Available online at www.sciencedirect.com

ScienceDirect

Advances in Space Research xxx (xxxx) xxx

ADVANCES IN SPACE RESEARCH (a COSPAR publication)

www.elsevier.com/locate/asr

Enabling astronaut self-scheduling using a robust advanced modelling and scheduling system: An assessment during a Mars analogue mission

Michael Saint-Guillain [a,*], Jean Vanderdonckt [a], Nicolas Burny [a], Vladimir Pletser [b], Tiago Vaquero [c], Steve Chien [c], Alexander Karl [d], Jessica Marquez [e], Cyril Wain [a], Audrey Comein [a], Ignacio S. Casla [a], Jean Jacobs [a], Julien Meert [a], Cheyenne Chamart [a], Sirga Drouet [a], Julie Manon [a]

[a] *Universite catholique de Louvain, Belgium*
[b] *European Space Agency (ret.), Blue Abyss, UK*
[c] *Jet Propulsion Laboratory, California Institute of Technology, CA, USA*
[d] *Space Applications Services, Belgium*
[e] *NASA Ames Research Center, Moffett Field, CA, USA*

Received 6 October 2022; received in revised form 23 March 2023; accepted 27 March 2023

Abstract

Human long duration exploration missions (LDEMs) raise a number of technological challenges. This paper addresses the question of the crew autonomy: as the distances increase, the communication delays and constraints tend to prevent the astronauts from being monitored and supported by a real time ground control. Eventually, future planetary missions will necessarily require a form of *astronaut self-scheduling*. We study the usage of a computer decision-support tool by a crew of analog astronauts, during a Mars simulation mission conducted at the Mars Desert Research Station (MDRS, Mars Society) in Utah. The proposed tool, called *Romie*, belongs to the new category of *Robust Advanced Modelling and Scheduling* (RAMS) systems. It allows the crew members (i) to visually model their scientific objectives and constraints, (ii) to compute near-optimal operational schedules while taking uncertainty into account, (iii) to monitor the execution of past and current activities, and (iv) to modify scientific objectives/constraints w.r.t. unforeseen events and opportunistic science. In this study, we empirically measure how the astronauts, who are novice planners, perform at using such a tool when self-scheduling under the realistic assumptions of a simulated Martian planetary habitat.

Keywords: Scheduling; Uncertainty; Long duration exploration missions; Astronauts autonomy; Operations management

1. Introduction

Past space missions have had very limited experience in human self-scheduling. In fact, Marquez et al. (2019) states that current human operations, including extravehicular activities (EVAs), are *"carefully choreographed, and rehearsed events, planned to the minute by a large team of EVA engineers, and guided continuously from Earth"* (Bell and Coan, 2012; Bell and Coan, 2012; Miller et al., 2015; Miller et al., 2015). Activities on the International Space Station (ISS) for example are planned to various detail months and weeks in advance, and transition about two weeks ahead of the planned day into the real-time

* Corresponding author.
E-mail address: m.stguillain@gmail.com (M. Saint-Guillain).

https://doi.org/10.1016/j.asr.2023.03.045
0273-1177/

environment to be reviewed by all teams involved in the activities of that day to allow for further fine tuning in the days before execution. In case of unexpected events requiring an adaptation or re-planning of the day's activities, e.g. equipment failure, or an activity taking considerably longer than anticipated, the decision on how the rest of the day's timeline will be impacted lies with the Flight Director based on inputs by the activity stakeholders and ISS Planners, and then communicated to the crew on board. Activities can move to a different astronaut if there's extra time available, can replace another activity with lower priority, or be moved to another day. The ISS Planners are 24/7 on console and working on the schedules of the coming days and weeks. Today's events impacting tomorrow's timeline will be worked over night while the astronauts are asleep. As the distances increase however, the communication delays rapidly become an obstacle to remote real time monitoring and management of operations from Earth. However, human operations on Mars are expected to be carried out at a faster rate than current rover missions (Mishkin et al., 2007; Mishkin et al., 2007), which implies new planning strategies and tools that account for latency-impacted interactions (Eppler et al., 2013; Eppler et al., 2013). Current Mars rover missions are commanded by the ground operations team at most once per Martian day, or sol, and operate independently in between such contacts. In addition, future planetary EVAs are likely to be driven by science (Drake et al., 2010; Drake et al., 2010; Drake and Watts Kevin, 2014; Drake and Watts Kevin, 2014), requiring flexible adaptations according to scientific samples. In such context, future human space missions will have to enable some degree of crew autonomy and self-scheduling capabilities.

The problem of scheduling a set of operations in a constrained context such as the *Mars Desert Research Station* (MDRS, Fig. 1) is not trivial, even in its classical deterministic version. It should be seen as a generalization of the well-known NP-complete *job-shop scheduling problem* Lenstra and Kan (1979), which has the reputation of being one of the most computationally demanding problems Applegate and Cook (1991). Hall and Magazine (1994) raise on the importance of mission planning, as 25% of the budget of a space mission may be spent in making these decisions beforehand, citing the Voyager 2 space probe for which the development of the a priori schedule involving around 175 experiments requiring 30 people during six months. Nowadays, hardware and techniques have evolved. It is likely that a couple human brains, together with brand new laptops, may suffice in that specific case. Yet, the problems and requirements have evolved too. Instead of the single machine Voyager 2, space missions have to deal with teams of astronauts.

1.1. Rescheduling on-the-fly: objectives, constraints and opportunistic science.

Classical space missions are currently scheduled days ahead. Complex decision chains and communication delays prevent schedules from being arbitrarily modified, hence online reoptimization approaches are usually not appropriate. A human mission on Mars is different. It will necessarily be a long duration mission. The communication delays, in each direction, range from 3 to 22 min. Finally, in the current configuration of Mars orbiters, only a few short communication windows with Earth are possible per each Martian day (called a *sol*), with limited data rate (2 Mega bits per second).

In such conditions, any deviation from the original plan must be managed on the fly by the astronauts themselves. However, Marquez et al. (2021) demonstrated the fact that astronauts are not good at solving such complex problems by hand. This is not surprising. The sheer complexity of space systems means that thousands of constraints must be accounted for in decision making, and balancing of a large number of competing soft objectives must also be considered. An articulation of the size of this problem space for the Rosetta Orbiter mission science planning is described in Chien et al. (2021) and a future human mission to Mars is likely to be orders of magnitude more complex. Furthermore, the astronauts must also be able to adapt their schedules according to new scientific goals and requirements, such as conducting opportunistic science (*e.g.*, recording a dust devil), or even a new scientific project, or unexpected events such as machine breakdowns. In other words, the human machine team must be able to track evolving scientific objectives and operations con-

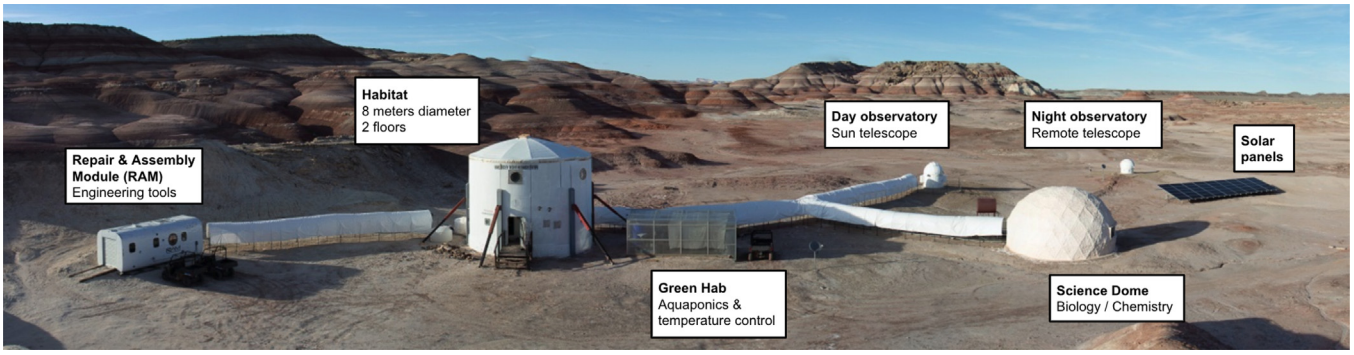

Fig. 1. The Mars Desert Research Station (MDRS), located in the Utah desert, is a Mars analog planetary habitat (Mars Society).

straints to re-optimize activities in an ever changing mission context.

Whereas our study focused on short-term scheduling, it is worth noting that there are several perspectives, ranging from tactical planning (short term) to strategic planning (long-term). In other words, various levels of granularity: mission objective vs instrument-specific procedures. Some may easily be transferred into a crew autonomy (like scheduling a repair), others will have to reside on the ground segment, as they need for instance a scientific debate on reshaping scientific objectives along the mission.

1.2. The impact of uncertainty

At the MDRS, computing an optimal schedule becomes significantly less attractive as problem data, such as the manipulation time of experiments, are different from their predicted values. In a constrained environment with shared resources and devices, such deviations can propagate to the remaining operations, eventually leading to global infeasibility. The purpose of this paper is to investigate, based on the real case study of a Mars analog mission, the impact of stochastic robust modeling against a classical deterministic approach on the reliability of a priori mission planning.

Consider the simple project depicted in Fig. 2 (top). Suppose all tasks have to be scheduled on the resource, then one must necessarily begin with A and end with activity D. There are only two valid schedules, shown in Fig. 2 (bottom). In fact, schedule (A, B, C, D) looks much more efficient, as all tasks are completed on the first day. On the contrary, schedule (A, C, B, D) requires an additional day. However, this is *only true on the paper*, when everything is predictable. If you account for (temporal) uncertainty, then the story is different. If operation A lasts for more than 1 h, schedule (A, B, C, D) is *not valid anymore*: B will have to be resumed or rescheduled on day 2 (an additional day, that was not expected!). When starting C, we realize the worst: it actually requires to be processed the same day as A. Mission failed. Remark that if the true average processing time of A is 1 h, then this scenario happens with at least 50% probability. On the contrary, (almost) whatever happens to A, under schedule (A, C, B, D) everything goes fine. This schedule is said to be *robust*. Its success probability is simply 1 minus the probability that A exceeds four hours (which we assume to be fairly unlikely).

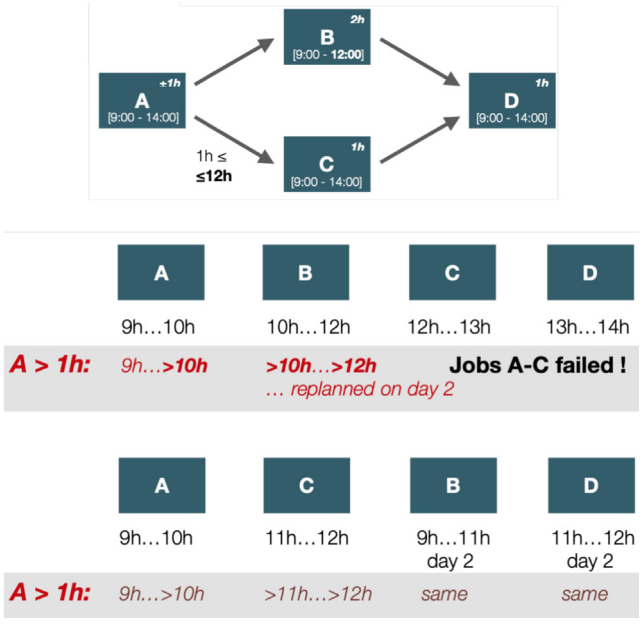

Fig. 2. Illustrative example with four tasks: A, B, C, D, to be scheduled on the same line. Each task has a processing time of 1 or 2 h, and a time window spanning either the entire work day (9 am to 2 pm) or part of it (9 am to 12 pm or 12 noon). Tasks B and C require A to be completed, and D requires both B and C to be completed, before being started. Task C must wait at least one hour after completion of A to start, and must be completed during the same day.

1.3. Application and contributions

Contrary to current space missions, in which astronaut operations are paced to the minute and supervised in real time by experienced planners on Earth, future long duration exploration missions (LDEMs) will necessarily involve a certain level of autonomy, or *self-scheduling*.

In this paper, we study the ability of astronauts, being novice planners, to organize themselves their operations. The problem at stake being intrinsically complex, a decision-support system is provided. We hence measure, report and analyse how efficiently the astronauts are exploiting such technology to pursue their scientific objectives, *autonomously*, and during a real Mars analog mission carried out at the MDRS. We demonstrate the usability and usefulness of a decision support system such as *Romie*, and compare its user experience with an existing one (Minerva, NASA), in terms of human–machine interactions.

2. Operations management software systems

Existing systems usually fall into *a)* being specifically designed for a particular application/mission or operational context, and/or *b)* not having an integrated optimization system able to generate *robust* schedules, from a probabilistic point of view. Instead, the *Romie* RAMS system is used in this study. Compared to classical frameworks, called APS for advanced planning and scheduling systems, a robust advanced modelling and scheduling (RAMS) system such as *Romie* provides the following *technological innovations*:

i) Graphical problem modelling ($\neg a$). The user is able to graphically draw and manipulate the structure and constraints of its scheduling problem, including stochastic models for task durations.
ii) Optimization under uncertainty ($\neg b$). An optimization engine allows the user to generate, or adapt existing schedules, in a way that produces schedules robust w.r.t. uncertainty.

2.1. Planning and scheduling in space

The first planning and scheduling tools for space missions were dedicated software systems, specialized to specific application domains. Johnston and Miller (1994) described the SPIKE system, a general framework for scheduling, developed by the Space Telescope Science Institute for NASA's Hubble Space Telescope. Other examples of aerospace scheduling tools and applications are: Chien et al. (1999), developed for scheduling the operations of a particular shuttle science payload (DATA-CHASER) with primary focus on solar observation; Jónsson et al. (2000) for the Deep Space One mission; Ai-Chang et al. (2004) for the Mars Exploration Rover mission; Chien et al. (2005) for NASA's Earth Observing One Spacecraft; and Cesta et al. (2007) for the Mars-Express mission. Chien et al. (2012) provides a detailed survey on (semi-) automated planning & scheduling systems developed for space applications.

As the need for more generic approaches to support multiple mission and multiple domains increased, a planning/scheduling C++ library has been proposed: ASPEN (Fukunaga et al., 1997; Fukunaga et al., 1997; Rabideau et al., 1999; Rabideau et al., 1999; Chien et al., 2000; Chien et al., 2000). At that time, ASPEN provided the elements that were commonly found in existing complex planning and scheduling systems, for example for generating operation schedules for the Rosetta orbiter Chien et al. (2021). In 2009, ESA's Advanced Planning and Scheduling Initiative (APSI) aimed at developing a general software framework for supporting development of AI planning and scheduling prototypes, for various types of space missions. The APSI is described in Steel et al. (2009).

Presented in Yelamanchili et al. (2020), the *Copilot* system for Perseverance Rover mission does have a modelling system called COCPIT, and a planner, but it is specifically designed for that mission. This ground automated planning system is intended for use with an onboard planner in preparation for deployment Rabideau and Benowitz (2017), Agrawal et al. (2021b,a). Of particular relevance to this work is the Copilot ground scheduler with explanation capability Agrawal et al. (2020) and Monte Carlo variation of execution to set parameters for onboard rescheduling Chi et al. (2019). Again, these systems are fairly tailored to the specifics of the Perseverance rover mission.

A key differentiation in space missions is human surface missions versus automated orbital, flyby and other space mission modalities. Flyby and orbital missions can be well predicted, enabling pre-planning of observation campaigns (often days or weeks in advance) and executed (excepting fault protection) primarily open loop. Some exceptions to this generalization over non surface missions exist. For example VML was used onboard Spitzer to enable it to handle variable execution time or failure to acquire guide stars for observations. JWST has a similar capability. Some examples of such predictable missions that have used automated scheduling include (non exhaustive list) MAMM, Orbital Express, Hubble, Spitzer, Earth Observing One, and Rosetta Orbiter to name a few (more are described in Chien et al. (2012)). In contrast, surface missions, especially those involving astronauts (such as human exploration of Mars), involves more intimate interaction with the environment and are therefore harder to predict. Previous lander and rover missions have encountered challenges in variability of action duration (e.g. driving), challenges in physical manipulation (e.g. placing measurements, drilling and coring, ...) which might mean activity failure. Such challenges in unpredictability of execution are strong motivation for the capability of any human–machine joint system to be able to continuously replan in light of such occurrences (see Gaines et al. (2016) for an excellent study of such challenges for the Mars Science Laboratory Mars Rover Mission and Gaines et al. (2020) for work at increasing the ability of future Mars Rovers to autonomously redirect their activities in such situations.). Note that this autonomous handling of uncertainty is at a premium for future missions to explore unknown environments such as the Europa Lander Mission Concept Wang et al. (2022).

2.2. Human self-scheduling in space

In Deans et al. (2017), a suite of software tools called Minerva is proposed in order to support operations planning and execution. Minerva and its components (xGPS, Playbook, SEXTANT) have been tested during several planetary and space simulation missions, including the BASALT research program (described in Brady et al. (2019), Brady et al. (2019)) and four analog missions at NEEMO (Chappell et al., 2017; Chappell et al., 2017; Marquez et al., 2017; Marquez et al., 2017). Compared to Minerva, the key differences of our proposed tool *Romie*, in terms of functionalities, rely on the modelling interface and the scheduling optimization engine, which enable strategical a priori planning. In addition, the optimization is conducted while taking uncertainty into account. The Minerva suite is rather focused on tactical planning, including geospatial planning, which allows crew path planning and coordination using satellite maps. The strategical planning is assumed to be performed before the start of the mission, and is therefore not covered by the Minerva suite. However, even when a predefined schedule is provided prior to the start of the operations, it is very likely that the schedule will require online modifications as the operations go. Marquez et al. (2021) showed the limits of human self-scheduling when operators must solve and adapt the planning manually while taking hard constraints into account (not even thinking about uncertainty). By providing both a way to adapt the model and solve it using an embedded optimization engine, *Romie* is complementary to Minerva.

2.3. Romie

Recall the two technological innovations of Romie, presented in the beginning of this section: **i)** domain-

independent graphical modelling (and scheduling) interface and **ii)** optimization under uncertainty. Unlike all existing tools, both modelling and modifying the problem is now made accessible to the end-user, which is critical for a reliable self-scheduling. Up to our knowledge, the MapGen tool presented in Ai-Chang et al. (2004) was one of the very first tools to propose a visual constraints editor. However, the latter was not generic, but specific to its application case, the NASA's MER mission.

Romie is the first scheduling tool to propose an integrated robust (*i.e.* under uncertainty) optimization engine. Having more robust (*i.e.* reliable) schedules, the end users are more likely to avoid last minute rescheduling. *Eventually, what-if analysis, as well as sensitivity analysis, become less relevant*: by considering the uncertainties related to task execution, the solutions are optimized following directly the *expected values* of the chosen key performance indicators (KPIs).

We believe that both Romie's technological innovations **i)** and **ii)** provide significantly more autonomy to the end users, whom remain otherwise highly dependent of planning and scheduling experts. Based on the theoretical foundations defined in Saint-Guillain et al. (2021a), the empirical contribution of point **ii** (optimization under uncertainty) has been extensively validated by Saint-Guillain (2019), Saint-Guillain et al. (2022a) and Saint-Guillain et al. (2022b). Testing the ability of the non-experts end-users to actually "self-schedule" using **i)**, *i.e.* the graphical modeling formalism, is the main goal of this study.

Fig. 3 depicts the key functionalities of our system:

- *Graphical modeling* of the problem at stake, in its own operational context: human and physical resources, operational constraints, key performance indicators (KPIs), execution uncertainties.
- *Robust scheduling*: the optimization engine takes the time uncertainty on each task's duration into consideration, using modified-PERT distributions, yielding schedules with high probability of success.
- *KPI-guided optimization*: The schedules are optimized while pursuing (a combination of) *various KPIs*, including success probability, expected cost, expected quality, and even operator wellness.
- *Operations update* and *online reoptimization*. The system knows the difference between past and future operations. As the schedules can be extensively modified by hand, in particular in the past (but also in the future), reoptimization on future decisions can be performed based on what actually happened in the past.

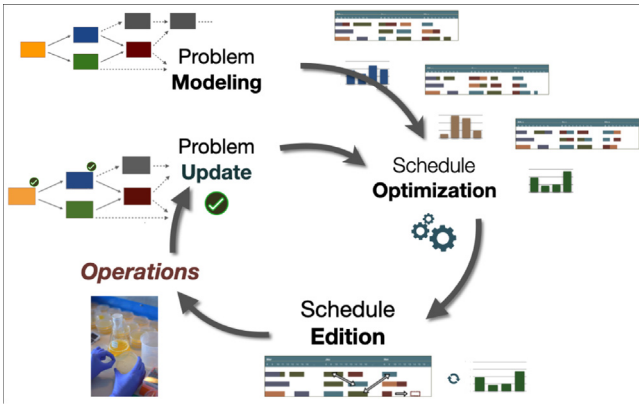

Fig. 3. Overview of the *Romie* modeling and scheduling system.

3. The M.A.R.S. UCLouvain 2022 mission

Our study on astronaut self-scheduling is driven by the scientific research projects to be carried out by the crew members in the context of the simulation. Before the actual beginning of the mission, the selected projects have been modelled in the *Romie* system, and provisional schedules have been designed. In what follows, the different projects are described. Thereafter, their modelling and a priori scheduling is analyzed, from the user's point of view.

3.1. Experimental plan

Our study aimed at answering the following questions: how long does it take for a novice user before setting up correct schedules (on-boarding time), and are our astronauts all able to adapt their scientific objectives as the operations evolve? We tackled these questions by focusing on the temporal evolution of these following two complementary KPIs: system usability and user experience. The ISO-9241–210 standard International Standard Organization (2019) defines the usability as *the extent to which a system, product or service can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use*. User Experience is defined by the same standard as *the user's perceptions and responses that result from the use and/or anticipated use of a system, product or service* and is generally understood as inherently dynamic, given the ever-changing internal and emotional state of a person and differences in the circumstances during and after an interaction with a product Vermeeren et al. (2010).

Several scientific research projects were conducted at the MDRS. Each project was carried on in place, by either one or two astronauts. Some projects (such as health projects) involve the participation of all the crew members. Yet, these projects were designed and prepared months ahead. During that period, preliminary experiments were conducted on *Romie*, providing first results on the system's usage by the astronauts, in offline (supervised) conditions. The actual *M.A.R.S. UCLouvain 2022* mission period, which lasted 12 days on field at the MDRS (see Fig. 1), constituted the main material of this study. Day after day, each crew member used the *Romie* system to monitor and update their operations.

3.2. The Mars desert research station

The MDRS in the desert of Utah has been in operation since 2002 from November through April every year. The geologic features of the surrounding Jurassic–Cretaceous terrain also make the desert environment seem Mars-like to crew members. The MDRS habitat itself is a vertical cylindrical structure of approximately 8 m diameter and 6 m high, composed of two floors. The ground floor (lower deck) includes a front door airlock used for simulated EVA, an EVA preparation room, a large room used as a laboratory for geology and biology activities, a small engineering workshop area, a second back door airlock for engineering activities, a small bathroom and a toilet, three small windows, and a stair leading to the first floor. The first floor (upper deck) includes a common area or living room with a central table, a wall-attached circular computer/electronic table, a kitchen corner, six small bedrooms, and a loft on top of the small bedrooms. Some panoramic pictures from the inside are provided in Fig. 4.

3.2.1. A typical day on Mars

The day-to-day operations at the MDRS is as follows. The crew wakes up at 7:30. Then directly follows a twenty minute morning sport session, before having breakfast, which is typically the right moment for daily medical examinations.

Extra-vehicular activities (EVAs) take place during the morning. Between three and five crew members get prepared for the daily EVA. That takes roughly one hour, during which the crew members that are not participating in the EVA help the others to don their spacesuits, and parameterize the communication devices. The EVA should start no later than 9:30, as it must necessarily be ended before 12, which provides roughly two hours to reach all the EVA objectives. The crew members that remain inside MDRS stay in permanent contact with the EVA party, while performing the daily chores.

Fig. 4. Panoramic pictures of some of the Mars Desert Research Station (MDRS) elements, from inside. From top to bottom: upper deck, lower deck, EVA preparation room, science dome, green hab.

Scientific activities then take place every day from 1:30 pm to 6 pm. The crew members work on separate places, depending on their research field: the crew botanist stays in the green hab, biologists and chemists in the science dome, the astronomer takes pictures of the sun in the day observatory, engineers work in the repair & assembly module (RAM), ...During each afternoon, the crew members, one by one, use the RAMS system to monitor and schedule their operations. Therefore, each crew member uses the *Romie* system *–and answer the questions and exercises defined in the scope of our study–* once a day, for approximately 30 to 60 min.

From 6 pm, all the crew members would generally interrupt their activities, in order to prepare for the daily communication window, from 7 to 8 pm. At exactly 7 pm all the specific reports are sent to ground control: engineering, medical, green hab, EVA and EVA request, journalist, and commander report. While sharing the diner, the entire crew remains available to answer questions on these reports. The remaining of the evening constitutes a privileged, necessary moment for socialising.

3.2.2. Time-eaters at MDRS

Previous studies (Pletser et al., 2009; Pletser et al., 2009; Boche-Sauvan et al., 2009a; Boche-Sauvan et al., 2009a; Boche-Sauvan et al., 2009b; Boche-Sauvan et al., 2009b; Pletser, 2010b; Pletser, 2010b; Pletser, 2010a; Pletser, 2010a; Thiel et al., 2011; Thiel et al., 2011; Pletser and Foing, 2011; Pletser and Foing, 2011) have shown that there are many 'time-eaters' in a day at the MDRS during a simulated Mars stay mission. Table 1 reports the measured average unproductive time of the 76th rotation in 2009. A hypothetical average crew member would have only approximately 7 h left for scientific work. Even if this average estimation is very crude, it shows that the remaining average time for scientific work is significantly low and that a lot of time is spent on unproductive tasks, chores and maintenance. These data are consistent with previous findings for crew 5 in 2002 (Clancey, 2006; Clancey, 2006) in terms of activity duration. Furthermore, the time-sharing of occupation of the ground floor laboratories between geologists and biologists was always difficult to

Table 1
Average durations with standard errors for an average crew member. Note: Based on measured activity durations of Crew 76 (see Pletser and Foing, 2011; Pletser and Foing, 2011); h = hours; m: minutes. Maint.: maintenance. Evening: evening common activities. It sums up to 17 h 01 min, ±53 min.

Breakfast	Lunch	Dinner	Chores	Maint.	Evening	Sleep
44 m ±02 m	48 m ±02 m	57 m ±01 m	3h08m ±18 m	1h23m ±10 m	1h35m ±13 m	8h26m ±07 m

establish and necessitated a lot of good-will of all parties. Both scientists' teams had different needs and expectations, e.g., biologists need a clean, pristine and well-lit environment to analyse and process soil samples, while geologists need a darker environment to handle, manipulate, crush and process samples with instruments, often generating dust and noise. In order to share the use of the single room laboratory, both groups of scientists had to work during the night alternately. Traffic of crew members through common and scientific areas was another point of study as it created also interruption of science work by engineers to assess the Hab systems and by other crew members to access stowage areas, etc. It highlights the importance of having a proper and performing dynamic planning tool that can be used on the spot and on the run, fine tuning and adapting an already agreed day planning, for example, either in the evening at dinner or in the morning during breakfast.

Several recommendations were made to improve the design in order to optimize the traffic and to decrease the time spent unproductively from a scientific point of view. Yet, the "time-eaters" cannot be completely avoided. The system proposed in the current study comes in addition to these recommendations, as we investigate an AI based decision system to optimise productivity while leveraging unpredictable time deviations.

Naturally, being a non-scientific work, using the *Romie* system should also be considered as a time eater. Its per-person usage duration (half an hour to one hour per day) should probably be reduced to be usable in an actual mission context. Another way to save time would be to assign the responsibility to only one astronaut, thus being the crew planner, to manage the global mission schedule by using the *Romie* system. However, our study required several end users.

3.3. Research projects

Each analogue astronaut has her/his own research objectives for the mission. In fact, each astronaut (experimenters) prepared one different research project to be carried out at MDRS. There are thus eight research projects, from eight different fields such as biology, botanic, engineering, astronomy or medicine:

Soil dielectric 3D mapping: Using a ground penetrating radar, installed on a vehicle, the dielectric properties of the soil surrounding the station are measured and projected on a 3D map, constructed by photogrammetry using a drone. Such a map could be exploited to optimize future irrigation systems. **3D printing:** This experiment exploits 3D printing scaffolds in bio-ink to seed stem cells, and performs mechanical stress–strain tests on the resulting micro-architecture. **Sleeping hypnosis:** This project tests an hypnosis technique, used in medicine before falling asleep, to help the astronauts having better, deeper sleeps. **ExFix:** Accidents and injuries on Mars are dangerous. Manon et al. (2023) study a low-cost external fixator to stabilize broken bones, which remains accessible, fast and easily achievable by any astronaut without surgical training. **Metabolic changes:** The lower gravity of Mars, its environment and the nutrition changes will have a big impact on future crews' metabolisms. Here, a protocol is developed for the monitoring of essential parameters of the health and metabolism of the crew members. **Insects in the astronauts' diet:** Insects constitute a potential alternative food solution for astronaut crews. The viability and yield rate of three insect species (orthoptera, beetle and lepidopteran) are experimented under Martian conditions. **Human flora bacteria on Mars:** The survival of some human flora bacteria and the efficacy of several antibiotics under Martian environmental conditions is experimentally studied. **Biofertilizers in Martian soil substrate:** This experiment analyzes how a closed environment like the MDRS station and with a Martian regolith, the caloric intake of astronauts can be filled thanks to biofertilizers in small quantities.

3.3.1. Modelling and scheduling on the RAMS system

Fig. 5 shows the modelling of one research project, as encoded in *Romie*. In fact, this modelling started with a discussion with the experimenter, which lead to a hand-drawn sketch. From this, a first encoding could be made on the system, using the graphical modelling interface, which formally encodes all the activities and constraints involved.

Fig. 6 shows another research project. From an operational point of view, this model has interesting properties. It involves a resource shared with other scientists: the laminar air flow (LAF). Since there is only one LAF in the station's science dome, this prevents other activities (belonging to other projects), also requiring the LAF, to be carried out at the same time. Another point of interest is the temporal constraint between *Cult.LQ B* and *Expo. TEST + CTL*, which involves a stochastic delay. In fact, the delay that must be waited between those two activities (1, 2 or 3 days) depends on the time needed by the bacteria to grow, and it is totally unpredictable by nature. Finally, there are temporal constraints, stating that some activity should not start sooner and/or later than a defined amount of time after some other activity.

The temporal constraints present in this model may potentially lead to a project failure, due to the underlying temporal uncertainty. Yet, another kind of complexity lies in models that involve the participation of several crew members, in addition to shared equipment. Fig. 7 shows an example of an optimized *provisional* schedule, as obtained using *Romie*'s optimization engine, for all eight research project during the entire mission. In this schedule, the activities involved in research project *ExFix* are highlighted. We directly see that many of these require time within the schedule of the other crew members.

The RAMS system here not only allows to check the deterministic KPIs, but also some probabilistic ones. From a *deterministic* point of view, when all the durations are assumed to require their nominal operational time, this planning is *feasible*. For example, in the project modelled

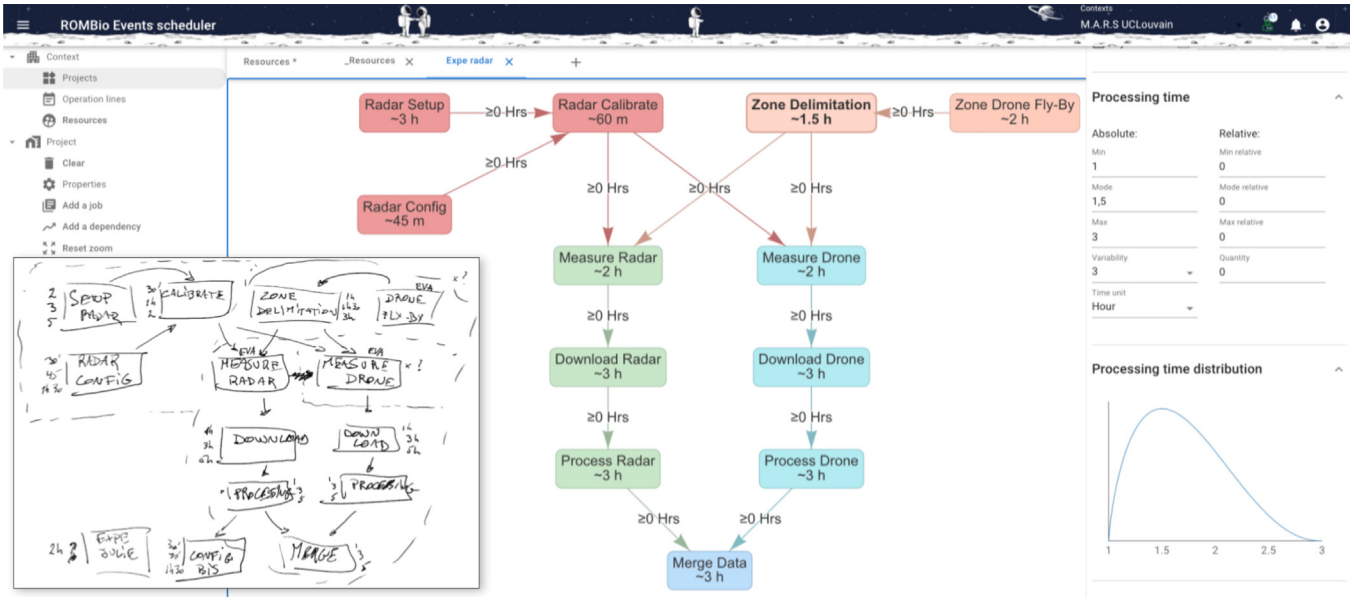

Fig. 5. Modelling research project *"Soil dielectric 3D map"* within *Romie*'s graphical interface. Boxes represent activities. Arrows represent (temporal) precedence constraints. Here the *Zone Delimitation* activity is selected, showing the parameters (right panel) that define its temporal uncertainty: min, mode, max. This activity requires *Zone Drone Flyby* to be completed before, and is a prerequisite to both activities *Measure Radar* and *Measure Drone*. Bottom left: original hand-drawn sketch.

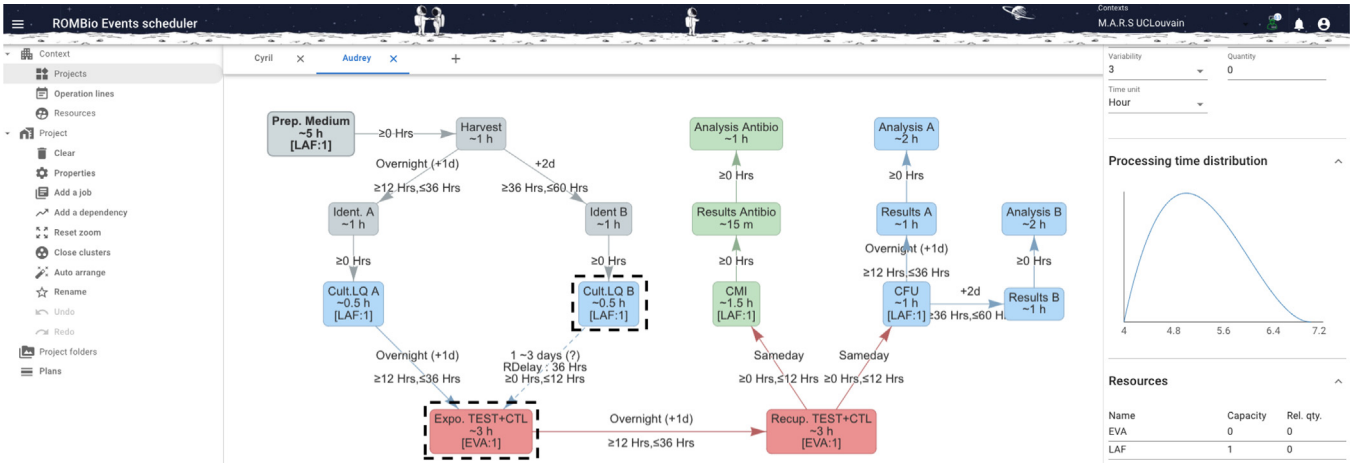

Fig. 6. Research project *"Survival of human flora bacteria"*. Amongst the activity properties in the right panel, we note that the selected activity *Prep. Medium* requires the *LAF* resource. The temporal constraint between *Cult.LQ B* and *Expo.TEST + CTL* (dashed) involves a stochastic delay, between 1 and 3 Martian days.

in Fig. 6, provided that the delay between *Cult.LQ B* and *Expo.TEST + CTL* will reveal to be exactly two sols. However, when taking *uncertainty* into account, then the mission probability of success is of 86.2%. Here the system only takes temporal uncertainty into account, not the fact that the activities themselves could be failed, requiring a rescheduling. Rescheduling operations, as well as adding new operations on the fly, will be part of the astronauts' daily manipulation on the system.

4. Theoretical foundations

In stochastic contexts such as space missions, computing optimal schedules becomes significantly less attractive as problem data, such as the manipulation time of the modelled activities, are different from their predicted nominal values. This is what we refer to as *uncertainty*. In a constrained environment with shared resources and devices, when they arise such temporal deviations can propagate to the remaining operations, eventually leading to global infeasibility, that is, a project failure. Given a schedule, a central question is then the following: considering temporal uncertainty, what is the actual probability of success of the mission?.

4.1. Project management is hard

The problem of scheduling a set of operations under constraints should be seen as a generalization of the

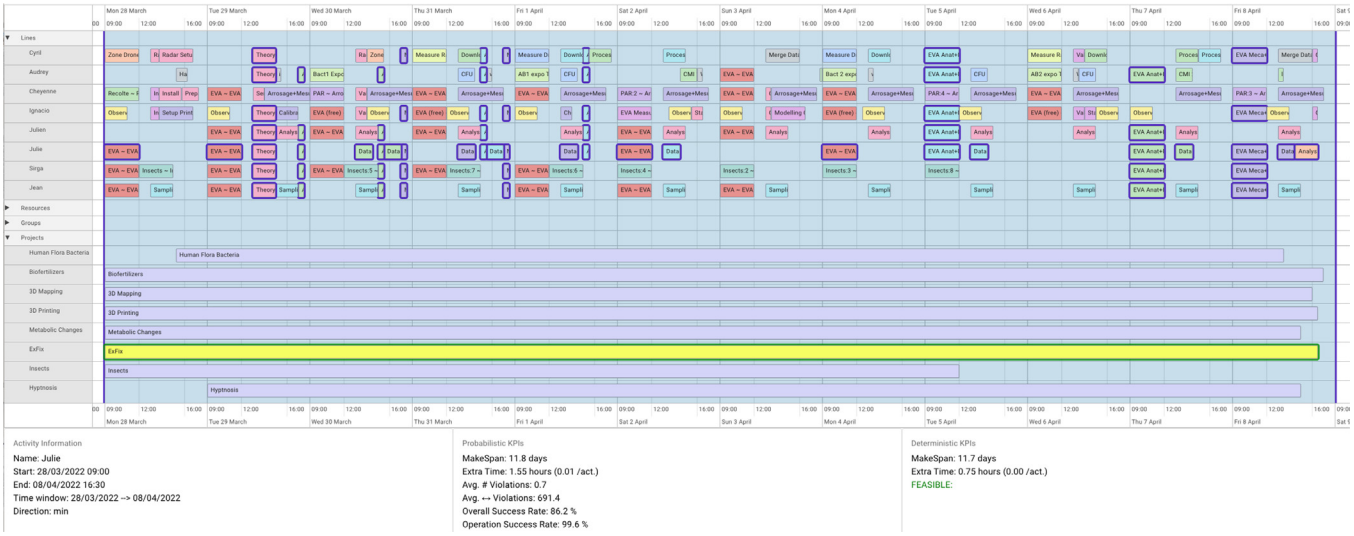

Fig. 7. A provisional schedule. This is the a priori schedule computed before the beginning of the operations. It involves 162 activities, each having contingent durations. The overall success probability is of 86.2%. Highlighted in yellow, the *ExFix* research project with all the related activities above, framed with blue rectangles. We see that this project imposes activities to several of (in fact, almost all) the crew members.

well-known *NP*-complete *job-shop scheduling problem* Lenstra and Kan (1979), which has the reputation of being one of the most computationally demanding (Applegate and Cook, 1991; Applegate and Cook, 1991). When taking uncertainty into account, the problem then becomes strongly *NP*-hard, an even harder family of problems. In a nutshell, *NP*-complete means that, no matter the available computational resources, the problem is conjectured as impossible to solve in practice, for realistic instance sizes, such as the number of activities and resources. In fact, whereas the problem depicted in Fig. 2 admits only two solutions, in practice the number of possible schedules grows exponentially with the number of tasks and resources. Back to Voyager 2 space probe mission, suppose we are interested in all the possible permutations between its 175 operations, then we have $175! \approx 10^{318}$ possible permutations.

Solution methods for combinatorial optimization problems. Fortunately, algorithmic and mathematical techniques exist in order to solve the problem without enumerating all the 10^{318} permutations and schedules. Mathematical methods, such as *integer programming*, *constraint programming*, *SAT solving*, and so on are known to be powerful methods, in the sense that given the right formulation of the the problem, generic solvers (e.g. Gurobi, CPLex) are able to find solutions and eventually provide proofs of optimality (without performing an exhaustive enumeration). Unfortunately, finding the correct formulation is usually the most complicated part of the problem, and the languages accepted by these generic solvers do not allow to express complicated specific operational constraints or objectives.

On the other hand, heuristic methods, such as *local search* which is exploited here in the *Romie* system, or other methods such as *genetic algorithms*, trade the completeness of the exact methods for more flexibility. Such methods will be able to find (hopefully) good solutions but, even when a solution found by the algorithm is optimal, it will not be able to prove (or even determine) it. However, describing what makes a solution acceptable or not, and what is the quality of a solution, is made much easier when there is no mathematical proof generation framework in behind.

The optimization engine of our decision system *Romie* is based on local search. In a local search algorithm, the key ideas are the following. (S) Start from an initial (potentially infeasible) solution x, such as a random permutation. This is the current solution. (M) Apply a local modification to x leading to another solution x', for example, by permuting two activities at random. Then, (E) evaluate the quality $f(x')$ and decide, according to $f(x)$, whether or not x' becomes the new current solution. Finally, repeat (M) and (E) until some stopping criterion (time or solution quality) is met. In the end, return the best solution encountered.

4.2. Uncertainty management

As shown in our introductory example, provided two different schedules, determining the best one (*e.g.* the more reliable) in light of the uncertainty is not trivial. Now, suppose this must be done for each and every permutation that is considered.

Computing the probability of success of a given permutation of activities can be done by computing the degree of dynamic controllability, or *robustness*, of the associated *probabilistic simple temporal network* (PSTN). Different approaches have been proposed to either approximate or compute this robustness. It is important to state that the robustness of a system depends on the uncertainty of course, but also on how clever is the system at reacting

to random events. In Saint-Guillain et al. (2021a), we refer to this "cleverness" as the *execution (or dispatching) protocol* $\mathcal{P}$, sometimes called *policy*, which defines how the system reacts to random events. The protocol may consist of simple rules, such as "start every operation as soon as possible". It could also be more elaborated strategies, involving preventive waiting times. A particular case is that of a protocol $\mathcal{P}$ solving the perfect online reoptimization problem (a multistage stochastic program). Depending on $\mathcal{P}$, the *robustness* of a system, namely a PSTN, can be defined in general terms as:

$$r^{\mathcal{P}}(N) = \sum_{\xi \in \Omega^N} \mathbb{P}\{\xi\} \Phi^{\mathcal{P}}(N, \xi) \quad (1)$$

where Ω^N is the support of all possible scenarios in which the PSTN could fall due to random events, $\mathbb{P}\{\xi\}$ is the probability to fall into a particular scenario ξ, and $\Phi^{\mathcal{P}}(N, \xi)$ simple returns one if executing the protocol $\mathcal{P}$ in scenario ξ leads to a successful mission, zero otherwise. In our case, a scenario ξ is an assignment of a duration to each activity, with support $\{\xi \in \mathcal{R}^n : \mathbb{P}\{\xi\} > 0\}$.

The computation of (1) remains intractable in practice, as the size of Ω^N grows exponentially with the number of contingent operations. For instance, consider the 162 operations involved in the provisional schedule of our M.A.R.S. UCLouvain 2022 mission, as depicted in Fig. 7. Suppose each operation can take any duration between 20 and 60 min, precise to the minute, then we have $20^{40} \approx 10^{88}$ scenarios possible. This motivates all kinds of sampling based methods, such as Monte Carlo, which therefore restricts the summation in (1) to a limited subset of $S \subset \Omega^N$.

In Saint-Guillain et al. (2021a), we showed that by reasoning on the connections between the random variables, instead of the scenarios, then the exact robustness could be computed exactly, in pseudo-polynomial (*i.e.* efficient) time. However, this is only provided that $\mathcal{P}$ actually is the natural "as soon as possible" strategy (and nothing more elaborated), and that the PSTN is well formed. The PSTN formalism being quite restrictive, there are many considerations of a space mission that cannot be modelled as a well formed PSTN, such as exotic constraints and resource issues (e.g. energy consumption). On the contrary, sampling-based methods are much more versatile, they generally do not impose a formalism as strict as PSTN's. Our *Romie* decision system implements such a method, the sample average approximation (Kleywegt et al., 2002; Kleywegt et al., 2002), within its optimization engine.

4.3. System architecture

Romie uses a decentralized architecture, as depicted in Fig. 8. The user interface is decoupled from the optimization engine part. There can be potentially many users connected to the system (*e.g.* the whole team of astronauts), using a classical web browser, since the user interface is implemented using a recent web technology (React). The optimization engine is actually composed of an arbitrary number of optimization agents, not necessarily hosted on the same servers, hence enabling parallel computing.

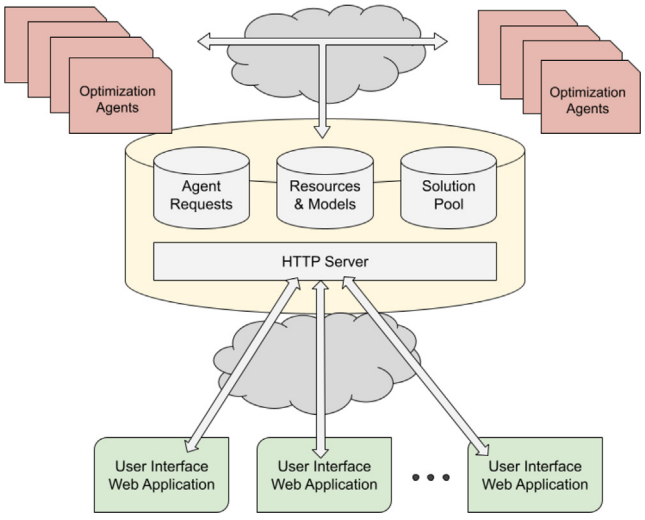

Fig. 8. Global architecture of the Romie robust advanced modelling and scheduling (RAMS) system.

The whole system is organized upon a database, which enables the end users to communicate with the optimization agents, but also the optimization agents to communicate altogether in order to distribute and parallelize the work. The database stores the following three key information:

- **Resources & Models**. Stores the operational context and problem descriptions, namely the models (*e.g.* the scientific project modelled in Fig. 6) describing the operational projects at stake. The user interface provides a visual representation of the logical information stored in the database, which is rather described in terms of discrete mathematical structures such as sets and graphs.
- **Solution pool**. The current states of computations. The solution pool is used by the optimization agents to communicate and share their results. An element of the pool is naturally a schedule (e.g. Fig. 7). For the same optimization problem, the pool may contain several (*e.g.* 10) solutions, corresponding to the bests solutions found so far by all the agents.
- **Agent requests**. A list of requested user actions. An action could be either one-shot, such as adding or removing a project for instance, or a running action, such as optimizing. One-shot requests are picked up by exactly one idle agent (at random), which will perform the associated action (*e.g.* remove a specific operation from a given project in a given schedule, and then recompute the KPIs, such as the overall probability of success). The only possible running request is an optimization action, which triggers all the available idle agents to concurrently try to improve a given schedule.

Whereas the architecture is designed to be decentralized, in practice all the elements could easily be integrated on the same computer, or machine (such as a spacecraft).

The communication formalism between the computing agents is however robust to latency, which enables a same physical system to host its own limited set of agents, whereas remote agents can be solicited in support, even with important delays (*e.g.* a couple of seconds from Earth to Moon). Or a fleet of autonomous planetary rovers could distribute the computational effort on any possible computational support in an acceptable range (say, five to ten light seconds), including the rovers themselves.

Also, the system easily recovers (actually, is not impacted at all) from the disconnection and the death of optimization agents, and new agents can be added on the course of the computations. Human operators connect remotely to the system, using a simple web browser, and may disconnect and reconnect without loss of any information, and without interrupting any ongoing computation. Finally, several different computations, for instance optimizing the same initial planning under three different combinations of KPI preferences, or under different operational resource limitations, may be carried out simultaneously (in which case these different optimization problems are distributed over all the available agents).

4.4. Online reoptimization

Whereas the provisional schedule depicted in Fig. 7 is computed before the beginning of the operations, in practice things rarely happen exactly as initially planned. In comparison, Fig. 10 shows the current state of past (executed) and future (planned) operations, at Sol 6. Remark for instance that most of the activities initially planned at Sol 2, during the morning, disappeared and had to be rescheduled. In fact, these corresponded to an EVA that had to be cancelled, due to bad weather conditions.

This leads to an online (*i.e.* dynamic) re-optimization problem, in which the decisions must be optimized given a fixed current state of the system, including past activities. During the mission, the astronauts actually updated their schedule at the end of every day, encoding what really happened, and reoptimized for the rest of the mission.

The best one can do is therefore to compute the schedule that is the more likely to succeed, knowing that in fact, no one knows what will actually happen. Mathematically speaking, this can be described as playing a game against Nature, in which at current time t, we take the decisions x^t that maximize their expected outcome (Saint-Guillain et al., 2021b; Saint-Guillain et al., 2021b), leading to the following multistage stochastic program:

$$\underset{x^t \in X^t}{\operatorname{argmax}}\ E_{\xi^{t+1}}\left[\max_{x^{t+1}\in X^{t+1}} E_{\xi^{t+2}}\left[\dots \max_{x^{h-1}\in X^{h-1}} E_{\xi^h}\left[\max_{x^h \in X^h} V^{x^{1..h}}(N,\xi)\right]\dots\right]\right] \tag{2}$$

where the maximum value of the first expectation is, by definition, equal to the current probability of success under perfect reoptimization.

The nested expectations in (2) form a tree structure, well known as the *scenario tree*. Unfolding the maximization operators as well leads to a full decision-scenario tree as illustrated in Fig. 9. Each path of the tree constitutes a possible scenario realization together with associated decisions, a sequence $\xi^t, x^t, \dots, \xi^h, t^h$. At time t, decisions x^t depend on the current history $\xi^{1..t}$ and maximize the expected value $E_{\xi^{t+1}}[\max_{x^{t+1}} \dots]$ of the future decisions at time $t+1$ given the remaining uncertainty, and so on until time h is reached.

The number of scenario $\xi \in \Omega$ in Eq. (1) is equal to the number of different paths in Fig. 9. We directly see that the size of our scenario tree grows exponentially with the number of decision steps and outcomes, which explains why computing (1) is intractable in practice.

4.5. Previous researches

Based on the real case study of a Mars analogue mission in 2018, in Saint-Guillain (2019) we proposed a first (incomplete) probabilistic formulation, as well as solution method, for the problem of scheduling a set of various human operated projects. In fact, the problem of scheduling a set of operations in a constrained context such as the *Mars Desert Research Station* (MDRS, Fig. 1) is not trivial, even in its classical deterministic version. We hence measured the gains and costs, on a priori mission planning, of robust schedules (optimized under uncertainty) compared to schedules optimized under classical deterministic assumptions.

In Saint-Guillain et al. (2021a), the theoretical insights obtained from the former study were successfully extended to *probabilistic simple temporal networks* (PSTNs), a formalism able to mathematically describe operational problems in general, such as scheduling a space mission or a biomanufacturing campaign. In this paper written with the Jet Propulsion Lab (NASA), our probabilistic model is applied to the operation management of Mars 2020 planetary rover. We also formally define some of the most important theoretical concepts for describing schedule robustness to uncertainty, we introduce new ones, and give

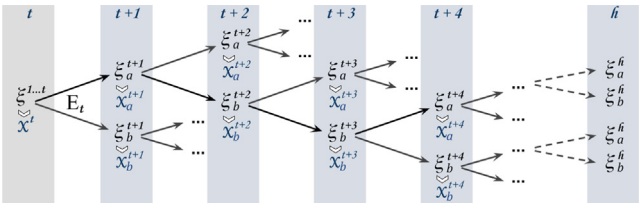

Fig. 9. Tree structure of the problem. The root node represents the current state (past decisions and realizations) at time t. For simplicity, decision (*resp.* random) variables have only two possible choices (*resp.* outcomes).

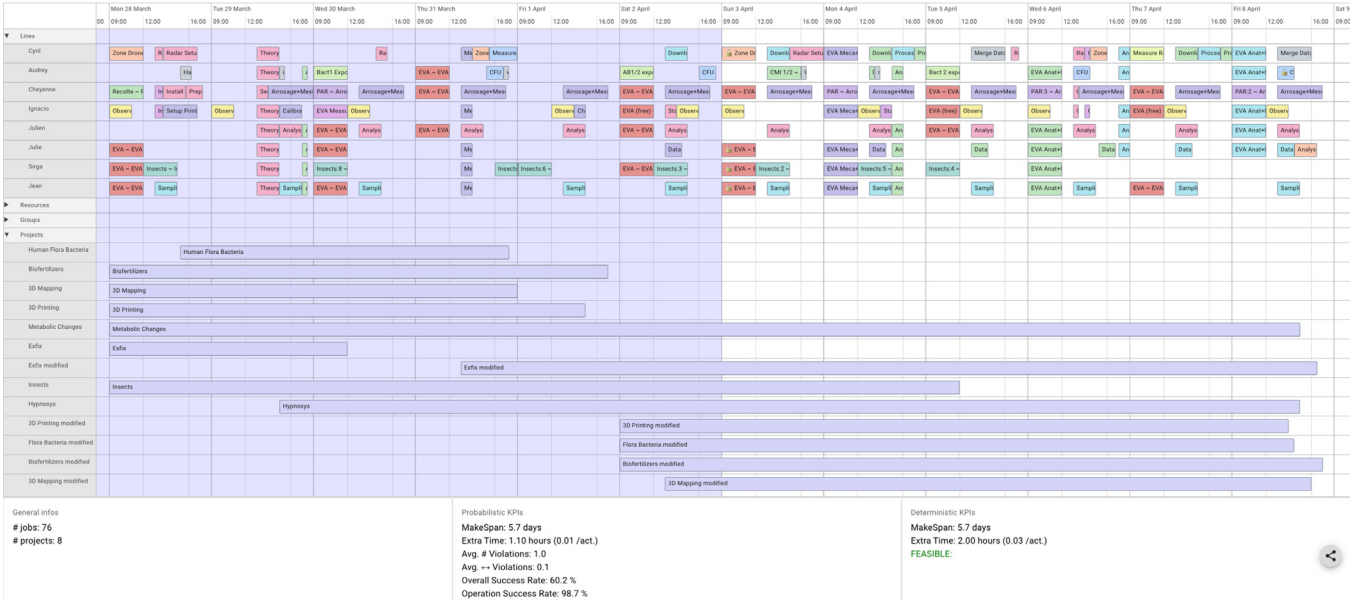

Fig. 10. Current schedule at Sol 6. Many scientific projects had to be remodeled. As the operations are conducted, the project models are likely to be adapted by the astronauts as some constraints or activities reveal to be inadequate to actual real world conditions. For example, the ExFix project has been interrupted on March 30th, and rescheduled from then on, based on a modified model.

proofs for theoretical bounds. This contributed to filling the theoretical gap between specific mission planning and general operations management.

Finally, an asymptotically optimal approach to robustness computation and online reoptimization is briefly explored in Saint-Guillain et al. (2021b). In the later study, the problem of dynamically dispatching the activity execution times is modelled as a *single-player game against Nature*, and solved using Monte-Carlo tree search (MCTS). This only constitutes a preliminary study, which still has to be further studied.

On novice self-scheduling. These previous studies mainly aimed at evaluating and demonstrating, both empirically and theoretically, the need and the advantages of using probabilistic assumptions (*i.e.* optimizing under uncertainty) in the context of operations management. In the context of space exploration, past missions (*e.g.* UCL to Mars 2018, Saint-Guillain (2019) Saint-Guillain (2019)) have shown the importance of online reoptimization and, in particular, the need for the crew to autonomously adapt their science projects to unforeseen events. In the current paper, the scientific focus is rather put on the user experience. More specifically, using techniques from human computer interaction (HCI), we measure how well a team of novice users succeeded (or not) at using our system to schedule, and reschedule online, their own activities.

4.6. A new risk-aversion paradigm

What if analysis and *sensitivity analysis* are classical, well-known techniques for coping with uncertainty in operations management. In fact, Papavasileiou et al. (2007) and Petrides et al. (2014) both argue for the importance of simulation and the ability of performing what-if and/or sensitivity analysis in addition to optimization. What if analysis consists of optimizing several solutions (usually a few numbers), each solving a predefined scenario, such as best-case, average-case and worst-case scenarios. In Saint-Guillain et al. (2021a) we formally prove that the well known *what-if analysis* technique is fundamentally flawed, as it arbitrarily underestimates a schedule's risk. The degree of weak controllability (DWC) can be interpreted as a perfect what-if analysis, that is, when not only considering best, middle and worst cases, but all the possible scenarios. The demonstration is then reached in Saint-Guillain et al. (2021a) at inequality (11), with the result $\mathrm{DDC}(N) \leqslant \mathrm{DWC}(N)$, where DDC (degree of dynamic controllability) is the true maximal robustness of the schedule N. On the contrary, Romie's optimization engine is proven to never underestimate the risk.

Provided a schedule (this however does not help at finding the right schedule in the first place!), *sensitivity analysis* approximates its average quality under uncertainty, how sensible (brittle) it is to stochastic variations. In that sens, the solutions computed by Romie directly optimize their response to a sensitivity analysis. The proposed RAMS framework introduces a new paradigm, replacing both what-if and sensitivity analysis.

5. Astronaut's ability to self-schedule

We finally dive into our actual research question: how well the astronauts succeed at self-scheduling their scientific operations, provided a RAMS decision-support system such as *Romie*. The astronauts were asked to evaluate their experience of the system and the quality of the computed

decisions, before and at different stages during the mission. The a priori stage, before the mission, is called *sol zero* (S_0). A *sol* is a day on Mars. The subsequent stages are S_4, S_8 and S_{12}, for sols four (early mission), eight and twelve (end of the mission).

We collected demographic information from the participants at sol zero (S_0), that is *before the beginning of the mission*. Participants were instructed to complete an UEQ + questionnaire (User Experience Questionnaire) (Schrepp and Thomaschewski, 2019), a modular extension of the UEQ evaluation method in which we selected 12 scales (i.e., ATTRACTIVENESS, EFFICIENCY, PERSPICUITY, DEPENDABILITY, STIMULATION, NOVELTY, TRUST, ADAPTABILITY, USEFULNESS, VISUAL AESTHETICS, INTUITIVE USE, and TRUSTWORTHINESS OF CONTENT) among 20 scales to focus on evaluating the user experience of participants interacting with the system. Each scale is in turn decomposed into four subscales or items to be evaluated (e.g., attractiveness is decomposed into four subscales: annoying vs. enjoyable, bad vs. good, unpleasant vs. pleasant, and unfriendly vs. friendly), each subscale being a differential scale with 7 points between items of each pair (e.g., annoying → enjoyable). We measure each item employing a 7-point Likert-type scale with response categories "Strongly disagree" (=1) to "Strongly agree" (=7).

UEQ+ was selected as an evaluation method because it is a modular and modern interface evaluation method where scales can be decided based on the interface to evaluate and covers the user experience (UX), not just usability, as assessed by questionnaires such as IBM PSSUQ (Lewis, 2006). UEQ+ is also easy to administer to participants and remains valid even with a limited number of participants. For some scales, a benchmarking of their values leads to an interpretation of five effect sizes (Schrepp et al., 2017): bad, below average, above average, good, and excellent. Consequently, for each sol S_0, S_4, S_8, S_{12}, we have two additional dependent variables:

1. The SCALE MEAN SCORE, a real variable measuring the average score obtained on all items of a particular scale, normalized in the interval $[-3, \ldots, +3]$. In order to better classify positive values (above 0) and negative values (below 0), UEQ+ averages all scores of all items, which are ranging from 1 to 7, then reduces them by 4 (-4) to come up with a scale mean score between -3 and $+3$.
2. The SCALE MEAN IMPORTANCE, a real variable measuring the average importance of a particular scale. Similarly to the scale mean score, the scale mean importance is also normalized in the interval $[-3, \ldots, +3]$, as recommended by UEQ+ Schrepp and Thomaschewski (2019).

Participant answers are interpreted with the UEQ data analysis tool. According to (Schrepp and Thomaschewski, 2019), "it is extremely unlikely to observe values above +2 or below −2,..., the standard interpretation of the scale means is that values between −0.8 and 0.8 represent a neutral evaluation of the corresponding scale, values superior to 0.8 represent a positive evaluation, and values inferior to −0.08 represent a negative evaluation". The same interpretation holds for the scale mean importance. Fig. 11 shows the scale means for all sessions S_0, S_4, S_8, S_{12} with their corresponding mean importances.

5.1. Before the mission: S_0

Considering the measurements done at sol zero (S_0), that is *before the beginning of the mission*, only two scales of twelve are negatively assessed in the neutral zone. First, PERSPICUITY ($M = -0.22$, $SD = 1.36$) expresses that the participants did not quickly familiarize themselves with the system, which they nevertheless judged to be very important ($M = 1.88$, $SD = 0.60$), since this was their first discovery of the system. Second, VISUAL AESTHETICS ($M = -0.34$, $SD = 1.63$) was also negatively assessed for user interface aspects estimated unimportant with the lowest score

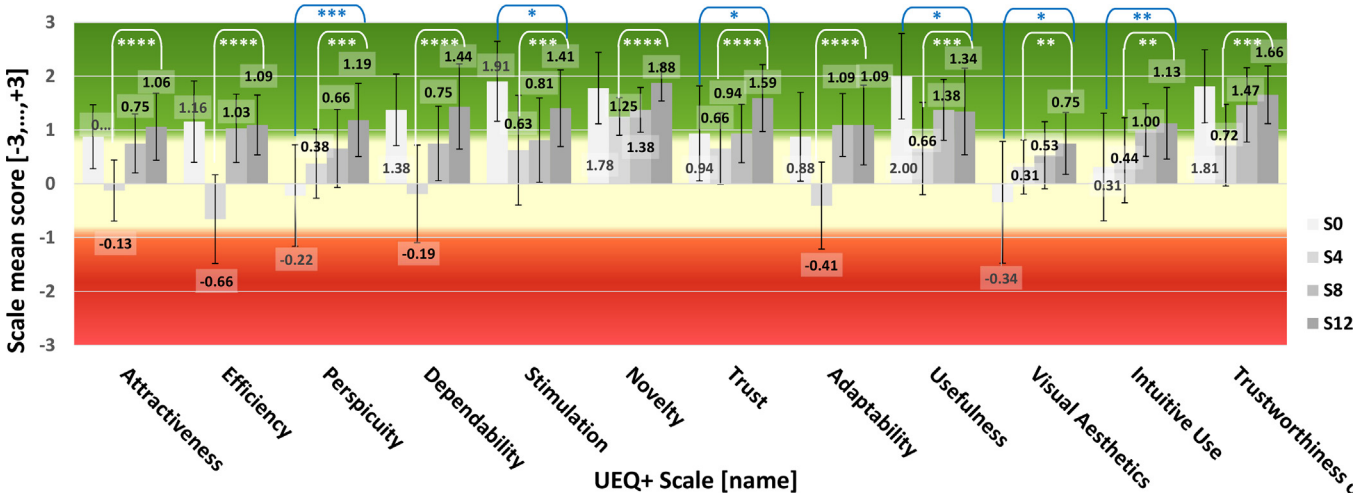

Fig. 11. UEQ+ scale mean scores for all sessions S_i. Significant differences between between S_0 and S_{12} are represented in blue on the top, between S_4 and S_{12} are represented in white. Error bars show a confidence interval of 95%.

($M = -0.38$, $SD = 1.58$). INTUITIVE USE ($M = 0.31$, $SD = 1.45$) is the only scale assessed positively in the neutral zone although very important too ($M = 2.13$, $SD = 0.60$). However, three scales were borderline, that is, ATTRACTIVENESS ($M = 0.86$, $SD = 0.88$), ADAPTABILITY ($M = 0.88$, $SD = 1.19$), and TRUST ($M = 0.94$, $SD = 1.27$), thus reflecting that participants were still not convinced that the system fulfilled their needs for these three important aspects. Fortunately, six of 12 scales are positively assessed, even above the threshold, thus suggesting that the participants felt these aspects are already well fulfilled at first glance: USEFULNESS ($M = 2.00$), STIMULATION ($M = 1.91$), TRUSTWORTHINESS ($M = 1.81$), NOVELTY ($M = 1.78$), DEPENDABILITY ($M = 1.38$), and EFFICIENCY ($M = 1.16$).

5.2. During the mission: S_4, S_8, S_{12}

We now consider the scales for $\boldsymbol{S_4}$, that is, at an early stage of the mission, at the end of the fourth day. Only NOVELTY ($M = 1.25$) exceeds the 0.8 threshold with small dispersion ($SD = 0.50$), thereby meaning that participants recognise that the software was original, partly because they were never confronted to any similar software. Seven of 12 scales are positively assessed in the neutral interval, representing a slight improvement with respect to S_0: TRUSTWORTHINESS OF CONTENT ($M = 0.72, SD = 1.10$) TRUST ($M = 0.66$), USEFULNESS ($M = 0.66$), STIMULATION ($M = 0.63$), INTUITIVE USE ($M = 0.44$), PERSPICUITY ($M = 0.38$), and VISUAL AESTHETICS ($M = 0.31$). The three most positive scales refer to the utility character of the application, which is considered as the most important part. Four scales are negatively assessed in the neutral interval, thus calling for improvement: EFFICIENCY ($M = -0.66$), ADAPTABILITY ($M = -0.41$), DEPENDABILITY ($M = -0.19$), and ATTRACTIVENESS ($M = -0.13$).

While NOVELTY received the highest mean score, it also received the lowest importance ($M = -0.13$), because participants become more accustomed with the software and therefore reduce its importance over time. Similarly, VISUAL AESTHETICS ($M = 0.25$) were no longer considered as important as before. Utility scales take precedence over usability scales in terms of mean importance: EFFICIENCY ($M = 2.38$), USEFULNESS ($M = 2.25$), PERSPICUITY ($M = 2.13$), TRUSTWORTHINESS OF CONTENT ($M = 1.88$), ADAPTABILITY ($M = 1.88$), DEPENDABILITY ($M = 1.75$), INTUITIVE USE ($M = 1.75$), TRUST ($M = 1.50$), ATTRACTIVENESS ($M = 1.13$), and STIMULATION ($M = 0.88$).

For the first time, at sol eight $\boldsymbol{S_8}$, all scales become positively assessed with only four belonging to the neutral zone: ATTRACTIVENESS ($M = 0.75$), DEPENDABILITY ($M = 0.75$), PERSPICUITY ($M = 0.66$), and VISUAL AESTHETICS ($M = 0.53$). The remaining eight scales are located above the threshold: TRUSTWORTHINESS OF CONTENT ($M = 1.47$), USEFULNESS ($M = 1.38$), ADAPTABILITY ($M = 1.09$), NOVELTY ($M = 1.38$), EFFICIENCY ($M = 1.03$), INTUITIVE USE ($M = 1.00$), TRUST ($M = 0.94$), and STIMULATION ($M = 0.81$). TRUSTWORTHINESS received the highest scale means and a high importance ($M = 1.75$), thereby suggesting that participants progressively acquire more trust in manipulating the data. The functions attached to these data are well perceived based on USEFULNESS with a high importance ($M = 2.00$). EFFICIENCY ($M = 2.13, SD = 0.60$) was rated the most important factor although its scale was not the highest one.

The last session $\boldsymbol{S_{12}}$ (sol twelve, last day of the mission) obtained all scale means above the threshold, thereby indicating the most positive appreciation of the software, except for VISUAL AESTHETICS ($M = 0.75$, $SD = 0.83$), which is also consistently rated as the least important factor ($M = 0.63$, $SD = 1.32$). Surprisingly, NOVELTY obtained the highest scale mean ($M = 1.88$) with the smallest deviation ($SD = 0.48$), with a moderate importance ($M = 1.75$, $SD = 0.66$), thus suggesting that participants estimate that the software stays original, even after several usages. TRUSTWORTHINESS OF CONTENT remains the second highest scale means ($M = 1.66$, $SD = 0.77$) as it was the case before, with a very high importance ($M = 2.00$, $SD = 1.00$). ATTRACTIVENESS ($M = 1.06$, $SD = 0.90$) suffered from the lowest mean with the second lowest importance rate ($M = 1.13$, $SD = 1.36$), thus suggesting that this factor does not deteriorate much the overall software quality. Just before this factor, ADAPTABILITY ($M = 1.09$, $SD = 1.07$) and EFFICIENCY ($M = 1.09$, $SD = 0.80$) share the second lowest scale mean, with the highest importance for EFFICIENCY though ($M = 2.38$, $SD = 0.48$).

5.3. Inter-session evolution: Results and discussion

In this section, we first use statistical tools (inter-rater agreement, inter-rater consistency) to address two questions: Do we have obvious consensus, or did the astronauts answer independently?; Did the participants answer in a random fashion, or based on logical assumptions? We use two statistical tools: Kendall's coefficient of agreement, and Cronbach's coefficient of consistency. The later measures how relevant is the measurement tool (form), while the first quantifies the quality of the sample (group of participants). Finally, we focus on *the evolution of the results*, during the mission, from Sol 0 (before the mission) to Sol 12 (end of the mission).

5.3.1. Inter-rater agreement

Table A.4 reports Kendall's coefficient of concordance W (Legendre, 2005), a measure of agreement among raters which is equal to 0 when there is no agreement among them. The lower the concordance, the more heterogeneous is the survey sample (participants). A high coefficient reflects either one (or several) of the following facts: the sample is too small, the participants were selected with a

bias, the participants communicated while answering the questions.

Although all W values are positive, some of them are low, indicating that there is limited agreement (e.g., DEPENDABILITY in S_0 received $W = 0.041$ interpreted as poor agreement and VISUAL AESTHETICS in S_0 received $W = 0.21$ interpreted as fair agreement). Some others depart more from 0, but rarely in a significant way. In particular, USEFULNESS in S_0 received $W = 0.34$ with $p = .041^*$, which is the scale benefiting the most from inter-rater agreement in a significant way, thus rejecting the null hypothesis that there is no agreement among participants. W slightly decreases across sessions, but stays interpreted as 'fair.' Another example is EFFICIENCY, which received a fair agreement ($W = 0.40, p = .021^*$) for S_4, but decreases over sessions. The limited agreement can be partially explained by the diversity of the profiles of the participants, but also by the varying experimental conditions: S_0 was carried out in a room with limited pressure, while S_4, S_8, and S_{12} were carried out with more mental, temporal, and physical pressure.

5.3.2. Inter-rater consistency

Table A.2 reports Cronbach's α coefficient computed to quantify the internal consistency, which expresses the extent to which the scale measurements remain consistent within a session or over subsequent sessions under identical or different conditions. This test measures how the different components of the form permit to well reflect the global user experience, or in contrary, if the questions asked in the form are not relevant. A value above 0.7 is considered as acceptable (Nunnally, 1975), meaning that the scale is most likely to be relevant for the study.

Again, S_0 was conducted in lab, while S_4 to S_{12} were conducted under experimental conditions mimicking the target real conditions. Individually speaking, scales' α range from an unacceptable interpretation (e.g., PERSPICUITY received $\alpha = 0.48$ for S_8) to an excellent interpretation (e.g., ADAPTABILITY received $\alpha = 0.93$ for S_{12}). S_0 obtained 8 values above the 0.7 threshold and 4 values below. The first real condition session, i.e. S_4, obtained a balanced consistency: 6 values above the threshold and 6 below. This balance evolves positively across sessions in favour of a consistency above the threshold: 7 above and 5 below for S_8 to 9 above and 3 below for S_{12}, thus suggesting that participants rated scales more consistently over time. Indeed, the mean α starts at 0.79 (acceptable) for S_0 and always improves session after session: 0.61 (questionable) for S_4, 0.70 (acceptable again) for S_8 and 0.80 (good) for the final S_{12}.

5.3.3. Evolution of scales and importance rates

Fig. 11 and Fig. 12 show how the mean scores and importance evolved across all four sessions. Overall, most scales obtained a high mean for the first S_0, which dramatically decreased for S_4 carried out in real conditions, revealing a different appreciation of the software between the ideal conditions *in vitro* and the real conditions *in* vivo. Fortunately, these mean scores positively evolved until reaching positive values above the threshold during the last session. These results suggest that participants, although they were probably influenced by the difficult conditions of S_4, progressively improved their assessment, being less influenced by these contextual constraints and more accustomed to deal with them. The results obtained for the last sessions S_{12} therefore represent an overall stable assessment of the software after several continuous usages.

More precisely, Table A.3 shows how scale mean scores and their mean importance evolved in terms of difference of percentage between sessions: first, between the initial *in vitro* S_0 session and the first *in vivo* S_4 session, then between the two next iterations and, finally, between the first session S_0 and the last session S_{12}. Some scales largely improved since the beginning: PERSPICUITY received the best mean gain from one session to the last ($\Delta = 643\%$), followed by VISUAL AESTHETICS ($\Delta = 318\%$) and INTUITIVE USE ($\Delta = 260\%$), suggesting that the user experience gained

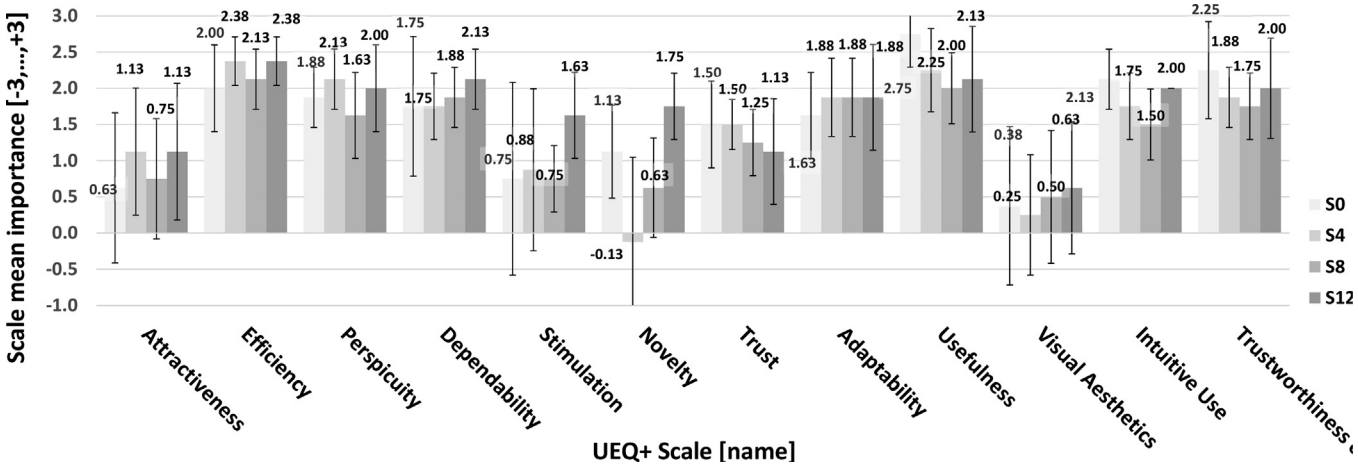

Fig. 12. UEQ+ mean importance for all sessions S_i. Error bars show a confidence interval of 95%.

during the sessions positively impacted these scale means, even if their mean importance changed over time. Four scales decreased between the first and the last session: USEFULNESS is reduced by $\Delta = -33\%$, followed by STIMULATION by $\Delta = -26\%$, TRUSTWORTHINESS OF CONTENT by $\Delta = -9\%$, and EFFICIENCY by $\Delta = -5\%$, suggesting that participants expressed their needs at a higher level of expectation during the first session than during the last one. This does not depreciate the overall user experience of the interface, but indicates that the experience accumulated by participants let them to adjust their assessment more precisely since all scale means in S_{12} were highly positive (Fig. 11). Participants also increased their importance rates of nine scales and decreased the rates for three scales only: TRUST by $\Delta = -25\%$, USEFULNESS by $\Delta = -23\%$, and TRUSTWORTHINESS OF CONTENT by $\Delta = -11\%$, suggesting that participants have lowered the importance due to the experience gained and the rapid learning curve. The progress acquired during successive sessions is therefore a determining factor for the adjustment of the scales and their importance to converge towards an equilibrium representing a stable value after a continuous interaction. We investigated whether these differences are statistically significant by computing a Wilcoxon signed-rank test for paired samples between S_0 and S_{12}, then between S_4 and S_{12}.

Between S_0 and S_{12}. PERSPICUITY is significantly lower ($p = .00064^{***}$) with a small effect size ($r = 0.34$), thus suggesting that participants felt that they were in control much more in the end of the mission than before ((Fig. 11-blue top bars); STIMULATION is significantly smaller ($p = .049^{*}$) with a small effect size ($r = 0.21$), TRUST is significantly smaller ($p = .013^{*}$) with a small effect size ($r = 0.28$), USEFULNESS is significantly larger ($p = .020^{*}$) with a small effect size ($r = 0.25$), VISUAL AESTHETICS are significantly smaller ($p = .0037^{**}$) with a small effect size ($r = 0.33$), INTUITIVE USE is significantly larger ($p = .0049^{**}$) with a small effect size ($r = 0.32$). With respect to importance, no significant difference was found between means of all corresponding scales, thus suggesting that participants estimated the importance of the respective scales not in a very different way.

Between S_4 and S_{12}. Many scales saw their mean scores significantly higher in S_{12} than in S_4 (Fig. 11-white): ATTRACTIVENESS is significantly different ($p \leqslant .0001^{****}$) with a medium effect size ($r = 0.51$), EFFICIENCY is different ($p \leqslant .0001^{****}$) with a medium effect size ($r = 0.54$), PERSPICUITY is different ($p = .00033^{***}$) with a small effect size ($r = 0.42$), DEPENDABILITY is different ($p \leqslant .0001^{****}$) with a small effect size ($r = 0.48$), STIMULATION is different ($p = .00047^{***}$) with a small effect size ($r = 0.40$), NOVELTY is different ($p \leqslant .0001^{****}$) with a small effect size ($r = 0.46$), TRUST is different ($p \leqslant .0001^{****}$) with a small effect size ($r = 0.47$), ADAPTABILITY is different ($p \leqslant .0001^{****}$) with a medium effect size ($r = 0.50$), USEFULNESS is different ($p = .0062^{***}$) with a medium effect size ($r = 0.31$), VISUAL AESTHETICS are different ($p = .01007^{*}$) with a small effect size ($r = 0.29$), INTUITIVE USE is different ($p = .0076^{**}$) with a small effect size ($r = 0.30$), TRUSTWORTHINESS is different ($p = .00028^{***}$) with a small effect size ($r = 0.41$). The effect size is interpreted as 'medium' more frequently between S_4 and S_{12} than between S_0 and S_{12} or as 'small'. With respect to importance, a significant difference was found between means only for NOVELTY ($p = .15^{*}$) with a large effect size ($r = .52$).

5.4. Benchmarking of scales

Each UEQ+ scale is typically evaluated as follows: between -0.8 and 0.8 for a neutral evaluation, superior to 0.8 for a positive evaluation, and inferior to -0.08 for a negative evaluation, as explained in the end of Section 5. Beyond this universal evaluation, Schrepp et al. (2017) mentions some more precise intervals for interpreting some of these scales based on a benchmarking obtained by observing the distribution of the values over a large set of evaluated cases. The mean value of each benchmarked scale therefore falls into one of five categories defined as follows (Schrepp et al., 2017): *excellent* (among the best 10% of all cases), *good* (10% of the cases are better than the evaluated product), *above average* (25% of the cases are better than the evaluated product), *below average* (50% of the cases are better than the evaluated product), and *bad* (the evaluated product is among the worst 25% of cases).

Fig. 13 shows the distribution of the benchmarked scales according to the five categories and compares it with PlayBook (Shelat et al., 2022; Shelat et al., 2022), another operations management system, developed and tested by NASA in analogue conditions. Playbook is one of the three components of the Minerva suite (Section 2.2). Scales for S_{12} are benchmarked as follows: ATTRACTIVENESS is 'excellent' ($M = 1.06 \geqslant 0.75$) as well as PlayBook ($M = 1.53 \geqslant 0.75$), PERSPICUITY is 'excellent' ($M = 1.19 \geqslant 0.6$), EFFICIENCY is 'excellent' ($M = 1.09 \geqslant 0.72$), DEPENDABILITY

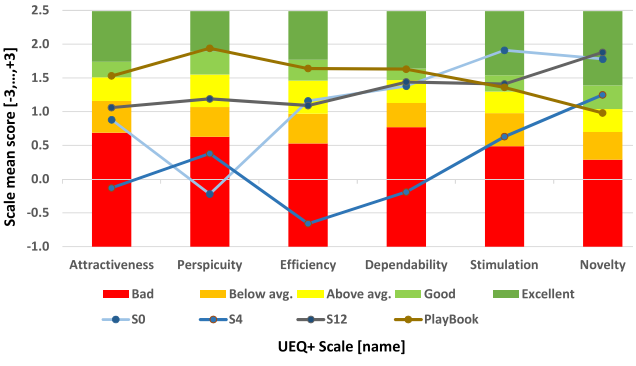

Fig. 13. Benchmarking of UEQ+ scales for each session with respect to PlayBook (Shelat et al., 2022) in brown. S_4, S_8, and S_{12} are represented in light blue, medium blue, and dark blu.e, respectively.

is 'excellent' ($M = 1.44 \geqslant 0.85$), STIMULATION is 'excellent' ($M = 1.41 \geqslant 0.95$) and slightly above PlayBook ($M = 1.36 \geqslant 0.75$), and NOVELTY is 'excellent' ($M = 1.88 \geqslant 1.1$) and above PlayBook ($M = 0.98 \leqslant 0.75$, interpreted as 'good'). Overall, Playbook provides significantly better PERSPICUITY. Compared to Playbook, *Romie* has additional features, such as the visual modeling framework and the (re) optimization engine. These come at the cost of a slightly increased complexity of the system from the user's point of view at discovery stage. After a training period of a couple of weeks, these features no longer affect the usability as its factors are no longer deteriorated.

6. Conclusion and future work

We study the capability of a crew of analogue astronauts, composed of novice planners, to manage the operational schedule of their mission in an autonomous setup, by using a computer-aided decision system. Techniques from human computer interaction (HCI) were exploited to measure and analyse how well the participants succeed at doing so: the astronauts were asked to evaluate their experience of the system and the quality of the computed decisions using UEQ+.

The results gathered before, and at different stages of the mission, show that the proposed decision system appears as being an adequate approach, from a functional point of view (usefulness), whereas it is perceived as difficult to use by the participants, especially during the first days of the mission.

Empirical evidence has shown that even provided a strong provisional schedule, rethinking and reshaping all the a priori decisions related to the research projects, to be carried out during the mission, is unavoidable. As activities take place, the scientific objectives and constraints must be adapted according to unpredictable events. EVAs must be cancelled due to bad weather conditions. The entire project must be adapted to fit the limited duration of the mission. Fig. 14 shows the planning at the end of the mission. Due to the inherent complexity of the underlying combinatorial problem, modifying the schedule by hand is not an option. To that extent, the tested decision system includes an artificial intelligence, which proved its usefulness by computing optimised solutions to the scheduling problem, for the astronauts, based on a graphical description of their objectives and constraints. The main limitation of the approach lies in the learning time required by the participants to master the system. Future missions will need a more adequate preparation.

Astronaut qualitative impression. Finally, apart from the quantitative analysis of the questionnaires, the eight astronauts have been asked to describe their impression on the technology, what they liked or disliked: "*We believe that Romie is a very useful program for this type of mission, which involves a lot of constraints simultaneously in terms of personnel, time and equipment. Although it was not yet fully functional at the beginning of the mission, it was able to perform a maximum number of pre-programmed activities. The interface could be improved to become more intuitive. Indeed, without adequate prior training, it is very difficult to get used to it, which was relatively energy consuming and time consuming at the beginning. However, as the mission evolved, the program was continually readapted to finally offer us the adequate and appreciated handling for the autonomous management of operations.*"

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

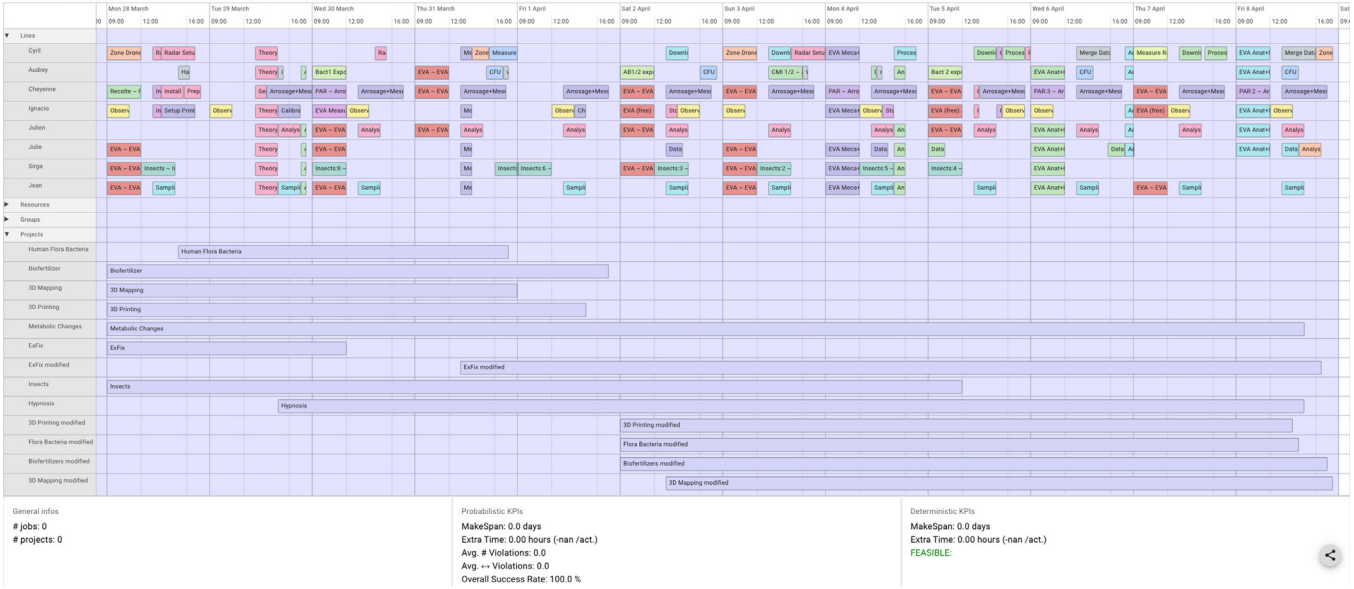

Fig. 14. Final schedule at the end of the mission (Sol 12). In the end, the astronauts managed to complete all the scientific projects, even though several projects had to be adapted during the course of the operations to fit the field realities.

Acknowledgements

We thank the Mars Society, Shannon Rupert and Mission Support personnel, for creating the conditions for realistic Mars analogue sojourns. We thank Agathe Florio and Nathan Gurnet for their helpful feedback on statistics. Portions of this work were performed by the Jet Propulsion Laboratory, California Institute of Technology under a contract with the National Aeronautics and Space Administration (80NM0018D0004). Nicolas Burny and Jean Vanderdonckt are supported by the EU EIC Pathfinder-Awareness Inside challenge "Symbiotik" project under Grant No. 101071147.

Appendix A. Evaluation results tables

See Tables A.2, A.3, A.4.

Table A.2
Consistency reliability. Cronbach's α : $\geqslant 0.9$=excellent (E), $0.9 > \alpha \geqslant 0.8$=good (G), $0.8 > \alpha \geqslant 0.7$=acceptable (A), $0.7 > \alpha \geqslant 0.6$=questionable (Q), $0.6 > \alpha \geqslant 0.5$=poor (P), $0.5 > \alpha$=unacceptable (U).

Scale	Cronbach's α (interpretation)			
	S_0	S_4	S_8	S_{12}
Attractiveness	0.62 (Q)	0.68 (Q)	0.55 (P)	0.91 (E)
Efficiency	0.46 (U)	−0.10 (U)	0.70 (A)	0.52 (P)
Perspicuity	0.95 (E)	0.56 (P)	0.48 (U)	0.78 (A)
Dependability	0.67 (Q)	0.75 (A)	0.72 (A)	0.67 (Q)
Stimulation	0.94 (E)	0.96 (E)	0.93 (E)	0.89 (G)
Novelty	0.89 (G)	0.63 (Q)	0.63 (Q)	0.63 (Q)
Trust	0.76 (A)	−0.25 (U)	0.82 (G)	0.88 (G)
Adaptability	0.85 (G)	0.80 (G)	0.68 (Q)	0.93 (E)
Usefulness	0.69 (Q)	0.91 (E)	0.81 (G)	0.88 (G)
Visual Aesthetics	0.96 (E)	0.70 (A)	0.82 (G)	0.76 (A)
Intuitive Use	0.88 (G)	0.53 (P)	0.53 (P)	0.83 (G)
Trustworthiness	0.76 (A)	0.55 (P)	0.77 (A)	0.84 (G)
Mean	0.79 (A)	0.61 (Q)	0.70 (A)	0.80 (G)

Table A.3
Evolution of scale means and their importance rates across sessions in terms of difference of percentage (M = mean scale, Imp.=importance rate).

Scale	$S_0 \rightarrow S_4$		$S_4 \rightarrow S_8$		$S_8 \rightarrow S_{12}$		$S_0 \rightarrow S_{12}$	
	M	Imp.	M	Imp.	M	Imp.	M	Imp.
Attractiveness	−114%	80%	700%	−33%	42%	50%	21%	80 %
Efficiency	−157%	19%	257%	−11%	6%	12%	−5%	19 %
Perspicuity	271%	13%	75%	−24%	81%	23%	643%	7 %
Dependability	−114%	0%	500%	7%	92%	13%	5%	21 %
Stimulation	−67%	17%	30%	−14%	73%	117%	−26%	117 %
Novelty	−30%	−111%	10%	600%	36%	180%	5%	56 %
Trust	−30%	0%	43%	−17%	70%	−10%	70%	−25 %
Adaptability	−146%	15%	369%	0%	0%	0%	25%	15 %
Usefulness	−67%	−18%	110%	−11%	−2%	6%	−33%	−23 %
Visual Aesthetics	−191%	−33%	70%	100%	41%	25%	318%	67 %
Intuitive Use	40%	−18%	129%	−14%	13%	33%	260%	−6 %
Trustworthiness	−60%	−17%	104%	−7%	13%	14%	−9%	−11%

Table A.4
Inter-rater agreement. Kendall's $W \leqslant 0.2$=low (L), $0.21 \leqslant W \leqslant 0.4$=fair (F), $0.41 \leqslant W \leqslant 0.6$=moderate (M), $0.61 \leqslant W \leqslant 0.8$=high (S), $0.81 \leqslant W \leqslant 1$=very high (V).

Scale	Kendall's W (p-value, interpretation)				
	S_0	S_4	S_8	S_{12}	Mean
Attractiveness	0.14 (0.35, L)	0.31 (0.057, F)	0.094 (0.52, L)	0.08 (0.59, L)	0.156 (L)
Efficiency	0.098 (0.50, L)	0.40 (0.021, F)	0.034 (0.84, L)	0.023 (0.90, L)	0.14 (L)
Perspicuity	0.11 (0.43, L)	0.0394 (0.81, L)	0.18 (0.23, L)	0.12 (0.40, L)	0.11 (L)
Dependability	0.041 (0.80, L)	0.21 (0.15, F)	0.16 (0.26, L)	0.11 (0.44, L)	0.13 (L)
Stimulation	0.16 (0.26, L)	0.077 (0.60, L)	0.095 (0.51, L)	0.023 (0.90, L)	0.08 (L)
Novelty	0.042 (0.79, L)	0.014 (0.95, L)	0.03 (0.87, L)	0.13 (0.37, L)	0.05 (L)
Trust	0.20 (0.17, L)	0.02 (0.92, L)	0.05 (0.75, L)	0.058 (0.71, L)	0.08 (L)
Adaptability	0.11 (0.43, L)	0.058 (0.70, L)	0.045 (0.78, L)	0.056 (0.72, L)	0.07 (L)
Usefulness	0.34 (0.041, F)	0.25 (0.11, F)	0.13 (0.36, L)	0.26 (0.10, F)	0.24 (F)
Visual Aesthetics	0.21 (0.16, F)	0.15 (0.32, L)	0.039 (0.81, L)	0.10 (0.48, L)	0.12 (L)
Intuitive Use	0.17 (0.24, L)	0.084 (0.57, L)	0.13 (0.36, L)	0.028 (0.88, L)	0.10 (L)
Trustworthiness	0.077 (0.60, L)	0.24 (0.12, F)	0.14 (0.34, L)	0.22 (0.14, F)	0.17 (L)
Mean	0.14 (L)	0.15 (L)	0.09 (L)	0.10 (L)	

References

Agrawal, J., Chi, W., Chien, S., et al., 2021a. Enabling limited resource-bounded disjunction in scheduling. J. Aerospace Informat. Syst. 18: 6 (6), 322–332. https://doi.org/10.2514/1.I010908.

Agrawal, J., Chi, W., Chien, S.A., et al., 2021b. Analyzing the effectiveness of rescheduling and flexible execution methods to address uncertainty in execution duration for a planetary rover. Robot. Auton. Syst. 140, 103758. https://doi.org/10.1016/j.robot.2021.103758.

Agrawal, J., Yelamanchili, A., Chien, S., 2020. Using explainable scheduling for the Mars 2020 Rover Mission. In: Proc. of the Workshop on Explainable AI Planning, International Conference on Automated Planning and Scheduling ICAPS XAIP '20. https://doi.org/10.48550/arXiv.2011.08733.

Ai-Chang, M., Bresina, J., Charest, L., et al., 2004. MAPGEN: mixed-initiative planning and scheduling for the Mars exploration rover mission. IEEE Intell. Syst. 19 (1), 8–12. https://doi.org/10.1109/MIS.2004.1265878.

Applegate, D., Cook, W., 1991. A computational study of the job-shop scheduling problem. ORSA J. Comput. 3 (2), 149–156. https://doi.org/10.1287/ijoc.3.2.149.

Bell, E., Coan, D., 2012. A review of the approach to iss increment crew eva training. In: Proc. of AIAA SPACE 2007 Conference & Exposition, p. 6236. https://doi.org/10.2514/6.2007-6236.

Boche-Sauvan, L., Pletser, V., Foing, B. et al., 2009a. Human aspects study through industrial methods during an mdrs mission. In: NASA Lunar Science Forum, NASA Ames Conference Centre.

Boche-Sauvan, L., Pletser, V., Foing, B. et al., 2009b. Human aspects and habitat studies from eurogeomars campaign. In: EGU General Assembly Conference Abstracts, p. 13323.

Brady, A.L., Kobs Nawotniak, S.E., Hughes, S.S., et al., 2019. Strategic planning insights for future science-driven extravehicular activity on mars. Astrobiology 19 (3), 347–368. https://doi.org/10.1089/ast.2018.1850.

Cesta, A., Cortellessa, G., Denis, M., et al., 2007. Mexar2: Ai solves mission planner problems. IEEE Intell. Syst. 22 (4), 12–19. https://doi.org/10.1109/MIS.2007.75.

Chappell, S.P., Beaton, K.H., Newton, C., et al., 2017. Integration of an earth-based science team during human exploration of mars. In: Proc. of the IEEE Aerospace Conference. IEEE, pp. 1–11. https://doi.org/10.1109/AERO.2017.7943727.

Chi, W., Agrawal, J., Chien, S. et al., 2019. Optimizing parameters for uncertain execution and rescheduling robustness. In: Proc. of Int. Conf. on Automated Planning and Scheduling. Berkeley, CA, USA volume 29 of ICAPS 2019. https://doi.org/10.1609/icaps.v29i1.3552.

Chien, S., Johnston, M., Policella, N. et al., 2012. A generalized timeline representation, services, and interface for automating space mission operations. In: Proc. of 12th International Conference of Space Operations SpaceOps 2012. URL: https://ai.jpl.nasa.gov/public/documents/papers/chien_spaceops2012_generalized.pdf.

Chien, S., Rabideau, G., Knight, R., et al., 2000. Aspen-automating space mission operations using automated planning and scheduling. In: SpaceOps 2000. AIAA Press, Toulouse, France, URL: https://ai.jpl.nasa.gov/public/projects/aspen/.

Chien, S., Rabideau, G., Willis, J., et al., 1999. Automating planning and scheduling of shuttle payload operations. Artif. Intell. 114 (1–2), 239–255. https://doi.org/10.1016/S0004-3702(99)00069-7, URL: https://www.sciencedirect.com/science/article/pii/S0004370299000697.

Chien, S., Sherwood, R., Tran, D., et al., 2005. Using autonomy flight software to improve science return on earth observing one. J. Aerospace Comput. Informat. Commun. 2 (4), 196–216. https://doi.org/10.2514/1.12923, URL: https://arc.aiaa.org/doi/10.2514/1.12923.

Chien, S.A., Rabideau, G., Tran, D.Q., et al., 2021. Activity-based scheduling of science campaigns for the rosetta orbiter. J. Aerospace Informat. Syst. 18 (10), 711–727. https://doi.org/10.2514/1.I010899.

Clancey, W.J., 2006. Participant observation of a mars surface habitat mission simulation. Habitation 11 (1), 27–47. https://doi.org/10.3727/154296606779507132.

Deans, M., Marquez, J.J., Cohen, T., et al., 2017. Minerva: user-centered science operations software capability for future human exploration. In: Proc. of IEEE Aerospace Conference. IEEE, pp. 1–13. https://doi.org/10.1109/AERO.2017.7943609.

Drake, B.G., Hoffman, S.J., Beaty, D.W., 2010. Human exploration of mars, design reference architecture 5.0. In: Proc. of IEEE Aerospace Conference. IEEE, pp. 1–24. https://doi.org/10.1109/AERO.2010.5446736.

Drake, B.G., Watts Kevin, D., 2014. Human exploration of mars design reference architecture 5.0, addendum # 2.

Eppler, D., Adams, B., Archer, D., et al., 2013. Desert research and technology studies (drats) 2010 science operations: Operational approaches and lessons learned for managing science during human planetary surface missions. Acta Astronaut. 90 (2), 224–241. https://doi.org/10.1016/j.actaastro.2012.03.009.

Fukunaga, A., Rabideau, G., Chien, S. et al., 1997. Towards an application framework for automated planning and scheduling. In: Proc. of IEEE Aerospace Conference, vol. 1, IEEE, pp. 375–386. https://doi.org/10.1109/AERO.1997.574426.

Gaines, D., Doran, G., Justice, H. et al., 2016. Productivity challenges for mars rover operations: A case study of mars science laboratory operations. Technical Report D-97908, Jet Propulsion Laboratory,

URL: https://ai.jpl.nasa.gov/public/documents/papers/gaines_report_roverProductivity.pdf.

Gaines, D., Doran, G., Paton, M., et al., 2020. Self-reliant rovers for increased mission productivity. J. Field Robot. 37 (7), 1171–1196. https://doi.org/10.1002/rob.21979.

Hall, N.G., Magazine, M.J., 1994. Maximizing the value of a space mission. Eur. J. Oper. Res. 78 (2), 224–241. https://doi.org/10.1016/0377-2217(94)90385-9.

International Standard Organization, 2019. ISO/IEC 9421–210:2019, ergonomics of human-system interaction — part 210: Human-centred design for interactive systems.

Johnston, M., Miller, G., 1994. Intelligent scheduling. S Pike: Intelligent scheduling of Hubble Space Telescope observations, pp. 391–422.

Jónsson, A.K., Morris, P.H., Muscettola, N. et al., 2000. Planning in interplanetary space: Theory and practice. In: Chien, S.A., Kambhampati, S., Knoblock, C.A. (Eds.), Proceedings of the Fifth International Conference on Artificial Intelligence Planning Systems, AAAI. pp. 177–186.

Kleywegt, A.J., Shapiro, A., Homem-de Mello, T., 2002. The sample average approximation method for stochastic discrete optimization. SIAM J. Optim. 12 (2), 479–502. https://doi.org/10.1137/S1052623499363220.

Legendre, P., 2005. Species associations: the kendall coefficient of concordance revisited. J. Agric. Biol. Environ. Stat. 10 (226). https://doi.org/10.1198/108571105X46642.

Lenstra, J.K., Kan, A.R., 1979. Computational complexity of discrete optimization problems. In: Annals of Discrete Mathematics, vol. 4. Elsevier, pp. 121–140, https://doi.org/10.1016/S0167-5060(08)70821-5.

Lewis, J.R., 2006. Sample sizes for usability tests: Mostly math, not magic. Interactions 13 (6), 29–33. https://doi.org/10.1145/1167948.1167973.

Manon, J., Saint-Guillain, M., Pletser, V. et al., 2023. Astronaut's fast learning to treat a tibial shaft fracture: A mars analogue simulation. npj Microgravity (submitted).

Marquez, J.J., Edwards, T., Karasinski, J.A., et al., 2021. Human performance of novice schedulers for complex spaceflight operations timelines. Hum. Factors. https://doi.org/10.1177/00187208211058913, URL: https://journals.sagepub.com/doi/full/10.1177/00187208211058913.

Marquez, J.J., Hillenius, S., Kanefsky, B., et al., 2017. Increasing crew autonomy for long duration exploration missions: Self-scheduling. In: Proc. of IEEE Aerospace Conference. IEEE, pp. 1–10. https://doi.org/10.1109/AERO.2017.7943838.

Marquez, J.J., Miller, M.J., Cohen, T., et al., 2019. Future needs for science-driven geospatial and temporal extravehicular activity planning and execution. Astrobiology 19 (3), 440–461. https://doi.org/10.1089/ast.2018.1838.

Miller, M.J., McGuire, K.M., Feigh, K.M., 2015. Information flow model of human extravehicular activity operations. In: 2015 IEEE Aerospace Conference. IEEE, pp. 1–15. https://doi.org/10.2514/6.2007-6236.

Mishkin, A., Lee, Y., Korth, D., et al., 2007. Human-robotic missions to the moon and mars: Operations design implications. In: Proc. of IEEE Aerospace Conference. IEEE, pp. 1–10. https://doi.org/10.1109/AERO.2007.352960.

Nunnally, J.C., 1975. Psychometric theory—25 years ago and now. Educ. Researcher 4 (10), 7–21. https://doi.org/10.3102/0013189X004010007.

Papavasileiou, V., Koulouris, A., Siletti, C., et al., 2007. Optimize manufacturing of pharmaceutical products with process simulation and production scheduling tools. Chem. Eng. Res. Des. 85 (7), 1086–1097. https://doi.org/10.1205/cherd06240.

Petrides, D., Carmichael, D., Siletti, C., et al., 2014. Biopharmaceutical process optimization with simulation and scheduling tools. Bioengineering 1 (4), 154–187. https://doi.org/10.3390/bioengineering1040154.

Pletser, V., 2010a. A Mars Human Habitat: Recommendations on crew time utilisation and habitat interfaces. J. Cosmol. Special Issue 'The Human Mission Mars: Colonizing Red Planet' 12, 3928–3945, URL: https://thejournalofcosmology.com/Mars123.html.

Pletser, V., 2010b. Crew time utilisation and habitat interface investigations for future planetary habitat definition studies: field tests at mdrs. 38th COSPAR Scientific. Assembly 38, 2, URL: https://ui.adsabs.harvard.edu/abs/2010cosp...38.435P/abstract.

Pletser, V., Boche-Sauvan, L., Foing, B., et al., 2009. European contribution to human aspect field investigation for future planetary habitat definition studies: field tests at mdrs on crew time utilization and habitat interfaces. ELGRA Biannual Symposium 'In the Footsteps of Columbus', Bonn, Germany, vol. 26, p. 64.

Pletser, V., Foing, B., 2011. European contribution to human aspect investigations for future planetary habitat definition studies: Field tests at mdrs on crew time utilisation and habitat interfaces. Microgravity Sci. Technol. 23 (2), 199–214. https://doi.org/10.1007/s12217-010-9251-4.

Rabideau, G., Benowitz, E., 2017. Prototyping an onboard scheduler for the mars 2020 rover. In: Proc. of International Workshop on Planning and Scheduling for Space IWPSS 2017. Pittsburgh, PA, USA. URL: http://hdl.handle.net/2014/47716.

Rabideau, G., Knight, R., Chien, S. et al., 1999. Iterative repair planning for spacecraft operations using the aspen system. In: Artificial Intelligence, Robotics and Automation in Space, vol. 440. p. 99, URL: https://ai.jpl.nasa.gov/public/papers/search-isairas99.ps.

Saint-Guillain, M., 2019. Robust operations management on mars. In: Proceedings of the International Conference on Automated Planning and Scheduling, vol. 29, pp. 368–376. https://doi.org/10.1609/icaps.v29i1.3500.

Saint-Guillain, M., Brion, M., Pauly, E. et al., 2022a. Robust advanced modelling and scheduling system (rams): From space exploration to real-world biomanufacturing. https://doi.org/10.21203/rs.3.rs-2093105/v1.

Saint-Guillain, M., Gibaszek, J., Vaquero, T., et al., 2022b. Romie: A domain-independent tool for computer-aided robust operations management. Eng. Appl. Artif. Intell. 111, 104801. https://doi.org/10.1016/j.engappai.2022.104801, URL: https://www.sciencedirect.com/science/article/pii/S0952197622000756.

Saint-Guillain, M., Vaquero, T., Chien, S., et al., 2021a. Probabilistic temporal networks with ordinary distributions: Theory, robustness and expected utility. J. Artif. Intell. Res. 71, 1091–1136. https://doi.org/10.1613/jair.1.13019.

Saint-Guillain, M., Vaquero, T.S., Chien, S.A., 2021b. Lila: Optimal dispatching in probabilistic temporal networks using monte carlo tree search. In: Proc. of Fifth ICAPS Workshop on Integrated Planning, Acting, and Execution IntEx '21. Pasadena, CA, USA: Jet Propulsion Laboratory, National Aeronautics and Space Administration. URL: http://hdl.handle.net/2014/55076.

Schrepp, M., Hinderks, A., Thomaschewski, J., 2017. Construction of a benchmark for the user experience questionnaire (UEQ). Int. J. Interact. Multim. Artif. Intell. 4 (4), 40–44. https://doi.org/10.9781/ijimai.2017.445.

Schrepp, M., Thomaschewski, J., 2019. Design and validation of a framework for the creation of user experience questionnaires. Int. J. Interact. Multim. Artif. Intell. 5 (7), 88–95. https://doi.org/10.9781/ijimai.2019.06.006.

Shelat, S., Karasinski, J.A., Flynn-Evans, E.E., et al., 2022. Evaluation of user experience of self-scheduling software for astronauts: Defining a satisfaction baseline. In: Harris, D., Li, W.-C. (Eds.), Engineering Psychology and Cognitive Ergonomics. Springer International Publishing, Cham, pp. 433–445. https://doi.org/10.1007/978-3-031-06086-1_34.

Steel, R., Niézette, M., Cesta, A., et al., 2009. Advanced Planning and Scheduling Initiative- MrSpock Aims for Xmas. Language 1050, 3, URL: https://ui.adsabs.harvard.edu/abs/2009ESASP.673E...8S/abstract.

Thiel, C.S., Pletser, V., Foing, B., 2011. Human crew-related aspects for astrobiology research. Int. J. Astrobiol. 10 (3), 255–267. https://doi.org/10.1017/S1473550411000152.

Vermeeren, A.P.O.S., Law, E.L.-C., Roto, V. et al., 2010. User experience evaluation methods: Current state and development needs. In: Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries NordiCHI '10, Association for

Computing Machinery, New York, NY, USA, p. 521–530, https://doi.org/10.1145/1868914.1868973.

Wang, D., Russino, J.A., Basich, C. et al., 2022. Analyzing the efficacy of flexible execution, replanning, and plan optimization for a planetary lander. In: Kumar, A., Thiébaux, S., Varakantham, P., Yeoh, W. (Eds.), Proc. of International Conference on Automated Planning and Scheduling. vol. 32 of ICAPS '22, AAAI Press, Palo Alto, CA, USA, https://doi.org/10.1609/icaps.v32i1.19838.

Yelamanchili, A., Agrawal, J., Chien, S. et al., 2020. Ground-based automated scheduling for the mars 2020 rover. In: Proc. of International Symposium on Artificial Intelligence, Robotics, and Automation for Space i-SAIRAS 2020. URL: http://hdl.handle.net/2014/53236.