Abductive Reasoning in Environmental Decision Support Systems

Franz Wotawa, Ignasi Rodriguez-Roda, and Joaquim Comas

Abstract Decision trees and rule-based system including variants based on propositional and fuzzy logic have been the method of choice in many applications of environmental decision support systems. Reasons are the ease of use, the capability of representing uncertainty, and the fast computation of results at runtime when using decision trees or other similar means for knowledge representation. Unfortunately there are drawbacks related with these modeling paradigms. For example, the cause-effect relationships between quantities are not captured correctly. The resulting model is well appropriated for a certain purpose but can hardly be re-used. Moreover, maintaining the knowledge base can be an intricate task. In this paper we focus on the problems related with decision trees in the context of environmental decision support systems using an example from the domain. We further present abductive reasoning as an alternative for modeling and show how these technique can be easily implemented using existing techniques.
Keywords: environmental decision support systems, abductive reasoning, modelling

1 Introduction

Decision support systems gain importance. This holds especially in the environmental domain where decisions have to be drawn but where knowledge is not commonly accessible and not easy to obtain. In most cases environmental decision support systems like [1, 2] are based on decision trees, rule-based systems, case-based systems, or fuzzy logic. Although, these methodologies have been successfully used in practical applications, e.g., [16], they have some drawbacks. One drawback is that the

Franz Wotawa
Technische Universität Graz, Institute for Software Technology, Inffeldgasse 16b/2, A-8010 Graz, e-mail: `wotawa@ist.tugraz.at`

Ignasi Rodriguez-Roda and Joaquim Comas
Laboratory of Environmental and Chemical Engineering University of Girona, E-17071 Girona, Catalonia (Spain) e-mail: {`ignasi.rodriguezroda`, `joaquim.comas`}`@udg.edu`

models used are different from the models used in physics or chemistry. Hence, it is necessary to rewrite the model in order to fit the purpose. On the other hand there is the advantage that decisions can be easily obtained from the models and thus explaining the reasons behind a decision is easy.

In order to make modeling straightforward without loosing the capabilities of providing a decision and the reasons behind in an easy way, model-based reasoning has been developed. In model-based reasoning a model is directly used to provide solutions for a problem like diagnosis. The model itself should be as close as possible to the original model. Model-based reasoning has been successfully applied to diagnosis [17]. There are applications also in the environmental domain. See for example the work by Struss and colleagues [12, 13, 18, 14]. All described applications are based on consistency-based diagnosis where models of the correct behavior have to be available. However, there is another methodology for model-based reasoning which relies on abductive reasoning. In abductive reasoning models of the faulty behavior have to be formalized in order to get an explanation for a given problem. In the environmental domain the faulty behavior is usually known as well as their consequences. Hence, abductive reasoning seems to provide a good foundation for environmental decision support systems.

In this paper, we discuss the problems behind decision trees and other similar methods used for modeling in detail by means of using a knowledge-based model to detect the risk of microbiology-related solids separation problems, which is one of the main critical perturbation in the biological treatment of wastewater. Afterwards we introduce abductive reasoning and present an algorithm that allows for computing minimal explanations. Finally, we give a brief presentation of our implementation and a conclusion.

2 Problem description

In this section we discuss modeling using decision trees in detail. We use Comas et al. [2] decision tree model that is used to predict the risk of bulking, foaming, and rising sludge, microbiology-related operational problems when simulating biological wastewater treatment. Beside the decision trees for the different types of risks the authors give a verbal explanation and tables where the involved parameters and their corresponding risks are related to each other. In order to be more comprehensive we focus on only one simplified phenomenon, i.e., the risk of foaming. According to Comas et al. [2] the risk of foaming is influenced by the food-to-microorganism ratio (F/M_fed), the sludge residence time (SRT), the dissolved oxygen concentration (DO), and the ratio between readily biodegradable substrate concentration (S_s) and the slowly biodegradable substrate concentration (X_s). The verbally given explanations relate these parameters to the growth of certain microorganism, i.e., *Nocardioforms* and *Microthrix parvecilla*, which cause two different types of foaming. The given explanation is modeled using decision trees. Figure 1 depicts the given decision tree. Moreover, the authors also specify the behavior by means of a table.

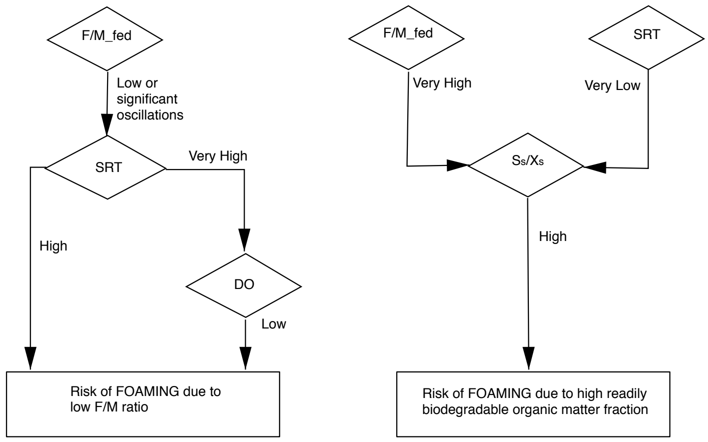

Fig. 1 Decision tree for evaluating the risk of foaming taken from [2]

In the given table each entry specifies a qualitative value for the risk of foaming given a qualitative value for a parameter. In order to be self contained the table representing the knowledge stored in the left decision tree of Figure 1 is given in Table 1 ignoring the DO parameter. Beside the fact that the DO parameter is not represented it is interesting to note that the authors use a qualitative representation for the parameters and the risk assessment. To obtain a qualitative representation of a parameter value requires an additional step. In the case of Comas et al. [2] a fuzzification step is used. After obtaining a qualitative value for the risk defuzzification can be used to finally receive a quantitative risk value.

Table 1 Table representing the knowledge behind foaming due to low F/M

	F/M_fed			
SRT	Low	Normal	High	Very High
Very Low	Low	Low	Low	Low
Low	Low	Low	Low	Low
Normal	Moderate	Low	Low	Low
High	High	Moderate	Low	Low
Very High	High	Moderate	Low	Low

Given the table and the decision trees it becomes obvious that the original decision tree does not represent the cases where the risk assessment is at the level moderate or low. Moreover, the decision trees from Figure 1 should be put together in order to represent the whole knowledge at once. Both of these weaknesses of the original decision tree model from [2] can be easily incorporated in a decision tree by introducing new leaf nodes for the different sort of risk assessments and new connections. However, there are other issues which hardly can be tackled in such a simple way.

Consider the SRT values given in Table 1. If SRT is known to be low the foaming risk is low and there is no need to know any value for F/M anymore. Hence, the size of the decision tree, which corresponds to the number of decisions, depends on the ordering of the decision. Algorithms for decision tree induction take care of the size of the decision tree by selecting the optimal ordering of decisions to be taken. But even in the case of optimal orderings a decision tree requires to answer decisions that need not to be answered always. Hence, decision trees cannot adapt to certain situations. We term this problem as *poor flexibility* problem.

Another drawback of decision trees is that the absence of knowledge cannot be handled appropriately. In particular it is hardly possible to conclude any default knowledge. For example, it might be useful to assume that there is a low risk of foaming in case we have no knowledge at all. Such reasoning can only be represented if adding a new decision asking whether there is knowledge available or not. However, in this case the knowledge that no knowledge is there is now explicit available, which is not the same in default reasoning where something can be conclude unless it contradicts given knowledge. Therefore, decision trees *lack handling default reasoning* in case of unavailable knowledge.

Since decision trees are constructed in a way that supports a certain task, they usually do not represent the whole knowledge available. For example, in the waste water treatment plant domain there is knowledge regarding the growth of certain microorganisms as function of parameters. In the decision tree this knowledge is not represented. Instead only decisions regarding parameter values are represented. Decision trees represent a sequence of questions to be answered by the user in order to distinguish final conclusions. Hence, it is not necessary to represent all cause-effect relationships, which do not support this task. The fact that decision trees are *not complete* representations of models for a domain is a problem for extracting knowledge to be generally used.

Because of the mentioned problem of decision trees, i.e. the:

- the poor flexibility,
- the absence of default reasoning capabilities, and
- the incompleteness

they are usually hard to maintain. In many cases small changes of the underlying theory causes huge changes of the decision tree. This might not be a problem in cases where either the domain knowledge is small or not very much elaborated, or where the decision tree is automatically extracted from a set of data. In the latter case the decision trees help to extract useful information regarding relationships of certain parameters.

Modeling in the domain of natural sciences is a difficult and time consuming task. In many cases especially when explanations are important the models represent *cause-effects* relationships because effects can be explained in terms of their causes. In most cases decision trees do not capture cause-effect relationships. Instead decisions are based on effects whereas the leaf nodes represent the causes. Hence, reasoning is applied from effects to causes and not vice versa. Note that in our example application, i.e., the risk assessment, someone can argue that the decision trees indeed represent the reasoning from causes to effects although, the used causes are not necessarily the root causes from which all explanations start. A more typical example of the successful use of decision trees for modeling is discussed in [1] where cause-effect reasoning is not handled correctly.

Given the fact that decision trees are very successfully used in practice a question arises. Is cause-effect modeling essential in practice? No, this is not the case, and even from a more philosophical perspective, causality is not the only way of expressing explanations in science (see for example [15]). However, as already mentioned causality-based models are usually available when it comes to explanations. Hence, when using a modeling paradigm like decision trees, which is not able to represent causality in all cases, we have to convert the cause-effect models into models allowing for reasoning from effects to causes. This conversion is a tedious task because it requires the elimination of multiple explanations. We always are able to eliminate multiple explanations. For this purpose distinguishing observations, i.e., decisions, have to be introduced. In summary, we can say that it is not inevitable for successful practical use to represent cause-effect reasoning in the model but this requires additional effort in order to eliminate ambiguities in explanation.

In the next section, we discuss an alternative reasoning schema, i.e., abductive reasoning, which allows using models representing cause-effect relationship directly and thus avoiding the mentioned problems.

3 Abductive reasoning

Given the problems regarding modeling using decision trees or rule-based systems, which we discussed in the previous section, we now focus on a different modeling paradigm. In abductive reasoning the causes are inferred from a logical model representing cause-effect relationships. Therefore, the logical model is most closely to models available. Note that for example in medicine the available model describe causes, i.e., diseases, and their effects, i.e. symptoms but the medical doctors have to conclude the disease from the available symptoms. Hence, medical doctors always use abductive diagnosis. The formalization of this process including therapy is discussed in [11]. Wotawa [19] describes the application of abductive reasoning in the environmental domain and focuses on effects. In particular [19] introduces an algorithm for computing the next optimal observation necessary to reduce possible explanations. The underlying ideas came from work on consistency-based diagnosis, i.e., [17, 8, 9]. The difference between consistency-based diagnosis and abductive diagnosis is that the former uses the correct behavior only whereas the latter has knowledge regarding the behavior in case of faults. Console et al. [4, 3] formally prove the relationship between abductive and consistency-based diagnosis.

In this section, we focus on the implementation of abductive reasoning. The idea is to rely on well-known algorithms. In particular we show how assumption-based truth maintenance systems [5, 6] can be used for computing abductive explanations for given effects. Before introducing the algorithm we briefly give the basic definitions. For a more detailed explanation we refer the interested reader to [19]. We start with the definition of knowledge base.

Definition 1 (Knowledge base (KB)). A knowledge base (KB) is a tuple (A, Hyp, Th) where A denotes the set of propositional variables, $Hyp \subseteq A$ the set of hypothesis, and Th the set of horn clause sentences over A.

In the definition of KB hypotheses corresponds directly to causes such that for every cause there is a propositional variable that allows to hypothesis about the truth value of the cause. Hence, we use the terms hypotheses and causes in an interchangeable way. Having knowledge about a system and some observations we are interested in finding explanations. This leads naturally to the definition of abduction.

Definition 2 (PHCAP). Given a knowledge base (A, Hyp, Th) and a set of observations $Obs \subseteq A$ then the tuple (A, Hyp, Th, Obs) forms a propositional horn clause abduction problem (PHCAP).

Given a PHCAP we are interested in a solution. Hence, similarly to [11] we define solutions as follows:

Definition 3 (Diagnosis; Solution of a PHCAP). Given a PHCAP (A, Hyp, Th, Obs). A set $\Delta \subseteq Hyp$ is a solution if and only if $\Delta \cup Th \models Obs$ and $\Delta \cup Th \not\models \bot$. A solution Δ is parsimonious or minimal if and only if no set $\Delta' \subset \Delta$ is a solution.

In this definition diagnoses need not to be minimal or parsimonious. In most practical cases only minimal diagnoses or minimal explanations for given effects are of interest. Hence, from here on we assume that all diagnoses are minimal diagnoses if not specified explicitly.

Example 1. We use the rightmost decision tree from Figure 1 and model the knowledge represented there as KB. We use `fm_fed`, `srt`, `do` to represent the involved variables. `foaming_risk` is used to represent risk of foaming. The values of the variables are given in parantheses. The horn clauses for representing the knowledge can be formulated as follows:

`fm_fed(low)` $\wedge$ `srt(high)` $\rightarrow$ `foaming_risk(high)`

`fm_fed(low)` $\wedge$ `srt(very_high)` $\wedge$ `do(low)` $\rightarrow$ `foaming_risk(high)`

This model is not complete because there are currently no hypothesis specified, which are of interest to explain a certain observation. In this example we are interested in explaining the assessment of risk. Hence, we introduce the hypotheses `FM_fed_L`, `SRT_H`, `SRT_VH`, `DO_L` that represents certain values of the involved variables. Extending the KB with information regarding hypothesis requires to add the following rules:

`FM_fed_L` $\rightarrow$ `fm_fed(low)`

`SRT_H` $\rightarrow$ `srt(high)`

`SRT_VH` $\rightarrow$ `srt(very_high)`

`DO_L` $\rightarrow$ `do(low)`

Moreover, from Table 1 we might conclude that a SRT value that is low or very low always leads to a low risk. This holds also for F/M_fed in case of being high or very high. Such knowledge can also be introduced in a similar way: `FM_fed_H` $\rightarrow$ `fm_fed(high)`

`FM_fed_VH` $\rightarrow$ `fm_fed(very_high)`

`SRT_L` $\rightarrow$ `srt(low)`

`SRT_VL` $\rightarrow$ `srt(very_low)`

`fm_fed(high)` $\rightarrow$ `foaming_risk(low)`

`fm_fed(very_high)` $\rightarrow$ `foaming_risk(low)`

`srt(low)` $\rightarrow$ `foaming_risk(low)`

`srt(very_low)` $\rightarrow$ `foaming_risk(low)`

What is missing to complete the KB is information regarding inconsistencies. Obviously it can never be the case that a variable has different values assigned at the same time. Hence, we introduce rules like `foaming_risk(high)` $\wedge$ `foaming_risk(low)` $\rightarrow \perp$ to KB where $\perp$ represents the contradiction, i.e., an always wrong proposition.

¿From the KB we can abductively conclude the following explanations, i.e., diagnoses, from the observation `foaming_risk(high)`: { `FM_fed_H`, `SRT_H` }, { `FM_fed_H`, `SRT_VH`, `DO_L` }. Both diagnoses are parsimonious. { `FM_fed_H`, `SRT_H`, `DO_L` } is also an explanation but not minimal. { `FM_fed_H`, `SRT_H`, `SRT_L` } is not a diagnosis because it leads to an inconsistency.

Abductive reasoning, i.e., providing a parsimonious explanations for observations given a KB, can be implemented by checking all subsets of the hypotheses whether they allow inferring the observations in a consistent way. This unfortunately is not effective in practice. Another way is to rely on available systems and algorithm for reasoning based on explanations. Assumption-based truth maintenance systems (ATMS) [5, 6] can be used for this purpose. An algorithm implementing the ATMS has been provided by de Kleer [7]. Many improvements for computing solutions based on ATMS has been suggested including [10]. An ATMS also works on KB defined in this paper when using the word assumptions instead of hypothesis. The ATMS works on a graph representation of KB. Assumptions, propositions, and the contradiction are represented as nodes. The contradiction is named NoGood in terms of the AMTS. The rules are represented as set of connections between nodes. Every node in an ATMS has a label. The label comprises all sets of assumptions from which the corresponding proposition can be derived.

Example 2. The hypothesis `FM_fed_H` is represented by a node. The label of this node is a set comprising the hypothesis because this node is only true if the hypothesis, i.e., the assumption, is assumed to be true. The label of the proposition `fm_fed(high)` comprises the set `FM_fed_H` because of rule `FM_fed_H` $\rightarrow$ `fm_fed(high)`.

The task of the ATMS is to ensure consistency. This is done by changing the label of nodes. Each node has a label, i.e., a set of sets of assumptions from which the node can be inferred and from which the contradicting node cannot be inferred. The latter requirement causes an ATMS algorithm to remove elements from the label that also lead to the NoGood. Hence, an abductive explanation for a single proposition is an element of the label of the corresponding node. Because of the ATMS these elements provide a consistent explanation and fulfill the definition of abductive diagnosis. The only thing that remains now is the extension to the case where we have a set of observations to be explained. This extension can be easily done by adding a rule where the left side is the conjunction of all observations and the right side is a new proposition `explanation` not used in the KB. Hence, the label of `explanation` provides all abductive diagnoses for the given PHCAP. The following algorithm for computing all parsimonious abductive explanations relies on this generalization.

Algorithm abductiveExplanations

Input: A PHCAP (A, Hyp, Th, Obs)
Output: All minimal diagnoses

1. Store Th in an ATMS
2. Add the rule $\bigwedge_{o \in Obs} o \rightarrow$ `explain` to the ATMS where `explain` is not an element of A.
3. Return the label of `explain` as result.

4 Implementation

We have implemented an abductive reasoning system based on an ATMS using the programming language Java. Figure 2 depicts the main window of our implementation where the user can edit, save, or load a KB. Instead of $\wedge$, $\rightarrow$, and $\perp$ a comma ',' '->', and 'false' are used. To distinguish hypothesis from ordinary propositions the former start with a capitalized character. Moreover, every rule has to be ended with a period '.'. The KB given in Figure 2 is the one discussed in Example 1. Figure 3 shows the window where the results are presented to the user. The label of the NoGood as well as the label of the node `foaming_risk(high)` are given. For the latter we obtain the result also given in Example 1. The labels of the other nodes can be obtained by expanding the nodes.

Beside the given small abductive theory we tested out implementation on other KBs having from 10 to about 50 rules and from about 4 to 12 hypotheses. For all examples, the running time was less than 10 ms on a standard notebook. Because of the fact that the computational complexity of the underlying problem is exponential, we do not expect to be able to handle larger systems comprising hundreds of hypotheses. However, in the environmental domain the number of hypotheses

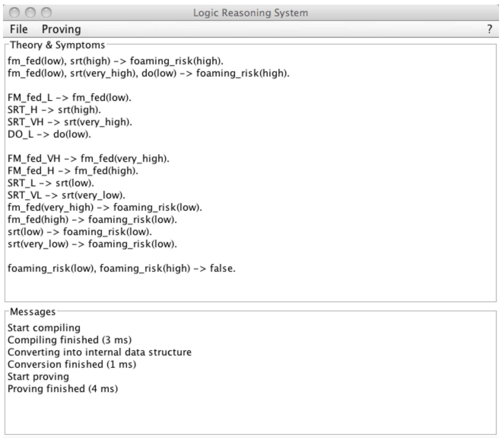

Fig. 2 The main GUI of our implementation

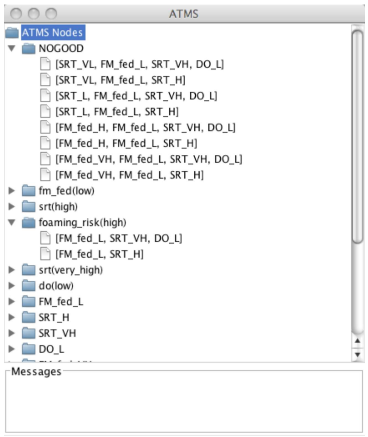

Fig. 3 The GUI providing the obtained results

is expected not to be too high. It is worth noting that the current implementation does neither use the most well elaborated ATMS algorithm nor is itself optimized. Hence, for the desired domain and given today's computational power, the proposed methodology seems to be appropriated.

5 Conclusion

The purpose of this paper is twofold. First, it discusses problems related to currently used techniques for environmental decision support systems, which often rely on decision trees, rule-bases, or fuzzy logic. Problems are the missing flexibility, failing to model default reasoning, and incompleteness. These problems may not impact a certain application. However, they prevent the models to be adapted and used in other applications. Moreover, in most cases creating and maintaining such models is not as easy than expected. Second, the paper provides a solution to the mentioned problem. In particular we propose the use of abductive reasoning as basis for environmental decision support systems. In abductive reasoning models based on cause-effect relationships can be directly used. Moreover, default reasoning is also possible.

¿From a practical view the use of abductive reasoning for applications has been limited because of the unavailability of tools that can be used by people who are not expert in logical-based modeling. Currently, there is an implementation available but the development of models might still not be that easy. In the future we want to focus on usability regarding modeling. Moreover, the whole process of decision making has to be captured by an implementation. Hence, getting more information in a smart way has to be ensured. Again we leave this topic for future research.

Acknowledgements The work described in this paper was partially funded by Austrian Academic Exchange Service (OeAD) under contract ES 19/2008.

References

1. Comas J, Rodriguez-Roda I et al (2003) A knowledge-based approach to the deflocculation problem: integrating on-line, off-line, and heuristic information. *Water Research*, 37:2377–2387.
2. Comas J, Rodriguez-Roda I et al (2008) Risk assessment modelling of microbiology-related solids separation problems in active sludge systems. Environmental Modelling and Software **23** 1250–1261
3. Console L, Dupré D. T et al (1991) On the relationship between abduction and deduction. *Journal of Logic and Computation*, 1(5):661–690.
4. Console L, Torasso P (1990) Integrating models of correct behavior into abductive diagnosis. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, pages 160–166, Stockholm. Pitman Publishing.
5. De Kleer J (1986) An assumption-based TMS. *Artificial Intelligence*, 28:127–162.
6. De Kleer J (1986) Problem solving with the ATMS. *Artificial Intelligence*, 28:197–224.
7. De Kleer J (1988) A general labeling algorithm for assumption-based truth maintenance. In *Proceedings AAAI*, pages 188–192, Saint Paul, Minnesota, Morgan Kaufmann.
8. De Kleer J, Williams B. C (1987) Diagnosing multiple faults. *Artificial Intelligence*, 32(1):97–130.
9. De Kleer J, Mackworth A. K et al (1992) Characterizing diagnosis and systems. *Artificial Intelligence*, 56.
10. Forbus K. D, De Kleer J (1988) Focusing the ATMS. In *Proceedings AAAI*, pages 193–198, Saint Paul, Minnesota, Morgan Kaufmann.
11. Friedrich G, Gottlob G et al (1990) Hypothesis classification, abductive diagnosis and therapy. In *Proceedings of the International Workshop on Expert Systems in Engineering*, Vienna. Springer Verlag, Lecture Notes in Artificial Intelligence, Vo. 462.
12. Heller U, Struss P (1996) Transformation of Qualitative Dynamic Models – Application in Hydro-Ecology. In *Proceedings of the 10th International Workshop on Qualitative Reasoning*, pages 83–92. AAAI Press.
13. Heller U, Struss P (1997) Conceptual Modeling in the Environmental Domain. In *Proceedings of the 15th IMACS World Congress on Scientific Computation, Modelling and Applied Mathematics*, volume 6, pages 147–152, Berlin, Germany.
14. Heller U, Struss P (1998) Diagnosis and therapy recognition for ecosystems - usage of model-based diagnosis techniques. In *12th International Symposium Computer Science for Environmental Protection (UI-98)*, Bremen.
15. Okasha S (2002) Philosophy of Science - A Very Short Introduction. Oxford University Press.
16. Poch M, Comas J et al (2004) Designing and building real environmental decision support systems. Environmental Modelling and Software 19(9):857–873.
17. Reiter R (1987) A theory of diagnosis from first principles. *Artificial Intelligence*, 32(1):57–95.
18. Struss P (1998) Artificial intelligence for nature - why knowledge representation and problem solving should play a key role in environmental decision support. In O. Herzog and A. Günter, editors, *KI-98: Advances in Artificial Intelligence, LNAI 1504*. Springer Verlag, Berlin.
19. Wotawa F (2009) On the use of abduction as an alternative to decision trees in environmental decision support systems In *1st International Workshop on Intelligent Systems for Environmental Engineering and EcoInformatics (i-Seek)*, Fukuoka, Japan.