des. % ($\Delta H_{23}^{\neq}$)**	**Des. % ($\Delta S_{23}^{\neq}$)**
5	1.2307	16.956	$-5.2924.10^{-2}$	10.760	1.2270	18.085	$2.4460.10^{-1}$	4.8174
6	**1.2304**	**19.781**	**$-3.0505.10^{-2}$**	**-4.1098**	**1.2347**	**19.449**	**$-1.1937.10^{-1}$**	**-2.3625**
7	1.2298	17.248	$1.6711.10^{-2}$	9.2234	1.2270	18.096	$2.4090.10^{-1}$	4.7587
8	1.2298	20.862	$1.6479.10^{-2}$	-9.7972	1.2330	19.905	$-2.4050.10^{-1}$	-4.7615
9	1.2301	15.864	$-4.0925.10^{-3}$	16.506	1.2250	17.452	$4.1270.10^{-1}$	8.1394
10	1.2299	15.491	$4.9065.10^{-3}$	18.471	1.2247	17.253	$4.6637.10^{-1}$	9.1970
12	1.2292	17.886	$6.7485.10^{-2}$	5.8637	1.2278	18.316	$1.8178.10^{-1}$	3.5994
15	1.2301	14.645	$-7.8094.10^{-3}$	22.921	1.2298	16.860	$5.7118.10^{-1}$	11.265
20	1.2306	13.510	$-4.6183.10^{-2}$	28.895	1.2215	16.387	$6.9365.10^{-1}$	13.755
22	1.2303	16.025	$-2.1074.10^{-2}$	15.659	1.2253	17.565	$3.8303.10^{-1}$	7.5515

The prediction process of $\Delta H_{23}^{\neq}$ and $\Delta S_{23}^{\neq}$ from experimental kinetic data coming from kinetic experiences in which its value is unknown, is carried out with the optimal ANN determined in the *training* process. Table 6 shows the results obtained for the most meaningful configurations corresponding to the central point of the 37 ED curves. The pair of values of *outputs* $\Delta H_{23}^{\neq}/\Delta S_{23}^{\neq}$ and the subsequent deviations (*"Des. %"*) show that it is necessary to improve the results through the application of the AGDC algorithm.

4.2.3.2 Optimization of $\Delta H_{23}^{\neq}$ and $\Delta S_{23}^{\neq}$ through AGDC

The second part of table 6 show the optimized values of $\Delta H_{23}^{\neq}/\Delta S_{23}^{\neq}$ after the application of AGDC and the deviations from the real values. In this case, we can see that after applying the AGDC algorithm, the deviation of $\Delta S_{23}^{\neq}$ considerably decreases, but the deviations of $\Delta H_{23}^{\neq}$ increases with respect to the results obtained with ANN. After seeing these results, we have applied the AGDC algorithm exclusively to improve the values of $\Delta S_{23}^{\neq}$ obtained by means of ANN. Table 7 shows the results obtained for this case, which let us conclude:

- The value of $\Delta H_{23}^{\neq}$ obtained by means of ANN possesses a small deviation with respect to the real value and can be considered acceptable.
- The application of the AGDC algorithm for the individual optimization of $\Delta S_{23}^{\neq}$ leads to a considerable decrease of the deviation with respect to the obtained results exclusively through ANN. The value of $\Delta S_{23}^{\neq}$ that ANN provides can be improved considerably when the AGDC algorithm is applied exclusively for the optimization of the parameter mentioned.

Table 7. Results of the *training* and *prediction* processes of the neural network constituted by an architecture with 1 *hidden* layer and different configurations, when $\Delta S_{13}^{\neq}$ is computed by HA.

ANN					**HA (ANN-AGDC)**	
Config.	**$\Delta H_{23}^{\neq}/10^5$ Jmol^{-1}**	**$\Delta S_{23}^{\neq}$/ JK^{-1}mol^{-1}**	**Des. % ($\Delta H_{23}^{\neq}$)**	**Des. % ($\Delta S_{23}^{\neq}$)**	**$\Delta S_{23}^{\neq}$/ JK^{-1}mol^{-1}**	**Des. % ($\Delta S_{23}^{\neq}$)**
5	1.2307	16.956	$-5.2924.10^{-2}$	10.7601	19.383	$-9.6541.10^{-1}$
6	1.2304	19.781	$-3.0505.10^{-2}$	-4.1098	19.099	$-5.1888.10^{-1}$
7	1.2298	17.248	$1.6711.10^{-2}$	9.2234	18.920	$4.2204.10^{-1}$
8	1.2298	20.862	$1.6479.10^{-2}$	-9.7972	18.921	$4.1732.10^{-1}$
9	**1.2301**	**15.864**	**$-4.0925.10^{-3}$**	**16.5058**	**18.999**	**$7.5058.10^{-3}$**
10	1.2299	15.491	$4.9065.10^{-3}$	18.4713	18.965	$1.8683.10^{-1}$
12	1.2292	17.886	$6.7485.10^{-2}$	5.8637	18.728	1.4337
15	1.2301	14.645	$-7.8094.10^{-3}$	22.9209	19.013	$-6.6509.10^{-2}$
20	1.2306	13.510	$-4.6183.10^{-2}$	28.8945	19.158	$-8.3115.10^{-1}$
22	1.2303	16.025	$-2.1074.10^{-2}$	15.6586	19.063	$-3.3096.10^{-1}$

Lastly, we have applied the hybrid algorithm to determine the values of $\Delta H_{23}^{\neq}$ and $\Delta S_{23}^{\neq}$ in a series of non-isothermal experimental kinetic experiences which values is unknown. In table 8, the results obtained after the application of HA are shown (columns 3 and 4). As in the previous case, we have exclusively optimized $\Delta S_{23}^{\neq}$ by means of the application of the AGDC algorithm. The results are shown in table 8 (column 4) and it allows us to check the decrease in the deviations in the majority of the cases if we compare it with the results obtained when ANN is exclusively applied.

Table 8. Results of the application of the hybrid algorithm (ANN-AGDC) for the *prediction* of $\Delta S_{23}^{\neq}$ and $\Delta H_{23}^{\neq}$, 8 experiences (architecture of the neural network ($9.00.10^5$, 3, 2) with 1 *hidden* layer and configuration 6).

		HA			
$(\Delta H_{23}^{\neq})_{Eyring}$/ 10^5 Jmol^{-1}	**$(\Delta S_{23}^{\neq})_{Eyring}$ / JK^{-1}mol^{-1}**	**$(\Delta H_{23}^{\neq})_{\mathrm{HA}}$/ 10^5 Jmol^{-1}**	**$(\Delta S_{23}^{\neq})_{\mathrm{HA}}$/ JK^{-1}mol^{-1}**	**% Des. ($\Delta H_{23}^{\neq}$)**	**% Des. ($\Delta S_{23}^{\neq}$)**
1.3074	18.881	1.3063	19.220	$-3.3413.10^{-1}$	-7.4453
1.1461	20.272	1.1524	18.321	$4.8714.10^{-1}$	8.9013
1.2779	19.134	1.2794	18.691	$-1.0954.10^{-1}$	-2.3659
1.2533	19.470	1.2550	18.942	$-8.0850.10^{-2}$	-1.6764
1.2055	20.151	1.2070	19.686	$-8.3970.10^{-2}$	-1.6272
1.1801	20.412	1.1882	19.783	$-9.4882.10^{-3}$	$-2.1373.10^{-1}$
1.2285	19.802	1.2294	19.542	$5.2294.10^{-2}$	1.0103
1.2304	19.781	1.2315	19.449	$-1.1935.10^{-1}$	-2.3625

5 Conclusions

- The application of the hybrid algorithm (ANN-AGDC) to non-isothermal kinetics allows to determine the activation thermodynamic parameters (ATPs), directly, without the need to calculate the kinetic constants in a previous step.

- The treatment of non-isothermal kinetic data has great advantages over the classic isothermal methods in which it is necessary to first calculate the value of the kinetic constant at each temperature and then carry out the linearization of the Arrhenius and/or Eyring equation to obtain the activation thermodynamic parameters. This implies making a mathematical transformation of the equations and the experimental data that produces a modification, both of their intrinsic errors and of the errors and precisions in the values of the activation parameters obtained. This problem does not exist when using the hybrid algorithm (ANN-AGDC) for the determination of activation thermodynamic parameters from non-isothermal kinetic data. The treatment of the non-isothermal kinetic data is more complex, but the HA can be perfectly applied since it is a rigorous and robust method, as shown by the obtained results.

- The application of ANN for the determination of activation thermodynamic parameters (pre-exponential factor (A), the activation energy (E), the activation enthalpy ($\Delta H^{\neq}$) and the activation entropy ($\Delta S^{\neq}$)) provide results that are not acceptable as shown by the graphical representations of the *output/target* regressions. The subsequent application of the AGDC optimization algorithm is necessary in a second step to improve the results obtained with ANN, the application of these two processes has caused the hybrid algorithm (ANN-AGDC).

- The great advantage of ANN is that it is a treatment that does not need initial estimates, so the hybrid algorithm initially applies ANN and the results obtained are used as values for the initial estimates in the second method (AGDC) which does require the use of previous values for the parameters to be determined.

- The application of AGDC serves to improve the values previously obtained in ANN to achieve the optimal values of the ATPs.

- The individual application of the ANN and AGDC methods provides unacceptable or insufficient results, however, the sequential application of both methods (hybrid algorithm) provides a set of optimal results. In the graphical representations of the *outputs/targets* regressions it is easily verified that the values of the residuals obtained with ANN are much higher than those obtained with the hybrid algorithm.

- In view of all the results obtained, we can conclude that the hybrid algorithm (ANN-AGDC) is a robust computational method for the treatment of non-isothermal kinetic data and for the determination of ATPs values in the treated systems. These good results allow us to propose it as a very useful and effective method for the treatment of experimental kinetic data acquired in the laboratory.

References

[1] A. A. Frost, R. G. Pearson, *Kinetics and Mechanism,* Wiley, New York, 1961.

[2] C. H. Bandford, C. F. H. Tipper, *Comprehensive Chemical Kinetics,* Elsevier, Amsterdam, 1973.

[3] M. A. Wolfe, *Numerical Methods for Unconstrained Optimization,* Van Nostrand, New York, 1978.

[4] J. Zupan, J. Gasteiger, *Neural Networks for Chemists,* VCH–Wiley, Weinheim, 1993.

[5] S. Curteanu, H. Cartwright, Neural networks applied in chemistry. I. Determination of the optimal topology of multilayer perceptron neural networks, *J. Chemom.* **25** (2011) 527–549.

[6] N. H. T. Lemes, E. Borges, J. P. Braga, Rate constants and absorption coefficients from experimental data: an inversion procedure based on recursive neural networks, *Chemom. Intell. Lab. Syst.* **96** (2009) 84–87.

[7] F. Amato, J. L. González-Hernández, J. Havel, Artificial neural networks combined with experimental design: A "soft" approach for chemical kinetics, *Talanta* **93** (2012) 72–78.

[8] M. M. Canedo, J. L. González-Hernández, S. Encinar del Dedo, Combining artificial neural networks and experimental design to prediction of kinetic rate constants, *J. Math. Chem.* **51** (2013) 1634–1653.

[9] M. M. Canedo, J. L. González-Hernández, A new computational application of the AGDC algorithm for kinetic resolution of multicomponent mixtures (static and dynamic), *Chemom. Intell. Lab. Syst.* **66** (2003) 63–78.

[10] M. M. Canedo, J. L. González-Hernández, ANALKIN (AGDC): a multipurpose computational program for the kinetic resolution of multicomponent mixtures (static and dynamic), *Chemom. Intell. Lab. Syst.* **66** (2003) 93–97.

[11] M. M. Canedo, J. L. González-Hernández, KINMODEL(AGDC): a multipurpose computational method for kinetic treatment, *J. Math. Chem*. **49** (2011) 163–184.

[12] M. M. Canedo, J. L. González-Hernández, S. Encinar del Dedo, Application de computational method KINMODEL(AGDC) to the simultaneous determination of kinetic and analytical parameters, *App. Math. and Comp.* **219** (2013) 7089–7101.

[13] J. L. González-Hernández, M. M. Canedo, C. Grande, Combining different mathematical optimization methods: A new "hard-modelling" approach for chemical kinetics, *MATCH Commun. Math. Comput. Chem.* **70** (2013) 951–970.

[14] S. Encinar del Dedo, J. L. González-Hernández, M. M. Canedo, A new computational treatment in non-isothermal chemical kinetics by application of a robust hybrid algorithm, *MATCH Commun. Math. Comput. Chem.* **72** (2014) 427–450.

[15] S. Encinar del Dedo, J. L. González-Hernández, M. M. Canedo, D. Juanes, A robust hybrid algorithm (Neural Networks-AGDC) applied to non-isothermal kinetics of consecutive chemical reactions, *J. Math. Chem.* **53** (2015) 1080–1104.

[16] J. L. González-Hernández, M. M. Canedo Alonso, S. Encinar del Dedo, Application of a robust hybrid algorithm (Neural Networks-AGDC) for the determination of kinetic parameters and discrimination among reaction mechanisms, *MATCH Commun. Math. Comput. Chem.* **79** (2018) 619–644.

[17] J. L. González, M. M. Canedo, C. Grande, Classic and multivariate modeling treatment of the kinetics and mechanism of isomerization of 5-cholesten-3-one catalyzed by sodium ethoxide, *Int. J. Chem. Kinet.* **38** (2006) 38–47.

[18] J. L. González-Hernández, M. M. Canedo, C. Grande, Computational methods for the treatment of kinetics and mechanisms of the oxidation of 4-cholesten-3-one by molecular oxygen by sodium ethoxide, *Int. J. Chem.* **17** (2007) 289–302.

[19] S. Chaudhary, J. D. Van Horn, Biphasic kinetics in the reaction between amino acids or glutathione and the chromium acetate cluster, $[Cr_3O(OAc)_6]^+$, *Mutation Res.* **610** (2006) 56–65.

[20] S. Chaudhary, J. D. Van Horn, Breakdown kinetics of the tri-chromium(III) oxo acetate cluster ($[Cr_3O(OAc)_6]^+$) with some ligands of biological interest, *J. Inorg. Biochem.* **101** (2007) 329–335.

[21] K. J. Laidler, A glossary of terms used in chemical kinetics, including reaction dynamics, *Pure Appl. Chem.* **68** (1996) 149–192.

[22] T. Kohonen, An introduction to neural computing, *Neural Networks* **1** (1988) 3–16.

[23] C. F. Gerald, P. O. Wheatley, *Applied Numerical Analysis*, Adinson–Wesley, Massachusetts, 1984.

[24] MATLAB & Simulink, © COPYRIGHT by MathWorks, 1984–2016.

[25] J. Topping, *Errors of Observation and Their Treatment,* Chapman & Hall, London, 1978.

[26] R. D. Cook, S. Weisberg, *Residuals and Influence in Regression,* Chapman & Hall, New York, 1982.

[27] M. A. Wolfe, *Numerical Methods for Unconstrained Optimization*, Van Nostrand, Berkshire, 1978.

[28] P. Gill, W. Murray, M.H. Wright, *Practical Optimization*, Academic Press Inc., London, 1981.