NTIA Report 90-269

Design of Experiments to Quantify Communication Satellite System Performance

Robert D. Cass
Martin J. Miles

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary

Janice Obuchowski, Assistant Secretary
for Communications and Information

November 1990

PREFACE

The Institute for Telecommunication Sciences (ITS) is conducting a series of projects concerned with the roles of advanced communication satellites in Integrated Services Digital Networks (ISDN). One project in this series is the internally funded ITS advanced satellite study program. This program provides technical support and guidance on end-to-end system performance experiments relating to the NASA Advanced Communication Technology Satellite (ACTS). This report is concerned with the application of American National Standard X3.141 for developing experiments to assess the end-to-end system performance of advanced communication satellites, including ACTS.

Certain commercial systems, equipment, and software products are identified in this report to adequately describe the designs and conduct of research or experiment. In no case does such identification imply recommendation or endorsement by the National Telecommunications and Information Administration, nor does it imply that the material or equipment identified is necessarily the best available for the purpose.

CONTENTS

LIST OF FIGURES

LIST OF FIGURES (cont.)

LIST OF TABLES

LIST OF ACRONYMS

ACTS	Advanced Communications Technology Satellite
ANS	American National Standard
ANSI	American National Standards Institute
ASCII	American Standard for Communication and Information Interface
BBP	baseband processor
BCO	bit comparison outcomes
BRI	basic rate interface
C/No	carrier-to-noise density ratio
CO	central office
CPE	customer premises equipment
DAMA	demand assignment multiple access
Eb/No	bit energy-to-noise density ratio
GOES	Geostationary Operational Environmental Satellite
IF	intermediate frequency
ISDN	Integrated Services Digital Network
ITS	Institute for Telecommunication Sciences
LBR	low burst rate
MCS	Master Control Station
NASA	National Aeronautics and Space Administration
NGS	NASA Ground Station
NIST	National Institute of Standards and Technology
NTIA	National Telecommunications and Information Administration
Pe	probability of bit error
S/N	signal-to-noise ratio
SS/TDMA	satellite-switched time-division multiple-access
TDMA	time-division multiple-access
UF	User Fraction
VSAT	very-small-aperture terminal

DESIGN OF EXPERIMENTS TO QUANTIFY COMMUNICATION SATELLITE SYSTEM PERFORMANCE

Robert D. Cass and Martin J. Miles*

This report describes the steps for designing experiments to quantify the performance of a communication satellite system according to the methods specified by ANS X3.141. Performance is described in terms of performance parameters that are user-oriented and system-independent. The design is intended to efficiently obtain estimates of performance parameters that are relatively free of bias and can be stated with known precision. Service, traffic, propagation path, and propagation impairment are the primary factors. Factorial designs are recommended for the experiment.

Key words: Advanced Communications Technology Satellite; American National Standards; dependence; experiment design; factorial designs; factors; performance parameters; precision; sample size

1. INTRODUCTION

The role of communication satellites is evolving from that of a simple point-to-point transmission link to one of a network switching center. The force behind this evolution is the advancing technology of communication satellites.

Lovell and Cuccia (1984) of the National Aeronautics and Space Administration (NASA) have classified communication satellites into three types:

1. Type A satellites carry one or more simple transponders and an earth-coverage antenna.

2. Type B satellites carry multiple fixed antennas and multiple transponders where, using ground control, the interconnectivity between the beams and transponders can be rearranged.

3. Type C satellites are similar to type B satellites but they carry on-board message-switching capabilities.

The transponders for both type A and type B satellites are analog transponders which support single or multiple communication channels. Type C satellites dynamically control beam coverage areas and route the message traffic to the appropriate beams. In some designs, NASA's Advanced Communications Technology

*The authors are with the Institute for Telecommunication Sciences, National Telecommunications and Information Administration, U.S. Department of Commerce, Boulder, CO 80303-3328.

Satellite (ACTS), for example, the transponders are digital transponders which regenerate the signal.

The first commercially viable communication satellites were simple analog microwave repeaters in geostationary orbit. Their main function was to provide point-to-point, high volume trunking for telephone and television systems. Communication satellites are still not much more than repeaters - even with today's increased transponder capacity and down-link power, higher operating frequencies, time-division multiple-access (TDMA), digital modulation, and frequency reuse.

The NASA ACTS program takes the satellite beyond this simple repeater role by providing an experimental advanced communication satellite that will demonstrate improved repeater technology and on-board baseband switching for traffic routing and network control. The satellite will operate in the Ka band (17 to 31 GHz) with small diameter spot beam dwells and cross-polarization to reduce interchannel interference and increase frequency reuse. The on-board switch will operate in either a microwave mode, switching at the intermediate frequency (IF) like type C satellites (INTELSAT VI), or in a baseband mode, operating as a demand-assignment multiple-access (DAMA) system and include digital signal regeneration and forward error control coding.

Satellites, such as ACTS, and inexpensive very-small-aperture terminals (VSAT) will enable economical, thin-route, two-way communication services to sparsely populated and inaccessible areas. These satellite systems will also facilitate two-way mobile communication. Additionally, they should fit into Integrated Services Digital Networks (ISDNs) where the satellite itself becomes one of the ISDN switching nodes.

Satellites with on-board ISDN switching capability could be used to provide private ISDN services for large companies. They could interconnect geographically dispersed locations with high-volume 23B+D Primary Rate ISDN (1.544 Mb/s) or multiple Mb/s broadband ISDN circuits. These satellites could also support two-way, thin-route traffic with direct links of the Basic ISDN interface (2B+D channels at 144 kb/s) to the customer premises equipment (CPE). Other satellite-based or satellite-augmented ISDN applications include small transportable terminals for temporary network access and mobile ISDN connectivity. These ISDN-compatible satellites working in conjunction with fiber trunks and terrestrial switches could also provide network backup and quick emergency service restoration.

1.1 The Need to Specify System Performance

Since communication satellite systems are evolving into new roles, methods to measure their performance must also evolve. There are two strong reasons for this need:

1. Communication satellite systems are being used for a larger portion of data communication networks.

2. Users are becoming less concerned with the technical design of data communication systems and more concerned with performance.

When satellites provide simple analog repeater service, users are primarily concerned with the system capacity (measured as the number of analog voice circuits or video channels the system can support) and baseband signal-to-noise ratio (S/N). These parameters are a function of the modulation techniques and the carrier-to-noise density ratio (C/No) of the total satellite link (where C/No is established by the basic link budget calculations). Very often, users are neither experienced nor interested in the details of satellite link budget calculations; however, they are very interested in how well the system meets their capacity, S/N, and availability requirements.

The situation for digital satellite transmission links is not much different from that for analog links. Capacity is still the yardstick for measuring performance, and the capacity continues to depend on link budget calculations. (The signaling rate-dependent, bit energy-to-noise density ratio [Eb/No] is used in place of C/No as the primary link budget parameter.) However, capacity is now measured as the user information bit transfer rate with a specified bit error probability (Pe). The Eb/No link parameter fixes the Pe for given source and error-control coding and modulation techniques. Users are now interested in bit and block error probabilities and information transfer rate in addition to system availability.

With advanced communication satellites incorporating more of the network switching functions (and thus accommodating more users who access the network on a DAMA basis), link budget calculations no longer fully specify system performance. Users are becoming concerned with parameters such as access and disengagement time and blocking probability. The transmission link is also more dynamic in these type C satellites, with forward error-control coding and reduced transmission rates "switched in" during rain fades.

Quantifying the performance of these advanced type C satellites can become an arduous task. However, if one views these satellite systems as the digital communication networks they are becoming, the task is quite manageable. This is due to two standards approved by the American National Standards Institute (ANSI). The first standard, ANS X3.102 (ANSI, 1983), defines a set of user-oriented, system-independent performance parameters, and the second standard, ANS X3.141 (ANSI, 1987), specifies methods for estimating the performance parameters. Together, they provide a uniform method to specify the performance of data communication systems. That is, they provide a common "yardstick" by which to compare the performance of dissimilar systems that use satellites, optical fibers, microwaves, or any other transmission media, switching systems or design. The suitability of these standards for quantifying satellite system performance has been discussed (Cass, 1988).

ANS X3.102 and ANS X3.141 offer a convenient means for comparing competing services. These standards can also be used to track system degradations and identify potential bottlenecks. They augment the link budget and capacity calculations and help describe the system performance.

1.2 Purpose and Scope of Report

The purpose of this report is to describe the design of experiments to characterize the performance of communication satellite systems according to ANS X3.102 and ANS X3.141. Although the experiment design described in this report is primarily intended for NASA's ACTS, it is applicable to any digital communication satellite system.

The main reasons for designing an experiment are to obtain unambiguous results at minimum cost, to learn about interactions among existing conditions, and to measure experimental error. (A well conceived design is more important than subsequent sophisticated statistical analysis.)

The ANS X3.141 standard defines seven steps in the design of an experiment to estimate the performance of a data communication system/service:

- Define the objectives.
- Select the performance parameters.
- Define the population of trials.
- Identify the conditions of the experiment.

- Select random samples of a specified size from the population.
- Specify the combination of levels of conditions to be observed.
- Select a mathematical model.

Figure 1 is a structured design diagram showing relationships among the seven major activities of experiment design.[1] These activities are the subject of this report; note that the number of the section within this report that discusses each activity is shown within the appropriate oval (which indicates activity).

The National Telecommunications and Information Administration (NTIA) has created software and supporting documentation that implements these standards (to be published circa 1990-1991)[2]. For each test, the process implemented by the software starts with the source user's computer automatically calling the destination user's computer, transferring a preset number of pseudorandom characters, and then disconnecting the call. Both user/system interfaces are monitored (see Figure 2). At each step of the access, transfer, and disengagement functions, the relevant events are recorded and time-stamped. Files of these time-stamped events along with copies of the transferred data are stored in both user's computers. During data reduction, the "speed" parameters are estimated from the time-stamped event files and the "accuracy" and "reliability" parameters are estimated by comparing the source and destination copies of the pseudorandom data. Although any suitable method can be used to implement this measurement process, this report assumes that the NTIA methods will be used.

This report is divided into several major sections. Section 2 presents an overview of the system performance measurement standards, ANS X3.102 and ANS X3.141. Section 3 discusses the objectives of the experiment, the appropriate analysis, and why the most important performance parameters must be

---

[1]Structured design diagrams show activities in ovals and input/output in rectangles. Arrows indicate the flow of activity and information without regard to time.

[2]This six volume report shows how NTIA implements the standards. The enabling software is a set of routines written in C and FORTRAN programming languages. The source code consists of more then four megabytes and is available on a set of seven PC-DOS™ and MS-DOS™ 5-1/4" diskettes. It operates under the UNIX™ system V.3 operating system with U.C. Berkeley, C shell enhancements.

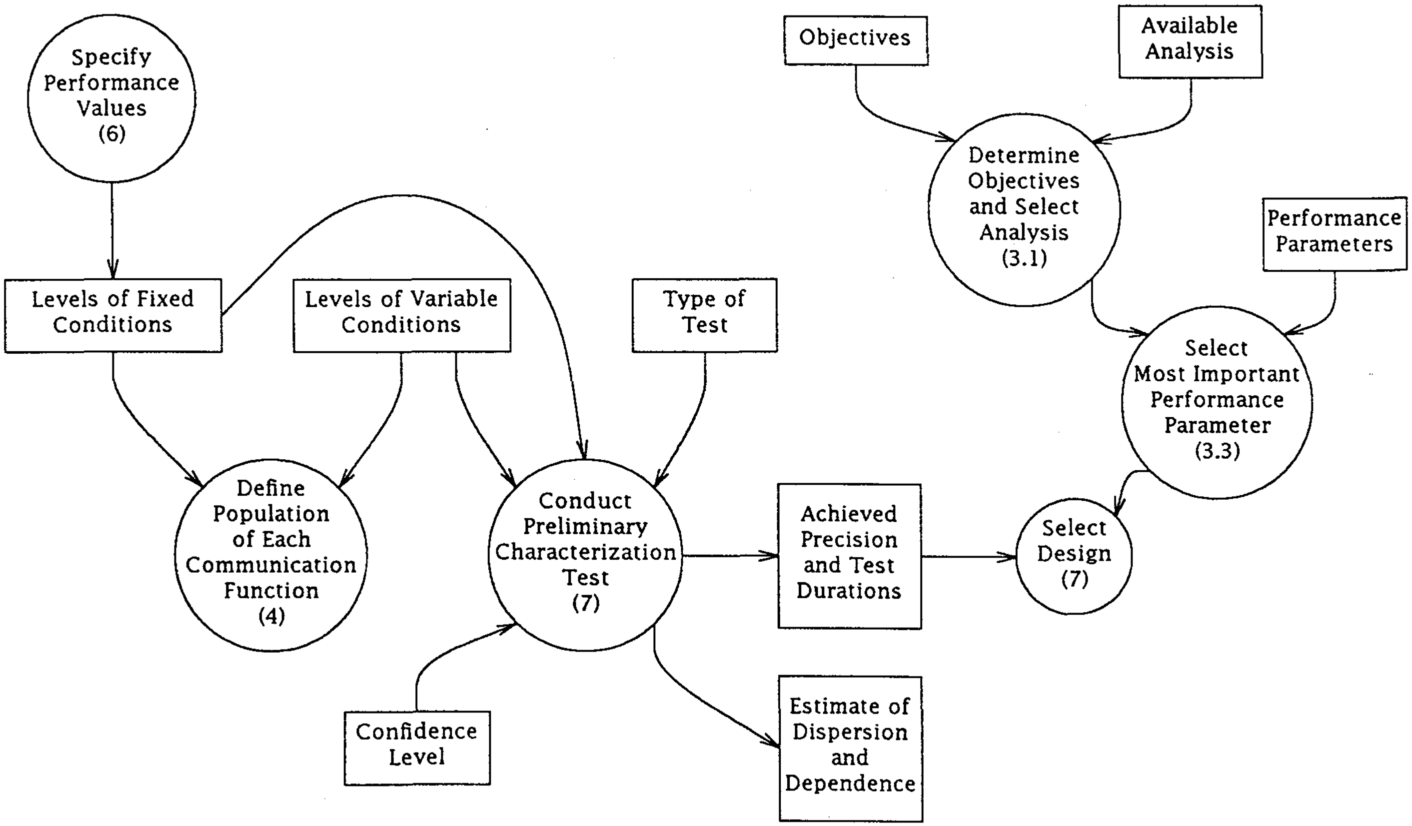


Figure 1. Structured design diagram showing the relationship among the major activities of experiment design.

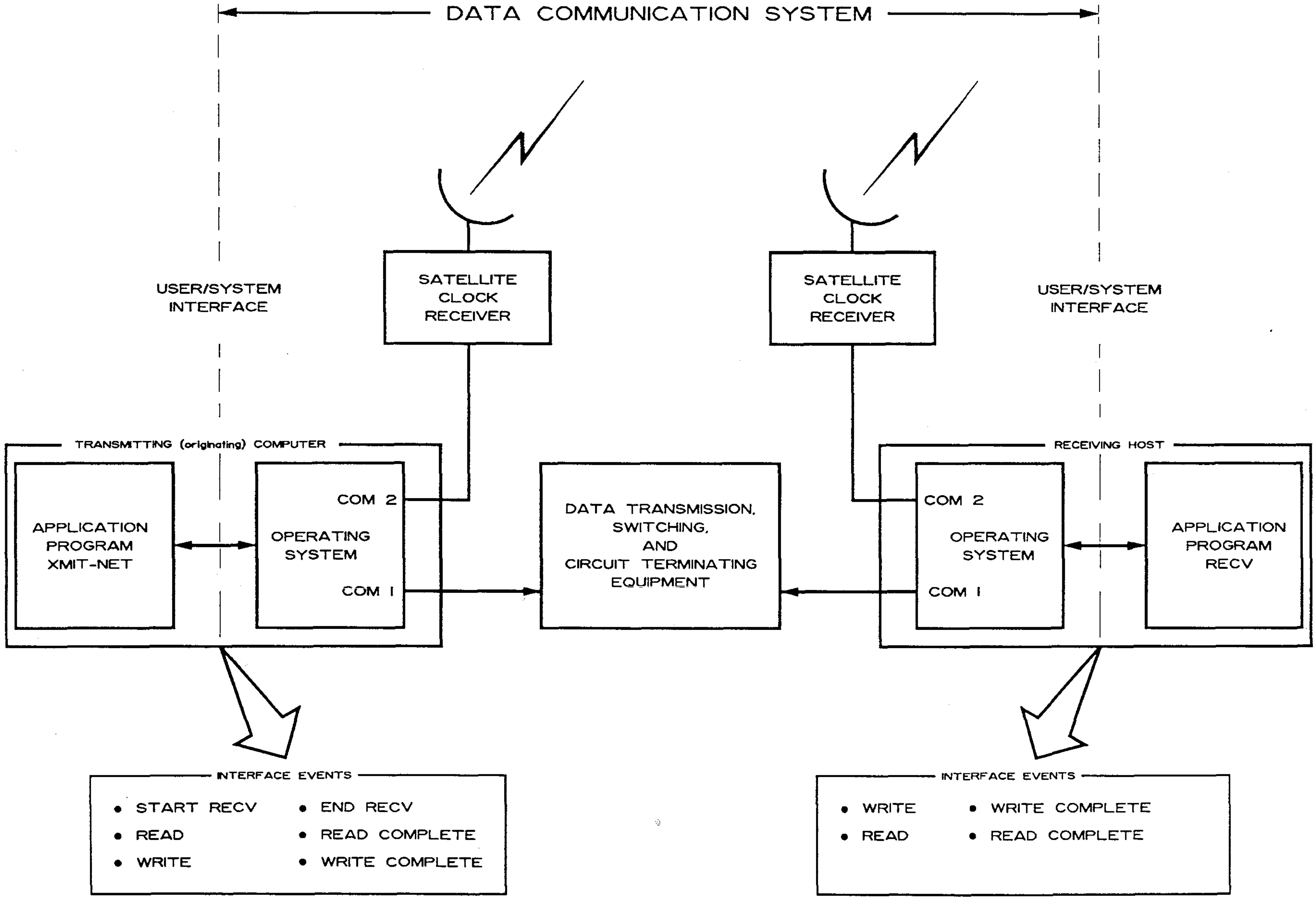


Figure 2. Schematic diagram of a data communication system and end users.

selected. Section 4 shows how the population of each communication function is defined. Section 5 discusses a few possible experiment designs. Section 6 explains how to specify performance values for the performance parameters. Section 7 shows how to conduct a preliminary characterization test, and Section 8 is a summary of the important topics of this report. There are also two appendices: Appendix A describes the NTIA software that implements ANS X3.141, and Appendix B shows how to create specification files.

2. SYSTEM PERFORMANCE MEASUREMENT STANDARDS

In the past, members of the data communication community have had difficulty determining the most suitable data communication system for their use. To help solve this problem, ANSI approved the two data communication systems performance standards:

- The American National Standard for Information Systems - Data Communication Systems and Services - User-oriented Performance Parameters, ANS X3.102 (ANSI, 1983).
- The American National Standard for Information Systems - Data Communication Systems and Services - Measurement Methods for User-oriented Performance Evaluation, ANS X3.141 (ANSI, 1987).

These two standards form the basis of the "functional approach" to data communication system procurement described in Seitz and Grubb (1983).

The functional approach and the ANSI standards provide many benefits to users, designers, and experimenters:

- Specifications of data communication systems are precisely defined for users; they don't need to become system designers and they can specify the system that best fits their needs and budgets without being constrained to a particular design.
- Data communication system designers and service providers can better assess existing and proposed services from the user's perspective, allowing them to identify areas for improvement and/or cost reduction without sacrificing system performance.
- Data communication system experimenters and researchers now have the methods and tools for conducting and analyzing repeatable data communication system performance measurements on experimental systems such as ACTS.

2.1 Data Communication Process

As defined in the ANSI standard, ANS X3.102 (ANSI, 1983), the data communication process involves a data communication system, end users, user/system interfaces, interface events, transferred information, the data communication session, the primary data communication functions, and performance parameters.

2.1.1 Data Communication System

A data communication system is a collection of transmission facilities, switches (for routing), data terminals (for input/output), and protocols (for access and disengagement). It includes all functional and physical elements that participate in transferring information between two or more end users. Figure 2 is a schematic diagram of a data communication system, user/system interfaces that join the computer operating systems and the end users, interface events, and satellite clock receivers.

The satellite clock receiver is a feature of the NTIA measurement system. It receives the National Institute of Standards and Technology (NIST) time standard information from the Geostationary Operational Environmental Satellite (GOES), thus providing a common time reference for both the transmitting and receiving users in the NTIA data communication measurement system. This time information identifies when performance-significant interface events occur (see Section 2.1.4.)

Figure 3 shows the ACTS low burst rate (LBR) system configuration for system performance measurement experiments between a single user pair. During the experiments both users have 80386-class personal computers running copies of the NTIA software described in Appendix A. A detailed description of the ACTS LBR communication system is given by Naderi and Campanella (1988).

2.1.2 End Users

An end user may be the operator of a terminal or an application program that simulates an operator.[3] The NTIA implementation uses an application program as each end user.

2.1.3 User/System Interfaces

An interface is the boundary between two entities. In a data communication system the two entities are the end users and the system. There is an interface between the source end user and the system, and an interface between the destination end user and the system.

2.1.4 Interface Events

An interface event occurs when a discrete transfer of information crosses a user/system interface. The protocols of the equipment, the operating systems, and the network, determine the nature and occurrence of interface events. The

[3]The data medium employed by the end user does not affect the definition of an end user.

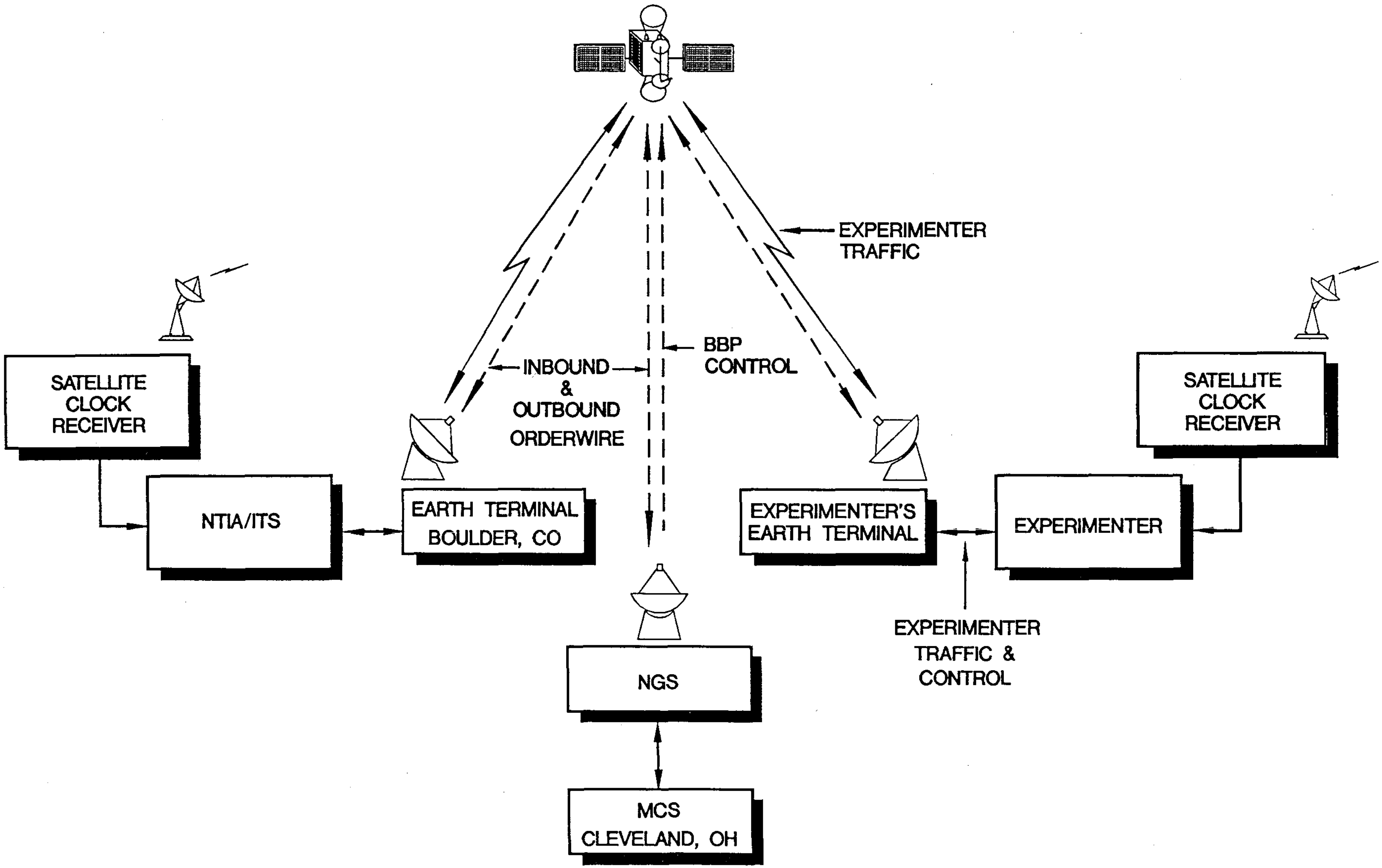


Figure 3. Schematic diagram of the ACTS system configuration between a single user pair.

events are monitored at each interface for their time of occurrence. Table 1 lists examples of interface events.

Table 1. Examples of User/System Interface Events

Activities	No Data Medium	Data Medium
Physical Interface: Human Operator Using a Data Terminal	• Keying Chars. • Printing Chars. • Displaying Chars.	• Reading Cards • Punching Cards
Functional Interface: Application Program Using a Computer Operating System	• Issuing Operator System Calls • Setting Flags • Clearing Flags	• Reading Magnetic Tapes/Disks

2.1.5 Transferred Information

There are two types of transferred information: user information and overhead information. User information is information transferred from a source user to the system that is intended to cross the system/destination user interface. Overhead information is all information that is not user information. There are three types of overhead information:

- System Control Information. This is information transferred from a user to the system for system control. Examples are party address digits in public switched networks, American Standard for Communication and Information Interface (ASCII) characters (such as DLE, ESC, and ENQ), and Boudot characters (such as "letter shift" and "figures shift").

- Status Report Information. This is information transferred from the system to a user for reporting status or to control user operation. Examples are circuit busy and ringing signals in the public switched network.

- Inter-system Coordination Information. This is information transferred between elements of a system to coordinate their operation. Examples are parity bits and ASCII characters (such as SYN, ACK, and NAK).

2.1.6 Data Communication Session

A data communication session is a coordinated sequence of user and system events to transfer user information (from one or more source users to one or more destination users). For each pair of end users, a data communication session can be described by a session profile - a diagram of the flow of transferred information among the participating entities.

2.1.7 Primary Communication Functions

The primary functions in a data communication session are access, user information transfer, source disengagement, and destination disengagement. For simplicity, the standard defines these functions for pairs of end users, but the definitions can be extended for more than two end users.[4] In connection-oriented services these functions correspond to connection, user information transfer, and disconnection, but they are also applicable to connectionless services (e.g., electronic mail).

A. Access

This function comprises those activities that the users and the system must accomplish for the system to accept source user information.

B. User Information Transfer

This function comprises those activities that the accessed system must accomplish to transfer user information (from the source user, through the system, and to the destination user). This function could be divided into two functions, one for bits and one for blocks.

---

[4]The NTIA measurement system is designed to collect and process performance data from sessions involving one-way (i.e., simplex) transmission between a single pair of users. Sessions can involve either two-way (duplex) transmission or multiple pairs if the on-line data extraction procedures are modified:

- Each session is treated as the superposition of two simplex sessions.
- A given user acts as the source in one session and as the destination in the other.
- Each interface monitor generates one set of extracted data files in which the local user acts as the source and another set in which the local user acts as the destination.
- These sets of extracted data are input to separate data conversion runs to produce two performance sets of data - one for each direction of transmission.
- The two sets of data are then reduced and analyzed in the usual manner.

C. **Source Disengagement**

This function comprises those activities that terminate the session for the source end user.

D. **Destination Disengagement**

This function comprises those activities that terminate the session for the destination end user.

2.1.8 Reference Events

Interface events are generally system-dependent, and not useful in developing a standard. However, interface events that perform similar functions (i.e., have common performance-significance) common to all data communication systems, are referred to as system-independent interface events. These interface events are called reference events. Primary reference events are associated with primary performance parameters and ancillary reference events with ancillary performance parameters.

To demonstrate the difference between interface events and reference events, consider two examples from the access function - one from a public switched telephone network and one from a packet-switched network. The request for service in a public switched telephone network can be indicated by removing the handset (an interface event occurs when a bit is transferred from the user to the system). The request for service in a packet-switching network can be indicated, for example, by typing **connect** at a data terminal (an interface event occurs when this character string is transferred from the user to the system). These two examples illustrate that interface events requesting service are system-dependent. All interface events, such as these, that communicate a user's request for data communication service (i.e., have common performance-significance) are grouped in the category called Access Request. Access Request is the reference event that defines the beginning of the access function.

A. **Primary Reference Events**

The standard defines nine primary reference events associated with the 19 primary performance parameters. They are listed in Table 2 along with their effect on the system, and their performance significance.

Table 2. Primary Reference Events

PRIMARY REFERENCE EVENTS

FUNCTION	REFERENCE EVENT	SYSTEM EFFECT	PERFORMANCE SIGNIFICANCE
ACCESS	1. ACCESS REQUEST	REQUESTS INITIATION OF DATA COMMUNICATION SESSION AND COMMITS THE ORIGINATING USER TO PARTICIPATE	BEGINS ACCESS FUNCTION. STARTS THE COUNTING OF ACCESS TIME
	2. NONORIGINATING USER COMMITMENT	IN A CONNECTION-ORIENTED DATA COMMUNICATION SESSION, INDICATES NONORIGINATING (CALLED) USER WILLINGNESS TO PARTICIPATE	ELIMINATES INCORRECT ACCESS AS A POSSIBLE ACCESS OUTCOME
	3. SYSTEM BLOCKING SIGNAL	NOTIFIES ORIGINATING USER THAT THE SYSTEM CANNOT SUPPORT A REQUESTED DATA COMMUNICATION SESSION	IDENTIFIES ACCESS ATTEMPT OUTCOME AS ACCESS DENIAL
	4. USER BLOCKING SIGNAL	NOTIFIES SYSTEM THAT THE ISSUING USER WILL NOT SUPPORT A REQUESTED DATA COMMUNICATION SESSION	IDENTIFIES ACCESS ATTEMPT OUTCOME AS USER BLOCKING (EXCLUDED FROM SYSTEM PERFORMANCE MEASUREMENT)
USER INFORMATION TRANSFER	5. START OF BLOCK INPUT TO SYSTEM	TRANSFERS ONE OR MORE BITS AT BEGINNING OF USER INFORMATION BLOCK FROM SOURCE USER TO SYSTEM	WHEN BLOCK IS THE FIRST BLOCK IN A DATA COMMUNICATION SESSION (AFTER NONORIGINATING USER COMMITTMENT IN CONNECTION-ORIENTED SESSIONS), COMPLETES ACCESS FUNCTION AND BEGINS USER INFORMATION TRANSFER. STOPS THE COUNTING OF ACCESS TIME
	6. START OF BLOCK TRANSFER	AUTHORIZES THE SYSTEM TO TRANSMIT A GIVEN USER INFORMATION BLOCK	(1) BEGINS BLOCK TRANSFER FUNCTION AND STARTS THE COUNTING OF BLOCK TRANSFER TIME. (2) WHEN BLOCK PRECEDES THE FIRST BLOCK IN A TRANSFER SAMPLE, BEGINS COLLECTION OF THE SAMPLE AND STARTS THE COUNTING OF SAMPLE INPUT TIME. (3) WHEN BLOCK IS THE LAST BLOCK IN A TRANSFER SAMPLE, COMPLETES INPUT OF SAMPLE AND STOPS THE COUNTING OF SAMPLE INPUT TIME.
	7. END OF BLOCK TRANSFER	TRANSFERS A GIVEN USER INFORMATION BLOCK TO THE DESIGNATION USER, WITH APPROPRIATE NOTIFICATION TO THAT USER WHERE REQUIRED	(1) COMPLETES BLOCK TRANSFER FUNCTION AND STOPS THE COUNTING OF BLOCK TRANSFER TIME. (2) WHEN BLOCK PRECEDES THE FIRST BLOCK IN A TRANSFER SAMPLE, BEGINS OUTPUT OF THE SAMPLE AND STARTS THE COUNTING OF SAMPLE OUTPUT TIME. (3) WHEN BLOCK IS THE LAST BLOCK IN A TRANSFER SAMPLE, COMPLETES COLLECTION OF THE SAMPLE AND STOPS THE COUNTING OF SAMPLE OUTPUT TIME.
DISENGAGEMENT	8. DISENGAGEMENT REQUEST	REQUESTS TERMINATION OF A USER'S PARTICIPATION IN A DATA COMMUNICATION SESSION	BEGINS DISENGAGEMENT FUNCTION. STARTS THE COUNTING OF DISENGAGEMENT TIME.
	9. DISENGAGEMENT CONFIRMATION	CONFIRMS TERMINATION OF A USER'S PARTICIPATION IN A DATA COMMUNICATION SESSION	COMPLETES DISENGAGEMENT FUNCTION. STOPS THE COUNTING OF DISENGAGEMENT TIME.

B. **Ancillary Reference Events**

Ancillary reference events identify the entity (system or user) responsible for the next interface event. An interface event may have any one of three effects at the local user/system interface:

- It may leave the system responsible for generating the next event (e.g., a calling user's off-hook action leaves the system responsible for the dial tone at the calling interface.)
- It may leave the user responsible for generating the next event (e.g., a system's issuance of a dial tone leaves the user responsible for dialing.)
- It may leave responsibility for generating the next event undefined, because the next event will occur at the remote interface (e.g., when the user requests a packet, the next event occurs at the remote interface.)

The six combinations of responsibility for each entity at a local interface are numbered in Table 3. The ancillary reference events define 5 ancillary performance parameters.

Table 3. Ancillary Reference Events

		Local Interface Responsibility		
		System	User	Undefined
Remote Interface Responsibility	None	1	2	3
	System	4	5	6

Sometimes a reference event can signify both a primary and an ancillary reference event: for example, when a calling user issues an X.25 Call Request packet, the interface event is an Access Request and both the user and the system are relieved of responsibility at the local interface.

2.2 Performance Parameters: ANS X3.102

ANS X3.102 defines 24 parameters that describe the performance of data communication systems and services from the end user's point of view, regardless of protocol or architecture; they are system-independent. Performance parameters are divided into two categories, primary and ancillary.

Primary parameters measure speed, accuracy, and reliability. Speed parameters are estimated by the time-stamped reference events. Accuracy and reliability parameters are generally estimated by comparing pseudorandom data files from the source and destination.

Ancillary parameters measure the degree to which the primary speed parameters are influenced by the user (rather than by the system). That is, they are the fraction of user delay-to-total delay:

$$(\text{user delay}) / (\text{user delay} + \text{system delay}).$$

Table 4 lists the 24 ANS X3.102 performance parameters.[5] Ancillary parameters are shown in italics. The list is organized according to the four primary communication functions and the performance criteria that most frequently concern users: speed, accuracy, and reliability.

2.3 Measurement Methods: ANS X3.141

ANS X3.141 specifies uniform methods of measuring the 24 ANS X3.102 performance parameters. The measurement methods are general, system-independent, and may be used to obtain parameter estimates at any pair of digital interfaces connecting a data communication system to its users. They require four activities: experiment design, data extraction, data reduction, and data analysis. Figure 4 is a structured design diagram of these four activities and an additional activity that displays the data for visual analysis. The dashed line in the diagram indicates that examination of the analyzed data might prompt redesign of the experiment. Each activity is briefly described below:

- The steps of experiment design are the subject of this report.
- Data extraction involves recording primary and ancillary reference events and their time of occurrence. It also involves recording the content of transferred and received data.
- Data reduction transforms the performance data from the data extraction process into performance parameter estimates.

[5]Bit and Block Misdelivery are optional parameters. They are not part of the NTIA implementation.

Table 4. ANS X3.102 Performance Parameters

COMMUNICATION FUNCTION	PERFORMANCE CRITERION: SPEED	PERFORMANCE CRITERION: ACCURACY	PERFORMANCE CRITERION: RELIABILITY
ACCESS	• ACCESS TIME • *USER FRACTION OF ACCESS TIME*	• INCORRECT ACCESS PROBABILITY	• ACCESS DENIAL PROBABILITY • ACCESS OUTAGE PROBABILITY
USER INFORMATION TRANSFER	• BLOCK TRANSFER TIME • *USER FRACTION OF BLOCK TRANSFER TIME* • USER INFORMATION BIT TRANSFER RATE • *USER FRACTION OF INPUT/OUTPUT TIME*	• BIT ERROR PROBABILITY • BIT MISDELIVERY PROBABILITY • EXTRA BIT PROBABILITY • BLOCK ERROR PROBABILITY • BLOCK MISDELIVERY PROBABILITY • EXTRA BLOCK PROBABILITY	• BIT LOSS PROBABILITY • BLOCK LOSS PROBABILITY
		• TRANSFER DENIAL PROBABILITY	
SOURCE DISENGAGEMENT	• SOURCE DISENGAGEMENT TIME • *USER FRACTION OF SOURCE DISENGAGEMENT TIME*	• SOURCE DISENGAGEMENT DENIAL PROBABILITY	
DESTINATION DISENGAGEMENT	• DESTINATION DISENGAGEMENT TIME • *USER FRACTION OF DESTINATION DISENGAGEMENT TIME*	• DESTINATION DISENGAGEMENT DENIAL PROBABILITY	

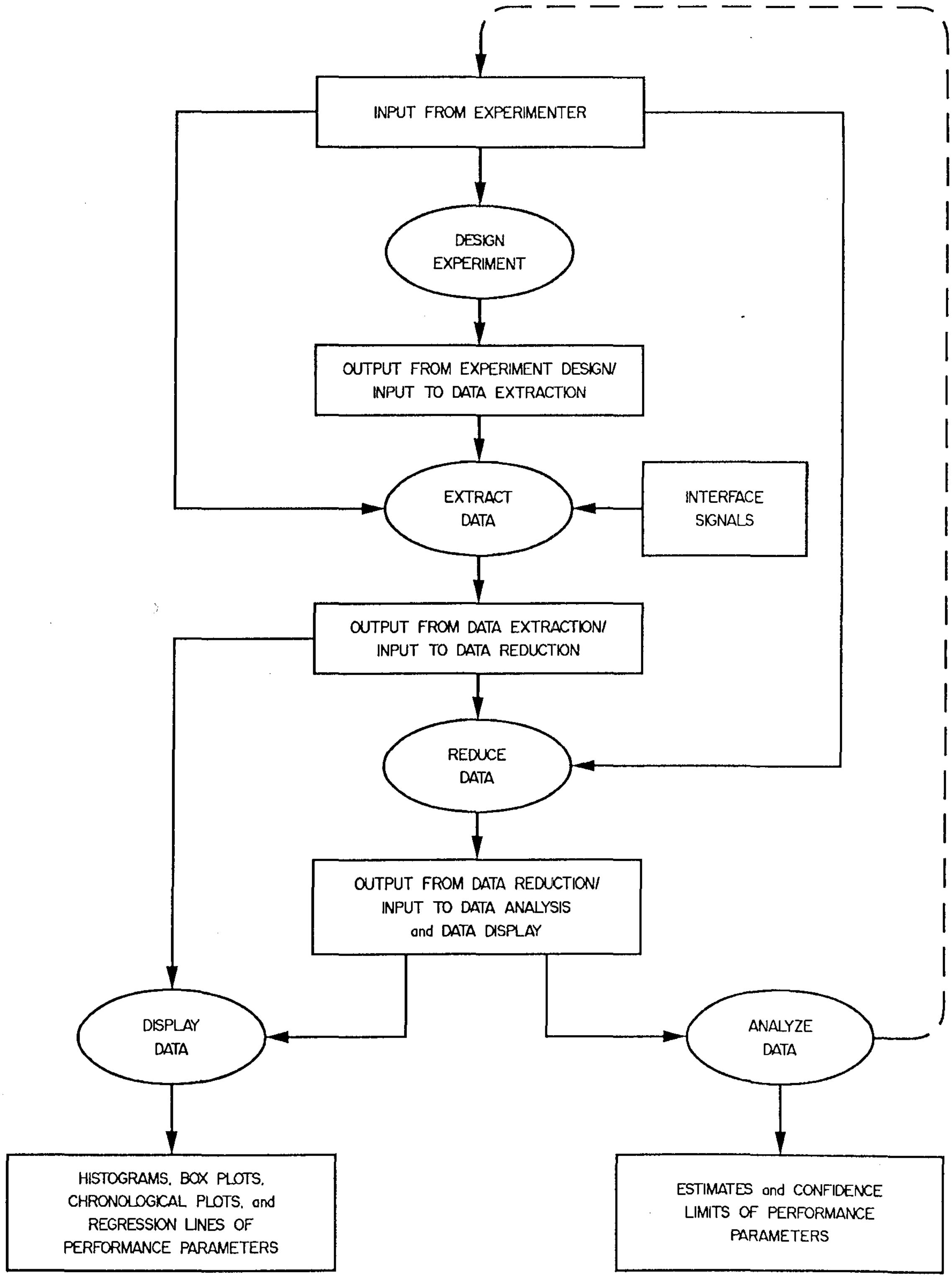


Figure 4. Structured design diagram of ANS X3.141 measurement method.

- Data analysis provides

  - estimates and confidence limits of performance parameters (from either single or multiple tests),
  - tests of acceptance for performance parameters,
  - comparison tests for performance parameters, and
  - tests to determine if a variable condition is a factor (i.e., whether it affects a parameter value).

- Data display draws histograms, box plots (abbreviated histograms), chronological plots, and regression plots.

The NTIA software structure is discussed in Appendix A.

The system performance experiments NTIA is proposing for the ACTS LBR system is very similar to data communication system experiments NTIA conducted in 1983. The intent is to gather performance information in order to obtain performance parameter estimates similar to those shown in Table 5. In 1983, NTIA conducted an extensive experiment to estimate performance parameters of data communication systems according to ANS X3.141 specifications (Spies et al., 1988). Two microcomputers were assembled: one emulated a network-accessible host computer, and the other emulated a remote terminal. This experiment collected performance data between remote terminal users and host application programs via three public data networks and two direct distance dial networks (Spies et al., 1988). Table 5 lists the estimates of the parameters and their 95% confidence limits for a public data network (labelled "Network A"). Data obtained from tests under a variety of conditions (including 1-3 cities) were pooled to obtain these estimates. Briefly, the third footnote in each table refers to pooling data according to the results of significance tests. Although networks and operating conditions during future experiments will certainly differ from these, knowledge of these estimates and their achieved precisions might contribute to enhancing future experiment design.[6]

---

[6]Precision can be measured by the length of the confidence interval about an estimate; a short confidence interval is related to greater precision. Precision is usually greater with a larger sample size.

Table 5. Estimates and 95% Confidence Limits for Performance Parameters for a Public Data Network

Performance Parameter Summary For PDN A Connections

Performance Parameter	*95% Lower Limit*	*Mean Estimate*	*95% Upper Limit*	*Pooling Disposition***	*Number of Cities*	*Number of Tests*	*Number of Trials*
Access Time (s)	40.7	41.8	42.9	2	3	11	209
User Fraction of Access Time	0.034	0.036	0.037	2	3	11	209
Incorrect Access Probability	0	0	0.062*	1	3	11	220
Access Outage Probability	0	0	0.062*	1	3	11	220
Access Denial Probability	0.018	0.050	0.107	1	3	11	220
Block Transfer Time (s)	3.61	3.79	3.97	2	2	7	559
User Fraction of Block Transfer Time	0.077	0.089	0.102	2	2	7	559
User Fraction of Input/Output Time	0.110	0.214	0.317	3	2	7	7
User Information Bit Transfer Rate (bps)	421	814	1207	3	2	7	7
Bit Error Probability	6.0×10^{-7}	7.0×10^{-6}	3.0×10^{-5}	1	2	7	573440
Bit Misdelivery Probability	-	-	-	-	-	-	-
Extra Bit Probability	0	0	3.0×10^{-5}*	1	2	7	573440
Bit Loss Probability	0	0	3.0×10^{-5}*	1	2	7	573440
Block Error Probability	0	2.0×10^{-3}	3.0×10^{-2}	1	2	7	560
Block Misdelivery Probability	-	-	-	-	-	-	-
Extra Block Probability	0	0	3.0×10^{-2}*	1	2	7	560
Block Loss Probability	0	0	3.0×10^{-2}*	1	2	7	560
Transfer Denial Probability	0	0	5.0×10^{-2}*	1	2	7	273
Source Disengagement Time (s)	14.3	15.1	15.8	2	3	11	194
User Fraction of Source Disengagement Time	0.058	0.061	0.065	2	3	11	194
Source Disengagement Denial Probability	0.042	0.072	0.116	1	3	11	209
Destination Disengagement Time (s)	4.9	5.2	5.4	1	3	11	207
User Fraction of Destination Disengagement Time	0.121	0.128	0.134	1	3	11	207
Destination Disengagement Denial Probability	0.008	0.018	0.033	2	3	11	209

* Conditional probability assumed to be 0.8
\- Not measured

** 1 means no significant difference among tests or cities, so all trials pooled.
2 means no significant difference among cities, so all test means pooled.
3 means significant difference among cities, so no pooling; only 1 or 2 degrees of freedom for confidence limit.

3. EXPERIMENT OBJECTIVE AND RECOMMENDED ANALYSES

Usually, the two principals most interested in the results of an experiment are the user and the vendor. The objective of the experiment for each party should be clearly understood before the experiment is designed. Then, the economic benefits of the system should be matched with the scope and precision of the experiment.

3.1 Experiment Objectives

Some common objectives of a data communication system performance experiment are:

- acceptance,
- characterization (quantification),
- maintenance,
- selection,
- design,
- optimization, and
- management.

The primary objective of the proposed ACTS system performance experiment is to characterize the service provided by the satellite. Characterization will allow quantification of the ACTS communication system performance. The improvements brought about by the advanced technology can be assessed and areas of improvement can be identified for use in future advanced communication satellite systems. Additionally, efficient control protocols could be designed and evaluated for new generations of switched satellite systems. Characterization of the ACTS system will also allow performance comparisons of advanced satellites with other telecommunication technologies.

3.2 Recommended Statistical Analysis

ANS X3.141 recommends one (or more) of four analyses of the ANS X3.102 performance parameters. These analyses are:

- estimation of a performance parameter,
- a test to determine if a performance parameter is acceptable,
- a test to compare a performance parameter estimated from two systems/services, and
- a test to determine whether (and how) a variable condition affects a performance parameter (i.e., whether it is a factor).

The selection of these analyses depends upon the objective of the experiment. Table 6 matches some common experiment objectives and plausible analyses. The "√" indicates that the objective can be analyzed by that method. The analyses must be selected before one can judiciously choose the number of levels of the variable conditions (as in Section 4.2.).

Table 6. Common Experiment Objectives and Plausible Analyses

	PLAUSIBLE ANALYSES			
EXPERIMENT OBJECTIVE	Estimation	Acceptance	Comparison	Conditions
Acceptance		√		
Characterization	√			
Maintenance		√		
Selection			√	
Design	√	√	√	√
Optimization				√
Management/Other	√	√	√	√

Because the system/service is to be characterized, estimation (over the levels of conditions that will probably exist) is the preferred analysis for this experiment. A performance parameter and its confidence limits are estimated from either a single test (conducted under a single combination of levels of variable conditions) or pooled data from multiple tests (conducted under multiple combinations of levels of variable conditions). By virtue of the larger number of trials from multiple tests, pooled data might provide a good overall estimate. Moreover, data from multiple tests can be used to determine which variable conditions do not effect the system/service and which do (i.e., which are factors).

3.3 Most Important Performance Parameters

The suitability of the ANSI standards for quantifying communication satellite system performance is discussed in Cass (1988), and performance parameters that are significant for the various classes of satellite systems are selected and justified. Table 7 summarizes this selection and indicates the appropriate parameters for various satellite systems by a "√"[7]. The ACTS is represented by the DAMA satellite systems.

Table 7. Significant Performance Parameters for Satellite Systems

PERFORMANCE PARAMETERS	TYPE A & TYPE B	TDMA	SS/TDMA	DAMA
Access Time				√
User Fraction (UF) of Access Time				√
Incorrect Access Probability				
Access Denial Probability				√
Access Outage Probability				√
Block Transfer Time	√	√	√	√
UF of Block Transfer Time				
User Information Bit Transfer Rate	√	√	√	√
UF of Input/Output Rate	√	√	√	√
Bit Error Probability	√	√	√	√
Bit Misdelivery Probability				
Bit Loss Probability		√	√	√
Extra Bit Probability		√	√	√
Block Error Probability	√	√	√	√
Block Misdelivery Probability				
Extra Block Probability		√	√	√
Block Loss Probability		√	√	√
Transfer Denial Probability	√	√	√	√
Source Disengagement Time				√
UF of Source Disengagement Time				√
Source Disengagement Denial Prob.				
Dest. Disengagement Time				√
UF of Dest. Disengagement Time				√
Dest. Disengagement Denial Prob.				

The NTIA software is designed to analyze two types of tests, depending upon the communication functions: access-disengagement tests and user information transfer tests. The experimenter must determine which performance

[7]Subsequent to the publication of Cass (1988), the disengagement function has been considered to be two functions: Source Disengagement and Destination Disengagement.

parameter is the most important performance parameter from each type of test. Then each test should achieve the precision specified for that performance parameter. Other performance parameters will be estimated with a precision that is inversely proportional to the dispersion of their populations - a precision that may be greater or less than that specified for the most important performance parameter.

For the proposed ACTS system performance experiment, Access Time is the most important performance parameter for access-disengagement tests, and Bit Error Probability and User Information Bit Transfer Rate are the two most important performance parameters for user information transfer tests. Note, however, that each user information transfer test produces only one trial for User Information Bit Transfer Rate and User Fraction of Input/Output Time. These performance parameters should not be selected as the most important parameters since the single trial provides no estimate of precision - and a precision estimate is a requirement of the standard. Therefore, Bit Error Probability remains the most important performance parameter for user information transfer tests for the ACTS system.

4. POPULATION OF EACH COMMUNICATION FUNCTION

A data communication session has four functions: Access, User Information Transfer, Source Disengagement, and Destination Disengagement. The set of all conceivable attempts to complete each function is a population (e.g., each access attempt is an element of the hypothetical population of access attempts).

The performance parameters of each function are characteristics of the population. For example, the performance parameter, Access Denial Probability, is a characteristic of the hypothetical population of access attempts. Each communication function performance parameter is considered a random variable and has a distribution, the goal is to estimate the mean of each distribution with a specified precision and confidence level.

The distributions are defined by a set of existing conditions. For our purposes, it is convenient to divide the set of conditions into those that are fixed and those that are variable - as far as the experiment is concerned. The fixed conditions are those for which there are either no options (such as a feature supplied by a vendor) or a single level that the experimenter will choose (such as the block size). The variable conditions have more than one level (value), providing an opportunity to design an experiment; the experiment is conducted over all or selected combinations of levels of the variable conditions.

In this section, conditions existing in this type of experiment will be discussed in a general manner. In Section 5, levels and combinations of levels of variable conditions will be selected - as well as the design.

4.1 Fixed Conditions

Many conditions of an experiment are fixed, and should be identified. They either cannot be changed or will not be changed for this experiment. The following conditions are fixed for both types of tests in experiments of this kind:

- Source Site: The source site will be fixed.

- Source End User: The source end user is an application program provided by NTIA. It will perform end user activities and record the nature and time of interface signals in the source (transmitting) computer.

- Destination End User: The destination end user is an application program provided by NTIA. It will perform functions analogous to those at the source site.

Other fixed conditions are listed below according to the type of test.

4.1.1 Access-Disengagement Tests

The following conditions are fixed during access-disengagement tests.

- Measurement Points: Measurement points are located at interfaces where reference events occur.

- Reference Events: A reference event corresponds to a system-independent interface event.

- Specified Values: Certain values must be specified. For example, if an observation of a delay parameter exceeds three times its specified value, it is classified as a failure. This concept and corresponding values are specified in Section 6. The performance parameters requiring a threshold are:

    - Access Time
    - User Fraction of Access Time
    - Source Disengagement Time
    - User Fraction of Disengagement Time
    - Destination Disengagement Time
    - User Fraction of Destination Disengagement Time

- Intersession Delay: This delay between sessions (access attempts) should be non-zero to attenuate dependence between trials.[8] The intersession delay can be between 0 and 99 s; however, a longer delay will tend to attenuate intersession dependence. Select a single intersession delay estimated from a preliminary characterization test.

- Block Size: The block size is the minimum size, 64 bytes, because the block size is irrelevant to access-disengagement tests.

- Number of Blocks: Because the number of blocks is irrelevant to access-disengagement tests, the number of blocks is assumed to be one.

---

[8]Roughly speaking, the delay between consecutive sessions begins with a Disengagement Confirmation signal and ends with the Access Request on the following trial.

4.1.2 User Information Transfer Tests

The following conditions are fixed for user information transfer tests.

- Measurement Points: Measurement points are located at interfaces where reference events occur.

- Reference Events: A reference event corresponds to a system-independent interface event.

- Specified Values: Certain values are specified. They serve a purpose analogous to that in access-disengagement tests. The performance parameters that require a threshold are:

  - Block Transfer Time
  - User Fraction of Block Transfer Time
  - Bit Error Probability (for the Transfer Sample)[9]
  - Bit Loss Probability (for the Transfer Sample)
  - Extra Bit Probability (for the Transfer Sample)
  - User Information Bit Transfer Rate (for the Transfer Sample)

- Interblock Delay: The delay between transferred blocks should be positive to attenuate dependence between consecutive blocks. However, the interblock delay should be set at 0 s if this is the delay that will exist during normal operation of the network. Select a single interblock delay.

- Block Size: The block size is assumed to be constant. NTIA software is designed for block sizes between 64 and 512 bytes.

- Access Attempts: The number of access attempts is assumed to be one.

4.1.3 Session Profile

A session profile is a diagram that defines the event sequence at the monitored interfaces during a successful session and the service refusal/interrupt sequence allowed by the system. The event sequence is determined by the protocol. The protocol can be determined from equipment manuals, protocol analyzers, and experts. A session profile should be drawn for each network to be tested.

[9]The Transfer Sample is defined in Section 6.2.

4.2 Variable Conditions

Variable conditions are conditions having more than one value - called levels[10]. It is the variable conditions that influence the design of the experiment. Controlled variable conditions are conditions whose levels can be selected by the experimenter. A variable condition might be a factor for one performance parameter but not for another. Moreover, a performance parameter may be affected by more than one factor. When feasible, it is desirable to replicate tests under each combination of levels of variable conditions to possibly reveal unknown factors.[11] Variable conditions can be divided into two types, primary and extraneous.

Primary variable conditions are those whose effects are of interest; they are thought to have a direct effect upon the value of a performance parameter. Primary variable conditions can be quantitative (as is the transmission rate) or qualitative (as is the error control that may be provided). Moreover, qualitative variable conditions can be of two types: those having a few, known levels (called "fixed effects" or Type I variables) or those having randomly selected levels (called "random effects" or Type II variables). Service and Traffic are two primary variable conditions that might affect both types of tests. Service can be controlled, but Traffic cannot be controlled.

Extraneous variable conditions are those whose effects are not of interest; they are considered to be "nuisance" variable conditions. Time of Day, and Day of Week are examples of (controllable) extraneous variable conditions.[12] In this experiment, Traffic is a primary variable condition and a function of

---

[10]If a condition can have more than one value, but has only one value <u>in this experiment</u>, it is considered to be a fixed condition.

[11]Replication is implicitly required by ANS X3.141: the experimenter must state the precision of the estimate of the performance parameter, and precision requires more than one observation.

[12]Of course, time itself cannot affect an experiment, but events occurring during time may. Listing Time of Day and Day of Week as conditions is a tacit admission that unknown or unknowable events may occur during time that could affect the experiment.

time. Therefore, Time of Day and Day of Week will not appear (explicitly) as extraneous variable conditions.[13]

Propagation Path is a controlled extraneous variable condition that might affect both types of tests. Possible Propagation Paths for the satellite portion of the network are shown in Figure 5. There are two sectors (called A and B). Sector A has two spot-beam dwells (called a and b) and sector B has one spot-beam dwell (called a). It includes the three possible combinations of sectors and spot-beam dwells:

- P_1: from (A, a) to ACTS and from ACTS to (A, a),
- P_2: from (A, a) to ACTS and from ACTS to (A, b), and
- P_3: from (A, a) to ACTS and from ACTS to (B, a).

The experiment will include one source end user and one destination end user within each spot-beam dwell (i.e., there will be only three Propagation Paths).

4.2.1 Access-Disengagement Tests

Table 8 is a 2x2 matrix of variable conditions existing during access-disengagement tests. They are listed according to type (primary or extraneous) and the state of control.

Table 8. Variable Conditions for Access-Disengagement Parameters Listed According to Their Type and Control Status

	Controlled	Non-Controlled
Primary	• Service	• Traffic
Extraneous	• Propagation Path	-

[13]Day of the Week is a variable condition whose levels are computed by a shell script from calendar data. Time (period) of Day is a variable condition whose levels are computed by a shell script from the computer clock. There are six periods, each having four hours. Levels of the variable condition, Traffic, are obtained from selected combinations of Day of Week and Time of Day.

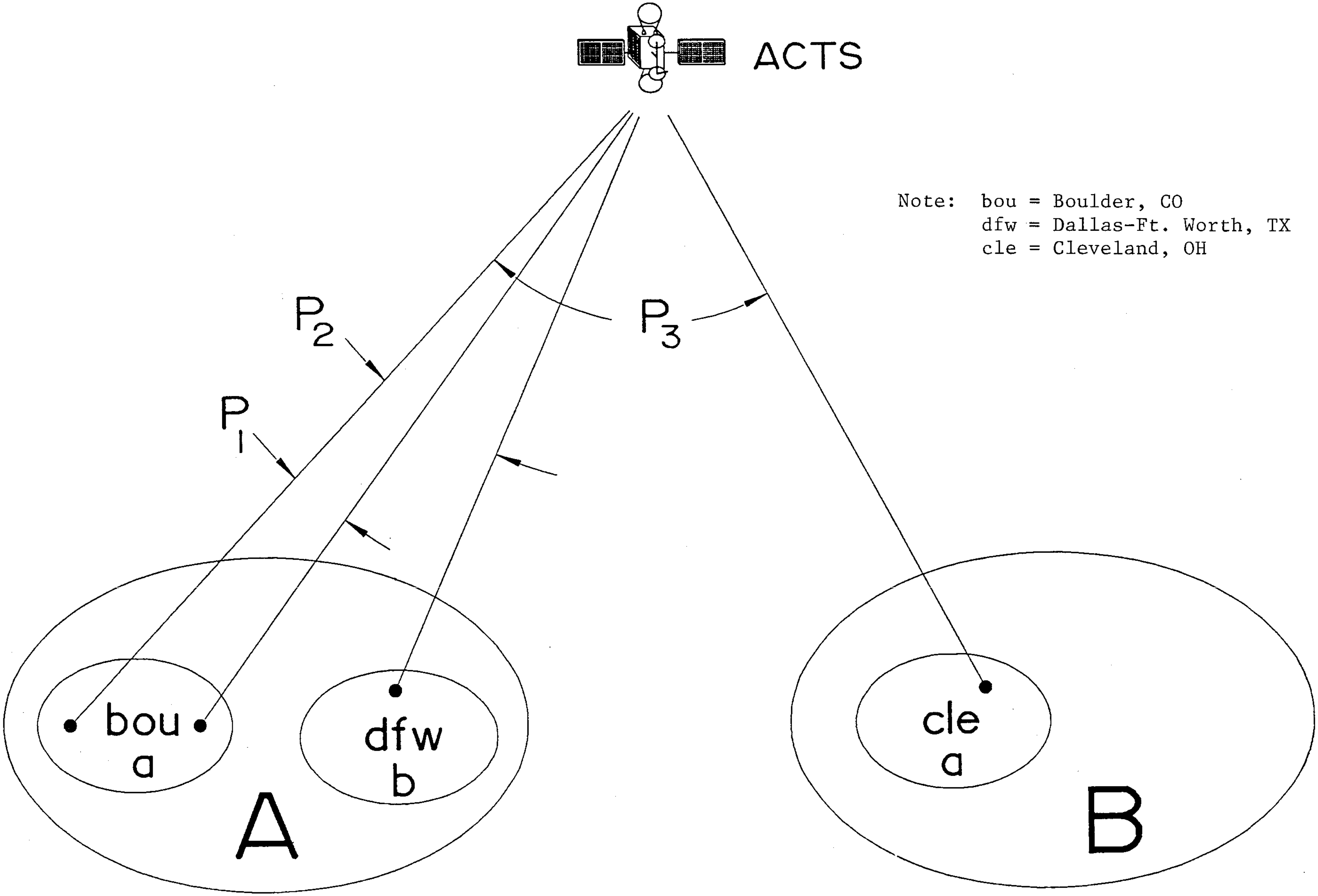


Figure 5. Propagation paths for the satellite portion of the network.

These conditions are briefly described below:

- Service: Service is a variable condition that can incorporate many network features.

- Traffic: Levels of Traffic are delineated by busy and non-busy periods (selected from combinations of levels of the two extraneous variable conditions, Time of Day and Day of Week).

- Propagation Path: The Propagation Path is determined by the locations of the source end user and the destination end user.

4.2.2 User Information Transfer Tests

Table 9 is a 2x2 matrix of variable conditions existing during user information transfer tests. Although Traffic is listed as a non-controlled condition, it can be controlled to some extent by selecting Time of Day and Day of Week and, perhaps, by fooling the master control station.

Table 9. Variable Conditions for User Information Transfer Parameters Listed According to Their Type and Control Status

	Controlled	Non-Controlled
Primary	• Service	• Propagation Impairment • Traffic
Extraneous	• Propagation Path	• Packet Size

These conditions are briefly described below:

- Service: Service is a variable condition whose levels can contain different network features.

- Propagation Impairment: Levels of Propagation impairment are delineated by rainy and clear days.

- Traffic: Levels of Traffic are delineated by busy-hour and non-busy periods (selected from combinations of levels of the extraneous variable conditions, Time of Day and Day of Week).

- Propagation Path: The Propagation Path depends upon the locations of the source end user and the destination end user.

- <u>Packet Size</u>: The packet size is presently unknown. It is not known if it will be fixed or variable. Hence, it is listed as a non-controllable variable condition, but packet size will not influence the design of the experiment.

5. EXPERIMENT DESIGN EXAMPLES

The test durations required to achieve a set of precisions will be determined by the preliminary characterization test (Section 7). Some acceptable designs are discussed in this section. Then, the experimenter must select a precision and a design which, combined, meet time and budget requirements, and adequately characterize the communication system.

Two or more primary variable conditions may interact (i.e., their effects may be dependent). Hence, it is desirable to observe a performance parameter for all combinations of levels of primary variable conditions.

Extraneous variable conditions should be allowed to assume their natural levels so that the conclusions will be representative. It is essential that levels of primary and extraneous variable conditions not vary together (i.e., they should not be correlated). They should be neutralized by design techniques such as randomization (which tends to remove bias), blocking, and balance (which tends to provide uniform precision). When levels cannot be controlled, the order of testing should be randomized (by using a table of random permutations).

Following is a brief description of four commonly used designs:

- Factorial Design. A (full) factorial design requires that a test be conducted at every combination of levels of the primary conditions. For example, if four primary variable conditions have 4, 3, 3, and 2 levels, there will be 4 x 3 x 3 x 2 = 72 tests. Factorial design can be considered if there is more than one primary variable condition.

- Fractional Factorial Design. Fractional factorial designs are factorial designs in which some combinations of levels are missing. This design might be considered if it is impractical to implement the larger factorial design. Fractional factorial designs are available that enable the main effects of variables to be estimated, but not the interactions between them. Fractional factorial designs should sometimes be favored over factorial designs for preliminary experiments because they are smaller and the results of such experiments cannot easily be predicted (i.e., redesign is likely).

- Blocking Designs. It is important that primary and extraneous variable conditions not vary together (i.e., as if they are correlated). For example, it is not wise to test Service S_1 always on Monday and Service S_2 always on Wednesday. In this case, the effects of Day of the Week and Service could not be separated; they would be confounded. To separate the effects of primary and extraneous variable conditions, their levels can be combined either randomly or systematically. For each level of an extraneous variable condition (called a block in the blocking design), the levels of the primary variable condition are selected. Moreover, it is desirable to achieve balance by including all levels of each primary variable condition the same number of times within each block (to obtain equal precision). If they are randomly selected, the design is called a randomized block. If they are selected systematically, the design might be an incomplete block design, a Latin square, or others. At each level of an extraneous variable condition, obtain an observation for all combinations of levels of primary variable conditions. This allows the effects of the extraneous variable conditions to be estimated and removed. Then, the decision concerning statistical significance of the primary variable condition is more accurate.

- Response Surface Design. If the primary variable conditions are quantitative, orthogonal central composite designs can be used, and a regression surface can be fitted.

5.1 Access-Disengagement Tests

The levels of the fixed conditions and the variable conditions will be selected and then the design will be selected. The examples given here pertain to the ACTS system performance experiment ITS is proposing; however, similar experiment design steps would be required for any communication satellite system performance experiment. The steps to implement the test using the NTIA software are given in Appendix B.[14]

5.1.1 Select the Values of the Fixed Conditions

- Source Site: This is always Boulder, CO. Enter **bou** in the **default** file and in the **netcodes** file.
- Specified Performance Values: These are selected as described in Section 7 and entered in the **spi.acd** file.
- Intersession Delay: Enter this in the **default** file. (See Figure B-1 for the ACTS default file.)
- Block Size: Enter this in the **default** file. Use the minimum size (64 bytes) because the block size condition is irrelevant to access-disengagement tests.
- Blocks: Because this is an access-disengagement test, the number of blocks is automatically set equal to 1.

[14]The software examples and files listed in this report correspond to the NTIA system performance software.

5.1.2 Select the Levels of the Variable Conditions

The primary variable conditions are Service and Traffic, and the explicit extraneous variable condition is Propagation Path.

- Service: Service is a variable condition with four levels:

  - S_1: basic rate interface (BRI) 2B+D with 64 kb/s circuit switched over the B channel,
  - S_2: basic rate interface 2B+D with 64 kb/s packet Service over the B channel,
  - S_3: basic rate interface 2B+D with 16 kb/s packet Service over the D channel, and
  - S_4: low speed data via dialed circuit switched connection through the ACTS central office (CO) interface.

  S1, **S2**, **S3**, or **S4** is entered in the command line of the shell script **runxt**.

- Traffic: Since the level of Traffic will be known as a function of time, select two levels from pairs of Times of Day and Days of the Week:

  - T_1: Busy period, 0800-1200 on Monday, and
  - T_2: Non-busy period, 2000-2400 on Sunday.

  T1 and **T2** are not entered in the NTIA software. Rather, tests are conducted during the appropriate times.

- Propagation Path: As in Figure 5, consider two sectors, A and B, each containing two spot-beam dwells, a and b. Select three spot-beam dwells identified by the 2-tuples: (A, a), (A, b), (B, a). Select one source site and three destination sites from these spot-beam dwells so that propagation paths over the following three qualitative pairs of source-destination sites are tested:

  - P_1: from Boulder in (A, a) to ACTS and from ACTS to Boulder in (A, a),
  - P_2: from Boulder in (A, a) to ACTS and from ACTS to Dallas-Ft. Worth in (A, b), and
  - P_3: from Boulder in (A, a) to ACTS and from ACTS to Cleveland in (B, a).

  P1 (or **bou**), **P2** (or **dfw**), **P3** (or **cle**) are entered in the **default** file.

Edit the **default** file and the **netcodes** file as indicated above. Then, to conduct an access-disengagement test, type

runxt o <level of Service>

5.1.3 Select the Type of Design

In order to determine the number of access-disengagement tests required to adequately measure the various system conditions and levels, an appropriate experiment design must be selected. (The number of trials within each test required to attain a specified precision will be determined by the preliminary characterization test discussed in Section 7.) The preferred design is the factorial design because it includes all combinations of levels. This design also seems to be practical for access-disengagement tests since the number of combinations of levels is not too large. There are

(4 Services) x (3 Paths) x (2 levels of Traffic) = 24 tests,

each having the number of observations required for the specified precision.

The 24 tests should be conducted in a random order. To do this it would be necessary to relocate an experimenter and equipment randomly from destination-to-destination - an impractical procedure. However, the desirability of relocating is not too important since, at each destination site, an experimenter can conduct tests at random times of day and days of week. Only the time of year would not be random, and some variation from time of year is probably taken into account by Traffic.

The following 3-tuples identify the 24 access-disengagement tests. They are numbered 1-8 for each Propagation Path:

1 (S_1, P_1, T_1) 2 (S_1, P_1, T_2)
3 (S_2, P_1, T_1) 4 (S_2, P_1, T_2)
5 (S_3, P_1, T_1) 6 (S_3, P_1, T_2)
7 (S_4, P_1, T_1) 8 (S_4, P_1, T_2)

1 (S_1, P_2, T_1) 2 (S_1, P_2, T_2)
3 (S_2, P_2, T_1) 4 (S_2, P_2, T_2)
5 (S_3, P_2, T_1) 6 (S_3, P_2, T_2)
7 (S_4, P_2, T_1) 8 (S_4, P_2, T_2)

1 (S_1, P_3, T_1) 2 (S_1, P_3, T_2)
3 (S_2, P_3, T_1) 4 (S_2, P_3, T_2)
5 (S_3, P_3, T_1) 6 (S_3, P_3, T_2)
7 (S_4, P_3, T_1) 8 (S_4, P_3, T_2).

To randomize the order of the tests, arbitrarily select from a table of random permutations (Moses and Oakford, 1963), the third, fourth, and fifth groups of 9 random permutations of the numbers 1 through 9:

3 1 7	5 8 2	7 3 5
2 6 9	9 4 7	6 8 2
8 5 4	1 3 6	9 4 1

Because there are eight tests on each Propagation Path, ignore the number nine. The tests can be conducted in the following random order for each Propagation Path.

	Path		
Order	P_1	P_2	P_3
1	3	5	7
2	1	8	3
3	7	2	5
4	2	4	6
5	6	7	8
6	8	1	2
7	5	3	4
8	4	6	1

5.2 User Information Transfer Tests

The levels of the variable conditions will be selected first, and then the design. Again, the examples given here pertain to the ACTS system performance experiment, however, similar experiment design steps would be required for any communication satellite system performance experiment.

5.2.1 Select the Values of the Fixed Conditions

- <u>Source Site</u>: This is always Boulder, CO. Enter **bou** in the **default** file and in the **netcodes** file.
- <u>Specified Performance Values</u>: These are selected in Section 7 and entered in the **spi.xfr** file.
- <u>Interblock Delay</u>: Enter this delay (in seconds) in the **default** file.
- <u>Block Size</u>: Enter this in the **default** file. NTIA software is designed for blocks having from 64 to 512 bytes. Use a block size of 128 bytes for the ACTS experiment.

- Access Attempts: Because this is a user information transfer test, the number of access attempts is automatically set equal to 1.

5.2.2 Select the Levels of the Variable Conditions

There are three primary variable conditions: Service, Traffic, and Propagation Impairment. There are two explicit extraneous variable conditions: Propagation Path and Packet Size.

- Service: Service is a variable condition with four levels:

  - S_1: basic rate interface (BRI) 2B+D with 64 kb/s circuit switched over the B channel,
  - S_2: basic rate interface 2B+D with 64 kb/s packet Service over the B channel,
  - S_3: basic rate interface 2B+D with 16 kb/s packet Service over the D channel, and
  - S_4: low speed data via dialed circuit switched connection through the ACTS CO interface.

  S1, S2, S3, or S4 is entered in the command line of the shell script **runxt**.

- Traffic: Since the level of Traffic will be known as a function of time, select two levels from Day of Week and Time of Day:

  - T_1: Busy period, 0800-1200 on Monday, and
  - T_2: Non-busy period, 2000-2400 on Sunday.

  T1 and T2 are not entered in the NTIA software. Rather, tests are conducted during the appropriate network traffic conditions.

- Propagation Impairment: Propagation can be impaired from weather along either or both links. Select two levels:

  - I_1: rainy, and
  - I_2: clear.

  I1 and I2 are not entered in the NTIA software. Rather, tests are conducted during the appropriate link weather conditions.

- Propagation Path: As in Figure 5, consider two sectors, A and B, each containing two spot-beam dwells, a and b. Select three spot-beam dwells identified by the 2-tuples: (A, a), (A, b), (B, a). Select one source site and three destination sites from these spot-beam dwells so that Paths over the following three qualitative pairs of source-destination sites are tested:

- P_1: from Boulder in (A, a) to ACTS and from ACTS to Boulder in (A,a),
- P_2: from Boulder in (A, a) to ACTS and from ACTS to Dallas-Ft. Worth in (A, b), and
- P_3: from Boulder in (A, a) to ACTS and from ACTS to Cleveland in (B, a).

P1 (or **bou**), **P2** (or **dfw**), **P3** (or **cle**) are entered in the **default** file.

- Packet Size: The packet size is presently unknown and will not affect the design of the experiment.

Edit the **default** file and the **netcodes** file as indicated above. Then, to conduct a user information transfer test, type

runxt u <level of Service>

5.2.3 Select the Type of Design

In order to determine the number of user information transfer tests required to adequately measure the various system conditions and levels, an appropriate experiment design must be selected. (The number of trials within each test required to attain a specified precision will be determined by the preliminary characterization test discussed in Section 7.) User information transfer tests have more combinations of levels than do access-disengagement tests. For this reason it is instructive to discuss a few designs.

A. Factorial Design

For this design, there are

(4 Services) x (3 Propagation Paths) x (2 levels of Traffic) x (2 levels of Propagation Impairment) = 48 tests,

each having the number of observations required to achieve the specified precision.

The 48 tests should be conducted in a random order. To do this it would be necessary to relocate an experimenter and equipment randomly from destination-to-destination - an impractical procedure. However, the desirability of relocating is not too important since, at each destination site, an experimenter can conduct the tests at random times of day and days of week.

Only the time of year would not be random, and some of the variation of time of year is probably taken into account by Traffic and Impairment.

The tests are identified by the following 48 4-tuples. They are numbered 1-16 for each Propagation Path:

1	(S_1, P_1, T_1, I_1)	2	(S_1, P_1, T_1, I_2)
3	(S_1, P_1, T_2, I_1)	4	(S_1, P_1, T_2, I_2)
5	(S_2, P_1, T_1, I_1)	6	(S_2, P_1, T_1, I_2)
7	(S_2, P_1, T_2, I_1)	8	(S_2, P_1, T_2, I_2)
9	(S_3, P_1, T_1, I_1)	10	(S_3, P_1, T_1, I_2)
11	(S_3, P_1, T_2, I_1)	12	(S_3, P_1, T_2, I_2)
13	(S_4, P_1, T_1, I_1)	14	(S_4, P_1, T_1, I_2)
15	(S_4, P_1, T_2, I_1)	16	(S_4, P_1, T_2, I_2)

1	(S_1, P_2, T_1, I_1)	2	(S_1, P_2, T_1, I_2)
3	(S_1, P_2, T_2, I_1)	4	(S_1, P_2, T_2, I_2)
5	(S_2, P_2, T_1, I_1)	6	(S_2, P_2, T_1, I_2)
7	(S_2, P_2, T_2, I_1)	8	(S_2, P_2, T_2, I_2)
9	(S_3, P_2, T_1, I_1)	10	(S_3, P_2, T_1, I_2)
11	(S_3, P_2, T_2, I_1)	12	(S_3, P_2, T_2, I_2)
13	(S_4, P_2, T_1, I_1)	14	(S_4, P_2, T_1, I_2)
15	(S_4, P_2, T_2, I_1)	16	(S_4, P_2, T_2, I_2)

1	(S_1, P_3, T_1, I_1)	2	(S_1, P_3, T_1, I_2)
3	(S_1, P_3, T_2, I_1)	4	(S_1, P_3, T_2, I_2)
5	(S_2, P_3, T_1, I_1)	6	(S_2, P_3, T_1, I_2)
7	(S_2, P_3, T_2, I_1)	8	(S_2, P_3, T_2, I_2)
9	(S_3, P_3, T_1, I_1)	10	(S_3, P_3, T_1, I_2)
11	(S_3, P_3, T_2, I_1)	12	(S_3, P_3, T_2, I_2)
13	(S_4, P_3 ,T_1, I_1)	14	(S_4, P_3, T_1, I_2)
15	(S_4, P_3, T_2, I_1)	16	(S_4, P_3, T_2, I_2)

Again, to randomize the order of the tests, arbitrarily select the third, fourth, and fifth groups of random permutations of the numbers 1 through 16 (Moses and Oakford, 1963):

15	3	16	11	14	15	6	16	5	3	12	7
10	12	9	2	1	4	2	3	15	13	2	9
7	4	13	5	13	9	11	8	8	10	6	1
8	1	6	14	10	12	7	5	14	4	11	16

	Path		
Order	P_1	P_2	P_3
1	15	14	5
2	3	15	3
3	16	6	12
4	11	16	7
5	10	1	15
6	12	4	13
7	9	2	2
8	2	3	9
9	7	13	8
10	4	9	10
11	13	11	6
12	5	8	1
13	8	10	14
14	1	12	4
15	6	7	11
16	14	5	16

If 48 tests are thought to be too many or impractical (because Propagation Impairment cannot be controlled), the following fractional factorial design could be considered.

B. Fractional Factorial Design

Fractional factorial designs omit some combinations of levels. If a pair of variable conditions are independent, it does not matter that their results may be confounded because some combinations of levels are omitted.

Even if the full factorial design contains a manageable number of tests, it is somewhat impractical to implement since Propagation Impairment is non-controllable; the experiment could require an extensive period of time waiting for the proper weather conditions to occur with the required combination of levels of Propagation Path, Traffic, and Service.

Moreover, it seems that Propagation Impairment is independent of both Propagation Path and Traffic; so this primary variable condition can be assigned at random with the levels of the other variable conditions. Use a fractional factorial design that is a balanced randomized block design: randomly assign two levels of Propagation Impairment to 24 3-tuples. For example, Table 4 of Tables of Random Permutations (Moses and Oakford, 1963) contains many sequences of 30 random permutations (from 1 through 30). These four steps could be followed to adapt one of these sequences to this situation:

- Since the table contains sequences of 30 random numbers and there are two levels (i.e., I_1 and I_2), the sequence is divided into two sequences: 1-15 and 16-30.

- Since only 24 random numbers are needed, the numbers 13, 14, and 15 are arbitrarily ignored in the first sequence and 28, 29, and 30 in the second. Note: an equal number of tests is assured at each level (i.e., achieving balance) - but in a random order.

- The integers 1-12 are assigned to I_1 and 16-27 to I_2. That is, the first 3-tuple is assigned to I_1 or I_2 depending upon whether the first integer in the permuted sequence is in the sequence 1-12 or 16-27, respectively. Then the second 3-tuple is assigned to I_1 or I_2 according as whether the second integer in the permuted sequence is in the sequence 1-12 or 16-27, respectively. Assignments are continued through the 24th 3-tuple.

- 24 4-tuples are developed using the 24 3-tuples of Section 5.1.2 coupled with the resulting random levels of Propagation Impairment.

For example, the fifth set of 30 random permutations (Moses and Oakford, 1963) is arbitrarily selected:

26	5	23	28	21
14	29	19	27	15
17	12	6	4	9
8	16	11	2	10
18	1	24	13	7
30	22	25	3	20

The 24 3-tuples are assigned Propagation Impairments accordingly. Listed below is each integer, from 1-24, its corresponding random permutation (r.p.) from this sequence, and its resulting 4-tuple:

Order	r.p.	4-tuple
1	26	(S_1, P_1, T_1, I_2)
2	5	(S_1, P_1, T_2, I_1)
3	23	(S_1, P_2, T_1, I_2)
4	21	(S_1, P_2, T_2, I_2)
5	14	(S_1, P_3, T_1, I_2)
6	19	(S_1, P_3, T_2, I_2)
7	27	(S_2, P_1, T_1, I_2)
8	15	(S_2, P_1, T_2, I_2)
9	12	(S_2, P_2, T_1, I_1)
10	6	(S_2, P_2, T_2, I_1)
11	4	(S_2, P_3, T_1, I_1)
12	9	(S_2, P_3, T_2, I_1)
13	8	(S_3, P_1, T_1, I_1)
14	16	(S_3, P_1, T_2, I_2)
15	11	(S_3, P_2, T_1, I_1)
16	2	(S_3, P_2, T_2, I_1)
17	10	(S_3, P_3, T_1, I_1)
18	1	(S_3, P_3, T_2, I_1)
19	24	(S_4, P_1, T_1, I_2)
20	13	(S_4, P_1, T_2, I_2)
21	7	(S_4, P_2, T_1, I_1)
22	22	(S_4, P_2, T_2, I_2)
23	25	(S_4, P_3, T_1, I_2)
24	3	(S_4, P_3, T_2, I_1)

This set of 24 4-tuples identifies the 24 user information tests - provided the weather cooperates. Otherwise, an available subset of tests must be obtained and suitable analysis performed. The 4-tuples are numbered 1-8 for each Propagation Path.

1	(S_1, P_1, T_1, I_2)	2	(S_1, P_1, T_2, I_1)
3	(S_2, P_1, T_1, I_2)	4	(S_2, P_1, T_2, I_2)
5	(S_3, P_1, T_1, I_1)	6	(S_3, P_1, T_2, I_2)
7	(S_4, P_1, T_1, I_2)	8	(S_4, P_1, T_2, I_2)

1	(S_1, P_2, T_1, I_2)	2	(S_1, P_2, T_2, I_2)
3	(S_2, P_2, T_1, I_1)	4	(S_2, P_2, T_2, I_1)
5	(S_3, P_2, T_1, I_1)	6	(S_3, P_2, T_2, I_1)
7	(S_4, P_2, T_1, I_1)	8	(S_4, P_2, T_2, I_2)

1	(S_1, P_3, T_1, I_2)	2	(S_1, P_3, T_2, I_2)
3	(S_2, P_3, T_1, I_1)	4	(S_2, P_3, T_2, I_1)
5	(S_3, P_3, T_1, I_1)	6	(S_3, P_3, T_2, I_1)
7	(S_4, P_3, T_1, I_2)	8	(S_4, P_3, T_2, I_1)

These tests can be conducted in a random order for each Propagation Path. Using the sixth, seventh and eighth set of nine random permutations (Moses and Oakford, 1963)

4	9	8	5	7	6	9	2	8
3	2	6	1	4	2	4	5	1
7	1	5	8	9	3	6	3	7

Because there are eight tests on each Propagation Path, ignore the number nine. The tests can be conducted in the following random order for each Propagation Path:

	Path		
Order	P_1	P_2	P_3
1	4	5	2
2	8	7	8
3	3	6	4
4	2	1	5
5	6	4	1
6	7	2	6
7	1	8	3
8	5	3	7

6. SPECIFY PERFORMANCE VALUES

The experimenter must specify a performance value for most performance parameters. The specified performance value can be a nominal value (as determined from a preliminary characterization test), a required value, or any other reasonable value. (When a trial delay is sufficiently long, the trial must be considered a failure. For example, after a certain length of time an access attempt is no longer an observation of the performance parameter, Access Time, but is instead an observation of the performance parameter, Access Denial.) In this way Access Denial is defined. The performance parameter, Transfer Denial, is similarly defined in terms of other parameters that describe the quality and rate of user information transfer.

The values from Table 5 may help the experimenter to specify performance values. If a preliminary test to characterize the system (Section 7) is conducted, the specified times should be large enough so that the times are achieved (and the trials are not terminated as failures). Even though not all performance parameters are required for this experiment, reasonable values for them must be specified in the specification files. The creation of the specification files, **spi.acd** and **spi.xfr**, for the ACTS preliminary characterization test is described in Appendix B. It will reside in **/usr/net/spi**.

6.1 Access-Disengagement Tests

Access Time, Source Disengagement Time, and Destination Disengagement Time are all considered to be denials if they are not completed within the threshold time - three times the specified performance time. The specified times (in seconds) are entered in the file, **spi.acd**.

6.1.1 Access Time

Specify an Access Time and a User Fraction of Access Time. The threshold value of Access Time is three times the specified value. The threshold value for User Fraction of Access Time is used to assign responsibility for access timeout: if the user fraction is larger than the threshold value, responsibility is assigned to the user; otherwise it is assigned to the system. Outcomes are determined by data reduction software as in Figure 6.

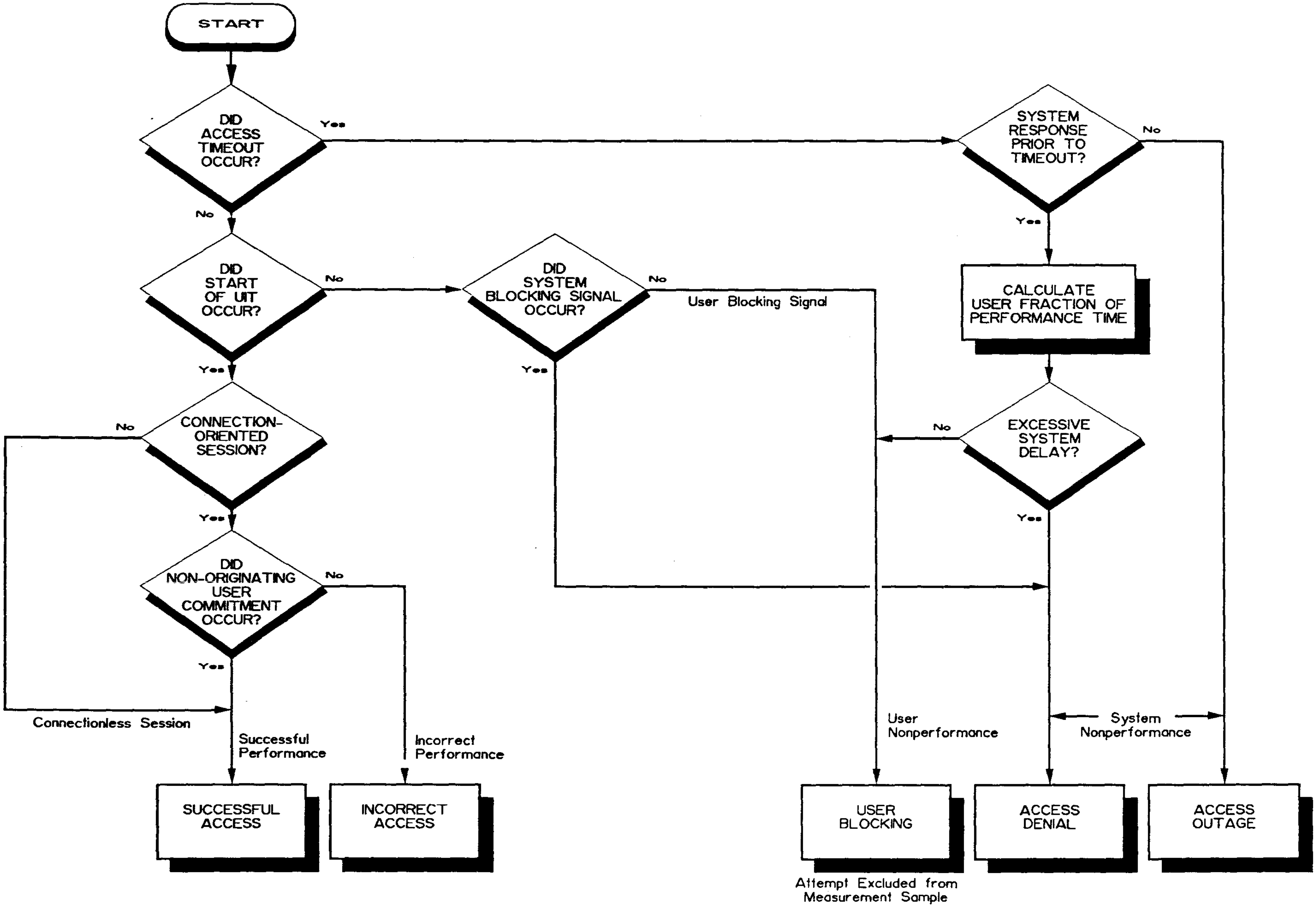


Figure 6. Scheme for classifying outcomes of access attempts.

6.1.2 Source Disengagement Time

Specify a Source Disengagement Time and a User Fraction of Source Disengagement Time. The threshold value of Source Disengagement Time is three times the specified value. Outcomes of disengagement (either source or destination disengagement) are determined as in Figure 7.

6.1.3 Destination Disengagement Time

Specify a Destination Disengagement Time and a User Fraction of Destination Disengagement Time. The threshold value of Destination Disengagement Time is three times the specified value. Outcomes are determined as in Figure 7.

6.2 User Information Transfer Tests

The performance for user information transfer parameters are not as easily specified. The Transfer Denial Probability, for example, requires thresholds of other parameters for its specification. The specified values for these parameters are entered in the file, **spi.xfr**.

6.2.1 Block Transfer Time

Specify Block Transfer Time and User Fraction of Block Transfer Time. The threshold Block Transfer Time is three times the specified value. Outcomes are classified as in Figure 8.

6.2.2 User Information Bit Transfer Rate and User Fraction of Input/Output Time

Specify the User Information Bit Transfer Rate. The threshold rate is one-third this value. User Information Bit Transfer Rate has a successful outcome if its rate is higher than the threshold rate.

Specify the User Fraction of Input/Output Time. The threshold value is three times this value. The User Fraction of Input/Output Time is a successful outcome if the Input/Output Time of a throughput sample is less than the threshold time.

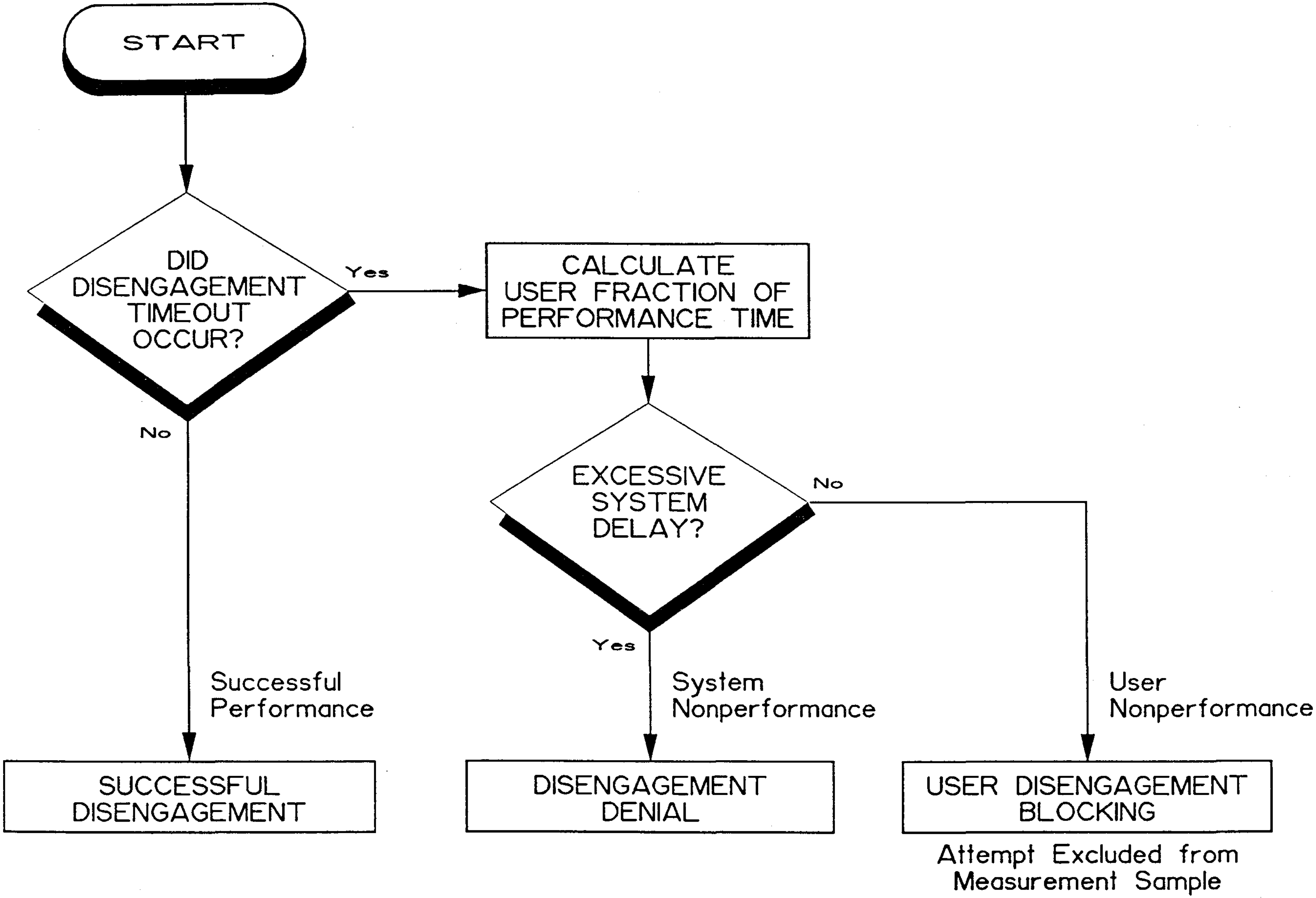


Figure 7. Scheme for classifying outcomes of disengagement attempts.

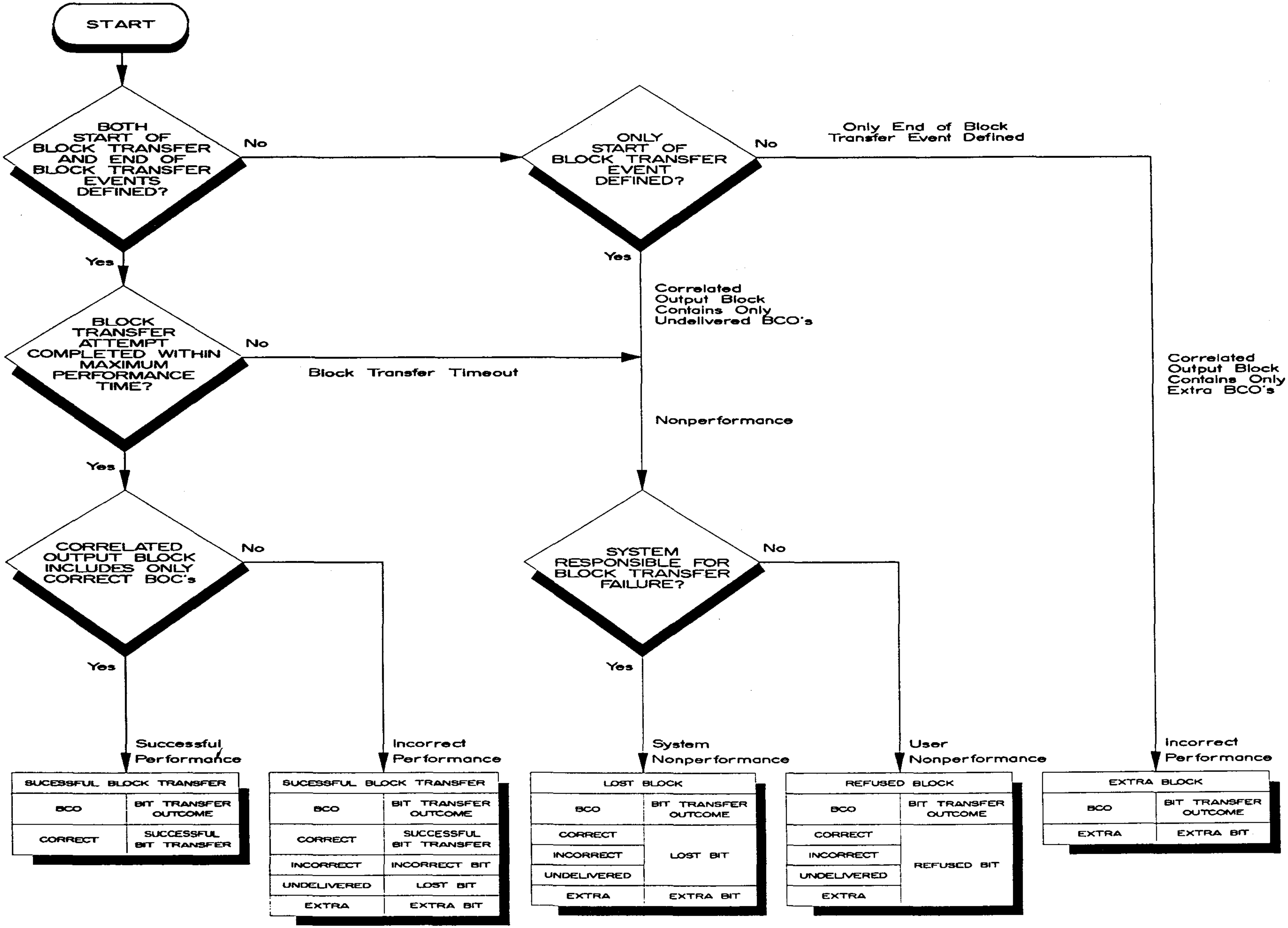


Figure 8. Scheme for classifying outcomes of bit and block transfer attempts.

6.2.3 Bit Error, Bit Loss, and Extra Bit Probability

These probabilities are specified only to define Transfer Denial Probability.[15] They are discussed in the following section.

6.2.4 Transfer Denial Probability

This parameter is defined in terms of four other performance parameters, (called support parameters). They are the Bit Error Probability, Bit Loss Probability, Extra Bit Probability, and the User Information Bit Transfer Rate. The specified values for these parameters are entered in the file, **spi.xfr**.

The threshold values of the three supported probability parameters at which a Transfer Denial is declared are the fourth roots of their specified values, and the threshold value of the User Information Bit Transfer Rate is one-third its specified value. Table 10 lists the supported performance parameters, the variable indicating their specified performance values, and the corresponding threshold values.

Table 10. Specified Performance Values and Threshold Values of Supported Performance Parameter

SUPPORTED PERFORMANCE PARAMETERS	SPECIFIED VALUE	THRESHOLD VALUE
Bit Error Probability	p_e	$p_e^{1/4}$
Bit Loss Probability	p_l	$p_l^{1/4}$
Extra Bit Probability	p_x	$p_x^{1/4}$
User Info. Bit Transfer Rate	r	r/3

To define Transfer Denial Probability, first determine the size of a Transfer Sample (i.e., a certain number of bits), and then determine whether each transfer sample has been successfully transferred.

[15]Since these are failure parameters there is no threshold (at which they become failures).

A. **Transfer Sample Size**

A transfer sample is a selected sample of user information bits transferred between a specified source and destination user.[16] The size of a transfer sample is defined in terms of the three supported failure probability parameters: A transfer sample must

- include an integral number of blocks - that includes at least one interblock gap, and
- be large enough to provide an estimate of the three supported failure probability parameters with a relative precision of 50% at the 95% confidence level.[17] The transfer sample size should be entered in line 5 of the **spi** file. (See Table B-3.)

Example 1 (Transfer Sample Size): A user information transfer test transfers blocks of 128 characters with interblock gaps of 0.10 second. Assume the specified values for the four supported parameters (Bit Error Probability, Bit Loss Probability, Extra Bit Probability, and User Information Bit Transfer Rate) are, respectively,

$$p_e = 10^{-6},\ p_l = 10^{-8},\ p_x = 10^{-8},\ \text{and}\ r = 9{,}600\ \text{bits/s}.$$

The maximum conditional probability (of a failure given that a failure occurred on the previous trial) for the four supported failure probability parameters is thought to be 0.4, 0.2, and 0.4, respectively. Determine the transfer sample size and the minimum time during which Transfer Denial must be observed.

Solution: The corresponding threshold values of the three supported parameters are

$$10^{-1.5},\ 10^{-2},\ 10^{-2},\ \text{and}\ 3{,}200\ \text{bits/s}.$$

- Program **star** determines the number of failures required for 50% relative precision at the 95% confidence level as 38, 25, and 38, respectively.

---

[16]This number of bits should probably be called a transfer trial instead of a transfer sample because it is one of many bit strings to be examined.

[17]ANS X3.102 states that the transfer sample must be large enough to estimate all of the four supported performance parameters with a relative precision of at least 50% at the 95% confidence level. However, this is not possible for the User Information Bit Transfer Rate:

- There is but one observation per test using NTIA software; hence, its precision is not measurable.
- If there were more than one observation per test, this parameter would be measured with absolute precision.

- The corresponding numbers of bits are random variables. As such they are unknown. A rough estimate of the number of bits are

  $38/10^{-1.5} = 1{,}202$ bits,
  $25/10^{-2} = 2{,}500$ bits, and
  $38/10^{-2} = 3{,}800$ bits.

  The Transfer Sample must be at least as large as the largest number of bits (i.e., 3,800 bits). The number of bits must be entered into the **spi.xfr** file.

- Since each block has 128 characters (and there are 8 bits/character), each block has 128 x 8 = 1,024 bits. Hence, the Transfer Sample must consist of at least four blocks (i.e., 4 x 1,024 = 4,096 bits > 3,800 bits).

- The specified transfer time for the four blocks is

  $$4 \times (1{,}024 \text{ bits})/(3{,}200 \text{ bits/s}) + 3 \times 0.10 \text{ s} = 1.28 \text{ s} + 0.30 \text{ s} = 1.58 \text{ s}.$$

- Transfer Sample Input/Output time out (i.e., observation period) is three times the specified transfer time:

  $$3 \times 1.58 \text{ s} = 4.74 \text{ s}.$$

B. Transfer Denial

A successful transfer occurs when the estimates of all <u>four</u> supported parameters evaluated for the Transfer Sample are "better" than their threshold values. Otherwise, the outcome is Transfer Denial if the system is responsible (and Rejected Sample if the user is responsible).[18] This outcome scheme is depicted in Figure 9.

Example 2 (Transfer Denial): Assume a Transfer Sample consists of 5 blocks of 128 characters (with interblock gaps of 0.10 s). Suppose they are transferred without a failure within 0.9667 s. The estimates of the four supported parameters are

$$\bar{p}_e = 0,\ \bar{p}_l = 0,\ \bar{p}_x = 0, \text{ and } \bar{r} = 9{,}035 \text{ bits/s.}$$[19]

---

[18]The user is considered to be responsible if the Input/Output Time exceeds the threshold value of User Fraction of Input/Output Time. (See Section 4.2.2.)

[19]As in Example 1, but solve for $\bar{r}$: $(5 \times 1{,}024)/\bar{r} + 4 \times 0.1 = 0.9667$.

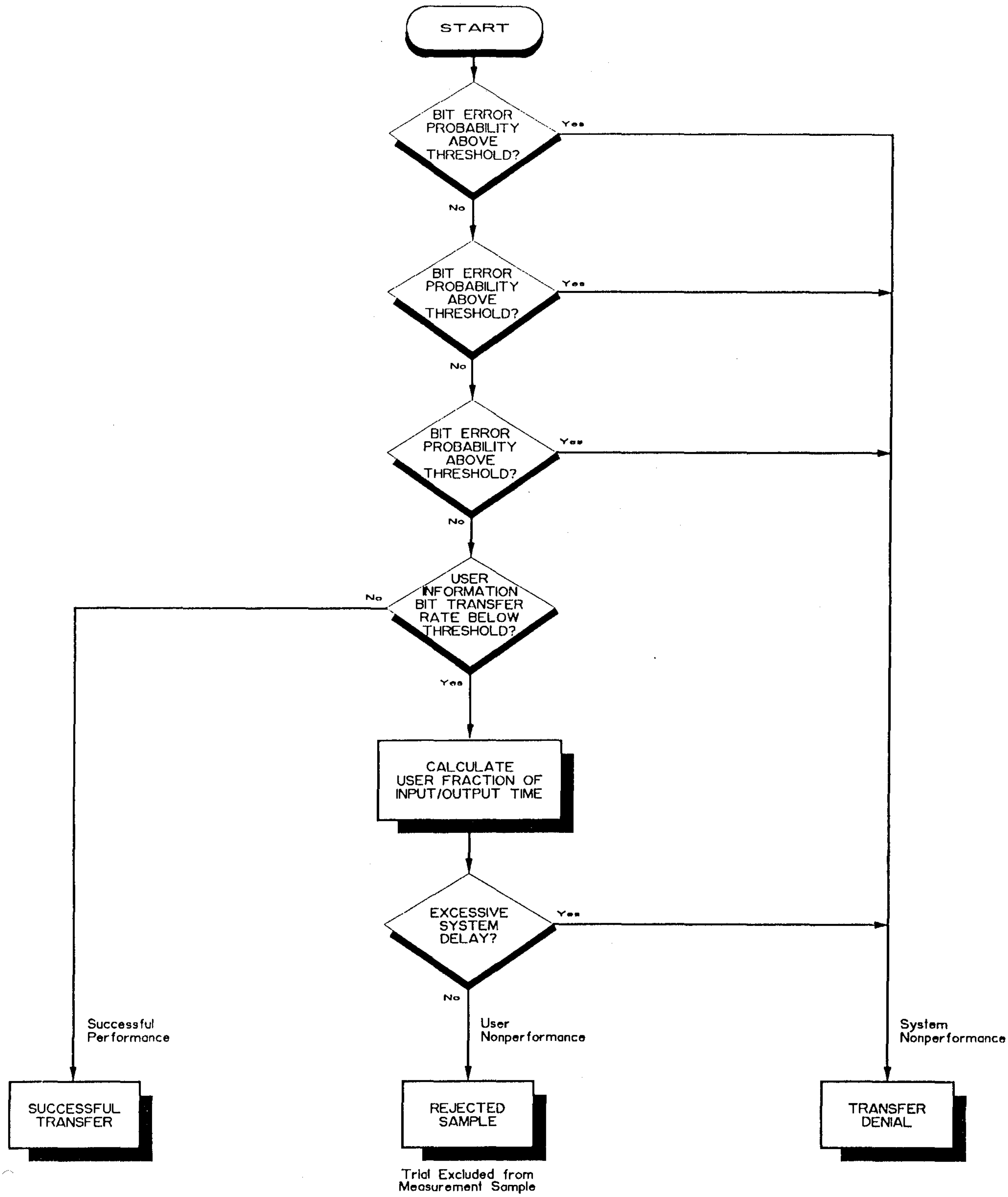


Figure 9. Scheme for classifying outcomes of transfer samples.

Using the threshold values from Example 1 (i.e., $10^{-1.5}$, 10^{-2}, and 10^{-2}, and 3,200 bits/s), determine whether a Transfer Denial occurred in this Transfer Sample.

Solution: Since none of the estimates is "worse" than its threshold, Transfer Denial did not occur.

6.2.5 Correlator Specifications

A set of values must be specified to detect bit and block failures.

A. Bit Comparison Outcomes

A bit comparison outcome (BCO) is the outcome of a bit as determined by comparing a source bit with its corresponding destination bit. There are four types of bit comparison outcomes:

- Correct BCO: a bit transfer attempt in which the destination bit has the same binary value as its corresponding source bit.
- Incorrect BCO: a bit transfer attempt in which the destination bit has a different binary value than its corresponding source bit.
- Undelivered BCO: the outcome of a bit transfer attempt in which the source bit has no corresponding destination bit.
- Extra BCO: the outcome of a bit transfer attempt in which the destination bit has no corresponding source bit.

B. Typical Sequences of Bit Failures

Bit failures tend to occur in two types of bit sequences:

- A string (or a run in statistical parlance) is a sequence of bits in which all bits have the same outcome. Correct bits, undelivered bits and extra bits tend to occur in strings.
- A cluster is a sequence of bits in which a certain fraction of its bits have the same failure outcome (a fraction greater than the failure probability). Bit errors tend to occur in clusters because they are dependent: The conditional probability of a bit error, given an error in the previous bit, exceeds the probability of a bit error. Dependence in the NTIA implementation of the standard is measured by the autocorrelation of lag 1.

C. Error Detection Procedure

When a failure is detected, a sequence of source bits and a sequence of destination bits are stored in arrays called user information windows. Since the occurrence of a failure causes the sequences to be stored, the initial bits in each sequence do not match. The two sequences are shifted past these windows in unison and all bits are compared until

- the contents of each window match,
- a specified number of bits has been shifted, or
- no bits remain to be compared.

1. Bits Per Shift

The sequences are shifted either one bit at a time or one word at a time. (The bit shift is the default value.) If the word shift (called fast correlator algorithm) is selected, only word failures will be detected. If the word shift option is selected, specify this in subroutine **BITCOR** (in **/src/d4/alyz**).

2. Window Size

The size of these two windows (which must be equal) is set to 16 bits[20]. If no failures are detected, source words are compared with destination words (using the fast correlator algorithm as described above). If failures are detected, windows are used. Enter the user information window size in line six of the **spi.xfr** file.

3. Maximum Bit Shifts

Selection of the maximum bit shift depends upon speed and the degree of certainty with which the experimenter wishes to detect failures. The correlator cannot detect any bit transfer failure whose length exceeds the maximum bit shift.

---

[20]If the window size is changed, change the array size in two variables: IDIWIN and ISIWIN. These are located in common blocks DUIWIN and SUIWIN, respectively. DUIWIN is contained in subroutines BITCOR, DWLOAD, DSHIFT, and COMPAR. SUIWIN is contained in subroutines BITCOR, SWLOAD, SSHIFT, and COMPAR. If the window size exceeds 32, a dimension statement in BITCOR must be increased.

a. **Speed**

If the maximum bit shift is large, correlation may be slow.

b. **Detection**

- to detect no failures, set it equal to zero.
- to detect all failures, set it equal to the number of bits in several blocks,
- to detect all failures, set it equal to the number of bits transferred (or at least to the largest failure string or cluster - but this is unknown).

Specify the maximum bit shift for each type of failure in line six of the **spi.xfr** file.

- For bit errors, this is arbitrarily set at 256.
- For undelivered bits and extra bits, this could be set at the number of bits in several blocks.

Section B.3 shows how to create the specifications file, **spi.xfr**.

7. PRELIMINARY CHARACTERIZATION TEST AND TEST DURATION

Before testing, the performance parameters are probably known only within an order of magnitude or so. The experiment can be conducted with this limited knowledge; however, it is strongly recommended that a preliminary, "typical" test is conducted to roughly characterize the system or service. The knowledge obtained will allow the experiment to be designed more efficiently:

- The (minimum) sample size required to achieve a specified precision can be determined more accurately because the dispersion and dependence of trial values will be estimated.[21]
- The time required to conduct a test having the required precision can be determined. Hence, the number of tests and the selected levels of variable conditions can be allocated judiciously.
- Meaningful specified performance values can be assigned, particularly for the four supported parameters required for Transfer Denial Probability (Section 6).

Figure 10 is a structured design diagram depicting the input and output from a preliminary test supported by the NTIA software.

7.1 Access-Disengagement Tests

For the access-disengagement tests, the number of accesses (the sample size, typically 20) required for a preliminary characterization of the Access Time parameters is entered in the **default** file (located in the home directory). Figure B-1 shows a sample **default** file for the ACTS experiment.

7.1.1 Process the Preliminary Characterization Test

Once the **default** file has been completed and the test run, the test data can be processed by the shell script **dopre**. Type

dopre nnnn

where **nnnn** is the number assigned to the preliminary test. Select any integer between 1,000 and 9,999.

---

[21]This is true for all common objectives with the exception of acceptance and maintenance. Also, if the number of failures is zero or one, the upper confidence limit can be determined only if the number of trials exceeds a certain size (a size that depends upon the conditional probability of a failure given that a failure occurred in the previous trial).

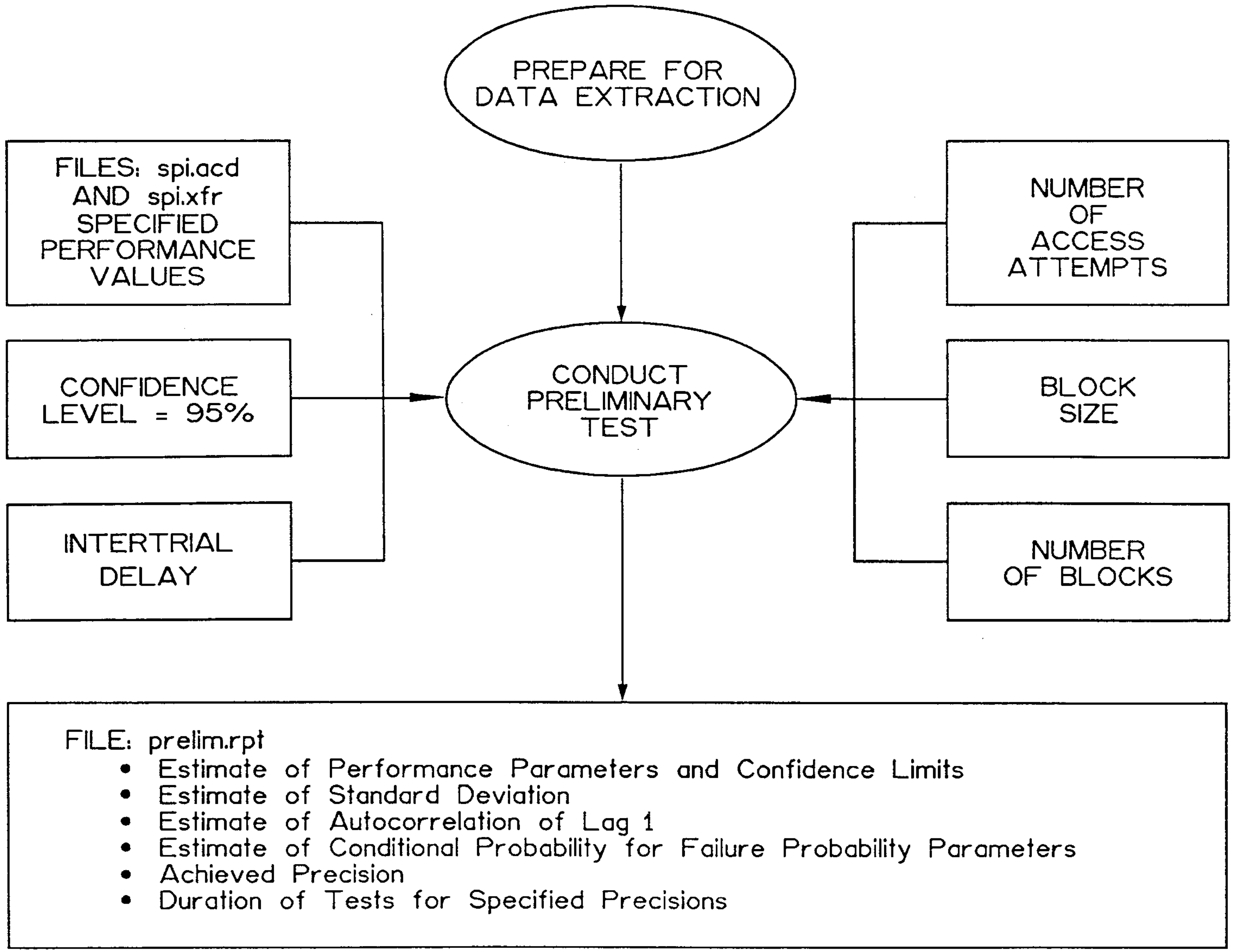


Figure 10. Structured design diagram depicting the preliminary characterization test and the determination of test durations.

The results are contained in **usr/net/prelim.rpt**. Figure 11 is an example of the first part of this file (the results shown are from the measurements conducted in Spies et al., 1988). For each performance parameter, it lists the number of trials and estimates of the mean, standard deviation, and autocorrelation of lag 1.[22] It also lists the precision (half-length of the confidence interval if absolute precision is specified) achieved at the 95% confidence level.[23]

7.1.2 Select the Precision

The standard deviation and autocorrelation of lag 1 are used to determine the sample size required to obtain a specified absolute precision for time parameters. Similarly, the conditional probability of a failure given that a failure occurred in the previous trial is used to determine the number of failures required to obtain a specified relative precision for failure probability parameters.[24]

Using the values obtained from the preliminary characterization test, the required sample size and the time required to complete a test of this size are estimated for all access-disengagement performance parameters for various precisions at the 95% confidence level. This information is the last part of file **usr/net/prelim.rpt**; an example is listed in Figure 12.

7.2 User Information Transfer Tests

For the user information transfer tests, the number of blocks (the sample size, typically 40) required to achieve a specified precision for the most important user information transfer performance parameter (Bit Error Probability) is entered in the **default** file (located in the home directory). This size is limited to 320 samples, since the minimum block size is 64 bytes and the maximum number of bytes that can be transferred is 20,480: (number of

[22]If nominal values are used as specified values (in the previous section), these estimates of the time parameters can replace them.

[23]The achieved precision depends upon all listed values except the mean.

[24]Program **star** asks for the maximum of these three values. This would provide a conservative estimate of the sample size. However, since a preliminary test has been conducted, estimates of these values - not their maximum - should be used if the estimates of the conditional probabilities are obtained from the preliminary test.

PRELIMINARY CHARACTERIZATION TEST

Access-Disengagement Parameters

Test Number 2115

Confidence Level = 95%
Time between Access Attempts = 55 s

Time Parameters
(in seconds)

Performance Parameter	Number of Trials	------------- Estimate of ------------- Mean	St. Dev.	Autocorr. of Lag 1	Achieved Absolute Precision
Access Time	20	4.268e+01	1.344e-00	0.000e+00	5.180e-01
U. F. of Access Time	20	3.084e-03	5.232e-04	9.055e-04	1.025e-03
Source Diseng. Time	20	1.886e+00	2.664e-01	6.494e-02	1.169e-01
U. F. of S. Diseng. Time	20	4.139e-02	1.232e-03	0.000e+00	2.414e-03
Destination Diseng. Time	20	3.810e-01	1.821e-02	0.000e+00	8.410e-03
U. F. of D Diseng. Time	20	1.605e-01	1.745e-03	3.267e-02	3.420e-03

Failure Probability Parameters
(number of trials = 20)

Performance Parameter	Number of Failures	---------- Estimate of ----------- Proportion	Autocorr. of Lag 1	Achieved Relative Precision
Incorrect Access	0	0.000e+00	0.000e+00	Indet.
Access Outage	0	0.000e+00	0.000e+00	Indet.
Access Denial	0	0.000e+00	0.000e+00	Indet.
Source Diseng. Denial	0	0.000e-00	0.000e-00	Indet.
Dest. Diseng. Denial	0	0.000e+00	0.000e+00	Indet.

Figure 11. Results of an access-disengagement preliminary characterization test.

REQUIRED SAMPLE SIZE AND TEST DURATION

Access-Disengagement Parameters

Confidence Level = 95%
Time between Access Attempts = 55 s

Access Time

Specified Absolute Precision	Required Number of Delays	Test Duration
1.036e-01 (seconds)	439	12.54h
2.072e-01	110	3.14h
3.108e-01	49	83.98m
4.144e-01	28	47.99m
5.180e-01	18	30.85m

Source Disengagement Time

Specified Absolute Precision	Required Number of Delays	Test Duration
2.337e-02 (seconds)	440	12.57h
4.674e-02	110	3.14h
7.011e-02	50	85.70m
9.348e-02	29	49.71m
1.169e-01	18	30.85m

Destination Disengagement Time

Specified Absolute Precision	Required Number of Delays	Test Duration
1.682e-03 (seconds)	439	12.54h
3.364e-03	110	3.14h
5.046e-03	50	85.70m
6.728e-03	29	49.71m
8.410e-03	19	32.57m

Figure 12 (Part 1). Specified precisions and test durations for access-disengagement performance parameters.

Incorrect Access

Specified Relative Precision	Required Number of Failures	Required Number of Trials	Test Duration
20%	871	unknown*	unknown
50%	145	unknown	unknown
100%	41	unknown	unknown
200%	14	unknown	unknown
500%	6	unknown	unknown

Access Outage

Specified Relative Precision	Number of Required Failures	Required Number of Trials	Test Duration
20%	871	unknown	unknown
50%	145	unknown	unknown
100%	41	unknown	unknown
200%	14	unknown	unknown
500%	6	unknown	unknown

Access Denial

Specified Relative Precision	Required Number of Failures	Required Number of Trials	Test Duration
20%	871	unknown	unknown
50%	145	unknown	unknown
100%	41	unknown	unknown
200%	14	unknown	unknown
500%	6	unknown	unknown

Source Disengagement Denial

Specified Relative Precision	Required Number of Failures	Required Number of Trials	Test Duration
20%	871	unknown	unknown
50%	145	unknown	unknown
100%	41	unknown	unknown
200%	14	unknown	unknown
500%	6	unknown	unknown

Destination Disengagement Denial

Specified Relative Precision	Required Number of Failures	Required Number of Trials	Test Duration
20%	871	unknown	unknown
50%	145	unknown	unknown
100%	41	unknown	unknown
200%	14	unknown	unknown
500%	6	unknown	unknown

* Since no failures were detected, estimates of these values are not possible

Figure 12 (Part 2). Specified precisions and test durations for access-disengagement performance parameters.

blocks) x (number of bytes/block) ≤ 20,480 bytes. The expected Bit Error Probability is also entered in the **default** file. Figure B-1 shows a sample **default** file for the ACTS experiment.

7.2.1 Process the Preliminary Characterization Test

Once the default file has been completed and the test run, the test data can then be processed by the shell script **dopre**. Type

dopre nnnn

where **nnnn** is the number assigned to the preliminary test.

The results are contained in **usr/net/prelim.rpt**. Figure 13 is an example of the first part of this file (again, the results shown are from measurements conducted in Spies et al., 1988). For each user information transfer performance parameter, it lists the number of trials (i.e., blocks), and estimates of the mean, standard deviation, and autocorrelation of lag 1. It also lists the precision achieved at the 95% confidence level.

This information is used (as described in Section 4) to estimate the supported performance parameters that are required to define Transfer Denial Probability.

7.2.2 Select the Precision

The estimate of the standard deviation and autocorrelation of lag 1 is used to determine the sample size required to obtain a specified absolute precision for time parameters (from program **star** which is contained on diskette 5). Similarly the conditional probability of a failure given that a failure occurred in the previous trial is used to determine the number of failures required to obtain a specified relative precision for failure probability parameters. An example of the test durations are shown in Figure 14 (the last part of **usr/net/prelim.rpt**).

There is one trial of User Information Bit Transfer Rate and User Fraction of Input/Output Time per test. Hence, no estimate of absolute precision is possible. To characterize these performance parameters several preliminary tests are required.

Using the values obtained from the preliminary test, the sample size and the time required to complete a test of this size are estimated for all user information transfer performance parameters (except for User Information Bit

PRELIMINARY CHARACTERIZATION TEST

User Information Transfer Parameters

Test Number 2187

Confidence Level = 95%
Time between Access Attempts = 0 s

Time Parameters

Performance Parameter	Number of Trials	Estimate of Mean	Estimate of St. Dev.	Estimate of Autocorr. of Lag 1	Achieved Absolute Precision
Block Transfer Time	33	2.099e+00	9.303e-01	7.524e-01	8.778e-01
U. F. of Block Transfer Time	33	2.021e-02	1.267e-03	1.498e-01	2.483e-03
U. I. Bit Transfer Rate	1	5.981e+03	N/A	N/A	N/A
U. F. Input/Output Time	1	6.955e-02	N/A	N/A	N/A

Failure Probability Parameters
Number of Trials = 33

Performance Parameter	Number of Failures	Estimate of Proportion	Estimate of Autocorr. of Lag 1	Achieved Relative Precision
Bit Error	0	0.000e+00	0.000e+00	Indet.
Extra Bit	0	0.000e+00	0.000e+00	Indet.
Bit Loss	24975	1.524e-01	1.000e+00	218%
Block Error	2	5.714e-02	-6.061e-02	156%
Extra Block	0	0.000e+00	0.000e+00	Indet.
Block Loss	5	1.250e-01	7.949e-01	237%
Transfer Denial	2	5.000e-01	3.333e-01	91%

Figure 13. Results of a user information transfer preliminary characterization test.

REQUIRED SAMPLE SIZE AND TEST DURATION

User Information Transfer Parameters

Confidence Level = 95%
Time between Block Transfer Attempts = 0 s

Block Transfer Time

Specified Absolute Precision	Required Number of Delays	Test Duration
1.756e-01 (seconds)	765	14.01m
3.511e-01	192	3.52m
5.267e-01	85	1.56m
7.022e-01	50	54.96s
8.778e-01	36	39.57s

Bit Error

Specified Relative Precision	Required Number of Failures	Required Number of Trials	Test Duration
20%	871	unknown*	unknown
50%	145	unknown	unknown
100%	41	unknown	unknown
200%	14	unknown	unknown
500%	6	unknown	unknown

Extra Bit

Specified Relative Precision	Required Number of Failures	Required Number of Trials	Test Duration
20%	871	unknown	unknown
50%	145	unknown	unknown
100%	41	unknown	unknown
200%	14	unknown	unknown
500%	6	unknown	unknown

Bit Loss

Specified Relative Precision	Required Number of Failures	Required Number of Trials	Test Duration
20%	64499683	42309650	6.31h
50%	10677797	7004703	62.66m
100%	297941	1954480	17.48m
200%	102912	675098	6.04m
500%	42875	281258	2.52m

* Since no failures were detected, estimates of these values are not possible

Figure 14 (Part 1). Specified precisions and test durations for user information transfer performance parameters.

Block Error

Specified Relative Precision	Required Number of Failures	Required Number of Trials	Test Duration
20%	97	1697	31.09m
50%	17	297	5.44m
100%	5	87	1.59m
200%	2	34	27.37s
500%	1	17	18.69s

Extra Block

Specified Relative Precision	Number of Required Failures	Required Number of Trials	Test Duration
20%	871	unknown	unknown
50%	145	unknown	unknown
100%	41	unknown	unknown
200%	14	unknown	unknown
500%	6	unknown	unknown

Block Loss

Specified Relative Precision	Required Number of Failures	Required Number of Trials	Test Duration
20%	981	7848	2.40h
50%	163	1304	23.89m
100%	46	368	6.74m
200%	16	128	2.34m
500%	7	56	61.56s

Transfer sample is 8 blocks

Transfer Denial

Specified Relative Precision	Required Number of Failures	Required Number of Trials	Test Duration
20%	484	968	2.36h
50%	81	162	23.74m
100%	23	46	6.74m
200%	8	16	2.34m
500%	4	8	70.35s

Figure 14 (Part 2). Specified precisions and test durations for user information transfer performance parameters.

Transfer Rate and User Fraction of Input/Output Time); they are determined for various precisions at the 95% confidence level.[25]

The duration of tests of User Information Bit Transfer Rate and User Fraction of Input/Output Time can be determined in the following way, using User Information Bit Transfer Rate as the guide:

- Determine the sample size required to achieve the specified precision by entering the following values in the interactive program, **star**.
- Specify an absolute precision.
- Estimate the standard deviation from the sample (i.e., from the trials from each test).
- Assume the autocorrelation of lag 1 to be zero; this is probably true since each trial comes from a different test.
- Determine the duration of the tests: the product of the sample size and the duration of an average test. (The beginning and ending times for each test are listed in file **pretimes.xfr**.)

The sample size required to estimate a performance parameter with the specified precision can be determined from program **star**. To use this program, the operator must enter

- **star**
- **0** (to determine the sample size),
- the number corresponding to the type of random variable of the performance parameter (delay, rate, or failure probability),
- the selected confidence level,
- the specified precision (relative or absolute, depending upon the type of performance parameter), and
- some information about the dispersion of the population (e.g., the estimate of the maximum standard deviation and the dependence between trials as measured by the autocorrelation of lag 1).

---

[25]Estimates of all bit and block failure probabilities are listed in this table, although it is likely that some values are too small to be estimated accurately by a sample of this size.

The program returns

- the sample size necessary to achieve the specified precision (if that is desired), and

- A code number to be entered for subsequent analysis of the test.

8. SUMMARY

ANS X3.102 defines a set of 24 user-oriented performance parameters that can define a data communication system or service, independent of topology, protocol, code, or design. ANS X3.141 specifies the methods to measure these performance parameters and, thus, accurately evaluate the performance of data communication systems and services. These standards can also evaluate the performance of systems employing satellite switching and transmission facilities, such as ACTS.

Performance parameters are defined in the following way:

- Data communication systems operate according to a sequence of commands and expected responses (protocols).
- An interface event occurs when a signal (such as a command or response) crosses a user/system interface.
- Those interface events that have common significance and also are performance-significant are given a common name. The name is, therefore, system-independent. These interface events are called reference events.
- Reference events are monitored and recorded along with their time of occurrence.
- Transferred and received data are also recorded and compared.

Information about the reference events and the compared data are used to estimate the performance parameters.

This report has shown how to design experiments to quantify communication satellite system performance using these two standards. The experiment design for ACTS is summarized in Table 11 for the access-disengagement performance parameters and in Table 12 for the user information transfer performance parameters.

The objective of the experiment, to a large degree, determines the analysis that will be performed (Table 7). In this case, the objective is quantification (or characterization). Therefore, the preferred analysis is estimation.

The most important parameter from each type of test must be selected so that each test contains the number of trials that will obtain the specified precision for that parameter at a given confidence level; the remaining parameters will be measured with precisions commensurate with their dispersions. The most important performance parameter for access-disengagement tests is

Table 11. Experiment Design for Access-Disengagement Performance Parameters

Objective of Experiment	Characterization
Type of Analysis	Estimation
Design	Full Factorial
Most Important performance Parameter	Access Time
Confidence Level	95%
Specified Precision for Most Important Parameter	50%
Variable Conditions & Levels	● Service: B-channel ckt switched (S_1), B-channel Packet switched (S_2), D-channel Packet switched (S_3), Low-speed Data (S_4) ● Traffic: Busy Period (T_1), Non-Busy Period (T_2) ● Propagation Path: bou-bou (P_1), bou-dfw (P_2), bou-cle (P_3)

Combinations of Levels of Variable Conditions

(S_1, P_1, T_1)	(S_3, P_3, T_1)	(S_2, P_2, T_2)
(S_1, P_2, T_1)	(S_4, P_1, T_1)	(S_2, P_3, T_2)
(S_1, P_3, T_1)	(S_4, P_2, T_1)	(S_2, P_1, T_2)
(S_2, P_1, T_1)	(S_4, P_3, T_1)	(S_3, P_2, T_2)
(S_2, P_2, T_1)	(S_1, P_1, T_2)	(S_3, P_3, T_2)
(S_2, P_3, T_1)	(S_1, P_2, T_2)	(S_4, P_1, T_2)
(S_3, P_1, T_1)	(S_1, P_3, T_2)	(S_4, P_2, T_2)
(S_3, P_2, T_1)	(S_2, P_1, T_2)	(S_4, P_3, T_2)

Table 12. Experiment Design for User Information Transfer Performance Parameters

Objective of Experiment	Characterization
Type of Analysis	Estimation
Design	Fractional Factorial
Most Important performance Parameter	Bit Error Probability
Confidence Level	95%
Specified Precision for Most Important Parameter	50%
Variable Conditions & Levels	• Service: B-channel ckt SW(S_1), channel Packet SW (S_2), D-channel Packet SW (S_3) Low-speed Data (S_4) • Traffic: Busy Period (T_1), Non-Busy Period (T_2) • Propagation Impairment: Rainy (I_1), Clear (I_2) • Propagation Path: bou-bou (P_1), bou-dfw (P_2), bou-cle (P_3)

Combinations of Levels of Variable Conditions:

(S_1, P_1, T_1, I_2)	(S_2, P_2, T_1, I_2)	(S_3, P_3, T_1, I_2)
(S_1, P_1, T_2, I_1)	(S_2, P_2, T_2, I_1)	(S_3, P_3, T_2, I_1)
(S_1, P_2, T_1, I_2)	(S_2, P_3, T_1, I_1)	(S_4, P_1, T_1, I_2)
(S_1, P_2, T_2, I_2)	(S_2, P_3, T_2, I_1)	(S_4, P_1, T_2, I_2)
(S_1, P_3, T_1, I_2)	(S_3, P_1, T_1, I_1)	(S_4, P_2, T_1, I_1)
(S_1, P_3, T_2, I_2)	(S_3, P_1, T_2, I_2)	(S_4, P_2, T_2, I_2)
(S_2, P_1, T_1, I_2)	(S_3, P_2, T_1, I_2)	(S_4, P_3, T_1, I_2)
(S_2, P_1, T_2, I_2)	(S_3, P_2, T_2, I_1)	(S_4, P_3, T_2, I_1)

Access Time, and the most important performance parameter for user information transfer tests is Bit Error Probability. (User Information Bit Transfer Rate is also an important performance parameter for user information transfer, but it is not selected as most important because the single observation obtained from each test precludes measuring its precision.)

The population of each communication function is defined by a set of fixed and variable conditions. The fixed conditions should be known, but it is the variable conditions that require designing the experiment.

Some of the fixed conditions are specified performance values for performance parameters. Generally, they specify when a delay is so excessive as to be considered a failure to perform. These values are, of course, subjective, and their selection will influence the perceived performance of the data communication system.

The explicit variable conditions for access-disengagement tests are Service, Traffic, and Propagation Path. The explicit variable conditions for user information transfer tests are Service, Traffic, Propagation Path, Propagation Impairment, and Packet Size.

Probably not much is initially known about the population of the performance parameters. A preliminary characterization test will provide some indication of the number of trials required to achieve the specified precision. That is, for each performance parameter, the preliminary characterization test provides

- some statistics required to estimate the distribution and dependence, and
- a set of precisions from which one is to be selected. To aid the selection, listed with each precision is
  - the number of trials required to achieve that precision, and
  - the test duration required to achieve that precision.

A considerable amount of the time required to conduct an experiment is spent redesigning it, so more than one preliminary characterization test may be desirable. With the knowledge obtained from these preliminary tests, the number of combinations of levels of variable conditions and the desired precision might be adjusted to match the time or resources available.

The levels of each variable condition should be judiciously selected. The ideal design is the (full) factorial design because all combinations of levels of variable conditions can be tested (possibly with replication, since some known precision is required by ANS X3.141). If a factorial design is impractical, fractional factorial are available. For this experiment, the full factorial design requires 24 access-disengagement tests. The full factorial design of user information tests require 48 tests. Even if 48 tests is considered to be manageable (considering cost), the fact that they involve the non-controllable variable condition, Propagation Impairment, probably means that this design is impractical (considering time). Therefore, a balanced randomized block design might be appropriate for user information transfer tests.

Although ANS X3.141 can be implemented by any method that follows its specifications, NTIA has developed a method that has been successfully employed (Spies et al., 1988).[26]

[26]The data extraction portion of the NTIA implementation must be modified for this or any other data communication systems.

9. ACKNOWLEDGMENTS

The authors wish to thank Mr. Ken Spies for his help and guidance in developing an understanding of the system performance software capabilities and its application to the ACTS, and Mr. John Harman for his help in making all the figures and tables presentable. The authors also wish to thank Drs. Edwin Crow and Robert Kubicheck for their technical review, Ms. Marla Mahoney for her editorial review, and especially Ms. Kathy Mayeda for preparing the final form of the report.

10. REFERENCES

ANSI (1983), American National Standard for Information Systems - Data communication systems and services - user-oriented performance parameters, X3.102-1983 (American National Standards Institute, New York, NY).

ANSI (1987), American National Standard for Information Systems - Data communication systems and services - measurement methods for user-oriented performance evaluation, X3.141-1987 (American National Standards Institute, New York, NY).

Cass, R. D., (1988), Suitability of ANSI standards for quantifying communication satellite system performance, NTIA report 88-234, August (NTIS Order Number 89-113526/AS).

Lovell, R. R., and C. L. Cuccia (1984), Type C communications satellites being developed for the future, Microwave Sys. News 14, No. 3, March.

Moses, L. E., and R. V. Oakford (1963), Tables of Random Permutations (Stanford University Press, Stanford, CA)

Naderi, F. M., and S. J. Campanella (1988), NASA's advanced communications technology satellite (ACTS): an overview of the satellite, the network, and the underlying technology, Proc. AIAA 12th Int. Commun. Satellite Systems Conf., March.

Seitz, N. B., and D. S. Grubb (1983), American National Standard X3.102 user reference manual, NTIA Report 83-125, October (NTIS Order Number PB 84-155571).

Spies, K. P., D. R. Wortendyke, E. L. Crow, M. J. Miles, E. A. Quincy, and N. B. Seitz (1988), User-oriented communication services: measurement design, conduct, and results, NTIA report 88-238, August (NTIS Order Number PB 89-117519/AS).

APPENDIX A: NTIA CODE

The Version 2.1 of the NTIA software code consists of many programs, subprograms, shell scripts, and processes that are written in FORTRAN 77 and C programming languages (Kernighan and Richie, 1978). Shell scripts in both the AT&T Bourne shell and the Berkeley C shell link most of the software into an automatic process on each PC. The source code is not reproduced in this report but is contained in a set of seven PC-DOS™ and MS-DOS™ 5-1/4" diskettes available from NTIA.[27]

A.1 Diskettes

The software is designed to operate under the UNIX™ system V.3 operating system and some U.C. Berkeley enhancements, namely the C shell. NTIA used MicroPort 386 Version 3.0e (or later).

- Diskette #1 (Sample Data Files): This diskette contains ASCII text files of sample data and some documentation.
- Diskette #2 (Experiment Design): The code is both FORTRAN 77 and C language. The FORTRAN program **star** is the same as that on diskette #5 and is not included on this diskette.
- Diskette #3a (Data Extraction): The code is C language. It is system-dependent because it uses system calls that may be unique under the current vendor's software.[28]
- Diskette #3b (Data Conversion): The code is C language.
- Diskette #4 (Data Reduction): The code is FORTRAN 77. Due to the complexity of the code, it is recommended that it be run on a fairly fast CPU.
- Diskette #5 (Data Analysis): The code is FORTRAN 77 (i.e., program **star**) except for one C program.
- Diskette #6 (Data Display): The code is C language (it must be modified if neither the HP7475A vector plotter nor an emulator is not used).

[27]PC-DOS is a trademark of International Business Machines, Inc, and MS-DOS is a trademark of Microsoft, Inc.

[28]Future versions will conform to the ANSI C Language Standard and the IEEE POSIX Standard when vendors provide this capability, thus providing portability of the software to other operating systems.

A companion set of demonstration diskettes is also available. These companion diskettes operate in MS-DOS™ or PC-DOS™. Each of the major software modules (except those of data extraction) is included on a demonstration diskette along with a batch file and sample data to perform the calculations. The diskettes and the modules are

- Diskette #2D (Experiment Design): **star**
- Diskette #3bD (Data Conversion): **merge** and **reform**
- Diskette #4D (Data Reduction): **prolog**, **analyz**, and **epilog**
- Diskette #5D(Data Analysis): **star**
- Diskette #6D (Data Display): **vp**, **vpaxes**, **vplegend**, **histogrm**, **box**, and **hpgraph**

To use the demonstration diskettes, place a demonstration disk in a personal computer having a math coprocessor chip, look in a **readme** file for the names of the demonstrations, and type each of the names.

A.2 Install The Software

The software from the seven diskettes must be installed in both the source and destination computers. In order to install the software, the UNIX™ system must have a **/dev/dos** directory that contains disks linked to the proper I/O devices:

- Disk drive A is linked to a high density (1.2MB) I/O device.
- Disk drive B is linked to a low density (360KB) I/O device.
- Hard disk C is partitioned for DOS™.

Each computer should have a Microport System V.3 operating system (version 3.0e or later) with a FORTRAN 77 compiler.[29]

To install the software, follow these steps:

A. Log in by typing **net**.

B. Enter the password. The default password is **test**.

---

[29]If another operating system is used, the command **doscp** must be replaced with an appropriate command and **install** must be edited. If the FORTRAN 77 compiler is not available, either create a C shell F77 to invoke your compiler or alias your F77 compiler.

C. Insert diskette #1 in drive B.

D. Type **doscp B: install.** and press return.

E. Type **sh install.**[30] Follow these instructions from the screen:

- log out by typing **logout** or ^D,
- have the superuser log in,
- have the superuser change the directory to **net**'s home directory,
- have the superuser type **Rootdir**,[31] and
- have the superuser log out by typing **logout** or ^D.

F. Log in by typing **net**

G. Type **sh install**. Now **install** requests that diskettes #2 - #6 be inserted. When **install** is finished, the message **install complete** appears on the screen.

H. Type **make all**. This command compiles the programs by invoking **make** in each directory. The procedure may require several minutes.

I. Type **make move**. This command moves the compiled programs to directory **~net/bin**.

J. Type **make clean**. This command removes unnecessary files created by **make**.

K. Type **rehash**. This command renders the programs available.

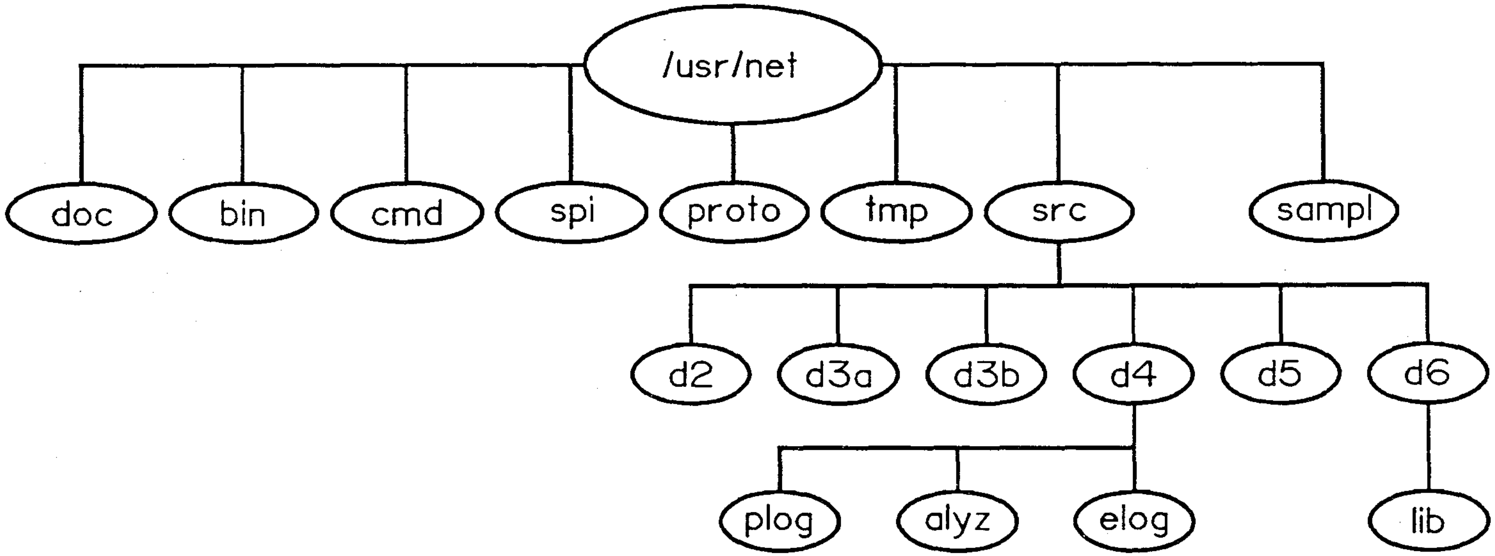
Installation is now complete, and the files have been placed in the appropriate directories.

A.3 Directories

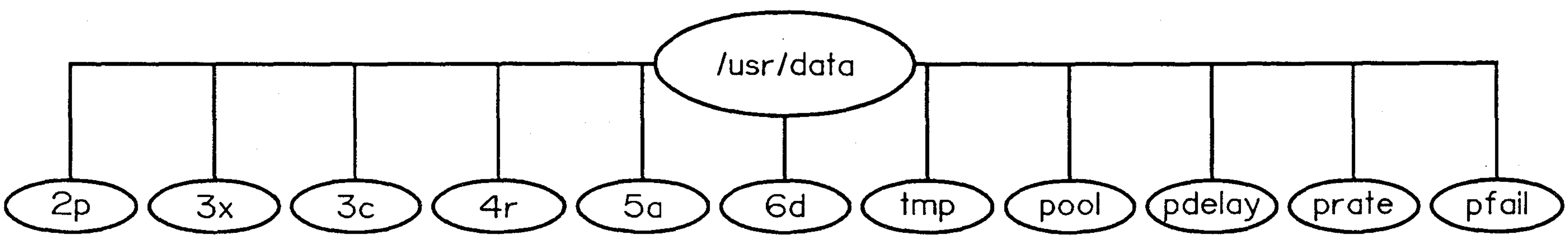
The NTIA software has been placed in two directories, **/usr/net** and **/usr/data**. The first directory is the home directory (i.e., the directory in which you are placed when you log on). It contains the software necessary to process a test. The second directory will contain data from the experiment. The structures of these directories are shown in Figure A-1.a and A-1.b,

[30]The default shell is the Berkeley C shell.

[31]This program creates subdirectory **data** in the **net** parent directory.



a. home directory



b. data directory

Figure A-1. Directory structure of NTIA code.

respectively. Figure A-2 is a list of files contained in each directory and subdirectory.

A.3.1 Directory /usr/net

Log on as **/usr/net** (or **~net**). The default password is **test.**[32] This directory contains programs, shell scripts, and files necessary to process each test. It has six subdirectories.

doc: software documentation

bin: compiled programs

cmd: shell script files

spi: specified performance values files

proto: protocol files

tmp: temporary (scratch) files

src: diskettes 2 - 6. Each diskette should have an **all** option. This subdirectory contains five other subdirectories of programs and shell scripts:

- d2: source code for experiment design FORTRAN 77 program
- **d3a**: source code for data extraction (on-line) C programs
- **d3b**: source code for data conversion (off-line) C programs
- d4: source code for data reduction FORTRAN 77 programs
  - **plog**: program to check for errors and consolidate specifications and overhead information
  - **alyz**: program to identify next trial and determine outcome
  - elog: program to format results
- d5: source code for data analysis FORTRAN program (i.e., **star**)
- d6: source code for data display C programs

[32] Before use, this password should be changed to a word that is not listed in a dictionary. The command **passwd** permits this.

```
Makedir      cmd/         install.doc     preface.doc   spi/
Makefile     codes        log             preface.r     src/
Rootdir      data.x       log.bak         preface.x     sys/
bin/         doc/         net-vm96.doc    proto/        tmp/
clkcal       install      netcodes        sampl/

./bin:
acctime*     gentem*      mkpre*      recv*       show-o*     tablt-a*    vp*
analyz*      histogrm*    mkvpleg*    reform*     st*         tablt-x*    vpaxes*
batchid*     hpgraph*     movit*      relate*     star*       truetime*   vplegend*
calibr*      merge*       prolog*     settime*    swappit*    tt*         vpz*
epilog*      mkdate*      qklog*      show-h*     tablf-a*    tweak-h*    xfertime*
firsttwo*    mklog*       quartile*   show-hd*    tablf-x*    tweak-o*    xmit_net*

./cmd:
cksum*       fail*        mkhist*     pretabt*    reduc*      runxt.tst*
clear*       fail-a*      mkinfo*     prmt-a*     reduc-a*    runxtf*
clkcal*      fail-x*      mktab*      prmt-x*     reduc-x*    runxtf.tst*
conver*      faildev*     mktem*      qkhist*     repeat*     time-a*
delay*       fileck*      mkvp*       qklook*     runr*       time-x*
display*     head*        move.rcv*   qktab*      runumb*     tweakall*
do*          junk         move.xmt*   qkvp*       runx*       tweaknon*
dopre*       mean-dev*    mover*      qp*         runx.tst*
doqik*       meanrdev*    movex*      quarter*    runxf*
envvar*      mkasci*      pretabf*    rate*       runxt*

./doc:
cover.doc      cshrc          install.doc    logout         promotn.doc
cover1.doc     info-d1.doc    kermrc         profile        roadmap.doc

./proto:
net-bak.bol    net-vm96.bol   net-xs96.bol   xmt-vm19.bol   xmt-xs96.bol
net-vm19.bol   net-vm96.doc   old            xmt-vm96.bol

./sampl:
1424.chr       1424info.or    1424xfr.tab    cor            scr
1424.leg       1424info.ort   b1fail         csp            soi
1424.men       1424info.ox    b1o            data.x         spi
1424.tem       1424info.oxt   b2fail         doi            spi.acd
1424data.r     1424log.r      b2o            dui            spi.xfr
1424info       1424log.x      b3fail         log            sts
1424info.hr    1424logn.x     b3o            log.wrk        sui
1424info.hrt   1424mrs.fpr    b4o            par            sum
1424info.hx    1424mrs.tim    clkdata.cal    preface.r      thrput
1424info.hxt   1424xfr.his    coi            preface.x

./spi
spi            spi.acd        spi.xfr

./scr:
d2/            d3a/           d4/            d5/            d6/

./src/d2:
Makefile       mkpre.c

./src/d3a:
```

Figure A-2 (Part 1). Files in each subdirectory of usr/net.

Makefile	clockusr.h	makefile.mp	pc.c	show-o.c
calibr.c	comsubs.c	makefile.pc	pdn_test.h	truetime.c
clksubic.c	connect.c	makefile.sco	recv.c	tty.c
clksubs.c	files.dir	maketime	settime.c	xmit_net.c
clock.c	io.h	mklog.c	show-h.c	
clock.h	jtom.c	parsecmd.c	show-hd.c	

./src/d3b:

Makefile	firsttwo.c	reform.c	tablf-a.c	tablt-x.c	xfertime.c
acctime.c	merge.c	relare.c	tablf-x.c	tweak-h.c	
batchid.c	mkdate.c	swappit.c	tablt-a.c	tweak-o.c	

./src/d4:

Makefile	alyz/	elog/	plog/

./src/d4/alyz:

Makefile	analyz.f	bitcor.f	dshift.f	exsbit.f	rdspec.f	ttsamp.f
access.f	ancill.f	catgap.f	dwload.f	exsrec.f	sshift.f	uiotim.f
aliput.f	assess.f	compar.f	exdbit.f	instst.f	swload.f	wrpars.f
aloput.f	bitcop.f	diseng.f	exdrec.f	orstat.f	transf.f	wrstat.f

./src/4d/elog:

Makefile	dattim.f	efchar.f	rdcsum.f	rdstat.f
aspref.f	dfchar.f	eliput.f	rdpars.f	summjry.f
asumry.f	dsumry.f	epilog.f	rdspec.f	tsumry.f

./src/d4/plog:

Makefile	ckspec.f	conrsp.f	datxam.f
ckinfo.f	conrev.f	datcon.f	prolog.f

./src/d5

Makefile	checkx.f	enterx.f	limit.f	qklog.c	xfile.f
anzflr.f	chisq-r.f	fdist-r.f	main.f	ssdflr.c	yfile.f
anztim.f	entera.f	ftest-r.f	multip.f	ssdtim.f	zfile.f
checki.f	enteri.f	level.f	poiss.f	studnt.f	

./.src/d6:

Makefile	histogrm.c	lib/	quartile.c	vplegend.c
box.c	hpgraph.c	mkvpleg.c	vp.c	vpz.c
gentem.c	junk	movit.c	vpaxes.c	

./src/d6/lib:

arc.c	comment	exitplot.c	plot.h	reldir.c	speed.c
axis.c	daset.c	fill.c	plotchar.c	relsize.c	
circle.c	dir.c	initplot.c	pmove.c	rmove.c	
cirflat.c	dsset.c	label.c	poiflat.c	scale.c	
clabel.c	ellipse.c	line.c	rect.c	size.c	
commands	execplot.c	pen.c	rectang.c	slant.c	

./sys:

print*	turn*

Figure A-2 (Part 2). Files in each subdirectory of usr/net.

./tmp:

2183.cor	2186.sum	2222db2.chr	2222db2.mnu	vol3.txt
2183.sum	2218info	2222db2.his	2222db2.tem	
2186.cor	2221info	2222db2.leg	C2	

./uw:

aces-tes	betime*	myfile	uope
atest.c	betime.c	run	uope.c

Figure A-2 (Part 3). Files in each subdirectory of usr/net.

lib: library C routines for data display

sampl: sample data files

A.3.2 **Directory /usr/data**

This directory contains data from the tests. It has eleven subdirectories:

2p: preliminary characterization test data

3x: extracted binary data (extracted on-line)

3c: converted ASCII data (converted off-line)

4r: reduced ASCII data (i.e., outcomes)

5a: analyzed data (i.e., measurement result summary)

6d: displayed data (i.e., "quick look")

tmp: working directory for post-test processing

pool: working directory for pooling

pdelay: delay performance data files for pooling (e.g., **nnnnnac**).

prate: User Information Bit Transfer Rate and User Fraction of Input/Output Time performance data file for pooling (i.e., **thrput**)

pfail: failure performance data file for pooling (e.g., **failac**).

A.4 Reference

Kernighan, B.W., and D.M. Richie (1978), The C Programming Language (Prentice-Hall, Inc., Englewood Cliffs, NJ).

APPENDIX B: EXPERIMENT DESIGN INFORMATION REQUIRED FOR NTIA SOFTWARE

This appendix contains information required by the NTIA software for the ACTS preliminary characterization test and system performance measurement experiment design. Tables B-1 and B-2 contain a set of conditions identified by O_1, ..., O_8 for access-disengagement (overhead) tests and by U_1, ..., U_9 for user information transfer tests; this experiment leaves two unused conditions for each type of test. The conditions are listed in the order they appear in the multivariate analysis of the NTIA software. **default** and **netcodes** are files (described below) and **runxt** is a shell script.

B.1 File default

The **default** file contains seven lines of data. To the right of each is the full description. Figure B-1 is a sample **default** file for this experiment.

B.2 File Netcodes

The **netcodes** file contains the three character code **sss** for source site and its name, the three character code **ddd** for each destination site and its name, and the four character code **aaaa** for each service and its name. There can be any number of sites and services and they can be listed in any order.

For this experiment, the source site is always Boulder (**bou**). There are three destination sites: Boulder (**bou**), Cleveland[33] (**cle**), and Dallas-Ft. Worth (**dfw**). There are four services: S_1, S_2, S_3, and S_4. Figure B-2 is the resulting **netcodes** file.

B.3 Specification Files

There is one specifications file for each type of test. Data in the third line of this file indicate whether the specifications are for parameters of access-disengagement or user information transfer.

B.3.1 Access-Disengagement Tests

The **spi.acd** file for access-disengagement tests consists of five lines of ASCII characters. Table B-3 shows the character fields, edit descriptors, and contents of each line. Figure B-3 shows the **spi.acd** file for the ACTS preliminary characterization access-disengagement test.

---

[33]Cleveland, OH is the location of NASA's Lewis Research Center.

Table B-1. **Conditions for Access-Disengagement Tests and Their Mode of Entry or Computation**

#	Condition	Where Entered
O_1	Source Site	**default** **netcodes**
O_2	*Service	**runxt** **netcodes**
O_3	Day of Week	(computed by **runxt**)
O_4	Time of Day	(computed by **runxt**)
O_5	Intersession Delay	**default**
O_6	Destination Site	**default** **netcodes**
O_7	-	**runxt**
O_8	-	**runxt**

* indicates primary variable condition.

Table B-2. **Conditions for User Information Transfer Tests and Their Mode of Entry or Computation**

#	Condition	Where Entered
U_1	Source Site	**default** **netcodes**
U_2	*Service	**runxt** **netcodes**
U_3	Day of Week	(computed by **runxt**)
U_4	Time of Day	(computed by **runxt**)
U_5	Interblock Delay	**default**
U_6	Destination Site	**default** **netcodes**
U_7	Block Size	**default**
U_8	-	**runxt**
U_9	-	**runxt**

* indicates primary variable condition.

```
bou    source site
cle    destination site
128    block size (bytes or chars)
---    --xfr info-----
40     number of blocks
0.1    interblock delay (sec)
---    --ovh info-----
20     number of accesses
55     interaccess delay (sec)

ALL lines below this are comments:

NOTE:  Update appropriate preface file(s) to reflect source/dest.
site changes

Maximum sizes permitted as shown -

Maximum number of bytes or characters per test = 20480
Maximum number of blocks per test               =   320
Maximum number of accesses per test             =    40

Block sizes used     Maximum number of blocks or accesses:
        512                  40
        173                 118
        128                 160
         80                 236
         64                 320
```

Figure B-1. Default file for ACTS preliminary characterization test.

```
bou    Boulder, CO
bou    Boulder, CO
cle    Cleveland, OH
dfw    Dallas-Ft. Worth, TX
S1     circuit switched over the B channel
S2     packet service over the B channel
S3     packet service over the D channel
S4     circuit switched, dial-up
```

Figure B-2. Netcodes file for ACTS system performance measurement experiment.

Table B-3. spi.acd File Designation for Access-Disengagement Tests

Line	Character Field	Edit Descriptor	Contents
1	1-32	A32	File Descript: **SPECIFICATIONS INPUT**
2	1-64	A64	Batch Identifier
3	1- 4 5- 8 9-12	I4 I4 I4	Access Assessment Option: 1 U. I. Transfer Assessment Option: 0 Disengagement Assessment Option: 1
4	1-16 17-32	E16.0 E16.0	Access Time U. F. Access Time
5	1-16 17-32 33-48 49-64	E16.0 E16.0 E16.0 E16.0	Source Disengagement Time U. F. Source Disengagement Time Destination Disengagement Time U. F. Destination Disengagement Time

```
SPECIFICATIONS INPUT
ACTS Preliminary Access-Disengagement Test
                  1    0    1
                      3.5E+01      5.00E-01
                      2.0E+01      7.50E-01     4.0E+00      2.0E-01
```

Figure B-3. **spi.acd** file for the ACTS preliminary characterization access-disengagement test.

The first line contains (left-justified) the words **SPECIFICATIONS INPUT.**

The second line contains an identifier that uniquely describes the test. Either leave it blank or enter a place holder in the first 60 columns; the identifier will be supplied by the identifier in the **soi** file, and the four digit test number for columns 61-64 will be supplied by a shell script.

The third line contains the three assessment option codes, 1 or 0 depending upon whether performance parameters of that function are to be analyzed or not, respectively.[34] For access-disengagement tests, the assessment option line is

1 0 1

(A sample **spi.acd** file for user information transfer tests is shown in the following section.)

B.3.2 User Information Transfer Tests

The **spi.xfr** file for user information transfer tests consists of six lines of ASCII characters. Table B-4 shows the character fields, edit descriptors, and contents of each line. Figure B-4 shows the **spi.xfr** file for the ACTS preliminary characterization user information transfer test.

The first line contains (left-justified) the words **SPECIFICATIONS FILE.**

The second line contains an identifier that uniquely describes the test. Either leave it blank or enter a place holder in the first 60 columns; the identifier will be supplied by the identifier in the **sui** file, and the four digit test number for columns 61-64 will be supplied by a shell script. As stated above, the first 60 columns of the identifier must be identical to the batch identifier field in each overhead and user information file for the test.

The third line contains the three assessment option codes. For access-disengagement tests, the assessment option line is

0 1 0

---

[34]The reduction portion of the NTIA software permits these combinations of assessment options: (0,0,1), (0,1,0), (1,0,0), (0,1,1), (1,0,1), (1,1,0), and (1,1,1). However, the NTIA linkage software permits only (1,0,1) for access-disengagement tests and (0,1,0) for user information transfer tests.

Table B-4. **spi.xfr** File for User Information Transfer Tests

Line	Character Field	Edit Descript.	Contents
1	1-32	A32	File Descript: **SPECIFICATIONS INPUT**
2	1-64	A64	Batch Identifier
3	1-4 5-8 9-12	I4 I4 I4	Access Assessment Option: **0** U. I. Transfer Assessment Option: **1** Disengagement Assessment Option: **0**
4	1-16 17-32 33-48 49-64	E16.0 E16.0 E16.0 E16.0	Block Transfer Time U. F. Block Transfer Time U. I. Bit Transfer Rate U. F. Input/Output Time
5	1-16 17-32 33-48 49-64	E16.0 E16.0 E16.0 E16.0	Bit Error Prob. for Transfer Sample Bit Loss Prob. for Transfer Sample Extra Bit Prob. for Transfer Sample Minimum Number of Bit Transfer Attempts for Transfer Sample
6	1-8 9-16 17-24 25-32	I8 F8.0 F8.0 F8.0	User Information Window Size (bits) Maximum Bit Shift in Bit Error Identification Algorithm (bits) Maximum Bit Shift in Undeliver. Bit Identification Algorithm (bits) Maximum Shift in Extra Bit Identification Algorithm (bits)

```
SPECIFICATIONS INPUT
ACTS Preliminary User Information Transfer Test
   0    1    0
             3.0E+00      5.0E-01      9.6E+03      5.0E-01
             1.0E-06      1.0E-08      1.0E-08      4.0E+04
      16          256.         8192.        8192.
```

Figure B-4. **spi.xfr** file for ACTS preliminary characterization user information transfer test.

FORM **NTIA-29**
(4-80)

U.S. DEPARTMENT OF COMMERCE
NAT'L. TELECOMMUNICATIONS AND INFORMATION ADMINISTRATION

BIBLIOGRAPHIC DATA SHEET

	1. PUBLICATION NO.	2. Gov't Accession No.	3. Recipient's Accession No.

4. TITLE AND SUBTITLE
Design of Experiments to Quantify Communication Satellite System Performance

5. Publication Date

6. Performing Organization Code
NTIA/ITS.N1

7. AUTHOR(S)
Robert D. Cass and Martin J. Miles

8. PERFORMING ORGANIZATION NAME AND ADDRESS
National Telecommunications and Information Admin.
Institute for Telecommunication Sciences
325 Broadway
Boulder, CO 80303-3328

9. Project/Task/Work Unit No.

10. Contract/Grant No.

11. Sponsoring Organization Name and Address
National Telecommunications and Information Admin.
Herbert C. Hoover Building
14th and Constitution Avenue, NW
Washington, DC 20230

12. Type of Report and Period Covered

13.

14. SUPPLEMENTARY NOTES

15. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here.)*

This report describes the steps for designing experiments to quantify the performance of a communication satellite system according to the methods specified by ANS X3.141. Performance is described in terms of performance parameters that are user-oriented and system-independent. The design is intended to efficiently obtain estimates of performance parameters that are relatively free of bias and can be stated with known precision. Service, traffic, propagation path, and propagation impairment are the primary factors. Factorial designs are recommended for the experiment.

16. Key Words *(Alphabetical order, separated by semicolons)*

Advanced Communications Technology Satellite; American National Standards; dependence; experiment design; factorial designs; factors; performance parameters; precision; sample size

17. AVAILABILITY STATEMENT	18. Security Class. *(This report)*	20. Number of pages
☒X UNLIMITED.	Unclassified	102
☐ FOR OFFICIAL DISTRIBUTION.	19. Security Class. *(This page)* Unclassified	21. Price:

☆U.S. GOVERNMENT PRINTING OFFICE: 1990-573-002/21017