Proceedings of DETC ’98
1998 ASME Design Engineering Technical Conferences
September 13 - 16, 1998, Atlanta, GA

DETC98/CIE-5702

ADVANCED FMEA USING META BEHAVIOR MODELING FOR CONCURRENT DESIGN OF PRODUCTS AND CONTROLS

Steven Kmenta
Design Division
Mechanical Engineering Dept.
Stanford University
Stanford, CA 94305-4022
kmenta@cdr.stanford.edu

Kosuke Ishii
Design Division
Mechanical Engineering Dept.
Stanford University
Stanford, CA 94305-4022
ishii@cdr.stanford.edu

ABSTRACT

This paper presents the use of Advanced Failure Modes and Effects Analysis (AFMEA) as a methodology for the concurrent design of electro-mechanical products and their control systems. The past two years have seen the extension of AFMEA to simulate dynamic changes of device operations using meta-behavior modeling. This approach can help engineers identify failure modes associated with controls and their interaction with physical systems and drive system design toward more reliable solutions. The proposed method uses behavior modeling to map control functions to physical entities and identifies failure modes as the departure from intended control functions. AFMEA provides a framework for controls and hardware developers to discuss and understand the relationship between sub-systems, controls, and overall system performance. An example of a power generation system illustrates how AFMEA applies to the early stages of layout and controls design.

KEYWORDS: behavior modeling, FMEA, reliability, concurrent engineering, systems engineering

1. INTRODUCTION

1.1 Ownership Quality

Ownership quality is the customers' perspective of quality during the use of the product. Reliability, maintainability, and serviceability are essential attributes of ownership quality and customer satisfaction (Makino, et al., 1989; Berzak, 1991; Eubanks & Ishii, 1993). Ullman (1992) reports on a survey (*Time*, November 13, 1989) which asked customers, “What determines quality?” He responds “...*quality* is a composite of factors that are the responsibility of the design engineer” and highlights *basic functionality*, *reliability*, and *ease of service* as key quality indicators.

Probabilistic methods for reliability assessment have been a mainstay of engineering systems development for many years (Levinson, 1964: Leemis, 1995). While statistical approaches are useful, they require information available late in the detailed design stage. Product development teams need to build-in reliability at the early stages of design and Failure Modes and Effects Analysis (FMEA) can help address this challenge.

1.2 Background and Shortcomings of FMEA

Failure Modes and Effects Analysis is a key design method to help engineers improve quality of ownership. FMEA is an engineering technique used to define, identify, and eliminate known and/or potential failures, problems, and errors from the system, design, or process before they reach the customer (Omdahl, 1992). What is a failure mode? A failure mode is essentially an undesired cause-effect chain of events. Once the development teams identify and prioritize failure modes, they can make design decisions leading to improved reliability, quality, and safety (Stamatis, 1995). Table 1 explains the three main phases of FMEA.

Table 1. Three aspects of FMEA

Phase	Question	Output
Identify	What can go wrong?	Failure Modes
Analyze	How likely is a failure mode and what are the consequences?	Risk Priority Evaluation (likelihood × severity × detection difficulty)
Act	What can be done to eliminate the cause or alleviate the severity?	Design solutions, test plans, manufacturing changes, error proofing, etc.

McKinney (1991) emphasizes the need to apply FMEA at an early, system level in order to effectively impact the design and reliability of the device. FMEA teams frequently identify failure modes by assessing component failures and their effects. Unfortunately, detailed information on the constituent components is available only after completion of layout design. At this late stage, causes of failures identified by FMEA can be very expensive or impossible to correct. According to Kara-Zaitri et al. (1991) FMEA is almost useless when treated as an after-the-fact "checklist" to satisfy management or contractual agreements with customers. Figure 1 compares the early and continuous application of FMEA to what often happens: performing the FMEA late or not at all.

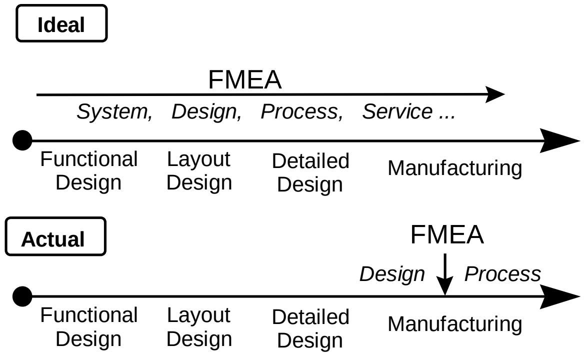

Figure 1. Comparison of ideal and actual deployment of FMEA

The literature does not prescribe detailed procedures for identifying failure modes. A crucial step in predicting the overall reliability of a system is formulating an extensive list of failure modes identifying *what might go wrong*. While FMEA has the potential to predict and mitigate failure modes during design, our industry collaborators have reported difficulty using FMEA to identify system-wide failure modes. Without a systematic approach, engineers produce a subjective analysis depending on their experience level. Some failure modes are difficult to anticipate during the pre-manufacturing stages, including:

- interfaces with other systems
- interaction with controls
- unexpected operating conditions
- unanticipated customer use
- assembly and service errors
- manufacturing variation

Failure modes particularly difficult to identify are those which fall between development teams, especially when teams correspond to systems with functional interdependence. It is challenging to understand operational interactions between systems and to facilitate continuous communication among groups. For example, a new gas/steam turbine (combined-cycle) plant design uses exhaust steam to help cool the gas turbine. This interaction of the gas and steam turbine systems leads to a new complexity in both plant and controls design (Figure 2.) Advanced FMEA provide an means for discussing the interface between independent design teams.

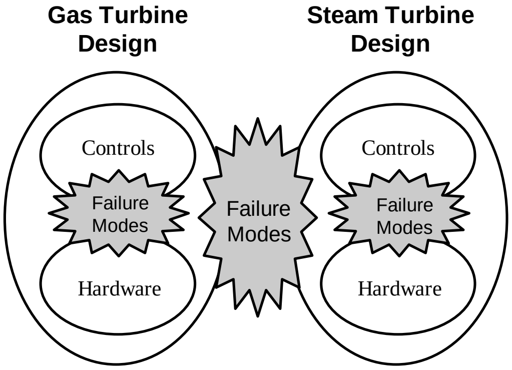

Figure 2. Interfaces between functional teams lead to failure modes that are difficult to anticipate

1.3 Advanced FMEA Using Behavior Modeling

Problems with timing and execution lead to one of the biggest problems with FMEA: not identifying critical failure modes that occur during use. It is impossible to assess the probability and consequences – much less provide design solutions – for a failure scenario that engineers do not anticipate. Our goal in developing "Advanced FMEA" is to provide a systematic approach to identify a comprehensive set of failure modes early in the design process.

Advanced FMEA uses behavior modeling to link desired behaviors with the components, operating environment, related systems, and control logic. Qualitative behavior simulation provides the framework for identifying failure modes and estimating their effects. The proposed method builds on preliminary work by Eubanks (1996) and Eubanks et al. (1997) which used behavior-based AFMEA on an automatic ice maker design. In this paper, we extend the scope of AFMEA to include control systems and their interaction with physical systems. This study illustrates how AFMEA can (1) drive concurrent engineering of products and their corresponding control systems, and (2) provide the foundation for computer support of a structured design method.

Several automated FMEA systems exist particularly in the field of electrical circuit design. Ormsby et al. (1991) proposed a concept for automated FMEA employing qualitative reasoning in a model-based environment to make the analysis extensible to other domains. Price et al. (1995) developed an automated FMEA that combines intended functions of a circuit with a proposed model and analyzes the design's safety. Montgomery et al. (1996) performed a computer simulation of failure modes and their effects for electrical circuits, including qualitative simulation at the early stages. Pelaez et al. (1994) introduced a fuzzy cognitive map to facilitate automated FMEA in mechanical systems. Palumbo (1994) used mode variables and behavioral logic to automate FMEA of an actuator control system. Russomanno et al. (1994) related the FMEA process to various artificial intelligence techniques. These approaches, while powerful, generally require domain specific system models, particularly when modeling electrical circuits and logic.

The next section reviews the concepts of behavior modeling used for Advanced FMEA. Section 3 describes the application of AFMEA to a conceptual design and its corresponding control system. The paper introduces "meta-behavior" modeling as a framework for evaluating systems whose behavior-to-structure constructs are changing in time. A gas turbine cooling system for a conceptual power plant serves as an illustrative example. Section 4 discusses opportunities for AFMEA as an automated concurrent engineering tool, and Section 5 relays specific future activities.

2. BASIC CONCEPTS OF BEHAVIOR MODELING

Conventional design texts use functional decomposition as an early conceptual design tool. The design team describes the overall function of the device and progressively decomposes the required functions in order to manage and understand the design (Suh, 1990: Ullman, 1992). A mapping between the functions and the structure forms a link between the descriptions of the device operation and the physical entities implementing those actions. Figure 3 is an example of a hierarchical function decomposition mapped to the structure.

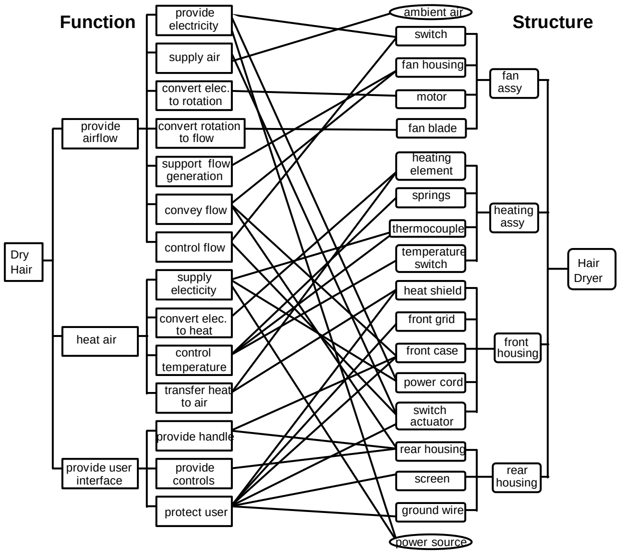

Figure 3. Function-structure mapping for a hair dryer

Using the function-structure map, designers can easily establish cause-effect failure scenarios. A failure in the hardware or supporting systems (such as power) could result in a sub-function not occurring as intended. We define a failure as the negation of a function, or "not(function)." Using this definition, we can capture outright failures where the function does not occur at all, as well as deviations from the intended functions. Figure 4 uses the a function-structure map of a hair dryer to generate two failure scenarios.

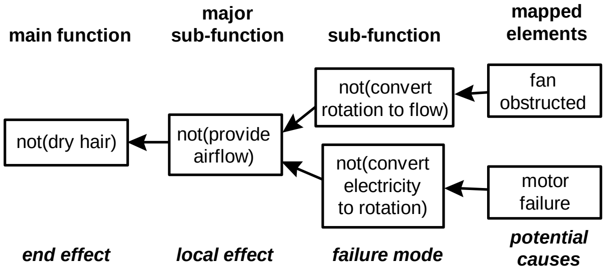

Figure 4. Two failure scenarios from the function-structure map

Similar to a functional decomposition, the behavior model hierarchically decomposes intended behaviors and maps them to physical entities. This definition of behavior modeling builds on functional and causal representation described by Iwasaki and Simon's behavior modeling and simulation (1994). The behavior model provides more information than normal functional representations. Each behavior maps to a specified state transition as well as to the physical systems responsible for the execution of the behavior. State variables are objects with one or more attributes, forming the (<object>, <attribute>, <value>) triple (Eubanks et al., 1997).

<object>	can be any physical or conceptual entity
<attribute>	is an identifiable quality or measurable characteristic of the object
<value>	is a quantification or discrete qualification of the attribute

This construct allows the designer to define easily various properties of materials, flows, components and systems. To simplify modeling, we have adapted a partial state description as described by Chandrasekaran et al. (1993) as a relevant subset of the variables.

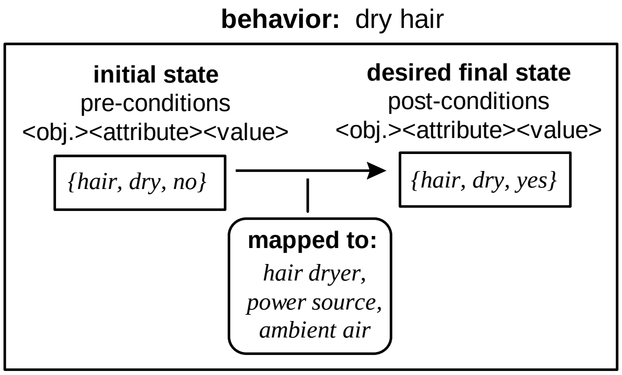

Figure 5. A behavior-structure fragment for a hair dryer

The specification of steady-state pre- and post- conditions builds behavior paths through the model. The behavior paths facilitate reasoning about failure propagation through the system. Failure propagation leads to the assignment of effects for a specific failure mode.

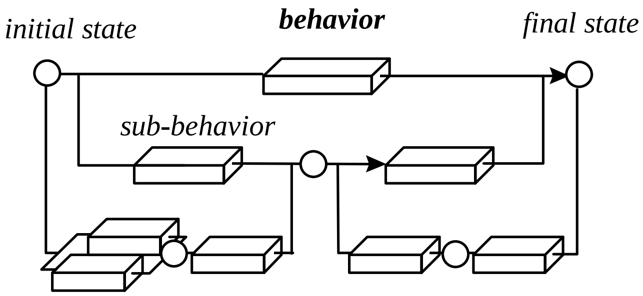

Figure 6. Decomposition of behaviors and intermediate states form inferencing paths

Behavior modeling provides a robust basis for performing AFMEA early during the design analysis for several reasons. Behaviors do not rely entirely on the physical structure of the device. Although physical elements or components change as the design develops, one can use behavior modeling for the following:

- to define general behaviors at the early stages of design,
- to decompose into more specific sub-behaviors, and
- to map to physical systems and components as the design develops.

Figure 7 is an example of an early design schematic, in this case, an air cooling system. Even this basic level of design detail can lead to a behavior model of the system.

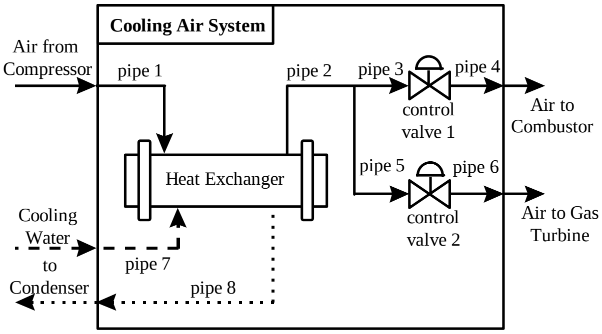

Figure 7. An example design schematic

After setting overall system requirements for the highest level of the behavior model, we can decompose the model into sub-behaviors with physical elements and state changes. Figure 8 shows a simplified example of a behavior model based on the Cooling Air System schematic diagram.

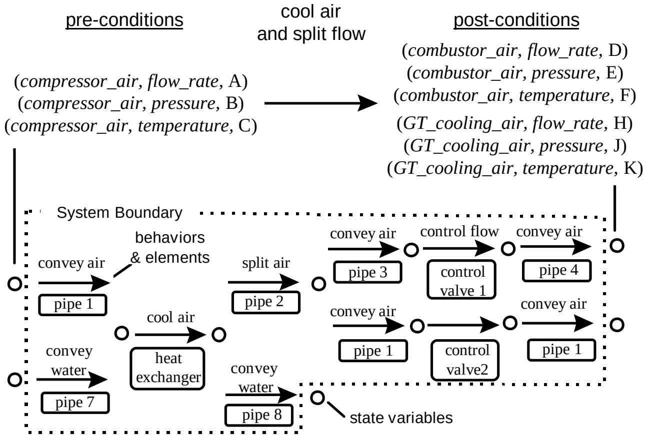

Figure 8. Overview of a behavior model

Behavior modeling provides a systematic framework for generating failure modes compared to brainstorming and other techniques. Moreover, behavior modeling has a greater scope of analysis than a component based failure mode analysis.

3. APPLICATION OF AFMEA FOR CONCEPTUAL DESIGNS & CONTROLS

3.1 Power Plant Gas Turbine Cooling System

This section describes AFMEA as applied to a gas turbine cooling system of a combined cycle power plant. This conceptual design uses steam to cool the gas turbine in order to increase thermal efficiency. Figure 9 shows a schematic diagram of the proposed design.

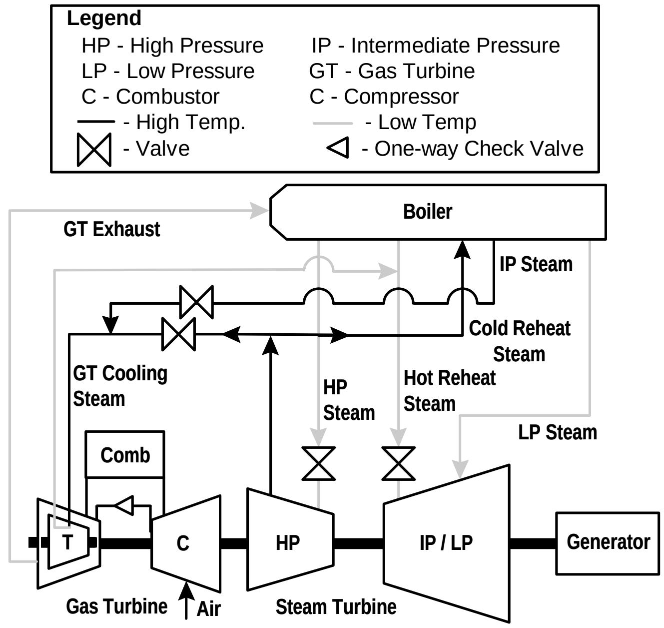

Figure 9. Conceptual power plant using steam to cool the gas turbine

The target example is a single-shaft combined-cycle plant. The unique function of the new design is cooling of the gas turbine blades with exhaust steam from the high pressure (HP) steam turbine. The gas turbine blades, in turn, heat the cooling steam in parallel with the reheater before powering the intermediate pressure (IP) steam turbine. Unfortunately, there is no cooling steam available when starting-up the plant. Thus, the development team proposed three distinct cooling modes during plant start-up. Initially, air from the compressor cools the gas turbine blades. After the gas turbine exhaust has created sufficient boiler pressure, the boiler will provide cooling steam for the blades. When the high pressure steam turbine reaches steady state operating pressure, the GT blades will use the HP exhaust steam for cooling. These distinct cooling modes resulted in three steady state behavior models:

- Air Cooling Mode
- Intermediate Steam Mode
- HP Steam Cooling Mode

The schematic diagrams – used by the gas turbine, steam turbine, and controls engineers – provided the basis for the behavior models. Input parameters and output requirements provided an overall system behavior used for decomposition into detailed sub-behaviors. The behavior models helped identify potential failure modes, causes, and effects. Sub-system engineers, and component specialists, used AFMEA as a basis for discussing design rationale and potential improvements. However, these three analyses did not address failures associated with the transition from one mode to the next.

3.2 Concurrent Engineering Challenges

In previous combined cycle power plants, the gas and steam turbines operated almost independently of each other. The only functional interaction between the two systems was the gas turbine exhaust heated steam in the boilers. Design engineers understood this configuration and its failure modes well and the designs had not changed much from year to year.

The controls for the gas turbines and the controls for the steam turbines were largely independent as were the engineering groups. Because the design was not evolving rapidly, controls engineers could adapt easily to design changes when they occurred. Design changes in the steam turbine configuration did not effect performance or controls of gas turbines and vice versa. For the new concept in plant design, the gas turbine's performance is intimately inter-related with the steam turbine system (Figure 10).

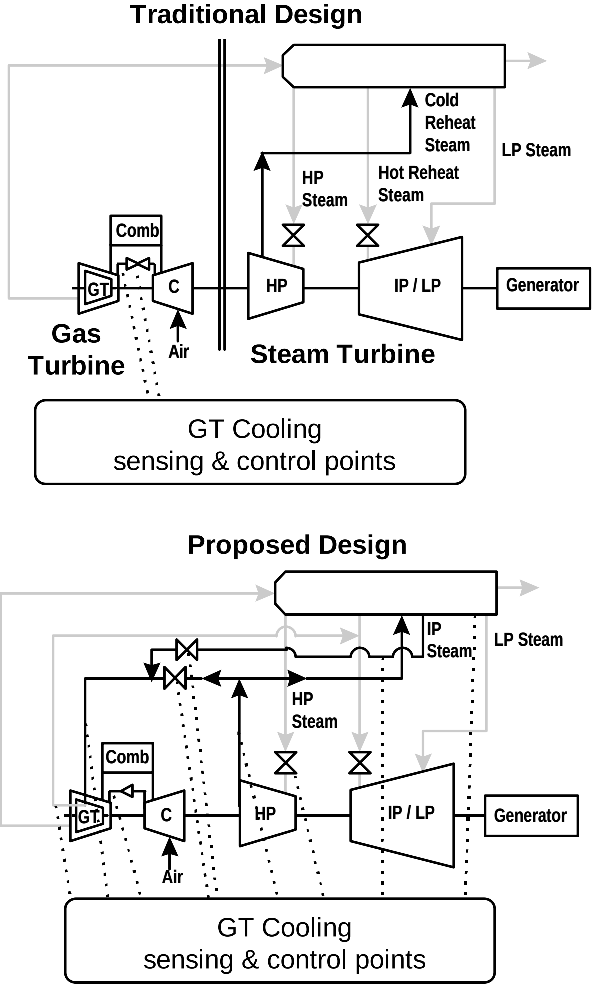

Figure 10. The new GT cooling system is intertwined with the steam system

The controls engineers must now understand both systems and their functional interaction. The authors observed that the controls people were struggling to stay up-to-date with the frequent design changes. The design of the controls followed the system layout design and component selection. Conventional engineering efforts, such as periodic design reviews, were not adequate. Design reviews typically consisted of one functional group while other groups and controls engineers learned of design changes after-the-fact.

3.3 AFMEA for Controls-Hardware Interaction

The performance of the control system is critical to plant performance and safety. A major challenge for the controls engineer is to identify failure modes during the start-up of the power plant when the cooling of the gas turbine (GT) is not at steady state. During start-up, the behavior “cool the gas turbine” goes through three distinct modes: Air Cooling, Intermediate Steam Cooling, and HP Steam Cooling. Within each mode, the behavior model does not change since the source of cooling is constant for each mode.

Since the development teams were concerned about potential failures when changing from one mode to another, we felt the need to augment the existing behavior models. The model should address discrepancies during the transient start-up sequence. In order to extend the scope of the behavior model, we introduce the concept of “meta-behaviors” as behaviors that change behavior-structure relationship. During start-up, the overall behavior “cool the gas turbine” remains constant but the medium responsible for the cooling is changing. Figure 11 illustrates the use of meta-behavior to move from one mode to the next desired mode.

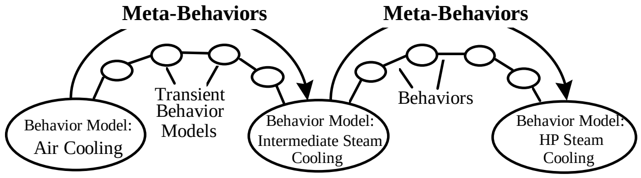

Figure 11. Meta-behaviors describe transition of operational modes

Meta-behaviors represent changes in the governing behavior model of the system. Changing from one behavior model to another generally requires the following:

- gathering information about the state of the system,
- using that information to make a decision,
- acting on the decision.

A human might be responsible for this decision making process or, as in this example, a control system. Figure 12 shows the relationship between the *see* / *think* / *do* process of decision making and the *sensing* / *reasoning* / *actuating* sequence of a control system.

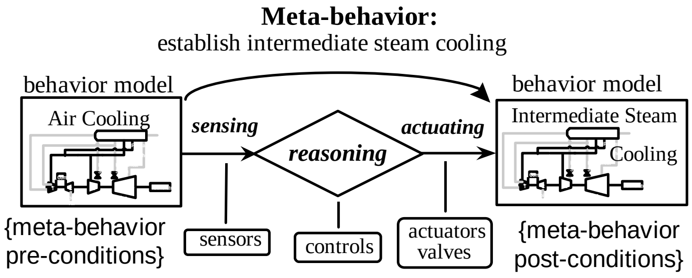

Figure 12. Meta-behaviors transition from one behavior model to the next

The control system essentially acts as the brains or logic of a system, gathering information from the sensors, making decisions, and sending impulses to actuators. In order for a system to operate as desired, the control system must receive accurate information and reason correctly. In addition, the actuators must receive the control command and act upon the command. As with behaviors, meta-behaviors have pre-conditions described by a set of state variables: requirements that must exist in order to enable the meta behavior. When the meta behavior executes, the variable-set updates to reflect the new state of the system. Figure 13 shows the changing of modes through meta behaviors as governed by the control system.

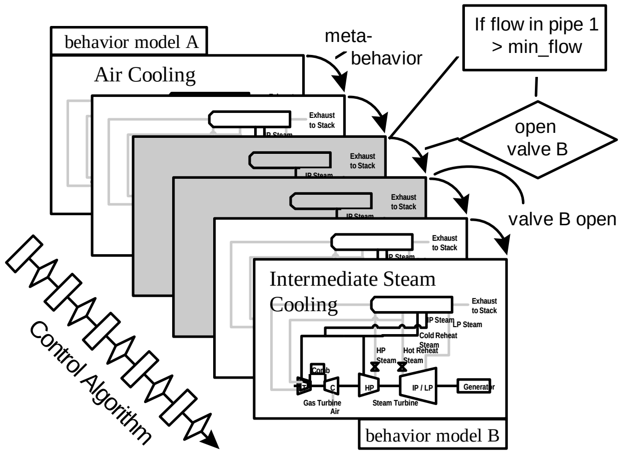

Figure 13. The control algorithm corresponds to meta-behaviors.

3.4 Steps for AFMEA using Meta-behaviors

Our approach for Advanced FMEA of the combined cycle plants’ start-up sequence is as follows

Step 1: Define the initial behavior model and its outputs (post-conditions)

Step 2: Define the final desired "target" behavior model and its inputs (pre-conditions)

Step 3: Identify all intermediate control decisions

Step 4: Translate control decisions into meta behaviors: list the purpose of the control action

Step 5: For each, identify the requisite trigger parameters for the control decision, or pre-conditions (e.g., steam temperature) and list the responsible physical systems (e.g., steam supply system, steam temperature sensors)

Step 6: Identify the desired resultant state of the control action, or post-conditions (e.g., valve position) and the responsible agents (e.g., valve, actuator)

Step 7: For each meta-behavior, tabulate potential failure modes based on pre-conditions, sensing, reasoning, and acting and undesired post-conditions

3.5 Identifying Failure Modes

For the meta-behaviors to execute successfully, the system must satisfy all pre-conditions, sensing accurately, reasoning properly, actuating appropriately, all while being supported by intrinsic systems and hardware. Specifically:

- system state variables must reach specified values (*pre-conditions.*) These variables can be intrinsic (e.g., valve position) or extrinsic (system inputs such as supply steam)
- current state of the system must be sensed and reported correctly (*sensing*)
- control system must recognize sensor signals and other logical inputs (*reasoning*)
- control logic should reason as prescribed (*reasoning*)
- control system must issue appropriate signals (*reasoning*)
- signals must be received correctly by the associated systems (*actuating*)
- associated hardware should comply with the actuation signal (*actuating*)
- supporting hardware & systems must function properly (*intrinsic*)

Each meta-behavior can fail if any of the above requirements breakdown. This study classifies failures into five different categories depending on the nature and timing.

Pre-condition failures result from a requisite "trigger" variable not reaching the state required for the control system to execute an action. If pre-conditions for the action (or meta-behavior) do not exist, then there is no trigger for the control system to take action. For example, sufficient steam flow might serve as a pre-condition for a meta-behavior. If a failure in the boiler system provides in inadequate steam flow, this would result in a pre-condition failure. Theoretically, a steady-state behavior model can identify these failures at that instance.

Sensing failures occur when sensing agents do not report accurately the true state of the systems. Sensing failures include Type I errors (false positives) and Type II errors (false negatives). Other sensing failure could include the lack of a signal or an operator mis-reading a gauge. These failures can result in an improper control action such as a premature action, or no action when the control system intends one.

Reasoning failures occur when the information is available to make a prescribed decision but the control system responsible does not execute the appropriate command. These are essentially control logic flaws to test extensively during software validation and debugging.

Actuating failures occur when the mechanical or electrical system fails to receive or execute the control command, e.g., if a solenoid does not activate or a valve sticks.

Intrinsic failures are failures of the systems or hardware that do not relate directly to the sensing, reasoning, or actuation of the meta-behavior. These failures are problems with the behavior of supporting hardware and systems. As with pre-condition failures, a behavior model for the entire system can help identify these failure modes at quasi steady state "snapshots" of the system state between meta-behaviors. Intrinsic failures might include a valve that changes position when it is not expected or a check valve not closing on positive pressure.

Table 2 summarizes failure categories and corresponding examples. Once the engineers tabulate these failures, they can reason about the effects based on the undesired state of the system. In an actuation failure, for example, if a valve is stuck closed, the system will operate with the valve in the closed position. If a sensor gives a false "high" reading then the controls will act as if the requisite condition were attained. From these failures, one can identify the causes and effects.

Table 2. Meta-behavior Failure Categories

Failure	Responsible System	Example
Pre-condition	input to the system, previous meta-behavior	insufficient steam flow
Sensing	sensors	steam flow sensor fails
Reasoning	control system	no signal to open valve A
Actuating	mechanical	valve stuck closed
Intrinsic	mechanical	pipe failure

3.6 Results of AFMEA Analysis

A team of gas turbine, steam turbine, controls, component engineers, and managers discussed the potential failure modes associated with each meta-behavior and rated each for severity and likelihood. The discussion and ensuing ratings resulted in the following:

- determined “trip” or shut-down protection logic during start-up
- prioritized sensor placement and redundancy for high risk control actions
- initiated detailed FMEA’s for critical components based upon the risk evaluation
- established the basis for a diagnostic engine used for real-time monitoring

In addition to these specific results, Advanced FMEA proved valuable as communication tool. AFMEA facilitated key discussions and information exchange between different groups which might not have happened until much later in the development process. The methodology helped engineers understand their systems’ impact on others as well as on overall performance.

4. OPPORTUNITY FOR COMPUTER SUPPORT

AFMEA is a powerful method for addressing failure modes in the early stages of design. However, the AFMEA is tedious in its present form and the authors see an opportunity for automating the process. The goal of automation should not be to replace face-to-face communication, nor to under-emphasize careful engineering analysis, but rather to enhance both. Careful implementation of an automated approach could hold significant advantages:

- information management: linking behavior, design, and controls
- semi-automated generation of meta-behaviors from control logic
- formalized logic and reasoning
- knowledge and model retention
- speed of analysis
- integration of sub-system analysis and performance with the overall system

Some level of automation of the process is critical for implementing the AFMEA as a product development tool. If automation increases the speed of analysis, then it will provide extra motivation for using the tool at the early stages of product development. Figure 14 shows the relationship of automated AFMEA with design of controls, functional design of hardware, and their interaction.

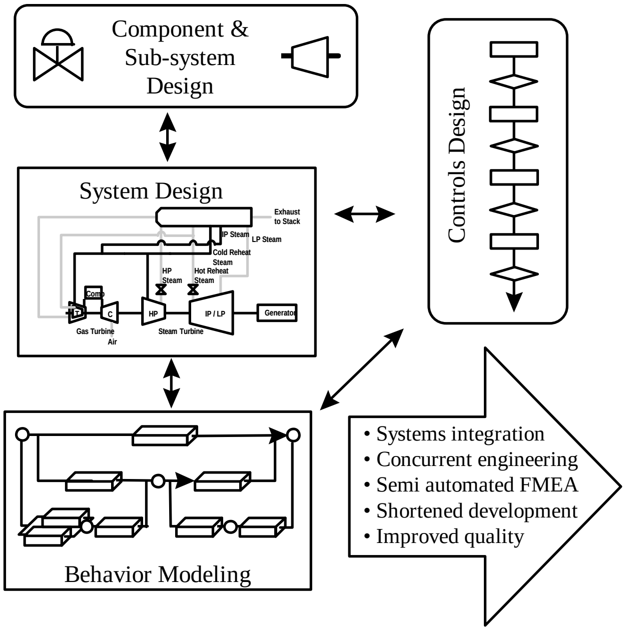

Figure 14. Automation of AFMEA can facilitate communication and expedite product development

5. CONCLUSIONS AND FUTURE WORK

Behavior modeling facilitates concurrent engineering efforts between controls and hardware design. AFMEA is a flexible yet systematic approach, applicable at the early stages of design and throughout development. In addition, the method lends itself to automation as a product development tool. Further extensions of AFMEA include detailed budgeting of reliability, manufacturing process design, prioritization of testing, development of diagnostics and service logistics (Figure 15).

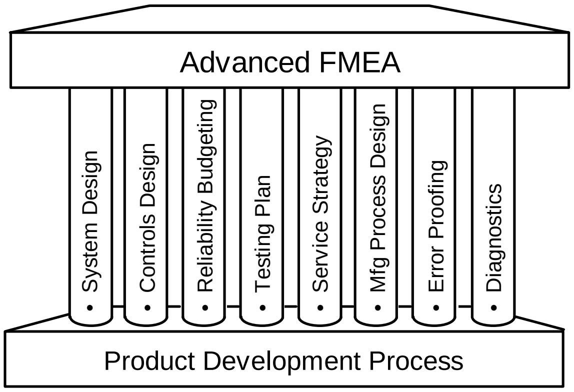

Figure 15. AFMEA benefits parallel aspects of product development

To extend and validate AFMEA as a design methodology, we plan on the following activities:

- extend the behavior model to manufacturing processes
- link AFMEA with design and process Error Proofing
- automate the procedure using software
- document additional case examples
- develop Advanced FMEA training modules

ACKNOWLEDGMENTS

We would like to thank General Electric and the Department of Energy for providing motivation and funding for this research. The authors would like to express sincere appreciation to Dr. Harindra Rajiyah and Brian Ray for facilitating the application of our AFMEA and providing valuable feedback. A special thanks to Dr. C. Fritz Eubanks for his guidance during the development and application of behavior modeling, and to Mark Martin and Jason Grant for comments and help putting this paper together.

REFERENCES

Berzak, N. (1991), "Serviceability by Design," Proceedings. of the 23rd Int'l SAMPE Technical Conference, October 21-24, 1991, pp. 1060-1071.

Chandrasekaran, B., A. Goel, and Y. Iwasaki (1993), "Functional Representation as a Basis for Design Rationale," IEEE Computer, 26(1):48-56, January, 1993.

Eubanks, C. and K. Ishii (1993), "AI Methods for Life-cycle Serviceability Design of Mechanical Systems," Artificial Intelligence, 8(2), pp. 127-140.

Eubanks, C. F. (1996), "A Framework for Computer-based Failure Modes and Effects Analysis of Mechanical Systems in the Conceptual Design Phase," Ph.D. Thesis, Graduate School, The Ohio State University, December, 1996.

Eubanks, C., S. Kmenta, and K. Ishii, (1997) "Advanced Failure Modes and Effects Analysis Using Behavior Modeling," Proceedings of the ASME 9th International Design Theory & Methodology Conference, Sept. 1997, Sacramento, CA.

Eubanks, C., S. Kmenta, and K. Ishii (1996), "System Behavior Modeling as a Basis for Advanced Failure Modes and Effects Analysis," ASME Computers In Engineering Conference, Sept. 1996, Irvine, CA.

Iwasaki, Y. and H. Simon (1994), "Causality and Model Extraction," Artificial Intelligence, Kluwer Academic Publishers, Boston, pp. 597-616.

Kara-Zaitri, C., Keller, A., Barody, I., and Fleming, P., "An Improved FMEA Methodology," Proceedings of the 1991 IEEE Annual Reliability and Maintainability Symposium, pp. 248-252.

Leemis, L. A. (1995), Reliability: Probabalistic Models and Statistical Methods," Prentice-Hall, NJ, 1995.

Levinson, C. (1964), "System Reliability Analysis," Mechanical Design and Systems Handbook, (H. Rothbart, ed.) McGraw-Hill, Inc., New York.

Makino, A., P. Barkan, L. Reynolds and E. Pfaff (1989), "A Design for Serviceability Expert System," Proceedings. of the ASME Winter Annual Meeting 1989, San Francisco, CA, pp. 213-218.

McKinney, B. (1991), "FMECA, The Right Way," Proceedings of the 1991 IEEE Annual Reliability and Maintainability Symposium, pp. 253-259.

Montgomery, T., Pugh, D., Leedham, S. and Twitchett, S. (1996), "FMEA Automation for the Complete Design Process," Proceedings of the 1996 IEEE Annual Reliability and Maintainability Symposium, pp. 30-36.

Ormsby, A., J. Hunt, M. Lee (1991), Towards an Automated FMEA Assistant, Applications of Artificial Intelligence in Engineering VI, eds. Rzevski, G. and Adey, R., Computational Mechanics Publications, Southampton, Boston, pp. 739-752.

Palumbo, D. (1994) "Automating Failure Modes and Effects Analysis," Proceedings of the 1994 IEEE Annual Reliability and Maintainability Symposium, pp. 304-309.

Pelaez, C. E., J. B. Bowles, (1994) "Using Fuzzy Cognitive Maps as a System Model for Failure Modes Effects Analysis (FMEA)", Third Annual FT&T, International Conference, November 1994.

Price, C. J., D. R. Pugh, N. Snooke, J. E. Hunt, M. S. Wilson (1997), "Combining Functional and Structural Reasoning for Safety Analysis of Electrical Designs," *Knowledge Engineering Review* vol. 12(3), pp271-287, 1997.

Russomanno, D. J, R. D. Bonnell and J. B. Bowles (1994), "Viewing Computer-aided Failure Mode and Effects Analysis from an Artificial Intelligence Perspective," *Integrated Computer Aided Design*, Vol. 1 (3), pp. 209-228, 1994.

Stamatis, D.H., *Failure Mode and Effect Analysis: FMEA from Theory to Execution,* ASQC Quality Press, Milwaukee, WI, 1995.

Suh, N. (1990), The Principles of Design, Oxford University Press, New York, 1990.

Ullman, D. (1992), The Mechanical *Design Process*, McGraw-Hill, Inc., New York.