ONBOARD SCHEDULING AND EXECUTION TO ADDRESS UNCERTAINTY FOR A PLANETARY LANDER

Steve Chien, Jean-Pierre de la Croix, Joe Russino, Caleb Wagner, Gregg Rabideau, Daniel Wang, Grace Lim, Dustin Aguilar, Yuliya Marchetti, Scott Tepsuporn, Philip Twu, Cecilia Mauceri, Grace Tan-Wang, and Glenn Reeves

Jet Propulsion Laboratory[*]
California Institute for Technology
4800 Oak Grove Drive
Pasadena, CA 91109-8099 USA

ABSTRACT

The Europa Lander Mission Concept presents a number of challenges. The current mission concept would land with a fixed amount of energy, an expected mission lifetime of approximately 30 days, and would be able to communicate with the Earth in less than 42 out of every 84 hours due to the Europa-Jupiter orbit. Additionally, planned activities such as trenching and sampling, will interact significantly with a largely unknown environment and therefore may encounter failure or highly variable duration or energy consumption when executing. All of these factors present challenges for a conventional ground operations paradigm.

We describe advanced prototyping of onboard autonomy software to address such challenges. Onboard event driven execution and rescheduling offer promise to enable the lander to adjust to execution feedback without incurring costly ground interaction. We describe prototyping with the TRACE execution system and the Mexec scheduling and execution system. We show how these autonomy systems can address: uncertainty in domain modeling, stochasticity at execution time, and the presence of exogenous events.

Key words: surface mission autonomy; artificial intelligence; scheduling; task execution; flight software; operations.

1. INTRODUCTION

Autonomy for robotic applications often must address variation in execution and uncertainty in the quality of environment models. In space-based applications, this can be especially challenging when the environment is largely unknown, reducing the quality of our *a priori* models of the world. To address these problems, we describe an integrated approach to planning and execution in an unknown, unpredictable environment. We describe two system level autonomy frameworks for scheduling and execution .

The primary empirical context of our model is a mission concept to perform *in situ* analysis of samples from the surface of the Jovian moon Europa [Han17]. Unlike prior NASA missions, *a priori* domain knowledge is severely limited and uncertain, and communication with Earth is limited by long blackout periods (over 42 hours out of every 84 hours). Consequently, a successful mission requires a planning and execution framework that can operate autonomously for extended periods of time, is robust to unprecedented levels of uncertainty, and is still capable of maximizing its overall utility. Additionally, because of the harsh radiation environment at Europa, mission lifetime and onboard computing are severely limited [1].

On the other hand, the Europa Lander concept has a fairly rigid definition of what actions the lander must perform in order to produce utility. The ultimate goal for a Europa Lander would be to analyze surface material and communicate the resulting data products back to Earth. To reward accomplishment of these goals, we assign utility to tasks such as sample excavation and seismographic data collection, but the overwhelming majority of the mission utility is not awarded until the lander communicates the data down to Earth.

Our prototyping with the TRACE execution system [dlCL20] leverages this structure. TRACE represents task networks in the BPMN process notation to enforce operations constraints and achieve Europa Lander Mission Concept goals.

---

[*]

[1] As a point of reference, the RAD750 processor used by the Mars 2020 rover Perseverance has measured performance in the 200-300 MIPS range. In comparison, a 2016 Intel Core i7 measured over 300,000 MIPS, or over 1000 times faster. Furthermore, autonomy software would only be allocated a portion of the computing cycles onboard the flight processor resulting computation *several thousand times* slower than a typical laptop.

Our prototyping with the Mexec [TMH$^+$20, WRBC22] planning algorithm leverages this domain-specific knowledge by making use of a hierarchical task network (HTN) and using heuristic-guided search to examine various task combinations to maximize utility. In the HTN framework, this means that tasks in a hierarchy produce very little utility until the full hierarchy is executed.

In the remainder of this paper, we first describe the Europa Lander Mission Concept. We then describe the TRACE execution and Mexec scheduling and execution software systems. We then discuss the current status of the prototypes. We then discuss related and future work and conclusions.

2. DOMAIN DESCRIPTION

The primary goal of the Europa Lander mission concept is to excavate and sample the surface, analyze the sampled material for signs of biosignatures, and communicate that data back to Earth [Han17]. Additionally, there are secondary objectives to take panoramic imagery of the Europan surface and collect seismographic data. Lander operations are generally limited to the accomplishment of these two overarching goals. This provides significant structure to the problem, since the concept mission clearly defines the sequence of actions required to achieve these goals. Figure 1 displays the strong dependency structure inherent to the Europa Lander concept mission. In order to sample, the lander needs to have excavated a trench; in order to analyze, the lander needs to have collected a sample; etc.

As a minimum requirement, the lander should excavate a trench in the Europan surface, collect three samples from that site, analyze those samples, and return that data to Earth. The basic requirements of a mission would require only a single site to be excavated. However, there is value in excavating additional sites, because the material at different sites may possess different properties. On the other hand, the lander may choose to resample the same location, for example, in order to verify the discovery of a biosignature. In the baseline mission concept, all three of the lander's samples are chosen from the same target. Note that after the first site is excavated, no further excavations are needed to sample from that trench; all three sampling activities can share a single excavation site. After excavation and sample collection, samples must be transferred into scientific instruments that analyze the material and produce data products. Then, for a mission to achieve any actual utility, those data products must be communicated back to Earth.

In addition to sampling tasks, the lander may engage in seismographic data collection and period panoramic imagery tasks. These are considered lesser goals, with lower utility associated with their completion. As such, the data products that these tasks generate are considered to have lower value. However, these tasks also involve no surface interaction, and have less uncertainty associated with them as a result.

It is important to note that primary utility is only achieved when data is downlinked back to Earth [2]. This is true for both the sampling and seismograph/panorama tasks. Some excavation sites or sampling targets may provide more utility than others if, for example, one of those targets has a positive biosignature and the other does not. However, regardless of the quality of the material that the lander samples, no utility is achieved unless that data is communicated. This dynamic means that while potential utility is generated during the sampling and analysis phases, it is only realized by completing relevant communication tasks.

The Europa Lander mission concept is also constrained by a finite battery that cannot be recharged. Battery life is a depletable resource, and the lander must use its energy as efficiently as possible. Each task saps energy from the battery, and our algorithm must plan accordingly to maximize utility in face of this constraint. In addition to this challenge, the surface characteristics of Europa are uncertain, and any prior mission model that is generated before landing is sure to have inaccuracies. In particular, the energy consumption of the excavation and sample collection tasks is largely unknown. There is also significant variation in the utility of any given sample, since the value of sampling a given target on Europa depends on whether the material is scientifically interesting, e.g. whether a biosignature is present.

3. THE TRACE EXECUTIVE

TRACE (Traceable Robotic Activity Composer and Executive) is a tool designed to holistically address the modeling, verification, and execution of planned, opportunistic, and contingency activities during robotic missions from an event-driven execution perspective.

The Business Process Modeling and Notation (BPMN) language is a standard maintained by the Object Management Group that graphically models business processes [Von]. Elements within BPMN include events based on state or data, tasks for humans or automation, logical gateways for process flows, and pools and lanes for organizing resources (see reference guide for more details [Cam20]). TRACE tailors BPMN to the robotics domain. Automated tasks, like service tasks, correspond to robotic activities, like navigation or grasping an instrument with a robotic arm. Data-driven elements, like conditional events or exclusive gateways, use system data to allow the executive to make decisions on how to flow through the mission.

TRACE follows the BPMN rule set and integrates these elements with the robotic system. Consequently, this standardized language can be used to model a sequence

---

[2] While context imagery while preparing sampling locations has some science value, the primary science value is from analysis of the samples.

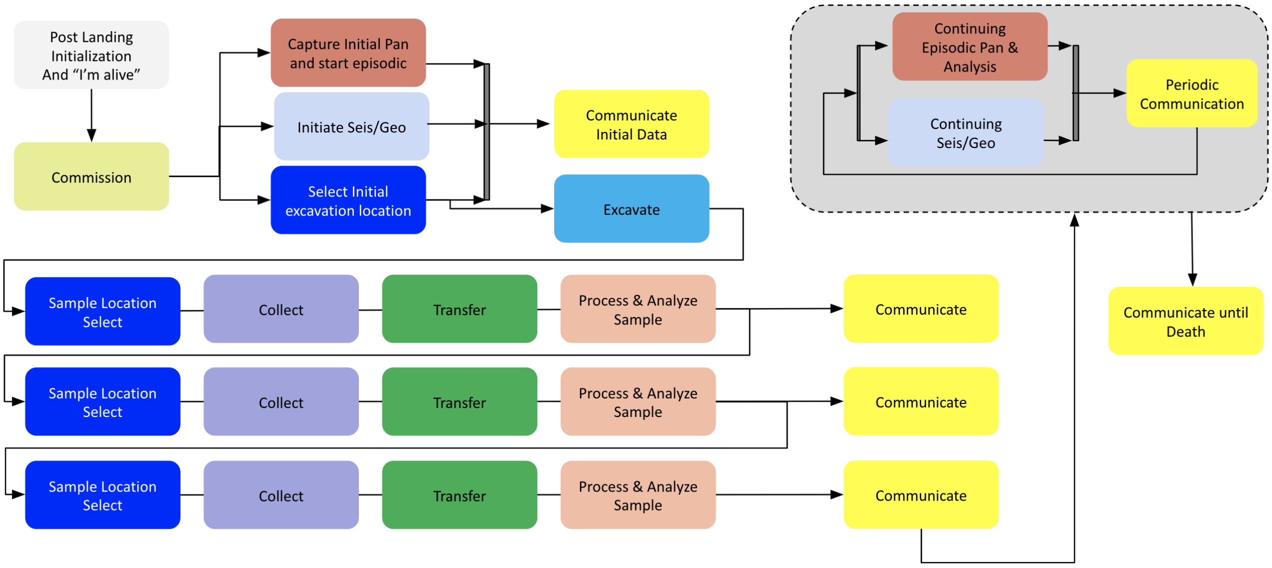

Figure 1. A task network for the Europa Lander mission concept. The diagram represents a potential execution trace of the mission that would fulfill baseline requirements.

of robotic activities tied together by logical constructs, events, and time. A BPMN mission model encodes the sequence of activities for the mission. The BPMN event constructs provide a way to encode responses to opportunities (e.g., detection of a new, interesting science tar-get) or non-deterministic events (e.g., a subsystem fault) and diverge the process flow to a secondary process that contains a sequence of alternate activities. These contingent (or opportunistic) activities can be modeled to execute in parallel to planned activities (e.g., start up an instrument) or to interrupt and then resume planned activities once completed. Since TRACE's modeling tool (composer) allows a large degree of flexibility in designing the mission, TRACE provides verification tools to ensure feasibility.

Specifically, the BPMN mission models can be translated into the Process Meta Language (PROMELA) and in-put into the Spin Model Checker with a set of mission requirements expressed in Linear Temporal Logic (LTL). Consequently, TRACE is able to verify if the mission completes (e.g., no dead locks, no starvation) and if it satisfies the mission requirements (e.g., per-form science measurements at least k times) [dlCLH$^+$17].

Lastly, verified mission models are then executed by TRACE's executive, which integrates into the autonomy subsystem via a connector. The executive accesses data (e.g., sensors, system health) and invokes subroutines (e.g., navigate to coordinate) according to the mission model by interpreting it at runtime. Consequently, TRACE is a potential end-to-end solution for flight missions by providing integrated tools for modeling, verification, and execution of planned, opportunistic, and contingency robotic activities. The TRACE executive handles planned excavation and science activities, as well as, contingency activities in response to subsystem faults.

4. THE MEXEC SCHEDULER AND EXECUTIVE

We are also prototyping system level autonomy using MEXEC [TMH$^+$20], an integrated planner and executive originally built for NASA's Europa Clipper mission [VGR$^+$17]. Using Mexec we compare four approaches to planning on the Europa Lander problem [WRBC22] similar to those used in prior missions: a static plan without failure recovery mechanisms, a static plan with ground input for failure recovery (similar to current Mars Rover operations) [Gea16], flexible execution without replanning, and flexible execution with replanning optimization. We explore the value of onboard autonomy: flexible execution and replanning with plan optimization, and examine these techniques' effects on utility in these scenarios. We demonstrate that, true to our model's prediction, each technique shows significant improvement in utility achievement in the Europa Lander domain.

We design our planning system to respond intelligently to stochasticity at execution time, since we expect this to be a significant factor in our domain. Planning and execution are integrated in our approach, in order to respond to variation and therefore better optimize overall utility achieved. We achieve this integration through the use of two techniques: flexible execution and replanning with plan optimization.

4.1. Flexible Execution

Flexible execution is a lightweight rescheduling algorithm that runs at a much higher cadence than the planner. This algorithm has two main properties: (1) it is much less computationally demanding than replanning as it does not search, and (2) it is less capable than replanning. Flexible execution allows the system to handle less-severe unexpected events without incurring the cost of replanning. The Mars 2020 Perseverance rover has plans to deploy flexible execution [ACC+21a]. Our implementation differs in focus, emphasizing responses to adverse events.

In our system, flexible execution consists of two major components. The first is *task push*. If a task's preconditions are not met, before failing the task, we allow it to wait for some amount of time for this inconsistency to resolve. Such a situation might occur, for example, if required preceding activities are delayed or run long. The executive checks the task's preconditions and delays dispatch until either the conditions have been met, or the task's wait timeout has been exceeded.

The second component of flexible execution is *automated retry*. If a task completes with a failure code, flexible execution can immediately re-schedule the task if its preconditions are still met (and plan updated with new predicted end time of the task and resource usage), avoiding replanning cost and delay.

In the context of the Europa Lander domain, flexible execution offers significant value because many robotic tasks such as trenching and sample acquisition can vary significantly in duration and and hence resource consumption. Flexible execution handles this variation without disrupting the execution flow.

4.2. Replanning with Plan Optimization

For more complex execution variations, we turn to replanning during execution. Replanning uses search to construct the plan based on the current state. The current state includes: state and resource values (accounting for failed activities, and resource under/over runs) as well as the current time (accounting for execution time variation). Additionally, utility estimation, predicted duration, and predicted resource expenditures for future tasks may be updated (e.g. imaging may indicate that a sampling site now looks more promising and/or may take longer/shorter to excavate or sample). Replanning enables incorporation of all of this new information into decision-making and searches the space of possible plans to optimize utility [BWCZ21].

We model this problem using a hierarchical task network (HTN) to compile the domain-specific knowledge of the dependency structure into the task network. HTNs have been used successfully in industrial and other real-world applications to improve the tractability of planning problems in systems such as SHOP2 [NAI+03] and SHOP3 [GK19]. In an HTN, hierarchical tasks are decomposed to a set of subtasks. We refer to the higher-level tasks as "parent tasks", and refer to their children as "subtasks". Parent tasks may decompose into a number of different sets of subtasks; we refer to each of these sets as a potential "decomposition" of that parent task. Finally, we refer to tasks with no decompositions as "primitive tasks". These primitive tasks represent tasks that the lander can be directly commanded to perform.

4.3. Proactively planning for Robustness to Uncertainty

The above approaches to uncertainty are reactive in that they adjust execution (or reschedule) in response to execution variations. An alternative approach is to proactively plan/or execute in ways that are known to be resilient to environmental and execution variation. The approach outlined by Basich and others [BWCZ21] utilizes this approach. In this approach, plans are explicitly evaluated against against resource varying contingencies (specifically variations in available energy from predicted levels) and search bias it towards plans that perform best across the possible outcomes of more, less, or predicted energy consumption/availability. One would expect such approaches to have the potential to outperform reactive approaches to execution variation.

5. CURRENT STATUS OF THE PROTOTYPING

Iterative prototyping with TRACE and Mexec has been taking place since mid 2019. Initially, TRACE and Mexec adaptations for Europa Lander were prototyped as standalone units operating against stubs to simulate execution feedback. Later these were replaced by a Robot Operating System (ROS) [ROS22] infrastructure that provided a uniform interface to execution feedback (success, failure, duration, resources). In time this simulation has been replaced by a full fledged infrastructure that supports: links to robotic simulations (e.g. for robotic arm interactions and eventually actual hardware; vision simulations (and eventually hardware vision systems); and links to other functions of spacecraft flight software such as data management and downlink management. Key to the prototyping is the utility estimation module which estimates utility from future activities (e.g. computes priors for estimated utility of sampling at a location) as well as for a range of data products from executed activities (e.g. send down a summary, intermediate, or full data products from analysis of a collected sample).

Current prototyping efforts are shifting from adding functionality to evaluation and study of strengths and challenges of the autonomy prototype. From an execution and scheduling standpoint, study is focusing on the relative challenges of encoding the domain primarily in an execution system such as TRACE versus in a hybrid execution and rescheduling approach such as in Mexec. TRACE allows a more direct encoding of expert domain knowledge from task networks into BPMN. Mission and Operations staff already have some familiarity with task networks facilitating encoding, trust, and validation. Introducing a scheduling capability as in Mexec enables a more natural handling of decision-making that requires temporal projection and/or interaction with exogenous events (e.g. only start a new excavation if is is projected to complete before the next downlink). However understanding and validating scheduler behavior is more challenging.

Current prototyping is focused on the team understanding system performance as well as how to adapt autonomy engines such as TRACE and Mexec to accomplish mission goals over a range of scenarios including:

- variations in energy projected (and executed) to complete trenching (and sampling),
- variations in utility values for sample sites and data products,
- variations in data product size,
- variations in ground in the loop operations paradigm (every sample, first sample, none), and
- variations in communications system efficiency.

These scenarios and many others are being studied to better understand the challenges in deploying an operational onboard autonomy system for the Europa Lander Mission Concept.

In parallel to this effort, design simulation using the Blackbird mission planning tool [LRK+20] is performing mission trades and analyses [KRY+21]. These trades and analyses are performed in the Blackbird mission planning tool and model and are not integrated into higher fidelity simulations or hardware. This enables greater coverage of mission scenarios but prevents sensitivity analysis to individual events such as failures, execution variations, and utility variations.

6. RELATED AND FUTURE WORK

With respect to proactively handling uncertainty, decision-theoretic planning provides a formal model for reasoning about problems in which actions have stochastic outcomes or the agent has incomplete information about its environment [IJLM16, SZS17, ZWBM02]. The primary objective of decision-theoretic planning is to produce plans or policies that define the potential trajectories of actions that the agent may take which maximizes its expected utility, rather than maximizing or guaranteeing goal-reachability [BDH99]. A standard approach in decision-theoretic planning for modeling domains is to use a Markov decision process (MDP) [Bel57] when the agent knows the full evaluation of every state at each timestep, or a partially observable Markov decision process (POMDP) [Spa12] where this holds only for a subset of the variables that define the statespace.

However, several issues in spacecraft or rover operations complicate the use of said decision making models. First, these models traditionally do not support durative or concurrent actions, but rather assume that all actions are instantaneous and fully sequential in nature. Second, although there have been a number of approaches over the years aimed at improving the scalability of these approaches [GKPV03, WWZ17, YFG07], most algorithms that solve MDPs produce policies that account for all contingencies and provide actions for all states in the domain. This is generally impractical or impossible in spacecraft and rover operations where computational power is (often severely) limited, and more so in our problem where the battery is non-rechargeable and the domain model is expected to be modified repeatedly throughout the agent's operation.

Onboard planning and execution are of great interest to the space domain. Flexible execution of tasks is a central focus of execution engines like PLEXIL [VEJ+05].

In 1999, the Remote Agent Experiment flew onboard the Deep Space One Mission for two periods totalling about 48 hours [MNPW98, JMM+00]. This planner worked in concert with an executive [PGK+97, PGG+98] and performed batch planning of engineering activities such as optical navigation.

The Earth Observing One (EO-1) spacecraft [CST+05], which flew for over 12 years from 2004-2017, responded to dynamic scientific events using the CASPER planner (taking 10s of minutes to replan and) with the SCL executive. The flight and ground planners [CTR+10] both used a domain specific search algorithm that enforced a strict priority model over observations for a limited model of utility.

The Intelligent Payload Experiment (IPEX) [CDT+16] was a cubesat flight demonstration of high throughput onboard product generation for the intelligent payload module of the proposed HyspIRI mission concept [CSDM09]. IPEX used the CASPER onboard planner in concert with a linux shell-based task executive. IPEX flew for approximately fourteen months.

Flexible execution implemented in VML [GL08] has been used effectively on the Spitzer to handle failure or variable execution time to acquire guidestars. Flexible execution is used in a similar fashion on JWST [Zon20].

The M2020 Perseverance rover also plans to fly an onboard planner [RB17] to reduce lost productivity from following fixed time conservative plans [Gea16]. Like the planning approach we propose in this paper, the M2020 planning architecture also relies on rescheduling and flexible execution [ACC+21a], ground-based compilation [CAC+19], heuristics [CCA20], and very limited handling of planning contingencies [ACC+21b]. However, it uses a non-backtracking planner, which limits its ability to optimize plans and the M2020 flight software does not support utility discovery. Our work also takes a different focus, primarily examining the effects of task failure and considering integrated planning in the context of failure resolution. Finally, the Europa Lander mission concept has stronger drivers for mission autonomy than

M2020 due to lack of reliable a priori model parameters, the inability to recharge the battery, and the long communications blackout time windows.

7. CONCLUSIONS

We have described efforts to prototype autonomy software targeting the Europa Lander Mission Concept. This mission concept presents severe challenges for autonomy in that the current concept will: land with limited energy and mission lifetime, have limited ability to communicate with Earth (less than 42 out of every 84 hours), and will need to deal with significant execution uncertainty stemming from interaction with a largely unknown environment (activity failure, unpredictable time and energy required).

In order to explore onboard autonomy options, we are prototyping with the TRACE executive and the Mexec executive/scheduler. This prototyping is studying the challenges and performance gains from deploying such onboard autonomy capabilities. Additionally we are studying the range of operations strategies possible with such capabilities such as ground in the loop at all samples, first samples, or no samples and the effect of said strategies on mission return.

ACKNOWLEDGMENTS

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

REFERENCES

[ACC$^+$21a] 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.

[ACC$^+$21b] J. Agrawal, W. Chi, S. A. Chien, G. Rabideau, S. Kuhn, D. Gaines, T. Vaquero, and S. Bhaskaran. Enabling limited resource-bounded disjunction in scheduling. *Journal of Aerospace Information Systems*, 18:6:322–332, 2021.

[BDH99] Craig Boutilier, Thomas Dean, and Steve Hanks. Decision-theoretic planning: Structural assumptions and computational leverage. *Journal of Artificial Intelligence Research (JAIR)*, 11:1–94, 1999.

[Bel57] Richard Bellman. A Markovian decision process. *Journal of Mathematics and Mechanics*, pages 679–684, 1957.

[BWCZ21] Connor Basich, Daniel Wang, Steve Chien, and Shlomo Zilberstein. A sampling-based optimization approach to handling environmental uncertainty for a planetary lander. In *International Workshop on Planning and Scheduling for Space (IWPSS)*, July 2021.

[CAC$^+$19] W. Chi, J. Agrawal, S. Chien, E. Fosse, and U. Guduri. Optimizing Parameters for Uncertain Execution and Rescheduling Robustness. In *International Conference on Automated Planning and Scheduling (ICAPS 2019)*, Berkeley, California, USA, July 2019.

[Cam20] Camunda. Bpmn modeling reference, 2020.

[CCA20] 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.

[CDT$^+$16] S. Chien, J. Doubleday, D. R. Thompson, K. Wagstaff, J. Bellardo, C. Francis, E. Baumgarten, A. Williams, E. Yee, E. Stanton, and J. Piug-Suari. Onboard autonomy on the intelligent payload experiment (ipex) cubesat mission. *Journal of Aerospace Information Systems (JAIS)*, April 2016.

[CSDM09] Steve Chien, Dorothy Silverman, Ashley Gerard Davies, and Daniel Mandl. Onboard science processing concepts for the hyspiri mission. *IEEE Intelligent Systems*, 24(6):12–19, 2009.

[CST$^+$05] Steve Chien, Rob Sherwood, Daniel Tran, Benjamin Cichy, Gregg Rabideau, Rebecca Castano, Ashley Davis, Dan Mandl, Stuart Frye, Bruce Trout, et al. Using autonomy flight software to improve science return on Earth Observing One. *Journal of Aerospace Computing, Information, and Communication*, 2(4):196–216, 2005.

[CTR$^+$10] Steve Chien, Daniel Tran, Gregg Rabideau, Steve Schaffer, Daniel Mandl, and Stuart Frye. Timeline-based space operations scheduling with external constraints. In *Twentieth International Conference on Automated Planning and Scheduling*, 2010.

[dlCL20] Jean-Pierre de la Croix and Grace Lim. Event-driven modeling and execution of robotic activities and contingencies in the europa lander mission concept using bpmn. In *International Symposium on Artificial Intelligence, Robotics and Automation in Space (iSAIRAS)*, 2020.

[dlCLH$^+$17] J.-P . de la Croix, G. Lim, J. Vander Hook, A. Rahmani, G. Droge, A. Xydes, and C. Scrapper. Mission modeling, planning, and execution module for teams of unmanned vehicles. In *Unmanned Systems Technology XIX*, 2017.

[Gea16] D Gaines and et al. Productivity challenges for Mars rover operations. In *Proceedings of 4th Workshop on Planning and Robotics (PlanRob)*, pages 115–125. London, UK, 2016.

[GK19] Robert P. Goldman and Ugur Kuter. Hierarchical Task Network Planning in Common Lisp: the case of SHOP3. In *Proc 12th European Lisp Symposium*, pages 73– 80. Zenodo, April 2019.

[GKPV03] Carlos Guestrin, Daphne Koller, Ronald Parr, and Shobha Venkataraman. Efficient solution algorithms for factored MDPs. *Journal of Artificial Intelligence Research (JAIR)*, 19:399–468, 2003.

[GL08] Christopher Grasso and Patricia Lock. Vml sequencing: Growing capabilities over multiple missions. In *SpaceOps 2008 Conference*, page 3295, 2008.

[Han17] Kevin P Hand. *Report of the Europa Lander science definition team.* National Aeronautics and Space Administration, 2017.

[IJLM16] Luca Iocchi, Laurent Jeanpierre, Maria Teresa Lazaro, and Abdel-Illah Mouaddib. A practical framework for robust decision-theoretic planning and execution for service robots. In *International Conference on Automated Planning and Scheduling (ICAPS)*, 2016.

[JMM$^+$00] A. Jónsson, P. Morris, N. Muscettola, K. Rajan, and B. Smith. Planning in interplanetary space: Theory and practice. In *AIPS*, pages 177–186, 2000.

[KRY$^+$21] So Young Kim, Kenny Roffo, Sean C Ye, Grace Tan-Wang, Sharon Laubach, Guy Pyrzak, and Glen Reeves. Designing for operating autonomous space missions: Concept of operations design for europa lander using mission-level modeling and simulation. In *ASCEND 2021*, page 4116. 2021.

[LRK$^+$20] Christopher R Lawler, Forrest L Ridenhour, Shaheer A Khan, Nicholas M Rossomando, and Ansel Rothstein-Dowden. Blackbird: Object-oriented planning, simulation, and sequencing framework used by multiple missions. In *2020 IEEE Aerospace Conference*, pages 1–20. IEEE, 2020.

[MNPW98] N. Muscettola, P. Nayak, B. Pell, and B. Williams. Remote agent: To boldly go where no ai system has gone before. *Artificial intelligence*, 103(1-2):5–47, 1998.

[NAI$^+$03] D. S. Nau, T. C. Au, O. Ilghami, U. Kuter, J. W. Murdock, D. Wu, and F. Yaman. SHOP2: An HTN Planning System. *Journal of Artificial Intelligence Research*, 20:379–404, Dec 2003.

[PGG$^+$98] Barney Pell, Edward B Gamble, Erann Gat, Ron Keesing, James Kurien, William Millar, P Pandurang Nayak, Christian Plaunt, and Brian C Williams. A hybrid procedural/deductive executive for autonomous spacecraft. In *Proceedings of the second international conference on autonomous agents*, pages 369–376, 1998.

[PGK$^+$97] Barney Pell, Erann Gat, Ron Keesing, Nicola Muscettola, and Ben Smith. Robust periodic planning and execution for autonomous spacecraft. In *IJCAI*, pages 1234–1239, 1997.

[RB17] G. Rabideau and E. Benowitz. Prototyping an onboard scheduler for the mars 2020 rover. In *International Workshop on Planning and Scheduling for Space (IWPSS 2017)*, Pittsburgh, PA, June 2017.

[ROS22] ROS. Robot operating system https://ros.org, 2022.

[Spa12] Matthijs TJ Spaan. Partially observable Markov decision processes. In *Reinforcement Learning*, pages 387–414. Springer, 2012.

[SZS17] Sandhya Saisubramanian, Shlomo Zilberstein, and Prashant Shenoy. Optimizing electric vehicle charging through determinization. In *International Conference on Automated Planning and Scheduling (ICAPS) Workshop on Scheduling and Planning Approaches*, 2017.

[TMH$^+$20] M. Troesch, F. Mirza, K. Hughes, A. Rothstein-Dowden, R. Bocchino, A. Donner, M. Feather, B. Smith, L. Fesq, B. Barker, and B. Campuzano. Mexec: An onboard integrated planning and execution approach for spacecraft commanding. In *Workshop on Integrated Execution (IntEx) / Goal Reasoning (GR), International Conference on Automated Planning and Scheduling (ICAPS IntEx/GP 2020)*, October 2020. Also presented at International Symposium on Artificial Intelligence, Robotics, and Automation for Space (i-SAIRAS 2020) and appears as an abstract.

[VEJ$^+$05] Vandi Verma, Tara Estlin, Ari Jónsson, Corina Pasareanu, Reid Simmons, and Kam Tso. Plan execution interchange language (plexil) for executable plans and command sequences. In *International Symposium on Artificial Intelligence, Robotics and Automation in Space (iSAIRAS)*, 2005.

[VGR$^+$17] Vandi Verma, Dan Gaines, Gregg Rabideau, Steve Schaffer, and Rajeev Joshi. Autonomous Science Restart for the Planned Europa Mission with Lightweight

Planning and Execution. In *International Workshop on Planning and Scheduling for Space (IWPSS 2017)*, Pittsburgh, PA, June 2017.

[Von] Volze. *BPMN 2010: Business Process Modeling Notation*, chapter An Overview of BPMN 2.0 and Its Potential Use. Springer, London.

[WRBC22] D. Wang, J. A. Russino, C. Basich, and S. Chien. Analyzing the efficacy of flexible execution, replanning, and plan optimization for a planetary lander. In *International Conference on Automated Planning and Scheduling*, June 2022.

[WWZ17] Kyle Hollins Wray, Stefan J. Witwicki, and Shlomo Zilberstein. Online decision-making for scalable autonomous systems. In *Intl Joint Conference on Artificial Intelligence (IJCAI)*, 2017.

[YFG07] Sung Wook Yoon, Alan Fern, and Robert Givan. FF-Replan: A baseline for probabilistic planning. In *International Conference on Automated Planning and Scheduling (ICAPS)*, volume 7, pages 352–359, 2007.

[Zon20] S Zonak. Event-driven operations, oss and jwst. In *STSCI Newsletter 2020 v. 37 Issue 02 https://www.stsci.edu/contents/newsletters/2020-volume-37-issue-02/event-driven-operations-oss-and-jwst*, 2020.

[ZWBM02] Shlomo Zilberstein, Richard Washington, Daniel S. Bernstein, and Abdel-Illah Mouaddib. Decision-theoretic control of planetary rovers. In *Advances in Plan-Based Control of Robotic Agents*, pages 270–289. Springer, 2002.