The Knowledge Engineering Review, Vol. 5: 2, 1990, 69–95

AI applications in process design, operation and safety

D. HUTTON, J. W. PONTON† and A. WATERS

Department of Chemical Engineering, University of Edinburgh, King's Buildings, Edinburgh EH9 3JL, Scotland

Abstract

This review provides an overview of the present state of the art in applying AI techniques to the work of Chemical and Process Engineers. Such work is described, where necessary, in order to make this paper understandable to an audience outside the process engineering community.

The first section discusses the nature of chemical engineering design and draws the important distinction between *process* design and *plant* design. The complete design cycle is covered a stage at a time and AI applications and issues identified. The section finishes with a discussion of integrated or concurrent design, which is the ultimate aim of much current research.

The second section discusses AI applications in process operation. This covers process monitoring, alarm processing and fault diagnosis.

The third section discusses, in some depth, design stage loss prevention, that is the design of plants to be safe and reliable. Safety is an integral part of design, but the inclusion of a separate section is appropriate as the qualitative and judgemental nature of many safety related problems has encouraged the use of AI techniques. Applications of AI planning to the problem of automatic operating procedure synthesis and applications of qualitative simulation to chemical engineering problems are also discussed in this section.

1 Introduction

This review is intended to provide an overview of the present state of the art in applying AI techniques to the work of Chemical and Process Engineers. This work covers a very wide range of activities. Practically the only common theme in these relates to their area of application, namely the chemical and process industries. However, since these cover a spectrum of volume and sophistication from tonnage inert solids production to small quantities of highly toxic liquefied gases the common ground is not immediately apparent.

Nonetheless, there are a number of common features to the work of chemical engineers in industry, be they concerned with the manufacture of sodium bicarbonate (baking powder) or the manufacture of methyl isocyanate, the highly toxic insecticide intermediate released at Bhopal in 1982.

Firstly, the major focus of most process engineers' activity is on design. Predominantly, this is the design of the plant, large or small, to manufacture a chemical product. Design of the product itself is becoming of more importance, but it is still overshadowed by the design of the means of production. This is one way in which the activities of chemical engineers differ from those of, say, mechanical or electrical engineers, although those concerned with production or microcircuit fabrication are now finding common ground with the chemical engineering community.

Once a chemical plant has been designed it must be built. Traditionally the process engineer delegates this task, usually to the mechanical engineer. However, he continues to play a coordinating role to ensure that the original designers' intention is adhered to. It is convenient to regard plant construction as a direct extension of design.

† To whom correspondence should be addressed.

Once built, the plant is commissioned and operated. These activities are very much the sphere of the process engineer, although they are ones to which he brings a variety of tools from other disciplines, such as the optimization and scheduling techniques of mathematicians and operational research workers. An area of great importance is that of controlling the process, and hence chemical and control engineers work closely together.

The first two sections of this review will therefore be devoted to AI techniques in Design and Operation respectively. Finally, the crucial importance of safety in all areas of the chemical or process engineer's activity justifies the inclusion of a separate section on AI techniques in safety. Since safety considerations are integral to design and operation this will inevitably repeat some of what has been said before. However, the importance of the subject can hardy be overstated. Furthermore, the nature of safety considerations, which unlike much else in process design or operation can be highly qualitative or subjective, means that this has been a fruitful and interesting area for the application of AI techniques.

2 AI applications in chemical product, process and plant design

We may distinguish three separate design phases. Firstly the design of the chemical product, secondly the design of the process to manufacture the product and finally the design of the physical implementation of the process, the actual chemical plant. The distinction between process and plant is often blurred, however it always exists. The process is the abstract definition of steps to be carried out to produce the product. Its physical existence is on paper or in a computer only. The plant is the hardware of tanks, pipes and machinery built to carry out these steps.

2.1 Product design

Product design is a relatively new field for the chemical engineer. Traditionally chemists designed chemicals. However, increasingly products are required to be more than just chemicals; they must exhibit a variety of chemical, physical, biological or pharmacological properties.

The design of biological products using expert systems has been proposed by Stephanopoulos & Townsend (1985). Methods for designing molecules with desired properties, which would be readily updated using AI techniques, are discussed by Joback & Stephanopoulos (1989). A limited implementation of these is described by Ponton et al. (1989). A special type of chemical product, whose end user is the chemical industry itself, is the catalyst, a substance which accelerates the progress of a particular chemical reaction. The design of these is traditionally viewed as a branch of "black magic", but has been tackled by Banares-Alcantara et al. (1987, 1988) using a hybrid expert system.

As previously indicated, we would expect this to be an area of growth and also a particularly fruitful one for AI techniques due to the loosely defined nature of the tasks and knowledge involved. All the work in this area has used rule-based expert systems with various implementations. This would seem to be an appropriate technique.

2.2 Process or conceptual design

2.2.1 Process route

Process design starts with a requirement to produce a given product and ends with a "design manual" for the process to make it. The design manual is limited in the detail which it provides. Most major items of equipment will be listed, but their sizes, shapes and materials of construction or erection will be left to the next stage, the plant design.

The first step in process design normally includes the choice of a chemical reaction route. Because it involves knowledge of chemistry, it has much in common with product design. Essentially it is a matter of searching out alternate routes from a knowledge base. This is a fruitful field for AI techniques, though not one much explored to date. Two rather different approaches to the problem are described by May & Rudd (1976) and Conti (1987).

2.2.2 Process structure

Having determined a processing route, it is necessary to determine the structure of a process to carry out the operations defined by that route. The difficulty here is that there are many alternatives. The breadth of the chemical engineer's discipline makes him well placed to consider all the process structures possible to implement a given process route. The difficulty thus lies not in generating alternatives, but in screening what is usually too large a number to be evaluated in detail. This "combinatorial explosion" of alternative process structures has led to much work in this area being concentrated on special types of process or process substructures. The two main subsets are those of energy recovery and separation processes.

The first of these has been the more highly developed and offers some interesting experience and important cautions. The earliest work in this area adopted a rule based approach. Indeed, the first publication by Masso in 1969 (Masso & Rudd, 1969) described a "learning" system with rule weights adjusted by subsequent evaluation of the processes generated. The approach was not very successful. In contrast, a more successful rule based scheme used only two rules (Ponton et al., 1979), emphasizing the crucial importance of getting the rules right in the first place. It would thus appear that the insight which leads to the derivation of rules is more important than the techniques for their application. Indeed, the most notably successful work in synthesis of energy recovery systems, that of Linnhoff & Flower (1978) and the subsequent work of Linnhoff's group (Linhoff et al., 1982) is due entirely to analysis of an insight into the problem, and not to the development in AI of search or planning techniques.

Another feature of this area of process synthesis was the almost simultaneous emergence of what might seem to be a quite different technique, namely that of classical mathematical programming (e.g. Kessler & Parker, 1969). This dichotomy of approach has continued with the subsequent parallel development of mixed integer non-linear programming (MINLP) as a competitor to heuristics guided by "physical insight" (Lang et al., 1988).

These two features indeed characterise the whole area of AI applications in process engineering and raise two corresponding questions which have not yet been satisfactorily answered:

1 To what extent can we benefit from rule-based techniques when identification of correct rules is of such crucial importance and is so difficult? Is not the fundamental analysis of problems by classical methods the most important activity?
2 What is the relationship between AI methods and classical mathematical programming? Is there a risk that AI is merely rediscovering inferior search methods?

In the process engineering community there is a current of opinion which suggests that most real problems cannot benefit from rule-based expert systems, as they either cannot be reduced to simple rules, or because once rules have been identified by analysis, the problem is effectively solved.

The second question reflects a continuing debate amongst process engineers (Sargent & Stephanopoulos, 1989) which should give AI workers in the planning field cause for consideration. It is worth noting that in a comparison of AI and mathematical programming Best et al. (1987) found no particular superiority for either approach.

Work on the synthesis of separation processes and on the synthesis of complete flowsheets has tended to confirm experience with energy recovery systems. Ingenious application of new programming techniques, such as by Beltramini & Motard (1988), have improved the ability to handle and display information, but provided no significant insights into process development. The programming of old heuristics using new technology (Niida et al., 1985) has inevitably been no more successful than the same heuristics applied by earlier methods (Thompson & King, 1972).

The conclusion from experience seems to be that the value of AI techniques in process synthesis lies in the ability to represent complex information and not in the provision of planning or rule-based reasoning. Rules are an insufficiently rich representation for complex knowledge. While planning methods may be as good as mathematical programming when the problem involves discrete choices, as is largely the case in the synthesis of process structures, mathematical programming has an overwhelming advantage when we require decisions to be taken on continuous numerical variables, as is the case when process synthesis becomes process design.

2.2.3 *Quantitative process design*

As has already been noted, the phases or stages of the design activity are not so distinct that one can identify a clear break between "process synthesis", the choice of basic process structure, and "process design", the elaboration of the first stage of detail. In particular, it is almost always necessary to proceed sufficiently far into "design" to enable alternative structures to be evaluated and a choice be made between them. However, it is fair to say that "synthesis" is mainly concerned about structure and discrete decisions. In contrast, "design" is more concerned with the choice of sizes and quantities, although discrete decisions about presence or absence of equipment are still required.

The knowledge associated with the process design stage is thus largely quantitative rather than qualitative. AI techniques, and in particular rule-based methods, would not seem to be particularly appropriate. However, one exception is the application of rule-based expert systems to intellectually trivial, though economically important, selection decisions, such as the choice of equipment or materials of construction (Rijckaert & Bogaerts, 1987). More selection systems will be described in section 2.3.

Much process knowledge at the process design stage exists in the form of mathematical models. These are various in form, but the greater part of this knowledge is in the form of a flowsheet model. The methodology of flowsheet models is a technology which has been developed over the last twenty five years (Westerberg et al., 1979) and is now well established in the chemical industry. It is unlikely that any present day continuous process plant does not have such a model.

The structure of a flowsheet model corresponds strongly with the object oriented paradigm. The chemical process "objects" are processing units or operations and the streams of material which flow between them. However, flowsheet models are normally embodied in Fortran programs and data files, and so have not been able to exploit advances in object oriented programming systems. In particular, the model has only two components, data files and a program. Consequently, only the read/write to file relationship of the conventional programming language exists between procedural and data elements of the model.

Along with the increase in quantitative modelling and decision making, the process design phase starts to generate a very large amount of data. The volume of data will increase as the design proceeds through this phase, and into detailed plant design and construction. The importance of data management in process design was recognized only relatively lately (Cherry, 1975; Winter & Rosen, 1986). However, the growing awareness of this importance, and indeed of the amount of data generated in a typical process design, has led to the commercial implementation of process engineering databases. These have generally used standard DBMS and have also tended to concentrate more on the plant and equipment design phase discussed in the next section (e.g. Liles et al., 1988). However, since the power and availability of object-oriented languages became apparent to the process engineering community, a number of relational and object-oriented systems have been implemented as vehicles for research (Stephanopoulos et al., 1987; Yamashita & Motard, 1986). These systems tend to be more flexible than previous attempts so they could potentially be used from earlier stages of design.

The object oriented DESIGN-KIT, developed by Stephanopoulos and his co-workers (1987), exploits the OOL and other facilities of KEE and is leading towards something far beyond a data management system. In fact, what is emerging is a complete environment for performing process design. It is also clear in principle that the database or environment should be extended for use in the next phase of design, that of plant and equipment design, discussed below. Indeed, it is now seen as desirable to perform all design activities, from synthesis to final detailing of steelwork, if not

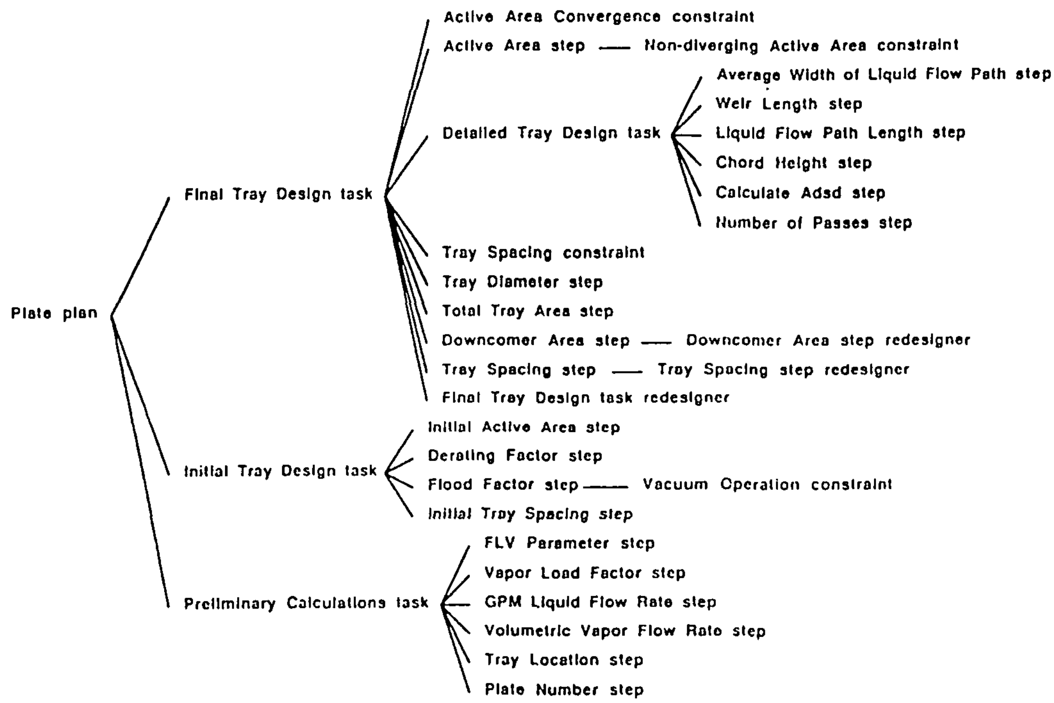


Figure 1 A task network for the routine design of a distillation column (after Myers et al., 1988).

in a common environment, then at least in some integrated manner. Developments in this direction will be discussed in the subsequent section on Integrated Design.

2.3 Plant and equipment design

As with the previous design phases, the boundary between Process and Plant design is a diffuse one. However, one can say that in Plant design one is less concerned with what can be done, which generally requires a fundamental knowledge of chemical processing, than with the means by which it will be achieved, which generally involves a choice between well-defined alternatives. Process design is based on fundamental and thus deep knowledge. Plant and equipment design is routine (Myers et al., 1988) and based on shallow knowledge. Plant design has therefore rather more in common with other areas of manufacturing, and can benefit from expansion in these fields.

The two main categories of plant design activity are selection and planning. In the selection area there have been a number of industrial applications of rule-based systems. Several of these have been concerned with the choice of materials, e.g. Baker-Counsell (1975) and Berkovitch & Baker-Counsell (1985), and others with the choice of alternative equipment items such as pipe fittings and valves (Baker-Counsell, 1985). Almost all of these applications, some of which have proved quite successful, were based on standard shells. However, one chemical company, ICI, has developed its own shell called SAVOIR (Preston & Ponton, 1988), which was written in Pascal.

The best example of a planning application is that of Myers et al. (1988). This is the most extensively developed example of the design of a significant equipment item, namely a distillation unit. This is a complicated system with a number of components whose specification involves quite complex choices. An example of a detailed design plan is shown in Figure 1. The authors used DPSL (Brown, 1984) which was originally conceived as a mechanical design planner.

This type of design is "routine" in the sense that it could be carried out in its entirety by a human designer equipped with a suitably detailed set of design manuals. Indeed, much equipment design is performed in just this way. The power of AI techniques here lies in providing a framework for organizing the design manual information as a knowledge base, and in ensuring consistency of rule application. In conceptual process design, on the other hand, the "rule book" approach has generally failed because it is not possible to reduce the tasks to a set of consistent rules.

2.4 Integrated design

The historical division of design tasks into process design and plant design is largely because of the rather different ways in which the two stages are carried out. Process design is the responsibility of a comparatively small team of chemical engineers with, perhaps, a chemist in an advisory role. Plant and equipment design, sometimes called functional design, involves a wider range of skills. Often, mechanical, electrical and civil engineers will all be involved in a group coordinated by a chemical engineer.

The process design phase is an iterative activity. A number of designs will be considered initially and carried forward in enough detail to enable evaluation and a choice to be made between them. The number of alternatives will be reduced as the degree of detail is increased, until a single process design may be chosen for progression to the plant and equipment design stage. At any point, more detailed information can invalidate assumptions made previously. It may then be necessary to iterate over the previous design stages.

In contrast, the number of different specialists involved, and the very large amounts of information generated at the plant design stage, have so far prevented the overall design procedure from being significantly iterative in the above manner. In particular, the first of these factors forces a division between the two design phases because process design has to be completed before it becomes economically justifiable to involve the various specialists. This makes backtracking between the stages prohibitively expensive. However, not all aspects of process design remain uncertain throughout that phase. Those that are finalized could in principle be carried forward into the plant design stage, provided cognisance is taken of their interaction with other parts of the design. In other words, we should be able to start some detailed design before process design is completed. Not only will this shorten the design cycle but it will also provide detailed information at a stage when backtracking is still possible. This "incremental design" has always been possible in theory, but the difficulty is making the necessary specialist skills available at the relevant time and providing the appropriate information.

The information problem is now being overcome with the use of databases. The specialist skills problem has been minimized now that a large number of those are encapsulated in computer programs, particularly in the mechanical CAD field, and to a lesser, but increasing, extent in expert systems. Much of plant and equipment design is essentially mechanical engineering, rather than chemical engineering, and so can benefit from computer and AI developments in that area.

The greater ease with which the detailed, and previously time consuming, activities of equipment design can now be performed means that the plant design stage can be brought forward into the period previously associated with process design. The procedure can also become iterative provided a database system is available to manage the very large amounts of information generated.

This new approach to process and plant design is called integrated design, and incorporates the concept of incremental design noted above. It depends firstly on the availability of suitable database systems. However, it quickly became clear that a DBMS alone was insufficient, and that in some way the system must encapsulate design knowledge. Finally, current developments are towards the production of integrated design environments which manage both data and knowledge about the design in hand, design methodology in general, and the CAD tools in use.

2.4.1 Integrated data management

In routine design, computer tools greatly help the designer perform the many numerical and database aspects of the design process.

Initially, integration was attempted with the use of databases to link individual programs. Branch (1985) describes the implementation of an interlinked database architecture where a number of small databases, each relating to a particular application task, e.g. synthesis, plant layout, etc. were coordinated by a DBMS. Existing design programs were incorporated into the various tasks.

Britt et al. (1989) point out that such systems have not evolved as a design tool, because they do not realistically represent the creative, trial and error approach to conceptual design tasks performed by engineers. The systems lack the knowledge that an engineer uses to guide the design process. To this end, many researchers have concentrated effort on the development of knowledge bases and associated tools.

The advent of knowledge bases does not signify the demise of databases. Databases are essential for storing the data generated during the course of a design, while knowledge bases store the methodologies and assumptions employed by the engineer. Advances in the area of design will involve the combination of databases and knowledge bases.

The most common vehicle for the knowledge base is object oriented programming (e.g. DESIGN-KIT (Stephanopoulos et al., 1987), (Beltramini & Motard, 1988), which has provided a format that is natural for use in engineering design due to its modularity. Objects directly correspond to the processing units in a plant. Its suitability is further enhanced by inheritance hierarchies which provide a means of increasing the level of conceptual detail as design proceeds. However, it should be stressed that the levels in the hierarchy do not necessarily correspond to the incremental increases in detail in the design.

Much research has been aimed at modelling the design procedure, incorporating the sorts of knowledge identified as being used by experienced designers. Some types of knowledge have already reached a high level of sophistication. For instance, numerical and procedural tasks, such as physical and chemical property data for materials, and mathematical models for simulation are well established techniques. However, implementation of strategic and heuristic knowledge is comparatively limited. Beltramini and Motard (1988) have divided such knowledge into the following categories:

- subjective decision making
- ways of combining information that might not conform to a mathematical model
- knowledge that directs the design towards an optimal solution
- skills in focusing on different parts of the design and deriving information necessary for decision making
- skills in resolving goal conflicts
- strategic knowledge for controlling the overall problem solving activity.

Methods for representing this knowledge are still relatively uncommon and limited. The system described by the authors incorporates a frame-like data structure, an OPS-like production system for general tasks and a second inference engine called the "analytic" engine. This is a dedicated inference engine which is intended to support the heuristic and judgemental knowledge used for hypothesis assessment. Belief networks are used to provide inferential links between evidence and hypotheses. The authors discuss the advantages of this structure over alternative strategies.

The process of design can be viewed as having a hierarchical structure; a structure that has been utilized by many researchers in this field. A plant can initially be modelled by a simple block structure representing high level chemical processes (such as reaction and separation), and can subsequently be expanded incrementally to a more detailed description. Hierarchies can be used both for directing the course of the design towards a solution and for providing a structure for the different models that can be used throughout the design.

The hierarchical approach suggested by Douglas (1985) fits into the first category. Douglas presents a hierarchy of decision levels which increase the detail of the process description in an evolutionary fashion. At each level, the whole plant is considered, increasing the amount of detail with the use of heuristics. In situations where no heuristics are available, alternative processes are generated. The model proposed has been successfully implemented by Kirkwood et al. as an expert system for synthesizing chemical processes (Kirkwood et al., 1988). The program allows the designer to rapidly invent and evaluate a chemical process and possible alternatives.

A model of the design procedure itself has been suggested by Talukdar & Westerberg (1988) who describe a structure incorporating tests, aspects and operators. Aspects are the viewpoints of a design from initial specification to final production i.e. equivalent to levels in a hierarchy. Operators convert input aspects to intermediate and output aspects, i.e. perform the steps between levels which can be manual or automatic. Tests compare aspects for consistency and report the results in another aspect. This is similar to the procedure adopted by Douglas, but is intended for a wider range of application.

The second use of hierarchies is discussed by Lien et al. (1986). They identify that different levels of model are used throughout the design, ranging from qualitative or order of magnitude models at the earliest stages, to rigorous mathematical models for later calculations. During the design, the designer moves between high and low level strategies and back, as required. For instance, at one level in a design, a rapid, approximate calculation will determine the feasibility of distillation for the separation of two components. Once the decision has been made that distillation may be used, a simple model will provide the information necessary to evaluate the whole design on an economic basis. If that analysis is favourable, a detailed model may be used to determine mechanical information such as the dimensions of the column and its fittings. If, however, the design had proved unfavourable then another approximate calculation could be used to evaluate other possibilities. During this procedure, the design has only advanced one step, i.e. a separation may be carried out by distillation, but the detail of the models used to describe it is dependent on the task that the designer is trying to perform.

Lien et al. describe this knowledge as models of the designers and of the available aids. The knowledge describes the tools available and their applicability to a given task. Ideally, the tools would be mathematical models of engineering principles, which, when combined in the correct manner, would provide a description of the design at an appropriate level of detail. However, many large organizations have extensive libraries of computer tools for use in the evaluation of design, ranging from material and heat balances for entire plants, to programs concerned with the detailed design of individual plant items. The utilization of this existing design knowledge, in the form of programs and routines, is an alternative that is being investigated.

An object-oriented approach for CAD tool control based on a blackboard model has been suggested by Daniell & Director (1988). Such an opportunistic approach ensures that the tool most appropriate to a given problem is invoked. In the proposed system, tools are represented by objects which contain knowledge describing the abilities and requirements of the tool. In the above example, the different models would be selected from the library of design programs based on their ability to perform the required calculation. Daniell et al. emphasize that their approach does not require a central data model; which is in contrast to approaches taken in chemical engineering (e.g. the DESIGN-KIT system, Stephanopoulos et al., 1987, described below). The existence of generally available translation programs, for direct communication between pairs of CAD tools, is probably a significant feature of this difference. Few such translation programs exist for chemical engineering computing packages.

2.4.2 Design environments

Design environments provide an interface between designer and tools, maintaining data in a consistent form. Environments do not automatically design chemical plants. They are intended as a vehicle by which the designer can access the tools required throughout the integrated design process.

In DESIGN-KIT Stephanopoulos et al. (1987) have represented plant items by objects within an inheritance hierarchy. The designer can select from a number of standard units or define the operation of a unit by selecting appropriate constituent parts. Rules are used to ensure that such selections are consistent. Further rules infer the configuration of the unit, and, hence, the associated mathematical model for the desired task. The intended applications of DESIGN-KIT are: process synthesis, control loop configuration, plant-wide planning and scheduling of operations and operational analysis of chemical plants.

Westerberg et al. (1989) recognize that the role of design environments is not to automatically design chemical plants, but is to provide a medium for a team of designers to communicate on a common subject (the design). Individual designers may have different points of view pertaining to their specific fields, and should, therefore, be able to access their own models. Obviously, for consistency, only one model should be used. The authors suggest that, to overcome this apparent contradiction, individual designers should be able to access a common model but from their own viewpoint.

The authors are developing a modelling language, ASCEND, which has some object-oriented properties, but is tailored to suit large scale design activities. Units, more traditionally represented by objects, are represented by "models", which have defined "parts" equivalent to slot definitions. The body of the model defines its relationships e.g. is_a, has_part, etc, and contains procedural code defining its mathematical description i.e. the modelling equations. ASCEND has object oriented facilities, such as class definitions and inheritance. However, it also incorporates the ability to directly access all variables. An operator called "ARE_THE_SAME" is used to identify that variables or structures should access the same data. This allows different designers to create models from their own viewpoint, but accessing a common model.

An environment for computer-aided synthesis and analysis has been proposed by Britt et al. (1989). They adopt the approach similar to that suggested by Daniell et al., whereby an object-oriented executive allows the user to interact with application programs, in this case synthesis and analysis programs, and a graphical interface. The existing application programs consist of programs for synthesis and analysis of heat exchange networks, as well as a simulator. The authors suggest the extension of the system to handle entire plants by using the synthesis hierarchy of Douglas and a database.

3 Process operation

Outside the field of AI applications much more research effort has been devoted to process design than process operation. This is largely because design is seen as a "challenging" activity; the design of a large plant involves a significant number of graduate engineers from all disciplines. Operation, on the other hand is felt to be "routine" and is largely the province of technicians rather than graduates. However, a plant normally spends many more years in operation than is spent on its design. And it is of course only in operation that a plant makes the product and profit which is the purpose of its existence.

As a result, with the exception of the special case of real-time process control, the development of computer techniques in the area of operation lags far behind that of aids for design. Most aids for plant operation are still produced on a "one-off" basis for a given plant. In particular, no one has produced a unified methodology of plant operation in the way that Douglas and others (1985) have produced a systematized, hierarchical approach to design.

Nonetheless, not only is the field of operation one of vital importance, it also presents specific opportunities for AI techniques because much of plant operation appears to be based on judgemental knowledge. In process operation we still use a mixture of qualitative and quantitative information but the balance is much less strongly towards the latter than in process design. AI techniques may therefore provide help for a number of specific applications in plant operation. In addition, they may offer aids for the development of a systematic approach to operation as a new intellectual discipline. This would complement the now well developed science of process design.

This section is concerned with techniques used during plant operation to deal with problems that could not be foreseen or prevented during design. We will consider the following chemical engineering problems:

- Process monitoring
- Alarm processing
- Fault diagnosis.

and will refer to AI fault diagnosis and neural nets. We will not say much about real-time issues, although they are obviously of great importance when evaluating the effectiveness of the technology to do an on-line job. The applications discussed in this section, such as fault diagnosis, are largely reactive. In contrast, in section 4 we discuss the application of AI to design stage loss prevention, that is to design for safe and reliable operation. Many of these applications involve planning for process operation, such as the synthesis of operating procedures. These are placed in section 4 because of the crucial importance of safety to them. However, they are also relevant to this section because they could lead towards the general systemization of process operation.

3.1 Process monitoring

George Stephanopoulos (1987) has given wide-ranging consideration to the application of AI techniques in plant operation. He discusses how the planning, scheduling, monitoring, diagnosis and control of process operations can be aided by improved knowledge representation schemes and advanced reasoning control strategies. The central issue is considered to be modelling the following types of knowledge:

a. Processing units
b. Process operations
c. Information handling and process systems
d. Design, planning and diagnostic methodologies

He considers that AI might help by providing the following benefits:

1 Knowledge representation methods that significantly extend the range of utilisable knowledge.
2 A rich repository of procedural models for the diagnosis and/or planning of process operations.
3 Search techniques to tame the complexity of diagnostic and/or planning procedures.
4 Intelligent interfaces to allow high level communication between man and machine.
5 The future potential for on-line "learning" of new knowledge about process operations.

He discusses process trends analysis and the need for a high level, linguistic description of what is happening on plant (such as "distillation column T-1 is adjusting to the new production flowrate"). Most of his comments are speculative, though he does make reference to planning and qualitative simulation work being carried out by his group at MIT.

In another paper Stephanopoulos (1988) divides plant monitoring and control functions into the following three levels:

1 The plant Operations Management Level which is responsible for carrying out corporate plans by the use of:
   - A Production Planner
   - A Start-up/Shut-down Planner
   - A Production Optimiser
   - A Maintenance Scheduler
   - A Crisis Manager
2 The Control Strategies and Co-ordination Level which sends out schedules of control actions to control systems in order to carry out the plans produced by level one by the use of:
   - A Control Optimizer
   - A Response to Faults Scheduler
   - A Process Trends Analyser
   - A Control Systems Assessor
   - A Control Loop Manager
3 The Direct Control Level which sends out signals to actuators and receives signals from sensors. The control units will be at this level but must be more intelligent than at present so that they can change their behaviour at the request of level two.

The author considers that the object-oriented paradigm is the only suitable framework for programming as large a structure as he has described. Again, the paper is speculative and implementation is not really discussed (although the quantitative and qualitative knowledge needed for process trends analysis is described in some detail).

The analysis of process trends has also been discussed by Milne (1989) who describes a pc-based process monitoring system called Annie. This author feels that it is too early to attempt the real-time application of AI. Instead, we should be concentrating on the creation of on-line monitoring systems that are capable of predicting future problems, understanding the current state of the process and recommending a course of action. Annie provides rule-based facilities for attempting such tasks, together with flexible methods of data collection and storage. Process measurements are time-stamped and can be collected at different rates so that slowly changing measurements are sampled less frequently than rapidly changing ones. Statistical methods are used to examine data-trends in order to answer questions such as "How long will it be before that parameter exceeds the alarm value?". However, it is not clear whether domain knowledge can be used in this process in order to give a better fit and a better appreciation of what is going on. Annie is marketed by Intelligent Applications Ltd, Livingston, Scotland. The author feels that the next step is towards "meta process control" that can make recommendations based on a wider view of the whole process.

3.2 Alarm processing

Early work on computer-aids for process monitoring concentrated on the problem of alarm analysis (Andrew & Lees, 1975), which aims to reduce the number of alarms that are displayed in crisis situations to just a few that indicate the real cause of the problem. This is to try to prevent the operator over-load that is thought to have been a major contributor to many plant disasters, such as the one at Three Mile Island. For a discussion of this and other problems associated with the presentation of information to plant operators see Eberts & Lafayette (1985), who describe some of the expertise and cognitive skills required of a process operator.

The problem of alarm analysis is obviously similar to the problem of fault-diagnosis; indeed, early attempts at both of these tasks used techniques similar to those used for automatic fault-tree synthesis (see section 4.1). Now, however, expert systems work is providing new representations of the diagnostic process and new tools for use. Some recent work exploiting these will be reviewed below.

A major experiment in chemical plant fault diagnosis is underway in the form of the FALCON project, which is briefly described in Venkatasubramanian & Dhurjati (1987). The aim is to demonstrate the applicability of Expert System technology to plant monitoring and diagnosis by its large-scale use on an adipic acid plant in Delaware, USA. This has resulted in a knowledge base which uses 600 rules to relate about 200 concepts. The authors claim some success with this project but point out the following deficiences with the approach that was used:

1 The knowledge base is process specific which makes it difficult to maintain or to use on a different plant (even one of the same type).
2 The system is limited to diagnosing faults due to a single basic event.
3 The reliance upon shallow, experiential knowledge means that the system does not degrade gracefully i.e. it cannot give a partial diagnosis of an unanticipated fault.

These problems are addressed in further work described in Rich & Venkatasubramanian (1987) and Venkatasubramanian & Rich (1988) in which the authors discuss a model-based, process independent method for fault diagnosis. In the first reference they describe using a mixture of OPS rules and frames but by the second one they seem to have rejected this in favour of a purer object-oriented, agenda based approach. This allows both the modelling of the topology concerned and the functional modelling of aggregate systems such as control loops. In addition, the second paper describes a two-tier diagnostic strategy that alternates between using compiled knowledge (process specific methods or rules) and using deep-level knowledge (model-based reasoning). Thus they achieve the efficiency benefits of compiled knowledge as well as the ability of model-based reasoning to cope with novel situations. The papers mention the use of confluences, and other qualitative simulation representations, but this seems to be restricted to chaining through influences rather than being used to perform constraint propagation, keep track of landmark values or model device operating regions.

A multi-tiered diagnostic strategy is described by Ramesh et al. (1988) who use the task-oriented approach to expert system development, which is being developed by the AI in Engineering group at the University of Ohio. For the process engineering domain the framework is composed of two problem-solving structures. The first is associated with a task called classification and uses a hierarchy of malfunction hypotheses to allow localization of a fault. The second is used to explain complex product quality problems in terms of plant operating parameters so that the classification structure can then be used to localize the fault. Each node in the classification hierarchy is an integrated knowledge base and inference mechanism, i.e. it contains both the knowledge required to assess the associated hypothesis and the local control strategy to decide where to go next. Using this one can proceed from a general malfunction hypothesis at the top of the tree (e.g. a hypothesis that the fault is located in a particular section of plant) to successively refined hypotheses further down (i.e. narrowing down the fault to smaller sub-sections or individual units). In this structure compiled knowledge is no longer just experimental or heuristic but instead represents an abstraction of more detailed knowledge, which can be used to narrow the search. Consequently, the upper-levels of the tree will contain compiled knowledge whilst the lower levels approach model-based knowledge. The task framework is said to provide a degree of modularity and transparency to the user, in the form of a problem-solving method specifically designed for diagnosis in process plants. This leaves the implementor free to concentrate on the domain knowledge, rather than on program development.

A slightly different approach to diagnosis is taken by Kramer (1987) who presents a General Diagnostic 'Framework (GDF) to exploit the inverse relationship between process behaviour and diagnosis. He considers that there are only two basic approaches to diagnosis:

1 A detection-oriented (DO) approach that compares the real process to a model of the fault free process.
2 A best-fit approach which models a range of scenarios to see which best fits the data.

This distinction seems to be similar to the distinction between top-down and bottom-up reasoning, with the DO approach being essentially top-down. This will be further explained in section 4.1. The DO approach is the one taken in the GDF, which introduces the notion of a switch to specify whether or not a process model is fed the expected (reference) value for a particular parameter or is fed the measured value for that parameter. This means that for a system with n switches (inputs) and m comparison points (outputs) there are potentially $m \times 2^n$ patterns to consider. This number will be reduced because many parameters will be unmeasured and some may not have reference values. Nevertheless, it may still represent an enormous amount of untapped knowledge available for diagnosis.

Like the approach of Ramesh et al., which is described above, Kramer's approach also views diagnosis in a hierarchical manner (although the GDF only operates within a single level of abstraction). It is shown that the GDF reduces to constraint-based diagnosis when all the switches send measured values to the model and to digraph-based diagnosis when they all send reference values. In addition, the GDF is extended to include an "open-loop" version which detects deviations in comparison points and a "closed-loop" version which tries to change the process model in order to satisfy the comparison points. The GDF is therefore a superset of many other diagnostic strategies. The paper does not discuss implementation of the GDF. The author has compiled a bibliography of over 400 references on fault diagnosis and expert systems, which he makes available on request.

The papers reviewed above have all described a symbolic approach to fault diagnosis. A rather different view is taken by Hoskins and Himmelblau (1988), who have been working on neural nets. In this paper the authors describe the characteristics of neural nets that are useful for knowledge representation in chemical engineering. They explain the concepts of nodes, connections, weights and learning algorithms and describe an environment called ZNL that can be used to simulate different neural net architectures for the purposes of experimentation. To illustrate the concepts they teach a network to recognize six types of faults in a system consisting of three reactor tanks in series.

This is similar to statistical approaches to pattern recognition but requires less information and makes less assumptions about underlying probability distributions. Nevertheless, the learned information is presumably process-specific and so will be brittle to changes in process structure and to unexpected faults. No integration with deep-level knowledge is discussed.

This section has so far discussed some recent papers concerned with the application of AI techniques to the monitoring and fault diagnosis of chemical plants. While this work is of great interest it should be noted that much of the work concerned with diagnosis and operation does not make reference to AI techniques; either because it preceded the development of such or because the approach taken is mathematical, rather than representational. This is particularly true with regards to design for diagnosability and operability; for example see Park & Himmelblau (1987) and Russell & Perkins (1987).

An area missing from our discussion so far is that of process control and the design of systems for such tasks. This is an inter-disciplinary topic which is sometimes researched in chemical engineering departments, sometimes in electrical engineering departments and sometimes in specialised control departments. There have recently been some large projects concerned with the application of modern computing tools and techniques to control system design—such as the ECSTACY project at the University of Manchester (Munro, 1987) and an ESPRIT project called "Design and Experimentation of a KBS Development Toolkit for Real-Time Process Control" (Leitch, 1988). In addition, there are now some large AI Toolkits that are specifically intended for process monitoring and control, such as MUSE (Peel & Morris, 1989) and G2 (Matthews et al., 1988). We mention these for the sake of completeness but a review of these projects and toolkits is outside the scope of this paper.

4 Safety and loss prevention

In 1974 a pipe failure at the Nypro plant in Flixborough caused the release of a large amount of cyclohexane. This then exploded causing 28 deaths and 36 million pounds worth of damage (I. Chem. E., 1977). Since then, Loss Prevention has been recognized to be a crucial part of the design and operation of chemical plants. Consequently, formal methods have been used for assessing plant safety, both at the design stage and during plant operation. Modern plants are very complex, making information management the key to ensuring that projects remain both safe and economically viable. In addition, predictive techniques must be used to assess the result of changes and to determine what action to take to avoid undesirable outcomes.

This section will review chemical engineering AI research with respect to its relevance to some of the formal methods used for Loss Prevention. In section 3 we have already covered the use of AI for process operation. This covered problem areas such as process monitoring, alarm processing and fault diagnosis; all of which are partly concerned with loss prevention. In this section we will concentrate on methods used at the design stage to identify and solve problems by redesign. The chemical engineering problems considered will be:

- Fault-tree Synthesis
- Hazard and Operability (HAZOP) Studies
- Operating Procedure Synthesis

These problems will be shown to be relevant to the following AI areas:

- Rule Induction
- Qualitative Simulation
- Planning
- Use of AI-toolkits

Identifying problems at the design stage is crucial to the safety and viability of a plant project. Manual and automatic on-line systems will not react quickly, or correctly, to spurious problems unless they are given the hardware and software to do so. The plant cannot be built, started-up, controlled and shut-down safely unless we have the necessary equipment and procedures in place. Oper-

Hazard Study 1 (project exploration stage):
This ensures that the hazardous properties of all the materials involved in the process and their potential interactions are understood. Also that full consideration is given to constraints imposed by the local environment and legislative constraints.

Hazard Study 2 (process flow diagrams available):
The main plant sections are studied in turn and potential hazards identified in a top-down manner. Fault-trees and quantified hazard analyses are used where appropriate. Some hazards will be designed out, others will be prevented by protective equipment.

Hazard Study 3 (Engineering Line Diagram Completed):
This is conducted as a Hazard and Operability (HAZOP) study by the design team and a HAZOP study leader. The ELD is examined on a line by line basis. A list of "guidewords" are used to stimulate the imagination of the study members into identifying causes and consequences of process deviations. Thus problems are identified in a bottom-up manner, which makes the study intensive and laborious, though extremely thorough.

Hazard Study 4 (Pre start-up):
The commissioning manager conducts this study to ensure that all the recommendations made during hazard study 3 have been carried out. In addition, he or she must ensure that operating and emergency procedures are available.

Hazard Study 5 (After construction):
The operating management are responsible for conducting this on-site inspection to ensure that the workforce are protected against everyday physical hazards and have adequate means of escape. Physical hazards, such as tripping or slipping, continue to cause more deaths or injuries than process hazards.

Hazard Study 6 (After some operation):
This study reviews changes made during commissioning and early operation. It provides feedback to the design team and information for ongoing control and auditing tasks.

Figure 2 ICI's six stage approach to process safety.

ators may fail to control, or even exacerbate, plant perturbations unless given sufficient information about what is happening and what to do about it. In addition, the plant cannot be maintained safely unless there is room to manoeuvre and a strict requirement for permits to work. All these are obviously problems for plant operation but they must be pre-empted as early as possible in the design process to avoid costly redesigns, loss of plant availability and serious damage to equipment and personnel.

For more information about the hazards associated with chemical plants see Wells (1980) which has a complex and extensive design safety check-list. In addition, Duxbury and Preston (1987) give a useful review of current approaches to design safety assessment and discuss the application of computers to the following, safety related, areas:

- physical property determination
- synthesis
- simulation/modelling
- design/evaluation
- databases
- expert systems
- information systems

Most of the cited examples involve conventional technology, while the section on expert systems is largely speculative. The paper does, however, describe an interesting vision of a future in which plants are physically small, inherently safer and virtually unmanned. To realise such advances the design and operation of such plants must utilise new, possibly knowledge-based, computing techniques. For example, a future design environment should be capable of automatically producing the design manuals and of passing information to plant operating systems in a usable way.

The authors have also described ICI's six stage approach to ensuring the safety of new process plants, which we have summarized in Figure 2. This describes a discrete approach to safety assessment; perhaps because the lack of integrated, computational techniques limit our ability to make it a continual part of the design process. In the future we may hope that process plant design will become more integrated, as described in section 2.4, and all features of design will be considered more regularly. Nevertheless, ICI'S approach gives us a useful way of categorising the common ground between AI research and Loss Prevention. The rest of this section will broadly follow this structure.

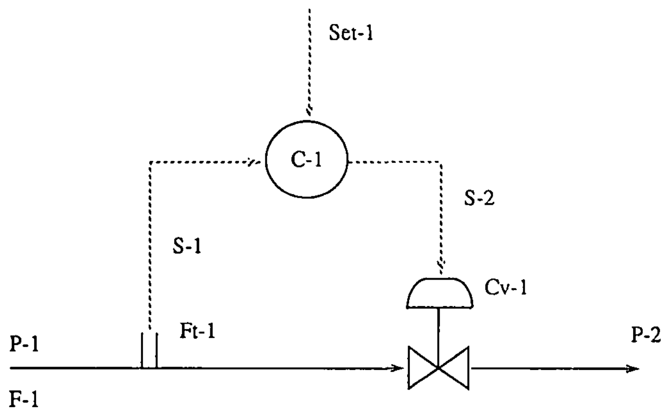
4.1 Fault-tree synthesis

Hazard Study 1 is essentially an information gathering exercise to obtain as much data as possible about the hazards likely to be associated with the plant. This is important as we can sometimes immediately rule-out alternatives on the basis of safety. In addition, it may be necessary to begin experimental programs to obtain vital data which is missing. There is not much AI research specifically relevant to this stage although recent work on physical property prediction (Ponton et al., 1989) could be useful. More importantly, however, Hazard Study 1 will probably benefit from improved techniques of information management and retrieval. Such problems have already been discussed in Section 2.4 and will be further discussed when we consider Hazard Study 3 below.

Hazard Study 2 is performed when the process flow diagrams are available and reasonably fixed. Usually this means that a small number of process routes have been chosen for further consideration and have been compared by numerical simulation and optimization. At this stage we start looking at the hazards that we wish to avoid, such as vessel overpressurization, and at how they can arise in a particular process configuration. There are two ways of doing this, which we may call the "bottom-up" approach and the "top-down" approach. The first requires simulation in some quantitative or heuristic manner to assess the outcome of plausible scenarios such as pump failures. The second approach takes a top event of interest, such as "overpressurization of tank-1" and reasons backwards to the trigger-events that may cause it. The "bottom-up" approach is inherently more combinatorial than the "top-down" one because it may require the simulation of events that do not lead to a hazard. However, simulation is more capable of correct reasoning because time and sequence can be properly captured and alternative worlds modelled in a consistent way. Such comparisons are more fully explored by Waters & Ponton (1989).

The top-down approach usually involves the synthesis of fault-trees to represent the causes of the undesirable events. The use of fault-trees is described in Wells (1980) and an example fault-tree is given in Figure 3 of this review. They are constructed by starting with a top-event and identifying the sets of sub-events that could immediately cause it. We then identify the immediate causes of the sub-events, and so on, until all branches terminate in events which are basic enough to require no further development. The tree can be quantified by placing probabilities on the occurrence of the basic events and combining them as for a decision tree. A fault-tree is therefore a useful representation but its construction is something of an art because of the difficulty of discretizing continuous variables and of connecting the resulting events together in a consistent manner. Constructing fault-trees is a difficult and laborious task hence there has been a large amount of research concerned with automatic fault-tree synthesis. Unfortunately, little of this has resulted in usable software, possibly because of the limitations of the "traditional" software tools employed.

There have been two approaches to the representation of process plant behaviour for the purposes of automatic fault-tree synthesis. One of these we may call the "tree-approach" (Martin-Solis et al., 1982; Taylor, 1982), which is device-centred, and the other we may call the "structure approach" (Shafaghi et al., 1984; Lapp & Powers, 1977), which exploits wider knowledge of the system involved. The former of these is essentially rule-based, though the references quoted precede the development of rule-based shells and languages. The latter of these has only been programmed in an ad-hoc way by using complicated algorithms to model the behaviour of different sub-systems, such as control or trip loops. The former approach is easy to program but difficult to get right and develop since the component models will typically have to embed assumptions about the behaviour of other parts of the process. Perhaps modern shells and rule-based languages will make it easier to



P-1 & P-2 are pressures

F-1 is a mass flow

Ft-1 is a flow transmitter (sensor)

C-1 is a control unit

S-1 & S-2 are control signals

Set-1 is a set point

Cv-1 is a control valve

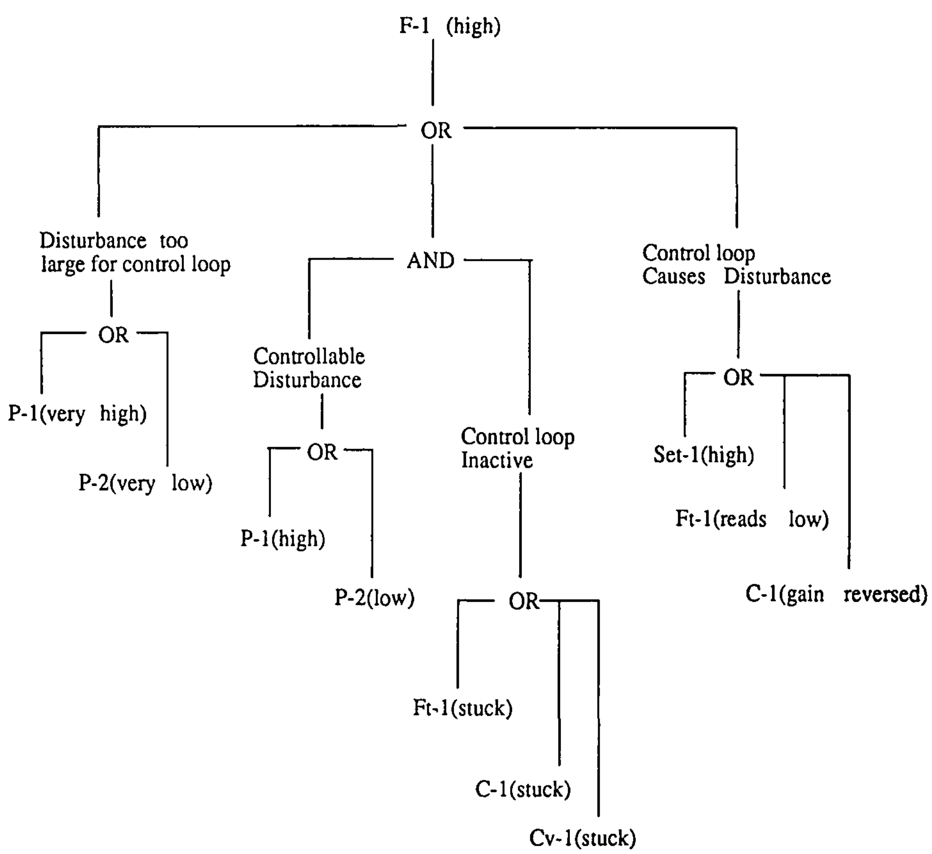


Figure 3 (a) A simple flow control loop. (b) A fault-tree describing how high flow can arise at the exit of the control loop in figure 3(a).

manage such problems. Indeed, work is currently underway to apply Lisp & Ops5 to the problem of automatic fault-tree synthesis (Booth & Jones, 1987; Harris & Jones, 1987) but the results are not yet available. The structure approach shows more promise, from a representational point of view, but is complicated to program. Perhaps frame-based systems and AI toolkits will help here by providing the flexibility needed to represent the range of non-exclusive and interacting process systems needed for a real plant design. We know of no work exploring such avenues to the particular end of the fault-tree synthesis but there is other work concerned with the use of AI-toolkits to represent process plant behaviour. These could have a bearing on the problem and have either been discussed already or will be reviewed below.

4.2 Hazard operability (HAZOP) studies

In between Hazard Study 2 and Hazard Study 3 the detailed design proceeds and culminates in the production of the Engineering Line Diagram (ELD), which is also known as the Piping and Instrumentation Diagram (P & ID). During this phase detailed safety questions will be considered, such as "Do we need a pressure relief valve on this tank?" or "Should we fit fire insulation around this structure?". Bunn & Lees (1988) have examined a large range of such small, detailed decisions in order to consider their suitability for expert system techniques. For each type of task he discusses the sort of expertise involved and briefly points to the technology that may be useful. In addition, he gives the example of deciding whether or not to fit emergency isolation valves as a case where the expertise exists in the form of case studies. He then describes an attempt to apply rule-induction techniques to this problem. Really he is only considering rule-based technology, though he makes passing reference to frames.

ICI's Hazard Study 3 is the biggest of the six stages and is conducted as a Hazard and Operability Study (HAZOP Study). This is the most important part of the six safety reviews but is unfortunately very labour intensive. It can tie-up the top design personnel for weeks, or even months, which limits its use to a single application when the chemical engineering is essentially completed. This is not the ideal situation because mistakes will be identified rather late in the design, with significant financial penalties.

The ELD should represent every pipe, vessel or flange that is intended to be on the final plant. Accordingly, it is rather complex and often describes thousands of items of equipment. To manage this complexity a systematic method is used whereby the team consider a line at a time and run through a list of guide-words to generate possible problems around that line. For example, the team would consider possible causes and consequences of "no-flow" in the line being examined and would decide upon corrective action if there is a problem. The emphasis is on group interaction and the use of guide-words to stimulate the imagination of the team members in order to release their enormous amount of background and design specific knowledge. The theory of HAZOP studies is described in (CISHC, 1977; Kletz, 1983) while Roach and Lees (1981) have examined study records to determine what really goes on. In addition, Andow (1988) has quoted Ferguson (1985) as having briefly investigated the knowledge needed for HAZOP studies. It was found that several problems exist in defining HAZOP study knowledge as a knowledge base for expert system implementation:

1 The ELD contains a lot of pictorial information that it is not easy to represent textually.
2 One of the guide-words used is essentially a "catch-all" for anything not yet mentioned.
3 A lot of the knowledge is common sense, or at least common sense to engineers. It is not explicitly stated in design manuals.
4 Humans know what they do not know. Being able to ask for further information is an essential part of the study.

Andow concludes that the knowledge needed for a HAZOP study is "broad and deep", which is the antithesis of the classic application area for expert systems.

Rule-based approaches to HAZOP study aids have been attempted by Weatherill & Cameron (1988) and Suokas et al. (1987). Weatherill and Cameron report the use of a pc prolog to construct an interactive aid for HAZOP studies, and claim that their system is modular and easily extended for serious industrial use. However, their tool seems little more than a check-list that asks the user the questions he should be asking himself. There does not appear to be any significant reasoning done by the system and the representational methods are simplistic, which makes their claim barely credible. Similar criticisms can be applied to the work of Suokas et al., although this is on a larger scale and uses the AI toolkit KEE. The failure here is that the system does not seem to do anything approaching simulation, which as we will see below is a large part of the HAZOP process. This can be seen from the example output in the paper in which "high input concentration" is given as a possible cause of "high concentration" in a vessel. From this we see that it has been left to the user to find out the cause of the high input concentration in the rest of the process. In addition, causes do not seem to be properly associated with consequences and corrective action, so one does not know which cause will result in which consequences and how to protect against it. To use such a system the user will have to significantly interpret the output. It may be a large and useful repository of hazard information but based on the information given in the paper it seems unlikely to be a very useful aid for HAZOP studies.

The HAZOP method uses a mixture of top-down and bottom-up techniques to determine the causes and consequences of a guide-word event such as "more-flow". However, neither of these techniques is used exhaustively or in a formal manner. Some interactions are left to be explored at later stages in the study and some are recognized to have already been explored. Nevertheless, simulation, or bottom-up reasoning, is a major part of the study and cannot be ignored in attempts at full or partial automation. For this reason the applications of Qualitative Simulation to chemical engineering will be reviewed at a later stage in this section. In addition, there is a database dimension to HAZOP studies—both to present background process engineering knowledge and to present knowledge about the particular design concerned. The former type will currently be in the form of reference books and company design manuals, which can be consulted during the study for quick checks. The latter will be in the form of manuals for the design being considered, as well as in the heads of the study team members.

4.3 Representing designer's intention

The use of databases in process engineering has already been discussed in section 2 but here we will review additional work that is concerned with using data-bases to represent and check constraints upon design. One of the most important concepts in the HAZOP method is the notion of *designer's intention* which is used to consider questions such as "Why did the designer choose to install this particular piece of equipment?" or "How has this equipment been sized?". If you do not know why something was done then it is very difficult to reason about whether or not it is correct. In addition, if you do not keep the designer's original specification and assumptions in mind then it is very easy to make later changes to the design that invalidate his work. These are often the kinds of problems that the HAZOP study is intended to identify by exploring in detail the interaction between different parts of the design. However, Waters et al. (1990) consider that the HAZOP study is rather a late stage to perform such checks, and describe the development of a tool to represent and check constraints continually, throughout the design process. In particular, they describe the use of an AI toolkit called Knowledge Craft to provide the following functionality:

1 Representation of process design knowledge in a frame-based manner.
2 Creation, display and editing of constraints.
3 Explicit constraint checking.
4 Automatic checking of selected constraints when the relevant information changes. Full control over which constraints are automatically checked for which objects.
5 Perhaps provide some simulation facilities to help check constraints.

The constraints can be very general and on very general objects, such as constraints on an entire separation train, or they can be very specific and attached to particular plant items, such as a constraint on the flow-rate in a particular stream. The aim is to provide a tool that is intended to be part of a larger design environment and acts as a watchdog for that environment. It can be thought of as an automated check-list that is easy to specialise for a particular company, design type or design team.

The authors discuss the existence of a trade-off between writing complicated constraints that span several systems and writing simpler constraints but needing to do a lot of simulation to check them. In addition, they discuss the advantages of representing whole systems as objects or frames, as well as individual equipment items. This work has continued and is now being evaluated by its application to an undergraduate design project involving nine people. The paper cited is now somewhat out of date because some of the representations have changed and the simulation aspect has been played down in favour of enhancing the constraint functions. In addition, Knowledge Craft contexts are used to implement some of the ideas of hierarchical design (as discussed in section 2.4). However, a full review of the current status of this project is outside the scope of this review.

Waters et al. write constraints using LISP, with some special syntax to make it easier to access slot values. This gives the user expressive power but makes it difficult to use the constraints in generative mode for constraint satisfaction purposes. This is therefore different from the uses of constraints for AI planning or constraint propagation. In addition, it differs from the optimisation view of constraints (Sargent, 1989) in which most design constraints are not hard but just more or less heavily weighted and added into the objective function. This latter viewpoint is an appealing one but suffers from the drawback that all constraints must be expressible in numerical terms so that you can measure how well they are satisfied. It is our feeling that this restriction, together with the awkward syntax of most linear programming approaches, will severely limit what one is able to say about a design. In addition, we feel that some constraints will have to be hard, such as legal constraints, though we also admit that some must be flexible. Our approach to the latter is to allow the easy turning-on and turning-off of automatic checking for such constraints.

4.4 Qualitative simulation

We shall now turn our attention to qualitative simulation which has attracted some interest in the chemical engineering literature in recent years. In an already mentioned paper Waters and Ponton (1989) give a detailed description of their experiences with qualitative simulation and give a comparison of such bottom-up techniques with the top-down fault-tree synthesis methods. They begin by drawing the distinction between "quasi-steady-state" methods and "dynamic" methods of qualitative simulation. In the former category they place the work of DeKleer and Brown (1984), because their confluences are derived from steady state equations, whilst in the latter category they place the worth of Kuipers (1986) and of Forbus (1984). The distinction is important in chemical engineering because different processes react on widely different time-scales. For example, the fluid flow in a pipe changes almost instantaneously with the pressure drop across the ends whilst the temperature in a tank will take a much longer time to react to changes in the inlet temperature. The distinction also defines the information that you can obtain from a method and how it can be used. From a dynamic method you can obtain a qualitative plot of system behaviour versus time and can account for effects that depend upon history. In contrast, from a quasi-steady-state method you obtain a plot of the new system steady-state versus the magnitude of the continuous input deviation. This makes it very difficult to model situations where the history of the change is important. For example, it is difficult to capture the fact that the final pressure in a tank depends upon how the tank was vented when the level last changed.

The quasi-steady-state method appears to be more limited than a dynamic method but should be correspondingly simpler. In addition, it was noted that DeKleer's representation was the only one which was component-centred and so most easily used for simulating networks of devices. For these reasons, Waters and Ponton chose to use DeKleer's method for their qualitative simulation work. The resulting Prolog program extended the notion of landmark values to allow the use of binary relational facts to specify transition points for devices. Such facts were updated as transitions were modelled and were searched exhaustively to rule out infeasible transitions. This increased the reasoning power of the technique but unfortunately proved to be highly combinatorial. Indeed, it was felt that the high dimensionality of network problems would make all of the qualitative simulation techniques highly inefficient and probably limit their usefulness for such applications.

Dohnal (1987) also describes some work using DeKleer-type confluences and mentions two qualitative simulation programs. The first is written in prolog and can be used for the solution of qualitative non-linearities, for a limited number of confluences. The second is written in Fortran and can deal with an unlimited number of linear confluences. Presumably the difference is that in the former case the non-linear equations are treated in a piece-wise linear fashion by keeping track of transitions between regions. The limit on the number of confluences suggests some efficiency problems though it is not clear whether these are memory or time limitations. Dohnal describes the use of additional confluences to resolve qualitative ambiguities. These are derived from equations which would be redundant in a quantitative scheme—an idea that has also been introduced by Stephanopoulos (1987) in a paper which was described in section 3.

Dalle Molle et al. (1988) describe the application of Kuipers "QSIM" programs to some representative chemical engineering systems. Initially they discuss the modelling of first, second and third order systems with a single input and a single output (SISO systems). QSIM correctly generates all possible behaviours, and no others, for step inputs and ramp inputs to such systems. Next the authors discuss an example involving a hot/cold mixing tank. They demonstrate that the construction of a QSIM model by a straightforward transcription of the differential equations for this system results in unnecessary ambiguity. This is because of the loss of global information about parameter dependencies and is solved by the algebraic manipulation of the system equations to produce a redundant constraint. The authors then show that the QSIM algorithm correctly predicts off-set and overshoot when modelling a proportional feedback control loop around the tank example. Finally they mention that QSIM can also correctly predict the lack of off-set if a proportional-integral controller is used instead, but allude to further problems with trying to model such integral action. The QSIM models used in this work are constructed by hand and cannot be straightforwardly combined to model a network. Indeed, the algebraic manipulation needed for the tank example is not necessarily intuitive, which makes model building something of an art. The authors conclude that qualitative simulation shows promise but to scale-up the technology to larger and more complicated systems requires better techniques to handle intractable branching and to incorporate available quantitative and functional information. Such work is underway in Kuipers' group at Texas.

Joshi & Motard (1988) is proposing to use Forbus' QPE program but no information is available on how far this has progressed. In his proposal he reviews the three main qualitative reasoning formalisms (DeKleer's, Kuipers' and Forbus') and concludes that Forbus' is the most natural for representing chemical engineering knowledge. The proposed work is concerned with dynamic simulation in the early stages of design, before quantitative information is available, in order to assess controllability as early as possible. The proposal discusses QPE at length but does not give much chemical engineering detail or any QPE models specific to the chemical engineering domain.

Before we close this review of qualitative reasoning in chemical engineering it is worth mentioning the work of Mavrovouniotis & Stephanopoulos (1987, 1988) on Order of Magnitude (O(M)) reasoning. The idea is to extend the relational algebra notion of greater-than, equal-to or less-than to the 7 primitives:

much-smaller than
moderately-smaller than
slightly-smaller than
exactly-equal-to
slightly-larger-than
moderately-larger-than
much-larger-than

In the paper in AAAI-87 the authors compare their work to the previous work of Raiman (1986), which they consider to suffer from several drawbacks such as the lack of a concrete semantics and the inability to reason about numerical information. In the paper in Computers & Chemical Engineering they discuss the application of O(M) reasoning to some process engineering and bio-engineering problems. They consider that the O(M) formalism can be used to bridge the gap between qualitative reasoning with signs and full quantitative reasoning with exact numbers. They point out that in the real world there are always many positive and negative effects on any aggregate result. From this they conclude that reasoning techniques should concentrate on deciding which effects are important before attempting to determine the sign of the overall result.

4.5 Operating procedure synthesis

We have now covered HAZOP, Qualitative Simulation and the recording of constraints upon design and will go on to consider automatic operating procedure synthesis, which has been tackled with reference to AI planning. ICI's Hazard Study 4 checks that all the recommendations from Hazard Study 3 have been carried out and that the plant operating procedures have been written. However, Hazard Study 3 requires a complete ELD which even shows lines and equipment only needed for start-up and shut-down. Consequently, operation must be partly considered in Hazard Study 3 and there is some confusion as to when operating procedures should be written. The provision of computer-aids for the synthesis of operating procedures would perhaps allow their earlier production, thus allowing them to be more formally considered during the HAZOP study. This could be the design usage of operating procedure synthesis; there is also an on-line usage, in order to plan for recovery from unexpected situations. At present, however, most of this work is directed towards design, which enables the researchers to side-step real-time considerations.

Fusillo and Powers have written several papers describing their attempts at automatic operating procedure synthesis (Fusillo & Powers, 1987; 1988a,b). In the second paper they describe the following strategy:

1 Decompose the system (plant section) into smaller sub-systems that can be physically isolated from each other.
2 Enumerate the constraints that the system is subject to.
3 Identify the goals to be satisfied.
4 Solve the goals by a double-backtracking planning routine.

The first two of these sub-problems are performed manually. The system decomposition is for efficiency purposes and is likened to "island driven planning". To facilitate this the decompositions should result in a system that can be brought close to its desired final state and held there. In addition, the decomposition facilitates modelling since the authors take a lumped-parameter approach and use a STRIPS like functional model. For example, they use a single value for the temperature of the fluid in a recycle loop and model the effect on this value as more heat is put into the loop. Thus they avoid the complication of modelling different temperatures at different parts of the loop.

The constraints divide into the following forms:

1 Global constraints that apply throughout the plant.
2 Local constraints that apply to individual pieces of equipment.

The former are typically constraints on streams, such as forbidding certain combinations of chemicals in order to prevent explosions, whilst the latter are more specific to a piece of equipment, often to prevent equipment damage. In the later paper (Fusillo & Powers, 1988b) they sub-divide local constraints into:

1 Pre-conditions on the equipment concerned.
2 Exogenous constraints that prevent damage to another piece of equipment and are derived (manually) by constraint propagation.

The operating goals are constructed by a means ends analysis to determine the difference between the initial state and the final, desired state. They can then be satisfied by the selection and ordering of operators to perform the necessary increase, decrease or other change. The goals are solved in parallel by an outer loop which generates a remaining goal to solve and an inner loop that generates operators to solve it. In the earliest reference (Fusillo & Powers, 1988a) the authors discuss the use of production rules to select potential operators.

Constraints are checked by a "simulation" after each operator selection, which determines the suitability of the choice. The backtracking algorithm should allow for all combinations and orders of operators to be attempted. No comments are made upon the efficiency of this.

The approach just described is perhaps limited by the weakness of the qualitative reasoning and by the requirement for a manual decomposition and constraint propagation. Some of these problems are addressed in the most recent paper (Fusillo & Powers, 1988b), in which the authors discuss the satisfaction of distributed, interacting goals and the use of implicit sub-goaling to relax constraints and patch plans. They describe the use of digraphs (influence graphs), to determine which constraints to relax, and the use of numerical routines, to derive the new constraint. In addition, they discuss other approaches to selecting constraints for relaxation, such as the use of priority weightings for constraints, but they have not implemented such schemes.

In a two-part paper Lakshmanan and Stephanopoulos describe a non-linear planning methodology for the synthesis of operating procedures. In part one (Lakshmanan & Stephanopoulos, 1988a) they review some of the previous planning work, particularly with regard to the modelling approaches used. They point out the deficiencies of the STRIPS approach when modelling distributed, network-like structures such as chemical plants. In particular, they point to the dependencies between different parts of a network, which make it very difficult to model a local effect without accounting for the state of the rest of the system. For example, we must make the outcome from opening a valve dependent upon whether or not there is fluid in the attached pipe and a pressure drop across it. Consequently, the authors consider the use of conditional operators but show that planning with them will be NP-hard. This justifies the adoption of a domain-specific approach to the planning of process operations. In their case it leads them to a hierarchical, distributed modelling structure (in order to capture enough domain knowledge) and the use of automatic constraint propagation (to represent the interactions between parts of the network).

The planning strategy is described in part two (Lakshmanan & Stephanopoulos, 1988b together with a case study which requires the determination of a sequence of valve actions that will regenerate the catalyst in a reactor. The non-linear planning methodology only manipulates constraints that are temporal in nature, i.e. that enforce one operation to be performed before another. The paper demonstrates how some of the operational constraints can be directly transformed into this form by the exploitation of domain specific features. In addition, they describe an additional planning phase which takes the partial plan from the non-linear planner and resolves remaining constraints by a systematic, linear, generate and test strategy. The papers decribe various algorithms for manipulating constraints and demonstrate that they are probably correct. The system has been programmed in KEE and interfaces to the DESIGN KIT system already described in section 2.4 (Stephanopoulos et al., 1987).

O'Shima (1987) and Foulkes et al. (1988) also describe computer-aids for operating procedure synthesis. However, rather than using a rich knowledge representation and applying techniques from AI planning these authors simply rely upon PROLOG type clauses to represent connectivity and other boolean information. Really they are addressing themselves to simpler problems that reduce to path-finding, such as finding all the valves that need to be opened in order to create a flow path between two points. Such approaches probably lack the modelling power needed to represent and check all plant constraints but could nevertheless provide useful tools for plant operation.

4.6 Conclusions

This concludes our review on the relevance of AI to Loss Prevention problems. Andow has given several previous reviews of this area (Andow & Galluzzo, 1987; Galluzzo & Andow, 1987; Andow,

1988) but these are now somewhat out of date. In particular, they tend to concentrate rather heavily on rule-based approaches rather than considering AI planning, qualitative simulation and frame-based plant representation.

If a single conclusion had to be drawn from the work described in this section it would be to recognize the central importance of qualitative representation and modelling to the largely symbolic problems of loss prevention. The work on qualitative simulation has shown some promise in capturing adequate knowledge but this seems to be off-set by the efficiency problems described. Thus a purely qualitative system will not be enough. In addition, the work on automatic operating procedure synthesis has identified the close relationship between modelling and planning. This leads to the conclusion that one cannot separate a planning algorithm from the modelling representation that it uses. Also, fault-tree synthesis relies upon qualitative knowledge to define the connections between the discretised events on the fault-tree. Finally, the constraints needed to represent designer's intention are also dependent upon the qualitative and semi-quantitative reasoning that can be performed to check them. Thus many of the loss prevention problems are dependent upon qualitative reasoning and further development of such is necessary. However, it is important that such development is conducted in parallel with that of the problem solving systems that depend upon it. The construction of monolithic qualitative reasoning programs is not necessarily the best way to proceed.

5 Conclusions and future trends

This has been a necessarily brief overview of what is a very wide field. We will now summarize the experience of AI applied to process design and operation, with particular emphasis on safety, and suggest likely future developments.

5.1 Design

Two conclusions emerge forcefully.

1 The range of knowledge required in process and plant design is extremely wide and its representation remains a major challenge for AI techniques in this field. No single methodology is appropriate and simplistic models, particularly rules, are of extremely limited use.
2 Much quantitative knowledge is encapsulated in the mathematical models now routinely developed for every chemical process in the course of its design. Rendering this knowledge explicit and capturing it in a form compatible with representations more suited to qualitative knowledge is probably the most important task for future work.

The development of design environments for integrated, incremental and hierarchical process and plant design is likely to have a major impact on the way design is carried out in industry in the very near future. These will make "one man and a workstation" design of a whole process possible in principle, if not necessarily desirable in practice.

5.2 Operation

Development in monitoring and fault diagnosis have had less impact than originally anticipated because of the very large software effort which has been required for even relatively modest projects. The eventual emergence of domain specific knowledge bases that are relevant to the general field of process operation, rather than application specific knowledge bases that are only relevant to a given process, should remedy this problem. However, this development presents a major challenge both to AI workers, because of the complexity and richness of the knowledge involved, and to process engineers because no systematized methodology of "operation" at present exists as it does with "design".

5.3 Safety and loss prevention

Although we have reviewed this area separately, safety is, most emphatically, not an additional activity to design or operation. It is in practice integral with all aspects of the process engineers work. Since design stage loss prevention is both time consuming and requires much decision making on qualitative judgements, this has to be a fruitful field for AI applications.

Earlier attempts to automate the work of hazard analysis and HAZOP studies have given way to the less ambitious, more realistic, and probably more ethical aim of providing "intelligent" aids in this field. We believe that this work will continue. It will become particularly important as integrated design becomes established practice. The use of smaller design teams, producing new designs faster, is one of the aims of current integrated design developments. It becomes even more crucial in these circumstances that safety considerations do not become short circuited, or the experience presently gained from the use of interdisciplinary HAZOP teams be lost. The continued presence of safety expertise can be assured by integrating the techniques described in section 4 with the design environments discussed in section 2.

References

Andow, PK, 1988, "The use of expert systems to improve process safety" *Proceedings of I. Mech. E. One Day Symposium on AI Applications in the Process Industries* Manchester.

Andow, PK and Galluzzo, M, 1987, "Process plant safety and artificial intelligence" *Proceedings of CEF-87 (The Use of Computers in Chemical Engineering)*.

Galluzzo, M and Andow, PK, 1987, "Expert systems in chemical engineering" *Proceedings of CEF-87 (The Use of Computers in Chemical Engineering)*.

Andow, PK and Lees, FP, 1975, "Process computer alarm analysis: Outline of a method based on list processing" *Trans. Instn Chem. Engrs.*

Baker-Counsell, J. 1985b, "Flowmeter selection: Expert help is on its way" *Process Eng. 66* (10) 71–73.

Banares-Alcantara, R, Ko, EI, Westerberg, AW and Rychener, MD, 1988, "DECADE—A hybrid expert system for catalyst selection—2. Final architecture and results" *Computers and Chemical Engineering* vol. 12 (9/10) pp 923–938, Pergamon Press.

Banares-Alcantara, R, Westerberg, AW, Ko, EI and Rychener, MD, 1987, "DECADE: a Hybrid Expert System for Catalyst Selection—1, Expert System Considerations" *Computers and Chemical Engineering 11* 267–277.

Beltramini, L and Motard, RL, 1988, "KNOD—A knowledge based approach for process design" *Computers and Chemical Engineering* vol. 12 (9/10) pp 939–958, Pergamon Press.

Berkovitch I and Baker-Counsell, J, 1985, "Proving the use of expert systems" *Process Eng.* 25–26.

Booth, SH and Jones, MC, 1987, "Use of rule based programming for operability studies" *Annual Research Meeting—University of Nottingham* I. Chem. E., Nottingham.

Best, RJ, Dhallu, NS and Johns, WR, 1987, "Enumerative and AI methods for process design" *Chem. and Industry* 510–515.

Branch, J, 1985, "Towards integrated process design" *Processing* 37–39.

Britt, HI, Smith, JA and Wareck, JS, 1989, "A computer-aided process synthesis and analysis environment" *Conference Proceedings: Foundations of Computer-Aided Process Design—Snowmass Village, Colorado, July 1989* Cache Corporation.

Brown, DC, 1984, "Expert systems for design problem solving, using design refinement with plan selection and redesign", *PhD Thesis*, Ohio State University.

Bunn, AR and Lees, FP, 1988, "Expert design of plant handling hazardous materials" *Chemical Engineering Research & Design 66.*

Cherry, DH, 1975, *PhD Thesis*, Cambridge.

CISHC (Chemical Industry Safety and Health Council), 1977, *A Guide to Hazard and Operability Studies*, Chemical Industries Association, London.

Conti, GAP, 1987, *PhD Thesis*, Cambridge.

Dalle Molle, DT, Kuipers, BJ and Edgar, TF, 1988, "Qualitative modelling and simulation of dynamic systems" *Computers and Chemical Engineering* vol. 12 (9/10) pp 853–866. Pergamon Press.

Daniell, J and Director, SW, 1989, "An object orientated approach to CAD tool control within a design framework" CMUCAD-89-15, Research Centre for Computer-Aided Design, Carnegie Mellon University.

DeKleer, J and Brown, JS, 1984, "A qualitative physics based on confluences" *Artificial Intelligence 24* 7–83.

Dohnal, M, 1987, "Naive models in chemical and system engineering" *Proceedings of CEF-87 (The Use of Computers in Chemical Engineering)*.

Douglas, JM, 1985, "A hierarchical decision procedure for process synthesis" *A.I.Ch.E. Journal 31* 353–362.
Duxbury, HA and Preston, ML, 1987, "The process systems contribution to process safety" *Conference Proceedings: Foundations of Computer-Aided Process Operation, 1987* Cache Corporation.
Eberts, RE and Lafayette, W, 1985, "Cognitive skills and process control" *Chem. Eng. Progress.*
Ferguson, GH, 1985, "The knowledge problem in artificial intelligence with special reference to hazard and operability studies" M.Sc. Project, Loughborough University of Technology, Loughborough, UK.
Forbus, KD, 1984, "Qualitative process theory" *Artificial Intelligence 24* 85–168.
Foulkes, NR, Walton, MJ, Andow, PK and Galluzzo, M. 1988, "Computer-aided synthesis of complex pump and valve operations" *Computers and Chemical Engineering* vol. 12 (9/10) pp 1035–1044, Pergamon Press.
Fusillo, RH and Powers, GJ, 1987, "A synthesis method for chemical plant operating procedures" *Computers and Chemical Engineering 11* (4) 369–382.
Fusillo, RH and Powers, GJ, 1988a, "Computer-aided planning of purge operations" *A.I.Ch.E. Journal 34* (4) 558–566.
Fusillo, RH and Powers, GJ, 1988b, "Operating procedure synthesis using local models and distributed goals" *Computers and Chemical Engineering* vol. 12 (9/10) pp 1023–1034, Pergamon Press.
Harris, DJ and Jones, MC, 1987, "Expert system shells in process safety assessment" *Annual Research Meeting—University of Nottingham* I. Chem. E., Nottingham.
Hoskins, JC and Himmelblau, DM, 1988, "Artificial neural network models of knowledge representation in chemical engineering" *Computers and Chemical Engineering* vol. 12 (9/10) pp 881–890, Pergamon Press.
I. Chem. E., 1977, *A First Guide to Loss Prevention* I. Chem. E., Rugby.
Joback, KG and Stephanopoulos, G, 1989, "Designing molecules possessing desired physical property values" *Conference Proceedings: Foundations of Computer-Aided Process Design—Snowmass Village, Colorado, July 1989* Cache Corporation, Austin, Texas.
Joshi, AP and Motard, RL, 1988, "Qualitative simulation of chemical engineering systems using qualitative process theory" Doctoral Proposal, Department of Chemical Engineering, Washington University, Missouri.
Kessler, MG and Parker, RO, 1969, "Optimal networks of heat exchangers" *Chem. Eng. Prog. Symp. Ser. 65* (92) 111–120.
Kirkwood, RL, Locke, MH and Douglas, JM, 1988, "A prototype expert system for synthesising chemical process flowsheets" *Computers and Chemical Engineering 12* 329–343.
Kletz, TA, 1983, "Hazop and Hazan—Notes on the identification and assessment of hazards" *I. Chem. E. Workshop Modules* I. Chem. E., Rugby.
Kramer, MA, 1987, "Expert systems for process fault diagnosis: A general framework" *Conference Proceedings: Foundations of Computer-Aided Process Operation, 1987* Cache Corporation.
Kuipers, B, 1986, "Qualitative simulation" *Artificial Intelligence 29* 289–338.
Lakshmanan, R and Stephanopoulos, G, 1988a, "Synthesis of operating procedures for complete chemical plants—1. Hierarchical, structured modeling for nonlinear planning" *Computers and Chemical Engineering* vol. 12 (9/10) pp 985–1002, Pergamon Press.
Lakshmanan, R and Stephanopoulos, G, 1988b, "Synthesis of operating procedures for complete chemical plants—2. A nonlinear planning methodology" *Computers and Chemical Engineering* vol. 12 (9/10) pp 1003–1022, Pergamon Press.
Lang, Y-D, Biegler, LT and Grossman, IE, 1988, "Simultaneous optimisation and heat integration with process simulators" *Computers and Chemical Engineering 12* (4) 311–328.
Lapp, SA and Powers, GJ, 1977, "Computer-aided synthesis of fault-trees" *IEEE Transactions on Reliability R-26* 2–11.
Lien, K, Suzuki, G and Westerberg, AW, 1986, "The role of expert systems technology in design" Carnegie Mellon University, Engineering Design Research Center, EDRC-06-13-86.
Leitch, RR, 1988, "A task-orientated architecture for process control applications" *Proceedings of IEE Colloquium—Qualitative Modelling in Diagnosis and Control.*
Liles, JA, Lott, DH and Preston, ML, 1988, "Process definition to plant design: Integration at last?" *Proceedings of PSE'88* pp 117–124, Inst. Eng. Australia.
Linnhoff, B and Flower, JR, 1978, "Synthesis of heat exchanger networks I. Systematic generation of optimal networks" *A.I.Ch.E. Journal 24* 633–642.
Linnhoff, B, Townsend, DW, Boland, D, Hewitt, GF, Thomas, BEA, Guy, AR and Marsland, RH, 1982, *A User Guide for the Efficient Use of Energy* I. Chem. E.
Martin-Solis, GA, Andow, PK and Lees, FP, 1982, "Fault tree synthesis for design and real time applications" *Transactions of the I.Chem.E 60* 14–25.
Masso, AH and Rudd, DF, 1969, "Synthesis of system designs II. Heuristic structuring" *A.I.Ch.E. Journal 15* (1) 10–18.
Matthews, B, Lindenfelzer, P, Hawkinson, L and Moore, R, 1988, "Process control with the G2 real-time expert system" *Proceedings of the First International Conference on Industrial and Engineering Applications of Artificial Intelligence & Expert Systems—Tullahoma, Tennessee, 1988.*

Mavrovouniotis, ML and Stephanopoulos, G, 1987, "Reasoning with order of magnitude and approximate relations" *Proceedings of AAAI–87.*

Mavrovouniotis, ML and Stephanopoulos, G, 1988, "Formal order of magnitude reasoning in process engineering, *Computers and Chemical Engineering* vol. 12 (9/10) pp 867–880, Pergamon Press.

May, D and Rudd, DF, 1976, "Development of solvay clusters of chemical reactions" *Chem. Eng. Science 31* 59–69.

Milne, R, 1989, "Monitoring process control systems" *Workshop on AI in Process Engineering—Preprints of one-day workshop at the Artificial Intelligence Applications Institute, Edinburgh.*

Munro, N, 1987, "ECSTACY—An environment for control system design" internal report, University of Manchester Institute of Science & Technology (UMIST), UK.

Myers, DR, Davis, JF and Herman, DJ, 1988, "A task-orientated approach to knowledge-based systems for process engineering design" *Computers and Chemical Engineering* vol. 12 (9/10) pp 959–972, Pergamon Press.

Niida, K, Itoh, J, Umeda, T, Kobyashi, S and Ichikawa, A, 1985, "Some expert system experiments in process engineering" *I.Chem.E. Symposium Series* vol. 92, *Process Systems Engineering '85* Pergamon Press.

O'Shima, E, 1987, "Sequencing and scheduling of plant operations" *Conference Proceedings: Foundations of Computer-Aided Process Operation,* Cache Corporation.

Park, SW and Himmelblau, DM, 1987, "Structural design for systems fault diagnosis" *Computers and Chemical Engineering 11* (6) 713–722.

Peel, D and Morris, AJ, 1989, "Towards knowledge-based process control" *One Day Workshop on AI in Process Engineering* Artificial Intelligence Applications Institute, Edinburgh.

Ponton, JW, Donaldson, RAB, Johns, WR and Romero, D, 1979, "The automatic generation and evaluation of process flowsheets" *Computers and Chemical Engineering 3* 251–260.

Ponton, JW, Hutton, D, Jones, G and Skilling, N, 1989, "A demonstration 'intelligent' physical properties package" *Proceedings of Chempor '89* Lisbon, 1989.

Preston, ML and Ponton, JW, 1988, "ICI process engineering expert systems—Present & future" *Proceedings of AIChE National Meeting,* Spring 1988.

Raiman, O, 1986, "Order of magnitude reasoning" *Proceedings of AAAI-86* pp 100–104, American Association for Artificial Intelligence.

Ramesh, TS, Shum, SK and Davis, JF, 1988, "A structured framework for efficient problem solving in diagnostic expert systems" *Computers and Chemical Engineering* vol. 12 (9/10) pp 891–902, Pergamon Press.

Rich, SH and Venkatasubrãmanian, V, 1987, "Model-based reasoning in diagnostic expert systems for chemical process plants" *Computers and Chemical Engineering 11* (2) 111–122.

Rijckaert, MJ and Bogaerts, W, 1987, "Expert systems for materials selection and process control" *Conference Proceedings: Foundations of Computer-Aided Process Operation,* Cache Corporation 687–694.

Roach, JR and Lees, FP, 1981, "Some features of and activities in hazard and operability (HAZOP) studies" *The Chemical Engineer,* October, 1981.

Russell, LW and Perkins, JD, 1987, "Towards a method for diagnosis of controllability and operability problems in chemical plants" *Chemical Engineering Research & Design 65.*

Sargent, RWH, 1989, "Process design—What next?" *Conference Proceedings: Foundations of Computer-Aided Process Design—Snowmass Village, Colorado, July 1989* Cache Corporation.

Sargent, RWH and Stephanopoulos, G, 1989, "AI in chemical engineering—informal discussion session summary" *Conference Proceedings: Foundations of Computer-Aided Process Design—Snowmass Village, Colorado, July 1989* Cache Corporation.

Shafaghi, A, Andow, PK and Lees, FP, 1984, "Fault tree synthesis based on control loop structure" *Chemical Engineering Research and Design 62* 101–110.

Stephanopoulos, G, 1987, "The scope of artificial intelligence in plant-wide operations" *Conference Proceedings: Foundations of Computer-Aided Process Operation,* Cache Corporation.

Stephanopoulos, G, 1988, "Autonomy in process control systems" *Proceedings of One Day International Symposium on Advanced Process Supervision and Real-time Knowledge Based Control* Newcastle.

Stephanopoulos, G, Johnston, J, Kriticos, T, Lakshmanan, R, Mavrovouniotis, ML and Siletti, C, 1987, "Design-kit: An object-oriented environment for process engineering" *Computers and Chemical Engineering* vol. 11 (6) pp 655–674, Pergamon Press.

Stephanopoulos, G and Townsend, DW, 1985, "Synthesis in process development: Issues and solution methodologies" *I. Chem. E. Symp. Series 92—Process Systems Engineering '85,* Pergamon Press.

Suokas, J, Heino, P and Karvonen, 1987, "The development of an expert system to support Hazop analysis" in: *Reliability '87—Proceedings,* Birmingham.

Talukdar, S and Westerberg, AW, 1988, "A view of next generation tools for design" EDRC 05-17-88, Engineering Design Research Centre, Carnegie Mellon University.

Taylor, JR, 1982, "An algorithm for fault-tree construction" *IEEE Transactions on Reliability R-31* (2) 137–146.

Thompson, RW and King, CJ, 1972, "Systematic synthesis of separation schemes" *A.I.Ch.E. Journal 18* (5) 941–948.

Venkatasubramanian, V and Dhurjati, P, 1987, "Towards an object-orientated diagnostic methodology for the expert system FALCON" *Conference Proceedings: Foundations of Computer-Aided Process Operation,* Cache Corporation.

Venkatasubramanian, V and Rich, SH, 1988, "An object-orientated two-tier architecture for integrating compiled and deep-level knowledge for process diagnosis" *Computers and Chemical Engineering* vol. 12 (9/10) pp 903–922, Pergamon Press.

Waters, A, Chung, PWH and Ponton, J, 1990, "Representing safety constraints upon chemical plant designs" *Expert System Applications,* ed. Vadera S, Sigma Press, Wilmslow, England, 1990.

Waters, A and Ponton, J, 1989, "Qualitative simulation and fault propagation in process plants" *Chemical Engineering Research and Design 67* 407–422.

Weatherill, T and Cameron, I, 1988, "Preliminary HAZOP studies using expert systems" in: *Proceedings of the Third International Symposium on Process Systems Engineering* (PSE '88), Sydney Inst. Eng. Australia.

Wells, GL, 1980, *Safety in Process Plant Design* George Godwin Ltd, London.

Westerberg, AW, Hutchinson, HP, Motard, RL and Winter, P, 1979, *Process Flowsheeting* Cambridge University Press.

Westerberg, AW, Piela, PC, Subrahmanian, E, Podnar, G and Elm, B, 1989, "A future computing environment for preliminary design" *Conference Proceedings: Foundations of Computer-Aided Process Design—Snowmass Village, Colorado, July 1989* Cache Corporation.

Winter, P and Rosen, EM, 1986, "Process engineering databases" *Chemical Engineering 93* (13) 65–76.

Yamashita, Y, and Motard, RL, 1986, "Object oriented integration of process engineering computing" *Paper presented at AIChE Spring Meeting* New Orleans.