

# Optimal Sensor Planning for SSA using System Identification Concepts

Per Hägg

*FOI Swedish Defence Research Agency*

## ABSTRACT

The rapid growth of man-made objects in Earth's orbit has created a need for planning and prioritizing sensor measurements of these objects in order to keep track of their motion in orbit. In this paper, we adapt a concept from the field of system identification, namely application oriented input design, to the sensor-planning problem. Previous methods have predominantly focused on the problem of planning the measurements of different object such that they are as accurate as possible (in a broad sense) from a given set of possible observations. Here we view the problem differently; the objective is to find the least costly number of measurement such that the estimated orbital parameters are good enough for their intended application. To this end, we introduce the concept of an application set. The application set contains all orbital parameters that are considered of sufficient accuracy for the intended use. The objective of the optimization is hence not to find the most accurate set of parameters but find the cheapest combinations of observations such that the estimated parameters are guaranteed to be within the application set with a high probability.

We show how to formulate the planning problem as a convex optimization problem that can be solved efficiently with modern algorithms, even for large scale-problems. We then demonstrate the feasibility of the method in a simulation example. Finally, the paper discusses some interesting topics for future research.

## 1. INTRODUCTION

The number of space objects in Earth's orbit is growing rapidly. This growth drives a need for improved space situation awareness, both in terms of increased sensor capacity and in terms of a more efficient use of sensor resources. The orbits of space objects are predominantly determined using data from networks of radar and optical sensors, but other techniques such as laser and passive radio ranging are used as well. Intergovernmental organizations, governments, and commercial companies continuously expand their sensor networks to cope with the rapid growth. This provides a necessary increase in sensor capacity, but it also adds further complexity to prioritization and optimization of the sensors network for efficient use.

Planning, tasking and scheduling of sensors for space situation awareness has been studied extensively, see e.g. [1, 2, 3, 4]. The problem is however not unique for space situation awareness and has been studied in many related areas. A good overview on the subject can be found in [5] and the references therein. A common approach is to try to pick an assignment of sensors that minimizes some cost. The cost is usually a function of information gain, that is, the inverse of the uncertainty in the estimated orbital parameters and includes covariance measures as well as more complex information measures. The tasking is then performed by distributing all possible observations times among the objects of interest that gives the most accurate parameters, by the selected measure. In [6] the sensors are selected for observation, not to find the most accurate estimate but instead to form the parameter uncertainty matrix into a desired shape.

In this paper, we study the problem of optimal sensor planning, with the objective of finding an efficient method for the scheduling of space object measurements using a set of different sensors in a sensor network, in order to retain sufficient accuracy in the orbital parameters of the object. The network could consist of multiple sensors of different types, with different operational modes, and with different accuracy and performance. The main contribution of this paper is to apply the idea of application-oriented input design from the field of system identification to the sensor-planning problem. The field of system identification handles the problem of constructing a model of a dynamical system based on experimental data [7, 8, 9].

The concept of application-oriented input design for system identification was first introduced in [10]. The idea is to find the cheapest possible identification experiment such that the identified model satisfies the performance

specification given by the intended use of the model. The optimization is performed over the properties of the input signal such as the signal spectrum, power or length. To quantify the required model accuracy an application set is introduced. The application set constrains the accuracy needed of the identified parameters by the application in which the model is used. The objective is hence not to find the most accurate parameters, but parameters that are sufficiently accurate for their intended use.

In the sensor-planning problem, instead of designing the input signal, we select at which time a certain sensor should be used to make an observation. Each observation is associated with a cost describing the effort, for example time or money, to perform the measurement. The objective is thus to find the optimal combination of possible observations, with the smallest total cost, such that the estimated orbital parameters are accurate enough for their intended application. The approach we take in this paper hence differs from the more common approach of trying to find the most accurate parameters (by some measure) utilizing all possible observations.

The outline of the rest of the paper is as follows: a brief introduction to orbit determination and covariance analysis is given in Section 2; the sensor-planning problem is formulated in Section 3; Section 4 introduces the concept of an application set and shows how to formulate the sensor planning problem as a convex optimization problem; the feasibility of the proposed method is shown in simulations in Section 5: finally, conclusions and directions for future research are given in Section 6.

## 2. ORBIT DETERMINATION AND COVARIANCE ANALYSIS

In this section we give an introduction to the orbit determination and covariance analysis concepts required for the development of the planning algorithm. We will mainly follow the derivations and the notation from [11]. Vectors are denoted by bold face lower case letters (e.g.,  $\mathbf{x}$ ) and matrices by bold face upper case letters (e.g.,  $\mathbf{X}$ ).

Let

$$\mathbf{x}(t) = [\mathbf{r}(t) \quad \mathbf{v}(t) \quad \mathbf{p} \quad \mathbf{q}]^T,$$

denote a time-dependent parameter vector comprising of an object's position ( $\mathbf{r}$ ), velocity ( $\mathbf{v}$ ) as well as additional parameters related to the force and to the measurement model ( $\mathbf{p}$  and  $\mathbf{q}$ , respectively). The parameter vector may equally well be parameterized differently, for example by using the classical orbital elements instead of the position and velocity.

The position and velocity evolve in time as

$$\dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}).$$

The objective is to estimate the parameter vector  $\mathbf{x}(t)$  at a certain time  $t = t_e$  from  $N$  observations

$$z_i(t_i) = h_i(t_i, \mathbf{x}(t_e)) + e(t_i),$$

at times  $t_1, \dots, t_N$ . Note that the measurements could come from different sensors. For simplicity of notation, we here assume that the measurements  $z_i(t_i)$  are scalars. However, the case with multiple measurement from a single sensors can easily be handled by adding multiple scalar measurements per sensor. The function  $h_i$  describes how the measurement at time  $t_i$  depends on the state at the epoch  $t_e$  and  $e(t_i)$  are the measurement errors.

Here we model the measurement errors as zero-mean independent Gaussian random variables with expected value

$$E[e(t_i)] = 0,$$

and variance

$$E[e(t_i)e(t_j)] = \begin{cases} \sigma_i^2 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}.$$

An estimate of the vector  $\mathbf{x}(t_e)$  can now be found using least-squares by minimizing

$$J(\mathbf{x}(t_e)) = \sum_{i=1}^N \frac{(z(t_i) - h_i(t_i, \mathbf{x}(t_e)))^2}{\sigma_i^2}. \quad (1)$$

We denote the solution to the least squares problem as  $\hat{\mathbf{x}}(t_e)$ . Details on how to find the solution can be found in e.g. [11] or [12]. Under some assumptions on the measurements and the modelling of the dynamics, the estimate is Gaussian distributed with expected value

$$E[\hat{\mathbf{x}}(t_e)] = \mathbf{x}_0.$$

This requires for example that there are more measurements than unknowns to estimate and that the measurements are rich enough, i.e., all free parameters in the model are observable in the data and that there exists a representation of the parameter vector,  $\mathbf{x}_0$ , describing the true underlying dynamics. If these requirements are satisfied the estimate will hence, on average, be correct. Furthermore, it can be shown ([11]) that the (inverse) covariance matrix of the estimate is given by

$$\mathbf{P}^{-1} = \left( E \left[ (\hat{\mathbf{x}}(t_e) - \mathbf{x}_0) (\hat{\mathbf{x}}(t_e) - \mathbf{x}_0)^T \right] \right)^{-1} = \sum_{i=1}^N \underbrace{\frac{\mathbf{H}(t_i)^T \mathbf{H}(t_i)}{\sigma_i^2}}_{\mathbf{I}_f(i)} = \sum_{i=1}^N \mathbf{I}_f(i), \quad (2)$$

where

$$\mathbf{H}(t_i) = \left. \frac{\partial h_i(t_i, \mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}_0}. \quad (3)$$

Often there is some apriori information about the parameters, usually from a previous orbit determination. This information can be incorporated when solving the least squares problem [11]. If the apriori information and its uncertainty is  $\mathbf{x}_{apr}(t_e)$  and  $\mathbf{P}_{apr}$ , respectively, the covariance of the estimate can be written as

$$\mathbf{P}^{-1} = \mathbf{P}_{apr}^{-1} + \sum_{i=1}^N \mathbf{I}_f(i). \quad (4)$$

From the statistical properties of the estimated parameters, a confidence ellipsoid can be derived. With probability  $\alpha$  the estimated parameters  $\hat{\mathbf{x}}(t_e)$  lies within the confidence ellipsoid [13]

$$\mathcal{E}_{est} = \left\{ \mathbf{x} \mid (\mathbf{x} - \mathbf{x}_0)^T \mathbf{P}^{-1} (\mathbf{x} - \mathbf{x}_0) \leq \chi_{\alpha}^2(n) \right\},$$

where  $\chi_{\alpha}^2(n)$  is the  $\alpha$ -percentile of the  $\chi^2$ -distribution with  $n$  degrees of freedom. Here  $n$  is the dimension of the parameter vector  $\mathbf{x}$ .

### 3. PROBLEM FORMULATION

From equations (2) and (4) we see that the inverse of the covariance depends on the sum of the contributions from the individual measurements. Hence, by choosing whether to perform the measurement or not, the shape of the covariance matrix, or the confidence ellipsoid, can be altered. Note that since each contribution in the sum  $\mathbf{H}(t_i)^T \mathbf{H}(t_i)$  is positive (semi-) definite, every measurement will contribute and improve the estimate, that is, making the covariance smaller.

The idea in this paper is to only make the measurements necessary to achieve the accuracy required by the intended use of the estimated parameters. As (2) and (4) do not depend on the actual measurements, they can be calculated beforehand and used to plan which measurements to make.

First, all possible observations from the different sensors are determined by propagating the position of the object forward in time and checking when the object is observable. This step requires an initial estimate of the object's orbit of sufficient accuracy. For each possible measurement, we calculate its contribution to the covariance matrix using (3) and (4). Next, we introduce the selection variables

$$s_i \in \{0, 1\}, i = 1, \dots, N,$$

where  $N$  is the number of possible measurements. The selection variables determines if a measurement should be performed (1) or not (0). The inverse covariance matrix (4) can now be written as

$$\mathbf{P}^{-1} = \mathbf{P}_{apr}^{-1} + \sum_{i=1}^N s_i \mathbf{I}_f(i),$$

and the confidence ellipsoid as

$$\mathcal{E}_{est} = \left\{ \mathbf{x} \mid (\mathbf{x} - \mathbf{x}_0)^T \left( \mathbf{P}_{apr}^{-1} + \sum_{i=1}^N s_i \mathbf{I}_f(i) \right) (\mathbf{x} - \mathbf{x}_0) \leq \chi_{\alpha}^2(n) \right\}. \quad (5)$$

With each measurement we associate a cost. The cost describes the effort, in a broad sense, to perform the measurement. For example, certain sensors might be costlier to utilize both in time and money than others. Different observations using the same sensor may have different costs depending on when the measurement is made. The final selection of the measurement set will be based on this cost, where we will aim to minimize the total cost while still achieving the required accuracy.

The planning problem can thus be stated as: find the cheapest combination of measurement such that the estimated parameters, using the measurement data, satisfy the accuracy requirement for their intended use. This can be expressed as an optimization problem:

$$\begin{aligned} \min_{s_i} \quad & \text{total measurement cost} \\ \text{s.t.} \quad & \mathbf{P}^{-1} \text{ is accurate for the application} \\ & s_i \in \{0, 1\}. \end{aligned}$$

Note that the user is free to lump together multiple contributions to the inverse covariance matrix. For example, if multiple measurements are possible with one sensor during each passage, the user can choose to define a selection parameter either per measurement or per passage, or even split each passage into multiple parts, if desired.

In the next section we will show how to mathematically quantify when the estimated parameters are sufficiently accurate for the intended application, how to define the cost of a measurement, and how to formulate the planning problem as a convex optimization problem.

## 4. OPTIMIZATION PROBLEM

### 4.1 Application Cost and Application Set

To quantify if the estimated parameters are accurate enough for the intended application, we define an *application cost function* and an *application set*. The application function is a scalar function of the parameter vector  $\mathbf{x}$ , and is denoted  $V_{app}(\mathbf{x})$ . The function measures the degradation in performance when using the estimated parameters instead of the true parameters in the application. The application function should satisfy  $V_{app}(\mathbf{x}) \geq 0$  for all  $\mathbf{x}$ , and  $V_{app}(\mathbf{x}_0) = 0$ , i.e., the application cost is zero when the true underlying parameters are used.

The application set is then all the parameters that are accurate enough, i.e., all parameters that degrade the performance within an acceptable level. The set is defined as

$$\Theta_{app}(\gamma) = \left\{ \mathbf{x} \mid V_{app}(\mathbf{x}) \leq \frac{1}{\gamma} \right\},$$

where  $\gamma$  is a positive scalar accuracy parameter and  $1/\gamma$  is subsequently the maximum acceptable degradation. The objective is thus to perform an orbit determination such that the estimated orbital parameters,  $\hat{\mathbf{x}}$ , lies within the application set for a given accuracy  $\gamma$ , i.e.,  $\hat{\mathbf{x}} \in \Theta_{app}(\gamma)$ .

### 4.2 Ellipsoid Approximation of the Application Set

As discussed in Section 2, the estimated parameters are Gaussian distributed. Due to the infinite support of the distribution, it is not possible to guarantee that the parameters lie within the application set. Instead we require that the application requirement is satisfied with a certain probability  $\alpha$ , i.e.,

$$P(\hat{\mathbf{x}} \in \Theta_{app}) \geq \alpha. \quad (6)$$

The probability (6) is in general hard to evaluate and the constraint is non-convex in the decision variables  $s_i$ . To this end, we perform two approximations to convexify the constraint. The first approximation is to require the estimation ellipsoidal (5), with confidence level  $\alpha$ , to lie within the application set instead of the estimate  $\hat{\mathbf{x}}$  itself, that is,

$$\mathcal{E}_{est} \subseteq \Theta_{app}(\gamma). \quad (7)$$

The next step is to approximate the application set by an ellipsoid. The Taylor expansion of the application cost function at the true parameters  $\mathbf{x}_0$  is

$$V_{app}(x) \approx V_{app}(\mathbf{x}_0) + (\mathbf{x} - \mathbf{x}_0)^T V'_{app}(x_0) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^T V''_{app}(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0).$$

Since  $V_{app}(\mathbf{x}_0) = 0$  and  $V'_{app}(\mathbf{x}_0) = \mathbf{0}$  by definition, the application set can be approximated by

$$\Theta_{app}(\gamma) \approx \mathcal{E}_{app}(\gamma) = \left\{ \mathbf{x} \mid \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^T V''_{app}(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) \leq \frac{1}{\gamma} \right\}. \quad (8)$$

In some cases the Taylor ellipsoidal approximation may be an inaccurate representation of the application set, especially for low accuracy requirements. In this case a scenario approach can be used where a finite number of samples from within the application set is used to formulate the constraints, see [14]. Other ellipsoidal approximations are also possible, e.g., by using Khachiyan's algorithm [15].

With the ellipsoidal approximation, the set requirement (7) can be approximated as

$$\mathcal{E}_{est} \subseteq \mathcal{E}_{app}. \quad (9)$$

The problem of estimating the orbital parameters accurately enough for the intended application can hence be achieved by shaping the confidence ellipsoid, by including the appropriate observations, to fit within the application ellipsoid. Using the expression for the confidence ellipsoid (5), the constraint (9) can be written as [10]

$$\mathbf{P}^{-1} = \sum_{i=1}^N s_i \mathbf{I}_f(i) \succeq \chi_\alpha^2(n) \frac{\gamma V''_{app}(\mathbf{x}_0)}{2}, \quad (10)$$

where  $\mathbf{A} \succeq \mathbf{B}$  means that  $\mathbf{A} - \mathbf{B}$  is positive semi-definite. Equation (10) is a linear matrix inequality (LMI) and is a convex constraint on the decision variables  $s_i$  [16].

To illustrate the use of the application set we present two examples.

#### 4.2.1 Example 1

In the first example we want to guarantee that the estimated position of a satellite, at a certain epoch, is within 100 m of the true position of the satellite. The application cost function can hence be written as the squared distance from the estimated position to the true position

$$V_{app}(r) = \|\mathbf{r} - \mathbf{r}_0\|_2^2 = (r_x - r_{0,x})^2 + (r_y - r_{0,y})^2 + (r_z - r_{0,z})^2,$$

where  $\mathbf{r} = [r_x, r_y, r_z]^T$  is the position vector and  $\mathbf{r}_0 = [r_{0,x}, r_{0,y}, r_{0,z}]^T$  is the true position of the satellite. The application set is

$$\Theta_{app} = \{ \mathbf{r} \mid V_{app}(\mathbf{r}) \leq 100^2 \}.$$

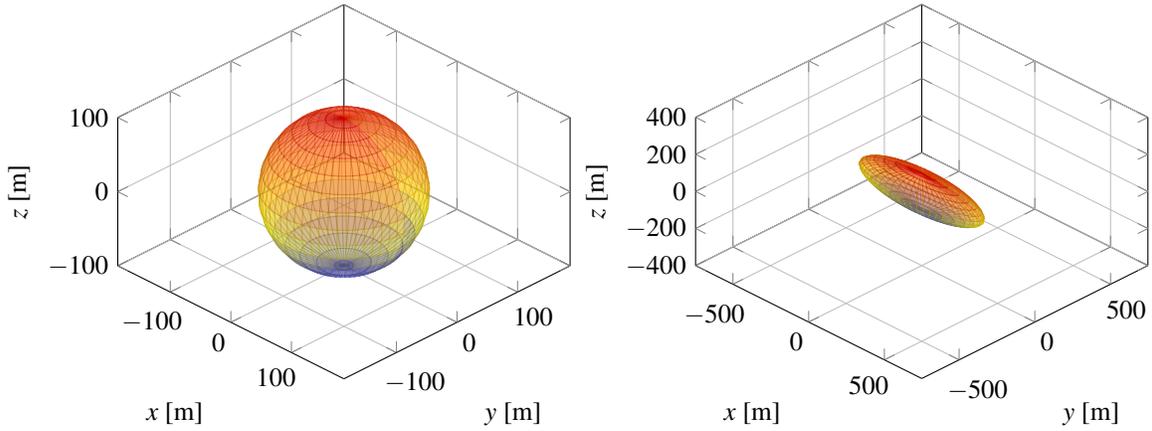
As the application set already represents an ellipsoid, the ellipsoidal approximation (8) is exact and can be written as

$$\mathcal{E}_{app} = \Theta_{app} = \left\{ \mathbf{r} \mid \frac{1}{2}(\mathbf{r} - \mathbf{r}_0)^T \underbrace{\begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}}_{V''_{app}} (\mathbf{r} - \mathbf{r}_0) \leq 100^2 \right\}.$$

The application ellipsoid is shown in Fig. 1a. Assume now that the error in the  $x$ -direction is less important and is only required to be known with an accuracy of 400 m. This could for example represent the along-track direction. The application set can in this case be written as

$$\Theta_{app} = \left\{ \mathbf{r} \mid \frac{(r_x - r_x^0)^2}{4^2} + (r_y - r_y^0)^2 + (r_z - r_z^0)^2 \leq 100^2 \right\}.$$

The ellipsoid with the relaxation along the  $x$ -axis is shown in Fig. 1b.



(a) Equal requirement on all axis

(b) Relaxed requirement along the x-axis

Fig. 1: Application ellipsoids for Example 1.

#### 4.2.2 Example 2

In the second example, we want to estimate the specific angular momentum of a satellite. The angular momentum is given by [12]

$$h(a, e) = \sqrt{\mu a (1 - e^2)},$$

where  $a$  is the semi-major axis,  $e$  the eccentricity and  $\mu$  is the gravitational constant. In this application we consider a satellite in a geostationary transfer orbit ( $350 \times 35785$  km). The objective is to estimate the semi-major axis and the eccentricity with such accuracy that the angular momentum is within 10% of the true angular momentum. An application cost function and the set reflecting this are selected as

$$\Theta_{app} = \left\{ a, e \mid (h(a, e) - h(a_0, e_0))^2 \leq (0.1h(a_0, e_0))^2 \right\},$$

where  $a_0$  and  $e_0$  are the true values of the semi-major axis and the eccentricity, respectively. The Taylor approximation of the application ellipse can, after some calculations, be written as

$$\varepsilon_{app} = \left\{ r \left| \frac{1}{2} \begin{bmatrix} a - a_0 & e - e_0 \end{bmatrix} \underbrace{\begin{bmatrix} \frac{\mu(1-e_0^2)}{2a} & -\mu e_0 \\ -\mu e_0 & \frac{\mu a_0 e_0^2}{1-e_0^2} \end{bmatrix}}_{V''_{app}(a_0, e_0)} \begin{bmatrix} a - a_0 \\ e - e_0 \end{bmatrix} \leq (0.1h(a_0, e_0))^2 \right. \right\}.$$

Fig. 2 shows the level curves for the accuracy of the specific angular momentum as a function of the deviation in semi-major axis and eccentricity from their true values. The area where the accuracy is better than the required 10% is shown as well as the approximated ellipse. The application ellipse approximates the application set reasonably well. The semi-major axis of the application ellipse is infinite. This is expected since for each  $a$  there exist an  $e$ , such that the accuracy requirement is satisfied, and vice versa.

#### 4.3 Cost Function

We want to select the cheapest combination of observations such that the estimated parameters satisfy the application requirements. First, we need to define the cost associated with each observation. The cost could be the monetary cost of performing the measurement but it can be used more generally to describe any effort associated with the observation, such as measurement time. It could also be used as a weighting to prioritize certain observations over others. Let  $c_i$  be the cost of performing observation  $i$ . The total cost of the included observations can then be written as

$$C = \sum_{i=1}^N s_i c_i. \quad (11)$$

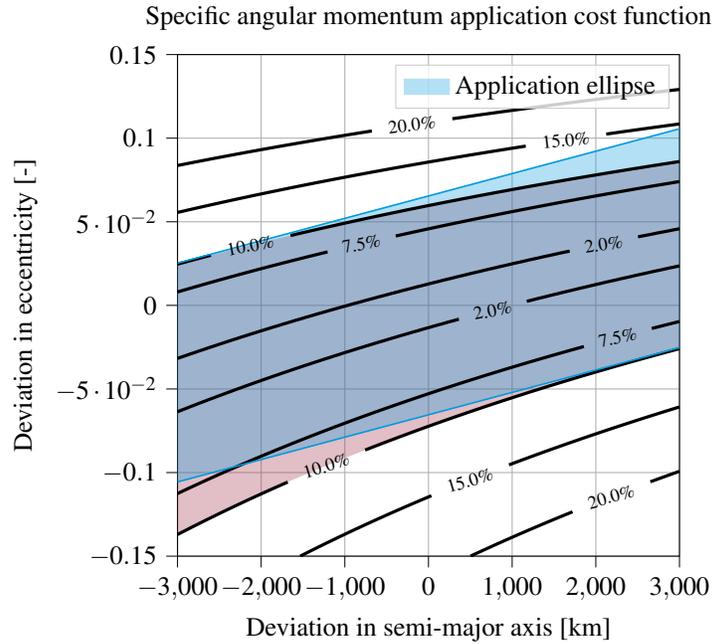


Fig. 2: Application ellipse for Example 2. The black lines represent level curves of the application cost function. The shaded red area is the true application set while the blue area is the ellipse approximation.

#### 4.4 Optimization Problem

We are now ready to formulate the complete optimization problem. Using (11) and (10) we get

$$\begin{aligned}
 \min_{s_i} \quad & \sum_{i=1}^N s_i c_i \\
 \text{s.t.} \quad & \sum_{i=1}^N s_i \mathbf{I}_f(i) \succeq \chi_\alpha^2(n) \frac{\gamma V_{app}''(\mathbf{x}_0)}{2} \\
 & s_i \in \{0, 1\}.
 \end{aligned} \tag{12}$$

Although the objective function and the first constraint of (12) are both convex, the constraint that the decision variables are either zero or one is not and makes the optimization problem non-convex. In fact it is a combinatorial problem in the variables  $s_i$  and the number of combinations to evaluate is in the order of  $2^N$ , where  $N$  is the number of possible observations. Hence, the number of combinations is growing fast with the number of possible observations and quickly becomes computational prohibitive. To make the problem tractable, we propose to relax the binary constraint  $s_i \in \{0, 1\}$  to  $0 \leq s_i \leq 1$ . The new optimization problem then becomes

$$\begin{aligned}
 \min_{s_i} \quad & \sum_{i=1}^N s_i c_i \\
 \text{s.t.} \quad & \sum_{i=1}^N s_i \mathbf{I}_f(i) \succeq \chi_\alpha^2(n) \frac{\gamma V_{app}''(\mathbf{x}_0)}{2} \\
 & 0 \leq s_i \leq 1,
 \end{aligned} \tag{13}$$

and is now convex in  $s_i$ . The optimization problem, with the LMI constraint is a semi-definite program (SDP) and can be solved efficiently, even for large problems, with modern optimization algorithms [16].

At first, the relaxation might seem unintuitive as the solution may involve parts of observations. However, with the positivity constraint on  $s_i$ , the cost function can be seen as the weighted  $\ell_1$ -norm of the decision variables, weighted with the observation costs. The  $\ell_1$  is a widely used heuristic to impose sparsity to the solution of an convex optimization

problem, i.e., that many of the elements of the solution is zero, see [17] and [16]. The solution to the optimization problem (13) will in many cases, as we will see in the simulation example in Section 5, be sparse and many of the elements will be zero. However, all of the decision variables are not zero or one, although usually most of them are. This is expected since there are only a discrete number of values of  $\gamma$  yielding solutions containing only zeros or one. How to handle this is still an open question. Possible solutions include thresholding the decision variables or solving the combinatorial optimization problem (12) but only for the elements that are not zero or one.

#### 4.5 Extension to Multiple Objects

To keep the notation simple, the derivation of the optimization problem in the previous sections was performed for a single object. However, the extension to finding the optimal observations times for multiple objects is straightforward. We here denote the new decision variable  $s_i^j$  where  $j = 1, \dots, N_{obj}$  and  $N_{obj}$  is the number of satellites or objects the observations should be planned for. As the number of possible observations could vary between the different objects, we now denote the number of possible times for observations for object  $j$  as  $N_j$ . The cost function (11) can now be written as

$$C = \sum_{j=1}^{N_{obj}} \sum_{i=1}^{N_j} s_i^j c_i^j.$$

The cost coefficients  $c_i^j$  now also depend on the specific object and could potentially be used to reflect different priorities among the objects.

The application cost can be different for the different objects and hence the constraint (10) is similarly extended to

$$\sum_{i=1}^{N_j} s_i^j \mathbf{I}_f(i, j) \succeq \chi_{\alpha}^2(n) \frac{\gamma \mathcal{V}_{app}''(j, \mathbf{x}_0)}{2}, \quad j = 1, \dots, N_{obj}.$$

The case when two or more observations are mutually exclusive, for example if two objects pass over a ground station at the same time and only one of them can be measured, can be handled by adding additional constraints to the optimization problem. If observation  $i_1$  of object  $j_1$  is mutually exclusive with observation  $i_2$  of object  $j_2$ , the constraint can be formulated as

$$s_{i_1}^{j_1} + s_{i_2}^{j_2} \leq 1.$$

If there are no overlapping, mutually exclusive, observations, the optimization problem can be decomposed into a number of smaller optimization problems, one for each object.

#### 4.6 The Unknown True Parameter Vector

The astute reader notes that the application set, and therefore the solution to the optimization problem, depends on the true orbital parameter vector  $\mathbf{x}_0$ . However, the objective is to select the observations to improve the knowledge of the parameters. If the true values are already known there is no need to do the observations in the first place. This contradiction is well known within the fields of system identification and experimental design but it is fundamentally true in any planning problem: without any initial knowledge, no planning is possible.

The solution to this problem is often to use an initial estimate or apriori information of the parameter vector in the observation planning instead of the true parameter vector [14]. The idea is that the shape of the application set is similar regardless of whether the initial estimate or the true parameter vector is used. In the orbit determination problems considered in this paper, the objective is to improve an estimate of the parameter vector from an earlier orbit determination. Hence, an initial estimate is often available.

Additionally, the planning can be updated iteratively. An initial estimate of the parameter vector is used to find an initial observation plan. As the measurements from the first actual observations are available, the data is used to estimate a new parameter vector. Future observations can then be re-planned by solving the optimization problem with the most recent estimate. This procedure can then be repeated when needed.

## 5. SIMULATION

The feasibility of the proposed observation-planning method is demonstrated in a simulation example. The example is idealised and simplified to readily illustrate the important concepts of the method.

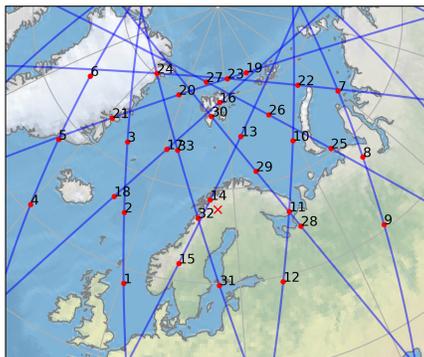
The objective is to plan sensor measurements to estimate the orbital parameters of a satellite to sufficient accuracy. Two fictitious radar sensors, one in Kiruna in northern Sweden and the other in French Guiana, are available. The two sensors only measure the range to the satellite. Their location and properties are shown in Table 1. The simulated satellite orbit is given in Table 2. The orbital parameters are given in an inertial frame, aligned with the Earth-centered, Earth-fixed (ECEF) frame at the start of the simulation.

Sensor	Location	Noise (at 600 km)	Cost ( $c_i$ )	Property	Value
Kiruna	67.860°N, 20.219°E	$\sigma = 25$ m	1.0	Altitude	600 km
French Guiana	5.191°N, 52.835°W	$\sigma = 30$ m	1.2	Eccentricity	0.0
				Inclination	98°
				Arg. of perigee	0°
				RAAN	25°
				Mean anomaly	-30°

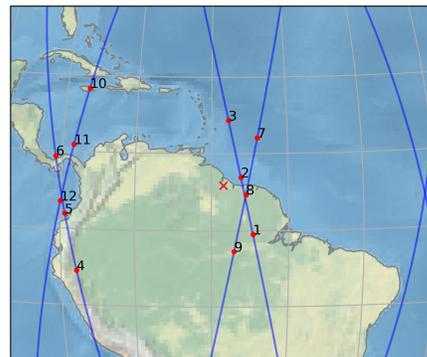
Table 1: Ground sensors and their properties used in the simulation example.

Table 2: Orbital parameters for the simulated satellite.

The objective is to estimate the relative state in a Hill-frame [11] of the satellite at 89900 seconds, or 15.5 orbits, after the start of the simulation. During the 15.5 orbits the satellite is visible from Kiruna at 12 passages and from French Guiana at 4 passages. For each passage, we assume that three measurement times are possible; the first at the highest elevation of the passage and the other two are two minutes before and after this moment. Hence, there are potentially 36 measurements from Kiruna and 12 from French Guiana to be used in the planning. The passages and the possible measurements are shown in Fig. 3, where the number indicates the temporal order of the measurements.



(a) Passages over Kiruna radar sensor



(b) Passages over French Guiana radar sensor

Fig. 3: Satellite passages (blue lines) over the two radar sensors (red crosses). The potential measurements are indicated by red dots and are enumerated by the corresponding number.

We use the Clohessy–Wiltshire equations [18] to model the dynamics relative to the ideal satellite. To be able to easily illustrate the ellipses we only estimate two parameters: the relative along-track and cross-track positions. The radial position and the velocity are assumed known. The application set is

$$\varepsilon_{app} = \Theta_{app} = \left\{ r_y, r_z \left| \frac{1}{2} \begin{bmatrix} r_y & r_z \end{bmatrix} \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}}_{V_{app}''} \begin{bmatrix} r_y \\ r_z \end{bmatrix} \leq \frac{1}{\gamma} \right\},$$

where  $r_y$  and  $r_z$  are the relative along-track and cross-track positions, respectively. The accuracy parameter  $\gamma$  will be

varied to show the effect on the solution to the optimization problem. The optimization problem (13) is solved using CVXOPT [19], a Python package for convex optimization. We want to satisfy the application constraint with  $\alpha = 95\%$  thus  $\chi_\alpha^2(n) = 5.99$ . The values of the selection parameters,  $s_i$ , are shown in Fig. 4 as  $\gamma$  is varied. We see that only a few selection parameters at a time increase from zero to one, as the accuracy requirement increases. Furthermore, once a measurement has been selected ( $s_i$  has a value of one) it will in general be used also for higher values of  $\gamma$ . This shows that the convex relaxation of  $s_i$  works as the resulting selection vector is sparse.

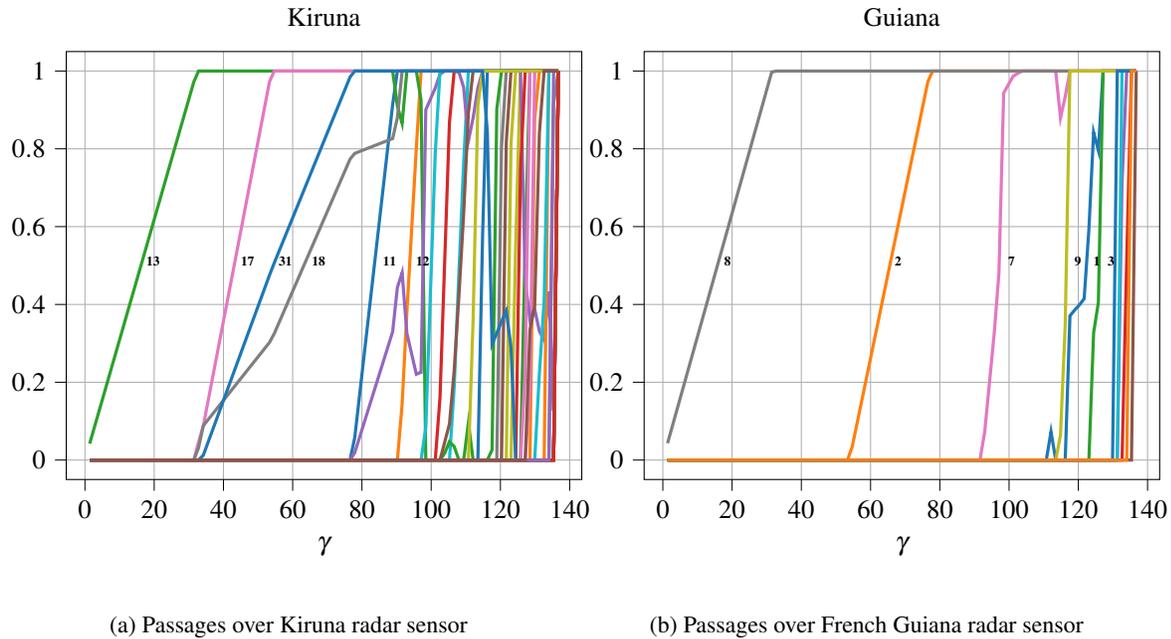


Fig. 4: The value of the selection parameters  $s_i$  as a function of the accuracy  $\gamma$ . The number shows the measurement order. Note that only the first few numbers are shown to aid readability.

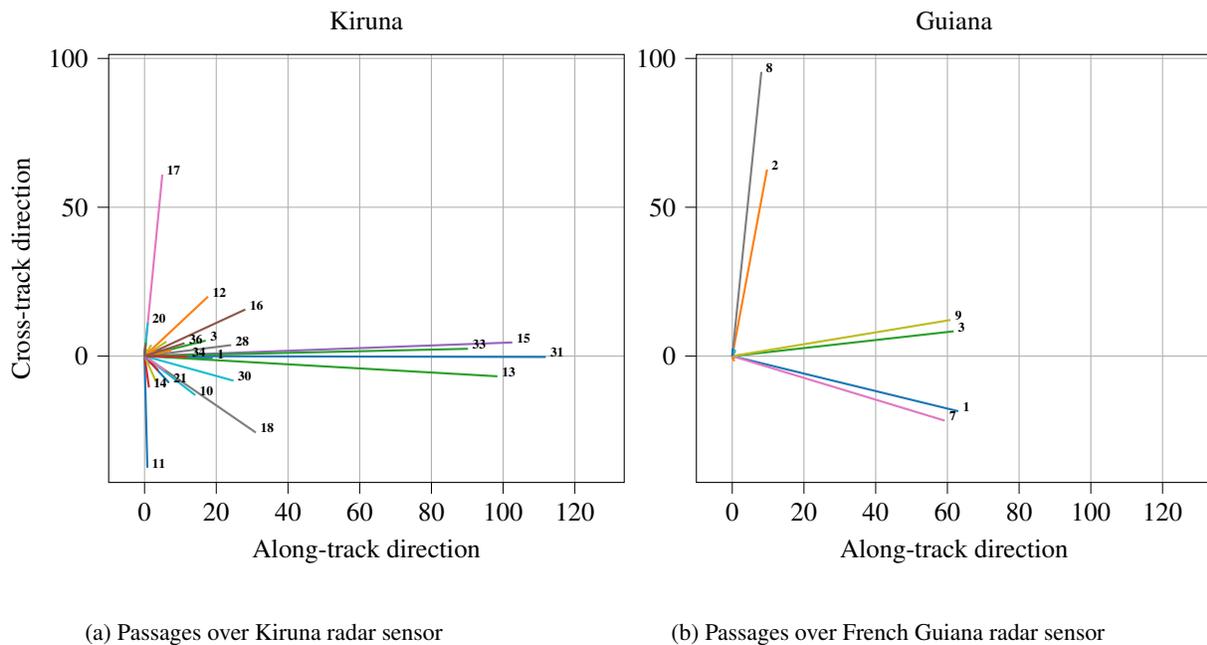
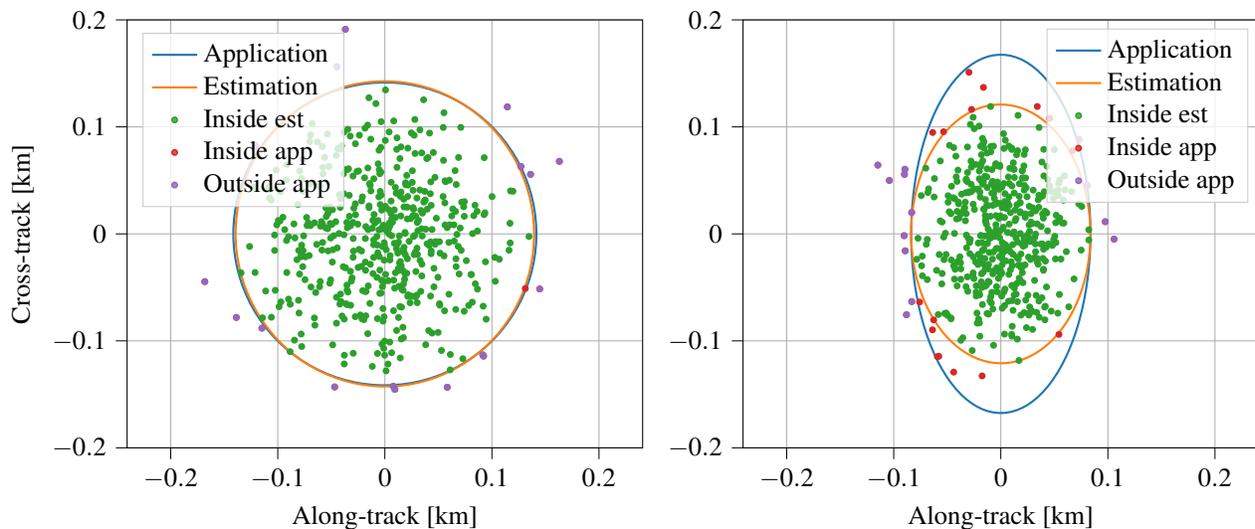


Fig. 5: Information content in the different direction from the different possible measurements.

To gain some insight into why certain measurements are prioritized over others, we show the information content of each possible measurement in Fig. 5. As the sensors only measures the range to the satellite, each measurement is one dimensional. This gives that the information matrix  $\mathbf{I}_f(i)$  in (2) has rank 1. The direction, and the amount of information, that each measurement contributes is calculated as the non-zero eigenvalue times the corresponding eigenvector of  $\mathbf{I}_f(i)$ . For example, measurement 31 from Kiruna mostly contributes information in the along-track direction. The amount of information in a measurement depends on a number of factors. First it depends on the geometry during the observation. In Fig 3 we see that due to the geometry of measurement 31 from Kiruna, it contains mostly information in the along-track direction. As the measurement noise standard deviation scales with the distance squared, the further away the satellite is from the sensor the less information the measurement contains. Lastly, the amount of information depends on how much information the observable parameters at the measurement time contain about the orbital parameters at the estimation epoch.

In this example, when the application set is an circle, see Fig 6a, measurement 13 from Kiruna and measurement 8 from French Guiana are the first to be used. Even though the measurements by themselves are not the ones with most information, they are almost perpendicular to each other and hence is the optimal combination for finding the lowest-cost ellipse within the set. This can be seen also for higher values of  $\gamma$  where measurement with perpendicular information content are selected pairwise. For high  $\gamma$  values all measurements are selected, even though they have a low information content in the directions of interest.



(a) Example with equal application requirements along the cross-track and along-track directions.

(b) Example with higher accuracy requirement in the along-track direction.

Fig. 6: Application and the optimized estimation ellipses. The dots show the estimated parameters from the 500 Monte Carlo simulations, inside the estimation ellipse, inside the application ellipse and outside the application ellipse.

Fig. 6a shows the application and estimation ellipses from the solution to the optimization problem for  $\gamma = 100$ . To verify the results we run 500 Monte Carlo simulations where actual measurement data is generated for the selected measurements. From the generated measurement data the orbital parameters are estimated by solving the non-linear least squares problem (1) using Powell's method [20]. The resulting 500 estimates of the parameters are shown in Fig. 6a. Out of the 500 estimates,  $483 \approx 97\%$  are inside the estimation ellipse and  $484 \approx 97\%$  inside the application ellipse, close to the desired 95%.

We now instead require higher accuracy in the along-track direction, i.e.,

$$V''_{app} = \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix}$$

The selection parameters, as a function of the accuracy  $\gamma$ , are shown in Fig. 7. Using Fig. 5, it can be clearly seen

that measurements with a large amount of information in the along-track direction have higher priority. The resulting application and estimation ellipses as well as the results from 500 Monte Carlo simulations are shown in Fig. 6b.

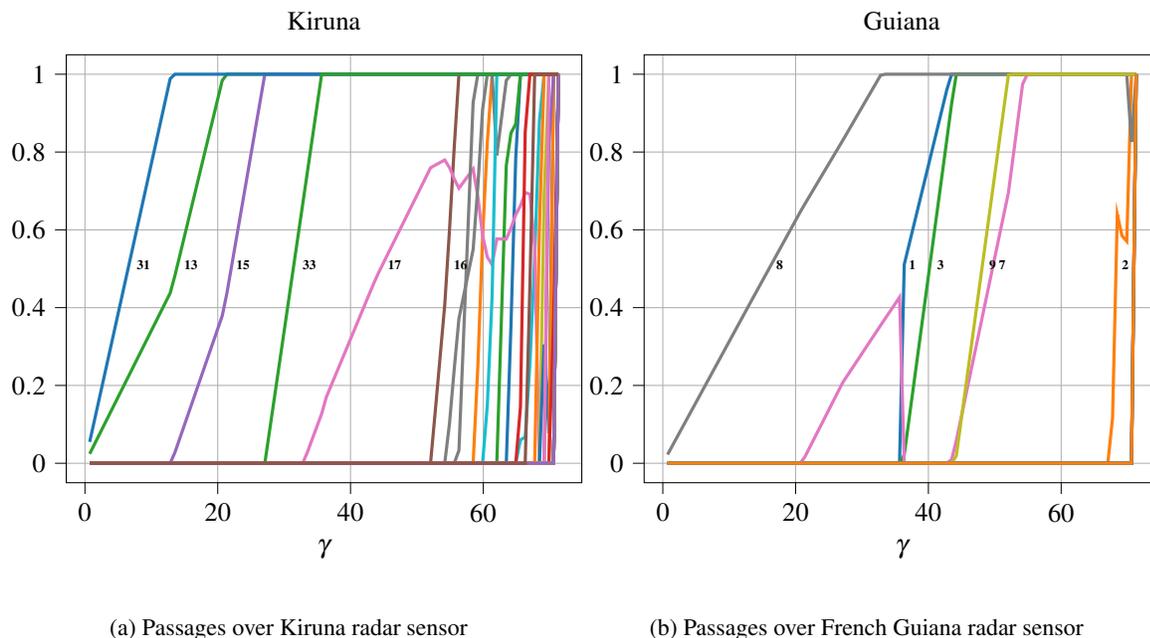


Fig. 7: The value of the selection parameters  $s_i$  as a function of the accuracy  $\gamma$ . The number shows the measurement order. Note that only the first few numbers are shown to aid readability.

## 6. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a method to plan sensor measurement based on ideas from the field of system identification and input design. Previous methods have mainly focused on the problem of planning the measurements of different object such that they are as accurate, in a wide sense, as possible, from a given set of possible observations. Here we view the problem differently; the objective is to find the least costly number of measurement such that the estimated orbital parameters are good enough for their intended application. We are hence not looking for the highest possible accuracy. To quantify if an estimated parameter vector is good enough we defined the application function and the application set. We then showed how to formulate the planning problem as a convex optimization problem, which can be solved efficiently. Finally, the feasibility of the method was demonstrated in an idealised simulation example.

As this is, to the best of the authors understanding, a novel approach to sensor planning much work remains to be done. Several questions remain to be answered before it can be applied in a real-life context:

- As discussed in Section 4.6, the optimal planning strategy is based on knowledge of the true parameter vector. One solution to this is to base the planning on some initial estimate of the parameters and update the planning iteratively as more and more information is available. Another interesting approach could be to make the planning robust with respect to the uncertainties in the initial estimate, see [21] for examples from input design. These remedies and the sensitivity of the method to uncertainty in the initial estimate requires further investigation.
- The application functions and application sets shown in this paper are somewhat idealised and used only to illustrate the method. More realistic and practically useful application functions and sets are needed based on real-world applications.
- The expression for the covariance matrix (2) is often too simple to represent the true uncertainty in the estimated parameters [5]. Extending the method to use more accurate expressions for the covariance and the contribution of each measurement, should be explored further.

- In this paper, we only applied the algorithm to relatively small problems with a limited number of objects and observation times. The performance of the method when applied to large-scale problems needs to be investigated.
- As shown in the simulation example, the decision variables are sometimes between 0 and 1. How to handle this is a topic for future research and the solution will probably depend on the application at hand.

## REFERENCES

- [1] T. de la Rosa, R. Fuentetaja, D. Borrajo, and C. L. López, “Sensor planning system for the space situational awareness (ssa) project,” in *2017 6th International Conference on Space Mission Challenges for Information Technology (SMC-IT)*, pp. 107–113, 2017.
- [2] A. Herz, E. Herz, K. Center, J. Neel, and M. Simpson, “Combined ssa sensor tasking for space-to-space and ground-to-space,” *Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS)*, 2017.
- [3] P. M. Siew, D. Jang, and R. Linares, “Sensor tasking for space situational awareness using deep reinforcement learning - AAS 21-741,” 08 2021.
- [4] R. S. Erwin, P. Albuquerque, S. K. Jayaweera, and I. Hussein, “Dynamic sensor tasking for space situational awareness,” in *Proceedings of the 2010 American Control Conference*, pp. 1153–1158, 2010.
- [5] A. B. Poore, J. M. Aristoff, J. T. Horwood, R. Armellin, W. T. Cerven, Y. Cheng, C. M. Cox, R. S. Erwin, J. H. Frisbee, M. D. Hejduk, B. A. Jones, P. Di Lizia, D. J. Scheeres, D. A. Vallado, and R. M. Weisman, “Covariance and uncertainty realism in space surveillance and tracking,” tech. rep., Numerica Corporation, 2016.
- [6] M. Kalandros and L. Pao, “Multisensor covariance control strategies for reducing bias effects in interacting target scenarios,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 1, pp. 153–173, 2005.
- [7] T. Söderström and P. Stoica, *System Identification*. Prentice Hall, 1989.
- [8] L. Ljung, *System Identification: Theory for the User*. Prentice Hall, second ed., 1999.
- [9] R. Pintelon and J. Schoukens, *System Identification: A Frequency Domain Approach*. Wiley-IEEE Press, second ed., 2012.
- [10] H. Hjalmarsson, “System identification of complex and structured systems,” *European Journal of Control*, vol. 15, no. 3, pp. 275–310, 2009.
- [11] O. Montenbruck and E. Gill, *Satellite Orbits - Models, Methods and Applications*. Springer Verlag, 2000.
- [12] D. Vallado, *Fundamentals of Astrodynamics and Applications*. Space Technology Library, second ed., 2001.
- [13] N. Draper and H. Smith, *Applied Regression Analysis*. John Wiley & Sons, 1966.
- [14] M. Annergren, *Application-Oriented Input Design and Optimization Methods Involving ADMM*. PhD thesis, KTH, Automatic Control, 2016. QC 20160602.
- [15] M. J. Todd and E. A. Yıldırım, “On Khachiyan’s algorithm for the computation of minimum-volume enclosing ellipsoids,” *Discrete Applied Mathematics*, vol. 155, no. 13, pp. 1731–1744, 2007.
- [16] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.
- [17] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.
- [18] W. H. Clohessy and R. S. Wiltshire, “Terminal guidance system for satellite rendezvous,” *Journal of the Aerospace Sciences*, vol. 27, no. 9, pp. 653–658, 1960.
- [19] M. Andersen, J. Dahl, and L. Vandenberghe, “CVXOPT: A python package for convex optimization, version 1.3.0..” Available at [cvxopt.org](http://cvxopt.org), 2022.
- [20] M. J. D. Powell, “An efficient method for finding the minimum of a function of several variables without calculating derivatives,” *The Computer Journal*, vol. 7, pp. 155–162, 01 1964.
- [21] C. R. Rojas, J.-C. Aguero, J. S. Welsh, G. C. Goodwin, and A. Feuer, “Robustness in experiment design,” *IEEE Transactions on Automatic Control*, vol. 57, no. 4, pp. 860–874, 2012.