Synthesis of Failure-Safe Operations

J. ROBERTO RIVAS

Instituto Tecnologico y de Estudios Superiores de Monterrey, Monterrey, N.L. Mexico

and

DALE F. RUDD

Department of Chemical Engineering and Mathematics Research Center University of Wisconsin Madison, Wisconsin 53706

Methods are developed for the computer-aided synthesis of sequences of valve operations to reach complex operation goals with safety. Given dangerous events which must not occur and operation goals to be reached, sequences of valve openings and closings are formed rapidly for industrially significant problems.

SCOPE

The engineer is guided by experience and intuition during the synthesis of failure-safe operating instructions, and these methods cannot be automated directly. Especially during emergency situations there is a need for computer assistance to speed up and increase the accuracy of the synthesis. This is the first step taken in that direction.

The process is modeled as a network of connectors through which material is driven by pressure differences guided by the position of valves. The variety of hazardous conditions to be avoided are given, along with general statements of operating objectives. The synthesis involves the sequencing of valve opening and closings to reach the operation objectives while avoiding the hazards. This forms an enormous combinatorial problem, similar to that found in games such as chess.

Synthesis is accomplished by the formation of a hierarchy of goals which identify critical operations and the order in which they are to be performed. Industrially significant problems involving a score or more operations can be handled rapidly and accurately.

CONCLUSIONS AND SIGNIFICANCE

This work is significant in that it deals with an important field of engineering problem solving that had received little or no systematic study. It has now been demonstrated that the difficult problems in inductive reasoning associated with operation instruction synthesis can be formulated and solved. Optimism is warranted in the generality and usefulness of this approach.

The fields of artificial intelligence and pattern recognition have reached the point where the computer can perform tasks which might be thought to require human intelligence (Newell and Simon, 1971; Uhr, 1973). The inductive reasoning used in the synthesis of complex plans of action, such as the development of failure safe operations, in the past has been completely dominated by the human mind. We present our latest developments in the use of the computer in the synthesis of intelligent appearing plans of action (Masso and Rudd, 1969; Lee, Masso, and Rudd, 1970; Siirola, Powers, and Rudd, 1971; Siirola and Rudd, 1971).

The computer-aided analysis of proposed industrial operations yielded to sequential logic, leading to the practical solution of the enormous combinatorial problems arising in safety interlock system design (Rivas and Rudd, 1974). Given a proposed sequence of valve operations, the computer can perform the bookkeeping required to detect hazardous operations at speeds sufficient to monitor industrially large processes. However, an equally important and intellectually different problem is the synthesis or creation of the operating procedure rather than analysis. Analysis requires deductive reasoning and synthesis appears to require a high level of inductive reasoning.

We draw a parallel between the classic test case of artificial intelligence, the computer playing of chess, and the problem of concern here, the synthesis of failure safe operations. Chess pits two minds against each other in a situation so complex that neither can understand it completely, yet the game is sufficiently well defined and the rules so simple that a beginner can follow the play of world champions. The synthesis of large industrial operations takes on many of the troublesome characteristics of a chess game.

Claude Shannon in 1949 proposed that the computer could be programmed to play chess merely by the rapid look-ahead of proposed continuations of play to a depth sufficient to identify probable winning moves. The examination of all continuations is not feasible since there are something like 10^{120} possible continuations and only 10^{16} microseconds in a century to explore them. A sequence of valve operations corresponds to a continuation of play in chess, and in large industrial operations the combinatorial problems are sufficiently difficult to prevent the examination of even a small fraction of all possible operations in the search for failure safe operations. The skilled chess

---

Correspondence concerning this paper should be addressed to D. F. Rudd. Work performed at the University of Wisconsin with the support of the National Science Foundation, Shell Oil Foundation and the United States Army.

player and the skilled process engineer have much in common in their intuitive ability to create useful continuations.

Shannon (1949), Turing (1950), Kister et al. (1956), Berstein et al. (1957), Newell, Shaw, and Simon (1958), Botvinnik (1970) and others have shown the computer how to play fairly good chess. It would be extremely useful to industry if the computer could be shown how to synthesize safe operations with equal skill.

The nomenclature, methods of analysis, and the test processes used here are presented in the report on Computer-aided Safety Interlock Systems (Rivas and Rudd, 1974) and the reader should be familiar, at least superficially, with that work. In the next section we describe a simple problem in the synthesis of failure-safe operation to introduce the concept of a goal hierarchy. Then we introduce the formal problem of developing a synthesizer capable of industrially significant problems.

SYNTHESIS OF GOAL HIERARCHY

We begin with a simple example of operation procedure synthesis to ease comprehension of the principles developed. In this example, Figure 1, hydrogen flow is to be initiated through a system initially filled with air. The valving sequence sought is one in which air and hydrogen are not in contact. Below is a summary of the mental processes we use to solve this problem intuitively.

> The only difference between the initial and final valve positions in Figure 1 are the positions of valve 1, valve 3 and valve 5. However, the operating procedure [open 1, close 3, open 5] is decidedly hazardous. The operation objective, "initiate hydrogen flow to the low pressure outlet" is too far removed from the identification of valving sequences. We need a sequence of less ambitious goals to bridge the gap.
>
> "Evacuate Air" might be a reasonable place to begin. To reach this goal we must reach these two goals in this sequence "Stop air flow" and "open system to low pressure outlet." "Stop air flow" is accomplished by [close 3] and "open system to low pressure outlet" by [open 5].
>
> Now the goal "Initiate hydrogen flow" points directly to the operation [open 1]. In summary, a safe operation sequence is [close 3, open 5, open 1].

We now examine the details of this intuitive synthesis of operation procedures. The original goal, the operation objective "Initiate hydrogen flow to low pressure outlet" drew attention to places in the system where hydrogen was present, namely above valve 1. However, any changes in valve 1 brought air and hydrogen in contact. Thus, the only hint of where to begin, the word *hydrogen* in the original goal brought us to a dead-end, see Table 1.

However, attempts to use the original goal brought up the word *air,* and this suggested a level I goal "Evacuate air." But the opening of valve 5 to evacuate air merely opens a path through the system for air to flow. The inability to reach the level I goal suggested level II goals of "stop air flow" and "open system to low pressure outlet" which identified valid valve operations.

Table 1 shows the penetration into the operation procedure synthesis problem by a hierarchy of goals until the point is reached where unique valving operations are identified. It appears then that the cutting-edge in the tools of synthesis is a hierarchy of goals designed to slice into the problem to the depth of individual valve operations. This is the direction we take in this report.

COMPLEXITY AND COMMUNICATION

There is a direct and inverse relation between the complexity of a concept and the communication of the concept. A computer which understands the manipulation of valves might not be able to comprehend a complex concept such as "initiate hydrogen flow to the low pressure outlet" in the same way we did not initially comprehend the same concept in the previous section. To solve that communication problem—transferring information from the part of our mind concerned with the operation objective to that part concerned with legal valve operations—we reduced the complexity of the concept by forming a sequence of simpler concepts which are bite-size.

If we are forced to go all the way to single valve manipulation concepts to communicate our operation goal to the computer, nothing has been gained. The generation of the concept or goal sequences is the intellectual challenge. However, if the computer can be taught to catch on to our operation concept well before the valve operation level is reached, a useful capability has been gained. Thus, a measure of our success is ease by which the computer grasps the operation concept we have in mind as we attempt to reduce the complexity of the concept by a sequence of operation goals.

Figure 2 shows the test case used here. The operation objective, or goal at level 0, is

> Level 0 goal
> remove the reactor from service, regenerate it, and place it back in service

This concept is too complex to communicate to an experienced engineer, much less to a computer. Thus, we replace it by a sequence of sixteen less complex concepts which can be understood by an engineer experienced with this catalyst regeneration system. The level I goals are shown in the left-hand column of Table 3. How much more explicit must we be for the computer to catch on? We must express the goals as Boolean statements in symbolic logic.

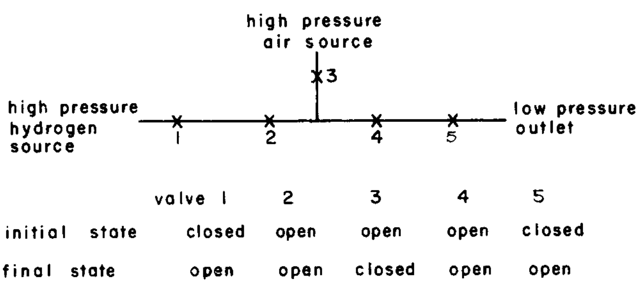

Fig. 1. Initially the system is filled with air. The operation objective is the initiation of hydrogen flow to the low pressure outlet. The safety criterion is that hydrogen and air must not be in contact.

TABLE 1. THE HIERARCHY OF GOALS TO IDENTIFY A SAFE OPERATION PROCEDURE FOR THE SYSTEM IN FIGURE 1

Level 0 goal	Level I goal	Level II goal	Valve operation
Initiate hydrogen flow to low pressure outlet	Evacuate air	Stop air flow	Close 3
		Open system to low pressure outlet	Close 5
	Initiate hydrogen flow		Open 1

Final sequence [close 3, open 5, open 1]

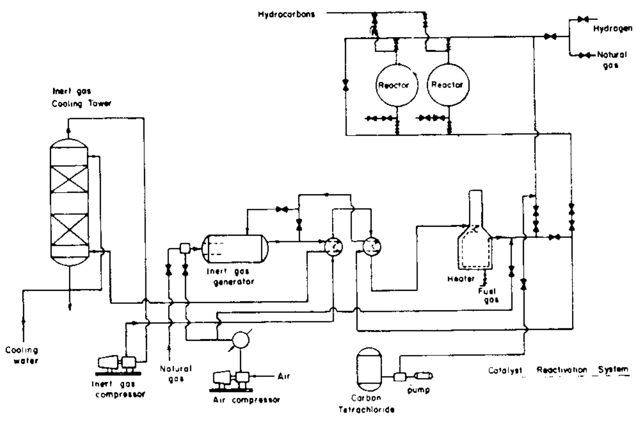

Fig. 2a. Catalyst reactivation system.

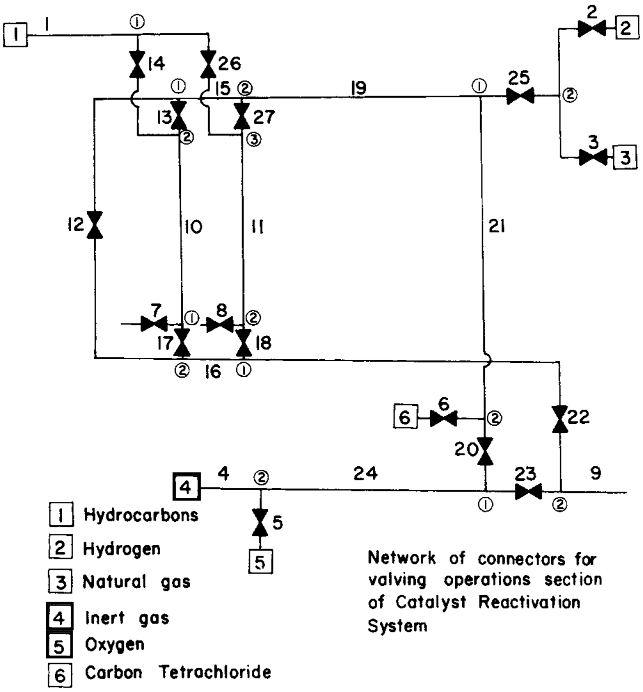

Fig. 2b. Network of connectors for valving operations section of catalyst reactivation system.

The current Failure Safe Operation Synthesizer programs require a set of level II goals defined by these propositions

BTL(n, z): Species AS(n, z) is to flow at side FN(n, z) of connector EC(n, z)
CIL(n, z): Species AS(n, z) is to be blocked from an inlet at side FN(n, z) of connector EC(n, z)
DSL(n, z): Species AS(n, z) is to be trapped at side F(n, z) of connector EC(n, z)

where the index z is a goal set identification and the index n denotes members of the set. Table 2 shows how combinations of these logic propositions define eight types of level II goals.

Table 3 shows the transformation of the level I goals into the level II goals for the Catalyst Regeneration System.* This is the depth to which the engineer must penetrate into the problem during the development of the goal hierarchy. The complexity of the synthesis problem must be reduced to this level before communication can be made with the computer at the current state of the art.

Table 2. Meaning of the Combination of Values of the Variables BTL(n, z), CIL(n, z), DSL(n, z)

BTL(n, z)	CIL(n, z)	DSL(n, z)	Meaning
1	0	0	Species AS(n, z) is to be flowing at side FN(n, z) of connector EC(n, z)
0	1	0	Coming from an inlet only
0	0	1	Trapped
1	1	0	Flowing or coming from an inlet
0	1	1	Coming from an inlet or trapped
1	0	1	Flowing or trapped
1	1	1	Present in any state
0	0	0	Not present at all

SYNTHESIS OF CONTINUATIONS

The original operation goal has been replaced by a set of z verbal goals at level I, and each verbal goal replaced by a set of n logic propositions at level II. We now develop principles for scanning the level II goals to identify candidate valve operations or to identify the need for the synthesis of level III goals. The synthesis of these continuations of action takes the form of process operation instructions and/or the penetration further into the problem by the formation of a new hierarchy of goals.

The set of verbal goals are examined in the order provided by the engineer since the precedence ordering at level I often is critical. Attention then focuses on the n logic propositions which compose the z*th* verbal goal. We seek the location of connectors in the system where a valve action is positively related to any one of the set of n level II goals (or any of the level III goals which may have been generated earlier by the computer, see below).

The valve on connector R is positively related to the operation objective if, at levels II and higher,

A there exists an unmet goal that requires the flow of a species at a site: and the valve action causes that flow to occur or causes the species to be newly blocked at the site

or *B* there exists an unmet goal that requires the blockage of a species at a site: and the valve action causes the blockage or causes the material now to appear at that site by flow.

or *C* there exists an unmet goal that requires the trapping of a species at a given site: and the valve action causes the trapping or causes the material now to appear newly at the site by flow.

or *D* there exists an unmet goal that requires the evacuation of a site: and the valve action causes the evacuation of the site or traps material currently blocked there.

or *E* if the connector is empty.

These conditions which define a positive relation to the goals examine not only the immediate goal achievement but also look ahead to events which may lead to eventual goal achievement. For example, condition A deals with the need for material flow but also involves the development of a blocking condition, which is one way of setting up a sequence of valve operations to cause flow. Figure 3

* The goals in Table 3 marked with an asterisk are not members of the minimum set and are part of the advice taking capability discussed in the next section.

TABLE 3. SETS OF GOALS FOR THE CATALYST REACTIVATION SYSTEM

Level I goals Verbal description	z	n	Level II goals AS(n, z)	FN(n, z)	EC(n, z)	BTL(n, z)	CIL(n, z)	DSL(n, z)
Replace reactor in service and stop hydrocarbon flow in the reactor to be reactivated	1	1	1	2	11	1	0	0
		2	1	1	10	0	0	0
Purge hydrocarbon from reactor with hydrogen	2	1	2	1	10	1	0	0
Pressurize reactor with hydrogen	3	1	2	1	10	0	0	1
Depressure and evacuate hydrogen from reactor	4	1	2	1	10	0	0	0
Pressurize reactor with inert gas	5	1	4	1	10	0	1	0
Circulate inert gas	6	1	4	1	10	1	0	0
	*6	2	4	1	22	1	0	0
Divert total flow of inert gas to reactor	7	1	4	1	23	0	1	0
	7	2	4	1	12	0	1	0
Chlorinate and rejuvenate	8	1	5	1	10	1	0	0
	8	2	6	1	10	1	0	0
Pressurize reactor with inert gas, chlorine and oxygen	9	1	4	1	23	1	0	0
	9	2	4	1	10	0	0	1
	9	3	5	1	10	0	0	1
	9	4	6	1	10	0	0	1
	*9	5	4	1	19	0	0	1
	*9	6	5	1	19	0	0	1
	*9	7	6	1	19	0	0	1
Depressurize and evacuate reactor	10	1	4	1	10	0	0	0
	10	2	5	1	10	0	0	0
	10	3	6	1	10	0	0	0
	*10	4	4	1	19	0	0	0
	*10	5	2	2	25	0	0	0
Pressurize with natural gas	11	1	3	1	10	0	1	0
Purge with natural gas	12	1	3	1	10	1	0	0
Stop flow of natural gas	13	1	3	1	10	0	0	0
Purge with hydrogen	14	1	2	1	10	1	0	0
Pressurize with hydrogen	15	1	2	1	10	0	1	0
Depressurize and evacuate hydrogen, start hydrocarbon flow in reactivated reactor, stop hydrocarbon flow in the other reactor and leave hydrogen in the upper header	16	1	2	1	10	0	0	0
	16	2	1	1	10	1	0	0
	16	3	1	2	11	0	0	0
	16	4	2	1	13	0	1	0

TABLE 4. ITERATIONS FOR SYNTHESIZING OPERATION SEQUENCE FOR LEVEL I, GOAL 1 IN TABLE 3. EDITED TO INCLUDE ONLY MAJOR EVENTS

Positively related connector	Interlock test	Possible operation	Level I goal reached
13	Hazardous		
14	Hazardous		
17	Hazardous		
26	Safe	Open valve 26	Not yet
27	Hazardous		
8	Safe	Open valve 8	Not yet
13	Hazardous		
14	Safe	Close valve 14	Yes

Final sequence (open 26, open 8, close 14).

shows how this transition from goal hierarchy generation to valve evaluations is accomplished by such operation sequence look-ahead.

Figure 4 shows the sequence of events which lead to the generation of failure safe operation instruction. We now examine this in outline. The existing Operation Analysis Program and the Adaptive Safety Interlock System developed previously are used in the system analysis parts

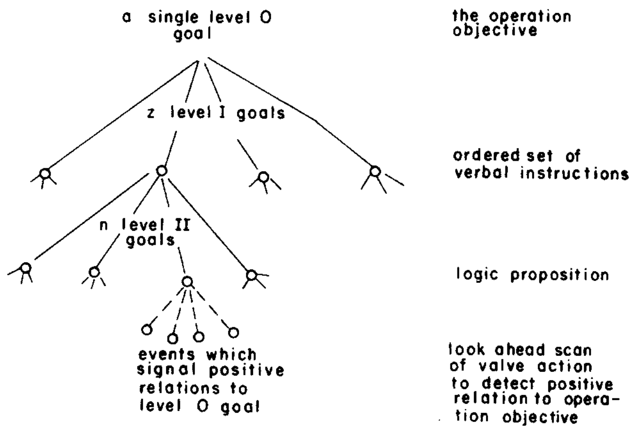

Fig. 3. The logic structure of goal hierarchy generation and evaluation. The upper levels involve goal generation and the low levels involve look ahead evaluations of material movement initiated by candidate valve operations. The generation of level III goals and the receipt of advice is not included in this representation.

of this program (Rivas and Rudd, 1974).

The first several operations in Figure 4 merely keep track of the progress. Given a set of n level II or higher goals to be reached, single valve changes are proposed and the changes in the system analyzed. One of three things

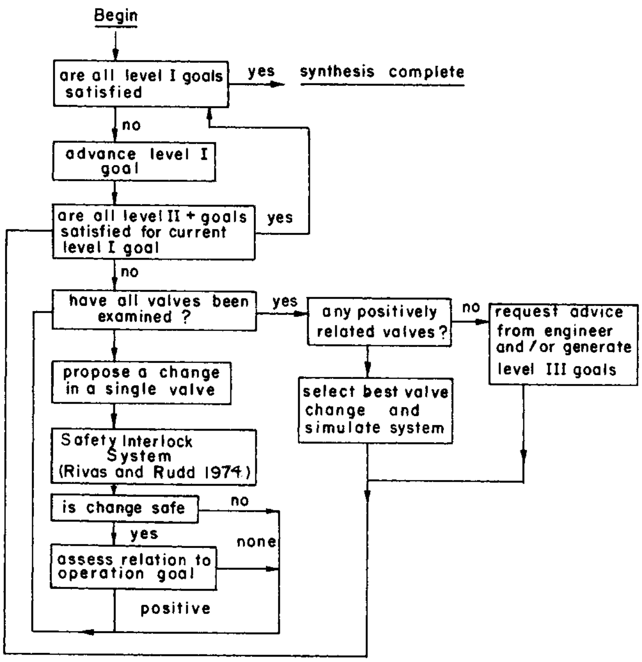

Fig. 4. Broad outline of Failure Safe Operation Synthesis Program. The actual program differs only in detail from this general plan.

can occur, the change causes a hazardous condition; the change is not related to any of the goals or the change is positively related to one or more of the goals.

If none of the proposed changes are positively related to any of the goals, advice is requested from the engineer and/or level III goals are generated. This action is discussed in the next section. Otherwise, the valve change which is the most positively related to the goals is implemented as part of the evolving operation sequence. The definition of *most positively related* is examined in the section following on Operation Criteria, and for the time being we select any positively related valve.

Table 4 shows the progress towards the first of the level I goals shown in Table 3. Included are only the major events, and the detailed cycling through the algorithm is omitted. Table 5 shows the complete sequence of valving operations synthesized by the computer and also includes the sequence created by the engineers of a major petroleum processing organization. The differences between these operation procedures result from the operation criteria used. At the present state, the computer seeks only failure-safe operations, a criterion which does not lead to a unique set of operations. Of particular interest is that only one minute and thirty-four seconds of UNIVAC 1108 computer time was needed to complete the synthesis of the operating instruction for this industrially significant process. For the more elaborate Hydrogen Drying System, not included here, two minutes and twenty-six seconds were required.

TABLE 5. SUMMARY OF ENGINEER AND COMPUTER SYNTHESIZED OPERATION PROCEDURES FOR THE CATALYST REACTIVATION SYSTEM

Goal	Engineer	Computer
Replace reactor in service and stop hydrocarbon flow in the reactor to be reactivated	Open valve 26 Open valve 8 Close valve 14	Open valve 26 Open valve 8 Close valve 14
Purge hydrocarbon from reactor with hydrogen	Open valve 13	Close valve 7 Open valve 13 Open valve 7
Pressurize reactor with hydrogen	Close valve 7 Close valve 25 Close valve 2	Close valve 7 Close valve 13
Depressure and evacuate hydrogen from reactor	Open valve 12 Open valve 17 Open valve 7	Open valve 7
Pressurize reactor with inert gas	Close valve 7 Open valve 20	Close valve 25 Open valve 13 Close valve 7 Open valve 20
Circulate inert gas	Open valve 22	Open valve 12 Open valve 22 Open valve 17
Divert total flow of inert gas to reactor	Close valve 23 Close valve 12	Close valve 12 Close valve 23
Chlorinate and rejuvenate	Open valve 5 Open valve 6	Open valve 5 Open valve 6
Pressurize reactor with inert gas, chlorine and oxygen	Open valve 23 Close valve 22 Close valve 20 Close valve 5 Close valve 6	Open valve 23 Close valve 13 Close valve 17 Close valve 20 Close valve 6
Depressurize and evacuate reactor	Open valve 7	Close valve 2 Open valve 7 Open valve 13 Open valve 25
Pressurize reactor with natural gas	Open valve 25 Close valve 7 Open valve 3	Open valve 3 Close valve 7
Purge reactor with natural gas	Open valve 7	Open valve 7
Stop flow of natural gas	Close valve 3	Close valve 3
Purge with hydrogen	Open valve 2	Open valve 2
Pressurize with hydrogen	Close valve 7	Close valve 7
Depressurize and evacuate hydrogen, start hydrocarbon flow in reactivated reactor, stop hydrocarbon flow in the other reactor and leave hydrogen in the upper header	Close valve 13 Open valve 7 Open valve 14 Close valve 26 Close valve 8	Close valve 2 Open valve 7 Open valve 2 Close valve 7 Close valve 13 Open valve 14 Open valve 7 Close valve 26
Total number of operations	35	43

ADVICE TAKING BY LEVEL III GOALS

It is possible that convergence to a failure safe sequence of operations does not occur, given the minimum set of level II logic proposition goals. This happens frequently when the system is filled with hazardous material and the goals do not explicitly mention the removal of the hazardous material. The Safety Interlock System then blocks every move the synthesizer proposes. With only the minimum set of goals, the synthesizer has no hint of the importance of a prior purge operation sequence. Other, more complicated examples of situations not involving hazards which stymie the synthesizer can be cited, and these occur in the test problems we examined.

The essential feature is that the synthesizer must be drawn away from the most direct appearing route to the goals and forced to examine round-about avenues which may involve valving operations which are not positively related to the minimum set of level II goals. The synthesizer seeks advice in the form of level III goals. The level III goals point to regions of operation which the synthesizer would normally not examine. Certain of the level III goals are generated automatically by the computer, and others are sought as advice from the engineer.

Should the synthesizer discover that all its proposed operations are hazardous and the hazards involve a species which does not enter into the minimum set of level II goals, the computer will generate its own level III goal of purging that species from the system. These level III goals are added to the existing level II goals and synthesis is attempted. New purge goals might then arise resulting in a layering of level III goals until synthesis is accomplished or this route to convergence fails. The computer successfully generated level III goals in both the Catalyst Regeneration System and the Hydrogen Dryer examples.

The engineer is a second source of level III goals, as he offers advice to the computer. In Table 3 the goals with the asterisk are really level III goals provided by the engineer. These particular goals were added in anticipation of problems the computer might have, and others are added when the difficulties occur and the computer seeks help. For example, the additional goal of set 6 indicates that if it is desired to have species 4, inert gas, flowing in connector 10, it would also have to be flowing in connector 22. Of course more information could be provided by stating that inert gas has to be flowing in connectors 13, 17, 19, etc., but the single level III hint was sufficient for the synthesizer.

The additional goal of set 9, which states that the species are to be trapped in the upper header as well as the reactor, makes it possible to evacuate them later. The additional goals of set 10 means that hydrogen and inert gas have to be evacuated from the upper header to continue with the next set of goals.

The ability to generate and receive level III goals greatly increases the power of the synthesizer and provides access to the intuitive problem solving capability of the experienced engineer.

OPERATION CRITERIA

A satisfactory synthesis has been considered to be one which achieves the operation objective with no unsafe operations. There usually are a large number of satisfactory operation sequences just as there are a number of ways of winning a game of chess. However, we can imagine other criteria which might take on significance.

Operator error is a major source of industrial disaster. Therefore, we may seek the operation sequence which allows the greatest number of operation errors before a hazardous condition occurs. This might be approached by modifying the synthesis algorithm to test not only for hazards on a given operation, but for hazards on the worst continuation of K operations, where K is the number of operator errors to be tolerated. Starting with large K, no sequence of operations might be found, and the first sequence synthesized as K is reduced and would be optimal according to this criterion.

Another approach to reducing the effect of operator error is to synthesize the operation sequence which is most direct and involves the fewest valve changes. According to this criterion the engineer's operation procedure (35 operations) is preferred to the computer's (43 operations).

A hierarchy of dangerous situations might arise in which disaster initiating events rank higher than events which merely damage equipment or cause minor economic losses. The operations which skirt the most dangerous operations by the widest margin would then be optimal according to this criterion.

The valving operations initiate changes in the process system which differ greatly in magnitude. The evacuation of a short length of pipe is much easier than the evacuation of a five-story high reactor, yet we have not included this information in the program. One approach would be to assign durations to each event. The synthesized operation sequence then forms a critical path program to be solved by standard methods. The sequence which gives the earliest completion time to goal achievement is then optimal.

We have not investigated these higher operation criteria at all and have the suspicion that this will open a research area as challenging as that of finding a mere satisfactory sequence.

CONCLUSIONS

It is entirely feasible to program the computer to synthesize failure-safe operation within the limits of the operation problems examined. These limits include the macro-operation of industrially large processes in which we are concerned only with the control of the presence of material in the system. The more sophisticated problems involved in actual flow including pressure drop distribution, velocity-distance time lags, residence time distributions, and other realities of process operation were not included in our model. These refinements are not necessary to the understanding of failure safe operation synthesis.

The next step in this research is not in the direction of process model sophistication, for we know that can be done and there are no new principles to be found exploring that direction. Rather, attention should focus on the principles for the synthesis of failure-safe operation using more severe criteria than merely goal achievement with no hazardous events. Particularly intriguing is the criterion of goal achievement with no hazards subject to K operation errors. This is interesting in that an operator error forms a new synthesis problem invalidating any a priori operating instructions. Such a synthesis problem is a real game in which the opponent is allowed K trials to force the process into disaster, and the synthesizer attempts to prevent the disaster. There is much to be learned about the design and operation of disaster tolerant processes from the study of this game.

ACKNOWLEDGMENT

Work performed at the University of Wisconsin with the support of the National Science Foundation, Shell Oil Foundation and the United States Army.

LITERATURE CITED

Bernstein, A., and M. D. Roberts, "Computer vs. chess-player," *Scientific Am.*, **198** (June, 1958).

Kister, J., P. Stein, S. Ulam, W. Walden, M. Wells, "Experiments in chess," *J. Assoc. Computing Machinery*, **4**(2), 174 (1957).

Lee, K. F., A. H. Masso, and D. F. Rudd, "Branch and Bound Synthesis of integrated Process Design," *Ind. Eng. Fundamentals*, **9**, 48 (1970).

Masso, A. H., and D. F. Rudd, Synthesis of System Design—Heuristic Structuring, *AIChE J.*, **15**, 10 (1969).

Newell, A., J. C. Shaw, and H. A. Simon, A. General Problem Solving Program for Computers, *IBM J. R & D*, **2**, 4 (1958).

Newell, A., and H. R. Simon, *Human Problem Solving*, Prentice-Hall, Englewood Cliffs, N. J. (1971).

Rivas, J. R., and D. F. Rudd, "Adaptive Safety Interlock Systems," *AIChE J.*, **20**, (1974).

Shannon, C. E., "Automatic Chess Player," *Scientific Am.*, **48**, 182 (Feb., 1950).

Sirrola, J. J., G. J. Powers, and D. F. Rudd, "Synthesis of System Design—Concept Generation," *AIChE J.*, **17**, 677 (1971).

Sirrola, J. J., and D. F. Rudd, "Computer-Aided Synthesis of Chemical Process Designs," *Ind Eng. Fundamentals*, **10**, 353 (1971).

Turing, A. M., "Computing Machinery and Intelligence," *Mind*, **59**, 433 (Oct., 1950).

Uhr, L., *Pattern Recognition*, Prentice-Hall, Englewood Cliffs, N. J. (1973).

Manuscript received September 11, 1973; revision received September 19 and accepted September 20, 1973.