A Science-Based Executive for Autonomous Planetary Vehicles

Stephen F. Peters
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109
USA

phone: (818) 354-1738
fax: (818) 354-8172

stevep@jpl.nasa.gov

The performance, and indeed ultimate success, of operations executed by machines in contact with natural environments is uncertain. When the machine is a vehicle on a distant planet, light time alone can result in a feedback loop through an operator on Earth from tens of minutes to even hours. With other Earth-bound considerations such as deep space antenna scheduling, a reasonable expectation for planetary surface operations is one uplink per day.

During its day, the vehicle travels to various sites, positions its instruments, and takes measurements desired by scientists on Earth. To maximize the effective use of the spacecraft's operational time, it is desirable for the spacecraft to have the ability to adjust its operations based upon actual performance that day.

Traditionally, requests for scientific operations are converted on the ground into a plan of timed actions and then uplinked to the spacecraft. If these requests themselves are uplinked, the spacecraft can have the knowledge required to optimize operations based upon execution feedback. A system which supports this science-based uplink could also support opportunistic observation requests, triggered by events sensed locally in the remote planetary environment.

With these goals in mind, a science-based executive control system has been developed and demonstrated with the Rocky7 research rover.

Given a set of science requests, a time deadline, and continuously monitoring the rover state, the executive, assisted by a simple planner, performs preferred tasks in the available time.

Each request embodies the following information:
* location in the local planetary surface frame where observation is to be made
* relative ordering constraint (first, last, anytime)
* relative priority
* duration of operation (excluding time for traverse to location)
* action or observation to be performed
* execution status (request, active, complete)

The rover state includes a great deal of information, including:
* current location in the local planetary frame
* state of the camera/instrument mast
* presence of a sample in the scoop

Other rover properties modelled include:
* nominal traverse velocity

Rocky7 implements a broad suite of robust behaviors as finite state machines in ControlShell. These range in abstraction from simple actions like driving wheels and taking images, to detection of navigation obstacles, to driving to a specified location in the local planetary frame while avoiding obstacles. These behaviors provide feedback of success and failure in addition to updates of the rover state.

At the top of the Rocky7 autonomy architecture is the science-based executive with its planning assistant, implemented in common lisp and the Executive Support Language (ESL) library.

The executive has a function for each science request supported. At its heart is a simple function which monitors execution and reacts to trigger events (which implement opportunistic requests and safing actions), calling the planning assistant to replan from the set of uplinked requests as needed.

For this implementation, Network Data Distribution System (NDDS) productions were used for communication between the executive and the finite state machines.

Many tests with Rocky7 have been performed, and a demonstration was videotaped in JPL's Mars Yard.

The science-based executive was given a deadline and a set of requests:

standard science requests (each with parameters listed above)
* acquire a sample
* make a spectrometer measurement
* take a panorama
* rover startup initialization

opportunistic science request
* observe clouds -- triggered by reduced illumination of the sun sensor voltage

safing operation
* stow mast -- triggered by low power voltage

The executive called the planning assistant to make a plan which could be executed in the available time, estimating the amount of time required for scheduled observations and traverses between, and obeying other constraints.

initial plan:
* rover startup initialization
* make a spectrometer measurement
* acquire a sample
* take a panorama

Initialization was executed and then Rocky7 started a traverse to the site for the spectrometer measurement. During this traverse, a reduced-sunlight event was issued, triggering the opportunistic observation. The traverse was interrupted, the mast deployed and pointed in the direction of the sun, an image was taken, and the mast was stowed. Ready to resume normal operations, the planning assistant was again called.

replan:
* acquire a sample
* take panorama

There was no longer enough time to perform all the requested observations, so the planner skipped the lower-priority spectrometer measurement. Execution resumed with this new plan. During the panorama, a low-power event was issued, triggering the safing operation. The panorama was interrupted, the mast stowed, and Rocky7 was left in an idle state.

Although the science-based executive was fully functional and all interfaces with the ControlShell finite state machines were fully implemented, some portions of the demonstration were simulated.

The executive lisp/ESL implementation itself was not run in the Rocky7 on board VxWorks environment, but alternatively from a Sparc workstation and a Macintosh powerbook. The VxWorks lisp implementation used for the Deep Space One mission's Remote Agent Experiment was too large for the memory- and power-constrained Rocky7. A compact implementation of lisp for VxWorks capable of supporting ESL has since been implemented.

ControlShell data flow diagram generation of low-power and reduced-sunlight events was not implemented, so these events were issued manually via NDDS productions.

The executive and planning assistant implementation is compact, consisting of eleven science request functions, nineteen planning assistant functions, and a thirteen-line main executive function. More than fifty percent of the code is the command and telemetry interface with the Rocky7 finite state machines.

A simple but powerful executive for control of planetary robotic vehicles, capable of accepting science requests directly, has been implemented, tested, and demonstrated. This was greatly facilitated by Rocky7's hierarchy of robust behaviors with articulate feedback.