Jakob Karg, Julian Zobel, Bruno Sousa, Steffen Bamfaste, Giulio Pinzan, Artur Scholz. *Cluster-II: A Recommendation System for Semi-Automated Scheduling of Ground Station Passes.*
In Proc. 16th International Conference on Space Operations (SpaceOps), 2021.

16th International Conference on Space Operations, Cape Town, South Africa – 3.-5. May 2021.

SpaceOps–2021,6,x1294

Cluster-II: A Recommendation System for Semi-Automated Scheduling of Ground Station Passes

Jakob Karg[*], Julian Zobel[†], Bruno Sousa[‡], Steffen Bamfaste[§], Giulio Pinzan[¶], Artur Scholz[**]
[*] Solenix Deutschland GmbH, Darmstadt, Germany, jakob.karg@esa.int
[†] Technical University of Darmstadt, Germany, julian.zobel@kom.tu-darmstadt.de
[‡] European Space Agency, Darmstadt, Germany, bruno.sousa@esa.int
[§] Solenix Deutschland GmbH, Darmstadt, Germany, steffen.bamfaste@esa.int
[¶] European Space Agency, Darmstadt, Germany, giulio.pinzan@esa.int
[**] VisionSpace Technologies GmbH, Darmstadt, Germany, artur.scholz@esa.int

The Cluster-II mission by the European Space Agency consists of four identical spacecraft studying Earth's magnetosphere and its interaction with the solar wind. One major aspect of mission operations is the planning and scheduling of ground station passes to downlink collected data. These plans are created manually in time-consuming trial-and-error approaches, considering a vast number of constraints and competing goals. Previous attempts for automatic scheduling based on artificial intelligence performed significantly worse than manual plan creation. This paper proposes to combine human experience with modern constrained optimisation tools to improve and shorten the planning process by a semi-automated recommendation system with enhanced, interactive visual planning support. A user study showcased that the recommendation system significantly reduces the required planning time for both experienced and inexperienced users. Furthermore, a long-term comparison of historic schedules and automatic plan generation on the same data highlights the applicability of the proposed system to automatically generate valid schedules.
Keywords: Scheduling, Planning, Recommendation Systems, Optimisation

1. Introduction

Launched in the year 2000, the Cluster-II mission is among the oldest flying missions at the European Space Agency (ESA). The four spacecraft making up the mission fly in a close formation on a highly elliptical orbit and carry several instruments to study Earth's magnetosphere and its interaction with the solar wind [1]. The spacecraft are operated from the European Space Operations Centre (ESOC), where the Flight Control Team (FCT) conducts routine and contingency operations and manages the downlink of scientific data through a set of ground stations provided through the European Space Tracking Network (ESTRACK).

The scheduling of ground station passes for space missions is usually done by the operators of a ground station network. The different missions using one specific network provide the network operators with a list of their requirements. The operators take all requests they receive from all missions to create a schedule using dedicated tools and algorithms (cf. [2]). But scheduling the passes for ESA's Cluster mission is more challenging than for other missions due to the unusual orbit as well as the way data is produced and stored onboard, where certain science operations may either completely prevent simultaneous passes or limit the downlink data rate. In addition to that, the spacecraft are now well beyond their planned lifetime and some components underwent heavy degradation, imposing further constraints on the scheduling problem. Eventually, it has become infeasible to have the scheduling done by the ground station network operator due to the significant number of requirements. Instead, the scheduling is currently done manually by a member of the FCT in a process relying on experience and requiring trial-and-error approaches [3]. There have been previous efforts to automate this process by using artificial intelligence (AI) [3, 4, 5]. Due to the vast number of different constraints, however, the solutions generated by the AI-based tool were worse than the solutions created by the human planner [3]. Therefore, the AI tool is not being used in routine operations and, instead, the schedules are still created manually.

In contrast to fully automated approaches, this work proposes to combine available human experience with modern constrained optimisation tools. The scheduling process is improved by an interactive recommendation system and an enhanced, intuitive user interface. We present the prerequisites

16th International Conference on Space Operations, Cape Town, South Africa – 3.-5. May 2021.

for the application of such a recommendation system, including the representation of spacecraft behaviour and ground station visibilities, and formalise them in an optimisation problem. Based on a user study, in which members of the FCT created plans for several weeks, we found that the time required to create a plan for a certain time period is significantly shorter when using the recommendation system compared to the manual approach for both experienced and inexperienced planners.

In particular, we make the following contributions:

- We present a novel, semi-automated recommendation system for scheduling ground station passes within a highly constrained environment. Optimal passes are identified by solving the formalised problem functions with state-of-the-art constrained optimisation tools using a sliding window approach.

- The recommendation system is integrated as an interactive component in a visually enhanced version of the timeline visualisation tool *OpsWeb*, which provides planners with intuitive recommendations and the required information to improve and shorten the planning process.

- The benefits of the recommendation system on this process are highlighted in a user study with participants from the FCT. Furthermore, we demonstrate the quality of calculated recommendations by automatically generating plans with historic data and comparing these recommendation plans against the manually generated historic schedules based on the same data.

The rest of the paper is structured as follows. In Section 2, the Cluster-II mission and some of the constraints on the Cluster pass scheduling problem are introduced. Section 3 gives an overview of related work on scheduling problems and recommendation systems. In Section 4, the design and implementation of the recommendation system is explained. This includes the configuration of the user interface, as well as a description of the algorithm used to generate recommendations. Section 5 describes the evaluation of the system by a user study and a comparison to historic data. Finally, Section 6 summarises this work and suggests further improvements and future work.

2. The Cluster-II Mission

The Cluster-II mission started as one of the cornerstone missions in ESA's Horizon 2000 program. It consists of four identical spacecraft flying in a close tetrahedral constellation in a highly elliptical polar orbit, ranging from a height of about 20 000 km to about 110 000 km, studying the interaction of the solar wind with Earth's magnetosphere. Initially proposed in 1982, the satellites were launched on the maiden flight of the new Ariane 5 rocket in 1996. However, this flight was not successful and resulted in a break-up of the booster after 37 seconds. After this setback, ESA decided that the potential science return of the mission was so important that rebuilding the destroyed satellites would be worth the additional cost. The four newly built satellites were then launched in pairs of two on two Soyuz rockets in July and August 2000 as the Cluster-II* mission [1]. Initially intended to last two years, the mission has been extended multiple times and it is still running and producing valuable scientific data today.

The scheduling of ground station passes for the downlink of scientific data from the Cluster spacecraft is subject to a variety of different requirements and constraints. One of the main issues is the limited storage space of the onboard mass memory, which will result in a loss of scientific data if a downlink is not possible in time and the storage overflows. Additionally, the monetary cost to use ground stations for data downlink should be kept low and, therefore, the goal is to downlink all data in as little time as possible. The Cluster spacecraft can downlink data in two different modes, high bitrate (hbr) and low bitrate (lbr). The available data rate depends on the link budget, which varies significantly, depending on the ground station used and the position of the spacecraft in the highly elliptical orbit. The satellites carry a High Power Amplifier (HPA), which, when enabled, allows using high bitrate throughout the entire orbit. Without the HPA, high bitrate is only available in parts of the orbit close to the perigee, as indicated in Fig. 1. Due to the long duration of the mission, the batteries are not usable any more and the solar arrays have degraded significantly. Thus, available power onboard is heavily limited and it is not possible to operate the HPA and the scientific instruments at the same time. To maximise the collection of scientific data, the HPA is usually disabled and passes have to be scheduled with that limitation.

As a consequence of the close formation of the spacecraft, it is sometimes possible to schedule Multiple Spacecraft per Aperture (MSPA) passes. There, one ground station is used to downlink data from

*For the sake of readability, the suffix 'II' is omitted when talking about the Cluster-II mission in the rest of this work.

16th International Conference on Space Operations, Cape Town, South Africa – 3.-5. May 2021.

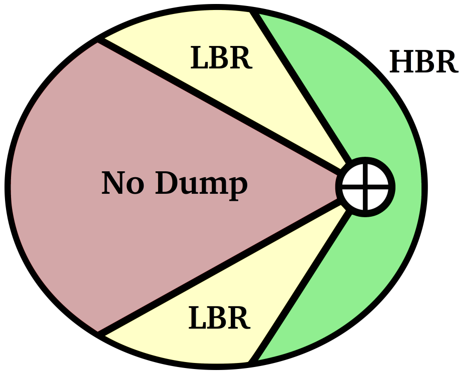


Fig. 1: The orbit of the Cluster spacecraft around the Earth and the achievable bitrate at different parts of the orbit when the HPA is disabled. If the HPA onboard the spacecraft were enabled, high bitrate would usually be achievable throughout the orbit. Figure adapted from [6].

two different spacecraft, reducing ground station usage time and cost. However, only one of the two spacecraft can be commanded during MSPA passes, which may impose a severe limitation and confines the applicability of MSPA in certain situations.

Finally, while the operations of the four Cluster spacecraft are highly automated and routine passes can be fully handled by the automation system [7, 8], it is still favoured to have the control room staffed during passes to handle potential anomalies. For this, a team of four spacecraft controllers is available. Since four people are not enough to cover 24/7 operations when accounting for holidays and other absences, the control room cannot be staffed at all times. Therefore, when planning the passes, the organisation of control room shifts also has to be taken into account. This is done by scheduling passes for all four spacecraft in parallel, making use of different ground stations. With this approach, more data can be transmitted within one shift and there are extended periods between the shifts without passes, where the control room can be unstaffed. As already stated, the operators of the ground station network have no insight into most of these factors and the pass scheduling is done manually by members of the FCT. Each week, the ground station network operators first create a schedule with passes for all missions except Cluster. This schedule is then provided to the planner in the Cluster FCT, who creates the plan for Cluster by checking the remaining ground station availabilities and adding passes one shift at a time.

3. Background and Related Work

Within this section, we provide an overview of the most important related work and background knowledge for this work, which is the scheduling of ground station passes for a set of spacecraft and recommendation and decision support systems in general.

3.1 *Satellite Range Scheduling*

The problem of allocating tasks among a set of satellites and a set of earthbound objects is known as Satellite Range Scheduling (SRS). It addresses not only the scheduling of ground station contacts, but also scheduling tasks such as target observation on the Earth by instruments onboard the spacecraft. Possible approaches to solve such scheduling problems are manual, automated, and mixed-initiative planning [9]. To schedule ground station passes, the spacecraft operators use orbital models to identify visibility windows during which contacts are possible. The duration of these windows depends on the spacecraft orbits and the position of the ground stations. Based on available windows and the specific mission requirements, the spacecraft operators generate a set of requests including minimum or maximum duration of a contact, the earliest or latest execution time, and, in some cases, the priority of the request. These requests are forwarded to the ground station network operator, who generates a schedule from the requests of all missions. This process is followed, for example, by ESTRACK for all ESA missions except Cluster [10], by the US Air Force Satellite Control Network [2], and by NASA's Deep Space Network (DSN) [11]. Typically, this kind of problem is oversubscribed, since there are more requests than can be accommodated [2, 12]. The goal is to schedule as many requests as possible, also considering prioritised requests in some cases [13].

For the Cluster mission the pass scheduling problem is slightly different. Typically, there is enough ground station time available to downlink all data. The objective is, therefore, not to schedule as many passes as possible, but to schedule the optimal set of passes. However, defining an optimal schedule is non-trivial due to the large set of constraints. Additionally, subsequent passes of Cluster are not independent from each other, in contrast to most SRS problems. Each pass of a single spacecraft depends on fill level evolution of the mass memory since the last pass, which is not necessarily a linear increase over time. Similarly, passes of different spacecraft depend on each other based on factors such as shifts in the control room and MSPA opportunities. There-

16th International Conference on Space Operations, Cape Town, South Africa – 3.-5. May 2021.

fore, many algorithms and approaches applicable on general SRS problems are not directly applicable to Cluster.

3.2 *Recommendation and Decision Support Systems*

Decision support systems are used in various different domains and are meant to aid decision makers with their analysis by presenting them relevant information and allowing them to interact with the system, compare different scenarios, and perform 'what-if' trade-off analysis [14, 15]. Recommendation systems can be seen as a subclass of decision support systems. While in a decision support system the focus is on structuring information, presenting it in a usable format, and allowing 'what-if' analysis, recommendation systems take things one step further by recommending specific items of interest to the user [16].

Since these systems present the consequences of a decision in an understandable form, they are especially useful for problems where optimality is hard to define, when users lack experience, or when the information space is too large for the user to consider all the data [15, 16]. Therefore, decision support systems and recommendation systems have applications in many different areas, such as healthcare [17], entertainment [18], computer security [19], software engineering [16], or, as in this case, scheduling problems [20].

Ecker *et al.* [15] introduce a framework to apply decision support systems to scheduling problems. They identify three main components that are typical for decision support systems:

1. Database Management System
2. Model-based Management System
3. User Interface Management System

The Database Management System stores the information relevant to the specific decision problem. The Model-based Management System implements different domain specific models. They act on the stored data in the Database Management System and the user input to generate additional information or to validate decisions. In a recommendation system, this component is adapted to analyse the data and generate recommendations, thereby forming the recommendation engine [16]. The User Interface Management System is the frontend of the application, which presents both the data stored in the Database Management System and created by the Model-based Management System, allowing the user to explore the search space. Ecker *et al.* propose to implement the Model-based Management System for scheduling problems using a constraint satisfaction formulation [15]. This allows the identification of feasible schedules and applies to a wide variety of scheduling problems. However, they also note that an implementation of this approach is highly dependent on the specific areas and problem domains.

Fagerholt [20] successfully uses a decision support system for the scheduling of a vessel fleet for ocean shipping by a heuristic algorithm. Before the decision support system was introduced, the vessel fleet scheduling problem was also solved manually by experienced planners. This problem, similar to the Cluster pass planning problem, has many complex and interdependent constraints making an optimal solution hard to define. But due to the different nature of the problem domains, the constraints themselves and the models describing the problem are very different and cannot be adapted easily.

4. Design and Implementation

Within this work, we apply the general approach of a decision support system that includes and aids the experienced human planners in the process of plan creation to the complex problem of scheduling ground station passes in the Cluster mission. The problem is formalised as a constrained optimisation problem and solved by a recommendation system that is integrated within an interactive and intuitive user interface. For the ease of understanding, we first describe the frontend user interface followed by an in-depth description of the recommendation system in the backend.

4.1 *User Interface*

To manually create a ground station pass schedule, the planner requires a variety of information, such as the availability, time, and duration of ground station visibility windows and the spacecraft memory fill levels. The FCT currently uses a self-developed tool called *ClusterWeb* to display all this information in a timeline for the planner to use. But *ClusterWeb* was developed early in the mission and is both technologically outdated and difficult to maintain. Hence, a new timeline visualisation tool called *OpsWeb* [21, 22] has been developed at ESOC using modern web technologies like the Angular framework [23] and D3.js [24]. The *OpsWeb* backend is based on Python and the Django web framework [25]. A RESTful API provides information to the frontend and external services.

Within a planning tool, information should be presented in an accessible and intuitive manner. Specifi-

Jakob Karg, Julian Zobel, Bruno Sousa, Steffen Bamfaste, Giulio Pinzan, Artur Scholz. *Cluster-II: A Recommendation System for Semi-Automated Scheduling of Ground Station Passes.*
In Proc. 16th International Conference on Space Operations (SpaceOps), 2021.

Fig. 2: The improved timeline visualisation in the new tool *OpsWeb*, displaying passes, science modes, the mass memory fill level, and ground station visibilities and their properties all in one view.

cally, the following information was identified as necessary for the planner:

1. The mass memory fill level of each spacecraft and how it evolves over time.
2. The science operating modes of each spacecraft.
3. The time and duration of ground station visibility windows.
4. The bookings of ground stations by other missions.
5. The theoretically achievable bitrate, as given by the link budget, for each visibility window.
6. The availability of MSPA passes.

Additionally, as already stated, the visualisation shall allow the 'what-if' analysis for human planners. The old tool *ClusterWeb* only offers very basic support for this, since not all relevant information is displayed immediately, but needs to be accessed manually one by one. Further, making changes like modifying a pass requires multiple steps, making the comparison of different plans very slow.

The interactive visualisation with the recommendation system presented in this work is built on top of *OpsWeb*. We designed the user interface to display all required inputs in the main view. An excerpt for one of the four spacecraft can be seen in Fig. 2. Passes are displayed by blue boxes and labelled with the ground station and the start and end time. The background colour gives the operating modes of the scientific instruments onboard and the green line shows the fill level of the mass memory. Ground station visibilities are displayed with purple bars, with times where the ground station is booked for another mission crossed out by a darker line. Below each visibility, the possible bitrate as given by a link budget model is indicated in green (high bitrate), yellow (low bitrate), or red (no dump possible). MSPA opportunities are displayed by grey bars over a visibility. Additional details are displayed when the user hovers the mouse over an item. To improve the overview, only visibilities that fulfil a minimum requirement of usable downlink time are displayed, unusable visibilities are filtered out.

Once the planner starts the recommendation system, the generated pass recommendations are displayed in the timeline as well and can be accepted, modified, or deleted by the planner. Since the tool should allow the planner to interactively create a plan, there is a time limit on how long the creation of recommendations should take. When the planner takes an action, like modifying a pass, a new set of recommendations based on the new situation should be generated in the order of a few seconds. This limits the possible complexity of the algorithm generating the recommendations, which will be discussed later on.

4.2 *Creating Recommendations*

As discussed in Section 3, the problem of creating pass recommendations can be modelled as a constrained optimisation problem. An objective function that describes the quality of a schedule is optimised with respect to certain constraints. These constraints are imposed by the ground stations, the spacecraft, and the mission objectives. They are the following for the Cluster pass scheduling problem:

- All data generated by the spacecraft should be downlinked, no data should be lost.
- Some science operations cannot be paused and

prevent the parallel downlink of data.

- Utilisation of ground station time should be minimal to save cost.
- As a consequence of the previous constraints, the number of high bitrate passes and MSPA opportunities should be maximal.
- Multiple passes should be grouped to simplify the shifts for spacecraft controllers.

The main challenge is the definition of a suitable objective function, since the notion of an optimal schedule is hard to define due to this variety of partially competing goals. In the following, we will discuss the formalisation of these goals and constraints as well as the approach to define and find an optimal solution.

4.2.1 *Formalisation of Constraints*

Altogether, the formalisation of the Cluster pass scheduling problem is comprised of 14 different hard constraints and 4 soft constraints. Within the scope of this paper, we limit this to provide a basic overview of our formalisation.

Booked passes $\mathbf{p}$, visibilities $\mathbf{v}$, and the recommendations created by the algorithm $\mathbf{r}$ are given by vectors consisting of the spacecraft sc, ground station gs, start time $start$, and end time end:

$$\begin{aligned} \mathbf{p} &= (p_{\text{sc}}, p_{\text{gs}}, p_{\text{start}}, p_{\text{end}}) \in SC \times GS \times \mathbb{R} \times \mathbb{R}, \\ \mathbf{v} &= (v_{\text{sc}}, v_{\text{gs}}, v_{\text{start}}, v_{\text{end}}) \in SC \times GS \times \mathbb{R} \times \mathbb{R}, \\ \mathbf{r} &= (r_{\text{sc}}, r_{\text{gs}}, r_{\text{start}}, r_{\text{end}}) \in SC \times GS \times \mathbb{R} \times \mathbb{R}. \end{aligned} \quad (1)$$

The set of spacecraft SC and the set of ground stations GS are given by

$$\begin{aligned} SC &= \{\text{CLU1, CLU2, CLU3, CLU4}\}, \\ GS &= \{\text{KIR1, KRU, MSP, NNO, VIL1, YAT}\}. \end{aligned} \quad (2)$$

The pass start and end time are given as real numbers describing the number of seconds after the start of the planning horizon.

With this definition, constraints are modelled using first order logic. For example, the basic constraint that all recommended passes have to occur during a visibility is given by

$$\begin{aligned} &\forall \mathbf{r} \in \textit{Recommendations}\ \exists \mathbf{v} \in \textit{Visibilities}: \\ &r_{\text{sc}} = v_{\text{sc}} \land r_{\text{gs}} = v_{\text{gs}} \land r_{\text{start}} \geq v_{\text{start}} \land r_{\text{end}} \leq v_{\text{end}}. \end{aligned} \quad (3)$$

Additional constraints relating to the maximum or minimum time between different passes, as well as the relation between passes and science modes, were added in the same manner.

Constraints related to the memory fill level are defined using a fill level model, which provides the function

$$\textit{fill}: SC \times \mathbb{R} \to [0, 100], \quad (4)$$

where $\textit{fill}(sc, t)$ gives the memory fill level of spacecraft sc at time t in percent based on the scheduled science modes and previous passes.

The mass memory should never fill up completely, since this would result in the loss of data. To keep some margin in case of ground station anomalies, there should always be a pass before the fill level reaches 80 %:

$$\begin{aligned} &\forall \mathbf{r} \in \textit{Recommendations}, \forall sc \in SC: \\ &\textit{fill}(sc, r_{\text{start}}) \leq 80. \end{aligned} \quad (5)$$

Similarly, passes should end once the memory was downlinked completely:

$$\begin{aligned} &\forall \mathbf{r} \in \textit{Recommendations}: \\ &r_{\text{end}} - r_{\text{start}} \leq \frac{\textit{fill}(r_{\text{sc}}, r_{\text{start}})}{\textit{Downlink Rate}}. \end{aligned} \quad (6)$$

As described above, it is favourable to take as many passes as possible in high bitrate mode to minimise the cost associated to ground station usage. In routine operations, there are usually enough ground station visibilities available to downlink all data using high bitrate only. Therefore, we simplify the problem by adding a constraint that all recommendations should be high bitrate passes. If a low bitrate pass is necessary in special circumstances, the planner can add it manually.

4.2.2 *The Objective Function and Soft Constraints*

As described above, defining optimality for an entire planning period of one week is not possible due to the partially competing goals. Instead, inspired by the approach of the human planner, we create the schedule one shift at a time. Defining optimality for one shift is straightforward. To make best use of the time where the control room is staffed, as much data as possible should be downlinked during each shift. This can easily be formalised and solved by a constrained optimisation solver:

$$\begin{aligned} &\max f_1, \text{ where} \\ &f_1(\textit{Recommendations}) = \\ &\sum_{\mathbf{r} \in \textit{Recommendations}} \left((r_{\text{end}} - r_{\text{start}}) * \textit{Downlink Rate}\right). \end{aligned} \quad (7)$$

Jakob Karg, Julian Zobel, Bruno Sousa, Steffen Bamfaste, Giulio Pinzan, Artur Scholz. *Cluster-II: A Recommendation System for Semi-Automated Scheduling of Ground Station Passes.*
In Proc. 16th International Conference on Space Operations (SpaceOps), 2021.

16th International Conference on Space Operations, Cape Town, South Africa – 3.-5. May 2021.

Since all recommendations are limited to high bitrate only, maximising the downlinked data volume is equal to maximising the summed duration of all passes, since the constraint in Eq. 6 limits the maximum pass duration to prevent the memory from running empty during the downlink:

$$\begin{aligned} &\max f_2, \text{ where} \\ &\quad f_2(\mathit{Recommendations}) = \\ &\quad\quad \sum_{\mathbf{r} \in \mathit{Recommendations}} (r_{\text{end}} - r_{\text{start}})\,. \end{aligned} \tag{8}$$

This approach is shown in Fig. 3. The planning horizon for the algorithm starts with the end of the last scheduled shift, indicated by the last booked ground station pass, as shown in Fig. 3a. The end of the planning horizon is given by Eq. 5, i.e., when the first spacecraft reaches a memory fill level of 80 %. This is the latest instance a pass should start based on safety margins to overcome possible contingencies on ground stations. Within this planning horizon, recommendations are calculated for one shift only, and the planning horizon updated accordingly, as shown in Fig. 3b. Scheduling one shift at a time significantly reduces the considered length of the planning horizon and the number of passes that need to be scheduled. Therefore, the search space with this approach is smaller, which reduces the computational complexity and the time it takes to find an optimal recommendation.

To further specify certain limitations, additional soft constraints are introduced to the objective function. As an example, if the mass memory of one spacecraft were still relatively full at the end of a shift, an overflow would be imminent in the next shift and require a quick resolution. Thus, the available planning horizon for the next shift would be very short, potentially resulting in an overall sub-optimal schedule. To prevent this, the fill level of all spacecraft should be low at the end of a shift. This is formalised as a soft constraint by applying a penalty to the objective function:

$$\mathit{penalty}(\mathbf{r}) = \begin{cases} 0 & \text{if } \mathit{fill}(r_{\text{sc}}, r_{\text{end}}) \leq 40, \\ 60 & \text{if } 40 < \mathit{fill}(r_{\text{sc}}, r_{\text{end}}) \leq 50, \\ 120 & \text{if } \mathit{fill}(r_{\text{sc}}, r_{\text{end}}) > 50. \end{cases} \tag{9}$$

As a result, scenarios in which all spacecraft have medium fill levels at the end of a shift are favoured over scenarios in which some spacecraft have low and some have high fill levels, making the scheduling of the next shift easier.

Conclusively, the final objective function used for the constrained optimisation problem is given by

$$\begin{aligned} &\max f, \text{ where} \\ &f(\mathit{Recommendations}) = \\ &\quad \sum_{\mathbf{r} \in \mathit{Recommendations}} ((r_{\text{end}} - r_{\text{start}}) - \mathit{penalty}(\mathbf{r}))\,. \end{aligned} \tag{10}$$

4.2.3 *Implementation*

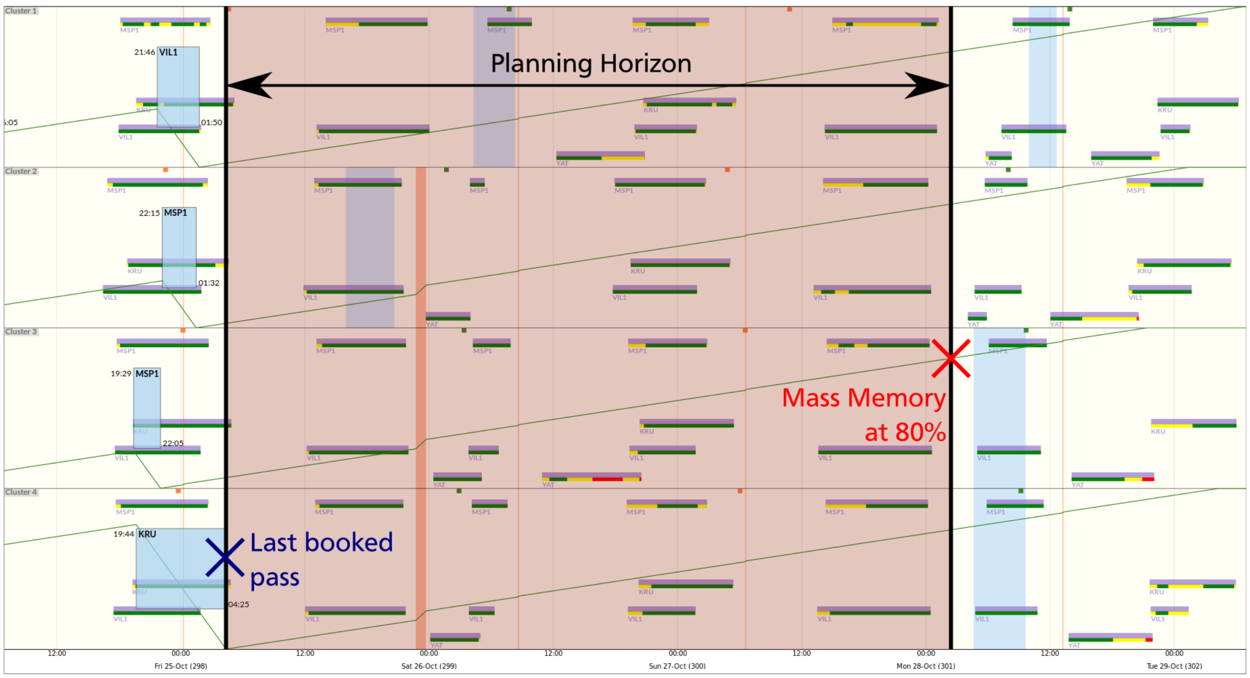
We integrated the ground station pass recommendations component into the *OpsWeb* backend. Available interfaces were extended to provide access to the recommendation component from the frontend and vice versa. Furthermore, we exposed available interfaces that provide access to ground station visibility windows, science operations, and similar information to the recommendation component.

The constrained optimisation problem as given in Eq. 10 with all hard and soft constraints is implemented in the backend using the theorem prover *z3* [26]. It is a freely available, open source optimisation framework that allows to easily model constraints and optimisation functions. Additionally, the *z3* Python wrapper allows a straightforward integration into the *OpsWeb* backend.

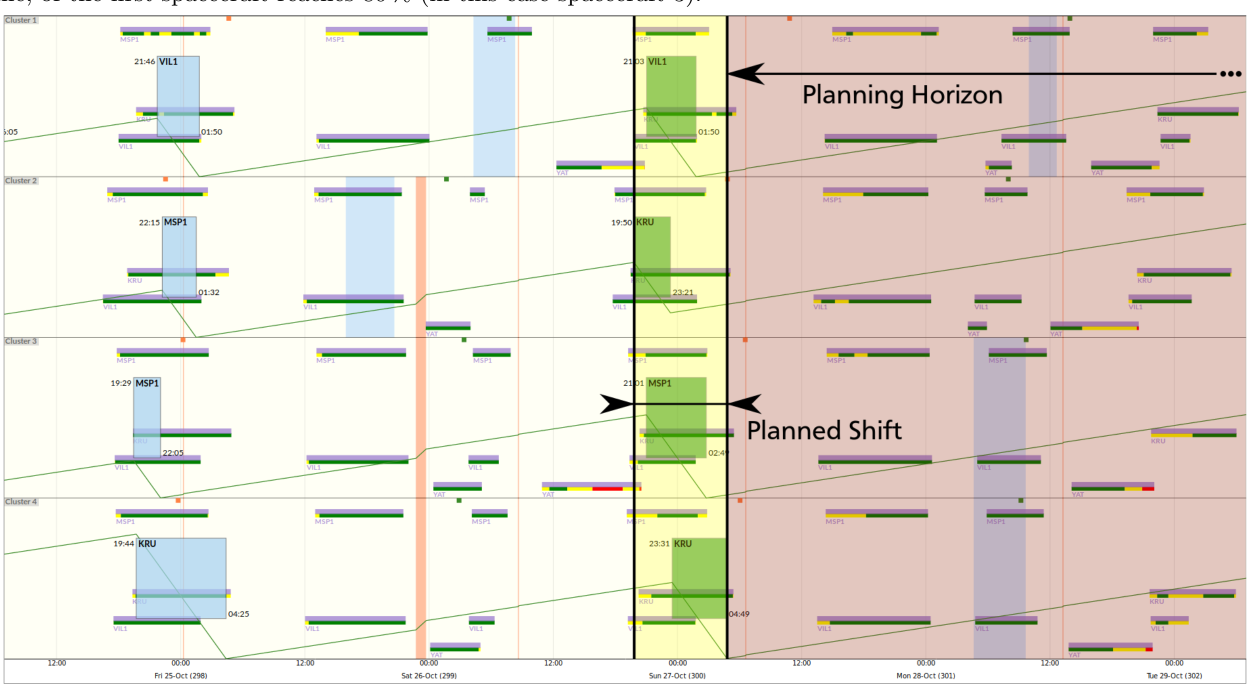
The vectors for recommendations **r** are modelled as a set of variables as defined in Section 4.2.1. These variables are set as target variables for *z3* that need to be found. The constraints of the scheduling problem were similarly modelled as *z3* constraints. The objective function described in Eq. 10 and the *z3* constraints were combined in an optimisation object, which then is used by the *z3* framework to solve the optimisation problem.

When initialised, the recommendation component gathers all required information for the scheduling of ground station passes, populates the *z3* model, and provides the calculated results to the *OpsWeb* frontend. Here, the information and the recommendations are visualised together. The planner can interact with the given recommendations and is able to accept, modify, or delete them. Every time the planner modifies the schedule or a specific ground station pass, the recommendations are updated to take the new situation into account. Similarly, new recommendations are created when the former recommendations are accepted for the schedule.

Calculating multiple pass recommendations within the planning horizon is, nevertheless, complicated, most notably due to the interdependency between passes on the same spacecraft. An earlier pass



(a) The planning horizon of the proposed scheduler. The planning horizon of the scheduler starts at the end of the last booked pass (here for spacecraft 4) and goes until the time where the memory fill level, as indicated by the green line, of the first spacecraft reaches 80 % (in this case spacecraft 3).



(b) The pass recommendations created by the recommendation engine inside the previous planning horizon. The planning horizon is then updated for the next shift.

Fig. 3: The planning horizon starts with the end of the last booked pass and ends when the first spacecraft reaches a memory fill level of 80 % (top). The shift was selected to allow for the highest possible data volume while considering all constraints (bottom).

Jakob Karg, Julian Zobel, Bruno Sousa, Steffen Bamfaste, Giulio Pinzan, Artur Scholz. *Cluster-II: A Recommendation System for Semi-Automated Scheduling of Ground Station Passes.*
In Proc. 16th International Conference on Space Operations (SpaceOps), 2021.

16th International Conference on Space Operations, Cape Town, South Africa – 3.-5. May 2021.

reduces the memory fill level, which consequently changes the requirements for later passes. To find the optimal passes in one step, even within the limited scope of a single shift, thus would require considering all possible combinations. Because of the imposed long run times for the recommendation algorithm, this would prevent an interactive planning process that provides the human planner with quick updates and recommendations for every change that is made.

To overcome this issue, we use a two-step approach for the creation of recommendations instead. In each step, a maximum number of one pass per spacecraft – in total a maximum of four passes – is scheduled. The first step schedules up to one pass per spacecraft such that the overall downlinked data volume is maximised, as shown in Fig. 3b. After that, the evolution of the memory fill level is updated accordingly. If there are usable ground station windows left during the shift and the memory was not completely emptied, a second step then adds again up to one additional ground station pass per spacecraft, for a total maximum of eight passes after both steps, as shown in Fig. 4. The passes recommended by the algorithm are shown in green and numbered with the step in which they were created. For spacecraft one and four, the pass created in the first step is sufficient to fully empty the mass memory by the end of the shift. For spacecraft three, an additional pass was added in the second step, since a ground stations was available and there was still data left to be transmitted. Spacecraft two also has data left to be dumped, but a science operation as indicated by the blue background prevented further downlink operation. The red pass for spacecraft one in Fig. 4 is not recommended by the algorithm but shows why keeping track of the memory fill level between the steps is necessary. Looking at the red pass only, it would make sense to schedule it, since it is during the same shift as the other passes and the memory is not empty during this pass. However, adding this pass would make the later pass added in the first step unnecessarily long, wasting ground station time. Hence, the fill level remaining after the first step is taken into account when creating passes in the second step to prevent this.

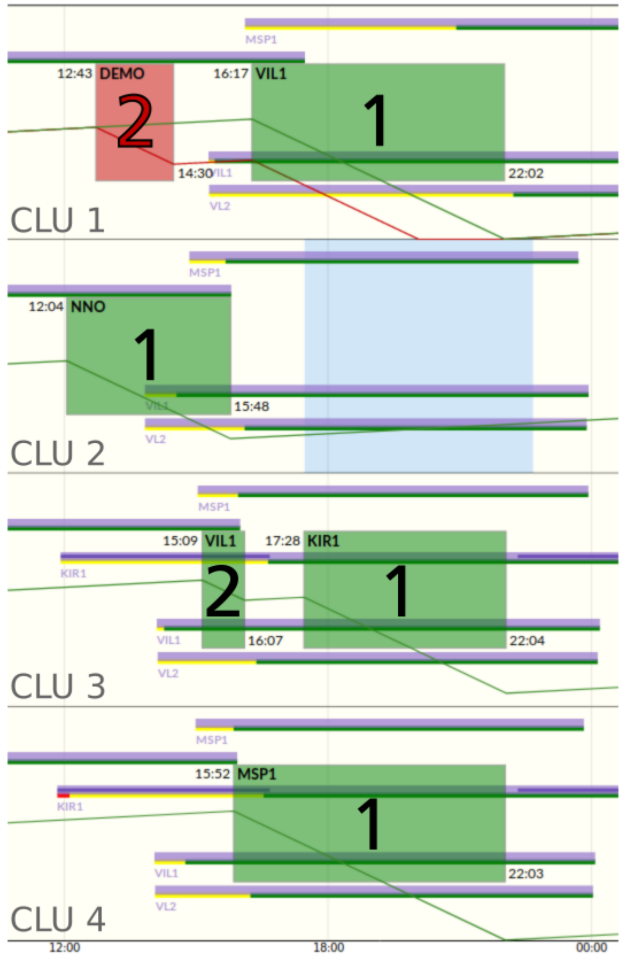


Fig. 4: The passes in green are a set of recommendations generated by the recommendation algorithm. Numbers label the step during which they were created. The red pass of spacecraft one would not be recommended, since the memory would run empty during the later pass.

Besides the given recommendations for conceptually optimal plans, a major aspect of our approach is that the human planner is not constraint to these recommended passes. Especially soft factors like personal preferences of individual staff in the control room for shift start and end times or requirements for extra ground contacts to support special operations like manoeuvres or eclipses can hardly be integrated in a formal model. It is much easier for a human planer to consider such factors, but also to make the decision to tolerate a less optimal plan for individual requirements. Nevertheless, the recommendations provide the best choices given the formalised constraints and, therefore, should reduce the required time to create a pass schedule, which is evaluated in the next section.

5. Evaluation

This work provides a twofold evaluation of the proposed recommendation system. First, we conduct a small user study that assesses the impact of the given recommendation system on the manual planning process. We compare the presented system based on *OpsWeb* against the fully manual planning process in the currently used tool *ClusterWeb*. Secondly, we used the recommendation system to retro-actively create plans for the years 2016 – 2019 by accepting all recommendations without changes. These automatically generated schedules are compared against the available, manually created historic schedules of the same time period by a set of quality metrics.

5.1 *User Study*

Both the improved user interface of *OpsWeb*, as well as the pass recommendations are expected to have an impact on the scheduling process. We asked four members of the Cluster FCT to create pass schedules in three scenarios: (i) Manual planning using the current *ClusterWeb* tool, (ii) manual planning without recommendations using *OpsWeb*, and (iii) manual planning with recommendations using *OpsWeb*. For each scenario, each participant created a schedule for one week. The dates of the weeks were chosen such that they are representative for routine operations where no manoeuvres or eclipses occur. For the comparison of scenario (ii) and (iii), the participants were split in two groups to counterbalance the bias of planning different weeks with and without the recommendation system. Thus, one group planned a week using scenario (ii), while the other group planned the same week using scenario (iii), and vice versa. Similarly, the execution order of the scenarios was shuffled for each participant. We measured the time it took each participant to create the different schedules in the different scenarios.

Note that scenario (i) serves as a baseline, providing necessary information on the individual performance of participants. The measured planning time correlates with the subjective expertise of each participant, which is based on how often the task is performed in regular work time. In this scenario, the most experienced planner of the FCT required around 30 minutes to plan one week. In contrast, the less experienced planners required between one hour and two hours for the same task.

Within the second scenario, using the improved user interface of *OpsWeb* without recommendations, a significant decrease in the required time can be noted in comparison to the first scenario for all participants. Most planners required around 30 minutes to plan one week, with the fastest completing the planning process in just over 15 minutes. This result may affirm our hypothesis, that the improved user interface eases the planning process by providing an intuitive and direct visualisation of all required information in one view.

With interactive recommendations activated in the third scenario, the required planning time was reduced further. The direct comparison of the required planning times for each participant with and without recommendations, respectively, is presented in Table 1. All four participants required around 10 minutes to complete one week of planning, which is only about a third of the time without recommendations.

In general, we conclude that the goal of improving the planning process with the new recommendation system was achieved. Especially for the inexperienced planners there were significant time savings and it can be observed that with the improved user interface and the recommendation system the planning times are largely independent of the planners experience level, which was not the case for the *ClusterWeb* tool. Modifications of the recommendations were mostly done due to personal preferences of shift times and shift duration, resulting in less efficient shifts that however suited the work time of the FCT much better.

5.2 *Long-term Comparison of Recommendation Quality on Historic Data*

While the user study provides some measurement of the impact of recommendations on the scheduling process, it offers no insight into the quality of the created recommendations. For this, we performed a long-term assessment of the recommendation quality based on historic data and the manually created schedules for this period. The schedules were automatically created by calculating recommendations as described in Section 4.2 and accepting all of them without change. Records of the same data that is currently used as input were collected for the years of 2016 – 2019. We then compared the automatically generated schedules with the manually created schedules during this time using the following metrics:

- **Tracking Hours** specifies how many ground station hours are used; it is the summed duration of all passes of all four spacecraft.
- **Downlink Hours** measures the time data is downlinked from the Cluster spacecraft. Because

Jakob Karg, Julian Zobel, Bruno Sousa, Steffen Bamfaste, Giulio Pinzan, Artur Scholz. *Cluster-II: A Recommendation System for Semi-Automated Scheduling of Ground Station Passes.*
In Proc. 16th International Conference on Space Operations (SpaceOps), 2021.

Table 1: The times it took the probands to plan each week. Depending on the group, either week 1 or week 2 was planned with the interactive recommendation engine enabled.

	Week 1		Week 2	
	Interactive Recommendations			
Proband	on	off	on	off
A	-	31 minutes	9:45 minutes	-
B	9:33 minutes	-	-	33:30 minutes
C	-	27:30 minutes	8:30 minutes	-
D	11:40 minutes	-	-	15:30 minutes

data is downlinked from two spacecraft at the same time using a single ground station during MSPA passes, downlink hours is typically larger than tracking hours w.r.t. the number of used MSPA passes.

- **Billed Hours** is a metric for the cost associated to the usage of a ground station. In addition to the number of ground station hours, it also depends on additional factors such as the station reconfiguration time, which, in some cases, is billed to the mission as well.
- **Shifts per Week** states for how many shifts spacecraft controller need to be present in the control room to cover all passes, measured per week. It indicates how well passes are clustered together and how hard it would be to create a shift plan.
- **Mean Shift Length** gives the average duration of all shifts during the period.

The mean values for the given metrics are provided for both the manual historic and the automatic schedules in Table 2. The values for the downlink, tracking, and billed hours are given as monthly averages aggregated for each year.

Similarly, Fig. 5 provides the same metrics as a box plot visualisation for weekly aggregates of the year 2018. The other years in the considered time period show similar results. The left box in solid lines shows the automatically generated plan and the right box in dashed lines shows the manual plan for each metric, respectively. The bold purple line is the median, the box denotes the 25th and 75th percentiles, and the whiskers represents the 5th and 95th percentiles. Values outside this range are considered outliers and drawn with individual markers. The colour of the boxes match the colour of the respective metric on the ordinate.

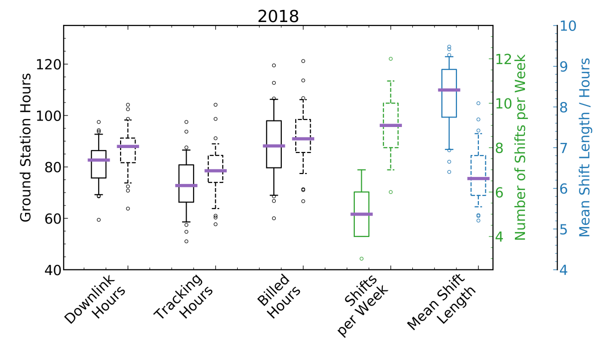


Fig. 5: Box plot comparison of the different metrics aggregated for each week. For each metric, the left box denotes the generated and the right box the manual historic plan. The bold line denotes the median, boxes the 25th and 75th percentiles, whiskers the 5th and 95th percentiles. Outliers are drawn with individual markers.

Comparing the number of shifts and the average shift length shows that the recommendation system generally schedules fewer, but longer shifts compared to the human planner. This is due to the objective of downlinking as much data during a shift as possible, which has the side effect of extending shifts as much as possible. This could be adjusted according to preferences by modifying the constraints for the maximum shift duration or by adding penalties for longer shifts. Comparing the downlink, tracking, and billed hours, it can be seen that the automatic and the manual plan behave relatively similar. The automatic plan generally uses fewer hours than the manual plan in all three metrics. One reason for this is that the manual plan contains additional passes for special operations, such as manoeuvre preparation and verification or eclipse operations. The recommendation engine creates passes only for routine operations based on the available data, and there-

16th International Conference on Space Operations, Cape Town, South Africa – 3.-5. May 2021.

Table 2: The different plan metrics over the years for automatically generated plans with the recommendation system and the historic plans created manually over the same period. On average, the two approaches perform very similar, but the automatic generation results in fewer, but longer shifts in general.

Year	Plan	Mean Monthly Downlink Hours	Mean Monthly Tracking Hours	Mean Monthly Billed Hours	Shifts per Week	Mean Shift Length
2016	manual	347.1 h	351.7 h	441.2 h	9.4	5.9 h
	auto	316.5 h	292.6 h	354.9 h	5.2	7.3 h
2017	manual	371.8 h	371.8 h	412.6 h	9.1	6.2 h
	auto	311.7 h	311.7 h	377.2 h	4.9	8.1 h
2018	manual	360.3 h	329.5 h	384.9 h	8.6	6.3 h
	auto	333.8 h	300.8 h	362.8 h	4.7	8.4 h
2019	manual	372.1 h	354.5 h	415.3 h	7.0	6.4 h
	auto	338.2 h	320.5 h	388.4 h	4.0	8.6 h

fore, cannot include such activities in this evaluation. Also, note that the ordinates do not start at zero due to visualisation reasons.

Although the algorithm is designed to be part of an interactive, human-controlled tool, it showed that automated schedule creation in this iterative, sequential approach is possible nonetheless. The results of the automatic assessment show that the recommendation system provides feasible schedules for long periods without human interaction. However, this is true for routine operations but not for special operations and constraints that may arise during the live mission operation still require a human planner to interfere. Nevertheless, we conclude that the recommendation system in its current state is capable of providing recommendations of high quality that are directly comparable to the results of an experienced human planner. The recommendation system is, thus, applicable for the scheduling of ground station passes even with inexperienced planners. And as shown in the first part of this evaluation, using the recommendation system greatly enhances the planning efficiency of a human planner regardless of the experience level.

6. Conclusion and Outlook

In this paper, we presented an interactive, semi-automated recommendation system to improve the process of creating a ground station pass schedule for ESA's Cluster mission. The vast number of constraints and competing goals in the Cluster pass planning problem are the reason that plans created in previous automation efforts performed worse than plans created manually by experienced team members. We formalised these constraints as a constrained optimisation problem and used the open source theorem prover *z3* to solve this optimisation problem. By combining the experience of human planners with an interactive and visually intuitive recommendation tool, we provide a significant improvement to the overall planning process. In a user study with members of the Cluster FCT, planning a full week of ground station passes took only a third of the time with the recommendation tool compared to without the tool. Furthermore, we highlighted that the human-like iterative approach for the creation of shift-wise pass recommendations can also be used to automatically create plans for routine operation by comparing four years of historic plans with generated plans of the same time period.

Future work may include the improvement of the recommendation system, for example by adding more quality-of-life features in the *OpsWeb* frontend, or extending the set of hard or soft constraints in the backend. As already stated, the adaptation of constraints like the maximum shift length or including a penalty for prolonged shifts is also possible to suit the needs of individual planners and operators. This could also include the use of shift plans and individual constraints for each of the operators in the respective shifts as input for further optimisation.

At the time of writing this paper, *OpsWeb* and the recommendation tool are gradually integrated into the everyday mission operation of the Cluster FCT and the Cluster mission itself. With the modular definition of constraints and their implementation, this tool can be adapted to suit altering or completely new needs in the ground station pass planning. Furthermore, is ensures that the planning process can also be performed by inexperienced users, in an efficient and intuitive manner.

16th International Conference on Space Operations, Cape Town, South Africa – 3.-5. May 2021.

Acknowledgement

The authors would like to thank the Cluster FCT at ESOC for the opportunity to work on this project, for providing feedback during the development process, as well as for participating in the user study. Similarly, the authors would like to thank the Multimedia Communications Lab (KOM) at the Technical University of Darmstadt, Germany for their continuous support throughout this work.

References

[1] A. Gianolio and J. Ellwood, "Cluster II – The spacecraft and the Mission," in *51st International Astronautical Congress*, (Rio de Janeiro, Brazil), p. 8, International Astronautical Federation, 2000.

[2] L. Barbulescu, A. Howe, and D. Whitley, "AFSCN scheduling: How the problem and solution have evolved," *Mathematical and Computer Modelling*, vol. 43, pp. 1023–1037, May 2006.

[3] S. Fratini, N. Policella, N. Faerber, R. N. Neves Madeira, A. Donati, and B. Teixeira De Sousa, "Ground Station Pass Allocation in Operations using AI," in *15th International Conference on Space Operations*, (Marseille, France), American Institute of Aeronautics and Astronautics, May 2018.

[4] N. Faerber, S. Fratini, M. Bartesaghi, T. Costa, B. Teixeira De Sousa, N. Policella, and A. Donati, "Cluster-II: Using Artificial Intelligence for Automated Ground Station Scheduling," in *SpaceOps 2016 Conference*, (Daejeon, Korea), American Institute of Aeronautics and Astronautics, May 2016.

[5] N. Faerber, *Automation of Ground Station Scheduling for the Cluster Mission Using an Artificial Intelligence Framework*. Master's Thesis, Luleå University of Technology, 2016.

[6] C. Amin, B. Sousa, M. Arza, M. Bartesaghi, and T. Costa, "Cluster Multiple Spacecraft Per Aperture operations," in *SpaceOps 2016 Conference*, (Daejeon, Korea), American Institute of Aeronautics and Astronautics, May 2016.

[7] I. Clérigo, M. Bartesaghi, M. Bolland, and J. Volpp, "From VAX to iPhone: 20 Years of Cluster Mission Ground Segment Evolution.," in *62nd International Astronautical Congress*, p. 11, International Astronautical Federation, 2011.

[8] J. Volpp and D. Sieg, "The Cluster Mission after 13 Years - Operations beyond its Design Limits," in *SpaceOps 2014 Conference*, (Pasadena, CA), American Institute of Aeronautics and Astronautics, May 2014.

[9] Consultative Committee for Space Data Systems, "Mission Planning and Scheduling," Informational Report CCSDS 529.0-G-1, CCSDS, Washington, DC, USA, 2018.

[10] G. P. Calzolari, T. Beck, Y. Doat, M. Unal, H. Dreihahn, and M. Niezette, "From the EMS Concept to Operations: First Usage of Automated Planning and Scheduling at ESOC," in *SpaceOps 2008 Conference*, (Heidelberg, Germany), American Institute of Aeronautics and Astronautics, May 2008.

[11] M. Johnston, "Deep Space Network Scheduling Using Multi-Objective Optimization with Uncertainty," in *SpaceOps 2008 Conference*, (Heidelberg, Germany), American Institute of Aeronautics and Astronautics, May 2008.

[12] J. C. Pemberton, "Toward Scheduling Over-Constrained Remote-Sensing Satellites," in *Proceedings of the 2nd International Workshop on Planning and Scheduling for Space.*, p. 6, 2000.

[13] A. J. Vazquez Alvarez and R. S. Erwin, *An Introduction to Optimal Satellite Range Scheduling*, vol. 106 of *Springer Optimization and Its Applications*. Cham: Springer International Publishing, 2015.

[14] T.-P. Liang, "Recommendation systems for decision support: An editorial introduction," *Decision Support Systems*, vol. 45, pp. 385–386, June 2008.

[15] K. Ecker, J. N. Gupta, and G. Schmidt, "A framework for decision support systems for scheduling problems," *European Journal of Operational Research*, vol. 101, pp. 452–462, Sept. 1997.

[16] M. Robillard, R. Walker, and T. Zimmermann, "Recommendation Systems for Software Engineering," *IEEE Software*, vol. 27, pp. 80–86, July 2010.

[17] E. S. Berner and T. J. La Lande, "Overview of Clinical Decision Support Systems," in *Clinical Decision Support Systems: Theory and Practice* (E. S. Berner, ed.), Health Informatics, pp. 3–22, New York, NY: Springer New York, 2007.

[18] J. Bennett and S. Lanning, "The netflix prize," in *Proceedings of KDD Cup and Workshop*, Aug. 2007.

[19] D. Dasgupta and F. A. Gonzalez, "An Intelligent Decision Support System for Intrusion Detection and Response," in *Information Assurance in Computer Networks* (V. I. Gorodetski, V. A. Skormin, and L. J. Popyack, eds.), Lecture Notes in Computer Science, pp. 1–14, Springer Berlin Heidelberg, 2001.

[20] K. Fagerholt, "A computer-based decision support system for vessel fleet scheduling—experience and future research," *Decision Support Systems*, vol. 37, pp. 35–47, Apr. 2004.

[21] N. Faerber, K. Ojala, J. da Silva, A. Scholz, L. Brach, A. Demirsan, B. Sousa, and R. Boumghar, "A Grammar-based Timeline for Increasing Fleet Situational Awareness," in *15th International Conference on Space Operations*, (Marseille, France), American Institute of Aeronautics and Astronautics, May 2018.

[22] K. Ojala, *Optimization of Spacecraft Operator Workflow Through Modern Web Technologies.* Master's Thesis, University of Oulu, 2018.

[23] Angular, "One framework. Mobile & desktop.." https://angular.io/, Aug. 2019.

[24] D3.js, "Data-Driven Documents." https://d3js.org/, Aug. 2019.

[25] Django, "The Web framework for perfectionists with deadlines.." https://www.djangoproject.com/, Aug. 2019.

[26] L. de Moura and N. Bjørner, "Z3: An Efficient SMT Solver," in *Tools and Algorithms for the Construction and Analysis of Systems* (C. R. Ramakrishnan and J. Rehof, eds.), (Berlin, Heidelberg), pp. 337–340, Springer Berlin Heidelberg, 2008.