Optimization and Engineering
https://doi.org/10.1007/s11081-023-09834-8

RESEARCH ARTICLE

Matheuristics approaches for the satellite constellation design problem

Luca Mencarelli[1] · Julien Floquet[2] · Frédéric Georges[2]

Received: 2 February 2023 / Revised: 12 June 2023 / Accepted: 15 August 2023

Abstract

In this paper, we propose two novel matheuristic algorithms, i.e., heuristics based on mathematical formulations of the problem, in order to find a good feasible solution to the *satellite constellation design problem* for discontinuous coverage with a constrained revisit time. This problem consists in searching for a constellation able to periodically observe several targets at the Earth surface with the smallest number of satellites achievable. A *Feasibility Pump* approach is described: we specifically adapt the Feasibility Pump procedure to our design problem and we report results highlighting the benefits of this approach compared to the base Mixed Integer Nonlinear Programming (MINLP) algorithm it is derived from. Then, we propose a second matheuristic based on the discretized Mixed Integer Linear Programming (MILP) formulation of the problem, which outperforms the plain MILP formulation.

Keywords Satellite constellation design · Discontinuous coverage · Revisit time · Mission planning satellites · Feasibility pump · Mixed integer (non)linear programming

Mathematics Subject Classification 90

✉ Luca Mencarelli
luca.mencarelli@unipi.it

Julien Floquet
julien.floquet@onera.fr

Frédéric Georges
frederic.georges@onera.fr

[1] Dipartimento di Informatica, Università di Pisa, 56127 Pisa, Italy

[2] DTIS, ONERA, Université Paris Saclay, 91123 Palaiseau, France

Published online: 27 September 2023

1 Introduction

Satellites constellations are useful for a multitude of applications, but they are also extremely costly to set up. As a consequence, minimizing the number of satellites required for a constellation is of very high interest. In the case of applications requiring a continuous coverage of the Earth, optimization techniques are already well figured out, but not so much in the case of discontinuous coverage. One can refer to our previous work (Mencarelli et al. 2022) for further details on the satellite constellation design problem.

Mencarelli et al. (2022), we proposed new Mixed-Integer NonLinear Problem (MINLP) and Mixed-Integer Linear Problem (MILP) approaches in order to provide an exact solution to the satellite constellation design problem with constrained revisit time and discontinuous coverage, where targets at the surface of the Earth have to be observed periodically by at least one satellite. Both approaches aimed at increasing the number of parameters taken into account when compared to existing techniques. However, because of the high number of parameters, both approaches suffered from a scalability issue: an average desktop computer would be limited to solving small instances of the problem.

As a consequence, in this paper, we propose two new matheuristic algorithms in order to solve larger instances of the problem than in our previous work: a Feasibility Pump and a MILP-based procedure. To the best of our knowledge, we are the first to propose or adapt heuristic methods based on the mathematical formulation of the design problem for this emerging field of application.

The Feasibility Pump algorithm is built upon our previous MINLP formulation, while the MILP-based matheuristic algorithm is derived from our previous MILP algorithm. Both aim at outperforming the previous mathematical formulation from which they are derived.

The remainder of this paper is organized as follows: in the next two sections we summarize respectively the MINLP and discretized MILP mathematical models that we proposed in Mencarelli et al. (2022). In Sect. 4, we briefly describe the general Feasibility Pump framework for (nonconvex) MINLP, and we explain how it was adapted to our constellation design problem in Sect. 5. The MILP-based procedure approach is described in Sect. 6. Section 7 is dedicated to the presentation and discussion on the obtained results in terms of solution quality and computational cost. Finally, the conclusions of this work are presented in Sect. 8.

2 MINLP Mathematical model

The MINLP model is fully described in Mencarelli et al. (2022): in this section, we briefly resume the mathematical formulation, underlining the modifications we introduce in this work to improve the realism of the physical model.

In this paper, as in our previous work, we consider the following (restrictive) assumptions:

- the Earth is considered to be a perfect sphere;

- the orbits of the satellites are circular and described by a Keplerian model with 4 free orbital parameters:
  - the radius of the orbit a;
  - the inclination I;
  - the right ascension of the ascending node (RAAN) Ω;
  - the mean anomaly M;
- the time is discretized.

As usual, we consider two kinds of variables:

- a set of binary variables $\xi_{i,t,m}$ indicating if the satellite $i \in [s]$ observes the target $m \in \mathcal{X}$ at time $t \in [T]$;
- a set of continuous variables $\sigma_i \in \Sigma_i$ $(i \in [s])$ representing the orbital parameters of the i-th satellite.

Moreover, let:

- lat_m and $long_m$ be the latitude and the longitude, respectively, of the target m to observe;
- $lat(\sigma_i, t)$ and $long(\sigma_i, t)$ be the latitude and the longitude, respectively, of the projection on the Earth surface of the i-th satellite at the t-th time-step (these two functions depend from the orbital parameters of the corresponding satellite);
- θ_i^{max} be the maximum observation aperture angle for satellite $i \in [s]$, measured form the center of the Earth;
- Σ_i be the set of possible orbital parameters for the i-th satellite;
- $\Delta t \in \mathbb{R}^{+*}$ be the revisit time, defining the length of the time intervals during which each target must be observed at least once.
- $dt \in \mathbb{R}^+$ be the time discretization step,
- $\mathcal{T}(t_k, \Delta t) := \{t_k, t_{k+1}, t_{k+2}, \ldots, t_{k+\Delta t/dt-1}\}$ be the set of time-steps in a time interval $[\![t_k, t_{k+\Delta t/dt-1}]\!]$;
- T be the duration of the simulation;
- $[N] := \{1, \ldots, N\}$ be a notation to define a set (for instance, $[T] = \{t_0, t_0 + dt, t_0 + 2dt, \ldots, T - dt\}$).

We aim to find the minimum number $s \in \mathbb{R}^+$ of satellites such that:

$$\sum_{\substack{t \in \mathcal{T}(k, \Delta t) \\ i \in [s]}} \xi_{i,t,m} \geq 1 \quad \forall k \in [\lfloor T/\Delta t \rfloor],\ \forall m \in \mathcal{X}$$

$$\begin{aligned} \xi_{i,t,m} &= 0 \Longrightarrow \max\{|lat_m - lat(\sigma_i, t)|, |long_m - long(\sigma_i, t)| \cos(lat_m)\} \\ &\qquad \geq \theta_i^{max} \quad \forall i \in [s], \forall t \in [T], \forall m \in \mathcal{X} \\ \xi_{i,t,m} &\in \{0, 1\} \quad \forall i \in [s], \forall t \in [T], \forall m \in \mathcal{X} \\ \sigma_i &\in \Sigma_i \quad \forall i \in [s]. \end{aligned}$$

The first constraint requires that at least one satellite observes the target $m \in \mathcal{X}$, i.e., at least one binary variable $\xi_{i,t,m}$ is equal to 1 in the period corresponding to the revisit time.

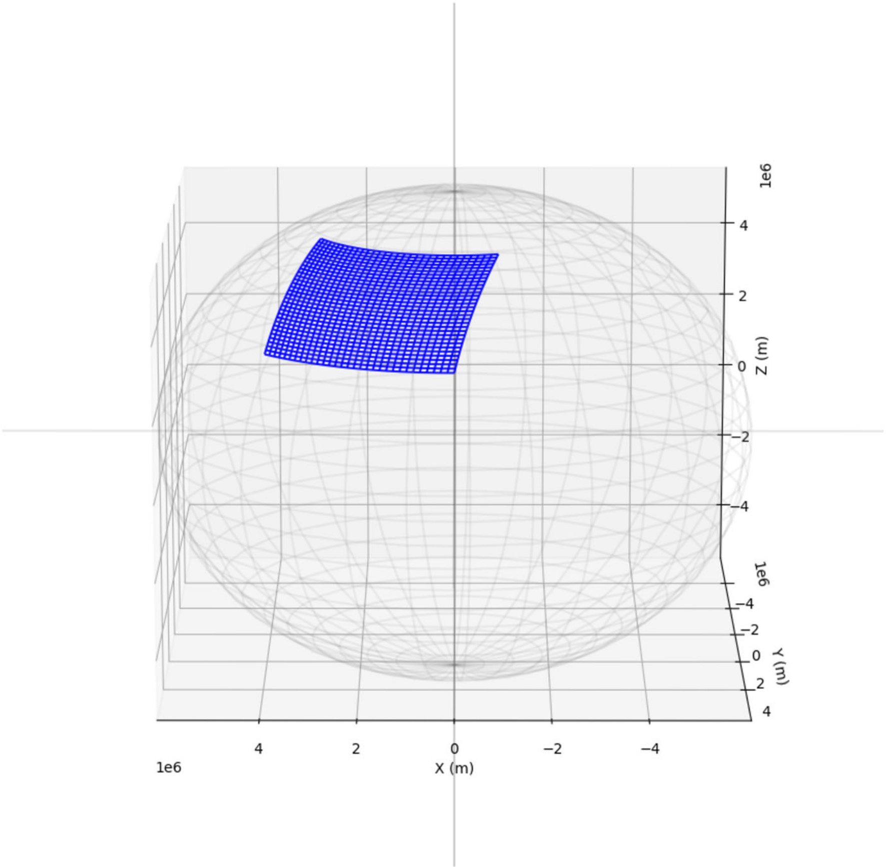

Fig. 1 Illustration of the ground surface observed by a satellite (blue). (Color figure online)

The constraints on the indicators $\xi_{i,t,m}$ are observability constraints: they reflect that if the target m is observed at time t by the satellite i, then $\xi_{i,t,m} = 1$. The target is observed if it belongs to the Earth region observed by the satellite (see Fig. 1). In this paper, the formula for the ground surface observed by the satellite is slightly more realist than the one previously introduced in Mencarelli et al. (2022).

In order to determine if a target belongs to the observation field of a satellite, one can compare the latitudes and longitudes of the target and the projection of the satellite on the surface of the Earth:

$$\|lat_m - lat(\sigma_i, t)\| \leq \theta_i^{max} \tag{1}$$

$$\|long_m - long(\sigma_i, t)\| \cdot \cos(lat_m) \leq \theta_i^{max} \tag{2}$$

These inequalities allow to obtain an observation area with a size and shape independent from the latitude, unlike in our previous work (Mencarelli et al. 2022). The covered area is a square defined by the maximum aperture angle θ^{max} measured from the Earth center.

The latitude and longitude used in the MINLP formulation are computed as follows (see Mencarelli et al. 2022 for further details):

$$\begin{aligned} lat(\sigma_i, t) = {} & \arcsin(\sin_i^I a_i \sin(M_{i,t}) \cos(V_i/a_i t) a_i^{-1} \\ & + \sin_i^I V_j \cos(M_{i,t}) \sin(V_i/a_i t) V_i^{-1}) \end{aligned} \tag{3}$$

$$\begin{aligned} long(\sigma_i, t) = {} & (-(\Phi + w_e(t - T_0)) + \arctan((\sin_n^\Omega a_i \cos(M_{i,t}) \\ & + \cos_n^\Omega \cos_l^I a_i \sin(M_{i,t})) \cos(V_i/a_i t) a_i^{-1} \\ & + (-\sin_n^\Omega V_i \sin(M_{i,t}) + \cos_n^\Omega \cos_l^I V_i \cos(M_{i,t})) \sin(V_i/a_i t) V_i^{-1}, \\ & (\cos_n^\Omega a_i \cos(M_{i,t}) - \sin_n^\Omega \cos_l^I a_i \sin(M_{i,t})) \cos(V_i/a_i t) a_i^{-1} \\ & + (-\cos_n^\Omega V_i \sin(M_{i,t}) - \sin_n^\Omega \cos_l^I V_i \cos(M_{i,t})) \sin(V_i/a_i t) V_i^{-1})) \\ & \pmod{2\pi}. \end{aligned} \tag{4}$$

with $(\sin_i^I, \cos_i^I)$ the sinus and cosinus of the inclination of the i-th satellite and $(\sin_i^\Omega, \cos_i^\Omega)$ the corresponding notations for the RAAN. a_i is the radius of the orbit of the i-th satellite and V_i is a coefficient depending on a_i (see Mencarelli et al. 2022). w_e is the Earth rotation velocity and $\Phi \in]-\pi, \pi]$ is the rotation angle at T_0 between the inertial frame in which the satellites positions are computed, and the Earth reference frame in which the targets positions are fixed.

3 MILP Mathematical model

In the MILP formulation, we discretize the solution space by considering C different values for the satellites orbit configuration, i.e., $\sigma_{i,c}$ for $c \in [C]$. We pre-compute all the possible distances in terms of latitude and longitude between the targets and the satellite projections on the Earth surface (see Eqs. (1)–(2)), as follows:

$$\begin{aligned} & \Delta lat_{i,c,t,m} := |lat_m - lat(\sigma_{i,c}, t)| \quad \forall m \in \mathcal{X}, \forall i \in [s], \forall c \in [C], \forall t \in [T] \\ & \Delta long_{i,c,t,m} := |long_m - long(\sigma_{i,c}, t)| \cdot \cos(lat_m) \quad \forall m \in \mathcal{X}, \forall i \in [s], \forall c \in [C], \forall t \in [T] \end{aligned}$$

We introduce $s\,C$ binary variables $\pi_{i,c}$ indicating if the orbit of satellite i is in the c-th configuration ($c \in [C]$). The goal consists in finding the minimum number $s \in \mathbb{R}^+$ of satellites, as in the MINLP model, such that:

$$\sum_{\substack{t \in \mathcal{T}(k, \Delta t) \\ i \in [s]}} \xi_{i,t,m} \geq 1 \quad \forall k \in [\lfloor T/\Delta t \rfloor], \forall m \in \mathcal{X}$$

$$\begin{aligned} \xi_{i,t,m} = 0 \Longrightarrow \max \left\{ \sum_{c \in [C]} \pi_{i,c}\, \Delta lat_{i,c,t,m}, \sum_{c \in [C]} \pi_{i,c}\, \Delta long_{i,c,t,m} \right\} \\ \geq \theta_i^{max} \quad \forall i \in [s], \forall t \in [T], \forall m \in \mathcal{X} \end{aligned}$$

$$\sum_{c\in[C]} \pi_{i,c} = 1 \quad \forall i \in [s]$$
$$\xi_{i,t,m} \in \{0,1\} \quad \forall i \in [s], \forall t \in [T], \forall m \in \mathcal{X}$$
$$\pi_{i,c} \in \{0,1\} \quad \forall i \in [s], \forall c \in [C].$$

Once the previous problem is solved, one can easily reconstruct the values of the orbit parameters of each satellite with $\sigma_i^* := \sum_{c\in[C]} \pi^*_{i,c}\sigma_{i,c}$ $\forall i \in [s]$.

For further details on the way the solution space is discretized in order to guarantee that the optimal orbit configuration for each satellite is preserved, we refer the reader to our previous paper (Mencarelli et al. 2022).

4 The feasibility pump heuristic

In this section, we briefly describe the basic aspects and characteristics of the Feasibility Pump heuristic for a general Mixed-Integer Nonlinear Problem:

$$\begin{array}{ll} \min_{x,y} & f(x,y) \\ \text{s.t.} & g(x,y) \le 0 \\ & x \in X \\ & y \in Y \cap \mathbb{Z}^p, \end{array} \tag{MINLP}$$

The Feasibility Pump is a well-studied heuristic (see, for instance, Achterberg and Berthold 2007; Bertacco et al. 2007; Boland et al. 2012; De Santis et al. 2010, 2013, 2014; Fischetti and Salvagnin 2009; Hanafi et al. 2010), first introduced by Fischetti et al. (2005) for MILPs and extended to convex MINLPs by Bonami et al. (2009) and to nonconvex MINLPs by D'Ambrosio et al. (2010a) (see Belotti and Berthold 2014; D'Ambrosio et al. 2012). For an overview on the Feasibility Pump, we refer the reader to a survey by Berthold et al. (2019).

The basic idea of the Feasibility Pump is to deal with the two sources of difficulties of MINLP separately, namely the nonlinearity of the constraints and the integrality requirement for (a subset of) the variables. The Feasibility Pump bounces between two sub-problems: the Nonlinear Problem (NLP) obtained from MINLP by relaxing the constraints involving integers, and a MILP with linear and linearized constraints of the original MINLP. The Feasibility Pump produces two sequences of solutions: the first one feasible with respect to the nonlinear constraints appearing in MINLP, and the second one integer feasible. In the generic iteration, we solve first the NLP sub-problem obtaining a solution $(\tilde{x}, \tilde{y})$ which respects the nonlinear constraints, and then we solve the MILP to determine the integer solutions closest to $(\tilde{x}, \tilde{y})$.

D'Ambrosio et al. (2012) observe that the Feasibility Pump heuristic is a special case of the well-known Successive Projection Method (Bauschke and Borwein 1996), in which two sequences of points are generated by repetitively solving two different problems of the form:

$$\bar{z}^k \in \arg\min_{z \in \mathcal{A}} \|z - \widehat{z}_{i-1}\|, \qquad \widehat{z}^k \in \arg\min_{z \in \mathcal{B}} \|z - \bar{z}_{i-1}\|. \tag{5}$$

Depending on the choices of the two sets $\mathcal{A}$ and $\mathcal{B}$ several variants of the Feasibility Pump are possible. The work presented in D'Ambrosio et al. (2012) defines $\mathcal{A} := \{(x, y) : g(x, y) \leq 0, x \in X, y \in Y\}$ and $\mathcal{B} := \{(x, y) : g_c(x, y) \leq 0, x \in X, y \in Y \cap \mathbb{Z}^p\}$, where $g_c(x, y)$ represents the convex constraints in MINLP. Both the resulting problems are **NP**-hard, therefore solving them using heuristic methods is suggested.

Following the idea introduced by Achterberg and Berthold (2007) for MILP, Sharma et al. describe in (2016) a version of the Feasibility Pump in which a multi-objective optimization approach is developed considering at the same time two conflicting goals: quickly finding a feasible solution, and ensuring a good quality in terms of the original objective function. Belotti and Berthold (2017) describe a different implementation of the Feasibility Pump considering the Hessian of the Lagrangian function associated to the MINLP in the distance function in the MILP sub-problem, linearization cuts and a hierarchy of rounding procedures. Li and Liu (2017) consider inexact solving procedures for the NLP sub-problem in the Feasibility Pump method.

A version of the Feasibility Pump that does not require solving any MILP is presented by Bonami and Gonçalves (2012): the simple rounding operation of the NLP sub-problem solution to nearest integer is implemented. This version of the algorithm could cycle: whenever a cycle is detected, a random flip of several entries of the NLP sub-problem solution is implemented, similarly to the approach originally introduced in Fischetti et al. (2005).

In the original version of Feasibility Pump, a tabu list, i.e., a list of the already generated solutions, which are forbidden for the next iterations is applied in order to avoid generating the same integer feasible point twice (or no-good cuts if this method fails D'Ambrosio et al. 2010b).

5 Feasibility pump for the satellite constellation design

In order to implement the Feasibility Pump for the satellite constellation design problem described in Sect. 2, we have to specify the definition of the sets $\mathcal{A}$ and $\mathcal{B}$ and the norms occurring in Eq. (5).

5.1 MILP sub-problem

The MILP sub-problem is obtained from the MINLP formulation by discarding the nonlinear part. In particular, we introduce two new continuous variables $lat_{i,t} \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ and $long_{i,t} \in [0, 2\pi]$ designating respectively the latitude and the longitude of the satellite i at time-step t. One will notice that the introduced variables $lat_{i,t}$ and $long_{i,t}$ do not depend on the orbital parameters of the satellites: the physics constraints have been relaxed in this sub-problem. As a consequence, the feasibility of a solution will be verified at the end of the NLP sub-problem instead of the MILP one. The MILP sub-problem is expressed as follows:

$$\begin{aligned}
\min\ & \textstyle\sum_{t\in[T],i\in[s],m\in\mathcal{X}} \left|\xi_{i,t,m} - \widehat{\xi}_{i,t,m}\right| \\
\text{s.t.}\ & \textstyle\sum_{\substack{t\in\mathcal{T}(k,\,\Delta t)\\ i\in[s]}} \xi_{i,t,m} \geq 1 \quad \forall k \in [\lfloor T/\Delta t \rfloor], \forall m \in \mathcal{X} \\
& \xi_{i,t,m} = 0 \Longrightarrow \max\left\{|lat_m - lat_{i,t}|, |long_m - long_{i,t}|\cos(lat_m)\right\} \\
& \quad \geq \theta_i^{max} \quad \forall i \in [s], \forall t \in [T], \forall m \in \mathcal{X} \\
& \xi_{i,t,m} \in \{0, 1\} \quad \forall i \in [s], \forall t \in [T], \forall m \in \mathcal{X},
\end{aligned}$$

where $\widehat{\xi}_{i,t,m}$ is the solution of the NLP sub-problem (see Sect. 5.2). To solve the MILP sub-problem we reformulate the ℓ_1-norm in the objective function by introducing a new positive variable $abs_{i,t,m}$ satisfying the following constraints:

$$abs_{i,t,m} \geq \xi_{i,t,m} - \widehat{\xi}_{i,t,m} \quad \forall i \in [s], \forall t \in [T], \forall m \in \mathcal{X} \tag{6}$$

$$abs_{i,t,m} \geq \widehat{\xi}_{i,t,m} - \xi_{i,t,m} \quad \forall i \in [s], \forall t \in [T], \forall m \in \mathcal{X}\,, \tag{7}$$

and we rewrite the objective function as follows:

$$\min \sum_{t\in[T],i\in[s],m\in\mathcal{X}} abs_{i,t,m} \tag{8}$$

We do not explicitly rewrite the indicator constraints, since modern solvers, like for instance `CPLEX`, can directly deal with them.

5.2 NLP sub-problem

In the NLP sub-problem we relax the requirements for the ξ variables to be integers and we reformulate the indicator constraints with big-M terms. The NLP sub-problem is expressed as follows:

$$\begin{aligned}
\min\ & \sqrt{\sum_{\substack{t\in[T],i\in[s],\\ m\in\mathcal{X}}} \left(\xi_{i,t,m} - \widetilde{\xi}_{i,t,m}\right)^2} + \sqrt{\sum_{t\in[T],i\in[s]} \left(lat(\sigma_i, t) - \widetilde{lat}_{i,t}\right)^2} \\
& + \sqrt{\sum_{t\in[T],i\in[s]} \left(long(\sigma_i, t) - \widetilde{long}_{i,t}\right)^2} \\
\text{s.t.}\ & \textstyle\sum_{\substack{t\in\mathcal{T}(k,\,\Delta t)\\ i\in[s]}} \xi_{i,t,m} \geq 1 \quad \forall k \in [\lfloor T/\Delta t \rfloor], \forall m \in \mathcal{X} \\
& \max\left\{|lat_m - lat(\sigma_i, t)|, |long_m - long(\sigma_i, t)|\cos(lat_m)\right\} \\
& \quad \leq \theta_i^{max} + M\,(1 - \xi_{i,t,m}) \quad \forall i \in [s], \forall t \in [T], \forall m \in \mathcal{X} \\
& \xi_{i,t,m} \in [0, 1] \quad \forall i \in [s], \forall t \in [T], \forall m \in \mathcal{X} \\
& \sigma_i \in \Sigma_i \quad \forall i \in [s],
\end{aligned}$$

where $(\widetilde{\xi}_{i,t,m}, \widetilde{lat}_{i,t}, \widetilde{long}_{i,t})$ is the solution of the MILP sub-problem. In the numerical experiments, we set $M := 2\pi$, which is a valid upper bound for the left-hand side of the reformulated constraints. The NLP obtained is non-convex, hence we solve it with a MultiStart approach, namely we locally solve the problem starting from $n := 5$ randomly generated starting points in our computational experiments (see Sect. 7).

5.3 Feasibility pump algorithm

We solve the couple of previous sub-problems starting from a number of satellites $s = 1$ and progressively increasing s if an iteration limit is reached or the current problem is infeasible. When a feasible solution to the NLP problem is reached, we verify if the indicator constraints are satisfied: if so, we return the current solution (see Algorithm 1). We start the algorithm by solving the MILP sub-problem from an initial integer feasible solution.

In order to avoid generating twice the same integer solution we add the following "no-good cut" to the MILP sub-problem once a feasible solution is calculated:

$$\sum_{t\in[T],i\in[s],m\in\mathcal{X}} \left|\xi_{i,t,m} - \widetilde{\xi}_{i,t,m}\right| \geq 1. \tag{9}$$

The flow of the algorithm is summarized in the following pseudo-code:

Algorithm 1 Feasibility Pump for the satellite constellation design problem

```
Initialization: set $s := 1$ and $iter := 0$.
while $s \leq \lfloor T/\Delta t \rfloor \cdot |\mathcal{X}|$ do
    solve MILP sub-problem and obtain solution $(\widetilde{\xi}_{i,t,m}, \widetilde{lat}_{i,t}, \widetilde{long}_{i,t})$
    $(\widehat{\xi}_{i,t,m}, \widehat{\sigma}_i) := +\infty$
    for $j \in \{1, \dots, iter_{ms}\}$ do
        generate a random starting point $(\xi'_{i,t,m}, \sigma'_i)$
        solve NLP sub-problem via local solver and obtain solution $(\xi^*_{i,t,m}, \sigma^*_i)$
        if $(\xi^*_{i,t,m}, \sigma^*_i)$ is better than $(\widehat{\xi}_{i,t,m}, \widehat{\sigma}_i)$ then
            $(\widehat{\xi}_{i,t,m}, \widehat{\sigma}_i) := (\xi^*_{i,t,m}, \sigma^*_i)$
    if solution $\widehat{\sigma}_i$ $(i \in [s])$ is feasible for the original MINLP then
        return the current solution $(\widehat{\xi}_{i,t,m}, \widehat{\sigma}_i)$
    else
        if $iter = iter\_max$ or $(\widehat{\xi}_{i,t,m}, \widehat{\sigma}_i) = +\infty$ then
            $s := s + 1$
            $iter := 0$
        else
            $iter := iter + 1$
```

6 MILP-based matheuristic

In the MILP formulation of the constellation design problem that we aim to solve with a MILP-based matheuristic, we consider two kinds of binary variables:

- the $\xi_{i,t,m}$ variables already introduced to indicate if a target $m \in |\mathcal{X}|$ is observed by a satellite $i \in [s]$ at a given time-stamp $t \in [T]$;
- the $\zeta_{k,m}$ variables which are equal to 1 if the target $m \in |\mathcal{X}|$ is observed in a given period $k \in \lfloor T/\Delta t \rfloor$.

One can notice that for a feasible solution of the satellite constellation design problem, we have $\zeta_{k,m} \equiv 1$ for all $m \in |\mathcal{X}|$ and $k \in \lfloor T/\Delta t \rfloor$. Thus, in the new MILP formulation we seek to maximize the number of $\zeta_{k,m}$ variables equal to 1 by selecting a satellite orbit configuration $\pi_{i,c}$ $(i \in s, c \in [C])$.

In order to obtain a more robust solution we also consider in the objective function the $\xi_{i,t,m}$ variables, trying to maximize the number of time-stamps in which a target is observed. Finally, we opportunely re-weight the two terms in the objective function in order to balance their numeric values.

$$
\begin{aligned}
\max_{\zeta,\xi,\pi} \;& s\Delta t \sum_{\substack{k \in \lfloor T/\Delta t \rfloor, \\ m \in |\mathcal{X}|}} \zeta_{k,m} + \sum_{\substack{i \in [s], t \in [T], \\ m \in |\mathcal{X}|}} \xi_{i,t,m} \\
\text{s.t. }\; & \xi_{i,t,m} = 0 \Longrightarrow \max \\
& \qquad \left\{ \sum_{c \in [C]} \pi_{i,c}\, \Delta lat_{i,k,t,m}, \sum_{c \in [C]} \pi_{i,c}\, \Delta long_{i,k,t,m} \right\} \\
& \quad \geq \theta_i^{max} \quad \forall i \in [s], \forall t \in [T], \forall m \in \mathcal{X} \\
& \quad \zeta_{k,m} = 1 \Longrightarrow \sum_{\substack{t \in \mathcal{T}(k, \Delta t) \\ i \in [s]}} \xi_{i,t,m} \geq 1 \quad \forall k \in [\lfloor T/\Delta t \rfloor], \forall m \in \mathcal{X} \\
& \quad \sum_{c \in [C]} \pi_{ic} = 1 \quad \forall i \in [s] \\
& \quad \xi_{i,t,m} \in \{0, 1\} \quad \forall i \in [s], \forall t \in [T], \forall m \in \mathcal{X} \\
& \quad \zeta_{k,m} \in \{0, 1\} \quad \forall k \in [\lfloor T/\Delta t \rfloor], \forall m \in \mathcal{X} \\
& \quad \pi_{i,c} \in \{0, 1\} \quad \forall i \in [s'], \forall c \in [C] .
\end{aligned}
$$

The pseudo-code of the MILP-based matheuristic is given in Algorithm 2.

Algorithm 2 MILP-based matheuristic for the satellite constellation design

```
**Initialization**: set $s := 1$, $j := 1$, $J := 3$, and $flag\_feas := 0$.
find an initial (possibly feasible) solution $\sigma'$
**while** $flag\_feas = 0$ **do**
    set $iter := 1$
    fix the set of all orbital parameters but the $j$-th to $\sigma'$
    **while** $flag\_feas = 0$ **and** $iter \leq iter\_max$ **do**
        solve MILP and obtain solution $\sigma'$
        **if** $\sigma'$ is feasible for original problem **then**
            $flag\_feas := 1$
        **else**
            **if** a cycle is detected **then**
                randomly re-initialize the set of the fixed orbital parameters
            set $j := (j + 1) \mod J$
            fix the set of all orbital parameters but the $j$-th to $\sigma'$
            $iter := iter + 1$
    $s := s + 1$
**return** solution $\sigma'$
```

We first initialize the MILP-based procedure with an initial (possibly feasible), probably sub-optimal, solution to the satellite constellation design problem. In this work we chose the starting point randomly.

The MILP formulation is solved by starting from $s = 1$ satellite and by iteratively fixing all the orbital parameters but one until a better feasible constellation configuration is determined, or the maximum number of iterations is reached (Lines 6–16). If a feasible solution is not found within the maximum number of iterations, the number of satellites s of the constellation is incremented by 1, and the discretized MILP problem is solved again.

Cycles may appear in this process: to avoid this drawback, when a cycle is detected, i.e., the solution σ' has already been produced at a former iteration, the fixed satellites orbital parameters are randomly re-initialized (Lines 11–12).

It is important to note that this algorithm is an heuristic method and as such it does not guarantee the optimality of the final solution.

7 Computational results

All the developed codes have been written in Julia (version 1.8.5) and parts of the Julia library `Satellite Toolbox` (Ronan et al. 2019) related to spatial mechanics have been adapted to fit in our framework. The characteristics for the hardware we use in computational experiments are the following: MacBook Pro mounting macOS Ventura 13.4 with Apple M2 chip and 8 GB memory RAM. The solver `CPLEX 22.1.1` was used to solve the MILP problems. Note that `CPLEX` can directly deal with indicator constrants. The NLP problems, on the other hand, were solved with `Ipopt 3.14.10`, which is an exact solver for convex NLPs.

For the results presented in this work, the satellites are constrained on a Low Earth orbit, with an altitude between 400 and 1400 km. Constraining the problem to be periodic over a 24 h period, it follows that only three values for the altitude of the satellites are allowed: 566.805, 893.700, and 1261.990 km (Mencarelli et al. 2022).

The surface observed by a satellite is described by Eqs. (1) and (2), resulting in a square area projected on the surface of the Earth. This is an upgrade compared to the formulation used in our previous work in which the observed area was a trapezoidal shape shrinking in size for greater values of the latitude (Mencarelli et al. 2022).

Because of the time discretization, an overlap of two consecutively observed areas is required in order to minimize the size of unobserved areas between two consecutive time steps, which would have otherwise been observed in a continuous time paradigm. In this work the angle θ^{max} and time steps $t_{k+1} - t_k$ are designed to ensure a 20% overlap distance-wise.

The latitude of the targets were constrained between θ^{max} and $\pi/2-\theta^{max}$: this is the most difficult and hence interesting case from a computational viewpoint, otherwise the result is trivial and consists of one satellite with a polar or equatorial orbit. In the presented results, unless stated otherwise, the coordinates of the targets are summarized in Table 1.

Table 1 Latitude and longitude (deg) of the test targets

	Latitude	Longitude
Target 1	32	42
Target 2	35	134

Table 2 CPU time in seconds depending on the number of targets and the revisit time for the Feasibility Pump algorithm. The simulation time step is set to 3 min. The number in brackets is the number of satellites in the constellation

Number of targets	Revisit time		
	12 h	8 h	6 h
1	375 (1)	1296 (2)	8233 (3)
2	609 (2)	1422 (2)	5820 (3)

Table 3 CPU time in seconds depending on the number of targets and the revisit time for the MILP-based matheuristic algorithm. The simulation time step is set to 3 min. "T.L." means that a feasible solution was not found within a time limit of 4 h, but the algorithm can provide a feasible solution. The number in brackets is the number of satellites in the constellation

Number of targets	Revisit time		
	2 h	1.5 h	1 h
1	435 (2)	1249 (2)	14,286 (3)
2	662 (2)	10,193 (2)	T.L

The hyper parameters of the Feasibility Pump algorithm were set as $iter_{max} := 50$ and $iter_{ms} := 5$ (see Algorithm 1). The hyper parameter for the MILP-based matheuristic was set as $iter_{max} := 100$ (see Algorithm 2).

In order to test the performance of the developed algorithms, we solve the models for various values of the revisit time, time step, and numbers of targets. The tests were stopped when either the computer memory or a 4 h time limit was reached.

The evolution of CPU time usage of the Feasibility Pump and MILP-based algorithms, depending on the number of targets and the revisit time, are reported in Tables 2 and 3 respectively, with a time step set to 3 min.

Tables 4 and 5 show the evolution of the CPU time of the Feasibility Pump algorithm depending on the coordinates of the targets and the revisit time, for 1 and 2 targets respectively. Tables 6 and 7 are the corresponding results for the MILP-based algorithm.

In Table 8, the CPU time usage of the MILP-based algorithm depending on the time step and the revisit time are reported. In this test case, a single target is considered, with latitude and longitude equal to 30.0 and 42.0 degrees respectively.

Table 9 shows the variability of the CPU time of the MILP-based algorithm depending on its randomized initialization. Finally in Table 10, we compare the CPU time of the exact MILP formulation developed in our previous work (Mencarelli et al. 2022) and the proposed MILP-based matheuristic, depending on the revisit time, with a time step of 3 min and the same target as for Table 8.

Table 4 CPU time in seconds for 1 target depending on the target position and the revisit time for the Feasibility Pump algorithm. The simulation time step is set to 3 min. The number in brackets is the number of satellites in the constellation

Lat/Long (degree)	Revisit time		
	12 h	8 h	6 h
80/358	4 (1)	364 (2)	1676 (2)
61/116	5 (1)	363 (2)	738 (2)
33/139	1 (1)	467 (2)	1588 (2)
22/343	2 (1)	645 (2)	1106 (2)
16/340	4 (1)	673 (2)	1196 (2)
8/55	31 (1)	680 (2)	4479 (3)
1/220	2 (1)	649 (2)	1702 (2)
– 13/24	458 (2)	695 (2)	12,598 (4)
– 15/100	36 (1)	575 (2)	2125 (2)
– 24/9	332 (2)	481 (2)	5586 (3)

Table 5 CPU time in seconds for 2 targets depending on the targets position and the revisit time for the Feasibility Pump algorithm. The simulation time step is set to 3 min. The number in brackets is the number of satellites in the constellation

Lat/Long (degree)	Revisit time		
	12 h	8 h	6 h
80/388, 61/116	31 (1)	695 (2)	3042 (2)
33/139, 22/343	519 (2)	788 (2)	9446 (3)
16/340, 8/55	1001 (2)	1248 (2)	10,976 (3)
1/220, – 13/24	516 (2)	3128 (3)	5398 (3)
– 15/100, – 24/9	1508 (2)	3792 (3)	15,856 (4)

Table 6 CPU time in seconds for 1 target depending on the target position and the revisit time for the MILP-based matheuristic. The simulation time step is set to 3 min. The number in brackets is the number of satellites in the constellation

Lat/Long (degree)	Revisit time			
	6 h	4 h	3 h	2 h
80/358	4 (1)	46 (1)	427 (2)	257 (2)
61/116	1 (1)	609 (2)	567 (2)	632 (2)
33/139	97 (1)	670 (2)	684 (2)	637 (2)
22/343	37 (1)	683 (2)	681 (2)	695 (2)
16/340	3 (1)	2 (1)	2 (1)	2 (1)
8/55	4 (1)	3 (1)	2 (1)	2 (1)
1/220	4 (1)	3 (1)	3 (1)	2 (1)
– 13/24	1 (1)	2 (1)	3 (1)	3 (1)
– 15/100	1 (1)	3 (1)	3 (1)	2 (1)
– 24/9	309 (2)	684 (2)	690 (2)	562 (2)

As can be seen in Table 2, the Feasibility Pump algorithm shows major improvements compared to the MINLP algorithm it is based upon, since in our previous work, the MINLP did not yield results even for a single target and a 12 h revisit time (Mencarelli et al. 2022). However, scalability to larger problems is still a challenging issue. Compared to the "MILP-routine" algorithm presented in our previous work (which

Table 7 CPU time in seconds for 2 targets depending on the targets position and the revisit time for the MILP-based matheuristic. The simulation time step is set to 3 min. The number in brackets is the number of satellites in the constellation

Lat/Long (degree)	Revisit time			
	6 h	4 h	3 h	2 h
80/358, 61/116	7 (1)	883 (2)	860 (2)	906 (2)
33/139, 22/343	214 (1)	880 (2)	962 (2)	1071 (2)
16/340, 8/55	6 (1)	7 (1)	6 (1)	6 (1)
1/220, − 13/24	7 (1)	5 (1)	6 (1)	6 (1)
− 15/100, − 24/9	381 (1)	930 (2)	961 (2)	1014 (2)

Table 8 CPU time in seconds depending on the time step and the revisit time for the MILP-based matheuristic algorithm. A single target is considered. “T.L.” means that a feasible solution was not found within a time limit of 4 h, but the algorithm can provide a feasible solution. The number in brackets is the number of satellites in the constellation

Time step (sec)	Revisit time					
	12 h	8 h	6 h	4 h	3 h	2 h
180	0.9 (1)	1 (1)	57 (1)	442 (2)	355 (2)	435 (2)
120	3 (1)	6 (1)	744 (2)	854 (2)	803 (2)	8672 (2)
60	68 (1)	1759 (1)	10,013 (2)	T.L	T.L	T.L

Table 9 CPU time in seconds for 2 target and 2 h of revisit time for the MILP matheuristic starting from 10 different random seeds. The simulation time step is set to 3 min. The number in brackets is the number of satellites in the constellation

Random seed	CPU time
1	800 (2)
2	1324 (2)
3	1196 (2)
4	1686 (2)
5	989 (2)
7	764 (2)
8	977 (2)
9	972 (2)
10	1204 (2)
Mean	1072
St.dev.	286

Table 10 CPU time comparison between the exact MILP formulation and the MILP-based matheuristic, depending on the revisit time, for a single target. The time step is set to 3 min. The number in brackets is the number of satellites in the constellation

Method	Revisit time	
	12 h	8 h
Exact MILP	389 (1)	623 (1)
Matheuristic MILP	0.9 (1)	1 (1)

consisted in iteratively solving the discrete MILP by gradually increasing the number of satellites of the constellation) the Feasibility Pump seems to be somewhat faster and able to yield a solution for shorter revisit time constraints. However, the Feasibility Pump solution is not actually the optimum, since we were able to empirically find solutions with fewer satellites. Moreover, the algorithm struggles when additional targets are requested. On these criteria, it is actually outperformed by the "MILP-routine" algorithm. It is also important to note that the Feasibility Pump computation time can fluctuate quite significantly, as is shown by Tables 2, 4 and 5. This fluctuation seems independent from the targets coordinates since both large and small CPU times are obtained for low latitude targets. In particular, one will also notice that the Feasibility Pump does not find a solution for equatorial targets particularly fast despite the solution being trivial.

On the other hand, the results reported in Table 10 show that the MILP-based matheuristic algorithm greatly outperforms the MILP formulation it is based upon, which could not perform with a revisit time below 6 h. Moreover, Table 3 shows that the MILP-based matheuristic also yields much better results than the Feasibility Pump algorithm, being able to find a solution with more targets and shorter revisit times. One will also notice in Tables 6, 7 and 9 that the MILP-based algorithm CPU time is much more stable than for the Feasibility Pump, and problems with equatorial targets consistently yield very fast results. In addition, Table 8 exhibits that the MILP-based matheuristic also allows, up to a certain extend, to improve the time discretization. With a smaller time step, it is possible to consider a smaller and hence more realistic satellite swath α. In this case we have:

- $\alpha \approx 50.814$ degrees for a time step equal to 180 s;
- $\alpha \approx 40.469$ degrees for a time step equal to 120 s;
- $\alpha \approx 23.649$ degrees for a time step equal to 60 s.

In order to improve further the MILP-based matheuristic algorithm, we tried to replace its random initialization step with a "shortened" Feasibility Pump or Genetic algorithm which was interrupted before complete convergence. However, the obtained results were at most equivalent, and usually worse than the ones produced by random initialization because of the additional time required to run these algorithms.

8 Conclusion

In this paper, we proposed two novel matheuristic approaches for the satellite constellation design problem with discontinuous coverage and revisit time, and we tested them on a computer with limited power.

The Feasibility Pump approach allows to find a solution to the MINLP formulation developed in a previous work. Preliminary results are presented, showing that the new Feasibility Pump approach can handle more complex problems than the analytic approach described in Mencarelli et al. (2022), but scalability is still problematic.

The second matheuristic based on the discretized MILP formulation greatly outperforms both the previous MILP formulation introduced in Mencarelli et al. (2022), and the Feasibility Pump developed in this work. Although further research will be required to be able to tackle large scale problems, significant progress has been made in a promising direction.

Funding Open access funding provided by Università di Pisa within the CRUI-CARE Agreement.

References

Achterberg T, Berthold T (2007) Improving the feasibility pump. Discret Optim 4(1):77–86

Bauschke H, Borwein J (1996) On projection algorithms for solving convex feasibility problems. SIAM Rev 38(3):367–426

Belotti P, Berthold T (2014) Heuristics for nonconvex MINLP. In: 18th Combinatorial optimization workshop, Aussois

Belotti P, Berthold T (2017) Three ideas for a Feasibility Pump for nonconvex MINLP. Optim Lett 11(1):3–15

Bertacco L, Fischetti M, Lodi A (2007) A feasibility pump heuristic for general mixed-integer problems. Discret Optim 4(1):63–76

Berthold T, Lodi A, Salvagnin D (2019) Ten years of Feasibility Pump, and counting. EURO J Comput Optim 7(1):1–14

Boland N, Eberhard A, Engineer F, Fischetti M, Savelsbergh M, Tsoukalas A (2012) Bootsing the feasibility pump. C-opt 2012-03, The University of Newcastle, Callaghan

Bonami P, Cournéjols G, Lodi A, Margot F (2009) A feasibility pump for mixed integer nonlinear programs. Math Program 119(2):331–352

Bonami P, Gonçalves J (2012) Heuristics for convex mixed integer nonlinear programs. Comput Optim Appl 51(2):729–747

D'Ambrosio C, Frangioni A, Liberti L, Lodi A (2010a) Experiments with a feasibility pump approach for nonconvex MINLPs. In: Festa P (ed) Experimental algorithms. 9th International symposium (SEA), Ischia Island, Naples, May 20–22, 2010, vol 6049. Springer, UK, pp 350–360

D'Ambrosio C, Frangioni A, Liberti L, Lodi A (2010b) On interval-subgradient and no-good cuts. Oper Res Lett 38(5):341–345

D'Ambrosio C, Frangioni A, Liberti L, Lodi A (2012) A storm of feasibility pumps for nonconvex MINLP. Math Program 136(2):375–402

De Santis M, Lucidi S, Rinaldi F (2010) New concave penalty functions for improving the feasibility pump. 2010-10, Dipartimento di Informatica e Sistemistica, Sapienza Università di Roma

De Santis M, Lucidi S, Rinaldi F (2013) A new class of functions for measuring solution integrality in the feasibility pump approach. SIAM J Optim 23(3):1575–1606

De Santis M, Lucidi S, Rinaldi F (2014) Feasibility pump-like heuristics for mixed integer problems. Discret Appl Math 165:152–167

Fischetti M, Glover F, Lodi A (2005) The feasibility pump. Math Program 104(1):91–104

Fischetti M, Salvagnin D (2009) Feasibility pump 2.0. Math Program Comput 1(2–3):201–222

Hanafi S, Lazić J, Mladenović N (2010) Variable neighbourhood pump heuristic for 0–1 mixed integer programming feasibility. Electron Notes Discret Math 36:759–766

Li M, Liu Q (2017) Inexact feasibility pump for mixed integer nonlinear programming. Inf Process Lett 118:110–116

Mencarelli L, Floquet J, Georges F, Grenier D (2022) Mixed integer (non)linear approaches for the satellite constellation design problem. Optim Eng 1–22

Ronan A, Fabiano L, Arcélio C, Willer G (2019) Modeling and design of a multidisciplinary simulator of the concept of operations for space mission pre-phase: a studies. Concurr Eng 27(1):28–39

Sharma S, Knudsen B, Grimstad B (2016) Towards an objective feasibility pump for convex MINLPs. Comput Optim Appl 63(3):737–753

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.