Andrea L. Henke, Richard H. Stottler
Stottler Henke Associates, Inc.
916 Holly Road
Belmont, CA 94002
(415) 595-1692 (415) 595-0358 (FAX)
E-mail: 76420,3622@compuserve.com

N95- 23755

KEY WORDS AND PHRASES

Artificial intelligence, intelligent entities, object-oriented, planning, scheduling.

ABSTRACT

Planning and scheduling of NASA Space Shuttle missions is a complex, labor-intensive process requiring the expertise of experienced mission planners. We have developed a planning and scheduling system using combinations of artificial intelligence knowledge representations and planning techniques to capture mission planning knowledge and automate the multi-mission planning process. Our integrated object-oriented and rule-based approach reduces planning time by orders of magnitude and provides planners with the flexibility to easily modify planning knowledge and constraints without requiring programming expertise.

MISSION PLANNING PROBLEM

High-level mission planning is begun from 5 to 10 years prior to launch. The goal of this planning is to establish a flight manifest, define the objectives, capabilities and constraints of the missions comprising the manifest, and translate those into hardware, software and flight procedures. The manifest must reflect the precedence and duration of Shuttle processing activities, constraints such as facility utilization, work shift requirements, interval between launches, maintenance requirements, and other processing ground rules, to achieve a specified flight rate. Each mission flow consists of a standard set of processes of varying durations applied to a specific Orbiter. The manifest must reflect the precedence of certain processes, the facilities required and the constraints upon Shuttle processing. Additionally, unplanned or non-standard activities must be incorporated into a specific mission's flow.

Another important objective of high-level mission planning is to explore alternative planning options. These exercises determine how the flight manifest is affected when program ground rules are changed, new facilities are constructed, launch delays are anticipated, or new vehicles are introduced. The planning options can be very diverse and speculative, involving concepts ranging from the impact of facility repairs, to crew rescue at the space station, to concepts still on the drawing board. Additionally, there is considerable time pressure to produce answers to "what if" questions quickly.

Until recently, the manifest planning process was largely manual, performed by planners with many years of experience in the domain. Because of the great importance, diversity and complexity of the high-level studies, mission planners can dramatically benefit from our automated system for manifest planning. The object-oriented approach results in a system that is comprehensive and flexible and can accommodate their changing needs.

AUTOMATED PLANNING SOLUTION

In a project funded by NASA, we developed the Automated Manifest Planner (AMP) to solve the multi-mission planning problem. AMP is a flexible, comprehensive planning tool which draws on artificial intelligence techniques from a number of different areas to meet the requirements for manifest representation, manifest design and manifest analysis. AMP is designed to capture the expertise of experienced manifest planners and provide comprehensive, interactive manifest planning assistance. The planner can choose among different planning methods for use at various levels of the scheduling process. AMP can automatically plan missions, taking into consideration resources, ground rules, constraints and planner heuristics to improve the scheduling. By making use of generic mission definitions and relevant constraints, AMP will generate a manifest from scratch or replan all or portions of an existing manifest. The resulting manifest has no resource conflicts, no broken ground rules, and all processing performed in the correct order. By utilizing planner rules of thumb, AMP allows novices to produce quality manifests.

AMP provides flexibility by allowing the planners themselves to modify ground rules, facilities and missions and interactively edit the manifest produced. AMP improves the turnaround time on planning options by orders of magnitude and dramatically reduces the time needed to modify and maintain the manifest. The tool allows timely response to both simple and complex studies, from slips in dates or modified task durations, to new facilities, Orbiters, or different types of launch vehicles.

The manifests generated by AMP are displayed immediately on-screen in bar chart format. The planner may use the mouse to graphically edit flows, activities and other aspects of the manifest in order to bend the rules or seize particular opportunities. Although automated planning will never produce manifests with resource conflicts, these problems may be introduced through the editing process. AMP will shift dates forward to accommodate delays or minor resource changes where possible, and flag remaining conflicts. The planner can then either fix these problems by hand, or more efficiently, automatically replan that portion of the manifest.

Interactive explanation capabilities are provided in the AMP tool to give the planners insight into the reasoning that produced the manifest. This includes the reasons for particular resource/facility assignments, the reason float time is present, or the reason launch dates or other processing dates were pushed back. These explanations allow the planner to identify opportunities to improve the manifest and give the planners greater confidence in the manifests produced.

Because of the diverse and dynamic demands of manifest planning, AMP was necessarily designed to be a general scheduling tool, offering planners a host of planning methods and techniques for customizing the system for a particular planning situation. For this reason, AMP has broad applicability beyond NASA manifest planning.

TECHNICAL APPROACH

AMP uses a combination of artificial intelligence techniques to allow both the automatic generation of correct manifests and the improvement of these manifests through captured planner heuristics. We employ an object-oriented representation for capturing ground rules, constraints, activities, missions and resources. The heuristics planners use in generating and analyzing manifests are represented as rules. The planning techniques combine object-oriented programming and rule inference strategies.

Representing the Manifest

In order to automate the manifest planning process and allow comprehensive manifest design and analysis, one must first establish a representation of the manifest and its components. These components include the generic flows and processing activities, scheduled flows and processing activities, ground rules, planning constraints involving task sequencing and desirable conditions, and the available resources. These resources are varied and include Orbiters, payloads, launch pads, Orbiter Processing Facilities (OPFs), Mobile Launcher Platforms (MLPs) and other facilities, and time resources, relating to time needed by certain processes and time required at certain locations and on certain equipment, and calendar time constraints.

These diverse manifest components can be captured using object-oriented techniques. A generic flow for a type of mission is an object containing a list of generic activities which are themselves objects which include slots for the types of resources needed to perform the activity, as well as associated scheduling methods. A manifest is an object which contains a list of particular flows. These particular flows are copies of the corresponding generic flows and contain a list of copies of the generic activities. These activities are linked together in a network which describes the required sequencing of operations.

The resources required by activities are organized into an object class hierarchy. The super-class is Required Facilities which has subclasses of OPFs, MLPs, and vehicles, for example. The OPFs class contains the three OPF instances - OPF1, OPF2, and OPF3 - corresponding to the three available Orbiter processing facilities. The Vehicles class has subclasses of Orbiters and HLLVs (Heavy Lift Launch Vehicles). The Orbiters class contains 4 instances representing the four Space Shuttle Orbiters.

Constraints and ground rules may be represented using a combination of objects and rules, as appropriate. For example, one special required facility is called Space and has one instance. This one instance, along with the flight activity's requirement for a Space resource, represents the constraint that only one Orbiter can be in space at a time. Typical ground rules include Orbiter Maintenance Down Period (OMDP) times and locations, the influence of payloads on durations, and special procedures.

Capturing Planner Expertise

An important aspect of many AI development efforts is the capture of the corporate knowledge of the experts. By eliciting and storing the details of a process, novices can be productive even when the experts are unavailable. The required knowledge for manifest planning can be captured in a number of ways. First, the expert's knowledge about the events and processes in a typical mission is captured in a generic flow. The generic flow represents the overall sequence of the processing activities in a mission. This flow preserves the required order of those activities and the resources required for each activity. Second, alternative planning methods are used to capture the expert's approach to planning and resource allocation for the activities in a flow and the flows in a manifest. For example, the expert planner may schedule certain flow activities in a forward direction, a backward direction, or in a priority order from certain dates or activities. Finally, rules are used to capture exceptions or additions to the standard flow. A rule is attached to the object to which it relates. Rules often add or delete activities to the specific flow. For example, a rule adds the activities of transporting the Orbiter to and from Palmdale, California if OMDP processing is required and that processing should take place in California rather than at Kennedy Space Center.

Intelligent Entities

An object-oriented approach allows the system to represent activities and activity scheduling information as objects. The objects are organized into an object hierarchy or class structure, where objects in the same class share characteristics. The object hierarchy for AMP includes objects and classes of objects to represent manifests, individual missions, processing activities, facilities, vehicles, etc. These objects are not passive data, but individual, intelligent entities that can be requested to perform actions on themselves or each other. These objects know how to schedule and unschedule themselves, and plot and erase themselves.

When the planner wants to initiate planning of a manifest, he or she in effect sends a message to the manifest object telling it to plan itself. The manifest object responds by sending scheduling messages to each of its missions. Each mission schedules itself by sending scheduling messages to each of its constituent processing activities. Each activity schedules itself by sending messages to other activities and making scheduling requests of each of its required resource classes, such as the class of OPFs or the class of MLPs. The resource classes respond to schedule requests by sending messages to each particular resource in their class. Each particular resource then checks its own availability and sends that information back to the class which makes the best resource selection. As each activity responds to scheduling requests, it checks its own local slots for rules and scheduling method choices, firing rules and executing the appropriate scheduling methods. After all these recursive planning calls have been made, the manifest object plots itself on-screen. Plotting follows the same level-by-level sequence.

The concept of intelligent entities, described above, allows the planner to mix and match different scheduling methods for different entities. It also facilitates capture of the planners' heuristic knowledge by the planners themselves. Because the scheduling problem is broken down into so many separate smaller problems, very complex scheduling is performed by relatively simple methods. These simple methods allow the easy inclusion of rules to alter planning methods in certain circumstances. Because each entity represents such a small part of the overall problem, the rules required for each entity are very simple and few in number and are tailored to each object's planning method. There is almost no interaction between the rule bases, because they are only related to the intelligent entity (such as an activity) to which they are attached. The small number and simple form of the rules makes it easier for the planners to enter these rules themselves or to have semi-automatic learning capabilities generate the rules.

Another design principle of AMP is the philosophy of permitting the planners to access all parts of the system, including the resource hierarchy, generic and specific missions and activities, plot definition files, and rules attached to each entity. This philosophy gives the planners maximal flexibility to tailor AMP to fit their changing needs without requiring programming expertise.

AMP DEVELOPMENT

The AMP project involved extensive knowledge engineering with the NASA expert planners. AMP was developed as a series of incremental releases which provided extensive planning, plotting, and editing options and methods. The Mission Planning Office is using AMP to perform Shuttle manifest planning and the more speculative alternative planning studies. AMP can plan one year of Shuttle flows in one minute on a 486 PC.

CONCLUSION

AMP substantially reduces the time required to maintain NASA's flight manifest and perform studies. This improves response time and allows planners to play a more proactive role in the studies. By allowing the planners more time to concentrate on the significant or unusual aspects of scheduling, they may be able to generate better manifests, and produce them more quickly. Additionally, by modeling planner expertise, less experienced planners can take advantage of the knowledge of planning experts and generate better manifests or work with less supervision.

The flexibility required by the mission planners dictates that the tool be so flexible as to make AMP adaptable to almost any scheduling problem, including planning for detailed Shuttle and payload processing, manufacturing scheduling, etc. We recently completed a project for Johnson Space Center in which we applied AMP techniques to the planning of the crew activity timeline for both Shuttle and space station flight planners. We expect to implement a full-scale version for their daily use.