The Journal of the Astronautical Sciences (2022) 69:918–940
https://doi.org/10.1007/s40295-022-00317-z

ORIGINAL ARTICLE

Maximizing Dust Devil Follow-Up Observations on Mars Using Cubesats and On-Board Scheduling

Robyn Woollands[1,2] · Federico Rossi[1] · Tiago Stegun Vaquero[1] · Marc Sanchez Net[1] · S. Sandra Bae[1,3] · Valentin Bickel[4] · Joshua Vander Hook[1]

Accepted: 4 March 2022 / Published online: 31 May 2022

Abstract

Several million dust devil events occur on Mars every day. These events last, on average, about 30 minutes and range in size from meters to hundreds of meters in diameter. Designing low-cost missions that will improve our knowledge of dust devil formation and evolution, and their connection to atmospheric dynamics and the dust cycle, is fundamental to informing future crewed Mars lander missions about surface conditions. In this paper we present a mission for a constellation of low orbiting Mars cubesats, each carrying imagers with agile pointing capabilities. The goal is to maximize the number of dust devil follow-up observations through real-time, onboard scheduling. We study scenarios where cubesats are equipped with a 2.5 degree boresight angle camera that accommodates twenty-one slew positions (including nadir). We assume a concept of operations where the cubesats autonomously survey the surface of Mars and can autonomously detect dust devils from their surface imagery. When a dust devil is detected, the constellation is autonomously re-tasked through an onboard distributed scheduler to capture as many follow-on images of the event as possible, so as to study its evolution. The cubesat orbits are propagated assuming two-body dynamics and the ground tracks and camera field of view are computed assuming a spherical Mars. Realistic inter-agent communication link opportunities are computed and included in our optimization, which allow for real-time event detection information to be shared within the constellation. We compare against a powerful "omniscient" oracle which has a priori knowledge of all dust devil activity to show the gap between predicted performance and the best possible outcome. In particular, we show that the communications are especially important for acquiring follow-up observations, and that a realistic distributed scheduling mechanism is able to capture a large fraction of all dust devil observations that are possible for a given orbit configuration, significantly outperforming a nadir-pointing heuristic.

✉ Robyn Woollands

Extended author information available on the last page of the article

1 Introduction

Martian dust storms have had a major impact on Mars exploration efforts. Planet-wide storms can bring exploration operations to a halt by reducing available power or obscuring view. This has serious implications for both the explore-ability and habitability of Mars. An understanding of the Martian dust cycle, with its connections between regional and global storms, thermal exchange between the surface and atmosphere, and the circulation that can eventually lead to atmospheric escape, remains a top scientific priority [1].

As dynamic, widespread phenomena, these processes are not only challenging to capture within their global context from the lower orbits that have dominated Mars orbiter missions, but also nearly impossible to capture in the required detail if higher orbits are to be assumed (e.g., Areostationary). Our goal in this paper is to find a middle ground using a constellation of small orbiters collaborating in low Mars orbit. We present initial feasibility studies of a constellation of small orbiters using medium-resolution cameras to track dust devils on the surface of Mars in real-time.

By using a combination of medium-resolution imagers, onboard processing, and cross-platform cueing, the constellation will enable short and long time-scale studies of dust devil formation, propagation, and global distribution over seasonal weather variations. We take, as inspiration, the recent NASA study on a multi-satellite network (MOSAIC) [1] for Mars.

We proceed as follows. We make use of a five satellite network where we assume that the five satellites are sufficiently small enough to be delivered as a secondary payload, perhaps on an ESPA ring accompanying a main mission[2]. The satellites are assumed to have basic telemetry and commanding interfaces direct to Earth, but otherwise relay their data through the main mission platform. We do not model the main satellite or nearby telecommunications relays, and simply assume there is "sufficient" capacity for this role. Note that MOSAIC [1] includes such relays for their mission concept, and significant literature points towards to the need and high likelihood of such relays being deployed after the retirement of the Mars Reconnaissance Orbiter[3, 4].

However, we do assume that the five satellites are equipped with telecom payloads to communicate among members of the network. The main use of this limited communication ability is to exchange information about the surface conditions between each observer and "cue" other members. There is a precedent for this concept of operation. For instance, it resembles the "A-train" configuration near Earth, but includes significantly more autonomy to cope with the long propagation delays between Earth and Mars [5]. Similarly, it also closely resembles the "sensor network" configuration, in that cues are sent among members of the network to schedule follow-up observations [6]. The total data volume exchanged over the network will be kept small so that communications is not the limiting factor of the design.

The members of the constellation are assumed to have medium-resolution framing imagers. The ground sample distance is assumed to be similar to the

Mars Context Camera [7] (CTX). This setup (approximately 6 meters / pixel) allows the largest dust devils to be resolved [8]. For example, Fig. 1 shows a famous, large dust devil at CTX resolution.

Global coverage is achieved by “being vigilant” that is, taking significantly more imagery than what is usual. Details are preserved since the images are taken in low Mars orbit and with higher temporal frequency than usual. However, this introduces a data bottleneck: the constellation is gathering significantly more imagery than can be downlinked. As such, significant onboard processing is required to prepare downlink reports of surface activity, and onboard autonomy is required to schedule follow-up observations of the most promising dust devil candidates, as discussed in [9].

One way of automating the detection of dust devils is by using convolutional neural networks (CNNs). Testing of a basic, ResNet 50-based CNN trained with a few hundred labels shows that one can achieve Average Precision (AP) values of around 0.7 in CTX data (a detailed description is out of scope of this paper but forthcoming). The AP could be further optimized by applying a stricter score threshold value (optimize precision), which would reduce the recall, however. Initial testing with CTX data indicates that a CNN-based detector is fairly robust against varying geomorphic backgrounds (dunes, craters, etc.), image quality, and illumination conditions, while being able to detect small, large, faint, and distinct dust devils alike (see Fig. 2). Experience has shown that dust devils as small as ~60 meters across can be detected and mapped in CTX images. In this work, we assume that a well trained classifier is available for use on-board the cubesats.

With onboard processing and modest cross-platform communication, we now present our core problem statement: how to position the constellation and schedule the observations to achieve maximum science return? The remainder of the paper focuses on designing an autonomous onboard scheduling system that trades

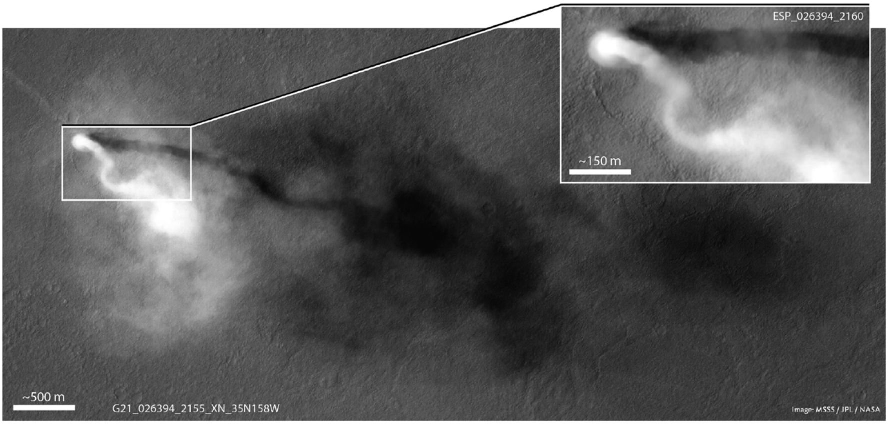


Fig. 1 A CTX image and HiRISE zoom-in (inset) of a large dust devil in Amazonis Planitia. A framing version of CTX, likely well within today’s photo-imaging capabilities, may be able to record near-real-time imagery and track dust devil size, distribution, and shape over time when combined with onboard processing. Credit: NASA JPL, University of Arizona

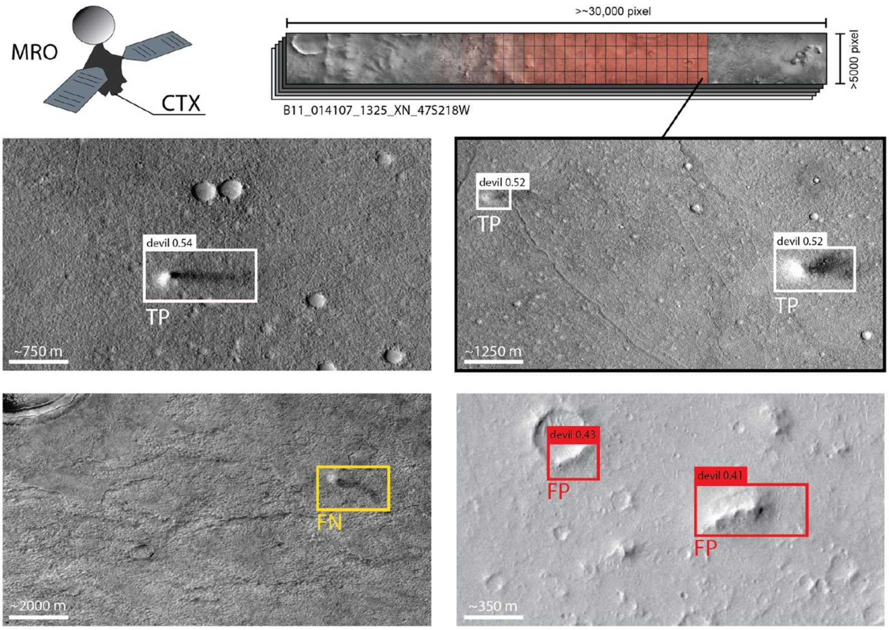


Fig. 2 Performance illustration of an exemplary, CNN-based dust devil detector, operating on rectangular tiles of various sizes cropped from full-frame CTX images. **A**) & **B**) correct detections of small and large dust devils (white, TP = true positive); **C**) a dust devil missed by the detector (yellow, FN = false negative); **D**) false detections (red, FP = false positive). The network confidence score is displayed with its respective detection. The smallest (and faint) dust devil in this example is ~100 meters across

exploration (i.e., look in areas that have not been photographed recently) versus *exploitation* (i.e., take another image of a known dust devil).

2 State of the Art

Automated tasking, as part of a ground-orbiting system, has a long history in space missions[10, 11], including successful demonstrations for earth observing spacecraft [12]. A more detailed review of autonomous agents for space exploration can be found in [13] and [14]. The state of the art is timeline-based, meaning exact subsystem sequences are derived for a single agent using constraints and resource-based cost functions [15] in a framework known as ASPEN. ASPEN is based on its previous generation, Continuous Activity Scheduling Planning Execution and Replanning (CASPER), and along with Eagle Eye, a specialized version of the software, the ASPEN scheduler generates a baseline mission operations plan from observation requests. The scheduler "greedily" schedules commands in a priority-first manner. A number of heuristics have been considered for scheduling, see e.g. [16–19], and [20] (for communication scheduling). In addition to generating an initial plan, ASPEN/Eagle Eye, steps through a *plan execution* phase, an *on board image analysis* phase, an *onboard replanning* phase, and a *target reimaging* phase.

When rescheduling, the algorithm again follows a greedy approach and searches the current plan within the desired time window and replaces the earliest available observation that can be replaced with an updated observation request. Recent work (demonstrations of which are two decades in the making) has moved toward expressing mission goals and decomposing them into activities onboard the spacecraft [21].

An alternative approach to agile satellite tasking relies on mixed-integer programming; this approach has been proposed both for single satellites [22] and for satellite constellations [23, 24], including deployment on Planet's Earth-observing constellation (albeit with ground-based planning) [24].

There are relatively few precedents for multi-agent variants of these planning and scheduling technologies, and even fewer examples of demonstrations, beyond the work in [24]. See [25] for an example of multi-rover autonomous operations and [26] for multi-aircraft / multi-spacecraft networks for earth observations, and [27] for recent work on constellation communication networks and tasking. A particularly close example is work by Nag et al. [28], which discusses the science benefit of a constellation for multi-angle simultaneous observations of the Earth atmosphere. In that work, Nag et al. optimize the constellation's configuration to maximize science gain. However, onboard autonomy was not considered. In [29] an optimization framework is presented that can schedule multi-satellite constellations. However, the framework in [29] (like all related work mentioned) is optimized for quick convergence; in contrast, the work in this paper produces *provably optimal* schedules. We focus on analyzing the benefit of communications and onboard planning. We take as input, models of dust devil activity and a simplified model of the spacecraft dynamics and controls to allow a comprehensive, long-term study of the efficacy of the network.

It is worth noting that automated detection of dust devils on Mars is not a new concept. The work from Castano et al. [30] describes a dust-devil detection system that has been proposed in [30] for the Opportunity rover on Mars. The technique for autonomous targeting relevant surface science phenomena (including dust devils) is presented in [31]. In this work we focus on the multi-asset aspect of dust-devil detection and more specifically on the planning and coordination of observations.

Given the many decade history of automated planning and scheduling for space operations, our study takes as granted a framework for turning high-level commands into low-level control sequences, such as ASPEN [32], and the network design considerations discussed in [27].

3 Problem Statement

We formulate the constellation follow-up observation scheduling as follows.

We assume that dust devils occur across the Martian surface according to a stochastic process with known spatial distribution. Once a dust devil forms, its temporal duration and spatial evolution are also stochastic, with known distribution.

A constellation of spacecraft observes the Martian surface from orbit with cameras as described in the Sect. 1. Each spacecraft is equipped with a dust devil detector (with imperfect precision and recall) that can detect dust devils present in images and infer their location on the surface. The orbits are fixed; however, the spacecraft can decide where to point their cameras, subject to slewing constraints that enforce a maximum slewing rate.

Spacecraft can share their observations (specifically, the location of detected dust devils) through communication links when they are in line-of-sight contact of each other.

The goal of the problem is to maximize the number of follow-on observations of dust devils; specifically, we assume that it is more valuable to collect multiple observations of the same dust devil than to gather single observations of distinct dust devils, up to a maximum number of observations per event.

We formalize the scheduling problem as follows.

We define:

- $\mathcal{R}$, a set of regions of interest $\mathcal{R} = \{r_1, ..., r_R\}$ on the surface of Mars where dust devils may occur; spacecraft observe these regions to look for previously-undetected dust devils. Each region should be imaged at most once over a given time window.
- $\mathcal{E}$, a set of observed transient events $\mathcal{E} = e_1, \dots, e_E$ on the surface of Mars, modeling dust devils that have been detected and for which follow-up observations are desirable. Each event is associated with a given time window, which captures the expected duration of the dust devil. Within that window, the event should be observed as many times as possible.
- $H = (H_{start}, H_{end})$, a planning time horizon with start and end time points.
- $\mathcal{S}$, a set of spacecraft agents $\mathcal{S} = \{a_1, ..., a_N\}$ orbiting Mars.
- $\mathcal{A}$, a set of attitudes that each spacecraft $a \in \mathcal{S}$ can assume at every time step.
- $\mathcal{O}$, a set of observation opportunities $\{o_{i,a,t}\}$ for agent $i \in \mathcal{S}$ at time t in attitude a within the time (H_{start}, H_{end}). For each observation opportunity, an observation opportunity function $(o_{i,a,t}) \mapsto (\mathcal{O} \cup \mathcal{E} \cup \emptyset)$ reports what regions and what transient events can be observed by agent $a \in \mathcal{S}$ in attitude $a \in \mathcal{A}$ at time $t \in (H_{start}, H_{end})$.
- $\mathcal{U}(r, o) : \mathcal{O} \cup \mathcal{E}, \mathcal{N} \mapsto \mathbb{R}$, a scoring function that maps regions of interest $r \in \mathcal{R}$ and transient events $e \in \mathcal{E}$, and the number o of previous observations of the region or event of interest, to a score value. For regions $\mathcal{R}$, the priority function encodes the expected number of new dust devils that will be observed there, multiplied by the value of a new observation; for events $\mathcal{E}$, the scoring function denotes the scientific interest of collecting the o-th follow-on observation of the event.

Our objective is to produce an observation schedule by selecting $\hat{O} \subseteq \mathcal{O}$ to maximize $\sum_{r\in\mathcal{R}} \mathcal{U}(r)$ subject to agents' attitude constraints.

We are now in a position to formalize the constellation follow-up observation scheduling problem.

Problem 1 Find the set of observations $\hat{O} \subseteq O$ that maximize $\sum_{r\in R} U(r, o)$ subject to instrument constraints.

4 Autonomous Pointing for Science Planning

4.1 A Centralized ILP Algorithm

We start by presenting a centralized integer programming formulation to solve Problem 1.

We discretize the time horizon of the problem in equally-spaced intervals $\mathcal{T}= [t_1, \dots , t_T]$. We also discretize the set of attitudes that each spacecraft can point to into set $\mathcal{A} = [k_1, \dots , k_K]$ of discrete attitudes.

Pointing function We assume that a pointing function $p : (\mathcal{S}, \mathcal{A}, \mathcal{T}) \mapsto \{\mathcal{R} \cup \mathcal{E} \cup \emptyset\}$ is known. The set $p(i, a, t)$ represents the regions $r \in \mathcal{R}$ and the known, ongoing transient events $e \in \mathcal{E}$ that can be observed by spacecraft $i \in \mathcal{S}$ in attitude $a \in \mathcal{A}$ at time $t \in \mathcal{T}$.

Slewing constraints We also assume that the spacecraft cannot transition between arbitrary pairs of attitudes in one time step due to slewing constraints. For a given attitude $k \in \mathcal{A}$, we denote the set of feasible prior attitudes (i.e., the set of attitudes from which it is possible to transition to P(k) in one time step) as P(k).

Variables We define the following variables:

- $X(s, a, t), s \in \mathcal{S}, a \in \mathcal{A}, t \in \mathcal{T}$, a set of Boolean variables. $X(s, a, t)$ is 1 iff spacecraft s assumes attitude a at time t.
- $Y(s, a, t), s \in \mathcal{S}, a \in \mathcal{A}, t \in \mathcal{T}$, a set of Boolean variables. $X(s, a, t)$ is 1 iff spacecraft s captures data in attitude a at time t.
- $O(e, o), e \in \mathcal{R} \cup \mathcal{E}, o \in [1, \dots , \overline{O}(e)]$, where $\overline{O}(e)$ is the maximum number of observations of region or event e. A set of Boolean variables. $O(e, o)$ is 1 iff region or event e is observed at least o times.

Problem Formulation We are now in a position to formalize Problem 1 as a ILP.

$$\max_{X,Y,O} \sum_{e\in\mathcal{R}\cup\mathcal{E}} \sum_{o\in[1,\dots,\overline{O}(e)]} U(e,o)O(e,o) \tag{1a}$$

subject to

$$O(e,o) \le O(e,o-1) \qquad \forall e \in \mathcal{R} \cup \mathcal{E}, o \in [2, \dots , \overline{O}(e)] \tag{1b}$$

$$\sum_{o=1}^{\overline{O}(e)} O(e,o) = \sum_{t\in\mathcal{T}} \sum_{s\in\mathcal{S}} \sum_{a\in\mathcal{A}} Y(s,a,t) \cdot \mathbf{1}_{e\in p(s,a,t)} \qquad \forall e \in (\mathcal{R} \cup \mathcal{E}) \tag{1c}$$

$$Y(i,a,t) \le X(i,a,t) \qquad \forall i \in \mathcal{S}, a \in \mathcal{A}, t \in \mathcal{T} \tag{1d}$$

$$\sum_{a\in\mathcal{A}} X(i,a,t) = 1 \qquad \forall i \in \mathcal{S}, t \in \mathcal{T} \tag{1e}$$

$$X(i,a,t+1) \leq \sum_{l \in P(a)} X(i,l,t) \qquad \forall i \in \mathcal{S}, \forall t \in \mathcal{T} \setminus t_T, \forall a \in \mathcal{A} \tag{1f}$$

Equation 1a captures the reward for observing a given event or region for the o-th time. Equation 1b ensures that the o-th observation of a region or event if only performed after the $(o-1)$-th observation. Equation 1c enforces that the overall number of observations of a region matches the number of times a spacecraft has captured an image pointing at the said region. Equation 1d guarantees that a spacecraft only captures an observation in a given attitude if it is indeed pointed in that attitude. Equation 1e ensures that each spacecraft assumes only one attitude at every time step. Finally, Eq. 1f ensures that the sequence of scheduled observations is compatible with the spacecrafts' attitude slewing constraints.

4.2 Distributed Implementation

Solving Eqs. 1a-1f requires knowledge of *all* observed events, and the solution prescribes a set of attitudes for *all* spacecraft—therefore, the problem cannot directly be used to control a constellation where spacecraft are not in constant communication with each other and must make independent decisions based on their own observations. To overcome this, we propose a *shared-world* implementation of the ILP, shown in Fig. 3. Each spacecraft independently detects dust devils in its own observations and adds its detections to a local copy of a global event map. When two spacecraft are in contact, they reconcile their copies of the global event map by sharing their observations; if a conflict arises (e.g., if two spacecraft have observed the same region but only one has detected a dust devil), we optimistically assume that the detection was correct. In the following section, we discuss our selection of a Sun synchronous orbit in order to ensure that the lighting conditions (shadow angles and shadow lengths) are the same for every pass, thus reducing false positive detections. Each spacecraft individually solves Eqs. 1a-1f based on its local copy of the global

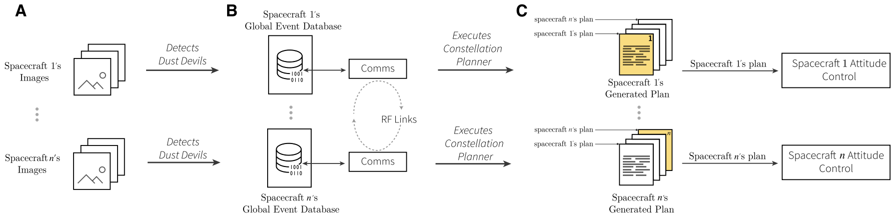

A
Spacecraft 1's Images
Detects Dust Devils
Spacecraft n's Images
Detects Dust Devils
B
Spacecraft 1's Global Event Database
1001 0110
Comms
RF Links
Comms
1001 0110
Spacecraft n's Global Event Database
Executes Constellation Planner
Executes Constellation Planner
C
spacecraft n's plan
spacecraft 1's plan
1
Spacecraft 1's Generated Plan
Spacecraft 1's plan
Spacecraft 1 Attitude Control
spacecraft n's plan
spacecraft 1's plan
n
Spacecraft n's Generated Plan
Spacecraft n's plan
Spacecraft n Attitude Control

Fig. 3 Architecture of the proposed constellation scheduler. (**A**) Each spacecraft captures images of the Martian surface. A dust devil detector searches for dust devils in captured imagery. (**B**) When a dust devil is detected, its location is added to a map database which tracks possible active dust devils that should be targeted for revisit. Each spacecraft executes an instance of the constellation planner (described in the Sect. 4), which uses the locations of the detected dust devils to prescribe pointing locations for every spacecraft. (**C**) The spacecraft then executes the pointing and image capture commands assigned to itself by the constellation planner, and discards the assignments for other spacecraft. The map database of detected dust devils is opportunistically synchronized across spacecraft when a communication link is available

event map, obtaining a set of prescribed attitudes for the entire constellation. Each spacecraft then executes the pointing commands prescribed to itself by the ILP, and disregards the pointing commands for other spacecraft.

With this approach, if the spacecrafts' global events map are perfectly synchronized, all spacecraft act in accordance with the same solution to Eqs. 1a-1f. If the global events maps are not synchronized (because, e.g., information about a detected event has not yet been propagated to all spacecraft), each spacecraft is guaranteed to have access to a solution, even though the spacecraft may act in an inconsistent manner (e.g., a region may be imaged twice, or it may not be imaged because each spacecraft thinks that another one will capture it). In the Numerical Results, we show that, for the orbits and network topologies considered in this paper, information propagates readily through the constellation, and the impact of imperfect synchronization between the spacecrafts' global event maps is negligible.

4.3 Receding-horizon Approach

Obtaining a solution to Problem 1 requires knowledge of observed dust devils: therefore, the problem should be re-solved whenever a new dust devil is observed. To address this, we implement the algorithm in a receding-horizon fashion. Each spacecraft solves Eqs. 1a-1f within a fixed time horizon – in the simulation results, we use a fifteen-minute time horizon. The selection of the time horizon represents a trade-off between computational performance and optimality of the resulting schedule; specifically, the fifteen-minute time horizon is sufficient to plan slewing motions from any pointing attitude to any other pointing attitude, and allows spacecraft to plan revisits of dust devils detected by multiple spacecraft ahead of them in a train; at the same time, the optimization problem can consistently be solved in under 100 ms, as discussed in Sect. 5.5. As soon as new information is received (either through a direct observation, or through an update to the global event map provided by another spacecraft), the spacecraft re-solves the problem with the same horizon (that is, for the fifteen minutes following the update). Also, before the horizon is reached, the problem is solved again, irrespective of whether new information is available. The receding horizon approach ensures that the computation cost is highly contained (as shown in the Sect. 5), while incorporating new information in the solution as soon as it becomes available.

5 Numerical Simulation & Results

5.1 Dust Devil Model

Dust devils form around low-pressure air pockets, where the air is drawn into a narrow rising column through the surrounding cooler air. A study that combined data from the High Resolution Stereo Camera (HRSC) and the Mars Orbiter Camera (MOC) found that some large dust devils on Mars have a diameter of 700 m and last for at least 26 minutes [8]. The largest dust devils can reach a height of 8 km. A recent study found that on average, during any given day, one 13 m wide dust devil appears in every square kilometer on Mars [33].

As a dust devil moves across the surface of Mars, it lifts the top layer of dust into the atmosphere and exposes the dark underlying surface. These dark tracks can last a few weeks, at which point they are either covered up as a result of wind action or they become oxidized from exposure to sunlight and the Martian atmosphere, and thus return to same red color as the rest of the planet. Dust devil tracks can be more than 30 m wide and extend for more than 4 km.

In our simulations, we use a simple *random model* for dust devil generation. We assume that no dust devils exist at latitudes greater than 70° and no dust devils exist before 11 am or after 4 pm local solar time. We center the active dust devil region at 1:30 pm local solar time. Over this region, we simulate dust devils with an average density of one per ten square kilometers. Though this is a more conservative estimate than those found in the literature [33], it is adequate to test our onboard scheduler. In our simulations, the mean dust devil lifetime is 26 minutes and start and end times are computed for each dust devil. Furthermore, we have not taken into account the motion of the dust devils across the surface and their associated ground track. This will be left as future work, along with a more sophisticated dust devil model.

5.2 Orbits

In this paper we propagate two-body orbits for a train of five cubesats in a low Mars (200 km altitude), Sun synchronous orbit. Two different train constellations are considered. The first has satellites in the train separated by one minute time intervals, and the second uses six minute intervals. We have selected a low Mars orbit such that the cubesat camera (for example, HiREV [34]) is capable of imaging the surface with a resolution of less than 10 m. A Sun synchronous orbit is chosen to ensure that the lighting conditions (shadow angles and shadow lengths) are the same for every pass. Although we do not simulate dust devil event detection through image processing in this paper, we assume that using the same lighting conditions for each pass will aid the eventual dust devil event detection algorithm—thus reducing the potential for false positives. Each cubesat camera has a field of view with 2.5° boresight angle [34]. For these simulations, we have selected twenty-one camera positions (Fig. 4), spaced by approximately two to three degrees apart in the along-track direction $(0, \pm 3, \pm 5,$ and $\pm 7.5°)$ and five degrees apart in the across-track direction $(0, \pm 5°)$. We assume that the slew/settle time to move between adjacent camera positions is one minute [35].

Figure 5 shows a snapshot taken during the simulation where the cubesats in the train are separated by one minute intervals. The top panel shows a spherical Mars that is experiencing a southern hemisphere summer. Each yellow dot on the surface represents an active dust devil. The low-Mars orbit is also shown in cyan/red, along with the ground track. Note that the five satellites are colored red, blue, green, black and cyan respectively. The magenta circles represent currently detected dust devils. The bottom panel shows a 2D projection of Mars, along with the same information plotted as in the top panel.

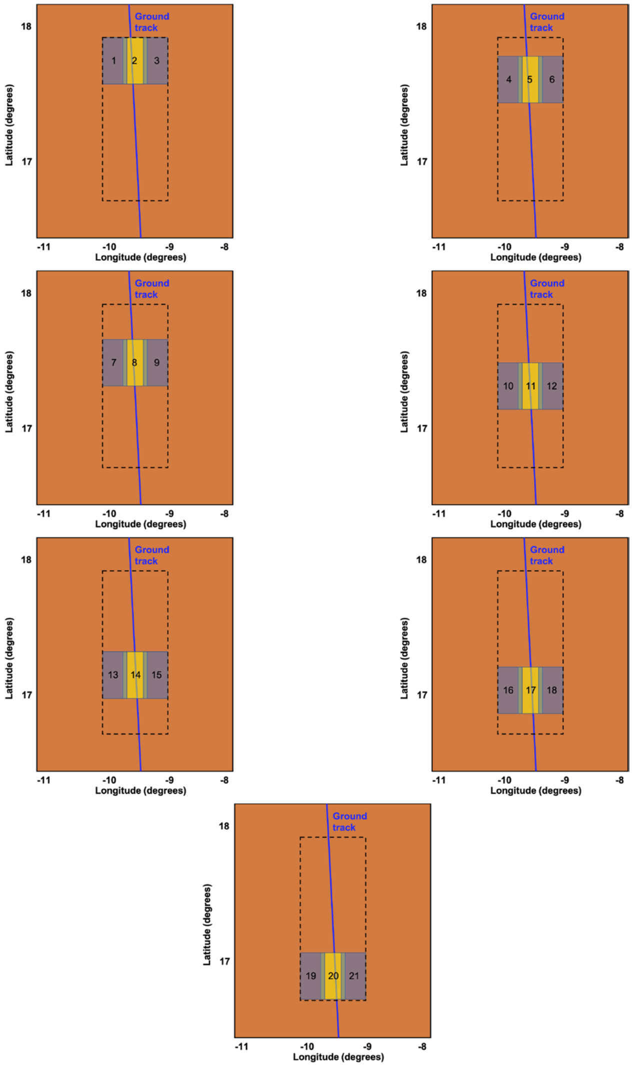


Fig. 4 Schematic showing the 21 possible field of view regions that the camera could point to in order to observe dust devils. Position 11 is nadir pointing

Figure 6 shows an enlarged view for six snapshots taken at one minute time intervals. To prevent clutter in these figures, the ground coverage bounding box is shown for just 5 of the 21 possible camera positions. Of course, in reality, the camera can only image one of these regions at a time. Looking at the top-left panel, if the lead satellite (red) was pointing 5° ahead of nadir (i.e. *camera position 5*), it would detect one dust devil event (yellow dot with magenta circle), and if it was pointing 5° behind nadir (i.e. *camera position 17*) it would detect two dust devil events. The next snapshot (top-right) shows that one minute later the second satellite (blue) is capable

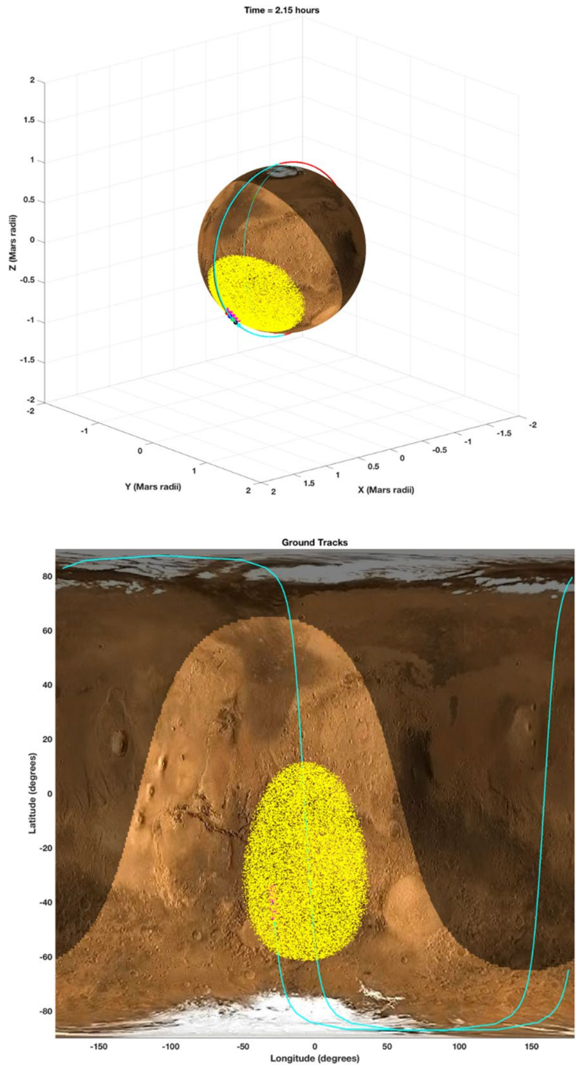


Fig. 5 A snapshot taken during the simulation that shows the five sun synchronous, low-Mars orbiting satellites, flying over a region of dust devil activity (yellow dots) during a southern hemisphere summer. The top panel shows a spherical Mars and the bottom panel shows a two-dimensional projection. The respective orbits and ground tracks are also shown

of imaging the dust devils that were previously detected by the lead satellite one minute earlier. The remaining panels show the satellite train moving further along in its orbit as each minute passes.

5.3 Field of View

The camera field of view is modelled as a square and the corresponding observable surface of Mars is computed as the region enclosed within the four corners of this square projected onto the surface of a spherical Mars. A dust devil is detected if it falls within this visible surface element on Mars. In our simulations, all the orbits are

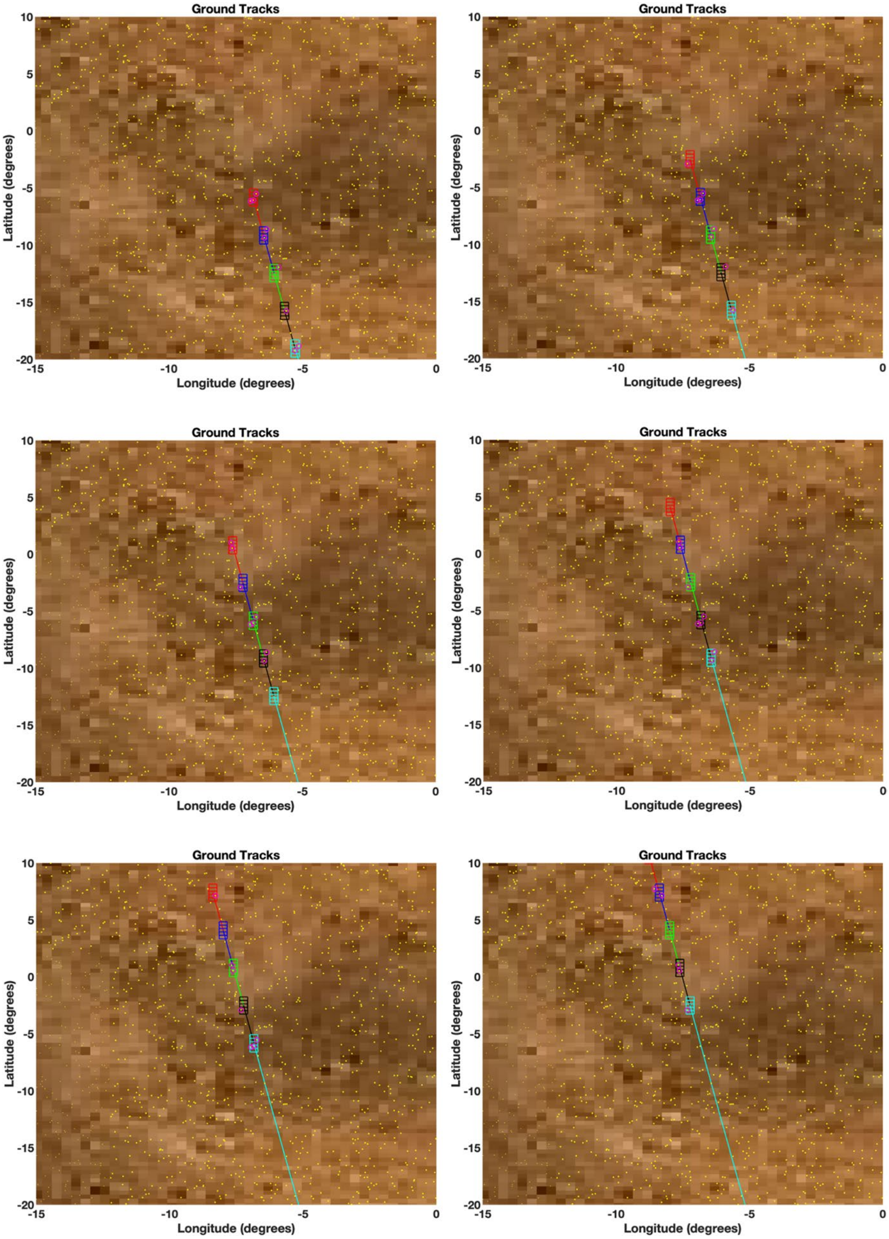


Fig. 6 A series of six snapshots taken at one minute intervals that show the instantaneous ground coverage for each of the five cubesats in the chain. Note that all five camera slew positions are shown simultaneously in this figure. Yellow dots represent active dust devils and magenta circles represent possible detections

sampled at one minute time intervals. Thus, if the lead spacecraft detects an event in, say, *camera position 1*, it can communicate the detection to the second spacecraft, whose onboard scheduler can decide to slew from its current camera position to *camera position 1* in order to re-image the dust devil event one minute later — thus enabling the dust devil time history to be recorded. Similarly, the second spacecraft can communicate to the following one so its scheduler can, in turn, command a slew to the same position, and so on. If the lead spacecraft detects a dust devil event and the second spacecraft also detects an event, then the onboard scheduler, running in real-time, will determine which positions the following spacecraft should slew to in order to maximize the total number of follow-on dust devil observations.

5.4 Communication

In our simulations, we assume that each cubesat is equipped with a patch antenna. For this orbit geometry (200 km altitude), with the one minute separation between satellites, all elements of the chain are constantly within line-of-sight communication, thus enabling information to be transferred between the satellites at any time. In the case where satellites are separated by six minutes, the line-of-sight, and hence the communication link, only extends to the satellite immediately in front and behind of it. We assume that communication between satellites within line-of-sight of each other takes 60 s (one simulation step), a highly conservative estimate for the data volumes considered in this work. We assess the performance of the proposed approach in both scenarios.

5.5 Results

Comparison with upper and lower bounds First, we compare the performance of the proposed constellation scheduling approach with an “oracle” upper bound and a naive lower bound.

The oracle solves Eqs. 1a-1f once with full knowledge of the location of all present and future dust devils. The problem is solved in a centralized manner. The solution provided by the oracle represents an (unattainable) upper bound on the performance of the constellation. The lower bound is obtained by simply setting the attitude of all spacecraft to be in a nadir-pointing direction at all times.

The two scenarios are compared with the outcome of the simulation where each spacecraft solves Eqs. 1a-1f in a receding-horizon fashion, as described in the Autonomous Pointing section, and spacecraft exchange observations along available communication links as discussed in the Communication section. To ensure a fair comparison, all spacecraft are assumed to have access to an ideal event detector with perfect precision and recall (that, is, it never produces false positives or negatives); the effect on performance of a non-ideal event detector is discussed in the next section.

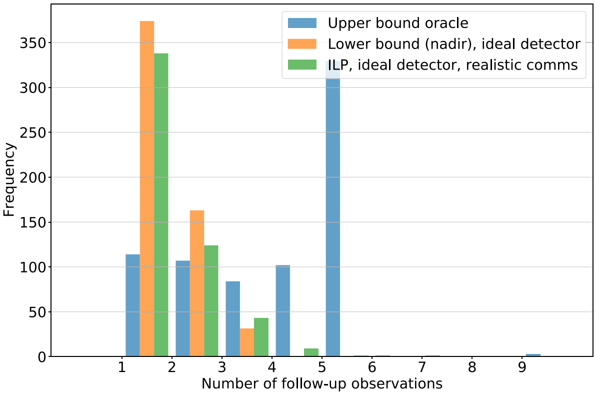


Fig. 7 Closely-spaced orbits: distribution of the number of follow-on observations for the proposed approach, compared to an upper and lower bound. The effect of a non-ideal event detector are also shown

Table 1 Performance of the proposed distributed pointing approach compared to an "oracle" upper bound and a fixed-attitude lower bound

	Events observed				Follow-ups	
	Total	Unique	2+	3+	Mean	Max
Oracle (upper bound)	2674	741	627	520	3	9
Nadir-pointing (lower bound), ideal detector	793	568	194	0	0.40	2
This paper, ideal detector	756	515	177	53	0.46	4
Nadir-pointing (lower bound), non-ideal detector	549	440	99	10	0.25	2
This paper, non-ideal detector	571	437	111	22	0.31	3

Closely spaced orbits (one minute gap between spacecrafts) are considered

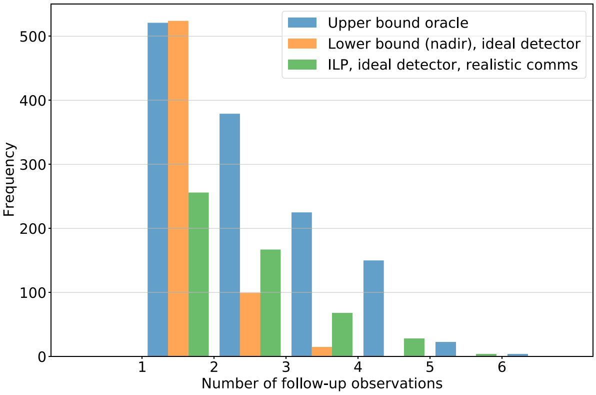


Fig. 8 Widely-spaced orbits: Distribution of the number of follow-on observations for the proposed approach, compared to an upper and lower bound

The results are reported in Table 1 and Fig. 7 for a set of orbits with one minute spacing between spacecraft, and in Table 2 and Fig. 8 for orbits with six minute spacing.

In the widely-spaced case, the proposed approach results in a 230% increase in the number of events that receive at least one follow-up observation compared to the nadir-pointing approach. The average number of follow-up observations per event increases from 0.21 to 0.77, an almost four-fold increase. The overall number of observations is 20% higher compared to the nadir-pointing case. In contrast, the number of unique events detected decreases by 18% as expected, the autonomous pointing approach privileges follow-on observations as opposed to new event detections, in line with the proposed reward function, which privileges revisits over new detections.

The impact of on-board autonomy is smaller, but still significant, in the closely-spaced case: the approach is able to increase the number of events observed three or more times from zero to 53, and the average number of follow-up observations increases by 15%, at the price of a 9.3% reduction in the number of unique events observed.

Effect of inter-spacecraft communication As discussed in the Sect. 4.2, the distributed implementation relies on inter-spacecraft communication to maintain a common picture of observed events; this can result in sub-optimal, and potentially inconsistent, behavior at the constellation level if inter-spacecraft links are sporadic or if communication between spacecrafs requires multiple hops.

To explore the impact of inter-spacecraft communications on system performance, we compare the proposed approach, where spacecraft communicate with their neighbors when a line-of-sight link is available, with two scenarios: an *idealized* scenario where all spacecraft can exchange information instantly, and a *no-communications* scenario where each spacecraft only relies on information it has collected itself. We focus our attention on the widely-spaced scenario, where each spacecraft can only communicate with its immediate neighbors in the constellation.

Results are shown in Table 2 and Fig. 9.

There is no difference in performance between the proposed approach and the upper bound where all spacecraft can communicate with each other, showing that the proposed approach is able to effectively spread information about event detections across the constellation. In contrast, the case where satellites do not communicate with each other results in performance comparable to the nadir-pointing case, with a small *reduction* in the average number of follow-up observations (vs. the

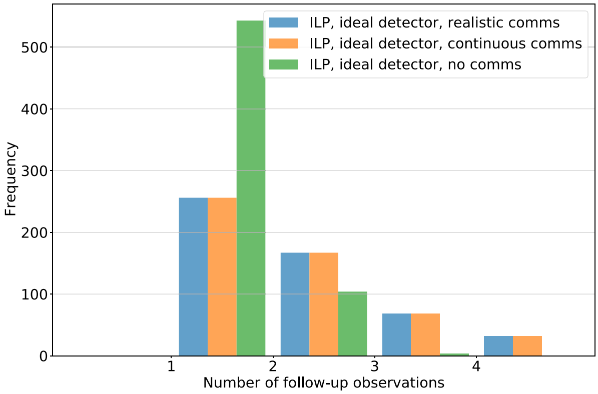


Fig. 9 Widely-spaced orbits: Effect of communications on distribution of the number of follow-on observations

Fig. 10 Widely-spaced orbits: Effect of non-ideal detector on distribution of the number of follow-on observations

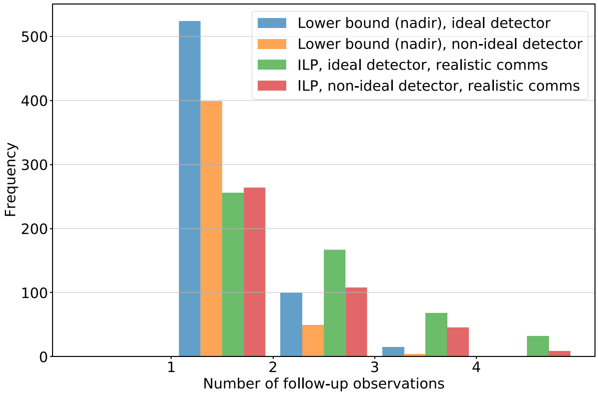


Table 2 Performance of the proposed distributed pointing approach compared to an "oracle" upper bound and a fixed-attitude lower bound

	Events observed				Follow-ups	
	Total	Unique	2+	3+	Mean	Max
Oracle (upper bound)	2695	1302	781	402	1.07	6
Nadir-pointing (lower bound), ideal detector	771	639	115	16	0.21	3
This paper, ideal detector	926	523	267	100	0.77	4
Nadir-pointing (lower bound), non-ideal detector	509	452	53	4	0.13	2
This paper, non-ideal detector	653	426	162	54	0.53	4
This paper, ideal detector, continuous comms	926	523	267	100	077	4
This paper, ideal detector, no comms	760	650	107	3	0.17	2

More widely spaced orbits (six minute gap between spacecrafts) are considered

almost four-fold improvement obtained with communications). Collectively, these results show that the proposed distributed optimization approach is able to approach the performance of a centralized controller, and that constellation-level optimization through inter-satellite communication is critical to achieve this performance.

Non-ideal event detector We also assess the impact of using a non-ideal event detector with non-perfect precision and recall. Specifically, we simulate the performance of a detector with 70% recall, i.e., a 30% false negative rate. Due to limitations of the simulation environment, the study of the effect of non-ideal precision is left as future work. Results are shown in Tables 1 and 2 and in Figs. 7 and 10.

Predictably, the imperfect recall results in a roughly 30% reduction in the overall number of observations collected. Specifically, we observe 15% and 19% reductions in the number of unique detected events in the closely-spaced and widely-spaced cases respectively; the reduction in the average number of follow-up images collected for each event is 31% in both the closely-spaced case and the widely-spaced case. Compared to a nadir-pointing case with imperfect recall, the proposed approach performs similarly to the perfect case, with a two-fold to fourfold increase in the number of events with at least two observations, and a threefold to five-fold increase in the average number of follow-up observations per event, in the widely-spaced case; and only slightly better performance compared to the nadir-pointing case in the closely-spaced case.

Computational performance The ILP was solved with the CPLEX solver. The time required to formulate the problem (i.e., encode the cost function and constraints) and solve it on a commodity desktop workstation is shown in Fig. 11. The time required to formulate the problem remains below 500 ms except for rare outliers, and the time required to solve it is consistently always below 100 ms - results which suggest that the proposed approach could be readily implemented on more modest CPUs well-suited for spaceflight.

6 Visualization System

To visualize and interact with the simulation, we designed a visualization system that adheres to the same mission design methodology as discussed in the previous sections. Past work shows interfaces [36–38] are critical to (i) foster trust in the data and tool from the operator's perspective, and (ii) understand the details of what the scheduler is doing onboard. Interactions provide a dialog between the user and the system [39], and this allows operators to gain confidence in the system as they explores datasets to uncover insights and be part of the decision making process. To allow the operator to explore the onboarding scheduler, the system is composed of three interactive views: overview, coverage map, and time scrubber (Fig. 12). All views are interlinked: the overview and coverage map visualize data corresponding to the particular timestamp from the time scrubber.

Overview The overview (Fig. 12b) provides a quick assessment of the constellation performance in terms of dust devil observations. Dust devils are represented as

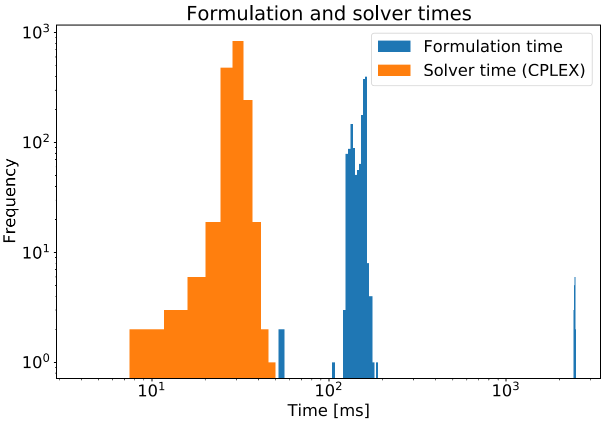
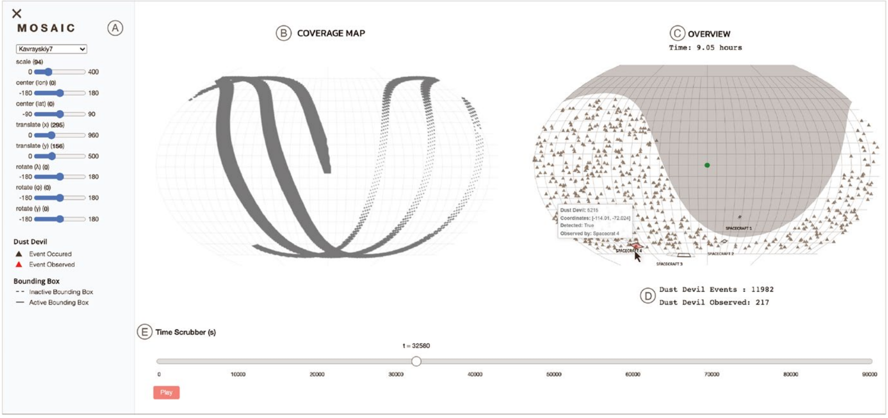


Fig. 11 Computation times to solve the proposed ILP on a Xeon E5-2687W


MOSAIC

Kavrayskiy7

scale (94) 0 400

center (lon) (0) -180 180

center (lat) (0) -90 90

translate (x) (295) 0 960

translate (y) (156) 0 500

rotate (λ) (0) -180 180

rotate (φ) (0) -180 180

rotate (γ) (0) -180 180

Dust Devil: 5215

Coordinates: [-114.01, -72.024]

Detected: True

Observed by: Spacecrat 4

Dust Devil Events : 11982

Dust Devil Observed: 217

t = 32580

Fig. 12 A visualization system (**a**) Side-bar menu with options to change the map projection and orientation (**b**) Coverage map depicts the ground track of the constellation (**c**) Overview provides the state of the constellation and dust devils (**d**) Counters for the total number of dust devil events and dust devil observations (**e**) Time scrubber depicts the current time position within the simulation

brown triangles, which appear and disappear throughout the simulation run. A counter (Fig. 12d) helps count the total number of dust devil events. In contrast, spacecraft are represented by their bounding boxes and spacecraft number. As mentioned in the Sect. 5.3, these bounding boxes are determined from the 21 slew positions each spacecraft is capable of achieving (only five are shown in the visualization in Fig. 12). All bounding boxes are mapped with two visual encodings. First, for each spacecraft, a bounding box has solid edges if it represents the spacecraft's current camera position, while the rest are represented with thin dashed edges. Second, a bounding box is colored red if a spacecraft has observed one or more dust devils in its field of view at a particular time point (e.g., Spacecraft 4 has observed a dust devil at $t = 32580$). Otherwise, the color remains transparent. The visual encoding for the dust devil follows a similar logic, where the dust devil is colored red if it has been observed and remains brown otherwise. To help operators keep track of the simulation, a second counter (Fig. 12d) counts the total number of dust devil events the constellation has observed.

The combination of these visual encodings allows the operator to not only visually see when and where spacecraft have observed dust devils, but also highlights where the vehicles *could* have observed a dust devil event (i.e., a dust devil is within the field of view of one of the bounding boxes that are not in use).

The overview also visualizes the terminator to showcase the daylit side (white illuminated region) and the night side (solid dark grey filled-in region) of Mars throughout the simulation run. This corresponds to the timer (unit in hours) at the top of the overview. The default map projection is Kavrayskiy VII, but operators can change different map projections as well as interactively zoom-in, rotate, and pan using the side-bar menu (Fig. 12a). Operators can also see details on-demand (i.e., via an interactive tooltip).

Coverage Map The coverage map (Fig. 12b) dynamically visualizes the ground track of the constellation throughout the simulation ($t = 0$ to $t = 88320$). The ground track is colored dark steel-blue, and the path corresponds to the movement of the spacecraft in the overview (Fig. 12b). The coverage map provides a visual insight as to which regions the constellation has frequently visited and vice versa. This can help operators compare different constellation formations, and, in particular, different inter-spacecraft spacing.

Time Scrubber The time scrubber (Fig. 12e) is a horizontal slider control that lets the operator dynamically seek any time position within the currently running simulation. The scrubber 'knob' (circle) moves to indicate the current time position. The position of the scrubber knob determines the information being displayed in the overview and coverage map. Operators can also pause and play when analyzing the simulation.

7 Conclusions

Several million dust devil events occur on Mars every day. These events last, on average, about 30 minutes and range in size from meters to hundreds of meters in diameter. Designing low-cost missions that will improve our knowledge of dust devil formation and evolution, and their connection to atmospheric dynamics and the dust cycle, is fundamental to informing future crewed Mars lander missions about surface conditions. In this paper we presented a mission for a constellation of low orbiting Mars cubesats, each carrying imagers with agile pointing capabilities. The goal was to maximize the number of dust devil follow-up observations through real-time, onboard scheduling. Realistic inter-agent communication link opportunities were computed and included in our simulation, which allowed for real-time event detection information to be shared within the constellation. In particular, we found that the communications are especially important for acquiring follow-up observations, and that the proposed distributed scheduling mechanism is able to capture a large fraction of all dust devil observations that are possible for a given orbit configuration, significantly outperforming a nadir-pointing heuristic. This is a significant result that will aid in the study of dust devil science on the surface of Mars, in preparation for future crewed missions.

Acknowledgements The research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration (80NM0018D0004).

Declarations

Conflicts of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Lillis, R.J., Mitchell, D., Montabone, L., Heavens, N.G., Harrison, T., Stuurman, C.M., Guzewich, S.D., England, S., Withers, P., Chaffin, M., et al.: Mars orbiters for surface-atmosphere-ionosphere connections (MOSAIC). In: AGU fall meeting 2020, AGU (2020)
2. Moog Inc.: ESPA: The evolved secondary payload adapter. MOOG Data Sheet (2020)
3. Breidenthal, J.C., Edwards, C.D., Greenberg, E., Kazz, G.J., Noreen, G.K.: End-to-end information system concept for the Mars Telecommunications Orbiter. In: 2006 IEEE Aerospace Conference, IEEE, pp. 13–pp (2006)
4. Breidenthal, J., Xie, H., Lau, C.-W., MacNeal, B.: Space and earth terminal sizing for future mars missions. In: 2018 SpaceOps Conference, p. 2426 (2018)
5. L'Ecuyer, T.S., Jiang, J.H.: Touring the atmosphere aboard the A-Train. Phys. Today **63**(7), 36–41 (2010)
6. Chien, S., Cichy, B., Davies, A., Tran, D., Rabideau, G., Castano, R., Sherwood, R., Mandl, D., Frye, S., Shulman, S., et al.: An autonomous Earth-observing sensorweb. IEEE Intell. Syst. **20**(3), 16–24 (2005)
7. Malin, M.C., Bell, J.F., Cantor, B.A., Caplinger, M.A., Calvin, W.M., Clancy, R.T., Edgett, K.S., Edwards, L., Haberle, R.M., James, P.B., et al.: Context camera investigation on board the Mars Reconnaissance Orbiter. J. Geophys. Res. **112**(E5) (2007). https://doi.org/10.1029/2006JE002808
8. Kahanaa, H., Lemmon, M.T., Reiss, D., Raack, J., Mason, E., Battalio, M.: Martian dust devils observed simultaneously by imaging and by meteorological measurements. In: 49th Lunar and Planetary Science Conference, No. 2083 (2018)
9. Vander Hook, J., Castillo-Rogez, J., Doyle, R., Vaquero, T.S., Hare, T.M., Kirk, R.L., Fox, V., Bekker, D. and Cocoros, A.: Nebulae: A proposed concept of operation for deep space computing clouds. In: Proceedings of the 2020 IEEE Aerospace Conference. Big Sky 1–14 (2020)
10. Chien, S., Jonsson, A., Knight, R.: Automated planning & scheduling for space mission operations. In: Ground System Architectures Workshop (GSAW), Manhattan Beach, California (2005)
11. Chien, S., Rabideau, G., Knight, R., Sherwood, R., Engelhardt, B., Mutz, D., Estlin, T., Smith, B., Fisher, F., Barrett, T., et al.: ASPEN-Automating space mission operations using automated planning and scheduling. In: Space Ops 2000 Toulouse, France (2000)
12. Chien, S., Sherwood, R., Tran, D., Cichy, B., Rabideau, G., Castano, R., Davis, A., Mandl, D., Frye, S., Trout, B., et al.: Using autonomy flight software to improve science return on Earth Observing One. J. Aerospace Comput. Inform. Commun. **2**(4), 196–216 (2005)
13. Gao, Y., Chien, S.; Review on space robotics: Toward top-level science through space exploration. Sci Robot **2**(7) (2017)
14. Amini, R., Chien, S., Fesq, L., Frank, J., Kolcio, K., Mennsesson, B., Seager, S., Street, R.: Enabling and enhancing astrophysical observations with autonomous systems. arXiv:2009.07361 (2020)
15. Fukunaga, A., Rabideau, G., Chien, S., Yan, D.: Aspen: A framework for automated planning and scheduling of spacecraft control and operations. In: Proc. International Symposium on AI, Robotics and Automation in Space (1997)
16. Liu, X., Laporte, G., Chen, Y., He, R.: An adaptive large neighborhood search metaheuristic for agile satellite scheduling with time-dependent transition time. Comput. Operat. Res. **86**, 41–53 (2017). https://doi.org/10.1016/j.cor.2017.04.006

17. Wang, P., Reinelt, G., Gao, P., Tan, Y.: A model, a heuristic and a decision support system to solve the scheduling problem of an earth observing satellite constellation. Comput. Ind. Eng. **61**(2), 322–335 (2011)
18. Lemaître, M., Verfaillie, G., Jouhaud, F., Lachiver, J.-M., Bataille, N.: How to manage the new generation of agile Earth observation satellites. In: Proceedings of the International Symposium on Artificial Intelligence, Robotics and Automation in Space, (2000)
19. Wolfe, W.J., Sorensen, S.E.: Three scheduling algorithms applied to the Earth observing systems domain. Manag. Sci. **46**(1), 148–166 (2000)
20. Marinelli, F., Nocella, S., Rossi, F., Smriglio, S.: A Lagrangian heuristic for satellite range scheduling with resource constraints. Comput. Operat. Res. **38**(11), 1572–1583 (2011)
21. Chien, S., Smith, B., Rabideau, G., Muscettola, N., Rajan, K.: Automated planning and scheduling for goal-based autonomous spacecraft. IEEE Intell. Syst. Appl. **13**(5), 50–55 (1998)
22. She, Y., Li, S., Zhao, Y.: Onboard mission planning for agile satellite using modified mixed-integer linear programming. Aerospace Sci. Technol. **72**, 204–216 (2018). https://doi.org/10.1016/j.ast.2017.11.009
23. Cho, D.-H., Kim, J.-H., Choi, H.-L., Ahn, J.: Optimization-based scheduling method for agile earth-observing satellite constellation. J. Aerospace Inform. Syst. **15**(11), 611–626 (2018). https://doi.org/10.2514/1.I010620
24. Monmousseau, P.: Scheduling of a constellation of satellites: creating a mixed-integer linear model. J. Optim. Theory Appl. 1–28 (2021). https://doi.org/10.1007/s10957-021-01875-2
25. Estlin, T., Gray, A., Mann, T., Rabideau, G., Castano, R., Chien, S.A., Mjolsness, E.: An integrated system for multi-rover scientific exploration. In: AAAI/IAAI, 613–620 (1999)
26. Nag, S.: Sensor webs of agile, small satellite constellations and unmanned aerial vehicles with satellite-to-air communication links. In: 1st IAA Latin American Symposium on Small Satellites (2017). http://sreejanag.com/Documents/IAALA_17_SN.pdf
27. Nag, S., Sanchez Net, M., Li, A., Ravindra, V.: Designing a disruption tolerant network for reactive spacecraft constellations. In: ASCEND 2020, 4009 (2020)
28. Nag, S., Gatebe, C.K., Weck, O.D.: Observing system simulations for small satellite formations estimating bidirectional reflectance. Int. J. Appl. Earth Obs. Geoinform. **43**, 102–118 (2015)
29. Nag, S., Li, A.S., Ravindra, V., Net, M.S., Cheung, K.M., Lam-Mers, R., Bledsoe, B.: Autonomous scheduling of agile spacecraft constellations with delay tolerant networking for reactive imaging. In: Workshop on Scheduling and Planning Applications woRKshop (SPARK), International Conference on Automated Planning and Scheduling (ICAPS) (2019). https://arxiv.org/abs/2010.09940
30. Castano, A., Fukunaga, A., Biesiadecki, J., Neakrase, L., Whelley, P., Greeley, R., Lemmon, M., Castano, R., Chien, S.: Automatic detection of dust devils and clouds on Mars. Machine Vision Appl. **19**(5–6), 467–482 (2008). https://doi.org/10.1007/s00138-007-0081-3
31. Estlin, T.A., Bornstein, B.J., Gaines, D.M., Anderson, R.C., Thompson, D.R., Burl, M., Castaño, R., Judd, M.: AEGIS automated science targeting for the MER Opportunity rover. ACM Trans. Intell. Syst. Technol. **3**(3), 1–19 (2012)
32. Chien, S., Knight, R., Stechert, A., Sherwood, R., Rabideau, G.: Integrated planning and execution for autonomous spacecraft. In: 1999 IEEE Aerospace Conference. Proceedings (Cat. No. 99TH8403), Vol. 1, IEEE, pp. 263–271 (1999). https://doi.org/10.1109/AERO.1999.794242
33. Jackson, B., Lorenz, R., Davis, K.: A frame work for relating the structures and recovery statistics in pressure time-series surveys for dust devils. Icarus **299**, 166–174 (2018)
34. Cho, D.-H., Choi, W.-S., Kim, M.-K., Kim, J.-H., Sim, E., Kim, H.-D.: High-resolution image video CubeSat (HiREV): development of space telescope test platform using a low-cost CubeSat platform. Int. J. Aerosp. Eng. **2019**, 8916416 (2019). https://doi.org/10.1155/2019/8916416
35. Tibbs, M.L.: Design and test of an attitude determination and control system for a 6U CubeSat using AFIT's CubeSat testbed. MSc thesis. Department of the Air Force Air University, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio (2015)
36. Bae, S.S., Rossi, F., Hook, J.V., Davidoff, S., Ma, K.-L.: A visual analytics approach to debugging cooperative, autonomous multi-robot systems' worldviews. In: 2020 IEEE Conference on visual analytics science and technology (VAST), pp. 24–35 (2020) https://doi.org/10.1109/VAST50239.2020.00008
37. Sacha, D., Senaratne, H., Kwon, B.C., Ellis, G., Keim, D.A.: The role of uncertainty, awareness, and trust in visual analytics. IEEE Trans. Visual. Comput. Graph. **22**(1), 240–249 (2016). https://doi.org/10.1109/TVCG.2015.2467591

38. Alper Ramaswamy, B.B., Agrawal, J., Chi, W., Kim Castet, S.Y., Davidoff, S., Chien, S.: Supporting automation in spacecraft activity planning with simulation and visualization. In: AIAA Scitech 2019 Forum, 2348 (2019)

39. Yi, J.S., Kang, Y.A., Stasko, J., Jacko, J.A.: Toward a deeper understanding of the role of interaction in information visualization. IEEE Trans. Visual. Comput. Graph. **13**(6), 1224–1231 (2007)

Authors and Affiliations

Robyn Woollands[1,2] · Federico Rossi[1] · Tiago Stegun Vaquero[1] · Marc Sanchez Net[1] · S. Sandra Bae[1,3] · Valentin Bickel[4] · Joshua Vander Hook[1]

[1] Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109, USA

[2] Department of Aerospace Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA

[3] University of Colorado Boulder, Boulder, CO 80309, USA

[4] Max Planck Institute for Solar System Research, Göttingen, Germany