A STIGMERGY-BASED PARADIGM FOR MISSION PLANNING AND SCHEDULING OF MULTIPLE SPACECRAFT

C. Iacopino, P. Palmer, and N. Policella

1 Surrey Space Centre, University of Surrey, UNITED KINGDOM
2 European Space Operation Centre, ESOC, GERMANY

ABSTRACT

Missions involving multiple spacecraft, autonomously working together, have become of great interest in the last decade as they offer a number of scientific and engineering advantages. This trend is responsible for an increasing demand on mission planning & scheduling systems able to coordinate the different spacecraft and to allocate tasks amongst them. New approaches are therefore needed to handle this new level of complexity, combining together autonomous solutions for the ground and space segment.

In this paper we explore the potentiality of Self-Organizing Multi Agent Systems and Swarm Intelligence to develop a distributed coordination architecture based on the principle of Stigmergy. We start presenting a previously proposed architecture where Stigmergy has been used to achieve self-organizing coordination among a constellation of satellites [17]. We then introduce a new architecture where Stigmergy will play a key role in the decision process of each agent. Such a solution is based on the Polyagent paradigm proposed by Parunak and Brueckner [12] where advantages of reactive coordination are combined with optimization properties [1] similarly exploited by algorithms such as Ant Colony Optimization.

After outlining the general system, we show how this system can be applied to a specific case study designed considering the NigeriaSat-2 satellite (produced by Surrey Satellite Technology Limited - SSTL, due to be launched in June 2011). This satellite will be part of Disaster Monitoring Constellation - DMC. In order to demonstrate the planning & scheduling capabilities of our system, the case study focuses on the imaging campaign planning problem associated to NigeriaSat-2. We shall outline the extension to the whole DMC constellation to show the coordination benefits of our solution. The paper presents some preliminary results on the NigeriaSat-2 case study and proposes the future directions.

Key words: Distributed Mission; Swarm Intelligence; ACO; Polyagent; Imaging Campaign; Self-Organization; Emergence.

1. INTRODUCTION

As Clement [3] summarizes, missions involving multiple spacecraft offer a number of key scientific drivers over single monolithic spacecraft: signal separation (e.g., large synthetic apertures), signal space coverage (e.g., multi-point sensing) and signal combination (e.g., data fusion). Beside them, distributed missions bring unique advantages also on the engineering level: increase of reliability and extensibility, application of economies of scale and of new computational paradigm - concepts like grid, cloud computing, organic computing or self-organization present potentialities to be successfully applied to swarm satellites.

Mission Planning is an activity dealing with problems of high complexity that is usually faced with methods and techniques from the wide field of Automatic Planning & Scheduling. Due to the complexity and the computation power required, historically the ground segment has been responsible for the mission planning; few major examples are the SPIKE scheduling system [7], designed for the NASA’s Hubble Space Telescope and used since 1990; the autonomous planner ASPEN (Autonomous Scheduling and Planning Environment) [2] developed by the AI group at the NASA’s JPL, a framework able to solve a wide range of problems modelled as a set of resources, tasks and temporal constraints. A parallel work is the ESA APSI [16] that has been demonstrating the validity of AI techniques in several case studies. Clear advantages, as reducing operational costs and increasing the efficiency, are pushing the planning & scheduling activity on board of the spacecraft. Promising results have been achieved by missions such as the NASA DS-1 [11], the first mission functioned for a few days completely autonomously, NASA EO-1 with the re-planning system CASPER [8] or the more recent ESA demonstrator mission Proba-2 [10]. However solutions developed for individual spacecraft are not necessarily transplantable in distributed mission contexts. In essence, the distributed missions introduce a new element of complexity that must be faced in the mission planning: the coordination mechanism implemented between the spacecraft. Such a paradigm is strictly dependent on the inter-satellite link and the computation power that are critical resource constraints. Because of these limitations, finding the right level of sharing of responsibilities between ground and
space segment is one of the main challenges for future missions.

Nowadays work on collective partitioning of tasks amongst multiple spacecraft is still first steps. So no collective autonomous operations have been demonstrated for satellite clusters in space, for which is an active area of theoretical research. An example is the Shared Activity and Coordination model [55], where spacecraft modelled as agents negotiate with each other directly to partition the work among them. Projects like the control system D-SpaCpanS [52] and the NASA TechSat21 have shown instead how a hierarchical planning architecture is the best trade-off between excessive communications and excessive computation. However these are based on inter-satellite links either for direct negotiation between the spacecraft or just to share system state. Although distribution via clever and efficient protocols can be envisaged, in the perspective of future satellite platforms, the difficulty of maintaining inter-satellite links is likely to be prohibitive. Generally therefore most of the work that considers planning for multiple spacecraft operations lack of scalability (due to unrealistic intercommunication requirements) or would be practical for only a handful of spacecraft. This is the main motivation to investigate alternative communication paradigms like stigmergy, an indirect communication mechanism that use the environment as a means of communication. The following sections introduce the multi agent paradigm, a general framework for modelling distributed system, before focusing on the emergent properties that such systems can present using stigmergy. In particular we are showing two distinct solutions: an on-board coordination system for a cluster of satellites attempting to reduce the computational and communication overhead and a ground segment based planning & scheduling system aiming to be highly responsive to the user requests and scalable to support multiple spacecraft. This last solution is still in developing, the paper therefore focuses mainly on the design concepts and on the possible extensions.

2. MULTI AGENT SYSTEM AND SELF-ORGANIZATION

The majority of the work on planning for multiple spacecraft has adopted the multi agent paradigm to model the coordination and control aspects of such missions. Multi Agent Systems (MAS) is a relatively new field bringing together techniques and theories from multiple disciplines. When multiple agents coordinate together for a common purpose there are a number of different mechanisms that can be used. These approaches are strictly connected with the capabilities of agents that range across the spectrum from reactive to deliberative architecture.

Reactive coordination is a mechanism in which individual reactive agents have relatively simple rules of interaction, but the overall system behaviour that emerges shows complexity which can be utilized to solve global objectives. Deliberative approaches are characterised by explicitly planning the individual behaviours of all agents in advance. The planning can be more or less centralised and each agent has specific roles and objectives that match the overall system goal.

In essence therefore these two ends of the spectrum can be characterised as performing a task in a highly planned manner (deliberative), or just relying instead on an instantaneous spontaneous manner (reactive). The reactive approach is highly suited with problems with uncertainty. It is the most suitable for describing natural complex systems with high number of entities interacting with complex dynamics. Deliberative approaches are generally more efficient than reactive planning for more well understood problems. However they require large quantities of processing power to resolve all the rules built into the plans and produce in general more rigid solutions not able to face dynamic problems. The majority of work on autonomous systems for spacecraft, mentioned in the previous paragraph, focus on deliberative techniques exemplified by methods for deliberative planning and re-planning.

Discussions on reactive behaviours naturally lead on to the concept of self-organization and emergence. In a swarm consisting of a large number of entities, the result of combining simple behaviours at local level can end in an emergent complex behaviour at the system level able to achieve significant results. A system is called self-organizing when it shows structures or patterns at system level without a central or external authority. The concept of emergence is instead generally attributed to a system property that arises out of the multiplicity of relatively simple interactions and that cannot be reduced as a sum of such interactions. Self-organization and emergence are then concepts strictly interconnected though in some cases they appear as separate phenomena. As presented, self-organization and emergence are desirable characteristics that need to be imported in artificial systems that cope with high uncertainty and dynamic environments, like in space applications. The challenge in designing a self-organizing system is that there is no systematic way to formulate required micro-level behaviours given desired top-level macro behaviours. Researchers have been experimenting with several mechanisms leading to self-organisation and often at the same time to emergent phenomenon. The different approaches, as presented by Serugendo [15], can be divided in four categories: agent cooperation using negotiation paradigm, agent learning by means of reinforcement, simple direct interactions or indirect interactions like the stigmergy paradigm. This last one looks the most promising as it has been demonstrated achieving complex system behaviours.

3. STIGMERGY

The term stigmergy has been introduced in the 1950’s by the French biologist Grassé [6]. It comes from the Greek words stigma “sign” and “ergon” work indicating how the communication mechanism is based on traces left in the environment. This information stored in the environment forms a field that supports agent coordination stimulating their actions. Such techniques are common in biological distributed decentralized systems such as insect colonies where the information assumes usually the shape of pheromones. It is possible to identify two major
The decision process on the spacecraft is formed by

4. STIGMERGY-BASED ON BOARD COORDINATION SYSTEM

Sematectonic stigmergy can be efficiently applied to coordination systems for a cluster of satellites. This section describes a previously proposed architecture [17] aimed at reducing the computational and communication overhead and the task duplication.

It takes inspiration by the IPSS system [19]. Intelligent Production Plant Scheduling System, where the agents (responsible for individual pieces of machinery) propagate their intentions downstream while resource agents propagate load forecasts upstream. Crucially, the information is modified by intermediary nodes using only local and partial knowledge. The agents continually seek to optimise local resource usage. In essence, environment information flows up and down the production line with agents modifying it as it passes through. This two-way feedback and feedforward protocol provides a self-organizing system that continually evolves in front of the load and conditions on the line.

Similarly the architecture adopted foresees that the ground segment plays the only role of broadcasting all the tasks to the spacecraft with some additional high-level aggregated information, feedback/feedforward, previously received from the spacecraft. This broadcast provides a common environment that is modified by the single spacecraft. The spacecraft communicates to the ground segment the tasks executed, feedback, and the future intentions, feedforward. The ground segment is then in charge to aggregate and retransmit this information received from all the spacecraft. Fig. 1 shows a representation of the environment where the feedback, tasks already performed by other agents, and the feedforward components, tasks planned by the agent, covers the same resource space as the tasks.

Figure 1. Environment: tasks along with the feedback and feed-forward information.

The following sections will first describe a sematectonic inspired approach for on board coordination systems. We then introduce a new architecture where marker-based stigmergy is used at ground segment level both to bias the agents’ decision process aiming at achieving optimization and to achieve coordination among the agents.
two main components: the task prioritization block and the scheduler. The first one is in charge of associating weights, priorities to the tasks using a weighting function that describe the behaviour of the specific spacecraft. The scheduler instead is in charge of reordering the task, attempting to satisfy all the constraints, and scheduling the highest weighted tasks as early as possible. A spacecraft behaviour is modelled by a probability distribution over the resource space so that a higher probability means that the spacecraft is more likely to complete tasks with that associated resource value. This probability distribution is a Chebyshev polynomial sequence formed by four simpler distributions that define four main types of behaviours the spacecraft can have: greedy, considerate, proactive, obstinate. Each of them is described by a number of parameters optimized by a specific genetic algorithm developed for dynamic optimization. Lastly, a weighting parameter is assigned to each of these basic functions so that the spacecraft behaviour is really a hybrid in the general case. These parameters are the translation of the high level goals coming from the human operators. Fig. 2 shows the resulting behaviours of each agent in different colours in front of the global task map. The tasks selections of each agent are naturally contending with each other raising the problem of avoiding behavioural overlaps.

Concluding, the spacecraft bases its decision process on simple on-board selection strategies that require a much lower processing power than negotiation protocol and no inter-satellite links are required as well. Such a system using the broadcast is inherently scalable and resistant to failure, although the major disadvantage is the lack of guarantee on task completion or conflicts as spacecraft are not coordinating with each other directly. The results presented by this solution showed the feasibility of a stigmergy task coordination system and demonstrated surprisingly low levels (6%) of task duplication even with 18 spacecraft.

5. SOLUTION UNDER DEVELOPING

The main goal of the system we are currently developing is to extend the stigmergy principles to a new class of problems, driven by end users. The challenge is in giving the ability for a satellite constellation to respond to a number of users, making asynchronous requests, and having to schedule their tasks to respond in reasonable time. The complexity here is also on the ground segment that needs to handle the multiplicity of user needs. Further the system aims to be responsive to high level goals by the ground station managers and to generate imaging schedules that account for different priority levels and the needs of different user groups. The focus now is therefore on the ground segment. The system indeed is foreseen to run centrally on the ground segment abstracting from on-board processing and communication aspects among the satellites.

The coordination problem is handled with a paradigm similar to the previous solution that in the meantime has been extended and formalized by Parunak and Brueckner in the polyagent model [12], [1]. The name polyagent reflects the idea that the relevant domain entities can be represented by multiple agents with different responsibilities and capabilities. Considering the domain of satellite constellation, a spacecraft is going to be represented by a persistent agent also called avatar that is able to generate a swarm of stigmeric agents, ghosts, that explore large search spaces, interacting each other through digital pheromone, scalar variables that the agents deposit and sense in the environment. These ghosts agents are transient as they operate for a specific period of time or until a specific event occurs. In this way they serve only a particular function requested by the avatar that generated them.

This type of paradigm gives the possibility to combine self-organizing coordination techniques among swarms with classical reasoning approaches optionally supported by the avatar. Alternatively, for a physical distributed problem, like a satellite constellation, it is possible to have a second level of self-organization when the different avatars (spacecraft) coordinate each other using the same environment and digital pheromone used by the ghost agents. In this way coordination (avatar) and exploration (ghost) can effectively operate in synergy. Traditionally the topology of the space over which the ghosts explore is a representation of the geospatial aspects of the problem domain. Such an environment is not a passive element, it supports a number of functions relative to the pheromones like aggregation, diffusion and evaporation. It actually reflects the dynamic of the problem itself increasing drastically the reactivity of the system.

Considering one single avatar and a problem of planning & scheduling where we expect to optimize a cost function, this approach converges towards the Ant Colony Optimization (ACO) metaheuristic [4]. The term metaheuristic indicates a set of algorithmic concepts that form a general algorithm framework. The ACO metaheuristic is then the family of algorithms inspired by the ant foraging pattern, described in the previous sections. In such an algorithm a colony of artificial ants collaborates in finding a good solution to the discrete combina-

Figure 2. The behaviours of five homogeneous spacecraft on the global task map.
torial optimization problem. Good solutions are emergent properties of the agents’ interactions. ACO algorithms are constructive algorithms, as opposed to local search, they generate solutions from scratch adding solution components iteratively until completion. The single artificial ant is the computational element that builds such solutions. To use this technique, the problem needs to be represented as a graph where the nodes are the possible components of the solution and the edge expresses the choices that the ants have to perform. In general such a graph represents directly the feasibility space to help the optimization process. In an environment represented as a graph the pheromones can be located on the local entities (node or edge) and can be spread on the neighbourhood (nodes connected). The key point that allows optimization is that the ant exploring the graph select a move by applying a probabilistic decision rule that is function of the locally available pheromones and of a heuristic value that represent a priori information on the problem instance. Such value is related to the cost functions that need to be optimized; it can represent the cost or the benefit of that particular decision. The ant then in each location has from one side indication of the cost/benefit of that local decision and on the other side the recent history of the decisions of the previous ants represented as pheromones. In this model, the agents instead that adapting themselves during the exploration, thanks to the pheromone trail they modify how the problem is represented. Stigmergy is indeed a way to transfer cognition from the agents to the environment and to yield cognitively complex outcomes among relatively simple agents.

The polyagent model can therefore be considered an extension of the ACO metaheuristic to a multi agent platform where the focus is not anymore optimization but a more general collaboration between the agents. We believe then that the experience achieved by these techniques need to be applied together for solving complex problems like planning & scheduling for distributed missions. We decided to apply these concepts to the problem of imaging campaign planning & scheduling for the DMC constellation.

6. CASE STUDY: NIGERIASAT-2

The scenario considered is the Disaster Monitor Constellation (DMC) produced by SSTL. This platform is the first Earth observation constellation of low cost small satellites; it provides daily images for a wide range of applications, commercial or of public interest including disaster monitoring. Such system is currently composed of 5 satellites (Beijing-1, NigeriaSat-1, UK-DMC-1, UK-DMC-2, Deimos-1) plus 2 satellites (NigeriaSat-2 and NigeriaSat-X) that are going to be launched into the constellation by the end of 2011. This constellation offers multispectral imagery, wide swath (600km), 32m ground sample distance (GSD) and 4m panchromatic (PAN) resolution. The problem of imaging campaign planning & scheduling for this constellation rises because the number of requests and the typology of customers that such platform has to satisfy is quite varied and exceeds the capabilities of the whole system.

6.1. NigeriaSat-2

We decided to approach the problem considering one satellite of the DMC constellation: the new NigeriaSat-2 (N-2). This spacecraft is an agile mini-satellite of 300 kg flying at 700km of altitude with a period of about 1.7h. It allows high resolution imaging at 2.5m PAN and 5.0m GSD 4-band multi-spectral with 20km swath width. Further it performs wide area mapping at a resolution of 32m GSD 4 band with a 300km swath width. Thanks to its agility it has 45°roll/pitch off-pointing for high resolution spot imaging, stereo mode imaging and area mode imaging for artificially increasing the swath width.

The mission has to satisfy a number of customers that request images of specific targets within certain time windows. Because of the limited memory on-board, time constraints between requests and limited number of downlink passes, it is required to determine a subset of such requests that satisfy all the constraints and maximize certain performance metrics. Fig. 3 is an example of the satellite swath in one day. Notice that on average the satellite will take 3 days to cover the whole Earth. Some of the targets then can only be imaged once every 3 days.

6.2. Problem modelling

The first step applying the concepts presented in the previous section is to transform the problem in a graph-like environment that ant-like agents could explore. A natural way to build a solution responsive to the user requests is to use such requests as main environment component. They become therefore the nodes of the graph. The edges instead represent the time constraints between them. The result is a directed graph where the direction of the edges reflects the real time direction and the order between the requests reflect their position on the globe within the swath. The temporal conflicts between requests that within the swath are parallel or are too close to each other are represented with bifurcations to force a choice in the agents exploring the graph. To complete the representation, the ground station passes need to be included in the graph as, though they have different functions, they have static temporal and physical attributes easily mapped in the graph. All the ground station passes allow commands uplink but not all of them allow download of data. Fig. 4 shows a simple representation of this environment where the squares represent the imaging request and the triangle the ground station passes.

![Figure 4. Schematic representation of the environment.](image)

Each of the entities is characterized by a number of
attributes: for the imaging request the memory needed and the quality that indicates the importance of the specific request, while for the downlink pass only the memory downloaded is needed. Using a terminology familiar in the planning & scheduling field, this problem is a knapsack problem with scheduling constraints that can be modelled as a binary reusable resource: the camera, strictly dependent on a depletable resource: the memory. The imaging request is an activity that consumes memory while locking on the camera. The downlink pass instead is an activity that produces memory. The constraints considered then are the memory available that is a limited resource and access to the camera that is exclusive. Incorporating all the temporal constraints in the environment, the graph becomes the feasibility space. Inconsistencies on the memory utilization instead must be checked by the agents themselves. Lastly the environment implements the typical set of functions for the pheromone communication: aggregation, diffusion and evaporation. The pheromone is a scalar quantity indicating the past decision of the ant agents.

6.3. Architecture

Taking inspiration by the polyagent architecture, the system can be modelled in 3 main categories of agents: environment, real and virtual agents. Environment agents are the imaging requests, downlink and uplink passes; as mentioned in the previous paragraph they are the nodes of the environment. The real agents reflect physical agents like the spacecraft. They are able to create virtual agents, ant agents, the equivalent of the ghost agents in the polyagent paradigm. The ant agents are in charge of exploring the environment and finding a solution for the planning problem. They are small computational entities that communicate among them using stigmergy. Tab. 1 summarizes this classification.

All this defined, the main algorithm is quite simple. Any time the spacecraft needs to calculate the solution, it creates an ant cycle, during which a swarm of ants explore the graph. At the end of it, the spacecraft determines the plan from the pheromone distribution created during the ant cycle and acts deterministically in front of it: the imaging requests with high level of pheromone are going to be included in the plan. The algorithm that is behind the behaviour of the ant agents can be seen as a proper ACO algorithm. This graph takes care of the temporal constraints, leaving the assignment problem (a knapsack problem) where the resource constrained is the memory. In literature there are a number of ACO algorithms that have been successfully applied to the knapsack problem [9], [5]. However in this paper we propose a little variant to the traditional approach. These algorithms operate on a quasi-fully connected graph where at each step the ant can choose among all the elements not assigned. The solution is a specific path in the graph. In our approach the ant has to respect the order of the requests as positioned in the ground track. At each step if the ant decides to assign the request to the plan drops a positive pheromone otherwise it drops a negative pheromone. This means that the solution at the end of the ant cycle is not an entire path of the graph but a subset of it. This choice eases changes of the graph at runtime increasing the flexibility of the system for dynamic planning.

6.4. One day of planning

The solution proposed has been tested considering a one day planning horizon. Such time period corresponds to the short-term planning operated in DMCI. As at the moment of writing this paper, N-2 is still due to be launched, the data used comes from the mission planning system for the spacecraft UK-DMC-2 that shares similar characteristics with N-2.

Tab. 2 and Tab. 3 show respectively the list of imag-
### Table 1. Classification of the system's agents.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Type</th>
<th>Attributes</th>
<th>Time Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent Id</td>
<td>Static</td>
<td>Description</td>
<td>Id</td>
<td>Static</td>
</tr>
<tr>
<td>Quality</td>
<td>Memory Required</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Uplink</td>
<td>Env. Agent</td>
<td>Commands Upload</td>
<td>Memory</td>
<td>Static</td>
</tr>
<tr>
<td>Downlink</td>
<td>Virtual Agent</td>
<td>Commands Upload</td>
<td>Memory</td>
<td>Static</td>
</tr>
<tr>
<td>Data</td>
<td>Physical Agent</td>
<td>Data Download</td>
<td>Memory</td>
<td>Static</td>
</tr>
<tr>
<td>Spacecraft</td>
<td>Real Agent</td>
<td>Memory Av.</td>
<td>Virtual Time</td>
<td>Real Time</td>
</tr>
</tbody>
</table>

Table 2. List of imaging requests for one day.

- Imaging requests (AOI) that can potentially be performed in one specific day and the ground station passes for the same day. It is possible to note that the amount of memory downloaded for each pass is variable and is never more than 3 GB. Considering that the whole memory onboard is 8 GB, it is clear that the planning problem goes far beyond the next pass.

- At the moment the algorithm does not consider different constraints in front of different imaging mode. As safety condition, a period of 60s between two consecutive imaging requests is required. If such a period is not available the system will be forced to choose one of the two conflicting requests by means of a bifurcation on the graph. This type of graph has the advantage of keeping a compact representation without branching exponentially.

Concerning the specific details in the ant agents’ behaviour, a number of different ACO techniques has been tested to define the decision rule, the pheromone update rule and the local heuristic. The actual implementation sees the AS decision rule [4] and a constant global pheromone update only for the best ant found from the beginning of the run. The local heuristic information for the imaging request is given by a parametric exponential ratio between the attributes quality and memory [5].

The ant colony converges to the best solution in the 96% of the case by the first 50 ant cycles though it keeps oscillating in the following cycles. The wide research available on ACO algorithms is one of the advantages that motivate this design methodology. However no formal methods are available to evaluate the impact of these rules on the global dynamic. Simulation alone is not sufficient, in particular for a careful tuning of the parameters. These types of solutions are indeed strongly sensitive to the parameters. We are currently investigating formal methods that could be used side by side with simulation-based design and evolutionary design. A detailed analysis and comparison of the algorithm performances is therefore postponed to later works.

7. CONCLUSION AND FUTURE EXTENSIONS

The algorithm presented in the previous section is designed to solve a quite simple problem. It is the output of preliminary studies. Our focus is in fact to propose a new approach and a new methodology for solving real planning & scheduling applications taking advantage of synergies in self-organizing technology like swarm intelligence. The approach presented becomes indeed particularly promising looking at the possible extensions for more significant problems.

Dynamic Planning - Problems like imaging campaign planning are inherently dynamic as they deal with a wide user community with different requirements. In case of natural disaster the timeliness in acquiring significant data becomes a major priority. In our approach, it
is extremely simple and low computational expensive to make such changes in the graph environment, as the impact on the overall system is very limited. Further a system responsive to high level goals defined by operation managers is extremely desirable. Such goals can be translated in different pheromone flavours able to strengthen or inhibit specific areas of the environment. The ACO algorithm shall run in correspondence of each ground station pass that corresponds to the possibility to upload the new plan.

**Multiple spacecraft** - The most immediate extension for the system presented is the scenario with multiple spacecraft such as the whole DMC constellation. Depending on the service level agreements, the users of the DMC community are allocated to one specific satellite or more. Translating these requirements to our graph environment means building a graph for each satellite and letting them intersecting only on the imaging requests shared. Taking inspiration from the distributed coordination systems in manufacturing environments [18], we can use different pheromone flavours for different satellites. The pheromone of one satellite will be able to inhibit the actions of the others spacecraft avoiding conflicts on the shared imaging requests.

**Ground Station Planning** - In the scenarios presented so far the number of ground station passes and the duration of each of them are determined by separate systems. A significant problem in the space community is the ground station planning as well. The traditional approach whereby missions demand specific activities at specific times results in a higher likelihood of conflict. In our problem modelling a ground station pass that is conflicting between two missions can be shared between the graphs of these missions and treated in the same way of an imaging request. Further a cost attribute added to the ground station pass, could help in optimizing the number of passes requested for the specific load of the spacecraft.

**ACKNOWLEDGMENTS**

The data used for the problem instance has been provided by DMCii responsible for the imaging campaign planning. This work is co-funded by the Surrey Space Centre, SSTL and the Operations Centre of ESA (ESOC).

**REFERENCES**


