Comparing two types of knowledge-intensive CBR for optimized oil well drilling

Samad Valipour Shokouhi[1], Agnar Aamodt[2], Pål Skalle[1] and Frode Sørmo[3]

[1] Department of Petroleum Technology (IPT)
[2] Department of Computer and Information Science (IDI)
Norwegian University of Science and Technology (NTNU)
NO-7491, Trondheim, Norway
[3] Verdande Technology AS
Stiklestadveien 1- Trondheim, Norway
valipour@ntnu.no, agnar.aamodt@idi.ntnu.no, pal.skalle@ntnu.no, frode@verdandetechnology.com

Abstract. This paper describes a new architecture for reasoning that combines case-based and model-based reasoning, referred to as knowledge intensive CBR (KiCBR). The case retrieval process is explained and compared through different reasoning approaches; plain CBR, and two forms of knowledge-intensive CBR. The mentioned methods are applied to problems in oil well drilling, a challenging domain. Knowledge-intensive methods in CBR will improve the case retrieval process. Our experiments show that one of the KiCBR methods, in which root causes of problems are included in the case description, has the highest accuracy compared to plain CBR and KiCBR without root causes included.

Keywords: Case-based reasoning, Knowledge-intensive CBR, Oil well drilling

1 Introduction

Case-based reasoning (CBR) is an approach to problem solving that recalls previous successful experiences. New problems are solved by retrieving related situations from cases solved in the past. To retrieve a similar case, two main steps are imperative; indexing of cases that describe problematic situations and determining the similarity among cases.

CBR has been widely applied in different domains such as law [1], medicine [2], music [3] and petroleum engineering [4, 5].

After drilling an oil well (hole), all the produced materials must be transported to the surface. This process is known as the hole cleaning process. Hole cleaning remains a major concern for oil well drilling operations because of unwanted repair time that adds more cost to the drilling operation. In this study hole cleaning problems from the petroleum engineering domain were chosen to be solved by CBR. A case's features hence describe the hole cleaning problem.

CBR has been integrated with other reasoning modalities to improve case retrieval. Model-based reasoning was first combined with CBR in CASEY, a medical application to diagnose heart failures [6]. Later, the CREEK framework for building knowledge-based systems that integrate CBR with model-based reasoning (MBR) was introduced [7]. Shokouhi[8] presents how to determine the root causes of poor hole cleaning episodes by means of KiCBR, applying a method that here will be referred to as KiCBR-1. In the research presented here a new approach referred to as KiCBR-2 is introduced and compared to both plain CBR and KiCBR-1. In KiCBR-2 the root cause of a drilling problem is part of the case's features. The CREEK framework was used as the basis for developing the integrated system.

The rest of the paper is structured as follows: In chapter 2 we explain the hole cleaning problem, related to the functionality of our system. Chapter 3 and 4 explains the difference between the various reasoning methods. In chapter 5 results from the study of the effect of adding causality to the cases in KiCBR is reported. The chapter presents how reasoning integrates with other modalities to enhance the reasoning process in oil well drilling. The last chapter summarizes and concludes the paper.

2 The hole cleaning problem

An oil well is drilled by rotating a rock bit attached to the bottom part of a drillstring (i.e. the combination of the drill pipes). Rock materials, produced intentionally by the rock bit and coming unintentionally from the hole wall as drilling progresses, must be removed from the drilled hole. A major function of the circulating system (i.e. drilling fluid being pumped through the hole) is to remove and transport rock materials. Rock materials are thus lifted to the surface by circulating a fluid down the drillstring that transmits drilling fluid, and up the annular space between the hole and the drill pipe [9]. The high cost of a drilling operation and the time dedicated to this operation call for a need to reduce unintended downtime as much as possible.

Due to the number of parameters influencing the hole cleaning operation and the complex mechanisms involved, the hole cleaning process has not yet been fully understood [10]. Hole cleaning is therefore a challenging task in oil well drilling. Sometimes the hole cleaning problem is not essential in itself but it initiates other crucial issues that may even cause the well, or parts of it, to be given up. Experience gained from different wells can make the problem more predictable, which is the main motivation for using CBR to help dealing with this type of problem.

3 Methodology

The CBR cycle [11] shows that after obtaining a problem description (in terms of a case's features), similar cases stored in a case base with their known solutions are retrieved. Retrieved cases propose a solution to unsolved case and after revising, the case can be retained and the case base strengthened. In order to make a decision, the most similar case is the desirable one to use. An important step is therefore to retrieve the correct previous case that can be used to facilitate the present problem. Improving and optimizing the retrieval process through two different approaches is the focus of this paper. After presenting the case structure and the similarity assessment method the following section will outline the three CBR approaches to be compared: plain CBR, KiCBR-1 and KiCBR-2.

3.1 Case structure

In the CBR approach, a problem is solved by recalling a previously solved case. Therefore, cases need to represent the problem reasonably and properly. Fig. 1 illustrates the case structure and the set of case features in our system.

Administrative Data	Wellbore Formation Characteristic	Variable Data
Operator Company Well Identification Oil Field Identifier Drilling Contractor….	Lithology Sandstone Siltstone … Geological Period	Interpreted Activity Drilling Mud circulating Tripping in Tripping out Connection Reaming ... Inferred Parameter Exposure time Openhole length ... Interpreted Event Packoff Took weight Tight spot Increased torque Torque vibration ...
Activity Before Case Occurrence	**Drilling Operational Data**	
Case Series Start Time Possible Case Start Time...	Well Geometry Parameter Target Depth Section TVD … Fluid Operational Parameter Mud Weight PV YP Water Activity Of Mud …. Drill String Parameter BHA Length Bit Run Number .. .	
Response Activity Description		
Case End Time Response Action…		
Conclusion		
Final Section Consequence Lesson Learned General Lesson Specific Lesson ….		

Fig. 1. Case structure

To identify the problem, all the relevant data and information were analyzed and the most descriptive features indexed. The case captures the description of a particular problematic situation, how the problem was solved and what experience was gained after implementing of the solution. There are informative and descriptive sections presented in the case structure. Informative sections, the first column, e.g. ‘Administrative Data’ and ‘Activity Before Case Occurrence’, are not used in the case matching process. This information describes where and when the problems happens. It helps us to figure out which well sections had likely problematic situations for further analysis and for building cases. Descriptive sections, the second and third column, are the ones that have been used in case matching routines.

In this study a case represents a hole cleaning problem. 35 cases were made and retained in the case base. A range of downtime incidents (i.e., unwanted time to fix the problem) was used to evaluate the performance wrt. predicting hole cleaning problems. The downtime problems were divided into two problem *groups*, each containing three problem *classes*. Group 1 represents hole cleaning quality during the drilling operation. Group 1 discriminates between the following three classes, each represented by a number of cases in the case base. There are 17 cases in the *insignificant downtime* class, 12 cases in the *significant downtime* class and 6 cases in the *gave up well section* class respectively.

A section is a part of the drilled hole, typically in the range of 500-2000 meters long. Whenever a section has been drilled to completion, a casing is placed into the hole. A casing is a large-diameter steel pipe lowered into the hole to keep the hole section stable. Hole cleaning problems can also lead to significant downtime while casing a poorly cleaned hole. It can be manageable if such problems are prevented through recognizing the level of the downtime during the hole cleaning process. Problems related to casing operations is labeled as group 2 problems. Group 2, 'Downtime While Casing Poorly Cleaned Hole', is also divided into the same three sub-classes as group 1. In Group 2 there are 3, 8 and 24 cases in the insignificant downtime, significant downtime and gave up well section class respectively.

In summary, as shown in Fig. 2, cases have been classified into two main groups; 'Downtime While Cleaning Hole' and 'Downtime While Casing Poorly Cleaned Hole'. These cases represent hole cleaning problems at different hazardous levels. Each of these groups was also divided into three sub-classes in terms of the severity of downtime; insignificant downtime, significant downtime and gave up well section.

35 cases were made and categorized with respect to their downtime. Each of the cases has two classes as output according to the legend, shown in Fig. 2. For instance, the case 1 has *insignificant downtime* and *gave up well section* in terms of downtime classes in group 1 and group 2. Three different reasoning methods have been tested against these categorized cases and results will be presented in chapter 4 and 5.

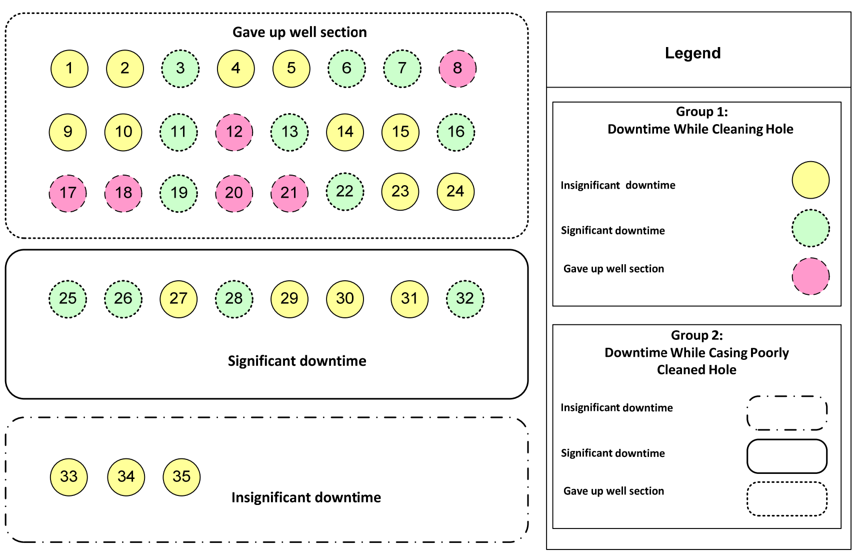

Fig. 2. Classification of 35 cases with respect to their downtime

3.2 Similarity assessment

A robust retrieval process requires an effective similarity assessment. Two different mechanisms are used to compute the values of similarity between a new problem case and a case in the case base. Linear similarity is used for those features that have numeric values. Semantic similarity, relying on concepts abstraction, is being used for direct or indirect match of symbolic feature values. The latter, indirect match, is used when the model based module is utilized.

The linear approach explicitly computes the values of similarity according to the minimum and maximum values of each concept. The maximum and minimum of each feature give an interval, and the values of the two cases are compared on this scale, giving a value of 0 if the difference between the values is the same as the difference between the minimum and the maximum, and a value of 1 if the values are the same.

For symbolic concepts:

$$sim(f_1, f_2) = \begin{cases} 1 & if\ f_1 = f_2 \\ 0 & if\ f_1 \neq f_2 \end{cases} \quad (1)$$

For linear concepts:

$$sim(f_1, f_2) = 1 - \left|\frac{f_1 - f_2}{Max - Min}\right| \quad (2)$$

In addition to these local measures that are defined for each feature in the case, a global similarity measure is computed as a weighted average of the local similarities. It is necessary to define weight of the features according to their relevance to the problem. Fig. 3 illustrates two-layer similarity measurement. Note that this example presents four features of each case. Feature A and B were considered as category 1.

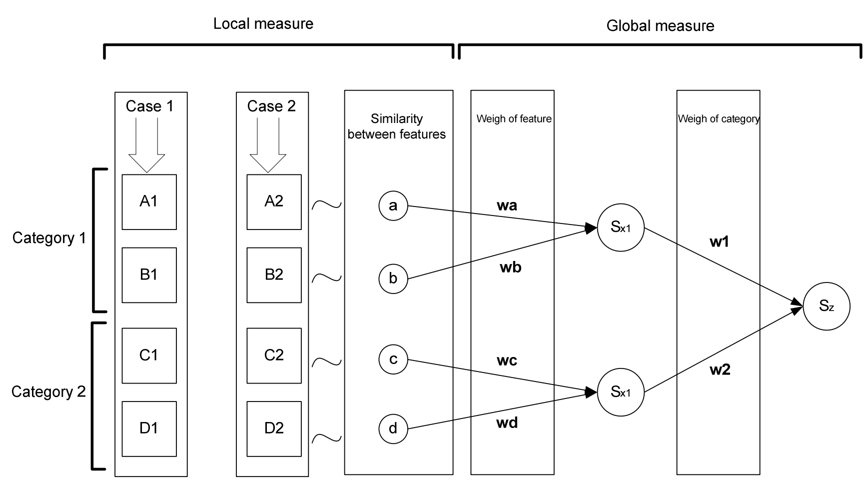

Fig. 3. Two-layer similarity measurement

Likewise, category 2 consisted of two features; C and D. Category 1 and category 2 are representative of descriptive sections, as shown in case structure.

In order to assess total similarity between two cases, similarity between features is determined by means of the linear or semantic measures. *w1* and *w2* show the effect of each category on final similarity. The following equations compute similarity of two cases, case 1 and case 2.

$$S_{x1} = (wa \times a + wb \times b)/(wa + wb)$$
$$S_{x2} = (wc \times c + wd \times d)/(wc + wd)$$
$$S_z = (S_{x1} \times w1 + S_{x2} \times w2)/(w1 + w2)$$

where a, b, c and d are representative of similarity between two features. Weights are shown by ‘w’ and S_z is the similarity between two cases.

4 Reasoning approaches

4.1 Plain CBR

CBR generally consists of four steps i.e., retrieve, reuse, revise, and retain. CBR is able to utilize the specific knowledge of previously experienced, concrete problem situations (cases). Central tasks that all case-based reasoning methods have to deal with are to identify the current problem situation, find a past case similar to the new one, use that case to suggest a solution to the current problem, evaluate the proposed solution, and update the system by learning from this experience [11].

The most important experience source in the oil well drilling domains, e.g. for handling hole cleaning problem, is past cases. Drillers usually reason with cases that happened in the past and try to find the most relevant and similar one. In this paper, a case base of 35 cases based on information from oil wells in the North Sea is used. All case features have been defined in collaboration with experienced engineers. In plain CBR, cases is the only source of knowledge, i.e. no additional model of general domain knowledge is utilized in addition.

Linear and direct symbolic similarity measurements are used for case matching within the plain CBR platform, explained in the previous chapter (for more information see [8]). Indirect symbolic similarity measurement will be part of the similarity assessment in the two integrated KiCBR approaches.
Basic similarity is computed by the following equation.

$$sim(C_{IN}, C_{RE}) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} sim(f_i, f_j) \times relevancefactor_{f_j}}{\sum_{j=1}^{m} relevancefactor_{f_j}} \quad (3)$$

C_{IN} and C_{RE} are the input and retrieved cases, n is the number of findings in C_{IN}, m is the number of findings in C_{RE}, f_i is the i^{th} finding in C_{IN}, f_j the j^{th} finding in C_{RE}, and $sim(f_1,f_2)$ is simply given as:

The relevance factor is a number that represents the weight of a feature for a stored case. The relevance factor of each feature was defined according to the feature’s importance for the case description.

4.2 Plain MBR

The model-based module used is a semantic net-based model of entities linked by relations. Each relation is labeled. The case features are all represented as entities in this model, and the model-based reasoner works by finding paths of causal relationships from the entities representing case findings in one case to entities representing case findings in the other. In order to determine legal paths, plausible inheritance is used. This method is a generalization of normal subclass inheritance that allows inheritance of relationships over other relation types than 'subclass of' relations. Plausible inheritance is governed by a set of rules declaring which relation-types can be inherited over which relation-types. In this paper, causal relationships are transitive, and any relationship can be inherited over 'subclass of ' relationships. This is defined so that for instance the path "A causes B causes C subclass of D causes E" is a legal path from A to E, but "A causes B caused by C" and "A subclass of B has subclass C" are not. For more information, see [12].

Assume there is a legal path from a finding of an unsolved case related to another finding of a solved case. Its strength is the product of the strengths of each relation connecting the two findings [7]:

$$Path\ strength = \prod_{i=1}^{n} relation\ strength_i \tag{4}$$

where n is the number of serial relations. Sometime there is more than one explanatory path between the two findings. The total explanation strength connecting the two findings is determined with Eq. (5).

$$Explanation\ strength = 1 - \prod_{i=1}^{m}(1 - path\ strength_i) \tag{5}$$

The Ontology is described at a symbolic level and the concepts in relation to the case features define the model-based system. The simplest model-based system is referred to as Plain MBR-1. Plain MBR-1 is used to retrieve the most similar cases without integrating it with case-based reasoning. In addition to Plain MBR-1, another type of model-based system will be introduced later in section 4.4. This is referred to as Plain MBR-2, and this system makes particular use of the specified entities that we call root causes.

A comparison between the plain CBR and the Plain MBR-1 and MBR-2 will be presented at the end of this chapter.

4.3 KiCBR-1

A KiCBR system achieves its reasoning power through the set of previous cases combined with some other source of knowledge about a certain domain [13]. The KiCBR-1 approach combines plain CBR with model-based knowledge in order to improve the local similarity measures. The CREEK system [7] is used for integration of cases, a concept ontology, and the model-based knowledge. An ontology and causal model for the drilling domain has been developed, in which all the entities are linked by binary relations, see Fig. 4.

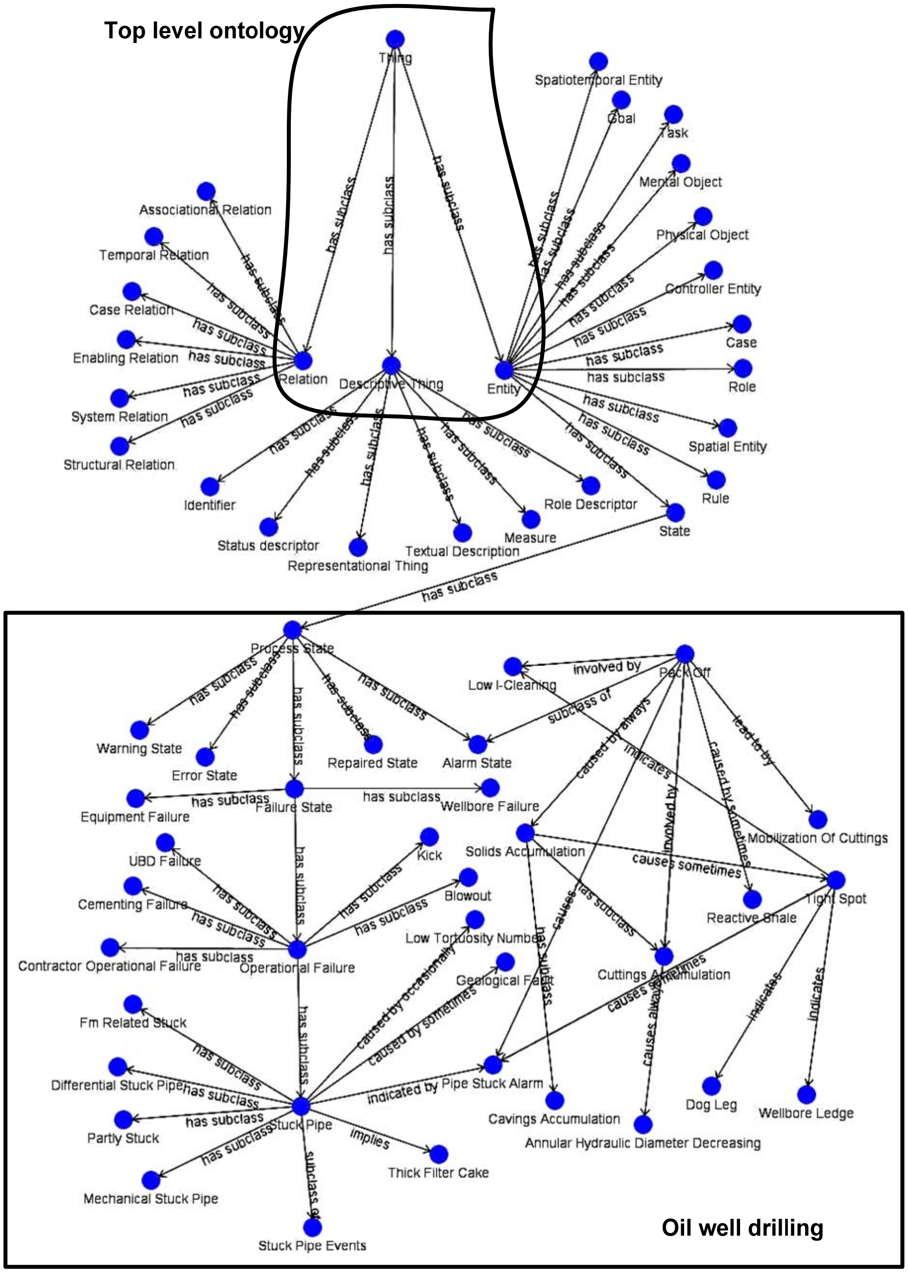

Fig. 4. Part of the Oil well drilling ontology including top level ontology

In our ontology, the most ubiquitous top-level is starting from ‘Thing’, and the three next-level concepts are Entity (i.e. a real world object), Descriptive Thing (i.e. any object in the world that is textually described) and Relation which relates existing entities and descriptive things. The top-level was, of course, integrated with the oil well drilling ontology, shown in the lower part of Fig. 4. This will cause the system and the case retrieval process to become less vulnerable for syntactical variations that do not reflect semantically differences.

In the KiCBR-1 method, each case in the case-base is expanded by adding as case features any entities in the model for which there exists a legal path from any existing feature to the entity. For instance if a case has the feature “Pack Off” the feature “Solids Accumulation” will be added to it as there exists a causal path from “Pack Off” to “Solids Accumulation”. The weight of this feature is based on the strength of the combined paths supporting it. The input case is expanded in the same way, which allows two cases with different symptoms of the same underlying problem to match partially.

4.4 KiCBR-2

In contrast to the KiCBR-1 method, while also using a case expansion method, the KiCBR-2 uses a more focused approach. The KiCBR-2 will only expand the cases with the specified entities of general knowledge that we call them root causes. By calculating the explanation strength supporting each entity representing a root cause based on the features in a case, the model-based reasoner will have a good indicator of the actual root cause for the situation represented by the case. The total explanation strength for each target entity (root cause) is determined with Eq. (6). This equation is a modification of Eq. (5) which considers weight of the legal path.

$$Explanation\ strength = 1 - \prod_{i=1}^{m}(1 - path\ strength_i \times weight) \qquad (6)$$

where m is the number of paths. The strength of the indicating entities was decided based on a survey among five experts to reduce subjectiveness of these values.

Fig. 5 illustrates the architecture of the KiCBR-2. Relevant concepts are categorized as root causes for the specified problem that has to be solved. It means instead of putting all the entities popped up from ‘Ki’, the determined root causes play roles of case’s features.

The root cause is determined starting out from three types of features that are directly represented in the cases: Direct *observations* – i.e. measurements, *inferred parameters* – i.e. values derived from observations, and interpreted *events* – i.e. particular concepts describing important states which require particular awareness or action. These entities are part of the ontology and the causal model links them (possibly through intermediate entities) to the entities representing the root causes. This means that there exists a number of paths supporting each root cause from the observations in the cases. For details, see [8].

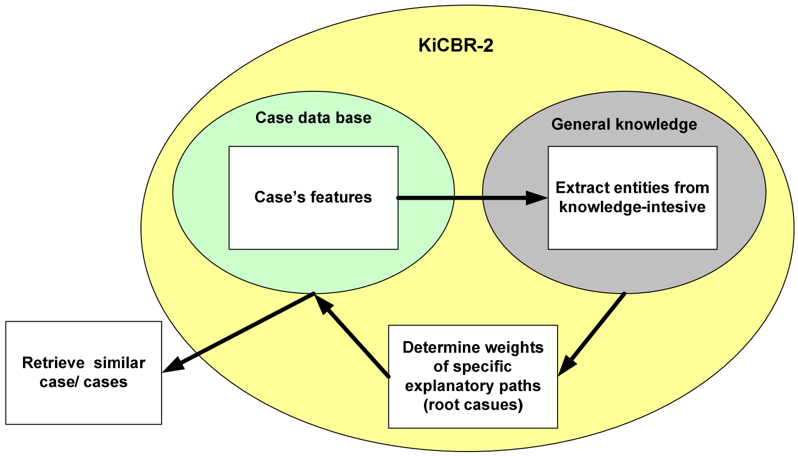

Fig. 5. Architecture of the KiCBR-2

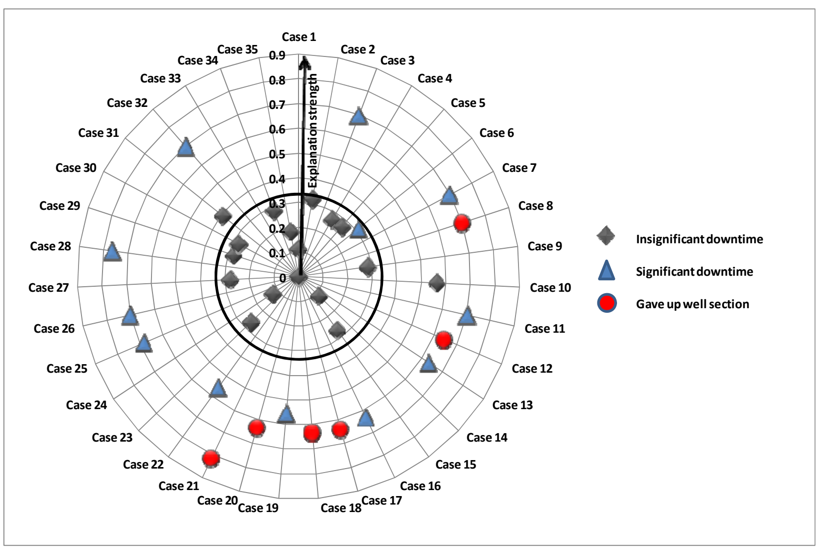

Fig. 6. Strength of the hole cleaning problem by Model-based

One such root cause is "Hole cleaning problem". Using Eq. (4) and Eq. (6), the support for this root cause is calculated for each case in the case-base. Fig. 6 depicts the explanation strength of the hole cleaning problem for all of the 35 cases, stored in our case base.

Results showed a good correlation between classification of cases and explanation strengths of the root cause. Cases with significant downtime or gave up well section class have higher explanation strength, in which the hole cleaning is a problem for drilling operation. For those cases with insignificant downtime, the explanation strength is lower than 35 % and located inside the circle, shown in Fig. 6. However the explanation strength was unable to distinguish between the two other classes; significant downtime and gave up well section. Five other root causes were identified and all of them follow this procedure. Firstly, the path strength is computed with Eq. (4). Afterwards, all the computed path strengths are combined with Eq. (6). Finally, the explanation strength of root causes complement cases' features to improve the case matching process.

Specific explanatory paths (root causes), as explained above, are also used for reasoning purpose (excluding case-based reasoning). It means that all inferred root causes define another model-based system, called Plain MBR-2.

Fig. 7 presents the accuracy of the case matching process by two different model-based reasoning methods compared to the plain CBR. It shows that the plain CBR retrieved the cases more effectively than the two others; plain MBR-1and MBR-2. All the methods retrieved almost the same number of similar cases in the *insignificant downtime* class. But for the *significant downtime* and *gave up well section* classes, each method had different results in terms of case matching. In the *insignificant downtime* and *gave up well section* classes, 17 % of the cases were retrieved correctly by the plain MBR-1.

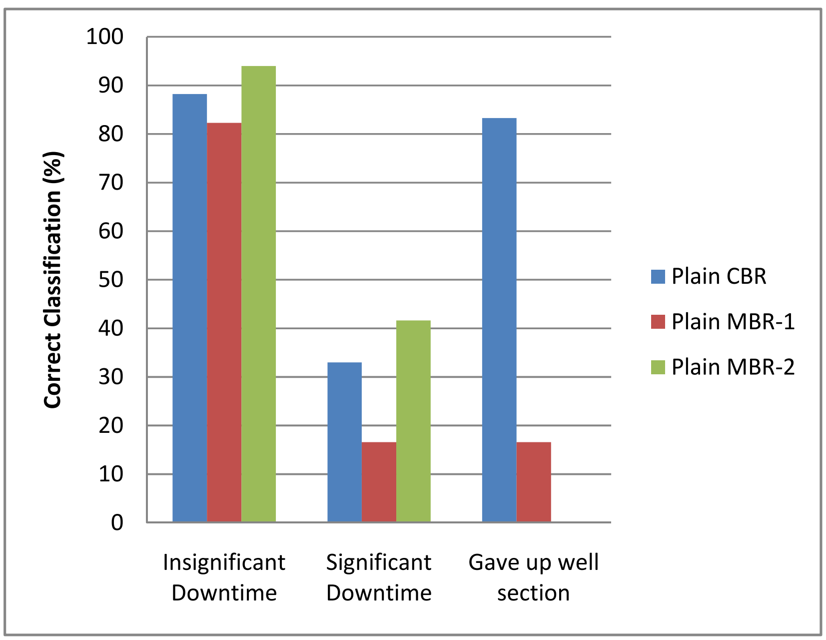

Fig. 7. Accuracy of case retrieval process by means of plain CBR and two model-based reasoning methods

The plain MBR-2 retrieved some cases correctly in the *insignificant downtime* and *significant downtime* classes while it was unable to retrieve any correct case in the *gave up well section* class. But the total retrieved cases by the plain MBR-2 is higher than the plain MBR-1.

As can be seen from Fig. 7, the number of correctly retrieved cases is higher by the plain CBR than any of the two plain MBR methods.

To take full advantage of plain CBR and model-based reasoning, the two different reasoning methods are combined and results are presented in chapter 5.

5 Results and discussion

Thirty five cases based on oil wells drilled in the North Sea were made and categorized based on the downtime occurring while drilling and while casing, as explained in chapter 3. Three reasoning approaches, the plain CBR, KiCBR-1 and KiCBR-2 were applied to these 35 cases.

To discuss the results adequately and clearly, results are concluded in two groups. The first and second groups are '*Downtime While Cleaning Hole*' and '*Downtime While Casing Poorly Cleaned Hole*', shown in Fig. 2.

Leave-one-out cross-validation was used, and the single most similar case was used to classify the case that was taken out.

Group 1 results: The three methods, plain CBR, KiCBR-1 and KiCBR-2 were applied with respect to classes in group 1. Fig. 8 presents the accuracy of case retrieval process for the different approaches. The number of correctly retrieved cases by the KiCBR-1 and KiCBR-2 was different compared to the plain CBR. In this study the plain CBR is a basis for comparison between CBR and integrated CBR approaches. Out of 17 cases in the *insignificant downtime* class; 15, 15, and 16 cases were correctly retrieved by the plain CBR, KiCBR-1 and KiCBR-2 respectively. For the second class, *insignificant downtime*, KiCBR-1 and KiCBR-2 improved case retrieval process from 4 correctly retrieved cases to 8 and 6. KICBR-1 doubled case matching accuracy compared to the plain CBR. However, the number of the correctly retrieved cases for the *gave up well section* class was reduced for KiCBR-1. For that class, the KiCBR-2 method is significantly better (using a t-test with 5% significance level) than the plain CBR results.

Fig. 9 summarizes the case matching capability of the three combined reasoning methods and two model-based reasoning methods applied to the group 1 in the hole cleaning problem. In this figure the total accuracy of case matching is presented for all the methods. As can be seen in Fig. 9, the two different model-based methods, Plain MBR-1 and Plain MBR-2, can poorly reason by itself. The total accuracy of the case matching capability by the plain MBR-1 and MBR-2 are 48 % and 60 % respectively. Since they are combined with the plain CBR, the reasoning results are improved significantly. The Combination of the plain MBR-1 and plain CBR made better reasoning than plain CBR and MBR-1 alone. It means that the KiCBR-1 had 74 % correctly retrieved cases instead of 68 % by the plain CBR.

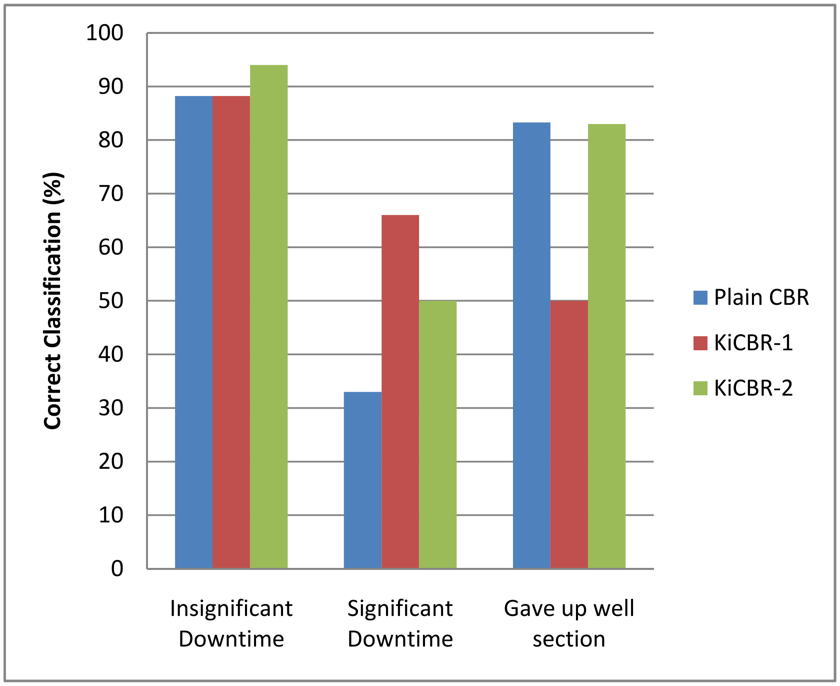

Fig. 8. Accuracy of case retrieval process by means of different reasoning methods

As shown in Fig. 9, there is an increasing trend from the plain CBR to KiCBR-2. The KiCBR-1 improves the ability of the case retrieval routine by calling all the dependant entities through the case's features. It means that introducing new entities to the case structure by using case's features will improve similarity assessment among cases. But absence of new entities in an unsolved case compared to a solved case may not improve the similarity in a good way. On the other hand, the KiCBR-2 will only expand the cases' features with the specified entities of general knowledge, called root causes. The root causes are good indicators of the situation represented by a case [8]. Therefore, the use of root causes can potentially provide higher accuracy in case matching process. The different methods show that the general knowledge can generally increase the accuracy of the case matching process. The main conclusion is that the KiCBR-2 can reason and accomplish the case retrieval process better than the two others. KiCBR-2 is able to boost the classification accuracy compared to the plain CBR and KiCBR-1. Accuracy was increased by almost 9 % by the KiCBR-2 method over the plain CBR method. However, KiCBR-1 also increased case matching results by 6 %. It should be noticed that 9 % is a big improvement in oil well drilling because it is very crucial to hit the exactly similar case and get the right solution for the problematic situation.

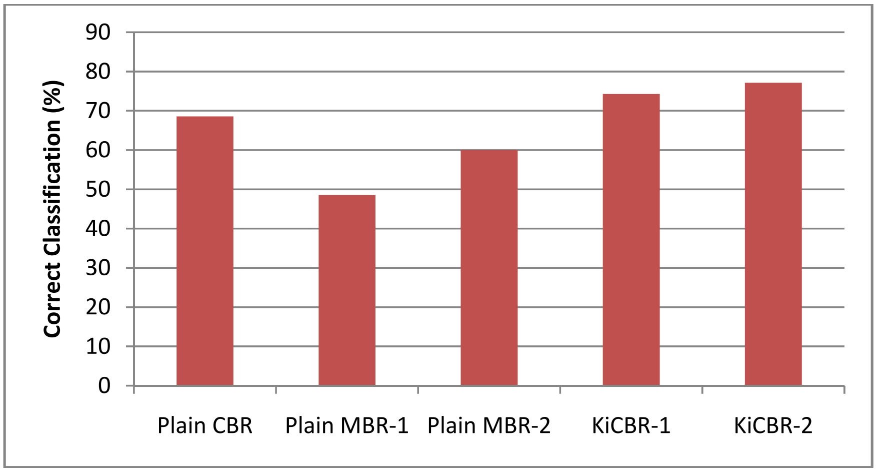

Fig. 9. Total accuracy of case matching capability by means of different reasoning approaches

Group 2 results: In the second set of experiments the same cases were used for on Group 2, '*Downtime While Casing Poorly Cleaned Hole*'. The results reveal that the KiCBR-1 had an influence on case matching assessment and increased the accuracy of the retrieval process by a small percentage. While the KiCBR-2 did not change the case matching compared to the plain CBR. Analysis of the second group becomes more difficult because there are not significant differences between different reasoning approaches and the imbalance of membership between classes makes it a very hard classification task.

6 Conclusion

The focus of this paper has been to present and evaluate methods that use a model-based approach in an attempt to improve the accuracy of case-based retrieval. In the reported study, two knowledge-intensive systems were compared to the plain CBR with no additional general domain knowledge. The first of these, named KiCBR-1, uses a semantic network model to infer features that expand the set of features in the input case. The second, named KiCBR-2 used a more focused approach in which root causes failure situations were particularly emphasized. The CREEK framework was used to integrate the case-based and model- based methods. Furthermore, two different similarity measure mechanisms, linear and semantic, were used. The experiments were conducted in the domain of oil well drilling, specifically hole cleaning problems. The study shows that the integration of different reasoning methods improves the reasoning and the retrieval process, with the KiCBR-2 as good as, or better than the two other methods. A part of future work is to test the approach on a substantially larger case base.

Further work

Some of the cases reveal that insignificant downtime while cleaning hole can turn into insignificant or significant downtime while casing poorly cleaned hole and vice versa. From this point of view, the combination of group 1 and group 2 is as important as the individual groups. As cases contains information both about the outcome of the hole cleaning problem while cleaning and while casing, they allow us to form a combined group of classes. It should be emphasized that few cases exist in our data based and this test should be done with more cases.

Acknowledgements

The authors would like to express appreciation to people from StatoilHydro, through its project leader Erik Nyrnes, and colleagues from Verdande Technology, for their contributions to the results presented.

References

1. Brüninghaus, S., Ashley, K.D., Combining Model-Based and Case-Based Reasoning for Predicting the Outcomes of Legal Cases. (ICCBR-03). Trondheim, Norway. June 2003. Springer Verlag: Heidelberg, Germany. Lecture Notes in Artificial Intelligence LNAI 2689. Pages 65-79, 2003.
2. Grimnes, M., Aamodt, A., A two layer case-based reasoning architecture for medical image understanding. In Smith, I and Faltings, B (eds) Advances in Case-Based Reasoning. Berlin: Springer, pp. 164-178., 1996.
3. Arcos, J.L., deMántaras, R. L., Serra,X. , Saxex: A case-based reasoning system for generating expressive musical performances. Journal of New Music Research, vol. 27, no. 3, pp. 194-210, 1998.
4. Skalle, P., Aamodt, A., Knowledge-based Decision Support in Oil Well Drilling. Proceedings of the ICIIP, International Conference on Intelligent Information Systems. Beijing, 21-23 October 2004.
5. Popa, A., Popa, C., Malamma, M., Hicks, J., Case-based reasoning approach for well failluer diagnostics and planning. SPE paper 114 229, presented at the 2008 SPE Western Regiounal and Pacific Section AAPG Joint Mtg, Bakersfield, 2008.
6. Koton, P., Reasoning about evidence in causal explanations. Proceedings of the Seventh National Conference on Artificial Intelligence (AAAI-88), Menlo Park, California: AAAI Press, pp. 256-261., 1988.
7. Aamodt, A., Knowledge-Intensive Case-Based Reasoning in CREEK. Lecture Notes in Computer Science, 16 November 2004: p. 1-15.
8. Shokouhi, S.V., Aamodt, A., Skalle, P., Sørmo, F, Determining root causes of drilling problems by combining cases and general knowledge. Proceedings of the 8th International Conference on Case-Based Reasoning, ICCBR 2009, Seattle, July, 2009.

9. Bourgoyne, A.T., et al., Applied Drilling Engineering. Second printing ed. SPE Textbook series. Vol. 2. 1991: Society of Petroleum Engineers. 502.
10. Datta, B.K., Ratnayake, C., Saasen, A., Omland, T.H., Hole Cleaning and Pressure-Loss Prediction From a Bulk Transport Perspective. Paper SPE 96315 presented at Offshore Europe, 6-9 September 2005, Aberdeen, United Kingdom, 2005.
11. Aamodt, A., Plaza, E., Case-Based Reasoning: Fundamental Issues, Methodological Variations, and System Approaches. Artificial Intelligence Communications, 1994. 7 (1): p. 39-59.
12. Sørmo, F., Aamodt, A., Knowledge elaboration for improved CBR. In Sixteenth International Joint Conference on Artificial Intelligence, Workshop ML-5: Automating the Construction of Case-Based Reasoners. Stockholm, 1999: p. 39-43.
13. Díaz-Agudo, B., González-Calero, P. A., An Architecture for Knowledge Intensive CBR Systems. In Blanzieri, E., Portinale, L., (Eds.): Advances in Case-Based Reasoning (Procs. of the 5th European Workshop on Case-Based Reasoning, EWCBR 2000), Lecture Notes in Artificial Intelligence, 1898, Springer, 2000.