Automated Planning and Scheduling EO Constellations' Operations with Ant Colony Optimization

Evridiki Vasileia Ntagiou
Surrey Space Centre, University of Surrey

Abstract

In this work we are interested in Automating the process of Planning and Scheduling the operations of an Earth Observation constellation. To this respect, we represent the problem with a directed graph and use Ant Colony Optimization technique to find the optimal solution. In order to verify the quality of the solution, we employ a dynamical system. We check the scalability of the software system performing simulations. We discuss the next steps of this work which involve the coordination of multiple spacecraft by means of stigmergy and the consideration of more than one objectives that need to be optimized.

Motivation and Scope

The increasing interest in the design and development of space missions consisting of multiple coordinated spacecraft cannot be missed, in recent years. Ranging from low cost due to less system reliability requirements, to giving man the ability to perform concurrent scientific observations, the advantages of using constellations of spacecraft have attracted the complete attention of the Space community [T. A. Wagner et al.]. The Earth Observation market, in specific, is expected to grow at a rate of 16% per year over the next decade [N. Muscettola et al.]. The current trend is towards constellations consisting of many small satellites, with an increasing number of start-up companies aiming at launching such constellations of 20 hundreds or more mini-satellites. [G. Richardson et al.][E. Buchen]

The reduction of the satellites' size and corresponding shrinking of their cost has allowed many end users to benefit from data coming from satellites. Since we are dealing with the cooperation of numerous miniaturized satellites of simple capabilities, which altogether form a very complex system, the need to automate its management arises. Traditional techniques have failed to cope with such a level of complexity. *Planning and scheduling* (P&S) the operations of an EO satellite is the process of determining the time when the satellite performs specific arranged tasks, as the available resources, images' collection goals, weather condition and user requirements evolve. More specifically, the P& S system is responsible for coordinating a constellation's satellites' activities in order for the total value of the downlinked data to be maximized. The Earth observing satellites (EOSs) picture the Earth's surface, in order to satisfy an assigned goal, which in our case will be the imaging of the *Area of Interest* (AoI). EOSs can acquire images, while moving on their usually low altitude orbits. The acquired data will then need to be transmitted to the ground station. Until that is possible, the data are stored in the limited on-board memory of the satellites, limiting the images that can be acquired before the downlink.

There is a wide interest for automating the P&S process in the EO field, emanating not only from research organizations and universities [C. Iacopino et al], but also from commercial operators and agencies [S. A. Chien et al.]. The main benefit of autonomy in the planning & scheduling field is in being able to gain maximum value from the spacecraft by maximizing the use of on board resources and providing a greater level of responsiveness to sudden changes of priority, such as when natural disasters strike. Automating the P&S process of an Earth Observation mission involves *optimization* and *coordination*. It is a combinatorial optimization problem that takes place in an uncertain dynamic environment. The development of an automated P & S system also follows the needs of the upcoming missions. These employ dozens of agile satellites, where a change of attitude translates to a tilt of the imager. We consider agile EOSs that can be steered up to 45° off-nadir in the roll axis.

An EO mission may have a single goal e.g. maximize the imaged area, and many constraints, e.g. resource or weather constraints. It could also have multiple goals which are conflicting e.g. maximize the imaged area, while minimizing the resource used, and again numerous constraints. In fact, the nature of the problem is such that it includes many constraints, when realistic scenarios are studied. In most of the studies, a single-objective optimization problem with numerous constraints is considered. This alone, means that our solution will be valid under several assumptions. In order to lift those assumptions we try to decrease the number of constraints and increase the number of goals. In this case, the P&S problem is a multi-objective optimization problem. In order for a mission to be successful, the trade-off among the several objectives needs to be studied and a solution depending on the user requirements needs to be produced.

The main challenges that arise when developing a software system that is meant to be autonomous can be grouped in three main categories: Reliability, Scalability and Adaptability. When dealing with a continuously changing environment like space, a system must be able to quickly adapt to new circumstances and adjust its output correspondingly, not allowing the changes to interfere with the quality of the solution. The case is the same, when the users' preferences or the platform's characteristics change, e.g. increase in the dimensions of the Area of Interest for an EO satellite, or increase in the number of satellites available for a task in a constellation. In order to address these challenges we propose a *self-organising* architecture for the software system and a *dynamical system*, by which we can analytically study the optimization method and guarantee the convergence to a solution. Furthermore, the method we employ can easily be extended to a study of the multi-objective nature of the problem.

Novelty

This Ph.D. seeks to make contributions in three areas:

1. Development of a Self – organizing software tool as an Automated Mission Planning System. In our novel approach, we will employ a multi-agent system to manage the coordination among the constellation's spacecraft.
2. Modelling of a Probabilistic optimization technique with a nonlinear dynamical system. We formally verify the reliability of our algorithm employing a non-linear dynamical system to model its behaviour.
3. Multi objective Optimization techniques using Swarm Intelligence methods. Ant Colony Optimization technique has not been widely employed for multiple objectives optimization purposes.

Problem definition

We consider a large area of polygon shape in the surface of the Earth that we are interested in imaging, within a specific time window. What is the best way to cover the Area of Interest (AoI) with the satellites' swaths? This is a *coverage planning* problem. *Optimally* planning the images' acquisitions and assigning them to the satellites of a constellation is a combinatorial optimization problem. In order to quantify the level of optimality, we need to introduce an objective function.

This problem is of highly dynamic nature. This is due to the constant changes regarding the user requests, the weather conditions, e.t.c. Hence, the *challenge* is to solve this problem in a way that these continuous environment changes do not interfere with the quality of the solution. In other words, we want a Planning and Scheduling system that is *adaptable* to the changes of the environment, while preserving its *efficiency*.

Problem Representation

In our research we assume that the satellites are agile. In other words, when passing over an area that we desire to image, a satellite has many options to choose from, regarding the angle in which the imager will tilt to capture an image. We represent the problem with a directed graph, which will form the common environment the ants will traverse and update to find a solution. In the graph:

- Each *Node* represents an *pass over the AoI*
- Each *Edge* represents a *roll angle* the imager can be tilt.

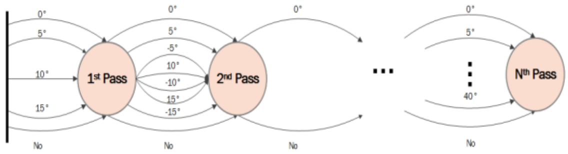
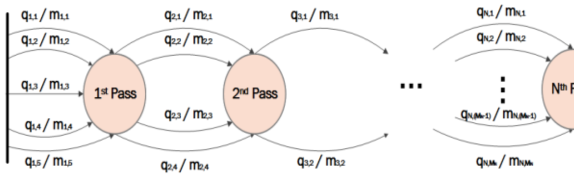


Figure 1.1. Problem representation using a directed graph.

The nodes are put in a chronological order, as the orbit of the satellite dictates. Each node is now connected to the next one with an arbitrary number of edges, each one representing an option of angle the imager will be tilt, and one representing the option of not taking a picture in this pass. Hence, each *path* starting from the first node, until the last one represents a *sequence of choices* of angles in which the imager should be tilt for each pass or a *schedule* that the satellite follows to complete the task of imaging the AoI. In order to quantify the differences among the strips available to be imaged, *quality values* (q_{ij}) are assigned to the edges, as functions of the area that they cover and the distortion of the image. Each strip is assumed to have a different quality and consume a certain amount of memory (m_{ij}). We assume to have a limited total on-board memory, M, that is only renewed when passing over a Ground Station. Formulating the problem, we assume the directed graph $G = (V,E)$, where

- V = set of Nodes
- $E = \{E_1,E_2,..,E_N\}$ and E_i = set of incoming edges in Node i.



Our goal is to find the path that visits all the Nodes and maximizes the objective function:

$$f = \sum_{i=1}^{N} q_{ij},$$

$$j = \{edges\ that\ belong\ in\ a\ selected\ Path\}$$

Under the memory constraints:

$$\sum_{i=1}^{N} m_{ij} \leq M,$$

$$M = total\ on\ board\ memory$$

Ant Colony Optimization

Ant Colony Optimization (ACO) meta-heuristic is a *probabilistic* algorithm used to find the solution in Computer Science and Operations fields' problems that can be reduced to finding optimal paths in graphs [M. Dorigo et al.] The method is summarized below:

When the ants are searching for food, in the natural world, they first wander randomly. After finding a source of food, they return to their colony but lay down pheromone trails in the path that they follow. If other ants while also looking for food find such a trail, they will probably not continue their wandering, but follow the trail instead. In case it leads them to food, they will also reinforce it when they return to their colony. The pheromone trails, however, start to evaporate over time. Hence, the longer it takes for an ant to travel back to the colony through the path it chose, the more the pheromones will evaporate. Hence, with this mechanism the amount of pheromone will become higher on the shorter paths than the longer ones, since a short path will get marched over more frequently.

The pheromone deposition helps the colony converge to an optimal solution. On the other hand, the pheromone evaporation is a means of helping the colony avoid convergence to a *locally* optimal solution. Were there no evaporation rate of the pheromone, the following ants would be more likely to choose the paths chosen by the first ones. This fact shows the importance of exploring for a sufficiently long time period and then converge to a solution. This way, the technique has higher chances of being successful.

In our case, we model the problem as a graph, and search for the path that optimizes a specific objective function. Artificial ants will run through the graph and find the desired path. Since this behaviour is inspired by nature and real ants, the artificial ants will they lay pheromone on the edges of the graph and they choose their next step with respect to a probabilistic function of the previously laid pheromone, by ants that have already traversed the graph.

$$P(e_i) = \frac{g(pheromone\ in\ e_i)}{g(total\ amount\ of\ pheromone)}$$

where $P(e_i)$ is the probability of choosing edge i, (e_i), and g is a function of the pheromone amount.

This is an indirect way of *communication* that aims at enhancing the environment (graph) with information about the quality of path components. This mechanism will lead following ants to the shortest path. The amount of pheromone an ant deposits depends on the quality of the path, which is evaluated by the objective function. This is a way to give feedback of the quality of the path an ant constructed.

ACO verification using a dynamical system

The ACO behaviour has been modelled using Ordinary Differential Equations previously, by Gutjahr [W. J. Gutjahr]. He studied the convergence speed of a number of problem representations using ODE. However, analyses are usually directed to specific algorithms, including no stability analysis. In our case, we aim at identifying the conditions of convergence. In order to apply the ACO technique to our problem, we need to understand and describe the dynamics of the long-term behaviours of the ACO algorithm. This translates to the study of under which conditions the ACO technique has the property of convergence in a solution. Hence, we are interested in understanding which is the long-term behaviour that characterises the system. We can expect numerous possible pheromone distributions, but we have a solution to our optimization problem when the system converges to one path. In this section we will present the analytical model, for the basic problem size of *1* node and M incoming edges. This translates to a choice among M roll angles in a single pass. In this way, we can easily show the basic structure of the system's dynamics by looking into its phase portrait, and thus have a deeper understanding of how the system will behave when more nodes are added. The $M-$dimensional dynamical system is:

$$\dot{\tau_1} = -\rho\tau_1 + kP_1$$
$$\dot{\tau_2} = -\rho\tau_2 + kP_2$$
$$\vdots$$
$$\dot{\tau_M} = -\rho\tau_M + kP_M$$

Where:

- τ_i = amount of pheromone in edge i
- ρ = pheromone evaporation rate
- k = amount of pheromone deposited
- $Pi = \frac{\tau_i^\alpha}{\sum \tau_j^\alpha}$, probabilistic rule based on which an edge will be chosen by an ant

We study the stability of the system using Nonlinear Dynamical Systems Theory. First, we calculate the number of equilibrium points it has, their analytical form and then define the stability of each. The results are summarized in the following Table.

	α : [0, 1)	**α= 1**	**α : (1, ∞]**
Exploration	High	Medium	Low
Convergence	Low	Medium	High

By the study of this system, we concluded that there is a critical parameter that controls the stability of the system, which is α. In the table we identify three main system's behaviors and highlight that none of them is perfect in terms of optimization. Thanks to these insight, a novel algorithm was developed in [C. Iacopino] that *combines* these behaviors to exploit their benefits. This algorithm is capable of regulating the trade-off of exploration vs convergence by oscillating alpha between the two areas.

Self – organising software system

In order to test the algorithm, a self contained component of software was designed, which wraps the entire system. It is written in Java and incorporates a fully open-source discrete-event agent-based modelling framework called MASON. The system's parameters and the objective function are configurable. The planning problems are passed in input as lists of imaging opportunities with their quality values and consumed memory while the output is a list of solutions containing the set of planned tasks. Each entity of the problem is assumed to be one agent. They are divided in three main categories: Environment agents representing the graph, Ant agents representing the computational units that update the graph and check the memory constraints and a Master agent that checks the convergence of the colony and the evaluation of each path produced.

We are considering the case of a single spacecraft. The planning problem is represented in a graph in each test, and the ant agents find the best sequence of actions.

Simulations results

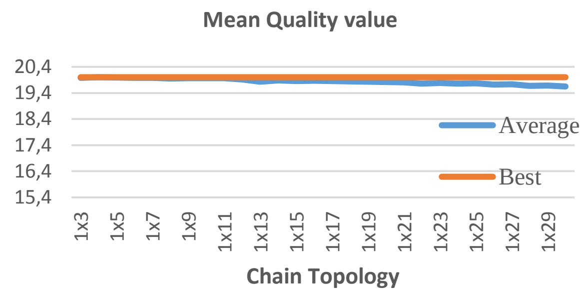
We test the *efficiency* and the *scalability* of the system. The metrics to quantify the performance of the system in these two fields will be:

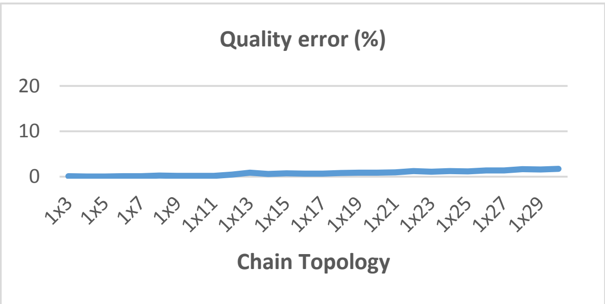
- **Quality value**. We compare the system's solution to the one given by a deterministic algorithm, performing an exhaustive search, and compute the error.
- **Convergence Time**. We measure the number of ants it takes to converge to a solution for different graph topologies.
- **Computation Time**. We check the change of the computation time, when increasing the dimensions of the problem.

We perform two types of tests. First we assume a single Node and increase the number of incoming Edges. This corresponds to increasing the number of roll angle choices in one pass. Next, we fix the number of incoming Edges to 3 and increase the number of Nodes. This corresponds to having 3 roll angles to choose from in each pass, but increasing the number of passes. We note that in this type of test, each time we add a Node, we triple the search space.

Efficiency tests

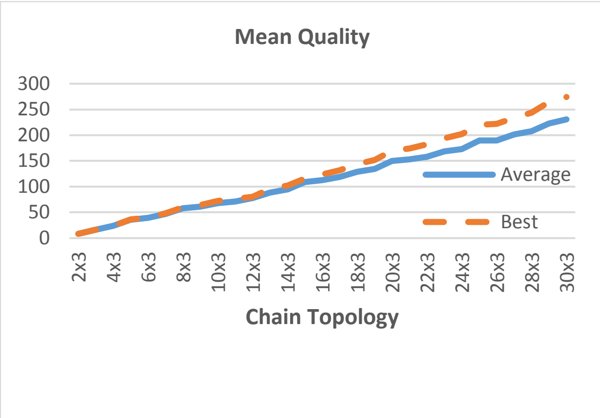
Increasing the Edges

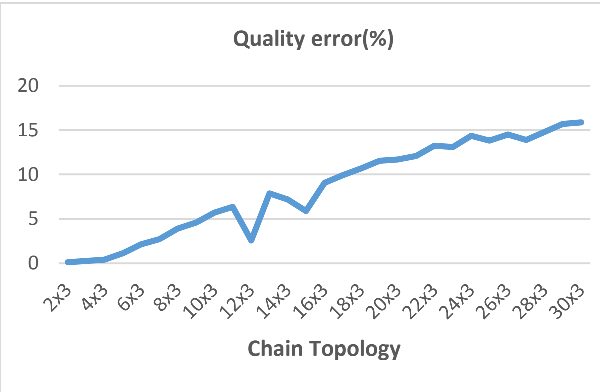




The above results show that the system is highly efficient to the increase of Edges in the graph. The error between the system's output and the best solution is up to *1.75%* with respect to the best, when having 30 Edges, or 30 roll angle choices.

Increasing the Nodes

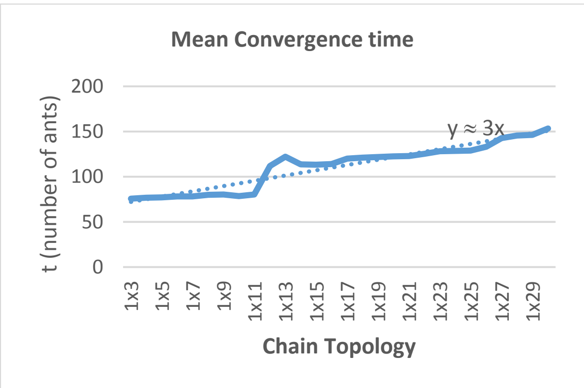


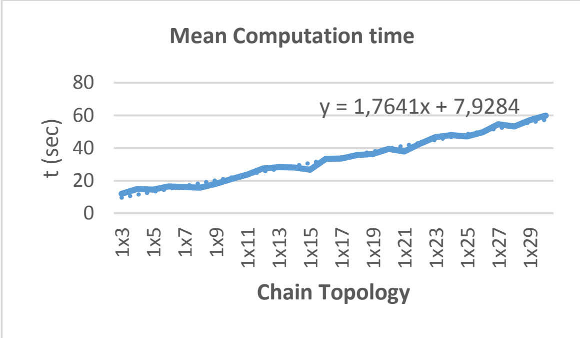


When increasing the number of Nodes, the system becomes less efficient, with the error being up to 15.8%. That is due to the fact that each Node addition results in a triple search space.

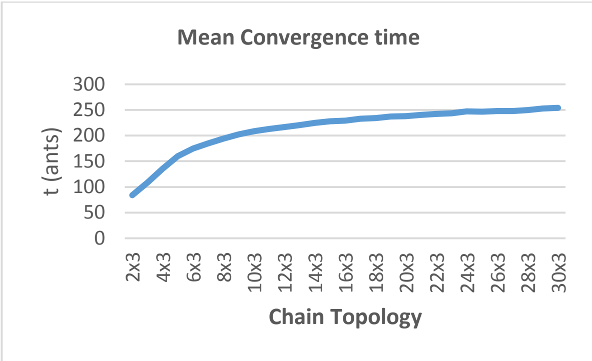
Scalability tests

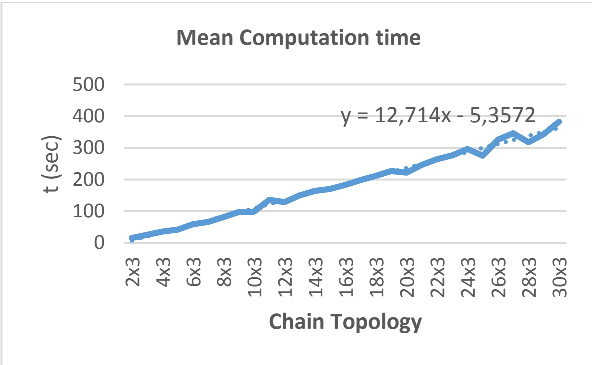
Increasing the Edges





Increasing the Nodes





The Scalability tests, for both types of tests are very encouraging. The system's convergence time increases either linearly, or logarithmically. Also, the computation time is increasing linearly, making the system *very scalable*.

Future work

SSTL Case Study

The first case study we will consider in this research is the Disaster Monitor Constellation (DMC3) produced by Surrey Satellite Technology Ltd. It is an Earth Observation mission which was launched in July 2015 and is currently in commissioning phase. The platform consists of 3 agile Earth Observation satellites at 1 m resolution.

They can change their attitude up to 45° off-nadir pointing in pitch and roll axes. 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. This constellation offers multispectral imagery, wide swath (600km), 32m ground sample distance (GSD) and 4m panchromatic (PAN) resolution. Currently SSTL are given requests to image certain areas of the globe, and their operators manually determine how best to achieve this. We aim at using realistic data from this mission in order to test the tool that we designed. Furthermore, our goal is to integrate our method as one of SSTL's Mission Planning Systems.

ESA Case Study

During the current Ph.D. we aim at integrating the single and multiple objectives methods, in ESA's missions that employ agile constellations. A great Case Study would be the European Data Relay System (EDRS). It is a planned European constellation of state of the art GEO satellites that will relay information and data between satellites, spacecraft, UAVs, and ground stations. Given the complexity of the system, the scheduling of these activities would certainly be better performed if more than one objectives were able to be modelled and optimized. The trade-off among such objectives would definitely give an insight on the management of a system with such high level of complexity.

Multi Objective Optimization

In the *Motivation and Scope* section we stated the need for employing Multi-Objective optimization techniques to design MPS that are more adaptable and applied to a wider range of missions. We now need to understand the system's characteristics that make one method more efficient than another, in order to decide which one we are going to include in our research. The visualization of the Pareto front seems like a more desirable way to solve the problem. It allows for the trade-off between each of the objectives to appear. The problem representation that we have employed allows for many additions to the algorithm. For example, having two values of quality per edge, one for each of the two objectives we have, can be very easily integrated and will result in the Pareto front visualization, with the use of just a little more system memory. Nevertheless, this is a decision we still have to make, comparing all the advantages and disadvantages that each method carries.

Coordination Mechanism

When it comes to having multiple spacecraft collaborating to achieve a task, without communicating with each other, or having an external central controller, the coordination needs to take place by means of stigmergy. The spacecraft will share a common environment, the graph. All the possible strips now need to be represented in the graph, in order for the ants to find the path that optimizes the shared objective function. The cooperation needs to take place using the pheromone trails in the environment that all the satellites will share.

Acknowledgments

This work is co-funded by the Surrey Space Centre (SSC) of the University of Surrey, the Surrey Satellite Technology Ltd (SSTL) and the Operations Centre of the European Space Agency (ESA/ESOC).

References

N. Muscettola, P. Nayak, B. Pell, B. C. Williams, 1998. *Remote Agent: to boldly go where no AI system has gone before*, Artificial Intelligence v103, # 1-2, pp. 5–47.

T. A. Wagner et al., 2004. *An Application Science for Multi – Agent systems.*

G. Richardson, Dr K. Schmitt, M. Covert, C. Rogers, 2015. *Small Satellite Trends 2009-2013*, 29th Annual AIAA/USU Conference in Small Satellites.

E. Buchen, 2015. *Small Satellite Market Observations*, 29th Annual AIAA/USU Conference in Small Satellites.

S. A. Chien, M. Johnston, J. Frank, M. Giuliano, A. Kavelaars, C. Lenzen and N. Policella, 2012. *A generalized timeline representation, services, and interface for automating space mission operations*. In Proceedings of the 12th International Conference on Space Operations, SpaceOps.

C. Iacopino, P. Palmer, N. Policella, and A. Donati, 2014. *Planning the GENSO Ground Station Network via an Ant Colony-based approach*, In The 13th International Conference on Space Operations,
SpaceOps 2014. Pasadena, California.

M. Dorigo, T. Stuzle, 2004. *Ant Colony Optimization*, MIT Press.

W. J. Gutjahr, 2008. *First steps to the runtime complexity analysis of ant colony optimization*. Computers & Operations Research, 35(9): 2711–2727.

C. Iacopino, 2013. *Automated Planning & Scheduling for Earth Observation Constellations: an Ant Colony Approach*, PhD Thesis, University of Surrey.