8-2018

Estimating the location of a Nuclear Source with Multiple Drones in an Urban Environment

Upasana Bhattacharyya
Clemson University, ubhatta@g.clemson.edu

Estimating the location of a Nuclear Source with Multiple Drones in an Urban Environment

---

A Dissertation
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
Electrical Engineering

---

by
Upasana Bhattacharyya
August 2018

Accepted by:
Dr. Carl W. Baum, Committee Chair
Dr. Harlan B. Russell
Dr. Kuang-Ching Wang
Dr. Brook T. Russell

Abstract

The problem of locating the source of radioactive emissions using a network of sensors is considered. Estimating the three-dimensional location of a nuclear source is especially difficult in environments in which no sensor can be placed in close proximity to the source. In this dissertation, maximum-likelihood (ML) estimation is applied to a Poisson process model for radiation received at sensors that is proportional to the inverse square of the distance between the source and the sensor. The joint multivariate density for the sensors is then maximized in order to estimate the location and strength of the radioactive source. Additionally, a limited number of sensors is used to implement a two-stage adaptive algorithm. In the first stage the drones sit at the center of a building's faces and an approximate location of the radiation source is obtained. Based on the results of the first stage, in the second stage the drones move to additional locations to collect more data. The data from both stages is utilized to obtain a more accurate estimate of the location of the radiation source. A third topic involves the effects of spatially non-homogeneous attenuation due to highly absorbing materials such as concrete. A novel metric is presented for identifying situations in which non-homogeneity significantly skews estimation results. This metric is used to drive a multiple iteration multi-stage estimation algorithm utilizing multiple applications of ML estimation. The algorithm is analyzed in realistic situations such as highly absorbing walls and a central shaft. Finally, a hybrid algorithm is proposed

that first determines with a high degree of reliability whether non-homogeneous attenuation is present. If non-homogeneous attenuation is declared absent, the sensors move according to the adaptive algorithm. If non-homogeneous attenuation is declared present, the multiple-iteration algorithm is employed. This hybrid algorithm performs extremely well whether non-homogeneous is present or absent.

Acknowledgments

I would like to thank my advisor, Dr. Carl W. Baum, for his guidance and support throughout my graduate studies at Clemson University. I am grateful to Dr. Harlan B. Russell, Dr. Kuang-Ching Wang, Dr Brook Russell and Dr John N. Gowdy for serving on my committee and providing their valuable feedback.

Finally, I would like to thank my parents and my fiancé for their love and unwavering support.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

With growing risk of contraband radioactive material, low-cost spatially distributed radiation detectors have emerged as an important topic in reconnaissance research. Gamma photons emitted by radioactive sources are registered at detectors at a rate dependent on the source intensity and its distance from the detector. Using sensor fusion algorithms to combine the data collected by a network of sensors, source parameters can be estimated to perform effective localization. The risk of terrorist attacks using improvised nuclear devices has led to growing concern in recent years [26]. With increased interest in the application of drones for various defense purposes, researchers have investigated the plausibility of using many kinds of unmanned aircrafts in a variety of scenarios [8, 11, 12, 29, 34, 38]. The use of a single autonomous helicopter is considered in [34] for obtaining overhead images, gathering radiation measurements, and mapping both the structural and radiation content of the environment in a post-disaster scenario. Mapping radiation in a post-disaster scenario in three dimensions is also considered in [18].

Much recent work such as [1, 4, 7, 19, 22, 28, 30, 32] concentrates on radiation sources in a *two dimensional* environment with unknown source intensity in which

the detectors can be placed in close proximity to the source. Some of these works such as [30] and [19] also take into account partial prior knowledge of the source location and strength. Localizing a radiation source in a three-dimensional search space is considered in [39]. However, in the simulation results, the author assumes that the source is on the same plane as the deployed sensors, essentially converting this to a two-dimensional localization problem. This paper also uses 100 sensors deployed in a 10×10 lattice within the periphery of the search space. In comparison, [7] uses 16 radiation detectors within the search space to localize the radiation source.

While the proposed methods in these papers generally work reasonably well in theory, in many real-world scenarios, the search for illicit radioactive material of unknown strength is necessarily conducted from the periphery of the search space. In fact, in urban environments, the search space is inherently three-dimensional. Additionally, deploying a larger number of radiation detectors is often not a cost-effective solution in real-world applications. These factors make the search problem significantly more difficult.

Some flexibility in the sensor locations can be achieved by utilizing small inexpensive drones with a payload consisting of low-cost radiation detectors, GPS, and basic communication devices providing TCP/IP or connectivity over cellular networks [29] in order to share measurements and location information. The sensors on these drones can be fitted with either portable Geiger counters or scintillation detectors [11, 34]. Because Gamma and high energy beta particles are highly penetrating radiations [6], these detectors are capable of detecting radiation activity even at low concentrations [29, 30].

Drones pose a unique advantage in offering the flexibility of placing sensors at the most optimum locations (in three-dimensional space) for data collection. Due to their mobility and portability, drones can respond quickly to intelligence about emerg-

ing threats or in response to data from other nuclear surveillance devices or networks. Moreover, drones enable surveillance procedures to be performed in contaminated areas without exposing personnel to radioactive materials.

Both classical and Bayesian approaches to estimation have been considered in [1, 7, 8, 10, 12, 19, 20, 29, 34] for the general problem of estimating locations of radioactive sources. In general, classical approaches include minimum variance unbiased (MVU) estimation [13] and maximum likelihood (ML) estimation [9]. Due to the difficulty in determining the MVU estimator, to our knowledge it has not been employed for nuclear source estimation. In contrast, ML estimation has been used extensively. The ML approach is applied in [8] to estimate the number of nuclear sources present in a search region and each of their locations. Additionally, ML estimation is used in conjunction with the Expectation Maximization (EM) algorithm to track multiple radiation sources in [7]. Attenuation is not incorporated into the model in [8]; exponential attenuation through a homogeneous medium is modeled in [7], however.

An important property of ML estimation is that as the sample size tends to infinity, performance asymptotically achieves the Cramer-Rao Lower Bound (CRLB) [5, 27]. The CRLB can be useful as a predictor of performance under asymptotic conditions, but it does not claim to predict the performance when the sample sizes are relatively small. The CRLB is used in [10] to quantify the accuracy with which it is possible to find the two-dimensional location and strength of the radiation source. The results are extended in [19] to multiple sources. The models are two-dimensional and do not incorporate the effects of attenuation. The work shows that ML estimates sometimes does not converge for small values of SNR. Furthermore, the results show difficulty in estimating the strength of the sources even under high SNR in some cases.

In contrast to classical approaches, the Bayesian approach to estimation assumes knowledge of the prior distribution of unknown parameters. The goal of the

Bayesian estimator, using a minimum mean square error (MMSE) criterion, is to calculate the conditional expected value of the unknown random variable given the available data [3]. In [20] a Bayesian approach is implemented for estimating multiple moving sources in a two dimensional field assuming a known number of sources and a known prior distribution of the location of each source. Because the conditional expected value is intractable, the authors in [20] approximate the conditional expected value using importance sampling [21]. The focus of the results in [20] are on sensors placed along a single stretch of road. Thus the distances between the source and sensors are small and the motion is highly constrained.

Recursive Bayesian estimation (RBE) is an implementation of Bayesian estimation in which the priors are sequentially updated as new data is obtained. If a priori knowledge is available and statistical models are suitably simple (e.g., finding the mean of a Gaussian random vector), this method can be easily implemented using matrix multiplication and convolution algorithms. Although the RBE can be reasonably effective in locating the two dimensional position of the source [34], it suffers from the "curse of dimensionality" such that an increase in any dimension of the search space causes an exponential increase in the time and memory required for the search process. A grid-based RBE is implemented in [34] assuming prior knowledge of the isotope, the number of sources, and the strength of each source. RBE is also used in [10] to locate a single radiation source with known strength in a two-dimensional search space. The posterior PDF of the modeled Gaussian distribution is approximated using both the extended Kalman Filter (EKF) and the Unscented Kalman Filter (UKF). (An overview of these filters can be found in [2, 35].) In reality, radiation data is non-Gaussian, and Bayesian estimation in non-Gaussian environments leads to intractable integrals as a part of the estimator. Importance sampling is implemented in [19] to approximate the integrals using a progressive correction method

in order to estimate the posterior density of the source.

Bayesian *detection* has also been considered for the nuclear problem. For example, in [25], Bayesian detection is used to determine whether a moving vehicle contains nuclear material. The work assumes that the vehicle is tracked by means other than by radiation measurements.

A problem related to the one considered in this paper, but yet distinct, is that of providing a search strategy for one or more sensors seeking to find one or more stationary or mobile radiation sources hidden over a large area. Examples of papers exploring this problem are [33], [31], and [15]. This problem is formulated as a detection problem in [33], where the trajectory of a source is presumed to be known, and the goal (beyond determining whether a nuclear source is present) is to navigate sensors around obstacles so as to improve detection performance. In [15], the problem of directing a mobile radiation sensor to find multiple sources is considered. The approach taken uses an artificial potential field in conjunction with a particle filter that estimates the direction and strength of the attractive forces in the field. A particle filter is used in conjunction with sequential Bayesian estimation and a partially observed Markov decision process to estimate multiple sources and direct mobile sensors in [31]. From a national security perspective, both problems are important. However, finding a stationary source is more likely to avert a terrorist attack well prior to final implementation and is therefore more likely to successfully avert the attack.

Additionally, most of these papers do not take into account attenuation of the radioactive decay. Papers such as [1,10,19,20,29–31,33,34] do not model attenuation. While attenuation is insignificant in open spaces over very short distances, larger distances with material between source and sensor can exhibit a considerable amount of attenuation. If not taken into account, absorption and the corresponding attenuation can introduce considerable error in the parameter estimates of the source. Absorp-

tion and attenuation and their effects on tracking radioactive materials in crowded areas are explored in [37]. The paper investigates various shielding phenomena that affect the counts registered by sensors, including scattering, detector shadowing and exponential decay. The paper explores these models but does not consider specific algorithms for estimation of source location. Estimation with attenuation modeling for air is considered in [7]. However, this paper assumes a homogeneous absorption medium between source and sensor. Non-homogeneous attenuation due to a combination of materials such as air and dense concrete with high mass attenuation coefficients [24] has the potential to significantly skew results. To our knowledge, estimation in the presence of severe attenuation has not been explored.

In this dissertation, three-dimensional estimation of the strength and location of a single nuclear source is explored using ML estimation. It is assumed that the source lies inside a large building (or complex of buildings) and, as a result, sensors (on a network of drones) are unable to move closer to the source. Furthermore, the ML algorithm and the performance results both explicitly model the effects of air attenuation. In addition, the effects of non-homogeneous attenuation from buildings on the performance of the algorithm are also analyzed. The use of a small number of strategically placed sensors in spatially diverse locations is explored. It is shown that choosing the right locations for the sensors gives better localization performance compared to using a large number of sensors distributed uniformly outside the periphery of the search space.

This dissertation proposes a novel technique to estimate the effects of unknown non-homogeneous attenuation. A metric is proposed that uses drone data already collected to identify situations in which attenuation non-homogeneity is likely to result in non-negligible estimation error. This metric is then used to provide a correct location estimate despite the presence of unknown absorption using a multi-iterative

multi-stage algorithm based on ML estimation. Finally, a hybrid method is proposed that incorporates the strengths of both the adaptive and multi-stage attenuation estimation methods. This method produces highly reliable location estimates with absolutely no prior knowledge of the presence or absence of non-homogeneous attenuation within the search space.

The remainder of this dissertation is organized into the following chapters. Chapter 2 describes a method of estimating the location of a nuclear source in a three dimensional environment using static sensor locations. Chapter 3 explores the idea of adaptively moving a small number of physical sensors to new locations, based on a coarse initial estimate of the source location. A metric for detecting whether non-homogeneous attenuation is present or absent is proposed in Chapter 4. In Chapter 5, this metric is used along with a multi-stage algorithm that employs ML estimation in each stage to estimate an unknown source location in the presence of non-homogeneous attenuation. Additionally, a hybrid method is also presented in this chapter which combines the adaptive sensor-moving method and the attenuation estimation algorithm in order to locate an unknown nuclear source irrespective of the presence or absence non-homogeneous attenuation. Finally, conclusions and future work are presented in Chapter 6.

Chapter 2

Estimating the Location of a Nuclear Source in a Three-Dimensional Environment

In this chapter, it is assumed that a single isotropic point source emits detectable radiation particles. Radiation detectors are assumed to be deployed at known fixed locations around the periphery of the search space encompassing the source. The detectors are assumed to be omnidirectional; that is, the detector (built out of multiple individual sensors attached to a single drone) counts the number of interactions received from all directions. Each detector records the number of interactions over a given amount of time. The number of counts (or interactions) reported by each sensor depends on the collection interval, the drone's position and distance from the source, and the source intensity. The effect of a radioactive source on a detector is modeled as a Poisson process. In this chapter, a variety of different sensor configurations for estimating the location of the unknown source are compared. The performance results provide insight into the relationship between sensor placement and the unknown

location of the source.

2.1 System Model

The count arrival rate at the ith drone is denoted by λ_i for $i = 1, \ldots, N$, where N is the number of drones being deployed. The count arrival rate is related to the distance D_i between source and the drone via

$$\lambda_i = \lambda_B + \frac{\mu A I}{4\pi D_i^2} e^{-\rho_g D_i} \tag{2.1}$$

where μ is the photo-peak efficiency (including the branching factors for the photo peaks of the different isotopes), A is the cross-sectional area of the detector, ρ_g is the mass attenuation coefficient of air, and I is the source intensity. The exponential term in (2.1) incorporates the effects of homogeneous attenuation, and the quantity λ_B is due to background radiation. As in [7], the background radiation is assumed to be constant for the photo-peak region considered.

Note that in (2.1) the quantities μ, A, and I only appear together as a product. Defining $V = \ln(\mu A I)$, it follows that

$$\lambda_i = \lambda_B + \frac{1}{4\pi D_i^2} e^{V - \rho_g D_i} \tag{2.2}$$

Because the units of I are Becquerels and μA is dimensionless, so it follows that the units of V are ln(Bq).

It is assumed that a set of sensors are placed at fixed locations outside the periphery of the search space. The search space is modeled as a rectangular cuboid with corners $(\pm\frac{L_1}{2}, \pm\frac{L_2}{2}, \pm\frac{L_3}{2})$ in three-dimensional space. Drones are placed α m away from the periphery of the search space.

If the position of the ith drone is denoted by (x_i, y_i, z_i), then the distance $D_i = \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2}$ where (x_0, y_0, z_0) is used to indicate the (unknown) location of the source. The observed nuclear decay is modeled as a Poisson process [14]. Each detector receives only a small portion of the radioactive particles. A well-known result of "splitting" a Poisson process is that each split stream is also a Poisson process, and furthermore the split processes are mutually independent [36]. Let T_i denote the amount of time over which the data has been collected at the ith drone. Then the number of counts received during T_i at the ith drone is Poisson $(\lambda_i T_i)$, and these counts are mutually independent for all the drones. Denote the count at the ith sensor as K_i. The probability of measuring k_i counts at the ith sensor is accordingly given by

$$P(K_i = k_i; x_0, y_0, z_0, I) = \frac{(\lambda_i T_i)^{k_i} \, e^{-(\lambda_i T_i)}}{k_i!}, \quad k_i = 0,1,\ldots \tag{2.3}$$

Furthermore, by independence the joint mass function is the product of the individual mass functions of each sensor; that is

$$P(\mathbf{K} = \mathbf{k};\ x_0, y_0, z_0, I) = \textstyle\prod_{i=1}^{N} P(K_i = k_i;\ x_0, y_0, z_0, I) \tag{2.4}$$

Consider the problem of jointly estimating (x_0, y_0, z_0) and I. The maximum likelihood (ML) estimator is given by

$$[\hat{x}_0, \hat{y}_0, \hat{z}_0, \hat{I}] = \underset{x_0, y_0, z_0, I}{\arg\max}\ P(\mathbf{K} = \mathbf{k}; x_0, y_0, z_0, I) \tag{2.5}$$

In order to obtain a more numerically tractable result, the natural logarithm of $P(\mathbf{K} = \mathbf{k}; x_0, y_0, z_0, I)$ is equivalently maximized. Taking the natural logarithm

of (5.3) gives

$$\begin{aligned} &\ln P(\mathbf{K}=\mathbf{k}; x_0, y_0, z_0, I) \\ &\quad = \textstyle\sum_{i=1}^{N} \Big(k_i \ln(\lambda_i T_i) - \lambda_i T_i - \ln(k_i!) \Big) \end{aligned} \tag{2.6}$$

Using Eqs. (2.1) and (2.2), Eq. (2.6) can be rewritten as follows:

$$\begin{aligned} &\ln P(\mathbf{K}=\mathbf{k}; x_0, y_0, z_0, I) \\ &\quad = \textstyle\sum_{i=1}^{N} \Big(k_i (V + \ln T_i + \ln(\lambda_B e^{-V} + \frac{\mu A}{4\pi D_i^2} e^{-\rho_g D_i})) \\ &\qquad - e^{V} T_i (\lambda_B e^{-V} + \frac{\mu A}{4\pi D_i^2} e^{-\rho_g D_i}) - \ln(k_i!) \Big) \end{aligned} \tag{2.7}$$

ML estimation is implemented via a numerical maximization of the joint multivariate mass function using the MATLAB routine `fminsearchbnd` [16]. The algorithm is based on the built-in MATLAB routine `fminsearch` [17] that computes the minimum value of the objective function. The routine `fminsearchbnd` enables the implementation of upper and lower constraints on the variables on which the function depends. For the specific case of the problems discussed in this thesis, these constraints are the boundaries of the search space.

Similar to [7], 0.0762 mNaI based scintillating detectors are assumed, and the detection efficiency is calculated based on the measurements for the Cs-137 photopeak region of 662-keV. The background radiation level is assumed to be uniformly distributed at an intensity of $\lambda_B = 10^3$ cps as in [7]. Because performance depends on μ, A, and I solely through $V = \ln(\mu A I)$, the individual values of μ, A, and I do not need to be specified, if performance is given as a function of V. The mass attenuation coefficient for air is $\rho_g = 0.0775$ cm^2/g and the density of air is 0.001225 g/cm^3 [24].

Numerical results are presented for a high-rise building of dimension 100 m $\times$100 m $\times$100 m (so that $L_1 = L_2 = L_3 = 100$ m). Drones are placed $\alpha = 10$ m away from the search space boundary planes as described earlier.

Because the location of the source in the building is unknown, and because performance can depend strongly on the location of the source, a set of 125 source locations are considered, uniformly distributed about the search space. These locations have co-ordinates $(20i, 20j, 20k)$ where i,j and k can take values $\pm 2, \pm 1$ and 0. The maximum and average distance between the true source location and its estimate for these locations are used to analyze the performance of the algorithm. The highest possible error among the 125 locations is referred to as the maximum error and the mean of all the errors is defined as the average error.

The estimation results are averaged over n iterations. The estimate of the source location from the jth iteration is denoted by (x_j, y_j, z_j). The average distance of this estimate from the true location of the source (x, y, z) is calculated as the error in location estimate, as follows:

$$\bar{D} = \frac{1}{n} \sum_{j=1}^{n} \hat{D}_j \tag{2.8}$$

where $\hat{D}_j = \sqrt{(\hat{x}_j - x_0)^2 + (\hat{y}_j - y_0)^2 + (\hat{z}_j - z_0)^2}$. Throughout this dissertation, results have been averaged over $n = 500$ iterations.

2.2 Performance results

2.2.1 Six faces accessible

This first scenario is primarily hypothetical. It is assumed that the drones have access to all six sides of the building, including underneath the building. Even though this configuration is unlikely to be able to be implemented in practice, it provides important insight into the behavior of the ML algorithm, especially as a function of the number of sensors.

Although sensors can be placed in an infinite number of locations, minimal flexibility is lost by restricting locations to be near the centers of the sides (faces), near corners, and near the center of edges. In all, these choices result in 26 locations which are presented in Fig. 2.1. The face locations are $(\pm(\frac{L_1}{2}+\alpha),\ 0,\ 0), (0, \pm(\frac{L_2}{2}+\alpha),\ 0)$ and $(0,\ 0, \pm(\frac{L_3}{2}+\alpha))$, and the locations of the corners are $(\pm(\frac{L_1}{2}+\alpha), \pm(\frac{L_2}{2}+\alpha), (\pm\frac{L_3}{2}+\alpha))$. The edges are located at $(\pm(\frac{L_1}{2}+\alpha), \pm(\frac{L_2}{2}+\alpha), 0), (\pm(\frac{L_1}{2}+\alpha), 0, \pm(\frac{L_3}{2}+\alpha))$ and $(0, \pm(\frac{L_2}{2}+\alpha), (\pm\frac{L_3}{2}+\alpha))$. From these 26 possible locations, a number of subsets are considered: the faces alone, the corners alone, the edges alone, the faces with corners, the faces with edges, the corners with edges, and all locations (corners, faces and edges). The number of sensors for each of these subsets is listed in Table 2.1.

Classification	N (number of sensors)
Faces	6
Corners	8
Edges	12
Faces and Corners	14
Faces and Edges	18
Corners and Edges	20
Corners, Faces and Edges	26

Table 2.1: Subsets of 26 sensor locations.

In Figs. 2.2 and 2.3, the maximum and average errors for the 125 locations are plotted for all the subsets listed in Table 2.1 as a function of data collection time. The source strength, represented by V, is kept constant at 16.5 ln(Bq). Similarly, Figs. 2.4 and 2.5 present performance as a function of source strength (V) for a fixed data collection time of 210 seconds.

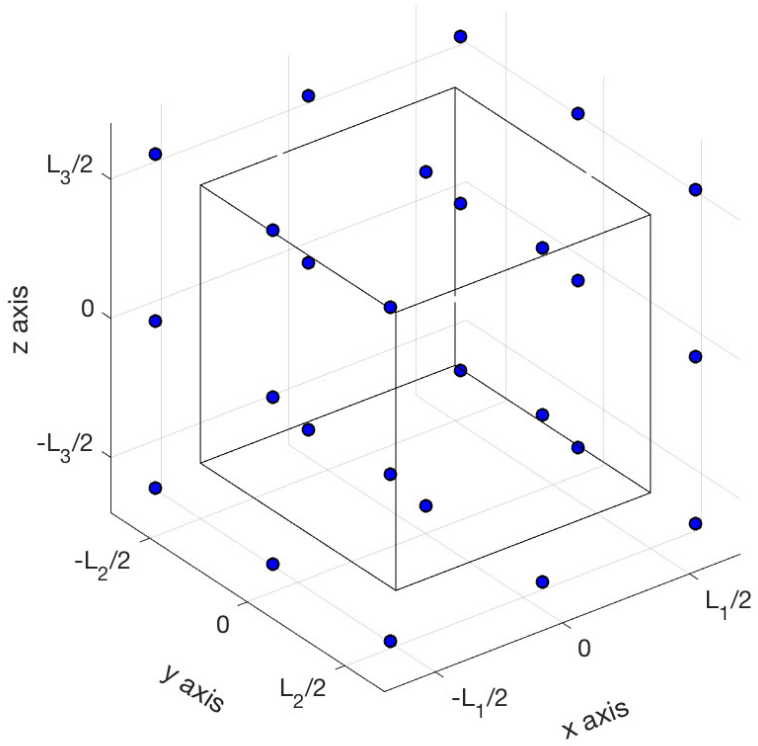

Figure 2.1: Sensor locations for $N = 26$ sensors.

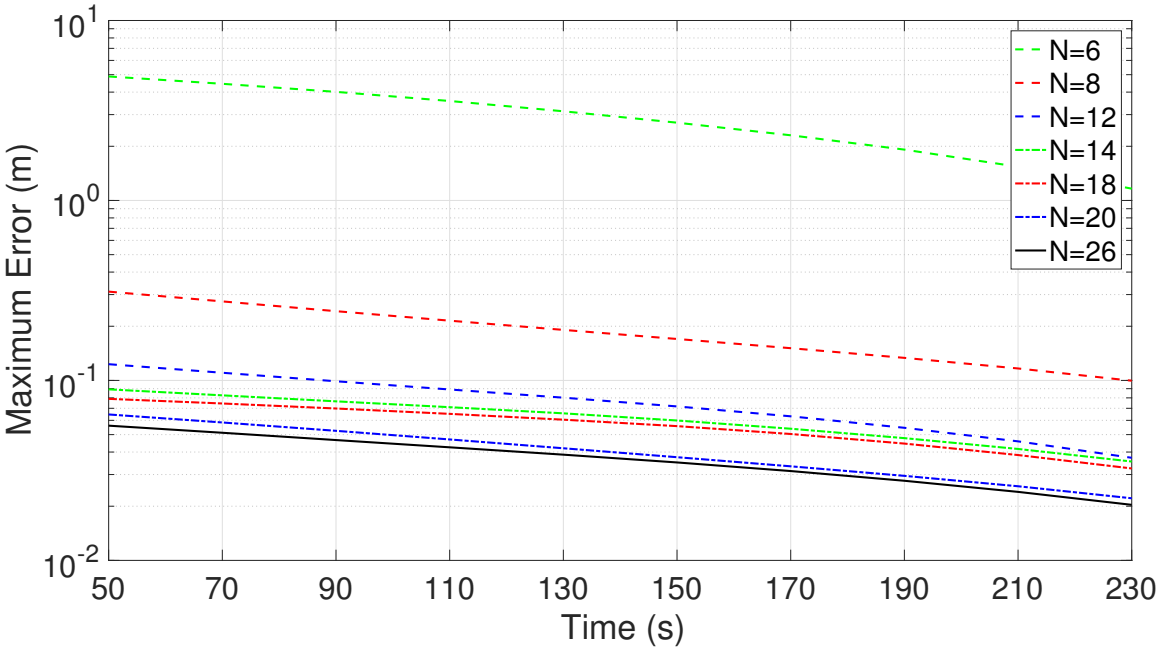

Figure 2.2: Maximum error vs. time, N fixed sensor locations, six faces accessible.

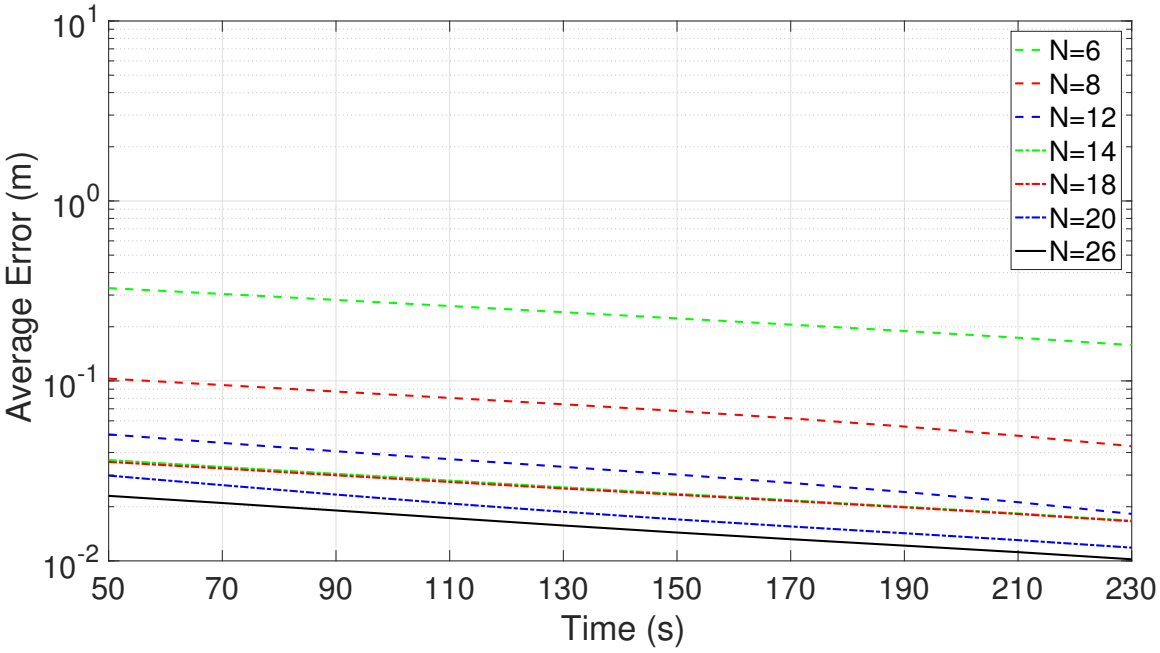

Figure 2.3: Average error vs. time, N fixed sensor locations, six faces accessible.

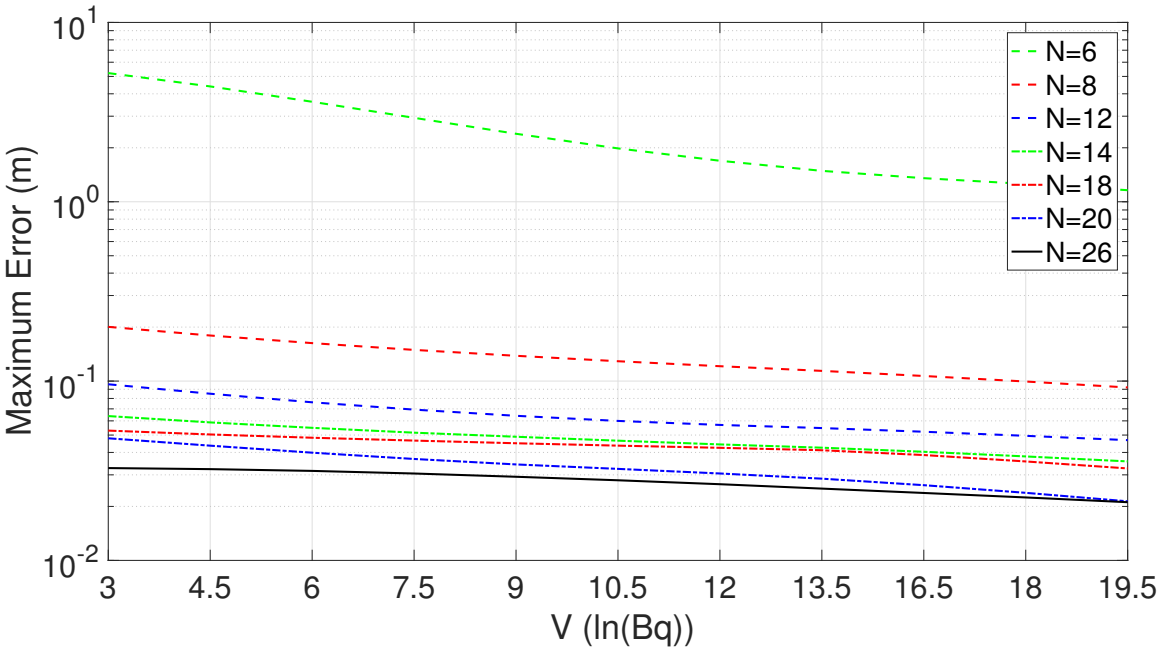

Figure 2.4: Maximum error vs. source strength, N fixed sensor locations, six faces accessible.

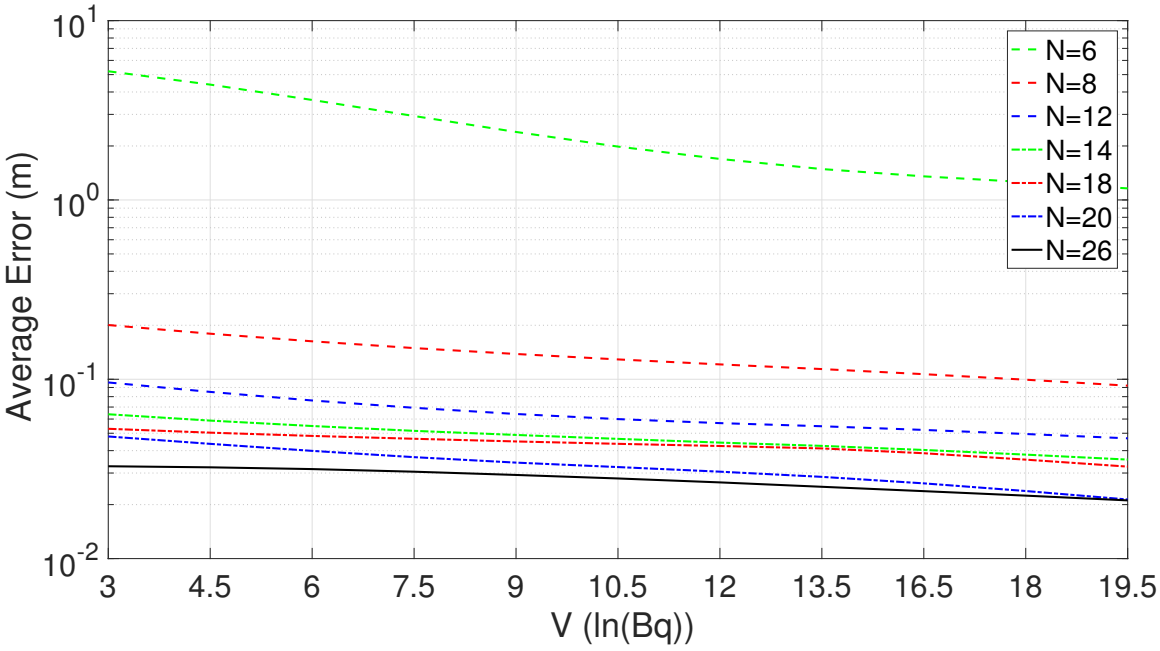

Figure 2.5: Average error vs.. source strength, N fixed sensor locations, six faces accessible.

It is evident from these results that a higher number of drone locations results in better localization of the source. In fact, as can be seen in Fig. 2.2, errors less than 0.1 m can be achieved at all source locations with $N = 26$ sensors using less than 50 seconds of data collection time. On average, error performance can be as low as a hundredth of a meter. From Figs. 2.4 and 2.5, it is seen that localization errors remain low even for very weak sources provided that a sufficiently large number of drones are used.

2.2.2 Five faces accessible

It is now more realistically assumed that sensors cannot be placed underneath the building. Additionally, the sensors along the bottom edge of the building are now moved $\alpha = 10$ m above the ground. As a result, this scenario loses the inherent symmetry of the sensor configuration in Section 2.2.1. The available face locations are $(\pm(\frac{L_1}{2}+\alpha), 0, 0)$, $(0, \pm(\frac{L_2}{2}+\alpha), 0)$ and $(0, 0, (\frac{L_3}{2}+\alpha))$. The corner locations are $(\pm(\frac{L_1}{2}+\alpha), \pm(\frac{L_2}{2}+\alpha), (\pm\frac{L_3}{2}+\alpha))$ and the edge locations are at $(\pm(\frac{L_1}{2}+\alpha), \pm(\frac{L_2}{2}+\alpha), 0)$, $(\pm(\frac{L_1}{2}+\alpha), 0, (\pm\frac{L_3}{2}+\alpha))$ and $(0, \pm(\frac{L_2}{2}+\alpha), (\pm\frac{L_3}{2}+\alpha))$. These locations are illustrated in Fig. 2.6. Similar to Section 2.2.1, subsets of these 25 sensor locations are considered. These subsets are listed in Table 2.2.

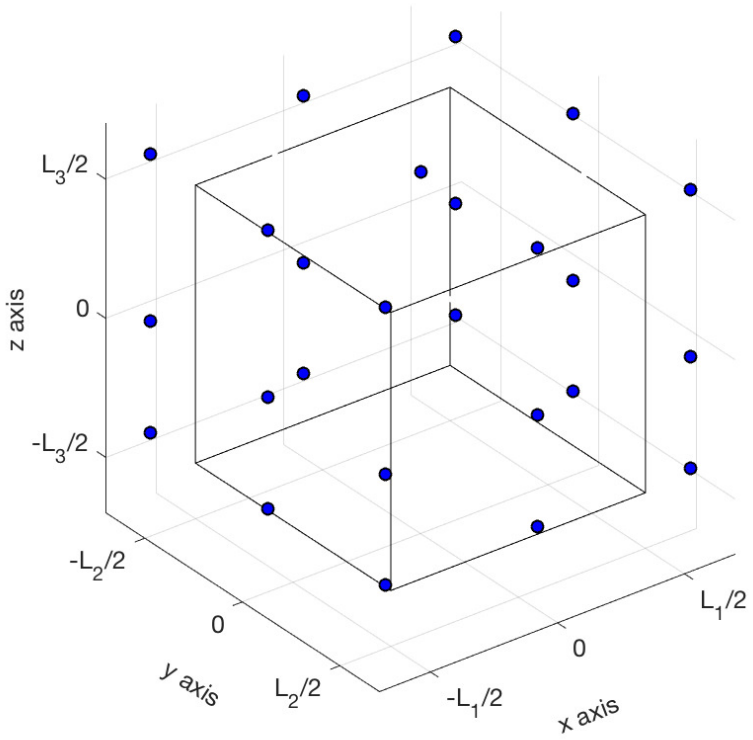

Figure 2.6: Sensor locations for $N = 25$ locations.

Classification	N (number of sensors)
Faces	5
Corners	8
Edges	12
Faces and Corners	13
Faces and Edges	17
Corners and Edges	20
Corners, Faces and Edges	25

Table 2.2: Subsets of 25 sensor locations.

The maximum and average errors for this sensor configuration are plotted against data collection time in Figs. 2.7 and 2.8. The source strength is again constant at $V = 16.5$ ln(Bq). Similarly, in Figs. 2.9 and 2.10, maximum and average errors are plotted against increasing source strength V with a fixed data collection time of 210 seconds.

It is evident from Figs. 2.7 to 2.10 that a higher number of sensor locations still result in better error performance. In fact, comparison of Fig. 2.7 with Fig. 2.2 and comparison of Fig. 2.8 with Fig. 2.3 shows that there is no significant loss of performance in reducing the number of sensor locations from $N = 26$ to $N = 25$, despite the loss of symmetry. These results show that sufficiently large numbers of sensors can locate the unknown source in spite of the asymmetry of the sensor locations.

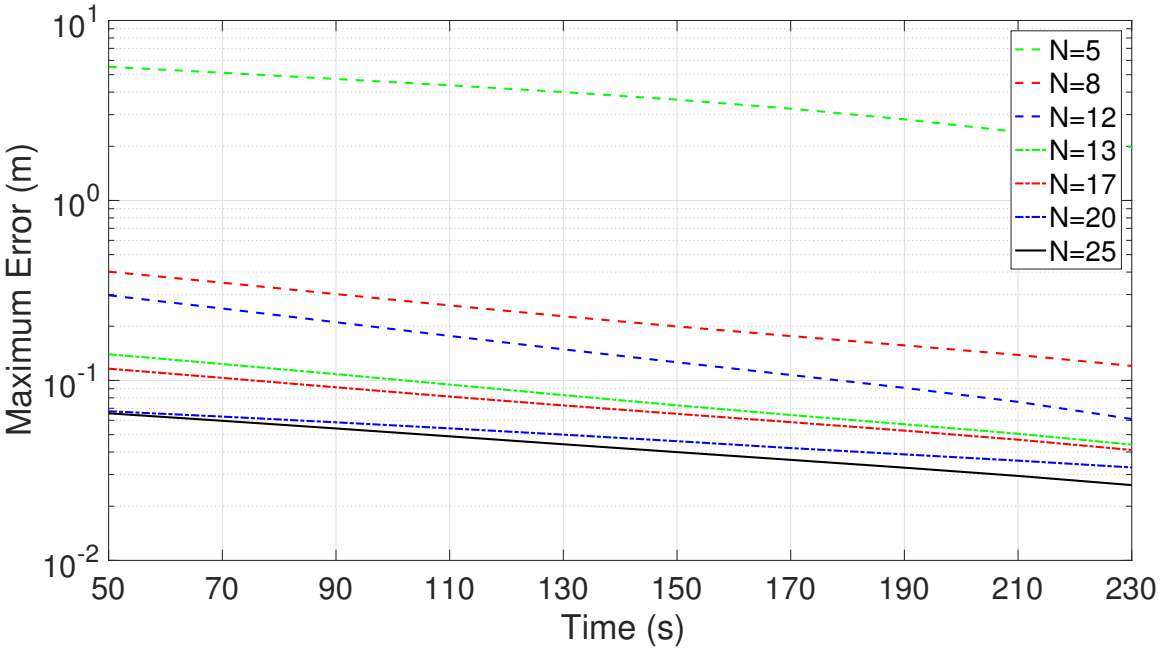

Figure 2.7: Maximum error vs. time, N fixed sensor locations, five faces accessible.

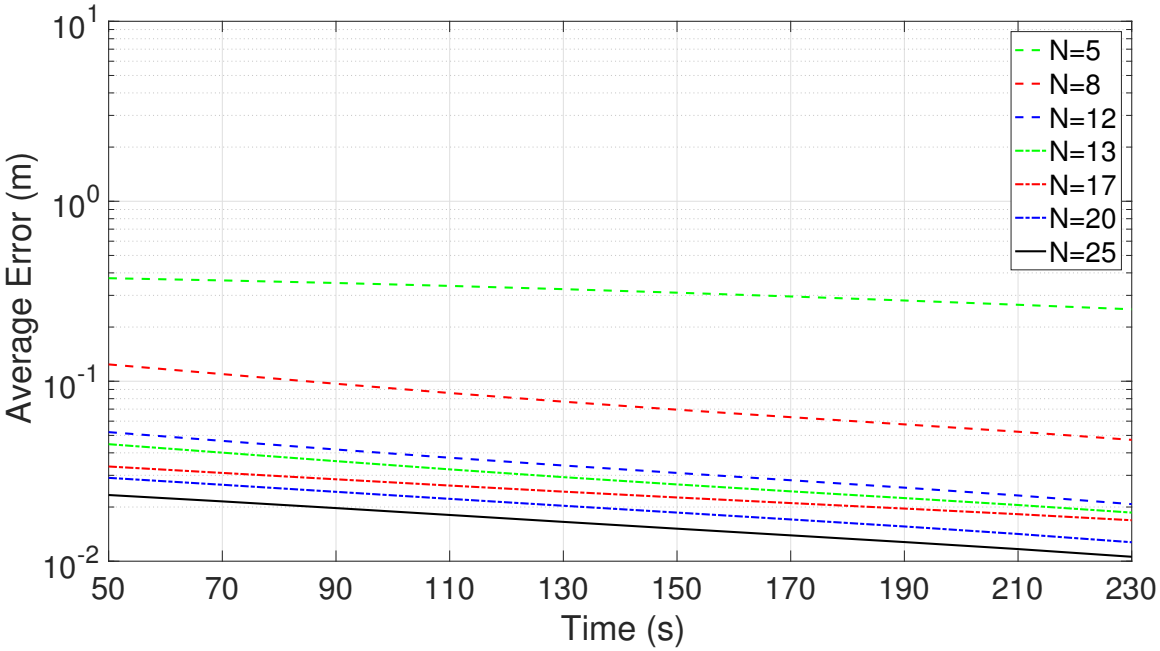

Figure 2.8: Average error vs. time, N fixed sensor locations, five faces accessible.

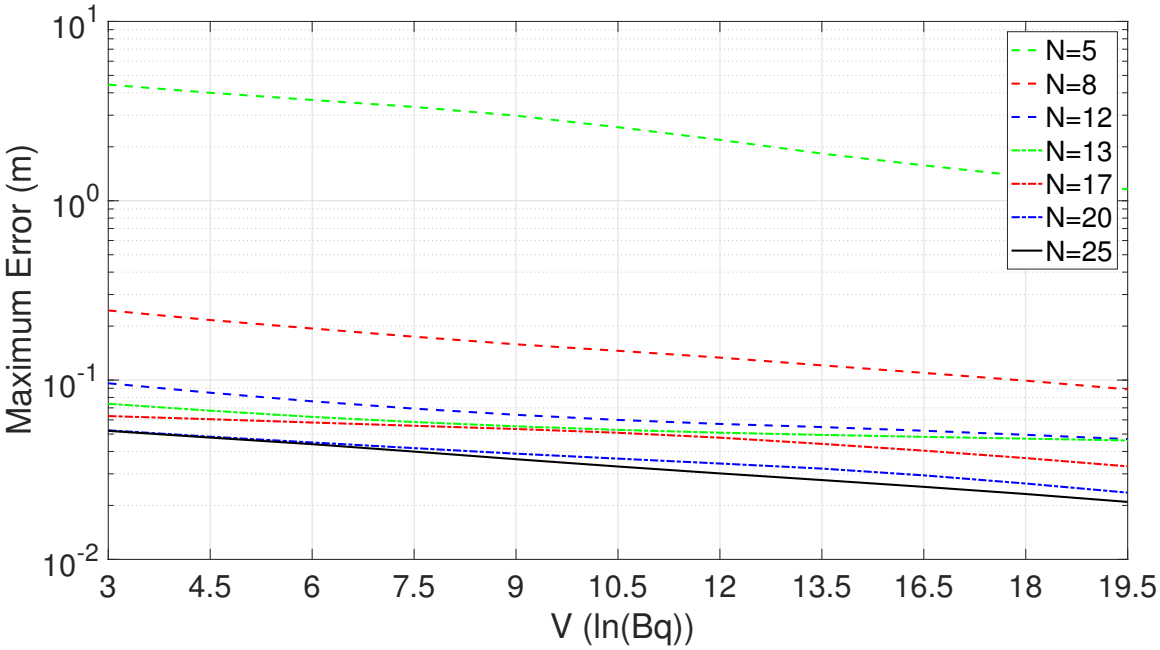

Figure 2.9: Maximum error vs. source strength, N fixed sensor locations, five faces accessible.

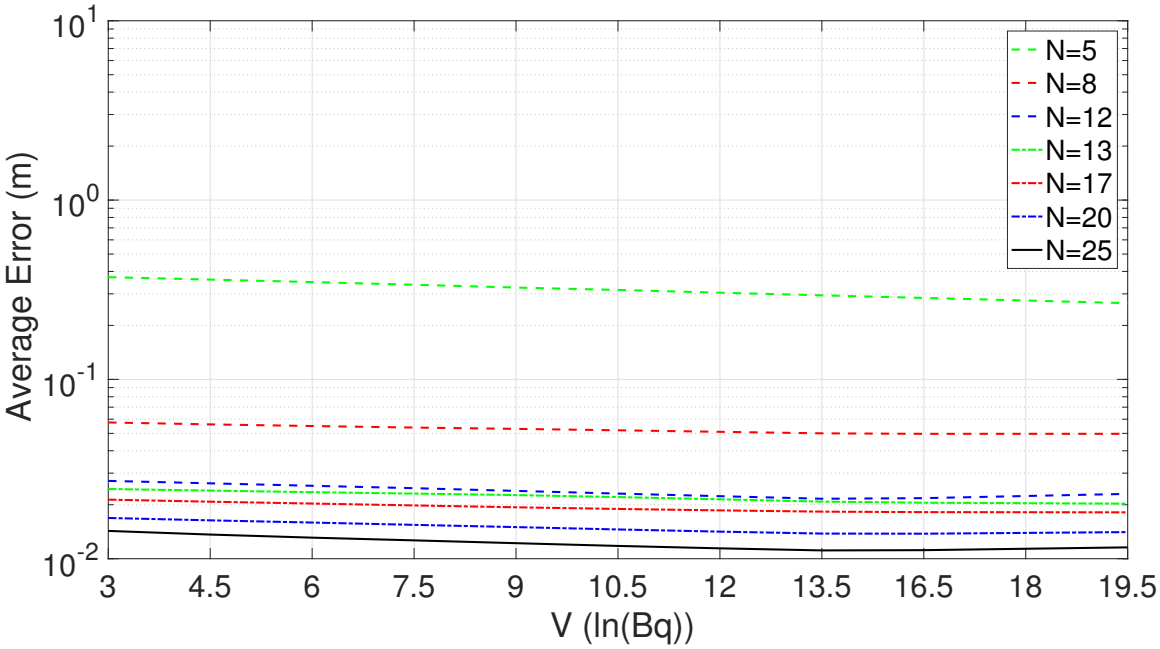

Figure 2.10: Average error vs. source strength, N fixed sensor locations, five faces accessible.

2.2.3 Three faces accessible (opposite)

In this scenario, it is assumed that two opposite faces of the building are not available to the sensors, so only the top face and the remaining opposite faces are accessible. Here there are three face locations, eight corner locations, and ten edge locations. The face locations are given by $(0, \pm(\frac{L_2}{2} + \alpha), 0)$ and $(0, 0, (\frac{L_3}{2} + \alpha))$, the corner locations are at $(\pm(\frac{L_1}{2} + \alpha), \pm(\frac{L_2}{2} + \alpha), (\pm\frac{L_3}{2} + \alpha))$ and the edge locations at $(0, \pm(\frac{L_2}{2} + \alpha), (\frac{L_3}{2} + \alpha))$, $(\pm(\frac{L_1}{2} + \alpha), 0, (\frac{L_3}{2} + \alpha))$, $(\pm(\frac{L_1}{2} + \alpha), 0, (-\frac{L_3}{2} + \alpha))$ and $(\pm(\frac{L_1}{2} + \alpha), \pm(\frac{L_2}{2} + \alpha), 0)$. This results in a total of 21 possible sensor locations which are shown in Fig. 2.11, and the corresponding subsets are listed in Table 2.3. As before, error performance is compared for all of these subsets.

Classification	N (number of sensors)
Faces	3
Corners	8
Edges	10
Faces and Corners	11
Faces and Edges	13
Corners and Edges	18
Corners, Faces and Edges	21

Table 2.3: Subsets of 21 sensor locations.

The maximum and average errors for the 125 possible source locations are again simulated and the results are presented in Figs. 2.12 to 2.15. Results for faces only ($N = 3$) are not shown because performance is quite poor, with errors consistently 30 m or more. This poor performance can be explained by noting that there are fewer sensors than there are unknown values to estimate (x_0,y_0,z_0 and I). Overall, the algorithm exhibits low localization errors with $N \geq 8$ sensor locations, and performance improves with increasing N.

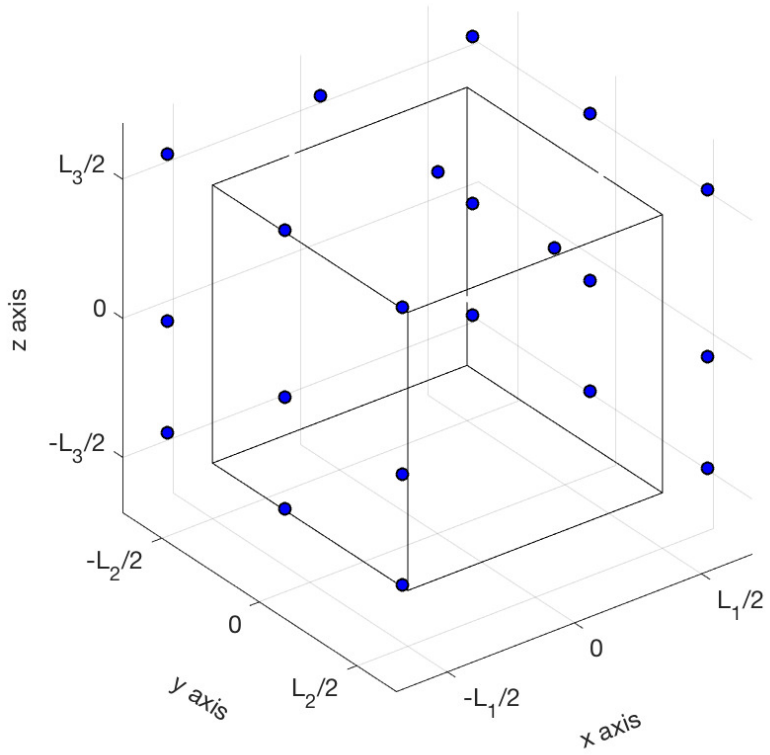

Figure 2.11: Sensor locations for $N = 21$ locations.

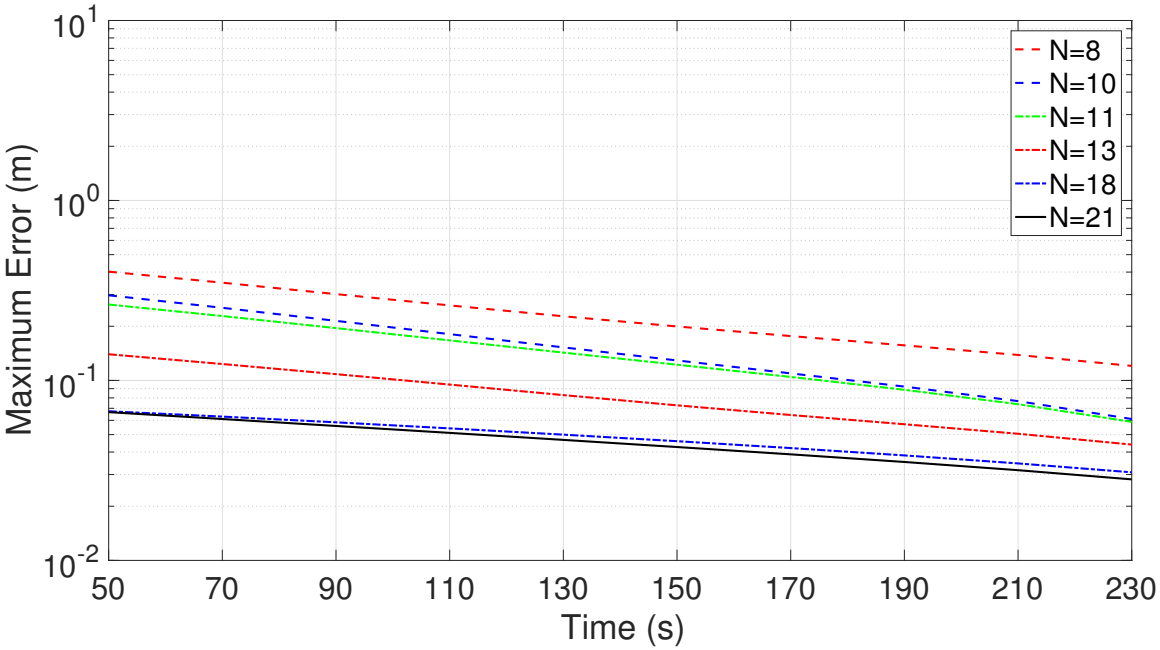

Figure 2.12: Maximum error vs. time, N fixed sensor locations, three faces accessible (opposite).

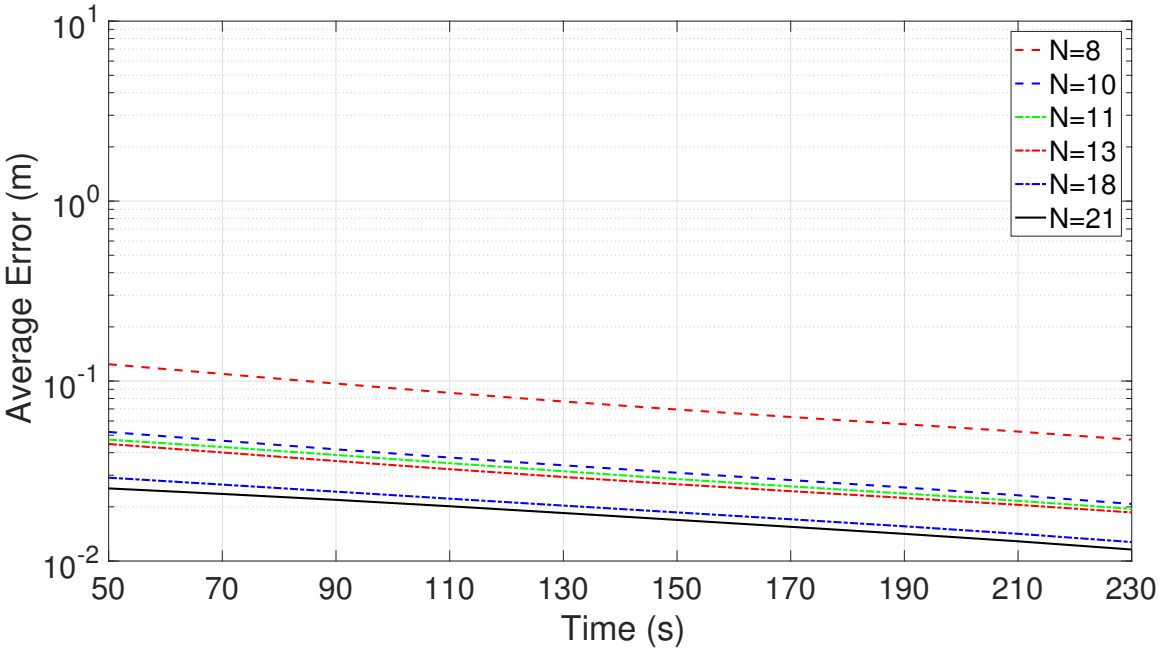

Figure 2.13: Average error vs. time, N fixed sensor locations, three faces accessible (opposite).

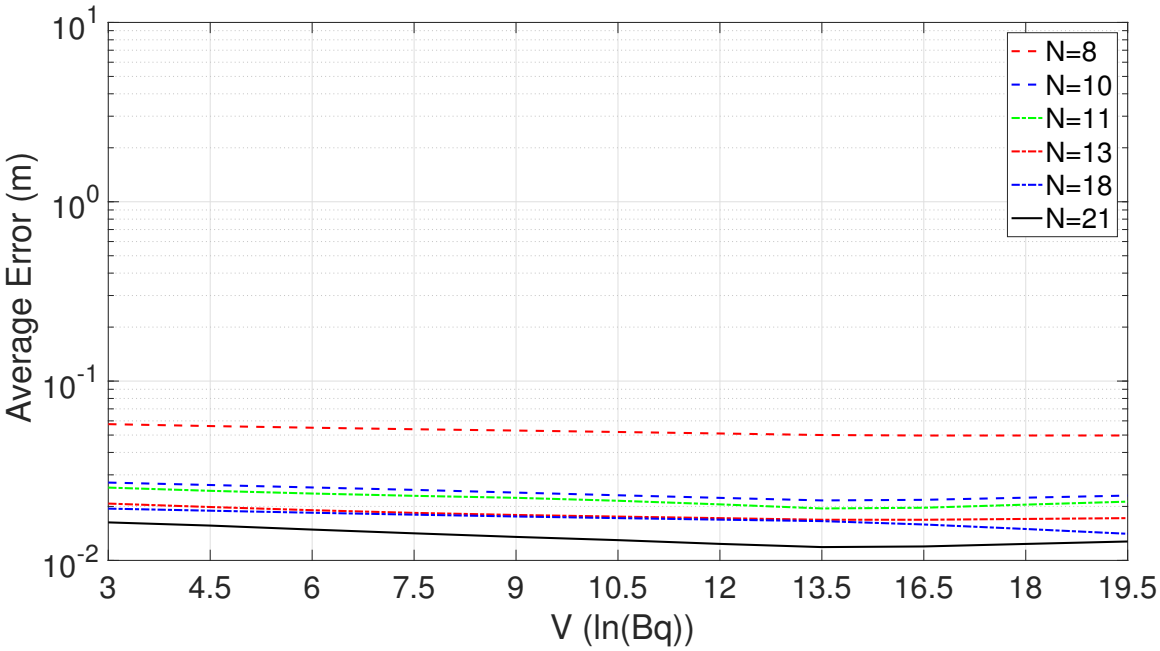

Figure 2.14: Maximum error vs. source strength, N fixed sensor locations, three faces accessible (opposite).

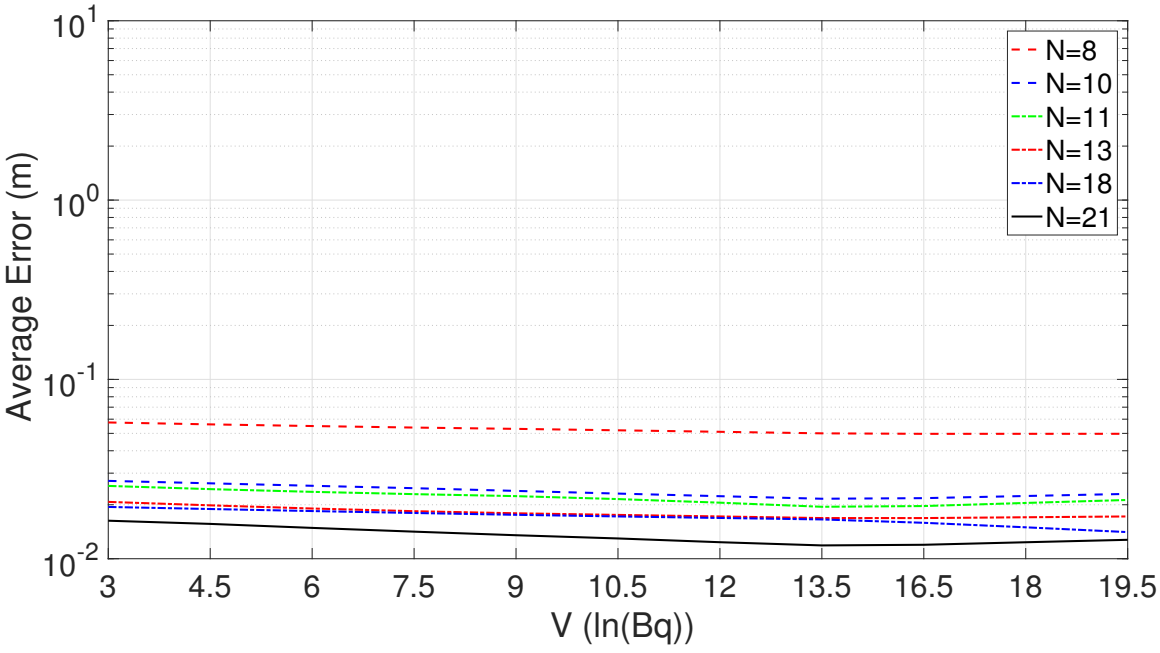

Figure 2.15: Average error vs. source strength, N fixed sensor locations, three faces accessible (opposite).

2.2.4 Three faces accessible (adjacent)

In contrast to Section 2.2.3, it is assumed that only two adjacent faces (in addition to the top face) are available to the sensors. Here there are three face locations, seven corner locations, and nine edge locations. The face locations are given by $((\frac{L_1}{2}+\alpha), 0, 0)$, $(0, (\frac{L_2}{2}+\alpha), 0)$ and $(0, 0, (\frac{L_3}{2}+\alpha))$, the corner locations are at $(\pm(\frac{L_1}{2}+\alpha), \pm(\frac{L_2}{2}+\alpha), (\frac{L_3}{2}+\alpha))$, $((\frac{L_1}{2}+\alpha), \pm(\frac{L_2}{2}+\alpha), (-\frac{L_3}{2}+\alpha))$ and $(-(\frac{L_1}{2}+\alpha), (\frac{L_2}{2}+\alpha), (-\frac{L_3}{2}+\alpha))$, and the edge locations are at $(\pm(\frac{L_1}{2}+\alpha), 0, (\frac{L_3}{2}+\alpha))$,$(0, \pm(\frac{L_2}{2}+\alpha), (\frac{L_3}{2}+\alpha))$,$((\frac{L_1}{2}+\alpha), \pm(\frac{L_2}{2}+\alpha), (-\frac{L_3}{2}+\alpha))$ and $(-(\frac{L_1}{2}+\alpha), (\frac{L_2}{2}+\alpha), (-\frac{L_3}{2}+\alpha))$. This results in a total of 19 possible sensor locations which are illustrated in Fig. 2.16, and the corresponding subsets are listed in Table 2.4. The maximum and average errors of 125 possible source locations are used to analyze the performance of the algorithm for all of these subsets.

Classification	N (number of sensors)
Faces	3
Corners	7
Edges	9
Faces and Corners	10
Faces and Edges	12
Corners and Edges	16
Corners, Faces and Edges	19

Table 2.4: Subsets of 19 sensor locations.

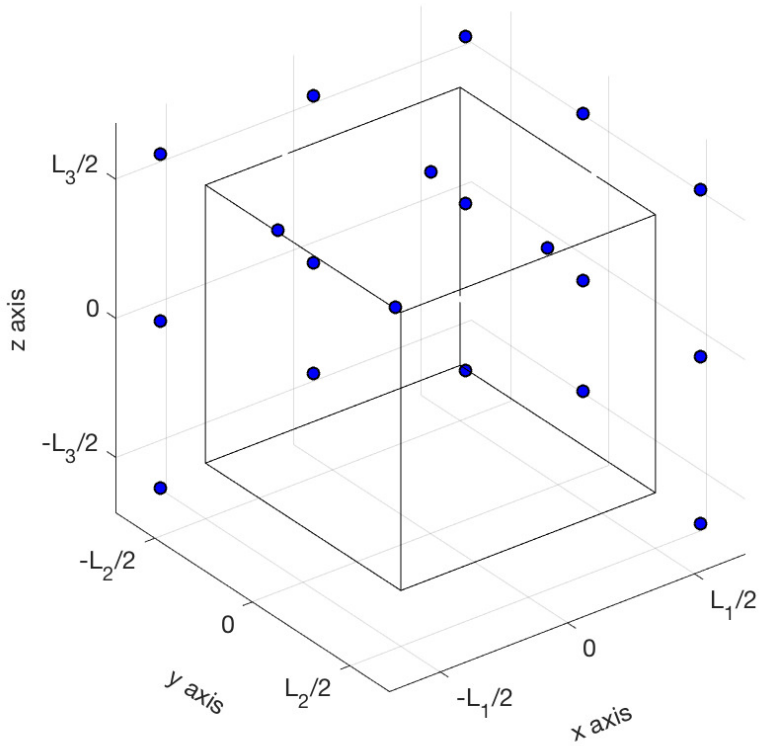

Figure 2.16: Sensor locations for $N = 19$ locations.

Results are shown in Figs. 2.17 to 2.20. Similar to Section 2.2.3, the subset containing only faces (i.e. $N = 3$) exhibits poor error performance, and is omitted from the figures. Performance with $N = 3$ is slightly better here than with two opposite faces available, around 20 m in the best case (instead of 30 m). For $N = 7$ and larger, performance is far better, but in general two opposite faces available is slightly better than two adjacent faces being available because more sensor locations are utilized (a maximum number of $N = 21$ for two opposite faces vs $N = 19$ for two adjacent faces).

2.2.5 Two faces accessible

In this final scenario, it is assumed that the sensors have access to only two faces of the building, the top and a single side. Accordingly, the only two face locations available are $((\frac{L_1}{2}+\alpha), 0, 0)$ and $(0, 0, (\frac{L_3}{2}+\alpha))$. There are six corner locations available at co-ordinates $(\pm(\frac{L_1}{2}+\alpha), \pm(\frac{L_2}{2}+\alpha), (\frac{L_3}{2}+\alpha))$ and $((\frac{L_1}{2}+\alpha), \pm(\frac{L_2}{2}+\alpha), (-\frac{L_3}{2}+\alpha))$ and seven edge locations available at co-ordinates $(0, \pm(\frac{L_2}{2}+\alpha), (\frac{L_3}{2}+\alpha))$,$(\pm(\frac{L_1}{2}+\alpha), 0, (\frac{L_3}{2}+\alpha))$,$((\frac{L_1}{2}+\alpha), \pm(\frac{L_2}{2}+\alpha), 0)$ and $((\frac{L_1}{2}+\alpha), 0, (-\frac{L_3}{2}+\alpha))$. This leads to a total of 15 possible sensor locations which are illustrated in Fig. 2.21 and whose subsets are listed in Table 2.5.

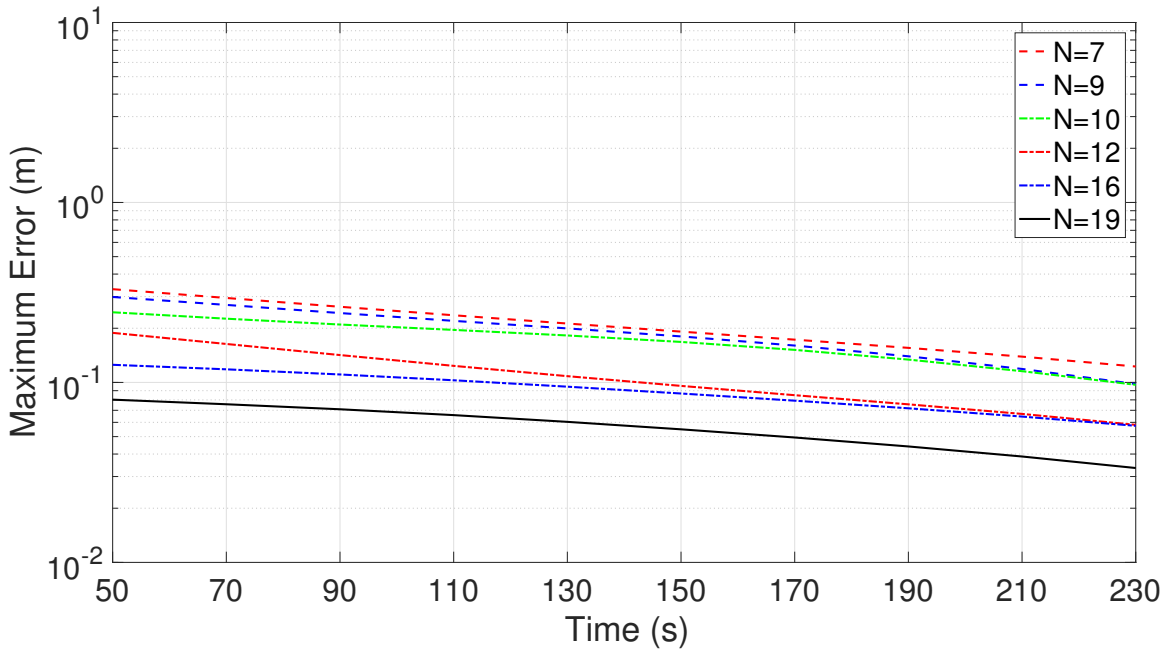

Figure 2.17: Maximum error vs. time, N fixed sensor locations, three faces accessible (adjacent).

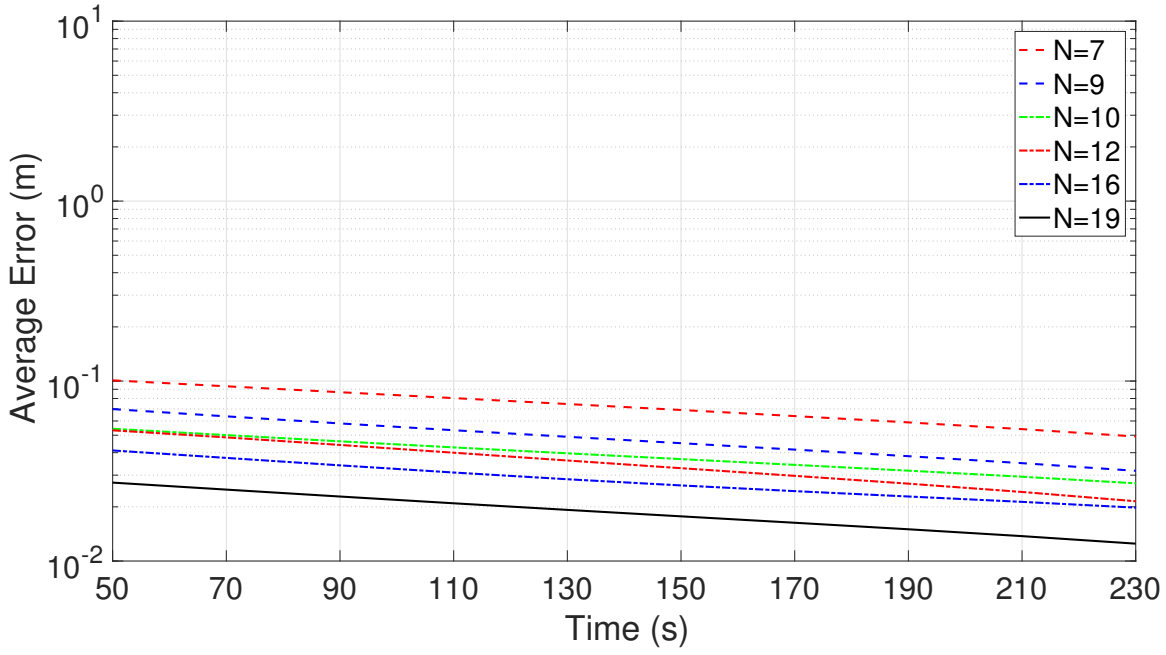

Figure 2.18: Average error vs. time, N fixed sensor locations, three faces accessible (adjacent).

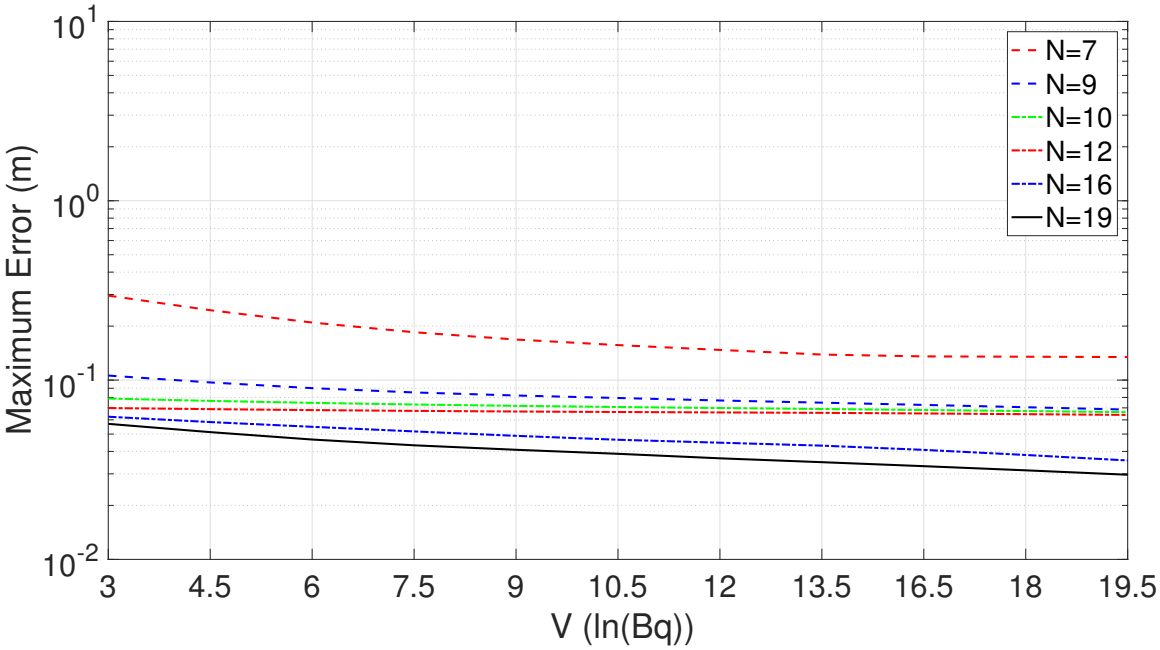

Figure 2.19: Maximum error vs. source strength, N fixed sensor locations, three faces accessible (adjacent).

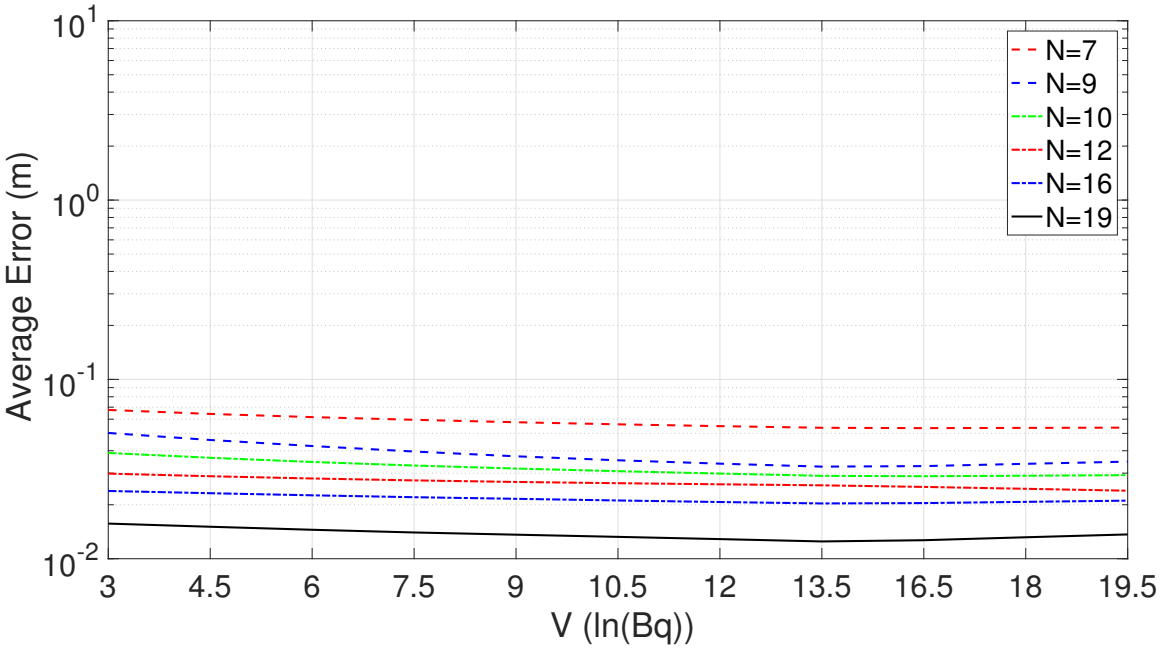

Figure 2.20: Average error vs. source strength, N fixed sensor locations, three faces accessible (adjacent).

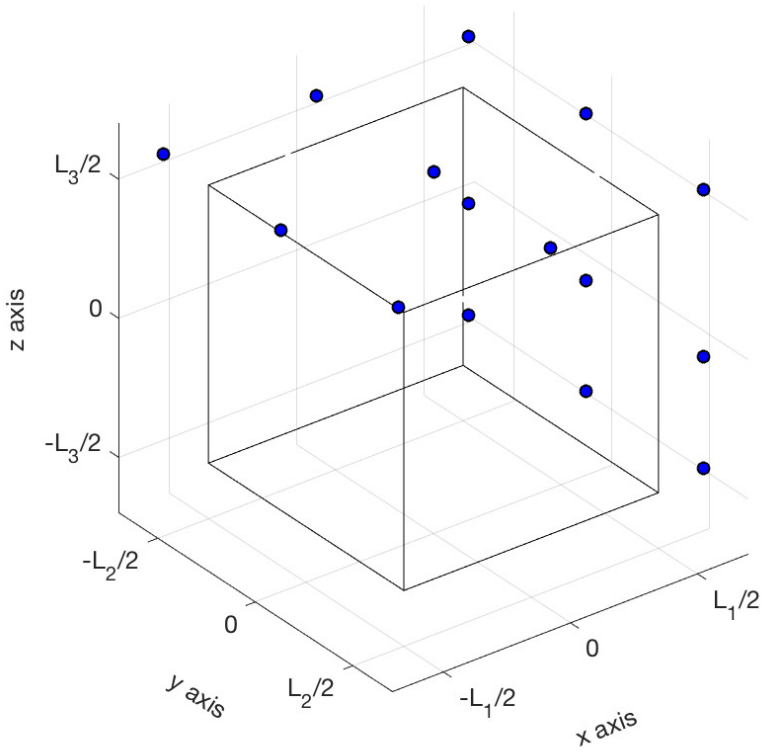

Figure 2.21: Sensor locations for $N = 15$ locations.

Classification	N (number of sensors)
Faces	2
Corners	6
Edges	7
Faces and Corners	8
Faces and Edges	9
Corners and Edges	13
Corners, Faces and Edges	15

Table 2.5: Subsets of 15 sensor locations.

Figs. 2.22 to 2.25 present performance results. Again, due to poor performance, results for faces only ($N = 2$) are omitted; in this case, the errors are again 30 m or more. For larger values of N, performance is far better, again improving as N increases.

To evaluate the effects of sensor placement on performance with N fixed sensors, Figs. 2.26 and 2.27 present performance with five faces accessible, three faces accessible (opposite faces), and two faces accessible with $N = 13$. The results show that the best performance occurs with five faces accessible; that is better performance occurs when sensors can be as spread out as possible.

2.3 Conclusion

In conclusion, ML estimation has been presented as an effective means of detecting the location and strength of a nuclear source in a three-dimensional environment. The estimation scheme incorporates the effects of attenuation due to air.

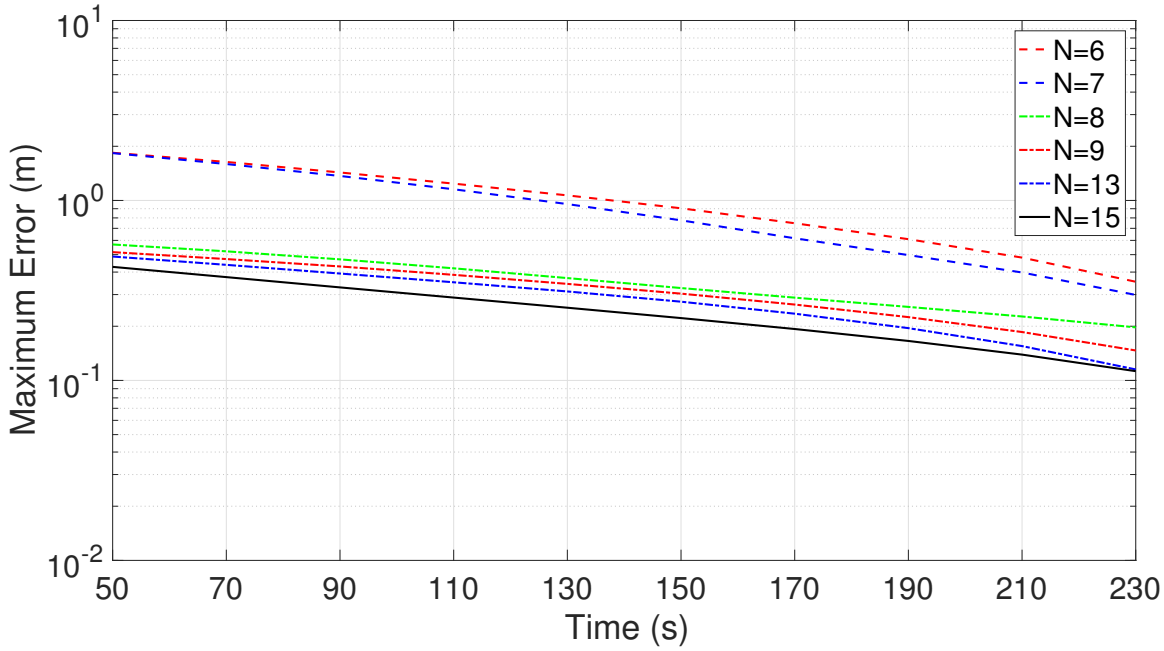

Figure 2.22: Maximum error vs. time, N fixed sensor locations, two faces accessible.

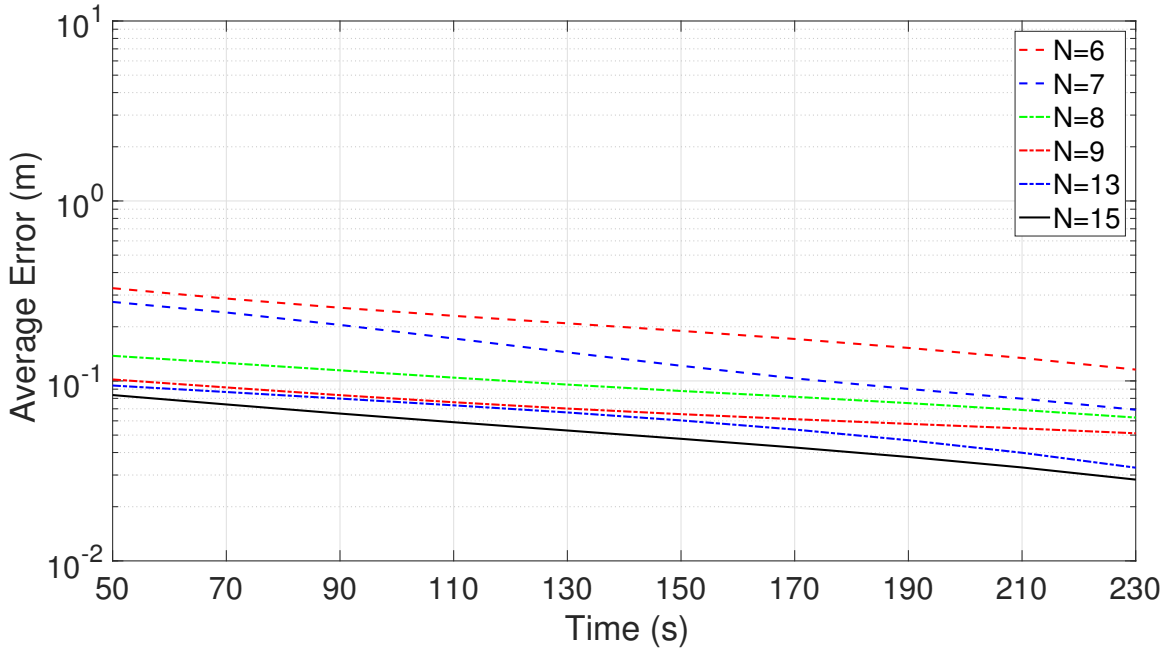

Figure 2.23: Average error vs. time, N fixed sensor locations, two faces accessible.

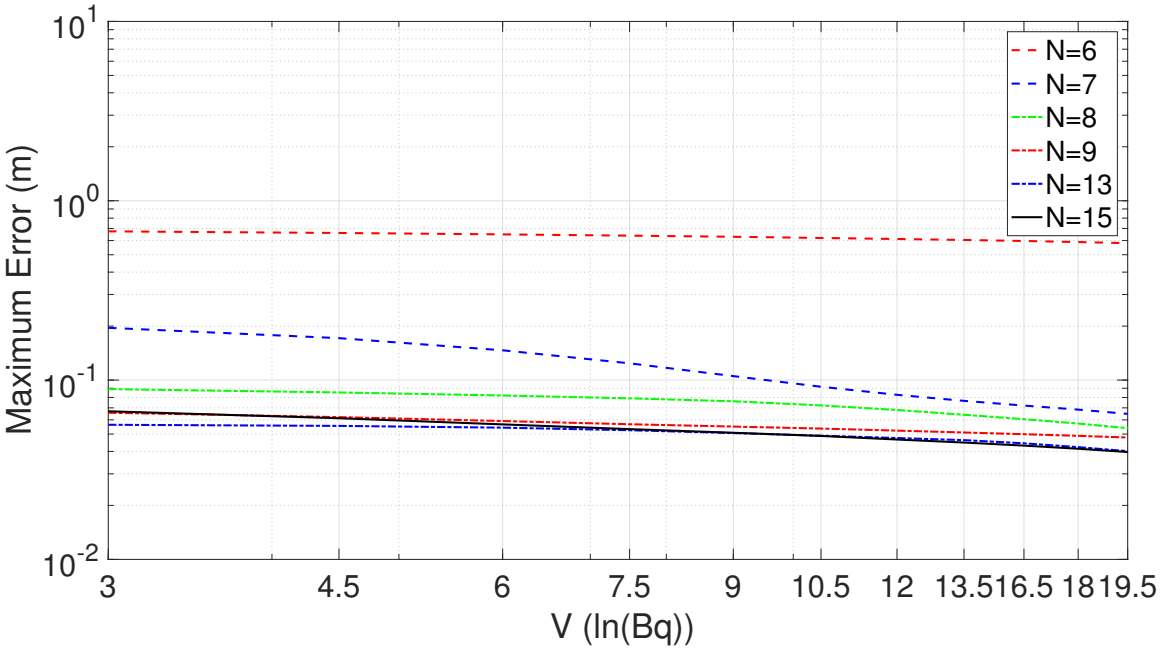

Figure 2.24: Maximum error vs. source strength, N fixed sensor locations, two faces accessible.

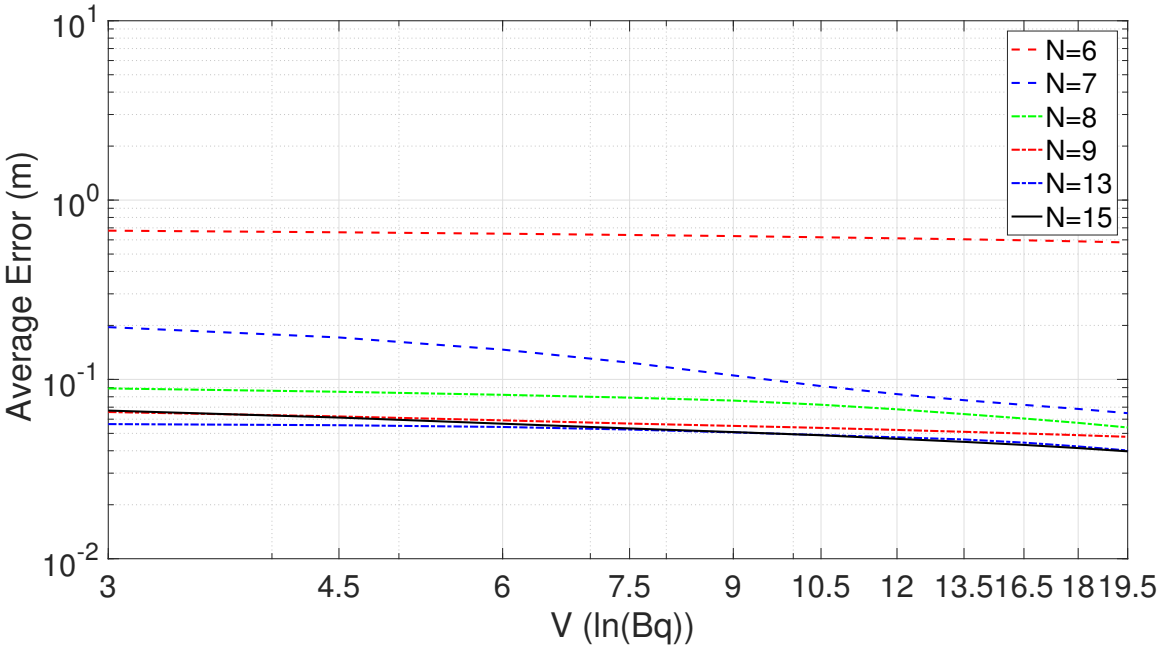

Figure 2.25: Average error vs. source strength, N fixed sensor locations, two faces accessible.

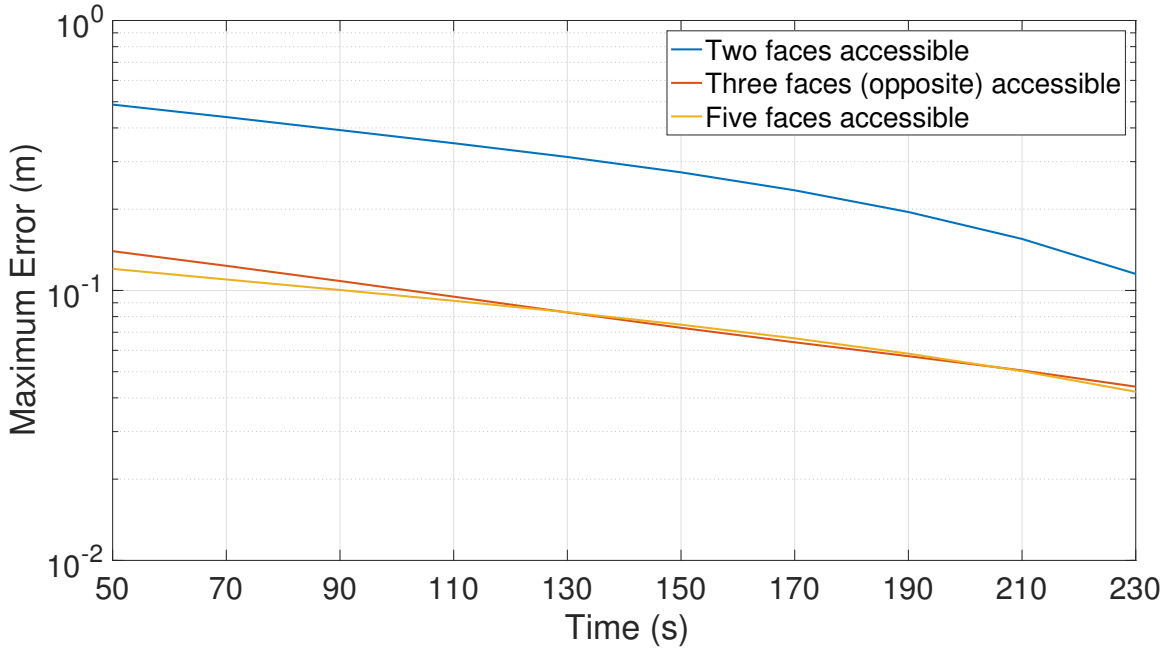

Figure 2.26: Maximum error vs. time, 13 fixed sensor locations.

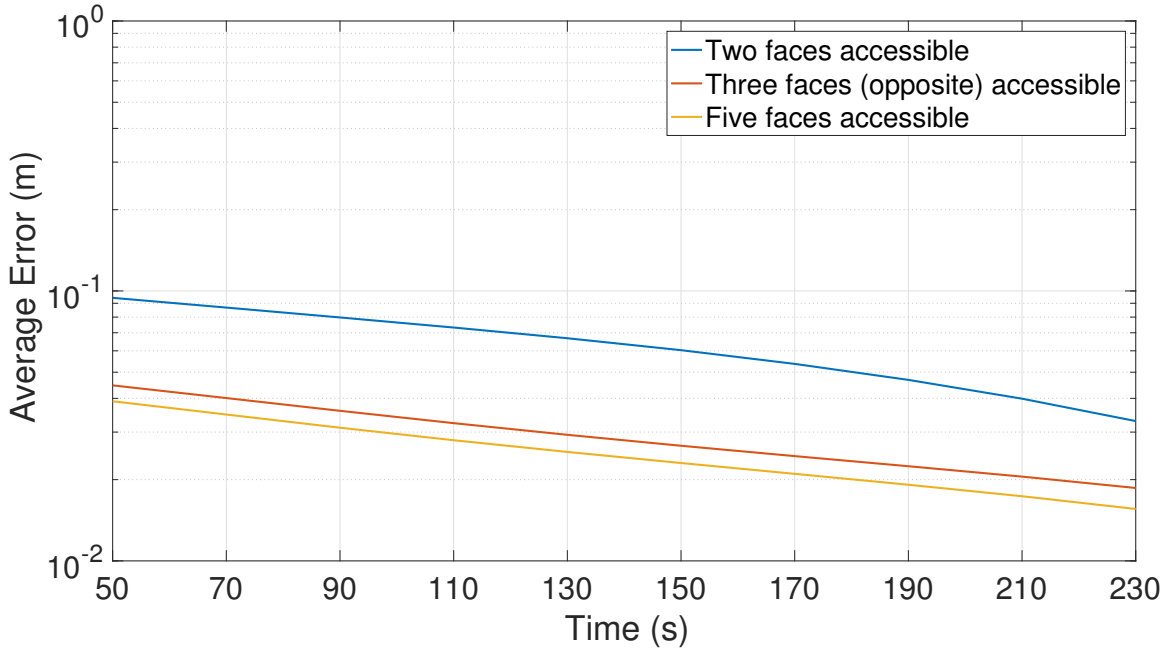

Figure 2.27: Average error vs. time, 13 fixed sensor locations.

Results show that, generally, performance improves with an increasing number of sensors. The results also show that even if various sides of the periphery of the search space are unavailable, accurate estimation is possible provided a large number of sensors are used. In addition, the large gap in performance between maximum and average error demonstrates that estimation performance depends strongly on the location of the source.

Chapter 3

Moving Sensors During the Estimation Process

The previous chapter demonstrates that the performance of a particular sensor configuration depends strongly on the source location. Consequently, it maybe possible to improve performance by moving sensors during the estimation process based on data already received. Such algorithms can be described as adaptive. In this chapter, algorithms are considered in which the sensors, after obtaining an initial estimate of source location, move to new locations to estimate the source location more accurately. The counts recorded by the drones in both stages are used collectively to obtain a final estimate of the source location.

In the remainder of this chapter, several adaptive algorithms are presented along with their performance, based on accessibility situations similar to those of Chapter 2.

3.1 System Model

The radioactive decay is modeled as a Poisson process as in Chapter 2. Therefore, the count arrival rate λ_i is a function of λ_B, V, ρ_g and D_i as in Eq. (2.2), where the subscript i denotes the location at which data is collected. At the ith sensor location, the Poisson process has parameter $\lambda_i T_i$, where T_i is the time spent collecting data at the ith location.

The MLE is obtained by maximizing

$$[\hat{x}_0, \hat{y}_0, \hat{z}_0, \hat{I}] = \underset{x_0, y_0, z_0, I}{\arg\max}\ P(\mathbf{K} = \mathbf{k}; x_0, y_0, z_0, I) \tag{3.1}$$

where $\mathbf{k}$ incorporates the counts recorded at the sensor locations in both stages.

Both stages of the algorithm are specific to the geometry of the search space and the availability of drone locations around this space. As in Chapter 2, the search space is modeled as a rectangular cuboid with corners $(\pm\frac{L_1}{2}, \pm\frac{L_2}{2}, \pm\frac{L_3}{2})$ in three-dimensional space. The cuboid rests on the ground, so drones are assumed to be unable to access the area underneath the bottom face.

3.2 Adaptive Two-Stage Algorithms

In the first stage, five sensors are placed at the centers of the accessible faces of the building, as shown in Fig. 3.1. The locations of the five sensors are $s_{[\pm1,0,0]} = (\pm(\frac{L_1}{2}+\alpha), 0, 0)$, $s_{[0,\pm1,0]} = (0, \pm(\frac{L_2}{2}+\alpha), 0)$ and $s_{[0,0,1]} = (0, 0, (\frac{L_3}{2}+\alpha))$, where α is the separation in meters between a drone and the search space. In the second stage sensors may move to additional locations close to corners and edges of the cuboid. Therefore, only a total of five sensors are used in the entire application (as opposed to 16 detectors in [7] and 100 detectors in [39]).

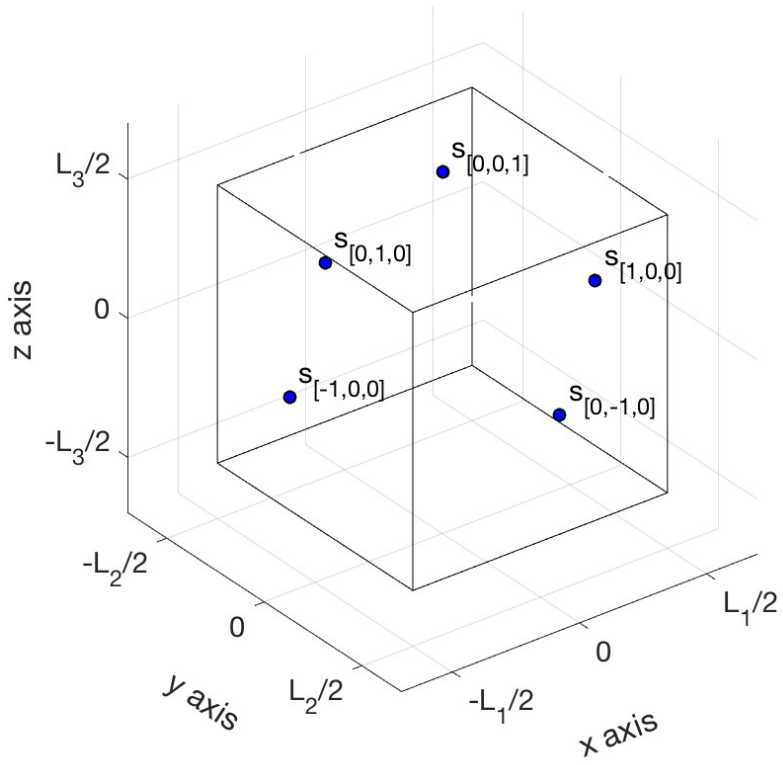

Figure 3.1: Drone locations for the first stage, five faces accessible.

The second stage locations are selected from 25 possible locations: the five faces from the first stage, 8 corner locations, and 12 edge locations, as shown in Fig. 3.2. The corner locations are $s_{[\pm1,\pm1,\pm1]} = \big(\pm(\frac{L_1}{2}+\alpha), \pm(\frac{L_2}{2}+\alpha), (\alpha \pm \frac{L_3}{2})\big)$. The vertical edge locations are $s_{[\pm1,\pm1,0]} = \big(\pm(\frac{L_1}{2}+\alpha), \pm(\frac{L_2}{2}+\alpha), 0\big)$, the top edge locations are $s_{[0,\pm1,1]} = \big(0, \pm(\frac{L_2}{2}+\alpha), (\frac{L_3}{2}+\alpha)\big)$ and $s_{[\pm1,0,1]} = \big(\pm(\frac{L_1}{2}+\alpha), 0, (\frac{L_3}{2}+\alpha)\big)$, and the bottom edge locations are $s_{[0,\pm1,-1]} = (0, \pm(\frac{L_2}{2}+\alpha), -(\frac{L_3}{2}-\alpha))$ and $s_{[\pm1,0,-1]} = \big(\pm(\frac{L_1}{2}+\alpha), 0, -(\frac{L_3}{2}-\alpha)\big)$. In the first stage, data collected by the five drones at the face locations obtain an initial estimate of the source location, and this location is identified as belonging to one of the twenty-seven sub-cuboid regions as shown in Fig. 3.3. The sub-cuboids are not uniform in size; it has been found that better performance occurs if the center sub-cuboid is smaller than the others. The boundaries of the center cuboid are $x = \pm\beta\frac{L_1}{2}$, $y = \pm\beta\frac{L_2}{2}$, and $z = \pm\beta\frac{L_3}{2}$, where β is a fixed constant between 0 and 1. These boundaries along with the outer boundaries of the search space define all the boundaries of the sub-cuboids. The drones are moved in the second stage based on the sub-cuboid identified in the first stage.

The new drone locations in the second stage are not decided based solely on proximity to the estimate of the location of the source. Instead, the algorithm moves the sensors to locations that are reasonably close but spatially separated from one another. As mentioned in [7], spatial diversity of sensor locations benefits the localization problem. The heart of the adaptive algorithm in this chapter is the choice of where to send the drones in the second stage.

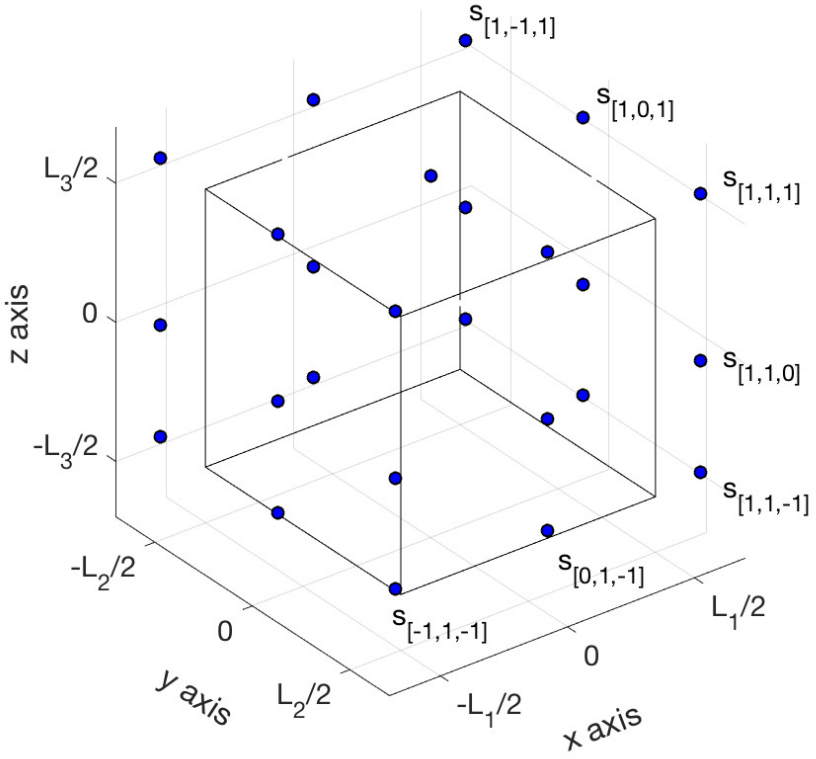

Figure 3.2: Possible drone locations for the second stage, five faces accessible.

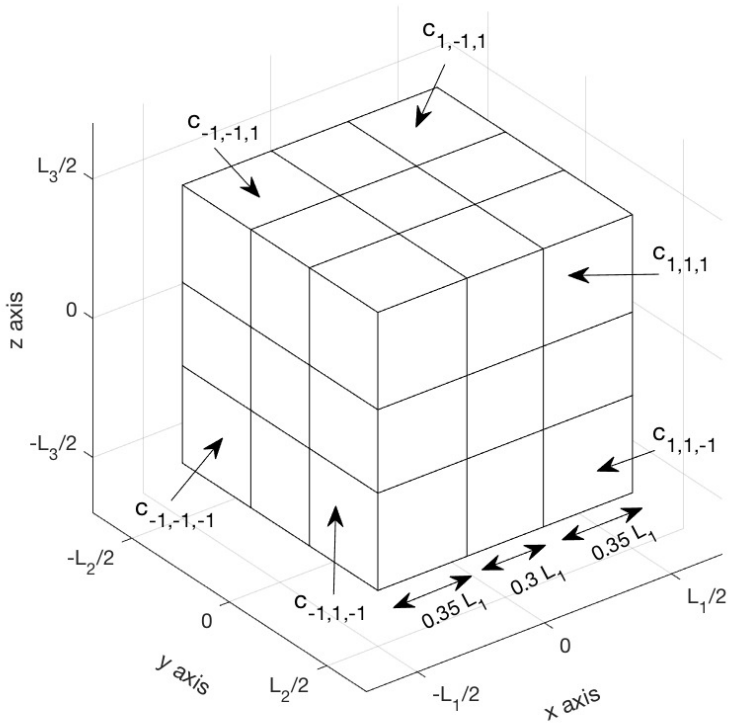

Figure 3.3: Grid view of search space, $\beta = 0.3$.

The sensors move in the second stage according to the following rules:

1. If the first stage locates the source at the *center* of the search space, none of the sensors move, and they continue to collect data from their original positions.

2. If the sub-cuboid detected in the first stage is the *top face*, the sensor with the *second largest* count stays in its place, whereas the other four sensors are moved to the four top edges of the building.

3. If the source is located close to the *bottom face*, the sensor with the *lowest* count remains in its place, whereas the other four sensors move to the four bottom edges.

4. If the source is detected near a *top corner*, the three sensors with the largest counts remain where they are, while the other two sensors are moved to the closest edges.

5. If the source is detected near a *bottom corner*, only two sensors with the largest counts remain where they are, while the other three sensors are moved to the closest edges.

6. If the detected sub-cuboid is a *vertical face*, the sensor with the *lowest* count does not move. The remaining four sensors are moved to the four closest edges.

7. If the first stage detects the sub-cuboid *top edge*, the two sensors with the highest counts do not move; the other three sensors move to form a triangular shape, with two sensors along the two side edges and one above the top edge.

8. If the sub-cuboid is classified as a *bottom edge*, the sensor with the highest counts stays in its place, while the other four sensors move to the bottom plane of the building, two at the bottom corners and two along the bottom edges.

9. If the first stage classifies the source as a *vertical edge*, then the sensor with the highest count stays in its place, and the four remaining sensors are moved to the four closest corners.

Table 3.1 presents the details of what has been described above.

If fewer than five faces are accessible, the algorithm must be modified appropriately. Figs. 3.4 and 3.5 show the first stage and available second stage locations when three faces are accessible (opposite faces accessible), and Table 3.2 presents the details of where to move in the second stage. Similarly, Figs. 3.6 and 3.7 show the first and second stage locations when three faces are accessible (adjacent faces accessible), and the algorithm is shown in Table 3.3. Finally, Figs. 3.8 and 3.9 and Table 3.4 show the corresponding information for the case that two faces are accessible.

The maximum distance a drone must traverse is the longest diagonal distance of the search space. Suppose $L_3 \geq L_2 \geq L_1$. Because the drone can only move outside the perimeter of the building, this distance is $d = \sqrt{(L_1 + 2\alpha)^2 + (L_2 + 2\alpha)^2} + L_3$. The maximum number of locations from which the sensors collect data is nine (considering both stages). This means that the sensors would have to move only once in order to visit the maximum number of locations. It is assumed that the drones do not collect data while moving from first stage to second stage locations, therefore only upper bounds on movement times are considered in the adaptive algorithm. Assuming that the average speed of each drone is v m/s, the total time required by the adaptive algorithm is

$$T_{tot} = \max_i \{T_i\} + \frac{d}{v} \tag{3.2}$$

where T_i is the time spent collecting data at each sensor location.

Type	Initial Location Estimate	Unchanged Sensor Locations	Changed Sensor Locations
Center	$c_{[0,0,0]}$	$s_{[0,0,1]}$, $s_{[1,0,0]}$, $s_{[0,-1,0]}$, $s_{[0,1,0]}$,$s_{[-1,0,0]}$	None
Top Face	$c_{[0,0,1]}$	$s_{[0,-1,0]}$	$s_{[-1,0,1]}$, $s_{[1,0,1]}$, $s_{[0,1,1]}$, $s_{[0,-1,1]}$
Bottom Face	$c_{[0,0,-1]}$	$s_{[0,0,1]}$	$s_{[-1,0,-1]}$, $s_{[1,0,-1]}$, $s_{[0,1,-1]}$, $s_{[0,-1,-1]}$
Top Corner	$c_{[1,-1,1]}$	$s_{[0,0,1]}$,$s_{[1,0,0]}$, $s_{[0,-1,0]}$	$s_{[1,0,1]}$, $s_{[1,-1,0]}$
	$c_{[1,1,1]}$	$s_{[0,0,1]}$, $s_{[1,0,0]}$, $s_{[0,1,0]}$	$s_{[1,0,1]}$, $s_{[1,1,0]}$
	$c_{[-1,1,1]}$	$s_{[0,0,1]}$, $s_{[-1,0,0]}$, $s_{[0,1,0]}$	$s_{[-1,0,1]}$, $s_{[-1,1,0]}$
	$c_{[-1,-1,1]}$	$s_{[0,0,1]}$, $s_{[-1,0,0]}$, $s_{[0,-1,0]}$	$s_{[-1,0,1]}$, $s_{[-1,-1,0]}$
Bottom Corner	$c_{[1,-1,-1]}$	$s_{[1,0,0]}$,$s_{[0,-1,0]}$	$s_{[1,0,-1]}$, $s_{[0,-1,-1]}$, $s_{[1,-1,0]}$
	$c_{[1,1,-1]}$	$s_{[1,0,0]}$, $s_{[0,1,0]}$	$s_{[1,0,-1]}$, $s_{[0,1,-1]}$, $s_{[1,1,0]}$
	$c_{[-1,-1,-1]}$	$s_{[-1,0,0]}$, $s_{[0,-1,0]}$	$s_{[-1,0,-1]}$, $s_{[0,-1,-1]}$, $s_{[-1,-1,0]}$
	$c_{[-1,1,-1]}$	$s_{[-1,0,0]}$,$s_{[0,1,0]}$	$s_{[-1,0,-1]}$, $s_{[0,1,-1]}$, $s_{[-1,1,0]}$
Vertical Face	$c_{[-1,0,0]}$	$s_{[1,0,0]}$	$s_{[-1,-1,0]}$, $s_{[-1,1,0]}$, $s_{[-1,0,-1]}$, $s_{[-1,0,1]}$
	$c_{[1,0,0]}$	$s_{[-1,0,0]}$	$s_{[1,-1,0]}$, $s_{[1,1,0]}$, $s_{[1,0,-1]}$, $s_{[1,0,1]}$
	$c_{[0,1,0]}$	$s_{[0,-1,0]}$	$s_{[-1,1,0]}$, $s_{[1,1,0]}$, $s_{[0,1,1]}$, $s_{[0,1,-1]}$
	$c_{[0,-1,0]}$	$s_{[0,1,0]}$	$s_{[-1,-1,0]}$, $s_{[1,-1,0]}$, $s_{[0,-1,1]}$, $s_{[0,-1,-1]}$
Top Edge	$c_{[1,0,1]}$	$s_{[0,0,1]}$, $s_{[1,0,0]}$	$s_{[1,1,0]}$, $s_{[1,-1,0]}$, $s_{[1,0,1]}$
	$c_{[-1,0,1]}$	$s_{[0,0,1]}$, $s_{[-1,0,0]}$	$s_{[-1,1,0]}$, $s_{[-1,-1,0]}$, $s_{[-1,0,1]}$
	$c_{[0,1,1]}$	$s_{[0,0,1]}$, $s_{[0,1,0]}$	$s_{[-1,1,0]}$, $s_{[1,1,0]}$, $s_{[0,1,1]}$
	$c_{[0,-1,1]}$	$s_{[0,0,1]}$, $s_{[0,-1,0]}$	$s_{[-1,-1,0]}$, $s_{[1,-1,0]}$, $s_{[0,-1,1]}$
Bottom Edge	$c_{[1,0,-1]}$	$s_{[1,0,0]}$	$s_{[1,-1,-1]}$, $s_{[1,1,-1]}$, $s_{[0,-1,-1]}$, $s_{[0,1,-1]}$
	$c_{[-1,0,-1]}$	$s_{[-1,0,0]}$	$s_{[-1,-1,-1]}$, $s_{[-1,1,-1]}$, $s_{[0,-1,-1]}$, $s_{[0,1,-1]}$
	$c_{[0,1,-1]}$	$s_{[0,1,0]}$	$s_{[-1,1,-1]}$, $s_{[1,1,-1]}$, $s_{[-1,0,-1]}$, $s_{[1,0,-1]}$
	$c_{[0,-1,-1]}$	$s_{[0,-1,0]}$	$s_{[-1,-1,-1]}$, $s_{[1,-1,-1]}$, $s_{[-1,0,-1]}$, $s_{[1,0,-1]}$
Vertical Edge	$c_{[1,-1,0]}$	$s_{[1,0,0]}$	$s_{[1,-1,-1]}$, $s_{[1,1,-1]}$, $s_{[1,1,1]}$, $s_{[1,-1,1]}$
	$c_{[1,1,0]}$	$s_{[1,0,0]}$	$s_{[1,-1,-1]}$, $s_{[1,1,-1]}$, $s_{[1,1,1]}$, $s_{[1,-1,1]}$
	$c_{[-1,1,0]}$	$s_{[-1,0,0]}$	$s_{[-1,-1,-1]}$, $s_{[-1,1,-1]}$, $s_{[-1,1,1]}$, $s_{[-1,-1,1]}$
	$c_{[-1,-1,0]}$	$s_{[-1,0,0]}$	$s_{[-1,-1,-1]}$, $s_{[-1,1,-1]}$, $s_{[-1,1,1]}$, $s_{[-1,-1,1]}$

Table 3.1: Location of sensors for adaptive algorithm, five accessible faces.

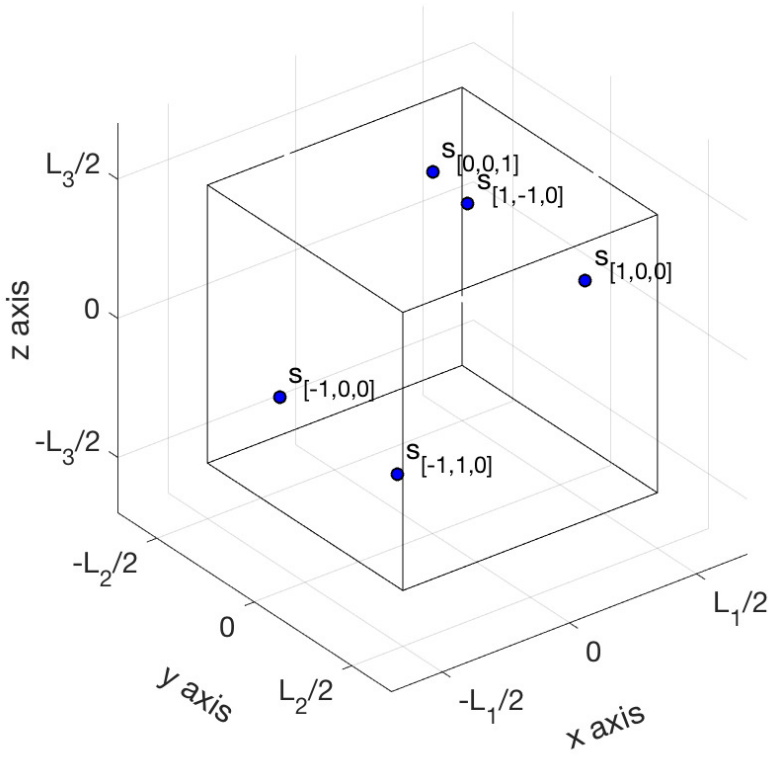

Figure 3.4: Drone locations for the first stage, three faces accessible (opposite).

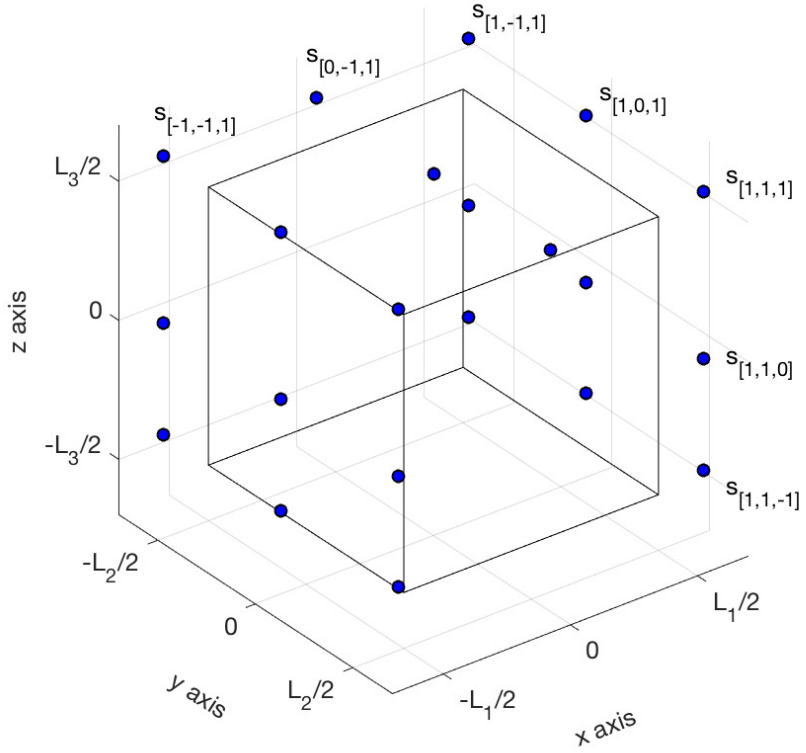

Figure 3.5: Possible drone locations for the second stage, three faces accessible (opposite).

Type	Initial Location Estimate	Unchanged Sensor Locations	Changed Sensor Locations
Center	$c_{[0,0,0]}$	$s_{[0,0,1]}$, $s_{[1,0,0]}$, $s_{[-1,-1,0]}$, $s_{[1,1,0]}$,$s_{[-1,0,0]}$	None
Top Face	$c_{[0,0,1]}$	$s_{[-1,-1,0]}$	$s_{[-1,0,1]}$, $s_{[1,0,1]}$, $s_{[0,1,1]}$, $s_{[0,-1,1]}$
Bottom Face	$c_{[0,0,-1]}$	$s_{[0,0,1]}$	$s_{[-1,0,-1]}$, $s_{[1,0,-1]}$, $s_{[1,1,-1]}$, $s_{[-1,-1,-1]}$
Top Corner	$c_{[1,-1,1]}$	$s_{[0,0,1]}$,$s_{[1,0,0]}$, $s_{[-1,-1,0]}$	$s_{[1,0,1]}$, $s_{[1,-1,0]}$
	$c_{[1,1,1]}$	$s_{[0,0,1]}$, $s_{[1,0,0]}$, $s_{[1,1,0]}$	$s_{[1,0,1]}$, $s_{[1,1,0]}$
	$c_{[-1,1,1]}$	$s_{[0,0,1]}$, $s_{[-1,0,0]}$, $s_{[1,1,0]}$	$s_{[-1,0,1]}$, $s_{[-1,1,0]}$
	$c_{[-1,-1,1]}$	$s_{[0,0,1]}$, $s_{[-1,0,0]}$, $s_{[0,-1,0]}$	$s_{[-1,0,1]}$, $s_{[-1,-1,0]}$
Bottom Corner	$c_{[1,-1,-1]}$	$s_{[1,0,0]}$,$s_{[-1,-1,0]}$	$s_{[1,0,-1]}$, $s_{[-1,-1,-1]}$, $s_{[1,-1,0]}$
	$c_{[1,1,-1]}$	$s_{[1,0,0]}$, $s_{[1,1,0]}$	$s_{[1,0,-1]}$, $s_{[1,1,-1]}$, $s_{[1,1,0]}$
	$c_{[-1,-1,-1]}$	$s_{[-1,0,0]}$, $s_{[0,-1,0]}$	$s_{[-1,0,-1]}$, $s_{[-1,-1,-1]}$, $s_{[-1,-1,0]}$
	$c_{[-1,1,-1]}$	$s_{[-1,0,0]}$,$s_{[1,1,0]}$	$s_{[-1,0,-1]}$, $s_{[1,1,-1]}$, $s_{[-1,1,0]}$
Vertical Face	$c_{[-1,0,0]}$	$s_{[1,0,0]}$	$s_{[-1,-1,0]}$, $s_{[-1,1,0]}$, $s_{[-1,0,-1]}$, $s_{[-1,0,1]}$
	$c_{[1,0,0]}$	$s_{[-1,0,0]}$	$s_{[1,-1,0]}$, $s_{[1,1,0]}$, $s_{[1,0,-1]}$, $s_{[1,0,1]}$
	$c_{[0,1,0]}$	$s_{[-1,-1,0]}$	$s_{[-1,1,0]}$, $s_{[1,1,0]}$, $s_{[0,1,1]}$, $s_{[1,1,-1]}$
	$c_{[0,-1,0]}$	$s_{[1,1,0]}$	$s_{[-1,-1,0]}$, $s_{[1,-1,0]}$, $s_{[0,-1,1]}$, $s_{[-1,-1,-1]}$
Top Edge	$c_{[1,0,1]}$	$s_{[0,0,1]}$, $s_{[1,0,0]}$	$s_{[1,1,0]}$, $s_{[1,-1,0]}$, $s_{[1,0,1]}$
	$c_{[-1,0,1]}$	$s_{[0,0,1]}$, $s_{[-1,0,0]}$	$s_{[-1,1,0]}$, $s_{[-1,-1,0]}$, $s_{[-1,0,1]}$
	$c_{[0,1,1]}$	$s_{[0,0,1]}$, $s_{[1,1,0]}$	$s_{[-1,1,0]}$, $s_{[1,1,0]}$, $s_{[0,1,1]}$
	$c_{[0,-1,1]}$	$s_{[0,0,1]}$, $s_{[-1,-1,0]}$	$s_{[-1,-1,0]}$, $s_{[1,-1,0]}$, $s_{[0,-1,1]}$
Bottom Edge	$c_{[1,0,-1]}$	$s_{[1,0,0]}$	$s_{[1,-1,-1]}$, $s_{[1,1,-1]}$, $s_{[-1,-1,-1]}$, $s_{[1,1,-1]}$
	$c_{[-1,0,-1]}$	$s_{[-1,0,0]}$	$s_{[-1,-1,-1]}$, $s_{[-1,1,-1]}$, $s_{[-1,-1,-1]}$, $s_{[1,1,-1]}$
	$c_{[0,1,-1]}$	$s_{[0,1,0]}$	$s_{[-1,1,-1]}$, $s_{[1,1,-1]}$, $s_{[-1,0,-1]}$, $s_{[1,0,-1]}$
	$c_{[0,-1,-1]}$	$s_{[-1,-1,0]}$	$s_{[-1,-1,-1]}$, $s_{[1,-1,-1]}$, $s_{[-1,0,-1]}$, $s_{[1,0,-1]}$
Vertical Edge	$c_{[1,-1,0]}$	$s_{[1,0,0]}$	$s_{[1,-1,-1]}$, $s_{[1,1,-1]}$, $s_{[1,1,1]}$, $s_{[1,-1,1]}$
	$c_{[1,1,0]}$	$s_{[1,0,0]}$	$s_{[1,-1,-1]}$, $s_{[1,1,-1]}$, $s_{[1,1,1]}$, $s_{[1,-1,1]}$
	$c_{[-1,1,0]}$	$s_{[-1,0,0]}$	$s_{[-1,-1,-1]}$, $s_{[-1,1,-1]}$, $s_{[-1,1,1]}$, $s_{[-1,-1,1]}$
	$c_{[-1,-1,0]}$	$s_{[-1,0,0]}$	$s_{[-1,-1,-1]}$, $s_{[-1,1,-1]}$, $s_{[-1,1,1]}$, $s_{[-1,-1,1]}$

Table 3.2: Location of sensors for adaptive algorithm, three accessible faces (opposite).

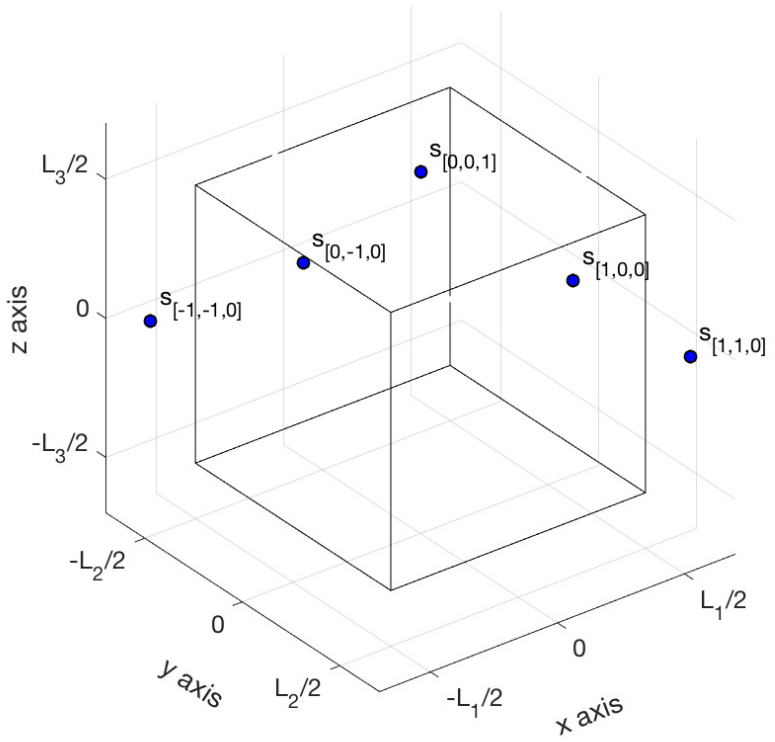

Figure 3.6: Drone locations for the first stage, three faces accessible (adjacent).

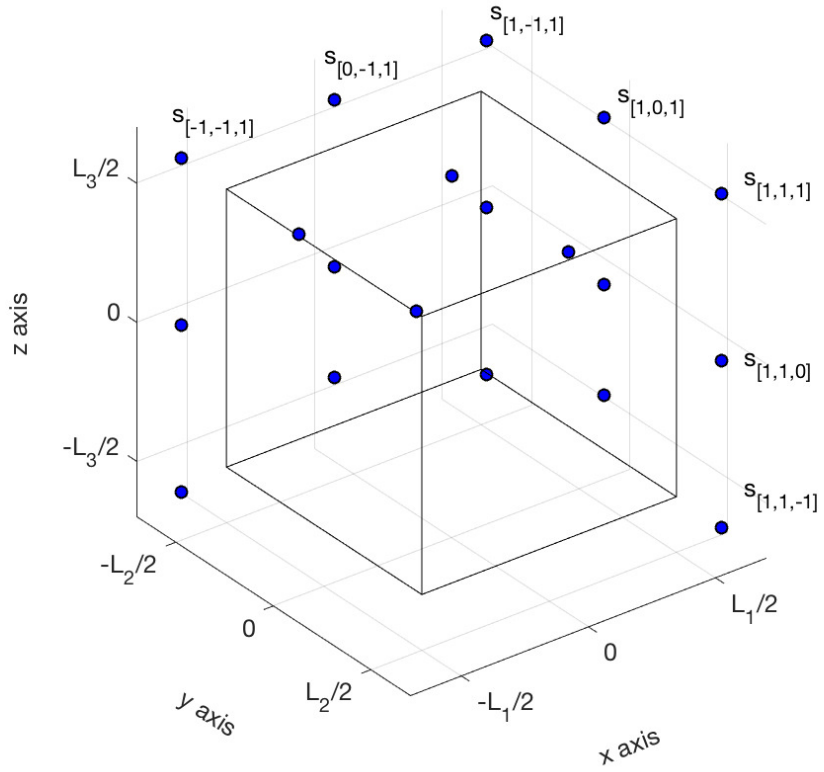

Figure 3.7: Possible drone locations for the second stage, three faces accessible (adjacent).

Type	Initial Location Estimate	Unchanged Sensor Locations	Changed Sensor Locations
Center	$c_{[0,0,0]}$	$s_{[0,0,1]}$, $s_{[1,0,0]}$, $s_{[0,-1,0]}$, $s_{[-1,-1,0]}$,$s_{[1,1,0]}$	None
Top Face	$c_{[0,0,1]}$	$s_{[0,-1,0]}$	$s_{[-1,0,1]}$, $s_{[1,0,1]}$, $s_{[0,1,1]}$, $s_{[0,-1,1]}$
Bottom Face	$c_{[0,0,-1]}$	$s_{[0,0,1]}$	$s_{[-1,-1,-1]}$, $s_{[1,0,-1]}$, $s_{[1,1,-1]}$, $s_{[0,-1,-1]}$
Top Corner	$c_{[1,-1,1]}$	$s_{[0,0,1]}$,$s_{[1,0,0]}$, $s_{[0,-1,0]}$	$s_{[1,0,1]}$, $s_{[1,-1,0]}$
	$c_{[1,1,1]}$	$s_{[0,0,1]}$, $s_{[1,0,0]}$, $s_{[1,1,0]}$	$s_{[1,0,1]}$, $s_{[1,1,0]}$
	$c_{[-1,1,1]}$	$s_{[0,0,1]}$, $s_{[-1,-1,0]}$, $s_{[1,1,0]}$	$s_{[-1,0,1]}$, $s_{[1,1,0]}$
	$c_{[-1,-1,1]}$	$s_{[0,0,1]}$, $s_{[-1,-1,0]}$, $s_{[0,-1,0]}$	$s_{[-1,0,1]}$, $s_{[-1,-1,0]}$
Bottom Corner	$c_{[1,-1,-1]}$	$s_{[1,0,0]}$,$s_{[0,-1,0]}$	$s_{[1,0,-1]}$, $s_{[0,-1,-1]}$, $s_{[1,-1,0]}$
	$c_{[1,1,-1]}$	$s_{[1,0,0]}$, $s_{[1,1,0]}$	$s_{[1,0,-1]}$, $s_{[1,1,-1]}$, $s_{[1,1,0]}$
	$c_{[-1,-1,-1]}$	$s_{[-1,-1,0]}$, $s_{[0,-1,0]}$	$s_{[-1,0,-1]}$, $s_{[0,-1,-1]}$, $s_{[-1,-1,0]}$
	$c_{[-1,1,-1]}$	$s_{[-1,-1,0]}$,$s_{[1,1,0]}$	$s_{[-1,0,-1]}$, $s_{[1,1,-1]}$, $s_{[1,1,0]}$
Vertical Face	$c_{[-1,0,0]}$	$s_{[1,0,0]}$	$s_{[-1,-1,0]}$, $s_{[1,1,0]}$, $s_{[-1,-1,-1]}$, $s_{[-1,0,1]}$
	$c_{[1,0,0]}$	$s_{[-1,-1,0]}$	$s_{[1,-1,0]}$, $s_{[1,1,0]}$, $s_{[1,0,-1]}$, $s_{[1,0,1]}$
	$c_{[0,1,0]}$	$s_{[0,-1,0]}$	$s_{[-1,1,0]}$, $s_{[1,1,0]}$, $s_{[0,1,1]}$, $s_{[1,1,-1]}$
	$c_{[0,-1,0]}$	$s_{[1,1,0]}$	$s_{[-1,-1,0]}$, $s_{[1,-1,0]}$, $s_{[0,-1,1]}$, $s_{[0,-1,-1]}$
Top Edge	$c_{[1,0,1]}$	$s_{[0,0,1]}$, $s_{[1,0,0]}$	$s_{[1,1,0]}$, $s_{[1,-1,0]}$, $s_{[1,0,1]}$
	$c_{[-1,0,1]}$	$s_{[0,0,1]}$, $s_{[-1,0,0]}$	$s_{[1,1,0]}$, $s_{[-1,-1,0]}$, $s_{[-1,0,1]}$
	$c_{[0,1,1]}$	$s_{[0,0,1]}$, $s_{[1,1,0]}$	$s_{[1,1,0]}$, $s_{[1,1,1]}$, $s_{[0,1,1]}$
	$c_{[0,-1,1]}$	$s_{[0,0,1]}$, $s_{[0,-1,0]}$	$s_{[-1,-1,0]}$, $s_{[1,-1,0]}$, $s_{[0,-1,1]}$
Bottom Edge	$c_{[1,0,-1]}$	$s_{[1,0,0]}$	$s_{[1,-1,-1]}$, $s_{[1,1,-1]}$, $s_{[0,-1,-1]}$, $s_{[0,1,-1]}$
	$c_{[-1,0,-1]}$	$s_{[-1,-1,0]}$	$s_{[-1,-1,-1]}$, $s_{[-1,-1,1]}$, $s_{[0,-1,-1]}$, $s_{[1,1,-1]}$
	$c_{[0,1,-1]}$	$s_{[1,1,0]}$	$s_{[-1,-1,1]}$, $s_{[1,1,-1]}$, $s_{[-1,-1,-1]}$, $s_{[1,0,-1]}$
	$c_{[0,-1,-1]}$	$s_{[0,-1,0]}$	$s_{[1,-1,-1]}$, $s_{[1,-1,-1]}$, $s_{[-1,-1,-1]}$, $s_{[1,1,-1]}$
Vertical Edge	$c_{[1,-1,0]}$	$s_{[1,0,0]}$	$s_{[1,-1,-1]}$, $s_{[1,1,-1]}$, $s_{[1,1,1]}$, $s_{[1,-1,1]}$
	$c_{[1,1,0]}$	$s_{[1,0,0]}$	$s_{[1,-1,-1]}$, $s_{[1,1,-1]}$, $s_{[1,1,1]}$, $s_{[1,-1,1]}$
	$c_{[-1,1,0]}$	$s_{[-1,-1,0]}$	$s_{[-1,-1,-1]}$, $s_{[1,1,-1]}$, $s_{[-1,1,1]}$, $s_{[-1,-1,1]}$
	$c_{[-1,-1,0]}$	$s_{[-1,-1,0]}$	$s_{[-1,-1,-1]}$, $s_{[1,1,-1]}$, $s_{[-1,1,1]}$, $s_{[-1,-1,1]}$

Table 3.3: Location of sensors for adaptive algorithm, three accessible faces (adjacent).

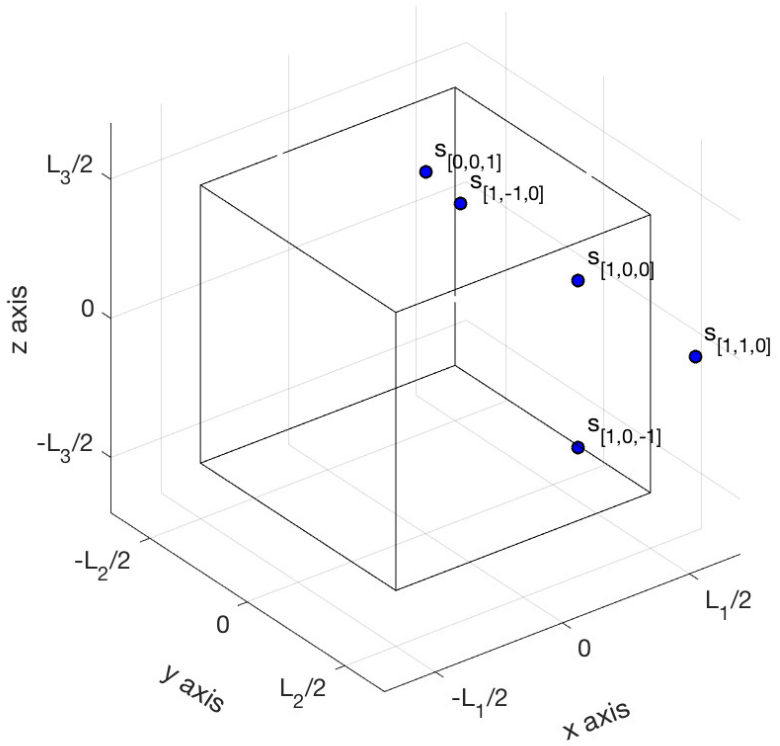

Figure 3.8: Drone locations for the first stage, two faces accessible.

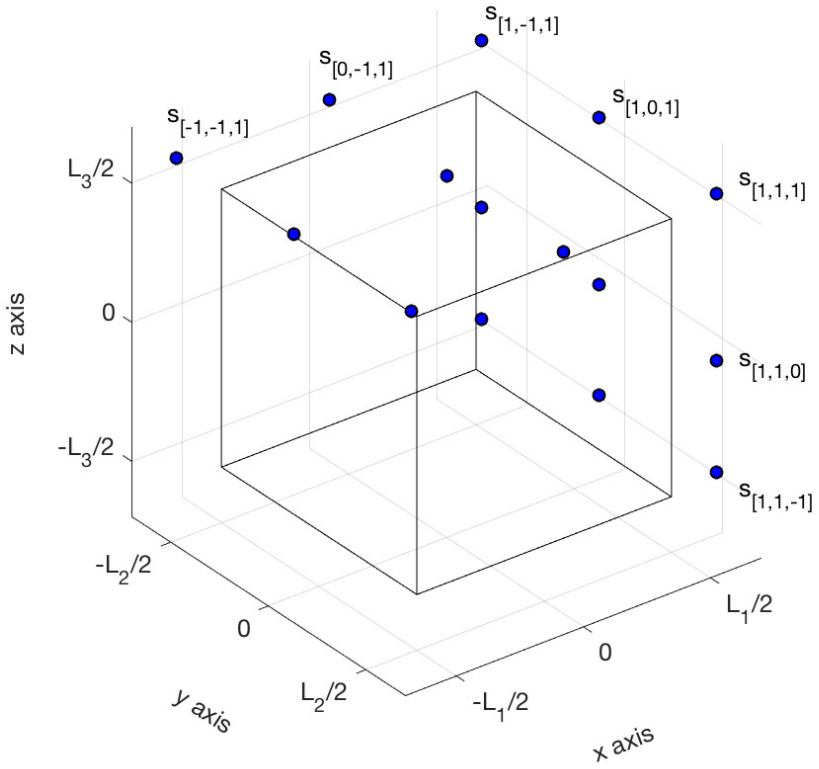

Figure 3.9: Possible drone locations for the second stage, two faces accessible.

Type	Initial Location Estimate	Unchanged Sensor Locations	Changed Sensor Locations
Center	$c_{[0,0,0]}$	$s_{[0,0,1]}$, $s_{[1,0,0]}$, $s_{[1,1,0]}$, $s_{[1,-1,0]}$,$s_{[1,0,-1]}$	None
Top Face	$c_{[0,0,1]}$	$s_{[1,-1,0]}$	$s_{[-1,0,1]}$, $s_{[1,0,1]}$, $s_{[0,1,1]}$, $s_{[0,-1,1]}$
Bottom Face	$c_{[0,0,-1]}$	$s_{[0,0,1]}$	$s_{[-1,1,-1]}$, $s_{[1,0,-1]}$, $s_{[1,1,-1]}$, $s_{[1,-1,-1]}$
Top Corner	$c_{[1,-1,1]}$	$s_{[0,0,1]}$,$s_{[1,0,0]}$, $s_{[1,-1,0]}$	$s_{[1,0,1]}$, $s_{[1,-1,0]}$
	$c_{[1,1,1]}$	$s_{[0,0,1]}$, $s_{[1,0,0]}$, $s_{[1,1,0]}$	$s_{[1,0,1]}$, $s_{[1,1,0]}$
	$c_{[-1,1,1]}$	$s_{[0,0,1]}$, $s_{[1,0,-1]}$, $s_{[1,1,0]}$	$s_{[-1,0,1]}$, $s_{[-1,1,0]}$
	$c_{[-1,-1,1]}$	$s_{[0,0,1]}$, $s_{[1,0,-1]}$, $s_{[1,-1,0]}$	$s_{[-1,0,1]}$, $s_{[-1,-1,0]}$
Bottom Corner	$c_{[1,-1,-1]}$	$s_{[1,0,0]}$,$s_{[1,-1,0]}$	$s_{[1,0,-1]}$, $s_{[1,-1,-1]}$, $s_{[1,-1,0]}$
	$c_{[1,1,-1]}$	$s_{[1,0,0]}$, $s_{[1,1,0]}$	$s_{[1,0,-1]}$, $s_{[1,1,-1]}$, $s_{[1,1,0]}$
	$c_{[-1,-1,-1]}$	$s_{[1,0,-1]}$, $s_{[1,-1,0]}$	$s_{[-1,0,-1]}$, $s_{[1,-1,-1]}$, $s_{[1,-1,0]}$
	$c_{[-1,1,-1]}$	$s_{[1,0,-1]}$,$s_{[1,1,0]}$	$s_{[-1,0,-1]}$, $s_{[1,1,-1]}$, $s_{[1,1,0]}$
Vertical Face	$c_{[-1,0,0]}$	$s_{[1,0,0]}$	$s_{[-1,-1,0]}$, $s_{[-1,1,0]}$, $s_{[-1,1,-1]}$, $s_{[-1,0,1]}$
	$c_{[1,0,0]}$	$s_{[1,0,-1]}$	$s_{[1,-1,0]}$, $s_{[1,1,0]}$, $s_{[1,0,-1]}$, $s_{[1,0,1]}$
	$c_{[0,1,0]}$	$s_{[1,-1,0]}$	$s_{[1,1,1]}$, $s_{[1,1,0]}$, $s_{[0,1,1]}$, $s_{[1,1,-1]}$
	$c_{[0,-1,0]}$	$s_{[1,1,0]}$	$s_{[1,-1,0]}$, $s_{[1,-1,1]}$, $s_{[0,-1,1]}$, $s_{[1,-1,-1]}$
Top Edge	$c_{[1,0,1]}$	$s_{[0,0,1]}$, $s_{[1,0,0]}$	$s_{[1,1,0]}$, $s_{[1,-1,0]}$, $s_{[1,0,1]}$
	$c_{[-1,0,1]}$	$s_{[0,0,1]}$, $s_{[1,0,-1]}$	$s_{[-1,1,0]}$, $s_{[-1,-1,0]}$, $s_{[-1,0,1]}$
	$c_{[0,1,1]}$	$s_{[0,0,1]}$, $s_{[1,1,0]}$	$s_{[-1,1,0]}$, $s_{[1,1,0]}$, $s_{[0,1,1]}$
	$c_{[0,-1,1]}$	$s_{[0,0,1]}$, $s_{[1,-1,0]}$	$s_{[-1,-1,0]}$, $s_{[1,-1,0]}$, $s_{[0,-1,1]}$
Bottom Edge	$c_{[1,0,-1]}$	$s_{[1,0,0]}$	$s_{[1,-1,-1]}$, $s_{[1,1,-1]}$, $s_{[1,-1,-1]}$, $s_{[1,1,-1]}$
	$c_{[-1,0,-1]}$	$s_{[1,0,-1]}$	$s_{[-1,-1,1]}$, $s_{[-1,1,1]}$, $s_{[1,-1,-1]}$, $s_{[1,1,-1]}$
	$c_{[0,1,-1]}$	$s_{[1,1,0]}$	$s_{[-1,1,1]}$, $s_{[1,1,1]}$, $s_{[-1,1,-1]}$, $s_{[1,0,-1]}$
	$c_{[0,-1,-1]}$	$s_{[1,-1,0]}$	$s_{[-1,-1,1]}$, $s_{[1,-1,-1]}$, $s_{[-1,1,-1]}$, $s_{[1,0,-1]}$
Vertical Edge	$c_{[1,-1,0]}$	$s_{[1,0,0]}$	$s_{[1,-1,-1]}$, $s_{[1,1,-1]}$, $s_{[1,1,1]}$, $s_{[1,-1,1]}$
	$c_{[1,1,0]}$	$s_{[1,0,0]}$	$s_{[1,-1,-1]}$, $s_{[1,1,-1]}$, $s_{[1,1,1]}$, $s_{[1,-1,1]}$
	$c_{[-1,1,0]}$	$s_{[1,0,-1]}$	$s_{[-1,-1,1]}$, $s_{[-1,1,-1]}$, $s_{[-1,1,1]}$, $s_{[-1,-1,1]}$
	$c_{[-1,-1,0]}$	$s_{[1,0,-1]}$	$s_{[-1,-1,1]}$, $s_{[-1,1,-1]}$, $s_{[-1,1,1]}$, $s_{[-1,-1,1]}$

Table 3.4: Location of sensors for adaptive algorithm, two accessible faces.

For purposes of comparison, non-adaptive approaches with five sensors are also considered. Here, the combined data from all the sensors is used to estimate the source location.

It is assumed that the five sensors spend an equal amount of time in all locations they visit. The sensors are moved using a scheduling process. If L denotes the number of locations visited, the number of movements required is $n_m = \lceil \frac{L}{5} \rceil - 1$.

	L			
Faces accessible	5	3 (Opposite)	3 (Adjacent)	2
Faces	5	3	3	2
Corners	8	8	7	6
Edges	12	10	9	7
Faces and Corners	13	11	10	8
Faces and Edges	17	13	12	9
Corners and Edges	20	18	16	13
Faces, Corners and Edges	25	21	19	15

Table 3.5: Location of sensors for non-adaptive algorithms with five physical sensors and L fixed sensor locations.

The maximum distance the sensors are required to move is d and the speed of each drone is v m/s. Similar to the adaptive algorithm, the drones are assumed to not collect data while moving. Therefore, the total time required by the non-adaptive algorithms is given by

$$T_{tot} = \frac{L}{5} T_i + \frac{d}{v} n_m \tag{3.3}$$

The values of L considered are presented in Table 3.5. Each column corre-

sponds to Tables 3.1 to 3.4 respectively, based on accessibility situations.

3.3 Performance Results

ML estimation is implemented in both stages using the MATLAB routine `fminsearchbnd` [16]. In the second stage, both first stage and second stage data are used to obtain the final estimate of location. As in Chapter 2, numerical results are presented for a high-rise building of dimension 100 m $\times$100 m $\times$100 m and drones are placed $\alpha = 10$ m away from the search space boundary planes. Good performance results with $\beta = 0.3$, the corners of the central sub-cuboid being $(\pm 15, \pm 15, \pm 15)$. Scintillating detectors based on 0.0762 mNaI are assumed, and the detection efficiency is calculated based on the measurements for the Cs-137 photo-peak region of 662-keV. The background radiation level is assumed to be uniformly distributed at $\lambda_B = 10^3$ cps. The individual values of μ, A, and I are unspecified as before, so that performance depends on those quantities solely through V. The mass attenuation coefficient and the density of air are as specified in Chapter 2.

It is assumed that the average speed of each drone when moving from one set of locations to the next is $v = 14.5$ m/s. It follows that $\frac{d}{v} \approx 20$ seconds. Equations (3.2) and (3.3) are used to calculate total times of movement and data collection. In order to provide a fair comparison, T_{tot} is kept constant for both the adaptive and non-adaptive approaches. To compare the algorithms, the worst case and average performance is determined, using 125 source locations spaced uniformly throughout the search space. These source locations are the same locations used in Chapter 2.

As before, average distance between estimated and true location of the source is calculated as $\bar{D} = \frac{1}{n}\sum_{j=1}^{n} \hat{D}_j$, where $\hat{D}_j$ is this distance for the jth iteration, as described in Chapter 2.

The worst case errors for five accessible faces are depicted in Fig. 3.10 and the average errors in Fig. 3.11. The worst case error for the adaptive method occurs at the center of the search space, due to the fact that the center location is relatively far away from all sensors.

The adaptive method dramatically outperforms the non-adaptive algorithms. The maximum errors for $L = 5$ are relatively large with as much as 7 m for a total time of 100 seconds. The limited number of locations in $L = 8$ implies that more time is spent collecting data than all of the other non-adaptive cases; however, more data collection time does not translate to better estimation of the source.

The best performance for non-adaptive methods is obtained with $L = 12$, which strikes the right balance between high data collection time and good enough spatial diversity. The non-adaptive algorithm with $L = 25$ locations starts off with higher estimation errors, but due to the larger number of sensor locations employed, these methods eventually catch up in performance to the $L = 12$ non-adaptive algorithm in performance.

In Figs. 3.12 and 3.13, in that two opposite sides and the floor are assumed to be inaccessible to the drones, and Figs. 3.14 and 3.15 where two adjacent sides and the floor are inaccessible. In addition, Figs. 3.16 and 3.17 present results for the case that only one side and the roof are accessible. In each case, it is seen that the adaptive algorithm outperforms all of the non-adaptive options.

The four geometric restrictions are compared to one another in Figs. 3.18 and 3.19. As expected, the best performance is achieved when all five faces are accessible to the sensors. However, the performance remains quite good even when fewer faces are available, despite theaccessibility issues.

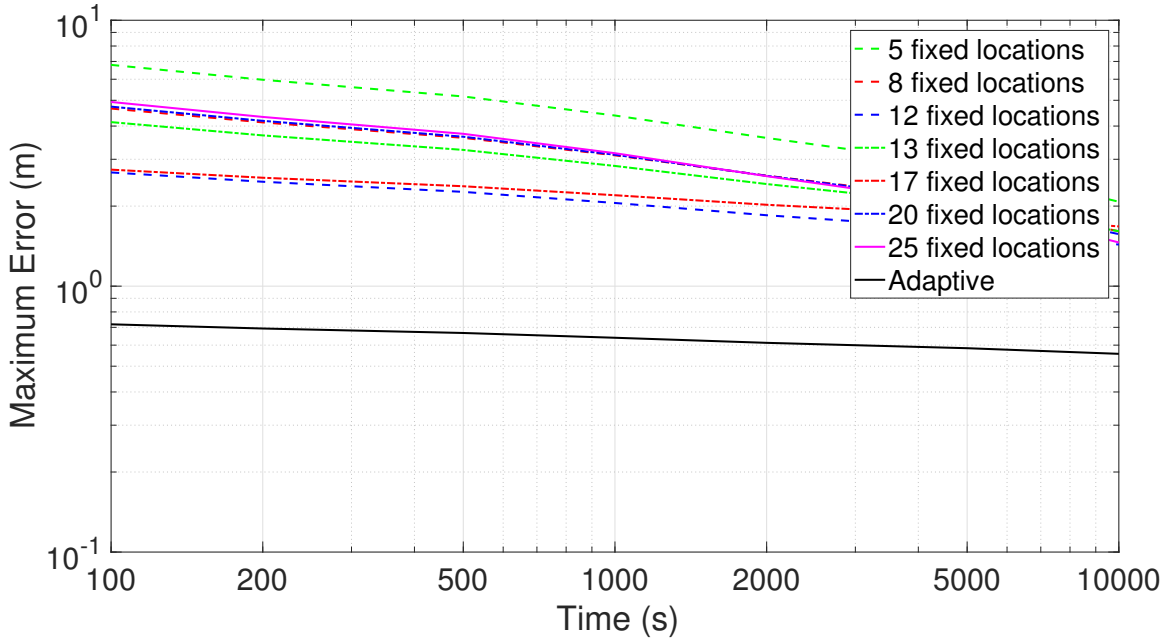

Figure 3.10: Maximum error for adaptive and non-adaptive methods, five accessible faces.

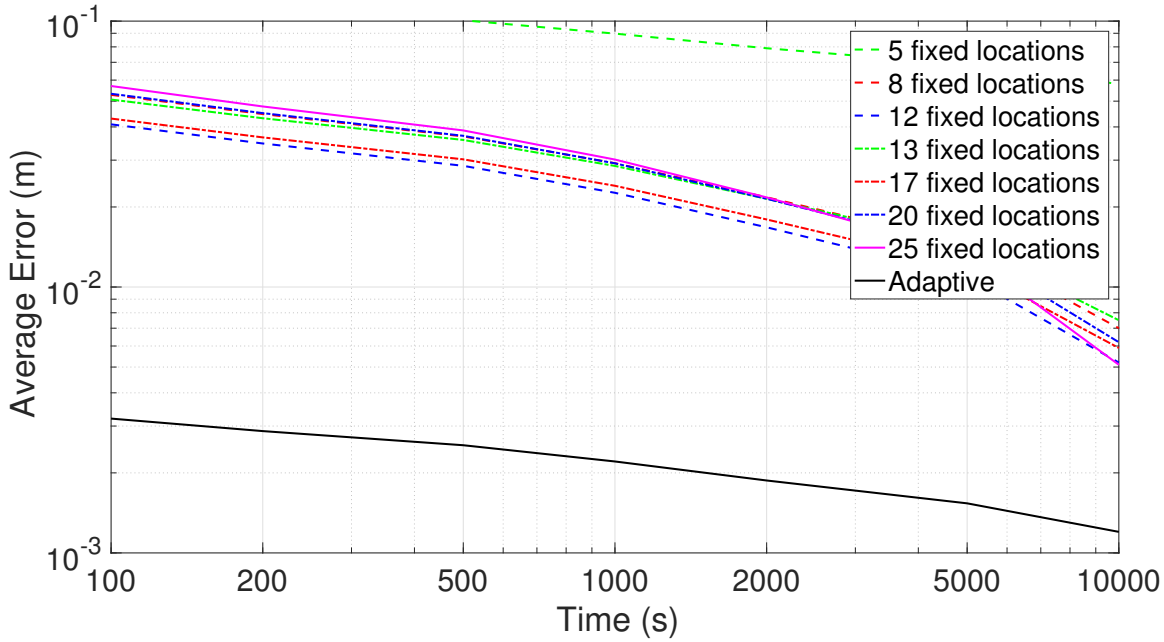

Figure 3.11: Average error for adaptive and non-adaptive methods, five accessible faces.

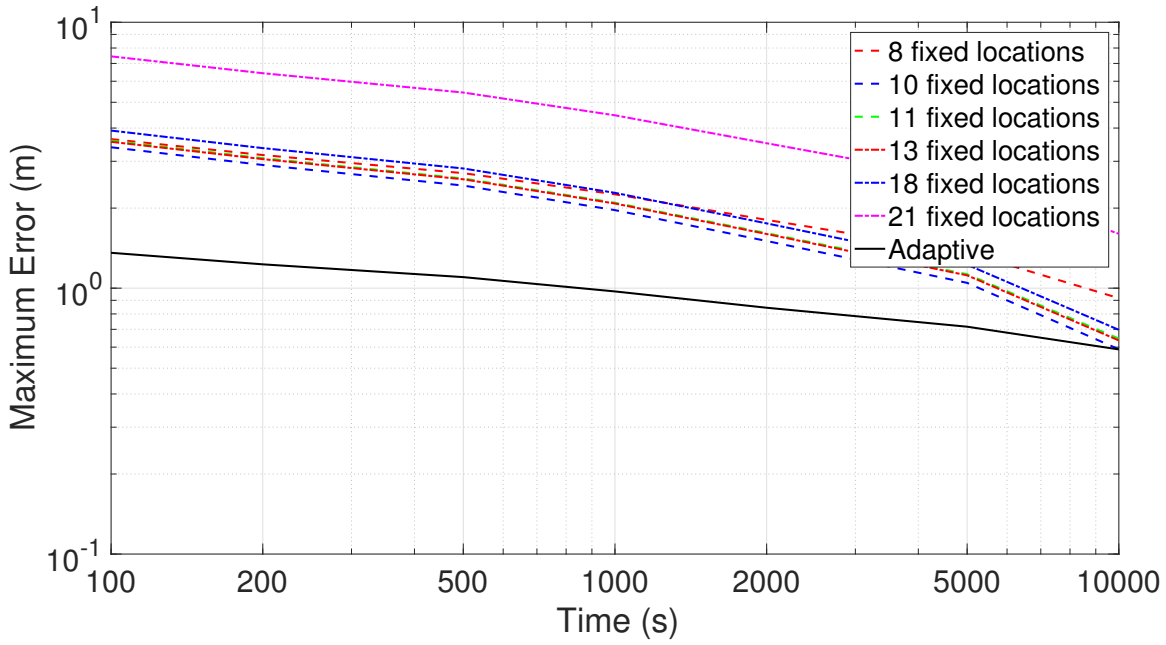

Figure 3.12: Maximum error for adaptive and non-adaptive methods, three faces accessible (opposite).

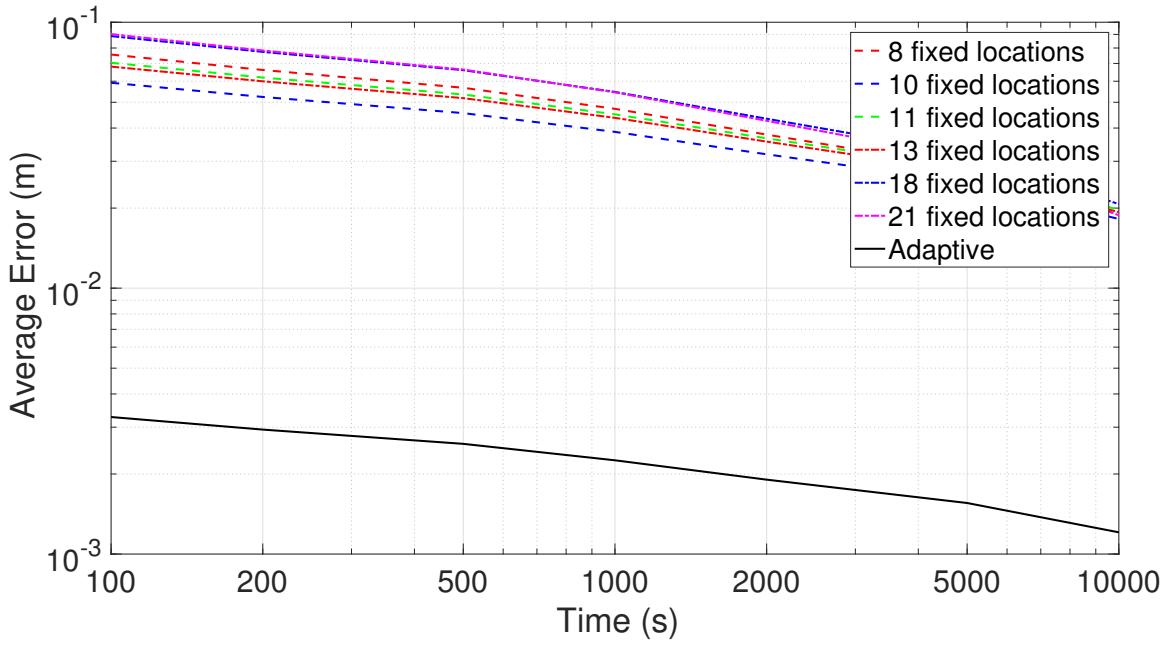

Figure 3.13: Average error for adaptive and non-adaptive methods, three accessible faces (opposite).

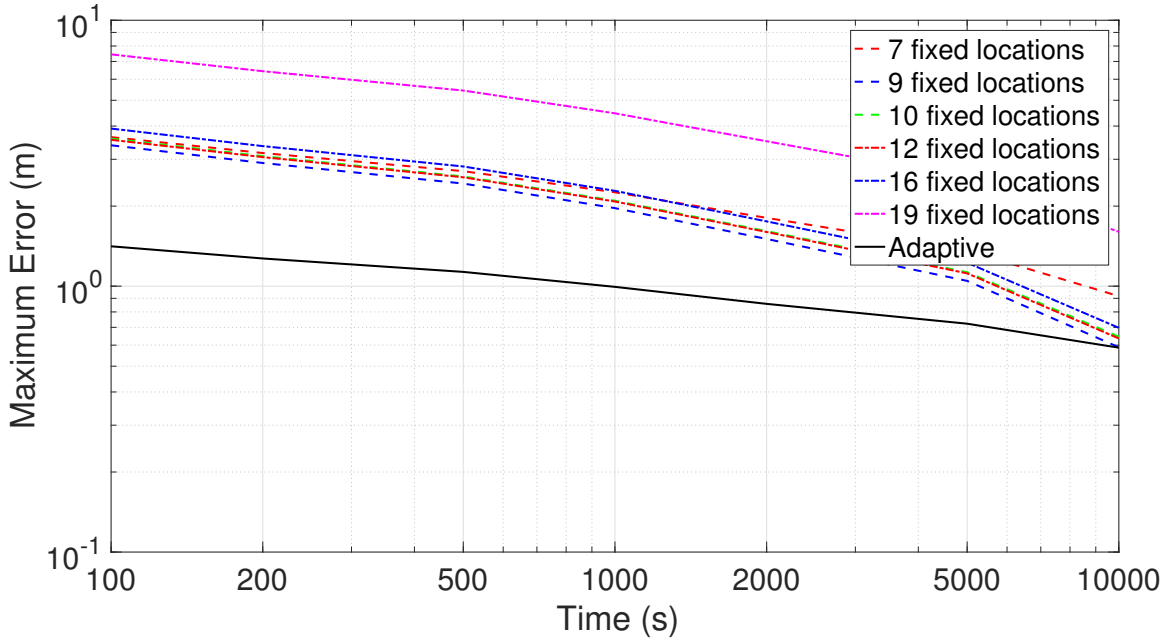

Figure 3.14: Maximum error for adaptive and non-adaptive methods, three accessible faces (adjacent).

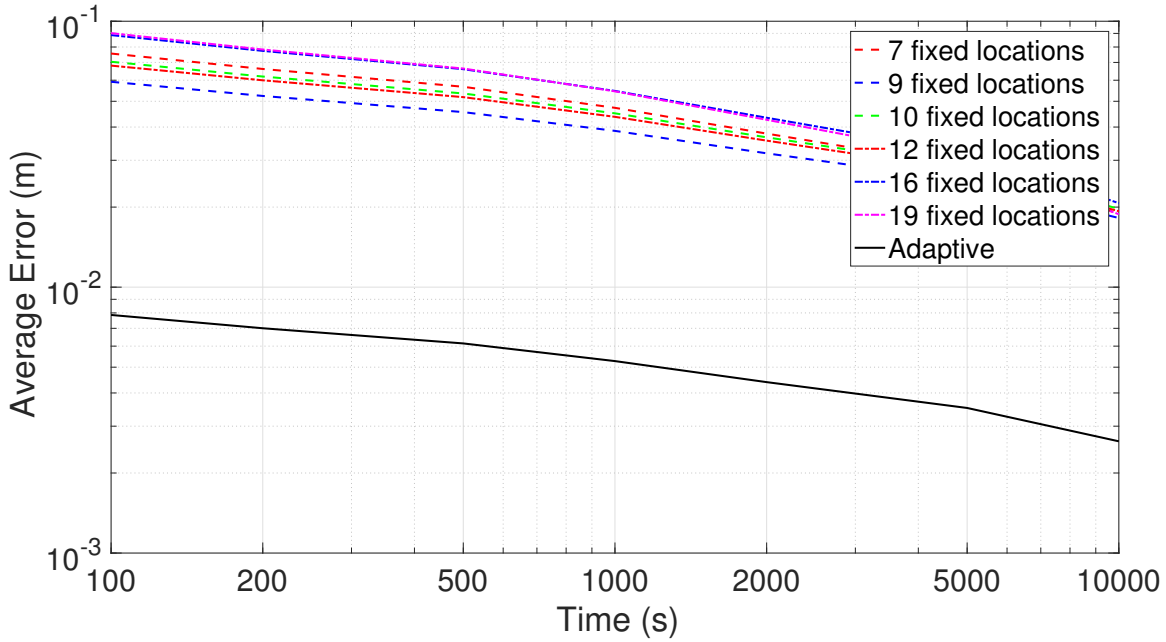

Figure 3.15: Average error for adaptive and non-adaptive methods, three faces accessible (adjacent).

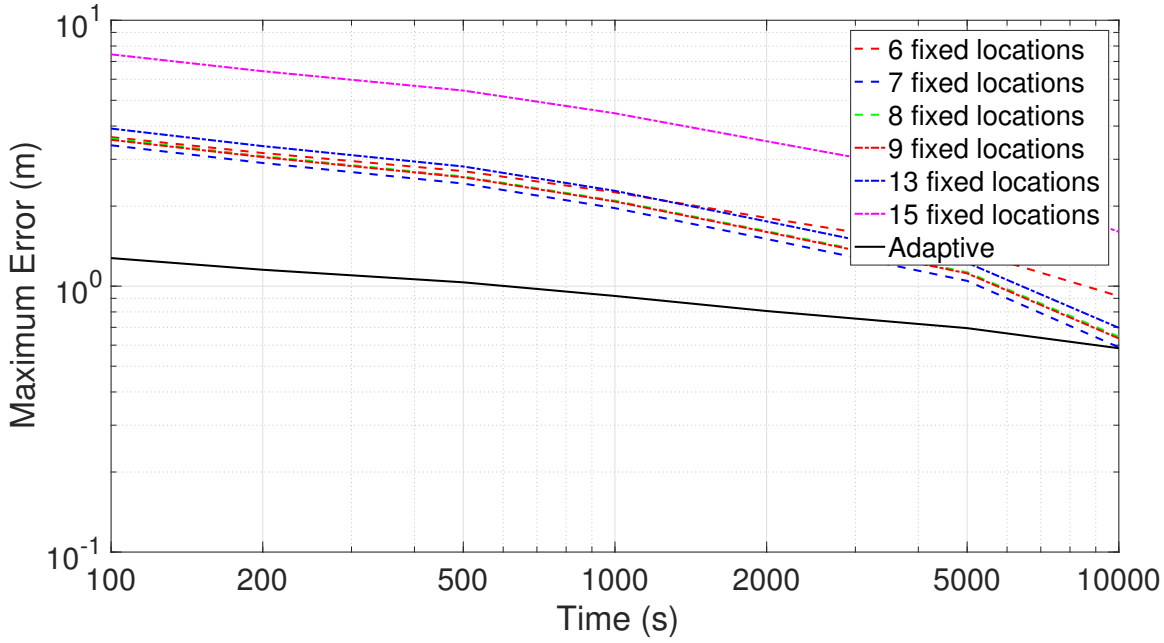

Figure 3.16: Maximum error for adaptive and non-adaptive methods, two faces accessible.

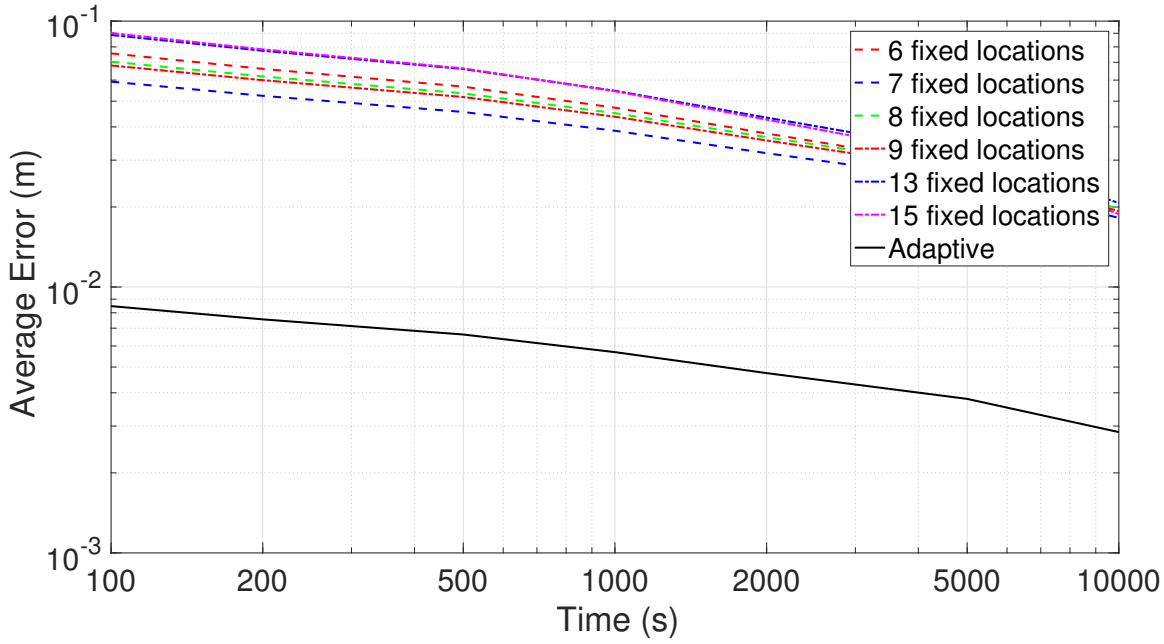

Figure 3.17: Average error for adaptive and non-adaptive methods, two faces accessible.

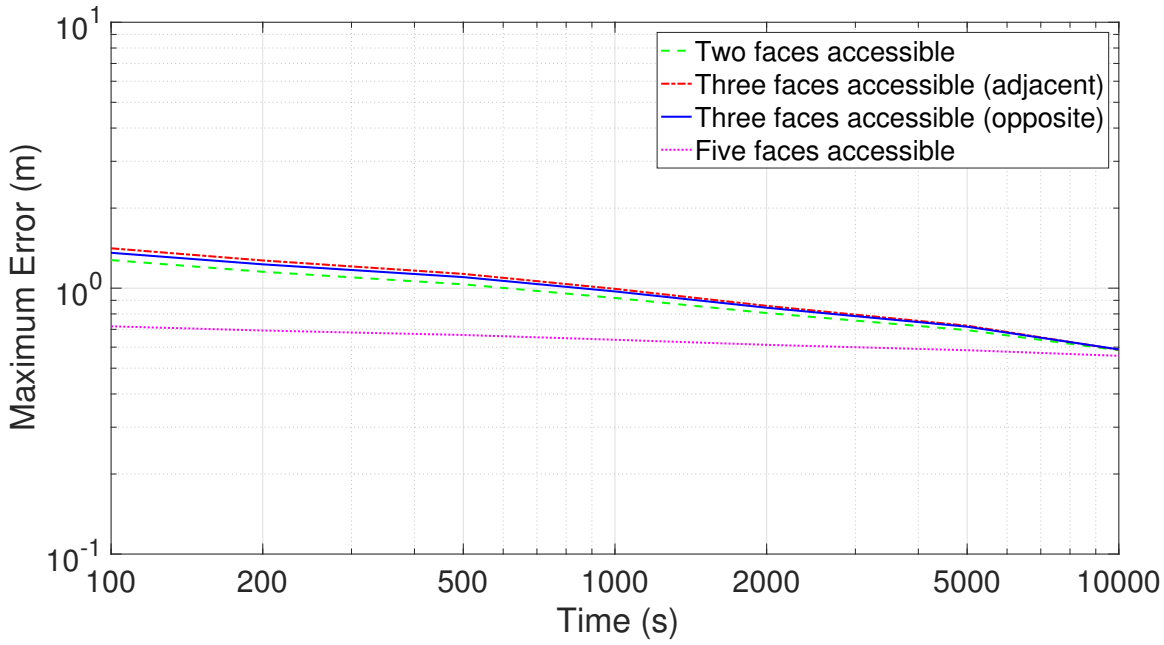

Figure 3.18: Maximum error vs. time, comparison of adaptive algorithms.

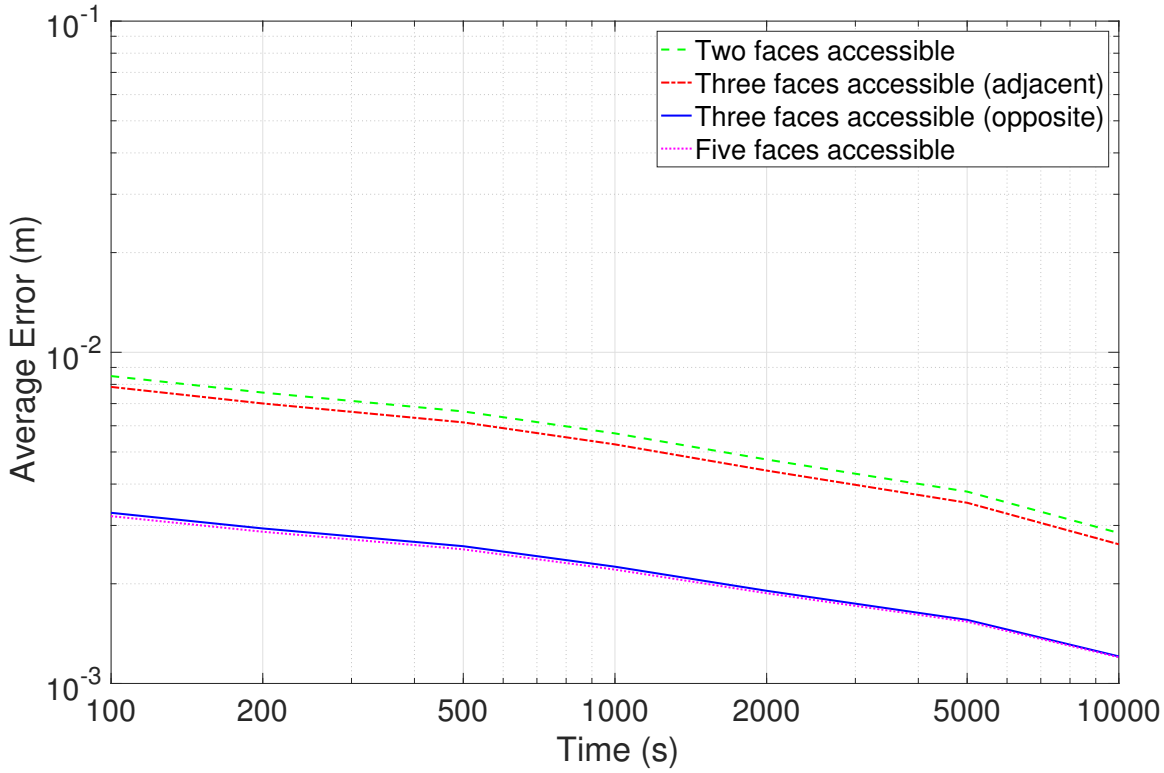

Figure 3.19: Average error vs. time, comparison of adaptive algorithms.

3.4 Conclusion

This chapter proposes an adaptive two-stage method for estimating the location of a radiological point source in a three-dimensional space while incorporating multiple real world constraints. Only five sensors are used to estimate the unknown location and strength of the source. This chapter highlights the advantage of using a small number of detectors in strategically chosen locations, based on a coarse initial estimate of the location of the source. This method greatly outperforms the "fixed sensor location" methods.

Chapter 4

Detection of Non-Homogeneous Attenuation

Building materials such as high-density concrete [24] can cause significant attenuation of a radioactive source if the materials lie between the source and sensor. Due to the nature of a building's structure, such materials are almost certain to affect some sensor locations more than others, leading to non-homogeneous effects. If not taken into account, absorption and the corresponding attenuation can introduce considerable error in the parameter estimates of the source. To our knowledge, estimation in the presence of severe non-homogeneous attenuation has not been explored previously.

This chapter proposes a novel technique to detect unknown non-homogeneous attenuation. A metric is proposed that uses drone data already collected to identify situations in which attenuation non-homogeneity is likely to result in non-negligible estimation error.

4.1 System Model

The system models considered in Sections 2.1 and 3.1 assume that non-homogeneous attenuation is negligible. These models can be generalized to account for non-homogeneous attenuation. Assume that the arrival rate is affected by both gaseous and solid materials in the path between source and sensor. The count arrival rate at the ith sensor location is given by

$$\lambda_i = \lambda_B + \frac{\mu A I}{4\pi D_i^2} e^{-\rho_{\mathrm{g}}(D_i - W_i)} e^{-\rho_{\mathrm{s}} W_i} \tag{4.1}$$

In this equation, the mass attenuation coefficient of a (highly absorbing) solid material is denoted by ρ_{s} and the thickness of this solid material in the path between the source and ith sensor is denoted by W_i. As before, D_i denotes the distance between source and sensor. Because W_i is assumed to be relatively negligible compared to D_i, λ_i can be approximated as

$$\lambda_i \approx \lambda_B + \frac{\mu A I}{4\pi D_i^2} e^{-\rho_{\mathrm{g}} D_i - \rho_{\mathrm{s}} W_i} \tag{4.2}$$

4.2 Attenuation Detection

The estimates of the source location and intensity are denoted as $(\hat{x}_0, \hat{y}_0, \hat{z}_0)$ and $\hat{I}$. These estimates can be obtained from static sensors via the algorithm in Chapter 2 or via sensors that move as in the algorithms of Chapter 3. Note that these algorithms assume that non-homogeneous attenuation is absent. Define $\hat{D}_i$ to be an estimate of D_i based on $(\hat{x}_0, \hat{y}_0, \hat{z}_0)$ and define $\hat{V}$ to be an estimate of V based on $\hat{I}$ so that $\hat{D}_i^2 = (x_i - \hat{x}_0)^2 + (y_i - \hat{y}_0)^2 + (z_i - \hat{z}_0)^2$ and $\hat{V} = \log(\mu A \hat{I})$. We estimate the

arrival rate at the ith location as

$$\hat{\lambda}_i = \lambda_B + \frac{1}{4\pi \hat{D}_i^2} e^{\hat{V} - \rho_g \hat{D}_i} \tag{4.3}$$

Recall that the counts recorded at the ith sensor location is represented as k_i and the time duration over which data is collected is T_i. The quantity $\frac{k_i}{T_i}$ is also a measure of the arrival rate, and it is expected to be close to $\hat{\lambda}_i$ if non-homogeneous attenuation is absent, but there is no reason to expect $\frac{k_i}{T_i}$ to be close to $\hat{\lambda}_i$ if significant non-homogeneous attenuation is present. To estimate the degree to which a given location may be affected by non-homogeneous attenuation, the following normalized error detection metric is proposed:

$$A_i = \frac{\frac{k_i}{T_i} - \hat{\lambda}_i}{\hat{\lambda}_i} \tag{4.4}$$

Using the collection of metrics A_1 through A_L gives a set of statistics that together may be used to detect the presence or absence of non-homogeneous attenuation.

4.3 Performance Results

It is assumed that the sensors have access to all sides of the building, excluding the bottom face. As before, numerical results are presented for a high-rise building of dimension 100 m $\times$100 m $\times$100 m (so that $L_1 = L_2 = L_3 = 100$ m). Drones are placed $\alpha = 10$ m away from the search space boundary planes. There are five physical drones visiting the 25 possible source locations around the search space, as shown in Fig. 3.2. The drones move according to the scheduling process described in Chapter 3 for the non-adaptive algorithms.

The mass attenuation coefficient for air is $\rho_g = 0.0775$ cm^2/g and the density of air is 0.001225 g/cm^3 [24], as described before. The mass attenuation coefficient for solid concrete is assumed to be $\rho_s = 8.236 \times 10^{-2}$ cm^2/g with density 2.3 g/cm^3 [23,24]. These values correspond to a high density formulation of concrete.

Two possible scenarios are considered, a concrete vertical shaft and a concrete vertical wall. In the first scenario, it is assumed that a rectangular structure consisting of four concrete walls runs vertically through the center of the building as shown in Fig. 4.1. In the second case, a single concrete wall is assumed to be present along an entire vertical face of the building as shown in Fig. 4.2. It is further assumed that the four walls of the vertical shaft and the single vertical wall have thicknesses of $\Delta = 0.25$ m.

The values of $\log_{10} |A_i|$ are plotted against increasing data collection time in Figures 4.3 to 4.6. The attenuating structure is assumed to be the vertical shaft as shown in Fig. 4.1. Figure 4.3 places the source at $(40, 40, 0)$, which is close to a side edge of the building. For this source location, 21 sensor locations are affected by non-homogeneous attenuation. Figure 4.4 places the source at $(40, 40, 40)$, close to a corner, and in this case 16 locations experience non-homogeneous attenuation. Figures 4.5 and 4.6 place the source at $(0, 10, 0)$ and $(0, 40, 0)$, close to the center of the building and to the face of the outer wall, respectively, and the numbers of sensor locations affected by non-homogeneous attenuation are correspondingly 20 and 18.

Figs. 4.7 to 4.10 present results for the attenuating vertical wall of Fig. 4.2, the sources are placed in the same locations as before. For the source located at $(40, 40, 0)$, 10 sensor locations are affected by non-homogeneous attenuation. Only 6 sensor locations are affected for the source at $(40, 40, 40)$. When the source is at $(0, 10, 0)$, the number of sensor locations affected is 9. Finally, for the source at $(0, 40, 0)$, a total of 8 sensor locations are affected by non-homogeneous attenuation.

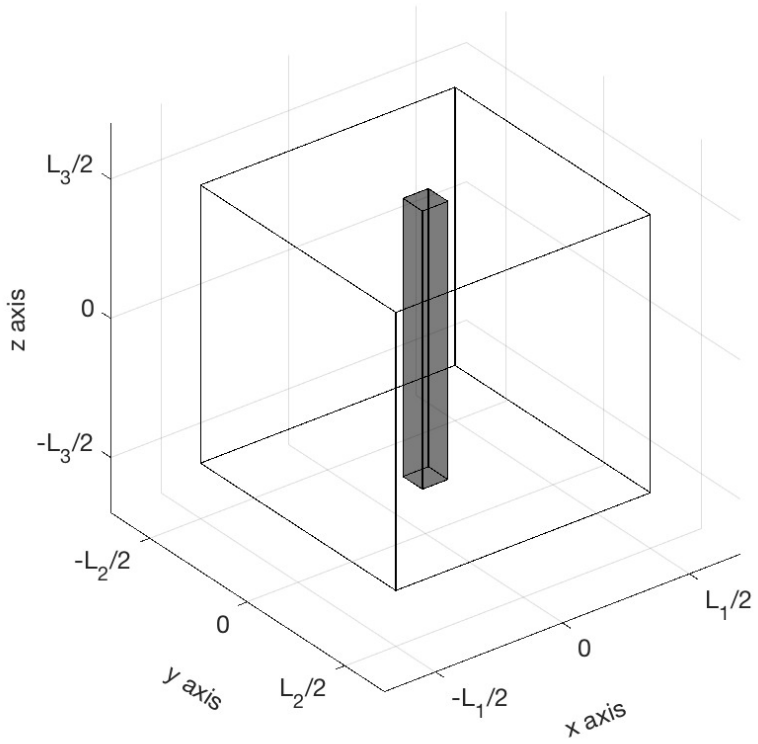

Figure 4.1: Vertical shaft in the center of the building.

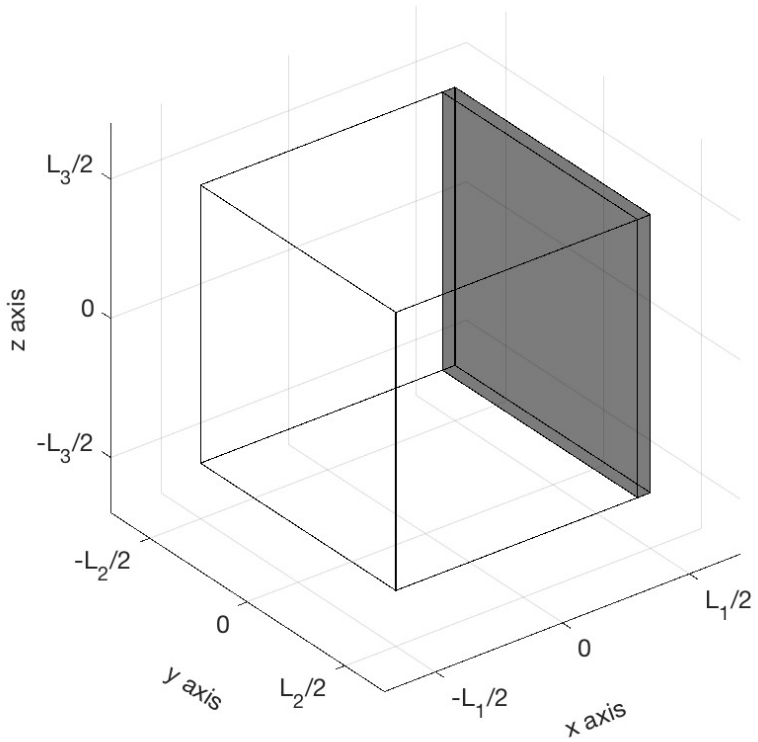

Figure 4.2: Single concrete wall along a face of the building.

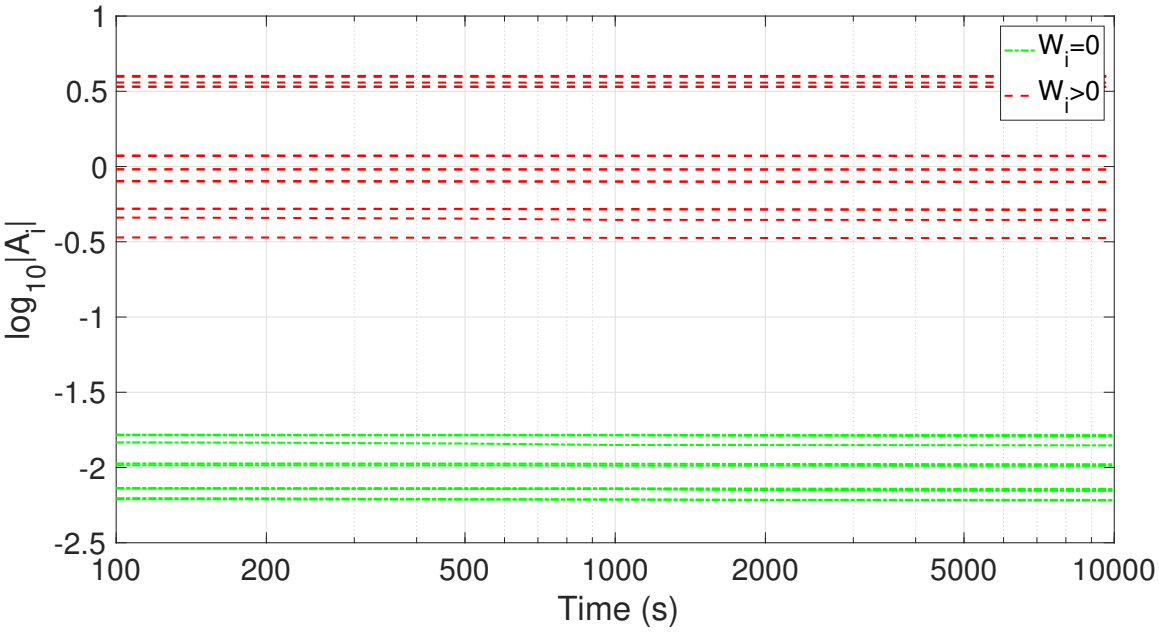

Figure 4.3: Attenuation detection metric vs. time, vertical shaft, source located at $(40, 40, 0)$.

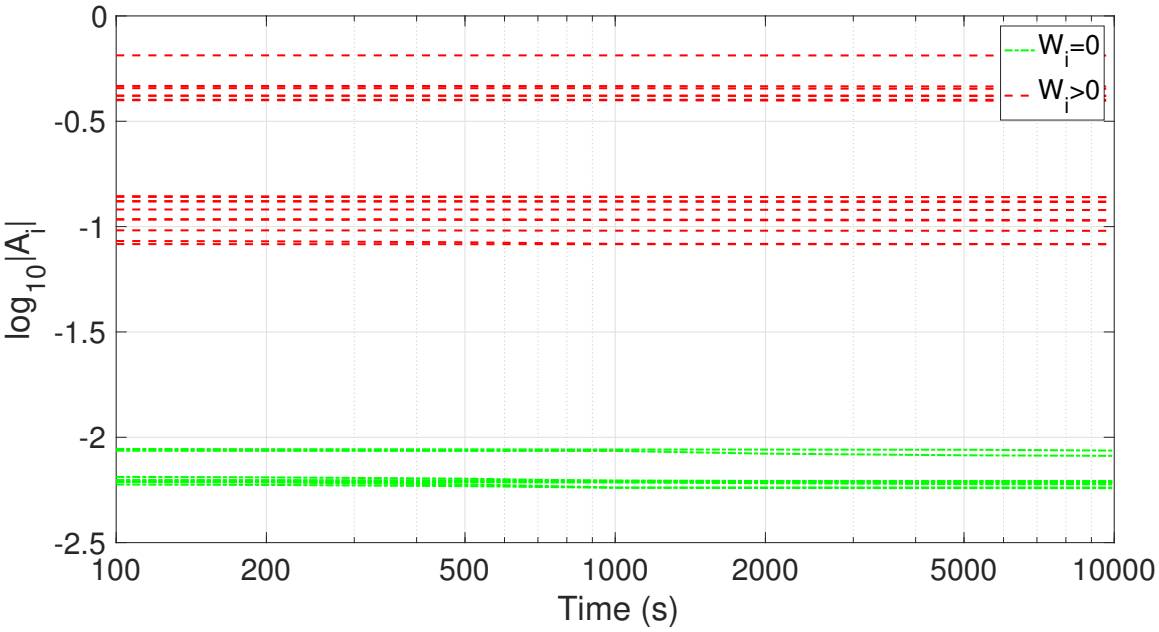

Figure 4.4: Attenuation detection metric vs. time, vertical shaft, source located at $(40, 40, 40)$.

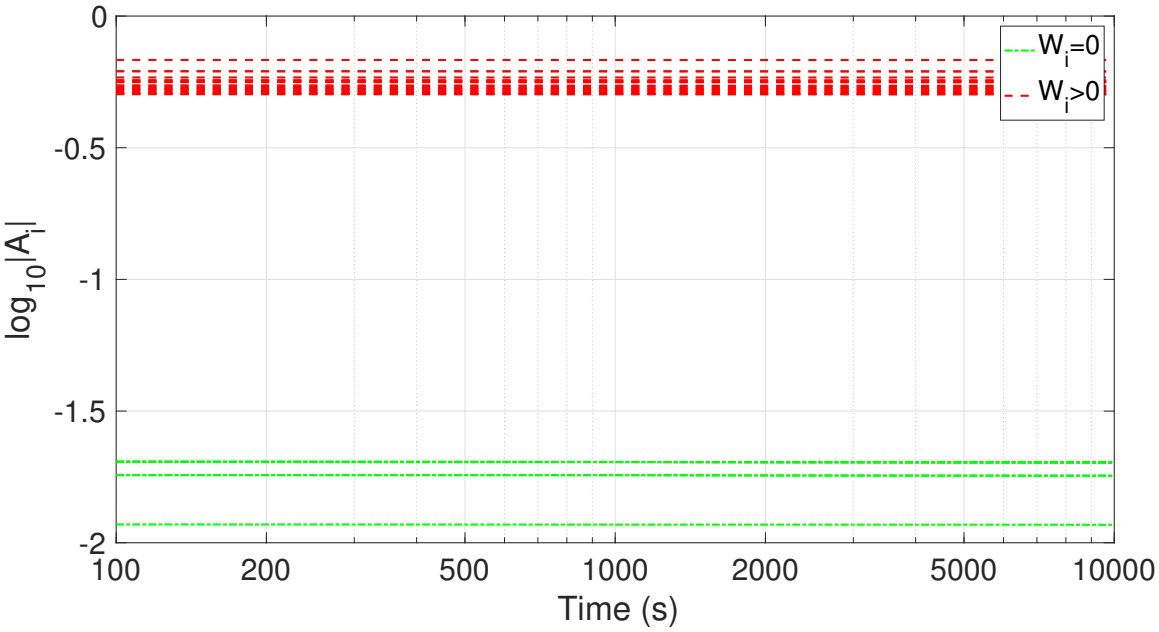

Figure 4.5: Attenuation detection metric vs. time, vertical shaft, source located at $(0, 10, 0)$.

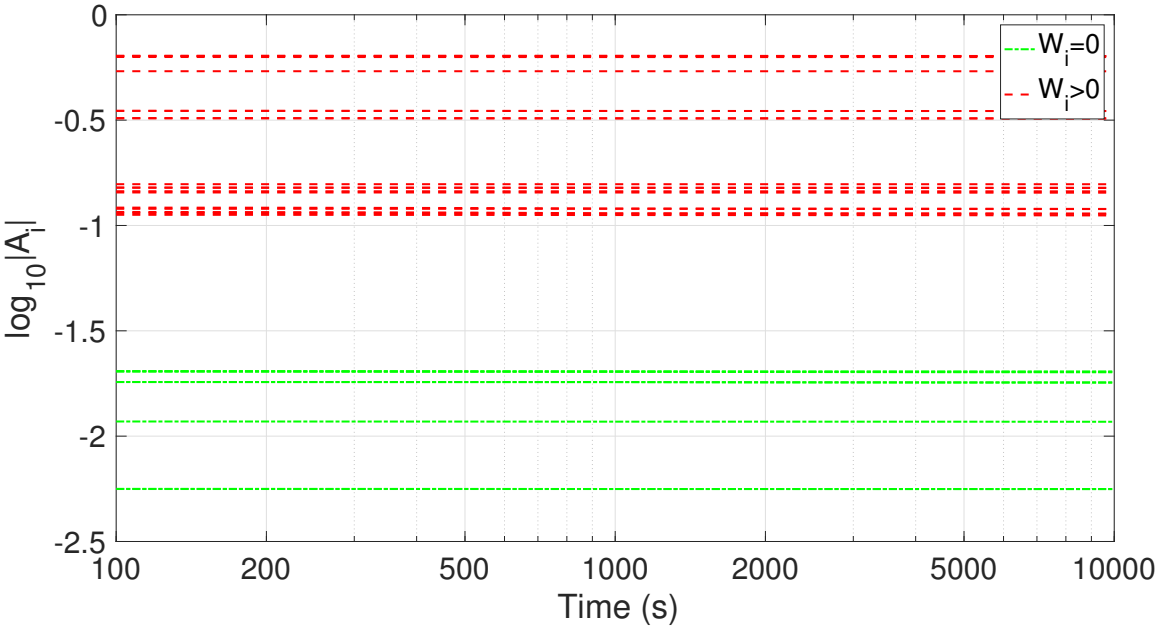

Figure 4.6: Attenuation detection metric vs. time, vertical shaft, source located at $(0, 40, 0)$.

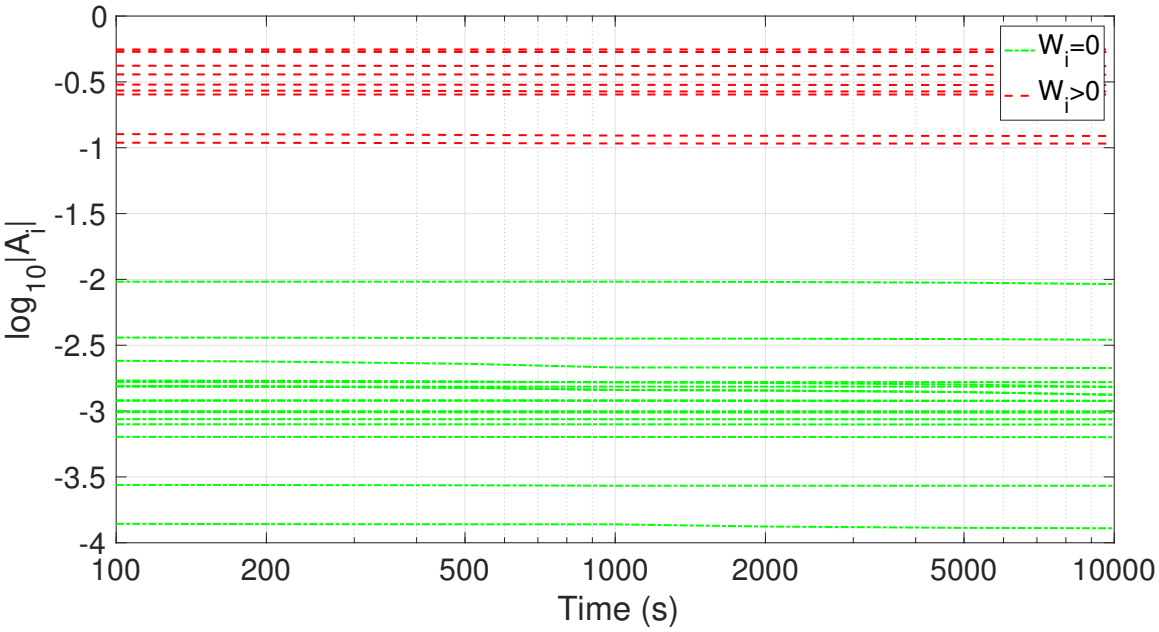

Figure 4.7: Attenuation detection metric vs. time, vertical wall, source located at $(40, 40, 0)$.

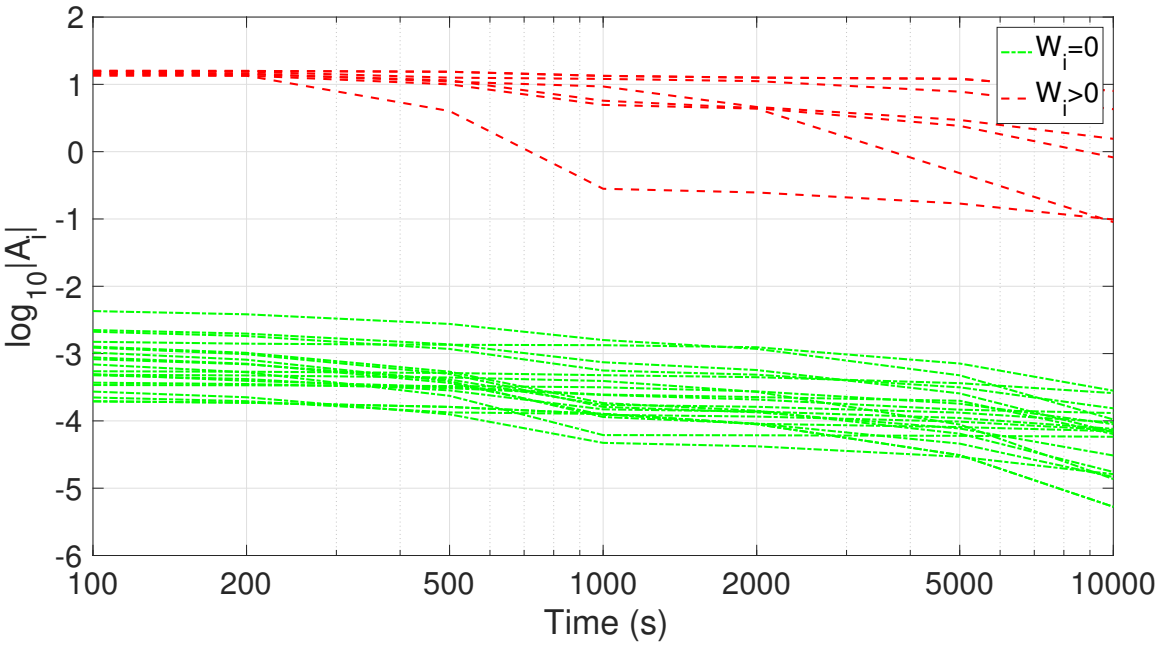

Figure 4.8: Attenuation detection metric vs. time, vertical wall, source located at $(40, 40, 40)$.

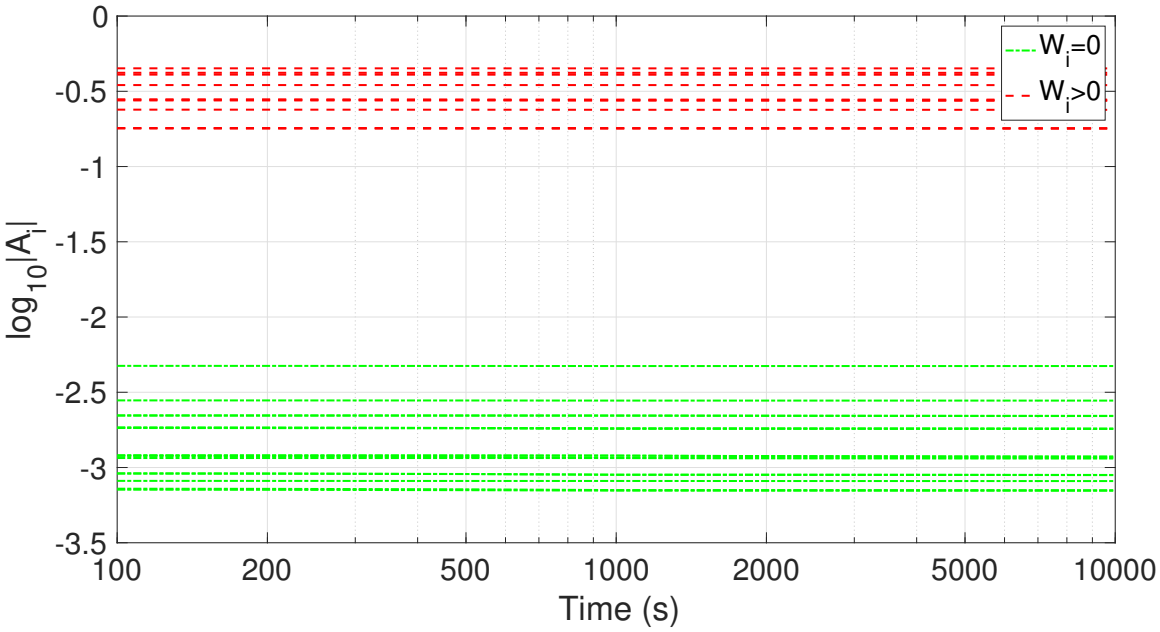

Figure 4.9: Attenuation detection metric vs. time, vertical wall, source located at $(0, 10, 0)$.

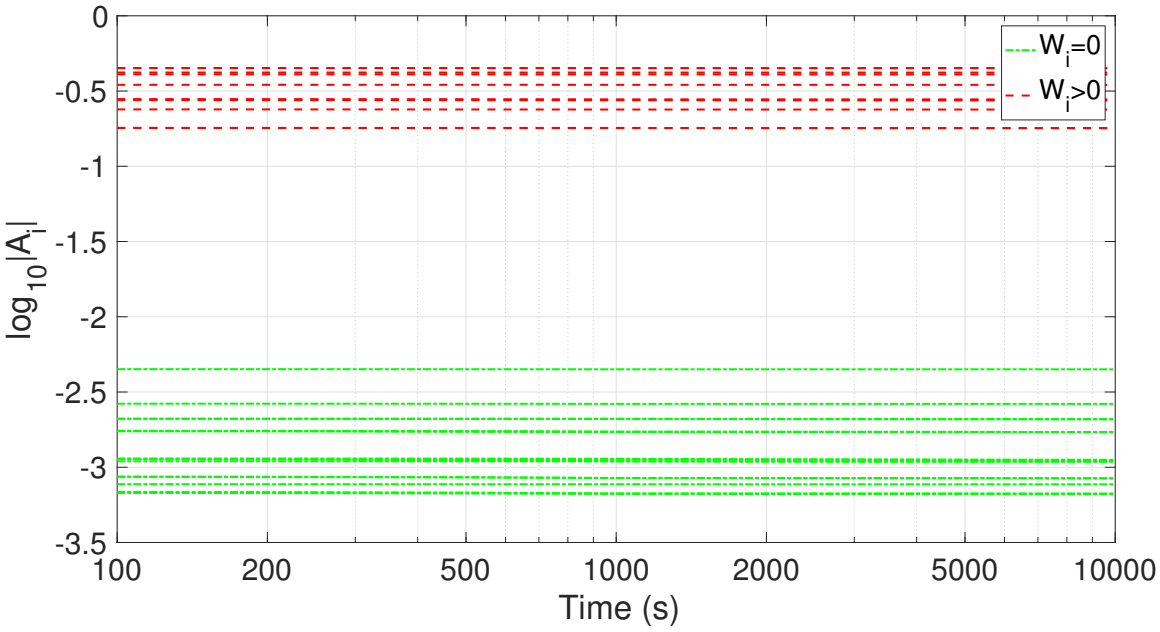

Figure 4.10: Attenuation detection metric vs. time, vertical wall, source located at $(0, 40, 0)$.

It is also important to determine the behavior of these statistics when non-homogeneous attenuation is absent. These results are presented in Figures 4.11 to 4.14 for the same four source locations presented in the earlier figures.

Taken together, the figures show that not only does the A_i metric distinguish between the presence and absence of non-homogeneous attenuation on the structure as a whole, but it also successfully identifies which particular sensor locations are affected by non-homogeneous attenuation. In general, the larger the value of A_i, the higher the degree of non-homogeneous attenuation. Thus, one way to use the A_i metrics to detect whether non-homogeneous attenuation is present or absent is to compare the maximum value of A_1 through A_L to a fixed threshold and declare that non-homogeneous attenuation is present if this threshold is exceeded.

In most cases, the A_i metrics do not vary significantly beyond the minimum total time of 100 seconds, but the results of Fig. 4.8 are an exception. The reason in this case is that attenuation only affects six of the 25 sensor locations, fewer than in every other case in which non-homogeneous attenuation is present. As a result, the source location estimation algorithm is able to continue to improve accuracy over time as more data is collected, whereas this is not the case when more sensor locations are affected by non-homogeneous attenuation. The results of Figures 4.11 to 4.14 (in which non-homogeneous attenuation is absent) also demonstrate the improvement over time as expected.

4.4 Conclusion

In this chapter, it has been verified through simulations that the error detection metric is quite sensitive to the presence or absence of non-homogeneous attenuation. Therefore, by comparing this metric to a fixed threshold, it can be determined whether

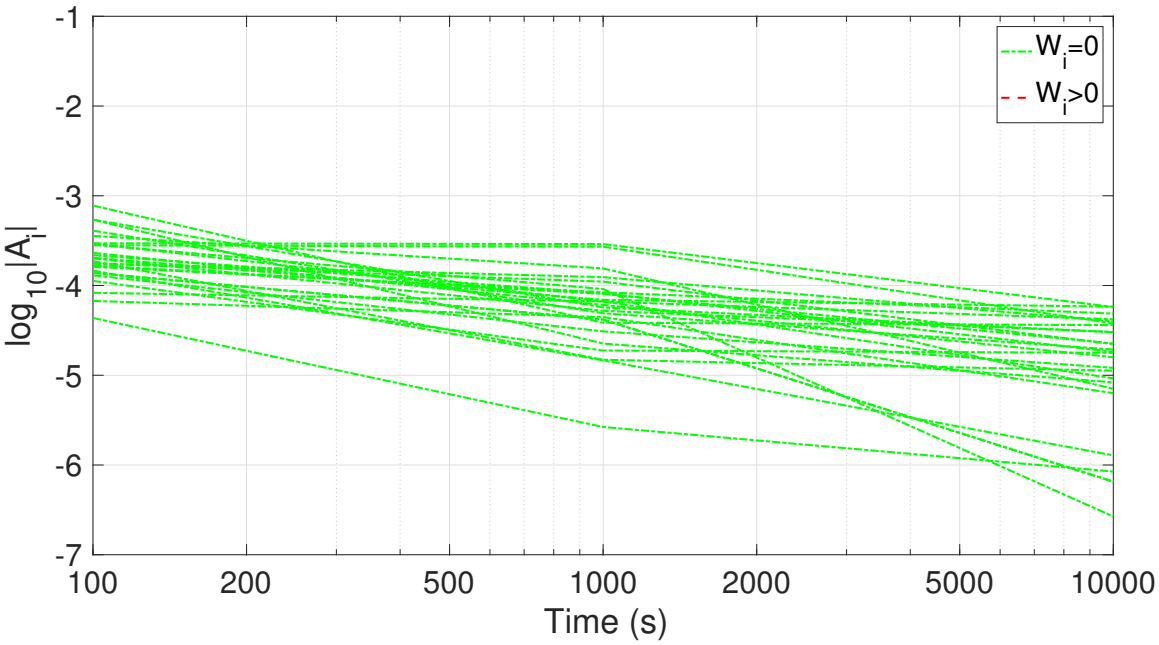

Figure 4.11: Attenuation detection metric vs. time, no attenuation, source located at $(0, 10, 0)$.

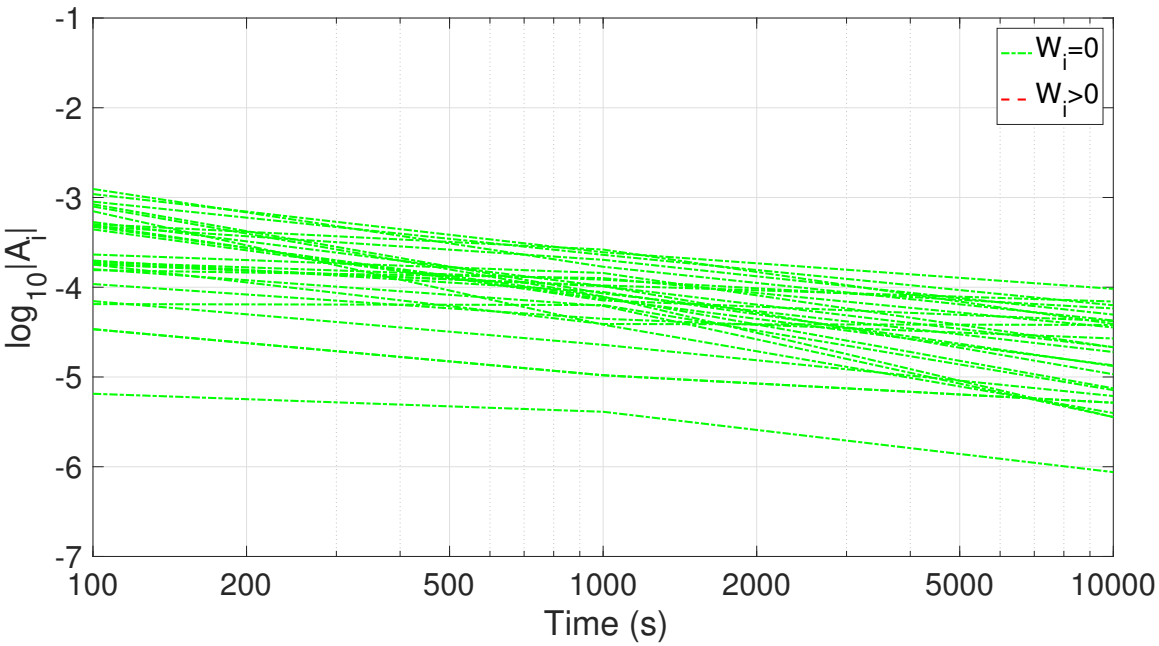

Figure 4.12: Attenuation detection metric vs. time, no attenuation, source located at $(0, 40, 0)$.

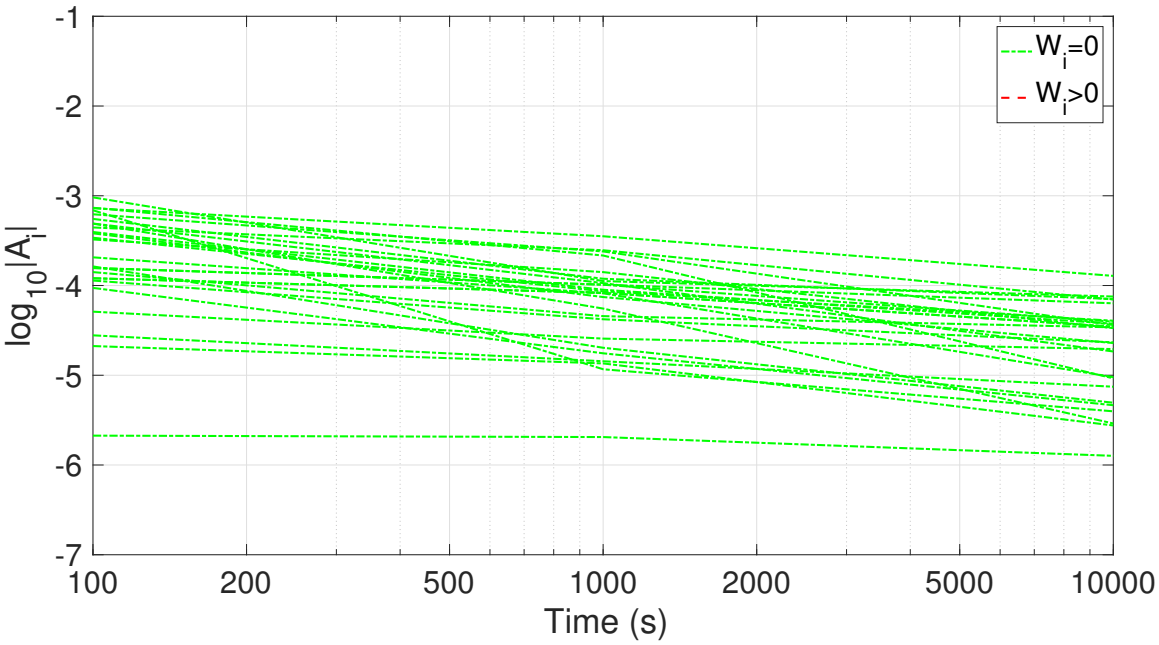

Figure 4.13: Attenuation detection metric vs. time, no attenuation, source located at $(40, 40, 0)$.

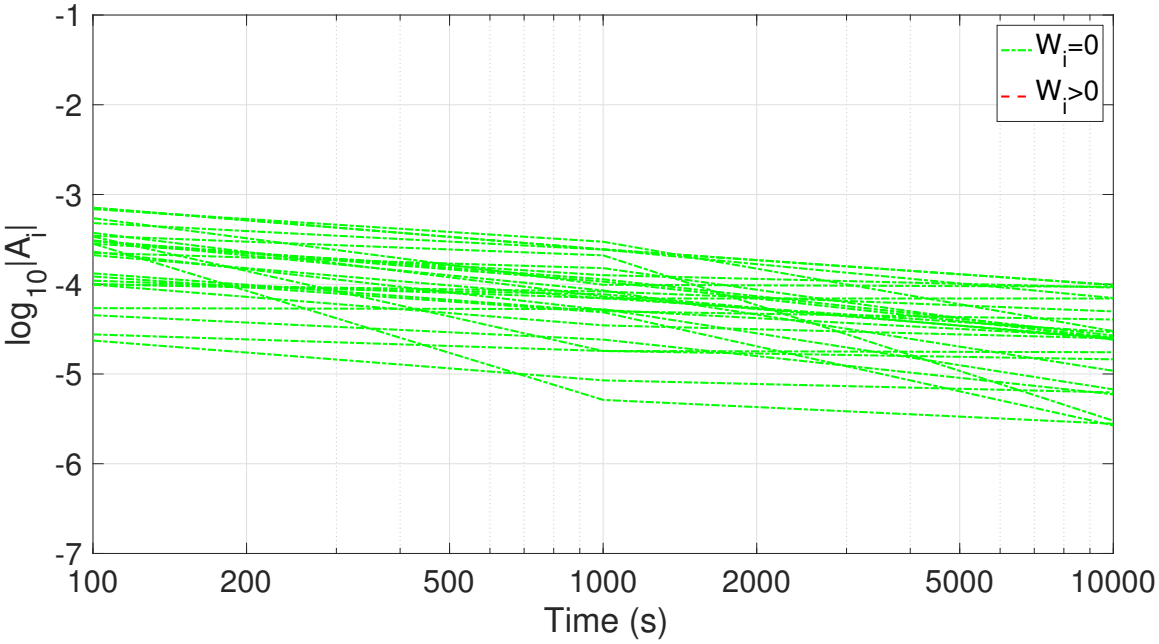

Figure 4.14: Attenuation detection metric vs time, no attenuation, source located at $(40, 40, 40)$.

the counts at a particular location have been significantly affected by non-homogeneous attenuation. This metric also indicates which particular sensor locations experience non-homogeneous attenuation, a fact that is exploited in the following chapter.

Chapter 5

Estimation in the Presence of Non-homogeneous Attenuation

In Chapter 4, it has been shown that the use of the metrics A_i enables the detection of situations in which significant non-homogeneous attenuation is present due to absorbing materials such as high-density concrete. It has furthermore been shown that the metrics enable the ability to determine which particular sensor locations experience attenuation. In this chapter, these properties of the metrics are exploited to effectively estimate the source location despite the presence of non-homogeneous attenuation. The essence of the approach is to use the A_is to identify attenuated sensor locations and estimate their attenuation levels in addition to the unknown source location and strength.

5.1 System Model

The updated system model presented in Section 4.1 is used again, this time for estimation in the presence of unknown attenuation. As before, the thickness of the

attenuating material W_i is considered to be relatively small compared to the distance D_i between the source and the ith sensor location. Therefore, the count arrival rate at the ith location is again given by

$$\lambda_i \approx \lambda_B + \frac{\mu A I}{4\pi D_i^2} e^{-\rho_g D_i - \rho_s W_i} \tag{5.1}$$

where ρ_s is the mass attenuation coefficient of solid material, ρ_g is the mass attenuation coefficient of air, I is the intensity of the source, μ is the photo-peak efficiency (including the branching factors for the photo-peaks of the different isotopes), A is the cross-sectional area of the detector, and λ_B is the background radiation level.

5.2 Attenuation Estimation

The position of the ith data collection location is (x_i, y_i, z_i), and (x_0, y_0, z_0) is the location of the source; it follows that $D_i = \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2}$. The observed nuclear decay is again modeled as a Poisson process [14]. Let L denote the number of locations visited by N physical sensors, let T_i denote the amount of time over which the data has been collected at the ith location, and let λ_i be the count arrival rate. Then, the number of counts received over duration T_i at this location is Poisson $(\lambda_i T_i)$, and these counts are mutually independent. Denoting the count at the ith location as K_i, the probability of measuring k_i counts at this location is given by

$$P(K_i = k_i; x_0, y_0, z_0, I, W_1, \ldots, W_L) = \frac{(\lambda_i T_i)^{k_i}\, e^{-(\lambda_i T_i)}}{k_i!}, \quad k_i = 0,1,\ldots \tag{5.2}$$

Furthermore, by independence the joint mass function is the product of the individual mass functions of each sensor; that is

$$P(\mathbf{K} = \mathbf{k};\ x_0, y_0, z_0, I, W_1, \ldots, W_L) = \prod_{i=1}^{L} P(K_i = k_i;\ x_0, y_0, z_0, I) \tag{5.3}$$

A multi-stage algorithm based on ML estimation is proposed for estimating the location of the nuclear source. In the first stage, ML estimation is employed assuming that the W_is in Eq. (5.1) are all equal to zero; that is, it is assumed that non-homogeneous attenuation is absent. Denote the resulting estimates of the source location and intensity as $\hat{x}_0$, $\hat{y}_0$, $\hat{z}_0$, and $\hat{I}$. Furthermore, define $\hat{D}_i^2 = (x_i - \hat{x}_0)^2 + (y_i - \hat{y}_0)^2 + (z_i - \hat{z}_0)^2$ and $\hat{V} = \log(\mu A \hat{I})$. From these definitions, the estimated arrival rate at the ith location can be estimated as $\hat{\lambda}_i = \lambda_B + \frac{1}{4\pi \hat{D}_i^2} e^{\hat{V} - \rho_g \hat{D}_i}$. We can use the error detection metric from Chapter 4 given by $A_i = \frac{\frac{k_i}{T_i} - \hat{\lambda}_i}{\hat{\lambda}_i}$ in order to detect which sensor locations are likely to be affected by non-homogeneous attenuation.

Defining τ as a threshold used by the algorithm, if for a given sensor location $\log_{10} |A_i| > \tau$, it is concluded that the counts at that location have undergone non-negligible non-homogeneous attenuation. The specific value of threshold that is most appropriate for a given attenuating structure will depend on the structure itself. One approach is to consider multiple thresholds and compare the $\log_{10} |A_i|$ values to each threshold. That is, n_τ iterations of the non-homogeneous attenuation detection process are used, and for the jth iteration, significant non-homogeneous attenuation is declared present at the ith sensor location if $\log_{10} |A_i| > \tau_j$ and declared absent otherwise.

For the sensor locations declared to be affected by non-homogeneous attenuation, the W_is are treated as unknowns, in addition to x_0, y_0, z_0, and I. ML estimation can be used to estimate the unknown W_is along with source location and strength.

Using these estimates, a new count arrival rate $\hat{\lambda}_i$ is calculated to obtain new A_is. Typically, the new A_is will have lower values than they did previously, and as a result, fewer $\log_{10}|A_i|$ values will be above the threshold τ_j. ML estimation is performed again, using the previously estimated W_is that correspond to $\log_{10}|A_i| < \tau_j$ and re-estimating the W_is that are such that $\log_{10}|A_i| > \tau_j$ along with re-estimating x_0, y_0, z_0, and I. This process is repeated until either $\log_{10}|A_i| < \tau_j$ for all i or a maximum number of iterations c is reached.

If all L sensor locations were such that $\log_{10}|A_i|$ exceeded the threshold τ_j, there would be more unknowns than sensor counts, and the ML algorithm would be indeterminate. Thus no more than a maximum number M of W_is should be estimated. This maximum number should be such that the total number of unknowns is less than L. For example, with $L = 25$ sensor locations, if $M = 15$ is used, the total number of unknowns cannot exceed $M + 4 = 19$, which lies significantly below L. In theory, M could be as large as 21 for $L = 25$ locations, but the ML algorithm tends to perform poorly when the number of unknowns is nearly as large as L.

In the event that more than M A_is are such that $\log_{10}|A_i| > \tau_j$, the W_is corresponding to the M largest values of $\log_{10}|A_i|$ are estimated. In later iterations, as $\log_{10}|A_i|$ values drop, more of the W_is that were set aside can be estimated.

This entire procedure is repeated for each threshold τ_j to obtain n_τ different estimates of $\hat{x}_0$, $\hat{y}_0$, $\hat{z}_0$, and $\hat{I}$. The joint mass function in Eq. (5.3) is then calculated for each of the n_τ sets of estimates, and the estimates that maximize the joint mass function are chosen to be the final result. The algorithm is summarized in Algorithm 1 and Fig. 5.1. In this algorithm and flowchart, we indicate which W_is are to be estimated via ML estimation by setting them equal to negative one.

Algorithm 1 Estimation Procedure for Non-Homogeneous Attenuation

1: Collect data k_is for L sensor locations
2: **for** $j = 1, \ldots, n_\tau$ **do**
3: Set all W_is to 0
4: **for** $n = 1, \ldots, c$ **do**
5: Estimate $\hat{x}_0$, $\hat{y}_0$, $\hat{z}_0$, $\hat{I}$, and $\hat{W}_i$s that were -1 via ML estimation
6: Calculate $\hat{D}_i$s, $\hat{\lambda}_i$s, and A_is from ML estimates
7: **if** all A_is satisfy $\log_{10}|\mathrm{A_i}| < \tau_j$ **then** go to Step 14
8: **if** at most M A_is satisfy $\log_{10}|\mathrm{A_i}| > \tau_j$ **then**
9: Set W_is that correspond to $\log_{10}|\mathrm{A_i}| > \tau_j$ to -1
10: **else**
11: Set W_is that correspond to M largest $\log_{10}|\mathrm{A_i}|$ to -1
12: **endif**
13: **endfor**
14: Calculate $P_j = P(\mathbf{K} = \mathbf{k}; \hat{x}_0, \hat{y}_0, \hat{z}_0, \hat{I}, \hat{W}_1, \ldots, \hat{W}_L)$
15: Set $\hat{x}_{0j} = \hat{x}_0$, $\hat{y}_{0j} = \hat{y}_0$, $\hat{z}_{0j} = \hat{z}_0$, and $\hat{I}_j = \hat{I}$
16: **endfor**
17: Set $\hat{x}_0 = \hat{x}_{0j}$, $\hat{y}_0 = \hat{y}_{0j}$, $\hat{z}_0 = \hat{z}_{0j}$ and $\hat{I} = \hat{I}_j$ that correspond to the largest P_j

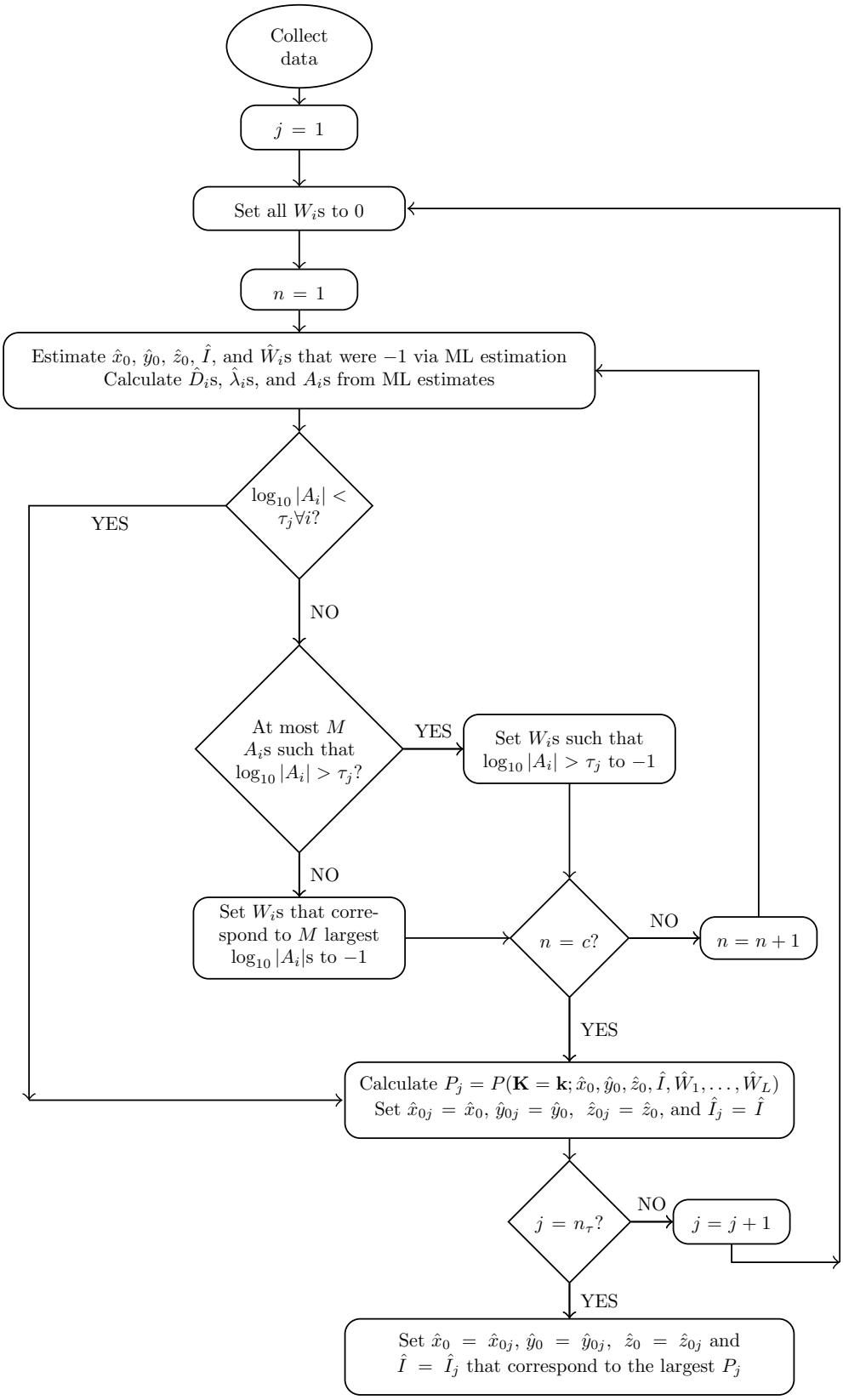

Figure 5.1: Estimation algorithm incorporating non-homogeneous attenuation.

5.3 Performance Results

A total of five different attenuating structures are considered, the shaft and the wall from Chapter 4, as shown in Figs. 5.2 and 5.3, and three new structures: two opposite attenuating walls, as shown in Fig. 5.4, two adjacent attenuating walls, as shown in Fig. 5.5 and three attenuating walls, as shown in Fig. 5.6.

As before, a set of 125 source locations uniformly distributed across the search space are used to calculate the maximum and average errors. Figs. 5.7 and 5.8 present results for the shaft, Figs. 5.9 and 5.10 show results for a single wall, Figs. 5.11 and 5.12 present results for two opposite walls, two adjacent walls are considered in Figs. 5.13 and 5.14, and results for three walls are shown in Figs. 5.15 and 5.16. Additionally, Figs. 5.17 and 5.18 present results when non-homogeneous attenuation is absent.

For each figure, six cases are shown. Three of the cases use $n_\tau = 21$ threshold values ranging from $\tau_1 = -2$ to $\tau_2 1 = 0$, spaced at multiples of 0.1. These results are shown as solid horizontal lines on the figures. The other three cases, shown as dashed lines, use $n_\tau = 1$, a single fixed threshold. The value of this threshold varies in the figures so that the results are not straight lines. Within each group of three plots, the values of c are $c = 2$, $c = 15$, and $c = 30$. In every case, there are five physical sensors moving to 25 locations, and the formulas for travel time and total time in Chapter 3 are used here as well. For the figures, the total time is 500 seconds, $V = 18$ ln(Bq), and $\lambda_B = 10^3$ cps. All walls, including the walls of the shaft, are assumed to have thickness $\Delta = 0.25$ m.

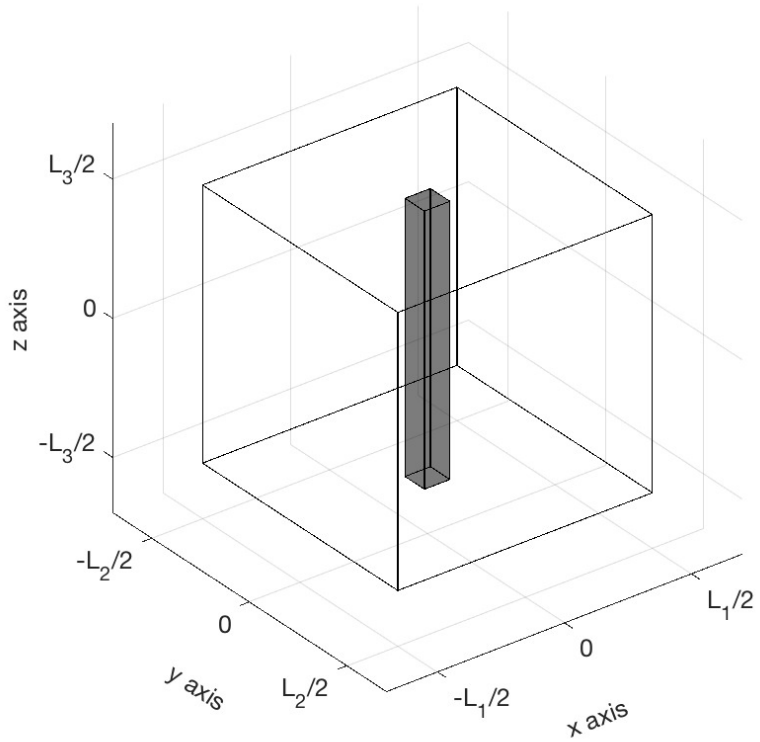

Figure 5.2: Vertical attenuating shaft in the center of the building.

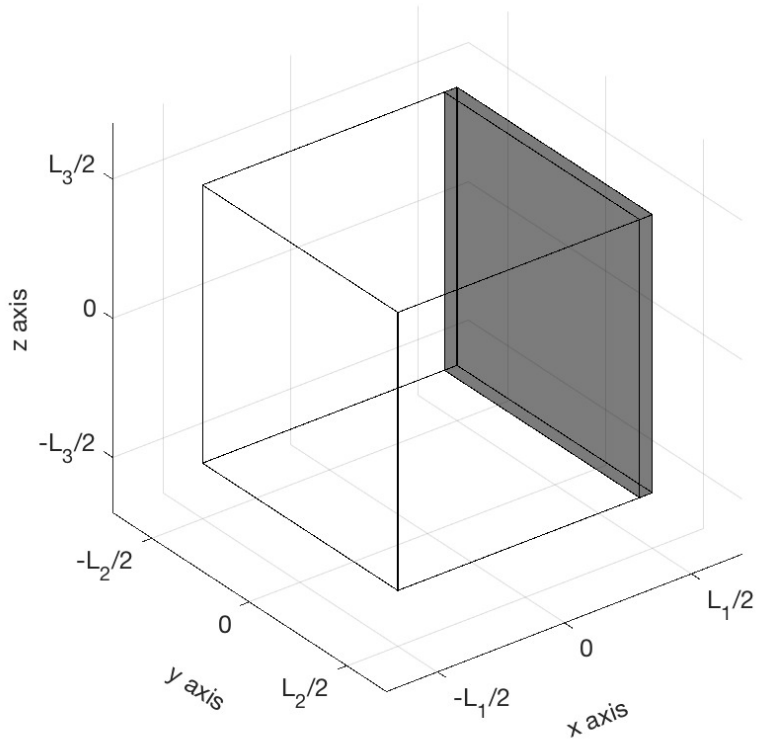

Figure 5.3: Single attenuating wall along a face of the building.

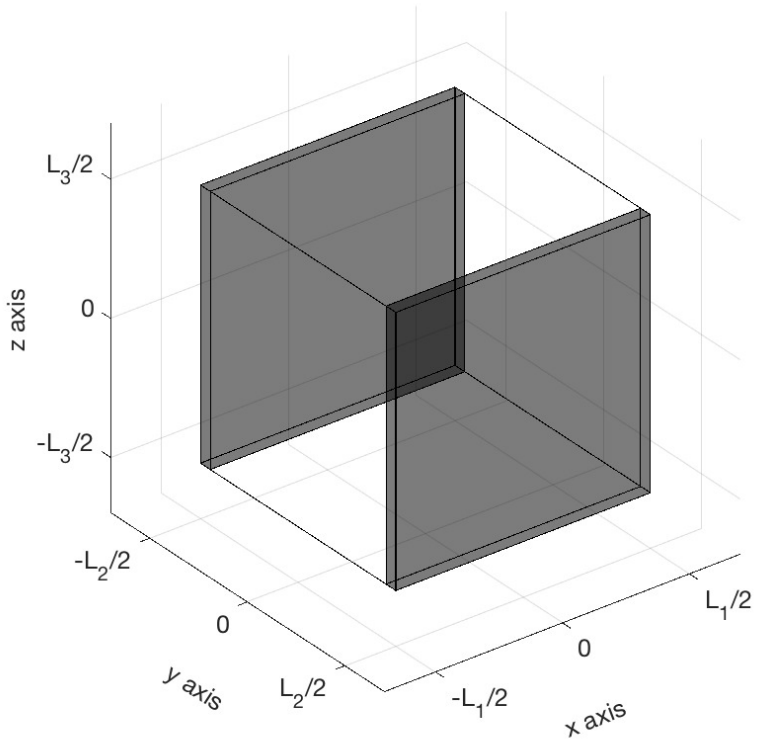

Figure 5.4: Two attenuating walls along faces of the building (opposite).

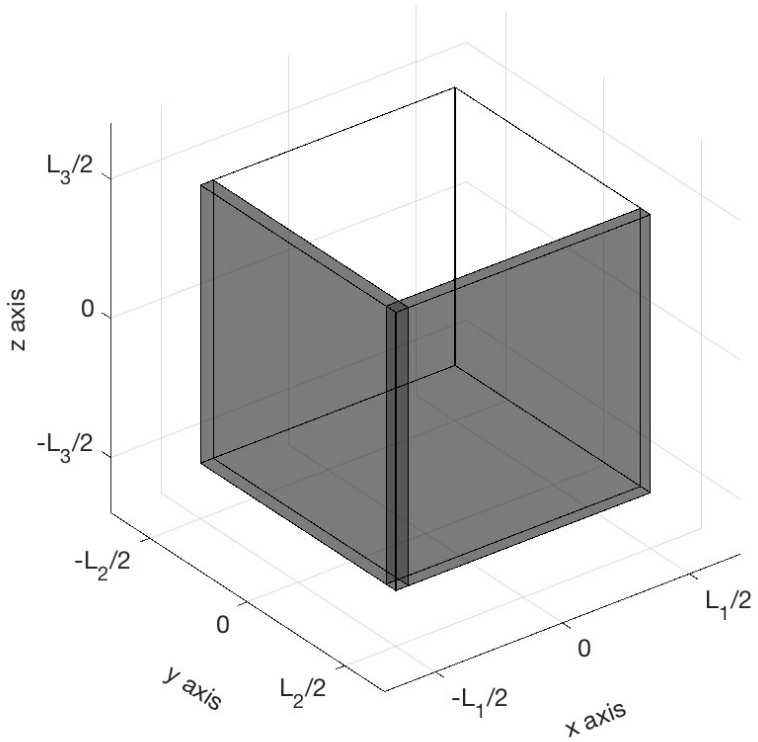

Figure 5.5: Two attenuating walls along faces of the building (adjacent).

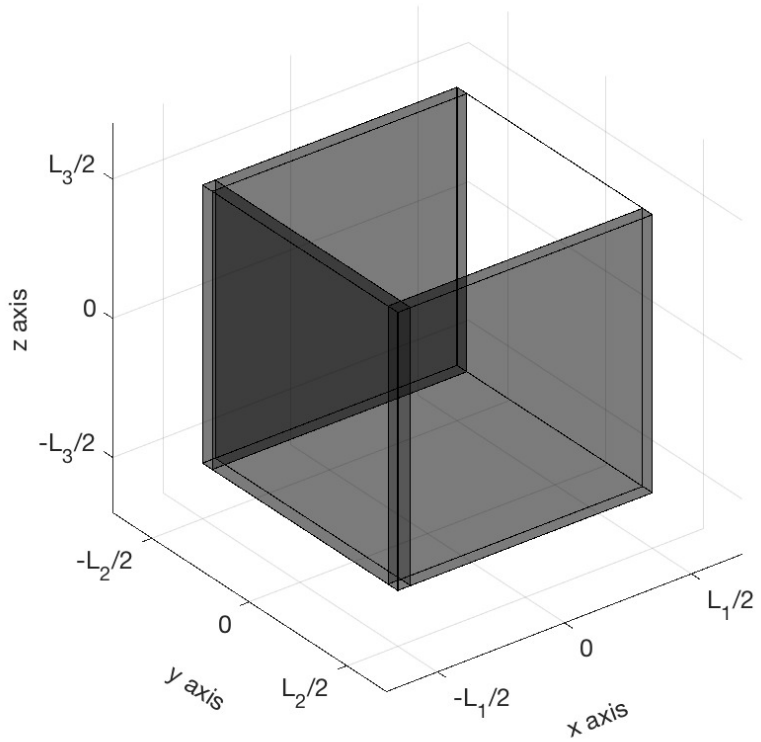

Figure 5.6: Three attenuating walls along faces of the building.

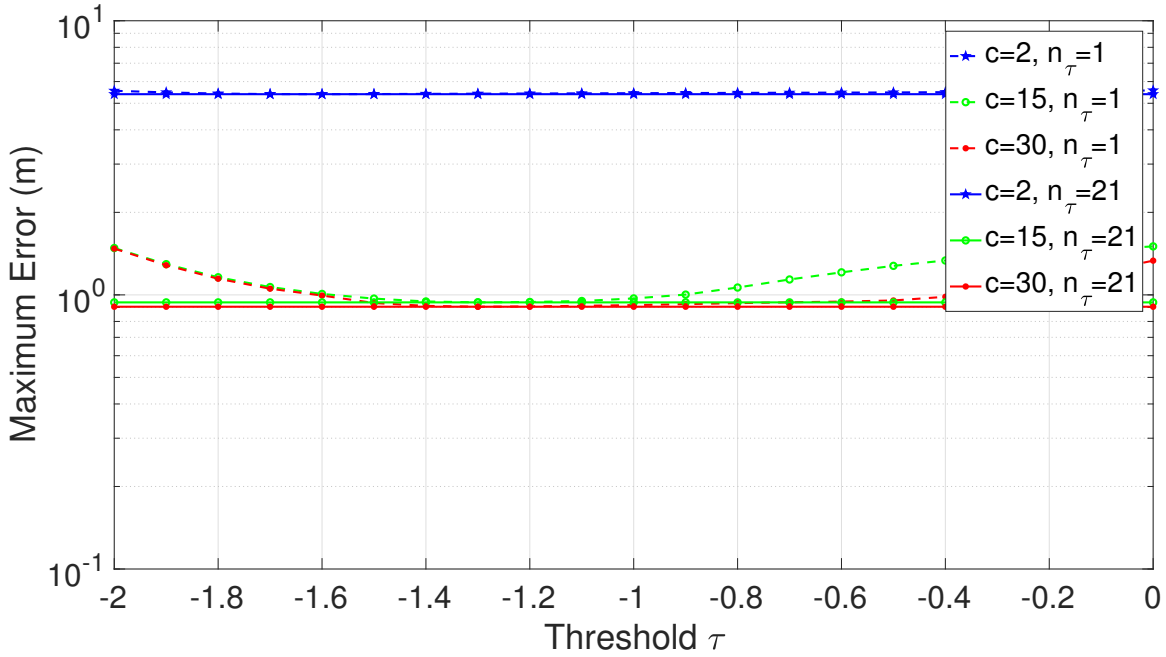

Figure 5.7: Maximum error of attenuation estimation algorithm, central attenuating shaft.

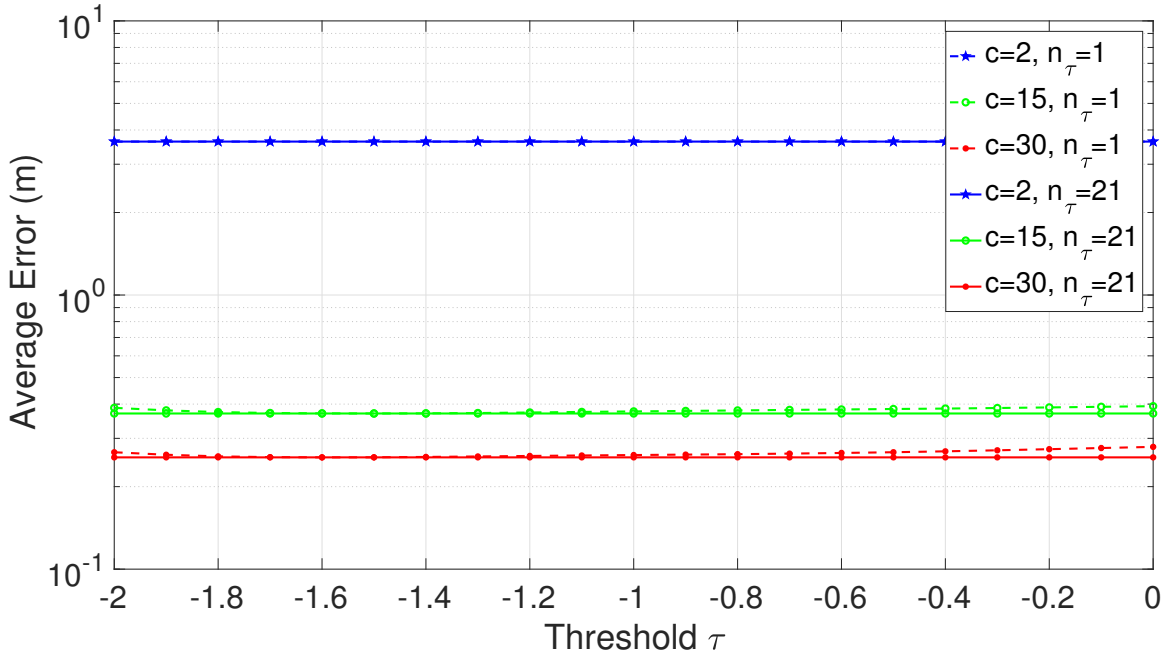

Figure 5.8: Average error of attenuation estimation algorithm, central attenuating shaft.

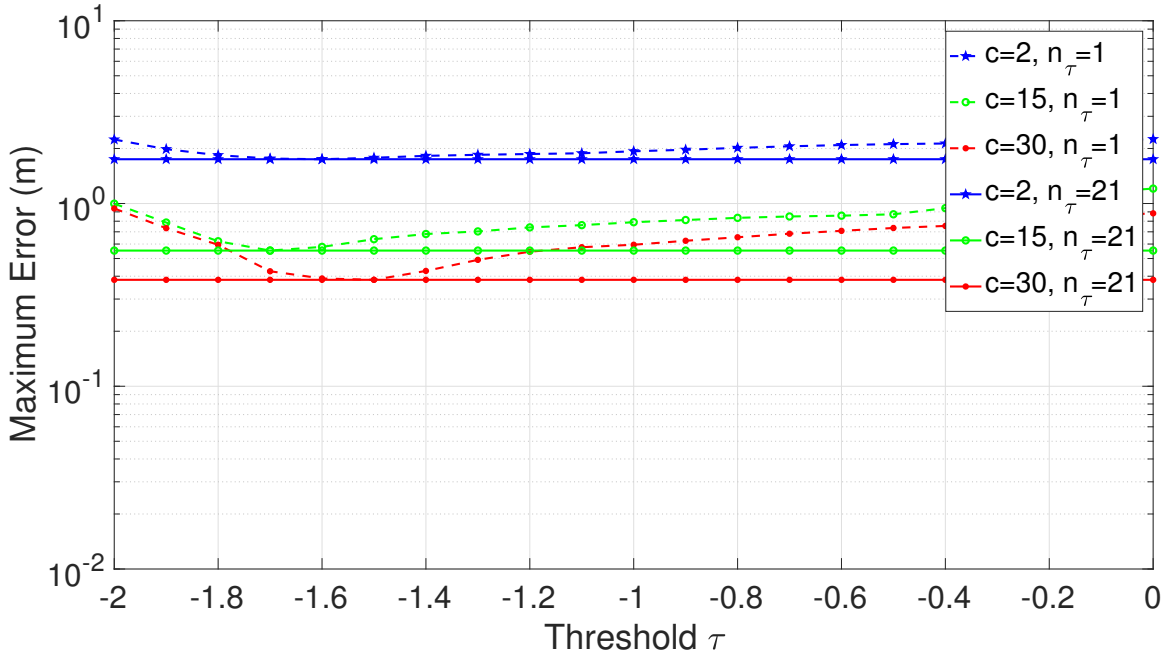

Figure 5.9: Maximum error of attenuation estimation algorithm, single vertical attenuating wall.

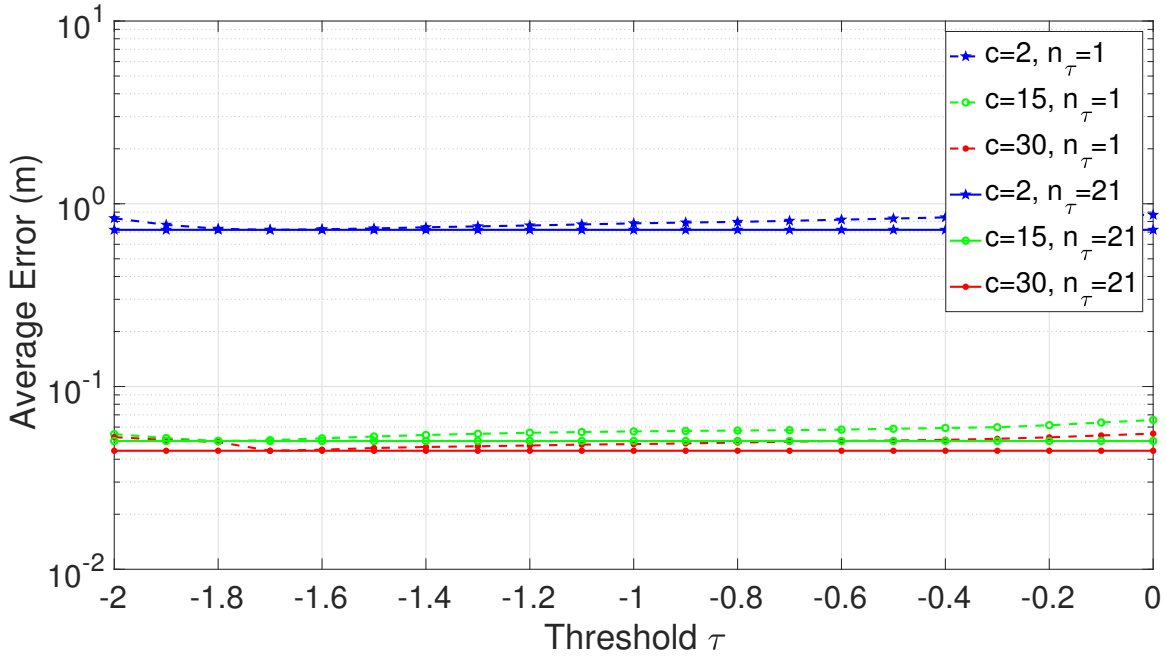

Figure 5.10: Average error of attenuation estimation algorithm, single vertical attenuating wall.

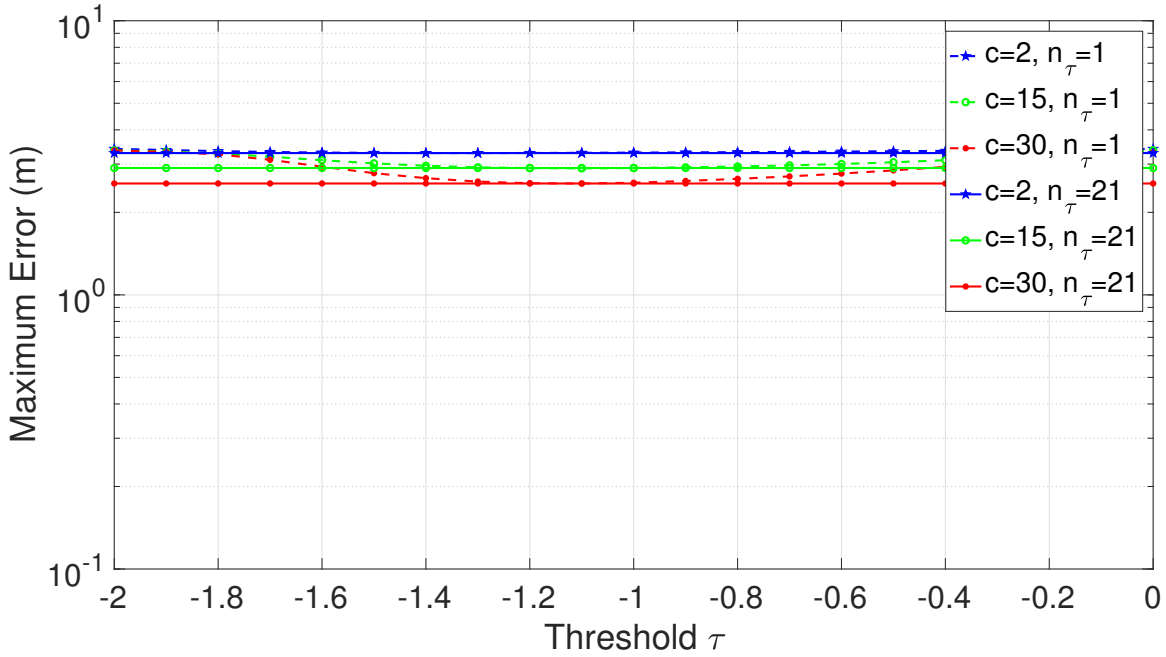

Figure 5.11: Maximum error of attenuation estimation algorithm, two attenuating walls (opposite).

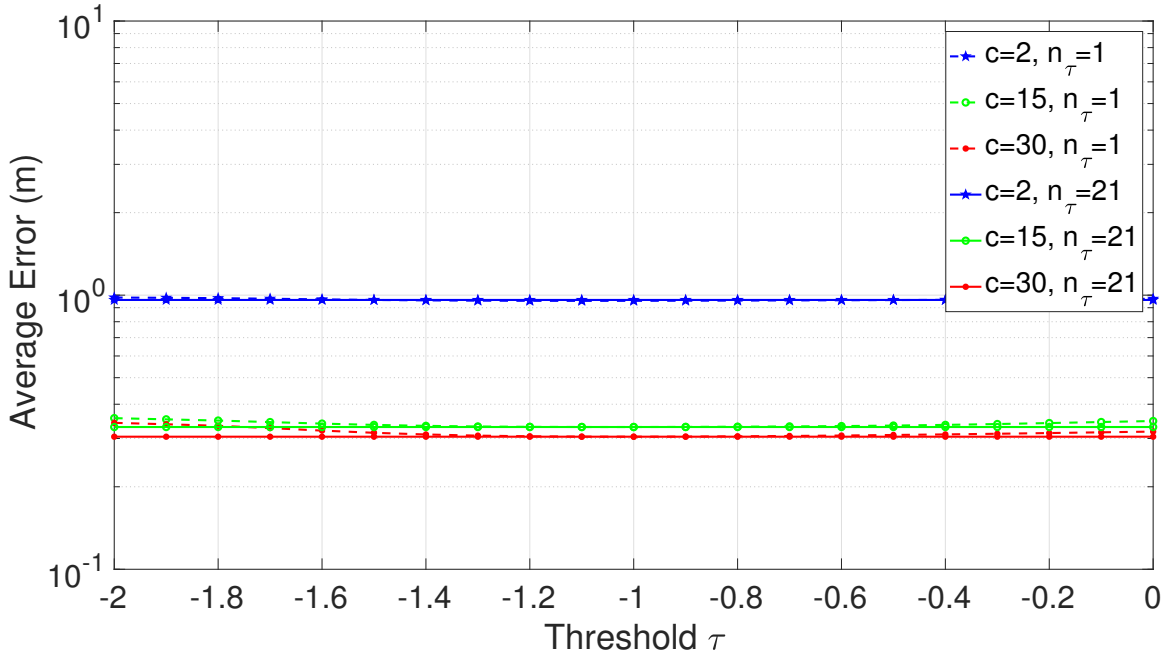

Figure 5.12: Average error of attenuation estimation algorithm, two attenuating walls (opposite).

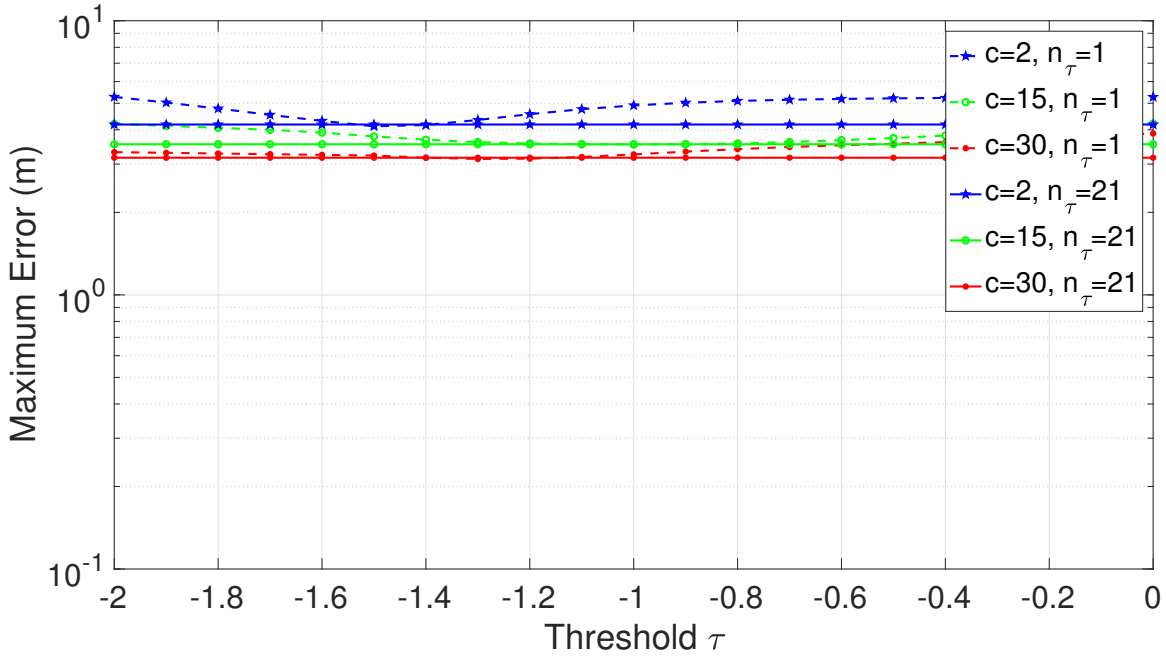

Figure 5.13: Maximum error of attenuation estimation algorithm, two attenuating walls (adjacent).

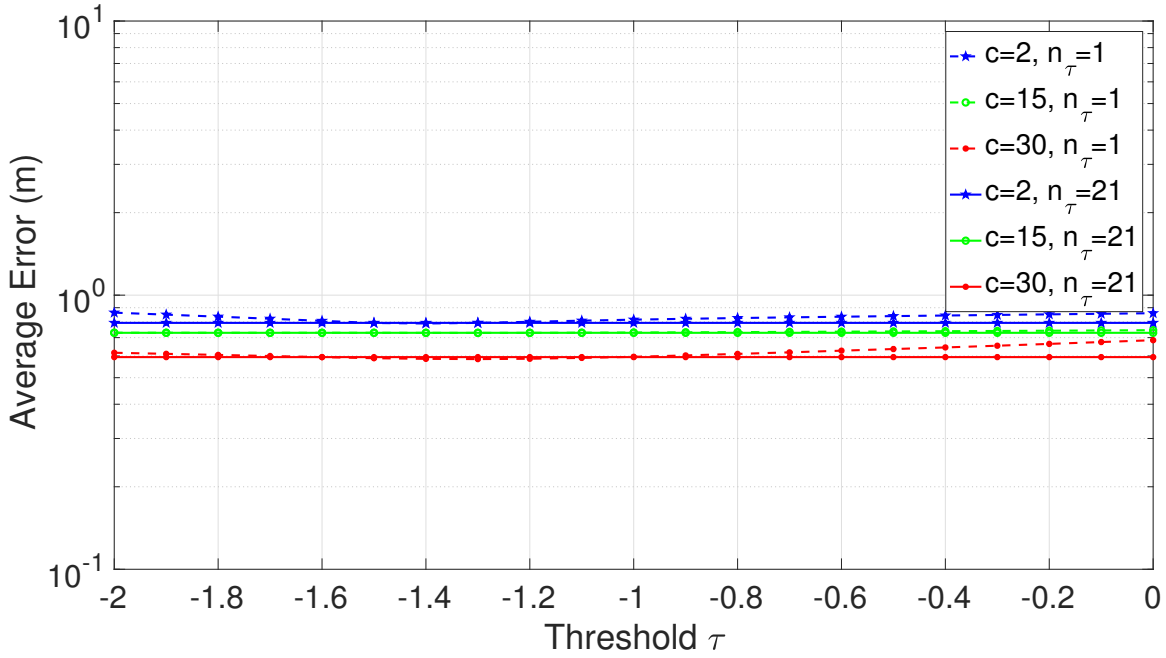

Figure 5.14: Average error of attenuation estimation algorithm, two attenuating walls (adjacent).

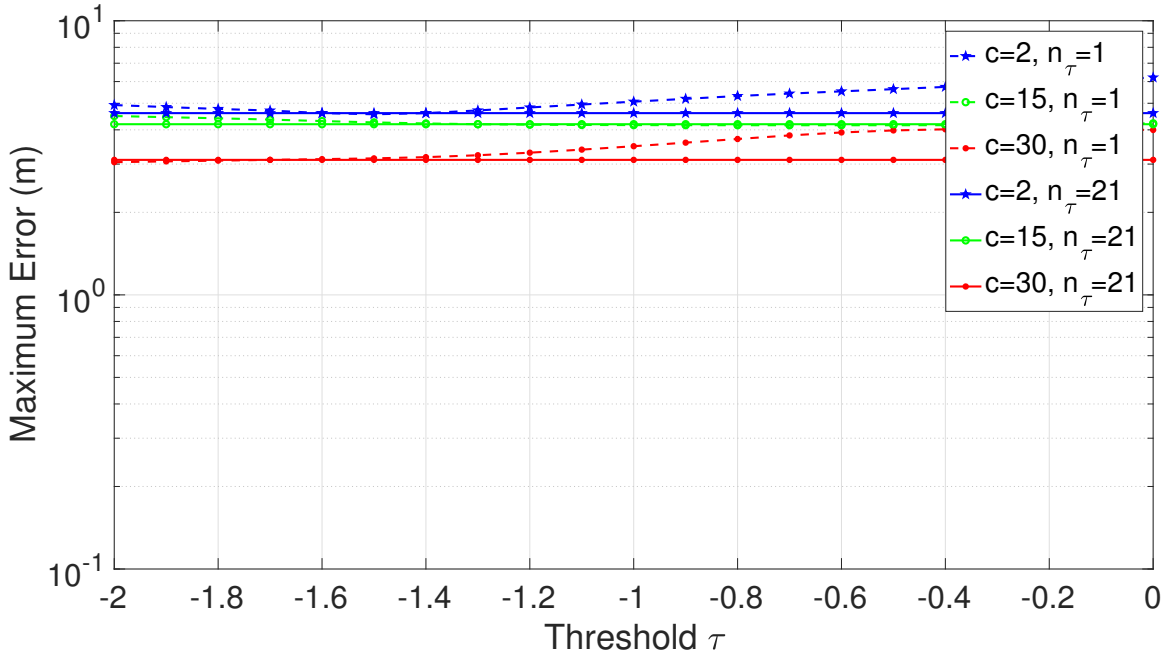

Figure 5.15: Maximum error of attenuation estimation algorithm, three attenuating walls.

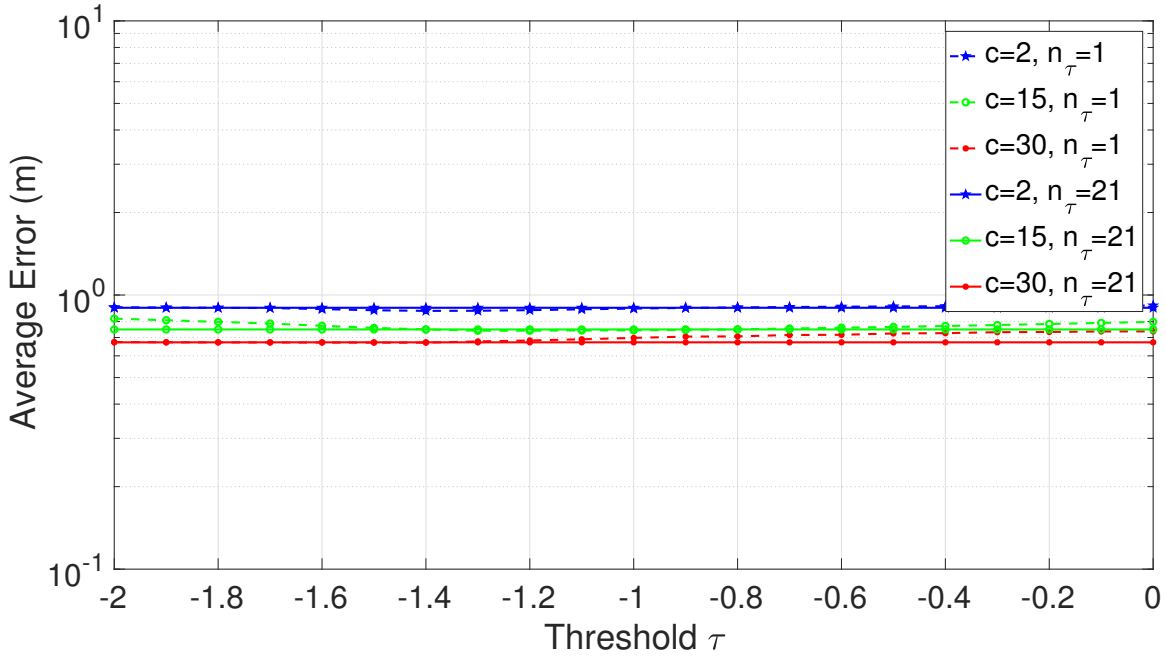

Figure 5.16: Average error of attenuation estimation algorithm, three attenuating walls.

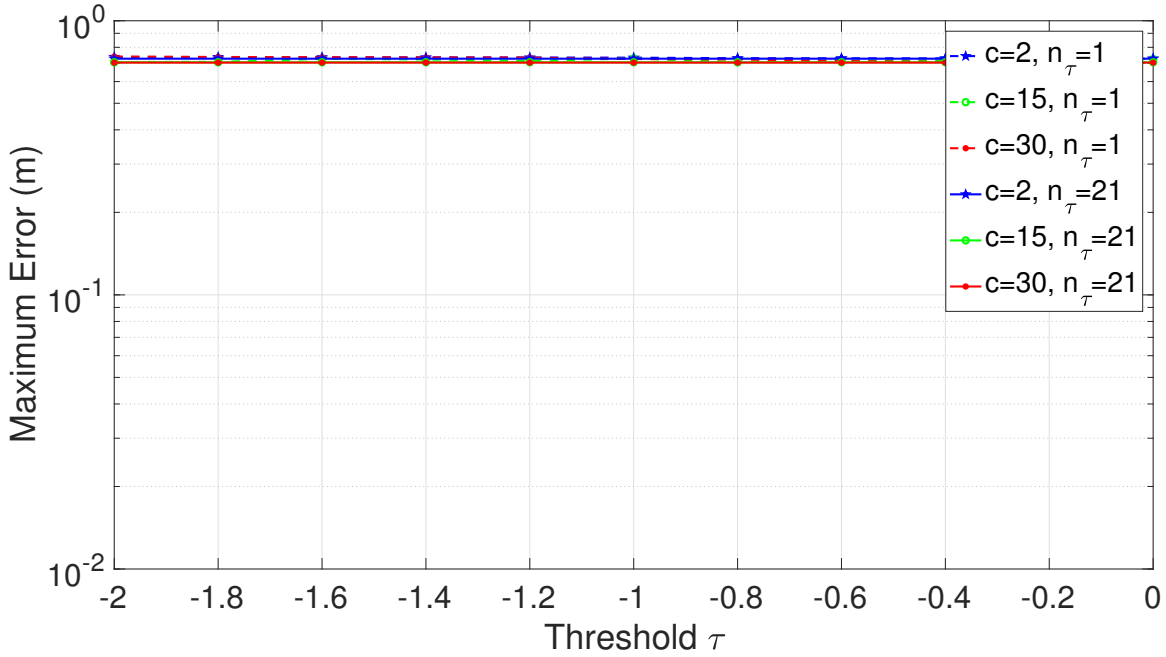

Figure 5.17: Maximum error of attenuation estimation algorithm, no non-homogeneous attenuation.

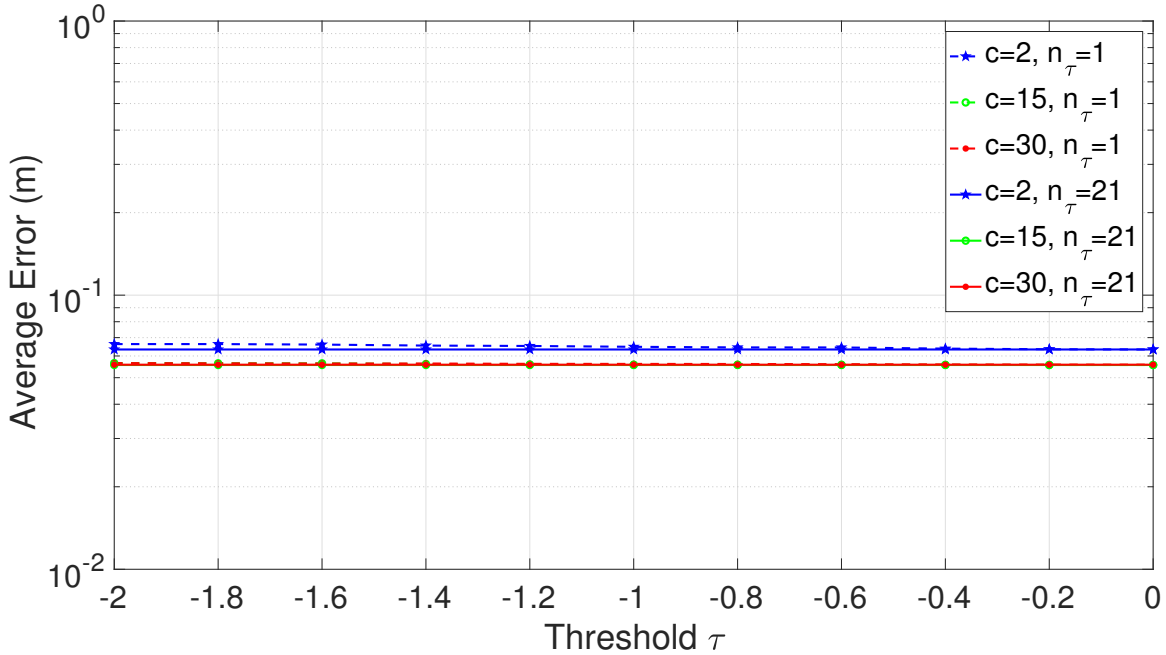

Figure 5.18: Average error of attenuation estimation algorithm, no non-homogeneous attenuation.

An examination of the various dashed lines in the figures show that there is not one optimal threshold for all the structures, and thus it is of value to use $n_\tau > 1$. An examination of the various solid lines shows that the final threshold selection method of the algorithm is effective at finding the location estimate that corresponds to the best threshold. It also shows that using $c = 15$ iterations is much better than using $c = 2$ iterations, but only negligibly worse than $c = 30$. Note that the algorithm requires a total of $n_\tau(c + 1)$ uses of the ML algorithm, and thus neither n_τ nor c should be larger than necessary.

Figs. 5.19 and 5.20 compare the performance of the algorithm (with $c = 15$ and $n_\tau = 21$) for the various attenuating structures. These plots show the maximum and average errors as a function of total time. It is evident that the estimation errors are lowest when the attenuating structure is a single wall along a face of the building, and the worst performance results when attenuating structure is three faces. The shaft performance is worse than the single wall but better than the two walls. To a degree, the different slopes of the plots indicates the degree to which attenuation is estimated accurately.

The figures also include the performance of the algorithm when no attenuating structures are present. Surprisingly, the results are worse than that of a single attenuating wall. This performance is due to the fact that the algorithm is specifically designed for attenuating structures. An enhancement to this algorithm that incorporates both scenarios is considered in the following section.

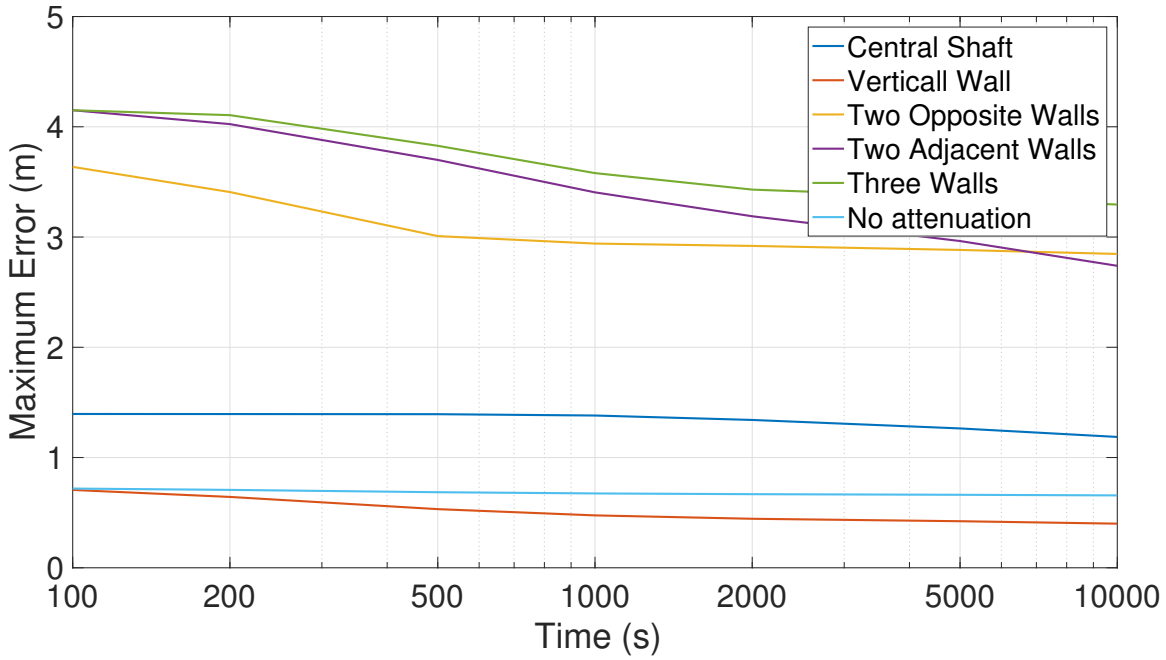

Figure 5.19: Maximum error vs. time for attenuation estimation algorithm for different attenuating structures.

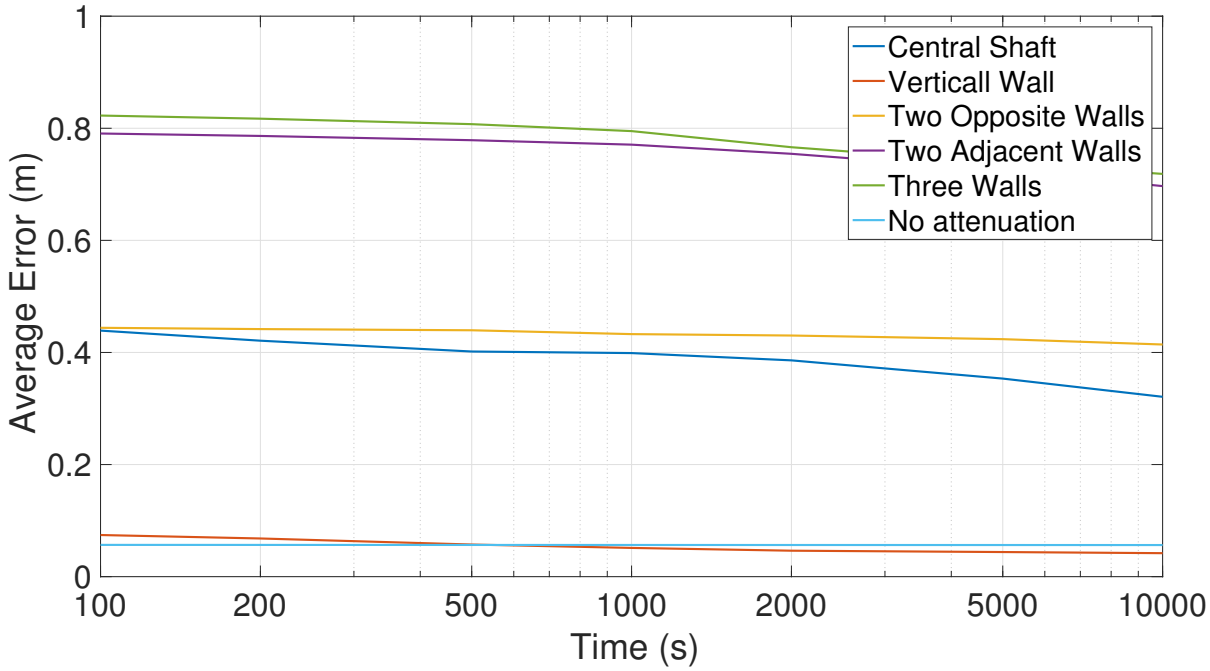

Figure 5.20: Average error vs. time for attenuation estimation algorithm for different attenuating structures.

5.4 Hybrid Algorithm

In Chapter 3, it has been shown that moving sensors adaptively based on initial estimates of source location dramatically improves the performance of location estimation when non-homogeneous attenuation is absent. In Section 5.3 it has been shown that non-homogeneous attenuation can be estimated along with unknown source location and intensity. In this section, we consider whether the strengths of these two approaches can be combined.

A key requirement for the adaptive algorithm to perform well is that the first stage estimates are sufficiently accurate so as to identify the correct sub-cube in which the source is located. If the initial stage fails in this task, then the sensors may be moved to very inappropriate locations resulting in performance that is worse than if they had not moved at all. Unfortunately, the presence of significant non-homogeneous attenuation makes accurate initial-stage estimation difficult if not impossible. Furthermore, locations that were good to move to in the absence of non-homogeneous attenuation may become quite poor in the presence of non-homogeneous attenuation. However, it is easy to detect the presence of non-homogeneous attenuation even in a very short first stage.

For these reasons, a hybrid method is presented that combines the performance of the best features of both the adaptive and the attenuation estimation algorithm. In the first stage of this hybrid method, N physical sensors collect data from the sensor locations shown in Fig. 5.21. In this figure, $N = 5$. An initial estimate $(\hat{x}_0, \hat{y}_0, \hat{z}_0, \hat{\lambda})$ is obtained for the source location.

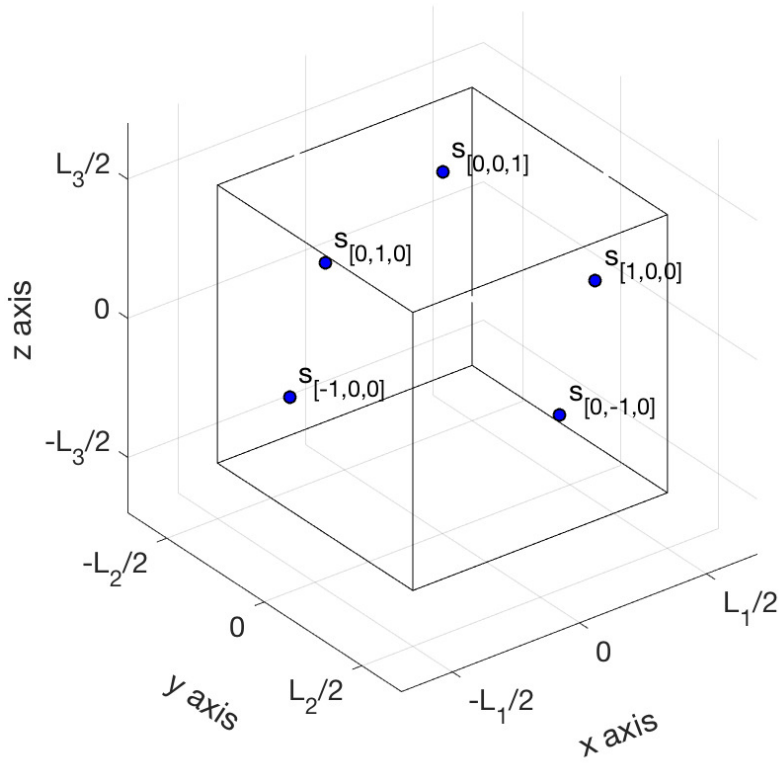

Figure 5.21: Drone locations for the first stage of the hybrid algorithm.

Using this estimate, values of the A_is are obtained using Eq. (4.4) for $i = 1, \ldots, N$. For the first stage of the hybrid algorithm, if $\log_{10} |A_i| > \tau_h$ for any i, it is concluded that sources of non-homogeneous attenuation are present within the search space. Here, τ_h is a single fixed threshold. In this case, the attenuation estimation algorithm described in Section 5.2 is used to obtain source location estimates in the presence of non-homogeneous attenuation. In contrast, if it is concluded that non-homogeneous attenuation within the search space is negligible, then the initial estimate $(\hat{x}_0, \hat{y}_0, \hat{z}_0, \hat{\lambda})$ is used to select a sub-cube from Fig. 3.3, and the algorithm from Table 3.1 is used. A block diagram of the hybrid algorithm is shown in Fig. 5.22.

For performance results, the five attenuating structures of Section 5.3 are considered along with the case that non-homogeneous attenuation is absent. These results, comparing the hybrid algorithm with the adaptive algorithm of Section 3.2 and the attenuation estimation algorithm of Section 5.2, are shown in Figs. 5.23 to 5.34. For these figures, $N = 5$, $L = 25$, $V = 18$ ln(Bq), and $\lambda_B = 1000$ cps. For the attenuation estimation algorithm, both as a standalone algorithm and as a component of the hybrid algorithm, $c = 15$ and $n_\tau = 21$, and τ ranges from -2 to 0, with a step size of 0.1. For the hybrid algorithm, $\tau_h = -1.4$.

The results in the presence of non-homogeneous attenuation (Figs. 5.23 to 5.32) show that the hybrid algorithm performs dramatically better than the adaptive algorithm and only slightly worse than the attenuation estimation algorithm. The gap between the hybrid algorithm and the attenuation estimation algorithm tends to be the largest in situations in which symmetry causes the hybrid algorithm to more frequently decide that non-homogeneous attenuation is absent (i.e., the shaft and the two opposite walls).

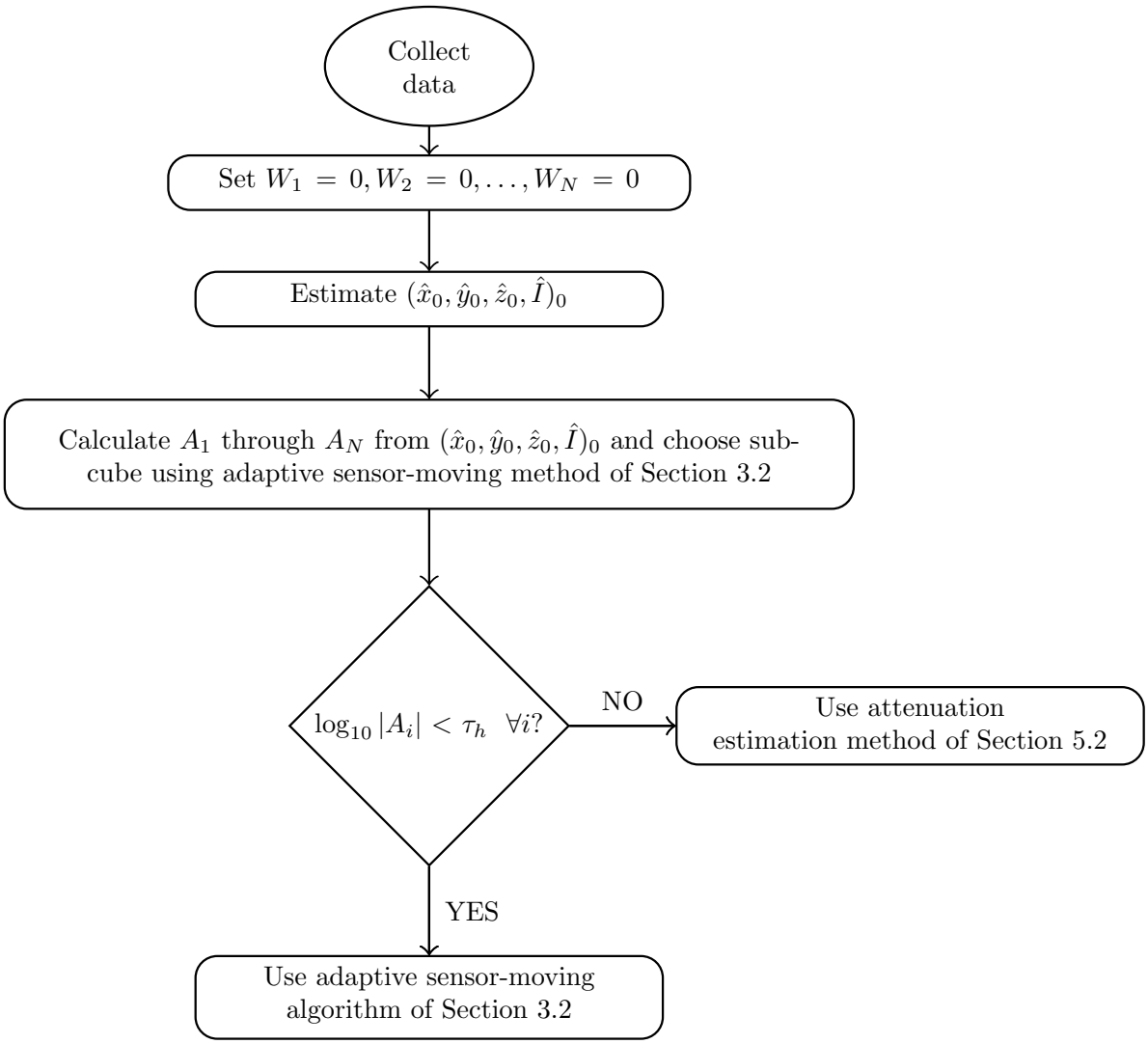

Figure 5.22: Hybrid estimation algorithm incorporating homogeneous and non-homogeneous attenuation.

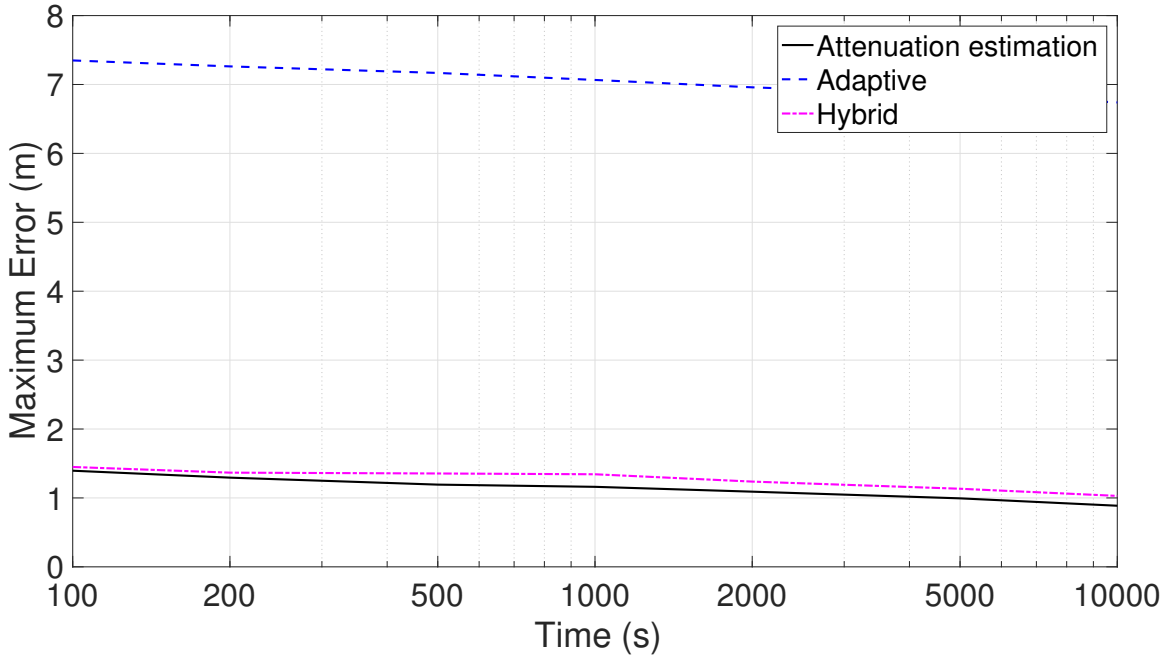

Figure 5.23: Maximum error vs. time for attenuation estimation, adaptive and hybrid algorithms, central attenuating shaft.

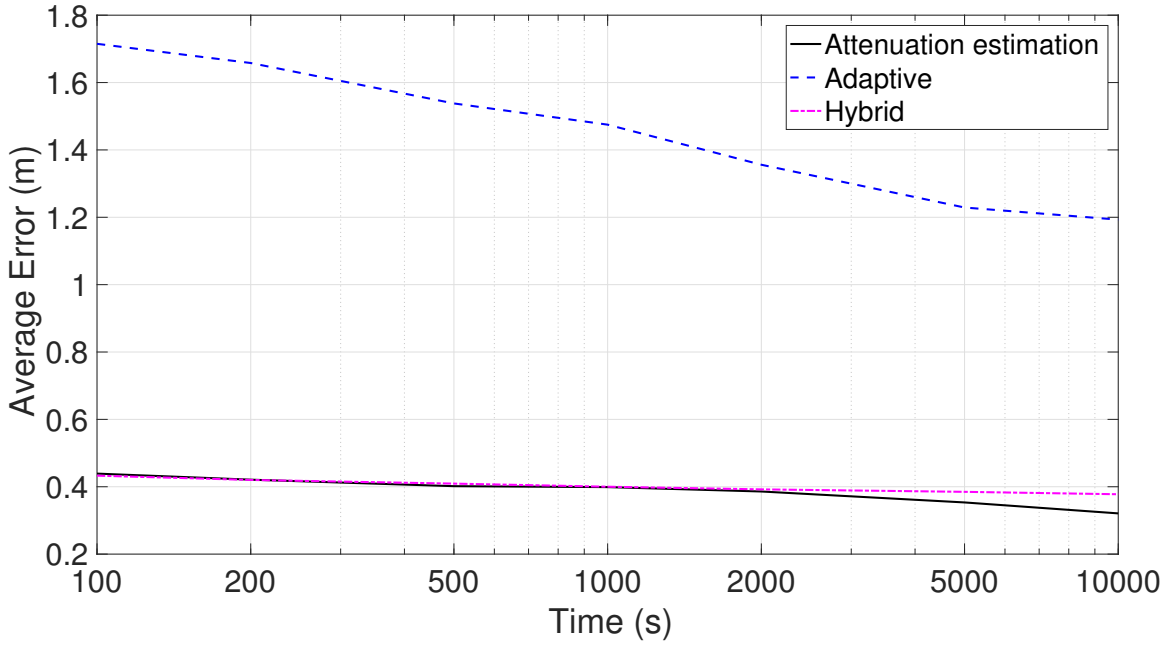

Figure 5.24: Average error vs. time for attenuation estimation, adaptive and hybrid algorithms, central attenuating shaft.

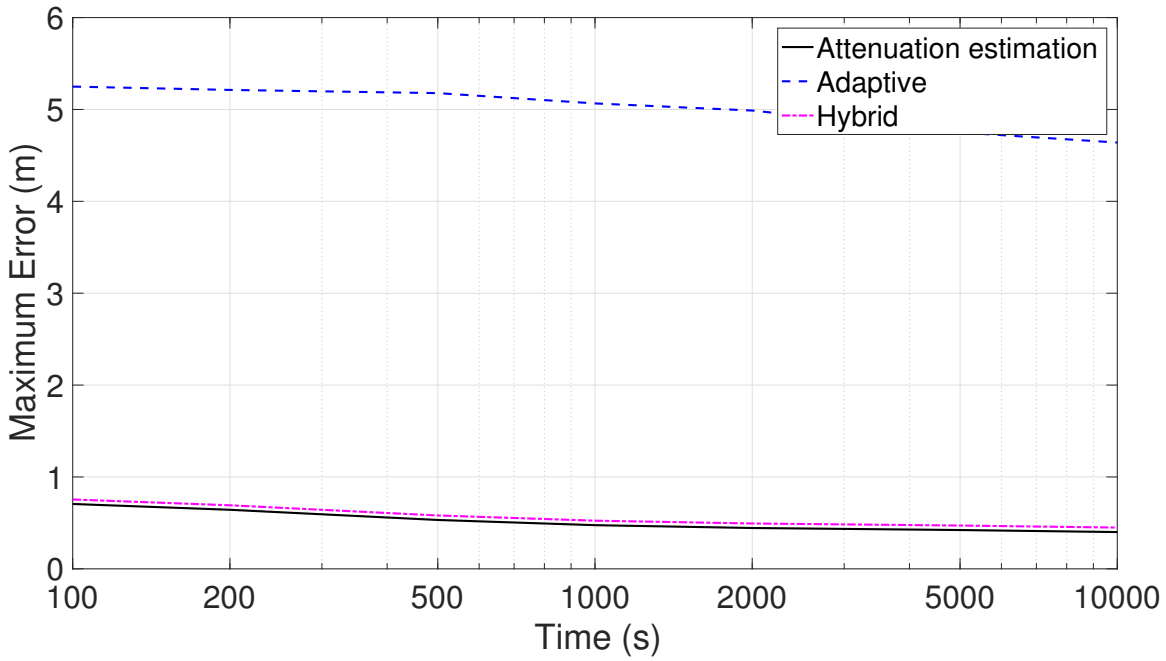

Figure 5.25: Maximum error vs. time for attenuation estimation, adaptive and hybrid algorithms, vertical attenuating wall.

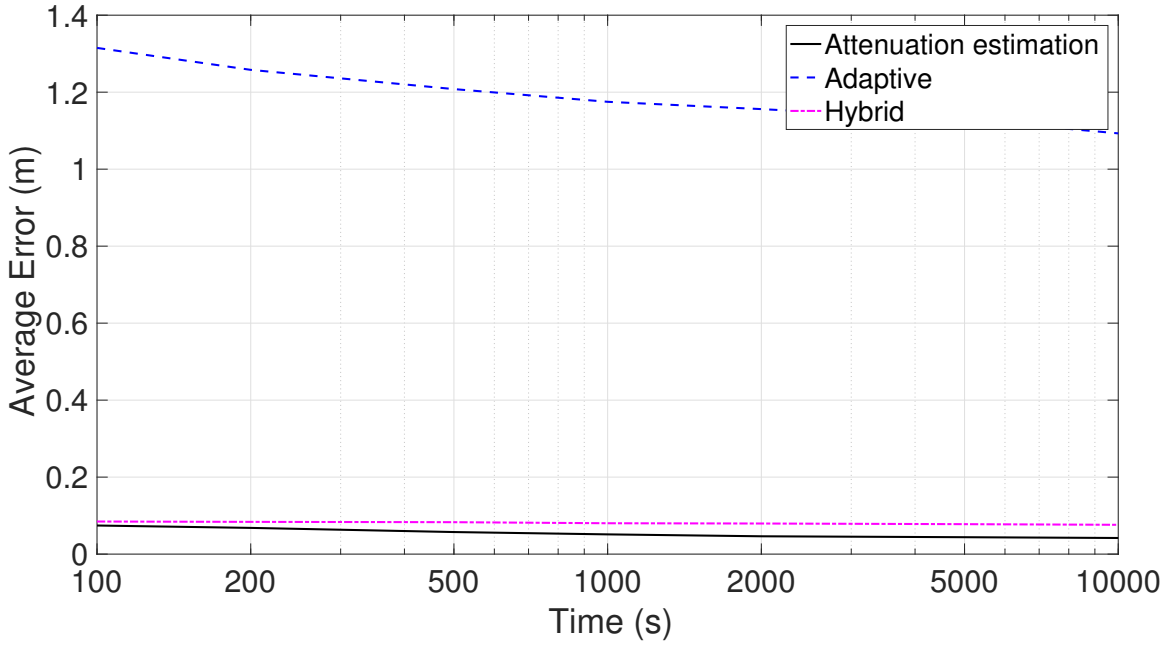

Figure 5.26: Average error vs. time for attenuation estimation, adaptive and hybrid algorithms, vertical attenuating wall.

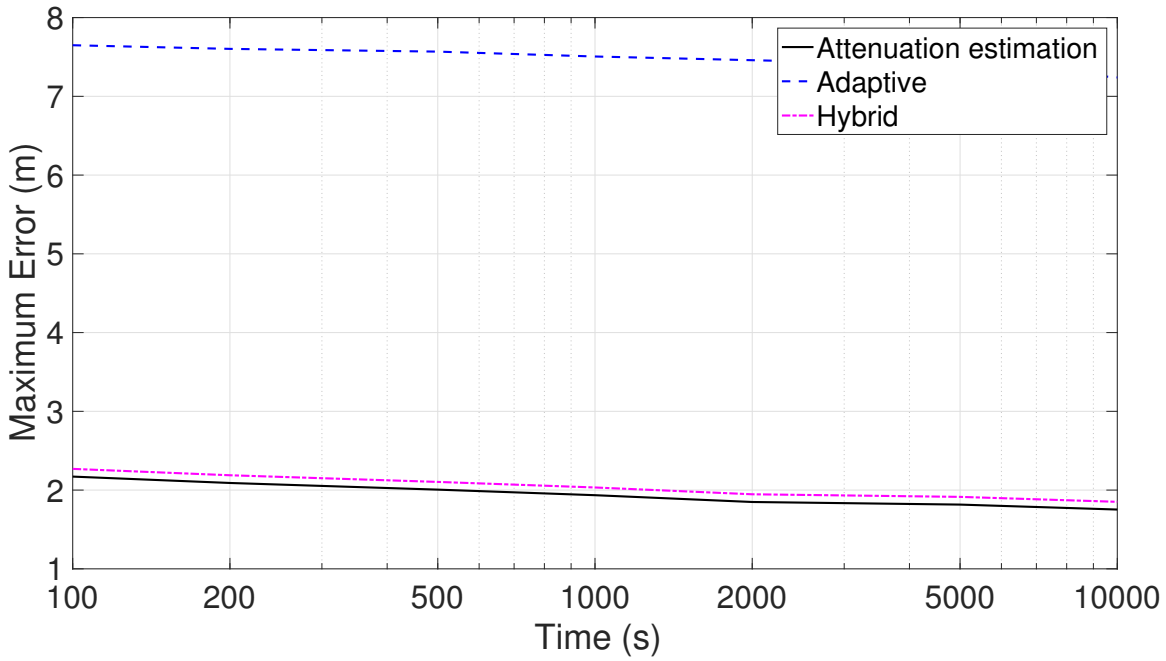

Figure 5.27: Maximum error vs. time for attenuation estimation, adaptive and hybrid algorithms, two attenuating walls (opposite).

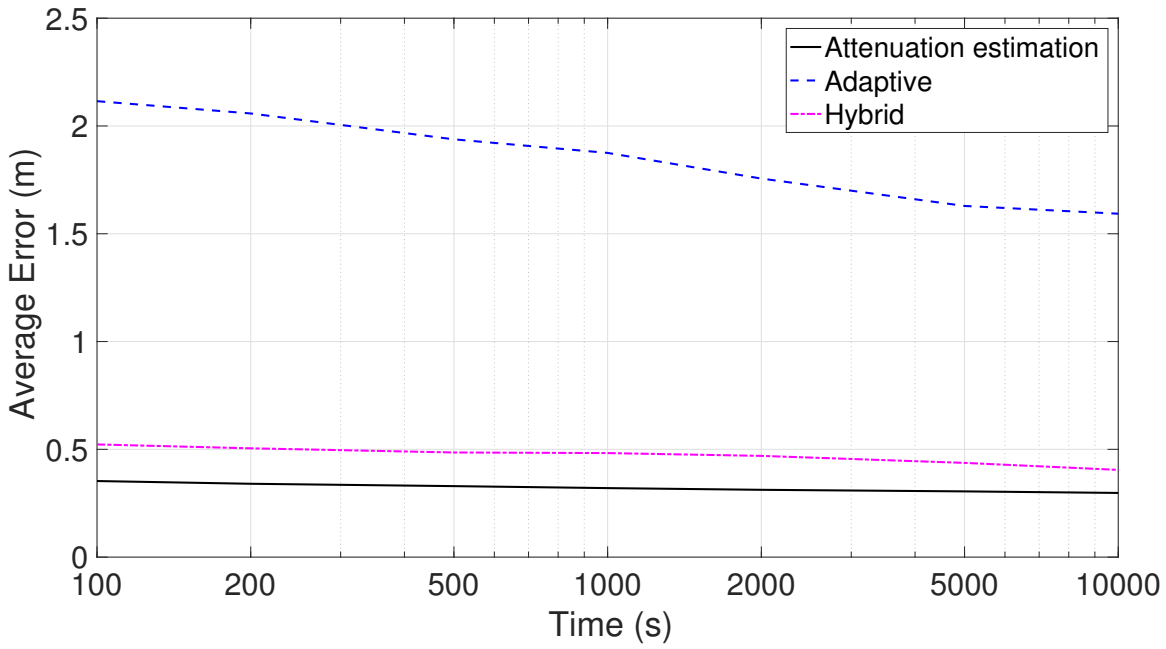

Figure 5.28: Average error vs. time for attenuation estimation, adaptive and hybrid algorithms, two attenuating walls (opposite).

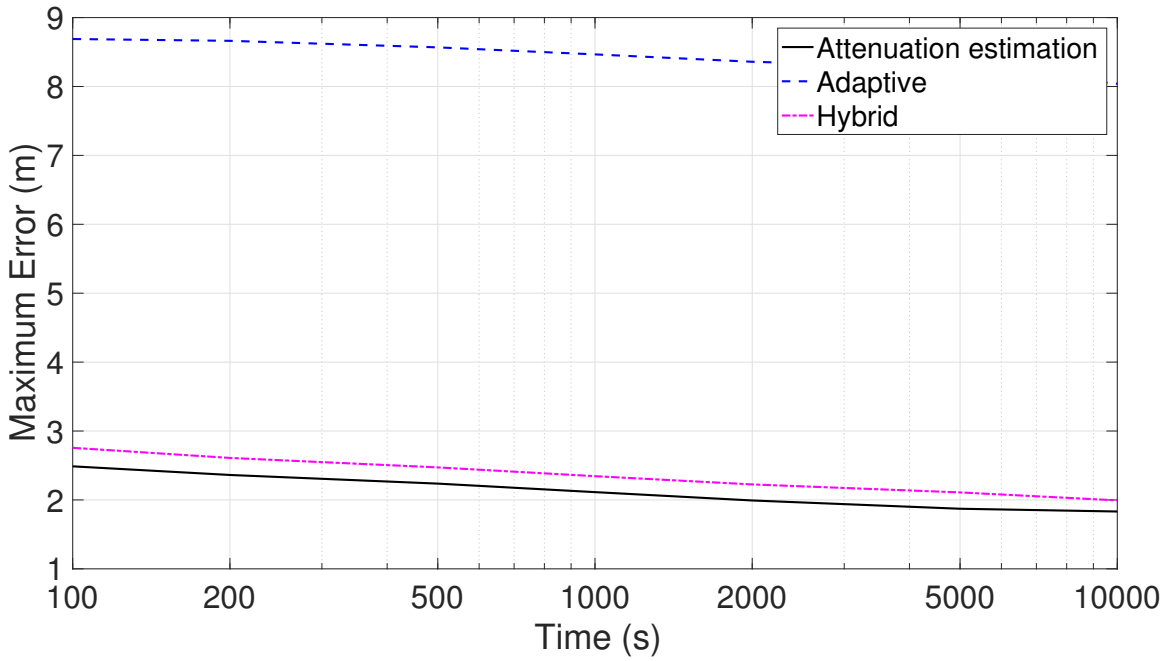

Figure 5.29: Maximum error vs. time for attenuation estimation, adaptive and hybrid algorithms, two attenuating walls (adjacent).

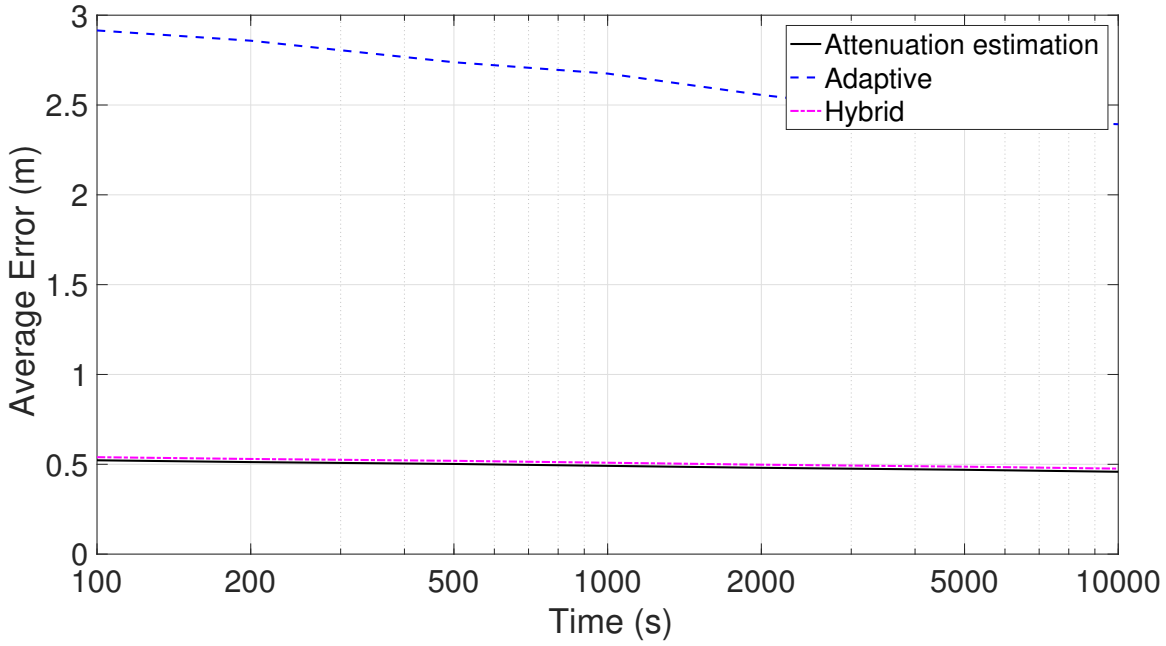

Figure 5.30: Average error vs. time for attenuation estimation, adaptive and hybrid algorithms, two attenuating walls (adjacent).

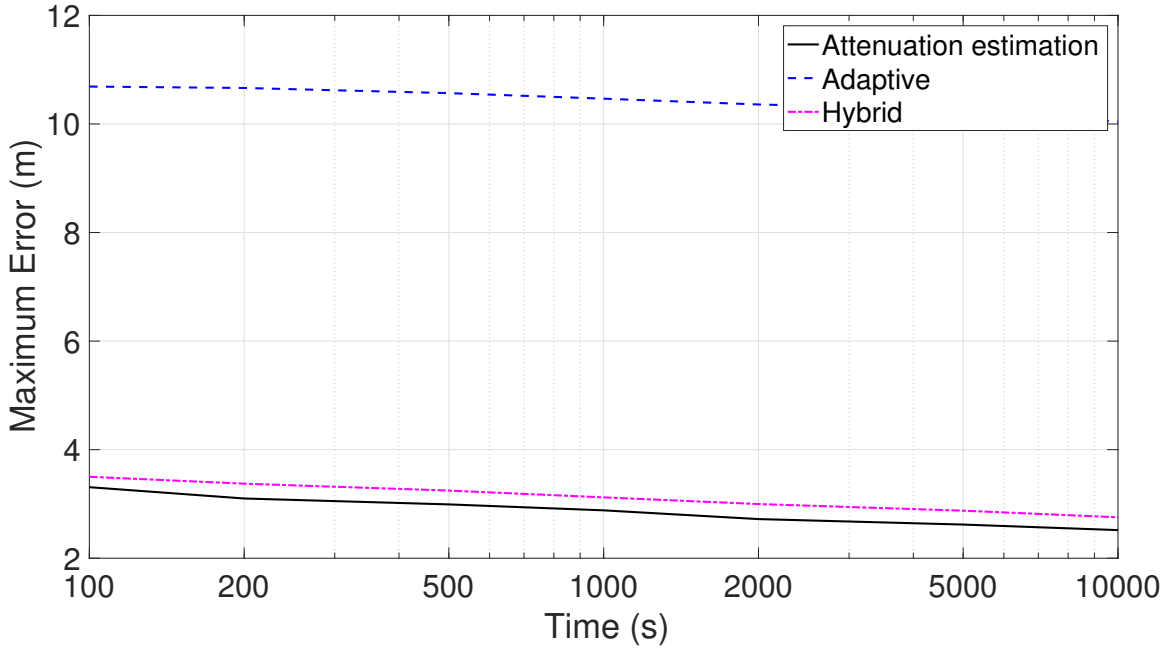

Figure 5.31: Maximum error vs. time for attenuation estimation, adaptive and hybrid algorithms, three attenuating walls.

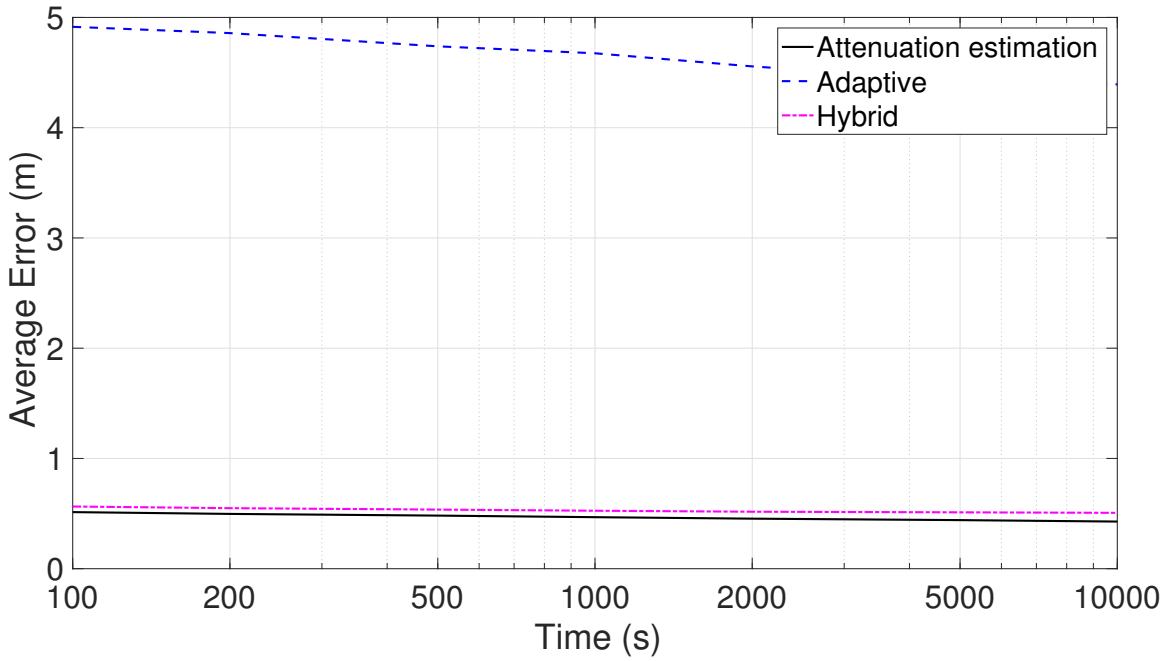

Figure 5.32: Average error vs. time for attenuation estimation, adaptive and hybrid algorithms, three attenuating walls.

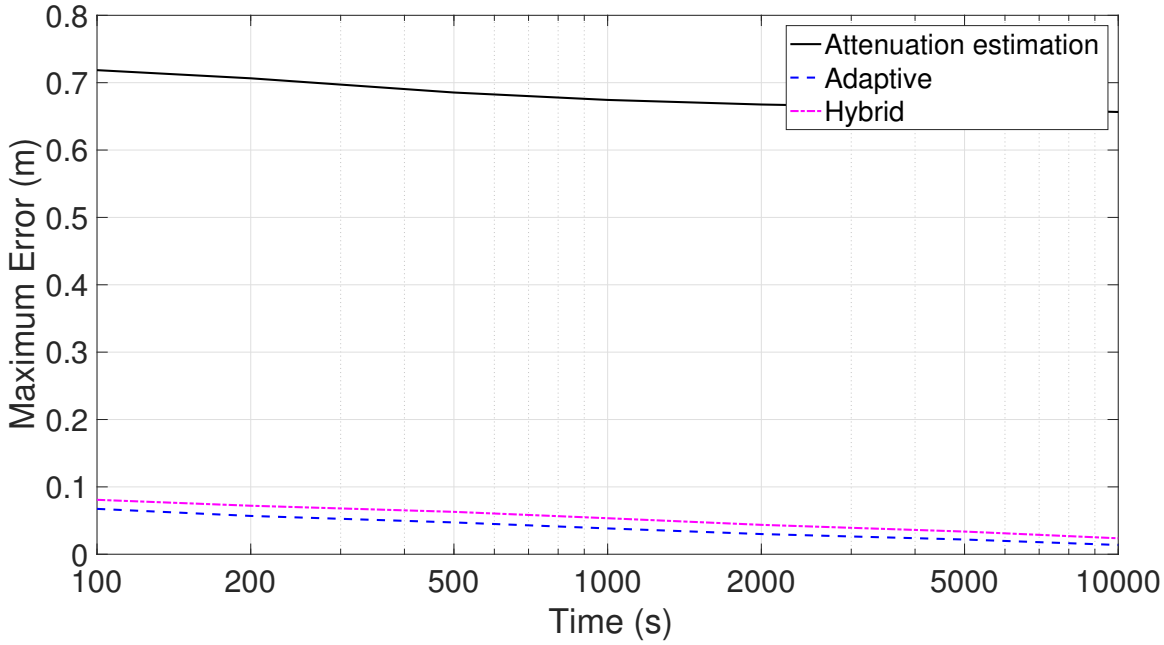

Figure 5.33: Maximum error vs. time for attenuation estimation, adaptive and hybrid algorithms, no attenuation.

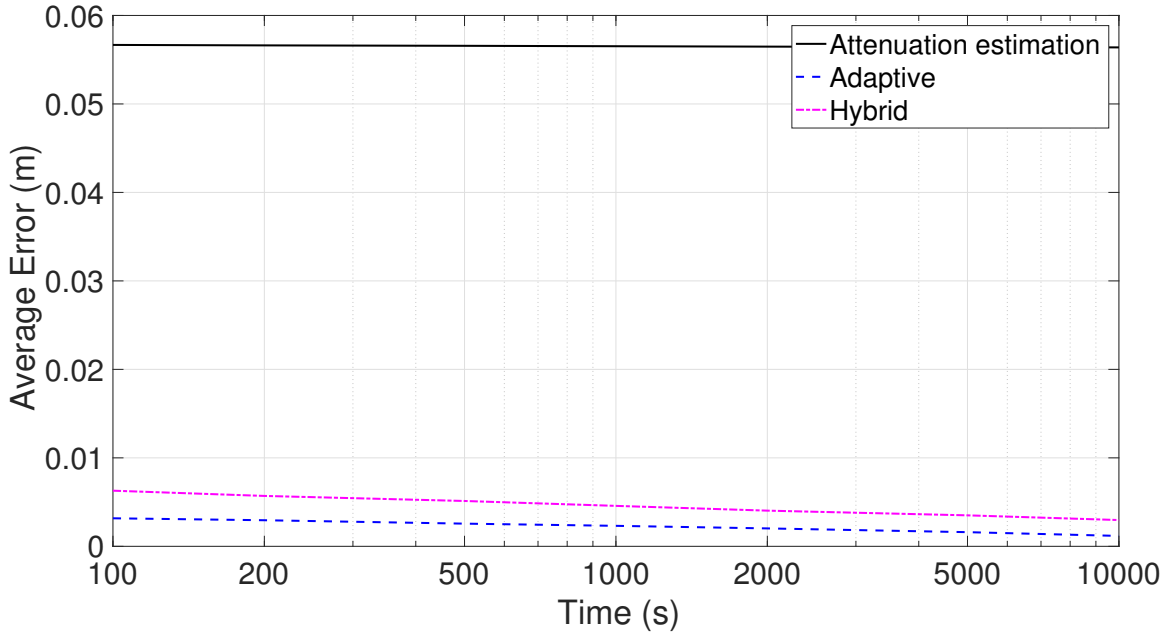

Figure 5.34: Average error vs. time for attenuation estimation, adaptive and hybrid algorithms, no attenuation.

The results in the absence of non-homogeneous attenuation (Figs. 5.33 and 5.34) show that the hybrid algorithm performs dramatically better than the attenuation estimation algorithm and only a little worse than the adaptive algorithm, especially with regards to the maximum error (Fig. 5.33). In general, it can be concluded that the hybrid algorithm is effective at determining whether attenuation is present and proceeding appropriately.

Finally, Figs. 5.35 and 5.36 compare the performance of the hybrid algorithm in the various attenuating environments (and in the absence of non-homogeneous attenuation). The results are similar to those of the attenuation estimation algorithm (Figs. 5.19 and 5.20) except that the performance in the absence of non-homogeneous attenuation is far better than it was previously.

5.5 Conclusion

This chapter presents several novel techniques for estimating the location of a nuclear source in which significant non-homogeneous attenuation maybe present. The algorithm of Section 5.2 is designed for non-homogeneous attenuation, whereas the algorithm of Section 5.4 is designed to operate well, whether non-homogeneous attenuation is present or absent. The results of the latter algorithm perform nearly as well as the former algorithm when non-homogeneous attenuation is present and nearly as well as the adaptive algorithm of Chapter 3 when non-homogeneous attenuation is absent.

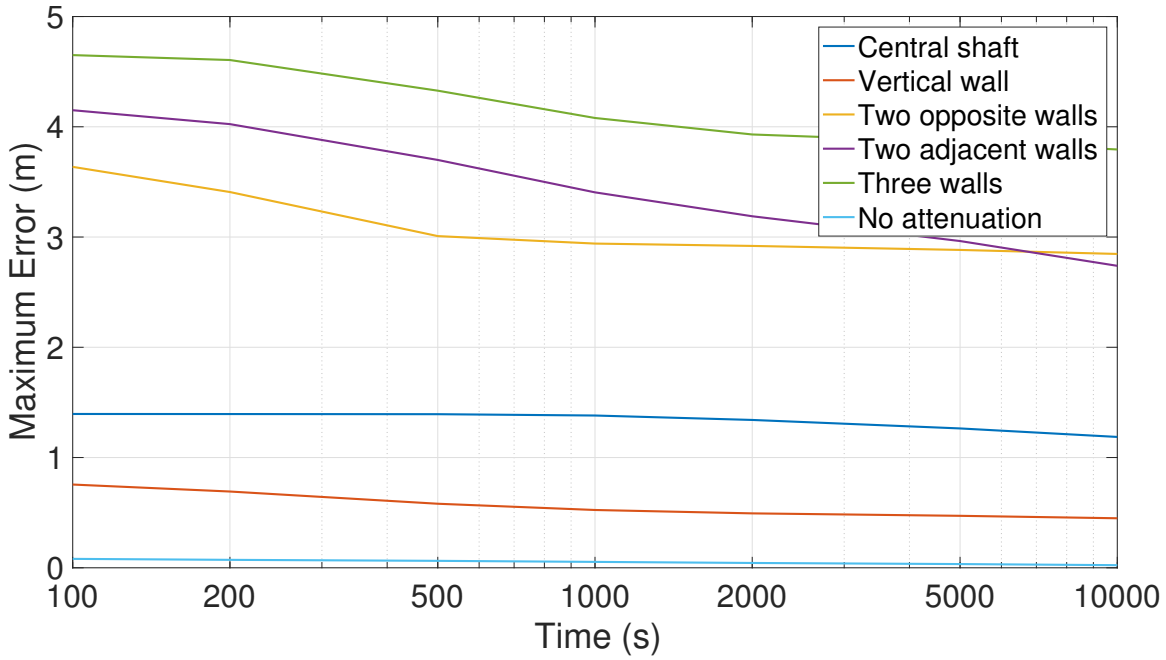

Figure 5.35: Maximum error vs. time for the hybrid algorithm for different attenuating structures.

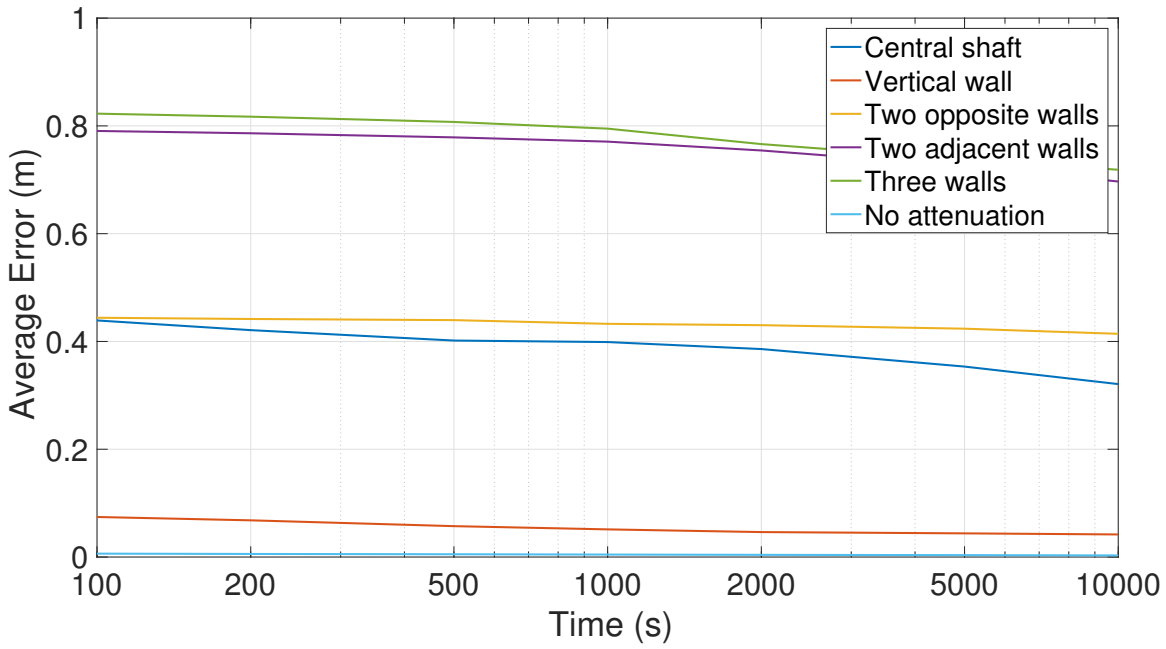

Figure 5.36: Average error vs. time for the hybrid algorithm for different attenuating structures.

Chapter 6

Conclusions and Future Work

This dissertation explores how to mitigate real-world constraints encountered in the problem of locating a nuclear source in crowded metropolitan areas. In Chapter 2, a network of sensors with fixed locations are employed. Various accessibility constraints are explored. It is seen that the location estimates are most accurate when the sensors have access to all faces of the building. However, even with limitations on where sensors can operate, it is possible to estimate the source location and strength accurately with a sufficiently large number of sensors.

Chapter 3 explores the idea of using only a small number of physical sensors to visit a set of strategic sensor locations as opposed to collecting data from a large number of sensor locations. An adaptive algorithm is presented that is based on the fact that spatial diversity of sensor locations is as important as proximity to the source when it comes to problem of location estimation. For purposes of comparison, a set of non-adaptive algorithms are also presented that involve a small number of physical sensors visiting a set of fixed locations. The accessibility constraints explored in Chapter 2 are also considered here. It is seen from the results that the adaptive algorithm vastly outperform the non-adaptive methods, even when only a

small number of sensor locations are available.

The effects of non-homogeneous attenuation are explored in Chapter 4. Specifically, an error detection metric is proposed, based on the estimated count arrival rate and the actual count arrival rate at each sensor location. This metric can be used to identify situations in which significant non-homogeneous attenuation has skewed estimation results. It is seen from the results that this metric is capable of identifying specific sensor locations affected by non-homogeneous attenuation.

Based on the metric from Chapter 4, a multi-stage iterative estimation algorithm is presented in Chapter 5. This algorithm uses the error detection metric of the previous chapter to identify sensor locations where significant absorption has taken place and iteratively estimates the attenuation effects along with the source location and strength. This attenuation estimation method is specifically designed to counter the effects of non-homogeneous attenuation when no prior knowledge regarding the attenuating structure is available. The results show that the attenuation estimation algorithm can successfully locate a source of nuclear radiation even in the presence of significant non-homogeneous attenuation. Finally, this chapter also combines the adaptive method of Chapter 3 to the attenuation estimation method of Chapter 5 in order to propose a hybrid method that can estimate the source location irrespective of the presence or absence of non-homogeneous attenuation.

This dissertation presents a number of algorithms for estimating the location and strength of a single isotropic point source. A problem meriting future research is the situation in which multiple isotropic sources are present within the search space, and the number of such sources is unknown. While the error detection metric can predict with a high level of reliability whether the estimates obtained are close to the true location of the source, it does not say anything about the source of such error. Therefore, it would need to be enhanced further in order to detect the number of

sources present within the search space. Combined with unknown non-homogeneous attenuation, the presence of multiple sources within a search space poses an interesting problem in the domain of nuclear radiation source estimation.

Another interesting topic for future consideration is estimating the effect of shielding materials on the location estimation problem. While this dissertation explores how non-homogeneous attenuation affects the estimation results, the attenuation considered is purely due to the specific structure of the building. In real world scenarios, additional shielding materials of non-uniform thickness may be used to hide the nuclear material and can significantly affect the location estimates. This is an important problem that should be explored in future research.

Bibliography

[1] HE Baidoo-Williams, R Mudumbai, E Bai, and S Dasgupta. Some theoretical limits on nuclear source localization and tracking. In *Information Theory and Applications Workshop (ITA), 2015*, pages 270–274. IEEE, 2015.

[2] Yaakov Bar-Shalom, X Rong Li, and Thiagalingam Kirubarajan. *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software*. John Wiley & Sons, 2004.

[3] James O Berger. *Statistical Decision Theory and Bayesian Analysis*. Springer Science & Business Media, 2013.

[4] Mani Chandy, Concetta Pilotto, and Ryan McLean. Networked sensing systems for detecting people carrying radioactive material. In *Networked Sensing Systems, 2008. INSS 2008. 5th International Conference on*, pages 148–155. IEEE, 2008.

[5] Harald Cramér. *Mathematical Methods of Statistics*, volume 9. Princeton University Press, 1945.

[6] George B Dantzig, Alex Orden, Philip Wolfe, et al. The generalized simplex method for minimizing a linear form under linear inequality restraints. *Pacific Journal of Mathematics*, 5(2):183–195, 1955.

[7] Budhaditya Deb. Iterative estimation of location and trajectory of radioactive sources with a networked system of detectors. *IEEE Transactions on Nuclear Science*, 60(2):1315–1326, 2013.

[8] Paul E Fehlau. Comparing a recursive digital filter with the moving-average and sequential probability-ratio detection methods for snm portal monitors. *Nuclear Science, IEEE Transactions on*, 40(2):143–146, 1993.

[9] Ronald Aylmer Fisher. *Statistical Methods for Research Workers*. Genesis Publishing Pvt Ltd, 1925.

[10] Ajith Gunatilaka, Branko Ristic, and Ralph Gailis. On localisation of a radiological point source. In *Information, Decision and Control, 2007. IDC'07*, pages 236–241. IEEE, 2007.

[11] Jinlu Han, Yaojin Xu, Long Di, and YangQuan Chen. Low-cost multi-uav technologies for contour mapping of nuclear radiation field. *Journal of Intelligent & Robotic Systems*, 70(1-4):401–410, 2013.

[12] Kenneth D Jarman, L Eric Smith, Deborah K Carlson, and Dale N Anderson. Sequential probability ratio test for long-term radiation monitoring. In *IEEE, Nuclear Science Symposium Conference Record*, volume 2, pages 1458–1462, 2003.

[13] Robert W Keener. *Statistical Theory: Notes for a Course in Theoretical Statistics*. Springer, 2006.

[14] Glenn F Knoll. *Radiation Detection and Measurement*. John Wiley & Sons, 2010.

[15] Hsien-I Lin et al. Search strategy of a mobile robot for radiation sources in an unknown environment. In *Advanced Robotics and Intelligent Systems (ARIS), 2014 International Conference on*, pages 56–60. IEEE, 2014.

[16] Mathworks. fminsearchbnd, fminsearchcon.

[17] Mathworks. Nonlinear programming solver.

[18] Gaku Minamoto, Eijiro Takeuchi, and Satoshi Tadokoro. Estimation of ground surface radiation sources from dose map measured by moving dosimeter and 3d map. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 1889–1895. IEEE, 2014.

[19] Mark Morelande, Branko Ristic, and Ajith Gunatilaka. Detection and parameter estimation of multiple radioactive sources. In *IEEE 10th International Conference on Information Fusion*, pages 1–7, 2007.

[20] Mark R Morelande and Branko Ristic. Radiological source detection and localisation using bayesian techniques. *IEEE Transactions on Signal Processing*, 57(11):4220–4231, 2009.

[21] Christian Musso, Nadia Oudjane, and Francois Le Gland. Improving regularised particle filters. In *Sequential Monte Carlo Methods in Practice*, pages 247–271. Springer, 2001.

[22] Robert J Nemzek, Jared S Dreicer, David C Torney, and Tony T Warnock. Distributed sensor networks for detection of mobile radioactive sources. *IEEE Transactions on Nuclear Science*, 51(4):1693–1700, 2004.

[23] National Institute of Standards and Technology. Composition of concrete.

[24] National Institute of Standards and Technology. X-ray mass attenuation coefficients.

[25] Chetan D Pahlajani, Ioannis Poulakakis, and Herbert G Tanner. Networked decision making for poisson processes with applications to nuclear detection. *Automatic Control, IEEE Transactions on*, 59(1):193–198, 2014.

[26] Wolfgang KH Panofsky. Nuclear proliferation risks, new and old. *Issues in Science and Technology*, 19(4):73–74, 2003.

[27] C Radhakrishna Rao. Information and accuracy attainable in the estimation of statistical parameters. *Bull Calcutta. Math. Soc.*, 37:81–91, 1945.

[28] Nageswara SV Rao, Mallikarjun Shankar, Jren-Chit Chin, David KY Yau, Chris YT Ma, Yong Yang, Jennifer C Hou, Xiaochun Xu, and Sartaj Sahni. Localization under random measurements with application to radiation sources. In *Information Fusion, 2008 11th International Conference on*, pages 1–8. IEEE, 2008.

[29] Nageswara SV Rao, Mallikarjun Shankar, Jren-Chit Chin, David KY Yau, Srinivasagopalan Srivathsan, S Sitharama Iyengar, Yong Yang, and Jennifer C Hou. Identification of low-level point radiation sources using a sensor network. In *Information Processing in Sensor Networks, 2008. IPSN'08. International Conference on*, pages 493–504. IEEE, 2008.

[30] Branko Ristic, Ajith Gunatilaka, and Mark Rutten. An information gain driven search for a radioactive point source. In *Information Fusion, 2007 10th International Conference on*, pages 1–8. IEEE, 2007.

[31] Branko Ristic, Mark Morelande, and Ajith Gunatilaka. Information driven search for point sources of gamma radiation. *Signal Processing*, 90(4):1225–1239, 2010.

[32] Daniel L Stephens and Anthony J Peurrung. Detection of moving radioactive sources using sensor networks. *IEEE Transactions on Nuclear Science*, 51(5):2273–2278, 2004.

[33] Jianxin Sun, Herbert G Tanner, and Ioannis Poulakakis. Active sensor networks for nuclear detection. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 3549–3554. IEEE, 2015.

[34] Jerry Towler, Bryan Krawiec, and Kevin Kochersberger. Radiation mapping in post-disaster environments using an autonomous helicopter. *Remote Sensing*, 4(7):1995–2015, 2012.

[35] J Uhlmann, S Julier, and HF Durrant-Whyte. A new method for the nonlinear transformation of means and covariances in filters and estimations. *IEEE Trans. on Automatic Control*, 45(3):477–482, 2000.

[36] Harry L Van Trees. *Detection, Estimation, and Modulation Theory*. John Wiley & Sons, 2004.

[37] RB Vilim, RT Klann, SC de la Barrera, PL Vilim, and IA Ross. Tracking of weak radioactive sources in crowded venues. In *2009 IEEE Nuclear Science Symposium Conference Record (NSS/MIC)*, pages 995–1001. IEEE, 2009.

[38] Roland E Weibel and R John Hansman. An integrated approach to evaluating risk mitigation measures for uav operational concepts in the nas. *Proc. of InfoTech at Aerospace: Advancing Contemporary Aerospace Technologies and their Integration*, 1:509–519, 2005.

[39] Tonglin Zhang. Radioactive target detection using wireless sensor network. In *Computer, Informatics, Cybernetics and Applications*, pages 293–301. Springer, 2012.