

Learning in Intelligent Systems for Process Safety Analysis

Chunhua Zhao, Venkat Venkatasubramanian*

{chunhua, venkat}@ecn.purdue.edu

School of Chemical Engineering

Purdue University

West Lafayette, IN, USA

Abstract

Process safety analysis is necessary for analyzing and assessing in detail the inherent hazards in chemical processes. We have developed a tool (called PHASuite) to assist experts conducting process safety analysis. PHA is knowledge intensive, and the analysis capacity and quality of PHASuite depend exclusively on the quality of domain knowledge. It is, however, impossible and impractical to encode all kinds of knowledge into the knowledge base during development phase of PHASuite. Thus, the major aim of this work is to address the important practical learning needs. The learning-from-experience strategy using case-based reasoning methodologies and learning from data using Bayesian learning, are investigated.

Keywords: Process safety, Automated process safety analysis, Intelligent systems, Machine learning, Case-based reasoning, Bayesian learning

1. Introduction

Occupational safety and health are very important issues in process industries. As modern chemical plants have become large and extremely complex, it has become very difficult to analyze and assess in detail the inherent hazards in the plants, to effectively and safely manage changes, to perform maintenance safely, to better control of abnormal events online, and to effectively train operators. PHA review is considered as one of the most important safety related activities within the OSHA PSM regulation framework. For chemical process plants, Hazard and Operability (HAZOP) analysis is the most commonly practiced PHA methodology (Kletz, 1999). HAZOP analysis is the study of systematically identifying every conceivable deviation from normal plant operation, and finding all the possible abnormal causes and the adverse hazardous consequences of those deviations. While HAZOP analysis is a thorough, systematic and successful procedure, it is also a difficult, labor-intensive and time-consuming process requiring weeks or months of effort by several human experts.

A software system, PHASuite (Process Hazards Analysis Suite, Zhao, 2002) has been developed to assist experts conducting PHA. The most important components in PHASuite are: (1) facilities for entering or gathering from other resources, the information necessary for the analysis; (2) creating a proper representation suitable for

* Author to whom correspondence should be addressed: venkat@ecn.purdue.edu

automated analysis based on the process information; (3) knowledge for safety analysis, wrapped in models, to model the process; (4) reasoning engine, consisting of the control knowledge which uses the knowledge to perform analysis on the representation of the process; (5) result management facilities for user reviewing the results generated from analysis.

2. Why Learning?

Consider the following scenario: when analyzing a particular process, after process information is entered into PHASuite, the reasoning engine is invoked to analyze the process using the safety knowledge stored in the models, and results are presented for users reviewing. Users may find some of the results are not appropriate for this particular process. Users may also want to add or modify some of the results. This is to be expected given the generic models in PHASuite. Without proper mechanism to incorporate this kind of knowledge based on users feedback, if similar situation arises later when analyzing another process, PHASuite would generate similar inappropriate results and users would have to make the same changes to the results again. The mechanism of enabling software programs improving automatically through experience is the research area of machine learning (Mitchell, 1997).

The aim of this work is to explore the machine learning methodologies, and investigate how to apply these methodologies in PHASuite, given the prior knowledge in the models. Besides the user feedback, process historical data, which contains real-world information on the relations between process variables, is another resource of learning. The reminder of this paper is organized as follows. Learning from experience using case-based learning methodology is discussed in next section. Learning from data using Bayesian learning is briefly discussed in Section 4. And finally Section 5 gives a conclusion and outlook into future work.

3. Case-Based Learning

HAZOP analysis performed by human experts depends largely on their experience. Experience plays two important roles here. Firstly experience contributes to refinement and modification of reasoning process. Successful experience is solidified into causal relationships between process variables and rules for cause/consequence analysis. Experience's second role is equally important. Experience helps analysis of new processes by recalling similar situation encountered during earlier HAZOP analyses.

Case-based reasoning techniques provide a formal way to organize the different kinds of experience into a formally organized knowledge base, which is easy to access, easy to modify, and easy to expand. In case-based reasoning, new problems are solved by retrieving and adapting solutions of similar problems encountered in the past. Once a new solution is created, it can be stored in memory for potential reuse in future. Thus, learning capability based on CBR makes it possible for PHASuite to organize its 'experience' obtained from previous analysis, and to reuse such experience to improve quality of analysis on new processes.

All CBR methods have similar operations. From the processes point of view, the major steps in CBR are illustrated in Figure 1 (Aamodt and Plaza, 1994). These steps are: (1) Retrieve the most similar case or cases; (2) Reuse the information and knowledge in those cases to solve the problem; (3) Revise the proposed solution; and (4) Retain the parts of this experience likely to be useful for future problem solving. Accordingly, there are five major challenges in order to apply CBR to solve real problems, including case representation, retrieval methods, reuse methods, revision methods, and retain methods. A very important feature of case based reasoning is its coupling to learning. It enables sustained learning by updating the case base after a problem has been solved. When a problem is successfully solved, the experience is retained in order to solve similar problems in the future. When an attempt to solve a problem fails, the reason for the failure can be identified and remembered in order to avoid the same mistake in the future.

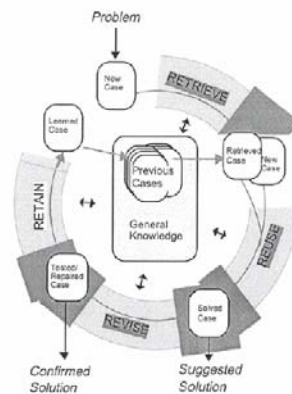


Figure 1. CBR circle: basic CBR operations

In this work, the safety models for operations and equipments are represented as cases. The features that are used to index operation models are: Function type; Subfunction; Physical properties of the processing materials, with values liquid, gas, solid; Processing conditions: Pressure, with values of high, low, normal, or Temperature, with values of high, low, normal; Components (equipments) and the corresponding process variables. The types of functions for device can be divided into: ToMake, ToMaintain, ToPrevent, ToControl (Chandrasekaran, 1994). And for operations, where the functionality is defined on the materials or substance, the functions may be divided into the following types: ToMake, such as reaction related operations; ToSeparate, such as separation related operations; ToMaintain, such as hold; ToMove, such as Transfer; ToChange, such as heat, cool; ToClean: clean, purge; etc.

Case retrieval is one of the main steps when selecting models from knowledge base. The task includes selection of candidate models and ordering of candidate models based on their similarity to the current situation. The process to be analyzed is first assessed and the features that are used to index the models are determined. For example, for charge operation, which needs an operation model with function ToCharge, and the things to be

charged are amount of materials, so the generic add material is located. The specific models are determined by process information. Here the index feature is physical state of material added. So the process information about the material added is assessed. If the material added contains solid material, load model is selected. This process is shown in Figure 2.

If the functional or structure-behavior specification of the desired case and a stored case match at least partially, then the stored case is judged as potentially useful and is selected as a candidate case. For example, consider a model to be selected for a tank, whose component list contains agitator. However, after searching through the knowledge base, there is no index feature for a component agitator. The closest model, which is a model for a tank without an agitator, is selected as the candidate.

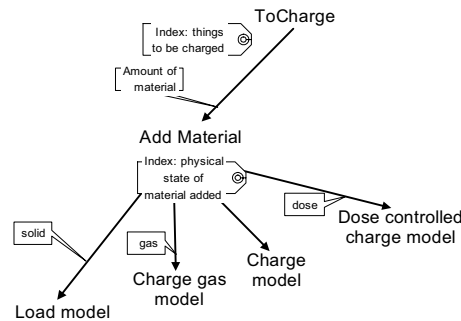


Figure 2. Retrieve a case from case base according to process conditions

Retrieved operation and equipment models are analyzed to see if they are suitable for the present situation. Adequate modification methods are applied to those models if modification is necessary before they are used. The modifications which can be made to the model include: (1) Process variables: add, delete; (2) Causal relations: modify, delete, add; (3) Rules for cause/consequences: modify, delete, add; (4) Rules for digraph propagation: modify, add. Case modification starts after the easiest to adapt candidate model is selected from the ordered set of candidate models. Though this system supplies aids to this modification task, user interaction is heavily involved, which can be called as “on-site model modification”. The aids supported by this system include: commonsense knowledge, and rule-guided repair. Considering the tank example described in last section, when a new component is present in the tank, commonsense knowledge tells that a new process variable should be added to the model to represent the effect of this new component. In the tank example, a process variable *agitation_speed* is added to the model. The relations between this new added variable and other variables should be decided. The system searches the knowledge base to see if there is index for component agitator and finds the corresponding model. Analysis is then carried out to find out the part of the model which gives information which may help in the construction of relations between *agitation_speed* and other variables. Similarly, this approach helps modifications of other parts of the model. This corresponds to recalling the experience of only those parts which are useful for the new situation. In the implementation of this module, knowledge builder will be called up for

the “on-site model construction”. Further research on this topic will help to reduce the involvement of user.

When a new case is created and added into case base, a proper index path needs to be built to store the case into the right position. In most circumstances, the newly added case should have similar index path as the model from which it is created. But new index feature may have to be added to the index path and another depth level may be added to the path. Considering the example of tank, since the agitator is a component, so only index feature needs to be added, without changes to the index path. By changing the index structure and adding the newly created model to the knowledge base, the model can be selected next time when a similar situation is encountered.

4. Bayesian Learning

Current models in PHASuite are deterministic models. Consider the partial model of Filtration operation shown in Figure 3. When analyzing the deviation of high *duration_of_operation*, PHASuite concludes that *separation_extent* is high. However, the probability of *duration_of_operaiton* is high and *separation_extent* is high given *duration_of_operation* is high, is not analyzed. We argue that process historical data, contain real-world information on relations between process variables. So the problem is how can PHASuite learn from the process historical data, and how to combine prior knowledge in PHASuit with the data.

The causal models in PHASuite can be mapped to Bayesian Belief Network (BBN) (Mitchell, 1997) in a straightforward fashion. Similar to the models in PHASuite, BBN is a directed acyclic graph, consisting of nodes which represent variables, and arcs to represent assertion that a variable is conditionally independent of its non-descendants (given immediate predecessors). Direction of the arc represents causality. BBN describes the joint probability distribution governing a set of variables by specifying conditional independence assumptions, and conditional probabilities. Conditional probability table for each variable describes probability distribution for that variable given values of its immediate predecessors. A completely described BBN provides a complete description of the domain. Each entry in the joint probability distribution can be calculated from the information in the network.

Since it is practically difficult to specify conditional probabilities for each variable in BBN manually, the solution is to learn the BBN from data. If assuming that network structure is given in advance and all network variables directly observable in each set of process data, learning conditional probability is straightforward by using naïve Bayes classifier.

After the BBN is successfully learned from process data, more detailed analysis such as risk assessment can be carried out by inference on the network, e.g. by calculating the $P(\text{cake_mass}=\text{low}|\text{duration_of_operation}=\text{high})$. Moreover, with the probability information, it provides better diagnosis capability when using PHASuite real time for online fault diagnosis. When diagnosis of the possible cause of *cake_mass*=high, the

probabilities of possible causes of duration_of_operation, pressure, and cloth_porosity are calculated, instead of only listing three deviations as equal probable causes for this problem using the deterministic models.

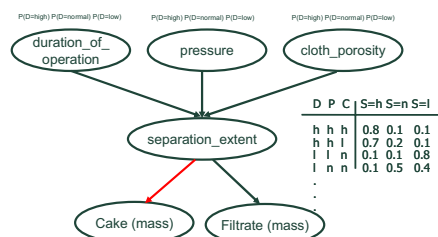


Figure 3. Graphic view of the partial model of Filtration operation in PHASuite

5. Summary and Future Research Plan

In this work, possible machine learning methodologies that can be applied to PHASuite were investigated. Two learning sources are identified: experience from users feedback when using PHASuite conducting PHA, and process historical data which provide information for a probabilistic approach to inference. Case-based learning is discussed in details as well as its applications in PHASuite. Currently, model is considered as a case. Using cases to represent knowledge components other than models, and types of knowledge that are difficult to be modeled by relations between local process variables will be further studied. Case-based techniques are also useful to combine different PHA methodologies, such as HAZOP and What-If to improve the quality of analysis. Cases can also be used to represent past accidents to incorporate this experience into the knowledge base of PHASuite. Ontologies can be developed for the cases, and Semantic Web tools can be used to annotate the document based on the ontologies for knowledge representation and acquisition. Bayesian learning is introduced and discussed as a methodology to combine prior knowledge in the models with process data. The benefits of learning from data are also discussed. As next step, real process data will be gathered and the proposed methodologies will be tested. Based on that, efficient learning and inference algorithms will be identified to be incorporated in PHASuite.

References

- Aamodt, A. and E. Plaza (1994). AI Communications. 7(1): 39-59.
- Kletz, T. (1999). Hazop and Hazan: Identifying and Assessing Process Industry Hazards 4th Edition. The Institute of Chemical Engineers, Rugby England.
- Mitchell, T., (1997). Machine Learning. McGraw Hill.
- Ramakrishnan, R., J. Gehrke (1999). Database Management Systems. 2nd Edition.
- Zhao, C. (2002), "Knowledge Engineering Framework for Automated HAZOP Analysis", Ph.D Thesis, Purdue University, West Lafayette, IN.

Acknowledgements

This project is supported by University of Cincinnati Occupational Health and Safety Education and Research Center Pilot Research Training Program.