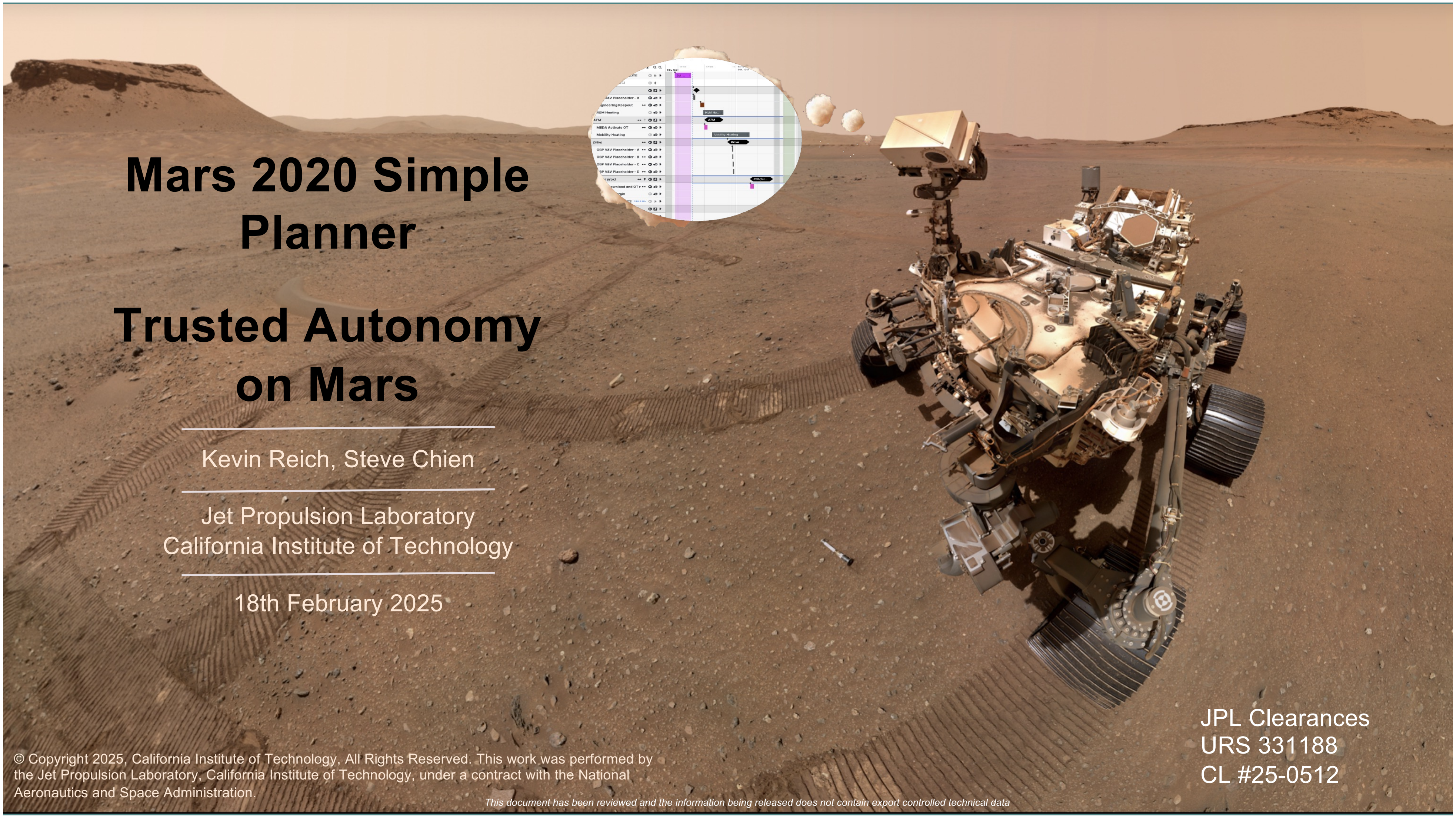


Mars 2020 Simple Planner

Trusted Autonomy on Mars

Kevin Reich, Steve Chien

Jet Propulsion Laboratory
California Institute of Technology

18th February 2025

JPL Clearances
URS 331188
CL #25-0512

Introduction

Simple Planner is flight and ground system that enables the Mars 2020 Perseverance Rover to adjust to: **unexpected state**, such as Martian temperature fluctuations or battery performance, and **activity execution feedback**, such as activities failing, ending earlier or later than expected.

Simple Planner development began in 2016, with Verification and Validation (V+V) test campaign beginning October of 2021, and deployment to operations October 5th, 2023.

Building and Deploying Trusted Autonomy is ***a full lifecycle process*** that begins with *conception*, continues through *design* and *prototyping*, product *build*, *testing*, *training*, and *deployment*.
[Potentially multiple iterations per multiple deliveries]

This talk describes how the Onboard Planner (OBP) moved through this process, from formulation, design, analysis and prototyping through testing.

MARS 2020 PERSEVERANCE

Talks on the M2020 Simple Planner

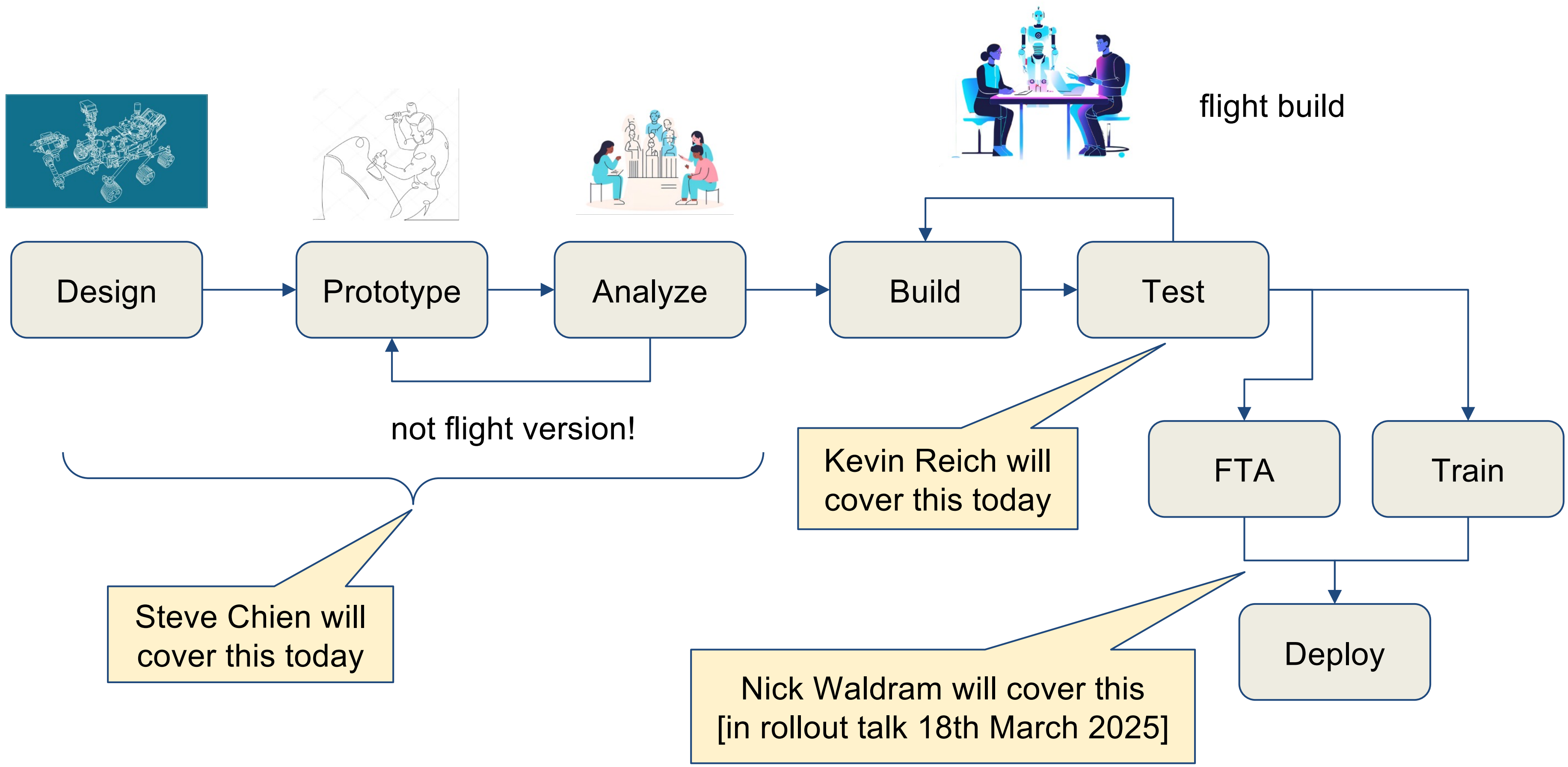
Topic	Speaker	Date
Overview of Simple Planner	Moffi	5th December 2024
Onboard Planner: Flight Software	Gaines	4th February 2025
Onboard Planner: Trusted AI on Mars	Reich, Chien	18th February 2025
Simple Planner: Ground Tools for Operations	Connell	25th February 2025
Simple Planner: Systems Engineering Operations with Autonomy	Hazelrig	11th March 2025
Rollout of the Simple Planner	Waldram	18th March 2025

You are here

Location: All talks are in Pickering Auditorium, Building 321, JPL Campus.
Time: All talks are 12 noon - 1 PM PST
Miss it? Recordings of all talks will be archived on JPLTube
Slides will be posted at https://ai.jpl.nasa.gov/public/projects/m2020-scheduler/

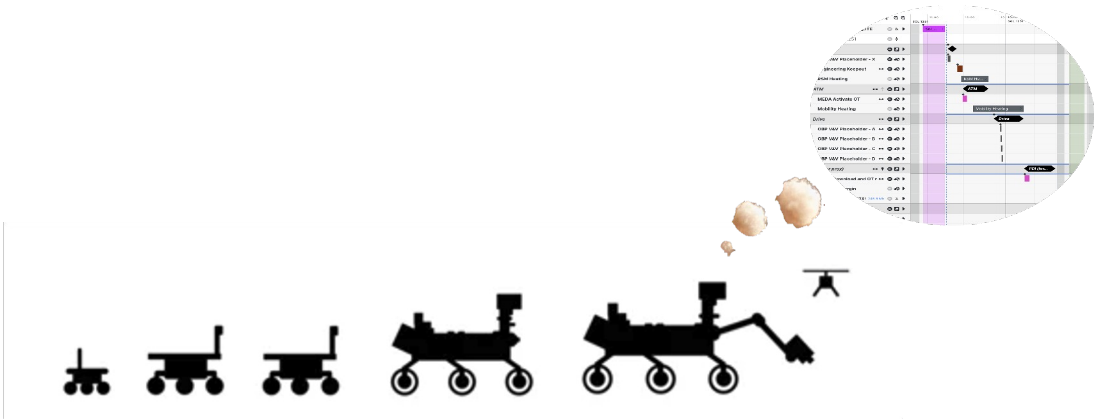
MARS 2020 PERSEVERANCE

Trust built throughout the Design - Build Lifecycle

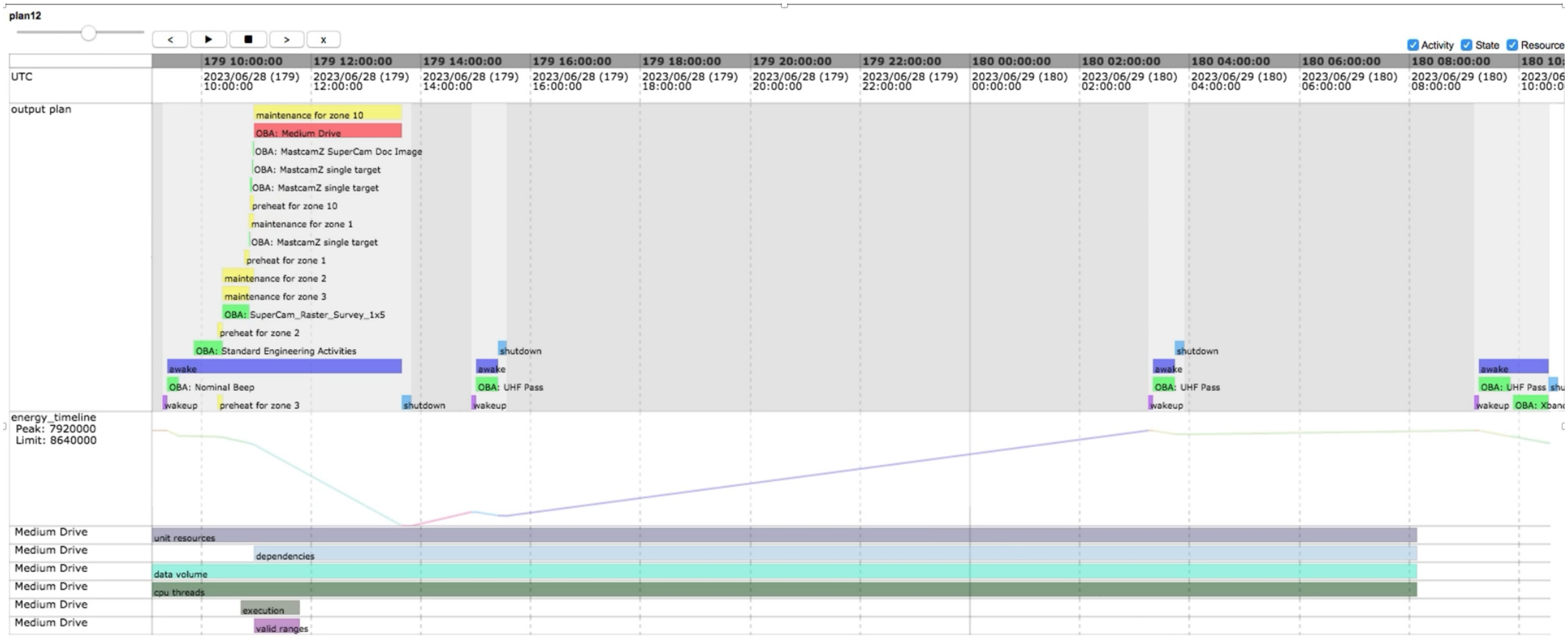


MARS 2020 PERSEVERANCE

Onboard Planner: Background



OBP Scheduling

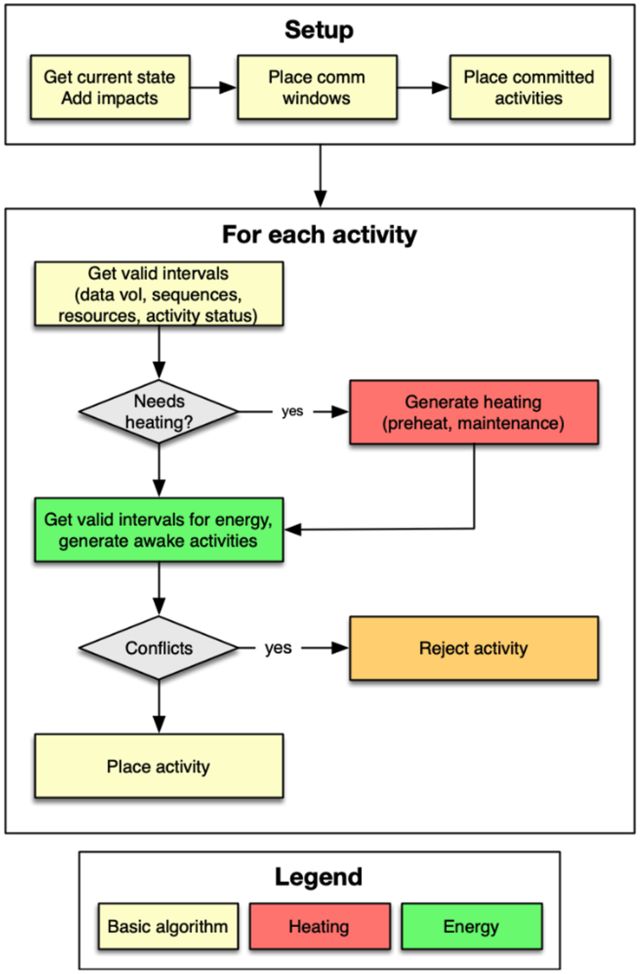


MARS 2020 PERSEVERANCE

OBP Scheduling

Onboard Planner iteratively constructs Schedule

- Event driven scheduler invocation:
  - Plan activation
  - Wakeup
  - Activity vetoed, failed, or aborted
  - Activity ends earlier/later than scheduled time by Δ
  - Timer
- To minimize computation time, scheduler does not backtrack across activity placements
  - Ground awareness of search ordering for specific sols



See:

Gaines, D.; Rabideau, G.; Wong, V.; Kuhn, S.; Fosse, E.; and Chien, S. The Mars 2020 On-Board Planner: Balancing Performance and Computational Constraints. *In Flight Software Workshop*, February 2022.

Gaines, D.; Chien, S.; Rabideau, G.; Kuhn, S.; Wong, V.; Yelamanchili, A.; Towey, S.; Agrawal, J.; Chi, W.; Connell, A.; Davis, E.; and Lohr, C. Onboard Planning for the Mars 2020 Perseverance Rover. *In 16th Symposium on Advanced Space Technologies in Robotics and Automation,* June 2022.

MARS 2020 PERSEVERANCE

OBP Challenges - Limited Computing

Challenge:

- The Rad750 onboard computer for the Perseverance rover is very limited in its computational power
    - Computer produces ~133 MIPS (a modern cell phone has 50x more compute!)
    - OBP must share the processor with many other critical software processes

Approach:

- To mitigate the OBP load on the Rad750 flight processor:
    - The M2020 OBP uses lightweight Flexible Execution operating at 1 Hz to adjust execution times without rescheduling
    - OBP uses event-driven rescheduling to control scheduler invocation
    - OBP does not backtrack to keep scheduler runtimes to less than 30s estimated time (60s hard estimate)
    - OBP uses a throttling mechanism prevents the scheduler from invoking so frequently that it starves other software processes

See: Gaines, D.; Rabideau, G.; Wong, V.; Kuhn, S.; Fosse, E.; and Chien, S. The Mars 2020 On-Board Planner: Balancing Performance and Computational Constraints. In Flight Software Workshop, February 2022.

OBP Challenges - Software Priority

Challenge:

- If OBP Priority is too low → OBP will complete slowly
  → commit window too large → OBP will not reclaim time

- If OBP Priority is too high, it will starve other critical FSW processes

Approach:

- Set the OBP priority higher so that it can complete quickly

- Use Flexible Execution, Event-driven scheduling, and Throttling, to reduce OBP invocation so that other processes can complete

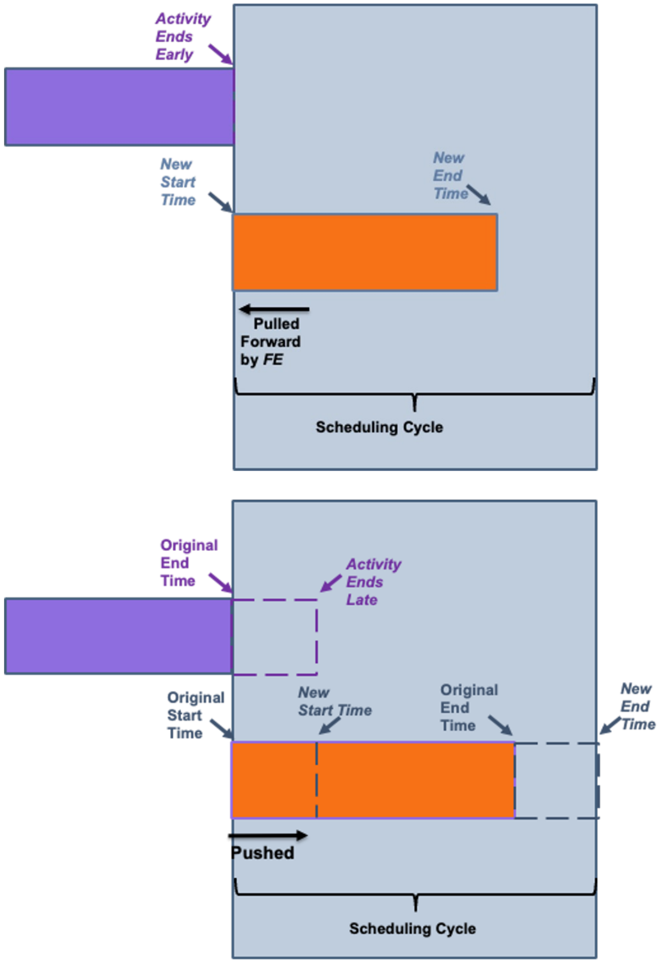
See:
Gaines, D.; Rabideau, G.; Wong, V.; Kuhn, S.; Fosse, E.; and Chien, S. The Mars 2020 On-Board Planner: Balancing Performance and Computational Constraints. In Flight Software Workshop, February 2022.

MARS 2020 PERSEVERANCE

OBP - Flexible Execution (FE)

Flexible Execution is a lightweight computational process that allows:

- “pull” Activities to execute earlier if states, resources, execution time ranges, and predecessors complete early
- “push” Activities to start late if prior activities run late
- FE reduces scheduler computation and avoids scheduler runtime loss



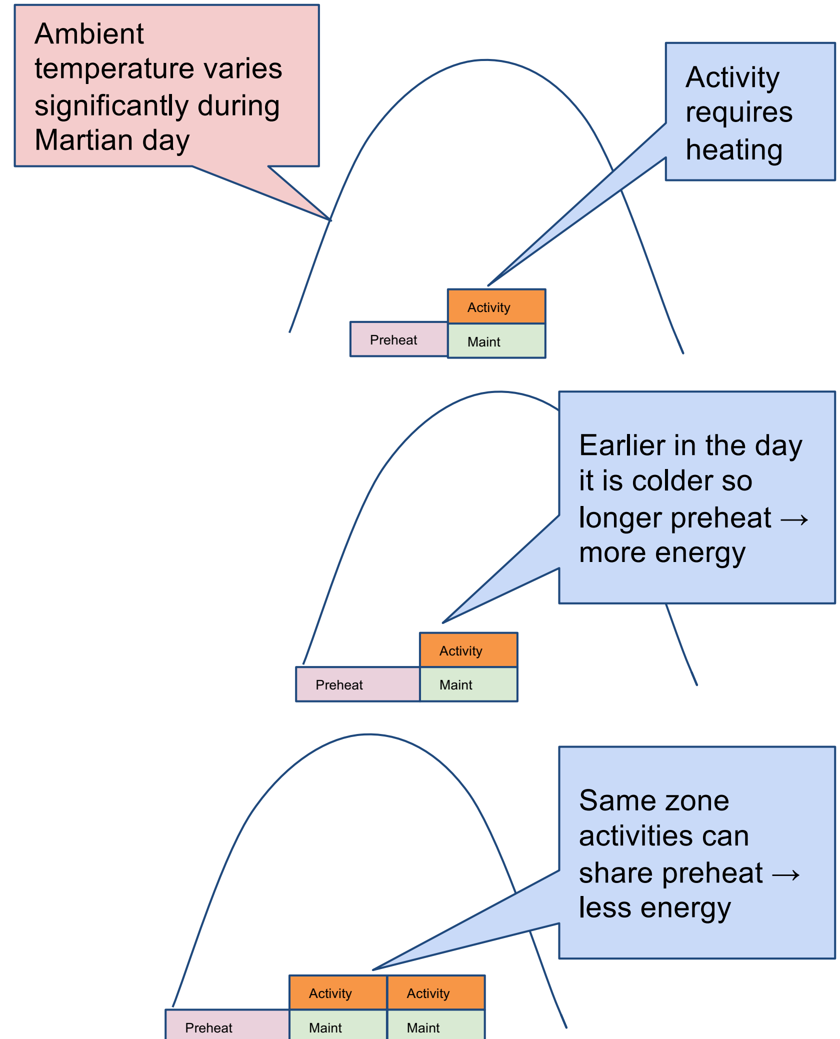
See:
J. Agrawal, W. Chi, S. A. Chien, G. Rabideau, D. Gaines, and S. Kuhn. Analyzing the effectiveness of rescheduling and flexible execution methods to address uncertainty in execution duration for a planetary rover. *Robotics and Autonomous Systems*, 140 (2021) 103758, 2021.

MARS 2020 PERSEVERANCE

OBP Challenges - Thermal

OBP must implement thermal management for the Perseverance rover.

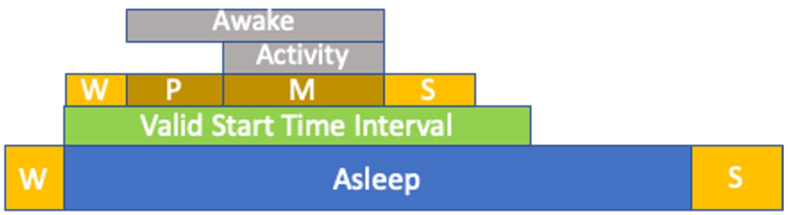
- Mechanisms onboard the Perseverance rover must be at a safe temperature before being used (such as to drive motors, arm motors, or instruments).
- Therefore OBP needs to schedule preheat and maintenance heating to support these activities.
- Preheat lead time (and energy consumption) can be affected by time of day and shared costs with other activities.

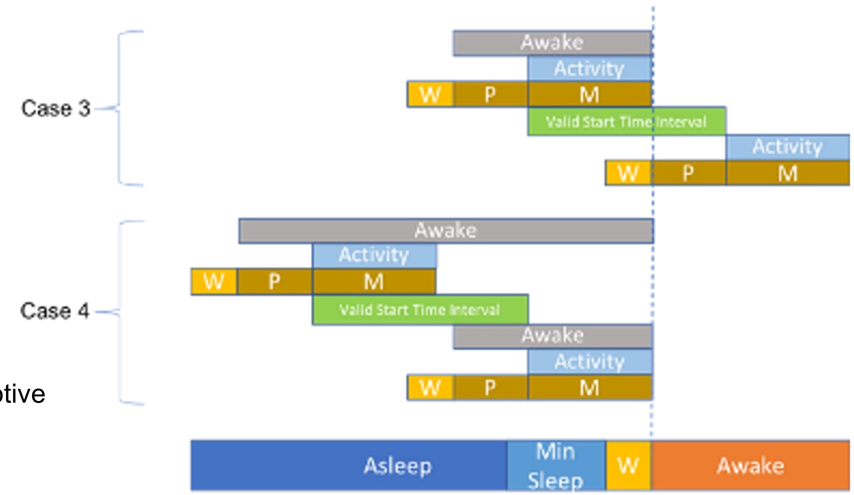


OBP Challenges - Energy

The Perseverance rover is energy constrained

- The rover just being awake → power negative state
- Most activities require that the rover (RCE) be awake in order to be performed
- Common pattern of operations is that the rover takes naps throughout the martian day, or *sol*, in order to conserve energy
- OBP must manage the wake sleep schedule of the rover

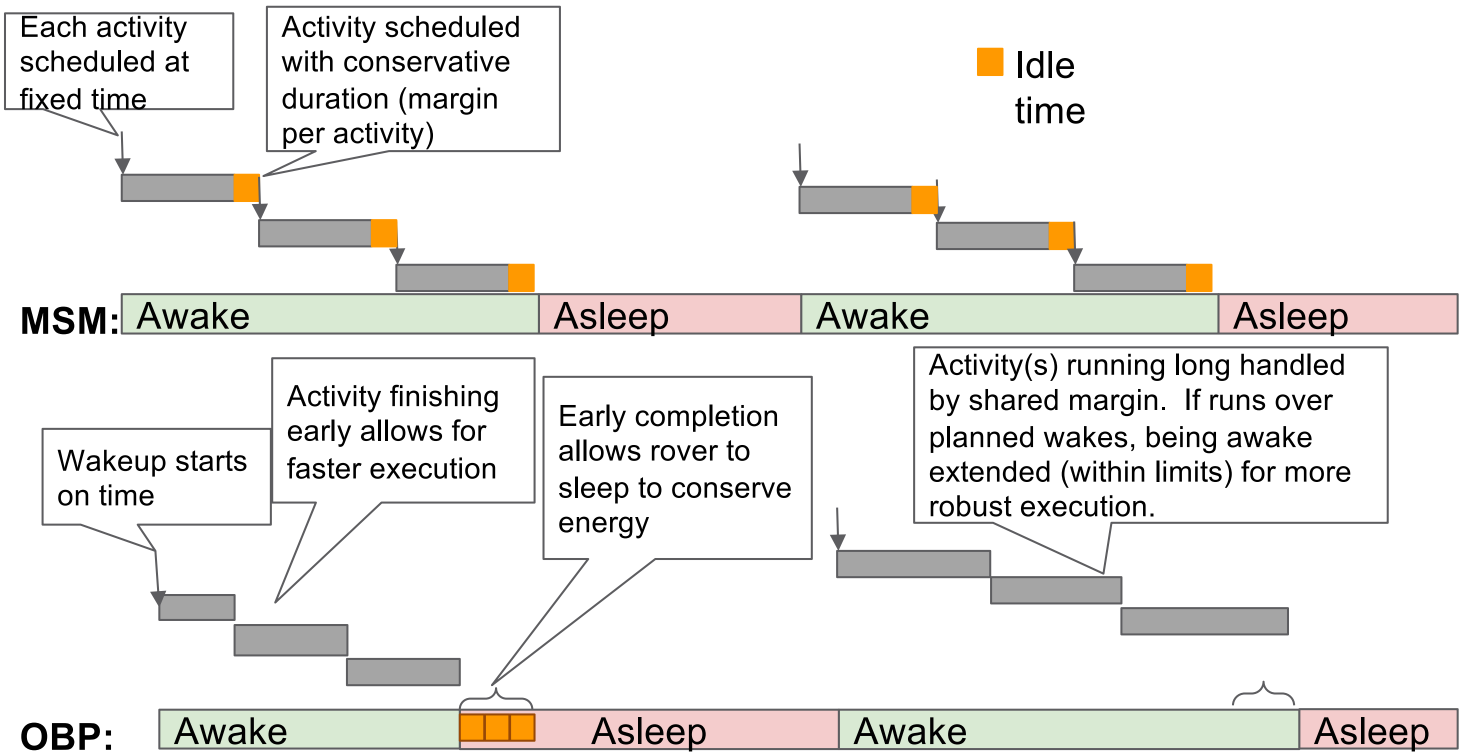




See:
W. Chi, S.Chien, and J. Agrawal. Scheduling with complex consumptive resources for a planetary rover. In International Conference on Automated Planning and Scheduling (ICAPS 2020), Nancy, France, October 2020.

MARS 2020 PERSEVERANCE

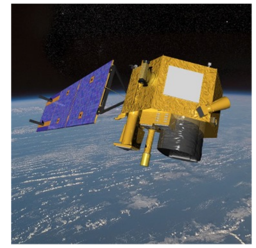
Use Case Example: MSM vs. OBP



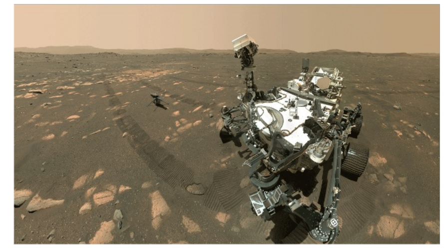
OBP SP1 Focuses on saving energy in the current sol for future sols.
OBP SP2 enables greater activity flexibility in re-ordering activities!

MARS 2020 PERSEVERANCE

M2020 SP - Related Work



Earth Observing One
Image courtesy NASA



M2020 Perseverance Rover
Image courtesy NASA

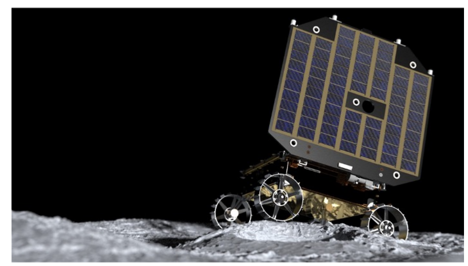
Prior Onboard Planners:

- Remote Agent Experiment (RAX) 48h 1999[1]
- CASPER on IPEX 14 months 2013-2014[1]
- Mexec on ASTERIA 4-20 September 2019[1]

- **CASPER on Earth Observing One over 12 years 2003-2017[2]**
- **M2020 Perseverance much more complex and high-value mission[2]**

Future Rover Mission Autonomy (Lunar):

- Moonranger[1] rover autonomy (scheduled 2025)
- CADRE[1] flight of Mexec (scheduled 2025)



Moonranger Rover
Image courtesy
Astrobotic, CMU

CADRE Rovers
Image courtesy NASA

[1] - Technology demonstration mission
[2] - Operational usage

MARS 2020 PERSEVERANCE

Developing Trusted Autonomy: SW Development - Analyze (Informal Methods)

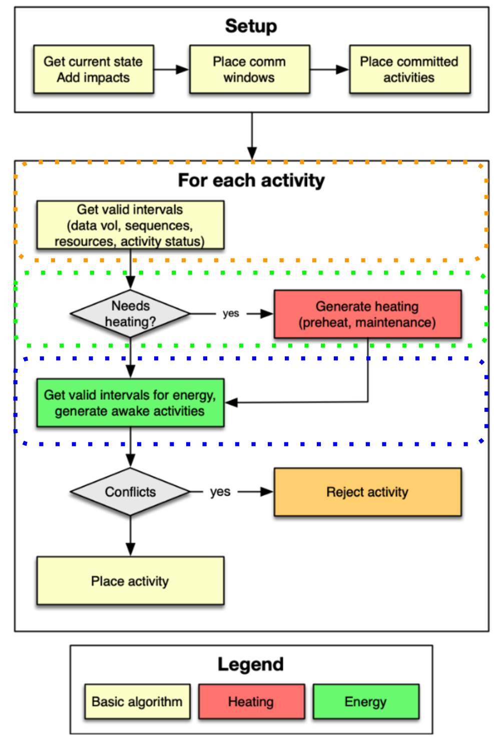
Trusted M2020 OBP – Informal Methods Computational Model of Scheduler Runtime

- Developed an in-depth computational complexity model for onboard planner runtime (Chi, Chien, et al. 2017).

  $$((4d + 4dp) + 2*(4d^2 + 4d^{2p}))n + (4r^2 + T + (4+4p)^2 + (16 + 32p + 16p^2) + 2*(32d + 64dp + 32dp^2) + 2)n^2 + (8 + 2*(64 + 192p + 192p^2 + 64p^3))n^3$$

  where
  n = Number of Activities (excluding preheats, maintenance, awakes)
  d = Number of discrete intervals
  T = Number of Timelines
  P = Total Number of preheats

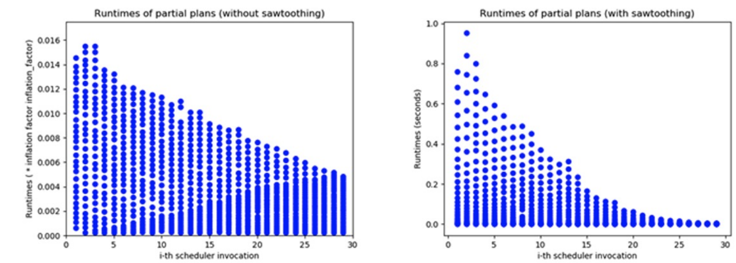
- In part driven by the complexity model - the root finding algorithm was eliminated in favor of inverting a preheat table, producing constant time and guaranteeing termination
- The complexity model highlights that cumulative rate timelines are dominate the other aspects of the algorithm at $O(n^2)$.
- With this knowledge, there is a more complex algorithm that would speed up the cumulative rate timeline to $O(n)$
- Termination is proved by showing that each portion of the algorithm is bounded in input.



MARS 2020 PERSEVERANCE

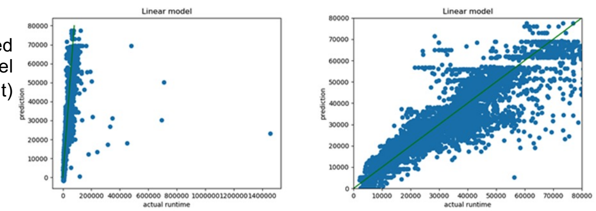
Trusted M2020 OBP – Informal Methods
Empirical Model of Scheduler runtime

- Developed an empirical model of onboard planner runtime*.
    - Uses parameters from scheduler call to estimate runtime
        - # of activities, preheat estimates, last invocation runtime, …
- Said model enables adaptive setting of the commit window to enable more effective rescheduling
    - However, gains were deemed not worth the added FSW complexity (incl. V+V) so the OBP as flown uses a static commit window
    - FE enables pulling forward activities when OBP completes rescheduling early

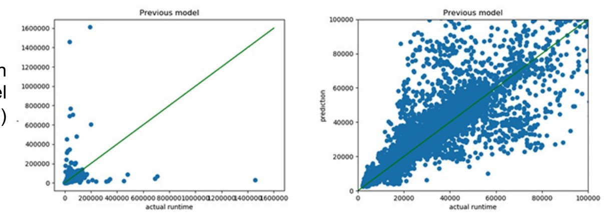


Runtimes decrease later in the plan with fewer activities to schedule

Linear feature-based predictive model (zoom in at right)



Prior invocation runtime model (zoom in at right)

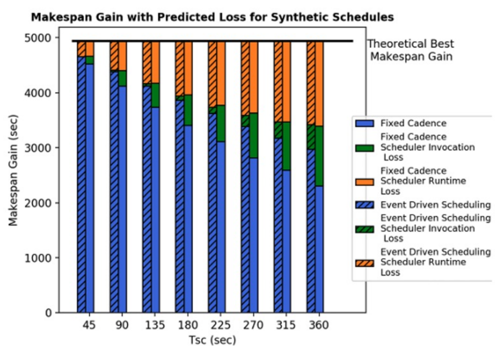


* - Bhaskaran, S.; Agrawal, J.; and Chien, S. Using a Model of Scheduler Runtime to Improve the Effectiveness of Scheduling Embedded in Execution. In Workshop on Integrated Execution (IntEx) / Goal Reasoning (GR), International Conference on Automated Planning and Scheduling (ICAPS IntEx/GP 2020), October 2020.

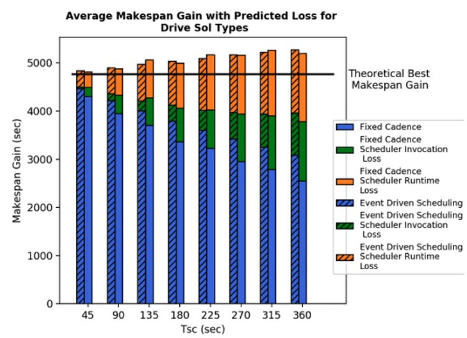
MARS 2020 PERSEVERANCE

Trusted M2020 OBP – Informal Methods
Analysis of FE and Scheduling Invocation methods

- Developed an analytical model scheduler invocation and FE makespan gains [Agrawal et al. 2021]
  - This analysis supported the design decision for event driven rescheduling
  - This analysis supported some aspects of FE design



(b) Synthetic schedules are entirely serial: Makespan gain + predicted loss = theoretical best makespan gain.



(b) Makespan gain + predicted loss ≠ theoretical best makespan gain (if we had an instantaneous scheduler). Real schedules are not all perfectly serial and have other execution constraints (e.g. preheats).

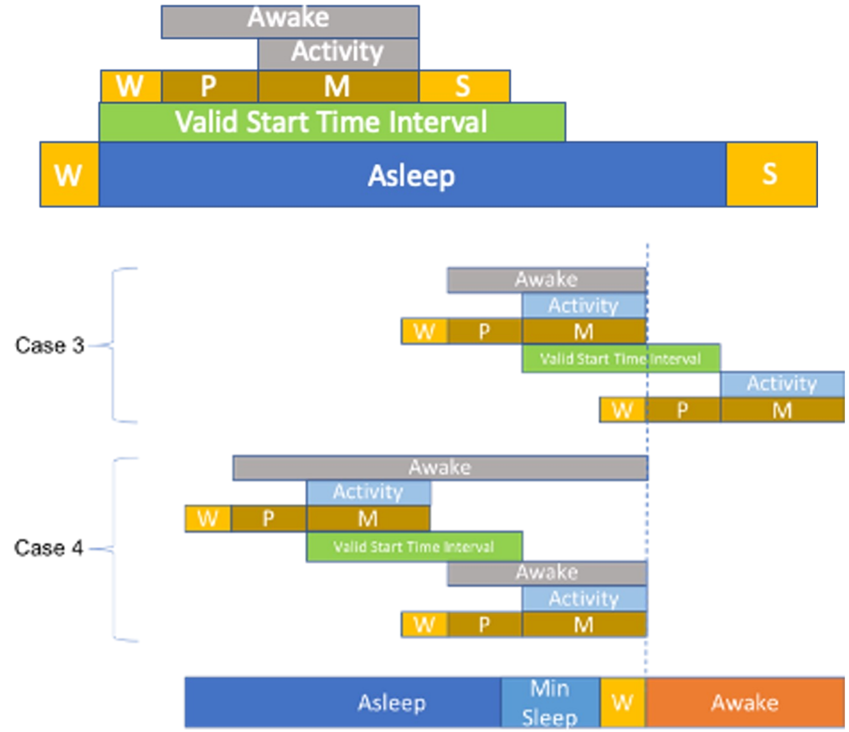
See:

Agrawal, J.; Chi, W.; Chien, S. A.; Rabideau, G.; Gaines, D.; and Kuhn, S. Analyzing the Effectiveness of Rescheduling and Flexible Execution Methods to Address Uncertainty in Execution Duration for a Planetary Rover. *Robotics and Autonomous Systems*, 140 (2021) 103758. 2021.

Trusted M2020 OBP – Informal Methods Analysis of Energy Scheduling Algorithms

- Analyzed three competing methods of implementing energy constraints in scheduling [Chi et al. 2020]:
  - Max Duration
  - Probe
  - Linear

- This analysis was used to support the design decision to implement the heuristic probe algorithm for energy scheduling in the OBP



* - Chi, W.; S.Chien; and Agrawal, J. Scheduling with Complex Consumptive Resources for a Planetary Rover. In *International Conference on Automated Planning and Scheduling (ICAPS 2020)*, Nancy, France, October 2020.

MARS 2020 PERSEVERANCE

Developing Trusted Autonomy: SW Development - Build

M2020 FSW Processes → OBP

- Informal methods in software development includes:
  - code walkthroughs,
  - coding guidelines,
  - coding rules (see MSL process*)
  - design reviews, and
  - software documentation.

- Formal methods in software development includes:
  - use of static code analyzers (CodeSonar) as part of the M2020 software development process
  - runtime analyzers as part of unit test: Valgrind (memory, performance), AddressSanitizer (Memory)

- Testing
  - Unit test coverage analysis via Gcov

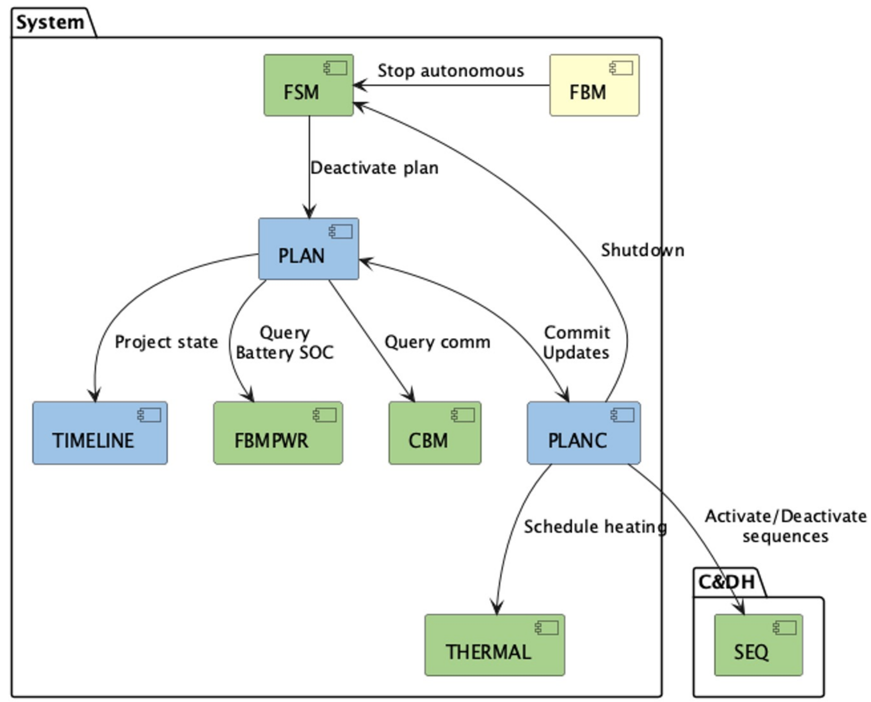
* Gerard J. Holzmann, “Mars Code,” Communications of the ACM, February 2014, Vol. 57 No. 2, Pages 64-73
10.1145/2560217.2560218

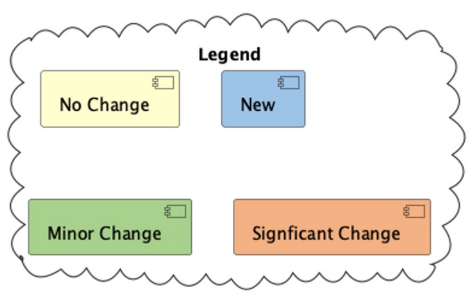
Developing Trusted Autonomy: Test Campaign

M2020 OBP Testing - Test Challenges

Challenge: Ensure system is and *will remain* safe in dynamic environment with OBP-in-control

- Complex **interfaces** with critical health and safety subsystems
  - Sequence control
  - Wakeup/shutdown control
  - Thermal control
  - Battery SOC estimator
  - Comm window manager
  - System fault protection



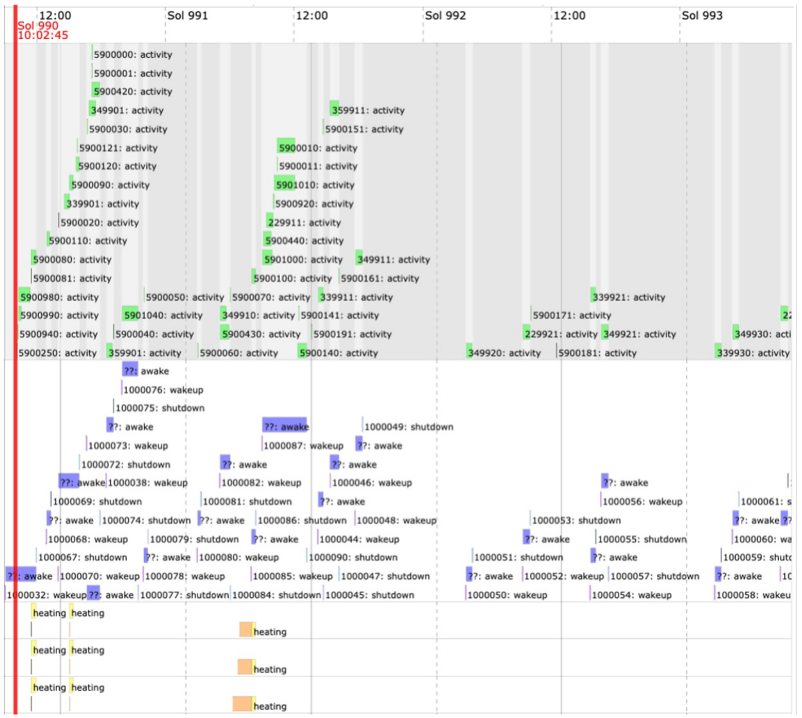


MARS 2020 PERSEVERANCE

M2020 OBP Testing - Test Challenges

Challenge: Flexible scheduling/execution can lead to highly variable outcomes

- Schedule structure **sensitive to variation**
  - Activity duration
  - Activity completion status
  - Hardware temperatures
  - Energy available
  - Time available
- Periodic rescheduling creates many **branch points**
  - Schedule revised hourly while awake
  - Execution deviations trigger scheduling
- Numerous possible execution outcomes
  - Need to constrain to finite number of tests



MARS 2020 PERSEVERANCE

M2020 OBP Testing - Test Campaign Overview

Opportunity: Deploy OBP during stable surface operations campaign

- Test campaign performed after ~500 sols of Master/Submaster operations
- Extensive Mars surface operations experience within V+V team
- Regular interfacing with mission system through Simple Planner Working Group

Approach: Release capabilities in phases to allow parallel V+V and operations

Test Strategy: Verify specific capabilities in flight-like scenarios where practical

- Primary focus: **Functional Testing** (bottom-up)
  - Targeted individual capabilities
  - Verified L3 and L5 requirements and SWG (software guidance) artifacts
- Secondary focus: **Scenario Testing** (top-down)
  - Transformed as-flown Master/Submaster plans into OBP plans

MARS 2020 PERSEVERANCE

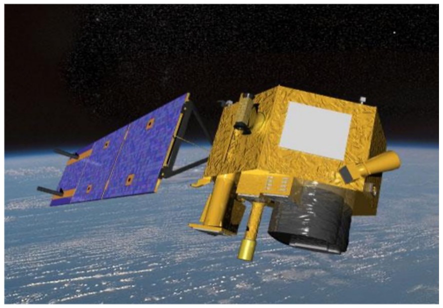
M2020 OBP Testing - Contrast with Prior Test Campaigns

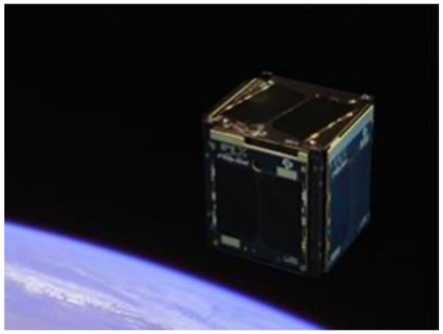
Spacecraft autonomy test strategies on EO-1[1] and IPEX[2] were largely scenario-focused

- **Baseline-Scenario**: Expected sequence/parameter values
- **Stochastic**: Varying relevant parameter values
- **Environmental**: Extension of stochastic set including execution variation points

Scenario-driven testing not sufficient for Mars 2020's Onboard Planner

- Mars 2020 vehicle and mission **significantly more complex** than EO-1 and IPEX
- Martian environment **more variable** than orbital environment
- Rover activities **more unpredictable** than orbiter activities





[1] - Cichy, B.; Chien, S.; Schaffer, S.; Tran, D.; Rabideau, G.; and Sherwood, R. Validating the Autonomous EO-1 Science Agent. In International Workshop on Planning and Scheduling for Space (IWPSS 2004), Darmstadt, Germany, June 2004.
[2] - Chien, S.; Doubleday, J.; Thompson, D. R.; Wagstaff, K.; Bellardo, J.; Francis, C.; Baumgarten, E.; Williams, A.; Yee, E.; Stanton, E.; and Piug-Suari, J. Onboard Autonomy on the Intelligent Payload EXperiment (IPEX) CubeSat Mission. Journal of Aerospace Information Systems (JAIS). April 2016.

M2020 OBP Testing - Functional Testing

Unbounded problem space must be discretized into finite number of tests

- Developed tests targeting **individual capabilities**
  - Plan Activation and Handovers
  - Resource Management (time, power, energy, activity resources)
  - Inter-Activity Dependencies
  - Heating
  - Wakeup and Shutdown
  - Activity Execution
  - Performance and Timeliness
  - Fault Protection Interaction
- Grouped capabilities by **release phase**
  - SP1: basic capabilities needed for Simple Planner SP1 rollout
  - SP2: extended capabilities developed/tested in parallel with SP1 operations

Test cases crafted to reflect expected **flight scenarios**, where practical

- Simple Planner Working Group discussions informed expected flight use-cases

M2020 OBP Testing - Scenario Testing

Rich set of flight plans available after ~500 sols of M/SM surface operations

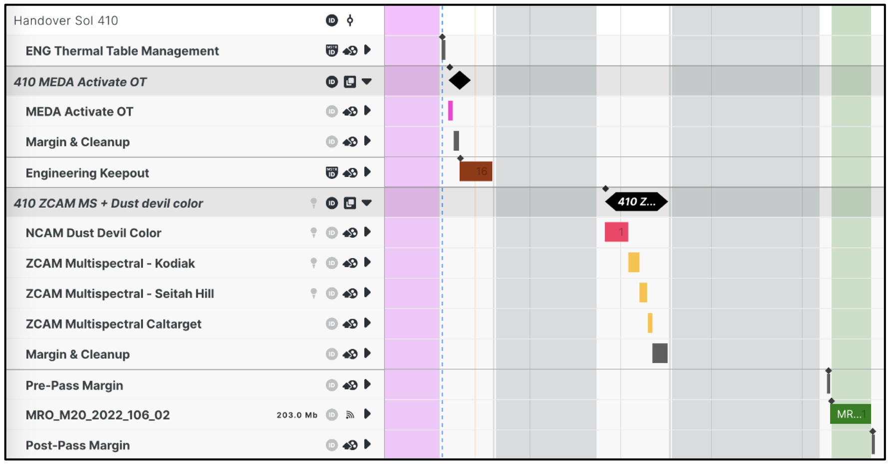
- Plan types: *Drive*, *Remote Science*, *Proximity Science*, and *Sampling*
- Trending data on activity execution durations, thermal conditions, battery state of charge

Plan transforms enabled early *Build phase* testing of OBP

- V&V team had early access to sandbox builds, enabling agile *Build - Test* iteration

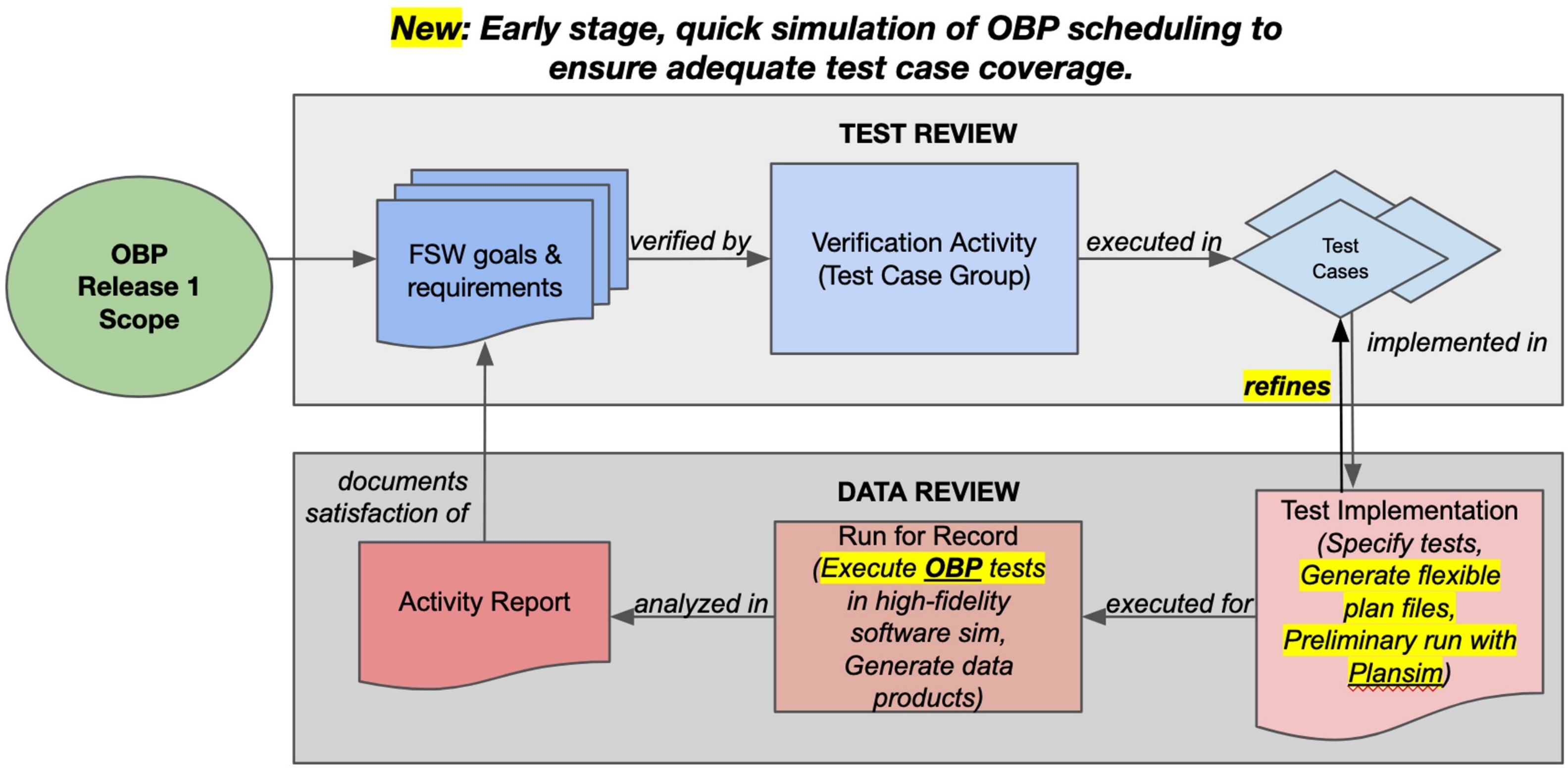
Formal Monte Carlo testing descoped

- Superseded by having real flight data/plans
- Would have been critical at landing



MARS 2020 PERSEVERANCE

M2020 OBP Testing - Test Cycle



M2020 OBP Testing - Test Venues

Fast, Lightweight

↕

High fidelity

	OBP FSW	Other FSW	Rad750	Battery	Thermal	Mechanisms	Instruments
Plansim	Real	Sim	Sim	Sim	Sim	Sim	Sim
WSTS	Real	Real	Sim	Sim	Sim	Sim	Sim
MSTB	Real	Real	Real	Sim	Sim	Sim	Sim
VSTB	Real	Real	Real	Real	Real	Real	Real

- ***Plansim***: New, light-weight OBP simulation tool
  - OBP FSW with simulated interfaces and environmental conditions
  - Enabled rapid test case development
- ***WSTS*** (Work Station Test Set):
  - 8x real-time, time jumps, parallel execution
  - Bulk of run-for-record OBP V+V performed here
- ***MSTB*** (Mission System Testbed):
  - Run-for-record V+V performed here when real flight computer and avionics required
- ***VSTB*** (Vehicle System Testbed):
  - Utilized for system integration tests running real activities (driving, imaging, proximity science, etc.)

MARS 2020 PERSEVERANCE

M2020 OBP Testing - Test Automation

Developing, executing, and reviewing such a large number of test cases (often spanning multiple sols each with significant idle time for operator) **required extensive automation**

- *Jupyter Notebook* (open-source computational environment) utilized to configure, execute, and analyze test cases
- *Python* library code developed to automate all aspects of testing
    - Command product generation
    - Venue configuration
    - Test case configuration
    - Test case execution
    - Test case cleanup
    - Schedule visualization
    - Telemetry analysis
    - Venue cleanup
- Enabled fire-and-forget execution of entire VAs

4.6.2 Execution

In [144]:

```
with obp.Execute(tc) as e:
    # Activate plan file
    base.set_zone_temp(zoneC1,-40)
    base.set_zone_temp(zoneD1,-40)
    # Activate plan file
    base.activate_plan(fname,Execute_obj=e)
    # Set C1 zone to something over the setpoint
    base.wait_until(sclk=sclkC - 60)
    base.set_zone_temp(zoneC1,20)
    # Set D1 zone to something over the setpoint
    base.wait_until(sclk=sclkD - 60)
    base.set_zone_temp(zoneD1,20)
    # Wait for the plan to finish
    base.wait_plan_finished()
```

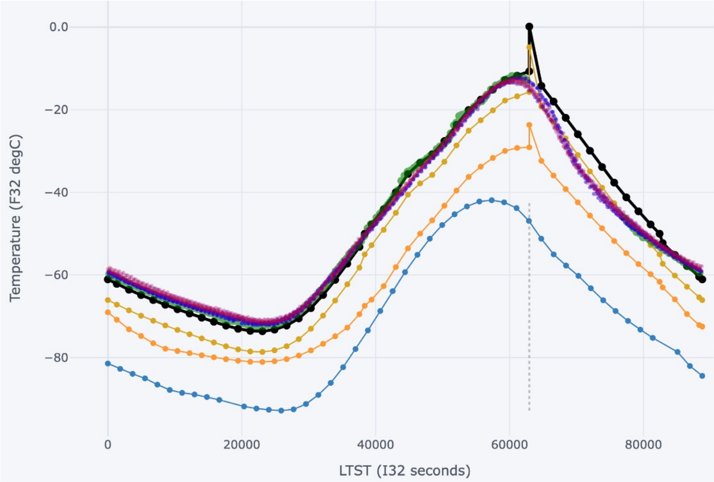
M2020 OBP Testing- Thermal Configuration

The OBP-Thermal interface is complex, with many special cases that increase uncertainty

- Heating extension or discretization across multiple activities
- Handoff of thermal control between flight computer and Remote Engineering Unit (REU) when heating overlaps sleep cycles
- Periods of day where devices not able to be heated
- Incomplete preheat yields uncertainty of how much heat has permeated the thermal zone

Approach:

- Temperature poking (both at sensor and REU sim)
  - Diurnal thermal cycles
  - Preheat ramp curve
- Modification of prescription tables
  - Elicit specific preheat scheduling behaviors
- Re-configuring parameters and device health



MARS 2020 PERSEVERANCE

M2020 OBP Testing - System Visibility

Need to balance **visibility** into OBP subsystem against the impact to downlink **data volume**

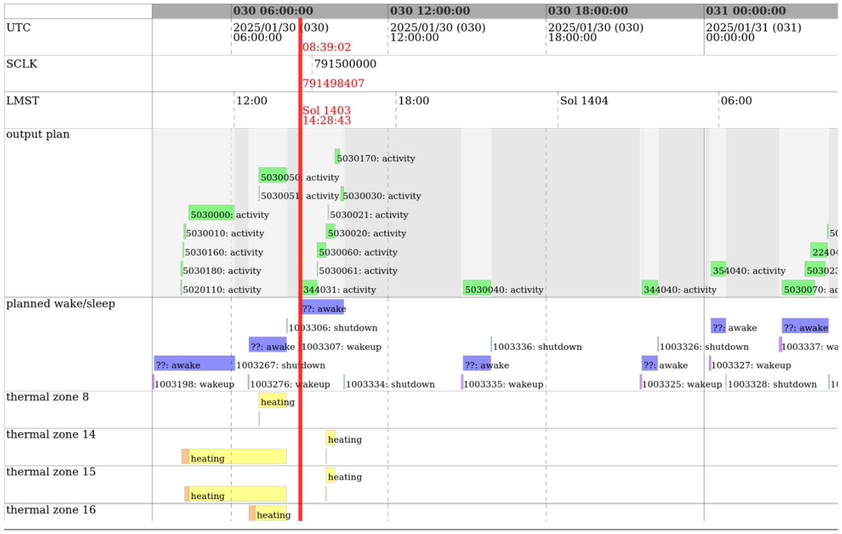
- Complex decision-making during scheduling
- Many iterations of schedule within a single scheduling cycle
- Many scheduling cycles per sol (observed min 5, max 55, mean ~16 in flight)

Key telemetry: **PlanSummary** data product

- Snapshot of entire schedule, down to activity-level schedule/executions states
- Includes initial and final conditions

test_gen: Diagnostic tool that reruns OBP scheduling code against flight initial conditions

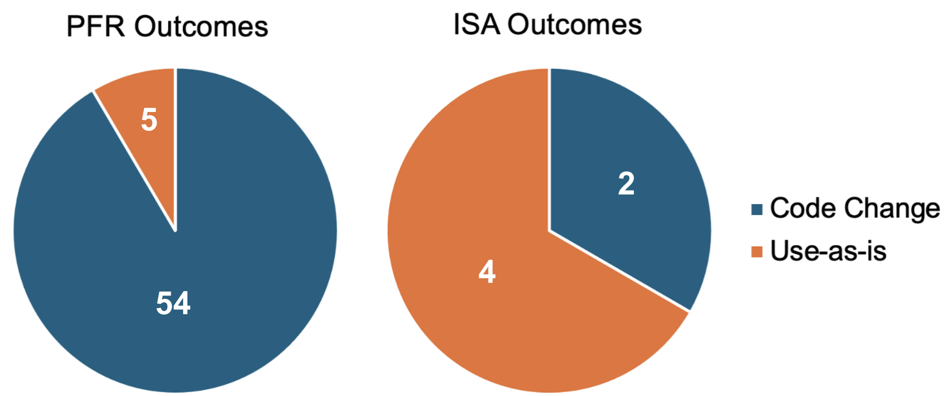
- Enables full trace diagnostic visibility into scheduling steps
- Reconstructs flight timeline models (energy, peak power, resource claims, etc.)



M2020 OBP Testing - Results

Verification Activities Completed	35
Test Cases Completed	182
Requirements/SWGs Passed	139
OBP PFRs[1] opened	59
OBP ISAs[2]	6

1 - "Problem Failure Report": Discovered in testing
2 - "Incident, Surprise Anomaly": Discovered in flight



DNG ID	Title	SURF Closure Activity	VnV Status	VA 330318	VA 330324	VA 330326	VA 330327	VA 610349	VA 880091	VA 881472
72008	REQ: L3FS - Autonomous Activity Execution	OBP_SP1	Passed for SURF			Passed				
72010	REQ: L3FS - Autonomous Activity Termination	OBP_SP1	Passed for SURF				Passed			
72012	REQ: L3FS - On-Board State of Charge Estimation	OBP_SP1	Passed for SP1							Passed
72015	REQ: L3FS - Activity Completion Status Telemetry	OBP_SP1	Passed for SURF							
326644	SWG: UHF Pass Activity	OBP_SP1	Passed for SURF		Passed					
326646	SWG: X-band pass activity	OBP_SP1	Passed for SURF		Passed					
326647	SWG: Plan Attributes	OBP_SP1	Passed for SURF		Passed					
326649	SWG: Parameterized plan constraints	OBP_SP1	Passed for SP1				Passed			
326677	SWG: Preheating activity	OBP_SP1	Passed for SURF							
326678	SWG: Maintenance heating activity	OBP_SP1	Passed for SURF							
326680	SWG: Activity constraints	OBP_SP1	Passed for SP1							
326681	SWG: OBP summary report	OBP_SP1	Passed for SP1		Passed					
326686	SWG: Activity dependency constraint	OBP_SP1	Passed for SP1							
326688	SWG: FS Resource Constraint	OBP_SP1	Passed for SURF			Passed				
326692	SWG: Evaluation of activity execution	OBP_SP1	Passed for SP1		Passed					

MARS 2020 PERSEVERANCE

Trusted M2020 OBP - Beyond the Test Campaign

Super Thread Tests:

- Performed on *VSTB* July and November of 2022
- Partial validation of OBP use in operations

First-Time Activities (FTA):

- Performed on *Perseverance* May - July 2023
- 3 toe-dips, demonstrating basic capabilities

Operational Readiness Test (ORT):

- Performed on *VSTB* August of 2023
- Final validation of OBP use in operations

These will be covered in greater detail at:

Simple Planner Rollout
Nick Waldram
12 noon Pickering
18th March 2025.

MARS 2020
PERSEVERANCE

Discussion

- Onboard Planner represents a *paradigm shift* in that the flight system has considerable latitude to alter execution

- OBP team embraced *full life cycle* development of trusted autonomy
  - Informal methods
    - Termination and complexity analysis of scheduler; empirical analysis of runtime
    - Prototyping of alternative algorithms
    - Estimates of V+V effort for various algorithms → design decisions
  - Software practices - informal and formal methods
  - Continuous emphasis of going after biggest, lowest hanging fruit of productivity gains

- OBP testing campaign proved critical to infusing autonomy for a flagship mission
  - Heavy reliance on functional tests informed by flight use-cases
  - Less reliance on scenario-based testing than EO-1 and IPEX
  - Leveraged hundreds of actual plans run on Mars

- Very few anomalies (Flight Software or Operations) have been encountered in flight, validating the trust that operators have placed in this new system

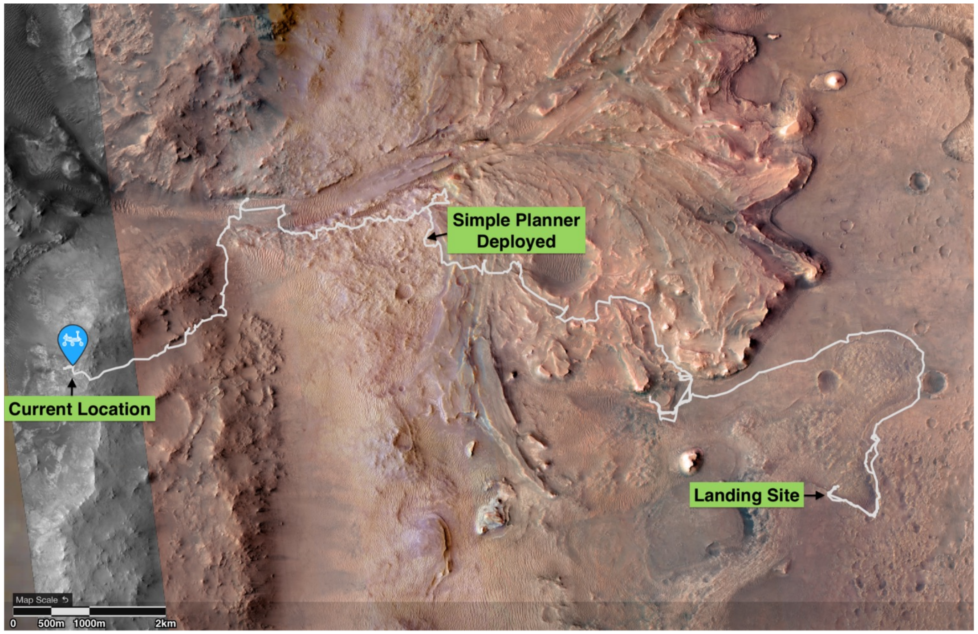
MARS 2020
PERSEVERANCE

Closing Remarks

- Simple Planner has been the baseline for M2020 operations since rollout in October 2023
  - Anomalies have not required operations to revert to Master/Submaster paradigm

- As of 29 January 2025, OBP has executed **257** plans covering **429** sols on Mars:
  - **6917** onboard scheduling cycles
  - **7810** user activities executed
  - **13** km driven
  - **70,000**+ images acquired
  - **4** rock core samples acquired

- The M2020 Simple Planner approach to Trusted Autonomy has been overwhelmingly successful!



MARS 2020 PERSEVERANCE

Acknowledgements

OBP FSW Team

- Dan Gaines
- Gregg Rabideau
- Eddie Benowitz
- Vincent Wong
- Amruta Yelamanchili

Funding

- Mars 2020 Project
- Europa Lander Feed-Forward
- 5x R&TD
- Directors Innovation Fund
- OTIS

OBP V+V Team

- Dan Gaines
- Stephen Kuhn
- Elyse Moffi
- Shreya Parjan
- Kevin Reich
- Ansel Rothstein-Dowden
- Danny Tran
- Nick Waldram
- Sean Wenzel

5x R&TD Team

- Steve Chien
- Wayne Chi
- Jagriti Agrawal
- Sarah Bhaskaran

M2020 Operations Team

MARS 2020 PERSEVERANCE

Jet Propulsion Laboratory
California Institute of Technology

jpl.nasa.gov