TARGET SEARCH METHODS FOR SPACE SITUATIONAL AWARENESS

BY

MIHIR PATEL

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Aerospace Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2018

Urbana, Illinois

Advisor:

Assistant Professor Koki Ho

Abstract

This work studies methods to detect target in an orbit around the Earth using a space based sensor. Searching for a target among a large set of candidate orbits is a difficult and time consuming problem. Considering orbital dynamics, sensor uncertainties and the initial size of candidate location distribution, it is desirable to develop efficient search techniques. In this work, information-theoretic methods for searching a target in a large probability distribution using a space based sensor is considered. One intuitive approach is to steer the sensor towards regions of high probability density. Alternatively, information-theoretic methods steer the sensor based on metrics of the information gain in the posterior probability distribution. Through simulation, it is shown that information-theoretic search methods produce greater knowledge about probability distribution of the target's orbit. We also present methods to lower the computing expense imposed on the computer on-board a space based sensor. The issue is addressed using data clustering technique called K-means clustering. It is shown that errors resulting from searching the target after clustering is much lower compared to errors resulting from searching targets at the locations of higher probability.

Acknowledgments

I would like to thank my advisor Prof. Koki Ho for his continuous support and guidance during this work. Working with him was a pleasure and very rewarding. I would also like to thank Dr. Andrew Sinclair at Air Force Research Laboratory (AFRL). This work was initiated under his guidance during my summer internship at AFRL in summer of 2017. I would also like to appreciate the help I received along the way from Dr. Hoong Chieh Yeong and Ryne Beeson. Lastly, I thank the Jack Kent Cooke Foundation for the financial support.

Table of Contents

List of Abbreviations

X	=	set of candidate orbits
r	=	position vector
v	=	velocity vector
P	=	set of probabilities associated with candidate orbits
x^s	=	sensor orbit
x^t	=	target orbit
y_k	=	measurement type
θ	=	sensor half-cone angle
l	=	sensor's boresight
u	=	unit vector associated with sensor's boresight
P_{dp}	=	probability that sensor gets an accurate detection
P_{na}	=	probability that sensor gets an accurate no detection
P_{np}	=	probability that sensor gets a missed detection
P_{na}	=	probability that sensor gets a false alarm
H	=	entropy
$\bar{H}$	=	mean conditional values of entropy
D_{KL}	=	Kullback-Leibler divergence
r_{nom}	=	position vector of the nominal orbit
v_{nom}	=	velocity vector of the nominal orbit
FOV	=	field of view

Chapter 1

Introduction

Over the years, launching and deploying satellites in Earth's orbit has become easier and affordable. This has led to congestion in the space environment. The traffic in orbit is made up of operational and in-operational satellites as well as debris formed due to collision of satellites. Additionally, small Near Earth Objects (NEO) may also be present. Although, the United States Air Force (USAF) keeps a catalog of objects in Earth's orbit, they are not constantly monitored. Newly detected objects may have large uncertainties in their orbits. Therefore, tasking a follow-up sensor to search for the object can be a challenging problem.

In recent times, developments in the estimation, filtering and control theory have led to numerous studies for sensor management in Space Situational Awareness problems. These investigations have generally focused on the global SSA problem. That is, given a network of sensors and prior probability distributions for the entire population of space objects, which sensors should be tasked to observe which objects? Stochastic as well as information-theoretic approaches have been explored to address the problem [1, 2, 3, 4, 5, 6, 7, 8]. This work deals with a more local SSA problem. A single sensor is used to observe a single object. However, the uncertainty in the object's orbit is assumed to be large enough relative to the sensor's field of view, that tasking the sensor is nontrivial. The problem of target search in a large probability distribution has been studied in the past using probabilistic greedy algorithms [9, 10, 11]. However, information theoretic methods that have been proved in the networked sensor problem have not been explored.

In this work, we compare information-theoretic approaches with greedy probabilistic method to solve the target search problem. A widely used information-theoretic quantity that is considered is the Kullback-Leibler (KL) divergence. The Kullback-Leibler divergence is quantification of the difference between two probability distributions. In information-theoretic sensor tasking, the Kullback-Leibler divergence has been used to maximize the difference between the prior and posterior distribution. The entropy reduction from the prior to posterior distribution can be considered as the information gain due to an observation. The tasking method can be defined to minimize the expected posterior entropy. We compare the Kullback-Leibler divergence target search method to greedy probabilistic approach which we call the maximum probability approach. In subsequent sections, we introduce each of these methods in detail and then compare their

performance.

The second part of this work involves reducing the computation loads associated with the target search problem. In the above mentioned methods, the sensor is tasked to look at each candidate orbit and take a measurement. After taking the measurement, the onboard computer needs to compute the Kullback-Leibler divergence and choose the next pointing location. However, in the presence of large available pointing directions (i.e. a lot of candidate orbits), the algorithm quickly becomes computationally expensive. One way to optimize the computation of sensor pointing direction is by reducing the available number of candidates for the sensor to choose from. This can be done by clustering the the candidates according to sensor field of view (FOV) characteristics. This entails that the geometrical size of clusters is for the most part equal to sensor's FOV angle. By accomplishing this, one can just point the sensor at the cluster center, compute Kullback-Leibler divergence and then proceed to choose next optimal cluster center to point the sensor at. This method is compared to another computationally efficient target search method called the 'top x%' method. In this method, the available pointing locations for the sensor are restricted to the top x% of the total number of candidates. Here, by "top x%", we mean the first x% of candidates when they are ordered in descending order of their probabilities.

The remainder of this thesis is organized in the following order. In this chapter, a mathematical model of the target search problem is introduced. Chapter 2 introduces the basics of Information theory. These concepts are then described in the target search frame work. An example simulation comparing the information-theoretic target search and maximum probability target search methods is also given in this chapter. Chapter 3 introduces K-means clustering and 'top x%' methods. Again, an explanation of using them in the target search problem is given. Following this, an example simulation is then shown to validate the effectiveness of K-means clustering method. Chapter 4 wraps up the thesis with conclusions of this work and possible future directions.

1.1 Problem statement

Consider that the probability distribution of a target's orbit is approximated with a particle representation where each particle is considered a candidate orbit. Let a set of these particles to be denoted as $X(t) = \{\boldsymbol{x}_1(t), \boldsymbol{x}_2(t), \ldots, \boldsymbol{x}_n(t)\}$. Each particle $\boldsymbol{x}_j(t)$ defines a position, $\boldsymbol{r}_j(t)$ and velocity, $\boldsymbol{v}_j(t)$ of the candidate orbit:

$$\boldsymbol{x}_j(t) = \begin{bmatrix} \boldsymbol{r}_j(t) \\ \boldsymbol{v}_j(t) \end{bmatrix} \text{ for } j = 1, ..., n \tag{1.1}$$

The candidates each have an associated probability, and the set of probabilities is denoted as $P(t) = \{p_1(t), p_2(t), \ldots, p_n(t)\}$, such that $\sum_P p_j = 1$. As one possibility, the candidates in X could result from a Monte Carlo sampling of the target orbit's probability density function, and the probabilities would be initialized with equal probability, $p_j = 1/n$. However, other possibilities could also be admitted.

Similarly, consider an orbiting sensor and let its orbit be denoted by $\boldsymbol{x}^s(t)$. Here, $\boldsymbol{x}^s(t)$ defines the position of sensor, $\boldsymbol{r}^s(t)$ and the velocity of the sensor $\boldsymbol{v}^s(t)$. The sensor is assumed to have a canonical field of view with a fixed half-cone angle, θ, and the pointing of the sensor's boresight $\boldsymbol{l}(t_k)$ can be arbitrarily chosen.

Furthermore, the sensor is assumed to provide a binary detection measurement, i.e. either a detection or no detection measurement related to the presence of target in the field of view:

$$y_k = \begin{cases} 1, & \text{if detection at } t_k \\ 0, & \text{if no detection at } t_k \end{cases} \tag{1.2}$$

where, y_k is the measurement taken by the sensor at time t_k. Let these discrete set of measurements be stored in a vector $z_k = y_1, ..., y_k$. Based on the sensor tasking choice l_k and measurement y_k, the probabilities of the candidate orbits are updated using Bayes theorem.

$$p_j(t_k) = \frac{p(y_k|\boldsymbol{x}_j)p_j(t_{k-1})}{p(y_k)} \tag{1.3}$$

To compute probabilities $p(y_k|\boldsymbol{x}_j)$ and $p(y_k)$, define a set N_k as the candidates that are within the sensor's field of view, when it is pointed at some chosen boresight l_k:

$$\boldsymbol{u}_j(t) = \frac{\boldsymbol{r}_j(t) - \boldsymbol{r}^s(t)}{||\boldsymbol{r}_j(t) - \boldsymbol{r}^s(t)||} \tag{1.4}$$

$$N_k(\boldsymbol{u}_j) = \{\boldsymbol{x}_j(t_k) \mid \cos^{-1}(\boldsymbol{u}_j(t_k) \cdot \boldsymbol{l}_k) < \theta\} \subset X \tag{1.5}$$

Note that u_j is the unit vector pointing in the direction of the sensor's boresight $l(t_k)$. Additionally, the probability $p(y_k|\boldsymbol{x}_j)$ takes a value P_{dp}, P_{na}, P_{np}, or P_{da} i.e. the probability that the sensor takes an accurate detection, an accurate no detection, a missed detection or a false alarm. These values are obtained depending on the value of y_k and whether the candidate is in the sensor's field of view, i.e. whether $x_j \in N_k$. The

probability $p(y_k)$ is computed as follows :

$$p(y_k) = \begin{cases} \sum_{j \in N} P_{dp}\ p_j + \sum_{j \in X \setminus N} P_{da}\ p_j, & \text{if } y_k = 1 \\ \sum_{j \in N} P_{na}\ p_j + \sum_{j \in X \setminus N} P_{np}\ p_j, & \text{if } y_k = 0 \end{cases} \tag{1.6}$$

Thus, the target search problem is defined as the selection of pointing directions for the sensor's boresight at each measurement time. Here, several performance indices are considered to define optimal pointing directions. To reduce the optimization space to finite dimension, the possible pointing directions are restricted to the set of unit vectors parallel to the relative position vectors from the sensor to each candidate, i.e. $l(t_k) \in \{u_1(t_k), ..., u_n(t_k)\}$.

Chapter 2

Information-theoretic quantities in the target search problem

In this chapter, information-theoretic concepts of Shannon Entropy and Kullback-Leibler divergence are introduced. Entropy quantifies the amount of information present in a given probability distribution. On the other hand, the Kullback-Leibler divergence measures the information gained from a new probability distribution resulting from an observation made on the original probability distribution.

2.1 Information-theoretic quantities

One of the most important quantities in Information-theory is Entropy. It is defined as the average amount of information contained in a probability distribution. Consider a discrete probability distribution $P(t_k) = \{p_1, p_2 ... p_n\}$, the average amount of information, or entropy of $P(t_k)$ is defined as :

$$H(P(t_k)) = \sum_{j=1}^{n} p_j(t_k) \log \frac{1}{p_j(t_k)} \tag{2.1}$$

Entropy of a probability distribution takes the value between 0 and $\log(n)$. It takes the value of 0 when exactly one p_j is 1 and all other values in the probability distribution take the value, 0. On the other hand, the value of $H(P(t_k)) = \log(n)$ entails all events in the probability distribution being equally likely.

Entropy can also be used to quantify difference between any two probability distributions. Often known as information divergence, it used extensively in estimation theory. Consider an apriopri probability distribution $P(t_{k-1})$ and consider another probability distribution $P(t_k)$ that is obtained after an observation. The gain in information due to the measurement is quantified by an information theoretic measure known as the Kullback-Leibler divergence. Kullback-Leibler divergence from $P(t_{k-1})$ to $P(t_k)$ is computed as:

$$D_{KL}(P(t_k)|P(t_{k-1})) = -\sum_{j=1}^{n} p_j(t_k) \log \frac{p_j(t_{k-1})}{p_j(t_k)} \tag{2.2}$$

Note that the Kullback-Leibler divergence is a non-symmetric quantity. That is $D_{KL}(P(t_k)|P(t_{k-1})) \neq D_{KL}(P(t_{k-1})|P(t_k))$.

2.2 Target search method formulation

Information-theoretic concepts are well-studied and applied to many engineering applications such as signal processing and communication. The concepts are widely applied in estimation theory to enhance the decision making process for future control inputs. The main contribution of information theory is to quantify information gained by taking or not taking a measurement at certain location. In our problem, information theoretic concepts can be used to decrease the size of a probability distribution by taking a measurement at an appropriate location. One important quantity in information theory is entropy. It is defined as the average amount of information contained in a probability distribution. Consider a discrete probability distribution $P(t_k) = \{p_1, p_2...p_n\}$, the average amount of information, or entropy of $P(t_k)$ is defined as :

$$H(P(t_k)) = \sum_{j=1}^{n} p_j(t_k) \log \frac{1}{p_j(t_k)} \tag{2.3}$$

Entropy of a probability distribution takes the value between 0 and $\log(n)$. It takes the value of 0 when exactly one p_j is 1 and all other values in the probability distribution take the value, 0. On the other hand, the value of $H(P(t_k)) = \log(n)$ entails all events in the probability distribution being equally likely.

Entropy can also be used to quantify difference between any two probability distributions. Often known as information divergence, it used extensively in estimation theory. Consider an apriopri probability distribution $P(t_{k-1})$ and consider another probability distribution $P(t_k)$ that is obtained after an observation. The gain in information due to the measurement is quantified by an information theoretic measure known as the Kullback-Leibler divergence. Kullback-Leibler divergence from $P(t_{k-1})$ to $P(t_k)$ is computed as:

$$D_{KL}(P(t_k)|P(t_{k-1})) = -\sum_{j=1}^{n} p_j(t_k) \log \frac{p_j(t_{k-1})}{p_j(t_k)} \tag{2.4}$$

Note that the Kullback-Leibler divergence is a non-symmetric quantity. That is $D_{KL}(P(t_k)|P(t_{k-1})) \neq D_{KL}(P(t_{k-1})|P(t_k))$.

For the target search problem, we can use information-theoretic objective to find optimal tasking of the sensor. However, in order to use it, we must know the probabilities at the current measurement, $P(t_k|l_k, y_k)$. It can also be noted that l_k and y_k also remain unknown at the current time step. Nevertheless, Kullback-Leibler divergence can be calculated for any choice of l_k. The conditional probabilities $P(t_k|l_k, y_k)$, can be calculated for each choice of l_k and the two possible values of y_k using the Bayes rule. For each value of $P(t_k|l_k, y_k)$, the entropy $H(P(t_k|l_k, y_k))$ and the Kullback-Leibler divergence $D_{KL}P(t_{k-1})|P(t_k|l_k, y_k)$ can also be computed. Finally, the mean conditional values Kullback-Leibler divergence are computed by taking

expected values over the measurement outcomes.

$$\begin{aligned}\bar{D}_{KL}(t_k, l_k) =& p(y_k{=}0) D_{KL}(P(t_{k-1})|P(t_k|l_k, y_k{=}0)) \\ &+ p(y_k{=}1) D_{KL}(P(t_{k-1})|P(t_k|l_k, y_k{=}1))\end{aligned} \tag{2.5}$$

2.2.1 Target search methods

In this section, three different target search methods are described. For the two information theoretic approaches, the quantities from the previous section are used as objective functions. However first, consider a greedy probabilistic approach. As mentioned earlier, we call it the maximum probability approach. In this approach, given a prior probability distribution $P(t_{k-1})$ of a candidate set X, the sensor points its boresight, l_k at the candidate orbit that gives the maximum total probability of the candidates in the sensor's field of view.

$$\begin{aligned} l_k &= u_{j^*} \\ \text{where } j^* &= \arg\max_j \sum_{i \in N_k(u_j)} p_i(t_{k-1}) \end{aligned} \tag{2.6}$$

According to the measurement outcome, probabilities of the candidate orbits are updated using Bayes rule. This process is continued for the duration of the observation campaign. A pseudo-code of this approach is given in Algorithm 1. It takes initial position and velocities of candidates, target and sensor orbit, $x^t(t_0), x^s(t_0), X(t_0)$ and the probabilistic model of the sensor dictated by the previously mentioned quantities $P_{dp}, P_{na}, P_{np}, P_{da}$. The output of the algorithm is the posterior discrete probability distribution following each measurement.

Algorithm 1 Maximum Probability Target Search (MPTS)

```
procedure MPTS(x^t(t_0),x^s(t_0),X(t_0),P(t_0),P_dp,P_na,P_np,P_da)
    for k = 1:Number of observations do
        Propagate x^t,x^s,X
        Compute field of views for line of sight of the sensor pointed at each x ∈ X
        For each line of sight sum total probability in field of view
        Take Measurement by pointing sensor to maximize probability in field of view
        Update candidate probabilities based on measurement: P(t_k)
    end for
end procedure
```

The Kullback-Leibler target search approach uses the Kullback-Leibler divergence as an objective function to plan measurements. Given an initial probability distribution $P(t_{k-1})$ of a candidate set X, the sensor points its boresight l_k at each orbit, temporarily updates the distribution to P based on all possible sensor

measurements, and then computes the mean Kullback-Leibler divergence according to Equation (2.5). The actual measurement is then taken at the location where the Kullback-Leibler divergence is the maximum:

$$\begin{aligned} l_k &= u_{j^*} \\ \text{where } j^* &= \arg\max_j \bar{D}_{KL}(t_k, u_j) \end{aligned} \tag{2.7}$$

Maximizing the Kullback Leigbler divergence shrinks the probability distribution following every measurement. A pseudocode of this target search approach is given in Algorithm 2. The inputs to this algorithm is same as inputs of Algorithm 1 and so is the output.

Algorithm 2 Kullback-Leibler divergence target search (KLDTS)

procedure KLDTS($x^t(t_0)$,$x^s(t_0)$,$X(t_0)$,$P(t_0)$,P_{dp},P_{na},P_{np},P_{da})
  for k = 1:Number of observations **do**
    Propogate x^t,x^s,X
    Compute Field of Views for each $x \in X$
    ▷ Check which candidates are in view when LOS is pointed at each $x \in X$
    Compute KL divergence
    ▷ **For each pointing location**
    ▷ Compute Probability of getting a detection P_d and nodetection P_n based on candidate orbit probabilities
    ▷ Update candidate probabilities for detect and nodetect scenarios
    ▷ Compute mean KL divergence : $D_{KL}(t_k, l_k)$
    ▷ **Select the candidate with highest KL divergence: max($D_{KL}(t_k, l_k)$)**
    Take Measurement
    Update candidate probabilities based on measurement: $P(t_{k+1})$
  end for
end procedure

The third sensor tasking method that is of interest in this work is called the minimum mean conditional entropy target search approach. In order to compute mean conditional entropy, prior probabilities are hypothetically updated based on a temporary measurement. Using these hypothetical updated probabilities, conditional entropies are computed at each location. The actual measurement is then taken at the location of minimum mean conditional entropy:

$$\begin{aligned} l_k &= u_{j^*} \\ \text{where } j^* &= \arg\min_j \bar{H}(t_k, u_j) \end{aligned} \tag{2.8}$$

The minimum mean conditional entropy approach differs from Kullback-Leibler approach from the fact that it plans future measurements just on the basis of posterior probabilities. On the other hand, Kullback-Leibler divergence approach employs both prior and posterior probabilities. A pseudocode of this target search approach is given in Algorithm 3. Again, this algorithm accepts the same inputs as Algorithm 1 and

Algorithm 2 and generates the same output.

Algorithm 3 Minimum Mean Conditional Entropy Target search (MMCE)

1: **procedure** MMCE($x^t(t_0)$,$x^s(t_0)$,$X(t_0)$,$P(t_0)$,P_{dp},P_{na},P_{np},P_{da})
2: **for** k = 1:Number of observations **do**
3: **Propogate** x^t,x^s,X
4: **Compute Field of Views for each** $x \in X$
▷ Check which candidates are in view when LOS is pointed at each $x \in X$
5: **Compute Mean Conditional Entropies**
▷ **For each pointing location**
▷ Compute Probability of getting a detection P_d and nodetection P_n based on candidate orbit probabilities
▷ Update candidate probabilities for detect and nodetect scenarios
▷ Compute Mean Conditional Entropy : $\bar{H}(t_k, l_k)$
▷ **Select the candidate with lowest Mean Conditional Entropy: min($\bar{H}(t_k, l_k)$)**
6: **Take Measurement**
7: **Update candidate probabilities based on measurement:** $P(t_{k+1})$
8: **end for**
9: **end procedure**

2.3 Simulation results

In this section, the introduced methods are simulated and compared. First, consider a case in which the target orbit is exactly one of the candidate orbits. Whereas this case is unrealistic, if the candidates result from Monte Carlo sampling of the target orbit's probability density function, it provides insight into the behavior of the search methods. In this case, it is expected that over the sequence of measurements, the sensor will be able to exactly resolve the target's orbit. In the second, a more realistic case, the target's orbit is distinct from all of the candidate orbits.

2.3.1 Target in candidate set

In this case, it is assumed that $x^t \in X$. A candidate set of orbits is created by normally distributing them around a nominal position and velocity, $r_{\text{nom}}(t_0)$ and $v_{\text{nom}}(t_0)$ respectively. The probabilities of these candidate orbits are initialized at the value of $1/n$ where n is the number of candidate orbits. The sensor and the target orbits, $x^s(t_0)$ and $x^t(t_0)$, are also initialized at some initial position and velocity. All these orbits are propagated in the Earth's inverse gravity field. The orbiting sensor is tasked to take measurements at each time step of the orbit propagation based on each of the previously mentioned target search methods. A set of 1000 Monte Carlo trials were simulated. In each trial, new candidate set was generated, and the target was chosen as the tenth candidate. Along with those orbits, measurement errors were randomized in each Monte Carlo trial. Expected entropy and field of view plots were generated across the 1000 samples to see

the average behavior of the target search methods. The 'target in candidate set' case is described visually in Figure 2.1. It is expected that each time step, the target search schemes will try to eliminate candidates and ultimately converge to a candidate that is infact the target. The parameters chosen for simulation are presented in Table 2.1.

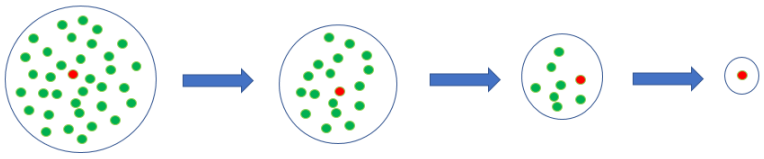


Figure 2.1: Target in candidate sequential evaluation

Table 2.1: Target in candidate set: simulation parameters

Parameter	Values
Nominal Initial Position	$[5715.1, 3325, 0]$ km
Nominal Initial Velocity	$[-3.4352, 5.95, 3.9666]$ km/s
Candidate orbit set position variance (in each coordinate)	50 km
Candidate orbit set velocity variance (in each coordiante)	0.0001 km/s
Number of candidate orbits , n	200
Target orbit number in candidate set	10
Sensor Initial Position, x_s^i	$[6062.2, 3500, 0]$ km
Sensor Initial Velocity, v_s^i	$[-3.2675, 5.6595, 3.7730]$ km/s
Number of measurements	20
$P_{dp} = P_{na}$	0.90
$P_{np} = P_{da}$	0.10
Number of Monte Carlo trials	1000
Time between measurements (sec)	100
Sensor cone half angle (field of view)	5°

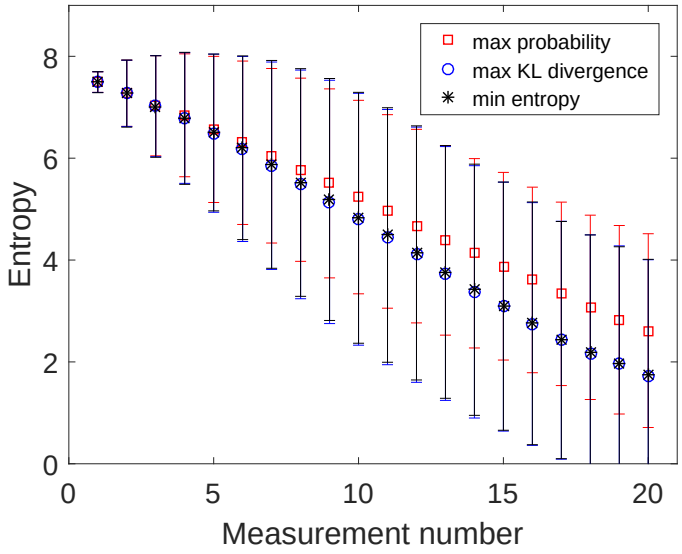


Figure 2.2: Entropy values of each target search method

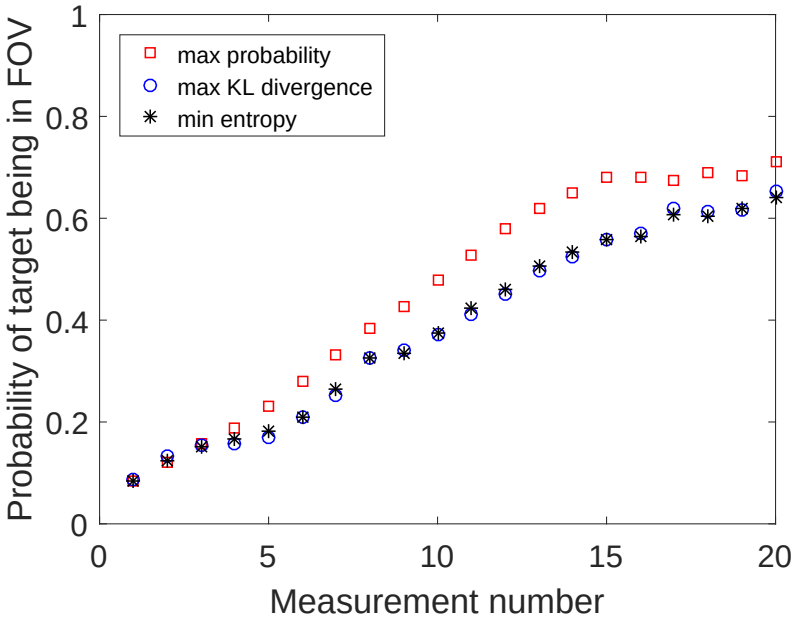
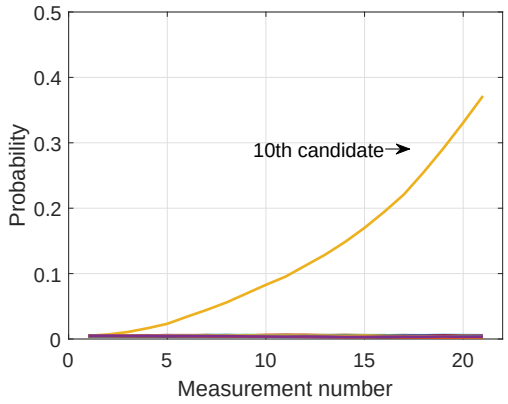


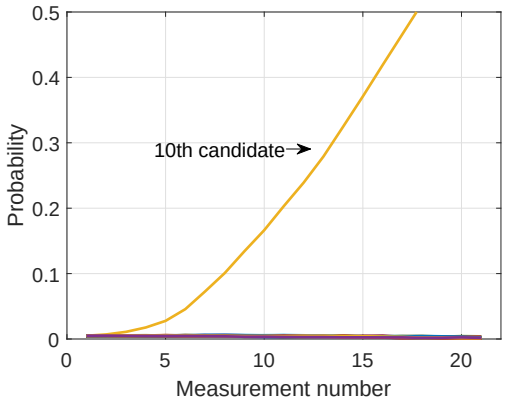
Figure 2.3: Probability of target being in the sensor's field of view

In Figure 2.2, through the use of markers, average values of entropy across 1000 Monte Carlo samples. Also plotted using the vertical errorbars is the variability in entropies from the mean entropy values across the Monte Carlo samples. The lengths of the errorbars in either direction of the average values are equal to one standard deviation of the data set from the average value. Note that entropy is a positive quantity, however in Figure 2.2 , from measurement 18 and onwards the lower bound of error bars extend beyond 0. This does not imply that in some Monte Carlo samples entropies were negative. The error bars are shown to emphasize the variability of the entropy values across the Monte Carlo trials. It can be seen that the amount of information gain is higher in the two information-theoretic target search methods. As expected, with the availability of more information.

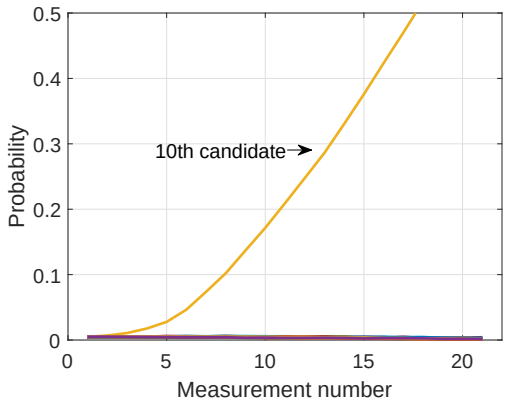
In Figure 2.3, it can be seen that over the course of an observation campaign, the maximum probability target search method gets the target in its field of view more often than information-theoretic target search methods. This suggests that to increase the information gain, information-theoretic search methods occasionally look away from the region of highest probability. It may not be desirable to keep looking at the candidates with high probability at all times because using the given sensor model, this does not help distinguish between these candidates.



(a) Maximum probability target search



(b) Maximum KL divergence target search



(c) Minimum entropy target search

Figure 2.4: Probability values of each target search method

Each subfigure in Figure 2.4 shows the updates of the probability distribution after every measurement. Over the course of 20 observations, it can be seen that the two information-theoretic methods converge to the target orbit with more certainty than the maximum probability target search method.

2.3.2 Target not in candidate set

In this more realistic case, consider the scenario when $x^t \notin X$. Again, a candidate set of orbits is created by normally distributing its members around a nominal position and velocity, $r_{\text{nom}}(t_0)$ and $v_{\text{nom}}(t_0)$ respectively. The probabilities of candidate orbits are initialized at the value of $1/n$. The sensor and the target orbits, $x^s(t_0)$ and $x^t(t_0)$, are also initialized at some initial position and velocity. All these orbits are propagated in the Earth's inverse gravity field. The orbiting sensor is tasked to take measurements at each time step of the orbit propagation based on each of the previously mentioned target search methods. The simulation parameters are the same as given in Table 2.1, however, the target orbit is initialized independently of

the candidate set. The initial position and velocity of the target orbit is normally distributed around the nominal orbit with a standard deviation of 50 km and 0.0001 km/sec respectively. Again, a total of 1000 Monte Carlo samples were simulated and the results were averaged across them. It was expected that target search methods will converge the sensor's boresight to the candidate orbits that are closest to orbit of the target. The 'target not in candidate set' scenario is visually described in Figure 2.5. It is expected that the target search schemes will try to point their boresight to the candidates closest to the target.

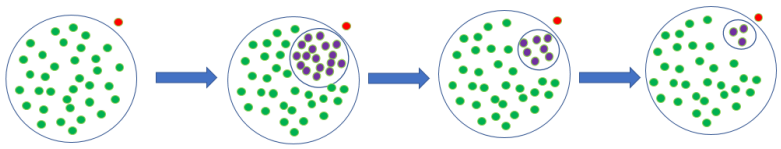


Figure 2.5: Target not in candidate sequential evaluation

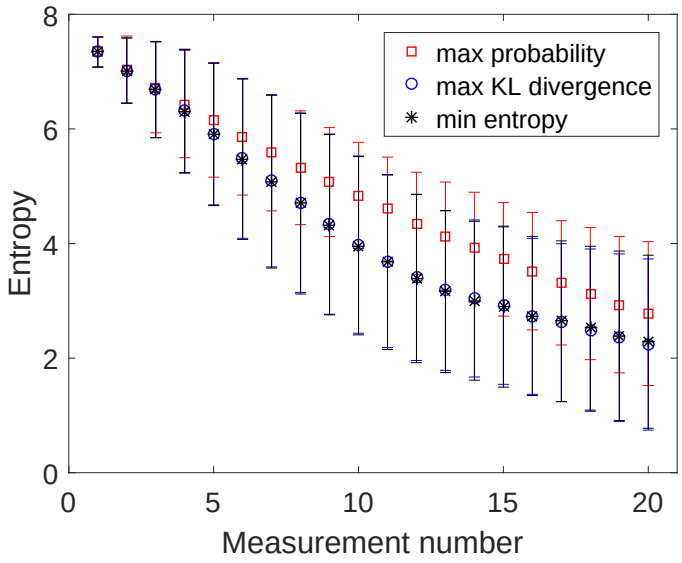


Figure 2.6: Entropy values of each target search method

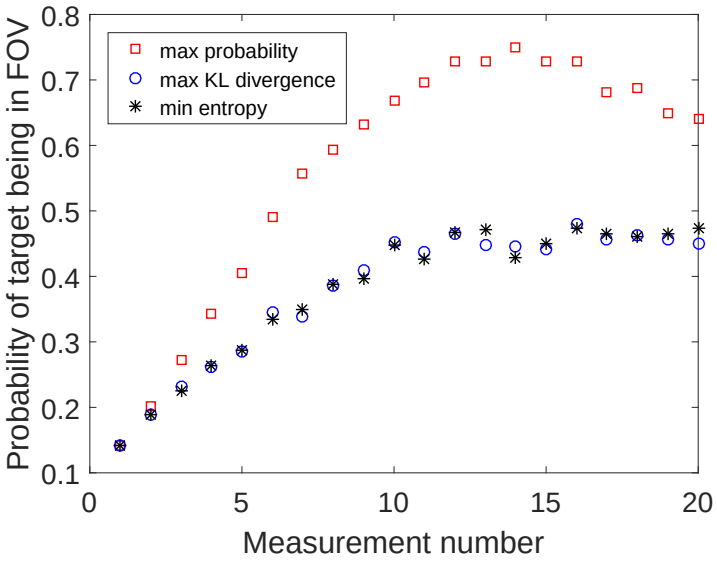


Figure 2.7: Probability of target being in the sensor's field of view

In consistency with the case of target in candidates, over the course of an observation campaign of 20 measurements, entropy of the information-theoretic methods is lower than the maximum probability tasking method. This implies that the information gain in the information theoretic methods is greater than the maximum probability search method. Similarly, the maximum probability tasking method gets the target in its field of view more often than the information-theoretic methods. These results are shown in Figure 2.6 and Figure 2.7, respectively.

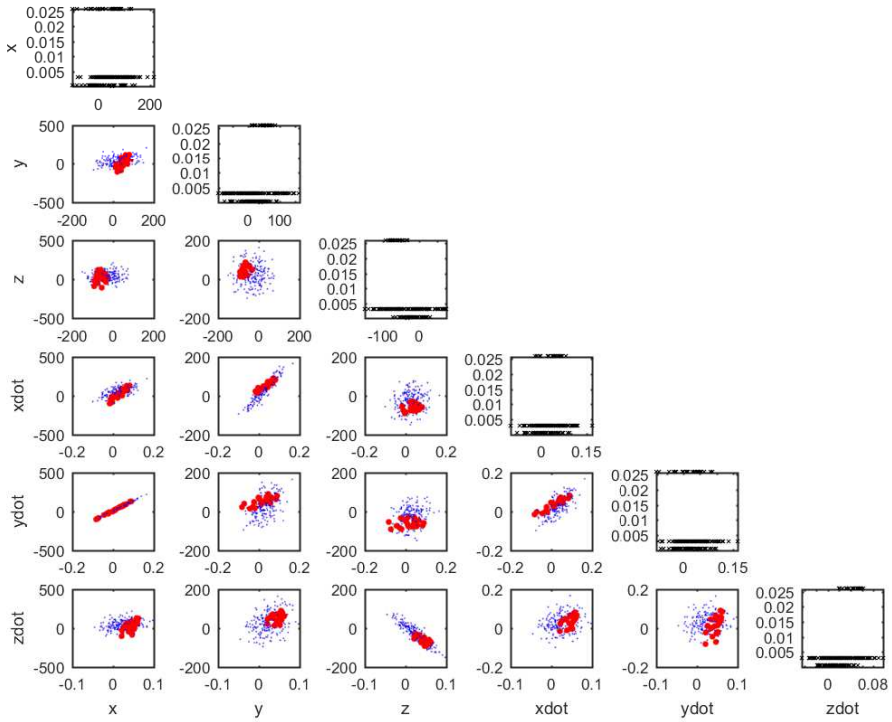


Figure 2.8: Probability distribution for maximum probability search after 5 observations

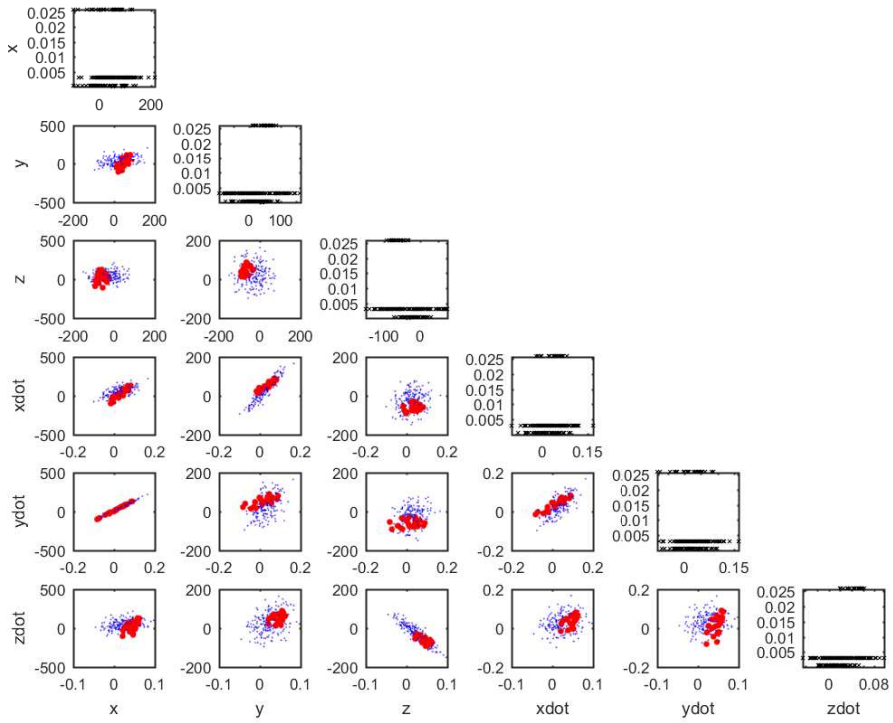


Figure 2.9: Probability distribution for maximum KL divergence search after 5 observations

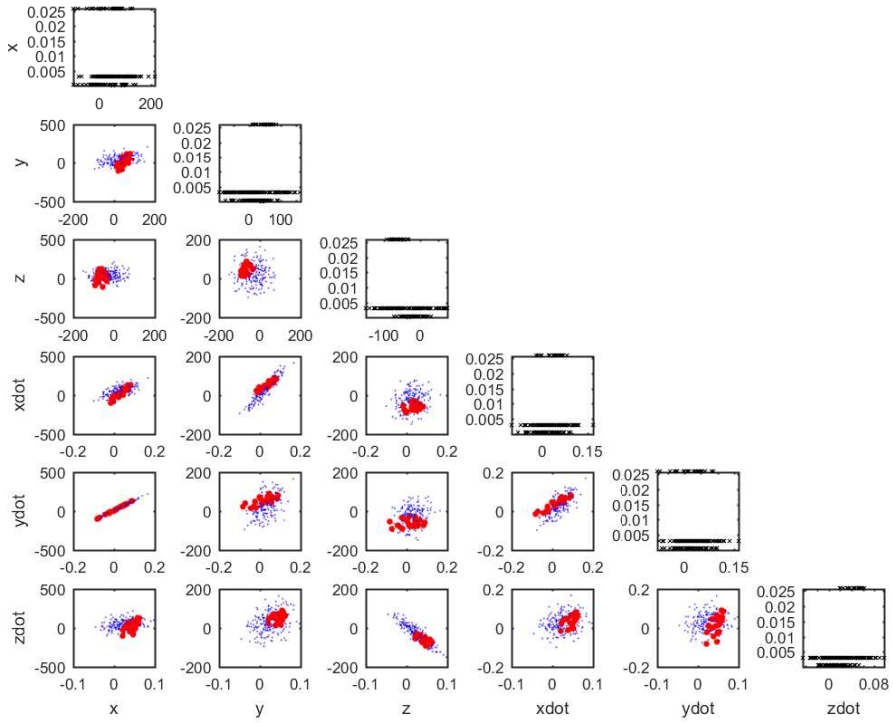


Figure 2.10: Probability distribution for minimum entropy search after 5 observations

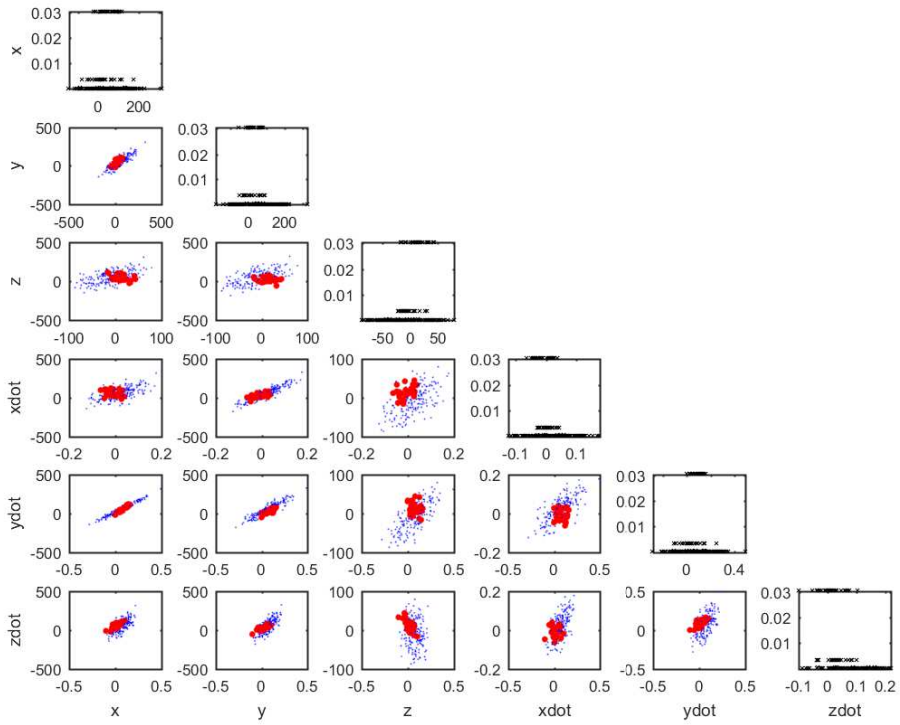


Figure 2.11: Probability distribution for maximum probability search after 10 observations

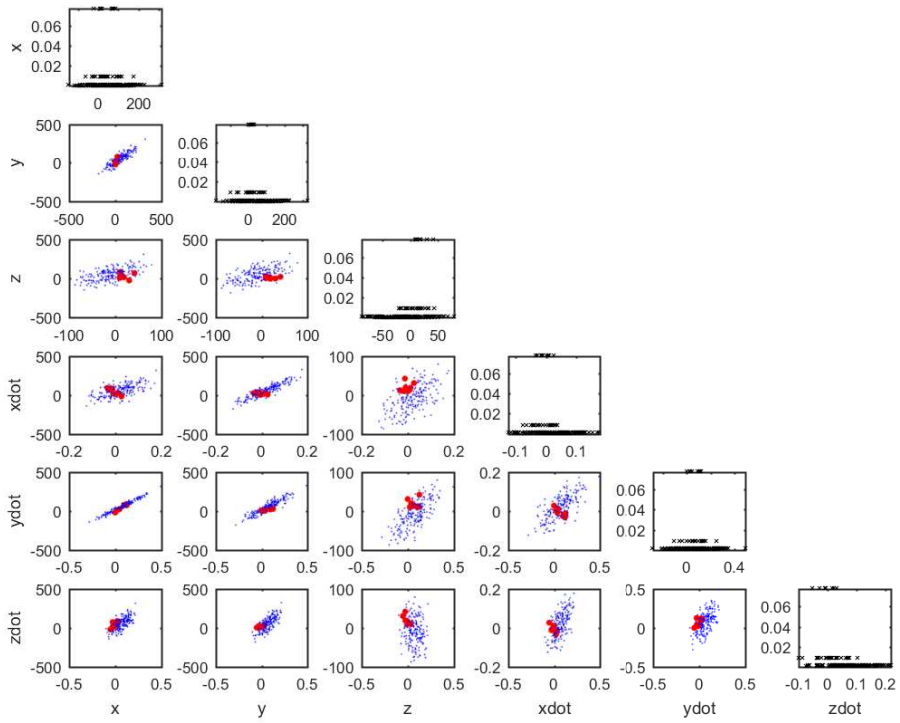


Figure 2.12: Probability distribution for maximum KL divergence search after 10 observations

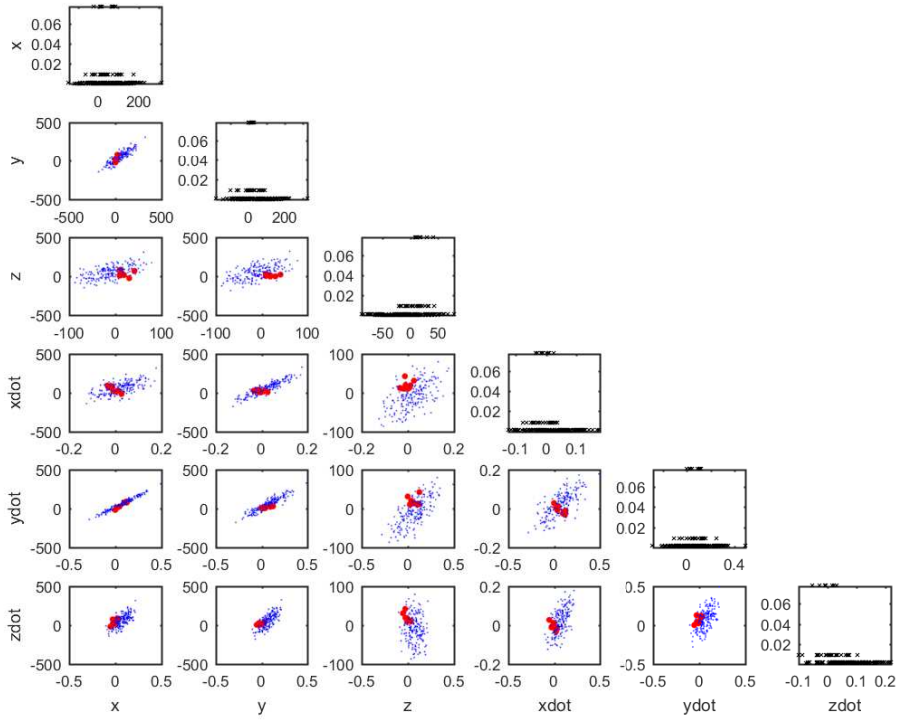


Figure 2.13: Probability distribution for minimum entropy search after 10 observations

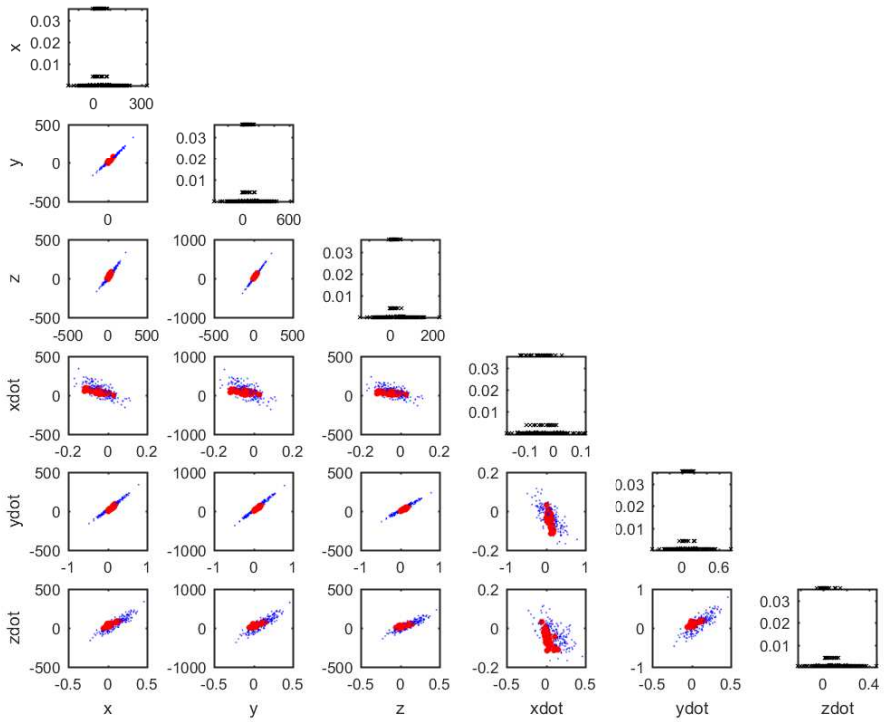


Figure 2.14: Probability distribution for maximum probability search after 15 observations

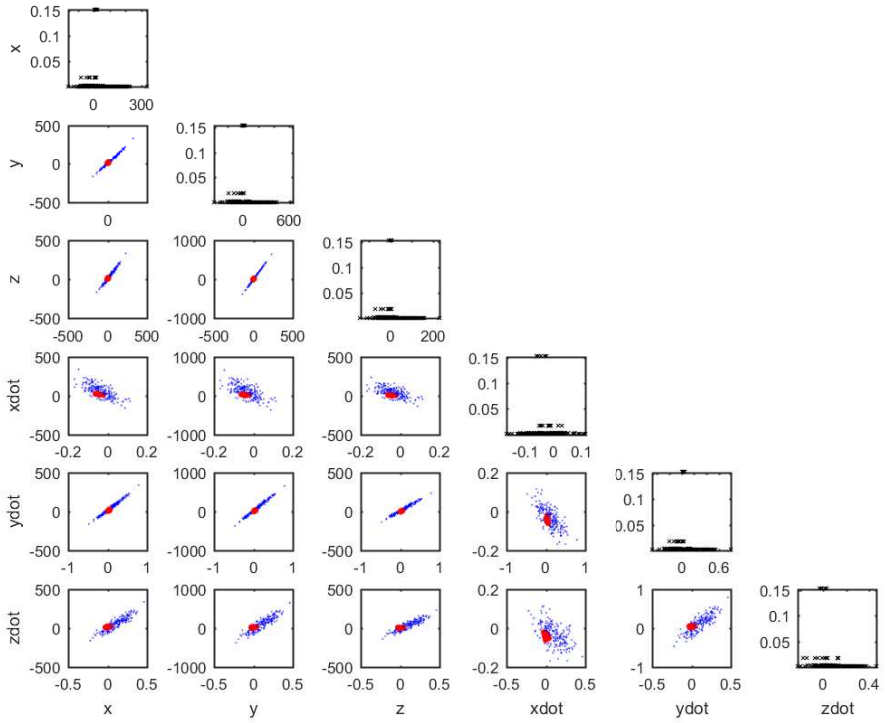


Figure 2.15: Probability distribution for maximum KL divergence search after 15 observations

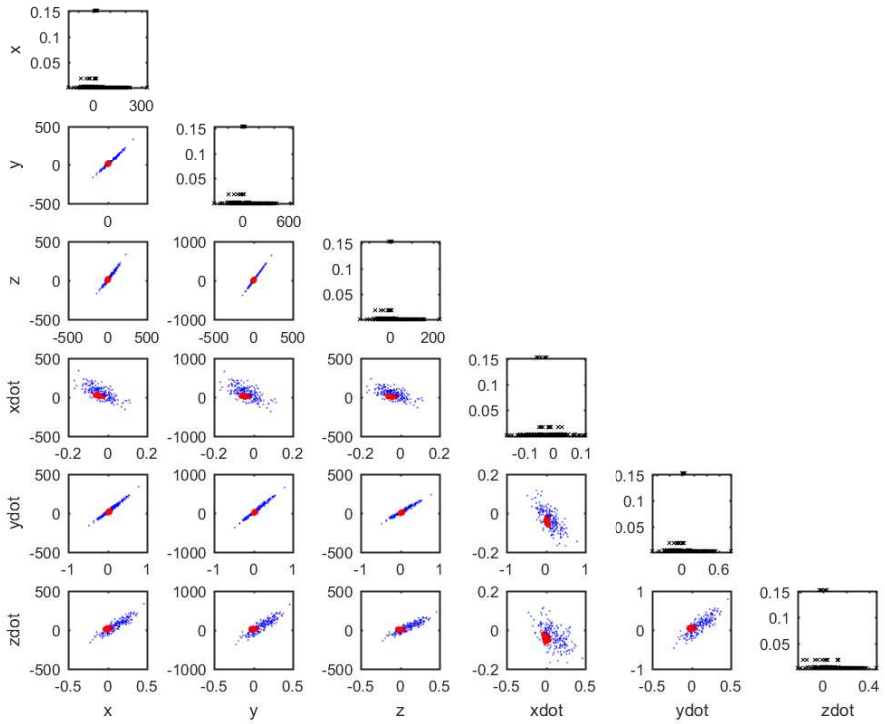


Figure 2.16: Probability distribution for minimum entropy search after 15 observations

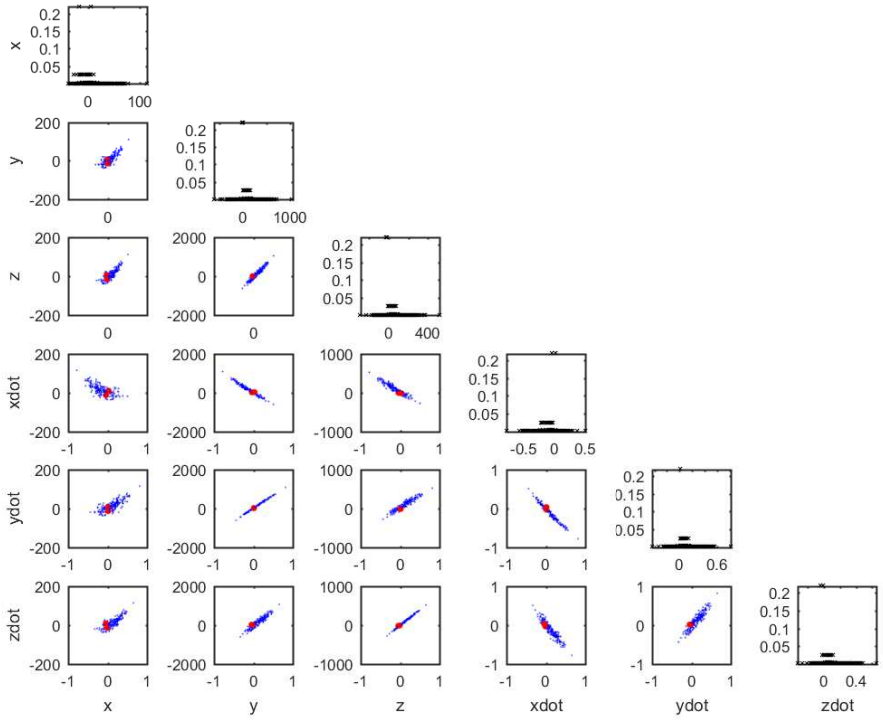


Figure 2.17: Probability distribution for maximum probability search after 20 observations

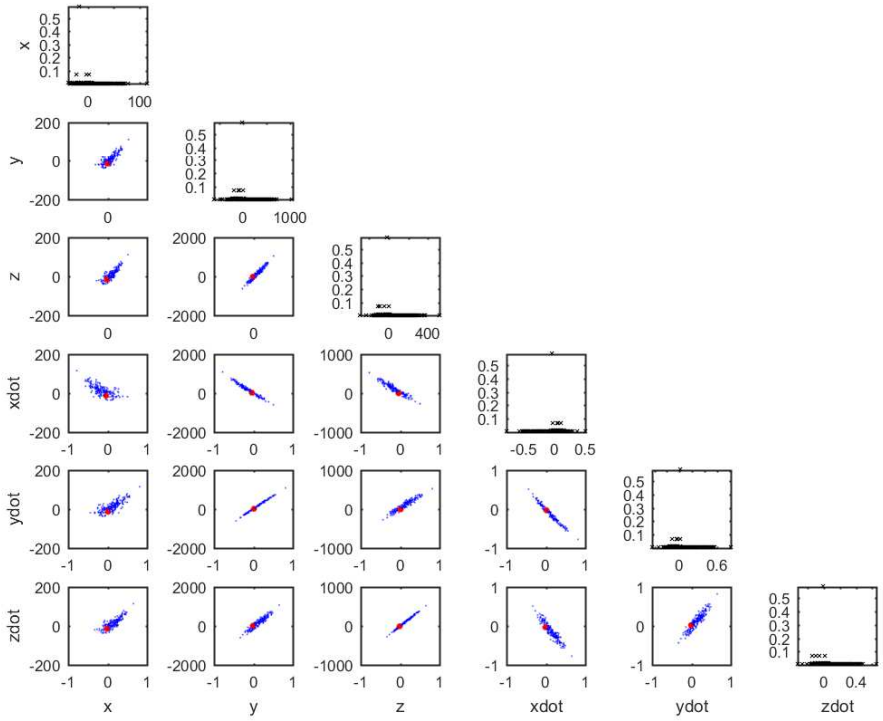


Figure 2.18: Probability distribution for maximum KL divergence after 20 observations

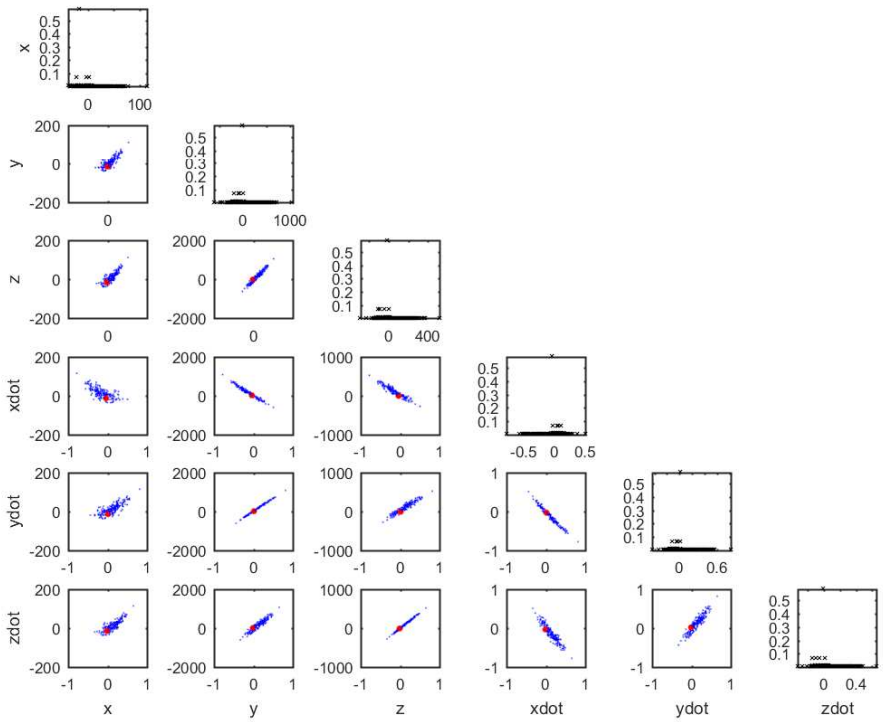


Figure 2.19: Probability distribution for minimum entropy search after 20 observations

In Figures 2.8 to 2.19, visualizations of the probability distributions of the three target search methods are shown. The results from four measurements taken at the 5th, 10th, 15th and 20th observation, respectively, are shown in the plots. Each figure contains 21 plots. Each of them visualize the candidate states relative to the target state i.e. $x_j(t) - x^t(t)$. Therefore, the candidates close to the target orbit are closer to zero on the plots. The plots on the diagonal show the relative candidate states with the associated probabilities on the vertical axis. The remaining plots show the values of one state versus the value of the other state for each candidate. The candidate probabilities are correlated to the marker size and color, on a color scale with low probabilities shown in blue and high probabilities shown in red. (Note that the marker size and color scales are normalized for each figure.) Overall, we expect the target search methods to show the candidate orbits in red, close to zero, and with high probability values in the plots on the diagonal.

From these figures, over the course of the 20 measurement observation campaign, several inferences can be made. Information-theoretic target search methods reduce the size of the candidate probability distribution with a higher rate than the maximum probability target search method. This is observed by noting that the number of candidates with higher probabilities on the diagonal plots decrease faster for the two information-theoretic target search methods than the maximum probability target search method. Information-theoretic target search methods identify the candidates closer to the target orbit with a higher certainty. This is apparent, especially in results for observation 20. In the case of maximum probability target search method, the candidates with highest probabilities have values at around 0.25 while the two information-theoretic target search methods put the accurately identified candidates at probability values close to 0.5.

2.4 Relationship between minimum Entropy and maximum KL divergence

Note that in the results shown above, min entropy and max KL divergence behave very similarly. The only discrepancies between the two may be resulting from numerical errors. This means that minimizing entropy and maximizing KL divergence conditioned on measurement types is the same. This can be seen analytically by expanding the formula for KL divergence.

$$
\begin{aligned}
\mathbb{E}[D_{KL}] &= \sum_{y}\sum_{x} p(x_k|y_{1:k}) \log \frac{p(x_k|y_{1:k})}{p(x_k|y_{1:k-1})} \\
&= \sum_{y}\sum_{x} p(x_k|y_{1:k}) \left[-\log p(x_k|y_{1:k-1}) + \log p(x_k|y_{1:k})\right] \\
&= H(x_{k|1:k-1}) - H(x_k|y_{k|1:k-1})
\end{aligned}
\tag{2.9}
$$

The detailed explanation of Equation 2.9 can be found in [12]. The expansion of KL divergnence conditioned on the measurements is equal to entropy given all previous measurements minus the conditional entropy at the present time step given all the previous measurements. Thus maximizing KL divergence conditioned on measurement types is equivalent to minimizing conditional entropy at the present time step given all previous measurements.

Chapter 3

Information-theoretic target search using K-means clustering

In this chapter, we introduce methods to relieve some of the computation costs involved in the target search methods described in the previous chapter. Earlier, we saw that in order to choose an optimal candidate to the point the sensor at, we must first fictionally point the sensor at each candidate to compute the expectation of the objective function, D_{KL} for example. We then proceed to point the sensor at the candidate where D_{KL} is maximum. An opportunity now arises to see if the requirement of pointing the sensor fictionally at each candidate can be relaxed. We approach to solve this problem by clustering the candidate orbits. In particular, we show that by clustering the candidate orbits prior to each measurement step, we can reduce the number of pointing directions available to the sensor. This can be seen visually in Figure 3.1 and Figure 3.2.

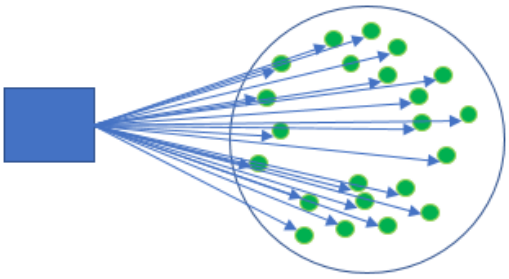


Figure 3.1: Brute-force sensor pointing options

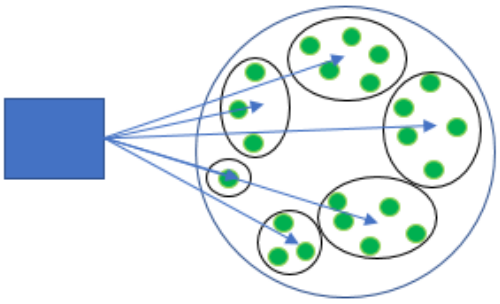


Figure 3.2: Sensor pointing options after candidate clustering

A well-known data clustering method called the K-means clustering is used to group the candidate orbits. It is observed that after clustering the candidates, along with reducing the number of pointing

locations available to the sensor, the accuracy of truly detecting the target is also higher when compared to just using the 'top x%' of candidate orbits as available pointing locations. The 'top x%' method is described later on in this chapter.

3.1 K-means clustering

K-means clustering is a standard algorithm that is widely used for data-mining in finance, feature detection in computer vision and vector quantization in signal processing. The basic idea of the algorithm is to group a set of data points based on their relative distances from each other. Mathematically, it is an optimization problem which can be stated as following:

Given a set of data points $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$, compute a set of clusters $y = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_k\}$ with $k < n$, while minimizing the relative distances between the cluster centroids and the data points:

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^j - c_j \right\|^2 \tag{3.1}$$

where c_j is the centroid of the cluster j. The algorithm that solves this optimization problem is quite simple and intuitive. The only requirement is that the algorithm needs a number k, the number of clusters that you want the data set to be grouped in. Here is a pseudo-code that solves the clustering problem.

Algorithm 4 General K-means clustering algorithm

```
1: procedure K-MEANS CLUSTERING(X, K)
2:     for i = 1:length(X) do
3:         Randomly assign k components of Y in space
4:         Compute euclidean distance from each component of Y to each data point in X
5:         Assign data points to the nearest centroid (component in Y)
6:         Compute the cluster centroid location by taking average of the coordinates of
           the data points assigned to that cluster
7:         Repeat steps 4-6 until the data points don't change their cluster allocation
8:     end for
9: end procedure
```

In Figure 3.3 on the next page, a set of data points were clustered using the K-means clustering algorithm. The cluster centroids are marked with black stars while the different clusters are marked with different colors.

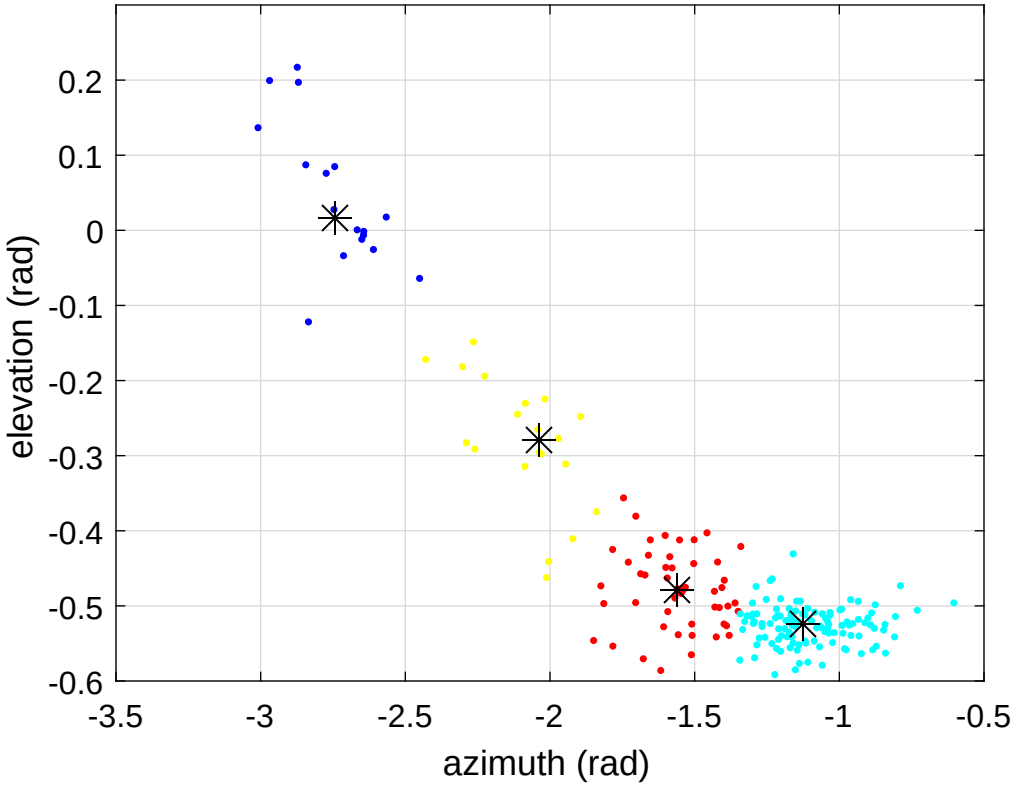


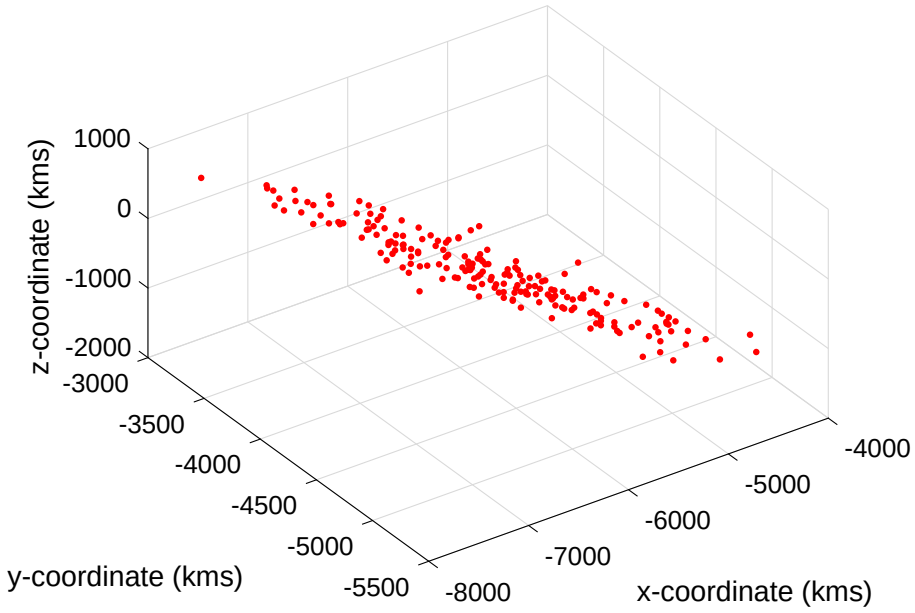
Figure 3.3: A sample visualization of data clustering using K-means algorithm

3.2 K-means clustering of candidate orbits

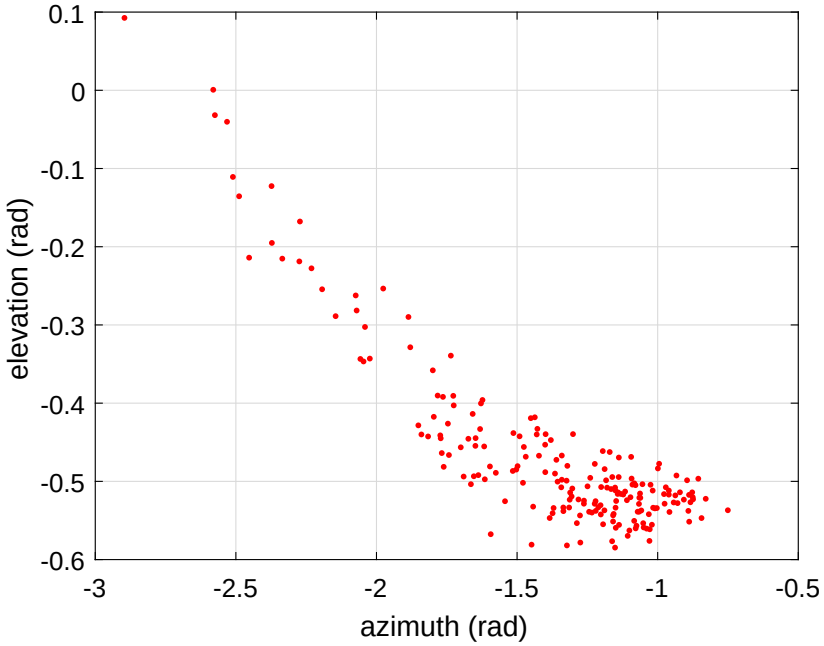
3.2.1 Coordinate transformation

In the target search problem, we have been using the particle based method i.e. each candidate orbits are represented as individual particles. Therefore, each particle at every instant of time can be treated as a data point in the above algorithm. In previously mentioned methods, we represented each candidate in three dimensions in space and assumed that the orbiting sensor has infinite range. These two characteristics of the problem must be carefully considered when clustering the candidates. It is much more convenient if 3 dimensional positions of the candidates can be transformed to a 2 dimensional plane. This can be achieved if we convert the 3 dimensional euclidean coordinates of candidates into three dimensional spherical coordinates. This is equivalent to projecting each candidate positions to surface of sphere centered on the sensor. Since, the range of the sensor is assumed to be infinite, we can ignore the radial components transformed coordinates of the candidates. We can then 'unwrap' the spherical surface and get the the candidate coordinates in the 2 dimensional angular coordinates. The following gives an example of a sample

coordinate transformation.



(a) Cartesian representation of candidates after 50 minute orbit propagation



(b) 2-D Spherical coordinate representation of candidates without the radial dimension

Figure 3.4: An example of candidate orbits coordinate transformation

3.2.2 Choosing an optimal number of clusters

Fitting an optimal number of clusters to a given data set is an optimization problem in itself. The number of clusters required is very much dependent on the 'spread' of data set. In traditional data clustering problem, there are many techniques to initialize k and iteratively converge to an optimal number of k. Elbow method [13] and Calinski-Harabasz [14] are a couple of well known methods. However, in the target search problem, we can take advantage of sensor's field of view and relate it to the spread of candidates at any given time. Relating these two quantities can give us an important implication in figuring out how many clusters we might need. For example, if we know that the sensor has a field of view of x degrees and we know the 2 dimensional coordinates of the candidates, we can impose a constraint to find an optimal number of K so that the sensor field of view covers atleast y % of candidate orbits. Below is a simple psedo-code explaining the logic. The algorithm receives sensor position r_s, candidate orbit positions in 3 dimensions X_{cand} and initial guess for k, K_{init}, sensor FOV in radians and required percentage of candidate coverage by the sensor's FOV, cov_{req} .

Algorithm 5 Finding k for target search problem

```
procedure FINDING K(r_s, X_cand, K_init, FOV, cov_req)
    for i = 1:length(X_cand) do
        Subtract 3d positions of candidates from sensor positions to move the candidates around the sensor
        Convert the 3d positions of candidates to 2d spherical coordinates
        while current coverage < cov_req do
            Compute k clusters using K-means clustering algorithm
            for j = 1:length(X_cand) do
                Check if the distance between candidate centroid and its components is less than sensor's FOV
            end for
            If the required coverage is not met, increase K_init by 1
        end while
    end for
end procedure
```

3.3 Target search with K-means method

In this section, we incorporate the K-means clustering method into the information-theoretic and maximum probability target search methods. We consider a set $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$ of candidate orbits, each initialized at randomized initial positions and velocities around a nominal position r_{nom} and velocity v_{nom}. Along with position and velocities, particles weights or the particle probabilities are initialized with uniform distribution. The candidates are then propagated in an inverse square gravitational field around an orbit

around the Earth. Once the candidate orbits are propagated, a computation of number of required clusters can be computed using the Algorithm 5 at each time step. Note that the K-means clustering method also provides the mean of each clusters i.e. the cluster centroid or the average of the positions of cluster members. The average cluster position can be used as possible pointing location of sensor's boresight l_k. Now, as done previously in Chapter 2, using Equation 1.5, the number of candidates in the field of view are computed when the sensor is pointed ar each cluster center. Following these computations, the sensor takes the actual measurement at the pointing location where the D_{KL} is maximum or the sum of probabilities of the candidates is maximum in the maximum probability target search scheme. The probabilities of candidates are then updated using Equation 1.6.

3.4 Top x percent method

The 'top x%' method reduces the available pointing locations to the sensor by considering only the candidates with high probabilities. For example, if we trust and just consider the top 20% of possible candidates out of total of 200 candidates, we can sort them at each time step in descending order of their probabilities and choose just the top 40 candidates. These 40 candidates are now the only available pointing locations for the sensor.

As done previously, we can then evaluate the Kullback-Leibler divergence or maximum probability for their FOV, take measurements and update probabilities. Next, we compare the performance of K-means clustering method with 'top x%' method in the following section.

3.5 Simulation results

In this section, the K-means method is compared to the 'top x%' method. Here, we compare the two methods in the 'target not in candidates' case and employing max D_{KL} and max probability tasking schemes. The parameters of the simulation were similar to the ones listed in Table 2.1. The only addition was that the number of clusters,k, fixed to 32. Similarly, only the top 32 candidates were chosen as available pointing location for the 'top x%' method. To be consistent, we enforce the same number of available pointing locations in the 'top x%' case as dictated by number of clusters generated by the K-means method.

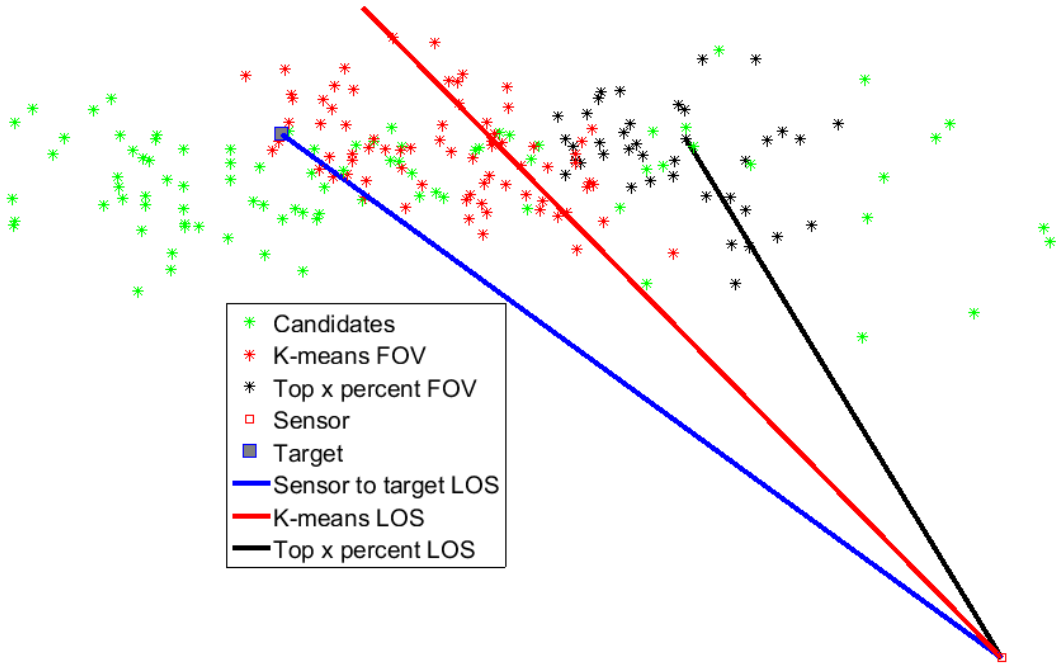


Figure 3.5: Sensor LOS chosen by maximizing KL divergence using K-means clustering and 'top x %' method at measurement step 10

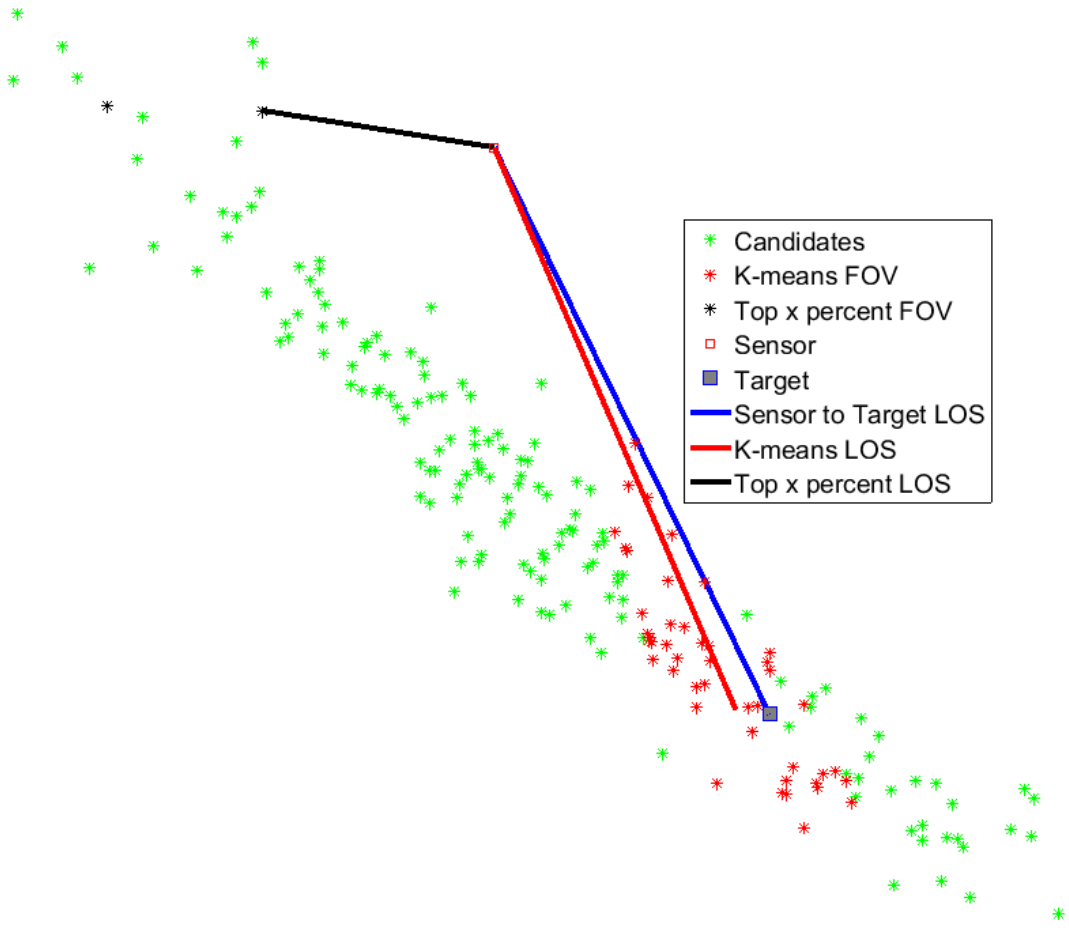


Figure 3.6: Sensor LOS chosen by maximizing KL divergence using K-means clustering and 'top x %' method at measurement step 30

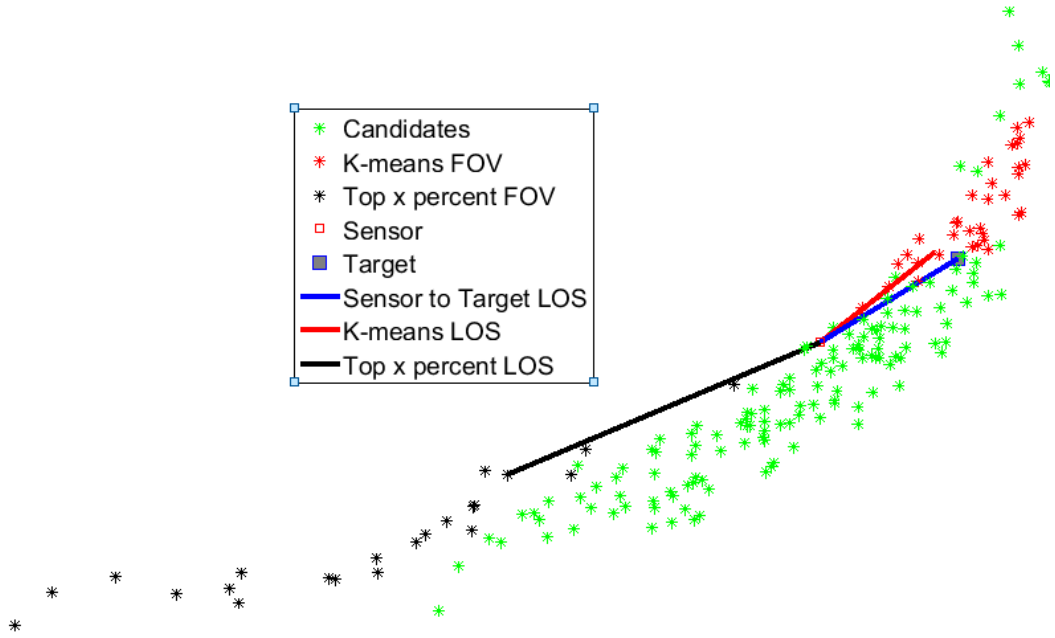


Figure 3.7: Sensor LOS chosen by maximizing KL divergence using K-means clustering and 'top x %' method at measurement step 50

In Figures 3.5 to 3.7 , the pointing direction chosen by the two methods at measurement steps 10,30 and 50 are shown. It can be seen that the K-means method points its LOS successfully towards the target while the 'top x%' method struggles to do so.

3.5.1 Target search with upper and lower bound of required number of clusters

In this section, we compare the K-means method to 'top x%' method using the upper bound and lower bounds of k. The candidate orbits were first propagated offline. Then, the time steps at which the candidates were separated the most and the least were computed. At these time steps, number of maximum required k's according to the field of view of the sensor were computed. These k's are the upper bound and lower bound of number of required clusters, respectively.

Method	Average Entropy	Average RMSE	Average Variance	Instances of finding target in FOV
K-means : max KL divergence	130.60	1.95	74.72	28.19
K-means : max probability	169.30	2.26	74.75	40.98
Top x % : max KL divergence	229.91	8.87	75.92	11.59
Top x % : max probability	274.66	8.84	75.73	12.23

Table 3.1: K-means and 'top x%' comparison using upper bound value of k

Method	Average Entropy	Average RMSE	Average Variance	Instances of finding target in FOV
K-means : max KL divergence	123.13	2.18	74.63	28.74
K-means : max probability	157.16	2.42	74.67	38.66
Top x % : max KL divergence	244.24	9.03	75.75	11.51
Top x % : max probability	269.64	8.82	75.60	12.54

Table 3.2: K-means and 'top x%' comparison using lower bound value of k

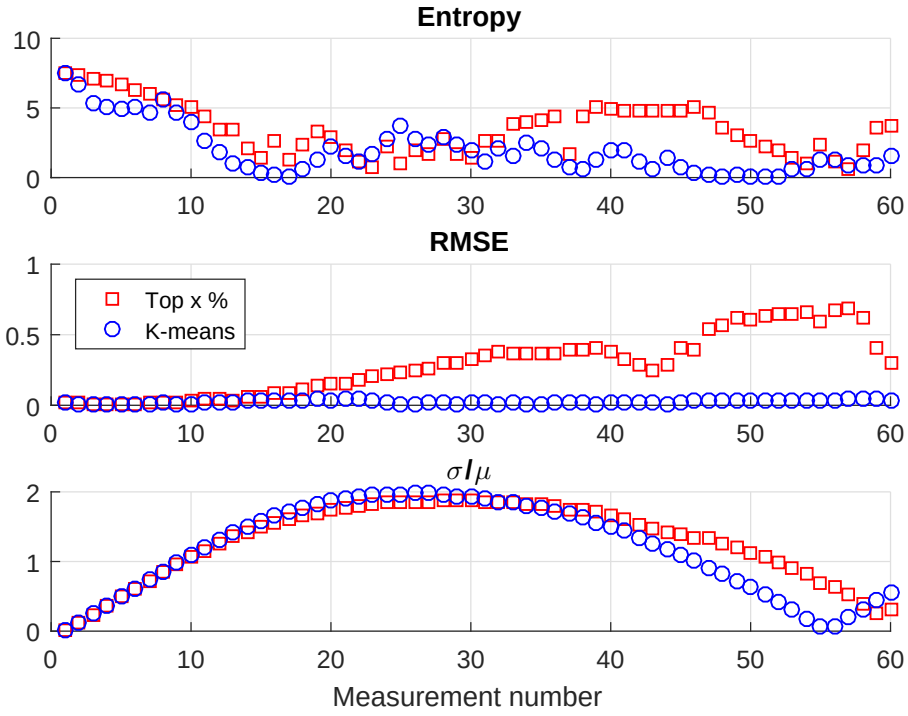


Figure 3.8: Entropy, Root mean square error and Coefficient of variation of one monte carlo sample using upper bound of k

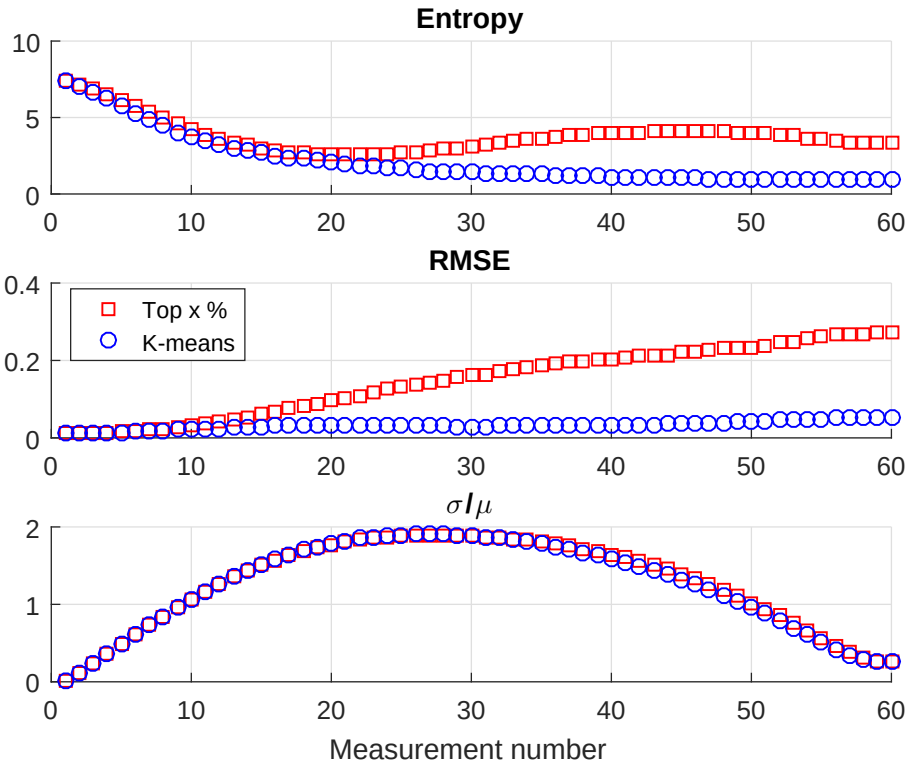


Figure 3.9: Entropy, Root mean square error and Coefficient of variation across all monte carlo samples using upper bound of k

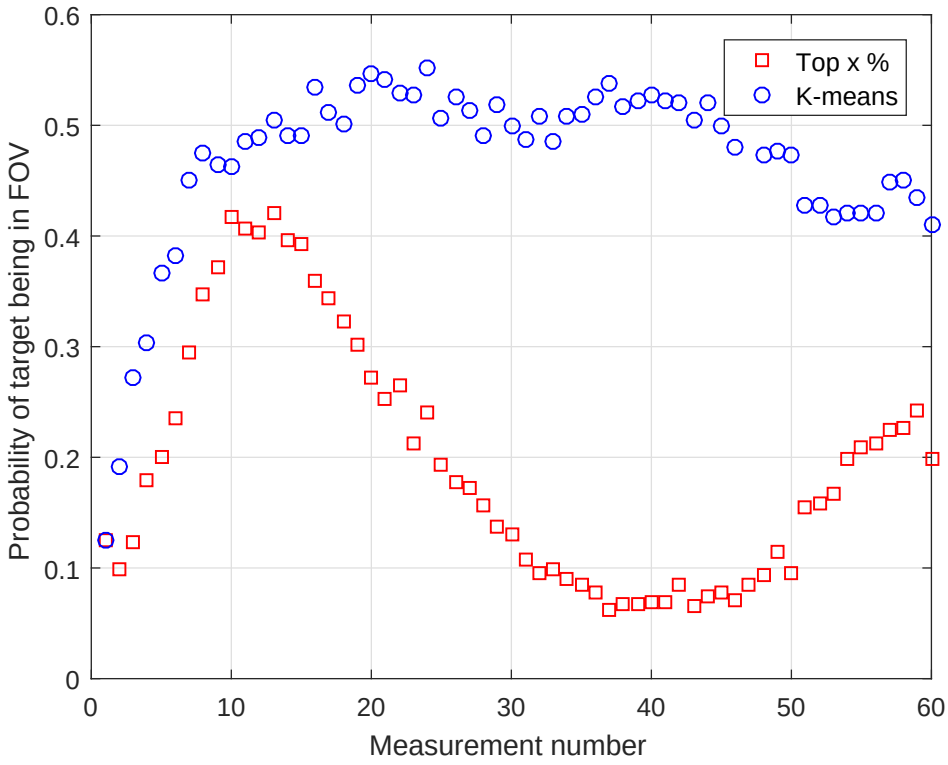


Figure 3.10: Field of view using upper bound of k

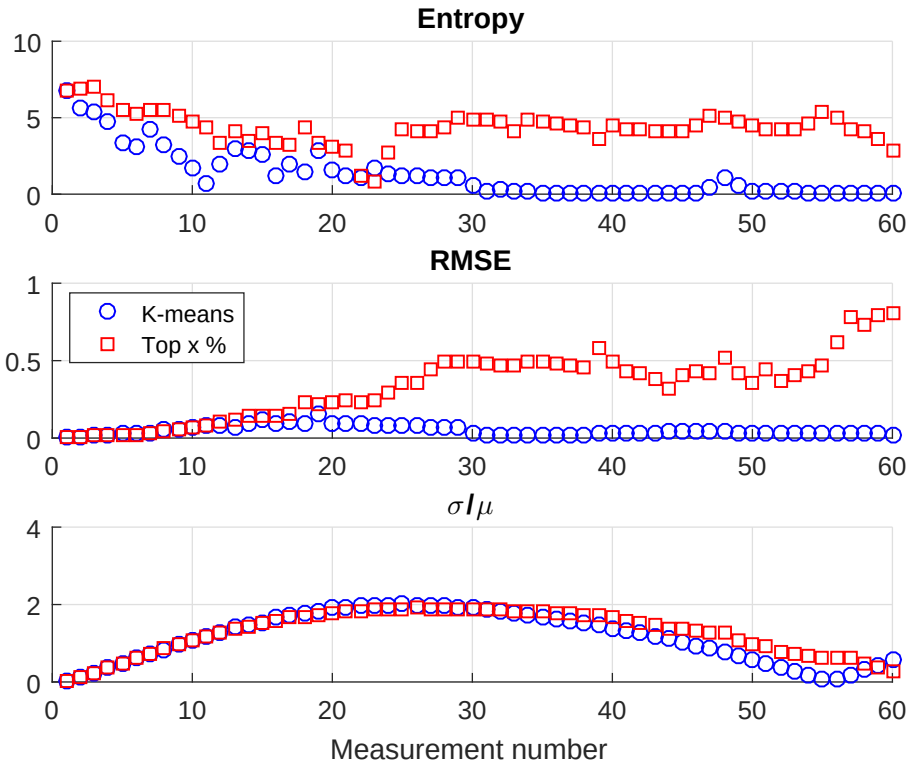


Figure 3.11: Entropy, Root mean square error and Coefficient of variation of one monte carlo sample using lower bound of k

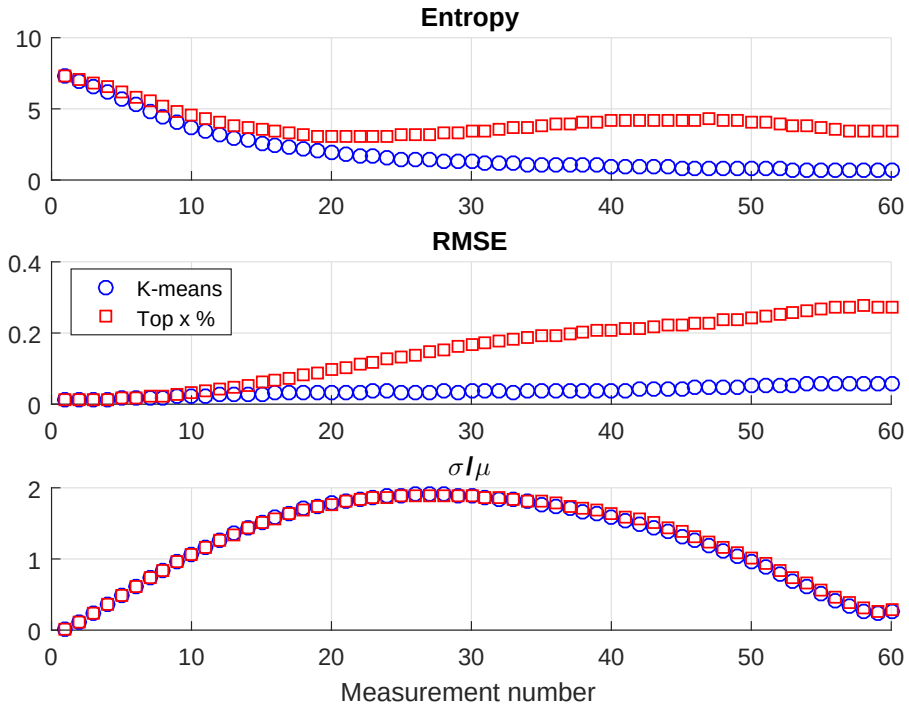


Figure 3.12: Entropy, Root mean square error and Coefficient of variation across all monte carlo samples using lower bound of k

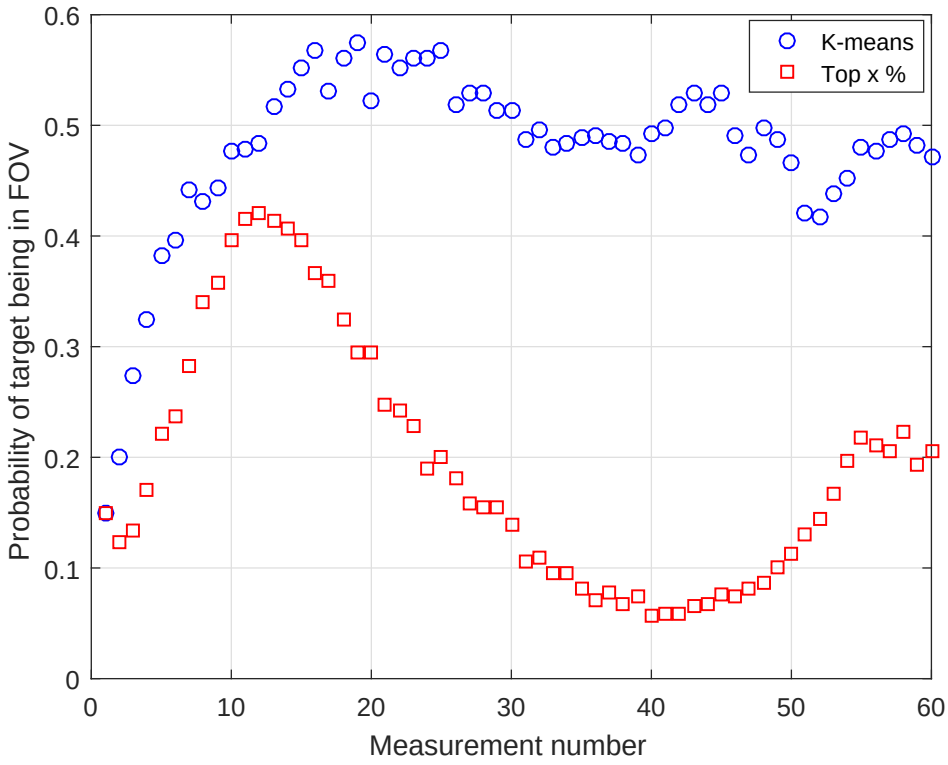


Figure 3.13: Field of view using lower bound of k

It can be seen that the K-means method produces lower entropy values compared the 'top x%' method. That means K-means method lowers the uncertainty of target location with a faster rate than the 'top x%' method. Additionally, we compute the root mean square error i.e. the difference between where the target was and where each method thought the target was. It can be seen that the RMSE error is much lower in the K-means method compared to the 'top x%' method. The RMSE error values are in coordination with the FOV figures in which the probability of target coming in sensor's FOV is higher in the K-means method compared to the 'top x%' method.

Chapter 4

Conclusions

In this thesis, we presented methods to search a target with a space based sensor for Space Situational Awareness. In Chapter 2, we compared information theoretic methods; max KL divergence and min entropy, to a greedy max probability method. We see that the information theoretic methods reduce uncertainty at a faster rate compared to the max probability target search method. On the other hand, chances of getting the target in FOV is much higher with max probability method compared to information-theoretic methods. We observe these behavior by simulating two scenarios. One in which the 'target is in candidates' and the other when 'target is not in candidates'.

In Chapter 3, we set off to reduce available pointing locations to the sensor in order to minimize the computational load. We do so by clustering the candidate orbits at each time step by using the K-means clustering algorithm. The pointing locations are then reduced to cluster centroids. Following the clustering process, we employ the same information-theoretic and max probability methods that were employed in Chapter 2. The K-means clustering method is compared to another method, the 'top x% method'. The 'top x%' method reduces available pointing locations by restricting them only to x% of total candidates with descending probabilities. The task is again set to minimize or maximize the objective functions mentioned in Chapter 2. The two methods are compared and the simulation results show that target search after clustering the candidates using K-means algorithm produces better results than taking measurements are candidates with high probabilities.

One possible future direction is optimal switching between clustering and particle sampling. This is attractive because through simulations, we have observed that the 'top x%' method does better when the candidate orbits are close to each other. However, due to orbital dynamics, when the candidate orbits spread out, the clustering method does better. So the question remains of how to optimally switch between the two methods during an observation campaign. Another possible future direction deals with the modeling of the sensor itself. Currently, the sensor is modeled using four deterministic quantities which quantifies the performance of the sensor. This primitive model of the sensor can be improved to portray the sensor more realistically.

References

[1] R. Erwin, P. Albuquerque, P. Jayaweera, and I. Hussein. Dynamic sensor tasking for space situational awareness. In *Proceedings of American Control Conference*, pages 1153–1158, Baltimore, MD, June 2010.

[2] Z. Sunberg, S. Chakravorthy, and R. S. Erwin. Information space receding horizon control for multi-sensor tasking problems. *IEEE Transactions on Cybernetics*, pages 1325–1336, June 2016.

[3] I. I. Hussein, M. K. Jah, and R. S. Erwin. An aegis-fisst sensor management approach for joint detection and tracking in ssa. In *AAS/AIAA Spaceflight Mechanics Meeting*, Kuai, HI, February 2013.

[4] K. J. DeMars, I. I. Hussein, C. Fruh, M. Jah, and R. S. Erwin. Multiple object space surveillence tracking using finite state statistics. *AIAA Journal of Guidance, Control, and Dynamics*, pages 1741–1756, 2015.

[5] Z. Sunberg, S. Chakravorthy, and R. S. Erwin. Information-space receding horizon control. *IEEE Trans. Systems, Man and Cybernetics - Part B*, pages 2255–2260, 2013.

[6] P. Williams, P. Spencer, and R. Erwin. Coupling of estimation and sensor tasking applied to satellite tracking. *AIAA Hournal of Guidance, Control and Dynamics*, pages 993–1007, 2013.

[7] I. I. Hussein, Z. Sunberg, S. Chakravorthy, M. K. Jah, and R. Erwin. Stochastic optimization for sensor allocation using aegis-fisst. In *AIAA/AAS Space Flight Mechanics Conference*, Santa Fe, NM, January 2014.

[8] M. J. Gualdoni and K. J. DeMars. An improved representation of measurement information content via the distribution of the kullback-leibler divergence. In *AAS/AIAA Astrodynamics Specialist Conference*, Stevenson, WA, August 2017. AAS 17-810.

[9] T. A. Hobson and I. V. L. Clarkson. An experimental implementation of a particle-based dynamic sensor steering method for tracking and searching for space objects. In *2014 IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP)*, 2014.

[10] T. S. Murphy, M. J. Holzinger, K. K. Luu, and C. Sabol. Optical sensor follow-up tasking on high priority uncorrelated track. In *AAS/AIAA Spaceflight Mechanics Meeting*, San Antonio, TX, February 2017.

[11] K. Hill, P. Sydney, K. Hamada, R. Cortez, K. Luu, M. Jah, P. W. S. Jr., M. Coulman, J. Houchard, and D. Naho'olewa. Covariance-based network tasking of optical sensors. In *AAS/AIAA Spaceflight Mechanics Meeting*, San Diego, CA, February 2010.

[12] E. H. Aoki, A. Bagchi, P. Mandal, and Y. Boers. A theoretical look at information-driven sensor management criteria. In *Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on*, pages 1–8. IEEE, 2011.

[13] T. M. Kodinariya and P. R. Makwana. Review on determining number of cluster in k-means clustering. *International Journal*, **1**(6):90–95, 2013.

[14] U. Maulik and S. Bandyopadhyay. Performance evaluation of some clustering algorithms and validity indices. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**(12):1650–1654, 2002.