N89-10069

/61031/
P.13

An Expert System That Performs A Satellite Stationkeeping Maneuver

M. Kate Lines-Browning and John L. Stone, Jr.

May 13, 1987

PRECEDING PAGE BLANK NOT FILMED

An Expert System That Performs A Satellite Stationkeeping Maneuver

M. Kate Lines-Browning and John L. Stone, Jr.

Abstract:

In this paper, we describe the development and capabilities of a prototype expert system that provides real-time spacecraft system analysis and command generation. At present, ESSOC (Expert System for Satellite Orbit Control) is capable of performing the stationkeeping maneuver for a geostationary satellite.

ESSOC guides the operator through the stationkeeping operation by recommending appropriate commands that reflect both the changing spacecraft condition and previous procedural action. Information regarding satellite status is stored in a knowledge base internal to the expert system. This knowledge base is continuously updated with processed spacecraft telemetry. Information on the procedural structure is encoded in production rules. The independence of the procedural rules from each other, and from the knowledge base, makes the system easy to maintain and expand.

Particular attention is directed to distinctive features of the ESSOC system and its development, namely, the structured methods of knowledge acquisition, and the design and performance-enhancing techniques that enable ESSOC to operate in a real-time environment.

1.0 Introduction

Certain properties of current satellite operation techniques indicate that significant benefit may be derived by introducing automation into the field of satellite operations. First, satellites are difficult to operate, requiring skilled teams that are both difficult to assemble and expensive to maintain. Second, errors on the part of the flight crew can be expensive to rectify or can even be irreversible. Automating satellite operations offers a number of distinct advantages:

1) Swift anomaly detection and response;
2) Identification of transient conditions;
3) Correct operational response to the aforementioned conditions; and
4) Capability to implement increasingly complex flight rules.

As proof of the concept that the use of expert systems is an efficient method of achieving the goals listed above we have developed ESSOC, a prototype expert system for satellite operations.

Rather than construct a system to completely handle all satellite operations, we limited the scope of ESSOC operation to a subset of the operations for a mission. Furthermore, once a prototype system was produced, the modular design inherent in ESSOC would enable us to expand the system over time.

We selected the stationkeeping maneuver for the TDRS-1 spacecraft as the domain for our development effort. The choice of spacecraft was predicated upon the availability of knowledge engineers familiar with the domain. The choice of the particular satellite operation to be automated was more arbitrary, but the stationkeeping maneuver met the following desirable criteria:

1) Need for swift response to problems;
2) Reasonable procedure duration (approximately 3 hours);
3) Manageable domain size/development complexity;

4) Universality of application to different satellites;
5) Critical need for correct commanding;
6) Greatest potential benefit to current TDRS operations.

Conveniently, the TSIM real-time TDRS Simulator was available to serve in the test bed for ESSOC, providing both a telemetry stream and a command response.

Using the techniques outlined herein, ESSOC prepares the satellite (in this case the TSIM real-time simulator) for the orbit adjustment by recommending and sending commands that route propellant to the appropriate thrusters for attitude control and firing the delta V thrusters. Attitude control modes are switched as necessary and the success or failure of each step of the procedure is verified continuously via telemetry. The user is notified of problems as problems are detected. ESSOC generates satellite commands in response to anomalies and displays them for user action.

2.0 The TDRS Spacecraft

The TDRS spacecraft (shown in Figure 1) is a 3-axis-controlled, bias-momentum-stabilized communications satellite. Launched in 1983 and stationed at 71^0 West longitude, TDRS-1 will be joined by at least two similar spacecraft yet to be launched. During normal on orbit operations, reaction wheels are used to control the attitude, while one pound (nominal) thrusters are used occasionally to remove accumulated wheel momentum. These same

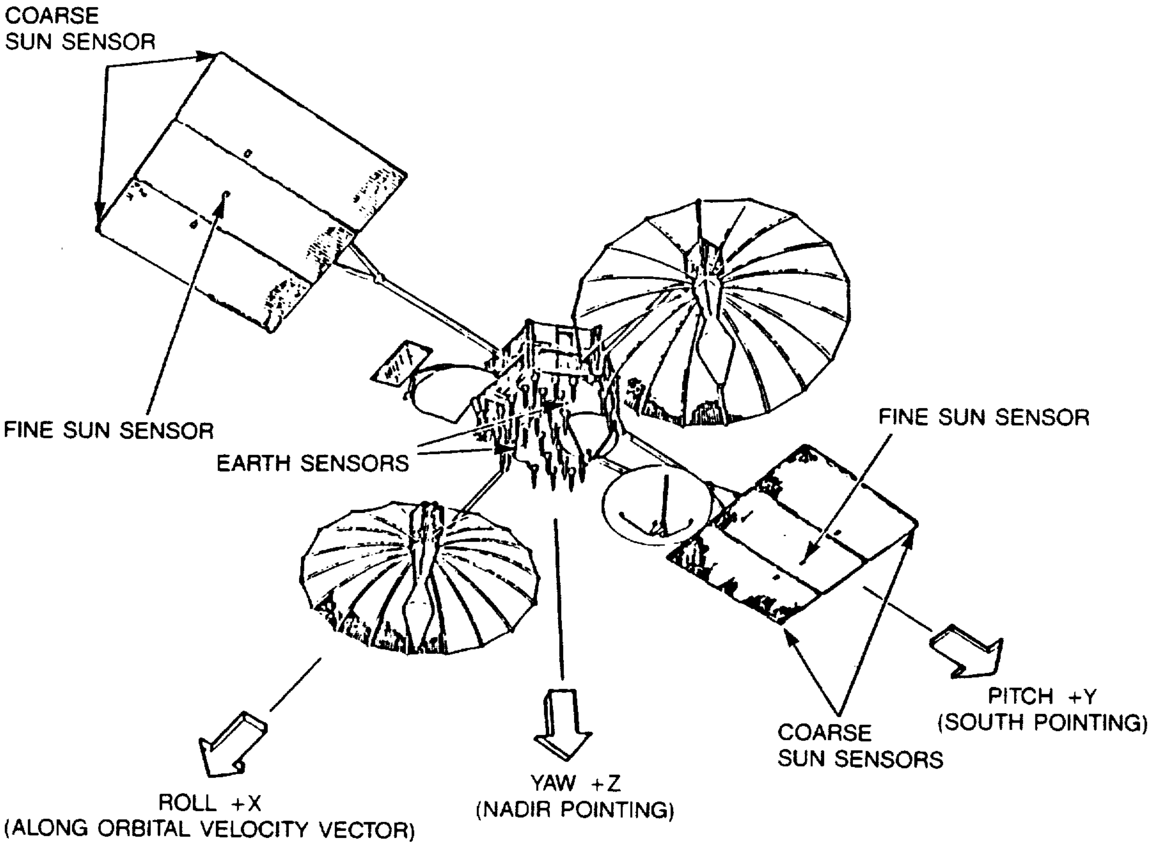


FIGURE 1. ON-ORBIT CONFIGURATION

thrusters are also used periodically to adjust the TDRS orbit to correct for perturbations, and to maintain the TDRS attitude and thrust vector during this correction procedure.

The thrusters use catalytic decomposition of hydrazine on a heated catalyst bed, which must have its temperature maintained through proper commanding. The earth sensors provide pointing error information provided the satellite is within five degrees of the proper attitude. The solar arrays provide electric power, and must be maintained in the correct, sun-pointing position by rotating them about the pitch axis as the satellite orbits. Various antennas included in the spacecraft payload may also be seen (Figure 1). Further details on the TDRS may be found in "TDRS Spacecraft Operations."[1]

3.0 The ESSOC Expert System

The ESSOC expert system resides on a Symbolics 3675 LISP machine. It receives processed telemetry data via Ethernet from the ESSOC front end processor which resides on a VAX 11/785. The telemetry data is provided by a real-time spacecraft simulator that also resides on the VAX. In this section, we will discuss each portion of this configuration separately. A discussion of the link between the two machines is found in section 3.3.5.

3.1 ESSOC Development Environment

The Symbolics 3675 LISP machine is a stand-alone, single-user LISP workstation. Our system is equipped with both a black-and-white console and a color graphics monitor. The programming environment supports multi-windowing, multi-tasking, incremental development of programs, and optimized LISP programming. We used LISP to implement the expert system functions that dealt with the color graphics, procedure timing, networking, and arithmetically intensive functions used by procedural rules. Most of the expert system, however, was developed using the expert system development shell, ART (Automated Reasoning Tool), which greatly expedited expert system development. ART provides an inference engine and mechanisms for representing frames (schemata), rules (backward and forward chaining), and inheritance relations.

Telemetry data for the expert system is provided by a real-time, high-fidelity simulator of the TDRS spacecraft (TSIM) that resides on the VAX 11/785. Because the simulator is able to model response to commanding in telemetry, the simulator provides a telemetry stream (1000 bits per second) functionally identical to that of the spacecraft. Hence, in designing the expert system, we were able to consider the simulator indistinguishable from the satellite.

ESSOC's front end processor, which resides on the VAX, is responsible for processing the raw telemetry data from the simulator into a specified format and placing it into a processed telemetry buffer on the VAX. The conversion of data is performed in two steps. First, the front end processor breaks the data from the simulator down into complete telemetry frames and stores these frames in a buffer on the VAX. Whenever the processed telemetry buffer becomes empty, the second step of the processing is performed. The second step of the conversion process changes this raw data into engineering units, performs trend determination, and labels the data values with ASCII tags. The front end processor places the resulting data into the processed telemetry buffer which holds up to 64 frames (32768 bits) of telemetry.[2]

3.2 Expert System Development

Using the ''rapid prototyping'' method of software development, a working prototype of the ESSOC expert system was developed within six months. The first three months of the project were spent in an intensive knowledge acquisition phase. The information collected at this time was used to select a scheme for representing knowledge in the expert system that would allow the system to operate in real time and to be expanded. After deciding on the general design of the system, the information gained from the domain experts (in this case, spacecraft engineers) was organized and converted into code. The expert system prototype generated from the initial data was evaluated by the domain expert and suggestions for improvements made by the spacecraft engineers were incorporated into the system. The development cycle then repeats: the spacecraft engineers are interviewed by the knowledge engineers to obtain more information about the problem domain, this knowledge is organized and encoded, and the resultant system is evaluated by the experts. With each iteration of the development cycle, the system becomes more refined and complete.

From our initial interviews with the spacecraft engineers, it was clear that there were two basic types of knowledge about the problem domain that were needed by the system: procedural knowledge and structural knowledge about the spacecraft. Hence, we drew the methods to organize our data from two distinct software design methodologies: Object Oriented Design (OOD) and Structured Analysis Design Technique (SADT).[3] In implementing the expert system, we encoded the structural knowledge in frames, and we encoded the procedural knowledge in forward-chaining rules. The ESSOC expert system therefore, may be described as a hybrid-frame/rule-based system.

3.3 Expert System Structure/Operation

The ESSOC expert system is composed of three main portions: 1) the rule base; 2) the knowledge base (ESSOC's internal representation of the satellite); and 3) the user interface. In addition, ESSOC requires processing of the satellite data by custom software. In the following sections, we discuss each of the parts of the expert system in more detail and the flow of data throughout the expert system and its test bed.

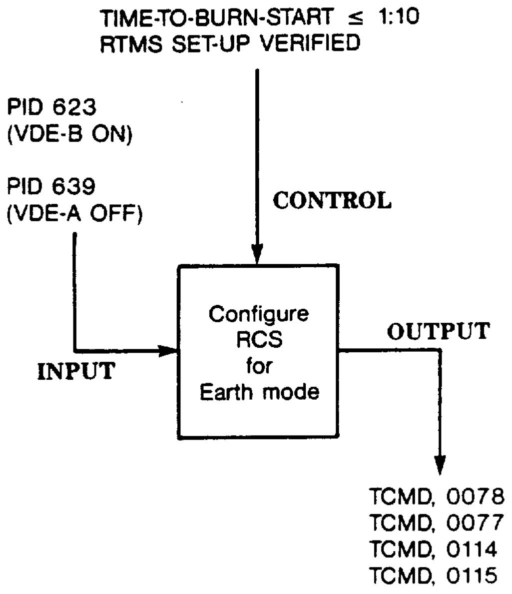
3.3.1 Rule Base

The rule base of the expert system contains all the procedural knowledge of the system (i.e., procedures for detecting and correcting anomalies as well as the procedure for the delta V itself). The procedural knowledge gained from interviewing the engineers was divided into specific phases which were then subdivided into discrete activities. As prescribed by the SADT methodology, each of these activities was then analyzed by identifying the inputs, outputs, and constraints associated with each of the activities.[4] In composing rules, the left (IF) side of the production rule contained the input conditions and the constraint conditions, whereas the right (THEN) side of the rule contained the items in the output portion of the activity description. Examples of an activity decomposition and the resultant ART rule are shown in Figure 2.

Because we anticipated that there would be a large number of rules in the system, and because ART's inference engine considers every rule for matching in every inference cycle, a strategy to speed the matching of rule patterns against the data base was employed. The rules were partitioned into functionally related sets called rulesets. Rulesets may be designated as active or inactive based on the relevance of the function of this ruleset to

ESSOC RULE BASE

SADT ACTIVITY DESCRIPTION:



RESULTANT ART RULE:

```
(defrule S-11-01
  (schema operation-status
    (current-rulesets configure-rcs-for-earth-mode)
    (current-message none)
    (current-command none))
  (schema TCMD-0077
    (status none))
  (schema thruster-package
    (valve-drive-electronics A))
 = >
  (modify
    (schema operation-status
    (current-command TCMD-0077)))
#L (new-line 'view-window)
#L (insert-string 'view-window "Rule S-11-01
    Fired.")
#L (choose-command-option "TCMD, 0077" 16)
#L (clear-window 'rule-window)
#L (insert-string 'rule-window

    IF      the current activity is configure rcs for
            earth mode
       AND  the A valve drive electronics are to be
            used
    THEN    send command TCMD, 0077 to turn
            on the A valve drive electronics"))
```

FIGURE 2. THE SADT DESCRIPTION IS READILY CONVERTED INTO AN ART RULE

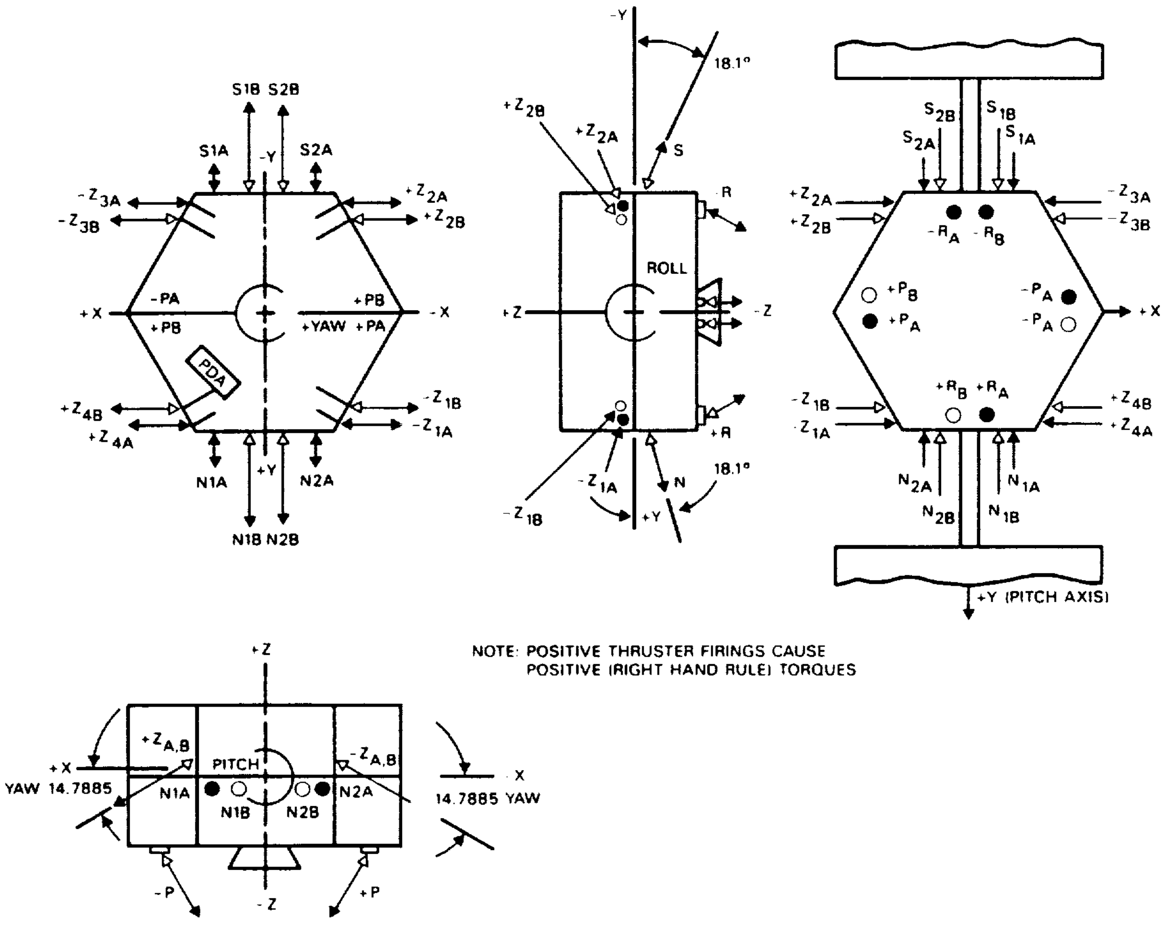
the current status of the maneuver. The status of a ruleset (active or inactive) is dynamically determined by a set of metarules that respond to specific telemetry, timing, and sequencing conditions. The first condition for matching a rule is that the rule be a part of a ruleset which is active. Since only a few of the rulesets are active at any one time in the maneuver, the time that the system spends pattern matching is greatly reduced. The list of rulesets which are active is stored in a data structure in the knowledge base.

In addition to the metarules, there are two other general categories of rulesets: phase-specific and phase-independent rules. Phase-specific rules perform the delta V procedure. For example, there is one ruleset that enables the catalyst bed heaters, and another that opens the propellant valves, etc. There are a total of 22 phase-specific rulesets. Phase-independent rulesets are those that perform monitoring functions. There are a total of six phase-independent rulesets. For a more detailed listing of these rulesets, see the paper "An Expert System for Satellite Orbit Control."[3]

As an example of the operation of a monitoring (phase-independent) ruleset, we discuss the Rhold monitor found in ESSOC. This monitor is active for a considerable period during the delta V procedure, during which several of the phase-dependent steps execute. At intervals that are unknown in advance, the Rhold monitor interrupts the normal procedure to recommend commands. This operation is detailed as follows.

Shortly following launch, a failure rendered 13 of the 24 TDRS hydrazine thrusters unusable,[5] further complicating control of the spacecraft. The failed thrusters are shown (in black) on the diagram of the spacecraft in Figure 3. In particular, the lack of an operating negative roll thruster required an alternative method to provide negative roll torque for attitude control. The workaround developed requires firing a pair of yaw thruster pulses that cancel in yaw but have a fractional negative roll torque. The command sequence for performing the pair firing is called "Rholdn," where n is a number from one to seven denoting the number of thruster pulse pairs. These command sequences must be performed to provide negative roll control authority whenever thrusters are used for attitude control, such as during the delta V procedure.

Currently, the attitude control system specialist instructs the satellite controller to command the spacecraft by observing the earth sensor roll error, and issuing corrective satellite commands based upon his/her intuition and experience. Incorporated in ESSOC is a roll axis controller, the Rhold monitor. The block diagram for this monitor is shown in Figure 4. While unremarkable in design (further details on automatic controllers of this type may be found in the book *Automatic Control Systems*[6]), the ability to use a real-time controller in an expert system illustrates the performance margin and flexibility found in ESSOC.



NOTE: POSITIVE THRUSTER FIRINGS CAUSE
POSITIVE (RIGHT HAND RULE) TORQUES

FIGURE 3. DUAL THRUSTER MODULE LOCATIONS

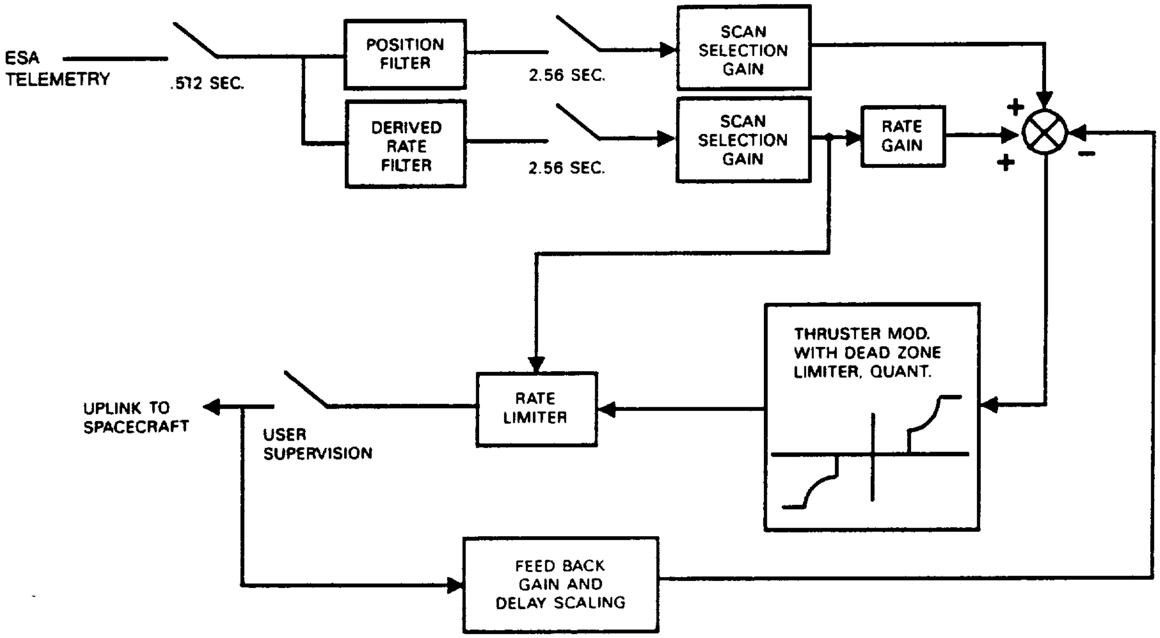


FIGURE 4. RHOLD MONITOR CONTROL LAW BLOCK DIAGRAM

In ESSOC, the controller itself is implemented in LISP on the Symbolics with the exception of the data filters, which are described in Section 3.3.5. Previous implementations of this type of controller by CONTEL, were done in FORTRAN as a closed (no operator control over commanding) loop.[7]

When the controller determines that commanding is required, a condition is set in the expert system knowledge base that causes a rule firing which displays the proper command for user action. As detailed in the user interface section, a monitor-generated rule preempts any procedural commands in requesting user action.

3.3.2 The ESSOC Knowledge Base

As prescribed by the OOD method, features of the spacecraft were identified as objects; the operations which act upon these objects were identified and the object's attributes and values were listed.[8] With the structural knowledge organized in this way, the conversion of the knowledge into ART's schemas (ART's frame-based knowledge representation) was a very natural process. Figure 5 shows an example of a schema which represents a spacecraft object in the knowledge base. ART's relational network allows the system to represent not only parts of the spacecraft, but also the relationships between spacecraft parts. Values from the telemetry data are stored in the knowledge base schemas and updated from incoming telemetry data. Thus, at any given moment the state of the knowledge base reflects the current status of the satellite; this portion of the knowledge base may be regarded as a satellite simulator internal to the expert system.

In addition to information about the spacecraft, certain control structures are also present in the knowledge base. One important control structure that we have already discussed is the schema that keeps track of the rulesets which are active. Another important control device is ESSOC's clock. The clock schema stores the current Greenwich Mean Time and additional needed timing information such as the time until thruster burn start, duration

An ESSOC Control Structure: the ESSOC Clock

```
(defschema operation-time        ;time of delta v operation
  (gmt "330:00:00:00")           ;(ddd:hh:mm:ss)
  (duration 100.0)               ;deci-seconds
  (new-time no)                  ;old/no/yes/new
  (new-duration no)              ;no/yes
  (universal-time 0))            ;seconds since 1/1/1900
```

A portion of the ESSOC Satellite Representation: the th-Z3B Thruster

```
(defschema th-Z3B)
  (instance-of thruster)
  (thruster-id th-Z3B)
  (status dis)                   ;pid 680 (en/dis)
  (health working)               ;(working/failed)
  (duty-cycle 0)                 ;(%)
  (temp 0)
  (redundant-thruster-id th-Z3B)
  (cat-bed-heater-id cat-bed-htr-Z3B)
  (propellant-valve-id prop-valve-Z3B)     ; prop-valve-xxx
  (thermistor-id thermistor-Z3B))
```

FIGURE 5. ESSOC KNOWLEDGE BASE

of thruster firing and whether this information has been changed while configuring for the delta V maneuver. ESSOC's clock is updated from the system clock and may be accessed easily by any of the rules.

3.3.3 ESSOC User Interface

The ESSOC system is equipped with two monitors: a black-and-white monitor and a high-resolution color monitor. The black-and-white console is an interactive user interface that functions as the command terminal. The color monitor is used for generating graphics that augment, rather than replace, displays that are currently available to the satellite controller.

ESSOC displays its recommendations and messages to the user in a specific window on the black-and-white screen. The operator may send or cancel a command, or confirm a message recommended by ESSOC by selecting the appropriate option from the command menu with the mouse. From this user interface, the expert system may query the user for information not found in the telemetry, and may request confirmation that certain procedures have been accomplished before proceding with the delta V. High-priority commands are displayed on pop-up menus that cover the command window, forcing the operator to respond before continuing with the procedure. In separate windows, ESSOC displays a history of recommended commands, a history of the commands that have been sent and a brief justification of the currently recommended command or message. In addition to the command interface present on the black-and-white screen, additional windows give helpful information to the operator concerning the Greenwich Mean Time, length of time prior to thruster firing, and the current phase of the delta V operation. The operator may also send a satellite commands at will from this screen.

The ESSOC color graphics are displayed on a high-resolution (1280 X 1064) 24-bit color graphics screen. While not strictly necessary for ESSOC operation (in contrast to the monochrome display), information on this display is provided to aid the user in his/her decision making.

To date, two real-time displays have been implemented. The first depicts the current configuration of the TDRS Attitude Control Subsystem. The values displayed on the screen are obtained from the spacecraft telemetry found in the ESSOC database. The second depicts the orientation of the spacecraft with respect to the earth. Attitude position limits are indicated by rectangles about the center of the earth (nadir). Earth sensor fields of view are indicated by animated rectangles that are repositioned to reflect spacecraft attitude motion. The coordinate transformations and graphics for the display are coded in LISP; the data are drawn from the expert system's knowledge base.

3.3.4 ESSOC's Inferencing Cycle

The ESSOC expert system's inferencing cycle is a slight modification of the standard match-select-fire inferencing cycle; the ESSOC cycle consists of four distinct steps rather than three. The operational cycle is as follows: first, the expert system examines the information in its database and determines which, if any, of the operational rules are matched; second, the system selects one of the rules that have been matched; third, ESSOC executes one of these rules; fourth, the system reads any data present in the data buffer on the Symbolics into the knowledge base. The pattern-matching and the rule selection, conflict resolution algorithms are provided by the expert system tool ART, whereas the data-polling and parsing functions were custom-made for this application and implemented in LISP. Note that the expert system does not wait for data; if no telemetry data are present in the buffer, the expert system continues inferencing.

Commands and messages are generated and displayed to the user during the third step of the cycle, as a result of rule execution.

3.3.5 ESSOC's Data Flow

Because ESSOC is data driven, it is important to discuss the methods by which data are generated and placed into the system's data structures. As previously discussed, the system and its knowledge base reside on a Symbolics 3675 while the source of data (the simulator TSIM) and the data preprocessor reside on a VAX 11/785 (see Figure 6).

Expert systems are CPU-intensive, often requiring 80 to 90 percent of the CPU's processing power. Because the processing of the telemetry data involves trend determination and conversion to engineering units, which are arithmetically intensive, and because performing of processing on the Symbolics would significantly interfere with the expert system's use of the processor, we decided to perform all the telemetry data processing on the VAX. This design decision greatly enhanced ESSOC's real-time performance. The following paragraphs describe the flow of data in the system, the operational cycle of ESSOC and the way in which the two are integrated.

The flow of information between the expert system and the simulator is bidirectional; spacecraft telemetry data are transmitted from the simulator to the expert system and commands are transmited from the expert system to the simulator.

The transmission of commands to the simulator from ESSOC is totally under the operator's control; commands may be sent at any time the expert system is in operation. Whenever the operator sends a command, the command is transmitted to the VAX. The system is designed so that the simulator will accept the commands coming over the link and will model a response in telemetry.[2]

In contrast, the transmission of telemetry to the Symbolics is controlled by software on the Symbolics. The simulator generates data continuously. To prevent data loss, data is buffered on both the VAX and the Symbolics side of the link. The ESSOC front end proces-

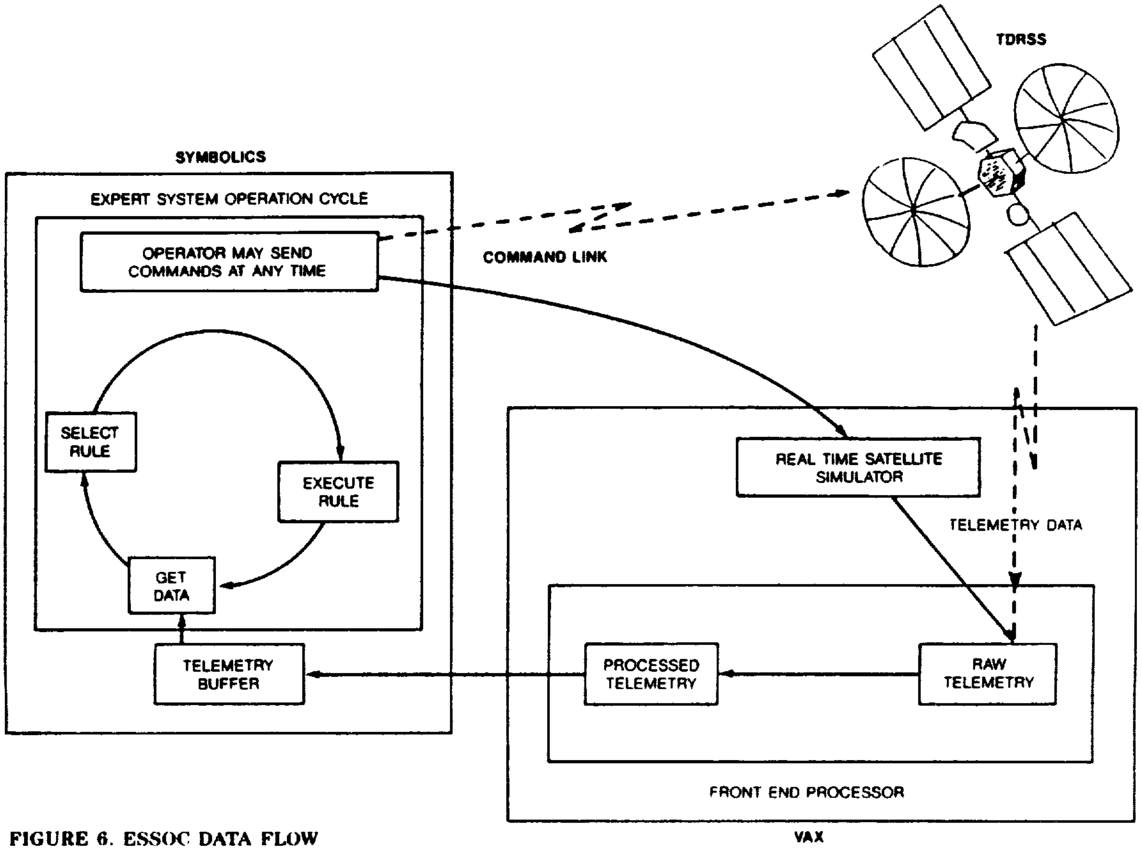


FIGURE 6. ESSOC DATA FLOW

sor converts the raw data into blocks of processed telemetry data.

The ESSOC front end preprocessor receives the binary bit stream and performs real-time conversion of the telemetry data to engineering units. Following the conversion, certain parameters are further processed through sliding average filters and derived rate filters. The data are then associated into object-value-attribute pairs and sent to the output buffer as ASCII strings. The derived parameters are sampled each 0.512 seconds, averaged or fit over 5.12 seconds, and are placed in the processed telemetry buffer at 2.56 second intervals. The front end processor is implemented in FORTRAN on a VAX 11/785, co-resident with the spacecraft simulator.

When the data buffer on the Symbolics side of the link is empty, one frame of processed telemetry is sent across the network in a block and stored in the buffer. At the proper time in its inferencing cycle, the expert system places this buffered information into its knowledge base. At this point, the data buffer on the Symbolics side is empty and another data block is requested from the processed telemetry buffer. Data transmission across the Ethernet occurs asynchronously with respect to the ESSOC expert system operational cycle. It is important to note that the system continues to inference and monitor telemetry data even when the system has requested user input. Because the system continues to check data and inference, anomalous conditions will not go undetected during the time that ESSOC is awaiting a user response.

4.0 Testing

In testing ESSOC, we first tested each individual rule by presetting its conditions in the knowledge base and then determining if the rule fired and if the proper actions were taken.

After the individual rules of the system were verified, the complete rule base was tested by transmitting known data generated by an off-line TDRS-1 simulator to the expert system through the expert system's TSIM-VAX link. Scenarios were constructed that generated data to test the function of each ruleset. Because we had complete control over the data in the scenario, we could determine whether the rules were firing at the correct time and under the proper circumstances. Using an off-line simulator rather than telemetry tapes allowed us to control, as well as generate, anomalous data with which to test the monitoring rule sets.

Future testing of ESSOC will be accomplished by linking the system to the simulator TSIM, although at the time of this writing, this link has not been tested.

5.0 Conclusions and Discussion

ESSOC has demonstrated that expert systems technology has promise for supplementing current communications spacecraft control and monitoring methods. By using an expert system to perform the TDRS-1 delta V procedure, the probability of incorrect commanding can be greatly reduced. Changing spacecraft conditions are detected as they occur, and the proper response made immediately. Virtually any failure mechanism that can be identified in advance may be entered in the knowledge base during the development and operational phases.

While some of the above capabilities may be achieved through other methods (e.g., rule-based expert systems, conventional programming techniques), the hybrid rule/frame-based expert system is faster and far easier to maintain. Since the rulesets are functionally independent, additional rules may be added easily to either expand capability or correct faults.

No discussion of ESSOC's capabilities would be complete without mentioning some of its limitations. The types of support that an expert system (or any other system) may provide for a given fault during a process are: 1) anomaly detection; 2) attaining safe configuration; 3) performing corrective action/identifying workarounds; and 4) cancelling the process. At present, ESSOC provides only anomaly detection, the ability to cancel the procedure, and a limited capacity for workarounds. Performing corrective action/identifying workarounds would theoretically appear to be the most desirable capability to develop in an expert system, but in a majority of cases it is more desirable in practice to attain a safe configuration. A number of reasons supporting this conclusion are listed below.

1) It is simpler to identify a type of problem than it is to correct a specific one. Each problem type has a procedure that corrects a family of problems and places the spacecraft into a safe configuration. Because there is one procedure per problem type rather than one procedure per problem, the number of corrective procedures that the experts must define is reduced.

2) It is more cost effective. While the chance of a particular failure scenario occurring is quite small, considerable expense is required to develop, implement, and test each of a large number of explicit failure scenarios. Conversely, simply configuring the satellite for a safe haven, prior to corrective action by specialist input, prevents this excessive level of expenditure.

3) It is safer. Any of a number of events, which may require distinct recovery procedures, may produce similar symptoms in the telemetry. Thus, the actual failure mode may not have been foreseen at the time of expert system development, leading to incorrect

response and unnecessary risk.

While the above discussion suggests that safe haven anomaly recovery is in general preferable to explicit recovery, there are exceptions. For certain failure modes, the failure causality and corrective action is relatively simple, and consequently, the costs of not performing these actions is great. Likewise, certain failure modes can be identified as having a greater probability of occurrence than others which warrants an increased level of expert system response capability. For these cases, it is preferable to implement a full recovery procedure.

While ESSOC is used for performing the delta V procedure, the techniques used by ESSOC can be generally applied to spacecraft control. A system like ESSOC could be expanded to handle many tasks in satellite operations. The ultimate goal would be to enlarge the expert system so that it could perform all phases of satellite control.

With slight modification to the system, the user can be eliminated from the control loop entirely; the expert system would request user interaction only as unforeseen circumstances occur. Under these circumstances, manpower needs would be greatly reduced.

Acknowledgements:

The funding for development of ESSOC was provided by Contel Federal System's internal research and development fund. The authors thank the TDRS Program Office for their support during the development of ESSOC and acknowledge the dedicated efforts of the other members of the ESSOC development team: Terry Rubin, Ted Russell, and Jim Young.

1 "TDRS Spacecraft Operations," volumes 1-5, TRW Defense Systems Group Document TMD 253, 1985.
2. M.K. Lines-Browning and J.L. Stone, Jr. "An Expert System for Conducting a Satellite Stationkeeping Maneuver." Submitted for publication to the 1987 MILCOM Conference.
3. F. Rook and J. Odibiyi. "An Expert System for Satellite Orbit Control." Expert Systems in Government Conference, Tyson's Corner, VA, 1986.
4. D. Ross and K. Schoman. "Structured Analysis for Requirements Definition." *IEEE Transactions on Software Engineering*, January, 1979.
5. "TDRS F-1 Launch to Synchronous Orbit Spacecraft Subsystem Performance" 31 March 1984. Space Communications Company Document W200A122.
6. B.C. Kuo, *Automatic Control Systems*, Third Edition, Prentice-Hall, 1975.
7. "Gyro-less Acquisition Operations Manual," Space Communications Company Document TMD Z53, 1984.
8. G. Booch. *Software Engineering with Ada*. Menlo Park, California: Benjamin/Cummings Publishing Company, 1983.