

Development of a Framework for the localization of Radioactive Sources and Evaluation Methods

Salvin George

Abstract—Write your abstract here.

Index Terms—Write up to three keywords about your work.
radioactive source localization, informed search,

I. INTRODUCTION

It is claimed that life would not be possible without radioactivity, which is an essential component of life. Life on Earth began when naturally occurring radioactivity heated the Earth's core [1]. The science of radioactivity has come a long way since Henri Becquerel discovered it in 1896, and it is now a prerequisite for many modern technologies. Radioactive elements are used in scientific studies to determine the age of rocks and fossils. For example, radioactive materials are employed in health imaging and disease treatment. In energy, radioactive materials are utilized in nuclear power plants and spacecraft.

However, the inherent properties of radioactive materials that make them valuable also present significant challenges in handling and security. These materials are invisible to the naked eye and can be easily concealed, making their detection and tracking a complex technological challenge. The risks associated with radioactive materials are compounded by their potential for malicious use, whether as weapons or as a means of environmental contamination.

A. Motivation

It is essential to develop trustworthy techniques for locating radioactive sources. First, these methods can significantly enhance security measures by helping law enforcement and security agencies spot possible threats faster. Second, quick detection of radioactive leakage is essential for preserving operational safety and safeguarding the environment and employees in industrial settings such as nuclear power plants. Localizing radioactive sources is intriguing yet challenging, highlighting the need for continued research. Strong simulation frameworks are required since testing with actual radioactive sources can be dangerous and complicated by nature. Furthermore, comprehending the complex physical behavior of radiation, particularly particle attenuation and scattering, presents intriguing modeling problems that affect simulations' accuracy and localization's efficacy. Although various methods

have been put forth, they are less feasible for rapid source detection because they are computationally intensive and necessitate broad area searches. It is very important to have a set of tools and methods that can effectively localize the radiation source without human interference, as the need for this is very relevant in recent times. This way, we can ensure public safety and environmental protection.

B. Problem Statement

The fundamental challenge in radioactive source localization lies in using sensor measurements and prior knowledge to estimate the position and intensity of unknown radiation sources within an environment. Current approaches face several key limitations: they often require an exhaustive and complete search of the entire search area, struggle with sensor noise and measurement uncertainty, perform poorly in environments with limited accessibility, and suffer from high computational complexity when dealing with dynamic and large search spaces.

The core research problem encompasses three main aspects. First, the development of a physics-accurate simulation framework that can reliably model radiation behavior, including particle attenuation and scattering effects. Second, a comprehensive evaluation framework capable of assessing and comparing different localization approaches must be created. Third, the implementation and analysis of various localization algorithms, examining their computational complexity, resource requirements, accuracy, and efficiency across different environmental conditions.

We present a framework that integrates simulation, evaluation, and algorithm implementation to tackle these challenges. The simulation component uses established radiation physics models to create realistic test scenarios, incorporating inverse square law, attenuation, and scattering effects. The evaluation framework provides quantitative metrics for assessing algorithm performance, including computational efficiency, accuracy, and robustness to noise. We implement and compare several localization methods, including rollout algorithms, entropy-based approaches, and optimization-based methods. Each method is systematically evaluated under various environmental conditions and constraints. We aim to develop an integrated system that allows for robust and flexible implementation and testing of various radiation source localization algorithms.

II. RELATED WORK

The subject of radioactive source localization has not been explored as extensively as other localization problems. The

*Submitted to the Department of Computer Science at Hochschule Bonn-Rhein-Sieg in partial fulfilment of the requirements for the degree of Master of Science in Autonomous Systems

[†]Prof. Dr. Sebastian Houben (Hochschule Bonn-Rhein-Sieg) and Claudia Bender (Fraunhofer FKIE)

[‡]Submitted in February 2025

main characteristic of the radioactive source search is about controlling the observer(s) position. The related works in the context of radioactive source localization have proposed different methods to solve the problem. Traditional methods relied heavily on using a high number of sensors in the search and then localizing the estimated source position using the least squares method [2] [3]. In the conventional methods, the observer moves through the search area following a predefined path [4]. Such a method was proposed by using uniform search [5]. Some of these autonomous methods use triangulation-based techniques to localize the source [6] [7]. These conventional methods, while functional, face several critical limitations in efficiently localizing radioactive sources. Static sensor networks and predefined search path approaches require extensive coverage of the search area, making them more time-consuming and computationally expensive. As noted in recent studies, traversal-based methods, though accurate, suffer from extremely low search efficiency [8]. These methods become highly unreliable with larger search areas as they lack the intelligence to adapt to the environment.

Bayesian estimation method is a commonly used method in the context of radioactive source localization. These methods are widely adopted due to their ability to handle uncertainties and incorporate prior information in the localization process [8]. According to the Bayesian criterion, the localization of an unknown radioactive source can be determined by solving the posterior probability distribution of the source parameter vector, which can be represented through various methods, including the particle filter [9] [4] [10] and other probabilistic methods. The particle filter-based method, Ling et al. [10] highlight that the particle filter stands out as a practical approach that approximates the posterior distribution by maintaining and iteratively updating a set of weighted particles. At each step, the particles' weights are updated based on how likely the observed measurements are under each particle's hypothesis, and low-likelihood particles are replaced by higher-likelihood ones. As shown by Ristic et al. [9], this resampling process helps to maintain particle diversity while converging towards the most likely source parameters. They extended the particle filter approach further by introducing the information-driven strategy. Their method combines sequential Bayesian estimation with Fisher information-based observer control to optimize measurement collection. Initially, when source detection confidence is low, the observer moves through the search area following an exploration pattern. After the detection threshold is reached with sufficient probability, the observer's motion and exposure time are controlled to maximize the information gain of future measurements. This method helps to guide the observer to positions that are expected to provide the most information about the source parameters. However, a key limitation of Fisher information that emerged in the study of Ristic et al. [4] is that it cannot be used before the source detection. To overcome this limitation, Ristic et al. proposed a method that uses Rényi divergence between current and predicted future posterior densities as the information gain metric. This Rényi divergence enables observer control even before the source detection by considering the complete probability density function rather

than just parameter estimates. This allows for more optimal data collection and observer control, leading to faster and more accurate source localization even before the source is detected.

Rollout algorithms, introduced by Bertsekas, represent a sequential optimization approach where variables are optimized one after another. Starting with a base policy or heuristic, rollout algorithms construct an improved policy through a one-step lookahead, often yielding significantly better performance than the original heuristic while maintaining implementation simplicity [11]. Several researchers have explored rollout algorithms for source localization and path planning. Hoffmann et al. proposed a rollout-based path planner that optimizes mobile sensor trajectories for RF source localization using lookahead policies and cost-based optimization. While their focus was on RF sources, the approach is generalizable to other source types, including radioactive sources, due to their similar signal characteristics [12]. Tian et al. developed a multi-step lookahead policy for UAV surveillance that incorporates a layered decision framework to balance multiple objectives, including safe navigation and target tracking. Their work demonstrates the effectiveness of rollout policies in anticipating future states and making informed decisions for path planning tasks [13]. These works demonstrate the potential of rollout algorithms for efficient path planning and source localization. The success of Hoffmann et al.'s approach with RF sources and Tian et al.'s implementation for UAV surveillance suggests that adapting rollout mechanisms shows significant potential for radiation source localization. The method's ability to improve upon base heuristics while maintaining computational efficiency makes it especially suitable for radiation scenarios where quick localization is crucial. Furthermore, the one-step lookahead policy could help navigate the complex radiation fields while accounting for particle attenuation and scattering effects.

[learning methods]

III. BACKGROUND

This is an optional section in which you can introduce concepts, terms, or methods that are important for understanding your approach and that would not directly fit in Sec. IV. If you do not need this section, comment out the respective line in *report.tex*.

IV. METHODOLOGY

A. Simulation Framework

A set of simulation frameworks was investigated to select the best simulation that is available for the project. Gazebo was selected as the simulation based on its compatibility with ROS2 and the availability of the plugins that are already made available due to the familiarity of the platform. The SJTU drone system was selected as the drone model for the simulation as it was lightweight and also easy to work with. A new radiation sensor plugin and radiation source plugin were developed to simulate the radiation source and the sensor. This plugin can be easily integrated into the simulation and can be used to simulate the radiation source and the sensor.

To accurately represent realistic radiation behavior, the radiation source is modeled as a point gamma radiation source

that emits radiation uniformly in all directions. The radiation sensor receives and publishes data in the form of photon count rates; however, this count rate is subject to attenuation caused by trees and the air surrounding the sensor.

The attenuation from trees is implemented through ray tracing, which helps to determine if the sensor has a clear line of sight to the radiation source. If trees obstruct the line of sight, the radiation is attenuated based on the type of tree and a specific attenuation factor. The positions of the trees are generated dynamically according to the map size, while ensuring a maintained specific distance between each tree.

B. Evaluation Framework

A new interface for handling the simulation and experiment metrics was developed. The interface is designed with PyQt5, a Python library for creating graphical user interfaces. Two interfaces have been developed that allow one to either handle all the parameters in the workspace or visualize any number of algorithms running within the workspace.

1) *Parameter Editor*: The parameter editor application serves multiple purposes beyond the current project. It was developed to provide a unified interface that consolidates all configuration files in YAML format. To ensure compatibility with ROS2 packages, the interface can load the names of the ROS2 packages of interest from a YAML file. This file is then processed to identify which packages should be searched for within the root of the ROS2 workspace. The identified YAML files are sorted according to the package or workspace in which they were found, allowing the user to select the package of interest. Once selected, the YAML files are loaded and displayed in a tabular format. This interface enables the users to add, remove and save changes to the YAML files.

Another feature of this interface is its ability to identify common parameters shared across different algorithms. These parameters can include the search area, source location, ROS2 topic names, and more. Changes made in this section of the application are applied to all instances of the same parameters across the project workspace. This functionality enables users to transition or migrate the project to different simulation frameworks or configurations with varying parameters.

2) *Evaluation Visualizer*: Each algorithm, upon successful completion, is saved in a standardized JSON file format. All metrics contained in these files are consistent across all algorithms, which facilitates easier comparison and visualization of the results.

The Evaluation Visualizer is an interface designed to visualize these results based on a set of configurations. When the interface is opened, the user can select the input directory containing the saved JSON files, followed by a directory where the evaluated files will be saved. Once the input directory is selected, the interface loads all JSON files and categorizes them by algorithm name. From this interface, users can specify the number of experiments to be evaluated from the total runs. After processing the analysis, the results of the algorithm runs are evaluated and displayed in the next tab, with a copy saved locally for future visualization.

The algorithms are assessed together based on predefined performance metrics, and various plots representing additional

metrics from the runs are generated along with a report that tabulates the evaluation results for each algorithm.

This interface also comes with an extra tab that gives the option to generate different types of plots based on the metrics that can be selected for both the x and y-axis. The corresponding plot will be generated and displayed within the same window.

This application additionally provides a feature to do statistical analysis on the results. The users can select the metrics they want to perform the analysis on, and this will generate a statistical analysis report that can be read and interpreted by the user.

C. Algorithms

1) *Information-Gain-based Algorithm*: The proposed method combines a particle filter for sequential Bayesian estimation with information-driven observer control using Rényi divergence. The particle filter approximates the joint posterior density of the source parameters through a set of weighted particles, while Rényi divergence guides measurement collection to maximize information gain. This method operates in two distinct phases: an initial exploration phase and an information-driven exploitation phase. Unlike previous approaches that use Fisher information for observer control after detection [9], this method uses Rényi divergence throughout both phases. This choice is motivated by ability of Rényi divergence to enable optimal measurement collection by considering the complete probability density function rather than just parameter estimates.

a) *Particle Filter Framework*: The source state $X = [x, y, I]^T$ comprises 2D position (x, y) and intensity I . The posterior distribution is approximated using weighted particles:

$$p(X|z_{1:k}) \approx \sum_{i=1}^N w_k^i \delta(X - X_k^i) \quad (1)$$

where X is the state vector containing position and intensity, $z_{1:k}$ represents all measurements up to time k , w_k^i is the normalized weight of particle i , N is the total number of particles, $\delta(\cdot)$ is the Dirac delta function

The radiation measurement model incorporates inverse square law decay and atmospheric attenuation:

$$\lambda(X, x_k) = \left(\frac{I}{d^2(x_k, X)} + \mu_b \right) \tau_k \cdot \exp(-\beta d(x_k, X)) \quad (2)$$

where, $\lambda(X, x_k)$ is the expected measurement rate at position x_k , I is the source intensity, $d(x_k, X)$ is the distance from measurement position to source, μ_b is the background radiation rate, τ_k is the measurement exposure time, β is the atmospheric attenuation coefficient.

b) *Two-Phase Search Strategy*: During the exploration phase, the detection probability is monitored using:

$$P(\text{detection}|z_k) = \frac{r_k}{1 + r_k}, \quad r_k = \frac{\frac{1}{N} \sum_{i=1}^N P(z_k|X_k^i)}{P(z_k|\text{background})} \quad (3)$$

where, $P(\text{detection}|z_k)$ is the probability of source presence, r_k is the likelihood ratio, $P(z_k|X_k^i)$ is the measurement likelihood for particle i , $P(z_k|\text{background})$ is the background-only likelihood.

In the exploitation phase, measurement locations are optimized using Rényi divergence:

$$D_\alpha(x'_k) = \frac{1}{\alpha - 1} \log \left(\frac{g_\alpha(x'_k)}{g_1(x'_k)^\alpha} \right) \cdot E(x'_k) \quad (4)$$

with:

$$g_\alpha(x'_k) = \frac{1}{N} \sum_{i=1}^N \exp(\alpha \hat{z}_k^i) \quad (5)$$

where, $D_\alpha(x'_k)$ is the Rényi divergence for candidate position x'_k , α is the Rényi divergence parameter ($\alpha > 0$), $g_\alpha(x'_k)$ is the α -weighted prediction sum, \hat{z}_k^i is the normalized predicted measurement, $E(x'_k)$ is the exploration bonus term.

Rényi divergence generalizes mutual information and provides a measure of how well a candidate measurement position x'_k can distinguish between the current posterior distribution of the source state and a uniform or background distribution.

c) *Sequential State Estimation*: The particle weights are updated using:

$$w_k^i = \frac{\tilde{w}_k^i}{\sum_{j=1}^N \tilde{w}_k^j} \quad (6)$$

with unnormalized weights:

$$\tilde{w}_k^i = w_{k-1}^i \cdot \exp(\log L_k^i - \max_j \log L_k^j) \cdot c_k^i \quad (7)$$

where, w_k^i is the normalized weight of particle i at time k , \tilde{w}_k^i is the unnormalized weight, L_k^i is the measurement likelihood for particle i , c_k^i is the geometric consistency term. The geometric consistency term c_k^i evaluates how well each particle explains the measurement patterns obtained from different observation angles, complementing the circular measurement strategy used for position selection. This circular measurement pattern is particularly significant as demonstrated by Ristic et al. [9], who in their first method showed that measuring in a circle around the estimated source position helps discriminate the true source location, since getting too close to the source can result in higher measurement uncertainty and estimation errors due to the inverse square law behavior of radiation.

This framework enables efficient source detection and localization by combining systematic exploration with information-driven measurement optimization. The particle filter handles the non-linear measurement model while maintaining uncertainty estimation throughout the process.

Figure 1 represents the run of the information gain-based radiation source localization algorithm for a source located at (30, 50). The particles are uniformly distributed across the search area initially, and the algorithm converges on the true source location after the source has been detected using 3 starting from figure 1.b. The final source location theoretically is the point where the particles converge as in figure 1.d. This run localized the source at a distance of 3 meters from the actual source location. [ADD RESAMPLING AND UPDATE MECHANISM]

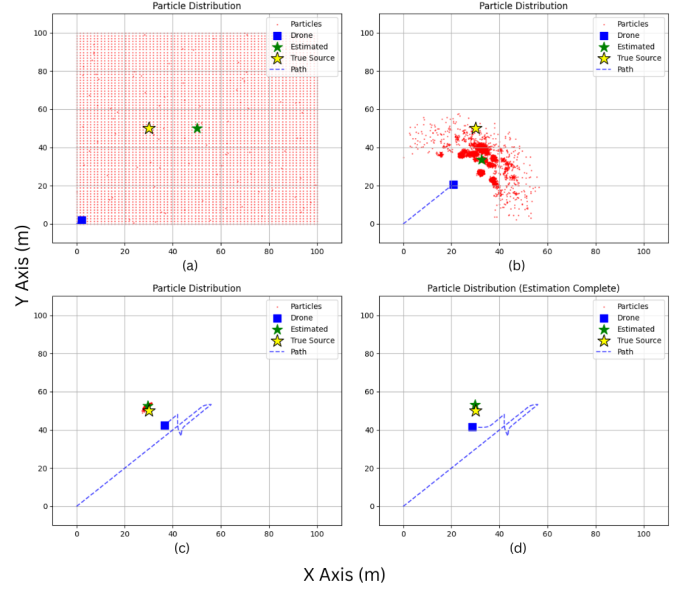


Fig. 1: Information gain-based radiation source localization for source location (30, 50): a. Initial particle distribution across search area, b. Particle resampling on source detection, c. Convergence of particles near true source location, and d. Final source position estimation with completed path

2) *Rollout-based Algorithm*: The rollout-algorithm in dynamic optimization problems, represent a form of approximate dynamic programming that uses a base heuristic through lookahead and policy iteration [11]. A rollout algorithm simulates multiple trajectories that represent possible future states from the current state, and uses the base heuristic to evaluate these trajectories and chooses the action that maximizes the objective. This algorithm was implemented for the localization of radioactive sources, by guiding the UAV to utilize a rollout-based path planning strategy to select the next best position to move. This is algorithm borrows concepts and observations from previous works by Hoffmann et al. [12] and Tian et al. [13]

While Hoffmann et al. focused on bearing only RF emitter localization, the approach for radiation source localization needed some adaptations as it had to account for the challenges in localizing radiation sources. Where the original work focussed on optimizing bearing measurements and movement costs, our approach needed to account for radiation characteristics such as inverse square law decay, intensity estimation, and radiation attenuations. The intensity estimation is relevant here as the algorithm does not have prior knowledge of the source intensity. Hoffmann et al. uses a Q-value $Q(s, a)$ which is basically the expected cumulative reward of taking action a in state s and following an optimal policy thereafter. In their work, the Q-value is estimated by considering the cost immediate cost combining movement and measurement time, and the expected future cost again considering the movement and measurement time.

To adapt the Q-value computation for radiation-specific physics and simulated radiation measurements, the following modifications were done:

$$Q(s_t, a_t) = \sum_{t=0}^T \gamma^t V(s_t, m_t) \quad (8)$$

where $V(s_t, m_t)$ is the value function combining belief values and simulated radiation measurements:

$$V(s_t, m_t) = \frac{\sum_{i,j \in M} b(i, j)}{1 + d(s_t, x_{i,j})} + \frac{1}{1 + 0.2 * d_{max}} \quad (9)$$

where m_t is the simulated measurements at time t , $\gamma = 0.9$ is the discount factor, $b(i, j)$ is the belief at grid cell (i, j) , $d(s_t, x_{i,j})$ is the distance from state s_t to grid cell (i, j) , d_{max} is distance to maximum belief point, M is the valid map region.

Belief updates follow a rigorous process that begins with likelihood computation using the measurement model. These updates are performed in log space to handle the wide dynamic range of probability values encountered. When new measurements arrive, the system computes the likelihood of these measurements given each possible source location and intensity. This computation takes into account the sensor noise model and the physics of radiation propagation. The algorithm is designed to be of two stages, exploration and exploitation phase. The exploration phase is designed to follow a shamrock pattern initially and terminate when the source is detected with a reasonable confidence. The exploitation phase has a termination criteria that ensures reliable source localization. These criteria include achieving a high belief confidence (exceeding 0.95) in the source location, demonstrating stability in the maximum likelihood estimate over multiple measurements, meeting a minimum number of measurements threshold to ensure reliability, and confirming proximity to the estimated source location. [NEEDED?]

The rollout algorithm perform the trajectory planning by generating multiple candidate trajectories at each point by calculating angles based on the current belief state and drone position, then simulates multiple potential paths with adaptive step sizes. Each path is evaluated using a belief-based value function that considers both the immediate information gain and the long-term exploration potential. This approach allows the system to make decisions that balance the immediate need for information with the strategic value of maintaining efficient coverage of the search area. [ADD INFORMATION GAIN EQUATION????] [TO REVIEW] The greedy approach is used as a fallback mechanism if the rollout method fails to generate a valid trajectory.

Figure 2 represents the run of the rollout algorithm-based radiation source localization algorithm for a source located at (30, 50). The blue line represents the path taken by the drone, the blue dotted line represents the trajectory rollouts from the current position, and the yellow star at the end of each algorithm chosen trajectory endpoint position. This run localized the source at a distance of 7 meters from the actual source location.

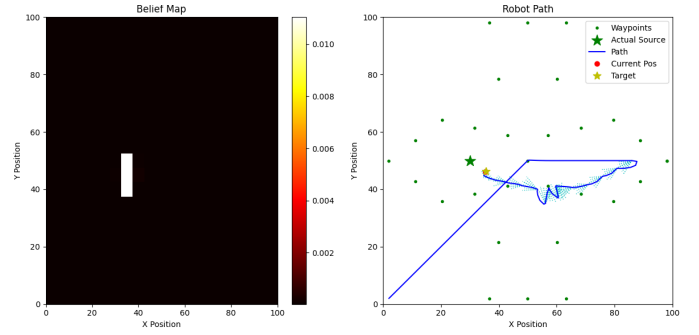


Fig. 2: Rollout algorithm-based radiation source localization for source location (30, 50)

3) Inverse Square Law Optimization: The implemented localization method is based on the inverse square law for radiation intensity and serves as a deterministic baseline for comparing different radiation source search strategies. The algorithm employs a concentric search pattern that scales automatically according to the search area consisting of three concentric circles positioned at 25%, 50%, and 75% of the search area radius. This pattern begins from the center and expands outward, with the number of measurement points per circle scaling proportionally with the map size to maintain consistent coverage across different search areas. The choice of using concentric circles for data collection is based on tests showing that other patterns, such as shamrock, were less effective in terms of coverage and accuracy. This approach balances the need to collect data covering most of the search space with the constraint of avoiding excessive time consumption.

This method localizes the radiation source by beginning with deciding the search pattern and number of measurement points on this pattern scale based on the map size. At each measurement point, the radiation sensors collect radiation count and compare it against the expected measurements based on the inverse square law, which dictates that the intensity decreases with the square distance from the source. These measurements are then weighted, assigning higher importance to stronger readings, as they indicate the proximity to the source. The collected data is fed into the Nelder-Mead optimization algorithm, which iteratively adjusts the source position and intensity by minimizing the log-space error between the expected and measured radiation counts. To handle uncertainty in source intensity, the algorithm employs multiple optimization attempts with varying initial intensity estimates (100x, 1000x, and 10000x the maximum measured count rate) to avoid local minima. The final estimated source position is accompanied by a confidence circle, whose radius is proportional to the standard deviation of the measurements, indicating the uncertainty of the estimate.

Many real-world problems involve objective functions that are not differentiable due to noise, discontinuities, and radiation measurements are one of those problems. Gradient-based optimization methods struggle with such functions as they need smooth, differentiable functions to work. The Nelder-Mead algorithm is effective for non-smooth or discontinuous functions because it does not rely on gradients. Instead, it

evaluates the function at simplex vertices and adjusts the simplex shape iteratively. This makes it a robust choice for optimization problems where derivative information is unavailable or unreliable. Compared to global optimization methods like evolutionary algorithms, the Nelder-Mead approach offers lower computational costs for problems with fewer variables or limited evaluations, which aligns well with the requirements of engineering optimization. [14]

This approach is an effective baseline for comparison due to its deterministic nature, foundation in established physics principles, reproducible behavior, and comprehensive evaluation metrics. The predefined path strategy and inverse square law optimization provide a methodical reference point against which more sophisticated, informed search strategies can be evaluated.

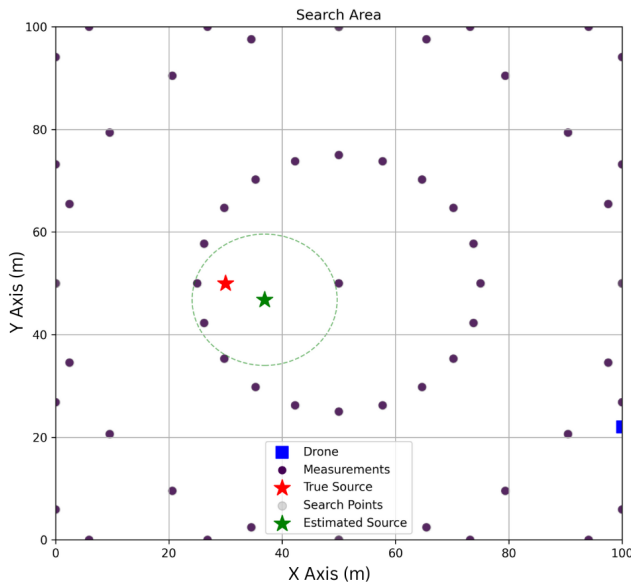


Fig. 3: inverse square optimization-based radiation source localization for source location (30, 50)

Figure 3 represents the run of the inverse square law optimization-based radiation source localization algorithm for a source located at (30, 50). the green ellipse represents the confidence circle around estimated source position (green star). The red star represents the true source position, black points represent the visited measurement points, and the blue square represents the estimated source position. This run localized the source at a distance of 7.6 meters from the actual source location.

V. EVALUATION

The efficiency of the simulation framework and the implemented algorithms were evaluated using a set of experiments. The experiments were performed under a set of different radiation source position chosen randomly. The experiments were conducted in a 100x100 meter area and the drone is assumed to take off From the origin of the area for each test.

For each source position, each algorithm is tested 10 times. Each test is considered successful if the drone is able to locate a source, and the main parameters of the test will be saved into a JSON file for further analysis. The tests under the same source position are evaluated together. The evaluation metrics considered for each run are, prediction error, time taken to detect the source, time taken to converge the source position, distance travelled by the drone to localize the source, number of iterations, and time taken for computation. The prediction error is calculated as the Euclidean distance between the predicted source position and the actual source position. Along with the performance metrics, the measurement history, trajectory, and computation times from each iteration are also saved for further analysis. The results of the experiments are presented in the following sections.

The tests were conducted in a 100x100 meter area with the source positioned in the following randomly selected locations: (25,80), (30,50), (50,50), (70,45), (80,70).

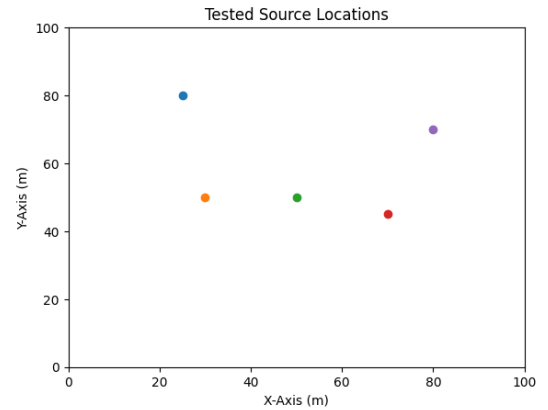


Fig. 4: Source Locations used for Evaluation

TABLE I: Algorithm Performance Comparison Across Source Locations

Location	Entropy		Rollout		Inverse Square	
	Error (m)	Time (s)	Error (m)	Time (s)	Error (m)	Time (s)
(25, 80)	6.01	125.57	10.24	177.54	14.38	415.17
(30, 50)	7.06	43.56	7.00	108.65	6.62	470.52
(50, 50)	7.21	46.12	6.89	112.45	6.83	468.24
(70, 45)	8.17	45.67	6.66	115.42	7.71	415.70
(80, 70)	8.08	61.85	6.33	91.62	4.53	431.01

Note: Error represents prediction error in meters, Time represents convergence time in seconds

A. Analysis

As shown in the table I, the source location (30, 50) presents an interesting case for detailed analysis. This location was chosen for further investigation because it represents the best performance across all algorithms.

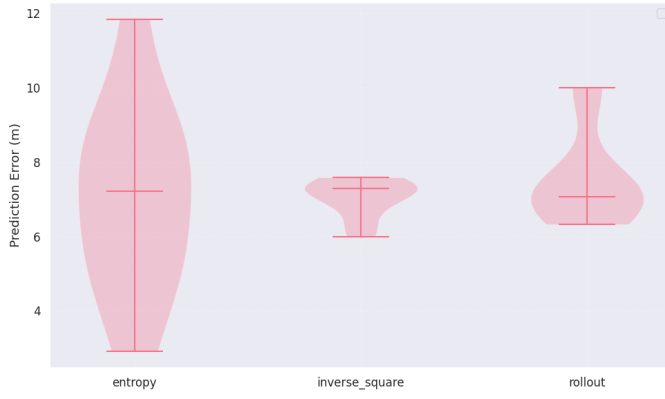


Fig. 5: Violin plot of prediction error for source location (30, 50)

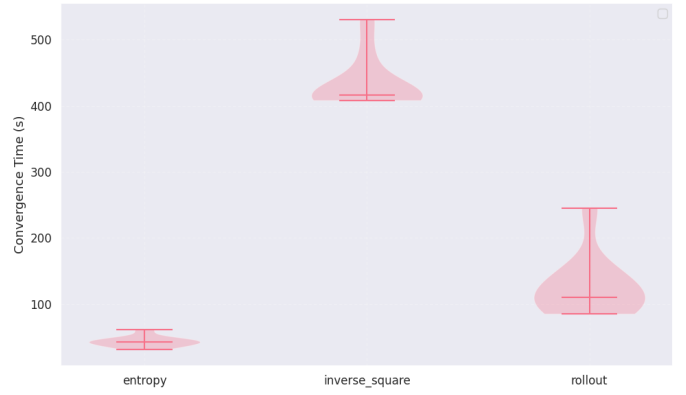


Fig. 6: Violin plot of convergence time for source location (30, 50)

1) *Prediction Error*: The violin plot in Figure 5 illustrates the distribution of prediction errors across all implemented algorithms. The entropy algorithm demonstrates the widest range of performance, achieving both the highest (approximately 12 m) and lowest (approximately 2 m) prediction errors among all algorithms. The plot's characteristic narrowing at higher error values indicates that these larger errors occur infrequently, suggesting they are outliers rather than typical performance. Notably, the algorithm's ability to achieve the lowest prediction errors in some runs demonstrates its potential for high accuracy under favorable conditions.

The median error of the entropy algorithm is approximately 7m, which aligns with findings from previous studies by Ristic et al. [9] and [4]. In contrast, the inverse square algorithm exhibits a distinctly narrow spread, with errors consistently falling between 6 to 7.5 meters. This concentrated distribution around 7 meters reflects the algorithm's consistency across runs, which can be attributed to its systematic measurement collection from fixed points within the search area relative to the source location.

The rollout algorithm shows prediction errors ranging from 6 to 10 meters, with most errors clustering around 7 meters, similar to the other algorithms. The algorithm's minimum error of 7 meters occurs frequently, which can be attributed to the chosen grid size (5 m) naturally leading to localization errors in the range of 5 to 7 meters. This pattern suggests that the grid resolution plays a significant role in determining the algorithm's precision limits.

2) *Convergence Time*: The plot for figure 6 illustrates the distribution of convergence time across different algorithms revealing some interesting insights. The entropy-based method exhibits the fastest convergence times, with most runs successfully localizing within 45 seconds. The tight horizontal spread of the entropy algorithm's convergence times indicates that the algorithm consistently converges within a narrow time frame making it the fastest algorithm to find the source once the source's presence is detected. The inverse square method shows significantly longer convergence times, centered around 420-470 seconds. This longer convergence time is likely due to its systematic approach of measuring at predefined points,

requiring more comprehensive coverage of the search area.

The rollout algorithm presents an intermediate performance profile, with convergence times typically between 90-180 seconds. Its distribution shows more variance than the entropy method but considerably less than the inverse square approach. The broader base of its distribution indicates more variability in convergence times, possibly due to the algorithm's adaptive decision-making process that can take different paths depending on the measurements received.

B. Statistical Analysis

Since radiation counts follow a Poisson Distribution [4], non-parametric tests were selected for algorithm comparison. Specifically, the Wilcoxon Signed-Rank Test was employed to provide detailed comparisons of paired samples, accounting for both the direction and magnitude of differences. All tests were conducted at a significance level of 0.05.

1) *Prediction Error Analysis*: The statistical analysis of prediction errors using the Wilcoxon Signed-Rank Test revealed no significant differences between the algorithms, with detailed results presented in Table II. These findings suggest that all three algorithms achieve comparable levels of accuracy in source localization.

a) *Detailed Analysis*: The pairwise comparisons revealed several key insights:

- The comparison between Entropy and Inverse Square methods showed perfect balance (5 wins each) with a negligible effect size ($r = 0.0483$), strongly suggesting equivalent performance.
- While Entropy showed a slight advantage over Rollout (6 vs 4 samples), the small effect size ($r = 0.2095$) and non-significant p-value (0.5566) indicate that this difference is not practically meaningful.
- Similarly, Inverse Square's slight advantage over Rollout (6 vs 4 samples) yielded a small effect size ($r = 0.2740$) and non-significant p-value (0.4316), suggesting comparable performance levels.

TABLE II: Wilcoxon Signed-Rank Test Results for Prediction Error

Method Comparison	Statistical Results	Performance Summary
Entropy vs. Inverse Square	Sample size: 10 Statistic (W): 26.0000 p-value: 0.9219 Effect size (r): 0.0483	Equal performance distribution with 5 samples favoring each algorithm. Negligible effect size indicates no practical difference in accuracy.
Entropy vs. Rollout	Sample size: 10 Statistic (W): 21.0000 p-value: 0.5566 Effect size (r): 0.2095	Slight advantage for Entropy (6 vs 4 samples) but small effect size and non-significant p-value indicate comparable performance.
Inverse Square vs. Rollout	Sample size: 10 Statistic (W): 19.0000 p-value: 0.4316 Effect size (r): 0.2740	Minor advantage for Inverse Square (6 vs 4 samples) but small effect size suggests minimal practical difference.

b) Interpretation of Results: The consistency in non-significant results across all three comparisons, combined with small to negligible effect sizes, provides strong evidence that the algorithms achieve similar levels of prediction accuracy. This is particularly noteworthy given the balanced distribution of positive and negative ranks across comparisons, suggesting that any observed differences are likely due to random variation rather than systematic performance differences.

2) *Convergence Time Analysis:* The convergence time analysis utilizing the Wilcoxon Signed-Rank Test uncovered systematic performance differences among the algorithms.

a) Wilcoxon Signed-Rank Test Results: Table III presents the results of the Wilcoxon Signed-Rank Test conducted to compare the ‘convergence_time’ metric across the three algorithms: entropy, inverse_square, and rollout. Each pairwise comparison assessed whether there were statistically significant differences in convergence times between the algorithms.

b) Detailed Pairwise Analysis: The Wilcoxon Signed-Rank Test results indicate significant and substantial differences in convergence times among the three algorithms.

Entropy consistently achieved faster convergence times than the Inverse Square method across all 10 paired samples. This difference is statistically significant ($p = 0.0020$) with a large effect size ($r = 0.8864$), indicating that the observed performance improvement is both meaningful and practically significant.

Similarly, Entropy outperformed Rollout in all 10 paired samples, demonstrating a statistically significant difference ($p = 0.0020$) and a large effect size ($r = 0.8864$). This suggests that Entropy not only performs better but does so consistently across all instances tested.

In contrast, Rollout consistently achieved faster convergence times than Inverse Square in all 10 paired samples. The results are statistically significant ($p = 0.0020$) with a large effect size

TABLE III: Wilcoxon Signed-Rank Test Results for Convergence Time

Method Comparison	Statistical Results	Performance Summary
Entropy vs. Inverse Square	Sample size: 10 Statistic (W): 0.0000 p-value: 0.0020 Effect size (r): 0.8864	Complete advantage for Entropy (10 vs 0 samples) with large effect size demonstrating clear performance superiority in convergence speed.
Entropy vs. Rollout	Sample size: 10 Statistic (W): 0.0000 p-value: 0.0020 Effect size (r): 0.8864	Complete advantage for Entropy (10 vs 0 samples) with large effect size showing consistent superior performance in convergence time.
Inverse Square vs. Rollout	Sample size: 10 Statistic (W): 0.0000 p-value: 0.0020 Effect size (r): 0.8864	Complete advantage for Rollout (10 vs 0 samples) with large effect size indicating consistently faster convergence than Inverse Square.

($r = 0.8864$), underscoring the superior performance of Rollout over Inverse Square.

c) Rank Summary: For each pairwise comparison, the rank summaries are as follows:

In the comparison between Entropy and Inverse Square, all 10 paired samples favored Entropy, resulting in 0 positive ranks (Inverse Square performed better) and 10 negative ranks (Entropy performed better), with no ties.

Similarly, in the Entropy versus Rollout comparison, all 10 paired samples favored Entropy, leading to 0 positive ranks (Rollout performed better) and 10 negative ranks (Entropy performed better), with no ties.

Lastly, in the Inverse Square versus Rollout comparison, all 10 paired samples favored Rollout, resulting in 10 positive ranks (Rollout performed better) and 0 negative ranks (Inverse Square performed better), with no ties.

d) Interpretation of Results: The Wilcoxon Signed-Rank Test outcomes reveal clear and consistent performance hierarchies among the three algorithms. Entropy consistently outperforms both Inverse Square and Rollout, achieving faster convergence times across all paired samples. Additionally, Rollout outperforms Inverse Square in all paired samples. These findings suggest that the Entropy algorithm is the most efficient in terms of convergence time, followed by Rollout, with Inverse Square being the least efficient among the three.

3) *Summary and Recommendations:* 1. ****Statistical Significance:**** The Wilcoxon Signed-Rank Test results indicate whether the observed differences between algorithms are likely real or due to chance, based on the comparison of p-values with the significance level ($\alpha = 0.05$).

2. ****Practical Significance:**** Effect sizes (r) provide insight into the magnitude of the differences. Large effect sizes, as observed in all comparisons, suggest that the differences are

not only statistically significant but also practically meaningful for real-world applications.

3. ****Decision Making:**** Both statistical and practical significances should be considered when making decisions. Large effect sizes combined with statistical significance offer the strongest evidence for meaningful differences. These findings should be aligned with specific use cases and requirements to inform the selection of the most appropriate algorithm.

VI. CONCLUSIONS

The time includes the movement time as well, a drone with a better controller might be achieve better performance in the the time taken.

A. *Summary*

B. *Contributions*

C. *Future Work*

REFERENCES

- [1] "Doe explains...radioactivity," <https://www.energy.gov/science/doe-explainsradioactivity>, accessed: 2024-06-25.
- [2] Y. Huang, J. Benesty, G. W. Elko, and R. M. Mersereau, "Real-time passive source localization: A practical linear-correction least-squares approach," *IEEE transactions on Speech and Audio Processing*, vol. 9, no. 8, pp. 943–956, 2001.
- [3] 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.
- [4] 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.
- [5] K. Ziocck and W. Goldstein, "The lost source, varying backgrounds and why bigger may not be better," in *AIP Conference Proceedings*, vol. 632, no. 1. American Institute of Physics, 2002, pp. 60–70.
- [6] G. Fragkos, K. Karafasoulis, A. Kyriakis, and C. Potiriadis, "Localization of radioactive source using a network of small form factor czr sensors," 03 2020.
- [7] A. H.-W. Liu, "Simulation and implementation of distributed sensor network for radiation detection," Ph.D. dissertation, California Institute of Technology, 2010.
- [8] X. Hu, J. Wang, J. Huo, Y. Zhou, Y. Guo, and L. Hu, "Autonomous exploration for radioactive sources localization based on radiation field reconstruction," *Nuclear Engineering and Technology*, vol. 56, no. 4, pp. 1153–1164, 2024.
- [9] B. Ristic, A. H. Gunatilaka, and M. Rutten, "An information gain driven search for a radioactive point source," *2007 10th International Conference on Information Fusion*, pp. 1–8, 2007. [Online]. Available: <https://api.semanticscholar.org/CorpusID:11418497>
- [10] M. Ling, J. Huo, G. V. Moiseev, L. Hu, and Y. Xiao, "Multi-robot collaborative radioactive source search based on particle fusion and adaptive step size," *Annals of Nuclear Energy*, vol. 173, p. 109104, 2022.
- [11] D. P. Bertsekas, "Rollout algorithms for discrete optimization: A survey," in *Handbook of combinatorial optimization*. Springer New York, 2013, vol. 5, pp. 2989–3013.
- [12] F. Hoffmann, H. Schily, A. Charlish, M. Ritchie, and H. Griffiths, "A rollout based path planner for emitter localization," in *2019 22th International Conference on Information Fusion (FUSION)*, 2019, pp. 1–8.
- [13] X. Tian, Y. Bar-Shalom, and K. R. Pattipati, "Multi-step look-ahead policy for autonomous cooperative surveillance by uavs in hostile environments," in *2008 47th IEEE Conference on Decision and Control*, 2008, pp. 2438–2443.
- [14] M. A. Luersen, R. Le Riche, and F. Guyon, "A constrained, globalized, and bounded nelder–mead method for engineering optimization," *Structural and Multidisciplinary Optimization*, vol. 27, pp. 43–54, 2004.

ACKNOWLEDGMENT

Write your acknowledgments here.

STATEMENT OF ORIGINALITY

[If AI assistants have not been used, use this sentence] I, the undersigned below, declare that this work has not previously been submitted to this or any other university and that it is, unless otherwise stated, entirely my own work.

[If an AI assistant has been used, use this sentence] I, the undersigned below, declare that this work has not previously been submitted to this or any other university and that it is, unless otherwise stated, entirely my own work. The report was, in part, written with the help of the AI assistant [AI assistant name] as described in the appendix. I am aware that content generated by AI systems is no substitute for careful scientific work, which is why all AI-generated content has been critically reviewed by me, and I take full responsibility for it.

Date

Signature

APPENDIX

Please limit the main part of the report to 20 pages (not including the references, the statement of originality, and the appendix).

In your appendix, you can add any additional details about your work, such as:

- extra results that do not necessarily belong in Sec. V
- more detailed justifications of certain algorithm design decisions
- algorithm proofs

Additionally, in the case of using AI assistants, describe in detail what content was generated using an AI assistant. In particular, name the AI assistant(s) that you used and how they were used (e.g. which prompts were used, and for which parts of the project).