SCENAREO – Science and Engineering Activity Reasoning and Optimization[1,2]

Shin-Ywan Wang
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove, Pasadena, CA 90272
818-354-0004
shin-ywan.wang@jpl.nasa.gov

Abstract—SCENAREO is a PC-based multi-mission scenario design tool for space missions based on inputs of subsystem performance models, information on various targets, and the specifications for the science objectives. This tool provides model-based analysis to compose and schedule subsystems' operations automatically and optimally to meet the needs of the science objectives. Moreover, SCENAREO has analytical capabilities to validate the feasibility of the science objectives based on subsystem design parameters and to recommend the valid setting in terms of subsystem performance and resources usages to fulfill the proposed science objectives. This tool's automation, optimization, and accuracy capabilities reduce the mission design lifecycle.

TABLE OF CONTENTS

1. INTRODUCTION 1
2. SOFTWARE LAYOUT 2
3. ANALYSIS MODULES 6
4. SEQUENCE/COMMAND GENERATOR.... 10
5. CONCLUSION 10
REFERENCES 10
ACKNOWLEDGEMENTSERROR! BOOKMARK NOT DEFINED.

1. INTRODUCTION

SCENAREO (Science and Engineering Activity Reasoning and Optimization) is one PC application in the Virtual Mission Operation Framework (VMOF) [1] developed by the Mission Simulation and Instrument Modeling (MSIM) group at JPL. VMOF streamlines the mission operations activity analysis, design, simulation, and monitoring tasks. VMOF is composed of Model Object Builder (MOB), SCENAREO (described below), and Time-based Spacecraft Operation Simulator (TSOS) [2]. MOB generates parametric subsystem model scripts, which contain missions and subsystems' design parameters and their values. SCENAREO designs the mission activity based on the output of MOB. TSOS simulates the execution of spacecraft activities created by SCENAREO.

SCENAREO plays the role of designing and analyzing mission scenarios in VMOF. It provides the capability to interpret the literally stated science-return objectives into mission activity scripts described in some low-level format. Science-return means the science data products planned to be acquired during a deep space mission.

For a mission designer, optimizing the mission system design requires compromise among system performance, resource usage, and subsystem operations—a challenging task. Subsystem operations involve the issue of the right commands at the proper time. The calculation of the time number and all relevant command parameters is determined by environmental conditions, subsystem performance capabilities, operation conditions and constraints, resource availability, and correlation among all subsystems. Without an automation design tool, such as SCENAREO, the mission designer may need to manually configure the operational commands offline. That is not efficient. The system design and validation ability is very limited. It can further affect the accuracy of science data acquisition in the real mission execution. These issues provide great motivation to upgrade the speed and accuracy of the design process. Consequently, SCENAREO has been developed to analyze and coordinate mission operations automatically and optimally based on the inputs of subsystem performance and resource parameters, target information, and science objective specification. In addition, SCENAREO creates the operation commands automatically based on the activity and command dictionaries. Many time-varying space environmental conditions and resource availabilities (such as on-board memory capacity, power supply, and downlink rate) are essential factors in the analysis. They are dynamically updated to assure analysis accuracy. The SCENAREO analysis algorithms can also detect design faults with respect to system performance, resource

[1] 0-7803-8155-6/04/$17.00© 2004 IEEE
[2] IEEEAC paper #1001, Version 3, Updated September 30, 2003

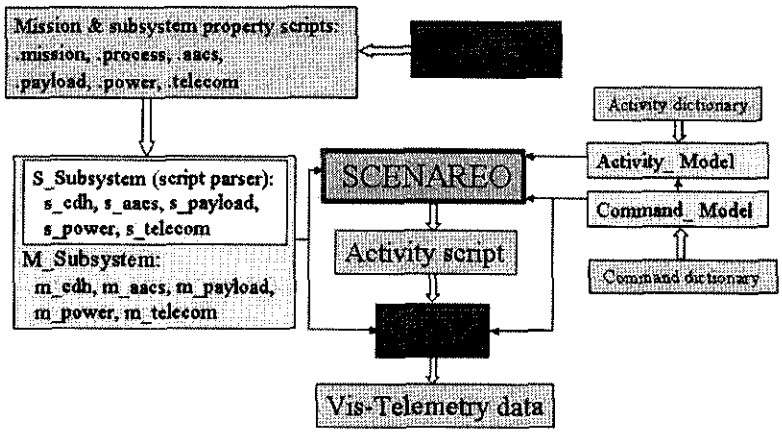

Figure 1 Virtual Mission Operation Framework

availability, and science objectives. This detection could effectively feedback to the mission and system design entity for fixing the errors as needed.

Figure 1 illustrates the composition of the VMOF system. MOB generates parametric subsystem model scripts for mission scenario design and simulation. The created model scripts include the parameters for mission condition and properties for science-data acquisition-related subsystems. With those property scripts as one of the inputs, SCENAREO composes mission activity scripts that serve as major inputs to the TSOS system. TSOS further simulates and validates the mission activities.

In Figure 1, S_Subsystem and M_Subsystem are two major libraries that SCENAREO builds upon. S_Subsystem parses subsystem property scripts and creates subsystem property objects. M_Subsystem, derived from S_Subsystem, provides methods to compute subsystem performance at various conditions or time points. Some of the static performance data are directly provided from subsystem property scripts, and some non-static performance data are computed by M_Subsystem based on time-varying factors. As shown in Figure 2, M_Subsystem mainly contains modules for subsystems such as attitude and articulation control system (AACS), payload, telelcom, command and data handling (CDH), and power subsystems. CDH is built upon all other modules in order to access all of their performance functions for all mission command and data handling purposes.

Activity_Model and Command_Model are another two major libraries that SCENAREO builds on. They provide functions for operational analysis and command composition. All calculation in these libraries relies on the information provided by the activity dictionary and the command dictionary. The activity dictionary defines the mapping between operations and commands, i.e. the potential command combination to achieve certain operation. All the possible operation constraints and conditions are also listed in the activity dictionary. Operation constraints and conditions are used to bound the activity model to what kind of variables should be used in analyzing the mission operations. The command dictionary defines the subsystem commands and the formats of command parameters. The possible conditions and constraints, and the resources to be consumed for issuing the command, are also listed in the command dictionary. Command conditions and constraints provides the simulation system the checking items while executing the command. Command resources triggers the operation analysis module to check and comprise the availability of resource usage while designing the activity.

By use of these linked libraries, SCENAREO analyzes the mission operations in a model-based style. Each model has its own capability and can be easily expanded for various mission needs. As compared to other mission planning and scheduling works [3], SCENAREO with its well organized analysis structure makes the mission activity design work in more robust, flexible, and efficient ways.

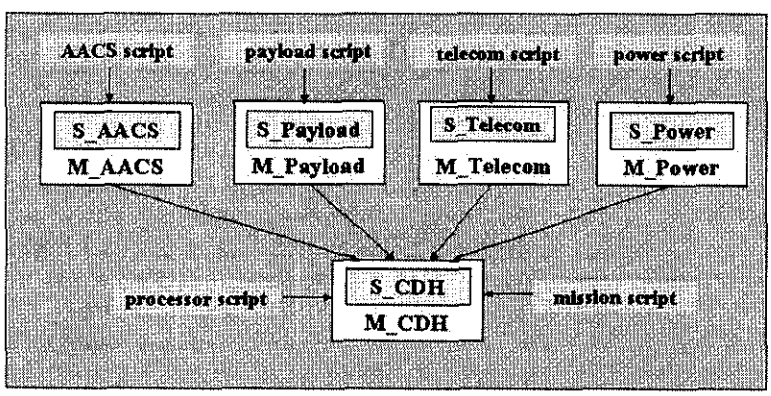

Figure 2 Subsystem Models

The main focus of current SCENAREO development is to streamline the early design phase of the Project Design Center (PDC) at JPL with the science-return validation process.

2. Software Layout

The MSIM group at JPL has developed mission scenario design tools for flyby missions [4] and orbiter missions. SCENAREO software is mainly developed for use on orbiter missions. As shown in Figure 3, SCENAREO is composed of three major components: User Interfaces (UIs), Analysis modules (Pre-Activity Analyzer and

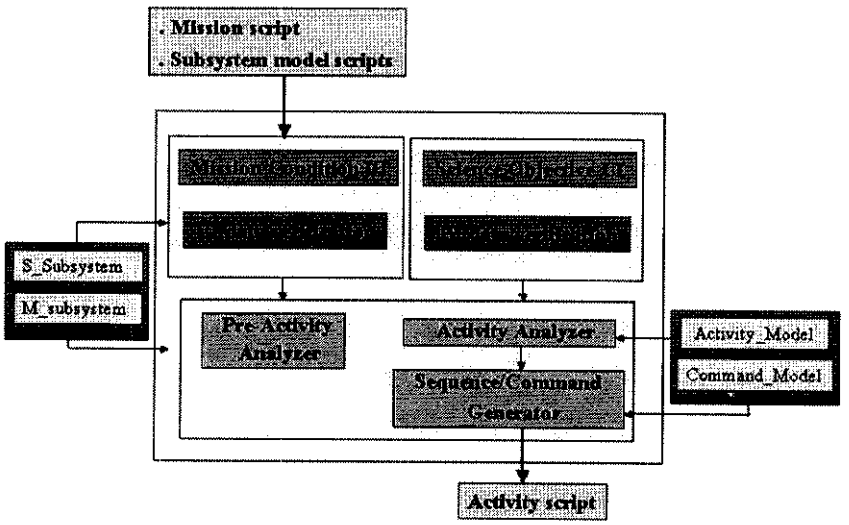

Figure 3 SCENAREO Software Layout

Activity Analyzer), and Sequence/Command Generator. Mission and subsystem property scripts are the main inputs to the software, and activity scripts are the main outputs. SCENAREO interacts with subsystem models, the activity model, and the command model to analyze and create the mission activity.

The Mission Condition UI and the Science Objective UI are two major UIs. The Mission Condition UI has entries to define mission target, mission time, orbit parameters, and subsystem model scripts as shown in Figure 4. The Science Objective UI (as shown in Figure 5) defines operation conditions and science objectives.

The entry values of the Mission Condition UI can be either read from a mission script (.mission) created by MOB or directly entered from UI users. After all required values have been entered, pressing the "make models" button at the bottom of this UI page creates spacecraft orbiter object and subsystem performance models for CDH, AACS, payload systems, telecom systems, power systems, etc. The M_Subsystem and S_Subsystem libraries construct all models. Each performance model contains methods to compute the operational performance of the relevant device or instrument at various conditions and constraints. The step of "Make Model" lays the foundation for further activity analysis.

Once "Make Model" is done, the characteristics of each available instrument or device are separately extracted for surface mapping, target observation, and downlink operations from the relevant models and filled into the Science Objective UI as shown in Figure 5. The Science Objective UI provides the entries to record the operation constraints and conditions that consistently meet the desire of the science objectives. The user of SCENAREO must know how to translate the literally stated science objectives into the operation condition entries in the Science Objective UI of Figure 5. The entered specifications are mainly for data-acquisition related operation, such as surface mapping operation, target observation operation, and downlink operation. For all entered operation constraints and conditions, SCENAREO interacts with the M_Subsystem, the Activity Model, and the Command_Model to evaluate the subsystem performance, trade off the resource usage, and produce commands for accomplishing those operations and meeting the science objectives.

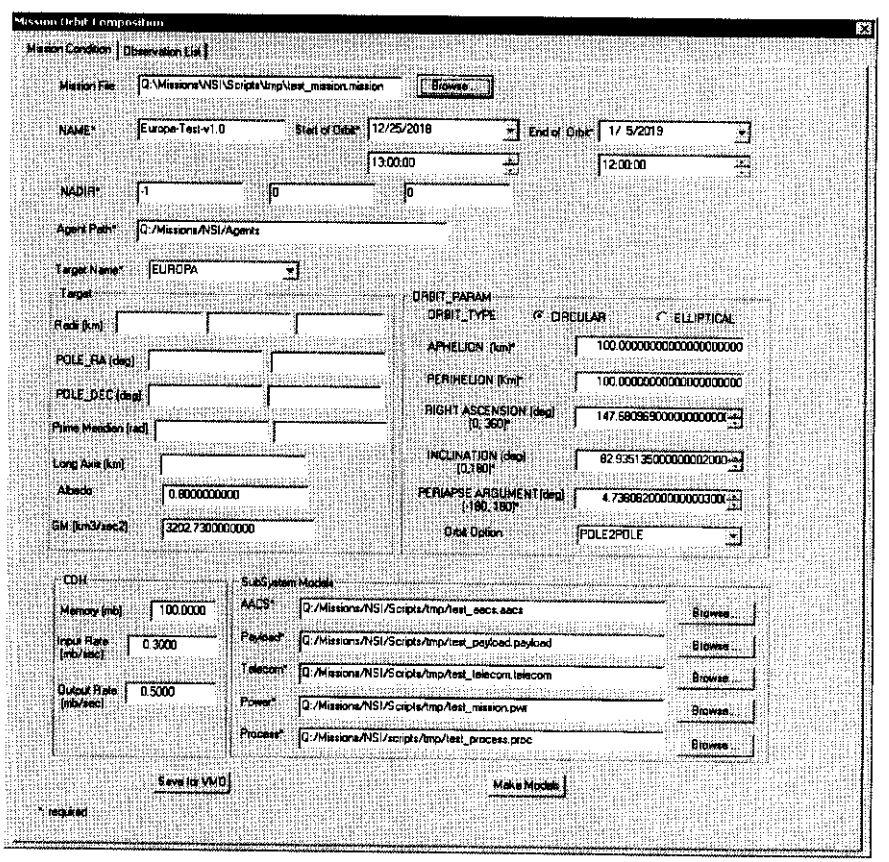

Figure 4 Mission Condition UI

Operation conditions contribute to the analysis of subsystem performance, either time varying or not. The operation condition entries listed in Figure 5 for surface mapping, target observation, and downlink are briefly explained as follows.

Surface mapping acquires data from the target within a valid period. If a payload instrument, such as a camera is selected to accomplish this operation, then a series of exposures will be taken during that period. The footprint coverage and spacecraft speed mainly determine the exposure frequency. If a payload instrument (such as a radar) is selected, then it needs be turned "ON" to execute swath operation during that period. The determination of valid operation period not only depends on the selected instrument and if it is gimbaled but also some other conditions, as listed in the mapping operation table of Figure 5. They are target surface incidence angle coverage, maximum observation time relative to the entire orbit period, and the instrument pointing angle relative to nadir for gimbaled instruments.

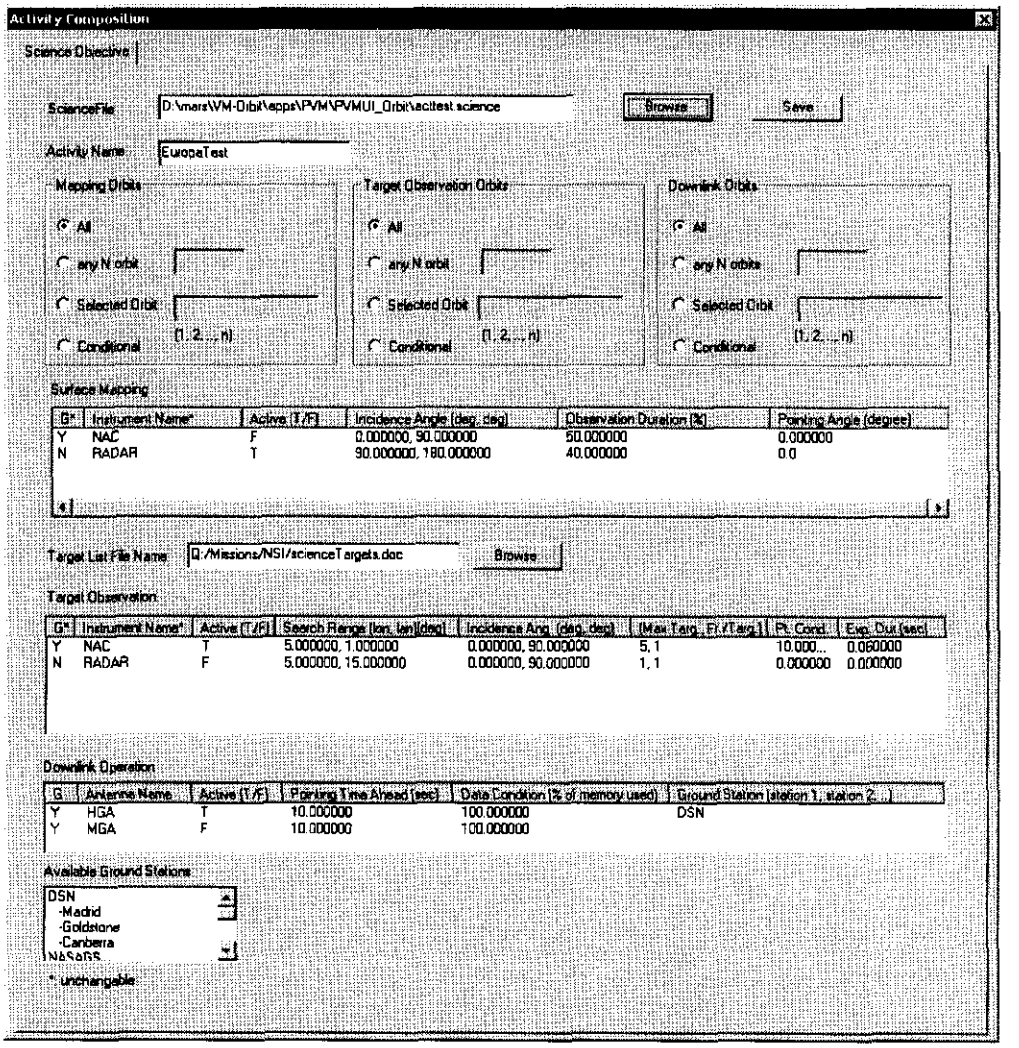

Figure 5 Science Objective UI

Target observation applies payload instruments to acquire data from some target spots that are optimally selected from a list of targets. The file, which contains all target candidates, is specified in the field of "Target List File Name," as shown in Figure 5. The qualification of each target contained in the file is examined based on if it meets the observation conditions entered in the target observation table, such as surface incidence angle coverage and maximum targets. The list of maximum targets and some target priority rules define the formula about how to pick up the target if there is more than one qualified target available during a limited operational period. The search range entered in the target observation table defines the latitude and longitude coverage of each search step for searching the targets. When and how to execute the "expose" operation for the selected target depends on the valid observation period of the target, the time needed to turn the instrument to point the target, and the observation condition entered in the target observation table, such as the minimum number of frames per target to be taken, the settling time after pointing and before exposure, and the exposure duration for each snap.

Simultaneous operation of multiple instruments is allowed for both surface mapping and target observation operation. Scheduling operations of both gimbaled and non gimbaled instruments based on all given constraints and conditions is a challenging task in SCENAREO analysis.

Downlink operation transmits all collected data from surface mapping and target observation back to the ground station system. One of the data-transmission antennas on the spacecraft can be manually selected from the UI or dynamically selected through internal optimal analysis. The ground system receiving the data also must be selected from a list of available ground stations. The valid downlink time and duration mainly depend on the pointing ability of the antenna, the magnitude of all collected data, the on-board memory input and output rates, the downlink rate, and the

required downlink conditions. These are shown in the downlink operation table of Figure 5, and they include "data condition" (meaning the capacity allowance of on-board memory), "pointing time ahead" (meaning time reserved after the antenna points to Earth and before data transmission starts), etc. The Deep Space Network (DSN) downlink rate is currently computed by interpolating the sampled downlink rates from a reference table, which is built by accessing a remote telecom performance analysis server at JPL [5]][6]. Future investigations will explore other ways to build the reference table for various telecom transmission systems.

The operational conditions specified for each operation in Figure 5 in default apply to all orbits. However, some times it is not necessary to have all of a day's orbits repeat every operation. For example, a mission may need to downlink only once per day. The execution of each operation may have its own "orbit condition" defined as shown above the surface mapping table in the Science Objective UI of Figure 5. The options of orbit condition for each operation include "operation for all day's orbits," "operation for any N orbits of a day (N is entered)," "operation for selected orbits of a day" (selected orbits are entered), and "operation for certain conditions." (More will be defined and implemented in the future.)

Based on the orbit condition specification of all operations,

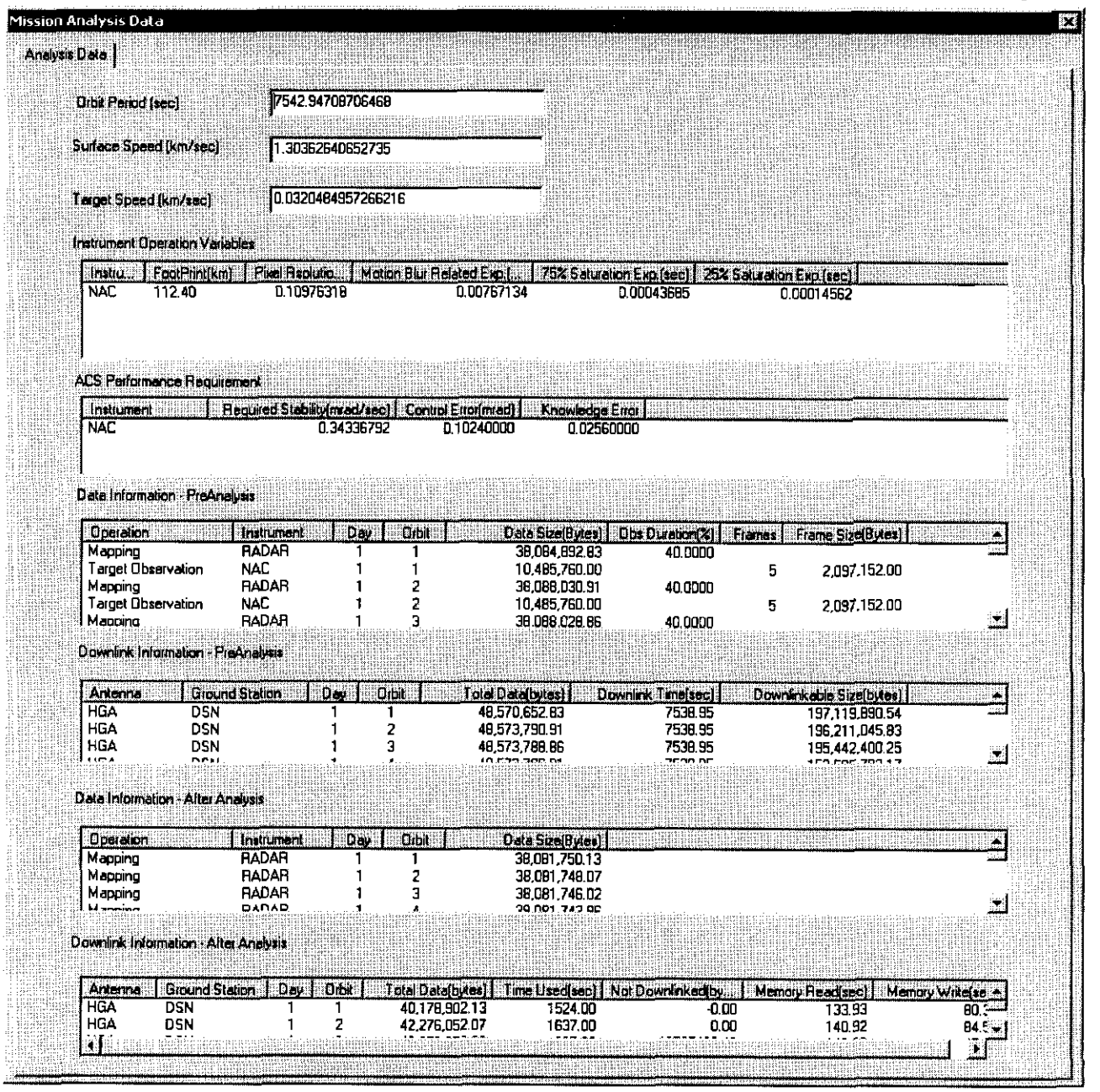

Figure 6 Analysis Data

either orbit-based or daily-based mission activity may be created through SCENAREO activity analysis.

3. Analysis Modules

As the desires of science objectives are translated into the operation conditions and orbit conditions specified in the Science Objective UI, how to operate the instruments or devices to meet the objectives is further analyzed by the SCENAREO central analysis modules, i.e. Pre-Activity Analyzer and Activity Analyzer. Pre-Activity does a preliminary analysis before activity analysis based on some static subsystem design parameters and some operation conditions specified in the Science Objective UI. This speeds the searching time to get good initial values for entering into the Science Objective UI in the early design stage. Activity Analyzer analyzes, creates, and optimizes operational activity according to the science objective requirements. This analysis step validates the setting of the Science Objective entries to a certain high accuracy level. Figure 6 shows the page that reports the analysis results before and after activity analysis.

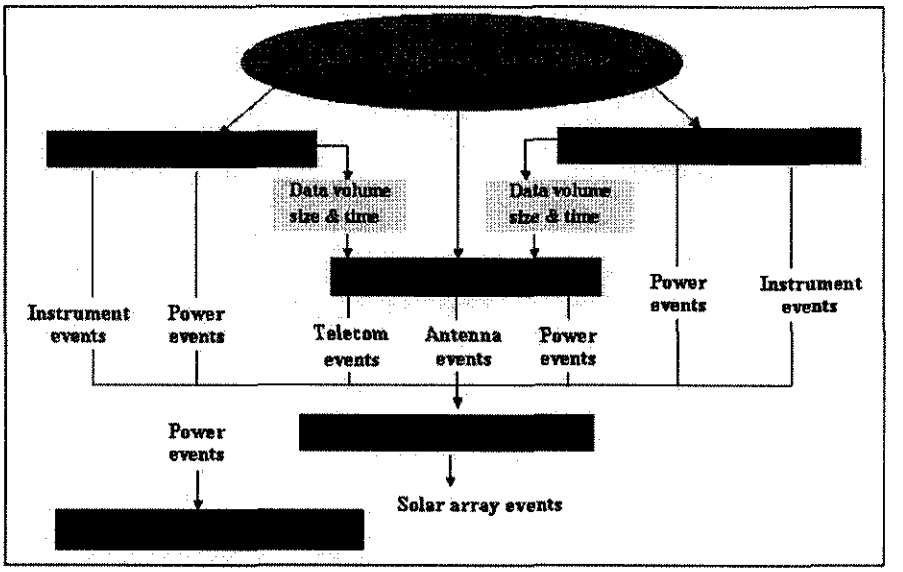

Figure 7 Mission Activity Analyzer

Pre-Activity Analyzer

Based on the subsystem static properties and some time-independent setting in the Science Objective UI, Pre-Activity interacts with subsystem models to make a rough estimation regarding operation requirements and performance. As shown in Figure 6, the items to be analyzed and reported mainly include instrument operation variables, ACS performance requirements, data information, and downlink information. Instrument operation variables provide data related to the payload instrument's footprint size, pixel resolution, motion blur, and saturation-related exposure duration. The calculation of those variables are respectively related to the values of some mission and subsystem parameters, such as the payload instrument's field of view (FOV), sensitivity, and detector size; the target's light reflectance; and the spacecraft altitude. The motion blur and saturation-related exposure duration give the activity designers reasonable data ranges while setting the exposure duration in the target observation table of Figure 5. ACS performance requirement provides the data to assure target tracking accuracy for non-gimbaled instruments. They are mainly derived from saturation-related exposure duration. The ACS performance requirement is useful for system design. Data information mainly calculates the data size and valid operational time for surface mapping and target observation for the instruments selected from the Science Objective UI. For surface mapping, the total data collected is according to the observation duration entered in the surface mapping table of the Science Objective UI for either radar-type or camera-type instruments. From the analysis result, users can get rough ideas about how much data are collected and if the desired observation duration is achievable or not. For target observation, the data collection is based on the size of one frame and the frames to be taken. The total number of frames to be taken is roughly estimated based on the maximum number of targets to be observed and the minimum frames per target, as specified in the target observation table of the Science Objective UI. Downlink information analysis calculates the science data collected from either surface mapping operation or target observation operation, the valid downlink period of each orbit, and the maximum data that can be sent back to Earth within the valid downlink time based on the ground station selection. From the analysis result, users can approximate the downlink time and memory capacity (one of the downlink operation conditions entered in the Science Objective UI). Data information and downlink information analysis directly aid the mission designer in setting some of the entries in the

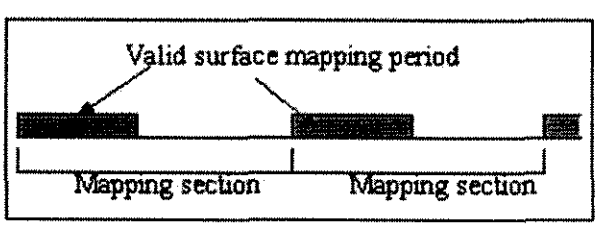

Figure 8 Mapping Section

science objective UI. Through these back-and-forth check-and-verification procedures, the Pre-Activity analysis not only helps the user set the requirements for the science objectives but also lets the user avoid some impossible settings while planning the mission objective. The more complicated interdependencies among all operations are further analyzed by Activity Analyzer.

Activity Analyzer

Activity Analyzer mainly interacts with all subsystem models, the activity model, and the command model to configure the proper operational commands to execute all operations according to the science objective entries. The creation of operational commands includes tasks to compute the command issuing time, compute the resource consumption, compose all command required parameters, and analyze the interdependency among all relevant devices.

Active Analyzer starts the analysis task by computing "mapping section" first. Then it repeats all analytic computation for each "mapping section" period. "Mapping section" is a time period defined as the interval between the beginning time of two consecutive valid surface mapping periods. The ending of each mapping section is the beginning of next mapping section, as shown in Figure 8. For a short orbit-period case, such as a 2-hour orbit, each mapping section should have the same time length as that of one orbit period. For the case of a long-orbit mission, in which one orbit may take a couple of days, one orbit period may contain more than one mapping section.

Surface mapping operation occurs within a continuous time period; however, target observation operation occurs at sparsely spread time points. Therefore, in order to schedule these two data-acquiring operations in a more time-efficient way, Activity Analyzer first searches all mapping sections for each orbit to be analyzed. Then, as the flow in Figure 7 shows, Activity Analyzer sequentially searches the valid surface mapping period, the valid targets observation time, and the valid downlink time period within each "mapping section" based on all conditions and constraints. Once the valid operation time is determined for all operations, how all subsystems coordinate to perform the operation is analyzed, and a series of subsystem activity events are created accordingly. Each activity event contains one subsystem command, the command time, and the command parameters.

SCENAREO features model-based analysis style. The determination of all operational command elements relies on the interaction among Activity Analyzer and all supported models. Subsystem models provide the functions to calculate various subsystem performances under various conditions and constraints. Activity model is in charge of analysis at the operation level. The Command model supplies the command level functions, such the command format setting, the command syntax checking, and the notification of command related condition and resource usage for mission operation planning and simulation. Subsystem models and Activity model provide the analysis functions on a time-varying basis. The command model mainly has the functions to deal with operation commands of time-independent portions.

Figure 9 briefly states the concepts about how Activity Analyzer conducts analysis flow and communicates with all supported libraries to accomplish the entire mission operation analysis. First, Activity Analyzer sequentially passes operations and operation-related conditions and constraints to the activity model. The activity dictionary defines the needed operation analysis information, including the potentially needed operation constraints, the operation conditions, and all possible subsystem commands to perform certain operations. Based on the operations and commands mapping defined in the activity dictionary, the activity model dispatches each operation-related command to a

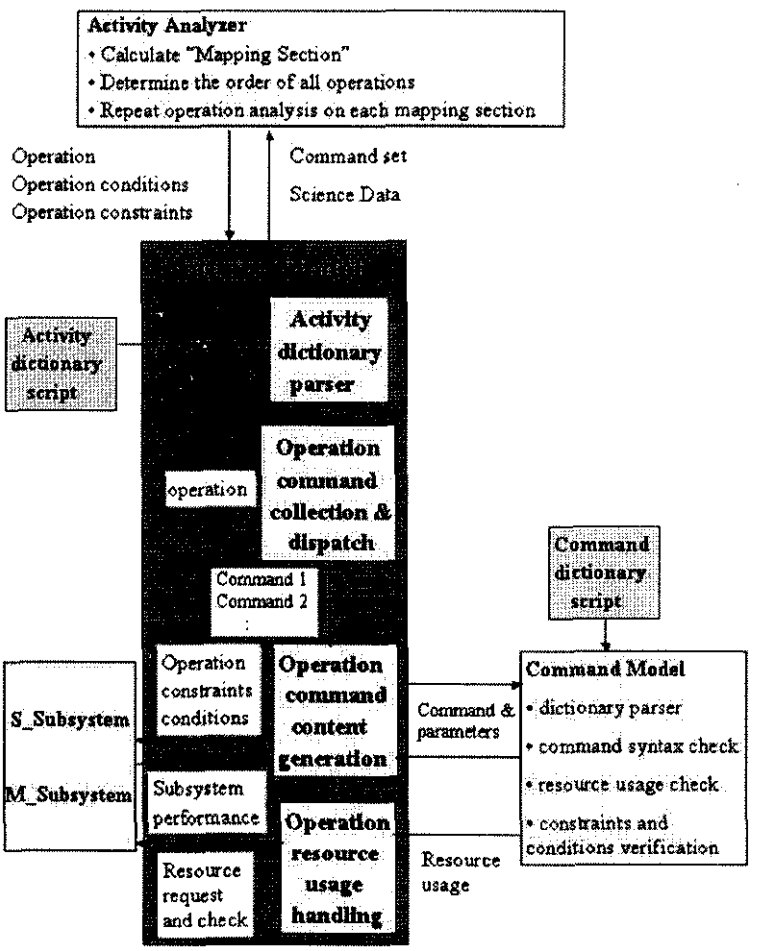

Figure 9 Activity Model

command content-generation module. This module plays the main role in generating the entire command content, such as command issuing time and command parameters. Simply speaking, the activity model converts one operation into a set of activity events containing commands and all command elements based on operational conditions and constraints. The command content generation module interacts with subsystem model (M_Subsystem) to get operation performance, seek a proper and best time value as the command time, and calculate all parameters based on the command time and conditions. As values of all command elements are configured, the command content generation module contacts command model to check if all command parameters have valid syntax and values. The command model does the checking based on the rules set in the command dictionary. Moreover, according to the command dictionary, the command model can provide operation planner or simulator the information of command related constraints, conditions, and resource usage while planning or simulating the execution of the command. The command dictionary defines the command syntax, command constraints, command conditions, and resource usage for all available subsystem commands.

What Activity Analyzer does is not only to analyze the subsystem operational activities to achieve the science objective but also to validate the feasibility of the science objective based on time-varying performance of the subsystems. The data information and downlink information after analysis is reported into Analysis Data page as shown in Figure 6.

The details of each operation analysis are as follows.

Surface Mapping Analysis

The valid operation period is computed for the surface mapping operation according to the selected instrument and the specified operational conditions, incidence coverage, maximum observation duration and pointing angle with respect to NADIR for the gimbaled device. Then the Activity Model collects commands for relevant subsystems to accomplish the mapping operation. If a radar-type instrument is selected, then "ON" and "OFF" commands for this type instrument are issued to start and end the mapping operation. If a camera-type instrument is selected, then a series of "EXPOSE" commands or a "PUSHBROOM" command spanning the entire valid operation period is issued to mimic the surface mapping operation. Each snap's coverage is one footprint. In order to seamlessly align all footprints, the footprint coverage and the spacecraft speed (or the target speed) are important factors for determining the time interval between two consecutive snaps. For either radar-type or camera-type instrument, all data collected during the entire valid operation period are calculated.

If the instrument is non-gimbaled, then ACS may need to turn to and track in a certain direction to keep the instrument aiming at the target while the mapping operation proceeds. The analysis will check if either ACS or the gimbaled instrument itself can point to the target and track the target pointing for data acquisition The series of commands used for this turning movement is "SET_RATE," "POINT," and "TRACK."

All available gimbaled instruments can be selected at the same time to operate surface mapping simultaneously. However, only a maximum of one instrument without gimbals can operate one time. If more than one non-gimbaled instrument is selected, two or more instrument's turning may involve that of ACS at the same time; then, some trade-off must be evaluated in terms of cost, data-acquiring priority, or other optimization concerns.

As the Activity Model analyzes the surface mapping operation in terms of operation time, device turning, and data size, a set of subsystem commands and relevant parameters is calculated and passed to the Command Model for further verification.

Target Observation Analysis

Target observation operational analysis starts after the surface mapping analysis. On the time line of the entire mission, the surface mapping operation takes pieces of continuous time periods, while target observation operation takes scattered time points.

If the instrument selected for target observation is gimbaled, the entire area of the "Mapping Section" that the analysis works on is used for searching the target candidates, pointing the instrument, and taking the science data. If the instrument is non-gimbaled, then the pointing and tracking of the instrument is limited to the ACS non-occupied period. If the movement of ACS is already assigned to aid surface mapping operation or target observation operation of any non-gimbaled instrument, then that period has to be excluded for that non-gimbaled instrument to do the target observation operation.

The first step in target observation operation analysis interacts with the subsystem model's target search module to find qualified targets from an available target list according to the valid operation period and some operation conditions, such as target surface incidence angle. As targets that meet the conditions are found, the associated information of each target (such as valid observation period, target location, and target size) are passed back to target observation operation analysis. For a camera-type instrument, either the "EXPOSE" command or the "PUSHBROOM" command is used to gather data from the target. In the target observation operation analysis module, a time point within the associated valid observation period of each target is sought as the optimal time point to schedule the command for data acquisition. The computation of this optimal time point is determined by some factors, such as the least moving steps

for a gimbaled instrument. If an instrument is gimbaled, "SET_RATE," "POINT," and "TRACK" commands are used to turn the instrument itself to aim the target before the time to gather the science data. For these commands, the time duration for turning the instrument to point at the target, the turning rate, and the track timing are computed based on the instrument's gimbal properties plus the settling time ahead of pointing specified in the science objective UI. If the instrument is non-gimbaled, then ACS controls the turning and pointing activities. If "EXPOSE" or "PUSHBROOM" commands are issued for a camera-type instrument, actual exposure time, exposure frequency, and the interval between the two exposures are analyzed based on the conditions specified in the UI, such as maximum target exposure, minimum exposure per target, and exposure duration.

With surface mapping and target observation analysis, all resource usage changes (such as power status changes on payload instruments and the ACS system and changes in memory consumption) are also considered for coordinating the operations.

Downlink Operation Analysis

Downlink operation transmits the data from surface mapping and target observation operation to Earth ground stations. The downlink operation can be executed orbit-by-orbit or day-by-day style according to the orbit requirements. The orbit-by-orbit case downlinks all data collected in one orbit in that orbit, and the day-by-day case downlinks all data collection of that day.

For each downlink operation, the eligible downlink period in each "Mapping Section" is analyzed based on the visibility of the Earth for either the gimbaled or the non-gimbaled antenna case and if the visibility could be tracked. Within the eligible downlink period, the actual durations used for downlink operations are determined based on the data size, the time during which the data are produced by previous target science acquisition operations, the memory accessing (read/write) time, and the downlink rate. Memory accessing time is computed by the CDH model such that the available memory size and memory read and write rate are kept. The downlink rate is time variant. With the telecom model, it is obtained by interpolating the downlink rate reference table containing rate samplings on some telecom variables (such as distance, elevation angle, encoding method, spacecraft antennas, and ground stations). The DSN downlink reference table used by SCENAREO is built by accessing a remote telecom forecast and prediction tool [6]. Since the antenna needs to point and track the Earth ground station before downlink operation starts, the duration used for pointing, the time when the pointing starts, and if the Earth pointing can be tracked while downlink occurs also needs to be computed and analyzed. To accomplish the downlink operation, If the antenna is gimbaled, the main commands for the selected antenna are Earth "POINT" and "TRACK," and the downlink rate-setting-related "SET" and "DOWNLINK" commands are issued for the telecom system to transmit the data. If the selected antenna is non-gimbaled, then the ACS controls the turning, and the downlink time is restricted within ACS non-occupied period.

With downlink operation analysis, all resource usage items (such as power state changes on telecom and antenna systems, and memory usage) are also considered. Downlink operation mainly generates activities for the telecom system, the antenna system or ACS, and the power system.

Sun Tracking Operation Analysis

If some activities are composed for either surface mapping operation or target observation operation, then Sun tracking operation analysis is required to meet the Sun tracking requirement for those operations. Sun tracking analysis mainly points and tracks the solar array system to the Sun based on the needs of Sun's visibility for all science data acquiring operations. Sun tracking operation analysis mainly generates activities for the solar array system and the power system.

Power Analysis

Power analysis is another essential feature in SCENAREO. It checks the power availability for the designed activities by linking with JPL's multi-mission power analysis tool, MMPAT [7], which provides the capability to simulate the power generation and consumption based on mission power settings. The subsystem power model wraps up MMPAT's functionality and provides power analysis functions to update the power states and check the power availability at certain time points. Two types of power scripts are needed for power analysis. One is power property, which defines the configurations of power subsystems (such as solar array and battery). The other is power profile, which defines the power consumption of devices at different operation states. These two scripts provide the static information for power analysis. The power analysis function first dynamically updates the power states of all devices as the operation changes. Then, it simulates and analyzes the power consumption and availability based on those time-invariant and the time-variant settings. In order to calculate the power availability accurately, the Sun angle and Sun distance from the spacecraft are periodically updated while the power simulation is ongoing. Through the power analysis, SCENAREO has the capability to report based on current configuration whether enough power can be provided to perform the designed activities. The ultimate goal of SCENAREO is to have an activity design module that can automatically feedback power analysis results to all operations analysis modules and re-adjust the activities of all devices to assure enough power and sufficient subsystem operation to achieve the mission objectives.

4. SEQUENCE/COMMAND GENERATOR

Because the activity analyzer has validated the science objectives and designed the mission activities according to the specifications entered in Mission Condition UI and Science Objective UI, all activities-associated commands can be dumped by the Command/Sequence Generator in a predefined format, as shown in Figure 10. Figure 10 and Figure 11 are examples of an activity script and an observation script output from Command/Sequence Generator. These scripts show an orbit-by-orbit case in which two orbits' activities reflecting the science objective entries shown in Figure 6 are listed.

Observation script mainly contains the reference time for each "Mapping Section." Each "ACTIVITY" in the activity script of Figure 11 represents the subsystem operational commands within the "Mapping Section." The command time listed in each command line represents the time relative to reference time of that activity. For surface mapping operation, "Radar" is chosen. For target observation operation, "NAC" is chosen to take frames from five targets at most. In orbit 1, only two targets are found to meet the science requirement. For downlink operation, antenna "HGA" is selected and has the pointing ready for data transmission. Power part has the operation state set for all the above-stated devices and instruments.

5. CONCLUSION

The development of SCENAREO shows that this mission scenario design tool can automatically convert mission-desired science objectives into a sequence of operational subsystem commands through effective subsystem performance analysis, subsystem inter-dependency analysis, and operation-to-command conversion analysis. With the easy-to-use, parameterized UIs, the mission designer enters values for mission conditions and science objectives. By using Pre-Activity analysis, the mission designer can get ideas about the reasonable operation ranges for most entries in the two UIs within a short time. The Activity Analyzer effectively designs, trades, and optimizes the mission activities by the use of model-based analysis. The models developed can be reused for analyzing other missions. This can greatly reduce the time for designing a new mission. The Sequence/Command generator hierarchically creates the activity script, which contains sufficient commands to accomplish the designed activities. The completeness, the efficiency, and the adaptability of SCENAREO greatly support and benefit the entire lifecycle of mission scenario design.

The well established infrastructure of the entire VMOF and SCENAREO system have proved that SCENAREO is a powerful and unique mission activity design and validation tool. Accuracy and the capability of optimization are two major goals to pursue in the next SCENAREO development stage. In the future, more optimization approaches will be investigated and integrated into the subsystem performance analysis models. As for accuracy, functions (which contribute more detail to the performance of subsystems), computation additions, and modifications are being considered for activity analysis.

REFERENCES

[1] Meemong Lee, "Virtual Mission Operation Framework," IEEE Aerospace Conference, Big Sky, Montana, March 2004.

[2] Wenwen Lu, "TSOS – Time-based Spacecraft Operation Simulator," IEEE Aerospace Conference, Big Sky, Montana, March 2004.

[3] Edward Bakewell, Dong-Kyoung Choe, M. HaQuang, S. Kala, S. Lawler, R. Schoen, "Planning and Scheduling Strategy for Non-Deterministic Events",

[4] S.Y. Wang, M. Lee, R. J. Weidner, "High Level Mission Scenario Design System," Presentation at JPL 2001 IT Symposium., JPL, Pasadena, California, May 2001.

[5] M. Lee, R. J. Weidner, S. Y. Wang, B. M. Lininger, "Science-Return Modeling and Simulation," IEEE Aerospace Conference, Big Sky, Montana, March 2003.

[6] S. Y. Wang, M. Lee, K. Peter, C. W. Lau, "Integration of Virtual Mission (VM) and Telecom Forecaster Predictor(TFP)," Presentation at JPL 2002 IT Symposium, JPL, Pasadena, California, November 2002.

[7] Mark Kordon, Eric Wood, "Multi-Mission Space Vehicle Subsystem Analysis Tools," IEEE Aerospace Conference Proceedings, Big Sky, Montana, March 2003.

Biography

Shin-Ywan Wang *is a senior engineer of the Mission Simulation and Instrument Modeling group at JPL. She supports virtual mission simulation system for integrating the subsystem performance analysis tools, designing mission scenario language, and developing instrument property modeling methods. She has a bachelor's degree in Electronics Engineering from National Chiao-Tung University, Taiwan and a master's degree and doctoral degree in Electrical and Computer Engineering from University of California, Irvine.*

ACKNOWLEDGEMENTS

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.

```
ACTIVITY_LIST
{
  ACTIVITY = "EuropaTest-0"              # orbit 1
  {
    AACS
    {
      ACS
      {
      }
      SOLARARRAY = "PrimaryPanel"
      {
            0 RESET;
            1 TRACK SUN;
         7542 TRACK_OFF;
      }
      ANTENNA = "HGA"
      {
            0 RESET;
            1 SET_RATE 0.057319 0.057319;
            3 POINT EARTH;
         3008 TRACK EARTH;
      }
    }
    PAYLOAD
    {
      CAMERA = "NAC"
      {
            0 RESET;
         2256 SET_RATE 1.333333 1.333333;
         4512 POINT EUROPA -1.130000 346.770000;
         4545 TRACK EUROPA -1.130000 346.770000;
         4547 EXPOSE 0.060000 "NAC-0-1.dat";      #site1, size = 2097152.000000 bytes
         4551 READ_ALL "NAC-0-1.dat";
         4552 SET_RATE 1.333333 1.333333;
         4553 TURN 0.0 0.0;
         5432 SET_RATE 1.333333 1.333333;
         6278 POINT EUROPA 74.600000 236.450000;
         6562 TRACK EUROPA 74.600000 236.450000;
         6564 EXPOSE 0.060000 "NAC-0-2.dat";      #site7, size = 2097152.000000 bytes
         6568 READ_ALL "NAC-0-2.dat";
         6569 SET_RATE 1.333333 1.333333;
         6570 TURN 0.0 0.0;
      }
      RADAR = "Radar"
      {
            1 ON "Radar-0.dat";      #size = 37406014.799151 bytes
         3018 OFF;
      }
    }
    TELECOM
    {
            2 ANTENNA HGA;
            3 SET DOWNLINK 211892.731130;      #rate unit: Bits Per Sec
         3018 DOWNLINK "Radar-0.dat" 523.000000;
         3543 SET DOWNLINK 211850.542560;      #rate unit: Bits Per Sec
         3544 DOWNLINK "Radar-0.dat" 890.000000;
         4552 DOWNLINK "NAC-0-1.dat" 80.000000;
         6569 DOWNLINK "NAC-0-2.dat" 80.000000;
    }
    POWER
    {
      HGA
      {
         3018 SET "DOWNLINK" 523.000000;
         3544 SET "DOWNLINK" 890.000000;
         4552 SET "DOWNLINK" 80.000000;
         6569 SET "DOWNLINK" 80.000000;
      }
      NAC
      {
         4545 SET "TRACK" 2.000000;
         4547 SET "EXPOSURE" 1.000000;
         4553 SET "TURN" 33.000000;
         6562 SET "TRACK" 2.000000;
         6564 SET "EXPOSURE" 1.000000;
         6570 SET "TURN" 284.000000;
      }
      SOLARARRAY
      {
         1 SET "TRACK" 7541.000000;
      }
      RADAR
      {
         1 SET "ON" 3017.000000;
      }
    }
  ACTIVITY = "EuropaTest-1"    // orbit 2
  {
    AACS
    {
      :
    }
    PAYLOAD
    {
          :
    }
    TELECOM
    {
      :
    }
    POWER
    {
      :
    }
  }
}
```

Figure 10 Activity Script

```
S_OBSERVATION
{
    ACTIVITY_DEF = "orbittest.activity"

    OBSERVATION_LIST
    {
        OBSERVATION = "ObsTest-0"
        {
            ORBIT = 0, 0
            ACTIVITY = "EuropaTest-0"
        }
        OBSERVATION = "ObsTest-1"
        {
            ORBIT = 1, 1
            ACTIVITY = "EuropaTest-1"
        }
    }
}
```

Figure 11 Observation script