W.J. Parkinson
P.D. Shalek
K.H. Duerre
Los Alamos National Laboratory
Los Alamos, New Mexico 87545

G.F. Luger
Department of Computer Science
University of New Mexico
Albuquerque, New Mexico 87131

M. Jamshidi
Department of Electrical &
Computer Engineering
University of New Mexico
Albuquerque, New Mexico 87131

Two Intelligent Control Systems for Silicon Carbide Whisker Production

Abstract

Silicon carbide whiskers, a strong material that resembles cat's whiskers, are produced, primarily, for strengthening ceramics and metals. The current method for producing these whiskers is to use a semibatch process, extremely difficult to model mathematically. An alternative model utilizes the expertise of operators accumulated from years of experience in initializing and running the process. We designed two expert systems in which the process can be set up and run by nonexperts. One system uses crisp logic and the other fuzzy logic. In this article, we compare the two systems. © 1993 John Wiley and Sons, Inc.

Introduction

Silicon carbide (SiC) whiskers, a strong material that resembles cat's whiskers, are produced primarily as reinforcing material for strengthening ceramics and metals. Whiskers can be used as randomly oriented chopped fibers or, in long lengths, can be made into yarns and woven, creating an even more effective directional reinforcement. Although the primary purpose of the whiskers is for compositing materials to add strength, we are also considering other uses.

The Los Alamos vapor–liquid–solid (VLS) whisker production process is a semibatch process that combines an almost steady-state flow process with an unsteady-state, batch, silicon-monoxide-generation process. This combined process is extremely difficult to model mathematically. The process is nonlinear and time varying, making it difficult to apply methods from classical or modern control theory. Proper whisker growth can occur only with the proper process conditions and, therefore, depends upon good process control. Yet, despite apparent control problems, high-quality whiskers can be produced when the process is set up and run using rules accumulated from many years of trial and error.

At least two catalyst types, several reactor configurations, and several sets of process conditions are available in the setup mode. Picking the best set of conditions requires experience and knowledge of the process.

In the past, a few experts set up the whisker production runs and operated them according to rules they had memorized or written in notebooks. When the whiskers process became a candidate for technology transfer, we had to solve the problem of how to transfer the expertise to industry without transferring the expert. To solve this problem, we designed an expert system capable of assisting users with the process setup and the control problems, thus creating a vehicle for technology transfer.

We built an expert consultant capable of providing users with enough information to set up a run that would produce the desired whisker type. The setup information was then incorporated into a rule base designed to drive the control system. The control system was designed to cor-

This work was performed under the auspices of the U.S. Department of Energy.

CCC 1064–1246/93/030199-16

rect perturbations during the process run to ensure that users obtain the desired product.

Over the years, the laboratory-scale whiskers production system has evolved and improved. The automatic control portion of the system represents a "brute-force" approach that requires an operator interface. The expert system portion of the system requires a human sensor, the operator, who checks for perturbations or upsets. After observing the controlled variable readings, the operator must ask the expert system whether the process is behaving correctly. The expert system responds by suggesting which, if any, corrections should be made, and the operator then makes corrections if required. Automatic controls and sensors can later be added to the process to automate the expert system operation.

Most of the rules for the expert system were obtained by interviewing the operators, and some were extracted from the data in a relational data base. The data base enabled us to observe and plot data in a wide variety of patterns. From these patterns, we found the optimum correlations, using classification methods from pattern recognition theory. These correlations yielded some excellent rules.

This rule-based expert system was designed to run on a PC, which can be located in the laboratory associated with the whisker process equipment for easy access by the operator.

Description of the Whisker Growth Process

The VLS laboratory-scale SiC whisker production process at Los Alamos is a semibatch process, that is, the silicon-monoxide-generation step is run in the transient or batch mode and other steps in the process are run in the steady-state continuous-flow mode.

Figure 1 shows one configuration of the SiC whisker reactor. In this configuration, silicon dioxide (SiO_2) bricks impregnated with graphite are placed inside the reactor. After being heated, the bricks produce silicon monoxide (SiO) (the ingredient used to make SiC) by the following reaction:

$$SiO_2 + C \rightarrow SiO + CO \qquad (1)$$

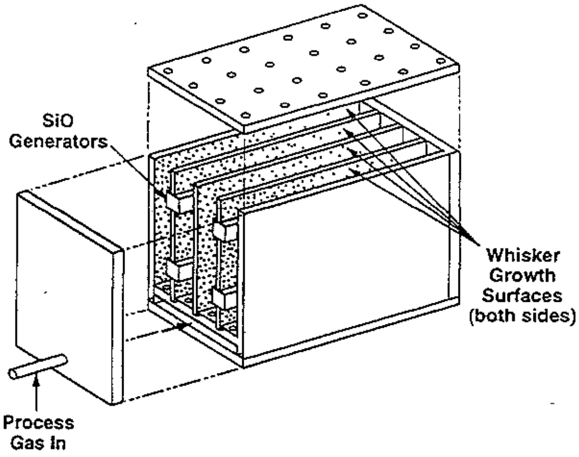

Figure 1. Los Alamos silicon carbide whisker production reactor.

The production rate for SiO is proportional to the concentration of SiO_2 and carbon (C) in the brick. This SiO generation step, where concentrations diminish with time, constitutes the transient or batch portion of the process. As shown in Figure 1, a mixture of gases containing methane (CH_4), the carbon source for the SiC production, is forced through the reactor. The composition and flow rate of this gas mixture never varies throughout the length of a production run and, thus, constitutes the steady-state portion of the process. The SiC is formed by the reaction shown in eq. (2):

$$SiO + CH_4 \rightarrow SiC + H_2O + H_2 \qquad (2)$$

The SiO formed by eq. (1) must mix with CH_4 in the process gas stream to form SiC. These gases must find their way to a whisker growth surface, as shown in Figure 1. Figure 2 shows the modes of mass transport from the SiO generator to the whisker growth surface. Figure 3 shows

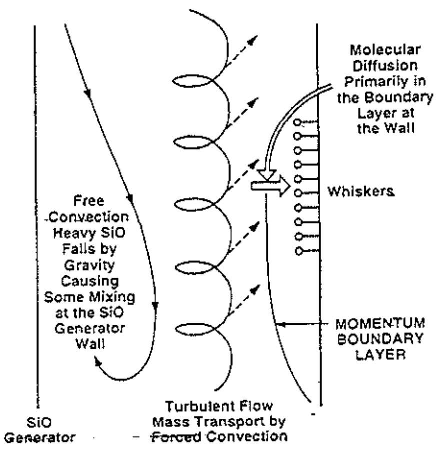
Molecular Diffusion Primarily in the Boundary Layer at the Wall
Whiskers
Free Convection Heavy SiO Falls by Gravity Causing Some Mixing at the SiO Generator Wall
MOMENTUM BOUNDARY LAYER
SiO Generator
Turbulent Flow Mass Transport by Forced Convection

Figure 2. Gas-phase mass transport modes.

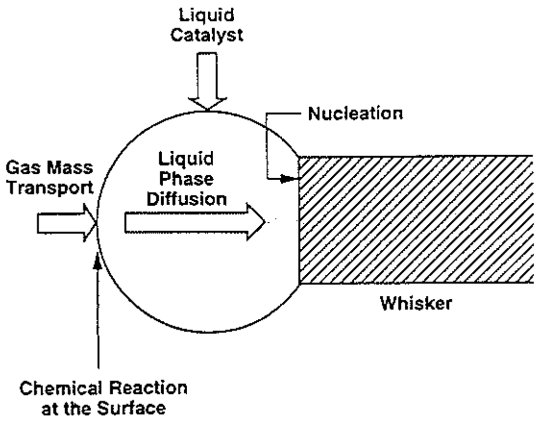

Figure 3. Steps in the overall kinetic process for growing silicon carbide whiskers.

the steps believed to be involved in the overall kinetic process for growing the whiskers. The whiskers grow away from the catalyst-coated substrate, with a ball of liquid catalyst attached to the growing end. This liquid catalyst ball is the main difference between the VLS process and the conventional vapor–solid whisker process. In the VLS process, the vapor is the gas that contains the compounds shown in eq. (2), the liquid is the catalyst, and the solid is the SiC whisker. As the process continues, the whiskers grow further and further into the gas stream, affecting the boundary layer around them and, hence, the mass transfer rate. This is one of many reasons that the process is both time varying and nonlinear.

The previous discussion illustrates the difficulties of modeling with normal mathematical-algorithmic techniques. We found that our mathematical models could not adequately predict whisker yield and type from a particular experimental setup.

Equation 3 is an example of a model derived from fundamental principles, that is, from mass and momentum balances about the reactor:

$$\theta = \phi k_m (c_0 - c_s) A - u c_a A_c \qquad (3)$$

where θ is the average SiC growth rate (kg/s); ϕ is a numerical constant; k_m is the integrated mean mass-transfer coefficient (m/s); c_0 is the concentration of SiO at the generator wall (kg/m^3); c_s is the concentration of SiO at the concentration boundary layer outer surface (kg/m^3); A is the surface area of the SiO generators (m^2); u is the average reactant gas flow (m/s) parallel to the catalyst and SiO generator surfaces; c_a is the average bulk concentration of SiO (kg/m^3) in the boundary layer; and A_c is the average cross-sectional area (m^2) of the reactor normal to the gas flow, minus the concentration boundary layer thickness.

Although the reactor configuration used to derive eq. (3) is similar in concept to the reactor shown in Figure 1, its geometry is a little different and it was easier to model. The concentration boundary layer is similar in concept to the momentum boundary layer discussed earlier, but it is a different boundary layer. The momentum boundary layer in Figure 2 is shown on the right-hand side, near the catalyst wall, where it covers the whiskers. The concentration boundary layer shown on the left-hand side of this figure represents the heavy concentration of SiO near the generator. This layer varies in thickness depending on the reactor height and the time. The integrated mean mass-transfer coefficient comes from a complicated equation that relies heavily upon knowing the thickness of the concentration boundary layer. The SiO concentration terms (especially c_0 and c_s) are unknown so they must be estimated. Using data from previous runs, we were able to estimate these parameters and, sometimes, make reasonable predictions using eq. (3).

Later in the development of the process, when we increased the gas velocity to get better mixing, this model was no longer valid. It predicted decreased yields when increased yields were actually obtained. We believe the predictions were incorrect because the increased flow rate changed the mass-transfer rate-controlling step and, thus, invalidated the assumptions used for deriving this model. Although assumptions were chosen carefully when building this model from basic principles, changes in the operating conditions resulted in significant deviations from the model.

Because this process was producing good whiskers, we wondered "How can we grow whiskers this well, without an adequate understanding of the physics of the process?" We concluded that the answer is that the expert operators have learned excellent rules of thumb, through years of trial and error, for setting up and running whisker growth experiments. Our next question was "How can we capture this expertise so that the process can be set up and run by people who are not experts?" This question was especially important because the technology was earmarked for transfer to industry and we wanted to be able to transfer the technology without transferring our experts. Our solution was to develop an expert system that models the experts rather than one that models the process.

Knowledge Acquisition

Most of the rules used in our expert systems were obtained directly from the experts who design the whisker growth experiments and run the process. The rest of the rules were obtained from experimental data. Data from our gas chromatograph were read directly into the computer. These, and other, experimental data were used in a computer program to generate additional information that is stored in a data base. A relational data base management system is used to query the data base and produce tables that contain many different combinations of the stored information. The data in these tables are then plotted and analyzed in an effort to develop new rules. A relational data base management system was an excellent tool for this job because it was easy to observe patterns, or relationships between almost any combination of variables desired. If a pattern looked interesting, more powerful techniques could be used to produce more quantitative information (rules). In this study, we used our data base management system to observe two different types of patterns: trends and clusters.

When the observed data patterns indicated that there was a single trend or a relationship between two or more variables, the method of least squares was used to fit the data to produce new rules. Figure 4 shows an example of the least-squares method used for developing a rule about whisker yield as a function of the number of uses of the growth plate.

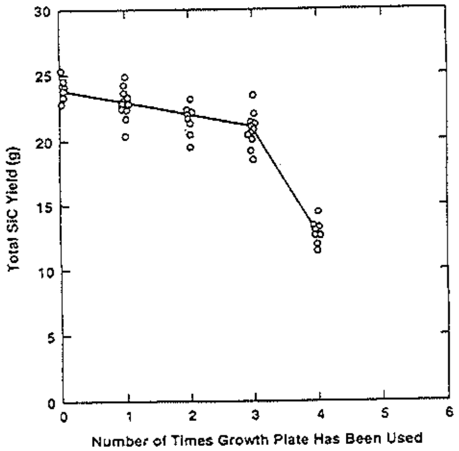

Figure 4. Least-squares method used to develop a rule to determine number of times a growth plate should be used.

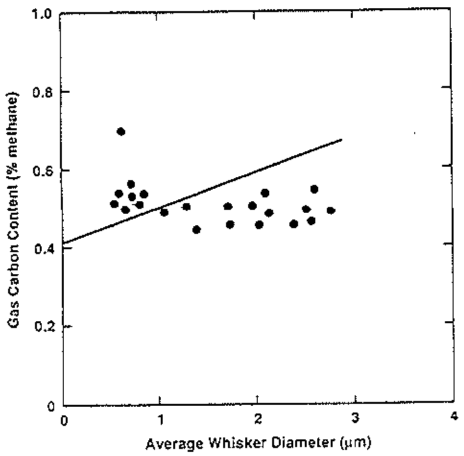

Figure 5. Perceptron method used to find a decision boundary to develop a rule that relates gas carbon content to mean whisker diameter.

In other cases, where the observed data seemed to fall into separate groups or clusters, pattern-recognition techniques were used to find decision boundaries between the data clusters. These boundaries were used to produce rules. Figure 5 shows the use of the perceptron algorithm, described by Tou and Gonzalez (1974), for finding a decision boundary. From this example, we obtained a rule from our data for determining the inlet gas carbon content (methane) required to produce whiskers in two given diameter ranges for a given gas SiO content, catalyst type, and catalyst particle size.

Expert Systems

The goal for this project was to develop a small PC-based expert system in two phases, as shown in Figure 6. In phase 1, we developed an expert whisker growth consultant to help operators set up a run that will enable them to attain the desired whisker type and quantity. The knowledge obtained from the expert consultant is entered into a knowledge base used by the expert control system, developed as phase 2 of the project. (Figure 7 shows how the control system operates.) The expert consultant can be used alone or with the expert control system. The

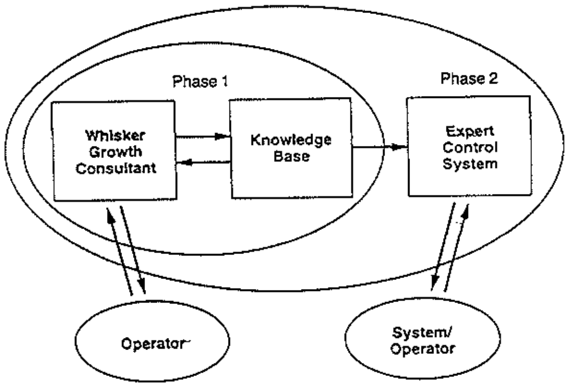

Figure 6. Two phases of the expert system design.

expert control system requires an operator to observe conditions and set the controllers. Based on input from the expert consultant, the operator sets control points for temperatures and gas compositions for each of the several time segments of the production run.

The process setup parameters obtained from the expert consultant would satisfy the control problem for an ideal system. But, because whisker production is a complicated semibatch process this initial setup will not always satisfy the problem. As shown in eq. (1), the success of the whisker growing process depends greatly on the ability of the system to produce SiO. SiO production is the *in situ* step and it is neither directly controllable nor observable. The SiO comes from measured and cut pieces of silicon dioxide bricks impregnated with carbon.

Although one goal is to strive for consistency in composition from run to run, this goal is not always achievable. As changes occur in the production rate of SiO, changes are also required in the operating conditions. The production rate can be controlled to some degree, by controlling the reactor temperature and the carbon monoxide (CO) input, but the important controlling factor is the carbon concentration in the generator. The carbon concentration is high at the beginning of the run and essentially zero at the end of the run, so the ability to control the SiO production varies throughout the run. The production rate of the SiC is determined by observing the CO, hydrogen (H_2), and methane concentrations in the exit gas. The expert control system is, therefore, set up to help the operator determine the new setting as depicted in Figure 7. We believe that with today's microprocessor technology the entire process could be automated and the operator could be eliminated from the loop after the initial setup, but this has not yet been done.

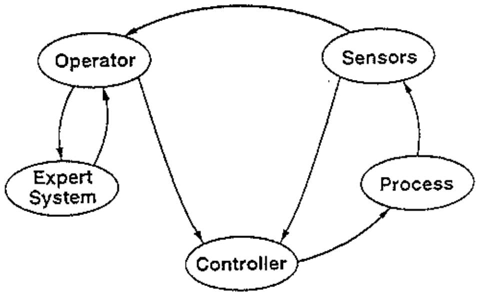

Figure 7. Demonstration of the operator interface with the expert system and control system.

Expert System Shells

An important part of this project was to find an expert system shell that would run on an inexpensive computer that could be used in a harsh and dirty environment. We found that the following three rule-based shells could be used with the PC family of computers and could easily handle the task:

- CLIPS (Giarratano, 1987),
- EXSHELL (Luger and Stubblefield, 1989), and
- Togai's Fuzzy-C Development System (Hill, Horstkotte, and Tiechrow, 1989).

All of these shells performed well. They can be run on a PC using the DOS operating system with no modifications, which means that the maximum hardware requirements are 640 kB RAM.

CLIPS, developed by NASA, is a forward-chaining, rule-based shell written in the C programming language. A knowledge of both C and LISP programming languages would be helpful, although not essential, for the programmer using the CLIPS shell.

EXSHELL, developed by the University of New Mexico Computer Science Department, is a backward-chaining, rule-based shell written in the PROLOG programming language. An understanding of PROLOG is essential for those who wish to use EXSHELL.

The Togai Fuzzy-C Development System, developed by Togai InfraLogic, Inc., is written in

C. It supplies the user with a framework in which to easily develop the membership functions and fuzzy rules to be used with the fuzzy logic approach to expert system design. However, users must write their own C program with which to drive the fuzzy expert system. Therefore, it is essential for those who wish to use this shell to have a good knowledge of the C programming language.

All of these shells are different, and all have strengths and weaknesses. For this project, we used the CLIPS shell and the Fuzzy-C Development System except at the beginning of the project, when we used EXSHELL extensively. Because EXSHELL is written in the PROLOG language, it is easy to add powerful support tools to the expert system, for example, explanation facilities that help users understand how and why the expert system has arrived at a given conclusion. An explanation facility is a nice feature, especially for the expert consultant. EXSHELL is limited because it was written as a university teaching tool and is not as well developed as the commercial shells. Although EXSHELL works well with symbols, it does not work well with numbers. Much of the expertise in the expert consultant can be expressed in a symbolic or discrete form, such as the logic for choosing the correct temperature profile from a set of temperature profiles. Giving symbolic advice such as "Use temperature-time profile A" is easy for EXSHELL expert systems. On the other hand, almost all of the expertise in the expert control system must be expressed in numeric form. EXSHELL programs have difficulty giving advice such as "Set the inlet CO composition to 9.1%." To do this properly, the EXSHELL expert system would require a separate rule for each possible CO composition. This was the primary reason for abandoning EXSHELL for the remainder of this project.

In our first attempt to write an expert control system (Parkinson et al., 1989), our approach essentially ignored the existing automatic control system, resulting in an expert system that gave general advice to the operator about almost anything that could go wrong. In our current expert control system, however, we tried to take full advantage of the automatic control system, enabling it to provide better information on how to upgrade its performance. Because EXSHELL has trouble giving this more precise information, we chose not to use it for the control part of this project.

Expert Consultant Development (Phase 1)

Figure 8, taken from Shalek et al. (1988), is an empirical-phase diagram of the whisker types that can be produced as a function of gas composition. The abscissa represents a change from silicon-rich to carbon-rich gas mixtures. The ordinate represents the SiO concentration in the gas phase. The properties of the whiskers from categories (1) through (7) (shown at the top of the chart) depend primarily upon the diameter of the whisker. Some commercial interest has been shown in all sections of the chart except areas E and F, which represent the combined species and the large bent needles.

For this study, the whisker types were lumped into groups slightly different than those shown in Figure 8. These groups are based on whisker lengths and diameters. The lengths, which vary from about 0.00318 m (0.125 in.) to about 0.0889 m (3.5 in), are divided into three categories: short, medium length, and long. The diameters, which vary from 1.0×10^{-7} to 1.5×10^{-5} m (0.1–15 μm), are divided into three groups: small, medium, and large. With short whiskers, we are interested only in small diameters; with long and medium-length whiskers, we are interested only in medium and large diameters.

In addition to developing rules to produce a particular whisker type, we have also developed some rules to help maximize the production of those whiskers under various operating constraints. These rules can be divided into three categories: (1) how to obtain the maximum yield with new growth plates; (2) how to obtain the

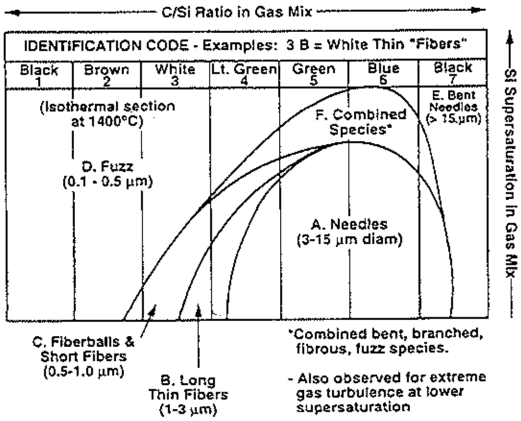

Figure 8. Empirical phase diagram for the growth of silicon carbide whiskers. Source: Shalek et al. (1988).

maximum yield with used growth plate; and (3) how to obtain a maximum yield in a limited run time. Combinations of these categories can also be used. The rules that are used depend on the type of whisker we are trying to produce, although all are included in the whisker growth consultant expert system.

For the laboratory-scale operation, the primary concern is with categories 1 and 2. The new growth plates require different gas compositions than do the used growth plates to produce the same quantity of whiskers. After one run, the new growth plates are coated with SiC, and from then on the SiC participates in the process chemistry. After about four runs, the whisker production has degraded enough that the plates must be replaced.

From our cost analyses, we found the whisker growth process to be both labor and material intensive. Replacing growth plates after every run is too expensive, even for a laboratory-scale process. Figure 9 shows the general shape of the whisker yield–time curve based on our observations of whisker experiments over several years. Because this was a laboratory operation, our approach was to run the reactor as long as possible to produce the maximum yield. The point of diminishing returns is reached long before the reactor is shut down. In a commercial

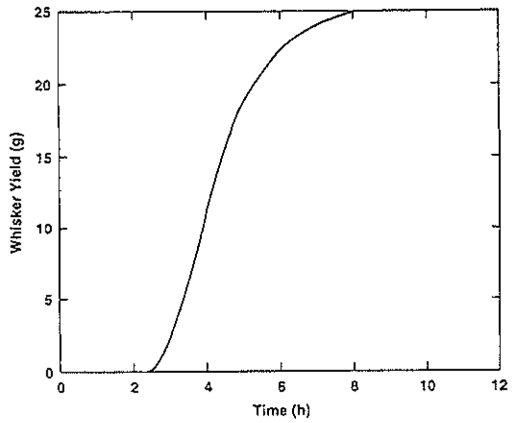

Figure 9. Yield–time curve for the silicon carbide whisker growth process.

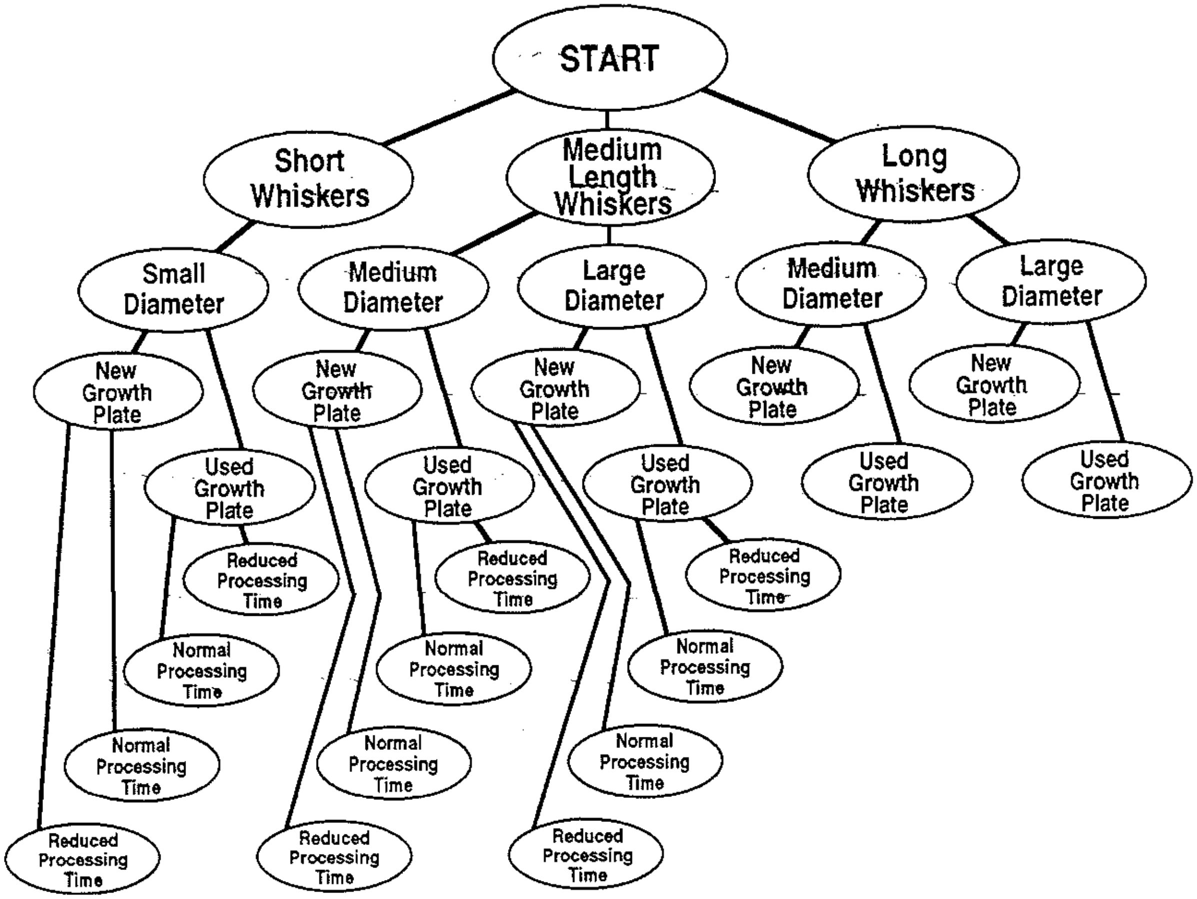

Figure 10. Search tree for the whisker growth consultant.

reactor, it may not be economically feasible to run longer than 6 h, the point of diminishing yield shown in Figure 9. This process is a candidate for technology transfer, so production rules have been developed for maximizing yields with shorter run times because we assumed that an industrial process, even a batch process similar to this one, would be optimized in a different manner. For example, if several batches were run to the point of diminishing returns in 1 day more whiskers would be produced than in the previous maximum production run, yet in the same amount of time.

Figure 10 is a simplified search tree for our whisker growth consultant. The leaves of the tree represent operating conditions that produce the types of whiskers desired. For example, the whisker product can be changed from medium length to long by changing the reactor and catalyst type. Producing short whiskers requires a gas composition change, while whisker diameter depends primarily on the catalyst type and particle size used.

Although choosing the proper production rules for a given run seems straightforward, the rules obtained from the data base show this procedure to be more complicated to set up and run optimally than it appears at first observation. For example, gas compositions and temperatures should be different depending upon the catalyst and the particle size used. The rules for maximizing the whisker yield in a shorter period of time depend upon the intended whisker diameter and length. Long whiskers are not typically produced in shorter runs.

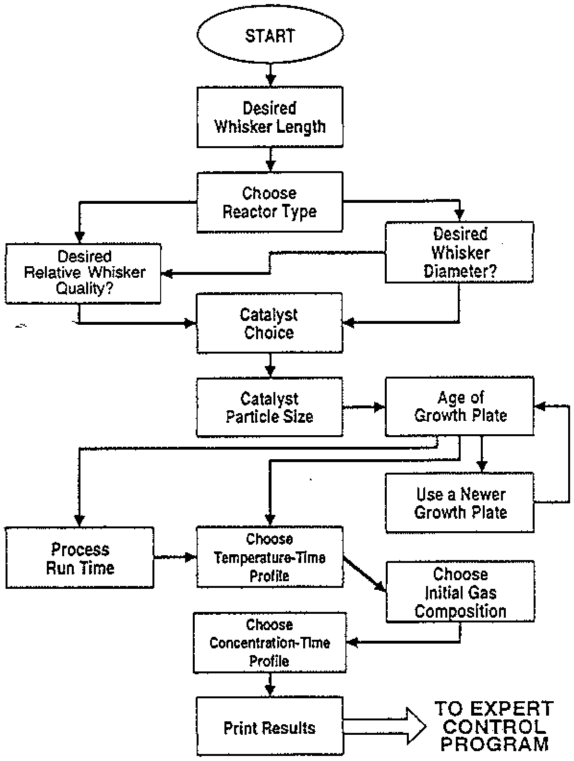
START
Desired Whisker Length
Choose Reactor Type
Desired Whisker Diameter?
Desired Relative Whisker Quality?
Catalyst Choice
Catalyst Particle Size
Age of Growth Plate
Use a Newer Growth Plate
Process Run Time
Choose Temperature-Time Profile
Choose Initial Gas Composition
Choose Concentration-Time Profile
Print Results
TO EXPERT CONTROL PROGRAM

Figure 11. Search space diagram for the expert consultant.

Figure 11 is the search space diagram for our expert consultant. The rectangular blocks represent the major decision points in the program. Figure 12 shows the CLIPS version of a dialog with the whisker growth consultant, following the search space shown in Figure 11. (Some of the values given in Figure 12 are fictitious because the whisker process falls under the purview of U.S. Export Control Laws and some information is subject to limited access.) The questions in Figure 12 are asked by the expert system and the input values (or answers) are supplied by the user. Note that, at each decision point, the user is given the opportunity to use values other than those recommended because, at the end of the consulting session, the correct values must be available for use by the control system.

The final version of our expert consultant was written with the CLIPS shell and required 46 rules. Earlier versions of the expert consultant were attempted with both the Fuzzy-C Development system and EXSHELL. A comparison of all of these systems is made in another section.

```
What is the desired average whisker length?
(In inches 0. to 3.5)
3.0

We recommend reactor type B, which will you use?
(A or B)
B

What is the desired average whicker diameter?
(In microns 0 to 12)
10

We recommend the manganese based catalyst, which one will you choose?
(manganese or iron)
manganese

We recommend sieve size 20-25, what will you use?
(25-32, 20-25, or 15-20)
20-25

How many times have you used the reactor growth plate?
(0 or greater)
0

We recommend temperature-time profile A, which will you use?
(A, B, C, or D)
A

We recommend that you use the following initial gas composition:
H2 = 80.0%
CO = 5.5%
N2 = 14%
CH4 = 0.5%
Would you use this? (yes or no)
yes

We recommend that you vary the CO concentration according to
  concentration-time profile A.
What profile will you use?
(A, B, C, or D)
A
```

Figure 12. Dialog with the expert consultant (CLIPS version).

Expert Control System Development (Phase 2)

The expert control system requires that the operator and expert consultant provide information from its knowledge base. Because system sensors do not communicate directly with the expert control system, the operator must observe the sensors, communicate with the control program, and, if necessary, manually adjust the system controls. The current system has only seven sensed variables and five possible control adjustment actions. The sensed variables are the temperature and the inlet and outlet compositions of three of the four process gases (hydrogen, carbon monoxide, and methane). The five control actions are the temperature adjustment, the rate of temperature change, and the flow of the three individual inlet gases (hydrogen, carbon monoxide, and methane). The amount of adjustment needed depends on the current reading and the run setup conditions supplied by the expert whisker growth consultant. Figure 13 shows the search tree for the expert control system. Figure 14 is a search space diagram for the expert control system. The current expert control system contains 81 production rules. A crisp version of the control system was developed using the CLIPS shell. It contains only 75 rules because the CLIPS shell makes it easy to combine rules in an IF–THEN–ELSE format. A fuzzy version of the control system was developed using the Fuzzy-C Development System. It contains all 81 rules.

Figure 15 shows the Fuzzy-C Development System version of dialog with the expert control system. Note that there is not much dialog with this control system. The input values are loaded on the command line, and the control system program immediately delivers all of the output values, in a manner similar to the way a microprocessor would handle this information for automatic control. The CLIPS expert control system dialog is similar to the CLIPs expert consultant dialog. There is interaction with the operator.

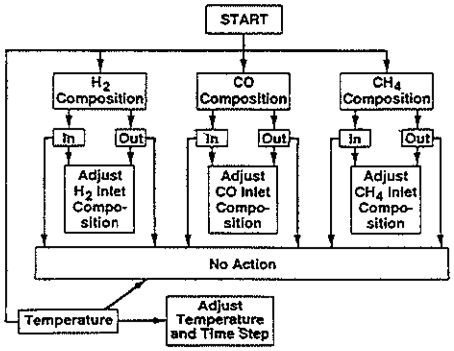
START
H_2 Composition
CO Composition
CH_4 Composition
In
Out
In
Out
In
Out
Adjust H_2 Inlet Composition
Adjust CO Inlet Composition
Adjust CH_4 Inlet Composition
No Action
Temperature
Adjust Temperature and Time Step

Figure 13. Search tree for the expert control system.

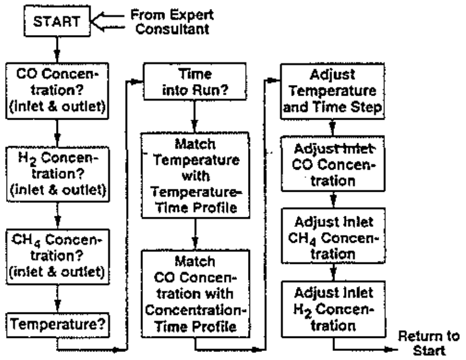
START
From Expert Consultant
CO Concentration? (inlet & outlet)
H_2 Concentration? (inlet & outlet)
CH_4 Concentration? (inlet & outlet)
Temperature?
Time into Run?
Match Temperature with Temperature-Time Profile
Match CO Concentration with Concentration-Time Profile
Adjust Temperature and Time Step
Adjust Inlet CO Concentration
Adjust Inlet CH_4 Concentration
Adjust Inlet H_2 Concentration
Return to Start

Figure 14. Search space diagram for the expert control system.

```
Temperature in 1200
Time is 120
CO In is 3
CO Out is 12
H2 In is 55
H2 Out is 57
CH4 In is 0.4
CH4 Out is 0.3

New CO In is 11
New Temperature is 1362
New Time Step is 37
New H2 In is 60
New CH4 In is 0.5
```

Figure 15. "Dialog" with the expert control system (Fuzzy-C version).

The current version of the expert control system is primarily concerned with variables that make a significant difference to the process, as opposed to early attempts to control every process parameter (Parkinson et al., 1989). Originally, rules were derived for nitrogen flow and reactor pressure. But, these parameters have only a relatively small effect on the VLS whisker growth process, and it is difficult to find meaningful rules for such parameters. We examined eqs. (1) and (2) to focus on important variables, giving particular attention to eq. (1), which governs the production of SiO. SiO is a necessary ingredient for producing SiC whiskers. SiO concentration is the most difficult of the process parameters to control and a significant amount of intelligence is required for its control. It is also the most difficult to observe.

Figure 16 is a generalized curve showing the concentration of SiO in the reactor as a function of time. The solid curve represents the uncontrolled case. The shape of this curve is related to the SiO production rate being a strong function

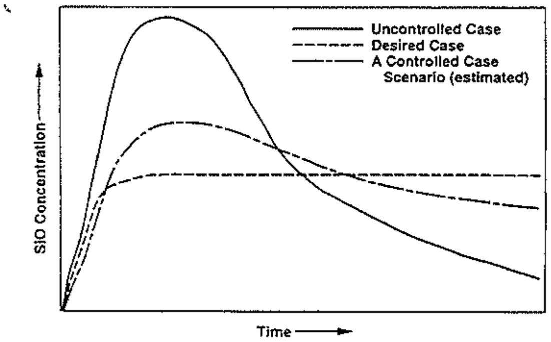

Figure 16. Concentration–time profile for silicon monoxide.

of carbon concentration in the SiO generator brick. The carbon is depleted with time, and, eventually, SiO is no longer produced. Here, the broken line is the desired case. A constant supply of SiO throughout the run is desirable. The broken line with dots, represents what we believe to be as close to the desired case as can be achieved with control. The curves representing the controlled and uncontrolled cases can both be generated from a simple differential equation describing the kinetics of eq. (1). The validity of the differential equation model has been verified many times by measuring the exit CO concentration from the reactor. Equation (1) shows that the exit CO concentration is related to the SiO production.

The amount of SiO produced by eq. (1) can be controlled by changing the reactor temperature and the CO inlet concentration. Early in the run, production can be reduced by decreasing the temperature and increasing the CO inlet concentration. This process, however, can be complicated because, in some cases, it is desirable to increase the temperature at the beginning of the run to initiate the reaction. Late in the run, production can be increased by increasing the temperature and reducing the CO inlet concentration. An expert system rule concerning eq. (1) should include the following: accumulated time since the run began; temperature; exit or outlet CO concentration as antecedents; new temperature; time step to reach that temperature; and new inlet CO concentration as consequents [i.e., IF (antecedents) THEN (consequents)].

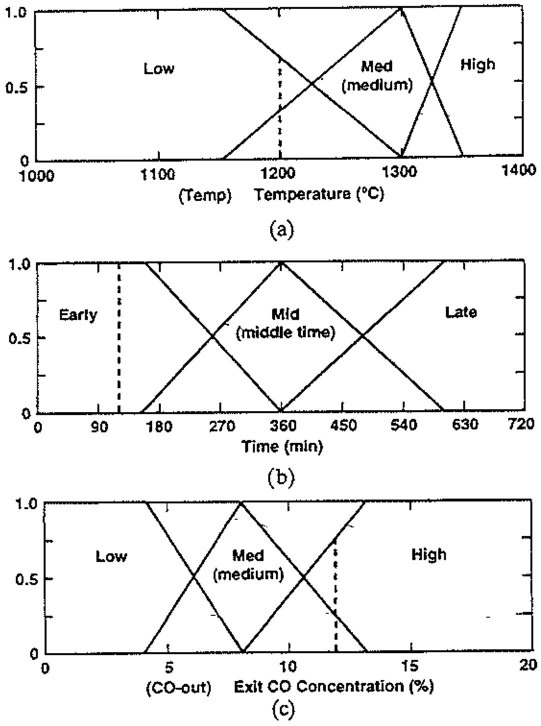

Figure 17. (a). Membership functions for Temp (reactor temperature) (– – – – = example value). (b). Membership functions for Time (time into run) (– – – – = example value). (c). Membership functions for CO_OUT (exit CO concentration) (– – – – = example value).

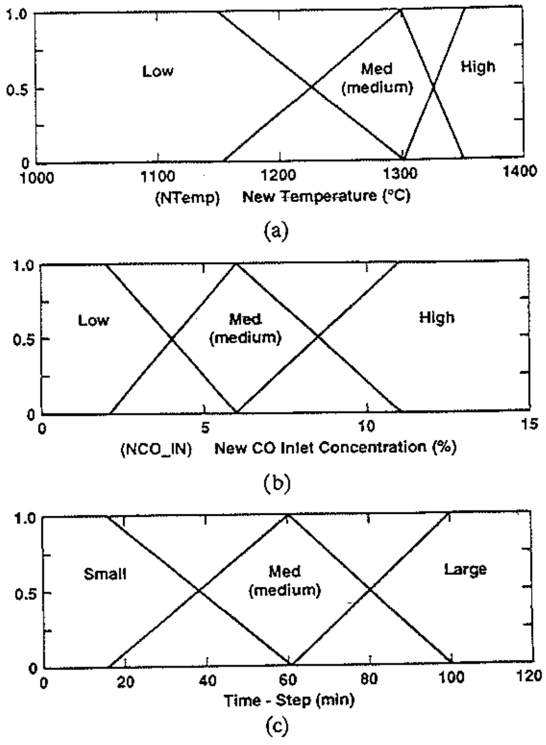

Figure 18. (a). Membership functions NTEMP (new reactor temperature). (b). Membership functions NCO_IN (new CO inlet concentration). (c). Membership functions for Time_Step (time allowed to reach the new temperature).

Equation (2) represents the rate of SiC production. This rate can be monitored indirectly by observing the amount of hydrogen and methane in the gas stream. Rules affecting this equation examine the amounts of hydrogen and methane in and out of the reactor, run time, and CO concentration.

To demonstrate how the fuzzy expert system works, we track the firing of the fuzzy rules that led to the settings for "New CO In," "New Temperature," and "New Time Step," shown in the sample output in Figure 15.

Figure 17 represents the fuzzy membership functions for the antecedents for the fuzzy rules used in this example. Figure 18 represents the membership functions for the consequents of the fuzzy rules used in the example. Again, some of the values given in Figures 17 and 18 are fictitious because some of the whisker information is limited-access information. The dotted lines in Figure 17 represent example input values (Temperature = 1200°C, Time = 120 min, and CO_OUT = 12%).

These triangular and trapezoidal membership functions make the difference between our fuzzy and crisp expert systems. The fuzzy parameters can have a membership value between zero and one in more than one fuzzy set. It is not necessary for the sum of the membership values for one parameter to add to one. (The crisp parameters can have a membership value of one in only one crisp set and must, therefore, have a membership value of zero in all other sets.) Because of this multiple membership, the input values for the example problem (the fuzzy expert system) will cause the following four (instead of one) rules to be fired:

As shown in Figure 17, the parameter value of 120 min for Time has a membership value in the fuzzy set Early of 1 [i.e., Time(Early) = 1]. The parameter value of 1200°C for Temp has a membership value of 0.35 in the set Med and 0.65 in the set Low [i.e., Temp(Med) = 0.35 and Temp(Low) = 0.65]. The parameter value of 12% for CO_OUT has a membership value of 0.15 in the set Med and 0.75 in the set High (i.e., CO_OUT(Med) = 0.15 and CO_OUT(High) = 0.75].

Our fuzzy expert system shell uses the Max–Min inference method described in the following example to resolve rules 1–4. The minimum membership function value for all of the antecedents in one rule that are connected by AND is the value used to resolve that rule. For separate rules that refer to the same consequent, or antecedents that are connected by OR in the same rule, the maximum membership function value is used to resolve the rule or choose between the rules. Consequent membership functions are clipped at the value of the membership function used to resolve the rule and combined where appropriate. The centroid of the clipped (and combined) membership function is the value used as a crisp result from the rule(s). This procedure is demonstrated in Figures 19 and 20.

Figure 19 shows portions of the appropriate antecedent membership functions on the left and portions of the appropriate consequent membership functions on the right. In Figure 19a, representing rule 1, the minimum membership function value for antecedent Temp(Med) = 0.35 is used to resolve the rule. All of the appropriate consequent membership functions are clipped at the height of 0.35. Similarly, Figures 19b –19d

Rule 1

  IF Time is Early AND Temp is Med AND CO_OUT is High THEN
    NTEMP = High, Time_Step = Med, NCO_In = High
  OR
Rule 2
  IF TIME is Early AND Temp is Med AND CO_OUT is Med THEN
    NTemp = High, Time_Step = Med, NCO_In = High
  OR
Rule 3
  IF Time is Early AND Temp is Low AND CO_OUT is High THEN
    NTemp = High, Time_Step = Small, NCO_In = High
  OR
Rule 4
  IF TIME is Early AND Temp is Low AND CO_OUT is Med THEN
    NTemp = High, Time_Step = Small, NCO_In = High.

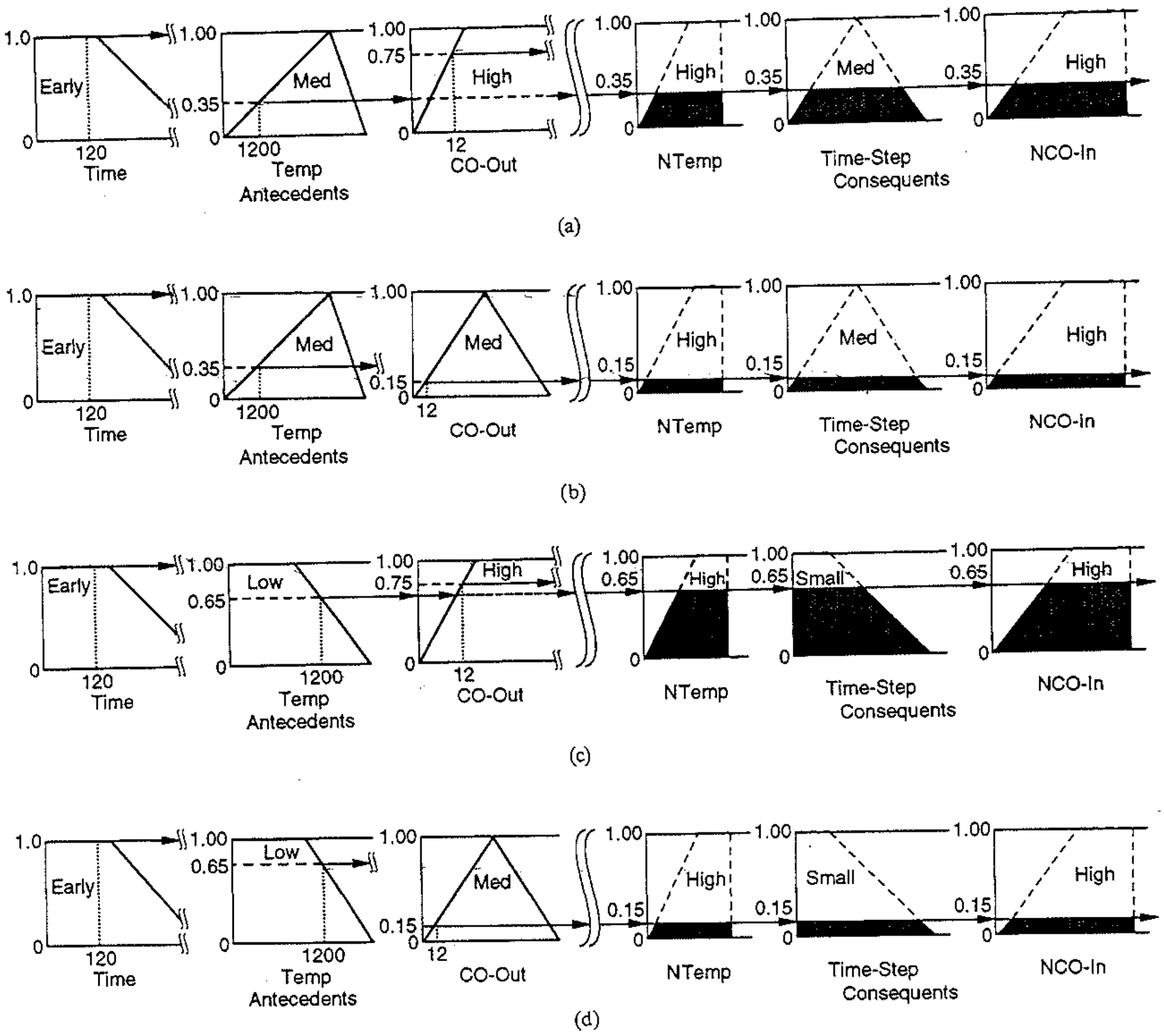

Figure 19. (a). Rule 1—minimum membership value of "ANDed" antecedents is 0.35. (b). Rule 2—minimum membership value of "ANDed" antecedents is 0.15. (c). Rule 3—minimum membership value of "ANDed" antecedents is 0.65. (d). Rule 4—minimum membership value of "ANDed" antecedents is 0.15.

represent the firing of rules 2–4. Rule 3, with a minimum membership function value of 0.65, supplies the maximum consequent membership functions for NTemp(High), Time_Step(Small), and NCO_In(High). NTemp and NCO_In stand for New Temperature and NewCO In, respectively. Rule 1, with a minimum membership function value of 0.35, supplies the maximum consequent membership function for Time_Step(Med). Clipped membership functions Time_Step(Med) and Time_Step(Small) are combined, and the centroids are calculated for the new temperature (NTemp), the new time step (Time_Step), and the new inlet CO concentration (NCO_In). This procedure is illustrated in Figure 20. These new values are then Temperature = 1326°C, CO in = 11%, and time step = 37 min. These values are shown in the sample problem in Figure 15.

In contrast, in the crisp expert control system (which has essentially the same number of rules as the fuzzy control system) only one rule is fired to resolve the same example problem. The crisp rule is as follows:

$$\text{IF}((0 < =\text{Time} < 250) \text{ AND } (1000 < =\text{Temp} < 1225) \text{ AND } (10.5 < =\text{CO\_OUT} < 20)$$

$$\text{THEN } (\text{NTemp} = 1363, \text{ Time\_Step} = 19, \text{ NCO\_In} = 12))\,.$$

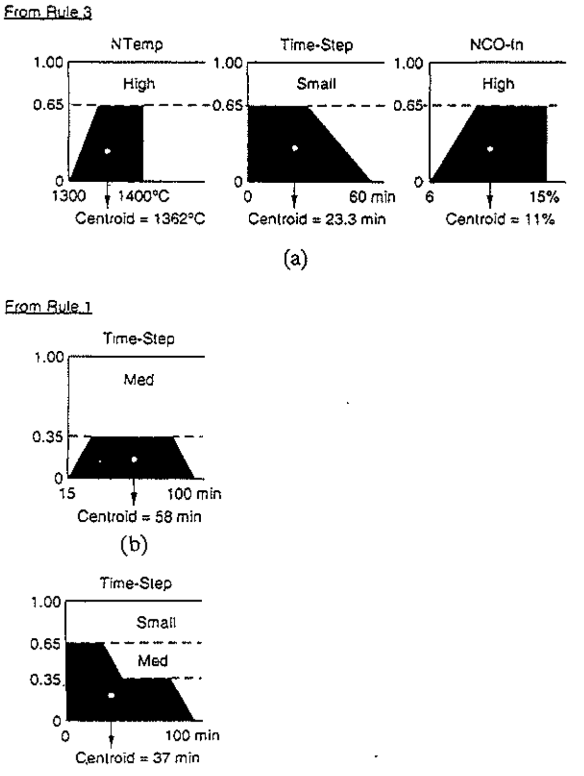

Figure 20. (a). Maximum "clipped" membership functions chosen from rule 3. The crisp consequent values are the centroids. (b). Maximum "clipped" membership function for Time_Step (med) from rule 1. The crisp consequent value is the centroid. (c). Combined "clipped" membership functions for Time_Step. The crisp consequent value for Time_Step is the centroid.

The only significant difference in the answers in this example vs. those of the previous example is the time step. The fuzzy solution, Time_Step = 37, has a membership value of approximately 0.5 in both the Time_Step(Small) and Time_Step-(Med) membership functions (see Fig. 18c). So, this value really represents a weighted average of each of the membership functions.

Figure 21 shows the crisp membership function boundaries for Time_Step, Small, Med, and Large. The fuzzy membership function boundaries are superimposed on this figure. Here, membership can be in only one set, and the membership values are either zero or one. The parameter value takes on the centroid of the set in which the consequent has membership. Hence, Time_Step(Small) = 19 and Time_Step (Med) = 0 for the crisp rule.

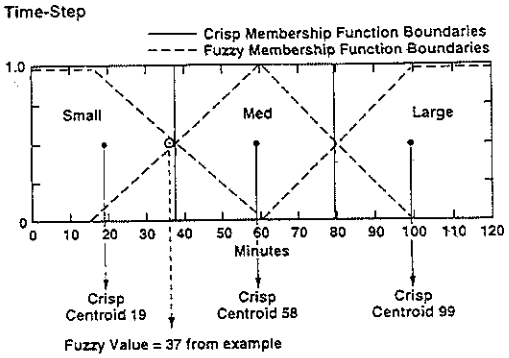

Figure 21. Demonstration of contrast between crisp and fuzzy membership functions for Time_Step.

Comparison of the Expert Systems

Phase 1, the expert consultant, was written with three different expert system shells, EXSHELL, CLIPS, and The Fuzzy-C Development System. The PROLOG-language-based EXSHELL does a credible job with the expert consultant because it deals largely with crisp, symbolic information. But, although EXSHELL can easily decide between an iron or a manganese catalyst it has difficult recommending a H_2 gas concentration of 80%. An important feature of EXSHELL, however, is its built-in explanation facility, which easily answers HOW and WHY questions about its decisions, an important feature for an expert consultant. Many existing problems with EXSHELL could be remedied by the further development of this university learning tool.

The CLIPS expert system shell is well developed. It uses a forward-chaining search strategy, as opposed to the backward-chaining strategy used by EXSHELL. It is, therefore, much more difficult to implement an explanation facility with CLIPS. Because most of the decisions made by the expert consultant are crisp (i.e., it makes decisions, such as which reactor to use, A or B? etc.), CLIPS is a good shell to use for the expert consultant construction.

The Fuzzy-C Development System was used to write a version of the phase-1 expert consultant. Although many of the antecedent parts of the rules are fuzzy [e.g., Is a 0.0762-m (3-in.) whisker long, medium length, or short?], most of the consequent parts of the rules are crisp (e.g., If the desired length is long, use reactor type B). The need for a fuzzy expert system for the expert consultant is questionable. Most of the consequents in the expert consultant rules are crisp because our original thinking about our data led us to set the problem up in a crisp manner. If we take the time to rethink some of this information in a fuzzy manner, we may find that we get better results. For example, there are many fuzzy temperature–time profiles between profiles A, B, C, and D (see Fig. 12). Can we use this to our advantage? At present, we do not know. But, if we find that fuzzy answers are better than crisp answers for this problem we will need to develop a fuzzy expert consultant.

EXSHELL was not used for the current version of the expert control system because it requires use of precise numbers for control settings, and precise numbers are difficult for EXSHELL to deal with. We wrote comparable expert control systems with both the CLIPS shell and the Fuzzy-C Development System. One of the great advantages of the fuzzy control system (over the crisp system) is the smoothness of operation that comes from both the rule antecedents and the consequents being fuzzy. Thus, with more than one rule being fired for each set of conditions the results are a weighted average of many possibilities. In the crisp system, only one rule is typically fired for each set of conditions and, thus, only one crisp answer is obtained.

The crisp expert system (the CLIPS version) works with rules that contain crisp antecedents and crisp consequents, so the rules and their results can change abruptly as the boundaries of the crisp sets or membership functions are crossed. Figure 22 shows the temperature–time profiles that are calculated by both expert systems when our example problem is continued for more than one time step, as well as the temperature–time profile we have attempted to model with our rules. The results shown in Figure 22 are obtained by changing only Time, Temperature, and Time_Step. Both models could be improved, although the crisp expert system is the rougher model. For the crisp expert system, the temperature–time slope between Time = 120 and Time = 149 is too steep because the initial temperature (1200°C) is near a set boundary. Here, a choice had to be made between the results of two rules. The choice in Time_Step made the slope too steep, but if the other rule had been used the slope would have been too flat. The fuzzy system averaged the two slopes by firing more than one rule. For the crisp system, the slope between the Times 248 and 347 min is too flat because the Time (248 min) is near a set boundary. As shown in Figure 22, the same crisp rule was fired twice in a row, even though the conditions were almost sufficiently different to fire a new rule. The solution to this roughness problem is to divide the crisp sets, or crisp membership functions, into smaller sets and write more rules. This solution could also be used to improve the fuzzy expert system. A better solution, however, would be to improve the shapes and boundaries of the existing membership functions.

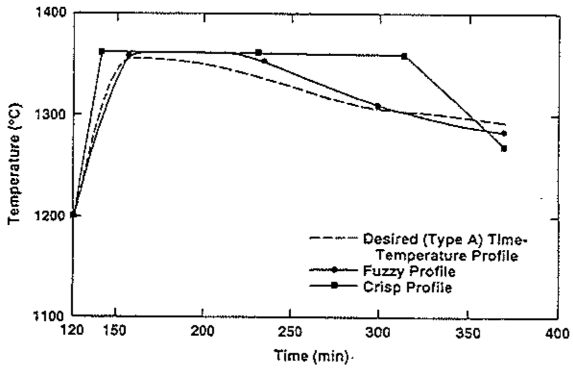

Figure 22. Comparison of fuzzy and crisp temperature–time profiles with the desired profile for the example problem.

CLIPS is an easy expert system shell to use. It stores facts in a stack, an abstract data structure, or a Fact Stack. The rules whose antecedents match the facts on the Fact Stack are the rules that are fired. With this architecture, it is relatively easy to add the facts generated by the expert consultant to the Fact Stack of the expert control system. It is also relatively easy to write the rule base to handle all of the possible combinations of facts that the expert consultant can provide. Adding the information from the expert consultant to the Fuzzy-C Development System version of the control system is more difficult, however, because of the differences between the architectures of the two shells (not because the fuzzy expert system has to deal with some crisp logic). The Fuzzy-C Development System is designed to facilitate ease in creating membership functions and writing rules, but all searches must be driven by a C program written by the user. This requirement makes the Fuzzy-C system more difficult to use than the CLIPS shell, but still it does allow great flexibility. Crisp rules and multiple options are resolved in the C driver program.

Finally, we believe that a crisp expert system is a good choice for an expert consultant with a knowledge base similar to the one developed for this study. However, we feel that our fuzzy expert control system is superior to our crisp one because the fuzzy system is better able to track setpoints and the transition from state to state is much smoother than in the crisp case, as demonstrated in Figure 22.

Conclusions and Future Work

We designed and implemented an expert consultant and control system for a material production process. This process is a good candidate for artificial intelligence and expert systems because it is a difficult process to model mathematically. The process is run in the laboratory using rules that are based on years of experience. Excellent whisker production was achieved using these rules.

We modeled our expert consultant with three different expert system shells. The CLIPS shell is our current choice as the best shell to use for the consultant, although the explanation facility capability available in EXSHELL is desirable. We believe it would be worthwhile to reevaluate our consultant data in fuzzy, rather than crisp, terms. If a benefit is seen in this approach, we would use the Fuzzy-C Development System instead of CLIPS for the consultant.

The expert control system was created in crisp form using CLIPS and in fuzzy form using the Fuzzy-C Development System. The fuzzy expert control system is favored because the response it produces is much smoother than that produced by the crisp expert system with essentially the same number of rules. It is cumbersome, however, to add facts from the expert consultant to our fuzzy control system. In the future, we will look at better ways to incorporate this information into our fuzzy control system (a new version of the Fuzzy-C Development System is more capable in this respect). There are indications that a future version of CLIPS will use fuzzy logic. A fuzzy CLIPS shell would be an excellent tool for developing expert control systems.

Our fuzzy expert system could be enhanced by improving our membership functions. Hill et al. (1989) state that, "Determining the number, range and shape of membership functions to be used for a particular variable is somewhat of a black art." The reference further states that trapezoids and triangles, such as those found in Figures 17 and 18, are good starting points for membership functions. Ideally, we might expect to replace the triangles with bell-shaped curves and the trapezoids with S-shaped curves. Giarratano and Riley (1989), Turksen (1991), Klir and Folger (1988), and Karr (1991) suggest methods for determining better membership functions. Improving these functions will require taking a harder look at our data. The idea of using neural nets, fuzzy pattern recognition, or genetic algorithms (Karr, 1991) to "teach" the membership functions to improve their shape is intriguing and will be considered for a future project.

Finally, we would like to implement our expert control system in hardware and run the process with less operator participation. This is certainly possible with today's technology.

References

Giarratano, JC (1987): *Clips User's Guide, Version 4.1 of CLIPS*, Artificial Intelligence Section, Lyndon B. Johnson Space Center.

Giarratano JC, Riley G (1989): *Expert Systems—Principles and Programming*. Boston, MA: PWS-Kent.

Hill G, Horstkotte E, Tiechrow J (1989): *Fuzzy-C Development System User's Manual, Release 2.1* Irvine, CA: Togai Infralogic.

Karr C (1991): "Genetic algorithms for fuzzy controllers" *AI Expert* February: 26–33.

Klir GJ, Folger TA (1988): *Fuzzy Sets, Uncertainty, and Information*. Englewood Cliffs, NJ: Prentice Hall.

Luger GF, Stubblefield WA (1989): *Artificial Intelligence and the Design of Expert Systems* Redwood City, CA: Benjamin Cummings.

Parkinson WJ, Shalek PD, Peterson EJ, Luger GF, (1989): "Designing an expert system for the production of silicon carbide whiskers" In Demeri, MY (ed): *Expert System Applications in Materials Processing and Manufacturing* Warrendale, PA: The Minerals, Metals & Materials Society

Shalek PD, Phillips DS, Christiansen DE, Katz JD, Parkinson WJ, Petrovic JJ (1988): "Synthesis and characterization of VLS-derived silicon carbide whiskers" In *Proceedings of the International Conference on Whisker and Fiber-toughened Ceramics* Oak Ridge, TN.

Tou TJ, Gonzalez RC (1974): *Pattern Recognition Principles* Reading MA: Addison-Wesley.

Turksen IB, (1991): "Measurement of membership functions and their acquisition" *Fuzzy Sets and Systems*, **40**: 538.