Artificial Immune Systems for Classification of Petroleum Well Drilling Operations

Adriane B.S. Serapião[1], José R.P. Mendes[2], and Kazuo Miura[3]

[1] UNESP/IGCE/DEMAC – C.P. 178 – CEP 13506-900 – Rio Claro (SP), Brazil
adriane@rc.unesp.br
[2] UNICAMP/FEM/DEP – C.P. 6122 – CEP 13081-970 – Campinas (SP), Brazil
jricardo@dep.fem.unicamp.br
[3] PETROBRAS, E&P-SSE/UN-BC/ST/EP – Macaé (RJ), Brazil
kazuo.miura@petrobras.com.br

Abstract. This paper presents two approaches of Artificial Immune System for Pattern Recognition (CLONALG and Parallel AIRS2) to classify automatically the well drilling operation stages. The classification is carried out through the analysis of some mud-logging parameters. In order to validate the performance of AIS techniques, the results were compared with others classification methods: neural network, support vector machine and lazy learning.

Keywords: Petroleum Engineering, mud-logging, artificial immune system, classification task.

1 Introduction

Offshore petroleum well drilling is an expensive, complex and time-consuming operation and it demands a high qualification level from the drilling executors. One of the trends of the oil industry is the application of real-time measurements and optimization of production operations with the purpose of guaranteeing a safe and effective/low cost drilling execution. The concept of *digital fields* has been widely used in current works to denote continuous optimization of production [1]. This trend has also been seen in drilling, as real-time measurements and control are as well gaining attention in this particular area. In the last two decades, the technological advances in drilling techniques have notably contributed to the lowering of costs and to the expansion of exploration areas.

Technological progress in the petroleum engineering area was partly motivated by the evolution in instrumentation techniques, which affected not only the exploration segment but also the production one. As a result of the increasing instrumentation level, today, there is a lot of data being measured and recorded. But the techniques of data interpretation and evaluation have not developed at the same speed, and there is a lack of tools able to make an efficient use of all the data and information available.

This work presents the development of a system that intends to make better use of the information collected by mud-logging techniques during well drilling operations. The mud-logging techniques collected a great amount of data, and these data nowadays, are being used in a superficial way. The proposed system aims to take advantage of some of the information potential that is still not being used.

L.N. de Castro, F.J. Von Zuben, and H. Knidel (Eds.): ICARIS 2007, LNCS 4628, pp. 47–58, 2007.

The proposed methodology is able to generate a precise report of the execution stages during an operation through the interpretation of mud-logging data. There are two possible applications. The first one is related to the performance analysis and normality investigations. In this sense, this tool could be used to carry out the latter analysis of the time spent drilling each well in a field and to investigate how much of the total operation time each stage consumed and based on this statistics to plan the drilling of other wells. The second one is related to the production of an on-line logging of the executed stages. The methodology could be used on-line in the rig, so the system would be able to produce a report of the execution stages, and this report will present the same time precision as that of the mud-logging data.

The main idea of this work is that there is a great amount of information that has not been properly used, and this information could be used to provide a process feedback and to produce performance enhancements. There are initiatives of development of automatic monitoring systems in other areas like the work presented by Yue *et al.* [2] in Mining Engineering.

Information concerning individual drilling performance can also be used to build benchmarking analysis. In this sense, a petrol company could use this information to compare the performance of different divisions. On a minor scale, the company could compare performance of rented rigs and identify weak points as part of ongoing improvement process. The results produced by an automatic classification system may help in the design of new wells. The information about the time spent to execute a determined stage could be used for planning new wells in the same region providing cost estimates.

Artificial Immune Systems (AIS) are a new class of algorithms inspired by how the immune system recognizes attacks and remembers intruders [3]. The recognition and learning capabilities of the natural immune system have been an inspiration for researchers developing algorithms for a wide range of application. In this paper we are interested in applicability of artificial immune systems for real world data mining, and classification is one of the most important mining tasks, so we focus on the Clonal Selection Algorithm (CLONALG) and on the Artificial Immune Recognition System (AIRS) algorithm for that task.

CLONALG was proposed in 2000 [4] and is based on the clonal selection principle, which is used by the immune system to describe the basic features of an immune response to an antigenic stimulus. It establishes the idea that only those cells that recognize the antigens proliferate, thus being selected against those that do not. The selected cells are subject to an affinity maturation process, which improves their affinity to the selective antigens. The computational implementation of the clonal selection algorithm takes into account the affinity maturation of the immune response [5].

AIRS was introduced by in 2001 as one of the first immune systems approaches to classification. It is a supervised learning paradigm based on the principles of resource-limited artificial immune systems [6], [7]. In 2002 Watkins and Timmis [8] suggest improvements to AIRS algorithm that are capable of maintaining classification accuracy, whilst improving performance in terms of computational costs and an increase in the data reduction capabilities of the algorithm. This algorithm is here named AIRS2. A new version for a parallel AIRS2 was present in 2004 to explore ways of exploiting parallelism inherent in an artificial immune system for decreased overall runtime [9]. This algorithm was used in the present paper.

The classification results using those two immune techniques were compared with classification elaborated by a Petroleum Engineering expert [10] and with others automated methods in solving of drilling operation stages identification problem. These methods are neural networks [11], Support Vector Machines (SVM) [12] and Locally Weighted Learning (LWL) [13].

2 Mud-Logging System

During the petroleum well drilling operation many mechanical and hydraulic parameters are measured and monitored in order to perform the drilling in a safe and optimized manner. There are many systems that work together in a rig to accomplish this task. One of these systems is called a mud-logging system and it is responsible for measuring and monitoring a set of mechanical and geological parameters.

Mud-logging system techniques were introduced in Brazil in the 80's. At that time, only a reduced number of parameters were monitored. Since the 80s, with the developments in instrumentation techniques, the number of measured parameters has increased and the use of mud-logging systems became a common practice in the oil industry.

Another aspect that contributed to the progress of mud-logging techniques in Brazil was the development of deep and ultra-deep water drilling technologies. The deep and ultra-deep-water environments require a more controlled drilling operation [14]. Any failure or inattention may cause great human and economic losses. In order to have a more controlled process, the information supplier systems needed to be improved. In this context, the mud-logging systems were enhanced to become an important information supplier system.

Nowadays, mud-logging systems have two distinctive dimensions, the first one is responsible for collecting and analyzing formation samples (shale-shaker samples), and the second one is responsible for measuring and monitoring mechanical parameters related to the drilling operation. Considering only the second dimension, the mud-logging system could be characterized as a complete instrumentation system.

To accomplish its mission, the mud-logging systems rely on a wide range of sensors distributed in the rig operative systems. One important characteristic of this technique is that there is no sensor inside the well, and all measurements are taken on the rig. The data collect by the sensors are sent to a central computer system, where the data are processed and displayed in real time through screens installed in the mud-logging cabin and in the company-man office. The checking of the parameter evolution is carried out using the monitors; the system not only permits the selection of the displayed parameters but also the selection of their presentation appearance (numbers or graphics). Throughout the whole drilling operation, there is a worker watching the parameters for any kind of abnormality. If an observed parameter presents an unusual behavior, the worker has to immediately communicate this to the driller that will carry out the appropriate procedures to solve the problem. In fact, the system permits the programming of alarms that will sound in the mud-logging cabin, alerting the mud-logging worker, always when the value of the observed parameter is not within the programmed range.

The number of observed parameters may vary according to the particular characteristic of the drilling operation. The most common measured parameters are: Well Depth (Depth), True Vertical Depth (TVD), Bit Depth, Rate of Penetration (ROP), Hook Height, Weight on Hook (WOH), Weight on Bit (WOB), Vertical Rig Displacement (Heave), Torque, Drillstring rotation per minute (RPM), Mud Pit Volume, Pump Pressure, Choke Line Pressure, Pump Strokes per minute (SPM), Mud Flow, Total Gas, Gas Concentration Distribution, H2S concentration, Mud Weight in/out, Drilling Fluid Resistivity, Drilling Fluid Temperature, Flow Line, LAG Time, and Stand Length.

It is important to mention that just some of the listed parameters are really measured using sensors. Some of them are calculated using the measured parameters. The WOB, for instance, is a calculated parameter. It is calculated using the WOH (a measured parameter) and the knowledge of the weight of drill string elements.

The mud-logging monitoring services are generally provided by a specialized company that, at the end of the drilling operation, makes a report relating the occurrences associated to the completed operation. During the drilling monitoring, a huge amount of data is generated, and due to difficulties of data storage, the data are summarized to make up smaller files. The common practice is to reduce measurements made on a second basis to measurements made on a minute basis. Although it solves the problem of the files volume, on the other hand it represents the loss of a large amount of information. There are some events that may occur and last only a few seconds, like the drag occurrence in tripping out. When the data is summarized, the information about the drag occurrence is partially lost.

Considering all the measured parameters, it can be noted that the parameters related to the gas invasion in the well (Mud Pit Volume and Total Gas) are used more often than the others, it indicates that there is still a great information potential that has not been properly used.

Another important question related to the mud-logging system is the redundancy in parameter measuring. Besides hook height, other parameters have been measured by more than one instrument system. It is common to find rigs where the same parameter is being measured by the mud-logging company, by the MWD company and by the rig itself. And it is not rare to observe cases where the three measurements taken do not present the same absolute value. This behavior has caused some questioning about the future of mud-logging systems.

The general tendency is that more modern rigs will have a higher level of instruments on their working systems, and maybe in the future the rig will be in charge of measuring and monitoring all drilling parameters while the mud-logging services will be restricted to shale-shaker sample analyses.

3 Individual Stages Associated to the Drilling Operation for the Classification System

The drilling of petroleum well is not a continuous process made up of one single operation. If one looks at in a minor scale, it is possible to note that the petroleum well drilling operation is made up of a sequence of discrete events. These minor events comprised into the drilling operation will be called drilling operation stages. Six basics stages associated to the drilling operation were identified to build the proposed classification system. A brief description of each considered stage is presented below:

- **Rotary Drilling:** in this stage the drilling itself occurs, the bit really advances increasing well depth. The drill string is rotating and there is mud circulation. The drill string is not anchored to the rotary table causing a high hook weight level.
- **Rotary Reaming:** in this stage despite the high hook weight level, mud circulation and drillstring rotation, the bit does not advance increasing the final well depth. In this situation, there is a back-reaming of an already drilled well section.
- **Oriented Drilling ("Sliding Drilling"):** in this stage, the bit really advances increasing the well depth. The difference here is that the drillstring is not rotating and the drilling occurs due to the action of the downhole motor. There is mud circulation and a high hook weight level.
- **Back-reaming or Tool adjusting:** in this stage, the bit does not advance increasing the final well depth. There is circulation and a high hook weight level. This condition indicates that back-reaming is being carried out or that the tool-face of the downhole tool is being adjusted.
- **Tripping:** this stage corresponds to the addiction of a new section to the drillstring. The drillstring is anchored causing a low hook weight level. The drill string dos not rotate.
- **Circulating:** in this stage there is no gain in the well depth. It is characterized by fluid circulation, a high hook weight level and a moderated rotation of the drillstring.

These six stages represent a first effort to individualize the basic components of a drilling operation. The stages were detailed considering the drilling phases with mud return to the surface. The drilling technology considered was the drilling using mud motor and bent housing. This classification may not be satisfactory for the initial drilling phases and for special operations, such as fishing, in the well. In the same way, if other drilling technologies are considered, like the rotary steerable systems, small adjustments in the definition of the stages will be required. For instance, when using rotary steerable systems, it makes no sense to make a distinction between rotary drilling and oriented drilling stages as they were defined in this work, because these systems are supposed to drill all the time using drillstring rotation.

4 Artificial Immune Systems

The immune system guards our bodies against infections due to the attacks of antigens. The natural immune system offers two lines of defense, the innate and adaptive immune system. The innate immune system consists of cells that can neutralize a predefined set of attackers, or 'antigens', without requiring previous exposure to them. The antigen can be an intruder or part of cells or molecules of the organism itself. This part of the immune system is not normally modeled by AIS systems.

Vertebrates possess an adaptive immune system that can learn to recognize, eliminate and remember specific new antigens. This is accomplished by a form of natural selection. The adaptive immune response in biological systems is based on two kinds of lymphocytes (or self-cells) in the body: T-cells, so named because they originate in the thymus gland, and B-cells, which originate in bone marrow [3].

The major responsibility of the T-cells and B-cells is the secretion of the receptors called the antibodies (*Ab*) as a response to the antigens that enter the body (*Ag*) (nonself-cells). The role of these receptors on the surface of the lymphocytes is to recognize and bind the antigen. An individual T-cell or B-cell responds like a pattern matcher - the closer the antigen on a presenting cell is to the pattern that a T-cell or B-cell recognizes, the stronger the *affinity* of that T-cell or B-cell for the antigen. T-cells are sometimes called helper T-cells because in nature, although the B-cells are the immune response mechanism that multiplies and mutates to adapt to an invader, it is only when a T-cell and B-cell respond together to an antigen that the B-cell is able to begin cloning itself and mutating to adjust to the current antigen ('clonal expansion' or 'clonal selection') [15].

Once a B cell is sufficiently stimulated though close affinity to a presented antigen, it rapidly produces clones of itself. At the same time, it produces mutations at particular sites in its gene which enable the new cells to match the antigen more closely. There is a very rapid *proliferation* of immune cells, successive generations of which are better and better matches for the antigens of the invading pathogen. B cells which are not stimulated because they do not match any antigens in the body eventually die [16].

The immediate reaction of the innate and adaptive immune system cells is called the primary immune response. A selection of the activated lymphocytes is turned into sleeper memory cells that can be activated again if a new intrusion occurs of the same antigen, resulting in a quicker response. This is called the secondary immune response. Interestingly, the secondary response is not only triggered by the re-introduction of the same antigens, but also by infection with new antigens that are similar to previously seen antigens. That is why we say that the immune memory is *associative*.

Artificial Immune System (AIS) are inspired in many aspects of the natural immune systems, such as adaptivity, associative memory, self/non-self discrimination, competition, clonal selection, affinity maturation, memory cell retention, mutation and so on. These artificial immune system algorithms (also known as immunocomputing algorithms) have been applied to a wide range of problems such as biological modeling, computer network security, intrusion detection, robot navigation, job shop scheduling, clustering and classification (pattern recognition). We are interested in this last kind of application for our problem of classifying the well drilling stages. We have considered the two most known classification algorithms based on immune systems to carry out this task: CLONALG and AIRS2.

4.1 Clonal Selection Algorithm (CLONALG)

The clonal selection algorithm, CSA, was first proposed by de Castro and Von Zuben in [4] and was later enhanced in their 2001 paper [5] and named CLONALG. It uses the clonal selection principle to explain the basic features of an adaptive immune response to an antigenic stimulus. It establishes the idea that only those cells that recognize the antigens are selected to proliferate. The selected cells are subject to an affinity maturation process, which improves their affinity to the selective antigens. The algorithm takes a population of antibodies and by repeated exposure to antigens, over a number of generations, develops a population more sensitive to the antigenic stimulus. The basic algorithm for pattern recognition is [5]:

1. Randomly generate an initial population of antibodies *Ab*. This is composed of two subsets *Abm* (memory population) and *Abr* (remaining population): $Ab = Abm \cup Abr\ (m + r = N)$.
2. Create a set of antigenic patterns *Ag*.
3. Randomly choose an antigen Ag_i from the population *Ag*.
4. For all the *N* antibodies in *Ab* calculate its affinity f_i to the antigen Ag_i using some affinity function (Hamming Distance).
5. The *n* selected antibodies will be cloned (reproduced) independently and proportionally to their antigenic affinities, generating a repertoire C_i **o**f clones: the higher the antigenic affinity, the higher the number of clones generated for each of the *n* selected antibodies.
6. The repertoire C_i is submitted to an affinity maturation process inversely proportional to the antigenic affinity, generating a population C_i^* of matured clones: the higher the affinity, the smaller the mutation rate.
7. Re-apply the affinity function f_i to each member of the population C_i^* and select the highest score as candidate memory cell *Abm*. If the affinity of this antibody with relation to Ag_i is greater than the current memory cell Abm_i, then the candidate becomes the new memory cell.
8. Remove those antibodies with low affinity in the population *Abr*. Finally, replace the *d* lowest affinity antibodies from *Abr*, with relation to Ag_i, by new randomly generated individuals.
9. Repeat steps 3-8 until all *M* antigens from *Ag* have been presented.

A *generation* is completed after performing the steps 3-9 above. The rate of clone production is decided using a ranking system. Mutation can be implemented in many ways, such as multi-point mutation, substring regeneration and simple substitution.

4.2 Parallel Artificial Immune Recognition System – Version 2 (Parallel AIRS2)

AIRS2 is a bone-marrow, clonal selection type of immune-inspired algorithm. AIRS2 resembles CLONALG in the sense that both algorithms are concerned with developing a set of memory cells that give a representation of the learned environment. AIRS2 also employs affinity maturation and somatic hypermutation schemes that are similar to what is found in CLONALG. AIRS2 has used population control mechanisms and has adopted use of an affinity threshold for some learning mechanisms.

AIRS2 is concerned with the discovery/development of a set of memory cells that can encapsulate the training data. Basically, this is done in a two-stage process of first evolving a candidate memory cell and then determining if this candidate cell should be added to the overall pool of memory cells [8]. This process can be outlined from [9] as follows:

1. *Initialization*: Create a random base called the memory cells pool.
2. *Clonal expansion*. Compare a training instance with all memory cells of the same class and find the memory cell with the best affinity (Euclidian distance) for the training instance. We will refer to this memory cell as mc_{match}.
3. *Affinity maturation*. Clone and mutate mc_{match} in proportion to its affinity to create a pool of abstract B-Cells.
4. *Metadynamics of B-Cells*. Calculate the affinity of each B-Cell with the training instance.
5. Allocate resources to each B-Cell based on its affinity.
6. Remove the weakest B-Cells (lowest affinity) until the number of resources returns to a preset limit.
7. *Cycle*. If the average affinity of the surviving B-Cells is above a certain level, continue to step 8. Else, clone and mutate these surviving B-Cells based on their affinity and return to step 4.
8. *Metadynamics of memory cells.* Choose the best B-Cell as a candidate memory cell (mc_{cand}).
9. If the affinity of mc_{cand} for the training instance is better than the affinity of mc_{match}, then add mc_{cand} to the memory cell pool. If, in addition to this, the affinity between mc_{cand} and mc_{match} is within a certain threshold, then remove mc_{match} from the memory cell pool.
10. *Cycle*. Repeat from step 2 until all training instances have been presented.

Once this training routine is complete, AIRS2 classifies instances using *k*-nearest neighbor (k-NN) with the developed set of memory cells.

Comparing with a data mining approach, AIRS2 is a cluster-based procedure to classification. It first learns the structure of the input space by mapping a codebook of cluster centers to it and then uses k-nearest neighbor on the cluster centers for classification. The attractive point of AIRS2 is its supervised procedure for discovering both the optimal number and position of the cluster centers.

Algorithmically, based on the above description, the parallel version of AIRS2 behaves in the following manner [9]:

a) Read in the training data at the root process.
b) Scatter the training data to the *np* processes.
c) Execute, on each process, steps 1 through 9 from the serial version of AIRS2 on the portion of the training data obtained.
d) Gather the developed memory cells from each process back to the root.
e) Merge the gathered memory cells into a single memory cell pool for classification.

5 Results

The classification problem consists in identifying the drilling operations described above as *Rotary Drilling* (RD), *Rotary Reaming* (RR), *Oriented Drilling ("Sliding Drilling")* (SD), *Back-reaming or Tool adjusting* (TA), *Tripping* (TR) and *Circulating* (CI).

In order to identify a given drilling stage in execution, the system needs some of the information monitored by the mud-logging system. This work uses: Bit Depth, Weight on Hook (WOH), Stand Pipe Pressure (SPP), Drillstring Rotation (RPM) and Weight on Bit (WOB) for this task, as in the previous works [11], [12].

Real records of mud-logging data consisting of 3784 samples of three days well drilling were used for the training and evaluation of the implemented immune classifier. A Petroleum Engineering expert classified previously these data [10]. When training AIS classifier, the whole data set (3784 samples) was randomly separated into two subsets: 75% as training set (2838 samples) and 25% as testing set (946 samples) after training. These sets were the same used in the others related classification methods for this problem [11], [12]. Table 1 shows the data distribution according to pre-defined classes for the training and test sets. The table clearly indicates the data imbalance issue among the classes, mainly for the Circulating (CI) and Tripping (TR) stages, which are the less usual operation in the drilling activity.

Table 1. Distribution of data per class in the training and test sets

Number of samples	**Drilling operations**						
	CI	**TR**	**TA**	**SD**	**RR**	**RD**	**Total**
Training Set	14	75	795	753	343	858	*2838*
Test Set	2	22	266	253	114	289	*946*

The application of CLONALG with 20 generations for the proposed task produced 400 incorrectly identified instances of the training set and 104 misclassified instances of the test set. The classification accuracy for training and test sets are 85.9% and 89.0%, respectively. Table 2 shows the correctness rate for the training and test samples for each evaluated method.

Parallel AIRS2's experiments are undertaken with the k-value for the k nearest neighbor approach is set to 7. The value for number of threads is 5. The learning evaluation of this approach has shown a reasonable performance, obtaining 2587 instances correctly classified (91.2%) for the training data and 879 instances (92.9%) for the test data. The performance of the Parallel AIRS2 is higher than of CLONALG.

A Multi-Layer Perceptron (MLP) neural network (NN) with backpropagation (BP) learning algorithm, which is widely used in numerous classification applications, has been investigated for this problem in our previous work [11] and its results are compared with the immune classifier systems CLONALG and Parallel AIRS2. MPL-BP has a better performance among all method, reaching an accuracy of 96.3% and 94.9% for the training and test sets, respectively.

Support Vector Machine (SVM) was used also to develop the automatic classification system of well drilling stages [12]. SVM correctly classified 2660 samples of the training set, reaching a reliability of 93.73%. For the testing set, 876 samples were well classified, with 92.6% of success.

Locally weighted learning (LWL) is a class of statistical learning techniques (*lazy learning*) that provides useful representations and training algorithms for learning about complex phenomena [13]. LWL uses locally-weighted training to combine training data, using a distance function to fit a surface to nearby points. It is used in conjunction with another classifier to perform classification rather than prediction.

Table 2. Correctness rate for each classification method

Method	Training Set	Test Set
MLP-BP	96.3%	94.9%
SVM	93.7%	92.6%
CLONALG	85.9%	89.0%
Parallel AIRS2	91.2%	92.9%
Lazy LWL	80.5%	81.3%

Table 3. Classification accuracy for each class in the training data

Method	Drilling operations					
	CI	TR	TA	SD	RR	RD
MLP-BP	100%	100%	97.5%	98.1%	89.2%	96.0%
SVM	100%	100%	96.6%	95.8%	83.1%	92.8%
CLONALG	0%	98.7%	85.0%	93.5%	45.5%	96.5%
Parallel AIRS2	57.1%	100%	90.1%	95.6%	69.7%	96.6%
Lazy LWL	0%	0%	91.7%	0%	91.5%	100%

Table 4. Classification accuracy for each class in the test data

Method	Drilling operations					
	CI	TR	TA	SD	RR	RD
MLP-BP	100%	100%	96.2%	98.4%	86.0%	93.8%
SVM	100%	100%	96.6%	94.1%	83.3%	90.7%
CLONALG	0%	100%	89.5%	96.0%	72.8%	97.2%
Parallel AIRS2	100%	100%	91.7%	97.6%	75.4%	98.3%
Lazy LWL	0%	0%	91.7%	0%	93.3%	100%

The four components that define LWL are: a distance metric, near neighbors, weighting function, and fitting the local model. In this application it is the technique with the worst result: precision of 80.5% for training set and 81.3% for test set.

In order to understand the difficulties of pattern discrimination of each method in the learning process, Tables 3 and 4 present the classification accuracy for each class related to each learning technique for both training and test data. Ten trial runs were performed for each method using a 10-fold cross-validation procedure.

It is important to mention that the circulating (CI), tripping (TR), rotating mode (consisted of rotary drilling (RD) and rotary reaming (RR) stages) and non-rotating mode (consisted of back-reaming or tool adjustment (TA) and sliding drilling (SD) stages) operations are linearly separable classes. However, RD and RR are non-linearly separable classes. The same for TA and SD classes.

Close examination of the Tables 3 and 4 revels that, as expected, for MLP-BP and SVM the accuracy on the non-linearly separable data set is less than the classification accuracy on the linearly separable dada set. The classification is well successful either for classes with a little amount of samples or a large one.

When CLONALG and Parallel AIRS2 algorithms use the imbalanced data set for training, antigens from the majority class may generate more memory cells than the ones from the minority class. If all the memory cells are represented in a high dimensional space, one minority class cell may be surrounded by many majority class cells; so taking votes (k-NN classification) from several memory cells closest to a test antigen may cause biased decisions. That explains the fact of CLONALG is unable to learn the CI class and Parallel AIRS2 has a low performance to this class. Nevertheless, even for complex mud-logging data sets CLONALG and Parallel AIRS2 algorithms are able to perform fairly well as a classifier.

LWL does not learn the minority classes and its behavior for SD class is unclear.

6 Conclusion

The classification systems presented can be used either to classify stored mud-logging data of a database of drilled wells or to classify mud-logging data on-line and online in a rig. Due to the detailed level regarding each executed stage provided by the classification systems, it can help to analyze the individual drilling performance of each well. Information about the total time spent on each stage combined with related economic costs can be used to assess the real cost reduction benefit caused by optimized drilling programs and introduction of new technologies.

The imbalanced real mud-logging data has a large impact on the classification performance of the AIS classifiers, since they can achieve high precision on predominant classes but very low correct classification on classes with a few samples, in contrast with the neural network and SVM, which recognize efficiently all patterns of the minority classes. The results suggest that Parallel AIRS2 could achieve a similar performance than MLP-BP and SVM do on data sets of others applications or problems with a better class's distribution.

This paper demonstrates that the development of a classification system for real multi-class problems using immune system inspired approaches is feasible.

Acknowledgements

The authors would like to thank the Brazilian Research Council (CNPq) for the financial support (Process CT-Petro/MCT/CNPq 504528/2004-1).

References

1. Unneland, T., Hauser, M.: Real-Time Asset Management: From Vision to Engagement–An Operator's Experience. In: Proc. SPE Annual Technical Conference and Exhibition, Dallas, USA (2005)
2. Yue, Z.Q., Lee, C.F., Law, K., Tham, L.G.: Automatic monitoring of rotary-percussive drilling for ground characterization – illustrated by a case example in Hong Kong. International Journal of Rock Mechanics & Mining Sciences 41, 573–612 (2003)

3. de Castro, L., Timmis, J.: Artificial immune systems: A new computational approach. Springer, London. UK (2002)
4. de Castro, L.N., Von Zuben, F.J.: The Clonal Selection Algorithm with Engineering Applications. In: Whitley, L.D., Goldberg, D.E., Cantú-Paz, E., Spector, L., Parmee, I.C., Beyer, H.-G. (eds.) Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '00), Workshop on Artificial Immune Systems and Their Applications, pp. 36–37. Morgan Kaufmann, Las Vegas, Nevada, USA (2000)
5. de Castro, L.N., Von Zuben, F.J.: Learning and optimization using the clonal selection principle. IEEE Transactions on Evolutionary Computation 6(3), 239–251 (2002)
6. Watkins, A., Boggess, L.: A new classifier based on resource limited Artificial Immune Systems. In: Proc. 2002 Congress on Evolutionary Computation (CEC2002), Honolulu, Hawaii. IEEE Press, New York (2002)
7. Watkins, A., Timmis, J., Boggess, L.: Artificial Immune Recognition System (AIRS): An Immune-Inspired Supervised Learning Algorithm. Genetic Programming and Evolvable Machines 5(3), 291–317 (2004)
8. Watkins, A., Timmis, J.: Artificial Immune Recognition System (AIRS): Revisions and Refinements. In: Timmis, J., Bentley, P.J. (eds.) Proc. of 1st International Conference on Artificial Immune Systems (ICARIS2002), pp. 173–181. University of Kent at Canterbury (2002)
9. Watkins, A., Timmis, J.: Exploiting Parallelism Inherent in AIRS, an Artificial Immune Classifier. In: Nicosia, G., Cutello, V., Bentley, P.J., Timmis, J. (eds.) ICARIS 2004. LNCS, vol. 3239, pp. 427–438. Springer, Heidelberg (2004)
10. Tavares, R.M., Mendes, J.R.P., Morooka, C.K., Plácido, J.C.R.: Automated Classification System for Petroleum Well Drilling using Mud-Logging Data. In: Proc. of 18th International Congress of Mechanical Engineer.Offshore & Petroleum and Engineering, Ouro Preto, Brazil (2005)
11. Serapião, A.B.S., Tavares, R.M., Mendes, J.R.P., Morooka, C.K: Classificação automática da operação de perfuração de poços de petróleo através de redes neurais. In: Proc. of VII Brazilian Symposium on Intelligent Automation (SBAI). São Luís-MA, Brazil (2005)
12. Serapião, A.B.S., Tavares, R.M., Mendes, J.R.P., Guilherme, I.R.: Classification of Petroleum Wells Drilling Operations Using Support Vectors Machine (SVM). In: Proc. of International Conference on Computational Intelligence for Modelling, Control and Automation (CIMCA'2006). IEEE Computer Society, Sydney, Australia (2006)
13. Atkeson, C.G., Moore, A.W., Schaal, S.: Locally weighted learning. Artificial Intelligence Review 11(1-5), 11–73 (1997)
14. Kyllingstad, A., Horpestad, J.L., Klakegg, S., Kristiansen, A., Aadnoy, B.S.: Factors Limiting the Quantitative Use of Mud-Logging Data. In: Proc. of the SPE Asia Pacific Oil and Gas Conference, Singapore (1993)
15. Ada, G.L., Nossal, G.J.V.: The Clonal Selection Theory. Scientific American 257(2), 50–57 (1987)
16. Berek, C., Ziegner, M.: The Maturation of the Immune Response. Imm. Today 14(8), 400–402 (1993)