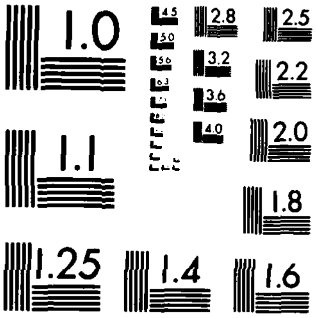
AD-A152 078 ARTIFICIAL INTELLIGENCE IN SPACE PLATFORMS(U) AIR FORCE 1/2
INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF
ENGINEERING M A WRIGHT DEC 84 AFIT/GSO/OS/84D-9
UNCLASSIFIED F/G 6/4 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

①

AFIT/GSO/OS/84D-9

ARTIFICIAL INTELLIGENCE IN SPACE PLATFORMS

THESIS
Michael A. Wright
Captain, USAF
AFIT/GSO/OS/84D-9

DTIC
ELECTE
APR 4 1985
S D
B

Approved for public release; distribution unlimited

Preface

The purpose of this study was to analyze the feasibility of implementing Artificial Intelligence techniques to increase autonomy for orbiting spacecraft. AI advances in the area of expert systems offer practical applications in many fields. The development of an expert system to provide routine housekeeping tasks on a satellite could reduce the dependence on vulnerable ground control centers.

A cost analysis was done to determine the value of an autonomous spacecraft expert system in comparison to the current ground system. Initial figures indicate that expert systems can be cost effective and increase satellite autonomy. More work needs to be done to reduce the size and weight of current AI machines and develop them for space application, but the expert system design process should begin soon.

During the research for, and writing of, this thesis I have had significant help along the way that deserves recognition. I would like to thank my faculty advisor, Lt Col Mekaru, for his critical eye and guiding hand. I would also like to express appreciation to Capt Steve Cross for technical advice and to Capt Ed Gjermundsen and the Air Force Space Technology Center for encouragement. Finally, I wish to thank my wife and typist, Joy, for her understanding and assistance throughout this entire project.

Michael A. Wright

Table of Contents

List of Figures

AFIT/GSO/OS/84D-9

Abstract

This ~~study~~ thesis determined the feasibility of implementing Artificial Intelligence techniques on orbiting spacecraft. The main thrust was to evaluate the current technology of expert systems (computer programs) and determine their value to satellite tasking. The goal for an expert system to be effective was that it must be able to perform spacecraft stationkeeping without ground assistance.

Analysis began by outlining the basic functions of the DSCS III and noting deficiencies as measured against an autonomy scale. Many of the defeciencies could be corrected with conventional computer programming, but stationkeeping required AI techniques for proper execution. Expert systems were then examined and studied for applicability to the primary task of orbit maintenance. R1, an expert system designed to perform computer configuration, was found to be a good baseline for comparison and further development. The process of orbit maintenance, as currently done by human experts, was explained and outlined for expert system design. Finally, a cost analysis provided information which supported further development of AI technology for spacecraft implementation.

ARTIFICIAL INTELLIGENCE for SATELLITE PLATFORMS

Section I

Introduction

> Many human mental activities such as writing computer programs, doing mathematics, engaging in commonsense reasoning, and understanding language are said to demand "intelligence." There are computer systems that can diagnose diseases, plan the synthesis of complex organic chemical compounds, solve differential equations in symbolic form, analyze electronic circuits, understand limited amounts of human speech and natural language text, or write small computer programs to meet formal specifications. We might say that such systems possess some degree of "artificial intelligence" (19:1).

Progress in the field of Artificial Intelligence (AI) is making it possible to realize practical applications in almost every field. A facet of the Artificial Intelligence research and development is the implementation of "expert systems," or smart computers (6:4). Professional people have found expert systems to be of tremendous potential value in daily operations. By performing as a knowledge base capable of analyzing complicated problems and providing the required solution, expert systems can augment or replace engineering specialists.

As a scientific discipline, Artificial Intelligence is not clearly defined. In fact, it is interpreted quite differently by people with varying backgrounds. However, for this study, analysis will be limited to that aspect of Artificial Intelligence involving expert systems. It might be helpful at this point to define AI and expert systems and a conventional computer.

A conventional computer is a machine that performs rapid, often complex calculations using stored instructions and information. A digital computer, the most common type, uses numbers in a binary format to perform logical and numerical calculations. In contrast, an analog computer uses voltages instead of numbers and manipulates numerical representations of data. Electronic computers can process data at extremely high rates of speed. Even the slowest of current computers operate at speeds in excess of 1/100th of a second. That means that a typical computer can process a transaction in less time than a blink of the eyes. Computers can store information in memory banks which are capable of holding thousands of pages of data. This enables the user to program computers to solve difficult and time-consuming problems using simple commands which trigger more complex procedures within the computer. Most problems take advantage of the algorithmic/number crunching capabilities. The basic process is to input programs and procedures which solve a typical problem, let the computer manipulate the data, and the result will be an organized, correlated output.

Artificial Intelligence is a concept which is derived from the idea of making computers behave in ways that mimic intelligent human behavior. In many cases problems do not lend themselves to strictly algorithmic solutions. Scientists and researchers who pursue AI are attempting to build a computer system that can learn or understand from experience, much the same as humans are said to do. An artifically intelligent machine has to be able to respond quickly and successfully to new situations. Its knowledge base is compiled of extensive data which has been input by experts in a particular field of science. AI is not an attempt to replicate the human race. It is unrealistic to expect computers to feel emotion and respond in all situations, but it certainly might be possible to build a machine that can solve problems faster and better than man and one which is capable of expanding its data base with increased experience in the problem solving realm (6:9). Artificial intelligence has begun the process by building "expert systems."

An expert system is a computer program capable of maintaining a vast knowledge base about a particular subject area, such as medicine or mineral exploration, and can use sophisticated problem solving techniques to get results beyond the reach of conventionally programmed computers. The importance of AI stems from the fact that these machines can be used by man to reduce the reliance on human experts, who are expensive and in short supply (4:177). Immortal machines serving as "clones" of human experts could perpetuate and spread expertise throughout an organization. An AI system is different from a conventional computer in its new approach to computer programming. Instead of being programmed to follow a step by step procedure, an expert system is programmed to follow a few general procedures of problem solving. Facts, models, rules of thumb, and other general knowledge about solving a particular type of problem are stored in the computer memory (12:32). Problem solution comes about as the computer uses facts about the problem and its knowledge base and general solution methods to find and apply a specific solution.

Early attempts at problem solution were fraught with frustration by proponents of AI. During the 1960's, such great minds as McCulloch, Turing, and von Neumann began to form the basis of what would evolve into Artificial Intelligence. The initial efforts were designed to employ a "few laws of reasoning coupled with powerful computers [to] produce expert and superhuman performance" (12:7). The realization that the detailed process of most intelligent human activities was unknown marked the beginning of AI knowledge base development. In 1977, a paper was presented by Feigenbaum to the International Joint Conference on Artificial Intelligence which provided insight to the growing field and was summarized as: "The power of an expert system derives from the knowledge it possesses, not from the the particular formalisms and inference schemes it employs" (12:6). The field has since progressed from an "all knowing, all seeing" direction to one of a more restrictive use of AI techniques in specific expert systems. These expert systems are limited in application and loaded with detailed knowledge of a specific subject to better apply inference schemes and provide practical use to human users.

One of the most widely known expert systems is called PROSPECTOR. PROSPECTOR is a computer-based consultation system designed to assist geologists working on specific mineral deposit problems. The "task of PROSPECTOR is probabilistic interpretation of soil and geological deposit data" (12:54). It is an expert system designed to work in a specialized domain. Its success has been quite promising when evaluated against expert geologist's decisions (4:161). PROSPECTOR predicted a molybdenum ore location in Washington State which was confirmed by drilling and a find worth $100 million.

Another expert system, MYCIN, is making progress in the medical field. Medicine is a fertile field for expert systems for two reasons. The first is the obvious benefits to society gained by a reliable and thorough diagnosis. Expert systems consider all possibilities where it may not be realistic to expect a doctor to be able to take the time to look at all possible ailments. The second reason is the solid knowledge base established in medicine is easily transferrable to a computer. The medical taxonomy is clear and experts in the field are identifiable (4:177). MYCIN is used to diagnose and offer therapy in certain cases of infectious blood diseases. Although it is not currently used in clinical work, it is being used as a medical teaching aid. It has provided excellent results with "performance comparable to experts in the field" (12:53). Expert systems are used in education, science, and medicine. Their ability to act as "experts" or "expert consultants" continues to increase.

AI Application to Space

Can Artificial Intelligence expert systems be designed to perform tasks for a space platform? One of the most complicated spacecraft tasks is orbit maintenance. The process of orbit maintenance or stationkeeping is a task that currently requires vast material resources and dedicated space specialists (2: slide 20). As our satellite systems become more complex and the space environment more congested, the job of maintaining proper orbits become critical (9:2). If an expert system could be placed in orbit as an integral part of the satellite system, it could greatly reduce the burden or routine "housekeeping" chores if not eliminate it.

The Air Force Space Technology Center is vitally interested in applying Artificial Intelligence to autonomous space systems. Increased spacecraft autonomy would enable the satellite to operate independently of ground systems for extended periods of time. Presently, satellites must normally be frequently contacted by ground stations for orbit correction and fault analysis. Since the ground control centers are the least survivable link in the space network (20:46) autonomous military spacecraft would enhance space systems survivability. Endurability or extended useful lifetimes result from the spacecraft maintaining more optimal orbits and can be increased by:

> 1) Reducing spacecraft dependence on ground stations, thereby enhancing the capability for continued payload mission accouplishment if ground stations are disabled.
>
> 2) Achieving an autonomous satellite health and ephemeris maintenance capability by Fiscal Year 1987, with spacecraft launched after this date capable of performing internal housekeeping activities for unattended periods of time on the order of six months.

Statement of Problem

The basic problem is to determine the feasibility of putting an expert system on a satellite. The expert system would be expected to perform all satellite services with emphasis on orbit maintenance, or stationkeeping. Stationkeeping is totally "ground controlled" at the present and would have to be assumed by the expert system (22:vol I,22). "Ground controlled" refers to the process of determining satellite orbits with tracking data from ground stations, determining spacecraft orbit errors and corrective action by teams of orbital analysts, and sending corrective action to the spacecraft by ground control centers. To be effective, expert system must be able to monitor, diagnose, and control a satellite in orbit around the earth. The expert system must be able to calculate the current position of the satellite using on-board sensors and processors and determine anomalies in the orbit. Diagnosis will include the ability to determine what type of correction is needed to retain proper orbit and the optimal time to implement correction action. The controlling function must be able to fire thrusters accurately and efficiently correct the orbit.

Resource and integrity functions are also tasks which could be directed by an on-board expert system. Resource and integrity functions, or health and welfare maintenance, include monitoring and directing power, maintaining proper attitude, monitoring thermal controls, and directing propulsion systems (22:vol III,55). An expert system would monitor all subsystems and update data bases to maintain proper orientation and operation levels.

Research Questions

Can an expert system be used effectively and efficiently on satellites? If so, the burden current ground systems experience would be reduced significantly. Current satellite systems are heavily dependent upon the ground control centers and absorb a great deal of time and money for routine "housekeeping" chores. Can on-board expert systems increase satellite autonomy and overall effectiveness? Independent satellites would increase endurability by reducing the vulnerability of space systems to command, control, and communications failure. Independent spacecraft would also reduce the need for extensive ground base systems and specialized teams of analysts. It is not likely that space borne Artificial Intelligence can eliminate all ground materiel and personnel resources, but it should be able to improve upon the current situation.

Factors which will be compared and analyzed in order to determine feasibility are cost, weight, and overall effectiveness. Cost will be analyzed in terms of actual expert system development (hardware and software) costs vs.

the cost of current equipment and personnel used to perform the various tasks which could be assumed by the AI component. Overall effectiveness will be measured by analogies to existing systems in terms of reliability, autonomy, deterrence, survivability, and trade offs in satellite orbit stability. Reliability in this case refers to the system's ability to make accurate and timely decisions.

Scope

The overall objective of this study will be to determine if it is possible to put an expert system capable of performing some of the typical housekeeping tasks on a satellite. Determination will depend on many factors, including expert systems capabilities, ability to match computer and satellite systems, and resolve by decision makers to allow a computer system to make critical decisions autonomously. This study will be limited to implementation of expert systems to perform satellite housekeeping tasks. Housekeeping tasks are defined as the routine tasks required to maintain a healthy and stable satellite platform capable of performing its operational mission. Eventual use for AI on space platforms may include battle management and command, control, and communication authority, but at this stage of AI development those do not appear to be realistic topics of discussion (24:41).

Section II

Tasking in Space

For Artificial Intelligence to be a feasible option for spacecraft design, it will have to accomplish many tasks. Some are partially done by on-board processors and require periodic ground support. Other tasks are currently ground dependent annd will require new computer capability. This section will describe the various tasks as they relate to the Defense Satellite Communication System (DSCS). The DSCS was chosen because it is a typical space platform with various missions and unclassified data is readily available for comparison.

DSCS III

The Defense Satellite Communication System is part of the wideband communications network. Key users of the wideband system are: Defense Communication Agency, Diplomatic Telecommunications Service, World Wide Military Command and Control System (WWMCCS), and the White House Communication Agency. The DSCS system began with the initial series launched in 1966-68. The DSCS was a solid step toward effective global communications. The follow-on system, DSCS II, had a substantial increase in transmitter power and transponder channels. The DSCS II system used existing technology and incorporated new capabilities for the military user. Rapid repositioning for various users was added with the DSCS II system. Securing command and telemetry modes became a reality. The DSCS II also boasted "modest" nuclear hardening. The addition of steerable antennas was a major step toward effective communications paths. Finally, in 1981, DSCS II was upgraded with the addition of more efficient digital communication subsystems and reliable ground terminals.

The DSCS II satellites were launched in pairs and operate in synchronous near-equatorial orbits. Four of the six satellites are active at any one time, leaving two for spares or contingency situations. They are equipped with two earth coverage horn antennas and two steerable spot-beam dish antennas. They also use an onmidirectional bicone horn antenna for command and control. The steerable antennas can concentrate their signals on small areas of the earth to link portable ground stations into the communications system. The DSCS II satellites operate in the "super-high frequency" range of 7-8 GHz.

The current space communications network is effective during normal circumstances. The six DSCS II satellites form the "wideband" system segment. At the time these systems became operational the users had a critical need for a working system. Therefore, time and cost restraints tempered the advancement of satellite communications. Current systems lack survivability! The ultra-high frequency range is quite vulnerable to jamming and disruption by nuclear events.

The DSCS III is beginning to replace the DSCS II. The first DSCS III was launched in October, 1981. The DSCS III in orbit is still being evaluated and studied to determine operational capabilities and future funding levels to complete the network. The DSCS III, as designed by General Electric Space Systems Division, will have a lifetime of ten years. It will use six channels and have the flexibility to support large and small terminal users. DSCS III has a small steerable dish antenna which can be fine tuned for users with small receivers. It also provides its own jam resistance through the use of four earth coverable horn antennas. The four horn antennas provide redundancy and flexible switching networks. DSCS III will continue to provide the Air Force's strategic link with nuclear forces via separate transmit and receive antennas.

The most notable feature of the DSCS III is its innovative phased array antenna system which provides additional security and flexibility. It has a 61 element multiple beam antenna (MBA) used extensively for receiving and two, 19 element MBAs for transmission. The MBA system can selectively null jamming signals with a minimum disturbance to friendly users. This process maximizes flexibility, in both the uplink and downlink. Of the six available channels, the receiver MBA can be connected to any or all of channels 1-4. At the same time the transmit MBAs can connect to the same four to develop whatever combination is required. Another advance for the DSCS III is the gimballed dish antenna. It provides the additional gain and anti-jam capability which is required for high priority users such as the White House Communication Agency, Ground Mobile Forces, and JCS contingencies. The lack of control security experienced by DSCS II has been corrected with an advanced, electronic telemetry, tracking, and command subsystem.

Autonomy Assessment of DSCS III

The DSCS III was assessed using "Goals for Autonomous Spacecraft," a Jet Propulsion Laborotory (JPL) internal document (Appendix A). The levels of autonomy, as defined in the JPL document range from 0 to 10. An overall look at the DSCS III revealed the following (22:Vol I, 9):

> (1) The existing DSCS III functions are at levels of autonomy ranging from 0 to 5. The average level appears to be about 2 or 3. This means that there is a high level of dependence on ground operations for analysis, planning, and decision making. The power and thermal control functions have many hard-wired, autonomous functions, and attitude control has considerable

autonomy implemented in both software and hardware. However, spacecraft resource management and health/welfare maintenance are almost entirely ground directed. Stationkeeping is completely directed by the ground.

(2) A primary goal for autonomy is for the spacecraft to operate for 60 days with nominal performance and for 6 months with acceptable performance, without ground intervention. A spacecaft autonomy level of about 5 is required to meet this goal. A level 5 spacecraft (see Appendix A) is capable of executing a prespecified program of events and is also autonomously fault tolerant.

(3) The autonomous DSCS III assessment philosophy assumes that the requirement for 6-month performance without ground intervention arises from a high-level-of-conflict situation. It has been assumed that under other conditions the ground will be able to periodically update the initial orbital state from which the spacecraft will have to operate independently. If this assumption is not valid, the spacecraft autonomy level may have to be increased beyond 5 to somehow provide its own initial state.

(4) On-board redundancy management is required for a high probability of meeting the 60-day/6 month requirement, particularly if hostile threats to the spacecraft are considered.

(5) Autonomous stationkeeping is also required, even for 60 days performance , since east-west stationkeeping maneuvers are required more frequently to meet the +.1 degree stationkeeping requirements. The maneuvers could occasionally occur as frequently as every few days (depending upon station location and sun-moon perturbation phasing).

The DSCS III currently in operation has some autonomy built in. Its power, thermal control, attitude control, and telecommunication service functions are quite capable of meeting the six month autonomy goal. Most of the integrity maintenance (health and welfare) can be done on board, but the "analyses and direction of redundancy management are done by the ground" (22: vol I, 10). One of the major obstacles for autonomous operation on the DSCS III is maintaining the correct orbit position, or ephemeris maintenance. According to the JPL DSCS Assessment, the "spacecraft cannot be made free of ground intervention for even 60 days."

The DSCS III was broken down into three functional areas to analyze autonomy; 1) services, 2) resources, 3) integrity. Each of these areas will be analyzed in detail later, but Figure 2-1 shows just how much of each area is currently autonomous and to what level of capability. Figure 2-2 further defines the components of each functional area for the DSCS III.

Satellite Tasks

There are three functional areas within the DSCS III design. They are spacecraft services, resources, and integrity. Spacecraft services include activities to maintain a stable platform, operate at satisfactory power and thermal considerations, and allow necessary communications for telemetry and command (22: vol I, 17). The most ground dependent task is stationkeeping. Stationkeeping is the process of maintaining the satellite in its particular orbit within a fine degree of accuracy. Orbit anomalies affect mission effectiveness and can

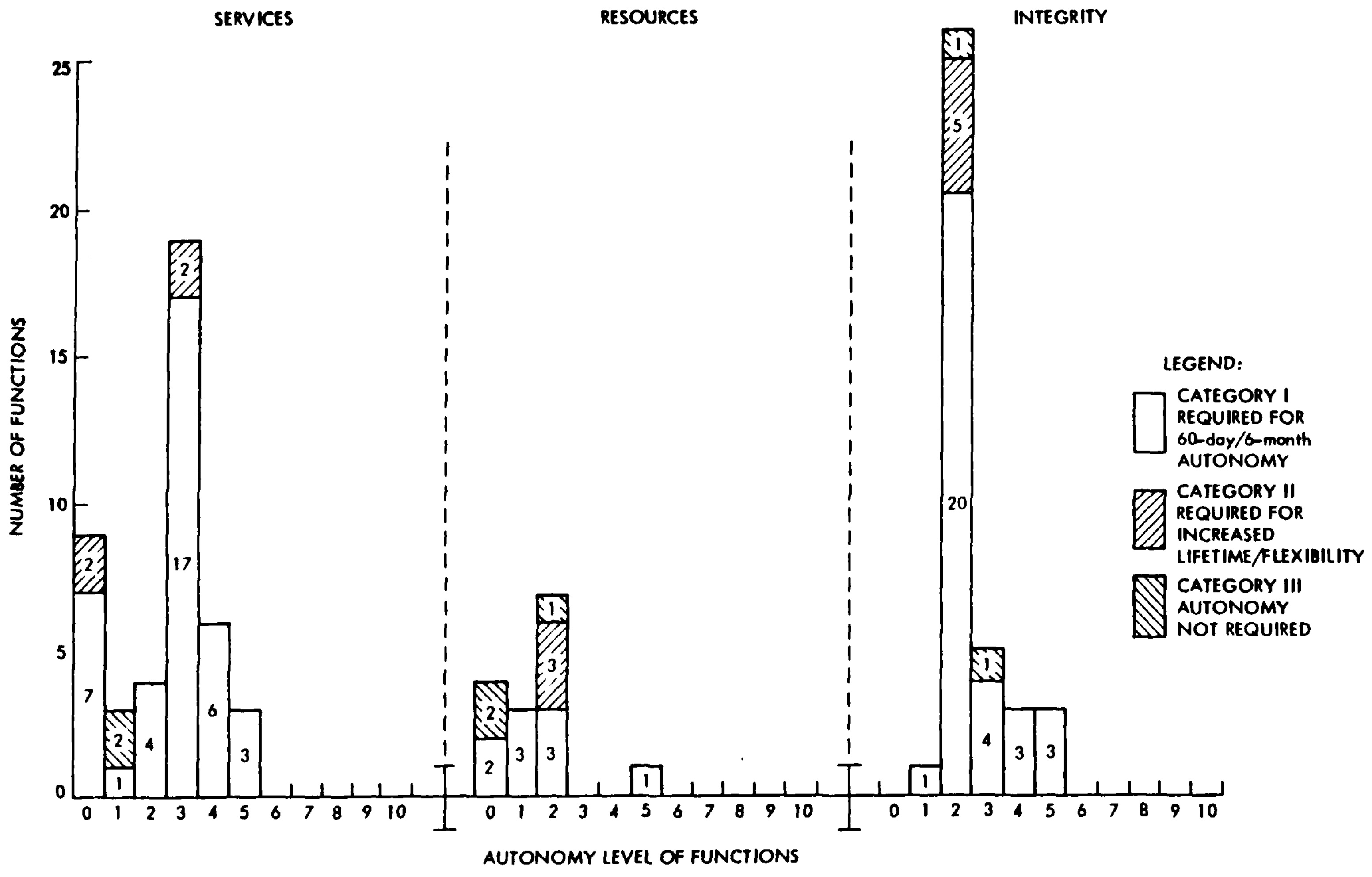


Figure 2-1 Summary of DSCS III Functions' Autonomy Levels and Need for Autonomy

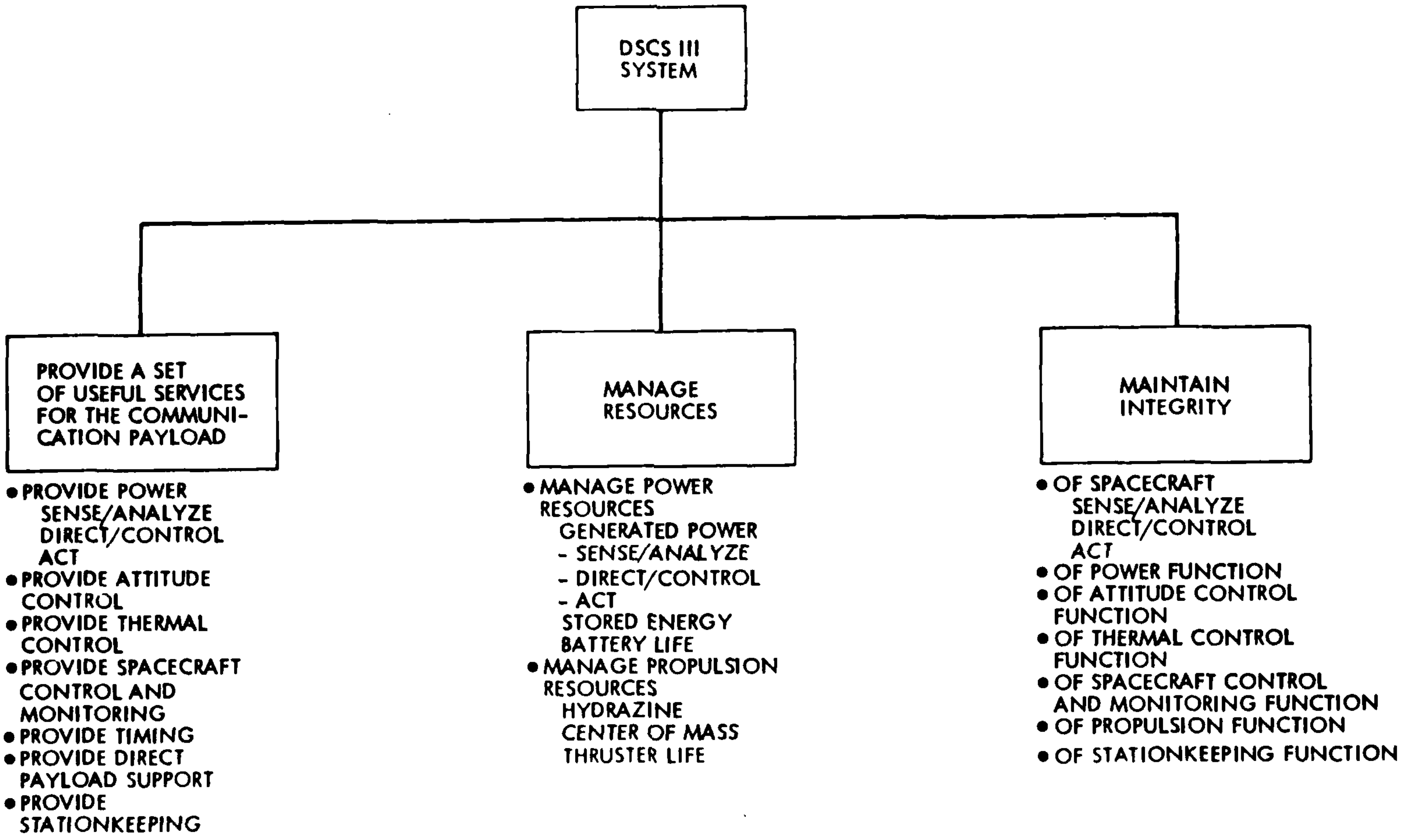


Figure 2-2 DSCS III Functional Hierarchy

decrease the satellite lifetime. The functions of this task include location measurement using on board sensors, maneuver planning, thruster selection, thruster firing, and many related tasks. Stationkeeping is a major portion of the spacecraft services function.

The second functional area for the DSCS III is managing spacecraft resources. The resources are the limited expendables which must be properly managed for the spacecraft to survive and perform (14: vol I, 20). There are two such resources on the DSCS III; power and propulsion. Some of the power related tasks include managing the generated energy, stored energy, battery life, and solar array attitude. Propulsion resources require management of hydrazine, center of mass (cm), and thruster life (22: vol I, 50).

The third functional area of the DSCS III which requires autonomous operation is integrity. Integrity refers to health and welfare and to the protection of the spacecraft from failures (22: vol I, 20). Integrity maintenance is currently a ground intensive activity. As with any complicated electronic equipment, redundancy is an important issue. Some of the requirements within the integrity section which are necessary to make the DSCS III autonomous are (22: vol III, 58):

1) Acquiring pertinent health status of all spacecraft subsystems.

2) Analyzing health information for fault occurrences.

3) Isolating fault sources and generating fault correcting commands.

4) Output fault correction and verify execution of commands.

5) Store current health information, faults observed and corrections taken for ground control records.

Integrity maintenance functions would have to be performed throughout the spacecraft and involves all of the major subsystems.

FUNCTIONAL AREA ONE - Spacecraft Services (Figure 2-3)

There are three activities within the services area which require increased capability to improve autonomous operation. They are power, attitude control, and stationkeeping.

POWER. Two areas of the power function which require increased autonomy are solar array orientation and power distribution. The solar array orientation must be able to account for mission phases, celestial events and develop a timeline for "issuance of solar array position and rate commands to stepper motors" (22: vol III, 23). The distribution of power is currently assessed at a level 2 and must reach level 5 for autonomous operation. Power

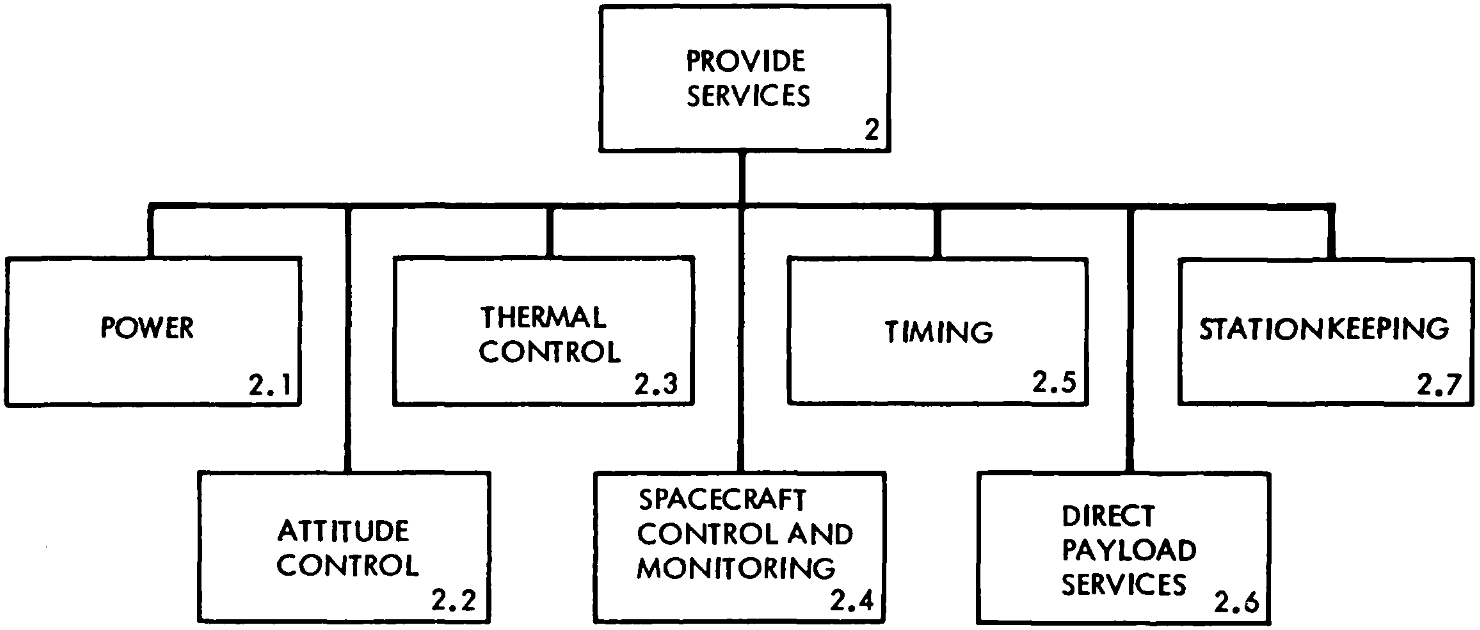


Figure 2-3 DSCS III Services Functional Hierarchy

distribution would need to be driven by various energy algorithms to indicate load requirements and consider alternate load configurations. Addtional capabilities needed in the power function include (22: vol III, 25):

1) Load prioritization table.
2) Load power for each operation mode.
3) Timing function.
4) Processing capability.

ATTITUDE CONTROL. The DSCS III provides a stable platform and most of its attitude control functions are autonomous. Some additional autonomy is required to establish post-launch earth acquisition, reference re-acquisition, and thruster selection. The satellite needs to be able to verify proper completion of sun acquisition after launch and then perform analysis of the parameters to acquire earth. Reaction wheels would then be given preset commands to establish proper configuation for operation of the spacecraft (22: vol III, 26). Re-acquisition could be automated in the same way as earth acquisition by verifying sun position. As the reaction wheels begin to unload and the spacecraft experiences some east/west drift, thrusters must be fired to maintain proper orbit. Autonomous navigation systems are required to independently select and fire thrusters. This function will be discussed in the stationkeeping section.

STATIONKEEPING. Since stationkeeping is currently totally ground controlled, all sensing and control functions have to be added to the spacecraft. Figure 2-4 shows the hierarchy of the stationkeeping function. The DSCS III does not currently have enough capability to independently sense orbital position. Navigation sensors and interfaces must be added to provide accurate data for orbit determination (22:vol III, 37). The computer processing capability would be necessary to direct, analyze, and control the satellite orbit.

The addition of an autonomous navigation system is costly and complicated, but must be done if any spacecraft is to be truly independent of ground systems. It is likely that the navigation package would include sensors and computer interfaces. The navigation subsystem could be a simple implementation of supplying the effective time, magnitude, direction, and type of velocity maneuver to other subsystems. The most complex navigation system would maintain propulsion system status, calculate optimal maneuver times, select and fire thrusters, and supply an integrated command sequence (22: vol III, 43).

FUNCTIONAL AREA TWO - Spacecraft Resources (Figure 2-5)

Within the resources area, there are two basic activities --managing power and managing propulsion. Each

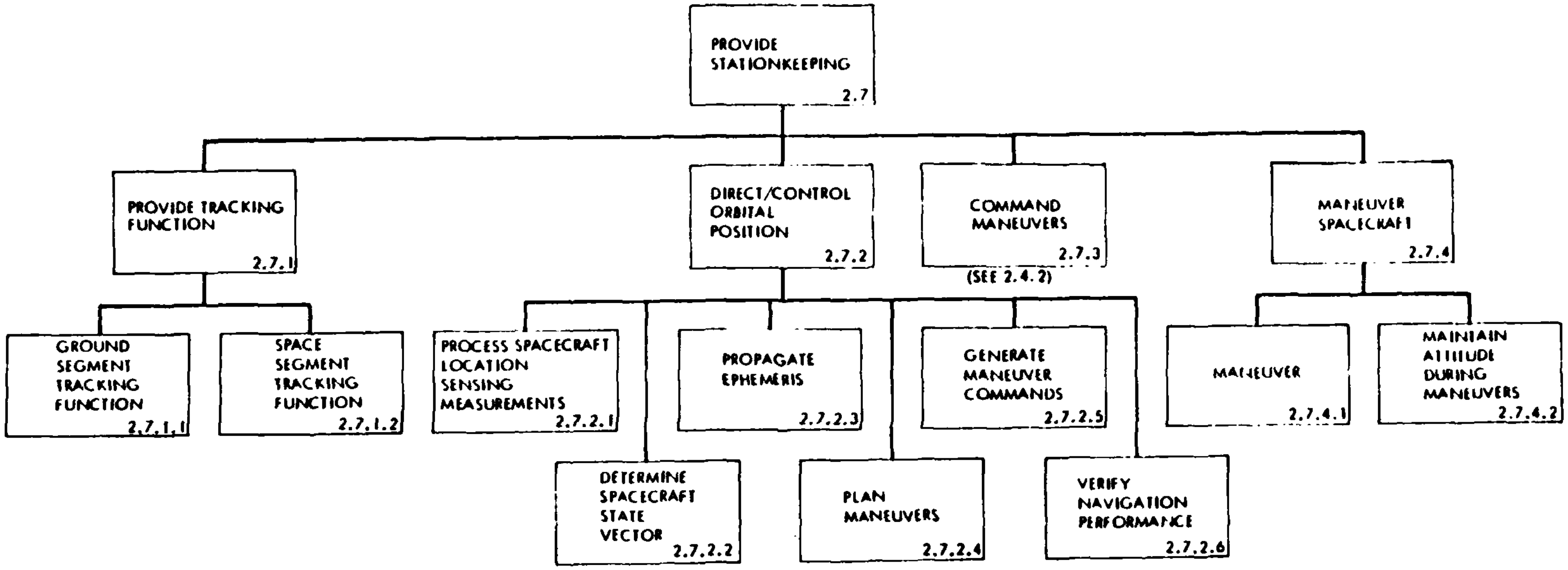


Figure 2-4 Stationkeeping Service Functional Hierarchy

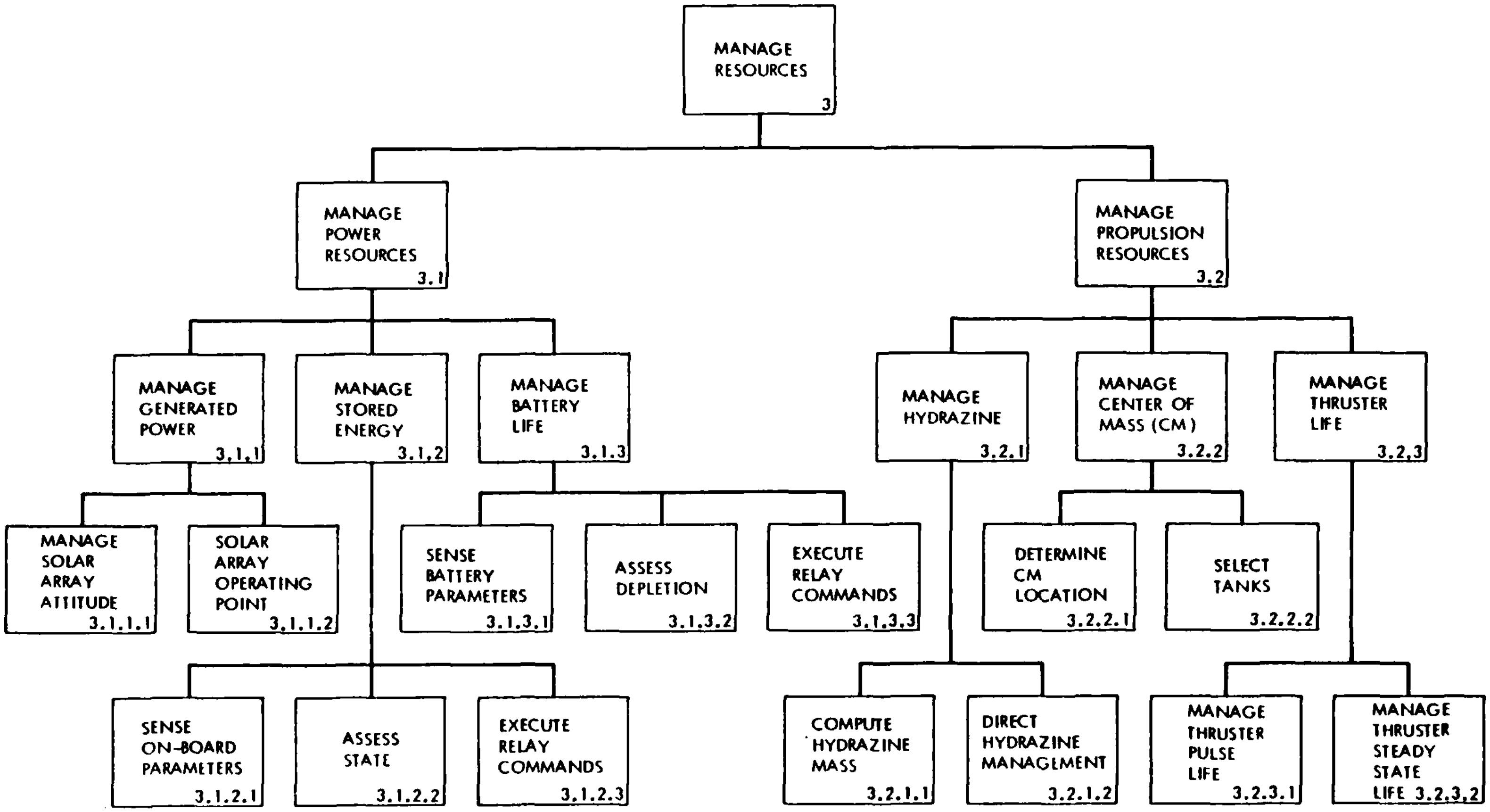


Figure 2-5 Resource Management Functional Hierarchy

of the activities includes monitoring and maintaining subsystems. On the DSCS III, management of power relies heavily on the stability of battery systems. Propulsion must be monitored and used to correct attitude and orbit.

MANAGE POWER. Most of the management for generating power and solar array maintenance is already autonomous. To increase autonomy, stored energy and battery life management require sensing functions to determine battery parameters and some on board capability for battery state assessment (22:vol III, 49). The measured parameters and a computer model could then determine the state-of-charge. It is necessary to predict the state-of-charge trend during eclipses using load profiles and solar array output. It may be necessary to make adjustments to the charge rate to properly maintain power storage.

Battery life management is not fully autonomous, but can be accomplished using the same procedures as described above for stored energy. Properly maintaining stored energy will increase battery life by closely monitoring all facets of battery operation. Battery depletion assessment does need some additional capability to ensure battery health. The standard maximum depth-of-discharge has been established as 80 percent and should not be exceeded to ensure battery life (22: vol III, 51). Some additional capabilities needed in this area are:

1) Flexible battery charge models.

2) Charge history.

3) Load profiles.

4) Power supply predictions.

5) Relay capability.

PROPULSION. This section of resource management is an integral part of the autonomous navigation and stationkeeping functions. On board analysis is required for each of the tasks to be properly integrated. Computation of propellant mass must be done and verified based on previous usage. Priority tables must be used to trade off required stationkeeping maneuvers and remaining fuel (22: vol III, 53). The thrusters must be maintained to ensure proper pulse and health. Thruster management would include taking pulse counts and monitoring pulse degradation over time.

FUNCTIONAL AREA THREE - Spacecraft Integrity (Figure 2-6)

Integrity maintenance involves the entire spacecraft. While all subsystems need redundancy, how are corrections made and backup systems called into action? The Jet Propulsion Laboratory has recommended the addition of a Redundancy Management Subsystem (RMS) "capable of providing fault detection and correction functions for the entire DSCS III spacecraft bus" (22:vol III, 55).

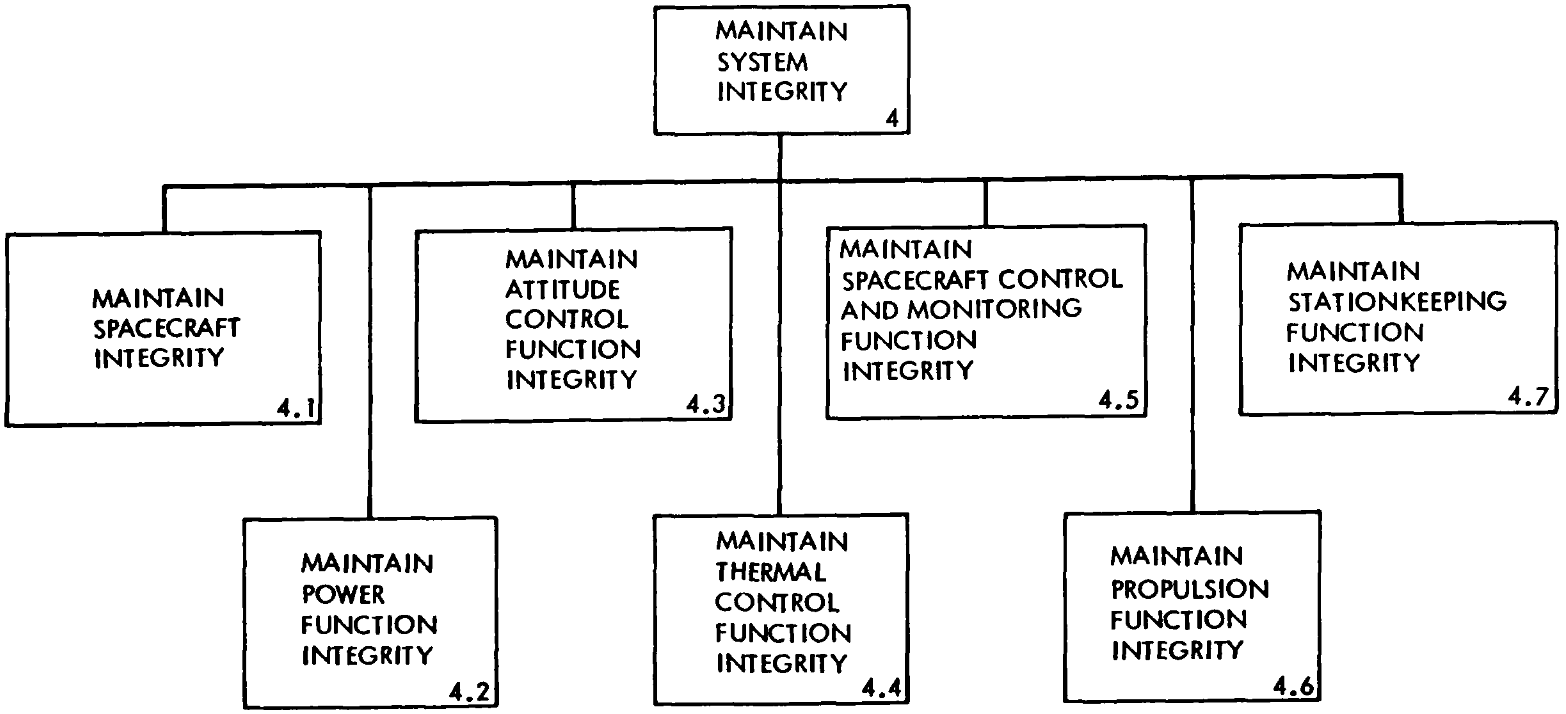


Figure 2-6 Integrity Maintenance Functional Hierarchy

The RMS would interact with each of the major functions through computer interfaces capable of detecting faults and relaying them for correction. Autonomous operations of the entire DSCS III would require a great deal of storage capability for data. Stored information would include spacecraft status, fault history, diagnostics, and software programs.

Using this structure of a RMS linked to the subsystems via Distributed Processing Units (DPUs) (See Figure 2-7) the functional requirements can be defined as follows (22:vol III, 71):

> (1) A DPU shall acquire health information from its host subsystem by monitoring selected subsystem sensor signals via dedicated lines.
>
> (2) A DPU shall store software subroutines required to analyze functional performance and determine needs unique to its host subsystem.
>
> (3) A DPU shall execute selected internally-stored software subroutines only upon receipt of commands from the RMS.
>
> (4) A DPU shall provide processed, subsystem-unique health information to the RMS upon request by the RMS.
>
> (5) The RMS shall analyze acquired subsystem health information by detecting fault occurrences, isolating fault sources, and defining the required commands to be issued for fault correction.
>
> (6) The RMS shall generate subsystem fault correction commands.
>
> (7) The RMS shall output validated fault correction commands.

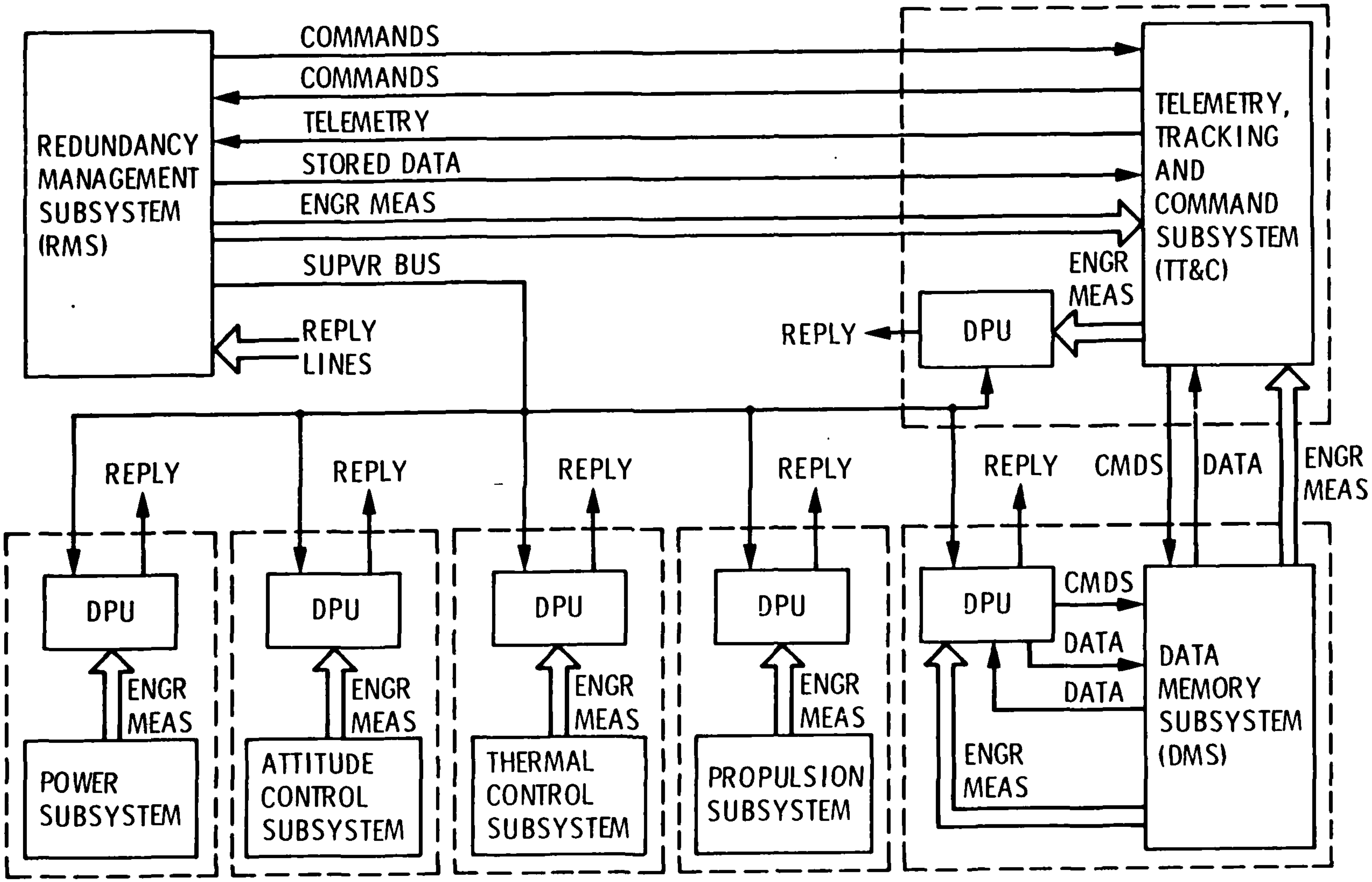


Figure 2-7 Spacecraft Integrity - Redundancy Management System

(8) The RMS shall verify proper execution of fault correction commands.

(9) The RMS shall store pertinent spacecraft diagnostic information.

(10) The RMS shall be capable of loading the memories of all DPUs.

(11) The RMS shall be inherently fault tolerant so that any internal single-point failure will not degrade its performance.

As one might expect, satellite operations are complicated and the tasks are arduous. It is not sufficient to plug in autonomous subsystems, they must be integrated into the entire spacecraft design. As the push for autonomous satellites continues, all of the tasks mentioned above and other related functions must be taken into account. Within the spacecraft services area, stationkeeping certainly provides the greatest challenge at this point. The resources and integrity functional areas require specific component modifications and integration. The tasks are defined quite well. Can Artificial Intelligence techniques be used to provide operational programs to accomplish spacecraft tasking and increase satellite autonomy? This question will be evaluated in the following sections.

Section III

Artificial Intelligence Capabilities

This section will examine the capabilities of state-of-the-art expert systems in an effort to determine spacecraft tasking feasibility. The architecture of expert systems will be described and further explained. Three critical components of the architecture are the knowledge base, inference engine, and data base. The process of knowledge engineering, or filling the knowledge base, will also be presented. Having established the structure, the emphasis will shift to current tasks expert system are capable of doing and brief explanations of some of the systems. Then, some guidelines will explain the complex process for constructing an expert system and illustrate the transition from research and development to practical application.

Expert Systems

As this section focuses on expert systems, it may be helpful to expand on the definition of an expert system. One of the leaders in early expert system development, Feigenbaum, wrote that:

> An "expert system" is an intelligent

> computer program that uses knowledge and inference procedures to solve problems that are difficult enough to require significant human expertise for their solution. The knowledge necessary to perform at such a level, plus the inference procedures used, can be thought of as a model of the expertise of the best practitioners of the field.
>
> The knowledge of an expert system consists of facts and heuristics. The "facts" constitute a body of information that is widely shared, publicly available, and generally agreed upon by experts in a field. The "heuristics" are mostly private, little-discussed rules of good judgment (rules of plausible reasoning, rules of good guessing) that characterize expert-level decision making in the field. The performance level of an expert system is primarily a function of the size and quality of the knowledge base that it possesses.

For an expert system to fill the bill as an expert, it must perform at the "expert" level. The word expert must be considered seriously. There are significant characteristics of an expert which can be identified and measured. Quality of performance is a major concern, but high quality is not enough. Ability to reach decisions quickly is a valuable trait only if the decision is a good one. Certainly, speed and quality must be balanced to produce the desired result. Most experts tend to be experts only in a specialized domain. Specialization dictates a trade-off in depth and breadth of knowledge. It is not really feasible to be an expert in many technical fields. An expert system is therefore allowed by necessity to work in a narrowly defined area.

Architecture

Using Feigenbaum's description to construct a working definition, an expert system can be broken down into fairly easily understood components. An expert system is made up of three basic components: 1) Knowledge base, 2) Inference engine, and 3) Data base (see Figure 3-1). The knowledge base contains the specialized domain facts and heuristics, or rules of thumb, associated with the particular area. The inference engine, or control structure, is the component which contains the search strategy for problem solution. The data base is a working memory which keeps track of input and output data and problem status.

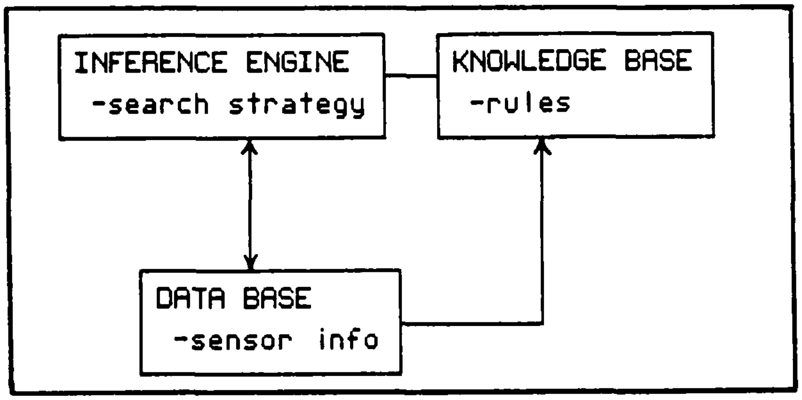


Figure 3-1. Expert System Components

For example, a very simple expert system could be designed to monitor and control room temperature (18:44). The inference engine would employ a forward search technique. The knowledge base could have the following four rules:

```
IF TEMP > 70 AND TEMP < 72 THEN STOP
IF TEMP < 32 THEN CALL REPAIR MAN, TURN ON HEATER
IF TEMP < 70 AND FURNACE = OFF THEN TURN ON
IF TEMP > 72 AND FURNACE = ON THEN TURN OFF
```

The data base would contain the current temperature as indicated by a sensing device. This example is an obvious simplification. It would not be cost effective to build an expert system to control a thermostat, but it serves to illustrate the three components.

The following architectual principles were presented by Randall Davis (6:6) which serve to mold the components into a cohesive unit.

Architectural Principles

1) Separate the inference engine and knowledge base.

2) Use as uniform a representation as possible.

3) Keep the inference engine simple.

4) Exploit redundancy in knowledge base.

By separating the inference engine and knowledge base, the knowledge is more accessible, easier to identify, and can be more explicit. Uniformity reduces the number of mechanisms required for translation and keeps the design simpler and more transparent. Simple control structures in the inference engine aids process explanation and should provide more comprehensive rational feedback. Since feedback, or explanation, is generated by replaying the actions of the system, simple actions generate better feedback. Finally, redundancy in the knowledge may overcome inexact knowledge by combining bits of information from varying sources to yield answers.

COMPONENT ONE - Knowledge Base

The knowledge base is, by far, the most critical component of most expert systems. This is the foundation for ultimate problem solution. The knowledge may be represented by presenting it as "IF-THEN" rules. If a system is considered to be rule-based, the "knowledge base is made up mostly of rules which are invoked by pattern matching with features of the task environment as they currently appear in the global data base" (10:6).

The production rule is a two-part construct with the first part representing some pattern and the second part specifying some action to be taken when data, from the data base, matches that pattern. The pattern may be made up of several clauses linked by the logical operators AND and OR. The pattern may also be procedures that operate on data in the data base to produce values for further rule comparison. The second part of the rule consists of verb phrases that specify the action to be taken. A typical example might be "IF the satellite east position is greater than 61 degrees OR the west position is less than 59 degrees THEN the satellite requires orbital correction." Assuming that 60 degrees is the optimal east-west position at a given time of orbital determination, the assertion "the satellite requires orbital correction" will be added to the knowledge base if either precondition is met.

Rules in a knowledge base represent the domain facts, beliefs and heuristics. Facts are bits of knowledge that are known to be true and their validity is unquestioned. Knowledge entered as opinion are beliefs, usually knowledge that is accepted as being valid. Heuristics are bits of information learned through experience or rules of thumb developed by experts that aid in the area of filling in incomplete knowledge. Rules combine to form the most powerful aspect of expert systems, the knowledge base. Rule based systems, as explained by Duda (8:242):

> contain hundreds of rules, usually obtained by interviewing experts for weeks or months...In any system, the rules become connected to each other [by association linkages] to form rule networks. Once assembled, such networks can represent a substantial body of knowledge...

During the early development of AI, emphasis was on search techniques. Intelligence was thought to be largely a domain independent effort, hence knowledge free. The game of chess was examined in detail to illustrate the various search techniques. Research indicated that human chess masters used the process of mental storage of large catalogs of pattern-based rules to play the game. It was said that human experts in the game could organize and utilize up to 50,000 rules to achieve their remarkable performance. Those rules are so powerful that only 30 rules are needed in an expert system to adequately cover the roughly 2,000,000 configurations for a subdomain problem of King and Knight against King and Rook (10:7). It was noted that chess is a fairly certain game with a well established knowledge domain, but as the domain becomes more complicated the rules may expand exponentially.

Within this rule-based network, if the knowledge is not well established some rules may have certainty factors (CF) attached to them. The CF is a numerical valve which indicates the degree of certainty associated with that rule. Human experts use judgement and empirical rules which sometimes cannot be fully supported by available data. In an expert system rules based on heuristics can be given CFs and then, when the certainty factors are combined with other problem data, the solution will have a certainty value attached to it.

Still within the context of the knowledge base is the topic of "knowledge engineering." Knowledge engineering has grown up with the field of AI. Knowledge for an expert system can be acquired in many ways. All of the ways involve transferring expertise needed for high performance problem solving in a particular domain from a source. In most cases, the source is a human expert, but the source could include case studies as does PUFF, an expert system designed to diagnose lung disease, built entirely from 100 case studies. Other sources include empirical data or documentation used to train the human expert who is being emulated. From the emphasis on knowledge, the expert systems credo has developed. The basic commandments were given by Feigenbaum in 1977 and expanded by Davis in 1982 as (6:6):

1) In the knowledge lies the power
2) The knowledge is often inexact, incomplete.
3) The knowledge is often ill-specified
4) Amateurs become expert incrementally.
5) Expert systems need to be flexible.
6) Expert systems need to be transparent.

The first statement about knowledge and power suggests that extensive stores of knowledge about the task and not domain-independent methods lead to successful problem solution. Most areas investigated for use by AI techniques do not have completely specified laws or theories and the knowledge tends to be inexact, incomplete, and informal. Ill specified knowledge is a common problem. It refers to the process of trying to establish for the experts what knowledge they have and what is required for problem solution. As mentioned previously, knowledge acquisition is an incremental process and therefore the transition from amateur to expert must follow a similar path.

Further emphasizing the need for separation of components are the concepts of flexibility and transparency. The system must be flexible enough to change easily because most of their system lifetime will include changes, updates, and improvements. Transparency allows the engineers to follow progress and maintain control during the changes by being able to distinguish changes in the three components separately. If the three components were not distinct, then changes to one section might inadvertently cause confusion in another section of the program.

The knowledge engineer quickly becomes an integral part of expert system development. The early attempts at knowledge transfer dictated that the computer programmer "transform the expert's knowledge into code without separating the knowledge from the reasoning mechanism." (12:130). This process requires that the programmer be an expert in the domain, or quickly become one.

Current procedures still allow a computer programmer to assist the expert in transferring knowledge to the system, but it can also be done by direct transfer of expert to machine via intelligent editing programs (see Figure 3-2). It now becomes possible to separate domain knowledge from the rest of the program and this enhances flexiblity and transparency. It is hopeful that a similar process can be developed for data through an elaborate induction system and for textbooks using a program capable of reading text and transferring information to the knowledge base. The latter two ideas are still just that, ideas, but they may assist the knowledge engineer in the future.

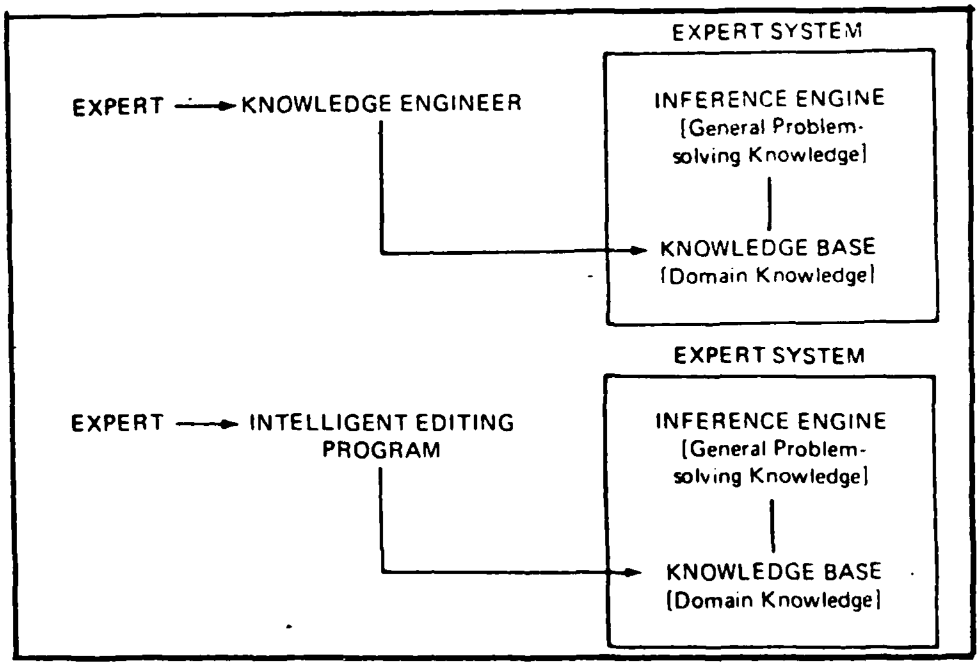


Figure 3-2. Knowledge Engineering (7:130)

Whatever the source, the process is basically the same -- transfer of domain knowledge for use in problem solving. The knowledge engineer must proceed through several stages during the construction of a knowledge base. These stages have been identified as problem identification, conceptualization, formalization, implementation, and testing as shown in Figure 3-3 (12:140). Although this is a well-defined figure, the actual process is anything but well-defined. The process will vary for situations and people, but the basic pattern will remain the same. After the knowledge transfer is complete, the knowledge base must be matched to an appropriate inference engine.

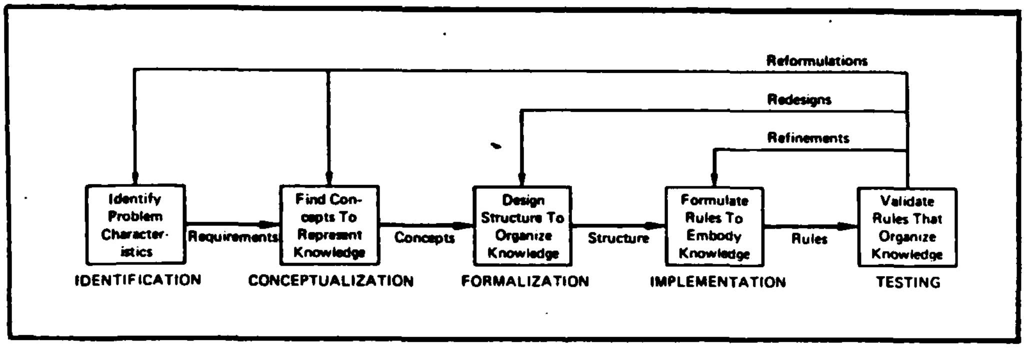


Figure 3-3. Stages of Knowledge Acquisition (7:140)

COMPONENT TWO - Inference Engine

Within the inference engine or control structure, there are basically two methods of search strategy employed: forward chaining and backward chaining. They can be used separately or in combination to produce the necessary results in an expert system. Both strategies rely heavily upon heuristics to work effectively. After the strategy is determined and implemented, the inference engine then becomes the controlling force for the solution strategy in the knowledge base.

In forward chaining, the system attempts to reason from given data to reach a logical goal. Facts about the problem must be given and then a forward search of the rules in the knowledge base may "fire" one or more of the rules. At this point the control strategy must determine which rule(s) to apply. As the rules continue to fire, a solution will be developed. In this case the chaining starts from a set of conditions and moves toward some conclusion. The speed and accuracy of this process depends on many factors--number of rules, time to search, computational difficulty, but it is heavily dependent upon the use of heuristics. Forward chaining is a good strategy to employ for monitoring functions.

Backward chaining is a goal-driven strategy. This type of strategy works backward from a hypothetical solution (goal) to find evidence which supports the solution (see Figure 3-4). This process might progress through many intermediate testings of hypotheses, or subgoals, to get to the end result. A system using backward chaining would search the knowledge base for a rule, that when fired, would give the desired result. The system "attempts to match the first part of the rule against the initial problem description stored in the working memory. If the first part of the rule matches the hypothesis, the search is finished" (15:34). If the match fails, the search continues, now trying to match the first part of the rule which fired but did not fully satisfy the hypothesis. If the match is not complete the system may request information from the user or it may guess at the desired goal. This strategy applies well to the processes of diagnosis or classification.

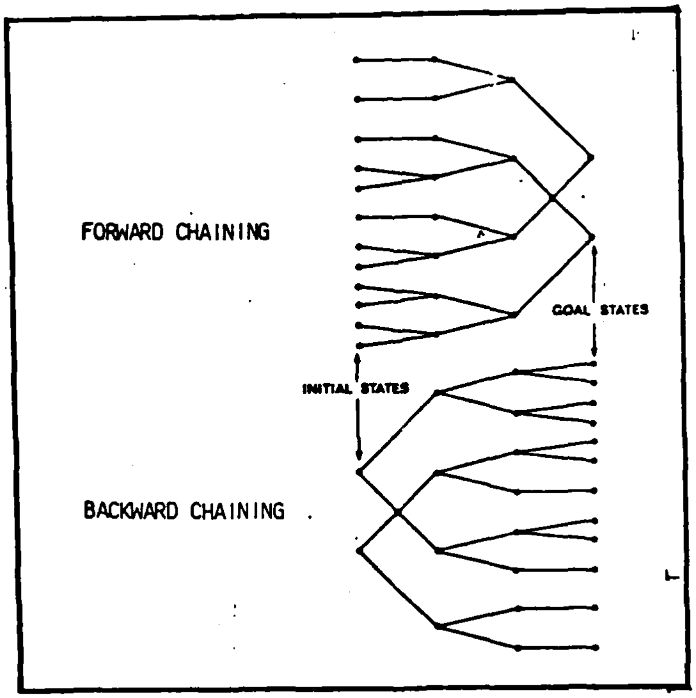


Figure 3-4. Inference Engine Search Strategies

Heuristics, educated guessing or rules of thumb, are a major player in expert systems, just as it is for human experts. There are always limits on the amount of time and data storage available to spend on search problems. As the problem domain becomes large so does the search space and blind search may not be realistic. Although search of every node is possible, it quickly becomes unreasonable. For that reason, and the fact that human experts use guessing when knowledge is incomplete, heuristic search is employed in expert systems. Using heuristic search techniques in conjunction with forward or backward chaining allows the system to cease searching when a satisfactory solution is found. There may or may not be an optimal solution, but the search time can be reduced significantly by allowing for a satisfactory answer.

If a simple mathematical equation could characterize the problem, then there would be no need for guessing. But "in many real problems, well-behaved functions are elusive. Sometimes a strategic retreat is necessary; that is, one must seem to move away from a goal (overriding some evaluation function) in order to achieve it. For example, to enter a room it is worth detouring to an unlocked door even though a locked door is closer - if there is no key" (12:69). Guessing is necessary when there is incomplete knowledge and it is not posible to determine a "best choice" to proceed. If a solution search space is dense and all solutions may be equally desirable, then guessing can be efficient (12:110). Complete knowledge is not possible, but a well designed knowledge base coupled with a properly arranged inference engine can produce excellent results.

COMPONENT THREE - Data Base

The simplest component, the data base, stores facts about the state of the world and provides a working memory space. Fact and data needed to manipulate the various rules in the knowledge base would be stored in the data base. It can also be used to store historical type facts about the solution process. Users may access the data base for problem status or to ensure the data being entered is correct.

Expert System Tasks

There are a number of ways to classify expert systems. One way emphasizes the function of the task. The various functions include interpretation, diagnosis, monitoring, prediction, planning, design, and control (12:83). Another way is to characterize the systems by problem domains, such as science, medicine, computer configuration, trouble-shooting, oil and mineral exploration, military, and computer-aided instruction. Each of these methods of classification is limited due to overlapping of functional areas and unique problem domains respectively. Dr. William Gevarter (10:12) suggested "a more fruitful approach appears to be to look at problem complexity and problem structure and deduce what data and control structure might be appropriate to handle these factors." Appendix B describes some of the more developed expert systems and outlines the basic approach and key elements of the knowledge base, data base, and the control structure.

There is obviously more data in Appendix B than necessary to explain the basic workings of expert systems. Some of the different types of control structures were not explained in the previous section, but this listing clearly shows the many uses for expert systems. Table 3-1 summarizes the characteristics of the systems presented in Appendix B. These are all systems that are currently working in some capacity. Some are strictly research and development, some are for academic training and experimentation, and some are being used in the corporate world to save money and increase efficiency.

Within the science and medicine domain; DENDRAL is used by industrial and academic researchers to identify chemical compounds; MOLGEN is used by leading genetic engineers to synthesize DNA molecules; MYCIN diagnoses and recommends treatment for infectious blood diseases for doctors in clinics associated with Stanford. One of the

Table 3-1

Characteristics of Systems in Appendix B

			Control Structure															
			Search Direction				Control							Search Space Transformations				
SYSTEM	FUNCTION	DOMAIN	Forward	Backward	Forward and Backward	Event Driven	Exhaustive Search	Generate and Test	Guessing	Relevent Backtracking	Least Committment	Multilines of Reasoning	Network Editor	Multiple Models	Break into Sub-Problems	Hierarchical Refinement	Hierarchical Resolution	Meta Rules
MYCIN	Diagnosis	Medicine		x			x											
DENDRAL	Data Interpr.	Chemistry	x					x										
EL	Analysis	Elec. Circuits	x						x	x								
GUIDON	C.A.I.	Medicine				x												
KAS	Knowl. Acquis.	Geology	x										x					
META-DENDRAL	Learning	Chemistry	x					x										
AM	Concept Formation	Math	x					x										
VM	Monitoring	Medicine				x	x											
GA1	Data Interpr.	Chemistry	x					x										
R1	Design	Computers	x				x								x			
ABSTRIPS	Planning	Robots		x												x		
NOAH	Planning	Robots		x							x				x			
MOLGEN	Design	Genetics			x				x	x	x				x	x		x
SYN	Design	Elec. Circuits	x											x				
HERARSAY II	Signal Interpr.	Speech Unders.			x						x	x					x	
HARPY	"	"	x															
CRYSALIS	Data Interpr.	Crystallography				x		x										x

expert systems used extensively in the corporate world is R1. Digital Equipment Corporation uses R1 to configure computer systems and it has reportedly saved the company millions of dollars in labor costs (15:37). The growth of expert systems in every field is phenomenal. Research engineers and big business are progressing in the field of AI to build systems with practical applications.

Expert System Construction

This section will not get into the details of hardware and software development, but merely establish some of the prerequisites and guidelines for successful expert systems. One of the prerequisites include (10:34) at least one human expert acknowledged to perform the task well and willing to impart his knowledge to a machine. The expert's knowledge should have primarily been obtained through experience and judgement. The expert must be able to "explain the special knowledge and experience and the methods used to apply them to particular problems. Finally, the task should have a well structured domain and high payoff.

The stages of development for an expert system have been outlined by Randall Davis as (6:10):

1. System design.
2. System development.
3. Formal evaluation of performance.

4. Formal evaluation of acceptance.
5. Extended use in prototype environment.
6. Development of maintenance plans.
7. System release.

Getting to Stage 3 is the difficult part. The time for construction of an expert system has been reduced from 20-50 man years for early systems to 5-10 man years (10:35). The construction will usually consist of only 2-5 people and tends to be rather time-intensive. Figure 3-5 graphically shows the time path construction for typical systems has taken during the last 20 years.

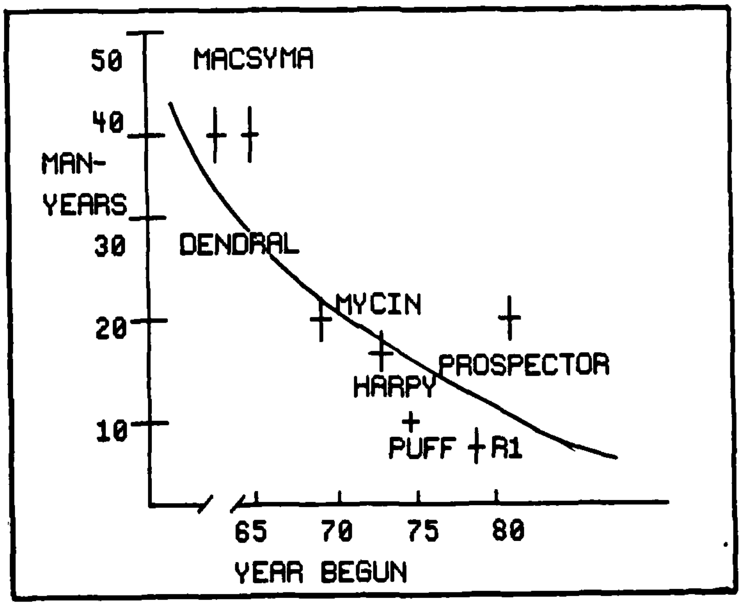


Figure 3-5. Expert System Construction Time (6:10)

Construction time has been reduced due to advancing technology and expert system building aids such as intelligent editors. The building process has been refined through experience and guidelines. The book, Building Expert Systems, briefly explains the guidelines in "the hope that they will be helpful to future knowledge engineers" (12:160). The guidelines are as follows:

1) Task Suitability

. Focus on a narrow specialty area that does not involve a lot of common sense knowledge.

. Select a task that is neither too easy nor too difficult for human experts.

. Define the task very clearly.

. Commitment from an articulate expert is essential.

2) Building Prototype System

. Become familiar with the problem before beginning extensive interaction with the expert.

.Clearly identify and characterize the important aspects of the problem.

. Record a detailed protocol of the expert solving at least one prototypical case.

.Choose a knowledge-engineering tool or architecture that minimizes the representational mismatch between subproblems.

. Start building the prototype version of the expert system as soon as the first example is well understood.

. Work intensively with a core set of representative problems.

.Identify and separate the parts of the problem that have caused trouble for AI programs in the past.

. Build in mechanisms for indirect reference.

. Separate domain-specific knowledge from general problem-solving knowledge.

. Aim for simplicity in the "inference engine."

. Don't worry about time and space efficiency in the beginning.

. Find or build computerized tools to assist in the rule-writing process.

. Pay attention to documentation

. Don't wait until the informal rules are perfect before starting to build the system.

. When testing the system, consider the possibility of errors in input/output characteristics, inference rules, control strategies, and test examples.

3) Extending the Prototype

. Build a friendly interface to the system soon after the prototype is finished.

. Provide some capabilities for examining the knowledge base and the line of reasoning soon after the prototype version is finished.

. Provide a "gripe" facility.

. Keep a library of cases presented to the system.

4) Finding and Writing Rules

. Don't just talk with the expert, watch him or her doing examples.

. Use the terms and methods that the experts use.

. Look for intermediate-level abstractions.

. If a rule looks big, it is.

. If several rules are very similar, look for an underlying domain concept.

. If tempted to escape from knowledge representation formalism into pure code, resist the temptation for at least a little while.

5) Maintaining Your Expert's Interest

. Give the expert something useful on the way to building a large system.

. Insulate the expert, as well as the user, from technical problems.

. Be careful about feeling expert.

6) Building the Operational System

. Throw away the first system.

. In the operational (and later) versions, begin to consider generality.

. Identify the intended users of the final system.

. Make system I/O appear natural to the users.

7) Evaluating the System

. Ask early about how the expert would evaluate the performance of the sysstem.

. The user interface is crucial to the ultimate acceptance of the system.

8) General Advice

. Exploit redundancy.

. Be familiar with the architecture of several expert systems.

. The process of building an expert system is inherently experimental.

Following these guidelines does not ensure success, but it can assist at trouble spots along the way. Whether the task is to find minerals, treat disease, or control a spacecraft, the process of development and construction of an expert system is the same. The next section will discuss Task Suitability for satellite systems and establish the need for expert systems on orbiting spacecraft.

Section IV

Expert System Implementation for Stationkeeping

Expert systems have progressed to a state such that it is reasonable to assume that a program could be developed to perform satellite housekeeping tasks. After thorough analysis of the various tasks for the DCSC III and the necessary upgrades for autonomous operation, it appears that Artificial Intelligence techniques are needed for certain functions. In particular, orbit determination, maneuver planning, manuever control, and automatic station move are precise functions for an expert system. However, not all functions require Artificial Intelligence techniques to achieve autonomy. To address these functions the Air Force Space Technology Center (AFSTC) is currently funding a project called the Autonomous Redundancy and Maintenance Management Subsystem (ARMMS). The ARMMS project is currently being managed at the Space Technology Center as a major thrust for the Satellite Autonomy Program.

Satellite Autonomy Program

Very simply, the design for ARMMS involves distributing mini-computers throughout the spacecraft.

Each processor "will control subsystem functions and implement contingency plans if faults are detected. This network of computers would be centrally managed by a supervisory computer which would also store contingency operations plans" (11:18). The project has developed computer algorithms for the various subsystems and plans to give a ground demonstration, using satellite simulation, in 1986.

The original plan of the Satellite Autonomy Program was to direct future effort to appling advances in spacecraft subsystems and build upon the ARMMS project using Artificial Intelligence (11:27). The ARMMS was to be the first step in autonomy where minimal redesign of the existing satellite would be required. Also part of the original plan was a joint program in autonomous systems technology involving NASA and Air Force Systems Command. The joint program has since been cancelled due to funding constraints, but it established near and far term objectives for spacecraft autonomy. These objectives are (11:43):

Objective 1 (near term)

- Develop and demonstrate by FY 1986 a flight qualifiable spacecraft management system that can perform on-board routine maintenance (including navigation) and fault management without ground or crew interaction.

Objective 2 (far term)

- Develop and demonstrate by FY 1990 a flight qualifiable spacecraft management system that is

> capable of independent task formulation and execution, and which can adapt to external environment changes, using methods of artificial intelligence.

The ARMMS study is designed to meet the near term objective and includes the power, telemetry, tracking and control, navigation, propulsion, attitude control, and communications subsystems. All of the algorithms are making good progress except for minor deficiencies in the navigation area. The accuracy attained by the conventional computer algorithm is about three degrees and it is desirable to reduce that to one degree or less. Navigation includes algorithms to perform orbit determination, maneuver planning, maneuver control, and automatic station move.

There appears to be a very real need for implementation of expert systems in the navigation area. Stationkeeping would then be a combination of various functions within the navigation subsystem. Assuming that the ARMMS project will be successful in all of the other subsystems using conventional computer algorithms, there still appears to be a need for Artificial Inteligence techniques in the navigation area. The reason is that navigaion, in particular stationkeeping, is not a hard and fast science that translates directly into formulas and procedures. There are human experts performing as orbital analysts for spacecraft at the present time. They work with computers, but the process still requires human judgement based on experience and knowledge to correct many spacecraft orbit anomalies.

Orbit Correction Process

Satellite systems are unique due to mission configuration and orbital position, but can be discussed generally for the purposes of understanding the stationkeeping process. Since the immediate need for AI has been narrowed to performing stationkeeping on orbiting spacecraft, a more detailed explanation of the process would be appropriate.

Stationkeeping *functions* begin with orbital position determination. Most of the current satellite systems depend upon ground stations to determine orbital position, but the move to autonomy will dictate the need for accurate earth and sun sensors on board the spacecraft to accomplish this task. The details of the sensors will not be explained here, but the Air Force is funding programs to equip spacecraft with advanced sensors capable of independently determining orbital position. One such program that could be used on the DSCS III is called Multimission Altitude Determination and Autonomous Navigation (MADAN). MADAN "is an advanced star sensor using charged coupled device technology. Two MADAN units plus an earth sensor could determine satellite location and attitude" (11:47). The accuracy (see Figure 4-1) is sufficient for most satellite systems and could be improved to about 400 meters with an improved near body sensor.

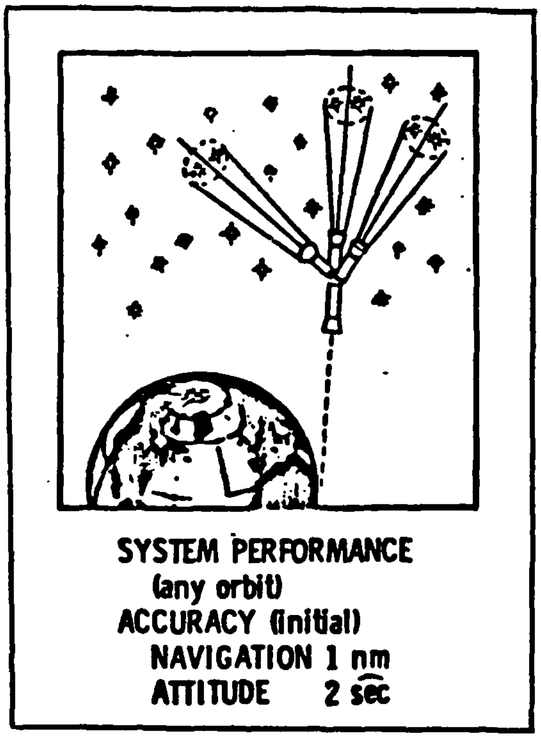


Figure 4-1. MADAN (11:48)

Once the satellite position has been determined, the delicate process of orbit correction begins. It should be emphasized that this process is currently performed on the ground by human experts. Some of these experts work for NASA at the Goddard Space Flight Test Center and some are civilian contractors from Lockheed working for the

Satellite Control Facility. There are other orbital analysts at varying sites in the country but all of them do basically the same job. Most of them are dedicated orbital analysts who concentrate on one or a couple of specific satellites in a particular orbit. It is not possible to get into the intricasies of orbital mechanics, but it is necessary to use two types of orbits (low earth orbit and geosynchronous) as examples to illustrate the process used to maintain proper positions.

Two orbital analysts who work for NASA are Mr. Richard Strafella and Mr. William Weston (26). Mr. Strafella has the responsibility of maintaining the orbit for LANDSAT, which is a satellite system in a low earth orbit approximately 700 km above the earth. Mr. Weston monitors and issues corrections for several systems at geosynchronus orbits, approximately 36,000 km above the earth. Explaining the process these experts employ to maintain proper orbits will illustrate the similarities and differences encountered for various spacecraft.

Satellites in a near circular orbit such as the LANDSAT encounter drag and tend to drift toward the earth. As the orbit changes shape, so does the ground trace or path the satellite shadows on the earth. Mr. Strafella monitors the LANDSAT orbit by ensuring that it passes over a certain longitude within plus or minus 10 km. Orbital data is taken by ground stations for a continuous 32 hours to accurately determine position and make orbit predictions. Actual positions are then plotted on graph paper to illustrate the trend of orbital error. In the case of LANDSAT, a figure of "errlong", or error in longitude is plotted. When that figure indicates LANDSAT will be outside the bounds of 10 km a decision is made by the orbital analyst to make an orbital correction.

The formal exchange between orbital analyst and operational personnel is shown in Appendix C. Mr. Strafella begins the process by sending an Orbit Adjust Request indicating the time and reason for this adjustment. He also selects the ground site to be used to monitor the adjustment based on satellite position. In this example, Mr. Strafella requested an orbit adjust for the LANDSAT-5 be done on 30 April 1984 at 2114:00 hours Zulu time. He made the request on 25 April, 1984 and was able to determine that by the 30th the LANDSAT may be out of the 10 km limit and a correction should be done.

The Orbit Adjust Preplan then provided preliminary data which was needed to suggest thruster burn times. The ground controller provided suggested contact times for the Madrid Station and fuel tank status. The pressure and temperature data were needed to calculate thrust reqired and burn time for thrusters A and C. Given this data from the satellite, calculated semimajor axis distance, and solar flux trends, Mr. Strafella calculated a rough delta v required. Delta v is the change in velocity required to boost the satellite into a slightly higher orbit and correct the anomaly. Refinement of the required delta v is done through repeated simulation runs on a computer using different values for the number of seconds of burn time for two thrusters and varying the solar flux.

The solar flux is an indication of drag which will be encountered by the spacecraft as the maneuver takes place. Mr. Strafella plots the flux on a routine basis to try and determine trends. If the solar flux is large, more drag will be encountered and more burn time wll be required. Likewise, if the solar flux is small then that same burn time may drastically overcorrect the orbit due to lower drag. At this point experience and judgement play a major part in the decision making. The solar flux is an unknown that must be predicted to best determine a solution.

Once the orbit analyst has determined the delta v required and translated it into thruster burn times, he can submit an Orbit Adjust Plan. This plan specifies the burn time for each thruster in milliseconds and provides a two minute window for the maneuver to begin. In this particular example a correction of 159 meters is planned (see semimajor axis change). Orbital period, fuel usage, and pressure change are also calculated by the analyst for comparison after the correction.

The Orbit Adjust Postburn Report is an accurate account of the actual maneuver. In the case of the LANDSAT-5 correction on 30 April, translation thrusters A and C were fired for 3840 msec each and the attitude thrusters B and D fired automatically during the maneuver for 15 counts each to maintain proper attitude. Fuel tank status was extremely close to planned data and varied only slightly in overall pressure and temperature for two of the tanks.

Finally, an Orbit Adjust Postburn Analysis was done to compare planned, replanned, and actual data. The numbers in this case are so close the differences seem insignificant. In this case the solar flux was very close to what was guessed, or predicted, and the maneuver went almost exactly as planned. Due to a slightly different pressure and temperature, the thruster efficiency dropped to about 95% instead of the planned 97%. As a result, the correction which was supposed to have produced a 159 meter orbit change only produced a 157 meter change. Given this information and current solar flux trends the anlyst predicted that the next orbit adjust will be required around the middle of June. This is a complicated procedure with many variables, but it really depends upon the judgement and experience of the orbital analyst to make the final decision.

The process is quite similar for Mr. Weston, who is responsible for several satellites operating at geosynchronous altitude. There are two significant differences though. The first involves the requirement to stay within plus or minus .5 degree of the established longitude versus 10 Km for the LANDSAT. The other difference is the cause of drift, or orbit change, at geosynchronous. At that altitude drag is insignificant, but the satellites do tend to drift out of their orbits. At the geosynchronous orbit there are two nodal points at 105 West and 75 East, points the satellites tend to drift toward. These nodes are stable points and tend to attract distant satellites to them.

The drift rate depends upon the distance a satellite is away from one of the stable points. There are no formulas which determine drift rate for given locations. This drift rate must be plotted by the analyst to determine at what point the satellite drifts outside its particular parameters. The following Figure 4-2 shows the drift for the TDRS-A from January-April 1984. Stationed at 40.5 degrees West longitude, the TDRS-A tends to drift to the 105 West nodal point. From this figure, it is obvious that an orbit adjust was required by 31 March or the satellite would have drifted outside its bounds. As the drift rate increases it also requires more delta v to correct its position.

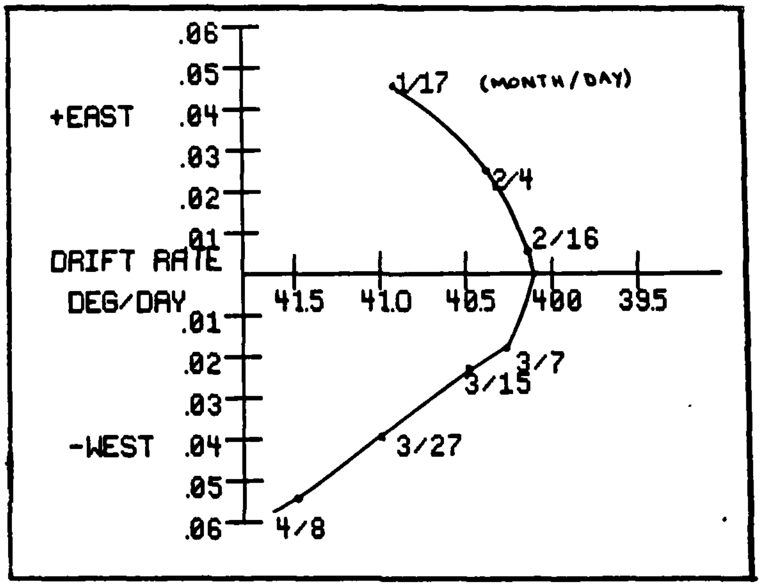


Figure 4-2. Drift Rate for TDRS-A

An orbit adjust would send the satellite to the farthest point, within limits, from the attracting node. This process requires careful calculations for thruster burn times much the same as for the LANDSAT. Although data on DSCS III orbit adjustment was not available , it is reasonable to assume the procedure is quite simlar because the DSCS is also at geosynchronous altitude. Normally, 12 hours of tracking data is accumulated from at least two ground stations to determine the satellite position at geosynchronous orbit. The average longitude is used to plot the position and determine drift rate. Most satellites at geosynchronus altitude require corrections every 30-90 days. The exact time depends upon the particular spacecraft configuration, its location, and the drift rate. Only two of those three factors are givens, the drift rate must be determined by the expert - the orbital analyst. This expert task can be performed using Artificial Intelligence techniques. The system required to perform the task would be similar in design to an existing expert system - R1.

Expert System Design

Previous sections have established the capability of Artificial Intelligence techniques through working expert systems and also the need for AI on spacecraft for stationkeeping. Based on the task structure and available technology, it is reasonable to expect that a stationkeeping expert system would be similar to the existing R1 system. A good analogy can be made for stationkeeping tasks and the design function of R1. A detailed explanation of R1 will serve to point out the similarities of the tasks and illustrate the design process an expert system would need to follow for spacecraft application.

R1 is a rule-based program which configures VAX-11 computer systems for Digital Equipment Corporation (DEC). The program began its development in 1978 at the Carnegie-Mellon University by John McDermott. The system uses a customer's purchase order to determine what substitutions and/or additions must be made to make the order consistent and complete. It then produces diagrams showing the spatial relationships of the 70-150 components which might constitute a finished system. The most recent additions to the program allow R1 to offer configuration recommendations to the individual customer. Given floor space allocations and room configuration (doors, windows, other machinery), R1 can provide plans to optimize the set up and maintenance positions.

R1 seems to provide an excellent analogy to spacecraft navigation. First, R1 is a forward chaining expert system using rules to determine a "best" solution out of many possible combinations. This is not unlike the navigation subsystem which would have to determine corrections needed and proper actions to take. Many combinations of thruster burn are available for stationkeeping maneuvers, but only one combination can be selected at a time.

Second, R1, also called XCON, has grown a great deal since its initial use in 1980 and is a proven system. Its knowledge base was sufficient to begin configuring

VAX-11/780's with about 800 rules. Rules would provide the basis for spacecraft maneuvering in much the same way. As R1 was used to configure other types of computers, rules had to be added. The knowledge base currently contains over 10,000 rules and can be used to configure more than six different computer systems (See Figure 4-3). The growth potential is an important factor in the design of a program for spacecraft navigation. The navigation task may begin to simply correct orbit anomalies, but it could grow to include optimization of fuel levels and spacecraft life through carefully orchestrated maneuvers.

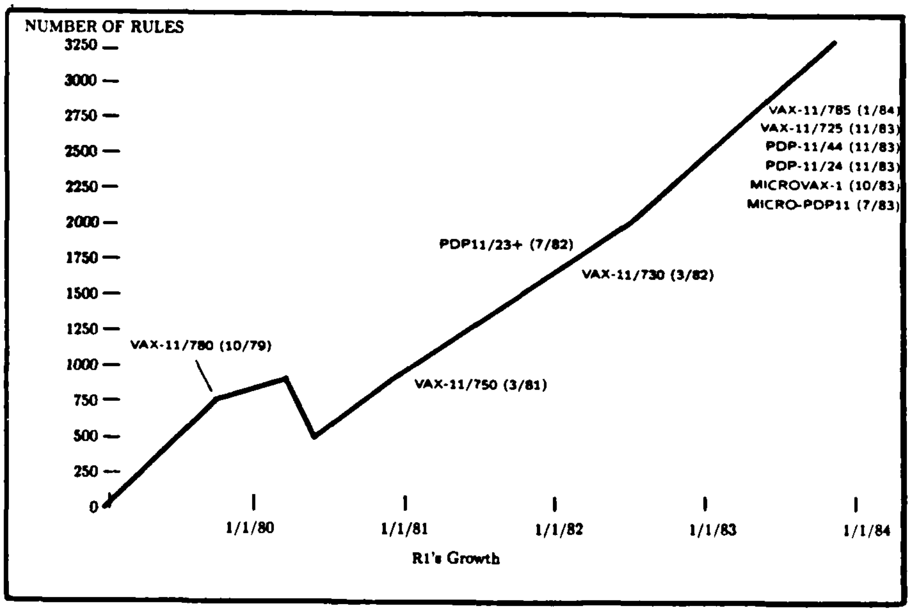


Figure 4-3. R1 Configuration Capability (3:22)

Expert System Development

Development, as well as design, of an expert system program to provide spacecraft stationkeeping would also be similar to that for the R1 system. Work on R1 began in 1978. Initial effort was spent on developing a demonstration version of R1 to convince DEC to pursue the program. Similarily, it would be necessary to provide a demonstration of spacecraft navigation capabilities for a typical system to convince satellite managers. By October 1979, R1 had 750 rules of computer configuration in its knowledge base and a data base which consisted of 450 component descriptions (3:23). By 1981, R1 had proven itself and DEC was anxious to expand its capabilities to configure more systems. Having added 4050 component descriptions to the data base and several thousand rules, R1 could configure all of DEC's biggest selling computer systems by late 1983.

The development of R1 has been incremental. There are basically four reasons why knowledge was added to R1 over the four year period. They are (3:25):

1) To make minor refinement (adding knowledge to improve R1's performance on an existing subtask).

2) To make major refinement (adding the knowledge required for R1 to perform a new subtask).

3) To configure new system types.

4) To extend the definition of the configuration

task in signinficant ways.

Of the knowledge added, about 65% of the rules extended the general configuration capabilities while the remaining 35% accounted for new rules specific to a single system type (3:25). Using this information, it appears to have been relatively easy to add new system configuration capability because most of the added knowledge expanded the initial task.

Why AI versus Conventional Programming?

The initial satellite stationkeeping task should be designed and developed at the simplest level. The first demonstration of R1's capability was limited to configuration of one type of computer system and with a limited number of components. The expert system designed to provide spacecraft maneuvering capability should also be demonstrated to show basic orbit maintenance. The critical components during the design and development of an expert system for spacecraft use will be the knowledge base and the data base.

The knowledge base will be filled with rules based on the input of one or more orbital analysts Since the tasks for stationkeeping vary slightly with the altitude of the satllite the simplest approach would be to build the knowledge base to analyze one type of orbit. Building the knowledge base around the geosynchronous altitude would involve analyzing such factors as current position, drift rate, and predicted position. Subroutines must be built into the knowledge base to calculate spacecraft position using sensor information from the data base. Current position must then be checked against established parameters for orbit accuracy. If the current position is outside the established boundaries, a orbit correction must be made.

Once it has been determined that a correction is needed, the knowledge and data bases must interact closely to calculate a solution. Drift rate affects how much delta v is required to correct position and must be determined. Drift rate is a factor of spacecraft position in relation to the two stable nodes and also the previous drift rate. Subroutines in the knowledge base must calculate drift rate each time position is determined and load that information into the data base. The data base would then transfer the current and past drift rates into the knowledge base where trend anlysis must be done. At this point, the expert system must call upon the expert knowledge and employ rules of thumb for drift rate projections. The expert system program must be able to make a "best guess" or estimate of what the drift rate will be at the time of thruster burn. It may seem that this prediction of drift rate is unnecessary if all data is current and orbit correction can be done immediately, but that is not the case. Many more factors have to be analyzed before thruster burn duration can be proposed.

Factors such as propulsion system status to include pressure and temperature will affect the amount of thrust produced by various systems. This data must be fed to the data base and then transferred to the knowledge base for verification of system status. A rough delta v may then be determined using relatively simple formulas. That velocity requirement must then be used to calculate the time of thruster burn. The maneuver must be simulated using elaborate satellite models and current system status. Simulation results would be analyzed for various thruster burn times and the "best" option would be selected.

This process sounds relatively simple until other aspects of spacecraft maneuvering are taken into account. Sensor accuracy must be verified based on previous positions. Sensor outage due to solar/lunar interference must be predicted. Spacecraft attitude, inclination, longitude, and eccentricity must all be controlled during the maneuver. The mission of the spacecraft must be considered during the station move to include antenna pointing and other payload restrictions. Even the initial design for an expert system to provide autonomous spacecraft stationkeeping must include most of these considerations if the system is to provide a stable and reliable platform.

Information required in knowledge base and data base (not inclusive):

Knowledge Base (rules)	Data Base
Orbit determination	Sensor info
Orbit prediction	Thruster status
Drift rate determination	Pressure
Drift rate prediction *	Temp
Calculation of Delta V	Historical info
Select Thrusters *	Position
Determine burn times	Drift rate
Simulate maneuver	Spacecraft mass
Mission consideration *	Component info
Select best burn time *	Payload
	Power

* currently done by human experts

The human expert currently must make decisions such as firing the thrusters 3.8 sec versus 3.9 sec depending on the results of the simulations and how accurately he feels all of the factors have been evaluated. In the example presented in Appendix C, the analyst assumed the thrust effeciency would be about 97%. It was actually only 95% and the correction was low by a few meters. If the analyst had assumed lower effeciency, and in fact it was higher, then the orbit may have been over corrected. Selection of the "best burn time" is not always the biggest correction, but it may involve being conservative to account for varying factors or incomplete knowledge.

The human may seem to be a small part of the entire process, but it is a critical one. It is plausible to develop conventional computer algorithms to correct a satellite orbit given normal circumstances. On the other hand, autonomy demands the program to execute proper corrections for ALL circumstances. Computer algorithms cannot account for sensor outage or predict low power levels due to solar eclipse. To try and build a Fortran program which could handle every type of orbit correction and under all conditions would be next to impossible. Even if a conventional program could be built using optimization techniques, what hapens when a thruster malfunctions and the weights must be changed to reconsider the problem? AI techniques allow an expert system to monitor all pertinent factors and use only the information which is available to perform the task at hand, even if the data is not complete.

Flexibility and the ability to deal with complex problems are advantages of AI programming techniques. There may be many goals during an orbit correction relating to mission considerations, position corection, fuel optimization, accuracy of sensor data, and storage and relay of maneuver procedures. Operations Research goal programming and optimization techniques can be used but the goals may change drastically for different corrections. Changing the weights of various factors for each orbit adjustment would have to be done by ground personnel and autonomy is lost. Also, changes to normal algorithms would affect other portions of the overall program and require programming experts working in concert with the orbital analyst.

NASA has made progress in a similar area of spacecraft navigation and recently disclosed information relating to a navigation expert system (NAVEX). NAVEX can be used to assist the space shuttle during reentry into the atmosphere (16:79). The system uses AI techniques and is programmed in the LISP language. The expert system reportedly can handle more data and make accurate decisions more quickly than the current team of human controllers and conventional computers. One of the engineers with the program said that the conventional programming techniques did not allow for *rapid decision-making or changing circumstances* (16). The NAVEX has an elaborate knowledge base capable of modifying the approach and quickly adjusting to the situation. After its performance has been thoroughly verified, NAVEX may be able to replace two thirds of the current human controllers who perform the same task (16:79).

Performance and Reliability

Performance is as important to NASA as it was to DEC with the R1 system. As dependence on R1 for system configuration grew, reliability became an issue. The initial measurement for success was "percentage of totally

	1st Qtr 1980	2nd Qtr 1980	3rd Qtr 1980	4th Qtr 1980	1st Qtr 1981	2nd Qtr 1981	3rd Qtr 1981	4th Qtr 1981	1st Qtr 1982	2nd Qtr 1982	3rd Qtr 1982	4th Qtr 1982	1st Qtr 1983	2nd Qtr 1983	3rd Qtr 1983	4th Qtr 1983
Incorrect Rules																
Problem Instances	6	21	12	26	55	-	-	171	123	313	243	342	100	58	13	80
Distinct Problems	6	7	8	13	18	-		30	25	41	38	37	16	14	6	19
Total Orders	54	194	133	210	824	-	-	3605	4283	7100	7503	8110	8192	8427	10775	20241
Problem Instances Percent	11.1%	10.8%	9.0%	12.4%	6.7%	-	-	4.7%	2.9%	4.4%	3.2%	4.2%	1.2%	0.7%	0.1%	0.4%
Distinct Problems Percent	11.1%	3.6%	6.0%	6.2%	2.2%	-	-	0.8%	0.6%	0.6%	0.5%	0.5%	0.2%	0.2%	0.1%	0.1%
Missing Part Descriptions																
Problem instances	6	51	16	22	33	-	-	136	78	214	629	997	697	535	906	1962
Distinct problems	6	44	16	19	31	-	-	43	42	40	105	157	162	141	368	472
Orders with errors	4	21	7	12	19	-		104	65	166	439	601	535	464	920	1275
Incorrect Part Descriptions																
Problem instances	1	9	17	15	21	-		74	31	86	94	86	23	19	10	28
Distinct problems	1	3	3	6	12	-	-	12	11	16	9	19	8	6	5	9
Parts Subtotal																
Problem instances	7	60	33	37	54	-	-	210	109	300	723	1083	720	554	916	1990
Distinct problems	7	47	19	25	43	-		55	53	56	114	176	170	147	373	481
Total orders	54	194	133	210	824	-	-	3605	4283	7100	7503	8110	8192	8247	10775	20241
Problem instances percent	13.0%	30.9%	24.8%	17.6%	6.6%	-	-	5.8%	2.5%	4.2%	9.6%	13.4%	8.8%	6.6%	8.5%	9.8%
Distinct problems percent	13.0%	24.2%	14.3%	11.9%	5.2%		-	1.5%	1.2%	0.8%	1.5%	2.2%	2.1%	1.7%	3.5%	2.4%
Parts and rules Subtotal																
Problem instances	13	81	45	63	109	116	190	381	232	613	966	1425	820	612	939	2098
Distinct problems	13	54	27	38	61			85	78	97	152	213	186	161	384	509
Total orders	54	194	133	210	824	1304	2040	3605	4283	7100	7503	8110	8192	8427	10775	20241
Problem instances percent	24.1%	41.7%	33.8%	30.0%	13.3%	8.9%	9.3%	10.5%	5.4%	8.6%	12.8%	17.6%	10.0%	7.3%	8.7%	10.4%
Distinct problems percent	24.1%	27.8%	20.3%	18.1%	7.4%	-		2.3%	1.8%	1.4%	2.0%	2.7%	2.3%	1.9%	3.6	2.5%
Operational Problems																
Problem instances	2	45	16	56	77	115	67	12	7	2	41	0	11	1	0	0
Controversial Issues																
Problem instances	0	1	3	3	9	11	47	80	21	33	19	26	50	46	19	25
Distinct problems	0	1	2	1	3	-		4	3	8	9	5	8	7	4	4
Desired Enhancements																
Problem instances	-					-		38	6	19	11	57	86	155	25	27
Distinct problems	-			-		-		3	2	6	1	2	5	7	3	4
Bogus Problems																
Problem instances	0	3	1	7	27	16	31	15	15	36	76	62	44	23	33	43
Total Problem Reports																
Problem instances	15	130	65	129	222	258	335	526	281	703	1113	1570	1011	837	1016	2193
Distinct problems	15	103	46	102	168			119	90	149	279	282	254	199	424	560
Total orders	54	194	133	210	824	1304	2040	3605	4283	7100	7503	8110	8192	8427	10775	20241
Problem instances percent	27.8%	67.0%	48.9%	61.4%	26.9%	19.0%	16.4%	14.6%	6.6%	9.9%	14.8%	19.3%	12.3%	9.9%	9.4%	10.8%
Distinct problems percent	27.8%	53.1%	34.6%	48.6%	20.4%			3.3%	2.1%	2.1%	3.7%	3.5%	3.1%	2.4%	3.9%	2.8%

Figure 4-4 R1 Performance

correct orders." This measurement did not discriminate between gross error and insignificant error. Figure 4-4 provides a detailed account of R1's performance over the past four years. It also shows the tremendous increase in usage. There has been a significant reduction in the percentage of problems attributable to missing or incorrect rules. Other areas of note in this evaluation are Controversial Issues and Bogus Problems. Controversial Issues include errors identified by human experts where the configuration works, but it may not be the same way a human expert would have configured to system. The area of Bogus Problems represents the number of times a human expert said the R1 configuration was wrong and, upon detailed examination, found out that the R1 configuration was the most correct.

R1 has configured over 80,000 cases and is still not perfect. A key issue about expert systems is that they will probably never be correct 100% of the time. They can only operate at the same level as a human expert. Just as human experts are not perfect, it is not reasonable to expect perfection from the expert system. The domain R1 works in is constantly changing, as is the space environment, and knowledge can be added or deleted as necessary to meet the situational demands.

The progression for R1 from initial development to implementation was short, but the changes necessary to build confidence and reliability were extensive and done over a four year period. "Expert systems supposedly are easy to develop incrementally, and, at some point, become as good as human experts. R1 lends some credence to both of these claims "(3:32).

Support for the first claim is the fact that, although R1 development was extensive, the process of gradual change over four years has increased system knowledge substantially without having to "start from scratch" with each change. The second claim is substantiated by the fact that the number of times human experts erroneously concluded that R1 misconfigured systems is about equal to the actual number of misconfigurations (3:32).

Given this simplified explanation of the workings and success of R1 and the process of orbital adjustment as performed for the LANDSAT and other satellites, it is reasonable to conclude that an expert system could be designed to perform spacecraft navigation. Orbital analysts would have to help create the rule base for the system and the data base would quickly be filled with satellite unique information. Now the real question: why should expert systems be placed on board satellites, and is it cost effective?

Section V

Expert System Feasibility for Spacecraft

The previous sections have established that AI techniques can perform satellite stationkeeping needs. The worth of that option is the subject of this section. The need for spacecraft autonomy will be explained in more detail as the groundwork is laid for a comparison of the existing ground control network and the proposed expert system. A brief description of the satellite ground control network will outline its complexity and vulnerability. Emphasis will then shift to the machines on which expert systems operate to include the size and weights of current equipment. A rough cost analysis will then be done to compare the existing ground support costs for the DSCS to the cost of implementation of an expert system.

Autonomy

Space assets have become critical to many countries, but particularly the United States. The use of space has matured and is increasing. Many systems vital to national security have been placed in orbit as the Department of Defense exhibits a willingnesns to make use of the "high frontier." Current space systems provide communication and navigation links and surveillance and meteorological information. Survivability becomes an issue when dependence on space platforms is established. The satellites are only one element of a space system. The ground stations and control centers are also vital parts of the entire system (see Figure 5-1).

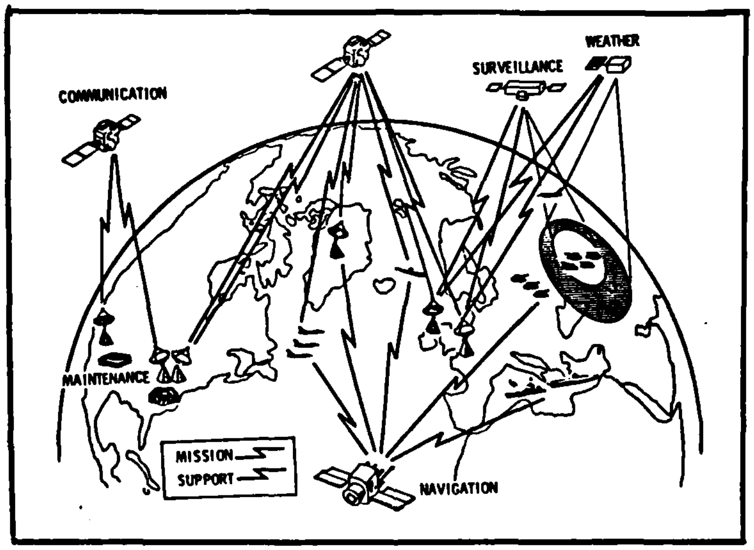


Figure 5-1. Military Dependence of Space (11:4)

This current system presents a survivability problem because the satellites depend on vulnerable fixed ground stations for support including (11:5):

1. Keeping the spacecraft healthy.
2. Maintenance of spacecraft and payload.
3. Location of platform.
4. Providing sequences of commands for mission performance. Commands can be either real time or stored.

Satellite dependence on the ground stations affects spacecraft endurability. Endurability refers to the "ability of the spacecraft to maintain a required level of performance during its designated life span throughout the spectrum of conflict" (11:5).

Ground System

The current ground system is responsible for the control of 80% of all U. S. military satellites and operation costs are in excess of $400 million a year (11). The Air Force Satellite Control Network (AFSCN) is the organization tasked with the Air Force command and control responsibilities. The mission is explained quite well by the AFSCN Communications Operations Concept (1:1-1):

> The AFSCN is a global network of space and ground tracking, telemetry, command, mission operations and data transfer resources that support manned and unmanned Department of Defense (DoD) and National Aeronautics and Space Administration (NASA) programs. Figure 5-2 shows the three basic elements of the AFSCN: (1) control centers, (2) remote ground facilities (RGFs), and (3) communication links.

This document goes on to say that

> Space communications through all levels of conflict was a major factor in consolidating space programs and ground control elements to establish a military space network called the AFSCN.

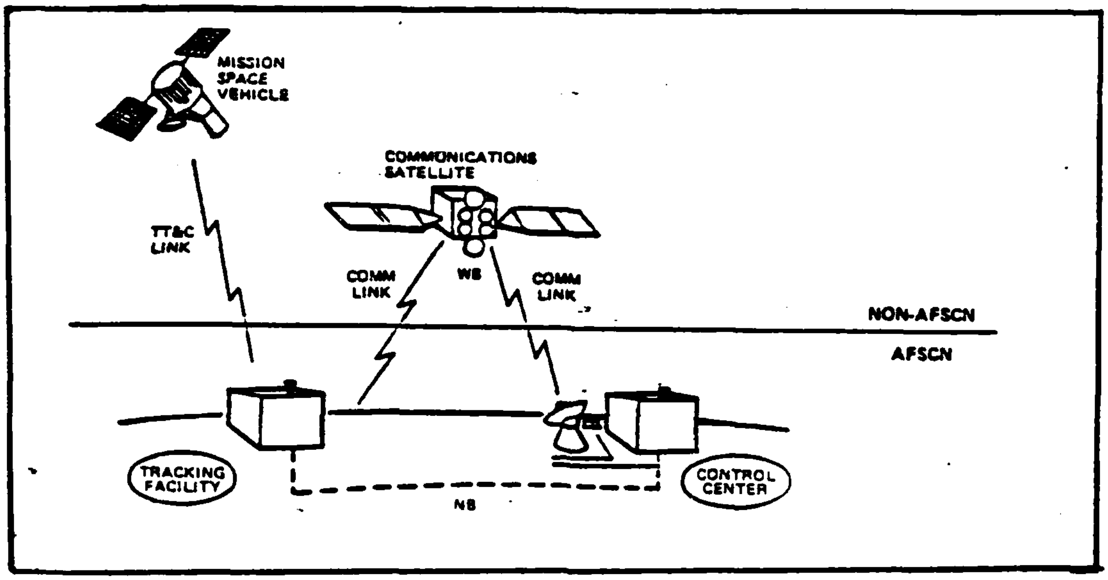


Figure 5-2. Elements of the AFSCN (1:1-2)

The command and control segment of the AFSCN is divided into five functional areas; mission control, range control, data distribution, system development and support, and remote interface. The first four are performed at the Satellite Test Center (STC), Sunnyvale AFS, California. Remote interface is done by seven Remote Tracking Stations (RTS) located world-wide.

The mission control function includes personnel and equipment distributed among eight mission control complexes (MCCs). The MCCs are the hub of the command and control segment and do mission planning, contact support, and post contact evaluation (9:9). A contact support plan is developed to coordinate control execution of the entire satellite control network during contact with the satellite. During satellite contact, tasks such as orbital maneuvers and routine health checks are made. This process may require two separate satellite contacts. Preliminary contact is made during a satellite pass over a designated RTS to collect information relating to orbit determination, command analysis, payload performance, and supporting resource status.

The next satellite contact would be real-time execution of directed commands. It may be necessary to make an orbit adjustment, reconstruct or load memory locations, or transfer fuel for proper distribution. All, or any of these actions and many more which relate to the health and maintenance of a satellite are planned on the ground by system experts before they can be directed to the spacecraft. After final communication with the designated spacecraft, an evaluation is conducted to include analysis of orbital parameters, performance and trends, command and control links, resource distribution, and the payload. Data collected during the evaluation is used to plan for the next contact event (9:24).

Range control, data distribution, and system development are the other three functions performed at the STC. Range control is responsible for controlling access to the capabilities available at the various Remote Tracking Stations and Mission Control Complexes. The Range Control Complex (RCC) manages range planning, schedule control, system control, and maintenance control (9:12). Data distributionn is responsible for the secure transfer of data between the RCC, MCCs, and the communication equipment at the Satellite Test Center. Finally, system development and support provide an independent center for software development, testing, training, and general management information support. That center, the System Development and Test Laboratory (SDTL), provides the necessary operational environment and simulation capability to accomplish the assigned tasks without tying up critical resources (9:64).

The fifth functional area of the command and control segment, remote interface, is located at the Remote Tracking Stations. The seven RTSs located worldwide provide the critical telemetry, tracking, and command (TT&C) link between the spacecraft and the satellite test center. Location of the RTSs dictates the primary mission and coverage capability. The Vandenburg RTS, located at Vandenburg AFB, California is used for launch support and ballistic missile support in addition to on-orbit support. The Thule, Greenland RTS provides support for all polar orbiting satellites on each of their revolutions. The Indian Ocean RTS monitors orbit injection during high altitude launches and is the primary station for orbital tests of the Space Shuttle. The RTS in Guam privides orbital suport for most of the synchronous spacecraft

(9:84). The rest of the RTSs have key locations and provide valuable data to the Satellite Control Network. The network is extensive and complicated, but necessary. A viable alternative to this complex network is satellite autonomy.

Satellite autonomy is the capability to perform routine health and maintenance functions onboard, independent of vulnerable ground support , for a specified useful period of time. Autonomy would allow spacecraft to receive new commands from surviving military users in a crisis situation. If the vulnerable fixed ground centers are destroyed, the satellites will have the capability to operate independently until contact is made by surviving units such as the Transportable/Mobile Ground Station (see Figure 5-3).

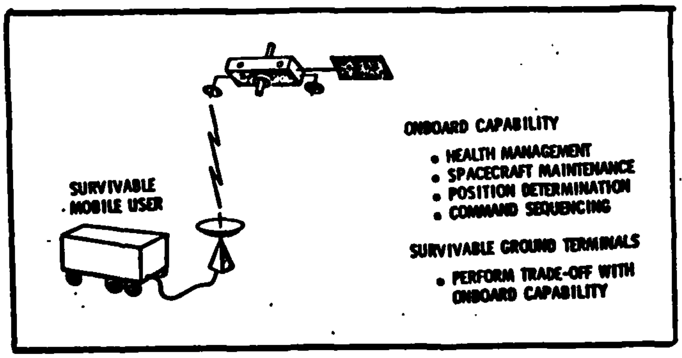


Figure 5-3. Mobile Ground Control Concept (11:14)

The future of satellite autonomy lies in the ability to produce computers that can close the control loop on board. There must be a transfer of command and control capability from the ground to onboard computers. This "transfer requires implementing past spacecraft experience into health, maintenance and navigation algorithms" (11:16). Once this experience and that of the human ground control experts is tapped, the hardware and software package need to be put on an operational satellite.

Expert System Machines

Five years ago it would not have been realistic to discuss putting a hardware and software package using AI technology into space due to size and weight limitations. During that time however, computers designed to operate with AI techniques have progressed rapidly. Likewise, competition has increased significantly and the market for expert system machinery is alive. Cost, although still high, is competitive when overall capabilities and flexibility are considered.

There are many emerging companies focusing on the AI market. First, a brief explanation will be provided to answer the question why AI has prompted "new" companies to develop unique "Artificial Intelligence Computers." The field of AI has been using a language called LISP for over

20 years. LISP offers greater symbol manipulation and generally a more flexible environment in which expert systems can develop. It has evolved into a language that can deal with complex and unpredictable data that traditional programming techniques cannot handle. Powerful sets of editing and debugging tools have been developed for programming in the language. As a result, large, complex programs can be written, tested, and modified much more easily with LISP than with any other programming language.

Before LISP machines were developed, however, the LISP programs were run on traditional mainframes. As most of those machines were designed to use Fortran and made use of numerical language optimization, LISP programs incurred substantial software penalties and included a great deal of overhead for proper execution. The LISP language then "remained in the research lab, where functionality, rather than speed, was the major consideration" (23:2-1).

AI laboratories were started in the early 1970s at Carnegie-Mellon University and Massachusetts Institute of Technology to further research. A direct result of those labs was the development of LISP machines. The machines were designed to run LISP efficiently. The first stand-alone LISP machine made at MIT was developed in 1977. Since that time, people involved with developing the initial machines have founded corporations and introduced LISP machine technology to the marketplace.

Two of the big names in the commercial world are Symbolics and LISP Machine, Inc. (LMI). Other major computer companies are also joining the LISP machine competition. Texas Instruments is now promoting a LISP machine said to be the most powerful, smallest, and lightest available. That machine is so new that detailed specifications are not available. Symbolics and LMI each have LISP machines in the marketplace and have made tremendous advances in technology during the last five years.

Symbolics' newest machine is called the 3670. The main processor has been reduced to only 24" wide by 55" high by 34.5" deep and weighs about 450 pounds. That is smaller than a standard four-drawer filing cabinet. For comparison, LMI produces the LAMBDA series of LISP machines and its system main cabinet is 21" by 60" by 35.5" and weighs about 500 pounds. The Texas Instrument machine is said to be about one-half the size of the current state-of-the-art, but exact measurements are not known.

Results of commercialization of AI dedicated machines have been increased capability, reduction in size and weight, and competitive cost. Cost will be further explained in the Cost Analysis section. Capability, size and weight of LISP machines are critical factors when considering them for spacecraft application. The DSCS III has a dry weight of 2000 pounds and is capable of carrying 600 pounds of propellent. The addition of 500 pounds for an expert system would not seem feasible at this time, but reduction in weight seems to be a factor of advancing LISP machine technology. As the new machines are developed, such as the one by Texas Instrument, size and weight are reduced. New computer technology such as Very High Speed Integrated Circuitry (VHSIC) will also serve to reduce size and weight of the LISP package substantially.

The computer currently used on the DSCS III is the PDP-11 made by General Electric and Digital Electronic Corporation (27:86). It is one of six computer systems that have been space qualified by Air Force Systems Command Space Division. The range of the weights for the six systems is from 40 pounds to 110 pounds. The Department of Defense has initiated a VHSIC program costing $320 million over six years which should yield significant benefits for space application (27:95). More detailed cost information unique to expert system development for spacecraft navigation will be presented next.

Cost Analysis

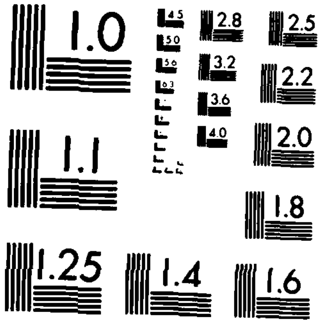
Spacecraft autonomy is an important factor in increasing the survivability of space assets, but in this day of tight budgets, cost is equally important. The use of expert systems on spacecraft could be used to reduce the dependence on ground centers, but at what cost? If the system cannot be made cost effective, it is not likely to be implemented. The previous section highlighted some companies in the AI machine building business. Given that competition and technology will only lower the costs and improve the capabilities, it is safe to assume that the cost of present day machines provide a good baseline for discussion.

Lisp Machine Cost

The government price for a Symbolics 3670 LISP machine is $102,500. That is a small amount considering that the cost of most complete satellite systems is in the hundreds of millions. The initial price of the LISP machine is deceiving because it is not a space qualified computer system. Seven characteristics must be considered when building a computer system for spacecraft use: 1) throughput, 2) memory, 3) input/output, 4) electrical power requirements, 5) reliability, 6) parts qualification, 7) radiation hardness (27:86). It is not within the scope of this project to explain each of these characteristics, but suffice it to say that meeting space qualifications is a costly process. It is a conservative estimate that initial hardware costs may increase by as much as ten fold due to stringent requirements. Therefore, the LISP machine hardware could cost as much as $1.02 million. After production, computers must be put through rigid space

AD-A152 078 ARTIFICIAL INTELLIGENCE IN SPACE PLATFORMS(U) AIR FORCE 2/2
INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF
ENGINEERING M A WRIGHT DEC 84 AFIT/GSO/OS/84D-9
UNCLASSIFIED F/G 6/4 NL

END
5-85



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963 A

qualification tests which cost about $1 million each (27:90). Adding hardware ($1.02 million) and space qualification ($1.0 million) costs raises the cost of the LISP machine to $2.02 million each. This is a more reasonable figure and one which can be used to make approximate cost comparisons.

Development Cost

Development costs for expert system programs have decreased during the past 10 years due to increased experience, developmental aids such as intelligent editing programs, and the ability to make use of existing AI systems. As already pointed out, it would be possible to use R1, computer configuration system, as a base for further refinement of the spacecraft navigation problem. If the time to develop the new expert system follows the pattern established by previous examples, it could take five man-years to complete.

Using the DSCS system as an example, the process would necessarily involve one orbital analyst devoted to filling the knowledge base for the navigation task. Assisting that expert would be one or two "knowledge engineers" who would be responsible for integration of the knowledge into the knowledge base and construction of the inference engine. Another technician would be resposible for filling the data base with DSCS particular information concerning thrusters, fuel capacity, subsystem interactions, and sensor interpretations. Given that these highly skilled people could command as much as $100,000 in salary per year, plus allowing some room for extra personnel assistance, the development cost should be about $500,000. This figure might seem quite low, but it is only the cost associated with development to the demonstration phase.

After the system has demonstrated adequate performance and been given further approval, more costs will be incurred as the same basic group of skilled people refine and test the system. Testing an expert system for spacecraft use would have to done on the ground due to the access needed for changes and fine tuning. It is reasonable to expect that the fine tuning period might take as long as it has taken R1 to become approximately 98% efficient. That time, from 1980-84, when R1 was being used and modified represents four years of refinement necessary to reach desired performance goals. Figuring that the basic development group is composed of four members for four years adds another $1.6 million to "development" costs before the satellite navigation expert system is ready for implementation. Thus the cost of development and testing an expert system for use on the DSCS is approximately $2.1 million.

The cost of implementing LISP machine design into space qualified hardware was previously given to be approximately $2.02 million. The cost to outfit an entire satellite system such as DSCS, which is currently made up of six DSCS IIs and one DSCS III, with the necessary expert system to perform autonomous navigation would be around $14.14 million. As previously mentioned, this cost is in addition to costs necessary to provide the appropriate improved sensor capability.

NAVIGATION EXPERT SYSTEM COSTS ($ MILLION)

Initial Development to Demonstration		.5
Refinement and testing		1.6
Hardware for current DSCS (7 satellites)		14.14
	Total	16.24

The DSCS orbital support is currently being provided by contract personnel at the Satellite Test Center. The orbital support contract employs about 53 civilians who are divided into shift workers and administrative personnel. The annual contract cost is approximately $3.25 million. Once the expert system reaches the demostration phase, it is likely to be able to reduce the need for contract personnel. R1 was able to function quite well in its initial capacity to configure one type of computer system, relieving human experts of that task. Some orbital analysts would still be required, but only to monitor the system and verify the orbit corrections. If personnel were reduced by only one-third, a conservative estimate, the first year savings would be: $1.08 million (contract savings) minus $.5 million (development cost to demonstration) equals $.58 million.

It makes more sense however, to analyze the savings at the end of four years when the expert system should be fully operational. By that time refinement and testing expenses make the total cost of the expert system $2.1 million. After the expert system has established itself and proven to be reliable, it would be possible to further reduce contract personnel. After four years of testing and adjustment, implementation of the spacecraft should be able to reduce the orbital support by about two-thirds. This is the same strategy being considered by NASA with the advent of NAVEX. The remaining personnel should be able to perform periodic checks on spacecraft position and verify autonomous performance. It may be necessary to update position or make mission specific requests which affect the orbit such as changing inclination to accommodate users. If the personnel and contract costs could be reduced by two-thirds, the annual savings would be about $2.15 million. During the fifth year then, the savings would be $50,000. Every year after that results in a savings of about $2 million, subtracting expenses for maintenance of the expert system. The savings are attractive, but autonomy has not been achieved and the link between ground and space is still vulnerable.

Inplementation of a space qualified expert system is the way to achieve autonomous satellite navigation. The cost to reach this stage was previously given to be $2.02 million for each satellite. This was assuming a ten fold increase in current hardware cost to develop a space qualified LISP machine. The following table presents the break even point (in years) by allowing hardware cost to vary from 5 to 15 times current prices and analyzing cost savings as the DSCS orbital support program is changed for personnel reductions from 1/3 to 2/3.

The final cost for a seven satellite system including machine cost, space qualification cost, and RDT&E is:

$12.6 million (assuming 5 fold increase)
$16.24 million (assuming 10 fold increase)
$19.8 million (assuming 15 fold increase)

Contract cost savings are calculated using $3.25 million as a base and reducing it by 1/3, 1/2, and 2/3.

Break Even Point (in years)

Expert System Cost ($ million)	Annual Contract Cost Savings ($ million)		
	1.08 (1/3)	1.63 (1/2)	2.15 (2/3)
12.6	11.5	7.7	5.8
16.24	15	10	7.5
19.8	18	12	9

Using this table as a basis for discussion it is obvious that if the DSCS has a lifetime of 8-10 years and if personnel are reduced by only one-third, cost savings do not materialize. The more likely cases fall between personnel reductions of one-half and two-thirds and in five of the six cases a break even point is reached or cost savings are realized.

These are rough figures used for comparison and it is obvious that implementation of an expert system would not pay for itself in the first year. It is interesting to note, however, that as the satellite extends into its useful lifetime, 8 to 10 years for the DSCS III, the cost savings do materialize in all except the extreme cases.

There are many associated costs that are difficult to analyze such as reduction of use of the Remote Tracking Stations for routine orbit adjusts, costs of expert system maintenance and modifications to existing satellite systems or future systems. It is certainly difficult to associate cost savings with increased survivability which is a result of autonomous spacecraft operation. Even the most conservative estimates result in a cost savings or break even point within the useful lifetime of the spacecraft. Spacecraft autonomy is a valuable asset that would be nice to get for free, but it may also be worthwhile to pursue it at some small cost to the overall program.

Obviously, these are fuzzy numbers and long projections which may not be 100% accurate. The fact remains, cost savings can be attained and satellite survivability can be increased by autonomous spacecraft systems. Artificial Intelligence techniques and expert systems can provide increased satellite autonomy at reasonable costs.

Section VI

Conclusion and Recommendations

This study has applied an aspect of Artificial Intelligence (expert systems) to the development of new technology for practical applications. The field of AI pertaining to expert systems was narrowly defined as "the building of machines which can mimic intelligent human behavior." There is much more to AI than just expert systems. AI research includes robotics, natural language processing and more. However, the initial focus of this study was implementation of an expert system on space platforms to perform housekeeping tasks. A review of expert systems demonstrated current capabilities and established the possiblity of implementation for spacecraft tasking. Spacecraft tasking for the DSCS III was analyzed in detail and deficiencies in autonomy levels were noted.

Artificial Intelligence research has made it possible to build expert systems capable of performing at the expert level in many narrowly defined areas. Through the use of an extensive knowledge base, expert systems can use sophisticated problem solving techniques to produce results beyond the scope of conventional computer programs. Application of AI techniques in expert systems is limited to narrowly defined tasks much the same as human experts are limited due to specialization of knowledge and experience needed. Experience is translated into heuristics for the expert system. The program is then able to operate even in the absence of complete knowledge.

Many successful expert systems such as MYCIN, PROSPECTOR, and R1 use heuristics to fill in gaps of information. Expert systems are able to provide "expert decisions" which have been evaluated to be as correct as human experts'. PROSPECTOR continues to assist geologists on specific mineral deposit problems. MYCIN has provided reliable medical advice in the area of infectious blood diseases and is ready for clinical use. R1 has been used exclusively by Digital Equipment Corporation to configure computer systems and is saving them time and money. The increasing number and expanding capabilities of expert systems made the idea of implementation on a spacecraft the next logical step.

That possibility was further examined by taking a close look at the spacecraft tasking to determine the need for AI techniques. An autonomy assessment of the DSCS III oulined tasks which needed enhancement in order to increase satellite autonomy. The Space Technology Center is proposing to increase satellite autonomy with a project called the Autonomy Redundancy and Maintenance Management

Subsystem. It was determined that an expert system is needed most to perform accurate and independent stationkeeping and could be augmented by mini-computers being proposed by the ARMMS for other routine tasks. The Space Technology Center has proposed that significant progress can be made in spacecraft autonomy with the implementation of ARMMS. The inclusion of an expert system for navigation would only serve to increase that progress.

It is likely that as AI technology is developed and an expert system is designed for spacecraft stationkeeping, all facets of satellite housekeeping can be incorporated into the program. The expert system could be destined to perform the overall spacecraft controlling function using inputs from distributed mini-computers. That possibility exists because AI provides the flexibility and capability to interact with various components, maintain overall control, and provide intelligent feedback to ground personnel. Since the interaction capability of expert systems is still being developed, this study limited spacecraft tasking to orbit maintenance or stationkeeping.

It was then possible to provide an explanation and examples of orbit maintenance by examining the process as done by orbital analysts at Goddard Flight Test Center. The orbit correction task was presented for a low earth orbit and contrasted with that for a geosynchronous orbit.

Although the factors causing orbit degradation varied for the two altitudes, the process of correction was quite similar. Both cases require human judgement to select the final spacecraft maneuver for orbit correction. This judgement is considered to have been developed by experience and gives credibility to the title "expert" for an orbital analyst. It would be necessary to transfer this expertise to a working expert system.

Research pointed to an expert system developed by Digital Equipment Corporation, R1, which was similarly designed and had proven itself reliable. A case was then made for the parallel design and development of an expert system for spacecraft stationkeeping. During the design of an expert system, the knowledge base stands out as a critical component. It would not be any less so for the stationkeeping task. The information required to perform the stationkeeping task and the transfer into rules for the knowledge base were briefly described. Interactions between the data base and knowledge base are extensive and only the basic functions were presented. From that presentation it should have been clear that AI technology could support the development of an expert system for spacecraft stationkeeping.

It was necessary to then take a hard look at expert system cost and the value of autonomy which is gained by deployment of the system. Autonomy was explained as a factor which can increase survivability of space resources. Current satellite survivability is limited by the vulnerable ground control link. Dependence on that link could be reduced by autonomous expert systems on board satellites. The cost of such systems was then presented using cost estimations and other government studies which examined ways to reduce satellite dependence on ground systems. The final result: AI technology is available to build a cost effective expert system to perform spacecraft stationkeeping.

The expert system was not only cost effective as an aid to the current ground system, but also as an integral part of an orbiting spacecraft. Satellite autonomy is increased as a result of an internal navigation expert system at very little or no cost. If satellite autonomy is deemed not necessary in future space system planning (unlikely, but possible), then AI can still contribute to effective ground control operations by assisting human orbital analysts with their current tasks. The development of expert systems is ripe for spacecraft exploitation. The question of AI use in space should turn from: Can we use AI is space to how best can we use AI in space?

There are still some technical issues such as size and weight that should be addressed. Can an expert system design be implemented on a small enough package to be placed on board the satellite? Research in this area supports reduction of size and weight as a normal "next-step." LISP machines are being produced with more capability and in smaller packages. This is an area that needs more study, but with the advent of VHSIC and increased competition, it should only be a short time before a reasonably sized LISP machine can make its way into space. If development began today on an expert system for orbit maintenance, it would still be 5-6 years before it would be ready for space use due to the need for demonstration, refinement, and validation. By that time the necessary hardware will surely be available.

RECOMMENDATIONS

Given that space assets are only going to increase in importance, it is essential that they be made more survivable. One aspect of that survivability is increased autonomy. Autonomy can be achieved through the use of Artificial Intelligence techniques and the development of an expert system to perform orbit maintenance. The Air Force needs to pursue this area of technology immediately. The Space Technology Center has begun to make efforts in this direction with the Satellite Autonomy Program, but more emphasis should be placed on the use of Artificial Intelligence.

One area of emphasis might be to determine accurate cost estimates for LISP machines and AI programs which are space qualified. Another might be to build an expert system to assist orbital analysts on the ground, similar to NASA's NAVEX, and let the system slowly work its way into space as technology becomes available. Certainly follow-on thesis work could be done to build an expert system program to demonstrate orbit correction capabilities. Once the process of development begins for an expert system, enthusiasm and realization of its capabilities will carry it to even greater performance.

APPENDIX A

LEVELS OF AUTONOMY

(Reproduced directly from Reference 10)

In performance of a space mission, four major policy goal categories have been identified. These are:

(1) Ground interaction reduction.

(2) Spacecraft integrity maintenance.

(3) Autonomous features transparency.

(4) On-board resource management.

The extent to which these goals have been accomplished to date has been through a mix of functions resident in either the space segment or the ground segment. Furthermore, the ground segment, as an integral part of the total system, has been responsible for accomplishing maintenance, navigation mission control, and payload data processing. Thus, only minimal spacecraft autonomy has been needed.

The levels of autonomy described in this appendix are used to define a step-wise increase in spacecraft autonomous capability. By proceeding through the levels, autonomous capability is increased in the space segment and dependency on the ground segment is reduced.

The levels of autonomy are described as follows:

Level 0. A design without redundant elements which meets all mission needs by operating without the on-board control of state parameters (such as rates and position). May respond to a prespecified vocabulary of external commands, but cannot store command sequences for future time-or event-dependent execution or validate external commands. (An open-loop, on-board system controlled from the ground.)

Level 1. Includes Level 0 but uses on-board devices to sense and control state parameters (such as rates and positions) in order to meet performance needs. Is capable of storing and executing a prespecified command sequence based on mission-critical time tags. Will respond to prespecified external commands, but cannot validate external commands. Functionally redundant modes may be available for a degraded-performance mission.

Level 2. Include Level 1 plus the use of block redundancy. Ground-controlled switching of spare resources is required. Uses cross-strapping techniques to minimize effect of critical command link (uplink) failure modes. Significant ground-operator interaction is required to restore operations after most faults if spare spacecraft resources are available.

Requires operator interaction for fault recovery. Is capable of storing and executing mission-critical events which are sensed on-board and may be independent of time.

Level 3. Includes Level 2 and is capable of sensing prespecified mission-critical fault conditions and performing predefined self-preserving (entering a safe-hold state) switching actions. Is capable of storing contingency or redundant software programs and being restored to normal performance (maintaining the command link with a single link fault) in the event of a failure. Timers may be used to protect resources. Requires ground operator interaction for fault recovery. In general, the failure to sense and/or execute the mission-critical event(s) will cause mission failure or loss of a major mission objective.

Level 4. Includes Level 3 but is also capable of executing prespecified and stored command sequences based on timing and/or sensing of mission events. Ground-initiated changes to command sequences may be checked on-board for syntactical errors (parity, sign, logic, time). Uses coding or other self-checking techniques to minimize the effects of internally generated data contamination for prespecified data transfers. Requires ground-operator interaction for fault recovery. In general, failure to sense and/or execute the mission event(s) or state-changes (excluding failure-induced state-changes) will cause mission failure or loss of a major mission objective.

Level 5. Includes Level 4 and is also autonomously fault-tolerant. Is capable of operating in the presence of faults specified a-priori by employing spare system resources, if available, or will maximize mission performance based upon available capability and/or available expendables (i.e., self-loading of contingency programs) without ground intervention.

Level 6. Includes Level 5 and is capable of functional commanding with on-board command-sequence generation and validation prior to execution. Functional commanding may include a high-level, pseudo-English language, spacecraft-system/operator communication and control capability.

Level 7. Includes Level 6 and is capable of autonomously responding to a changing external environment, defined a-priori, so as to preserve mission capability. The capability to change orbit in order to compensate for degradation or to protect the satellite from an external threat is included.

Level 8. Includes Level 7 and is capable of operating successfully within the presence of latent design errors which could cause loss of major mission objectives.

Level 9. Includes Level 8 and is capable of task deduction and internal reorganization based upon anticipated changes in the external environment. This situation is exemplified by multiple satellites operating in a cooperative mode. In the event of a satellite failure, remaining satellites would detect autonomously the condition (task deduction) and may generate and execute orbit-and spacecraft-reconfiguration commands.

<u>Level 10.</u> Includes Level 9 and is capable of internal reorganization and dynamic task deduction based on unspecified and unknown/unanticipated changes in external environment. The system will strive to maximize system utility. Thus, mission objectives should be adaptive and automatically reprogrammable. System resources should be maximized to preserve task adaptiveness.

APPENDIX B

SYSTEM: DENDRAL
INSTITUTION: Standford University
AUTHORS: Feigenbaum & Lederberg
FUNCTION: Data Interpretation

Characteristics of Example Expert Systems

		KEY ELEMENTS OF		
PURPOSE	APPROACH	KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
Generate plausible structural representations of organic molcules from mass spectrogram data	1. Derive constraints from the data. 2. Generate candidate structures 3. Predict mass spectrographs for candidates 4. Compare with data	Rules for deriving constraints on molecular structure from experimental data Procedure for generating candidate structures to satisfy constraints Rules for predicting spectrographs from structures	Mass spectrogram data Constraints Candidate structures	Forward chaining Plan, generate and test.

SYSTEM: AM

INSTITUTION: Stanford University

AUTHORS: Lenat

FUNCTION: Concept Formation

Characteristics of Example Expert Systems

		KEY ELEMENTS OF		
PURPOSE	APPROACH	KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
Discovery of mathematical concepts	Start with elementary ideas in set theory. Search a space of possible conjectures that can be generated from these elementary ideas. Choose the most interesting conjectures and pursue that line of reasoning.	Elementary ideas in finite set theory. Heuristics for generating new mathematical concepts by combining elementary ideas. Heuristics of "interestingness" for discarding bad ideas.	Plausible candidate concepts.	Plan, generate, and test.

SYSTEM: MYCIN
INSTITUTION: Stanford University
AUTHORS: Shortliffe
FUNCTION: Diagnosis

Characteristics of Example Expert Systems

		KEY ELEMENTS OF		
PURPOSE	APPROACH	KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
Diagnosis of bacterial infections and recommendations for antibiotic therapy.	Represent expert judgmental reasoning as condition-conclusion rules together with the expert's "certainty" estimate for each rule.	Rules linking patient data to infection hypotheses.	Patient history and diagnostic tests.	Backward chaining thru the rules.
	Chain backwards from hypothesized diagnoses to see if the evidence supports it.	Rules for combining certainty factors.	Current hypothesis	Exhaustive search.
		Rules for treatment.	Status.	
	Exhaustively evaluate all hypotheses. Match treatments to all diagnoses which have high certainty values.		Conclusions reached thus far, and rule numbers justifying them.	

SYSTEM: R1

INSTITUTION: CMU

AUTHORS: McDermott

FUNCTION: Design

Characteristics of Example Expert Systems

		KEY ELEMENTS OF		
PURPOSE	APPROACH	KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
Configure VAX computer systems (from a customer's order of components).	Break problem up into the following ordered subtasks: 1. Correct mistakes in order. 2. Put components into CPU cabinets. 3. Put boxes into unibus cabinets and put components in boxes. 4. Put panels in unibus cabinets. 5. Lay out system on floor. 6. Do the tabling. Solve each subtask and move on to the next one in the fixed order.	Properties of (roughly 400) VAX components. Rules for determining when to move to next subtask based on system state. Rules for carrying out subtasks (to extend partial configuration). (Approximately 800 rules total)	Customer order. Current task. Partial configuration (System state).	"MATCH" (data driven) (no backtracking)

SYSTEM: EL
INSTITUTION: MIT
AUTHORS: Stallman & Sussman
FUNCTION: Analysis

Characteristics of Example Expert Systems

PURPOSE	APPROACH	KEY ELEMENTS OF KNOWLEDGE BASE	 GLOBAL DATA BASE	 CONTROL STRUCTURE
Steady State analysis of resistor-diode-transistor circuits to determine voltages and currents.	Use assertions in the data base to trigger the rules which create new assertions in the data base. Assume operating states for transistors and diodes. Introduce variables for the parameters (e.g., ė for voltage) at one node in the circuit and use electrical laws to symbolically compute parameters at other nodes. Make conjectures when no further rules are applicable. Observe contradictions and revise conjectures and conclusions dependent upon them.	Rules that represent general electrical principles. Rules for making conjectures. Rules for deciding what to forget when contradictions occur.	Facts about the particular circuit being analysed represented as assertions in an associative data base. History of conjectures used and conclusions dependent upon them. Problem status.	Forward reasoning Guesses when needed Relevant backtracking Priority-oriented queue-based control.

SYSTEM: KAS
INSTITUTION: SRI
AUTHORS: Reboh
FUNCTION: Knowledge Acquisition

Characteristics of Example Expert Systems

PURPOSE	APPROACH	KEY ELEMENTS OF KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
Supervises interaction with an expert in building or augmenting a expert system knowledge base in a network form. Implemented for PROSPECTOR.	Take existing PROSPECTOR knowledge base and knowledge of mechanisms in PROSPECTOR, and add networks for expressing new knowledge and rules for interaction with a human domain expert. The main technique is to consider the KAS system as a general purpose editor, given knowledge of the specific network.	Prospector knowledge base. Inference networks for expressing judgmental knowledge. Semantic networks for expressing the meaning of the propositions employed in the rules. Taxonomic networks for representing basic knowledge among the terms in the domain. Knowledge of various mechanisms employed in PROSPECTOR for representing and using knowledge. Consistency mechanisms.	Partial candiates to be completed.	Network editor -creates, modifies or deletes various nodes and arcs in the networks. Grammar-driven command language to prompt user in the proper use of command language.

SYSTEM: GUIDON

INSTITUTION: Stanford University

AUTHORS: Clancy

FUNCTION: Computer-Aided Instruction (CAI)

Characteristics of Example Expert Systems

		KEY ELEMENTS OF		
PURPOSE	APPROACH	KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
Teach facts and problem-solving strategies for: - diagnosing and treating meningitis and bacterima or - pulmonary function analysis	Use MYCIN or PUFF knowledge base and add rules for teaching medical diagnosis	MYCIN or PUFF knowledge base 200 additional rules for: - guiding dialog with student - presenting medical diagnostic strategies - constructing a studen model - responding to the students initiations	Student model thus far Current interchange Status Relevant history of interchange	Event driven

SYSTEM: VM (Ventilator Manager)
INSTITUTION: Stanford University
AUTHORS: Fagan
FUNCTION: Monitoring and interpreting real-time data

Characteristics of Example Expert Systems

		KEY ELEMENTS OF		
PURPOSE	APPROACH	KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
Interpret the clinical significance of data from a physiological monitoring system of patient breathing.	1. Start with an initial patient state (context or situation) 2. Use initialization rules to determine expectations and unacceptable measurement limits for that state. 3. Run status rules to derive physiological states for use in therapy. 4. Run transition rules to see if state has changed each time new set of periodic measurements arrive. 5. Repeat 2 and 3 for each new state.	Transition rules. Initialization rules. Status rules Therapy rules.	Sets of periodic measurements of patient. Patient current state (context or situation) Expectations and unacceptable limits for measurements in current state. Recent patient history during monitoring. Physiological status.	Event driven. Exhaustive search for each state. State triggered expectations.

SYSTEM: META-DENDRAL
INSTITUTION: Stanford University
AUTHORS: Buchanan and Feigenbaum
FUNCTION: Learning from Experience

Characteristics of Example Expert Systems

		KEY ELEMENTS OF		
PURPOSE	APPROACH	KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
To generate a set of general fragmentation rules, of the form used by DENDRAL, given sets of known structure-spectrum pairs.	Generate, test and refine a set of candidate rules from known molecule structure-spectrum pairs.	Rules for: -interpreting spectral data and summarizing results. -generating candidate fragmentation rules from the evidence. -*generalizing or* specializing candidate rules to better fit the evidence.	*User supplied context* Input data -known structure-*spectrum pains* -plausible candidate fragmentation rules	Plan, generate, and test.

SYSTEM: ABSTRIPS

INSTITUTION: SRI

AUTHORS: Sacerdoti

FUNCTION: Planning

Characteristics of Example Expert Systems

		KEY ELEMENTS OF		
PURPOSE	APPROACH	KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
Devises plans for a robot to move objects between rooms.	Do hierarchical planning by first devising a top level plan based on the key aspects of the problem, then successively refining it by considering less critical aspects of the problem. Recipe: 1. Fix abstraction levels for solutions (plans). 2. Problem solution proceeds top down (most abstract to most specific). 3. Complete solution at one level and then move to next level below.	Criticality assignments of elements in robot planning domain. Configuration of the rooms. Objects and their properties in the domain. Rules for decrementing criticality level. Heuristic search rules for each level.	Goal Initial state of system (criticality at maximum). Plans thus far. Current criticality level	Goal directed (backward chaining at each level). Top down refinement of plans using hierarchical abstract search spaces.

SYSTEM: GA1

INSTITUTION: Stanford U.

AUTHORS: Stefik

FUNCTION: Data Interpretation

Characteristics of Example Expert Systems

PURPOSE	APPROACH	KEY ELEMENTS OF		
		KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
Infer a complete molecular structure from measurement of molecular pieces (resulting from digestion by an enzyme).	1. Refine the data. 2. Determine an initial set of generator constraints from data. 3. Combine data and constraints to generate candidate structures. 4. Prune candidate approaches during generation process that are inconsistant with general rules for molecular structures. 5. Test candidates to see if they satisfy data.	Data correction rules. Rules for determining an initial set of generator constraints. Generation rules. Rules for pruning inconsistant candidate classes. Rules for testing candidate molecular structures.	Measurement data. Corrected data. Derived set of generator constraints Partial solutions. Candidate molecular structures.	Hierarchical generate and test. Early pruning of inconsistant approaches.

SYSTEM: MOLGEN
INSTITUTION: Stanford U.
AUTHORS: Stefik*
FUNCTION: Design

Characteristics of Example Expert Systems

*Another Molgen version by Friedland under development.

PURPOSE	APPROACH	KEY ELEMENTS OF		
		KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
Designing molecular genetic experiments	Represent interactions between subproblems as constraints. Formulate constraints as goals to be solved. Use constraint propogation to reveal interactions between subproblems. Suspend problem-solving as necessary, until sufficient information is derived from the interchange of constraints (least committment, opportunistic expansion). Use heuristic guessing to make choices when there is otherwise no compelling reason to do so. Retract guesses as necessary when an unresolvable problem is encountered.	Explicit meta-level problem-solving operators to reason with constraints. Problem-solving rules. Rules for guessing. Rules for discovering interactions between subproblems via constraint propogration.	Partial solutions. History of guesses and their affects. Constraints.	Constraint propogation Least committment. Heuristic guessing. Relevant backtracking. Use of meta-rules to reason with constraints. Hierarchical refinement. Difference reduction.

SYSTEM: NOAH
INSTITUTION: SRI
AUTHORS: Sacerdoti
FUNCTION: Planning

Characteristics of Example Expert Systems

		KEY ELEMENTS OF		
PURPOSE	APPROACH	KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
Robot Planning System (assigns a time-ordering to operators in a plan.)	Expand, in parallel, individual plans for interacting subgoals, but initially assign only a partial time-ordering to operators. Stop when interference between the partial subgoal plans is observed, and adjust the ordering of the operators as needed to resolve the interference.	Operators. Rules for recognizing interference between plans. Rules for resolving interferences.	Subgoals Partial ordering of operators in subgoal plans. Interference between plans.	Least committment. Backward chaining.

SYSTEM: SYN

INSTITUTION: MIT

AUTHORS: Sussman, Steele, & Dekleer

FUNCTION: Design

Characteristics of Example Expert Systems

		KEY ELEMENTS OF		
PURPOSE	APPROACH	KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
Circuit synthesis	Use propogation ideas of EL.	Electrical laws.	Circuit.	Forward reasoning.
-determines values for components in electrical circuits.	Switch to equivalent representations of circuit portions when needed to overcome blockages in the propogation of constraints. (These "slices" - multiple views of circuit portions - provide redundant paths for information to travel. Slices combine the strengths of multiple models).	Rules for changing slices	Slice being considered	Changing representations as needed to continue analysis.
		Rules for creating appropriate slices.	Set of deduced values.	

SYSTEM: HEARSAY II

INSTITUTION: CMU

AUTHORS: Erman, Hayes-Roth, Lesser, Reddy

FUNCTION: Signal Interpretation

Characteristics of Example Expert Systems

		KEY ELEMENTS OF		
PURPOSE	APPROACH	KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
Speech understanding	Break the problem up hierarchically into levels (heterogeneous abstract search spaces) with sentences at the top and signal measurement parameters at the bottom. Do both bottom up and top-down processing in a relaxation approach to extend and combine partial candidates. Carry several candiates at each level (parallel lines of reasoning) to keep from being too focussed and missing the correct interpretation. Use a separate knowledge source (KS) for each level. Use a 7 level "blackboard" to display hypotheses. Have KS's communicate via the blackboard. Have KS's when activated: 1. create and extend hypotheses on blackboard, 2. record the evidential support between levels, 3. assign credibility level Use "opportunistic scheduling" of computational resources for changing the breadth of search depending on conditions of uncertainty resulting from interaction of KS assigned credibility ratings and scheduler-assigned priorities of pending KS activations.	Language knowledge. KSs for creating: 1. labeled segments for signal parameter measurements, 2. syllable hypotheses from segments, 3. word hypotheses from syllables, 4. word sequence hypotheses, 5. phrase hypotheses, 6. predictions of words following phrases, 7. sentence level interpretations for the infomation retrieval system. KSs for controlling - number of word hypotheses, - number of word-sequence hypotheses. KSs for rating - credibility of hypotheses, - consistency between segment hypotheses and word-phrase pairs	Hypotheses on 7 level blackboard. Record of evidential support between levels Credibility levels of hypotheses. Results thus far. Agenda queue of pending KS activations. Top-down vs bottom-up consistencies between word-phrase pairs and segment hypotheses.	Combination of top-down and bottom up processing. Opportunistic scheduling. Least committment. Variable-width search.

SYSTEM: HARPY
INSTITUTION: CMU
AUTHORS: Lowerre
FUNCTION: Signal interpretation

Characteristics of Example Expert Systems

		KEY ELEMENTS OF		
PURPOSE	APPROACH	KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
Speech understanding.	Represent the set of all possible utterances in HARPY's domain by production rules which relate signal syllables to words. Add juncture rules at word boundaries. Use a compiler to combine the syntax, lexical and juncture knowledge into a single large transition network in which each path from a start node to an end node represents a sequence of segments for some sentence.	Transition network.	Input speech.	Data driven "Beam Search" thru network.

SYSTEM: CRYSALIS
INSTITUTION: Stanford University
AUTHORS: Englemore & Terry
FUNCTION: Data Interpretation

Characteristics of Example Expert Systems

		KEY ELEMENTS OF		
PURPOSE	APPROACH	KNOWLEDGE BASE	GLOBAL DATA BASE	CONTROL STRUCTURE
Automatic interpretation of protein electron-density maps.	1. Use a skeletonization algorithm to convert the electron density map to a line-skeleton representation. 2. Use a algorithm to partition skeleton graph into chemical side chains and backbone elements. 3. Use rules to hypothesize (with associated confidence levels) and test atoms and super atoms based on the partioned skeleton graph and the known chemical model of the protein under study. 4. Use hierarchical meta-rules to select appropriate KS rule set. Recipe to develop hypotheses 1. Use hypothesis toeholds to fire strategy rules to determine which task rule set to consider next. 2. Match task rules against the "type" column of the event list to focus on one event and to choose the set of KS rules to consider. 3. Match the hypothesis state and the "where" portion of the event to the KS rules to increment the hypothesis and add to the event list. 4. Repeatedly cycle on 2 and 3 until no further matches occur, then return control to strategy-rule interpreter.	Strategy rule set. Task rule set. KS rule set. Skeletonization algorithm. Partitioning algorithm.	"Blackboard" which includes: -a hierarchically orgainized hypothesis data structure -support for the current hypothesis from the partioned skeleton graph and the chemical model. State of the hypothesis. -"toehold" opportunities for accelerated development of the hypothesis. Event list of recent changes by type and location.	Event driven. Hypothesize and test. Hierarchical rule interpreters -strategy -task -KS

LANDSAT-5 O.A. NO. 09
(1 OF 5)

APPENDIX C
ORBIT ADJUST REQUEST
(OCG TO OCC)

TO: BRUCE SHAPIRO X 0146 TIME: 840425 160000 (Z)

FROM: DICK STRAFELLA X 5049 TIME: 840425 160000 (Z)

REQUESTED O.A. TIME: 840430 211400 (Z)

REQUESTED STATION: MADRID

ORBIT ADJUST TYPE:
☑ ORBIT MAINTENANCE
☐ INCLINATION
☐ OTHER ______

ORBIT ADJUST MODE:
☐ PRIMARY
☑ BACKUP

REQUESTED BACKUP STATIONS: NORTHERN / ASCENDING PASSES

COMMENTS:

(2 OF 5)

ORBIT ADJUST PREPLAN
(OCC TO OCG)

TO: DICK STRAFELLA X 5049 TIME: 840425 160000 (Z)

FROM: BRUCE SHAPIRO X 0146 TIME: 840425 160000 (Z)

SUGGESTED STATION CONTACTS:

	STATION	REV	AOS	DURATION	MAX EL
a.	MADRID	877	210858	1250	67
b.	MADRID	878	224842	0926	12
c.					

FUEL TANK STATUS:

DATE 840424 / 115

TIME 033342 - 051235 (Z)

REV 779

PRESSURE 271.23 PSIA

TEMPERATURE T1 16.40 °C

T2 16.03 °C

T3 17.33 °C

T1A 13.30 °C

ORBIT AVERAGE

HYDRAZINE 496.57 LBS

TRANSLATION THRUSTERS: ☐ B, D ☑ A, C ☐ A, B, C, D

ATTITUDE THRUSTERS: ☑ B, D ☐ A, C

YAW MANEUVER: ☑ NOT PLANNED ☐ PLANNED ______ DEG

PITCH MANEUVER: ☑ NOT PLANNED ☐ PLANNED ______ DEG

SOLAR ARRAY: ☑ AT NOMINAL RATE ☐ AT 278° ☐ OTHER ____ DEG

COMMENTS:

(3 OF 5)

ORBIT ADJUST PLAN
(OCG TO OCC)

TO: BRUCE SHAPIRO X 0146 TIME: 840426 160000 (Z)

FROM: DICK STRAFELLA X 5049 TIME: 840426 160000 (Z)

DATE 840430/121

REV 877

STATION MADRID

NOMINAL START TIME 211400 (Z) LATEST START TIME 211600 (Z)

BURN DURATION (TOTAL) 7680 TOTAL THRUSTER MSEC } DO NOT EXCEED

BURN DURATION (PER THRUSTER) 3840 MSEC

TRANSLATION THRUSTERS: ☐ B, D ☑ A, C ☐ A, B, C, D

ATTITUDE THRUSTERS: ☑ B, D ☐ A, C

YAW MANEUVER: ☑ NOT PLANNED ☐ PLANNED ______ DEG

PITCH MANEUVER: ☑ NOT PLANNED ☐ PLANNED ______ DEG

SOLAR ARRAY: ☑ AT NOMINAL RATE ☐ AT 278° ☐ OTHER ______ DEG

ESTIMATES:

a. BURN START LATITUDE 34.9 °N

LONGITUDE 0.6 °E

b. SEMIMAJOR AXIS CHANGE +0.159 KM

c. INCLINATION CHANGE +0.00 DEG

d. ORBITAL PERIOD CHANGE +0.2 SEC

e. FUEL USAGE -0.16 LBS

f. PRESSURE CHANGE -0.2 PSIA

COMMENTS:

LANDSAT-5 O.A. NO. 07
(4 OF 5)

ORBIT ADJUST POSTBURN REPORT
(OCC TO OCG)

TO: DICK STRAFELLA X 5049 TIME: 840502 130000 (Z)

FROM: BRUCE SHAPIRO X 0146 TIME: 840502 130000 (Z)

	PLANNED		ACTUAL	
DATE	840430 / 121		840430 / 121	
REV	877		877	
STATION	MADRID		MADRID	
BURN START TIME	211400	(Z)	211403	(Z)
BURN STOP TIME	211404	(Z)	211407	(Z)

	PLANNED	ACTUAL
TRANSLATION THRUSTERS	☐ B, D	☐ B, D
	☑ A, C	☑ A, C
	☐ A, B, C, D	☐ A, B, C, D
ATTITUDE THRUSTERS	☑ B, D	☑ B, D
	☐ A, C	☐ A, C

	PLANNED		ACTUAL		
YAW MANEUVER	0	DEG	0	DEG	
PITCH MANEUVER	0	DEG	0	DEG	
SOLAR ARRAY	N.R.	DEG	N.R.	DEG	
TOTAL THRUSTER DURATION	7680	MSEC	7424	MSEC	LOADED
TIMER DURATION			6144	MSEC	

BURN TERMINATION ☑ COUNTER ☐ TIMER

TRANSLATION THRUSTERS

A	3840	MSEC
B	0	MSEC
C	3840	MSEC
D	0	MSEC
TOTAL	7680	MSEC

ATTITUDE THRUSTERS

B2	0	COUNTS
B3	15	COUNTS
B4	0	COUNTS
D2	15	COUNTS
D3	0	COUNTS
D1	0	COUNTS
TOTAL	30	COUNTS

(1 COUNT = 280 MSEC)

FUEL TANK STATUS:

DATE	840430 / 121		840501 / 122	
TIME	160739 - 174632	(Z)	065737 - 083631	(Z)
REV	874		883	
PRESSURE	271.05	PSIA	270.90	PSIA
TEMPERATURE				
T1	16.40	°C	16.40	°C
T2	16.03	°C	16.03	°C
T3	17.32	°C	17.30	°C
T1A	13.17	°C	13.06	°C
HYDRAZINE	496.52	LBS	496.48	LBS

ATTITUDE DATA APPENDED ☐ YES ☑ NO

COMMENTS:

(5 OF 5)

ORBIT ADJUST POSTBURN ANALYSIS
(OCG TO OCC)

TO: Bruce Shapiro X 0146 TIME: 840504 210000 (Z)

FROM: Dick Strafella X 5049 TIME: 840504 210000 (Z)

	PLANNED	REPLANNED	ACTUAL
DATE	840430 / 121	840430 / 121	840430 / 121
REV	877	877	877
BURN TIME	211400 (Z)	211403 (Z)	211403 (Z)
TRANSLATION THRUSTERS	☐ B, D	☐ B, D	☐ B, D
	☑ A, C	☑ A, C	☑ A, C
	☐ A, B, C, D	☐ A, B, C, D	☐ A, B, C, D
ATTITUDE THRUSTERS	☑ B, D	☑ B, D	☑ B, D
	☐ A, C	☐ A, C	☐ A, C
YAW MANEUVER	0 DEG	0 DEG	0 DEG
PITCH MANEUVER	0 DEG	0 DEG	0 DEG
TOTAL TRANS. DURATION	7680 MSEC	7680 MSEC	7680 MSEC
TOTAL ATT. DURATION	0 MSEC	0 MSEC	8400 MSEC
FUEL TANK STATUS			
PRESSURE	271.23 PSIA	271.05 PSIA	271.05 PSIA
TEMPERATURE			
T1	16.40 °C	16.40 °C	16.40 °C
T2	16.03 °C	16.03 °C	16.03 °C
T3	17.33 °C	17.32 °C	17.32 °C
T1A	13.30 °C	13.17 °C	13.17 °C
HYDRAZINE			
TRANSLATION	-0.16 LBS	-0.16 LBS	-0.16 LBS
ATTITUDE	0 LBS	0 LBS	-0.01 LBS
TOTAL	-0.16 LBS	-0.16 LBS	-0.17 LBS
REMAINING	497.77 LBS	497.77 LBS	497.77 LBS
SEMIMAJOR AXIS CHANGE	+0.159 KM	+0.159 KM	+0.157 KM
INCLINATION CHANGE	+0.00 DEG	+0.00 DEG	+0.00 DEG
ORBITAL PERIOD CHANGE	+0.2 SEC	+0.2 SEC	+0.2 SEC
TRANSLATION THRUSTER EFFICIENCY	0.9700	0.9700	0.9532

PREDICTED TIME OF NEXT ORBIT ADJUST 840619

COMMENTS: WRS ERROR WAS +9.9 Km.

Bibliography

1. AFSCN-OC-002. Air Force Satellite Control Network Communications Operations Concept, Revised Draft, HQ Air Force Satellite Control Facility, Sunnyvale AFS, CA, June 22, 1983. (For Official Use Only)

2. Air Force Satellite Control Facility. "Air Force Satellite Control Facility Data System Modernization Program," Briefing Slides. Undated.

3.Bachant, Judith and John McDermott. "R1 Revisited: Four Years in the Trenches," The AI Magazine, Fall 1984, pp. 21-32.

4. Barr, Avron and Edward Feigenbaum, editors. The Handbook of Artificial Intelligence, Vol II. Los Altos, CA: William Kaufman, Inc., 1982.

5. Brodsky, Robert F. and Bernard G.Morais. "Space 2020: The Technology, The Missions Likely 20-40 Years From Now," Astronautics and Aeronautics. 20(4):54-73 (May 1982).

6. Davis, Randall. "Expert Systems: Where Are We? And Where Do We Go From Here?" The AI Magazine, Spring 1982.

7. Davis, Randall and Jonathan King. "An Overview of Production Systems." In E. Elcock and D. Michie (eds.), Machine Interface. Chichester, England: Ellis Horwood, 300-332.

8. Duda, R.O. "Knowledge-Based Expert Systems Come of Age," BYTE, Vol. 6, No. 9, September 1981, pp. 238-281.

9. Federal Systems Division. Data System Modernization, Command and Control Segment, Stage II, System Operations Concept, International Business Machines Corp., Gaithersburg MD, Contract No. F04690-81-C-0003, DSM Control No. 7942367, Document No. 040-004, 31 March 1983.

10. Gevarter, William. "An Overview of Expert Systems," U.S. Department of Commerce, Government Printing, Washington, D.C., October 1982.

11. Gjermundsen, Edward, Captain, USAF. Air Force Space Technology Center/YLC, Kirtland AFB, New Mexico. Telecon interview and "Satellite Autonomy Briefing."

12. Hayes-Roth, Frederick, Donald Waterman, and Douglas Lenat, editors. Building Expert Systems. Reading, MA: Addison-Wesley Publishing Co., 1983.

13. Hendrix, Gary and Earl Sacerdoti. "Natural Language Processing, The Field in Perspective," BYTE, September 1981.

14. Jackson, Lt Col, USAF. Air Force Space Division/VO, Sunnyvale AFS, CA, October 1984. Telecon.

15. Kinnucan, Paul. "Computers That Think Like Experts," High Technology, January 1984.

16. Marsh, Alton K. "NASA to Demonstrate Artificial Intelligence in Flight Operations," Aviation Week and Space Technology, September 17, 1984, p. 79. Interview with Robert Savely, NASA Johnson Space Center.

17. Marshall, Michael. "Goals for Air Force Autonomous Spacecraft," Report No. SD-TR-81-72. Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, 31 March 1981.

18. Newell, Alan and Herbert Simon. Human Problem Solving, Prentice-Hall Incorp.: Englewood Cliffs, NJ, 1972.

19. Nilsson, Nils J. Principles of Artificial Intelligence. Palo Alto, CA: Tiogo Publishing Co., 1980.

20. O & M Training Section MS-60. SCF Telemetry, Tracking and Command System Orientation, Course No. 116, Ford Aerospace and Communications Corp., Space Missions and Engineering Services Operations, Sunnyvale, CA, Contract No. F04690-81-C-004, January 1983.

21. Picciano, Wayne and Michael Humnicky. "Satellite Control System (SCS) Final Report Volume I; Functional and Performance Requirements," Report No. SD/TR 8025, AD B089 5156. Stanford Telecommunications, Inc., Sunnyvale, CA, Contract No. F04701-79-C-0045, 15 December 1980.

22. Pivirotto, Donna and Michael Marcucci. "Assessment of Autonomous Options for the DSCS III Satellite System." Report No. TR 81 87, AD A106 062, 063, 064. Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, Contract No. NAS7 100, 6 August 1981.

23. Smith, Sarah. The LMI Lambda: Technical Summary, Los Angeles, CA, 1984.

24. "Spacecraft Survivability Boost Sought," Aviation Week and Space Technology, 118(24): 46-47 (16 June 1980).

25. Stanford Telecommications, Inc. "SCS Transition Plan, Vol IX," Report no. SD/TR 8025 for HQ Air Force Space Division, contract F04701-79-C-0045, Febuary 1980.

26. Strafella, Richard and William Weston. Personal interview at Goddard Space Flight Test Center, Greenbelt, Maryland, September 1984.

27. Theis, Douglas. "Spacecraft Computers: State of the Art Survey," Computer, IEEE Series, April 1983.

VITA

Captain Michael A. Wright was born on 10 March 1956 in Pontiac, Michigan. He graduated from Avondale Senior High, Auburn Heights, MI in 1974. He attended the United States Air Force Academy and graduated with a Bachelor of Science in Political Science/International Affairs in 1978. After graduation, he went to technical training for Signal Intelligence Officers and arrived at his first duty assignment, the 6993rd Electronic Security Squadron (ESS), Kelly AFB, Texas in January 1979. He served as Squadron Section Commander and Flight Commander before moving to the 6924th ESS, Wheeler AFB, Hawaii in October 1980. Captain Wright was a Flight Commander at the 6924th until entering the School of Engineering, Air Force Institute of Technology, in May 1983.

Permanent address: 1215E Kirts
Troy, MI 48084

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE

3. DISTRIBUTION/AVAILABILITY OF REPORT
Approved for public release
distribution unlimited

4. PERFORMING ORGANIZATION REPORT NUMBER(S)
AFIT/GSO/OS/84D-9

5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION
School of Engineering

6b. OFFICE SYMBOL *(If applicable)*
AFIT/ENS

7a. NAME OF MONITORING ORGANIZATION

6c. ADDRESS *(City, State and ZIP Code)*
Air Force Institute of Technology
Wright-Patterson AFB, OH 45433

7b. ADDRESS *(City, State and ZIP Code)*

8a. NAME OF FUNDING/SPONSORING ORGANIZATION
Space Technology Center

8b. OFFICE SYMBOL *(If applicable)*
AFSTC/XLC

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS *(City, State and ZIP Code)*
Air Force Space Technology Center
Kirtland AFB, NM 87117

10. SOURCE OF FUNDING NOS.

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT NO.

11. TITLE *(Include Security Classification)*
ARTIFICIAL INTELLIGENCE IN SPACE PLATFORMS

12. PERSONAL AUTHOR(S)
Wright, Michael A.

13a. TYPE OF REPORT
MS Thesis

13b. TIME COVERED
FROM ________ TO ________

14. DATE OF REPORT *(Yr., Mo., Day)*
1984 December

15. PAGE COUNT
137

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB. GR.
06	04	
[illegible]	05	

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
Artificial Intelligence, military satellites,
expert systems, satellite autonomy.

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*
on reverse

Approved for public release: IAW AFR 190-17.
LYNN E. WOLAVER 21 Feb 85
Dean for Research and Professional Development
Air Force Institute of Technology (ATC)
Wright-Patterson AFB OH 45433

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT
UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS ☐

21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL
Mark M. Mekaru, Lt Col, USAF

22b. TELEPHONE NUMBER *(Include Area Code)*
(513) 255-6623

22c. OFFICE SYMBOL
AFIT/ENS

DD FORM 1473, 83 APR

EDITION OF 1 JAN 73 IS OBSOLETE.

SECURITY CLASSIFICATION OF THIS PAGE

This study determined the feasibility of implementing Artificial Intelligence techniques on orbiting spacecraft. The main thrust was to evaluate the current technology of expert systems and determine their value to satellite tasking. The goal for an expert system to be effective was that it must be able to perform spacecraft stationkeeping without ground assistance.

Analysis began by outlining the basic functions of the DSCS III and noting deficiencies as measured against an "autonomy scale." Many of the defeciencies could be corrected with conventional computer programming, but stationkeeping required AI techniques for proper execution. Expert systems were then examined and studied for applicability to the primary task of orbit maintenance. R1, an expert system designed to perform computer configuration, was found to be a good baseline for comparison and further development. The process of orbit maintenance, as currently done by human experts, was explained and outlined for expert system design. Finally, a cost analysis provided information which supported further development of AI technology for spacecraft implementation.

End

5285