To the Graduate Council:
I am submitting herewith a dissertation written by David M. Raji entitled "Multi-Objective Radiological Analysis in Real Environments." I have examined the final paper copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Nuclear Engineering.

Jason Hayward, Major Professor

We have read this dissertation
and recommend its acceptance:

Reynold Cooper

Sandra Bogetic

Hairong Qi

Accepted for the Council:

Dixie Thompson

Vice Provost and Dean of the Graduate School

To the Graduate Council:
I am submitting herewith a dissertation written by David M. Raji entitled "Multi-Objective Radiological Analysis in Real Environments." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Nuclear Engineering.

Jason Hayward, Major Professor

We have read this dissertation
and recommend its acceptance:

Reynold Cooper

Sandra Bogetic

Hairong Qi

Accepted for the Council:

Dixie Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Multi-Objective Radiological Analysis in Real Environments

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

David M. Raji

داود متی راجی

May 2024

© by David M. Raji, 2024
All Rights Reserved.

برای قزه

Acknowledgements

First, it should be acknowledged that this work took place in large part upon the historical lands of the Miccosukee, Tsalaguwetiyi, and S'atsoyaha peoples. In my time residing here, I have tried to learn from and exist within the native environment as much as possible; I would very much encourage others to do the same.

To my committee – Profs. Jason Hayward, Sandra Bogetic, Hairong Qi, and Dr. Reynold Cooper – I very much appreciate the time and effort extended to me during this process. I would especially like to thank Jason and Ren as well as Dr. Tenzing Joshi for their guidance, direction, and advice over the past four years.

I would also like to give flowers to the various colleagues that have helped me out over the years, among them Micah, Emil, Ingo, Carl, and many more. And of course, Jaewon: I am beyond glad we finally got to meet last year, and I wish you all the best wherever you go.

To my union kin, I cannot say enough how much I appreciate your passion and energy for changing this place for the better. I sincerely hope I helped to make some small contribution towards positive action in my short time working with you all. I would also like to extend my deepest thanks to everyone and everything that has helped me make a home and find meaning here in Knoxville; without the people and places that matter most to me, I am unsure I would have made it to the finish line.

To my family: I much appreciate the support throughout this long process. Thank you m0m especially for always being happy for me to visit when I have been able.

To my close friends who have kept me sane, I cannot state my appreciation enough. Juwon, Samira, and Vidya: I do not know how I would have made it through the early COVID years without you all. Those weekly calls were really invaluable. And to Casey and

Eve, I am beyond grateful that I met you both even if in less than ideal circumstances. Your friendship is invaluable to me, and I cannot wait to be done with this so we can meet up again for more hijinks. I also can't forget Mica and Lakshmi; it means a lot to have our friendship even after such long times apart.

Lastly but certainly not least of all, I would like to thank Catherine; I appreciate so so much your patience, support, and love over these long years. I'm very happy to be almost done.

یه دیواره، یه دیواره، یه دیواره

Abstract

Designing systems to solve problems arising in real-world radiological scenarios is a highly challenging task due to the contextual complexities that arise. Among these are emergency response, environmental exploration, and radiological threat detection. An approach to handling problems for these applications with explicitly multi-objective formulations is advanced. This is brought into focus with investigation of a number of case studies in both natural and urban environments. These include node placement in and path planning through radioactivity-contaminated areas, radiation detection sensor network measurement update sensitivity, control schemes for multi-robot radioactive exploration in unknown environments, and adversarial analysis for an urban nuclear threat detection network. For each of these case studies, the multi-objective modeling process is undertaken along with the development of problem-specific algorithmic methods. Results take the form of outputs from multi-objective optimization algorithms; attention is given to the characteristics of both Pareto front estimations computed and samples constructed using the obtained optimal parameterizations. In the urban environment case, a novel multi-objective competitive optimization framework as well as static methods are used to produce network configurations; a quantification study is undertaken to compare performance these configurations against the adversarial behavior model.

Table of Contents

1 Introduction **1**

1.1 Consequences of Nuclear Technology 1

1.1.1 Environmental Contamination 1

1.1.2 Threats in Urban Areas 3

1.2 Primary Applied Analysis Methods 4

1.2.1 Optimal Sensor Network Modeling 4

1.2.2 Path Planning 6

1.3 Optimization and Multi-Objective Modeling 7

1.3.1 Optimization 7

1.3.2 Multi-Objectivity 9

1.3.3 Multi-Objective Optimization 10

1.4 Motivation 12

2 Review of Prior Work **14**

2.1 Radiation Detection and Mapping 14

2.1.1 Environmental Radioactivity 14

2.1.2 Urban Radiation Detection 16

2.2 Multi-Objective Modeling & Optimization 18

2.3 Sensor Networks 20

2.4 Path Planning 21

2.5 Adversarial Analysis 24

2.6 Specialized Models for Optimization 26

2.6.1 Dynamic Optimization 26

2.6.2 Multi-Population Optimization 27

3 Statement of Original Contributions **30**

4 Natural Environments **32**

4.1 Modeling Framework 32

4.1.1 Data Sources 33

4.1.2 Reconstruction 34

4.2 Node Placement 40

4.2.1 Evolutionary Optimization 41

4.2.2 Results 41

4.3 Path Planning 44

4.3.1 Heuristics 44

4.3.2 Algorithm Formulation 45

4.3.3 Objectives 46

4.3.4 Parameters 52

4.3.5 Optimization 52

4.3.6 Results 52

4.4 Measurement Update Sensitivity 59

4.4.1 Drift Detection 62

4.4.2 Objectives 63

4.4.3 Optimization & Results 64

4.5 Multi-Robot Exploration 66

4.5.1 Multi-Objective Methodology & Optimization 66

4.5.2 Results 70

5 Urban Environments **74**

5.1 Detector Network Modeling Considerations 74

5.1.1 Adversarial Threats in Urban Areas 74

5.1.2 Assumptions & Approximations 75

5.2 Data Sources 76

5.2.1 Road Network . . . 76
5.2.2 Route Data . . . 77
5.3 Detector Placement . . . 79
5.4 Adversarial Analysis Methodology . . . 80
5.4.1 Iterative Adversarial Approach . . . 80
5.4.2 Competitive Co-Evolutionary Optimization . . . 81
5.5 Results . . . 89
5.5.1 Network Configuration Evaluation . . . 89
5.5.2 General Detector Positioning Analysis . . . 95

6 Conclusions **110**
6.1 Natural Environments . . . 111
6.1.1 Node Placement . . . 111
6.1.2 Path Planning . . . 112
6.1.3 Measurement Update Sensitivity . . . 113
6.1.4 Multi-Robot Exploration . . . 114
6.2 Urban Environments . . . 114

Appendix **134**

A Reconstruction Method Comparisons **135**

B Pseudocode for FPA* and its Heuristics **138**

C Optimal Path Samples from FPA* **141**
C.1 Output Run *B-103* . . . 142
C.2 Output Run *B-109* . . . 143

D Adversarial Targets **144**

E Pseudocode for IIA* **146**

F Quantitative Adversarial Analysis: Complete Figures **147**

F.1 50 Detectors . . . 148

F.1.1 Route Dataset: 0% Targeting . . . 148

F.1.2 Route Dataset: 5% Targeting . . . 150

F.1.3 Route Dataset: 50% Targeting . . . 151

F.1.4 Route Dataset: 100% Targeting . . . 153

F.2 100 Detectors . . . 155

F.2.1 Route Dataset: 0% Targeting . . . 155

F.2.2 Route Dataset: 5% Targeting . . . 157

F.2.3 Route Dataset: 50% Targeting . . . 158

F.2.4 Route Dataset: 100% Targeting . . . 160

F.3 200 Detectors . . . 162

F.3.1 Route Dataset: 0% Targeting . . . 162

F.3.2 Route Dataset: 5% Targeting . . . 164

F.3.3 Route Dataset: 50% Targeting . . . 166

F.3.4 Route Dataset: 100% Targeting . . . 168

G Sensor Scoring: Complete Figures **170**

G.1 50 Detectors . . . 171

G.1.1 Route Dataset: 0% Targeting . . . 171

G.1.2 Route Dataset: 5% Targeting . . . 174

G.1.3 Route Dataset: 50% Targeting . . . 176

G.1.4 Route Dataset: 100% Targeting . . . 179

G.2 100 Detectors . . . 182

G.2.1 Route Dataset: 0% Targeting . . . 182

G.2.2 Route Dataset: 5% Targeting . . . 185

G.2.3 Route Dataset: 50% Targeting . . . 188

G.2.4 Route Dataset: 100% Targeting . . . 191

G.3 200 Detectors . . . 194

G.3.1 Route Dataset: 0% Targeting . . . 194

G.3.2 Route Dataset: 5% Targeting 197
G.3.3 Route Dataset: 50% Targeting 200
G.3.4 Route Dataset: 100% Targeting 203

Vita **206**

List of Tables

4.1 Example of incomparable node choices. Each of the three nodes here are non-dominated with respect to the other two. It is noted that these values are chosen purely for purpose of example and are not necessarily realistic in the context of the problem. 47

5.1 Corners of bounding box of the region of Chicago used for investigation in this Section. 78

5.2 Characteristics of route datasets used in urban radiation detection sensor network. In this context, the "size" column corresponds to the number of distinct origin-destination pairs contained in the dataset. 82

5.3 Network configuration characteristics key. 90

D.1 List of target locations modeled as destinations of adversaries. Priority metric is determined by frequency of occurrence of destination in sources and is used to inform the chances of selection in path generation (lower priority numbers get selected more often). 145

List of Figures

1.1 Example of Pareto front structure from non-dominated solutions for the welded beam optimization problem. This is a reference problem which seeks to optimize for both cost and displacement with parameters of weld thickness and beam dimensions. 11

4.1 Diagram of software modeling framework for radiation detection sensor network. 36
4.2 Topography of Region A (top) and Region B (bottom). 37
4.3 Sample cropped simulated depositions and their corresponding radiological intensity distributions. 38
4.4 Reconstructions of radiological intensity distributions using interpolation, GPR (a), and p2pGAN (b) methods. Scenario is with 25 nodes in Region A (see Section 4.1.1). 39
4.5 Example maps of node placement on real-world topography. 47
4.6 Evolutionary MO optimization outputs in the form of the historical solution distribution in objective space. Each plot depicts a 2D projection of the problem's 4D objective space. Optimal solutions found and stored in elitist archive are displayed in red, while all solutions evaluated throughout optimization are displayed in blue. 48
4.7 Evolutionary MO optimization outputs in the form of the historical solution distribution in objective space. Each plot depicts a 2D projection of the problem's 4D objective space. Optimal solutions found and stored in elitist archive are displayed in red, while all solutions evaluated throughout optimization are displayed in blue. 49

4.8 Node placement algorithm parameter shading of approximated Pareto front projection shown in Fig. 4.6b. 50
4.9 Node placement algorithm parameter shading of approximated Pareto front projection shown in Fig. 4.7a. 50
4.10 Reconstruction method parameter shading of approximated Pareto front projection shown in Fig. 4.7a. 51
4.11 Quantity of nodes placed parameter shading of approximated Pareto front projection shown in Fig. 4.7a. 51
4.12 Plot of radiological intensity distribution and topography along with path source and destination optimized for in run ***B-3***. 54
4.13 Distribution of final population of the NSGA-II MOEA for run ***B-3*** visualized as a surface in the three-dimensional objective space. Points shown correspond to the sample paths displayed in Fig. 4.14. 55
4.14 Selection of path samples using optimized parameterizations for FPA* from run ***B-3***. Path shown in blue, source and destination of path displayed in fuchsia and lime green, respectively. 57
4.15 Complete population histories from run of MOEA shaded by distance cost weights (top, (a) and (b)), terrain cost weights (middle, (c) and (d)), and exposure cost weights (bottom, (e) and (f)). Pearson correlations between the weight parameter and the relevant objective value are provided for each plot. This output is from run ***B-3***. 58
4.16 Plot of Pearson correlation between hyper-descriptor values and optimized parameter and objective result values. Best objective values are taken from the final population and parameter values used are those corresponding to that objective. 60
4.17 Convergence behavior of FPA* parameter optimization using NSGA-II. Quantified using objective space hypervolume gain. Average and standard deviation reported for optimization on the 25 distinct combinations of environmental hyper-descriptors. 61

4.18 Distribution of FPA* complexity FoM for 900 optimal path parameterizations; empirical CDF also shown. 61
4.19 Sample of plume intensity over time compared with the current measurement; total area between curves used as one objective in multi-objective formulation. Updates (inflections) determined by drift detection algorithm. 65
4.20 Sample results in objective space of MOEA in applied to measurement update sensitivity problem. Generational histories of population (blue) and final archive (red) of solutions are shown here. 65
4.21 Optimization results presented in Fig. 4.20 shaded according to input parameter values. 67
4.22 Physical robots deployed for data collection experiment in Nuclear Environmental Site Area 2 at TA-49 on August 24, 2022. 68
4.23 Approximated Pareto front found via MOEA optimization for MR-MOPSO robot mapping with GPR reconstruction every 5 steps in a 100-by-100 discretized grid. Two-dimensional projections from three-dimensional objective space shown here. 71
4.24 Simulated MR-MOPSO robot mapping results with GPR reconstruction every 5 steps in a 100-by-100 discretized grid, with three robot paths shown. 73

5.1 Region of Chicago selected for investigation. Edges of graph model are formed by roads, which are colored here according to their access grade. 78
5.2 Sample IIA* result for 150 sensors placed using Strategy A (as described in Section 5.3). Google Maps recommendation shown in navy. Overlapping routes offset for visibility. 82
5.3 Diagram of COCO-EVO optimization framework. Directional arrows represent evaluations of one population (hollow circle) on another (arrow point). . . . 85
5.4 Letter-value plots of IIA* minimal path cost distributions across detector network configurations. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 5% targeted routes. 92

5.5 Comparison of summarized IIA* results across network configurations using route dataset of 5% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations. 93
5.6 Detector position scoring with naive and adversarial pathing evaluation using test route dataset of 5% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown for Loop area of downtown Chicago. 97
5.7 Residual of detector position scoring KDEs shown in Fig. 5.6. Positive values indicate higher position scoring in naive path evaluation, whereas negative values are indicative of higher scoring with adversarial path evaluation. Detail shown for Loop area of downtown Chicago. 98
5.8 Residual between naive and adversarial evaluation of detector position scoring KDEs using test route dataset of 5% targeted routes. Positive values indicate higher position scoring in naive path evaluation, whereas negative values are indicative of higher scoring with adversarial path evaluation. Detail shown for Loop area of downtown Chicago. 99
5.9 Detector position scoring with naive and adversarial pathing evaluation using test route dataset of 50% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown for Loop area of downtown Chicago. 100
5.10 Residual of detector position scoring KDEs shown in Fig. 5.9. Positive values indicate higher position scoring in naive path evaluation, whereas negative values are indicative of higher scoring with adversarial path evaluation. Detail shown for Loop area of downtown Chicago. 101

5.11 Residual between naive and adversarial evaluation of detector position scoring KDEs using test route dataset of 50% targeted routes. Positive values indicate higher position scoring in naive path evaluation, whereas negative values are indicative of higher scoring with adversarial path evaluation. Detail shown for Loop area of downtown Chicago. 102
5.12 Detector position scoring with naive and adversarial pathing evaluation using test route dataset of 50% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown for museum campus area of Chicago on the shore of Lake Michigan. 105
5.13 Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using naive pathing evaluation and a test route dataset of 5% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown. 106
5.14 Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 5% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown. 107
5.15 Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 5% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown for the Loop area of downtown Chicago. 108
5.16 Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 50% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown for the Loop area of downtown Chicago. 109

A.1 Comparison of performance distributions for various reconstruction methods. Evaluations are conducted for networks comprised of 25 nodes each, placed in topography using farthest-first traversal in two dimensions. 137

C.1 Selection of path samples using optimized parameterizations for FPA* from run ***B-103***. Path shown in blue, source and destination of path displayed in fuchsia and lime green, respectively. 142

C.2 Selection of path samples using optimized parameterizations for FPA* from run ***B-109***. Path shown in blue, source and destination of path displayed in fuchsia and lime green, respectively. 143

F.1 Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 50 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 0% targeted routes. 148

F.2 Comparison of summarized IIA* best path cost results across 50-detector network configurations using route dataset of 0% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations. 149

F.3 Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 50 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 50% targeted routes. 151

F.4 Comparison of summarized IIA* best path cost results across 50-detector network configurations using route dataset of 50% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations. 152

F.5 Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 50 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 100% targeted routes. 153
F.6 Comparison of summarized IIA* best path cost results across 50-detector network configurations using route dataset of 100% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations. 154
F.7 Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 100 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 0% targeted routes. 155
F.8 Comparison of summarized IIA* best path cost results across 100-detector network configurations using route dataset of 0% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations. 156
F.9 Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 100 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 50% targeted routes. 158
F.10 Comparison of summarized IIA* best path cost results across 100-detector network configurations using route dataset of 50% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations. 159

F.11 Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 100 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 100% targeted routes. 160
F.12 Comparison of summarized IIA* best path cost results across 100-detector network configurations using route dataset of 100% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations. 161
F.13 Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 200 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 0% targeted routes. 162
F.14 Comparison of summarized IIA* best path cost results across 200-detector network configurations using route dataset of 0% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations. 163
F.15 Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 200 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 5% targeted routes. 164
F.16 Comparison of summarized IIA* best path cost results across 200-detector network configurations using route dataset of 5% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations. 165

F.17 Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 200 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 50% targeted routes. 166
F.18 Comparison of summarized IIA* best path cost results across 200-detector network configurations using route dataset of 50% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations. 167
F.19 Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 200 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 100% targeted routes. 168
F.20 Comparison of summarized IIA* best path cost results across 200-detector network configurations using route dataset of 100% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations. 169

G.1 Detector position scoring with naive and adversarial pathing evaluation using route dataset of 0% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown. 171
G.2 Residual of detector position scoring KDE for evaluation using route dataset of 0% targeted routes with sensor networks of 50 allotted detectors each. . . 172
G.3 Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 0% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown. 173

G.4 Detector position scoring with naive and adversarial pathing evaluation using route dataset of 5% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown. 174
G.5 Residual of detector position scoring KDE for evaluation using route dataset of 5% targeted routes with sensor networks of 50 allotted detectors each. . . 175
G.6 Detector position scoring with naive and adversarial pathing evaluation using route dataset of 50% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown. 176
G.7 Residual of detector position scoring KDE for evaluation using route dataset of 50% targeted routes with sensor networks of 50 allotted detectors each. . . 177
G.8 Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 50% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown. 178
G.9 Detector position scoring with naive and adversarial pathing evaluation using route dataset of 100% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown. 179
G.10 Residual of detector position scoring KDE for evaluation using route dataset of 100% targeted routes with sensor networks of 50 allotted detectors each. . 180
G.11 Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 100% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown. 181
G.12 Detector position scoring with naive and adversarial pathing evaluation using route dataset of 0% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown. 182
G.13 Residual of detector position scoring KDE for evaluation using route dataset of 0% targeted routes with sensor networks of 100 allotted detectors each. . . 183

G.14 Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 0% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown. 184
G.15 Detector position scoring with naive and adversarial pathing evaluation using route dataset of 5% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown. 185
G.16 Residual of detector position scoring KDE for evaluation using route dataset of 5% targeted routes with sensor networks of 100 allotted detectors each. . . 186
G.17 Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 5% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown. 187
G.18 Detector position scoring with naive and adversarial pathing evaluation using route dataset of 50% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown. 188
G.19 Residual of detector position scoring KDE for evaluation using route dataset of 50% targeted routes with sensor networks of 100 allotted detectors each. . 189
G.20 Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 50% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown. 190
G.21 Detector position scoring with naive and adversarial pathing evaluation using route dataset of 100% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown. 191
G.22 Residual of detector position scoring KDE for evaluation using route dataset of 100% targeted routes with sensor networks of 100 allotted detectors each. 192

G.23 Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 100% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown. 193
G.24 Detector position scoring with naive and adversarial pathing evaluation using route dataset of 0% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown. 194
G.25 Residual of detector position scoring KDE for evaluation using route dataset of 0% targeted routes with sensor networks of 200 allotted detectors each. . . 195
G.26 Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 0% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown. 196
G.27 Detector position scoring with naive and adversarial pathing evaluation using route dataset of 5% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown. 197
G.28 Residual of detector position scoring KDE for evaluation using route dataset of 5% targeted routes with sensor networks of 200 allotted detectors each. . . 198
G.29 Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 5% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown. 199
G.30 Detector position scoring with naive and adversarial pathing evaluation using route dataset of 50% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown. 200
G.31 Residual of detector position scoring KDE for evaluation using route dataset of 50% targeted routes with sensor networks of 200 allotted detectors each. . 201

G.32 Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 50% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown. 202
G.33 Detector position scoring with naive and adversarial pathing evaluation using route dataset of 100% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown. 203
G.34 Residual of detector position scoring KDE for evaluation using route dataset of 100% targeted routes with sensor networks of 200 allotted detectors each. 204
G.35 Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 100% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown. 205

Nomenclature

MO	Multi-Objective
MOO	Multi-Objective Optimization
EA	Evolutionary Algorithm
MOEA	Multi-Objective Evolutionary Algorithm
PSO	Particle Swarm Optimization
DM	Decision Maker
GPR	Gaussian Process Regression
KDE	Kernel Density Estimate
OD Pair	Origin-Destination Pair

Chapter 1

Introduction

1.1 Consequences of Nuclear Technology

The emergence of nuclear science and technology has been and continues to be a source of incredible upheaval. The development and proliferation of both nuclear weapons and energy have profoundly affected the fabric of global society and security. The fallout of this transformation has been significant; the world is now faced with the seemingly endless task of assessment of nuclear threats, may they be to the natural environment, human development, or humans themselves. Once these threats are understood, effective mitigation strategies to confront them in their specific context must then be designed and employed.

Yet, this last part of the process is far from trivial. Practical solutions to radiological problems are often constrained by a number of factors, including environmental settings, resource or financial limitations, and finite time. When developing systems or methods with aims to address these problems, it can consequently often be unclear how to balance those competing demands with the relevant radiological risk mitigation task.

1.1.1 Environmental Contamination

Past nuclear activity and accidents have seen radioactive contamination dispersed over wide geographical areas [1, 2, 3, 4]. Questions as to what consequences such contamination will have on the natural environment have been open for many decades [5]. Those observed

to this point have been very far reaching; they range from effects on consumer behavior towards seafood products [6] to changes in the rates and tendencies of infectious diseases in wildlife populations [7]. A major area of concern and study has been the reactions of plants and wildlife to these sources of contamination. Due to their nature, many types of longer lived plants such as trees can accumulate significant quantities of radioactivity as a result of nutrient or other particulate uptake over the course of years.

It is clearly an objective of environmental science to understand more about such processes, in part in inform means of mitigation. Yet the infrequency of such catastrophic contamination has paradoxically made this very difficult. In one study assessing pine forests contaminated by the Chernobyl accident, concentrations of ^{137}Cs and ^{90}Sr were measured in soils and cones [8]. Elevated rates of aberrant cells were found, indicating high possibilities of cytogenetic damage to the trees. Even so, changes to the reproductive process were not observed. The authors argue that there is still much unknown about the response of flora to radioactive stimulus, including under what conditions and to what degree radioresistivity consistently develops. In another case, discoveries of radioactive contamination of some rice crops in the Fukushima prefecture forced large scale investigations into radioactive cesium uptake in paddy fields as well as the creation of a rice bag radiation measurement protocol [9].

More directly, radioactive contamination incidents have major effects on humans as a result of necessitated evacuation in addition to the obvious risks of radiation exposure. This has been studied in the context of the Fukushima disaster; as detailed by Hasegawa et al. in [10], the evacuation directly led to a number of negative health outcomes for the population. These included excess deaths among the elderly and those in hospital and nursing facilities, psychological distress, and increases in rates of chronic health problems [11]. Furthermore, coordinating an effective evacuation procedure has proven challenging. Ohba et al. in [12] assess the body surface contamination levels found on evacuees in different groups, finding that differences in time of day of evacuation potentially resulted in much increased doses to individuals. The importance of planning and preparedness for future nuclear accidents has been emphasized in response [11]. This process includes (but is not limited to) evacuation planning, consideration of radioactive plume movement, maintaining

measurement and monitoring capabilities, assuring information sharing capability, and planning for contingencies involving compounding disasters.

1.1.2 Threats in Urban Areas

Radioactivity-based threats, such as radiological dispersal devices (RDDs; also called "dirty bombs"), present grave risks to densely populated areas [13]. RDDs are generally conceptualized as weapons designed with the goal of spreading radioactive material over a large area, with the assumed purpose of rendering that area uninhabitable. There is debate as to what exact effects the detonation of an RDD would have; physical and psychological harm to health and economic damage are among the most frequently raised [13].

Despite the very low likelihood of such a risk materializing, the potential for harm is extremely high should an RDD attack ever occur. Therefore, and as in the previous section, preparedness is again a driving force. This process has been contributed to in various ways; for example, an aerosolization facility has been used to investigate what radius of radioactive hazard might result from an RDD [14]. In that study, the work was motivated by a desire to establish best practices for first responders to minimize exposure. Similar experiments have also been conducted at scale using radioactive tracers (^{140}La) as surrogate dispersal material [15].

Other efforts have targeted entirely different areas of preparation, such as information transmission during and after the crisis that would follow an attack with an RDD [16]. Much work has also been dedicated to systems designed with the idea of preventing these attacks before they happen; the work described in Chapter 5 of this text focuses specifically on this goal. In this context, these security systems often take the form of radiation detection equipment, sometimes in combination with other technologies.

Radiation portal monitors (RPMs) represent one relatively simple incarnation of this; these are systems with large radiation detector volumes that are designed to detect radioactive materials or materials contaminated with radioactive particulate being transitted, usually onboard a vehicle. RPMs can be effective in providing high sensitivity at specific points of ingress or egress, such as at a border crossing. Yet their size and immobility limits

the usability of RPMs in urban areas, where space is often at a premium and choke-points of travel are much less common [17].

The risk of proliferation possible as an outcome of the smuggling of nuclear material through cities is also of great concern [18].

More recently, networks of radiation detectors co-located with other contextual sensors have been investigated as viable tools for counter-terrorism objectives [19, 20, 18]. The ability to distribute sensitivity across an environment is very attractive for this application, as there is little surety as to what in an urban area would be the most likely target. The sensor network approach also offers continual monitoring, an advantage over many handheld or vehicle-deployed systems.

Regardless of approach, a very high level of societal responsibility for collective safety in this context has tended to be entrusted to technology. It is consequently paramount that the systems employed are sufficiently trustworthy, and this must be ensured by adequate evaluation for robustness. Part of this has to include consideration of possible adversarial action; this has come into greater focus with widespread understanding of the threat of cyberattacks to critical infrastructure, most especially following the Aurora vulnerability demonstration in 2007 [21, 22]. In this frame, it can be imagined that a radiation detection sensor network designed with some expectation of "normal" behavior patterns may maintain adequate sensitivity towards the vast majority of traffic in a city while failing to detect adversaries intentionally avoiding nodes of the network. It is an open question to what extent this can be planned for and preemptively acted against.

1.2 Primary Applied Analysis Methods

1.2.1 Optimal Sensor Network Modeling

Research into sensor networks has accelerated a great deal within the past 20 years as options for embedded hardware have rapidly grown more powerful, lightweight, and versatile [23]. The enabling technologies associated with this boom have opened the door for myriad explorations into application potentials, as alluded to in Section 1.1.2.

A few separate systems utilizing sensor networks are modeled in this text. The first is described in Section 4.1; in this application, the network is comprised of widely-distributed nodes in a region of natural environment. The second application – detailed in Chapter 5 – models a network of radiation detectors deployed in a city, with a focus on road-level sensitivity to passing vehicles. It is also possible to conceptualize the multi-robot system used for exploration and mapping (covered in Section 4.5) as a type of sensor network due to its use of swarm intelligence to guide motion planning.

Regardless of specific case, it is intuitive that networks comprised of greater numbers of sensors will generally provide more overall sensitivity. Conversely, when limited numbers of sensors must be utilized to construct a network, it is worthwhile to ensure maximal sensitivity is obtained from each of them. This is often a function of physical sensor positioning in an environment and can be described broadly with the formulation

$$\max_{n \in N} \sum_{i=1}^{n} S(x_i) \tag{1.1}$$

for a sensitivity function $S(x_i)$ positioning a sensor node at a location x_i and using some selected number of sensor nodes n of possible quantities N. The specific sensitivity function will be unique to each application and implementation.

Ensuring that maximum marginal sensitivity is obtained from sensors is crucial to a number of practical applications. For instance, a sensor network system could be used to monitor a wide geospatial area for contamination across an extended period of time. Increased numbers of detectors will result in higher deployment costs for and topological complexity of the system. Costs associated with utilizing large quantities of sensors will also likely be found in urban network deployments, with further complications introduced by restrictions on possible sensor node positions.

Another challenge in selecting best positions for static sensors involves the potential for environmental dynamics. As mentioned in the previous section, radiological threat detection networks in population centers must contend with the potential for intelligent adversarial strategies. How exactly to position the detectors of such a network to counteract this

possibility is so far unclear. This is discussed in further detail in Section 2.5 and investigated in Section 5.4.

1.2.2 Path Planning

Finding a path to a destination is a universal problem. A task as simple as a trip to the market from home requires a route of some design. In many trivial cases, the best path will be that which minimizes the total distance of travel. To focus specifically on this matter, some mathematical formalization is warranted. Given an undirected graph G, an origin node n_0, and a destination node n', the shortest path problem can be described with

$$\min_{p \in P} \sum_{i=0}^{N-1} w_{e_{i,\,i+1}} \quad \text{for path} \quad p\,(n_0,\, n_1,\, \ldots\, n_N) \tag{1.2}$$

for an edge $e_{i,\,i+1}$ between nodes n_i and n_{i+1}, the weight of said edge $w_{e_{i,\,i+1}}$, and with $n' = n_N$, where N is the length of path p in the set of possible paths P. In other words, the objective is to minimize the sum of edge weights between the origin and destination nodes of the graph. This formulation holds for path planning problems that solely emphasize distance; this is not the case in every application, however.

Path planning is relevant to a number of applied radiological contexts. Following from the discussion in Section 1.1.1, it is clear there exists a need for navigation that avoids radioactive environmental contamination. This is relevant to a number of circumstances, such as evacuation of civilians from a contaminated disaster area and maintenance of safety for emergency response personnel at an accident site or for decontamination workers carrying out routine activities. In attempting to solve these problems, it is obvious that attention to more than the length of the path is required. For example, if the minimal length path from an evacuee's home to a temporary shelter would proceed directly through highly radioactive areas, it would not be worth consideration. In this scenario, it can be said that the objective of minimizing path length is in conflict with that of minimizing exposure. Discussion on dealing with objectives in conflict is continued in Section 1.3.2, and an approach to path planning in this context is presented in Section 4.3.

One type of path planning analysis of interest in the urban setting is that conducted to evaluate deployed radiation detection sensor networks. Adversarial path planning strategies are needed for this purpose in order to effectively model threats. It follows that these strategies also are necessarily motivated by more than minimizing distance; a successful path in this context will avoid detection by the sensor network in addition to being relatively short. Work in this area is detailed in Section 5.4 of this text.

1.3 Optimization and Multi-Objective Modeling

1.3.1 Optimization

Study in the relatively novel fields of metaheuristics and associated optimization methods has proven a boon in countless applications across a wide variety of fields. Such techniques are often able to be utilized when a problem suffers from a large parameter space that lacks obvious correlation to an objective value which is to be minimized or maximized. Optimization can be mathematically described with

$$\min_{x \in X} \ f(x) \tag{1.3}$$

for solution x in set of possible solutions X evaluated by objective function $f(x)$. The objective function in any optimization problem serves the purpose of transfer between the input parameter space and the output objective space; the characteristics of this function are necessarily unique to the particular problem. Recalling the general formulations of the node placement problem in eq. 1.1 and the path planning problem in eq. 1.2, it is briefly noted that these can easily be mathematically posed as optimization problems themselves.

Metaheuristics

Before discussing optimization algorithms themselves, it must first be understood that these algorithms are most often constructed using some combination of search metaheuristics [24]. While the term "metaheuristic" is at times used interchangeably with "optimization algorithm", the former is more accurately described as a specific tactic used to guide the

search process through a problem's parameter space. Several metaheuristics are typically used in combination to form a complete optimization algorithm. Some examples of metaheuristics are briefly described below:

- Neighborhood Search: exploration of parameter space immediately surrounding an already evaluated solution
- Hill Climbing: use of small parameter space steps that are accepted if the new solution's objective value shows improvement
- Adaptive Memory Programming: utilization of previously evaluated solutions to plan search strategy in the future

Optimization Algorithms

Evolutionary algorithms (EAs) represent a large and diverse group of optimization methods, all sharing the characteristic of drawing on evolutionary-inspired operators to guide the solution search process [25, 26]. The general premise involves the treatment of solution candidates as individuals in a population that are guided towards maximal fitness via evolutionary operators; these are usually selection, mutation, and recombination (reproduction). In keeping with the above discourse on metaheuristics, it is noted that EAs utilize all of those specifically listed as well as others in their operation. An important characteristic of EAs is their use of a population-based search metaheuristic, while allows multiple concurrent parameter space searches [24]. Loosely, EAs function as follows: a population of solution candidates with random parameterizations are initialized. At each step, they are evaluated according to some objective function; based on the fitness results, individuals are either selected out of the population or made to "reproduce" with other individuals to create "offspring" that share a mix of the two parents' parameters. Parameter mutations are also routinely applied to offspring to encourage diversity in the parameter exploration.

Particle swarm optimization (PSO) is another common archetype of optimization algorithm. In addition to several already mentioned metaheuristics, it makes use of directional search to navigate the parameter space [27, 28, 24]. The generic version

of PSO models individual solution candidates as "particles" in parameter space. Each individual particle calculates local and global velocity vectors that influence the change in parameterizations; local vectors point towards a particle's best known historical location of itself, while global vectors are directed towards the position with the best objective result found by any particle. These best positions are determined by objective function evaluations, which take place after each incremental movement.

In addition to the two categories of algorithms described above, ant colony optimization has emerged as a distinctive method with its introduction of the search space mapping metaheuristic [29]. In this method, a parameter space overlay is built and updated during the optimization procedure; this mapping guides search processes towards areas of the parameter space evaluated to provide better objective values. Pheromone trails left by ants in nature were the inspiration for this development.

Although the above three detailed constitute perhaps the most popular optimization algorithms, many other valid options exist. In more recent years, a wide variety of new optimization algorithms drawing inspiration from natural processes have been developed and presented; some of the most popular among these have been artificial bee colony optimization [30] and cuckoo search [31].

1.3.2 Multi-Objectivity

Somewhat more uncommon but nonetheless important is the sub-problem wherein multiple competing objectives are considered. Multi-objective (MO) describes the state of a problem wherein there is more than one measurable quantity for which it is sought to obtain the best value. Optimization in the MO scheme can be described mathematically with

$$\min_{x \in X} \ (f_1(x), \ f_2(x), \ \ldots, \ f_k(x)) \tag{1.4}$$

for solution x in the set of possible solutions X evaluated by objective function $f_i(x)$ with the quantity of objectives $k \geq 2$.

At the core of MO modeling are the decisions that must be made between inherently competing objectives. Strategies to manage tradeoffs between them can largely be grouped

into either *a priori* or *a posteriori* categories. In the former, the multiple objectives are collapsed into some singular meta-objective; this is usually accomplished by utilizing some means of scalarizing the multi-dimensional objective space. However, it should be noted that while scalarization allows for a relatively simple approach to deal with the complexity of MO problems, it also results in a number of issues. One of these is that it is very difficult to accomplish without resorting to semi- or wholly-arbitrary weights to assign to the objectives. Beyond introducing subjectivity into the process, *a priori* methods also remove information at the results stage of the optimization, meaning a decision maker (DM) is unable to perceive the full scope of optimal options. With *a posteriori* methods, all information is preserved, thereby avoiding these issues. Emphasis is instead placed on finding solutions that embody the best possible tradeoffs between all objectives.

In such cases, the fact that the simultaneous optimization of all objectives may not be possible leads us to search for a set of "Pareto-optimal" solutions rather than a singular one [32, 33]. These Pareto-optimal solutions are those that achieve the optimal trade-offs between competing objectives. To be precise, if a solution $\boldsymbol{\alpha}$ has objective values that are all equal to those of a solution $\boldsymbol{\beta}$ save for one which is superior, it may be said that solution $\boldsymbol{\alpha}$ dominates solution $\boldsymbol{\beta}$. A solution is part of the Pareto-optimal set if there exist no other solutions that would dominate it and is therefore "non-dominated" [34]. Collectively, this set is commonly referred to as the Pareto front, frontier, or boundary, and is comprised of mutually non-dominated solutions. An example of this for the welded beam reference problem [35] is visualized in Fig. 1.1; here, the structure of a Pareto front can be observed distinct from sub-optimal points in the two-dimensional objective space.

1.3.3 Multi-Objective Optimization

Following the separate descriptions of optimization and MO modeling above, attention is now turned to how to combine these ideas. At their core, MO optimization (MOO) methods are aimed at finding the complete Pareto front (or some approximation of it) of non-dominated solutions for an MO problem. In practice, this is usually accomplished with the use of MOO algorithms, the most popular of which are modifications or extensions of the optimization algorithms described previously in Section 1.3.1.

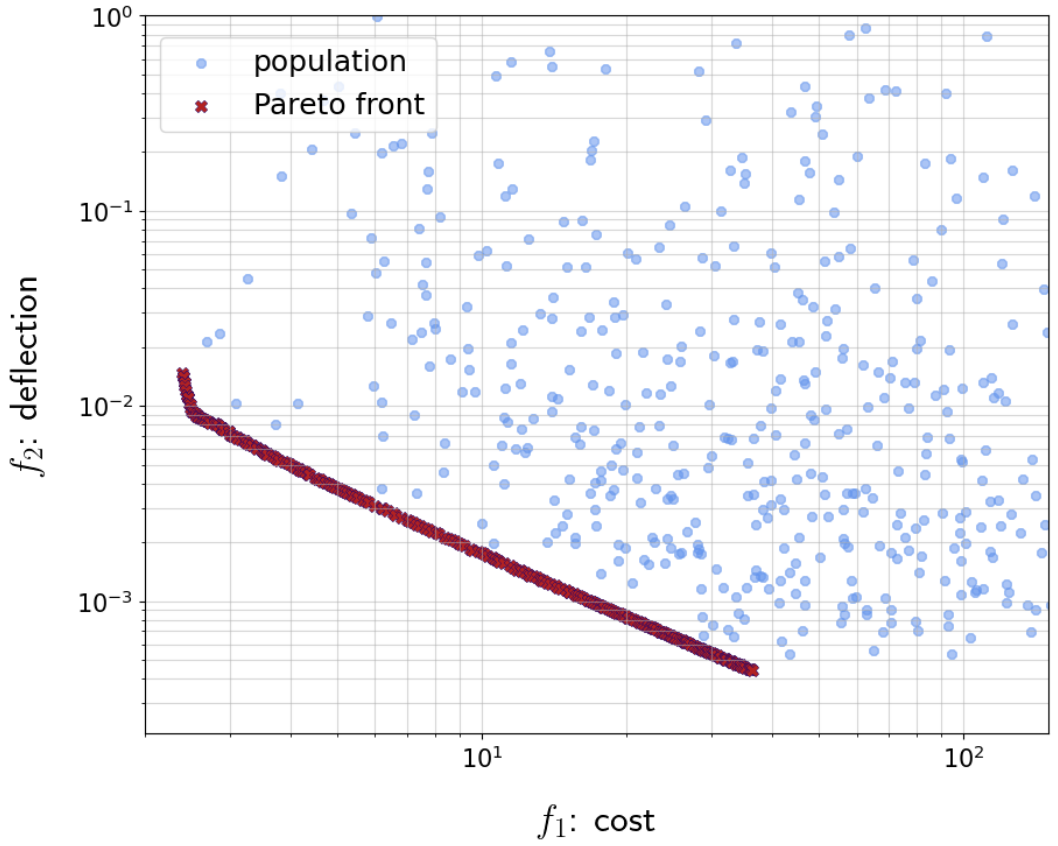


Figure 1.1: Example of Pareto front structure from non-dominated solutions for the welded beam optimization problem. This is a reference problem which seeks to optimize for both cost and displacement with parameters of weld thickness and beam dimensions. Pareto front points and evaluation conducted using pymoo [36].

Common archetypes of MOO methods include multi-objective evolutionary algorithms (MOEAs, [32]) and multi-objective particle swarm optimization (MOPSO, [37, 38]). Within these groups, there exists a significant degree of diversity in implementation. But a key shared characteristic is the use of Pareto optimality principles to assess solution candidate fitness and therefore guide the search in the parameter space. Rather than solutions being compared on a scalar objective value basis to determine rankings, they are instead assessed for domination with non-dominated candidates receiving top rankings. For example, the popular non-dominated sorting genetic algorithm II (NSGA-II, [33]) ranks solutions in the population in reverse order according to the number of times that they are dominated. Many algorithms also utilize some form of "elitism", which describes some mechanism to save the best solutions that have been found so far during the optimization process. The rationale for this is to allow the algorithm to more aggressively explore the search space without risking losing information on optimal solutions that have already been discovered.

More specifics on implementation details and important prior contributions are reviewed in Section 2.2.

1.4 Motivation

As described above, constraints on deploying technologies in the real world are ever-present. They may take the form of limited time, finite financial and material resources, hazardous conditions, and/or other restricting forces. Faced with these complications, accuracy and attention to complexity in modeling is imperative. It is very often insufficient to seek solutions that are constructed based solely on consideration of radiological goals, such as avoiding dangerous exposures to radioactivity or detecting smuggled nuclear materials. These solutions will tend towards unrealism, and implementation may be simply impossible.

To address these challenges, it is argued that explicit consideration of applied context throughout the problem-solving process is indispensable. Doing this not only allows for a more realistic view of the problems themselves, it also preserves information as to potential tradeoffs in solution implementations for a DM. Following from this logic, this text presents an investigation of applied radiological problems in real environments with the view that

they are best understood and modeled as fundamentally multi-objective in nature. This is carried out through a process described below:

1. Identification of complex problems in applied context; this text selects a number of case studies for this purpose:

    - Optimizing sensor network node positions and measurement update schema for monitoring radioactive contamination
    - Path planning through contaminated environments
    - Collaboratively exploring and mapping areas for radioactivity with multiple robots
    - Evaluating urban radiation detector network performance against intelligent adversaries

2. Definition of all objectives along with relevant parameters and any constraints that apply to them; each applied problem is fully characterized in the MO paradigm.

3. Utilization of optimization algorithms; in order to find a range of optimal solutions, MO optimization algorithms such as EAs are used to efficiently search large parameter spaces.

4. Review of results; Pareto-optimal solutions identified for each problem are presented and trends in optimal tradeoffs between objective values are discussed.

Chapter 2

Review of Prior Work

2.1 Radiation Detection and Mapping

2.1.1 Environmental Radioactivity

Scientific interest in measuring radioactive contamination in the environment can be traced back to the era immediately following the development of nuclear weaponry. As Dunning notes in his book on radioactivity in the Pacific Ocean [39], periodic surveys of the degree of contamination were initiated soon after the Castle Bravo test on a number of the Marshall Islands. As nuclear technology progressed, other potential hazards were created; the work of Eichholz et al. [40] studies the potential for radionuclides originating from high-level waste repositories to be transported via suspended particulate carriers.

Major nuclear accidents have been a driver of study in this area as well unsurprisingly; they often unfortunately serve as valuable testing grounds for model validation. Albergel et al. write in [1] on the modeling of the radioactive plume over Europe with comparisons of predictions to real measurements. The aftermath of the Fukushima disaster in 2011 saw much research as well, both aimed at assessing the health and environmental impacts as in the work of Koo et al. in [3] and again towards modeling the movement of contamination, described by Nagao et al. in [41].

The Fukushima accident also came at a time of significant progress in radiation detection and mapping technology, and as a result, much investigation was undertaken into the

possibility of imaging contaminated environments with newly designed systems. Among those efforts are the work of Galloway et al. [42] and that of Towler et al. [43]. Many efforts in this vein involved robotic systems, with a significant subset of these being unmanned aerial vehicles (UAVs) [44, 45, 46, 47]. In [45], Martin et al. detail their work using a low-flying UAV mounted with radiation detection equipment along with a laser rangefinder to map an area of radioactive contamination surrounding the 2011 nuclear disaster site in Fukushima, Japan. Connor et al. in [46] review the use of these aerial systems for radioactivity mapping broadly and speculate on future designs.

However, relatively few efforts have been conducted into radiation measurement over wide areas with sensor networks, with those that have being mostly targeted towards anomalous source detection rather than large area mapping. One example in that category is the work of Rao et al. in [48], which examines the capability of a small network of NaI scintillators to detect a source passing through the area. In the context of a different application, Nicholson et al. in [49] describe an approach utilizing a sensor network for reactor operation monitoring.

A few investigations blend the mobile robot approach with that of networked sensing; in these circumstances, the robot agents are often fitted with additional sensors such as GPS in order to tie together location information with radiation measurements. One of the works in this area is that of Cortez et al. in [50], wherein the authors examine a system with multiple ground-based robots equipped with radiation detectors for mapping an area of potentially contaminated environment. The control algorithm the authors analyze here makes use of information gradients to guide the search process. Han et al. in [44] describe a multi-UAV approach to mapping radioactive contamination, citing advantages in time necessary to map an area in urgent situations as well as improved robustness through redundancy. Also in this hybrid category is the work of Shin et al. in [51]; here, the methodology models multiple ground-based robots affixed with radiation detectors and assumes ability to intercommunicate location information between them. The authors evaluate the system's ability to characterize an unexplored area by use of trajectory directed by Voronoi partitioning of difference maps between Gaussian process regression prior and posterior estimates of spatial uncertainty. For reference, it is noted that Voronoi cells represent a division of some space given a fixed number of centers allowed within that space;

this division is calculated in such a way as to balance as best as possible a quantity that usually varies throughout the space. An algorithm to compute Voronoi cells is given by Lloyd in [52]; this method informs some of the node placement schemes evaluated in Section 4.2.

In more recent years, increasingly complex designs for radiation mapping systems have been developed and fielded for experimentation. A notable example in this vein is the PRISM imager, described by Hellfeld et al. in [53]. As explained by the authors, this system employs a novel spherical coded aperture with an active mask detector array; this design allows for handheld omnidirectional 3D imaging using both the coded aperture and Compton imaging modalities. The Localization and Mapping Platform (LAMP), described by Pavlovsky et al. in [47], provides a flexible solution for implementing 3D real-time radioactivity mapping either as a handheld or UAV-mounted system. In both of these above-described works, the fusion of radiological and physical scene data (termed "scene-data fusion" by the authors) relies heavily on the use of light detection and ranging (LiDAR) sensors to produce voxelizations of surroundings upon which radioactivity can be reconstructed. For both of these as well as other human-portable or UAV-mounted designs, a completely different set of system constraints comes into play than those for sensor networks; these include system weight, battery life, and ease of use. Consequently, design problems in this context are of interest but outside the scope of the investigation of this text.

2.1.2 Urban Radiation Detection

Broadly, previous efforts studying radiological detection in urban environments can be grouped into two categories: methods premised on mobile systems and those based on sensor networks.

In the first grouping, there are such works as those of the Multi-agency Urban Search Experiment (MUSE) collaboration [54, 55, 56], which has sought to leverage the Radiological Multi-sensor Analysis Platform (RadMAP) described by Bandstra et al. in [57] to investigate correlations between gamma-ray and contextual datastreams in urban environments. The RadMAP system is a large vehicle equipped with NaI scintillation and HPGe semiconductor radiation detectors along with cameras, LiDAR, and other sensors. Nicholson et al. in [54] present the development of a test bed for evaluation of urban radiological detection systems.

Their modeling is based on a realistic 3D computer representation constructed of the Fort Indiantown Gap Combined Arms Collective Training Facility in Fort Indiantown Gap, PA, USA (FTIG), along with radiation data collected at this same site. FTIG was also utilized by Bandstra et al. in [55], wherein the RadMAP system was fielded with the objective of quantifying differences in detector response between distinct materials by combining video data with radiation measurements. In [56], Bandstra et al. describe a similar procedure of experimentation and analysis for a deployment of the RadMAP system in a portion of Oakland, CA, USA.

In a similar vein are the efforts of Curtis et al. in [58]; the authors here describe the evaluation of another vehicle-based system combining multiple radiation detectors with contextual sensor data for radioactive source localization.

Citing the importance of detecting illicit radiological materials being transported into or through urban areas, the work of Ghawaly et al. in [59] presents a set of radiation data from simulations of detectors moving through city-scale environments. Details of the methodology of the data generation process are given by Nicholson et al. in [60]. The stated purpose of these data is to aid development of detection and isotope identification algorithms that are effective in these complex contexts. As many of these works fall into the realms of data collection, analysis, or modeling, they are somewhat less focused on consideration of constraints on long-term or scenario-specific system deployments.

The latter aforementioned category of urban detection strategies includes a wide range of approaches. Relatively early investigations in the networked vein include that of Nemzek et al. in [19] and Brennan et al. in [17]. In this first work, improvements in signal-to-noise ratio (SNR) for detection of a ^{137}Cs source are found in comparing the use of a network of scintillators to a single detector. That second work analyzes a sensor network along a roadway with the motivation of improving monitoring coverage over portal monitor-based systems against nuclear threats to urban areas such as RDDs. They argue that the networked approach provides a more flexible and extensible solution as compared to the visibility and constraints of portal monitors.

King et al. in [20] cite a similar counter-terrorism motivation in their development of a simulation methodology to evaluate design choices of a radiation-detecting sensor network.

Their proposed network incorporates both fixed sensor nodes as well as nodes equipped onboard mobile platforms already moving throughout a city, including public transit and public works vehicles. The design and evaluation of a network of all mobile sensors is described by Hite et al. in [61]; focus here is placed on quantifying the source localization capability within environments containing mixed and complex materials.

A different tact is taken by Osthus et al. within the sensor network detection paradigm in [62]; here, the authors emphasize the value of constraining attempts to reconstruct a source location to relevant surroundings such as roadways. It is argued that the addition of such domain awareness to the solution can be leveraged for more accurate results with unchanged radiation count information.

Cooper et al. in [18] describe a complex approach to urban radiological sensing that integrates systems termed Platforms and Algorithms for Networked Detection and Analysis (PANDA) and the Domain Aware Waggle Network (DAWN). In combination, these are described as the "PANDAWN" network. Generally, the authors envision deployments of network nodes at specifically selected locations throughout a city, with each node being equipped with an array of sensors. Similar to some of the mobile detection systems above, this networking also aims to fuse radiation data gathered from detectors with additional contextual datastreams, such as video and LiDAR. The authors enumerate the advantages of their approach as real-time radiological anomaly detection, dynamic adaptation, and data fusion across the network among others. It should be noted that the efforts described in Chapter 5 are contributions to the PANDA collaboration discussed in this passage.

In many of these static networked detection cases, there is generally limited exploration of design spaces as well as attention towards tradeoffs between design objectives that must be made in this process. In this text, modeling in this vein aims to explicitly include such considerations.

2.2 Multi-Objective Modeling & Optimization

Ideas of MO analysis were formally developed by Pareto in [34] in the context of describing economic efficiency. The central concept presented by the author – now known as "Pareto

optimality" or "Pareto efficiency" – concerns problems with two or more objectives that are in competition with one another. As discussed in Section 1.3.2, Pareto optimality refers to the state of a solution to such a problem wherein further improvements to the obtained value in one of the objectives cannot be made without negatively affecting the values of the others. This idea is core to the premise of MO optimization; algorithms for this purpose attempt to find the set of solutions that are Pareto optimal.

On the topic of algorithms for optimization, there exist myriad different methods. Evolutionary algorithms represent one group of methodological approaches to implementing optimization. Having first been generally theorized by Turing in [63] and later generally described by Fraser and Burnell in [25], they have gone on to become one of the most common archetypes of optimization algorithm. The basic premise is explained in Section 1.3.1.

The field of evolutionary MO optimization contains a rich history of approaches which is catalogued by Coello Coello in [32]. This review details most all of the important concepts and advancements that have shaped the current directions of the field's research. Also enumerated are a few of the most influential algorithms developed, including the Multi-Objective Genetic Algorithm (MOGA, [64, 65]), the Strength Pareto Evolutionary Algorithm (SPEA) by Zitzler and Thiele [66], and the Pareto Archived Evolution Strategy (PAES) by Knowles and Corne [67]. Also in this group and of major import to this work is the algorithm presented by Deb et al. in [33]; here, the authors present the non-dominated sorting genetic algorithm II (NSGA-II). Since its publication, this algorithm has become widely used and often functions as a quasi-standard approach to new applied MO problems. It blends a means to efficiently rank solution candidates found in the population, both by evaluating to what degree they are dominated and by calculating a "crowding" metric that is to be minimized in order to encourage completion of the Pareto front. NSGA-II is used for optimization of the MO problem described in Section 4.3.

Besides evolutionary methods, there exist other optimization procedures such as PSO. It has become widely used relatively quickly after having been first formulated by Kennedy and Eberhart in [27] along with Shi and Eberhart in [28]. The core of the method is discussed in Section 1.3.1. Parsopoulos and Vrahatis in [37] and Coello Coello et al. in [38] describe an MO extension to the PSO method; this is accomplished by utilizing elitist storage of

non-dominated solutions discovered along with the use of a fitness-sharing adaptive grid in order to promote exploration in the search and therefore prevent clumping of particles. The basis of MO-PSO is built upon for the work presented in Section 4.5.

2.3 Sensor Networks

Research in sensor networks can be traced back several decades to notable works such as that of Tenney and Sandell in [68] on the subject of their conceptual use for decentralized hypothesis testing. Since then, significant advances in sensor development have allowed these systems to be realized on large scales and for a wide range of applications. Min et al. provides in [23] a review of these technologies with a specific focus on low power implementations, a crucial element for any sensor network aimed at deployment in the natural environment or anywhere it cannot easily be routinely accessed. In [69], Antolín et al. build on this point by the development of a framework to reliably estimate lifetimes for sensors in such applications.

Another keystone idea is that of multi-hop information routing within a sensor network, which provides a solution to many practical cases where not all deployed nodes can easily communicate directly with an intermediate-level or user-facing system. Woo et al. in [70] analyze this problem in depth, covering the areas of inter-node link quality, node neighborhoods, and routing protocols. Ideas in these areas are combined with the previous topic of low-power node capability in [71] by Schiller et al.; they describe a methodology for energy-aware routing of data, in a sense applying a constraint to the multi-hop routing problem.

Placement of the physical nodes of a sensor network is a key concept that has received attention from literature as well. In this vein, several notable efforts have been made towards identifying methods of optimizing wireless sensor network (WSN) coverage, with many of them featuring PSO. The effort of Ab Aziz et al. in [72] focuses on WSN coverage as a sole objective; the authors use PSO to place each node and Voronoi partitioning to judge the fitness, which is somewhat like a reverse of the general approaches used in the work presented in Section 4.2 of this text. Ngatchou et al. in [73] describe the method of sequential

PSO (S-PSO) for placement of sonar sensors; this involves iteratively placing then moving individual nodes, which is a similar process to the aforementioned Lloyd's algorithm for finding Voronoi centroids. Other literature hones in on a specific application, an example being the work of Hu et al. in [74]. Here, the authors study the contextual case of optimizing topology of traffic-sensing networks with attention to message-receiving nodes' positioning. However, an MO approach is not taken here, and it is consequently more difficult to divine the relationship between the quantities of sensing nodes, quantities of message-receiving nodes, node transmission radii, and efficacy of the network.

The aforementioned work of Hu et al. with [74] focuses on a sensor network application for an explicitly urban setting, which is a common subgenre of applied research in the field. Rashid and Rehmani in [75] review the variety of approaches in this domain, with consideration towards the hazards of "environmental interference" that are more pertinent to urban contexts. Another survey is presented by Kafi et al. in [76]; here, focus is placed on WSNs specifically designed for traffic monitoring in urban areas. The authors identify a number of general motivators of research in this area as effective transmission of safety information, reduction of traffic congestion, and laying foundations for other smart city goals. Urban sensor networks focusing on meteorological monitoring are reviewed by Muller et al. in [77]. According to the authors, the coincidence of demand for climate and air quality monitoring in population centers and deployable sensor technology has resulted in many efforts in this subfield.

Finally, it is noted that sensor networks for radiation detection have been explored in previous works; a selection of these are reviewed in Section 2.1 earlier in this chapter. As noted there, these investigations mostly utilize relatively limited parameter space analyses and do not explicitly model competing objectives.

2.4 Path Planning

Much of the path planning work described in this text relies heavily on the foundation of the A* algorithm, first developed by Hart in [78], which itself essentially extends Dijkstra's path planning algorithm [79]. Both methods make use of an open and closed set, which

store graph nodes to consider visiting and those already visited, respectively. In the node selection step, both algorithms pick that node that minimizes a cost metric. While Dijkstra's algorithm uses only a cumulative cost that measures the total distance to reach a node, A* also integrates a heuristic that serves as an estimate of the remaining distance from the node to the destination. This addition makes the algorithm much faster and more efficient. However, in MO scenarios, A* can become restrictive as it demands availability of heuristics which may not always be trivial to calculate for all objectives. One approach taken to this particular problem in the work presented here is described in Section 4.3.1.

An important foundation in specifically MO path planning is the effort of Steward and White in [80]. There, the authors formulate a MO version of the A* algorithm that weaves in the selection of non-dominated nodes for the agent to move to at each step. Many works have extended that of Stewart and White's MO A*, including the efforts of Mandow et al. in [81] and [82]. The latter is of particular interest as it seeks to simultaneously find all non-dominated path options from a source to destination. However, approaches described in this text diverge from these as well as those of Mandow and Pérez de la Cruz in [83] and the work described by Pulido et al. in [84] in that no *a priori* preferential ordering of objectives is required in the methods explored here. While the goals of the NAMOA* algorithm outlined in the aforementioned literature is to make irrelevant this issue by obtaining all optimal paths, it is argued that this becomes both unnecessary and less tenable for application spaces with large, well-connected graphs and non-trivial competition between objectives such as is the case for the work addressed in Section 4.3.

The ideas developed in the above enumerated efforts have seen use in a number of applied works, such as those of Lavin in [85] and Jeddisaravi et al. in [86]. The latter explores the application of a slight variation of MO A* involving scalarization for the path planning of a Mars rover. The objectives given consideration are path length, elevation change, and "risk", which pertains to the robotic agent's likelihood of encountering obstacles such as rocks. Other previous works have incorporated elements of fuzzy analysis; among these are the work of He et al. in [87]. The authors of this paper focus primarily on designing MOEAs that overcome the breakdown of Pareto dominance criteria that is liable to occur in the many-objective case. Soltani and Fernando in [88] argue for the use of principles of fuzzy logic

in order to formulate a practically-applicable means to integrate their path safety objective. The authors remark that, while in an ideal world objectives such as safety level should always be maximized to their greatest possible extent, this is not a realistic way to approach the problem. Therefore, they design a means to quantify a preferred amount of safety achieved by the agent. While the authors' implementation and application space is quite different than that of the path planning studies presented in this text, the same principle of real-world practicality led towards the formulations used. Similar fuzzy ideas were utilized in the MO path planning approach presented in Section 4.3 to overcome incomparability stalemates in next-node selection.

A further review of literature reveals many works concerned with MO path planning as relating to obstacle avoidance for mobile robotics applications. While much of the applied context differs from that of the problems encountered in this work (see Sections 4.3, 4.5 and 5.4), the methods used still possess general merit and are worth consideration. The work of Ajeil et al. in [89] demonstrates an MO approach to path planning for an autonomous mobile robot using a hybrid algorithm of PSO and modified frequency bat (MFB), the objectives under consideration being path length and path smoothness. These are also the objectives of focus in [90], wherein Masehian and Sedighizadeh use PSO for global planning with the addition of the probabilistic roadmap algorithm to allow for local obstacle avoidance. It is noted that smoothness is of much greater concern when planning a path for robotic agents, since many designs are highly inefficient when performing sharp changes in motion vector. Differing slightly in this respect, Zhang et al. in [91] consider a different applied scenario in which elements of the environment pose some danger to the agent. Therefore, the authors consider a "risk degree" objective in addition to one for path length.

As described previously in Section 2.2, PSO models artificial particles representing solutions that change velocity over the course of the optimization period. The works [89], [90], and [91] all make use of PSO in some form, demonstrating its versatility. The former two applied efforts relate the particles directly to the modeled agent in some way, an approach suggested previously by works including that of Pugh and Martinoli in [92]. The latter work

by Zhang et al. [91] uses a more traditional implementation of PSO by encoding the path solutions into the artificial particles themselves.

A different tact is taken by Davoodi et al. in [93], which describes a methodology wholly using genetic operations to model discretized paths as artificial chromosomes; these subsequently are evolved towards an ideal solution. The authors here consider objectives of path length as well as "clearance", which they define as the closest approach of the agent to obstacles in the environment. Similarly, Ahmed and Deb in [94] model paths using four different genetic representation schemes in order to allow for ease of optimization using evolutionary methods. While this work excels in its application of finding suitable paths through densely-obstructed environments, the obstacles evaluated are binary in that any traversal renders the path "infeasible"; this differs greatly from the nature of the applications in the various contexts discussed in this text. In addition, the path-encoding approach would seem to preclude transferability of models between different applied cases, a significant limitation for practicability.

Some related works seek to map an unknown spatially-distributed radiological source term. One in this category is that of Mascarich et al. in [95], in which an estimate of radiation field intensity is updated iteratively. The measurement path is planned via waypoints, which are selected sequentially based on the intensity estimates in combination with spatial uncertainty. The application is somewhat similar to the context of the work described in Section 4.5, although an entirely different solution approach is used.

Yet another area of related inquiry is that focused towards applications wherein path planning through an adverse environment is considered. One effort in this area is that of Galyaev et al. in [96], for which path planning to minimize detection of an underwater vehicle via sonar is studied and an analytical solution developed. Further review of adversarially-cognizant analysis is presented in Section 2.5 later in this chapter.

2.5 Adversarial Analysis

Cox in [97] distinguishes between relatively simple adversarial analyses and more complex approaches that involve game theoretical considerations. The authors explain that those

methodologies in the latter category may seek to explicitly model interactions between opposing parties whereas the more simple methods may examine just one instance of defensive resource allocation. The utility of this distinction lies in the danger that oversimplification in risk analyses can pose, an example being that attackers can be capable of changing plans of attack after observing the action(s) of defenders. The ultimate approach taken with the adversarial analysis in this work (described in Section 5.4.2) aims to comprehensively model such dynamics in the context of the applied scenario.

Much recent applied research in adversarial analysis has been within the context of Machine learning (ML). ML models face "adversarial examples", or inputs that have been constructed in a way intended to induce an incorrect output. This phenomenon was formally described by Szegedy et al. in [98], and design methods aiming to mitigate it have been developed subsequently. Among these is the work of Ueseto et al. in [99], which studies adversarial risk in the context of deep learning models facing adversarial examples in real-world deployments. The authors argue that the applied domain is such that there could be a great diversity of adversarial threats, which is addressed in part by seeking to model the risk via a surrogate objective to be minimized. Another work in this area is that of Carlini et al. in [100], which focuses on attempting to establish a consistent means of evaluation of ML model defenses against adversarial examples. According to the authors, crucial to that process are steps including deciding on a threat model and usage of adaptive attack schemes in order to determine an upper robustness bound for the model being evaluated. The motivation behind that latter step is easily understood and applicable well beyond the ML model design field.

A number of recent investigations into adversarial analysis draw motivation from counterterrorism objectives. Among those is the work of Insua et al. in [101], wherein the authors seek to tie together disparate elements from the fields of risk analysis and game theory in a formalized manner. This is intended to improve on more traditional risk analysis methodologies by adding explicit means to model decision-making of the adversarial actor. Similarly, Rios and Insua in [102] study different paradigms of adversarial modeling within the counterterrorism context. Attention is drawn to the difference between scenarios within which defending and attacking actors take action simultaneously – and

therefore, without information – and those wherein the actions are sequential, allowing a response by the latter actor. The adversarial analysis described in Section 5.4 considers this nuance as well, and it is integrated via attention towards varying information levels as hyperparameters. Furthermore, it is noted that the work presented in this text expands on theoretical adversarial models; a real-world scenario is studied in detail utilizing the aforementioned principles.

2.6 Specialized Models for Optimization

2.6.1 Dynamic Optimization

For some complex problem domains, the most accurate modeling can only be conducted with special methodologies. Dynamic optimization encapsulates a wide range of approaches, all of which share the premise that non-stationary conditions in the problem being modeled are non-trivial and require specific consideration. Nguyen et al. in [103] present an overview of this field, including similarities and differences from stationary optimization as well as with current approaches.

A modeling tactic that has become popular is that of diploidy, or the dual chromosome representation; this was introduced to the optimization field by foundational works such as that of Goldberg and Smith in [104]. The approach derives inspiration from genomics, with the theoretical argument being that the secondary set of parameters in a second chromosome allows for faster adaptation of the population to new environmental conditions. The work of Yang in [105] describes a modified diploidy approach; in the author's formulation, pseudo-diploid genotypes are implemented by way of treating singular chromosomes as having a latent paired chromosome that exists with maximal parametric distance separating it. These latent chromosomes are only explicitly evaluated for some number of badly-performing individuals at each generation; this approach is argued to improve efficiency in the solution search.

Besides diploid chromosome inspired methods, other techniques have been investigated. A scheme using dynamic implementation of the differential evolution (DE) algorithm is

given by Mendes and Mohais in [106] to address problems with gradually-changing objective functions. In the authors' approach, multiple populations are used with the DE algorithm; the motivating idea is that each population will find a local minima and therefore give the procedure a faster adaption in the case that the global minima shifts. In a sense, this represents an exponential doubling of the population-based search metaheuristic at the core of EAs.

2.6.2 Multi-Population Optimization

Certain specializations in optimization involve the use of multiple populations which interact with each other. In an evolutionary paradigm, the evolutionary changes in one of these populations may trigger corresponding changes in the other population(s) as well; this is generally termed "co-evolution". These methods may be broadly split into those where the populations model sub-elements of individuals and behave collaboratively in nature and those in which the populations model groups of individuals that interact. In this second category, the groups of individuals may be simply delineated or alternatively may represent wholly distinct types of individuals.

Potter and Dejong in [107] describe "cooperative co-evolutionary genetic algorithms" (CCGAs) wherein partial elements of a complete solution are provided with their own distinct populations. These component populations each supply individuals to build a whole solution for evaluation. Ma et al. in [108] present a wide-ranging survey of relatively current cooperative co-evolutionary methodologies in this vein, while Antonio and Coello Coello review MO co-evolutionary practices in [109].

Within the field of multi-objective study and with the stated goal of improving search efficiency, Coello Coello and Reyes Sierra in [110] present an approach that calls for multiple populations to divide the parameter search space for a given problem. Depending on the productivity in terms of finding non-dominated solutions, the different populations' reproduction rates are manipulated.

Other efforts have sought to exploit a different model of interaction between populations; rather than using populations to hold separate elements of individuals, each population is tasked with holding a different set of individuals altogether. Maher and Poon in [111] explore

such a model of co-evolution for problems containing non-stationary elements, similar to the circumstances discussed in Section 2.6.1 above. Here, the authors propose a modified evolutionary algorithm in which the objective function is subject to evolutionary change, either as part of the parameter space or with its information contained in a separate population. This work (and subsequent works such as that of Maher and Tang in [112]) form a theoretical foundation for some of the design elements used in the work described in Section 5.4.2.

A game theoretical approach to multi-population optimization is presented by Sim et al. in [113]. The authors' formulation uses separate populations to create multiple "players"; at each step, these individual-players are drawn from each population and are assessed against each other in a competitive scheme. Individual-players that perform well against their competition according to objective functions are rewarded with an increase in fitness within their population. Another approach in this same vein of competitive optimization comes from Atashpaz-Gargari and Lucas in [114]. The authors present a model of multi-population optimization that they describe as inspired by imperialism. In their conception, each population is treated as an empire, with one leading "imperialist" individual that is selected based on best objective performance. The remainder of the population is composed of "colony" individuals, which are slowly moved towards the imperialist individual in parameter space. The complete method is given the name "imperialist competitive algorithm" (ICA).

A co-evolutionary method for dynamic MO optimization is put forward by Goh and Tan in [115]. With this effort, the authors aim to decompose MO problems using different populations to optimize disparate parts of a solution. They also analyze the performance of their approach in dynamic environments; by having separation between contributions to a complete solution to the problem, it is conjectured by the authors that faster adaptation to specific changes in the environment is possible.

Another competitive optimization formulation from Sharafi et al. in [116] proposes to synthesize a single methodology from the various types of nature-inspired optimization approaches developed in recent years (as discussed by Lones in [24]). The authors accomplish this by utilizing the ICA described earlier in this section to rank the efficacy of multiple populations of solution candidates, where each population makes use of a separate

optimization algorithm itself. Poor performance from members of a particular population will result in some of its individuals being removed and redistributed between the other populations. It is noted that this overall treatment shares characteristics with hyperheuristic approaches, which are described by Burke et al. in [117]; these methods have been theorized as providing effective mechanisms to deal with highly complex and/or dynamic problems.

Much of the work in this area as well as that reviewed in Section 2.6.1 has remained in the theoretical space to this point. The methodology advanced in Section 5.4.2 aims to apply techniques in this area to effectively model the dynamics of separate and distinct populations, each vying for optimality.

Chapter 3

Statement of Original Contributions

The effort presented in this text is thematically centered on the formulation of applied multi-objective radiological and radiation detection/monitoring problems and their evaluation using optimization methods. Encompassed within this premise are a number of case studies, which may be split into two broad groups based on their environmental context. The structured list below describes in detail these contributions.

- Modeling and Analysis in Natural Environments
  - Distributed sensor network modeling, with an emphasis on multi-objective modeling and optimization of solutions for the following problems:
    - Sensor network optimal node placement in natural topography to maximize sensitivity and spatial reconstruction ability with limited numbers of nodes
    - Path planning through areas of natural topography contaminated with radioactivity; consideration towards radiation exposure of traversing agent as well as path difficulty and length
    - Measurement update frequency optimized in relation to minimization of cumulative error of measurement as well as power consumption used by the deployed sensor node to transmit new measurements
  - Investigation of multi-robot exploration and mapping of radioactivity, with a focus on minimizing survey deployment time, radiation intensity reconstruction error, and spatial measurement uncertainty at survey end

- Modeling and Analysis in Urban Environments
  - Study of city-deployed radiation detection network in terms of optimal locations to maximize overall sensitivity using a limited number of sensor nodes
  - Assessment of detector network performance against intelligent adversarial avoidance:
    * Development and application of a dynamic multi-population competitive optimization framework for modeling non-stationary aspects of optimal strategies for network deployment and adversarial avoidance path planning
    * Evaluation of detector node placement strategies and general robustness of positions for detectors in the urban environment using an iterative path planning method

Chapter 4

Natural Environments

4.1 Modeling Framework

A network of sensor modules distributed across some region of natural environment is investigated as a means of comprehensively measuring radiation in the area over time. This forms the underpinning premise for a number of the applied problems examined in this chapter. The network modules are modeled to have some radiation detector along with capability for intercommunication with other nodes of the network. In this context, the sensor network has been modeled with a fusion center as well, this being the hub of datastream combination and local processing.

Generally, the modeling approach, visualized in Fig. 4.1, begins with the selection of some real-world location of interest. Real-world topographic and historical wind data are obtained for that location, which are then used to inform atmospheric deposition simulations. The output particulate distributions are collected into a source term dataset. Further review of the data used is presented in Section 4.1.1.

With these elements of the world in place, the sensor network itself can then be introduced. Sensor nodes are positioned in the environment using some algorithm and proceed to take measurements of local intensity based on the aforementioned distribution map. The fusion center module then acquires the data from each node and handles network algorithms such as intensity reconstruction and path-planning through the environment. Various methods have been tested for reconstruction, such as bicubic interpolation, Gaussian

process regression (GPR), and deep learned generative methods (pix-to-pix GANs and cycleGANs). More detail on this process is provided in Section 4.1.2 later in this chapter. Broadly, the modular approach to design used for the modeling of the sensor network allows for relatively easy modification of individual components of the simulated network to suit specific applications.

4.1.1 Data Sources

Topographic Data

Two regions with different characteristics are made the focus of investigation. The first (hereinafter "Region A") is centered on longitude and latitude of -112.729 and 43.874, respectively, near Idaho Falls, ID, USA. The second region (hereinafter referred to as "Region B") is centered on the longitude and latitude of -84.2425 and 36.00, respectively, close to Oak Ridge, TN, USA. For both regions, a $640,000\ m^2$ square section of topography is used, discretized into a 128-by-128 grid. These regions are visualized in Fig. 4.2. The topographic data for both of these regions are retrieved using the Py3DEP library [118]. It is also noted that this figure along with all other figures in this study are created using Matplotlib [119].

These two regions are selected to provide a range of terrain gradient variation. This is quantified by the measure of viewsheds, which describe for a specific point in a discretized topographic space the number of other points visible (not occluded) from it. This is calculated for all points in both regions along with the average; Region A is calculated to have an average viewshed magnitude of 668.62 whereas that of Region B is 386.63. This indicates quantitatively that there is significantly more terrain variation (hills, valleys, etc.) in Region B than Region A, and therefore it is expected that the analysis is more robust by evaluating models in both.

Deposition Data

In order to assess the capabilities of the systems and algorithms designed, data describing environmental radiation are needed. Unfortunately, real data of this type is extremely difficult to acquire as alluded to in 1.1.1. To overcome this limitation, a decision was

made to use simulations of particulate atmospheric depositions using GRASP (GPU Resident Atmospheric Simulation Program, [120]). These simulations are computed with consideration both to the real-world topographic maps discussed above in Section 4.1.1 and to the corresponding wind direction and magnitude distribution of the topography location based on historical wind data [121]. In each deposition simulation, samples are taken from these historical distributions to acquire wind parameterizations. The sampled parameters also inform the positioning of the contamination release point in the region. For example, if strong winds blowing eastward are stochastically chosen, the release will be positioned towards the western edge of the region studied so that a larger extent of the plume's behavior will be captured within the region of study.

In total, 1000 samples are generated for Region A. For Region B, a dataset of 2385 samples is generated. All samples are cropped to a sixteenth of the total size of the region ($40,000\ m^2$ square, forming a 32-by-32 discretized grid), with the crop location being based on the position of the highest concentration of deposition. This is done in order to enforce a lower degree of sparsity in the samples than would otherwise be present. Each cropped sample is matched to a corresponding cropped map of its topography for later analysis considerations. Samples are shown for both regions in Fig. 4.3.

4.1.2 Reconstruction

Let it be said that the latent mapping between two domains in data, which may be termed *source* and *target*, can be stated in the form $G : source \rightarrow target$. This source domain defines the data that are obtainable, whereas the target domain describes the goal of the reconstruction or translation. Finding a mapping between these domains can be accomplished with a wide range of methods. For this purpose in the context of this work, the techniques of bicubic interpolation and GPR are examined.

It was also sought to test the viability of using an image-to-image (I2I) model as a spatial intensity reconstruction method in addition. Specifically, a pix-to-pix GAN (p2pGAN) and a cycleGAN were implemented and evaluated. The former model was first described extensively in [122] and is a conditional approach, meaning that data samples in the source and target domains must be paired. The cycleGAN model, presented by Zhu et

al. in [123], sidesteps the need for paired samples via the addition of a cycle consistency loss, which constrains the solution space to also minimize the loss of the reverse mapping $F : target \rightarrow source$ such that $F(G(source)) \approx target$. The source domain may describe the sparse measurements collected with our sensor network or the results of either the interpolation or GPR methods. Conversely, the target domain represents the ground truth of the source term. Ideally, the translational model would be able to learn some latent mapping between the source domain and the target that may partially account for the lack of complete data in the measurements and/or imperfections of the two basic reconstruction methods employed.

Qualitative examples of reconstruction of the radiological intensity distribution from sparse measurement data using bicubic interpolation and GPR are shown in Fig. 4.4a, while a sample usage of the p2pGAN reconstruction is provided in Fig. 4.4b.

Subsequent analysis of the comparative quantitative performance of these methods indicated that interpolation and GPR tend to provide more accurate reconstructions than the deep-learned I2I models as implemented. These results are briefly detailed in Appendix A. It is for this reason that solely interpolation and GPR are utilized for reconstruction in the work described in Section 4.2.

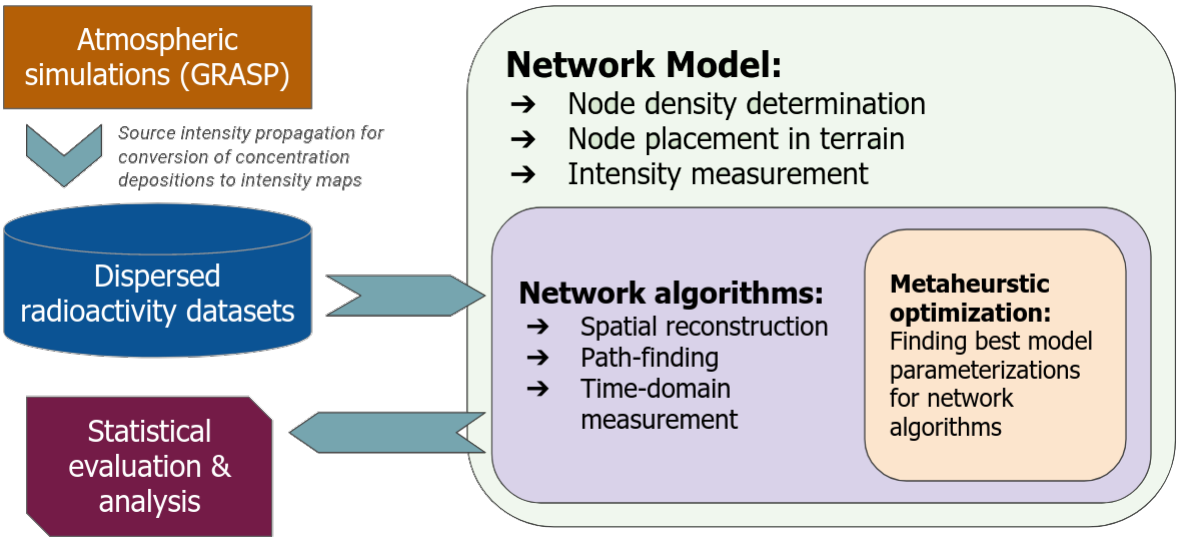


Figure 4.1: Diagram of software modeling framework for radiation detection sensor network.

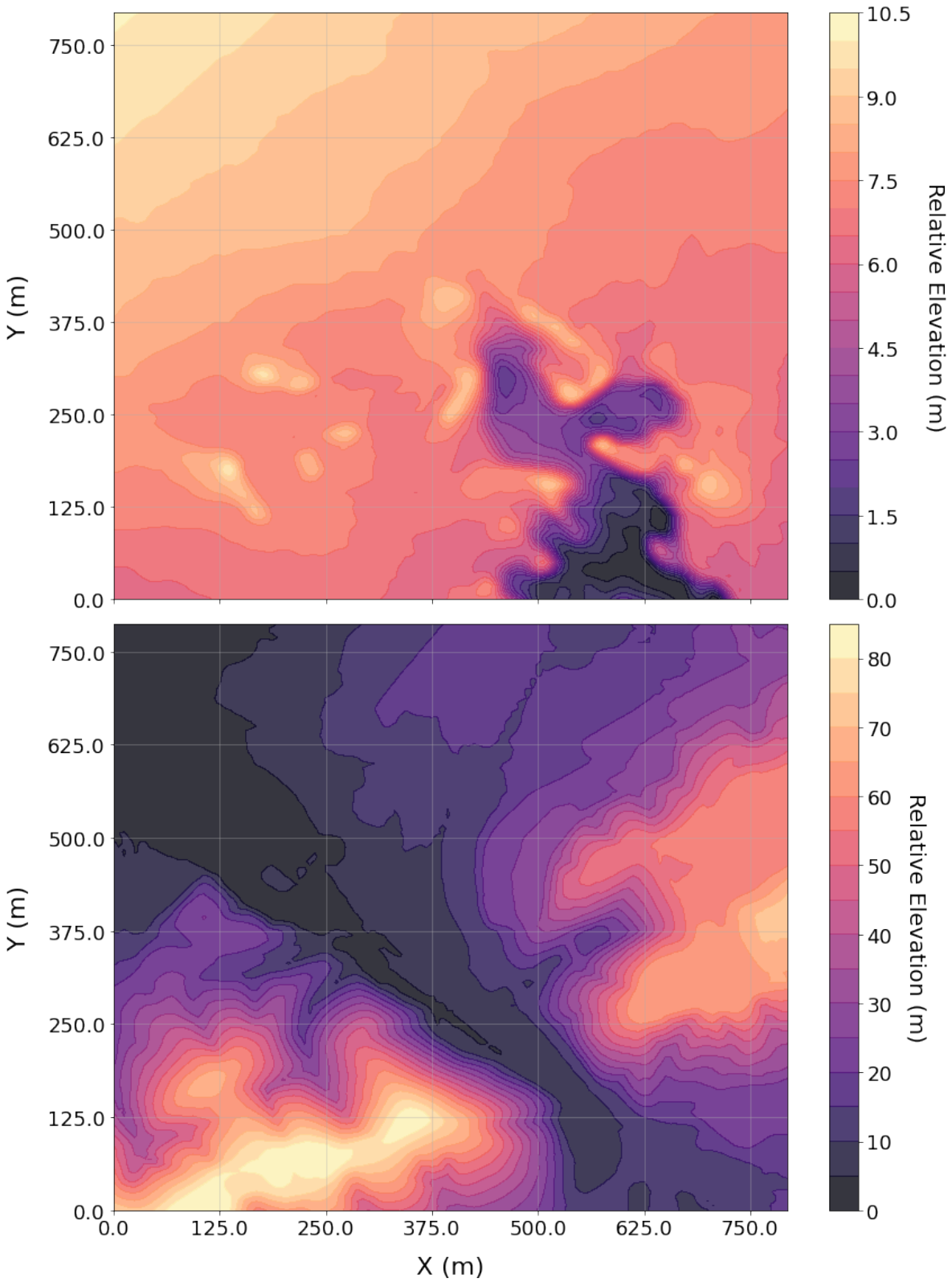
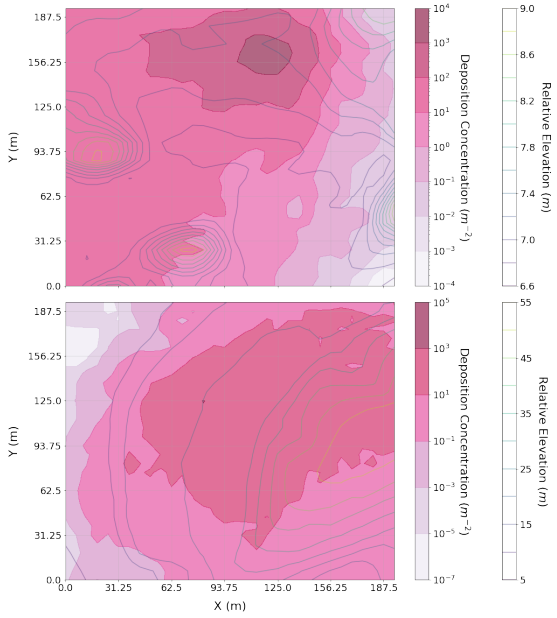
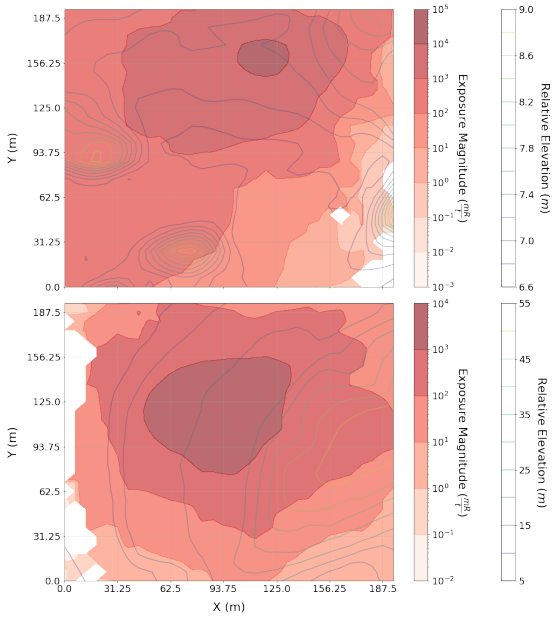


Figure 4.2: Topography of Region A (top) and Region B (bottom).

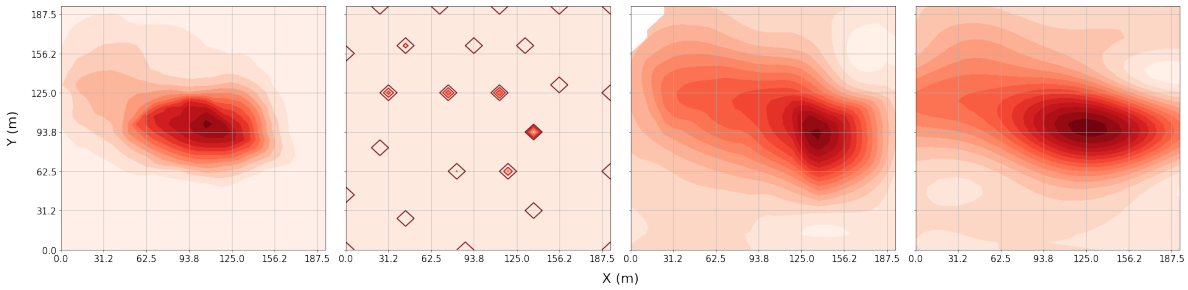


(a) Cropped simulated radioactive contaminant depositions in Region A (top) and Region B (bottom).

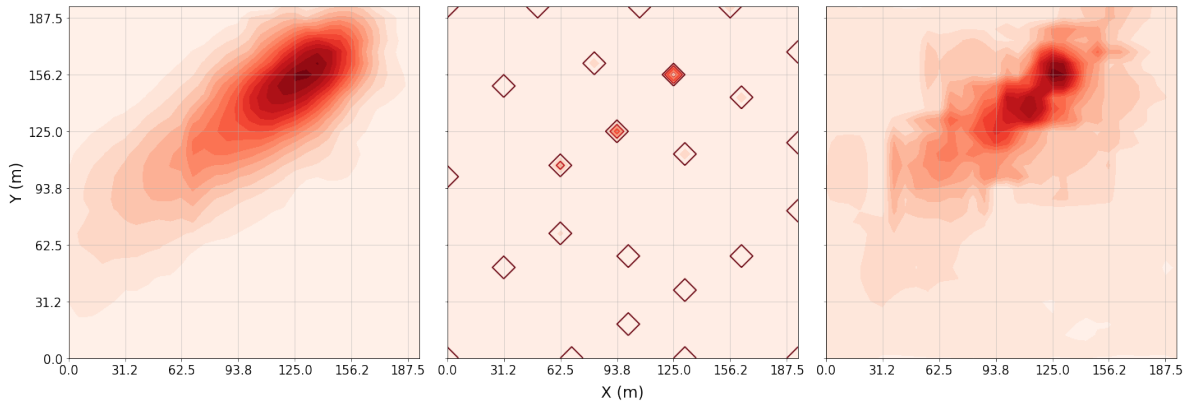


(b) Radiological intensity distributions in Region A (top) and Region B (bottom) corresponding to depositions in **(a)**.

Figure 4.3: Sample cropped simulated depositions and their corresponding radiological intensity distributions.



(a) Reconstructions of the ground truth intensity distribution (**far left**) based on sparse sensor measurements (**center left**). Interpolation (**center right**) and Gaussian process regression (**far right**) results shown.



(b) Example of p2pGAN reconstruction (**right**) of ground truth (**left**) from sparse sensor node measurements (**center**).

Figure 4.4: Reconstructions of radiological intensity distributions using interpolation, GPR (a), and p2pGAN (b) methods. Scenario is with 25 nodes in Region A (see Section 4.1.1).

4.2 Node Placement

For some number of nodes comprising a sensor network, it is not at all obvious how to place them within an environment to maximize their ability to measure distributed radiological source terms. Obviously, it would be best to simply saturate the environment with hundreds or thousands of nodes, but as discussed in Section 1.2.1, this is likely not a practical solution in the applied context due to cost and complexity of the resulting system.

Without knowing where the source term will be located in the environment *a priori*, it is intuitive to imagine that the sensor nodes must be spread throughout the space to account for the possibility of areas of high concentration being present anywhere. Determining an optimal spread pattern has been an important part of researching the sensor network system for this application. Various algorithms from literature have been investigated and then modified to suit the contextual needs. These include a random placement, farthest-first traversal [124], and non-weighted and weighted versions of Lloyd's algorithm for finding Voronoi centroids [52]. The weighted versions of Lloyd's algorithm are designed to bias node placement towards areas of higher environmental sensitivity[1] or sensitivity gradient based on the physical topography. The justification for these methods is that such regions will provide the network with greater overall robustness to an unknown source term or account for potential terrain occlusion that may affect sensor measurements or may have already affected source term formation or deposition. Example placements are shown in Fig. 4.5.

However, these node placement methods are only one parameter of the overall network design. The work is taken a step further by formulating the sensor node placement problem as multi-objective and subsequently applying a multi-objective optimization algorithm. In this analysis, the placement algorithm becomes one parameter amongst others, these being the number of nodes and the intensity reconstruction algorithm. Discrete parameter values for node quantity between 4 and 100 are evaluated, and the options for reconstruction algorithm include bicubic interpolation and GPR as discussed in Section 4.1.2.

Objectives to consider include performance of the resulting network's reconstruction with respect to the ground truth source term distribution, number of nodes used, overall sensitivity

[1] The sensitivity of a particular location in the environment is calculated by computing its viewshed based on the surrounding topography.

of the network based on the topography, and average minimal distance between nodes (intercluster distance). The reconstruction objective is quantified using the mean structural similarity (MSSIM, [125]), which is a more robust metric of assessing image similarity than the L1 or L2 metrics. Sensitivity of the network is formulated as an objective in order to provide an alternative and perhaps more robust means of estimating a network's ability to detect source terms; with this metric, the particular source term used in evaluation does not bias the result. An average of minimal distance between the nodes of the network is of interest as an objective since it greatly helps to quantify the degree of sensitivity robustness to source term placement in another way. Put in other terms, if a particular network deployment has very low average minimal intercluster distance, its nodes must therefore be clumped together; the network will consequently likely have very inconsistent reconstruction ability.

4.2.1 Evolutionary Optimization

To find an approximate Pareto set of the parameter values for FPA*, an explicitly MO optimization algorithm must be applied. As is recounted in Section 2.2, there are quite a number of ways to go about this. The specific method used is a slightly modified version of MOGA [64, 65] which also implements an external memory "archive" to store optimal solutions discovered as an elitism mechanism. This custom implementation was utilized in order to ease deployment into the development and analysis pipeline used in this work.

4.2.2 Results

The results of the optimization are the generational history of the individuals, which is shaped as non-dominated solutions are continually preferred and allowed to reproduce; an approximated Pareto front in the multi-dimensional objective space is ascertained by examination of the final results. Examining individual solutions on this front allows for insight about which parameterizations are best with regards to specific objectives. For instance, given an applied circumstance wherein optimal reconstruction performance was required while utilizing a minimal amount of nodes, looking the Pareto front approximation

in the two-dimensional space of those two objectives would guide towards a solution with a parameter set to use.

Pareto Fronts

The estimation of the Pareto front obtained by application of the MOEA in Region B topography is presented with 2D projections in Figs. 4.6 and 4.7. While the solutions reside in four-dimensional objective space, it is far easier to visualize these lower dimensional projections. However, it should be noted that this is the reason why it is somewhat difficult to pick out a curve akin to that shown in Fig. 1.1 in the distribution of archive points that forms an obvious Pareto front.

Immediately apparent from the results in Fig. 4.6a are the greatly diminishing marginal improvements to the reconstruction metric with increases in nodes placed beyond a quantity of roughly 35-40. In Fig. 4.6b, an intuitive asymptotic trend in the average node separation as the quantity of nodes placed increases is observable. The distribution seems also to be comprised of two distinct lobes as well.

Fig. 4.7 presents projections of the objective space with respect to the average minimum intercluster distance objective. Distinct clusters of individual solutions around values of average minimum intercluster distance are observable in the complete distribution in Fig. 4.7a. The reason for this is investigated further in the next part of this Section with Figs. 4.9, 4.10, and 4.11. An inverse trend is also present in the distribution shown by Fig. 4.7b, although the asymptote is positioned at a low y-value rather than a high one as in Fig. 4.7a. This can be argued to reflect the different natures and goals of the reconstruction similarity and sensitivity objectives, these having been outlined previously in Section 4.2.1. The reconstruction similarity will show optimal solution candidates for a particular applied case, which may introduce bias. The sensitivity aggregate will on the other hand likely select for more broadly well-performing network configurations, but is a more abstract measure that is not a guarantee of accurate reconstruction ability for any given scenario.

Parameter Shading

Further information can also be gleaned from analyzing the optimization input parameter distributions across the results. The MOEA output presented in Fig. 4.6b is colored according to the placement algorithm utilized in node deployment in Fig. 4.8. With Figs. 4.9, 4.10, and 4.11, the outputs presented in Figs. 4.7a are colored by the placement algorithm used, the spatial reconstruction method applied, and the number of network nodes placed, respectively. This procedure allows for much greater explainability of the optimization results.

With Fig. 4.8, it can now be determined that the multiple-lobe structure observed in the original distribution figure is a result of differences in node placement algorithm applied. Farthest-first traversal exhibits better results in terms of making optimal tradeoffs between these two objectives. The cluster of solution candidates that used a random placement also comes into focus as having very poor average node separation. This allows for the intuitive conclusion that networks constructed in this way will have extremely inconsistent performance in reconstructing source terms.

In another case, a trivial examination of Fig. 4.10 allows for the judgment that GPR tends to provide better results across the range of network sparsity metric values in terms of radioactivity reconstruction accuracy that bicubic interpolation. In Fig. 4.11, the nonlinear relationships between the number of nodes comprising the network and both the average network sparsity metric along with the resulting reconstruction accuracy come into focus. Specifically, it could be argued that marginal improvements to reconstruction metric values greater than about 0.80 can primarily be obtained via adding more nodes to the network. This is in contrast to the obvious great variation in reconstruction performance for lower-node-quantity networks.

Aside from an understanding to some degree about the relationships between the input parameters to the problem and the optimal objective value outputs, the practical takeaway from these results lies with the ability to select from amongst the optimal points located in the elitist archive after the optimization is complete. There is no singular best solution from among these; instead, a DM would be able to choose one based on a particular circumstance

or necessity, being assured that all provide the best possible trade-offs in performance across the objective space.

4.3 Path Planning

Introduced briefly in Section 1.2.2, the premise of MO path-planning through a radiologically-contaminated environment as considered thus far is that minimizing dose, maintaining a relatively short path, and avoiding physical obstacles all bear some degree of importance. This has been approached in the work described here by developing a Pareto-optimal A* algorithm that combines the ability to find non-dominated solutions with the efficiency of quick pathfinding.

4.3.1 Heuristics

The traditional A* as first described in [78] minimizes a sum F of G and H cost at each step, corresponding to sunk and heuristic cost, respectively. In this multi-objective version, G and H are not summed and their respective costs must be determined for each objective. This presents a problem; while it is trivial in the single-objective case to determine a heuristic for shortest distance, it is far less obvious what a heuristic for minimizing dose or terrain complexity traversed should look like. To solve this for the case studied, new heuristics and methods to calculate them had to be formulated.

Beginning with the terrain objective, focus is placed on designing a heuristic that measures the minimum topography gradient necessarily encountered in a path between a source and destination node. Taking inspiration from Dueck's great deluge algorithm [126], the method described in this section initializes a permissible level of gradient for traversal (akin to a "water level") at the local topography gradient of the source point. A breadth-first search is then performed wherein nodes that have a local gradient at or below the permissible level are added to a set. If the destination node is connected to the set, the algorithm terminates and returns the current gradient as the heuristic value. However, if the destination cannot be connected, the permissible gradient level is raised one step. This

process is repeated until the destination node is connected. The pseudo-code for the method is shown in Appendix B.

For the exposure cost heuristic, a modification of Dijkstra's algorithm [79] in the manner recommended by Tung and Chew [127] is used to utilize cumulative exposure rather than distance traveled as a cost metric. While being slow to calculate, this method allows for a reliable measure of the minimal possible exposure to radioactivity along any path from a source to destination. The pseudo-code for this approach is shown in Appendix B.

4.3.2 Algorithm Formulation

With the issue of heuristics for the new objectives solved, the MO A* algorithm can be assembled. As alluded to in Section 2.4, this approach aims to build upon the previous work of Stewart and White in [80] with the integration of attribute parameterization to find tractable solutions more often. In some regard, this may result in solutions somewhat "fuzzier" in terms of optimality than those that might be arrived at in a strictly MO, non-dominated procedure. However, many practical applications present circumstances that can make non-dominated selection impossible; all choices at a juncture may be incomparable. This often occurred for this problem during development and testing of early approaches to this problem, making non-dominated selection paradigms difficult to realize. For the reasons above, the novel algorithm is termed "fuzzy Pareto A*" (FPA*) for its approach to selecting the best choices for nodes in situations where no non-dominated solutions are available.

FPA* works generally with the framework of traditional A* by maintaining open and closed sets of nodes to consider visiting and nodes already visited, respectively. The first difference is the accountancy of multiple objectives instead of only distance; this, of course is common to any MO approach to A*. As previously mentioned, the uniqueness of FPA* comes from its parametric approach to management of these multiple objectives. This is accomplished with two key additions. The first is maintaining a "domination matrix" that stores the results of domination evaluations across the open set. It accomplishes this by tracking the number of times and for which objectives nodes are dominated by others. The second addition is that of objective weights; these are used not to directly weight the objective values, but instead to balance a sum calculated for each node of its row in the aforementioned

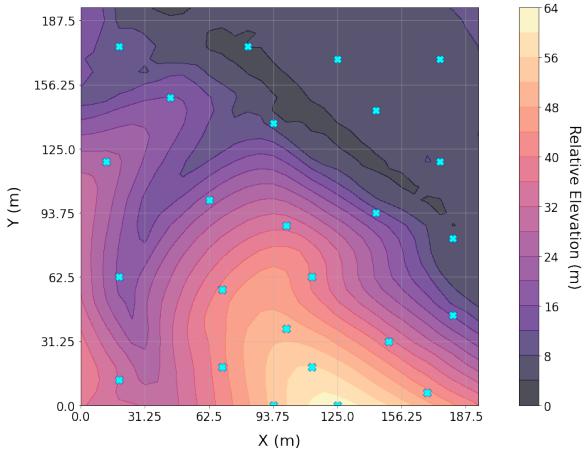
domination matrix. The minimum of these sum values is used to select the next active node. In this way, the algorithm can overcome incomparability stalemates and be guided towards preference of certain objectives depending on application. Psuedo-code for the FPA* algorithm is shown in Appendix Section B of this dissertation.

To more clearly put into focus the advantage of this novel approach, a case may be considered where there are three nodes for potential expansion in the open set in Table 4.1. With a strict MO approach, it would not be possible to make any distinction between the fitness of these nodes as all of them are non-dominated with respect to each other. FPA* allows for focus instead on the nodes' domination in each category; despite the nodes still each being dominated an equal number of times in total, the ability to use parametric weights to vary the importance of these domination categories will make it possible to break the tie and make a choice. For example, if a set of weights are distance cost: 0.5; terrain cost: 0.5; exposure cost: 1.0; the weighted domination numbers for node A, B, and C will be 1.5, 2.0, and 2.5, respectively. Therefore, node A should be expanded.

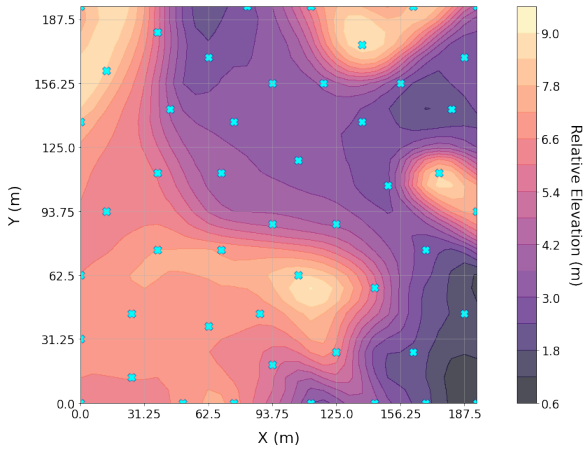
The parameterization design choice of the FPA* algorithm opens up MOO possibilities. As will be demonstrated, this affords even greater flexibility to the end-user as they receive options of a range of paths to choose from that span the approximate Pareto front.

4.3.3 Objectives

As discussed in Section 4.3.1, two additional objectives are taken into account along with that of path distance. The exposure cost is defined as the cumulative exposure of the traversing agent to radioactivity along the path. Subsequently, the terrain cost is the quantified measurement of the difficulty posed by variation in elevation along the path. With regards to their calculation, the reader may refer to the detail provided previously in Section 4.3.1. Together, these are the three objectives considered by FPA* and, therefore, also by the MO optimization of its parameters.



(a) Placement using 25 nodes in Region B topography (contours shown) using a sensitivity-weighted version of Lloyd's algorithm for finding Voronoi centroids.

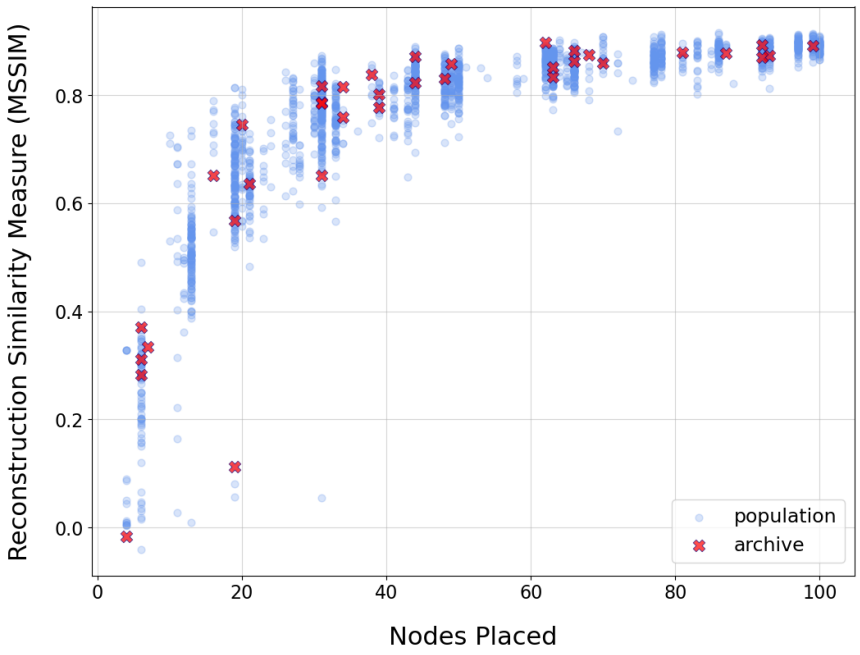


(b) Placement with farthest-first traversal algorithm using 49 nodes in Region A topography, serving to contrast with 4.5a as the region generally has much lower elevation variability.

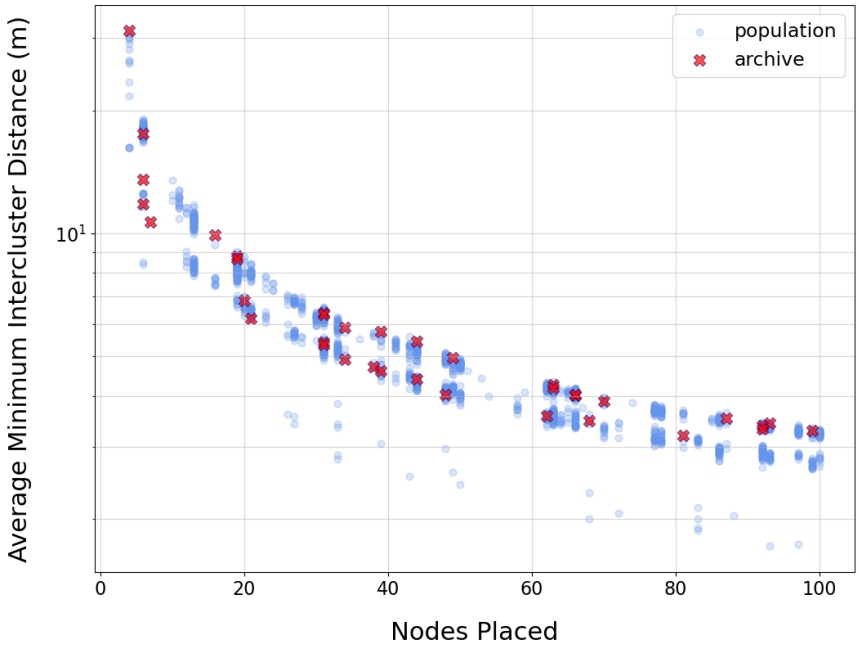
Figure 4.5: Example maps of node placement on real-world topography.

Table 4.1: Example of incomparable node choices. Each of the three nodes here are non-dominated with respect to the other two. It is noted that these values are chosen purely for purpose of example and are not necessarily realistic in the context of the problem.

Node ID	Distance Cost	Terrain Cost	Exposure Cost
A	4.0	2.5	0.1
B	1.0	3.3	2.0
C	1.2	0.2	8.0

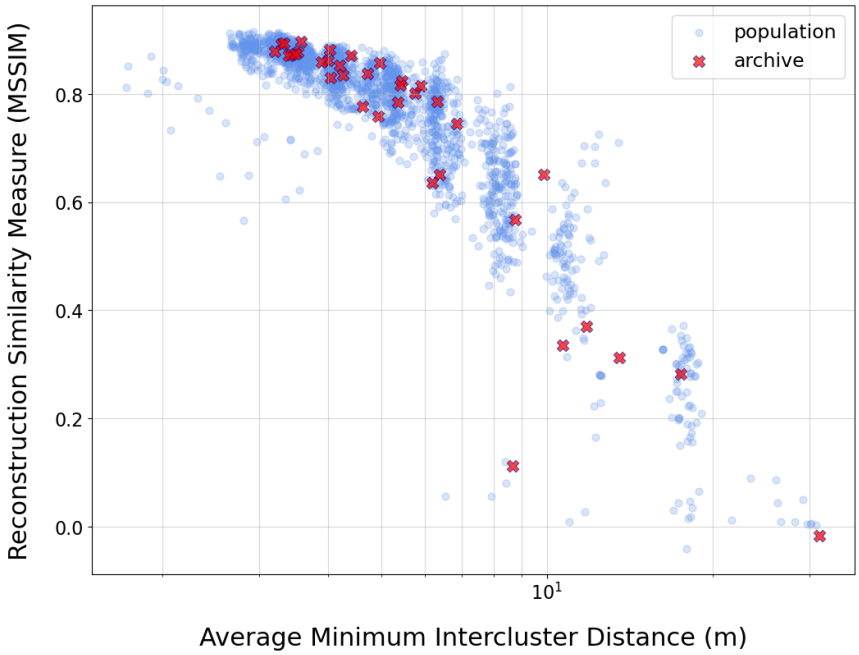


(a) Node quantity against reconstruction similarity.

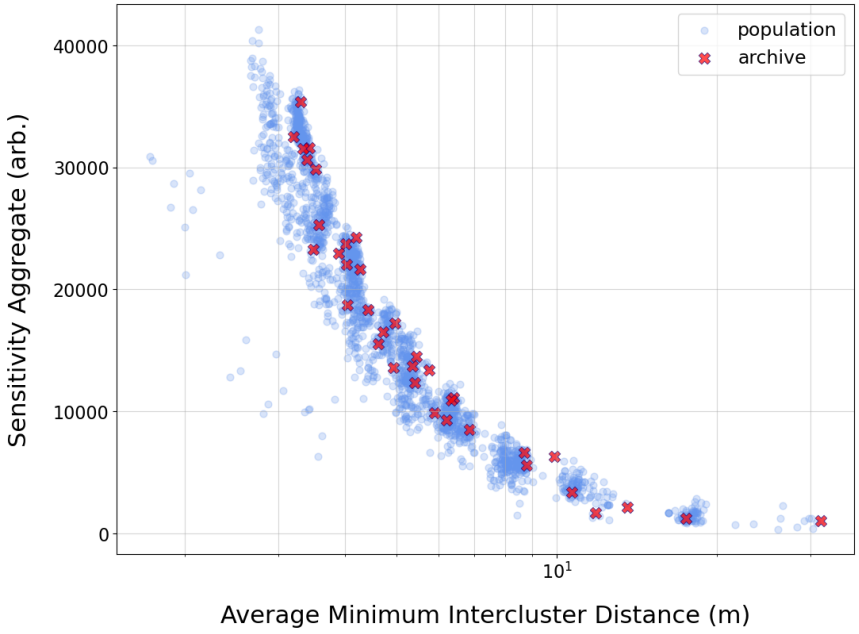


(b) Node quantity against average minimum intercluster distance.

Figure 4.6: Evolutionary MO optimization outputs in the form of the historical solution distribution in objective space. Each plot depicts a 2D projection of the problem's 4D objective space. Optimal solutions found and stored in elitist archive are displayed in red, while all solutions evaluated throughout optimization are displayed in blue.



(a) Average minimum intercluster distance against reconstruction similarity.



(b) Average minimum intercluster distance against aggregate network sensitivity.

Figure 4.7: Evolutionary MO optimization outputs in the form of the historical solution distribution in objective space. Each plot depicts a 2D projection of the problem's 4D objective space. Optimal solutions found and stored in elitist archive are displayed in red, while all solutions evaluated throughout optimization are displayed in blue.

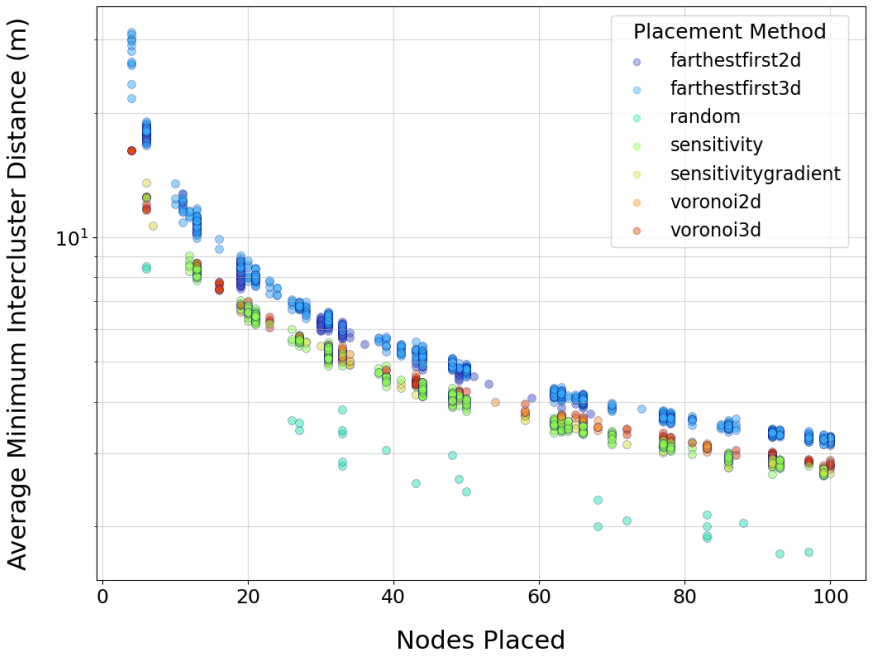


Figure 4.8: Node placement algorithm parameter shading of approximated Pareto front projection shown in Fig. 4.6b.

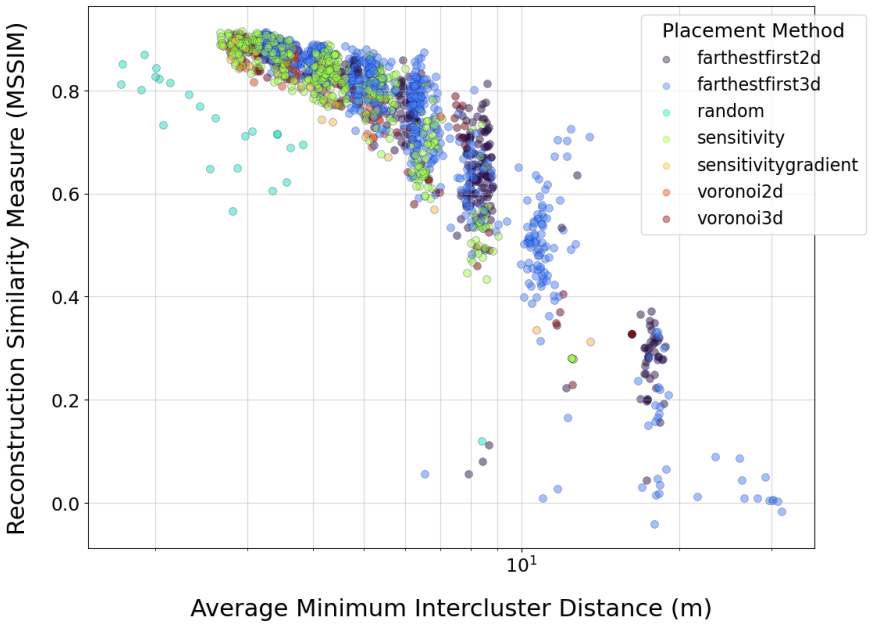


Figure 4.9: Node placement algorithm parameter shading of approximated Pareto front projection shown in Fig. 4.7a.

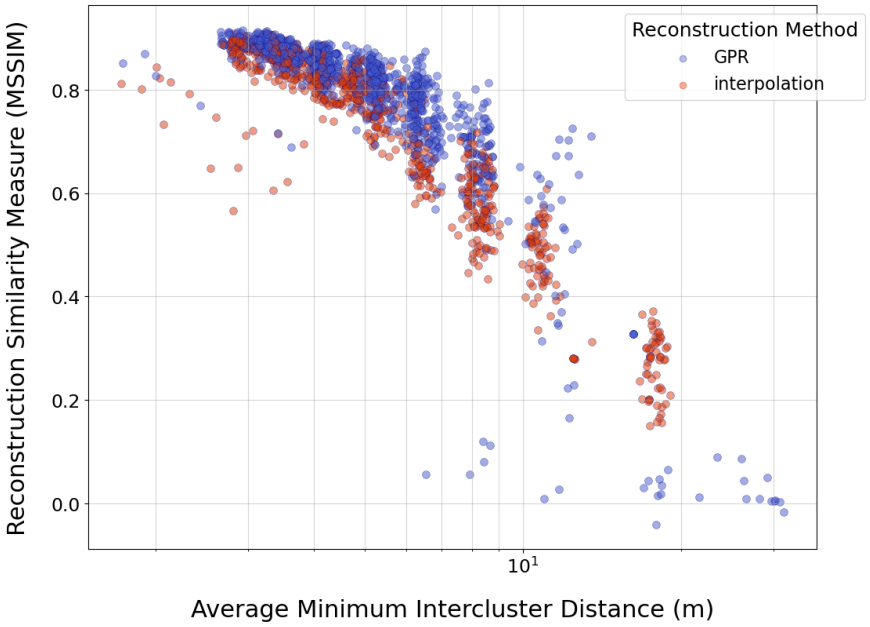


Figure 4.10: Reconstruction method parameter shading of approximated Pareto front projection shown in Fig. 4.7a.

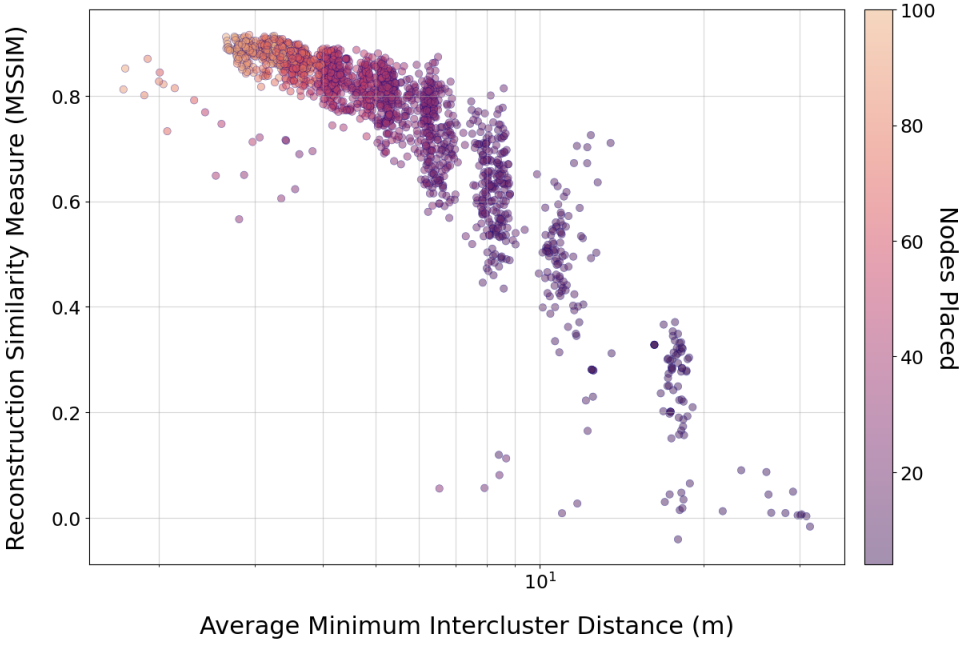


Figure 4.11: Quantity of nodes placed parameter shading of approximated Pareto front projection shown in Fig. 4.7a.

4.3.4 Parameters

The fuzzy design choice of the path planning algorithm allows for the incorporation of weight values that can provide differing levels of importance to the domination of path nodes with respect to certain objectives. Therefore, the parameters correspond to the objectives considered. There are six total, two each for the weights of the distance cost, terrain cost, and exposure cost. The purpose of this doubling has to do with the A* basis of the FPA* algorithm; one weight parameter is given for each of the g-value and the h-value costs which are calculated for each prospective node.

With respect to the values of the parameters to select from, a continuous search space ranging from 0.0 to 1.0 is used. On the subject of constraints, it is judged sensible to confine the parameter values to a range between values of 0.0 and 1.0 as this allows for easy comparison as normalized weights. No explicit objective value constraints are used in this analysis.

4.3.5 Optimization

As in Section 4.2, a MOEA is utilized to find optimal solution candidates. In this case, NSGA-II [33] is selected for this purpose; the specific implementation provided by the PyMOO library [36] is utilized. The following hyperparameters are used across 25 separate environment-path scenarios: population size of 36, 12 offspring per generation, and a maximum number of generations of 24. The quantity of 25 experiments is conducted to analyze the effect of what are termed "hyper-descriptors" on the optimal path parameter results. These hyper-descriptors include factors such as region selection, radiological distribution positioning, minimum possible path length, and objective costs at source and destination among others. In Section 4.3.6, the results of this are reviewed with the idea of providing a form of sensitivity analysis for FPA*.

4.3.6 Results

Optimization runs are conducted with different paths in different environments to obtain best parameterizations for FPA* in each case. For brevity, the results of one of these experiments

are put into close focus here. These results from this experiment will be referred to as "run *B-3*", as it was conducted for a path through an environment from Region B and the radiological distribution sample index 3. The environment and radiological intensity contours along with the source and destination points are visualized in Fig. 4.12. This combination is selected to capture an expressive case of the objectives in competition. For instance, to incur the least exposure to the radioactivity, the agent must plan a path over the steep incline in the east. On the other hand, a path that minimizes distance will expose the agent to a very high amount of radioactivity.

The output of the NSGA-II MOEA is its final population, which ideally approximates the Pareto front of the problem. Following from this, the output for run *B-3* contains the best solution candidates, which are shown as a 3D distribution in Fig. 4.13. A wide approximated Pareto front in the objective space is observed, which suggests many optimal paths. It is noted that the parameterized approach of FPA* preserves and affords the flexibility of choice between these options to a hypothetical DM.

From the optimized population presented in Fig. 4.13, paths are planned then sampled and displayed using the individuals' parameterizations in Fig. 4.14. The path in Fig. 4.14a minimizes exposure cost, but this forces the agent over the high elevation gradient which causes it to suffer in terms of terrain cost. The distance cost is also high as the path prioritizes staying away from the source term. With paths in Figs. 4.14b and 4.14c, there is a more balanced objective profile. In the former, the terrain cost weight parameters result in the circuitous route around elevation gradients; this also results in the path straying nearer to the highest intensity magnitude of the radioactivity and therefore incurring more exposure cost. The latter of these two paths has lower terrain weights but higher distance weights, resulting in a more straightforward path in comparison. The path in Fig. 4.14d has low exposure cost weights and high terrain cost weights, which provides a path that treks over easy terrain but exposes the agent to a very large amount of radiation. The path shown in Fig. 4.14e has a cost weight parameterization very similar to that seen in Fig. 4.14c, with the main difference being increased terrain weights. The effects of this are reflected in the path contorting to avoid areas with higher gradients and incurring slightly higher exposure in the process. With Fig. 4.14f, the path is again planned with a set of weights similar to

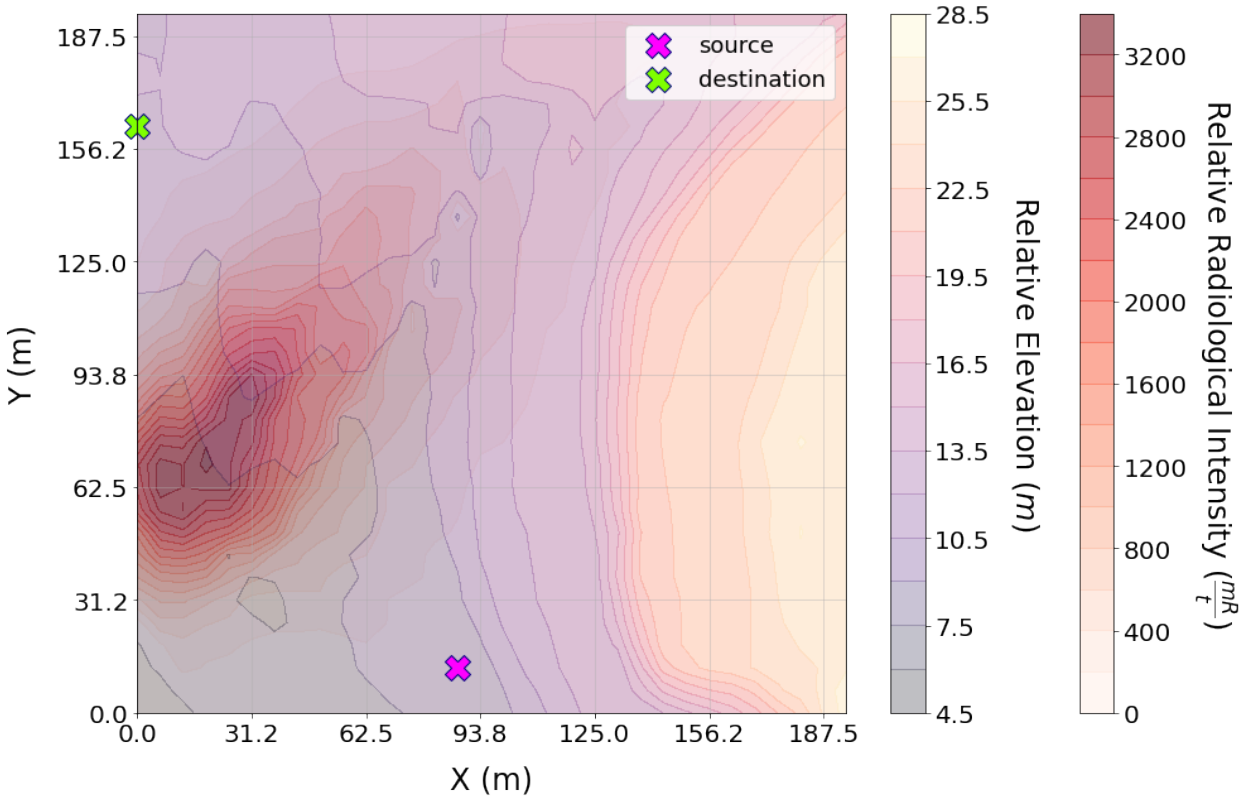


Figure 4.12: Plot of radiological intensity distribution and topography along with path source and destination optimized for in run ***B-3***.

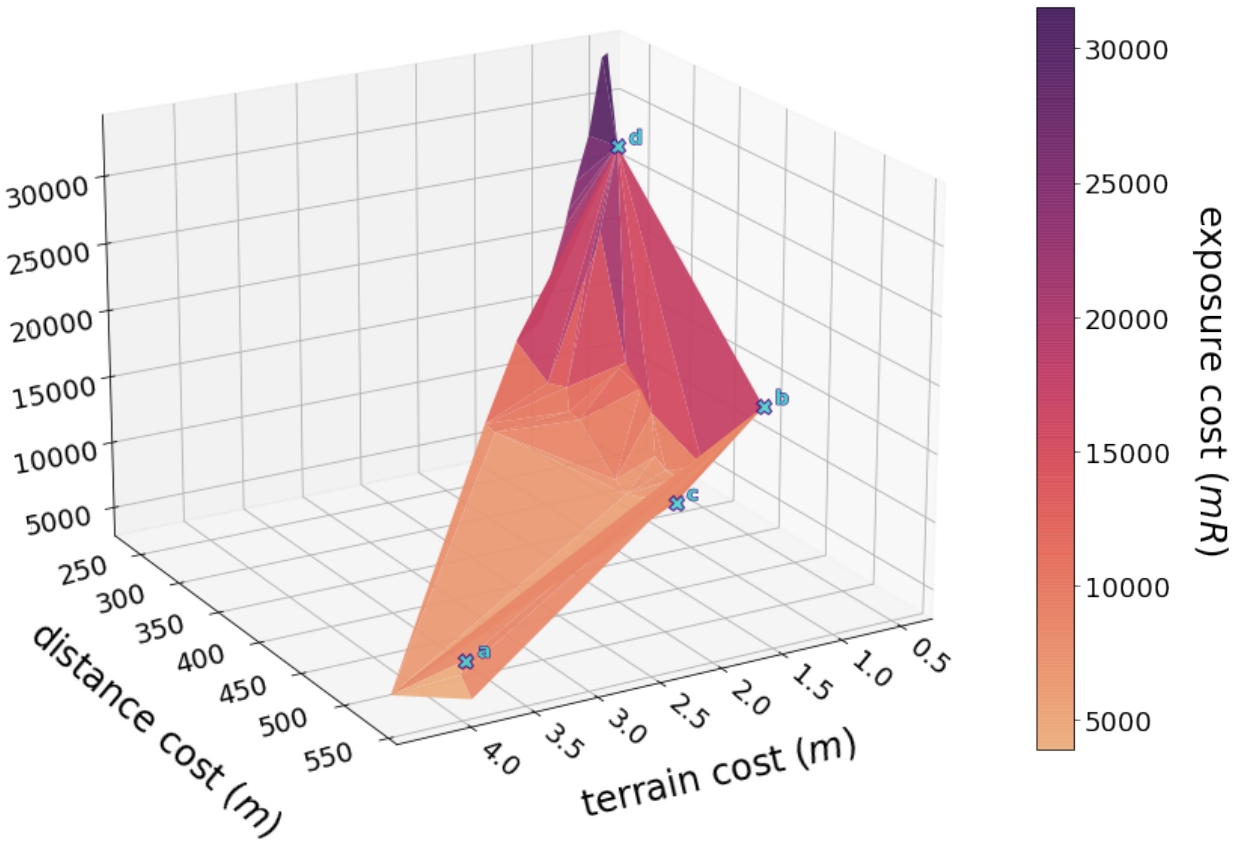


Figure 4.13: Distribution of final population of the NSGA-II MOEA for run ***B-3*** visualized as a surface in the three-dimensional objective space. Points shown correspond to the sample paths displayed in Fig. 4.14.

those used to compute that observed in Fig. 4.14d. The higher distance cost weights are manifested in a more direct path in this case. While these are only 6 samples out of the final optimized population of 36, they already provide a range of choice and, therefore, flexibility to a DM.

Additional optimal path outputs for optimizations of different problem configurations are presented in Appendix C.

Parameter Shading

Fig. 4.15 shows the complete population histories from run *B-3*, with each plot having its points shaded by a different parameter. The Pearson correlations between each parameter and its relevant objective are also provided in the top left of each plot. As would be expected, negative correlations are seen in all cases. It is noted with interest the discrepancies between these values for g- and h-value weights for the same objective; for the distance and exposure objectives, the h-value parameter has a significantly higher correlation with the resulting cost than the g-value. With the terrain cost, the g-value correlation is greater but to a lesser extent. One possible explanation is that the formulation of the terrain cost does not treat it as cumulative and instead uses the maximum traversed value.

Sensitivity Analysis

As mentioned in Section 4.3.5, optimization is run for FPA* on a range of environments, radiological distributions, and path sources and destinations. In total, 25 distinct combinations are tested.

A plot of the correlation between certain quantified values of these hyper-descriptors and model outputs is shown in Fig. 4.16. There is notable correlation between some of the hyper-descriptors and the optimized FPA* objective value outputs, which is to be expected; it is intuitive that a source or destination location on a path being positioned in an area of higher terrain cost or exposure cost would have a direct effect on the minimal possible value for that objective. However, it also is importantly noted that the largely weak correlation values between the optimal values' corresponding parameters and the hyper-descriptor values. This would seem to suggest that the optimal parameterizations for FPA* possess some degree

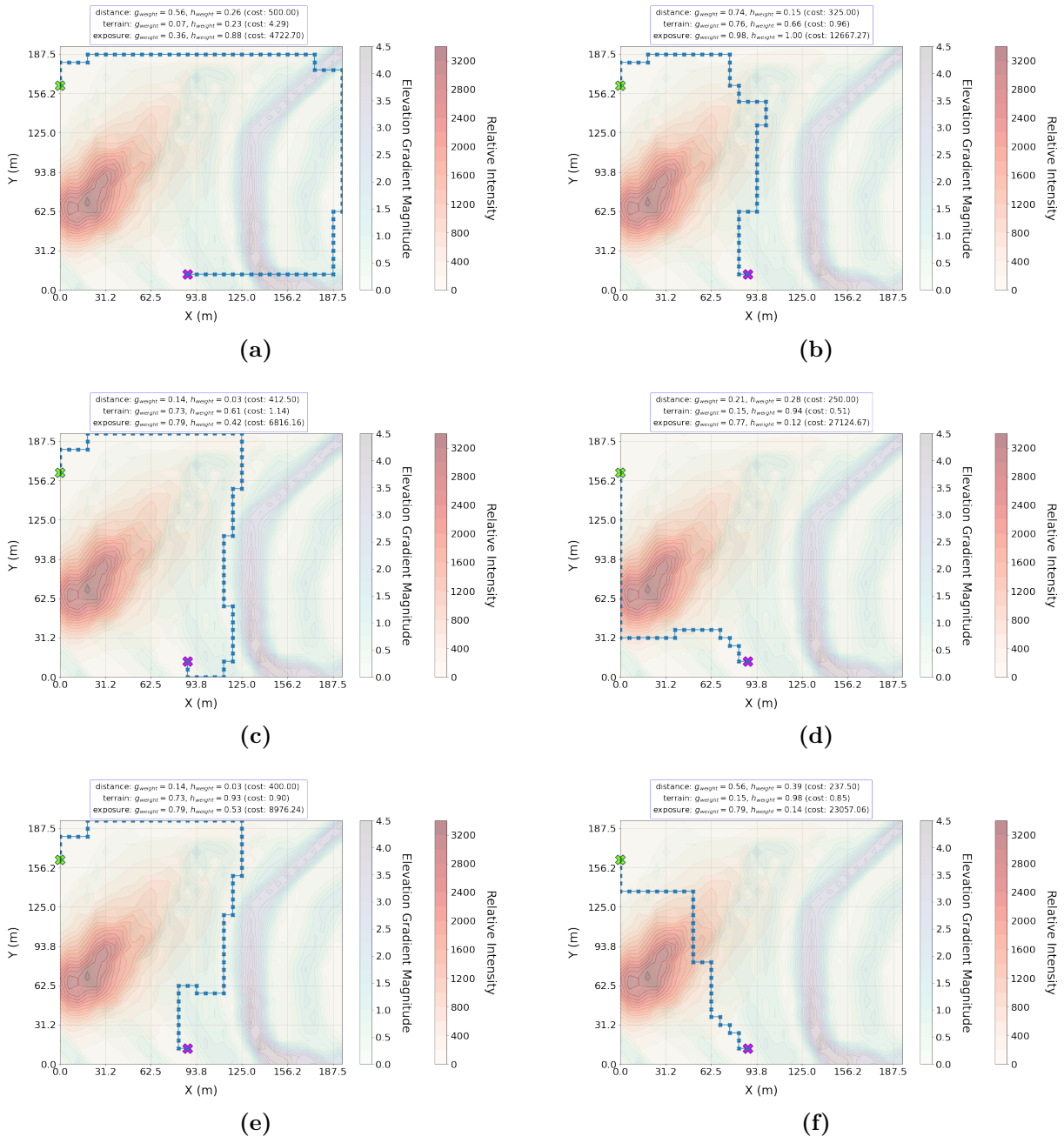


Figure 4.14: Selection of path samples using optimized parameterizations for FPA* from run ***B-3***. Path shown in blue, source and destination of path displayed in fuchsia and lime green, respectively.

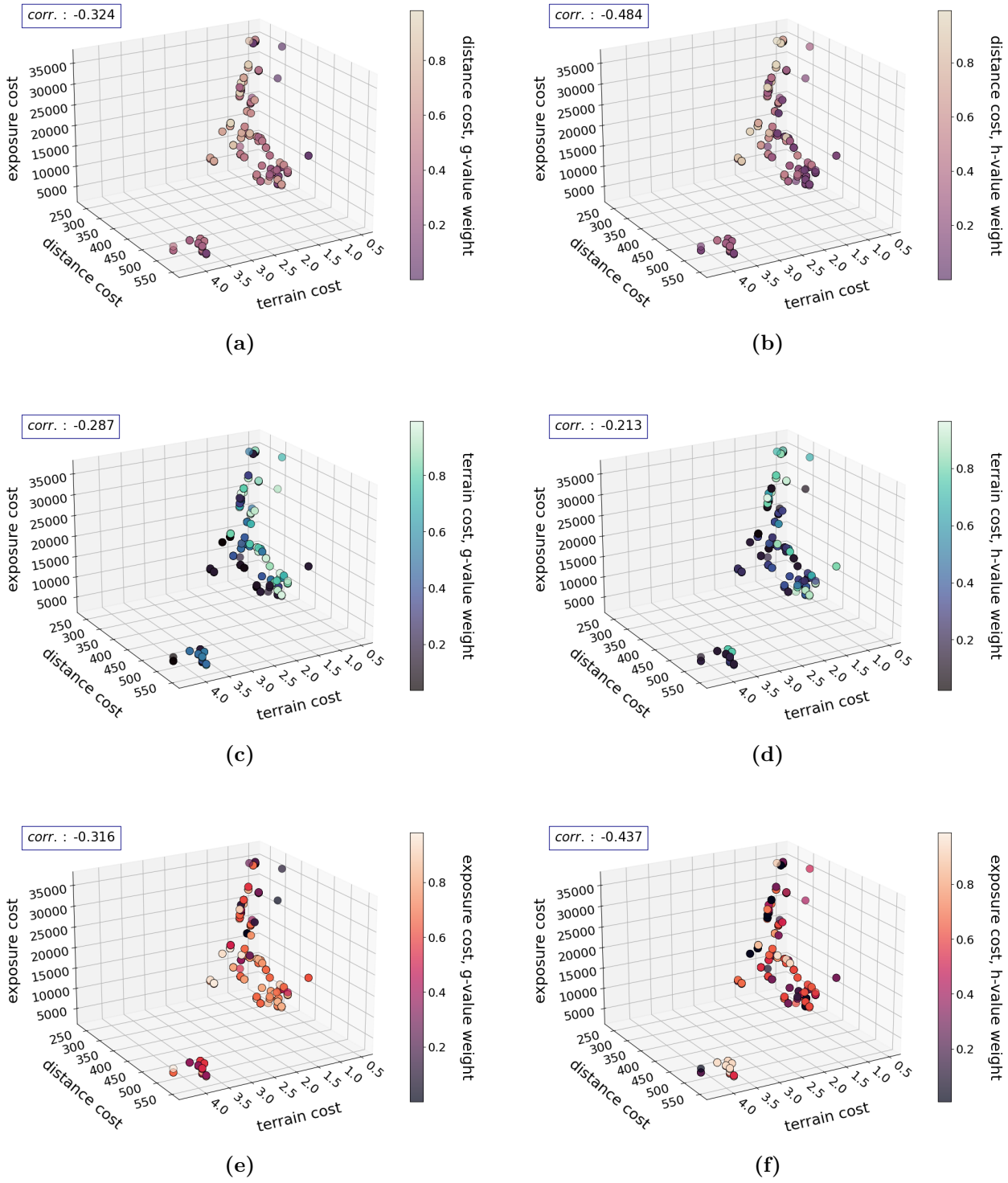


Figure 4.15: Complete population histories from run of MOEA shaded by distance cost weights (top, (a) and (b)), terrain cost weights (middle, (c) and (d)), and exposure cost weights (bottom, (e) and (f)). Pearson correlations between the weight parameter and the relevant objective value are provided for each plot. This output is from run ***B-3***.

of environmental context-independence. While this would require much larger statistical analysis to confirm, if correct it would be a highly valuable trait; re-running optimization algorithms to find the relevant weights for different environments would be unnecessary.

Convergence & Complexity

With the goal of more comprehensively describing the methods developed, the convergence of the optimization and the complexity of the FPA* algorithm itself are evaluated.

A plot of the convergence behavior of NSGA-II in finding optimal parameterizations for FPA* is shown in Fig. 4.17, with the gain in objective space hypervolume being used for quantification. It is observed that major gains are mostly confined to the first 10 generations of optimization with limited improvements throughout the rest of the process.

The complexity figure of merit (FoM) is defined with the equation

$$C_{FoM} = \frac{n_e}{l_{path}} \tag{4.1}$$

for a number of nodes n_e expanded by FPA* to the length of the optimal path found l_{path}. This essentially measures the efficiency of the algorithm in solving for the target optimal path; a perfectly efficient algorithm will always have a C_{FoM} equal to 1 with this formulation. A plot of the distribution of this FoM for 900 optimal paths is shown in Fig. 4.18. It is observed that the majority of paths are solved with a FoM between 1 and 10, roughly with equivalent probability. An analysis of Pearson correlation between hyper-descriptors as well as optimal path FPA* objective weights with the complexity FoM reveal no significant correlations.

4.4 Measurement Update Sensitivity

Another problem analyzed within the context of the framework described in Section 4.1 is that of determining the optimal sensor node measurement update sensitivity. Put more specifically, at what level of disparity between radioactivity rate ground truth and presently known radioactivity measurement should a new update be transmitted to the fusion center? It should be noted that this definition of sensitivity is not related to the sensitivity based

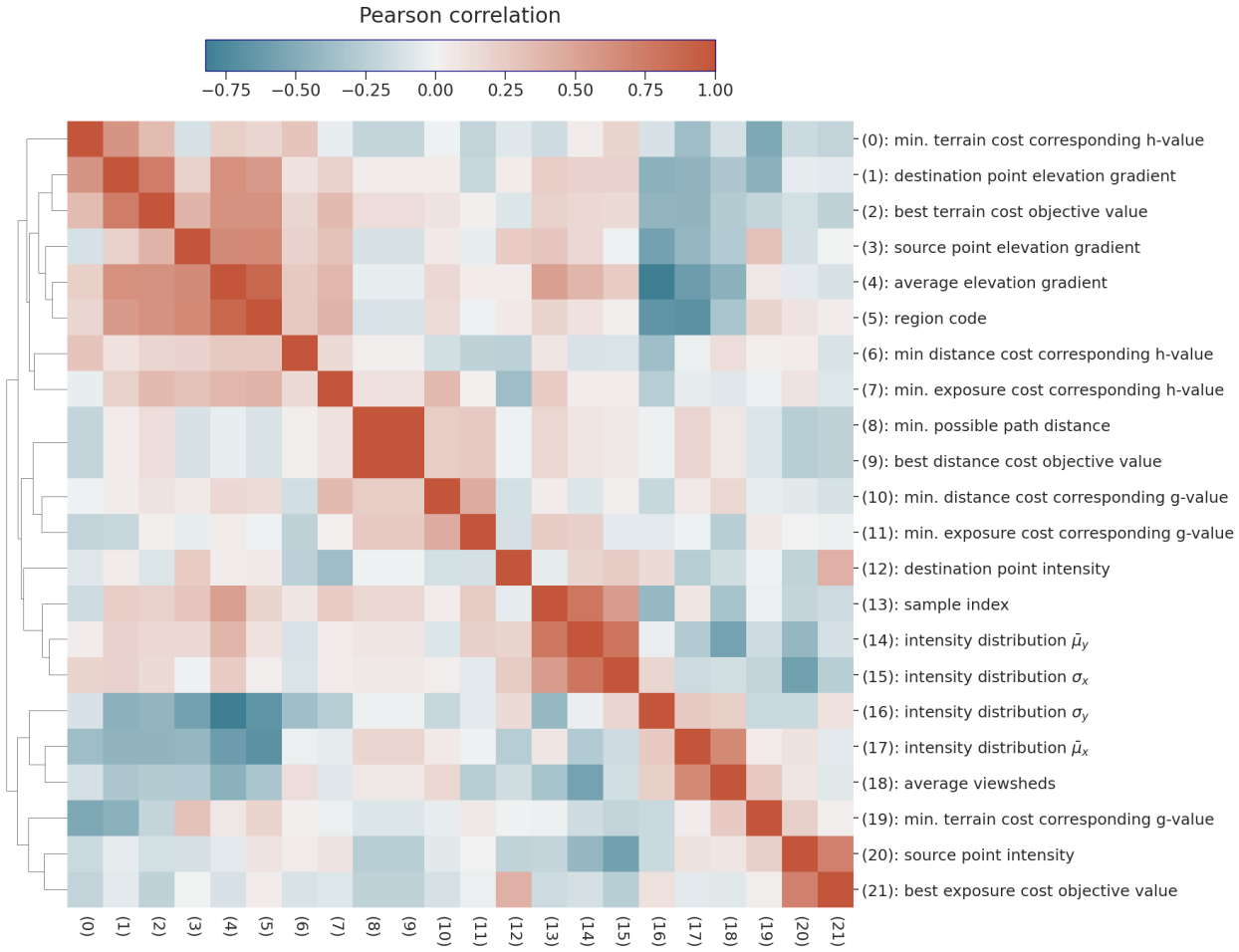


Figure 4.16: Plot of Pearson correlation between hyper-descriptor values and optimized parameter and objective result values. Best objective values are taken from the final population and parameter values used are those corresponding to that objective.

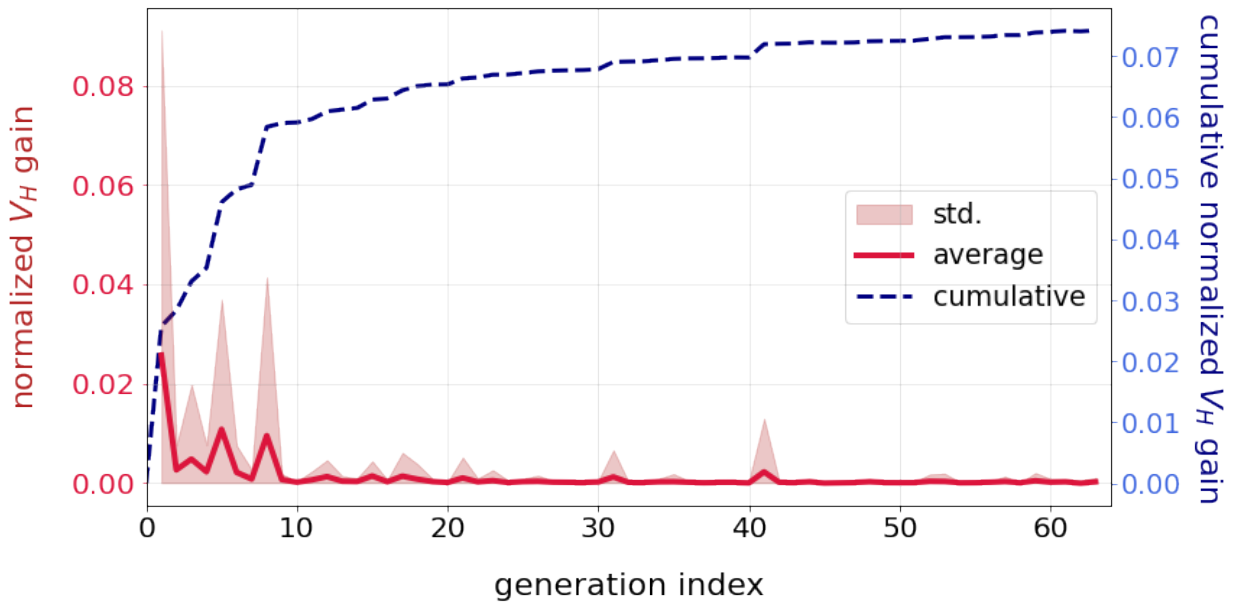


Figure 4.17: Convergence behavior of FPA* parameter optimization using NSGA-II. Quantified using objective space hypervolume gain. Average and standard deviation reported for optimization on the 25 distinct combinations of environmental hyper-descriptors.

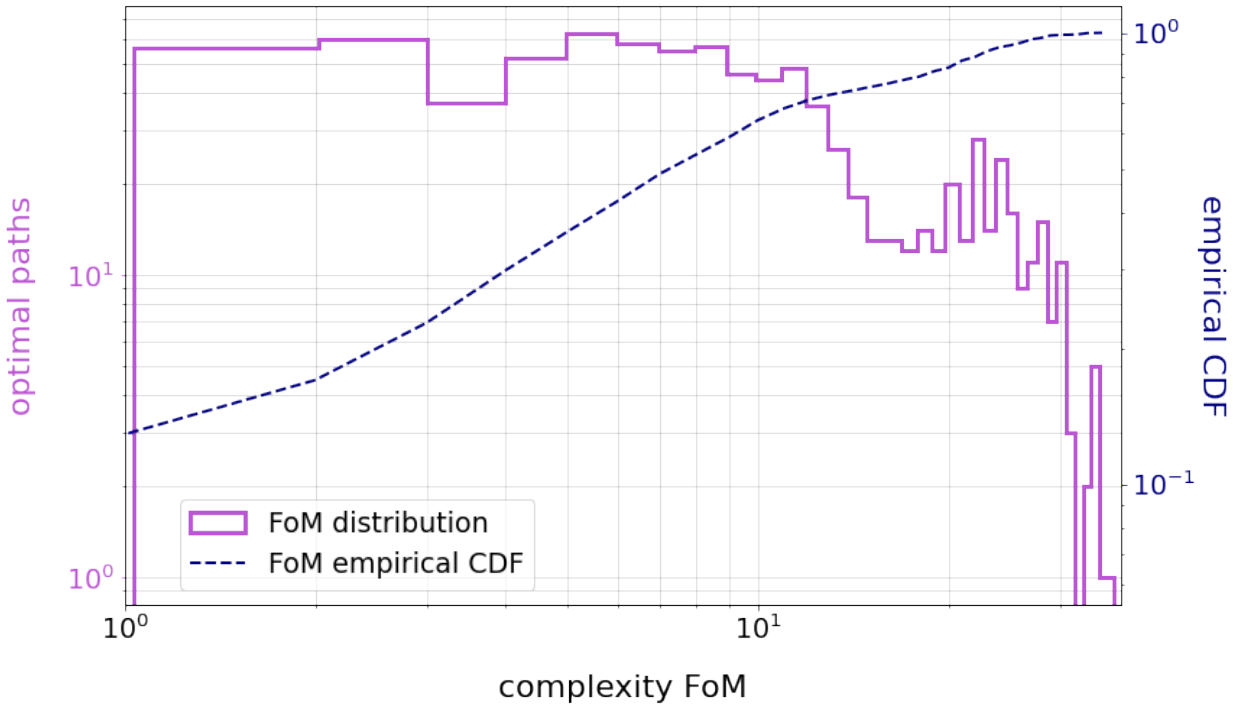


Figure 4.18: Distribution of FPA* complexity FoM for 900 optimal path parameterizations; empirical CDF also shown.

on topographical variation discussed in Section 4.2. This problem is fundamentally about accounting for the power usage inherent in transmitting data in this applied context. Efforts to minimize transmission while maintaining reasonable measurement accuracy will allow the system to stay functional after being deployed for longer periods of time.

The data examined in this problem are time series samples of radioactivity at a particular 3D coordinate position from a plume passing through an area of topography. The larger temporal plume movement dataset is generated by atmospheric simulations using GRASP [120] in a similar manner to the deposition data which was used in the problems described in Sections 4.2 and 4.3. The time series dataset used contains 312 equally-spaced discrete points of data. Many other samples could be theoretically tested from the overall plume dataset by varying the coordinate location of the measurement. The sample evaluated here is selected based on its high degree of variation, and it is therefore believed to represent a sufficiently challenging datastream.

4.4.1 Drift Detection

A concept drift detection (DD) algorithm is used as the solution to this problem as it is fundamentally a question of datastream analysis. The total area between the curve created by the current measurement over time and that representative of the ground truth is used to quantify measurement error; an example is shown in Fig. 4.19. The DD method as implemented uses a sliding window of dynamic size, starting large and shrinking over time to ensure that stale data is not preserved forever. The parameters become the minimum window size, the maximum window size, and the anomaly N-sigma, which is a control of the amount by which a measurement needs to differ from the mean of the window in order for it to be judged an anomaly, an event which triggers an update. As datastream points are added to the window, the mean and standard deviation of the measurements are continuously updated. If the window reaches its default maximum size, subsequent data points being added will require the removal of the two oldest data points. One is removed simply to make room for the newest, while another is removed in order to reduce the total size of the window to reflect a decremented maximum size. This is done until the window reaches a minimum size. At any point that the window contains data points equal to or greater than

the minimum window size, anomalies are detectable. In the case of an anomaly detection, the window is emptied of all previous measurements except the newest one (the anomaly), and the maximum window size is reset to the default value. Anomalies are detected according to

$$\begin{aligned} q &< \bar{v}_w - (n_{\sigma,a}\,\sigma_w) \\ q &> \bar{v}_w + (n_{\sigma,a}\,\sigma_w) \end{aligned} \tag{4.2}$$

for a new data point value to query q, the current window mean $\bar{v}_w$, the current window standard deviation σ_w, and the anomaly N-sigma parameter $n_{\sigma,a}$. If a value q satisfies either of these inequalities, an anomaly has been found and the procedure outlined above commences.

In this evaluation, a discrete parameter space is used. The domain of the anomaly N-sigma parameter is between 1.0 and 7.0, with steps of 0.25. For the minimum window size parameter, possible values are integers between 1 and 10; the maximum window size is similarly parameterized with integers in the range from 11 to 50.

4.4.2 Objectives

The MO modeling of this problem is somewhat more simple than the problems investigated in Sections 4.2 and 4.3. Here, there are 3 input parameters (detailed in the previous section) and 2 distinct objectives of interest. The first objective is the total measurement error accumulated over time; this is calculated as the cumulative discrepancy between the ground truth plume intensity and the current measurement of the intensity. The objective value will be dependent on the variation in the plume intensity throughout the temporal regime as well as the quantity of measurements taken to update the current known value. This leads to the second objective, which is simply the quantity of measurement updates. As described above, the goal is to limit the number of updates as much as possible in order to prevent excess power consumption in a realized sensor network. This is obviously in conflict with the first objective, motivating the MO formulation.

4.4.3 Optimization & Results

The same implementation of MOEA used in Section 4.2.1 is applied for this problem to find best solutions and to approximate a Pareto front. The optimization procedure is run for a total of 196 generations, with a population and archive size of 24 apiece.

A visualization of the results of the MO optimization are shown in Fig. 4.20. As the objective space of this problem is only two-dimensional, the Pareto front structure is easy to discern from the archive points. As expected, there is a downward trend in the cumulative measurement error with increased quantity of measurement updates; it is intuitive that more permitted updates should improve ability to align the measurement with the signal.

A feature of interest in this Pareto front is that increasing the measurement update value between roughly 30 to 60 is not reflected with notable improvements to the measurement error objective value. However, improvements seem to continue again with measurement values greater than 60. Based on this structure, it seems likely that 60 measurement updates is a large enough quantity that the DD algorithm can begin to overfit on the particular datastream sample evaluated.

Parameter Shading

As in Sections 4.2.2 and 4.3.6, population histories are shaded by input parameters to provide additional contextual understanding to their distributions; these are exhibited in Fig. 4.21. Continuing from the above discussion of potential overfitting, further evidence for this conjecture may be observed with the shading of the solution distribution by the minimum window size parameter in Fig. 4.21b. The right tail of the distribution is almost entirely comprised of points with very low minimum window size values, which could help to allow the DD algorithm to overfit on the sample. This is because a very small value of this parameter essentially lets the algorithm create an inflection at will with low statistical confidence. Additionally, there are other details supporting this overfitting hypothesis in the distribution shaded by the anomaly N-sigma parameter in Fig. 4.21a. It is observable the optimal points in the right tail of the distribution are those with very low values for this parameter, which would make on-demand inflection even more accessible for the DD

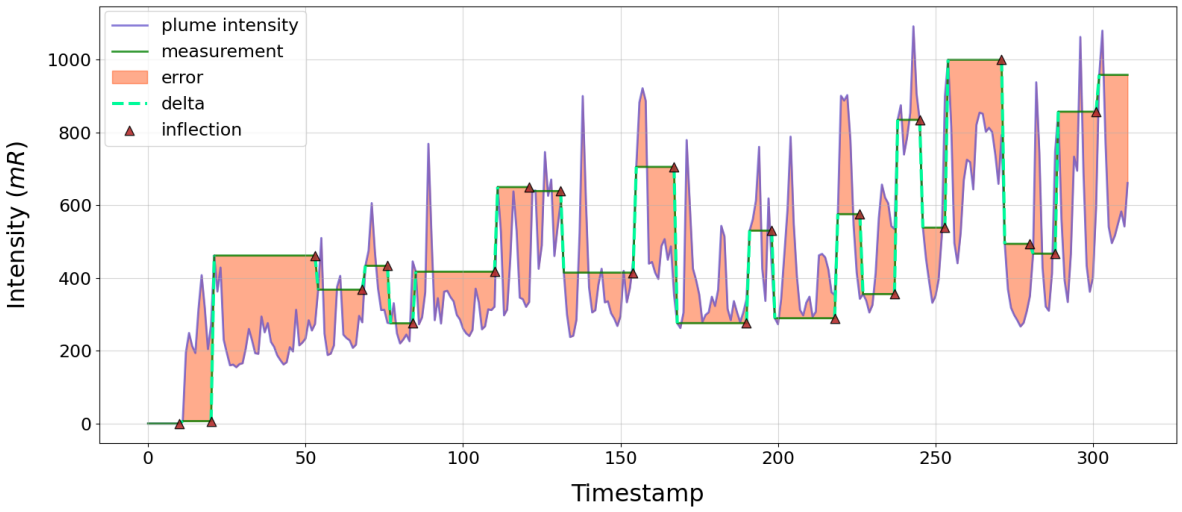


Figure 4.19: Sample of plume intensity over time compared with the current measurement; total area between curves used as one objective in multi-objective formulation. Updates (inflections) determined by drift detection algorithm.

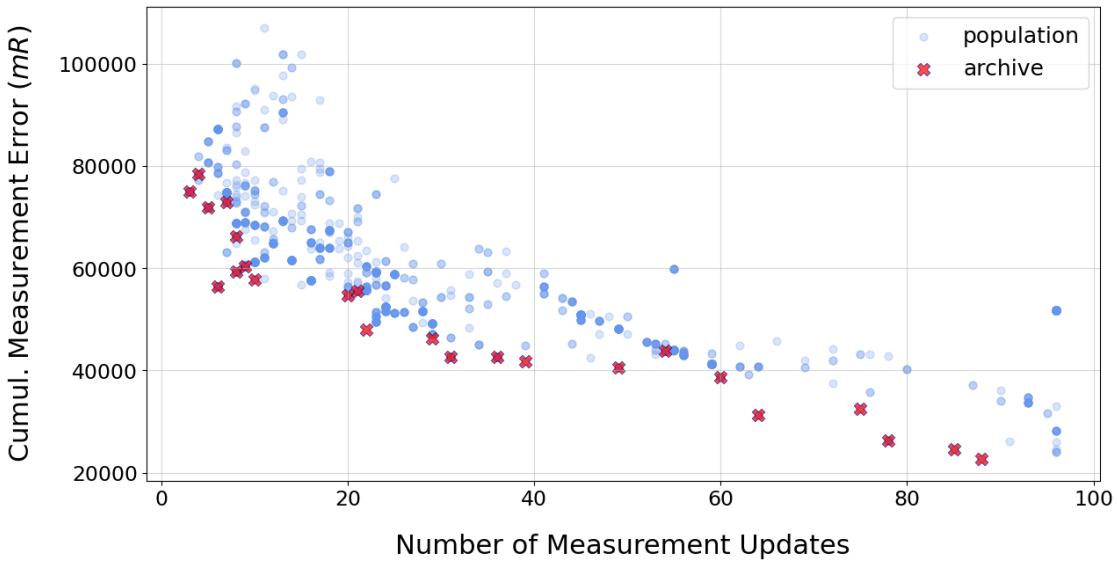


Figure 4.20: Sample results in objective space of MOEA in applied to measurement update sensitivity problem. Generational histories of population (blue) and final archive (red) of solutions are shown here.

algorithm. Aside from this, the cluster of high anomaly N-sigma valued solutions on the far left of the distribution demands attention. That the Pareto optimal candidates with the minimal measurement update quantity values hold very high values for this parameter would seem to indicate that high thresholds for inflection are key to configuring DD algorithms for low-power applications.

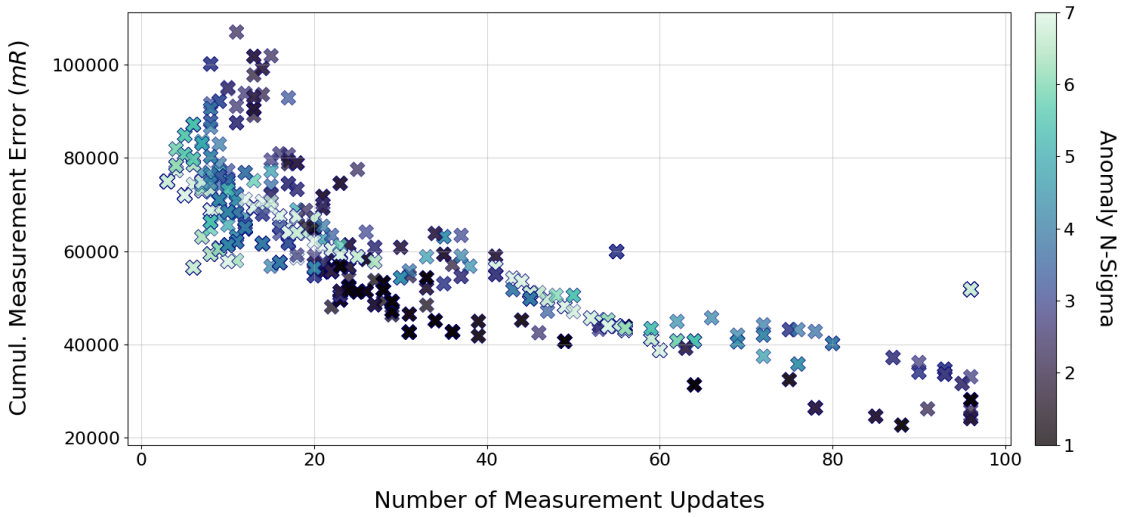
4.5 Multi-Robot Exploration

Work in this case study is aimed at the exploration and mapping of an unknown area of environment for distributed radioactive contamination with a team of multiple autonomous land-based robots. While this section almost entirely focuses on semi-abstract computational modeling of solutions to this problem, it should be noted that this is bounded by the constraints of a real-world physical robot system. These robots are equipped with NaI scintillation radiation detectors and readout electronics, GPS units to provide positioning information, WLAN router modules to allow for intercommunication and coordination across the robot team, and other sensors as well. An image of the physical robots during a test deployment at Technical Area 49 (TA-49) at Los Alamos National Laboratory is shown in Fig. 4.22.

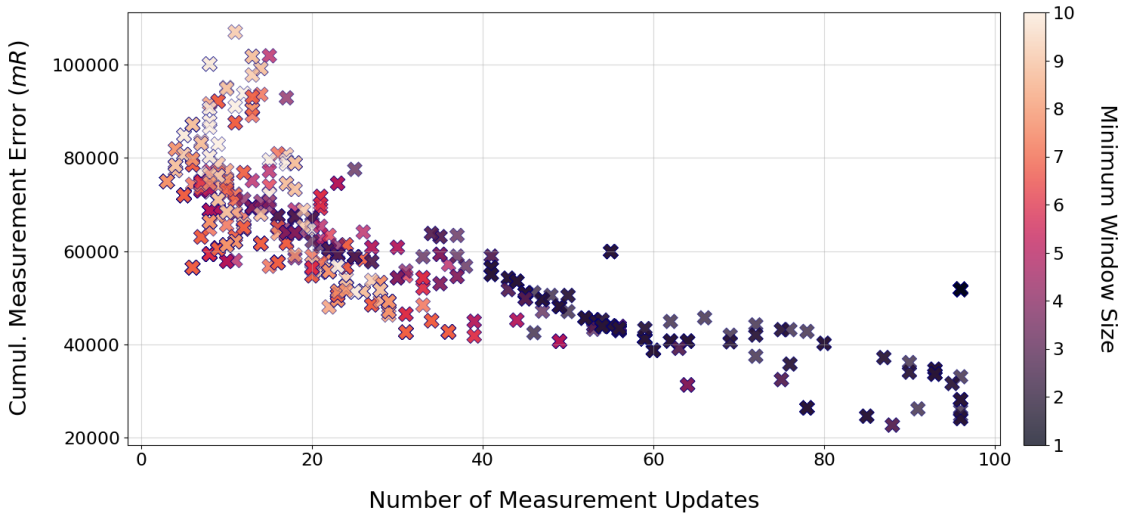
The previous algorithmic method used for this application is described by Shin et al. in [51] and discussed in this text previously in Section 2.1.1. This approach relies on assessing differences the predicted uncertainties to guide robot trajectories. These prediction-difference maps (PDMs) are subjected to Voronoi partitioning with a quantity of cells matching the number of robots used in the survey. The robots are then directed to move with a vector pointing to the resultant centroid of their respective cells. The approach detailed in this section is distinct from the PDM method in that it emphasizes an explicitly MO interpretation of the problem.

4.5.1 Multi-Objective Methodology & Optimization

In the practical conception of this problem, three distinct objectives are identified. These are the minimization of the time of system deployment, minimization of the remaining combined



(a) Histories of population shaded by anomaly N-sigma parameter, which controls the deviation from mean that will trigger a measurement update. Higher values indicate less sensitivity to fluctuating data.



(b) Shading by minimum window size parameter, which limits the lower bound of data accumulated before drift detection is allowable for the datastream.

Figure 4.21: Optimization results presented in Fig. 4.20 shaded according to input parameter values.



Figure 4.22: Physical robots deployed for data collection experiment in Nuclear Environmental Site Area 2 at TA-49 on August 24, 2022. Image courtesy of Raji [128].

uncertainty[2] in the area, and the maximization of the accuracy of the radiological intensity reconstruction with respect to ground truth. It is hypothesized that both of the latter two will innately compete with the time objective; it is natural to expect that given a greater length of time to navigate the environment and take measurements, better reconstructions and less uncertainty will follow. However, the precise nature of the relationship between the uncertainty and the reconstruction performance is less clear to imagine. It does seem likely that the two will be positively correlated (with respect to optimality), although not trivially so.

To optimize these objective values, a novel algorithmic method was developed based on PSO for multiple objectives. This approach requires special consideration of the application, since the particles comprising the swarm in this case are the robots doing the exploration and mapping. As described in Section 2.2, Pugh and Martinoli in [92] investigate the adaptation of PSO for real-world multi-robot applications, wherein the abstract particles are realized as physical robots with all of the associated constraints. In that conception as well as for the work described in this section, the abstract search space is replaced with the physical environment. Velocities towards found optima are realized as well, with limitations factoring in due to physical constraints on agent speeds. The complexity of the modeling is taken one step further in this application via the integration of MO considerations; the novel methodology is termed Multi-Robot Multi-Objective Particle Swarm Optimization (MR-MOPSO).

The MR-MOPSO approach requires many parameters, both for initialization and temporal behavior. This is the same as is the case with an unmodified implementation of PSO, only with this methodology there are many more due to its MO nature. The most impactful of these are the initial accelerations and the acceleration rates of change pertaining to each objective value. Faced with this large parameter space with no intuitive process to select values, another method is required to find optimal values for these parameters. This is an MO problem as well; by applying a MOEA and treating the MR-MOPSO as the black

[2]The combined uncertainty is a sum of aleotoric uncertainty from measurements and epistemic uncertainty from GPR reconstruction.

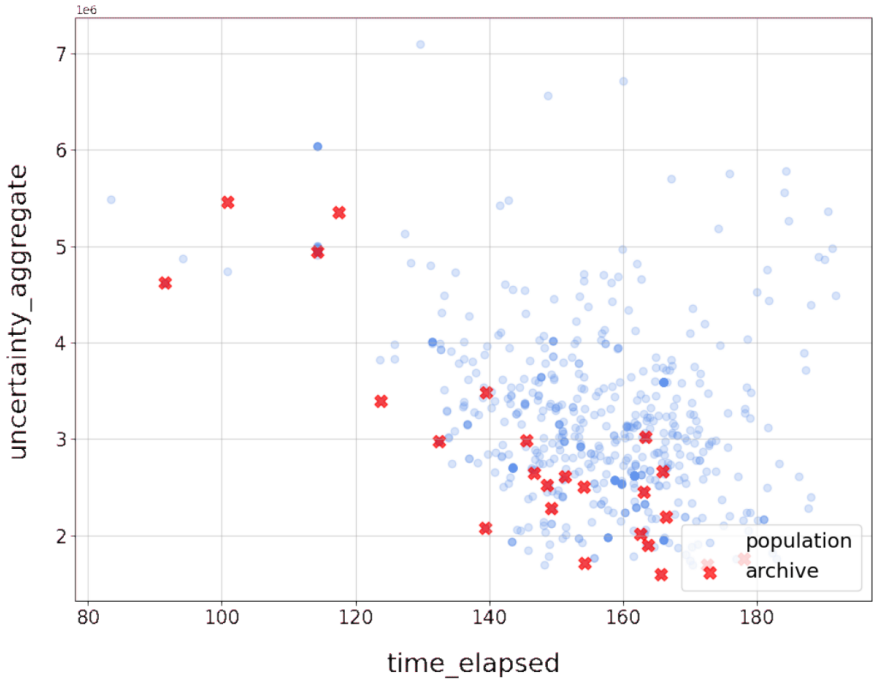
box evaluator algorithm, a suite of meta-solutions to the parameterization problem may be found.

4.5.2 Results

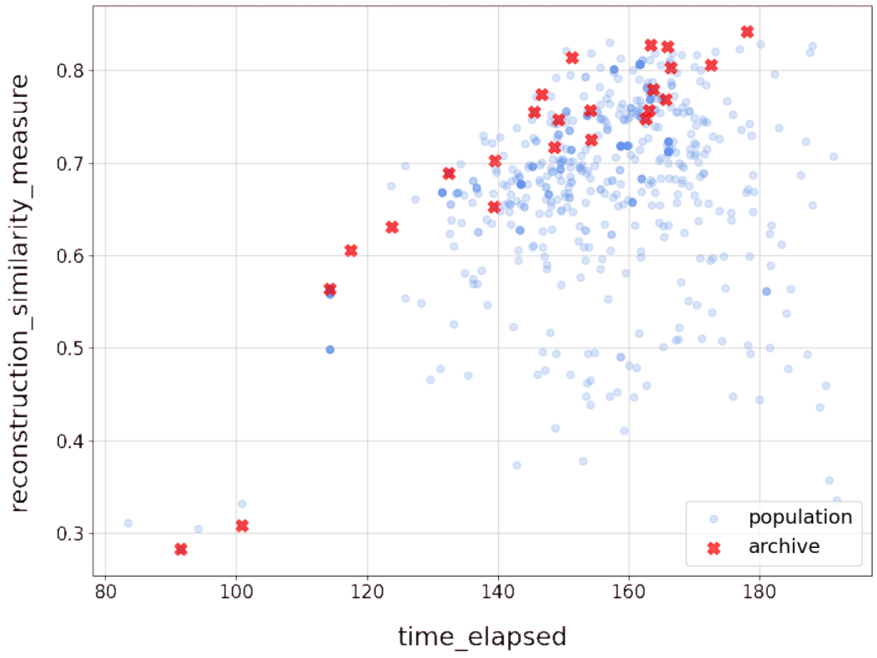
The same MOEA as that used in the problems presented in Sections 4.2 and 4.4 is utilized here for the MOO procedure. An estimation of the problem's Pareto front found by application of this MOEA is displayed in Fig. 4.23 by way of two 2D projections from the 3D objective space. As with the results of the node placement MO optimization in Section 4.2.2, the higher dimensional objective space makes easy visualization of a Pareto front curve somewhat difficult.

Immediately observable in the population distributions are the negative relationships (with respect to optimality) of both the aggregate uncertainty and reconstruction similarity objectives with the time elapsed in system deployment. This is very much in line with intuitive expectations, but there are also more details of interest in the Pareto curve structures. For example, in Fig. 4.23a, major reductions in combined uncertainty come after deployment times of about 120 steps. This type of distinct inflection in the Pareto front has been described in literature as a "knee" [129], the identification of which is crucial for the selection from among an array of non-dominated solutions by a DM. They represent points with very large marginal gains in one objective for relatively small sacrifices in another, or vice versa. In this problem's context, a highly time-constrained deployment of the mapping system may find it very consequential to know that major reductions in uncertainty of the environment can be achieved by fielding the system for about two minutes. With regards to the reconstruction similarity objective in Fig. 4.23b, the knee is observable at a slightly shorter time elapsed.

Granular examination of the optimal parameterizations identified is also possible. Fig. 4.24 displays visualizations of the results of a MR-MOPSO exploration parameterized according to one of the optimal members of the archive shown in Fig. 4.23. With this solution, a longer time for system deployment is allowed to improve performance in both the uncertainty and reconstruction objectives; the temporal relationship is demonstrated in Fig. 4.24d. The sharp spike in aggregate uncertainty around the beginning of the deployment



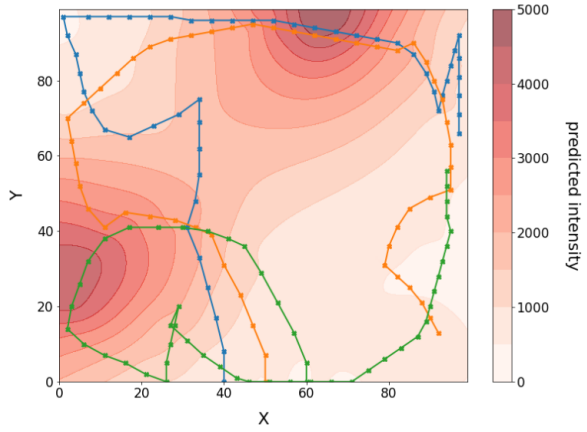
(a) Time against aggregate uncertainty.



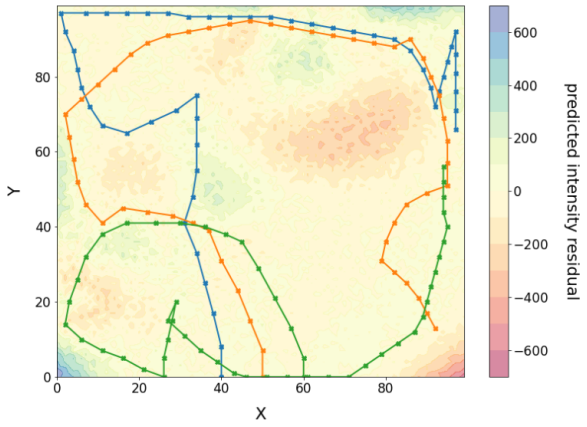
(b) Time against reconstruction similarity measure.

Figure 4.23: Approximated Pareto front found via MOEA optimization for MR-MOPSO robot mapping with GPR reconstruction every 5 steps in a 100-by-100 discretized grid. Two-dimensional projections from three-dimensional objective space shown here. Figures courtesy of Raji [128].

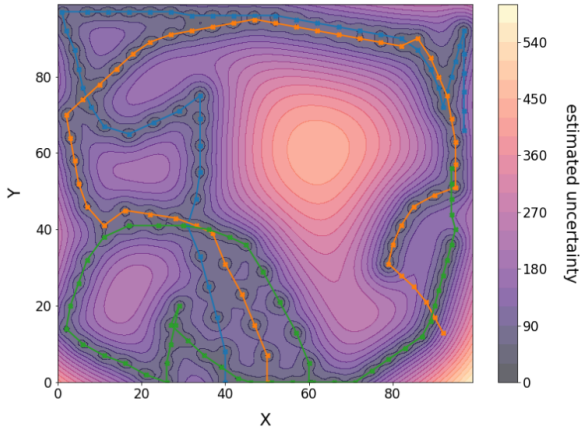
is due to the system becoming aware of the extent of the environment and therefore how much of it is unknown. The paths taken by the three robots during deployment can be observed in Figs. 4.24a, 4.24b, and 4.24c. During the first five steps prior to the first GPR spatial reconstruction, the paths are somewhat convergent towards the nearby source term hotspot. However, following the GPR update, there is rapid divergence in path as the algorithm motivates the robots to explore more of the environment faster. Both of the radiological source term hotspots are quickly found by the robots and thoroughly measured, after which remaining areas of the environment are measured to reduce uncertainty. The success of the method in reconstructing the radiological intensity of the area and cutting down the combined spatial uncertainty can be seen in the relatively flat and minimal value distributions of Figs. 4.24b and 4.24c, respectively.



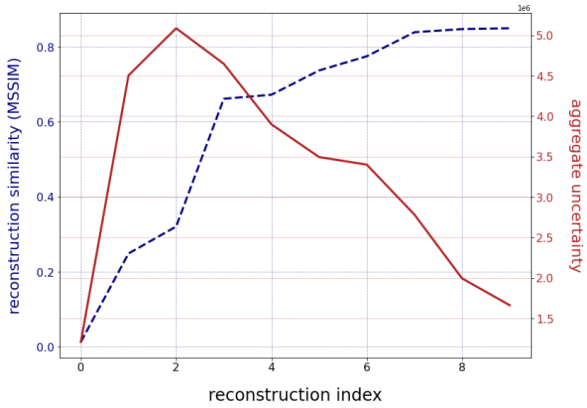
(a) Reconstruction of radioactivity intensity throughout region.



(b) Residual of reconstructed radioactivity.



(c) Resultant combined uncertainty field.



(d) Uncertainty and reconstruction similarity over time.

Figure 4.24: Simulated MR-MOPSO robot mapping results with GPR reconstruction every 5 steps in a 100-by-100 discretized grid, with three robot paths shown. Figures courtesy of Raji [128].

Chapter 5

Urban Environments

5.1 Detector Network Modeling Considerations

5.1.1 Adversarial Threats in Urban Areas

Distributed networks of radiation detectors have been studied for deployment in urban settings to identify potential radiological threats [20, 62, 18]. As visible, static systems, these networks may become subject to attack by way of intentional evasion. Although the likelihood of attack in such a manner is most probably low, the prospect of damage to life in densely-populated city areas by RDD or other smuggled nuclear material-based threat is enough to warrant preemptive risk analysis and mitigation. This section describes a case study in assessment of such threats by modeling intelligent adversarial behavior and possible network-side responses with consideration for multiple objectives on each front. As is mentioned in Section 2.1.2, the work of this chapter is a contribution to the more broad investigation of the PANDA collaboration described in [18].

For this investigation, a conception of adversarial behavior is developed that assumes intentional avoidance of the network as the means of attack. In the simplest terms, the model of the adversary aims to minimize total path length and "sensor cost." This sensor cost is a measure that approximates a quantification of how observable an agent is by the deployed sensor network while it is following its proposed path. Key components of this analysis are the region and route data used, detector placement strategies utilized to construct the

network, and path planning algorithm used to build a path for the agent to navigate through the region while remaining undetected.

5.1.2 Assumptions & Approximations

The work described in this chapter makes a number of assumptions as to the characteristics of the sensor network, mostly for the purposes of simplification of the modeling required. These can be grouped into two categories, with one relating to the radiation detector response and the other relating to details of the urban environment.

In this first group, the radiation detectors of the sensor network are modeled as 2-by-4-by-16" NaI scintillators with intrinsic efficiency of 75% at 662 keV and with isotropic response. The radioactive source carried is modeled as a 100 μCi source of ^{137}Cs. A detection limit at 100 m is assumed; this is used to compute the detection fractions presented in Section 5.5.1 later in this chapter. Given the above, the count rate at a distance of 1 meter is approximately $6.778(10^3)$ cps and that at 100 meters is about 0.2519 cps (using $\mu_{air} = 0.01$ m^{-1}). Additionally, it is assumed that source carriers transit nodes on the road graph in 1 second each. These assumptions, although fairly straightforward, should be contextualized within the goals of this investigation. It should first be noted that a number of the key takeaways from this study will be qualitative; for example, answers to the question of whether there are general principles that make an urban sensor network deployment more resistant to avoidance by adversaries. Second, it is argued that providing greater advantage to the sensors' abilities to detect adversaries (in the form of large detection limits) will mostly function to aid in quantitative evaluation of different network configurations. The algorithm used to model intelligent adversarial path planning (detailed in Section 5.4.1) is quite powerful in this applied context, and can generate paths that completely avoid deployed detector networks due to the inherent sparsity of realistic network configurations. However, comparatively little information is gleaned from path evaluations wherein the network is completely avoided; this is demonstrated later in this chapter in Section 5.5.1. By giving some advantage to the detector networks with these assumptions, fewer path evaluations are needed to provide adequate statistics in result data.

As concerns the urban surroundings of the sensor network, the effects of occlusion are ignored. In contrast to the rationale of the aforementioned assumptions, it was considered that – if modeled – the occlusion by buildings and other structures in the urban environment may provide a more favorable outlook for adversarial agents. However – and as alluded to above – initial results absent this modeling indicated that sensor networks performed very poorly in counteracting adversarial avoidance. It was therefore reasoned that in this context the occlusion effect may be less crucial to the qualitative results and seems more likely to be one affecting quantitative scale. This same logic can also be applied as justification of another simplification, specifically that of flattening the environment to a 2D plane. Intuition suggests that considering variation in height throughout the region would mainly serve to make more difficult the detection of source carriers.

5.2 Data Sources

As alluded to above, a major portion of this work relies on digital models of the regional road network of the area under study as well as vehicle route data. In this analysis, the road network data takes the form of a graph; the nodes (vertices) of this graph are used to inform locations for potential placement of network nodes, while the edges of this graph provide information as to possible paths through the city.

5.2.1 Road Network

The PyUDM framework (which builds upon the Urban Deployment Model described by Abgrall and Cooper in [130]) is utilized to obtain graph representations of real-world road networks from OpenStreetMap (OSM, [131]). In this study, a region of Chicago, IL approximately 6.1-by-8.5 km^2 centered on coordinates of approximately -87.650, 41.875 is modeled. The coordinates of the bounding box defining the area are given in Table 5.1. In this area, the road graph from OSM contains a total of 33,909 nodes and 61,905 edges.

This area contains all of the downtown area (primarily the Loop community area) and a large portion of the surrounding community areas (CAs), including Near South Side, Lower West Side, Near West Side, Near North Side, and West Town. A visualization of this area

with roads highlighted is shown in Fig. 5.1. Delineation between access grades of the roads modeled by the graph edges becomes an important detail to track what segments are eligible for detector placement. The reasons for this are contextual; detector nodes are generally only able to be positioned on surface streets due to infrastructure requirements.

5.2.2 Route Data

Initially, anonymized historical rideshare trip data were sampled from to provide an analogue for realistic routes within Chicago. Investigation used two datasets of 2500 routes each, one drawing from 12-18 August 2022 and another from 15-21 October 2022; both sets were obtained from the Chicago Data Portal [132]. The August set was utilized to inform the detector placement models, while origin-destination pairs were sampled from the October set in order to run path planning evaluation. However, this approach was demonstrated to have clear limitations. It was found that the origin-destination (OD) pairs in the rideshare data tended to skew towards a small number of specific locations, such as intersections adjoining large hotels in the downtown core of the city. While this may be easily understood given context (non-residents will likely use rideshare at a much higher rate), intuition suggests the question of whether the rideshare data are sufficiently representative of actual trips being taken throughout the city.

A new method was therefore developed; in this approach, route OD pair data is generated by sampling from a probabilistic model. This model is constructed by computing estimates of relative probabilities of route origination and destination to the distinct CAs of Chicago; these estimates are made by comparing vehicle ownership and intra-city commuting data particular to each CA [133]. Vehicle ownership figures are used to create route origination probabilities, whereas information on commuting is utilized for approximation of destination selection given an origin CA. For example, if the data indicates that residents of the Lake View CA mostly are employed in the Loop CA, the route generation model will reflect this with a high probability of choosing a destination within the Loop CA if the Lake View CA has been selected for route origination already.

In modeling specific adversarial routes, it was hypothesized that highly-trafficked civilian destinations could become potential targets. A list of such locations was curated based

Table 5.1: Corners of bounding box of the region of Chicago used for investigation in this Section.

Corner	Longitude	Latitude
Southwest	-87.686819	41.836594
Southeast	-87.686819	41.912782
Northwest	-87.613000	41.836594
Northeast	-87.613000	41.912782

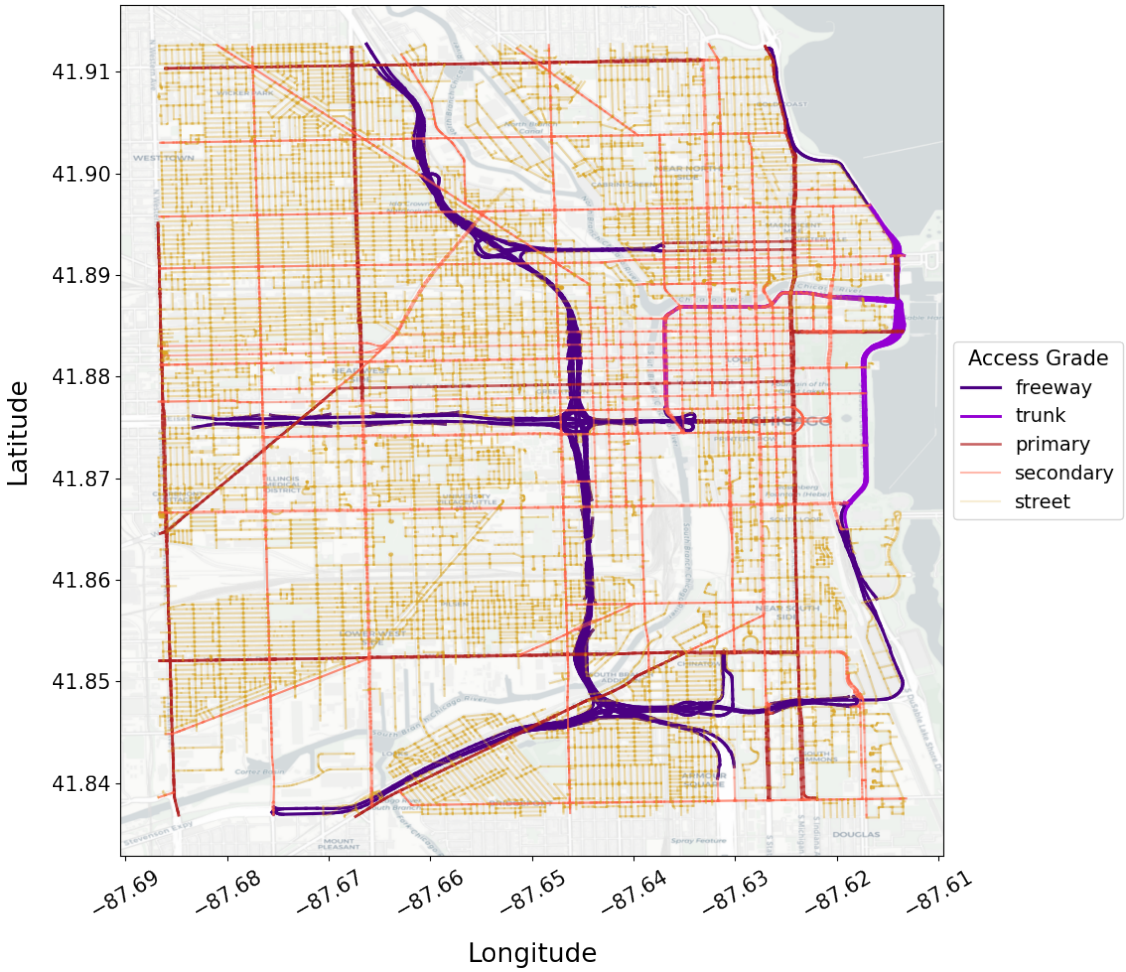


Figure 5.1: Region of Chicago selected for investigation. Edges of graph model are formed by roads, which are colored here according to their access grade.

on a review of a number of travel and tourism website pages detailing popular Chicago attractions [134, 135, 136, 137, 138]. This list is presented in further detail in Appendix D.

For the purposes of detector sensor network evaluation, a total of 8 datasets of routes are constructed. Half of these are dedicated to function as training data while the other four serve as test sets (utilized for evaluation). The characteristics of these sets are detailed in Table 5.2.

5.3 Detector Placement

The study of adversarial avoidance described in this chapter is preceded by the formulation of means to position the nodes of the sensor network within the modeled cityscape. For this purpose, three algorithmic strategies for node placement are evaluated in the course of this work. All of them function in a static capacity, which is to say that they utilize a finite set of route training data to ultimately make decisions as to where to place nodes (which contain the radiation detectors).

The first of these strategies functions by iteratively placing detectors at vertices on the graph until all routes in the training dataset are observed at least once or until the set number of detectors is reached; this method is termed "Strategy A". In implementation, this results in detector positions that provide the greatest overlap in training routes. Since this is an iterative method, the locations with the greatest overlap are selected first. The second method ("Strategy B") is largely similar to the first, with a slight modification. This change involves the introduction of a hyperparameter to allow a degree of repeated placement in proximity to other detectors in areas of very high route density. The motivating idea is to allow for greater network sensitivity in particular areas, such as those with high-profile potential targets of adversaries. Lastly, "Strategy C" iteratively grows clusters on a grid mesh of the urban area. The growth of each of these clusters is started and stopped based on the total route density (based on the training dataset of routes) of the area contained within them. A cluster with collective density greater than the average of all clusters plus one standard deviation no longer is provided with new points, while those below this mark continue to accumulate. The iteration is stopped once the clusters have collectively grown to

encompass a fraction of the totality of the grid mesh points. The maximum intensity point of each cluster is then selected as the eventual detector location.

As may be apparent from description, the detector locations given by each of these algorithmic methods are heavily dependent on the training dataset of routes used. For this reason, a number of different sets of detector positions are generated by using different training datasets of routes. All resultant sets of positions are evaluated to provide a more complete assessment of each of the algorithms. Detector placement for the dynamic methods studied in this chapter requires quite a different approach. Details of this are covered in Section 5.4.2. Following detector placement (regardless of method), a "cost map" is created that quantifies the network sensitivity at each point on the graph. Each vertex is initialized with a cost c_0 proportional to the number of detectors placed there. Neighbors of non-zero cost vertices then are given costs c_1 proportional to c_0 and inversely to the distance between them. In this frame, the cost descriptor is in relation to the penalty incurred by adversarial paths.

5.4 Adversarial Analysis Methodology

5.4.1 Iterative Adversarial Approach

For this work, methods of adversarial path planning were first devised beginning from foundational algorithms. One relatively simple approach was developed with the basis of the A* algorithm [78] and is termed Iterative Induction A* (IIA*). It functions by finding a shortest path between source and destination nodes using A* and iteratively inducing subgraphs not containing any high-cost vertices included in that path. This process is repeated until either a zero-cost path is found, the graph becomes disjoint, or a maximum number of iterations is reached. The output metric of interest can be described with

$$C_{best} \;=\; \min_{d_i \in D} \; f_{path}(G_{d_i},\; v_o,\; v_d) \tag{5.1}$$

wherein the best adversarial path cost C_{best} is equal to the minimum cost of the pathing method f_{path} as a function of the origin vertex v_o, destination vertex v_d, and the subgraph

G_{d_i} at iteration depth d_i. The IIA* approach is summarized in pseudocode in Alg. 4, which is located in Appendix E. This algorithm is a relatively simple but very powerful tool for finding undetected paths, especially so when allowed to run to greater iteration depth. In this application, it is utilized for primarily for the evaluation of detector network positioning, the results of which are presented later in this chapter in Section 5.5.

A sample output of the IIA* algorithm is visualized in Fig. 5.2. The detector network positioning is shown along with the iteration on planned paths which succeed at avoiding detection at greater depth.

5.4.2 Competitive Co-Evolutionary Optimization

Adversarial path planning and detector placement, taken together in this applied context, are problems for which the optimal solutions for each rely on the current solutions of the other. For instance, if path planning strategies are changed, the optimality of previous sensor positions may well degrade. During initial evaluation of the detector placement methods described in Section 5.3, it was hypothesized that a static-frame approach to modeling might not be sufficient for this problem to achieve robust results for the reasons above. Consequently, it was then theorized that a dynamic method could be better suited and provide network configurations more resilient to adversarial evasion.

Optimization Approach Characteristics

As reviewed previously in this text in Section 2.6.1, dynamic optimization techniques can be advantageously used for problems with non-stationary elements. Cues from efforts in this vein are taken in designing the optimization method employed here, along with some ideas from previous competitive optimization works (discussed at greater length in Section 2.6.2). This new approach is termed competitive co-evolutionary optimization (COCO-EVO).

As the name suggests, COCO-EVO uses an evolutionary algorithm as its foundational basis, although its design features a number of important distinctions. The first of these centers around the usage of two distinct groups of individuals represented with two separate populations. In the applied context dealt with in this chapter, one population contains

Table 5.2: Characteristics of route datasets used in urban radiation detection sensor network. In this context, the "size" column corresponds to the number of distinct origin-destination pairs contained in the dataset.

Usage	Size	Targeted Route Percentage
Train	2160	5
Train	2033	20
Train	1916	50
Train	1585	100
Test	150	0
Test	150	5
Test	150	50
Test	150	100

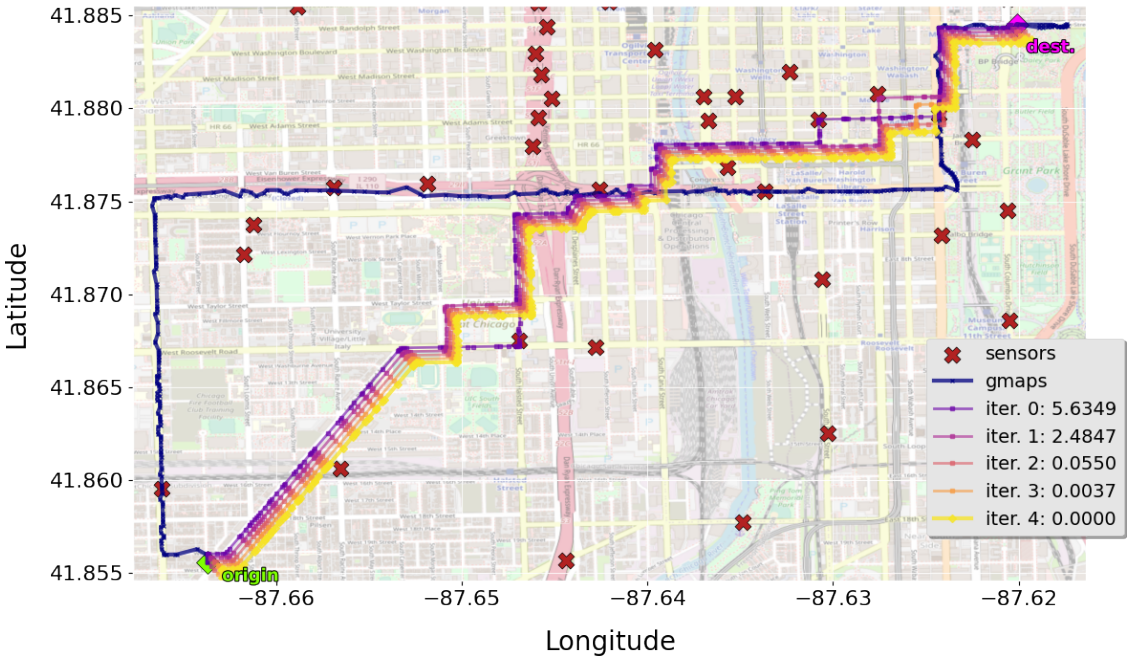


Figure 5.2: Sample IIA* result for 150 sensors placed using Strategy A (as described in Section 5.3). Google Maps recommendation shown in navy. Overlapping routes offset for visibility.

detector network configurations and optimizes positions for nodes of the network (hereafter "inciting population"). The other population contains beliefs as to the locations on the road graph where an agent transiting is likely to be detected (hereafter "respondent population"). Both populations are parameterized with a set of weights for the nodes of the road graph. In the inciting population, the network configuration is created via a greedy algorithm which selects a quantity n nodes with the highest weight value from the graph as positions for which to place n detectors. With the respondent population, graph nodes with high weight values are believed to have relatively high probabilities of being co-located with or nearby to detectors of the network and are therefore avoided by MO path planning evaluations taking the weighted graph as input.

The second primary difference from the typical evolutionary algorithm centers around the evaluation mechanism for individuals. These are not evaluated with some static function; rather, each individual is instead evaluated against another individual from the opposing population. To clarify the process for this application, an example of the procedure can be outlined. An individual "A" in the inciting population contains the information to construct a detector network on the graph. In the evaluation step of COCO-EVO, an OD pair is chosen from one of the training datasets described in Table 5.2, and an individual "B" from the respondent population is selected to inform the adversarial portion of the evaluation. This individual B can be intentionally or randomly selected; in the bulk of the analysis performed with this methodology, a stochastic approach was taken, with higher probabilities assigned to older individuals in order to discourage stagnation. The graph weight parameterization of individual B provides context to the adversarial path planning algorithm; the details of this point are given in Section 5.4.2 which follows this one. The resulting path generated is checked against the network of individual A, and the quantified detection performance becomes the objective value for individual A's adversarial detection score. Evaluation of individuals in the respondent population proceeds similarly, only with the order of selection being reversed.

One more crucial distinction of the COCO-EVO framework lies with the approach to handling dynamics. It should first be noted that two separate types of non-stationary characteristics exist, one innate to the COCO-EVO design and the other arising from the

approach to handling the first with additional relevance to the application. The former stems from the nature of procedure used to evaluate of individuals described above. Since evaluations for either population use functions with changing characteristics, it is not acceptable to evaluate any individual once and hold the objective results static. Instead, all individuals are evaluated at every generation, and objective values are averaged accordingly. However, this solution of averaging leads to a problem; it would seem possible that recent offspring introduced to one of the populations could spuriously displace other historically well-performing individuals by way of a relatively small number of high-performance evaluations. This effect in isolation could perhaps hurt the robustness of solutions identified, but is greatly magnified by the nature of the application evaluations themselves. To be specific, each evaluation considers one path, the start and end nodes of which are sampled from training data as stated above. It is intentional that a wide range of paths will be evaluated for individuals of both populations in order to ensure solutions are not overfitting for a particular case. Yet for low numbers of evaluations, an individual's average objective values will inevitably be skewed based on the few paths it has been exposed to up to that point. To counteract this, a "maturity" mechanism is introduced. Components of this include an "age" attribute that is attached to each individual. New individuals begin at age 0; the age of an individual is incremented by 1 for each generation that it continues to exist within a COCO-EVO optimization procedure. The next key part of the mechanism is a "maturity" threshold, which controls the minimum age at which individuals in COCO-EVO are subject to evolutionary operators. The functionality of this is made more simple by the additional of two "minor" populations to contain individuals that have not yet reached this threshold. The premise is to provide sufficient generational time so that new individuals' objective values will have the chance to acquire better statistical credibility via a greater number of varied evaluations. A diagram of the COCO-EVO optimization model framework is shown in Fig. 5.3 with labeled components of the features detailed above.

Objectives

Individuals are optimized for a set of multiple objectives based on which of the populations they lie in. For those in the inciting population, the pertinent objectives are the maximization

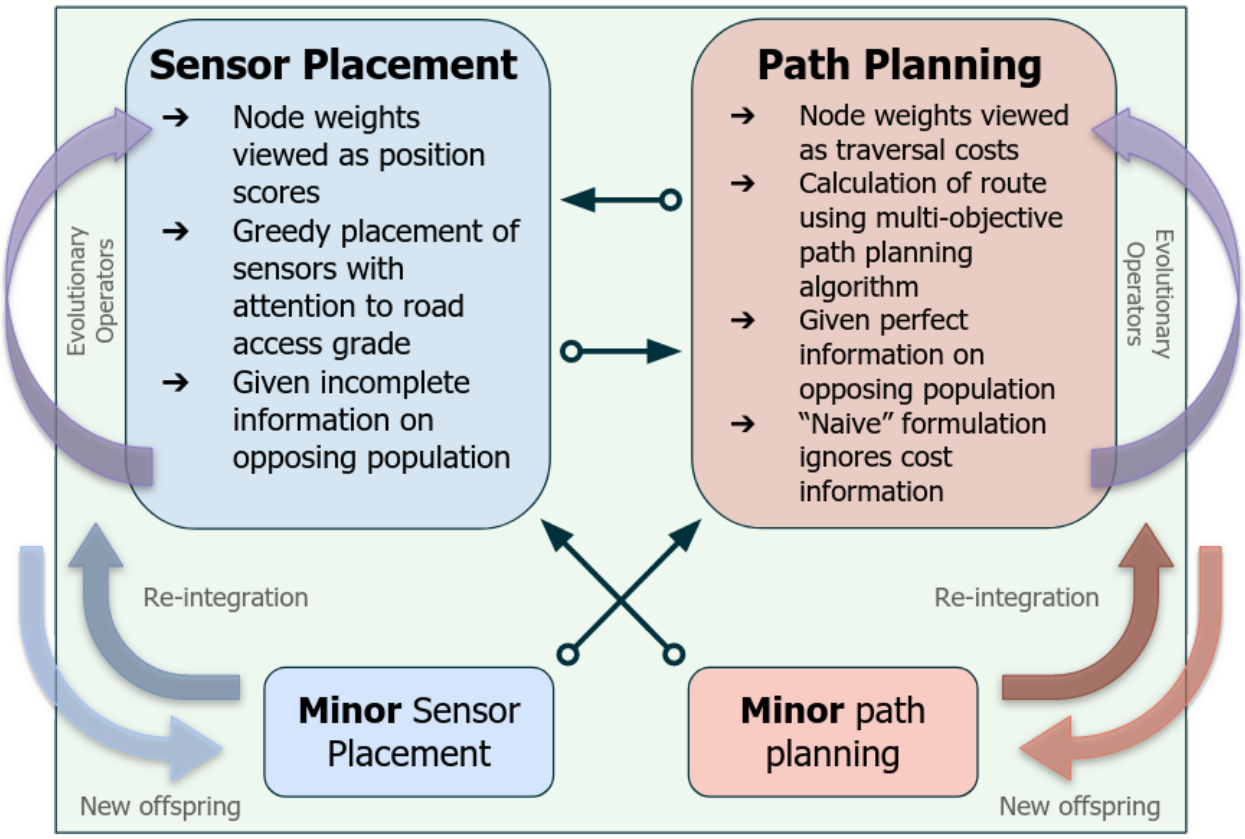


Figure 5.3: Diagram of COCO-EVO optimization framework. Directional arrows represent evaluations of one population (hollow circle) on another (arrow point).

of the network's detection score against naive – or non-adversarially planned – routes, as well as against those routes that are planned by an intelligent adversary seeking to avoid the network. The motivation here is to study whether networks that perform well against adversarial attacks must sacrifice performance with regards to detecting other transits. In the respondent population, objectives are the minimization of path detection by the network along with the minimization of the ratio of the planned route length to the length of the shortest possible route between the nodes of the OD pair used. These two are chosen in order to evaluate increases in necessary route complexity (and therefore length) that may come with avoidance of detection.

Parameterization Decomposition

As described in Section 5.2, the road graph covering the region of Chicago analyzed for this problem is comprised of 33,909 nodes. Weights for these nodes are used as parameters of solution candidates for both the inciting and respondent populations of the COCO-EVO framework as is explained in the previous section. This is a very large quantity of parameters, which is a characteristic that can cause issues with the optimization search as implemented by the evolutionary operators [139]. To ameliorate this, dimensionality reduction is conducted in the parameter space for the evolutionary operation step of each generation. Principle component analysis (PCA, [140]) models are fit using 24 components on a record of the parameterizations observed in the relevant population throughout the optimization process up to that point. Individual parameterizations are transformed into the low-dimensional space for selection, reproduction, and mutation, then subjected to an inverse transformation in order to interact with the evaluation mechanisms. With this approach, the evolutionary operators are able to operate on data with a much more manageable dimensionality; this provides greater confidence that spurious patterns in the parameters will not be inadvertently carried forward with consistency.

Adversarial Path Planning

Two methodologies are modeled for adversarial path planning within the COCO-EVO framework. In the first, perfect information is available to the adversarial agent. In practical

terms, this means that the exact locations of all detectors within the sensor network are known *a priori* with respect to a path being planned. This is implemented using the IIA* algorithm described above in Section 5.4.1. With the second mode, the adversary constructs a belief model of the detector positions over time; this model is updated over the generations of the competitive optimization process in response to the outcome of path evaluations (whether they are detected or not). This methodology uses the FPA* algorithm, which is presented first in Section 4.3.2.

Additional Hyperparameters

In addition to the choice of adversarial path method, there exist a few other hyperparameters that are set before running the COCO-EVO model. Among these are the initialization conditions of individual parameters in both populations, which is varied between a random exponential sampling method and a sampling method based on historical traffic intensity data. Also set *a priori* is the percentage of adversarial routes evaluated to use a specific target in the manner discussed in Section 5.2.2; in the studies conducted here, this percentage is set at either 50 or 100%. It should be noted that this is not a percentage of all planned routes evaluated; paths planned for evaluation of the networks' naive path detection capabilities (as mentioned in previously in this section) do not use the targeted OD pairs.

The total quantity of detectors that are allotted for construction of the network is decided before running the optimization procedure as well. It was originally investigated to use this characteristic as a dual parameter-objective as was done in the problem described in Section 4.2. However, early evaluations with this methodology led to the conclusion that its inclusion in this way would make results too difficult to interpret.

Outputs & Temporal Considerations

As alluded to previously, it is not intuitively clear that a temporally-expressive model in the vein that is attempted with the formulation of COCO-EVO will reach a stable equilibrium consistently. If one regime in either of the two populations becomes dominant, it will drive opposing trends in the competing population. For this reason, it is argued that temporal analysis of recurring patterns in output data will be the main output of value from the

framework. For example, placement solutions returning to common locations throughout optimization progression would suggest positions for detectors that consistently perform well against various adversarial strategies. This is in contrast to the study of singular generation results and/or a focus on the usage of specific stopping conditions.

Detector Positioning for Evaluation

Evaluation of the actual detector network configurations in a quantitative manner requires the export of data from the dynamic realm of the COCO-EVO method into a static frame. This follows from the above discussion on studying temporal outputs as opposed to those of a typical optimization algorithm. The step is necessary for the purposes of comparison to other detector placement methodologies, the process and results of which are detailed in Sec. 5.5.1. In order to accomplish this, two methods of stochastic sampling are utilized.

The first approach draws samples from the aforementioned KDEs of the detector positions over the COCO-EVO generational histories; sampling weight for a given node location is based on the magnitude of the KDE at that node. Deciding network placement in this manner can result in particular positions for detectors that in actuality have never been explicitly included in any observed network configuration previously. This can occur due to inherent smoothness in the KDE, causing relatively high-density localities to include a mix of real historical detector positions and their neighbors. Consequently, it is assumed that this design choice will lead to high variance in the resultant detector network.

With the second approach, only node positions that have been previously been used for placement as a part of detector networks during the competitive optimization generational history are sampled from. In this case, the frequency of detector placement at each location is used as the weight for sampling. While still stochastic in nature, this method is more controlled than the first; it can be assured that all detector positions selected will have some precedent.

5.5 Results

Results of the adversarial analysis can be sorted into two general groups. The first of these contains quantitative evaluations of the performance of each detector network configuration method. In the second group are qualitative results which are found via investigation of overall success at detection as a function of road graph positioning.

5.5.1 Network Configuration Evaluation

Quantitative results of different sensor network configurations' performance against various adversarial threat models are sought to compare their relative efficacy. Specific network configurations are generated via the means described in Sections 5.3 and 5.4.2, with variations of hyperparameter values allowing for a number of distinct networks to be created for each method.

Each configuration is subjected to the IIA* algorithm described in Section 5.4.1 for the test route datasets introduced in Table 5.2. Details of the hyperparameters used for each of these network configurations are provided in Table 5.3.

As previously stated, the primary output metric of the IIA* algorithm is the minimal detection cost able to be found for a path constructed connecting a given OD pair on the graph. This measure is utilized in this section to quantify detector network performance in a few different ways. Summarized results of the distributions of the best detection cost for the above described network configurations are presented in Fig. 5.4. These plots depict outputs of evaluations using the test route dataset with 5% targeted routes on networks comprised of 50 and 100 detectors. The distributions shown are constructed from the minimal costs achieved by IIA* for 150 path evaluations. In this case, distributions with larger values of median, quartiles, and octiles can be said to be better performing. For the same set of networks, the results are visualized in a different manner with Fig. 5.5. In these plots, the networks are sorted by the frequency for which they successfully detect a tested adversarial path, along with the relative degree of strength of those detections according to the gross counts accumulated across the network (computed using source and detector assumptions described in Section 5.1.2). Quantitative plots in the manners above described

Table 5.3: Network configuration characteristics key.

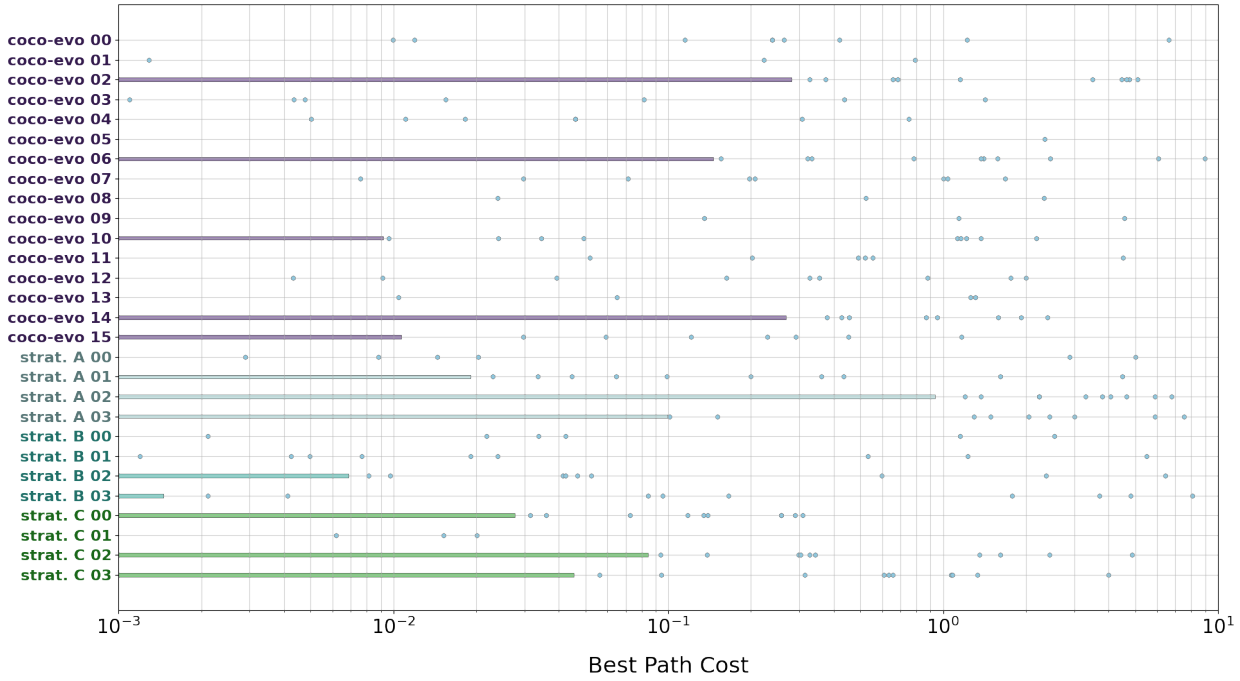
Placement Method	Targeted Training Percentage	Path Method	Initialization	Sampling
COCO-EVO 00	50	FPA	exponential	density
COCO-EVO 01	50	FPA	exponential	KDE
COCO-EVO 02	50	FPA	traffic	density
COCO-EVO 03	50	FPA	traffic	KDE
COCO-EVO 04	50	IIA	exponential	density
COCO-EVO 05	50	IIA	exponential	KDE
COCO-EVO 06	50	IIA	traffic	density
COCO-EVO 07	50	IIA	traffic	KDE
COCO-EVO 08	100	FPA	exponential	density
COCO-EVO 09	100	FPA	exponential	KDE
COCO-EVO 10	100	FPA	traffic	density
COCO-EVO 11	100	FPA	traffic	KDE
COCO-EVO 12	100	IIA	exponential	density
COCO-EVO 13	100	IIA	exponential	KDE
COCO-EVO 14	100	IIA	traffic	density
COCO-EVO 15	100	IIA	traffic	KDE
Strategy A 00	5	N/A	N/A	N/A
Strategy A 01	20	N/A	N/A	N/A
Strategy A 02	50	N/A	N/A	N/A
Strategy A 03	100	N/A	N/A	N/A
Strategy B 00	5	N/A	N/A	N/A
Strategy B 01	20	N/A	N/A	N/A
Strategy B 02	50	N/A	N/A	N/A
Strategy B 03	100	N/A	N/A	N/A
Strategy C 00	5	N/A	N/A	N/A
Strategy C 01	20	N/A	N/A	N/A
Strategy C 02	50	N/A	N/A	N/A
Strategy C 03	100	N/A	N/A	N/A

for all evaluated detector quantities as well as all test route datasets described in Table 5.2 are compiled in Appendix F.

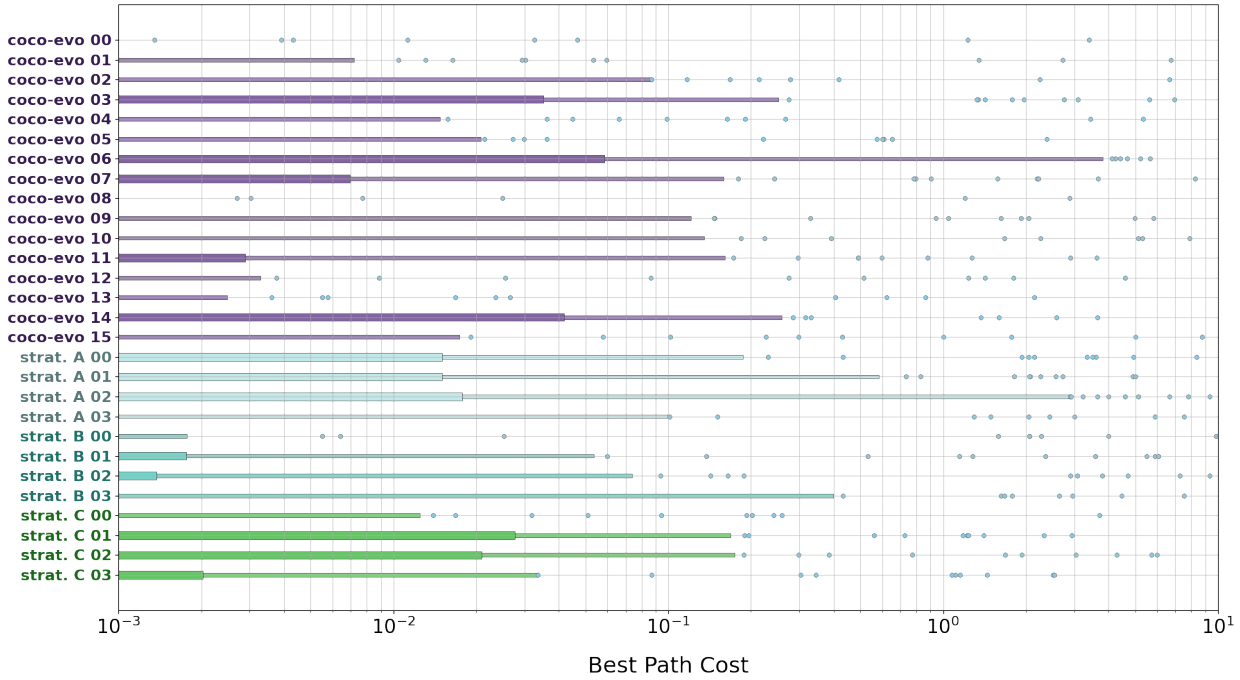
Plots of the best path cost metric distributions in Fig. 5.4 and in Appendix F reveal a number of valuable details. The first of these is the strength of the IIA* method; in evaluations for 50-detector networks or on the non-targeted route dataset, the majority of paths planned go completely undetected. This is reflected by the severe skewness of most of these distributions towards a value of zero. In contrast, largely-nonzero distributions are obtained for the evaluations with large detector-quantity networks and/or evaluations containing a large fraction of targeted routes. With regards to the former characteristic, this makes sense intuitively; more detectors allotted will allow for better coverage of the road graph on the whole. But that there would be a notable improvement in network performance when testing against the targeted route datasets requires more investigation. One rationale could lie with the relatively small number of target locations modeled as compared to the quantities of detectors in networks evaluated. Analysis of this pattern in more detail is continued in the following section.

With a synthesis of the above described results, the performance of particular network configurations can be put into focus. The configuration key in Table 5.3 is for the purpose of dereferencing the labels in Figs. 5.4, 5.5 and in the figures of Appendix F. First of note is the detection performance of networks created using Strategy A; these are observed to be at or near the maximum detection fraction of configurations tested in the majority of evaluations on networks of different quantities and with different test route datasets. This outcome is understandable given the design of the algorithm; as discussed in Section 5.3, Strategy A adopts a positioning scheme that aims to cover all training routes to at least some minimal degree. Yet this last detail is apparent in these results also, as the mean best path detection cost achieved by networks using this strategy tends to be average at best. In essence, this methodology may consistently provide some measure of adversary detection, but it is likely to be a relatively weak signal.

The networks created from the results of COCO-EVO procedures are also of interest. While not usually exhibiting the maximal detection fraction, configurations using this strategy do consistently have the highest mean estimated gross counts for those paths that

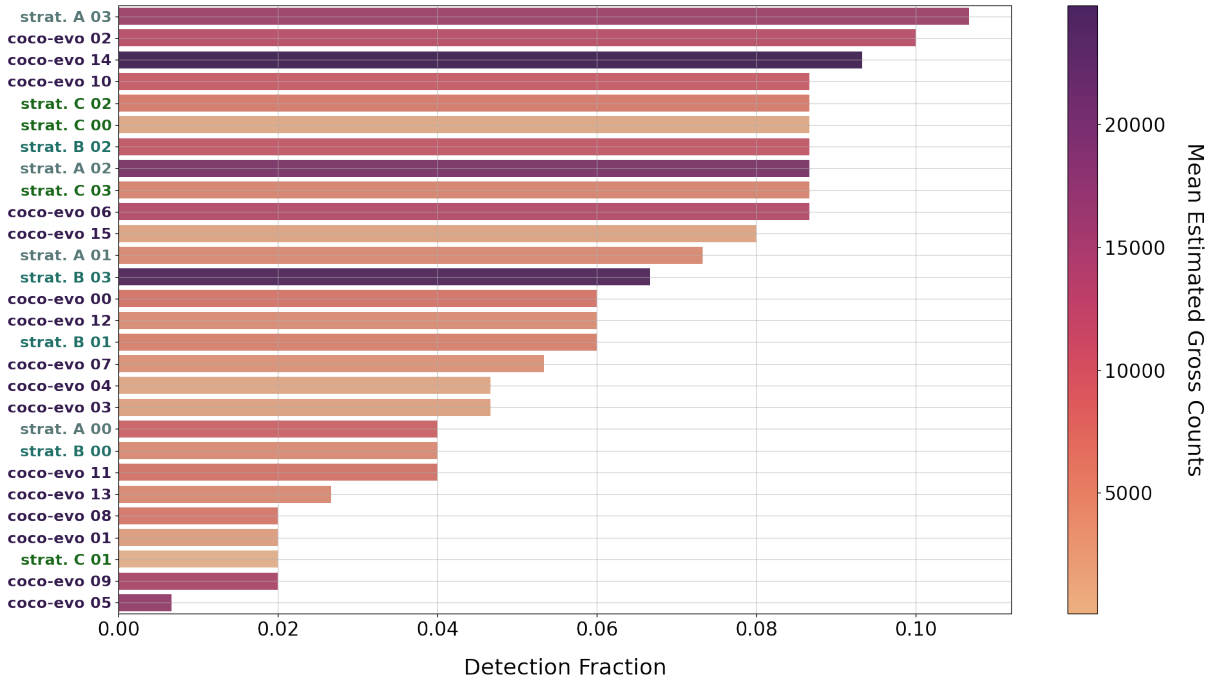


(a) 50-Detector Networks

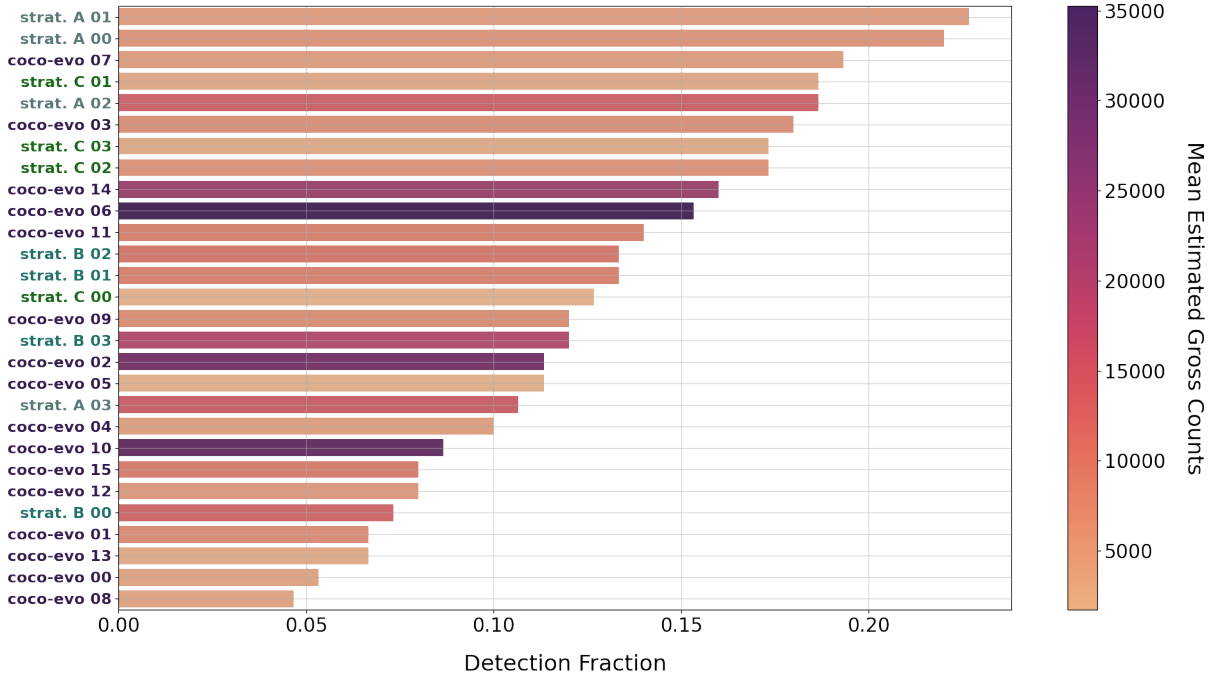


(b) 100-Detector Networks

Figure 5.4: Letter-value plots of IIA* minimal path cost distributions across detector network configurations. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 5% targeted routes.



(a) 50-Detector Networks



(b) 100-Detector Networks

Figure 5.5: Comparison of summarized IIA* results across network configurations using route dataset of 5% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations.

are detected by the network. Specifically of interest are the configurations 02, 10, and 14; the networks of these three demonstrate generally good performance across the range of evaluation conditions with some notable variation. Configuration 02 attains very high mean gross counts across the board, with slight dips for the combination of larger detector quantities and fewer targeted routes in the training dataset. In those same circumstances, it tends to detect a relatively low amount of total paths, but maintains an approximately average detection fraction in most of the evaluation studies. With configuration 10, the best average gross count quantities are achieved for 200-detector network evaluations. Its detection fraction tends neither towards worst or best performance, with an exception being for 50-detector network evaluation with low numbers of targeted routes in the test dataset. The results of evaluations with configuration 14 reveal its high rates of detection fraction with 50 and 100-detector networks, especially so for minimal ratios of targeted routes; this drops off steeply for 200-detector evaluations. Average gross count figures produced by this configuration follow an approximately inverse gradient. Poor metrics are obtained for 50-detector networks at low numbers of targeted routes, but the configuration scores very highly for the 200-detector cases.

The particulars of the COCO-EVO procedures that produced these network configurations should also be mentioned. Using Table 5.3, it can be noted that these three configurations all use traffic-based initialization as well as the density sampling techniques. This is a valuable takeaway that could be used to guide further study. The differences between the COCO-EVO runs leading to these network designs are isolated to the adversarial path method used and the targeted route training percentage. For this former variable, FPA* is used by 02 and 14, while 10 uses IIA*. With regards to the targeted percentage, 02 is created using a value of 50%, and both 10 and 14 utilize 100%.

The high average estimated gross count numbers obtained by the most successful COCO-EVO network configurations can also be analyzed in a statistical frame. In the context of the nuclear threat detection problem, this behavior could lead to high rates of true positive identification but simultaneously increased rates of false negatives due to the subpar detection fractions. Put another way, these high-performing networks should exhibit good values of precision, which is described with

$$Precision \;=\; \frac{TP}{PP} \tag{5.2}$$

where TP is the number of true positive identifications and PP is the number of predicted positives. However, poor recall performance should be expected; this metric is defined with

$$Recall \;=\; \frac{TP}{P} \tag{5.3}$$

wherein TP again describes the quantity of true positives while P represents the total number of positive events to detect in the datastream. This characterization can be compared in the same manner to the results of strategy A; with those networks' high comparative detection fraction they should be expected to express high recall metric values. Yet, the lower average signal strength could complicate setting an anomaly detection threshold and thereby result in significant false positive rates. This would hurt the resulting precision values, since the quantity of predicted positives PP is the sum of true positive and false positive identifications.

5.5.2 General Detector Positioning Analysis

A system of position scoring is devised to evaluate placement of detectors in various locations throughout the urban area studied. In this approach, nodes on the road graph are assessed based on the frequency of detection of paths which transit through them. A positive score is given to a node when a detector positioned on it or within close proximity registers a path passing through it. In other words, higher score equates to nearer transit to a detector. The metric value is calculated with

$$S_i \;=\; \frac{1}{k} \sum_{i=0}^{k} c_i \; d \tag{5.4}$$

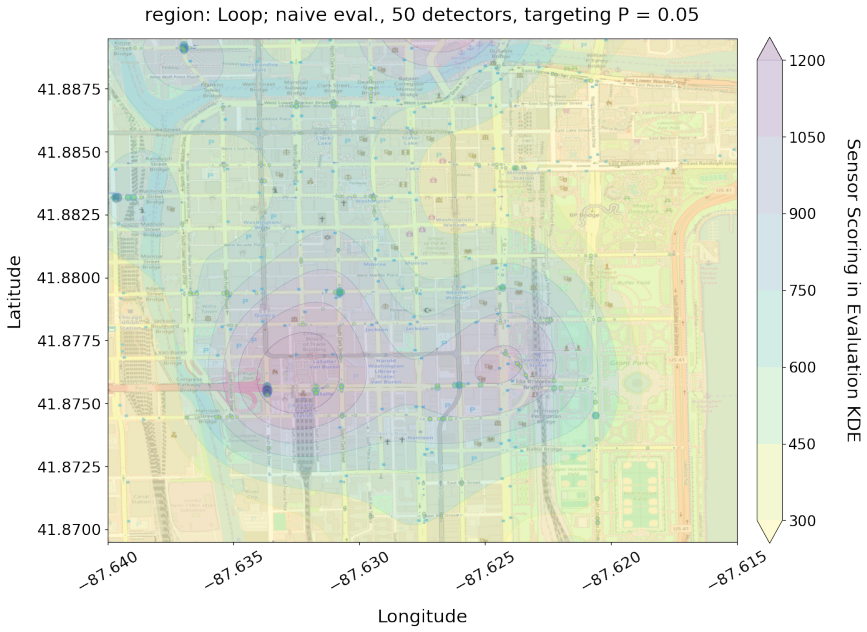
wherein the score S_i is defined for a specific node i on the graph, with k evaluation paths including i, a detection cost c_i based on the proximity of i to the actual detector placement, and a depth modifier d that amplifies the scoring if the path is detected at a greater adversarial iteration depth (see Section 5.4.1 for more detail). The use of d to amplify

scoring for increased depth detection is referred to as relaxation depth scoring modification (RDSM) in plotting later in this sections as well as in Appendix G.

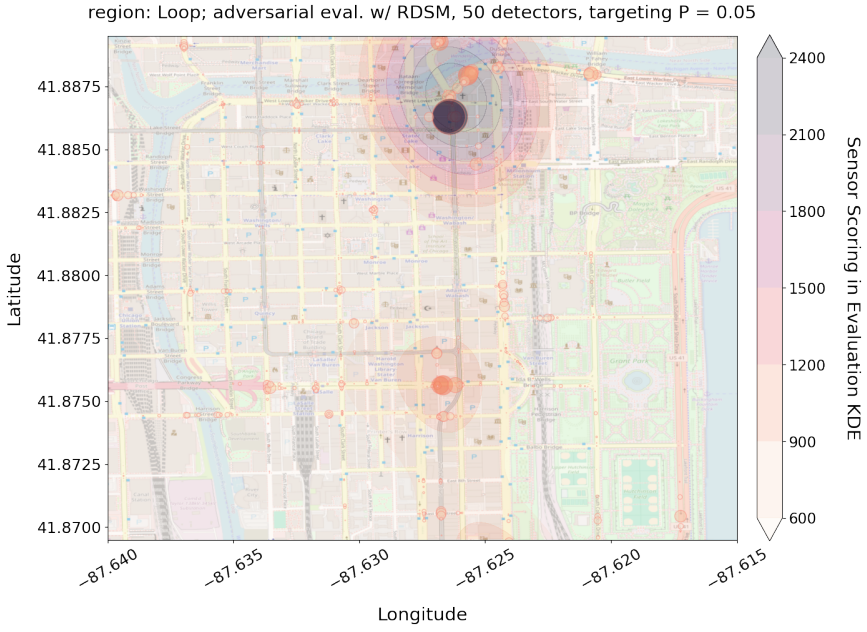
Scores are calculated in two different evaluation modes; these are termed "naive" and "adversarial". They are differentiated by their focus on the value of d. With naive evaluation, path detections are scored when $d = 1$; this is comparable to the use of a non-adversarial path planning method. In contrast, adversarial evaluation scores detections for values of $d > 1$. The motivating idea for this distinction is to compare the efficacy of detector positions against non-avoidant and adversarially avoidant paths. Large discrepancies should then indicate a crucial need for network modification in order to effectively detect adversaries in addition to unmodified transit.

Results using the above methodology for networks comprised of 50 detectors are displayed for a subregion of the study area mainly containing the Loop area of downtown Chicago in Figs. 5.6 and 5.9 for evaluations on test route datasets with 5% and 50% targeted routes, respectively. The corresponding residuals are shown in Figs. 5.7 and 5.10. Residuals for 100 and 200 detector networks for the same subregion and test route datasets are shown in Figs. 5.8 and 5.11. Information from separate network configurations is compiled for these results in order to improve the statistical picture. It is presumed that this will have negligible deleterious effects on ability to coherently assess strong positions for detectors. Complete detector position scoring results are presented in Appendix G.

A few emergent trends are observable from the visualizations of sensor scoring. The first of these is that it is generally apparent that detector positions scoring well in adversarial evaluation tend to be clustered around "choke points" of vehicle ingress and egress. For example, detectors placed around the Bataan-Corregidor Memorial Bridge and the Irv Kupcinet Bridge on the Chicago River consistently achieve high adversarial scores. The same detail is noticeable in Figs. 5.10 and 5.11a around the highway access at West Ida B. Wells Drive. Intuition would suggest that such choke points would be advantageous positions for detectors, and these results seem to bear that out. It can be conjectured that there are simply minimal options for path planning to connect certain OD pairs without using such edges on the graph. A distinction can also be made between these and areas that are simply high in traffic activity; many of those in the latter category are on highways or



(a) Naive evaluation



(b) Adversarial evaluation

Figure 5.6: Detector position scoring with naive and adversarial pathing evaluation using test route dataset of 5% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown for Loop area of downtown Chicago.

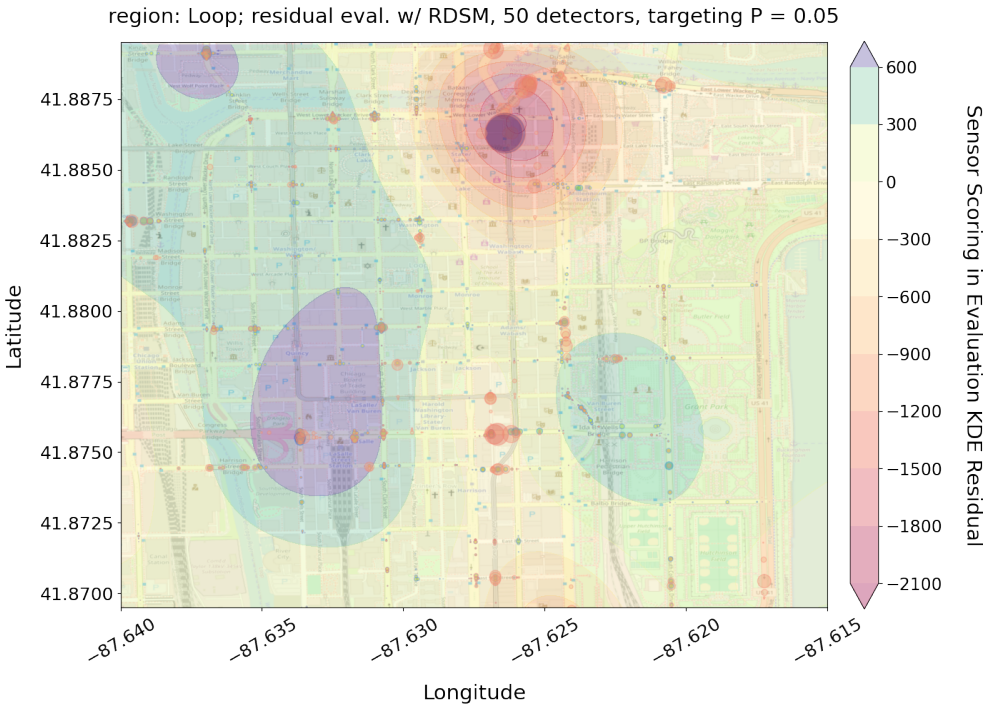
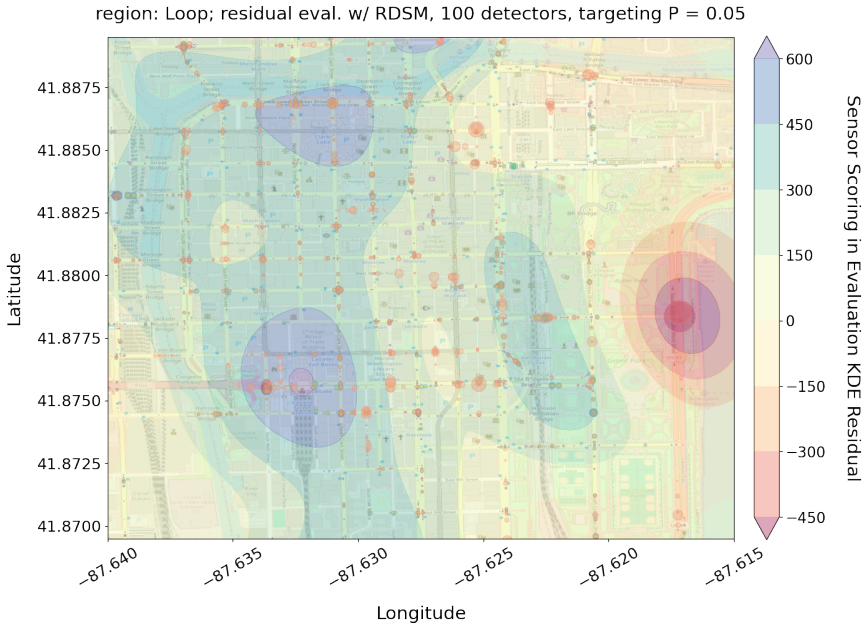
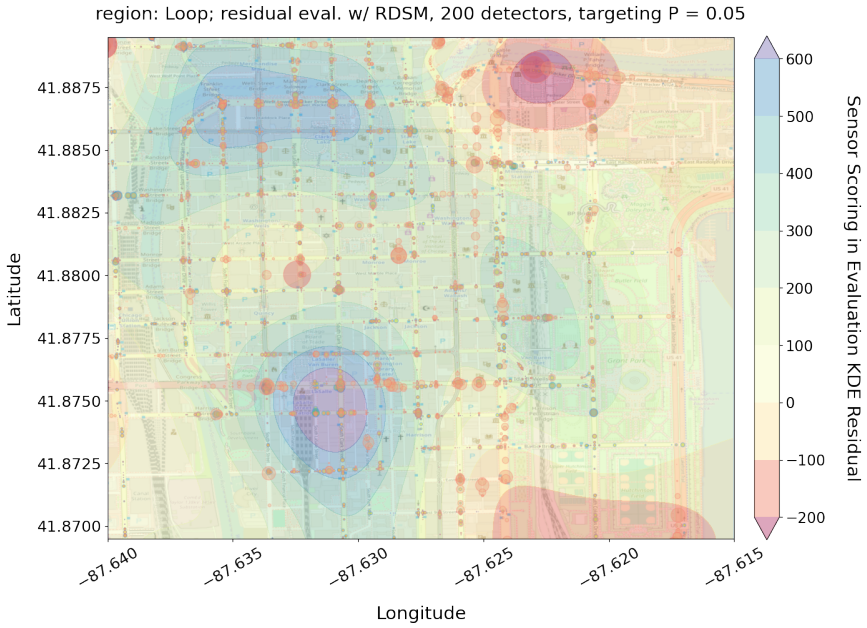


Figure 5.7: Residual of detector position scoring KDEs shown in Fig. 5.6. Positive values indicate higher position scoring in naive path evaluation, whereas negative values are indicative of higher scoring with adversarial path evaluation. Detail shown for Loop area of downtown Chicago.

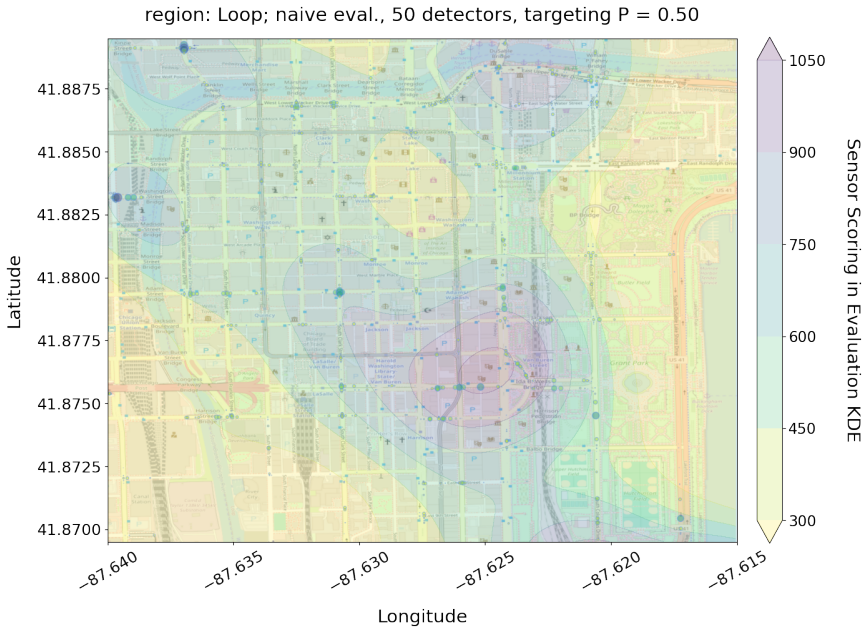


(a) 100-Detector Networks

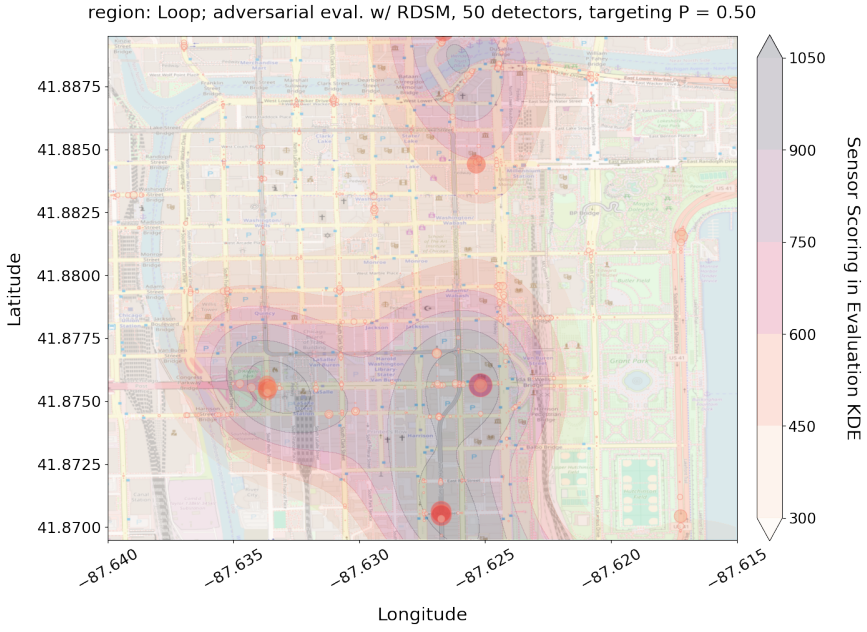


(b) 200-Detector Networks

Figure 5.8: Residual between naive and adversarial evaluation of detector position scoring KDEs using test route dataset of 5% targeted routes. Positive values indicate higher position scoring in naive path evaluation, whereas negative values are indicative of higher scoring with adversarial path evaluation. Detail shown for Loop area of downtown Chicago.



(a) Naive evaluation



(b) Adversarial evaluation

Figure 5.9: Detector position scoring with naive and adversarial pathing evaluation using test route dataset of 50% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown for Loop area of downtown Chicago.

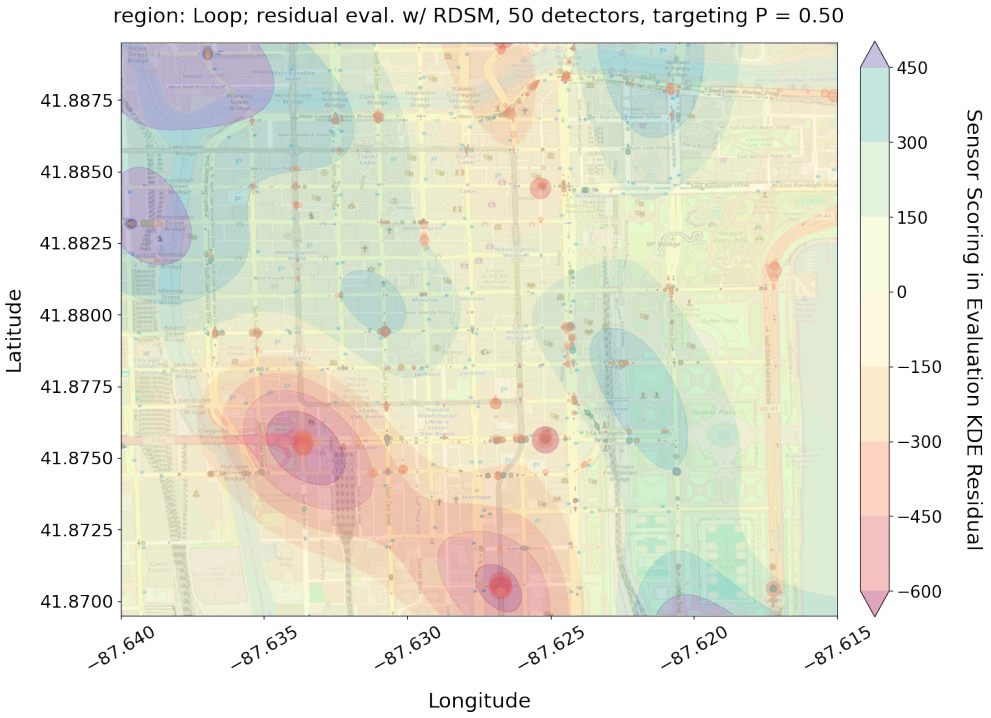
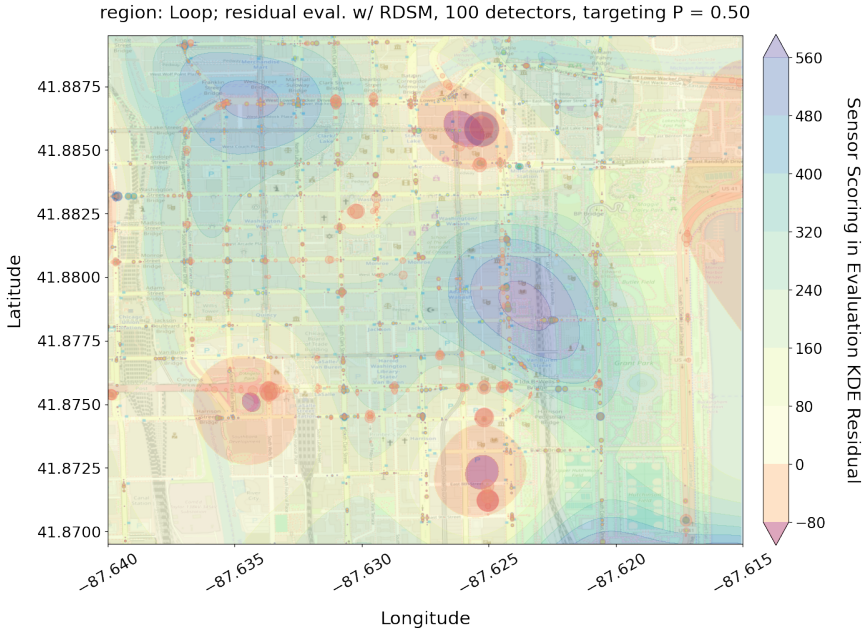
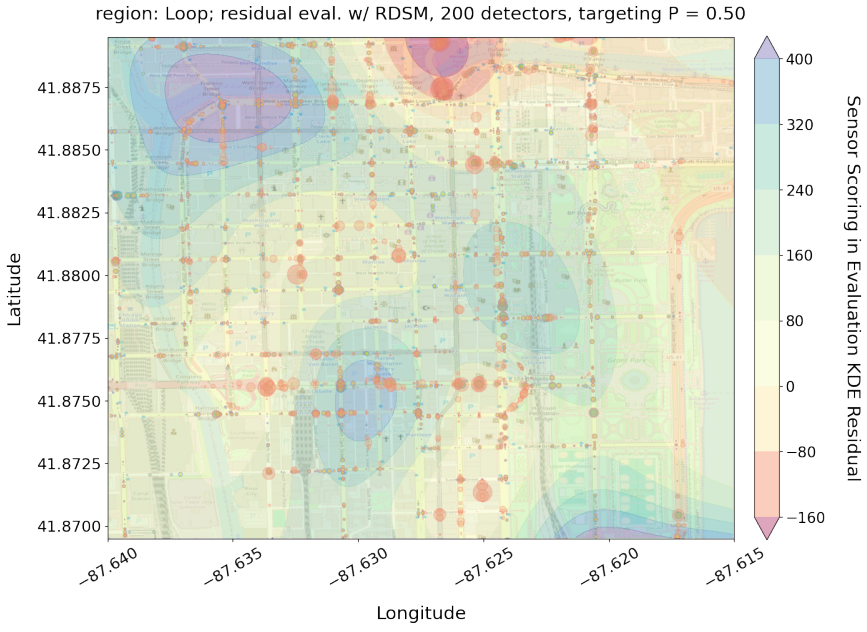


Figure 5.10: Residual of detector position scoring KDEs shown in Fig. 5.9. Positive values indicate higher position scoring in naive path evaluation, whereas negative values are indicative of higher scoring with adversarial path evaluation. Detail shown for Loop area of downtown Chicago.



(a) 100-Detector Networks



(b) 200-Detector Networks

Figure 5.11: Residual between naive and adversarial evaluation of detector position scoring KDEs using test route dataset of 50% targeted routes. Positive values indicate higher position scoring in naive path evaluation, whereas negative values are indicative of higher scoring with adversarial path evaluation. Detail shown for Loop area of downtown Chicago.

other high-access grade roads where detectors are unable to be placed in both reality and in modeling.

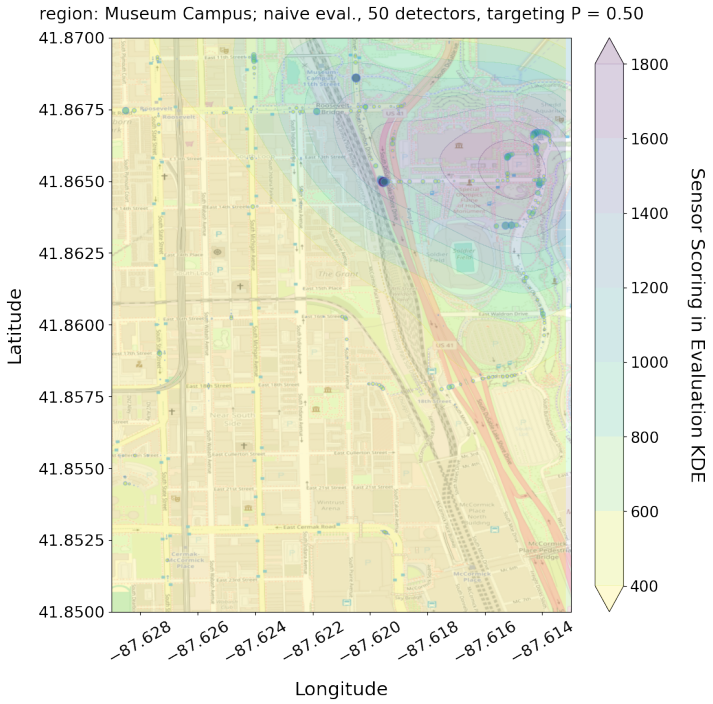
Another of the aforementioned trends is the apparent flattening of residuals between naive and adversarial path scoring with increased quantities of detectors. This is most notable in comparison between Figs. 5.7 and 5.8b. This could be reasonably explained by a saturating effect of the quantity of detectors, with more of the gaps in the road graph able to be filled in. In such cases, network design is torn to a far lesser extent between optimizing for the naive versus the adversarial case; one configuration can go a great deal further towards solving both problems.

Also of note in these results is the concentration of detector scoring around higher-priority target locations as presented in Table D.1. This is most observable in the outputs of evaluations with the test route datasets containing either 50 or 100% specifically targeted routes. It is visible in Figs. G.6, G.9, G.18, G.21, G.30, and G.33, but is also seen in closer detail in Fig. 5.12. This latter figure focuses on the museum campus subregion, which contains Soldier Field. In essence, this can be explained as the network guaranteeing detection of paths planned to these locations simply by positioning detectors at the destination. While this strategy appears functional to some extent in these evaluations as formulated, it is worth questioning whether it would translate to a more complete adversarial model. For instance, with a larger table of adversarial target locations, the quantity of detectors would at some point become too small to attempt positioning exactly at or nearby to them. This hypothesis can be examined by comparing the somewhat more diffuse nature of the KDE in Fig. G.9b with the more sparse and peak-heavy distribution shown in Fig. G.21b.

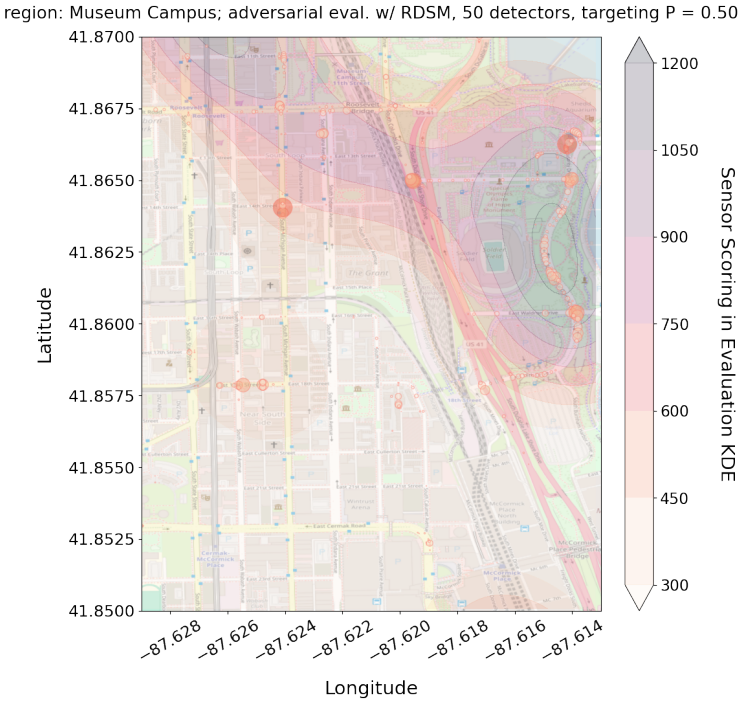
A related feature of interest is the heavy concentration of scoring at larger detector quantity evaluations at seemingly random points around the Northern, Western, and Southern borders of the region of study. In actuality, these points represent some of the "exit" reference points that are stochastically selected from by the route generation procedure described in Section 5.2.2. This approach samples from a finite list of exits from the region in order to model routes that have an origin or destination point outside the explicit study area. The same problem may be present as a result in this context as is presumed with the aforementioned supposition about insufficient quantity of route targets. Again, this can be

illustrated by comparing the differences between the results from evaluations with 50-detector networks in Fig. G.4b and those of 100 or 200-detector networks in Figs. G.15b and G.27b, respectively.

It is also sought to compare the scoring distributions produced with static (A, B, and C) and dynamic (COCO-EVO) network positioning strategies in order to further investigate the divergences in performance observed in Section 5.5.1. These are presented in Figs. 5.13, 5.14, 5.15 and 5.16. It is observed in Fig. 5.13 that the scoring of static strategies is generally concentrated around the Loop area in the city's downtown core; in contrast, that yielded by the dynamic configurations is much more widely dispersed. It could be argued that for these specific results, the static strategies are theoretically better; this is due to the fact that COCO-EVO configurations evaluated are sampled from the best performing adversarial individuals. With additional time and/or computational resources, it would be ideal to sample from and evaluate the best naive performers as well. With Fig. 5.14, a comparison is shown for adversarial scoring. A much greater degree of convergence in the distributions is readily apparent, further detail of which is shown in Fig. 5.15. Both static and dynamic approaches are able to make detections of intelligent adversaries traversing the roads around the Irv Kupcinet and DuSable bridges, making a case for this area's status as a significant chokepoint. When evaluating on route datasets with larger fractions of targeted routes, these results change significantly, as shown in Fig. 5.16. While both static and dynamic strategies seem to be in agreement as to the value of detectors positioned around access to the I-290 interstate (bottom left of the map), there is broadly much divergence between the distributions. One possible explanation for these differences is that the COCO-EVO approach finds more consistent scoring with a boundary-like approach since the adversary was allowed to reconfigure its understanding to particular areas with high probability of detection.

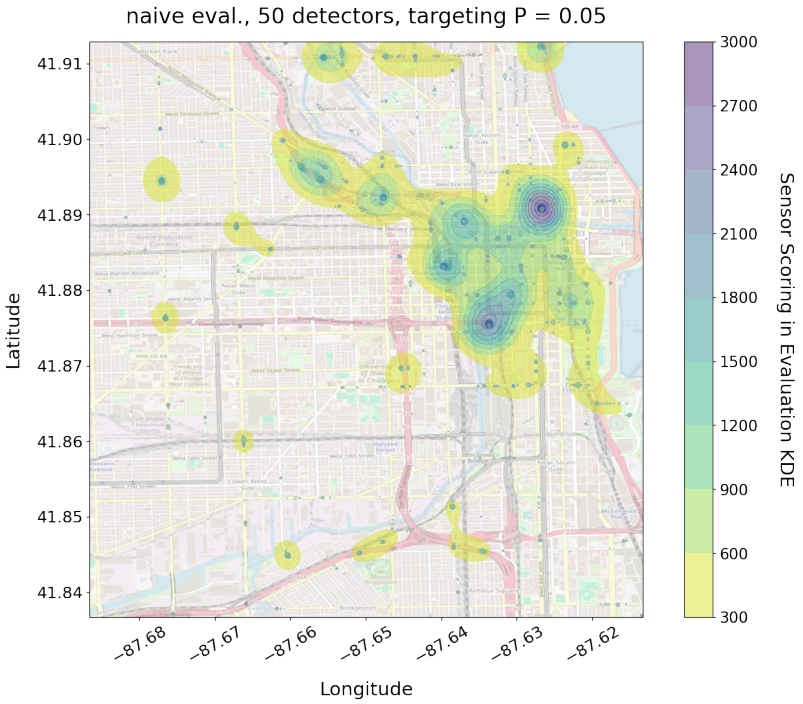


(a) Naive evaluation

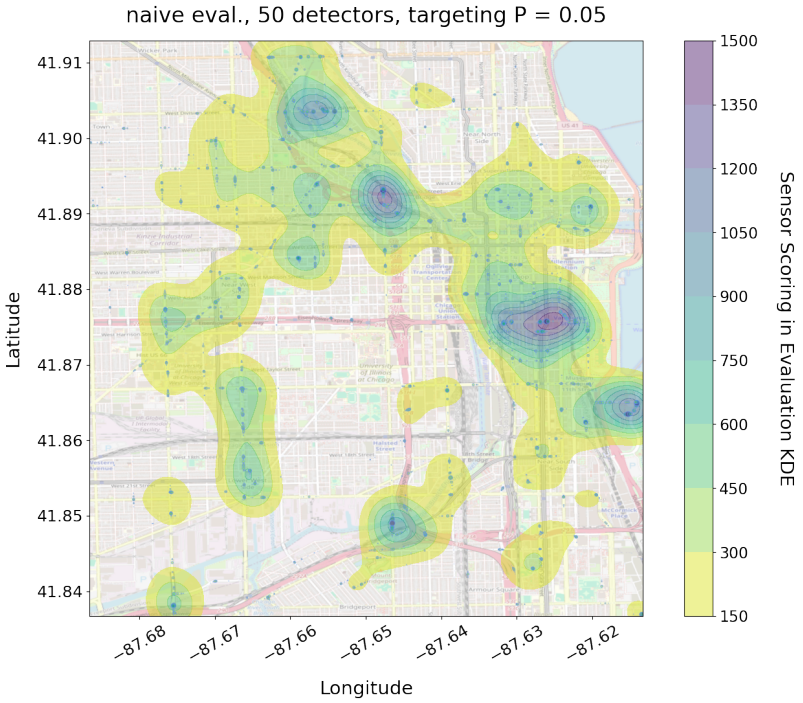


(b) Adversarial evaluation

Figure 5.12: Detector position scoring with naive and adversarial pathing evaluation using test route dataset of 50% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown for museum campus area of Chicago on the shore of Lake Michigan.

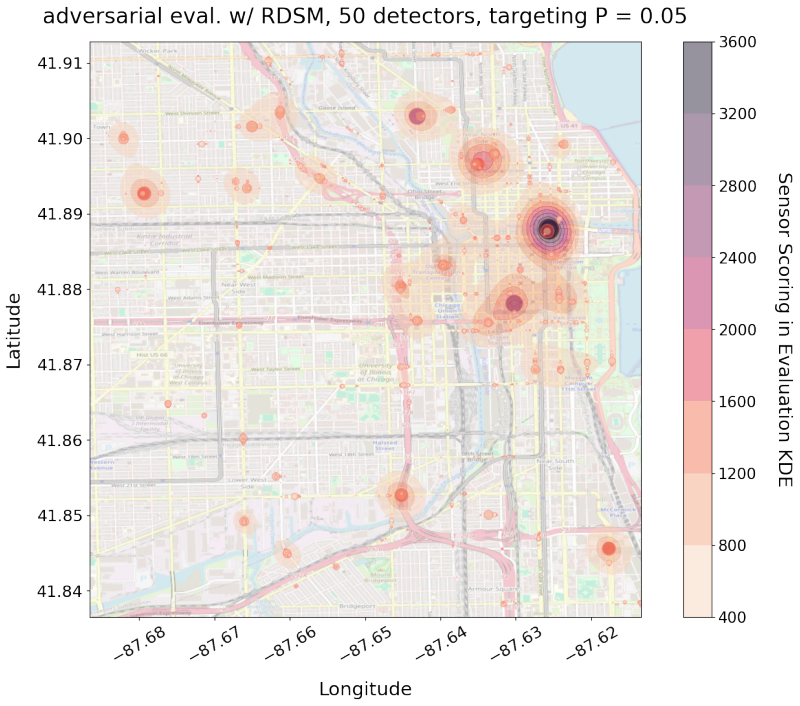


(a) Static strategies

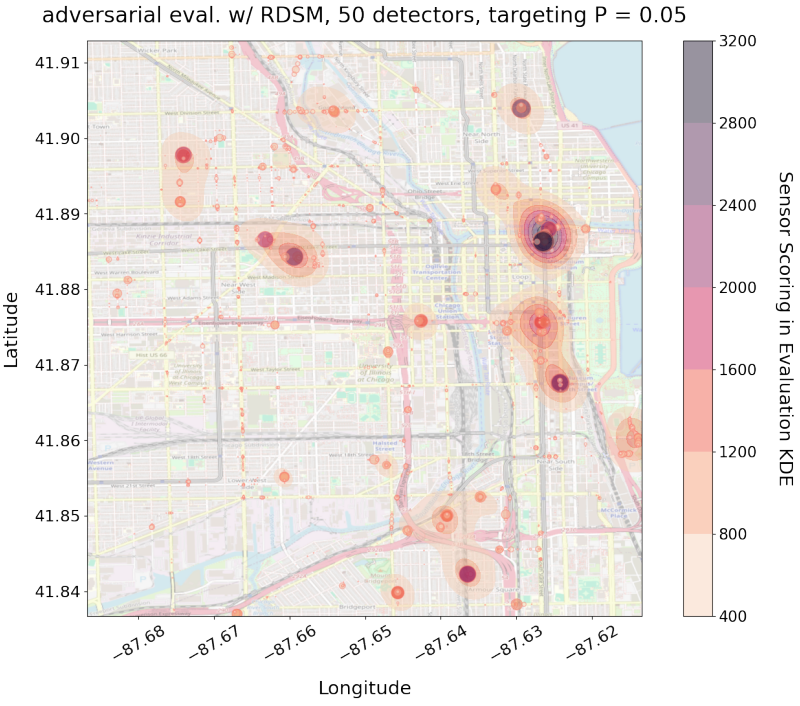


(b) Dynamic strategies

Figure 5.13: Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using naive pathing evaluation and a test route dataset of 5% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown.

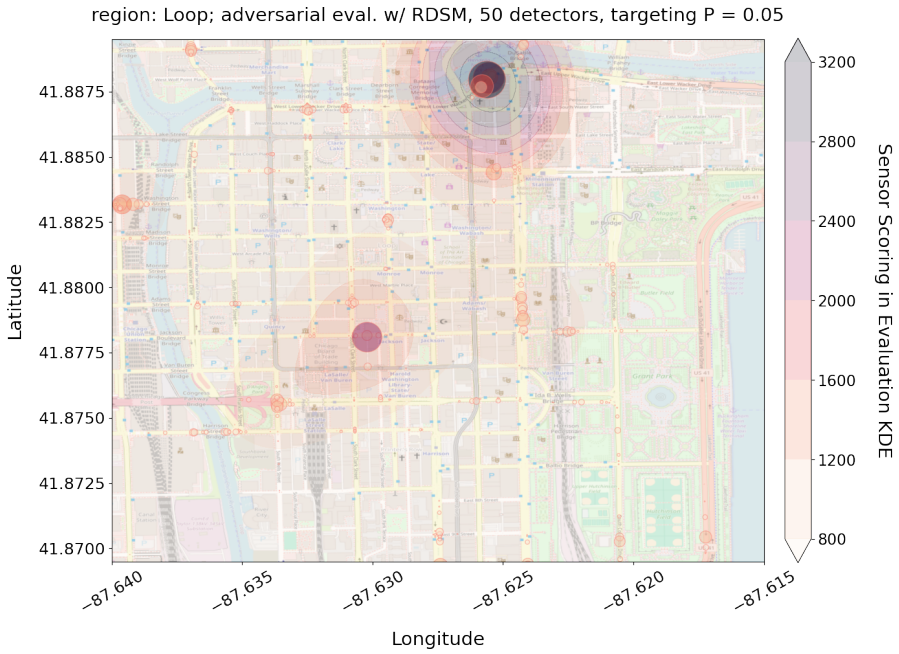


(a) Static strategies

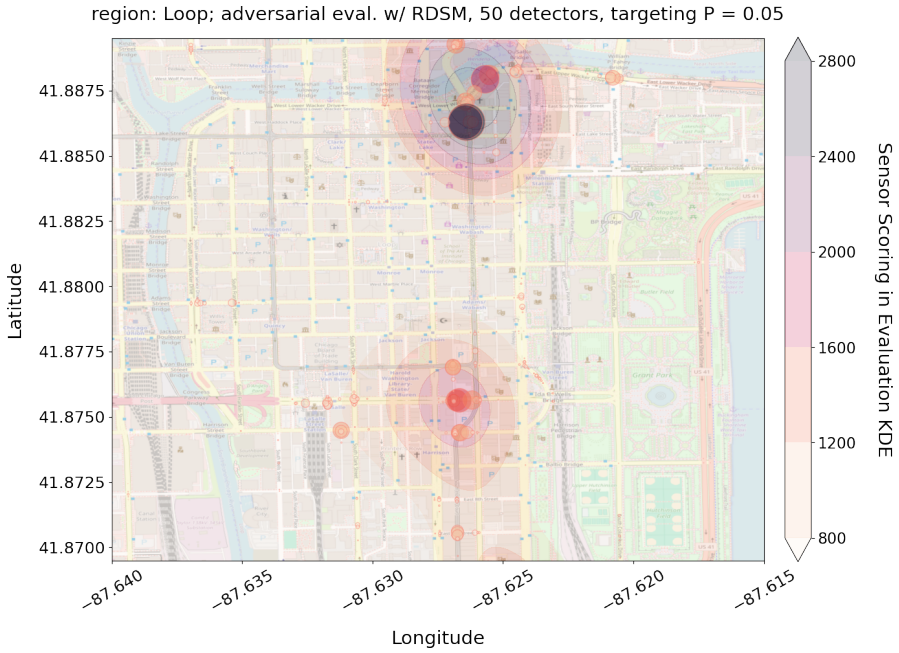


(b) Dynamic strategies

Figure 5.14: Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 5% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown.

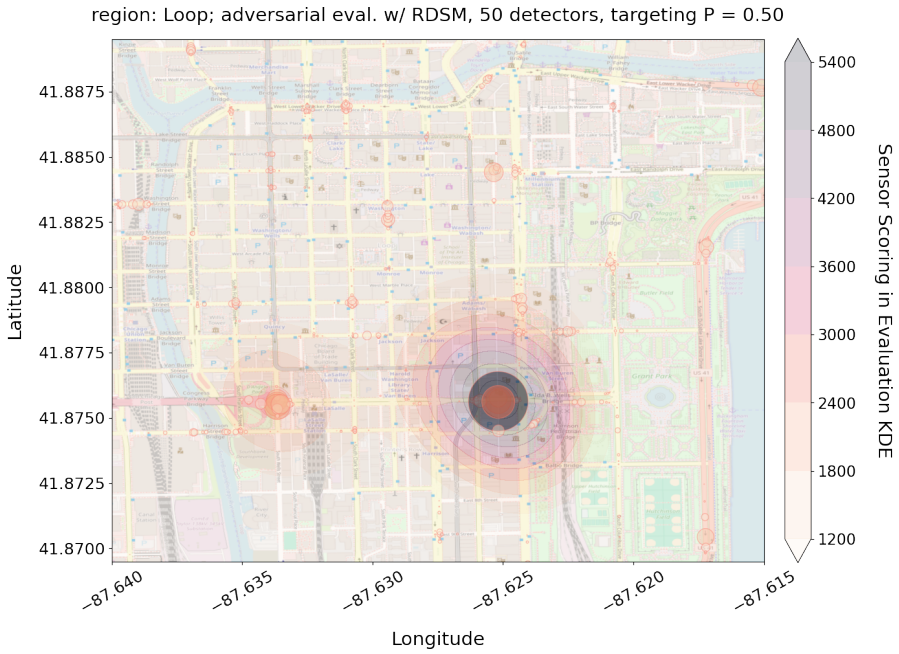


(a) Static strategies

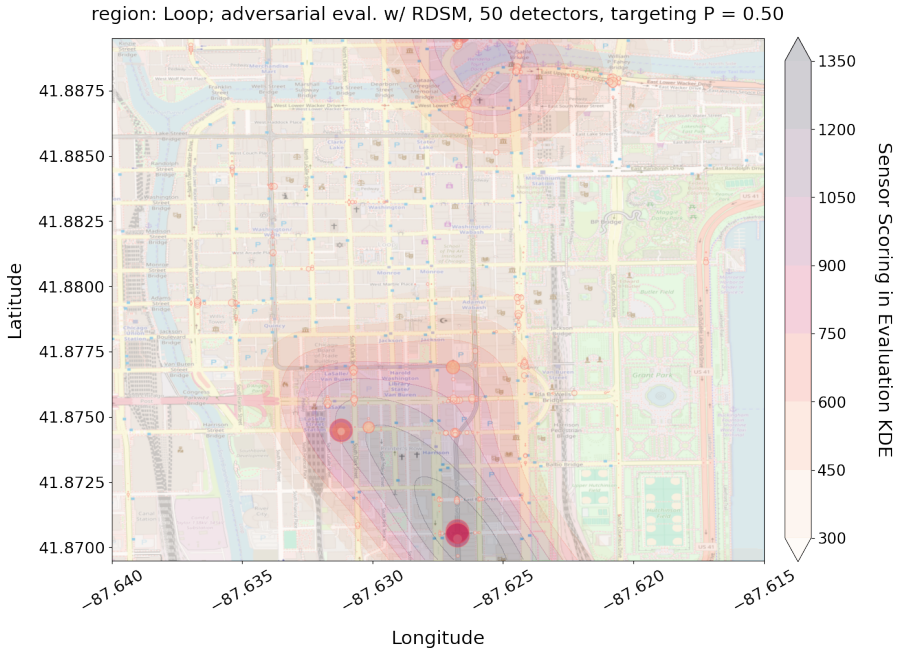


(b) Dynamic strategies

Figure 5.15: Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 5% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown for the Loop area of downtown Chicago.



(a) Static strategies



(b) Dynamic strategies

Figure 5.16: Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 50% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown for the Loop area of downtown Chicago.

Chapter 6

Conclusions

This text has presented investigations into several applied radiological problems with an overarching approach to modeling using multiple objectives and finding optimal solutions in that frame. With each case study, a modeling process was carried out in the manner described in Section 1.4. The input parameters, relevant objectives, and MO optimization algorithm selected were identified and explained before discussion was focused on details of the application-specific objective function. In several of the applications, this latter point involved the design and implementation of a novel algorithm.

Results obtained with this methodology explicitly preserve information as to tradeoffs between objectives, the values of which are easily correlated with design parameters. This not only serves to provide greater flexibility to a DM tasked with selecting from a range of optimal compromises; it also allows clear interpretability regarding the effects of parameter choices on output behavior. Based on the results obtained in the course of these investigations, it is argued that this is a powerful and worthwhile approach for studying applied problems in this domain of research.

The remainder of this chapter covers important takeaways specific to each case study. Considerations as to potential future work and possible broader applications of the novel methods developed are also reviewed.

6.1 Natural Environments

6.1.1 Node Placement

From the efforts detailed in Section 4.2, takeaways were derived from examination of the solution candidates distributed in objective space. The distribution's two-dimensional projection of node quantity against reconstruction similarity metric indicated that beyond a quantity of 35-40 nodes, optimal networks would only see limited marginal improvements in reconstruction performance with additional nodes.

Through parameter shading of the population distribution data, information was able to be gleaned as to the influence of the input parameterizations. There was no obvious best of the placement algorithms evaluated, although the random method predictably exhibited highly inconsistent performance. The sensitivity-weighted methods tended to produce solutions with the highest reconstruction similarity values when used in concert with larger quantities of nodes. Farthest-first traversal methods intuitively generated networks with less clumping and, therefore, arguably more robustness.

The quantity of nodes allotted for constructed of the network was revealed to not strictly control the reconstruction performance; instead, it produced clusters of solution candidates with variability depending on other parameters. It was also determined that GPR generally performed better than bicubic interpolation for efficiently reconstructing the radiological intensity distribution from the relatively sparse node measurements.

Further work in this area could evaluate the sensitivity of the presented results to other topographies and source terms. This type of analysis is important to ensure the robustness of the findings detailed here under other conditions. Additionally, other reconstruction methods could be used as parameterization options against the now-established contextual standard of GPR performance. Among candidates for this are variational autoencoders (VAEs, [141]); while sharing some characteristics with the deep learned I2I models previously tested, an important distinction lies with the VAE's attempt to learn the statistical distribution of the latent space. This would provide greater explainability along with making more sense in the context of the application, since some degree of statistical consistency is expected from the radiological source terms.

6.1.2 Path Planning

The main contribution put forward in Section 4.3 was the design of the FPA* fuzzy MO path planning algorithm and its evaluation with the context of environmental radioactive contamination and emergency response. Computational methods were also presented describing the calculation of application-specific heuristics for terrain and exposure cost objectives.

The results presented in Section 4.3.6 serve to demonstrate the ability of FPA* to provide an array of optimal path choices, each finding different tradeoffs between objectives. The objective space distribution results of run *B-3* demonstrated a clear inverse relationship (with respect to optimality) between both the distance and terrain cost objectives and the exposure cost objective. The nature of this relationship was visualized with a selection of optimal paths found for this run. In the context of the algorithm's optimization, the expected impact of each parameter value was confirmed by investigating their correlations with the relevant objective.

A small-scale sensitivity analysis using the results of 25 FPA* optimizations with different paths, regions, and radiological distributions was undertaken as well. With the results of this effort, it was argued that these contextual variables have limited effect on optimal FPA* parameter values. The convergence of the FPA* optimization was also reviewed to provide surety as to the optimality of the obtained parameterizations. Additionally, a FoM was calculated to ascertain the complexity of the FPA* algorithm; the resulting distribution was then shown and discussed.

In continuing analysis in this vein, a few areas that appear immediately promising in terms of potential for improvement or new inquiry are identifiable. The results could be made stronger through a greater quantity and/or diversity of environments evaluated. For example, using the quantification of regions described in Section 4.1.1, more varied topographies might be evaluated. This follows from the analysis in 4.3.6, which would become much more robust through these additional experiments. In addition, with greater compute resources, larger regions or finer discretizations could be analyzed.

Continuing investigation could also involve implementations of FPA* in other contexts. Resuming the discussion in Section 1.2.2, it is argued that environments requiring nontrivial navigation methods are more common than might be readily apparent in the world. This can be observed clearly from the diversity of applied work reviewed in Section 2.4, including that of construction site navigation in [88] and Martian surface path planning investigated in [86].

Of relevance here is the optimization mechanism of FPA*, which provides flexibility to a hypothetical DM to make an objective-prioritizing at decision time. This is an important detail pertaining to dynamic situations, such as extreme weather events or natural disasters. FPA* could be utilized to plan a range of evacuation paths from threatened communities, with the best one selected for use depending on the current status of the storm, fire, or other hazards. Considering this hypothetical community evacuation scenario closely, it is noted that the grid-graph approach to modeling physical space used in this study would need to be replaced with a graph representation of roadways in addition to the new objectives (and their heuristics) requiring development. The first part of this has been accomplished in the course of work covered in Chapter 5, with FPA* having been generalized to function on the OSM Chicago road graph for network evaluation purposes.

6.1.3 Measurement Update Sensitivity

Optimization of a dynamic window DD algorithm for a plume monitoring application was presented in Section 4.4, with consideration to the number of measurements made by a radiation detector and the accumulated error over time. It was found for the 312-timestep plume intensity sample evaluated that the cumulative measurement error could be significantly reduced with approximately 30 measurement updates. Additional updates were observed not to provide further reductions in error, until about 60 were used at which point it was determined via parameter analysis that the DD algorithm began to overfit on the sample.

A continuation of this work should integrate some form of sensitivity analysis, primarily with attention to differences in the datastream. This could range from evaluation using another sample from the same simulated plume to sampling from a new plume generated

with different topography and wind conditions. Another avenue could involve testing of other DD algorithms; these are mainly differentiated by their manipulation of the data window used as well as by the specific test utilized to evaluate whether a new sample is anomalous [142, 143].

6.1.4 Multi-Robot Exploration

In the case of the multi-robot exploration and mapping problem studied in Section 4.5, the novel MR-MOPSO algorithm was observed to provide satisfactory results under simple evaluation conditions. For the optimal sample solution studied in detail, it was demonstrated that the approach was able to steadily improve on both the aggregate uncertainty and reconstruction similarity objectives throughout the simulation duration. Additionally, a knee in the MOEA output distribution was identified around approximately 120 seconds of deployment time elapsed, indicating very high marginal gains in the other two objectives up until that temporal point in the system operation.

Unfortunately, limitations on codebase access prevented more comprehensive investigation on this problem including closer analysis of the results due to a change in affiliation. Assuming that this matter can be resolved in the future, further work should test the sensitivity of the algorithm's performance to other radiological source terms. Another continuation could involve the integration of real topography as is used in the problems discussed previously in Chapter 4. In addition to changing the properties of the source term intensity distribution, this would also require a modification of the simulated motion of the robots themselves to account for gradients in terrain.

6.2 Urban Environments

Chapter 5 presented a study on the optimization of urban radiation detection sensor networks with consideration towards intelligent adversarial avoidance behavior. The IIA* algorithm was introduced as a tool to model an adversarial agent planning a path through an urban area of interest. In addition, the COCO-EVO competitive multi-population optimization

framework was developed as part of the effort to model the dynamics at play between optimal strategies for detector networks and the adversarial strategies aimed at avoiding them.

Based on the results presented in Section 5.5.1, it was determined that the static placement method termed Strategy A was most consistently successful at detecting to a minimal degree the highest fraction of evaluated adversarial routes. In contrast, the configurations created using COCO-EVO outputs were found to maximize average quality of detection for routes detected by the network. It was then argued that these two methods represent opposite ends of the statistical performance metric spectrum; the COCO-EVO networks were posited as likely to exhibit high precision but low recall, with the inverse expected for Strategy A networks. In addition, the findings of the configuration evaluation pointed to the most advantageous hyper-parameterizations of COCO-EVO to utilize in subsequent analysis.

The positioning analysis and scoring detailed in Section 5.5.2 provides important insights as well. ... Differences between the positions most advantageous to static and dynamic placement strategies were also reviewed. These results – especially those using datasets with higher fractions of targeted routes – confirmed that there is significant divergence in realization of the detector positions that observe source carriers. It was ...

Results of the analysis suggest a number of subsequent directions for the work. The findings of the detector position scoring in Section 5.5.2 suggest that larger numbers of distinct adversarial targets are needed to prevent overfitting in detector network strategies. While those configurations are technically optimal given the evaluations used, they are somewhat unrealistic in the context of the real-world application. It is argued that an intelligent adversary, upon seeing detectors positioned at some particular prime target location, will likely just choose a different target rather than accept a high risk of detection.

Additionally, evolutionary approaches are by no means the only way to go about optimization in this context. One idea that appears promising is the use of realized-particle PSO in the style of the approach used in Section 4.5. In the place of robots, this implementation would model detectors as individual particles; from some initialization, the detectors theoretically would be incrementally moved towards an optimal configuration based on objective performance at each step.

For completeness, continued analysis on this problem should also examine modifications of the exact urban context used in this work as a form of sensitivity analysis. This could include consideration of a larger size area of study or regions of different cities altogether. Also in this vein, further work could add attention towards other modes of transit besides solely vehicles in modeling potential paths of detector network circumvention.

Other hyperparameters could also be more thoroughly searched. Among these are the quantities of detectors used to construct the networks, the makeup of route datasets sampled from for evaluation during the COCO-EVO procedure, and the time delay of information known by each population about the other. In practice, further exploration of these hyperparameter spaces was limited by time and computational resources.

Lastly, the COCO-EVO framework could potentially have broader applicability. The notion of two distinct populations in non-stationary competition is by no means limited to the application dealt with here. For instance, the dynamics of predator-prey interactions in nature [144] as well as those of immune response to pathogens on an epidemiological scale [145] present interesting modeling challenges. Further evaluation and refinement of COCO-EVO would be necessary in order to confidently extend it to contexts such as these.

Bibliography

[1] Armand Albergel, Daniel Martin, Bernard Strauss, and Jean-Michel Gros. The Chernobyl accident: Modelling of dispersion over Europe of the radioactive plume and comparison with air activity measurements. *Atmospheric Environment (1967)*, 22(11):2431–2444, 1988. 1, 14

[2] Kirk J. Cantrell. Transuranic contamination in sediment and groundwater at the US DOE Hanford site. Technical report, Pacific Northwest National Lab.(PNNL), Richland, WA (United States), 2009. 1

[3] Yang-Hyun Koo, Yong-Sik Yang, and Kun-Woo Song. Radioactivity release from the Fukushima accident and its consequences: A review. *Progress in Nuclear Energy*, 74:61–70, 2014. 1, 14

[4] Carlisle E. W. Topping, Maveric K. I. L. Abella, Michael E. Berkowitz, Monica Rouco Molina, Ivana Nikolić-Hughes, Emlyn W. Hughes, and Malvin A. Ruderman. In situ measurement of Cesium-137 contamination in fruits from the northern Marshall Islands. *Proceedings of the National Academy of Sciences*, 116(31):15414–15419, 2019. 1

[5] VA Knizhnikov and RM Barkhudarov. Radioactive contamination of the environment and the possible consequences. *Atomic Energy Review*, 13(2):171–214, 1975. 1

[6] Hiroki Wakamatsu and Tsutom Miyata. Effects of radioactive safety information on consumer fears of radioactive contamination from oyster products in japan. *Marine Policy*, 126:104401, 2021. 2

[7] Neil J Morley. The effects of radioactive pollution on the dynamics of infectious diseases in wildlife. *Journal of environmental radioactivity*, 106:81–97, 2012. 2

[8] Stanislav Geras'kin, Alla Oudalova, Nina Dikareva, Sergey Spiridonov, Thomas Hinton, Elena Chernonog, and Jacqueline Garnier-Laplace. Effects of radioactive contamination on scots pines in the remote period after the chernobyl accident. *Ecotoxicology*, 20:1195–1208, 2011. 2

[9] Tomoko M Nakanishi and Keitaro Tanoi. *Agricultural implications of the Fukushima nuclear accident: the first three years*. Springer Nature, 2016. 2

[10] Arifumi Hasegawa, Tetsuya Ohira, Masaharu Maeda, Seiji Yasumura, and Koichi Tanigawa. Emergency responses and health consequences after the fukushima accident; evacuation and relocation. *Clinical Oncology*, 28(4):237–244, 2016. 2

[11] Takashi Ohba, Koichi Tanigawa, and Liudmila Liutsko. Evacuation after a nuclear accident: Critical reviews of past nuclear accidents and proposal for future planning. *Environment international*, 148:106379, 2021. 2

[12] Takashi Ohba, Arifumi Hasegawa, Yoshitaka Kohayagawa, Hisayoshi Kondo, and Gen Suzuki. Body surface contamination levels of residents under different evacuation scenarios after the fukushima daiichi nuclear power plant accident. *Health Physics*, 113(3):175–182, 2017. 2

[13] Dana A. Shea. Radiological dispersal devices: select issues in consequence management. Technical report, Library of Congress, Congressional Research Service, 2004. 3

[14] Frederick T. Harper, Stephen V. Musolino, and William B. Wente. Realistic radiological dispersal device hazard boundaries and ramifications for early consequence management decisions. *Health Physics*, 93(1):1–16, 2007. 3

[15] Anna Rae Green, Lorne Erhardt, Luke Lebel, M. John M. Duke, Trevor Jones, Dan White, and Debora Quayle. Overview of the full-scale radiological dispersal device field trials. *Health physics*, 110(5):403–417, 2016. 3

[16] M. Brooke Rogers, Richard Amlôt, and G. James Rubin. The impact of communication materials on public responses to a radiological dispersal device (RDD) attack. *Biosecurity and bioterrorism: biodefense strategy, practice, and science*, 11(1):49–58, 2013. 3

[17] Sean M. Brennan, Angela M. Mielke, David C. Torney, and Arthur B. Maccabe. Radiation detection with distributed sensor networks. *Computer*, 37(8):57–59, 2004. 4, 17

[18] R. J. Cooper, N. Abgrall, G. Aversano, M. S. Bandstra, D. Hellfeld, T. H. Joshi, V. Negut, B. J. Quiter, E. Rofors, M. Salathe, et al. Networked sensing for radiation detection, localization, and tracking. In *Journal of Physics: Conference Series*, volume 2586, page 012125. IOP Publishing, 2023. 4, 18, 74

[19] Robert J. Nemzek, Jared S. Dreicer, David C. Torney, and Tony T. Warnock. Distributed sensor networks for detection of mobile radioactive sources. *IEEE Transactions on Nuclear Science*, 51(4):1693–1700, 2004. 4, 17

[20] Michael J. King et al. An urban environment simulation framework for evaluating novel distributed radiation detection architectures. In *2010 IEEE International Conference on HST*, pages 446–452. IEEE, 2010. 4, 17, 74

[21] Kim Zetter. *Countdown to zero day: Stuxnet and the launch of the world's first digital weapon*. Crown, 2015. 4

[22] Mark Zeller. Myth or reality—does the aurora vulnerability pose a risk to my generator? In *2011 64th Annual Conference for Protective Relay Engineers*, pages 130–136. IEEE, 2011. 4

[23] Rex Min, Manish Bhardwaj, Seong-Hwan Cho, Eugene Shih, Amit Sinha, Alice Wang, and Anantha Chandrakasan. Low-power wireless sensor networks. In *VLSI Design 2001. Fourteenth International Conference on VLSI Design*, pages 205–210. IEEE, 2001. 4, 20

[24] Michael A Lones. Metaheuristics in nature-inspired algorithms. In *Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation*, pages 1419–1422, 2014. 7, 8, 28

[25] Alex Fraser, Donald Burnell, et al. *Computer models in genetics.* Mcgraw-Hill Book Co., New York., 1970. 8, 19

[26] Thomas Bäck and Hans-Paul Schwefel. An overview of evolutionary algorithms for parameter optimization. *Evolutionary computation*, 1(1):1–23, 1993. 8

[27] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of ICNN '95 - International Conference on Neural Networks*, volume 4, pages 1942–1948, 1995. 8, 19

[28] Y. Shi and R. Eberhart. A modified particle swarm optimizer. In *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No. 98TH8360)*, pages 69–73, 1998. 8, 19

[29] Marco Dorigo, Mauro Birattari, and Thomas Stutzle. Ant colony optimization. *IEEE computational intelligence magazine*, 1(4):28–39, 2006. 9

[30] Dervis Karaboga and Bahriye Akay. A comparative study of artificial bee colony algorithm. *Applied mathematics and computation*, 214(1):108–132, 2009. 9

[31] Xin-She Yang and Suash Deb. Cuckoo search via lévy flights. In *2009 World congress on nature & biologically inspired computing (NaBIC)*, pages 210–214. Ieee, 2009. 9

[32] C. A. Coello Coello. Evolutionary multi-objective optimization: a historical view of the field. *IEEE computational intelligence magazine*, 1(1):28–36, 2006. 10, 12, 19

[33] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002. 10, 12, 19, 52

[34] Vilfredo Pareto. *Cours d'économie politique*, volume 1. Librairie Droz, 1964. 10, 18

[35] Tapabrata Ray and KM Liew. A swarm metaphor for multiobjective design optimization. *Engineering optimization*, 34(2):141–153, 2002. 10

[36] J. Blank and K. Deb. pymoo: Multi-objective optimization in Python. *IEEE Access*, 8:89497–89509, 2020. 11, 52

[37] Konstantinos E Parsopoulos and Michael N Vrahatis. Particle swarm optimization method in multiobjective problems. In *Proceedings of the 2002 ACM symposium on Applied computing*, pages 603–607, 2002. 12, 19

[38] C.A. Coello Coello, G.T. Pulido, and M.S. Lechuga. Handling multiple objectives with particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 8(3):256–279, 2004. 12, 19

[39] Gordon M Dunning. *Radioactive Contamination of Certain Areas in the Pacific Ocean from Nuclear Tests: A Summary of the Data from the Radiological Surveys and Medical Examinations*. United States Atomic Energy Commission, 1957. 14

[40] Geoffrey G. Eichholz, Barry G. Wahlig, Gregory F. Powell, and T. F. Craft. Subsurface migration of radioactive waste materials by particulate transport. *Nuclear Technology*, 58(3):511–520, 1982. 14

[41] Seiya Nagao, Masaki Kanamori, Shinya Ochiai, Mutsuo Inoue, and Masayoshi Yamamoto. Migration behavior of ^{134}Cs and ^{137}Cs in the Niida river water in Fukushima prefecture, Japan during 2011–2012. *Journal of Radioanalytical and Nuclear Chemistry*, 303(2):1617–1621, 2015. 14

[42] Michelle Galloway, Andreas Zoglauer, Mark Amman, Steven E. Boggs, and Paul N. Luke. Simulation and detector response for the High Efficiency Multimode Imager. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 652(1):641–645, 2011. 15

[43] Jerry Towler, Bryan Krawiec, and Kevin Kochersberger. Radiation mapping in post-disaster environments using an autonomous helicopter. *Remote Sensing*, 4(7):1995–2015, 2012. 15

[44] Jinlu Han, Yaojin Xu, Long Di, and YangQuan Chen. Low-cost multi-UAV technologies for contour mapping of nuclear radiation field. *Journal of Intelligent & Robotic Systems*, 70(1):401–410, 2013. 15

[45] Peter G. Martin, Simon Kwong, Nick T. Smith, Yosuke Yamashiki, Oliver D. Payton, Freddie S. Russell-Pavier, John S. Fardoulis, David A. Richards, and Thomas B. Scott. 3d unmanned aerial vehicle radiation mapping for assessing contaminant distribution and mobility. *International journal of applied earth observation and geoinformation*, 52:12–19, 2016. 15

[46] D. Connor, Peter George Martin, and Thomas Bligh Scott. Airborne radiation mapping: overview and application of current and future aerial systems. *International journal of remote sensing*, 37(24):5953–5987, 2016. 15

[47] Ryan Pavlovsky, Andrew Haefner, Tenzing Joshi, Victor Negut, Keith McManus, Erika Suzuki, Ross Barnowski, and Kai Vetter. 3-D radiation mapping in real-time with the localization and mapping platform LAMP from unmanned aerial systems and man-portable configurations. *arXiv preprint arXiv:1901.05038*, 2018. 15, 16

[48] Nageswara S. V. Rao, Satyabrata Sen, Nicholas J. Prins, Daniel A. Cooper, Robert J. Ledoux, James B. Costales, Krzysztof Kamieniecki, Steven E. Korbly, Jeffrey K. Thompson, James Batcheler, et al. Network algorithms for detection of radiation sources. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 784:326–331, 2015. 15

[49] Andrew D. Nicholson, Daniel E. Archer, Irakli Garishvili, Ian R. Stewart, and Michael J. Willis. Characterization of gamma-ray background outside of the High Flux Isotope Reactor. *Journal of Radioanalytical and Nuclear Chemistry*, 318(1), 8 2018. 15

[50] R. Andres Cortez, Herbert G. Tanner, and Ron Lumia. Distributed robotic radiation mapping. In *Experimental Robotics*, pages 147–156. Springer, 2009. 15

[51] Tony H. Shin, Daniel T. Wakeford, and S. F. Nowicki. Multi-sensor optimal motion planning for radiological contamination surveys by using Prediction-Difference Maps. *Applied Sciences*, 12(11):5627, 2022. 15, 66

[52] Stuart Lloyd. Least squares quantization in PCM. *IEEE transactions on information theory*, 28(2):129–137, 1982. 16, 40

[53] Daniel Hellfeld, Paul Barton, Donald Gunter, Andrew Haefner, Lucian Mihailescu, and Kai Vetter. Omnidirectional 3D Gamma-ray imaging with a free-moving spherical active coded aperture. In *2017 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, pages 1–3, 2017. 16

[54] Andrew D. Nicholson, Irakli Garishvili, Douglas E. Peplow, Daniel E. Archer, William R. Ray, Mathew W. Swinney, Michael J. Willis, Gregory G. Davidson, Steven L. Cleveland, Bruce W. Patton, et al. Multiagency urban search experiment detector and algorithm test bed. *IEEE Transactions on Nuclear Science*, 64(7):1689–1695, 2017. 16

[55] Mark S. Bandstra, Brian J. Quiter, Joseph C. Curtis, Kyle J. Bilton, Tenzing H. Y. Joshi, Ross Meyer, Victor Negut, Kai Vetter, Daniel E. Archer, Donald E. Hornback, et al. Attribution of gamma-ray background collected by a mobile detector system to its surroundings using panoramic video. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 954:161126, 2020. 16, 17

[56] Mark S. Bandstra, Brian J. Quiter, Marco Salathe, Kyle J. Bilton, Joseph C. Curtis, Steven Goldenberg, and Tenzing H. Y. Joshi. Correlations between panoramic imagery and gamma-ray background in an urban area. *IEEE Transactions on Nuclear Science*, 68(12):2818–2834, 2021. 16, 17

[57] Mark S. Bandstra, Timothy J. Aucott, Erik Brubaker, Daniel H. Chivers, Reynold J. Cooper, Joseph C. Curtis, John R. Davis, Tenzing H. Joshi, John Kua, Ross Meyer, et al. RadMAP: The radiological multi-sensor analysis platform. *Nuclear Instruments*

and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 840:59–68, 2016. 16

[58] Joseph C. Curtis, Reynold J. Cooper, Tenzing H. Joshi, Bogdan Cosofret, Thomas Schmit, John Wright, Jonathan Rameau, Daisei Konno, Daniel Brown, Forrest Otsuka, et al. Simulation and validation of the Mobile Urban Radiation Search (MURS) gamma-ray detector response. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 954:161128, 2020. 17

[59] James M. Ghawaly Jr et al. Data for training and testing radiation detection algorithms in an urban environment. *Scientific data*, 7(1):328, 2020. 17

[60] Andrew D. Nicholson, Douglas E. Peplow, James M. Ghawaly, Michael J. Willis, and Daniel E. Archer. Generation of synthetic data for a radiation detection algorithm competition. *IEEE Transactions on Nuclear Science*, 67(8):1968–1975, 2020. 17

[61] Jason Hite, John Mattingly, Dan Archer, Michael Willis, Andrew Rowe, Kayleigh Bray, Jake Carter, and James Ghawaly. Localization of a radioactive source in an urban environment using Bayesian Metropolis methods. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 915:82–93, 2019. 18

[62] Dave Osthus et al. Tracking the location of a road-constrained radioactive source with a network of detectors. *Nuclear Instruments and Methods in Physics Research Section A*, 1039:166992, 2022. 18, 74

[63] A. M. Turing. I. — Computing Machinery and Intelligence. *Mind*, LIX(236):433–460, 10 1950. 19

[64] Carlos M Fonseca, Peter J Fleming, et al. Genetic algorithms for multiobjective optimization: formulationdiscussion and generalization. In *Icga*, volume 93, pages 416–423. Citeseer, 1993. 19, 41

[65] Tadahiko Murata, Hisao Ishibuchi, et al. MOGA: multi-objective genetic algorithms. In *IEEE international conference on evolutionary computation*, volume 1, pages 289–294. IEEE Piscataway, NJ, USA, 1995. 19, 41

[66] Eckart Zitzler and Lothar Thiele. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE transactions on Evolutionary Computation*, 3(4):257–271, 1999. 19

[67] Joshua D. Knowles and David W. Corne. Approximating the nondominated front using the Pareto Archived Evolution Strategy. *Evolutionary Computation*, 8(2):149–172, 2000. 19

[68] Robert R. Tenney and Nils R. Sandell. Detection with distributed sensors. *IEEE Transactions on Aerospace and Electronic systems*, (4):501–510, 1981. 20

[69] Diego Antolín, Nicolás Medrano, and Belén Calvo. Reliable lifespan evaluation of a remote environment monitoring system based on wireless sensor networks and global system for mobile communications. *Journal of Sensors*, 2016, 2016. 20

[70] Alec Woo, Terence Tong, and David Culler. Taming the underlying challenges of reliable multihop routing in sensor networks. In *Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 14–27, 2003. 20

[71] Jochen Schiller, Achim Liers, Hartmut Ritter, Rolf Winter, and Thiemo Voigt. Scatterweb-low power sensor nodes and energy aware routing. In *Proceedings of the 38th annual Hawaii international conference on system sciences*, pages 286c–286c. IEEE, 2005. 20

[72] Nor Azlina Bt Ab Aziz, Ammar W. Mohemmed, and Mohammad Yusoff Alias. A wireless sensor network coverage optimization algorithm based on particle swarm optimization and voronoi diagram. In *2009 international conference on networking, sensing and control*, pages 602–607. IEEE, 2009. 20

[73] Patrick N. Ngatchou, Warren L. J. Fox, and Mohamed A. El-Sharkawi. Distributed sensor placement with sequential particle swarm optimization. In *Proceedings 2005*

IEEE Swarm Intelligence Symposium, 2005. SIS 2005., pages 385–388. IEEE, 2005. 20

[74] Jianming Hu, Jingyan Song, Mingchen Zhang, and Xiaojing Kang. Topology optimization for urban traffic sensor network. *Tsinghua Science and Technology*, 13(2):229–236, 2008. 21

[75] Bushra Rashid and Mubashir Husain Rehmani. Applications of wireless sensor networks for urban areas: A survey. *Journal of network and computer applications*, 60:192–219, 2016. 21

[76] Mohamed Amine Kafi, Yacine Challal, Djamel Djenouri, Messaoud Doudou, Abdelmadjid Bouabdallah, and Nadjib Badache. A study of wireless sensor networks for urban traffic monitoring: applications and architectures. *Procedia computer science*, 19:617–626, 2013. 21

[77] Catherine L. Muller, Lee Chapman, CSB Grimmond, Duick T. Young, and Xiaoming Cai. Sensors and the city: a review of urban meteorological networks. *International Journal of Climatology*, 33(7):1585–1600, 2013. 21

[78] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968. 21, 44, 80

[79] Edsger W. Dijkstra et al. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959. 21, 45

[80] Bradley S. Stewart and Chelsea C. White III. Multiobjective A*. *Journal of the ACM (JACM)*, 38(4):775–814, 1991. 22, 45

[81] Lawrence Mandow and José-Luis Pérez de la Cruz. Multicriteria heuristic search. *European Journal of Operational Research*, 150(2):253–280, 2003. 22

[82] Lawrence Mandow, José-Luis Pérez de la Cruz, et al. A new approach to multiobjective A* search. In *IJCAI*, volume 8, 2005. 22

[83] L. Mandow and José-Luis Pérez de la Cruz. A heuristic search algorithm with lexicographic goals. *Engineering Applications of Artificial Intelligence*, 14(6):751–762, 2001. 22

[84] Francisco-Javier Pulido, Lawrence Mandow, and José-Luis Pérez de la Cruz. Dimensionality reduction in multiobjective shortest path search. *Computers & Operations Research*, 64:60–70, 2015. 22

[85] Alexander Lavin. A Pareto front-based multiobjective path planning algorithm. *arXiv preprint arXiv:1505.05947*, 2015. 22

[86] Kossar Jeddisaravi, Reza Javanmard Alitappeh, and Frederico G. Guimarães. Multi-objective mobile robot path planning based on A* search. In *2016 6th International Conference on Computer and Knowledge Engineering (ICCKE)*, pages 7–12. IEEE, 2016. 22, 113

[87] Zhenan He, Gary G. Yen, and Jun Zhang. Fuzzy-based Pareto optimality for many-objective evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 18(2):269–285, 2013. 22

[88] A. R. Soltani and T. Fernando. A fuzzy based multi-objective path planning of construction sites. *Automation in construction*, 13(6):717–734, 2004. 22, 113

[89] Fatin H. Ajeil, Ibraheem Kasim Ibraheem, Mouayad A. Sahib, and Amjad J. Humaidi. Multi-objective path planning of an autonomous mobile robot using hybrid PSO-MFB optimization algorithm. *Applied Soft Computing*, 89:106076, 2020. 23

[90] Ellips Masehian and Davoud Sedighizadeh. A multi-objective PSO-based algorithm for robot path planning. In *2010 IEEE international conference on industrial technology*, pages 465–470. IEEE, 2010. 23

[91] Yong Zhang, Dun-Wei Gong, and Jian-Hua Zhang. Robot path planning in uncertain environment using multi-objective particle swarm optimization. *Neurocomputing*, 103:172–185, 2013. 23, 24

[92] Jim Pugh and Alcherio Martinoli. Inspiring and modeling multi-robot search with particle swarm optimization. In *2007 IEEE swarm intelligence symposium*, pages 332–339. IEEE, 2007. 23, 69

[93] Mansoor Davoodi, Fatemeh Panahi, Ali Mohades, and Seyed Naser Hashemi. Multi-objective path planning in discrete space. *Applied Soft Computing*, 13(1):709–720, 2013. 24

[94] Faez Ahmed and Kalyanmoy Deb. Multi-objective optimal path planning using elitist non-dominated sorting genetic algorithms. *Soft Computing*, 17:1283–1299, 2013. 24

[95] Frank Mascarich, Christos Papachristos, Taylor Wilson, and Kostas Alexis. Distributed radiation field estimation and informative path planning for nuclear environment characterization. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 2318–2324. IEEE, 2019. 24

[96] Andrey A. Galyaev, Alexander V. Dobrovidov, Pavel V. Lysenko, Mikhail E. Shaikin, and Victor P. Yakhno. Path planning in threat environment for UUV with non-uniform radiation pattern. *Sensors*, 20(7):2076, 2020. 24

[97] Louis Anthony Cox, Jr. Game theory and risk analysis. *Risk Analysis: An International Journal*, 29(8):1062–1068, 2009. 24

[98] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. 25

[99] Jonathan Uesato, Brendan O'donoghue, Pushmeet Kohli, and Aaron Oord. Adversarial risk and the dangers of evaluating against weak attacks. In *International Conference on Machine Learning*, pages 5025–5034. PMLR, 2018. 25

[100] Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, Aleksander Madry, and Alexey Kurakin. On evaluating adversarial robustness. *arXiv preprint arXiv:1902.06705*, 2019. 25

[101] David Rios Insua, Jesus Rios, and David Banks. Adversarial risk analysis. *Journal of the American Statistical Association*, 104(486):841–854, 2009. 25

[102] Jesus Rios and David Rios Insua. Adversarial risk analysis for counterterrorism modeling. *Risk Analysis: An International Journal*, 32(5):894–915, 2012. 25

[103] Trung Thanh Nguyen, Shengxiang Yang, and Juergen Branke. Evolutionary dynamic optimization: A survey of the state of the art. *Swarm and Evolutionary Computation*, 6:1–24, 2012. 26

[104] David E. Goldberg and Robert E. Smith. Nonstationary function optimization using genetic algorithms with dominance and diploidy. In *Genetic algorithms and their applications: proceedings of the second International Conference on Genetic Algorithms: July 28-31, 1987 at the Massachusetts Institute of Technology, Cambridge, MA*, 1987. 26

[105] Shengxiang Yang. Non-stationary problem optimization using the primal-dual genetic algorithm. In *The 2003 Congress on Evolutionary Computation, 2003. CEC'03.*, volume 3, pages 2246–2253. IEEE, 2003. 26

[106] Rui Mendes and Arvind S. Mohais. DynDE: a differential evolution for dynamic optimization problems. In *2005 IEEE congress on evolutionary computation*, volume 3, pages 2808–2815. IEEE, 2005. 27

[107] Mitchell A. Potter and Kenneth A. De Jong. A cooperative coevolutionary approach to function optimization. In *International conference on parallel problem solving from nature*, pages 249–257. Springer, 1994. 27

[108] Xiaoliang Ma, Xiaodong Li, Qingfu Zhang, Ke Tang, Zhengping Liang, Weixin Xie, and Zexuan Zhu. A survey on cooperative co-evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 23(3):421–441, 2018. 27

[109] Luis Miguel Antonio and Carlos A. Coello Coello. Coevolutionary multiobjective evolutionary algorithms: Survey of the state-of-the-art. *IEEE Transactions on Evolutionary Computation*, 22(6):851–865, 2017. 27

[110] C. A. Coello Coello and Margarita Reyes Sierra. A coevolutionary multi-objective evolutionary algorithm. In *The 2003 Congress on Evolutionary Computation, 2003. CEC'03.*, volume 1, pages 482–489. IEEE, 2003. 27

[111] Mary Lou Maher and Josiah Poon. Modeling design exploration as co-evolution. *Computer-Aided Civil and Infrastructure Engineering*, 11(3):195–209, 1996. 27

[112] Mary Maher and Hsien-Hui Tang. Co-evolution as a computational and cognitive model of design. *Research in Engineering design*, 14:47–64, 2003. 28

[113] Kwee-Bo Sim, Dong-Wook Lee, and Ji-Yoon Kim. Game theory based coevolutionary algorithm: a new computational coevolutionary approach. *International Journal of Control, Automation, and Systems*, 2(4):463–474, 2004. 28

[114] Esmaeil Atashpaz-Gargari and Caro Lucas. Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. In *2007 IEEE congress on evolutionary computation*, pages 4661–4667. Ieee, 2007. 28

[115] Chi-Keong Goh and Kay Chen Tan. A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 13(1):103–127, 2008. 28

[116] Yousef Sharafi, Mojtaba Ahmadieh Khanesar, and Mohammad Teshnehlab. COOA: Competitive optimization algorithm. *Swarm and Evolutionary Computation*, 30:39–63, 2016. 28

[117] Edmund Burke, Graham Kendall, Jim Newall, Emma Hart, Peter Ross, and Sonia Schulenburg. *Handbook of Meta-heuristics*, chapter Hyper-Heuristics: An Emerging Direction in Modern Search Technology, pages 457–474. 01 2003. 29

[118] Taher Chegini, Hong-Yi Li, and L. Ruby Leung. HyRiver: Hydroclimate Data Retriever. *Journal of Open Source Software*, 6(66):1–3, 10 2021. 33

[119] J. D. Hunter. Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. 33

[120] Paul E. Bieringer et al. Large Eddy Simulation (LES) based system for producing coupled urban and indoor airborne contaminant transport and dispersion solutions. In *18th International Conference on Harmonisation within Atmospheric Dispersion Modelling for Regulatory Purposes*, 2017. 34, 62

[121] Iowa State University. Iowa Environmental Mesonet. https://mesonet.agron.iastate.edu/sites/locate.php, 2022. 34

[122] Philip Isola et al. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 34

[123] Jun-Yan Zhu et al. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 35

[124] Teofilo F. Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical computer science*, 38:293–306, 1985. 40

[125] Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 41, 135

[126] Gunter Dueck. New optimization heuristics: The great deluge algorithm and the record-to-record travel. *Journal of Computational physics*, 104(1):86–92, 1993. 44

[127] Chi Tung Tung and Kim Lin Chew. A multicriteria Pareto-optimal path algorithm. *European Journal of Operational Research*, 62(2):203–209, 1992. 45

[128] David M. Raji. Applied multi-objective modelling & optimization [slides]. Technical report, Los Alamos National Laboratory, 9 2022. 68, 71, 73

[129] Jürgen Branke, Kalyanmoy Deb, Henning Dierolf, and Matthias Osswald. Finding knees in multi-objective optimization. In *Parallel Problem Solving from Nature-PPSN VIII: 8th International Conference, Birmingham, UK, September 18-22, 2004. Proceedings 8*, pages 722–731. Springer, 2004. 70

[130] Nicolas Abgrall and Reynold Cooper. Urban Deployment Model (UDM) v1. 0. Technical report, Lawrence Berkeley National Laboratory (LBNL), Berkeley, CA (United States), 2019. 76

[131] OpenStreetMap contributors. Planet dump [Chicago] retrieved from https://planet.osm.org . https://www.openstreetmap.org, 2017. 76

[132] Chicago Data Portal. https://data.cityofchicago.org/. Accessed: April 2023. 77

[133] CMAP Data Hub: Community Data Snapshots. https://www.cmap.illinois.gov/data/community-snapshots. Accessed: June 2023. 77

[134] Things to do in Chicago. https://www.tripadvisor.com/Attractions-g35805-Activities-Chicago_Illinois.html. Accessed: 2023-12-30. 79, 144

[135] Zach Long. The 32 best Chicago attractions. https://www.timeout.com/chicago/things-to-do/chicago-attractions-the-best-sights-and-attractions-in-chicago. Accessed: 2023-12-30. 79, 144

[136] Cortney Fries. 28 fun things to do in Chicago. https://travel.usnews.com/Chicago_IL/Things_To_Do/. Accessed: 2023-12-30. 79, 144

[137] Sushmita Dutta. 15 best places to visit in Chicago that add charm to your American vacay! https://traveltriangle.com/blog/places-to-visit-in-chicago/. Accessed: 2023-12-30. 79, 144

[138] Alex Schultz and Fiona Fiorentino. 20 top tourist attractions in Chicago. https://www.touropia.com/tourist-attractions-in-chicago/. Accessed: 2023-12-30. 79, 144

[139] Thomas Bartz-Beielstein, Jürgen Branke, Jörn Mehnen, and Olaf Mersmann. Evolutionary algorithms. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 4(3):178–195, 2014. 86

[140] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11):559–572, 1901. 86

[141] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 111

[142] Albert Bifet and Ricard Gavalda. Learning from time-changing data with adaptive windowing. In *Proceedings of the 2007 SIAM international conference on data mining*, pages 443–448. SIAM, 2007. 114

[143] Christoph Raab, Moritz Heusinger, and Frank-Michael Schleif. Reactive soft prototype computing for concept drift streams. *Neurocomputing*, 416:340–351, 2020. 114

[144] Peter A Abrams. The evolution of predator-prey interactions: theory and evidence. *Annual Review of Ecology and Systematics*, 31(1):79–105, 2000. 116

[145] Bryan T Grenfell, Oliver G Pybus, Julia R Gog, James LN Wood, Janet M Daly, Jenny A Mumford, and Edward C Holmes. Unifying the epidemiological and evolutionary dynamics of pathogens. *science*, 303(5656):327–332, 2004. 116

[146] Eric Silva, Karen Panetta, and Sos Agaian. Quantifying image similarity using measure of enhancement by entropy. *Proceedings of SPIE - The International Society for Optical Engineering*, 6579, 05 2007. 135

Appendix

Appendix A

Reconstruction Method Comparisons

For this evaluation, networks are trained using a dataset of cropped deposition maps to act as the target domain and corresponding output maps from the modeling framework to serve as the source domain. For Region A, 900 samples are used for training and 100 for validation; for Region B, 2146 training and 239 validation samples are used (10% validation size in both regions). These are the same regions as those defined in Section 4.1.1.

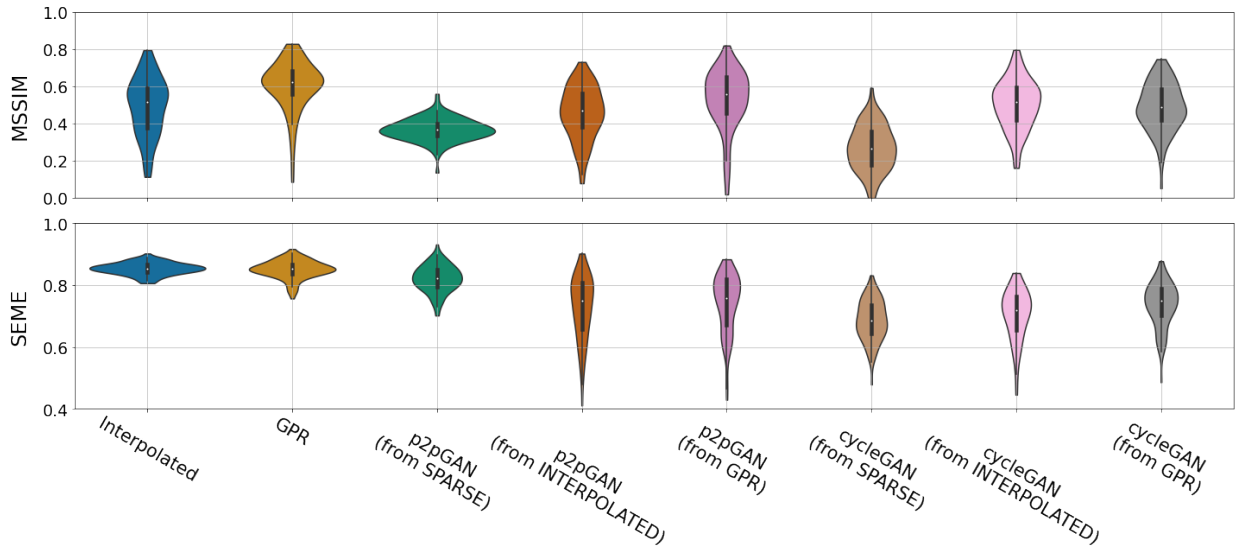
In all evaluations, a quantity of 25 nodes comprise the network. The reconstruction fitness is quantified with respect to ground truth using MSSIM (mean structural similarity, [125]) and SEME (similarity enhancement by entropy, [146]). These metrics both are more robust ways to measure the similarity between a source and approximation image in comparison to metrics such as L1 or L2. Their values range between 0.0 and 1.0, with higher values representing closer matches between the compared images.

Reconstructions are generated by the trained models using the validation source domain data, and summaries of the MSSIM and SEME density results of these are visualized in Fig. A.1. Also included here are the quantitative results of the base reconstruction methods for comparison.

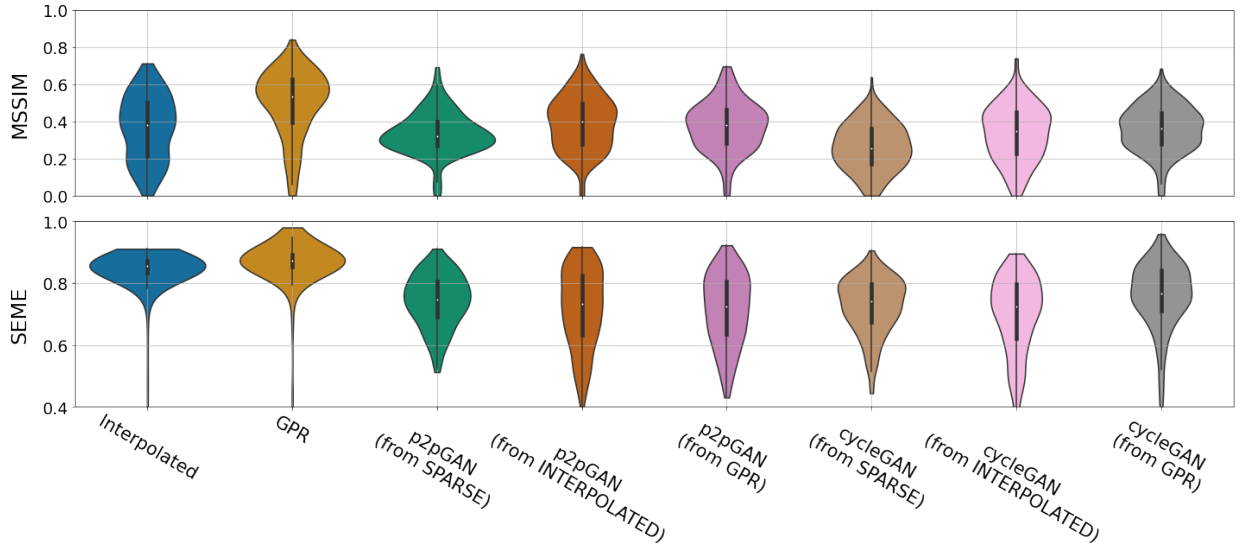
It is apparent from these metric density plots that none of the deep generative methods examined perform quite as well quantitatively as interpolation or GPR. Some of the generative models exhibit approximate parity with the base methods for one metric but are inferior when comparing the other, as is the case for the p2pGAN model generating from interpolated data.

One plausible explanation for the relatively inferior performance among the generative models is that the method for cropping subsections of the simulated environment based on the subregions with the highest deposition concentrations makes the task of learning a terrain-dependent transfer function between the input intensity measurements and the ground truth much more difficult.

In addition, there is an observable general reduction in median across the board when moving from evaluation in Region A to Region B. This shift in context also results in widening of the metric distributions for all reconstruction methods tested. Both phenomena can likely be explained as a consequence of the increased terrain variability in Region B as compared to that of Region A; this difference is quantified in Section 4.1.1.



(a) Region A, 100 evaluations



(b) Region B, 239 evaluations

Figure A.1: Comparison of performance distributions for various reconstruction methods. Evaluations are conducted for networks comprised of 25 nodes each, placed in topography using farthest-first traversal in two dimensions.

Appendix B

Pseudocode for FPA* and its Heuristics

The algorithms described in Section 4.3 are presented here with detailed pseudocode. Included in these are the computational methods used to calculate the topography gradient cost and the exposure cost heuristics, detailed in Alg. 1 and Alg. 2, respectively; the pseudocode of the FPA* algorithm itself is provided in Alg. 3.

Algorithm 1 Topography Gradient Cost Heuristic Algorithm

let *source_node*, *destination_node*, *topography_map* ← *inputs*;
let *gradient_magnitude_map* ← add in quadrature the X and Y direction gradient maps of the *topography_map*;
let *gradient_magnitude_steps* ← sorted unique values of *gradient_magnitude_map* greater than value at *source_node*;
for *this_gradient_magnitude* **in** *gradient_magnitude_steps* **do**
    let *masked_coordinates* ← set of coordinates *gradient_magnitude_map* ≤ *this_gradient_magnitude*;
    let *connected_set* ← breadth-first search of *masked_coordinates* beginning with *source_node*;
    if *destination_node* **in** *connected_set* **then**
        break;
    end if
end for
return *this_gradient_magnitude*, *connected_set*;

Algorithm 2 Exposure Cost Heuristic Algorithm

let *source_node*, *destination_node*, *intensity_map* ← *inputs*;
Initialize *open_set* containing *source_node* with a *g_value* of 0 and empty *closed_set*;
while length(*open_set*) > 0 **do**
    let *active_node* ← node with minimum *g_value* in *open_set*;
    open_set.pop(*active_node*);
    closed_set.append(*active_node*);
    if *active_node* == *destination_node* **then**
        break;
    end if
    let *valid_neighbor_set* ← subset of nodes adjacent to *active_node* that exist and are not in *open_set* or *closed_set*;
    for *valid_neighbor* **in** *valid_neighbor_set* **do**
        let *valid_neighbor*.*g_value* ← *active_node*.*g_value* + *intensity_map*[*valid_neighbor*];
        open_set.append(*valid_neighbor*);
    end for
end while
return *closed_set*;

Algorithm 3 Fuzzy Pareto A*

```
1: let source_node, destination_node, objective_weights, environment_maps ← inputs ;
2: Initialize open_set containing source_node and empty closed_set ;
3: while length(open_set) > 0 do
4:     for this_node in open_set do
5:         Calculate number of times and for which objective this_node is dominated by others in open_set ;
6:         Take weighted sum of this_node's objective-specific domination numbers using objective_weights ;
7:     end for
8:     let active_node ← node with minimum weighted sum in domination matrix ;
9:     open_set .pop( active_node ) ;
10:    closed_set .append( active_node ) ;
11:    if active_node == destination_node then
12:        break ;
13:    end if
14:    let neighbor_set ← subset of nodes adjacent to active_node ;
15:    for neighbor in neighbor_set do
16:        Update cumulative objective costs for neighbor (if applicable) ;
17:        if neighbor not in open_set or closed_set then
18:            open_set .append( neighbor ) ;
19:        end if
20:    end for
21: end while
22: return closed_set ;
```

Appendix C

Optimal Path Samples from FPA*

Included here are visualizations of sample optimal paths for additional optimization procedures in the manner described in Section 4.3. These two additional runs are also set in Region B, but focus on different subregions, radioactive source terms, and source and destination points than the run ***B-3*** reviewed in that section.

C.1 Output Run *B-103*

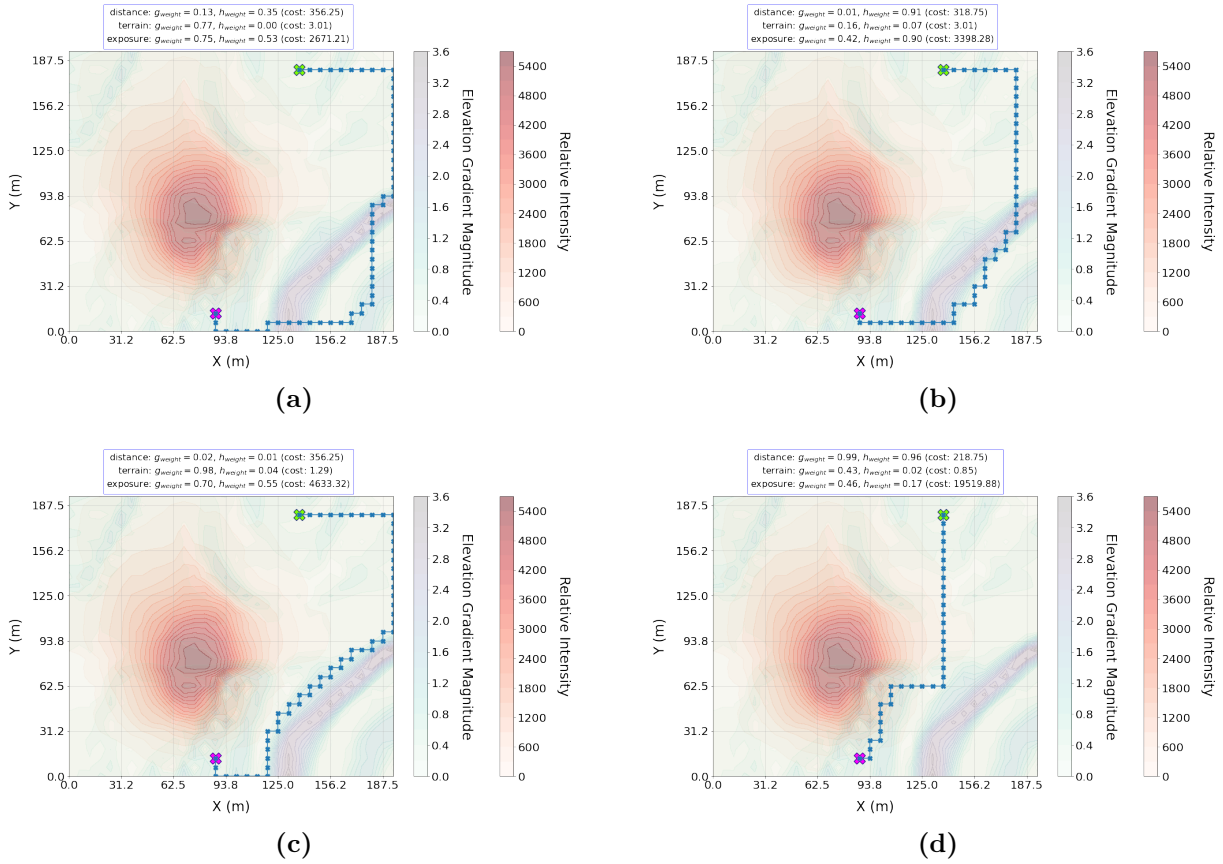


Figure C.1: Selection of path samples using optimized parameterizations for FPA* from run ***B-103***. Path shown in blue, source and destination of path displayed in fuchsia and lime green, respectively.

C.2 Output Run *B-109*

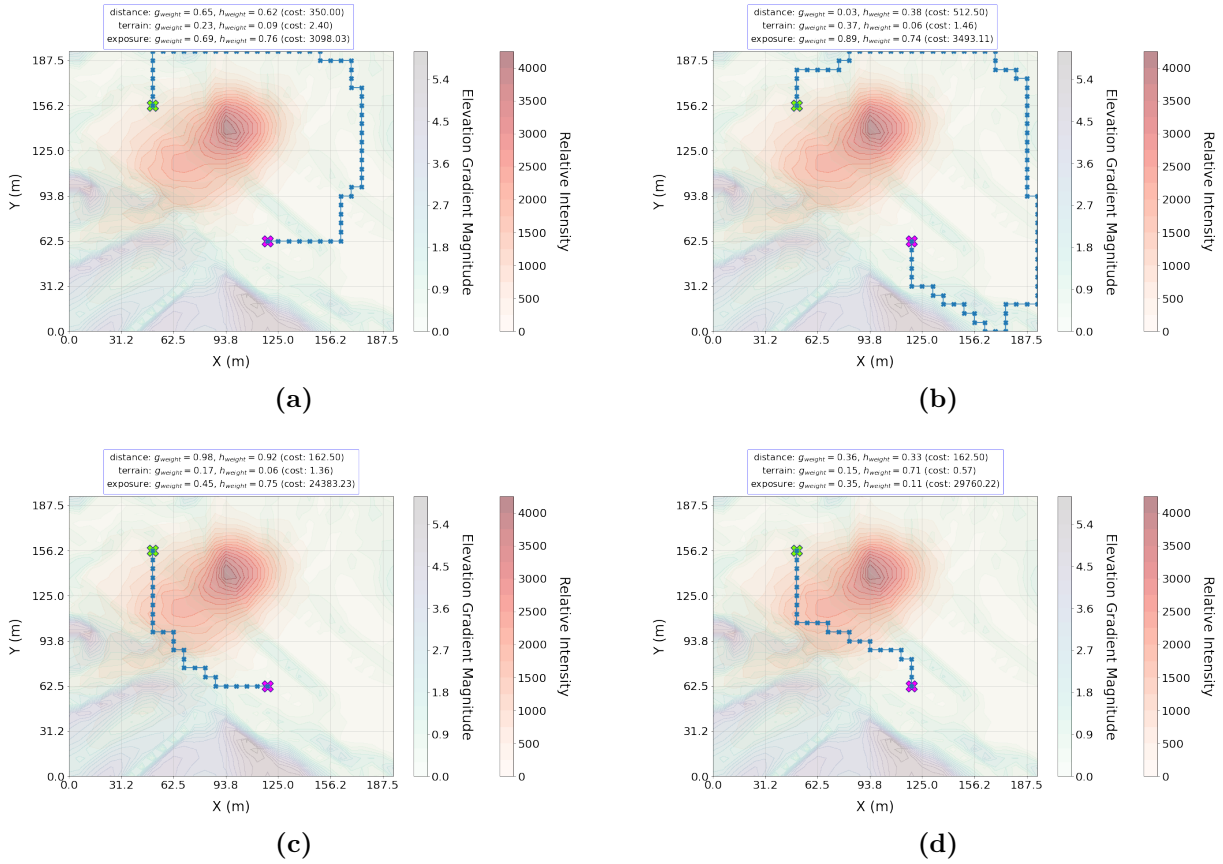


Figure C.2: Selection of path samples using optimized parameterizations for FPA* from run ***B-109***. Path shown in blue, source and destination of path displayed in fuchsia and lime green, respectively.

Appendix D

Adversarial Targets

With the aim of adequately modeling potential targets of adversaries in the course of the work described in Section 5.4, a list of locations of interest was developed. This list includes popular tourist destinations as well as other high-traffic points and was compiled based on a review of several guide websites [134, 135, 136, 137, 138].

Table D.1: List of target locations modeled as destinations of adversaries. Priority metric is determined by frequency of occurrence of destination in sources and is used to inform the chances of selection in path generation (lower priority numbers get selected more often).

Location Name	**Priority**	**Lon. Center**	**Lat. Center**
Millennium Park/"The Bean"	0	-87.62290	41.88427
Art Institute of Chicago	0	-87.62428	41.87956
Lincoln Park Zoo*	0	-87.63290	41.92114
360 Chicago (observation deck)	0	-87.62328	41.89906
Chicago River*	0	-87.61880	41.88858
Soldier Field	0	-87.61671	41.86251
Magnificent Mile*	1	-87.62418	41.89516
Navy Pier	1	-87.61005	41.89211
Field Museum	1	-87.61703	41.86558
Museum of Science and Industry	1	-87.58293	41.79165
Wrigley Field	1	-87.65528	41.94823
Shedd Aquarium	1	-87.61402	41.86699
Grant Park/Buckingham Fountain	2	-87.62061	41.87581
Willis Tower (skydeck)	2	-87.63596	41.87881
Chicago Theatre	2	-87.62760	41.88551
Garfield Park Conservatory	2	-87.71705	41.88628
Maggie Daley Park*	3	-87.61818	41.88264
North Avenue Beach*	3	-87.62635	41.91959
Second City	3	-87.63536	41.91135
DuSable Black History Museum	3	-87.60723	41.79194
Green Mill	3	-87.65992	41.96919
Chicago Cultural Center	3	-87.62495	41.88375
Robie House	3	-87.59594	41.78978
606 Trails*	3	-87.69434	41.91398
Lakefront Trails*	3	-87.61449	41.87560
Chicago History Museum	3	-87.63169	41.91179

* Exact coordinates for OD pair generation are sampled from area around center.

Appendix E

Pseudocode for IIA*

The algorithm described in Section 5.4.1, termed Iterative Induction A* (IIA*), is described here with pseudocode in Alg. 4.

Algorithm 4 Iterative Induction A*

```
1: let source, destination, graph, cost_map, threshold, max_iterations ← inputs;
2: Initialize empty path_histories and cost_histories;
3: let path ← A*(graph, source, destination);
4: let path_costs ← [cost_map[step] for step in path];
5: let cost ← sum(path_costs);
6: Append path to path_histories and cost to cost_histories;
7: let index ← 0;
8: while cost > threshold and index < max_iterations do
9:     let avg_c, std_c ← avg(path_costs), std(path_costs);
10:    let graph ← induced subgraph of graph without vertices in path with cost > avg_c + std_c;
11:    if graph is disjoint between source and destination then
12:        break;
13:    end if
14:    Repeat l.3-l.6;
15: end while
16: return path_histories, cost_histories;
```

Appendix F

Quantitative Adversarial Analysis: Complete Figures

This chapter of the Appendix provides a complete catalogue of results of the quantitative detector network configuration analyses described in Section 5.5.1.

F.1 50 Detectors

F.1.1 Route Dataset: 0% Targeting

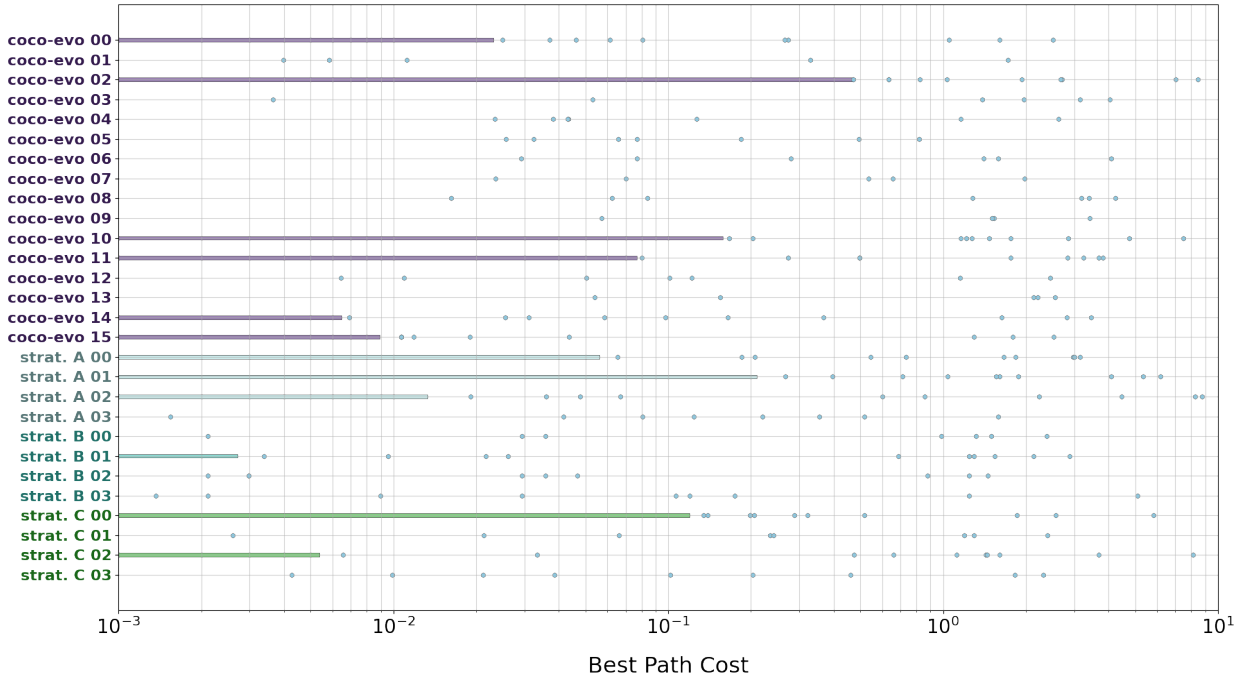


Figure F.1: Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 50 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 0% targeted routes.

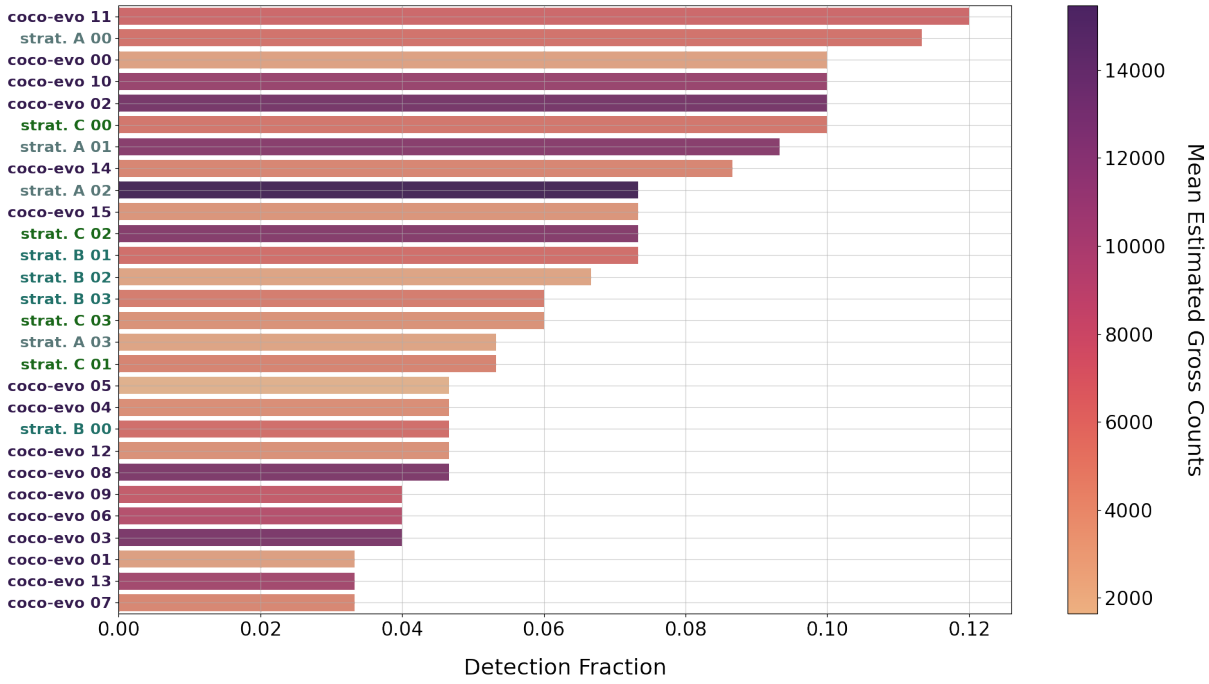


Figure F.2: Comparison of summarized IIA* best path cost results across 50-detector network configurations using route dataset of 0% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations.

F.1.2 Route Dataset: 5% Targeting

Quantitative results included in Section 5.5.1.

F.1.3 Route Dataset: 50% Targeting

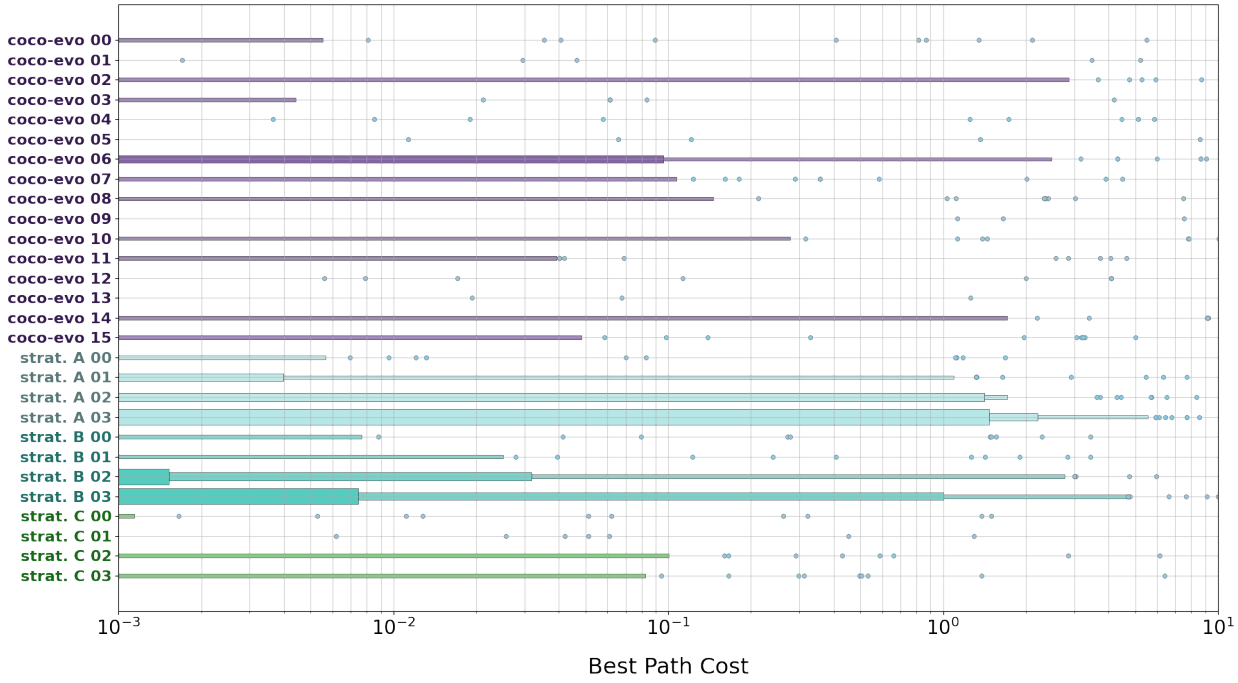


Figure F.3: Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 50 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 50% targeted routes.

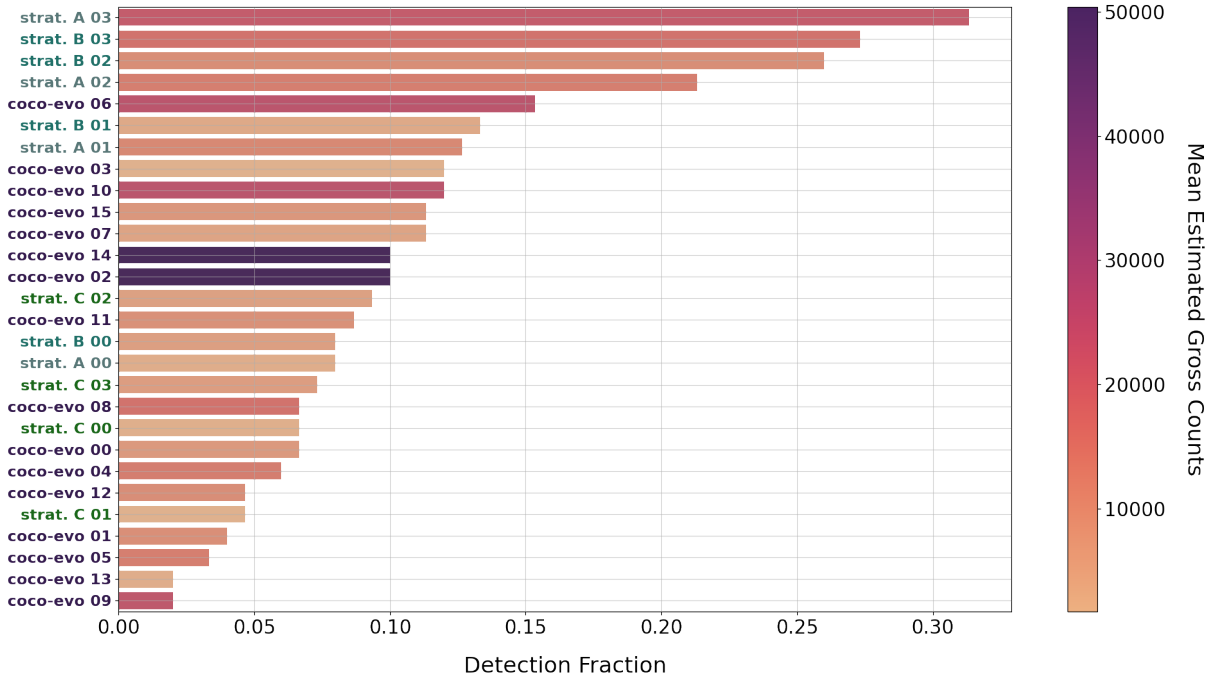


Figure F.4: Comparison of summarized IIA* best path cost results across 50-detector network configurations using route dataset of 50% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations.

F.1.4 Route Dataset: 100% Targeting

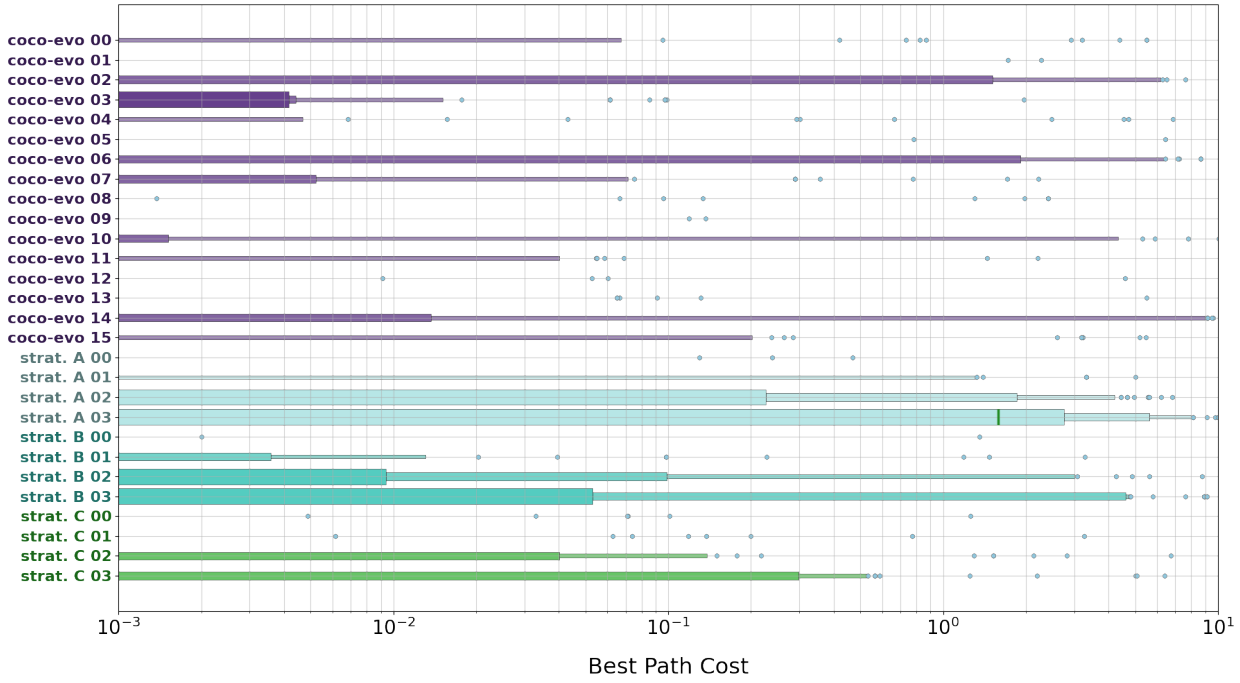


Figure F.5: Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 50 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 100% targeted routes.

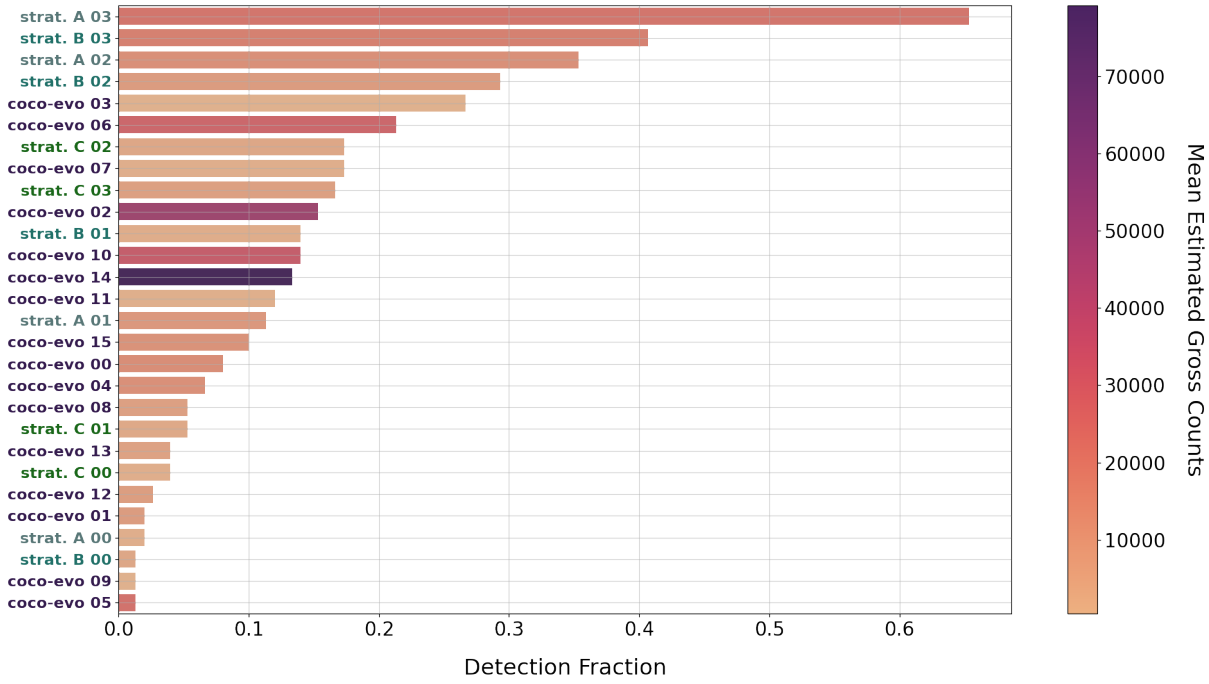


Figure F.6: Comparison of summarized IIA* best path cost results across 50-detector network configurations using route dataset of 100% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations.

F.2 100 Detectors

F.2.1 Route Dataset: 0% Targeting

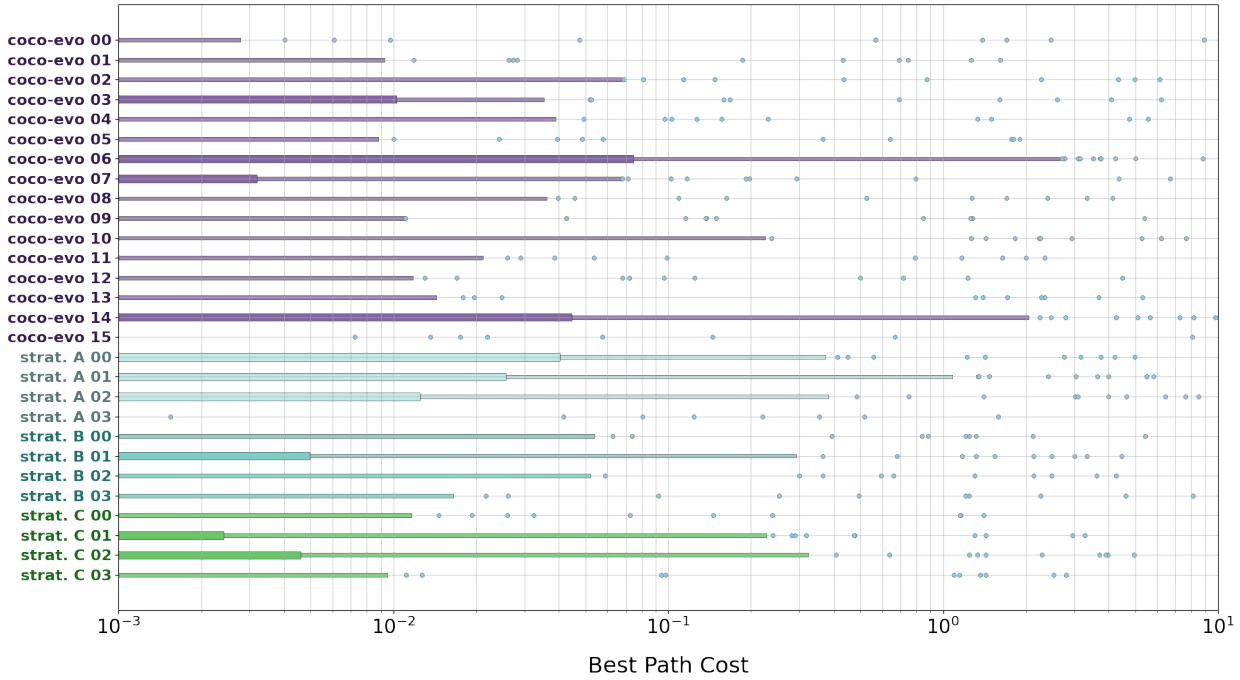


Figure F.7: Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 100 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 0% targeted routes.

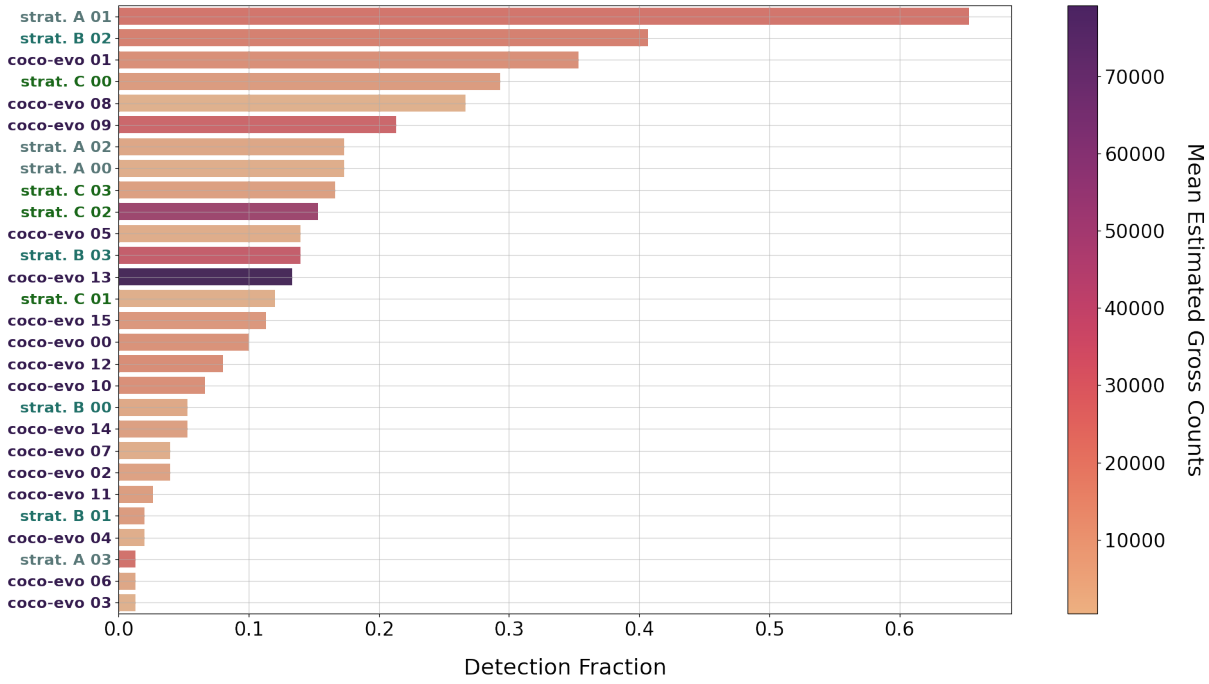


Figure F.8: Comparison of summarized IIA* best path cost results across 100-detector network configurations using route dataset of 0% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations.

F.2.2 Route Dataset: 5% Targeting

Quantitative results included in Section 5.5.1.

F.2.3 Route Dataset: 50% Targeting

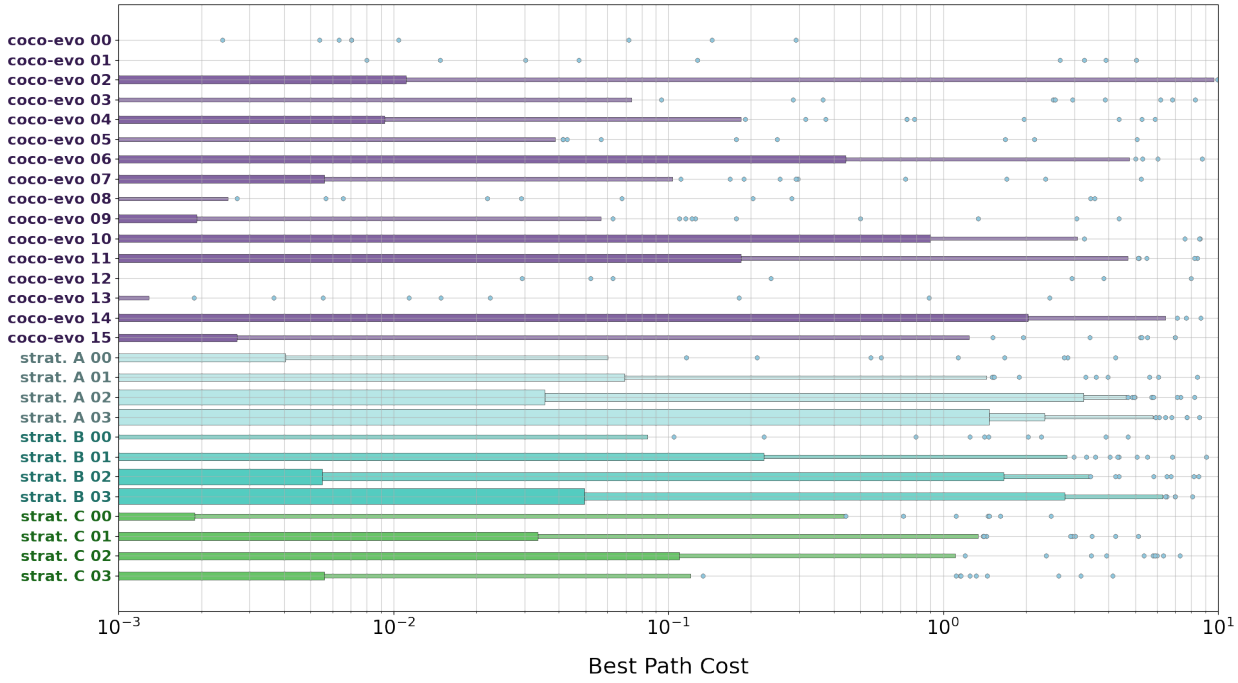


Figure F.9: Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 100 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 50% targeted routes.

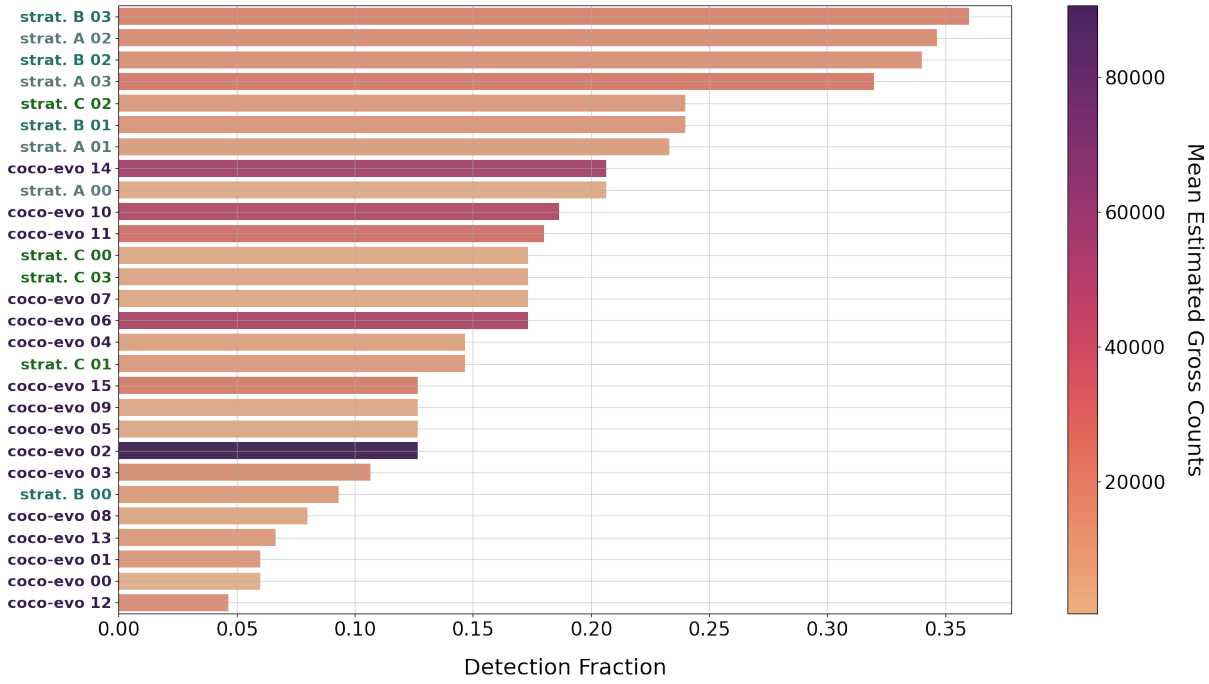


Figure F.10: Comparison of summarized IIA* best path cost results across 100-detector network configurations using route dataset of 50% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations.

F.2.4 Route Dataset: 100% Targeting

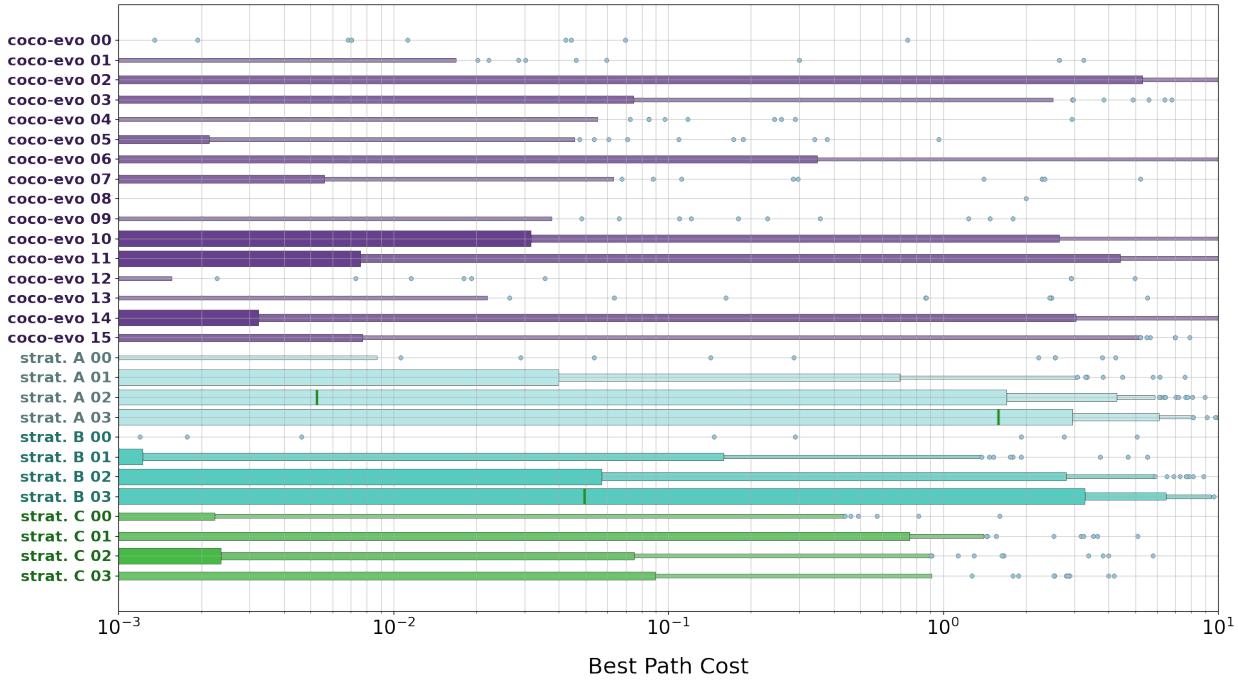


Figure F.11: Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 100 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 100% targeted routes.

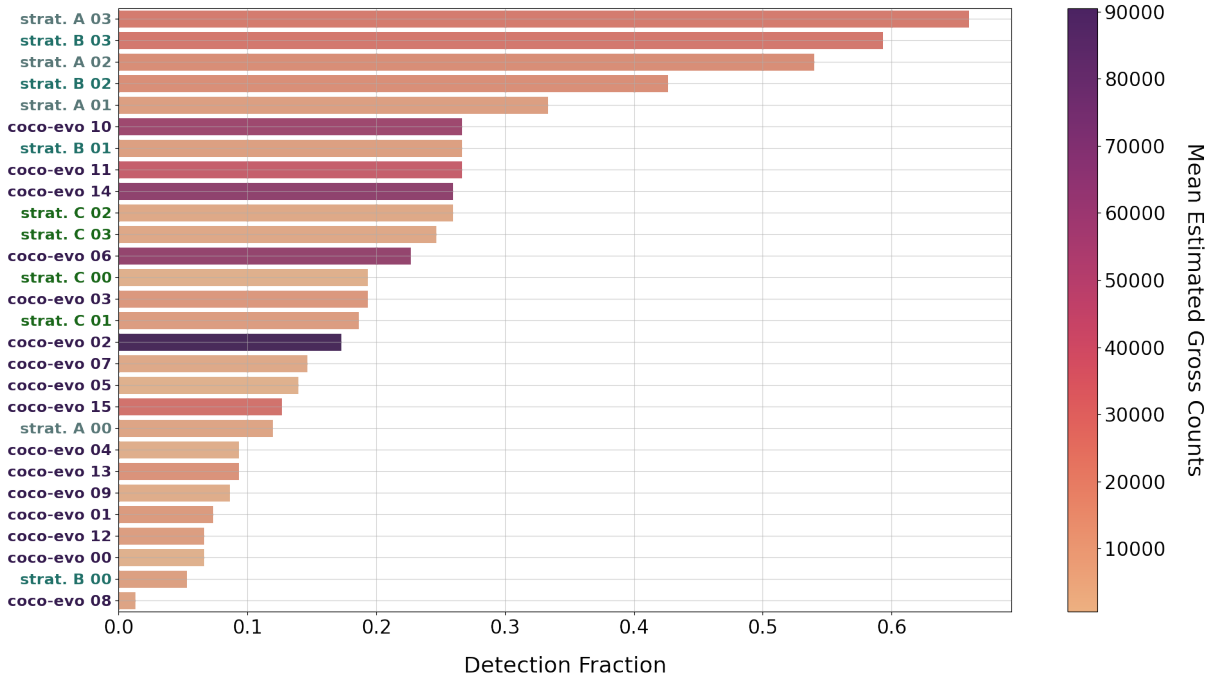


Figure F.12: Comparison of summarized IIA* best path cost results across 100-detector network configurations using route dataset of 100% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations.

F.3 200 Detectors

F.3.1 Route Dataset: 0% Targeting

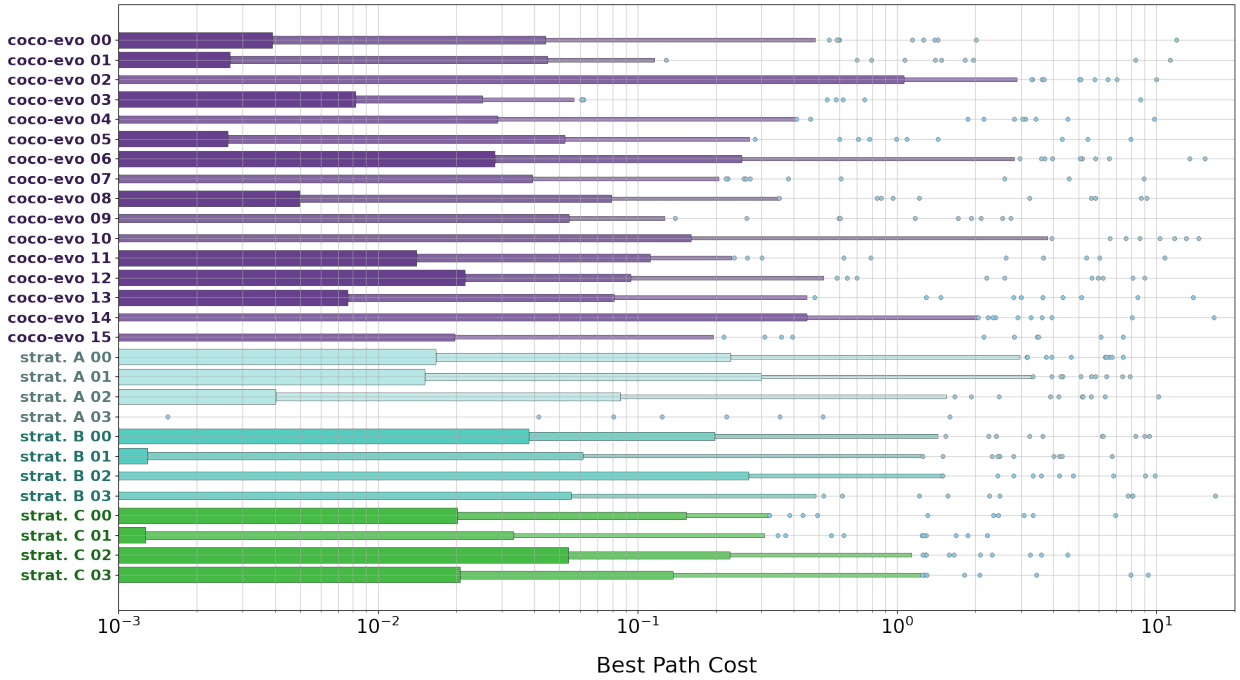


Figure F.13: Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 200 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 0% targeted routes.

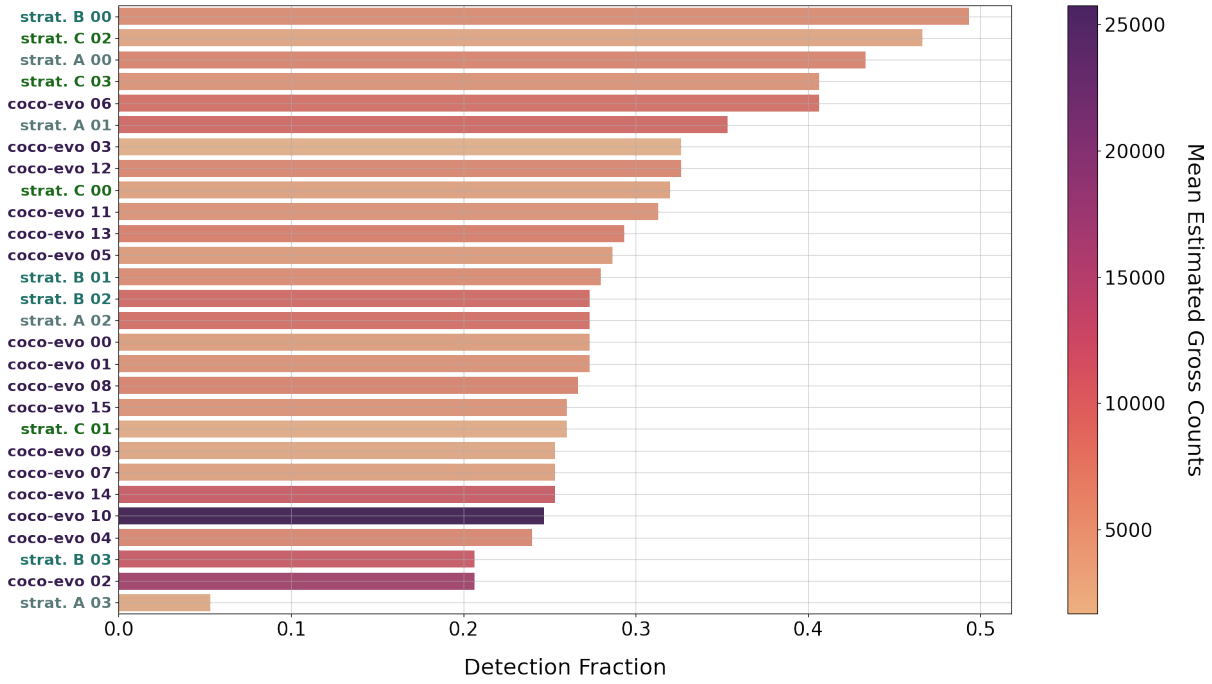


Figure F.14: Comparison of summarized IIA* best path cost results across 200-detector network configurations using route dataset of 0% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations.

F.3.2 Route Dataset: 5% Targeting

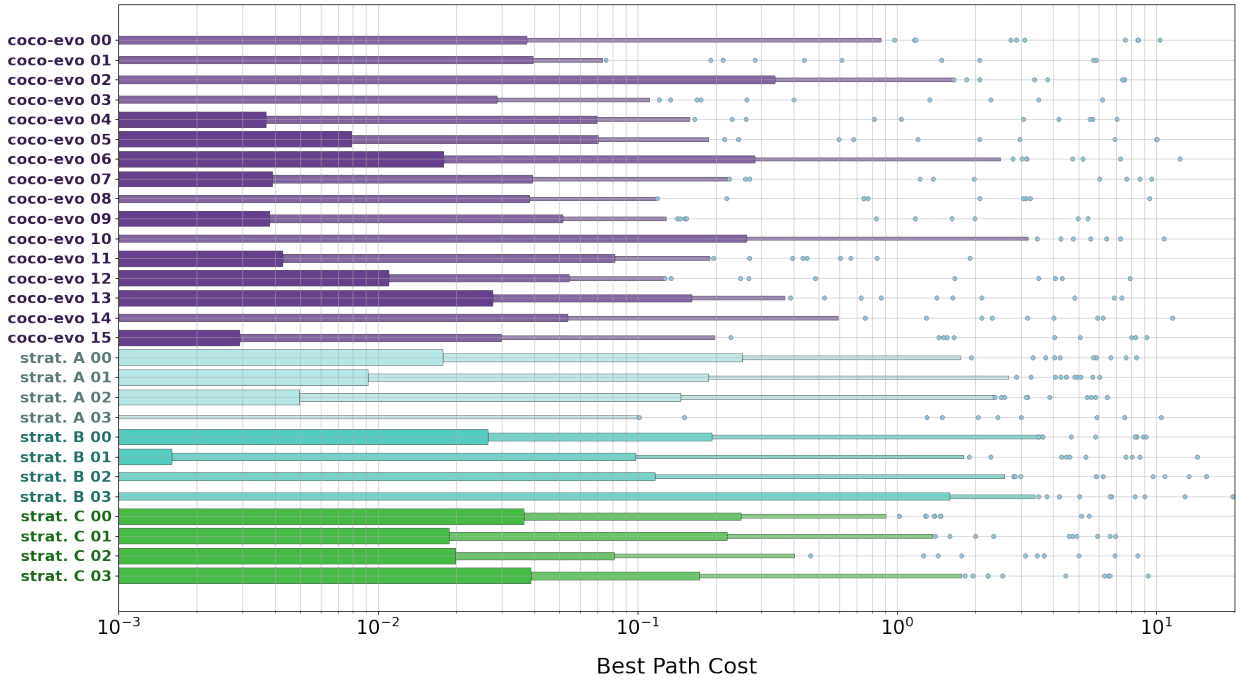


Figure F.15: Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 200 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 5% targeted routes.

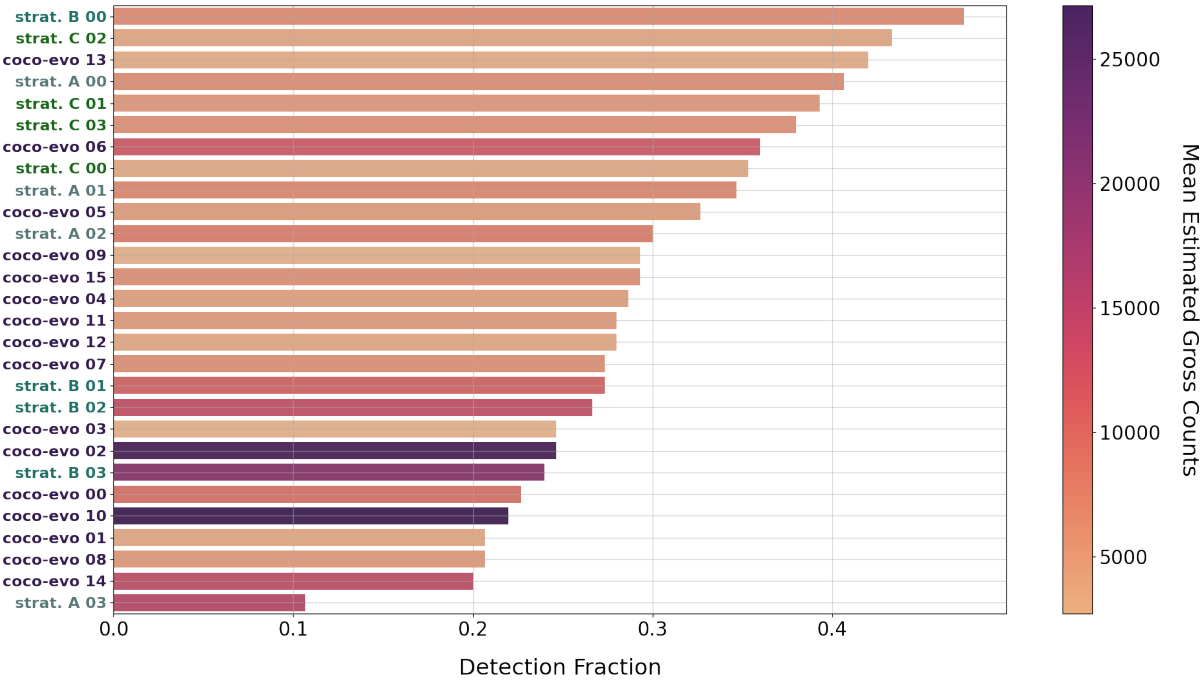


Figure F.16: Comparison of summarized IIA* best path cost results across 200-detector network configurations using route dataset of 5% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations.

F.3.3 Route Dataset: 50% Targeting

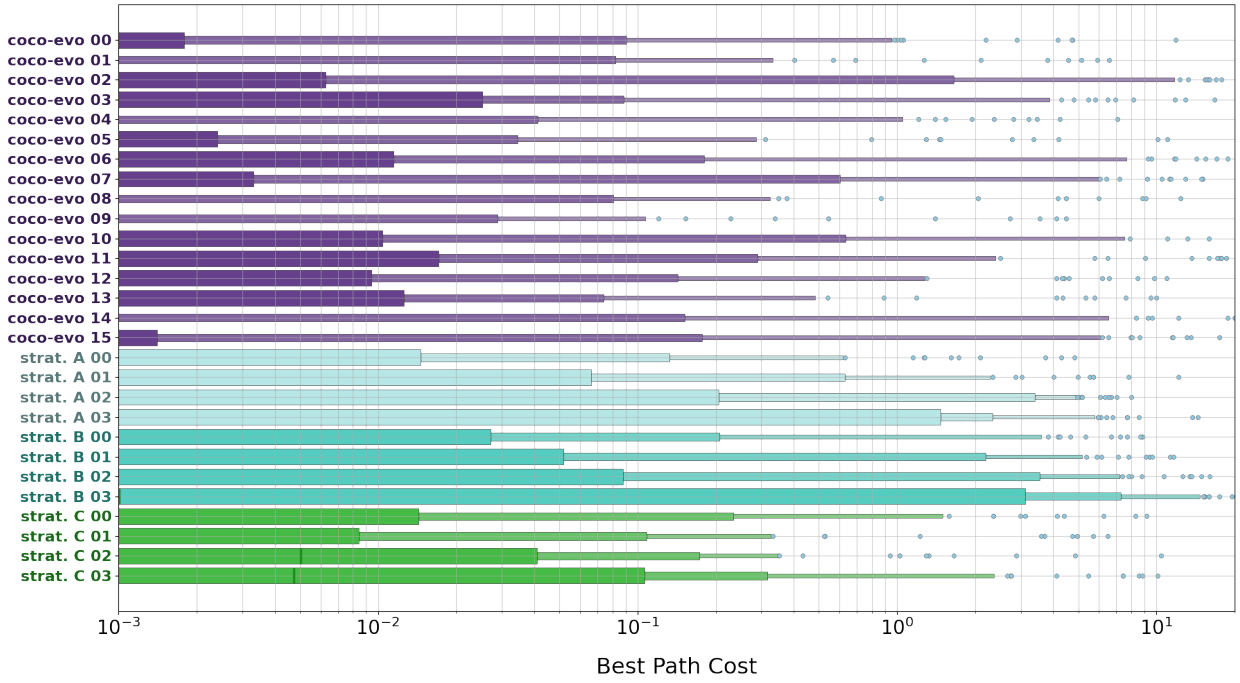


Figure F.17: Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 200 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 50% targeted routes.

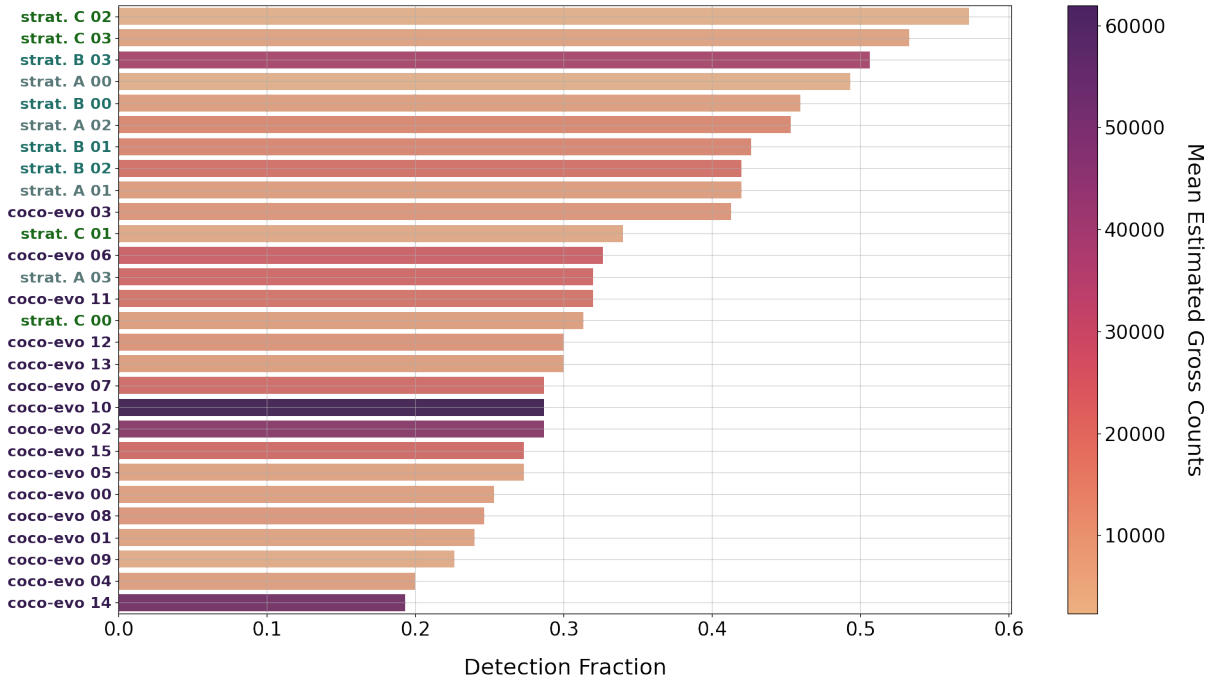


Figure F.18: Comparison of summarized IIA* best path cost results across 200-detector network configurations using route dataset of 50% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations.

F.3.4 Route Dataset: 100% Targeting

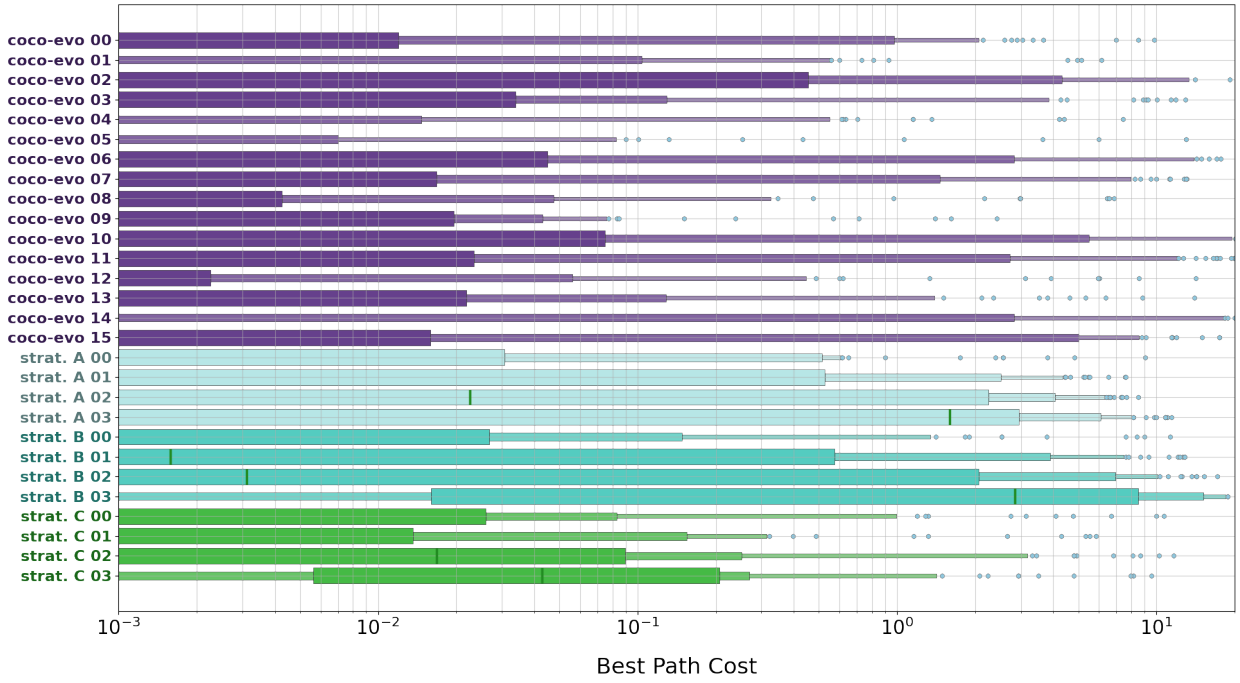


Figure F.19: Letter-value plots of IIA* minimal path cost distributions across detector network configurations constructed with 200 detectors each. Distributions each represent 150 adversarial path evaluations conducted using route dataset of 100% targeted routes.

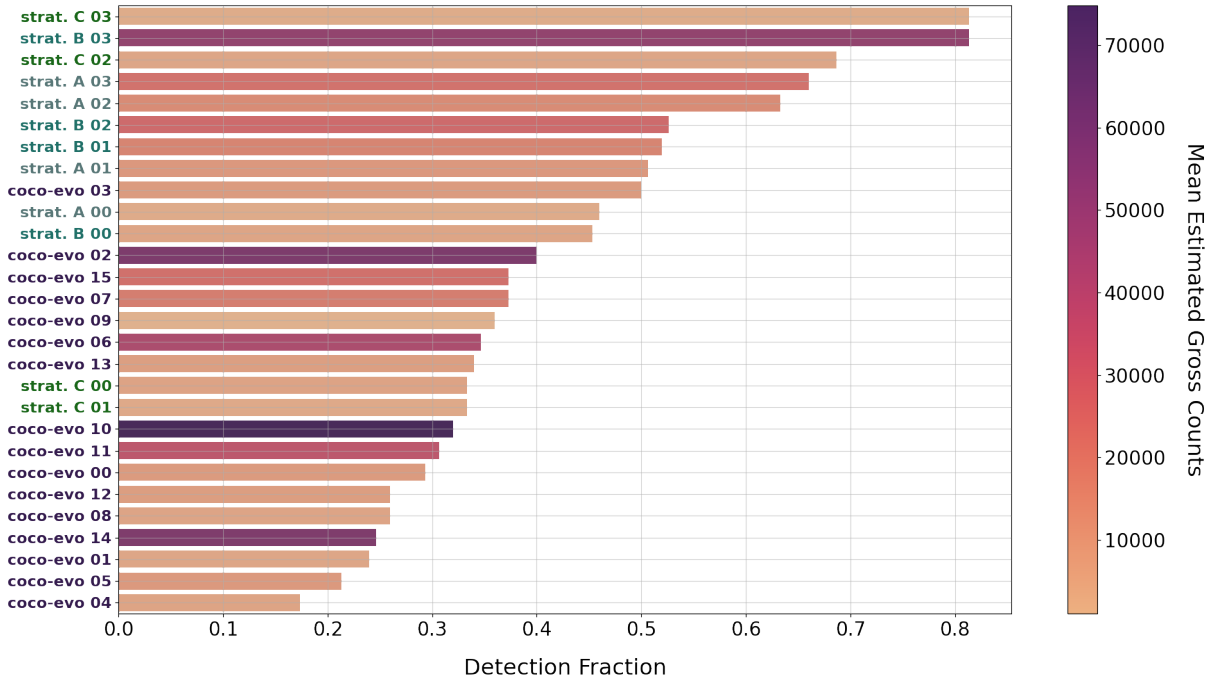


Figure F.20: Comparison of summarized IIA* best path cost results across 200-detector network configurations using route dataset of 100% targeted routes. Fraction of adversarial paths detected in some capacity by network shown along x-axis; bars shaded by average network-wide gross counts accumulated over those evaluations. Count estimates based on assumptions detailed in Section 5.1.2. Metrics calculated for 150 evaluations.

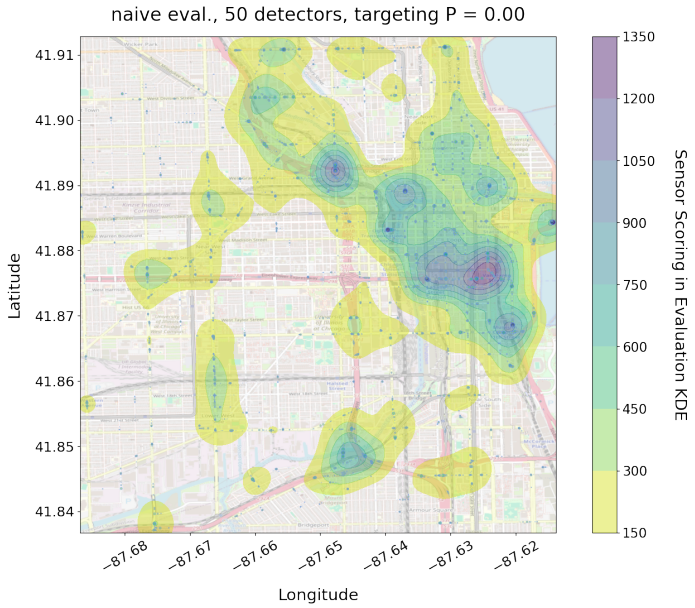
Appendix G

Sensor Scoring: Complete Figures

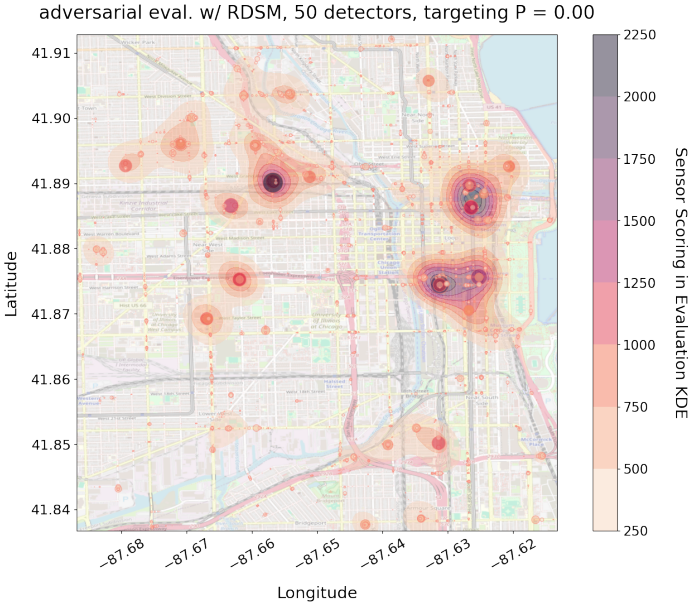
This chapter of the Appendix provides a complete catalogue of results of the sensor scoring analyses described in Section 5.5.2.

G.1 50 Detectors

G.1.1 Route Dataset: 0% Targeting



(a) Naive Evaluation



(b) Adversarial Evaluation

Figure G.1: Detector position scoring with naive and adversarial pathing evaluation using route dataset of 0% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown.

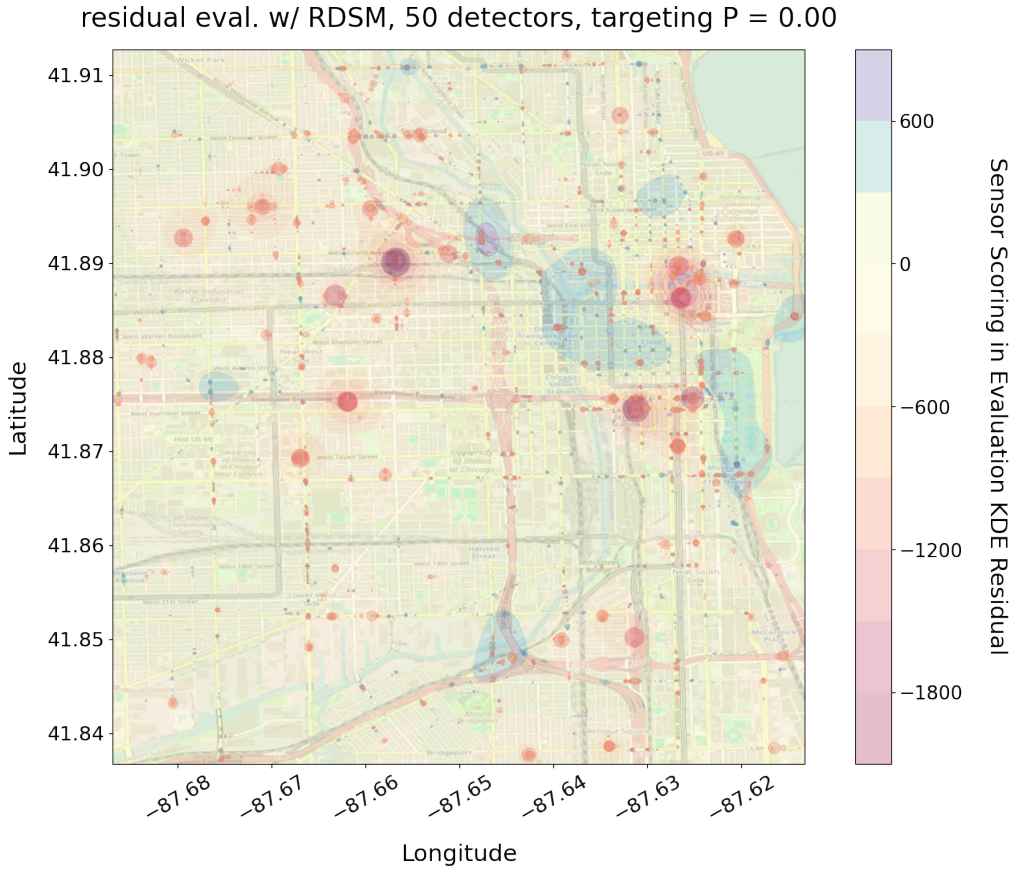
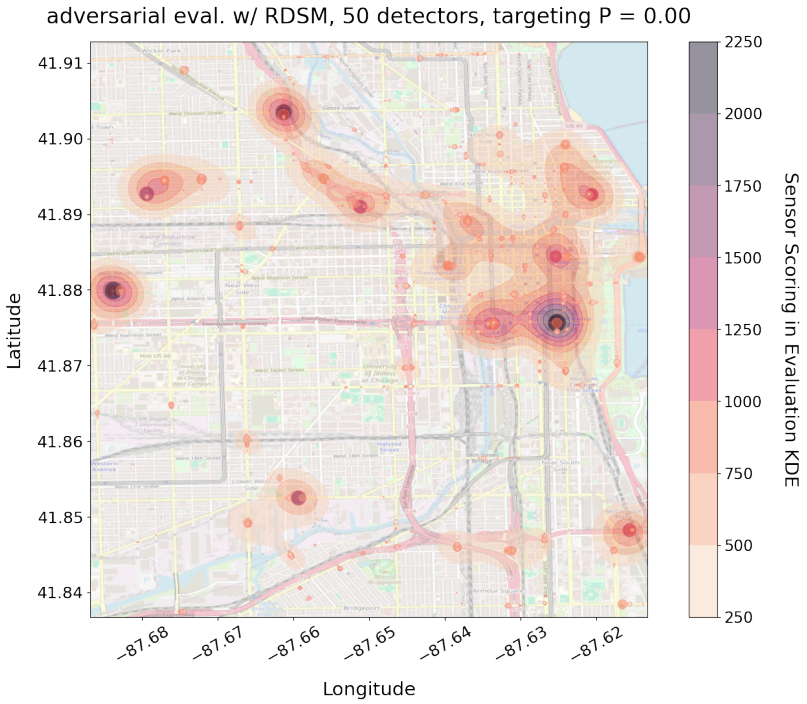
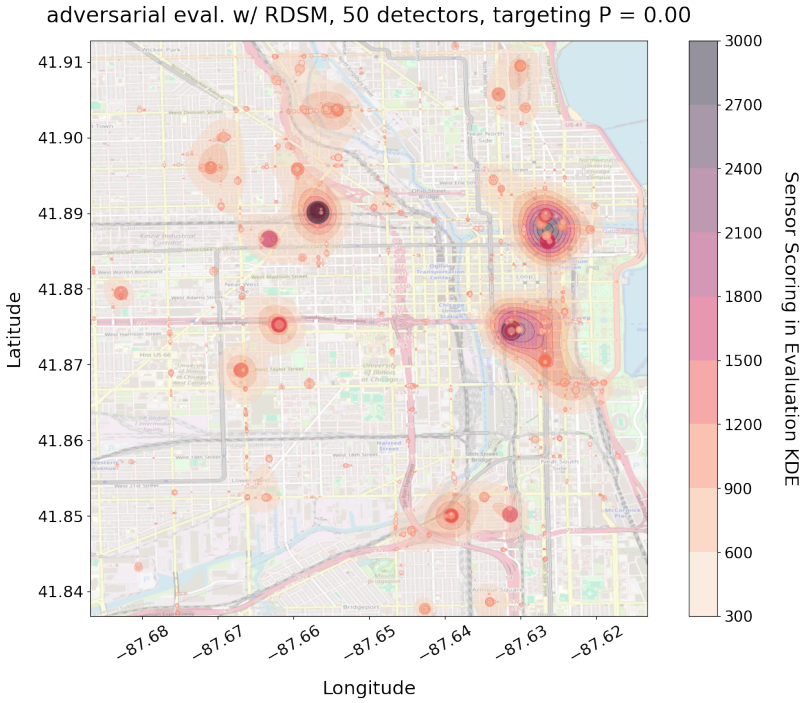


Figure G.2: Residual of detector position scoring KDE for evaluation using route dataset of 0% targeted routes with sensor networks of 50 allotted detectors each.



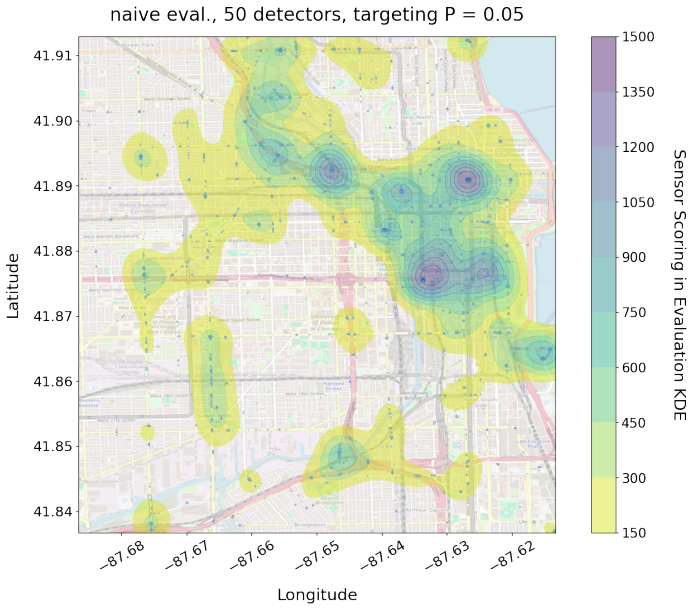
(a) Static strategies



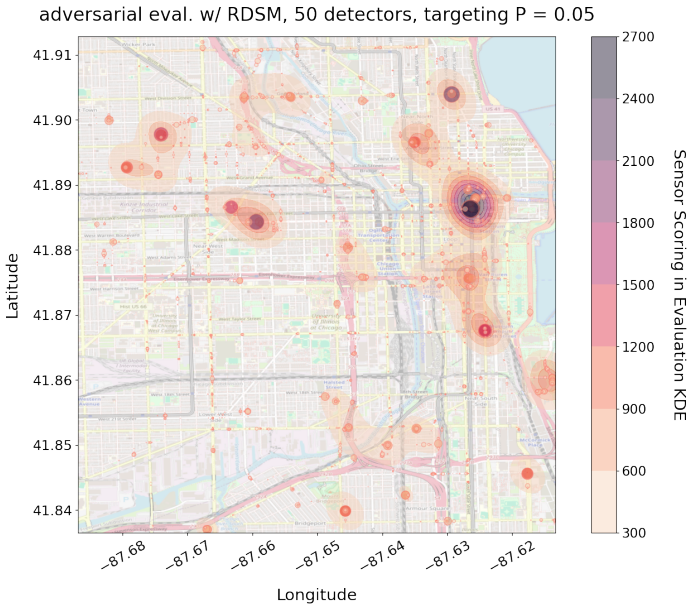
(b) Dynamic strategies

Figure G.3: Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 0% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown.

G.1.2 Route Dataset: 5% Targeting



(a) Naive Evaluation



(b) Adversarial Evaluation

Figure G.4: Detector position scoring with naive and adversarial pathing evaluation using route dataset of 5% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown.

Comparison between static and dynamic strategies shown in Section 5.5.2.

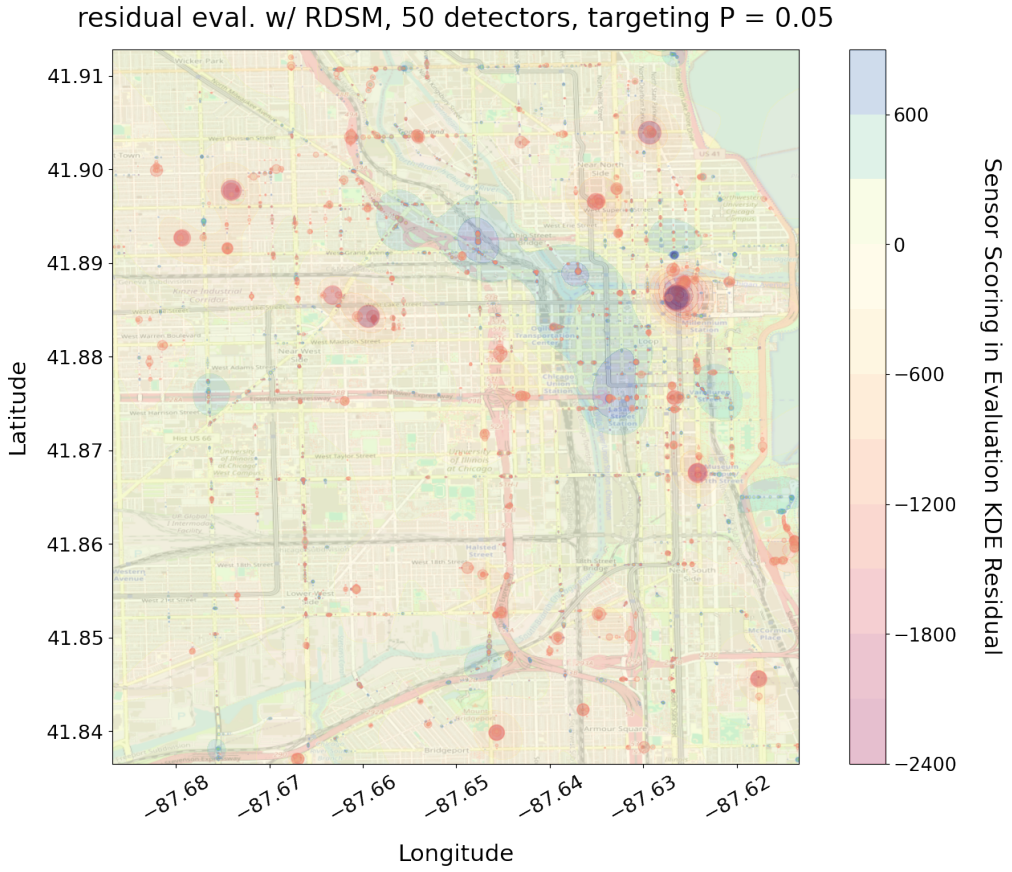
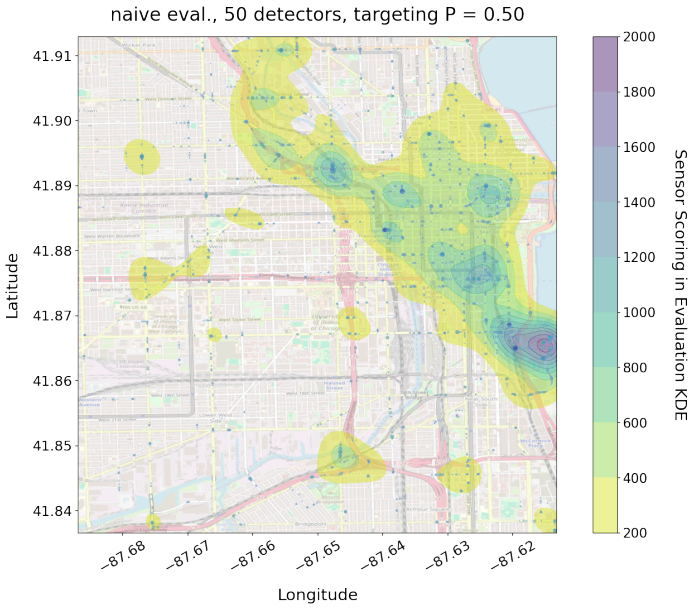
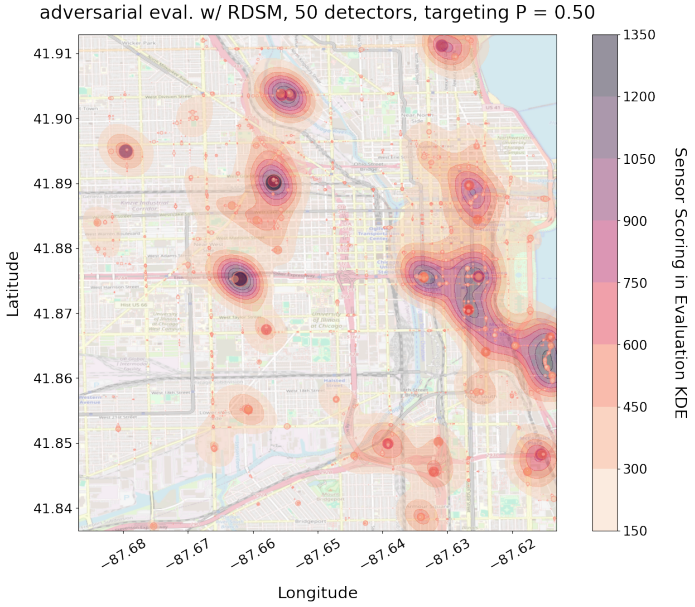


Figure G.5: Residual of detector position scoring KDE for evaluation using route dataset of 5% targeted routes with sensor networks of 50 allotted detectors each.

G.1.3 Route Dataset: 50% Targeting



(a) Naive Evaluation



(b) Adversarial Evaluation

Figure G.6: Detector position scoring with naive and adversarial pathing evaluation using route dataset of 50% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown.

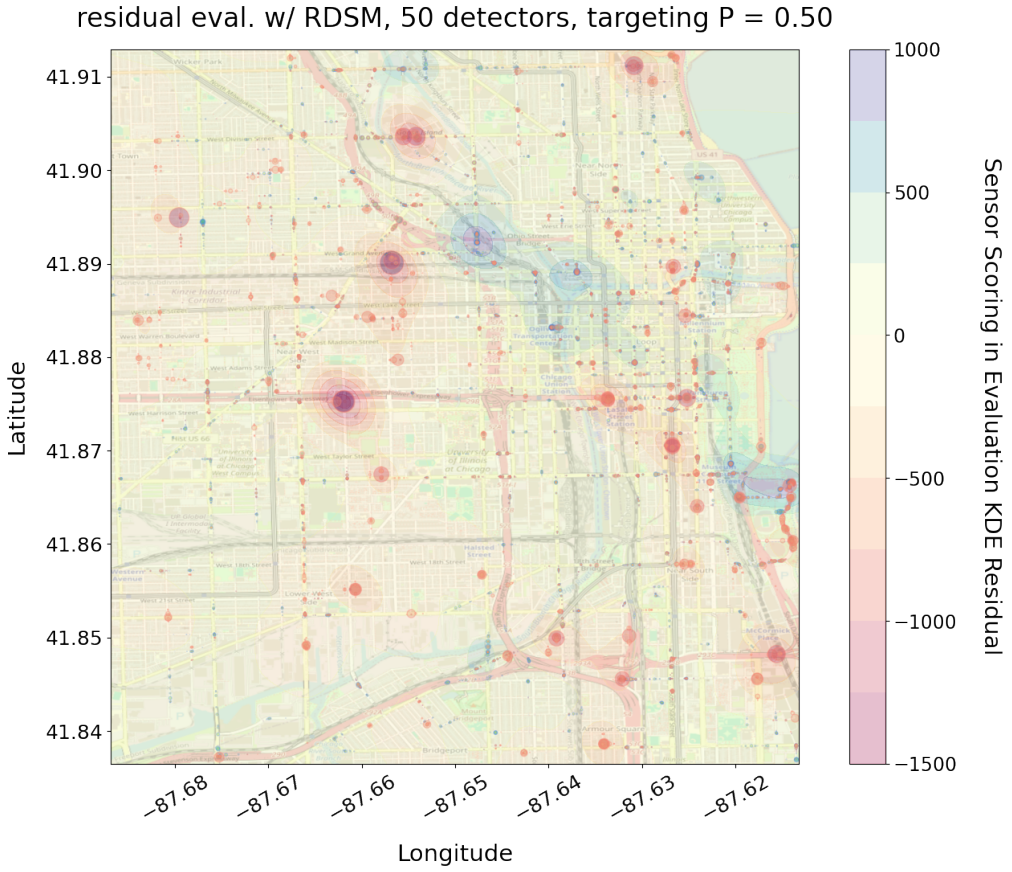
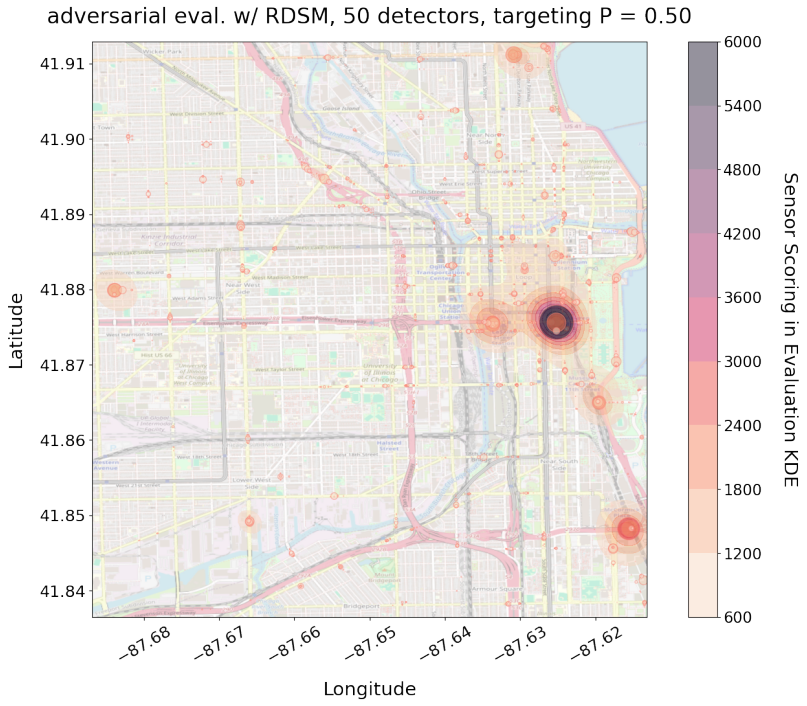
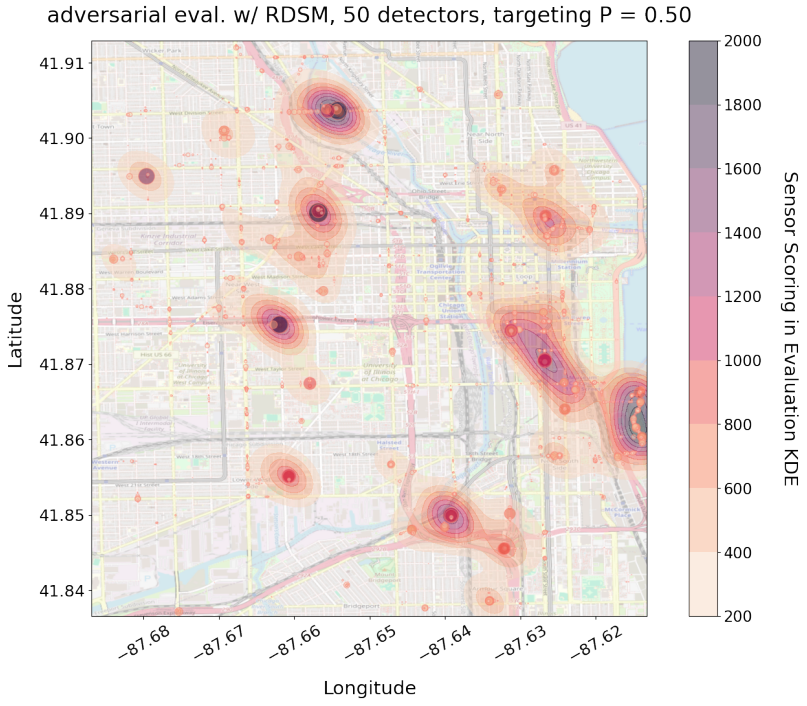


Figure G.7: Residual of detector position scoring KDE for evaluation using route dataset of 50% targeted routes with sensor networks of 50 allotted detectors each.



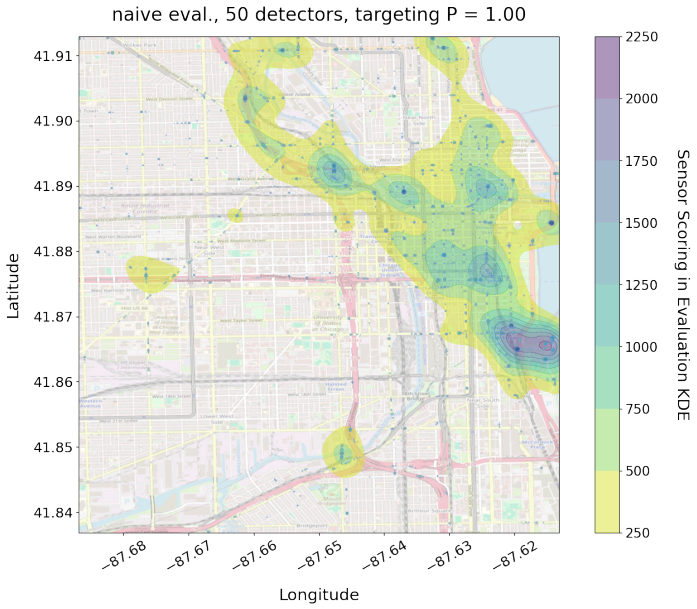
(a) Static strategies



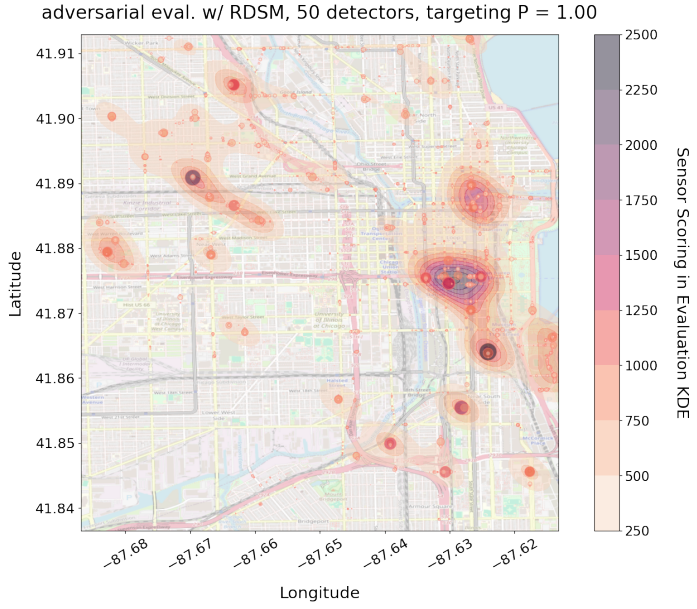
(b) Dynamic strategies

Figure G.8: Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 50% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown.

G.1.4 Route Dataset: 100% Targeting



(a) Naive Evaluation



(b) Adversarial Evaluation

Figure G.9: Detector position scoring with naive and adversarial pathing evaluation using route dataset of 100% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown.

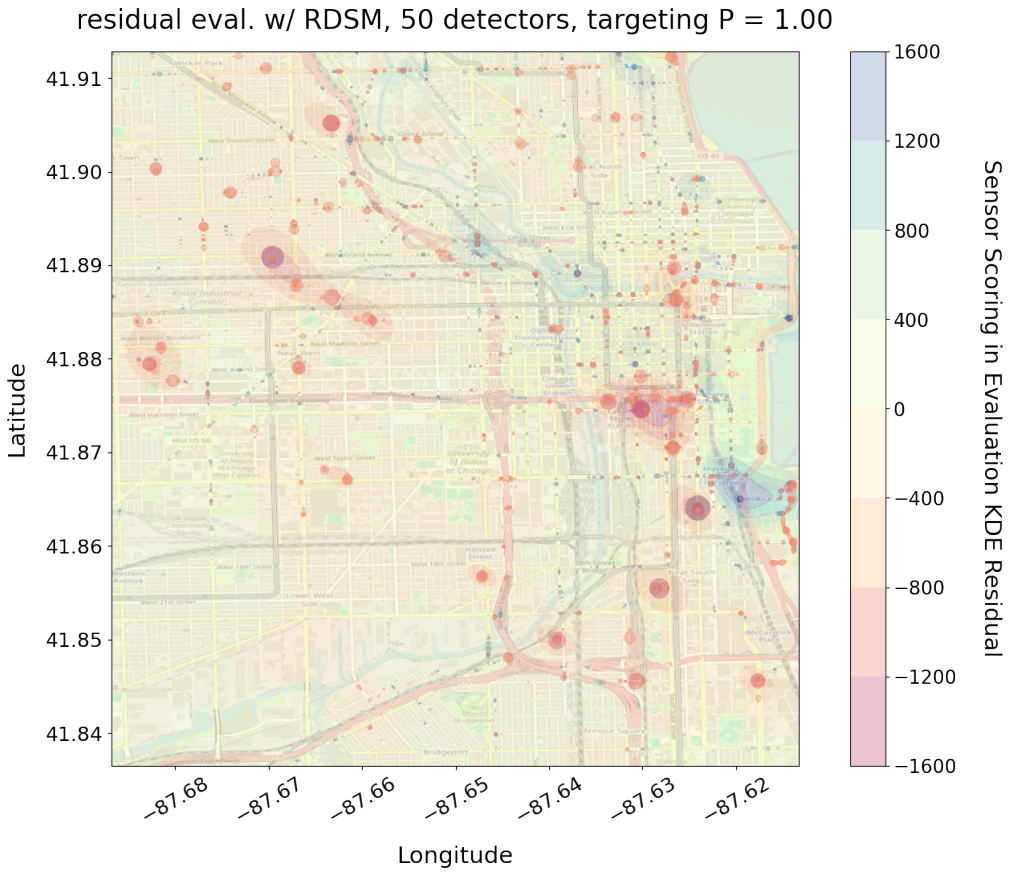
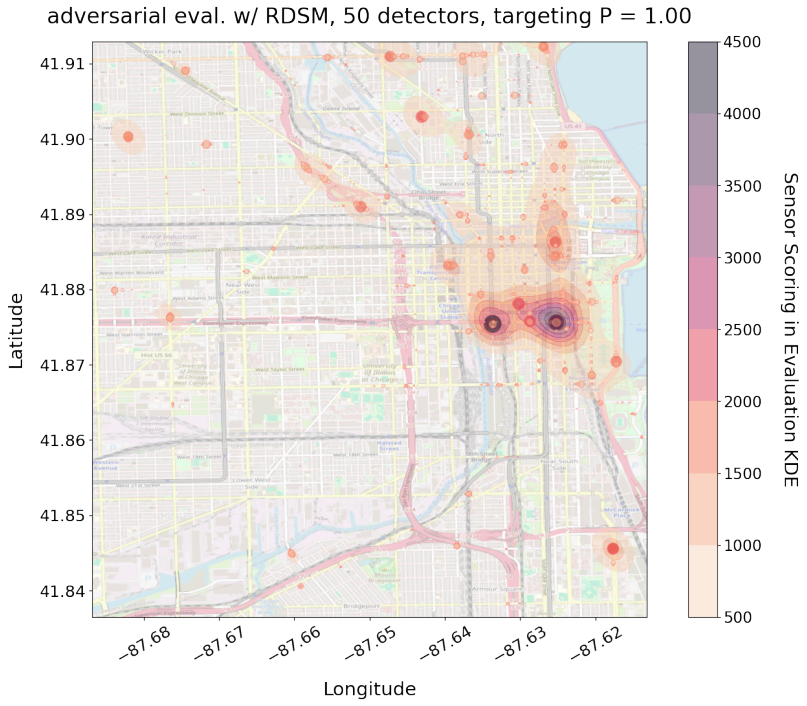
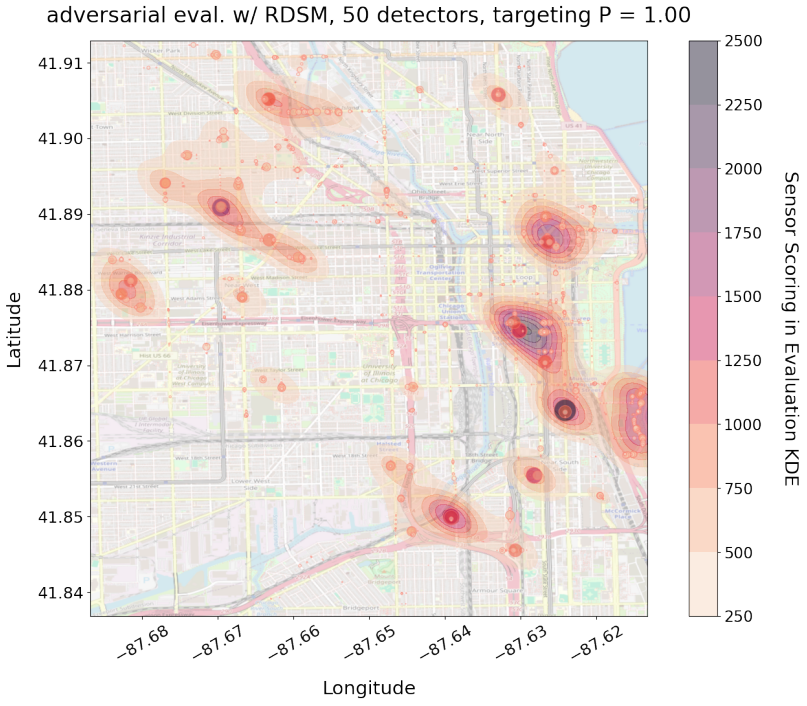


Figure G.10: Residual of detector position scoring KDE for evaluation using route dataset of 100% targeted routes with sensor networks of 50 allotted detectors each.



(a) Static strategies

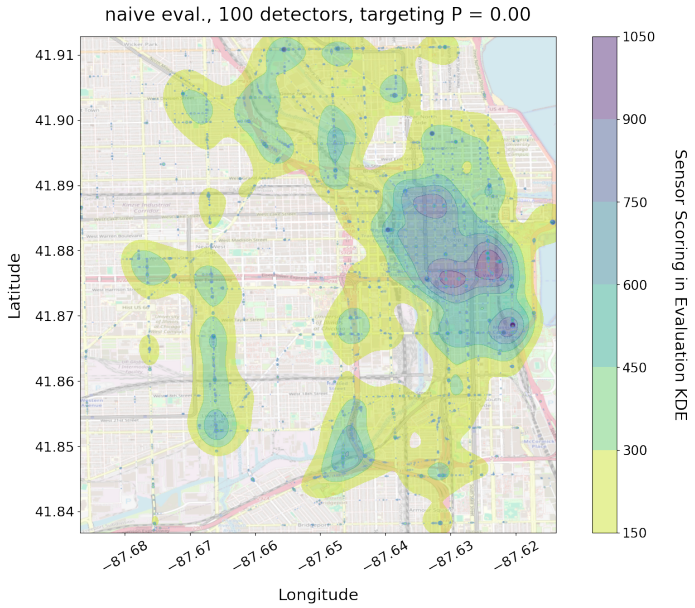


(b) Dynamic strategies

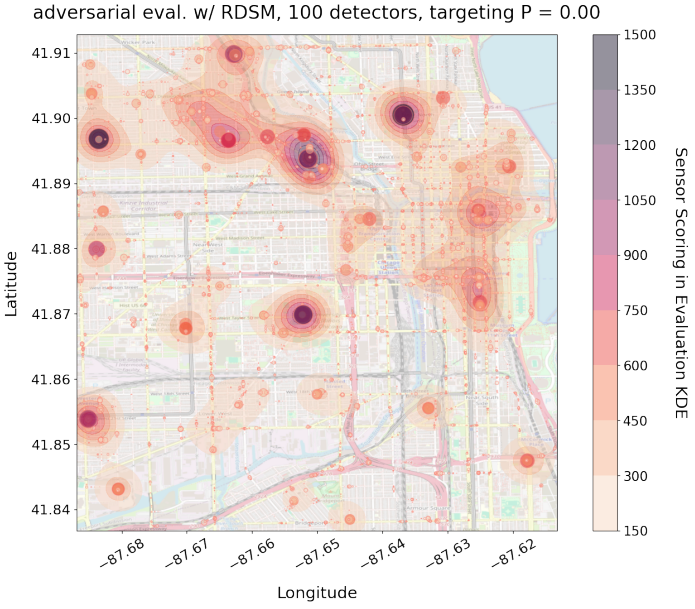
Figure G.11: Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 100% targeted routes with sensor networks of 50 allotted detectors each. Scatter plot of detection locations and KDE shown.

G.2 100 Detectors

G.2.1 Route Dataset: 0% Targeting



(a) Naive Evaluation



(b) Adversarial Evaluation

Figure G.12: Detector position scoring with naive and adversarial pathing evaluation using route dataset of 0% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown.

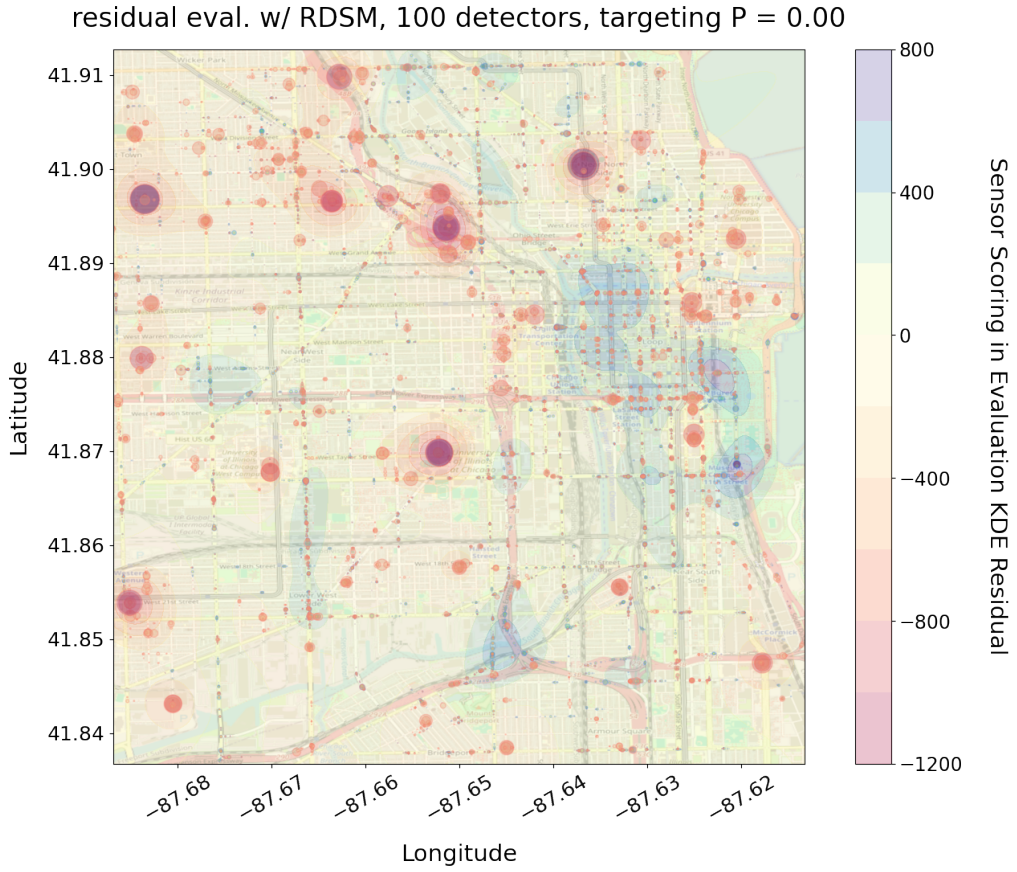
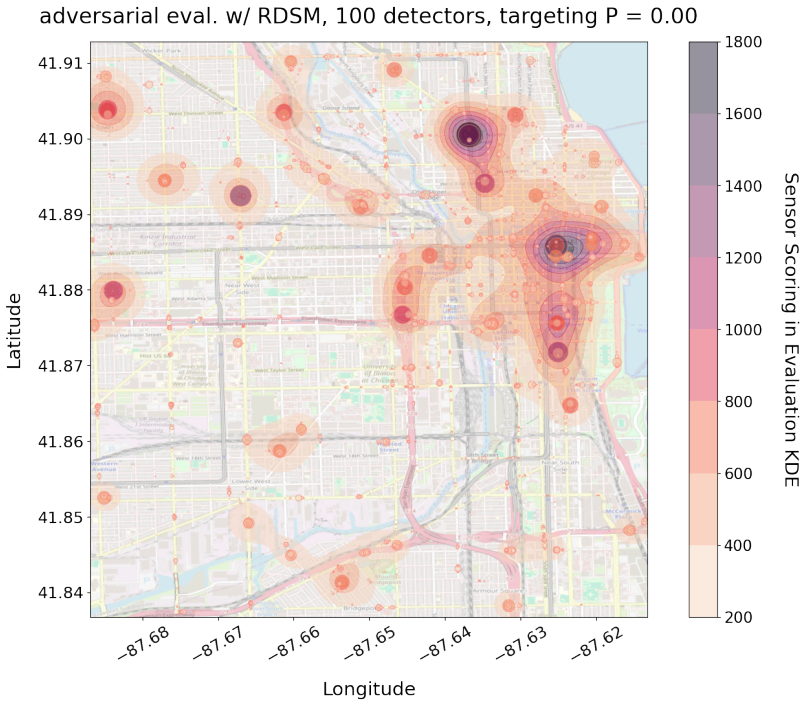
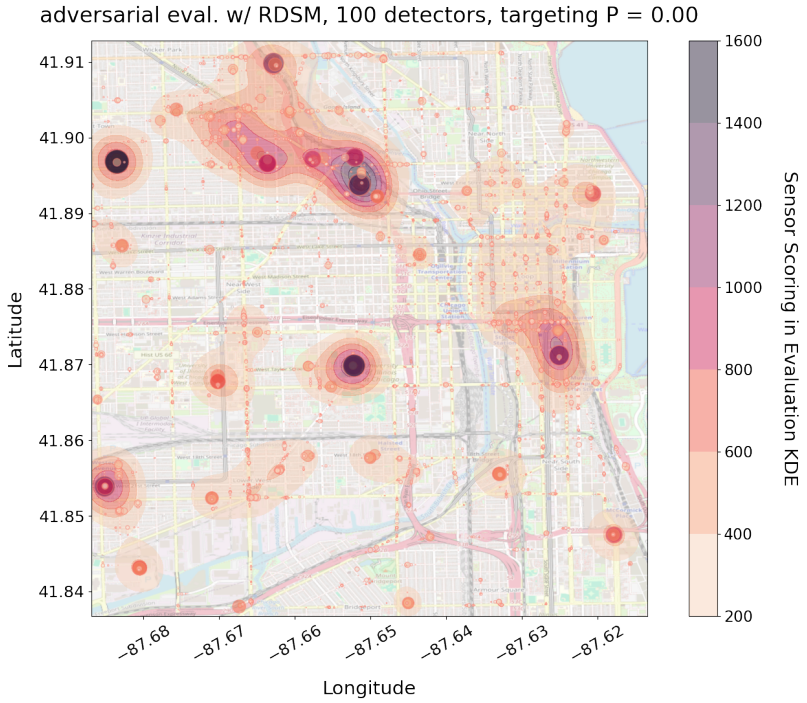


Figure G.13: Residual of detector position scoring KDE for evaluation using route dataset of 0% targeted routes with sensor networks of 100 allotted detectors each.



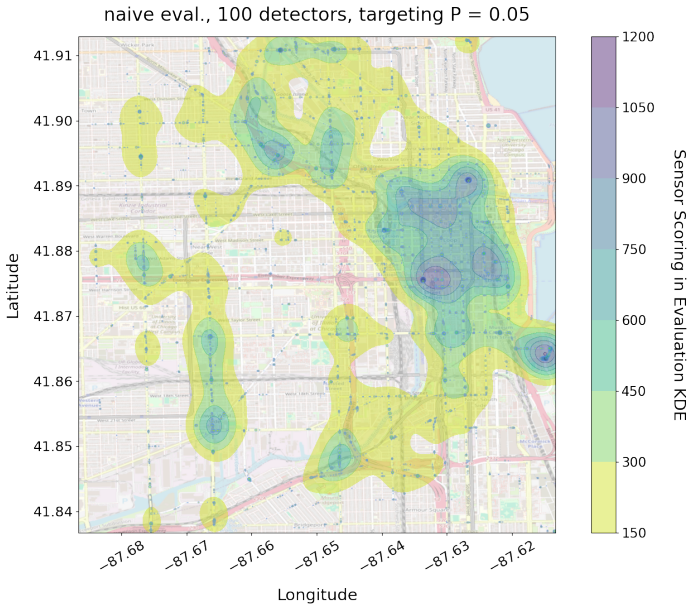
(a) Static strategies



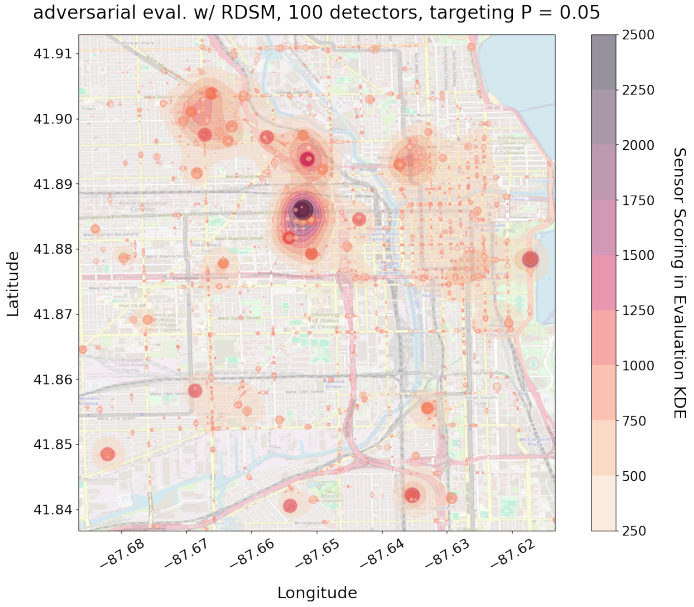
(b) Dynamic strategies

Figure G.14: Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 0% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown.

G.2.2 Route Dataset: 5% Targeting



(a) Naive Evaluation



(b) Adversarial Evaluation

Figure G.15: Detector position scoring with naive and adversarial pathing evaluation using route dataset of 5% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown.

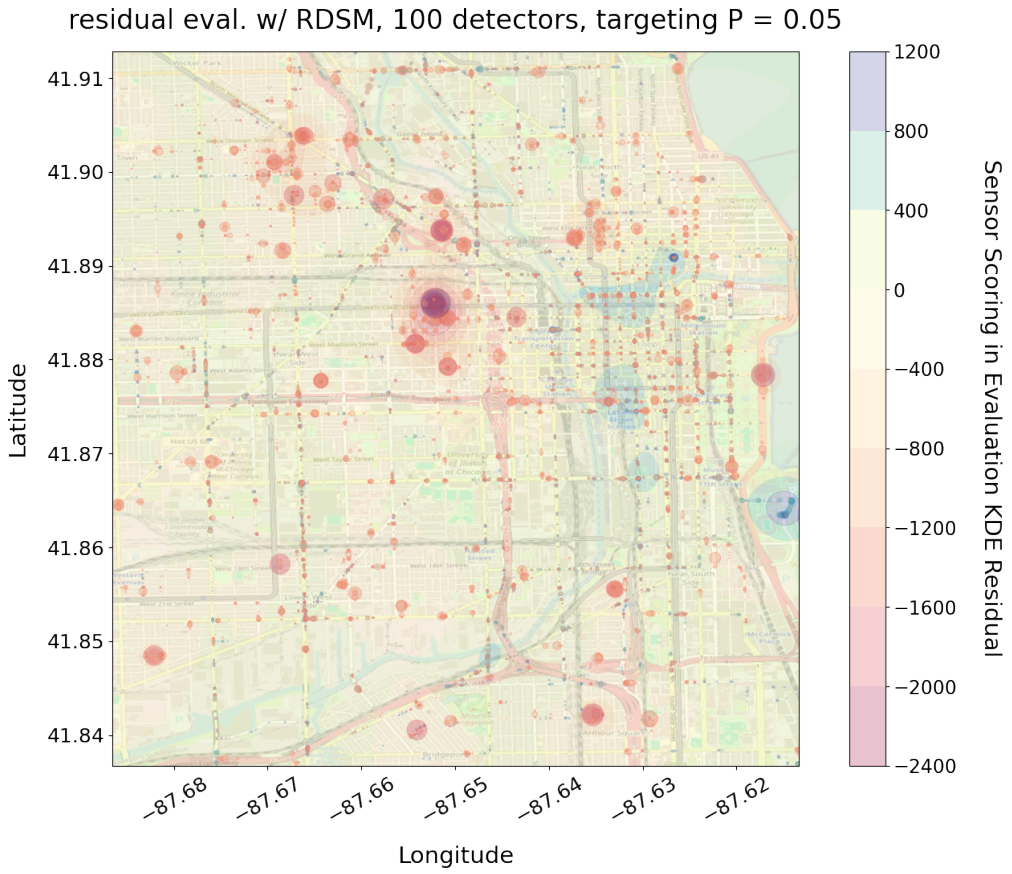
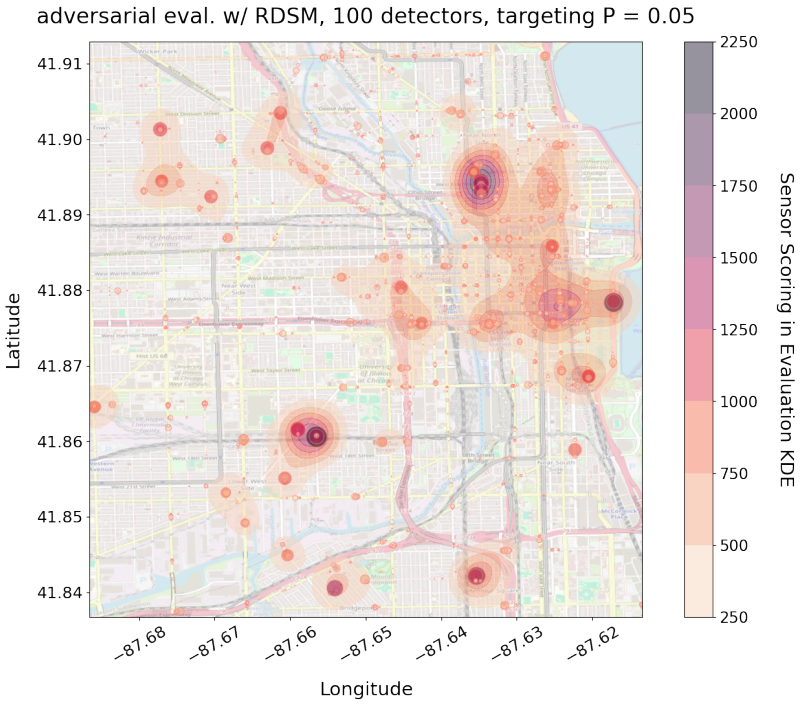
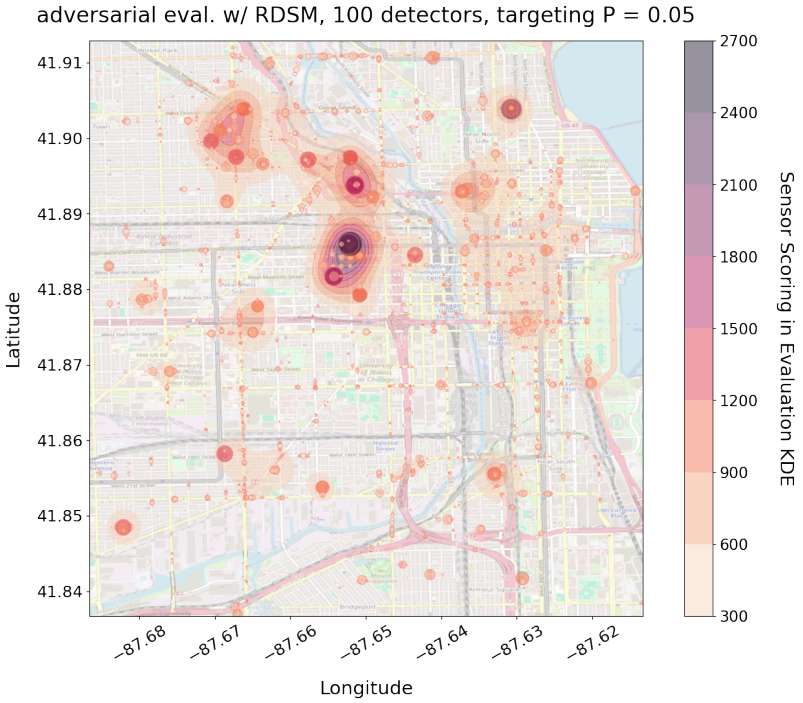


Figure G.16: Residual of detector position scoring KDE for evaluation using route dataset of 5% targeted routes with sensor networks of 100 allotted detectors each.



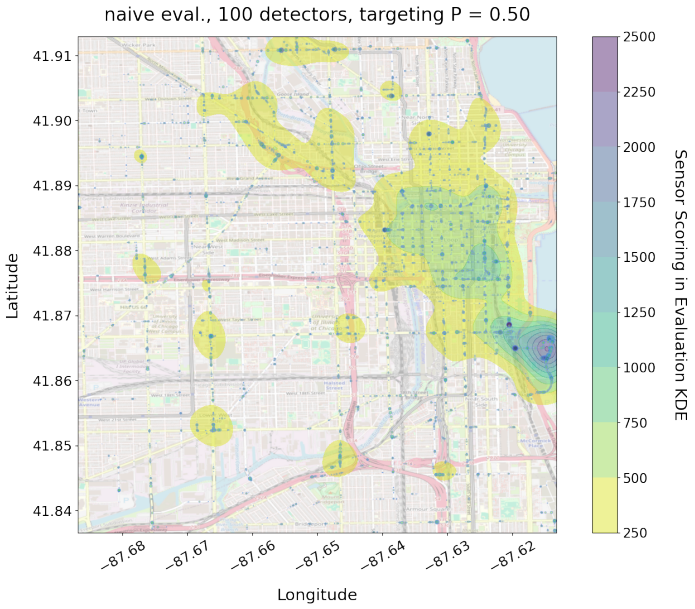
(a) Static strategies



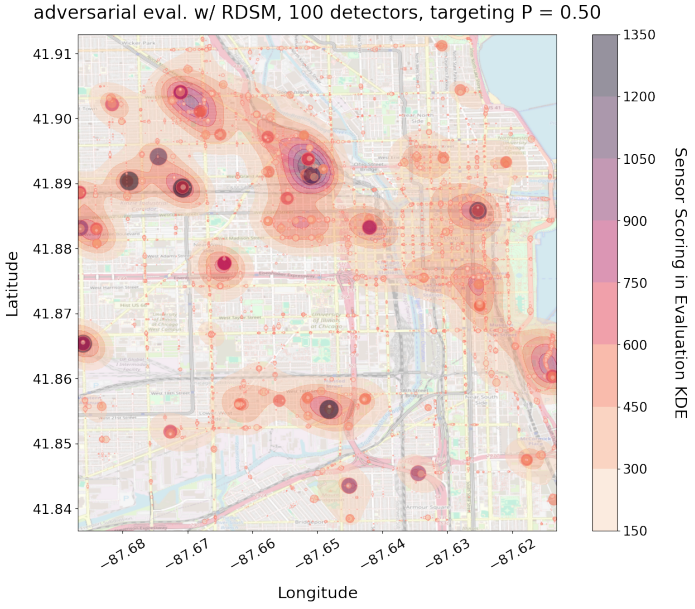
(b) Dynamic strategies

Figure G.17: Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 5% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown.

G.2.3 Route Dataset: 50% Targeting



(a) Naive Evaluation



(b) Adversarial Evaluation

Figure G.18: Detector position scoring with naive and adversarial pathing evaluation using route dataset of 50% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown.

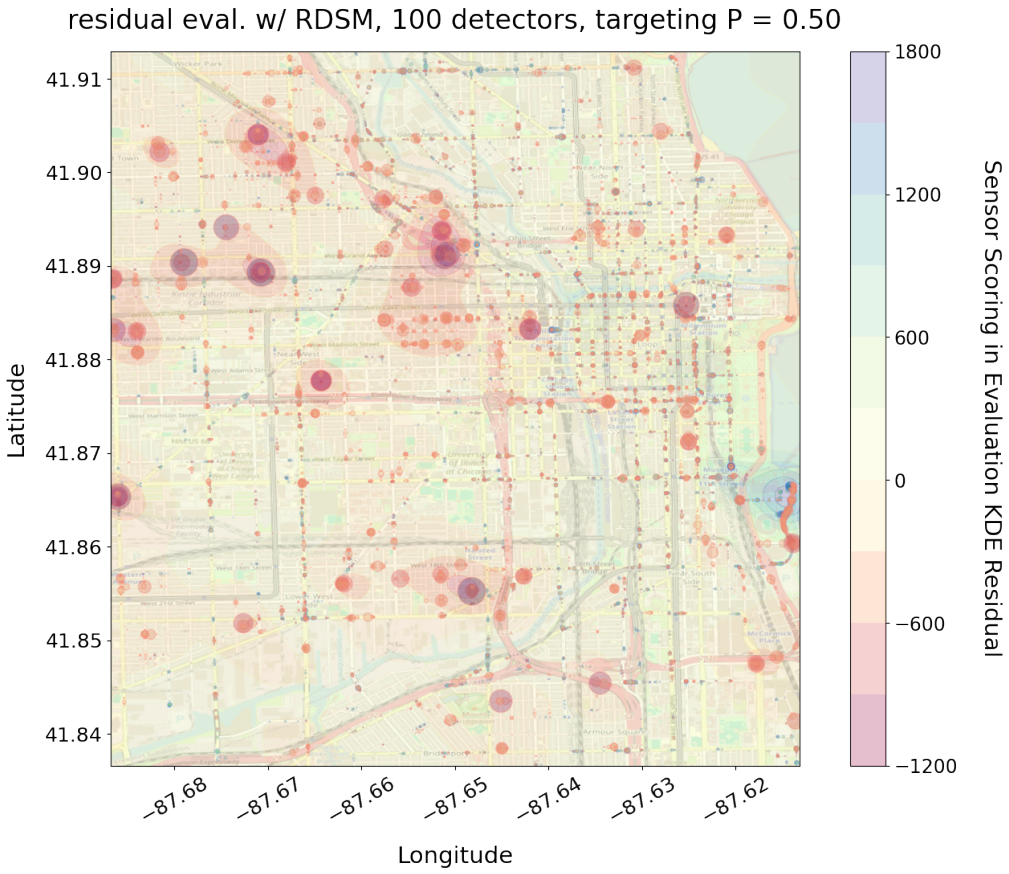
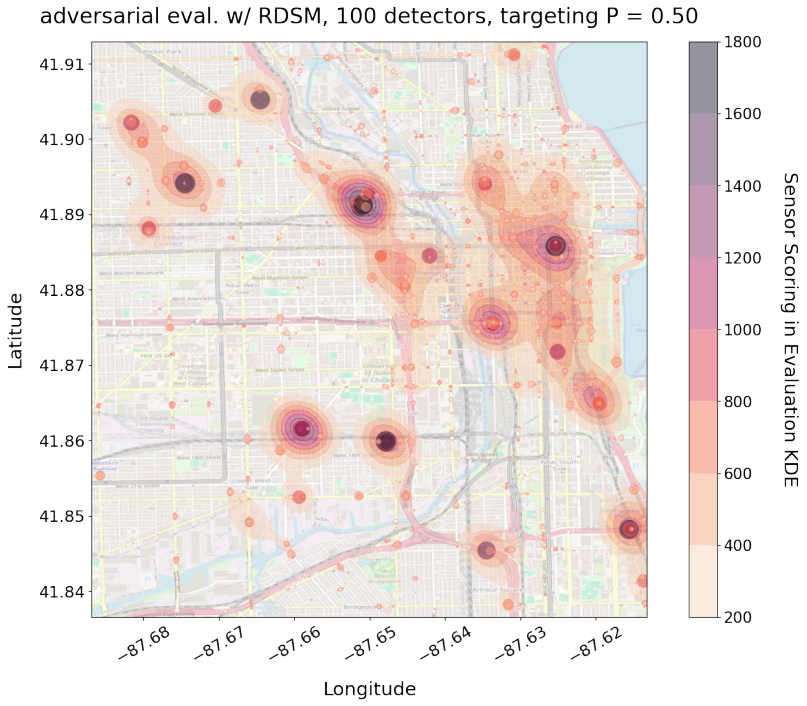
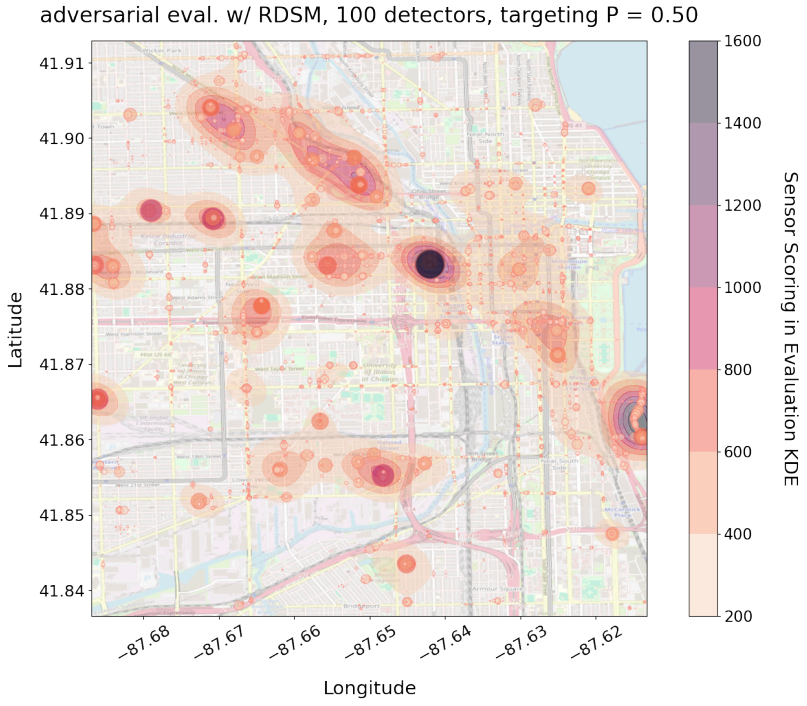


Figure G.19: Residual of detector position scoring KDE for evaluation using route dataset of 50% targeted routes with sensor networks of 100 allotted detectors each.



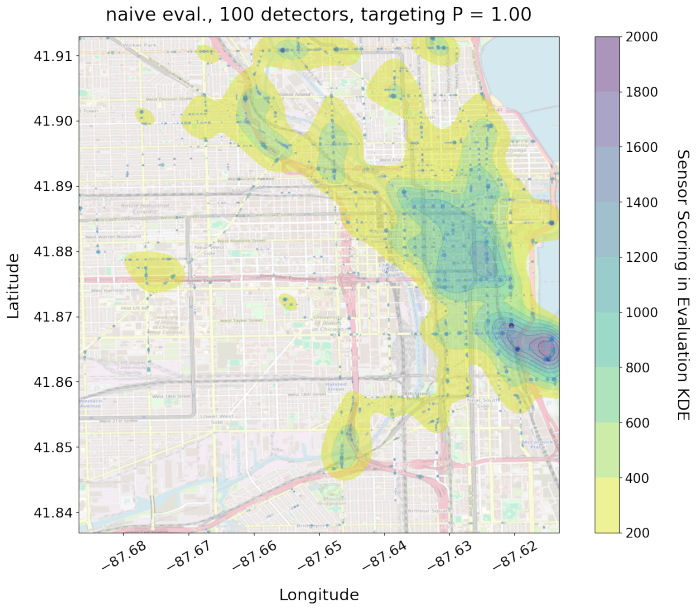
(a) Static strategies



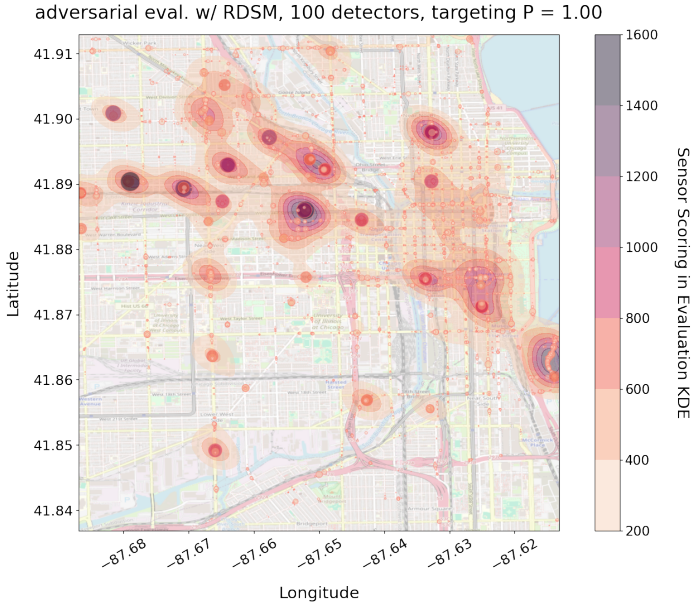
(b) Dynamic strategies

Figure G.20: Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 50% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown.

G.2.4 Route Dataset: 100% Targeting



(a) Naive Evaluation



(b) Adversarial Evaluation

Figure G.21: Detector position scoring with naive and adversarial pathing evaluation using route dataset of 100% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown.

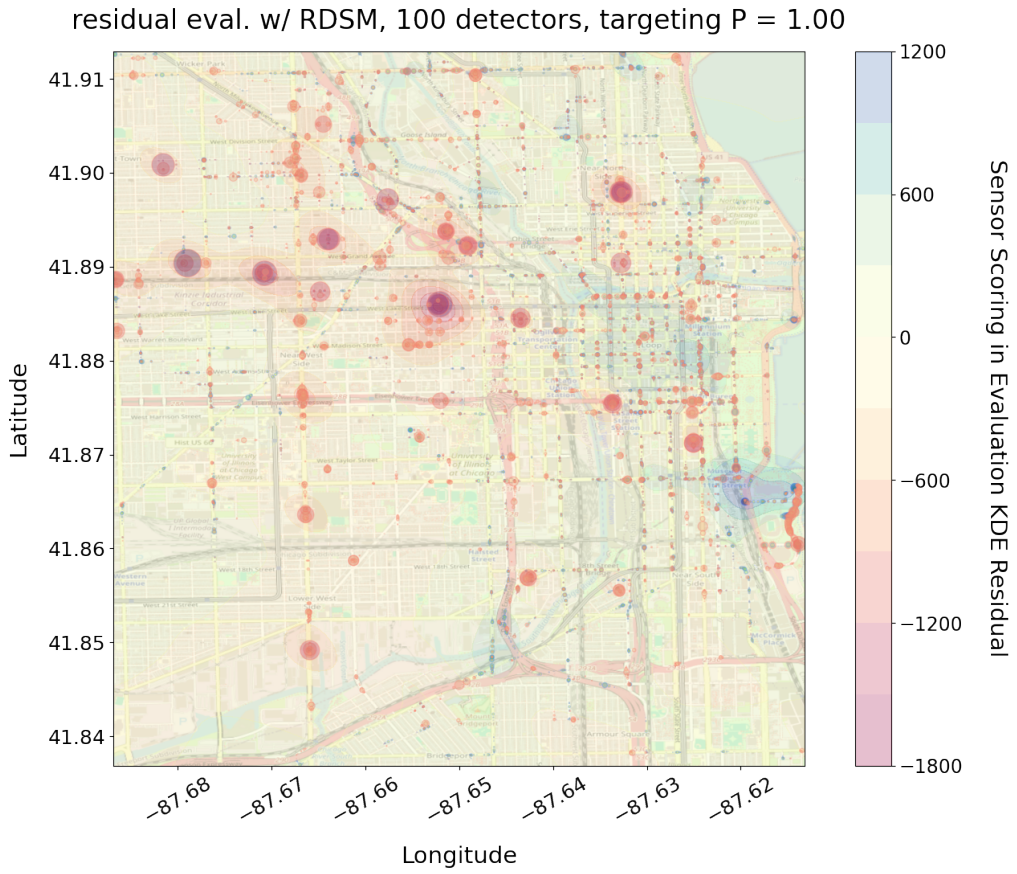
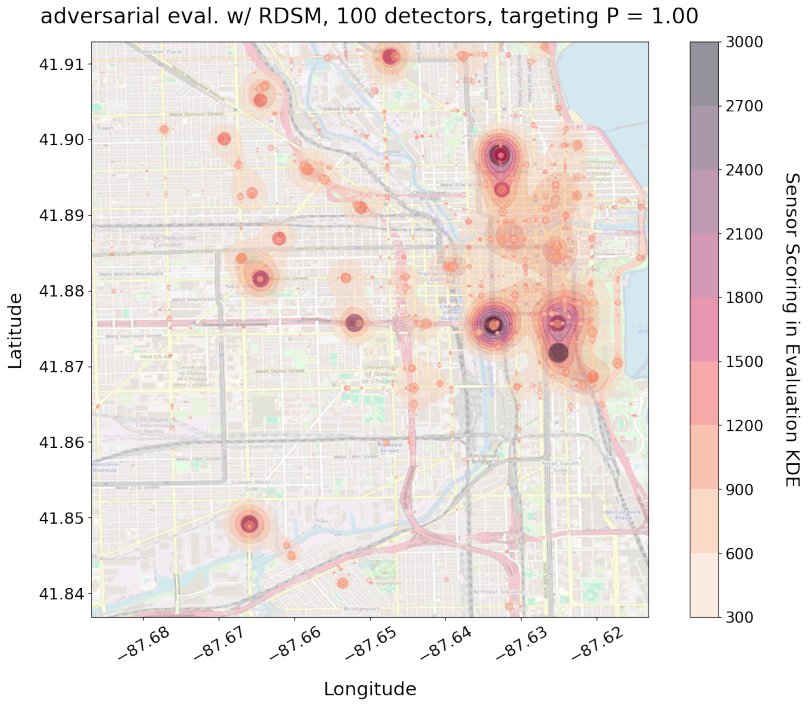
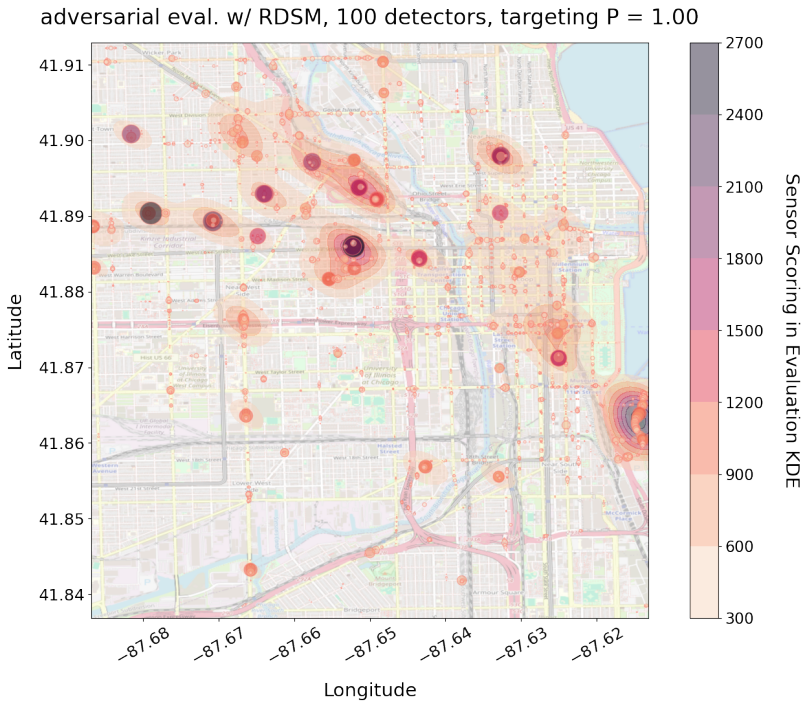


Figure G.22: Residual of detector position scoring KDE for evaluation using route dataset of 100% targeted routes with sensor networks of 100 allotted detectors each.



(a) Static strategies

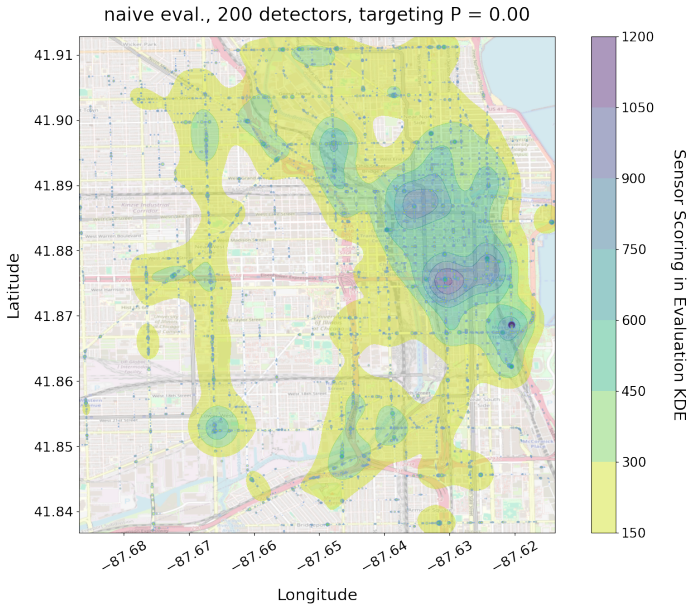


(b) Dynamic strategies

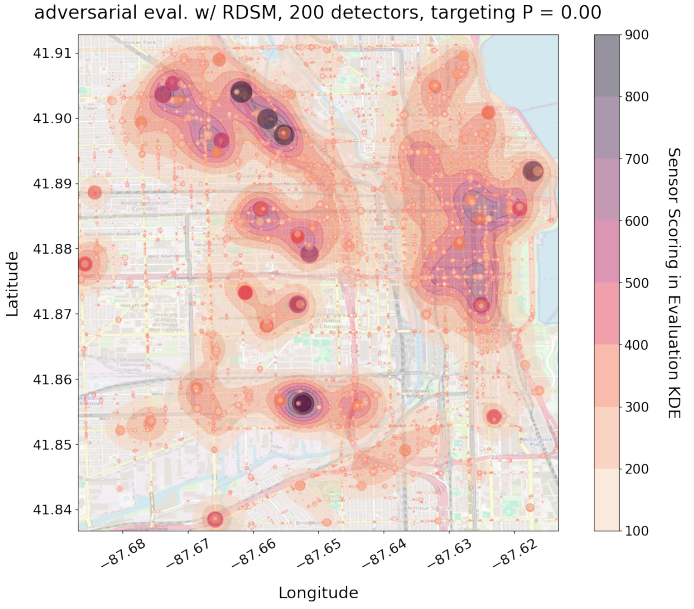
Figure G.23: Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 100% targeted routes with sensor networks of 100 allotted detectors each. Scatter plot of detection locations and KDE shown.

G.3 200 Detectors

G.3.1 Route Dataset: 0% Targeting



(a) Naive Evaluation



(b) Adversarial Evaluation

Figure G.24: Detector position scoring with naive and adversarial pathing evaluation using route dataset of 0% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown.

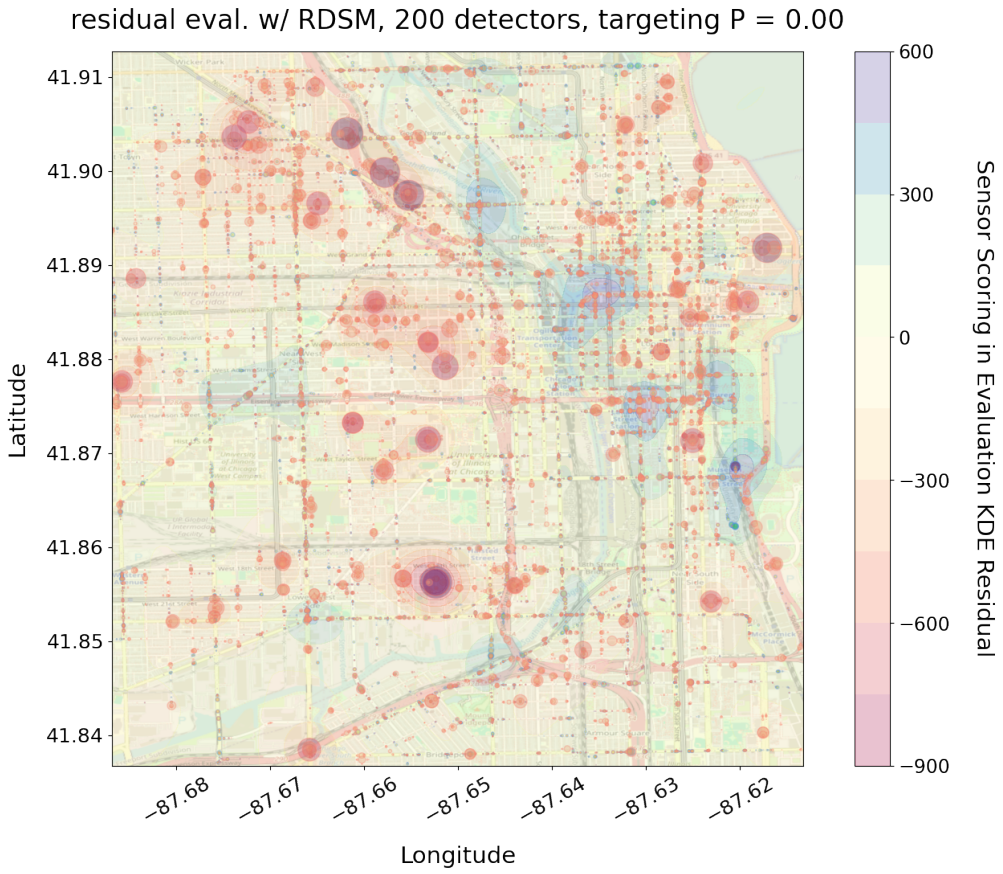
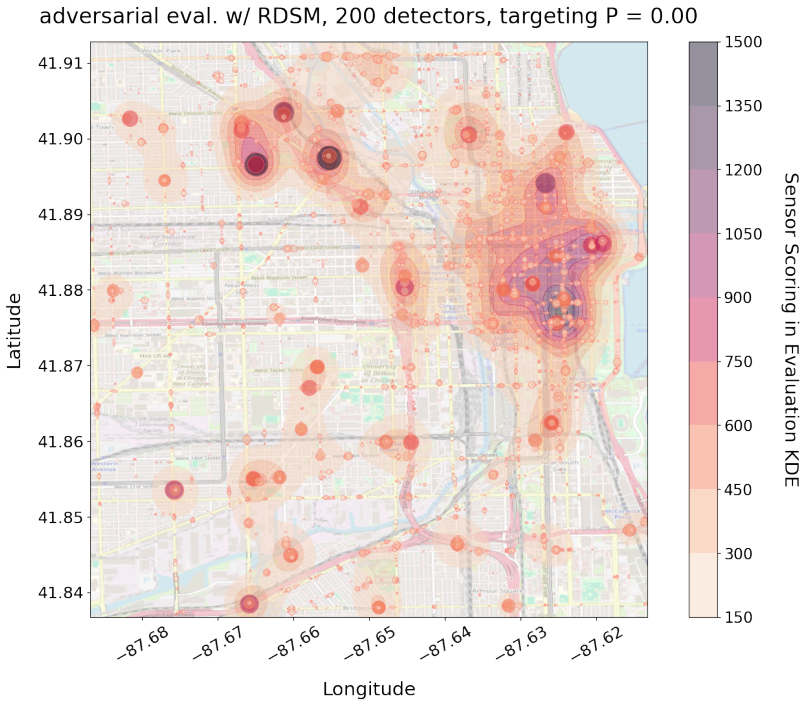
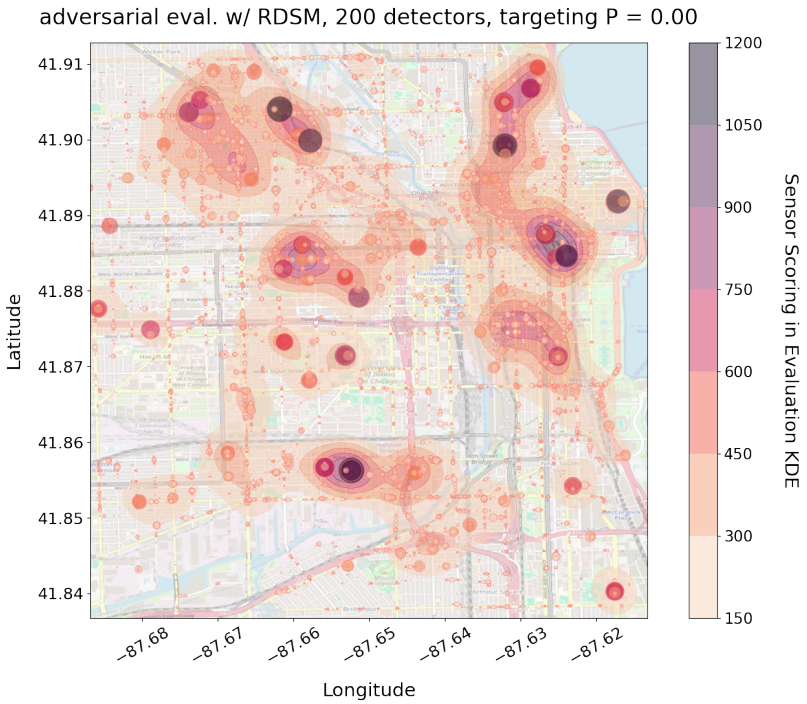


Figure G.25: Residual of detector position scoring KDE for evaluation using route dataset of 0% targeted routes with sensor networks of 200 allotted detectors each.



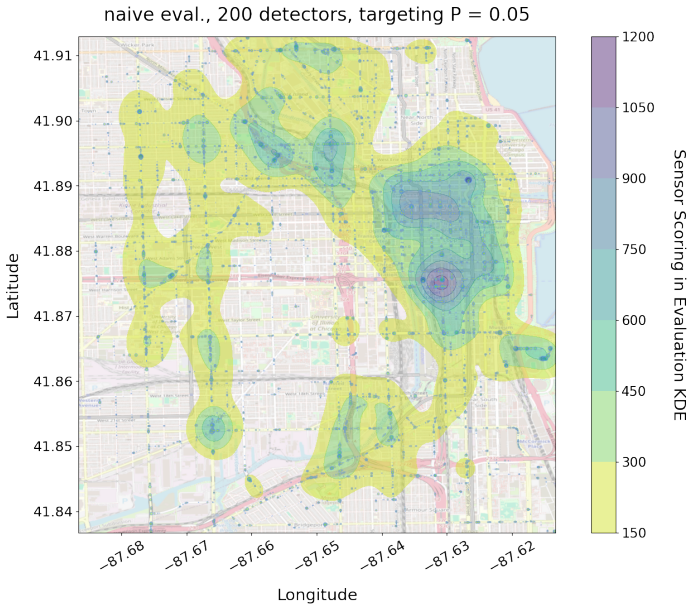
(a) Static strategies



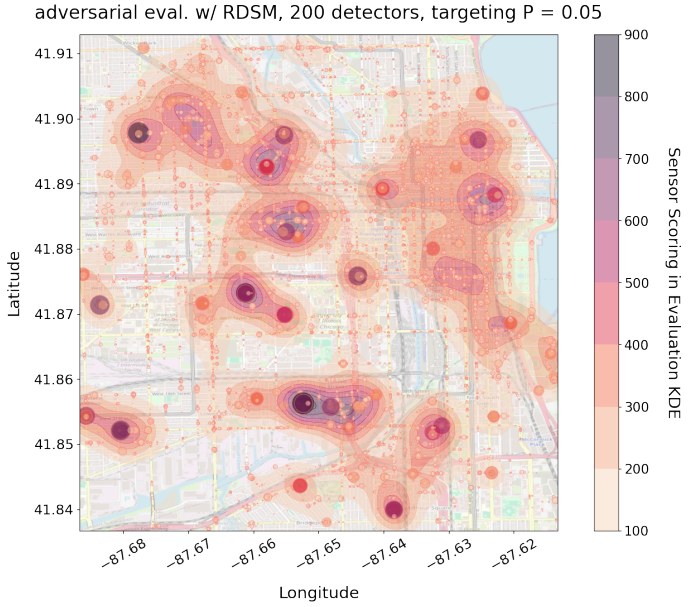
(b) Dynamic strategies

Figure G.26: Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 0% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown.

G.3.2 Route Dataset: 5% Targeting



(a) Naive Evaluation



(b) Adversarial Evaluation

Figure G.27: Detector position scoring with naive and adversarial pathing evaluation using route dataset of 5% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown.

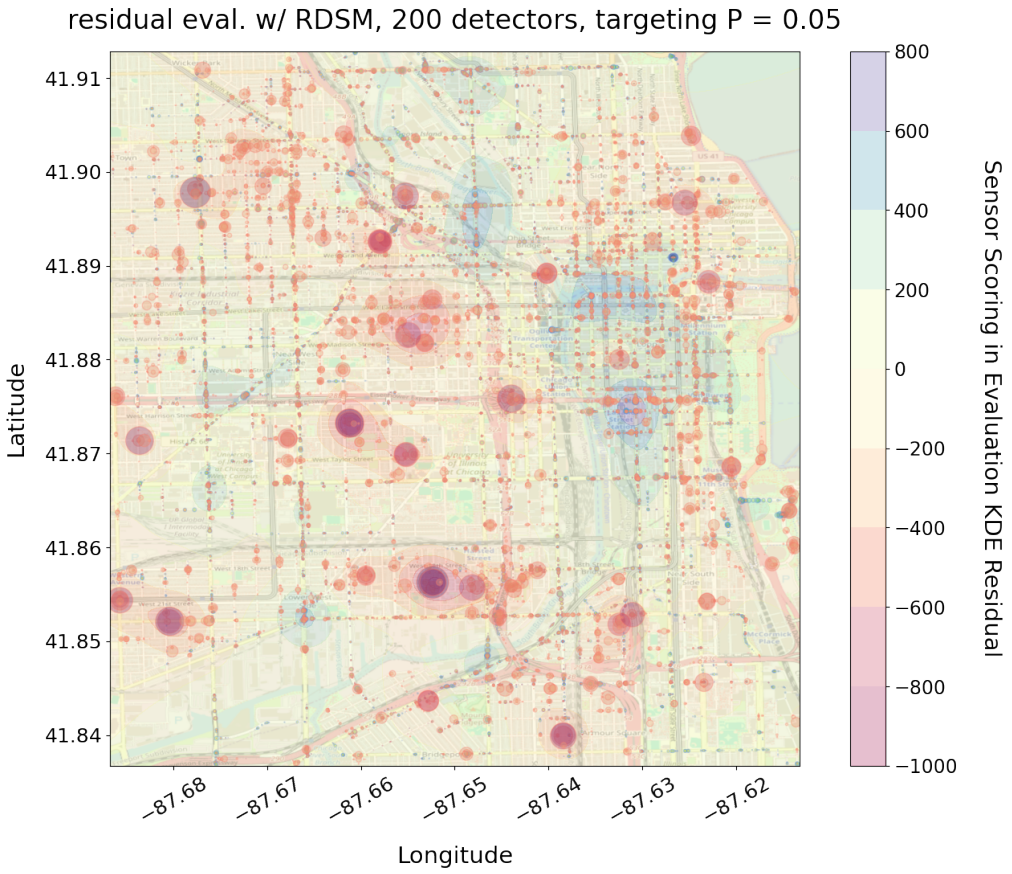
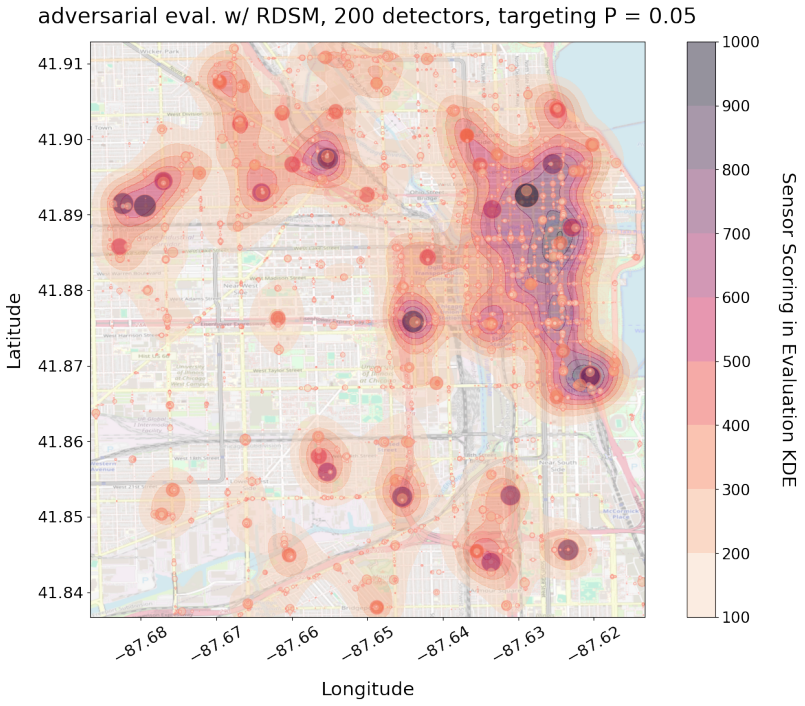
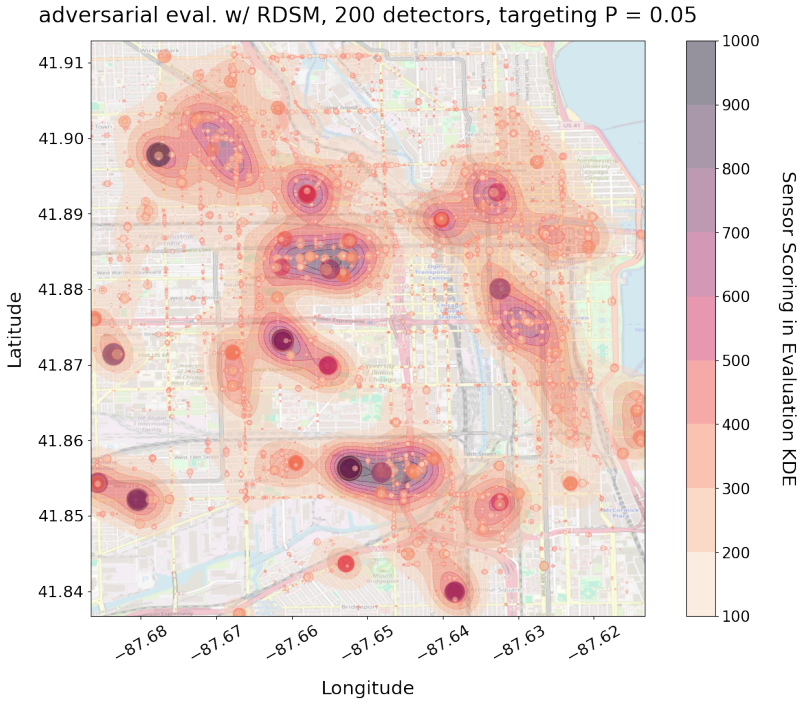


Figure G.28: Residual of detector position scoring KDE for evaluation using route dataset of 5% targeted routes with sensor networks of 200 allotted detectors each.



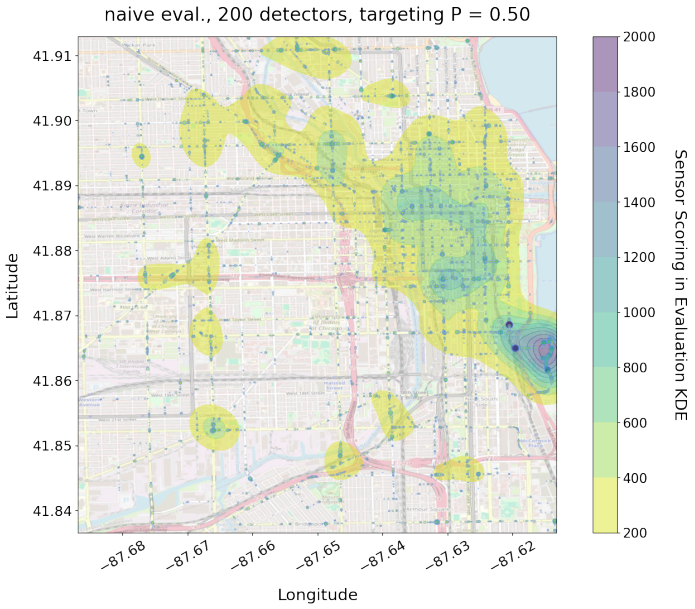
(a) Static strategies



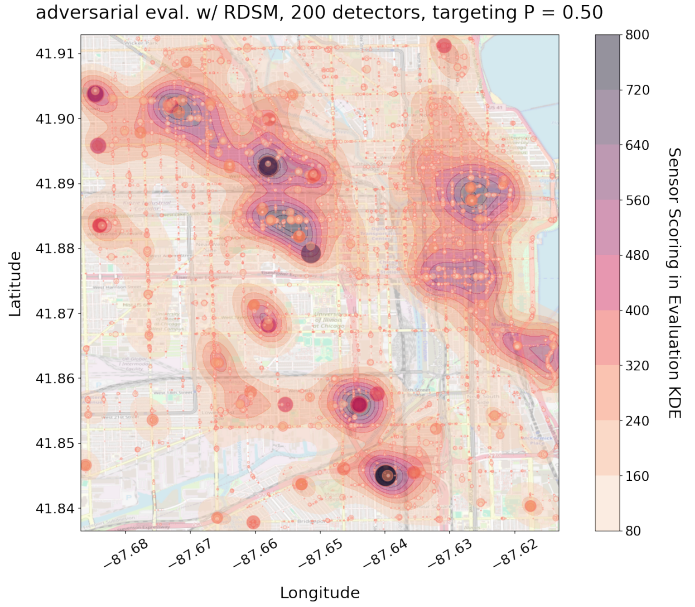
(b) Dynamic strategies

Figure G.29: Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 5% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown.

G.3.3 Route Dataset: 50% Targeting



(a) Naive Evaluation



(b) Adversarial Evaluation

Figure G.30: Detector position scoring with naive and adversarial pathing evaluation using route dataset of 50% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown.

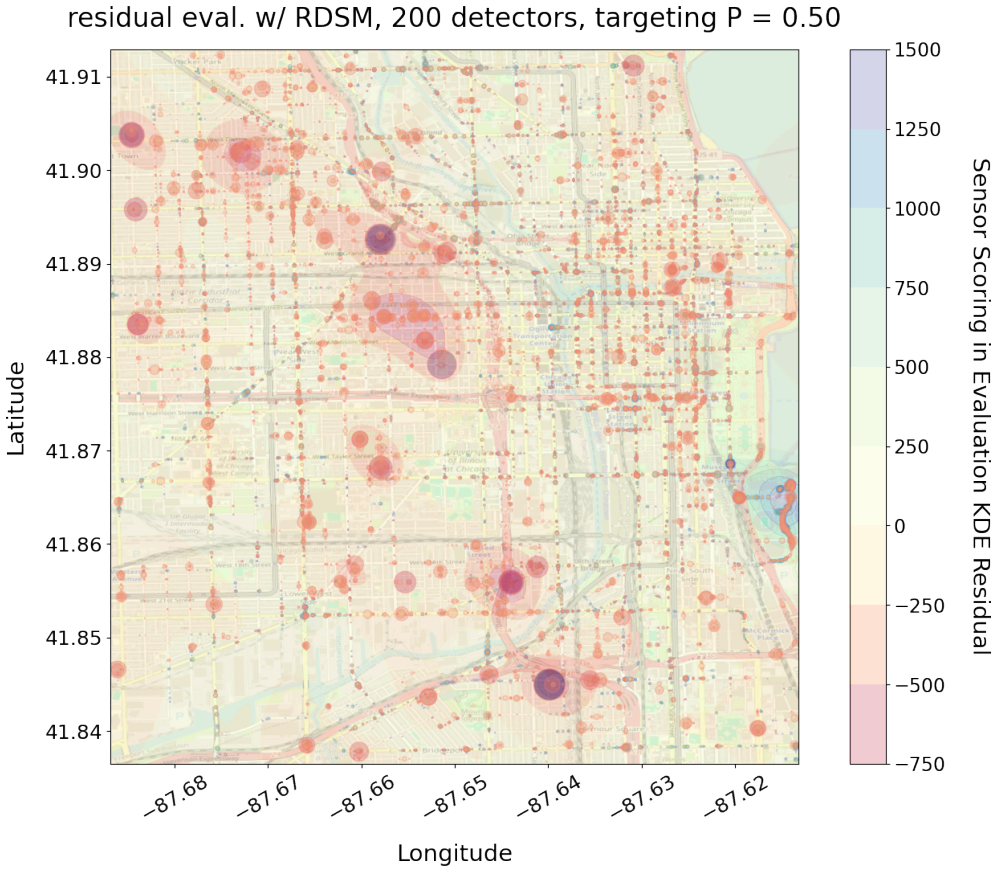
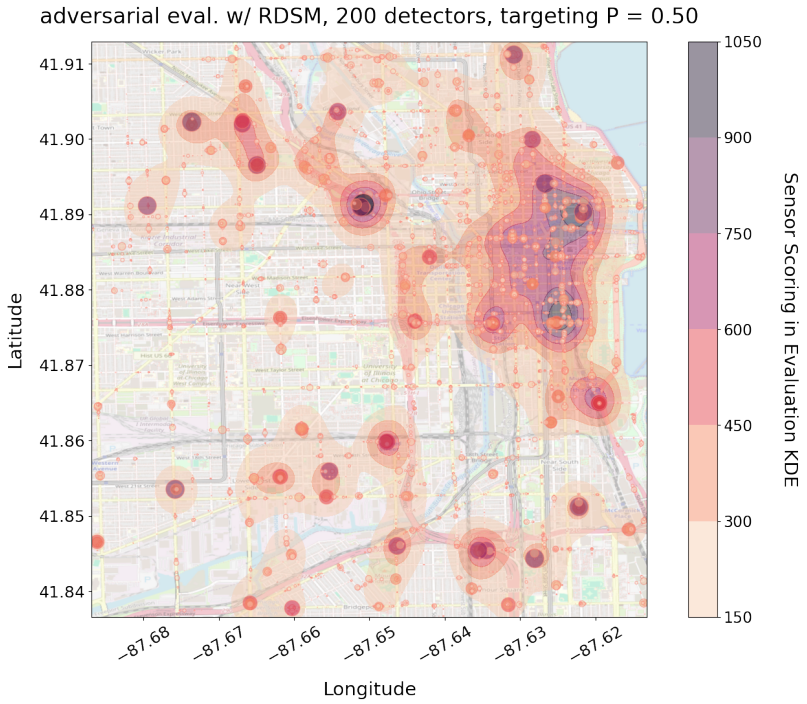
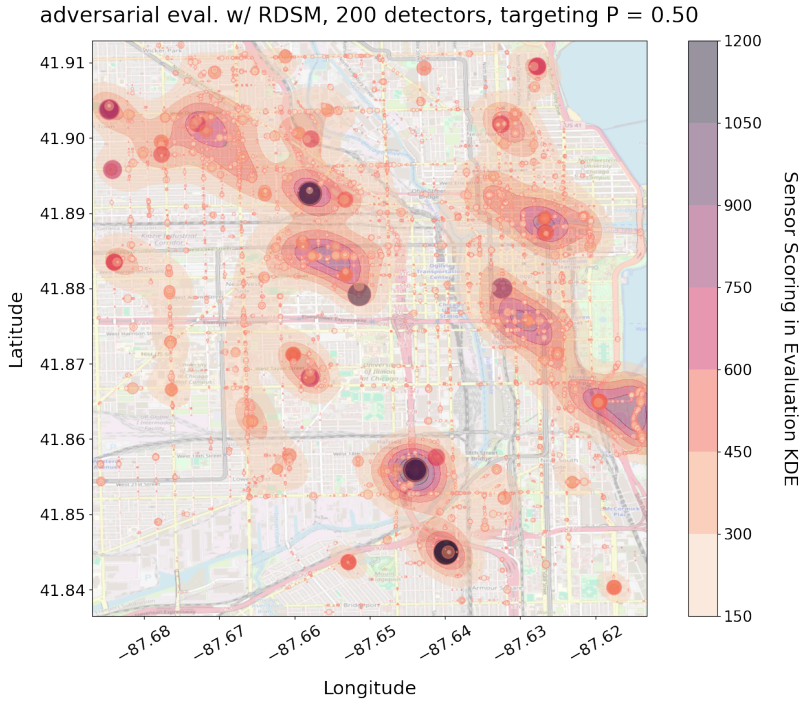


Figure G.31: Residual of detector position scoring KDE for evaluation using route dataset of 50% targeted routes with sensor networks of 200 allotted detectors each.



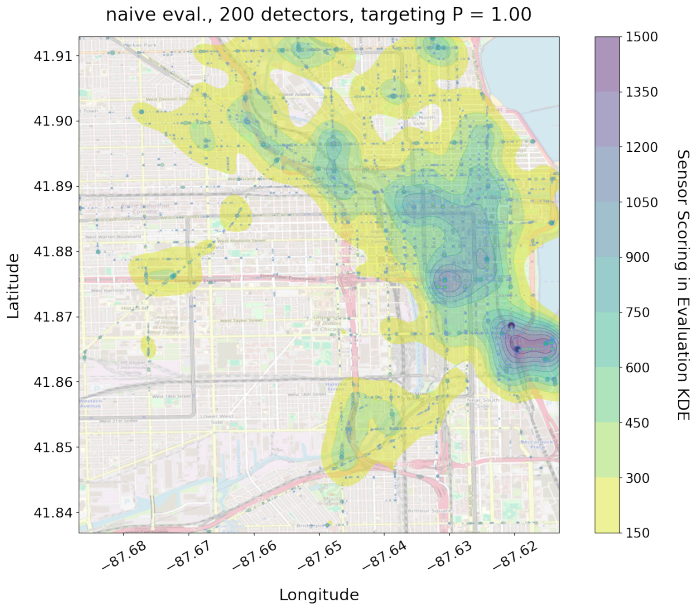
(a) Static strategies



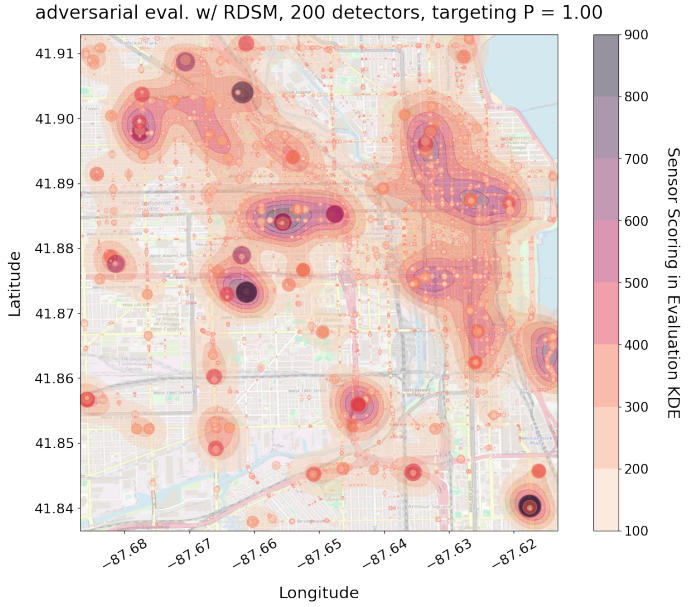
(b) Dynamic strategies

Figure G.32: Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 50% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown.

G.3.4 Route Dataset: 100% Targeting



(a) Naive Evaluation



(b) Adversarial Evaluation

Figure G.33: Detector position scoring with naive and adversarial pathing evaluation using route dataset of 100% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown.

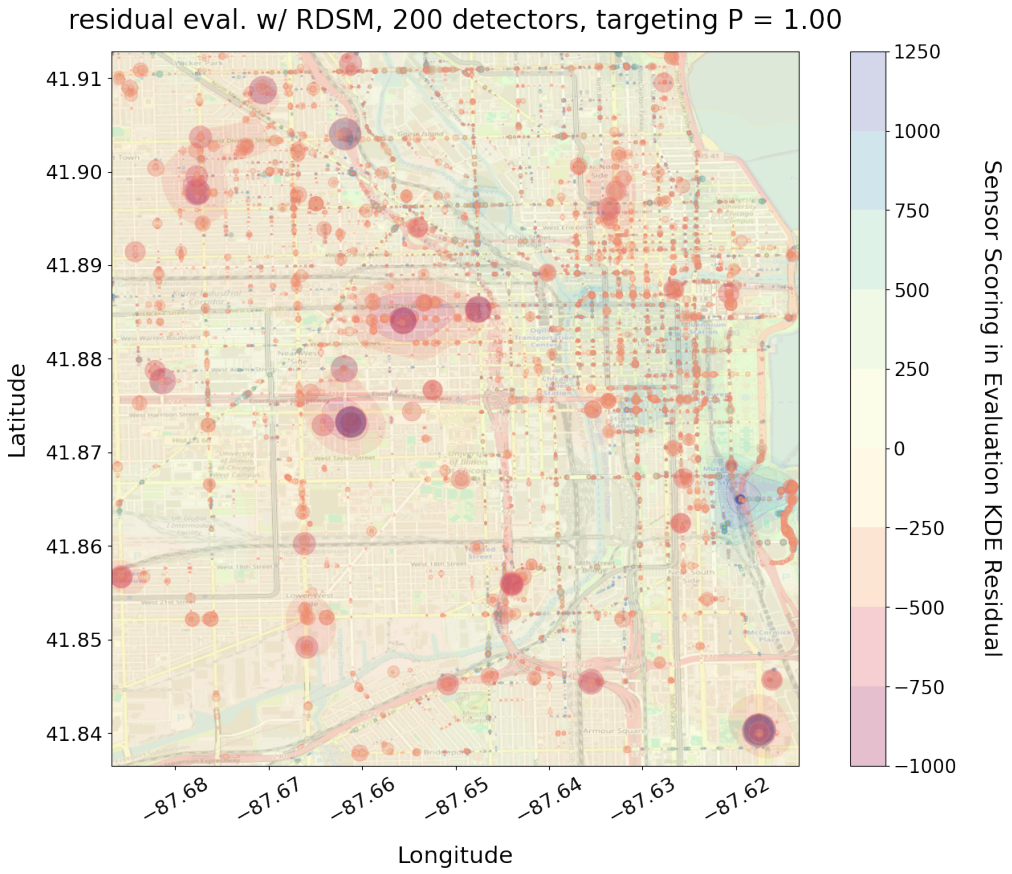
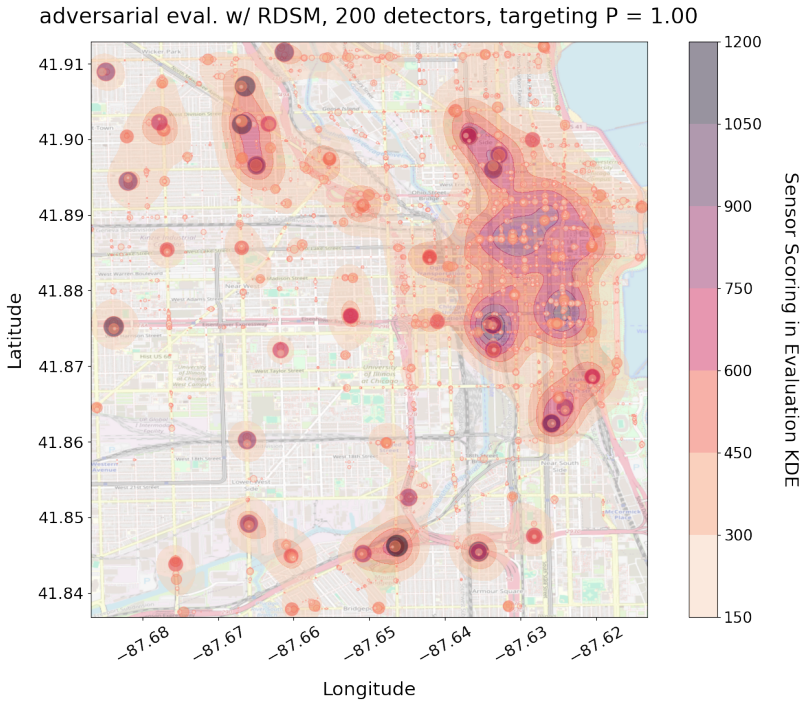
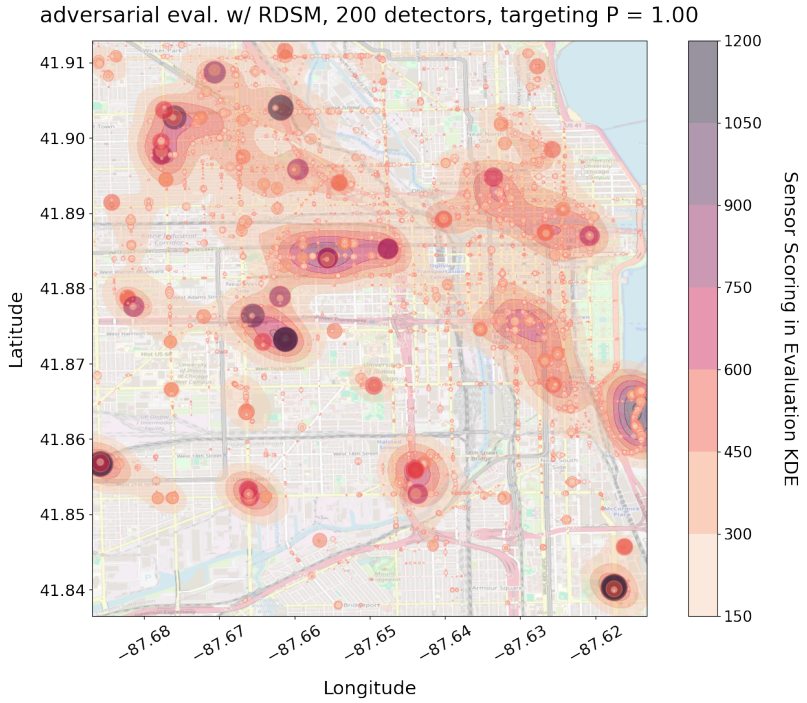


Figure G.34: Residual of detector position scoring KDE for evaluation using route dataset of 100% targeted routes with sensor networks of 200 allotted detectors each.



(a) Static strategies



(b) Dynamic strategies

Figure G.35: Detector position scoring with static (A, B, and C) and dynamic (COCO-EVO) strategies using adversarial pathing evaluation and a test route dataset of 100% targeted routes with sensor networks of 200 allotted detectors each. Scatter plot of detection locations and KDE shown.

Vita

David Raji (داود راجی) is from Atlanta, GA. He first enrolled at Georgia Institute of Technology in 2014 with a major in Computer Science but switched to Nuclear and Radiological Engineering after feeling that a connection to the physical sciences was missing in his studies. He graduated with a B.S. in that field in 2018. After a brief post-baccalaureate program with the Applied Nuclear Physics program (ANP) at Lawrence Berkeley National Laboratory (LBNL), he began graduate school in Nuclear Engineering at Georgia Institute of Technology. David transferred to University of Tennessee, Knoxville in January 2020 and has since worked on his graduate research in collaboration with ANP at LBNL. He earned his M.S. in Nuclear Engineering in August 2022. While initially focusing on radiation detection, David's research has stretched into a wide range of areas and he is very much hoping that trend continues in his future scientific endeavors.