Evaluating the air-sea interactions and fluxes using an instance-based reasoning system *

Juan M. Corchado [a,**]
Jim Aiken [b]
Emilio S. Corchado [c]
Florentino. Fdez-riverola [d]
[a] *Departamento de Informática y Automática, Universidad de Salamanca, Plaza de la Merced s/n. 37008, Salamanca - Spain.*
E-mail: corchado@usal.es
[b] *Centre for Air-Sea Interactions and fluxes (CASIX), Plymouth Marine Laboratory, Prospect Place, Plymouth, PL1 3 DH, U.K.*
E-mail: ja@pml.ac.uk
[c] *Departamento de Ingeniería Civil, Universidad de Burgos. C/ Francisco de Victoria s/n. 09006, Burgos - Spain*
E-mail: escorchado@ubu.es
[d] *Departamento de Informática, Universidad de Vigo, Campus Universitario As Lagoas s/n. 32004, Ourense - Spain.*
E-mail: riverola@uvigo.es

The interaction of the atmosphere and the ocean has a profound effect on climate, while the uptake by the oceans of a major fraction of atmospheric CO_2 has a moderating influence. By improving accuracy in the quantification of the ocean's CO_2 budget, a more precise estimation can be made of the terrestrial fraction of global CO_2 budget and its subsequent effect on climate change. First steps have been taken towards this from an environmental and economic point of view, by using an instance based reasoning system, which incorporates a novel clustering and retrieval method - a Cooperative Maximum Likelihood Hebbian Learning model. This paper reviews the problems of measuring the ocean's CO_2 budget and presents the model developed to resolve them.

Keywords: CBR, Instance-based reasoning system, Hebbian learning and clustering, application

*This work has been supported by the CASIX project.
**Corresponding author: Juan M. Corchado, Departamento de Informática y Automática, Universidad de Salamanca, Plaza de la Merced s/n. 37008, Salamanca - Spain.

AI Communications
ISSN 0921-7126, IOS Press.

1. Introduction

An understanding of the natural sources and sinks of atmospheric carbon dioxide is necessary for predicting future atmospheric loading and its consequences for global climate. Present estimates of emissions and uptake do not balance, and although some have attributed the imbalance to a terrestrial sink, the magnitude of the oceanic sink remains undefined [1]. The rapid increase in atmospheric CO_2 resulting from atmospheric changes in the carbon cycle has stimulated a great deal of interest. A continuing major cause of uncertainty is the role played by photosynthesis in providing a sink for anthropogenic emissions [2]. The solution to these types of problems requires the use of dynamic systems, capable of incorporating new knowledge and facilitating the monitoring and estimation work carried out by oceanographers.

This paper presents the results obtained with an instance based reasoning system (IBR) developed to estimate the partial pressure of CO_2 (pCO_2) from information extracted from satellite pictures, wind direction and strength and other parameters such as water temperature, salinity and fluorescence. The final goal of our project is to construct a model that calculates the global budgets of CO_2, a mean CO_2 flux for the whole oceanographic basin.

Case based reasoning (CBR) and IBR systems have been successfully used in several domains such as diagnosis, prediction, control and planning [3,4,5]. Based on recent successful experiments with connectionist systems [6,7] an instance based reasoning system has been developed for estimating the partial pressure of CO_2 in the ocean. The IBR system developed incorporates a novel Cooperative Maximum Likelihood Hebbian Learning model for the data clustering and retrieval and a radial-bases function neural network for instance adaptation and forecast.

A typical CBR system is composed of four sequential steps which are recalled every time a prob-

lem needs to be solved [8]: Retrieve the most relevant case(s), reuse the case(s) to attempt to solve the problem, revise the proposed solution if necessary, and retain the new solution as a part of a new case.

Each of the steps of the CBR life cycle requires a model or method in order to perform its mission. The algorithms selected for the retrieval of cases should be able to search the case base and select from it the most similar problems, together with their solutions, to the new problem. Cases should therefore represent, accurately, problems and their solutions. Once one or more cases in the case base are identified as being very similar to the new problem, they are selected for the solution of this particular problem. These cases are reused using a predefined method in order to generate a proposed solution (i.e. normally using an adaptation technique). This solution is revised (if possible) and finally the new case (the problem together with the solution obtained) is stored. Cases can also be deleted if they prove to be inaccurate; they can be merged together to create more generalised ones and can be modified.

According to [8] there are five different types of CBR systems, and although they share similar features, each of them is more appropriate for a particular type of problem: exemplar based reasoning, instance based reasoning, memory-based reasoning, analogy-based reasoning and typical case-based reasoning. Instance-based reasoning systems focus on problems in which there are a large number of instances which are needed to represent the whole range of the domain and where there is a lack of general background knowledge [5,7].

This paper reviews a method that can be used for the automation of IBR systems especially developed for estimating the partial pressure of CO_2 in an area of the Pacific ocean, which corresponds to a water mass situated off the Chile coasts of "Mejellones" and "Antofagasta". The Cooperative Maximum Likelihood Hebbian Learning method is a novel approach that features both selection, in which the aim is to visualize and extract information from complex, and highly dynamic data. The model proposed is a mixture of factor analysis and exploratory projection pursuit [9] based on a family of cost functions proposed by [10] which maximizes the likelihood of identifying a specific distribution in the data while minimizing the effect of outliers [11,10]. This method is used for the clustering of instances, and during the retrieval stage of the IBR cycle, the adaptation step is carried out using a radial basis function network while the revision stage is manually carried out by an oceanographer (since the specific aim of this project is to construct a tool for oceanographers). Finally, the system is updated continuously with data obtained from the afore mentioned satellites and sensors.

First, the present paper will describe the oceanographic problem that defines the framework of our research, then the Cooperative Maximum Likelihood Hebbian Learning method, used to automate the retrieval stage of the IBR systems, will be described. A presentation will then be made of the instance based reasoning model and finally, the results of the experiments will be described.

2. Air sea interaction and Fluxes

The oceans contain approximately 50 times more CO_2 in dissolved forms than the atmosphere, while the land biosphere including the biota and soil carbon contains about 3 times as much carbon (in CO_2 form) as the atmosphere [12]. The CO_2 concentration in the atmosphere is governed primarily by the exchange of CO_2 with these two dynamic reservoirs. Since the beginning of the industrial era, about 2000 billion tons of carbon have been released into the atmosphere as CO_2 from various industrial sources including fossil fuel combustion and cement production. It is important, therefore, to fully understand the nature of the physical, chemical and biological processes which govern the oceanic sink/source conditions for atmospheric CO_2 [12,13].

New satellite sensors: ENVISAT, Aqua and other new Earth Observation satellites herald a new era in marine earth observation. Satellite-borne instruments provide high-precision, high-resolution data on atmosphere, ocean boundary layer properties and ocean biogeochemical variables, daily, globally, and in the long term. All these new sources of information have changed our approach to oceanography and the data generated needs to be fully exploited.

Wind stress, wave breaking and the damping of turbulence and ripples by surface slicks, all affect the air-sea exchange of CO_2. These processes are closely linked to the "roughness" of the sea surface, which can be measured by satellite radars and mi-

crowave radiometers. Sea surface roughness consists of a hierarchy of smaller waves upon larger waves (photograph, left, and close-up, below). Different sensors give subtly different measurements of this roughness.

Our final aim is to model this problem, and it is believed that by assimilating Earth Observation data into artificial intelligence models these problems may be solved. Earth Observation data (both for assimilation and for validation) are vital for the successful development of reliable models that can describe the complex physical and biogeochemical interactions involved in marine carbon cycling.

The proposed system has been tested in a number of cruises carried out off Chile during the austral summer of 2000. The oceanographic cruises had several purposes including the calibration of new satellites and sensors, evaluation of the model proposed, etc. During the cruise, data was obtained *in situ* from temperature, chlorophyll, fluorescence and salinity sensors, and satellite images were also obtained. Partial pressure of CO_2 (pCO_2) was also calculated in real time. This data was used to calibrate satellite sensors and to feed the IBR system, with the intention of developing a model that may allow, in the future, the calculation of pCO_2 values from satellite images rather than from *in situ* cruises.

3. Cooperative Maximum Likelihood Hebbian Learning Model

The Cooperative Maximum Likelihood Hebbian Learning method used during the retrieval stage of an IBR system is closely related to factor analysis and exploratory projection pursuit. It is a neural model based on the Negative Feedback artificial neural network [14], which has been extended by the combination of two different techniques. Firstly, a cost function from a family of cost functions which identify different distributions was selected. This method is called Maximum-Likelihood Hebbian learning [10,15]. Secondly, cooperative lateral connections derived from the Rectified Gaussian Distribution [16] were added to the Maximum-Likelihood method by [17] which enforced a greater sparsity in the weight vectors. The Negative Feedback artificial neural network has been linked to the statistical techniques of Principal Component Analysis [14], Factor Analysis [18] and Exploratory Projection Pursuit [14]. Lateral connections to the basic Maximum-likelihood network for the identification of different filters from video images have been previously introduced [17].

3.1. The Negative Feedback Neural Network

First, we shall present the Negative Feedback Network [14], which is the basis of the Maximum-Likelihood model. Feedback is said to exist in a system whenever the output of an element in the system partially influences the input applied to that particular element. It is used in this case to maintain the equilibrium on the weight vectors.

Consider an N-dimensional input vector, $\mathbf{x}$, and a M-dimensional output vector, $\mathbf{y}$, with W_{ij} being the weight linking input j to output i and let η be the learning rate.

The initial situation is that there is no activation at all in the network. The input data is fed forward via weights from the input neurons (the $\mathbf{x}$-values) to the output neurons (the $\mathbf{y}$-values) where a linear summation is performed to give the activation of the output neuron. We can express this as:

$$y_i = \sum_{j=1}^{\mathbf{N}} W_{ij} x_j, \forall i \tag{1}$$

The activation is fed back through the same weights and subtracted from the inputs (where the inhibition takes place):

$$e_j = x_j - \sum_{i=1}^{M} W_{ij} y_i, \forall j \tag{2}$$

After that simple Hebbian learning is performed between input and outputs:

$$\Delta W_{ij} = \eta e_j y_i \tag{3}$$

The effect of the negative feedback is to stabilise the learning in the network. Because of this, it is not necessary to normalise or clip the weights to get convergence to a stable solution.

Note that this algorithm is clearly equivalent to Oja's Subspace Algorithm [19] since if we substitute Equation (2) in Equation (3) we get:

$$\Delta W_{ij} = \eta e_j y_i = \eta \left(x_j - \sum_k W_{kj} y_k \right) y_i \tag{4}$$

This network is capable of finding the principal components of the input data [14] in a manner that is equivalent to Oja's Subspace algorithm [19], and so the weights will not find the actual Principal Components but a basis of the Subspace spanned by these components.

Since the model is equivalent to Oja's Subspace algorithm, we might legitimately ask what we gain by using the negative feedback in such a way. Writing the algorithm like this gives us a model of the process which allows us to envisage different models which would otherwise be impossible [14,10].

Factor Analysis is a technique similar to PCA in that it attempts to explain the data set in terms of a smaller number of underlying factors. However Factor Analysis begins with a specific model and then attempts to explain the data by finding parameters which best fit this model to the data. [20] has linked a constrained version of the Negative Feedback network to Factor Analysis. The constraint put on the network was a rectification of either the weights or the outputs (or both). Thus if the weight update resulted in negative weights, those weights were set to zero; if the feed forward mechanism gives a negative output, this was set to zero. We will use the notation $[t]^+$ for this rectification: if t $<$ 0, t is set to 0; if t $>$ 0, t is unchanged.

3.2. ε-Insensitive Hebbian Learning

It has been shown [21] that the nonlinear PCA rule

$$\Delta W_{ij} = \eta \left(x_j f\left(y_i\right) - f\left(y_i\right) \sum_k W_{kj} f\left(y_k\right) \right) \quad (5)$$

can be derived as an approximation to the best non-linear compression of the data. Thus we may start with a cost function

$$J\left(W\right) = 1^T E\left\{ \left(\mathbf{x} - W f\left(W^T \mathbf{x}\right) \right)^2 \right\} \quad (6)$$

which we minimise to get the rule(5). [22] used the residual in the linear version of (6) to define a cost function of the residual

$$J = f_1(\mathbf{e}) = f_1\left(\mathbf{x} - W\mathbf{y}\right) \quad (7)$$

where $f_1 = \|.\|^2$ is the (squared) Euclidean norm in the standard linear or nonlinear PCA rule. With this choice of $f_1\left(\right)$, the cost function is minimized with respect to any set of samples from the data set on the assumption that the residuals are chosen independently and identically distributed from a standard Gaussian distribution.

We may show that the minimization of J is equivalent to minimizing the negative log probability of the residual, e, if e is Gaussian. Let

$$p(\mathbf{e}) = \frac{1}{Z} \exp(-\mathbf{e}^2) \quad (8)$$

The factor Z normalizes the integral of $p\left(\mathbf{y}\right)$ to unity.

Then we can denote a general cost function associated with this network as

$$J = -\log p(\mathbf{e}) = (\mathbf{e})^2 + K \quad (9)$$

where K is a constant. Therefore performing gradient descent on J we have

$$\Delta W \propto -\frac{\partial J}{\partial W} = -\frac{\partial J}{\partial \mathbf{e}} \frac{\partial \mathbf{e}}{\partial W} \approx \mathbf{y}(2\mathbf{e})^T \quad (10)$$

where a less important term has been has been discarded. See [23] for details.

In general, the minimisation of such a cost function may be thought to make the probability of the residuals more dependent on the probability density function (pdf) of the residuals [11]. Thus if the probability density function of the residuals is known, this knowledge could be used to determine the optimal cost function. [22] investigated this with the (one dimensional) function:

$$p\left(\mathbf{e}\right) = \frac{1}{2+\varepsilon} \exp\left(-\left|\mathbf{e}\right|_\varepsilon\right) \quad (11)$$

where

$$\left|\mathbf{e}\right|_\varepsilon = \begin{cases} 0\_\forall \left|\mathbf{e}\right| < \varepsilon \\ \left|\mathbf{e}\right| - \varepsilon\_otherwise \end{cases} \quad (12)$$

with ε being a small scalar ≥ 0.

[10] described this in terms of noise in the data set. However we feel that it is more appropriate to state that, with this model of the pdf of the residual, the optimal $f_1\left(\right)$ function is the ε-insensitive cost function:

$$f_1\left(\mathbf{e}\right) = \left|\mathbf{e}\right|_\varepsilon \quad (13)$$

In the case of the Negative Feedback Network, the learning rule is

$$\Delta W \propto -\frac{\partial J}{\partial W} = -\frac{\partial f_1\left(\mathbf{e}\right)}{\partial \mathbf{e}} \frac{\partial \mathbf{e}}{\partial W} \quad (14)$$

which gives:

$$\Delta W_{ij} = \begin{cases} 0\_if\,|e_j| < \varepsilon \\ \eta y_i(sign\,(e_j))\text{otherwise} \end{cases} \quad (15)$$

The difference with the common Hebb learning rule is that the sign of the residual is used instead of the value of the residual. Because this learning rule is insensitive to the magnitude of the input vectors **x**, the rule is less sensitive to outliers than the usual rule based on mean squared error. This change from viewing the difference after feedback as simply a residual rather than an error will permit us later to consider a family of cost functions each member of which is optimal for a particular probability density function associated with the residual.

3.3. A Family of Learning Rules

Now the ε-insensitive learning rule is clearly only one of a possible family of learning rules which are suggested by the family of exponential distributions. Let the residual after feedback have probability density function

$$p(\mathbf{e}) = \frac{1}{Z}\exp(-|\mathbf{e}|^p) \quad (16)$$

Then we can denote a general cost function associated with this network as

$$J = -\log p(\mathbf{e}) = |\mathbf{e}|^p + K \quad (17)$$

where K is a constant. Therefore performing gradient descent on J we have

$$\begin{aligned} &\Delta W \propto -\tfrac{\partial J}{\partial W} = -\tfrac{\partial J}{\partial \mathbf{e}}\tfrac{\partial \mathbf{e}}{\partial W} \\ &\approx y(p|\mathbf{e}|^{p-1}sign(\mathbf{e}))^T \end{aligned} \quad (18)$$

where T denotes the transpose of a vector. We would expect that for leptokurtotic residuals (more kurtotic than a Gaussian distribution), values of p<2 would be appropriate, while for platykurtotic residuals (less kurtotic than a Gaussian), values of p>2 would be appropriate. It is a common belief in the ICA community [24] that it is less important to get exactly the correct distribution when searching for a specific source than it is to use a model with an approximately correct distribution i.e. all supergaussian signals can be retrieved using a generic leptokurtotic distribution and all subgaussian signals can be retrieved using a generic platykurtotic distribution. The experiments [17] tend to support this belief to some extent but we often find accuracy and speed of convergence are improved when we are accurate in our choice of p.

Therefore the network operation is:

Feedforward:

$$y_i = \sum_{j=1}^{N} W_{ij}x_j, \forall i \quad (19)$$

Feedback:

$$e_j = x_j - \sum_{i=1}^{M} W_{ij}y_i \quad (20)$$

Weight change:

$$\Delta W_{ij} = \eta.y_i.sign\,(e_j)\,|e_j|^{p-1} \quad (21)$$

[17] described their rule as performing a type of PCA, but this is not strictly true since only the original (Oja) ordinary Hebbian rule actually performs PCA. It might be more appropriate to link this family of learning rules to Principal Factor Analysis since this method makes an assumption about the noise in a data set and then removes the assumed noise from the covariance structure of the data before performing a PCA. We are doing something similar here in that we are basing our PCA-type rule on the assumed distribution of the residual. By maximising the likelihood of the residual with respect to the actual distribution, we are matching the learning rule to the pdf of the residual.

This method has been linked to the standard statistical method of Exploratory Projection Pursuit (EPP) [9]: EPP also gives a linear projection of a data set but chooses to project the data onto a set of basis vectors which best reveal the interesting structure in the data; interestingness is usually defined in terms of how far the distribution is from the Gaussian distribution.

3.4. Rectified Gaussian Distribution

The Rectified Gaussian Distribution [16] is a modification of the standard Gaussian distribution in which the variables are constrained to be non-negative, enabling the use of non-convex energy functions. The multivariate normal distribution can be defined in terms of an energy or cost function in that, if realised samples are taken far from the distribution's mean, they will be deemed

to have high energy and this will be equated to low probability. More formally, we may define the standard Gaussian distribution by:

$$p(\mathbf{y}) = Z^{-1} e^{-\beta E(\mathbf{y})}, \tag{22}$$

$$E(\mathbf{y}) = \frac{1}{2}\mathbf{y}^T \mathbf{A}\mathbf{y} - \mathbf{b}^T \mathbf{y} \tag{23}$$

The quadratic energy function $E(\mathbf{y})$ is defined by the vector $\mathbf{b}$ and the symmetric matrix $\mathbf{A}$. The parameter $\beta = {}^1\!/_T$ is an inverse temperature. Lowering the temperature concentrates the distribution at the minimum of the energy function.

One advantage of this formalisation is that it allows us to visualise regions of high or low probability in terms of energy and hence to view movement to low energy regions as movement to regions of high probability.

The quadratic energy function $E(\mathbf{y})$ can have different types of curvature depending on the matrix $\mathbf{A}$. Consider the situation in which the distribution of the firing of the outputs of our neural network follows a Rectified Gaussian Distribution. Then it is possible to identify values of $\mathbf{A}$ which give increasingly sparse firings and in the extreme, a single neuron will respond to the whole data set. Two examples of the Rectified Gaussian Distribution are the competitive and cooperative distributions. The modes of the competitive distribution are well-separated by regions of low probability. The modes of the cooperative distribution are closely spaced along a non-linear continuous manifold. Our experiments focus on a network based on the use of the cooperative distribution.

Neither distribution can be accurately approximated by a single standard Gaussian. Using the Rectified Gaussian, it is possible to represent both discrete and continuous variability in a way that a standard Gaussian cannot.

Not all energy functions can be used in the Rectified Gaussian Distribution. The sorts of energy function that can be used are only those where the matrix $\mathbf{A}$ has the property:

$$\mathbf{y^T A y} > \mathbf{0} \text{forall} \mathbf{y} : y_i > 0, i = 1...N \tag{24}$$

where N is the dimensionality of $\mathbf{y}$. This condition is called co-positivity. This property blocks the directions in which the energy diverges to negative infinity.

The cooperative distribution in the case of N variables is defined by:

$$A_{ij} = \delta_{ij} + \frac{1}{N} - \frac{4}{N}\cos\left(\frac{2\pi}{N}(i-j)\right) \tag{25}$$

$$b_i = 1 \tag{26}$$

where δ_{ij} is the Kronecker delta and i and j represent the identifiers of output neuron. To speed learning up, the matrix $\mathbf{A}$ can be simplified [3] to:

$$A_{ij} = (\delta_{ij} - \cos(2\pi(i-j)/N)) \tag{27}$$

The matrix $\mathbf{A}$ is used to modify the response to the data based on the relation between the distances between the outputs. The outputs are thought of as located on a ring ("wraparound").

Note that the modes of the Rectified Gaussian are the minima of the energy function, subject to non-negativity constraints. The modes of the distribution characterize much of its behaviour at low temperature. Finding the modes of a Rectified Gaussian is a problem in quadratic programming. However we will use what is probably the simplest algorithm, the projected gradient method, consisting of a gradient step followed by a rectification:

$$y_i(t+1) = [y_i(t) + \tau(b - \mathrm{A}y)]^+ \tag{28}$$

where the rectification $[]^+$ is necessary to ensure that the y-values keep to the positive quadrant. If the step size τ is chosen correctly, this algorithm can probably be shown to converge to a stationary point of the energy function [25]. In practice, this stationary point is generally a local minimum.

The mode of the distribution can be approached by gradient descent on the derivative of the energy function with respect to $\mathbf{y}$. This is:

$$\Delta\mathbf{y} \propto -\frac{\partial E}{\partial \mathbf{y}} = -(\mathrm{A}\mathbf{y} - \mathbf{b}) = \mathbf{b} - A\mathbf{y} \tag{29}$$

which is used as in Equation (28).

Now the rectification in Equation (28) is identical to the rectification which [17] used in the Maximum-Likelihood Network. Thus we will use this movement towards the mode in the Factor Analysis version of the Maximum-Likelihood Network before training the weights as previously. The net result will be shown to be a network which can find the independent factors of a data set but do so in a way which captures some type of global ordering in the data set.

We use the standard Maximum-Likelihood Network but now with a lateral connection (which acts

after the feed forward but before the feedback). Thus we have

Feedforward:

$$y_i = \sum_{j=1}^{\mathbf{N}} W_{ij} x_j, \forall i \tag{30}$$

Lateral Activation Passing:

$$y_i\,(t+1) = [y_i(t) + \tau\,(b - Ay)]^{+} \tag{31}$$

Feedback:

$$e_j = x_j - \sum_{i=1}^{M} W_{ij} y_i, \forall j \tag{32}$$

Weight change:

$$\Delta W_{ij} = \eta . y_i . sign\,(e_j)\,|e_j|^{p-1} \tag{33}$$

Where the parameter τ represents the strength of the lateral connections.

4. The Instance-case based reasoning system for estimating pCO_2

An artificial intelligent tool has been constructed for obtaining the value of the surface partial pressure of CO_2 in oceanographic waters from biological parameters and satellite information. The IBR system uses the Cooperative Maximum Likelihood Hebbian Learning Model for clustering the Instance-base and for the retrieval of the instances most similar to the "problem instance" due to its topology preserving properties [26]. The selected cases are used during the reuse stage to train a radial function neural network [27,10,7] that provides the value of the pCO_2 for a given point and the result is evaluated by an oceanographer. The learning (retain stage) is carried out by updating the instance base [7], updating the weights of the radial basis function network and by re-calling the Cooperative Maximum Likelihood Hebbian Learning Model for the clustering of the data. Figure 1 presents a schema of the problem solving model.

Applying equations (30) to (33) to the instance-base, the algorithm automatically groups the instances into clusters, grouping together those of similar structure. This technique is a classification and visualisation tool for high dimensional data on a low dimensional display. One of the advantages of this technique is that it is an unsupervised method so we do not need to have any information about the data beforehand. When a new instance is presented to the IBR system, it is identified as belonging to a particular type by applying Equation (30) to it. This mechanism may be used as a universal retrieval and indexing mechanism to be applied to any problem similar to the one presented here.

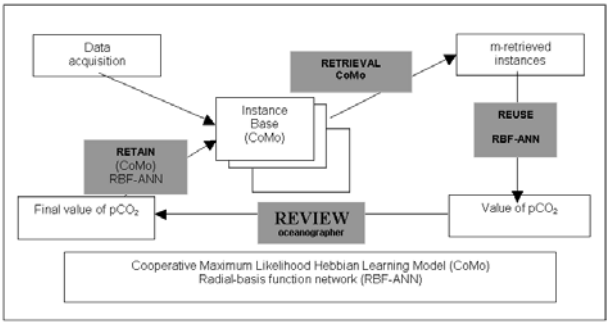

Fig. 1. IBR system model.

Each stored instance contains information relating to a specific situation.The problem case (instance) is described by teh following variable: Serial day of the year, Latitude, Longitude, Temperature, Salinity, Wind strength, Wind direction and fluorescence calibrated with chlorophyll which represent the problem description and surface partial pressure of CO_2 is the value that the IBR system has to identify from the problem descriptor. These values for a given point can be obtained from cruises using sensors or from satellite images. Two experiments were carried out with the purpose of testing and evaluating the efficiency of the systems.

4.1. Initial Experiment

The system was tested *in situ* during the cruise carried out in Pacific waters. The instance-base of the system was fed with 85% of the instances recorded during the cruise (over 100.000 instances). The other 15%, homogeneously spread along the cruise track, was left in order to test the system after the cruse was completed. Figure 2 shows the temperature (averaged) values along the cruise track. It presents a three dimensional view of the water temperature, where outside the track, the value that appears is the average value of the water temperature along the track. Figure

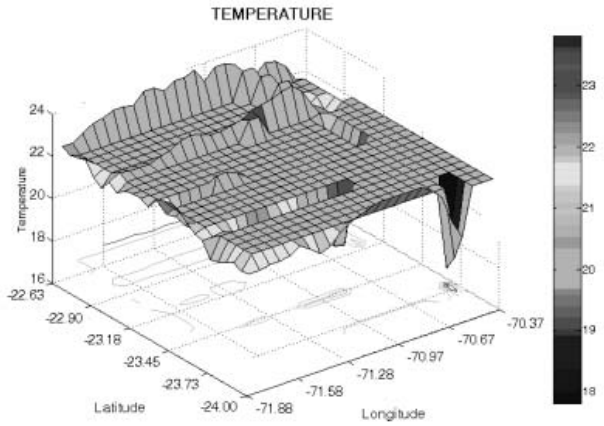

Fig. 2. Three dimensional view of the water temperature, where outside the track, the value that appears is the average value of the water temperature along the track.

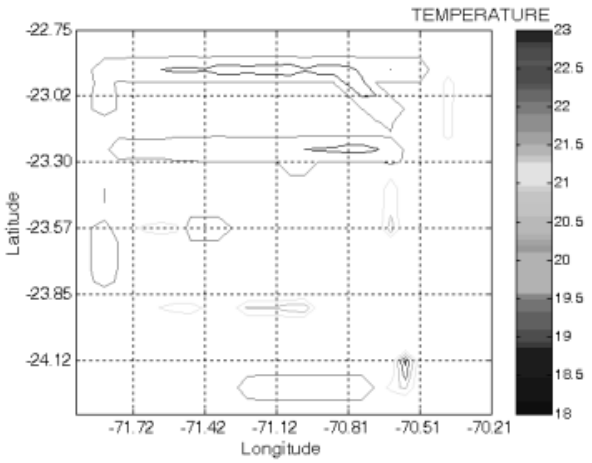

Fig. 3. Top view of the temperature values along the cruise track presented in Figure 2.

3 presents a top view of the temperature values along the cruise track showed in Figure 2.

The results obtained were very accurate, with an average error of 7,4%, which is less than the error provided by the other techniques we used to evaluate the IBR system. Table 1 presents the average error obtained with the Instance-based Reasoning System, a Radial-basis Function Neural Network, a Multi-layer Perceptron Neural Network, a Growing Cell Structures Neural Network and a K-nearest neighbour algorithm.

Starting from the error series generated by the different models, the Kruskall-Wallis test has been carried out. Since the P-value is less than 0,01,

Table 1
Average Error obtained with the IBR system and other methods

Method	Average Error
Instance-based System	7,4%
Radial-basis Function	9,8%
Multi-layer Perceptron	10,1%
Growing Cell Structures	16,2%
K-nearest neighbour	13,6%

there is a statistically significant difference between the models at the 99,0% confidence level. Figure 4 shows a multiple comparison procedure (Mann-Withney test) used to determine which models are significantly different from the others. The asterisk indicates that these pairs show statistically significant differences at the 99.0% confidence level. Figure 4, shows that the IBR system presents statistically significant differences from the other models. The proposed model generates the best results of all the tested techniques.

	Instance-based System	Radial-bases Function	Multi-layer Perceptron	Growing Cell Structures	K-nearest neighbor
Instance-based System					
Radial-bases Function	*				
Multi-layer Perceptron	*	=			
Growing Cell Structures	*	*	*		
K-nearest neighbor	*	*	=	*	

Fig. 4. Mann-Withney test results.

Figure 5 presents the error obtained in 40 cases in with the system was tested. These cases have been randomly obtained from the testing data set (15% of the whole data set), the other 85% of the data set was used to create the model.

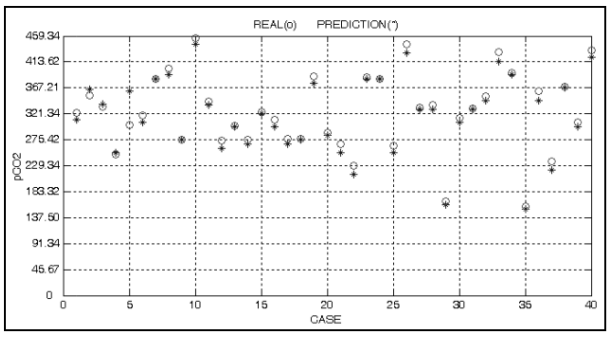

Fig. 5. Error obtained in 40 cases.

4.2. Second Experiment

Most of the values that represent the instance can be directly obtained from satellite photographs and others may be extracted with some simple calculation. For example, there exists a well known correlation between the water temperature and the salinity, and between salinity and Fluorescence [28], as can be seen in Figure 6.

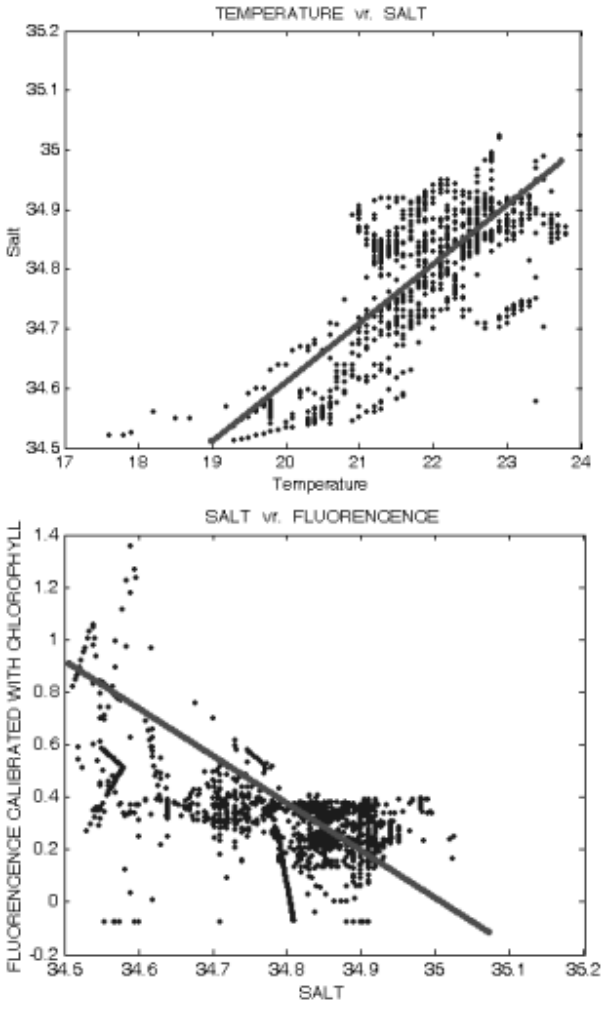

Fig. 6. (up) Salinity/Temperature relationship, (down) Fluorescence/Salinity relationship.

In this case, the IBR system was tested with data extracted from satellite images of the area in which the cruise took place. Problem instances were constructed, along the cruise track from such images and were fed into the IBR system, in order for it to obtain the value of the pCO_2. In this case the average error of the IBR system was slightly higher, but still very accurate compared with the results obtained with the other techniques. Oceanographers have also consider these results to be highly significant. The second column of Table 2 shows these results. The problem instances were obtained from the same photographs, but from points outside the cruise tracks, and similar results were obtained, as shown in the third column of Table 2.

Table 2
Average error obtained with the IBR system and other methods on Satellite data

Method	**Average Error** (Track data)	**Average Error** (Out side)
Instance-based System	9,7%	10,3%
Radial-basis Function	13,1%	14,5%
Multi-layer Perceptron	15,2%	14,7%
Growing Cell Structures	18,9%	18,8%
K-nearest neighbour	17,2%	18,1%

5. Conclusions and Future Work

This paper has presented a problem-solving method that combines a case-based reasoning system integrated with two artificial neural networks in order to create a real time autonomous problem solving system. The IBR system is able to produce a forecast with an acceptable degree of accuracy. The final constructed tool constitutes the first system developed for calculating the pCO_2 *in situ* and from satellite images. The IBR system incorporates a novel clustering technique capable of indexing huge instance-bases in an unsupervised way and of successfully retrieving instances with a similar structure, which is vital for constructing a model with a radial basis function neural network.

The Cooperative Maximum Likelihood Hebbian Learning Model has also performed better that other retrieval algorithms such as Kernel Methods [5], Growing Cell Structures [7], Maximum Likelihood Hebbian Learning [17] and Maximum Likelihood Hebbian Learning Based Retrieval Method [6], due to its fast convergence and clustering.

The results obtained in both experiments are very encouraging and the model presents great potential. The first experiment has allowed us to determine the efficiency of the model when the data

used to create the instance-base and the problem instances is reliable. The second experiment shows the potential of the model to automate the resolution of the problem with the help of satellite photographs. In this second experiment the error may be due to calibration imbalances, lack of definition of the photographs, presence of clouds, errors in the wind measures, etc.

More experiments need to be carried out for the model validation and techniques to facilitate the revision of the solution have to be obtained. The uncertainty and the dynamism of oceanographic systems have to be taken into consideration and techniques for monitoring such factors need to be incorporated into the system. The proposed model is a first step towards the resolution of this complex problem, which still requires a great deal more work and research.

References

[1] W.G. Finnegan, R.L. Pitter and L.G. Young, *Atmospheric Environment*, **A25** (1991), 2531–2534.

[2] J.L. Sarmiento and M. Dender, Carbon biogeochemistry and climate change, *Photosynthesis Research*, **39** (1994), 209–234.

[3] I. Watson and F. Marir F, Case-Based Reasoning: A Review, *The knowledge Engineering Review*, **9**(3), 1994.

[4] S.K. Pal, T.S. Dillon and D.S. Yeung, *Soft Computing in Case-based Reasoning*, (eds.), Springer Verlag, London, U.K., 2000.

[5] C. Fyfe and J.M. Corchado, Automating the construction of CBR Systems using Kernel Methods, *International Journal of Intelligent Systems*, **16**(4) (2001).

[6] J.M. Corchado, E.S. Corchado, J. Aiken, C. Fyfe, F. Fdez-Riverola and M. Glez-Bedia M, Maximum Likelihood Hebbian Learning Based Retrieval Method for CBR Systems, *5th International Conference on Case-Based Reasoning*, Springer, Trondheim, Norway, 2003.

[7] F. Fdez-Riverola F and J.M. Corchado, Forecasting Red Tides Using an Hybrid Neuro-Symbolic System, *AICOM* **16**(4) (2003), 221–233.

[8] A. Aamodt and E. Plaza E, Case-Based Reasoning: foundational Issues, Methodological Variations, and System Approaches, *AICOM* **7**(1) (1994), 39–59.

[9] J. Freedman and J. Tukey, A Projection Pursuit Algorithm for Exploratory Data Analysis, *IEEE Transaction on Computers*, **23** (1974), 881–890.

[10] C. Fyfe and E.S. Corchado, Maximum Likelihood Hebbian Rules, *European Symposium on Artificial Neural Networks*, 2002

[11] A.J. Smola and B. Scholkopf, A Tutorial on Support Vector Regression, *Technical Report NC2-TR-1998-030*, NeuroCOLT2 Technical Report Series, 1998.

[12] T. Takahashi, J. Olafsson, J.G. Goddard, D.W. Chipman and S.C. Sutherland, Seasonal Variation of CO_2 and nutrients in the High-latitude surface oceans: a comparative study, *Global biochemical Cycles*, **7**(4) (1993), 843–878.

[13] N. Lefevre, J. Aiken, J. Rutllant, G. Daneri, S. Lavender and T. Smyth, Observations of pCO_2 in the coastal upwelling off Chile: Sapatial and temporal extrapolation using satellite data, *Journal of Geophysical research*, **107**(0) (2002).

[14] C. Fyfe, *Negative Feedback as an Organising Principle for Artificial Neural Networks*, PhD Thesis, Strathclyde University, 1995.

[15] E. Corchado, D. MacDonald and C. Fyfe, Maximum and Minimum Likelihood Hebbian Learning for Exploratory Projection Pursuit, *Data mining and Knowledge Discovery*, Kluwer Academic Publishing, (*In press*), 2004.

[16] H.S. Seung, N.D. Socci and D. Lee, The Rectified Gaussian Distribution, *Advances in Neural Information Processing Systems*, **10** (1998).

[17] E. Corchado and C. Fyfe, Orientation Selection Using Maximum Likelihood Hebbian Learning, *International Journal of Knowledge-Based Intelligent Engineering Systems*, **7**(2) (2003).

[18] D. Charles, *Unsupervised Artificial Neural Networks for the Identification of Multiple Causes in Data*, PhD thesis, University of Paisley, 1999.

[19] E. Oja, Neural Networks, Principal Components and Subspaces, *International Journal of Neural Systems*, **1** (1989), 61-68.

[20] D. Charles and C. Fyfe, Modeling Multiple Cause Structure Using Rectification Constraints, *Network: Computation in Neural Systems*, **9** (1998), 167–182.

[21] L. Xu, Least Mean Square Error Reconstruction for Self-Organizing Nets, *Neural Networks*, **6** (1993), 627–648.

[22] C. Fyfe and D. MacDonald, e-Insensitive Hebbian learning, *Neuro Computing*, 2002.

[23] J. Karhunen and J. Joutsensalo, Representation and Separation of Signals Using Non-linear PCA Type Learning, *Neural Networks*, **7** (1994), 113–127.

[24] A. Hyvärinen, Complexity Pursuit: Separating interesting components from time series, *Neural Computation*, **13** (2001), 883–898.

[25] D.P. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, MA, 1995.

[26] D. MacDonald, E. Corchado and C. Fyfe, Analysing Spectroscopic Data Using Hierarchical Cooperative Maximum Likelihood Hebbian Learning, *Mexican International Conference on Artificial Intelligence*, Springer Verlag, 2004.

[27] J.M. Corchado and J Aiken, Hybrid Artificial Intelligence Methods in Oceanographic Forecasting Models, *IEEE SMC Transactions Part C*, **32**(4) (2002), 307–313.

[28] J.M. Corchado, J. Aiken and N. Rees N, *Artificial Intelligence Models for Oceanographic Forecasting*, Plymouth Marine Laboratory, U.K., (ISBN: 0-9519618-4-5), 2001.