N93-25968

An Architecture for Object-Oriented Intelligent Control of Power Systems in Space

Sven G. Holmquist, Prakash Jayaram, and Ben H. Jansen

Department of Electrical Engineering
University of Houston, Houston, TX

A control system for autonomous distribution and control of electrical power during space missions is being developed. This system should free the astronauts from localizing faults and reconfiguring loads if problems with the power distribution and generation components occur.

The control system uses an object-oriented simulation model of the power system and first-principle knowledge to detect, identify, and isolate faults. Each power system component is represented as a separate object with knowledge of its normal behavior. The reasoning process takes place at three different levels of abstraction: the Physical Component Model (PCM) level, the Electrical Equivalent Model (EEM) level, and the Functional System Model (FSM) level, with the PCM the lowest level of abstraction and the FSM the highest. At the EEM level the power system components are reasoned about as their electrical equivalents, e.g, a resistive load is thought of as a resistor. However, at the PCM level detailed knowledge about the component's specific characteristics is taken into account. The FSM level models the system at the subsystem level, a level appropriate for reconfiguration and scheduling.

The control system operates in two modes, a reactive and a proactive mode, simultaneously. In the reactive mode the control system receives measurement data from the power system and compares these values with values determined through simulation to detect the existence of a fault. The nature of the fault is then identified through a model-based reasoning process using mainly the EEM. Compound component models are constructed at the EEM level and used in the fault identification process. In the proactive mode the reasoning takes place at the PCM level. Individual components determine their future health status using a physical model and measured historical data. In case changes in the health status seem imminent the component warns the control system about its impending failure. The fault isolation process uses the FSM level for its reasoning base.

1 Introduction

Failure to provide a reliable, uninterrupted supply of electrical power under all circumstances may doom space missions. In case of impending or actual failures, decisions will have to be made about rescheduling load demand and/or reconfiguring the power generation and distribution system. These decisions will have to be made fast, often without the help of experienced control room operators, and often relying on incomplete information.

Knowledge-based (or intelligent) control systems have the ability to make decisions, and the capability to learn, and therefore seem ideally suited for the operation of complex systems such as electric power plants and distribution systems. However, practical applications of intelligent controllers are rare, and appear to be based on control strategies that use prewired solutions to a collection of potential problems, and/or use a supervisory planning approach to failure recovery. As a consequence, these systems have no way to deal with unanticipated, or multiple simultaneously occurring faults, and they have little or no capability to adapt to changing environments or to learn from past experiences.

We are working on overcoming these aforementioned limitations by developing an intelligent control system that uses quantitative and qualitative system models based on an object-oriented representation of the components of the physical system to be controlled. The object-oriented representation decentralizes intelligence by equipping each component with knowledge about how to detect its impending failure, and how to act in case of failure. This reduces the time required to detect faults when compared to an approach relying on a single central fault detector. Furthermore, the object-oriented representation can be implemented in a parallel computer, leading to even shorter response times. The intelligent controller will use these models to explore the "optimal" control actions to modify the system performance or operation. Also, by equipping the model components with knowledge about their behavior (e.g., a resistor will "know" how its temperature will rise in response to the voltage and current applied to it), and with memory (e.g., a record of its temperature for the last hour or so), *proactive* autonomous control can be achieved, even with incomplete sensor data.

Expert systems have been applied to the power engineering area before (see [10] for a review), but few such system are beyond the demonstration phase, and all were developed for large-scale, interconnected systems. The most promising approaches involve the use of object-oriented techniques because an object-oriented approach models the causal and functional relationships by inheritance and message passing mechanisms, and the part-of or component hierarchy [7]. Furthermore, objects are complete functional units that lend themselves to parallel implementations more easily than rule-based approaches, which is important for real-time applications.

A fairly small number of applications of object-oriented programming techniques for the intelligent control of power systems have been published [1, 2, 6, 9], with the prototypical system for event diagnosis and operation planning described in [3] being most closely related to our own work. However, it is unclear how much this system relies on reasoning from first principles (if it uses that concept at all), nor does it seem to have progressed beyond its first prototype state. Notwithstanding this criticism, [3] clearly shows that object-oriented, model-based methods are indeed advantageous for problems in control. The theory of model-based reasoning is explained by Kuipers [5]. Model-based systems are especially useful in the diagnosis of multiple faults as shown in [4]. Also, it is argued in [4] that diagnosing faults at multiple levels of abstraction, starting with the most abstract level, and examining the less abstract levels only when there is reason to suspect it, makes the generation of candidate solutions more efficient.

2 Architecture of the power system simulator and controller.

Our work is based on a multi-level model of the system, with intelligence built in at each level in the sense that each component can reason about its real-world state, as opposed to a higher level intelligence that reasons about

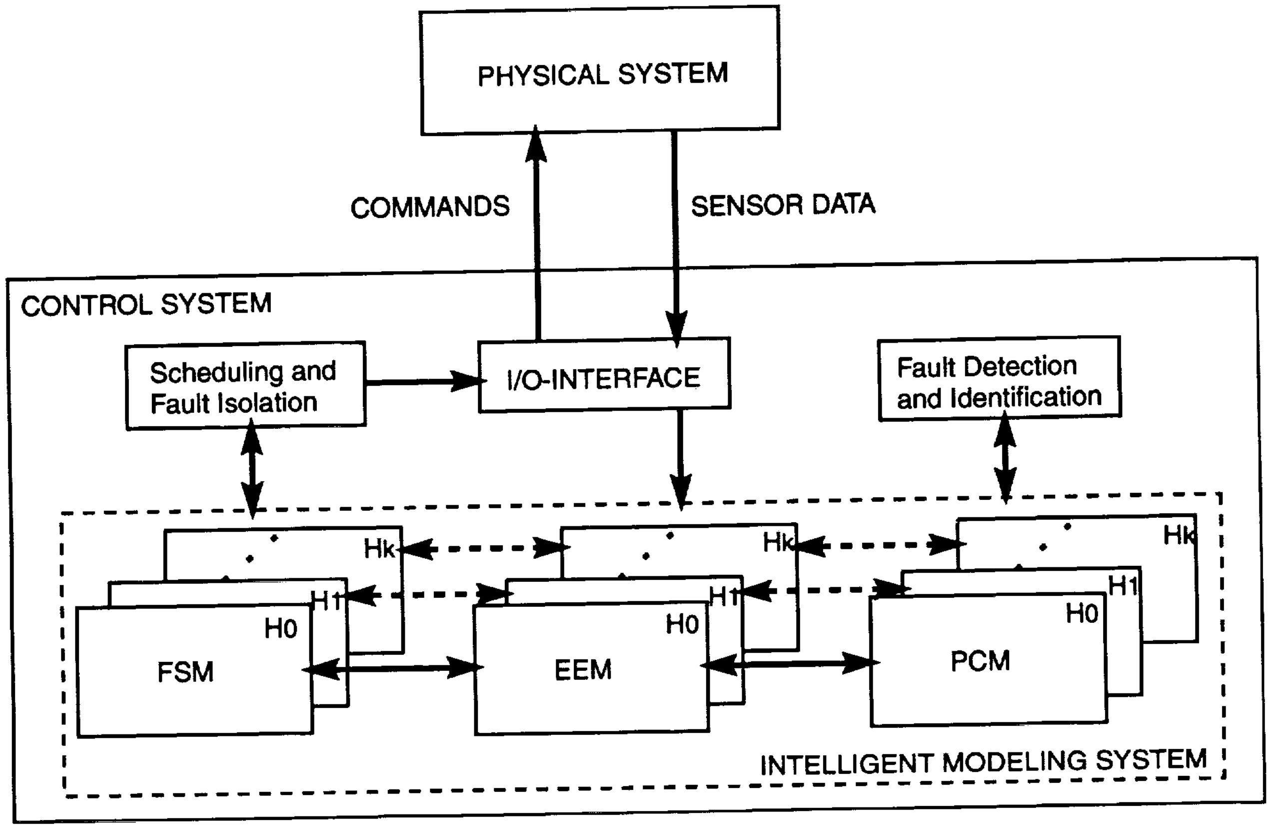

Figure 1: *Overview of the architecture of the model-based, object-oriented control system.*

all the "dumb" lower level objects. Also, the object-oriented design we follow is intended to support concurrency with only a minimal amount of knowledge being exchanged.

2.1 General System Description.

A diagram of the control system is presented in Figure 1. At its core is a model of the system to be controlled. This model represents the physical system under normal operating conditions, and is referred to as the H_0 simulator. At least three versions of H_0 exist, representing the physical system at various level of abstraction. First, there is the Physical Components Model (PCM), containing physically realistic models of the components of the system to be controlled. At the next level of abstraction, one finds the Electrical Equivalent Model (EEM). The latter is a representation of the physical system in terms of power sources, impedances, and switches. The Functional Subsystems Model (FSM) is the most abstract of all, and represents the system in the form of a reduced network in which sub-nets are represented by single functional blocks. An example of the PCM, EEM, and FSM of a simple physical system, consisting of a generator, switches, resistive loads (a light bulb and an electric heater) is shown in Figure 2. The electric heater consists of a fan, i.e., a motor (M1) and a resistive heating element (L2); and the light bulb is denoted by L3.

Each of the three models is an object-oriented representation of the actual system. That is, components are represented as data structures referred to as objects. The latter consist of attributes relating to properties of the component

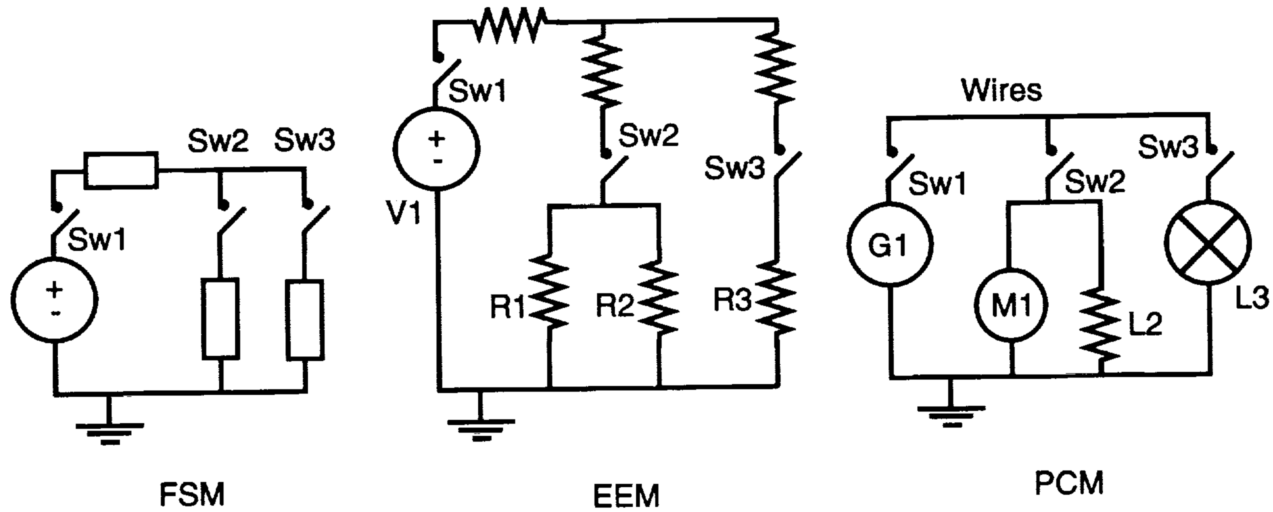

Figure 2: *An example of a Physical Component Model (right), its Electrical Equivalent Model (middle), and its Functional Subsystems Model (left).*

being represented, and attribute-values specifying the values of these properties and/or procedures that can be used to compute these values.

The topological relationships between components in the PCM, EEM, and FSM are specified by attributes describing the connections between the present component and others in the network. Expected voltages at nodes and currents through branches in the EEM are computed using the VIsolver. The VIsolver is an object that solves for the currents and voltages of the power system using the modified nodal formulation [8]. The solution is based on Kirchhoff's current and voltage laws in a matrix form with special considerations taken to reduce the size of the matrices but at the same time keeping it general. This method can be used on networks containing voltage and current sources, impedances, conductances, ideal two-ports, and switches. Historical data, intended for use in the proactive mode, for each component is stored in history attributes. Sensors placed at strategic positions in the physical system (in our case, the physical system is a software simulation as well) provide measurements of voltages and currents in the power system. The PCM and the EEM work in tandem, using the knowledge embedded in them, to detect potential faults. Once faults have been detected (see Section 2.2 below for an explanation of the fault detection process), additional versions of the PCM, EEM and FSM are automatically generated, representing models of the physical system modified in such a way as to account for the hypothesized cause of the fault. For example, H_1 and H_2 may be generated in case two explanations for the fault are possible. Competing hypotheses are eliminated on the basis of comparing future sensor data with predicted values, and/or heuristic reasoning. Once the fault has been determined (identified) remedial action is taken to return the system to a non-faulty state through reconfiguration of loads and sources.

2.2 Fault Detection

Faults may be present if discrepancies between sensor values and expected values are found in the EEM, or if a component in the PCM anticipates impending failure (on the basis of knowledge about the behavior of its physical equivalent, and the historical data available). In other words, the system works in both a *reactive* and a *proactive* mode simultaneously.

As an example of reactive operation, consider the system shown in Figure 2. Assume that voltages and current measurements are available at the output of the generator (V1), the input to the heater (R1 and R2), and the input of the light bulb (R3). Assume that the measured voltage and current at the heater suddenly drops. The voltage at the light bulb will also change slightly, and the current at the source will decrease. Therefore there is a discrepancy between measured sensor values and simulated sensor values and a fault is detected. It is not obvious from the measurements which component is faulty. However, by reasoning using knowledge of the fault models for each component and their health status it is possible to narrow down the number of possibilities and, eventually, the fault can be identified and isolated through simulation (see Section 2.3 for details).

An example of proactive fault detection is the following: Assume that M1 in the PCM finds that its real-world counterpart is about to overheat due to a continuous overload beyond its rating. The M1 object then immediately signals its impending fault state to its equivalent counterpart (R1) in the EEM and tells R1 that the current needs to be reduced. The control system formulates strategies to reduce the current through R1, using the knowledge encapsulated in it (in this case the only possibility is switching off the motor). It is clear that hypothesis selection needs to be based taking into account the importance of the various subsystems in accomplishing the mission objectives. The components in the FSM have knowledge about these aspects, and this knowledge is used to determine which of the reconfigured systems best meets future objectives, and the H_j, that accomplishes this, becomes the new H_0 after the appropriate commands have been issued to the power system.

2.3 Fault Identification

Once the existence of a fault has been detected the location of the fault must be determined. A small change in a single component value can cause many sensors to indicate the existence of a fault. To determine which component has caused the fault (in the reactive mode), branch currents and node voltages are computed using the measured data, and each component's impedance value is computed based on the current running through it and the voltage across it. The EEM component compares its calculated impedance with its "known" impedance and if there is a difference, then the component is suspected of having caused the fault. All components have a health status attribute which is determined by the PCM and verified by the EEM. The PCM determines the health status using heuristics, historical data, and physical knowledge of the component. Hypotheses regarding possible faults are generated, based on the component's health status and impedance discrepancy using the component's fault-model, supplied by the PCM.

The aforementioned approach will work if the environment is sensor-rich, i.e., there are enough sensors in the network to calculate the impedance of *all* components. However, if the environment is sensor-sparse, i.e., there are relatively few sensors in the network, then a strategy will be followed that converts the sensor-sparse environment into a virtual sensor-rich environment. This approach is based on the concept of compound component models. The latter are formed by combining components connected in series, parallel, or in a bridge configuration to a single compound component. Compound components can be part of other compound components. The location of the available volt-meters and current-meters guides the formation of compound models so that in the (reduced) environment the impedance of each compound component can be determined. In other words, the reduced network becomes virtually sensor-rich with respect to the compound components. The impedance and health status of the compound components is calculated based on the impedance, the health status, and the interconnection of the individual components that make up the compound component. The fault identification process can then function in a similar fashion in both a sensor-rich and a sensor-sparse environment. Of course, fault localization can then only pinpoint a compound component as the source of the trouble. However, using the fault models, heuristics, and historical data about the components making up the compound component can be used in a reasoning process to more precisely identify the fault location.

To illustrate the reasoning process, consider the case where a fault has been localized to a compound component consisting of two parallel resistive loads. Suppose that one of the loads is a motor, and the other a heater. Faults occurring in these components will reflect themselves as changes in the component's impedance (e.g., a short will cause a virtually zero impedance). Further, assume that only the voltage across the loads and the total current, but not the currents through each load, are known. In such a case, it will be impossible to determine which load is faulty based on the available measurements alone. However, using fault-models supplied by the PCM, coupled with the assumption that a single fault is considerably more likely to occur than a multiple fault, one or more hypotheses can be generated. For example, the PCM "knows" that a heater's most common failure mode is breakage of the heater element, causing the impedance to go to infinity. Therefore the H_1 hypothesis would replace the EEM of the heater by an infinite impedance, while leaving the EEM of the motor unchanged. In a similar manner H_2 would replace the motor EEM by an impedance reflecting its most prevalent fault state, i.e, a short in the motor coil. The voltages and currents predicted by each of the competing components are compared to the measured data, which will lead eventually to the elimination of all but one hypothesis. This process can be refined by utilizing the concept of the component's "health status". The latter can be used to determine the order in which components should be hypothesized as faulty. For example, the fact that a component has been in service for close to its expected life span, gives it a poor health status and thus it will be hypothesized as faulty prior to components with a good health status. The system will keep track of which components fail, and under what circumstances. This "failure log" is fundamental to the learning capabilities of the system, which will come to "recognize" previously encountered failure modes.

3 Design and implementation of the power system simulator and controller.

We are currently in the process of implementing the previously outlined architecture. The NeXT computer has been chosen as the implementation platform. The NeXT supports Objective-C and has extensive graphical interface capabilities.

The power system simulator has been designed and implemented. A graphics-based tool has been developed to interactively configure the power system to be simulated. A panel with icons, representing components typically encountered in a power system, is presented, and the user can "click-and-drag" these icons in the desired position in the power system window. The specifications for each component are entered by changing the attribute values, in an inspector window, for the component. The resulting power system can simulate voltage sources, switches, and resistive loads. We are only considering direct currents at the present, but a generalization to alternating currents is kept in mind.

A schematic diagram of the power system is shown on the screen in a power system simulator window with the component values and currents and voltages displayed next to each component. The power system's voltages and currents are calculated by the simulator's VIsolver. The VIsolver is an object that solves for branch currents and node voltages for any electric network including power systems using the nodal admittance matrix. The solution is based on Kirchhoff's current and voltage laws in a matrix form with special considerations taken to reduce the size of the matrices but at the same time keeping it general.

Changes in switch settings, load resistance, and source voltages can be made through an event queue or by clicking on the component in the schematic drawing of the power system. The event queue is editable and is used to insert faults into the power system. The power system's voltages and currents are automatically recalculated when the power system simulator receives an event or a switch position is changed by clicking on the switch with the mouse. The events are sent to the power system one after the other in order of occurrence in time.

A control system that reads data from the power system simulator has been implemented. It is possible to set which voltages and currents the control system can receive from the power system by inserting volt-meters and current-meters at the desired positions in the network. The data is displayed in a separate control system window containing the same diagram as shown in the power system simulator window. The control system is capable of issuing commands regarding switch settings to the power system. The control system is capable of forming compound models of components in series, parallel, and bridge configurations.

4 Future developments.

At present, a component library is being built for commonly used electric power components, including DC-motors, generators, circuit breakers. These components, with their embedded knowledge, form an important part of the fault detection system.

The current speeds of execution of the system suggest that parallel implementation is necessitated in order to achieve real-time implementation. Though we lack the hardware for such an implementation, a successful attempt has already been made at executing the various tasks in the program concurrently on the same processor using separate threads. We expect to implement the final system with a fair amount of distributed processing over a network of NeXT computers, so that each task will have its own processor, with the goal of achieving significant speed-ups.

Acknowledgements

This worked has been sponsored by grants from NASA/Johnson Space Center, and the Energy Laboratory of the University of Houston.

References

[1] Gholdston, E. W., Janik, D. F., and Newton, K. A. (1989). Hybrid approach to space power control utilizing expert sys-

tems and numerical techniques. *Proceedings of the Intersociety Energy Conversion Engineering Conference,* Piscataway, NJ: IEEE-Press, 177-182.

[2] Kao, C.Y., and Morris, W.S. (1989). Spacelab life sciences-1 electrical diagnostics expert system. *Telematics and Informatics, 6,* 201-220.

[3] Keronen, J.J. (1989). An expert system prototype for event diagnosis and real-time operation planning in power system control. *IEEE Transactions on Power Systems, 4,* 544-550.

[4] de Kleer, J., and Williams, B.C. (1987). Diagnosing multiple faults. *Artificial Intelligence, 32,* 97-130.

[5] Kuipers, B. (1986). Qualitative simulation. *Artificial Intelligence, 29,* 289-338.

[6] Lee, S.J., Yoon, S.H., Yoon, M.C., and Jang, J.K. (1990). Expert system for protective relay setting of transmission systems. *IEEE Transactions on Power Delivery, 5,* 1202-1208.

[7] Levitt, R.E., and Dym, C.L. (1991). *Knowledge Based Systems in Engineering.* New York: McGraw-Hill.

[8] Vlach, J., and Singhol, K. (1983). *Computer Methods for Circuit Analysis and Design.* New York: Van Nostrand Reinhold Company.

[9] Walls, B. (1989). Starr: An expert system for failure diagnosis in a space based power system. *Proceedings of the Intersociety Energy Conversion Engineering Conference,* Piscataway, NJ: IEEE-Press, 303-306.

[10] Zhang, Z.Z. , Hope, G.S., and Malik, O.P. (1989). Expert systems in electrical power systems - A bibliographic survey. *IEEE Transactions on Power Systems, 4,* 1355-1361.