Information Sciences 180 (2010) 2029–2043

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

A forecasting solution to the oil spill problem based on a hybrid intelligent system

Bruno Baruque [a], Emilio Corchado [b], Aitor Mata [b], Juan M. Corchado [b,*]

[a] Civil Engineering Department, University of Burgos, C/Francisco de Vitoria s/n, 09006 Burgos, Spain
[b] Departamento de Informática y Automática, University of Salamanca, Plaza de la Merced s/n, 37008 Salamanca, Spain

ARTICLE INFO

Article history:
Received 9 June 2009
Received in revised form 5 November 2009
Accepted 28 December 2009

Keywords:
Case-Based Reasoning
Oil spill
Self organizing memory
Radial Basis Function
Ensembles
Fusion algorithms

ABSTRACT

Oil spills represent one of the most destructive environmental disasters. Predicting the possibility of finding oil slicks in a certain area after an oil spill can be critical in reducing environmental risks. The system presented here uses the Case-Based Reasoning (CBR) methodology to forecast the presence or absence of oil slicks in certain open sea areas after an oil spill. CBR is a computational methodology designed to generate solutions to certain problems by analysing previous solutions given to previously solved problems. The proposed CBR system includes a novel network for data classification and retrieval. This type of network, which is constructed by using an algorithm to summarize the results of an ensemble of Self-Organizing Maps, is explained and analysed in the present study. The Weighted Voting Superposition (WeVoS) algorithm mainly aims to achieve the best topographically ordered representation of a dataset in the map. This study shows how the proposed system, called WeVoS-CBR, uses information such as salinity, temperature, pressure, number and area of the slicks, obtained from various satellites to accurately predict the presence of oil slicks in the north-west of the Galician coast, using historical data.

1. Introduction

When an oil spill occurs, the natural risks are evident. Complicated decisions must be made in order to keep the risk from actually becoming a natural disaster. The ability to predict if an area is going to be affected by the slicks generated after an oil spill will be highly useful in making those decisions.

The ocean is a highly variable environment where accurate predictions are difficult to achieve. The complexity of the modelling system increases if external elements are introduced into the analysis. In this case, oil spill data is added to the inherent complexity of the ocean, generating a rough set of elements. To model an environment similar to what is obtained after adding oceanic variables, weather conditions and oil spills, it is necessary to measure different parameters such as wind, current, and pressure. To predict the presence or absence of oil spills in a certain area, their previous positions must be known. That knowledge is provided by the analysis of satellite images, from which the position and size of the slicks are obtained.

The main objective of this interdisciplinary study is to present a new predicting hybrid intelligent model based on the Case-Based Reasoning methodology [58], which accomplishes the different phases of the CBR cycle by using different artificial intelligence techniques. A new algorithm called *WeVoS-SOM* (Weighted Voting Summarization of Self-Organizing Maps) [4] is introduced to organize the structure of the case base and support the recovery and retention of cases. This

[*] Corresponding author.
E-mail addresses: bbaruque@ubu.es (B. Baruque), escorchado@ubu.es (E. Corchado), aitor@usal.es (A. Mata), corchado@usal.es (J.M. Corchado).

0020-0255/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.ins.2009.12.032

novel algorithm improves the results obtained by previous methods [17]. The reuse phase is accomplished by applying the Growing RBF Network (GRBF) [30], a version of the classic Radial Basis Function neural network, which adapts itself to the data and grows as the case base grows. In the revision phase, explanations, which are an automatic way of justifying the solutions generated by the system, are used to internally justify the solution proposed. The *WeVoS-CBR* system presented here has been applied specifically to the data obtained after the Prestige accident in the north-west of Spain. However, the generalization capabilities of the CBR systems will permit the *WeVoS-CBR* system to be applied to other geographical areas and other natural phenomena by slightly adapting the system, which is something quite difficult to achieve with the existing approximations to this kind of problems.

The development of this system was possible thanks to a project founded by the Spanish Ministry of Science and Technology after the Prestige accident. Most of the data used to develop the proposed system was acquired from the ECCO (*Estimating the Circulation and Climate of the Ocean*) consortium [37]. Data related to the oil slicks, including their position and size, were obtained by studying SAR (*Synthetic Aperture Radar*) satellite images [39].

CBR systems have the ability to learn from past situations, and to generate solutions to new problems based on past solutions given to past problems. The system presented in this study combines the efficiency of the CBR systems with other artificial intelligence techniques in order to improve the results and to better generalize from past data. The generalization capabilities of the CBR systems permit this new system to extend the results obtained in the analyzed area to other open ocean areas. This is quite an innovative approach that improves the classic mathematical models, which are incapable of generating possibilities, by applying them to specific local regions. Several forecasting models have been applied to specific geographical zones, where the oceanic behaviour is quite unusual [42], but there does not exist a generic model, such as the one presented in this study, that can be applied in any open ocean region. Hybrid models can use both data and knowledge to forecast trajectories and evaluate possible risks after an oil spill [29].

The hybrid intelligent system proposed in this research incorporates a new ensemble summarization algorithm. The *WeVoS-SOM* algorithm [4] performs the classifications tasks in the CBR structure when creating the case base. This algorithm creates an inner structure within the case base that makes it easier to recover the cases by grouping similar cases together. When a new problem must be solved and the similar cases should be retrieved from the case base, it is very important to recover those cases quickly and accurately. An internal structure such as the one generated by the *WeVoS-SOM* model is crucial in this kind of system. When the similar cases are stored close one to another, then the recovery process does not need to search the entire case base, but only those elements close to the required one, which implies a great reduction of time.

The following section explains the oil spill problem, as well as some previous solutions and models used to try to solve that problem. The Section 3 provices a brief explanation of the CBR methodology detailing the different phases of the CBR cycle and some current applications of the CBR methodology. In Section 4, the *WeVoS* algorithm is developed, including its SOM foundation characteristics [31]. Finally, the hybrid system developed in this study is described, paying special attention to the different techniques used in the four main CBR phases, followed by the experimental results, conclusions and future work.

2. The oil spill problem and existing solutions

Once an oil spill occurs, the progression of the resulting oil slicks must be supervised or even predicted, in order to either determine if an area is going to be contaminated or, better yet, avoid contamination altogether in some critical areas. To get an accurate prediction, it is necessary to know how the oil slicks behave or, at the very least, what the probability is of finding oil slicks in an area. Increasing the number of variables involved in the analysis of the situation, however, also increases the difficulty of obtaining an accurate prediction.

First, the position, shape and size of the oil slicks must be identified. The most precise way to acquire that information is by using satellite images. SAR images are the most commonly used for automatically detecting these slicks [51]. The satellite images show certain areas where there seem to be no waves; but where there are, in fact, oil slicks. Fig. 1 shows an example of a SAR image with oil spills, while Fig. 2 shows the system's interpretation of the image shown in Fig. 1 properly locating the slicks and differentiating them from the coastal areas or even the islands present in the image. The interpretation of the images in conjunction with the variables taken into account (bottom pressure, salinity, wind, current,...) allows the WeVoS-CBR system to generate predictions about the future state of the oil slicks in a particular area. The images recognize the areas to analyze, and then, using the current parameters, a *case* is created. That case is used as a *problem* to be solved, by applying the WeVoS-CBR system.

With the SAR images, it is possible to distinguish between normal sea variability and slicks. It is also important to distinguish between oil slicks and look-alikes [55]. Oil slicks are quite similar to quiet sea areas. If there is not enough wind, the difference between the calm sea and the surface of a slick is less evident, which may result in more mistakes when trying to distinguish between an oil slick and something that is not a slick. This is a crucial aspect in this problem that can also be automatically addressed by a series of computational tools, whereby other meteorological factors are analyzed at the same time as the images, in order to completely distinguish between oil slicks and calm open ocean areas [39].

Once the slicks are identified, it is also essential to know the atmospheric and maritime situation that is affecting the slick at the moment that it is being analysed. Information collected from the satellites is used to obtain the necessary atmospheric

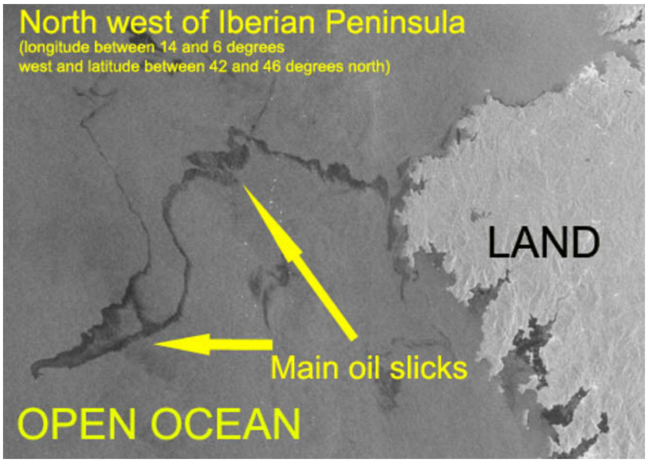

North west of Iberian Peninsula
(longitude between 14 and 6 degrees
west and latitude between 42 and 46 degrees north)
LAND
Main oil slicks
OPEN OCEAN

Fig. 1. A SAR image corresponding to the north-west coast of Spain.

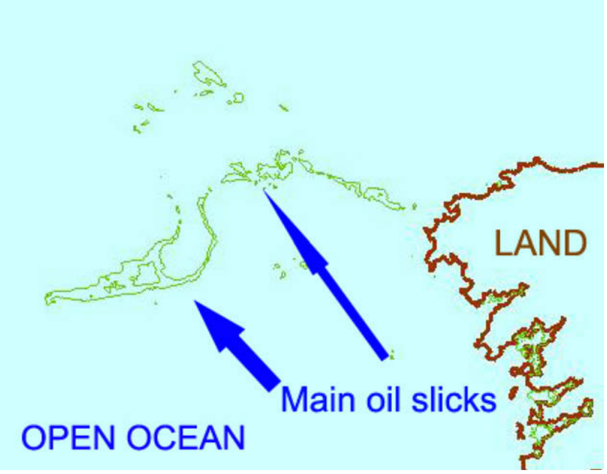


Fig. 2. Interpretation of a SAR image generated by the presented system.

data for such variables as temperature, sea height and salinity. These data are then measured and used to obtain a global model [53] that can explain how slicks evolve.

2.1. Former approximations to the oil spill problem

There have been different approaches for responding to an oil spill. If there is plenty of information about a specific area, a simulation can be done. A model is created [10] by introducing different parameters related to the weather, the currents and the wind. These models usually work in conjunction with a weather forecasting system. These kinds of solutions work well in small and specific areas [21], but it is quite difficult to generalize and apply the same solution to new zones. This is the biggest problem of this kind of system: it is possible to create a model for a specific and problematic area [41], which is a great help; but the models are quite limited, since it is not possible to apply the same solution to different geographical areas. Current data must be considered in order to create contingency plans that, although not normally applied to predicting tasks, could help minimize environmental risks [13] by taking action before the real risk can materialize.

Another previously used approach has been to situate drifters in places where a spill may be produced, or has already been produced in the past [47]. By supervising the way the drifters behave in the ocean, it is possible to make a comparison against real spill data. If the drifters follow a trajectory similar to that of the slicks, then a model can be created and the methodology can be used to create models for different areas. Wind driven drifts can also be combined with climate variables to produce a robust model for forecasting [11]. There are other trajectory models, such as the one created to comply with the NOAA (National Oceanic and Atmospheric Administration) standards [6], where two solutions are proposed: the 'best guess' and the 'minimum regret'.

2.2. Existing models

The previously explained systems can be improved by combining different elements and generating response models to solve the oil spill problem. One approach [20] is to analyze large data bases of various related knowledge fields (environmental, ecological, geographical and engineering), using expert systems. Using this combination of data bases, an implicit relationship between problem and solution is obtained. The relationship achieved shows no direct affiliation between past examples and current decisions, but new hidden correlations are discovered. Nevertheless this kind of solution involves a great data mining effort, which may imply high reaction time in quick changing conditions. Monitoring the spills [7] also gives a good quantity and quality of information, but serves only as a source of information that can be useful for other more complete systems where that information can be used to generate real solutions.

As mentioned before, it is critical to have contingency models that can generate a good solution as quickly as possible [48] when an oil spill is produced. The disadvantage of those models is that they are usually restricted to a specific geographical area where they work well, making any type of generalization nearly impossible, as the results are highly dependent on the studied area. Expert systems have also been used to obtain good solutions to the oil spill problem. Those systems use a repository of information stored from past situations which is in turn used by future applications to obtain structured information. Other complete models have been developed by integrating different variables related to the spills [43]. These models always try to get the best benefits by generating the solution that implies the lowest cost to the system. But these kinds of systems normally imply the creation of specific models for each specific behaviour of the considered elements. If those models were applied to a slightly different problem (i.e. changing oil by any other kind of contamination) the system would need to be completely redone. Hybrid models can forecast trajectories and evaluate possible risks after an oil spill [29] by combining modelling techniques that can help to predict the behaviour of the ocean and the spills in a certain area, and GIS systems (Geographic Information Systems) that may help to visualize the evolution of the spill and the possible solutions to be applied.

The objective of these complex systems is to become a decision support system that can help make all the decisions involved in the response system in an organized way. To accomplish that vast objective, different approaches have been used, from fuzzy logic [33], where the textual information is used but the solutions proposed are not always justified enough to solve the problem by themselves, to negotiation with multi-agent systems [26,34], where a set of proposals are translated to the users who can then choose among a smaller set of possibilities instead of having to consider the whole set of possible solutions.

In the following section, a short explanation of the Case-Based Reasoning methodology is presented, in order to introduce the structure of the system described in this study. That structure will be explained in depth in Section 5.

3. Brief introduction to Case-Based Reasoning systems

The origins of *Case-Based Reasoning* are found in knowledge based systems. CBR systems solve new problems by acquiring the knowledge needed from previous situations [1]. The principal element of a CBR system is the case base, a structure that stores the information used to generate new solutions. In the case base, data is organized into cases in which problems and their solutions are related. A case base can then be seen as a kind of database where a series of problems are stored, including a corresponding solution for each problem and relationship between them. In fact, keeping track of the relationship between past problems and their solutions is what gives CBR systems the ability to generalize and solve future problems.

The learning capabilities of the CBR system are rooted in its own structure, which is composed of four main phases [2]: *retrieval, reuse, revision* and *retention*. These phases are graphically represented in Fig. 3. The first phase is called *retrieve*, and consists of finding the most similar cases to the proposed problem from the case base. Once a series of cases are extracted from the case base, they must be *reused* by the system. In this second phase, the selected cases are adapted to fit the current problem. After giving a solution to the problem, that solution is *revised* to ensure that it is in fact an appropriate solution to the problem. If the proposal is confirmed as a solution, then it is *retained* by the system and could eventually serve as a solution to future problems.

Because it is a methodology [58], Case-Based Reasoning has been used to solve a great variety of problems. It is a cognitive structure that can be easily applied to solve problems such as those related to soft computing, since the procedures used by CBR are quite easy to assimilate by the approaches inherently used by soft computing. CBR has also helped to create applications for quite a variety of environments, such as health sciences [16,38], where images can often play an important role [8,26], or eLearning [3,18]. As it has evolved, CBR has been used to solve new problems, applied as a methodology to create plans, and broken down into a distributed version [44]. Oceanographic problems [22] have also been addressed using these techniques in order to predict the value of highly inconsistent parameters.

The use of past knowledge to generate new solutions allows CBR systems to be very useful as decision support systems. Distributed and multi-agent [12] systems have added the decision support capabilities of the CBR methodology to their characteristics. CBR methodologies have also been successfully applied to a variety of different knowledge fields, and combined with a vast array of techniques. Most of the techniques used within CBR systems serve to classify, adapt, revise solutions, etc. Artificial neural networks such as ART-Kohonen neural networks and fuzzy logic have also been used to complement the capabilities of the CBR methodology [28]. Similarity measures have been used to retrieve cases from the case base; these

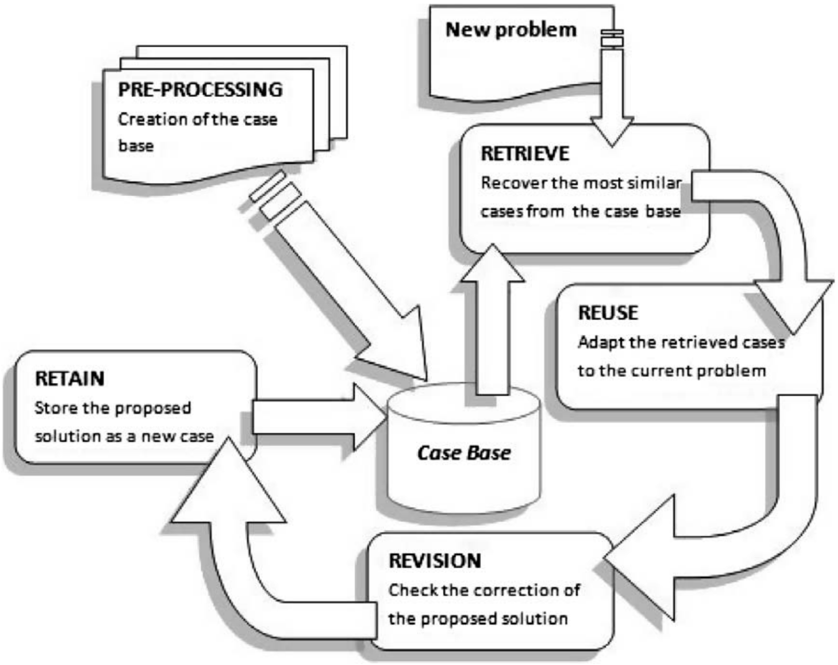

New problem
PRE-PROCESSING
Creation of the case base
RETRIEVE
Recover the most similar cases from the case base
REUSE
Adapt the retrieved cases to the current problem
RETAIN
Store the proposed solution as a new case
Case Base
REVISION
Check the correction of the proposed solution

Fig. 3. Basic structure of a CBR system.

include the *k-NN* (*k* nearest neighbours), and modern variations like Significant Nearest Neighbour [56] where the value of *k*, which is the number of neighbours to consider, is calculated by taking into account the dissimilarity between the new case and the past ones stored in the case base. Numeric situations, like those used in microarray problems, can be reused through neural networks like Growing Cell Structures [19], where the aim is to cluster the retrieved information. Another way to use neural networks to adapt the retrieved information is to change the weight of the connection between the neurons according to the retrieved cases [61]. Changing the weights allows the system to adapt the solution to the problem, as the retrieved cases will depend directly on the proposed problem. If the case base structure is integrated into a neural network, then the revision phase consists of changing the organization of the case base according to the correction of the proposed result and other neural variables such as neuron age, activation value and last use [59]. Genetic algorithms are also used to revise the correction of the solutions [40]. After running those algorithms, the solutions can be accepted and added to the case base.

Current trends in CBR explore the possibility of providing explanations from the actual CBR systems [52]. These techniques allow the CBR systems to give the users a better solution in the form of an explanation by adding additional information to the solution proposed by the system. With the explanations generated by the system, the solutions proposed by the system are justified and may be better understood.

Applying CBR to solve a problem generally implies using other artificial intelligence techniques. This process is not only a simple way of structuring the reutilization of the information, but a model that can combine different techniques to improve their individual results. CBR has been used in combination with artificial intelligence techniques to boost the power of the core methodology [32]. Different kinds of neural networks, from ART-Kohonen neural networks [60] to Growing Cell Structures [24], have been used with CBR to automatically create the internal structure of the case base [12]. Even fuzzy logic [23] has been used to complement the capabilities of the CBR methodology. In the present case, a novel algorithm will be used to structure the case base and to easily and accurately recover the most similar cases from the case base. That algorithm is the *WeVoS-SOM* model, which will be explained in Section 4, while Section 5 will discuss the implementation of the four main phases of the CBR cycle by using different techniques.

4. WeVoS-SOM: Weighted Voting Summarization of SOM ensembles

Case-Based Reasoning is a methodology that depends on past stored data from which knowledge is extracted in order to solve new problems. It is thus critical to properly organize the case base; that is, the structure where the information is kept [54]. The bigger the case base is, the better are the results that can be generated by CBR systems. On the other hand, increasing the size of the case base also implies a more difficult recovery of similar cases and maintenance of the stored data. So, for CBR systems it is crucial to make use of efficient algorithms to organize the case base and to recover similar cases from it. The *WeVoS-SOM* model presented here helps both to organize the case base, by means of ensembles of SOM networks [31], and to quickly recover the most similar cases from the case base, by using the intrinsic feature of the SOM networks that maintains the most similar elements close together.

The SOM is based on a type of unsupervised learning called competitive learning; an adaptive process in which the neurons in a neural network gradually become sensitive to different input categories (sets of samples in a specific domain of the input space). The main feature of the SOM algorithm is that the neighbours on the lattice are also allowed to learn, i.e., to adapt their characteristics to the input, as well as the winning neuron. Thus, the neighbouring neurons gradually come to represent similar inputs, and their representations become ordered on the map lattice.

This updating of neighbourhood neurons in SOM can be expressed as in Eq. (1):

$$w_k(t+1) = w_k(t) + \alpha(t)\eta(v,k,t)(x(t) - w_k(t)), \tag{1}$$

where w_k is the weight vector associated with neuron k; $\alpha(t)$ is the learning rate of the algorithm; $\eta(v,k,t)$ is the neighbourhood function (usually, the Gaussian function or a difference of Gaussians), in which v represents the position of the winning neuron in the lattice, or the Best Matching Unit (BMU) and k the positions of neighbouring neurons; finally, x is the network input.

The *WeVoS-SOM* model is based on the concept of the ensemble meta-algorithms, in which a committee of 'experts' work together to solve the same problem by improving the results that would yield a single solution [27,57]. The aim of this approach is to adapt the ensemble technique to work in combination with topology preserving algorithms. An ensemble of maps is trained on a dataset and a final map summarizing the main features detected by each one is calculated as a final result.

The *WeVoS* fusion algorithm presented in this study aims to obtain the final map by using the information contained in the maps comprising the ensemble on a unit-by-unit basis. Usually, the final characteristics vectors of a single map are calculated from a single training over the dataset. The *WeVoS* algorithm tries to generate the final characteristics vector for each unit by relying on an informed decision about the adaptation of its homologous units from an ensemble of maps, each of which has been trained on slightly different parts of the dataset [9]. This vector is also recalculated for the neighbours of the unit.

As a result, the final map obtained not only determines the best characteristics vector for each unit based on an informed decision, but also maintains one of the most important features of this type of algorithms: its topological ordering. *WeVoS* is an improved version of an algorithm presented in several previous works: superposition [5]. Although it has been successfully applied to the analysis of real-life data [4], in this study it is applied for the first time as part of a hybrid intelligence system to solve this kind of practical problem.

The first step in this meta-algorithm is to calculate the "quality" of each of the units comprising each map, in order to rely on some kind of informed decision for the fusion of units. This "quality" measure (or error measure) could be any one of the many "quality of map" measures presented in scientific literature regarding Self-Organizing Maps [45,46]; provided that it may be calculated on a unit-by-unit basis.

Then, the final map is obtained again on a unit-by-unit basis. Firstly, the units of the final map are initialized by calculating the centroids of the units in the same position of the map grid in each of the trained maps. Then, the final position of that unit is recalculated using the information associated with the units in that same position in each of the ensemble maps. For each unit, a voting process is performed as shown in Eq. (2):

$$V_{p,m} = \frac{\sum b_{p,m}}{\sum_{i=1}^{M} b_{p,i}} \cdot \frac{q_{p,m}}{\sum_{i=1}^{M} q_{p,i}}, \tag{2}$$

where $V_{p,m}$ is the weight of the vote for the unit included in map m of the ensemble, in its position p; M is the total number of maps in the ensemble; $b_{p,m}$ is the binary vector used for marking the dataset entries recognized by the unit in position p of map m; and, $q_{p,m}$ is the value of the desired quality measure for the unit in position p of map m.

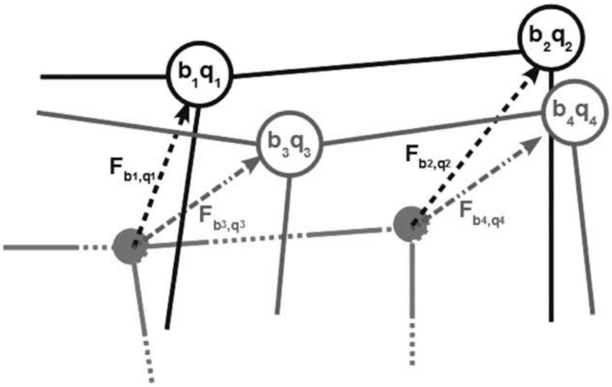


Fig. 4. Schematic diagram representing how the neurons of the final WeVoS-SOM map adapt their weights according to the quality and data recognition of the homologous neurons in each of the maps composing the ensemble.

The weights of each unit are fed into the final network in the same way as the data inputs during the training phase of a SOM, considering the 'homologous' unit in the final map as the BMU. The weights of the final unit will be updated towards the weights of the composing unit. The difference in the updating performed for each homologous unit in the composing maps depends on the quality measure calculated for each unit: the higher the quality (or the lower the error) of the unit in the composing map, the stronger the updating of the unit in the summary map towards the weights of that particular unit. With respect to quality determination, a single quality measure or a linear combination of several measures may be used. The number of data inputs recognized by each unit is also taken into account in the quantization of the 'most suitable' unit among those competing for the same position in the final map. In short, as it can be seen in Fig. 4, the summarization algorithm considers the most suitable weights of a composing unit to be the weights of the unit in the final map, according to both the number of inputs recognized and the adaptation quality of the unit. The fusion algorithm, referred to as *WeVoS*, is described in detail in Algorithm 1.

Algorithm 1. Weighted Voting Superposition (WeVoS).

```
1:  train several networks by using the bagging (re-sampling with replacement) meta-algorithm
2:  for each map ($m$) in the ensemble
3:    for each unit position ($p$) of the map
4:      calculate the quality measure/error chosen for the current unit
5:    end
6:  end
7:  calculate an accumulated total of the quality/error for each position $Q(p)$ on all maps
8:  calculate an accumulated total of the number of data entries recognized by a position $D(p)$ on all maps
9:  for each unit position ($p$)
10:     initialize the fused map (fus) by calculating the centroid ($w'$) of the units of all maps in that position ($p$)
11: end
12: for each map ($m$) in the ensemble
13:   for each unit position ($p$) of the map
14:     calculate the vote weight of the ($p$) in the map ($m$) by using Eq. (2)
15:     feed, to the fused map (WeVoS), the weights vector of the neuron ($p$) as if it was an input to the network. See
        Eq. (1).
        The weight of the vote calculated in step 14 is used as the learning rate ($\alpha$). The position of that neuron ($p$) is
        considered as the position of the BMU ($v$).
        This causes the neuron of the fused map ($w'_p$) to approximate the neuron of the composing ensemble ($w_p$)
        accordingly to the quality of its adaptation.
16:   end
17: end
```

This new approach not only takes into account each unit's characteristics, but also the topographic ordering of its neighbourhood. The approach is intended to generate more meaningful maps by representing the inner structure of the dataset more faithfully. Those capabilities are a great added value to a CBR system since they facilitate the creation of the structure of the case base, where grouping similar cases together is a great advantage. They are also important when trying to recover the most similar cases to the problem introduced in the system, because of the increased speed of the recovery that results when similar cases are close one to another. Another supplementary advantage of this novel approach is its ability to reduce the redundancy of the case base and therefore its size. Due to the inner pattern matching capabilities typical of the competitive learning algorithms, it is easy for the model to detect which of the input cases are similar to those that have been previously stored. This provides the system with a straightforward method of discarding cases with similar characteristics, thus preventing the addition of cases with no further useful information to the case base.

In Section 5, the main structure and elements of the system presented in this study will be explained. The main elements of the system correspond to the four main phases of the CBR cycle previously explained but with a different technique to implement each phase.

5. WeVoS-CBR: a new prediction solution to the oil spill problem

CBR has already been used to solve maritime problems [17] in which different oceanic variables were involved. In the present case, the data collected from different observations from satellites is pre-processed, and structured in cases. The created cases are the key to obtaining the solutions to future problems through the CBR system.

The developed system determines the probability of finding oil slicks in a certain area. To generate the predictions, the system divides the ocean surface into squares of approximately half a degree per side. The system then determines the amount of slicks present in a square. The squares where the slicks are located are coloured with different gradations depending on the quantity of the squared area covered by oil slicks (see Fig. 6).

Table 1 shows the structure of a case. The variables present in a case can be geographical (longitude and latitude), temporal (date of the case), atmospheric (wind, sea height, bottom pressure, salinity and temperature) or directly related to the problem (number and area of the slicks). All variables are used as inputs defining a case, with the exception of the "*Area of slicks*" which is considered the solution of each case. The "*Area of slicks*" is the value that is trying to be predicted, and is considered as part of the evolution of the present situation.

Once the data is structured, it is stored in the case base. Every case has its temporal situation stored, thus allowing every case to be related with the next situation in the same position. That temporal relationship is what creates the union between problem and solution. The problem is the past case, and the solution is the future case; that is, the future state of the analyzed square. The temporal relationship between two cases, however, is not taken into account by the system's functioning process. The "*Date*" field of each case only serves to determine which case is the future state of which other case, with the objective of assessing the final quality of the results obtained by the WeVoS-CBR.

The described hybrid system includes different artificial intelligence techniques for achieving the objectives of every CBR phase. As shown in Fig. 5, every CBR phase uses an artificial intelligence technique in order to obtain its solution. The phases with related techniques are going to be explained in the following sub-sections.

The data used to train the system was obtained over a period of six months just after the Prestige accident, between November 2002 and April 2003, in a specific geographical area off the north-west of the Galician coast (longitude between

Table 1
Variables that define a case in the *WeVoS-CBR* system.

Variable	Definition	Unit
Longitude	Geographical longitude	°
Latitude	Geographical latitude	°
Date	Day, month and year of the analysis	dd/mm/yyyy
Sea height	Height of the waves in open sea	m
Bottom pressure	Atmospheric pressure in the open sea	N/m^2
Salinity	Sea salinity	ppt (parts per thousand)
Temperature	Celsius temperature in the area	°C
Area of the slicks	Surface covered by the slicks present in the analyzed area	km^2
MeridionalWind	Meridional direction of the wind	m/s
Zonal wind	Zonal direction of the wind	m/s
Wind strength	Wind strength	m/s
Meridional current	Meridional direction of the ocean current	m/s
Zonal current	Zonal direction of the ocean current	m/s
Current strength	Ocean current strength	m/s

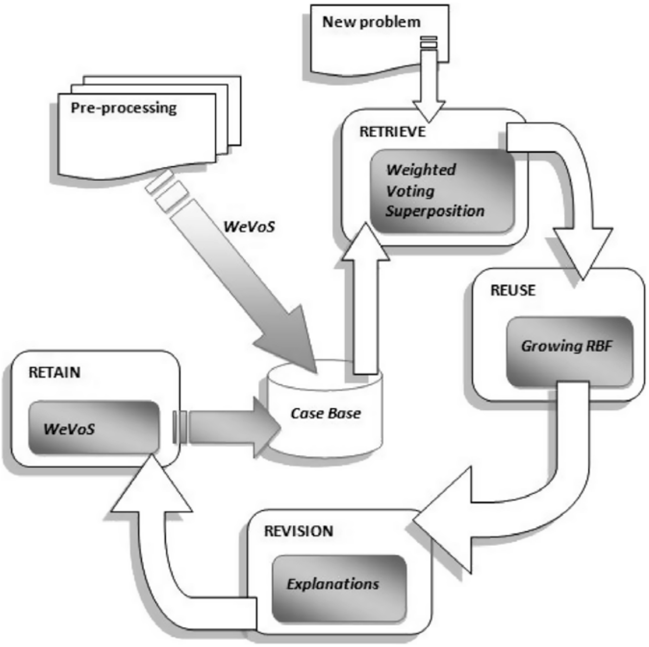
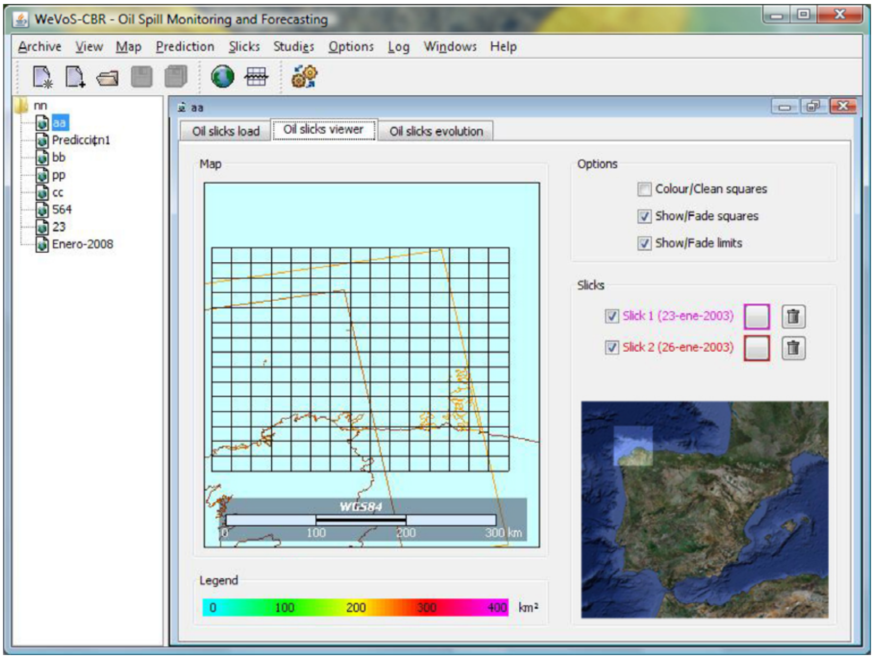


Fig. 5. Case-Based Reasoning cycle adapted to the *WeVoS-CBR* system, including the techniques used in each phase.



□ Colour/Clean squares
☑ Show/Fade squares
☑ Show/Fade limits

☑ Slick 1 (23-ene-2003)
☑ Slick 2 (26-ene-2003)

Fig. 6. Graphical user interface of the *WeVoS-CBR* system. The different components of the system can be observed here.

14 and 6 degrees west and latitude between 42 and 46 degrees north – north-west of the Iberian Peninsula). When all the information is stored in the case base, the system is ready to predict future situations. To generate a prediction, a problem situation must be introduced in the system. Then the cases that are most similar to the problematic situation are retrieved from the case base using the WeVoS-SOM model. Once a collection of cases is chosen from the case base, those cases must be used to generate the solution to the current problem. Growing Radial Basis Functions Networks are used to combine the chosen cases in order to obtain the new solution.

Fig. 6 shows the graphical user interface of the developed system. The image shows the different components of the application (maps, prediction, slicks, studies,...) as well as a visualization of an oceanic area with oil slicks and a squared area to be analyzed. The squared area defines the region to analyze by dividing it into the smaller squares shown. Those squares represent the basic *case* element, which will be stored in the *case base* with its corresponding parameters. The following subsection details each of the steps carried out by *WeVoS-CBR* system to predict the evolution of an oil spill.

5.1. Historical data pre-processing and retrieval of most similar cases

Historical data is used to create the *case base* and to evaluate the proposed model. As previously explained, cases are formed by a series of variables. Before the training of the system, part of the available information is randomly selected and reserved to be used after the construction of the case base to test the system. These cases are not used for the training and are only presented to the system in the test phase. Past solutions are stored in the system, in the case base. In the *WeVoS-CBR* system the cases contain information about the oil slicks (size and number) as well as atmospheric data (wind, current, salinity, temperature, ocean height and pressure).

The *WeVoS-SOM* model is used to structure the case base. Its topographical capabilities are used in this instance to create a model that represents the actual variability of the parameters stored in the cases. At the same time, the inner structure of the case base will make it easier to recover the cases most similar to the problem introduced in the system once it has been trained.

Once the case base has stored the historical data, and the *WeVoS-SOM* has created the internal structure of the case base, having learned from the original distribution of the variables, the system is ready to receive a new problem.

When a new problem enters the system, *WeVoS-SOM* is once again used. The algorithm behaves as if the new problem were going to be stored in the structure, and finds the cases most similar to the problem introduced in the system. In this case, the structure is not changed because it is only being used to obtain the cases most similar to the problem that was introduced. Only in the retain phase does the case base change again and introduce the proposed solution, if it is considered to be correct. The solution introduced in the case base will be eventually used in future problems exactly in the same way as the rest of the information kept in the case base. It will become a new element with its past situation (*problem*) and its future situation (*solution*).

5.2. Reusing historical data to create a new solution

Once the cases most similar to the problem to be solved are recovered from the case base, they are used to generate the solution. The prediction of the future probability of finding oil slicks in an area is generated by using an artificial neural network with a hybrid learning system. An adaptation of Radial Basis Functions Networks is used to obtain that prediction [25,36]. This type of networks was chosen because of the reduced training time compared to other artificial neural network systems, such as Multilayer Perceptrons [50].

Growing RBF networks [30] are used to obtain the predicted future values corresponding to the proposed problem. This adaptation of the RBF networks allows the system to grow during training, gradually increasing the number of elements (prototypes) which act as the centres of the radial basis functions. In this case the creation of the Growing RBF must be made automatically, which implies an adaptation of the original GRBF system [49]. The definition of the error for every pattern is shown in Eq. (3):

$$e_i = l/p \sum_{k=1}^{p} \|t_{ik} - y_{ik}\|, \tag{3}$$

where t_{ik} is the desired value of the kth output unit of the ith training pattern, y_{ik} the actual values of the kth output unit of the ith training pattern.

The Growing RBF pseudocode is shown in Algorithm 2. The threshold value used in Algorithm 2 was empirically determined by an expert on this kind of systems. Once the GRBF network is created, it is used to generate the solution to the proposed problem. The network is trained with the same set of historical data that is included in the case base at every moment. Training data is also used in the prediction generation process. The data used to train the GRBF network is stored as part of the case base and it can be used to generate future predictions without accumulating any additional "noise" to the prediction process.

Algorithm 2. Growing Radial Basis Function pseudocode.

```
1: Calculate the error, $e_i$ (Eq. (3)) for every new possible prototype.
   a. If the new candidate is not among those selected and the error calculated is less than a threshold error, then
      the new candidate is added to the set of accepted prototypes.
   b. If the new candidate already belongs to the accepted ones and the error is less than the threshold error, then
      modify the weights of the units in order to adapt them to the new situation.

2: Select the best prototypes from the candidates
   a. If there are valid candidates, create a new cell centred on the valid candidate.
   b. Else, increase the iteration factor. If the iteration factor reaches 10% of the training population, freeze the
      process.

3: Calculate global error and update the weights.
   a. If the results are satisfactory, end the process. If not, go back to step 1.
```

The GRBF network stays under a training process that continues until the results offered are considered good enough to stop training and begin generating predictions. To determine when a result is good enough, historical cases that were not used in its training are presented to the network and the values that the GRBF yields as output are compared to their corresponding historical values. If these "*Area of slick*" values differ within a threshold from the real values, the results are considered valid. The threshold is calculated by taking into account the number of cases stored in the case base: the more information available, the better the solutions should be, so the threshold used will be lower. When the amount of cases available is not big enough, the results are not brilliant, so the threshold is manually adjusted by consulting an expert. It decreases at the same time the size of the case base grows, improving the results when more information is stored into the case base.

After the training process, the GRBF is used to generate the solutions using the cases retrieved from the case base. This solution consists of a number representing the area of sea covered with oil (in km^2). When a problem enters the system, the network generates a solution for every recovered case similar to the problem, and the average of those solutions is the solution proposed. The solution will be the output of the network using the selected cases from the case base as input data. When new data is introduced in the case base, the GRBF is trained again to adapt itself to the new elements introduced.

The correction of the solution proposed is known when using test data, as all test data used are part of the historical information obtained at the time of the accident, so it is possible to compare the solution proposed by the system with the real value registered.

5.3. Revising the proposed solution and retaining it for future uses

Once the prediction is generated, it is shown to the user. The prediction shows the analyzed area divided into smaller squares. The squares are coloured depending on the presence or absence of slicks in those squares. The intensity of the colour corresponds to the possibility of finding oil slicks in that area. The areas coloured with a higher intensity are those in which there is the highest probability of finding oil slicks.

With this visual presentation of the prediction, the user can check the correction of the proposed solution. But the system provides an automatic method of revision that must be checked by an expert user, regardless of the outcome.

Explanations are used to check the correction of the proposed solution and to justify the solution [52]. To obtain a justification of the given solution, the cases selected from the case base are used once again. To create an explanation, a comparison between different possibilities was used. All the selected cases have their own associated future situation. If we consider the case and its solution as two vectors, we can establish a distance between them, calculating the evolution of the situation under the considered conditions. If the distance between the proposed problem and the solution given is not bigger than the distances obtained from the selected cases, then the solution is a good one, according to the structure of the case base.

The explanations pseudocode is shown in Algorithm 3:

Algorithm 3. Explanations pseudocode.

```
1: For every selected case in the retrieval phase, the distance between the case and its solution is calculated.
2: The distance between the proposed problem and the proposed solution is also calculated.
3: If the difference between the distance of the proposed solution and those of the selected cases is beneath a certain
   threshold value, then the solution is considered to be valid.
4: If not, the user is informed and the process goes back to the retrieval phase, where new cases are selected from the
   case base.
5: If after a series of iterations the system does not produce a good enough solution, then the user is asked to consider
   accepting the best of the generated solutions.
```

The distances are calculated according to the sign of the values, not using its absolute value. This decision is easily justified by the fact that is not the same to move to the north as to the south, even if the distance between two points is equal. If the prediction is considered correct, it will be stored in the case base, and it can then be used in future predictions to obtain new solutions. It will have the same category as the historical data previously stored in the system.

When inserting a new case in the case base, *WeVoS-SOM* is once again used. When adapting to the new solution introduced in the case base, the stored structure grows and improves its capability of generating good results since new knowledge have been introduced in the system.

After explaining the system presented in this paper, the following section will show the results obtained by applying it to the oil spill problem. A summary of the results obtained with the presented system will be explained, as well as a comparison with previous solutions given to the oil spill problem.

6. Experimental results

The data used to train the system were obtained from different satellites. Temperature, salinity, bottom pressure, sea height, number and area of the slicks, along with the location of the squared area and the date were all used to create a case as shown in Table 1. All these data define the problem case and also the solution case. Both are stored in the case base. The solution to a problem is defined as an area and its corresponding variables is the same area, but with the values of the variables changed to the prediction obtained from the CBR system. This can be compared with the real value that was registered at the time of the incident.

The *WeVoS-SOM* model has proved to be more efficient than other existing algorithms used to organize, classify and visualize information [14]. It has obtained better results than simple ensembles of SOMs, and other previous algorithms for the fusion of SOMs.

The main objective of this novel algorithm is the reliable visual representation of a multi-dimensional dataset by enhancing the topology preservation feature of the original SOM model. In the case of the system presented, the topology preservation of the model means that similarity relationships between cases are presented more accurately by the *WeVoS-SOM*, improving the results of the retrieval phase. The *WeVoS-CBR* system uses the *WeVoS-SOM* model to organize the information stored in the case base. The choice of that model will make the recovery process more efficient, as will be shown next.

When the *WeVoS-CBR* system was used with a subset of the data that had not been previously used to train the system, it produced quite hopeful results. Having at our disposal the data from a real state following a determined state facilitated simple and clear tests of the system. The predicted situation was contrasted with the actual future situation that was known a priori, as past data was used to train the system and also to test the correction of its results. In most of the variables, the proposed solution had near 90% accuracy rate when using an appropriate size for the case base (see Table 2).

Table 2 shows a summary of the results obtained. Four different techniques are compared, using an incremental case base containing from 100 to 5000 cases. The increasing number of cases is a result of the analysis of additional images of oil spills added to the set of images used in previous tests, and the reuse of solutions proposed by the system.

Table 2 shows the evolution of the results along with the increase in the number of cases stored in the case base. The numerical results shown represent the average result of a series of tests performed with the available information. The

Table 2
Percentage of good predictions obtained with four different techniques and for a different amount of cases.

Number of cases	RBF (%)	Oceanic CBR (%)	GRBF + CBR (%)	WeVoS-CBR (%)
100	45	39	42	43
500	48	43	46	46
1000	51	47	58	64
2000	56	55	65	72
3000	59	58	68	81
4000	60	63	69	84
5000	63	64	72	87

number of tests carried out in each iteration (every time the case base grows) is ten percent of the size of the case base (for instance, when the case base contained 1000 elements, 100 tests were performed, and so on). This percentage of the data used to test the correction of the system was not previously used in its training. Those cases are also part of the historical information used, so they can be used to check the correction of the predictions generated by the system. For every case stored in the case base there is a *future situation* corresponding to the solution of that situation. The cases used to test the system are chosen randomly from the overall amount of cases.

The results for each of the techniques being analyzed improved when the number of cases stored was increased. The "*RBF*" column represents a simple Radial Basis Function Network that is trained with all the available data. The network receives an area and its parameters as an input. The RBF network gives the probability of finding oil slicks in the analyzed area as an output value, which is considered a solution to the problem. The "*Oceanic CBR*" system represents a previously proposed CBR system that has been also applied to forecast oceanographic conditions [15]. This system uses neural networks in the adaptation process of the recovered cases, in particular a Radial Basis Function network. The neural network has a process of recovering elements from a network knowledge base, from where the neural network retrieves the parameters to calibrate the network. The "*GRBF + CBR*" column corresponds to the possibility of using a GRBF neural network combined with CBR. The recovery from the CBR is achieved by using the Manhattan distance to determine the closest cases to the introduced problem. The GRBF network works in the reuse phase, adapting the selected cases to obtain the new solution.

The "*WeVoS-CBR*" system presented in this work differs from the "*Oceanic CBR*" in the use of GRBF networks to generate the prediction instead of the simple RBF network, generating more accurate predictions. The improvement over the "*GRBF + CBR*" consists of using the WeVoS-SOM algorithm to structure the case base, enhancing the organizational characteristics of the case base.

Upon observing Table 2, it is clear that the results of the "*GRBF + CBR*" column are always better than those of the "Oceanic *CBR*", mainly because useless data are eliminated prior to generating the solution. Finally, the "*WeVoS-CBR*" column shows the results obtained by the proposed system, which are better still than the three previously analyzed solutions.

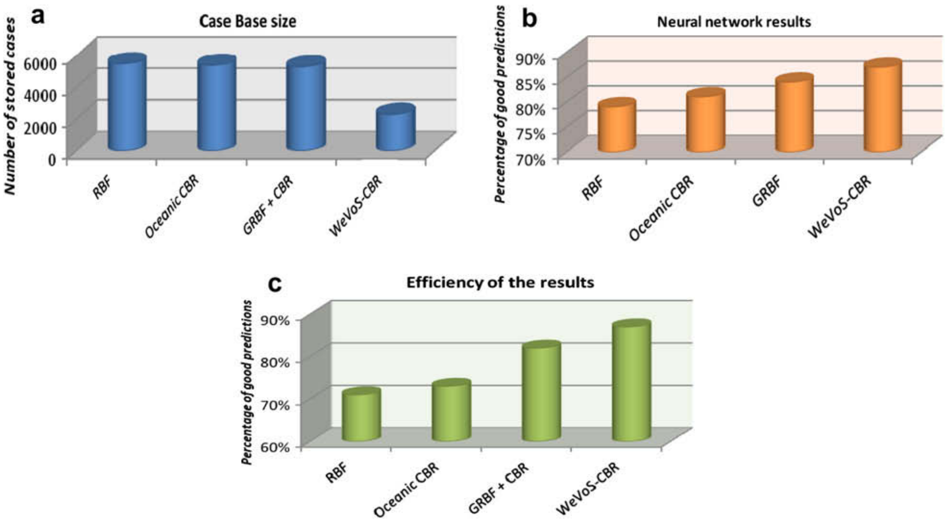


Fig. 7. Summary of some of the improvements achieved with WeVoS-CBR, compared with the other three systems. (a) Comparison of the size of the case base of the different systems compared: "RBF, "Oceanic CBR", "GRBF + CBR" and the "WeVoS-CBR". (b) Comparison of correct predictions achieved by the neural networks included in the different systems compared: "RBF, "Oceanic CBR", "GRBF + CBR" and the "WeVoS-CBR". (c) Comparison of the final accuracy in predictions obtained by each of the four systems used for comparison.

Table 3
Multiple comparison procedure among different techniques.

	RBF	Oceanic CBR	GRBF + CBR	WeVoS-CBR
RBF				
Oceanic *CBR*	*			
GRBF+CBR	*	=		
WeVoS-CBR	*	*	*	

More visual results of those experiments, including complementary information, can be found in Fig. 7.

Using *WeVoS-CBR* improved the efficiency and accuracy of the results over those from previous and simpler applications; in particular, the case base size was reduced mainly from using the new model (*WeVoS-SOM*), which organized the case base and recovered the similar cases. Those improvements can be seen in Fig. 7.

Fig. 7a provides a comparison between the size of the case base of the different methods used for comparison. In the first three methods, no analysis of the store information is performed and all the information available is stored, while in "*WeVoS-CBR*" data are structured and analyzed before being stored. The size of the case base in the WeVoS-CBR system is smaller mainly from using an organizing algorithm like WeVoS-SOM that structures the information depending on its values, and not only on the different elements available. With the same amount of initial information, the WeVoS-CBR system does not store redundant information, thus reducing the final quantity of cases stored and making it easier to recover the most similar cases to the problem introduced.

Fig. 7b shows a comparison between the results obtained in the reuse phase by the different neural networks used for comparison. The "*RBF*" neural network represents the classic version of this network, where the internal structure of the network does not change during the training process. That network is used in the two first elements, generating better results in the "*Oceanic CBR*" because of a better selection of the cases available from the case base. The "*GRBF*" network represents the new version used both within the "*GRBF* + *CBR*" and "*WeVoS-CBR*" system, where the growth of the neural network is progressive and adapted to the data through the training process. The results obtained by the *GRBF* network are better than the simple *RBF* because of neurons are better adapted to the data used to solve the problem. The structure of the GRBF is *similar* to the structure of the case base, growing at the same time the case base grows. The results are better in the "*WeVoS-CBR*" than in the "*GRBF* + *CBR*" because the organization of the case base is better in the proposed system, and so is the selection of the cases to be reused by the system.

In Fig. 7c, the *WeVoS-CBR* system predicting results are compared with those obtained with the other systems used for comparison. The first one, "*RBF*", is a simple RBF network, where data is introduced by training the network, and the results are obtained by generalization applied to the information internally stored in the network. The second system, called "*Oceanic CBR*" corresponds to the one previously explained as part of Fig. 7a. The third one "*GRBF* + *CBR*" is a conjunction of a basic CBR system with a GRBF network created to generate the predictions. The results shown here are a more visual summary of the numerical results previously shown in Table 2. The improvement obtained by the WeVoS-CBR system can be clearly seen both in the graph and in the table, compared to the rest of the techniques used in this work.

As shown in Fig. 7 the *WeVoS-CBR* system is better than the other three systems both in the quality of the predictions generated (Fig. 7c) and in the size of the case base that stores the information (Fig. 7a).

Table 3 shows a multiple comparison procedure (*Mann–Whitney* test [35]) 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. Table 3 shows that the *WeVoS-CBR* system presents statistically significant differences compared to the other models.

The results shown in this section confirm the initial theoretical beliefs that indicated that the system developed would generate more accurate predictions. It is, in fact, more accurate than previous approximations to the oil spill problem.

7. Conclusions and future work

In this study, a new hybrid forecasting system called *WeVoS-CBR* was presented and explained. It is a new solution for predicting the presence or absence of oil slicks in a certain sea area after an oil spill.

This system used data acquired from different orbital satellites in order to create a CBR environment. The data must be previously adapted to the structure required by the CBR system in order to be stored as a case. The explained system uses different artificial intelligence techniques in order to obtain a correct prediction.

A new summarization algorithm, the Weighted Voting Superposition algorithm, was applied to the SOM (*WeVoS-SOM*) and used both to organize the case base and to retrieve the most similar cases to the one introduced as a problem to the system. The great organization capabilities of the new algorithm allow the system to create a valid structure for the case base and to easily recover similar cases from the case base.

To obtain a prediction using the cases recovered from the case base, Growing Radial Basis Function Networks were used. The evolution of the RBF networks implies a better adaptation to the structure of the case base, which is organized using the *WeVoS-SOM* algorithm. The results using Growing RBF networks instead of simple RBF networks are about a 4% more accurate (average improvement of the different sizes considered of case base), as can be seen in Table 2 and Fig. 7, which is a good improvement. The improvement is calculated by considering the average of the different measurements done.

The system's predictive ability was proved with previously known conditions, and showed better results than previously used techniques. As shown in Table 2 and Fig. 7, the use of a combination of techniques within the WeVoS-CBR system makes it possible to obtain better results than when using the CBR alone (17% better by average, see Table 2), and also better than using isolated techniques, without the integration feature produced by the CBR (11% better by average, see Table 2). This model may be used in other oceanic areas if historical data is available or if this historical data can be generated by expert oceanographers.

The next step is to generalize the learning, acquiring new data to create a base of cases big enough to have solutions for every season. Another improvement is to create an on-line system that can store the case base in a server and generate the solutions dynamically in response to different requests. The on-line version will include real time connection to data servers providing weather information of the current situations in order to predict *real future* situations. Another line of future work will be focused on the application of different topology preserving models such as the ViSOM to improve the pattern recognition and clustering features of this system. Also different cases of study will be contemplated.

Acknowledgments

This research has been partially supported by the projects BU006A08 of the JCyL and VEM2003-20578-C08-03 (The use of teledetection and artificial intelligence systems in the detection, monitoring and prediction of oil spills in an open ocean) of the MCYT. The authors would also like to thank the manufacturer of components for vehicle interiors, Grupo Antolin Ingeniería, S.A. in the framework of the project MAGNO 2008 – 1028 – CENIT Project funded by the Spanish Ministry of Science and Innovation.

References

[1] A. Aamodt, A Knowledge-Intensive, Integrated Approach to Problem Solving and Sustained Learning, Knowledge Engineering and Image Processing Group, University of Trondheim, 1991.
[2] A. Aamodt, E. Plaza, Case-based reasoning: foundational issues methodological variations, and system approaches, AI Communications 7 (1994) 39–59.
[3] K.D. Althoff, J. Mänz, M. Nick, Maintaining experience to learn: case studies on case-based reasoning and experience factory, in: Proceedings of the 6th Workshop Days of the German Computer Science Society (GI) on Learning, Knowledge, and Adaptivity (LWA 2005), 2005, pp. 118–125.
[4] B. Baruque, E. Corchado, J. Rovira, J. Gonzalez, Application of topology preserving ensembles for sensory assessment in the food industry, in: Intelligent Data Engineering and Automated Learning (IDEAL 2008), 2008, pp. 491–497.
[5] B. Baruque, E. Corchado, H. Yin, ViSOM ensembles for visualization and classification, in: International Work-Conference on Artificial Neural Networks (IWANN'07), San Sebastián, Spain, Springer, Heidelberg, 2007, pp. 235–243.
[6] C.J. Beegle-Krause, GNOME: NOAA's next-generation spill trajectory model, in: OCEANS'99 MTS/IEEE. Riding the Crest into the 21st Century, vol. 3, 1999, pp. 1262–1266.
[7] A. Benmecheta, A. Lansari, Monitoring of oil pollution by GIS and remote-sensing case of West Algeria Harbours, in: IEEE International Symposium on Signal Processing and Information Technology, 2007, pp. 874–879.
[8] I. Bichindaritz, C. Marling, Case-based reasoning in the health sciences: what's next?, Artificial Intelligence in Medicine 36 (2006) 127–135.
[9] L. Breiman, Bagging predictions, Machine Learning 24 (1996) 123–140.
[10] I. Brovchenko, A. Kuschan, V. Maderich, M. Zheleznyak, The modelling system for simulation of the oil spills in the Black Sea, in: 3rd EuroGOOS Conference: Building the European Capacity in Operational Oceanography, 2002, p. 192.
[11] P. Carracedo, S. Torres-López, M. Barreiro, P. Montero, C.F. Balseiro, E. Penabad, P.C. Leitao, V. Pérez-Munuzuri, Improvement of pollutant drift forecast system applied to the Prestige oil spills in Galicia Coast (NW of Spain): development of an operational system, Marine Pollution Bulletin 53 (2006) 350–360.
[12] C. Carrascosa, J. Bajo, V. Julian, J.M. Corchado, V. Botti, Hybrid multi-agent architecture as a real-time problem-solving model, Expert Systems with Applications 34 (2007) 2–17.
[13] G. Copeland, W. Thiam-Yew, Current data assimilation modelling for oil spill contingency planning, Environmental Modelling and Software 21 (2006) 142–155.
[14] E. Corchado, B. Baruque, H. Yin, Boosting unsupervised competitive learning ensembles, in: International Conference on Artificial Neural Network (ICANN 2007), vol. 4668, 2007, pp. 339–348.
[15] J.M. Corchado, J. Aiken, Hybrid artificial intelligence methods in oceanographic forecasting models, IEEE SMC Transactions (2002) 307–313.
[16] J.M. Corchado, J. Bajo, A. Abraham, GERAmI: improving the delivery of health care, IEEE Intelligent Systems. Special Issue on Ambient Intelligence (2008) 19–25.
[17] J.M. Corchado, F. Fdez-Riverola, FSfRT: forecasting system for red tides, Applied Intelligence 21 (2004) 251–264.
[18] B. Decker, J. Rech, K.D. Althoff, A. Klotz, E. Leopold, A. Voss, eParticipative process learning—process-oriented experience management and conflict solving, Data and Knowledge Engineering 52 (2005) 5–31.
[19] F. Diaz, F. Fdez-Riverola, J.M. Corchado, Gene-CBR: a case-based reasoning tool for cancer diagnosis using microarray data sets, Computational Intelligence 22 (2006) 254–268.
[20] C. Douligeris, J. Collins, E. Iakovou, P. Sun, R. Riggs, C.N.K. Mooers, Development of OSIMS: an oil spill information management system, Spill Science and Technology Bulletin 2 (1995) 255–263.
[21] A.A. Elhakeem, W. Elshorbagy, R. Chebbi, Oil spill simulation and validation in the Arabian (Persian) Gulf with special reference to the UAE Coast, Water, Air, and Soil Pollution 184 (2007) 243–254.
[22] F. Fdez-Riverola, J.M. Corchado, FSfRT: forecasting system for red tides, Applied Intelligence 21 (2004) 251–264.
[23] F. Fdez-Riverola, E.L. Iglesias, F. Díaz, J.R. Méndez, J.M. Corchado, Applying lazy learning algorithms to tackle concept drift in spam filtering, Expert Systems with Applications 33 (2007) 36–48.
[24] B. Fritzke Unsupervised clustering with growing cell structures, in: IJCNN-91 – Seattle International Joint Conference on Neural Networks, vol. 2, 1991, pp. 531–536.
[25] S. Haykin, Neural Networks, Prentice-Hall, Upper Saddle River, NJ, 1999.
[26] A. Herrero, E. Corchado, M.A. Pellicer, A. Abraham, MOVIH-IDS: a mobile-visualization hybrid intrusion detection system, Neurocomputing, in press.
[27] T. Heskes, Balancing between bagging and bumping, in: M.C. Mozer, M.I. Jordan, T. Petsche (Eds.), Proceedings of the 1996 Conference on Advances in Neural Information Processing Systems, vol. 9, 1997, pp. 466–472.

[28] C.-C. Hsu, C.-S. Ho, A new hybrid case-based architecture for medical diagnosis, Information Sciences 166 (2004) 231–247.

[29] A. Jordi et al, Scientific management of Mediterranean coastal zone: a hybrid ocean forecasting system for oil spill and search and rescue operations, Marine Pollution Bulletin 53 (2006) 361–368.

[30] N.B. Karayiannis, G.W. Mi, Growing radial basis neural networks: merging supervised and unsupervised learning with network growth techniques, IEEE Transactions on Neural Networks 8 (1997) 1492–1506.

[31] T. Kohonen, Self-Organizing Maps, Springer, Berlin, Germany, 1995.

[32] J.H. Lee, S.H. Ha, Recognizing yield patterns through hybrid applications of machine learning techniques, Information Sciences 179 (2009) 844–850.

[33] X. Liu, K.W. Wirtz, Decision making of oil spill contingency options with fuzzy comprehensive evaluation, Water Resources Management 21 (2007) 663–676.

[34] X. Liu, K.W. Wirtz, Sequential negotiation in multiagent systems for oil spill response decision-making, Marine Pollution Bulletin 50 (2005) 469–474.

[35] H.B. Mann, D.R. Whitney, On a test of whether one of two random variables is stochastically larger than the other, The Annals of Mathematical Statistics 18 (1947) 50–60.

[36] B. Martin, A. Sanz, Redes neuronales y sistemas borrosos, Editorial Ra-Ma, Zaragoza, 1997.

[37] D. Menemenlis et al, NASA supercomputer improves prospects for ocean climate research, EOS Transactions 86 (2005) 89–95.

[38] S. Montani, L. Portinale, G. Leonardi, R. Bellazzi, Case-based retrieval to support the treatment of end stage renal failure patients, Artificial Intelligence in Medicine 37 (2006) 31–42.

[39] J.M.T. Palenzuela, L.G. Vilas, M.S. Cuadrado, Use of ASAR images to study the evolution of the Prestige oil spill off the Galician coast, International Journal of Remote Sensing 27 (2006) 1931–1950.

[40] R. Pavón, F. Díaz, R. Laza, V. Luzón, Automatic parameter tuning with a Bayesian case-based reasoning system. A case of study, Expert Systems With Applications (2008).

[41] R. Periánez, A. Pascual-Granged, Modelling surface radioactive, chemical and oil spills in the Strait of Gibraltar, Computers and Geosciences 34 (2008) 163–180.

[42] R. Periáñez, Chemical and oil spill rapid response modelling in the Strait of Gibraltar–Alborán Sea, Ecological Modelling 207 (2007) 210–222.

[43] R. Periáñez, A. Pascual-Granged, Modelling surface radioactive, chemical and oil spills in the Strait of Gibraltar, Computers and Geosciences 34 (2008) 163–180.

[44] E. Plaza, L. McGinty, Distributed case-based reasoning, The Knowledge Engineering Review 20 (2006) 261–265.

[45] D. Polani, Measures for the organization of self-organizing maps, in: Springer Studies in Fuzziness and Soft Computing Series, 2001, pp. 13–44.

[46] G. Polzlbauer, Survey and comparison of quality measures for self-organizing maps, in: Proceedings of the Fifth Workshop on Data Analysis (WDA'04), 2004, pp. 67–82.

[47] J.M. Price, Z.G. Ji, M. Reed, C.F. Marshall, M.K. Howard, N.L. Guinasso Jr., W.R. Johnson, G.B. Rainey, Evaluation of an oil spill trajectory model using satellite-tracked, oil-spill-simulating drifters, Proceedings of OCEANS 2003, vol. 3, 2003.

[48] M. Reed, N. Ekrol, H. Rye, L. Turner, Oil spill contingency and response (OSCAR) analysis in support of environmental impact assessment offshore Namibia, Spill Science and Technology Bulletin 5 (1999) 29–38.

[49] F. Ros, M. Pintore, J.R. Chrétien, Automatic design of growing radial basis function neural networks based on neighbourhood concepts, Chemometrics and Intelligent Laboratory Systems 87 (2007) 231–240.

[50] D. Rumelhart, G. Hinton, R. Williams, Learning internal representations by error propagation, Parallel Distributed Processing: Explorations in the Microstructure of Cognition 1 (1986) 318–362.

[51] A.H.S. Solberg, G. Storvik, R. Solberg, E. Volden, Automatic detection of oil spills in ERS SAR images, IEEE Transactions on Geoscience and Remote Sensing 37 (1999) 1916–1924.

[52] F. Sørmo, J. Cassens, A. Aamodt, Explanation in case-based reasoning – perspectives and goals, Artificial Intelligence Review 24 (2005) 109–143.

[53] D. Stammer, C. Wunsch, R. Giering, C. Eckert, P. Heimbach, J. Marotzke, A. Adcroft, C.N. Hill, J. Marshall, Volume, heat, and freshwater transports of the global ocean circulation 1993–2000 estimated from a general circulation model constrained by World Ocean Circulation Experiment (WOCE) data, Journal of Geophysical Research 108 (2003) 3007–3029.

[54] Z. Sun, G. Finnie, K. Weber, Case base building with similarity relations, Information Sciences 165 (2004) 21–43.

[55] K. Topouzelis, V. Karathanassi, P. Pavlakis, D. Rokos, Detection and discrimination between oil spills and look-alike phenomena through neural networks, ISPRS Journal of Photogrammetry and Remote Sensing 62 (2007) 264–270.

[56] C.Y. Tsai, C.C. Chiu, A case-based reasoning system for PCB principal process parameter identification, Expert Systems With Applications 32 (2007) 1183–1193.

[57] A. Ulaş, M. Semerci, O.T. Yıldız, E. Alpaydın, Incremental construction of classifier and discriminant ensembles, Information Sciences 179 (2009) 1298–1318.

[58] I. Watson, Case-based reasoning is a methodology not a technology, Knowledge-Based Systems 12 (1999) 303–308.

[59] J. Wu, Y. Yu, Connectionism-based CBR method for distribution short-term nodal load forecasting, in: TENCON 2005 IEEE Region, vol. 10, 2005, pp. 1–6.

[60] B.S. Yang, T. Han, Y.S. Kim, Integration of ART-Kohonen neural network and case-based reasoning for intelligent fault diagnosis, Expert Systems With Applications 26 (2004) 387–395.

[61] F. Zhang, M.H. Ha, X.Z. Wang, X.H. Li, Case adaptation using estimators of neural network, in: Proceedings of 2004 International Conference on Machine Learning and Cybernetics, vol. 4, 2004, pp. 2162–2166.