Computer-Aided HAZOP of Batch Processes

Stephen A. McCoy*, Paul W.H. Chung and D.F. Zhou
Department of Computer Science, Loughborough University,
Leicestershire, LE11 3TU. UK.

Abstract
Research into the automation of HAZOP has been motivated by the opportunity for reducing the considerable time spent by engineers on such hazard identification methods. However, most work so far has considered steady-state continuous process plants only. This paper describes a prototype for batch HAZOP emulation (CHECKOP), in which variations in operating instructions are considered as possible causes of hazards and operability problems. This extends the automatic modelling of hazards to some classes of human error and their effects on the process, which is modelled as a collection of state-based equipment objects whose behaviour is simulated through time.

Keywords: Model-Based Reasoning, Simulation, Batch HAZOP

1. Introduction

Computer-based hazard identification for chemical plants has been explored by a number of research groups over a period of several years. The most common approach is to use dynamic qualitative models of the process to predict plant behaviour in normal operation and under deviations from normal operating conditions. Therefore, emulation of the HAZOP study has been a focus of interest for the authors (McCoy et al., 1999 and 2000), and for many other researchers (Catino and Ungar, 1995, Heino et al., 1995, Vaidhyananathan et al., 1996, Shimada et al., 1996, Khan and Abbassi, 2000).

Such models have most often been based around a simulation of the important state variables in the equipment item (often using the signed-directed graph as a model), together with a range of information to capture the possible failures of the equipment, and the susceptibility of the equipment to deviations propagated from elsewhere. The equipment models are connected together to form plant models, within which the effects of deviations can be assessed by propagation of the disturbances modelled.

One significant limitation of most of the work in this domain so far is that the models do not handle batch plants and their operations very well, because they are not capable of adequately representing the state of the plant and its evolution throughout the different stages of a batch operating procedure. The purpose of this paper is to introduce recent work done on an initial prototype which tackling batch plant modelling for batch HAZOP analysis of a process.

* Author to whom correspondence should be addressed: s.a.mccoy@lboro.ac.uk
2. Representing the Batch Process

In order to model the behaviour of the plant, and to detect possible hazards in it, a representation needs to be chosen for the equipment in the plant and its intended operation. A state-based object-oriented approach has been taken to modelling physical equipment in the plant, where each component has a state which may change over time, depending on operations performed on it and on events (e.g. spontaneous equipment failures) which may occur within the scenario being modelled.

The decision of how to represent the operating instructions of the plant should consider that, to do batch HAZOP, we need to consider variations in the given set of instructions, and determine what happens to the plant in those cases.

2.1 Petri Nets as a Possible Representation for Operating Procedures

Petri Nets (PNs), as described for example in Murata (1989), are a mature technology for representing sequences of events in discrete event systems. However, are they a suitable representation for the operating instructions in batch processes? In order to use them, we need to introduce a mapping between the elements of a PN and the objects in the plant system:

- **Transitions** correspond to actions which cause a change in the state of the plant.
- **Places** correspond to states of the plant and its equipment.
- **Arcs** connect places to transitions and vice-versa.
- **Tokens** correspond to the associated state condition holding in the plant model.

In addition to the above elements, one place in the net must be labelled as the start of the procedure and one as the finish, to indicate when the procedure has been completed. Petri Nets model the inherent sequence/parallelism in a fully formed plan very well, and are therefore an attractive model for visualising the operations being modelled. However, they do not allow a flexible enough representation of the alternative plans which arise from deviations from intended operations, or where some part of the plant is modified. If we wish to consider what would happen if the order of two tasks were swapped, or if a new equipment item were inserted, we find that it is very difficult to modify the PN to take account of this.

The weakness of Petri Nets for this application seems to be that they integrate the places and transitions (i.e. the states of the plant and the actions that are performed on the plant) too closely. It is better to use a representation which de-couples the operations from the equipment in a plant, so that variations in either domain can be considered more cleanly.

2.2 Chosen Operating Procedure Representation

The representation chosen for operating procedures in this work is to represent a procedure as a sequence of operations, each of which may be decomposed into a sequence of “primitive” actions. In this way, it is relatively straightforward to move actions around in the procedure and see what hazards result in the plant from following the modified procedure.
3. The CHECKOP Prototype

An object-oriented approach is used to represent plant components, so that each component inherits the properties from its class definition. Furthermore, the class definition also includes the operations that objects of that class can carry out. Each operation includes the pre-conditions that must be true before that operation can be carried out and also the after effects of the operation. For example, a tank can be filled from a source. The pre-condition of the operation fill is that there must be a flow path from the source to the tank and that the flow path is not blocked in any way, e.g. any valve between the source and the tank must be open. The after effect is that the tank is filled to the required level and the material in the source is reduced by the same amount.

Given a plant description and a set of operating instructions, a qualitative simulation of the plant going through the state changes is generated. Thus the effect of following a sequence of operating instructions can be analysed. This simulation can be performed for the intended operating instructions, or for a modified set.

3.1 Batch HAZOP Emulation in CHECKOP

Each batch HAZOP guideword is applied by introducing a particular error into the operating instructions and the consequence is determined by running the qualitative simulation. For example, consequence of missing out an instruction or carrying out an instruction in the wrong order can be easily assessed. The guidewords tackled in CHECKOP so far are:

- No action – The identified step is not performed at all.
- Early/Late action – The identified step is performed earlier or later than it should be in the instruction sequence.
- Early/Late termination – Applied to an action which has a qualifying condition (a “do X until Y” action). The action is finished in error before the condition is met, or is continued after the condition is met.

A systematic approach is taken, where each operation in the procedure is considered, with each of the guidewords being applied to the resulting set of instructions. Any hazardous states or operability problems found by the simulation are reported by CHECKOP, associated with the relevant operational step and guideword.

Modelling a wider range of batch HAZOP guidewords will be the subject of further research. For instance, we may consider “Other action” applied to an operation on some equipment item, where the operator performed the stated action on another similar item of equipment (e.g. shutting down the spare pump instead of the running one).

3.2 CHECKOP Application and Results

To illustrate how CHECKOP is used to aid batch HAZOP of a process, we now consider the simple batch processing plant shown in Figure 1. The plant is intended to perform the exothermic reaction $A + B \rightarrow P$, with an excess of reactant B, and to store the resulting product in tank103 prior to purification downstream.
CHECKOP first reads in a description of the plant, which gives the equipment list, connectivity of streams between equipment, and information about the initial states of the equipment. The format of this file is the same as that used in the HAZID tool developed previously by McCoy et al. (1999, 2000). A description of the intended operating instruction, as shown in Figure 2, is then read into the program and stored internally.

charge reactor101 with reactantA:
(1) valve101 open
(2) pump101 start
(3) reactor101 fill_from tank101 with reactantA until volume 30 percent
(4) pump101 stop
(5) valve101 close

start_up reactor101 cooling and agitator:
(6) agitator turn_on
(7) valve104 open
(8) jacket1 cool_content until temperature 25 degree

charge reactor101 from tank102 with excess of reactantB:
(9) valve102 open
(10) pump102 start
(11) reactor101 fill_from tank102 with reactantB until volume 60 percent
(12) pump102 stop
(13) valve102 close

wait until reaction complete:
(14) agitator stir_content of reactor101 until elapsed_time 20 minutes

discharge product mixture for separation:
(15) valve103 open
(16) pump103 start
(17) reactor101 empty_to tank103 with [reactantB, productP] until volume 0 percent
(18) pump103 stop
(19) valve103 close

shut_down reactor101 cooling and agitator:
(20) agitator turn_off
(21) valve104 close

wash reactor101 thoroughly with water:
(22) valve106 open
(23) valve105 open
(24) reactor101 wash with water
(25) valve105 close
(26) valve106 close
When both input files have been processed, CHECKOP proceeds to examine deviations from the intended operating instructions as explained above. For each simulation, it detects when hazardous situations or operability problems arise and reports them. It also continues the simulation after such problems, to see if there are further consequences later on if the procedure is followed blindly. This policy is most likely to over-report hazards because problems are usually (but not always) noticed when they occur. Nevertheless, over-reporting is preferable to missing something important.

A sample of the type of output produced by CHECKOP is given in Table 1, which shows guidewords being applied to only one of the steps in the procedure (step 3). Note that consequences are reported for steps that occur much later than the one whose deviations are being considered – for example, there is an effect on the discharge of product to storage if too much of reactant A is added to the reactor in step 3.

<table>
<thead>
<tr>
<th>Guideword</th>
<th>Operation</th>
<th>Consequences</th>
</tr>
</thead>
<tbody>
<tr>
<td>No action</td>
<td>(3) reactor101 fill from tank101 with reactantA until volume 30 percent</td>
<td>(6) agitator running while vessel empty. (17) cannot empty [reactantB,productP] from reactor101 to tank103 because content of reactor101 is [reactantB]. reactor101 cannot be filled from tank101 because there is no flow path.</td>
</tr>
<tr>
<td>Early action</td>
<td>(3) reactor101 fill from tank101 with reactantA until volume 30 percent</td>
<td>reactor101 cannot be filled from tank101 because there is no flow path.</td>
</tr>
<tr>
<td>Late action</td>
<td>(3) reactor101 fill from tank101 with reactantA until volume 30 percent</td>
<td>reactor101 cannot be filled from tank101 because there is no flow path.</td>
</tr>
<tr>
<td>Early termination</td>
<td>(3) reactor101 fill from tank101 with reactantA until volume 30 percent</td>
<td>No consequence.</td>
</tr>
<tr>
<td>Late termination</td>
<td>(3) reactor101 fill from tank101 with reactantA until volume 30 percent</td>
<td>(17) cannot empty [reactantB,productP] from reactor101 to tank103 because content of reactor101 is [reactantA,productP]. (17) contamination in tank103. (17) exothermic reaction in tank103: reactantA + reactantB → productP.</td>
</tr>
</tbody>
</table>

5. Conclusions

The issue of automating batch HAZOP has been explored – two central issues are the representations chosen for process equipment and operations in the plant. Petri nets are not flexible enough to represent operating instructions, because even a small change to the procedure or to the plant can mean large changes in the associated PN. Therefore, a state-based representation of the physical equipment, and a simple model of the sequence of operations in a procedure, have been chosen for implementation in a prototype system for batch HAZOP emulation (CHECKOP).
The operation of CHECKOP, in terms of its examination of guidewords applied to operations, and the subsequent simulation and reporting activity, has been illustrated by application to a very simple batch process. Initial results are promising, but we need to extend the system so that it can consider a wider range of HAZOP guidewords.

References


