Paper No.
07553

NACE INTERNATIONAL CORROSION 2007 CONFERENCE & EXPO

Case-based Reasoning Model of CO_2 Corrosion Based on Field Data

Burzin Khajotia
Ohio University, IMSE Department
270 Stocker Center
Athens, OH 45701

Dusan Sormaz
Ohio University, IMSE Department
277 Stocker Center
Athens, OH 45701

Srdjan Nesic
Ohio University, Institute for Corrosion and Multiphase Technologies
342 W State Street
Athens, OH 45701

ABSTRACT

An important aspect in corrosion prediction for oil and gas wells and pipelines is to obtain a realistic estimate of the corrosion rate. Corrosion rate prediction involves developing a predictive model that utilizes commonly available operational parameters, existing lab/field data and theoretical models to obtain realistic assessments of corrosion rates. The Case-based Reasoning (CBR) model for CO_2 corrosion prediction is designed to mimic the approach of experienced field corrosion personnel. The model takes knowledge of corrosion rates for existing cases and uses CBR techniques and Taylor series expansion to predict corrosion rates for new fields having somewhat similar parameters. The corrosion prediction using CBR model is developed in three phases: case retrieval, case ranking, and case revision. In case retrieval phase, the database of existing cases is queried in order to identify the group of cases with similar values of critical corrosion parameters. Those cases are ranked in the second phase, using a modified Taylor series expansion of the corrosion function around each case. The most similar case is passed to the third phase: case revision. The correction of the corrosion rate by using a mechanistic corrosion model is utilized in order to predict the corrosion rate of the problem under consideration. The (CBR) model has been implemented as a prototype and verified on a large hypothetical case database and a small field database with real data.

Keywords: Case Based Reasoning, CBR, corrosion prediction

INTRODUCTION

An important aspect in corrosion prediction for oil and gas wells and pipelines is to obtain a realistic estimate of the corrosion rate based on the existing experience. During the last few years, several corrosion rate prediction models have been developed by oil companies and research institutes and are still currently being developed. Most of the available predictive models tend to be either very conservative in their interpretation of results or focus on a narrow range of parametric effects. The models have different approaches in calculating corrosion rates focusing to different extents on various factors such as CO_2 corrosion, iron carbonate film development, oil wetting, localized corrosion etc. thereby limiting the scope of the model's application in realistic assessment of corrosion rates. Most common models cannot be used in situations where H_2S or organic acids dominate the corrosion process. For corrosion rate prediction there are currently two types of computer programs: programs based on the field related data and programs based on the laboratory results. Neither group of programs can predict the order of magnitude of the corrosion rate for all field conditions.[1] The prediction of corrosion rate involves developing a model that utilizes commonly available operational parameters, utilizing existing lab/field data and theoretical models to obtain realistic assessments of corrosion rates while developing a computational approach that integrates both numerical (lab trends) and heuristic (field data and experience) information and knowledge about corrosion prediction.

PREVIOUS WORK

There are several approaches utilized in developing corrosion prediction models in the literature: mechanistic, empirical, and hybrid.[1] Mechanistic models provide mathematical formulation of the chemical and electrochemical phenomena of the corrosion using mass, energy and charge balances. Empirical models are based on development of equations based on experimental data with appropriate statistical analysis. Hybrid models are a combination of the two. Empirical or even hybrid models, lack the ability to provide an insight into the root cause of corrosion. Such models also lack the ability to confidently extrapolate predictions outside the calibration domain. These models work best on laboratory conditions and limited field conditions.

Models based solely on direct field data are developed in such a fashion that the variables and weights used are adjusted to fit the existing data to a high level of accuracy. Such models fit equations to give positive results against existing data available. However, they too are unable to provide the user with a deep insight into the root cause behind the problem. These models have the inherent drawback that any extensions of the model to include new phenomena cannot be done with ease and are at times not possible at all without recoding.

Majority of published mechanistic and hybrid corrosion models are rather basic and they tend to ignore the impact of H_2S and organic acid effects on corrosion, which at times actually dictate the corrosion process. One of the widely accepted corrosion models was the De Waard and Milliams mechanistic model of 1975 relating temperature, partial pressure of CO_2 and velocity.[2] This model was revised and extended several times and it has been converted into a semi-empirical model.[3] Several other mechanistic models are reviewed by Nyborg.[4] These include Kjeller Sweet Corrosion (KSC) model which focuses on the effect of various electrochemical and diffusion process and the OLI model of corrosion which combines various chemical, thermodynamic and electrochemical models with a stress on thermodynamic calculation of phase equilibrium. A combined multiphase flow corrosion model, MULTICORP, from Ohio University focuses on the effects of protective films and changing pH as well as organic acid, amongst other factors to predict corrosion in a multiphase flow environment.[5]

Neural network models are an extension of empirical models. They are not based on theoretical background, and their parameters do not have physical meaning, but they represent statistical best fit

(minimal error) set of parameters based on their training data set. Nesic et al has proposed a nonlinear empirical CO_2 corrosion neural network based model. The model was integrated with a genetic algorithm which assisted it during training providing a set of data which would cause the least training error.[3]

Case-based Reasoning (CBR) is also an empirical type of model which finds its application in areas where previous knowledge of conditions or situations provides a basis for prediction of outcomes for similar new situations.[7] CBR has its usage in fields varying from microbiology genome sequence, protein structure analysis spatial reasoning to manufacturing planning and scheduling problems. The CBR model searches through the cases to identify the closest set of cases based on input data and weights of different parameters considered. CBR is traditionally used in situations where factors considered are discrete in nature or can be classified into discrete sets. CBR's true flexibility lies in its ability to process incomplete data sets with real time learning by incorporating new cases directly into its dataset. Several popular CBR systems have been reviewed by Watson and Marir.[6] The CLAVIER system developed by Lockheed, Palo Alto is a well known commercial application of a CBR system which helps create layouts and sub-layouts of parts to be entered into autoclaves where it has provided a 90% process efficiency improvement. British Airways uses a system 'CaseLine' as an aircraft fault diagnostic and repair tool. It functions based on a combination of three different inputs explaining mechanical failures and retrieving their corresponding solutions from the case-base.[6]

METHODOLOGY

This section provides the explanation of the methodology developed for Case Based Reasoning – Taylor Series (CBR-TS) model. First, the brief overview of the model is presented. After that, three phases of the model, case search, case ranking and case revision are described in detail and illustrated on simple examples.

Case-based Reasoning – Taylor Series (CBR-TS) Model Overview

The CBR-TS model is designed to mimic the approach of experienced field corrosion personnel. The model takes existing knowledge of field parameters and conditions, and experimentally measured corrosion rates under such conditions, and uses this available knowledge to predict corrosion rates for new cases having "similar" parameters. This method of reasoning by referencing an existing knowledge base to extract similar cases is based on a case-based reasoning paradigm, mathematical formulae, and equations of Taylor Series.

Case-based reasoning (CBR) is a branch of artificial intelligence which uses available knowledge from previous experiences (cases) to perform intelligent reasoning.[7] CBR assumes that previous cases have been collected and organized, such that a set of independent parameters in each case is identified, and a conclusion, solution, or dependent variable from each case is also collected. The collection of these cases represents a so-called *case-base*. Case-based reasoning is performed through a series of steps in which a new problem is compared with a case-base and depending on its similarity to an existing case(s), a new conclusion is inferred, or a dependent variable is estimated. The steps of the general CBR cycle are as follows (see Figure 1):

1) **Retrieve**, in this phase a new problem is compared with the cases in the case-base and, according to previously established similarity criteria, one or more similar cases is returned from the search. Usually the search is reduced to the most similar case, but few cases may be forwarded to the next phase,

2) **Reuse**, in this phase a knowledge, reasoning or conclusion from the most similar case is utilized to derive the new conclusion for the problem by performing necessary modifications to the most similar case,
3) **Revise**, in this phase a revision of the conclusion from the reuse phase is performed by using other cases in the case-base, in order to avoid excessive influence from the most similar case, and to broaden the reasoning procedure,
4) **Retain** is the last phase of the CBR cycle in which the reasoning from the new problem is kept within the case-base for the use in finding solutions in the future problems.

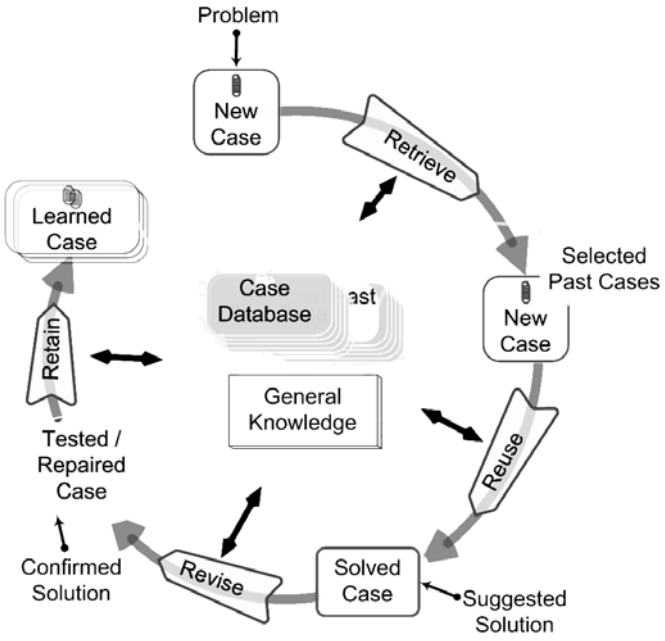


Figure 1. Case-based Reasoning (CBR) cycle

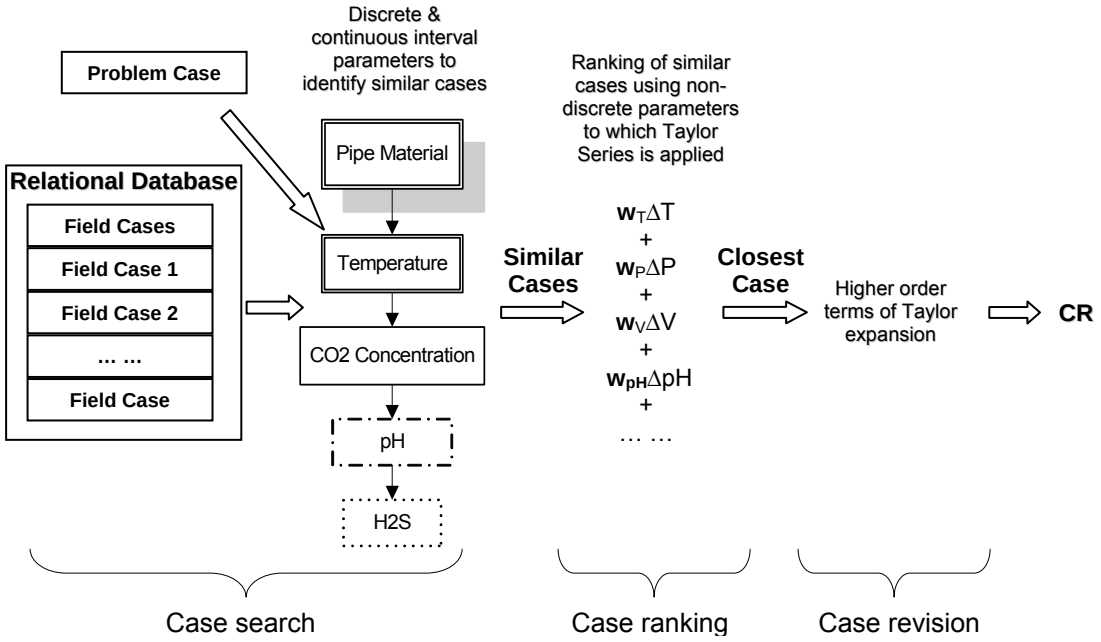


Figure 2. Case-Based Reasoning - Taylor Series corrosion algorithm.

The currently developed CBR-TS algorithm is slightly different and consists of three main parts as shown in Figure 2:

- **Case search:** The identification of similar cases is done on a set of discrete and grouped continuous parameters to eliminate cases that would cause misguided predictions. For example, comparing a problem having stainless steel piping material with an existing case which has all other parameters the same except pipe material being for example carbon steel would give erroneous results.
- **Case ranking:** The pool of similar cases is ranked for closeness to the problem at hand by using first order Taylor Series expansion and an existing corrosion model to help provide the weights that characterize parameter impact on the corrosion rate. For example, the impact of pH difference on the corrosion rate is typically much higher then the impact of total pressure difference between the problem case and the most similar case and this has to be accounted for when ranking the cases for similarity. Weights of several important parameters are used to calculate each case ranking, which is based on weighted difference between the parameters in the problem case and the similar case. In this way, i.e. by ordering the similar cases and selecting the case(s) with the closest ranking, a selection of the most similar case(s) is done.
- **Case revision:** The parameters of the most similar case are then used as a base to extrapolate using Taylor Series expansion and an existing corrosion model to provide a more precise answer for the new problem case at hand.

These three main parts of the CBR-TS algorithm are discussed in more details below.

Case Search

Similarity of cases is defined by similarity of discrete parameters, and by similarity of continuous parameters. For discrete parameters similarity is defined as the number of parameters with the same value, while for continuous parameters similarity is defined by belonging to the same value interval, that is to say, the domain of possible values for the parameter is divided into several intervals (for example, for pH value one may define four intervals: pH<4, 4<pH<6, 6<pH<8, 8<pH) and problem value is classified to one of those intervals.

Case search is performed for one parameter at a time, and if no cases are found, the similarity interval for the parameter is expanded to include more cases. Order of parameters and the number of intervals for expansion has been defined to accommodate various domains and case-bases with different number of cases, from small to very large. In order to accommodate a variety of possible domains, a total of four different CBR algorithms were developed differing in complexity and search mechanisms as well as efficiency which are based on the size of the database i.e. number of cases available. These algorithms are the following:

1. *Fixed order, complete expansion* algorithm maintains the order of parameter consideration. As parameters are sequentially taken into consideration to retrieve cases, when no cases are returned due to consideration of a particular parameter within a particular range, that parameter's range is expanded to include the entire range of possible values.
2. *Fixed order, immediate incremental expansion* algorithm also maintains the order of parameter consideration; however when no cases are returned due to consideration of a particular parameter within a particular range, the parameter's range is expanded by predetermined range increments.
3. *Fixed order, delayed incremental expansion* algorithm there is a fixed order in which the parameters are considered. When no cases are returned due to a particular parameter and its range, the parameter is dropped temporally and the next parameter is considered. Once all parameters are considered, the

dropped parameters are considered sequentially in their original order and the parameter's range is expanded one range step at a time till cases are returned.

4. *Flexible order* algorithm does not maintain the order in which the parameters are considered. The range step for each parameter as identified from the problem case is given more importance then the parameters themselves. The order of parameters to be considered depends on the proximity of the respective problem value of the parameter to its range border.

All four algorithms perform the search in the case-base, one parameter at a time, and then loop through each parameter in order to retrieve a set of similar cases based on all parameters. For example the detailed steps of the *Fixed order complete expansion* algorithm are the following:

1. Use predefined sequence of parameter priorities and predefined continuous parameter intervals;
2. Query case-base for each parameter based on the sequence and using the problem value of the parameter to select the interval, use the result of the previous query as the starting point for the next query;
3. If no cases are returned in the query, remove the parameter interval for the last parameter and query for all available cases again;
4. If all parameters are queried stop, else repeat step 2 for the next parameter in priority sequence.

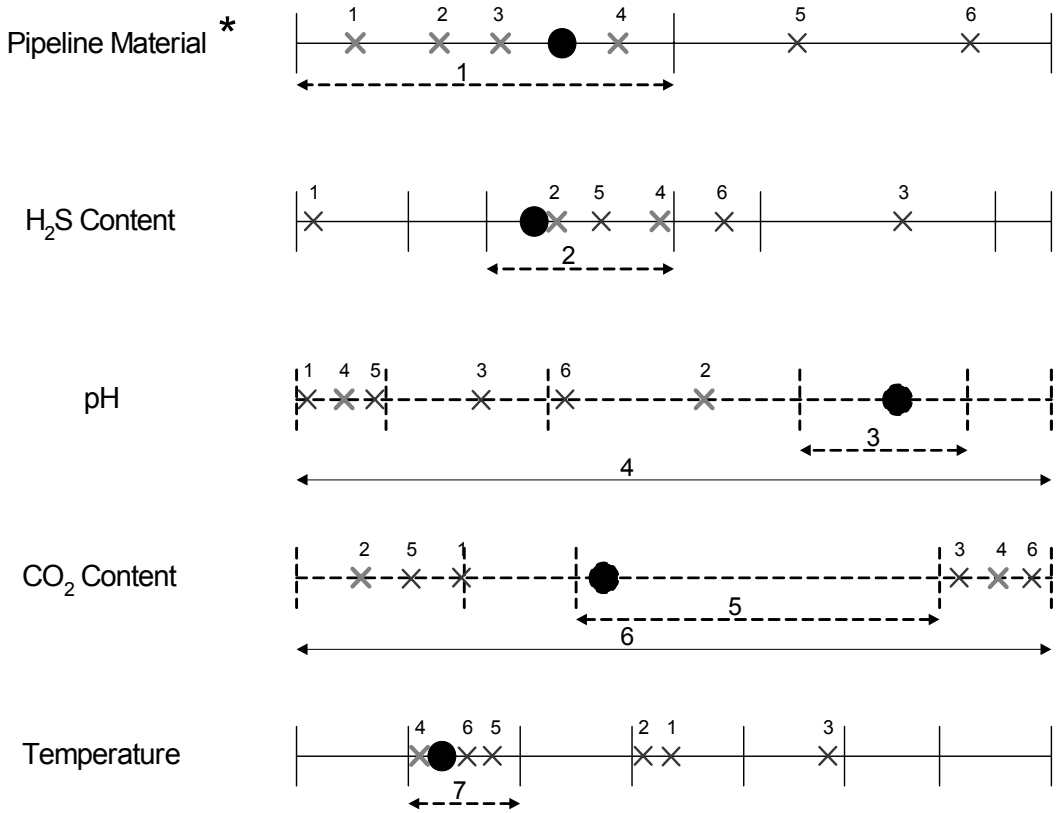


Figure 3. Fix Order Complete Expansion CBR Algorithm

The steps of fixed order, complete expansion algorithm are illustrated on a small example (with 6 cases) in Figure 3. In this example order of corrosion parameters is: pipe material, H_2S content, pH, CO_2 content, temperature. Vertical lines represent intervals for each parameter, X's represent the parameter values for the cases and black circles represent the parameter values for the problem. The search starts with pipeline material and cases 1, 2, 3, and 4 are selected (step 1, they are shown as gray X's on pipeline material line). H_2S content parameter is considered next (only among cases 1, 2, 3 and 4) and

cases 2 and 4 are selected (step2). After that search on pH value is performed (step 3) in which no case is selected, pH interval is expanded (step 4) and cases 2 and 4 are selected (kept). Procedure is repeated for CO_2 content (steps 5 and 6) and temperature (step 7) and only case 4 is selected for a set of similar cases.

The key features of all algorithms is their flexibility in considering different parameters as well as incomplete problem cases which have only a limited number of parameters specified. The lack of data in the problem case may however affect the accuracy of the final predicted corrosion rate, as it would in any other corrosion prediction model or even in the case of similar reasoning made by experienced corrosion personnel.

Case Ranking

In a one-dimensional Taylor series expansion, an unknown continuous function y = f(x) can be approximated at a point with a known value $y_0 = f(x_0)$ and known derivatives at that point, in the form of a polynomial series:

$$(1) \quad f(x) = f(x_0) + \frac{\partial f}{\partial x}(x - x_0) + \frac{1}{2!}\left[\frac{\partial^2 f}{\partial x^2}(x - x_0)^2\right] + \ldots$$

It is here assumed that the corrosion rate is a continuous function of several parameters, which can be expressed in the form CR = f(x, y, z …). Then the case ranking can be done by considering each parameter individually and expanding the CR function in a Taylor series around the known value of the CR in the database (case-base). For example an expansion of the corrosion rate which is a function of just two variables, e.g. temperature (T) and pressure (P), using a Taylor series would give:

$$\begin{aligned} CR(T_{Pr}, P_{Pr}) &= CR(T_{DB} + \Delta T, P_{DB} + \Delta P) \\ &= CR(T_{DB}, P_{DB}) + \Delta CR \\ &= CR(T_{DB}, P_{DB}) + \left[\left.\frac{\partial CR}{\partial T}\right|_{DB}(T_{Pr} - T_{DB}) + \left.\frac{\partial CR}{\partial P}\right|_{DB}(P_{Pr} - P_{DB})\right] + \varepsilon \end{aligned} \quad (2)$$

Where the subscript $_{Pr}$ refers to the unknown (*Problem*) case and subscript $_{DB}$ refers to the known (*Database*) case; ΔT, ΔP, and ΔCR are the deviations between the problem case and the database case for temperature, pressure and corrosion rate respectively; ε is the error involved in the Taylor series approximation that includes higher order derivatives of the function. In simple words using the Taylor series expansion, the corrosion rate of the problem case at hand CR_{Pr} can be found from the known corrosion rate value found in the database CR_{DB} plus a change in the corrosion rate ΔCR due to difference in parameters: ΔT and ΔP between the problem case and the database case:

$$(3) \quad CR_{Pr} = CR_{DB} + \Delta CR$$

In order to determine how similar a case in the database is to the problem case, it is possible to calculate a *dissimilarity* measure using Taylor series expansion shown above. This dissimilarity measure can be defined as the magnitude of difference in the corrosion rate due to the difference in parameters between the problem and the database case, which is actually the same as the ΔCR above. The overall difference in the corrosion rate $\Delta CR = CR_{Pr} - CR_{DB}$ can be broken down in a series of contributions from each individual parameter:

$$(4) \quad \Delta CR = \Delta CR_T + \Delta CR_P$$

This by comparison with the equation (2) above gives:

$$(5)\quad \Delta CR_T = \left.\frac{\partial CR}{\partial T}\right|_{DB} \left(T_{Pr} - T_{DB}\right)$$

$$(6)\quad \Delta CR_P = \left.\frac{\partial CR}{\partial P}\right|_{DB} \left(P_{Pr} - P_{DB}\right)$$

where the ΔCR_T is the dissimilarity in the corrosion rate between the problem case and database case due to difference in temperatures and ΔCR_P is the dissimilarity due to the difference in pressure. Values of temperature T_{Pr} and T_{DB} for the problem and database cases are known at the outset, so are the pressures P_{Pr} and P_{DB}. The unknown first derivatives can be seen as the sensitivities of the corrosion rate to the change in temperature and pressure respectively for a given set of conditions. These are not generally known in advance and will change with conditions. For example the sensitivity of the corrosion rate to a change in temperature is different at low temperature ($20^oC<T<50^oC$) from what it is at higher temperature ($60^oC<T<90^oC$) and it all depends on pH, etc. To account for all this complexity and to correctly calculate the sensitivities of the corrosion rate to various parameters a trusted corrosion model needs to be employed here.

The larger value of the overall dissimilarity measure ΔCR, the more different is the problem case compared to the particular database case. In order to include all parameters that need to be considered and to avoid fortuitous cancellation of positive and negative differences, the absolute values of the various parameter contributions to the dissimilarity are used and all values are added together to represent the overall dissimilarity of the problem case compared to the database case:

$$(7)\quad \Delta CR = \left|\Delta CR_T\right| + \left|\Delta CR_P\right| + \left|\Delta CR_{pH}\right| + \left|\Delta CR_{pCO_2}\right| ..$$

An example of the procedure for comparing two database cases for similarity to a problem case based on two parameters (temperature and pressure) is illustrated in Figure 4. Two cases DB_1 and DB_2 have values for pressure and temperature and corresponding corrosion rates recorded in the database which are denoted in Figure 4 with dark circles. For each of them dissimilarity measure can be calculated by using equation (4) above as:

$$(8)\quad \Delta CR_1 = \Delta CR_{T_1} + \Delta CR_{P_1} = \left|CR\left(T_{DB_1}, P_{DB_1}\right) - CR\left(T_{Pr}, P_{DB_1}\right)\right| + \left|CR\left(T_{DB_1}, P_{Pr}\right) - CR\left(T_{DB_1}, P_{DB_1}\right)\right|$$

$$(9)\quad \Delta CR_2 = \Delta CR_{T_2} + \Delta CR_{P_2} = \left|CR\left(T_{DB_2}, P_{DB_2}\right) - CR\left(T_{Pr}, P_{DB_2}\right)\right| + \left|CR\left(T_{DB_2}, P_{Pr}\right) - CR\left(T_{DB_2}, P_{DB_2}\right)\right|$$

The one with the smaller ΔCR is judged to be more similar to the problem case.

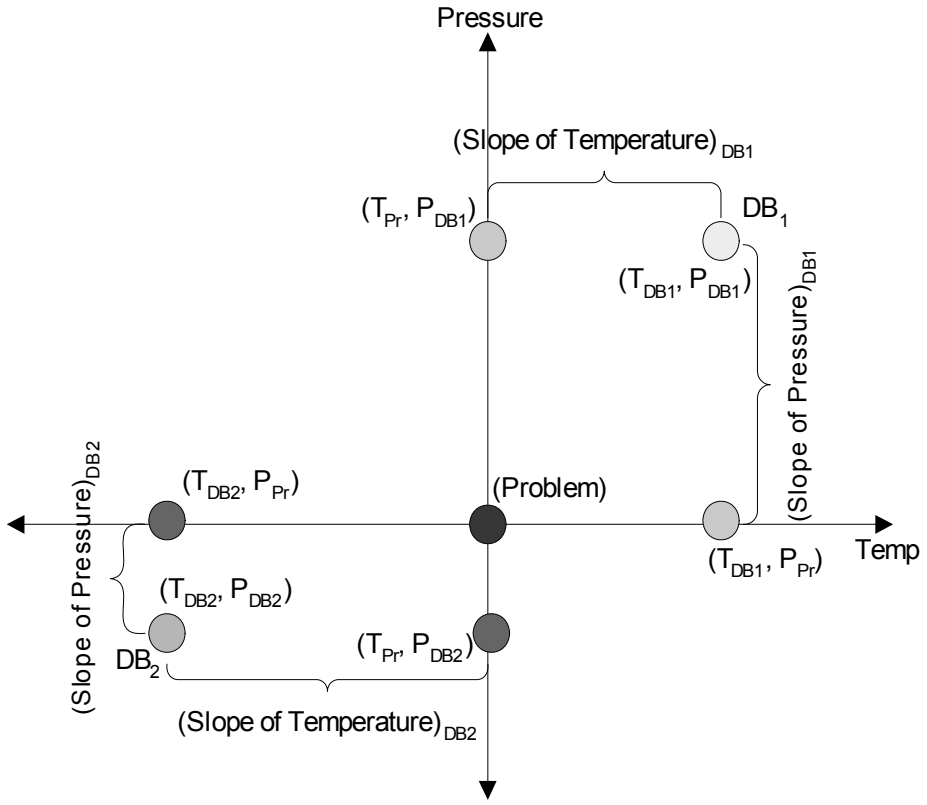


Figure 4: Similarity of two cases and the problem.

Case Revision

Even if the most similar case is found in the database there can still be a substantial difference between this case and the problem case in terms of the key parameters that define corrosion. Therefore the actual corrosion rate from the database can be corrected to compensate for this difference and get a more accurate estimate of the corrosion rate. Taylor series, equation (2) can be used again and this time to improve accuracy, one can retain the higher order terms. For example a second order scheme would read:

$$\begin{aligned} CR\left(T_{Pr},P_{Pr}\right) = CR\left(T_{DB},P_{DB}\right) &+ \left[\left.\frac{\partial CR}{\partial T}\right|_{DB}\left(T_{Pr}-T_{DB}\right)+\left.\frac{\partial CR}{\partial P}\right|_{DB}\left(P_{Pr}-P_{DB}\right)\right] \\ &+\frac{1}{2!}\left[\left.\frac{\partial^2 CR}{\partial T^2}\right|_{DB}\left(T_{Pr}-T_{DB}\right)^2+2\left.\frac{\partial^2 CR}{\partial T\partial P}\right|_{DB}\left(T_{Pr}-T_{DB}\right)\left(P_{Pr}-P_{DB}\right)+\left.\frac{\partial^2 CR}{\partial P^2}\right|_{DB}\left(P_{Pr}-P_{DB}\right)^2\right]+\varepsilon \end{aligned} \tag{10}$$

The unknown second derivatives can be calculated using a trusted corrosion model similarly as shown above for the first derivatives.

Another way of looking at this step is: the estimated corrosion rate is based on the value recorded in the database for the most similar case, while the adjustment of this value is done by the corrosion model to match the problem case parameters more accurately.

IMPLEMENTATION AND TESTING

This section describes the implementation of the prototype for the CBR-TS model and its testing.

Case-base model implementation

Implementation of the case-base model required the development of a method to store the cases and effectively search and retrieve them. Case-based reasoning also requires a hierarchical structure based on priorities of factors to be considered during case search and retrieval. This predefined hierarchical structure of the case-base also increases the search efficiency and forms the basis for several search techniques. For the present implementation the importance / priority of corrosion parameters was determined through existing understanding of the corrosion process. Apart from the hierarchical structure formed through parameter prioritizing, substructures based on parameter ranges needed to be considered when building the case-base structure. These substructures needed to be considered as CBR was to be applied in a continuous domain. Taking all these factors and incorporating flexibility into the CBR model to enable easy changes to CBR factors and continuous domain ranges, into consideration, the case-base was built from a database model. The database structure provided the advantage of case storage according to database rules thereby incorporating data consistency and removal of redundant information. A single case was divided into multiple subparts by grouping related information. These groups of related information formed the individual tables of the database (Figure 5). These tables were then linked as a relational database[8] so that an entry in the database would form a case. The advantage of grouping information into different tables and then linking the tables is that duplication of data can be avoided. The tables implemented in MS Access are:

- **System Information**, which stores general information of the case such as supplier information, pipe location, pipe material and dimensional details;
- **Flow Information**, including production rates, water cuts etc.;
- **Corrosion Data**, which contains field corrosion rates and corrosion mechanism, inhibition etc.;
- **Fluid Chemistry**, with information about temperature, pressure of gases, density, pH etc.; and
- **Corrosion Product Film Composition**, with the chemical composition of the corrosion film.

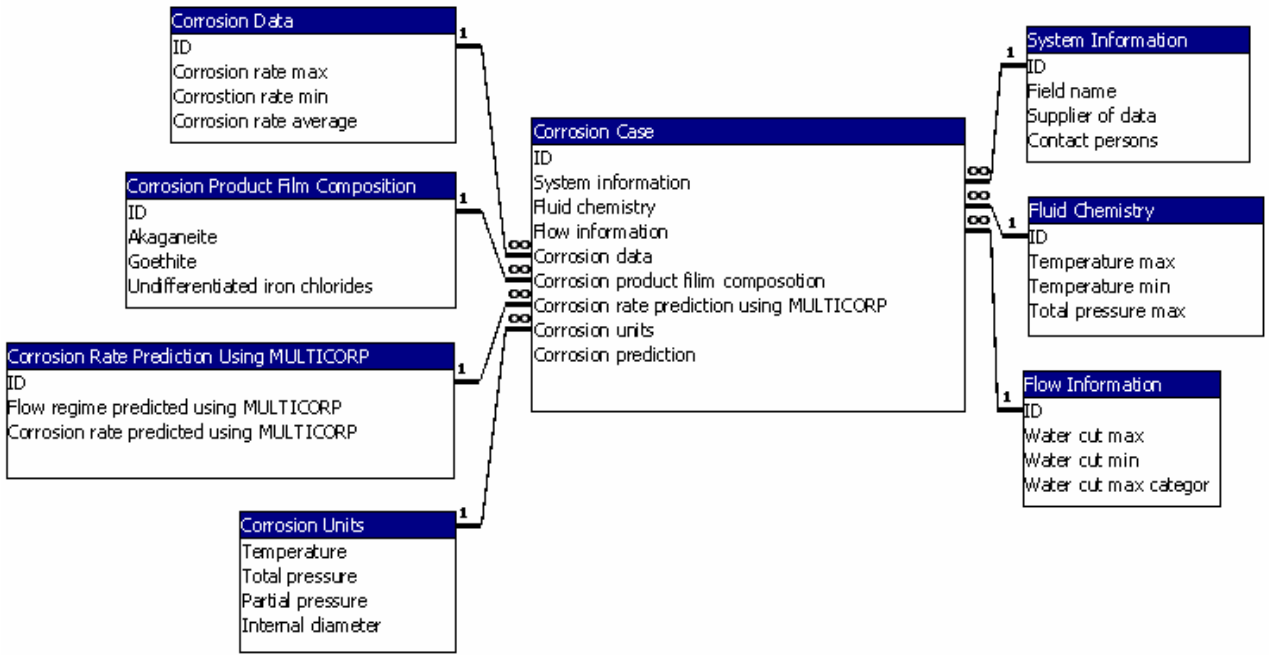


Figure 5. Case-base database representation

Parameter priority and intervals

The hierarchical structure of parameters and their similarity interval substructures were built into the model by reading parameter order and interval values from a text file which is defined by the database manager. Through the sequence of queries executed on the database a virtual tree structure of the retrieved cases is automatically formed during the model execution. This approach provides an extreme degree of flexibility into the CBR model advantageous for design and testing of the model.

Parameters used in the model are identified as discrete and continuous parameters as listed in Table 1. As shown, pipeline material is a discrete parameter with two possible values; other parameters are continuous, with three intervals for CO_2 pressure, pH and H_2S content, and two intervals for temperature.

Table 1. Parameters and respective ranges used for CBR-TS model execution

Priority	Parameter	Values		
	Discrete Parameters			
1	***Pipe Material***	*Stainless Steel* (SS304, SS306)	*Mild Steel* (10 - 18, 10 – 20, X – 65, Low Cr. Carbon Steel, High Cr. Carbon Steel, Alloy Steel, API - X42, L - 80)	
	Continuous Parameters			
2	***CO_2 Partial Pressure Max*** *(bar)*	0.0 - 1.0	1.0 - 5.0	> 5
3	***pH***	< 5.0	5.0 - 6.0	> 6.0
4	***H_2S Content*** *(ppm)*	0.0 - 10.0	10.0 - 1000.0	> 1000.0
5	***Temperature max (^oC)***	< 60.0	> 60.0	

Search algorithms were implemented to search through the case-base database to identify similar cases that fell into the ranges of the problem case values. Based on the search algorithm absence of data in certain situations was adjusted for to ensure continuity of the search process. The returned similar cases were then examined for the measure of dissimilarity and the most similar case identified to be used for the final corrosion rate estimation step. The list of cases and their dissimilarity measures are stored in a comma separated (CSV) for future examination (see for example Figure 6).

Link to Corrosion Model

Calculation of the case dissimilarity measures, ranking and final prediction was executed with the help of the corrosion model MULTICORP V3.0 (although any other trusted corrosion model would suffice). CBR-TS model has been implemented as a dynamic link library (DLL) that can be connected to any corrosion prediction model. Interactions between CBR-TS and the corrosion model as shown in Figure 7 are executed in the following order:

Step 1: The problem parameters are entered through corrosion model graphical user interface (GUI).

Step 2: The CBR-TS model (DLL) is initialized and problem information transferred to it.

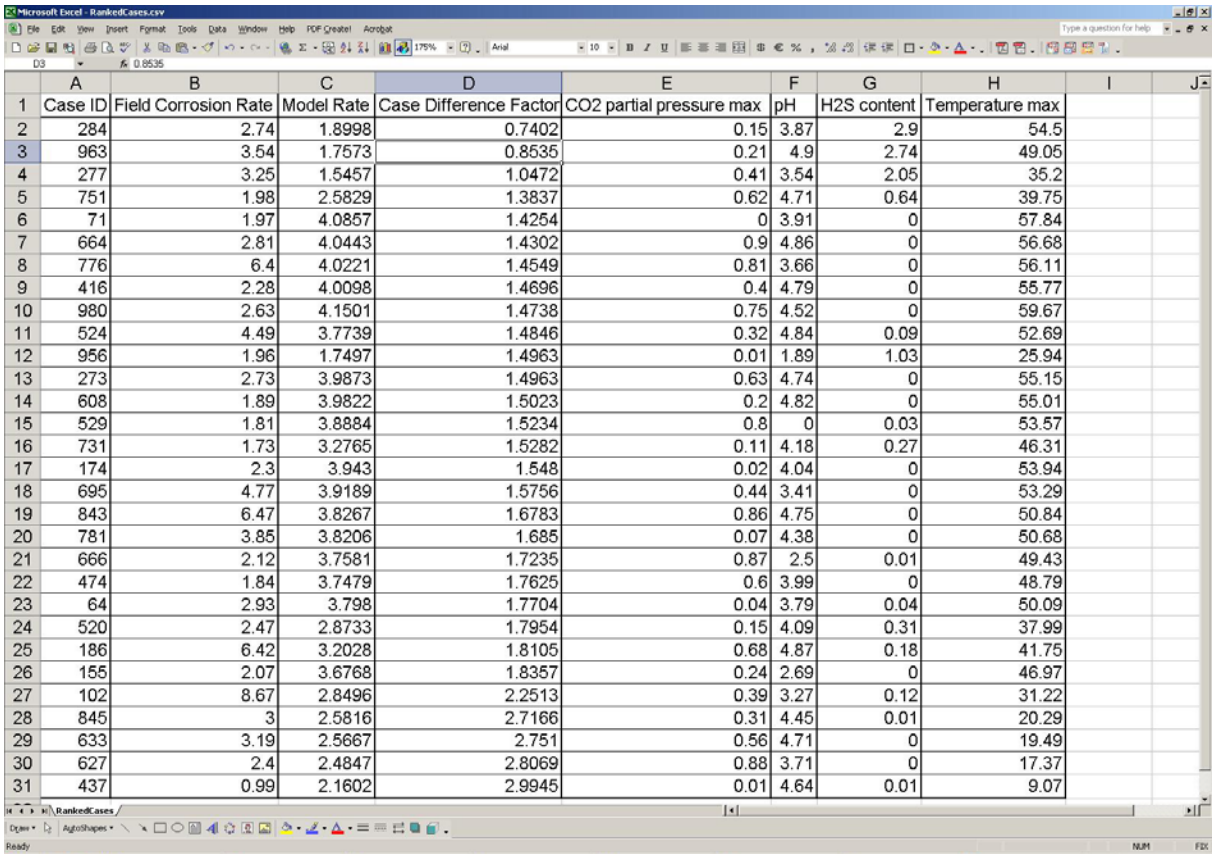
Step 3: Database search parameters and their respective ranges are read from parameter settings file.

Step 4: The database is queried using parameters and ranges obtained, to retrieve similar cases

Step 5: For each similar case, individual effects of each parameter on the corrosion rate of the problem are obtained by executing the corrosion model. These individual effects are summed up for each case to obtain a dissimilarity measure. Similar cases are ranked ascending based on dissimilarity measure and stored in a text (CSV) file (as shown in Figure 6).

Step 6: Model predictions of a corrosion rate for the most similar case and problem case are obtained by executing the corrosion model again. With these model predicted values and known actual corrosion rate of the most similar case corrosion rate is predicted for the problem.

Step 7: Problem data, the most similar case, and results (the correction factor and model calculated values) are stored in a text file to allow later retrieval (see Figure 8).



Case ID	Field Corrosion Rate	Model Rate	Case Difference Factor	CO2 partial pressure max	pH	H2S content	Temperature max
284	2.74	1.8998	0.7402	0.15	3.87	2.9	54.5
963	3.54	1.7573	0.8535	0.21	4.9	2.74	49.05
277	3.25	1.5457	1.0472	0.41	3.54	2.05	35.2
751	1.98	2.5829	1.3837	0.62	4.71	0.64	39.75
71	1.97	4.0857	1.4254	0	3.91	0	57.84
664	2.81	4.0443	1.4302	0.9	4.86	0	56.68
776	6.4	4.0221	1.4549	0.81	3.66	0	56.11
416	2.28	4.0098	1.4696	0.4	4.79	0	55.77
980	2.63	4.1501	1.4738	0.75	4.52	0	59.67
524	4.49	3.7739	1.4846	0.32	4.84	0.09	52.69
956	1.96	1.7497	1.4963	0.01	1.89	1.03	25.94
273	2.73	3.9873	1.4963	0.63	4.74	0	55.15
608	1.89	3.9822	1.5023	0.2	4.82	0	55.01
529	1.81	3.8884	1.5234	0.8	0	0.03	53.57
731	1.73	3.2765	1.5282	0.11	4.18	0.27	46.31
174	2.3	3.943	1.548	0.02	4.04	0	53.94
695	4.77	3.9189	1.5756	0.44	3.41	0	53.29
843	6.47	3.8267	1.6783	0.86	4.75	0	50.84
781	3.85	3.8206	1.685	0.07	4.38	0	50.68
666	2.12	3.7581	1.7235	0.87	2.5	0.01	49.43
474	1.84	3.7479	1.7625	0.6	3.99	0	48.79
64	2.93	3.798	1.7704	0.04	3.79	0.04	50.09
520	2.47	2.8733	1.7954	0.15	4.09	0.31	37.99
186	6.42	3.2028	1.8105	0.68	4.87	0.18	41.75
155	2.07	3.6768	1.8357	0.24	2.69	0	46.97
102	8.67	2.8496	2.2513	0.39	3.27	0.12	31.22
845	3	2.5816	2.7166	0.31	4.45	0.01	20.29
633	3.19	2.5667	2.751	0.56	4.71	0	19.49
627	2.4	2.4847	2.8069	0.88	3.71	0	17.37
437	0.99	2.1602	2.9945	0.01	4.64	0.01	9.07

Figure 6. Ranked cases with dissimilarity measure

Test cases

The proposed model has been evaluated on a set of field data available at Institute for Corrosion and Multiphase Technologies at Ohio University. The total of 82 cases were collected from industrial partners and analyzed in this work. However, the number of parameters (5) and their ranges (see Table 1) resulted in having 108 (=2x3x3x3x2) distinct cells in parameter value combinations for similarity considerations, which was insufficient for testing CBR portion of the model. In order to overcome this obstacle a hypothetical case-base with random generation of parameter values (according to parameter distributions of the small real field database), with a total of 1031 cases has been generated. Field corrosion rate data for this database was simulated by using a simple deWaard's model.[2]

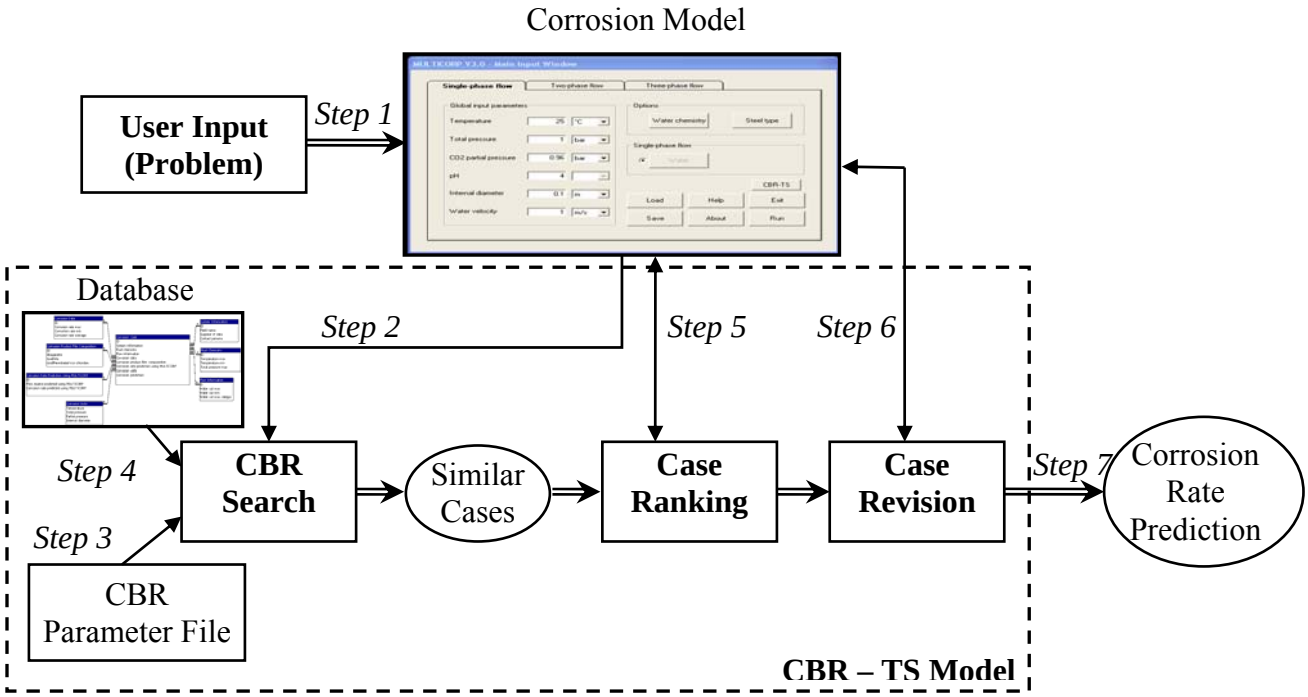


Figure 7: CBR-TS model execution block diagram

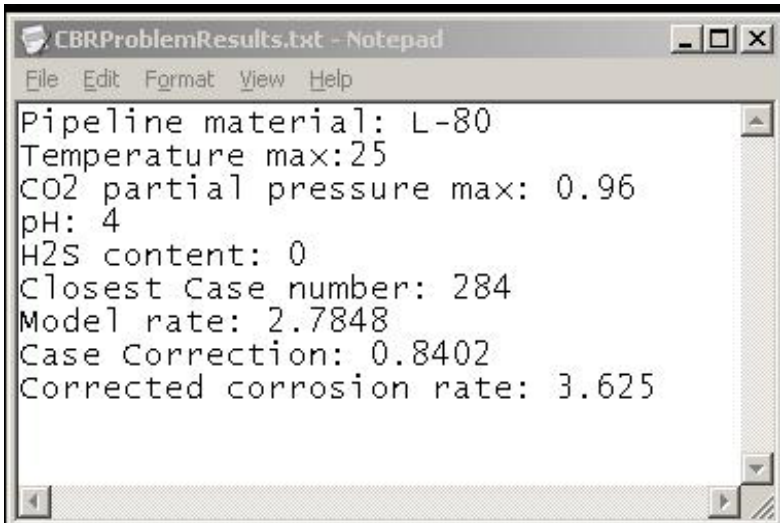


Figure 8. Result of CBR-TS corrosion rate prediction

CBR-TS model has been tested on both small field database and the hypothetical database. Tests were performed for the majority of possible parameter intervals, those with many cases and other with few cases. For example, problem (pipeline material=L-80, T=25, CO_2= 0.96, pH=4, H_2S=0) has returned 30 similar cases (Figure 6), CBR-TS ran 150 similarity calculations with the corrosion model in about 90 minutes on Pentium 4, 1.4 GHz processor and returned case 284 as the most similar case, calculated case correction as 0.8402 and reported corrected rate of 3.625 (Figure 8). Another problem instance (pipeline material=L-80, T=25, CO_2= 5.3, pH=4, H_2S=.8) returned only 2 similar cases which required only 10 similarity calculations that were performed within 2 minutes on the same computer.

The results obtained were satisfactory for both, case search and computational efficiency. However, since reliability of field data could not be guaranteed, extensive testing of the algorithm with more accurate experimental data is required. More reliable experimental data will also enable the testing of CBR portion of the model by verifying one of the known cases against the rest of case-base, which will allow the conclusion about the model robustness.

CONCLUSIONS

The paper has provided an overview of a new hybrid CBR-TS model for corrosion prediction. The model is designed to mimic the approach of experienced field corrosion personnel and utilizes several disciplines: artificial intelligence, chemistry, electrochemistry, and programming. The procedure of the model is sound and includes selection of similar cases, their ranking to obtain the most similar case, and utilization of mechanistic corrosion model to help predict the corrosion rate for a new problem.

The CBR-TS model has been implemented into a dynamic library that can be linked with various corrosion models. The prototype execution has been successful demonstrated. Testing has been performed on a small real field database and on a larger hypothetical case-base.

REFERENCES

1 Y. M. Gunaltun, “Combining Research and Field Data for Corrosion Rate Prediction”, CORROSION/96, Paper No. 27, (Houston, TX: NACE International, 1996).

2 C. de Waard, U. Lotz, D. E. Milliams, “Predictive Model for CO_2 Corrosion Engineering in Wet Natural Gas Pipelines”, Corrosion, Vol. 47, No. 12, p. 976, 1991.

3 S. Nesic, “Key Issues Related to Modeling of Internal Corrosion of Oil and Gas Pipelines - A Review”, ICMT Technical Report, Institute of Corrosion and Multiphase Technologies, Ohio University, 2005.

4 R. Nyborg, “Overview of CO2 Corrosion Models for Wells and Pipelines”, CORROSION/2002, Paper No. 02233, (Houston, TX: NACE International, 2002).

5 S. Nesic, J. Cai, K. J. Lee, “A Multiphase Flow and Internal Corrosion Prediction Model For Mild Steel Pipelines”, CORROSION/2005, Paper No. 05556, (Houston, TX: NACE International, 2005)

6 I. Watson and F. Marir, “Case-Based Reasoning: A Review”, THE KNOWLEDGE ENGINEERING REVIEW/1994, Vol. 9, Issue No. 4.

7 J. Kolodner, “CaseBased Reasoning”, Morgan Kaufmann Publishers. (San Mateo, CA, 1993).

8 C. N. Prague, M. R. Irwin and J. Reardon, “Access 2003 Bible”, Wiley Pub. (Indianapolis, IN, 2004).