An Onboard Autonomous Response Prototype for an Earth Observing Spacecraft

Steve Chien, Russell Knight, Steve Schaffer, David R. Thompson, Brian Bue, Martina Troesch
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, USA
steve.chien@jpl.nasa.gov

Abstract

Prior space missions have not routinely used onboard decision-making. The Autonomous Sciencecraft (ASE), flying onboard the Earth Observing One spacecraft, has been flying autonomous agent software for the last decade that enables it to analyze acquired imagery on board and use that analysis to determine future imaging. However ASE takes approximately one hour to analyze and respond.

This paper describes the Earth Observing Autonomy (EOA) project to increase the responsiveness of spacecraft flight software for onboard decision-making as well as to increase the capabilities such flight software.

We describe prototype flight software that simulates acquisition of imagery, onboard spectral analysis of the imagery, replanning of imaging to include re-imaging of detected phenomenon, and then execution of this response imagery - all within this eight minute single overflight including the spacecraft response time (e.g. To re-point the spacecraft, acquire the image, etc.).

Introduction

The Earth Observing Autonomy (EOA) project targets the development of a spacecraft autonomy capability to enable a spacecraft to rapidly image, analyze the image, and re-image based on that analysis. Ideally this entire cycle would occur within a single overflight, imposing a responsiveness constraint of 5-8 minutes. While this software is applicable to a wide range of spacecraft, we assess this software against Orbview Class spacecraft (such as Worldview-3) [Ball 2015, DigitalGlobe 2015, Wikipedia 2015]. This would represent a dramatic improvement over the current state of the art, ASE [Chien et al. 2005], which responds within roughly 1 hour.

We have developed a software prototype of the EOA capability that includes several autonomy components:

1. *Onboard science processing algorithms.* Science analysis algorithms process onboard image data to detect science events and suggest reactions to maximize science return. Specifically we investigate the use of the Mixture –tuned Match Filter (MTMF) [Boardman and Kruse 2011] for onboard spectral analysis of acquired imagery (however ASE has already demonstrated other types of onboard analysis - thermal analysis for volcanoes and wildfires [Davies et al. 2006], spectral analysis for flooding [Ip et al. 2006], spectral analysis for cryosphere study [Doggett et al 2006], as well as spectral unmixing for mineralogical analysis [Thompson et al. 2011]).
2. *Onboard planning and scheduling software.* The Continuous Activity Scheduling Planning Execution and Replanning (CASPER) [Chien et al. 2000] combined with the Eagle Eye Mission Planning Software [Knight et al. 2013] system generates a baseline mission operations plans from observation requests. This baseline plan is subject to considerable modification onboard in response to data analysis from step 1. The model-based planning algorithms enable rapid response to a wide range of operations scenarios based on models of spacecraft constraints.
3. *Robust execution software.* The JPL core flight software [Weiss et al. 2013] (CFS) expands the CASPER activity plans into low-level spacecraft commands and includes a powerful and expressive sequencing engine. The CFS sequencing engine monitors the execution of the plan and has the flexibility and knowledge to perform improvements in execution as well as local responses to anomalies.

One challenge to spacecraft autonomy is *Limited computing resources*. An average spacecraft CPU offers 200 MIPS and 128 MB RAM – far less than a typical personal computer. For the EOA prototype, we baseline a Rad 750 or Leon processor for all of the autonomy capability.

EOA demonstrates an integrated autonomous mission using onboard science analysis, replanning, and robust execution. EOA performs intelligent science data analysis, and spacecraft retargeting, leading to a reduction in data downlinked and an increase in science return. These

---

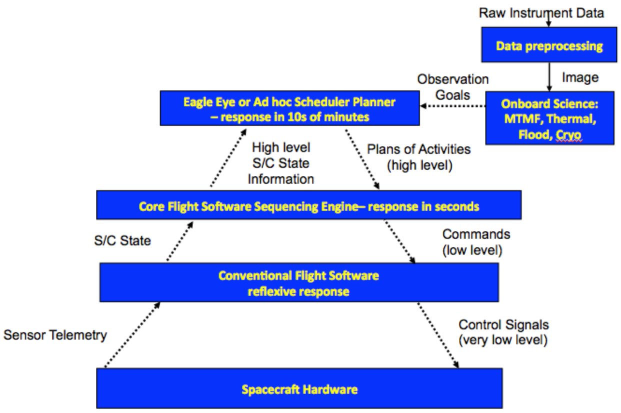

Figure 1: EOA Flight Software Architecture

capabilities enable radically different missions with significant onboard decision-making allowing the spacecraft to take advantage of novel science opportunities without the ground in the loop. The paradigm shift toward highly autonomous spacecraft will enable future NASA missions to achieve significantly greater science returns with reduced risk and reduced operations cost.

Autonomous Science Scenario

For our prototype we assume several baseline mission parameters:

Parameter	Value
Orbit	950 km Sun synchonous
Initial Science Images	30-40° lookahead from nadir
Response image	Nadir to 25° lookahead
Spacecraft slew rate	4.5° per second, instantaneous start and stop, no settle time
Imaging time	Total dwell of 1s per image

Our prototype highlights the following capabilities of EOA as it runs through an operations scenario:

1. *Autonomous Initial Plan Generation*. Eagle Eye/CASPER will generate a mission plan using an uplinked set of high-level goals requesting science observations and data downlinks. CFS will convert these plans to sequences of spacecraft commands and issue these commands to the Worldview-3 simulation. The initial schedule is shown by the light blue rectangles.
2. *Plan Execution*. As each scene in the initial schedule is imaged, the rectangle turns green.
3. *Onboard Image Analysis*. The onboard science analysis algorithms are run on each acquired initial image. If the onboard image analysis detects a feature/region/event of interest it will generate a request to re-image the site with a pre-specified priority.
4. *Onboard Replanning*. CASPER/Eagle Eye modifies the onboard schedule to respond to the science analysis recommendations to insert new observations and delete low-value future planned observations.
5. *Re-imaging of the targets*: targets that are re-imaged as responses are shown in yellow. Images that were in the initial schedule but are pre-empted due to contention with response images turn from light blue to red.

In Figure 2 we highlight some of the geometry characteristics of the EOA scenario. As the spacecraft orbits the earth, it has several viewing windows. The first viewing window is the initial science image window which covers from 30 to 40° in front of the spacecraft. The second viewing window is the response image which covers from 0° lookahead (nadir) to 25° lookahead. As the spacecraft flies over the earth it is imaging in large number of locations in the initial science window. As it acquires this imagery, software analyzes the imagery onboard the spacecraft. This analysis indicates the possible need to take follow-up imagery (in the response imaging window). For example, in the initial science window we might search for the thermal signature of a volcanic eruption or wildfire. In the response window we might further image to precisely determine the extent of the lava flow and the exact temperature map of the flow.

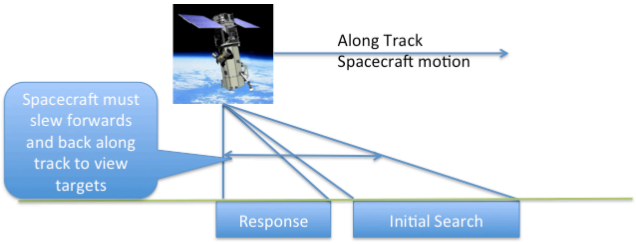

Figure 2: spacecraft must slew along track.
Distance between initial search and response windows large compared to distance between two requests in same window.

The goal of the scheduler is to accommodate as many of the initial science and response imaging requests but is guided by the priority of the requests and restricted by the pointing and slewing capabilities of the spacecraft (as well as any other operations constraints). As shown in Figure 2, from a side view, the speacecraft must slew forwards and backwards looking a variable amount ahead to view the image targets. At the same time, the spacecraft is moving forwards due to orbital motion (at approximately 7.5 km per second). Because processing the images requires some time, the initial search window is significantly ahead of the response window. This enables initial searched images to be processed/analyzed in time to allow for scheduling of follow-up imagery in the response window. The response window does not extend behind the spacecraft in order to maintain consistent lighting conditions.

This slewing forwards and backwards along the spacecraft motion track is complicated by two things. First, the angle at which the spacecraft must look forward to view the target is a non-linear function of when the spacecraft wishes to view the target. Specifically, at nadir, for the Earth, in a 950 km orbit, 1 degree of lookahead corresponds to 16.6 km ahead of nadir in the ground track. However, at 37° of lookahead, 1° of further lookahead (e.g. to 38° lookahead) corresponds to 30.7 km ahead in the ground track. The second issue is that typically the slew rate of the spacecraft is not linear, there is a ramp up acceleration of the spacecraft to some maximum slew rate, a portion of the slew at the maximum rate, and then a ramp down as the spacecraft arrives at the desired position.

Figure 3, Case 1 shows these two factors from the spacecraft pointing perspective. In this example the spacecraft is looking ahead and wants to view a target further ahead beyond the current look angle. The spacecraft could simply wait until the target comes into view, or it can slew ahead to meet the target. The blue line shows the track of a fixed point on the ground in terms of the look angle from the spacecraft as the spacecraft approaches the point. This line indicates that at time 0 the target is at 42° lookahead. The red line shows the angular position of the spacecraft reachable from the starting point of nadir as a function of time. The intersection of these two lines shows the earliest possible time that the spacecraft can view the target. The graph indicates that if the spacecraft begins slewing it will be able to reach the target but that the target will be at 38° lookahead when it is reached. In this case the motion of the spacecraft is helping us to meet the target earlier.

The right side of Figure 3 shows a different case, Case 2. In Case 2, the spacecraft is pointing at 38° lookahead, and wants to next view a target currently at 20° lookahead. In this case the spacecraft motion is carrying the target (relative to the spacecraft) away from the current spacecraft pointing and the slew must catch up. The graph shows that the by the time that the spacecraft can view the target it will be at 17° lookahead.

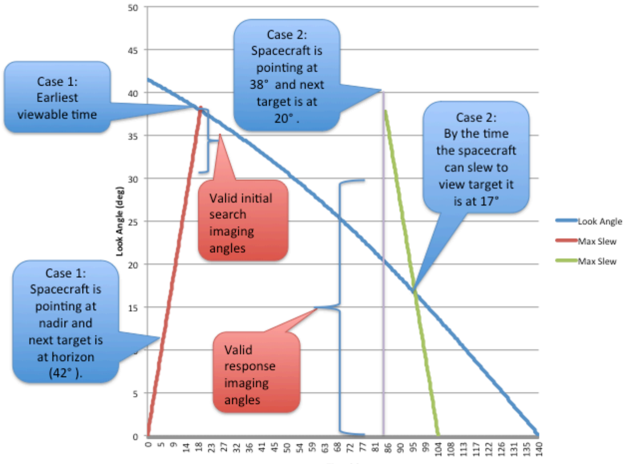

Figure 3: Location of target as viewed from spacecraft, as a function of time

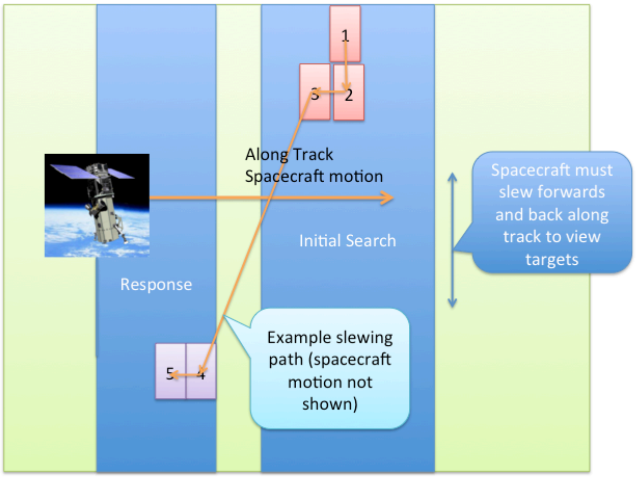

Figure 4: spacecraft must also slew across track.
Spacecraft moves along track as while slewing and imaging.

Figure 4 is a view from above the spacecraft looking down on the Earth. As the spacecraft moves along track (from left to right in Figure 4), the spacecraft must also slew across track (up and down in the Figure) as well as forwards and backwards along the ground track (left and right in the Figure) to image targets.

This scheduling problem is a challenging one for several reasons.

1. The spacecraft has limited ability to slew from one target to the next (e.g. each slew takes up valuable time).
2. Targets are distributed across the ground track of the spacecraft so that the amount of time required to image a target depends on the preceding and following (temporally) targets in the schedule.
3. Because the initial viewing and response doing windows are separated angularly, slewing back and forth between these windows can be wasteful of time.
4. Image analysis takes time. During this time spacecraft is moving towards the target(s). This is the reason why the initial image analysis and response image analysis windows are not overlapping, to allow the onboard software time to analyze the images.
5. Generating the schedule also takes time (the focus of this paper).
6. When calculating a start time to schedule an observation of a target, the spacecraft intercepts the target. The spacecraft must slew to a given position (of the target), reaching that position at

the exact time that the target is in that position relative to the spacecraft. This requires an accurate model of the spacecraft slew time as well as the ability to project where relative to the spacecraft any target will be at any point in time.

7. In addition to pointing, the scheduler must consider other resources such as power, thermal, data volume (e.g. [Chien et al. 2010, Chien et al. 2012]). However in this paper we focus on the pointing and slewing aspect of the problem as the state and resource management aspect of the problem has been considered elsewhere.

Two onboard re-scheduling software prototypes have been constructed. One is a standalone scheduler that operates with target information specified in the along/across spacecraft track coordinate frame of reference. The second uses the CASPER/Eagle Eye framework to operate in a lat/lon altitude frame of reference.

The standalone scheduler re-schedules the observation schedule from scratch each time a new request is received. This scheduler greedily schedules in a priority-first fashion, with each request being scheduled at the earliest possible start time. No backtracking across priority levels is performed therefore this algorithm is $O(n^2)$ where n is the number of requests that must be considered (the requests between nadir and the horizon).

The CASPER/Eagle Eye scheduler first generates an initial schedule by greedily scheduling in priority first order among all of the requests. When satisfying each request, it schedules the individual requests at the earliest possible start time. When rescheduling, CASPER/Eagle Eye takes the new request and searches the current plan within the valid time interval when the new request can be scheduled. It greedily replaces the earliest observation that can be replaced with valid slews to the preceding and following observations. Note that this algorithm presumes that the initial plan is packed tightly so that the new observation cannot be inserted in between currently scheduled observations without modification to the existing preceding and following observations. This rescheduling algorithm is O(n) where n is the number of requests in the valid along track viewing window for the new image request (a smaller number than the n used above but of the same order).

This image analysis software and response software was first implemented in a linux/workstation environment and then was ported into a VxWorks software simulation. In the future we plan to bring the software into a Rad 750 Hardware testbed which is the closest level to a flight testbed.

Current timing benchmarks show the image analysis, slewing, pointing, and re-planning within 0.5-2.0x real-time based on hardware and software assumptions. Because we have not optimized many of the computations this estimate is considered strong evidence that the performance of the current software prototype is close to flight worthy from a timing standpoint.

Summary

We have demonstrated onboard operations scheduling, image analysis, and re-imaging within a realistic flight software operating system and flight hardware performance environment. This prototype demonstrated the feasibility of performing such functions autonomously within a low earth-orbiting environment (roughly 5-8 minutes overflight time). Future efforts will further mature this concept and software by bringing the prototype into a relevant flight hardware testbed.

Acknowledgements

Portions of this work were performed at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

Ball Aerospace, The WorldView spacecraft series, http://www.ballaerospace.com/page.jsp?page=294, *retrieved 18 Feb 2015.*

Wikipedia, Worldview-3, http://en.wikipedia.org/wiki/**WorldView-3**, *retrieved 18 Feb 2015.*

DigitalGlobe, Worldview-3: DigitalGlobe, http://worldview3.digitalglobe.com, *retrieved 18 Feb 2015.*

Chien, S.; Sherwood, R.; Tran, D.; Cichy, B.; Rabideau, G.; Castano, R.; Davies, A.; Mandl, D.; Frye, S.; Trout, B.; Shulman, S.; Boyer, D. 2005a. Using Autonomy Flight Software to Improve Science Return on Earth Observing One. Journal of Aerospace Computing, Information, and Communication 2(4): 191–216.

J.W.Boardman and F.A.Kruse,"Analysis of imaging spectrometer data using N-dimensional geometry and a mixture-tuned matched filtering (MTMF) approach," *TGARS*, vol. 49, no. 11, pp. 4138–4152, 2011.

Davies, A. G., S. Chien, V. Baker, T. Doggett, J. Dohm, R. Greeley, F. Ip, R. Castano, B. Cichy, R. Lee, G. Rabideau, D. Tran and R. Sherwood (2006) Monitoring Active Volcanism with the Autonomous Sciencecraft Experiment (ASE). Remote Sensing of Environment, Vol. 101, Issue 4, pp. 427-446.

Ip, F., J. M. Dohm, V. R. Baker, T. Doggett, A. G. Davies, R. Castano, S. Chien, B. Cichy, R. Greeley, and R. Sherwood (2006) Development and Testing of the

Autonomous Spacecraft Experiment (ASE) floodwater classifiers: Real-time Smart Reconnaissance of Transient Flooding. Remote Sensing of Environment, Vol. 101, Issue 4, pp. 463-481.

Doggett, T., R. Greeley, A. G. Davies, S. Chien, B. Cichy, R. Castano, K. Williams, V. Baker, J. Dohm and F. Ip (2006) Autonomous On-Board Detection of Cryospheric Change. Remote Sensing of Environment, Vol. 101, Issue 4, pp. 447-462.

D. R. Thompson, B. Bornstein, S. Chien, S. Schaffer, D. Tran, B. Bue, R. Castano, D. Gleeson, A. Noell, Autonomous Spectral Discovery and Mapping Onboard the EO-1 spacecraft, IEEE Transactions on Geoscience and Remote Sensing . 2012 .

S. Chien, R. Knight, A. Stechert, R. Sherwood, and G. Rabideau, "Using Iterative Repair to Improve Responsiveness of Planning and Scheduling," *Proceedings of the 5th International Conference on Artificial Intelligence Planning and Scheduling*, Breckenridge, CO, April 2000.

R. Knight, A. Donnellan, J. Green, Mission Design Evaluation Using Automated Planning for High Resolution Imaging of Dynamic Surface Processes from the ISS, *International Workshop on Planning and Scheduling for Space (IWPSS 2013).* Moffett Field, CA. March 2013.

K. Weiss, "An Introduction to the JPL Flight Software Product Line", 2013 Workshop on Spacecraft Flight Software (FSW-13), Pasadena, CA, December 2013.