SSA Sensor Tasking Approach for Improved Orbit Determination Accuracies and More Efficient Use of Ground Assets

Alexander F. Herz
Orbit Logic Incorporated
Frank Stoner
Analytical Graphics, Inc.
Robert Hall
Analytical Graphics, Inc.
William Fisher
Optwise Corporation

CONFERENCE PAPER

Current SSA sensor tasking and scheduling is not centrally coordinated or optimized for either orbit determination quality or efficient use of sensor resources. By applying readily available capabilities for centrally generating de-conflicted schedules for all available sensors and determining optimal tasking times, both the quality of determined orbits (and thus situational awareness) and the efficient use of sensor resources may be measurably improved.

This paper provides an approach that is separated into two main sections. Part 1 focuses on the science of orbit determination based on tracking data and the approaches to tracking that result in improved orbit prediction quality (such as separating limited tracking passes in inertial space as much as possible). This part of the paper defines the goals for Part 2 of the paper which focuses on the details of an improved approach for sensor tasking and scheduling. Centralized tasking and scheduling of sensor tracking assignments eliminates conflicting tasking requests up front and coordinates tasking to achieve (as much as possible within the physics of the problem and limited resources) the tracking goals defined in Part I.

The effectivity of the proposed approach will be assessed based on improvements in the overall accuracy of the space catalog. The Sensor Tasking product within AGI's SSA Software Suite, which leverages existing commercial scheduling algorithms from Orbit Logic and Optwise, is used for computations and to generate schedules for the existing and improved approaches.

Part 1: Tracking Approaches for Improved Orbit Determination

The goal of sensor tasking/scheduling is to allocate collection resources in a manner that most significantly improves the ability to predict and detect events involving the resident space objects (RSOs). This allocation must take into consideration many factors that have bearing on the effectiveness of the observations requested such as observational capabilities of the individual sensors, historical performance of the system, current orbital accuracy for each of the RSOs, and orbital accuracy requirements for maintaining the catalog of space objects.

The current SSN tasking system accounts for these factors in a limited fashion, producing only prioritized lists of the RSOs assigned to each sensor for collection with each object's number of requested tracks [1]. The creation of a timeline for the collection of observations is left to the individual sensor sites. Some attempt is made to account for preferred viewing geometry, measurement type diversity, and spatial separation of measurements [2]. But it is difficult to ensure that those goals are achieved at the local sensor level and the scheduling is not centrally coordinated across the network,

Generally, improvements in orbit estimation accuracy can be realized by achieving diversity of measurement type, orbital separation of measurements, favorable observation geometry, and observational merit. Each of these topics is discussed in the sections that follow.

Diversity of measurement type:

Diversity in the types of measurements processed, provided the measurements are of comparable quality, generally helps to improve the accuracy of the orbit estimate since it increases the dimension of the measurement space and can improve observability of the estimated parameters including biases present in the collection system. SSN tracking of deep-space RSOs, objects with period greater than 225 minutes, illustrates the benefits of measurement diversity.

The SSN has limited options for collecting observations of deep-space RSOs. With the exception of space-based platforms, optical sensors are limited to collection during night-time periods with clear weather conditions. They can provide accurate angular and angular rate measurements but provide no measurement of range for the objects they observe. A small number of radar sites are available with the capacity to observe objects in the deep-space region. The radar observations provide accurate range and range rate measurements but less accurate angle measurements. Generally, orbit determination using only radar measurements will provide a more accurate orbit estimate than orbit determination using only optical measurements. But supplementing the angular data obtained from the optical sensors with range measurements obtained from radar sites yields the best solution.

Miller [3] demonstrates that a combination of optical tracking performed every other day with two supplemental radar tracks within the orbit determination period significantly improved orbit estimation accuracy over optical-only and radar-only estimations for three representative orbits: circular, semi-synchronous; highly eccentric, semi-synchronous; and geosynchronous.

Orbital separation of measurements:

Collecting observations when the RSO is at the same location on its orbit limits the accuracy of the estimation. Ground-based optical measurements of geosynchronous satellites can exhibit this characteristic. Since optical sensors detect light reflected by the RSO, the signal-to-noise ratio of the reflected light is usually highest when the solar phase angle is at its minimum. For geosynchronous objects, if the observations from a particular sensor routinely occurs at the minimum solar phase angle, the orbital geometry is similar for each collection and occurs when the geosynchronous satellite is roughly at the same location on its orbit. This scheme would be adequate for characterizing the period of the orbit, but limits observability of the eccentricity of the orbit.

Using a circular, semi-synchronous satellite, Miller [3] demonstrates that, as opposed to routinely collecting optical measurements at the time of minimum solar phase angle, varying the collection times around the time of minimum solar phase angle improves the orbit accuracy.

Favorable observation geometry:

The collection geometry has significant bearing on the probability of collection and the quality of the measurement. Atmospheric effects can cause the most degradation in signal when the observation occurs at low elevation angles. For optical sensors, a large solar phase angle reduces the signal strength at the observer. Constraints can be applied to the visibility computation to screen low-elevation or large solar phase angle passes of an RSO from tracking/scheduling consideration. But the maximum elevation or minimum phase angle varies from pass-to-pass, making some passes more favorable for collection than others. Similarly, these parameters can vary over a pass making some times within the pass more favorable for collection.

Observational merit:

Covariance-based network tasking algorithms provide a method to directly assess the relative merit of collection opportunities with respect to increased estimation accuracy. Hill, et.al. [4,5] discuss the use of various observational effectiveness metrics derived from the information gain matrix and covariance reduction matrix to more effectively schedule sensor collections.

The Kalman filter covariance estimate update accounts for the additional information gained from an observation at time t_k as in the following equation

$$\boldsymbol{P}_k(+) = [\boldsymbol{P}_k(-)^{-1} + \boldsymbol{H}_k^T \boldsymbol{R}_k^{-1} \boldsymbol{H}_k]^{-1}$$

where $\boldsymbol{P}_k(-)$ is the covariance estimate extrapolation based on the dynamical model, $\boldsymbol{P}_k(+)$ is the covariance estimate accounting for the measurement, $\boldsymbol{R}_k$ is the measurement error covariance matrix, and $\boldsymbol{H}_k$ is the observation matrix [6]. The information gain matrix, $\boldsymbol{G}_k$, is

$$\boldsymbol{G}_k = \boldsymbol{H}_k^T \boldsymbol{R}_k^{-1} \boldsymbol{H}_k$$

The change in the covariance due to the observation is purely a function of the observation matrix, $\boldsymbol{H}_k$, which maps the space of the state estimation to the measurement space and the uncertainty in the measurement, $\boldsymbol{R}_k$. It does not depend on the measurements themselves. Scalar metrics, like the determinant, trace, or various matrix norms, can be formed which indicate the "size" of the information gain matrix or its sub-matrices.These scalar metrics, like the geometrical parameters mentioned earlier, can be used to determine which passes are the more favorable collection opportunities as well as the most favorable collections times within a pass.

None of the approaches described above are currently drivers for the existing Air Force SSA sensor tasking process. This may be partly due to the fact that the current SSA tasking process does not include any scheduling, so it is not possible to try and drive specific relative tracking times or locations within this process. Until the SSA sensor tasking process also includes coordinated sensor network scheduling, none of the approaches described here for improved orbit determination accuracy can be implemented. The next section of this paper describes a tasking and scheduling approach that can be configured to achieve the tracking goals described above.

Part 2: Improved Sensor Tasking and Scheduling Approach

Current Approach:

The 614th AOC Space Surveillance Division responsible for SSN tasking uses an automated tasker [1]. The tasking algorithm used in this system is relatively simple and effective, but not very efficient. Tasking is performed for a 24-hour period for all SSN sensors. An object grouping system and priorities driven by orbit knowledge determine the number of tracks and number of sensors requested for each period. The tasking algorithm assigns tracks for each object in object priority order to specific sites with predicted visibility. A capacity (number of tracks per day) is assigned to each site and is used by the algorithm to limit tasking to each site. In addition, the current tasking software attempts to spread LEO tracking requests to sites at different latitudes in order to collect during different portions of the orbit.

It should be noted that the current automated tasker makes no attempt to schedule specific observations, leaving that to the individual sensor sites based on the number of requested tracks for each object assigned. The tasking algorithm can adjust for known outage periods, but this adjustment is limited to reducing the capacity of the site based on the duration of the outage period.

Limitations of Current Approach:

One of the limitations of the current tasking process is that tasking requests made to a particular site may conflict with each other. Because the current tasking system makes no attempt to schedule specific tracking passes, the tasker is not aware that two requested object tracks may be mutually exclusive due to simultaneous visibilities. This situation is currently resolved at the individual sites as part of their individual scheduling process. This in itself is not an issue, but these decisions are not coordinated with the rest of the sensor network. This means that some objects could be under-collected due to uncoordinated conflict resolution performed at the multiple tracking sites.

In addition, some sites may be tasked to track objects that they cannot actually observe since the current tasking system does not accurately model sensor attributes and capabilities or target attributes. This can also lead to the under collection of selected objects and to the rejection of specific tasking requests by the individual sites.

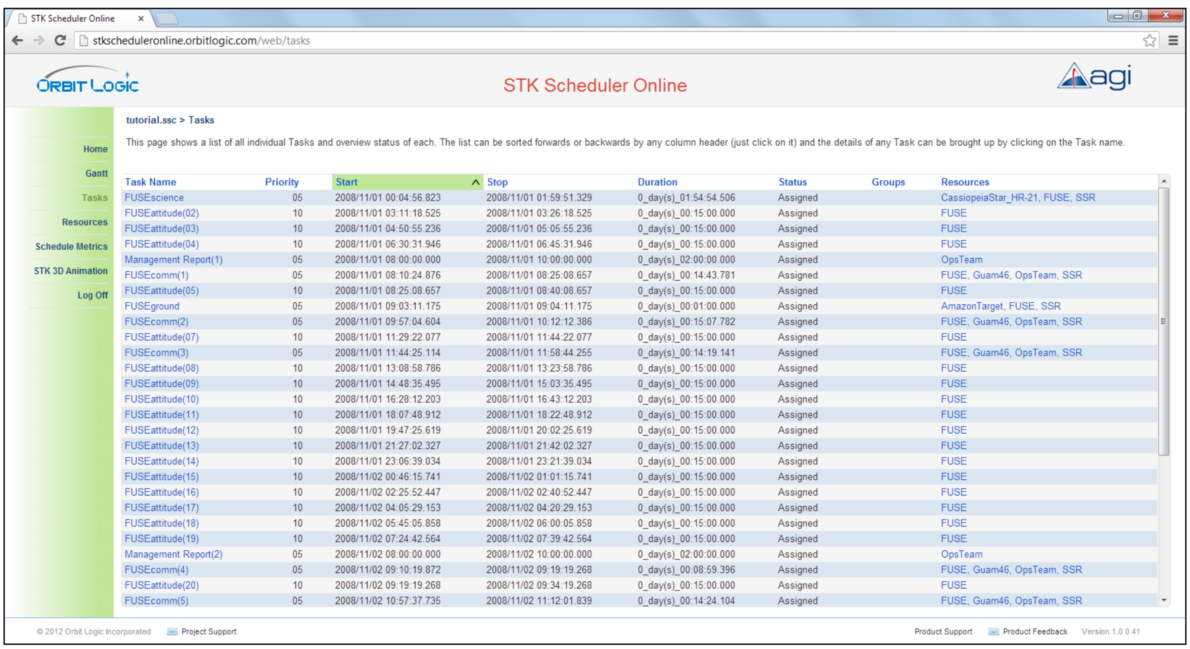
Also, the way that tracks are currently distributed based on latitude (for LEO objects) does not work well for objects with low inclination, leading to less than optimal inertial space distribution of tracking passes and lower quality orbit determination. In addition, there is no mechanism to achieve goals for diversity of measurement type, observation geometry, or observational merit as described in Part 1 of this paper.

The current SSA tasking process is performed every 24 hours, with previous period tracking results available as an input to the next period. Off-cadence planning is very limited to extreme situations and is performed manually because of the limited capabilities of the current system. The current process for generating 24-hour tasking takes several hours.

Improved Approach

Even ignoring the tracking goals for improved orbit determination accuracy specified in Part 1 of this paper, significant improvements can be made in the SSA Sensor Tasking process by applying readily available off-the-shelf modeling and planning software and centralizing sensor scheduling for the entire network of sensors.

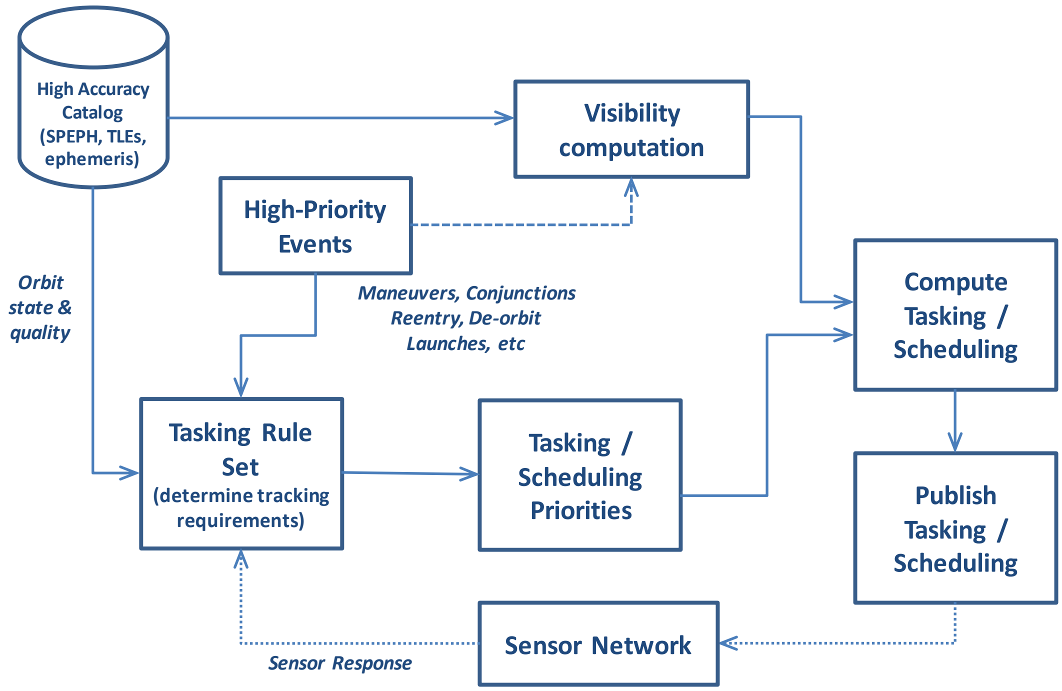
The Sensor Tasking portion of the SSA Software Suite from Orbit Logic and Analytical Graphics (AGI) has demonstrated the ability to perform 24-hours of SSA Sensor tasking and scheduling for all SSA sensors with a catalog of 10,000 space objects in less than 10 minutes on standard commercially available hardware. The generated schedules are conflict-free and consistent with input object tracking priorities, number of requested tracks per object, required sensor setup times, sensor capabilities and capacities, and object tracking constraints including per-object sensor priorities and limitations. The resultant schedules include start and end track times for each observation for each sensor in the network. These schedules can be easily parsed and disseminated to each sensor site through a variety of methods. STK Scheduler Online provides a means to disseminate the schedules via a secure website for any site that has internet access (see sample screenshot below).



Task Name	Priority	Start	Stop	Duration	Status	Groups	Resources
FUSEscience	05	2008/11/01 00:04:56.823	2008/11/01 01:59:51.329	0_day(s)_01:54:54.506	Assigned		CassiopeiaStar_HR-21, FUSE, SSR
FUSEattitude(02)	10	2008/11/01 03:11:18.525	2008/11/01 03:26:18.525	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEattitude(03)	10	2008/11/01 04:50:55.236	2008/11/01 05:05:55.236	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEattitude(04)	10	2008/11/01 06:30:31.946	2008/11/01 06:45:31.946	0_day(s)_00:15:00.000	Assigned		FUSE
Management Report(1)	05	2008/11/01 08:00:00.000	2008/11/01 10:00:00.000	0_day(s)_02:00:00.000	Assigned		OpsTeam
FUSEcomm(1)	05	2008/11/01 08:10:24.876	2008/11/01 08:25:08.657	0_day(s)_00:14:43.781	Assigned		FUSE, Guam46, OpsTeam, SSR
FUSEattitude(05)	10	2008/11/01 08:25:08.657	2008/11/01 08:40:08.657	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEground	05	2008/11/01 09:03:11.175	2008/11/01 09:04:11.175	0_day(s)_00:01:00.000	Assigned		AmazonTarget, FUSE, SSR
FUSEcomm(2)	05	2008/11/01 09:57:04.604	2008/11/01 10:12:12.386	0_day(s)_00:15:07.782	Assigned		FUSE, Guam46, OpsTeam, SSR
FUSEattitude(07)	10	2008/11/01 11:29:22.077	2008/11/01 11:44:22.077	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEcomm(3)	05	2008/11/01 11:44:25.114	2008/11/01 11:58:44.255	0_day(s)_00:14:19.141	Assigned		FUSE, Guam46, OpsTeam, SSR
FUSEattitude(08)	10	2008/11/01 13:08:58.786	2008/11/01 13:23:58.786	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEattitude(09)	10	2008/11/01 14:48:35.495	2008/11/01 15:03:35.495	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEattitude(10)	10	2008/11/01 16:28:12.203	2008/11/01 16:43:12.203	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEattitude(11)	10	2008/11/01 18:07:48.912	2008/11/01 18:22:48.912	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEattitude(12)	10	2008/11/01 19:47:25.619	2008/11/01 20:02:25.619	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEattitude(13)	10	2008/11/01 21:27:02.327	2008/11/01 21:42:02.327	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEattitude(14)	10	2008/11/01 23:06:39.034	2008/11/01 23:21:39.034	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEattitude(15)	10	2008/11/02 00:46:15.741	2008/11/02 01:01:15.741	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEattitude(16)	10	2008/11/02 02:25:52.447	2008/11/02 02:40:52.447	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEattitude(17)	10	2008/11/02 04:05:29.153	2008/11/02 04:20:29.153	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEattitude(18)	10	2008/11/02 05:45:05.858	2008/11/02 06:00:05.858	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEattitude(19)	10	2008/11/02 07:24:42.564	2008/11/02 07:39:42.564	0_day(s)_00:15:00.000	Assigned		FUSE
Management Report(2)	05	2008/11/02 08:00:00.000	2008/11/02 10:00:00.000	0_day(s)_02:00:00.000	Assigned		OpsTeam
FUSEcomm(4)	05	2008/11/02 09:10:19.872	2008/11/02 09:19:19.268	0_day(s)_00:08:59.396	Assigned		FUSE, Guam46, OpsTeam, SSR
FUSEattitude(20)	10	2008/11/02 09:19:19.268	2008/11/02 09:34:19.268	0_day(s)_00:15:00.000	Assigned		FUSE
FUSEcomm(5)	05	2008/11/02 10:57:37.735	2008/11/02 11:12:01.839	0_day(s)_00:14:24.104	Assigned		FUSE, Guam46, OpsTeam, SSR

STK Scheduler Online

The diagram below shows the software component process flow of the system. Individual components are described following the diagram.



SSA Sensor Tasking Process Flow Diagram (AGI SSA Solution)

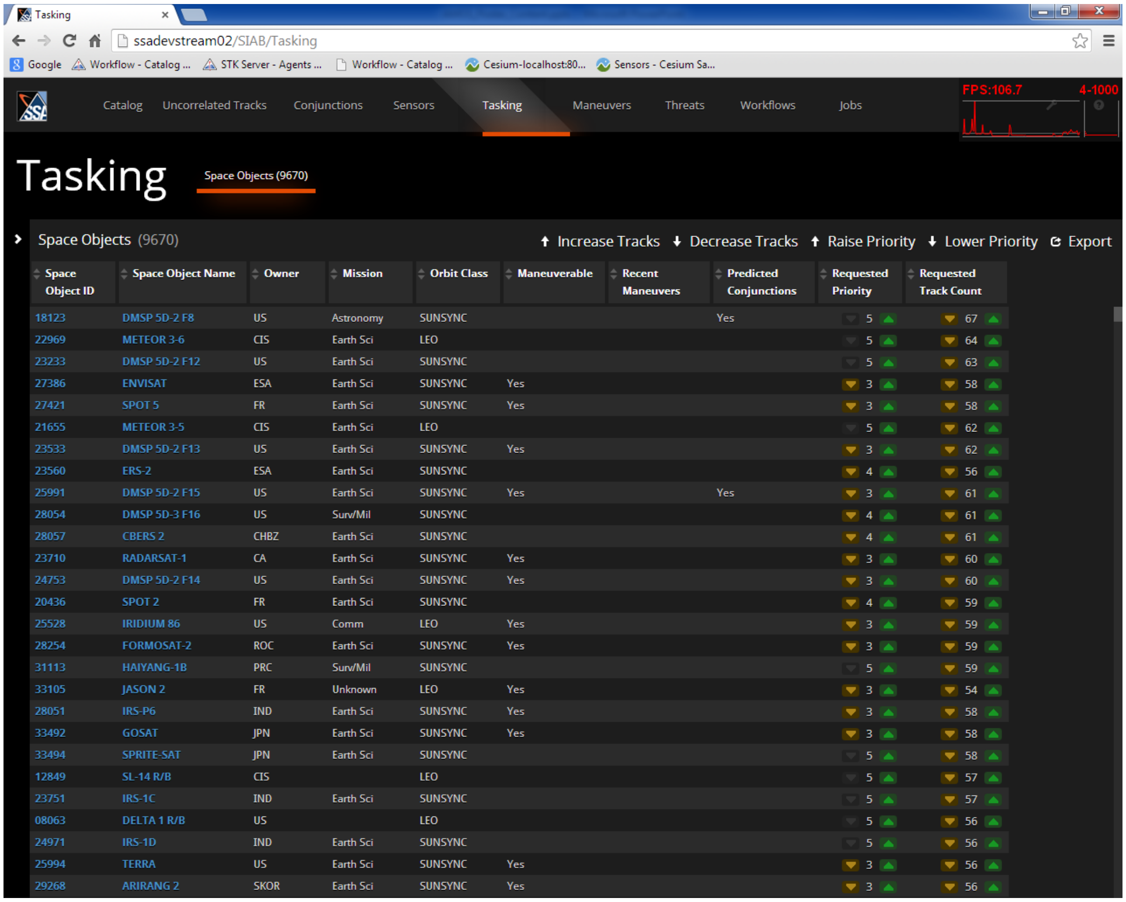
Visibility Computations

The first step in the process is the computation of all potential viewing opportunities between the network of sensors and the catalog. For this visibility computation Sensor Tasking performs multithreaded constrained STK access computations. The constraint processing architecture allows for the simultaneous trending and event detection of auxiliary parameters, such as those parameters related to favorable observation geometry (solar phase angle, etc.), and observation merit (covariance-based metrics, etc.) discussed in Part 1 of this paper. These parameters can be associated with each visibility interval and used to determine a ranking for each interval as an input to the scheduling algorithm.

The visibility computation is the longest portion of the overall process, but takes less than half an hour on a modest number of cores. The process is linearly scalable so the time can be reduced by adding more processors. It should be noted that the access computation process should only need to be run once a day for most objects. The access computation database generated by the tool can be well maintained by a daily computation cadence for the entire catalog with selected re-computations performed on an as-needed basis for any SSA events. This access update process can be performed in parallel with any sensor scheduling activities. The scheduling algorithm always uses the latest available access computations for scheduling activities.

Tasking Rule Set

A program-specific customizable rule set is included in the architecture to determine the tasking priority and number of requested tracks per object based on the orbit determination quality, the age of the orbit determination solution, high priority events, and the importance of the object (operational adversarial satellites may be considered more important to track than passive space debris, for instance). While the architecture provides for any way to define OD quality, subject to overall program requirements, the ability to marry such a function with the covariance-based solution provided by the filter and matched smoother of AGI's ODTK allows for the best possible decision making when building the schedule. OD quality is a function of several parametric indicators including covariance (size and shape), residuals, and filter-smoother consistency test among others. Tasking priorities, number of tracks, and the computed visibilities to each site are sent to the tasking and scheduling software component in a flat file referenced in the SOA service request API command.



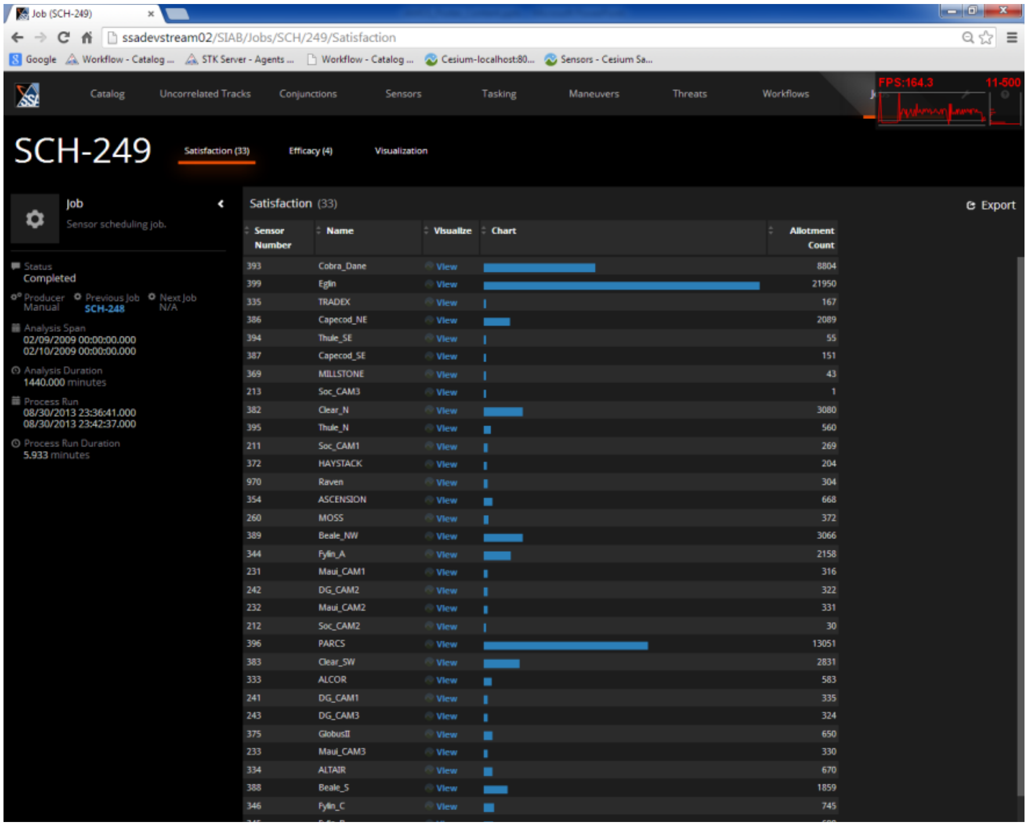
Tasking Request Display (priority & number of tracks)

Tasking and Scheduling Computations

The tasking and scheduling component of the system operates as a service. (Note that the system is capable of performing Tasking to support current CONOPS methodology as well as supporting true scheduling, based on a configuration setting when the call is made.) Requests for tasking OR scheduling are provided along with the supporting visibility information, priorities, and number of requested tracks for each object in the catalog. Tasking or scheduling is computed by the algorithms and results are returned via flat file. Tasking and scheduling results are not retained in the service component. The tasking and scheduling component can be used in one of three ways as described below:

Tasking

The tasking algorithm returns a non-temporal solution. Tracking sensors are individually configured with a daily capacity (via the same API used to make a tasking solution request). An efficient priority-biased load leveling algorithm is used to make tasking assignments in tracking object priority and object sensor preference order, essentially filling the sensor buckets until the requested tracks are all assigned or the sensor daily tracking capacities are reached. This option can be used to match the current approach used for SSA sensor tasking, but with high performance algorithms. Daily tasking for a 10,000 object catalog takes a few minutes for the underlying scheduling algorithm.



Scheduling Results Display

New Scheduling

A scheduling request returns a de-conflicted tracking schedule for all sensors in the network. In order to support the generation of a valid and optimized schedule, sensor setup times (time required between tracking events) and sensor accommodation (number of simultaneous tracking events a specific sensor can support) and sensor and visibility preferences (used to optimize the spacing of tracking events in inertial space) are configured via flat file as part of the scheduling service request. The STK Scheduler SSA scheduling algorithm generates the schedule solution using modified version of the standard STK Scheduler "Sequential" algorithm. This algorithm makes one pass though the possible task to possible opportunities after sorting the opportunities with priority grouping (to enforce priority-ordered tasking) and visibility preference. Scheduling takes a little longer than tasking. For a 10,000 object catalog the STK Scheduler SSA algorithm returns a valid 24-hour tracking schedule solution in less than 6 minutes.

The specific scheduling algorithm used for the demonstration was developed using the flexible algorithm structure [7] that is part of STK Scheduler. The STK Scheduler SSA scheduling algorithm generates the scheduling solution in a fashion similar to the standard STK Scheduler "sequential" algorithm.

The major modification needed for the SSA implementation was to add new manager code that allows each task to have multiple visits. In this case a task is to observe an RSO for at least a defined minimum number of visits, and up to a maximum desired number of visits. By posing the problem in this way it is a simple matter to use the existing periodic constraints to space out the observing times and to use resource preferences to adjust which asset is used to observe a particular object. This also allowed the size of the data description string constructed and passed to the scheduling service to be reduced significantly since duplicate information could be removed.

The manager code begins by grouping tasks into priority groups (groups of tasks with the same priority) and sorting the tasks within the group so that tasks with more desirability (based on resource preferences) are tried first. An ordered list of preferred resources for each task is also created at this time. During an initial pass the tasks in each priority group are processed using the resource preference order list. If there are two or more possible visibility windows with the same resource preference than the earliest start time is used. The task is assigned up to the minimum number of visits subject to the minimum and maximum time between visits. After all tasks in all priority groups have been tried a second pass is performed in which tasks with their minimum visits are allowed to claim more visits up to the maximum number desired.

This has turned out to be a very successful and efficient algorithm for this problem but with minor changes the manager could be modified to include repair heuristics. Once the problem is loaded in the server the process of finding a solution for the entire 10,000 RSOs takes less than 6 minutes for a number of revisits to each object representative of current sensor operations. Thus there is time to add additional loops through the data to find better solutions. The flexible architecture with the STK Scheduler interface also includes provisions of performing Monte Carlo, greedy, and user-defined search methods. A squeaky wheel technique is currently under development as well Using these methods an operation manager can fine tune the algorithm at a high level since the scheduling service has automated the work of de-conflicting the specific problem details.

Modified Scheduling

This option is similar to the New Scheduling feature except that a set of "locked" tracking assignments is submitted along with the normal scheduling service request. This option may be used if it is desirable to add new tasks to an existing scheduling without modifying the existing schedule. New sensor tracking assignments are scheduled to avoid conflicts with the locked tracking tasks. This method may be used in several iterations as necessary to fit in new high priority (or low priority) tasking with the minimum perturbation to the existing schedule.

Performance Enables Dynamic Retasking

The capability to perform re-scheduling or re-tasking of the entire sensor network for the entire space catalog in a few minutes opens up the option for improved responsiveness to non-emergency sensor network status updates and space object tasking priority changes. If a sensor goes offline because of a system failure, the entire sensor network schedule can be quickly adjusted to compensate. Similarly, if tracking priority updates are made for any reason during the day, a new tasking/tracking schedule can be responsively generated without having to wait up to a day for the next planning cycle. In addition, the success or failure of scheduled tasking/tracking during the day can be quickly fed back into the system to drive re-planning to replace missed tracks or eliminate unnecessary future planned tracks before any significant time has elapsed. In short, a faster and centralized tasking/scheduling solution makes the overall sensor network more responsive to changing events on the ground and in space, which leads to better maintenance of a more accurate space object catalog.

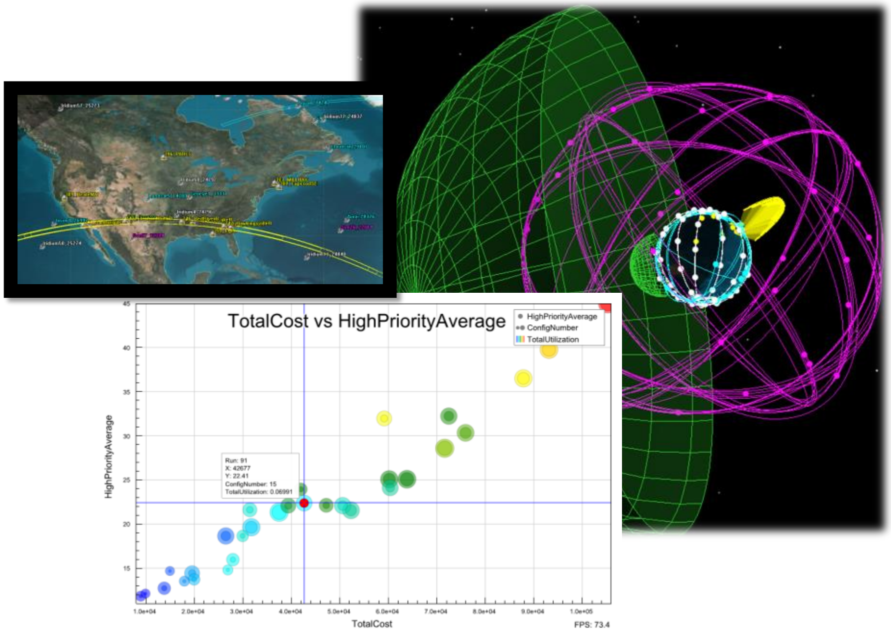
Parametric Analysis

The greatly improved performance of the scheduling component of the SSA Sensor Tasking tool enables not only improved operational responsiveness, it also enables parametric studies for both operational purposes and engineering analysis. Dozens of what-if sensor schedule scenarios could be generated and assessed in the time it currently takes to develop a single daily sensor tasking schedule today, even without additional automation.

By utilizing the automation and model-based engineering study capabilities of AGI's Modeling & Simulation Analysis tool, which includes an integrated Model Center component, these kinds of studies (for operational, programmatic, or engineering purposes) can be performed lights-out. Questions that could easily be answered with this existing software configured for SSN include:

- If the budget is not available to keep all sensors operating, which one(s) should be shut down to minimize the impact on overall catalog orbit accuracy?

- If the budget is available to add a new sensor, what type should it be and where should it be placed in order to provide the most benefit?



Sample SSA Network Parametric Analysis

The current SSN tasking software simply does not have the features or the performance to be used to answer these kinds of questions in an efficient manner.

Optimizing Scheduled Tracks for Improved Orbit Determination
The tracking goals described in Part 1 of this paper can be used to provide improved orbit determination accuracy. All of these goals can be achieved by leveraging existing features of the STK Scheduler COTS software as described in the following paragraphs.

> Using knowledge of the orbital period of the object and available minimum/maximum time between constraints in STK Scheduler for recurring tasks (such as tracking a space catalog object using one or more ground sensors), orbital separation of tracks can be guaranteed. For instance, if a space catalog object has an orbital period of 90 minutes and 48 tracks are requested per day that equates to 3 tracks every orbit (16 orbits in a 24 hour period). In order to best space out those tracking events, one could specify in STK Scheduler a minimum of 25 minutes and a maximum of 35 minutes between tracking event starts for that particular object.
>
> There are a variety of ways to use STK Scheduler resource preference features, accommodation attributes, capacity attributes, and/or task instance priorities to achieve observation type diversity. For instance, one could request a total of 24 observations for a particular object but limit the number of optical vs. radar tracking events.
>
> Favorable observation geometry and observational merit goals are easily achieved in the COTS solution through a combination of high fidelity modeling on the STK access computation side and the use of the opportunity (timeslot) preference attribute on the STK Scheduler side.

If for some reason the desired number of tracks per day cannot be met within these constraints (e.g. due to conflicts with higher priority tasking or lack of available sensor passes), the constraints can be relaxed to add additional tracking passes, even though they may not be ideal for orbit determination accuracy.

It is important to note that simply getting passes from multiple geographically-dispersed ground sensors does not guarantee that tracking will be adequately separated in inertial space. The rotation of the Earth and satellite orbital motion can converge to create repeated tracking over the same inertial region, even with multiple ground sites in use. By separating tracks based on fractions of the orbital period for each individual space catalog object (when possible), the inertial space separation of ground tracking is guaranteed and orbital accuracy improved.

COTS Quick Victory

The capabilities described in this paper are available today in the Sensor Tasking product of the SSA Software Suite, which is a presently available COTS solution with years of operational heritage on dozens of programs. Deploying these capabilities for a space tracking network such as the SSN will require (and enable) changes to the operations approach used today, but yield generous benefits in reduced planning timelines, reduced planner staffing requirements, improved responsiveness to changing events, and improved orbit determination accuracy providing a foundation for improved overall SSA. Since this capability has already been built, cost and schedule risks associated with new software development have been retired.

References

1. Beth L. Wilson, *Space surveillance Network Automated Tasker*, AAS 04-126, 14th AAS/AIAA Space Flight Mechanics Conference, 8-12 February 2004.
2. Beth L. Petrick, *Weighting Scheme For The Space surveillance Network Automated Tasker*, MSc Thesis, Air Force Institute of Technology, December 1994.
3. James G. Miller, *Covariance Analysis For Deep-Space Satellites With Radar And Optical Tracking Data*, AAS 05-314, AAS/AIAA Astrodynamics Specialists Conference, 7-11 August 2005.
4. K. Hill, P. Sydney, K. Hamada, R. Cortez, K. Luu, M. Jah, P. Schumacher, M. Coulman, J. Houchard, D. Noho'olewa, *Covariance-based Network Tasking of Optical Sensors*, AAS 10-150, AAS/AIAA Space Flight Mechanics Conference, 14-17 February 2010.
5. K. Hill, P. Sydney, K. Hamada, R. Cortez, K. Luu, P. Schumacher, D. Nishimoto, *Dynamic Tasking Of Networked Sensors Using Covariance Information*, Advanced Maui Optical And Space Surveillance Technologies Conference, 14-17 September 2010.
6. Stochastic Optimal Control, Theory And Application, Robert F. Stengel, John Wiley & Sons, Inc., New York, 1986.
7. W. Fisher, E. Herz, *A Flexible Architecture for Creating Scheduling Algorithms As Used in STK Scheduler,* 2013 International Workshop on Planning & Scheduling for Space March 26,27 2013