DOI: 10.13009/EUCASS2023-994
Aerospace Europe Conference 2023 – 10TH EUCASS – 9TH CEAS

A Concurrent Methodology for Optimizing Constellation Deployment and Launcher Selection

Brandon Israel Escamilla Estrada∗†, Giuseppe Di Pasquale*∗§,
Daniel González-Arribas∗, Manuel Sanjurjo-Rivo§ and Daniel Pérez Grande∗
∗*IENAI SPACE - Av. Gregorio Peces Barba, 1, Leganes, 28919, Spain*
§*Universidad Carlos III de Madrid - Av. de la Universidad, 30, Leganes, 28919, Spain*
brandon.escamilla@ienai.space · giuseppe.dipasquale@ienai.space
†Corresponding author

Abstract
The surge in small satellite constellations is spurred by the recent cost reductions in launching capabilities. This has facilitated the development of affordable constellations in Low Earth Orbit (LEO). Commercial space components are revolutionizing LEO systems, with a particular focus on electric propulsion. Deployment strategies and space mobility are also gaining significant attention. To ensure a successful constellation deployment, careful consideration should be given to factors such as launcher selection. A proposed concurrent engineering methodology utilizes optimization techniques to identify optimal combinations of launchers, propulsion systems, and deployment strategies. The methodology is demonstrated through a constellation deployment scenario, showcasing its effectiveness.

1. Introduction

Satellite constellations enable numerous applications, including high-speed internet access in remote areas, global telecommunication, Internet-of-Things (IoT), weather forecasting, disaster response, and many others [12]. These distributed systems are every day more reliant on small satellites [3]. Compared to heritage monolithic large satellites, this enables improved coverage and revisit time, while being comparable in terms of cost, thanks to the balance between performance and risk. The main drivers for cost-saving are the use of Commercial-Off-The-Shelf (COTS) components, and the decreased satellite mass, which lowers the cost-to-space which is directly proportional to the mass of the system. However, to achieve improved coverage performance and enable internet communication, the size of the constellations has risen exponentially, giving birth to mega-constellations. An exemplary case is SpaceX's internet Starlink constellation, which has currently launched more than 4300 satellites [14], although the company expects more than double for the complete constellation. Other operators are following a similar trend, such as OneWeb [2], which has launched more than 400 satellites, or Amazon's Kuiper constellation [9, 12], planned to have more than 3000 satellites in its final form. Although the mass of the satellites is quite low, the increase in the number of satellites in the constellation can bring the launch cost significantly up. In fact, the number of orbital planes that need to be covered can reach values north of 50: Starlink uses 72 orbital planes, for example. Therefore, strategies to reduce the number of launches must be considered in order to guarantee the feasibility and economic sustainability of these new systems. In addition, the selection of the best launch opportunity, which is highly coupled with the deployment strategy selection, is critical. This paper proposes a method to explore concurrently the launcher selection and in-orbit maneuvering strategy using a multi-objective optimization approach, in order to make the method suitable for early design phases. The method proposed leverages a flexible Multi-Disciplinary Optimization (MDO) architecture, which could be expanded to include additional disciplines. In fact, designing a satellite constellation is a highly constrained, multidisciplinary problem that requires iterative methodologies due to the interdependencies between the various domains. The optimization of each discipline has traditionally been performed independently, however, this siloed approach can lead to sub-optimal overall performance and increased costs. To address this issue, a systematic and concurrent approach is needed, considering all the relevant disciplines in an integrated design process. In addition to the common disciplines used for this kind of problem, such as configuration and orbit design, spacecraft design, optimization of common objectives (Earth Observation (EO), internet, and IoT services [1]) of the mission, it is also important to consider other factors that can impact the overall cost of the constellation. For instance, the selection of appropriate launcher opportunities, a comprehensive trade-off of maneuver strategies, and the manufacturing and total system costs are critical decisions that can affect the overall success of the mission. Therefore, a comprehensive

DOI: 10.13009/EUCASS2023-994

and integrated approach to satellite constellation design is necessary to achieve the desired performance and cost-effectiveness of any constellation mission. The relevant disciplines include constellation deployment strategies, orbit configuration, maneuver strategies, manufacturing, and system cost estimation. By evaluating each discipline and making trade-offs between conflicting objectives, it is possible to design and deploy a successful satellite constellation that meets the desired mission goals and constraints. Thus, the following provides a comprehensive review of each discipline involved in satellite constellation design, including a brief overview of the literature. Budianto and Old [4] analyzed a Low-Earth Orbit (LEO) constellation to acquire missiles' state vector data throughout their trajectory. In the paper, they include a configuration and orbit design module, with a simplified dynamics model for coverage computation, focusing on the impact of the payload on the mission. The optimization aimed at finding the minimum cost. The launch manifest is included as an Integer Programming (IP) problem, considering the launch cadence of the various launch vehicles, whose cost is averaged from various publications. In their work, they assumed that no more than eight of each type of launcher to be available to deploy the constellation within the given deployment schedule. Savitri *et al.* [15] proposed an approach based on the objective of designing a small satellite constellation with limited coverage. A Genetic Algorithm (GA) was used to optimize five out of six orbital elements. The multi-objective optimization study aimed at maximizing percent coverage and minimizing revisit time. Cornara *et al.* [5] presented a review of different commercial launchers, compiling databases for launch sites, launchers, and fairings. They proposed a methodology to deploy constellations using an indirect injection approach, which is useful when the launcher selected to perform the orbit injection is capable of launching a number of satellites greater than the number of spacecraft in each of the constellation orbital planes. In this case, the satellites can be launched into a drifting orbit and transferred to their target orbit when the drifting plane and the target plane overlap. They found that launch cost, deployment duration, and service availability are the main factors to determine the effectiveness of a deployment strategy. Finally, concluded that there seemed to be a "gap" in the launch capabilities provided by the state-of-the-art launch vehicle industry in 1999. The reduction of the launch cost and the development of a flexible injection capability could be a solution to fill this gap. Di Carlo *et al.* [6] presented a multi-objective optimization of constellation deployment using low-thrust, considering European launchers only. The method presented allows for exploring the payload capabilities of each launcher using the target inclination and semi-major axis, considering fairing limitations. The paper shows a trade-off between large and small launch vehicles in terms of cost and deployment time. In the paper, they exploit low-thrust transfers to reach Medium Earth Orbit (MEO), optimizing launch cost and mass of propellant. They explored both direct and indirect injections, the latter exploiting a similar methodo to the one proposed by Cornara *et al.* One of the main differences with other authors is the assumption of having an almost completely different injection orbit and nominal orbit in terms of semi-major axis, inclination, and Right Ascension of the Ascending Node (RAAN). This means that the different strategies adopted by Di Carlo *et al.* included in-plane and out-of-plane transfers for orbit raising and for inclination change as well, taking advantage of the effects of the drift of Right Ascension of the Ascending Node (Ω) due to J_2. Kohani and Zong [11] proposed a binary linear program to optimize launcher selection and minimize total cost. The program utilized decision variables and parameters such as launch location, orbits, launchers database, and satellite characteristics. While their approach included heterogeneous constellations, it did not incorporate maneuvering strategies for final deployment. Nonetheless, their focus on aligning satellite characteristics with specific launchers and emphasizing the importance of launch location and heterogeneous constellations provided valuable insights into the optimization of satellite constellations and launcher selection. Di Pasquale *et al.* [7] explored the use of sequential and parallel indirect maneuvers for satellite constellation deployment, generalizing on [8] by including out-of-plane maneuvers as well. Both methods exploit once again the J_2 perturbation, by allowing for differential RAAN drift. The authors found that the injection orbit plays a crucial role in both methods. In particular, the injection orbit should be as close as possible to the operational orbit for parallel deployment to be effective; as the injection distance from the target increases, the parallel method's effectiveness progressively degrades until the sequential method becomes dominant. Specifically, an increase in altitude might mean reducing the useful launcher payload. Thus, the launcher selection optimization is critical. The literature presented explores the combined problem of launcher selection and in-space maneuvering, although with the limitations and assumptions discussed for each case, and in some cases including aspects of the problem, such as the definition of the constellation geometry and the spacecraft. This paper focuses on the mobility aspects of the problem, presenting a methodology to optimize the deployment of constellations of arbitrary size, minimizing cost and deployment duration, and focusing on the use of electric propulsion systems.

2. Problem Statement

This study aims at developing a methodology for solving a multi-objective optimization problem which includes both launcher selection, and in-space maneuvering strategies concurrently. To this end, a Multidisciplinary Design Optimization (MDO) approach is followed. This approach involves optimizing different disciplines that are interconnected

DOI: 10.13009/EUCASS2023-994

through a subset of decision variables, constraints, input, and coupling variables. The study treats the constellation geometry and final orbit as input parameters, focusing on the launcher selection and in-space maneuvering strategy disciplines. Both are managed through a system-level optimizer, which in this case is the Non-dominated Sorting Genetic Algorithm II (NSGA-II). The optimization problem can be formulated mathematically as a multi-objective optimization problem as follows:

$$\begin{aligned} \min \quad & \vec{J} = [C, \Delta t]^T \\ s.t. \quad & \vec{g_s}(\vec{x}) = 0 \\ & \vec{h_s}(\vec{x}) \leq 0 \\ & \vec{x} \in \mathcal{X} \\ & \vec{x_1} \subseteq \vec{x} \qquad \vec{x_2} \subseteq \vec{x} \end{aligned} \tag{1}$$

The optimization problem requires finding the optimal constellation deployment that balances the cost (C) and deployment time (Δt) objectives. The vector $\vec{x}$ represents the design variables of the problem, bounded to a feasible region $\mathcal{X}$. The problem is subject to a set of nonlinear constraints expressed by the vectors $\vec{g}_s$ and $\vec{h}_s$. The goal is to find the Pareto front, which represents the set of optimal solutions that cannot be improved in cost or deployment time without sacrificing performance in at least one of the other objectives. The total cost (C) is the sum of the cost of the selected launchers and the total cost of thrusters carried on board by the satellites. Let l_{cpk} be the cost per kilogram of a launcher, l_{carr} be the number of satellites carried per launcher, and S_m be the mass of the constellation satellite. The total cost is given by:

$$C = \sum_{i=0}^{I} l_{cpki} \cdot l_{carri} \cdot S_m \quad \forall i \in I \tag{2}$$

On the other hand, the total deployment time is composed of the sum of the launcher deployment time, denoted as Δt_1, with respect to the start date of the mission, denoted as t_0, and the sum of the respective maneuvers for the last mile deployment of the satellites, denoted as Δt_2, if needed. The mathematical formulation for this objective is:

$$\Delta t = \max(\Delta t_{1i} + \Delta t_{2i}) \quad \forall i \in I \tag{3}$$

This problem assumes the orbital planes (M) in the satellite constellation are evenly distributed over an arc of angle α. The value of M is defined as one of the input parameters, as shown in Table 2. The required angular separation between adjacent planes, denoted as $\Delta\Omega_{req}$, can be defined as follows [7]:

$$\Delta\Omega_{req} = \frac{\alpha}{M} \tag{4}$$

Therefore, the equality constraints driving the problem are defined below:

$$\vec{g_s}(x) = \left[a^1_{req}(t) - a^1(\Delta t),\ \ldots\ a^M_{req}(t) - a^M(\Delta t),\ \ldots\ \Delta\Omega^1_{req}(t) - \Delta\Omega^1(\Delta t),\ \ldots\ \Delta\Omega^M_{req}(t) - \Delta\Omega^M(\Delta t)\right]^T \tag{5}$$

These constraints state that each satellite must reach its allocated slot, in terms of semi-major axis (a) and the remaining orbital elements, not reported for conciseness. In particular, the planes must match the desired separation ($\Delta\Omega_{req}$). The inequality constraints are expressed as:

$$\vec{h_s}(x) = \left[\Delta t_d - \Delta t, \Delta V_d - \Delta V\right]^T \tag{6}$$

The first constraint specifies that the final deployment time must not exceed the deadline assigned as an input parameter. This constraint is essential for ensuring that the constellation is fully operational within the desired time-frame and can provide the intended services. The second ensures that the ΔV expended per satellite does not exceed the maximum that the propulsion system allows delivering. In addition to these constraints, a logistics constraint is imposed at the Mixed-Integer Programming (MIP)-level to ensure that all launchers employed in a given solution use at least 60% of their capacity. The variables responsible for enforcing this constraint are denoted as L_{min}, and L_{max}, the former is the minimum capacity that can be loaded into a launcher and the latter is the maximum capacity which usually is 100%. These constraints allow considering the constellation as the primary payload on each launch, giving freedom to select the injection orbit as a decision variable. The decision vector comprises several variables, with each one corresponding to a specific launcher opportunity. The length is determined by the available launcher opportunities considered. The subscript k indicates the decision variable per launcher. The variables in the decision vector include

DOI: 10.13009/EUCASS2023-994

a_{inj} and i_{inj}, which represent the semi-major axis injection and inclination injection, respectively. Δ_a and Δ_i represent variations in the semi-major axis and inclination, respectively. Thr_{id} is an integer indicating which thruster from the set of thrusters $\vec{Thr}$ will be used for each iteration. $\vec{l_b}$ is a set of binary variables indicating which subset of launchers' opportunities from $\vec{l}$ is used. Thus, the decision vector can be formulated as follows:

$$\vec{x} = \left[\vec{a_{inj_k}},\ \vec{i_{inj_k}},\ \vec{\Delta a_k},\ \vec{\Delta i_k},\ \vec{Thr_{id k}},\ \vec{l_{bk}},\ \vec{Pr_k}\right]^T \in \mathbb{R}^{|\vec{x}|\cdot|\vec{l}|} \qquad k \in \{1,\ \ldots,\ |\vec{l}|\}$$

Table 1 presents a comprehensive list of the decision variables and their respective ranges. While all possible decision variables are included in this table, the final decision vector may be a subset of these variables. This is because certain decision variables may be excluded based on a variety of factors, such as changes in deployment strategies or the need to introduce or exclude a decision variable at the user level.

Table 1: System Level Design Variables

Variable	**Range**	**Unit**	**Type**
$\vec{a_{inj}}$	$200 \le a \le 600$	km	Continuous
$\vec{i_{inj}}$	$-2 \le i \le 2$	deg	Continuous
$\vec{\Delta a}$	$100 \le a \le 600$	km	Continuous
$\vec{\Delta i}$	$-2 \le i \le 2$	deg	Continuous
$\vec{Thr_{id}}$	$\{1,\ldots,\lvert Thr\rvert\}$	-	Integer
$\vec{l_b}$	-	-	Binary
$\vec{P_r}$	$0 \le P_r \le 100$	-	Continuous

Fig. 1 represents an Extended Design Structure Matrix (XDSM) for the problem at hand, using the notation from [13]. The matrix defines the correlation between the variables and constraints to each discipline of the optimization problem. By identifying the key variables and their interactions, it becomes possible to systematically explore the design space and determine optimal solutions. Concurrent analysis is essential, given the explicit dependencies between the various disciplines. The user inputs defined in the top left part of the matrix are essential variables and flags that drive the optimization process of the genetic algorithm at the system level. Table 2 lists the user inputs and their respective definitions. These inputs are fundamental to the optimization process, as they define the problem space and guide the search for optimal solutions. Careful selection and tuning of these inputs can have a significant impact on the quality of the solutions obtained.

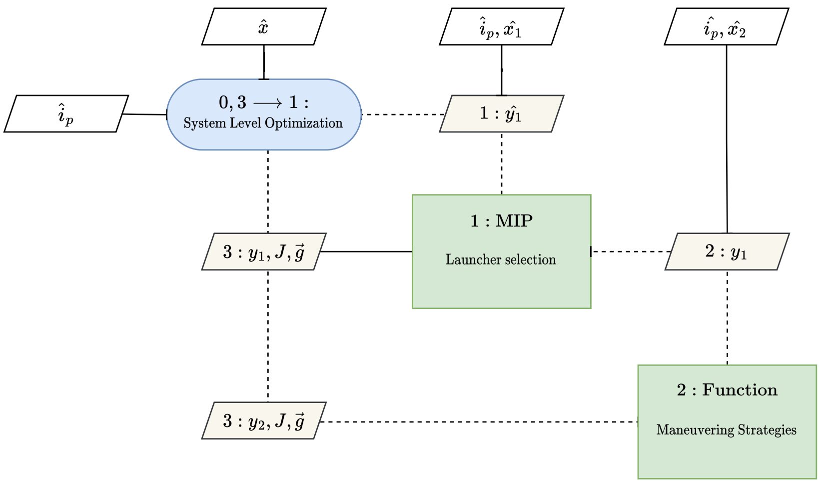

Figure 1: Extended design structure matrix for the constellation deployment problem.

The variable $\vec{l}$ refers to a database that contains information about launcher details, such as capacity, cost per kilogram, cadence, launch location, and availability. Thr is a collection of thrusters: this work focuses on electric propulsion systems. Each thruster object contains information regarding thrust (T), the specific impulse (I_{sp}), and

DOI: 10.13009/EUCASS2023-994

Table 2: Input variables used in the optimization problem

Name	Description	Type
N	Spacecrafts per plane	Integer
M	Number of planes	Integer
S_m	Mass of spacecraft	Float
S_{Pr}	Spacecrafts produced per month	Float
O_a	Initial orbit semi-major axis	Float
O_i	Initial orbit inclination	Float
O_{type}	Orbit type (e.g. LEO, SSO)	Categorical
t_i	Start date of the mission	Datetime
t_d	Deadline for fully operational	Datetime
$\vec{l}$	Launchers opportunities	List
$\vec{Thr}$	Available thrusters	List

Table 3: Local Variables of MIP

Name	Description	Range	Type
L	Full set of launchers and satellite allocation	$L_{min} \leq L \leq L_{max}$	Integer
P	Subset of R_{ij} using binary mask Z	$0 \leq P_{ij} \leq L_{max}$	Integer
Y	Subset of L using binary mask B	$0 \leq Y_i \leq L_{max}$	Integer
R_{ij}	Full set of satellites allocated per plane and launcher	$1 \leq R_{ij} \leq L_{max}$	Integer
Cl	Set for enforcing contiguous plane allocation	0 or 1	Integer
B	Binary variable for launcher selection	-	Binary
Z	Binary variable for plane allocation in each launcher selection	-	Binary

the cost. Both databases have been created by gathering publicly available information available on the internet. One interesting characteristic of the architecture adopted is the capability of increasing the design space by including new decision variables. Two variables that could be particularly useful for optimizing the deployment strategy of the constellation are thr_{exp} and Pr_{exp}. The thr_{exp} variable allows the genetic algorithm to select a different thruster for each spacecraft in the constellation, which could improve the overall performance and efficiency of the deployment. The Pr_{exp} variable enables the consideration of production rates as part of the decision vector, which is especially advantageous when the production rate is uncertain or when the user seeks to evaluate whether a different production rate can enhance the overall results of the constellation deployment.

2.1 Launcher Selection

The selection of launch opportunities is performed through Mixed-Integer Programming (MIP) since it is suitable to handle both integer and binary variables. Using heuristics to solve this problem could be computationally intensive due to the large number of variables and constraints involved. Instead, MIP is a more suitable approach, as noted by [11]. The MIP is used within the objective function of the Genetic Algorithm, where at each iteration of the GA, a global solution containing an optimal selection of launchers from $\vec{l}$ is obtained from the MIP. This solution is the coupling variable to the maneuvering strategies. A subset of the decision variable $\vec{x}$ is used, which is denoted as x_2. This subset includes $\vec{Thr}_{id}$ and $\vec{l_b}$, in addition to the input variables. Table 3 shows the local variables used to solve the problem.

Using MIP can improve the efficiency, cost-effectiveness, and reliability of the deployment process, as the selected launcher will always meet the required constraints and objectives of the problem. Moreover, the MIP is used in

DOI: 10.13009/EUCASS2023-994

the formulation to allocate planes and select the final launcher for a given satellite. A function needed to define some of the constraints is reported below:

$$\text{next}(j) = \begin{cases} 1, & j = M \\ j+1, & j \neq M \end{cases} \tag{7}$$

All variables used in the mathematical formulation are listed in Table 3. The set of equations governing this problem is presented below:

$$\begin{aligned}
\min \quad & \sum_{i=1}^{K} (Y_i p_i) \cdot S_m \\
\text{s.t.} \quad & g_1 : \sum_{j=1}^{i+1} Y_i \leq \text{floor}(S_{P_r} \cdot d_i), \quad \forall i \in I \qquad g_2 : \sum_{i \in I}^{P} P_{ij} = N, \quad \forall j \in J \\
& g_3 : L_{min} \leq L_i \cdot B_i \leq L_{max}, \quad \forall i \in I \qquad g_4 : N \cdot 0.2 \leq P_{ij} \cdot Z_{ij} \leq N, \quad \forall i \in I, \forall j \in J \\
& g_5 : \sum_{j=1}^{M} P_{ij} \leq Y_i, \quad \forall i \in I \qquad g_6 : \sum_{j=1}^{M} Cl_{ij} \leq 2, \quad \forall i \in I \\
& g_7 : Z_{ij} - Z_{\text{next}(j)} \leq Cl_{ij}, \quad \forall i \in I, \forall j \in J \qquad g_8 : -Z_{ij} - Z_{\text{next}(j)} \leq Cl_{ij}, \quad \forall i \in I, \forall j \in J
\end{aligned}$$

The optimization problem involves selecting a set of launchers to minimize the total cost, subject to constraints on the number of satellites produced and the mass of the satellites to be launched. Specifically, let n be the number of launchers, Y_i be the number of satellites carried per launcher, k_i be the cost per kilogram of launcher i, and m_s be the mass of the satellite to be launched. The optimization problem is subject to the following constraints:

- g_1: ensures that the number of satellites placed in a launcher opportunity does not exceed the total number of satellites produced by the launch date.
- g_2: enforces that the number of satellites allocated to each plane is consistent with the predetermined values, preventing any under-allocation or over-allocation of satellites to a specific plane.
- g_3: it limits the number of satellites that can be allocated to a launcher opportunity by restricting the product of L and a binary variable B. This ensures that the number of allocated satellites is within a specified range, preventing over-allocation or under-allocation of satellites to the launcher opportunity.
- g_4: This constraint regulates the allocation of satellites at each plane by imposing bounds on the decision variable P_{ij} and using the binary variable Z_{ij} to permit or deny allocation.
- g_5: To prevent over-allocation of satellites to any particular launcher, c_5 regulates the number of satellites assigned to a launcher. It achieves this by adding up the number of satellites assigned to each plane and verifying that the sum is less than or equal to the maximum allowed number of satellites per launcher.
- g_6, g_7 and g_8: ensure that planes are allocated contiguously in a launcher, regardless of the number of planes. These constraints limit the number of changes in the binary variable Z by using the sum of Cl_{ij}, resulting in the contiguous allocation of planes. To achieve this, the allocation of planes is tracked using C_{ij}, and its sum is constrained to be less than or equal to 2, enforcing contiguous allocation.

The optimization problem includes a constraint that is difficult to express in linear programming due to the product of integer and binary variables, which results in a non-linear problem. Moreover, the nature of the problem requires the inclusion of bounds on these variables, further complicating the constraint. To address this challenge, a linearization technique is employed to transform the constraint into a linear form that can be incorporated into the optimization problem. For each launcher in the set l, every constraint is applied, resulting in the final number of constraints being dependent on the number of launchers utilized during runtime. The output of the Mixed Integer Programming (MIP) algorithm consists of a set of vectors that indicate the allocation of constellation planes, the production rate of spacecraft over time, the final production date, and the cost of launchers. This information will be utilized by subsequent module deployment strategies. In addition to the primary constraints of the problem, various penalties are imposed at the system level to ensure that the optimization process adheres to practical design considerations. For instance, the optimization algorithm must satisfy the following constraint at each iteration: the total mass of the constellation must

DOI: 10.13009/EUCASS2023-994

not exceed the total mass capacity provided by all available launch opportunities. This constraint reflects the physical limitations of the launch system and is a crucial factor in ensuring the feasibility and practicality of the final solution.

$$g(x) = N \cdot M \leq \sum_{i=0}^{\vec{l}} c \tag{8}$$

2.2 Maneuvering Strategies

The in-space maneuvering is carried out using either a sequential or a parallel deployment strategy. By comparing these strategies, the subsection identifies the optimal approach for achieving the desired plane spacing and minimizing deployment time. The approach integrates a broader range of variables and constraints, including input parameters, coupling variables from the MIP model, and a subset of decision variables $\vec{x}_2$ derived from the system-level optimization, in contrast to conventional optimization problems that typically involve limited decision variables and constraints. The main idea behind this implementation is to generate a design space that enables trade-offs between total deployment time and total cost. By exploring this design space, it is possible to evaluate different options and determine the most effective approach for achieving the desired constellation geometry. For example, the trade-off may involve deciding whether it is better to use more or fewer launchers, or whether a direct injection is more efficient than maneuvering in space to reach the final orbit. It is worth noting that achieving the desired Right Ascension of the Ascending Node (RAAN) spacing can be accomplished through both direct out-of-plane and indirect maneuvers. The strategies considered rely on indirect maneuvering, namely exploiting the J_2 harmonic effect to lower the ΔV. The maneuver can be achieved by varying the Semi-Major Axis (SMA) and/or the inclination. The J_2 drift, as expressed in Eq. (9), is in fact dependent on these elements, and on the eccentricity as well (although not strongly). By obtaining a certain difference of elements, the $\Delta\Omega$ can be gradually accumulated over time until the required value is achieved.

$$\dot{\Omega}(a, e, i) = \frac{-3}{2} \cdot \frac{J_2 \cdot R_E^2 \cdot \sqrt{\mu}}{a^{\frac{7}{2}} \cdot (1 - e^2)^2 \cdot \cos(i)} \tag{9}$$

The orbits are assumed to be near-circular, thereby excluding the eccentricity in the maneuvering strategies. Additionally, this work simplifies the problem by modifying either the inclination or the semi-major axis, depending on the constellation's inclination. Specifically, if the inclination of the constellation falls within the near-polar range (80°-100°), the inclination is varied to take advantage of the effectiveness of J_2 drift. Otherwise, the semi-major axis is used. $\vec{x}_{S1}$ and $\vec{x}_{S2}$ are a subset of $\vec{x}_1$. This subset of the decision vector corresponds to either a_{inj} and Δ_a or i_{inj} and Δ_i, depending on whether the sequential or parallel maneuver is required. This decision is made prior to running the genetic algorithm so that the decision vector includes only one of the two options, but not both simultaneously. Furthermore, sequential and parallel maneuvers are computed at each iteration and compared to determine the maneuver that results in the lowest deployment time. If a maneuver violates any of the system level or local constraints, it is discarded and penalized in the next iteration of the genetic algorithm.

In addition to the previously mentioned constraints, it is important to note that the presence of only one plane in the launcher's coupling variable y_1 introduces another important limitation. Specifically, in such cases, any attempt to maneuver the satellites will be pointless as it will be treated as a direct injection scenario resulting in the satellites ending up in the targeted plane without any maneuvering. Table 2 and Table 1 provide a more detailed look at the variables used previously, including those that are part of the subset decision vector $\vec{x}_2$. The computation of a given maneuver ΔV depends on the element to be changed. For semi-major axis changes (Δa):

$$\Delta V_a = \left| \sqrt{\frac{\mu}{a}} - \sqrt{\frac{\mu}{a + \Delta a}} \right| \tag{10}$$

whereas, for inclination changes (Δi):

$$\Delta V_i = V_{inj} \cdot \sqrt{2 - 2 \cdot \cos\left(\frac{\pi}{2} \cdot \Delta_i\right)} \tag{11}$$

Finally, the propellant mass, denoted as m_p, used by the mission can be calculated as follows:

$$m_p = S_m \cdot \left(1 - e^{-\frac{\Delta V}{I_{sp} \cdot 9.81}} \right) \tag{12}$$

DOI: 10.13009/EUCASS2023-994

Here, S_m represents the mass of the satellite, which can be found in Table 2. The specific impulse (I_{sp}) is determined by the thruster used at each iteration, which is selected from $\vec{Thr}$ based on the decision variable index Thr_{id}, the same for the thrust T in Eq. (14). With the assumptions, constraints, and shared functions now defined, each approach will be presented in its respective section below, outlining its specific details.

2.2.1 Sequential Deployment

In this method, the launcher releases the satellites into a designated injection orbit, which is determined by the subset decision vector $\vec{x}_2$. Each satellite is then assigned a specific wait time before it can maneuver to the nominal orbit. The wait time is calculated considering the desired right ascension of the ascending node (RAAN), the RAAN drift at both the orbit injection and nominal orbit, and the satellite's propulsion capabilities. Since the deployment of the satellites in this method is sequential, the deployment time is given by the latest satellite reaching the nominal orbit. In mathematical terms:

$$\Delta t^k_{seq} = \frac{\Delta\Omega^{\max}_{req}{}_k}{\dot{\Omega}^k_{nom} - \dot{\Omega}^k_{inj}} + \Delta t^k_{man} \quad \forall k \in Y \tag{13}$$

Here, $\Delta\Omega^{\max}_{req}$ represents the maximum change in RAAN required for a given launcher in the subset of selected launchers Y. $\dot{\Omega}^k_{nom}$ is the RAAN rate at the nominal orbit, while $\dot{\Omega}^k_{inj}$ is the RAAN rate at the injection orbit. The time it will take the satellites to reach the target orbit, denoted as Δt^k_{man}, can be calculated using the following equation:

$$\Delta t^k_{man} = \frac{\Delta V_l}{T/S_m} \quad \forall k \in Y \tag{14}$$

where T is the thrust of the propulsion system. To define $\Delta\Omega^{\max}_{req}$, the set of values indicating the change in RAAN for a given launcher per satellite must be determined. This information can be obtained from the coupling variable y_1, which is used to determine the plane allocation. The plane allocation is represented as two sets of values, which we will denote as U and V. The set U contains the number of planes, while the set V contains the number of satellites allocated to each plane. To find the maximum distance between each plane and the others, the set V can be iterated over. This allows the definition of the maximum value of the plane distance array $\vec{P_D}$, which represents the maximum distance for a given plane to others containing satellites. It is important to note that this distance can only be calculated sequentially from left to right or right to left, depending on whether the orbit is retrograde or not.

$$\vec{P}_D = ||i - (j+1)|| \quad \forall i \in V, \quad \forall j \in V \tag{15}$$

By obtaining a subset of values from $\vec{P}_D$ called $\vec{P}_{Dmax}$, which contains only the maximum values of each plane array distances, we can write:

$$\vec{P}_{Dmax} = \max(\vec{P}_D) \tag{16}$$

Finally, to obtain the point where the minimum change of planes occurs the minimum scalar value of set $\vec{P}_{Dmax}$ can be obtained. The plane corresponding to this minimum value is the starting point for the sequential maneuver, also known as the RAAN injection.

$$\vec{\Omega}_{inj} = \min(\vec{P}_{Dmax}) \tag{17}$$

Eq. (4) represents the difference in RAAN required for each plane. By multiplying this value with the number of planes distributed evenly over a circle, an array of RAAN values representing each plane can be constructed. These values are stored in a set denoted as P_m. From this set, the necessary Δ RAAN for each plane from the injection point can be calculated using the aforementioned formula.

$$\Delta\vec{\Omega}_{req} = \begin{cases} i > 0, & P_{m,i} - \vec{\Omega}_{inj} \\ i \leq 0, & 0 \end{cases} \quad \forall i \in V \tag{18}$$

With the provided context, now the formula to calculate $\Delta\Omega^{\max}_{req}{}_k$ can be defined as follows:

$$\Delta\Omega^{\max}_{req}{}_k = \max\left(\left|\Delta\vec{\Omega}_{req}\right|\right) \tag{19}$$

DOI: 10.13009/EUCASS2023-994

2.2.2 Parallel Deployment

This method involves the simultaneous maneuvering of multiple satellites, resulting in a reduction in deployment time that varies depending on the orbit injection. This work assumes that the injection orbit is equal to the nominal orbit. This is because [7] suggests that it is advantageous to remain as close as possible to the nominal orbit. This method consists of three distinct phases. In the first phase, the satellites transfer from the injection orbit to a coasting one. The second phase, known as the coasting phase, allows to accumulate RAAN drift thanks to the difference in orbital parameters with respect to the nominal orbit. The duration of this phase is critical, as it directly affects the deployment time, which can be either longer or shorter depending on the coasting orbit. The third and final phase involves a second maneuver to regress to the nominal orbit, which is the desired orbit for the constellation. During all three phases, the RAAN drift accumulates and can be expressed as follows:

$$\Delta\Omega_i = \Delta\Omega^i_{man1} + \Delta\Omega^i_{coast} + \Delta\Omega^i_{man2} \tag{20}$$

The aforementioned equation states that the final $\Delta\Omega$ accumulated over the entire deployment for satellite i of a given launcher is equal to the sum of the drift in the phases of the parallel approach. In this method, the mathematical formulation to retrieve the total maneuvering time is as follows:

$$\Delta t^k_{par} = \frac{\Delta\Omega_{req} + (2 \cdot (\Delta t^k_{man} \cdot \dot{\Omega}^k_{man} - \dot{\Omega}^k_{coast}))}{\dot{\Omega}^k_{coast} - \dot{\Omega}^k_{nom}} \tag{21}$$

[7] demonstrated that the optimal in this case is to have all the satellites maneuver in order to achieve the deployment time indicated by Eq. (23). Compared to the sequential method, the parallel relies on distinct calculations for $\dot{\Omega}$ in each of the three phases. Another significant difference lies in how ΔV_a is calculated for the non-high inclination approach. In this case, the propellant mass used and ΔV_i remain the same as in the previous method. However, ΔV_a can be calculated using the following equation:

$$\Delta V_a = \left| \sqrt{\frac{\mu}{O_a}} - \sqrt{\frac{\mu}{O_a + \Delta a}} \right| \tag{22}$$

Upon calculating and adding the aforementioned value to the designated set $\Delta\vec{t^k_{par}}$, the maximum value can be determined. This value corresponds to the longest maneuver executed during the parallel approach and is commonly referred to as the total deployment time. It is worth noting that this is due to the fact that the maneuvers are performed concurrently. The aforementioned relationship can be represented mathematically, as shown in Equation 23.

$$\Delta t_{par} = \max(\Delta\vec{t^k_{par}}) \tag{23}$$

After completing the previous steps, the final task is to determine how to select the RAAN injection. While a complex solution exists, two recent studies [7, 8] have simplified the approach and identified the optimal method for allocating the RAAN injection in the parallel case. The studies suggest that the RAAN injection should be positioned at the midpoint. If the number of planes to be allocated is even, the midpoint in terms of RAAN should be calculated, and the RAAN injection should be selected at that point. On the other hand, if the number of planes is odd, the methodology is simpler: the RAAN injection should be deployed in the plane that is identified as the middle one.

3. Results

To provide a practical application of the optimization process, this study presents the results of a mission case study of a satellite constellation deployment. The case study not only demonstrates the effectiveness of the optimization process in achieving the desired deployment time for a satellite constellation, but also provides a set of solutions that meets the objectives and constraints given. This allows for trade-off analysis of the objectives, enabling decision-makers to choose the most optimal solution based on their specific needs.

By analyzing the case study results, insights can be gained into the optimal approach for deploying satellite constellations, including the maneuvering strategies and system-level and local constraints that must be considered. The case study provides a real-world scenario that can inform the design of future satellite constellations, and the insights gained from the analysis can contribute to the development of more efficient deployment strategies.

DOI: 10.13009/EUCASS2023-994

Table 4: 16U Constellation input parameters

Parameter	**Value**	**Unit**	**Parameter**	**Value**	**Unit**
N_{gen}	50	-	O_a	7080	km
pop_{size}	50	-	O_i	98.8	deg
N	23	-	O_{type}	LEO	Categorical
M	20	-	t_i	01 Jan 2020	Datetime
S_m	22	-	t_d	01 Jan 2025	Datetime
S_{Pr}	26	-	l	73 Opportunities	List
thr	Fixed Thruster	-	T	0.001475	N
Isp	925	s			

Table 5: Genetic algorithm solutions

Parameter	**Point 1**	**Point 2**	**Point 3**
Cost (M€)	364.98	264.35	190.44
Time (days)	600	785	1695.6
Nº Launchers	40	21	4
Ratio (Vega:Electron)	0:40	3:18	4:0

3.1 Mission Case Study

The objective of the case study is to evaluate the feasibility of the IoST (Internet of Space Things) constellation described by Kak *et. al.* [10]. The constellation consists of 23 satellites per plane, with a total of 460 satellites to be deployed across 20 equally spaced planes. A 16U CubeSat form factor with a mass of 22 kg is considered. The production of satellites is scheduled to begin on January 20th, 2020, and the full operational capability is expected to be achieved by 2025, assuming a production rate of 26 satellites per month. No geopolitical restrictions are imposed in the launcher selection, providing flexibility to take advantage of launch opportunities in the future. For this case study, two launchers are considered: Vega C and Electron. The launch opportunities of this selection are based on the expected cadence for each one, although a real-time manifest can be implemented for short-term planning. Although the methodology developed is capable of exploring the propulsion system selection, in order to contain the length of this study, a predefined and fixed electric propulsion system has been considered. The case study input parameters are provided in Table 4. The primary objective of this case study is to assess the deployment time and costs associated with various decision vector combinations.

The Pareto optimal front for total cost and deployment time is illustrated in Fig. 2. Three main solutions have been obtained. These offer a trade-off opportunity between the two objectives. The limited number of solutions is due to the discrete nature of the possibilities arising from the use of two launchers. On the right side, a solution is derived exclusively from cost-effective and high-capacity Vega C launcher opportunities. This solution requires significant maneuvering effort, in order to cover all the orbital planes from a very limited number of launches. The left-most point relies solely on more expensive and lower-capacity Electron launchers, exploiting only direct injections. The intermediate solution combines the utilization of both launchers, employing a mix of maneuvering and direct injection launches. The final cost, deployment time, and utilization ratio of the launchers are collected in Table 5 for the three solutions.

Upon closer examination of the Pareto front, it is possible to see that the fastest deployment time (about 1.6 years) corresponds to the most expensive case as well, around 364.9 M€. On the other hand, the longest deployment time case is priced at approximately 190.4 M€ paid with a penalty of four additional years to complete the roll-up. Consequently, deploying the constellation involves a trade-off of approximately 174 M€ to enable earlier operations. The middle point presents a compromise solution in both objectives, as it is slightly slower in deployment time than the fastest (about 2.14 years versus 1.6), and it has a total cost of 264.3 M€, again placing it as a compromise between

DOI: 10.13009/EUCASS2023-994

the first two points. Depending on the mission's budget and requirements, any of these options can be chosen.

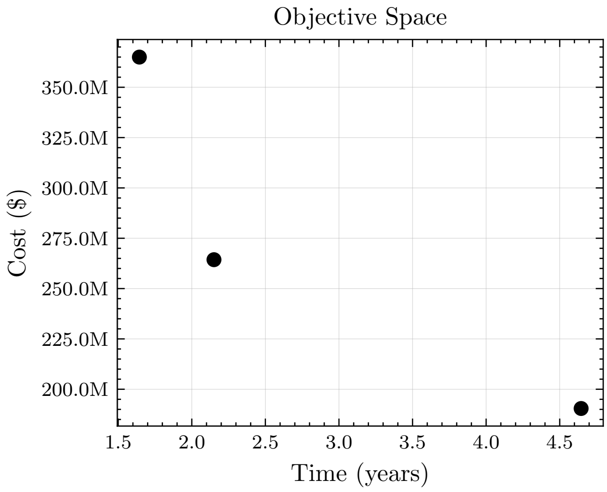

Figure 2: Pareto optimal front for the case study presented.

A detailed timeline of the constellation deployment is presented, aimed at gaining more insight into each solution. The timelines are presented for each point in Fig. 3, Fig. 5, and Fig. 7, showcasing various output variables plotted against a shared x-axis of time and the number of satellites represented on the y-axis. The figures present several variables that contribute to a comprehensive understanding of the constellation deployment process. The deployment deadline, which represents the maximum expected operational date specified in the mission requirements, is shown with a solid red line. The yellow horizontal arrows indicate the duration of maneuvers (if any) commencing shortly after the launcher injection, emphasizing that exceeding the deployment deadline is not permissible. This highlights the adherence of the selected solution to the mission case study's stringent constraint. Additionally, the maneuver strategy (parallel or sequential) and its duration are shown as well, for each maneuver. The time of completion of the satellite manufacturing is marked with a dashed green vertical line. This signifies that all satellites for the constellation have been fabricated in the factory. Beyond this point in time, there is no further increase in total production as the remaining variable in the warehouse diminishes. The warehouse's remaining satellites, which reflect the number of satellites produced over time, are shown with a solid blue line. Initially, this variable increases, but as launch opportunities arise, it decreases by the number of satellites launched into space. Lastly, the number of satellites remaining to complete the deployment is displayed with a black dash-dot curve. This value steadily declines over time, with each launch event denoted by a black dot.

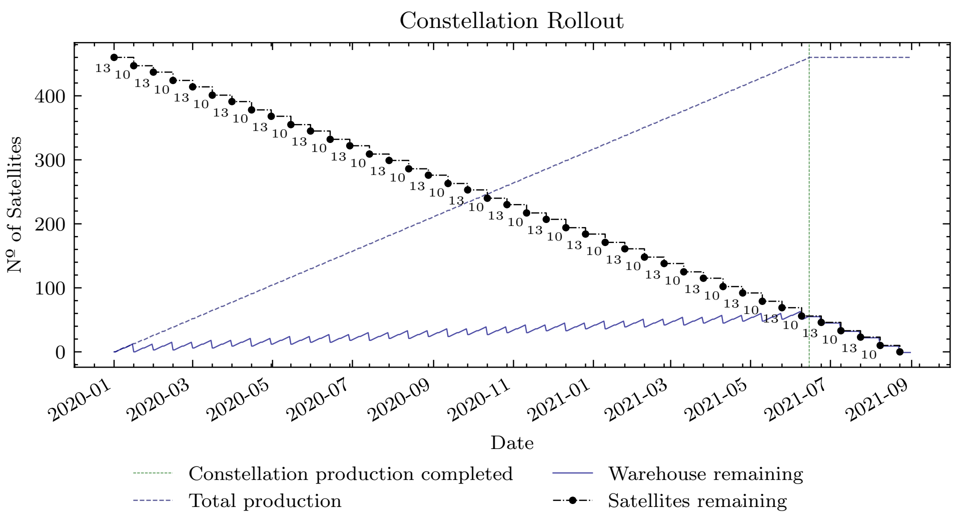

Figure 3: Constellation deployment timeline for the minimum-time solution from Fig. 2.

DOI: 10.13009/EUCASS2023-994

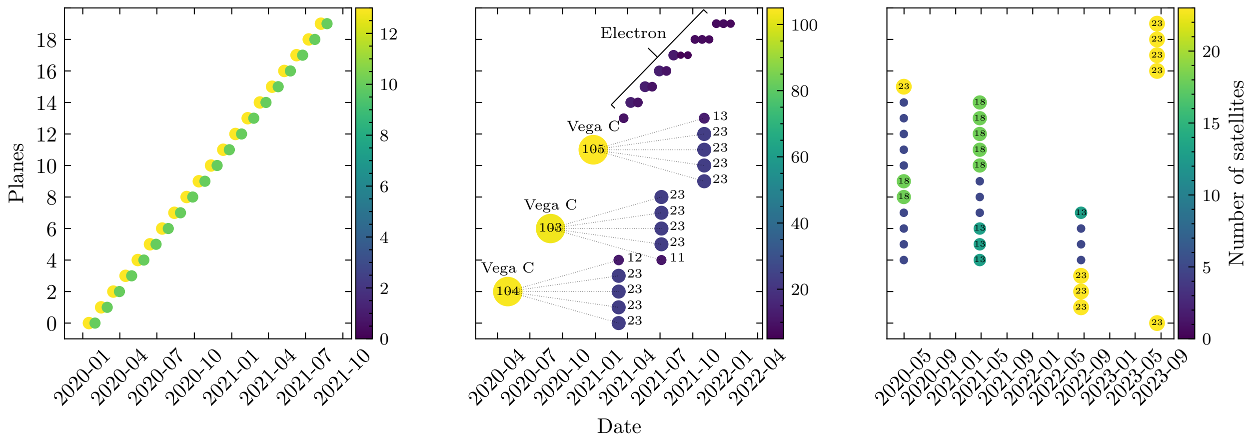

Figure 4: Plane allocations for each solution of the Pareto optimal front.

Fig. 3, representing the timeline of the minimum-time solution in the Pareto front, provides valuable insights into the deployment strategy. Notably, all launchers exhibit direct injections, indicated by black dots, as there are no yellow arrows representing maneuvers. The solution effectively exploits the high cadence of the Electron launcher, which is expected to conduct approximately two launches per month. This choice aligns with the production rate expected for the constellation, slightly exceeding 23 satellites launched per month in this scenario. The plot also highlights the warehouse's remaining variable, demonstrating an initial increase followed by a fluctuation pattern. The slightly steady increase of around 3 satellites per month indicates a higher production rate compared to the number of launches. This insight can be quickly grasped by observing the slope of the line: a steeper incline indicates more satellites being created than launched, while a shallower slope suggests more satellites being launched than produced. The chosen solution focuses on expediting the constellation deployment by eliminating maneuvers. In this approach, each launcher carries the satellites to their final plane destinations, utilizing two launches to fill a single plane, one with 10 satellites and the other with 13 as shown by the left plot in Fig. 4.

This strategy takes advantage of the maximum capacity of the Electron launcher, which can accommodate up to 13 satellites, in this case. By maximizing the launcher utilization, the deployment time can be reduced. However, a challenge arises in terms of the number of satellites. Each plane can accommodate 23 satellites, and by filling two Electron launches for the same plane, three of the satellites would require maneuvering to align with the final plane. This maneuvering introduces complexity and increases the overall deployment time, which would result in a dominated point in the front. Consequently, this peculiar 13/10 pattern offers the advantage of shorter deployment time but comes at the expense of higher costs compared to other solutions.

Fig. 5 shows the timeline for the middle point in Fig. 2. This solution involves initially utilizing the Vega C launcher and then embracing a cascade strategy by leveraging the higher cadence of the Electron launcher compared to the lower cadence of the Vega C. The optimal point for the constellation deployment of the middle option is achieved by initiating maneuvers early in the mission. As the Vega C launcher has a higher capacity, it is expected to carry multiple satellites, requiring maneuvers to disperse them effectively. To minimize the additional time required for maneuvering, these maneuvers need to be executed at the earliest stage of the deployment roll-out. For this approach to be effective, it is crucial that none of the maneuvers exceed the time of the final Electron launch. It can be observed in the plot that none of the maneuvers surpass the time of the last Electron launch, thus confirming its place within the Pareto front as an optimal point. Conversely, if any maneuver were to exceed the duration of the final Electron launch, it would indicate a sub-optimal approach. To execute this approach effectively, optimal plane allocation and utilization of extra Electron launches help minimize maneuvering time. The plane allocation for this solution is shown as the middle plot in Fig. 4. For the Vega C launches, all the satellites are released in the same injection plane. Through maneuvering, they are subsequently aligned with the correct plane, resulting in a final time equal to the injection time plus the maneuvering duration. It is important to note that based on the parallel methodology, when the number of planes is odd, the injection plane coincides with the middle plane. The presence of shorter maneuvers in the proposed approach is a result of reduced plane dispersion after injection. Additionally, the use of the Electron launcher contributes to reducing the load on the Vega C launches, as depicted in Fig. 6, although causing the cost to increase.

In comparison, the deployment strategy represented by the minimum-cost solution in the Pareto front, as depicted in Fig. 7, demonstrates longer maneuvering times due to the absence of Electron launches in the mission. Carrying a

DOI: 10.13009/EUCASS2023-994

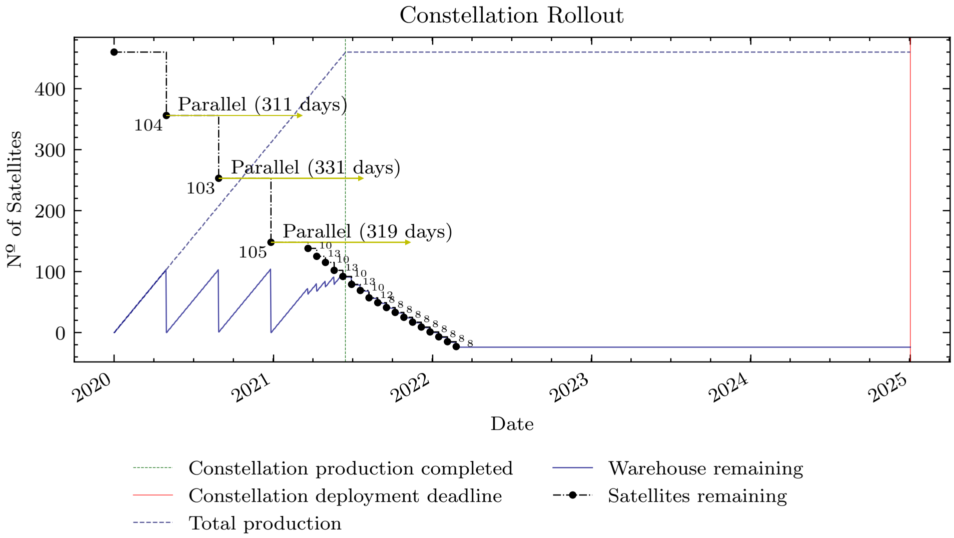

Figure 5: Constellation deployment timeline for the compromise solution from Fig. 2.

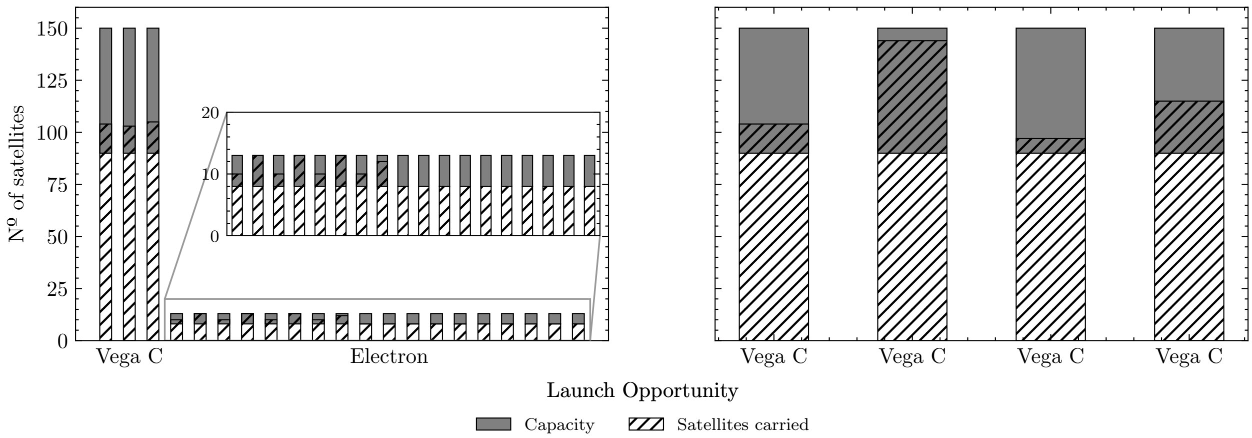

Figure 6: Launchers selection for middle point (left) and minimum-cost point (right) of Fig. 2.

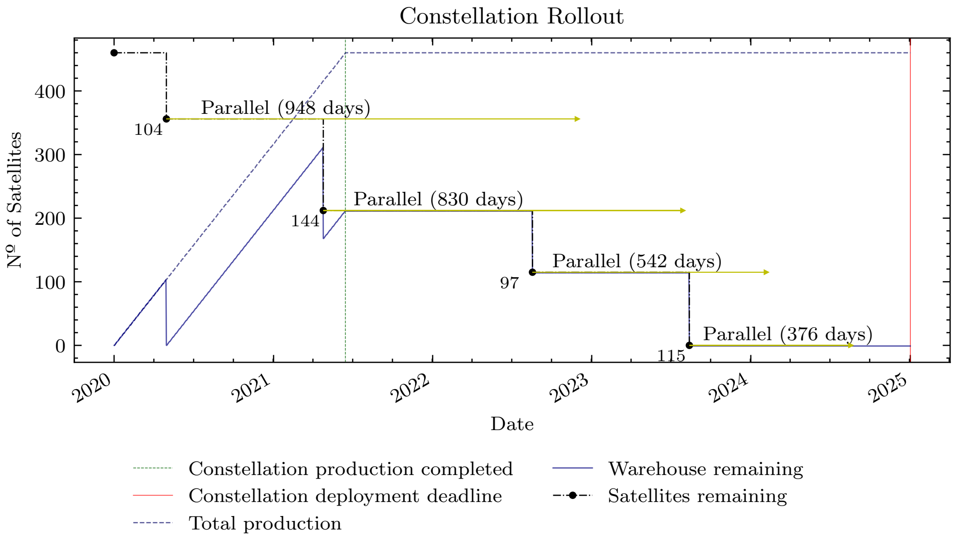

Figure 7: Constellation deployment timeline for the minimum-cost solution from Fig. 2.

DOI: 10.13009/EUCASS2023-994

larger number of satellites using the Vega C launcher necessitates dispersing more planes (larger total $\Delta\Omega$ required), resulting in increased maneuvering time, if compared to the maneuvers from the middle point. Fig. 4 (right) depicts the plane allocation for this solution. For example, the first launch requires approximately 948 days for maneuvers with 12 allocated planes, while the last launch takes around 376 days with 7 allocated planes. These durations are comparable to those observed for the middle point of the Pareto front. Although the choice of maneuvering strategy contributes to the maneuvering time, the primary factor influencing the duration is the number of planes to be dispersed. This holds true for both sequential and parallel maneuvering strategies, although the parallel approach seems to dominate in all the solutions, thanks to the lack of altitude constraints from the launchers. Another noteworthy observation is that the deployment time is extended due to the minimum capacity requirement for the launchers. Additionally, the completion of constellation production shifts towards the left side of the plot. This shift is a consequence of the need to produce additional satellites to meet the capacity of the Vega C launcher. The lower cadence of launches also contributes to the longer deployment time, as it necessitates taking advantage of spaced opportunities for deployment. These factors, including minimum launcher capacity requirements and the need for maneuvering, contribute to the overall extension of the deployment time. When considering Fig. 6, it becomes evident that the system-level optimizer primarily adjusts the number of satellites allocated to each launcher. This adjustment consequently affects the plane allocation in order to meet the deadline constraint. Notably, the optimizer consistently assigns the minimum required capacity, as depicted by the gray filled area representing the acceptable capacity range. While the existing solution is satisfactory, there is undoubtedly an opportunity for enhancement by allocating more satellites to the left side. Despite the first launcher already incorporating a substantial number of planes, there remains untapped capacity capable of accommodating additional satellites. By leveraging this available space and prioritizing increased allocation to the left, the overall deployment time on the right side can be significantly reduced. This strategic adjustment holds the potential to significantly improve the overall efficiency and effectiveness of the deployment strategy. Implementing such an allocation adjustment could be facilitated through the imposition of penalties during the system-level optimization process. The results presented provides evidence of the effectiveness of the optimization problem and the valuable solutions obtained in this study. They demonstrate the successful fulfillment of the hard constraints in the mission case study and offer insightful observations on the optimization process for satellite constellation deployment. The clear existence of a well-defined Pareto front underscores the achievement of minimized objectives and showcases the efficacy of the optimization approach in addressing the complex challenges associated with satellite constellation deployment.

4. Conclusions

This study presents an optimization methodology for satellite constellation deployment including launcher selection and in-space maneuvering, aiming to address the intricate challenges involved. To approach proposed, based on a Multidisciplinary Design Optimization philosophy, is based on a system-level optimization carried out by a genetic algorithm, which coordinates the various disciplines such as the launcher selection and maneuvering strategies. This approach allows for a comprehensive examination of the factors influencing constellation deployment, offering insights into the optimization process.

The design variables considered in the study demonstrate their efficacy in affecting the respective objectives. Both disciplines show promising and expected behavior, with the Mixed Integer Programming (MIP) converging effectively for launcher selection and the maneuver strategies displaying excellent performance by optimizing deployment time within their constraints. Additionally, the study utilizes a genetic algorithm, which proves to be efficient and fast, even when running the MIP at each iteration. These positive outcomes highlight the effectiveness of the chosen methodology and algorithms in addressing the constellation deployment problem.

The results of the case study presented offer valuable insights into the optimization process for the given problem. The study effectively minimized both time and cost objectives while satisfying the specified constraints and bounds of each decision variable. The solutions found show a large trade-off opportunity between the cost and time objectives, thanks to the concurrent selection of launcher logistics and maneuvering strategy optimization. The minimum-time solution found relies on dedicated small launchers which allow for direct orbital injection and high cadence. However, the high cost-per-kg makes the solution the most expensive in terms of cost. The minimum-cost solution found, on the other hand, relies on larger launchers and heavy usage of ‘in-space maneuvering. The lower cost-per-kg allows for exceptional cost savings, however, at the expense of roughly three times longer deployment times. This is due to the lower cadence and the need for in-space maneuvering. An additional point has been discovered as well, offering a compromise solution, which uses a mix of large, and small launchers and in-space maneuvering. In conclusion, this study presents a robust and effective approach to optimizing the deployment of satellite constellations, and the obtained results provide actionable insight for future advancements in this field.

DOI: 10.13009/EUCASS2023-994

References

[1] NewSpace Insight. https://newspace.im. Accessed: May 30, 2023.

[2] Oneweb confirms successful launch of 34 satellites. Feb 2022.

[3] Bryce and Space Technology. Smallsat by the numbers, 2021, oct 2022.

[4] Irene A. Budianto and John R. Olds. Design and Deployment of a Satellite Constellation Using Collaborative Optimization. *Journal of Spacecraft and Rockets*, 41(6):956–963, November 2004.

[5] Stefania Cornara, Theresa Beech, and Miguel BellÃ. Satellite Constellation Launch, Deployment, Replacement and End-of -Life Strategies. page 19, 1999.

[6] Marilena Di Carlo, Lorenzo Ricciardi, and Massimiliano Vasile. Multi-objective optimisation of constellation deployment using low-thrust propulsion. https://arc.aiaa.org/doi/10.2514/6.2016-5577, 2016.

[7] Giuseppe Di Pasquale, Manuel Sanjurjo-Rivo, and Daniel Pérez Grande. Optimization of constellation deployment using on-board propulsion and Earth nodal regression. *Advances in Space Research*, 70(11):3281–3300, December 2022.

[8] Hojat Mahdisoozani et al. Developing novel multi-plane satellite constellation deployment methods using the concept of nodal precession. *Advances in Space Research*, 68(7):1615–1624, 2021.

[9] Andrew Jones. Fcc approves amazon's ambitious project kuiper satellite constellation. https://www.space.com/fcc-approves-amazon-constellation-kuiper. Accessed: May 30, 2023.

[10] Ahan Kak and Ian F. Akyildiz. Large-Scale Constellation Design for the Internet of Space Things/CubeSats. In *2019 IEEE Globecom Workshops (GC Wkshps)*, pages 1–6, December 2019.

[11] Saeid Kohani and Peng Zong. Formulization of Cost Estimation in Optimizing Constellation Satellite, Launch Vehicle and Launch Place. 6:13, 2021.

[12] Erik Kulu. Satellite Constellations - 2021 Industry Survey and Trends.

[13] Joaquim R. R. A. Martins and Andrew B. Lambe. Multidisciplinary Design Optimization: A Survey of Architectures. *AIAA Journal*, 51(9):2049–2075, September 2013.

[14] Jonathan. C. McDowel. Jonathan's space report, oct 2022.

[15] Tania Savitri, Youngjoo Kim, Sujang Jo, and Hyochoong Bang. Satellite Constellation Orbit Design Optimization with Combined Genetic Algorithm and Semianalytical Approach. *International Journal of Aerospace Engineering*, 2017:e1235692, May 2017.