SimSup’s Loop: A Control Theory Approach to Spacecraft Operator Training

Brandon D. Owens
Stinger Ghaffarian Technologies
NASA Ames Research Center
Mail Stop 240-2
Moffett Field, CA 94035
650-604-0037
brandon.d.owens@nasa.gov

Alan R. Crocker
NASA Ames Research Center
Mail Stop 241-20
Moffett Field, CA 94035
650-604-1698
alan.r.crocker@nasa.gov

Abstract—**Immersive simulation is a staple of training for many complex system operators, including astronauts and ground operators of spacecraft. However, while much has been written about simulators, simulation facilities, and operator certification programs, the topic of how one develops simulation scenarios to train a spacecraft operator is relatively understated in the literature. In this paper, an approach is presented for using control theory as the basis for developing the immersive simulation scenarios for a spacecraft operator training program. The operator is effectively modeled as a high level controller of lower level hardware and software control loops that affect a select set of system state variables. Simulation scenarios are derived from a STAMP-based hazard analysis of the operator’s high and low level control loops. The immersive simulation aspect of the overall training program is characterized by selecting a set of scenarios that expose the operator to the various inadequate control actions that stem from control flaws and inadequate control executions in the different sections of the typical control loop. Results from the application of this approach to the Lunar Atmosphere and Dust Environment Explorer (LADEE) mission are provided through an analysis of the simulation scenarios used for operator training and the actual anomalies that occurred during the mission. The simulation scenarios and inflight anomalies are mapped to specific control flaws and inadequate control executions in the different sections of the typical control loop to illustrate the characteristics of anomalies arising from the different sections of the typical control loop (and why it is important for operators to have exposure to these characteristics). Additionally, similarities between the simulation scenarios and inflight anomalies are highlighted to make the case that the simulation scenarios prepared the operators for the mission.**

TABLE OF CONTENTS

1. INTRODUCTION

Immersive simulation—the act of practicing operational scenarios in mock environments that closely mimic (and usually utilize actual elements of) the operational environment—is an often used and well respected tool for training individuals for complex operations like spaceflight, aviation, power plant operations [1], industrial chemical processing [2], and medical surgery [3]. In the realm of government funded human and robotic spaceflight, immersive simulation is usually a flight readiness requirement. Moreover, simulation scenario developers are often praised in astronaut [4,5] and ground operator memoirs [6-8], and immersive simulations are even dramatized in fiction and non-fiction movies and television series such as *Apollo 13* and *From the Earth to the Moon*. In other words, effective immersive simulation scenarios and their developers are highly valued in complex system operations in general and in spaceflight in particular.

The literature describing simulators, simulation facilities, and simulations for the purpose of engineering analysis is extensive. In fact, the AIAA holds its annual Modeling and Simulation Technologies Conference to promote research in those areas. Moreover, the topics of learning and training are widely researched and documented. Multiple publications offering high level overviews of astronaut and spacecraft ground operator certification programs, for instance, have been released since the 1960s. [9-12] However, these publications do not provide an explicit model for developing simulation scenarios to train spacecraft operators.

In practice, simulation scenarios for spacecraft operator training are often based on what could be called the *Procedure Model*. Under the Procedure Model, the simulation scenario is developed in order to give the trainee an opportunity to execute specific nominal and contingency procedures. Underlying the Procedure Model is the notion that the procedures—which are usually developed with significant forethought and input from the operational community—will be the primary tools for guiding the operator through an operations experience and therefore, the operator should develop a great deal of familiarity with them.

While it is undeniable that spacecraft operators need to be adept at executing procedures, overreliance on the Procedure Model can have several drawbacks. First, the procedures have to be written to correctly cover the operational scenarios as exhaustively as possible and operators have to participate in numerous simulations to practice each variant of each procedure. Second, no matter how complete the procedures may be, the unknown unknowns of spaceflight will create situations that are not covered well by the procedures and thus the operator will have to improvise.

For these reasons, some have advocated for a shift from "task-based" to "skills-based" training. [12] Moreover, spacecraft simulation scenario developers often try to go beyond the scope of the Procedure Model. However, their efforts in this regard are widely considered an art form and left unformalized, thus creating a gap in our knowledge of how to repeatedly develop simulation scenarios that go beyond the Procedure Model.

Accordingly, in this paper the authors propose a formalized model for spacecraft simulation scenario development to complement the Procedure Model. This model treats spacecraft operators as controllers of control loops with generalizable elements that can each be the source of a disruption that opens the control loop. Underlying this model is the notion that exposing operators to disruptions in these general elements and allowing them to go through the action closing the loop in immersive simulations provides them with foundational experience to help them improvise a solution when the counterparts to these elements in other control loops are disrupted.

In the next section, the fundamental concepts of control theory and their applicability to complex systems operations are presented. Then in Section 3, an accident model based on control theory and its associated hazard analysis process are summarized to establish a connection between control theory, system safety, system security, and the role of the complex system operator. In Section 4, SimSup's Loop, an approach to immersive simulation scenario development based on control theory is detailed with examples from the application of this approach to the Lunar Atmosphere and Dust Environment Explorer (LADEE) mission. [13-21] This description of SimSup's Loop is then followed up by a discussion—featuring a case study of actual inflight anomalies during the LADEE mission—of its effectiveness as a training approach. Finally, the paper ends with concluding remarks and comments on the potential for future work in the application of SimSup's Loop to complex system operator training.

2. CONTROL THEORY CONCEPTS

Control theory is applied to technical, social, and economic problems in order to influence the behavior of engineered systems, often with the explicit recognition that the operating conditions and environments of these systems will not be exactly known by the control system designer. As stated in [22]:

> *"The central problem in control is to find a technically feasible way to act on a given process so that the process adheres, as closely as possible to some desired behavior. Furthermore, this approximate behavior should be achieved in the face of uncertainty of process and in the presence of uncontrollable external disturbances acting on the process."*

In the remainder of this section, the key control theory concepts needed to understand why one would apply control theory to complex system operator training are detailed.

System State Variables

A system's state variables are changeable conditions of the system (e.g., pitch of an airplane, velocity of a car, etc.) that determine the system's evolution over time (i.e., its dynamic behavior). Thus, the ability to deliberately affect the system state variables is the ability to *control* its dynamic behavior.

The values of system state variables can be discrete (e.g., the mode of a spacecraft) or continuous (e.g., the cabin pressure of a space station). In principle, they are measureable (e.g., temperature), but direct measurement of them can be practically impossible (e.g., the total number of people with infected with a specific disease). While it is common for control theorists and engineers to focus on system state as it applies to physical (i.e., electromechanical) systems, the concept of system state can also apply to biological, economic, and social systems. [23]

The Control Loop

The control system (also referred to as the control structure or control loop) is a system of logical and physical elements that convert an input describing the desired system state into actions upon the system to be controlled that are intended to achieve or maintain the goal. According to [24], there are three fundamental elements of control systems: controllers, actuators, and observers. The controller is the logic of the control system (stored in electronics, human minds, regulations, procedures, etc.) that determine the control actions. The controller contains (or implicitly assumes) a model of the controlled system. The actuator is the physical object (e.g., reaction wheel) or agent that imposes the intent of the controller on the system by executing the control action. The observer is the element of the control system (e.g., electromechanical sensor and estimation logic, human operator, etc.) that ascertains the system state.

Control systems can take two basic forms: open-loop or closed-loop (also called feedback). In an open-loop control system, the observer is completely passive or non-existent. In a closed-loop control system, the observer actively monitors the system state during the control actions and feeds that information back to the controller so that it may alter its instructions to the actuator(s). The advantage of closed-loop control over open-loop control is that it can allow the control system to correct for uncertainties in controller performance, actuator performance, the controlled process, and the system's operating environment. Effective open-loop control requires a large degree of certainty over these things prior to the control action (i.e., when the control system is designed and implemented). For complex systems, such a degree of certainty is not possible and thus, closed-loop systems are implemented. Even control loops that are advertised as open-loop are often "closed" (albeit on longer timescales) when the boundaries of the control loop are expanded to include the human operators of the systems.

Control Authority

Control authority is a system property determined by the design of the control system that permits the ability of the control system to affect the system state. Control systems (both open- and closed-loop) employ their system's control authority to achieve and maintain the desired system states. The three primary applications for applying a system's control authority are: task (or procedure) execution, disturbance rejection, and adaptation.

Task (or Procedure) Execution

Task or Procedure Execution is the planned alteration of system state under an assumed system environment and goal. Task execution can sometimes be performed effectively with open-loop control, but there is a possibility for unanticipated and undesired interactions between system components and between the system and its environment during task execution. Due to this type of uncertainty, closed loop control can be—and often is—applied in task execution.

Disturbance Rejection

Disturbance rejection is the alteration of system state to nullify undesired changes in the system state caused by external (i.e., environmental) influences on the system. Disturbances are a source of uncertainty because they can be entirely unexpected or expected but not predictable (e.g., it is expected that lightning will strike a skyscraper, but it is not possible to predict when it will strike). Open-loop control systems can reduce the state changes caused by disturbances by adding sources of energy dissipation (e.g., physical barriers, damping, etc) and so forth. However, closed-loop control is usually needed to nullify the changes in system state once they occur. Thus, closed-loop control systems can often be far more effective than open-loop control systems in rejecting disturbances. [25]

Adaptation

Adaptation is the change of system structure or settings in response to changes in context or goals. System adaptation is a source of uncertainty in system operation because it is by definition not planned in the initial architecting of the system. Closed-loop control can allow the system to effectively employ its control authority to maintain desired system states in the response to the system adaptation.

The Interplay of Task Execution, Disturbance Rejection, and Adaptation

The three applications of control authority are not mutually exclusive and thus the design of a control system should often take all of them in account. As noted by [24], disturbances occurring during task execution may need to be rejected, control loops may have to be adapted to compensate for internal inconsistencies in system's design or manufacturing, and control loops may have to be adapted to tolerate (or even take advantage of) disturbances.

A human operator's role in a controlled system

Complex systems, such as spacecraft, usually have interactions (both internally and with their operating environment) that can only be understood during their operation. Additionally, the operating environment and the goals for the system can evolve during their operation. In order to effectively deal with these sources of uncertainty, an extensive amount of closed-loop control is necessary. However, building hardware and software closed-loop control into these systems is not always sufficient, particularly when the design of these systems have to be frozen prior to operations. Thus, it could be said that the role of human operators in complex systems is to close the control loops that could not be closed by the system design. Indeed, control theoretic models (such as the Crossover Model) are commonly used to analyze and design for human operator performance in such systems. [26]

In some cases, operators must close control loops through supervision, guidance, and updating/upgrading of hardware and software control loops designed into the system. In other cases, they must close loops that the system designers explicitly intended for them to close. Finally, in other cases, they must close loops that designers never recognized by using whatever control authority they have at their disposal. In other words, complex system operators should operate their system with an understanding—at least on an implicit level—of the state variables that they need to control, the control authority that they have over those variables, and the different applications for that control authority (i.e., task execution, disturbance rejection, and adaptation).

3. STAMP AND STPA

In the previous section, the authors linked the role of complex system operators to control theory. In this section, the authors focus in on what operators—particularly spacecraft operators—attempt to control and provide background information on a model and analysis technique that can be applied to achieve such control.

Safety and Security

In most organizational structures, the operational objectives are generally determined by individuals (e.g., managers, politicians, customers, etc.) other than the actual operators. Additionally, many operators are often hired in the later stages of the development and deployment of complex systems, after many of the goals for the system (i.e., its *mission*) are defined. Thus, even though some operators can be (and should be) involved in setting mission goals, it could be said that operators are often handed a set of goals to achieve and any failure to achieve those goals can be considered a *loss*.

Leveson [27] defines *safety* as, "*freedom from accidents and losses*." NASA [28, 29] and the DoD [30] define safety similarly. *Security* is a closely related system property in that it is also related to freedom from loss events—albeit intentional loss events. [31]

These broad definitions for safety and security and the fact that operators are expected to prevent pre-defined losses that narrowly scope their work suggest that system safety and security are the major (if not only) roles of spacecraft operators. This connection of safety and security to the roles of the operators—and the aforementioned connection between control theory and role of operators—raises the question of how one can control safety and security?

STAMP

The Systems Theoretic Accident Model and Processes (STAMP) framework [32, 33] is an accident model that applies the control theory paradigm of system state management to safety. Hazards are defined in terms of unsafe system states, constraints are identified to restrict the hazards, and a safety control structure is created and operated to enforce the constraints. If a safety control structure is unable to enforce the constraints—due to ineffective design or adaptations within the system or in its environment—the system drifts into the hazard state, which would allow accidents or loss events to occur when certain uncontrollable conditions are present in the system's operating environment.

STPA

The Systems Theoretic Process Analysis (STPA) is an approach to hazard analysis based on STAMP. [33] It provides a generalizable taxonomy of inadequate control actions/executions and the flaws in the safety control structure that cause them. While the ultimate result of the analysis depends on the engineering expertise of the analyst, this taxonomy serves as a guide for the analyst to apply his or her engineering expertise to identify problems with the safety control structure in a more repeatable and complete manner.

There are several ways to address problems with the safety control structure that are identified via STPA. Throughout all stages of system design—but in the early stages in particular—the results of STPA can be used to influence design decisions through safety-guided design processes. [33-35] Alternatively, if the problem is discovered when it is too late to be fixed in design or it otherwise *must* be accepted in the operational system, the knowledge gained from the STPA can be applied to operator training to mitigate the risk.

STAMP, STPA, and Spacecraft Simulation Scenario Development for Operator Training

It is sometimes said that hazard (or threat) analysis is like investigating an accident (or security breach) before it occurs. [33] A similar statement could be made about immersive simulation scenario development for operator training. The simulation scenario developer seeks to create a problematic situation—that can develop into a loss event—that can be safely practiced (ideally before it occurs). In other words, opportunities abound for the use of hazard analysis techniques in the development of immersive simulation scenarios for operator training.

With their system-level focus on loss prevention, STAMP and STPA are applicable to problems of safety and security [31], which are issues of major concern to spacecraft operators. Additionally their conceptual foundation is in control theory, which the authors argue is fundamentally connected to the role of human operators of spacecraft systems. Finally, operator training provides an area of application for knowledge derived from STPA when that knowledge cannot be applied to the system design. Therefore, in the next section, the authors present an approach for using STAMP and STPA in the development of immersive simulation scenarios for spacecraft operator training.

4. "SIMSUP'S LOOP"

As mentioned above, this section contains a description of a process for using STAMP and STPA for the development of immersive simulation scenarios for spacecraft operator training. The process culminates in the development of training curriculum for a spacecraft operator that consists of multiple immersive simulations that expose the trainee to issues arising in each of the general components of a control loop. The curriculum (i.e., *SimSup's Loop*) is named after the generic control loop and the callsign used in many human and robotic spaceflight programs for the individual responsible for ensuring that the simulation achieves its objectives: SimSup (shorthand for Simulation Supervisor).

STPA can be applied at any stage of a system's lifecycle [33] and is most effective at influencing the system design if it is applied in the early phases of the system's design. Thus, it is possible that by the time that simulation scenarios need to be developed for operator training, an STPA would have already been performed and could be leveraged directly for the development of the scenarios. Moreover, the STPA and the design decisions influenced by it might have reduced the level of risk in mission operations, thereby reducing the requirements for operator training. However, for missions that are already in their operations phase or far along in their development without an STPA, there will be a need to start a new STPA on a mature operations concept.

Therefore, the steps laid out in this section are written in a manner that assumes that the simulation scenario developer is hired onto the mission several months before the start of the operator training campaign and that no prior STPA has been performed on the system. This assumption is in line with the manner in which operator training program was developed and executed for the LADEE mission. Accordingly, examples from the LADEE experience are provided with the description of each step.

Step 1: Define the accidents (loss events)

The initial step in the process is to identify the accidents or loss events in a general sense (i.e., without implicitly specifying the cause of the events). These loss events should—to the extent possible—cover a broad range of the concerns of the system stakeholders. For example, NPR-8715C [28] identifies the following loss events of interest to NASA:

1. General public death, injury, or illness.
2. Astronaut death, injury, or illness
3. Ground crew and other workforce death, injury, or illness.
4. Earth contamination.
5. Planetary contamination.
6. Loss of, or damage to, flight systems.
7. Loss or, or damage to, ground assets.

ACC1. Humans and/or human assets on Earth are killed/damaged (↓H4).

ACC2. Humans and/or human assets off of the Earth (e.g., ISS, historic lunar landing sites, etc.) are killed/damaged (↓H5).

ACC3. The payload data corresponding to the mission goals are not collected (↓H1).

ACC4. The payload data corresponding to the mission goals is rendered unusable (i.e., deleted and/or corrupted) before it can be fully investigated (↓H2, H3).

ACC5. An incident during this mission directly causes another mission to fail to collect, return, and/or use the payload data corresponding to its mission goals (↓H4, H5, H6).

Figure 1. Defined system accidents for the LADEE application.

The list of accidents used for the LADEE application—shown in Figure 1—were heavily derived from an STPA conducted for the early conceptual design of an Outer Planet exploration mission [35] and should be applicable to a wide range of spacecraft missions.

Step 2: Define the hazard state variables

The next step is to define the system hazards in terms of system state variables. These hazard definitions should be largely based on the accidents and should not take low-level aspects of the system design into account. As noted by [32], safety is an emergent property of systems that results from the interaction of system components and their environment, and it is therefore meaningless to define hazards in terms on the system components alone.

Moreover, these definitions should describe conditions—preferably controllable ones—rather than events. In STAMP, hazards are system states that can lead to loss events in certain system contexts. The occurrence of a system level hazard does not necessarily mean that a loss event will occur.

Figure 2 contains the list of the hazards used for the LADEE application. Like the accidents in Figure 1, they too were heavily derived from [35].

H1. Inability of mission to collect payload data (↑ACC3), (→SC1).

H2. Inability of mission to return collected payload data (↑ACC4), (→SC2).

H3. Inability of mission payload investigators to use returned data (↑ACC4), (→SC3).

H4. Exposure of Earth life or human assets on Earth to toxic, radioactive, and/or energetic elements of mission hardware (↑ACC1, ACC5), (→SC4, SC5).

H5. Exposure of Earth life or human assets off Earth to toxic, radioactive, and/or energetic elements of mission hardware (↑ACC2, ACC5), (→SC6).

H6. Inability of other space exploration missions to use shared space exploration infrastructure to collect, return, and/or use data (↑ACC5), (→SC5, SC6, SC7).

Figure 2. Defined hazards for the LADEE application.

Step 3: Define the high-level safety constraints

Once the hazards have been defined, the next step is to define the high-level safety constraints. These constraints should be defined to prevent the hazards from creating loss events. That said, the safety constraints should not over constrain the system by restricting any level of the hazards from occurring at any time. The overall list of hazards will likely create tradeoffs where some degree of a hazard may have to be accepted at certain times in order to prevent other hazards from occurring (e.g., data collection may have to be temporarily inhibited in order to perform a collision avoidance maneuver). Therefore, high-level aspects of the system design and the notions of accepted risks and risk priorities can factor into the definition of the high-level safety constraints.

The list of high-level safety constraints used for the LADEE application is shown in Figure 3. Once again, this list was largely derived from the early conceptual design of an Outer Planet exploration mission. [35] These constraints were primarily written to constrain system behavior during mission operations (additional constraints could be written to constrain the behavior during the other mission phases). Thus, the constraints related hazards H4-H6 simply relate to control of the launch vehicle and spacecraft trajectory, which must intentionally move along a collision course with Earth (and potentially human assets off of Earth) at certain times during the mission.

SC1. The mission must have the necessary functionality for payload data acquisition at the required times (←H1)

SC2. The mission must be able to return data at the required times (←H2)

SC3. The mission payload investigators must be able to use the data from the mission at the required times (←H3)

SC4. All physical elements of the mission must not unintentionally move along a collision course with Earth (←H4)

SC5. All physical elements of the mission that intentionally collide with the Earth must not cause damage to humans or human assets (←H4, H6)

SC6. All physical elements of the mission must not unintentionally move along a collision course with humans or human assets that are off of Earth (e.g., ISS, historic lunar landing sites, etc.) (←H5, H6)

SC7. The mission must not deny usage of shared space exploration infrastructure to another mission if such a denial would jeopardize that mission's goals (This constraint does not necessarily apply if the mission's goals are similarly or more severely jeopardized) (←H6)

Figure 3. Defined high-level safety constraints for the LADEE application.

Step 4: Define the lower level safety constraints per the existing functional decomposition and design

If one were to perform the process described in this section during the early stages of the design of the system or its operational concept, he or she would use the safety constraints defined in Step 3. However as mentioned above, the process described in this section is assumed to start shortly before operator training begins and thus, the system design and functional decomposition of operator roles are mature by the time Step 3 is complete. Therefore, the next step is to evaluate the existing functional decomposition and design against the high-level safety constraints and accordingly define lower level safety constraints on the system operators.

The lower level safety constraints ultimately trace back to the system hazards and should constrain the system further than the high-level safety constraints. Moreover, by this point in the process they should be detailed enough to be of direct interest specific operators—perhaps to point where they could become flight rules for the operator. As an example, the lower level safety constraints defined for the Guidance, Navigation, and Control (GNC) operator in the LADEE application are provided in Figure 4.

GNC-SC1. GNC must not permit excursions from acceptable maneuver attitudes during maneuvers (←H1, H4, H5, H6).

GNC-SC2. GNC must distribute solar radiation exposure around the spacecraft's surface by limiting the duration of time that the spacecraft spends in fixed attitudes (←H1, H2, H4, H5, H6)

GNC-SC3. GNC must not permit excursions from acceptable communications attitudes when communications with the ground segment are desired (←H1, H2, H4, H5, H6).

GNC-SC4. GNC must not permit excursions from acceptable payload attitudes during payload data collection opportunities (←H1)

GNC-SC5. GNC must limit inertial loads due to angular acceleration on its components and the spacecraft as a whole (←H1, H2, H4, H5, H6)

GNC-SC6. GNC must ensure that the operation of his/her subsystem components do not pose a non-justifiable, direct threat to other system components (←H1, H2, H3, H4, H5, H6)

Figure 4. Defined lower level safety constraints for the GNC operator in the LADEE application.

Step 5: Identify the state variables that can be used to evaluate whether or not the safety constraints are met

With the lower level safety constraints defined, the next step is to identify the state variables that should be controlled to enforce the safety constraints. In doing so, the analyst should strive to only identify the state variables that directly lead to violations of the safety constraints if inadequately controlled. The state variables that are indirectly related to the violation of the lower level safety constraints may be identified in Step 7. That said, if the analyst deems other state variables to be important enough, it is possible to reiterate on Step 4 and revise the overall list of lower level safety constraints.

In order to enforce the GNC safety constraints in Figure 4, the GNC operator needs to maintain control of the spacecraft's *Attitude* and the first and second time derivatives of it (i.e., *Angular Velocity* and *Angular Acceleration*). Additionally, the *Reaction Wheel Velocities* and *Reaction Wheel Accelerations* are a kinetic energy source that—if improperly controlled—could pose a threat to other spacecraft components. Finally, *Workstation Functionality*, *Workstation Cleanliness*, and *Workstation Accessibility*, are also of concern (particularly in regards to GNC-SC6), though it usually inappropriate to intentionally disrupt the control of the latter two state variables in an immersive simulation.

Step 6: Identify the control loops affecting the state variables associated with the lower level safety constraints

Step 6 calls for the identification of the control loops that control the state variables identified in Step 5. For each control loop, the analyst identifies the controller, actuator, and observer, and lists some high-level details about the control inputs, process inputs, control algorithms, and estimation algorithms. These details describe the control loops at a "black box" or block diagram level and do not necessarily delve into the detailed mathematics, coding, and electromechanical design specification used to implement the control loop. The overall goal is to keep the description at a general level, so that the analyst can effectively apply the generic STPA taxonomy to identify inadequate control scenarios in Step 7.

Figure 5 contains a description of one of the GNC control loops used for the LADEE application. This loop controlled the spacecraft attitude and it time derivatives and was automated on board the spacecraft. However, it was closely supervised and guided by the operators on Earth who routinely updated its control inputs and on occasion updated its control algorithm and estimator algorithm. That human supervisory loop included control of the reaction wheel speeds through ground commanded momentum management maneuvers utilizing the reaction control system thrusters. It was also facilitated by other control loops that allowed the operators—including the GNC operators—to maintain basic workstation functionality, cleanliness, and accessibility.

Step 7: Identify the potential causes for inadequate control of the state variables

Once the control loops have been characterized, the analyst then applies the STPA taxonomy to identify scenarios in which the state variables associated with the lower level safety constraints are inadequately controlled. In the STPA taxonomy, there are four generic types of inadequate control actions [33]:

1. A control action required for safety is not provided or followed.
2. An unsafe control action is provided.
3. A potentially safe control action is provided too early or too late (i.e., at the wrong time or in the wrong sequence).
4. A control action required for safety is stopped too soon or applied too long.

GNC-Control Loop 1:

State Variables: Spacecraft Attitude and its time derivatives

Controller: GNC control logic and S/C Fault Management logic

Actuators: 4 Reaction Wheels and 4 Reaction Control Thrusters

Observer: Star Tracker, Inertial Measurement Unit, Reaction Wheel Gyros, Course Sun Sensors, Reaction Wheel Speed Sensors, and Estimator Logic

Control Algorithm: Modal with mode transitions dictated by ground command, stored command sequences, and S/C Fault Management logic

Estimator Algorithm: Kalman Filter based for most modes and sun sensing logic for other modes

Control Inputs: Stored command sequences, solar reference, and ground command

Process Inputs: Electrical power, solar radiation, gravity gradient torques, thruster torques, RF link, thermal energy (from heaters)

Figure 5. Description of a GNC control loop in the LADEE application.

In performing the analysis, it is recommended to list customized versions of these inadequate control action statements for each lower level safety constraint. From there, the analyst then uses the STPA taxonomy of control flaws to identify potential problems that can arise in the control loop and cause an inadequate control action. Figure 6 [34] shows this taxonomy superimposed on the block diagram of a generic electromechanical control loop. In some cases, these control flaws may relate to controllable state variables that were overlooked in prior steps of the analysis and thus it may be necessary to reiterate Steps 4 through 7 to analyze these control loops as well.

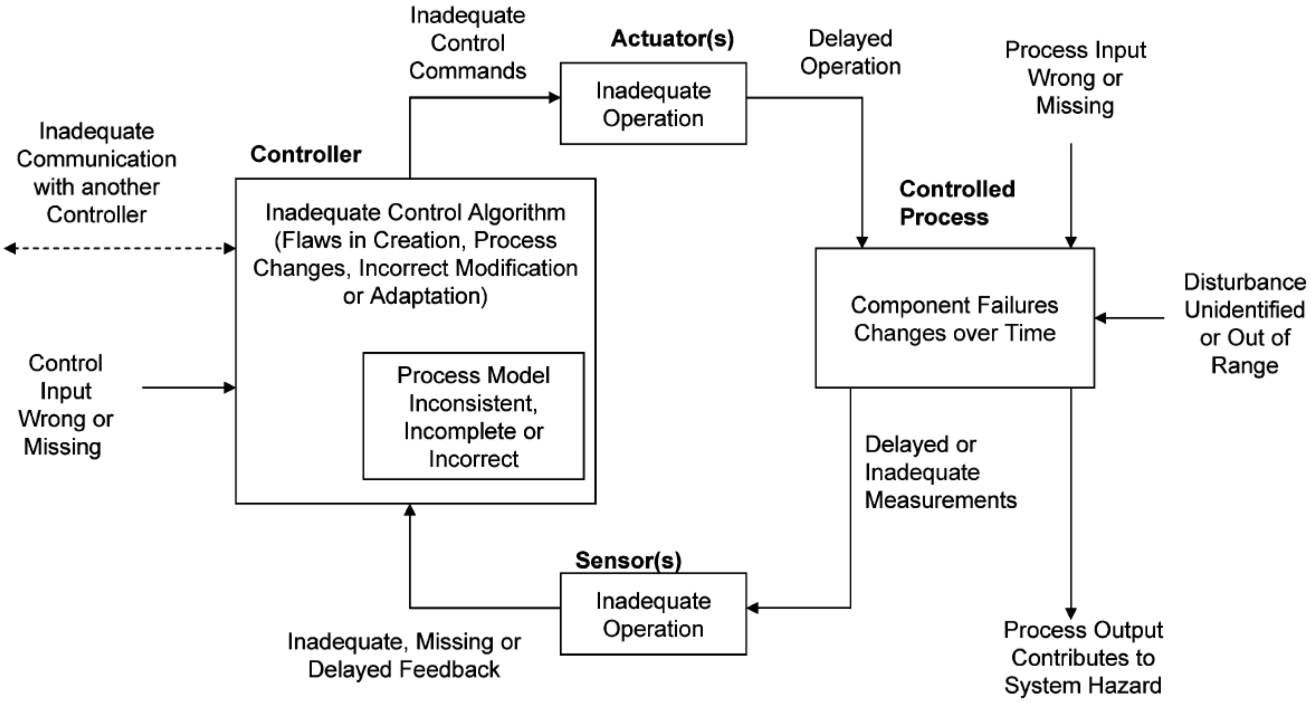

Inadequate Control Commands
Actuator(s)
Inadequate Operation
Delayed Operation
Process Input Wrong or Missing
Inadequate Communication with another Controller
Controller
Inadequate Control Algorithm (Flaws in Creation, Process Changes, Incorrect Modification or Adaptation)
Controlled Process
Component Failures Changes over Time
Disturbance Unidentified or Out of Range
Control Input Wrong or Missing
Process Model Inconsistent, Incomplete or Incorrect
Delayed or Inadequate Measurements
Sensor(s)
Inadequate Operation
Inadequate, Missing or Delayed Feedback
Process Output Contributes to System Hazard

Figure 6. STPA taxonomy superimposed on a generic control loop. [34]

Step 8: Define and execute SimSup's Loop

The inadequate control actions identified in Step 8—and the control flaws that can cause them—provide immersive simulation scenario developers with numerous cases upon which they can base their scenarios. However, it is likely that the analysis will yield far too many cases to simulate over the career of an individual operator—due to restrictions on the amount of time the operator can devote to training and potential limitations of the simulator(s). Thus, the scenario developer completes SimSup's Loop by selecting a set of scenarios—conducted over multiple simulations—that each expose the operator to issues with different aspects of a typical control loop (i.e., control inputs, controllers, actuators, controlled processes, process inputs, disturbances, observers/sensors).

The goal of SimSup's Loop is to provide the trainee with exposure to as complete of a set of generalizable control system problems as possible. Accordingly, the minimum set of scenarios required to complete SimSup's Loop can include cases related to *any* of the control loops in the operator's domain, but not necessarily *all* of them.

5. LADEE SIMULATION SCENARIOS AND INFLIGHT ANOMALIES

The approach described in the previous section was used for simulation scenario development for the LADEE mission. However, due to the limited number of simulations in the LADEE simulation campaigns, it was not practical to complete SimSup's Loop for an individual operator. Instead, the mission operations team as a whole was subjected to a SimSup Loop that targeted almost every part of the generic control loop several times.

In this section, many of the simulation scenarios from this SimSup Loop are grouped by the portion of the relevant control loop that they were intended to affect and are described at a very high level. Additionally, some of the actual in-flight anomalous scenarios that occurred during LADEE are described at a very high level, grouped by the portion of the relevant control loop that they affect.

Control Input Issues

The LADEE simulation scenarios involving control input issues include the following:

1. Shortly before the end of a shift, the SimSup instructed the Command Controller to *accidentally* uplink the command that changes the spacecraft attitude control mode to Safe Mode. At the time, the controller was performing a task that required her to be prepared to send the command intentionally if a problem was to arise. The scenario forced the team to perform a somewhat lengthy recovery process—during odd hours—to restore science operations.

2. During several simulations the SimSup forbid key operators from participating in certain parts of simulations (by telling them that they were absent due to illness or transportation issues). Doing so,

forced the operators to delay their control inputs or have someone else create the inputs.

3. During a few simulations, the SimSup introduced various workstation issues that hindered the operator's ability provide their control inputs to the processes that they were controlling.

The control input issues encountered during the LADEE mission include:

4. During the extended mission phase, one of the science instrument teams requested an increase to the duty cycle of their instrument. This increase resulted in several instances in which the instrument exceeded its thermal limits and shut itself off. The instrument was not damaged and the instrument team accepted the risk of the instrument shutting itself off, but the incidents triggered anomaly response actions that had to be waived off.

5. On a few occasions, the files that the operations team used to instruct the ground stations were improperly overwritten (i.e., the new file omitted the last contact from the old file). As a result, the ground station missed the start of these contacts, leading to a lost opportunity to collect tracking data and the triggering of the anomaly response process.

6. The voice communications lines between the LADEE Mission Operations Center (MOC) and the Deep Space Network (DSN) occasionally underperformed, making it difficult for the LADEE operators to provide real-time instructions to the DSN operators. These issues were similar to some of the issues described in item 3 above.

Controller Issues

The LADEE simulation scenarios involving issues with the controller of a control loop include the following:

7. For a launch and spacecraft activation simulation, the SimSup introduced a sign error in the control algorithm for the reaction wheels—the sign axis for one of the wheels was mathematically inverted in the algorithm. When the reaction wheels initialized after the simulated launch, the affected wheel spun up to its top speed and forced the other reaction wheels to spin up to their top speeds. With this issue, the spacecraft could not control its attitude without operator intervention.

8. For a lunar orbit insertion simulation, the SimSup attempted to mimic the effect of an inadequate thermal environment model by significantly increasing the lunar albedo and radiation in the simulator. During the lunar orbit insertion and each subsequent periselene in the simulation, spacecraft temperatures increased more than the operators expected that they would.

9. When spacecraft maneuvers were simulated, the SimSup would often introduce varying levels maneuver execution errors (i.e., underperformance or overperformance relative to the operator predictions). These errors would force operators to update their maneuver performance models and potentially re-plan events following the maneuver.

The controller issues encountered during the LADEE mission include the following:

10. On launch day, the reaction wheel control algorithm treated a nominal reaction wheel behavior as an anomalous behavior and turned all four of the spacecraft's reaction wheels off shortly after they were initialized. With this issue—as was also the case with scenario 7 above—the spacecraft could not control its attitude without operator intervention.

11. After lunar orbit insertion, the temperature of some spacecraft components exceeded operator expectations and flight rule limits, thus triggering operator intervention. The operator expectations were based on pre-launch analyses that did not account for all of the operating attitudes and transmitter duty cycles for the temporary orbit configuration between the first and second lunar orbit adjustment maneuvers.

12. Every maneuver had a certain level of maneuver execution error. [18] None of these errors were severe and in general, the error levels decreased over the duration of the mission as the operators refined their maneuver performance models.

13. The spacecraft's flight software issued an improper function call during a laser communications session that caused LADEE's flight computer to reboot. The reboot halted the laser communications session and placed the spacecraft in Safe Mode until the mission operations team could assess the spacecraft and command a return to normal Fine Pointing Mode operation. The software defect was identified and corrected in a flight software update performed during LADEE's extended mission.

Actuator Issues

The LADEE simulation scenarios covering issues with the actuators of relevant control loops included the following:

14. During a science operations simulation, the SimSup injected a short in the circuitry of a reaction wheel, causing that reaction wheel to power off. The loss of the reaction wheel significantly affected the control authority of the

attitude control system and sent the spacecraft into Safe Mode.

15. The SimSup started one simulation with the spacecraft in Safe Mode due to an inadvertent reaction control system thruster firing that occurred out of view. Though the firing did not cause permanent damage to the thruster, it triggered automatic firings of the other thruster and ultimately sent the spacecraft to Safe Mode.

16. During a launch and activation simulation, the SimSup initialized one of the reaction wheel with a significantly higher amount of friction than the other reaction wheels. The issue did not significantly affect the control authority of the reaction wheel, but raised questions about how to manage that wheel to keep it from degrading further.

17. During a lunar orbit insertion simulation, the SimSup introduced a temporary overcurrent condition in the unit that actuated the thruster valves. The condition powered off the unit several minutes prior to the maneuver and had to be corrected in order for the maneuver to occur.

Unlike many other spaceflight missions, the LADEE mission had no significant issues with the actuators in the control loops of interest to the operators. This result is likely due to the relatively short duration of the mission, the design of the spacecraft and instruments, preflight efforts to flight certify the mission hardware and performance of the launch vehicle.

Controlled Process Input Issues

Due to the numerous subsystem interdependencies on a spacecraft, it is often the case that an issue with one subsystem's process outputs will affect the inputs to another subsystem's processes. For example, whenever the GNC subsystem was unable to control the attitude, the Thermal, Power, and Communications subsystems were deprived of the pointing inputs that were needed for thermal control, power generation, and closing the communication link with Earth, respectively.

The LADEE simulation scenarios covering issues with the inputs of the controlled processes of relevant control loops also included the following:

18. The SimSup provided simulated tracking data files to the orbit determination operators during some of the simulations, some of which contained corrupted data. [21] These data corruptions simulated issues (e.g., misconfigurations and other equipment problems) with the ground stations collecting the data and had to be identified and removed from the orbit determination data set (if they could not be corrected).

19. During a launch and activation simulation, the launch vehicle imparted off nominal attitude rates on the spacecraft when it deployed it. The rates sent the spacecraft into Rate Reduction Mode, which uses the thrusters to reduce the attitude rates. They also affected the spacecraft's communication link with Earth as the spacecraft rotated its antennas away from Earth.

20. For several launch and activation simulations, the SimSup initialized the simulator with a state vector that simulated off nominal launch vehicle performance. These off nominal orbit insertions led to activity and maneuver replanning as well as problems with the spacecraft's communication link with Earth.

21. During several simulations, the SimSup degraded the performance of DSN, NEN, and SN communications assets or made them unavailable during critical times. The temporary issues with these assets degraded/removed an input to controlled processes that allowed for ground commanding and information

The processes controlled by LADEE operators and the lower level control loops that they supervised were occasionally hindered with input issues, included the following:

22. A Lunar Eclipse occurred during LADEE's extended mission, depriving the spacecraft of a solar radiation input for several hours. The eclipse was predicted prior to launch and served as a launch constraint (LADEE had to launch early enough to complete its primary mission before the eclipse). The spacecraft was not designed to survive the eclipse, but the operators utilized their control authority over that spacecraft orbit, payload activity, and spacecraft mode to allow it to survive the eclipse.

23. The ground stations providing tracking data occasionally provided files with corrupted tracking data. These instances forced the orbit determination team to identify and remove the data and work with to ground stations to identify and resolve the source of the data issues.

24. Throughout the mission, communications assets were occasionally unavailable or improperly configured to provide telemetry at the expected times—similar to the scenarios described in item 21. None of this incidents led to significant problems for the operations team.

Disturbances to the Controlled Process

The LADEE simulation scenarios involving disturbances to the controlled processes of relevant control loops includes the following:

25. During a lunar orbit insertion simulation, the SimSup injected a radiation related upset event that rebooted the spacecraft. The timing of this issue necessitated a rapid operator response to manually start the maneuver.

26. Throughout the simulation campaign, the SimSup regularly injected radiation related memory errors (both single-bit and multiple-bit) into the simulations. The single-bit errors were automatically corrected by the memory scrub algorithm, but the multiple-bit errors each had to be evaluated for their criticality and addressed accordingly.

The processes controlled by LADEE operators and the lower level control loops that they supervised were disturbed in several ways, including the following:

27. Throughout the mission, a number of radiation related memory errors (both single-bit and multiple-bit) occurred in various areas of the spacecraft's memory. [36] The single-bit errors were automatically corrected and each of the multiple-bit errors were investigated (none were in critical areas of memory).

28. LADEE's low altitude equatorial orbit created opportunities for conjunction with the Lunar Reconnaissance Orbiter's (LRO's) low polar orbit. A series of such conjunction opportunities posed a threat to both spacecraft and prompted LADEE to slightly modify its maneuver plan to increase miss distances between the two spacecraft. Interestingly, the SimSup planned this scenario for a simulation, but aborted it due to an issue that arose during that particular simulation.

Controlled Process Issues

In the absence of disturbances and issues with the inputs to the controlled process, there is still the potential for issues to arise in the controlled process.

The LADEE simulation scenarios involving issues with the controlled process include the following:

29. During a Science Phase simulation, the SimSup introduced a drift in the transponder oscillator output. [21] This issue resulted in varying delays in the ranging data, thus reducing the quality of the orbit solution.

30. The SimSup initialized a couple simulations with damage to some of the spacecraft's solar cells. Interestingly, the power system had excess power generation capacity for the mission phases being simulated and thus, the power control algorithm mostly kept the switches to these cells open. Because the switches the cells were only closed briefly during the simulation, this issue was very difficult for operators to detect this damage. Accordingly, the power system operators developed new analysis tools to diagnose damage to solar arrays.

The controlled process issues that occurred during the LADEE mission included the following:

31. Communications multipathing reduced the useful duration of communications sessions by interfering with command uplink capabilities. When the Earth was low on the lunar horizon, as seen from LADEE's point of view, radio signals reflected off the lunar surface mixed with signals received directly from Earth. This mixing compromised the data content of the uplinked command and resulted in temporary loss of command capability. As LADEE's orbit lowered, the intensity and duration of these effects increased. The team compensated by scheduling ground commanding at times in the orbit where multipathing was less of a concern.

Observer/Sensor Issues

The LADEE simulation scenarios covering issues with the observers—including sensors and estimator algorithms—of relevant control loops included the following:

32. During a lunar orbit insertion simulation, the SimSup caused the star tracker to incorrectly sense overcurrent conditions and power itself off twice before the maneuver. The timing of this issue required a relatively quick operator response to get ultimately get the spacecraft into the proper maneuver attitude.

33. During several simulations, the SimSup introduced various temperature sensor failures—some intermittent and others permanent. Several of these sensors fed state information to the control algorithm of heaters and thus required changes to the heater control inputs or control algorithms.

34. For a launch and spacecraft activation simulation, the SimSup introduced a proportional bias in the readings of the speed sensor for one of the reaction wheels. This issue affected the ability of the GNC operator to estimate and control the spacecraft's momentum state.

35. During a lunar orbit insertion simulation the SimSup injected a temporary overcurrent condition that powered off the Inertial Measurement Unit prior to the maneuver. The issue affected the ability of the GNC system to control the attitude of the spacecraft during the maneuver.

36. During one simulation the SimSup altered the reading from the antenna switch sensor to always indicate that the switch was in the Medium Gain Antenna (MGA) configuration. The impact of this

issue once it was diagnosed was a slight degradation in operator situation awareness. The indication was not used in any automated control loops and other indications were available to determine the actual antenna switch position.

37. The SimSup initiated one simulation with an uncharacterized transponder delay. [21] Upon diagnosing this issue the orbit determination team updated the orbit estimation algorithm.

The observer issues encountered during the LADEE mission include the following:

38. Performance issues in LADEE's star tracker caused it to output erroneous or stale attitude data to the flight software. The state estimator software interpreted these outputs as sudden jumps in spacecraft attitude, violating maximum attitude and attitude rate limits and forcing the spacecraft to enter Safe Mode. The mission operations team diagnosed this phenomenon and eventually uploaded flight software modifications to make the state estimator more robust in handling erroneous star tracker data.

39. Output from two propulsion system pressure transducers was lost in the last few days of LADEE's extended mission. Measurements from these sensors were normally used in ground software to predict thrust performance prior to firings for both orbit maintenance maneuvers and momentum dump events. Although alternative pressure measurements were available, the ground analysis tools had to be updated in order to use these alternative measurements.

Summary of Control Loop Issue Locations

The control loop locations of all of the issues described above are depicted in Figure 7. In this figure, the numbers associated with the issues mentioned above are placed over a generic control loop in the location in which they arose. As shown in the figure, issues—both simulated and real—arose from nearly every generic control loop location.

6. DISCUSSION

The use of SimSup's Loop to train the mission operations team for the LADEE mission demonstrates the kind of results that can be obtained from using a hazard analysis based approach for simulation scenario development in conjunction with a Procedure Model for simulation scenario development. In this section, key lessons from the LADEE experience are discussed.

The importance of disrupting each part of the control loop

The examples provided in the previous section highlight how issues can arise in any portion of a control loop of interest to an operator in a spaceflight mission. Additionally, these examples qualitatively demonstrate that the signature of, impact of, and recovery options for an issue often relate to the portion of the control loop in which it arises. Table 1 lists such traits—by the relevant control loop location—that were demonstrated during LADEE experience. These results support the claim that a well-rounded spacecraft operator training program should instill an appreciation of issues originating from every portion of a typical control loop (presumably through scenarios that provide the operator with experience dealing with such issues).

Table 1. Traits associated with issues arising in different parts of the generic control loop.

ISSUE LOCATION	ISSUE TRAITS
Control Input	• The health of each control loop component may appear to be fine • Full recovery is sometimes possible with a correction to the control input
Controller	• The health of each control loop component may appear to be fine • Full recovery is sometimes possible with a correction to the control algorithm
Actuator	• The health and status of individual actuators is adversely affected • Degraded control authority is a likely outcome
Controlled Process Input	• The health and status of multiple control loop components may be adversely affected • Application of control authority may mask the issue • Full recovery is sometimes possible with the restoration of the process input
Controlled Process (Internal)	• The health and status of multiple control loop components may be adversely affected • Application of control authority may mask the issue • Degraded control authority is a likely outcome
Controlled Process (Disturbance)	• The health and status of multiple control loop components may be adversely affected • Application of control authority may mask the issue • Full recovery is sometimes possible if the disturbance can be endured/rejected
Observer (Sensor)	• The health and status of individual sensors is adversely affected • Degraded control authority or loss/corruption of control loop data are likely outcomes
Observer (Estimation Logic)	• The health of each control loop component may appear to be fine • Full recovery is sometimes possible with a correction to the estimator logic

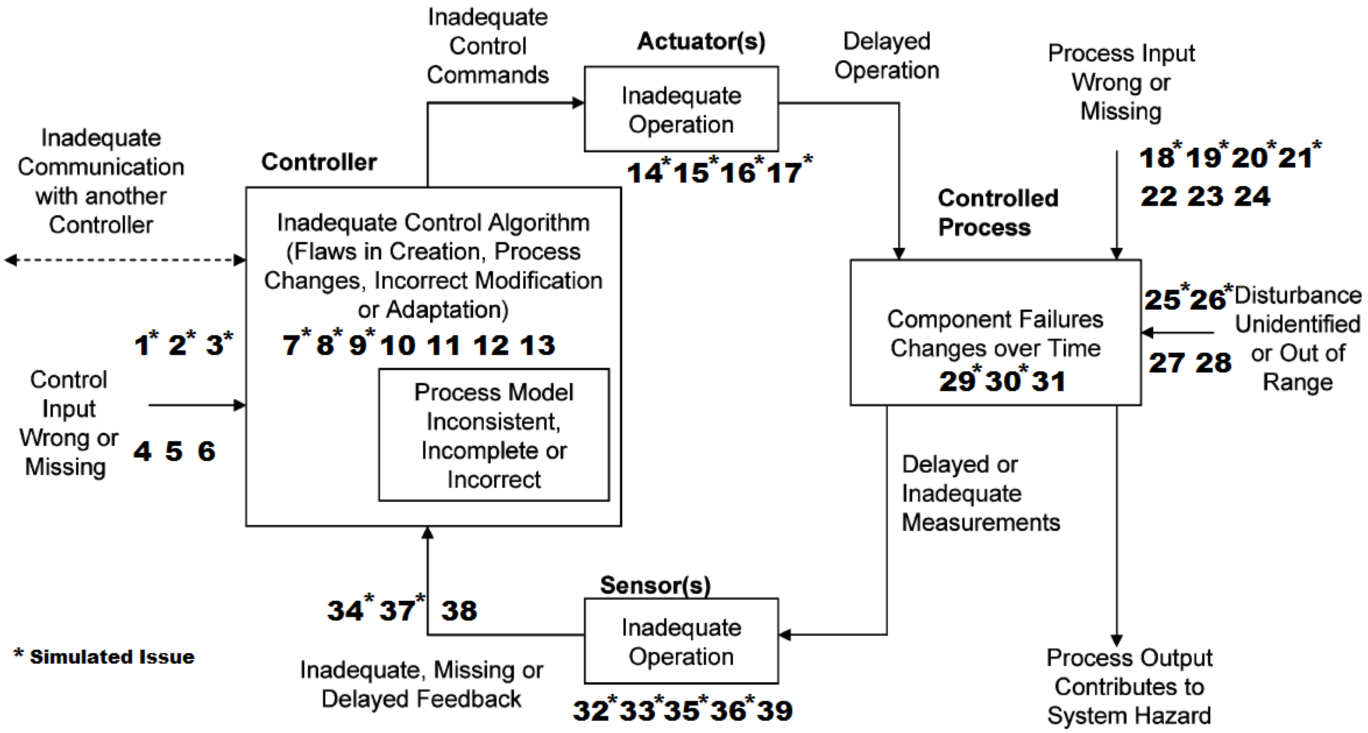

Inadequate Control Commands
Actuator(s)
Inadequate Operation
14* 15* 16* 17*
Delayed Operation
Process Input Wrong or Missing
18* 19* 20* 21*
22 23 24
Inadequate Communication with another Controller
Controller
Inadequate Control Algorithm (Flaws in Creation, Process Changes, Incorrect Modification or Adaptation)
7* 8* 9* 10 11 12 13
Process Model Inconsistent, Incomplete or Incorrect
1* 2* 3*
Control Input Wrong or Missing
4 5 6
Controlled Process
Component Failures Changes over Time
29* 30* 31
25* 26* Disturbance Unidentified or Out of Range
27 28
Delayed or Inadequate Measurements
Sensor(s)
Inadequate Operation
32* 33* 35* 36* 39
34* 37* 38
Inadequate, Missing or Delayed Feedback
Process Output Contributes to System Hazard
* Simulated Issue

Figure 7. The generic control loop locations of LADEE issues described in Section 5 (adapted from [34]).

Disruption of each part of the control loop could potentially be achieved through the use of the many hazard analysis techniques employed in spacecraft development and operations. Indeed, spacecraft simulation scenario developers often use a number of formal and informal hazard analysis approaches to complement the Procedure Model. However, without an explicit emphasis on disrupting each part of the control loop, such achievements would be serendipitous and presumably accomplished over a longer period of time due to redundancies in scenario design.

Similarly, disruption of each part of the control loop could be achieved through the use of the Procedure Model, but only if the procedures are comprehensive enough to cover contingencies in each part of the control loop and there are sufficient opportunities to exercise all of the appropriate legs of the procedure tree. Such levels of procedural completeness have arguably been obtained in some historical, high-cost human spaceflight programs where hundreds of procedures—many with their own subbranches—have been written and operators have spent hundreds of hours practicing as much of the procedure tree as possible. However, lower cost missions cannot afford this level of completeness in procedure development and simulation.

Moreover, the use of an exploratory approach to simulation scenario development could drive further development of the procedures. By trying to identify ways to disrupt a specific portion of the control loop without taking existing procedures into account, the scenario development could identify cases that warrant the writing of a new procedure. A total of 29 contingency procedures were developed in preparation for LADEE operations; these procedures focused on the identification of and response to sensor and actuator anomalies. In general, these procedures did not address other portions of the control loop. A Procedure Model approach to LADEE training would have only addressed sensor and actuator issues.

In flight, LADEE experienced no actuator anomalies and few sensor anomalies. As cited above, many LADEE anomalies occurred in other portions of the control loop. Effective response to these anomalies required a skilled mission operations team that could quickly and accurately develop an understanding of unanticipated conditions. For example, the reaction wheel shutdown anomaly encountered during spacecraft activation (i.e., issue 10 in Section 5) presented a controller issue that had not been previously identified nor documented. Through use of the SimSup Loop method (particularly in the development of simulation scenario described as issue 7 in Section 5), the mission operations team had the opportunity to exercise the skills necessary for rapid identification of and response to such controller issues. As a result, the mission operations team was able to isolate and resolve the issue in a matter of hours, avoiding potential loss of mission on the first day of flight.

Other more complex anomalies in controlled processes, such as the communications multipathing issues, required more time to assess and resolve, but the team's ability to address these issues was potentially enhanced by experience gained in simulations with other process anomalies.

Testing versus Training

The scenarios mentioned in the previous section comprise a portion of the total flight-like exercises that the LADEE mission operations team partook in to prepare for the mission. Many additional exercises took place with an emphasis on executing a planned activity and the expectation that no one would intentionally perturb the planned course of action for that activity.

These exercises trained the operators how to execute operational procedures in an operational environment—which is an essential skill for spacecraft operators. However, the goal of these exercises was often to verify that an operational product or process would work as planned. In other words, the exercises were more geared towards testing aspects of the operational system than training operators.

Thus, one may choose to draw an explicit distinction between testing and training and apply the different models for simulation scenario development to each. The Procedure Model would be associated with the testing exercises and SimSup's Loop (or another hazard analysis based approach) would be associated with the training exercises. Training and testing of the operational systems would be accomplished in both types of exercises, but the distinction would focus the preparation efforts and timing for these exercises.

The Importance of "Freebies"

Freebies are unplanned anomalies that occur during a simulation due to operator errors, oversights in the setup or operation of the simulator, ineffective operations procedures, equipment issues, etc. When presented with a freebie, the simulation supervisor must choose to make a declaration that the freebie is to be disregarded, restart the simulation, or allow the simulation participants to treat the freebie as if it were a real anomaly.

Each of these options represents a departure from the original plan for the simulation or the realism of the simulation, and thus freebies can easily undermine the objectives of a simulation. However, if a freebie is properly managed, it can present several opportunities—hence the use of the term freebie, which implies getting something of value for free. First, freebies often have the same cause or effect as anomalies that can occur during actual operations and therefore, the trainee (or simulation supervisor) might obtain valuable experience working through the freebie. Additionally, freebies add a level of uncertainty to each simulation that can reduce the temptation to intentionally or unintentionally *game* the simulation. All simulations occur in a context that may give the trainee hints as to what scenario developer is planning for the simulation and that in turn could affect how the trainee prepares for the simulation—unless he or she is deterred by the prospect of a freebie occurring.

The opportunities presented by freebies can potentially complement an overall training program based on both the Procedure Model and SimSup's Loop. Therefore, while it is important to keep freebies from becoming a distraction, they should not necessarily be dismissed in favor of the original simulation objectives when they occur.

A number of freebies were encountered during LADEE simulations. The SimSup attempted to opportunistically leverage a portion of these freebies with varying degrees of success. In the interest of brevity, freebies were not discussed in the previous section.

7. CONCLUSIONS

SimSup's Loop is a control theory approach to developing simulation scenarios that train spacecraft operators to maintain control over two major aspects of their roles: safety and security.

SimSup's Loop was used in conjunction with the opportunistic treatment of freebies and the Procedure Model for simulation scenario development to develop the training program for LADEE spacecraft operators. The trainees were exposed to issues arising in every portion of the generic control loop and subsequently encountered issues arising from all but one generic portion of the control loop in flight. The similarities between specific simulation scenarios and actual issues encountered in flight were instrumental in helping the operators craft solutions to these issues. Moreover, the one generic part of the control loop that did not generate troublesome issues (i.e., actuators) was given an apparently disproportionate level of attention in the writing of LADEE's contingency procedures. Consequently, the training provided in issues arising from other parts of the control loop provided operators with an experiential foundation to build upon in addressing the issues where the procedural foundation was lacking.

8. FUTURE WORK

Though SimSup's Loop was successfully applied to LADEE spacecraft operator training, questions remain as to how it could be applied to other missions. Included among these questions are the following:

- How does this approach to simulation scenario development compare with approaches based on other hazard analysis techniques? Do they lead to scenarios that cover the entire control loop?

- Should the operator be taught the theory behind the development of the simulation scenarios (i.e., can the theory enhance the operator's mental model of his or her responsibilities or could it be a distraction)? For the LADEE mission, the SimSup discussed—in passing—how he was disrupting the control loop during his simulation debriefs, but he did not require trainees to explicitly demonstrate an understanding of the theory.

- To what extent should the information derived from developing and executing a SimSup Loop feedback into procedure development? Should specific procedures be prepared for handling issues arising in each part of the control loop and if so, should these procedures leverage the SimSup's Loop terminology?

- What is the appropriate balance of scenarios developed via SimSup's Loop and scenarios developed via the Procedure Model when only a small number of scenarios can be run?

- To what extent should SimSup's Loop be applied to security issues? For the LADEE mission in particular, the security aspects of spaceflight operations were not emphasized in order to focus on other priorities. However, other missions may have more resources (or specific concerns) to warrant an emphasis on security issues.

- How can insights from the vast literature on learning and training be applied to analyze the effectiveness of SimSup's Loop or derive answers to the questions above?

An analysis of past missions can provide insight into some of these questions while future missions will provide opportunities to exercise different responses to these questions. However, the relatively low frequency with which spacecraft operator training programs are executed, and unique demands of each spaceflight mission will likely limit the ability to statistically demonstrate the efficacy of such responses. Therefore, engineering judgment will be required to evaluate these responses and determine how future spacecraft operators will be trained.

ACKNOWLEDGMENTS

The authors would like to thank all of the members of the LADEE mission operations and science operations teams. Among them, Peter Berg, Nathaniel Benz, Michael Logan, Scott Christa, John Costa, and Daniel Judnick helped set up and operate the simulators for the simulations discussed in this paper. Additionally, the following individuals played a significant role in planning the simulations: David "Rusty" Hunt, Matthew D'Ortenzio, David Lung, Laura Plice, Sarah Hobart, John Bresina, Howard Cannon, Alisa Hawkins, Lisa Policastri, Richard Elphic, Tina Kelly, and Jennifer Sager Cosgrove.

LADEE simulators were developed by Danillo Viazzo, Craig Pires, Fritz Renema, Karen Gundy-Burlet, Eleanor Crane, Pat Castle, Jesse Fusco, Douglas Forman, and some of the individuals who were already mentioned above.

James Woodburn and Jens Ramrath from Analytical Graphics Incorporated created simulated tracking data for many of the simulations discussed in this paper.

In order to independently test the spacecraft Fault Management System, the lead author attempted to develop simulation scenarios independent of the work done by the Fault Management Team. Nonetheless, the work of the team—led by Anupa Bajwa—was so comprehensive and influential to the mission operations concept that it undoubtedly influenced the simulation scenario development as well.

The authors learned much about immersive simulation over their careers from the Spaceflight Training Division at NASA JSC and individuals elsewhere (namely Tom Regiec at Orbital Sciences, Victor Gehr at General Dynamics C4 Systems, Kathryn Schimmels at NASA JPL, David Gaba at Stanford, and Brett Stroozas).

Additionally, the authors extend thanks to the peer reviewers of this paper. Their comments improved the quality of this publication.

This work and the LADEE mission was performed at the NASA Ames Research Center, under the Lunar Quest Program, and sponsored by the NASA Science Mission Directorate Division of Planetary Science.

REFERENCES

[1] M. Bronzini, S. Bruno, M. De Benedictis, S. Lamonaca, M. La Scala, G. Rotondo, and U. Stecchi, "Operator Training Simulator for Power Systems: training evaluation methodologies based on fuzzy logic," 2010 IEEE International Symposium on Industrial Electronics Proceedings, July 4-7, 2010.

[2] A. L. Ahmad, E. M. Low, S. R. Abd Shukor, "Safety Improvement and Operational Enhancement via Dynamic Process Simulator: A Review," Chemical Product and Process Modeling 5, August 2010.

[3] David M. Gaba, "The future vision of simulation in health care," Quality and Safety in Health Care 13, i2-i10, October 2004.

[4] R. Michael Mullane, Riding Rockets: The Outrageous Tales of a Space Shuttle Astronaut, New York, NY: Simon and Schuster, 2006.

[5] Thomas D. Jones, Skywalking: An Astronaut's Memoir, New York, NY: HarperCollins Publishers, 2006.

[6] Christopher C. Kraft, Flight: My Life in Mission Control, New York, NY: Penguin Putnum Inc., 2001.

[7] Eugene F. Kranz, Failure is not an Option: Mission Control from Mercury to Apollo 13 and Beyond, New York, NY: Berkeley Books, 2000.

[8] Sy Liebergot and David M. Harland, Apollo EECOM: Journey of a lifetime, 2nd Edition, Burlington, Ontario, Canada: Apogee Books, 2006.

[9] Donald K. Slayton, "Crew Functions and Training," 5th AIAA Annual Meeting and Technical Display Proceedings, October 21-24, 1968.

[10] Frank E. Hughes, "Simulation for STS Flight Crew Training," 20th AIAA Aerospace Sciences Meeting Proceedings, January 11-14, 1982.

[11] David J. Forrest, J. Benton Christman, Mark Wiederholt, and Reynaldo R. Rodriguez, "International Space Station Part Task Trainers," 1997 AIAA Modeling and Simulation Technologies Conference Proceedings, August 11-13, 1997.

[12] Gary Dittemore and Christie Bertels, "The Final Count Down: A Review of Three Decades of Flight Controller Training Methods for Space Shuttle Mission Operations," AIAA SPACE 2011 Conference and Exposition Proceedings, September 27-29, 2011.

[13] Butler P. Hine, Stevan Spremo, Mark Turner, and Robert Caffrey, "The Lunar Atmosphere and Dust Environment Explorer Mission," 2010 IEEE Aerospace Conference Proceedings, March 6-13, 2010.

[14] Matthew V. D'Ortenzio, John L. Bresina, Alan R. Crocker, Richard C. Elphic, Ken F. Galal, David R. Hunt, Brandon D. Owens, Alisa M. Hawkins, Laura Plice, and Lisa A. Policastri, "Operating LADEE: Mission Architecture, Challenges, Anomalies, and Successes," 2015 IEEE Aerospace Conference Proceedings, March 7-14, 2015.

[15] Arlen Kam, Laura Plice, Ken F. Galal, Alisa M. Hawkins, Lisa A. Policastri, Michel E. Loucks, John P. Carrico, Craig A. Nickel, Ryan L. Lebois, and Ryan Sherman, "LADEE Flight Dynamics: Overview of Mission Design and Operations," 25th AAS/AIAA Space Flight Mechanics Meeting Proceedings, January 11-15, 2015.

[16] Nathaniel A. Benz, Danilo Viazzo, and Karen Gundy-Burlet, "Multi-Purpose Spacecraft Simulator for the LADEE Mission," 2015 IEEE Aerospace Conference Proceedings, March 7-14, 2015.

[17] Lisa A. Policastri, John P. Carrico, Arlen Kam, Craig A. Nickel, Ryan L. Lebois, and Ryan Sherman, "Orbit Determination and Acquisition for LADEE and LLCD Mission Operations," 25th AAS/AIAA Space Flight Mechanics Meeting Proceedings, January 11-15, 2015.

[18] Alisa M. Hawkins, Arlen Kam, and John P. Carrico, "LADEE Maneuver Planning and Performance," 25th AAS/AIAA Space Flight Mechanics Meeting Proceedings, January 11-15, 2015.

[19] Bryan S. Robinson, Don M. Boroson, Dennis A. Burianek, Daniel V. Murphy, Farzana I. Khatri, Jamie W. Burnside, Jan E. Kansky, Abhijit Biswas, Zoran Sodnik, and Donald M. Cornwell, "The NASA Lunar Laser Communication Demonstration—Successful High-Rate Laser Communications To and From the Moon," 2014 SpaceOps Conference Proceedings, May 5-9, 2014.

[20] Howard N. Cannon, Anupa Bajwa, Peter P. Berg, and Alan R. Crocker, "LADEE Preparations for Contingency Operations for the Lunar Orbit Insertion Maneuver," 2015 IEEE Aerospace Conference Proceedings, March 7-14, 2015.

[21] James Woodburn, Lisa A. Policastri, and Brandon D. Owens, "Generation of Simulated Tracking Data for LADEE Operational Readiness Testing," 25th AAS/AIAA Space Flight Mechanics Meeting Proceedings, January 11-15, 2015.

[22] Graham C. Goodwin, Stefan F. Graebe, and Mario. E. Salgado, Control System Design, Upper Saddle River NJ: Prentice-Hall, 2001.

[23] Katsuhiko Ogata, Modern Control Engineering, 3rd Edition, Upper Saddle River, NJ: Prentice-Hall, 1997.

[24] Brandon D. Owens, Using phase space attractors to evaluate system safety constraint enforcement: Case study in space shuttle mission control procedure rework, Ph.D. Thesis, Engineering Systems, Massachusetts Institute of Technology, 2009.

[25] Gene F. Franklin, J. David Powell, and Abbas Emami-Naeini, Feedback Control of Dynamic Systems, 4th Edition, Upper Saddle River, NJ: Prentice-Hall, 2002.

[26] Ronald A. Hess, Pilot Control, in Pamela S. Tsang and Michael A. Vidulich (eds.), Principles and Practice of Aviation Psychology, Mahwah, NJ: Lawrence Erlbaum Associates Publishers, 2003.

[27] Nancy G. Leveson, Safeware: System Safety and Computers, Reading, MA: Addison-Wesley, 1995.

[28] National Aeronautics and Space Administration, NASA General Safety Program Requirements, Washington, DC: NPR 8715.3C, 2008.

[29] National Aeronautics and Space Administration, NASA System Safety Handbook, Volume 1: System Safety Framework and Concepts for Implementation, Washington, DC: NASA/SP-2010-580, 2011.

[30] U.S. Department of Defense, Standard Practice for System Safety, Washington, DC: MIL-STD-882D, 2000.

[31] William Young and Nancy G. Leveson, "An Integrated Approach to Safety and Security Based on Systems Theory," Communications of the ACM 57, 31-35, February 2014.

[32] Nancy G. Leveson, “A New Accident Model for Engineering Safer Systems,” Safety Science 42, 237-270, April 2004.

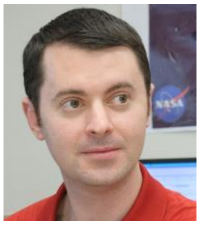
[33] Nancy G. Leveson, Engineering a Safer World: Systems Thinking Applied to Safety, Cambridge, MA: The MIT Press, 2011.

[34] Margaret Stringfellow, Nancy G. Leveson, and Brandon D. Owens, “Safety-Driven Design for Software-Intensive Aerospace and Automotive Systems,” Proceedings of the IEEE 98, 515-525, April 2010.

[35] Brandon D. Owens, Margaret Herring, Nicolas Dulac, Nancy G. Leveson, Michel D. Ingham, and Kathryn A. Weiss, “Application of a Safety-Driven Design Methodology to an Outer Planet Exploration Mission,” 2008 IEEE Aerospace Conference Proceedings, March 1-8, 2008.

[36] Gregory L. Limes, Scott E. Christa, Craig R. Pires, and Karen Gundy-Burlet, “EDAC Events during the LADEE Mission,” 2015 IEEE Aerospace Conference Proceedings, March 7-14, 2015.

BIOGRAPHIES



***Brandon D. Owens** is a Senior Research Engineer for Stinger Ghaffarian Technologies, Inc at the NASA Ames Research Center. He has led the mission operations test campaigns (serving as “SimSup”) for two NASA missions (i.e., LADEE and NuSTAR) and served as the Deputy Mission Operations Manager for LADEE. He also supported mission operations for other NASA missions and programs (i.e., THEMIS/ARTEMIS, Gravity Probe B, the Space Shuttle, and the International Space Station) in various roles that he previously held at the Space Sciences Laboratory at UC Berkeley, the W.W. Hansen Experimental Physics Laboratory at Stanford University, and NASA Johnson Space Center. He has a Ph.D. in Engineering Systems from MIT, an M.S. in Aeronautics and Astronautics from Stanford University, and a B.S. in Aeronautical and Astronautical Engineering from Purdue University.*

***Alan R. Crocker** serves in the Chief Engineer’s Office at NASA Ames Research Center. Previously, he served in several roles at the NASA Johnson Space Center. He has supported human and robotic space exploration projects as a systems engineer, flight controller, instructor, team lead, manager and instructor. For the LADEE mission, he formed and led the mission operations spacecraft engineering team. He received a B.S. in Aeronautical and Astronautical Engineering from Purdue University and is a graduate of NASA’s Systems Engineering Leadership Development Program.*