

2

AD-A254 342

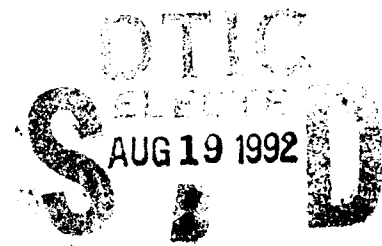


ON-ORBIT SUPERVISOR FOR CONTROLLING SPACE STATIONS

Richard J. VanderVoort

**Dynacs Engineering Co.
34650 US Highway 19 North, Suite 301
Palm Harbor, FL 34684-2157**

July 1992



Final Report

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

92-22991



**PHILLIPS LABORATORY
Directorate of Space & Missile Technologies
AIR FORCE MATERIALS COMMAND
EDWARDS AIR FORCE BASE CA 93523-5000**

9 2 8 18 042

NOTICE

When U.S. Government drawings, specifications, or other data are used for any purpose other than a definitely related Government procurement operation, the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise, or in any way licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use or sell any patented invention that may be related thereto.


FOREWORD

This Report was prepared by Dynacs Engineering Co, Palm Harbor FL, under contact F29601-91-C-0033, for Operating Location AC, Phillips Laboratory (AFMC), Edwards AFB CA 93523-50. Project Manager for Phillips Laboratory was Capt Richard M. Martin

This report has been reviewed and is approved for release and distribution in accordance with the distribution statement on the cover and on the SF Form 298.



RICHARD M. MARTIN, Capt, USAF
Project Manager



RANNEY G. ADAMS
Public Affairs Director



L. KEVIN SLIMAK
Chief, Structures and Controls Division
Directorate of Space and Missiles Technology

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE JULY 1992	3. REPORT TYPE AND DATES COVERED FINAL APRIL 3, 1992	
4. TITLE AND SUBTITLE ON-ORBIT SUPERVISOR FOR CONTROLLING SPACECRAFT		5. FUNDING NUMBERS C: F29601-91-C-0033 PE: 65502F PR: 2864 TA: 00DE	
6. AUTHOR(S) RICHARD J. VANDERVOORT			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) DYNAC ENGINEERING CO 34650 US HIGHWAY 19 NORTH, SUITE 301 PALM HARBOR, FL 34684-2157		8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) OLAC-PL/VTSS EDWARDS AFB, CA 93523-5000		10. SPONSORING/MONITORING AGENCY REPORT NUMBER PL-TR-92-3024	
11. SUPPLEMENTARY NOTES COSATI CODES: 22/02			
12a. DISTRIBUTION/AVAILABILITY STATEMENT APPROVED FOR PUBLIC RELEASE: DISTRIBUTION IS UNLIMITED		12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) Spacecraft systems of the 1990's and beyond will be substantially more complex than their predecessors. They will have demanding performance requirements and will be expected to operate more autonomously. This underscores the need for innovative approaches to Fault Detection, Isolation and Recovery (FDIR). A hierarchical expert system is presented that provides on-orbit supervision using intelligent FDIR techniques. Each expert system in the hierarchy supervises the operation of a local set of spacecraft functions. Spacecraft operational goals flow top down while response flow bottom up. The expert system supervisors has a fairly high degree of autonomy. Bureaucratic responsibilities are minimized to conserve bandwidth and maximize response time. Data for FDIR can be acquired local to an expert and from other experts. By using a blackboard architecture for each supervisor, the system provides a great degree of flexibility in implementing the problem solvers for each problem domain. In addition, it provides for a clear separation between facts and knowledge, leading to an efficient system capable of real time response.			
14. SUBJECT TERMS Expert system; Real-Time Supervisor; Fault Detection and Recovery; Distributed Artificial Intelligence; On-Orbit Health Monitoring			15. NUMBER OF PAGES
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT SAR

Contents

1	Introduction	1
1.1	Problem Identification	1
1.2	Scope of the system	4
2	System Requirements Definition	6
3	Proposed Design	8
3.1	Architecture Overview	8
3.2	Blackboard	10
3.3	Planning and Scheduling	12
3.4	Types of faults	13
3.5	Failure Detection, Isolation and Recovery	14
3.5.1	Output Variance Control	15
4	Design Issues and Approaches	16
4.1	Constraints	16
4.2	Design Issues	17
4.2.1	Interaction between experts	18
4.2.2	Subsystem autonomy	19
4.3	Design Philosophy	20
5	Demonstration System	22
5.1	Purpose	22
5.2	Overview	22

DTIC QUALITY INSPECTED 3

Accession For	
NTIS	<input checked="" type="checkbox"/>
DTIC	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution	
Availability Codes	
Dist	Avail and/or Special
A-1	

List of Figures

1	Hierarchy of expert systems	3
2	ASTREX testbed	9
3	Blackboard overview	11
4	Generic Supervisor Domain Blackboard	12
5	Demonstration System	23
6	Scenario 1	24
7	Scenario 2	25

1 Introduction

This report outlines the work performed under Phase I of this contract. The remainder of this document is organized as follows: This chapter identifies and defines the scope of the problem. The system requirements are then presented in the second chapter. The third chapter discusses a proposed design. Following this the design issues considered and the description of a demonstrator system are presented in the two chapters following.

1.1 Problem Identification

Modern spacecraft are among the most complex machines of our age. Spacecraft consist of several complex interacting subsystems, such as the structure subsystem, the thermal management subsystem, the electrical power subsystem, the communications subsystem, and the guidance, navigation, and attitude control subsystems. These subsystems are not only very complex by themselves, but also have intricate interdependencies which seriously affect the overall reliability of the spacecraft. Future spacecraft systems will be substantially more complex than their predecessors. Factors contributing to increased complexity are:

- An increasing number of spacecraft are being designed with multiple mission goals.
- Many spacecraft are no longer individual entities performing their single missions. They are part of a set of cooperating systems acting in unison to achieve complex objectives.
- Spacecraft systems are more autonomous with much less ground interaction or control.
- Spacecraft systems are being designed with demanding performance requirements. Performance requirements may be precision pointing, shape control, fast and accurate maneuvers, etc.

This increased complexity of space systems conflicts with the added requirement of extended operational life.

Increasing the lifespan and mission effectiveness of spacecraft involves improving the reliability of spacecraft components and incorporating Failure

Detection, Isolation, and Recovery (FDIR) systems which provide fault tolerance. Fault tolerance contributes to the reliability, maintainability, and survivability of spacecraft.

Traditional FDIR techniques provide limited levels of fault tolerance in modern spacecraft because of the large number of components in spacecraft subsystems and their complex interactions. Spacecraft subsystems may contain several layers of redundancy to provide a basic level of fault tolerance. Failure recovery may be "hard wired", limited to a fixed set of failures, or even based on decisions made from the ground control station. These systems are limited in their FDIR capabilities since:

- Potentially catastrophic failures (ie. structural damage) may require complex reconfiguration capabilities.
- Even simple failures may require real-time modifications in order to prevent performance degradation.
- Failure detection, as well as isolation and recovery may require intelligent decision making abilities.
- System level decisions require a knowledge of the overall architecture and inter-dependencies, which may be available only on the ground.

To achieve higher levels of fault tolerance intelligent systems for FDIR have been explored. Specifically, a hierarchy of expert systems for the On-Orbit Supervisor for Controlling Spacecraft has been investigated as depicted in Figure 1. By incorporating expert capabilities of reasoning methods, as rules of inference and factual information, on-orbit FDIR abilities can be made more robust and reconfigurable. Characteristic of experts is their ability to trace symptoms to faults in situations which had not been predicted previously. A system with such capabilities can respond to situations for which no "hard wired" solution exists. Additionally, experts have the ability to dynamically respond to performance constraints and apply different strategies to problem solving. Most importantly experts utilize their knowledge to efficiently solve problems, rather than use combinatorially explosive brute force techniques. Faced with the increasing complexity of spacecraft subsystems and their interactions, the application of intelligent methods promise to tame FDIR for future spacecraft systems.

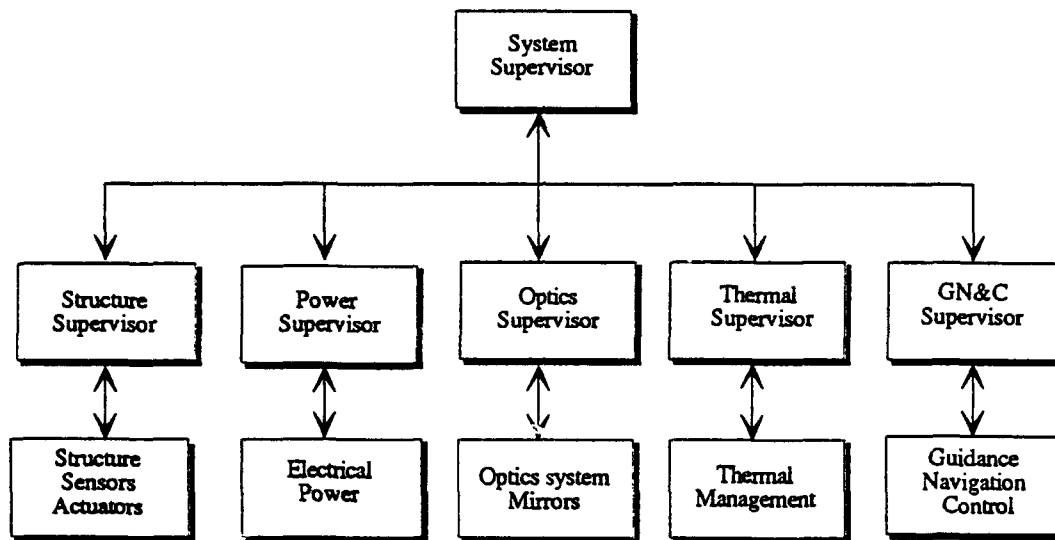


Figure 1: Hierarchy of expert systems

Using the hierarchal expert systems has shown several advantages:

- Modularizing expertise decreases the complexity of the expert system.
- Different approaches to problem solving and knowledge representation can be chosen for each module to increase its efficiency and effectiveness.
- Redundancy can be built into modules through duplication of services. (Functional Redundancy)
- Information analysis using different knowledge sources can be utilized.
- A hierarchical system of semi-autonomous experts is suited to distributed processing, which is becoming a standard in modern spacecraft.
- Expertise can be located near data sources requiring analysis. Transmission of analytical result reduces data transmission requirements.

- The ability to structure problem solving into self-contained processing modules makes the system more resilient to hardware and software errors than a monolithic system.

1.2 Scope of the system

The major tasks of the on-orbit supervisor are:

- Planning and scheduling of spacecraft operations and diagnostics
- Recovery and Reconfiguration enable the mission to continue when recoverable faults are encountered
- Isolation of the source of a fault
- Prediction of the behavior of components and systems
- Detection of faults by comparing predicted and observed behavior
- Execution of operations and their monitoring
- Controlling and coordinating the tasks of the on-orbit supervisor

Performing FDIR is one of the primary tasks of the supervisor. The various phases involved in this task are first defined below.

Failure detection is the identification of actual behavior from predicted behavior. For example, a state estimator may be used to predict future behavior of a device. A fault is detected by sensing the behavior of a component or components and comparing the actual values with values predicted by the state estimator.

Failure Isolation is the process that uses information about a fault to locate a component or components whose misbehavior is responsible for the fault. Isolation may be broken down into two steps. The first step is the process of identifying candidate components which may cause the fault. Component(s) are then singled out which are responsible for the fault(s). Isolation should be capable of identifying faults in the monitored device and faults in the monitor. Proper identification of failed component(s) assure reliable information is used during the spacecraft's mission. Reliable identification of faulty components assures that healthy components are available for use.

Failure Recovery and Reconfiguration attempts to restore performance and stability to within acceptable limits. Recovery and reconfiguration may modify the model or gains when the plant is altered. Missing sensor information may be reconstructible from other measurements. Controls may be designed off-line for all possible failures or may be redesigned on-line.

Real-time control of the processes of failure detection, isolation and recovery is also necessary. FDIR must respond dynamically to changes in the mission and to results of the subproblems of failure detection, isolation, and recovery. Alternate sources of information may be utilized to add confidence to facts or results derived from those facts.

Diagnostics may be planned to detect failed, degraded or fragile components. By performing routine diagnostics, faults may be isolated before they affect performance. Planning takes into consideration availability of resources, relevance of the the tested device to future mission exercises, detection of degrading components, and prediction of component degradation.

In addition to FDIR, the on-orbit supervisor is responsible for planning and scheduling of spacecraft operations. The on-orbit supervisor will execute and monitor operations. Running routine diagnostics will also be the on-orbit supervisors responsibility.

2 System Requirements Definition

This section describes the generic tasks which the On-Orbit Supervisor for Controlling Spacecraft must perform.

1. Detection of faults. Fault detection is identified as part of the task of monitoring. System status information triggers alarms when the behavior of a device(s) does not fit within the range of normal operation. What constitutes an alarm condition is context dependent.
2. Isolation of faults. The isolation of faults is a diagnostic problem. Diagnosis is based upon the interpretation of some potentially noisy data. To accomplish the task a diagnostician must have knowledge of the systems organization, relations between components and how the subsystems interact. The process of diagnosis is made difficult by intermittent failures, noise in data which may mask the problem, failures in diagnostic equipment, the inaccessibility of data, and the masking of faults due to multiple failures. The diagnostic process can be broken down into finding candidates components which may have caused the fault and the singling out of the component(s) which are responsible for the fault(s).

When data is insufficient or inaccessible for diagnosis, available data may be interpreted to find a plausible explanation of events. The explanations must be consistent with known data and correct with respect to properties of the system. Since data may be inconclusive, incorrect, and inconsistent interpretations must be capable of using partial descriptions, contradictory information, and the correctness of the interpretation is also suspect. The line of reasoning of such systems may be quite extensive.

3. Recovery and reconfiguration when faults occur. Recovery and reconfiguration may require the system to adapt itself to a new set of capabilities. This may require redesign of a control in catastrophic circumstances. Information from alternate sources substitute for data which is now inaccessible or unreliable. Recovery may be necessary from a nuisance trip for the system to reestablish its normal operation. Recovery and redesign may come from predetermined plans for certain failures.

4. Prediction of system behavior. Estimates of behavior will be in the time domain and may be generated both quantitatively and symbolically. For example, a Kalman Filter can be used to estimate the state given a component model. Symbolically we would expect the value of a variable derivable by separate systems to be the same.
5. Consistency enforcement. Data from sensors may be unreliable. Results based upon unreliable data may be inconsistent with accurate sources. Likewise results from different methods of derivation may be inconsistent. The role of consistency enforcement is to assure correct information is utilized and incorrect or unreliable data does not propagate throughout the system. Consistency enforcement has local and global considerations.
6. Fault impact. Assessment of the impact of a fault not only can help minimize degradation of the plant or mission (when there really is a fault), but also yields predictions about system behavior which may be useful for detecting nuisance trips, a false indication of failure. If system behavior is predicted assuming a fault has occurred, the predicted behaviors are not found there is reason to believe a fault has not occurred. The impact can be assessed locally and globally.
7. Diagnostics may be planned to detect failed, degraded or fragile components. By performing routine diagnostics, faults may be isolated before they affect performance. Planning may take into consideration availability of resources, relevance of the tested device to future mission exercises, detection of degrading components, and prediction of component degradation.

3 Proposed Design

Based on the requirements laid down in the previous section, along with all the design considerations discussed, a design is proposed for the Supervisor. It is proposed that the ASTREX facility be used to demonstrate a prototype of such a system. The remainder of this section is organized as follows. First an overview of the architecture of the hierarchical expert system is presented. This is followed by a discussion on the Scheduling and Planning algorithms. The different types of faults that can occur in a typical spacecraft system are then discussed. This is followed by a section that addresses Failure Detection, Isolation and Recovery.

3.1 Architecture Overview

The ASTREX test article will be used to demonstrate the supervisor system. The supervisors will interface with the controller, sensors, actuators, structure, and power subsystems (Figure 2). Three subsystem supervisors will be used (one each for the control, power and structure) along with a system level supervisor. Mission objectives will be set and programmed into the system level supervisor. Failures will be induced in each of these subsystems and the system will then act to achieve the mission objective using the appropriate FDIR techniques.

The supervisors are both goal directed and reactive. Supervisors will plan actions to accomplish mission goals, follow through with their execution, and monitor system health. They also react dynamically to faults that occur in their subsystems or occur in other subsystem and affect their operations. Each of the supervisors is semi-autonomous. Supervisors have their own problem solving capabilities and can provide FDIR services without the aid of their supervisor and peer supervisors. Each supervisor is responsible for FDIR of a specific subsystem or component. However, they are all working toward the common goal of system wide FDIR.

Communication between supervisors may be directed toward specific supervisors or broadcast to all supervisors. Each supervisor knows which supervisor should be notified when particular values are updated or certain events occur. Messages are categorized as Warnings, Expectations, Commands, Reports, and Questions. Messages are prioritized as low, medium, and high. Messages

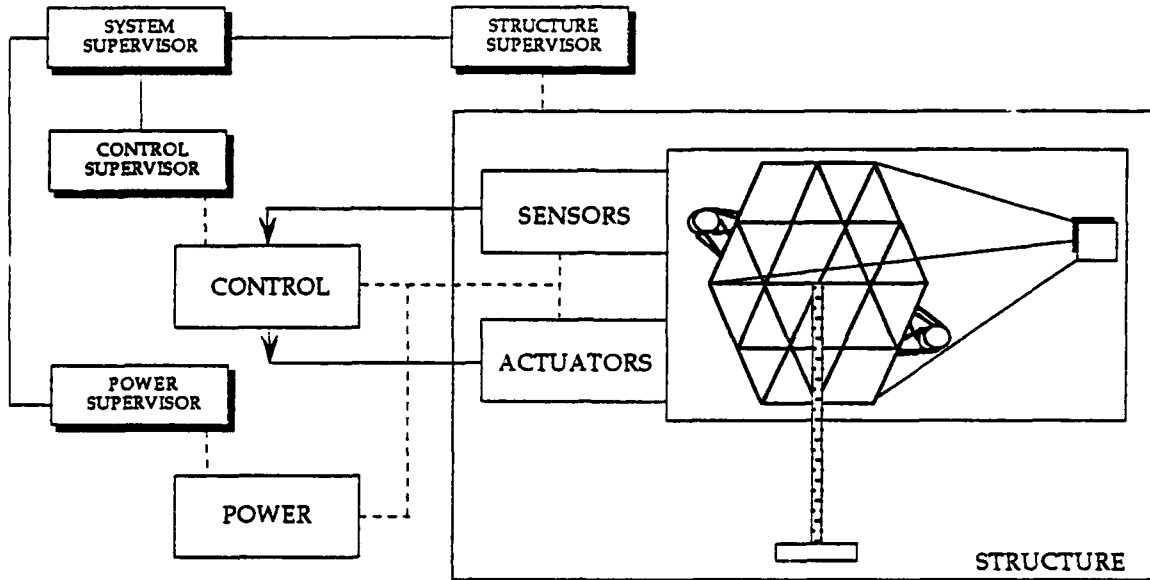


Figure 2: ASTREX testbed

from a supervisor's supervisor are weighed more heavily than others.

Messages may convey warnings about a possible situation which may be occurring. Warnings are issued as a precaution and are issued based on tentative results. Supervisors may transmit expectations of some event occurring. Expectations inform about future events such as operations to be executed, parameter level to exceed alarm levels, and changes of status. Commands flow top down. For each command, the executor will report on the success or failure of execution along with pertinent status information to its supervisor. Questions seek to attain information from other supervisors. In reply a Report will be issued.

For each of the tasks that the supervisor must perform, different constraints define the representation and processing methods. Forward or backward inference methods and confidence factors are among the choices to be made. Additionally, the different tasks must share information with each other.

3.2 Blackboard

To satisfy the above constraints and the distributed aspects of the supervisors a blackboard architecture is proposed for each supervisor (Figure 3). A blackboard separates the facts and knowledge used for each of the supervisor's tasks into small manageable units. The separation of facts and knowledge for each task enables the supervisor to switch its activity quickly and appropriately to suit the current situation. This approach offers a great deal of flexibility, since the most efficient representation can be chosen for each problem domain. The blackboard approach is also easily amenable to distributed processing.

A blackboard system consists of a global database called a *blackboard* and its associated problem solvers or *Knowledge Sources* (KSs) [7],[9]. The *blackboard* is separated into control and domain databases. The control and domain databases are subdivided into different areas of knowledge. Domain problem solvers (KSs) perform tasks such as fault detection, fault isolation, fault recovery and reconfiguration, state estimation, and mission planning. The control problem solver decides which pending domain problem should be solved next. In expert system terminology a *blackboard* is a factbase and a *knowledge source* is a rule base. Each KS employs its own method of problem solving. KSs share information through the objects in the *blackboard*. Information sharing among supervisors is done by message passing.

Consider a scenario on the ASTREX testbed where the mission objective is to perform a large angle slew maneuver. The reaction wheel control system as well as the 8lb and 200lb thrusters are available. The control problem solver in the system level supervisor will select the "Planning" KS for execution (see Figure 4) which in turn will write to the *blackboard* the parameters defining the maneuver. The "Planning" KS will, for example, attempt to optimize fuel consumption or maneuver time depending on mission criticality, and provide the appropriate slew rates or acceleration profiles. The system supervisor will then record in the I/O database a message to be sent to the GN&C supervisor containing the slew commands and profiles.

Within the GN&C supervisor, the local "Planning" KS will be selected to determine the combination of thrusters and/or reaction wheel commands to use [14]. Next the "Execution" KS which is basically the interface to the

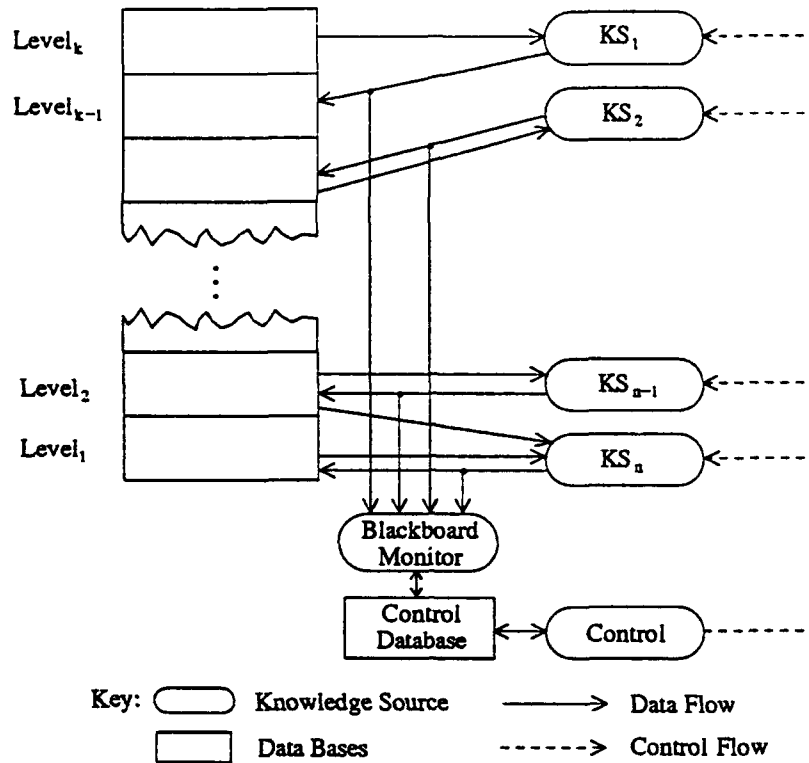


Figure 3: Blackboard overview

subsystem itself will be selected. The supervisor will then go into a monitoring phase where the "Sensing and Detection" KS will be selected. If a failure is detected then the "Fault Isolation" and "Recovery and Reconfiguration" KSs will be selected in sequence. The recovery process for example could involve the use of redundant actuators to compensate for the failure.

The generic structure for a supervisor is given in Figure 4. The domain blackboard is divided into eight segments. The segments are I/O, Planning, State Description, Recovery & Reconfiguration, Diagnosis, Prediction, Observation, and Execution. The I/O segment contains data for communication with other supervisor blackboards and with the spacecraft subsystems. The Planning segment holds data for planning mission activities, health monitoring, and coordinating with other supervisors. The State Description segment contains a description of the components under command of the supervisor and their status. The Recovery and Reconfiguration database has plans for recovery and reconfiguration including canned controls and control redesign procedures. Isolation data includes candidate faulty components, isolated faulty components, hypothesis about actual spacecraft behavior, knowledge

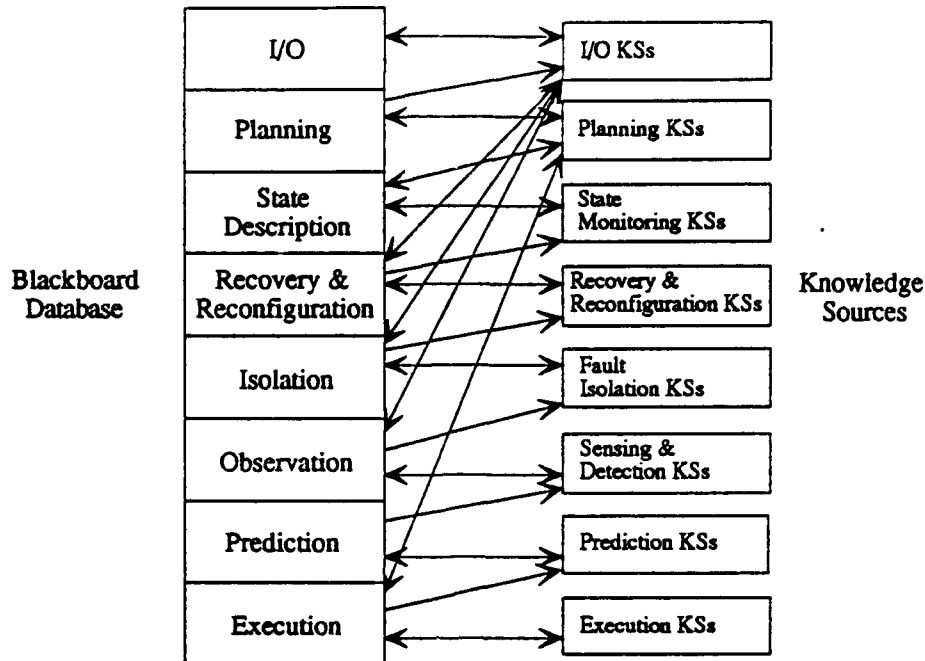


Figure 4: Generic Supervisor Domain Blackboard

of alternate sources of parameters, and functional and behavioral knowledge of the spacecraft. The Prediction segment contains information about the effects of actions and predictions of the state including the state estimate. The Observation space includes sensory and comparison information. The Execution segment contains knowledge of how to command components and how to react to failures.

3.3 Planning and Scheduling

Supervisors will need to plan operations. Planning is necessary for command execution, routine diagnostics, dynamic response to new situations and the satisfaction of real-time constraints. The system supervisor will also plan the mission. Mission planning requires the system supervisor to effectively use system resources and cope with degradation of performance and capabilities.

Routine diagnostics may be used to identify faulty components before their use. Diagnostics may be scheduled at regular intervals. The intervals may be changed if a pattern of faulty behavior is noticed. In addition to exercising components, diagnostics will be used to detect problems before alarm

levels are reached. Information from various parts of the spacecraft will be integrated to perform this task.

3.4 Types of faults

Failures in spacecraft subsystems can cause unexpected and/or undesirable conditions. These conditions may result in the loss of precision pointing capability, the loss of attitude control causing excessive spin rates, tumbling, overheating, and other system malfunctions. The failures in spacecraft operations can be attributed to failures in individual subsystems. A representative set of subsystem failures that can occur on the ASTREX facility are listed below.

- Supervisor failure
- Structure failures
- Sensor failures
- Actuator failures
- Air supply inadequacies
- Power supply/transmission failures

Structural failures could include for example the loosening of joints in the truss. In addition the structural characteristics can be changed by stiffening or softening active truss members. The controller can then be made to reconfigure to adapt to this change. Sensor failures could be classified as partial and total failures. Partial failures could be caused by excessive noise, loss of calibration or bias/offset/drift errors. Total failures will cause the sensor to become unusable. Actuator failures will be simulated by shutting off the thrusters or by reducing the thrust level. Failures can also be induced in the reaction wheels. Power supply failures can cause a component to be completely cut off, or cause a "brown-out" condition. Also, a partial failure within the power unit can result in reduced wattage output. This in turn will require decisions to be made on which components are essential and need to remain active.

3.5 Failure Detection, Isolation and Recovery

Failure detection in the plant will be based on State Estimation theories. A discrete Kalman Filter as discussed below will be used to estimate the system outputs around nominal operating conditions [1].

The discrete Kalman filter for a component model may be given as:

$$\hat{X}_{k+1} = A\hat{X}_k + Bu_k + F(z_k - M\hat{X}_k)$$

where z_k , u_k , A , B , M , F , are known. For this model, the variables are defined as:

$$\begin{aligned} z_k &\triangleq \text{measurements} \\ \hat{X}_k &\triangleq \text{discrete state estimator} \\ z_k - M\hat{X}_k &= \text{white, zero mean,} \\ &\text{gaussian sequence (same covariance, as} \\ &\text{process noise) , } v_k \\ V_k &\triangleq \text{white, zero mean} \\ &\text{gaussian measurement sequence} \end{aligned}$$

The measurement residual ($z_k - M\hat{X}_k$) will be provided to “*Failure Detection*”. If the measurement residual indicates an error condition the residual vector will be passed along with the error condition to trigger “*Failure Isolation*”.

Failure Isolation identifies the component or components responsible for the fault symptoms. A model of the devices and components can be used for failure isolation. The device description includes structural, functional, and behavioral information. Methods of knowledge representation for failure isolation using models include causal networks, qualitative physical models, and belief networks [10]. Components responsible for the fault are identified by propagating symptoms through the model and testing the results.

The effects of component interactions should be described locally in terms of interactions of connected components, the locality principle [4]. The degree of detail of the model depends on the architectural fidelity desired. Sufficient detail must be included to detect and isolate faults which are recoverable.

Once the failure has been isolated its impact must be assessed. The assessment will determine whether the failure is critical to the operations of the spacecraft and whether any action needs to be taken. Precautionary measures will be built into the system to avoid nuisance trips.

If the performance requirements of the spacecraft cannot be met in the presence of the failure condition the controller will be redesigned accounting for the failed Actuator or Sensor. "Model Identification" algorithms based on the q-Markov Cover theory developed for ASTREX will be used to identify the new system model [2].

3.5.1 Output Variance Control

Output variance control techniques developed for ASTREX will be used to redesign the controller with the identified model [3]. The OVC algorithm seeks to design a controller to minimize the input energy subject to output variance inequality constraints. Consider the following time - invariant stabilizable and detectable linear system.

$$\begin{aligned}\dot{x} &= Ax + Bu + Dw \\ y &= Cx \\ z &= Mx + v\end{aligned}$$

The output variance control design problem determines the control gain G (and filter gain F) that minimizes

$$J = E_{\infty} u^T R u$$

Subject to the inequality constraints

$$E_{\infty} y_i^2(t) \leq \sigma_i^2, \quad i = 1, \dots, n_y$$

The controller redesign will be performed as an off-line process and the redesigned controller will be downloaded to the real-time computer.

4 Design Issues and Approaches

In this chapter, we identify some of the potential problems and design issues, followed by suggestions on design approaches that address them.

4.1 Constraints

Several constraints are important to ensure the FDIR system is effective. Constraints on on-orbit computer systems include the real-time processing and communication requirements demanded by a system. The resources available and response time required of the system will vary with the current objectives of the spacecraft mission. The resource requirements of mission execution will decrease the resources available to the FDIR system. Flexibility in resource utilization is needed maximize mission effectiveness. Just as components may misbehave or fail so may the sensors that measure component behavior or state. False indication of failure, or nuisance trips should be located quickly to prevent operation abortion and maintain the availability of useful components. False information should be discovered to prevent its propagation and prevent decay in mission effectiveness.

Making decisions in the face of uncertainty is inherent to the domain. Data from sensors may be unreliable. Degradation of components is to be expected. Performance requirements of mission phases restrict our ability to gain and analyze information from redundant sources. Faults may occur simultaneously. Decision making using uncertain knowledge has its own problems. Propagation of misinformation can have devastating results. Associating faults with a reliable component has two results. The reliable component may no longer be properly utilized. The unreliable component is undetected and may further degrade the mission.

Using knowledge from multiple sources helps to reduce uncertainty. Multiple sources may come from redundant sensors and analytical models. Adding redundancy may reduce uncertainty, but also increases the likelihood of failures.

4.2 Design Issues

Important problems which may be encountered in the domain of the On Orbit Supervisor for Controlling Spacecraft are summarized below.

1. **Unreliable Data.** Just as the equipment being monitored may fail, the monitoring equipment can also fail or be lost. This complicates the task of the FDIR system. To overcome this problem evidence from multiple sources utilizing probabilistic, fuzzy or inexact reasoning models can be used.
2. **Time - Varying Data.** Information about the system state changes as it executes its mission. This include information about position, mission phases, power levels, model of the system, etc.
3. **Single Line of Reasoning may be too weak.** Providing redundant paths of reasoning can provide a confirmation of our conclusions. There is a potential for added fault tolerance through multiple perspectives. By incorporating this knowledge into a search the strengths of separate models can be combined. Propagating constraints from one line of reasoning to another reduces uncertainty.
4. **Single Knowledge source too weak.** A malfunctioning system can fool us. By using knowledge from different sources we can gain new evidence for failure detection and failure isolation. A failure in one system may have predicted effects in another. Consulting other systems can remove uncertainty and resolve conflicts.
5. **Unreliable Knowledge.** Diagnostic procedures may profitably make assumptions about the reliability of equipment and the number or nature of faults. In the diagnosis process these beliefs may be revised. Knowledge of the plant and how component are configured and perform will vary as the system experiences faults and degrades.
6. **Interaction of subproblems. (Least commitment principle)** The system does not make arbitrary or premature decisions until there is enough information. Least commitment principle coordinates decision making based on the information available from sub-problems (possible method

to minimize nuisance trips, however sometimes the system must commit itself, either due to lack of data or due to the mission mode.) Subproblem interaction is a result of using multiple lines of reasoning and knowledge sources.

7. Modes of the mission will limit the resources available for use and the priorities of the FDIR systems. During different phases of the mission, goals such as safety, reliability, and effectiveness will take on different sets of priorities.

4.2.1 Interaction between experts

The manner of cooperation has been the subject of many debates during this project. There are several factors which must be balanced to maximize the effectiveness. These factors are not independent and often conflict with one another. Below is a brief description of each.

Sharing information benefits the system by providing additional redundant sources of data, data for deriving redundant information, and data which may give clues about the "real" behavior of the plant. The last class of information helps restrict the set of possible behavior that the plant is exhibiting. No one system has a complete view of the plant. Not every component can be monitored and monitoring equipment can fail. By sharing information a more complete and accurate assessment of the system state can be maintained.

Shared information can also be detrimental to the system when the information is false or inaccurate. Propagation of false information throughout the system can have devastating effects. Some information may need to be hidden from other systems.

There are several options for sharing information. A global database with all the data and knowledge of the experts accessible to all has no hidden information. Sharing can be restricted by placing partial results in the global database instead of all information derived in intermediate steps. Instead of a global database, local databases may be used. Local databases may be nonintersecting or intersecting. Intersection may be total, i.e. duplicate global databases, or partial. Duplicating frequently used information can be more efficient than requesting information each time it is needed. However

duplicated information may become outdated in a real-time system. Nonintersecting databases generate the overhead of requesting information every time it is needed. The cost of nonintersecting databases may be prohibitive with a real-time expert system.

Communication bandwidth is limited. Sharing all knowledge in a global database may be expensive. Excessive consumption of communication bandwidth may not allow other objectives of the mission to be performed. Communication should be minimized, but remain effective to FDIR. The demand on communication bandwidth may vary with the phases of the mission. During times of heavy communication traffic, the experts must conserve communication bandwidth. During these phases decisions may have to be made with less certainty than other phases.

Processing bandwidth is limited. By distributing and modularizing knowledge, efficiency is gained by limiting the context of execution of the expert. This does not mean CPU cycles are free. Again demand on resources will depend on the phase of the mission and on the reliability of the plant. *Certainty of decision making may be lowered in these times.* This may be accomplished by lowering certainty levels or using methods which are faster, but are not as accurate. A record of events which occur and can not be investigated thoroughly can be recorded. When sufficient time is available further evaluation of reported anomalies can be examined and diagnostics can be scheduled.

4.2.2 Subsystem autonomy

Different experts have different perspective on the spacecraft. Experts supervise different subsystems and the supervisor looks over them all. Each does not have complete knowledge of the entire system by itself. Allowing redundancy of results by separate experts, parallel fault tolerant capabilities can be added. Inconsistent redundant results could be resolved with a voting mechanism or by analysis at a higher level of abstraction. Impossible or inconsistent results could be disregarded.

Giving the subsystem supervisors little autonomy makes them too dependent on the system supervisor. If the system supervisor becomes overloaded real-time response may be lost as well as accuracy. The system supervisor may experience failures in the hardware and software running the systems

and failures in communication hardware and software. Making a subsystem supervisor very autonomous decisions may be made prematurely, without using information from other subsystems. A completely autonomous subsystem supervisor makes judgements without reducing the uncertainty of its local knowledge. Each of the supervisors should not accept information from other supervisors as completely accurate.

4.3 Design Philosophy

Our design is motivated primarily by the need to combine multiple sources of information, the need to attain real-time performance and the need to effectively manage the complexity of spacecraft.

The hierarchical approach allows the distribution of processing capabilities onto multiple processors. Additional advantages have previously been outlined. The hierarchy defines the lines of authority and organizational responsibilities.

Further decomposition results in the construction of smaller efficient knowledge base modules. Our decomposition is along the functional/organizational lines of the spacecraft design. The approach is object centered [5],[11]. An additional benefit of this approach is that information is abstracted before being presented to upper layers. Data is abstracted by performing an analysis of data. The results of the analysis are an abstract representation of the raw data. Communication traffic is reduced by transmission of abstracted results.

Higher level layers incorporate more responsibility and have a wider range of vision over system functions. Abstraction limits the amount of information any one layer has to manage and control. Bounding the amount of information and decision making capabilities in any one layer is associated with intelligent capabilities [8]. Each layer has limited knowledge of the capabilities of its superior and peers, but does know its subordinate's capabilities.

Information exchange is permitted among peers to broaden their view of the system. Each node may know of redundant sources of information. Direct communication with those nodes is permitted. The hierarchy is mostly for authoritative lines and should not become a hindrance to performance.

Knowledge of where partial results are is also important. Partial results can be used to derive more results or constrain conclusions. Each peer must be capable of performing its own task without the aid of others. Sharing information is meant to enhance capabilities of other nodes.

5 Demonstration System

5.1 Purpose

To exercise our conceptual framework developed and to help illustrate the On-orbit Supervisor for Controlling Spacecraft Systems, we implemented some simple scenarios that are prototypical of the On-orbit supervisors tasks. The demonstration is to illustrate the roles of the supervisors, their cooperation during operation, and the communication among them. Since the blackboard facilities were not built into any of the expert system shells that we evaluated, a blackboard emulator was developed to further our evaluation of our architecture. The demonstration system has showed the opportunistic architecture of the blackboard system suitable for the problem.

5.2 Overview

There are three supervisors in the demonstration system, the system supervisor, attitude control supervisor, and the power supervisor. The supervisors of the demonstration system are depicted in Figure 5. It is the system supervisor's responsibility is to plan the mission. The demonstration system has a mission task to maintain a new attitude. The demonstration system begins with the system supervisor planning to seek a new attitude. The system supervisor must determine the parameters for this task. In this case a preference for the type of actuator used is determined. Additional parameter information might include acceleration and deceleration profiles. The "maintain new attitude" command is transmitted to the attitude control supervisor for execution. Two possible execution sequences are shown in Figures 6 and 7.

The attitude control supervisor carries out the command. To do so a control law must be determined and actuators and sensors selected to meet the requirements of the parameters of the command. In this process it may consult with the power supervisor to obtain the current power output of the solar panels and the current battery level. During execution of the maneuver the attitude control supervisor monitors execution to detect errors. It may also consult with the power supervisor to determine when conditions are suitable

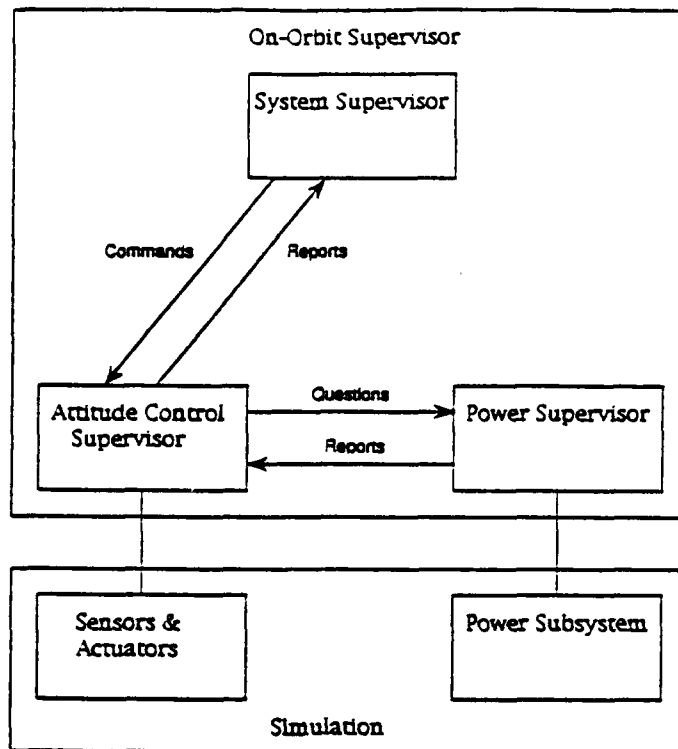


Figure 5: Demonstration System

to switch actuators. If an error is detected during execution the attitude control supervisor must isolate the fault and perform recovery and reconfiguration. If the attitude is lost or is uncertain due to a fault, the attitude control supervisor will consult the power supervisor to determine the power level. It will then attempt to use this information to deduce an approximate attitude. This is an example of information sharing for intelligent FDIR.

The demonstration system has 8 different KSs, 11 triggers, and over 30 rules. The architecture of the On-orbit supervisor was easy to use for constructing the demonstration system. The separation of "when" to apply knowledge and "how" to perform a task provided by the triggers and KSs, respectively, of the blackboard architecture simplifies the problem of controlling execution in the supervisors. This style easily permits using different KSs depending on the current context. KSs used in non-critical mission phases will not be triggered in critical mission phases. However, the drawback of the triggers is the overhead that they incur if the blackboard control becomes overly complicated. A simple control does not hinder performance while permitting the flexibility and adaptability of the blackboard to be used to efficiently solve problems.

Scenario - Monitor Execution to Use Torque Motor when Power Sufficient

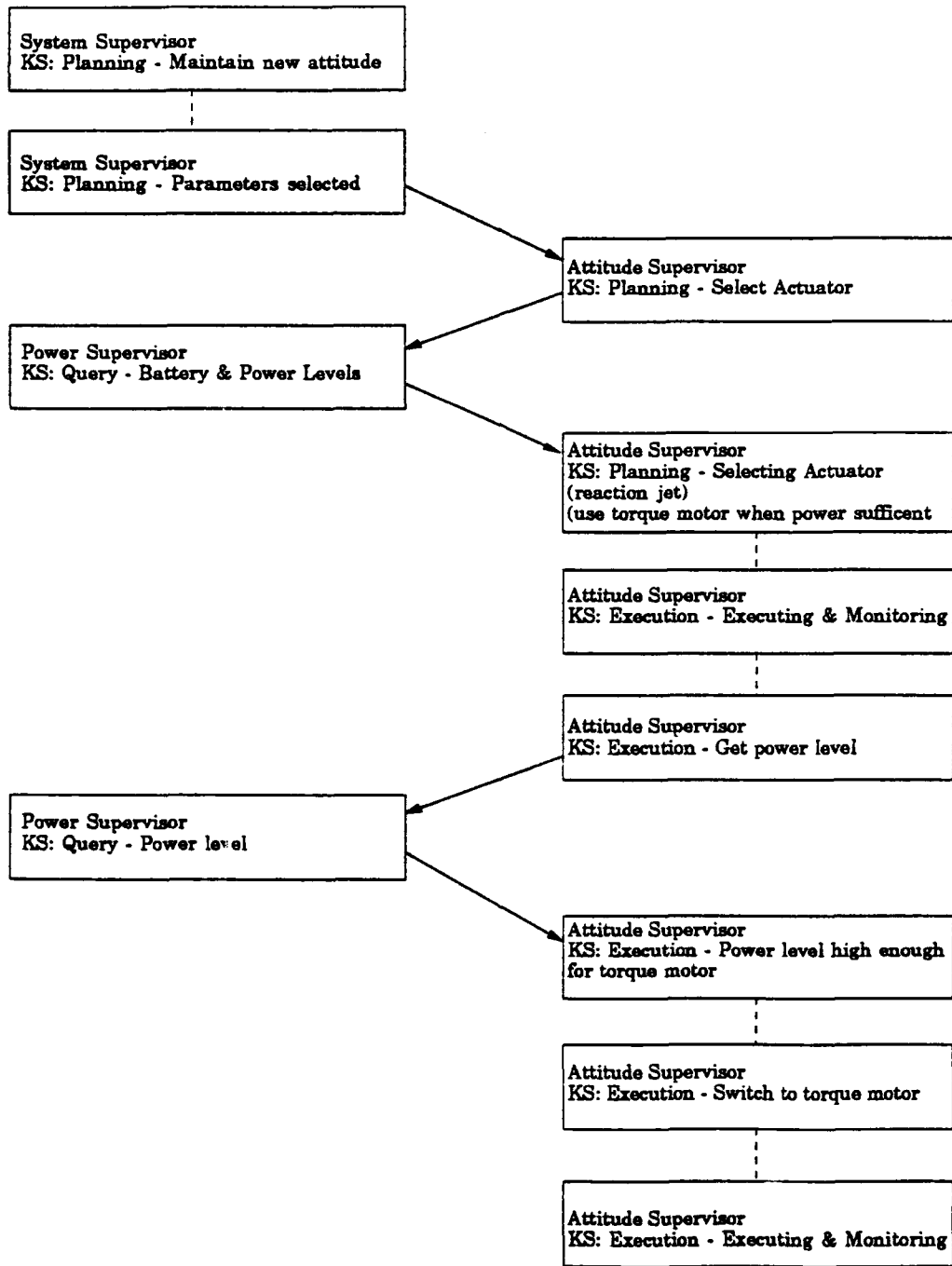


Figure 6: Scenario 1

Scenario - Failure Detection, Isolation & Recovery

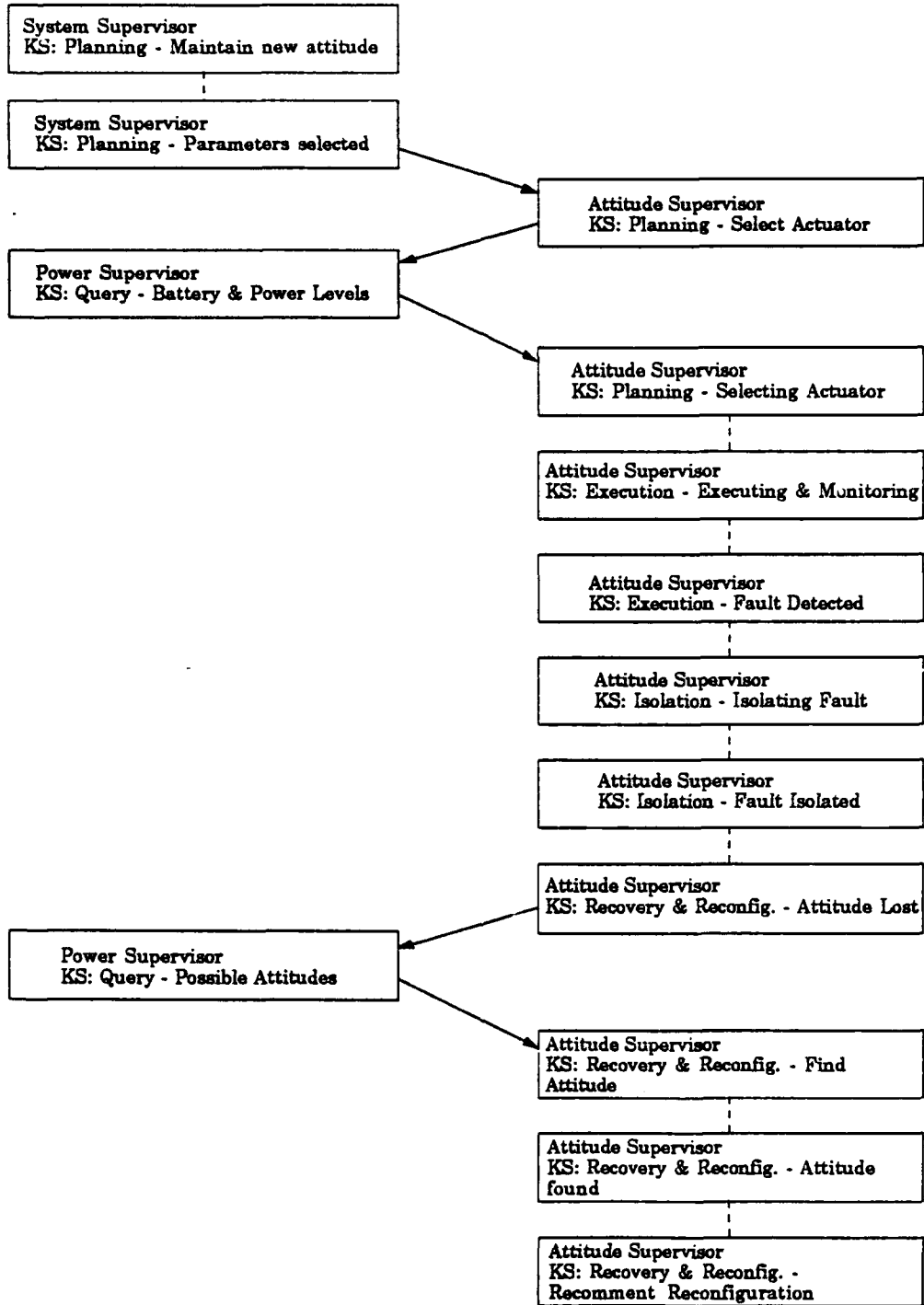


Figure 7: Scenario 2

References

- [1] Sage, Andrew P. and Melsa, James L., *Estimation Theory with Applications to Communications and Control*, McGraw Hill Inc., 1971
- [2] Ramakrishnan J., Hu A., VanderVoort R., Berg J., Cossey D.F., *Identification Experiments on ASTREX*, AIAA Guidance Navigation and Control Conference Proceedings, Vol. 2, August 1991.
- [3] Byun K.W., Ramakrishnan J., Skelton R.E., Cossey D.F., *Covariance Control of ASTREX*, AIAA Guidance Navigation and Control Conference Proceedings, Vol. 2, August 1991
- [4] DeKleer, Johan and Brown, John Seely, *A Qualitative Physics Based on Confluences*, Artificial Intelligence, Vol. 24, 1984
- [5] Hayes-Roth F., Waterman D. A., Lenat D. B., *Building Expert Systems*, Addison Wesley
- [6] Lesser, Victor R. and Corkill, David, *Functionally Accurate, Cooperative Distributed Systems*, IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-11, No. 1, January 1981.
- [7] Erman L.D, Hayes-Roth F., Lesser V.R., Reddy D.R., *The Hearsay-II Speech-Understanding System: Integrating Knowledge to Resolve Uncertainty*, Computing Surveys, Vol. 12, No. 2, June 1980
- [8] Fox, Mark S., *An Organizational View of Distributed Systems*, IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-11, No. 1, Jan. 1981
- [9] Hayes-Roth, Barbara, *A Blackboard Architecture for Control*, Artificial Intelligence, Vol. 26, 1985
- [10] Wellman, Michael P., *Fundamental Concepts of Qualitative Probabilistic Networks*, Artificial Intelligence, Vol. 44, Addison Wesley, Reading, MA, 1990
- [11] Barr A, Cohen P.R., Feigenbaum E.A., *The Handbook of Artificial Intelligence Volume IV*, Addison Wesley, Reading, MA, 1989

- [12] Bond Alan H. and Gasser Les, eds., *Readings in Distributed Artificial Intelligence*, Morgan Kaufman Publishers, Inc., Los Altos, CA, 1988
- [13] Huhns M, ed., *Distributed Artificial Intelligence*, Morgan Kaufman Publishers, Inc., Los Altos, CA, 1987
- [14] Dynacs Engineering Co., *ASTREX Actuator Thruster Force Determination for Pure Slew*, Final Report, LARC Contract NAS1-19158, October 1991.