

A Criterion Autoscheduler for Long Range Planning

Jeffrey L. Sponsler

Space Telescope Science Institute
3700 San Martin Drive, Baltimore, MD 21218 USA
(410) 338-4565 sponsler@stsci.edu

Abstract

A constraint-based scheduling system called SPIKE is used to create long-term schedules for the Hubble Space Telescope. A meta-level scheduler called the Criterion Autoscheduler for Long range planning (CASL) has been created to guide SPIKE's schedule generation according to the agenda of the planning scientists. It is proposed that sufficient flexibility exists in a schedule to allow high level planning heuristics to be applied without adversely affected crucial constraints such as spacecraft efficiency. This hypothesis is supported by test data which is described.

1. Introduction

The scheduling of the Hubble Space Telescope (HST) is complex and involves many software systems. A long range planning system called SPIKE is integral to the process. Recently, SPIKE has been augmented with a new subsystem for the following reasons:

1. The planning scientists (who use SPIKE) were not able to make important changes to the behavior of the scheduling system without requesting that software developers effect code changes.
2. A set of scheduling rules that was provided by the scientists could not easily be encoded in the scheduler. Expert system technology was proposed as a way to give users control over the scheduling process.

The *Criterion Autoscheduler for Long range planning* (CASL) has been implemented to satisfy these requirements. An expert system methodology called *functional knowledge representation* (Lucks, 1992) has been implemented as a generic shell called the multiple criterion network (MCN). MCN uses *criteria* and *knowledge functions* which are defined here:

Criterion: A problem solving heuristic that is associated with some aspect of the domain. CASL employs scheduling criteria such as "Attempt to schedule observations as early as possible."

Knowledge Function: A class of operator that transmutes application data into numeric scores. The mappings are criterion-specific and explicitly stored in the knowledge base.

These concepts are described in detail in later sections. CASL is the focus of this report and experiments with CASL are described and results are presented.

2. Description of the HST

NASA's Hubble Space Telescope (HST) is an orbital observatory that was launched by the Space Shuttle *Discovery* in 1990 and successfully repaired in December 1993 by the *Endeavor* crew. The success of the "First Servicing Mission" should restore HST optics to original specifications and improve its ability to do high quality science.

The Space Telescope Science Institute (STScI) is responsible for managing the ground-based scientific operations of HST. Proposals (i.e., experimental programs) are submitted to the Institute and are processed by a series of software programs. The results are sets of spacecraft commands which (ideally) produce image data that is returned to the STScI for further processing and archiving. For more details about HST, see Hall 1982.

3. Conceptual Description of the HST Scheduling Process

The scheduling problem has been divided into discrete layers of logic. The layers can be briefly described below. Several of these layers will be discussed in detail in a later section. The following list is in order of increasing abstraction from the spacecraft.

- An instrument on the spacecraft is commanded to expose a photosensitive element to light.
- The commands are derived from a micro-scheduled calendar created by astronomers using the Science Planning and Scheduling System (SPSS).
- The Long Range Planning (LRP) system operates on one year time intervals. The time resolution of the LRP is currently one week. The set of observations assigned to one particular week of a completely scheduled LRP are communicated to SPSS.

The long range planning process has the following logical layers:

- At the lowest level is a constraint analysis system, called Micro-SPIKE, that provides information such as "When is the observation's target visible to HST?" and "Since observation B must be after observation A, when is it legal to observe B?"
- The next higher layer is the Constraint Satisfaction Problem (CSP) system which provides a workbench for searching for a feasible set of assignments of observations to time slots. The CSP employs Micro-SPIKE to answer *what if* questions such as the aforementioned.
- At the most abstract layer, CASL employs the CSP system and its utilities to create an LRP that satisfies both the physical constraints of the spacecraft and the practical realities of the planning scientists.

4. The Scheduling Sequence

In this section, the major steps in the scheduling of HST proposals is described. Later sections will discuss the CASL Long Range Planning strategy in detail.

4.1. Proposal Creation

An astronomer currently creates a proposal that takes, as its initial form, a text file containing definitions of *targets* (objects to be observed), *exposures* (the images to be taken), and *special requirements* (constraints on exposures). Syntax checking is done via distributed software tools. The proposal is sent to the STScI where it is submitted to an analysis and correction process called *proposal preparation*. This process creates *scheduling units* (SUs) which are collections of exposures organized by a complex set of rules. The role of SPIKE is to place SUs on a long range plan that spans many months.

The proposal preparation phase includes running portions of the SPIKE software that check important proposal aspects such as violation of HST pointing restrictions (e.g., "Allow no pointing that is within 50 degrees of the sun.") and schedulability (e.g., "The AFTER constraint causes the linked exposures to have no legal scheduling windows"). The principle goal of preparation is to provide planning personnel with a proposal that will schedule without constraint violations "in isolation."

4.2. Long Range Planning

When a large fraction of proposals have been prepared, the Long Range Plan (LRP) is created. The LRP spans approximately a year and consists of week-long time *segments*. The main difference between preparation and planning is that the proposals must, in the context of the LRP, compete for resources such as available *spacecraft time*.

4.2.1. The Constraint Based Scheduling System

The SPIKE system has a subsystem called *micro-spike* that is used to analyze absolute constraints ("Don't point at the moon") and relative constraints ("Schedule SU A after SU B"). This section discusses how it operates.

SPIKE represents scheduling information primarily using the *suitability function*. The suitability function is a means for representing scheduling constraints and preferences (Johnston, 1990). The approach provides a way to represent the concept of "goodness over time." Suitabilities (in this discussion) are in the real numbers and range from zero to one where zero means unsuitable and one means nominally suitable. The encoding of a suitability function is via a *piecewise constant function* (PCF) which is a list of time/value pairs. For example, if one determines that the sun is blocking the view of a target from the first segment of our plan up to, but not including, the fifth segment (when the target becomes visible), a PCF representing this would appear as follows: (1 0 5 1).

For an SU, absolute constraints are computed and represented as suitabilities. The following list enumerates some of these:

Solar Exclusion: Allow no point that is less than 50 degrees from the sun.

Orbital Viewing: Determine first whether there is available viewing time (i.e., that the target is not occulted by the earth) and if so how much. The suitability for a specific time period will be inversely proportional to the number of orbits required for the SU's component observations.

Micro-SPIKE supports relative constraints via a constraint propagation algorithm. Informally, let A and B be SUs. Let $abs(A)$ represent the combined absolute constraint suitabilities of A and $after(A, B)$ represent the constraint that B must be scheduled after A. Micro-SPIKE computes the effects of $abs(B)$ and $after(A, B)$ deriving a new suitability for

B that upon combination with $\text{abs}(B)$ produces a new suitability for B. Arbitrary relative constraint networks are supported and so the algorithm iterates until no further changes in suitability (for any SU) is detected. See Miller (1988) and Sponsler (1991) for a more complete description.

In the SPIKE domain scheduling is treated as a *constraint satisfaction problem*. Such problems are known to be NP-complete (Garey, 1979) and so the exhaustive traversal of the entire search space is not computationally tractable if the number of SUs is large.

4.2.2. CSP

Another SPIKE subsystem called the Constraint Satisfaction Problem (CSP) system can be considered an general purpose problem solving engine.

In the context of HST scheduling CSP provides a workbench for searching for feasible LRPs (fully committed with no constraint violations). The SPIKE CSP system performs the following:

- Communicates with the Micro-SPIKE software to obtain data about constraints (e.g., if an SU is planned in a specific time segment, how does this affect related SUs?).
- Supports a framework that records static *preference* values for each SU/segment pair. Preference is (currently) defined to be the integral of the suitability function for a discrete time segment and so collapses a complex description to a single value.
- Supports a mapping from SU/segment pair to a count of constraint violations (called *conflicts*). This count provides information that not only describes where constraints are violated but also how many violations have occurred.
- Provides the capability to *commit* SUs to time segments. A fully committed CSP represents a complete long range plan. The action of commitment causes Micro-SPIKE to be consulted such that other SUs (related by relative constraints) may acquire new conflicts in certain segments).
- Tracks resources for each time segment. When all resources for a segment are consumed, a conflict is logged for all other SUs thereby communicating the undesirability of that segment for further commitments.

Minton (1992) found that heuristic algorithms (including *max preference min conflicts*) could be used to solve CSP problems effectively. In the context of HST, by iteratively searching for SU/segment pairs that maximize suitability and minimize constraint violations, good schedules result.

4.2.3. Meta-Scheduling

The max preference algorithm has been effective in supporting HST scheduling. However, the STScI personnel who employ SPIKE to create LRPs requested that the software provide another (more abstract) layer that would support important constraints provided by the planners themselves. For example, the constraint to schedule SUs as early as possible does not have a physical basis. It is important, however, because the near-term portion of an LRP should be as fully packed, with respect to available spacecraft resources, as possible.

The request for such software by the planners catalyzed the design and implementation of the *Criterion AutoScheduler for Long range planning* (CASL) which is the subject of this report. CASL is described in detail in later sections.

The resource constraint levels for the LRP are set to greater than 100% in order to provide a larger pool of SUs for micro-scheduling. This oversubscription is intended to compensate for the fact that certain SPSS scheduling constraints are not encoded in SPIKE currently. If, therefore, certain commitments are incorrect, the larger pool provides SPSS with alternative SUs.

4.3. Micro-scheduling

The final step in proposal processing, at the STScI, is accomplished by SPSS. The SUs committed to one week time segment in the LRP are communicated to SPSS operators who micro-schedule them very precisely on a *calendar* (a data structure that represents a time line). Upon completion, this calendar is converted into a sequence of spacecraft commands.

5. The Multiple Criterion Network Expert System Engine

An expert system technology called the multiple criterion network system (MCN) has been developed. MCN is based on *functional knowledge representation* which has been applied previously in two other systems (Lucks, 1992 and Lucks, 1990).

Several terms are now defined informally. The *MCN score* is a numeric value in the set of real numbers ranging from zero to one. Conventionally, a zero score is interpreted as poor and a unit score as good.

CASL computes a score that describes how well an SU schedules in a specific time segment. If the score is zero, then the fit is poor.

The *MCN criterion* is domain-specific and deemed an important attribute of the problem being solved by the expert and so is required information concerning the decision making process.

For example, in CASL, scheduling an SU as early as possible is a criterion.

This technology provides system builders with the following capabilities:

- i. The ability to store expert knowledge in the form of knowledge functions which are domain-specific and which calculate numeric scores (real numbers in the range from 0 to 1).
- ii. The ability to explicitly declare mappings from raw scores to more refined scores. These knowledge mappings clarify the decision making calculations for knowledge engineers and users alike.
- iii. The ability to define the way scores are accumulated into a single score via an aggregation function.

These capabilities support trade-offs between competing criterion in an automated decision support system. The components of MCN are described in the following sections.

5.1. The Knowledge Functions

The knowledge function is a program which encodes expert knowledge about the domain. Each function either produces a numeric score or maps one score to another. There are four types of knowledge functions in the MCN model. They are described below.

5.1.1. Measurement Functions

The *measurement function* is the initial source of data in the MCN model. This function is the point where the application information about a specific criterion is accessed directly and converted to a measurement value. This value is not required to be a score and in fact may be quantitative or qualitative.

CASL's earliest criterion measurement function maps (for example) to a measurement value of 0.085 (i.e., the temporal offset of a time segment with respect to the entire schedule). Another criterion, *preference* (i.e., static goodness of fit) might produce a value of 50 (where the maximum is 100).

5.1.2. Intensity Functions

The *intensity function* is defined as a mapping between measurement value and the *intensity value* which is required to be from zero to one. This mapping is a normalization of criteria to a single scale and may be any arbitrary function. The intensity value conventionally is not interpreted as good or poor.

The earliest criterion measurement of 0.085 maps to an intensity of 0.915 (the mapping is simply the additive inverse). The preference intensity for 50 is 0.50.

5.1.3. Compatibility Functions

The *compatibility function* maps from intensity value to *compatibility value*. This mapping can be any function and must be specified by the knowledge engineer with expert guidance. There are two important attributes to this mapping:

1. This compatibility value conventionally uses the classic 0.0 = poor and 1.0 = good meaning.
2. The mapping also provides a way to adjust the relative power or importance of the criterion. An intensity value may be either tempered or amplified by this mapping to diminish or increase its importance with respect to other criteria.

In CASL, a simple weighting scheme is used for this mapping. The weight is applied as follows. Let w be the weight, m be the maximum intensity value, and v be the intensity value. Computation of compatibility is performed by this formula: $m - (w * (m - v))$.

The earliest criterion maps (by application of weight = 0.5) from intensity 0.915 to a compatibility value of 0.941. The preference value maps (with weight = 1.0) from intensity 0.5 to a compatibility value of 0.5.

5.1.4. Aggregation Function

The aggregation function maps from the compatibility value to the aggregation value which is the final score of the network. In the functional knowledge representation technology, an augmented decision tree (and/or graph) is used to compute the score from the individual criteria. The augmentation includes other functions that may be defined by the engineer. These are described in Lucks, 1992. MCN employs a single function (e.g., multiply) to perform the aggregation.

Aggregating all criteria for a specific SU/segment pair yields a value of 0.211. One can select the segment with highest score as the winning match where the SU can be scheduled.

5.2. Parallel Observations Matching System

The Space Telescope is capable of executing observations in parallel provided a set of restrictions are satisfied (e.g., the same instrument cannot be used in parallel with itself). Functional knowledge representation has been applied to the problem of matching pre-defined parallel scheduling units with standard SUs (see Lucks, 1992). A large pool of such parallel SUs must compete for scheduled SUs. This system, the Parallel Observations Matching System (POMS), is in operational use at STScI.

6. CASL Architecture

The CASL system employs MCN technology to direct the LRP generation. Specifically, the aggregation decision tree is replaced with a simple combining function (the multiply function, for example). The function of CASL is to create Long Range Plans for HST that:

- do not violate constraints
- maximize the suitability of all commitments
- satisfy the planning goals of the LRP planners

The CASL system requires that prepared proposals be loaded into a CSP Scheduling System which provides the workbench upon which the expert system operates. There are two main phases to the application of CASL. They are described below.

6.1. Prioritization Phase

The prioritization phase of CASL is responsible for analyzing each scheduling unit using MCN techniques to determine the order by which SUs are placed on the schedule. This ordering does not specify the time ordering on the resulting schedule. A subset of the criteria used are described below:

Absolute Time Windows (ABSOLUTE). Characterize an SU based on the amount of time that is available to it based on its absolute constraints.

Relative Timing Links (RELATIVE). Characterize an SU based on the number of relative constraints that are attached to it.

Proposal Completion (COMPLETION). Consider a proposal and its component SUs. The fraction of SUs that have been scheduled by SPSS is used to compute the completion

measurement function. The priority of an SU (with respect to this rule) is proportional to the fractional completion.

The ordering is done based on the aggregate score for each SU. The SU with the highest score (i.e., has the highest priority) will be placed on the plan first.

6.2. Autoscheduling Phase

The next phase of the CASL algorithm is the actual scheduling. An SU is selected from the sorted list and is analyzed with respect to all week-long time segments in the scheduling interval (which can span a year or more).

6.2.1. Time Segment Selection

Some of the criteria used to select a time segment for each SU are described below:

Min-Conflicts (CONFLICTS): This criterion will select for segments that have few or no conflicts.

Max-Preference (PREFERENCE): This criterion will select for segments that have the highest preference value.

Earliest-Segment (EARLIEST): This criterion will tend to place commitments earlier on the schedule if possible.

Minimize-Proposal-Spread (SPREAD): This criterion will tend to keep SUs from the same proposal together on the schedule.

Level-SU-Duration (DURATION): This criterion will tend to place SUs into segments that have less SU Duration resource consumed.

Prior-Commit (PRIOR): This criterion attempts to preserve pre-existing commitments. Periodically, the LRP must be regenerated due to feedback from the micro-schedule, changes in spacecraft status, etc. This criterion attempts to preserve old LRP state by biasing for selection of segments previously selected.

Group-Instruments (GROUP): This criterion attempts to commit SUs with the same science instrument (SI) into the same segment. It determines an SI frequency distribution of the instrument for the SU and this becomes the score. For example, if the SU under study uses the Faint Object Camera (FOC) and the distribution of FOC in the segment is 0.70 then that is the score (since it is a fairly good score the segment will be attractive for commitment).

Continuous Viewing Zone (CVZ): The measurement function for this criterion determines whether the SU is a candidate for the CVZ which is a period of time (usually a few days) when the target is not occulted by the earth. Scheduling SUs in the CVZ improves the efficiency of spacecraft operation.

Random-Choice (RANDOM): This criterion should be used only when the autoscheduler is to be executed several times in order to search for relatively better plans. It injects an element of chance into time segment selection thereby allowing schedules to differ. The associated criterion weight specifies the degree to which randomness is inserted.

Table 1. This table provides example data for one SU/segment pair. Only a subset of the criteria is displayed. The aggregate score for this set of criteria is: 0.211.

Criterion	Measurement	Intensity	Weight	Compatibility
CONFLICTS	0.000	1.000	1.000	1.000
PREFERENCE	50.000	0.500	1.000	0.500
SU DUR	0.422	0.578	0.600	0.747
EARLIEST	0.085	0.915	0.500	0.941
SPREAD	8.500	0.200	0.500	0.600

In Table 1, the EARLIEST criterion has a measurement value of 0.085 (i.e., the week under analysis has an offset that is 8.5% of the scheduling interval and therefore very early). Normalization by the intensity function produces an intensity value of 0.915 (which is good). The weight of 0.5 reduces the effect of the criterion producing a compatibility value of 0.941.

6.2.2. Commitment

The result of the time segment selection process is the selection of the week that has the highest aggregate score. The SU is then committed to this week. This action produces the following side effects:

- i. Constraint propagation is performed to determine the effect of the commitment on other SUs that are linked (via timing constraints) to the committed SU.
- ii. For such a linked SU, conflicts accumulate for any week that has become unsuitable based on constraint propagation.
- iii. The committed SU consumes some portion of available resource constraints.
- iv. SUs that are yet to be analyzed will be affected by the change in conflict counts and resources.

7. CASL Behavior

In the following sections, CASL experimental data are presented and described. An informal Meta-Scheduling Hypothesis for these experiments is as follows. Let *schedule quality* be defined as average preference for all commitments. Preference takes into account the physical and proposer constraints on the SU.

Sufficient flexibility exists in the placement of scheduling units such that a meta-scheduling agenda can be followed without sacrificing schedule average preference.

The scheduling units, criteria, and weights in these studies have been obtained from the operational database currently in use by HST planners.

7.1. The Cycle 4 Long Range Plan

The Cycle 4 Proposal period extends from January 1994 (following the First Servicing Mission) to June 1995. For this report, 994 scheduling units from 281 proposals were automatically scheduled into 78 week-long time segments using CASL. This dataset has the following attributes:

- It is not randomly generated and therefore cannot be considered a general purpose benchmark dataset. This dataset may not adequately exercise CASL.
- The dataset contains a 4944 SU to SU timing links which make scheduling difficult.
- The pool represents a subset of the complete set of Cycle 4 SUs because the proposals have not been completely prepared.
- The resource ceilings are very high (approximately 200%) compared to the actual available time. This is unrealistic but does not really effect the outcome of the experiments.

The descriptions below include unit tests on several criteria and tests done on the integrated criterion set.

7.2. Criterion Unit Tests

In each of the following tests, the *max preference* and *min conflicts* criteria were in effect and set to a unity weight. The Meta-Scheduling Hypothesis proposes that meta-scheduling criteria (such as *earliest*) can be applied without violating the preference requirement.

The data format of the following tables is as follows:

Spread: Let n be the number of proposals, last-su_i be the latest commitment date of all SUs in a proposal, and first-su_i be the first commitment date of the same proposal. Spread is

$$\text{defined as: } \frac{\sum_{i=1}^n \text{last-su}_i - \text{first-su}_i}{n}.$$

Spread SD: This is the standard deviation (in weeks) for proposal spread. This gives an indication of how each proposal varies from the sample mean spread.

Completion: This is the number of SUs committed divided by the total.

Offset: For each proposal a mean offset (segment of commitment/total segments) is determined. This value is the mean of all mean offsets and provides a measure of when the proposal was scheduled relative to the entire schedule. This attribute is directly affected by the *earliest* criterion.

SU Dur Mean: This is the mean amount of available SU Duration resource consumed for all weeks.

SU Dur SD: This is the standard deviation of the SU Dur Mean. A value of 0.0 would indicate perfectly level consumption. This attribute is directly affected by the *level su duration* criterion.

Pref: Let n be the number of SUs and $\pi(\text{SU}_i)$ be the percent of maximum preference for

$$\text{the } i\text{th SU. Pref is defined as } \frac{\sum_{i=1}^n \pi(\text{SU}_i)}{n}.$$

Orbits-Min: This is the number of spacecraft orbits consumed by the schedule minus the theoretical minimum number of orbits. An SU may consume a different number of orbits at different times of the year due to variance in target viewing times. The optimal value for this measure is zero. It should be noted that CASL does not explicitly operate on this attribute and so no attempt is made to minimize changes to it. Preference references this information implicitly along with many other aspects of target viewing. Note that the schedule minimum orbits is equal to 2432 and that an Orbits-Min value of 100 represents 4% difference from the optimal.

The proposal spread criterion was tested with weights of 0.0, 0.1, 0.5, and 1.0. There results are in Table 2.

Table 2. This table contains data obtained from unit tests of the *proposal spread* criterion which tends to keep the SUs from a given proposal together.

Attribute	Wt =0.0	0.1	0.5	1.0
Spread	14.4	11.2	8.0	5.7
Spread SD	18.8	16.0	13.3	10.4
Completion	0.71	0.71	0.71	0.71
Offset	0.86	0.87	0.87	0.87
SU Dur Mean	0.46	0.45	0.45	0.45
SU Dur SD	0.25	0.22	0.23	0.23
Pref	0.99	0.99	0.98	0.95
Orbits-Min	91	99	104	104

The average Spread was decreased from 14.4 to 5.7 weeks (a 60% change) and the Spread SD decreased by 45% while the Pref value changed by only 4% and the Orbits-Min value change by 14%. These data indicate that this criterion can cause an improvement to a schedule without disrupting the crucial Pref value.

A second criterion, Level SU Duration, was analyzed in the same manner and the results are recorded in Table 3.

Table 3. These data were obtained from unit tests of the *Level SU Duration* criterion.

Attribute	Wt =0.0	0.1	0.5	1.0
Spread	14.4	16.2	16.5	17.2
Spread SD	18.8	18.8	19.0	19.4
Completion	0.71	0.71	0.71	0.71
Offset	0.86	0.89	0.88	0.88
SU Dur Mean	0.46	0.46	0.46	0.46
SU Dur SD	0.25	0.11	0.09	0.08
Pref	0.99	1.0	0.99	0.98
Orbits-Min	91	102	105	109

The SU Dur SD decreased from 0.25 to 0.08 (a 68% change) while the Pref value changed by only 1% and the Orbits-Min value change by 20%. These data indicate that this criterion can cause measurable changes to a schedule without disrupting the crucial Pref value.

The Earliest Criterion

The Earliest criterion was tested in a context that contained only the preference and conflicts criteria and the results are illustrated in Figure 1.

The behavior of the criterion is apparent in that 100% of the SU Duration resource is consumed in the early segments (e.g., 94.031) and decreases to 0% in the final segments (e.g., 95.065).

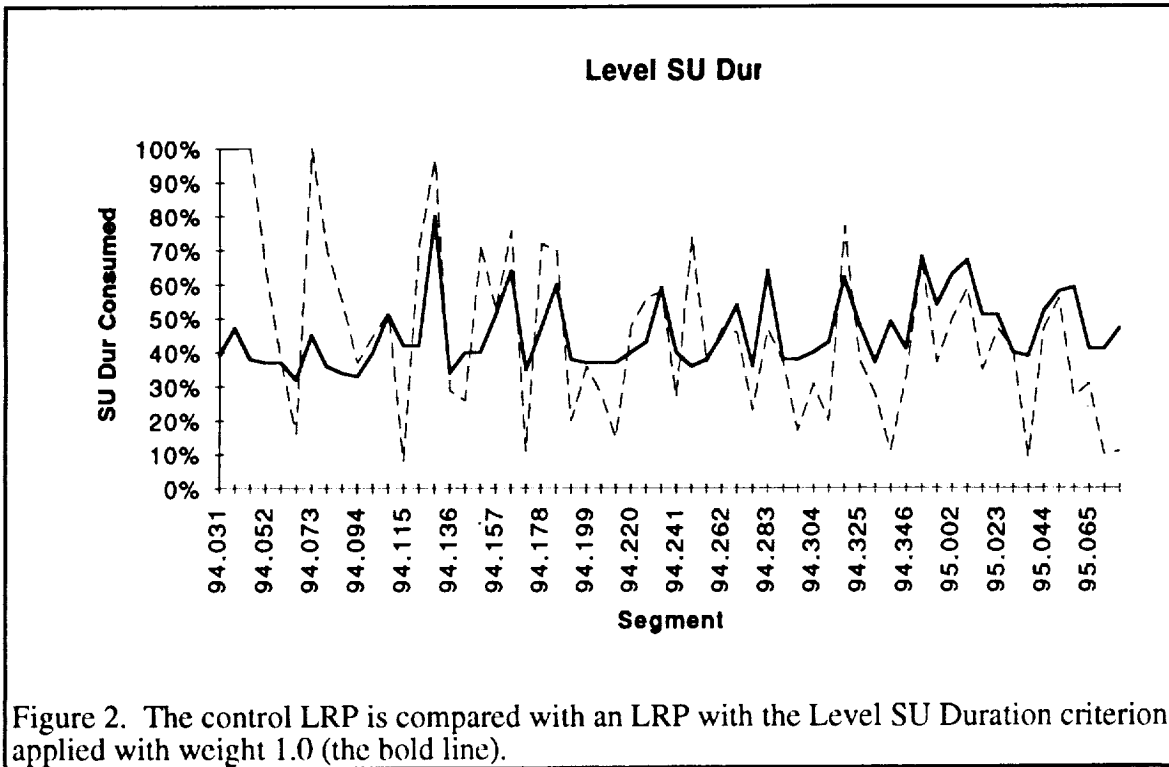
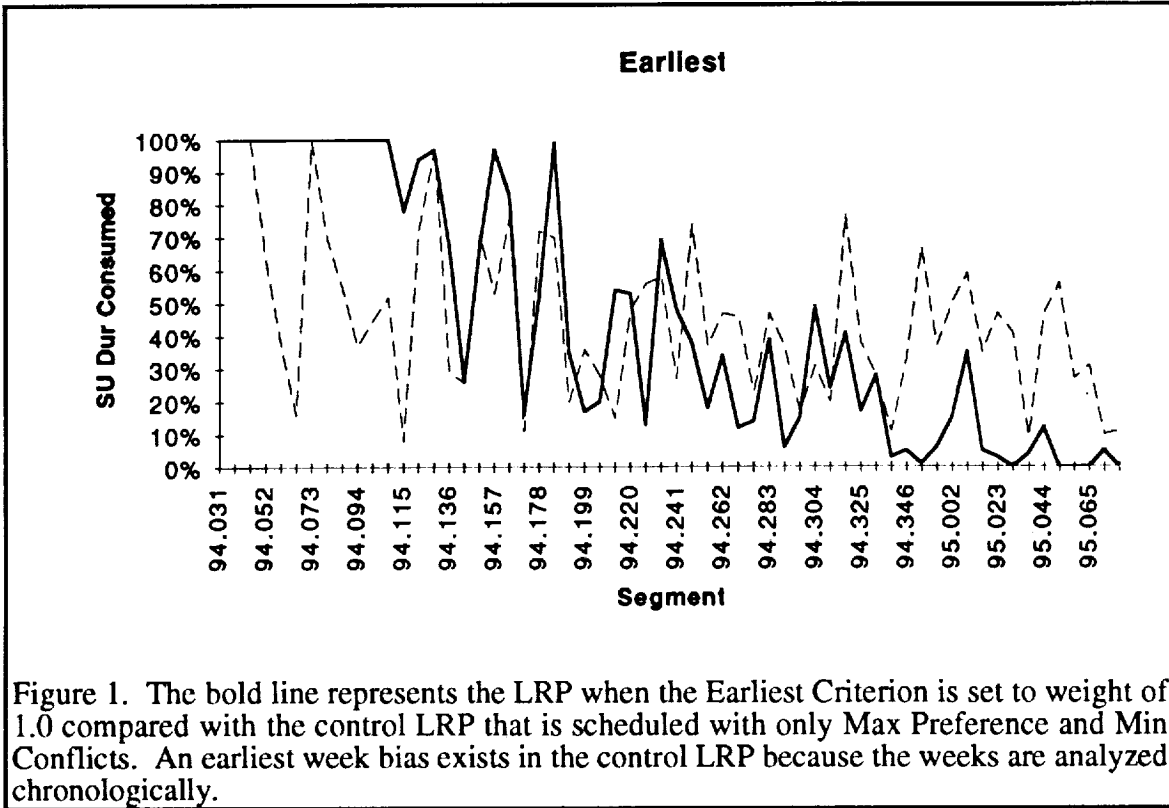
The Level SU Duration Criterion

The effect of the Level SU Duration criterion on the LRP can be seen in Figure 2 below.

In Figure 2, the peaks and valleys for SU Duration are tempered by the criterion's effect. The minimum and maximum values for the control LRP are 9% and 100%. The minimum and maximum for the criterion are 33% and 80%. It should be noted that if the preference and conflict criteria were inactivated, the bold line in Figure 2 would be flat.

7.3. Integrated Tests

In the integrated tests, a subset (Preference, Conflicts, Earliest, Spread, and Level SU Duration) of the criteria were activated and given the weights used operationally at STScI. Figure 3 illustrates some of the calculations used for time segment selection for one scheduling unit. Since no conflicted commitments are permitted, this criterion is omitted from further discussion.



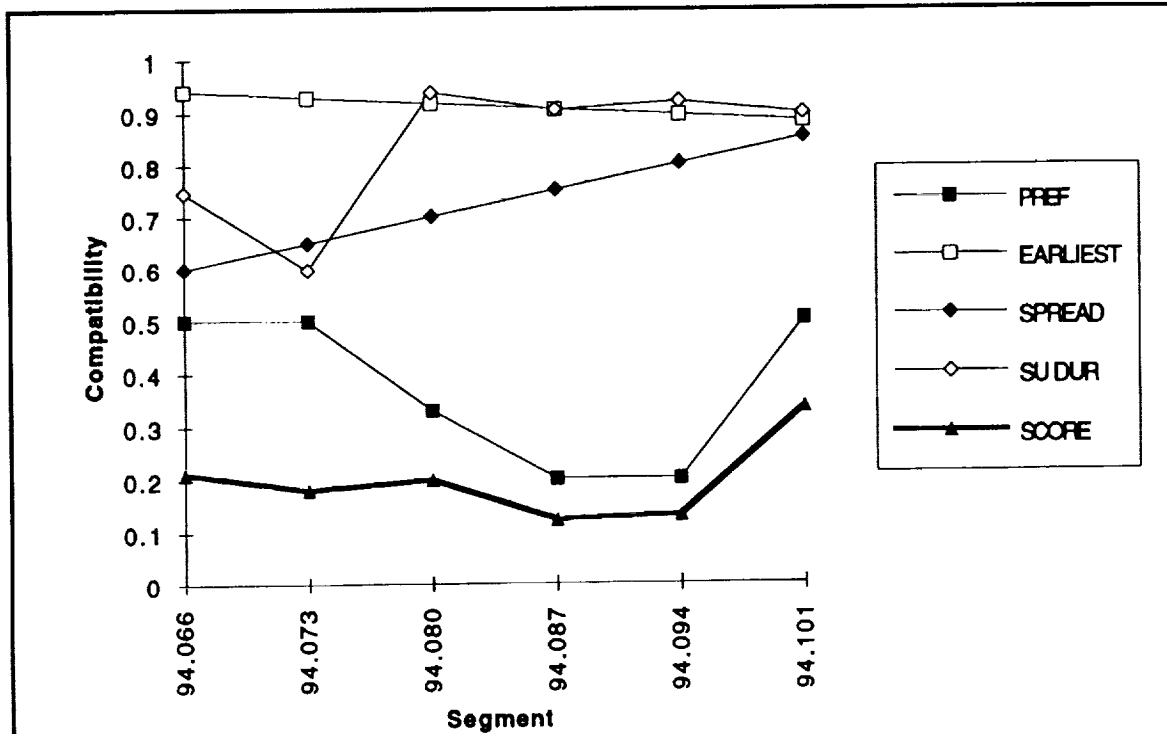


Figure 3. Four criterion compatibility values and the aggregate score (the product) for one scheduling unit are plotted as a function of time segment.

Figure 3 shows how the dominant Preference criterion has its highest value (0.5) in several different weeks. The score reaches its most favorable at 94.101 when the secondary criteria (Earliest, Spread, Level SU Dur) each contributes positively. These criteria can be considered tie-breakers and it is the presence of several weeks that have equally high Preference that allows the meta-scheduling criteria to take effect.

8. Discussion

CASL has demonstrated usefulness in generated long range plans for HST and the tests described in this report indicate a good capability in tailoring a plan according to specific high-level goals. Several important issues are discussed here:

1. Support for the Meta-Scheduling Hypothesis exists in the tests described. A mechanism has been implemented that can improve important scheduling criteria (earliest, su duration, etc.) while producing minimal changes to the average preference of SU placement.
2. CASL currently has a Random Criterion which can introduce non-determinism into the algorithm. Running the algorithm with this criterion activated will generate different results possibly finding better overall solutions. Traub (1994) states that "with a random selection of points, the computation complexity is at most on the order of the reciprocal of the square of the error threshold ($1/\epsilon^2$)." Testing the usefulness of this criterion is planned.

3. Complex aggregation function is missing. The aggregation of compatibility values by a single function was adequate for CASL. However, a decision tree is a more powerful means of exactly specifying how scores combine.
4. The MCN system is a shell and therefore can be applied to other applications. Currently, there are a number of potential candidates for such application including the intelligent combination of constraint PCFs.

Acknowledgements

The authors would like to thank the following persons for their ideas concerning this work: Mark Giuliano, Mark Johnston, Anthony Krueger, Michael Lucks, Melissa McGrath, Glenn Miller, Mary-Louise Shea.

References

- Garey, M., and Johnson, D. (1979). *Computers and Intractability*. New York: W. H. Freeman and Co.
- Hall, D., ed. (1982), *The Space Telescope Observatory*, NASA CP2244.
- M. Lucks, Detecting Opportunities for Parallel Observations on the Hubble Space Telescope, in *Proceedings of the 1992 Goddard Conference on Space Applications of Artificial Intelligence*, NASA Goddard Space Flight Center, Greenbelt, MD, pp. 22-24. Also in *Telematics and Informatics*, 9, 3-4, November/December 1992.
- M. Lucks, and I. Gladwell, "Automated Selection of Mathematical Software," *ACM Transactions on Mathematical Software* 18:1, March 1992, pp. 11-34.
- Miller, G., Johnston, M., Vick, S., Sponsler, J., and Lindenmayer, K. (1988). Knowledge Based Tools for Hubble Space Telescope Planning and Scheduling: Constraints and Strategies, in *Proc. 1988 Goddard Conference on Space Applications of Artificial Intelligence*; reprinted in *Telematics and Informatics* 5, p. 197 (1988).
- Minton, S., Mark D. Johnston, Andrew B. Philips, Philip Laird (1992). Minimizing conflicts: a heuristic repair method for constraint satisfaction and scheduling problems. *Artificial Intelligence* 58 (1992), pp. 161-205.
- Johnston, M. (1990). SPIKE: AI Scheduling for NASA's Hubble Space Telescope. *Proc of the Sixth IEEE Conference on Artificial Intelligence Applications* (March, 1990).
- Sponsler, Jeffrey L., Mark D. Johnston, Glenn Miller, Anthony Krueger, Michael Lucks, Mark Giuliano (1991). An AI Scheduling Environment for the Hubble Space Telescope. *Proc AIAA Computing in Aerospace* 8, Baltimore, MD (October, 1991), pp. 14-24.
- Traub J. and H. Wozniakowski. Breaking Intractability. *Scientific American* (January, 1994), pp. 102 - 107.

