Towards a Heuristic for Scheduling the James Webb Space Telescope

Mark E. Giuliano, Reiko Rager and Nazma Ferdous

Space Telescope Science Institute
3700 San Martin Drive, Baltimore, MD 21218
{giuliano, rager, ferdous}@stsci.edu

Abstract

Scheduling the James Webb Space Telescope (JWST) requires modeling and minimizing the buildup of angular momentum of the spacecraft. Accounting for momentum management is expected to be a major driver of spacecraft efficiency and lifetime. We develop a heuristic that reduces momentum buildup and show how the heuristic can be used in a simulation of the operations concept for JWST scheduling. Through experimental data we show how the heuristic improves momentum usage while maintaining high spacecraft efficiency on a representative science mission for JWST. Finally, we explore how the momentum heuristic closely mirrors the strategies required to efficiently schedule the Hubble Space Telescope.

Introduction

The James Webb Space Telescope (JWST) is a large, infrared-optimized space telescope, designed to find the first galaxies that formed after the Big Bang. Components of the mission are under construction and launch is planned for 2013. JWST requires a sun shield about the size of a tennis court to protect its science instruments from overheating. Solar radiation pressure on the sunshield causes angular momentum to accumulate in the spacecraft's reaction wheel assemblies. The wheels have a limited capacity to store momentum, and stored momentum must be dumped using spacecraft thrusters. The resulting use of non-renewable fuel to fire the thrusters makes momentum management a potential limiting factor in the lifetime of the mission. As the momentum accumulated by an observation varies over time, momentum management is expected to be a major constraint driving the efficiency of JWST scheduling. The JWST momentum resource constraint has several interesting features:

- The model is three dimensional.
- Resource consumption for an observation varies over time in non-linear manner.
- Resource consumption is additive in nature. Scheduling an observation at a time can either add or subtract from the overall resource consumption.
- Momentum provides both a hard constraint due to a limited capacity, and a preference to consume as little resource as possible.

These features are different from the types of resources covered in the planning and scheduling literature (Laborie 2003; Policella *et al.* 2004) where activities consume and release a constant capacity. In particular, the non-linearity of the domain prevents us from employing techniques commonly used to handle resource constraints. A previous study on JWST scheduling (Rager and Giuliano 2006) shows that simple heuristics preferring observations to schedule at a time that minimizes momentum change are not sufficient to create efficient schedules.

In this paper, we develop a scheduling heuristic that allows a multi-objective scheduler (Johnston 2006) to reduce momentum buildup without sacrificing spacecraft efficiency. The heuristic works by providing a mix of scheduling candidates that allows the momentum usage of one observation to be balanced by the momentum of other observations. We show how the balancing heuristic can be integrated with the expected operations concept for JWST scheduling. Using test observations, we present experiments showing that the strategy reduces momentum buildup while maintaining high spacecraft utilization. Finally, we explore how the momentum balancing heuristic closely mirrors strategies required to efficiently schedule the Hubble Space Telescope.

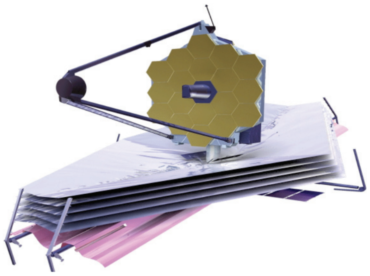

Figure 1: James Webb Space Telescope

JWST Mission Operations

JWST will have infrared sensitive detectors and a 6.5-meter primary mirror designed to look through dust clouds to see the earliest formation of stars and planets. The telescope will have a lifetime of 5 to 10 years and will be placed in an L2 orbit, 1.5 million km from Earth.

JWST will provide time to general observers through a time allocation board. Approved observers will prepare their programs using an automated tool. Programs will be submitted to the JWST Science Operations Center (SOC) and will be scheduled by SOC staff using a two phase scheduling process similar to the process used for the Hubble Space Telescope (Giuliano 1998). In the first phase, a long range plan assigns observations to overlapping least commitment plan windows that are nominally 60 days long. Plan windows are a subset of an observation's schedulable windows and represent a best effort commitment to schedule within the window. In the second phase, successive short term schedules are created for 22 day upload periods. The short-term scheduler uses plan windows to drive the creation of efficient telescope schedules. This two phase process allows a separation of concerns in the scheduling process: Plan windows globally balance resources, are stable with respect to schedule changes, and provide observers with a time window so they can plan their data reduction activities. Short term schedules provide efficient fine grained schedules to the telescope, handle slews between observations, and provide schedules robust to execution failure.

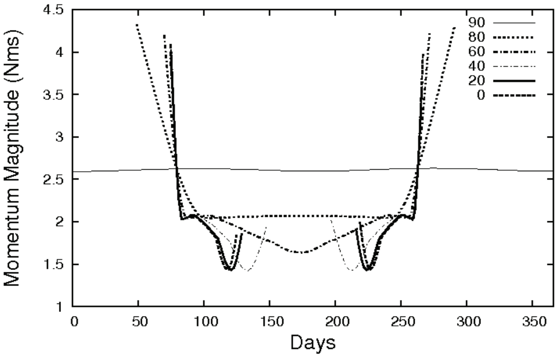

Figure 2: Momentum magnitude at different times of the year for 1 day long observations with longitude = 0 at varying latitudes. Target positions at negative latitudes produce the same momentum magnitude profile with different vector directions. Target positions at different longitudes have the same target visibility windows and momentum profile shifted to different times of the year.

JWST Scheduling Constraints

A scheduling system for JWST has to satisfy several types of constraints on observations. First, an observation has to obey all the requirements specified by the user. These requirements include the ability to specify time windows for observations, to link observations via timing constraints (e.g. observation$_i$ after observation$_j$ by 10 to 20 days), and to link observations via roll constraints. Second, an astronomical target can be observed by JWST only at certain times of the year determined by the location of JWST relative to the sun and the target. We call such time intervals *visibility windows*. The position of the target being observed defines the visibility windows. Ecliptic poles are visible throughout the year, while a target on the ecliptic equator (i.e. on the same plane as the Earth's orbit) has two visibility windows of about 49 days each. Third, schedules must satisfy the limit on momentum accumulation. Details on the momentum constraint are presented below.

Momentum limit constraint

JWST requires a large solar shield to prevent sun light from contaminating the science detectors and to protect the detectors from overheating. Solar pressure on the shield creates angular momentum force that is absorbed into the spacecraft's reaction wheel assemblies. The wheels have a finite capacity to absorb momentum buildup as measured in Newton meter seconds (Nms). The current assumption is that stored momentum will be dumped every 22 days during regularly scheduled station keeping activities. Momentum buildup during a 22-day period over a 24 Nms limit will require an extra momentum dump. As momentum dumps require burning scarce fuel, too many extra dumps will shorten the lifespan of the telescope. JWST momentum presents us with a resource constraint with the following features:

- **It is time-dependent**: Momentum usage depends on the spacecraft position. As the spacecraft position changes over time, the same observation will have different momentum accumulation depending on when the observation is executed. The amount of variation in momentum usage is latitude dependent. Targets with lower latitude have a higher variation in momentum usage (see Figure 2). The direction of momentum buildup also shifts with time.
- **It is multi-dimensional**: Momentum change can be expressed in a 3-D vector. The magnitude of momentum can be calculated by calculating the length of the vector.
- **It is additive**: After a momentum dump, momentum starts with a zero vector. Scheduling adds the momentum vector for the observation at the current time to the current momentum state. Scheduling an observation at a particular time can either reduce or add to the total momentum buildup (i.e. increase or decrease the momentum vector length).
- **It is both a hard and soft constraint:** Momentum must be dumped if a buildup of above 24 Nms occurs before regularly scheduled

momentum dumps. It is also preferable to minimize the amount of momentum dumped during the regularly scheduled dumps.

Schedule qualities

The JWST schedule qualities we desire are the following:

1. Low momentum accumulation.

The current operational plan is to dump momentum every 22 days during station keeping maintenance. The goal for the scheduler is to have no or very few 22-day periods that require additional momentum dumps. There are approximately sixteen 22-day periods in a year. It is acceptable for a few (< 10%) of the schedules to require additional momentum dumps. These schedules could either be manually adjusted to remove the extra dumps or the additional momentum dump and resultant use of fuel could be absorbed. In addition to the 24 Nms momentum limit, it is preferable to lower the amount of momentum dumped during scheduled station keeping maintenance as that reduces the amount of non-renewable fuel to be used.

2. Minimum gaps.

The JWST contract mandates that scheduling will result in schedules with less than 2.5% gaps. The input set of 1.2 years worth of observations provides 20% oversubscription to fill gaps for the 1 year schedule. We expect this level of oversubscription in operations and expect that operations will be able to utilize special gap filling observations. For the experiments we set a limit of achieving 1% or less gaps in our 22 day schedules. We use this lower limit because the input set does not contain many hard-to-schedule observations such as those with tight scheduling windows due to real-time contacts.

Momentum Heuristic

Previous Study

(Rager and Giuliano 2006) explored JWST momentum management in 2 ways; by trying out various general heuristics in scheduling and by analyzing the characteristics of the momentum resource in the domain. First, several techniques for incorporating momentum in scheduling were examined. Among the various value selection heuristics without a hard momentum constraint, the one that prefers the value that minimizes the observation's momentum contribution fared the best. When a hard resource constraint for momentum is introduced to scheduling, however, a value selection with no momentum heuristic did better than the minimum momentum heuristic. Overall, a hard momentum constraint without a momentum based value selection produced the best result in terms of total momentum buildup and total gaps in the schedule. The best schedule had an average of 19.99 Nms momentum buildup per bin with 5.7% schedule gaps. Second, an analytic experiment was run that simulated what kind of momentum management is possible in the domain if the system can select any observable target in the sky at any time. The finding was that momentum buildup can be reduced within a 22 day bin by careful selection of targets. However, these observations need to be scheduled near the edges of their visibility windows where their momentum magnitude is high.

Momentum Balancing Heuristic

The new momentum heuristic was motivated by the analytic portion of the previous study that found that for any time period, there exist schedulable observations that have offsetting momentum buildup. Expanding on this idea, we defined the **find_reducers** procedure as given in Figure 3. For each 22 day resource bin, sum the total momentum buildup for all of the observations that are schedulable in the bin. This yields a large momentum buildup (the magnitude ranges from 128 to 445 Nms) indicating that the momentum contributions for the schedulable activities in a bin do not naturally offset each other. Most of the schedulable activities in a bin build towards the same momentum vector direction. We next partition the schedulable observations in a bin into two sets. Those that add to the total buildup (*aka* **adders**), and those that subtract from the total buildup (*aka* **reducers**).

```
Procedure find_reducers (bin, observations)
bin_total_mom ← ∑compute_momentum(ob,bin)
Adders, Reducers ←{}
for each ob ∈ observations
    mom_offset(ob,bin) ← vector_magnitude(bin_total_mom + obs_mom(ob,bin)) - vector_magnitude (bin_total_mom)
    if mom_offset(ob,bin) < 0 then Reducers ← Reducers ∪ {ob} else Adders ← Adders ∪ { ob }
return Reducers, Adders
```

Figure 3: Algorithm to find the reducers for a bin.

The *reducers* are obviously a smaller set than the *adders* (since the *adders* build towards the total sum). Table 1 gives the distribution of *adders* and *reducers* in the first 14 bins for the year to be scheduled. The table shows that approximately 10% of the schedulable observations in a bin are *reducers* and that *reducers* can decrease the total momentum buildup by 10-25%. Since momentum usage changes over time, observations are *reducers* during some portions of the year and *adders* at other times. Figure 4 shows how the distribution of *adders* and *reducers* move over the sky between the 1st and 13th resource bin.

The heuristic is to ensure that as many as possible *reducer* observations are scheduled in each 22 day momentum bin. This provides opportunities for the momentum of one observation to offset or balance the momentum from other observations. We explain how the heuristic is incorporated in long range planning and in short term scheduling in a later section.

bin	Bin Momentum sum	# Adders	# Reducers	Reduced Momentum
1	174.47	2597	292	36.01
2	190.72	2609	280	29.18
3	128.01	2702	187	19.79
4	150.36	2531	358	40.49
5	398.91	2609	280	58.62
6	445.32	2640	249	25.72
7	329.44	2629	260	30.03
8	278.28	2683	206	25.35
9	240.16	2496	393	17.65
10	184.94	2662	227	24.98
11	350.09	2620	269	39.32
12	262.7	2708	181	27.95
13	285.4	2619	270	60.55
14	243.29	2574	315	42.04

Table 1: Numbers of adders and reducers in each bin.

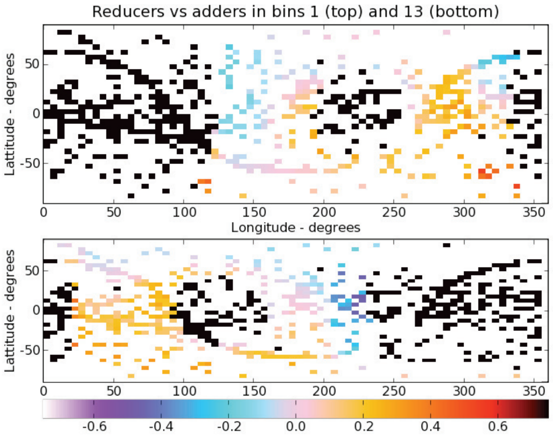

Figure 4: Distribution of momentum reducing and adding observations over the sky for resource bins 1 and 13. White space indicates target positions where we have no data, black indicates observations that are not schedulable in the bin, blues are reducers, and oranges are adders. When printed in black and white, a lighter gray indicates greater momentum reduction.

Experimental Setup

The JWST project has created a Science Operations Design Reference Mission (SODRM), which is a set of observations that closely match the expected mission duration, target distribution, instrument configuration, and constraint selection. It contains the specifications for both astronomical observations as well as calibration observations. The whole SODRM amounts to approximately 1.64 years of observations, including time for slews and other support activities. To allow us to compare with the previous study we use the same subset of the SODRM, totaling 1.2 years worth of observations, as input to this study. It consists of 2907 observations, including 1822 observations that are linked to at least one other observation. Observation duration varies from 70 minutes to 12 days with a median of 2.08 hours.

Scheduling constraints for observations in the SODRM were calculated using the JWST Mission Simulator (JMS). For each observation, JMS calculates the duration (= exposure time + support activity time + slew time), its visibility windows, and its momentum profile at every 3.65 days. JMS also passes user specified scheduling constraints such as links and between windows to SPIKE.

SPIKE CSP Scheduler

The SPIKE system is used in the evaluation. SPIKE (Johnston and Miller, 1994) is a planning and scheduling tool kit that was created for use on the Hubble Space Telescope and is currently being used for multiple orbital and ground based astronomical missions including FUSE (Calvani 2004), Chandra, Subaru (Sasaki 2000), and Spitzer (Kramer 2000).

The version of SPIKE used for the experiments treats scheduling as a Constraint Satisfaction Problem (CSP). Each observation is represented by a CSP variable. Each variable has a finite domain of discrete values that represent the times that the observation can schedule given its absolute constraints. The system counts conflicts on domain values during scheduling. Conflicts for a value can occur when: Another observation is scheduled in that time slot; The value does not obey a timing link from another observation; Scheduling in that time slot results in a momentum resource violation.

CSP SPIKE has several built-in scheduling strategies and provides templates for creating new strategies. The system supports iterative repair search algorithms. The scheduler first makes an initial guess that assigns a start time to every observation, possibly assigning observations to conflicting times. In the repair stage, SPIKE tries to reduce the number of conflicts by re-assigning the start time of conflicted observations. At the end of the repair stage, SPIKE removes the assignments for observations with existing conflicts to produce a conflict-free schedule. A simple set of gap filling routines were designed for the experiments below.

Time in the scheduler is represented in fixed quanta. Both the duration of observations and their possible start and end times are expressed in this manner. Since the scheduler checks for overlaps and resource conflicts before making a commitment to a particular start time, a smaller quantum requires greater run-time memory and time. We have picked the quantum size of 0.025 days (= 36 minutes) for the long term scheduler and a quantum of size 0.005 days (= ~ 7 minutes) for the short term scheduler.

Experimental Scenario

Our experimental setup mimics the two phase process expected to be used in JWST operations. In the first phase, plan windows are assigned to each observation during long range planning so as to balance resource consumption. In the second phase, observations are scheduled to precise times in short term scheduling.

In operations, the long range planning tool will create least commitment plan windows for observations. For our experiments the long range planning process is approximated by first creating a long range schedule (i.e. assignment of activities to times) and then expanding the scheduled times to adjacent momentum bins. The observations in the long range plan for bin N consist of those observations scheduled in bin N, and those that could be scheduled in bin N but are scheduled in bins N-1, or N+1. Long range schedules are created using the techniques described in the previous study. Momentum is modeled as a constraint that produces conflicts if a 22 day resource bin violates the 24 Nms limit on momentum buildup. Observations with a total duration of 1.2 years are scheduled in a year and a half time period. The long range plan provides a pool of observations to schedule in each momentum bin. To allow for approximately 20% oversubscription, all observations not scheduled in the first year are put in a pool available to all bins.

A short term schedule is run for each 22 day momentum bin using the candidates determined in the long range plan. Time is quantized to about 7 minutes. In JWST operations a limited number of 22 day bins will be allowed to have additional momentum dumps. To allow for these additional momentum dumps the momentum limit is set to 30 Nms during short term scheduling. The schedules for the first 14 bins in the year are run in succession. After the schedule for each bin is created, we record the observations that have start times assigned in the bin so they will not be considered in future bins.

The long range planner implements timing links between observations. The short term scheduler implements links within a single resource bin. The experiments do not handle links across multiple bins in short term scheduling.

Incorporating the Heuristic

The momentum balancing heuristic was incorporated into the guess and repair strategy used in the long range planner and the short term scheduler. In each case the heuristic was incorporated in a different manner. We first describe the short term scheduling algorithm and then describe how the algorithms are modified for the long term scheduler.

Pseudo code for the short term scheduler is given in Figures 3 and 5. The procedure **schedule_bin** is given an oversubscribed pool of candidate observations, *obs_in_bin*. The procedure uses randomization techniques by calling the lower level **schedule_obs** procedure 40 times to build the Pareto-optimal frontier with respect to minimizing schedule gaps and momentum buildup. The Pareto frontier consists of schedules that are not dominated by any other schedule for both criteria. After building the frontier, the best schedule is selected and instantiated, marking observations in the selected schedule as executed (so they are excluded from other bins). In the experiments presented, we select the schedule with the lowest momentum buildup that has less than 1% gaps. We examined alternative strategies for selecting the best solution on the frontier, such as a weighted average of the criteria, and did not find major impacts on the results.

During actual operations maintaining an explicit Pareto frontier would allow operations staff to select the best potential solution for scheduling. This allows the expertise of operations specialists to be effectively used to obtain schedules with features that are not or cannot be incorporated into criteria (Johnston 2006).

The procedure **schedule_obs** creates a schedule using the oversubscribed pool of observations for the bin. The procedure first selects a subset of the input pool, whose duration sums to the *bin_length*, using the procedure **select_initial_set**. Lines 8 and 9 of **schedule_obs** perform an initial guess on the selected subset of observations. Lines 10-12 of the algorithm attempt to repair any conflicts created in the initial guess. Lines 8-12 form the core of the algorithms described in (Johnston and Miller 1994). After removing conflicts the algorithm iteratively removes gaps from the schedule by first shifting scheduled observations to earlier times and then filling in gaps with unscheduled observations from the entire input set (*obs_in_bin*).

```
Procedure schedule_bin(obs_in_bin,bin,N = 40)
  1.  Pareto_set ← {}
  2.  For i = 1 to N do
  3.     sched ←schedule_obs(obs_in_bin,bin,i)
  4.     if Pareto_optimal(sched,Pareto_set)
            Pareto-add(sched,Pareto_set)
  5.  selected ← select_solution(Pareto_set)
  6.  mark_as_scheduled(selected)

Procedure schedule_obs(obs,bin,i)
  7.  to_sched = select_initial_set(obs,bin,i)
  8.  for each ob in randomize(to_sched)
  9.     schedule ob at time that minimizes conflicts
  10. while conflicts AND iterations <= length(to_sched)
  11. Unschedule random obs with max conflicts and
         reschedule to random time with min conflicts
  12. Remove all conflicts by successively unscheduling
      random obs   with max conflicts
  13. Iteratively compact and fill gaps in schedule using all obs.

Procedure select_initial_set(obs,bin,i)
  14. bin_length ← duration(bin)
  15. reducers,adders  ← find_reducers(bin,obs)
  16. Case (i mod 4)
      0: Return a random subset of obs whose total duration
         sums to bin_length.
      1: Return reducers plus a random subset of adders whose
         total duration sums to bin_length.
      2: Return reducers plus the adders with the lowest value
         of mom_offset(ob,bin) whose total duration sums to
         bin_length.
      3: Return reducers plus the adders with the lowest value
         of vector_magnitude(obs_mom(ob,bin)) whose total
         duration sums to bin_length.
```

Figure 5: Short-term scheduler algorithms.

The momentum heuristic is incorporated in the **select_initial_set** procedure which strongly biases the initial set to contain *reducers* and minimal *adders*. This procedure first finds the reducer observations using the **find_reducers** procedure (see Figure 3). Depending on the iteration index (i.e. i) the procedure incorporates the heuristic in one of 4 different ways. If i mod 4 = 0 then, no heuristic is used and the procedure returns a random subset of the observations that sums to the bin length. In the other cases the procedure returns the *reducers* plus a subset of the *adders* where the total duration sums to the bin length. The *adders* are selected randomly if i mod 4 = 2. In the other two cases the *adders* are selected based on minimizing quantities computed in **find_reducers**. The experiments were run using 40 iterations so each of the 4 strategies executed ten times. We did not track the performance of each individual strategy. For the non-heuristic runs, all 40 iterations return a random subset of the input whose duration sums to the bin length.

The long range planner incorporates the momentum heuristic in a different manner. The long range planner scheduling algorithm resembles the **schedule_obs** procedure with the following changes. The planner selects all observations to schedule over a 1.5 year horizon. The gap compacting and filling routines can move observations from the second year into the first year if possible. Finally, the heuristic is incorporated in a different manner:

- Use **find_reducers** to determine the *reducers* and *adders* for each bin using the set of observations schedulable in the bin.
- For each observation schedulable in a bin store the quantity *mom_offset(ob,bin)* from line 4 of the **find_reducers** algorithm (i.e how much it either adds to or subtracts from *bin_total_mom*).
- Observations are initially scheduled to times that minimize conflicts and that secondarily minimize the quantity *mom_offset(ob,bin)*.

When the long range plan is run without the momentum heuristic the initial guess selects a random time with minimum conflicts.

Scheduling Strategy Experiments

Four experiments were run varying whether or not a momentum balancing heuristic was used in Long Range Planning (LRP) and in Short Term Scheduling (STS). These runs are labeled as follows:

A. **Heuristic LRP and STS** – Short term schedules are generated with the momentum heuristic using a long range plan created with the momentum heuristic.

B. **Heuristic LRP** – Short term schedules are generated without the momentum heuristic using a long range plan created with the momentum heuristic.

C. **Heuristic STS** – Short term schedules are generated with the momentum heuristic using a long range plan created without the momentum heuristic.

D. **No Heuristics** – Short term schedules are generated without the momentum heuristic using a long range plan created without the momentum heuristic.

Bins 15 and 16 for the year were undersubscribed in all of the experiments so they are not presented in the results.

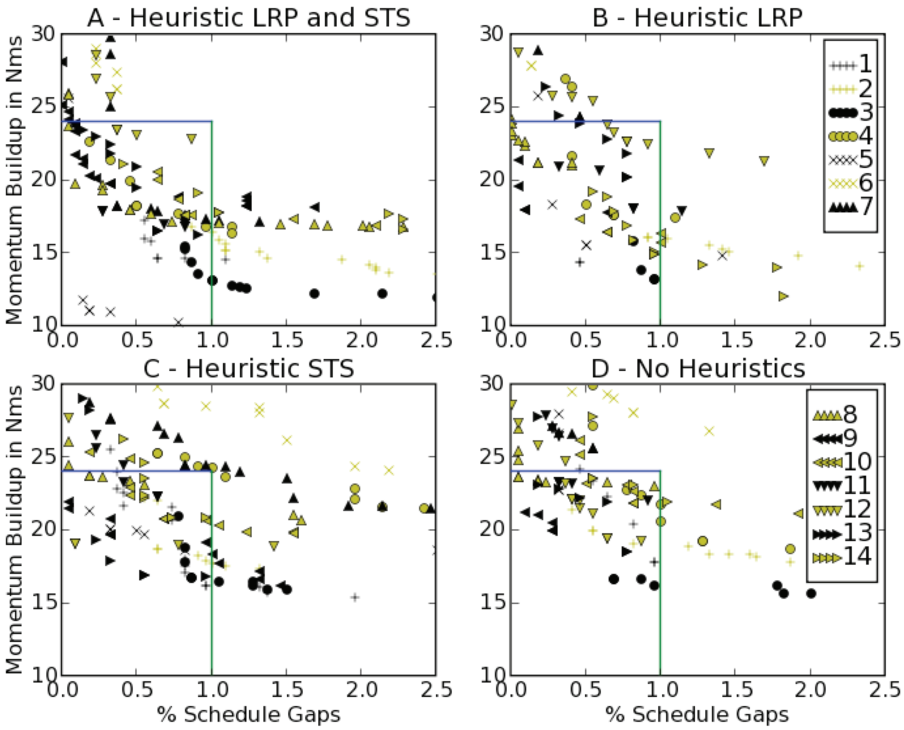

Figure 6: Pareto-optimal surfaces with respect to minimizing momentum buildup and gaps for the first fourteen resource bins in the year. Each symbol gives the surface for one bin. The legend in Figures 6B and 6D give the mapping from bin number to symbol. The horizontal line at 24 Nms and the vertical line at 1% schedule gaps bound the region of interest.

Figure 6 graphs the Pareto-optimal surfaces for the short term schedules generated for the first fourteen 22 day resource bins in the year. The points for each symbol (black +, yellow dot, …) represent the Pareto-optimal surface for one of the resource bins. The horizontal lines mark the 24 Nms limit and the vertical line the desired 1% gap level. The figures show a downward trend in momentum usage when the heuristic is applied in long range planning and short term scheduling. Figure 7 provides a summarized view of the best performance of each scheduling run where best is defined as the minimum momentum usage with less than 1% gaps. The graph plots the number of bins for each strategy where the buildup falls into a set of usage brackets. Using the momentum balancing heuristic in long range planning and short term scheduling decreases momentum buildup. The improvement is most clear when the heuristic is used in long range planning. Both runs of the long range plan with the heuristic have fewer bins that total above 20 Nms than the runs where no heuristic is used for creating the long range plan. Also, the two runs using the momentum balancing heuristic in the long range plan have 12 and 9 bins below 20 Nms while the two runs without a heuristic long range plan have 7 and 6 bins below 20 Nms. The momentum balancing heuristic in short term scheduling shows a clear benefit when the long range plan was created using the heuristic. The *heuristic LRP and STS* run has only two bins above 20 and one bin that violates the 24 Nms limit. The *heuristic LRP* run has 5 bins above 20 Nms and two that violate the 24 Nms limit. There is no clear benefit to using the momentum balancing heuristic during short term scheduling if the long range plan was created without the momentum balancing heuristic. Overall, the data shows that using the momentum balancing heuristic in both long range planning and short term scheduling decreases momentum usage while allowing for efficient schedules.

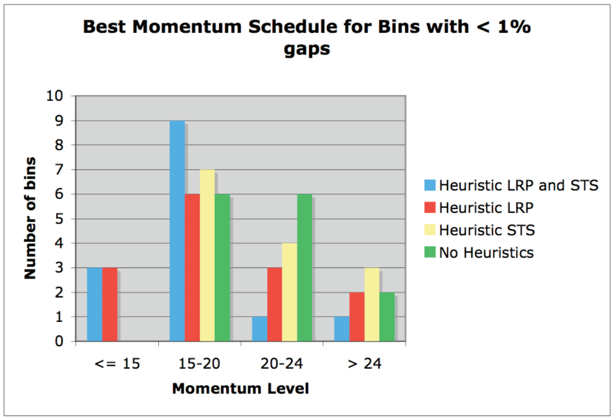

Figure 7: The number of resource bins where the momentum buildup falls in the given range for each scheduling strategy.

The results significantly improve upon those from the previous work. The previous study created year-long coarse-grained schedules using simple strategies for reducing momentum including a preference mechanism that biases towards scheduling where individual observation momentum buildup is minimized, and the use of a resource constraint to prevent momentum buildup over the legal limit. These strategies could keep momentum below the 24 Nms limit but resulted in unacceptable schedule gaps (>= 5.7%). The results presented in this paper use more complex heuristic based on finding the subset of the schedulable observations during a time period that reduce momentum from the general trend of momentum buildup for all schedulable activities in the time period. The heuristic was derived from detailed studies of the problem domain. The new heuristic results in less than 1% gaps and reduces the average momentum buildup in a bin by 12% to 17.4 Nms.

The experiments show the value of finding and exploiting properties of the application domain during scheduling.

Conclusion

We presented a momentum balancing heuristic for JWST observation scheduling based on balancing observation momentum buildup. We integrated the heuristic with a two phased scheduling operations concept that breaks scheduling into long range planning and short term scheduling. Schedule strategies were run on the JWST design reference mission and evaluated in terms of momentum usage and schedule gaps. The experimental data shows that:

- The best schedules are generated using the momentum balancing heuristic in both long range planning and short term scheduling.
- The momentum balancing heuristic has the most impact when used in long range planning.

The momentum balancing heuristic has similar features to strategies that are used operationally in scheduling the Hubble Space Telescope (Ferdous and Giuliano 2006). A major driver of Hubble scheduling efficiency is finding observations that can schedule during orbits that cross the South Atlantic Anomaly. The South Atlantic Anomaly is a region of high radiation that prevents the scheduling of science observations during a little more than half of the fifteen Hubble orbits in a day. However, South Atlantic Anomaly orbits can be used for science by overlapping the anomaly impacted portion an orbit with the portion of the orbit where the earth occults target visibility. An observation that achieves this overlapping is called a South Atlantic Anomaly hider. South Atlantic Anomaly *hiders* for the Hubble Space Telescope are parallel to JWST momentum *reducers*. Like the ability to reduce momentum, South Atlantic Anomaly hiding is a scarce resource that varies by target position and time of year. A Hubble observation may be a South Atlantic Anomaly *hider* for some time periods and a non-hider for other time periods. Likewise, a JWST observation may be a momentum *reducer* for some time bins and an *adder* for others. It is critical that momentum *reducers* and South Atlantic Anomaly *hiders* are scheduled to take advantage of their reducing and hiding capabilities. Empirically, we have found that the ability to schedule Hubble efficiently depends critically on scheduling strategies that find and effectively utilize observations that hide the South Atlantic Anomaly. Likewise, the experiments presented above show that strategies for finding and scheduling observations in times where they can reduce momentum enables the creation of efficient JWST schedules. Future work is needed to verify and strengthen the heuristic so it can be integrated into the operational JWST ground system.

Future Work

Design of the operational scheduling system for JWST is expected to start in earnest in 2008. A challenge will be to integrate the heuristics developed above into the operational system. This requires adding many features not included in the above experiments such as:

- The insertion of momentum dumps when momentum buildup violates the 24 Nms limit.
- Modeling slew durations between observations. These experiments assume a constant slew duration.
- Including time critical observations.
- Including links across short term schedules.
- JWST can roll up to 5 degrees from the nominal angle at a time. As the momentum is dependent on roll this provides an additional choice point for the planning process.

As some percentages of observations are projected to fail on board, a desired capability for the scheduler is to create a robust schedule that provides momentum stability in case of a failure and also provides backup observations in parallel to the main schedule.

We are currently investigating an extension to the momentum balancing heuristic that makes the heuristic dynamic. After a long term schedule has been created, we have attempted to manually improve schedules by swapping a momentum *reducer* into a bin. We have found that in some cases this raises the total momentum. The momentum balancing heuristic does a good enough job that the total buildup in a bin switches direction. We are investigating a technique that uses the current static momentum reducing heuristic until resource bins reach some level of buildup. After this point the heuristic prefers to schedule observations in bins where it reduces the momentum based on the observations already scheduled in the bin. Initial results from these experiments prove promising.

Acknowledgements

The authors would like to thank the ICAPS reviewers for their helpful comments, Bill Rumpl for providing JWST momentum data, and Rob Hawkins for his careful reading of the paper.

References

Calvani, H.M., Berman, A.F. Blair W.P. Caplinger, J.R. England, M.N. Roberts, B.A. Hawkins, R. Ferdous, N. and Krueger A.P., 2004. The evolution of the FUSE Spike Long Range Planning System. In *Proceedings of the 4th International Workshop on Planning and Scheduling for Space*. Darmstadt Germany, 25-31.

Giuliano, M. 1998. Achieving Stable Observing Schedules in an Unstable World. In *Astronomical Data Analysis Software and Systems VII*. 271-274.

Ferdous, N. and Giuliano, M. 2006. Validating Resource Usage in Least Commitment Planning. In *Proceedings of the 5th International Workshop on Planning and Scheduling for Space*. 82-91.

Johnston, M. and Miller, G. 1994. Spike: Intelligent Scheduling of Hubble Space Telescope Observations. In Zweben M. and Fox M. eds. *Intelligent Scheduling*, 391-422. Morgan-Kaufmann.

Johnston, M. 2006. Multi-Objective Scheduling for NASA's Future Deep Space Network Array. In *Proceedings of the 5th International Workshop on Planning for Space*. 27-35.

Kramer, L. 2000. Generating a Long Range Plan for a New Class of Astronomical Observatories. In *Proceedings of the 2nd NASA Workshop on Planning and Scheduling for Space*.

Laborie, P. 2003. Algorithms for Propagating Resource Constraints in AI Planning and Scheduling: Existing Approaches and New Results. *Artif. Intell.* 143(2):151--188.

Policella, N., Oddi, A., Smith S.F., and Cesta, A. Generating Robust Partial Order Schedules. In *Proc of CP 2004,*Lecture Notes on Computer Science (LNCS) Vol. 3258, pp. 496-511, M. Wallace (Ed.), Springer, 2004.

Rager, R. and Giuliano, M. 2006. Evaluating Scheduling Strategies for JWST Momentum Management. In *Proceedings of the 5th International Workshop on Planning and Scheduling for Space, 235-243.*

Sasaki, T., Kosugi, G., Kawai, J., Kusomoto, T., Koura, N., Kramer, L., Krueger, A., Miller, G. 2000. Observation scheduling scheme for the Subaru Telescope, In *Proceedings of Advanced Telescope and Instrumentation Control Software*, ed. Hilton and Lewis, *SPIE Vol 4009, 350-354.*