

511-63  
198561 p-11

N94-23267

# Link Monitor and Control Operator Assistant: A Prototype Demonstrating Semiautomated Monitor and Control

L. F. Lee and L. P. Cooper  
Advanced Information Systems Section

*This article describes the approach, results, and lessons learned from an applied research project demonstrating how artificial intelligence (AI) technology can be used to improve Deep Space Network operations. Configuring antenna and associated equipment necessary to support a communications link is a time-consuming process. The time spent configuring the equipment is essentially overhead and results in reduced time for actual mission support operations. The NASA Office of Space Communications (Code O) and the NASA Office of Advanced Concepts and Technology (Code C) jointly funded an applied research project to investigate technologies which can be used to reduce configuration time. This resulted in the development and application of AI-based automated operations technology in a prototype system, the Link Monitor and Control Operator Assistant (LMC OA). The LMC OA was tested over the course of 3 months in a parallel experimental mode on very long baseline interferometry (VLBI) operations at the Goldstone Deep Space Communications Center. The tests demonstrated a 44 percent reduction in precalibration time for a VLBI pass on the 70-m antenna. Currently, this technology is being developed further under Research and Technology Operating Plan (RTOP)-72 to demonstrate the applicability of the technology to operations in the entire Deep Space Network.*

## I. Introduction

The Jet Propulsion Laboratory (JPL) manages a world-wide network of antennas, the Deep Space Network (DSN), that provides a communications link with spacecraft. DSN operations personnel are responsible for creating and maintaining this communications link. Their tasks involve configuring the required subsystems and performing test and calibration procedures. The task of creating a communications link is known as precalibration and is a manual, time-consuming process that requires both operator input

of more than a hundred control directives and monitoring of more than a thousand event messages and several dozen displays to determine the execution status of the system. The existing Link Monitor and Control (LMC) system requires the operator to perform a large number of textual keyboard entries and to monitor and interpret a large number of messages in order to determine the state of the system and to selectively identify relevant information from dozens of predefined, data-intensive displays. The tasks required by the LMC create an environment in which it is difficult to operate efficiently.

The goal of the Link Monitor and Control Operator Assistant (LMC OA) task is to demonstrate automated operations techniques that will improve operations efficiency and reduce precalibration time. The LMC OA is a knowledge-based prototype system that incorporates artificial intelligence (AI) technology to provide semi-automated<sup>1</sup> monitor-and-control functions to support operations of the DSN 70-m antenna at the Goldstone Deep Space Communications Complex (DSCC). The AI technology improves operations by using a flexible and powerful procedural representation, by reducing the amount of operator keyboard entries, and by providing explicit closed-loop communications and control through an expert-system module.

The precalibration process used for VLBI on the 70-m antenna was selected as the test domain for the prototype. The LMC OA was field tested at the Goldstone DSCC and performed a semiautomated precalibration for VLBI using actual operational equipment. The test demonstrated that precalibration time can be reduced by 40 percent with the LMC OA prototype.

The LMC OA has three major components: the Temporal Dependency Network (TDN), the Execution Manager (EM), and the Situation Manager (SM). These three components work together to provide a closed-loop, system-level control system for precalibration. The TDN is a directed network that represents parallel procedural paths, precedence relations, preconditions, and postconditions. The TDN is the primary knowledge base for the system. The EM is responsible for traversing the TDN and sending control directives to the subsystems while maintaining the precedence, parallel, and sequential constraints specified in the TDN. The SM works in step with the EM and provides the situational awareness necessary to close the control loop, to detect anomalies, and to support recovery from anomalies. The SM maintains an internal model of the expected and actual states of the subsystems in order to determine if each control directive is executed successfully and to provide feedback to the user.

This article describes the LMC OA prototype and test results. Section II describes the problems with the existing LMC system. The following sections will explain the LMC OA approach used to address the identified problems. The two major concepts that drive the LMC OA design will

be presented, followed by a description of the TDN and the primary knowledge representation in the LMC OA. A detailed discussion of the three major modules and an operational scenario will be presented. In conclusion, the results of operational field testing and the lessons learned from this applied research project will be discussed.

## II. Problem Overview

Currently, for standard operations, an operator is allocated 45 min<sup>2</sup> to perform a precalibration. In the case of more complex operations, like VLBI, an operator may be allocated much more time. Precalibration is a time-consuming process because of limitations in the existing operational monitor-and-control system. Precalibration is a command-line, keyboard-entry system that requires operators to manually send hundreds of directives to subsystems and monitor more than a thousand incoming messages on a text-based scrolling log. The system lacks explicit, informative responses about the state of a directive and does not provide guaranteed communications between the monitor-and-control system and subsystems being controlled. For each directive sent by the operator, the subsystem usually returns a *directive response*; this is simply an acknowledgment from the subsystem informing the operator whether the directive was received or rejected. A directive response does not indicate the success or failure of the directive's execution. The subsystem may also send out *event notice messages*, which relay information about the state of some device in a subsystem. These messages, however, are not explicitly tied to any directive sent. Operators, therefore, must rely on their experience to determine which directive was most likely to have caused the subsystem to send the event notice message. *Monitor data*, which are sent periodically by the subsystems, also provide information about device states. However, monitor data are never displayed automatically or tied to any directive. Instead, a subset of monitor data is formatted into predefined displays that the operators can call up. The operators then must decide which piece of the data they need and which display contains that piece of information. Oftentimes, a display contains many pieces of information of which operators only need one or two.

The inability of the monitor-and-control system to keep up with input from the subsystems causes messages to be dropped at monitor and control. To compound the problem, the subsystems cannot detect when a message has

<sup>1</sup> The LMC OA provides *semiautomated* precalibration because operator interaction is required. Precalibration currently requires several manual operations which could not be done through existing DSCC monitor and control interfaces. Furthermore, certain support and subsystem data were inaccessible to the LMC OA, thus requiring input from the operator.

<sup>2</sup> The standard time allocated for precalibration and postcalibration for each user or project activity is listed in Appendix A of the *DSN Scheduling Code Dictionary*, JPL document 842-204: 10-009, Rev. B (internal document), Jet Propulsion Laboratory, Pasadena, California, June 20, 1989.

been dropped and, thus, cannot resend information. This situation causes false alarms that can inundate the user with messages and often hide real alarm situations. Finally, the system is prone to input errors. A simple precalibration pass requires more than a hundred directives. It also requires the operator to manually identify and type each directive and its parameters. A subsystem, therefore, can take several minutes to recover from a simple typographic error.

Operators use a variety of support data—schedules, predict files, sequence of events, and pass briefings—to determine the type of pass, the spacecraft being tracked, and the method used to configure the communications and processing equipment. Information contained in the support data files is also used to determine the correct parameters for the control directives. Because these files are not available electronically for easy viewing and usage, the operator must refer to the hard copy version of these files and manually enter numerical parameters for control directives, when the numbers oftentimes are accurate to 10 decimal places. An entry error in any one of the digits could cause a major problem in the system.

The most difficult part of precalibration is the determination by the operator of what directives need to be sent and how the directives should be ordered. Currently, end-to-end representation of operations procedures does not exist. The documentation that is available addresses a specific subsystem or spacecraft or provides a general overview of an activity. As a result, operators must rely on their own experience to assemble an end-to-end operational sequence. Thus, the operational sequences vary from operator to operator, leading to inconsistencies in operations and making recovery from anomalies difficult.

The following are the specific operability problems identified with the existing LMC system.

- (1) Extensive manual entry is required of the operator.
- (2) The lack of integrated monitoring tools for the operator makes it difficult or nearly impossible to perform parallel operations. The operator must mentally interpret displays and text messages to determine correct execution of a directive.
- (3) False alarms due to dropped messages occur frequently, and because dropped messages are not detected, they are retransmitted by the subsystem, giving the operator an incomplete picture of the system state.
- (4) The lack of on-line access to usable support data increases the need to integrate information from multi-

ple sources. Entry of complex numerical parameters increases the chances of typographical errors.

- (5) There is no end-to-end representation of the operations procedures.

### III. Closed-Loop Control and Situational Awareness

Two major design concepts found in the LMC OA system are closed-loop control and situational awareness. In the LMC OA context, closed-loop control means that all control actions (i.e., directives) have explicit feedback regarding the success or failure of the requested action. Under the existing monitor-and-control system, no single message can report the status of a directive. Rather, the operator must sift through many different data messages returned by the subsystems and many different displays to determine the status of the directive. Moreover, this present process of filtering and identifying pertinent data is time consuming. The LMC OA, however, integrates all available information sources and provides the operator with clear, consistent, explicit feedback for every control action.

Situational awareness, another feature of the LMC OA, allows the operator visibility into the state of the system and the state of procedure execution. In the current LMC, a large set of displays provides the operator with visibility into the state of the system. However, the information is difficult for the operator to interpret. Information important to the operator is not easily accessible because there are too many displays and none of them are user-definable. The LMC OA team did not redesign the displays because the resources to tackle such a significantly large problem were not available. Rather, the LMC OA prompts the user with the name of the display and the value to look for. In this manner, the LMC OA makes it slightly easier for the operator to determine the state of the system by explicitly providing the display name and monitor item to look at.

The second criterion for situational awareness is visibility into the state of procedure execution, which means that the operator knows the progress and status of procedure execution. Currently, since there does not exist end-to-end procedural documentation, the operator depends on experience to determine the procedure. To determine the state of procedure execution, the operator must interpret a large number of messages from the subsystem. However, through an extensive knowledge engineering effort, an end-to-end integrated procedure for VLBI was created and represented in a TDN. The TDN is a clear and intuitive way of representing the procedure to the user. Furthermore,

through the color-coded, graphical display of the TDN, the operator can immediately determine the execution status of the procedure.

## IV. Temporal Dependency Network

One of the problems with the existing LMC is the lack of end-to-end procedural documentation. To perform a VLBI precalibration, the operator must refer to several operation manuals which describe individual subsystems or portions of the procedure. The operator must then manually create an integrated procedure. In some cases, operators create and use, as a reference, personal "cheat sheets" that describe what needs to be done. The lack of a single source of documentation that describes the VLBI precalibration procedure results in inconsistent operations. Actual operations rely heavily on an individual operator's experience and expertise. To automate operations, an integrated procedure for VLBI precalibration was created.

The approach to knowledge engineering involved first learning about the system through existing documentation and noting inconsistencies and missing information. The next step involved discussing the procedure with operators, engineers, technicians, and scientists to get their viewpoints and to clear up inconsistencies as much as possible. This led to the development of an initial TDN. The TDN became the much-needed common language between the knowledge engineers and the knowledge sources. This LMC OA knowledge engineering effort is the only known attempt, within the DSN, to produce a single, coherent, and consistent baseline operational sequence for precalibration that merges the viewpoints of all users.

The TDN, shown in Fig. 1, illustrates an end-to-end operational sequence for VLBI. Sequential, parallel, and optional operation sequences are identified in the TDN. Each block in the TDN contains directives that are sent to the subsystems sequentially. Blocks have precedence constraints where the directives cannot be sent until all of its predecessor blocks' directives have successfully completed execution. Each block has associated precondition and postcondition constraints. These constraints define the state the system must be in *before* starting each block of directives and *after* successful execution of those directives, respectively. Each block may also have temporal constraints that limit the start and completion of the directives to a specific time or time interval.

## V. LMC OA Design

The goal of the LMC OA is to provide both closed-loop control and closed-loop communications for the oper-

ator. There are two major modules in the LMC OA: the TDN Execution Manager (EM) and the Situation Manager (SM). Other modules that will be discussed include the Block Execution module, Router, Monitor Data Handler, and DSN Data Simulator. An overview of the design is presented in Fig. 2.

### A. TDN Execution Manager and Block Execution Modules

The TDN EM traverses the TDN identifying blocks that are ready to execute. Blocks whose precedence constraints are satisfied are started. When a block is started, the user is asked to parameterize any unparameterized directives. The preconditions are then evaluated by the SM. A block's directives are sent only after the SM verifies that the preconditions are satisfied. Once a directive is sent, a directive response must be received before the next directive in the block can be sent to a subsystem. After the last directive is sent and its corresponding response is received, the block's postconditions are checked by the SM. If the postconditions are satisfied, the block of directives is considered completed.

### B. Situation Manager

The SM provides situational awareness within the LMC OA. It is also an AI-based module that verifies correct execution of blocks of directives by checking postcondition constraints. Problems can be detected and simple recovery assistance provided. To keep track of the state of the system, the SM keeps an internal model of all hardware and software devices that can be monitored in the system. Each device represented in the model has attributes that reflect the state of the device. Each attribute has a pair of values: an *expected value* and an *actual value*. The expected value of an attribute, in the form of a postcondition, is set when a directive is sent to the subsystem. The actual value of an attribute is set when the subsystem sends messages noting state changes in the subsystem. Every directive sent to a subsystem is expected to cause certain known changes on the states of the devices in the subsystem. Each time a directive is sent, the expected values of the attributes in the device model are updated.

In addition, several data types are used to set the actual values of the device attributes: event notice messages, directive responses, monitor data, and operator input. Event notice messages describe explicitly the actual states of devices. Directive responses provide information on whether the directive has been received by the subsystem. In some cases, these responses also provide progress and completion data. Monitor data are blocks of status information that are sent periodically by the subsystems. Monitor data

usually provide more information than event notice messages. In certain situations, operator input is requested. Although the operator is provided a set of predefined monitor displays, the information in these displays is not always available from the monitor data blocks. These displays are generated as bit-map displays at the subsystem level and are unavailable to the LMC OA because of format and DSN operational restrictions. Therefore, for certain directives, the operator must obtain information from the displays and enter it into the LMC OA. This information is used to set the actual value of an attribute in the SM internal device model. All four electronic data types provide information about the actual state of a device, but they do not give explicit information about whether a directive was executed successfully. However, by using information about the expected and actual states of devices, the success of a directive can be inferred. With the SM maintaining its device models, information about the state of the system and the state of the procedure is always available to the operator.

### **C. Router, Monitor Data Handler, and DSN Subsystem Simulator**

In addition to the TDN EM, block execution, and SM modules, there are several other supporting modules. The Router handles all communications between the LMC OA and the DSN subsystems and serves as a translator between the DSN 890-132 protocol and the LMC OA internal data representations. It receives and decodes input from the DSN subsystems and directs the input to either the TDN EM, the SM, or both. It also formats the directives into communication packages that are sent to the subsystems. The Monitor Data Handler receives Monitor Data blocks from the subsystems and stores them in the Monitor Data database. Since access to the operational environment is limited, a DSN Subsystem Simulator was implemented to simulate the directive responses and event notice messages from the subsystems for testing.

### **D. User Interaction and Status Displays**

One of the LMC OA goals is to provide consistent interaction and meaningful displays to keep the user aware of what is transpiring in the system. The primary method of interaction is through menu or button selections using a mouse. The operator may be asked occasionally to enter a value or response. The primary interaction window is a block-level display of the TDN and provides a high-level, end-to-end sequence of operations. A color bar in each block shows the status and progress of each block: a gray bar means the block is inactive, a green bar means the directives are executing, a red bar means an anomaly has occurred, and a blue bar means the directives have been

completed successfully. The portion of the color bar that is green is proportional to the number of executed directives in the block.

The operator can call up a lower-level display for each block that lists the preconditions and postconditions for each block and shows the state of the block and the state of each directive in the block (inactive, executing, paused, anomaly, etc.). At the TDN level, there are controls to pause, resume, and stop execution. Block-level and directive-level controls allow the user to pause, resume, and skip execution. Icons are used to show the user whether a block is paused or skipped.

The SM anomaly messages that require a user response are displayed in a separate window. A synopsis is displayed in a scrolling portion at the top of the window. By selecting a synopsis, the operator can display a description of the anomaly in the bottom portion of the window. The operator can then enter the requested input or select a default option.

The scrolling event log lists all the input to and output from the LMC OA system. A command line window allows operator control outside of the TDN. Another display shows the end-of-pass report as it is being filled in by the LMC OA. With the existing LMC system, the operator must write down the time at which certain directives were executed and their results. At the end of the pass, the operator must also write a set of paper reports. The LMC OA system, however, internally logs the time, parameters, and responses for each directive and automatically generates reports.

## **VI. Operational Scenario**

A typical operations scenario using the LMC OA follows: The operator starts the LMC OA system and selects a specific precalibration task, like VLBI. The corresponding TDN and knowledge bases are loaded, and the TDN is graphically displayed. The operator then enters the specific parameters for the next pass based on the support data. Directives that require real-time data input, like weather information, contain place holders for parameters. The operator can also tailor the TDN, skipping any unnecessary blocks, entering special directives, or establishing break points, as needed. The process of preparing the TDN for a specific pass can be done at any time preceding the designated pass start time. At the start of the pass, the operator selects the start option by a single click of the mouse to start execution of the TDN. The TDN can be paused or halted at any time during the process. The

operator watches the execution of the TDN by following the color coding in the graphical user interface.

At any time, the operator may bring up low-level displays to see the execution state of the individual control directives. The low-level display is updated automatically when each directive is sent and when completion is verified. The SM and TDN EM work in tandem to ensure that the control directives are correlated with the monitor data and event messages. This correlated information is then summarized and presented to the operator. If the SM detects a problem, it reports the problem along with recovery suggestions to the user. The user selects a recovery option which will cause the TDN execution to continue or halt the execution of the TDN. A command window is provided so the operator can enter any control directives into the system. Another window displays a scrolling log of all incoming directives, directive responses, and event notice messages. Additional windows provide a pass summary report and link status. During execution of the TDN, the operators are able to view the detailed subsystem displays on the LMC console. (These displays were not reimplemented in the LMC OA due to resource constraints.)

## VII. Results

The LMC OA prototype was tested at the Goldstone DSCC's 70-m antenna while performing a VLBI precalibration procedure. The LMC OA was successfully tested over a 3-month period at the Goldstone DSCC. The tests were made in conjunction with maintenance and, despite interruptions, the LMC OA performed a VLBI precalibration in 27 min, compared with the standard time of 45 min. This is a 44 percent reduction in precalibration time. In addition, the LMC OA reduced the number of operator-entered directives from more than a 100 to 0 directives and 14 parameters.

The LMC OA prototype was implemented using Objective-C, Interface Builder, and the C Language Integrated Production System on a NeXT workstation running the MACH operating system. In addition, a 386 personal computer (PC) running translation software served as the gateway between the DSN network running proprietary protocols and the NeXT workstation running TCP/IP. The PC was equipped with an IEEE 988 card which could communicate with an Ungermann Bass Interface (UBI) Network Interface Unit (NIU) running DSN proprietary protocols. The PC was also equipped with an Ethernet card running TCP/IP and the PC-NFS package, which provided socket communications to the NeXT workstation. The translation software running on the PC gateway was

developed by a previous project and modified by the LMC OA team as a gateway between the NeXT workstation and the DSN network.

## VIII. Lessons Learned

Many elements contributed to the success of the LMC OA system, yet there were difficulties to overcome. This section examines both successes and difficulties.

- (1) *The LMC OA prototype successfully applied AI technology to provide semiautomated precalibration.* The LMC OA prototype focused on two major concepts: closed-loop control and situational awareness. The LMC OA prototype demonstrates that with the right technology precalibration can be performed in significantly less time.
- (2) *The TDN is a powerful and flexible procedural representation.* The TDN is powerful enough to represent end-to-end operations, constraints, and parallel paths. The TDN can also be used to handle contingencies and anomalies. It is flexible enough to be easily changed and can be adapted for other domains. The representation is simple enough so that it is easy to encode internally and easy to explain to the users.
- (3) *The TDN, in addition to being a procedural representation, is a valuable knowledge engineering tool.* The TDN provides knowledge engineers and operators with a common focal point to work from. Its intuitive nature makes the format easy to understand and easy to use by both the operators and the knowledge engineers.
- (4) *Station operators and JPL training and engineering personnel have valuable knowledge and experience that should be used.* The successes described are due to the excellent support provided by the personnel at the Goldstone DSCC, CTA-21, and other JPL personnel in training, operations, and engineering. Because of their experience, they have a wealth of information, but it is not yet documented. This information is critical to developing the TDN.
- (5) *Knowledge engineering must be performed at the very beginning of the project.* The knowledge engineering process is an important part of building a knowledge-based system. The bulk of the LMC OA development effort was spent in knowledge engineering.
- (6) *Documentation must be kept current, operability standards must be enforced, and documents must be*

*integrated.* A directive dictionary and device models should be provided by the subsystems. In the knowledge engineering process, knowledge engineers often found out-of-date documentation and operational modifications documented in operators' personal notes. This leads to error-prone and inconsistent operations. Furthermore, standards specifying format and content must be followed in order to ensure that each document contains information as expected and at the same level of usefulness and quality. Documents must be integrated and cross-referenced. There are hundreds of documents, and not having some form of cross-reference to easily identify the documents' contents makes operations and knowledge engineering difficult. Providing a directive dictionary and device models for each subsystem can identify many of the side effects that are currently being ignored.

- (7) *The TDN should be used as end-to-end procedural documentation.* In the process of creating the TDN, the LMC OA knowledge engineers found many documentation manuals but none that contained descriptions of integrated, end-to-end procedures. End-to-end procedural information is necessary for efficient operations. The TDN is an effective tool for representing this type of information.
- (8) *On-line documentation and capabilities to search the documentation should be provided.* This will assist with fast and efficient searches.
- (9) *Complete access to monitor-and-control data must be provided.* This is essential for implementing closed-loop control, which in turn enables automation. The current environment does not provide electronic access to all data. Naming and usage of data items are inconsistent. There is also inconsistency in the meaning of directive responses. A centralized, automated monitor-and-control system must have electronic access and the ability to manipulate monitor-and-support data, reliable and guaranteed communications, and remote monitor and control of subsystems.
- (10) *A data management strategy to store and easily retrieve data should be developed.* Information that needs to be stored includes monitor-and-event data, knowledge bases, support data, directive libraries, and TDN libraries.
- (11) *An environment where both systems and knowledge verification testing can be performed needs to be built.* There are three types of testing for a knowledge-based system: compatibility, system, and knowledge verification. CTA-21 was used for net-

work compatibility testing. However, CTA-21 no longer has a full suite of test equipment, and this makes it impossible to perform system or knowledge verification tests. Systems testing and testing of the LMC OA functions were conducted at JPL, using a subsystem simulator, and at DSS 14. Knowledge verification could only be done at DSS 14 because equipment was unavailable at CTA-21.

- (12) *Visibility into the system and training tools that allows the operator to maintain operational skills must be provided.* There is no such thing as a fully automated system. When unforeseen events happen, operators must know what the system is doing so they can take over operations when the automated system is out of its league. Operators must have flexible control and the ability to completely override the system. Embedded training will provide operators with the mechanism for maintaining and further developing their analysis and problem-solving skills in an operational environment.
- (13) *Smarter subsystems should be built.* Subsystems should be able to perform their own self-test and anomaly detection, isolation, and recovery whenever possible.

## IX. Related Efforts

The technology demonstrated in the LMC OA can be extended to other operations in the DSN. Current efforts include extending the LMC OA to control multiple activities as well as using the system as the cornerstone of an operations automation thrust at DSS 13, the 34-m experimental antenna.

An architectural study of the DSN specifies that in the future one operator must be able to monitor and control multiple activities. At issue is the question of how much information an operator can process and how to organize the enormous amount of data so that the operator can at all times manage multiple tasks. The prototyping effort will involve the development of intelligent user interfaces and advanced data management systems. The current effort, sponsored by NASA Code O under Research and Technology Operating Plan (RTOP)-73 and NASA Code C under the AI-RTOP, is researching and identifying what technology is required to provide a multilink monitor-and-control capability for the operators. In 1994, the identified technology will be incorporated into the LMC OA to provide a semiautomated, multilink monitor-and-control capability.

The LMC OA is the starting point for an effort to demonstrate a systems approach to automation in the

DSN. The focus of the newly created RTOP-72, DSN Station Operability, is to research and identify technologies that will support automated and remote operations for the DSN. The technology will be demonstrated in the development of an automated monitor-and-control system with remote capabilities at DSS 13. The DSS-13 baseline monitor-and-control system, delivered for operational use in 1993, uses standard protocols such as Open Systems Interconnection (OSI), Manufacturing Messaging Specification (MMS), and commercially available packages such as RTWorks from the Talarian Corporation. The data communications and access infrastructure provided by the DSS-13 Monitor and Control system provides a strong baseline for testing automation concepts. In 1994, RTOP-72 will develop a systems approach to incorporate link-and subsystem-level automation. In addition, RTOP-72 will: (1) implement and deliver the LMC OA for DSS-13 operations; (2) develop an automated station monitoring prototype based on the Multimission Automation for Real-time Verification of Spacecraft Engineering Link (MARVEL) system; (3) develop a plan for integrated data management services, including the identification of data required for automation and improved operability; and (4)

develop a prototype of a link health and performance monitoring system.

## **X. Conclusion**

Knowledge-based systems will play a major and enabling role in improving operability and capabilities of future ground systems at the DSN. The LMC OA prototype demonstrates the feasibility and benefits of AI-based automation in DSN operations. The benefits of an operational, semiautomated monitor-and-control system are (1) reduction in precalibration time; (2) reduction in keyboard entry, which reduces occurrences of typographic errors; (3) capability of parallel operations; and (4) increased operator efficiency via closed-loop control. The LMC OA system demonstrates several operational improvements. It provides the operator with mechanisms for closed-loop control and situational awareness. It provides an end-to-end procedural representation for precalibration using a TDN. And it reduces the number of keyboard entries required by the operator. Furthermore, current efforts are showing that this technology is applicable to the DSN as a whole.

## **Acknowledgments**

Past and present team members who have contributed to this version of the LMC OA are Kristina Fayyad, Randall Hill, Juan Urista, Kathryn Sturdevant, Jay Wyatt, Crista Smyth, and Patricia Fernandez. The contributions of Pam Wolken, Terry Dow, Dave Girdner, Dan Kiewicz, and the many operations personnel at the Goldstone Deep Space Communications Complex are gratefully acknowledged.



## Appendix

### Bibliography

- L. P. Cooper, R. Desai, and E. Martinez, "Operator Assistant to Support Deep Space Network Link Monitor and Control," SOAR Symposium, Houston, Texas, August 1991.
- L. P. Cooper, "Operations Automation Using Temporal Dependency Networks," Technology 2001, San Jose, California, December 3-5, 1991.
- K. E. Fayyad and L. P. Cooper, "Representing Operations Procedures Using Temporal Dependency Networks," Space Ops '92, Pasadena, California, November 1992.
- R. W. Hill, and L. F. Lee, "Situation Management in the Link Monitor and Control System," Space Ops '92, Pasadena, California, November 1992.
- L. F. Lee and R. W. Hill, Jr. "Process Control and Recovery in the Link Monitor and Control Operator Assistant," Space Operations, Applications, and Research Symposium, Johnson Space Center, Houston, Texas, August 1992.
- NeXT Operating System Software*, Redwood City, California: NeXT Computer, Inc., 1990.
- J. Urista, "Using MACH Threads to Control DSN Operational Sequences," Space Ops '92, Pasadena, California, November 1992.
- B. Wilkinson, "Operability Engineering in the Deep Space Network," Space Ops '92, Pasadena, California, November 1992.

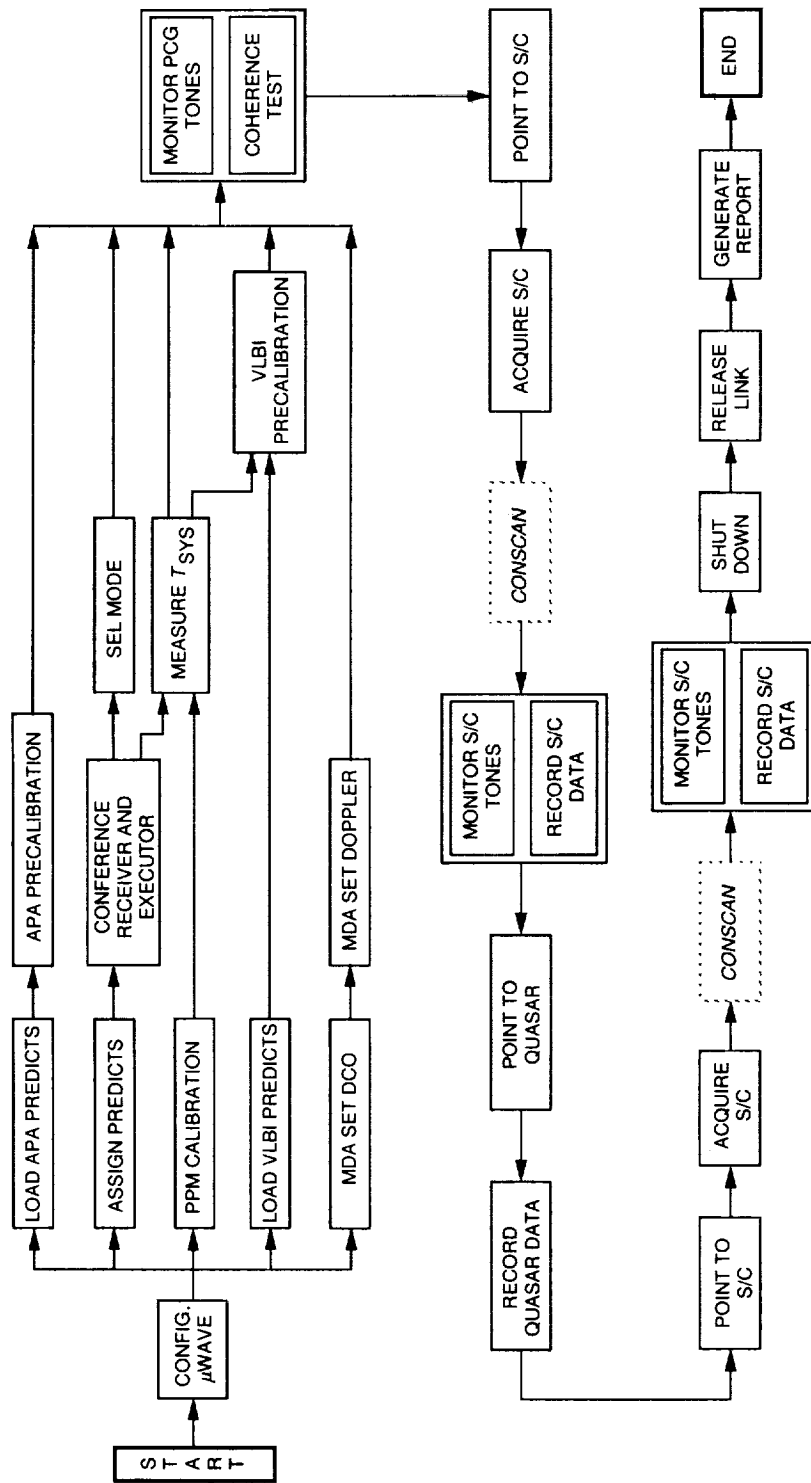


Fig. 1. A high-level VLBI temporal dependency network.

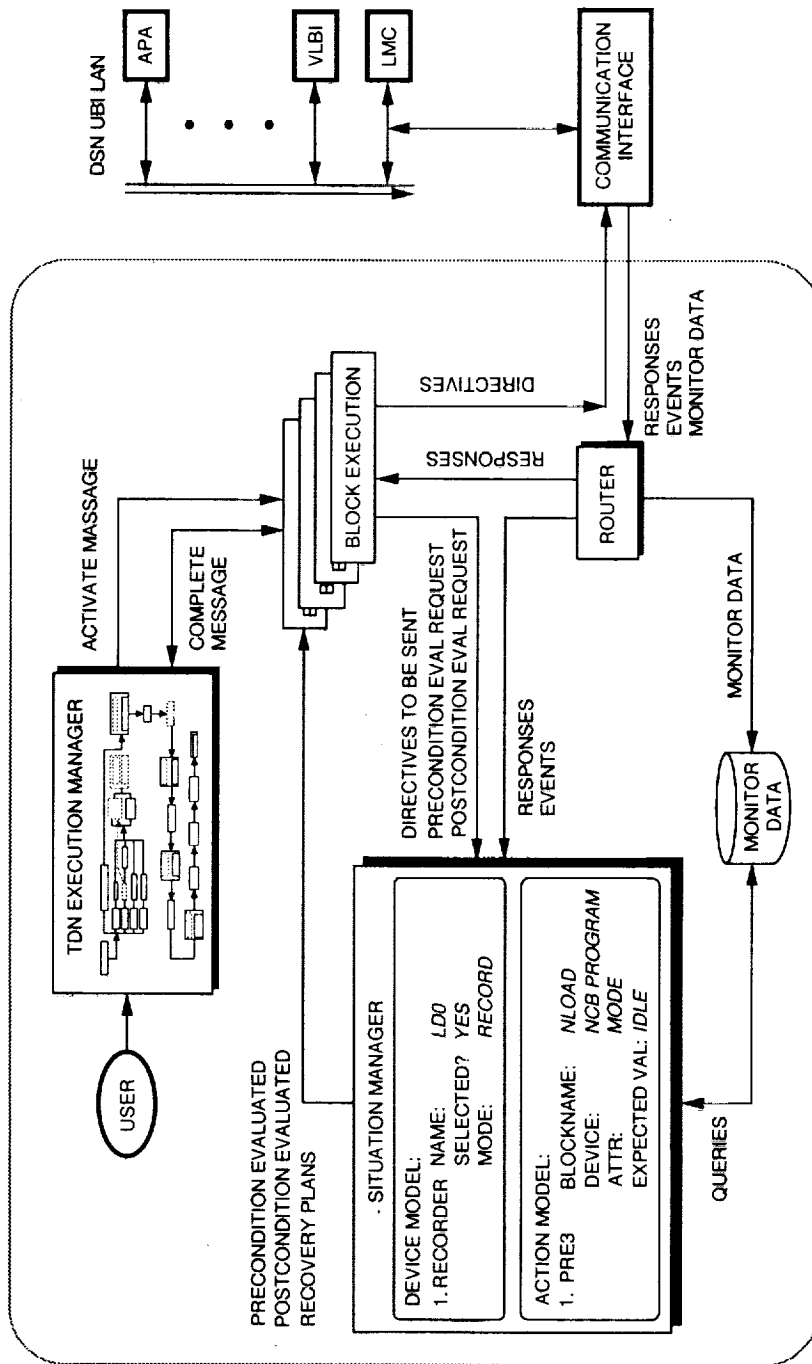


Fig. 2. The LMC OA design.