Genetic Algorithm-Driven Scheduling for Radar-Based Satellite Tracking

Kenny Andersson[*] **, Tim Kinnunen**[†]
Swedish Defence Research Agency
Seméli Papadogiannakis, Rolf Ragnarsson
Swedish Defence Research Agency

ABSTRACT

The ability to track, catalogue, monitor, and characterise satellites is critical for maintaining space situational awareness. Radar-based sensors play a key role in this effort, offering all weather, day-and-night capabilities. However, observation planning is complex and efficiently allocating radar resources to track a large number of satellites is a non-trivial task. This paper proposes an observation planning algorithm for allocating sensor resources to measure a population of satellites within a planning duration. The objective is to formulate this as an optimisation problem and implement a genetic algorithm to produce observation schedules. Our approach keeps selection variables binary, ensuring clear and unambiguous solutions. While genetic algorithms may not always converge to the global optimum, they require less computation time, which enables more frequent schedule updates. Additionally, instead of enforcing strict uncertainty requirements, which can waste resources on satellites with already well determined orbits, soft constraints are used that aims to improve overall performance across all satellites, even if it allows for a slight increase in individual uncertainties. The soft constraints, in the form of penalty functions, allow different strategies to be defined. These can be chosen to allow for more aggressive, conservative, or opportunistic scheduling strategies. Three different scheduling methods are simulated to illustrate the optimisation, ranging from purely greedy to more dynamic. The backbone of these methods are the same, with the difference being that the more dynamic methods have the ability to reallocate tasked observations.

1. INTRODUCTION

In recent years, the importance of maintaining *space situational awareness* (SSA) has become increasingly evident, as it is essential for ensuring safe satellite operations. The rapid increase in the amount of satellites in orbit around the Earth puts increasingly high strain on existing sensor networks and requires robust optimisation of the available resources. Additionally, knowledge of satellite orbits is not only a prerequisite for operations in space but also an asset for decision-making across several operational domains, directly contributing to freedom of action and strategic autonomy [1]. Achieving this relies on accurate and timely observations, often using ground-based radar systems, and allocating the sensor resources to track a large number of satellites is a non-trivial task. This is caused not only by limited observation windows and diverse orbital trajectories but also from the necessity to balance competing priorities. The sensors used for satellite tracking can be dual-purpose, meaning that their primary functions may have to take precedence over satellite observations. For example, this could apply to certain types of military sensors or civilian sensors used for research. Furthermore, misalignment or obstructions, coupled with the decision to forfeit measurement opportunities in favour of other critical sensor tasks, inherently leads to some measurement attempts being unsuccessful.

The combinatorial complexity of this type of problem increases with the number of sensors, satellites, and measurement opportunities, making brute-force or exact optimisation approaches computationally infeasible in most practical cases. Simplification techniques, such as relaxing binary decision variables into continuous domains, are sometimes used to reduce complexity; however, these approaches may introduce ambiguity in how the resulting tasking can be interpreted [2]. *Genetic algorithms* (GAs), a type of metaheuristic optimisation technique, are well suited for problems where *sufficiently good* solutions are acceptable, within a limited computational timeframe. In the context of observation planning, where schedules could be updated on short notice and run time is a critical bottleneck, GAs provide a

[*]Corresponding and co-principal author. Research engineer, Division for Defence Technology. *kenny.andersson@foi.se*
[†]Co-principal author. Research engineer, Division for Electromagnetic Warfare. *tim.kinnunen@foi.se*

practical alternative. While they do not guarantee convergence to the global optimum, their ability to navigate large search spaces provide an acceptable trade-off between solution quality and computational effort [3].

Previous research has demonstrated that GAs, and more broadly evolutionary algorithms, can be effective in solving scheduling problems across various domains, including industrial manufacturing [4], on-board satellite tasking [5], as well as satellite instrument usage and transmission scheduling [6]. The usage of machine learning techniques to generate observation schedules is also used [7, 8, 9]. One such example is the *Coordinated Planner* (COPLA) initiative, developed by the *European Space Surveillance and Tracking* (EUSST) partnership, which aims to produce a unified observation schedule for its sensor network [10].

This paper presents a GA-driven scheduling approach designed for radar-based tracking of satellites in *low Earth orbit* (LEO). The approach aims to determine a schedule for observing a set of satellites that minimises the work needed to determine their orbits to within a predefined threshold. Although this paper focuses on using a radar to measure satellites in LEO, the same concepts can likewise be applied to other sensors and orbital regimes. Building on the work done in [11], this paper solves the optimisation problem without relaxation of variables, and introduces additional aspects such as prioritisation of satellites, measurement conflicts, and sensor resource allocation. The simulated radar used in this work is assumed to have a primary objective other than SSA, and we therefore wish to optimise our sensor usage to allow it to complete its primary tasks while still maximising the SSA usage. Another consequence of such a sensor is that its availability can change on short notice.

To illustrate our algorithm, three different planning methods were implemented and compared against each other. The first and least dynamic method, called metod A, is hierarchical, and prioritises the observation of satellites based on their weighting. The second method, B, is conservative and is designed to minimise the utilisation of sensor resources while still aiming to reach the desired threshold. The third and most dynamic method, C, is redistributive, balancing resource allocation across satellites with varying priority levels. It achieves this by reallocating measurement opportunities initially assigned to one satellite to another when the former's uncertainty reaches a certain threshold. All methods have a simple over planning mechanism to compensate for the risk of failed observations where each satellite is initially planned for one more slot than what the optimisation deems necessary. For the last two methods, the schedule is dynamically changed as time passes and redundant taskings may be moved to another satellite or removed completely. These methods are designed to illustrate the flexibility of the scheduling algorithm across different scenarios, rather than achieving optimal performance. Additionally, they are designed with a dual-use sensor in mind, and we therefore want to maximise the information gain while minimising the usage of the sensor, which is done by introducing the threshold parameter and observation cost.

This paper is organised as follows: Section 2 formulates and describes the sensor planning problem. Section 3 presents the *observation planning algorithm* (OPA), including the optimisation problem and the simulation setup. Results are shown and interpreted in Section 4. Section 5 provides a summary and concludes the paper with a future outlook.

2. PROBLEM FORMULATION

Sensor planning is an umbrella term for the methods used to optimise the utilisation of one or more sensor resources in a way that maximises the desired information, given various constraints, such as the quality of the measurement. These methods are typically based on information theory, where sensor data and the desired information is modelled as stochastic variables. In this way, the sensor planning problem can be formulated as an optimisation problem in which the entropy of the desired variables is minimised [12].

This work focuses on applying observation optimisation techniques on radar sensors with limitations on sensor availability to produce an observation schedule that reduces the uncertainty in the states of selected satellites, such that a predefined cost is minimised. To achieve this, it is necessary to define: 1) the cost, 2) the uncertainty in the state of a satellite, 3) the sensors, and 4) the scheduling interval.

The cost represents an abstract value that quantifies both the general effort associated with performing an individual measurement and the penalty incurred when a predefined accuracy requirement is not met. The measurement cost itself can vary depending on factors such as sensor type and observation conditions [11]. The cost also relates to the usage of a sensor network, as an optimal algorithm will not task an observation if the tasking cost is greater than the theoretical gain of the observation.

The aim is to estimate the state of a satellite, $\mathbf{x}(t)$, for a given schedule. As shown in [11], the estimated parameter uncertainty is described by a covariance matrix, $\mathbf{P}$, where the estimated parameters in $\mathbf{x}(t)$ lie within this ellipsoid with some defined probability α. The state vector is defined by:

$$\mathbf{x}(t) = \begin{bmatrix} \mathbf{r}(t) & \mathbf{v}(t) & \mathbf{p} & \mathbf{q} \end{bmatrix}^T, \tag{1}$$

where $\mathbf{r}$ and $\mathbf{v}$ are vectors representing the satellite's position and velocity respectively, while $\mathbf{p}$ and $\mathbf{q}$ are parameters relating to the force and measurement models. In this work, focus is specifically put on the positional component, $\mathbf{r}$, where one standard deviation is considered as the defined probability α for the confidence ellipsoid.

A generic L-band radar is modelled, located in southern Sweden, oriented northward, using an *active electronically scanned array* (AESA) antenna with a rectangular *field of view* (FoV). It is assumed that the radar could perfectly track any satellite within its FoV and more complex atmospheric effects are ignored. This setup represents an idealised case, designed to illustrate the methodology rather than capture all real-world complexities.

The analysis focused on LEO satellites in high-inclination orbits ($i > 65°$) with lower eccentricity ($e < 0.1$), to filter satellites that may not be visible to the sensor. Under these conditions, individual satellites are observable approximately 1 – 4 times per day, with each passage lasting anywhere from a couple of seconds to several minutes. The problem is limited to catalogue maintenance, i.e., no initial orbit determination is performed.

3. ALGORITHM DEFINITION

A high-level sensor planning process is illustrated in Fig. 1. First, relevant satellites are selected from a database, followed by a visibility analysis and *signal-to-noise ratio* (SNR) simulations. The simulation results from this analysis are then used by the OPA to generate an observation schedule for a chosen planning duration. Measurements are subsequently conducted according to the observation schedule which are then used for orbit determination. The new orbital parameters are then used to update the catalogue.

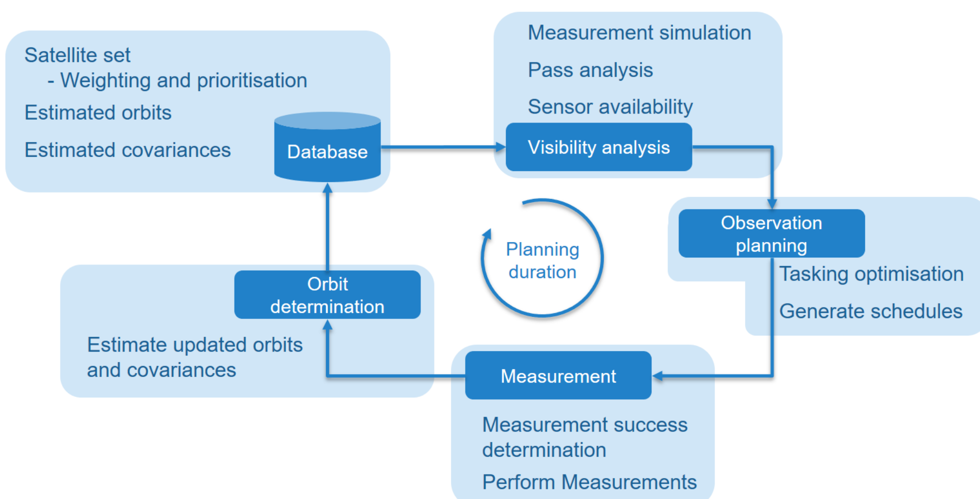

Fig. 1: High-level sensor planning diagram.

3.1 Satellite Selection and Visibility Analysis

The selection of satellites was made from publicly accessible sources [13]. A total of 519 satellites were chosen. After the selection process, each satellite was assigned a priority number, which can be updated when information is obtained, such as events that may have an affect on the satellite's significance. The priority number indicates the satellite's significance on a scale from one to five (in our case), where five represents the highest priority. A satellite's priority number can be influenced by factors such as its orbit, owner or operator, publicly stated or believed purpose, and risk factors. For this work, a priority number was randomly assigned to each satellite only to illustrate the methodology.

Before conducting the visibility analysis, the priority order is updated. The priority order is ranked based on the satellite weighting, which is meant to quantify the importance of performing a measurement on a specific satellite. The satellite weighting is calculated as according to (2):

$$w = \sum_{i=1} k_i f_i(q_i), \tag{2}$$

where q_i represents a parameter associated with each satellite, which can either be a constant property or a parameter that changes over time. Each parameter has a normalisation function f_i, which satisfies $0 \leq f_i(q_i) \leq 1$. The weighting coefficient k_i is a measure of the importance of the corresponding parameter and satisfies: $0 \leq k_i \leq 100$ and $\sum_i k_i = 100$. Two parameters were used in the simulations: **time since the last measurement** (q_1) and the **priority number** (q_2). This can be expanded to account for more parameters. The q_1 parameter allows a satellite that has not recently been observed to have a higher chance of being scheduled, and makes it more difficult for unforeseen events such as manoeuvres to go undetected.

For f_1, the rationale is that a low q_1 (i.e., a recent satellite measurement) corresponds to a decreased need for a new measurement, whereas a larger elapsed time since the last measurement increases the necessity for a new one. This effect is also captured in the growth of the uncertainty, but this also takes into account that a satellite that has not been observed recently may have performed a manoeuvre. A sigmoid function can capture this behaviour, as it provides a way to describe the transition from low to high, with the steepness, ψ, and the time of transition, t_p:

$$f_1(q_1) = \frac{1}{1 + e^{-\psi\left(q_1 - \frac{q_{1,max}}{t_p}\right)}}, \quad 0 \leq q_1 \leq q_{1,max}, \tag{3}$$

where $q_{1,max}$ is the maximum allowed time since the last measurement. The definition of f_2 is intended to capture the progressive increase in priority of the satellites. A linear equation implies that the incremental difference between adjacent priority levels remains constant, whereas a quadratic equation would reflect the idea that satellites with higher priority demand increasingly stringent conditions or prerequisites to be met. In this scenario, we used a quadratic formulation for f_2:

$$f_2(q_2) = \left(\frac{q_2}{q_{2,max}}\right)^2, \quad 1 \leq q_2 \leq q_{2,max}, \tag{4}$$

where $q_{2,max}$ is the maximum priority number, which is set to 5. The relative importance of q_1 and q_2 is incorporated when defining k_1 and k_2. The ratio of these coefficients determines to which extent high-priority satellites can jump the queue. In other words, if k_2 is set too low, the impact of prioritisation diminishes, while if k_2 is too high, lower-priority satellites may be neglected.

The visibility analysis determines the time intervals during which a satellite is within the FoV of the radar. During each passage, SNR values are calculated based on the satellite's orbit and sensor parameters. These SNR values are subsequently used by the OPA to optimise the timing of measurements, thereby maximising the information content retrieved from each measurement. The retrieved information content can be conceptualised as a measurement that quantifies the reduction in uncertainty regarding the satellite's state. This is explained in-depth in [11] where this is referred to as an alteration of the confidence ellipsoid. The reduction in uncertainty depends on several factors, including the SNR and the geometry between the radar and the satellite. In particular, the relative geometry is important, as measurements along a specific axis (e.g., the radial direction) primarily contribute to reducing uncertainty along that axis, while offering limited information about orthogonal axes. Ansys' *Systems Tool Kit* (STK) built-in access calculations feature was used to determine the satellite visibility and calculate the SNR values.

3.2 Observation Planning

The problem involves a sensor with primary objectives other than SSA, which is something that has to be accounted for. This is done by designing the optimisation problem and tasking strategies in such a way that the sensor is not used for satellite observations at all available times. In essence, this means that one wishes to optimise their timings for when a measurement is scheduled in order to reduce the total required tasking, allowing the sensor to perform other tasks. An uncertainty threshold, T, is introduced to define a maximum uncertainty for when a satellite's estimated state is acceptable. A satellite with uncertainty below this threshold is deemed sufficiently observed for the simulated duration.

The observation schedule generated by the OPA depends on the scheduling strategy. The three different strategies are referred to as method A, B, and C. All three methods are based on the same premise: during each planning period, a single schedule is generated for each satellite, and measurement opportunities are allocated sequentially according to satellite weighting. An additional measurement is scheduled to mitigate the risk of unsuccessful measurements and to detect manoeuvres, which might otherwise have gone undetected for longer. This way, satellites with higher weightings are allocated more sensor resources, meaning that the scheduling process is inherently hierarchical. This type of scheduling is greedy, and may not always be practical for problems where several sensors are used [2]. This hierarchical process describes method A. In contrast, methods B and C extend this approach by altering the observation schedule when the uncertainty falls below a specified threshold, T. In method B, if the uncertainty is below T, all subsequent measurement opportunities are cancelled, seeking to minimise the sensor utilisation by using the feedback from the observation outcomes. In method C, the cancelled measurement opportunities are instead opportunistically reassigned to the next highest weighted satellite, provided the information gained by the new measurement exceeds a predefined threshold. The allocation of measurement opportunities is ultimately determined by solving an optimisation problem that decides which opportunities to use and which to omit.

3.2.1 Optimisation Problem

Each measurement opportunity can be defined as a binary decision: (1) a measurement shall be made or (0) a measurement shall not be made. This is encapsulated by the selection variable, s_j:

$$s_j \in \{0,1\}, \quad j = 1,2,...,N, \tag{5}$$

where N is the number of possible measurements for a given scheduling period, which may differ between satellites. Furthermore, each measurement has an associated *cost*, which represents the effort or resources required to perform the measurement. Let c_j represent the cost of performing measurement j. The total cost C_m of all scheduled measurements for a specific satellite can then be expressed as [11]:

$$C_m(\mathbf{s}) = \sum_{j=1}^{N} c_j s_j, \quad \mathbf{s} = \left[s_1, s_2, ..., s_N\right]. \tag{6}$$

The optimisation problem thus becomes a matter of minimising the cost while ensuring the objective function satisfies a given accuracy requirement. Treating the selection variable as a binary decision transforms the problem into a combinatorial one of complexity 2^N. To mitigate this, one approach is to relax the binary constraint, allowing the selection variable to take values in the continuous range $(0 \leq s_j \leq 1)$ [11]. However, this relaxation introduces ambiguity, as fractional values can be difficult to interpret, suggesting either a partial measurement or a probabilistic decision to measure. Furthermore, if there is an overlap of measurement opportunities when multiple satellites are taken into account, this relaxation may yield unclear outcomes. If the decision variables for all visible satellites are low (due to the constraint $\sum_k s_{j,k} \leq 1$, for all k simultaneously visible satellites [11]) it becomes uncertain whether no satellite is worth measuring or if all satellites are worth measuring, leading to an unclear prioritisation of measurements.

Instead of relaxing the binary constraint on the selection variable, a GA can be used to solve the optimisation problem. This approach allows us to find a solution within a reasonable amount of time, even when N is large. However, it is important to note that this method is not guaranteed to find the optimal solution, as the algorithm may converge to a *local* optimum instead of the *global* optimum. In addition, the strict accuracy requirement to reduce the uncertainty to within the confidence ellipsoid can be relaxed by replacing *hard* constraints with *soft* constraints. Hard constraints, such as requiring that a satellite's uncertainty in its position remains below a specified threshold, may not always be achievable within a given time interval, even if all possible measurements are used. Moreover, imposing strict uncertainty requirements can be counterproductive. Allocating sensor resources to marginally reduce the uncertainty of a reasonably well-known satellite orbit may come at the expense of opportunities to reduce the uncertainty for other satellites. Soft constraints address this limitation by allowing the algorithm to aim for the best possible outcome overall, even when the desired result cannot be fully achieved. To implement soft constraints, a *penalty function* is introduced and (6) is reformulated as:

$$C(\mathbf{s}) = \sum_{j=1}^{N} s_j c_j + p(x(t,\mathbf{s})), \tag{7}$$

where $p(x(t,\mathbf{s}))$ is the penalty cost accrued by executing the schedule $\mathbf{s}$. Additionally, $x(t,\mathbf{s})$ represents a function that relates the target variable to the threshold at time t. A simple linear definition of the penalty function can be expressed as:

$$p(x(t,\mathbf{s})) = \begin{cases} 0, & \text{if } x(t,\mathbf{s}) \leq T \\ k_p \sum_{t=t_0}^{t_{max}} (x(t,\mathbf{s}) - T), & \text{if } x(t,\mathbf{s}) > T, \end{cases} \tag{8}$$

where T is the uncertainty threshold (e.g., $T = 1$ km), k_p is a scaling parameter used to tune the penalty cost, and the values of t_0 and t_{max} are user-defined paramters relating to the time interval taken into account when computing the uncertainty. The process of how to define them is further detailed later in this section. Intuitively, this means that as long as $x(t,\mathbf{s})$ is below the threshold T, no penalty is incurred. However, if $x(t,\mathbf{s})$ exceeds T, the penalty increases linearly. This provides an incentive for the OPA to ensure that $x(t,\mathbf{s})$ remains below the threshold to avoid additional costs, while still allowing for flexibility if the threshold cannot be met, as exceeding it increases the cost linearly. If the objective is to minimise the satellite's positional uncertainty, $x(t,\mathbf{s})$ can be defined as:

$$x(t,\mathbf{s}) = \sqrt{\mathrm{tr}(\mathbf{P}_{\mathrm{pos}}(t,\mathbf{s}))}, \tag{9}$$

where $\mathbf{P}_{\mathrm{pos}}(t,\mathbf{s})$ is the 3×3 submatrix corresponding to the positional covariance extracted from the full state covariance matrix at time t for a given schedule $\mathbf{s}$. This parameter will be referred to as the root trace covariance. Alternative formulations can be used to quantify the positional uncertainty, which can also include other quantities such as the velocity uncertainty or a combination of multiple uncertainty metrics.

If the sum in (8) sums the target variable function computed at various times throughout the entire planning interval, the algorithm will adopt a more aggressive strategy, where it will try to eliminate transients or other temporary effects that could cause the uncertainty to exceed the threshold. If the evaluation only considers the last orbit within the planning period, these earlier transient effects are ignored, but the final uncertainty is still minimised. The number of time steps selected affects the overall cost. If too many are included, the total cost from the penalty function will increase, which may necessitate an adjustment of the measurement costs, c_j, to maintain the behaviour of the scheduling strategy.

If only the uncertainty at the end of the planning cycle is of importance, the parameters t_0 and t_{max} can be defined as regularly sampled time points within the final orbit, or alternatively, the summation can be replaced with the maximum value over the final orbit. In this work, t_0 and t_{max} are set to span the final orbit for each satellite. This choice results in a more lenient scheduling strategy, not necessarily prioritising early measurements and instead aiming for convergence by the end of the simulation. This approach may miss some effects that are apparent for longer simulation durations, or effects that are apparent outside the computation window, but is computationally faster than considering longer durations.

The optimisation problem thus becomes a matter of solely minimising the total cost, which inherently accounts for both the measurement cost and the penalty associated with failing to meet the desired constraint:

$$\begin{aligned} \min_{\mathbf{s}} \; & C(\mathbf{s}) = \sum_{j=1}^{N} s_j c_j + p(x(t,\mathbf{s})) \\ & \text{s.t. } s_j \in \{0,1\}, \quad j = 1,2,...,N. \end{aligned} \tag{10}$$

This approach optimises for, and takes into account, only one satellite at a time. If satellite ω's observation opportunities overlap with those of other satellites $\Omega \in \{1,2,...,\omega-1\}$, the total cost must account for the measurement allocations of all overlapping satellites. To address this, a simple solution is to optimise for one satellite at a time, as defined in (10), in order of their weighting, as given by (2). For each satellite, the available measurement opportunities are determined by removing any opportunities already allocated to higher-weighted satellites. This means that each satellite is optimised sequentially, with higher-weighted satellites reserving their measurement opportunities first. As a result, the OPA systematically resolves conflicts by prioritising measurement opportunities for higher-weighted satellites while lower-weighted satellites only use the remaining, non-conflicting opportunities.

This approach assumes that the sensor can track only one satellite at a time, which may be a conservative assumption given the capabilities of an AESA antenna. Furthermore, this also assumes that there is only one single sensor available. The optimisation problem stated in (10) can be extended to account for more than one sensor as detailed in [11]. However, to simplify the problem and focus on the core aspects, the analysis is limited to one sensor tracking one satellite at a time.

The optimisation problem can be expanded to consider a larger number of satellites during the optimisation process. This is achieved by dividing the full satellite set into i distinct subsets (for instance based on their weightings), where each respective subset contains L_i satellites. Large subsets will yield large schedules which will increase the computational complexity. To reflect this, (10) can be reformulated as:

$$\begin{aligned} \min_{\mathbf{s}} \; C(\mathbf{s}) = \sum_{h=1}^{L_i} w_h \left(\sum_{j=1}^{N_h} s_{h,j} c_{h,j} + p(x(t, \mathbf{s}_h)) \right), \\ \text{s.t. } s_{h,j} \in \{0,1\}, \quad j = 1,2,...,N \\ h = 1,2,...,L_i \end{aligned} \tag{11}$$

where w_h is the weighting of satellite h in subset i, $\mathbf{s} = [\mathbf{s}_0, ..., \mathbf{s}_{L_i}]$, and the rest of the parameters are analogous to those in (10) but specific for satellite h. The optimisation can then be performed similarly to (10), but for one subset at a time instead of one satellite at a time, which will give more coordinated and synergetic scheduling. Each optimisation will have no knowledge of the lower weighted subsets, but will be affected by the subsets that came before since it inherits their planned observations. Equation (10) is a special case of this where $L_i = 1$.

3.2.2 Penalty functions

In this work, the linear penalty function defined in (8) was used, where no additional cost is incurred once the desired constraint has been met. Using an exponential penalty function would make the algorithm more aggressive when the objective function deviates significantly from the threshold, while being more lenient for values closer to or near the threshold. Alternatively, allowing negative costs below the threshold may make the algorithm more opportunistic, which could incentivise the algorithm to overplan, prioritising measurements even when the constraint is satisfied. Fig. 2 shows some alternative penalty functions that can be used. Each penalty function introduces a scheduling behaviour. Scaling the penalty term so that its magnitude is comparable to the measurement cost may be needed depending on the strategy used. Balancing the penalty and measurement cost allows both terms to interact more intuitively during the optimisation. Using a steeper penalty function often results in measurements that contain less information to be scheduled. Alternatively, a lower gradient function will only schedule the opportunities that contain more information and therefore schedule less aggressively. Proper tuning ensures the scheduler behaves as intended, whether it prioritises opportunism or conservative planning.

A more aggressive penalty function will prioritise observing higher priority targets, as it is more likely to schedule observations for these and therefore leave the lower priority targets with fewer available slots. The penalty functions shown in Fig. 2 have been applied to a set of 500 satellites to demonstrate their differences, and is shown in Fig. 3. The linear penalty, which is not as aggressive as the quadratic or exponential functions for high uncertainties, does not schedule as aggressively for the high priority targets resulting in a larger average uncertainty. The lower priorities, however, will have more free slots that can be scheduled and therefore their uncertainties are lower compared to the more aggressive functions. Overall the uncertainites are similar for each penalty function. The difference is how they treat the priority groups. These figures show the simulation for one day and for 500 satellites, and as such the uncertainties have not converged enough for the exponential function to clearly show its opportunism. Nevertheless, the figures help illustrate the effects of each penalty function on the scheduling outcomes.

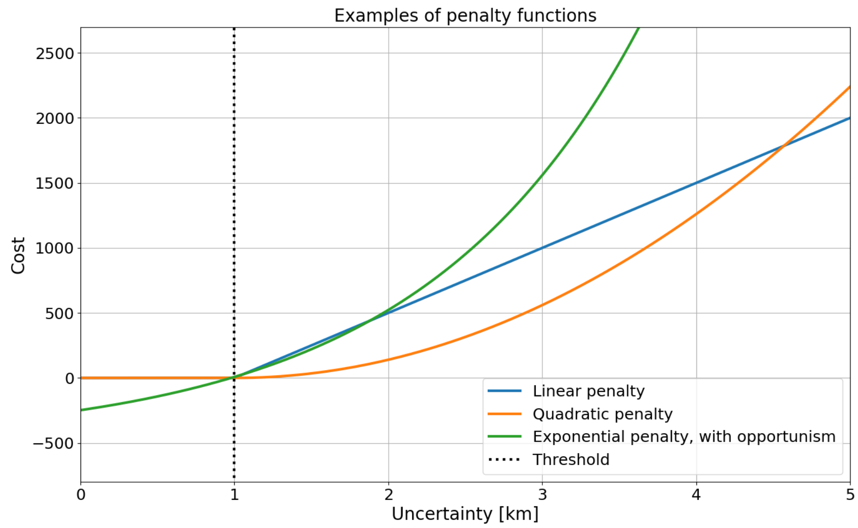

Fig. 2: Examples of different penalty functions.

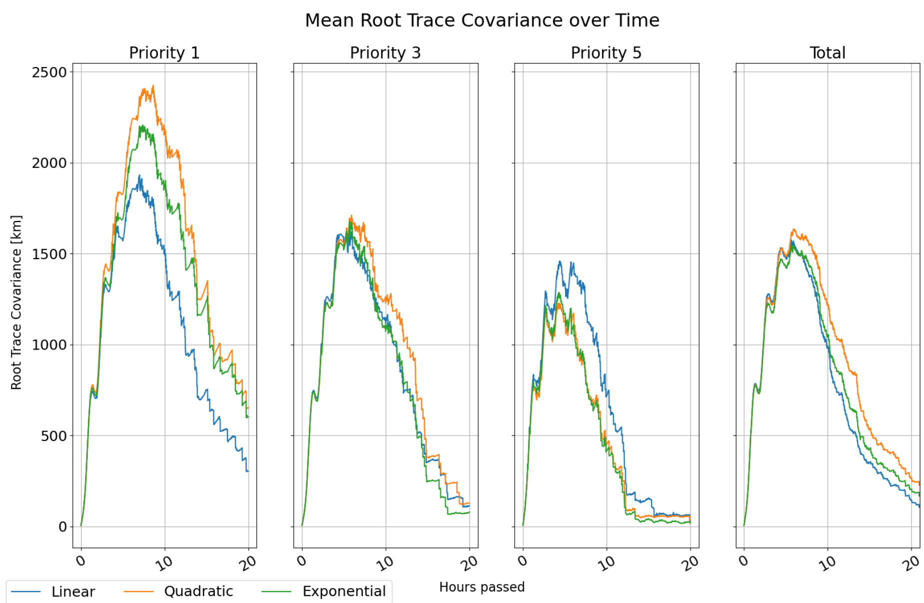

Fig. 3: Example with linear, quadratic and exponential penalty functions applied to 500 satellites and sorted by priority to show how the penalty function affects the behaviour of the uncertainties. The linear penalty is the same as in (8) and is scaled using $k_p = 5\text{E}{-4}$, the quadratic is similar to (8) but with an exponent of 2 and is scaled using $k_p = 5\text{E}{-8}$. The exponential function is tuned similarly to the quadratic and has opportunism.

3.2.3 Genetic Algorithm

GAs are a class of evolutionary algorithms inspired by the principles of natural selection. They are particularly effective for solving optimisation problems involving large search spaces, where traditional methods are computationally infeasible or would require excessive run times. The trade-off is that GAs may only find near-optimal solutions [3].

GAs work by iteratively refining a population of candidate solutions, represented as chromosomes, through selection, crossover, and mutation operators. Each chromosome constitutes a potential solution, with individual genes representing decision variables [3]. In the context of this work, a chromosome represents a unique observation schedule, where each gene corresponds to a decision about whether to use a specific measurement opportunity:

$$\mathbf{y} = [y_1, \cdots, y_j, \cdots, y_N], \tag{12}$$

where $y_j \in \{0, 1\}$ is the j:th gene in the chromosome $\mathbf{y}$. The initial population is generated by randomly creating a set of M chromosomes [3]:

$$population = \left[\mathbf{y}_1 \ldots \mathbf{y}_k \ldots \mathbf{y}_M\right]^T \tag{13}$$

where $y_{k,j}$ is the j:th gene in the k:th chromosome. The *fitness* F of each chromosome is evaluated using a cost function defined by the inverse of the objective function. Chromosomes with lower costs are considered more fit and are more likely to be selected for reproduction.

The next step involves producing the new generation, which consists of a combination of chromosomes from the previous generation and newly generated chromosomes. The proportion of parents to children is determined by the ratio $N_{p/c}$. From the parent population, chromosomes are selected pairwise with a crossover probability P_C to exchange genetic material and produce children. There are various methods for performing crossover, but in this work, *uniform crossover* was used. In this method, crossover points are randomly selected along the chromosome $\{1, ..., N\}$, where genes are exchanged [3]. All children then undergo mutation, where individual genes are randomly changed with some probability P_m. In the binary case, this means flipping the value of a gene.

The new generation then undergoes the same processes as the previous one, repeating until the termination criteria are met. In our case, the termination criteria are defined as either a maximum number of iterations, N_I, or a maximum number of iterations without any improvement, $N_{I,S}$.

3.3 Radar Measurements

Rather than using a high-fidelity radar simulator to produce measurements based on the observation schedule, the SNR values derived from the visibility analysis were directly used. These values were assumed as sufficiently reliable for input into the orbit determination process. Swerling case IV [14] was used to simulate the *radar cross section* (RCS) which affects the SNR through the received signal strength. The measurements were simulated, based on the true value of the observed parameter, where the measurement accuracies (one standard deviation from the true value) are approximated as [15]:

$$\sigma_r(t) = \frac{c}{2B_R\sqrt{2SNR(t)}} = \frac{\Delta r}{\sqrt{2SNR(t)}}, \tag{14a}$$

$$\sigma_{\dot{r}}(t) = \frac{\lambda}{2T_I\sqrt{2SNR(t)}} = \frac{\Delta \dot{r}}{\sqrt{2SNR(t)}}, \tag{14b}$$

where, c is the speed of light, B_R is the radar receiver processing bandwidth, λ is the wavelength and T_I is the radar integration time. Each measurement is assumed independent of all others and has a probability of either succeeding or failing. This probability is determined by two factors: the probability that the sensor is available, P_a, and the probability that the measurement itself succeeds, P_s. The sensor availability, P_a, accounts for scenarios where the sensor is not solely dedicated to satellite tracking but is instead used for other purposes without knowledge in advance. The measurement success probability, P_s, reflects the potential for failure due to factors such as obstructions (e.g., aerial vehicles, terrain, natural or man-made objects), or technical problems. The process for each measurement involves two sequential checks. First, a check is performed to determine whether the sensor is available. If the sensor is available, a second check determines whether the measurement succeeds. If a measurement fails, the corresponding time slot remains unavailable, indicating that an attempt was made to measure during that period, even though it did not succeed.

3.4 Orbit Determination

Each satellite is propagated using Keplerian two-body dynamics to determine both possible measurement times and the measurement information. Orbit determination is performed using the differential correction method, as described in [16]. The covariance matrix, $\mathbf{P}(t)$, at a given time is obtained by incorporating measurements and propagating the resulting covariance to the time of interest. This yields:

$$\mathbf{P}(t)^{-1} = \sum_{i=1}^{N} \mathbf{A}(t_i)^T \mathbf{W}(t_i) \mathbf{A}(t_i), \tag{15}$$

where t_i is the time of measurement i. The matrix $\mathbf{W}$ is a weighting matrix with diagonal elements given by the inverse squared standard deviations from (14):

$$\mathbf{W}(t_i) = \begin{bmatrix} \sigma_r(t_i)^{-2} & 0 \\ 0 & \sigma_{\dot{r}}(t_i)^{-2} \end{bmatrix}, \tag{16}$$

and $\mathbf{A}(t_i) = \Phi(\Delta t_i)\mathbf{H}(t_i)$, with $\Phi(\Delta t_i)$ representing the state transition matrix from the measurement time t_i to the epoch. The term:

$$\mathbf{H}(t_i) = \left. \frac{\partial \text{observations}}{\partial \hat{\mathbf{x}}_i} \right|_{\hat{\mathbf{x}}_i = \mathbf{x}_i} \tag{17}$$

represents the observation partial derivative matrix [16], which depends on the state at the measurement time, $\hat{\mathbf{x}}_i$, and the observables. For a radar sensor, the measured quantities are the range, r, and range rate, $\dot{r}$, yielding:

$$\frac{\partial \text{observations}}{\partial \hat{\mathbf{x}}_i} = \begin{bmatrix} \frac{\partial r}{\partial x_i} & \frac{\partial r}{\partial x_j} & \frac{\partial r}{\partial x_k} & \frac{\partial r}{\partial v_i} & \frac{\partial r}{\partial v_j} & \frac{\partial r}{\partial v_k} \\ \frac{\partial \dot{r}}{\partial x_i} & \frac{\partial \dot{r}}{\partial x_j} & \frac{\partial \dot{r}}{\partial x_k} & \frac{\partial \dot{r}}{\partial v_i} & \frac{\partial \dot{r}}{\partial v_j} & \frac{\partial \dot{r}}{\partial v_k} \end{bmatrix}. \tag{18}$$

Some radar sensors can also measure the azimuth and elevation angles. These derivatives can be computed either numerically or using analytical methods. This formulation provides the covariance matrix at a given epoch, which can then be used to optimise measurement timing and selection. Furthermore, these equations can be extended to incorporate additional sensor types, such as optical sensors, by including their respective measured quantities in $\mathbf{H}(t_i)$ and $\mathbf{W}(t_i)$. The Clohessy-Wiltshire equations [17] are used to model the dynamics for propagating covariance matrices.

3.5 Simulation Setup

The simulation follows the flowchart illustrated in Fig. 1 and is performed for the selected 519 satellites over seven 24-hour cycles, starting from 2025-02-07, as this increases the probability of each satellite being visible at least once per scheduling interval. Satellite data, including orbital parameters, and other data, are stored in *orbit mean-elements message* (OMM) files [18]. For our purposes, these files are extended to include the covariance matrix and its epoch, information about the RCS, and the priority number.

For the first iteration, if no data on the RCS, covariance matrices, or the priority numbers was available, these values were randomly generated. The RCS value was generated based on the RCS size specified in the OMM file, which was categorised as *small*, *medium*, or *large*. This classification follows the convention used by *Space-Track*[1], where a small RCS is defined as $RCS < 0.1\ m^2$, medium as $0.1 \le RCS < 1.0\ m^2$, and large as $RCS \ge 1.0\ m^2$. If the OMM file does not specify an RCS size, one was randomly assigned to the satellite. Priority numbers were generated from a discrete uniform distribution, $U\{1,5\}$ and covariance matrices were generated in the Hill frame based on a normal distribution with standard deviation 10 $\mathrm{m^2/s^2}$ and 18 $\mathrm{km^2}$ for the diagonal velocities and positions, respectively. The off-diagonals were generated similarly but scaled down by a factor of 10. The matrices were not generated to be realistic, but rather to give each satellite an initial covariance above the uncertainty threshold.

Before the visibility analysis, the priority order of the satellites was updated based on their weightings, as defined in (2). This involves calculating the time since the last measurement, q_1, using the current time in the simulation loop and the epoch of the OMM. The priority number, q_2, was extracted from the modified OMM.

At the end of each day, an observation schedule was produced by the OPA, once for each method. The OPA requires defining a threshold for the penalty function, along with the GA parameters, simulation step size, dt, and satellite

[1] https://www.space-track.org/documentation/loadLegendRCS

and sensor information. The simulation step size sets the minimum time between consecutive measurements. The measurements were then conducted according to the generated schedule, where the probability that the sensor is available and the probability that the measurement succeeds are required. To enable a comparable analysis between the different methods, the same seed was used for randomising measurement outcomes based on the time of the measurement. This way, a measurement at the same time in method A, B or C will always have the same outcome. This eliminates random variations and ensures that only the effects of the methods are compared. After generating the schedules and determining the measurements success, updated covariance matrices were estimated, and new OMM files were created to serve as inputs for the next day's simulation loop.

The OPA was implemented in Python and the Python STK API was used for the visibility analysis and SNR calculations. The Orekit Python wrapper [19] was used in the OPA to propagate the satellites' orbit, and to calculate the derivative matrix in (18) where relevant. Simulations were performed on a laptop with an i5-1335U processor on a single thread.

The parameters used for the GA and in the simulations are summarised in Table 1.

Table 1: User defined GA and simulation parameters.

Parameter	**Variable**	**Value**
Population size	M	100
Number of elites	N_E	1
Parents to children ratio	$N_{p/c}$	0.30
Crossover probability	P_C	50 %
Mutation probability	P_m	10 %
Maximum iterations	N_I	3000
Maximum iterations without improvement	$N_{I,S}$	100
Measurement availability probability	P_a	98 %
Measurement success probability	P_s	95 %
Radar range accuracy	Δr	70.7 m
Radar range rate accuracy	$\Delta \dot{r}$	28.3 ms^{-1}
Step size	dt	40 s
Measurement cost	c	500
Number of satellites	N_{sat}	519
Uncertainty threshold	T	1000 m
Planning duration	d	86400 s
Penalty cost scaling	k_p	5
Sigmoid steepness factor	ψ	$1.2\text{E}-4$
Sigmoid transition time	t_p	2
Maximum allowed time since last measurement	$q_{1,max}$	86400 s
Maximum priority	$q_{2,max}$	5
q_1 weighting	k_1	65
q_2 weighting	k_2	35

4. SIMULATION RESULTS AND DISCUSSION

Fig. 4 shows how the runtime of the optimisation problem scales with the number of satellites for different penalty functions. The sensor gets increasingly more saturated as more satellites are added to the simulation, which leads to a reduced amount of available slots for lower weighted satellites, resulting in a time complexity that is logarithmic in shape. More conservative penalty functions need more time to run as it takes longer for the sensors to get saturated.

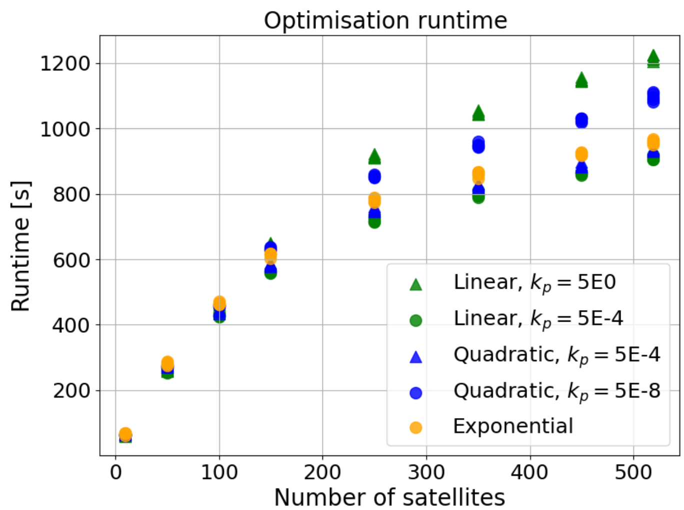

Fig. 4: Runtimes for different sized sets of satellites and penalty functions. Five runs are shown for each penalty function to illustrate the variance. For the linear and quadratic penalty function, the penalty cost scaling shown in (8) is varied to show how it affects runtime. The exponential function has opportunism and is scaled similarly to the quadratic function with $k_p = 5\text{E}{-8}$.

Using the configuration detailed in Section 3.5, three simulations were performed, one for each method. Fig. 5 shows the median root trace covariance for all satellites, computed using (9), and how it decreases as new measurements are available and the propagation of uncertainty for the different methods. Although they seem to behave similarly on a long timescale, the more dynamic method C barely converges faster. This is illustrated in Fig. 6 which shows the time required for satellites to reach the specified uncertainty threshold. The median is chosen as there are some outlier satellites with unfortunate orbits with respect to the sensor's search volume, and therefore their uncertainties increase more rapidly. Notably, for this simulation a linear penalty function is used with $k_p = 5$. Since this is a more conservative scheduling strategy compared to a quadratic function, we expect to not see a big difference between the priorities for a given method. However, since k_p is larger than the linear scaling used in Fig. 3 one expects this solution to be more aggressive in its tasking.

Fig. 5 shows how the OPA tasks the radar sensor for each method, for a representative run, but does not take into account the threshold. Since the penalty function is defined such that any uncertainty below the threshold is treated equal, one method can be better than another, but equal from the penalty functions perspective. Because of this, instead of showing the uncertainty over time, Fig. 7 shows the number of satellites below the threshold over time. This lets us compare the methods to each other, taking the threshold into account. Here it can be seen that method B lags behind both methods A and C since it cancels its tasks when the uncertainty is deemed sufficiently low. This affects how the uncertainty evolves, and it will therefore be more likely to increase above the threshold compared to if the measurement was used (as in method A).

These results support what was previously seen, that method C is more likely to ensure that lower-priority satellites also meet the uncertainty threshold, without sacrificing performance on higher-priority satellites. The reason not all satellite orbits are sufficiently determined after the weeklong simulation is in part due to some satellites not always being visible by the sensor due to the small search space. Additionally, the large number of satellites being planned simultaneously saturates the sensor which means that more time is required to measure all satellites.

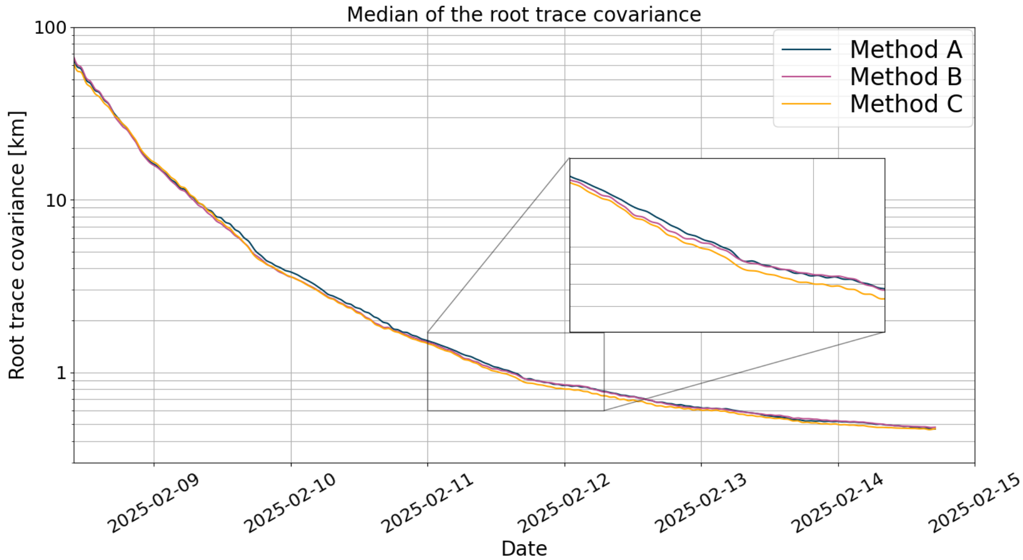

Fig. 5: Overview of the median root trace covariance of the total population of satellites, divided by case. The uncertainty threshold used is $T = 1$ km. The first two days are excluded as all methods are very similar on the resulting scale.

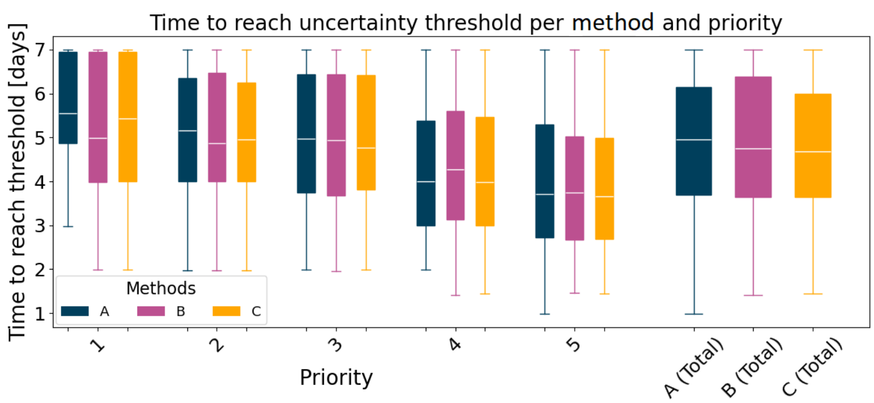

Fig. 6: Box and whiskers plots for the time required to reach the 1 km uncertainty threshold for the different cases and priorities.

Since methods B and C are simple alterations of method A, the differences between the three methods are not very large. Therefore the differences between them are due to the alterations, the ability to remove tasked observation as in method B (to reduce the sensor usage), or to reallocate them as in method C. With a purely greedy method as reference, one can see that making it more dynamic and allowing for reallocation improves the scheduling.

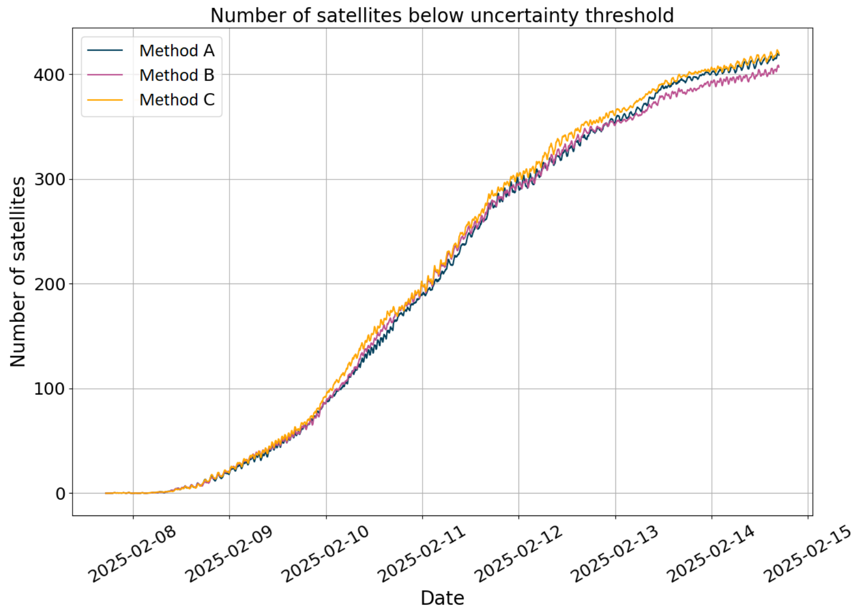

Fig. 7: Number of satellites with uncertainty below the uncertainty threshold, $T = 1$ km, for methods A, B, and C.

5. CONCLUSION

This study demonstrated the feasibility of applying a GA to dynamic sensor scheduling in space surveillance and tracking operations. A framework, which can be expanded upon to adapt it to other scheduling behaviours is presented. By simulating three different simple scheduling strategies, ranging from purely greedy to more opportunistic, we showed that all approaches seem to converge as expected provided sufficient time and measurement opportunities. Using a single radar and a population of 519 satellites, all methods had a high convergence rate within the simulated time interval, with the more opportunistic method C being marginally better in terms of both time to converge and total number of satellites below the uncertainty threshold. However, a comparison of more fundamentally different methods would likely have been more interesting, as it would have resulted in larger performance variations. Regardless, the methods were chosen just to be able to apply the GA framework and see if the results match expectations and illustrate its applicability.

Furthermore, the ability of the OPA to maintain the uncertainty below a defined threshold after convergence, and the ability to easily define a strategy that adds additional measurements when necessary, highlights its potential for use in situations where adaptability is essential, for instance when one includes the risk for measurements to unexpectedly become unavailable.

The incorporation of both a penalty function and a weighting function, which contribute to its flexibility, integrates well with the GA. We saw practical runtimes across various scenarios, which is useful for efficient resource allocation, particularly as satellite constellations grow and operational environments become increasingly dynamic. However, worth reiterating is that the simulations shown were based on Keplerian dynamics and a single sensor. Increasing the number of SSA sensors or the fidelity of the simulation may increase the execution times drastically. In these cases, it is important to carefully select penalty functions, as it requires the most amount of computation. The penalty functions also allow the operator to specify how the OPA should take the satellite weighting into account, which affects how conservative or opportunistic the solutions may be.

5.1 Future work

Three main areas for further development have been identified. Firstly, perform a more general analysis on how the penalty function interacts with uncertainty thresholds and measurement costs. Developing systematic tuning strategies could improve performance and reduce setup time. Also, investigating how the optimisation performs with more sensors may be interesting from a runtime perspective. Finally, future work could explore formulating the cost and penalty as separate objectives, using multi-objective genetic algorithms as well as exploring more practical strategies, for instance by taking into account larger satellite sets simultaneously when performing the optimisation, as described by (11).

6. ACKNOWLEDGEMENT

We would like to thank our colleagues: Ola Rasumusson, Christopher Blomqvist, Viktor Deleskog, Per Hägg, and Torbjörn Sundberg, at the Swedish Defence Research Agency for thoughtful and insightful discussions throughout the project.

7. FUNDING INFORMATION AND DATA AVAILABILITY

This work was funded by the Swedish Armed Forces' Research and Technology Development programme. For more information regarding the data used in this work, contact the authors.

REFERENCES

[1] M. Nylund et al. *Surveillance and Tracking of Space Objects*. (Swedish Title: *Inmätning av satelliter - En studie av radarsensortillämpning*). FOI-R–5270–SE, Swedish Defence Research Agency, 2022.
[2] R. Stottler and A. Li. *Automatic, intelligent commercial SSA sensor scheduling*. In *Advanced Maui Optical and Space Surveillance Technologies Conference*, Maui, Hawai'i, 2019.
[3] O. Bozorg-Haddad et al. *Meta-heuristic and Evolutionary Algorithms for Engineering Optimization*. Wiley, 2017.
[4] A. Ławrynowicz. *Genetic Algorithms for Solving Scheduling Problems in Manufacturing Systems*. *Foundations of Management*, 3(2):7–26, 2012.
[5] A. Globus et al. *Scheduling Earth Observing Satellites with Evolutionary Algorithms*. 2003.
[6] Z. Jiawei and X. Lining. *An improved genetic algorithm for the integrated satellite imaging and data transmission scheduling problem*. *Computers & Operations Research*, 139, 2022.
[7] R. Stottler. *Improved Space Surveillance Network (SSN) Scheduling using Artificial Intelligence Techniques*. In *Advanced Maui Optical and Space Surveillance Technologies Conference*, Maui, Hawai'i, 2015.
[8] N. K. Dhingra et al. *Machine Learning for Space Domain Awareness Sensor Scheduling*. In *Advanced Maui Optical and Space Surveillance Technologies Conference*, Maui, Hawai'i, 2024.
[9] J. A. Dahlke and R. A. Bettinger. *Enhanced Heuristic Algorithm for Optimal Cislunar Space Situational Awareness Architectures*. In *Advanced Maui Optical and Space Surveillance Technologies Conference*, Maui, Hawai'i, 2024.
[10] D. Moreno et al. *Coordinated planner algorithms*. In *International Conference on Space Situational Awareness*, Madrid, Spain, 2022.
[11] P. Hägg. Optimal sensor planning for ssa using system identification concepts. *The Journal of the Astronautical Sciences*, 70, 2023.
[12] A. O. Hero et al. *Foundations and Applications of Sensor Management*. Springer, 2008.
[13] Space-track.org. https://www.space-track.org/#catalog.
[14] R. J. Sullivan. *Radar Foundations for Imaging and Advanced Concepts*. The Institution of Engineering and Technology, 2004.
[15] P. J. Hannen. *Radar and Electronic Warfare Principles for the Non-Specialist, 4th edition*. SciTech Publishing, 2013.
[16] D. A. Vallado and W. D. McClain. *Fundamentals of Astrodynamics and Applications, 5th Ed.* Microcosm Press, 2022.
[17] W. H. Clohessy and R. S. Wiltshire. *Terminal Guidance System for Satellite Rendezvous*. *Journal of the Aerospace Sciences*, 1960.
[18] Management Council of the Consultative Committee for Space Data Systems. *Recommendation for Space Data System Standards: ORBIT DATA MESSAGES*. National Aeronautics and Space Administration, 2023.
[19] Orekit. https://www.orekit.org/. Accessed: 2025-05-05.