

# **RAPID ROBUST RADIOLOGICAL SEARCH AND MAPPING ALGORITHMS**

A Dissertation  
Presented to  
The Academic Faculty

By

Samuel Kemp

In Partial Fulfillment  
of the Requirements for the Degree  
Doctor of Philosophy in the  
Georgia Institute of Technology  
School of Mechanical Engineering

Georgia Institute of Technology

Sep 2023

© Samuel Kemp 2023

# RAPID ROBUST RADIOLOGICAL SEARCH AND MAPPING ALGORITHMS

Thesis committee:

Dr. Jonathan Rogers  
School of Aerospace Engineering  
*Georgia Institute of Technology*

Dr. Anirban Mazumdar  
School of Mechanical Engineering  
*Georgia Institute of Technology*

Dr. Anna Erickson  
School of Mechanical Engineering  
*Georgia Institute of Technology*

Dr. Craig Bakker  
Research Scientist  
*Pacific Northwest National Laboratory*

Dr. Seth Hutchinson  
School of Interactive Computing  
*Georgia Institute of Technology*

Date approved: January 26, 2023

Don't rely on motivation for anything. It is fleeting and unreliable. Discipline, however, is  
unyielding.

*Anonymous*

For my soon-to-be wife, Ashley

## **ACKNOWLEDGMENTS**

This material is based upon work supported by the Department of Energy / National Nuclear Security Administration under Award Number(s) DE-NA0003921. This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

## TABLE OF CONTENTS

<b>Acknowledgments</b> . . . . .	v
<b>List of Tables</b> . . . . .	x
<b>List of Figures</b> . . . . .	xii
<b>List of Acronyms</b> . . . . .	xiv
<b>Nomenclature</b> . . . . .	xvi
<b>Chapter 1: Introduction</b> . . . . .	1
1.1 Motivation . . . . .	1
1.2 Prior Work . . . . .	2
1.2.1 Mapping . . . . .	2
1.2.2 Source Term Estimation . . . . .	4
1.3 Dissertation Overview . . . . .	7
<b>Chapter 2: Heterogeneous Multi Agent Search</b> . . . . .	11
2.1 Problem Definition . . . . .	11
2.2 Measurement and Radiation Field Modeling . . . . .	11
2.2.1 Sensor Modeling . . . . .	11
2.2.2 Gaussian Process Radiation Field Modeling . . . . .	12

2.2.3	Integrated Squared Difference Error Metric . . . . .	14
2.3	Path Planning Algorithms . . . . .	14
2.3.1	Information Theoretic Path Planning . . . . .	14
2.3.2	Coverage Path Planning . . . . .	16
2.4	Simulation Experimental Setup . . . . .	17
2.4.1	Setup . . . . .	17
2.4.2	Individual Example . . . . .	18
2.5	Results . . . . .	19
<b>Chapter 3: Dynamic Discrete Particle Filter . . . . .</b>		<b>23</b>
3.1	Problem Definition . . . . .	23
3.2	Methodology . . . . .	25
3.2.1	Overview . . . . .	25
3.2.2	Precomputation of Attenuation Kernels . . . . .	26
3.2.3	Particle Filter Update Step . . . . .	33
3.2.4	Dynamic Particle Count Adjustment Algorithm . . . . .	38
3.3	Simulation Experimental Setup . . . . .	40
3.3.1	Setup . . . . .	40
3.3.2	Performance Metrics . . . . .	42
3.3.3	Individual Example . . . . .	44
3.4	Simulation Results . . . . .	45
3.4.1	Attenuation Kernel Study . . . . .	45
3.4.2	Particle Count Study . . . . .	48

3.4.3	Dynamic Particle Count Adjustment Algorithm Study . . . . .	48
3.4.4	Uncertain Absorption Parameters Study . . . . .	51
3.5	Lab Scale Experimental Setup . . . . .	53
3.5.1	Setup . . . . .	53
3.5.2	Measurement Conversion . . . . .	54
3.5.3	Performance Metrics . . . . .	55
3.6	Experimental Results . . . . .	55
3.6.1	Limitations . . . . .	56
<b>Chapter 4: Parallel Log-DDPF . . . . .</b>		<b>65</b>
4.1	Problem Definition . . . . .	65
4.2	Methodology . . . . .	66
4.2.1	Dynamic Discrete Particle Filter . . . . .	68
4.2.2	Log Particle Filter (PF) . . . . .	70
4.2.3	Log Minimum Mean Squared Error Estimate . . . . .	73
4.2.4	Gamma Spectrum Unfolding Algorithm . . . . .	73
4.2.5	Parallel PF . . . . .	79
4.3	Lab Scale Experiments . . . . .	80
4.3.1	Setup . . . . .	80
4.3.2	Detector Model . . . . .	82
4.3.3	Results . . . . .	85
<b>Chapter 5: Conclusion . . . . .</b>		<b>88</b>
5.1	Contributions . . . . .	88

5.2	Recommended Future Work . . . . .	89
5.2.1	Optimal Kernel Generation Parameters . . . . .	89
5.2.2	Information Driven Active Search . . . . .	90
5.2.3	MCNP Kernel Generation Pipeline . . . . .	90
5.2.4	Measurements in Motion . . . . .	90
5.2.5	Hardware Development . . . . .	91
5.2.6	Source Measurement Factor Graph Integration . . . . .	91
	<b>References</b> . . . . .	<b>92</b>
	<b>Vita</b> . . . . .	<b>102</b>

## LIST OF TABLES

1.1	Levels of assumptions made for each category . . . . .	6
1.2	Categorizing references by level of assumptions made . . . . .	8
1.3	Categorizing references by type of measurements used . . . . .	8
3.1	Simulation parameters. . . . .	43
3.2	Monte Carlo results comparing Continuous Particle Filter (CPF) and Discrete Particle Filter (DPF) performance ( $N = 500$ ). . . . .	47
3.3	Monte Carlo results with DPF using varying $N$ . . . . .	49
3.4	DPF and Dynamic Discrete Particle Filter (DDPF) compared for varying $r_{\max}$ . . . . .	50
3.5	DDPF performance with obstacle attenuation parameter uncertainty. . . . .	53
3.6	Obstacle characteristics. . . . .	54
3.7	Hardware measurement scenarios. . . . .	54
3.8	Hardware results. . . . .	56
4.1	Gain-shift correction coefficients . . . . .	77
4.2	Table of isotopes after Unfolding Algorithm is run. . . . .	78
4.3	Source location coordinates . . . . .	82
4.4	Experiment matrix . . . . .	83
4.5	Source activities, strengths, and conversion factors . . . . .	85

4.6	Experimental results . . . . .	86
-----	--------------------------------	----

## LIST OF FIGURES

2.1	Uninhabited Ground Vehicle (UGV)-only Search Paths at 435 sec. . . . .	19
2.2	Uninhabited Aerial Vehicle (UAV)-UGV Team Search Paths at 435 sec. . .	20
2.3	Gaussian Process Regression (GPR) Radiation Model at 435 sec. . . . .	21
2.4	Integrated Squared Difference (ISD) vs Time for Example Simulation. . . .	22
2.5	Average ISD (solid) and +1 Standard Deviation (dashed) vs time for Monte Carlo study comparing UAV-UGV teams using Information Theoretic Path Planning (ITPP) and Random Sampling (RS) with a UGV-only team. . . . .	22
3.1	Overview of Dynamic Discrete Particle Filter algorithm. . . . .	25
3.2	Example of discretized ray tracing over source and measurement grids. . . .	31
3.3	Example of count rate values computed along ray. . . . .	31
3.4	Illustration of obstacle culling parameters. . . . .	32
3.5	Search area for simulation trials. (left) Satellite view of search area with buildings outlined in green. (center) Ray-traced radiation intensity from three example sources (red = high intensity). (right) Combined plot of radiation and environment. . . . .	40
3.6	Sequence of measurement locations. . . . .	42
3.7	Error associations for scenarios in which particle filter overestimates (left) and underestimates (right) source cardinality. . . . .	44
3.8	Stages of the PF. (top-left) initialization. (bottom-right) post-processing. (all others) update. . . . .	58
3.9	Convergence metrics and particle count for sample trial using the DDPF. . .	59

3.10	Position error for CPF and DPF in cases with and without obstacles. . . . .	59
3.11	Position error for DPFs with different particle counts. . . . .	60
3.12	Monte Carlo position errors for DPF and DDPF. . . . .	60
3.13	Mean negative log likelihoods over all update steps for DPF and DDPF. . .	61
3.14	Monte Carlo runtimes for DPF and DDPF. . . . .	61
3.15	Source strength estimation error for varying levels of obstacle attenuation parameter uncertainty. . . . .	62
3.16	Photo of testing area with obstacles and ground robot visible. . . . .	62
3.17	Recorded spectra from 2 source Cs-137 experiments. (left) All recorded spectra. (middle) Integrated area to become count measurements. (right) Example of saturated spectrum. . . . .	63
3.18	Interpolated radiation field given by measurements of 2 Cs-137 sources. . .	63
3.19	Final particle set for experiment with 2 Cs-137 sources. . . . .	64
4.1	Overview of Parallel Log DDPF algorithm. . . . .	67
4.2	Overview of Parallel Log DDPF phases of operation. . . . .	68
4.3	Overview of DDPF algorithm. . . . .	69
4.4	Linear calibration gain and gross counts plotted using Cs-137 photopeaks for calibration. . . . .	76
4.5	Example calibrated spectrum taken over 120 s with Kromek Sigma 50. Peak centroids shown by vertical lines. Peak areas are filled in with base- lines removed. . . . .	79
4.6	Search area for experiments. . . . .	82
4.7	Robot to take measurements. . . . .	83
4.8	Convergence of Parallel Log DDPF for Experiment C. . . . .	87

## LIST OF ACRONYMS

<b>APSL</b>	Additive Point Source Localization
<b>CDF</b>	Cumulative Density Function
<b>CPF</b>	Continuous Particle Filter
<b>DDPF</b>	Dynamic Discrete Particle Filter
<b>DPCA</b>	Dynamic Particle Count Adjustment
<b>DPF</b>	Discrete Particle Filter
<b>GIS</b>	Geographic Information System
<b>GP</b>	Gaussian Process
<b>GPR</b>	Gaussian Process Regression
<b>ISD</b>	Integrated Squared Difference
<b>ITPP</b>	Information Theoretic Path Planning
<b>LiDAR</b>	Light Detection and Ranging
<b>MAP</b>	Maximum <i>A Posteriori</i>
<b>MCNP</b>	Monte Carlo N-Particle Transport Code
<b>MMSE</b>	Minimum Mean Squared Error
<b>OSM</b>	OpenStreetMaps
<b>PC</b>	Progressive Correction
<b>PDF</b>	Probability Density Function
<b>PF</b>	Particle Filter
<b>RS</b>	Random Sampling
<b>SLAM</b>	Simultaneous Localization and Mapping
<b>STE</b>	Source Term Estimation

**UAV** Uninhabited Aerial Vehicle

**UGV** Uninhabited Ground Vehicle

**VFC** Viewing-Frustrum Culling

## NOMENCLATURE

$\beta$	absorption coefficient
$\Delta, \delta$	exponents used for iterative jacobian algorithm
$\epsilon_{\text{pos}}$	position error
$\epsilon_{\varphi}$	strength error
$\epsilon_{ex}$	detector extrinsic efficiency
$\epsilon_{in}$	detector intrinsic efficiency
$\eta_{\text{pt}}$	peak-to-total ratio
$\Gamma$	linear calibration gain
$\hat{r}$	estimated source cardinality
$\lambda$	mean radiation count
$\lambda_{\text{sat}}$	count saturation value
$\mathbb{N}$	natural numbers
$\mathbb{R}^d$	set of real coordinates in $d$ dimensions
$\mathbf{X}$	particle set
$\mathcal{A}$	search volume
$\mathcal{N}(0, \sigma)$	Gaussian kernel with standard deviation $\sigma$
$\mathcal{P}$	Poisson distribution
$\mathcal{Q} = \{q_1, \dots, q_k\}$	set of log Poisson likelihoods
$\mathcal{U}$	uniform distribution
$\mathcal{Z}$	set of measurements
$\mu$	expected count rate
$\mu()$	mean of

$\mu_b$	mean background radiation in cps
$\mu_{rec}$	recorded count rate
$\nu$	random variable from Gaussian kernel for regularization
$\Omega$	vector of cumulative sum of weights
$\omega$	true interaction rate
$\psi$	azimuth angle
$\rho$	distance
$\sigma$	standard deviation value
$\sigma()$	standard deviation of
$\tau$	exposure time interval
$\tau_d$	dead time factor
eye	identity matrix
md	medium specifier
$\theta$	source parameter matrix
$\Theta_{i,j}$	attenuation kernel as source location indexed by $i$ and measurement location $j$
$\tilde{\mathcal{P}}$	saturated Poisson distribution
$\varphi$	equivalent source strength
$\xi$	placeholder for any particle source parameter
$\zeta$	ideal Poisson distributed count
$A$	source activity
$a, b, c, d$	tuning coefficients for gain calibration
$A_d$	detector cross sectional area
$B$	basis function coefficients
$Bq$	conversion from Curies to Becquerels
$C$	gross counts
$C(R_i)$	reward function of each subregion
$d$	source-detector distance

$d(R^{(k)} - R_i)$	distance between current and $k^{\text{th}}$ subregion
$E$	energy level (KeV)
$E_d$	detector efficiency
$f(x)$	random variable sampled from Gaussian Process (GP)
$f_h$	function to increase particle count
$f_l$	function to decrease particle count
$G$	geometric solid angle
$GP$	Gaussian process
$gpr$	indicates Gaussian Process Regression (GPR) estimate
$h(x)$	Gaussian process basis functions
$I$	transmittance
$i$	source location index
$J$	number of particles sampled by Dynamic Particle Count Adjustment algorithm
$j$	measurement location index
$K$	covariance
$k$	measurement index
$k(x, x')$	Gaussian process covariance function
$M$	absorption efficiency
$N$	number of particles
$n$	particle index
$n_1, n_2$	sizes of discrete 2D sampling grid
$N_s$	number of subregions
$p_{\text{dn}}$	probability of decreasing particle cardinality
$p_{\text{up}}$	probability of increasing particle cardinality
$Q, P$	tunable positive constants for uncertainty and distance respectively
$q_h$	log likelihood threshold to increase particle count
$q_l$	log likelihood threshold to decrease particle count

$r$	source cardinality
$r_d$	calibrated distance to source
$R_i$	subregion indexed by $i$
$r_{\max}$	maximum number of sources that the PF will estimate
$r_{sd}$	source-detector distance
$r_{\text{true}}$	true number of sources present
$S$	number of progressive correction stages
$s$	source index
$SNR$	signal to noise ratio
$t$	time
$TB$	total branching ratio
$u$	random variable used for resampling
$u_i, l_i$	upper and lower 95% confidence intervals for a subregion
$v$	zero mean noise
$W$	log normalization factor
$w$	normalized weights
$w^*$	un-normalized weights
$w_n$	particle weight
$x$	GPR input variable
$x, y, h$	coordinates
$y$	GPR response variable
$z$	observed counts
cps	counts per second
nint	nearest integer function

## SUMMARY

Radiological search and mapping are two separate problems that are currently performed by human operators in the field. These tasks could not only be more effective when performed by robotic agents, doing so would also keep human operators from being exposed to gamma radiation. Radiological mapping is the process of taking measurements to build an understanding of the contamination of an area as quickly as possible. This usually implies some degree of coverage for a predefined area. Radiological search is a similar problem that focuses on inferring what the parameters of a source of emissions might be and localizing them as quickly as possible. While a variety of techniques exist for both of these problems, they are usually slow or do not use all of the information available, and their performance suffers for it.

This work has two goals. The first is to use Monte Carlo simulations to quantify the improvements gained by using information driven search with a novel configuration of air and ground robots equipped with counting instruments. This method will be compared to the same configuration of robots performing random sampling and a configuration of ground robots performing a systematic rectilinear search. A linear reduction in mapping error with time is observed for the systematic search while exponential reductions are observed for the teams using both air and ground robots. The information driven search demonstrates the quickest reduction of mapping error with time.

The second goal is to propose and refine a particle filtering algorithm for localizing, identifying, and characterizing point sources of gamma radiation in the presence of obstacles. The proposed algorithm has five major improvements over the current state of the art. Firstly, it uses discrete precomputed attenuation kernels to perform radiation transport thousands of times per second. Secondly, it uses an introspective algorithm to dynamically adjust computational load to balance speed and accuracy. Thirdly it uses a gamma spectrum unfolding algorithm to incorporate spectral data. Fourthly, it uses multiple parallel particle

filters for each isotope of interest, thus tailoring the attenuation kernels to the appropriate isotope. Finally, it performs all likelihood calculations in the logarithmic domain to improve robustness and accuracy. The overall methodology is evaluated with Monte Carlo simulations and lab scale results using live sources of gamma radiation. The results show vast reductions in computational burden for embedded hardware, increased search speed, and reduced error.

# CHAPTER 1

## INTRODUCTION

### 1.1 Motivation

Atomic fission is the most powerful technology available to humanity today. A single gram of U-235 can release the same energy as 3 metric tons of coal [1]. For technology sectors that use nuclear energy, there is significant interest in continually improving safety and security procedures. In 2021, the United States alone allocated \$19.8 billion to nuclear security [2] and \$1.6 billion on nuclear energy research [3]. While the intention of nuclear security procedures is to prevent disasters from occurring, it is still imperative that disaster mitigation research be pursued as it is inevitable that nuclear accidents will occur at some point.

Nuclear accidents [4], trafficking [5], and weapons proliferation [6] are very real, persistent problems. The tools available for dealing with these issues are limited, and thus the capability to mitigate consequences of these disasters is limited as well. For example, it is widely recognized that the response to the Fukushima disaster in 2011 was inadequate [7], [8]. Information is crucial to an effective disaster response – however, during the Fukushima disaster response, critical information was not available quickly enough. In addition to the “information avoiding behaviors” by key actors [8], there was limited technology available to assess and map radiation levels around the plant until nearly a month later when remote controlled air and ground robots were deployed [9]–[11].

Governments and emergency responders are increasingly interested in using robotics to address these issues. Robotics is a powerful tool for nuclear disaster response management. The data received from robotic surveys can inform evacuation plans, mitigation strategies, locate illicit material, etc—all without endangering the operator. Even as far back as the

Chernobyl disaster in 1986, robots were considered an effective tool in disaster mitigation [12]. There is significant historical precedent for the deployment of robotic systems for nuclear disaster mitigation and surveying of sites with radiation contamination [9]–[12]. Now, aerial robotics represent the cutting edge of nuclear safety for the quick, terrain agnostic maneuvering [13]. Real time algorithms for the detection, localization, and search for radioactive material using lightweight sensors must be developed to further facilitate this technology.

## **1.2 Prior Work**

Missions for nuclear detection robots typically fall into two categories: mapping and Source Term Estimation (STE). Mapping missions usually involve sites with widespread radiation contamination and require the robotic system to characterize the radiation environment over a defined area [14]–[16]. In contrast, radiological STE is the process of localizing and identifying parameters (location, strength, cardinality, etc.) of point sources of radiation [17]–[20]. Such STE missions may, for instance, stem from cases where nuclear materials are stolen or trafficked and must be located by law enforcement. This dissertation presents research regarding both mapping and STE missions.

### 1.2.1 Mapping

There are myriad considerations to the problem of mapping a radiation field (or any scalar field). Is the mapping performed with a single agent [21], [22] or multiple agents [23]? Should optimal coverage paths [24], [25] or information driven paths [21]–[23] be used? Is myopic path planning [21], [22] or non-myopic path planning implemented [23]? There are hundreds of papers written about each of these considerations.

The simplest conceptual approach to obtaining a high-resolution radiation map is to cover the entire area of interest with numerous radiation sensors. Since this is usually not practical, there is significant interest in algorithms and methods to efficiently map large

areas with a fairly small number of detectors mounted on mobile platforms. Numerous studies have been performed over the past several decades exploring the radiation mapping problem using single agents [21], [22] and multiple agents [23]. Prior work has explored the use of optimal coverage methods as well as information-driven search [24], [25]. It is well-known in the radiation sensing community that different vehicle and detector combinations yield maps of differing spatial resolutions. For instance, one can rapidly obtain a map with low spatial resolution using sensors mounted to one or a handful of manned helicopters or fixed-wing aircraft. Such systems typically produce ground maps with resolutions on the order of 500 m [26], [27]. Better spatial resolutions of 1-5 m have been achieved using low-altitude Uninhabited Aerial Vehicle (UAV)s (remote controlled helicopters or multi-rotor drones) [9], [28]. Ground based surveys using human operators or ground vehicles offer even higher resolutions at the cost of increased time-to-map [29], [30].

Combining the resolution of ground-based surveys with the speed of an aerial survey may enable a best-case scenario in which high-resolution maps can be obtained quickly. This may aid in evacuation, rescue, and containment operations. Combined teams of UAVs and Uninhabited Ground Vehicle (UGV)s have been studied previously [31]–[33]. These experiments used the UAV to provide maps and a general search space for the UGV(s). While these experiments locate point sources faster than a UGV team alone, they do not map the entire area with a high resolution. Another approach pursued in [34] used a helicopter with a small tethered UGV that could be deployed for soil samples. While this soil-sampling capability may be important in a disaster response scenario, the work did not consider the mapping capability of a combined UAV-UGV team. A multi agent system using information driven path planning for the UAV segment combined with optimal coverage path planning for the UGV segment combines the strengths of UAVs and UGVs.

### 1.2.2 Source Term Estimation

The STE problem for a single source has been solved by a variety of authors using least-squares regression [35], Kalman filters [36], particle filters [22], [37]–[41], neural networks [42], and other heuristic algorithms [43]. STE for multiple sources can be much more difficult, however. In this case, the difficulty of the problem depends on whether measurements at any given location are only influenced by a single source (non-overlapping sources), or whether measurements at certain locations are influenced by multiple sources (overlapping sources). The degree to which sources overlap depends on their spacing and strengths as well as the obstacle and terrain environments.

The non-overlapping STE problem may be solved by estimating the source parameters independently, one source at a time, by decomposing the problem into multiple single-source STE problems. This was the approach taken by Peterson *et al.* [33] and Chin *et al.* [44]. In [44], the authors introduce the concept of “fusion range,” which limits the spatial range of influence that measurements can have, thereby restricting the method to use in non-overlapping cases only.

Several papers have addressed overlapping STE problems [21], [45]–[50]. In [45] and [46], the authors performed radiation contour mapping using simulated measurements but did not consider attenuation from obstacles or terrain. The work in [47] and [48] used Additive Point Source Localization (APSL) algorithms to reconstruct the 3D positions and activities of multiple gamma-ray point sources ignoring the effects of obstacles. An extension of this research by Bandstra *et al* [49] addressed online determination of attenuation coefficients and source parameters for a single source case. They found that their method relied heavily on knowledge of the geometry of the obstacle beyond just the surface level provided by Light Detection and Ranging (LiDAR). The work in [50] proposed a grid-based particle filter capable of localizing multiple overlapping sources in the presence of obstacle attenuation, but the number of sources had to be known *a priori* to generate the particle grid. This method was experimentally tested for two sources of the same isotope

and two sources of differing isotopes. Measurements from different isotopes were relegated to independent particle filters, and thus the number of particles grew linearly with the number of isotopes considered. In the case of multiple sources of the same isotope, combinations of sources had to be considered, which caused the number of particles to grow exponentially. On the other hand, Ristic *et al.* [21] proposed a particle filter that handles an unknown number of multiple overlapping sources using a fixed number of particles. While the algorithm was shown to exhibit good performance using experimental data, it was only validated with up to two sources and did not include a mechanism to account for attenuation by obstacles or terrain. However, an earlier paper by the same group of researchers [51] addressed the obstacle attenuation problem via ray-casting.

Several challenges and limitations are present in the literature. First, STE for larger numbers of sources (i.e., three to ten) remains a challenging problem due to the ambiguity caused by overlapping sources. Second, for particle filters, which have commonly been applied to STE, scalability problems arise due to the very large number of particles needed to represent hypotheses associated with numerous sources. Additionally, STE in obstacle-rich environments has proven challenging for algorithms that are designed to run in real time. This is because radiation transport modeling must be performed from each hypothesized source to calculate the predicted measurement at a particular location. For this reason, the majority of STE work has focused on a small number of sources (two or less) in mostly obstacle-free environments.

Prior work on radiological STE can be categorized based on the levels of capabilities shown in Table 1.1. The highest level in each category of Table 1.1 represents the pinnacle relaxation of assumptions. An algorithm at that level for all categories would need to know nothing *a priori* to return the parameters for all sources in the environment given a sufficient set of measurements. The challenge of developing such an algorithm is that the search space grows larger and more complex as each guiding assumption is removed. For example, in the single source case with a known isotope and no attenuation, an inference

Table 1.1: Levels of assumptions made for each category

	<b>Attenuation</b>	<b>Source Cardinality</b>	<b>Isotopes</b>
<b>1</b>	none	1	isotope agnostic
<b>2</b>	known parameters	known (2+)	known
<b>3</b>	unknown parameters	unknown	unknown

algorithm must only divine 4 hidden states: 3 spatial and 1 strength. However, relaxing the source cardinality one level to a known number of sources will *at minimum* double the search space.

The relevant references have been grouped in Table 1.2. This table shows at which level each paper sits in for each category. The categories presented are “Attenuation,” “Source Cardinality,” and “Isotopes.” Each of those is broken down into three levels. For context, the ratings of Chapter 3 and Chapter 4 of this dissertation are included as well.

The attenuation category describes the capabilities to perform STE in an unknown environment. Level one means that the environment and attenuation are ignored. Level two means that attenuation is factored in, but the properties of the environment are known *a priori*. Level three indicates that there is no prior knowledge of the environment, and additional sensors such as LiDAR, cameras, or ultrasonic sensors are required. The trade-off between knowledge of the environment and knowledge of the sources is best seen in [49] where they use a method for source localization that has been extended to the multi-source case in [47], but they stay with the single source case to reconstruct the unknown environment and attenuation parameters with the help of LiDAR.

The source cardinality category describes how many sources are present in the environment. Level one is a single source being present (and it is known to be only a single source). Level two is where there could be multiple sources present, but the cardinality of the sources is known *a priori*. Level three is where any number of sources are present (this includes the null and single source case). Level three could be further split into how well algorithms scale with more than a few sources, but this is so far not documented well in the literature. “Scale” in this context refers to the computational complexity of an algorithm.

For example, the time complexity of the work in [50] scales with the number of samples  $n$  as  $O(n)$ . This is the same as most particle filters. However, in that paper, the number of samples scales with the number of sources  $s$  as a binomial combination  $O\binom{k}{s}$  where  $k$  is a constant related to resolution. So the performance of that algorithm would be markedly worse than the research presented in this dissertation even though they fall in the same level in the source cardinality category.

The isotopes category defines the degree to which complex radiation properties were utilized. Level one indicates that no consideration was given to the effects of energy level of the photon. This means attenuation is ignored, detector type is a counting instrument, and effects of background are ignored or approximated. Level two indicates that the isotope being searched for is known *a priori*. This usually indicates that effects of attenuation and properties of a detector were factored in. Level three indicates that the algorithm is able to work with unknown sources. This is distinct from simply not caring about the isotope (isotope agnostic). Algorithms at this level are able to leverage information from the detectors to perform isotopic identification in some way.

Another important consideration with these algorithms is what kind of measurements they have been verified with. The relevant papers are categorized in Table 1.3. The difference between using simulated measurements and experimental measurements is explored in this dissertation. There are many more considerations to be made when using experimental measurements that can sometimes alter the overall performance of an inference algorithm. The difference between using count rate instruments and spectral instruments also affects the overall performance of an algorithm. An algorithm using counting instruments will be fundamentally different as there will be significantly less information available.

### **1.3 Dissertation Overview**

Chapter 2 investigates the use of multiple UAVs, each carrying several deployable and recoverable UGVs, to completely map a given area. The UAVs are not equipped with sensors

Table 1.2: Categorizing references by level of assumptions made

	<b>Attenuation</b>	<b>Source Cardinality</b>	<b>Isotopes</b>
<b>1</b>	[21], [22], [33], [35]–[41], [43], [47], [48]	[22], [33], [35]–[43], [49]	Chapter 3, [21], [22], [35]–[44], [47]–[49]
<b>2</b>	Chapter 3, Chapter 4, [42], [44], [50]	[50]	[33]
<b>3</b>	[49]	Chapter 3, Chapter 4, [21], [44], [47], [48]	Chapter 4, [50]

Table 1.3: Categorizing references by type of measurements used

<b>Simulated Counts</b>	Chapter 3, [37], [39], [41], [42], [44], [48]
<b>Experimental Counts</b>	[21], [22], [35], [36], [38], [40], [47]
<b>Experimental Spectra</b>	Chapter 4, [33], [43], [49], [50]

themselves, but rather serve as carriers to transport the UGVs to locations of interest. The location into which the UAVs deploy the UGVs is selected through Information Theoretic Path Planning (ITPP). Once the UGVs are deployed, they use an optimal coverage method to map their assigned region. Simulation results measure the time required to achieve a certain mapping accuracy (measured by the integrated squared difference, or Integrated Squared Difference (ISD), between the actual and estimated map), and compare the mapping rate between the UAV-UGV team using ITPP, the UAV-UGV team using random sampling, and a team of UGVs only. Overall, results show that the multi-agent UAV-UGV team combines the unique strengths of UAV and UGV agents. Specifically, the UAV-UGV team produces a map with faster ISD reduction than can be achieved with UGVs alone, and with higher resolution than can be achieved with UAVs alone.

Chapter 3 introduces a novel particle filtering algorithm called the Dynamic Discrete Particle Filter (DDPF). The goal of the algorithm is to find point-sources of gamma radiation in a defined search area that may include obstacles and terrain features. The number, location, and strengths of the sources is assumed to be unknown. The particle filter is designed to fuse multiple count measurements from Geiger-Müller counters, which may be fielded by personnel or mounted on vehicles. The DDPF improves upon existing particle filters for STE (such as that in [21]) by including two unique elements designed to

improve runtime performance in cases with multiple sources and attenuation from terrain and obstacles. First, there is a discrete pre-processing step in which attenuation kernels are computed from discrete source locations to discrete measurement locations. Since attenuation calculations can be computationally intensive, this pre-processing step enables heavy computation to be done offline prior to the search, with results stored in a database for use in online processing of measurements. Such a decomposition allows the online portion of the algorithm to run on computationally-lightweight hardware (e.g., embedded computers) while also incorporating modeling results from complex 3-D radiation transport codes [52], [53]. The second innovation is the dynamic adjustment of the number of particles and their distribution over particle sets associated with different source cardinalities. This dynamic modification of the number and allocation of particles, modeled on the algorithm proposed in [54], allows the algorithm to handle scenarios with a large number of sources efficiently. The DDPF is tested primarily in Monte Carlo trials using simulated count data. The performance is also evaluated on two trials using experimentally gathered count rate measurements.

In Chapter 4, an algorithm is presented that ranks at the most challenging level in all categories of Table 1.2 except for attenuation modeling. Chapter 4 improves upon Chapter 3 with the addition of a gamma spectrum unfolding algorithm. This allows for isotopic identification and accurate attenuation modeling for different isotopes. To leverage this capability, a mixing algorithm is incorporated to combine the outputs of multiple particle filters running in parallel. Each particle filter in this parallel group corresponds to a particular isotope of interest and has an associated attenuation model of the environment. This allows for the favorable performance with an unknown number of the same isotope as shown in Chapter 3, but now the capabilities are extended to an unknown set of isotopes.

Chapter 3 also only dealt with simulated radiation count data. In contrast, Chapter 4 introduces improvements to handle experimental spectral data. Alongside the unfolding algorithm, which allows the use of spectral measurements, is the introduction of the loga-

rithmic domain. Key elements of the core particle filtering algorithm have been transformed into the log domain. This allows for a sparser set of particles to remain useful when ingesting low SNR, low accuracy measurements. Overall stability is improved in the log domain; several edge cases and degeneracy issues are removed when compared to the linear domain particle filters.

## CHAPTER 2

### HETEROGENEOUS MULTI AGENT SEARCH

#### 2.1 Problem Definition

The goal of this chapter is to investigate the use of UAV-UGV teams to map an area contaminated by an arbitrary number of radioactive point sources. The point sources are modeled using basic Poisson statistics ignoring obstacle attenuation. The map of the radiation field is generated via a GPR, also referred to as Kriging, model. The UAVs and UGVs that make up the team have different purposes to leverage their unique strengths. The UGVs are fitted with a simple Geiger-Müller counter. They are responsible for taking measurements of the radiation field because there is no energy cost for them to sit stationary and record data (dwell time). The UAVs are responsible for transporting the UGVs and have no sensors of their own. This is because dwell-time and payload constraints limit the quality of sensing that can be conducted from the UAVs, especially in comparison with UGVs which travel slowly and can dwell for long periods. The UAVs can carry multiple UGVs at a time and can both deploy and recover the UGVs. Future work may investigate the benefit of outfitting the UAVs with sensors as well to facilitate the mapping process.

#### 2.2 Measurement and Radiation Field Modeling

##### 2.2.1 Sensor Modeling

Particle counts due to nuclear decay can be modeled with Poisson statistics [21], [36], [55]. The mean radiation count  $\lambda$  due to a source  $s$  at a 3D coordinate  $(x^s, y^s, h^s)$  at time  $t$  by a Geiger counter located at the coordinates  $(x, y, h)$  is given by Equation (2.1):

$$\lambda_t = \tau_t \left[ \mu_b + \sum_{s=1}^r \frac{\varphi^s}{(d_t)^2} e^{-\beta d_t} \right] \quad (2.1)$$

where  $\tau_t$  is the exposure time interval,  $\mu_b$  is the average count rate due to background radiation,  $r$  is the total number of sources in the environment,  $\varphi^s$  is the equivalent strength of source  $s$ ,  $\beta$  is the air absorption coefficient, and  $d_t$  is the Euclidean distance from the Geiger-Müller counter to the source at time  $t$ . The counts,  $\lambda$ , are also limited by a saturation value,  $\lambda_{sat}$ . The saturation value is the maximum number of counts the meter can report given the dead/recovery time. This time is usually on the order of a few hundred microseconds, although lower dead times have been achieved [56]. A zero mean noise,  $v_t$ , of the form  $\mathcal{N}(0, K)$  with covariance  $K$ , is added to the mean radiation count,  $\lambda_t$ . This addition and the saturation value modification yield a simulated measurement at time  $t$  of the form  $z_t = \max(\text{round}(\lambda_t + v_t), \lambda_{sat})$ . Note that all measurements for this chapter are assumed to take place on a plane, and thus the  $h$  axis will be neglected.

### 2.2.2 Gaussian Process Radiation Field Modeling

Most mapping methods rely on some type of interpolation between measured data points. It is also possible to generate a map based on source parameters from STE using a model of how sources behave in the environment. This is called “inverse mapping.” The spatial distribution of the measurements, the accuracy of the measurements, and the method of interpolation determines the map’s utility. Gaussian Process Regression is a method of interpolation that, under certain conditions, yields the best linear unbiased prediction at unsampled locations [57]. GPR also provides prediction intervals that can be used as a measure of uncertainty. This characteristic is particularly important for information-driven search algorithms. Numerous authors have used GPR representations to build maps of radiation fields [15], [58], [59]. GPR algorithms for informative sampling that are designed to be updated online as measurements are gathered have also been documented [60], [61].

The GPR model used in this work is implemented in the MATLAB Statistics and Machine Learning Toolbox [62]. The measurements,  $w$ , at a point in space  $(x, y)$  are used as training data for the model, and the estimated particle counts,  $z_{gpr}$ , at any sample ref-

erence point  $(x_{gpr}, y_{gpr})$ , can be sampled from the continuous distribution GPR. Although effective from a modeling perspective, this method is not computationally efficient and modifications to this formulation of GPR are being explored.

This toolbox generates a GPR model of the vector form given in Equation (2.2), where  $x$  is the input variable,  $y$  is the response variable,  $f(x) \sim GP(0, k(x, x'))$ , i.e.  $f(x)$  is sampled from a zero mean GP with covariance function  $k(x, x')$ ,  $h(x)$  are a set of basis functions that transform the original vector  $x$  into a new feature vector  $h(x)$  in  $\mathbb{R}^d$ , and  $B$  is a  $p$ -by-1 vector of basis function coefficients.

$$\mathbf{P}(y \mid f, X) \sim N(y \mid HB + f, \sigma^2 \text{eye}) \quad (2.2)$$

$$X = \begin{bmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_n^T \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix},$$

$$H = \begin{bmatrix} h(x_1^T) \\ h(x_2^T) \\ \vdots \\ h(x_n^T) \end{bmatrix}, \quad f = \begin{bmatrix} (x_1) \\ (x_2) \\ \vdots \\ (x_n) \end{bmatrix}$$

Using a GPR model as the agents explore the space gives a map that can be used while the search is still in progress. A GPR model is most certain at measured values, and its certainty would fall off further from measurements. A GPR cannot extrapolate as there is no radiation model encoded into the GPR; it is non-parametric. The GPR model will also not return true source locations or parameters. A particle filter is used in later chapters on

the measurements to give true source locations and parameters [41].

### 2.2.3 Integrated Squared Difference Error Metric

A primary goal of this work is to evaluate the accuracy of the map produced by the multi-agent team. While numerous potential metrics may be used, in this paper mapping error is quantified using the ISD between the GPR approximation of the radiation field and the actual radiation field. A discrete approximation to the ISD is provided in Equation (2.3). In a grid of  $n_1 \times n_2$  locations (denoted by coordinates  $(x_i, y_j)$ ), the difference between the GPR model predicted counts and the true reference model counts are squared and then summed for all locations in the grid. The reference counts are given by taking a zero noise measurement using the radiation measurement model,  $z_{ref}(x_i, y_j) = \text{round}(\lambda(x_i, y_j))$ .

$$ISD = \left[ \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} [w_{gpr}(x_i, y_j) - w_{ref}(x_i, y_j)]^2 \right] \frac{1}{n_1 n_2} \quad (2.3)$$

## **2.3 Path Planning Algorithms**

The two types of robots that make up the team of agents in this work – UAVs and UGVs – serve very different roles in the scenario envisioned here. The UAVs are responsible for selecting the subregion that the UGVs will map next, while the UGVs need to map the assigned subregion as efficiently as possible. Thus, the UAVs use an information-driven search approach, while the UGVs use a coverage path planning algorithm to optimally map the assigned subregion. Note that for this chapter, the region to be mapped is assumed to be rectangular, flat, and obstacle-free. Later chapters relax these assumptions to consider more realistic terrain and obstacle environments.

### 2.3.1 Information Theoretic Path Planning

Various methods of quantifying information gain are available for an information-driven search. Commonly used methods include Fisher information [41], entropy [63], Kullback-

Leibler (KL) divergence [64], and Rényi divergence [21], [64]. Gaussian Process Regression for ITPP of multi-vehicle autonomous teams has been explored previously [60], [61] with favorable results. The measure of information gain used in this paper is the prediction interval given by the GPR model. The prediction interval can be thought of as a measure of uncertainty in the model. Areas where the prediction interval, i.e. uncertainty, are high correspond to areas that have few or noisy measurements. Decreasing this uncertainty leads to a higher-quality model while simultaneously incentivizing coverage of the environment.

The information theoretic path planner in this paper seeks to maximize a reward function along a path for multiple agents. A simple myopic path planner is used to select the next subregion that the UAVs will search. Prior to the mapping process, the total area to be mapped is decomposed into a set of  $N_s$  subregions. These subregions are labeled  $R_i$  where  $i = \{1, \dots, N_s\}$ . The goal of the UAV planner is simply to select the next subregion in which to deploy its UGVs. Thus, at the beginning of the scenario the UAVs make initial decisions about which subregion to visit. Once mapping is completed in that subregion, they recover their UGVs and select a new subregion. This region selection and mapping process continues until there are no more subregions to be mapped.

The subregion selection process proceeds as follows. Consider the  $k^{th}$  UAV. Let the current region being mapped by UAV  $k$  be denoted as  $R^{(k)}$ , and let the next subregion selected for mapping by UAV  $k$  be denoted as  $R^{(k*)}$ . At the current time, the prediction interval in each subregion is computed from the GPR. To compute the prediction interval, the average upper and lower 95% confidence intervals are calculated over each subregion, denoted as  $u_i$  and  $l_i$ , respectively. Then, define a reward function as,

$$C(R_i) = Q(u_i - l_i) - Pd(R^{(k)} - R_i) \quad (2.4)$$

where  $d(R^{(k)} - R_i)$  is the Euclidean distance between the centroids of regions  $R^{(k)}$  and  $R_i$ . In Equation (2.4),  $Q$  and  $P$  are tunable positive constants. The  $k^{th}$  UAV then selects the

next subregion to map as,

$$R^{(k^*)} = \arg \max_{R_i \in \mathcal{R}} C(R_i) \quad (2.5)$$

where  $\mathcal{R}$  is the set of subregions that have not been mapped.

The  $Q$  and  $P$  values in Equation (2.4) capture the tradeoff between prioritizing areas with high uncertainty and penalizing areas that are far from the current UAV location. If  $P$  is very high compared to  $Q$ , the UAVs will simply select the nearest available subregion to search, regardless of the degree of uncertainty in the radiation field. In the limit that  $P$  is set to zero, the UAVs may expend significantly more energy than necessary to map the area by flying back and forth across the domain repeatedly. In practice, these constants need to be tuned based on the specific scenario and the degree to which battery limitations constrain the flight times of the UAVs. Robust tuning of these parameters would require the development of an outer-loop optimizer. The parameters selected for the simulations in this work were tuned through numerous trials on a trial and error basis.

### 2.3.2 Coverage Path Planning

There are a plethora of approaches to coverage path planning [25]; however, for the flat and obstacle-free area considered here the simplest optimal path is given by a Boustrophedon cellular decomposition [65]. The Boustrophedon path (translated as “way of the ox”) is a back and forth pattern that traverses the space. Other space-filling curves, such as the Peano curve or Hilbert curve, can cover the space in the same distance travelled, but, subjectively, are not as simple as the Boustrophedon path. Given a rectilinear grid of search points generated by some metric such as effective sensor radius, the Boustrophedon path can be generated to visit and take a measurement at each point. This is the job of the UGVs which are assigned to map a subregion of the domain. They further divide this subregion between the UGVs that are deployed to the area by a UAV. Each UGV then executes a Boustrophedon path in its assigned area of the subregion. Upon completion of the mapping task, the UGVs rendezvous for pickup by the UAV, to be transported to the next subregion.

## 2.4 Simulation Experimental Setup

### 2.4.1 Setup

A Monte Carlo experiment was performed consisting of 1,000 simulations. Each individual simulation took place in a 100 m by 100 m search area with three randomly-generated point sources. In the figures below, these point sources are labeled as red dots. The sources have an equivalent strength that is randomly-generated between 500 and 2,500 counts per second. The UAVs are assumed to fly at a speed of 5 m/s, and the UGVs drive at a speed of 0.5 m/s. The deployment of UGVs from a UAV takes 5 sec, and the loading of UGVs onto UAVs takes 10 sec. Measurements take 20 sec at a given point. These values can be tuned to compare the performance of the UAV-UGV teams with a UGV-only control experiment. The subregion selection algorithm uses values of  $Q = 3$  and  $P = 1$ .

At the beginning of the simulation, three UAVs start at the origin, (0,0), each with three UGVs on-board for a total of nine UGVs. The search area is divided into a grid of 100 subregions ( $N_s = 100$ ) for the UGVs to search once deployed by the UAVs. In order to spread the UAVs around the space initially, they are commanded to fly to starting locations which are determined as the centroids of a centroidal Voronoi tessellation of the domain generated using the number of UAVs. The subregion within which these centroids lie is the initial subregion for each UAV. Upon arrival at the initial subregion, the UGVs are deployed and perform a Boustrophedon search, taking measurements every 1 m. After the UGVs finish taking measurements, they are reloaded onto the UAVs. The next subregions visited by the UAVs are determined by the information-driven planning method described previously, and the UGVs are deployed again. As this cycle repeats, the paths, GPR model, and ISD are recalculated continuously. The simulation proceeds until all of the subregions have been mapped.

Comparison simulations are performed for a UGV-only team, and also for a UAV-UGV team that performs random search area selection rather than ITPP. In the UGV-only

experiments, the entire search area is treated as a single UGV subregion and thus the UGVs are deployed into a Boustrophedon path to cover the entire space. The number of UGVs, density that the UGVs take measurements, the UGV speeds, and all other parameters are the same as they are in the UAV-UGV experiments. For the UAV-UGV team experiments with random search area selection, the next area in which the UGVs are deployed is selected via uniform random sampling rather than the ITPP scheme described above. Comparison against the UGV-only team is designed to isolate the benefit of UAV rapid mobility, while comparison against the UAV-UGV team with random sampling quantifies the benefit of the ITPP scheme.

#### 2.4.2 Individual Example

Figures 2.1-2.4 show an example simulation for the UGV-only team (Figure 2.1) and the UAV-UGV team executing ITPP (Figures 2.2-2.4). The red dots denote the randomly-generated source locations, the blue lines in Figures 2.1 and 2.2 denote the UGV paths, and the red lines in Figure 2.2 denote the UAV paths. Figure 2.3 shows the GPR model at one instant during the simulation, where the 95% confidence bounds are also shown. Figure 2.4 shows a time history of the ISD for the UGV-only and UAV-UGV team. Note the rapid ISD reduction achieved by the UAV-UGV team in comparison to the UGV-only team. The UAV-UGV team reaches an ISD of 12 (representing an 80% reduction) in only 1,211 sec, compared to 2,039 sec for the UGV-only team. This rapid reduction in the ISD, which results in the production of a more accurate radiation map in a shorter amount of time, is the result of being able to transport the UGVs to high-priority search locations quickly combined with an information-driven search technique that prioritizes uncertainty reduction.

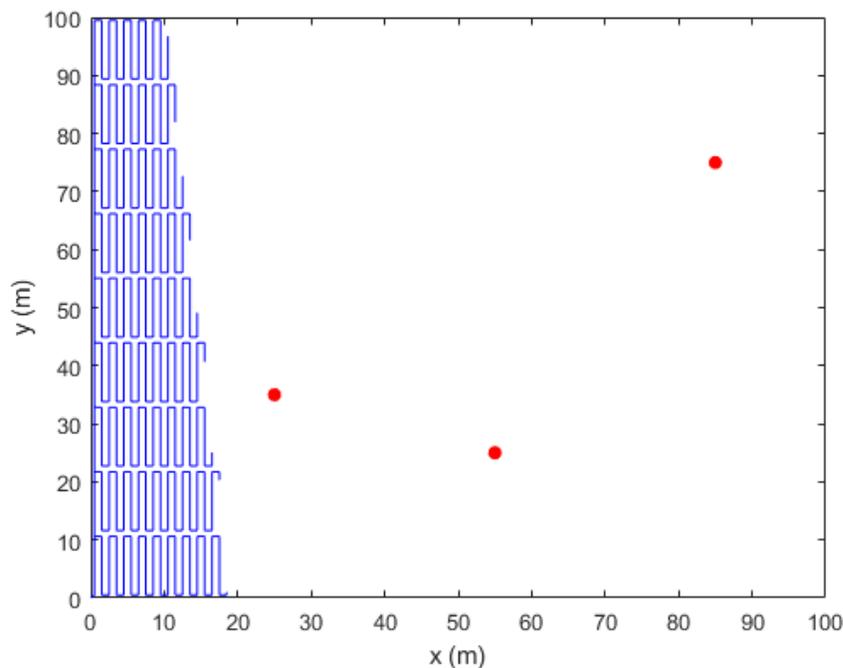


Figure 2.1: UGV-only Search Paths at 435 sec.

## 2.5 Results

For each trial, the mapping problem was solved separately using the UAV-UGV team with ITPP, the UAV-UGV team with random area selection, and the UGV-only team. ISD time histories for each of the UAV-UGV (with ITPP), UAV-UGV (with Random Sampling (RS)) and UGV-only experiments were recorded. Note that these ISD histories can change substantially depending on the source locations, hence the need for Monte Carlo analysis.

The ISD histories for each trial were averaged at each time instant, producing average ISD histories for the UAV-UGV team with ITPP, UAV-UGV team with random selection, and UGV-only team. In addition, the standard deviation of the ISD from these average values was calculated. Figure 2.5 shows the average ISD and +1 standard deviation bounds for this Monte Carlo simulation (−1 standard deviation bounds are not shown in order to reduce clutter). These results show a linear decay rate for the error in the UGV-only experiments. Both UAV-UGV team experiments seem to exhibit an approximately exponential

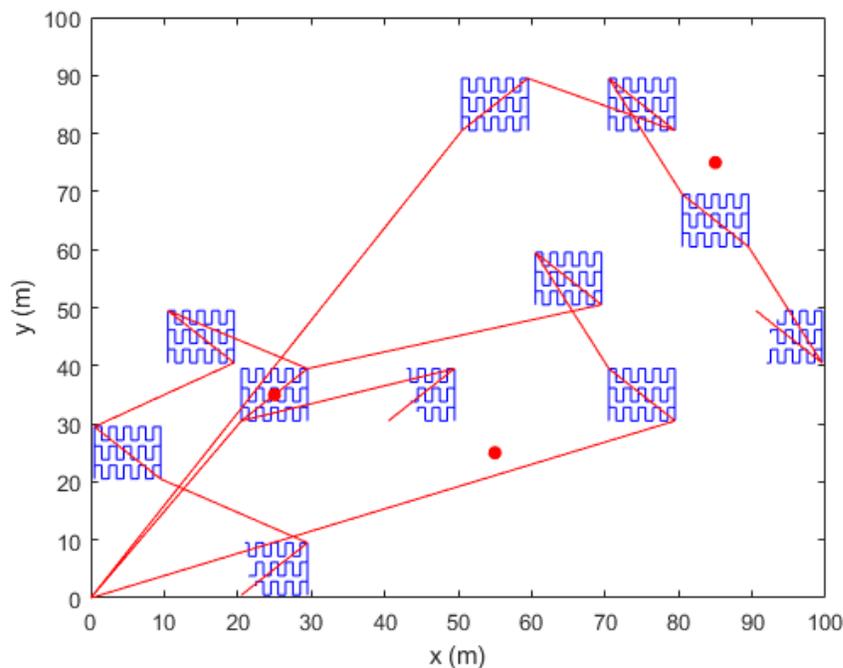


Figure 2.2: UAV-UGV Team Search Paths at 435 sec.

decay rate in the error, but the ITPP results exhibit a noticeably faster decay rate.

On average, the simulations reached an 80% reduction of the ISD in 783 sec for the UAV-UGV team using ITPP, 1,371 sec for the UAV-UGV using random selection, and 1,829 sec for the UGV-only team. An ISD reduction over 95% (this representing total area coverage) was reached in roughly equal times on average between the UAV-UGV team with ITPP (2,483 sec) and the UGV-only team (2,277 sec), although the UAV-UGV team using random selection took noticeably longer on average (3,454 sec). The roughly equal performance of the UGV-only team occurs with respect to this 95% reduction metric because the Boustrophedon path will cover the entire area more quickly as it is an optimal coverage path.

Overall, these results illustrate the benefit of leveraging UAVs as transportation platforms for UGVs tasked with radiological mapping, particularly in instances where it is important to obtain a radiation map as quickly as possible. The benefit of both the UAV mobility concept, as well as the ITPP algorithm, is captured in the average time to 80%

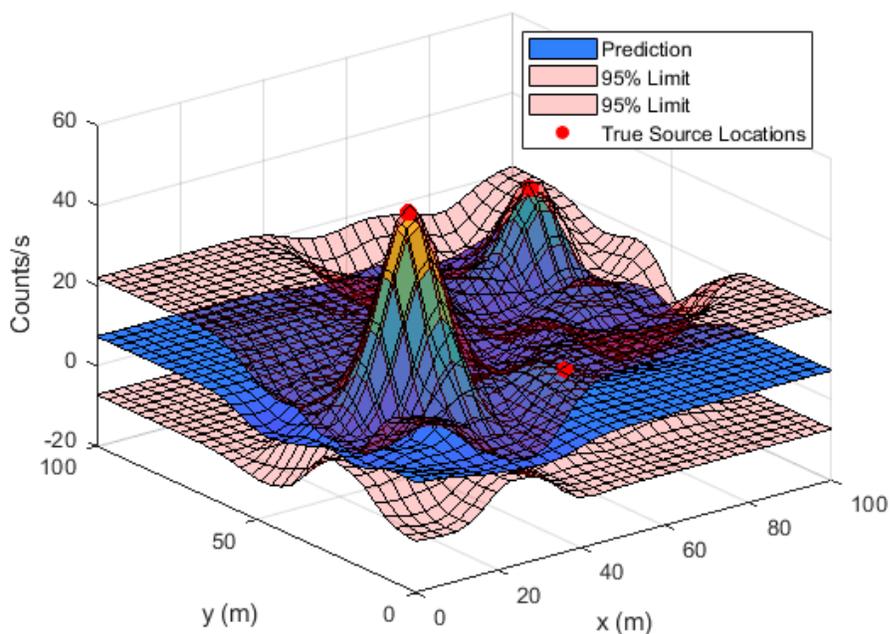


Figure 2.3: GPR Radiation Model at 435 sec.

ISD reduction results. The UAV-UGV case using random sampling improves upon the UGV-only result by an average of 458 sec, isolating the benefit of using UAVs for mobility. Furthermore, the UAV-UGV case using ITPP improves upon the random area selection case by a further 588 sec on average, isolating the benefit of using the information-driven search approach. The combined use of UAVs for mobility and the information-driven search algorithm therefore yield a significant improvement in time-to-map compared to a basic UGV-only scheme. Future work may investigate the benefits of also equipping the UAVs with radiation sensors, as well as an alternative team arrangement in which UGVs act as carrier platforms for teams of deployable and recoverable UAVs that execute the radiation mapping task.

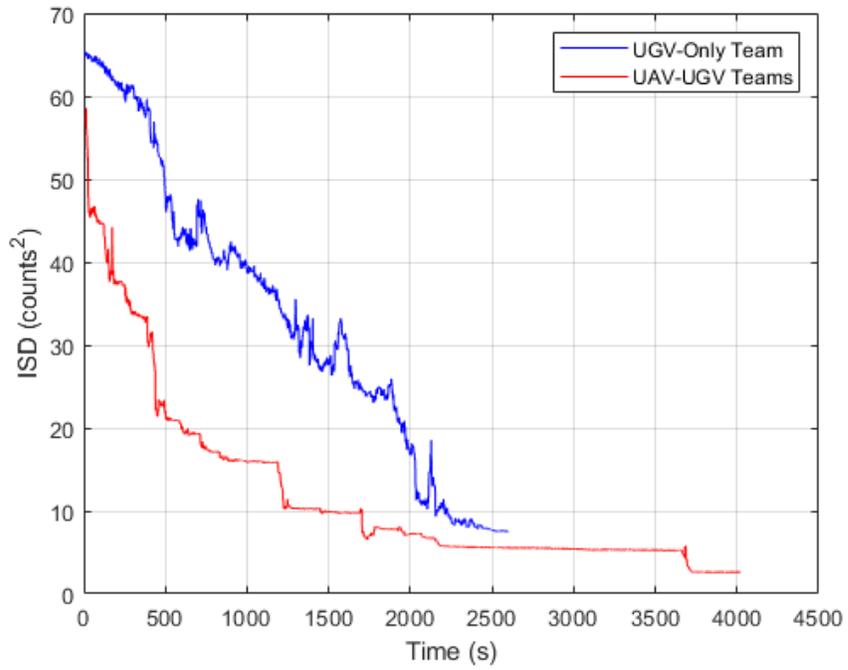


Figure 2.4: ISD vs Time for Example Simulation.

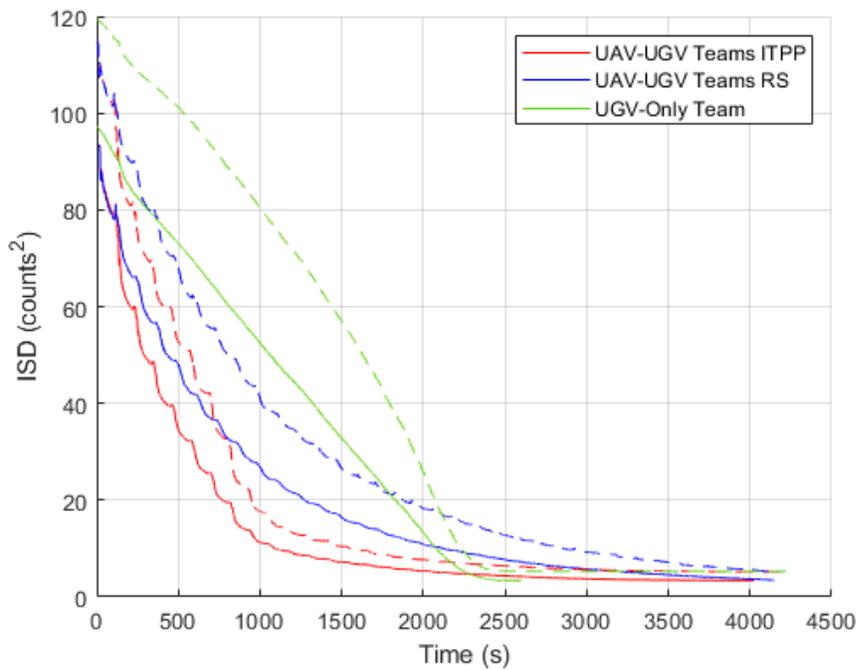


Figure 2.5: Average ISD (solid) and +1 Standard Deviation (dashed) vs time for Monte Carlo study comparing UAV-UGV teams using ITPP and RS with a UGV-only team.

## CHAPTER 3

### DYNAMIC DISCRETE PARTICLE FILTER

#### 3.1 Problem Definition

The goal of this chapter is to introduce a particle filtering algorithm that can characterize an unknown number of point sources of a single known isotope. This algorithm performs source term estimation in contrast to the previous chapter which was concerned more with mapping and coverage. This chapter refines the measurement model for simulated measurements. While the previous chapter used Gaussian noise combined with a saturation threshold to approximate a measurement, this chapter uses a “saturated” Poisson distribution. This is a more realistic approximation of counting measurements.

Consider a search volume  $\mathcal{A}$  containing known obstacles, terrain, and constant background radiation  $\mu_b$ . The coordinate system is defined as  $(x, y, h)$  with respect to a local ground frame, where  $x$  and  $y$  represent Easterly and Northerly directions, respectively, and  $h$  is height above a reference altitude. There are an unknown number  $r_{\text{true}} > 0$  of sources of gamma radiation present in  $\mathcal{A}$ . Point sources are indexed by  $s \in \{1, \dots, r_{\text{true}}\}$ .

Each point source is parameterized by both its spatial coordinates  $(x_s, y_s, h_s) \in \mathcal{A}$  and its equivalent strength  $\varphi_s$ . The equivalent strength as shown in Equation (3.1) accounts for the source’s activity,  $A$ , the detector’s intrinsic efficiency  $\epsilon_{in}$ , and the detector’s geometric efficiency given a cross sectional area  $A_d$  and calibrated distance to the source  $r_d$  [66]. This value is traditionally given at  $r_d = 1$  m. A set of  $K$  measurements, indexed by  $k \in \{1, \dots, K\}$ , are taken in the space and parameterized by spatial coordinates  $(x_k, y_k, h_k) \in \mathcal{A}$  and value  $z_k \in \mathbb{N}$  which represents the total number of counts observed over time interval  $\tau$ .

$$\varphi_s = \frac{A\epsilon_{in}A_d}{4\pi r_d^2} \quad (3.1)$$

The problem statement is then to infer the source parameters  $(x_s, y_s, h_s)$  and  $\varphi_s \forall s$  given the measurements. Furthermore, for the purposes of this work, it is desired that estimates for the source parameters be updated in real time as new measurements are gathered.

As mentioned in Chapter 2, the likelihood of measuring  $z$  counts during time interval  $\tau$  from a source emitting an expected count rate  $\mu$  at a given measurement location is Poisson distributed [21], [36], [55]. Specifically, given the expected number of counts  $\lambda = \mu\tau$ , an idealized measurement  $\zeta$  is drawn from Equation (3.2) where  $\mathcal{P}$  denotes the Poisson distribution. However, due to the effects of saturation, the actual measured value  $z$  is given according to Equation (3.3) which means that  $z$  is not necessarily Poisson distributed and is instead sampled from a saturated Poisson distribution  $\tilde{\mathcal{P}}$  with parameters  $\lambda$  and  $\lambda_{\text{sat}}$  as shown by Equations (3.4) and Equation (3.5),

$$\zeta \sim \mathcal{P}(\lambda) \quad (3.2)$$

$$z = \min(\zeta, \lambda_{\text{sat}}) \quad (3.3)$$

$$z \sim \tilde{\mathcal{P}}(\lambda, \lambda_{\text{sat}}) \quad (3.4)$$

$$\tilde{\mathcal{P}}(z; \lambda, \lambda_{\text{sat}}) = \begin{cases} \mathcal{P}(z; \lambda) & z < \lambda_{\text{sat}} \\ \sum_{z \geq \lambda_{\text{sat}}} \mathcal{P}(z; \lambda) & z = \lambda_{\text{sat}} \\ 0 & z > \lambda_{\text{sat}} \end{cases} \quad (3.5)$$

where  $\lambda_{\text{sat}}$  is the detector saturation value. This is the maximum number of counts the detector can report given its dead/recovery time, which is usually on the order of a few hundred microseconds [56]. As the radiation field propagates through the environment, the expected count rate  $\mu$  is subject to attenuation through the phenomena of absorption and scattering.

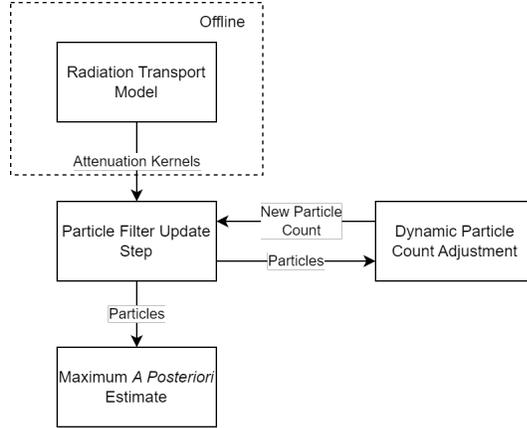


Figure 3.1: Overview of Dynamic Discrete Particle Filter algorithm.

## 3.2 Methodology

### 3.2.1 Overview

The structure of the DDPF proposed in this work is shown in Figure 3.1. Two innovative aspects of the algorithm are highlighted. First, prior to gathering measurements, a radiation transport model is used to compute so-called “attenuation kernels”. This pre-computation is meant to be done offline using a transport model of any desired complexity. As described in Section 3.2.2, the attenuation kernels encode radiation transport modeling from a discrete set of candidate source locations to a discrete set of measurement locations. These attenuation kernels are used online, as measurements are taken, to perform recursive Bayesian estimation via sequential Monte Carlo methods (i.e., particle filtering). During the estimation process, the particle filter uses the Dynamic Particle Count Adjustment (DPCA) algorithm based on [54] to adjust the number of particles in order to balance speed and accuracy. This element comprises the second major innovation of the DDPF design. The current best estimate of the source terms is updated online, as new measurements are gathered, via maximum *a posteriori* estimation applied to the particle set. The various algorithm components are described in detail below with an emphasis on the novel elements.

Prior to describing each algorithm component, several brief definitions are provided.

Since the particle filter is updated after each new measurement, iterations of the algorithm correspond to the current measurement index  $k$ . The particle filter uses a set of  $N$  particles  $\mathbf{X}_k$ , indexed by  $n \in \{1, \dots, N\}$ . A single “particle” at iteration  $k$ , denoted  $\mathbf{X}_k^{(n)}$ , consists of a source parameter matrix  $\boldsymbol{\theta}$  and weight  $w$ , and this particle represents a single hypothesis for the solution of the STE problem posed in Section 3.1. Unlike other particle filters designed for STE [22], [36], [38]–[41], [44], [50], a particle in this work can represent a combination of any number of sources,  $r \in \{1, \dots, r_{\max}\}$ . Note that in this work, the null hypothesis of no sources present ( $r = 0$ ) is excluded (which differs from [21]). The source parameter matrix for each particle is an  $r \times 3$  matrix,  $\boldsymbol{\theta} = (\theta_1, \dots, \theta_r)$ , where each row is a  $1 \times 3$  source parameter vector  $\theta_s = [x_s, y_s, \varphi_s]$  representing an individual source hypothesized by the particle. It is assumed that all sources lie on the terrain surface, and thus the height coordinates of the sources  $h_s = h_s(x_s, y_s)$  are not included as independent parameters.

### 3.2.2 Precomputation of Attenuation Kernels

Let  $\mathbf{X}_k^{(n)}$  be an arbitrary particle at iteration  $k$ , and let measurement  $z_k$  be taken at coordinate  $(x_k, y_k, h_k)$ . Define  $\lambda(\mathbf{X}_k^{(n)})$  as the expected counts observed at location  $(x_k, y_k, h_k)$  due to the sources described by  $\mathbf{X}_k$  as,

$$\lambda(\mathbf{X}_k^{(n)}) = \mu_{k,n} \tau \quad (3.6)$$

where  $\mu_{k,n}$  is the expected count rate due to the arbitrary particle  $\mathbf{X}_k^{(n)}$ . Then the likelihood of observing  $z_k$  given  $\lambda(\mathbf{X}_k^{(n)})$ , including the effects of detector saturation, is given by the saturated Poisson distribution,  $\tilde{\mathcal{P}}$ ,

$$p(z_k | \mathbf{X}_k^{(n)}) = \tilde{\mathcal{P}}\left(z_k; \lambda(\mathbf{X}_k^{(n)}), \lambda_{\text{sat}}\right) \quad (3.7)$$

The particle filter requires that the likelihood in Equation (3.7) be computed at least

$N + NS$  times every time a new measurement is obtained, where  $N$  is the number of particles and  $S$  is the number of progressive correction stages (described in Section 3.2.3). Computing  $\lambda(\mathbf{X}_k^{(n)})$  requires knowledge of the expected count rate  $\mu_{k,n}$  given the source parameters in  $\mathbf{X}_k^{(n)}$ . In settings involving obstacle and terrain occlusions,  $\mu_{k,n}$  must be determined via a radiation transport model. Such models exhibit varying levels of fidelity (e.g., [52], [53]), but in general models that capture complex phenomena such as obstacle absorption or scattering can be so computationally intensive that they are infeasible to run online  $N + NS$  times every time a measurement is obtained.

To address this computational bottleneck, the proposed DDPF replaces online radiation transport modeling with offline transport modeling over a set of hypothesis-measurement pairs. The linear relationship between source strength and expected count rate that the kernels exploit allows the online likelihood calculation step to be reduced to a simple linear algebra operation. First, the search volume  $\mathcal{A}$  is discretized into a set of  $\mathcal{I}$  discrete source locations and  $\mathcal{J}$  discrete measurement locations. Second, a so-called *attenuation kernel*  $\Theta_{i,j}$  is defined between source location  $i \in \mathcal{I}$  and measurement location  $j \in \mathcal{J}$ ; this kernel models the effects of attenuation, reflection, transmission, etc. These sets of locations need not be on regular grids, but are for the results presented in this paper. Then the expected count rate  $\hat{\mu}_{k,n,j}$  due to particle  $n$  at iteration  $k$ , not accounting for background interference, at location  $j$  is given by,

$$\hat{\mu}_{k,n,j} = \sum_{s=1}^{r_{k,n}} \varphi_k^{(n)}(s) \Theta_{i,j} \quad (3.8)$$

In (3.8),  $r_{k,n}$  is the maximum number of sources hypothesized by particle  $\mathbf{X}_k^{(n)}$ , and  $\varphi_k^{(n)}(s)$  is the strength of hypothesized source  $s$  of particle  $\mathbf{X}_k^{(n)}$ . The expected count rate accounting for background radiation is then be obtained as,

$$\mu_{k,n,j} = \mu_b + \hat{\mu}_{k,n,j} \quad (3.9)$$

where  $\mu_b$  is the average count rate due to background radiation.

Each time a new measurement is obtained, the particle filter computes the likelihood associated with all  $N$  particles. This likelihood calculation requires that the expected count rates for all particles be recomputed and compared with the observed count rate. The simplicity with which the expected count rates can be computed using the attenuation kernels makes this process fairly trivial computationally. Define vector  $\Theta_{k,n} = [\Theta_{i(k,n,1),j}, \dots, \Theta_{i(k,n,r_{k,n}),j}]$  where  $i(k,n,s)$  is the source location of source  $s$  of particle  $\mathbf{X}_k^{(n)}$ . Likewise, define vector  $\varphi_{k,n} = [\varphi_k^{(n)}(1), \dots, \varphi_k^{(n)}(r_{k,n})]$ . Then, using Equation (3.8), the expected count rates for all particles can be computed at the  $k^{\text{th}}$  measurement update (assumed to occur at location  $j$ ) through the following  $N$  inner products:

$$\begin{bmatrix} \hat{\mu}_{k,1,j} \\ \vdots \\ \hat{\mu}_{k,N,j} \end{bmatrix} = \begin{bmatrix} \Theta_{k,1} \cdot \varphi_{k,1} \\ \vdots \\ \Theta_{k,N} \cdot \varphi_{k,N} \end{bmatrix} \quad (3.10)$$

Note that Equations (3.8) and (3.10) only make sense if source locations in each hypothesis  $\mathbf{X}_k^{(n)}$  are restricted to lie at locations in  $\mathcal{I}$ . This assumption enables radiation transport modeling to be shifted offline, allowing for arbitrarily complex models to be used in real-time filtering. In practice, the discretization of  $\mathcal{A}$  limits the spatial accuracy that can be achieved by the filter, and thus the resolution of the discretization must be chosen such that the solution for the source locations meets the desired precision requirements. An obvious trade-off exists between the spatial precision that can be achieved and the number of attenuation kernels that must be computed offline and stored. Note that, while the

source locations are converted to discrete variables, the source strengths remain continuous variables.

For purposes of illustration, a simplified radiation transport model as described in [66] and [67] is used in this paper. Given a set of sources,  $s \in \{1, \dots, r\}$ , the expected count rate observed at a particular location  $(x, y, h)$  is given by,

$$\mu = \mu_b + \sum_{s=1}^r \varphi_s \left( \frac{r_d}{d_s} \right)^2 e^{-\beta_{\text{md}} d_s} \quad (3.11)$$

where  $d_s = \sqrt{(x - x_s)^2 + (y - y_s)^2 + (h - h_s)^2}$ . In (3.11),  $\beta_{\text{md}}$  is the absorption coefficient of the medium. The absorption coefficient varies with the gamma-ray energy level, so attenuation kernels will be specific to a given isotope (or set of isotopes with similar energy photopeaks). When using a single set of kernels to model attenuation from multiple isotopes with different energy spectra, some error mitigation may be achieved by taking multiple measurements across the domain. This is explored further in Section 3.4.4.

Since the path from measurement to source may pass through various media, Equation (3.11) is only valid along paths with a constant  $\beta_{\text{md}}$ . One method to handle this is through ray tracing (or casting) [49], [51]. In this work, ray tracing is used to compute the attenuation kernels from a grid of possible source locations to a grid of measurement locations. This process is depicted in Figures 3.2 and 3.3. In Figure 3.2, the source grid is shown with light dots, and the measurement grid is shown with heavy dots. Starting from the source location, a ray is cast to the measurement location and the intersections where the ray passes through obstacles of different absorption coefficients are recorded (dashed lines in Figure 3.3). Using a reference source strength of  $\varphi_s = 1$  at the source location, Equation (3.12) is used to compute a count rate at the first intersection using the  $\beta_{\text{md}}$  value of the first region (initial red line in Figure 3.3 from 0-20 m).

$$\mu_s = \varphi_s \left( \frac{r_d}{d_s} \right)^2 e^{-\beta_{\text{md}} d_s} \quad (3.12)$$

Then, a new equivalent source strength is computed by solving Equation (3.12) using the  $\beta_{\text{md}}$  of the next segment and the count rate at the first intersection. These two steps are repeated consecutively over all segments until the measurement location is reached, resulting in a count rate at the measurement location ( $\mu_{i,j}$ ) given a unit source at the original source location. The attenuation kernel is then simply  $\Theta_{i,j} = \mu_{i,j}$ . While this ray tracing approach is used here to compute the attenuation kernels, any radiation transport model of desired complexity may be used as discussed above.

The ray tracing approach described in this section requires knowledge of obstacle boundaries. In some scenarios, obstacle locations may be known precisely from detailed site surveys, while in others data may be obtained from Geographic Information System (GIS) databases such as OpenStreetMaps (OSM) [68] or via LiDAR measurements of the environment.

### *Ray Tracing Optimizations*

Because Particle Filter (PF)s test thousands of source hypotheses, we must find the intersections of thousands of rays with many polyhedrons each time the likelihood of a particle is computed. This is a large computational burden. We use two methods to counter this. First, obstacles are assumed to be prisms in this work because they are buildings. This allows us to cull buildings if their 2D cross section and a projection of the ray do not intersect. Second, obstacle intersections do not need to be checked if they do not lie in the path of the ray; this is similar to the computer graphics technique, Viewing-Frustum Culling (VFC) [69].

We perform an analogous technique to VFC. For each obstacle, the minimum and maxi-

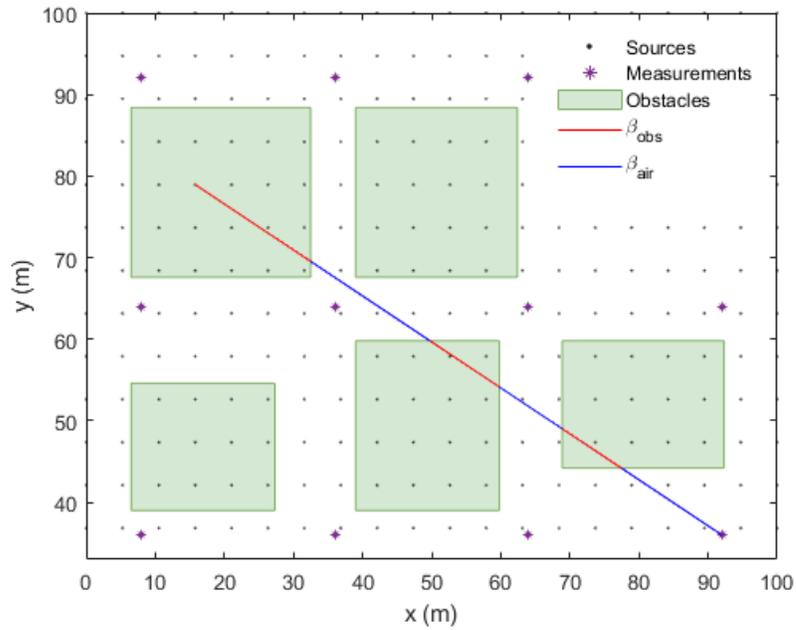


Figure 3.2: Example of discretized ray tracing over source and measurement grids.

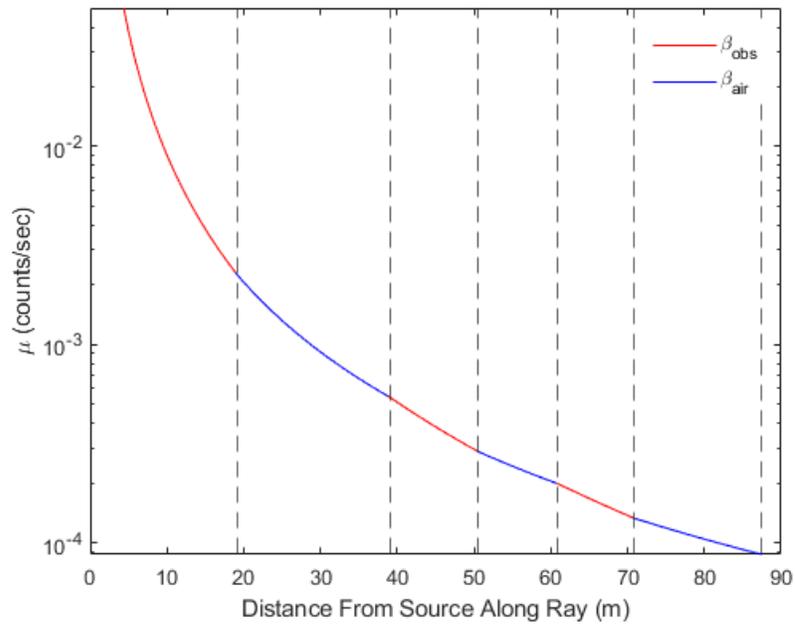


Figure 3.3: Example of count rate values computed along ray.

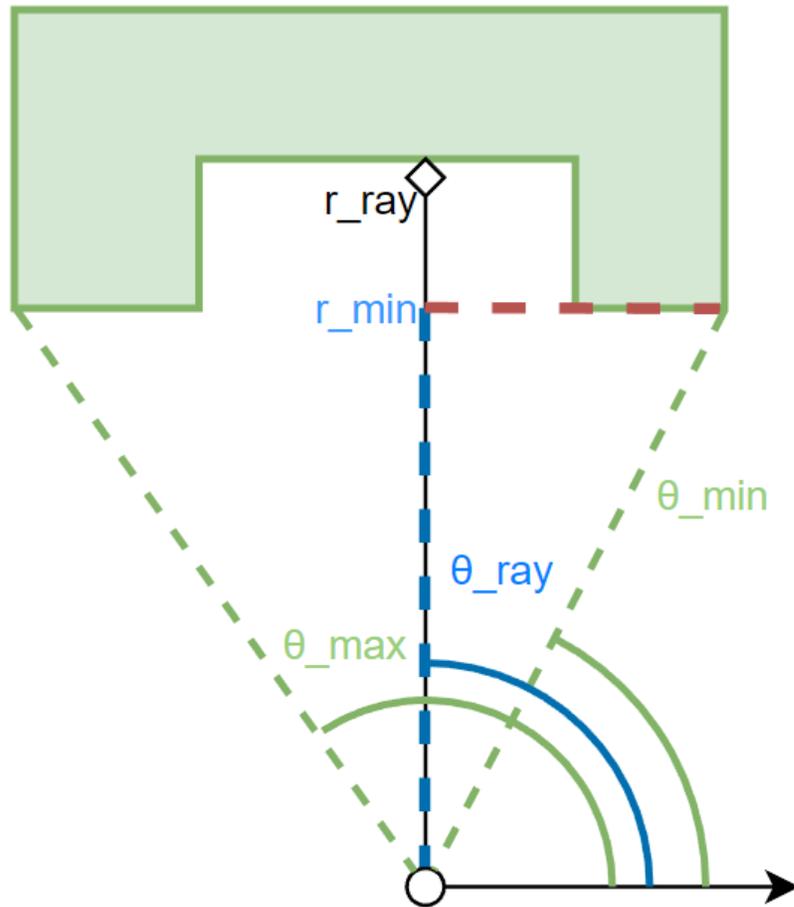


Figure 3.4: Illustration of obstacle culling parameters.

mum azimuth ( $\psi_{min}, \psi_{max}$ ) and smallest projected distance  $\rho_{min}$  of each vertex is found relative to a source location. The ray of interest from source to measurement is also expressed in polar coordinates  $(\rho_{ray}, \psi_{ray})$ . Candidate intersection obstacles satisfy  $\rho_{min} < \rho_{ray}$  and  $\psi_{min} \leq \psi_{ray} \leq \psi_{max}$ . This is shown in Figure 3.4 where the filled shape is the obstacle, the white circle is the source, and the white diamond is the measurement. This example is an edge case—this obstacle would not be culled even though doesn't intersect the ray. This is acceptable considering the alternative where obstacles that intersect the ray are culled.

### 3.2.3 Particle Filter Update Step

---

#### Algorithm 1 Particle Filter Update Step

---

```

1: function  $\mathbf{X}_k = \text{UPDATE}(\mathbf{X}_{k-1}, z)$ 
2:    $X = \mathbf{X}_{k-1}$ 
3:   for  $r = 1, \dots, r_{\max}$  do ▷ Begin Progressive Correction (PC)
4:      $m = 0$ 
5:     for  $n = 1, \dots, N$  do
6:       if  $\text{size}(X, 2) = r$  then
7:          $m = m + 1$ 
8:          $I^{(m)} = n$ 
9:          $Y^{(m)} = X^{(n)}$ 
10:      end if
11:    end for
12:    Select coefficients  $\gamma_s$  s.t.  $\sum_s \gamma_s = 1$ 
13:    for  $s = 1, \dots, S$  do
14:       $Mr = m$ 
15:      for  $m = 1, \dots, Mr$  do
16:         $w^{(m)} = C * \text{Likelihood}(X^{(m)}, z)^{\gamma_s}$ 
17:      end for
18:      Resampling:  $Y = \text{Resample}(Y, w)$ 
19:      Regularization:  $Y = \text{RegularizeParams}(Y)$ 
20:    end for
21:    for  $m = 1, \dots, Mr$  do
22:       $X(I^{(m)}) = Y^{(m)}$ 
23:    end for
24:  end for
25:  for  $n = 1, \dots, N$  do ▷ Begin FF
26:     $w^{(n)} = C * \text{Likelihood}(X^{(n)}, z)$ 
27:  end for
28:  Resampling:  $X = \text{Resample}(X, w)$ 
29:  Regularize R:  $X = \text{RegularizeR}(X)$ 
30:   $\mathbf{X}_k = X$ 
31: end function
32: function  $w = \text{LIKELIHOOD}(X, z)$ 
33:    $w_{last} = 1$ 
34:   for  $k = 1:K$  do
35:      $w = w_{last} * p(z_k | X)$ 
36:      $w_{last} = w$ 
37:   end for
38: end function

```

---

#### Initialization

The filter is initialized by sampling  $N$  particles. Each particle has an associated source cardinality  $r$ . In [21], the  $r$  values are sampled uniformly from the integers in the set  $\{1, \dots, r_{\max}\}$ . While this correctly models the fact that the source cardinality is (pre-

sumably) unknown *a priori*, as the number of sources grows, the spatial coverage of the particle set becomes very poor in high dimensions. For example, 100 particles may be adequate to represent possible source locations in a 2-dimensional space, but 100 particles in a 6-dimensional space results in extremely sparse coverage of the spatial domain. Better results were obtained by initializing the particle set such that the number of particles allocated to higher dimensions grows exponentially with  $r$ . Specifically,  $r$  is sampled from a probability distribution wherein the probability of selecting  $r = r'$  is given by

$$P(r = r') = p^{r'} \quad (3.13)$$

where  $r' \in \{1, \dots, r_{\max}\}$  and  $p$  is the solution to

$$1 = \sum_{r=r_{\min}}^{r_{\max}} p^r \quad (3.14)$$

The maximum source cardinality  $r_{\max}$  is chosen by the operator based on external information about the number of possible sources for which to search. Given  $N$  particles, each with a sampled  $r$  value, the spatial coordinates of each source for each particle may be sampled from uniform distributions dictated by the boundaries of  $\mathcal{A}$ . Likewise, the source strengths may be sampled from a uniform distribution within selected minimum and maximum values. Note that additional *a priori* information may of course be used to change the initial sampling approach as desired in a specific scenario.

### *Likelihood Calculation*

Algorithm 1 shows the particle filter update (i.e., Bayesian inference) step, which is executed every time a new measurement  $z_k$  is obtained. The algorithm takes as input all past measurements  $z$  and the source parameter hypotheses at the previous update step, denoted  $\mathbf{X}_{k-1}$ . The update step outputs an updated set of hypotheses (particles) given the current measurement. There are three critical operations in the update step: likelihood computa-

tion, resampling, and regularization.

The likelihood computation steps are shown in Algorithm 1, line 32. This is accomplished using Equation (3.7) which shows that the probability of a measurement given a source parameter array  $\mathbf{X}$  is given by sampling a saturated Poisson distribution with parameter  $\lambda$  found by using Equation (3.6). This computation is not just performed on the current measurement but on all past measurements for a given particle. Since each measurement is independent of the others, the probabilities of each measurement resulting from a given particle can be cumulatively multiplied together. This returns the probability that every measurement could have come from the given particle. The reasoning for using all the measurements (prior and current) is explained further in Section 3.2.3. The particle weights are found by normalizing the probabilities of each particle by a constant,  $C$ , such that the weights sum to one.

### *Resampling Algorithm*

The second critical operation is resampling as shown in Algorithm 1, lines 18 and 28. Systematic resampling [70], [71] is used. This process involves computing the cumulative sum vector of the weights and using it to build a vector of indices representing the particles to be resampled.

The vector of the cumulative sum of the weights,  $\Omega$ , has elements given by  $\Omega_n = \sum_{i=1}^{(n)} w_i$ .  $N$  ordered random numbers,  $u_n$ , are generated according to Equation (3.15). For each  $u_n$ , the first index,  $i$ , where  $u_n > \Omega_i$  is found and recorded. This makes an  $n \times 1$  vector of non-unique indices where each unique value represents a particle to be resampled.

$$u_n = \frac{(n-1) + \tilde{u}}{N}, \quad \tilde{u} \sim \mathcal{U}[0, 1), \quad n = \{1, \dots, N\} \quad (3.15)$$

### *Regularization*

The final major element of the update step is regularization (lines 19 and 29 of Algorithm 1). In line 19, the parameters themselves are regularized. Each parameter,  $x_s$ ,  $y_s$ , and  $\varphi_s$  is perturbed by a Gaussian kernel,  $\nu \sim \mathcal{N}(0, \sigma)$ ;  $h_s$  is not directly perturbed as the terrain dictates the mapping from  $(x_s, y_s)$  to  $h_s$ . Here,  $\sigma$  is selected as either  $\sigma_{x,y}$  or  $\sigma_\varphi$ , which are the variances for the spatial coordinates and source strengths respectively. In the discrete PF, the spatial parameters are fixed to a grid of points, and during regularization, the new perturbed coordinates are rounded to the nearest discrete point. Regularization helps counter particle degeneracy [72] by increasing particle coverage. A variance that is too small will result in insufficient coverage of the space for small numbers of particles, while a variance that is too large invalidates the Markov assumption: particles could retread old ground, losing information from past measurements. In this work, a larger variance is selected for greater coverage, and an informed “retraining” step is added to overcome information loss. Informed retraining is a strategy wherein the weights of the particle set are computed based on a selection of prior measurements instead of just the most recent one. In this work, since the set of measurements is limited and the search space is relatively small, all previous measurements are used for retraining. Note that this retraining step is further enabled by the ability to rapidly compute the expected count rates using the attenuation kernels per Equation (3.10) — this step would be far more expensive if complex transport models had to be used online.

Regularization is also performed on the estimated number of sources  $r$  for each particle (line 29 of Algorithm 1). This creates a small probability of increasing or decreasing  $r$  for a particle based on tunable parameters  $p_{up}$  and  $p_{dn}$ , which represent the probability of adding a new source or removing a source from the particle, respectively. A new source is added to a particle by copying a source, selected uniformly at random, from the combined set of all hypothesized sources in the PF. When removing a source, a source is selected uniformly at random from those of a given particle and removed.

### *Progressive Correction*

PC is a stochastic form of the particle flow with log-homotopy designed to reduce particle degeneracy [73]–[75]. More particles survive the correction step, which is the combination of the regularization and resampling steps, during PC because the variance in the likelihood is reduced by the exponent  $\gamma_s$ . Essentially, PC splits the correction step into several smaller substeps. The progressive correction step begins on line 3 of Algorithm 1. Note that the particles are only competing against particles of the same source cardinality, which is in contrast to full field correction in line 25, where all particles compete against all other particles. Normally, PC is a substitute for the correction step. In this case, since PC is performed separately for each source cardinality hypothesis, the full field correction is needed to allow particles of different source cardinalities to compete against each other.

### *Maximum A Posteriori Estimate*

The posterior Probability Density Function (PDF) for all measurements up to step  $k$ ,  $p(\mathbf{X}_k | z_{1:k})$ , is approximated by the empirical distribution of particles  $\{w^{(n)}, \mathbf{X}_k^{(n)}\}_{n=1}^N$ , where  $w^{(n)}$  is the weight of the  $n^{\text{th}}$  particle. The Maximum A Posteriori (MAP) estimate for a source parameter is the mean of the marginal PDF of the parameter—since the weights are uniform after each update step, this simplifies to the mean of the given parameter across the particle set. The first step in estimating the source parameters from the particle set is to find the estimated source cardinality according to

$$\hat{r} = \text{nint} \left( \frac{1}{N} \sum_{n=1}^N |\mathbf{X}_k^{(n)}| \right) \quad (3.16)$$

where  $|\cdot|$  extracts the cardinality, or number of sources, of a particle and  $\text{nint}(\cdot)$  is a function that rounds to the nearest integer. Then, particles for which  $r \neq \hat{r}$  are removed from the particle set and  $k$ -means clustering is performed on the remaining particles where  $k = \hat{r}$ . These clusters represent individual location and source strength hypotheses for each of the

identified point sources. The MAP estimates of  $(x_s, y_s, \varphi_s)$  are then found as the centroid of each cluster. The heights of each source may then be estimated by interpolating a terrain map using the MAP estimates of  $x_s$  and  $y_s$ .

### 3.2.4 Dynamic Particle Count Adjustment Algorithm

The number of particles  $N$  is a tunable parameter with well-known trade-offs between computational burden (high  $N$ ) and inaccurate estimates due to sparse coverage of the domain (low  $N$ ). Elvira *et al.* [54] developed an algorithm to dynamically adapt  $N$  in response to estimates of filter convergence as measurements are gathered. This method was shown to effectively balance the trade-offs between computational burden and performance. In the context of this chapter, the ability to accurately identify large numbers of sources in the environment (e.g., more than five) hinges on use of a sufficiently large number of particles that produce adequate spatial coverage. By dynamically adjusting  $N$ , the filter maintains sufficient coverage when initial measurements are gathered but is not saddled with large numbers of extraneous particles later on, once the particles converge to the likely source parameters. In this and the following chapter, convergence is defined as the degree to which the PF can explain the measurements (e.g., “converged” indicates all measurements could feasibly have come from the PF’s hypotheses).

A Dynamic Particle Count Adjustment (DPCA) algorithm is developed in this work based on [54]. This method, shown in Algorithm 2, is executed after the PF’s update step. All of the measurements taken up to the latest measurement step  $k$  are used. This set of measurements is denoted as  $\mathcal{Z}_{1:k} = \{z_1, \dots, z_k\}$ . Each measurement is paired with a set of fictitious measurements  $\tilde{\mathcal{Z}} = \{\tilde{z}_1, \dots, \tilde{z}_J\}$  drawn from  $J$  particles randomly and uniformly sampled from the PF (line 5) to compute a set of the negative log Poisson likelihoods  $\mathcal{Q} = \{q_1, \dots, q_k\}$  (line 8) that each of the true measurements could have come from the set of fictitious measurements given by the PF. The set is represented by its mean measurement value,  $\bar{z}$ . Using a *negative* log likelihood means that a higher value represents a less

likely source configuration. The maximum value of  $\mathcal{Q}$ ,  $q_{\max}$ , is compared to tunable  $q_h$  and  $q_l$  values, and the particle count is increased or decreased according to tunable functions  $f_h$  or  $f_l$  (Lines 20 and 25). If more particles are added, they are added according to the initialization function used at the start of the PF algorithm (line 21). If the number of particles decreases, they are randomly and uniformly removed (line 26). Tuning  $J$ ,  $q_h$ ,  $q_l$ ,  $f_h$ , and  $f_l$  is required to achieve a suitable balance between runtime and accuracy; for example, a higher  $q_h$  indicates a higher tolerance for unlikely particles, and a more aggressive  $f_h$ —meaning one that adds a larger number of particles when  $q_{\max} > q_h$ —would prioritize accuracy over runtime.

---

**Algorithm 2** Dynamic Particle Count Adjustment algorithm

---

```

1: function  $\mathbf{X}_k = \text{DYNAMIC}(\mathbf{X}_k, \mathcal{Z}_{1:k})$ 
2:   for  $\kappa = 1 : k$  do
3:     for  $j = 1 : J$  do
4:        $i \sim \mathcal{U}\{1, N\}$ 
5:        $\tilde{z}_j \sim p(z_\kappa | \mathbf{X}_k^{(i)})$ 
6:     end for
7:      $\bar{z} = \text{mean}(\{\tilde{z}_1, \dots, \tilde{z}_J\})$ 
8:      $q_\kappa = -\log \mathcal{P}(z_\kappa; \bar{z}, \lambda_{\text{sat}})$ 
9:   end for
10:   $q_{\max} = \max(\mathcal{Q})$ 
11:  if  $q_{\max} > q_h$  then
12:     $X, N = f_{up}(X, N)$ 
13:  end if
14:  if  $q_{\max} < q_l$  then
15:     $X, N = f_{dn}(X, N)$ 
16:  end if
17:   $\mathbf{X}_k = X$ 
18: end function
19: function  $[X, N] = f_{up}(X, N_{old})$ 
20:    $N = f_h(N_{old})$ 
21:    $X_{new} = \text{initialize}(N - N_{old})$ 
22:    $X = X_{new} \cup X$ 
23: end function
24: function  $[X, N] = f_{dn}(X, N_{old})$ 
25:    $N = f_l(N_{old})$ 
26:    $s \in \mathbb{N}, s \in [1, N - N_{old}]$ 
27:    $X = X^{(s)}$ 
28: end function

```

---

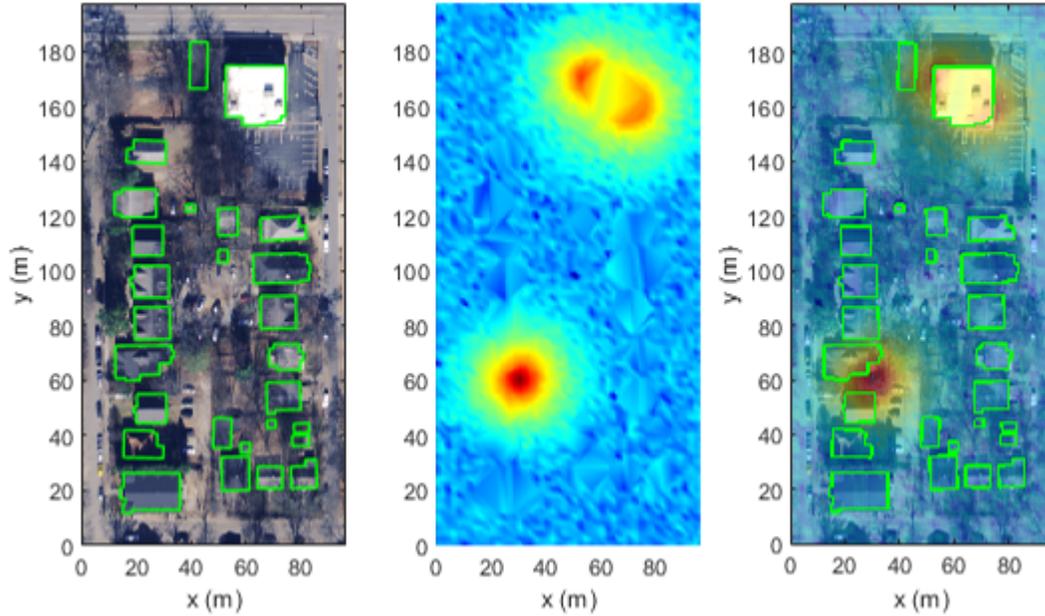


Figure 3.5: Search area for simulation trials. (left) Satellite view of search area with buildings outlined in green. (center) Ray-traced radiation intensity from three example sources (red = high intensity). (right) Combined plot of radiation and environment.

### 3.3 Simulation Experimental Setup

#### 3.3.1 Setup

Monte Carlo simulation results are presented in the next section to demonstrate the performance of the algorithm and to compare it to the particle filter design in [21]. All of the results in this paper use the search area shown in Figure 3.5, which measures  $100\text{m} \times 200\text{m}$ . Building footprints were obtained from OSM [68], and for ease of implementation were treated as solid structures. Furthermore, the terrain in this example location is relatively flat, so for the purposes of this work all terrain and source heights are assumed to be zero. Buildings are modeled as prisms such that the ground footprint extends upward uniformly to the building height. Building height data was taken from OSM, and buildings without height data were assumed to be 4.3 m tall.

The attenuation kernels were calculated across this domain at a grid of 44 measurement locations and 4,900 possible source locations. The number of measurements was

chosen to be as sparse as possible considering the range of possible source strengths and the size of the environment. This is why it is much less than the number of possible source locations, which was chosen to allow for a high-resolution solution. The assumed measurement height was 3 m (simulating, for example, a detector mounted on top of a truck) and measurement locations are restricted to lie outside buildings. The grid of measurement locations is shown in Figure 3.6 by yellow dots. When computing the attenuation kernels, linear absorption coefficients of  $\beta_{\text{obs}} = 1 \times 10^{-2} \text{ m}^{-1}$  and  $\beta_{\text{air}} = 1 \times 10^{-6} \text{ m}^{-1}$  were used for transmission through buildings and air, respectively. The absorption coefficients in this work are scaled down from real-world values for two reasons. First, modeling the buildings as solid structures results in a much longer path length in obstacles than if they were modeled as hollow, so the absorption coefficient is reduced to compensate for the longer path lengths. Second, higher absorption values decrease the radius of the energy deposition field. This means more measurements would be needed to get finer coverage of the area. Reducing the absorption coefficients in the simulation allows fewer measurements to be used, and thus much reduced simulation times. However, this does not actually result in decreased simulation fidelity since the ratio between the gamma ray linear attenuation coefficients of air and brick in [76] is preserved.

In the examples below, simulating measurements requires selecting an exposure time  $\tau$ , which is calculated as follows. First,  $\lambda_{\text{min}}$  is selected representing the minimum acceptable measured count. This value can be selected arbitrarily, or it can be derived by solving Equation (3.17) for  $\lambda_{\text{min}}$  [36]:

$$SNR_{\text{min}} = 10 \log_{10}(\lambda_{\text{min}}/\mu_b) \quad (3.17)$$

where  $SNR_{\text{min}}$  is a design variable representing the minimum acceptable signal-to-noise ratio. Then, given a count rate  $\mu$ ,  $\tau$  is selected as  $\tau = \lambda_{\text{min}}/\mu$ . Finally,  $\tau$  is constrained on

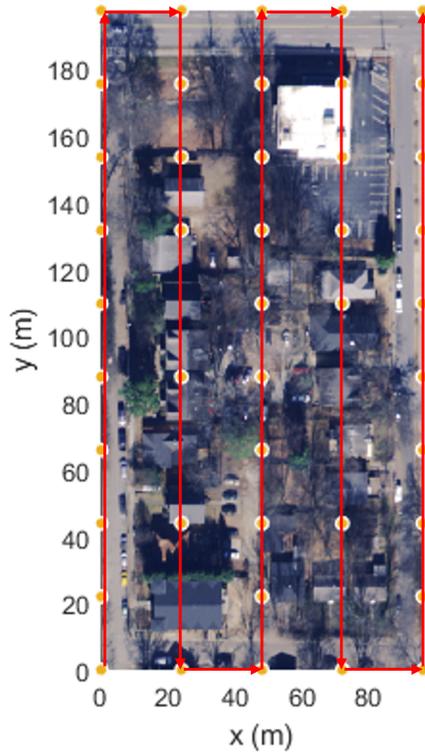


Figure 3.6: Sequence of measurement locations.

the interval  $[1, 60]$  s.

With the attenuation kernels pre-computed and stored, the online portion of the algorithm is executed as measurements are obtained. Measurements are obtained in the sequence shown in Figure 3.6. This sequence was chosen arbitrarily, no measurements fall within an obstacle, and the detector is simply moved to each point without considering any vehicle dynamics. All simulation parameters are listed in Table 3.1.

### 3.3.2 Performance Metrics

Performance of the algorithm is evaluated in two ways: runtime and accuracy. Runtime measures the computation time involved in the online portion of the algorithm only. Accuracy is measured differently depending on the source parameter. Accuracy of the source cardinality estimate is measured as the absolute difference between the true and estimated number of sources.

Table 3.1: Simulation parameters.

Parameter	Value	Parameter	Value
$N$	varies	$r_{\max}$	varies
Number of measurements	44	$r_{\text{true}}$	varies
Possible source location points	4,900	$\varphi_{\min}$ (counts/s)	5,000
$\beta_{\text{obs}}$ $\text{m}^{-1}$	varies	$\varphi_{\max}$ (counts/s)	12,000
$\beta_{\text{air}}$ $\text{m}^{-1}$	$1 \times 10^{-6}$	$S$	20
$\mu_b$ (counts/s)	1	$\sigma_{\text{pos}}$ (m)	3
$\lambda_{\text{sat}}$ (counts/s)	5,000	$\sigma_{\varphi}$ (counts/s)	20
$SNR_{\min}$ (dB)	25	$p_{\text{up}}$	0.003
$\tau_{\max}$ (s)	60	$p_{\text{dn}}$	0.003
$\tau_{\min}$ (s)	1	$q_h$	30
Measurement height (m)	3	$q_l$	10
Total search area width (m)	100	$f_h(N_{\text{old}})$	$50N_{\text{old}}$
Total search area length (m)	200	$f_l(N_{\text{old}})$	$N_{\text{old}}/1.2$

To assess the accuracy of the source location and strength estimates, comparisons are made between the estimated sources and the nearest true source by distance. In cases where the estimated number of sources  $\hat{r}$  is higher or lower than the true number of sources  $r_{\text{true}}$ , the selection of the “nearest” source is not necessarily straightforward. Figure 3.7 shows how this is handled. If the number of sources is overestimated or if  $\hat{r} = r_{\text{true}}$ , the estimated sources are compared against the nearest actual sources (Figure 3.7, left). This means a true source may factor into the error calculations more than once. If the number of sources is underestimated, each true source is compared to the nearest estimated source (Figure 3.7, right). This means an estimated source might contribute to error calculations more than once.

With the “nearest sources” defined, accuracy of the source location estimate is assessed as the sum of the Euclidean distances between the true and estimated spatial parameters according to,

$$\epsilon_{\text{pos}} = \sum_{i \in \mathcal{R}} \Delta d_i \quad (3.18)$$

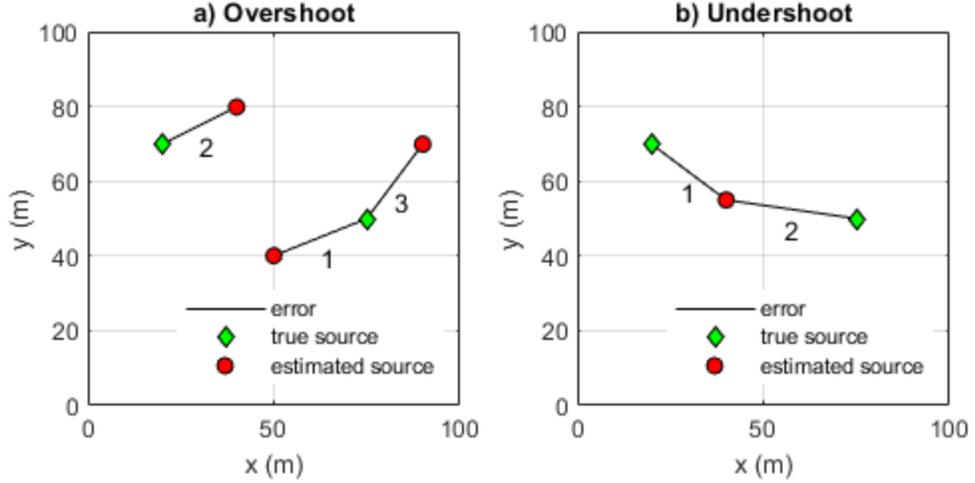


Figure 3.7: Error associations for scenarios in which particle filter overestimates (left) and underestimates (right) source cardinality.

where  $\mathcal{R} = \{1, \dots, \max(\hat{r}, r_{\text{true}})\}$  and  $\Delta d_i$  is the Euclidean distance between the  $i^{\text{th}}$  estimated source and its nearest true source for the  $\hat{r} \geq r_{\text{true}}$  case, or the Euclidean distance between the  $i^{\text{th}}$  true source and its nearest estimated source for the  $\hat{r} < r_{\text{true}}$  case. Likewise, accuracy of the source strength estimate is measured as,

$$\epsilon_{\varphi} = \sum_{i \in \mathcal{R}} \Delta \varphi_i \quad (3.19)$$

where  $\Delta \varphi$  is the absolute value of the difference in source strength between the nearest true and estimated sources.

### 3.3.3 Individual Example

An example simulation is provided to contextualize the Monte Carlo studies presented in the following sections. It is performed using three sources with random locations and source strengths. The locations are shown in Figure 3.8 as green diamonds, and the source strengths (from the source at the maximum  $y$  value to the minimum  $y$  value) are 7,000, 12,000, and 12,000 cps, respectively. Figure 3.8 also shows the hypothesized sources at

various stages of the particle filter operation, from initialization (top left), successive update steps, and the final maximum *a posteriori* estimate (bottom right). Hypothesized sources are shown as colored dots where color indicates the particle’s source cardinality estimate (blue = 1 source, red = 2 sources, yellow = 3 sources). The size of the particles correspond to their likelihood after a measurement update. Measurement locations are shown as small black asterisks. The “Final” plot shows the building footprints as green lines and estimated source parameters as large blue asterisks. The initial particle count is  $N = 5,000$ , and time histories of  $N$  and  $q_{\max}$  as a function of time is shown in Figure 3.9. The values for  $q_h = 30$  and  $q_l = 10$  are shown as dashed lines labeled “Increase” and “Decrease,” respectively. Note that at  $k = 33$  the DPCA algorithm grows the particle set significantly when  $q_{\max}$  exceeds  $q_h$  due to the aggressive  $f_h$  function shown in Table 3.1. This results in a large set of low-likelihood particles scattered across the domain as seen in the bottom center panel of Figure 3.8.

The final MAP results for this example case exhibit excellent accuracy, with  $\hat{r} = 3$  sources,  $\epsilon_{\text{pos}} = 2.3$  m, and  $\epsilon_{\varphi} = 1,463$  cps. The measurement updates, MAP estimation, and DPCA algorithm steps (i.e., the steps performed online) required 32 seconds to run on an AMD Ryzen 5 3600X 4.4 GHz processor. This runtime could be reduced further through GPU parallelization of kernel generation, weight calculations, and DPCA likelihood calculations. Note that this runtime number also excludes the detector dwell time,  $\tau$ , required each time a measurement is obtained, as this is not pertinent to the algorithm runtime itself.

### 3.4 Simulation Results

#### 3.4.1 Attenuation Kernel Study

In the following sections, a series of Monte Carlo simulations is presented to highlight the trade-offs of various components of the proposed algorithm. The first study investigates the effect of problem discretization using the attenuation kernels. Two algorithms are com-

pared. The first is the proposed Discrete Particle Filter (DPF), which uses the discretized formulation presented in Section 3.2.2. The second is the so-called Continuous Particle Filter (CPF), which is based on the algorithm in [21]. This algorithm does not employ any spatial discretization, and instead requires ray tracing to be performed online for each hypothesis at each update step. In order to isolate the effects of discretization only, the DPF in this example does not include the DPCA algorithm; the particle count is fixed at  $N = 500$  instead. This relatively low value of  $N$  is used to keep computation times reasonable for the CPF case.

A Monte Carlo simulation was performed to investigate general performance trends. In all cases, the particle filter was configured with  $r_{\max} = 3$ . Thirty random source configurations were generated. The true number of sources,  $r_{\text{true}}$  used in each scenario was randomly and uniformly selected between 1 and  $r_{\max}$  (since the assumed maximum number of sources is not necessarily equal to the true number of sources). The source locations were then randomly sampled uniformly across the search area, and the strengths were randomized uniformly between 5,000 and 12,000 cps (these bounds were arbitrarily selected and do not correspond to any specific source activity range). For each of the 30 random source configurations, 5 trials were run with the CPF and DPF using different random seeds to generate the initial particle set. This resulted in a total of 150 simulation trials for the CPF and DPF. Furthermore, these 150 trials were repeated with and without the obstacles.

The results of these simulations are shown in Table 3.2. The first row shows the portion of trials in which the source cardinality estimate  $\hat{r}$  is correct. The means and standard deviations of  $\epsilon_{\text{pos}}$  and  $\epsilon_{\varphi}$  are shown as  $\mu(\cdot)$  and  $\sigma(\cdot)$ , respectively. Overall, the accuracy of the CPF and DPF are not significantly different, as expected. However, the CPF exhibits average runtimes that are over two orders of magnitude greater than the DPF, even in the obstacle-free case. Furthermore, because the CPF must compute radiation transport online, its runtime increases with the number of obstacles (as seen by the difference in average CPF runtime between cases with and without obstacles). Because this computation is shifted

Table 3.2: Monte Carlo results comparing CPF and DPF performance ( $N = 500$ ).

<b>Particle Filter Obstacles</b>	<b>CPF</b>		<b>DPF</b>	
	No	Yes	No	Yes
$\hat{r}$ <b>Correct (%)</b>	84	83.33	87.33	86
$\mu(\epsilon_{\text{pos}})$ <b>(m)</b>	4.736	4.494	5.592	5.259
$\sigma(\epsilon_{\text{pos}})$ <b>(m)</b>	4.357	5.121	7.094	6.469
$\mu(\epsilon_{\varphi})$ <b>(counts/s)</b>	736.4	488.8	414.2	427.3
$\sigma(\epsilon_{\varphi})$ <b>(counts/s)</b>	2534	2392	2247	2653
<b>Avg. Runtime (s)</b>	1730	31,910	13.5	12.9
<b>Std. Dev. Runtime (s)</b>	51.08	5070	1.407	1.147

offline for the DPF, its online computation time is independent of the number of obstacles.

While runtime improvements are achieved by the DPF without significant loss of accuracy when viewed from the perspective of mean errors, the DPF is subject to more outlier solutions compared to the CPF. This is due to the low number of particles used in these cases. Figure 3.10 shows a series of violin plots corresponding to each of the configurations run. Each violin contains 150 trials. The left side of the violin is a histogram, and the right side is the Inter-Quartile Range with the white dot as the median. The values plotted in the violin exclude failure cases which are shown as colored points above the violin plots (failed cases are defined as those with performance metrics which are more than three scaled mean absolute deviations from the median). Note that outliers are excluded only for the violin plots. Statistics (mean, median, etc.) in tables or graphics other than the violin plots use every data point. As shown in the figure, a handful of cases for both the CPF and DPF resulted in very high position errors, usually caused by undershooting or overshooting the source cardinality. The DPF resulted in a higher number of these “failure” cases, which is the cause of the slightly higher mean position errors in Table 3.2. Note, however, that the median of the DPF position error is still lower than the CPF—this means that outside of these failure cases, the accuracy of the DPF was actually marginally better.

### 3.4.2 Particle Count Study

The occasional poor solutions in the previous Monte Carlo study are caused by the use of a relatively low number of particles. A study was performed to examine how particle count affects errors in the solution. In these cases, the same 150 simulations (30 source configurations with 5 random seeds) were run with obstacles using the DPF only. These simulations were run with varying values of  $N$ , with results shown in Table 3.3 and Figure 3.11. In Figure 3.11, the solid line indicates the mean. As expected, higher  $N$  leads to improved accuracy. The rate at which  $\hat{r}$  is correct increases monotonically with  $N$ . This in turn reduces the number of “failed” solutions (i.e., cases exhibiting unusually high position errors) as  $N$  grows, as seen in Figure 3.11. However, increasing  $N$  shows diminishing returns in terms of improving accuracy, especially when measured against the runtime penalty when using a larger particle set as shown in Table 3.3. These diminishing accuracy returns are due to measurement sparsity. With sparse measurements, there can be many parameter vectors with high likelihoods between which the PF cannot distinguish. Therefore, increasing the number of particles without increasing the density of measurements (particularly in certain areas) will not necessarily improve estimates. An illustrative example of this can be seen in Figure 3.5. In the top right of the three subplots, two sources lie close together inside an obstacle. The radiation field contours are nearly circular outside of that obstacle. Without a measurement inside the obstacle between these sources, the addition of more particles in and of itself may not enable the PF to determine that there are two sources inside the obstacle.

### 3.4.3 Dynamic Particle Count Adjustment Algorithm Study

The prior section illustrates the well-known trade-off between the desire to use a high number of particles for improved accuracy, and the desire to use a low number of particles to reduce runtime. For a practical system that is expected to operate in a range of scenarios with different spatial limits, obstacle densities, etc., it is nearly impossible to choose a

Table 3.3: Monte Carlo results with DPF using varying  $N$ .

<b>Particle Filter</b> <b>Particle Count</b>	<b>DPF</b>					
	<b>100</b>	<b>500</b>	<b>2 500</b>	<b>5,000</b>	<b>25,000</b>	<b>50,000</b>
$\hat{r}$ <b>Correct (%)</b>	78.67	86	92	92.67	94.67	96
$\mu(\epsilon_{\text{pos}})$ <b>(m)</b>	9.457	5.259	3.703	3.579	3.415	3.194
$\sigma(\epsilon_{\text{pos}})$ <b>(m)</b>	13.92	6.469	3.43	3.87	3.076	2.821
$\mu(\epsilon_{\varphi})$ <b>(counts/s)</b>	466.4	427.3	521.2	474.4	186.5	342
$\sigma(\epsilon_{\varphi})$ <b>(counts/s)</b>	3030	2653	2578	2139	1788	1826
<b>Avg. Runtime (s)</b>	7.524	12.95	34.66	76.57	1131	4137
<b>Std. Dev. Runtime (s)</b>	0.5937	1.147	1.918	5.964	168.9	666.6

suitable value of  $N$  that will work well in all scenarios. The DPCA algorithm described in Section 3.2 offers a solution to adapt the particle count in real-time, thereby balancing the accuracy-runtime trade-off automatically for a specific scenario.

To investigate the performance of the DPCA, the DPF, which uses a fixed  $N$ , is augmented with the DPCA as depicted in Figure 3.1. This algorithm is denoted as the Dynamic Discrete Particle Filter, or DDPF. Monte Carlo simulations were performed with the DPF and DDPF using the same methodology discussed in Section 3.4.1. However, to study the effect of the number of sources, each Monte Carlo simulation was repeated for different values of  $r_{\text{max}}$  ranging from 2 to 8. The true number of sources,  $r_{\text{true}}$ , is uniformly selected between 1 and  $r_{\text{max}}$ . Increasing  $r_{\text{max}}$  tends to challenge the algorithm as the particle filter must search a solution space that grows exponentially with the number of sources. Note that the initial particle count used by the DDPF,  $N_0$ , was set to 5,000 in each simulation.

The results of these Monte Carlo studies are shown in Figures 3.12-3.14 and Table 3.4. In each figure, the solid lines indicate the mean values. The bolded numbers in Table 3.4 show whether DPF or DDPF is superior in each specific case. In Table 3.4, it is evident that the DDPF only achieves a small improvement in mean position accuracy compared to the DPF, which is already quite accurate. Furthermore, the DDPF reduces the average runtime compared to the DPF for  $r_{\text{max}} < 7$  but only by a small amount, and it even shows higher mean runtimes for  $r_{\text{max}} = 7$  and  $r_{\text{max}} = 8$ . However, the main advantage of the DDPF is its ability to reduce the number of “failure” cases with very high position errors. Figure 3.12

Table 3.4: DPF and DDPF compared for varying  $r_{\max}$ .

Particle Filter $r_{\max}$	DPF ( $N = 5,000$ )						
	2	3	4	5	6	7	8
$\hat{r}$ Correct (%)	<b>98.67</b>	<b>92.67</b>	<b>85.33</b>	76	60.67	54.67	47.33
$\mu(\epsilon_{\text{pos}})$ (m)	<b>2.509</b>	<b>3.579</b>	4.492	5.514	6.45	6.593	8.373
$\sigma(\epsilon_{\text{pos}})$ (m)	<b>3.445</b>	3.87	3.465	4.069	4.943	5.049	5.774
$P_{95}(\epsilon_{\text{pos}})$ (m)	<b>5.44</b>	11.08	11.41	13.9	15.88	13.22	20.3
$\mu(\epsilon_{\varphi})$ (counts/s)	387.1	474.4	510.8	527.6	<b>616.4</b>	<b>491.4</b>	<b>1064</b>
$\sigma(\epsilon_{\varphi})$ (counts/s)	<b>1688</b>	<b>2139</b>	2462	<b>2628</b>	2770	3142	3875
Mean Runtime (s)	75.46	76.57	77.87	79.01	79.95	<b>82.82</b>	<b>83.05</b>
Median Runtime (s)	77.25	78	79.1	79.9	81.15	83.99	83.81

Particle Filter $r_{\max}$	DDPF ( $N_0 = 5,000$ )						
	2	3	4	5	6	7	8
$\hat{r}$ Correct (%)	<b>98.67</b>	<b>92.67</b>	84	<b>76.67</b>	<b>63.33</b>	<b>62.67</b>	<b>49.33</b>
$\mu(\epsilon_{\text{pos}})$ (m)	2.636	3.76	<b>4.454</b>	<b>5.33</b>	<b>5.92</b>	<b>6.091</b>	<b>7.471</b>
$\sigma(\epsilon_{\text{pos}})$ (m)	2.846	<b>3.064</b>	<b>3.035</b>	<b>3.484</b>	<b>3.576</b>	<b>3.519</b>	<b>4.038</b>
$P_{95}(\epsilon_{\text{pos}})$ (m)	6.115	<b>10.56</b>	<b>10.74</b>	<b>12.66</b>	<b>12.41</b>	<b>12.19</b>	<b>14.68</b>
$\mu(\epsilon_{\varphi})$ (counts/s)	<b>304.7</b>	<b>348</b>	<b>298.5</b>	<b>460.7</b>	833.2	814.8	1170
$\sigma(\epsilon_{\varphi})$ (counts/s)	2090	2229	<b>2264</b>	2632	<b>2648</b>	<b>3119</b>	<b>3539</b>
Mean Runtime (s)	<b>31.68</b>	<b>43.41</b>	<b>69.26</b>	<b>68.47</b>	<b>77.63</b>	110.3	134.1
Median Runtime (s)	<b>27.07</b>	<b>26.95</b>	<b>27.39</b>	<b>27.98</b>	<b>29.11</b>	<b>36.3</b>	<b>42.11</b>

shows that the DDPF almost eliminates these failure cases entirely at values of  $r_{\max} > 5$  by increasing the size of the particle set adaptively as measurements are gathered. This allows the filter to operate at maximum efficiency, using large particle sets when necessary to address ambiguity, and small particle sets as uncertainty is reduced. The improvement in the number of failed cases is seen in the 95% upper bounds on position error shown in Table 3.4 (denoted  $P_{95}(\cdot)$ ), which are better for the DDPF in all cases of  $r_{\max} > 2$ .

The reason for the DDPF's reduction of failure cases is encapsulated in Figure 3.13. This figure shows the mean of the set of negative log likelihoods  $\mathcal{Q}$  over all update steps. This is the metric used to increase or decrease  $N$  as described in Algorithm 2. A low average log likelihood (or high negative log likelihood) over the particle set means that the particle set poorly matches the measurements that have been obtained. Thus, a high mean negative log likelihood indicates that the PF may suffer high estimation errors without more

particles. Figure 3.13 shows that the DDPF is much less susceptible to very large mean negative log likelihoods, meaning that the quality of the particle set is generally better than the DPF in many cases. This improvement is achieved by increasing the number of particles when necessary, particularly at high  $r_{\max}$ .

The DDPF's ability to grow the particle set when necessary has obvious implications for runtime. Figure 3.14 shows runtime statistics for the DPF and DDPF, where the dashed line indicates the median values. The DPF runtimes are nearly flat with respect to  $r_{\max}$ , since runtime depends primarily on  $N$ , which is fixed. For the DDPF, most of the runtimes cluster around 20-30 sec, but several outlier cases are evident with much higher runtimes; the frequency of outliers grows with  $r_{\max}$ . In these cases, the DPCA algorithm increased  $N$  substantially at some point during the simulation. As a result, the mean runtime of the DDPF varies as a function of  $r_{\max}$  and even exceeds that of the DPF for  $r_{\max} > 6$ . However, as verified in Table 3.4, the median runtime of the DDPF is substantially lower than the DPF for all  $r_{\max}$ . In general, these results show the favorable performance of the DDPF compared to the DPF. In most cases, the DDPF exhibits similar estimation accuracy to the DPF with lower runtimes. In challenging cases, the DDPF achieves much better estimation accuracy than the DPF, albeit at the expense of increased runtime. This trade-off can be adjusted by tailoring the DPCA tuning parameters in Table 3.1. However, depending on the measurement scenario, longer runtimes can be an acceptable price to pay in order to reduce the errors in parameter estimates.

#### 3.4.4 Uncertain Absorption Parameters Study

In practical scenarios, it may be difficult or impossible to derive accurate absorption coefficients for all obstacles in an environment. It is therefore important to characterize performance of the particle filter in the presence of absorption parameter uncertainty. To test this, a Monte Carlo simulation was performed using the DDPF and the methodology described in Section 3.4.1 with  $r_{\max} = 3$ . In each simulation, the absorption coefficients for

each obstacle,  $\beta_{\text{obs}}$ , were randomized, while the PF’s attenuation kernel was built with an assumed constant value of  $\beta_{\text{obs}} = 10^{-2} \text{ m}^{-1}$  for all obstacles. This creates a mismatch between the true obstacle absorption coefficient and the assumed value. Random absorption coefficients for each obstacle were generated by first sampling  $\nu \sim \mathcal{N}(m, \sigma)$ , and then computing  $\beta_{\text{obs}} = 10^{-\nu}$ . Several Monte Carlo simulations were performed at different values of  $m$  and  $\sigma$ , as shown on the  $x$ -axis of Figure 3.15. The “no-obs” and “ $m = 2 \sigma = 0$ ” cases are control trials where there is perfect agreement between the kernels and the simulated environment. The “no-obs” case indicates there are no obstacles in the environment and all attenuation kernels use  $\beta_{\text{air}}$  only. The value of  $m$  dictates the bias in absorption coefficient estimates, while  $\sigma$  affects the deviation of the coefficients between buildings. The perturbed absorption parameter for a given obstacle stayed consistent across all 150 trials.

The results of this experiment are shown in Table 3.5. The position and source strength errors exhibit excellent performance for all except the  $m = 1, \sigma = 1$  case. In this case, the large bias led to numerous failed solutions. This is expected: the effects of  $\beta_{\text{obs}}$  decrease geometrically as its exponent  $-\nu$  increases linearly. Note that in Table 3.5, mean runtime is roughly correlated to the difficulty of each trial since the DDPF increases  $N$  substantially in the failed cases to try to obtain accurate estimates.

The trend in source strength estimation error as a function of uncertainty in  $\beta_{\text{obs}}$  is shown in Figure 3.15. This figure shows the mean source strength estimation error for each case with the failed cases removed for each  $(m, \sigma)$  combination. In general, the source strength estimate undershoots when the true obstacle absorption is larger than assumed and overshoots when the attenuation is higher than assumed. This is expected: when the true obstacles attenuate more than the particle filter assumes, the particle filter posits that the sources must simply be weaker and vice versa.

In the scenarios considered here, the DDPF appears to be fairly robust to errors in absorption parameters, exhibiting less than a 6% increase in position errors for cases other than the  $m = 1$  trials. This is because errors in the attenuation kernels are mitigated

Table 3.5: DDPF performance with obstacle attenuation parameter uncertainty.

Particle Filter	DDPF ( $N_0 = 5,000$ )								
	Obstacle $m$	none	2	1	1	2	2	3	3
Obstacle $\sigma$	N/A	0	1	0.5	1	0.5	1	0.5	
$\hat{r}$ Correct (%)	92.67	92.67	88.67	98	96	94	91.33	90.67	
$\mu(\epsilon_{\text{pos}})$ (m)	3.51	3.76	16.03	5.736	4	3.576	3.801	3.968	
$\sigma(\epsilon_{\text{pos}})$ (m)	2.726	3.064	26.81	8.758	2.52	2.637	2.779	2.969	
$\mu(\epsilon_{\varphi})$ (counts/s)	730.8	348	1673	2364	1513	288	567.9	497.1	
$\sigma(\epsilon_{\varphi})$ (counts/s)	1925	2229	9068	1584	1874	2079	2245	2738	
Mean Runtime (s)	49.7	43.41	2297	327	234.8	50.12	48.26	53.32	

by taking measurements in multiple places throughout the environment, including along obstacle-free paths in some cases. While the attenuation kernels will, in general, be different for isotopes with different energy spectra, this robustness may allow for multiple isotopes to share a single set of attenuation kernels. Further investigation of the effects of absorption parameter uncertainty is beyond the scope of this work but warrants additional research.

### 3.5 Lab Scale Experimental Setup

#### 3.5.1 Setup

Hardware experiments were performed indoors in a facility with motion tracking available as shown in Figure 3.16. Two different densities of concrete bricks were used, and their absorption coefficients were measured for Cs-137 and Co-60 as shown in Table 3.6. There were 17 obstacles, each consisting of two bricks. The search area is 12m x 4m. The detector used was a Kromek Sigma-50 CsI(Th) scintillating detector. The detector was positioned in the space on top of an iRobot Create 3 using positional feedback control from an external Vicon motion capture system. This allowed for the detector position to be known precisely in the environment. The positions of the obstacles were also known precisely. There were 45 measurements taken throughout the space, each with a 2 minute exposure time. The measurements are taken along a boustrophedon pattern and do not inform the search path.

Table 3.6: Obstacle characteristics.

<b>Medium</b>	<b>Air</b>	<b>Light Brick</b>	<b>Heavy Brick</b>
<b>Density</b> (kg * m <sup>-2</sup> )	1.29	1.65 × 10 <sup>3</sup>	2.25 × 10 <sup>3</sup>
<b>Cs-137</b> β m <sup>-1</sup>	1 × 10 <sup>-6</sup>	8.84	12.0
<b>Co-60</b> β m <sup>-1</sup>	1 × 10 <sup>-6</sup>	6.40	8.73

Table 3.7: Hardware measurement scenarios.

<b>Case</b>	<b>Scenario 1</b>	<b>Scenario 2</b>	<b>Scenario 3</b>
<b>Obstacles</b>	No	Yes	Yes
$r_{\text{true}}$	1	1	2
<b>Sources</b>	Cs-137 @ 24.69 mCi	Co-60 @ 4.76 mCi	Cs-137 @ 24.69 mCi Cs-137 @ 0.152 mCi

Three different scenarios were experimented with the configurations shown in Table 3.7.

### 3.5.2 Measurement Conversion

Spectrum measurements from the detector must first be converted to count rate measurements before they can be used by the particle filter. A set of example spectra taken as a set of measurements for the two source Cs-137 case can be seen in the left image of Fig. 3.17. This is done by integrating under the curve of interest. This is difficult for a few reasons. Firstly, as a detector saturates, the location of the photopeaks shift (sometimes called gain-shift). A peak fitting algorithm must be used, and it will need to be able to deal with noise, especially for weak sources or measurements with a small exposure time. Second, if multiple isotopes are present, it can be difficult to deconvolve the primary photopeak of one isotope with the Compton shelf of another. In this research, results are presented for Cs-137 and Co-60 isotopes. The counts were integrated for bins above a threshold found via a peak finding algorithm as shown in the center image of Fig. 3.17. Some measurements were thrown out because the detector was fully saturated, and there were no clear peaks as seen in the right image of Fig. 3.17.

### 3.5.3 Performance Metrics

The positions of the isotopes are known accurately via the motion capture system. The activities of the isotopes are taken from the Georgia Institute of Technology Office of Radiation Safety. The activities were then converted to equivalent source strengths using Equation (3.20) where  $A$  is the activity of the source in Curies,  $Bq$  is the number of Becquerels per Curie,  $A_d$  is the detector's cross sectional area,  $d_{eq}$  is the source equivalent strength distance (1m in our case), and  $E_d$  is the detector efficiency. The detector efficiency was calibrated using a Cs-137 source of known activity at a known distance.

$$\varphi = A * Bq * E_d * \frac{A_d}{d_{eq}^2} \quad (3.20)$$

The performance metrics presented are calculated exactly the same as they are in the simulation section except for source strength error. Source strength was first converted to activity, and then the percent error is given. The PFs were only ran once for each case, so only the error is reported and not statistics such as mean and standard deviation. The runtime for each case is the runtime of the PF algorithm only, and does not factor in the measurement exposure times.

## **3.6 Experimental Results**

Results of each of the three scenarios are presented in Table 3.8. Each source was localized within a circle whose area represents less than 1% of the total search area. The source strengths were also found within acceptable tolerances. The poorer performance of the Co-60 experiment was likely due to the measurement conversion process, and the error should decrease when the measurement conversion method is improved. The results presented for scenario 3 did require finding a lucky initial set of particles. This is due to the fact that the strength difference between the sources is very large. The smaller source had such a narrow region of convergence that particles were resampled out of the area since the local

Table 3.8: Hardware results.

Particle Filter Case	DDPF ( $r_{\max} = 3$ )		
	Scenario 1	Scenario 2	Scenario 3
$N_0$	500	5 000	500
$\epsilon_{\text{pos}}$ (m)	0.05746	0.1296	0.2294
$\epsilon_{\varphi}$ (%)	1.04	16.1	1.21
<b>PF Runtime (s)</b>	14.2	20.07	9.647

measurements were not significantly beyond background noise. “Region of convergence” refers to the area in the state space around a source in which a particle would persist and particles’ source hypotheses in that area would converge towards that source.

Figure 3.18 demonstrates why PF methods in general would outperform other methods such as contour following or peak finding. As seen in the figure, the contours and peaks do not indicate where the sources are located. The final particle set and MAP estimates for the case with 2 Cs-137 sources are shown in Figure 3.19. One thing to note in this figure is the top corner of measurements is unused. This is because the shielded sources being stored for later experiments were detectable in that area. When those measurements are used, particles cluster towards the top right of the area and desert the smaller source. This makes the region of convergence of that solution even smaller.

### 3.6.1 Limitations

Some important limitations to the proposed algorithm are worth highlighting. First, prior knowledge of obstacle locations and dimensions is required, and the algorithm currently offers no mechanism to update these parameters online. This may reduce accuracy if obstacle locations change or are unknown. Second, because measurements are assumed to lie on a discrete grid, the ability to model continuously moving detectors is limited. This could be addressed by increasing the density of the measurement grid and mapping continuous measurements to the nearest grid point. Third, the algorithm currently uses a single set of attenuation kernels, which may lead to errors in cases where multiple isotopes are present or the source spectrum is unknown. Finally, the algorithm does not distinguish between

isotopes, so further sensing and/or algorithmic extensions would be required for isotopic identification.

The last two limitations described above are manifest in the results. Data with more than one isotope present in the search area could not be processed. While the accuracy of the solutions was adequate, the next chapter will demonstrate the improvement gained by introducing a more refined algorithm for deconvolving a spectra into counts for individual isotopes. This addition will lead to improved accuracy of results, and it will add more functionality by incorporating isotopic identification. The results in this chapter represent the best case scenario when using a counting instrument instead of a spectral instrument.

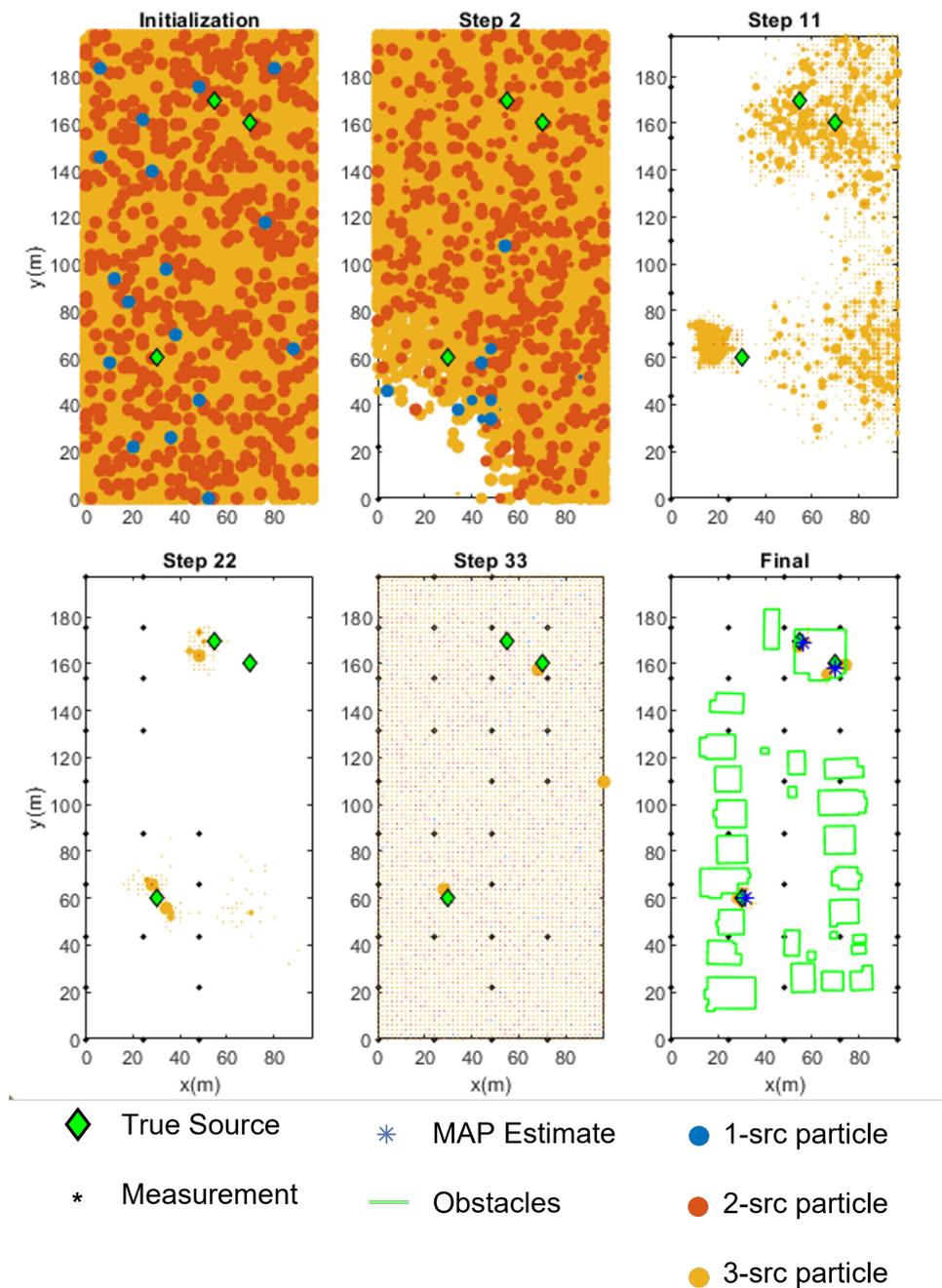


Figure 3.8: Stages of the PF. (top-left) initialization. (bottom-right) post-processing. (all others) update.

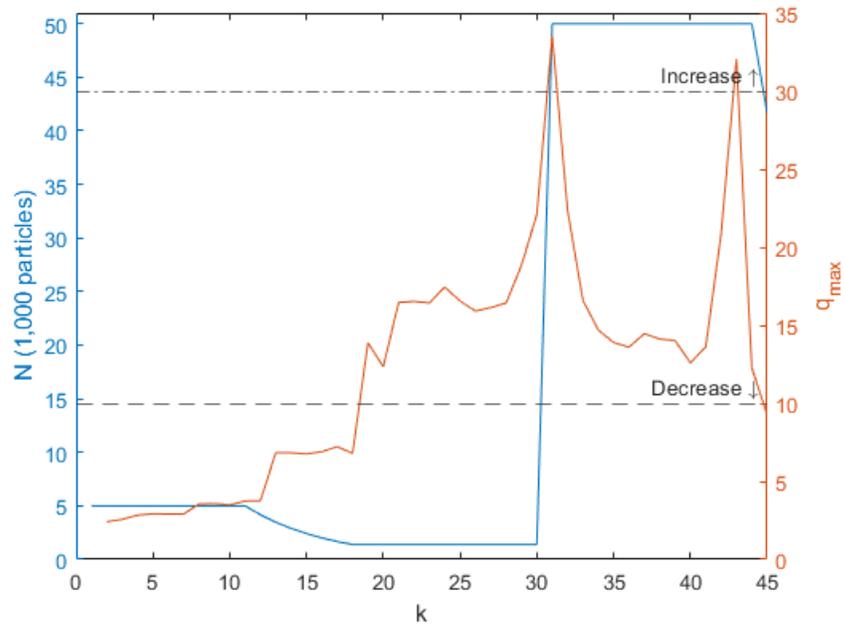


Figure 3.9: Convergence metrics and particle count for sample trial using the DDPF.

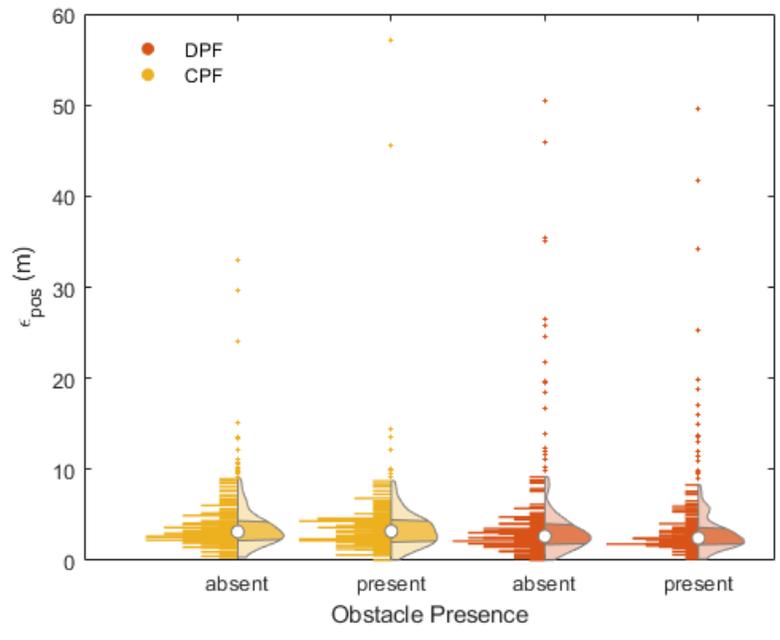


Figure 3.10: Position error for CPF and DPF in cases with and without obstacles.

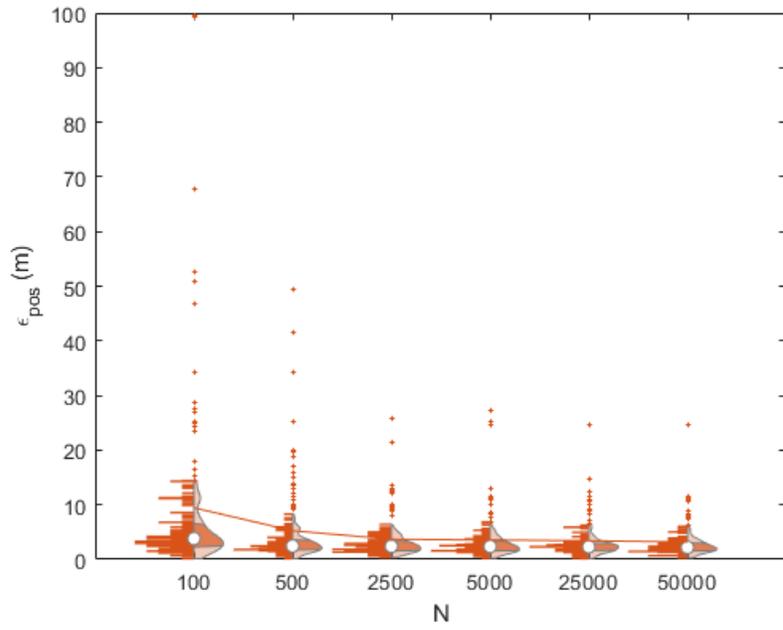


Figure 3.11: Position error for DPFs with different particle counts.

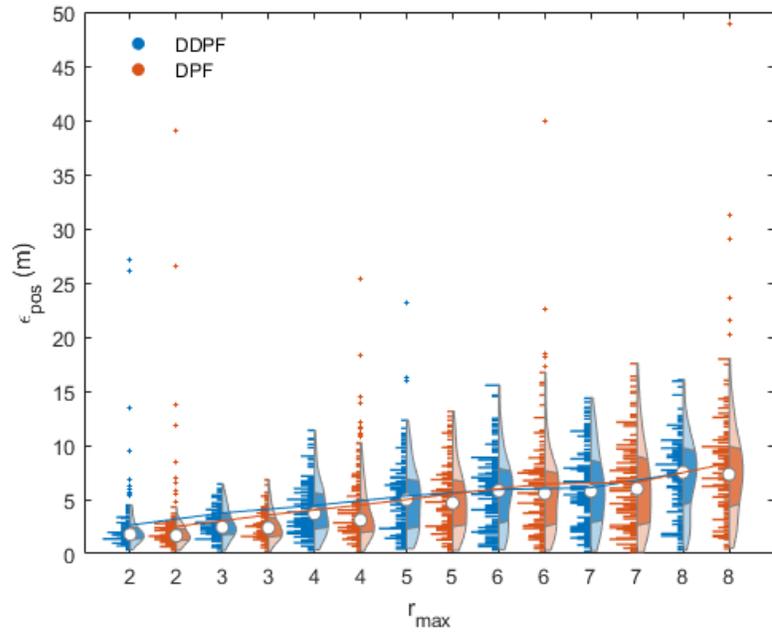


Figure 3.12: Monte Carlo position errors for DPF and DDPF.

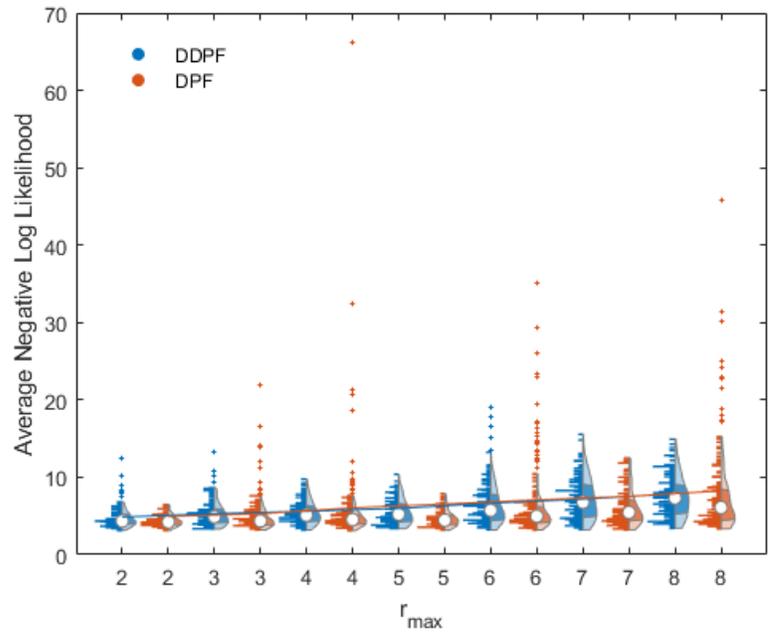


Figure 3.13: Mean negative log likelihoods over all update steps for DPF and DDPF.

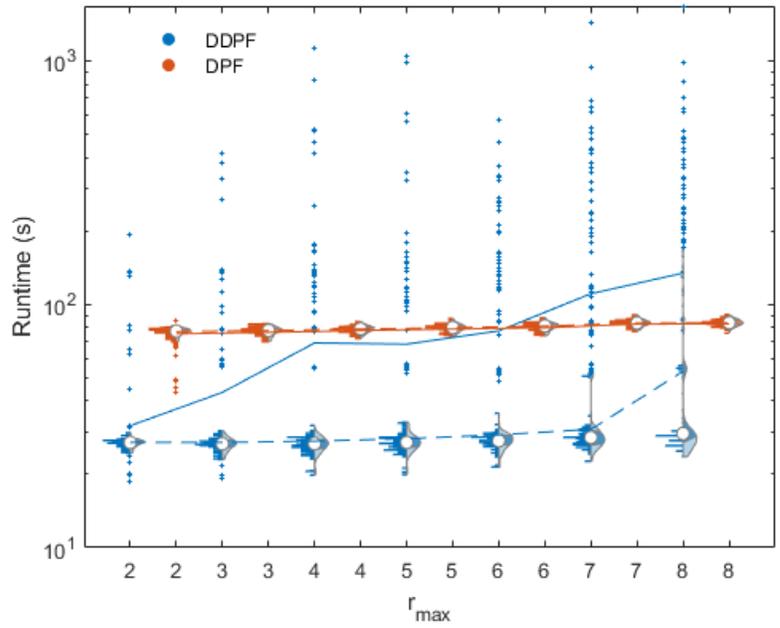


Figure 3.14: Monte Carlo runtimes for DPF and DDPF.

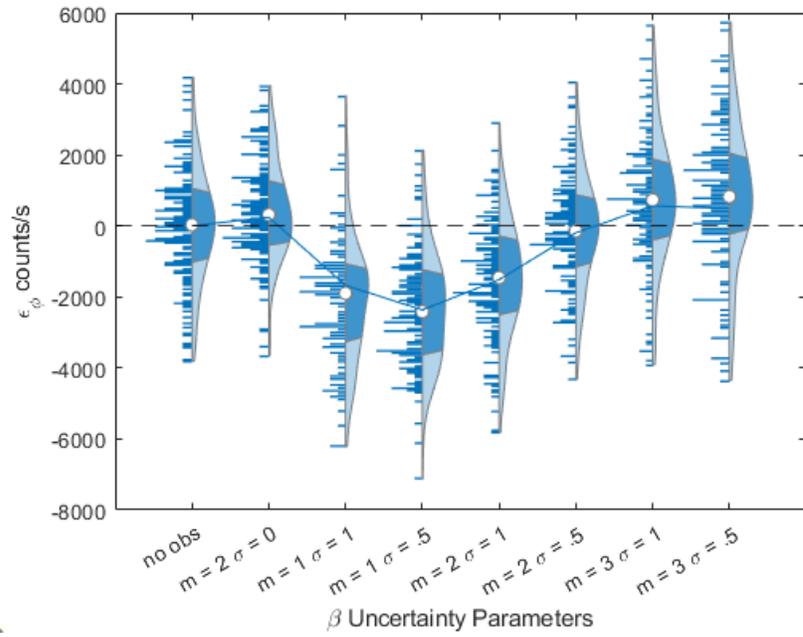


Figure 3.15: Source strength estimation error for varying levels of obstacle attenuation parameter uncertainty.

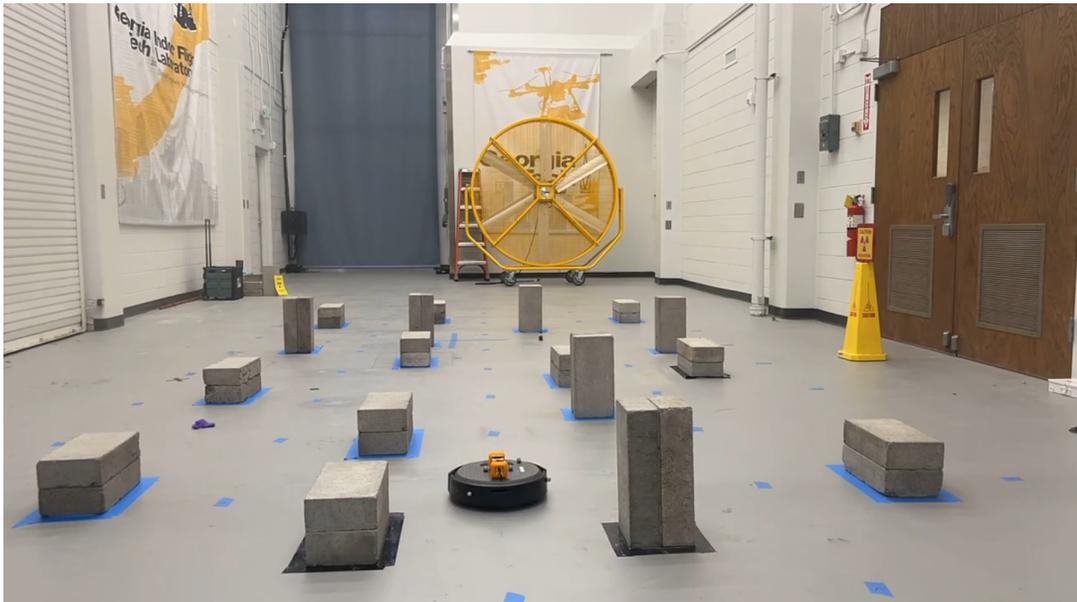


Figure 3.16: Photo of testing area with obstacles and ground robot visible.

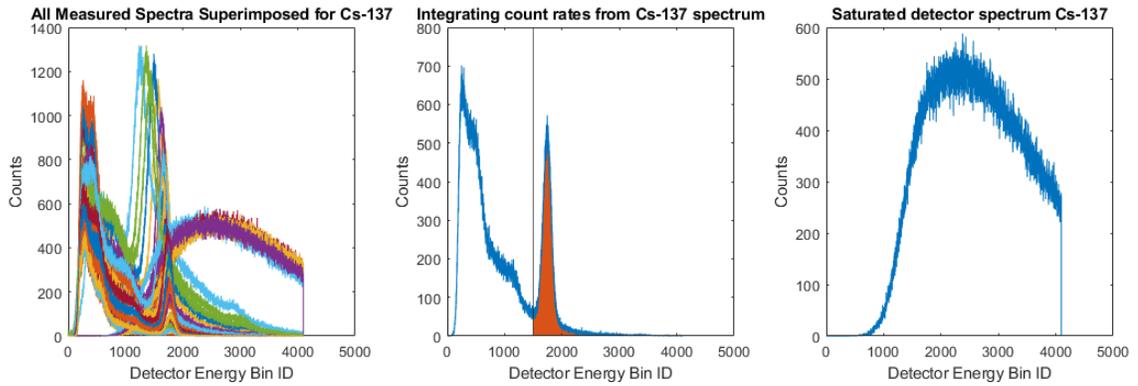


Figure 3.17: Recorded spectra from 2 source Cs-137 experiments. (left) All recorded spectra. (middle) Integrated area to become count measurements. (right) Example of saturated spectrum.

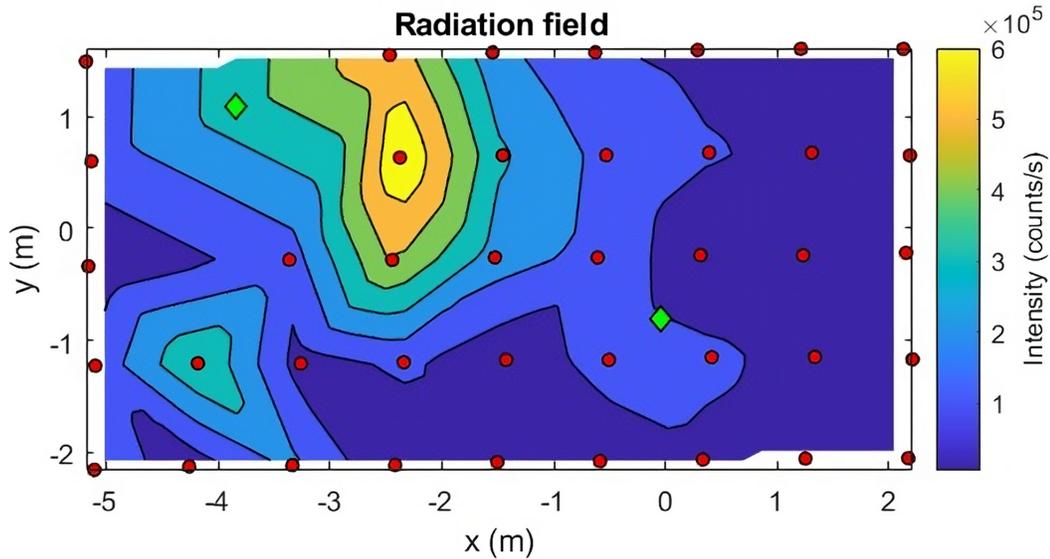


Figure 3.18: Interpolated radiation field given by measurements of 2 Cs-137 sources.

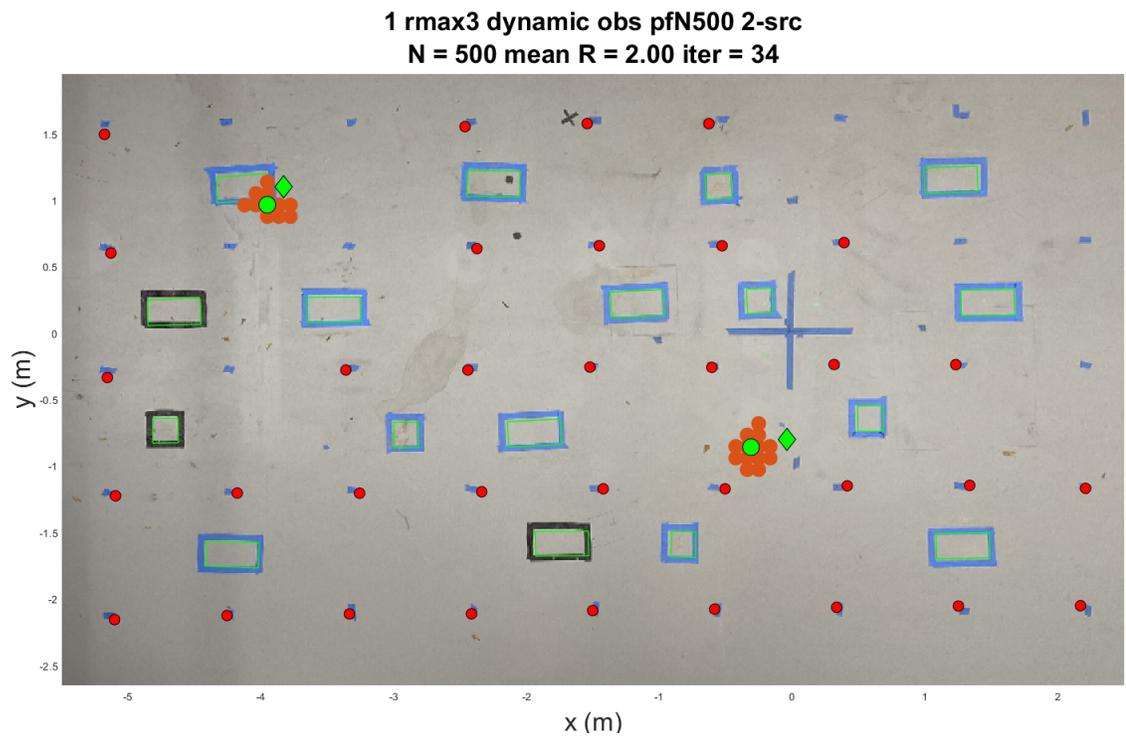


Figure 3.19: Final particle set for experiment with 2 Cs-137 sources.

## CHAPTER 4

### PARALLEL LOG-DDPF

#### 4.1 Problem Definition

The problem definition in this chapter is left in full for convenience even though there are many overlaps with the problem definition in Chapter 3. There are a few minor differences in notation, but the major differences to note are as follows:

1. The isotope is specified as a parameter of interest.
2. The computation of the equivalent source strength is refined to account for the properties of the detector used.
3. The Poisson distribution is not a saturated Poisson as a more refined model of detector saturation is used.
4. The measurements are taken from a spectral instrument and must be converted to counts.
5. The attenuation coefficient is specific to a given isotope.

The coordinate system is defined as  $(x, y, h)$  with respect to a local ground frame, where  $x$  and  $y$  represent right and up directions, respectively, and  $h$  is height above a reference altitude. There are an unknown number  $r_{\text{true}} > 0$  of sources of gamma radiation present. Point sources are indexed by  $s \in \{1, \dots, r_{\text{true}}\}$ . Each point source is parameterized by its isotope, spatial coordinates  $(x_s, y_s, h_s)$ , and equivalent strength  $\varphi_s$ . The equivalent strength calculation is discussed in depth in Section 4.3.2. A set of  $K$  measurements, indexed by  $k \in \{1, \dots, K\}$ , are taken in the space and parameterized by spatial coordinates  $(x_k, y_k, h_k)$  and value  $z_k \in \mathbb{N}_0$  which represents the total number of counts observed over time interval

$\tau$ . The problem statement is to infer the source parameters: isotope,  $(x_s, y_s, h_s)$ , and  $\varphi_s$  for all  $s$  given the measurements. For the purposes of this chapter, it is assumed sources lie on the  $h = 0$  plane.

The likelihood of measuring  $z$  counts during time interval  $\tau$  from a source emitting an expected count rate  $\mu$  at a given measurement location is Poisson distributed. Specifically, given the expected number of counts  $\lambda = \mu\tau$ , a measurement  $z$  is drawn from the Poisson distribution  $\mathcal{P}(\lambda)$ . As the radiation field propagates through the environment, the expected count rate  $\mu$  is subject to attenuation via absorption according to Equation (4.1). In this equation,  $\beta_{md}$  is the linear absorption coefficient of the medium,  $d_{md}$  is the distance travelled through the medium, and  $\mu_0$  is the unattenuated count rate.

$$\mu = \mu_0 e^{-\beta_{md} d_{md}} \quad (4.1)$$

## 4.2 Methodology

The proposed methodology is a series of algorithms collectively dubbed the Parallel Log Dynamic Discrete Particle Filter (Parallel Log DDPF). An overview of the Parallel Log DDPF is shown in Figure 4.1. By leveraging the additional information provided by gamma spectrometers instead of simple counting instruments, the Parallel Log DDPF is capable of accurately modeling attenuation in an environment, identifying distinct isotopes, and performing precise source term estimation in obstacle rich environments for multiple sources of the same or different species.

First, unmodified spectra are fed into the Gamma Spectrum Unfolding Algorithm from the detectors. The Unfolding Algorithm isolates the counts due to each isotope given in the table of isotopes. This involves applying the detector calibration, denoising, background subtraction, and overlapping peak separation. The methods presented in this section are designed to be as detector agnostic as possible; however, the overall measurement model of the detector greatly impacts the Unfolding Algorithm and will be discussed more in

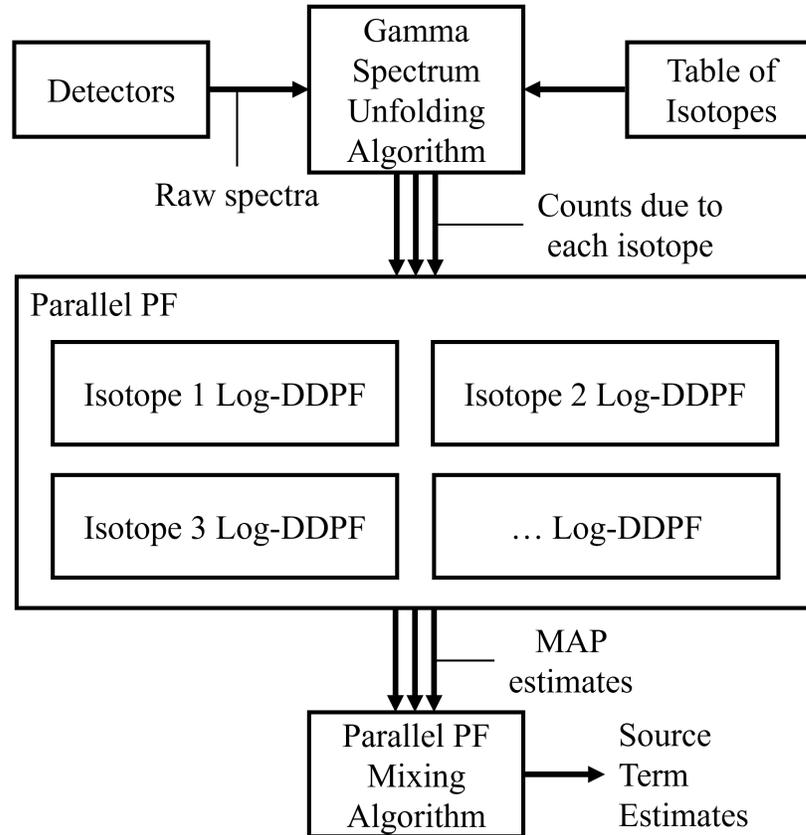


Figure 4.1: Overview of Parallel Log DDPF algorithm.

#### Section 4.3.2.

Once counts have been found for each isotope, they are passed separately to each corresponding Log DDPF which comprise the Parallel PF. The Log DDPF is nearly the same algorithm shown in Chapter 3 with the only difference being that the particle weights are manipulated in the logarithmic domain instead of the linear domain. Each individual Log DDPF is associated with a specific isotope from the table of isotopes. They each use absorption coefficients tailored to their specific energy to compute their attenuation kernels. Each Log DDPF runs completely independently and passes their final estimates to the Parallel PF Mixing Algorithm.

Finally, the Mixing Algorithm evaluates each set of isotope-specific measurements and final parameter estimates to determine which isotopes are actually present in the environment. The output of the Mixing Algorithm is a table containing the strength, position, and

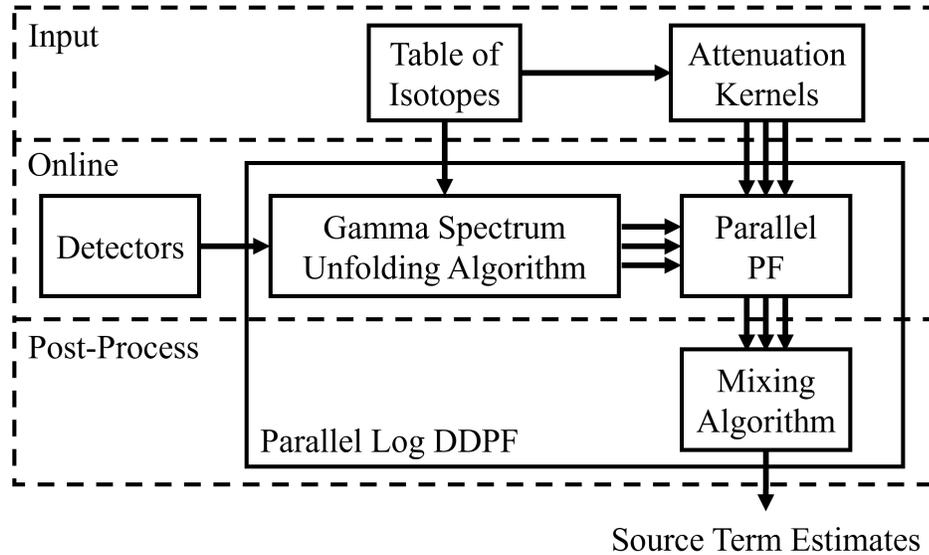


Figure 4.2: Overview of Parallel Log DDPF phases of operation.

isotope for each source of gamma emissions present in the environment.

As shown in Figure 4.2, all but two steps of this process operate during the online phase (while the measurements are being gathered). The attenuation kernels are precomputed, and thus they are considered an input alongside the table of isotopes. The Mixing Algorithm is a post-processing step; however, if desired, the Mixing Algorithm could be modified to continuously provide the Parallel Log DDPF’s best estimates during a search.

#### 4.2.1 Dynamic Discrete Particle Filter

The Dynamic Discrete Particle Filter (DDPF) is explained in detail in Chapter 3. For convenience, a summary of the essential elements is provided here. Relevant context to the goals of this chapter is provided as well. Figure 4.3 shows an overview of the DDPF.

##### *Regularized PF*

The core of the algorithm is a regularized PF found in [21]. The “particles” represent hypotheses about which sources are present in the environment. A particle can represent any number of sources, and a particle’s likelihood is evaluated as a whole (i.e. all sources

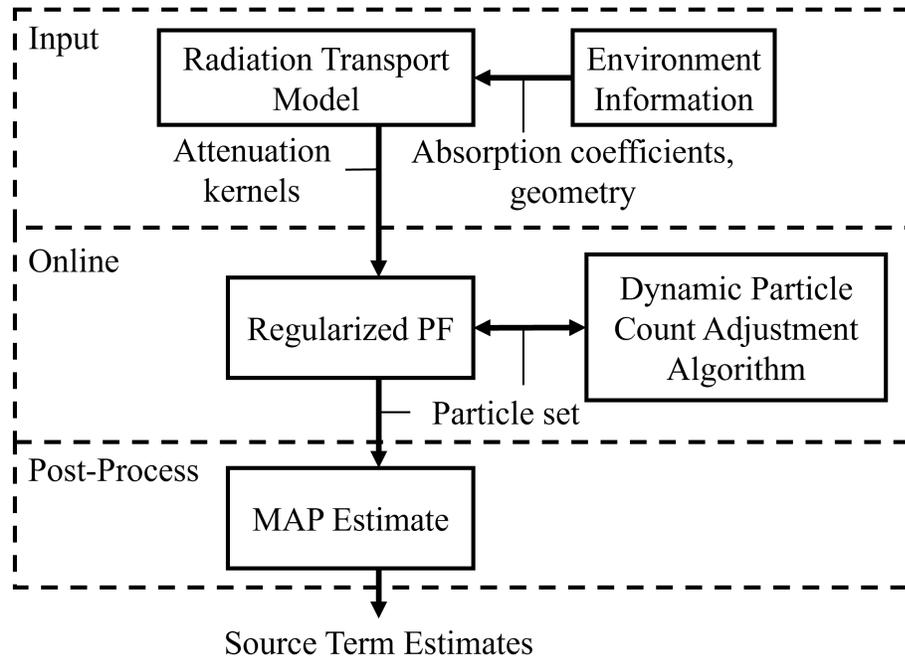


Figure 4.3: Overview of DDPF algorithm.

that comprise that particle contribute to the likelihood computation). Thus this algorithm can estimate the cardinality and source terms of the sources present in the environment.

Being a “regularized” PF means the particle’s states (number of sources, source positions, source strengths) are perturbed according to a Gaussian kernel. This facilitates a form of stochastic gradient ascent which allows for more coverage of a large state space with fewer particles.

#### *Discrete Attenuation Kernels*

The “Discrete” property allows the regularized PF to use “Attenuation Kernels” provided by the radiation transport model. These kernels describe the attenuation of gamma radiation from a set of fictional point sources to a set of proposed measurement locations. The attenuation is calculated via ray tracing through the environment using the absorption coefficients of the material and the geometry of the environment. The absorption coefficients are specific to the energy level of the radiation. Any method can be used to compute the attenuation kernels beforehand, even Monte Carlo methods.

This main benefit of this precomputation is speed and efficiency. The expected count rate must be computed for every particle with every measurement update. The kernels ( $\Theta_{k,s}$ ) link the source ( $s \in \{1, \dots, r\}$ ) and measurement ( $k$ ) locations. They allow the expected count rate to be computed during runtime via matrix product given Equation (4.2). Because the expected count rate scales linearly with equivalent source strength  $\varphi$ , the same kernels can be reused for colocated sources of different strengths. On the contrary, an entire radiation transport simulation would need to be performed in a “continuous” PF for each colocated source of differing strengths.

$$\hat{\mu} = \begin{bmatrix} \Theta_{k,1} & \cdots & \Theta_{k,r} \end{bmatrix} \begin{bmatrix} \varphi_1 \\ \vdots \\ \varphi_r \end{bmatrix} \quad (4.2)$$

#### *Dynamic Particle Count Adjustment*

The “Dynamic” aspect refers to the Dynamic Particle Count Adjustment (DPCA) algorithm. This computes the likelihood that the current set of measurements could’ve come from the current set of particles. It then raises, lowers, or maintains the particle count based on that likelihood. This allows the PF to balance speed and accuracy depending on how it is tuned.

#### 4.2.2 Log PF

Particle weights can approach numbers very close to zero. This is due to three factors: the state space is very large relative to the number of particles used, measurement updates frequently have narrow distributions that can be far into the tails of the prior distribution, and the likelihood computation uses a Poisson distribution for the measurement update. While this accurately models counting statistics, it does not accurately model a physical

detector's response. The detector's net response would be given by a complicated compound probability distribution due to the physics of radiation detection. An example of such confounding factors are the light collection efficiencies of a detector.

To overcome low particle weights, all computations involving the likelihood of particles (the likelihood computation, resampling, DPCA, and final parameter estimation algorithms) were moved to the logarithmic domain to preserve accuracy and stability. Instead of using the MAP estimate like the previous chapter, this chapter uses the Minimum Mean Squared Error (MMSE) estimate in the log domain. Although there is an equivalent MAP estimate in the log domain, the MMSE estimate was found to produce much more accurate results. The log resampling and MMSE estimates come directly from Gentner et. al [77] with no modifications. The weighing (likelihood computation) and DPCA steps are where novel contributions are made.

### *Log Likelihood Calculation*

The likelihood computation step computes the log Poisson probability for each particle according to Equation (4.3). Note the conversion from a factorial to a sum of  $z$  values. To increase computation speed, the terms are normalized by  $z$  such that it becomes Equation (4.4), except for the rare case when  $z = 0$ .

$$\log(p(z; \lambda)) = z \log(\lambda) - \sum_{i=1}^z \log(i) - \lambda \quad (4.3)$$

$$\log(p(z; \lambda)) \sim \log\left(\frac{\lambda}{z}\right) - \log(1) - \frac{\lambda}{z} \quad (4.4)$$

### *Iterative Jacobian Algorithm*

The resampling and MMSE estimates in the log domain hinge on the iterative Jacobian algorithm to compute the logarithm of a sum of two exponentials i.e  $\log(e^{\delta_1} + e^{\delta_2})$ . This is

also called Jacobi's Logarithm or Zech's Logarithm and is shown in Algorithm 3.

---

**Algorithm 3** Iterative Jacobian algorithm

---

```

1: function  $\log(\sum_{i=1}^n e^{\delta_i}) = \text{JACOBI}(\{\delta_i\}_{i=1}^n)$ 
2:   Init:  $\Delta_1 = \delta_1$ 
3:   for  $l = 2 : n$  do
4:      $\delta_l = \max(\delta_l, \Delta_{l-1}) + \log(1 + e^{-|\delta_l - \Delta_{l-1}|})$ 
5:   end for
6:    $\log(\sum_{i=1}^n e^{\delta_i}) = \Delta_n$ 
7: end function

```

---

*Log Resampling*

The resampling step uses the likelihoods  $\log(p(z; \lambda))$  as un-normalized weights,  $w^*$ . The weights are normalized by subtracting each weight by the normalization factor  $W$  given in Equation (4.5). To compute this sum while staying in the log domain, the iterative Jacobian algorithm is used.

$$W = \log \left( \sum_{i=1}^N e^{w_i^*} \right) \quad (4.5)$$

The normalized weights,  $w = w^* - W$ , are used to compute the log Cumulative Density Function (CDF). Then particles  $X_{1:n}$  are sampled from using the systematic resampling algorithm in the log domain. This process is shown in Algorithm 4. The new log weights of the particles are all equal to  $\log(n^{-1})$ .

*Log Dynamic Particle Count Adjustment*

The Log DPCA computes the likelihood that the measurements could have come from the current particle set in the log domain according to Equation (4.3). This contrasts the DPCA in Chapter 3 which calculates likelihood in the linear domain and converts it into the log domain. Because subsequent steps of the DPCA were already designed to work in the log domain, no further modifications were made.

---

**Algorithm 4** Systematic Resampling in the Logarithmic Domain

---

```
1: function  $X = \text{LOGRESAMPLE}(X, w)$ 
2:   Initialize the Log-CDF
3:    $c_1 = w_1$ 
4:   Construct the Log-CDF
5:   for  $l = 2 : n$  do
6:      $c_l = \max(w_l, c_{l-1}) + \log(1 + e^{-|w_l - c_{l-1}|})$ 
7:   end for
8:    $i = 1$ 
9:   Draw starting point
10:   $u_1 = \mathcal{U}[0, n^{-1}]$ 
11:  for  $j = 1 : n$  do
12:     $u_j = \log(u_1 + n^{-1}(j - 1))$ 
13:    while  $u_j > c_i$  do
14:       $i = i + 1$ 
15:    end while
16:     $X_j = X_i$ 
17:  end for
18: end function
```

---

### 4.2.3 Log Minimum Mean Squared Error Estimate

The MMSE of any given parameter of the particle filter  $\xi^{\text{MMSE}}$  is given in Equation (4.6).  $\xi$  is a vector containing all of a given parameter for a given source (e.g.  $\varphi_s$ ) from each particle. The notation  $\mathcal{A}+$  and  $\mathcal{A}-$  indicates the parameter vector entries that are positive and negative respectively. This is because the parameters absolute values must be fed into the iterative Jacobian algorithm as logarithms are only defined for positive numbers. Note that the case where a parameter equals zero is missing. This is because in a weighted average, zero multiplied by a weight is zero anyway. This algorithm is equivalent to performing a weighted sum in the linear domain, but it preserves the precision of the logarithmic domain.

$$\xi^{\text{MMSE}} = e^{\text{Jacobi}(\{w + \log(|\xi|\})_{\mathcal{A}+})} - e^{\text{Jacobi}(\{w + \log(|\xi|\})_{\mathcal{A}-})} \quad (4.6)$$

### 4.2.4 Gamma Spectrum Unfolding Algorithm

Gamma spectrum unfolding is used to separate out the counts due to each isotope from the spectra to feed into the Parallel PF. There are many methods for gamma spectrum unfolding [78]; in general, each method contains these steps: denoising, peak finding, baseline

subtraction, and overlapping peak separation. Gamma spectrum unfolding is a difficult process with the complexity depending on the detector used and the isotopes present. This makes each Unfolding Algorithm quite bespoke.

### *Denoising*

There are many methods for denoising a spectrum which are generally split into two categories: Time domain filtering and frequency domain filtering. Within time domain filtering are methods like Finite Impulse Response filtering (FIR) [79], least square basis function fitting [80], and Support Vector Machine (SVM) methods [81]. Within frequency domain filtering, methods such as wavelet transforms [82], Fourier transforms [83], and Noise Adjusted Singular Value Decompositions (NASVD) [84] are used.

In this work, the adaptive smoothness Penalized Least Squares (asPLS) method [85] is used. It is a least squares basis function fitting technique that was originally developed for infrared spectra. This algorithm was implemented using Pybaselines' [86] whittaker.aspls method.

### *Peak Localization*

There are myriad ways to find peaks in data. In addition to standard numerical methods, a few other notable methods are the wavelet transform [87], Gaussian curve fitting [88], or Savitzky–Golay filtering [89].

This work uses Gaussian curve fitting via Lawrence Berkeley National Laboratory's python package, becquerel [90]. Specifically, peaks were found by convolving the spectrum with a Gaussian kernel using the PeakFinder method with a GaussianPeakFilter kernel. This kernel uses the detector resolution to search for peaks that roughly fit the resolution. The peak locations were then compared to the expected locations of the gamma lines. If they were located within 10% of the expected energy value and the signal-to-noise ratio (SNR) was above 5, then the peak was assumed to be due to the given set of isotopes.

Otherwise, they are discarded.

The expected locations of the gamma lines  $E$  is provided in the table of isotopes. The gamma lines are associated with a mode of decay. Only one gamma line need be chosen for each isotope, and the choice will depend on detector resolution and which other isotopes are on the table. For example, with a low resolution detector, it is better to use the 1332 KeV over the 1173 KeV gamma line for Co-60 when Ra-226 is present because of the 1120 KeV line emitted by one of Ra-226's daughter isotopes, Bi-214. The more that is known about an isotope, the more accurate the unfolding process will be. However, the Parallel Log DDPF could be run to perform STE for the gamma lines with no *a priori* knowledge of isotopes in general. It could simply return the location and relative strengths of sources for each peak that is detected, and those peaks could be matched to certain sources *a posteriori*. (For example, the algorithm could estimate there are two colocated sources with similar strengths corresponding to 1173 KeV and 1332 KeV gamma lines. An operator could then infer that there is a single Co-60 source at that location.)

### *Gain-Shift Correction*

One challenge to the described peak localization approach is something called “gain-shift.” This was briefly mentioned in Chapter 3 Section 3.5.2, and it is shown in Figure 3.17. Gain-shift is where the locations of the photopeaks shift to lower energy levels. Thus the expected location of a gamma line will no longer coincide with the actual location of a photopeak. To fix this, a gain-shift correction was put into place. The “gain” ( $\Gamma$ ) corresponds to the linear calibration from energy bins to energy (KeV) as seen in the abscissa of spectra. This gain is usually a scalar multiplied by the energy bin to yield the approximate energy level. It is calibrated using a source with a very distinct photopeak (such as Cs-137). However, the value of this calibration constant changes with the count rate. Too high of a count rate, and this gain shifts.

Multiple trials were performed logging the photopeak location and gross counts ( $C$ ) to

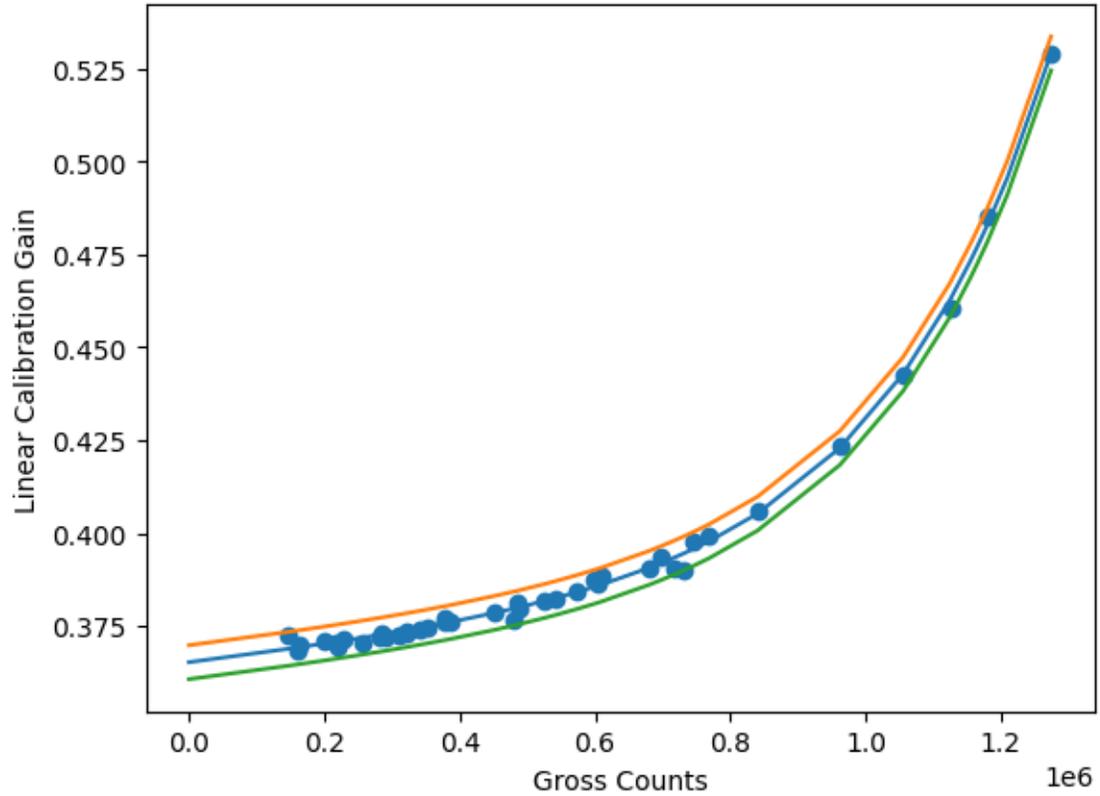


Figure 4.4: Linear calibration gain and gross counts plotted using Cs-137 photopeaks for calibration.

generate the data points seen in Figure 4.4. (Note that gross counts were used instead of counts per second. Since all measurements were taken for 120s, the gross counts and count rate are directly proportional.) An exponential function given in Equation (4.7) is fit to the data. The solid blue line in Figure 4.4 is Equation (4.7). The orange and green lines are the upper and lower tolerance bounds, respectively. These lines represent the furthest a linear calibration gain can be from the curve before it is rejected as an invalid measurement. The coefficients ( $a, b, c, d$ ) are given in Table 4.1.

$$\Gamma = ae^{bC} + ce^{dC} \quad (4.7)$$

Table 4.1: Gain-shift correction coefficients

<b>Coefficient</b>	<b>Value</b>
<i>a</i>	0.3645
<i>b</i>	5.564e-08
<i>c</i>	0.000743
<i>d</i>	4.099e-06

### *Baseline Correction*

Baseline correction, also referred to as background subtraction, involves the removal of elements of the spectrum that are not directly associated with the peak. This can be done via 2<sup>nd</sup> difference background subtraction [91], peak clipping [92], discrete derivative filtering [93], iterative filtering [94], Sensitive Non-linear Iterative Peak clipping (SNIP) [95], or using the Fourier transform [96].

The simplest method for calculating the baseline of a defined peak is a linear baseline [97]. For each peak found, the linear baseline is drawn between the counts at the  $\pm 3\sigma$  energy bin locations as they generally occur where the peak blends back into the spectra. This range of  $\sigma$  was chosen to be large enough to fully cover the base of the peak, but not too large as to encroach on neighboring peaks.

### *Overlapping Peak Separation*

Overlapping peak separation, sometimes referred to as “deconvolution,” is the most difficult part of gamma spectrum unfolding. There are many complex methods such as Fourier auto reverse convolution [98], gold’s algorithm [99], or iterative convolution [100]. In addition to finding the peaks, Gaussian curve fitting can be used to deconvolve them as well [88]. There are also various geometric methods such as the tangent method [101].

However, because we have a limited set of isotopes to work with, we can perform spectrum stripping [97]. This is the process of subtracting out the relative contributions of each peak until nothing is left. For example, our table of isotopes has Cs-137 with a gamma line at 662 KeV and Ra-226. Ra-226 has a strong gamma line at 609 KeV due

Table 4.2: Table of isotopes after Unfolding Algorithm is run.

<b>Isotope</b>	<b><math>E</math> (KeV)</b>	<b>Counts</b>
Ra-226	351.932	2177
Cs-137	661.657	134
Co-60	1332.492	1547

to the Bi-214 daughter, and with a low resolution detector this peak overlaps with the 662 KeV peak. However, Ra-226's 352 KeV peak, due to its Pb-214 daughter, doesn't interfere with any other isotopes of interest. So, using the intensity of the 609 KeV peak relative to the 352 KeV peak, the approximate amount of counts due to the 609 KeV peak can be subtracted from the 662 KeV peak.

A sample isotope table with counts given by the Unfolding Algorithm is given in Table 4.2. The corresponding output spectra is shown in Figure 4.5. One thing to note is that the 662 KeV Cs-137 peak seems the largest in Figure 4.5, but has a very low count value in Table 4.2. This is because the Cs-137 in this measurement is located a few meters away, and the peak is in fact the 609 KeV peak due to the daughter, Bi-214.

#### *Non-Paralyzable Detector Model*

The counts from the Unfolding Algorithm do not yet account for the detector's dead time, so one additional step is needed. This work uses a non-paralyzable detector model as shown in Equation (4.8) to correct for the dead time of the detector [97]. This calculation yields the true interaction rate  $\omega$ , given the dead time factor  $\tau_d$  (for the Kromek Sigma 50, this is  $5.813 \times 10^{-9}$  s [102]), and recorded count rate  $\mu_{rec}$ . The true interaction rate is multiplied by our sampling time  $\tau$  to yield a measurement  $z$  that is actually used by the Parallel Log DDPF. The recorded count rate is the counts given by the Unfolding Algorithm up to this point divided by the sampling time.

$$\omega = \frac{\mu_{rec}}{1 - \mu_{rec}\tau_d} \quad (4.8)$$

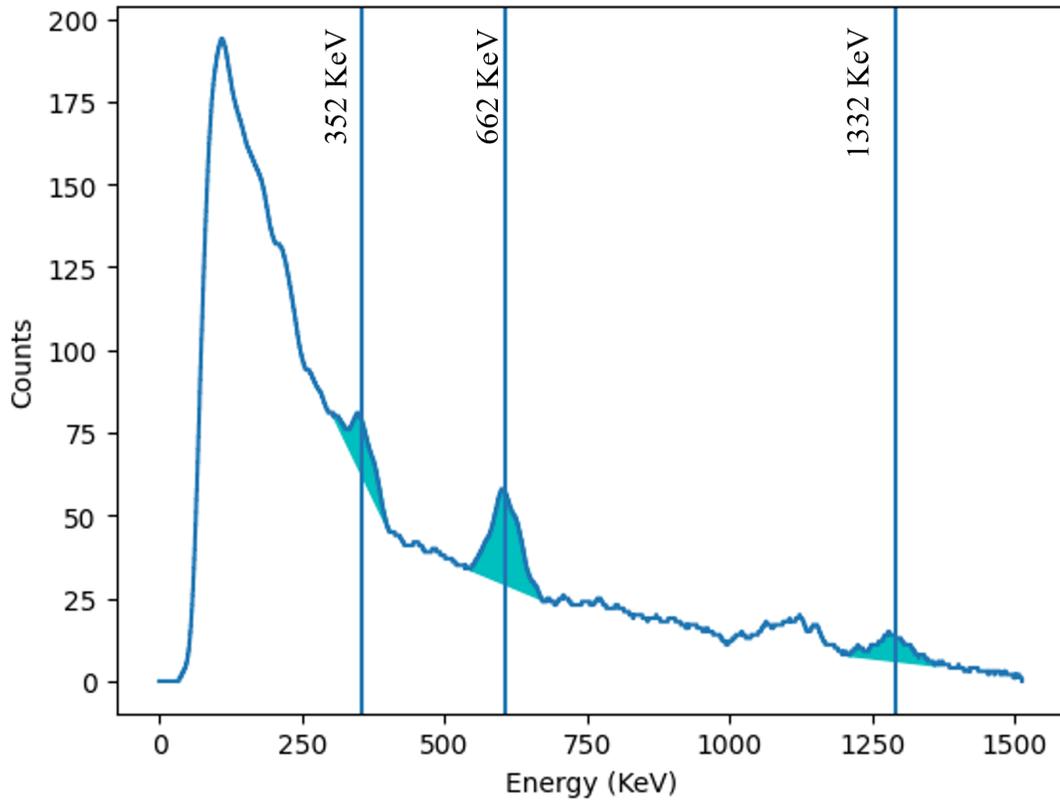


Figure 4.5: Example calibrated spectrum taken over 120 s with Kromek Sigma 50. Peak centroids shown by vertical lines. Peak areas are filled in with baselines removed.

#### 4.2.5 Parallel PF

The object of the Parallel PF is to facilitate operation with various isotopes simultaneously. Whereas the original DDPF could only handle multiple sources of the same isotope, extending it into the Parallel PF allows for multiple sources of the same or different isotopes. This is facilitated by the Unfolding Algorithm because a single spectra can now independently generate count measurements for multiple isotopes. In most cases, these measurements will be entirely independent of one another and allow for several Log DDPF algorithms to work in parallel. Using multiple independent Log DDPFs also allows for the use of attenuation kernels tailored to the specific isotope.

### *Parallel PF Mixing Algorithm*

The particles used in this work cannot represent the “zero sources present” hypothesis, unlike [21]. However, when there are no sources of a certain isotope present, the Log DDPF’s estimated source terms will be a single source of the minimum strength at an arbitrary location. Thus, we need a post processing step to determine which sources are spurious; this is the job of the Mixing Algorithm.

The Mixing Algorithm collects the set of output sources from the Parallel PF once all the constituent Log DDPFs yield their MMSE estimates. The Mixing Algorithm finds the output sources near the minimum strength threshold. At each of these low-strength output sources’ locations, a fictional source of the minimum strength is created. The expected count rate due to this fictional source is computed for all the measurement locations. The measurement location with the largest expected count rate is examined. This fictional value represents the minimum threshold that the recorded measurement at that location should be. If the recorded measurement is less than 90% of the fictional measurement, the corresponding output source is removed from the set of output sources.

Once this process is complete, the remaining output sources represent the final source term estimates for the Parallel Log DDPF.

## **4.3 Lab Scale Experiments**

### 4.3.1 Setup

All experiments were performed in a scaled down indoor test facility shown in Figure 4.6. This is for two reasons. First, there were limitations on transportation, storage, and use of the sources available. Second, the facility is equipped with a Vicon motion-capture system to eliminate the hassle of performing robotic localization. In Figure 4.6, the coordinates are centered on the origin labelled “O” with the x, y, and h axes shown. There are three points labelled a, b, and c. Their coordinates are shown in Table 4.3, and they are the locations

where sources can be during the experiments. The floor is not perfectly level, otherwise their h-coordinates would be zero. These positions are not known to the search algorithm *a priori*.

Each experiment was performed with all 17 obstacles present. Each obstacle is comprised of two concrete bricks. In Figure 4.6, certain bricks are outlined with black tape, these bricks are 50% more dense than the bricks outlined with blue tape (corresponding to a roughly 50% higher absorption coefficient  $\beta_{\text{md}}$ ).

The orange line in Figure 4.6 denotes the search path of the robot. The simple path was chosen instead of using an active search method in order to keep the quality of measurements consistent across experiments. The same 45 measurements locations were used in the same order for all experiments. The measurements lie on a  $9 \times 5$  grid with a slightly less than 1 meter basis. The search area comprises in total a  $9 \text{ m} \times 5 \text{ m}$  area.

The robot performing the search is shown in Figure 4.7. It is an iRobot Create 3 with a Kromek Sigma 50 detector on top controlled by a Raspberry Pi 4b running ROS. The robot communicates with a base station over WiFi. The robot receives its position from the Vicon motion-capture system with sub-millimeter precision at 100 Hz. The robot also relays the detector's raw count data. The robot is programmed to move along a given path of measurement locations. At each location, it sits stationary and records count data for 120 s.

The experiments that were performed are tabulated in Table 4.4. The experiments are labelled with capital A, B, and C. There were 4 unique sources used in the experiments. They were 2 Cs-137 sources of different activities (labelled Sm and Bg corresponding to the weaker and stronger of the sources, respectively), a Co-60 source, and a Ra-226 source. The columns of the table indicate if and where a source was present in the given experiment. For experiment C, the Co-60 was shielded with 38 mm of lead, and the Ra-226 was shielded with 3.2 mm of lead. The source activities and shielding values can be found in Table 4.5.

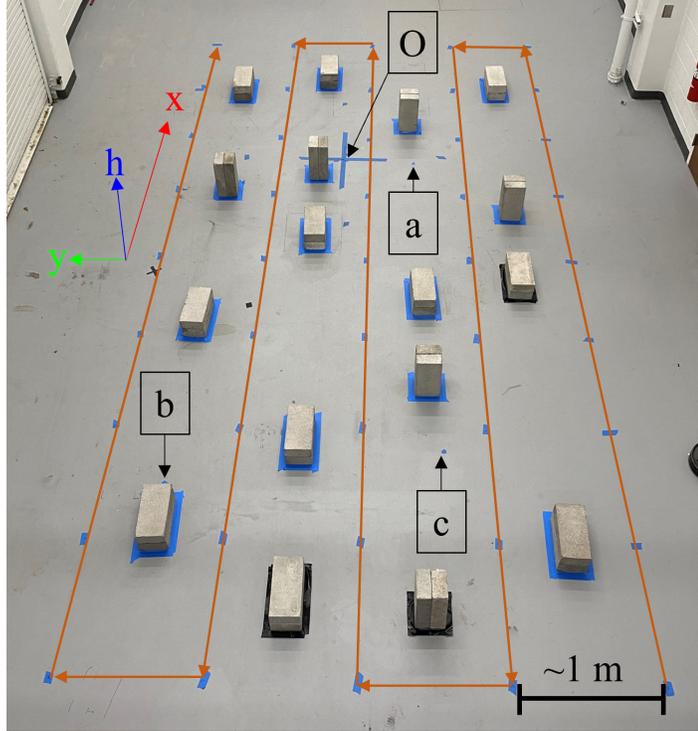


Figure 4.6: Search area for experiments.

Table 4.3: Source location coordinates

Location	x (m)	y (m)	h (m)
a	-0.0371	-0.7024	0.0017
b	-3.8321	1.0961	0.0057
c	-3.5048	-0.8778	0.0067

### 4.3.2 Detector Model

In order to gauge the accuracy of the equivalent source strength estimate  $\hat{\varphi}$  given by the Parallel Log DDPF, it is necessary to convert the activity of the sources into their reference equivalent source strengths  $\varphi_{\text{true}}$ . The reference equivalent source strength is the number of counts a detector will report in a photopeak at 1 m from a source in 1 s with the minimal set of attenuating factors between the source and detector. Equation (4.9) shows how to convert from activity to equivalent source strength [97], [103].

Table 4.4: Experiment matrix

Src \ Exp.	A	B	C
<b>Sm Cs-137</b>	a	a	a
<b>Bg Cs-137</b>	-	b	-
<b>Co-60</b>	b	-	b*
<b>Ra-226</b>	-	-	c*

\* The source at that location was isotropically shielded to approximate a lower activity source.



Figure 4.7: Robot to take measurements.

$$\varphi = A \times \epsilon_{\text{ex}} \times \epsilon_{\text{in}} \quad (4.9)$$

In Equation (4.9), the activity  $A$  in Becquerels is multiplied by extrinsic  $\epsilon_{\text{ex}}$  and intrinsic  $\epsilon_{\text{in}}$  efficiency factors. The extrinsic factors are shown in Equation (4.10). They encompass things that are not specific to the scintillating properties of the detector crystal.  $G$  is the fraction of all space that the detector subtends, sometimes referred to as the geometrical solid angle or extrinsic efficiency. This value is given in Equation (4.11) as simply the ratio of the area of the detector face  $A_d$  to the area of a sphere with radius  $r_{\text{sd}}$  equal to the source-to-detector distance.  $I$  is the fraction of photons transmitted by the intervening materials between source and detector. This value is found using the absorption coefficients of the intervening materials. For the reference case, only air and the thin aluminum casing of the

detector would contribute. Thus,  $I$  is assumed to be 1 unless it is shielded ( $I_{\text{shield}}$ ).  $TB$  is the total branching ratio for a specific gamma line and is sourced from [104].  $TB$  is the product of  $BF$  and  $BR$  where  $BF$  is the branching fraction for the mode of decay (gamma line) of interest, and  $BR$  is the branching ratio for that photon energy.

$$\epsilon_{\text{ex}} = G \times I \times TB \quad (4.10)$$

$$G = \frac{A_d}{4\pi r_{\text{sd}}^2} \quad (4.11)$$

The detector intrinsics are the efficiency values directly associated with the scintillating crystal's composition.  $M$  is the absorption efficiency, i.e. the fraction of photons absorbed by the detector.  $\eta_{\text{pt}}$  is the detector's peak-to-total ratio. This value relates the net counts in the photopeak to the total counts in the detector. Both  $M$  and  $\eta_{\text{pt}}$  vary with the energy level of the incident photon and in this work are interpolated via data given by [103]. A few experiments were performed using Cs-137 and Co-60 to validate this data. The source activities and extrinsic efficiencies were known *a priori* from manufacturer datasheets. The reference equivalent source strength is calculated for a 1 meter source-to-detector distance and 1 second measurement time using count measurements output from the Unfolding Algorithm. Finally, Equation (4.9) was used to compute the intrinsic efficiency. It was calculated to be within 1% of the intrinsic efficiency found when using the interpolated data. Table 4.5 shows the exact values used for the conversion of the source activities to their reference equivalent strengths.

$$\epsilon_{\text{in}} = M \times \eta_{\text{pt}} \quad (4.12)$$

Table 4.5: Source activities, strengths, and conversion factors

<b>Source</b>	<b>Sm Cs-137</b>	<b>Bg Cs-137</b>	<b>Co-60</b>	<b>Ra-226</b>
<b>Param</b>				
<b>Activity (mCi)</b>	0.152	24.7	4.80	0.950
<b><math>E</math> (KeV)</b>	661	661	1332	351
<b><math>\varphi_{\text{true}}</math> (cps)</b>	72.43	11790	856.0	-
<b><math>\varphi_{\text{true, shield}}</math> (cps)</b>	-	-	85.60	88.18
<b><math>I_{\text{shield}}</math></b>	-	-	0.10	0.23
<b><math>TB</math> (%)</b>	85.1	85.1	100	35.7
<b><math>M</math></b>	0.73	0.73	0.43	0.82
<b><math>\eta_{\text{pt}}</math></b>	0.52	0.52	0.22	0.75
<b><math>A_d</math> (m<sup>2</sup>)</b>	$6.452 \times 10^{-4}$			
<b><math>r_{\text{sd}}</math> (m)</b>	1			

### 4.3.3 Results

The results are shown in Table 4.6. The position error  $\epsilon_{\text{pos}}$  is simply the Euclidean distance between the source's true position and the Parallel Log DDPF's estimate. A comparison between the Parallel Log DDPF and a Parallel DDPF (working in the linear domain) was attempted, but the linear domain algorithm failed after a few measurements due to continuously degenerating to zero weight particles (this was not a problem in Chapter 3 where simulated count measurements were used). To contextualize the position error, a circle with a diameter of 10 cm comprises an area 0.017% of the 9 m  $\times$  5 m search area. Moreover, due to the discretized attenuation kernels, the source hypotheses lie on a grid with spacing of just over 8.5 cm. The source equivalent strength error is given by Equation (4.13).

$$\epsilon_{\varphi} = \left| \frac{\varphi_{\text{true}} - \hat{\varphi}}{\hat{\varphi}} \right| \times 100\% \quad (4.13)$$

Experiments A and C show favorable performance. Experiment B was the most challenging experiment because it was two sources of the same isotope with drastically different activities. The smaller activity source lies just above the allowable minimum source strength hypothesis of 50 cps. There are no measurements where the larger source does not dominate. Curiously, the larger source strength was underestimated by 25% while the

Table 4.6: Experimental results

<b>Experiment</b>	<b>Isotope</b>	$\epsilon_{\text{pos}}$ (cm)	$\epsilon_{\varphi}$ (%)
<b>A</b>	Sm Cs-137	6.512	2.573
	Co-60	4.673	4.279
<b>B</b>	Sm Cs-137	12.27	24.49
	Bg Cs-137	5.153	25.10
<b>C</b>	Sm Cs-137	3.688	9.326
	Co-60*	4.614	3.855
	Ra-226*	7.535	1.887

\* The source was isotropically shielded to approximate a lower activity source.

smaller source was overestimated by 25%. This indicates that there are a spectrum of valid hypotheses, and that the system is underdetermined by the measurement set. More measurements at critical locations could improve this performance.

The convergence of the particle filter in Experiment C is shown in Figure 4.8. In this case, the word “convergence” is used to indicate the movement of particle filter’s estimated parameters towards the true values shown as dashed horizontal lines. These plots also show the cardinality via the number of colored lines plotted. As the particle filters progress, their cardinality also arrives at the true value. All values in this plot are found using the MMSE estimates at each measurement iteration,  $k$ . Most parameters seem to quickly find their true value and stay nearby except for the Ra-226 y-coordinate. There is some instability in the value over time until the last several measurements. The cause of this behavior is likely due to the second source estimate not being filtered out until later in the experiment, and the MMSE estimate vacillates between selecting one, the other, or both sources.

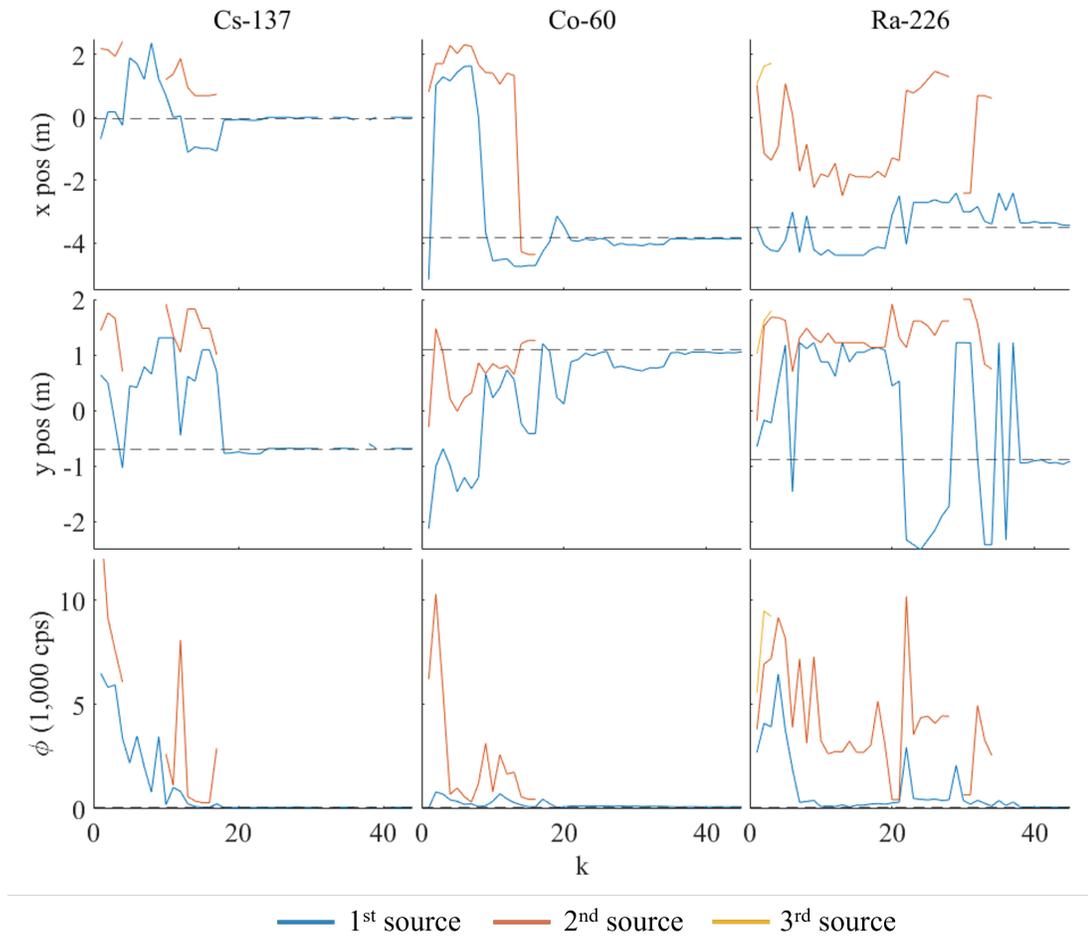


Figure 4.8: Convergence of Parallel Log DDPF for Experiment C.

## CHAPTER 5

### CONCLUSION

#### 5.1 Contributions

- Chapter 2 demonstrates a potential methodology for rapid radiological mapping using teams of UAVs and UGVs. The concept leverages the combined rapid transportation capability of UAVs and the accurate sensing performance of UGVs to quickly estimate accurate radiation maps. A one-step-ahead information-driven search method was formulated for UAV path planning, and a simple Boustrophedon scanning pattern was employed for the UGVs. Simulation results compared performance of the UAV-UGV teams against a UGV-only team and against UAV-UGV teams using random sampling, highlighting the performance benefits obtained from the UAV's rapid transportation capability and the benefits of information-driven planning. Results illustrate the potential of employing UAV-UGV teams for radiation mapping in scenarios where estimates of the radiation environment must be obtained quickly.
- In Chapter 3, particle filtering algorithm is proposed for radiological source term estimation. In light of the complexity of the likelihood calculations, the continuous parameter estimation problem is reduced in the spatial dimensions to a discrete estimation problem. This allows measurements for each candidate source-measurement location pair to be simulated offline for a reference source strength using a radiation transport model of any desired complexity. The algorithm also includes a dynamic component that adjusts the number of particles in real-time to achieve a suitable balance between estimation accuracy and filter runtime. The proposed methodology may be applied to other estimation problems outside of radiological applications and may be advantageous whenever likelihood calculations involve complex models that

are better suited for offline rather than online simulation.

- Monte Carlo simulation results illustrate that the particle filter is capable of producing accurate source term estimates even in obstacle-rich settings with relatively large numbers of sources (up to eight). Furthermore, simulation results show that the proposed particle filter is reasonably robust to uncertainty in absorption coefficients, and there is a drastic improvement over the state of the art in both accuracy and speed.
- The performance of the DDPF was also examined in lab scale experiments. Using a spectral instrument to simulate a counting instrument, the DDPF was found to perform as well as it did in the Monte Carlo studies. These experiments also clearly revealed shortcomings to the DDPF when handling varying isotopes and low quality measurements.
- Chapter 4 sought to alleviate the shortcomings of the DDPF when handling hardware measurements. A gamma spectrum unfolding algorithm was implemented to refine the input into the particle filter. This also allowed for the use of multiple attenuation kernels simultaneously. The entire algorithm was converted into the logarithmic domain to manage low quality measurements. All of these additions allowed for the rapid and accurate localization and identification of multiple isotopes in lab scale experiments.

## **5.2 Recommended Future Work**

### 5.2.1 Optimal Kernel Generation Parameters

The current method of generating the attenuation kernels involves two regular grids for the measurement and source locations. The regularity of the source grid keeps the resampling step unbiased; however, the measurement grid could be computed at any set of points without affecting inference. Using the knowledge of obstacle shapes and locations, measurements could be prescribed to positions that maximize the unobstructed line of sight in

the environment. This would allow for measurements to be less attenuated overall, thus producing higher quality measurements and minimizing any inaccuracies with the absorption parameters of the material.

### 5.2.2 Information Driven Active Search

In a search scenario, time matters. Active search algorithms have repeatedly been proven to reduce search times [33], [41], [105]. Fortunately, there are several frameworks and metrics to build upon [41], [106]. These algorithms would allow for efficient real time search with multiple observers. These methods would first be evaluated in simplified Monte Carlo simulations before being deployed on the final hardware.

### 5.2.3 MCNP Kernel Generation Pipeline

The current transport model is primitive compared to state of the art Monte Carlo N-Particle Transport (MCNP) software. Work has begun to develop a pipeline to perform environment characterization and attenuation kernel generation to rapidly model a new environment. Current plans would be to leverage DAGMC: Direct Accelerated Geometry Monte Carlo [53]. This process would also allow for further validation of the current transport model. The runtime of Monte Carlo N-Particle Transport Code (MCNP) for our work would be investigated, and if it found unacceptable, further work will be done to refine, generalize, and expand the current transport model.

### 5.2.4 Measurements in Motion

Current hardware validation of this algorithm has measurements taken at a standstill for a fixed period of time. Ideally, observers should be constantly moving. A continuous position and count rate datastream would make for a large number of measurement locations for attenuation kernels. There exists prior work on methods of spatial deconvolution or mesh generation to allow for continuous measurements to inform a hypothetical measurement at

a discrete point, so perhaps this would only be a minor challenge of implementation.

### 5.2.5 Hardware Development

Development of the heterogeneous UAV and UGV teams would need to be performed to adequately test their performance. However, a much more significant contribution respective of hardware would be the addition of models to deal with new types of detectors. One specific improvement could be the adjustment of the measurement model such that it can handle a directional detector (e.g. a collimated detector). Improving the ability of the inference algorithms to work with a broader array of detectors would be much more valuable than developing any individual robotic platform.

### 5.2.6 Source Measurement Factor Graph Integration

A Simultaneous Localization and Mapping (SLAM) system has been developed which could use information from the particle filter to inform vehicle positions so long as the sources are static in the environment. This would allow the particle filter estimates and radiation sensors be used to help orient and localize the vehicles position. This could be an interesting addition to scenarios where a vehicle is GPS denied.

## REFERENCES

- [1] M. K. Hubbert *et al.*, “Nuclear energy and the fossil fuel,” in *Drilling and production practice*, American Petroleum Institute, 1956.
- [2] *Budget*, <https://www.energy.gov/nnsa/budget>, Energy.gov.
- [3] *Our budget*, <https://www.energy.gov/ne/our-budget>, Energy.gov.
- [4] B. K. Sovacool, “A critical evaluation of nuclear power and renewable electricity in asia,” *Journal of Contemporary Asia*, vol. 40, no. 3, pp. 369–400, 2010.
- [5] G. I. Balatsky and W. R. Severe, “Illicit trafficking of radioactive and nuclear materials,” in *Nuclear Safeguards, Security, and Nonproliferation*, Elsevier, 2019, pp. 357–387.
- [6] T. S. Carless, K. Redus, and R. Dryden, “Estimating nuclear proliferation and security risks in emerging markets using bayesian belief networks,” *Energy Policy*, vol. 159, p. 112 549, 2021.
- [7] Y. Funabashi and K. Kitazawa, “Fukushima in review: A complex disaster, a disastrous response,” *Bulletin of the Atomic Scientists*, vol. 68, no. 2, pp. 9–21, 2012.
- [8] A. Thatcher, A. C. Vasconcelos, and D. Ellis, “An investigation into the impact of information behaviour on information failure: The fukushima daiichi nuclear power disaster,” *International Journal of Information Management*, vol. 35, no. 1, pp. 57–63, 2015.
- [9] Y. Sanada, T. Sugita, Y. Nishizawa, A. Kondo, and T. Torii, “The aerial radiation monitoring in japan after the fukushima daiichi nuclear power plant accident,” *Prog. Nucl. Sci. Technol.*, vol. 4, no. 7, 2014.
- [10] S. Kawatsuma, M. Fukushima, and T. Okada, “Emergency response by robots to fukushima-daiichi accident: Summary and lessons learned,” *Industrial Robot: An International Journal*, 2012.
- [11] K. Ohno, S. Kawatsuma, T. Okada, E. Takeuchi, K. Higashi, and S. Tadokoro, “Robotic control vehicle for measuring radiation in fukushima daiichi nuclear power plant,” in *2011 IEEE International Symposium on Safety, Security, and Rescue Robotics*, IEEE, 2011, pp. 38–43.
- [12] J. Abouaf, “Trial by fire: Teleoperated robot targets chernobyl,” *IEEE Computer Graphics and Applications*, vol. 18, no. 4, pp. 10–14, 1998.

- [13] D. Connor, P. G. Martin, and T. B. Scott, “Airborne radiation mapping: Overview and application of current and future aerial systems,” *International journal of remote sensing*, vol. 37, no. 24, pp. 5953–5987, 2016.
- [14] D. T. Connor, K. Wood, P. G. Martin, *et al.*, “Radiological mapping of post-disaster nuclear environments using fixed-wing unmanned aerial systems: A study from chornobyl,” *Frontiers in Robotics and AI*, p. 149, 2020.
- [15] A. West, I. Tsitsimpelis, M. Licata, *et al.*, “Use of gaussian process regression for radiation mapping of a nuclear reactor with a mobile robot,” *Scientific Reports*, vol. 11, no. 1, pp. 1–11, 2021.
- [16] P. G. Martin, S. Kwong, N. T. Smith, *et al.*, “3d unmanned aerial vehicle radiation mapping for assessing contaminant distribution and mobility,” *International journal of applied earth observation and geoinformation*, vol. 52, pp. 12–19, 2016.
- [17] M. Hutchinson, H. Oh, and W.-H. Chen, “A review of source term estimation methods for atmospheric dispersion events using static or mobile sensors,” *Information Fusion*, vol. 36, pp. 130–148, 2017.
- [18] L. Delle Monache, J. K. Lundquist, B. Kosović, *et al.*, “Bayesian inference and markov chain monte carlo sampling to reconstruct a contaminant source on a continental scale,” *Journal of Applied Meteorology and Climatology*, vol. 47, no. 10, pp. 2600–2613, 2008.
- [19] F. K. Chow, B. Kosović, and S. Chan, “Source inversion for contaminant plume dispersion in urban environments using building-resolving simulations,” *Journal of applied meteorology and climatology*, vol. 47, no. 6, pp. 1553–1572, 2008.
- [20] G. Katata, M. Chino, T. Kobayashi, *et al.*, “Detailed source term estimation of the atmospheric release for the fukushima daiichi nuclear power station accident by coupling simulations of an atmospheric dispersion model with an improved deposition scheme and oceanic dispersion model,” *Atmospheric chemistry and physics*, vol. 15, no. 2, pp. 1029–1070, 2015.
- [21] B. Ristic, M. Morelande, and A. Gunatilaka, “Information driven search for point sources of gamma radiation,” *Signal Processing*, vol. 90, no. 4, pp. 1225–1239, 2010.
- [22] M. Hutchinson, C. Liu, P. Thomas, and W.-H. Chen, “Unmanned aerial vehicle-based hazardous materials response: Information-theoretic hazardous source search and reconstruction,” *IEEE Robotics & Automation Magazine*, vol. 27, no. 3, pp. 108–119, 2019.

- [23] Y. Kantaros, B. Schlotfeldt, N. Atanasov, and G. J. Pappas, “Asymptotically optimal planning for non-myopic multi-robot information gathering,” in *Robotics: Science and Systems*, 2019, pp. 22–26.
- [24] C. Di Franco and G. Buttazzo, “Energy-aware coverage path planning of uavs,” in *2015 IEEE international conference on autonomous robot systems and competitions*, IEEE, 2015, pp. 111–117.
- [25] E. Galceran and M. Carreras, “A survey on coverage path planning for robotics,” *Robotics and Autonomous systems*, vol. 61, no. 12, pp. 1258–1276, 2013.
- [26] N. R. Authority, “Results of airborne monitoring by the ministry of education, culture,” *Sports, Science and Technology and the US Department of Energy*, 2018.
- [27] P. Guss, “Doe response to the radiological release from the fukushima dai-ichi nuclear power plant,” in *NEI RETS/REMP Workshop*, 2011.
- [28] Y. Sanada, A. Kondo, T. Sugita, *et al.*, “Radiation monitoring using an unmanned helicopter in the evacuation zone around the fukushima daiichi nuclear power plant,” *Exploration Geophysics*, vol. 45, no. 1, pp. 3–7, 2014.
- [29] A. Proctor, “Aerial radiological surveys,” Bechtel Nevada Corp., NV (US); Nevada Test Site, Mercury, NV (US), Tech. Rep., 1997.
- [30] S. Tsuda, T. Yoshida, M. Tsutsumi, and K. Saito, “Characteristics and verification of a car-borne survey system for dose rates in air: Kurama-ii,” *Journal of environmental radioactivity*, vol. 139, pp. 260–265, 2015.
- [31] T. Lazna, P. Gabrlik, T. Jilek, and L. Zalud, “Cooperation between an unmanned aerial vehicle and an unmanned ground vehicle in highly accurate localization of gamma radiation hotspots,” *International Journal of Advanced Robotic Systems*, vol. 15, no. 1, p. 1 729 881 417 750 787, 2018.
- [32] G. Christie, A. Shoemaker, K. Kochersberger, P. Tokekar, L. McLean, and A. Leonessa, “Radiation search operations using scene understanding with autonomous uav and ugv,” *Journal of Field Robotics*, vol. 34, no. 8, pp. 1450–1468, 2017.
- [33] J. Peterson, W. Li, B. Cesar-Tondreau, *et al.*, “Experiments in unmanned aerial vehicle/unmanned ground vehicle radiation search,” *Journal of Field Robotics*, vol. 36, no. 4, pp. 818–845, 2019.
- [34] K. Kochersberger, K. Kroeger, B. Krawiec, E. Brewer, and T. Weber, “Post-disaster remote sensing and sampling via an autonomous helicopter,” *Journal of Field Robotics*, vol. 31, no. 4, pp. 510–521, 2014.

- [35] J. W. Howse, L. O. Ticknor, and K. R. Muske, “Least squares estimation techniques for position tracking of radioactive sources,” *Automatica*, vol. 37, no. 11, pp. 1727–1737, 2001.
- [36] A. Gunatilaka, B. Ristic, and R. Gailis, “On localisation of a radiological point source,” in *2007 Information, Decision and Control*, IEEE, 2007, pp. 236–241.
- [37] Y. Liu, Y. Xuan, D. Zhang, and S. Zou, “Localizing unknown radiation sources by unscented particle filtering based on divide-and-conquer sampling,” *Journal of Nuclear Science and Technology*, pp. 1–13, 2022.
- [38] N. S. Rao, S. Sen, N. J. Prins, *et al.*, “Network algorithms for detection of radiation sources,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 784, pp. 326–331, 2015.
- [39] N. Pinkam, A. Elibol, and N. Y. Chong, “Informative mobile robot exploration for radiation source localization with a particle filter,” in *2020 Fourth IEEE International Conference on Robotic Computing (IRC)*, IEEE, 2020, pp. 107–112.
- [40] M. Hutchinson, C. Liu, and W.-H. Chen, “Information-based search for an atmospheric release using a mobile robot: Algorithm and experiments,” *IEEE Transactions on Control Systems Technology*, vol. 27, no. 6, pp. 2388–2402, 2018.
- [41] B. Ristic and A. Gunatilaka, “Information driven localisation of a radiological point source,” *Information fusion*, vol. 9, no. 2, pp. 317–326, 2008.
- [42] A. Fathi and S. F. Masoudi, “Lost gamma source detection algorithm based on convolutional neural network,” *Nuclear Engineering and Technology*, vol. 53, no. 11, pp. 3764–3771, 2021.
- [43] P. Gong, X.-B. Tang, X. Huang, *et al.*, “Locating lost radioactive sources using a uav radiation monitoring system,” *Applied radiation and isotopes*, vol. 150, pp. 1–13, 2019.
- [44] J.-C. Chin, D. K. Yau, and N. S. Rao, “Efficient and robust localization of multiple radiation sources in complex environments,” in *2011 31st International Conference on Distributed Computing Systems*, IEEE, 2011, pp. 780–789.
- [45] Z. Cook, “Radiative contour mapping using uas swarm,” Ph.D. dissertation, University of Nevada, Las Vegas, 2017.
- [46] Z. Cook, M. Kazemeini, A. Barzilov, and W. Yim, “Low-altitude contour mapping of radiation fields using uas swarm,” *Intelligent Service Robotics*, vol. 12, no. 3, pp. 219–230, 2019.

- [47] J. R. Vavrek, D. Hellfeld, M. S. Bandstra, *et al.*, “Reconstructing the position and intensity of multiple gamma-ray point sources with a sparse parametric algorithm,” *IEEE Transactions on Nuclear Science*, vol. 67, no. 11, pp. 2421–2430, 2020.
- [48] M. Greiff, E. Rofors, A. Robertsson, R. Johansson, and R. Tyllström, “Gamma-ray imaging with spatially continuous intensity statistics,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2021, pp. 5257–5262.
- [49] M. Bandstra, D. Hellfeld, J. Vavrek, *et al.*, “Improved gamma-ray point source quantification in three dimensions by modeling attenuation in the scene,” *IEEE Transactions on Nuclear Science*, vol. 68, no. 11, pp. 2637–2646, 2021.
- [50] R. B. Anderson, M. Pryor, A. Abeyta, and S. Landsberger, “Mobile robotic radiation surveying with recursive bayesian estimation and attenuation modeling,” *IEEE Transactions on Automation Science and Engineering*, 2020.
- [51] B. Ristic, M. Morelande, and A. Gunatilaka, “A controlled search for radioactive point sources,” in *2008 11th International Conference on Information Fusion*, IEEE, 2008, pp. 1–5.
- [52] P. Wilson, R. Feder, U. Fischer, *et al.*, “State-of-the-art 3-d radiation transport methods for fusion energy systems,” *Fusion Engineering and Design*, vol. 83, no. 7-9, pp. 824–833, 2008.
- [53] P. P. Wilson, T. J. Tautges, J. A. Kraftcheck, B. M. Smith, and D. L. Henderson, “Acceleration techniques for the direct use of cad-based geometry in fusion neutronics analysis,” *Fusion Engineering and Design*, vol. 85, no. 10-12, pp. 1759–1765, 2010.
- [54] V. Elvira, J. Míguez, and P. M. Djurić, “Adapting the number of particles in sequential monte carlo methods through an online scheme for convergence assessment,” *IEEE Transactions on Signal Processing*, vol. 65, no. 7, pp. 1781–1794, 2016.
- [55] J. R. Bourne, M. N. Goodell, X. He, J. A. Steiner, and K. K. Leang, “Decentralized multi-agent information-theoretic control for target estimation and localization: Finding gas leaks,” *The International Journal of Robotics Research*, vol. 39, no. 13, pp. 1525–1548, 2020.
- [56] H. Friedman, “Geiger counter tubes,” *Proceedings of the IRE*, vol. 37, no. 7, pp. 791–808, 1949.
- [57] A. Richmond, “Financially efficient ore selections incorporating grade uncertainty,” *Mathematical Geology*, vol. 35, no. 2, pp. 195–215, 2003.

- [58] B. A. Khuwaileh and W. A. Metwally, “Gaussian process approach for dose mapping in radiation fields,” *Nuclear Engineering and Technology*, vol. 52, no. 8, pp. 1807–1816, 2020.
- [59] R. A. Viscarra Rossel, R. Webster, and D. Kidd, “Mapping gamma radiation and its uncertainty from weathering products in a tasmanian landscape with a proximal sensor and random forest kriging,” *Earth Surface Processes and Landforms*, vol. 39, no. 6, pp. 735–748, 2014.
- [60] S. Kemna, J. G. Rogers, C. Nieto-Granda, S. Young, and G. S. Sukhatme, “Multi-robot coordination through dynamic voronoi partitioning for informative adaptive sampling in communication-constrained environments,” in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2017, pp. 2124–2130.
- [61] N. Fung, J. Rogers, C. Nieto, H. I. Christensen, S. Kemna, and G. Sukhatme, “Coordinating multi-robot systems through environment partitioning for adaptive informative sampling,” in *2019 International Conference on Robotics and Automation (ICRA)*, IEEE, 2019, pp. 3231–3237.
- [62] “Gaussian process regression models.” (2014).
- [63] J. Machta, “Entropy, information, and computation,” *American Journal of Physics*, vol. 67, no. 12, pp. 1074–1077, 1999.
- [64] T. Van Erven and P. Harremoës, “Rényi divergence and kullback-leibler divergence,” *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 3797–3820, 2014.
- [65] H. Choset and P. Pignon, “Coverage path planning: The boustrophedon cellular decomposition,” in *Field and service robotics*, Springer, 1998, pp. 203–209.
- [66] A. Martin and S. Harbison, “An introduction to radiation protection,” Environmental Safety Group, Associated Nuclear Services, Epsom, Tech. Rep., 1986.
- [67] T. Nicholas, *Measurement and detection of radiation*, 1995.
- [68] OpenStreetMap contributors, *Planet dump retrieved from <https://planet.osm.org>, <https://www.openstreetmap.org>*, 2017.
- [69] J. H. Clark, “Hierarchical geometric models for visible surface algorithms,” *Communications of the ACM*, vol. 19, no. 10, pp. 547–554, 1976.
- [70] G. Kitagawa, “Monte carlo filter and smoother for non-gaussian nonlinear state space models,” *Journal of computational and graphical statistics*, vol. 5, no. 1, pp. 1–25, 1996.

- [71] J. D. Hol, T. B. Schon, and F. Gustafsson, "On resampling algorithms for particle filters," in *2006 IEEE nonlinear statistical signal processing workshop*, IEEE, 2006, pp. 79–82.
- [72] F. Daum and J. Huang, "Particle degeneracy: Root cause and solution," in *Signal Processing, Sensor Fusion, and Target Recognition XX*, SPIE, vol. 8050, 2011, pp. 367–377.
- [73] F. Daum and J. Huang, "Particle flow for nonlinear filters with log-homotopy," in *Signal and Data Processing of Small Targets 2008*, International Society for Optics and Photonics, vol. 6969, 2008, p. 696 918.
- [74] N. Oudjane and C. Musso, "Progressive correction for regularized particle filters," in *Proceedings of the Third International Conference on Information Fusion*, IEEE, vol. 2, 2000, THB2–10.
- [75] T. Hanselmann, Y. Zhang, M. Morelande, *et al.*, "Self-localization in wireless sensor networks using particle filtering with progressive correction," in *2010 5th International ICST Conference on Communications and Networking in China*, IEEE, 2010, pp. 1–6.
- [76] J. Hubbell, "Photon cross sections, attenuation coefficients and energy absorption coefficients," *National Bureau of Standards Report NSRDS-NBS29*, Washington DC, 1969.
- [77] C. Gentner, S. Zhang, and T. Jost, "Log-pf: Particle filtering in logarithm domain," *Journal of Electrical and Computer Engineering*, vol. 2018, pp. 1–11, 2018.
- [78] F. Li, Z. Gu, L. Ge, *et al.*, "Review of recent gamma spectrum unfolding algorithms and their application," *Results in Physics*, vol. 13, p. 102 211, 2019.
- [79] H. Li, X. Jiang, and Z. Li, "Robust estimation in gaussian filtering for engineering surface characterization," *Precision Engineering*, vol. 28, no. 2, pp. 186–193, 2004.
- [80] Z. Meng-Hua, L. Liang-Gang, Q. Dong-Xu, Y. Zhong, and X. Ao-Ao, "Least square fitting of low resolution gamma ray spectra with cubic b-spline basis functions," *Chinese Physics C*, vol. 33, no. 1, p. 24, 2009.
- [81] X. Yang, L. Tan, and L. He, "A robust least squares support vector machine for regression and classification with noise," *Neurocomputing*, vol. 140, pp. 41–52, 2014.
- [82] D. Bianchi, C. Lenauer, G. Betz, and A. Vernes, "A wavelet filtering method for cumulative gamma spectroscopy used in wear measurements," *Applied Radiation and Isotopes*, vol. 120, pp. 51–59, 2017.

- [83] M. Safari, F. A. Davani, H. Afarideh, S. Jamili, and E. Bayat, “Discrete fourier transform method for discrimination of digital scintillation pulses in mixed neutron-gamma fields,” *IEEE transactions on nuclear science*, vol. 63, no. 1, pp. 325–332, 2016.
- [84] B. Minty, “Accurate noise reduction for airborne gamma-ray spectrometry,” *Exploration Geophysics*, vol. 34, no. 3, pp. 207–215, 2003.
- [85] F. Zhang, X. Tang, A. Tong, B. Wang, and J. Wang, “An automatic baseline correction method based on the penalized least squares method,” *Sensors*, vol. 20, no. 7, p. 2015, 2020.
- [86] D. Erb, *pybaselines: A Python library of algorithms for the baseline correction of experimental data*, version 1.0.0.
- [87] Y. Zheng, D. Tian, K. Liu, *et al.*, “Peak detection of tof-sims using continuous wavelet transform and curve fitting,” *International Journal of Mass Spectrometry*, vol. 428, pp. 43–48, 2018.
- [88] Y. Hu, W. Li, and J. Hu, “Resolving overlapped spectra with curve fitting,” *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, vol. 62, no. 1-3, pp. 16–21, 2005.
- [89] K. Lam and W. Zhang, “Gamma peak search and peak fitting algorithm for a low-resolution detector with applications in gamma spectroscopy,” *Journal of Radioanalytical and Nuclear Chemistry*, vol. 322, pp. 255–261, 2019.
- [90] M. Bandstra, A. Persaud, J. Curtis, *et al.*, *Becquerel (bq) v0.4.0*, Sep. 2021.
- [91] M. Mariscotti, “A method for automatic identification of peaks in the presence of background and its application to spectrum analysis,” *Nuclear Instruments and Methods*, vol. 50, no. 2, pp. 309–320, 1967.
- [92] P. J. Statham, “A comparative study of techniques for quantitative analysis of the x-ray spectra obtained with a si (li) detector,” *X-Ray Spectrometry*, vol. 5, no. 1, pp. 16–28, 1976.
- [93] S. Gerasimov, “Simple method for background subtraction in gamma-ray spectra,” *International journal of radiation applications and instrumentation. Part A. Applied radiation and isotopes*, vol. 43, no. 12, pp. 1529–1531, 1992.
- [94] M. Zhu, L. Liu, Y. Cheng, T. Dong, Z. You, and A. Xu, “Iterative estimation of the background in noisy spectroscopic data,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 602, no. 2, pp. 597–599, 2009.

- [95] C. Ryan, E. Clayton, W. Griffin, S. Sie, and D. Cousens, “Snip, a statistics-sensitive background treatment for the quantitative analysis of pixe spectra in geoscience applications,” *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, vol. 34, no. 3, pp. 396–402, 1988.
- [96] C. Hampton, B. Lian, and W. C. McHarris, “Fast-fourier-transform spectral enhancement techniques for  $\gamma$ -ray spectroscopy,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 353, no. 1-3, pp. 280–284, 1994.
- [97] G. F. Knoll, *Radiation detection and measurement*. John Wiley & Sons, 2010.
- [98] A. Michael, Y. N. Zhou, M. Yavuz, and M. I. Khan, “Deconvolution of overlapping peaks from differential scanning calorimetry analysis for multi-phase niti alloys,” *Thermochimica Acta*, vol. 665, pp. 53–59, 2018.
- [99] M. Morháč, J. Kliman, V. Matoušek, M. Veselský, and I. Turzo, “Efficient one-and two-dimensional gold deconvolution and its application to  $\gamma$ -ray spectra decomposition,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 401, no. 2-3, pp. 385–408, 1997.
- [100] R. P. Gardner, X. Ai, C. R. Peeples, *et al.*, “Use of an iterative convolution approach for qualitative and quantitative peak analysis in low resolution gamma-ray spectra,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 652, no. 1, pp. 544–549, 2011.
- [101] E. Larionova and S. Romanenko, “Determination of overlapping peaks heights by tangent method,” in *2012 7th International Forum on Strategic Technology (IFOST)*, IEEE, 2012, pp. 1–4.
- [102] *Sigma 25/50 datasheet*, SIG-A4E-SPF-DOC-1225-Rev8-160317-2, Rev. 8, Kromek, May 2018.
- [103] Saint-Gobain, *Efficiency calculations for selected detectors*, <https://www.crystals.saint-gobain.com/sites/hps-mac3-cma-crystals/files/2021-09/Efficiency-Calculationns-Brochure.pdf>, Accessed: 2023-6-20, 2021.
- [104] IAEA, *Live chart of nuclides*, <https://www-nds.iaea.org/relnsd/vcharthtml/VChartHTML.html>, Accessed: 2023-6-20, 2023.
- [105] S. Kemp and J. Rogers, “Uav-ugv teaming for rapid radiological mapping,” in *2021 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, IEEE, 2021, pp. 92–97.

- [106] M. Park and H. Oh, “Cooperative information-driven source search and estimation for multiple agents,” *Information Fusion*, vol. 54, pp. 72–84, 2020.

## VITA

Samuel Kemp was born in Lewisville, TX and raised in Gainesville, GA. After graduating high school in 2014, Sam began attending the University of North Georgia. He then transferred to the Georgia Institute of Technology in 2016 and graduated with a B.S. in Aerospace Engineering in the Fall of 2019. During his time as an undergraduate, Sam interned with Spaceworks Enterprises, Inc. and Area-I, Inc. focusing primarily on manufacturing and testing automation. Following the completion of his bachelor's degree, Sam continued his studies at the Georgia Institute of Technology, where he pursued both an M.S. and a Ph.D. in Mechanical Engineering. His technical interests include 3D printing, mechanical design, fabrication, manufacturing, automation, controls, and estimation.