Pursuing a Lingua Franca for Space Mission Planning Applications

Dr. Jeremy Frank
NASA Ames Research Center
Outline

• Space Mission Planning
• Automated Planning
• Model-Based Planning and Languages
• Model-Based Planning Applications
• Pros and Cons
• Roadmap
• Conclusion
Space Mission Planning

- Mission planning is a key part of the Mission Operations System (MOS).
- Mission Planning is the process of determining how and when spacecraft subsystems will act.
- The Mission Planning System (MPS) is the software that helps the Mission Planner perform this task.
Space Mission Planning

• The Mission Planning System is integrated with other parts of the MOS:
  – Sequencing, Flight Dynamics, Attitude determination, Command and Telemetry, Flight software
  – Surface Operations, Science operations

• The Mission Planning System integrates this information, allows planning and scheduling, checks constraints.
Automated Planning

- Automated planning functions or services in the Mission Planning System include:
  - Automated planning, e.g. action selection
  - Automated scheduling, e.g. action ordering
  - Constraint checking, e.g. evaluation of resource constraints, time constraints, checking conditions
Automated Planning

• Mixed Initiative planning blends automated planning with user actions
  – Manual planning or scheduling
  – Adding new constraints
  – Waiving constraint violations and removing constraints
Model Based Planning and Languages

• A model-based planner is
  – A configurable automated planner
  – Employing a language to describe the planning problem

• The language elements include
  – Objects, states and resources
  – Actions
    • Conditions and Effects
Model Based Planning and Languages

• A planning problem consists of
  – A model
  – An initial state description
  – A set of goal states

• The software reads the description and produces a plan

• There are myriad algorithms with many different properties
Model Based Planning and Languages

• Model elements in more detail
  – Objects – things in the world
    • E.g. targets, spacecraft components
  – State variables – time-varying properties
    • E.g. available power, mode of system
  – Actions
    • Conditions – what must be true for an action to have the desired effect
    • Effects – what changes when the action is executed
Model Based Planning and Languages

(:durative-action slew
 :parameters (?from – attitude ?to - attitude)
 :duration (= ?duration 5)
 :condition
 (and (at start (pointing ?from))
  (at start (cpu-on))
  (over all (cpu-on))
  (at start (>= (sunangle) 20.0))
  (over all (>= (sunangle) 20.0))
  (at start (communicating))
  (over all (communicating))
  (at start (>= (batterycharge) 2.0))))

:effect
 (and (at start (decrease (batterycharge)2.0))
  (at start (not (pointing ?from)))
  (at end (pointing ?to))))
Model Based Planning and Languages

• Comparison: Linear Programming
  – LP solver is a piece of software
  – It accepts as input a language:
    • Variables, linear constraints over variables, linear objective function over the same variables
  – Objective: maximize objective while satisfying all linear constraints
    • User has limited (or no) control over solver
      – Max: x+y-3z
      – Subj: x-z<4, y-z<2, 0≤x≤10, 0≤y≤10, 0≤z≤10
Model Based Planning and Languages

• Comparison: Satellite Toolkit
  – Configure spacecraft
    • Orbit, sensors, solar panels & battery, ACS...
  – Simulate orbits
  – Report on metrics
    • Coverage, orbit ephemerides, power, etc.

• Automated planners can be built on top of StK but it does not do the actual planning and scheduling itself.
  – http://www.riversideresearch.org/labs/modeling_and_application_development_lab
Model Based Planning Applications

• Fielded space mission applications:
  – Modified Antarctic Mapping Mission Scheduler
    • Antarctic Synthetic Aperture Radar scheduling
  – MAPGEN
    • Mars Exploration Rover science activity planning
  – Phoenix Science Interface
    • Phoenix Mars mission science activity planning
  – MEXAR II
    • Mars Express downlink scheduling
Model Based Planning Applications

• Fielded space mission applications:
  – Orbital Express Ground System
    • Communications and commanding scheduling
  – Solar Array Constraints Engine
    • ISS solar array planning
  – Data Chaser Automated Planner and Scheduler
    • STS-85 payload planner and scheduler
  – EO-1 Ground System
    • Earth observing satellite scheduler
Model Based Planning Applications

• Space mission applications in development:
  – LASS
    • LADEE (Lunar Atmospheric Dust Explorer) planning
  – MSLICE
    • Curiosity Mars rover science activity planning
  – NGPS
    • ISS crew activity planning and scheduling
Model Based Planning
Pros and Cons

• Pros
  – Flexibility during MPS development
    • Models can be revised with no code changes
  – Expressive
    • Able to model many applications
    • More intuitive than (e.g.) linear programming
  – Open and public domain
    • Many languages developed in academia
    • Many planners and tools are freely available
Model Based Planning
Pros and Cons

• Cons
  – Simplistic modeling language
    • Unable to describe complex continuous constraints
    • Limited built-in (semantic) integration with other engines e.g. Matlab, StK
    • Can be tricky to use (modeling, planners)
  – User interfaces lacking
    • Critical for plan and model visualization
  – Not an industry standard
  – Not (quite) an academic standard either
    • Several languages, many dialects!
Roadmap

• Definitions of terms
  – State, action, object, constraint, resource, etc.
  – Use academic work as a springboard

• Model elements
  – Logical vs visual (UI) elements
  – Ontologies

• Tool integration standards
  – More ontologies?

• OMG standard (see backup slides)
Conclusion

- Model-based planning is a paradigm worth following for space mission application development.
- Model based planning has been used in several space mission applications – As well as numerous other planning applications
- The model-based paradigm can reduce up-front and recurring costs of MPS development, but need a roadmap for progress.
Conclusion

• Resources for automated planning and model-based planning:
  – http://eecs.oregonstate.edu/ipc-learn/
Conclusion

- References for model-based space-mission applications:
  - FRATINI, S., PECORA, F., and CESTA, A. Unifying Planning and Scheduling as Timelines in a Component-Based Perspective. *Archives of Control Sciences, 18(2):231-271, 2008.*
Planning vs Scheduling: The Computer Science View

• Scheduling
  – Given a fixed list of activities and a set of constraints (and possibly preferences)
  – Order the activities so that all constraints are satisfied (and there is no preferred ordering)

• Planning
  – Given a state of the world, a set of desired states, and a set of action types that change the world
  – Choose actions to execute to reach the goal

• “Inside every planning problem there is lurks a scheduling problem.”
Planning vs Scheduling: The Computer Science View

• How long does it take?
  – Checking a plan or schedule is ‘easy’
    • Because the constraints are ‘easy’ to check
  – Scheduling n activities requires generating n! schedules in the worst case
    • Ensuring all constraints are satisfied allows you to stop ‘early’
    • Optimal scheduling usually takes longer
  – Planning takes longer than scheduling
    • Since you have to select actions and order them
  – Special cases take less time (e.g. LPs)
Flavors of Modeling Languages

- STRIPS
- ADL
- PDDL
  - PDDL+
  - Processes
- Timeline-centric
  - SAS
  - IxTeT
  - ‘NASA’ languages (NDDL, ANML, AML)
Models and Plans

• If you have a model:
  – You can pose planning problems
    • can you reach the goal from the initial state?
  – You have the basics for representing plans too
    • A plan is an ordered list of action names
    • Possibly coupled with an ordered list of states of state variables, resources
    • Also possibly coupled with the rules linking actions, conditions and effects
Object Mechanism Group
Standards (sort of)

• OWL (Web Ontology Language)
  – Objects and Relationships

• UML
  – Petrie Nets / State Charts and SysML

• SPS (Sensor Planning Service)
  – Open GIS interface for managing sensors

• XTCE / XTEDS
  – Spacecraft command and data

• Procedure Representation Language
  – XML representation of procedures