Collaborative Position Location for Wireless Networks in Harsh Environments

Tao Jia

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Electrical Engineering

Committee Members:
Dr. R. M. Buehrer, Chair
Dr. Jeffery H. Reed
Dr. Timothy Pratt
Dr. Christopher Wyatt
Dr. Raghu Pasupathy

March 18, 2010

Key words: collaborative position location, non-line-of-sight (NLOS) propagation
Copyright 2010, Tao Jia

Collaborative Position Location for Wireless Networks in Harsh Environments

Tao Jia

(ABSTRACT)

Position location has become one of the more important tasks for improving communication and networking performance for future commercial wireless systems. It is also the enabling technology for many control and sensing applications envisioned by the wireless sensor networks (WSN). Despite its meaningfulness and many algorithms being developed in the past several years, position location in harsh propagation environments remains to be a challenging issue, due mainly to the lack of sufficient infrastructure support and the prominent phenomenon of non-line-of-sight (NLOS) signal propagation.

Recently, adopting the concept of collaborative position location has attracted much research interest due to its potential in overcoming the abovementioned two difficulties. In this work, we approach collaborative position location from two different angles. Specifically, we investigate the optimal performance of collaborative position location, which serves as a theoretical performance benchmark. In addition, we developed a computationally efficient algorithm for collaborative position location and incorporated an effective NLOS mitigation method to improve its performance in NLOS-dense environments. Overall, our work provides insight into both theoretical and practical aspects of collaborative position location.

Dedication

This work is dedicated to my parents and my wife.

Acknowledgment

First and foremost, I would like to thank my advisor, Dr. R. Michael Buehrer, for his guidance, help and commitment throughout my study here at Virginia Tech. I have benefited tremendously from his unique blend of knowledge and personality. I also would like to thank all my committee members for their suggestions on my dissertation.

I extend great gratitude to my colleagues in the Mobile and Portable Radio Research Group (MPRG), especially Swaroop Venkatesh, Rekha Menon, Haris Volos, Chris Phelps, Jeong-Heon Lee, Haesoo Kim, Justin Kelly, Umair Ahsan, Harpreet Singh Dhillon, Xuetao Chen, An He and many others who have together created an excellent work environment here. I would specially like to thank Swaroop Venkatesh for his help in the beginning of my PhD study.

I would like to particularly thank my parents and my wife for believing in me, and for their encouragement and support.

Contents

Abstract **ii**

Dedication **iii**

Acknowledgement **iv**

List of Figures **ix**

List of Tables **xix**

List of Acronyms **xx**

1 Introduction **1**

1.1 Position Location in Harsh Environments 1

1.2 Motivation for Collaborative Position Location 4

1.2.1 Challenges for Position Location in Harsh Environments 4

1.2.2 Collaborative Position Location 5

1.3 Specific Research Challenges for Collaborative Position Location 8

1.4 Contributions and Organization of this work 10

2 Background **13**

2.1 Problem Formulation 13

2.2 Measurement Techniques 14
2.3 Distance Measurement Model 16
2.4 Related Work: A Chronological Survey 18
2.5 A Taxonomy of Existing Algorithms 21

3 The CRLB for Collaborative Position Location **24**

3.1 Introduction 24
3.2 The CRLB for Collaborative Position Location 25
3.3 Numerical Results 28
3.4 Discussions 32
3.4.1 Distance-dependence of Measurement Noise 32
3.4.2 Path Loss Exponent 32
3.4.3 GDOP's Suitability for Indoor Localization 32
3.5 Summary 33

4 Optimal Performance of Collaborative Position Location **40**

4.1 Motivation and Related Work 40
4.2 The MLE for Collaborative Position Location 42
4.3 The BB/RLT framework for solving the MLE 43
4.3.1 BB/RLT algorithm 44
4.3.2 Reformulation and linearization of the MLE 46
4.3.3 Partitioning variables, initial bounding interval and partitioning rule . 50
4.3.4 Computational complexity and convergence speed-up techniques . . . 51
4.4 Numerical results 53
4.4.1 Validation of the BB/RLT formulation 55
4.4.2 MLE versus LLS estimator 59

4.4.3 MLE versus CRLB 60
4.4.4 MLE for collaborative position location 64
4.5 Summary 69

5 A Collaborative Quasi-Linear Programming Approach 70
5.1 Node collaboration with NLOS measurements 71
5.2 A Collaborative Quasi-Linear Programming Framework 74
5.2.1 System Models and Assumptions 74
5.2.2 LS estimator and LP approach 75
5.2.3 Handling the degenerate cases 79
5.2.4 Sequential location estimation 80
5.2.5 Node collaboration 80
5.3 Simulation results 83
5.4 Summary 91

6 A Set-Theoretic Approach to Collaborative Position Location 92
6.1 Motivation and Related Work 92
6.2 Problem Formulation 94
6.3 The Iterative Parallel Projection Method 95
6.3.1 PPM and non-collaborative position location 95
6.3.2 Iterative PPM for collaborative position location 102
6.4 Node localizability via iterative PPM 105
6.5 Numerical results 107
6.5.1 Localization accuracy 107
6.5.2 Algorithm running time 125
6.5.3 Localizability and location coverage 127

6.6 Summary 131

7 NLOS Mitigation for Collaborative Position Location 132

7.1 Motivation 132

7.2 Knowledge about the NLOS Propagation 133

7.3 NLOS Mitigation Method 134

7.3.1 NLOS identification only 134

7.3.2 NLOS identification plus the minimum NLOS bias 136

7.3.3 NLOS identification plus the mean NLOS bias 137

7.4 Simulation Results 139

7.4.1 Perfect NLOS identification 140

7.4.2 Imperfect NLOS identification 148

7.4.3 Imperfect NLOS identification plus inaccurate estimate of min and mean NLOS bias 152

7.5 Summary 154

8 Conclusions 156

8.1 Research summary 156

8.2 Future work 158

Bibliography 159

List of Figures

1.1 An example of position location in harsh environments: position location for commercial wireless devices in urban environments. If the laptop in the center of the figure is equipped with a GPS receiver, and wants to determine its location, it may be difficult to do so due to limited GPS satellite visibility among high-rise buildings. The dashed links represent non-line-of-sight (NLOS) propagation. 2

1.2 Another example of position location in harsh environments: wireless sensor network self-localization in a forested area. Only a few anchors with GPS capability can determine the locations. The presence of the forest and canopy causes the NLOS propagation. 3

1.3 An example to illustrate the concept of collaborative position location. In non-collaborative position location, only node A is able to determine its location. In collaborative position location, nodes B, C and D will also be able to determine their locations. The presence of the NLOS link between nodes C and D can degrade the localization accuracy and needs to be specially handled. 6

2.1 Illustration of the NLOS propagation problem. The two links shown by the dashed lines are obstructed by two objects, resulting in the NLOS propagation. In terms of TOA estimates, they will generally appear to be much larger than they should be. The corresponding range estimates are then positively biased. 17

3.1 Difference between the old CRLB with our new CRLB. The noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$. Note that *darker* regions correspond to those having *smaller* values. It can be seen that, with the knowledge of the variance model of range estimation noise, the new CRLB is lower than the old CRLB. 29

3.2 The CRLB of different nodes within the network, showing the effect of different values of β and K_E on the CRLBs (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$). Note the different scales on the two figures. The larger β and K_E, the more benefit we can get from knowing the variance model of range estimation noise. . . . 30

3.3 Comparing the LS estimator's performance with the CRLBs, for $K_E = 0.001$ and $\beta = 4$ (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$). The fact that the new CRLB is lower than the old one suggests that having the knowledge of distance-dependence model of the noise variance has the potential of improving the localization accuracy. In addition, the LS estimator, despite having a closed-form solution, still performs much worse than the CRLB suggests. 31

4.1 Generic branch-and-bound/reformulation linearization technique (BB/RLT) solution procedure. 45

4.2 Polygonal bounding region of $Z_{ij} = \sqrt{W_{ij}}$ for $W_{ij}^L = 0.2$ and $W_{ij}^U = 2$. The four tangential lines define a polyhedral bounding region for the square root function between W_{ij}^L and W_{ij}^U. 47

4.3 Improved polygonal bounding region of $Z_{ij} = \sqrt{W_{ij}}$ for $W_{ij}^L = 0.1$ and $W_{ij}^U = 200$. The two additional tangential lines (dash) E and F refine the bounding region and reduce the search space around the small value region of W_{ij}. . . 52

4.4 The convergence of the UB and LB on the MLE objective value for noise-free case, with 3 anchors at (5, 5), (10, 15) and (15, 5), and unlocalized node at (3, 5). The final objective value is found to be 1.17×10^{-6} and the solution is (2.999, 4.999). 53

4.5 Validation of the BB/RLT framework for solving the MLE in the absence of range estimation noise. The three anchors (solid circles) are located at (5, 5), (10, 15) and (15, 5), respectively. The red circles are different tested true locations of the unlocalized node, while the blue asterisks are the MLE solution obtained by the BB/RLT. As shown, the BB/RLT always finds the true solution, which validates our framework formulation. 54

4.6 Illustration of contours of the residual. The solution (3, 5) is the global optimum, while the solution (5, 3) is a local optimum and indeed a flipped location solution. The two solutions have very similar residuals. 55

4.7 Example estimated locations using the LLS and the MLE, where $K_E = 0.001$ and $\beta = 2$ ($\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$), and the unlocalized node's true location is (10, 10), with good anchor geometry. 57

4.8 Example estimated locations using the LLS and the MLE, where $K_E = 0.001$ and $\beta = 2$ ($\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$), and the unlocalized node's true location is (10, 10), with bad anchor geometry. 57

4.9 Mean localization errors for the LLS, the MLE, for the good anchor geometry in Fig. 4.7. K_E is the proportionality constant in modeling the noise variance, $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$. 58

4.10 Mean localization errors for the LLS, the MLE, for the bad anchor geometry in Fig. 4.8. K_E is the proportionality constant in modeling the noise variance, $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$. 58

4.11 Mean localization errors of the LLS and the MLE versus the noise variance σ^2 with good geometry, where the CRLB serves as a good benchmark. Four anchors are placed at four corners (0 0), (0 40), (40 0), (40 40) and the unlocalized node is at (20 20). 61

4.12 Mean localization errors of the LLS and the MLE versus the noise variance σ^2 with bad geometry, where the CRLB is no longer a meaningful benchmark. Four almost-collinear anchors are placed at (0 0), (10 11), (28 29), (40 40) and the unlocalized node is at (20 20). 61

4.13 Node geometry to demonstrate the bias, MSE, estimator variance of the MLE. The pentagons represent the testing locations of the third anchor, starting from $\alpha = 0^\circ$ to 360°. 62

4.14 Biases on x and y of the MLE with respect to different testing locations for the third anchor, indexed by α, for the node setup in Fig. 4.13. It is clear that the MLE is a biased estimator. 63

4.15 The comparison between the CRLB and the MSE of the MLE, as well as the comparison between the uniform CRLB (UCRLB) and the standard deviation of the MLE, with respect to α, for the node setup in Fig. 4.13. 64

4.16 The MLE solution computed by the BB/RLT and the lsqnonlin solver in Matlab, as well as SDP solution, for a particular network and noise realization. The noise variance is $\sigma^2 = 1\,\mathrm{m}^2$ 65

4.17 The convergence of the UB and LB on the MLE objective value for one noise realization for the network in Fig. 4.16. It takes much more iterations as well as running time for the BB/RLT to converge to the final solution. 66

4.18 The network-average localization error with respect to 50 different noise realizations for the network in Fig. 4.16. The noise variance is $\sigma^2 = 1\,\mathrm{m}^2$. The three additional horizontal lines represent the mean network-average localization error for the three methods, respectively. 67

4.19 The mean localization error versus range estimation noise for the network in Fig. 4.16, averaged over 100 noise realizations. As can be seen, the optimal performance suggested by the MLE solution obtained by the BB/RLT framework serves as a performance benchmark. The SDP outperforms the MLE by the lsqnonlin for small ranging noise, but the performance difference diminishes as ranging noise increases. 68

5.1 Illustration of the effect of node collaboration on the localization accuracy. Case 1: noncollaborative localization with only LOS range measurements. . . 71

5.2 Illustration of the effect of node collaboration on the localization accuracy. Case 2: collaborative localization with only LOS range measurements. 72

5.3 Illustration of the effect of node collaboration on the localization accuracy. Case 3: collaborative localization with LOS and NLOS measurements. 73

5.4 Illustration of lines formed by subtracting circular equations defined by LOS range estimates. 76

5.5 Illustration of creating virtual NLOS constraints over multihop connections. (a) For θ_m, adding range estimates r_{n1} and r_{nm} produces an artificial NLOS range estimate between anchor A_1 and θ_m, shown by the dashed line connecting the two, since it is very likely that the sum is greater than their true distance. The same mechanism is applied to generate other artificial NLOS range estimates between θ_m and other anchors, also for θ_n, shown by all the dashed lines. (b) Using the mechanism described in (a), θ_n is equipped with many artificial NLOS range estimates towards different directions, which has the potential of combating bad node geometry. 81

5.6 The complete CQLP algorithm. 82

5.7 CDF of the coverage of the sequential LS and the CQLP framework for 250 network realizations, when $N = 20$, $M = 9$, $R = 6\,\text{m}$. 84

5.8 CDF of root localization errors of the sequential LS and the CQLP framework for 250 network realizations, when $N = 20$, $M = 9$, $R = 6\,\text{m}$. 85

5.9 Average localization coverage (over 1000 network realizations) versus the number of unlocalized nodes, when the number of anchors is $M = 9$, the communication range $R = 6\,\text{m}$. Unlocalized nodes are randomly distributed into the area. 86

5.10 Average root localization error (over 1000 network realizations) versus the number of unlocalized nodes, when the number of anchors is $M = 9$ and the communication range $R = 6\,\text{m}$. Unlocalized nodes are randomly distributed into the area and each range estimate has a probability of 0.1 being NLOS, and if that happens, an NLOS bias uniformly distributed between 0 and 0.5 m is added onto the range estimate. 87

5.11 Results of applying the sequential LS and the CQLP with 3-hop node collaboration to localizing a large network with 100 unlocalized nodes (circles) and 10 anchors (squares), for one particular noise realization. The communication range is $R = 20\,\text{m}$ and the average node connectivity is 11.8 for (a) and (b), 12.4 for (c) and (d). The estimated locations are represented by asterisks and the dashed lines quantize the localization error. 89

5.12 The histogram of localization errors of nodes being localized in the first four iterations in the sequential LS localization approach. As we can see, the nodes being localized in the first iteration have smaller errors, while those being localized in later iterations have progressively larger localization error. 90

6.1 Generic unrelaxed parallel projection method, where a^{l+1} is simply the sum of a^l's weighted projections onto all convex feasibility sets, denoted by S_i's. $P_k(a_l)$ denotes the projection of a_l onto the kth convex set. $\{w_k\}$ is a set of convex weights. 97

6.2 Application of the modified PPM to non-collaborative position location for two cases: (a) less noisy range estimates where the intersection region of the three range estimates contains the true location; (b) one range estimate is extremely noisy. The blue circles denotes ranging circle and the dash dot lines represent the trajectories along which the solution of modified PPM converges. The initial guesses are denoted by the corresponding values noted $[xy]$. In (a), different initial guesses lead to the same solution, while in (b) different initial guesses result in different final solutions. 99

6.3 Comparison between original and modified PPM with the application to two cases of non-collaborative position location. Top row: 3 ranging circles do not intersect each other. Bottom row: 3 ranging circles intersect each other. (a) and (c): the results of original PPM. (b) and (d): the results of modified PPM. Solid lines represent rangine circles and circles are anchor locations. Stars represent different initial solutions and dashed lines represent the trajectories following which the PPM solution converges. Triangles represent final solutions. 100

6.4 Comparison between original and modified PPM with the application to another two cases of non-collaborative position location. Top row: one ranging circle is extremely large. Bottom row: 3 anchors are almost collinear. (a) and (c): the results of original PPM. (b) and (d): the results of modified PPM. Solid lines represent rangine circles and circles are anchor locations. Stars represent different initial solutions and dashed lines represent the trajectories following which the PPM solution converges. Triangles represent final solutions. 101

6.5 An example of applying the iterative PPM to collaborative position location for a network with 9 anchors (squares) and 50 unlocalized nodes (circles). We assume pure LOS scenario with $K_E = 0.0025$, $\beta = 2$ and $R = 8\,\text{m}$. The asterisks are the estimated locations and dash-dot lines quantize the estimation error. 105

6.6 An example of using iterative PPM to investigate node localizability. We only highlight the results for the 2nd and 3rd unlocalized nodes. The triangles and stars represent possible solutions of the 2nd and 3rd unlocalized nodes, respectively, resulting from 50 distinct initial solutions, for a particular noise realization. 107

6.7 The effect of the number of anchors on iterative PPM for: (a) 4 anchors at the four corners and 5 randomly within the network (RMSE = 0.34 m); (b) 4 anchors at the four corners (RMSE = 0.63 m); (c) 3 anchors within the network (RMSE = 2.37 m); Assume pure LOS range estimates, with $R = 8\,\text{m}$, $\beta = 2$ and $K_E = 0.01$, corresponding to 10% of inter-node distance. The circles and squares are the true locations of unlocalized nodes and anchors, respectively. The asterisks represent the estimates locations and the dashes lines quantize the localization error. 109

6.8 The corresponding mean localization errors versus K_E (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$. The larger K_E, the larger ranging noise is.) for the three cases in Fig. 6.7. 110

6.9 A regular-shaped random network with 200 unlocalized nodes (circles) and 20 anchors (squares). The communication range is $R = 15\,\text{m}$, the average node connectivity is 12.9 and the average number of connected anchors is 1.1. We will compare the localization accuracy of the iterative PPM, the SDP and the MDS-MAP(P, R) using this network. 111

6.10 Results of applying different localization schemes to the network in Fig. 6.9 for one particular noise realization in the pure LOS scenario with $K_E = 0.01$ and $\beta = 2$ (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$). 112

6.11 Cumulative density function of the localization errors of SDP, MDS-MAP(P, R) and iterative PPM for both pure LOS and 40% NLOS scenarios, with $K_E = 0.04$ and 0.16 (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$, $\beta_{ij} = 2$ for LOS and $\beta_{ij} = 3$ for NLOS), respectively. 113

6.12 The mean localization error versus the amount of range estimation noise (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$), for the network shown in Fig. 6.9, with pure LOS and 40% NLOS range estimates. K_E varies from 0.0025 to 0.16, corresponding to 5% up to 40% range estimation noise. 114

6.13 An E-shaped network with 200 unlocalized nodes (circles) and 20 anchors (squares). The communication range is $R = 15\,\text{m}$, the average node connectivity is 19.9 and the average number of connected anchors is 1.9. We will compare the localization accuracy of the iterative PPM, the SDP and the MDS-MAP(P, R) using this network. 116

6.14 Results of applying different localization schemes to the network in Fig. 6.13 for one particular noise realization in the pure LOS scenario with $K_E = 0.01$ and $\beta = 2$ (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$). 117

6.15 Cumulative density function of the localization errors of SDP, MDS-MAP(P, R) and iterative PPM for both pure LOS and 40% NLOS scenarios, with $K_E = 0.04$ and 0.16 (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$, $\beta_{ij} = 2$ for LOS and $\beta_{ij} = 3$ for NLOS), respectively, applying to the E-shape network. 118

6.16 The mean localization error versus the amount of range estimation noise (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$), for the network shown in Fig. 6.13, with pure LOS and 40% NLOS range estimates. K_E varies from 0.0025 to 0.16, corresponding to 5% up to 40% range estimation noise. 119

6.17 A ring-shaped network with 200 unlocalized nodes (circles) and 20 anchors (squares). The communication range is $R = 15\,\text{m}$, the average node connectivity is 20.7 and the average number of connected anchors is 1.7. We will compare the localization accuracy of the iterative PPM, the SDP and the MDS-MAP(P, R) using this network. 120

6.18 Results of applying different localization schemes to the network in Fig. 6.17 for one particular noise realization in the pure LOS scenario with $K_E = 0.01$ and $\beta = 2$ (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$). 121

6.19 Cumulative density function of the localization errors of SDP, MDS-MAP(P, R) and iterative PPM for both pure LOS and 40% NLOS scenarios, with $K_E = 0.04$ and 0.16 (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$, $\beta_{ij} = 2$ for LOS and $\beta_{ij} = 3$ for NLOS), respectively, applying to the ring-shape network. 122

6.20 The mean localization error versus the amount of range estimation noise (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$), for the network shown in Fig. 6.17, with pure LOS and 40% NLOS range estimates. K_E varies from 0.0025 to 0.16, corresponding to 5% up to 40% range estimation noise. 123

6.21 Localization results of iterative PPM with closest-anchor initialization, random initialization and MDS-MAP(P,R) and SDP for a network where anchors are not well spread into the network. Anchors and the true locations of unlocalized nodes are shown by squares and circles, respectively. The estimated locations are shown by asterisks and the dashed lines quantize localization error. The communication range is $R = 6$ m and the average node connectivity is 7.2. We assume pure LOS scenario and the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$ with $K_E = 0.09$ and $\beta_{ij} = 2$. 124

6.22 A network of 50 unlocalized nodes (circles) and 5 anchors (squares), where the solid lines represent node connectivity. We use this network to test our method to evaluate solution ambiguity and thus infer node localizability. . . 127

6.23 (a) the normalized values of the mean 100% solution radius, the CRLB and the GDOP for all unlocalized nodes; (b) the normalized values of mean localization error of the itrative PPM, MDS-MAP(P, R) and SDP. The simulation parameters are $K_E = 0.04$ and $\beta = 2$. (c), (d) and (e): the correlation (dot product) between the normalized mean localization errors of the three methods and the normalized mean 100% solution radius, the normalized CRLB and the normalized GDOP, with respect to K_E. 128

6.24 Using the 100% solution radius from one particular noise realization with $K_E = 0.04$ and $\beta = 2$ for the network in Fig. 6.22. If the target solution radius for nodes to be considered as localizable is 7 m, nodes 1, 18 and 42 will be considered as unlocalizable since they have a 100% solution radius larger than 7 m. 130

7.1 Using our proposed NLOS mitigation methods to localize the network in Fig. 6.9 for one noise and 40% NLOS bias realization. The dash lines quantize the localization errors and Ω is the network-average localization error. 141

7.2 Cumulative distributiuon function of the localization errors of the methods in Fig. 7.1. 142

7.3 Using our proposed NLOS mitigation methods to localize the network in Fig. 6.9 for one noise and 80% NLOS bias realization. The dash lines quantize the localization errors and Ω is the network-average localization error. 143

7.4 Mean localization error (averaged over 100 noise and NLOS bias realizations) versus the percentage of NLOS range estimates, when using our proposed NLOS mitigation methods to localize the network in Fig. 6.9, with NLOS bias uniformly distributed between $B_{\min} = 4\,\text{m}$ and $B_{\max} = 8\,\text{m}$. 145

7.5 Using our proposed NLOS mitigation methods to localize the network in Fig. 6.13 for one noise and 40% NLOS bias realization. The dashed lines quantize the localization errors and Ω is the network-average localization error. 146

7.6 Using our proposed NLOS mitigation methods to localize the network in Fig. 6.17 for one noise and 40% NLOS bias realization. The dashed lines quantize the localization errors and Ω is the network-average localization error. 147

7.7 Mean localization error (averaged over 100 noise and NLOS bias realizations) versus the percentage of NLOS range estimates with 5% NLOS identification error, when using our proposed NLOS mitigation methods to localize the network in Fig. 6.9, with NLOS bias uniformly distributed between $B_{\min} = 4\,\text{m}$ and $B_{\max} = 8\,\text{m}$. 149

7.8 Mean localization error (averaged over 100 noise and NLOS bias realizations) versus the percentage of NLOS range estimates with 20% NLOS identification error, when using our proposed NLOS mitigation methods to localize the network in Fig. 6.9, with NLOS bias uniformly distributed between $B_{\min} = 4\,\text{m}$ and $B_{\max} = 8\,\text{m}$. 150

7.9 Simulation results similar to Fig. 7.8, for the E-shape network in Fig. 6.13 . 151

7.10 Simulation results similar to Fig. 7.8, for the ring-shape network in Fig. 6.17 152

7.11 Simulation results of applying the proposed NLOS mitigation techniques to Fig. 6.9, in the presence of 20% NLOS identification error and larger or smaller estimate of the mean and the minimum NLOS bias. 153

List of Tables

4.1 The BB/RLT solutions and the final objective values of the MLE for an example of position location with three almost-collinear anchors 59

6.1 Total algorithm running time (seconds) of iterative PPM, MDS-MAP(P, R) and SDP, obtained in Matlab R2008a on an Intel 3.3 GHz CPU, for different number of unlocalized nodes and average node connectivity. 126

List of Acronyms

AOA	angle of arrival
BB	branch and bound
CRLB	Cramer-Rao lower bound
FIM	Fisher information matrix
GDOP	geometric dilution of precision
GPS	global positioning system
LOS	line-of-sight
LP	linear programming
LS	least-squares
MDS	multi-dimensional scaling
MLE	maximum likelihood estimate
NLOS	non-line-of-sight
pdf	probability density function
PDF	probability distribution function
PPM	parallel projection method
RLT	reformulation linearization technique
RMSE	root mean square error
RSS	received signal strength
SDP	semi-definite programming
SOCP	second order cone programming
TDOA	time difference of arrival
TOA	time of arrival
TOF	time of flight
WSN	wireless sensor network

Chapter 1

Introduction

1.1 Position Location in Harsh Environments

The past ten years have witnessed a tremendous proliferation of commercial wireless devices, such as cell phones, laptops, smart phones, smartbooks and stand-alone global positioning system (GPS) devices. The rapid development in wireless communication and networking technologies has enabled these devices to communicate with each other in an unprecedented scale. Among the many necessary technology advancements to boost on communication and networking capabilities, location discovery has emerged as one of the more important tasks. It has been demonstrated that accurate location information of wireless devices can greatly improve the performance of other communication or networking tasks, e.g., location-aided routing (LAR) [1], as well as provide new commercial revenue opportunities, e.g., location based services (LBS) [2]. However, determining the locations of wireless devices is a nontrivial task, especially for indoor or urban environments. A simple example is illustrated in Fig. 1.1, where we want to determine the locations of several wireless devices in a typical urban environment. Clearly, it is relatively easy for a GPS unit or E-911 enabled cellphone to determine its location with the help of either GPS satellites [3] or cellular base stations [4]. However, for those devices without built-in position location functionality, e.g., the laptop in the center of the figure, determining its location is not easy. Even if a device is equipped with GPS receivers or if E-911 is enabled, its location accuracy can be significantly deteriorated due to the limited visibility of GPS satellites or base stations. In some extreme cases, *i.e.*, if the device is in a building or goes underground, it may not be able to obtain a position fix at all.

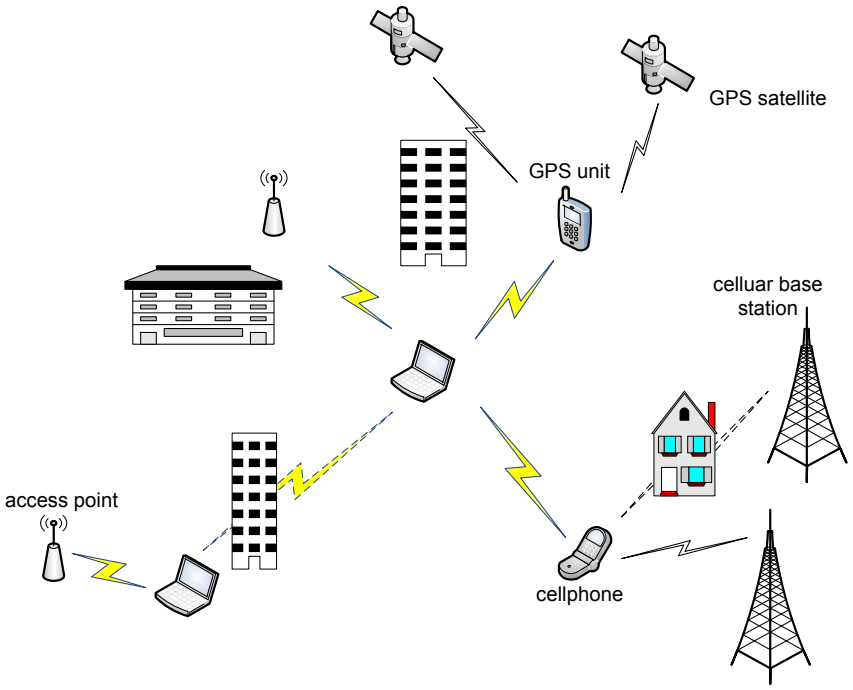


Figure 1.1: An example of position location in harsh environments: position location for commercial wireless devices in urban environments. If the laptop in the center of the figure is equipped with a GPS receiver, and wants to determine its location, it may be difficult to do so due to limited GPS satellite visibility among high-rise buildings. The dashed links represent non-line-of-sight (NLOS) propagation.

On the other hand, in the military and public safety domains, the deployment of networks of low-cost, low-power, small form factor and multi-functional wireless sensors is creating enormous opportunities for monitoring and control applications, including intrusion detection, military personnel or first-responder tracking, and scientific applications [5]. For the successful deployment of such wireless sensor networks (WSN), self-localization[1] is a critical prerequisite, simply because any subsequently sensed data or detected events become meaningless without the location information of the reporting sensors. A straightforward but naive way to enable the location-awareness for a WSN is to equip each sensor with a GPS receiver. However, the associated device cost, complexity and power consumption are prohibitively high, and thus prevent a large scale deployment of the WSN. Further, even if all the sensors were equipped with GPS receivers, it is most likely that the majority of them

[1]In this work, we will use the terms *position location* and *localization* interchangeably.

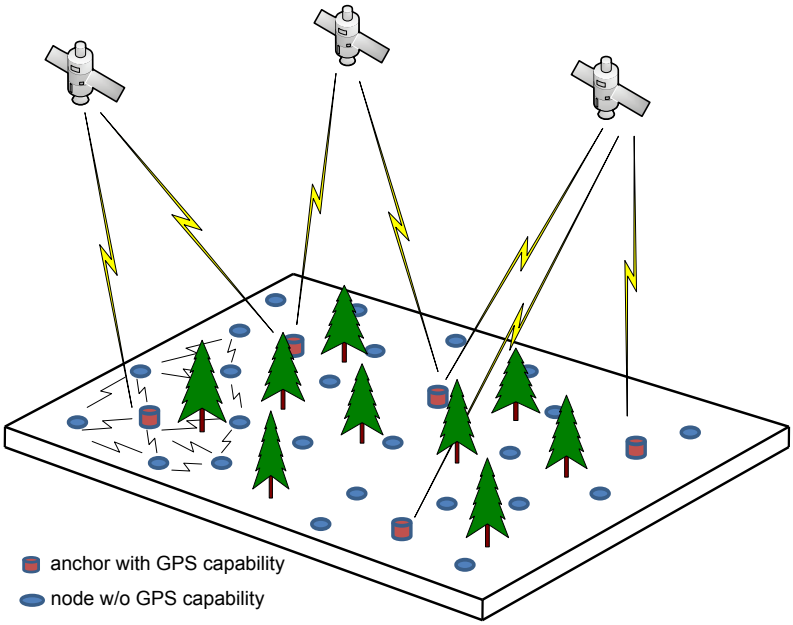


Figure 1.2: Another example of position location in harsh environments: wireless sensor network self-localization in a forested area. Only a few anchors with GPS capability can determine the locations. The presence of the forest and canopy causes the NLOS propagation.

may not have GPS visibility in evironments such as a heavily forested area and thus still cannot obtain a position fix. Alternatively, relying on other pre-existing infrastructure, e.g., base stations as in an E-911 system, or manually placing sensors at pre-determined locations seems plausible, but significantly limits the applicability of WSN in some hostile environments, such as battlefields, heavily forested areas or a burning building where fire-fighters are about to enter. Taking all these into account, it is highly desirable that a network of wireless sensors, if being distributed into a harsh environment with little support from a conventional infrastructure-based system, is able to determine the node locations. An illustrative example is given in Fig. 1.2, where a network of wireless sensors wants to determine its location with the help of a few GPS-enabled sensors and the available inter-sensor wireless communication links. The presence of the forest and canopy would significantly limit the communication capability and degrade the localization accuracy.

1.2 Motivation for Collaborative Position Location

In this section, we first present several unique challenges for position location in harsh environments, which have motivated the concept of collaborative position location. Then, we will describe in more detail what the collaborative position location is and give a simple example to illustrate the concept and several heuristic schemes.

1.2.1 Challenges for Position Location in Harsh Environments

The examples in Section 1.1 illustrate the research problem we will investigate in this work. That is, *how do we locate a network of nodes*[2] *in a harsh environment, which has limited support from existing infrastructure and deteriorated propagation conditions.* The two key phrases, "limited support" and "deteriorated propagation conditions", imply several unprecedented challenges to the design of a position location scheme. In conventional infrastructure-based position location system, such as GPS [3] or an E-911 system [4], wireless devices measure signals from visible GPS satellites or cellular base stations to obtain their location estimate. However, as we mentioned above, equipping a node with a GPS receiver or deploying pre-existing infrastructure turns out to be an expensive and impractical solution in many applications. Therefore, in the position location problem of interest, only a small portion of all network nodes can be expected to have known locations (these nodes are commonly referred to as *anchors* or reference nodes, and those who initially do not know their locations are called unlocalized nodes). On the other hand, it is usually assumed that wireless nodes have a limited communication range for the sake of energy efficiency and longer battery life. These two facts together lead to a unique challenge to position location in harsh environments, that is, the majority of nodes do not have enough anchors within their communication range to make measurements. Consequently, these nodes cannot *directly* estimate their locations and the position location scheme has to rely on other mechanisms to propagate the known location information throughout the network, which is indeed a challenging design issue.

A second challenge stems from the fact that most applications we will consider are expected to be in environments with deteriorated propagation conditions, e.g., indoor, urban canyon or forested areas. In such environments, a prominent phenomenon is non-line-of-sight (NLOS)

[2]From now on, we will use a more general term *nodes* to simply refer to devices within a wireless network, which can be laptop, cellphones, GPS units or wireless sensors.

signal propagation, which refers to the case where the signal along the direct line-of-sight (LOS) path between the transmitting and the receiving nodes is either blocked or significantly attenuated. NLOS propagation often leads to largely biased measurements, which in turn results in erroneous position estimation. The NLOS problem is exacerbated if the location information has to be propagated when there is limited infrastructure.

A third challenge is related to the notion of localizability, *i.e.*, whether a node is uniquely localizable or not. This a result of the first two challenges, but is highlighted here because of its great practical importance. It is well known that in 2-dimensional (2D) distance-based localization, an unlocalized node needs distance measurements to at least 3 non-colinear anchors to obtain a unique location estimate. However, due to the combined effects of an insufficient number of directly-connected anchors and the presence of measurement noise and NLOS bias, node localizability is made intertwined with the *solution reliability*, *i.e.*, how trustworthy a location solution is, and cannot be investigated based solely on connectivity information as in conventional infrastructure-based position location.

Besides these critical challenges, other interesting issues, such as the design of computationally efficient and scalable algorithms for large network deployment, incorporating multiple types of measurement data, and tracking mobile nodes, etc., all make the position location in harsh environments a nontrivial problem.

1.2.2 Collaborative Position Location

Over the past several years, there have been many approaches proposed or experimental systems developed for position location in harsh environments. For example, the Cricket location-support system [6], RADAR [7], the anchor-centroid based localization system in [8] and the APIT algorithm proposed in [9] are some notable earlier developments. However, these systems either require many powerful anchors to transmit beacons within the network [6, 8], or need an extensive pre-deployment measurement campaign [7], which are not suitable for the envisioned applications in this work. In addition, it has been widely acknowledged that using range estimate information or bearing information, rather than connectivity information only as in [8, 9], for position location leads to a fine-grained localization system with improved localization accuracy, despite the higher device and computational complexity. In fact, there has been a significant amount of research work in this regard and a detailed review of existing algorithms can be found in [10, 11, 12]. Among them, a notable common notion is that, the system should use not only the measurements between unlocal-

ized nodes and anchors, but also those among unlocalized nodes to estimate their locations. This concept has been referred to as *collaborative* or *cooperative* localization (or multihop localization, or sometimes simply as network localization [11]), to emphasize the use of additional measurements among unlocalized nodes to improve both localization accuracy and location coverage. In contrast, we refer to the conventional infrastructure-based position location approaches, e.g., GPS, E-911 or similar systems with an unlocalized node estimating its location based on measurements to directly connected anchors, as *non-collaborative* position location.

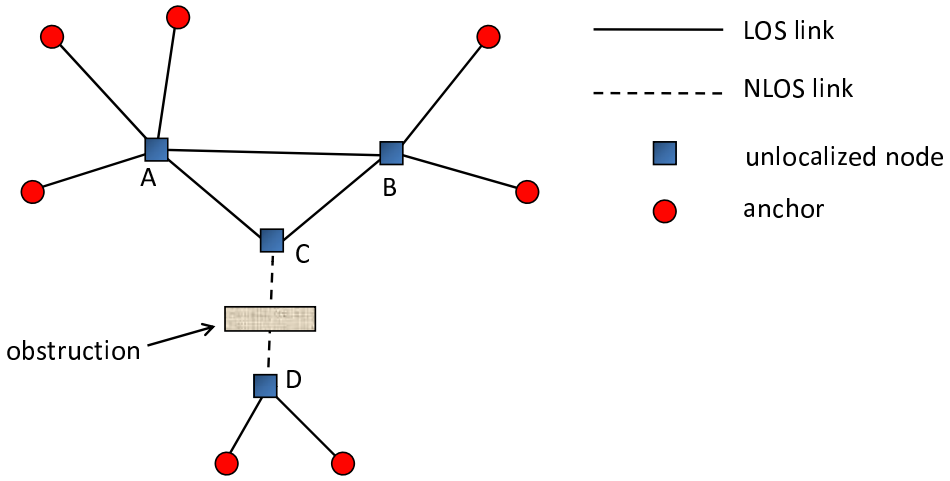


Figure 1.3: An example to illustrate the concept of collaborative position location. In non-collaborative position location, only node A is able to determine its location. In collaborative position location, nodes B, C and D will also be able to determine their locations. The presence of the NLOS link between nodes C and D can degrade the localization accuracy and needs to be specially handled.

We now use Fig. 1.3 as an example to illustrate the basic concept and some of the benefits from collaborative position locaiton. It is well known that for 2D distance-based position location, an unlocalized node needs range estimates to at least three non-colinear anchors to estimate its location without ambiguity. Based on this knowledge, it can be easily concluded that in Fig. 1.3, only node A is able to obtain its location estimate and its localization accuracy depends entirely on range measurements to its three directly-connected anchors. This is in fact what we referred to as non-collaborative position location. On the other hand, in collaborative position location, those range estimates among the four unlocalized nodes

will also be used. Below, we give a few heuristic schemes to demonstrate possible ways to utilize these measurements.

Scheme 1: Node A estimates its location first, then it declares itself as a virtual anchor. With the range measurement between A and B, node B now has a sufficient number of anchors and can thus estimate its location. Yet, nodes C and D cannot be localized.

Scheme 2: Relying on some packet flooding mechanism within the network, each unlocalized node obtains an approximate distance measurement to each of the anchors, even though they are not directly connected. In doing so, nodes B, C and D obtain enough range measurements to anchors and can then estimate their locations.

Scheme 3: Each unlocalized node uses its local information only to build a local coordinate system. Then all nodes collaborate with each other to merge their local coordinate systems and construct the global coordinate system.

Scheme 4: Setup a global optimization framework, with the objective of minimizing the sum of the squared differences between the measured distances and the distances calculated using estimated locations, including those among unlocalized nodes. This will jointly compute locations for all four unlocalized nodes.

It is straightforward to see some of the drawbacks associated with each of these schemes. For instance, Scheme 1 suffers from the problem of localization error propagation, *i.e.*, node A acts as an anchor, but in fact has an erroneous location estimate and this error will be passed on to subsequently localized nodes. This idea has been applied in existing work such as [14, 32]. Scheme 2 not only needs much higher communication overhead, but also in general results in coarse localization accuracy because of the approximated distance measurements. The concept has appeared in earlier works such as [8] and in more refeind form in later works [16, 17]. Scheme 3 practically needs high node connectivity and regularly shaped, e.g., isotropic, networks. The idea has found application in [20, 35]. Scheme 4, on the other hand, has the problems of scalability and the possibility of converging to a local instead of the global optimum. In fact, the objective of minimizing the sum of squared differences between the measured distances and the distances calculated using estimated locations is shown to yield the MLE in the case of Gaussian range estimation noise [24]. Moreover, the presence of NLOS distance measurements, as illustrated in Fig. 1.3, will affect all four schemes and may significantly degrade the localization accuracy if not properly handled.

Despite the drawbacks associated with these four schemes, we can clearly see the basic rationale of collaborative position location. That is, by letting unlocalized nodes collaborate with each other, we can increase the location coverage and potentially improve the localization accuracy.

1.3 Specific Research Challenges for Collaborative Position Location

Despite the novelty and research activities in collaborative position location, the aforementioned challenges have not been completely tackled and several fundamental questions remain unanswered. First and foremost, although many collaborative position location algorithms have been proposed, whether the computation is carried out in a centralized fashion, e.g., [15, 25, 30] or a distributed fashion, e.g., [14, 16, 18, 32], the fundamental question regarding the *optimal* performance of collaborative position location has not been answered. Specifically, although the closed-form expression of the maximum likelihood estimate (MLE) for collaborative position location is given in [24] and it is well known the MLE is optimal in the mean squared error (MSE) sense, there is generally no closed-form expression for the global optimum solution, due mainly to the nonlinear and nonconvex nature of the associated optimization problem, where classical gradient-search-based methods may return a local instead of the global optimum. Consequently, the authors in [25] and [30] resort to semi-definite programming (SDP) and second-order cone programming (SOCP) relaxations, respectively, to the original MLE problem and obtain the corresponding globally optimal solutions. However, both SDP- and SOCP-relaxed problems are no longer the original MLE and thus do not demonstrate the optimal performance. The authors in [36] further formulated the MLE in the presence of NLOS bias, but the solution procedure still does not guarantee global optimality. The authors in [37] presented the MLE for collaborative localization using RSS measurements. However, the MLE is only evaluated via a grid-search based numerical strategy, and thus is not guaranteed to be optimal. On the other hand, metaheuristic algorithms, e.g., simulated annealing (SA), particle swarm optimization (PSO) or genetic algorithms (GA), are proposed to address the solution optimality issue [38, 39, 40]. Nevertheless, it is known that these methods cannot guarantee the solution optimality either, despite the much higher computational complexity and their better chances of finding the globally optimal solution. **Therefore, there is a need to investigate the optimal performance of collaborative position location.**

In addition, although there are many localization algorithms that have explored the idea of node collaboration in different ways, most of them have not considered the issue of NLOS propagation. This is illustrated by the fact that simulation results of most existing algorithms have only been presented in the presence Gaussian range estimation noise. The authors in [36] did consider the NLOS issue and formulated the associated MLE, but have not presented extensive performance results, particularly for different network topologies and NLOS situations. Besides, the required centralized computation of the proposed algorithm does not scale well for networks with a larger number of nodes. It is worth mentioning that there exist methods to mitigate the adverse effect of the NLOS propagation, mainly in the context of mobile location for cellular systems, e.g., [41]-[47], and some more recent developments [48, 49]. However, these approaches are either computationally too expensive or have additional requirements on the minimum number of anchors, and thus are not well-suited for the envisioned applications. **In view of the state-of-the-art and considering the prominent NLOS conditions in the applications of interest, we believe that the performance results of different collaborative position location algorithms in the presence of NLOS propagation are indispensable in order to make any algorithm comparison meaningful and practical. Additionally, there is a need for an effective and low complexity NLOS mitigation method that is ready for large scale deployment**

Another important aspect of collaborative position location is the algorithm efficiency in terms of both computational complexity and network scalability. So far, the SDP approach [26, 27] and the MDS-MAP(P, R) [20] using range estimates presented the most comprehensive performance results and are among the best available in terms of localization accuracy. However, the SDP has high computational complexity and does not scale well as the number of nodes and the average node connectivity increase. The MDS-MAP approach is originally proposed as a centralized method. Its distributed version MDS-MAP(P, R) is developed to deal with the scalability issue and demonstrates good performance only for regular isotropic network topologies. A few other developments, e.g., [31] and [32], achieve comparable performance and are based on distributed computation. **Nevertheless, there is still a need to develop computationally more efficient, scalable collaborative position location algorithms for large networks to achieve improved localization accuracy across different range estimation noise scenarios, NLOS conditions and network topologies.**

Finally, as far as the node localizability issue is concerned, the authors in [50]-[52] formulated network localization as a graph realization problem and investigated the issue of node localizability via the theory of graph rigidity. However, in their work, both range estimation noise and NLOS bias are ignored. This renders the derived conditions for node localizability less instructive in practical situations. In this regard, the framework of probabilistic localization recently adopted in [53, 54, 55, 56], is of interest. Specifically, network localization is formulated as a statistical inference problem, where the goal is to compute (approximate) the probability distribution of each node's location based on the given measurements. Unlike other algorithms, the probabilistic position location generates *a set of possible location estimates*, namely particles, rather than a single location estimate. Each particle can be associated with a *weight*, which quantifies the uncertainty about the corresponding location estimate. Existing metrics such as the distance residual [14], geometric dilution of precision (GDOP) and error ellipses [3], although related to the solution uncertainty, cannot handle the presence of NLOS bias and are mainly useful for the case where an unlocalized node has enough anchors, e.g., at least four GPS satellites in the case of GDOP for 3D localization. In this sense, the probabilistic position location framework is naturally suited for handling node localizability and the preliminary results in [53] have demonstrated that nodes which end up with bimodal particle distributions can be regarded as being not localizable. **However, the computational complexity of the probabilistic position location is prohibitively high and we believe that there exist numerically simpler alternatives to achieve the same goal.**

1.4 Contributions and Organization of this work

Considering the potential and state-of-the-art of collaborative position location, our primary goals in this research are divided into two aspects, namely, theoretical performance limits and practical algorithm design. In particular, our contributions are summarized as follows:

- Chapter 3: We derive a new Cramer-Rao Lower Bound (CRLB) for collaborative position location, which takes the range estimation noise model into account and gives some insight into the benefit of having the knowledge of noise model and the impact of node geometry;
- Chapter 4: We formulate and develop a branch-and-bound/reformulation linearization technique (BB/RLT) based framework to solve the MLE for collaborative position

location. This is the first work that solves the MLE for collaborative position location with guaranteed solution optimality. We also show that the optimal performance of the MLE is a more meaningful performance benchmark than the CRLB, especially in the case of bad node geometry;

- Chapter 5: We develop a collaborative quasi-linear programming (CQLP) framework to deal with both node collaboration and NLOS propagation and demonstrate the error-propagation issue associated with the paradigm of sequential position location, especially for large networks;

- Chapter 6: We design an iterative and distributed collaborative position location method based on the parallel projection method (PPM), which achieves comparable and sometimes better localization accuracy than existing methods, but with significantly less computational complexity and much better network scalability. In addition, we show that it can be used to investigate node localizability in practical situations with much less computational effort than probabilistic localization;

- Chapter 6: We provide a comprehensive performance comparison between our method and state-of-the-art methods in the presence of non-line-of-sight(NLOS) propagation and across different network topologies;

- Chapter 7: We develop an NLOS mitigation method based on different levels of knowledge about the NLOS conditions. The method is incorporated into our iterative PPM method, demonstrates good performance and is suitable for large network deployment.

Published, submitted and to-be-submitted papers resulted from this work are as follows:

- Tao Jia and R. M. Buehrer, "NLOS Mitigation for Collaborative Position Location," to be submitted, *IEEE Transactions on Vehicular Technology*, Mar. 2010.

- Tao Jia and R. M. Buehrer, "A Set-theoretic Approach to Collaborative Position Location for Wireless Networks," in revision, *IEEE Transactions on Mobile Computing*, Jan. 2010.

- Tao Jia and R. M. Buehrer, "On the Optimal Performance of Collaborative Position Location," *IEEE Transactions on Wireless Communications*, vol. 9, no. 1, pp. 374-383, Jan. 2010.

- Tao Jia and R. M. Buehrer, "Distance-based Collaborative Position Location using Parallel Projection Method," to be submitted, *IEEE Global Telecommunications Conference (GLOBECOM)*, Mar. 2010.
- Tao Jia and R. M. Buehrer, "NLOS mitigation in Collaborative Position Location for large Wireless Sensor Networks," to be submitted, PIMRC workshop on Indoor Position Location, 2010.
- Tao Jia and R. M. Buehrer, "A New Cramer-Rao Lower Bound for TOA-based Localization," Proceedings of *IEEE Military Communications Conference (MilCom)*, Nov. 2008.
- Tao Jia and R. M. Buehrer, "A Collaborative Quasi-linear Programming Framework for Ad Hoc Sensor Localization," Proceedings of *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 2379-2384, Apr. 2008.
- R. M. Buehrer, S. Venkatesh and T. Jia, "Mitigation of the propagation of localization error using multi-hop bounding," Proceedings of *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 3009-3014, Apr. 2008.

The rest of this work is organized as follows. In Chapter 2, we give some necessary background information on the general position location problem, e.g., measurement techniques, distance measurement model, as well as background specifically on collaborative position location. In addition, we provide a brief survey and taxonomy of existing work over the past ten years. In Chapter 3, we present our work on the fundamental limits on collaborative position location via the CRLB. In Chapter 4, we present our theoretical work on solving the MLE for collaborative position location with guaranteed solution optimality. In Chapter 5, we describe one of our practical algorithms for collaborative position location, which demonstrates the error-propagation issue associated with the paradigm of sequential position location. In Chapter 6, we develop an iterative and numerical method based on the PPM and provide a performance comparison with existing methods. In Chapter 7, we present our NLOS mitigation method that is particularly designed for the collaborative position location method proposed in Chapter 6. Finally, in Chapter 8, we draw conclusions and suggest future work.

Chapter 2

Background

In this chapter, we provide some necessary background information on both the general position location problem and specifically collaborative position location. We first give a formal mathematical formulation of the problem of interest. Then we describe measurement techniques available, followed by the mathematical model we used for distance measurements. In addition, we demonstrate the basic concept of collaborative position location through a few simple examples. Finally, we review existing algorithms for collaborative position location, followed by a taxonomy of and comparison between these algorithms. Notations and definitions given in this chapter will be used throughout this work. Note that to be more consistent and facilitate a fair comparison with the majority of the existing work, we will primarily focus on 2D position location.

2.1 Problem Formulation

Our research problem can be stated as: To determine the location of a network of nodes in a harsh environment with limited support from existing infrastructure and in the presence of deteriorated propagation conditions. In such a context, each node within the network is viewed as a point, located on a 2D plane. Nodes with unknown positions are referred to as unlocalized nodes, denoted by $\Theta = [\theta_1^T, \theta_2^T, \ldots, \theta_N^T]$ and N is the number of unlocalized nodes. On the other hand, nodes with known locations are referred to as anchors, whose coordinates are denoted by $\mathbf{A} = [\theta_{N+1}^T, \theta_{N+2}^T, \ldots, \theta_{N+M}^T]$ and M is the number of anchors. In practice, anchor locations can be known by either equipping the nodes with GPS receivers or

manually placing them at pre-determined locations. $\theta_i = [x_i \; y_i]^T$ is the coordinate of the ith node, for $i = 1, 2, \ldots, N+M$. Furthermore, we use $\mathcal{X} = \{X_{ij} \,|\, 1 \le i \le N,\, 1 \le j \le N+M\}$ to denote the set of inter-node measurements, where X_{ij} is the measurement between the ith and the jth nodes. Depending on whether two nodes are able to communicate with each other[1], a particular measurement may or may not exist in $\mathcal{X}$. If the ith and the jth nodes can communicate with each other and obtain a measurement, we refer to them as being connected or neighbors of each other, denoted by $j \in \mathcal{N}(i)$ and vice versa, where $\mathcal{N}(i)$ is the set of the ith node's neighbors. In addition, we assume inter-node measurements are symmetric[2], *i.e.*, $X_{ij} = X_{ji}$. Using these notations, the task of position location can be stated as to obtain location estimates $\hat{\Theta} = [\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_N]$ for the unlocalized nodes, given the set of anchor positions $\mathbf{A}$, and the set of available inter-node measurements $\mathcal{X}$.

To evaluate the performance of a position location scheme, we need to define a metric. Given a noise realization and the resulting location estimate $\hat{\Theta}$, the *network-average* error Ω is defined as the square root of the average squared localization error

$$\Omega \;\triangleq\; \sqrt{\frac{1}{N}\sum_{i=1}^{N}\big[(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2\big]}, \tag{2.1}$$

where $[\hat{x}_i \; \hat{y}_i]^T$ denotes the ith node's estimated location in $\hat{\Theta}$. The *mean* localization error $\overline{\Omega}$, on the other hand, refers to the square root of the network-average MSE, *i.e.*,

$$\overline{\Omega} \;\triangleq\; \sqrt{\frac{1}{N}\sum_{i=1}^{N}\mathbf{E}\big\{(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2\big\}}, \tag{2.2}$$

where $\mathbf{E}\{\cdot\}$ denotes the expectation over different noise realizations.

2.2 Measurement Techniques

Measurements for position location can be divided into three categories: distance related measurements, angle-of-arrival (AOA) measurements and received signal strength (RSS) fingerprinting [11]. Distance related measurements can be typically obtained from estimates of

[1] In practice, this depends on factors such as the devices' maximum communication range, obstacles between devices, etc.

[2] This can be true when measurements obtained by two neighboring nodes are averaged before being used for location estimation.

time-of-arrival (TOA) with accurate clock synchronization or time-of-flight (TOF) obtained from round trip propagation time measurement, time-difference-of-arrival (TDOA) and RSS. One thing to note is that using RSS to obtain distance estimates requires a proper propagation model, which can be difficult to obtain or inaccurate in practice. AOA measurements can be obtained in two ways. The first relies on the use of the receiving antenna's amplitude response and translates the received signal amplitude to an arrival angle estimate according to the known antenna pattern. The second uses an antenna array to measure phase differences in the array and then estimates the incoming signal's arrival angle. RSS fingerprinting, on the other hand, relies on an extensive pre-deployment measurement campaign to construct a database of the signal strengths at different testing locations, received from multiple transmitters within the coverage area. In doing so, a mapping from a vector of RSS estimates to a location is established and can be used for online position location. Since our main focus in this research is on designing position location algorithms *given* inter-node measurements, we will not discuss the details of these measurement techniques and refer interested readers to [11] and references therein for an extensive review. Instead, we only briefly describe popular techniques to obtain distance-related measurements.

We need to emphasize that we will primarily focus on using TOA-based inter-node distance measurements. This is not only to be more consistent with the majority of the existing literature, thus facilitating a fair comparison between our and existing position location schemes, but also due to several practical considerations. First, compared to the RSS fingerprinting techniques, distance related measurements or AOA measurements do not require an extensive measurement campaign prior to the actual localization, therefore are more suitable and cost-effective. This is especially desirable for the envisioned applications in harsh or hostile environments. Second, AOA measurements not only require a direct line-of-sight (LOS) path from the transmitter to the receiver, but also have additional device cost such as multiple antennas at the receiver, which can be a limiting factor for small form factor and low power requirement of future wireless device or sensors. Third, among distance related measurements, using the RSS to determine distance requires the additional knowledge about the propagation environment, e.g., path loss exponent and shadowing, which is either not easily available or time-varying, and can lead to inaccurate distance estimates. TOA-based distance measurements, on the other hand, do not require the knowledge about the propagation environment or directly LOS path (though NLOS propagation can render range estimates biased) and generally yield better localization accuracy. Therefore, despite the additional complexity associated with either accurate clock synchronization in the one-way propagation time measurement or communication overhead involved in the two-way

roundtrip propagation time measurement [57, 58], TOA-based distance measurements and its associated distance measurement model have been widely used for collaborative position location, as seen by the great amount of existing work [13]-[40].

As far as the physical layer technology is concerned, acoustic signals and narrow-band radio frequency (RF) have been used [11]. A recent popular idea in propagation time measurements is to use ultra wideband (UWB) RF signals. A UWB signal is defined as a signal whose total bandwidth is more than 500 MHz or 20% of its center frequency. Benefiting from the extremely short (sub-nanosecond) pulse duration, a UWB signal has fine timing resolution and is naturally suited for ranging purposes, *i.e.*, obtaining distance estimates from TOA measurements. Therefore, in this work, we assume the underlying physical layer technology is a UWB signal. A survey of UWB-based ranging techniques and the details of a measurement campaign conducted particularly for position location at Virginia Tech can be found in Chapters 2 and 6 of [58], respectively.

2.3 Distance Measurement Model

We now describe a mathematical model for TOA-based distance measurements that will be used in this work. This model essentially attempts to capture the statistical properties extracted from practical measurement data in [58] and is consistent with most existing work. Inter-node range estimates[3] can be obtained from TOA estimation using different physical layer communication technologies. In realistic propagation environments, TOA estimation is corrupted by receiver noise and different channel effects, such as shadowing and multipath, which leads to erroneous distance measurements [10]. The use of UWB signals for ranging has been demonstrated as a promising technique to combat the multipath fading effect [59]. A mathematical model for TOA-based distance measurement, *i.e.*, $X_{ij} = r_{ij}$, is given by

$$\begin{aligned} r_{ij} &= d_{ij} + n_{ij}, && \text{if it is LOS} \\ r_{ij} &= d_{ij} + n_{ij} + b_{ij}, && \text{if it is NLOS} \end{aligned} \tag{2.3}$$

where

$$d_{ij} = d(\theta_i, \theta_j) = ||\theta_i - \theta_j|| = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{2.4}$$

[3]We will use the terms range estimates or measurements and distance estimates or measurements interchangeably.

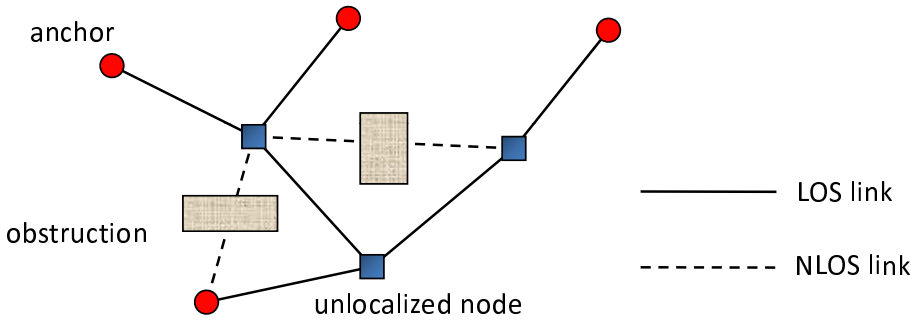


Figure 2.1: Illustration of the NLOS propagation problem. The two links shown by the dashed lines are obstructed by two objects, resulting in the NLOS propagation. In terms of TOA estimates, they will generally appear to be much larger than they should be. The corresponding range estimates are then positively biased.

is the Euclidean distance between the ith and jth nodes. The range estimation noise, n_{ij} is a zero-mean Gaussian random variable with a variance of σ_{ij}^2 given by

$$\sigma_{ij}^2 = K_E \, d_{ij}^{\beta_{ij}}, \tag{2.5}$$

where K_E is a receiver-specific proportionality constant capturing the effect of signal-to-noise ratio (SNR) and estimation technique. The last term b_{ij} represents an unknown NLOS bias, which is positive and much larger than the noise standard deviation σ_{ij}, due to the fact that a NLOS signal travels over a longer distance than the LOS signal [58]. The most common way to describe NLOS bias is to model b_{ij} as being uniformly distributed over $[B_{\min}, B_{\max}]$ where $B_{\max}$ is the maximum NLOS bias and $B_{\min}$ is the minimum NLOS bias and sometimes is simply set to be $B_{\min} = 0$. Note that exponentially distributed NLOS bias is also sometimes observed [60]. The NLOS problem can be illustrated by Fig. 2.1, where the solid and dashed lines denote LOS and NLOS communication links, respectively. As can be seen, some signal propagation paths are obstructed by physical objects and signals at the corresponding receivers either arrive at a much later time or are significantly attenuated, which generally leads to largely biased inter-node distance measurements.

2.4 Related Work: A Chronological Survey

In Chapter 1, we gave a simple example in Fig. 1.3 to illustrate the basic concept as well as a few heuristic schemes for collaborative position location, including several representative works utilizing each of the four schemes. We now give a brief survey of the large number of algorithms developed in the past decade for collaborative position location, especially in the context of 2D distance-based position location for wireless sensor networks (WSN). By arranging the survey in a chronological order, we demonstrate how the concept of collaborative position location has emerged, matured and recently raised interests in some new topics.

Beginning of the decade: Most of the earlier developments have not explicitly used the term collaborative position location and focused only on position location without the help of GPS [6]-[9]. For example, in [8], the authors proposed a simple localization method where anchors broadcast beacons into the network and each unlocalized node uses the centroid of the anchors from which it has received beacons as its location estimate. Similar work but with improved performance can be found in [9]. In the RADAR system developed in [7], a pre-deployment measurement campaign was conducted to establish a mapping from the vector of RSS to locations. When a wireless device comes into the coverage area, it reports its RSS vector to the central database, which will return a location that best matches with the RSS vector to the device as its location estimate. The DV-hop approach proposed in [16] first calculates the average distance per hop by flooding anchor beacons within the network. Then each unlocalized node uses this distance information, as well as the number of hops it is from any anchor which has been learned in the beacon-flooding stage, to estimate its location by performing trilateration. This can also be viewed as an effort to improve the work in [8]. Nonetheless, its performance still suffers from the average-hop distance estimation step. To improve this, in [17], the authors adopted a local trilateration-based refinement stage to refine the initial solution obtained by the "hop-terrain" method in the first stage. A similar refinement approach has been adopted in [18], where the proposed algorithm has a pre-processing step of constructing so-called collaborative subtrees to include only those nodes considered as uniquely localizable, although the conditions for determining unique localizability only considers connectivity information and are thus questionable in practice. Another well-formulated approach is the work in [15], where the authors proposed a centralized convex programming approach to determine node locations using the connectivity information of the entire network. In general, these earlier developments are mostly coarse-grained localization systems and focus more on demonstrating the feasibility of collaborative

position location, and therefore have not carefully considered practical issues such as range estimation noise and NLOS propagation.

Around the middle of the decade: Later on, one approach using only connectivity information is proposed in [19], where multi-dimensional scaling (MDS) technique was applied to the network connectivity matrix to construct a relative node location map. Further, in their following work [20] and some parallel work [21, 22], distributed MDS using distance estimates has been proposed to improve the localization accuracy as well as to address the algorithm scalability problem associated with centralized MDS localization. In [24], the authors described the maximum likelihood estimator (MLE) for sensor localization, based on the assumption that the TOA range estimation noise is Gaussian distributed. Despite the fact that the MLE is known to be optimal in terms of the mean-squared-error (MSE), a closed-form expression for the global optimal solution is unavailable due to the associated nonlinear and non-convex optimization problem. In [25, 26], the authors resorted to a semi-definite programming (SDP) relaxation to the original MLE of collaborative position location and demonstrated its robust performance across different network topologies. In [30], a second-order cone programming (SOCP) relaxation is adopted aiming at faster computation than the SDP approach at the cost of some localization accuracy degradation. However, both the SDP and the SOCP are no longer the original MLE and do not provide the true optimal performance of collaborative position location. In [37], the authors used a grid-based numerical solution search strategy to solve the MLE for collaborative position location based on the RSS measurements, but the solution is again not guaranteed to be optimal. In [28], the authors proposed the use of robust quadrilaterals to particularly handle flip ambiguity in node locations. However, the requirement of having many quadrilaterals in the network can be too stringent. The authors in [32] focused on a sequential localization method and developed a measurement selection procedure to improve localization accuracy and handle the problem of error propagation. One interesting research direction taken by the authors in [50]-[52] is the use of graph rigidity theory to investigate node localizability. However, one obvious drawback of the approach is that it does not consider the presence of range estimation noise or NLOS bias. Another notable and distinguishing approach is use of the nonparametric belief propagation (NBP) for collaborative position location [53] and some subsequent similar works [54]-[56]. In the NBP approach, the network localization problem is formulated as a probabilistic inference problem, where the goal is, instead of obtaining a snap-shot location estimate, to obtain an approximated probability distribution for each unlocalized node's location. Belief propagation (BP) refers to the process of nodes communicating and exchanging information with each other. The resulting approx-

imated probability distribution naturally captures the uncertainty of the location estimate and can be used to determine node localizability in the presence of range estimation noise. However, the biggest drawback with NBP-based or other probabilistic localization methods is the prohibitively high computational complexity, which can be problematic for low-cost and low-power sensor deployment. Overall, all these developments have greatly advanced the techniques of collaborative position location as well as drawn special attention to topics such as the optimal performance, algorithm efficiency and scalability, localization accuracy in NLOS conditions and node localizability in practical situations.

The past three years or so: More recently, there has been an increased amount of effort spent on the topics we just mentioned above. In particular, the authors in [36] derived the MLE in the presence of NLOS biases and developed a numerical method to solve it. However, the global optimality of the solution is still not guaranteed. In [61], we developed a branch-and-bound/reformulation linearization technique(BB/RLT) framework to solve the MLE for distance-based collaborative position location. This is the first work that guarantees the global optimality of the solution to the MLE. Based on this, we identified cases where the MSE performance indicated by the CRLB can be surpassed by the MLE due to the fact that the MLE is biased and concluded that the optimal performance of the MLE is a more meaningful performance benchmark in such cases. To address the issue of algorithm efficiency and scalability, the authors in [31] developed a distributed SOCP approach to network localization. Similar efforts were described in [62], where a localization scheme based on sum of squares (SOS) relaxation is proposed. Also, the authors in [63] proposed a curvilinear component analysis (CCA) based method to improve the MDS-MAP approach. However, these methods still have considerable computational complexity and do not scale well as the number of nodes and average node connectivity increase. Regarding NLOS conditions, in [64], we developed a collaborative quasi-linear programming framework to consider both node collaboration and NLOS mitigation. However, the performance is still not satisfactory, especially when sequential localization has to be used. In terms of node localizability, the works in [65] and [66] have the SDP and graph rigidity approaches, respectively. However, both have not considered the issue of range estimation noise and are thus not applicable in practical situations. To summarize, these more recent works have spent much effort to address several important practical issues we described above, but significant work still remains ahead.

2.5 A Taxonomy of Existing Algorithms

To better our understanding of existing localization algorithms, we now provide several different ways to classify them and briefly describe the pros and the cons associated with each class of methods. Within each class, we only strive to describe the key ideas of the most popular methods, which shall be only considered as being representative rather than being comprehensive.

Type of measurement data: Distance, AOA and RSS fingerprinting

Depending on the measurement data being used, localization algorithms can be classified as distance-based (via RSS or TOA), connectivity-based, AOA-based, RSS fingerprinting and hybrid methods. For example, algorithms proposed in [8, 15, 16, 19, 20, 67] are all connectivity-based localization, while algorithms such as those in [13, 14, 17, 18, 25, 32] are all distance-based localization. In general, connectivity-based localization has lower device and computational complexity, but only generates coarse-grained location information. Distance-based localization, on the other hand, leads to more accurate location estimation, but requires more sophisticated devices and has heavier computational load. Some algorithms, e.g., [33], use connectivity-based location estimation as a starting point, and then apply distance-based location refinement. Considering the potential benefit offered by distance measurements and also to be more consistent with most existing work, we will focus on distance-based localization to facilitate the discussion and performance comparison.

Where the computation is performed: Centralized or Distributed

Another way to classify localization algorithms is based on where the computation is performed. In the so-called centralized approaches [9, 15, 19, 20], [24]-[27] and [30, 33, 36, 61], all the anchor locations and measurement data are forwarded to a central processor to compute the unlocalized nodes' positions in a joint manner. In distributed approaches [14], [16]-[18], [21, 22, 28, 31, 32] and [34, 64], the computation is spread over the entire network and thus only local information exchange is required. Distributed approaches have the advantage of being scalable and more robust to node failures. Centralized approaches, on the other hand, utilize the information about the entire network and are supposed to yield more accurate location estimation. However, an efficient solver for large scale nonlinear optimization problems associated with centralized approaches is needed to truly achieve the global optimum. Considering all the tradeoff and complexity issues, distributed approaches have attracted much more attention and are more practical for the applications of interest.

How the computation is performed: Sequential or Concurrent

Depending on how the location information is *propagated* through the network, localization algorithms can also be classified as sequential localization or concurrent localization. In sequential approaches [14, 32, 64], each unlocalized node updates itself as a *virtual* anchor to assist other unlocalized nodes to localize themselves. In the sequential approaches, only measurements with neighboring nodes are used. An obvious drawback of the sequential approaches is the propagation of localization error [68, 69], which refers to the fact that virtual anchors themselves have localization error and will affect any unlocalized node that uses range estimates to them to perform location estimation. On the other hand, in the concurrent approaches, unlocalized nodes do not act as virtual anchors even after they have been localized. Examples include all the centralized approaches and the distributed algorithms proposed in [18, 28, 22]. Generally speaking, although sequential location estimation appears to be a simple way of disseminating location information throughout the network, the problem of localization error propagation has seriously limited its performance.

How the problem is formulated: Probabilistic or Non-probabilistic

Another way to classify existing algorithms is regarding whether the position location problem is formulated in a *probabilistic* manner. Specifically, depending on the nature of the framework as well as the returned location estimates, existing algorithms can be divided into *deterministic* and *probabilistic* approaches. Deterministic approaches solve position location problem in different ways, including pure optimization-based algorithms [15, 25, 30, 33, 48], estimation-based methods [24, 70], and other ad hoc algorithms, e.g., in [8]-[14], [16]-[18], [22, 28, 29]. A common feature of these algorithms is that they only compute a deterministic *one-shot* location estimate for nodes that have been localized. In other words, there is no additional information about the quality or the reliability of this solution. As we mentioned earlier, metrics such as distance residual, GDOP and error ellipses, which do provide some sense of reliability, may not be appropriate for collaborative position location, especially in the presence of NLOS propagation. On the other hand, in probabilistic localization approaches [53]-[56], the position location problem is formulated as a probabilistic inference problem, with the goal of computing (approximating) the probability distribution of each unlocalized node's location. Under such a framework, a position location algorithm returns a set of possible location estimations, each of which is preferably, although not necessarily, associated with a weight quantifying the uncertainty, or equivalently the reliability, about the corresponding location estimate. In this sense, probabilistic approaches return a more direct representation of the solution quality. However, existing algorithms under the prob-

abilistic localization framework have much higher computational complexity than one-shot estimate based algorithms, which can be a limiting factor for low-power sensor deployment.

To summarize, considering the state-of-the-art in the research field as well as different trade-offs between the device cost, computational complexity and localization accuracy, we believe a distributed and concurrent, non-probabilistic collaborative position location approach using distance measurements with a proper NLOS mitigation method is the path to take as far as the design of a practical algorithm is concerned.

Chapter 3

The CRLB for Collaborative Position Location

3.1 Introduction

As we described earlier, collaborative position location utilizes not only the measurements between unlocalized nodes and anchors, but also those among unlocalized nodes, which has been shown to have the potential to improve the overall localization accuracy [24]. Despite its usefulness and many existing algorithms, how to effectively utilize the measurements among unlocalized nodes remains a challenging issue. On the other hand, the fundamental limit via the Cramer-Rao lower bound (CRLB) on the variance of an unbiased estimator serves as a performance benchmark and has been used to evaluate collaborative position location schemes [36, 71]. In [24] and [72], the CRLB for localization error for any unbiased location estimator using range estimates was derived, based on the assumption that the range estimates are corrupted by zero-mean Gaussian noise. Additionally, in [24], the authors have shown that as long as certain connectivity conditions are met for a newly-added unlocalized node, the overall CRLB on localization error for those unlocalized nodes already in the network will always become lower. In [73], the authors considered the CRLB in an NLOS environment and showed that without the knowledge of the NLOS error, the minimum variance unbiased estimator (MVUE) simply discards all NLOS range estimates and uses only LOS range estimates. In all the existing work, one assumption that has been made is that the variance of the range estimation noise is not related to the true inter-node distance. However, as demonstrated in [58], range estimation noise variance does depend on the true

inter-node distance, meaning that nodes that are further away from each other obtain range estimates at a higher noise level, and we believe the knowledge of this may potentially affect the design of a collaborative position location scheme.

In this chapter, we derive a new CRLB based on the knowledge of a distance dependent variance model for range estimation noise. As demonstrated, this distance-dependent noise model influences the derivation of the Fisher Information Matrix (FIM), and finally leads to a CRLB different from existing work, e.g., those in [24] and [72]. We also show that our new CRLB is always lower than the previous CRLB, meaning that having the knowledge of range estimation noise model helps improve the localization accuracy. Furthermore, through the distance-dependent noise model, the new CRLB provides a means of understanding the immediate impact of node geometric configuration on the localization accuracy as well as node collaboration.

3.2 The CRLB for Collaborative Position Location

The CRLB is a lower bound on the variance of any *unbiased* estimator. Without any knowledge about NLOS range measurements, an unbiased location estimator will discard any NLOS range measurement to maintain the unbiasedness of the estimator [73]. Therefore, as in existing work, we assume all range measurements are LOS. In terms of our range measurement model defined in (2.3), this means[1]

$$r_{ij} = d_{ij} + n_{ij}, \tag{3.1}$$

Based on these assumptions, if there exists a range measurement r_{ij} between the ith and jth nodes, its probability density function (pdf), conditioned on the locations, θ_i and θ_j, can be written as

$$f(r_{ij}|\theta_i, \theta_j) \;=\; \frac{1}{\sqrt{2\pi}\sigma_{ij}} \exp\left(-\frac{(r_{ij} - d_{ij})^2}{2\sigma_{ij}^2}\right) \tag{3.2}$$

where d_{ij} and σ_{ij}^2 are the true distance and the variance of measurement noise, given by (2.4) and (2.5), respectively. Substituting these into (3.2), the log-likelihood function of (3.2) is

[1]To be consistent with existing literature, we focus on the Gaussian distributed noise in this work. However, in Appendix 3.B and 3.C, we include the derivation of the CRLB based on exponentially distributed noise which only has positive support, for both distance-independent and distance-dependent noise modeling.

given by

$$\log f(r_{ij}|\theta_i,\theta_j) = -\log\sqrt{2\pi K_E} - \frac{\beta_{ij}}{4}\log\left[(x_i-x_j)^2+(y_i-y_j)^2\right] - \frac{1}{2K_E}\times\frac{\left[r_{ij}-\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}\right]^2}{\left[(x_i-x_j)^2+(y_i-y_j)^2\right]^{\beta_{ij}/2}} \tag{3.3}$$

Assuming range measurements are corrupted by independent noise[2], the joint pdf of the set of all range measurements is given by

$$f(\mathcal{X}|\Theta) = \prod_{i=1}^{N}\prod_{\substack{j>i\\ j\in\mathcal{N}(i)}}^{N+M} f(r_{ij}|\theta_i,\theta_j) \tag{3.4}$$

where $j > i$ in the second summation is to ensure each range measurement is only counted once, while $j \in \mathcal{N}(i)$, as mentioned in Chapter 2, indicates that the jth node is a neighbor of the ith node. Therefore, the log-likelihood of the joint pdf of all range measurements is thus given by

$$\begin{aligned}\log f(\mathcal{X}|\Theta) &= \sum_{i=1}^{N}\sum_{\substack{j=i+1\\ j\in\mathcal{N}(i)}}^{N+M}\log f(r_{ij}|\theta_i,\theta_j)\\ &= \sum_{i=1}^{N}\sum_{\substack{j=i+1\\ j\in\mathcal{N}(i)}}^{N+M}\left\{-\log\sqrt{2\pi K_E} - \frac{\beta_{ij}}{4}\log\left[(x_i-x_j)^2+(y_i-y_j)^2\right]\right.\\ &\quad\left. - \frac{1}{2K_E}\times\frac{\left[r_{ij}-\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}\right]^2}{\left[(x_i-x_j)^2+(y_i-y_j)^2\right]^{\beta_{ij}/2}}\right\}\end{aligned} \tag{3.5}$$

The Fisher Information Matrix (FIM) J_Θ is defined by [74]

$$\begin{aligned}J_\Theta &= -\mathbf{E}_\Theta\left\{\nabla_\Theta[\nabla_\Theta\log f(\mathcal{X}|\Theta)]^T\right\}\\ &= \begin{pmatrix} J_{1,1} & J_{1,2} & \dots & J_{1,2N-1} & J_{1,2N}\\ J_{2,1} & J_{2,2} & \dots & J_{2,2N-1} & J_{2,2N}\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ J_{2N-1,1} & J_{2N-1,2} & \dots & J_{2N-1,2N-1} & J_{2N-1,2N}\\ J_{2N,1} & J_{2N,2} & \dots & J_{2N,2N-1} & J_{2N,2N}\end{pmatrix}.\end{aligned} \tag{3.6}$$

[2]In this work, we limit our attention to independent range estimation noise. The effect of correlated noise is beyond the focus of this work.

Note that the FIM has a dimension of $2N \times 2N$, since there are totally $2N$ location parameters to be estimated.

Using (3.3) and (3.5), for $i = 1, 2, \ldots, N$, the main diagonal elements and the first diagonal below/above the main diagonal of the FIM are derived as (see Appendix 3.A for the detailed derivation.)

$$J_{2i-1,2i-1} = \sum_{j \in \mathcal{N}(i)} \frac{w_{ij} \cos^2 \alpha_{ij}}{\sigma_{ij}^2}, \tag{3.7}$$

$$J_{2i,2i} = \sum_{j \in \mathcal{N}(i)} \frac{w_{ij} \sin^2 \alpha_{ij}}{\sigma_{ij}^2}, \tag{3.8}$$

$$J_{2i-1,2i} = J_{2i,2i-1} = \sum_{j \in \mathcal{N}(i)} \frac{w_{ij} \cos \alpha_{ij} \sin \alpha_{ij}}{\sigma_{ij}^2}, \tag{3.9}$$

where

$$w_{ij} = 1 + \frac{\beta_{ij}^2 K_E}{2} d_{ij}^{\beta_{ij}-2} \tag{3.10}$$

is a distance-dependent unitless scaling factor always greater than 1. Note that due to the definition of (2.5), K_E is unitless only when $\beta_{ij} = 2$.

On the other hand, for $i, j = 1, 2, \ldots, N$, $j \neq i$ and $j \in \mathcal{N}(i)$, the non-diagonal elements, except the first diagonal below/above the main diagonal, of the FIM are given by

$$J_{2i-1,2j-1} = J_{2j-1,2i-1} = -\frac{w_{ij} \cos^2 \alpha_{ij}}{\sigma_{ij}^2}, \tag{3.11}$$

$$J_{2i,2j} = J_{2j,2i} = -\frac{w_{ij} \sin^2 \alpha_{ij}}{\sigma_{ij}^2}, \tag{3.12}$$

$$J_{2i-1,2j} = J_{2j,2i-1} = J_{2i,2j-1} = J_{2j-1,2i} = -\frac{w_{ij} \cos \alpha_{ij} \sin \alpha_{ij}}{\sigma_{ij}^2}, \tag{3.13}$$

If $j \notin \mathcal{N}(i)$, the above three non-diagonal elements simply are zeros.

Finally, the CRLB for the localization error of the ith node can be described by

$$\mathbf{E}[(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2] \geq J_{2i-1,2i-1}^{-1} + J_{2i,2i}^{-1}. \tag{3.14}$$

where $[\hat{x}_i \ \hat{y}_i]^T$ is an unbiased estimate of the true coordinate $[x_i \ y_i]^T$ and $J_{i,j}^{-1}$ denotes the $(i,j)^{\text{th}}$ element of the inverse of the FIM J_Θ. A more useful metric is given by

$$\sqrt{\mathbf{E}[(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2]} \geq \sqrt{J_{2i-1,2i-1}^{-1} + J_{2i,2i}^{-1}}, \tag{3.15}$$

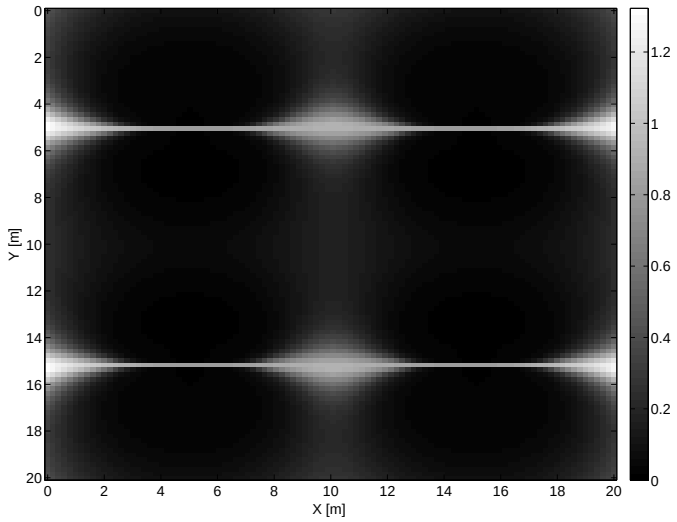
We emphasize that our derived FIM differs from the result in [72], by having the additional distance-dependent scaling factor w_{ij}, given in (3.10), which captures the impact of having the knowledge of range estimation noise variance model on the localization accuracy.

3.3 Numerical Results

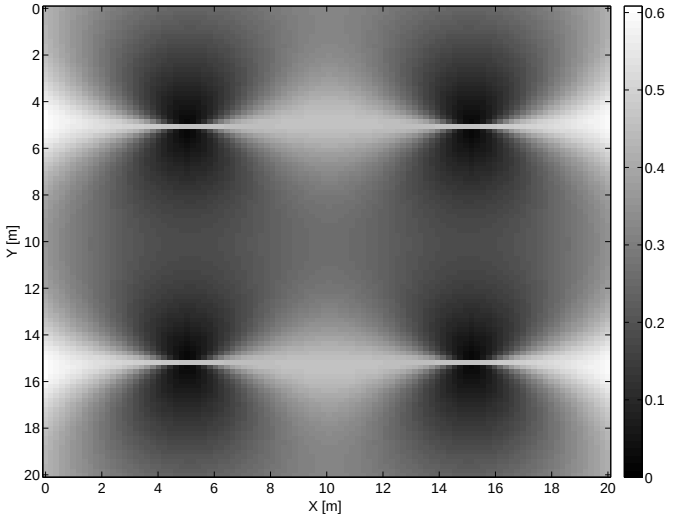
In this section, we present some numerical results for the CRLB we derived. By comparing it with the previous results [72], which we refer to as the old CRLB, we demonstrate the benefit of our new CRLB in providing direct insight into the impact of node geometry on the localization accuracy. Finally, we evaluate the performance of the popular linear least-squares (LS) location estimator against both the new and old CRLBs.

First, we examine the old and the new CRLB for a simple single-node localization scenario. We assume the network is a $20 \times 20\,\mathrm{m}^2$ square area and four anchors are placed at (5, 5), (15, 15), (5, 15) and (15, 5), respectively. A single unlocalized node is placed within the network area and is assumed to be able to communicate with all four anchors, *i.e.*, the effect of limited communication range is not considered. Fig. 3.1 (a) and (b) present the difference between the old and the new CRLB, *i.e.*, (old CRLB - new CRLB), for each unknown location with the network, when $K_E = 0.001$ and $\beta_{ij} = \beta = 4$, for any β_{ij}, and $K_E = 0.16$ and $\beta = 2$, respectively. Note that *darker* regions correspond to those having *smaller* values. As we can observe, the new CRLB is always smaller than the old one, since the difference is always larger than zero. This is due mainly to the fact that we have implicitly used additional knowledge on the range measurements, embedded in the distance-dependent noise model, and this translates to an increased FIM, which in turn lowers the CRLB. This knowledge is straightforward to understand and can be available in practical situations.

Next, we examine the CRLB for a network of unlocalized nodes with respect to different values of the path loss exponent β and proportionality constant K_E. Both network size and anchor locations are the same as in Fig. 3.1. The communication range is now limited to be $R = 8\,\mathrm{m}$, *i.e.*, nodes that are separated by more than R cannot obtain range measurement to each other. We randomly generated 50 unlocalized nodes within the network. In Fig. 3.2, we present the new and the old CRLBs for 25 out of 50 nodes, with respect to different path loss exponents, for $K_E = 0.005$ and 0.01, respectively. A larger value of K_E indicates the presence of more range measurement noise. We observe that the CRLB increases as the path loss exponent becomes larger, which is a straightforward result because of the increased noise

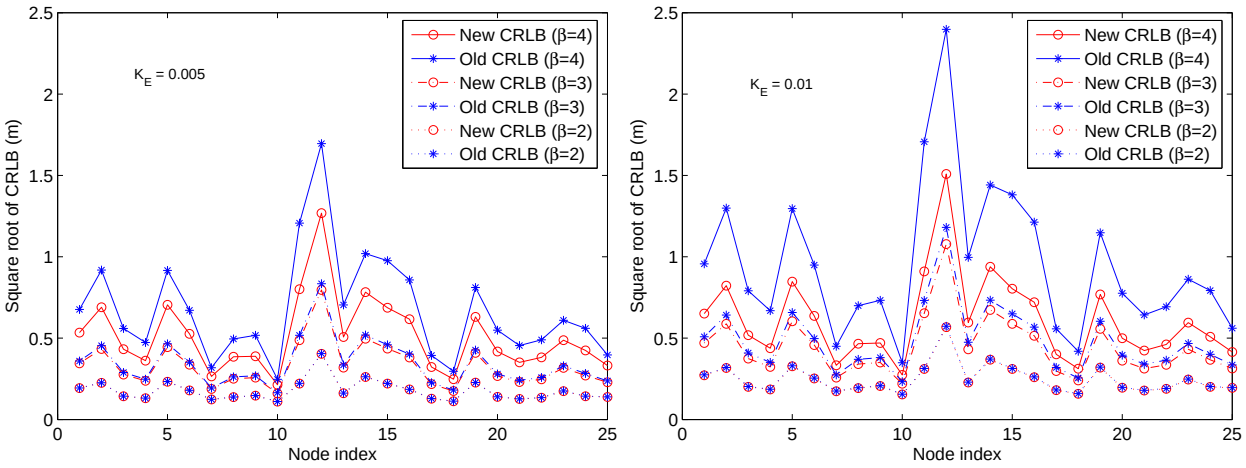


(a) (Old CRLB - new CRLB) for $K_E = 0.001$ and $\beta = 4$



(b) (Old CRLB - new CRLB) for $K_E = 0.16$ and $\beta = 2$

Figure 3.1: Difference between the old CRLB with our new CRLB. The noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$. Note that *darker* regions correspond to those having *smaller* values. It can be seen that, with the knowledge of the variance model of range estimation noise, the new CRLB is lower than the old CRLB.



(a) CRLB when $R = 8\,\text{m}$ and $K_E = 0.005$

(b) CRLB when $R = 8\,\text{m}$ and $K_E = 0.01$

Figure 3.2: The CRLB of different nodes within the network, showing the effect of different values of β and K_E on the CRLBs (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$). Note the different scales on the two figures. The larger β and K_E, the more benefit we can get from knowing the variance model of range estimation noise.

variance. In addition, it is seen that the new and the old CRLBs are almost identical when $\beta = 2$, while the difference becomes larger as β increases from 2 to 4. Furthermore, the new CRLB is always smaller than the old one and the difference becomes larger as K_E increases from 0.005 to 0.01. This can be seen from the fact that w_{ij} in (3.10) is always greater than one, which ultimately leads to a smaller CRLB. The larger β and/or K_E becomes, the larger w_{ij} is, and the larger the difference between the new and the old CRLBs.

Finally, we present a comparison between the CRLB with the popular linear LS estimator [70] in Fig. 3.3, when localizing an unlocalized node using different numbers of anchors. The RMS localization error for the LS estimator is computed according to (2.2) by averaging over 1000 noise realizations for a randomly generated set of anchor locations. We assume the unlocalized node is able to communicate with all anchors. Additional anchors are incorporated in ascending order of their corresponding estimated range values, *i.e.*, larger range estimates are incorporated later than smaller ones. This intuitively makes sense since anchors further away usually result in larger measurement noise as indicated by (2.5). As we can see, the CRLB serves as a benchmark to evaluate how well a practical LS estimator performs. The more anchors being used, the better the localization accuracy becomes. Furthermore, there

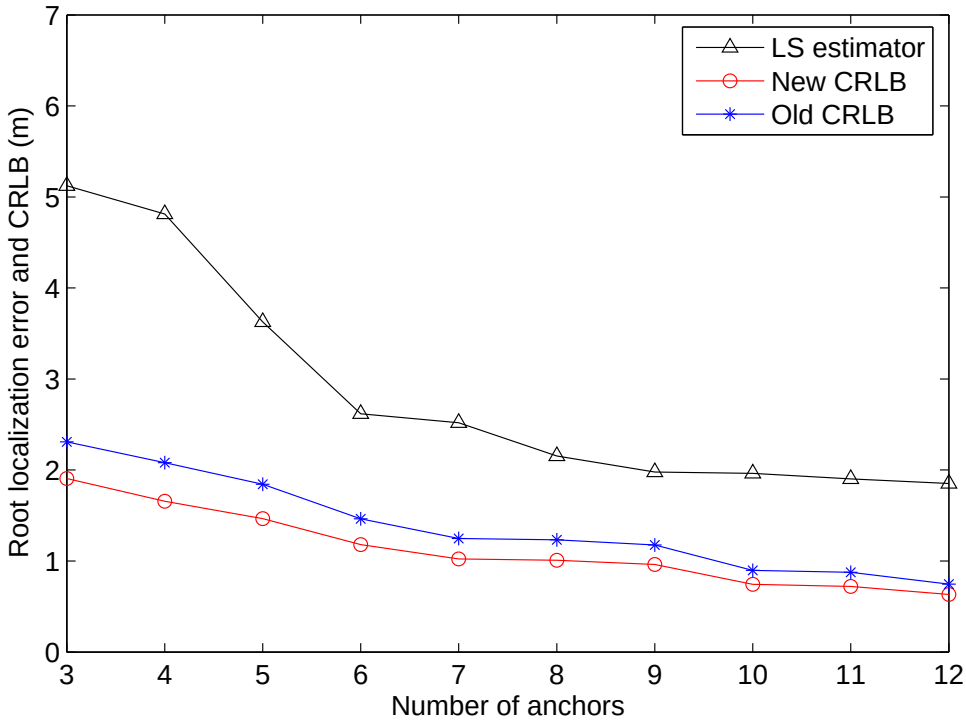


Figure 3.3: Comparing the LS estimator's performance with the CRLBs, for $K_E = 0.001$ and $\beta = 4$ (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$). The fact that the new CRLB is lower than the old one suggests that having the knowledge of distance-dependence model of the noise variance has the potential of improving the localization accuracy. In addition, the LS estimator, despite having a closed-form solution, still performs much worse than the CRLB suggests.

is a relatively large performance gap between the popular LS estimator and that indicated by the CRLB. In fact, it is easy to show analytically that the LS estimator is not even an unbiased estimator. Our new CRLB indicates the possibility that the localization accuracy can be further improved, as compared to the old CRLB. In other words, it demonstrates the value of the additional knowledge about range measurements' dependence on the true distance in position location. However, the search for a good estimator remains a challenging issue.

3.4 Discussions

3.4.1 Distance-dependence of Measurement Noise

As mentioned earlier, existing work on the CRLB for position location has ignored this distance-dependence of the noise variance. Specifically, in [24], the authors simply ignored this effect and assigned equal variance to all range measurements. In [72], although the derived CRLB considered different noise variances for different range measurements, its dependence on the inter-node distance was not considered in the derivation of the FIM. This is equivalent to the fact that when taking derivatives of (3.3) with respect to (θ_i, θ_j), only the mean d_{ij} is considered to be dependent on the θ_i, θ_j. However, as experimental results suggest, the variance σ_{ij}^2 is dependent on the θ_i, θ_j. Although the relationship between the noise variance and the true distance may not be exactly captured by (2.5), we claim that it should be taken into consideration since it represents some easily-available *additional knowledge* about the measurement.

3.4.2 Path Loss Exponent

We need to emphasize that a path loss exponent of $\beta = 3$ or $\beta = 4$ practically represents the case of NLOS propagation. This is opposed to the fact that the CRLB is for an unbiased estimator and our assumption that all range measurements are LOS. However, we argue that our results are still meaningful in the sense that it provides a means of characterizing the best possible performance of an unbiased location estimator, if it is equipped with *perfect* knowledge of the NLOS bias in each NLOS range measurement. To be more general, we believe our new CRLB tells us that if all range measurements are LOS, the additional knowledge of the noise dependence on true distance does not help too much, as suggested by the CRLB in Fig. 3.2 for $\beta = 2$. On the other hand, if NLOS range measurements are prevalent, the new CRLB tells us that, with perfect knowledge about NLOS bias, it is desirable to use range measurements from nearer anchors.

3.4.3 GDOP's Suitability for Indoor Localization

What is more important, our approach of introducing distance-dependence into the noise variance no longer allows the extraction of geometric dilution of precision (GDOP), since

GDOP is a pure geometric quantity derived by assuming all range estimates have the *same* noise variance [24]. This is a reasonable assumption for GPS localization, since the distances from a GPS receiver to its visible satellites are not much different from each other. However, this assumption does not hold for indoor localization, where the difference between the distances from an unlocalized node to anchors may cause a non-negligible difference in noise variance, especially when the path loss exponent is large, e.g., 3 or 4, which is exactly the case for many indoor environments. Therefore, we argue that it may not be proper to attempt to de-couple geometric information from the noise variance for indoor localization and GDOP is no longer a good metric to gain insight into the localization system performance. In this regard, the existing derivation of the CRLB is not accurate, due to the fact that it ignores the distance-dependence in deriving the FIM. Our approach, on the other hand, is more accurate and offers direct insight into the impact of geometric configuration on the localization system performance.

3.5 Summary

In this chapter, we derived a new CRLB based on a distance-dependent noise variance model. This model impacts the FIM and ultimately leads to a different CRLB. We argue that the traditional way of computing GDOP, by assuming equal variance on all range estimates, may not be an appropriate way to characterize the impact of node geometric configuration for indoor localization, where the variance of range measurement can change dramatically for different wireless connections. Finally, the fact that the new CRLB is always smaller than the old one indicates the benefit of additional knowledge on measurement as well as a possibility for further improving localization accuracy.

Appendix 3.A: Derivation of the FIM for Gaussian distributed range estimation noise

Using (3.3), we define the following two terms

$$A = -\frac{\beta_{ij}}{4}\log\left[(x_i-x_j)^2+(y_i-y_j)^2\right] \tag{3.16}$$

$$B = -\frac{\left[r_{ij}-\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}\right]^2}{2K_E\left[(x_i-x_j)^2+(y_i-y_j)^2\right]^{\beta_{ij}/2}}. \tag{3.17}$$

Therefore, let $L(\theta_i,\theta_j) \triangleq \log f(r_{ij}|\theta_i,\theta_j)$, we have

$$\frac{\partial L(\theta_i,\theta_j)}{\partial x_i} = \frac{\partial A}{\partial x_i} + \frac{\partial B}{\partial x_i} \tag{3.18}$$

It is easy to derive that

$$\frac{\partial A}{\partial x_i} = -\frac{\beta_{ij}}{2}\times\frac{x_i-x_j}{(x_i-x_j)^2+(y_i-y_j)^2} \tag{3.19}$$

$$\begin{aligned}\frac{\partial B}{\partial x_i} &= \frac{1}{K_E}\times\frac{x_i-x_j}{[(x_i-x_j)^2+(y_i-y_j)^2]^{1+\beta_{ij}/2}} \\ &\times \left[r_{ij}-\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}\right] \\ &\times \left[\frac{\beta_{ij}}{2}r_{ij}+(1-\frac{\beta_{ij}}{2})\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}\right]\end{aligned} \tag{3.20}$$

The derivation of the second-order partial derivatives are more involved. After a few steps, we can show that

$$\frac{\partial^2 A}{\partial x_i^2} = \frac{\beta_{ij}}{2}\times\frac{(x_i-x_j)^2-(y_i-y_j)^2}{d_{ij}^4} \tag{3.21}$$

$$\begin{aligned}\frac{\partial^2 B}{\partial x_i^2} &= \frac{1}{K_E\, d_{ij}^{4+\beta_{ij}}}\times\Big[(1-\beta_{ij})r_{ij}(x_i-x_j)^2 d_{ij}+\frac{\beta_{ij}r_{ij}^2}{2}d_{ij}^2 \\ &- (2-\beta_{ij})(x_i-x_j)^2 d_{ij}^2+(1-\beta_{ij})r_{ij}d_{ij}^3-(1-\frac{\beta_{ij}}{2})d_{ij}^4 \\ &- \frac{\beta_{ij}r_{ij}}{2}(2+\beta_{ij})(x_i-x_j)^2(r_{ij}-d_{ij}) \\ &- 2(1-\frac{\beta_{ij}^2}{4})(x_i-x_j)^2(r_{ij}-d_{ij})d_{ij}\Big]\end{aligned} \tag{3.22}$$

Now, we know that

$$\mathbf{E}[r_{ij}] = d_{ij}, \tag{3.23}$$

$$\mathbf{E}[r_{ij}^2] = d_{ij}^2 + K_E d_{ij}^{\beta_{ij}}, \tag{3.24}$$

plug these into (3.21) to (3.22), and after a few mathematical manipulations, we can easily obtain

$$-\mathbf{E}\left[\frac{\partial^2 L(\theta_i,\theta_j)}{\partial x_i^2}\right] = -\mathbf{E}\left[\frac{\partial^2 A}{\partial x_i^2}\right] - \mathbf{E}\left[\frac{\partial^2 B}{\partial x_i^2}\right] = \frac{\cos^2\alpha_{ij}}{\sigma_{ij}^2}\left[1+\frac{\beta_{ij}^2 K_E}{2} d_{ij}^{\beta_{ij}-2}\right]. \tag{3.25}$$

Similarly, we can obtain

$$-\mathbf{E}\left[\frac{\partial^2 L(\theta_i,\theta_j)}{\partial y_i^2}\right] = -\mathbf{E}\left[\frac{\partial^2 A}{\partial y_i^2}\right] - \mathbf{E}\left[\frac{\partial^2 B}{\partial y_i^2}\right] = \frac{\sin^2\alpha_{ij}}{\sigma_{ij}^2}\left[1+\frac{\beta_{ij}^2 K_E}{2} d_{ij}^{\beta_{ij}-2}\right]. \tag{3.26}$$

The cross terms are evaluated as

$$-\mathbf{E}\left[\frac{\partial^2 L(\theta_i,\theta_j)}{\partial x_i \partial y_i}\right] = -\mathbf{E}\left[\frac{\partial^2 L(\theta_i,\theta_j)}{\partial y_i \partial x_i}\right] = \frac{\cos\alpha_{ij}\sin\alpha_{ij}}{\sigma_{ij}^2}\left[1+\frac{\beta_{ij}^2 K_E}{2} d_{ij}^{\beta_{ij}-2}\right]. \tag{3.27}$$

Further, we can use the symmetric property between x_i and x_j, given by

$$\frac{\partial^2 A}{\partial x_i \partial x_j} = -\frac{\partial^2 A}{\partial x_i^2}, \tag{3.28}$$

$$\frac{\partial^2 B}{\partial x_i \partial x_j} = -\frac{\partial^2 B}{\partial x_i^2}, \tag{3.29}$$

to derive the terms as follows.

$$-\mathbf{E}\left[\frac{\partial^2 L(\theta_i,\theta_j)}{\partial x_i \partial x_j}\right] = \mathbf{E}\left[\frac{\partial^2 L(\theta_i,\theta_j)}{\partial x_i^2}\right] = -\frac{\cos^2\alpha_{ij}}{\sigma_{ij}^2}\left[1+\frac{\beta_{ij}^2 K_E}{2} d_{ij}^{\beta_{ij}-2}\right], \tag{3.30}$$

$$-\mathbf{E}\left[\frac{\partial^2 L(\theta_i,\theta_j)}{\partial y_i \partial y_j}\right] = \mathbf{E}\left[\frac{\partial^2 L(\theta_i,\theta_j)}{\partial y_i^2}\right] = -\frac{\sin^2\alpha_{ij}}{\sigma_{ij}^2}\left[1+\frac{\beta_{ij}^2 K_E}{2} d_{ij}^{\beta_{ij}-2}\right], \tag{3.31}$$

$$\begin{aligned} -\mathbf{E}\left[\frac{\partial^2 L(\theta_i,\theta_j)}{\partial x_i \partial y_j}\right] &= -\mathbf{E}\left[\frac{\partial^2 L(\theta_i,\theta_j)}{\partial y_i \partial x_j}\right] = -\mathbf{E}\left[\frac{\partial^2 L(\theta_i,\theta_j)}{\partial y_j \partial x_i}\right] = -\mathbf{E}\left[\frac{\partial^2 L(\theta_i,\theta_j)}{\partial x_j \partial y_i}\right] \\ &= \mathbf{E}\left[\frac{\partial^2 L(\theta_i,\theta_j)}{\partial x_i \partial y_i}\right] = -\frac{\cos\alpha_{ij}\sin\alpha_{ij}}{\sigma_{ij}^2}\left[1+\frac{\beta_{ij}^2 K_E}{2} d_{ij}^{\beta_{ij}-2}\right]. \end{aligned} \tag{3.32}$$

Finally, the FIM can easily obtained by using the above results, where (3.25) - (3.27) are used to evaluate the diagonal terms in (3.6), while (3.30) - (3.32) are used to evaluate the non-diagonal terms in (3.6).

Appendix 3.B: Derivation of the FIM for exponentially distributed range estimation noise with distance independent noise modeling

We now derive the CRLB for collaborative position location if the range estimation noise is exponentially distributed, denoted by,

$$n_{ij} \sim \mathrm{E}xp(\lambda_{ij}) = \frac{1}{\lambda_{ij}} \exp\left(-\frac{t}{\lambda_{ij}}\right), \tag{3.33}$$

where λ_{ij} is the mean of the distribution and the variance is given by λ_{ij}^2. Considering the LOS range estimate model in (2.3), we know

$$\mathbf{E}\left[r_{ij}\right] = d_{ij} + \lambda_{ij}. \tag{3.34}$$

Following the similar procedure as in Appendix 3.A and denoting the log-likelihood function of r_{ij} as $L_{\exp}(\theta_i, \theta_j)$, given by

$$\log\{L_{\exp}(\theta_i, \theta_j)\} = -\log\lambda_{ij} - \frac{r_{ij} - d_{ij}}{\lambda_{ij}} \tag{3.35}$$

we can derive the following for the diagonal terms in the FIM,

$$-\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i, \theta_j)}{\partial x_i^2}\right] = -\frac{\sin^2\alpha_{ij}}{\lambda_{ij} d_{ij}}, \tag{3.36}$$

$$-\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i, \theta_j)}{\partial y_i^2}\right] = -\frac{\cos^2\alpha_{ij}}{\lambda_{ij} d_{ij}}, \tag{3.37}$$

$$-\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i, \theta_j)}{\partial x_i \partial y_i}\right] = -\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i, \theta_j)}{\partial y_i \partial x_i}\right] = \frac{\cos\alpha_{ij}\sin\alpha_{ij}}{\lambda_{ij} d_{ij}}. \tag{3.38}$$

On the other hand, using the symmetric property mentioned earlier, the terms for non-diagonal terms in the FIM are evaluated as follows,

$$-\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i, \theta_j)}{\partial x_i \partial x_j}\right] = \mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i, \theta_j)}{\partial x_i^2}\right] = \frac{\sin^2\alpha_{ij}}{\lambda_{ij} d_{ij}}, \tag{3.39}$$

$$-\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i, \theta_j)}{\partial y_i \partial y_j}\right] = \mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i, \theta_j)}{\partial y_i^2}\right] = \frac{\cos^2\alpha_{ij}}{\lambda_{ij} d_{ij}}, \tag{3.40}$$

$$\begin{aligned} -\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i, \theta_j)}{\partial x_i \partial y_j}\right] &= -\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i, \theta_j)}{\partial y_i \partial x_j}\right] = -\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i, \theta_j)}{\partial y_j \partial x_i}\right] = -\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i, \theta_j)}{\partial x_j \partial y_i}\right] \\ &= \mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i, \theta_j)}{\partial x_i \partial y_i}\right] = -\frac{\cos\alpha_{ij}\sin\alpha_{ij}}{\lambda_{ij} d_{ij}}. \end{aligned} \tag{3.41}$$

Finally, the FIM of the exponential distributed range estimation noise can easily obtained by using the above results, where (3.36) - (3.38) are used to evaluate the diagonal terms in the FIM, while (3.39) - (3.41) are used to evaluate the non-diagonal terms in the FIM.

Appendix 3.C: Derivation of the FIM for exponentially distributed range estimation noise with distance dependent noise modeling

Now, we consider the case where range estimation noise is dependent on the inter-node distance and this knowledge is available. In particular, we have the same model as Chapter 3 for the variance of range estimation noise, given by,

$$\lambda_{ij}^2 = K_E d_{ij}^{\beta_{ij}}. \tag{3.42}$$

The difference from Gaussian range estimation noise is that the mean of the range estimation noise is also dependent on the inter-node distance, $\lambda_{ij} = \sqrt{K_E} d_{ij}^{\beta_{ij}/2}$.

Using the log-likelihood function in (3.35), we define $\log L_{\exp}(\theta_i, \theta_j) \stackrel{\triangle}{=} A_{\exp} + B_{\exp}$, where

$$A_{\exp} = -\log K_E - \beta_{ij} \log d_{ij}, \tag{3.43}$$

$$B_{\exp} = -\frac{r_{ij} - d_{ij}}{\sqrt{K_E} d_{ij}^{\beta_{ij}/2}}. \tag{3.44}$$

Then, we can obtain

$$\frac{\partial A_{\exp}}{\partial x_i} = -\frac{\beta_{ij}(x_i - x_j)}{d_{ij}^2}, \tag{3.45}$$

$$\frac{\partial B_{\exp}}{\partial x_i} = \frac{(x_i - x_j)[2d_{ij} + \beta_{ij}(r_{ij} - d_{ij})]}{2\sqrt{K_E} d_{ij}^{\frac{\beta_{ij}}{2}+2}}, \tag{3.46}$$

and the second-order derivatives as follows,

$$\frac{\partial A_{\exp}^2}{\partial x_i^2} = -\frac{\beta_{ij}[(y_i - y_j)^2 - (x_i - x_j)^2}{d_{ij}^4}], \tag{3.47}$$

$$\frac{\partial B_{\exp}^2}{\partial x_i^2} = \frac{1}{2\sqrt{K_E} d_{ij}^{\beta_{ij}+4}} \Bigg\{ \Big[2d_{ij} + \beta_{ij}(r_{ij} - d_{ij}) + \frac{(x_i - x_j)^2(2 - \beta_{ij})}{d_{ij}}\Big] d_{ij}^{\frac{\beta_{ij}}{2}+2} - (\frac{\beta_{ij}}{2} + 2) d_{ij}^{\frac{\beta_{ij}}{2}+1} \frac{(x_i - x_j)^2}{d_{ij}} [2d_{ij} + \beta_{ij}(r_{ij} - d_{ij})] \Bigg\}. \tag{3.48}$$

Now use (3.34), we have $\mathbf{E}[r_{ij}] = d_{ij} + \sqrt{K_E} d_{ij}^{\frac{\beta_{ij}}{2}}$. Substitute this into (3.47) and (3.48), we will have

$$-\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i,\theta_j)}{\partial x_i^2}\right] = -\mathbf{E}\left[\frac{\partial^2 A_{\exp}}{\partial x_i^2}\right] - \mathbf{E}\left[\frac{\partial^2 B_{\exp}}{\partial x_i^2}\right] = \frac{\beta_{ij}[(y_i-y_j)^2-(x_i-x_j)^2]}{d_{ij}^4}$$

$$-\frac{1}{2\lambda_{ij}^2 d_{ij}^4}\left[2\lambda_{ij} d_{ij}^3 + \beta_{ij}\lambda_{ij}^2 d_{ij}^2 - 2(x_i-x_j)^2\lambda_{ij} d_{ij} - \beta_{ij}(\frac{\beta_{ij}}{2}+2)(x_i-x_j)^2\lambda_{ij}^2\right]$$

$$= -\frac{\sin^2\alpha_{ij}}{d_{ij}\lambda_{ij}} + \frac{\beta_{ij}^2\cos^2\alpha_{ij} + 4\beta_{ij}\sin^2\alpha_{ij} - 2\beta_{ij}}{4d_{ij}^2}. \tag{3.49}$$

Similarly, we can obtain

$$-\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i,\theta_j)}{\partial y_i^2}\right] = -\frac{\cos^2\alpha_{ij}}{d_{ij}\lambda_{ij}} + \frac{\beta_{ij}^2\sin^2\alpha_{ij} + 4\beta_{ij}\cos^2\alpha_{ij} - 2\beta_{ij}}{4d_{ij}^2}, \tag{3.50}$$

$$-\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i,\theta_j)}{\partial x_i \partial y_i}\right] = \frac{\cos\alpha_{ij}\sin\alpha_{ij}}{d_{ij}\lambda_{ij}} + \frac{\beta_{ij}\cos\alpha_{ij}\sin\alpha_{ij}\left[d_{ij} + \lambda_{ij}(\frac{\beta_{ij}}{4}-1)\right]}{\lambda_{ij} d_{ij}^2}. \tag{3.51}$$

As we can see, comared to (3.36)-(3.38), the results here (3.49)-(3.51) have additional terms, representing the benefit of the knowledge of the variance model.

Then, following similar steps and using the symmetric property between $x_i(y_i)$ and $x_j(y_j)$, we can obtain

$$-\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i,\theta_j)}{\partial x_i \partial x_j}\right] = \mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i,\theta_j)}{\partial x_i^2}\right]$$
$$= \frac{\sin^2\alpha_{ij}}{d_{ij}\lambda_{ij}} - \frac{\beta_{ij}^2\cos^2\alpha_{ij} + 4\beta_{ij}\sin^2\alpha_{ij} - 2\beta_{ij}}{4d_{ij}^2}, \tag{3.52}$$

$$-\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i,\theta_j)}{\partial y_i \partial y_j}\right] = \mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i,\theta_j)}{\partial y_i^2}\right]$$
$$= \frac{\cos^2\alpha_{ij}}{d_{ij}\lambda_{ij}} - \frac{\beta_{ij}^2\sin^2\alpha_{ij} + 4\beta_{ij}\cos^2\alpha_{ij} - 2\beta_{ij}}{4d_{ij}^2}, \tag{3.53}$$

$$
\begin{aligned}
-\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i,\theta_j)}{\partial x_i \partial y_j}\right] &= -\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i,\theta_j)}{\partial y_i \partial x_j}\right] = -\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i,\theta_j)}{\partial y_j \partial x_i}\right] \\
&= -\mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i,\theta_j)}{\partial x_j \partial y_i}\right] = \quad \mathbf{E}\left[\frac{\partial^2 L_{\exp}(\theta_i,\theta_j)}{\partial x_i \partial y_i}\right] \\
&= \frac{\cos\alpha_{ij}\sin\alpha_{ij}}{d_{ij}\lambda_{ij}} - \frac{\beta_{ij}\cos\alpha_{ij}\sin\alpha_{ij}\left[d_{ij}+\lambda_{ij}(\frac{\beta_{ij}}{4}-1)\right]}{\lambda_{ij}d_{ij}^2}. \qquad (3.54)
\end{aligned}
$$

Again, the FIM of the exponential distributed range estimation noise with a distance-dependent variance model can easily obtained by using the above results, where (3.49) - (3.51) are used to evaluate the diagonal terms in the FIM, while (3.52) - (3.54) are used to evaluate the non-diagonal terms in the FIM.

Chapter 4

Optimal Performance of Collaborative Position Location

4.1 Motivation and Related Work

The benefits from node collaboration have been well acknowledged. For example, by updating those nodes with enough connections to anchors as *virtual* anchors once they have been localized, and subsequently using them to assist neighboring nodes in determining their locations, the location coverage can be increased. More importantly, due to the additional measurements among unlocalized nodes, collaborative position location holds the potential to improve localization accuracy. In fact, the authors in [24] have shown that adding unlocalized nodes into a network will strictly lower the Cramer-Rao lower bound (CRLB) on the localization error as long as the newly-added nodes satisfy a simple connectivity requirement. In view of these benefits, many collaborative position location algorithms have been proposed. Depending on the nature of the computation, existing algorithms can be classified as either centralized or distributed algorithms [11]. For instance, the authors in [24] formulated the (centralized) maximum likelihood estimate (MLE) for distance-based localization, where inter-node distance can be derived from either time-of-arrival (TOA) or received signal strength (RSS) measurements. In [15, 19, 25, 30], several other centralized approaches have been developed for collaborative position location. On the other hand, distributed algorithms, e.g., [16, 17, 18], have been proposed to address the scalability and computational complexity issues associated with centralized approaches. Some other algorithmic developments and a more detailed survey can be found in [11]. Note that, using angle-of-arrival

(AOA) as either a substitute [75] or supplemental measurement [56], and filtering techniques based on time-varying measurements [76] are also applicable to collaborative position location, but are not the focus of this work.

Despite many algorithms being developed, the optimal performance of collaborative position location has not been fully investigated. Specifically, although the closed-form expression of the MLE is given in [24], there is generally no closed-form expression for the global optimum solution, due mainly to the nonlinear and nonconvex nature of the associated optimization problem, where existing numerical methods, such as trust-region method or Levenberg-Marquardt method, may return a local instead of the global optimum. Consequently, the authors in [25] and [30] resort to semi-definite programming (SDP) and second-order cone programming (SOCP) relaxations, respectively, to the original MLE problem. However, both SDP- and SOCP-relaxed problems are no longer the original MLE and thus do not demonstrate the optimal performance. The authors in [37] presented the MLE for collaborative localization using RSS measurements. However, the MLE is only evaluated via a grid-based solution search strategy, and thus is not guaranteed to be optimal and can be computationally difficult. On the other hand, metaheuristic algorithms, (e.g., simulated annealing), have also been proposed to address the solution optimality issue in [38]-[39]. In addition, numerical methods such as random restart [77], benefiting from the fast convergence property of Newton's methods, provide an alternative to finding the global optimum from local optima resulting from random initializations. Nevertheless, it is known that these methods cannot guarantee that the solution is in fact optimal and may become computationally complex.

In this chapter, we seek to fill this void, by developing and examining an approach which produces the globally optimal (in the ML sense) solution. Specifically, we develop a brand-and-bound (BB) solution search strategy, coupled with the reformulation linearization technique (RLT) originally developed in [84], to numerically solve the MLE associated with collaborative position location, as given in [24]. Our main contribution includes the mathematical development of the BB/RLT solution procedure for the MLE problem and the design of several convergence speed-up techniques. To the best of our knowledge, the MLE for collaborative position location has only been approximately solved by existing work. We believe our result is significant in the sense that it offers a means to evaluate the optimal performance of collaborative position location. We compare the performance of the MLE with several existing collaborative position location schemes and demonstrate that it can be used as a good performance benchmark for evaluating practical position location schemes. Additionally, we show that for non-collaborative position location, the CRLB for an unbi-

ased estimator (despite its wide-spread use for this problem) sometimes is not a meaningful performance benchmark (this was also observed in [79, 80]). The reason for this is that the MLE is in general a biased estimator and thus can sometimes have a mean square error (MSE) smaller than the CRLB. As a result, the MLE can serve as a more practical performance benchmark. Finally, we emphasize that for the sake of simplicity and in order to be consistent with most of the existing work on collaborative position location, we focus on TOA distance-based position location with a Gaussian noise model. However, the extension to other measurement data such as AOA, and noise models other than Gaussian, can be carried out in a similar way as long as the corresponding MLE is properly formulated.

4.2 The MLE for Collaborative Position Location

We use the notations defined in Chapter 2 and consider TOA-based range estimates corrupted by independent zero-mean Gaussian random noise. To recall, the range estimate between the ith and jth nodes is modeled as

$$r_{ij} = d_{ij} + n_{ij}, \tag{4.1}$$

where $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ is the true inter-node distance and $n_{ij} \sim \mathcal{N}(0, \sigma_{ij}^2)$ is a zero-mean Gaussian random variable with a variance of σ_{ij}^2. Note that in practice, the TOA measurement noise model is more complex than what is assumed here, especially in the presence of non-line-of-sight (NLOS) propagation. However, to be consistent with most existing work and facilitate the performance comparison, we adopt the Gaussian noise model and the incorporation of the NLOS is left as future work. Based on this model, the MLE for collaborative position location can be formulated as

$$\begin{aligned} \hat{\Theta}_{\mathrm{ML}} &= \arg\min_{\Theta} f(\Theta) \\ &= \arg\min_{\Theta} \sum_{i=1}^{N} \sum_{\substack{j=i+1 \\ j\in\mathcal{H}(i)}}^{N+M} \frac{1}{\sigma_{ij}^2} \left(r_{ij} - \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \right)^2 \end{aligned} \tag{4.2}$$

where $\mathcal{H}(i) = \{q \,|\, q = 1, 2, \ldots, N+M,\ d_{iq} \leq R\}$ denotes the set of the ith node's neighboring nodes. Essentially, the MLE is equivalent to minimizing the sum of the weighted squared-difference between the estimated and the measured inter-node distances. We need to emphasize that in the case of equal noise variance, *i.e.*, $\sigma_{ij}^2 = \sigma^2,\ \forall i, j$, the MLE does not have to include the weighting factor σ_{ij}^2, as in [24], while we always include σ_{ij}^2 into

$f(\Theta)$. Note that this does not alter the optimal solution procedure. On the other hand, if we include distance-dependent noise variance model explicitly, the BB/RLT framework can still be applied. But a different mathematical formulation is needed. Strictly speaking, although we assume the true node locations are constrained to be within the $(\mathcal{L} \times \mathcal{L})\,\mathrm{m}^2$ square network, it is possible that in the optimal solution to (4.2), a node's estimated location may fall outside the network depending upon its true location and the associated range estimates. This is more likely to happen if a node's true location is close to the network boundary and its range estimates are noisy such that a location outside the network leads to the optimal solution to (4.2). It is also worth mentioning that if multiple observations, i.e., range estimates, are available, the MLE can be formulated accordingly. As the number of observations goes to infinity, the MLE becomes asymtotically consistent and normal. However, in practice, multiple observations are generally not available for the position location problem, mainly due to the power consumption involved in inter-node communication and signal processing for performing range estimation.

Finally, as can be seen, the MLE problem given in (4.2) is a nonlinear and nonconvex optimization problem and there is generally no closed-form expression for the optimal solution. This is true even for the case of non-collaborative position location. Due to the difficulty of obtaining a closed-form expression for the optimal solution to (4.2), several alternatives have been developed. For example, in the case of non-collaborative position location (where (4.2) can be broken into n separate minimization problems), the authors in [70] proposed the linear least-squares (LLS) location estimator by linearizing the range estimates prior to location estimation. For collaborative position location, the authors in [25] resorted to a semi-definite programming (SDP) relaxation to the MLE problem in (4.2) and a global optimal solution to the SDP-relaxed problem can be obtained. In this paper, however, we focus on solving the original MLE problem in (4.2). To facilitate the comparison between the MLE and other existing methods, we define the following metrics. First, we term the unitless objective value of the MLE, *i.e.*, $f(\Theta)$, as the *residual* of Θ. Second, the *network-average* error Ω and the *mean* localization error $\overline{\Omega}$ are defined as (2.1) and (2.2) in Chapter 2, respectively.

4.3 The BB/RLT framework for solving the MLE

In this section, we first give a brief overview of the generic BB/RLT algorithm. Then, we describe in detail how we reformulate and linearize the optimization problem given by (4.2) and the necessary algorithm components. Finally, we discuss its computational complexity

and propose a few convergence speed-up techniques.

4.3.1 BB/RLT algorithm

The BB/RLT algorithm is outlined in [84] and was applied to finding the capacity of a multiuser MIMO system with interference in [83]. In general, for a nonlinear and nonconvex programming problem as in (4.2), nonlinear optimization techniques such as trust region method, Levenberg-Marquardt method or metaheuristic approaches cannot offer any performance guarantee on the final solution. The BB/RLT, on the other hand, is an effective technique to find a provably global optimum for a nonconvex programming problem. The principal idea of the BB/RLT is that, via the RLT, we can construct a linear programming (LP) relaxation to the original nonlinear minimization problem. It can be proved that the objective value of this LP problem serves as a lower bound (LB) on the original minimization problem, or an upper bound (UB) on a maximization problem [84]. Depending upon how the linear relaxation is carried out, the LP solution is either a feasible solution, or can be used as a starting point to perform a local search in order to find a feasible solution, to the original nonlinear programming problem. In our problem, the local search is not necessary since we induce the same constraints on x_i and y_i given in (4.2) into the LP and the LP solution is thus always feasible to the original problem. The objective value of the *original* problem using the LP solution is then an UB on the original minimization problem (or an LB on a maximization problem). With the relative ease of solving an LP problem and the guaranteed optimal solution, the BB strategy is brought in to iteratively bi-partition the solution space while tightening the UB and LB until the pre-determined optimality condition $\text{LB} > (1-\epsilon)\text{UB}$ is satisfied, where ϵ is the target optimality parameter. In fact, it has been proved that as long as the partitioning intervals are compact, the BB/RLT converges to the globally optimal solution [84]. The generic BB/RLT algorithm for a minimization problem is outlined in Fig. 4.1. In the following, we describe how we construct the LP relaxation and apply the BB/RLT solution procedure to the MLE problem.

BB/RLT solution procedure

Initialization:

1. Let the optimal solution $\psi^* = \emptyset$ and initialize UB $= +\infty$.
2. Initialize problem list with only the original problem, denoted by P_1.
3. Apply the RLT to P_1, determine partitioning variables and their initial bounding intervals. Obtain the LP solution $\underline{\phi}_1^*$, denote the LP objective value as LB_1. Let LB $= \text{LB}_1$.

Main Loop:

1. From the problem list, select P_k with the minimum LB_k.
2. Let LB $= \text{LB}_k$.
3. Find, if necessary, a feasible solution to the original problem ψ_k, via a local search from the LP solution $\underline{\phi}_k^*$; otherwise, let $\psi_k = \underline{\phi}_k^*$. Denote ψ_k's objective value in the original problem as UB_k.
4. If $\text{UB}_k <$ UB
   Update $\psi^* = \psi_k$ and UB $= \text{UB}_k$;
   If LB $> (1-\epsilon)$UB
   stop and return ψ^* as the $(1-\epsilon)$-optimal solution.
   Otherwise
   remove all problems k' from the problem list satisfying $\text{LB}_{k'} \geq (1-\epsilon)$UB.
5. Find the maximum relaxation error among all RLT variables and divide the underlying partitioning variable's current bounding interval into two new intervals at its corresponding value in $\underline{\phi}_k^*$.
6. Solve the two new LP sub-problems P_{k1} and P_{k2} associated with the two new intervals and denote their objective values as LB_{k1} and LB_{k2}.
7. Remove problem P_k from the problem list.
8. If $\text{LB}_{k1} < (1-\epsilon)$UB, add P_{k1} into the problem list.
   If $\text{LB}_{k2} < (1-\epsilon)$UB, add P_{k2} into the problem list.
9. If the problem list is empty
   stop and return ψ^* as the $(1-\epsilon)$-optimal solution.
   Otherwise
   go to the next iteration.

Figure 4.1: Generic branch-and-bound/reformulation linearization technique (BB/RLT) solution procedure.

4.3.2 Reformulation and linearization of the MLE

To apply the RLT, we first linearize the objective function $f(\Theta)$ by making the following variable changes

$$Z_{ij} = \sqrt{W_{ij}} \tag{4.3}$$
$$W_{ij} = (x_i - x_j)^2 + (y_i - y_j)^2, \tag{4.4}$$

and the linearized objective function $\ell(Z, W)$ is then given by

$$\ell(Z, W) = \sum_{i=1}^{N} \sum_{\substack{j=i+1 \\ j \in \mathcal{H}(i)}}^{N+M} \frac{1}{\sigma_{ij}^2} \left(r_{ij}^2 - 2\, r_{ij} Z_{ij} + W_{ij} \right). \tag{4.5}$$

where $Z = \{Z_{ij}\}$ and $W = \{W_{ij}\}$ are the sets of existing Z_{ij} and W_{ij}, respectively, meaning that if there is a range estimate between the ith and the jth nodes, Z_{ij} and W_{ij} exist.

Second, we need to linearize the constraint given by the square root function in (4.3). Specifically, given lower and upper bounds W_{ij}^L and W_{ij}^U for W_{ij}, two tangential lines of the square root function can be determined at W_{ij}^L and W_{ij}^U, respectively. If the middle point between W_{ij}^L and W_{ij}^U is denoted by W_{ij}^β, *i.e.*,

$$W_{ij}^\beta = \frac{1}{2}(W_{ij}^L + W_{ij}^U), \tag{4.6}$$

another tangential line of the square root function can be defined at W_{ij}^β. In addition, the chord can be formed by connecting the two end points on the square root function, corresponding to W_{ij}^L and W_{ij}^U, respectively. With all four tangential lines, a polygonal bounding region can be defined via the set of the following linear inequalities,

$$-W_{ij} + 2\sqrt{W_{ij}^L}\, Z_{ij} \leq W_{ij}^L, \tag{4.7}$$
$$-W_{ij} + 2\sqrt{W_{ij}^U}\, Z_{ij} \leq W_{ij}^U, \tag{4.8}$$
$$-W_{ij} + 2\sqrt{W_{ij}^\beta}\, Z_{ij} \leq W_{ij}^\beta, \tag{4.9}$$
$$W_{ij} - \left(\sqrt{W_{ij}^L} + \sqrt{W_{ij}^U} \right) Z_{ij} \leq -\sqrt{W_{ij}^L W_{ij}^U}. \tag{4.10}$$

An example of such a polygonal bounding region, for $W_{ij}^L = 0.2$ and $W_{ij}^U = 2$, is illustrated in Fig. 4.2, where tangential lines A, B, C and D form the boundaries of the polygonal bounding region. We need to point out that the choice of W_{ij}^β is not unique, as long as its associated

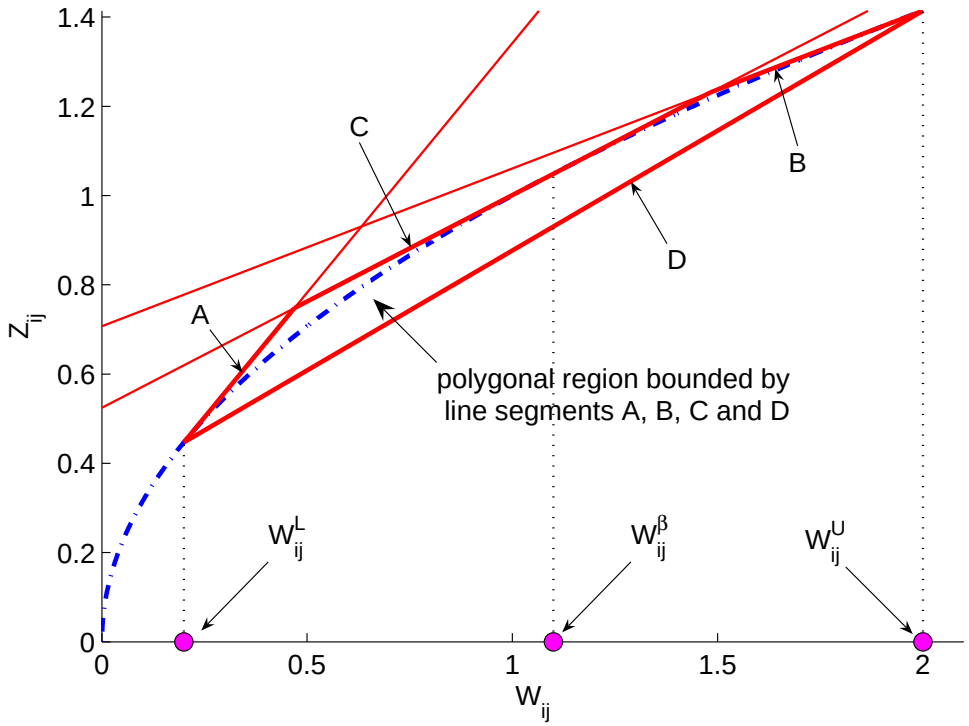


Figure 4.2: Polygonal bounding region of $Z_{ij} = \sqrt{W_{ij}}$ for $W_{ij}^L = 0.2$ and $W_{ij}^U = 2$. The four tangential lines define a polyhedral bounding region for the square root function between W_{ij}^L and W_{ij}^U.

tangential line is properly defined as in (4.9). An alternative is to choose the intersection point of the two tangential lines respectively defined at W_{ij}^L and W_{ij}^U, as in [83]. However, as we will discuss shortly, the choice of W_{ij}^β does have an impact on the convergence speed of the algorithm for our particular MLE problem.

Now, we linearize the quadratic constraint given by (4.4), which is related to the estimated distance between the ith and jth nodes. For that, we define

$$U_i = x_i^2, \tag{4.11}$$
$$V_i = y_i^2, \tag{4.12}$$

for $i = 1, 2, \ldots, N$. In addition, if $j \leq N$, *i.e.*, node j is an unlocalized node, we define

$$S_{ij} = x_i\, x_j, \tag{4.13}$$
$$T_{ij} = y_i\, y_j, \tag{4.14}$$

Substituting (4.11)-(4.14) into (4.4), we obtain the following linearized equality constraints for $i = 1, 2, \ldots, N$; $j = i+1, i+2, \ldots, N+M$; and $j \in \mathcal{H}(i)$,

i) if $j \leq N$, *i.e.*, node j is an unlocalized node:

$$W_{ij} - U_i + 2\,S_{ij} - U_j - V_i + 2\,T_{ij} - V_j = 0\,; \tag{4.15}$$

ii) if $j \geq N+1$, *i.e.*, node j is an anchor:

$$W_{ij} - U_i + 2x_i x_j - x_j^2 - V_i + 2y_i y_j - y_j^2 = 0. \tag{4.16}$$

With (4.15) and (4.16), we now linearize the quadratic constraints defined by (4.11)-(4.14). In particular, assuming the lower and upper bounds for x_i and y_i are given by x_i^L, x_i^U and y_i^L, y_i^U, respectively, we have, for x_i,

$$x_i - x_i^L \geq 0; \tag{4.17}$$

$$x_i^U - x_i \geq 0; \tag{4.18}$$

Adopting the RLT [84], we can derive the following so-called bounding-factor constraints,

$$x_i^2 - 2x_i^L x_i \;\geq\; -(x_i^L)^2 \tag{4.19}$$

$$x_i^2 - 2x_i^U x_i \;\geq\; -(x_i^U)^2 \tag{4.20}$$

$$x_i^2 - (x_i^L + x_i^U)x_i \;\leq\; -x_i^L x_i^U. \tag{4.21}$$

Now, substituting (4.11) into (4.19)-(4.21), we obtain three linear constraints involving U_i and x_i, which essentially define a triangular bounding region for (4.11) when $x_i^L \leq x_i \leq x_i^U$. Similar bounding-factor constraints can be derived for V_i via y_i.

Further, as for S_{ij}, we have, for x_i and x_j,

$$x_i - x_i^L \geq 0; \tag{4.22}$$

$$x_i^U - x_i \geq 0; \tag{4.23}$$

$$x_j - x_j^L \geq 0; \tag{4.24}$$

$$x_j^U - x_j \geq 0. \tag{4.25}$$

Again using the RLT, the bounding-factor constraints are given by

$$x_i x_j - x_i^L x_j - x_j^L x_i \;\geq\; -x_i^L x_j^L \tag{4.26}$$

$$-x_i x_j + x_i^L x_j + x_j^U x_i \;\geq\; x_i^L x_j^U \tag{4.27}$$

$$-x_i x_j + x_i^U x_j + x_j^L x_i \;\geq\; x_i^U x_j^L \tag{4.28}$$

$$x_i x_j - x_i^U x_j - x_j^U x_i \;\geq\; -x_i^U x_j^U. \tag{4.29}$$

Substituting (4.13) into (4.26)-(4.29), we obtain four linear constraints involving S_{ij}, x_i and x_j, which essentially define a polyhedral bounding region for (4.13) when $x_i^L \le x_i \le x_i^U$ and $x_j^L \le x_j \le x_j^U$.

Similar using the lower and upper bounds on y_i and y_j, *i.e.*,

$$y_i - y_i^L \ge 0; \tag{4.30}$$
$$y_i^U - y_i \ge 0; \tag{4.31}$$
$$y_j - y_j^L \ge 0; \tag{4.32}$$
$$y_j^U - y_j \ge 0, \tag{4.33}$$

we obtain the bounding-factor constraints as follows,

$$\begin{aligned} y_i y_j - y_i^L y_j - y_j^L y_i &\ge -y_i^L y_j^L && (4.34)\\ -y_i y_j + y_i^L y_j + y_j^U y_i &\ge y_i^L y_j^U && (4.35)\\ -y_i y_j + y_i^U y_j + y_j^L y_i &\ge y_i^U y_j^L && (4.36)\\ y_i y_j - y_i^U y_j - y_j^U y_i &\ge -y_i^U y_j^U. && (4.37) \end{aligned}$$

Then, replacing T_{ij} into (4.34) - (4.37), we the four linear constraints on T_{ij}.

By combining all the aforementioned linear constraints with the linearized objective function $\ell(Z, W)$, the LP relaxation to the original MLE problem in (4.2) is given by

$$\underline{\phi}^* = \arg\min_{\phi} \ell(Z, W) = \arg\min_{\phi} \sum_{i=1}^{N} \sum_{\substack{j=i+1 \\ j\in\mathcal{H}(i)}}^{N+M} \frac{1}{\sigma_{ij}^2}(r_{ij} - 2\, r_{ij} Z_{ij} + W_{ij}) \tag{4.38}$$

s.t.: polygonal bounding constraints on (Z_{ij}, W_{ij}) from (4.7)-(4.10),
equality constraints in (4.15) and (4.16),
linear constraints on U_i derived from (4.19)-(4.21),
linear constraints on V_i, similar to U_i,
linear constraints on S_{ij} derived from (4.26)-(4.29),
linear constraints on T_{ij}, similar to S_{ij}.

In (4.38), the set of optimization variables $\phi = \{\Theta, Z, W, U, V, S, T\}$, where $U = \{U_i\}$, $V = \{V_i\}$, $S = \{S_{ij}\}$ and $T = \{T_{ij}\}$ are the sets of U_i, V_i, existing S_{ij} and T_{ij}, respectively. The underline in the notation $\underline{\phi}^*$ emphasizes the fact that the resulting LP objective value is an LB on the optimal objective value of the original MLE problem.

4.3.3 Partitioning variables, initial bounding interval and partitioning rule

Given the above LP relaxation, the BB solution search can be applied. The partitioning variables in the BB process are those from the original problem, *i.e.*, Θ. The RLT variables, on the other hand, are those additional variables introduced for the purpose of LP relaxation, *i.e.*, $\{Z, W, U, V, S, T\}$. As shown by the generic BB/RLT algorithm in Fig. 4.1, once the initial bounding interval for Θ is given, we can easily evaluate the lower and upper bounds for W_{ij} according to (4.4) and the initial LP relaxation can be formulated. Then, in the BB process, the bounding interval for Θ will be iteratively partitioned. For each newly-generated bounding interval, a new LP relaxation is formulated by applying the similar RLT steps described in Section III. B.

We need to point out that the initial bounding interval for Θ is necessary to perform the LP relaxation. Although the original MLE problem in (4.2) does not have any constraints on Θ, for the purpose of position location, the knowledge of the network size and noise variance can be utilized to set the initial bounding interval. Therefore, as demonstrated in our simulation results, we always set the initial bounding interval for Θ to be sufficiently large such that it is guaranteed to include the global optimal solution to (4.2).

In each iteration of the BB/RLT algorithm, the partitioning variable resulting in the maximum relaxation error in the associated RLT variable is selected and its bounding interval is partitioned, while the bounding intervals of other partitioning variables remain the same. Note that since the choice of partitioning variable does not affect the final convergence result, there exist other rules for selecting the partitioning variable [84]. In addition, as mentioned in [84], the sequence of the maximum relaxation error will approach zero as the BB process goes on. Therefore, we adopt this rule for selecting partitioning variable in an effort to facilitate algorithm convergence. The relaxation error of an RLT variable is defined to be the absolute difference between its value in the LP solution $\underline{\phi}^*$ and the one calculated using the underlying partitioning variable(s). If an RLT variable involves more than one partitioning variable, the one currently with the largest bounding interval will be chosen. For example, the relaxation error of the RLT variable S_{ij} is given by

$$\Delta S_{ij} \stackrel{\triangle}{=} |\underline{S}^*_{ij} - \underline{x}^*_i \underline{x}^*_j|$$

where $\underline{S}^*_{ij}$, $\underline{x}^*_i$ and $\underline{x}^*_j$ are the values of S_{ij}, x_i and x_j in the LP solution $\underline{\phi}^*$, respectively. If ΔS_{ij} is the maximum relaxation error, we then examine the current lower and upper bounds

for x_i and x_j. If $(x_i^U - x_i^L) > (x_j^U - x_j^L)$, x_i is selected and its two new bounding intervals are $[x_i^L \ \underline{x}_i^*]$ and $[\underline{x}_i^* \ x_i^U]$; otherwise, x_j is selected and its two new bounding intervals are $[x_j^L \ \underline{x}_j^*]$ and $[\underline{x}_j^* \ x_j^U]$. Other partitioning variables are selected in a similar fashion.

4.3.4 Computational complexity and convergence speed-up techniques

The number of variables N_{var} and the number of constraints N_{con} of the relaxed LP problem (4.38) are calculated as

$$N_{\text{var}} = 4N + 2\sum_{i=1}^{N} N^{(i)} + 2\sum_{i=1}^{n} N_u^{(i)} \tag{4.39}$$

$$N_{\text{con}} = 10N + 5\sum_{i=1}^{N} N^{(i)} + 8\sum_{i=1}^{N} N_u^{(i)} \tag{4.40}$$

where N is the total number of unlocalized nodes and

$$N^{(i)} = \text{card}\{q \,|q \in \mathcal{H}(i), i+1 \le q \le N+M\} \tag{4.41}$$

$$N_u^{(i)} = \text{card}\{q \,|q \in \mathcal{H}(i), i+1 \le q \le N\} \tag{4.42}$$

denote the total number of the ith node's *effective* (*i.e.*, $r_{ij} = r_{ji}$ is only counted once) range estimates to other nodes and to other unlocalized nodes, respectively, and card$\{\cdot\}$ denotes the cardinality. In the extreme case, e.g., a large and fully-connected network, the computational complexity of the BB/RLT becomes high. However, we can explore the specific problem structure to develop some convergence speed-up techniques. As will be demonstrated earlier, the convergence speed of the BB/RLT algorithm is dominated by the LB, which is determined by the objective value, *i.e.*, $\ell(Z, W)$, of the LP problem in (4.38). This suggests that techniques that are able to reduce the search space for Z and W can improve the algorithm convergence speed. Consequently, we propose an improved polygonal bounding region for the square root function in (4.3). Specifically, rather than using only one point, *i.e.*, W_{ij}^{β}, we pick k_1 points $W_{ij}^{\beta_1}, W_{ij}^{\beta_2}, \ldots, W_{ij}^{\beta_{k1}}$ between W_{ij}^L and W_{ij}^U, satisfying

$$W_{ij}^{\beta_p} = (W_{ij}^{\beta_{p-1}} + W_{ij}^L)/2, \quad p = 1, 2, \ldots, k_1 \ \text{ and } \ W_{ij}^{\beta_0} \triangleq W_{ij}^U. \tag{4.43}$$

Then we define the corresponding k_1 tangential lines at these k_1 points. This will form a refined polygonal bounding region for (4.3), and thus reduce the search space for Z and

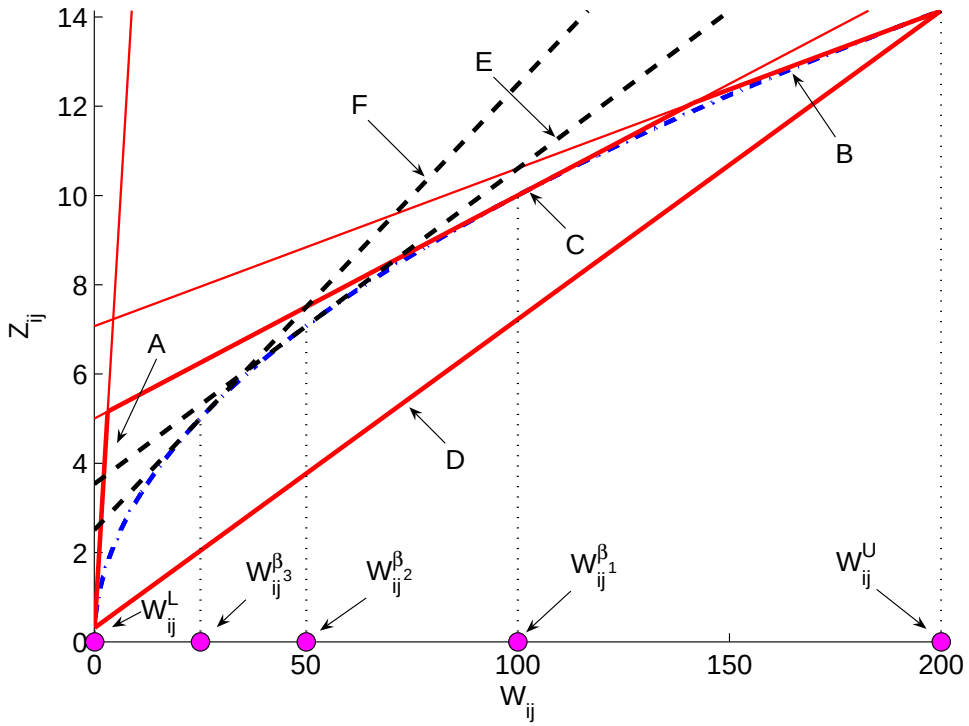


Figure 4.3: Improved polygonal bounding region of $Z_{ij} = \sqrt{W_{ij}}$ for $W_{ij}^L = 0.1$ and $W_{ij}^U = 200$. The two additional tangential lines (dash) E and F refine the bounding region and reduce the search space around the small value region of W_{ij}.

W. An example with $k_1 = 3$, $W_{ij}^L = 0.1$ and $W_{ij}^U = 200$ is shown in Fig. 4.3, where the two additional tangential lines E and F refine the polygonal bounding region and reduce the search space around the small value region of W_{ij}. Note that we choose to refine the lower half of the bounding interval, *i.e.*, between W_{ij}^L and $W_{ij}^{\beta_1}$, mainly because, for the square root function, this reduces the search space more than that if we choose to refine the upper half, *i.e.*, between $W_{ij}^{\beta_1}$ and W_{ij}^U. Similar techniques can also be applied to the bounding-factor constraints, e.g., those on U_i and x_i. However, our numerical results show that this has little impact on the algorithm convergence speed, simply because the convergence speed is dominated by the LB in our problem.

Finally, it is worth mentioning that although this technique can reduce the search space, which can in turn reduce the total number of iterations, it increases the size of the LP relaxation by adding additional constraints, and thus it takes longer time to solve each individual LP problem. This means that, depending on the nature of the problem, there exists a trade-off between the size of the LP relaxation and the total number of iterations

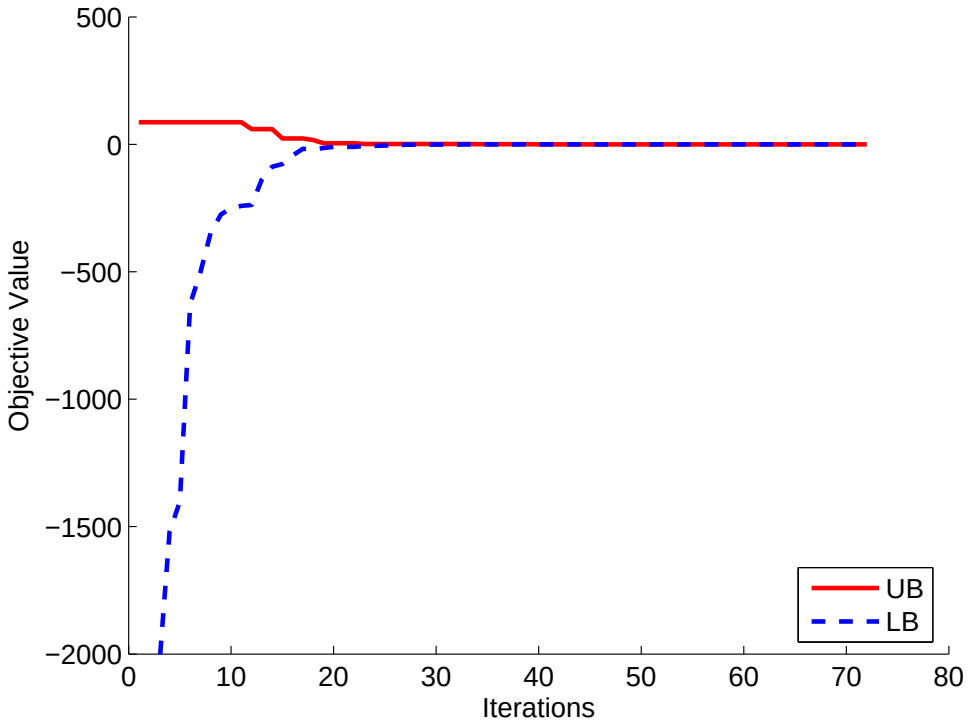


Figure 4.4: The convergence of the UB and LB on the MLE objective value for noise-free case, with 3 anchors at (5, 5), (10, 15) and (15, 5), and unlocalized node at (3, 5). The final objective value is found to be 1.17×10^{-6} and the solution is (2.999, 4.999).

in order for the algorithm to converge. We have observed in our simulations, using a small value of $k_1 = 2$ or $k_1 = 4$ can reduce the total algorithm running time, as compared to simply using $k_1 = 1$, which suggests that the benefit of reducing the search space by adding constraints outweighs the penalty from solving a larger LP problem.

4.4 Numerical results

In this section, we give simulation results using the BB/RLT algorithm to solve (4.2). In particular, we first consider the special case of non-collaborative position location, e.g., a single node surrounded by a few anchors, or a mobile station (MS) surrounded by a few base stations (BS) in a cellular system. This offers a simple way of validating our BB/RLT formulation of the MLE problem. After that, we investigate the case of collaborative position location with a moderate network size. For all simulations, we choose $\epsilon = 0.01$. The network size $\mathcal{L}$ and the communication range R will be self-evident in different simulations. The

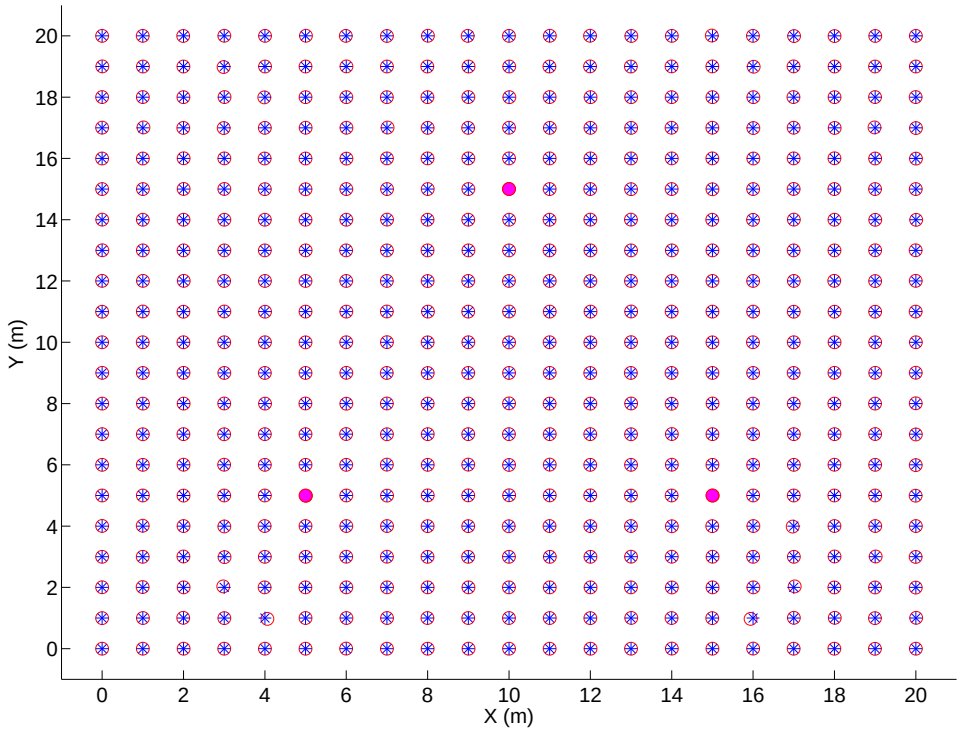


Figure 4.5: Validation of the BB/RLT framework for solving the MLE in the absence of range estimation noise. The three anchors (solid circles) are located at (5, 5), (10, 15) and (15, 5), respectively. The red circles are different tested true locations of the unlocalized node, while the blue asterisks are the MLE solution obtained by the BB/RLT. As shown, the BB/RLT always finds the true solution, which validates our framework formulation.

variance of the range estimation noise is assumed to be $\sigma_{ij}^2 = \sigma^2$. The initial bounding intervals of x_i and y_i are both set to be $[-\mathcal{L}\ \ 2\mathcal{L}]$ to ensure the true optimal solution is included in the search space. We emphasize that it is not the physical network size $\mathcal{L}$, but the size of the original MLE problem, *i.e.*, the number of unlocalized nodes and the number of range estimates, that determines how long it takes for the BB/RLT algorithm to converge. For non-collaborative position location, we compare the MLE solution obtained by the BB/RLT with the linear least-squares (LLS) location estimator [70] as well as the Cramer-Rao lower bound (CRLB) on localization error [24]. For collaborative localization, we compare our method with the semi-definite programming (SDP) approach [27] and the solution to the MLE obtained by the *lsqnonlin* solver in Matlab, which is based on trust region method or Levenberg-Marquardt method [85].

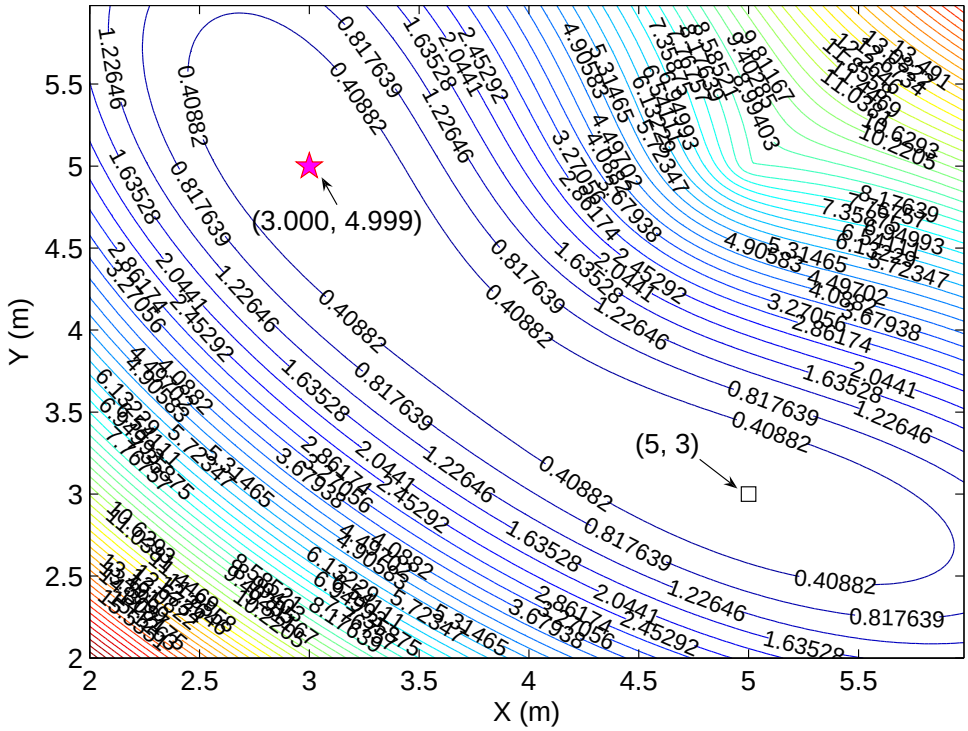


Figure 4.6: Illustration of contours of the residual. The solution (3, 5) is the global optimum, while the solution (5, 3) is a local optimum and indeed a flipped location solution. The two solutions have very similar residuals.

4.4.1 Validation of the BB/RLT formulation

First, we present the numerical validation of our BB/RLT formulation to the MLE problem. In Fig. 4.4, we plot the UB and LB at each iteration of the BB/RLT algorithm when an unlocalized node with a true location at (3, 5) uses *noise-free* range estimates to three anchors at (5, 5), (10, 15) and (15, 5) to compute its location. It is observed that, the UB and the LB converge to the global optimum (which is theoretically zero in the noise-free case) after about 70 iterations, and the optimal solution $(x^*, y^*) = (2.999, 4.999)$ is obtained, with a residual of 1.17×10^{-6}. From Fig. 4.4, it is easy to see that the LB plays the more critical role in determining the algorithm's final convergence, especially during the final iterations. Also note that although the initial LB is below -2500, the LB quickly converges to reasonable values near zero after about 15 iterations.

In Fig. 4.5, we provide a comprehensive test to validate our BB/RLT framework formulation. In this test, we place three anchors at (5, 5), (10, 15) and (15, 5) respectively, shown by

the solid circles. We then place a single unlocalized node at different testing positions (red circles) over a $20 \times 20\,\mathrm{m}^2$ network with a step size of 1 m. The unlocalized node is assumed to be able to communicate with all three anchors and obtain noise-free distance estimates. To validate the framework, we assume there is no range estimation noise, in which case, the solution computed by the BB/RLT should ideally converge to the true location. As shown by Fig. 4.5, for all tested locations, the solution obtained by the BB/RLT always converges to the true locations. This validates our BB/RLT framework formulation. Note that depending on the precision parameter ϵ, the MLE solution computed by the BB/RLT always has some slight numerical difference from the true solution, usually on the order of 10^{-3}.

In fact, the real power of the BB/RLT algorithm is demonstrated when the anchors are almost collinear. For example, consider an unlocalized node at (3, 5), two anchors at (5, 5) and (15, 15), and the 3rd anchor at (10, 10.1). With such an anchor setup, the anchors are almost collinear, but not exactly. In the absence of noise, we observed in our simulations that existing nonlinear optimization methods sometimes return the wrong solution of (5, 3), depending on the initialization, which is indeed the so-called flip ambiguity [50]. It is in fact a local optimum of the objective value which is very close to that of the true global optimum. This effect is shown by Fig. 4.6, where we plot the contours of the residual (the objective function) with the anchor setup and in the absence of range estimation noise. As we can see, the local optimum (square) at (5, 3) has an objective very close to that of the true global optimum (pentagon) at (3, 5), shown by the symmetric shape of the contour plot. In such cases, existing nonlinear optimization methods, such as *lsqnonlin* solver, often return the wrong solution. On the other hand, the BB/RLT is numerically guaranteed to find the optimal solution around (3, 5). In Table 4.1, we test the BB/RLT for different locations of the 3rd anchor, at (10, 10.1), (10, 10.01) or (10, 10.001) respectively, and give the final objective values, the BB/RLT solutions as well as the numbers of iterations it takes to converge, for the three cases respectively. We can see that in the order of the tested locations, it becomes more and more difficult to distinguish the flipped solution from the correct solution, since the objective values of the two are getting closer and closer. Therefore, it takes more and more iterations for the BB/RLT algorithm to converge. In addition, it requires a smaller and smaller precision parameter ϵ to guarantee the solution optimality.

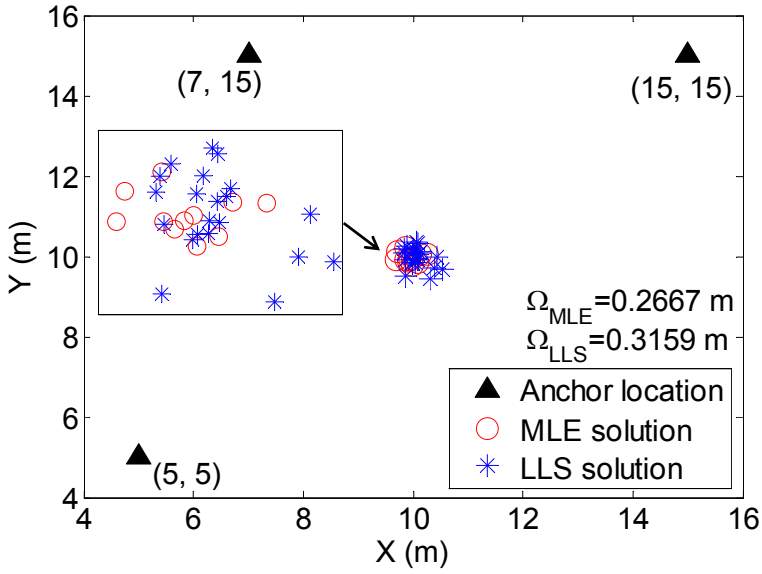


Figure 4.7: Example estimated locations using the LLS and the MLE, where $K_E = 0.001$ and $\beta = 2$ ($\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$), and the unlocalized node's true location is (10, 10), with good anchor geometry.

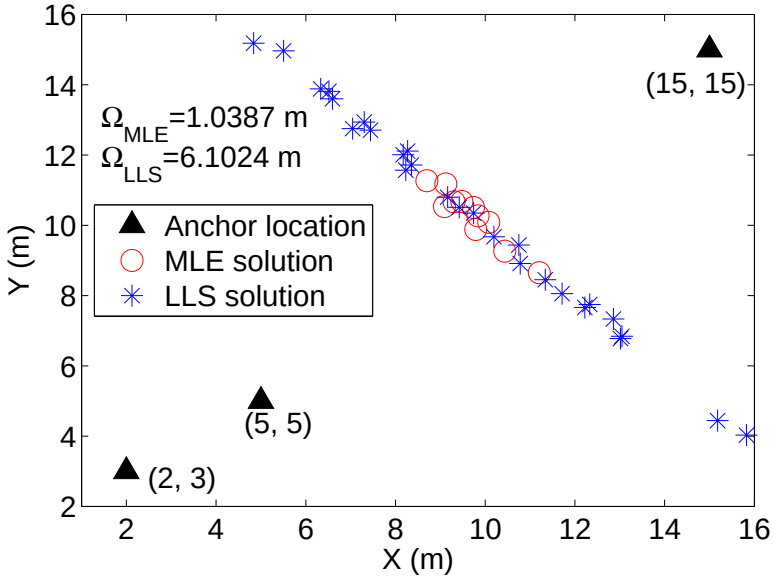


Figure 4.8: Example estimated locations using the LLS and the MLE, where $K_E = 0.001$ and $\beta = 2$ ($\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$), and the unlocalized node's true location is (10, 10), with bad anchor geometry.

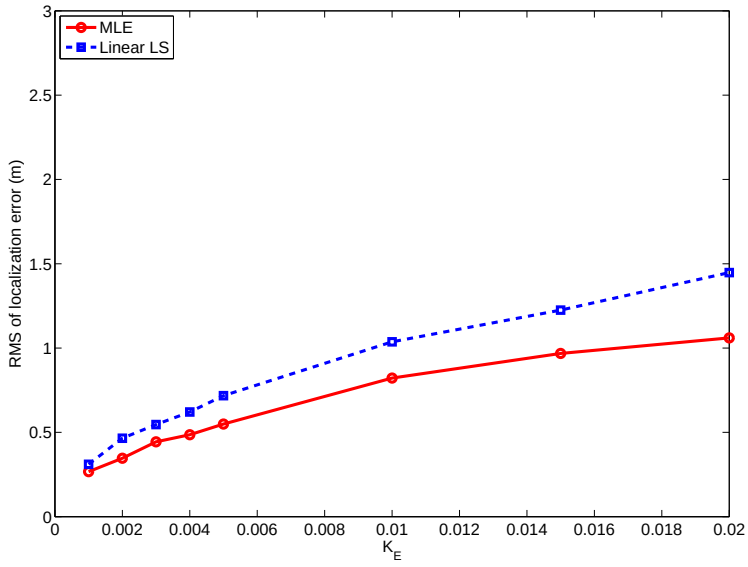


Figure 4.9: Mean localization errors for the LLS, the MLE, for the good anchor geometry in Fig. 4.7. K_E is the proportionality constant in modeling the noise variance, $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$.

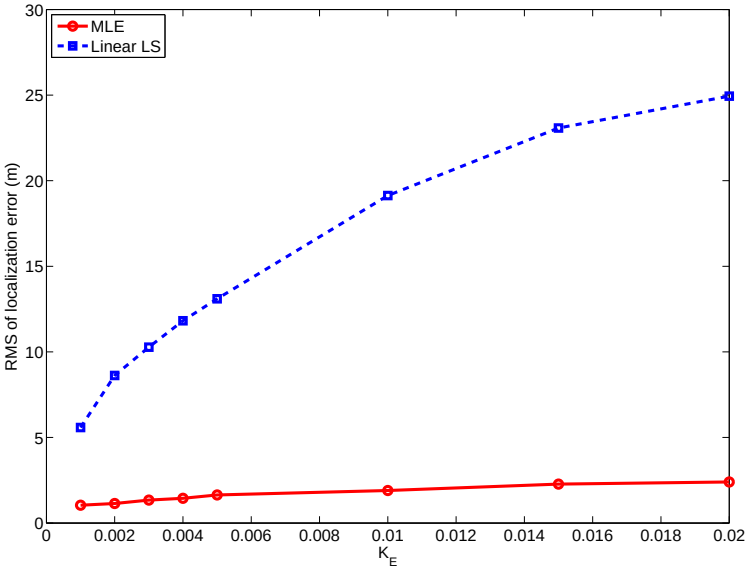


Figure 4.10: Mean localization errors for the LLS, the MLE, for the bad anchor geometry in Fig. 4.8. K_E is the proportionality constant in modeling the noise variance, $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$.

Table 4.1: The BB/RLT solutions and the final objective values of the MLE for an example of position location with three almost-collinear anchors

3rd anchor location	(10, 10.1)	(10, 10.01)	(10, 10.001)
objective value	3.6×10^{-6}	1.2×10^{-5}	2.5×10^{-6}
x^*	3.002	3.000	3.000
y^*	4.999	5.004	4.997
number of iterations	502	721	989

4.4.2 MLE versus LLS estimator

We now use the examples in Figs. 4.7 and 4.8 to further illustrate the effect of node geometry on the performance of the MLE solved by the BB/RLT and the LLS estimator. For the LLS estimator, we always use the first anchor as the reference to linearize the range estimates [70]. In Fig. 4.7, we have the unlocalized node's true location at (10, 10), while three anchors are located at (5, 5), (7, 15) and (15, 15), respectively, and therefore form a good anchor geometry. We simulate 100 noise realizations with $K_E = 0.001$ and $\beta = 2$ and plot some of the estimated locations by asterisks and circles (plotting all of the 100 estimated locations makes the figure not easily readable). As we can see, the estimated locations of both the MLE and the LLS estimator are around the unlocalized node's true location, where the MLE performs slightly better than the LLS. On the other hand, in Fig. 4.8, the three anchors are located at (5, 5), (2, 3) and (15, 15), almost being collinear. In this case, the LLS estimator performs extremely bad, in the sense than even a small amount of range estimation noise will lead to a very large mean localization error of 6.1 m. The MLE, however, still provides reasonable performance, where the estimated locations are around the true location, although more diverse than in Fig. 4.7, resulting in a mean localization error of only about 1.0 m. From these two figures, we can see that the MLE's performance is less sensitive to node geometry. In Figs. 4.9 and 4.10, we plot the mean localization error of the LLS and the MLE for $\beta = 2$ with respect to K_E. Again, we see that the optimal performance suggested by the MLE is robust to bad node geometry and much better than the LLS estimator. For example, when $K_E = 0.01$ (corresponding to an amount of range estimation noise equivalent to 10% of the inter-node distance), the mean localization error of the MLE is 0.2 m lower than the LLS in the case of good anchor geometry. In the case of bad anchor geometry, the performance of the MLE only degrades slightly, while that of the LLS becomes unacceptably bad, *i.e.*, approximately 20 m. Considering the case of a large number of sensors randomly spread

in the network, it is very likely an unlocalized node's neighbors will form a bad geometry. Therefore, the LLS in such cases is not a good choice.

4.4.3 MLE versus CRLB

In Figs. 4.11 and 4.12, we compare the BB/RLT solution to the MLE with the LLS location estimator, as well as the CRLB, for the case of non-collaborative position location. In Fig. 4.11, we plot the mean localization errors of the MLE and the LLS estimator for 1000 noise realizations with respect to σ^2. The unlocalized node's true location is (20, 20) and the four anchors' locations are at the four corners (0, 0), (40, 0), (0, 40) and (40, 40), respectively, which is a good node geometry. As can be seen, the MLE slightly outperforms the LLS, and is very close to the CRLB. This suggests that in the case of good node geometry, *i.e.*, the unlocalized node being well-surrounded by the anchors, the MLE performs as good as what the CRLB implies and the LLS location estimator performs fairly well. On the other hand, in Fig. 4.12, the four anchors' locations are almost-collinear, at (0, 0), (10, 11), (28, 29) and (40, 40) and the unlocalized node at (20, 20) lies almost on the same line roughly formed by connecting the four anchors, which is an extremely bad node geometry. Here the MLE not only significantly outperforms the LLS, but also is far more robust to range estimation noise than the LLS estimator. This is not too surprising since it is well-known that the LLS approach performs very poorly in bad anchor geometry. More surprisingly, however, the MLE performs even better than the CRLB. Despite being counter-intuitive, this has partly been observed in [79, 80]. Specifically, in the case of bad node geometry, e.g., almost but not exactly collinear, the CRLB does exist but becomes unreasonably large due to the fact that the Fisher information matrix (FIM) becomes nearly singular. In fact, it has been demonstrated in [79] that using a specially-designed position location method, the localization error can be significantly reduced. As we will show below, the phenomenon can be interpreted by the fact that the TOA-based MLE in (4.2) is in general a *biased* estimator. It is worth mentioning that the authors in [24] have explicitly shown that the RSS-based MLE is biased, but did not do so for the TOA-based MLE. With our results showing the TOA-based MLE is indeed biased and the fact that the MSE of a biased estimator can be much smaller than the CRLB, as in [81, 82], this observation is easily justified.

In Figs. 4.13-4.15, we present simulation results that demonstrate the bias of the TOA-based MLE and a comparison between the square root of the MSE and that of the CRLB. In addition, we apply the theoretical framework for the uniform CRLB (UCRLB) proposed

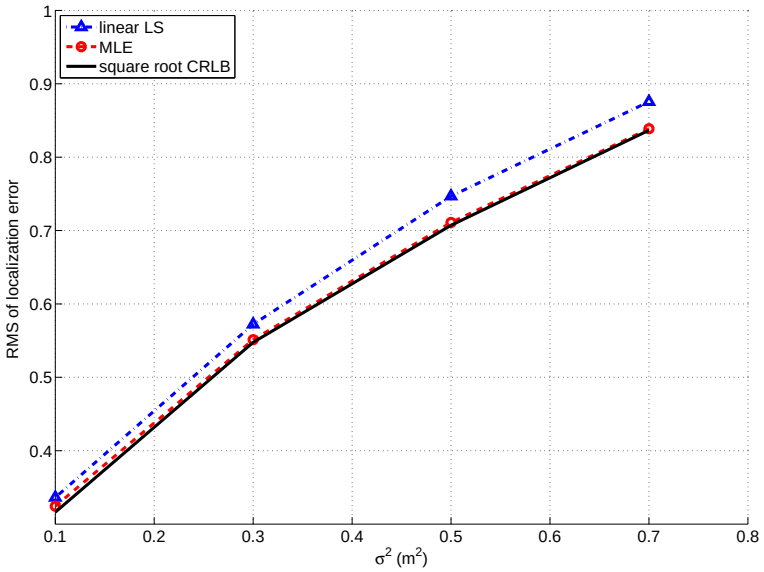


Figure 4.11: Mean localization errors of the LLS and the MLE versus the noise variance σ^2 with good geometry, where the CRLB serves as a good benchmark. Four anchors are placed at four corners (0 0), (0 40), (40 0), (40 40) and the unlocalized node is at (20 20).

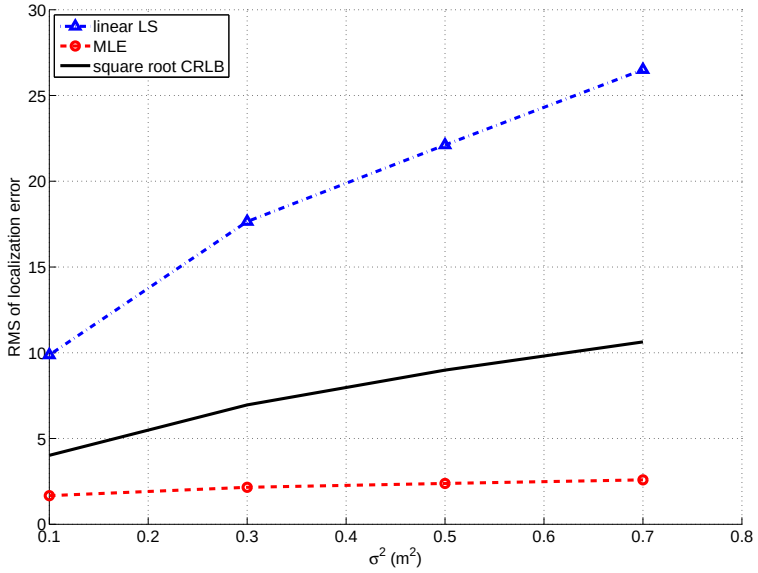


Figure 4.12: Mean localization errors of the LLS and the MLE versus the noise variance σ^2 with bad geometry, where the CRLB is no longer a meaningful benchmark. Four almost-collinear anchors are placed at (0 0), (10 11), (28 29), (40 40) and the unlocalized node is at (20 20).

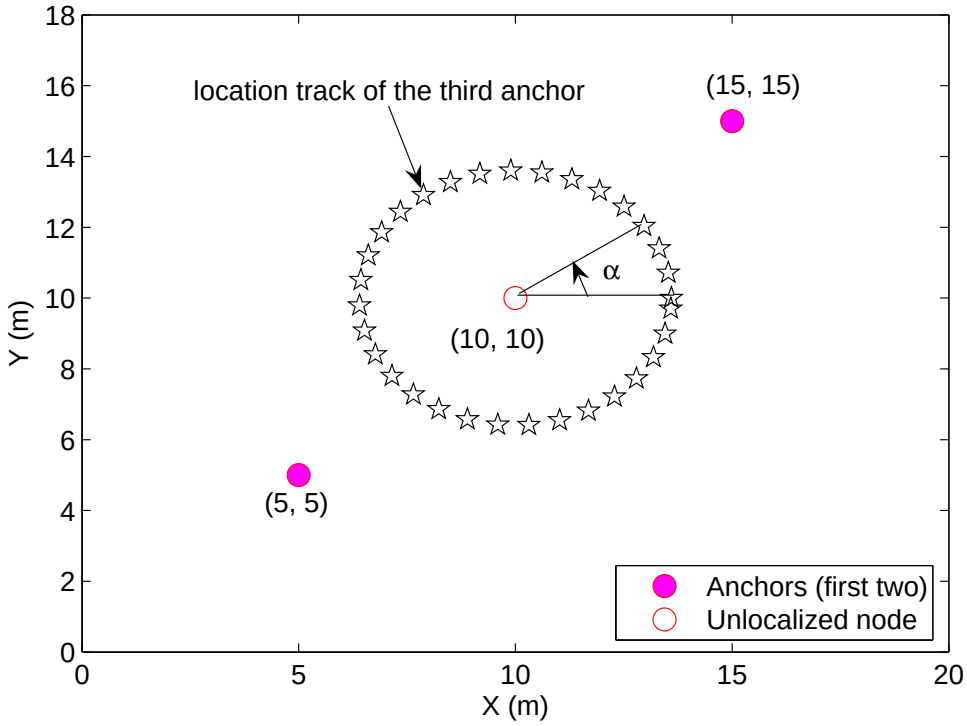


Figure 4.13: Node geometry to demonstrate the bias, MSE, estimator variance of the MLE. The pentagons represent the testing locations of the third anchor, starting from $\alpha = 0°$ to $360°$.

in [82] to empirically calculate the UCRLB for the MLE. Note that the UCRLB is a lower bound on the estimator *variance* of a biased estimator, while the CRLB is the lower bound on the variance of unbiased estimators, which is in fact equal to the MSE for unbiased estimators. In order to evaluate UCRLB, we need to empirically obtain the bias gradient matrix of the MLE solution. To do that, we simulate and obtain the biases of the MLE when the unlocalized node is placed at locations with an offset of 0.1 m from the nominal location (10,10) along X and Y axes. These empirical bias data are then used to approximate the 2×2 bias gradient matrix. Finally, we adopt the weighted squared Frobenius norm [82] as the measure of the bias matrix norm to calculate the UCRLB. In Fig. 4.13, we plot the node setup for the simulation. Specifically, we fix the locations of the first two anchors and the unlocalized node, while testing different locations of the third anchor, corresponding to different values of α from $0°$ to $360°$, as shown by the stars. This allows us to sweep through good and bad geometry conditions to see the effect that geometry plays.

In Fig. 4.14, we present the estimation biases of x and y with respect to α, obtained from

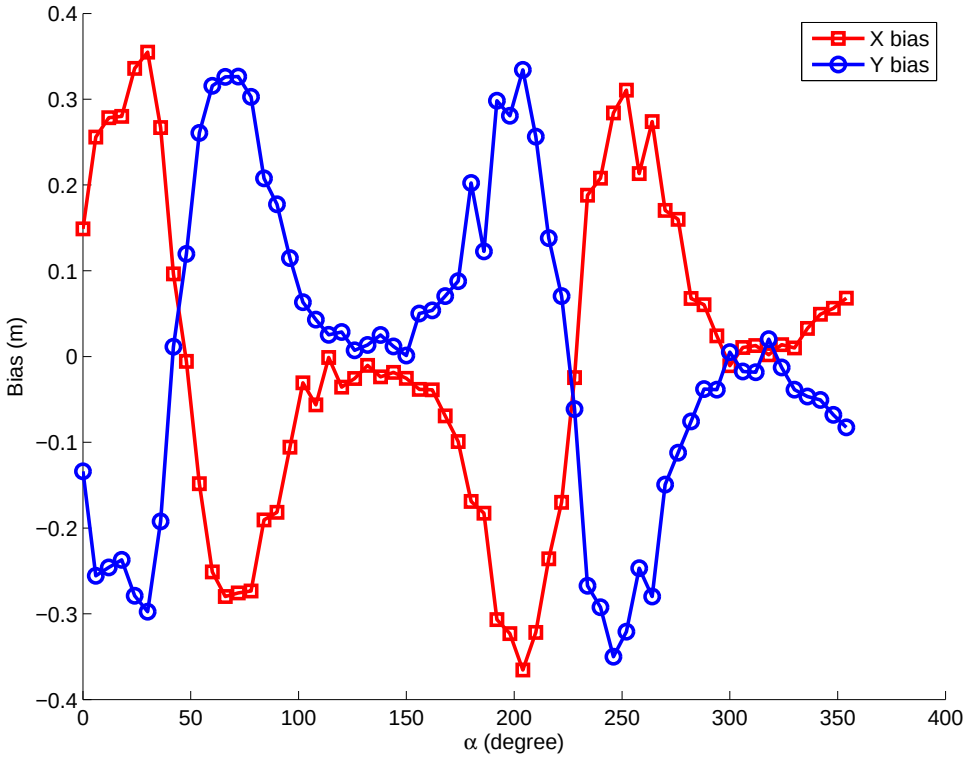


Figure 4.14: Biases on x and y of the MLE with respect to different testing locations for the third anchor, indexed by α, for the node setup in Fig. 4.13. It is clear that the MLE is a biased estimator.

1000 noise realizations. We can see that the TOA-based MLE in this case is clearly biased. In Fig. 4.15, we present the square root of the CRLB and the mean localization error of the MLE, which is essentially the root MSE, versus α. As can be seen, the mean localization error of the MLE is consistently lower than the square root of the CRLB, meaning that with the existence of estimator bias, the MLE can lead to an MSE smaller than the CRLB. Note that the CRLB becomes almost infinitely large when the three anchors and the unlocalized node are close to being collinear, *i.e.*, $\alpha = 45°$ and $225°$. Also included in Fig. 4.15 are the standard deviation (std) of the MLE and the square root of the UCRLB. We can see that the standard deviation of the MLE is always smaller than the root MSE, which is easy to understand since the MSE is essentially the sum of the estimator variance and the squared bias. In addition, we can see that the square root of the empirically-obtained UCRLB is indeed a lower bound on the standard deviation of the MLE. We finally emphasize that the empirical UCRLB obtained here can be made more accurate by running more noise realizations and allowing a step size smaller than 0.1 m. Nevertheless, our results properly demonstrate that

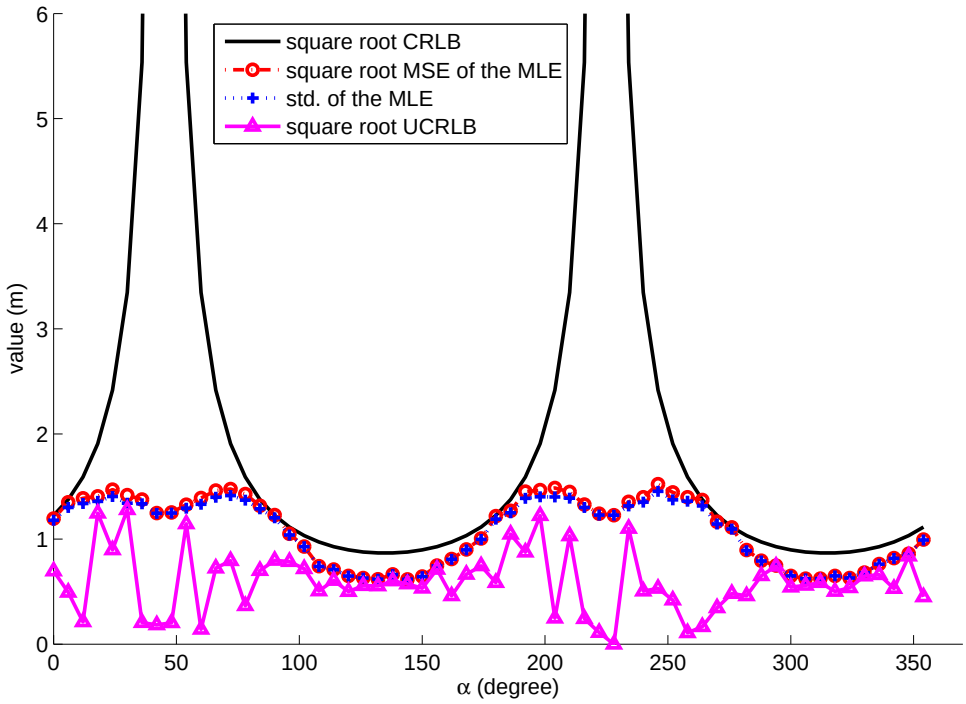


Figure 4.15: The comparison between the CRLB and the MSE of the MLE, as well as the comparison between the uniform CRLB (UCRLB) and the standard deviation of the MLE, with respect to α, for the node setup in Fig. 4.13.

the standard deviation of the biased MLE is lower bounded by the UCRLB. We also need to mention that the UCRLB is estimator-dependent since it requires the knowledge of the estimator bias norm, and thus cannot be conveniently computed or used as a *general* lower bound. Combining the results from Fig. 4.11 to Fig. 4.15, we conclude that in some node geometries, the CRLB is no longer a practically meaningful performance benchmark for evaluating localization performance. On the other hand, the optimal performance suggested by the MLE is a good performance benchmark, since it is robust to both range estimation noise and more importantly, node geometry.

4.4.4 MLE for collaborative position location

In Fig. 4.16, we present a result using the BB/RLT to solve the MLE for localizing a network of five unlocalized nodes using three anchors, for a particular network and noise realization with an equal noise variance assumption, *i.e.*, $\sigma_{ij}^2 = \sigma^2 = 1\,\mathrm{m}^2$. The solid (green) lines represent the connectivity and the dash-dot lines quantize the amount of localization error.

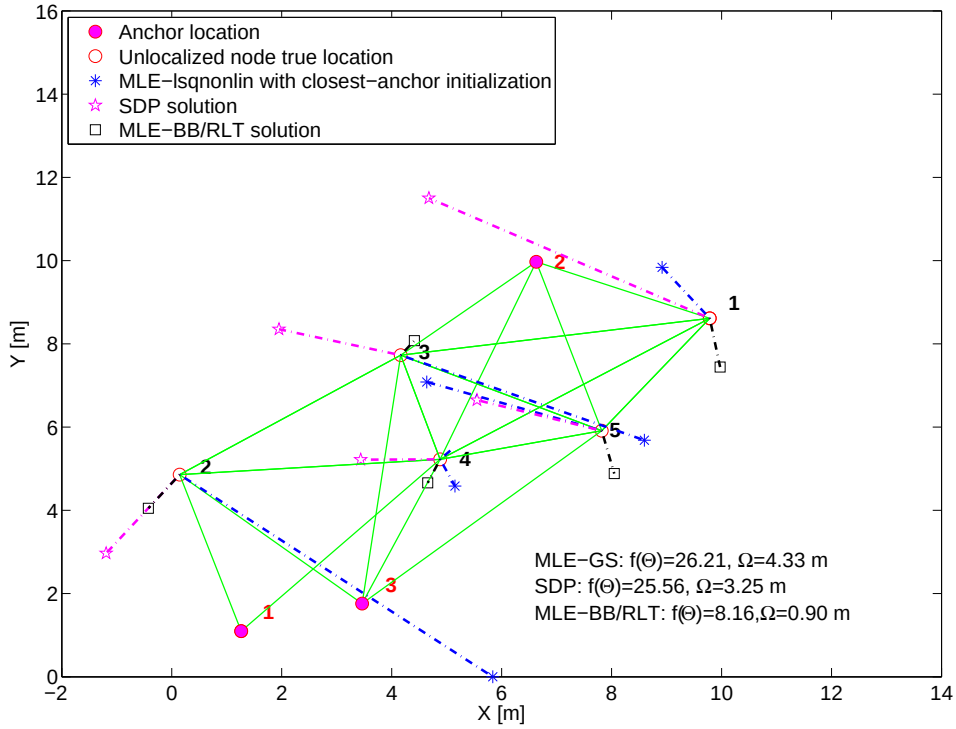


Figure 4.16: The MLE solution computed by the BB/RLT and the lsqnonlin solver in Matlab, as well as SDP solution, for a particular network and noise realization. The noise variance is $\sigma^2 = 1\,\mathrm{m}^2$

For comparison purposes, we also present the solution of the SDP approach [27] as well as the MLE solution computed using the classic nonlinear optimization method. For the later, we used the standard *lsqnonlin* function in Matlab to solve (4.2). Each unlocalized node has an initial solution set to the location of its closest anchor as long as it has one. If an unlocalized node does not have any connections to anchors, it uses the average of its neighboring un-localized nodes' initial solutions as its initial solution. For the SDP approach, we used the full SDP solver provided in [27]. As we can observe, the MLE solution computed by the BB/RLT algorithm has a final residual value of 8.16, smaller than the residuals of 25.56 for the SDP solution and 26.21 for the lsqnonlin solution. In fact, the network-average localization error of the BB/RLT solution is 0.9 m (averaged over 5 nodes), which is smaller than the 3.25 m of the solution using the SDP approach and 4.33 m of the solution obtained using the lsqnonlin. It is observed for this particular network geometry and range estimation noise realization, using the lsqnonlin solver to solve the MLE leads to the flip ambiguities which are essentially local instead of the global optima, while the BB/RLT algorithm is

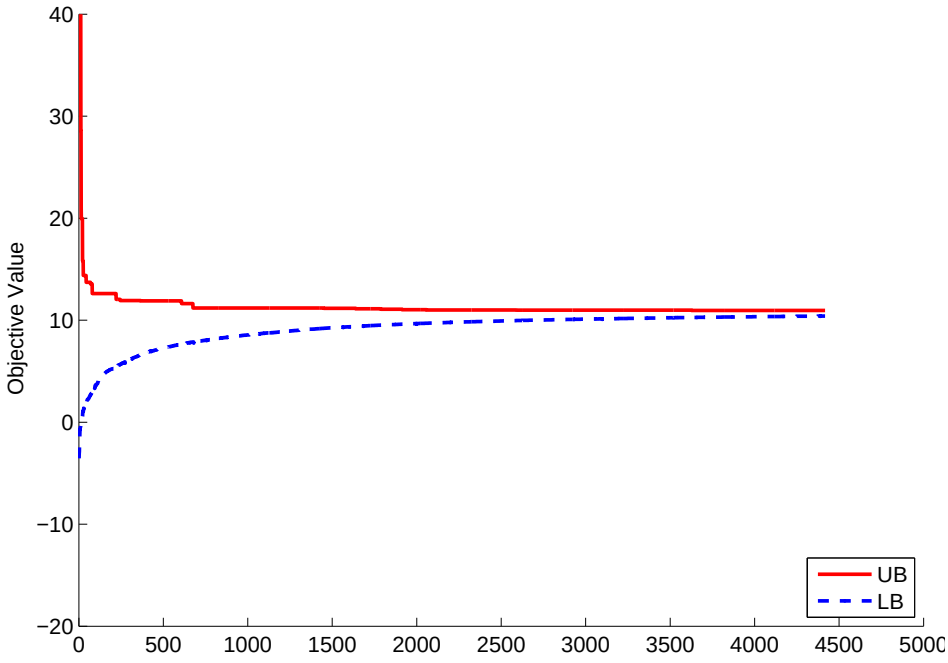


Figure 4.17: The convergence of the UB and LB on the MLE objective value for one noise realization for the network in Fig. 4.16. It takes much more iterations as well as running time for the BB/RLT to converge to the final solution.

guaranteed to find the global optimum to the MLE, leading to a much smaller localization error. In Fig. 4.17, we again plot the convergence of the UB and the LB for one particular noise realization. As can be seen, it takes more iterations for the algorithm to converge than the case in Fig. 4.4, due to a larger search space.

In Fig. 4.18, we plot the network-average localization error with respect to 50 different noise realizations with $\sigma_{ij}^2 = \sigma^2 = 1\,\mathrm{m}^2$ for the network in Fig. 4.16. The three additional horizontal lines represent the mean network-average localization errors, which are 3.16 m, 3.12 m and 2.48 m for the MLE using the lsqnonlin solver, the SDP and the MLE using the BB/RLT, respectively. It is seen that the optimal solution to the MLE obtained using the BB/RLT is the best and in fact can be used as a performance benchmark. We emphasize that the MLE using the BB/RLT does not necessarily outperform the SDP for *any* network and noise realization, since it aims to minimize the residual rather than the localization error. However, on average, the MLE does outperform the SDP and is in fact the theoretically optimal performance for any centralized or distributed position location scheme, as long as the underlying noise model is the same as what the MLE assumes. Finally, it is worth

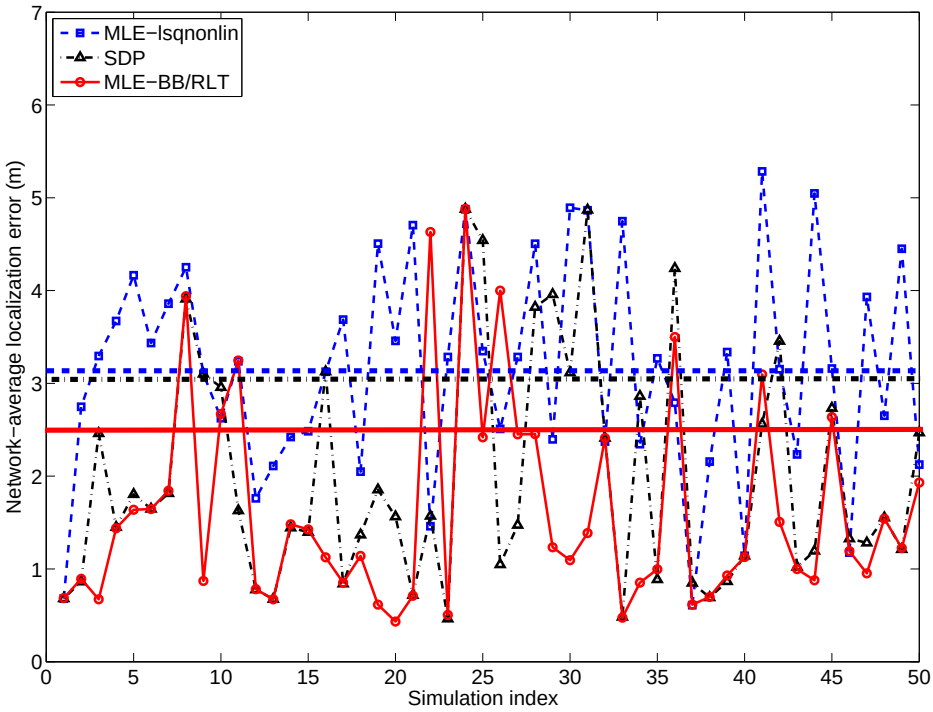


Figure 4.18: The network-average localization error with respect to 50 different noise realizations for the network in Fig. 4.16. The noise variance is $\sigma^2 = 1\,\mathrm{m}^2$. The three additional horizontal lines represent the mean network-average localization error for the three methods, respectively.

mentioning that the BB/RLT method has higher computational complexity than practical localization methods, due mainly to its thorough search over the solution space. For single node localization as in Figs. 4.11 and 4.12, the execution time is around 3 to 5 seconds on an Intel 3.3-GHz CPU, depending on the noise realization and the number of anchors. For the network shown in Fig. 4.16, the execution time increases to around 8 to 10 minutes, depending on the noise realization. For an even more challenging case, e.g., a large network with very high connectivity, the complexity increases exponentially. Therefore, our main focus has been on single node or small network localization. Nevertheless, our results demonstrate that using the MLE solution, obtained by the BB/RLT, rather than the CRLB as a performance benchmark is a meaningful way to lower bound more practical location estimation techniques.

In Fig. 4.19, we plot the mean localization error versus range estimation noise standard deviation. The standard deviation of ranging noise varied from 0.2 m to 1.0 m and the localization error is averaged over 100 noise realizations. We observe that the MLE solution

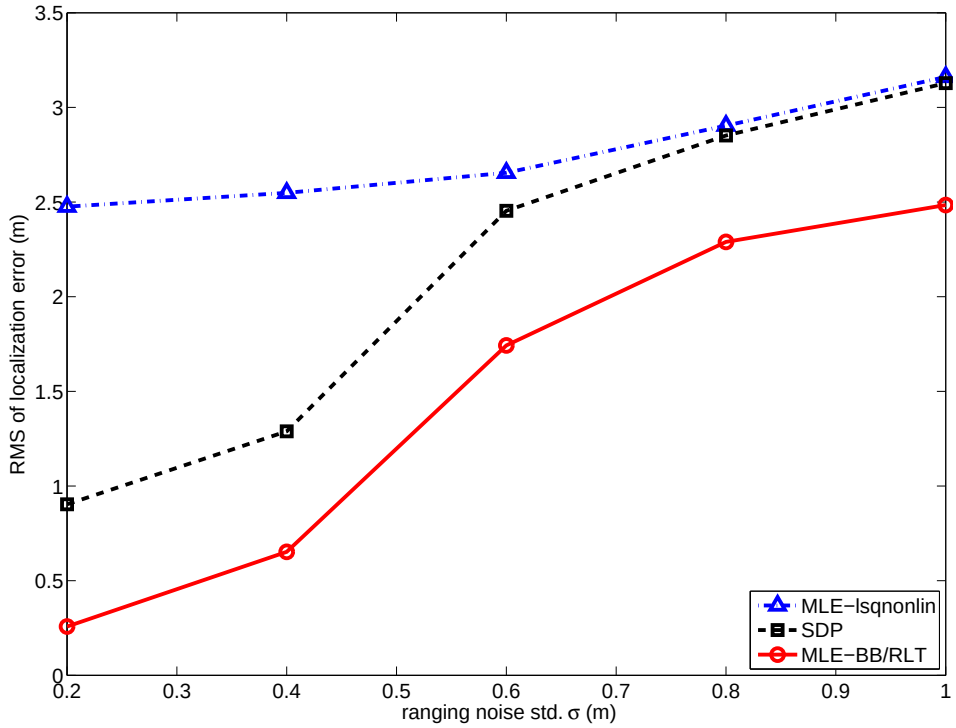


Figure 4.19: The mean localization error versus range estimation noise for the network in Fig. 4.16, averaged over 100 noise realizations. As can be seen, the optimal performance suggested by the MLE solution obtained by the BB/RLT framework serves as a performance benchmark. The SDP outperforms the MLE by the lsqnonlin for small ranging noise, but the performance difference diminishes as ranging noise increases.

obtained by the BB/RLT framework leads to the best performance compared to either the SDP or the MLE solved by the *lsqnonlin* solver. Therefore, the results in Fig. 4.19 corroborate that the performance indicated by the MLE solved by the BB/RLT framework indeed serves as a performance benchmark for practical collaborative position location methods. However, we need to emphasize that due to the possible exponential complexity increasing of the BB/RLT framework as the number of nodes and average node connectivity, the BB/RLT is more suitable for small and moderate size of networks. Nevertheless, our results still provide valuable insights into issues associated with the CRLB in the case of bad node geometry and the optimal performance of small size networks.

4.5 Summary

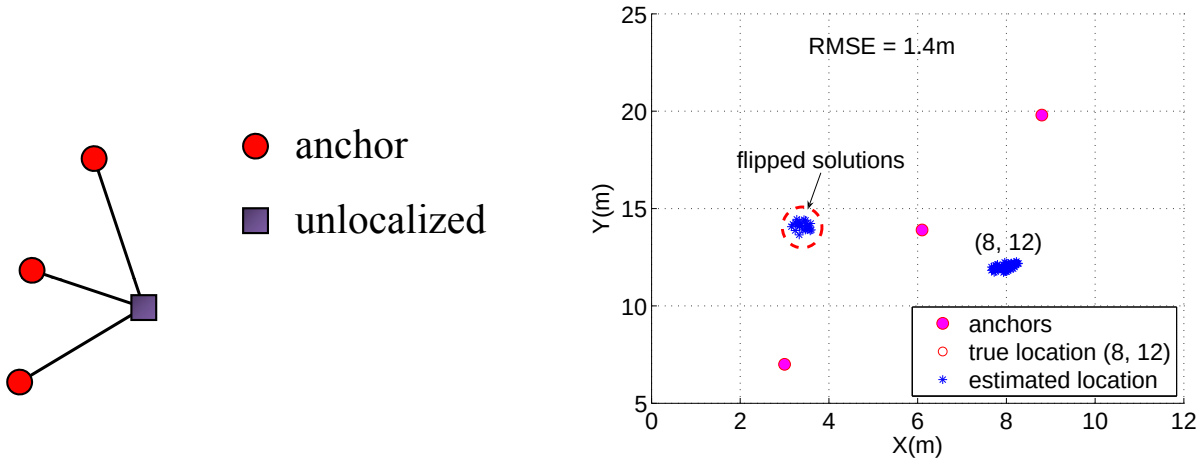
In this chapter, we have developed a branch-and-bound (BB) strategy, coupled with the reformulation linearization technique (RLT) in order to solve the maximum likelihood estimate (MLE) problem for collaborative position location. Unlike existing work which has only approximately solved the MLE for collaborative position location, our method guarantees the global optimality of the final solution. We further demonstrated that the TOA-based MLE is in general a biased estimator and that unlike the classic CRLB, the optimal performance implied by the MLE is robust to both range estimation noise and more importantly node geometry. Thus, the MLE can be used as a better performance benchmark than the CRLB to evaluate any practical position location scheme. Future work includes the development of the MLE in the presence of the non-line-of-sight (NLOS) bias and designing more computationally efficient suboptimal and ideally distributed collaborative position location algorithm, which can approach the MLE performance.

Chapter 5

A Collaborative Quasi-Linear Programming Approach

In Chapter 4, we investigated the optimal performance of collaborative position location in the presence of Gaussian distributed range estimation noise. However, the computational complexity of the BB/RLT formulation is still very high, especially for large and heavily connected networks. On the other hand, practical range estimates are usually corrupted not only by Gaussian ranging noise, but also bias resulting from NLOS propagation, clock synchronization error, etc. Therefore, it is very important to develop a practical collaborative position location algorithm that is computationally efficient and yields good localization accuracy in the presence of various measurement errors. In this chapter, we focus on developing a collaborative position location algorithm in NLOS propagation conditions. First, we demonstrate why the NLOS issue becomes important when nodes collaborate with each other. Then, we propose a collaborative quasi-linear programming (CQLP) framework to handle both node collaboration and NLOS measurements. The performance of this algorithm is compared to the standard sequential least-squares (LS) estimator in terms of both localization accuracy and location coverage. Note that in this chapter, we take the approach of sequential localization, mainly because of its superior computational efficiency and network scalability. However, as will be demonstrated later, there are two issues associated with the paradigm of sequential localization. First, it suffers significant performance degradation in large networks, due to the issue of error propagation when node geometry is bad, which is very likely to happen in randomly distributed networks. Second, despite the increased location coverage of our proposed CQLP framework, sequential localization may still not be

able to localize all the unlocalized nodes in the network. In the extreme case that no single unlocalized node has a sufficient number of anchors to communicate, none of unlocalized nodes can be localized.



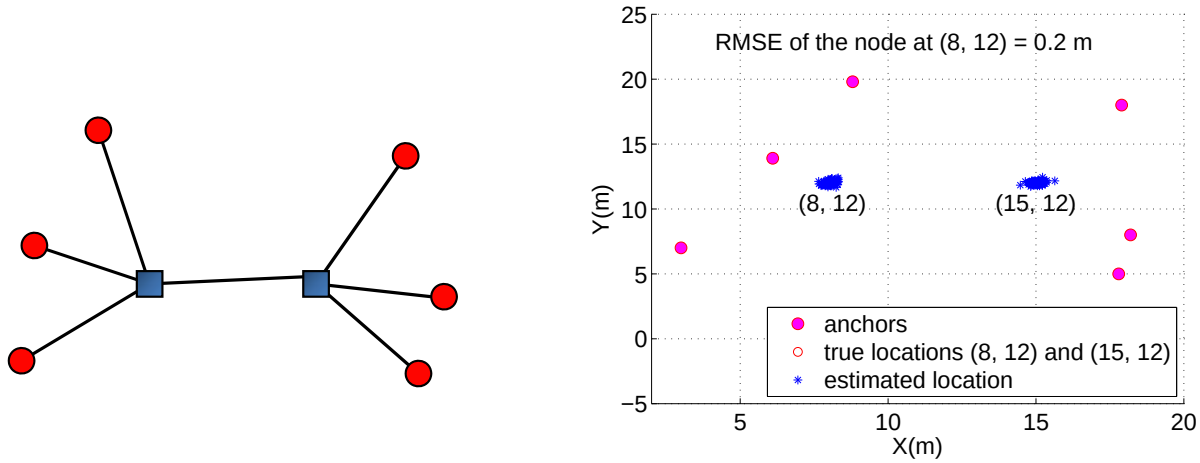
(a) Node setup

(b) Estimated locations for unlocalized node located at (8,12)

Figure 5.1: Illustration of the effect of node collaboration on the localization accuracy. Case 1: noncollaborative localization with only LOS range measurements.

5.1 Node collaboration with NLOS measurements

Collaboration among unlocalized nodes has been shown to reduce the CRLB on localization error as long as certain connectivity condition is satisfied [24]. However, the MLE does not always demonstrate this performance improvement, mainly because the numerical methods used to solve the nonlinear optimization problem associated with the MLE are often trapped into local minimum instead of the global minimum, especially when the number of variables is large. Even if we use the BB/RLT framework, in the presence of NLOS bias, the computed global minimum can still be erroneous location estimation. On the other hand, although the popular LS estimator has a closed-form expression and does not require an initial guess [70], it cannot admit node collaboration in a straightforward way. It is worth mentioning that centralized approaches [15, 19, 25, 30, 33] are inherently collaborative localization since the information of the entire network is utilized to collectively estimate all node locations.



(a) Node setup

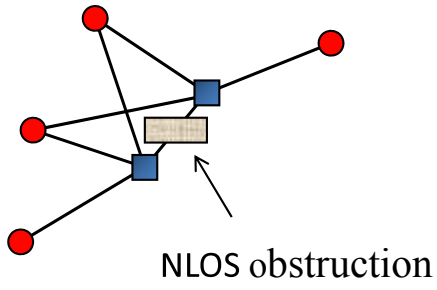
(b) Estimated locations for the two unlocalized nodes located at (8,12) and (15, 12), respectively

Figure 5.2: Illustration of the effect of node collaboration on the localization accuracy. Case 2: collaborative localization with only LOS range measurements.

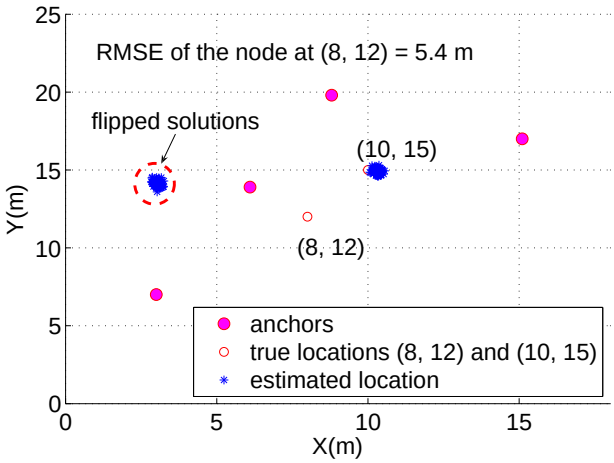
However, these algorithms have not considered node collaboration in the presence of NLOS measurements and how this impacts the localization accuracy. We now illustrate this issue via three cases of position location shown in Fig. 5.1-Fig. 5.3.

For the three networks shown in Figs. 5.1-5.3 (a), the MLE solved by *lsqnonlin* is applied to 200 noise realizations to estimate the location(s) of unlocalized node(s) based on noisy range measurements. The results are shown in Figs. 5.1-5.3 (b), respectively[1]. We assume range estimates are corrupted by zero-mean Gaussian distributed noise, except the one in Fig. 5.3 (a) noted as NLOS obstructed, which is further positively biased by 3 m. The MLE is numerically solved by *lsqnonlin* solver in Matlab with random initial solutions. The estimated locations for 200 noise realizations are plotted as blue asterisks. In particular, we calculate the mean localization error for the unlocalized node located at $(8, 12)$. In Fig. 5.1 (a), the unlocalized node simply computes its location using range estimates to the three anchors. Because the three anchors are almost (though not exactly) co-linear, location ambiguity,

[1]Note that we could use the BB/RLT framework developed in Chapter 4 to solve the MLE for all three networks. It can alleviate the flip ambiguity in Fig. 5.1 depending on the precision parameter ϵ. However, it will not eliminate the flip ambiguity in Fig. 5.3, since the flipped solutions indeed represent the global optimum of the MLE objective due to the presence of a large NLOS bias of 3 m.



(a) Node setup



(b) Estimated locations for the unlocalized nodes located at (8, 12) and (15, 12), respectively.

Figure 5.3: Illustration of the effect of node collaboration on the localization accuracy. Case 3: collaborative localization with LOS and NLOS measurements.

i.e., flip ambiguity, occurs shown by the fact that there are some erroneous solutions flipped about the straight-line approximately formed by connecting the three anchors, as shown in Fig. 5.1 (b). This results in a mean localization error of $\overline{\Omega} = 1.4$m. In Fig. 5.2-5.3 (a), one more unlocalized node, together with some anchors, is added and the two unlocalized nodes collaborate with each other to *collectively* compute their locations. As we can observe in Fig. 5.2 (b), flipped solutions for the original node have been completely eliminated and $\overline{\Omega}$ is reduced from 1.4 m to 0.2 m. This demonstrates that node collaboration in this case improves the localization accuracy. However, in Fig. 5.3 (b), because of the presence of an obstruction between the two unlocalized nodes and the resulting positively biased NLOS range estimate, erroneous flipped solutions will be favored over those around the true location, resulting in a much larger $\overline{\Omega} = 5.4$m. Compared to Fig. 5.2, this example clearly demonstrates that in realistic indoor environments where NLOS may occur, blindly using additional measurements among unlocalized nodes can degrade the performance of a localization algorithm depending upon many practical factors, such as node geometry, connectivity, the amount of measurement noise and NLOS bias, etc.

5.2 A Collaborative Quasi-Linear Programming Framework

In view of the above examples, we argue that node collaboration needs to be incorporated more cautiously, to ensure it benefits rather than hurts the localization accuracy. To achieve this, we extended the LP approach proposed in [48] and developed a collaborative quasi-linear programming (CQLP) framework to handle the degenerate cases based on the range scaling algorithm (RSA) proposed in [44], as well as to explore node collaboration by incorporating multihop-induced *artificial* NLOS range measurements.

5.2.1 System Models and Assumptions

We consider a 2D network of N unlocalized nodes and M anchors, where unlocalized nodes make range measurements to nodes within their communicaiton range and try to compute their locations. We follow the notations that have been defined in Chapter 2. To recall, the unlocalized nodes' locations are given by

$$\Theta = [\theta_1^T, \theta_2^T, \ldots, \theta_N^T],$$

and the anchors' locations are given by

$$A = [\theta_{N+1}^T, \theta_{N+2}^T, \ldots, \theta_{N+M}^T],$$

where $\theta_i = [x_i \; y_i]^T$ is the coordinate of the ith node, for $i = 1, 2, \ldots, N+M$ and $(\cdot)^T$ denotes matrix transpose. Furthermore, to emphazie the fact that collaborative localization uses measurements among unlocalized nodes, we divide the set of range measurements into two subsets, written as

$$\mathcal{X} = \mathcal{X}_A \cup \mathcal{X}_\Theta = \bigcup_{i=1}^{N} \left(\mathcal{X}_A^{(i)} \cup \mathcal{X}_\Theta^{(i)} \right) \tag{5.1}$$

where

$$\mathcal{X}_A^{(i)} = \left\{ r_{ij} \,|\, j = N+1, N+2, \ldots, N+M;\; j \in \mathcal{N}(i) \right\} \tag{5.2}$$

$$\mathcal{X}_\Theta^{(i)} = \left\{ r_{ij} \,|\, j = 1, 2, \ldots, N;\; j \in \mathcal{N}(i) \right\} \tag{5.3}$$

denote the set of range measurements between the ith unlocalized node and its neighboring anchors, and the set of range measurements between the ith unlocalized node and its neighboring unlocalized nodes, respectively.

To facilitate our later discussion of the LS estimator and the LP approach, we define

$$\mathcal{Q}_A^{(i)} = \left\{ j \,|\, j = N+1, N+2, \ldots, N+M;\; j \in \mathcal{N}(i) \right\} \tag{5.4}$$

to the set of the ith unlocalized node's connected anchors, and (5.2) can be arranged as

$$\mathcal{X}_A^{(i)} = \left\{ r_m^{(i)} \,|\, m = 1, 2, \ldots, |\mathcal{Q}_A^{(i)}| \right\} \tag{5.5}$$

where $r_m^{(i)}$ represents the ith unlocalized node's range estimate to its mth connected anchor and $|\cdot|$ is the cardinality.

5.2.2 LS estimator and LP approach

LS estimator: We first describe the LS estimator and the LP approach naturally follows. The two methods are basically non-collaborative and thus only the set of range measurements to anchors $\mathcal{X}_A$ are used. For the ith unlocalized node, each range estimate to an anchor defines a circle on which the estimated location $\hat{\theta}_i = [\hat{x}_i \; \hat{y}_i]^T$ must lie. Two such circles are given by

$$r_m^{(i)^2} = (\hat{x}_i - x_m^{(i)})^2 + (\hat{y}_i - y_m^{(i)})^2, \tag{5.6}$$

$$r_m^{(k)^2} = (\hat{x}_i - x_k^{(i)})^2 + (\hat{y}_i - y_k^{(i)})^2 \tag{5.7}$$

where $[x_m^{(i)} \; y_m^{(i)}]^T$ is the ith unlocalized node's mth anchor corresponding to the range estimate $r_m^{(i)}$. Subtracting (5.6) from (5.7), we obtain a *linear* equation which the estimated location $\hat{\theta}_i = [\hat{x}_i \; \hat{y}_i]^T$ must satisfy, given by

$$a_{mk}^{(i)} \hat{x}_i + b_{mk}^{(i)} \hat{y}_i = c_{mk}^{(i)}, \tag{5.8}$$

where

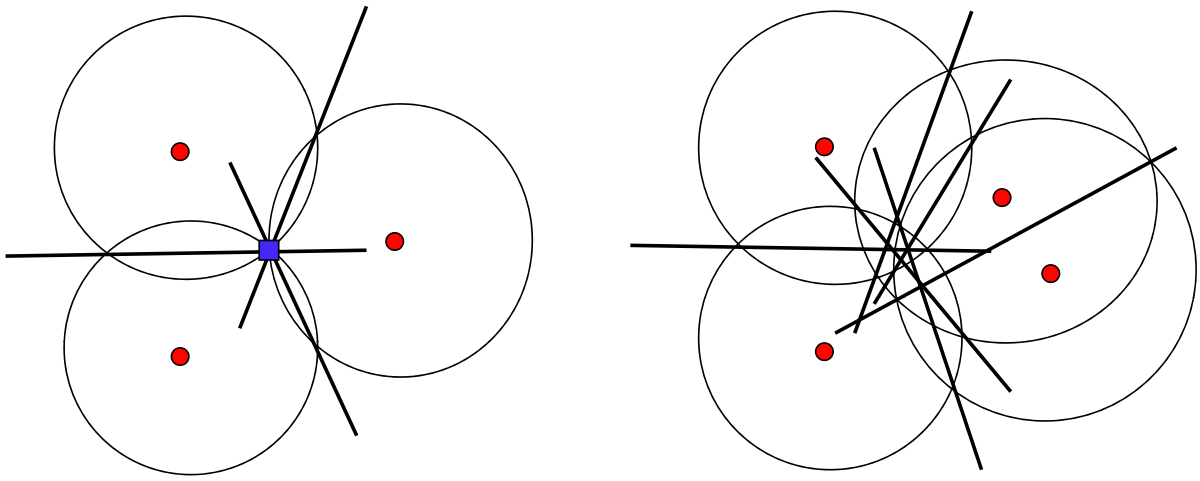
$$a_{mk}^{(i)} = x_m^{(i)} - x_k^{(i)}, \tag{5.9}$$

$$b_{mk}^{(i)} = y_m^{(i)} - y_k^{(i)}, \tag{5.10}$$

$$c_{mk}^{(i)} = \frac{1}{2}\left[(x_m^{(i)^2} - x_k^{(i)^2}) + (y_m^{(i)^2} - y_k^{(i)^2}) - (r_m^{(i)^2} - r_k^{(i)^2}) \right]. \tag{5.11}$$

If we denote the number of the ith unlocalized node's range estimates to anchors as $K_i = |\mathcal{X}_A^{(i)}|$, there will be $\mathrm{C}_2^{K_i} = \frac{1}{2}K_i(K_i - 1)$ such linear equations.

In the absence of any range estimation noise, these linear equations will intersect at the same point, which gives the unlocalized node's location, as shown in Fig. 5.4 (a). On the other



(a) Noise-free range estimates (only 3 range estimates)

(b) Noisy range estimates (4 range estimates)

Figure 5.4: Illustration of lines formed by subtracting circular equations defined by LOS range estimates.

hand, in the presence of range estimation noise and there are more than 3 range estimates, these lines will not intersect at the same point, shown by Fig. 5.4 (b), *i.e.*, the linear equations cannot be satisfied simultaneously. If we define a residual for each linear equation as

$$e_{mk}^{(i)} = a_{mk}^{(i)}\hat{x}_i + b_{mk}^{(i)}\hat{y}_i - c_{mk}^{(i)}, \tag{5.12}$$

by minimizing the sum of the squared residuals over all linear equations, *i.e.*,

$$\hat{\theta}_i^{\mathrm{LS}} = \arg\min_{\hat{\theta}_i} \sum_{m=1}^{|\mathcal{Q}_A^{(i)}|} \sum_{k=m+1}^{|\mathcal{Q}_A^{(i)}|} {e_{mk}^{(i)}}^2 \tag{5.13}$$

the closed-form linear LS solution is given by [70]

$$\hat{\theta}_i^{\mathrm{LS}} = (\mathbf{H}_i^T\mathbf{H}_i)^{-1}\mathbf{H}_i^T\mathbf{c}_i \tag{5.14}$$

where

$$\mathbf{H}_i = \begin{bmatrix} a_{12}^{(i)} & a_{13}^{(i)} & \dots & a_{1K_i}^{(i)} & a_{23}^{(i)} & \dots & a_{(K_i-1)K_i}^{(i)} \\ b_{12}^{(i)} & b_{13}^{(i)} & \dots & b_{1K_i}^{(i)} & b_{23}^{(i)} & \dots & b_{(K_i-1)K_i}^{(i)} \end{bmatrix}^T$$

$$\mathbf{c}_i = \left[\begin{array}{cccccccc} c_{12}^{(i)} & c_{13}^{(i)} & \dots & c_{1K_i}^{(i)} & c_{23}^{(i)} & \dots & c_{(K_i-1)K_i}^{(i)} \end{array} \right]^T$$

From the above, it is easy to see that $K_i \geq 3$ (at least 3 range estimates) is required to obtain the unlocalized node's location. Note that when $K_i = 3$, even if there is range estimation noise, the three lines still intersect at the same point since there are only two independent linear equations and the third one can be derived by subtracting the two. In general, for K_i range estimates, there are only $K_i - 1$ independent linear equations, although when solving (5.13), all lines have been used for a better measurement fitting.

LP Approach: It has been observed that the localization accuracy of the LS estimator in (5.14) may degrade when NLOS range estimates are present and blindly used [48]. In indoor environments, it is likely that the majority of the available range estimates are NLOS. Therefore, we may not have the luxury of simply discarding NLOS range estimates. The LP approach, by treating LOS and NLOS range estimates differently, provides an efficient way of utilizing NLOS range estimates without degrading the localization accuracy. Specifically, for the ith unlocalized node, we arrange its set of range estimates in such a way that the first K_i^L range estimates are LOS and the remaining K_i^N range estimates are NLOS, *i.e.*, $K_i = K_i^L + K_i^N$. Furthermore, we assume that an NLOS identification technique has been performed so that we know which range estimates are NLOS and which are LOS. In practice, this can be achieved by examining the received signal statistical characteristics, e.g., root-mean-square (RMS) delay [49]. With this knowledge, the LP approach only uses the first K_i^L LOS range estimates to define the objective function. Specifically, it linearizes the objective function by replacing the square fitting error, *i.e.*, ${e_{mk}^{(i)}}^2$, by the absolute fitting error[2], *i.e.*, $|e_{mk}^{(i)}|$, where

$$e_{mk}^{(i)} = {e_{mk}^{(i)}}^+ - {e_{mk}^{(i)}}^-, \tag{5.15}$$

$${e_{mk}^{(i)}}^+ \geq 0;\ {e_{mk}^{(i)}}^- \geq 0. \tag{5.16}$$

where ${e_{mk}^{(i)}}^+$ and ${e_{mk}^{(i)}}^-$ are two constrained variables. This will lead to the following linearized problem

$$\begin{aligned} \hat{\theta}_i^{\mathrm{LP}} &= \arg\min_{\hat{\theta}_i} \sum_{m=1}^{K_i^L} \sum_{k=m+1}^{K_i^L} \left({e_{mk}^{(i)}}^+ - {e_{mk}^{(i)}}^- \right), \qquad (5.17) \\ \text{s.t.:} \quad & {e_{mk}^{(i)}}^+ - {e_{mk}^{(i)}}^- = a_{mk}^{(i)} \hat{x}_i + b_{mk}^{(i)} \hat{y}_i - c_{mk}^{(i)} \\ & {e_{mk}^{(i)}}^+ \geq 0;\ {e_{mk}^{(i)}}^- \geq 0. \end{aligned}$$

[2]Numerically, this will less penalize large fitting errors.

Similar to the LS estimator, $K_i^L \geq 3$ (at least three LOS range estimates) is required to setup the above linearized problem.

On the other hand, the remaining K_i^N NLOS range estimates are used only to define additional linear constraints, based on the fact that the NLOS bias is typically much larger than the noise, resulting in positively biased range estimates. In particular, each NLOS range estimate defines a circle *within* which the unlocalized node must lie, *i.e.*,

$$(\hat{x}_i - x_n^{(i)})^2 + (\hat{y}_i - y_n^{(i)})^2 \leq {r_n^{(i)}}^2, \tag{5.18}$$

for $n = K_i^L + 1, \ldots, K_i$. We can relax the circular inequality in (5.18) into four rectangular inequalities, given by

$$\begin{aligned} \hat{x}_i - x_n^{(i)} &\leq r_n^{(i)}, \\ -\hat{x}_i + x_n^{(i)} &\leq r_n^{(i)} \\ \hat{y}_i - y_n^{(i)} &\leq r_n^{(i)}, \\ -\hat{y}_i + y_n^{(i)} &\leq r_n^{(i)}, \end{aligned} \tag{5.19}$$

where $[x_n^{(i)}\ y_n^{(i)}]^T$ is the anchor corresponding to the nth NLOS range estimate. The four linearized constraints in (5.19) can be combined with the linear objective function in (5.17) to form a complete linear program which can be solved by standard techniques. The complete LP is now given by [48]

$$\hat{\theta}_i^{\mathrm{LP}} = \arg\min_{\hat{\theta}_i} \sum_{m=1}^{K_i^L} \sum_{k=m+1}^{K_i^L} \left({e_{mk}^{(i)}}^{+} - {e_{mk}^{(i)}}^{-} \right), \tag{5.20}$$

$$\begin{aligned} \text{s.t.:} \quad & {e_{mk}^{(i)}}^{+} - {e_{mk}^{(i)}}^{-} = a_{mk}^{(i)} \hat{x}_i + b_{mk}^{(i)} \hat{y}_i - c_{mk}^{(i)} \\ & {e_{mk}^{(i)}}^{+} \geq 0;\ {e_{mk}^{(i)}}^{-} \geq 0 \\ & m = 1, 2, \ldots, K_i^L,\ k = m+1, m+2, \ldots, K_i^L \\ & \hat{x}_i - x_n^{(i)} \leq r_n^{(i)}, \\ & -\hat{x}_i + x_n^{(i)} \leq r_n^{(i)} \\ & \hat{y}_i - y_n^{(i)} \leq r_n^{(i)}, \\ & -\hat{y}_i + y_n^{(i)} \leq r_n^{(i)}, \\ & n = K_i^L + 1, K_i^L + 2, \ldots, K_i \end{aligned}$$

The above LP can be efficiently solved by many solvers available, such as the built-in LP solver *linprog* in Matlab, SeDuMi or Mosek optimization tools.

5.2.3 Handling the degenerate cases

As we have seen, the LP approach requires $K_i^L \geq 3$ to properly setup the linear objective function in (5.17). This is a more stringent requirement than what is required by the LS estimator (*i.e.*, $K_i \geq 3$), and thus it is very likely not to be satisfied in the applications of interest. In the following, we refer to the cases corresponding to $K_i^L < 3$ as *degnerate* cases and develop a scheme utilizing the range scaling algorithm (RSA) developed in [44] to handle the degenerate cases in the LP approach.

The RSA was originally proposed to deal with the NLOS problem when localizing mobile devices in a cellular system [44]. The basic idea is to view an NLOS positively-biased range estimate as a scaled version of the true distance, where the scaling factor is always larger than one. Then, three such NLOS range estimates can be used to setup a constrained nonlinear optimization problem whose outputs are the three estimated scaling factors. By multiplying the three scaling factors with the correpsonding NLOS range estimates, three LOS range estimates are generated and finally used to estimate the unlocalized node's location.

We now describe how our proposed scheme leverages the RSA to handle the degenerate cases. Assume the ith unlocalized node has $K_i^L \leq 2$ LOS range estimates and $K_i^N \geq 3 - K_i^L$ NLOS range estimates to anchors. For each of the K_i^L LOS range estimates, we produce an *artificial* NLOS range estimate by adding a uniformly-distributed bias, written as

$$r_m^{(i)'} = r_m^{(i)} + B_{\text{art}}, \quad 1 \leq i \leq K_i^L \tag{5.21}$$

where B_{art} is the bias to create artificial NLOS range estimates. Then, we can select (e.g., the smallest) $3 - K_i^L$ from K_i^N NLOS range estimates and combine those with the K_i^L artificial NLOS range estimates in (5.21). Those three NLOS range estimates can be used to set up the constrained nonlinear optimization as in [44], which finally outputs three scaling factors. We then select the scaling factors corresponding to the $3 - K_i^L$ NLOS range estimates and generate $3 - K_i^L$ LOS range estimates accordingly. Now, we have 3 LOS range estimates, including the original K_i^L LOS and $3 - K_i^L$ newly-generated LOS range estimates, and can apply the LP approach as described. Our simulation results suggest the efficacy of this scheme in handling the degenerate cases, as our proposed framework using this modified LP approach can provide better localization accuracy than the LS estimator. Note that due to the constrained nonlinear optimization involved when handling the degenerate cases, we term the overall framework as "quasi-linear". It is also possible to convert more NLOS into LOS range estimates. However, since the nonlinear optimization generally has higher computational complexity, we choose not to do so, but to use it only when it is necessary,

i.e., only in the degenerate cases.

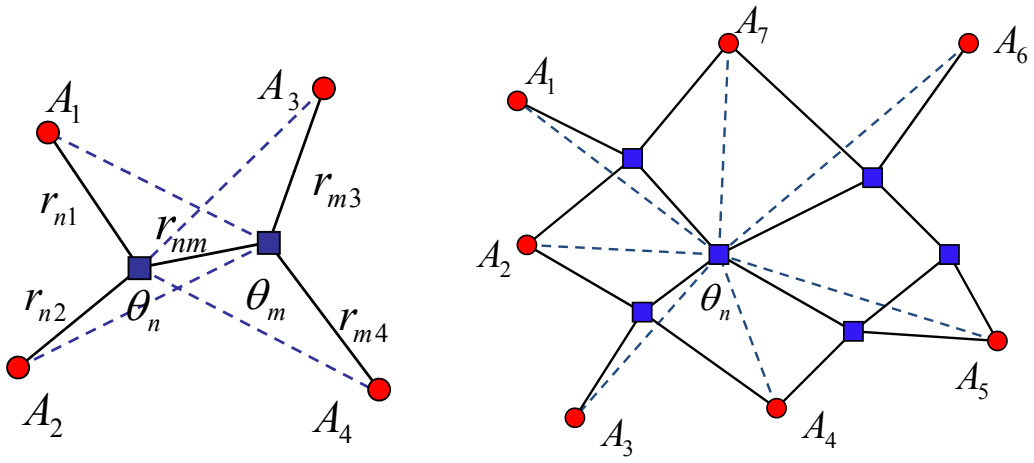
5.2.4 Sequential location estimation

As mentioned earlier, the LS estimator needs at least three range estimates to anchors in order to compute an unlocalized node's position unambiguously. The LP approach, on the other hand, needs at least three LOS range estimates to anchors in order to set up the linear objective function. However, in the context of WSNs, where there are few anchors, these conditions will rarely be satisfied. To localize more nodes and extend the location coverage, a *sequential* location estimation scheme [14] is adopted here. In this scheme, an unlocalized node immediately updates itself as a new *virtual* anchor to assist localizing other unlocalized nodes once its location has been computed. By doing this, the location information is *propagated* throughout the network. This paradigm, commonly referred to as sequential position location, appears to be a simple and practical way of extending location coverage, especially for large-area sensor deployment. However, it suffers from the propagation of the localization error and mitigating this adverse effect is not the main focus of this work, but dealt with in [69]. Nevertheless, we still apply the sequential localization to both the LS estimator, accordingly termed as the *sequential LS estimator*, and our proposed CQLP framework.

For simplicity, in the following, we will refer to the original anchors and the virtual anchors together as *localized* nodes.

5.2.5 Node collaboration

So far, the location estimation methods described above only use range estimates to localized nodes. In this section, we describe how our proposed framework can be extened to utilize range estimates to other unlocalized nodes, to improve the localization accuracy. The basic idea is to create artificial NLOS range estimates by adding multihop range estimates. This is illustrated in Fig. 5.5, where the circles and squares represent localized and unlocalized nodes, respectively. The existence of a solid line between two nodes indicates that they are within communication range of each other and can obtain a range estimate. In Fig. 5.5 (a), θ_n and θ_m are two unlocalized nodes. Each of them has two range estimates to localized nodes, and one range estimate to an unlocalized node. In a non-collaborative localization scheme, neither of them can be localized. However, in our proposed scheme, we utilize the



(a) Creating Virtual NLOS range estimates

(b) Geometrically diversified virtual NLOS range estimates

Figure 5.5: Illustration of creating virtual NLOS constraints over multihop connections. (a) For θ_m, adding range estimates r_{n1} and r_{nm} produces an artificial NLOS range estimate between anchor A_1 and θ_m, shown by the dashed line connecting the two, since it is very likely that the sum is greater than their true distance. The same mechanism is applied to generate other artificial NLOS range estimates between θ_m and other anchors, also for θ_n, shown by all the dashed lines. (b) Using the mechanism described in (a), θ_n is equipped with many artificial NLOS range estimates towards different directions, which has the potential of combating bad node geometry.

following triangular inequalities for node θ_n imposed by true inter-node distances as

$$\begin{aligned} d_{nm} + d_{m3} &\geq d_{n3}, \\ d_{nm} + d_{m4} &\geq d_{n4} \end{aligned} \tag{5.22}$$

This suggests that θ_n must lie *within* the circle centered at A_3 with a radius of $d_{nm} + d_{m3}$. Similarly, it must lie *within* the circle centered at A_4 with a radius of $d_{nm} + d_{m4}$. This resembles the role of NLOS range estimates in the LP approach. Specifically, we can view $r_{nm} + r_{m3}$ and $r_{nm} + r_{m4}$ as two *artificial* NLOS range estimates to two localized nodes A_3 and A_4, respectively. In practice, we can loosely use $r_{nm} + r_{m3}$ and $r_{nm} + r_{m4}$ instead of the true distances. Although this does not guarantee the validity of (5.22) in the presence of range estimation noise, it works with high probability and its efficacy can be seen from our

```
while # of localized nodes increases
  for n = 1 : N
     if  U_n  is not localized and its # of range estimates ≥ 3
           create virtual NLOS range estimates to multi-hop localized nodes;
           if  # of range estimates to localized nodes ≥ 3
               if  # of LOS range estimates to localized nodes ≥ 3
                    apply LP to obtain  Û_n , update U_n  as localized;
               else
                    apply the RSA to generate sufficient LOS range estimates;
                    apply LP to obtain  Û_n , update U_n  as localized;
               end
           end
     end
  end
end
```

Figure 5.6: The complete CQLP algorithm.

simulation results. Similar articifical NLOS range estimates can be created for node θ_m. The artificial NLOS range estimates are shown as dashed lines in Fig. 5.5 (a). All of these artificial NLOS range estimates can now be easily incorporated into the LP approach to assist node localization without degrading the localization accuracy. By doing this, unlocalized nodes utilize range estimates to each other and thus collaboratively determine their locations.

In addition, another benefit from our collaborative localization scheme is that the artificial NLOS range estimates tend to be geometrically diversified, as can be better shown in Fig. 5.5 (b) where the central unlocalized node has artificial NLOS range estimates from many directions. This benefits the localization in the sense that it can mitigate the effect of a bad

node geometry. It is to be noted that the above node collaboration can be extended to more than two hops. The more hops included, the more artificial NLOS range estimates that can be created, thus the more constraints we have for the LP approach and better localization accuracy is expected. However, the complexity associated with creating NLOS range estimates grows rapidly with the number of hops. In addition, as the number of hops increases, the looser the NLOS range estimate constraints will be. Therefore, in our scheme, we only consider two to three hops. It is also worth noting that if only one hop is considered, the CQLP is equivalent to the non-collaborative LP approach.

Combining all the components in each of the previous four subsections, the complete collaborative quasi-linear programming (CQLP) framework is as shown in Fig. 5.6. It should be noted that the major computational complexity with the CQLP framework lies in the nonlinear optimization involved in the RSA. Despite this, the proposed framework is a decentralized solution and does not require intense anchor deployment. The major component, *i.e.*, the LP approach, has lower complexity than other nonlinear processing in other algorithms.

5.3 Simulation results

In this section, we examine the performance of our proposed CQLP framework by comparing it with the sequential LS estimator. We also attempted to compare our scheme with the collaborative multilateration method proposed in [18]. However, we have observed that even for some basic collaborative subtree topologies, the initial location solution in the collaborative multilateration method suffers from the NLOS problem, thus resulting in an average localization error even larger than the sequential LS estimator. Therefore, we will compare our CQLP framework with the sequential LS estimator. Specifically, we present a comparison of two important metrics for node localization in ad hoc sensor networks: coverage (the percentage of the nodes which are localizable) and accuracy (the localization error). In addition, we investigate how the number of anchors and the number of unlocalized nodes affect the two metrics.

We consider a $25 \times 25\,\mathrm{m}^2$ area. Four anchors are fixed at the four corners. In this case, N unlocalized nodes and $M-4$ anchors, besides the four placed at the corners, are randomly distributed within the network. The radio communication range is set to be R, *i.e.*, two nodes can obtain a range estimate to each other if their distance is less than R. The range estimation noises and biases are assumed to be independent of each other. The path loss

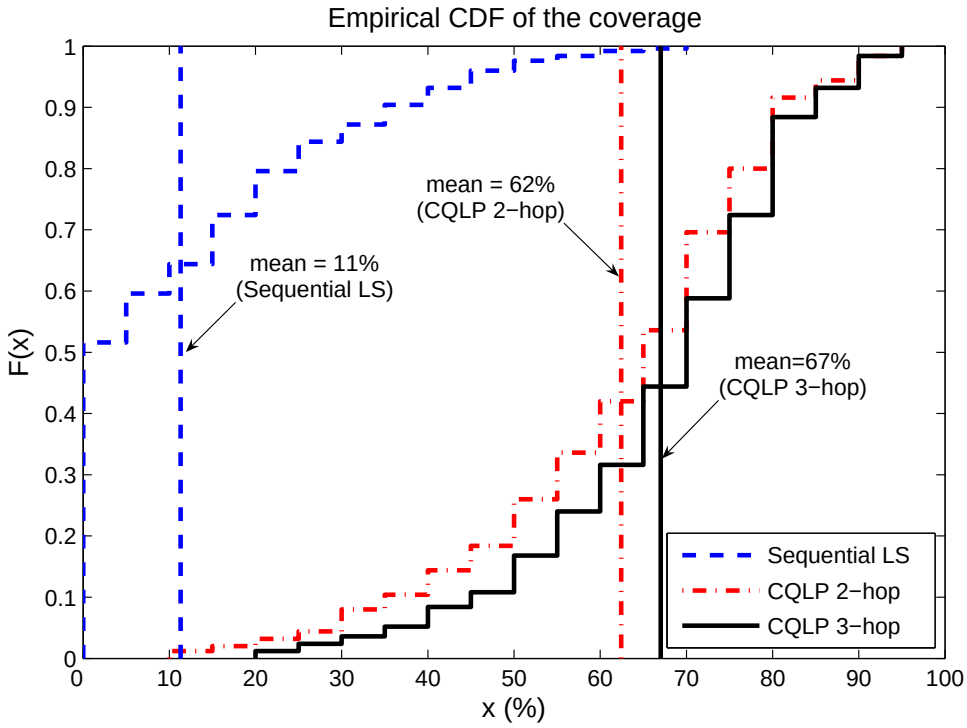


Figure 5.7: CDF of the coverage of the sequential LS and the CQLP framework for 250 network realizations, when $N = 20$, $M = 9$, $R = 6\,\text{m}$.

exponent $\beta = 2$ and the proportionality constant $K_E = 0.001$. The maximum NLOS bias[3] is chosen to be $B_{\max} = 0.5\,\text{m}$. The probability of a range estimate being NLOS is $p = 0.1$. When dealing with the degenerate cases, we have used $B_{\text{art}} = 2\,\text{m}$. With different values of N, M and R, we apply the sequential LS estimator and our proposed CQLP framework. The sequential LS estimator only utilizes range estimates to localized nodes to compute an unlocalized node's location, while the CQLP framework also utilizes range estimates to other unlocalized nodes by creating artificial NLOS connections to localized nodes that are originally more than one hop away. Both methods proceed until no additional nodes can be localized. For each set of parameters, we ran the simulation for different network realizations of unlocalized nodes, anchor's locations and NLOS indicators. For each network realization, we ran the simulation for 50 noise and bias realizations. The mean localization error is calculated according to (2.2) as described in Chapter 2.

[3]Note that the choice of $B_{\max} = 0.5\,\text{m}$ here is only to ensure $B_{\max}$ is much larger than the range estimation noise standard deviation. In later chapters for larger networks, we will use larger $B_{\max}$ when range estimation noise increases.

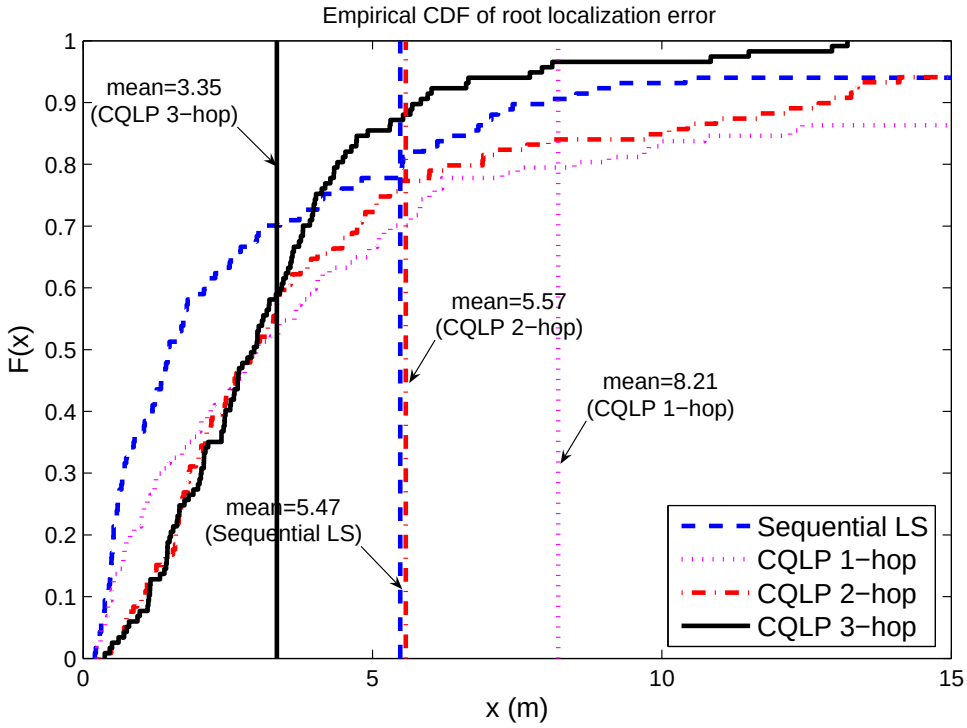


Figure 5.8: CDF of root localization errors of the sequential LS and the CQLP framework for 250 network realizations, when $N = 20$, $M = 9$, $R = 6\,\mathrm{m}$.

In Fig. 5.7, we compare the coverage (*i.e.*, the percentage of nodes that are able to be localized) of the proposed CQLP and the sequential LS estimator. The CDFs of the coverages over 250 network realizations of $N = 20$ unlocalized nodes and $M = 9$ anchors with $R = 6\,\mathrm{m}$ are plotted for both methods. As can be seen, the CQLP framework with both 2-hop and 3-hop node collaboration significantly increases the location coverage. For instance, the probability that the sequential LS estimator localizes less than 40% of the unlocalized nodes is about 0.9, while the probability of the CQLP framework with 2-hop and 3-hop node collaboration localizing less than 40% is about 0.05 and 0.1, respectively. Furthermore, the sequential LS estimator almost never localizes more than 70% of unlocalized nodes, while the CQLP framework achieves this coverage with a probabilty of greater than 0.55. The three vertical lines in Fig. 5.7 indicate the mean coverage based on 250 network realizations, which confirms that the proposed CQLP framework significantly increases the positioning coverage. Note that the 1-hop CQLP framework, *i.e.*, without node collaboration, results in the same coverage as the sequential LS estimator. It is also observed that the coverage improvement decreases as the number of hops increases.

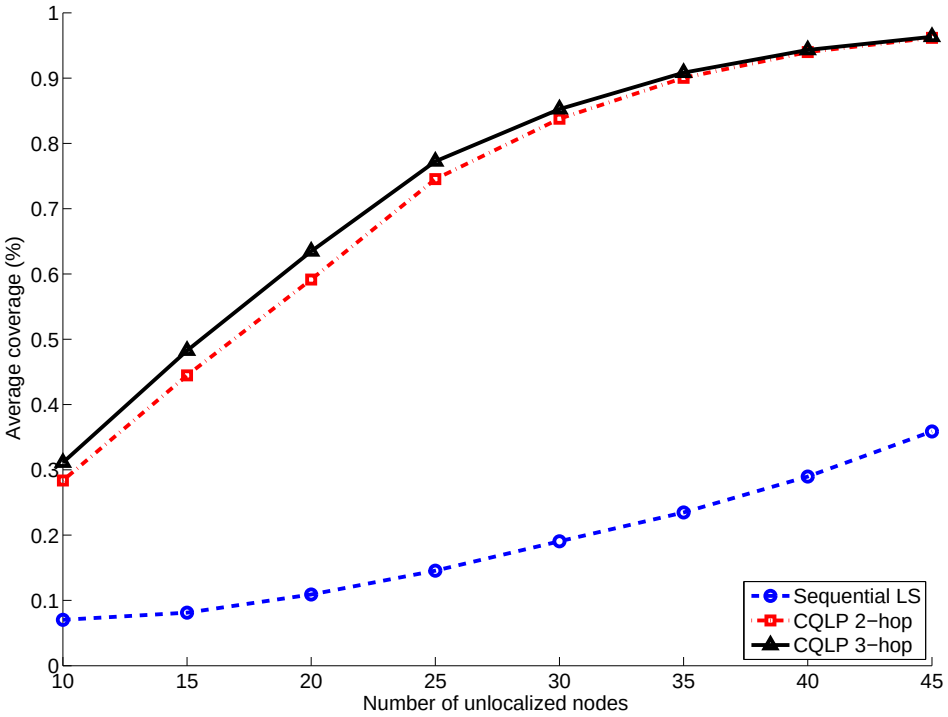


Figure 5.9: Average localization coverage (over 1000 network realizations) versus the number of unlocalized nodes, when the number of anchors is $M = 9$, the communication range $R = 6\,\mathrm{m}$. Unlocalized nodes are randomly distributed into the area.

In Fig. 5.8, we plot the CDF of the root localization error of the two methods over 250 network realizations. To make a fair and meaningful comparison, we compare only the nodes that both methods can localize. As we can see from Fig. 4, the 1-hop CQLP actually performs worse than the sequential LS. The reason is that the LOS range estimates corrected by applying the RSA method are still likely to be biased. Therefore, using these biased range estimates to construct the objective function in (5.17) degrades the localization accuracy of the LP approach, and finally leads to the worse performance. This negative effect is more obvious in an ad hoc network, since in most cases, an unlocalized node does not have enough localized nodes within its one-hop radio range to provide LOS range estimates, and thus will have to resort to the RSA to generate a sufficient number of LOS range estimates. However, the CQLP with 2-hop node collaboration performs comparable to the sequential LS, while the CQLP with 3-hop node collaboration performs better than the sequential LS, in terms of average localization error. This is mostly attributed to the suppression of large localization errors, which can be seen by the crossover of the CDF curves of the CQLP

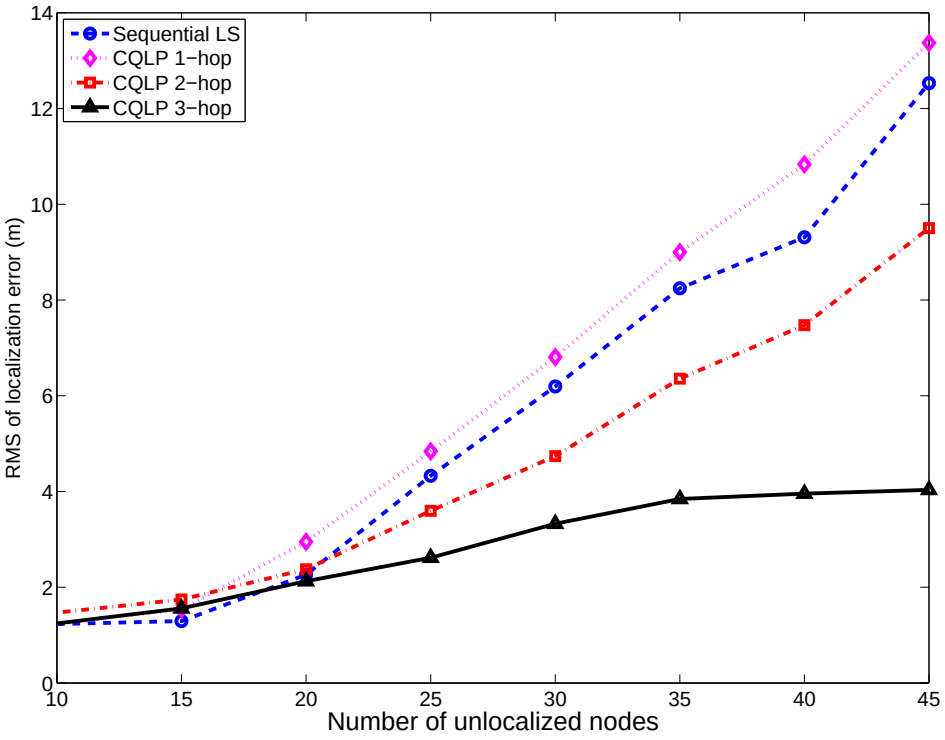


Figure 5.10: Average root localization error (over 1000 network realizations) versus the number of unlocalized nodes, when the number of anchors is $M = 9$ and the communication range $R = 6\,\text{m}$. Unlocalized nodes are randomly distributed into the area and each range estimate has a probability of 0.1 being NLOS, and if that happens, an NLOS bias uniformly distributed between 0 and 0.5 m is added onto the range estimate.

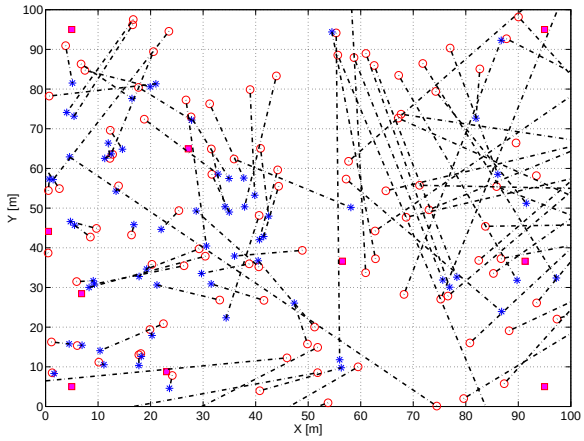
3-hop and the sequential LS. The performance improvement stems from the fact that 2-hop or 3-hop node collaboration adds more constraints by introducing more artificial NLOS range estimates. Together the improved localization coverage and the improved localization performance clearly demonstrate the benefit of node collaboration.

In Fig. 5.9, we compare the average localization coverage over 1000 network realizations for the two localization methods with different numbers of unlocalized nodes N. We observe that the CQLP with both 2-hop and 3-hop node collaboration significantly outperforms the sequential LS estimator by providing larger localization coverage. The more unlocalized nodes that are added into the network, the larger improvement we can obtain by using the CQLP framework. The improvement finally becomes smaller when the number of unlocalized nodes within the area is much larger, resulting in a much higher average node connectivity

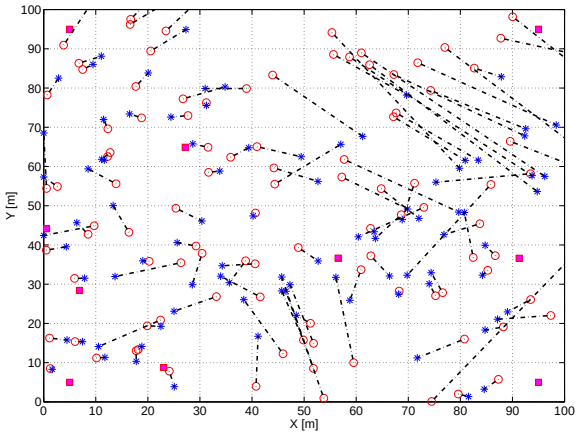
such that the sequential LS location estimator is able to localize most nodes. In addition, the coverage increment from 2-hop to 3-hop is not that significant, which coincides with our explanation that the more hops, the looser the constraints induced by the aritificial NLOS range estimates, therefore leading to diminishing gains.

In Fig. 5.10, we compare the mean localization error of the two methods over 1000 network realizations. We observe that the 1-hop CQLP performs slightly worse than the sequential LS estimator. The CQLP with 2-hop node collaboration performs worse than the sequential LS estimator when the node density is low (less than 20 unlocalized nodes), while it performs better than the sequential LS esimator when the node density is high (more than 20 unlocalized nodes). This is easy to understand since as the node density increases, unlocalized nodes have a better chance to have more range estimates to localized nodes. The CQLP with 3-hop node collaboration provides even better performance than the sequential LS estimator when the node density is high. In addition, as the node density increases, the improvement of the CQLP with 3-hop node collaboration over the sequential LS estimator is more prominent. However, the computational complexity associated with 3-hop node collaboration is also higher, and inceases quickly as the node density increases. Figs. 5.9 and 5.10 together suggest that our proposed CQLP provides significantly increased localization coverage and that it is desirable to incorporate 3-hop node collaboration to provide improvement in the localization accuracy. On the other hand, for a high node density network, if complexity is a concern, using 2-hop node collaboration is a reasonable choice, although its localization accuracy improvement over the sequential LS estimator is not as much as with 3-hop node collaboration.

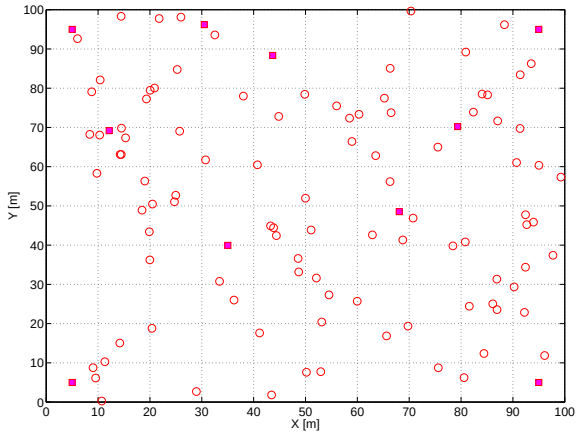
Issue of error propagation: When examining Fig. 5.10, we can see that the mean localization error generally increases as the number of nodes increases. We need to point out that for a fixed size network, *i.e.*, the edge length of the network remains the same, increasing the number of nodes within the area will increase the average node connectivity and ultimately improve the localization accuracy. However, for the envisioned application of our interest, the node density within the deployment area is usually the fixed parameter due to the cost constraint. This translates to the fact that each unlocalized node has only a limited number of neighbors and unlocalized nodes may be distributed over a large area. In such cases, the sequential localization can cause serious problem of error propagation. Note that there exist methods to combat this effect [68, 69] or one can use a carefully designed reference selection mechanism [32]. However, these methods have additional requirements and cannot fully eliminate the problem. To demonstrate the problem of error propagation, in Fig. 5.11



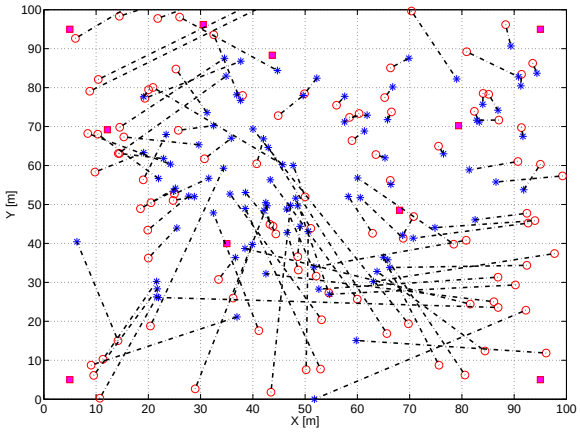
(a) Sequential LS estimator



(b) CQLP, 3-hop collaboration



(c) Sequential LS estimator: 0% coverage



(d) CQLP, 3-hop collaboration, full coverage

Figure 5.11: Results of applying the sequential LS and the CQLP with 3-hop node collaboration to localizing a large network with 100 unlocalized nodes (circles) and 10 anchors (squares), for one particular noise realization. The communication range is $R = 20\,\mathrm{m}$ and the average node connectivity is 11.8 for (a) and (b), 12.4 for (c) and (d). The estimated locations are represented by asterisks and the dashed lines quantize the localization error.

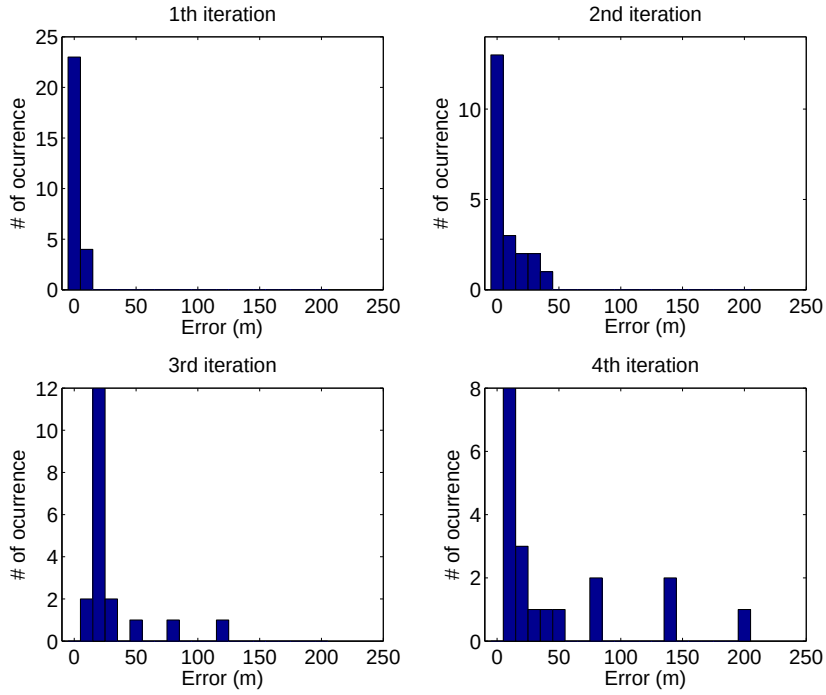


Figure 5.12: The histogram of localization errors of nodes being localized in the first four iterations in the sequential LS localization approach. As we can see, the nodes being localized in the first iteration have smaller errors, while those being localized in later iterations have progressively larger localization error.

(a) and (b), we plot the localization results of the sequential LS and the CQLP with 3-hop node collaboration for a large square network with a size of $100 \times 100\,\mathrm{m}^2$. The number of unlocalized nodes is 100, the number of anchors is 10 and they are randomly spread into the network. The range estimation noise is generated using a Gaussian distribution with $K_E = 0.01$ and $\beta = 2$ and we assume the pure LOS scenario. The anchor locations are shown by the squares and the true locations of unlocalized nodes are shown by the circles. The communication range is $R = 20\,\mathrm{m}$ and the average node connectivity is 11.8. The estimated locations are indicated by the asterisks and the dashed lines quantize the amount of localization error. We observe that the sequential LS is able to localize 98% of unlocalized nodes but suffers significant performance degradation. The CQLP with 3-hop node collaboration has better performance, but still has significant performance loss, especially for those on the upper right corner of the network which initially do not have sufficient anchors and

thus have to rely on sequential localization. In Fig. 5.12, we plot the histogram of the localization errors of nodes being localized at different iterations in the sequential LS approach. As we can see, the localization errors of nodes that are localized in the first iteration have smaller errors, while those that are localized at later iterations have progressively larger localization errors. In view of these, we believe the paradigm of sequential localization may cause serious problems of error propagation, especially for large area network deployment with low node connectivity.

Issue of location coverage: Although sequential localization is able to increase the location coverage, in the extreme case, where none of the unlocalized nodes has sufficient anchors to communicate, the sequential localization process cannot be initiated and thus the location coverage cannot be extended at all. Even if the CQLP, by the use of multihop virtual NLOS range estimates, is able to increase the location coverage, the resulting localization accuracy is highly questionable in such cases, due to the quality of the created virtual NLOS range estimates. As shown by the network in Fig. 5.11 (c) and (d), the sequential LS estimator cannot localize even a single node, while the CQLP, although being able to localize 100% nodes, results in very large localization error.

5.4 Summary

In this chapter, we have proposed a collaborative quasi-linear programming framework for the general network collaborative position location problem. Our technique incorporates collaboration among unlocalized nodes by creating artificial NLOS range estimates. In addition, the range scaling algorithm is adopted to deal with the degenerate cases where an unlocalized node has less than three LOS range estiamtes. Simulation results show that the proposed method provides both increased localization coverage and improved localization accuracy over sequential LS location estimator. We emphasize that even with sequential localization, the location coverage may still not be enough. In the extreme case, where there is no single unlocalized node having a sufficient number of anchors to communicate, none of the unlocalized nodes can locate itself and thus initiate the sequential localization process. We also demonstrated that both the CQLP and sequential LS estimator suffer from the problem of propagation of localization error when the node connecitivity is low. In view of these and also some existing work, we believe a concurrent position location approach is a better choice and that will be our approach in the next chapter.

Chapter 6

A Set-Theoretic Approach to Collaborative Position Location

6.1 Motivation and Related Work

In Chapter 5, we designed a CQLP framework for collaborative position location using the paradigm of sequential localization. Despite its superior computational efficiency and network scalability, the CQLP approach still suffers from issues such as error propagation and location coverage. Therefore, in this chapter, we resort to the concurrent approach, *i.e.*, using range estimates in a concurrent, instead of sequential, fashion. In fact, there have been many existing algorithms in this regard. For instance, the authors in [24] formulated the maximum likelihood estimator (MLE) for collaborative position location. In [15], the authors formulated the sensor network localization problem as a convex programming problem and inter-node measurements are modeled as convex feasibility sets. In [25] and [27], the authors proposed a semi-definite programming (SDP) approach to solve the sensor network localization problem. It has also been demonstrated in [26] that using the SDP solution as a starting point for a gradient based search can further refine the SDP solution. To address the computation efficiency issues associated with the SDP for large networks, the authors in [30] proposed a second-order cone programming (SOCP) relaxation approach, while the authors in [62] resorted to the sum of squares (SOS) relaxation method. Another notable approach is the multi-dimensional scaling (MDS) method followed by an optional least-squares (LS) refinement developed in [20]. It is worth mentioning that these methods are based on centralized computation and their distributed versions, e.g., distributed SDP [26], distributed

MDS [20]-[22] and distributed SOCP [31], have also been developed. On the other hand, some other distributed algorithms have been proposed, e.g., DV (distance vector) based positioning [16], robust positioning algorithm [17] and N-hop multilateration [18] and some more recent developments, e.g., robust quadrilaterals [28], iterative least-squares (ILS) with error propagation control [32] and map stitching [35].

In general, centralized methods use global information and tend to have better localization accuracy than distributed methods. Among them, the full SDP solver provided in [27] appears to work very well for different network topologies, even with very few anchors. However, its computational complexity is approximately $O(N^3)$, which becomes prohibitively high as the number of nodes and connectivity increase. Among distributed methods, MDS-MAP(P, R) in [20] using range estimates followed by global LS refinement yields good localization accuracy with acceptable computation time, but suffers a performance degradation in the presence of irregular network topologies. Another distributed method worth mentioning is iterative LS with error control described in [32], which demonstrated comparable and sometimes better localization accuracy than SDP and MDS. The more recent distributed SOCP [31] yields localization accuracy slightly worse than MDS-MAP(P, R) with less computational complexity. However, most existing works only consider a small amount of range estimation noise, e.g., 5% in [20] or up to 25% in [26]. In practice, range estimation noise can be higher, especially for received signal strength (RSS) based range estimates. Furthermore, none of the aforementioned methods has considered the presence of NLOS bias, which is a predominant phenomenon in indoor or urban environments. Therefore, there is a need to develop a computationally efficient and distributed localization method which has good performance in the presence of both high range estimation noise and more importantly, NLOS bias, regardless of network topology.

In addition to localization performance, an important aspect of sensor network localization is node localizability, i.e., whether a node has a unique location solution, especially in the presence of measurement noise. In [52] and later in [66], the authors used the theory of graph rigidity to determine if a network of nodes is uniquely localizable in the noise-free case. Also, in [65], the authors resorted to the SDP to solve flip ambiguity, again in the absence of range estimation noise. On the other hand, probabilistic localization methods, e.g., [53, 56], return a particle-based discrete approximation to the spatial distribution of node location. In doing so, the quality of the location estimate can be easily visualized and node localizability can be inferred, e.g., from the modality of the distribution. However, these methods have much higher computational complexity and do not scale well for large

networks. While we acknowledge that particle-based representation of the location solution does provide insight into solution quality, we believe there exist numerically simpler ways to achieve that goal.

In this chapter, with special emphasis on these two aspects, we develop a computationally efficient and distributed method for large-scale distance-based sensor network localization. The proposed method borrows the concept behind the parallel projection method (PPM), originally developed for signal recovery from inconsistent convex feasibility sets [86, 87], and further revises and extends it to an iterative and distributed numerical framework to estimate sensor nodes' locations. The proposed framework yields performance that is generally better than existing methods, especially in non-homogeneous networks. In addition, benefiting from the huge complexity savings, our algorithm can be executed using multiple distinct initializations. The obtained multiple solutions provide a natural approximation to the spatial distribution of node location estimate, similar to probabilistic localization methods, but achieved with significantly less computational load. We define a novel metric, called the *solution radius*, to capture the effect of solution ambiguity and show that it allows us to evaluate node localizability in practical situations.

6.2 Problem Formulation

To recall, consider a 2D square network with an area size of $(\mathcal{L} \times \mathcal{L})\,\mathrm{m}^2$, consisting of $M \geq 3$ anchors at known locations and n unlocalized nodes whose locations are to be estimated. The true locations of the unlocalized nodes are denoted by $\Theta = [\theta_1, \theta_2, \ldots, \theta_N]$ and the known locations of the anchors are denoted by $\mathbf{A} = [\theta_{N+1}, \theta_{N+2}, \ldots, \theta_{N+M}]$, where $\theta_i = [x_i \; y_i]^T$ is the 2D coordinate of the ith node, for $i = 1, 2, \ldots, N+M$ and $(\cdot)^T$ is the matrix transpose operation. If the distance between the ith and jth nodes, denoted by d_{ij}, is less than the physical communication range R, we say the two nodes neighbor each other, and thus can communicate and obtain a (noisy) range estimate r_{ij} of their true distance d_{ij}. Further, we assume range estimates are symmetric, i.e., $r_{ij} = r_{ji}$, $\forall i, j$. The task is to obtain an estimate $\hat{\Theta}$ of the true locations Θ. The model for range estimates has been described by (2.3) in Chapter 2.

Based on this model, the MLE of collaborative position location has been formulated in [24] for the pure-LOS scenario. The MLE is equivalent to finding the set of node locations that minimizes the sum of the weighted squared-difference between the estimated and the mea-

sured inter-node distances. However, there is generally no closed-form expression for the MLE solution due to the nonlinear and nonconvex nature of the associated optimization problem. Instead, several alternatives have been developed. For example, semi-definite programming (SDP) relaxation [25], or second-order cone programming (SOCP) relaxation [30] to the MLE have been proposed. In Chapter 4 and in [61], we applied the branch-and-bound/reformulation linearization techniques (BB/RLT) to find the solution to the MLE with guaranteed global optimality. However, the computational complexity associated with these centralized approaches increases quickly as network size and node connectivity increase. On the other hand, distributed methods based on virtual anchors, i.e., nodes act as anchors once they obtain location estimates, suffer from the problem of error propagation which needs special attention, e.g., via neighbor selection [32]. To compare different localization methods, the *mean* localization error $\overline{\Omega}$ is defined as the RMS value of the localization error, as in (2.2).

6.3 The Iterative Parallel Projection Method

In this section, we first briefly describe the parallel projection method (PPM) and its application to non-collaborative position location. Then, we elaborate on how we developed an iterative approach based on the PPM to solve collaborative position location and describe our approach to investigating node localizability in the presence of distance estimation noise.

6.3.1 PPM and non-collaborative position location

The set-theoretic signal recovery problem is to find a signal which satisfies all feasibility sets defined by prior knowledge and the observed data [86]. The mathematical formulation of the convex set feasibility signal recovery problem can be written as

$$\text{Find} \quad a^* \in \bigcap_{k=1}^{K} S_k \tag{6.1}$$

where S_k is the kth feasibility set and is assumed to be closed and convex. Methods such as projection onto convex sets (POCS), which sequentially projects an initial estimate onto each convex set until convergence, have been developed (see [87] for details) to solve (6.1).

In practice, due to inaccurate prior knowledge and noisy observations, it is possible that the feasibility sets are inconsistent, meaning that there exist one or more conflicting feasibility

sets such that $\bigcap_{k=1}^{K} S_k = \emptyset$. In such cases, the convergence behavior of POCS methods is not known and the formulation in (6.1) is no longer reasonable. Alternatively, since there exists no solution satisfying all convex feasibility sets, the feasibility problem can now be formulated as a weighted least-squares (LS) problem [87]

$$\begin{aligned} \min_{a} : \Phi(a) &= \sum_{k=1}^{K} w_k \, d_k(a)^2, \\ \text{s.t.: } \sum_{k=1}^{K} w_k &= 1, \text{ and } \ \forall k, w_k > 0, \end{aligned} \tag{6.2}$$

where $\Phi(a)$ is the proximity function which measures the infeasibility of the solution a, and $\{w_k\}$ are a set of strictly convex weights. $d_k(a)$ is the distance from a to its projection onto the kth convex set S_k. As can be seen, the objective now is to find a best feasible solution in the sense that it minimizes the sum of weighted square distances to all the feasibility sets. In [87], the PPM has been proposed to solve (6.2). In addition, by reformulating the problem of (6.2) in a product space, the author proved that the PPM in the original space corresponds to the alternating projection method in the product space, which leads to the desired LS solution to (6.2). In particular, if we denote the set of solutions to (6) as G, it is proved in [87] that as long as one of the convex sets S_k is bounded, for any initial estimate a^0, the sequence of a^l given by

$$a^{l+1} = a^l + \lambda_l \left(\sum_{k=1}^{K} w_k P_k(a^l) - a^l \right), \quad \text{for} \quad l \geq 0, \tag{6.3}$$

will converge to a point in G, where $\lambda^l \in [\epsilon, 2-\epsilon]$ for $0 < \epsilon < 1$ is called the relaxation parameter and $P_k(a^l)$ denotes the projection of a^l onto the kth convex set S_k, defined by

$$P_k(a^l) \in S_k, \forall b \in S_k, ||a^l - P_k(a^l)|| \leq ||a^l - b|| \tag{6.4}$$

where $||\cdot||$ denotes the norm. Note that when $a^l \in S_k$, its projection $P_k(a^l) = a^l$. Also, the choice of the relaxation parameter λ_l can affect the convergence speed of the PPM and there exist optimal values of λ^l [87]. In this work, we simply select $\lambda^l = 1$ to save the computation involved in obtaining an optimal value of λ^l in each iteration. In fact, choosing $\lambda^l = 1$ is called the unrelaxed form of (6.3), rewritten below as

$$a^{l+1} = \sum_{k=1}^{K} w_k P_k(a^l), \quad \text{for} \quad l \geq 0, \tag{6.5}$$

Fig. 6.1 gives an illustration of the unrelaxed PPM using (6.5), where a^{l+1} is simply the

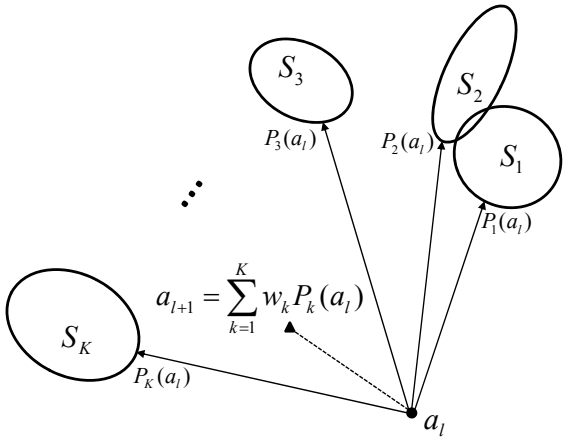


Figure 6.1: Generic unrelaxed parallel projection method, where a^{l+1} is simply the sum of a^l's weighted projections onto all convex feasibility sets, denoted by S_i's. $P_k(a_l)$ denotes the projection of a_l onto the kth convex set. $\{w_k\}$ is a set of convex weights.

sum of a^l's weighted projections onto all convex feasibility sets. This process continues until $\Phi(a^l) - \Phi(a^{l+1}) \leq \delta$, where $\delta > 0$ is the precision parameter.

It can be shown that the above formulation is naturally suited for the non-collaborative position location problem. Consider an unlocalized node making range estimates to a few surrounding anchors with known locations. Each range estimate defines a ranging circle that the unlocalized node should lie on. However, if we relax the constraint that the unlocalized node should lie *on* the circle to a constraint that the unlocalized node should lie *within* the circle (which essentially transforms the original non-convex constraint into a convex constraint), we can directly apply the PPM and term this as the *original* PPM. The corresponding projection, the distance function and the proximity function are given by

$$P_k^{\text{org}}(\hat{\theta}) = \begin{cases} A_k + r_k(\hat{\theta} - A_k)/||\hat{\theta} - A_k||, & \text{if } ||\hat{\theta} - A_k|| > r_k \\ \hat{\theta}, & \text{if } ||\hat{\theta} - A_k|| \leq r_k \end{cases} \tag{6.6}$$

$$d_k^{\text{org}}(\hat{\theta}) = \begin{cases} (r_k - ||\hat{\theta} - A_k||)^2, & \text{if } ||\hat{\theta} - A_k|| > r_k \\ 0, & \text{if } ||\hat{\theta} - A_k|| \leq r_k \end{cases} \tag{6.7}$$

$$\Phi^{\text{org}}(\hat{\theta}) = \frac{1}{K} \sum_{k=1}^{K} d_k^{\text{org}}(\hat{\theta}) \tag{6.8}$$

where $\hat{\theta}$ is the unlocalized node's current estimated location, A_k represents its kth connecting anchor and r_k denotes the range estimate between the unlocalized node and A_k. K is the number of anchors and $w_k = 1/K$ for $k = 1, 2, \ldots, K$ represents the fact that all range

estimates are treated equally. We can assign different weights to range estimates according to our confidence in their reliability.

In our work, however, we modify (6.6) - (6.8) into the following,

$$P_k^{\text{ncl}}(\hat{\theta}) = A_k + r_k \frac{\hat{\theta} - A_k}{||\hat{\theta} - A_k||} \tag{6.9}$$

$$d_k^{\text{ncl}}(\hat{\theta}) = (r_k - ||\hat{\theta} - A_k||)^2 \tag{6.10}$$

$$\Phi^{\text{ncl}}(\hat{\theta}) = \frac{1}{K} \sum_{k=1}^{K} d_k^{\text{ncl}}(\hat{\theta}), \tag{6.11}$$

We can easily see that if θ is outside of the ranging circle defined by r_k with A_k as the center, (6.9) is coincident with (6.6). If θ is inside the ranging circle, (6.9) differs from (6.6) in the sense that the projection will be *pushed* away from the center onto the ranging circle. Also note that the difference between (6.11) and (6.8) is that, in the original PPM, when the current solution is inside the kth ranging circle, the kth term in (6.8) becomes zeros, while the kth term in (6.11) of the modified PPM is not zero. Essentially, the objective of (6.11) is the sum of the weighted squared difference between range estimates and the distances calculated using the estimated location, which in fact the same as the objective in the MLE. The major difference here is the approach to solve the problem. In the standard nonlinear LS solver, such as *lsqnonlin*, the trust-region method or the Levenberg-Marquardt method is used. The PPM approach is merely another numerical method to solve the nonlinear LS problem, e.g., as in [88]. The main motivation for us to use it lies in the fact that the projection process can be easily modified to fit our application, especially for position location in NLOS conditions, as will be shown in Chapter 7. Additionally, the method is computationally very simple. Finally, we emphasize that with this modification, the theoretical convergence result for the PPM may not hold. Nevertheless, we argue, and will later demonstrate, that the modified PPM is better aligned with the objective of the MLE, which is known to be optimal in the MSE sense. The modified PPM for non-collaborative position location is given as follows.

Algorithm 6.1: Modified PPM for non-collaborative position location:

1. Initialize the unlocalized node at $\hat{\theta}$, set $l = 0$, $\Phi_l = \Phi^{\text{ncl}}(\hat{\theta})$ and δ a small positive number;
2. Update $\hat{\theta} \leftarrow (1/K) \sum_{k=1}^{K} P_k^{\text{ncl}}(\hat{\theta})$ and let $\Phi_{l+1} = \Phi^{\text{ncl}}(\hat{\theta})$;
3. If $|\Phi_{l+1} - \Phi_l| > \delta$, let $l = l + 1$ and go to 2);
4. Terminate and return $\hat{\theta}$.

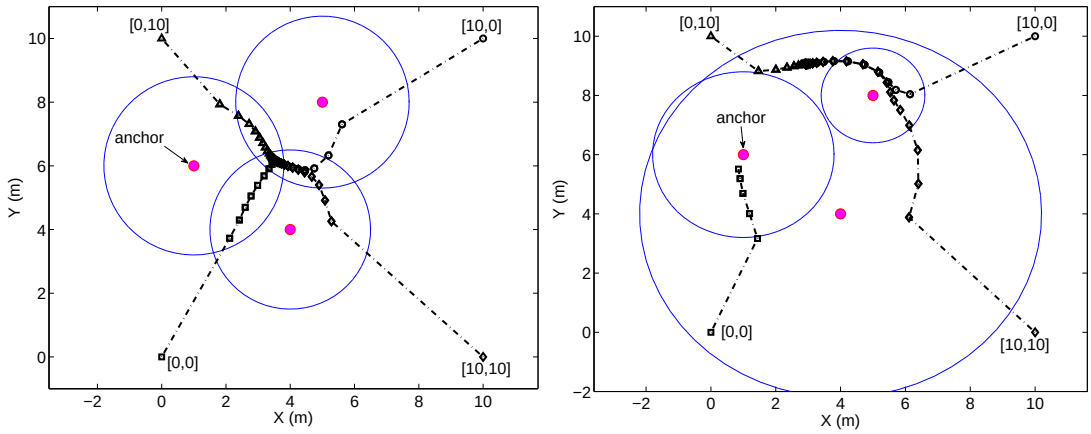


Figure 6.2: Application of the modified PPM to non-collaborative position location for two cases: (a) less noisy range estimates where the intersection region of the three range estimates contains the true location; (b) one range estimate is extremely noisy. The blue circles denotes ranging circle and the dash dot lines represent the trajectories along which the solution of modified PPM converges. The initial guesses are denoted by the corresponding values noted $[xy]$. In (a), different initial guesses lead to the same solution, while in (b) different initial guesses result in different final solutions.

We emphasize that due to the modified projection in (6.9), the final solution in many cases will depend on the initial guess. Fig. 6.2 gives two examples of applying the modified PPM to position location where an unlocalized node estimates its location using range estimates to three anchors, with 4 different initial solutions at [0, 0], [0, 10], [10, 0] and [10, 10] respectively. The three anchors are located at [1, 6], [4, 4] and [5, 8] respectively, while the unlocalized node is at [3.5, 6]. Fig. 6.2 (a) represents the case where the three range estimates are less noisy and have an intersection region containing the true location, while in Fig. 6.2 (b) one of the three range estimates is extremely noisy and is much larger. In either case, the modified PPM is able to converge to a final solution after a few iterations. As we can see, in the second case the final solution depends on the initial guess. Note that we do not see this effect in the first case when the range estimates are less noisy. We believe this phenomenon indeed contains information regarding the reliability of the solution, i.e., whether a location solution is trustworthy, and will explore it to address the issue of node localizability later.

Now we present some results to demonstrate why the modified PPM is more suited, than the original PPM with convex feasibility sets, for the position location problem. In particular, we show that the modified PPM is better aligned with the objective of the MLE for position

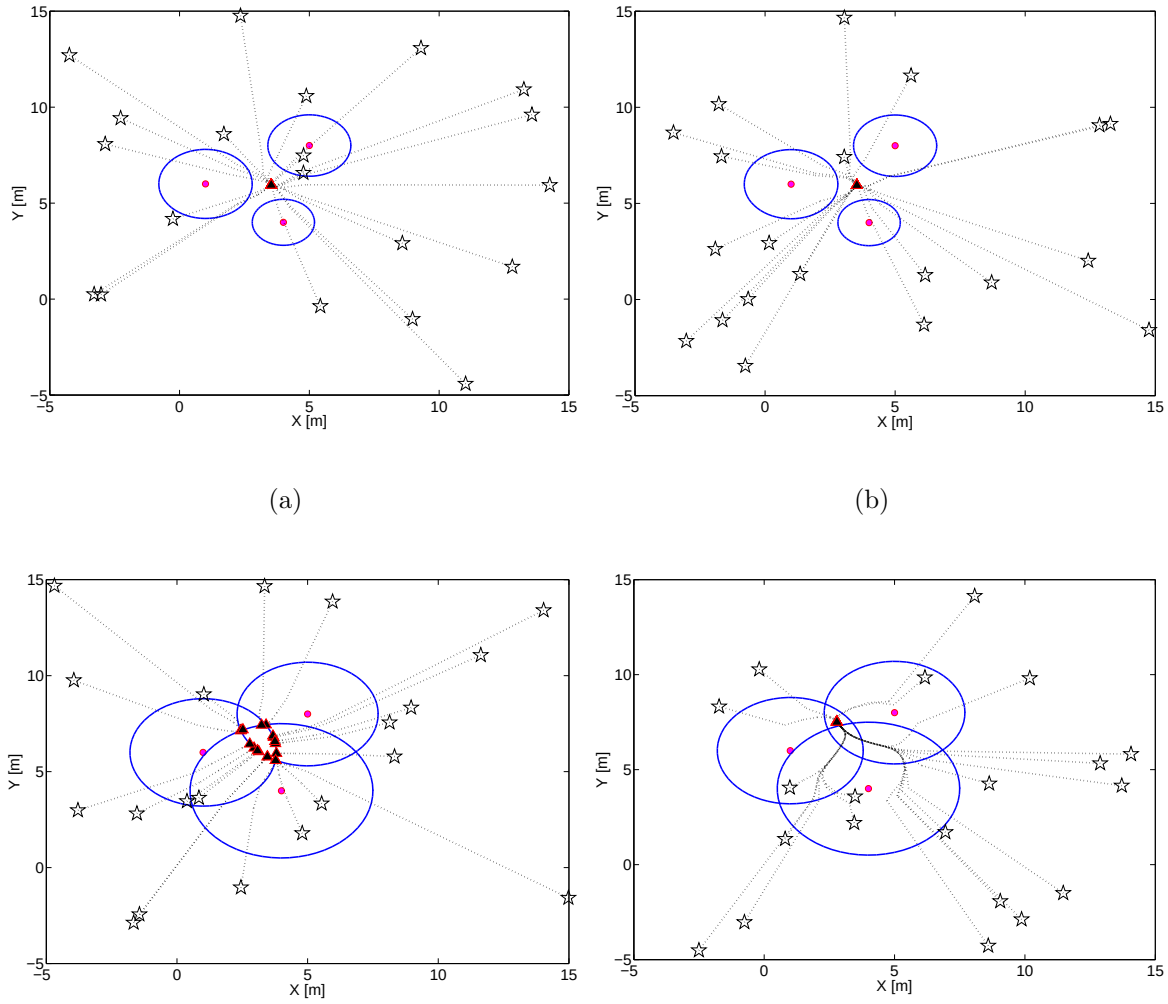
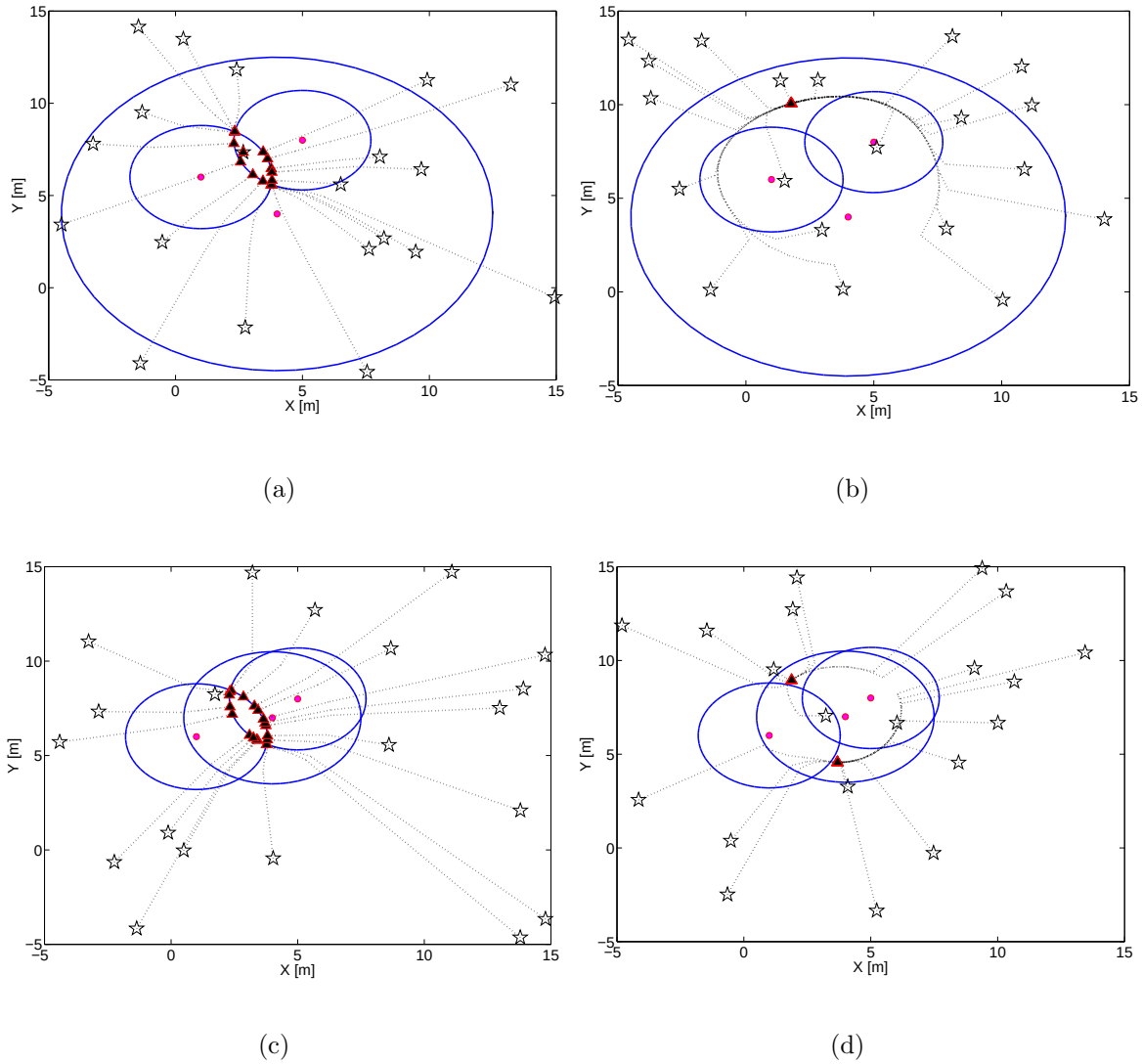


Figure 6.3: Comparison between original and modified PPM with the application to two cases of non-collaborative position location. Top row: 3 ranging circles do not intersect each other. Bottom row: 3 ranging circles intersect each other. (a) and (c): the results of original PPM. (b) and (d): the results of modified PPM. Solid lines represent rangine circles and circles are anchor locations. Stars represent different initial solutions and dashed lines represent the trajectories following which the PPM solution converges. Triangles represent final solutions.



(a) (b)

(c) (d)

Figure 6.4: Comparison between original and modified PPM with the application to another two cases of non-collaborative position location. Top row: one ranging circle is extremely large. Bottom row: 3 anchors are almost collinear. (a) and (c): the results of original PPM. (b) and (d): the results of modified PPM. Solid lines represent rangine circles and circles are anchor locations. Stars represent different initial solutions and dashed lines represent the trajectories following which the PPM solution converges. Triangles represent final solutions.

location. In Figs. 6.3 and 6.4, we give four examples of non-collaborative position location where both the original and modified PPM are applied, where the unlocalized node's true locations are (3, 6) and (2.5, 7.5) in Figs. 6.3 and 6.4, respectively. The solid circles represent anchor locations and the solid lines denote ranging circles. Stars represent different initial solutions and the dashed lines represent the trajectories along which the PPM solution converges. Triangles represent the final solutions. In the first example shown in Fig. 6.3 (a) and (b), both the original and modified PPM consistently converge to the same solution, which indeed is the global minimizer of the MLE objective in (4.2). However, in the second example shown in Fig. 6.3 (c) and (d), the original PPM approach converges to different solutions depending on the initialization, which, despite all being global minimizers of (6.8) leading to an objective value of zero, do not lead to the global minimum of (4.2). The modified PPM, on the other hand, is able to find the solution that minimizes the objective of (4.2). This is due to the fact that the objective of the modified PPM (6.11) is inherently the same as that of (4.2). Similar observations can be made in the first example shown in Fig. 6.4 (a) and (b), where one ranging circle is extremely large such that none of the solutions found by the original PPM, despite all being global minimizers of (6.8), is not the global minimizer of (4.2). The modified PPM, on the other hand, is still able to find the solution that minimizes the MLE objective (4.2). In the second example shown in Fig. 6.4 (c) and (d), we give an example where the modified PPM converge to two possible solutions depending on the initialization, where the solution at (1.9, 8.9) gives an objective of 0.78, while the solution at (3.7, 4.6) gives an objective of 2.11 which is in fact a local minimum. Overall, we argue that, despite the fact that it may converge to local minimum, the modified PPM is better aligned with the objective of the MLE and is computationally very efficient.

6.3.2 Iterative PPM for collaborative position location

We now focus on the more general problem of collaborative position location described earlier. In particular, we use the modified PPM as a basic element and extend it to an iterative and distributed numerical framework. The overall framework involves an initialization step and an iterative update step, where only local communications are necessary in both steps.

In the initialization step, each unlocalized node obtains an initial solution for its location. The approach we adopt is called closest-anchor initialization. Specifically, if an unlocalized node has connecting anchor(s), it will use the location of its closest anchor, determined from range estimates, as its initial solution. If an unlocalized node does not have a connection

to any anchors, it will use the average location of its surrounding nodes' initial solution as its initial solution. For isolated nodes without any connection, we simply use the network center as the initial solutions and due to the absence of range estimates, their estimates will thus not be updated in the ensuing iterative update step. The initialization step continues until every unlocalized node obtains an initial solution. We emphasize that there exist other methods to obtain initial solutions [18], including using the solution from the MDS as an initial solution, but those generally involve substantially more computation. Instead, we use the simple initialization method described above to illustrate the effectiveness of iterative PPM technique.

In the iterative update step, each unlocalized node uses modified PPM to update its location estimate, based on its range estimates to neighboring nodes, either anchors or other unlocalized nodes, and its current location estimate. In particular, if the ith and the jth nodes are neighbors, the projection of $\hat{\theta}_i$ onto the feasibility set given by the range estimate r_{ij} is

$$P_{ij}^{\text{col}}(\hat{\theta}_i) = \hat{\theta}_j + r_{ij}\frac{\hat{\theta}_i - \hat{\theta}_j}{||\hat{\theta}_i - \hat{\theta}_j||}. \tag{6.12}$$

Since only local information exchange is needed, the iterative update process can be distributed and the computational complexity scales linearly with network size. Each unlocalized node examines whether or not its residual changes over the previous iteration. The ith unlocalized node's residual based on its and its neighbors' current estimated locations is defined as

$$\Phi^{\text{col}}(\hat{\theta}_i) = \frac{1}{|\mathcal{N}(i)|}\sum_{j\in\mathcal{N}(i)}\left(r_{ij} - ||\hat{\theta}_i - \hat{\theta}_j||\right)^2, \tag{6.13}$$

where $\mathcal{N}(i)$ is the set of the ith node's neighboring nodes, and $|\cdot|$ denotes cardinality. If its residual has not changed more than the precision parameter δ for κ consecutive iterations, the ith node will quit the iterative update step and mark itself as *localized*. The overall update process terminates after all of the unlocalized nodes have been marked as localized. Iterative PPM for collaborative position location is described as the following.

Algorithm 6.2: Iterative PPM for collaborative position location:

- **Initialization:**
  Obtain initial guess $\Theta = [\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_N]$ using the closest-anchor initialization; Set $[\hat{\theta}_{N+1}, \hat{\theta}_{N+2}, \ldots, \hat{\theta}_{N+M}] = \mathbf{A} = [\theta_{N+1}, \theta_{N+2}, \ldots, \theta_{N+M}]$, which remain unchanged during the Main loop.
  Set $l = 0$, δ as a small positive number, and κ as a positive integer;

Let $F_i = 0$ and $W_i = 0$, for $i = 1, 2, \ldots, N$;
$\Phi_{i,l} = \Phi^{\text{col}}(\hat{\theta}_i)$, for $i = 1, 2, \ldots, N$;

- **Main loop:**

While (any of F_i is equal to 0) {

For $i = 1, 2, \ldots, N$; if $F_i = 0$

$$\hat{\theta}_i \leftarrow (1/|\mathcal{N}(i)|) \sum_{j \in \mathcal{N}(i)} P_{ij}^{\text{col}}(\hat{\theta}_i), \text{ and } \Phi_{i,l+1} = \Phi^{\text{col}}(\hat{\theta}_i);$$

If $|\Phi_{i,l} - \Phi_{i,l+1}| < \delta$

Let $W_i = W_i + 1$; If $W_i \geq \kappa$, Let $F_i = 1$;

Otherwise

Let $W_i = 0$;

$l = l + 1$;

}

In the above algorithm, F_i indicates whether the ith node has been localized and W_i records the number of consecutive iterations that the ith node's residual has not decreased more than δ. Once $W_i \geq \kappa$, we set $F_i = 1$ and consider the ith unlocalized node as localized. It is obvious that anchor locations, i.e., θ_j, for $j = N+1, N+2, \ldots, N+M$, will remain unchanged during the whole process. Again, as in the case of modified PPM for non-collaborative position location, the final solution of iterative PPM depends on the initial guess.

Compared to the case of non-collaborative position location, we can easily see that the major difference here is that we use W_i as a means of accumulating observations over multiple iterations regarding whether an unlocalized node is indeed localized. This is understandable since during the iterative update process, if an unlocalized node has updated its location, its neighboring unlocalized nodes may be affected in terms of their residuals, and therefore need to be re-examined. Another advantage of iterative PPM is that the computational load involved is low, mainly because of the simple operation involved in updating node locations which scales linearly as N. As we will demonstrate later, the algorithm's running time is significantly smaller than other existing algorithms while achieving generally better localization accuracy. Fig. 6.5 gives an example of applying the iterative PPM to the problem of collaborative position location for a particular network with the range estimation noise

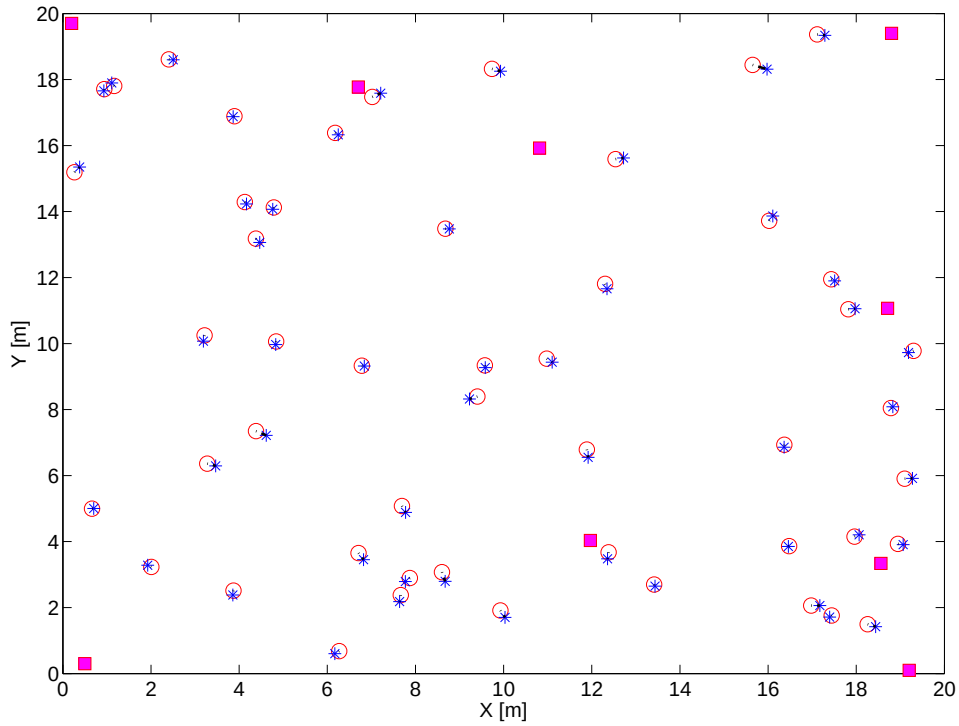


Figure 6.5: An example of applying the iterative PPM to collaborative position location for a network with 9 anchors (squares) and 50 unlocalized nodes (circles). We assume pure LOS scenario with $K_E = 0.0025$, $\beta = 2$ and $R = 8\,\mathrm{m}$. The asterisks are the estimated locations and dash-dot lines quantize the estimation error.

parameters of $K_E = 0.0025$ and $\beta = 2$, where there are 4 anchors near the four corners of the square network and another 5 anchors together with 50 unlocalized nodes randomly distributed in the network.

6.4 Node localizability via iterative PPM

In the presence of range estimation noise, it is difficult to investigate the problem of node localizability, i.e., whether an unlocalized node has a unique location solution. As we mentioned earlier, existing work which uses graph rigidity theory to address the issue of node localizability cannot handle the presence of range estimation noise [52, 66]. On the other hand, probabilistic localization methods provide a distribution of the node's location [53, 56], and thus naturally tells whether a node is uniquely localizable. The drawback is that such methods generally have high computational complexity and thus are not suitable for prac-

tical implementation. In our work, we develop a computationally efficient method to obtain a spatial approximation to each unlocalized node's possible location solution. In particular, taking advantage of the fast computation of the iterative PPM, we can run the algorithm with multiple distinct initial solutions for the unlocalized nodes. The localization results obtained in such a fashion will differ from each other and capture the perturbation effect of range estimation noise on the node's location estimation. Note that the output is similar to that of probabilistic localization, but computed in a faster way. Compared to graph rigidity theory based work on node localizability, our approach takes ranging noise into account and is more meaningful for practical scenarios. Compared to probabilistic localization approaches, our method is computationally much more efficient and suitable for practical implementation. Fig. 6.6 gives an example of using iterative PPM with $M_p = 50$ distinct initial guesses to examine node localizability. In particular, the qth initial solution for the ith node, denoted by $\theta_i^q = [x_i^q, y_i^q]$, is on the circle centered at the network center $[x_c, y_c]$ with a radius of $R_c = 10$, given by

$$x_i^q = x_c + R_c \cos\left[2\pi\left(\frac{i-1}{N} + \frac{q-1}{M_p}\right)\right], \tag{6.14}$$

$$y_i^q = y_c + R_c \sin\left[2\pi\left(\frac{i-1}{N} + \frac{q-1}{M_p}\right)\right], \tag{6.15}$$

for $i = 1, 2, \ldots, n$ and $q = 1, 2, \ldots, M_p$. With these initial solutions, we presented the possible solutions for the 2nd and 3rd nodes in Fig. 6.6. We also plot their respective estimated locations (blue asterisks) which, among the 50 possible solutions, is the one with the smallest residual. The 100% *solution radius* is hereby defined as the radius of the minimum circle centered at the estimated location that encloses all 50 solutions. It is easily seen that the 2nd node's possible solutions are concentrated within a small region, and thus result in a solution radius of about 2 m. In contrast, the 3rd node, which has a few solutions (around the 5th node) far away from its true location, has a much larger solution radius of about 10 m. This indicates that the 3rd node is not uniquely localizable. Considering the nodes that the 3rd node is connected to, it is easy to see that these nodes are nearly on a straight line, which causes a flip ambiguity for the 3rd node. We emphasize that obtaining multiple solutions is made computationally feasible by the iterative PPM due to its fast computation. In the results of Fig. 6.6, the running time with $M_p = 50$ is still comparable to that of either the SDP [27] or the distributed MDS-MAP(P, R) [20] approaches.

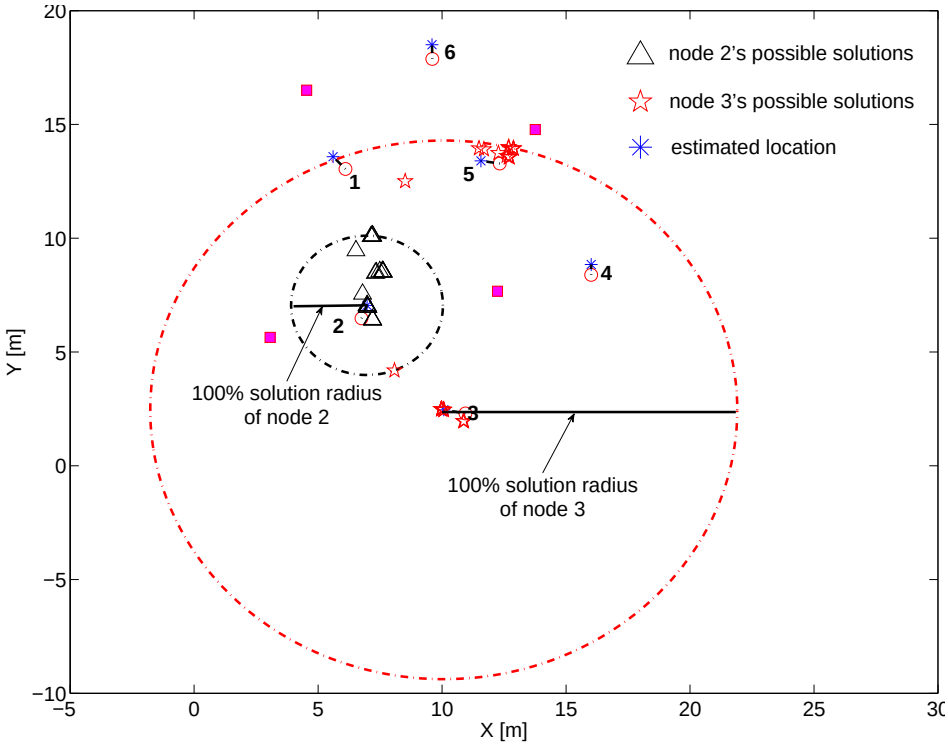


Figure 6.6: An example of using iterative PPM to investigate node localizability. We only highlight the results for the 2nd and 3rd unlocalized nodes. The triangles and stars represent possible solutions of the 2nd and 3rd unlocalized nodes, respectively, resulting from 50 distinct initial solutions, for a particular noise realization.

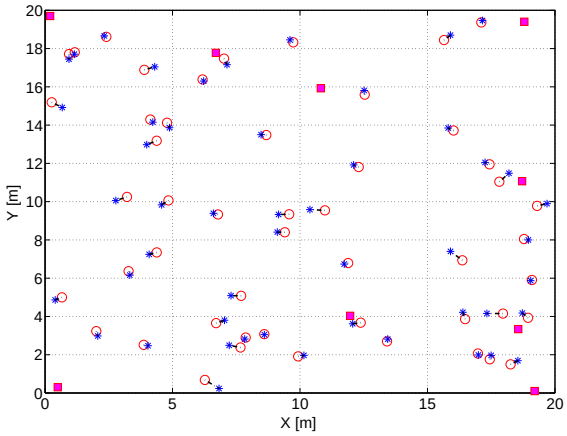
6.5 Numerical results

6.5.1 Localization accuracy

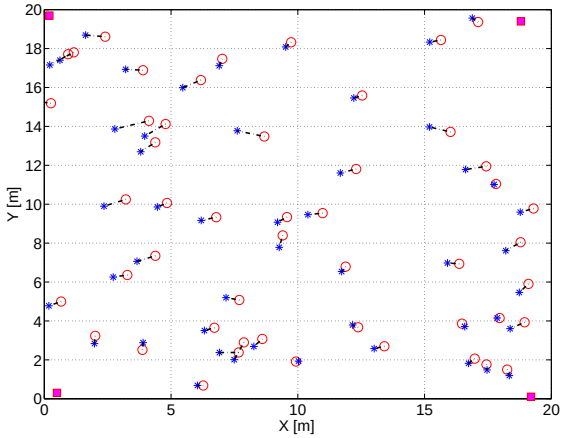
In this section, we present simulation results of our proposed iterative PPM approach for collaborative position location, in terms of localization accuracy as well as the algorithm running time, compared to two popular approaches, *i.e.*, SDP or MDS-MAP. As for the localization accuracy, we compute the mean localization error using the RMSE given by (2.2), while for the algorithm running time, we compare the time consumed in seconds in Matlab 2008 on an Intel 3.3GHz CPU. For the SDP approach, we use the centralized full SDP solver provided by [27]. For the MDS-MAP, we used the patch-based MDS followed by the LS refinement, i.e., MDS-MAP(P, R), with a radius of $2R$ for patching local maps and centralized LS refinement, which gives the best performance among all the four MDS

approaches according to [20]. Our experimental results show that the full SDP solver provides robust performance for different network shapes with reasonably distributed anchors. To the best of our knowledge and also as mentioned in [31], the MDS-MAP(P, R) appears to be among the best distributed methods with reasonably short running time. Our intention is to show that the iterative PPM approach is able to provide comparable and often better localization accuracy than the two methods, while requiring significantly shorter running time. In addition, our simulation results differ from most existing work in the sense that besides Gaussian range estimation noise, i.e., LOS scenario, we also investigate the robustness of the three localization methods against NLOS propagation. The percentage of NLOS range estimates and the amount of NLOS bias will be given in specific simulations. The range estimation noise and NLOS bias models are as described earlier. For our proposed iterative PPM, we use $\delta = 10^{-4}$ and $\kappa = 10$, as using a smaller δ and a larger κ does not further improve the performance of iterative PPM.

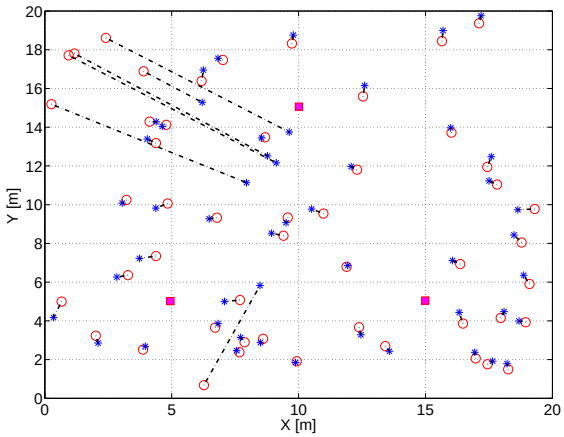
In Fig. 6.7, we plot the localization results from applying the iterative PPM approach to three networks with different numbers and configurations of anchors. In the three networks, the number of unlocalized nodes is 50, while the numbers of anchors are $m = 9$, $m = 4$ and $m = 3$ for Fig. 6.7 (a), (b) and (c), respectively. The communication range is $R = 8\,\mathrm{m}$ for all three networks and the resulting average node connectivity, i.e., the average number of neighbors per node, is approximately 9 for all three networks. In Fig. 6.7 (a) and (b), there are 4 anchors at the four corners of a $20 \times 20\,\mathrm{m}^2$ network, where in Fig. 6.7 (a), there are 5 more anchors randomly distributed within the network. In Fig. 6.7 (c), there are only 3 anchors located within the network forming an equilateral triangle. Clearly, Fig. 6.7 (a) represents the most favorable anchor deployment, while Fig. 6.7 (c) is the least favorable deployment, since as shown in [89], perimeter anchor deployment inherently yields better localization accuracy. As shown in Fig. 6.7, the estimated locations are denoted by asterisks and the dashed lines represent the amount of localization error. We assume a pure LOS scenario with $\beta = 2$ and $K_E = 0.01$. As suggested by our noise variance model in $\sigma_{ij}^2 = K_E d_{ij}^{\beta}$, this is equivalent to range estimate noise which is approximately 10% of the inter-node distance. We can easily observe that in Fig. 6.7 (c), a few nodes get flipped solutions, and therefore resulting in a much larger localization error of 2.37 m, while Fig. 6.7 (a) has the smallest localization error of 0.34 m, due to the larger number of as well as better distributed anchors. This makes intuitive sense since in our proposed iterative PPM approach, the initial solution follows the so-called closest anchor method, which generally leads to worse initial solution when the number of anchors is small and they are more concentrated. With a bad initial solution, it is more likely that the iterative PPM converges to local minima and thus provides



(a) $m = 9$



(b) $m = 4$



(c) $m = 3$

Figure 6.7: The effect of the number of anchors on iterative PPM for: (a) 4 anchors at the four corners and 5 randomly within the network (RMSE = 0.34 m); (b) 4 anchors at the four corners (RMSE = 0.63 m); (c) 3 anchors within the network (RMSE = 2.37 m); Assume pure LOS range estimates, with $R = 8$ m, $\beta = 2$ and $K_E = 0.01$, corresponding to 10% of inter-node distance. The circles and squares are the true locations of unlocalized nodes and anchors, respectively. The asterisks represent the estimates locations and the dashes lines quantize the localization error.

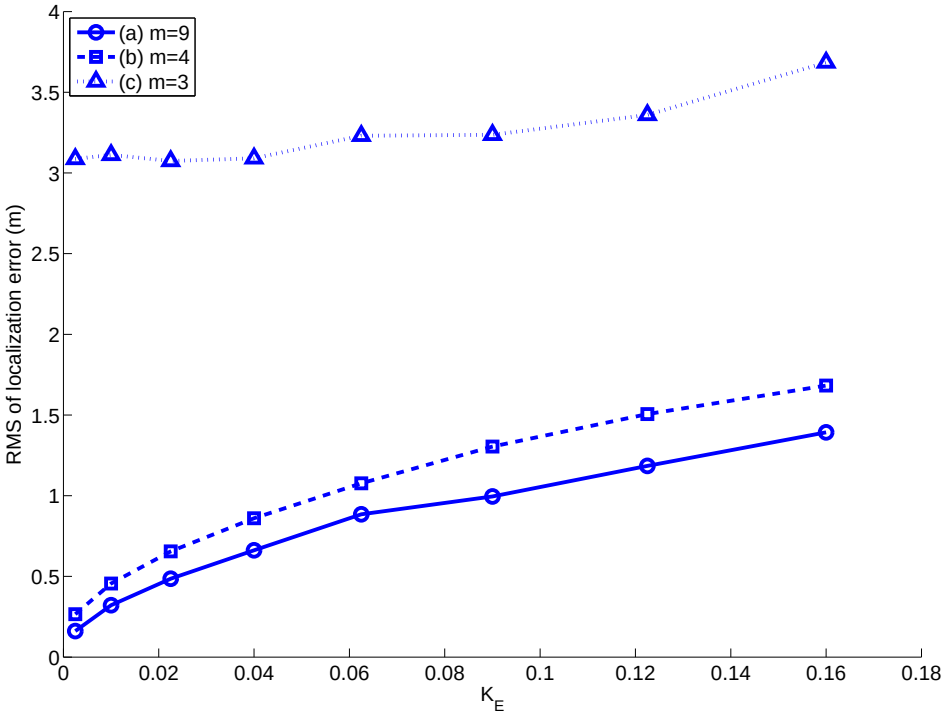


Figure 6.8: The corresponding mean localization errors versus K_E (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$. The larger K_E, the larger ranging noise is.) for the three cases in Fig. 6.7.

erroneous solutions.

In Fig. 6.8, we plot the mean localization error of the iterative PPM approach when localizing the three networks in Fig. 6.7 (a), (b) and (c), versus range estimation noise indicated by the value of K_E. We set $\beta = 2$ and $R = 8\,\text{m}$, while varying the value of K_E from 0.0025 to 0.16. As mentioned earlier, such a range of values for K_E corresponds to range estimation noise equivalent to 5% up to 40% of the inter-node distance. The larger K_E, the more range estimation noise. For each value of K_E, 500 noise realizations are simulated. As can be seen, the mean localization errors in all three cases increase as the amount of range estimation noise increases, among which the localization error is the smallest for Fig. 6.7 (a) due to more and better anchor deployment, while Fig. 6.7 (c) has the largest localization error due to fewer and poor anchor deployment. It is obvious that the effect of bad anchor deployment is much more prominent than the increase of range estimation noise for the case of Fig. 6.7 (c). Again, we emphasize that due to the closest-anchor initialization step adopted in our approach, its performance depends upon how well the anchors are distributed over the entire network. For example, if all anchors are concentrated within a small region, the

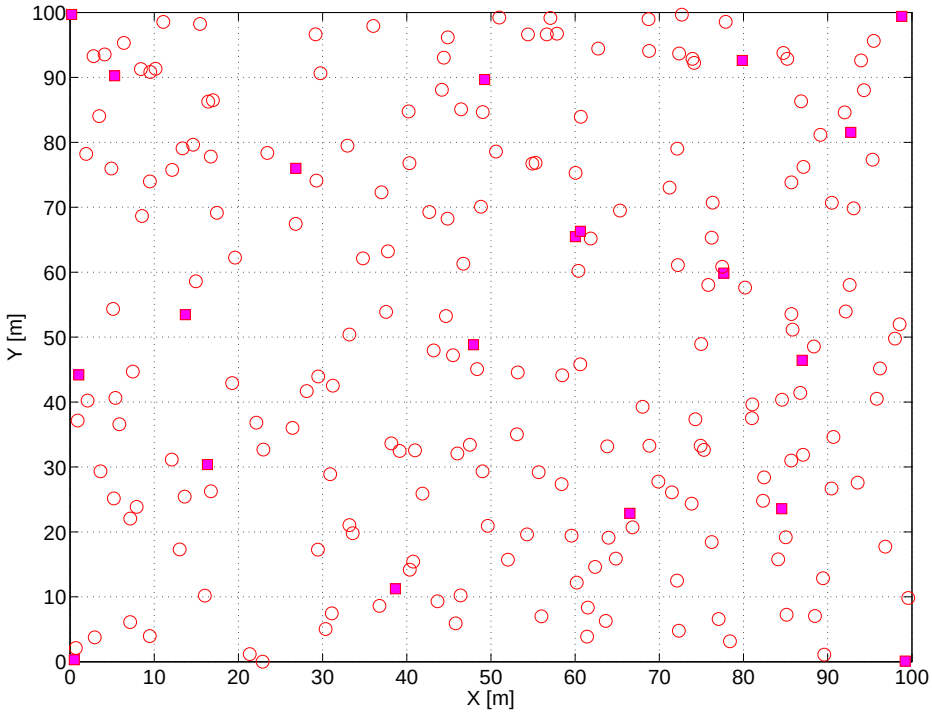
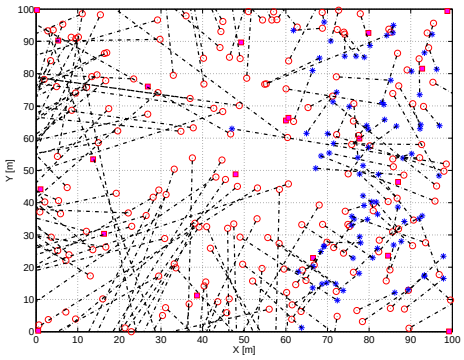


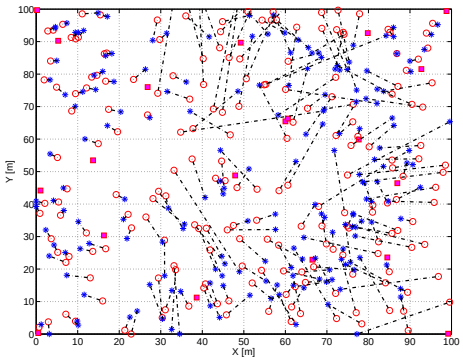
Figure 6.9: A regular-shaped random network with 200 unlocalized nodes (circles) and 20 anchors (squares). The communication range is $R = 15\,\mathrm{m}$, the average node connectivity is 12.9 and the average number of connected anchors is 1.1. We will compare the localization accuracy of the iterative PPM, the SDP and the MDS-MAP(P, R) using this network.

closest-anchor initialization is not a smart choice. Instead, using random initialization may be a better choice. It is also worth mentioning that the SDP and the MDS-MAP will also be affected by fewer and bad anchor placement. In this work, we will focus on reasonable anchor deployment (10% anchors) and optimal anchor deployment is beyond the scope of this paper.

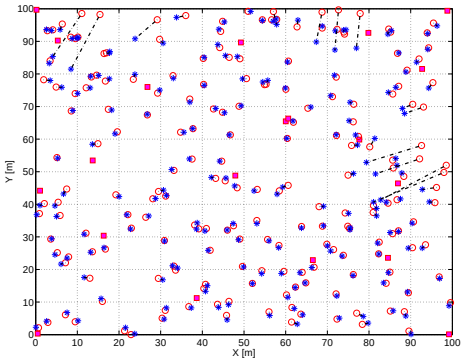
We now compare our proposed approach with the SDP and the MDS-MAP(P, R) on a large network with 200 unlocalized nodes and 20 anchors, randomly distributed within a $100 \times 100\,\mathrm{m}^2$ network, as shown in Fig. 6.9. In Fig. 6.10, we plot the estimated locations for one particular noise realization in the pure LOS scenario with $K_E = 0.01$ and $\beta = 2$, when applying our proposed approach, the SDP and the MDS-MAP(P, R), where the dashed lines again quantize the amount of localization error. To provide a reference, we also presented the results of the sequential LS estimator and the CQLP with 3-hop node collaboration. It is easily seen that in random large networks like this example, the paradigm of applying non-



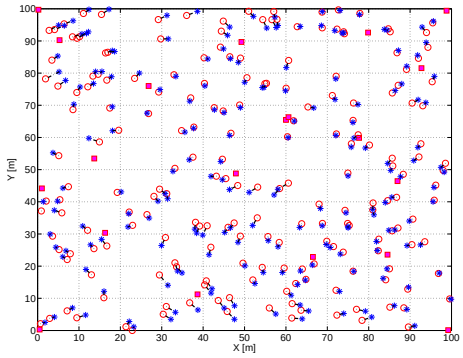
(a) Sequential LS, $\Omega = 107\,\mathrm{m}$



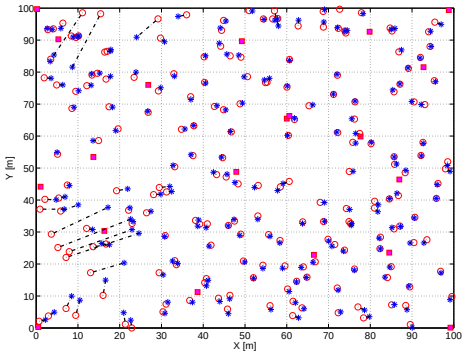
(b) CQLP(3-hop), $\Omega = 20\,\mathrm{m}$



(c) SDP approach, $\Omega = 2.8\,\mathrm{m}$

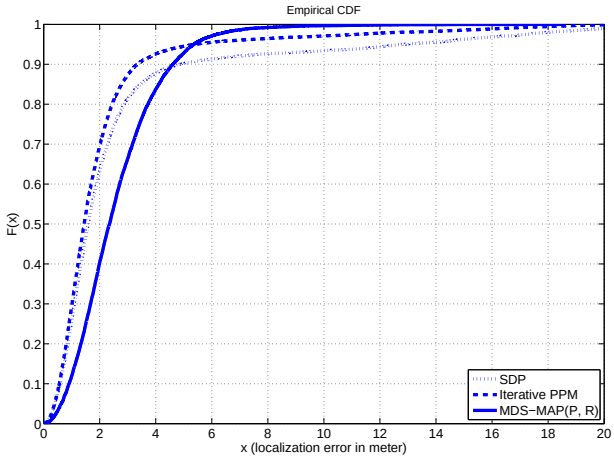


(d) MDS-MAP(P, R), $\Omega = 1.5\,\mathrm{m}$

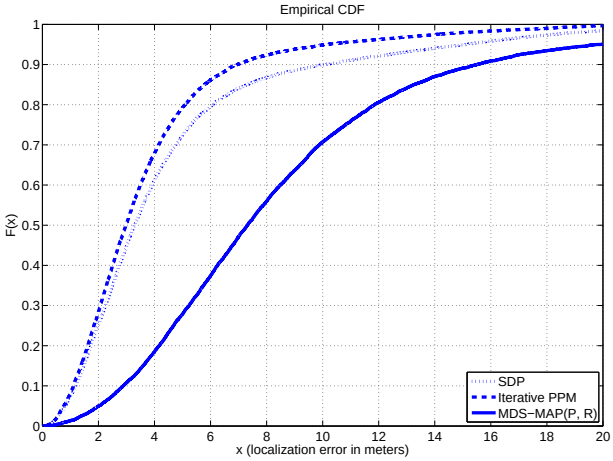


(e) Iterative PPM, $\Omega = 2.4\,\mathrm{m}$

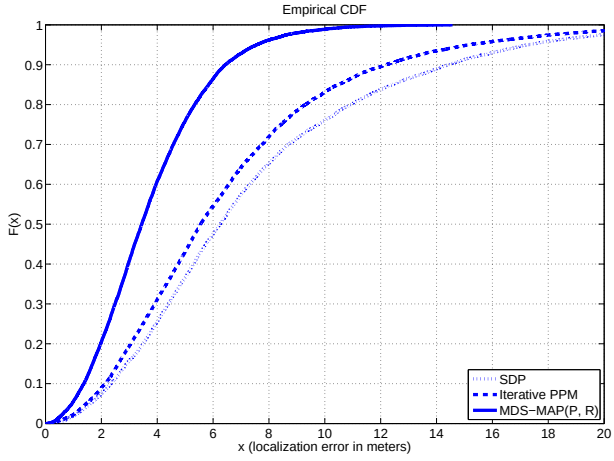
Figure 6.10: Results of applying different localization schemes to the network in Fig. 6.9 for one particular noise realization in the pure LOS scenario with $K_E = 0.01$ and $\beta = 2$ (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$).



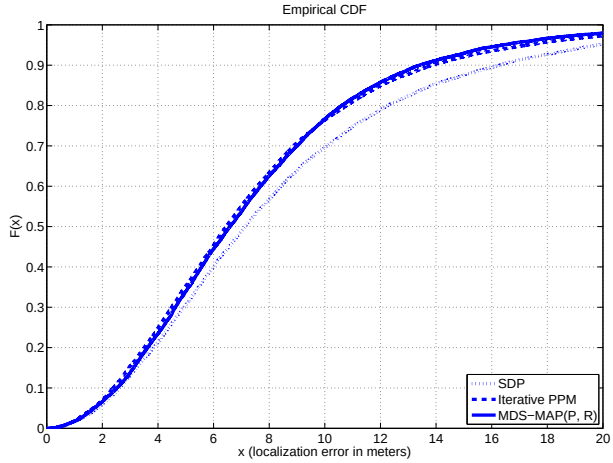
(a) $K_E = 0.04$, pure LOS



(b) $K_E = 0.16$, pure LOS



(c) $K_E = 0.04$, 40% NLOS



(d) $K_E = 0.16$, 40% NLOS

Figure 6.11: Cumulative density function of the localization errors of SDP, MDS-MAP(P, R) and iterative PPM for both pure LOS and 40% NLOS scenarios, with $K_E = 0.04$ and 0.16 (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$, $\beta_{ij} = 2$ for LOS and $\beta_{ij} = 3$ for NLOS), respectively.

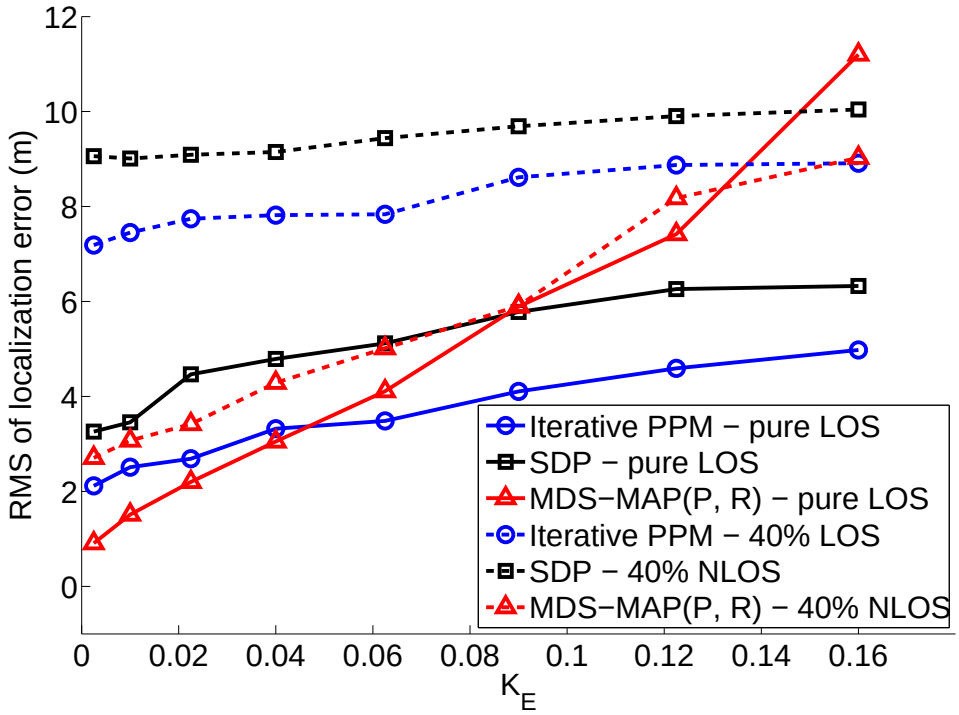


Figure 6.12: The mean localization error versus the amount of range estimation noise (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$), for the network shown in Fig. 6.9, with pure LOS and 40% NLOS range estimates. K_E varies from 0.0025 to 0.16, corresponding to 5% up to 40% range estimation noise.

collaborative position location schemes in a sequential fashion suffers significant performance loss due to the combined effect of error propagation and potentially bad node geometry.

We present the mean localization error versus range estimation noise, for both pure LOS and NLOS scenarios. Since the sequential LS estimator and the CQLP perform very badly, we only presented the results of the SDP, the MDS-MAP(P, R) and our proposed iterative PPM approach. The communication range is assumed to be $R = 15\,\text{m}$, the resulting average node connectivity is 12.9 and on average, each unlocalized node is connected to only 1.1 anchors. Compared to simulation results in [20] and [31] where the average node connectivity is usually larger than 15 and sometimes increases to thirty, we believe our simulation results represent a more practical case where sensor nodes have shorter communication range, which is more practical due to power consumption concerns. Nevertheless, with larger node connectivity, the localization accuracy of all three methods will be improved. For each value of K_E, we ran 50 noise realizations. We also compare the three methods in terms of their ro-

bustness against the presence of NLOS propagations. As for the NLOS scenario, we assume any range estimate has 40% chance of being NLOS. If a range estimate is NLOS, a uniformly distributed positive bias is added onto that range estimate with $b_{\min} = 4\,\text{m}$ and $b_{\max} = 8\,\text{m}$. The minimum NLOS bias of $b_{\min} = 4\,\text{m}$ is equivalent to 27% of the communication range and represents the fact that in practice, NLOS bias is usually much larger than the range estimation noise. In Fig. 6.11, we plot the empirical cumulative density function (CDF) of the localization error of SDP, MDS-MAP(P, R) and iterative PPM, for pure LOS and 40% NLOS scenarios, with $K_E = 0.04$ and $K_E = 0.16$, respectively. As shown in Fig. 6.11 (a), the MDS-MAP(P, R) is better than the other two in the sense that it reduces the large localization error. However, as the range estimation noise increases, as shown in Fig. 6.11 (b), MDS-MAP(P, R) performs worse than the other two. On the other hand, in the presence of 40% NLOS range estimates, both iterative PPM and SDP suffer performance degradation. MDS-MAP(P, R), although performs better in the case of smaller range estimation noise, i.e., $K_E = 0.04$, will have similar performance loss when range estimation noise increases, i.e., $K_E = 0.16$.

In Fig. 6.12, we plot the mean localization error, i.e., the RMS values of the localization error, of the three methods with respect to different range estimation noise. For the case of pure LOS range estimates, our proposed iterative PPM approach outperforms SDP for all values of K_E. The MDS-MAP(P, R) performs better for a small amount of range estimation noise, but performs worse than both the SDP and the iterative PPM approach as range estimation noise increases. It is obvious that both SDP and iterative PPM are more robust against increasing range estimation noise than MDS-MAP(P, R). From our experiments, we believe this can be attributed to the fact that the MDS step in the MDA-MAP(P, R) is impacted by an increase of range estimation noise more than the other two methods. As for the 40%-NLOS scenario, it is observed that both the SDP and the iterative PPM approach are strongly affected by the NLOS bias, with iterative PPM still outperforming the SDP approach. For instance, for $K_E = 0.09$ (30% noise), the localization error of iterative PPM increases from about 4 m to more than 8 m, while that of the SDP increases from about 5.5 m to more than 9 m. MDS-MAP(P, R), on the other hand, shows more robustness against the NLOS bias. As can be seen, its localization error in the case of 40%-NLOS scenario is worse than the pure LOS scenario, but becomes comparable to the pure LOS scenario as K_E increases. Another important observation is that as range estimation noise increases, the improvement of the MDS-MAP(P, R) over the SDP and the iterative PPM diminishes. This is again due to the degraded performance of the MDS-MAP(P, R) at higher levels of ranging noise.

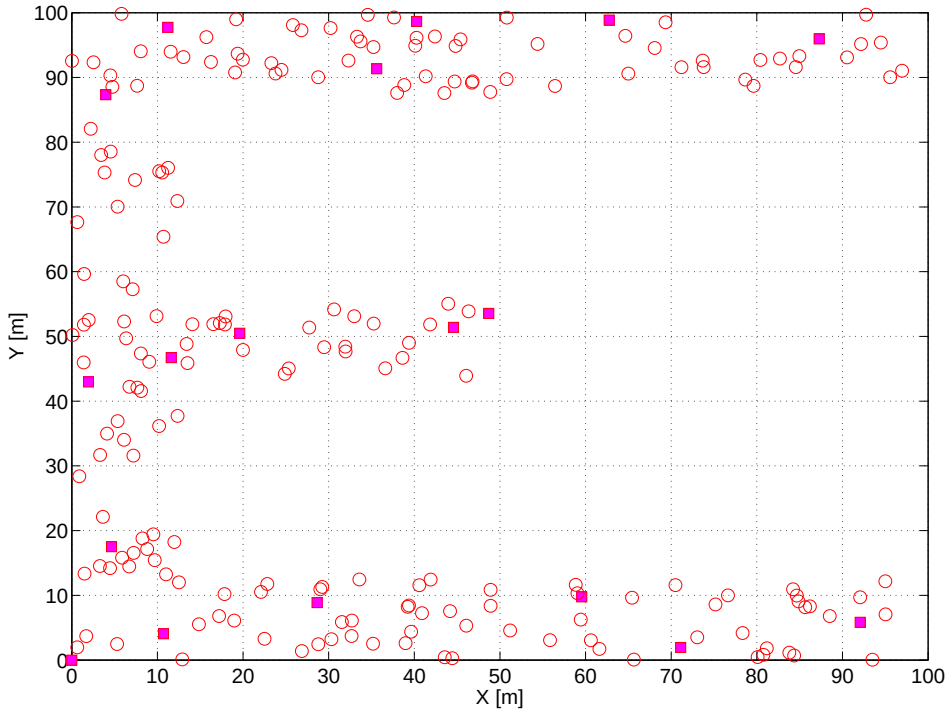
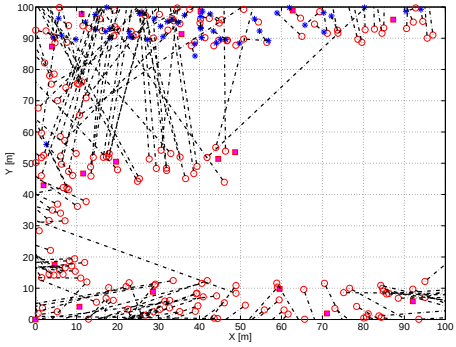
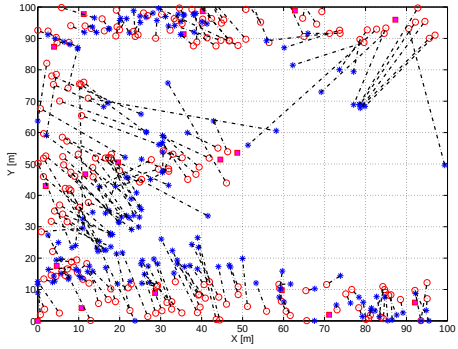


Figure 6.13: An E-shaped network with 200 unlocalized nodes (circles) and 20 anchors (squares). The communication range is $R = 15\,\text{m}$, the average node connectivity is 19.9 and the average number of connected anchors is 1.9. We will compare the localization accuracy of the iterative PPM, the SDP and the MDS-MAP(P, R) using this network.

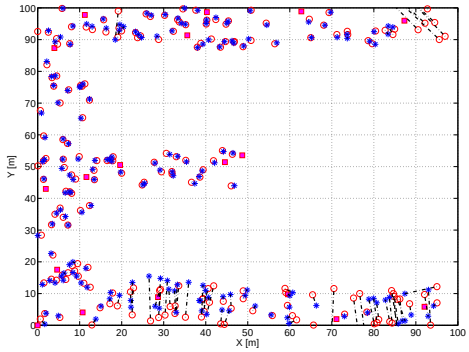
In practical sensor network deployment, it is likely that the network shape is irregular due to the presence of physical obstacles such as buildings. In Fig. 6.13, we give an example of an E-shaped network, with 200 unlocalized nodes and 20 anchors. The communication range is assumed to be $R = 15\,\text{m}$, and the resulting average node connectivity is 19.9 and on average, each unlocalized is connected to 1.9 anchors. We use the same parameters for the range estimation noise and NLOS bias and present localization results of all the methods in Fig. 6.14 for one particular noise realization in the pure LOS scenario with $K_E = 0.01$ and $\beta = 2$. It is clear that both the sequential LS estimator and the CQLP framework suffer huge performance loss again due to the combined effect of potentially bad neighboring node geometry and error propagation. This effect is even more severe in the case of irregular network shape since it is more likely that node geometry becomes ill-conditioned. Another important observation is that the MDS-MAP(P,R) also perform badly. This phenomenon has been reported in [20], because the step of building the complete distance matrix in the



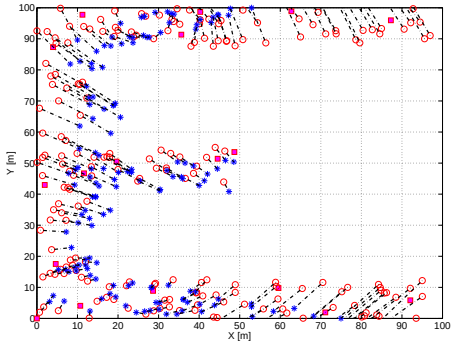
(a) Sequential LS, $\Omega = 403\,\mathrm{m}$



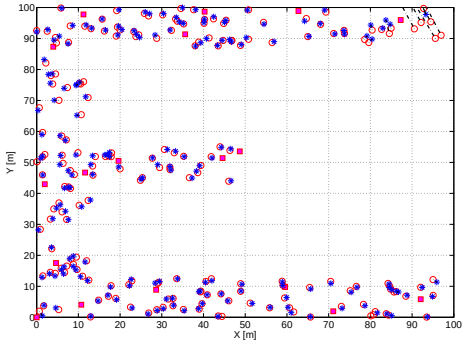
(b) CQLP(3-hop), $\Omega = 21.3\,\mathrm{m}$



(c) SDP approach, $\Omega = 3.0\,\mathrm{m}$

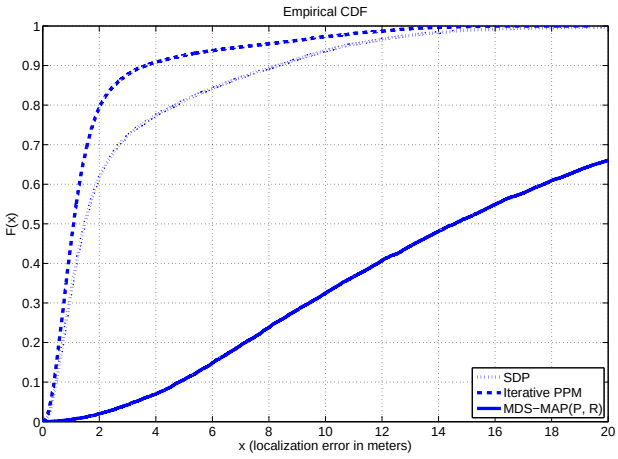


(d) MDS-MAP(P, R), $\Omega = 16.2\,\mathrm{m}$

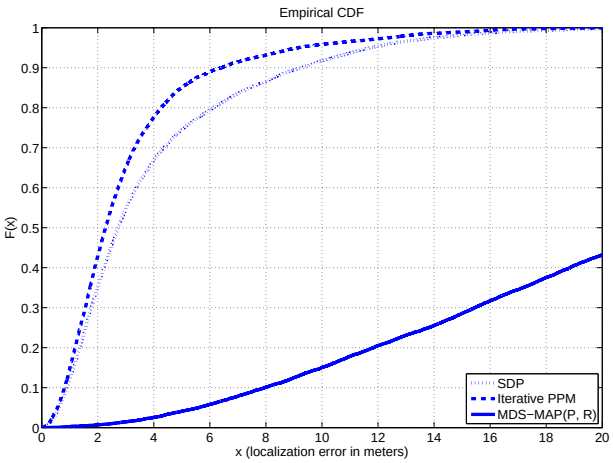


(e) Iterative PPM, $\Omega = 1.4\,\mathrm{m}$

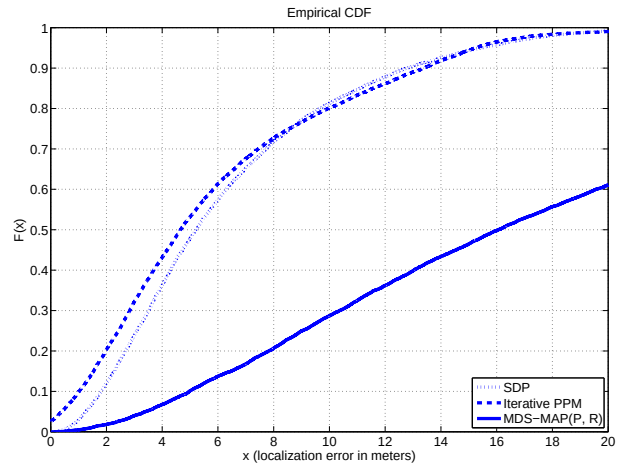
Figure 6.14: Results of applying different localization schemes to the network in Fig. 6.13 for one particular noise realization in the pure LOS scenario with $K_E = 0.01$ and $\beta = 2$ (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$).



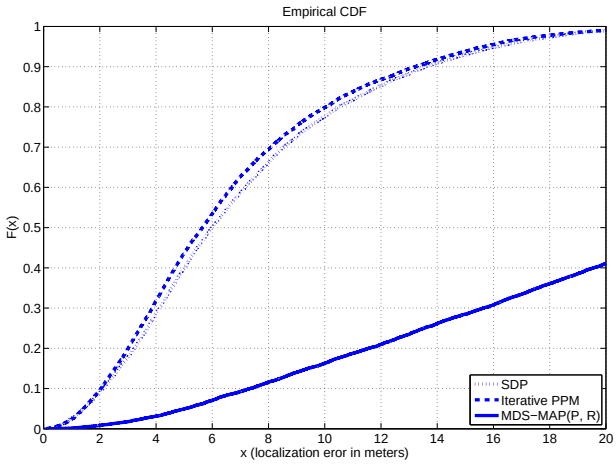
(a) $K_E = 0.04$, pure LOS



(b) $K_E = 0.16$, pure LOS



(c) $K_E = 0.04$, 40% NLOS



(d) $K_E = 0.16$, 40% NLOS

Figure 6.15: Cumulative density function of the localization errors of SDP, MDS-MAP(P, R) and iterative PPM for both pure LOS and 40% NLOS scenarios, with $K_E = 0.04$ and 0.16 (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$, $\beta_{ij} = 2$ for LOS and $\beta_{ij} = 3$ for NLOS), respectively, applying to the E-shape network.

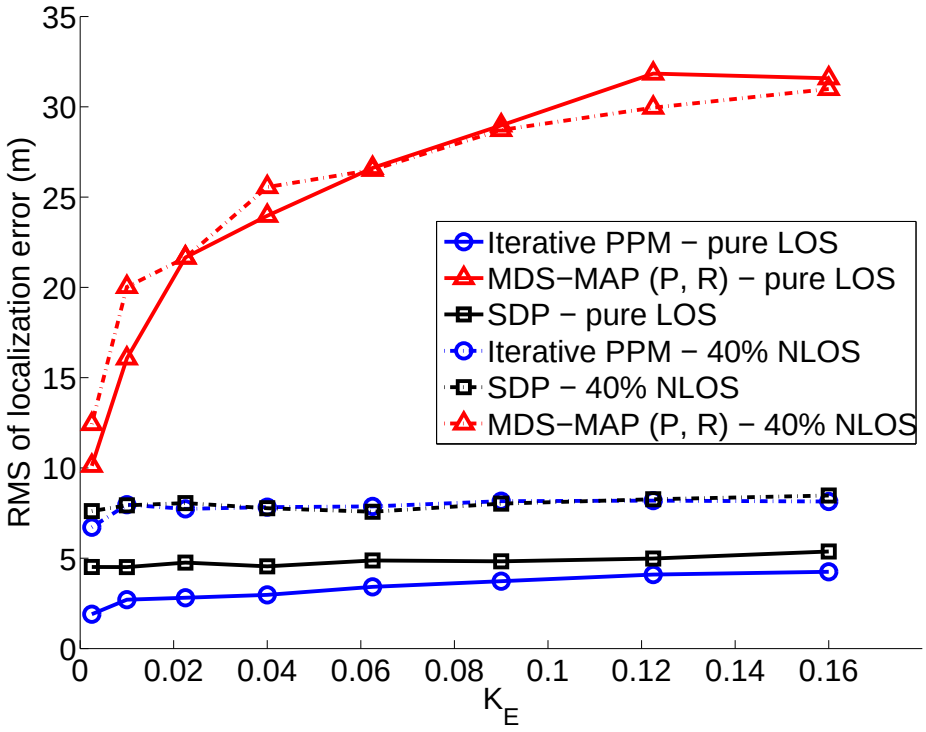


Figure 6.16: The mean localization error versus the amount of range estimation noise (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$), for the network shown in Fig. 6.13, with pure LOS and 40% NLOS range estimates. K_E varies from 0.0025 to 0.16, corresponding to 5% up to 40% range estimation noise.

MDS algorithm becomes unreliable when the network shape is anisotropic. In Fig. 6.15, we plot the CDF of the localization errors of the three methods, for pure LOS and 40% NLOS scenarios, with $K_E = 0.04$ and $K_E = 0.16$, respectively. As we can see, MDS-MAP(P, R) performs extremely bad in this irregular network shape. SDP and iterative PPM, on the other hand, have similar performance due to their better robustness against anisotropic network shapes.

In Fig. 6.16, we present the corresponding simulation results on the RMS localization error of SDP, MDS-MAP(P, R) and our proposed iterative PPM approach. It is observed that for both the pure LOS and 40%-NLOS scenarios, the mean localization error for iterative PPM approach and SDP approach are comparable to that of the regular network shape in Fig. 6.9. Both methods demonstrate similar robustness against range estimation noise. In particular, as K_E increases from 0.0025 to 0.16, the localization error will generally increase less than 2 m for both pure LOS and 40%-NLOS scenarios. On the other hand, MDS-

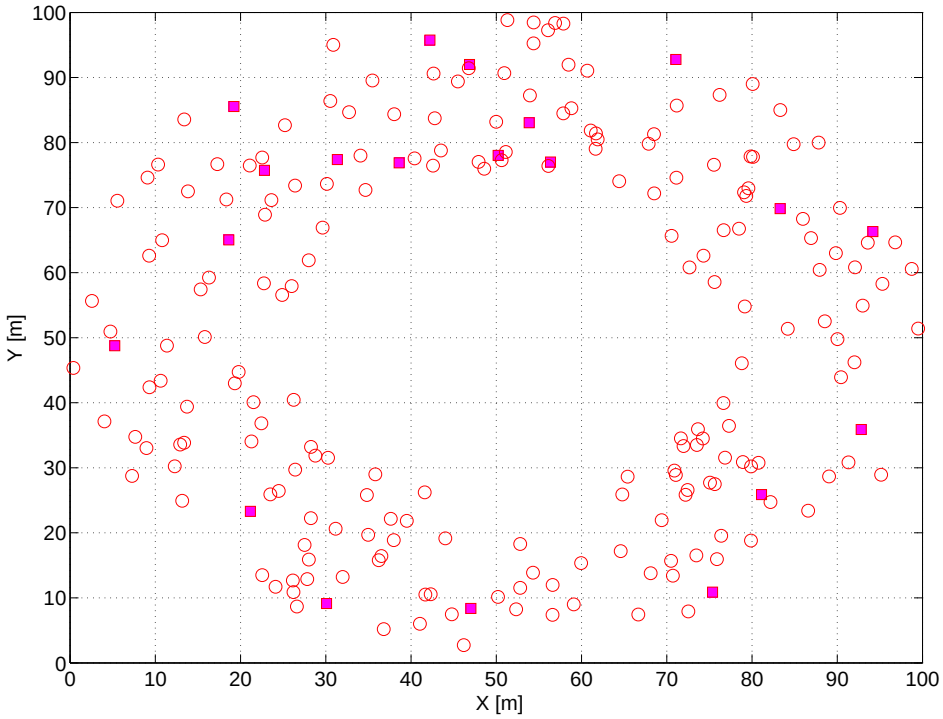
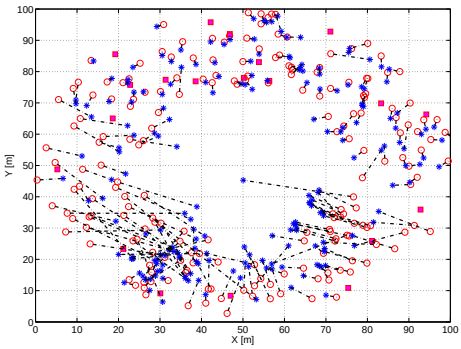


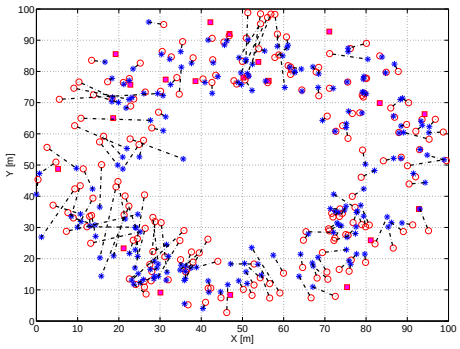
Figure 6.17: A ring-shaped network with 200 unlocalized nodes (circles) and 20 anchors (squares). The communication range is $R = 15\,\text{m}$, the average node connectivity is 20.7 and the average number of connected anchors is 1.7. We will compare the localization accuracy of the iterative PPM, the SDP and the MDS-MAP(P, R) using this network.

MAP(P, R) performs badly with an irregular network shape, whether it is a pure LOS or NLOS scenario. As shown in Fig. 6.16, the localization error of MDS-MAP(P, R) is always larger than 10 m and increases significantly as range estimation noise increases. This is, as mentioned in [20], due to the inherent nature of the MDS step which suffers from irregular network shapes, despite the distributed approach used here, i.e., MDS-MAP(P, R).

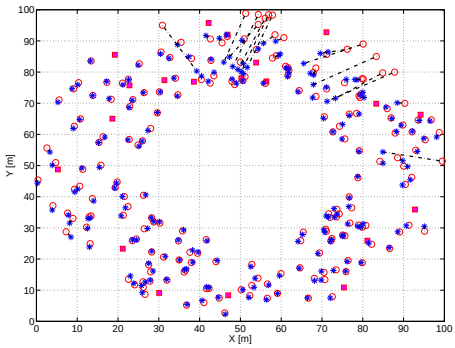
In Fig. 6.17, we give another example of an irregularly shaped network, where the network of nodes forms a "ring" shape. Again, the number of unlocalized nodes is 200 and the number of anchors is 20. The communication range is assumed to be $R = 15\,\text{m}$, and the resulting average node connectivity is 20.7 and on average, each unlocalized is connected to 1.7 anchors. We use the same parameters for generating range estimation noise and NLOS bias as the previous two networks and present the localization results of all the methods in Fig. 6.18 for one particular noise realization in the pure LOS scenario with $K_E = 0.01$ and $\beta = 2$. Again, both the sequential LS estimator and the CQLP framework perform



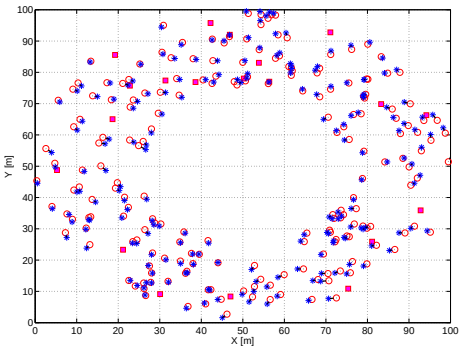
(a) Sequential LS, $\Omega = 11.6\,\mathrm{m}$



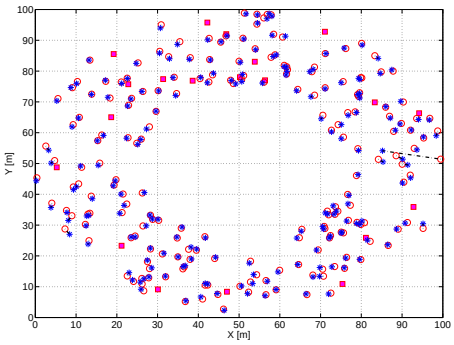
(b) CQLP(3-hop), $\Omega = 8.7\,\mathrm{m}$



(c) SDP approach, $\Omega = 4.8\,\mathrm{m}$

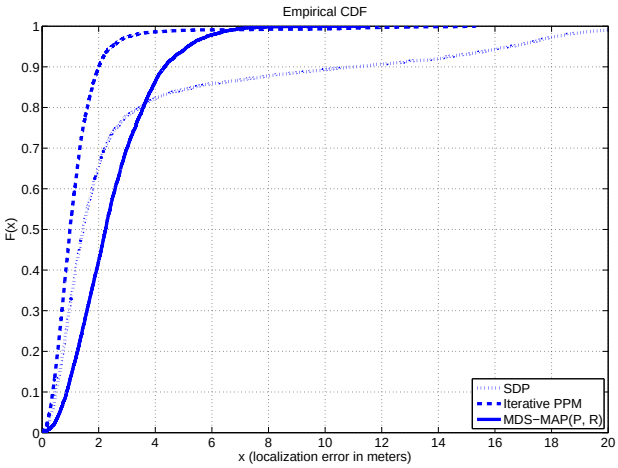


(d) MDS-MAP(P, R), $\Omega = 1.5\,\mathrm{m}$

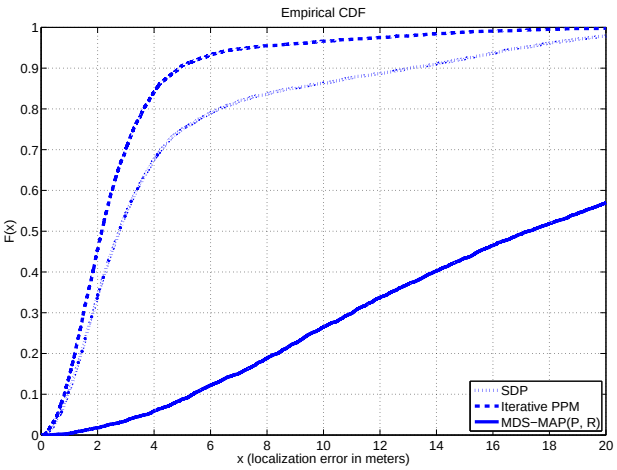


(e) Iterative PPM, $\Omega = 2.6\,\mathrm{m}$

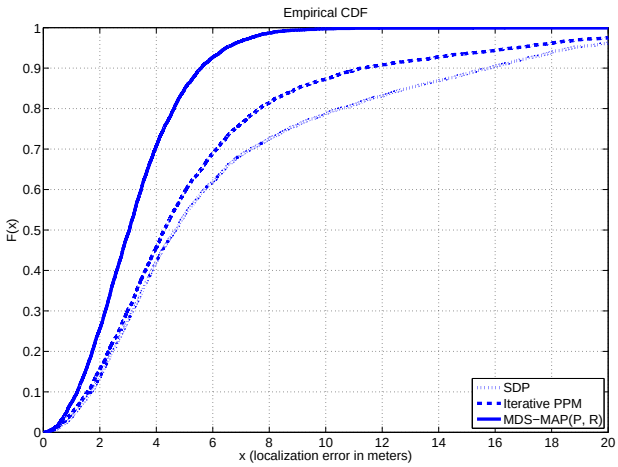
Figure 6.18: Results of applying different localization schemes to the network in Fig. 6.17 for one particular noise realization in the pure LOS scenario with $K_E = 0.01$ and $\beta = 2$ (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$).



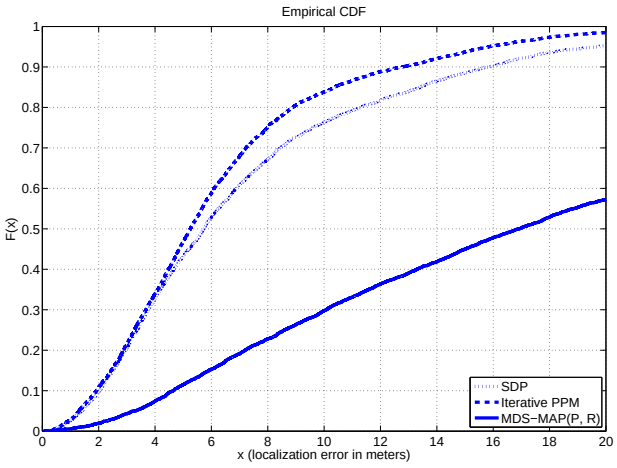
(a) $K_E = 0.04$, pure LOS



(b) $K_E = 0.16$, pure LOS



(c) $K_E = 0.04$, 40% NLOS



(d) $K_E = 0.16$, 40% NLOS

Figure 6.19: Cumulative density function of the localization errors of SDP, MDS-MAP(P, R) and iterative PPM for both pure LOS and 40% NLOS scenarios, with $K_E = 0.04$ and 0.16 (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$, $\beta_{ij} = 2$ for LOS and $\beta_{ij} = 3$ for NLOS), respectively, applying to the ring-shape network.

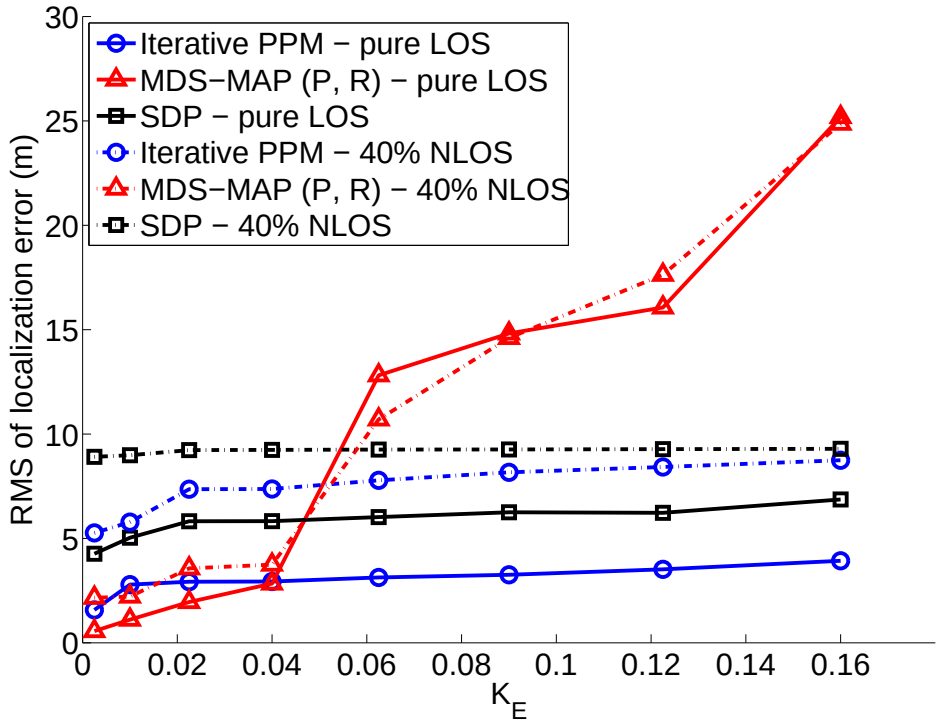
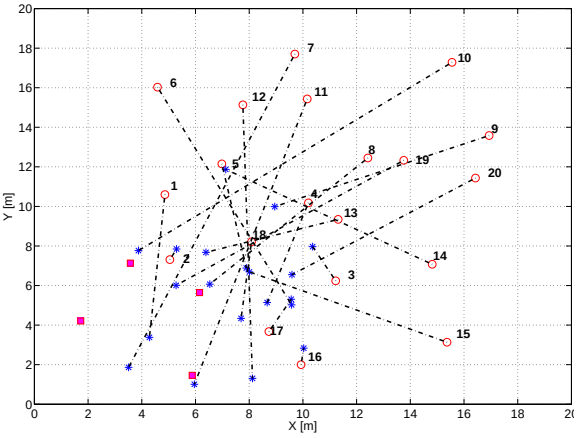


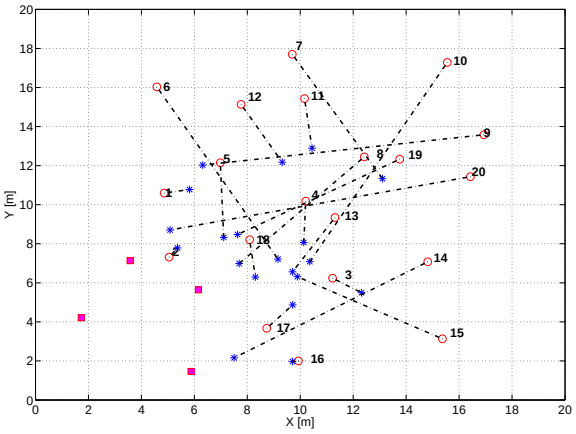
Figure 6.20: The mean localization error versus the amount of range estimation noise (the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$), for the network shown in Fig. 6.17, with pure LOS and 40% NLOS range estimates. K_E varies from 0.0025 to 0.16, corresponding to 5% up to 40% range estimation noise.

badly. MDS-MAP(P,R) in this case performs much better than in the E-shaped net, mainly because the network shape here is more isotropic. Our proposed iterative PPM approach, on the other hand, still performs better than the SDP and close to the MDS-MAP(P, R). In Fig. 6.19, we plot the CDF of the localization errors of the three methods, for pure LOS and 40% NLOS scenarios, with $K_E = 0.04$ and $K_E = 0.16$, respectively. As we can see, MDS-MAP(P, R) performs extremely bad in this irregular network shape. SDP and iterative PPM, on the other hand, have similar performance due to their better robustness against anisotropic network shapes. Again, we observe MDS-MAP(P, R) performs better when the range estimation noise is small, while performs worse as range estimation noise increases.

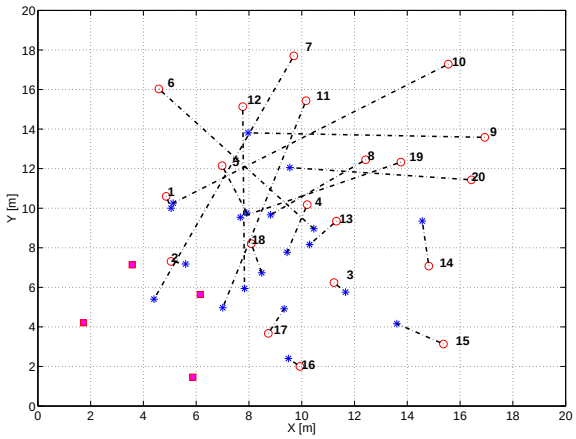
In Fig. 6.20, we present the results on the mean localization error. It is observed that both SDP and iterative PPM approaches achieve localization accuracy comparable to the regularly shaped network and demonstrate similar robustness against range estimation noise for both pure LOS and 40%-NLOS scenarios. In addition, we see a performance improvement from



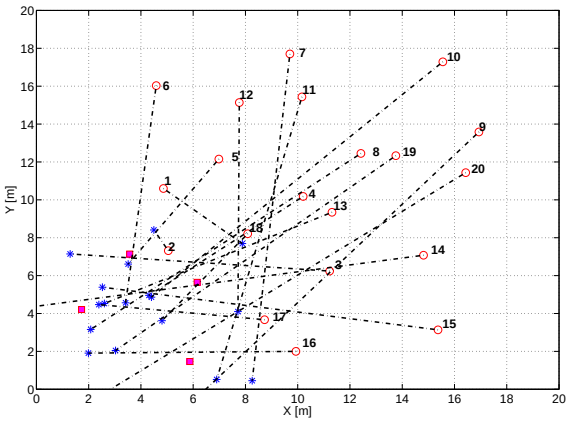
(a) Iterative PPM with closest-anchor initialization, $\Omega = 9.28\,\mathrm{m}$



(b) Iterative PPM with random initialization, $\Omega = 6.37\,\mathrm{m}$



(c) MDS-MAP(P,R), $\Omega = 6.52\,\mathrm{m}$



(d) SDP, $\Omega = 12.4\,\mathrm{m}$

Figure 6.21: Localization results of iterative PPM with closest-anchor initialization, random initialization and MDS-MAP(P,R) and SDP for a network where anchors are not well spread into the network. Anchors and the true locations of unlocalized nodes are shown by squares and circles, respectively. The estimated locations are shown by asterisks and the dashed lines quantize localization error. The communication range is $R = 6\,\mathrm{m}$ and the average node connectivity is 7.2. We assume pure LOS scenario and the noise variance is $\sigma_{ij}^2 = K_E d_{ij}^{\beta_{ij}}$ with $K_E = 0.09$ and $\beta_{ij} = 2$.

iterative PPM approach over the SDP, compared to Fig. 6.13. As observed in our simulation, this is because SDP is impacted more by the effect where some nodes are *dragged* toward anchors during iterations, while our proposed iterative PPM approach deals with these cases a little better, due to our proposed projection onto the ranging circle as in (6.9). MDS-MAP(P, R), on the other hand, performs better than the other two when the range estimation noise is very small, but eventually performs worse as the range estimation noise increases to 25% ($K_E = 0.0625$). Also, the performance of MDS-MAP(P, R) in Fig. 6.17 in better than that in Fig. 6.13, mainly because the network in Fig. 6.17 is more isotropic than Fig. 6.13, which yields a better result for the MDS step in MDS-MAP(P, R). Another observation in Fig. 6.20 is that, although the presence of NLOS bias degrades the performances of SDP and iterative PPM approaches more than it does for MDS-MAP(P, R), the benefit will finally be outweighed by the negative impact from larger amounts of range estimation noise. For example, the localization error of MDS-MAP(P, R) is consistently larger than those of the other two methods with K_E larger than 0.0625, even in the 40%-NLOS scenario.

Now, we present an example where the anchors are not well distributed within the network. In particular, in the network shown in Fig. 6.21, four anchors are concentrated within a small area around the bottom left corner. As we can see, since the anchors are not well spread into the network, the closest-anchor initialization will render unlocalized nodes' initial solutions very close to each other. This will lead to a bad localization result shown in Fig. 6.21 (a). However, if we simply use random initialization for each unlocalized node, the adverse effect of the closest-anchor initialization can be alleviated in some degree, as shown Fig. 6.21 (b). We need to point out that with such a bad anchor deployment, SDP approach also suffers significant performance degradation as shown in Fig. 6.21 (d), while MDS-MAP(P, R) gives comparable performance, as shown in Fig. 6.21 (c), as our iterative PPM with random initialization.

6.5.2 Algorithm running time

So far, we have focused on localization accuracy. Another important aspect of collaborative position location algorithm is the algorithm complexity. In Table 6.1, we give typical algorithm running time for the three methods on an Intel 3.3 GHz CPU in Matlab R2008a. In general, the number of unlocalized nodes n and the average node connectivity, denoted by n_c, jointly determine the algorithm running time. Therefore, we compare the running times of the three methods for different combinations of N and n_c. The iterative PPM approach

only involves simple computation, and its complexity scales as $O(N\, n_c\, L)$, where L is the number of iterations for the iterative PPM approach to converge. The SDP approach, on the other hand, takes a much longer time to run since it has a complexity of $O(N^3)$. The MDS-MAP(P, R), although faster than the SDP, still needs considerably more time than our proposed approach and its complexity is $O(N\, n_c^3)$ for the MDS step and $O(N^3)$ for the global LS refinement. As observed from Table 6.1, our proposed iterative PPM method has a significantly shorter running time and scales considerably better for large networks. Note that the running time presented here is the total algorithm running time. Combined with our previous results on the localization error for different network shapes, our proposed iterative PPM is a very efficient and distributed method with excellent localization accuracy.

Table 6.1: Total algorithm running time (seconds) of iterative PPM, MDS-MAP(P, R) and SDP, obtained in Matlab R2008a on an Intel 3.3 GHz CPU, for different number of unlocalized nodes and average node connectivity.

# of nodes (N)	ave. node connectivity (n_c)	Iterative PPM	MDS-MAP(P, R)	SDP
50	12.6	0.77	2.59	2.11
50	25.0	0.97	6.27	10.7
100	13.4	1.42	5.92	13.3
100	25.8	1.45	24.6	89.5
200	12.8	4.05	15.9	114
200	24.8	4.69	76.0	1280
500	13.4	12.1	N/A[1]	3001
500	24.8	13.2	N/A[1]	N/A[2]

We have compared the localization accuracy of the three methods as well as their algorithm running time. Besides the huge complexity savings offered by iterative PPM, we can easily see that our proposed iterative PPM method provides the best performance if taking all three network topologies as a whole into account. Specifically, in the absence of NLOS, the iterative PPM outperforms SDP in all network topologies. Although it performs worse than MDS-MAP(P, R) in the case of small range estimation noise for randomly uniformly distributed network, it outperforms MDS-MAP(P, R) for irregular networks, especially the E-shaped network. In fact, the performance degradation of the MDS based methods has

[1]The MDS-MAP(P, R) solver failed since the computation is out of memory.

[2]The SDP solver failed since the computation is out of memory.

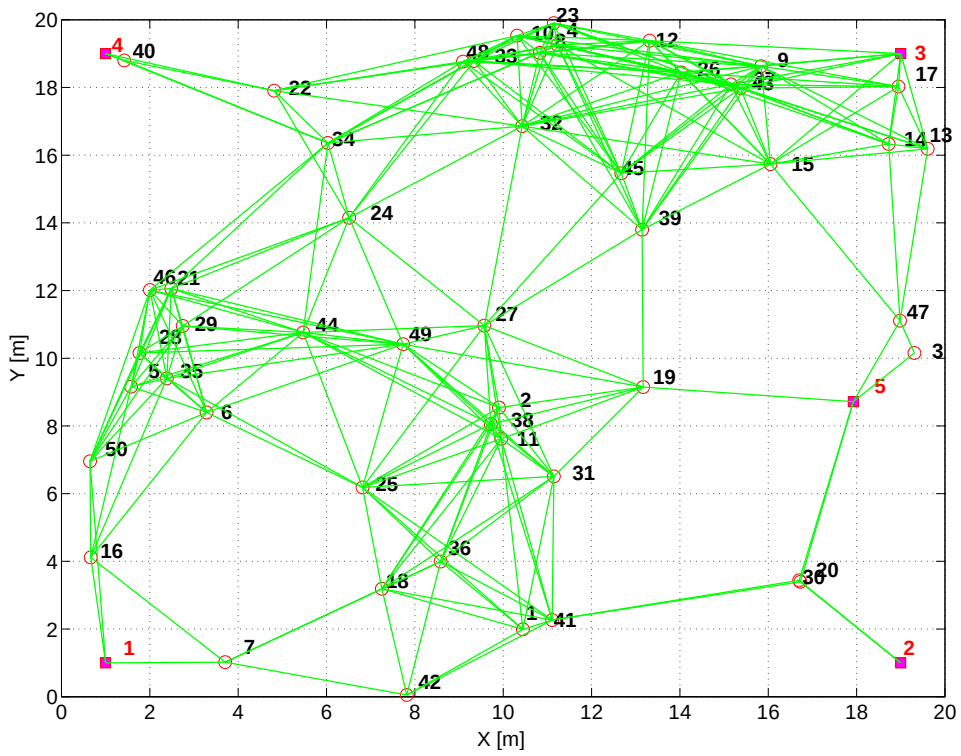
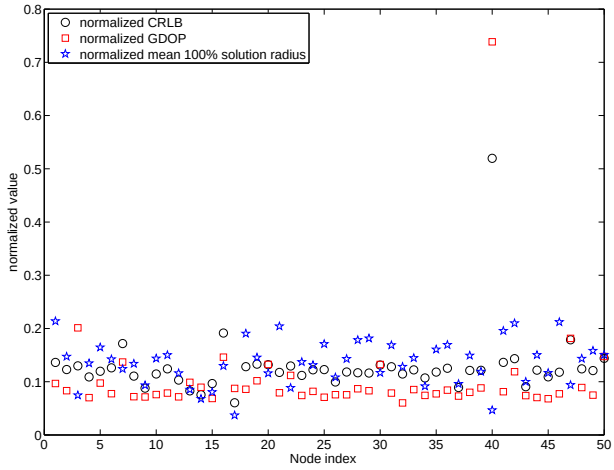


Figure 6.22: A network of 50 unlocalized nodes (circles) and 5 anchors (squares), where the solid lines represent node connectivity. We use this network to test our method to evaluate solution ambiguity and thus infer node localizability.

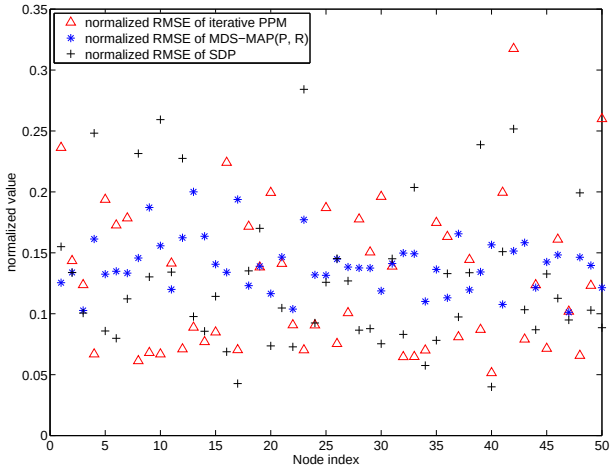
been well reported in existing literature. On the other hand, in the presence of NLOS, MDS-MAP(P, R) yields more robust performance than the iterative PPM and the SDP, almost the same as its NLOS-free cases, for all network topologies. However, the performance difference degrades as the amount of range estimation noise increases and finally the MDS-MAP (P, R) performs worse than the other two methods. This becomes more prominent for irregular network topologies, such as E-shaped or ring-shaped networks. Overall speaking, considering all possible network topologies, range estimation noise as well as the NLOS scenarios, our proposed iterative PPM provides the best performance.

6.5.3 Localizability and location coverage

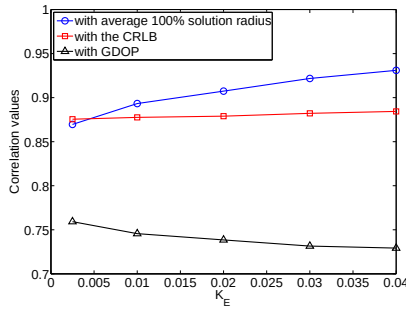
Now, we present some results of using our proposed iterative PPM technique to address the issue of node localizability in the presence of range estimation noise. In particular, due to



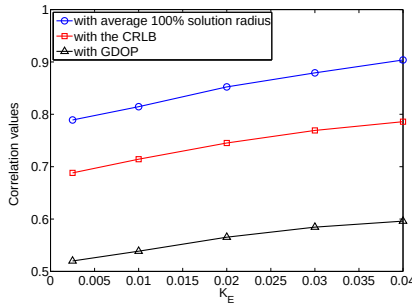
(a) Normalized solution radius, CRLB, GDOP



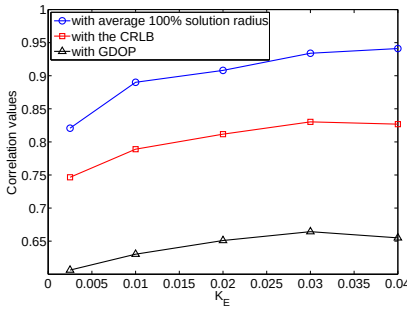
(b) Normalized mean localization error



(c) MDS-MAP(P, R)



(d) SDP



(e) Iterative PPM

Figure 6.23: (a) the normalized values of the mean 100% solution radius, the CRLB and the GDOP for all unlocalized nodes; (b) the normalized values of mean localization error of the itrative PPM, MDS-MAP(P, R) and SDP. The simulation parameters are $K_E = 0.04$ and $\beta = 2$. (c), (d) and (e): the correlation (dot product) between the normalized mean localization errors of the three methods and the normalized mean 100% solution radius, the normalized CRLB and the normalized GDOP, with respect to K_E.

the significant complexity savings from our proposed iterative PPM, we can afford to run the iterative PPM technique with multiple distinct initializations, each of which may or may not result in a different solution. Such a set of solutions naturally provides an approximation to the probability distribution of the location solution and serves as a characterization of the solution ambiguity. We follow the method of generating multiple initializations described in Section 6.4 and apply it to the randomly generated network in Fig. 6.22. Specifically, we run 100 noise realizations for the network with $K_E = 0.04$ and $\beta = 2$ for a pure LOS scenario. Using $M_p = 20$, i.e., 20 distinct initializations, we plot the mean 100% solution radius, the CRLB and the GDOP in Fig. 6.23 (a). In order to present the three results in a meaningful way, we normalized the three such that the squared sum of each is equal to one. Note that the GDOP differs from the CRLB only in the sense that it assumes equal noise variance for all range estimates. Clearly, the CRLB and the GDOP suggest that node 40 has a much larger error than other nodes, while our proposed metric, the mean 100% solution radius suggests the opposite. This is corroborated in Fig. 6.23 (b) which shows that the mean localization error of node 40 is indeed not too large for all three methods. On the other hand, both CRLB and GDOP do not realize that node 42 is close to having a flip ambiguity, since its neighbors are almost collinear. This flip ambiguity has been demonstrated by the fact that node 42 has very large errors in the case of iterative PPM and the SDP. In Fig. 6.23 (c), (d) and (e), we present the correlation values (essentially the dot product) between the normalized mean localization error and the three metrics, i.e., the mean 100% solution radius, the CRLB and the GDOP. The larger the value is, the better the metric statistically captures the relative error variation across different unlocalized nodes. As we can observe, the correlation values between the normalized mean localization error and the mean 100% solution radius is always larger than those of the other two metrics, for different K_E and across the three different methods, except only one case in Fig. 6.23 (c) with $K_E = 0.0025$. The GDOP is the poorest indicator, since it essentially assumes equal noise variance. These results suggest that the mean 100% solution radius, as an indicator of the solution uncertainty, better characterizes node localization error variation within the network due to a combined effect of node geometry and ranging noise.

In the above, we have shown that the *mean* 100% solution radius is a better indicator of the impact of node geometry and ranging noise on the solution quality. In practice, the concept of solution radius can be easily used to generalize the definition of location coverage. Recall that in Chapter 5, we investigated the location coverage for both the sequential LS estimator and the CQLP framework. However, whether or not a node is localizable is still determined based on whether it has sufficient number of localized nodes (could be anchors or virtual

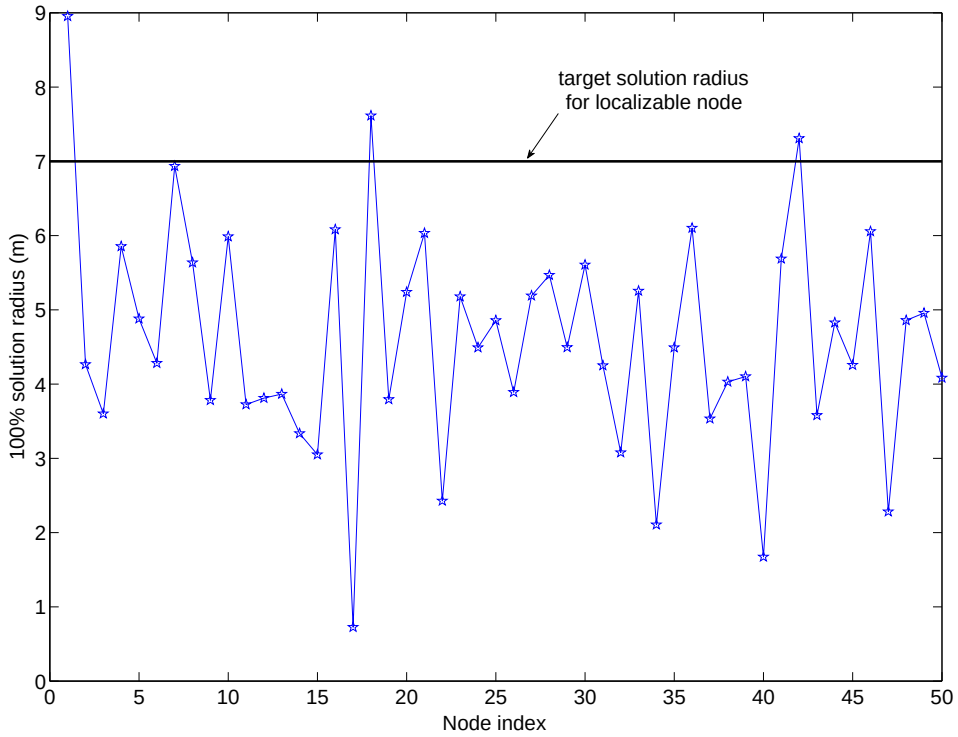


Figure 6.24: Using the 100% solution radius from one particular noise realization with $K_E = 0.04$ and $\beta = 2$ for the network in Fig. 6.22. If the target solution radius for nodes to be considered as localizable is 7 m, nodes 1, 18 and 42 will be considered as unlocalizable since they have a 100% solution radius larger than 7 m.

anchors) with which to communicate. Using the 100% solution radius, we can define node localizability for a given target solution radius. For example, as shown in Fig. 6.24, we can consider any node with a 100% solution radius smaller than 7 m as localizable, while others as unlocalizable, *i.e.*, nodes 1, 18 and 42. In doing so, the localizability is defined in a *soft* way. We believe that this is a more meaningful definition than conventional definition of localizability, which is purely based on node connectivity and ignores the effect of ranging noise and NLOS bias. Compared to the localizability derived from probabilistic localization, the computation involved in obtaining the solution radius is much less.

Finally, we emphasize that our proposed iterative PPM approach in general involves certain communication cost due to inter-node information exchange. Centralized approaches, on the other hand, also have communication overhead to collect all the information within the network and finally have to send the computed locations back to all the nodes. As mentioned

in [18], distributed approaches have on average similar communication cost as centralized approaches, while distributing the communication cost more evenly within the network. The algorithm running time presented in this work solely reflects the location computation time, excluding any communication latency which in practice has to be taken into account. However, such a comparison is beyond the scope of this work.

6.6 Summary

In this chapter, we presented an iterative and distributed numerical framework to address the issue of collaborative position location. Our proposed approach is computationally very efficient, while achieving comparable or better localization accuracy and NLOS robustness as well as robustness across different network shapes, compared to existing methods. In fact, one could argue that its performance, when considered as a whole, is better than the two most accurate techniques known in the art with substantially lower computational complexity. Finally, we show that the solution radius obtained from running iterative PPM with multiple distinct initializations serves as an indicator of the reliability of location solution. In the next chapter, we will incorporate a method for explicit NLOS bias mitigation.

Chapter 7

NLOS Mitigation for Collaborative Position Location

7.1 Motivation

As we mentioned earlier, NLOS propagation is a prominent phenomenon for the envisioned indoor position location or WSN applications. From Chapter 6, it has been observed that the presence of NLOS propagation can significantly degrade the localization accuracy of any collaborative position location method. Therefore, it is a necessity to develop methods to mitigate NLOS propagation.

In fact, there exist many approaches and algorithms for mitigating the adverse effect of NLOS propagation, but mainly in the context of cellular systems. For example, in [45], the authors proposed two NLOS mitigation algorithms for TDOA and TDOA/AOA position location schemes. The authors in [48] developed a linear programming (LP) framework to incorporate NLOS range estimates into TOA based position location without degrading the localization accuracy, based on the assumption that NLOS identification has been performed to certain accuracy. In [90], the authors proposed a technique utilizing the statistics about mean excess delay, first detected path power to subtract the statistical value of NLOS error from the measurement range. In [91], the authors proposed a weighted LS scheme based on NLOS identification using multipath channel statistics. Also, techniques based on Kalman filters [92] have have been developed for NLOS mitigation. A brief survey and comparison of various NLOS identification and mitigation techniques can be found in [93].

However, the major difficulty of directly applying state-of-the-art NLOS mitigation techniques to the problem of interest is the high computational complexity or additional requirements. We emphasize that the developed NLOS mitigation method is to be deployed for large number of low cost and low power wireless sensors. Consequently, it is not practical to use complicated methods developed in the context of cellular systems, which either significantly increase the computational complexity [44] or put additional constraints on the minimum number of LOS range estimates [48]. In that sense, the method we have developed in Chapter 5 is also less practical due to its possible use of the RSA method developed in [44] in order to handle the degenerate cases encountered in [48].

In view of these, in this chapter, we develop a simple yet effective NLOS mitigation method based on how much *a priori* knowledge of NLOS conditions we have. Our proposed NLOS mitigation is incorporated into our collaborative position location method developed in Chapter 6 and has scalable computational complexity depending on the utilization of the knowledge about NLOS conditions. Compared to existing methods developed for cellular systems, our proposed method does not involve nonlinear optimizations and has no additional requirements on the number of LOS range estimates.

7.2 Knowledge about the NLOS Propagation

To recall, we repeat our range estimate model defined in Chapter 2 as follows:

$$\begin{aligned} r_{ij} &= d_{ij} + n_{ij}, && \text{if it is LOS} \\ r_{ij} &= d_{ij} + n_{ij} + b_{ij}, && \text{if it is NLOS} \end{aligned}$$

where the very last term b_{ij} represents an unknown NLOS bias, which is positive and much larger than the noise standard deviation σ_{ij}. The most common way to statistically describe NLOS bias is to model b_{ij} as being uniformly distributed. Note that exponentially distributed NLOS bias is also sometimes observed [60] and some work has also used Rayleigh distributed NLOS bias [94].

In practice, it is possible that some knowledge about the NLOS conditions are either known *a priori* or collected online by examining the received signal statistics. For example, it is demonstrated in [49] that using the received signal statistics such as root-mean-square delay spread (RDS), NLOS links can be successfully identified with a reasonably high probability (about 90%). Furthermore, it is conceivable that an extensive measurement campaign can

provide some rough knowledge of the mean and the variance of the NLOS bias in certain propagation environments. Some may even assume that the complete statistics of the NLOS bias are known and derive the associated MLE [45, 36].

7.3 NLOS Mitigation Method

Despite the additional complexity associated with the corresponding signal processing, we believe NLOS identification is a practically useful step to mitigate the effect of NLOS propagations. Therefore, we will develop NLOS mitigation methods based on perfect NLOS identification and later on will evaluate the effect of imperfect NLOS identification. We will additionally consider the knowledge about mean, minimum and maximum NLOS bias and also examine the effect of inaccurate knowledge. We focus on incorporating the NLOS mitigation method into our collaborative position location method developed in Chapter 6 without significantly increasing computational complexity.

7.3.1 NLOS identification only

It has been demonstrated that NLOS bias is usually positive and much larger than range estimation noise [48]. In other words, if a range estimate is NLOS, it will appear to be much larger than the true distance, even in the presence of range estimation noise. This fact, coupled with NLOS identification, has been utilized in the method developed by [48] and demonstrated good performance. We adopt the same methodology to revise our iterative PPM method developed in Chapter 6. In particular, the update step in the iterative PPM will examine if a range estimate is NLOS or LOS. If it is NLOS, it further examines whether the current solution is within the ranging circle defined by the corresponding NLOS range estimate. If the current solution is outside the ranging circle, it proceeds as in the ordinary iterative PPM. However, if the current solution is already within the ranging circle, it will no longer contribute to the average of the projections. Mathematically, we define

$$\mathcal{N}(i) = \mathcal{N}_L(i) \cup \mathcal{N}_N(i) \tag{7.1}$$

where $\mathcal{N}_L(i)$ and $\mathcal{N}_N(i)$ are the sets of the ith node's neighbors with LOS and NLOS range estimates, respectively. We then define

$$\mathcal{N}^A_{\hat{\theta}_i}(i) = \mathcal{N}_L(i) \cup \{j | j \in \mathcal{N}_N(i), ||\hat{\theta}_i - \hat{\theta}_j|| \geq r_{ij}\} \tag{7.2}$$

The revised algorithm is given as follows.

Algorithm 7.1: Iterative PPM with NLOS mitigation, knowing NLOS identification:

- **Initialization:**
  The same as Algorithm 6.2 in Chapter 6.

- **Main loop:**
  While (any of F_i is equal to 0) {
  For $i = 1, 2, \ldots, n$; if $F_i = 0$
  $\hat{\theta}_i \leftarrow (1/|\mathcal{N}^A_{\hat{\theta}_i}(i)|) \sum_{j \in \mathcal{N}^A_{\hat{\theta}_i}(i)} P^{\text{col}}_{ij}(\hat{\theta}_i)$, and $\Phi_{i,l+1} = \Phi^{\text{col}}(\hat{\theta}_i)$;
  If $|\Phi_{i,l} - \Phi_{i,l+1}| < \delta$
  Let $W_i = W_i + 1$; If $W_i \geq \kappa$, Let $F_i = 1$;
  Otherwise
  Let $W_i = 0$;
  $l = l + 1$;
  }

where $\mathcal{N}^A_{\hat{\theta}_i}(i)$ is given by (7.2), which is essentially the set of the ith node's neighbors whose range estimates will participate in the update of $\hat{\theta}_i$. More specifically, the set of the ith node's neighbors with LOS range estimates will always participate, while any of its neighbors with NLOS range estimate will participate only if $||\hat{\theta}_i - \hat{\theta}_j|| \geq r_{ij}$ is satisfied, i.e., the current solution of θ_i is still outside the corresponding ranging circle. The subscript of $\mathcal{N}^A_{\hat{\theta}_i}(i)$ is to emphasize that the fact that $\mathcal{N}^A_{\hat{\theta}_i}(i)$ depends on the current estimate $\hat{\theta}_i$.

Essentially, the above method will treat NLOS range estimates differently from LOS range estimates. In particular, each unlocalized node examines its available range estimates. If a range estimate is NLOS, the node will compare the distance between itself and the corresponding neighbor computed using the current estimates of both nodes (if the neighbor happens to be an anchor, it will directly use the known anchor location) with the NLOS range estimate. If it is larger than the NLOS range estimate, the corresponding NLOS range estimate will contribute to the PPM step. If it is smaller, the NLOS range estimate is inactive and does not participate the PPM in the current iteration. This is an intuitively reasonble

way of using the knowledge that NLOS bias is positive and much larger than range estimation noise. Specifically, if the distance between a node and its neighbor is already smaller than the NLOS range estimate, it considers that the constraint defined by the NLOS range estimate has been satisfied and without further knowledge about NLOS bias, it is better not use it since there is an unknown bias contained in the range estimate. As we can see, the above method statistically tends not to use NLOS range estimates during the *later* iterations of the algorithm. By doing this, it is more likely to avoid the situations that NLOS range estimate degrades the localization performance.

7.3.2 NLOS identification plus the minimum NLOS bias

Another form of knowledge that may be available is an estimate of the minimum NLOS bias. The bound can be loose or even erroneous in practical situations. Denoting the estimated minimum NLOS bias as b_L, we can again define

$$\mathcal{N}_{\hat{\theta}_i}^{B}(i) = \mathcal{N}_L(i) \cup \{j | j \in \mathcal{N}_N(i), ||\hat{\theta}_i - \hat{\theta}_j|| \geq r_{ij} - b_L\}. \tag{7.3}$$

The projection onto the ranging circle is also revised in order to utilize the knowledge about the minimum NLOS bias, as follows

$$P_{ij}^{\text{col,NLOS-B}}(\hat{\theta}_i) = \begin{cases} \hat{\theta}_j + r_{ij}\frac{\hat{\theta}_i - \hat{\theta}_j}{||\hat{\theta}_i - \hat{\theta}_j||}, & j \in \mathcal{N}_L(i) \\ \hat{\theta}_j + (r_{ij} - b_L)\frac{\hat{\theta}_i - \hat{\theta}_j}{||\hat{\theta}_i - \hat{\theta}_j||}, & j \in \{k | k \in \mathcal{N}_N(i), ||\hat{\theta}_i - \hat{\theta}_k|| \geq r_{ik} - b_L\} \end{cases} \tag{7.4}$$

As we can see, the above modified projection is basically trying to correct the NLOS range estimates by subtracting the estimated minimum NLOS bias from each NLOS range estimate. This has an effect of mitigating NLOS bias, depending on the accuracy of the estimated minimum NLOS bias.

The corresponding position location algorithm is given by

Algorithm 7.2: Iterative PPM with NLOS mitigation, knowing NLOS identification and an estimate of the minimum NLOS bias:

- **Initialization:**
  The same as Algorithm 6.2 in Chapter 6.

- **Main loop:**

  While (any of F_i is equal to 0) {

  For $i = 1, 2, \ldots, n$; if $F_i = 0$

  $\hat{\theta}_i \leftarrow (1/|\mathcal{N}^B_{\hat{\theta}_i}(i)|) \sum_{j \in \mathcal{N}^B_{\hat{\theta}_i}(i)} P_{ij}^{\text{col,NLOS-B}}(\hat{\theta}_i)$, and $\Phi_{i,l+1} = \Phi^{\text{col}}(\hat{\theta}_i)$;

  If $|\Phi_{i,l} - \Phi_{i,l+1}| < \delta$

  Let $W_i = W_i + 1$; If $W_i \geq \kappa$, Let $F_i = 1$;

  Otherwise

  Let $W_i = 0$;

  $l = l + 1$;

  }

Again the estimate of the minimum NLOS bias can be connection-specific and the accuracy of the estimates affects the performance of the algorithm. Compared to Algorithm 7.1, the NLOS mitigation method outlined in Algorithm 7.2 is more aggressive in terms of using the knowledge about NLOS conditions. By subtracting the estimated minimum NLOS bias from each NLOS range estimates, it further provides correction to NLOS range estimates and thus is expected to perform better than Algorithm 7.1, as long as the estimated minimum NLOS bias is sufficiently accurate. In the results section, we will present simulation result to compare the two algorithms and demonstrate its efficacy.

7.3.3 NLOS identification plus the mean NLOS bias

In practice, it is also possible to have *a priori* knowledge of the mean NLOS bias in certain environments, although it may be inaccurate. With such knowledge, we can revise our algorithm accordingly. In particular, denoting the mean NLOS bias as $\bar{b}$, we subtract the mean NLOS bias from the identified NLOS range estimates when making the decision as to whether to include a particular NLOS range estimate into the solution update procedure. This is given by,

$$\mathcal{N}^C_{\hat{\theta}_i}(i) = \mathcal{N}_L(i) \cup \{k | k \in \mathcal{N}_N(i), ||\hat{\theta}_i - \hat{\theta}_k|| \geq r_{ik} - \bar{b}\}. \tag{7.5}$$

The projection onto the ranging circle is also revised, as follows

$$P_{ij}^{\text{col,NLOS-C}}(\hat{\theta}_i) = \begin{cases} \hat{\theta}_j + r_{ij}\frac{\hat{\theta}_i - \hat{\theta}_j}{||\hat{\theta}_i - \hat{\theta}_j||}, & j \in \mathcal{N}_L(i) \\ \\ \hat{\theta}_j + (r_{ij} - \bar{b})\frac{\hat{\theta}_i - \hat{\theta}_j}{||\hat{\theta}_i - \hat{\theta}_j||}, & j \in \{k | k \in \mathcal{N}_N(i), ||\hat{\theta}_i - \hat{\theta}_k|| \geq r_{ik} - \bar{b}\} \end{cases} \tag{7.6}$$

Compared to (7.4), the modified projection in (7.6) subtracts the estimated mean NLOS bias, in stead of the estimated minimum NLOS bias, from each NLOS range estimate. Considering that the actual NLOS bias may be larger or smaller than the mean NLOS bias, (7.6) may actually over- or under-estimate the NLOS bias contained in each individual NLOS range estimate. Therefore, its performance will be examined via simulations to see whether it on average performs better or worse than the previous two algorithms.

The corresponding algorithm is revised as follows.

Algorithm 7.3: Iterative PPM with NLOS mitigation, knowing NLOS identification and an estimate of the mean NLOS bias:

- **Initialization:**
  The same as Algorithm 6.2 in Chapter 6.

- **Main loop:**
  While (any of F_i is equal to 0) {
  For $i = 1, 2, \ldots, n$; if $F_i = 0$
  $\hat{\theta}_i \leftarrow (1/|\mathcal{N}_{\hat{\theta}_i}^C(i)|) \sum_{j \in \mathcal{N}_{\hat{\theta}_i}^C(i)} P_{ij}^{\text{col,NLOS-C}}(\hat{\theta}_i)$, and $\Phi_{i,l+1} = \Phi^{\text{col}}(\hat{\theta}_i)$;
  If $|\Phi_{i,l} - \Phi_{i,l+1}| < \delta$
  Let $W_i = W_i + 1$; If $W_i \geq \kappa$, Let $F_i = 1$;
  Otherwise
  Let $W_i = 0$;
  $l = l + 1$;
  }

We emphasize that the estimate of the mean NLOS bias may be inaccurate and it can be connection-specific. Compared to the two previous algorithms, Algorithm 7.3 differs

in the sense that it corrects NLOS range estimates even more aggressively. However, its performance may depend on the actual statistics of the NLOS bias. For example, if the distribution of the actual NLOS bias concentrates around its mean value, Algorithm 7.3 can be more effective.

7.4 Simulation Results

In this section, we present simulation results of our proposed NLOS mitigation techniques integrated into our iterative PPM method developed in Chapter 6 for collaborative position location, in terms of localization accuracy as well as the algorithm running time. To help understand the benefit of our NLOS mitigation method, we directly compared using the three networks in the results of Chapter 6. The three networks all have 200 unlocalized nodes and 20 anchors distributed over a $100 \times 100\,\mathrm{m}^2$ square area, forming a uniformly random network, an E-shape network and a ring-shape network, respectively. As for the localization accuracy, we again compute the mean localization error using the RMSE given by (2.2). For NLOS scenarios, we will consider different percentages of NLOS range estimates, i.e., 20%, 40%, 60% and 80%, corresponding to various degrees of NLOS conditions. The range estimation noise and NLOS bias models are as described earlier, i.e., zero-mean independent Gaussian distributed range estimation noise and uniformly distributed NLOS bias between $[B_{\min}, B_{\max}]$. To more accurately model range estimation noise, we set path loss exponents for LOS and NLOS range estimates differently, i.e., $\beta_{\mathrm{LOS}} = 2$ and $\beta_{\mathrm{NLOS}} = 3$. For our proposed iterative PPM, we use the same parameters $\delta = 10^{-4}$ and $\kappa = 10$, as in Chapter 6. First, we present simulation results of the proposed NLOS mitigation method with perfect NLOS identification, i.e., we know exactly which range estimates are LOS or NLOS. Later, we will examine the impact of imperfect NLOS identification on the performance of our proposed NLOS mitigation method. For notational simplicity, we define the following terms:

- **IPPM**: the iterative PPM approach without using any NLOS mitigation, as described in Chapter 6.
- **IPPM-NM(ID)**: the iterative PPM approach with NLOS mitigation based on NLOS identification only, as described in Section 7.3.1.
- **IPPM-NM(ID, min)**: the iterative PPM approach with NLOS mitigation based on NLOS identification plus an estimate of the minimum NLOS bias, as in Section 7.3.2.

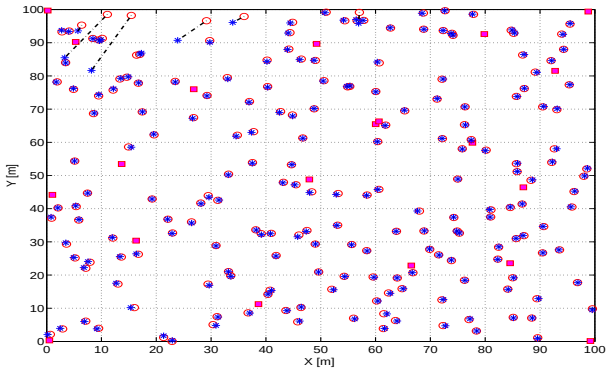
- **IPPM-NM(ID, mean)**: the iterative PPM approach with NLOS mitigation based on NLOS identification plus an estimate of the mean NLOS bias, as in Section 7.3.3.

Also, following the notation defined in Chapter 2, we use Ω defined in (2.1) to refer to the *network-average localization error* (averaged over all unlocalized nodes) for one particular noise and NLOS bias realization, and use $\overline{\Omega}$ defined in (2.2) to refer to the *mean localization error* average over all unlocalized nodes as well as different noise and NLOS bias realizations.

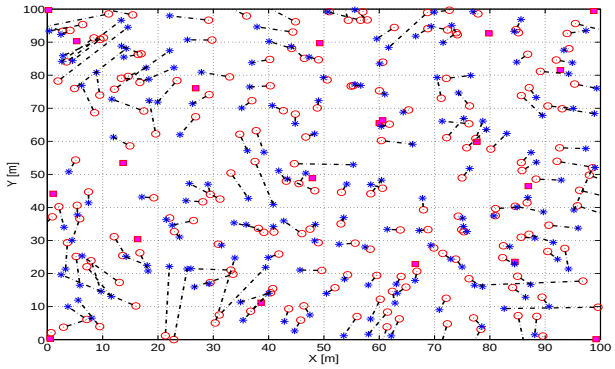
7.4.1 Perfect NLOS identification

In Fig. 7.1, we present the localization results of integrating our proposed NLOS mitigation methods into the IPPM to localize the randomly generated network shown in Fig. 6.9[1]. In particular, we present the localization results of using IPPM, IPPM-NM(ID), IPPM-NM(ID, min) and IPPM-NM(ID, mean) in the presence of 40% NLOS range estimates which are randomly selected from all the available range estimates, whether they are between unlocalized nodes and anchors, or among unlocalized nodes, for one particular range estimation noise and NLOS bias realization. The range estimation noise is generated with $K_E = 0.000625$. In the case of LOS range estimate, i.e., β_{LOS}=2, this corresponds to noise equivalent to 2.5% of the inter-node distance. For NLOS range estimate, i.e., β_{NLOS}=3, the noise becomes larger. The NLOS bias values are generated according to a uniform distribution with $B_{\min} = 4$ and $B_{\max} = 8$, corresponding to about 26% and 53% of the communication range R. In IPPM-NM(ID, min), we use $b_L = B_{\min}$ and in the IPPM-NM(ID, min), we use $\bar{b} = (B_{\min} + B_{\max})/2$. For the sake of comparison, we also present the localization result of IPPM in pure LOS scenario, i.e., there are no NLOS range estimates. The anchor locations are shown by solid squares and the true locations of the unlocalized nodes are shown by circles, while the estimated locations are denoted by the stars. The dashed lines connecting the true and the estimated locations quantize the localization errors. As shown in Fig. 7.1 (a), in a pure LOS scenario, IPPM performs very well and results in a network-average localization error equal to 1.8 m. However, in the presence of 40% NLOS range estimates, the performance of IPPM without any NLOS mitigation degrades and results in a network-average localization error equal to 6.8 m, as shown in Fig. 7.1 (b). On the other hand, if we apply IPPM-NM(ID), the network-average localization error is reduced almost by half to 3.5 m, as shown in Fig. 7.1 (c). Furthermore, using additional knowledge of the minimum NLOS bias,

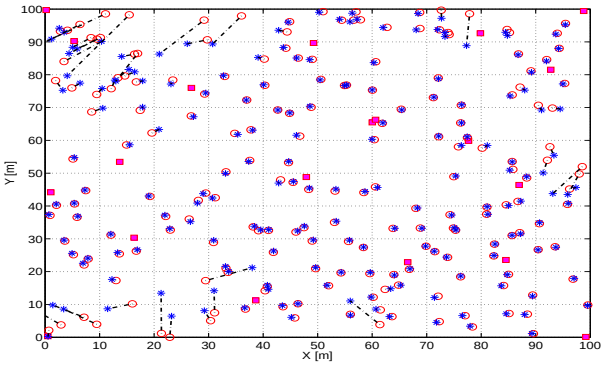
[1]To review, in this network the communication range is $R = 15$ m, the average node connectivity is 6.9 and the average number of connected anchors is 1.1.



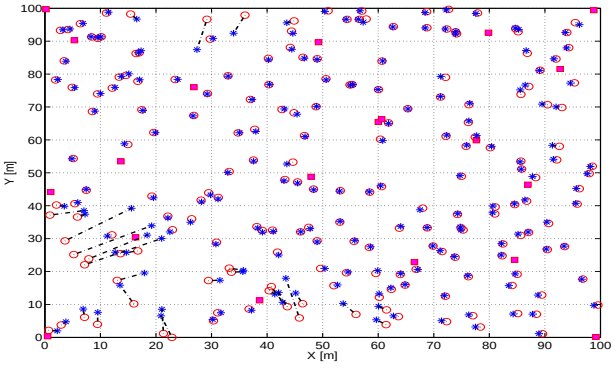
(a) IPPM in pure LOS, $\Omega = 1.8\,\mathrm{m}$



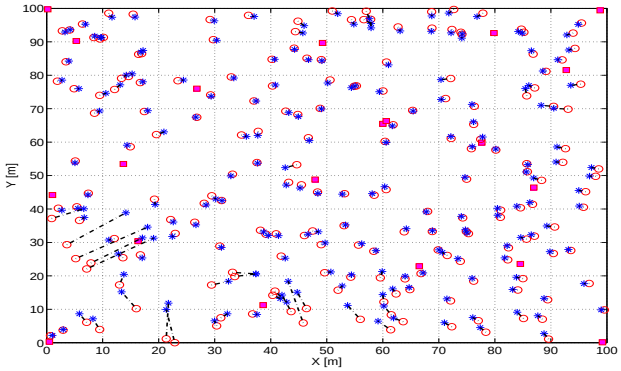
(b) IPPM, 40% NLOS, $\Omega = 6.8\mathrm{m}$



(c) IPPM-NM(ID), 40% NLOS, $\Omega = 3.5\mathrm{m}$



(d) IPPM-NM(ID, min), 40% NLOS, $\Omega = 2.8\mathrm{m}$



(e) IPPM-NM(ID,mean), 40% NLOS, $\Omega = 2.9\mathrm{m}$

Figure 7.1: Using our proposed NLOS mitigation methods to localize the network in Fig. 6.9 for one noise and 40% NLOS bias realization. The dash lines quantize the localization errors and Ω is the network-average localization error.

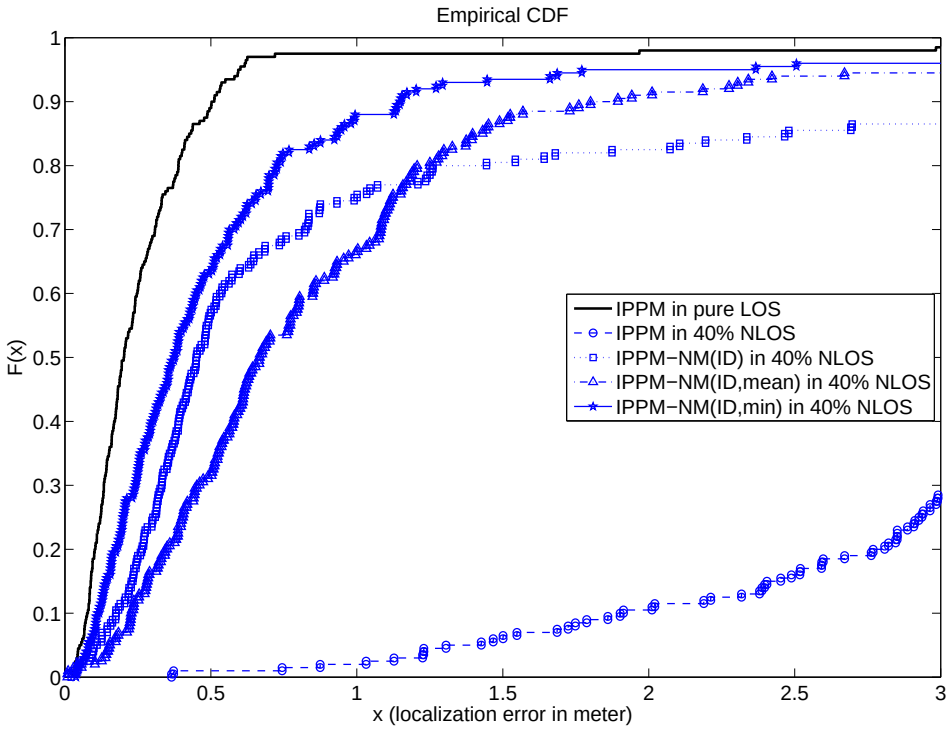
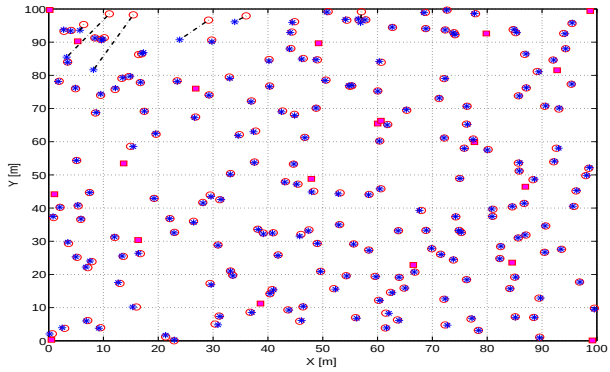


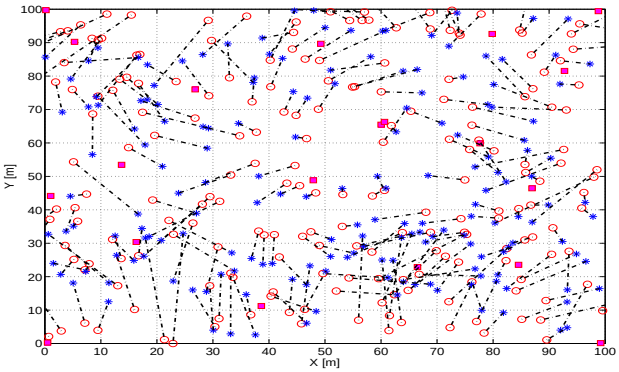
Figure 7.2: Cumulative distributiuon function of the localization errors of the methods in Fig. 7.1.

i.e., applying IPPM-NM(ID, min), the network-average localization error can be reduced even more to 2.8 m, as shown in Fig. 7.1 (d). IPPM-NM(ID, mean) provides slightly worse performance than IPPM-NM(ID, min), with a network-average localization error equal to 2.9 m, shown in Fig. 7.1 (e). All these results clearly demonstrate that with some knowledge about NLOS conditions, our proposed method is able to significantly improve the localization performance. More importantly, having further knowledge such as minimum NLOS bias or mean NLOS bias beyond just simple NLOS identification definitely helps the localization. In Fig. 7.2, we plot the CDF of the localization errors presented in Fig. 7.1. It demonstrates that IPPM-NM(ID, min) has the best performance and IPPM has the worse performance since it does not mitigate NLOS bias. Compared to IPPM-NM(ID), IPPM-NM(ID, mean) has the advantage of reducing large localization errors, leading a smaller mean localization error.

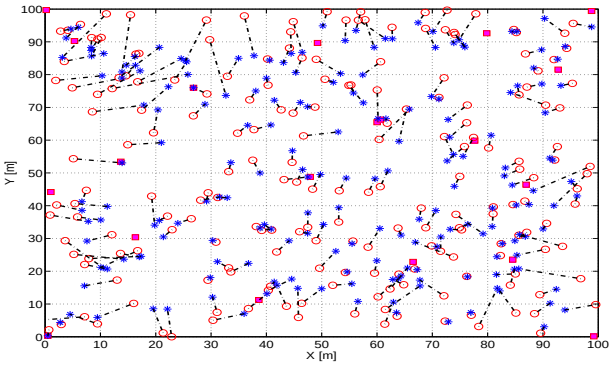
In Fig. 7.3, we present simulation results in the same setup as in Fig. 7.1. The only difference here is that we increase the percentage of NLOS range estimates to 80%. Considering the fact that the average node connectivity in the network shown in Fig. 6.9 is about 6.9, the



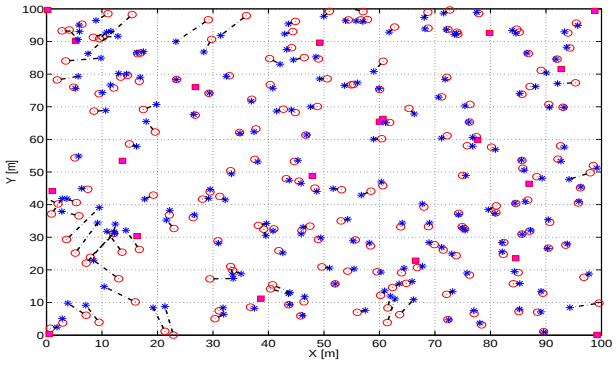
(a) IPPM in pure LOS, $\Omega = 1.8\,\mathrm{m}$



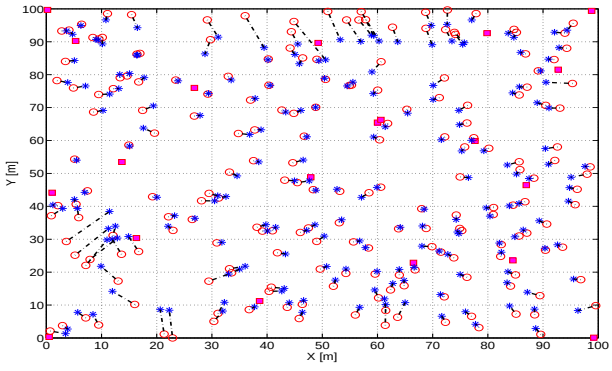
(b) IPPM, 80% NLOS, $\Omega = 12.0\mathrm{m}$



(c) IPPM-NM(ID), 80% NLOS, $\Omega = 6.0\mathrm{m}$



(d) IPPM-NM(ID, min), 80% NLOS, $\Omega = 3.0\mathrm{m}$



(e) IPPM-NM(ID,mean), 80% NLOS, $\Omega = 3.1\mathrm{m}$

Figure 7.3: Using our proposed NLOS mitigation methods to localize the network in Fig. 6.9 for one noise and 80% NLOS bias realization. The dash lines quantize the localization errors and Ω is the network-average localization error.

presence of 80% NLOS range estimates means that on average, each unlocalized node only has about 1.4 LOS range estimates to its neighbors. Such a situation indeed represents a very challenging NLOS condition and is of practical importance for the envisioned applications. As shown in Fig. 7.3 (b), without NLOS mitigation, the performance of IPPM degrades, as the network-average localization error increases from 1.8 m in pure LOS scenario to 12 m. IPPM-NM(ID) is able to reduce the network-average localization error to 6.0 m, as shown in Fig. 7.3 (c). With additional NLOS knowledge, IPPM-NM(ID, min) and IPPM-NM(ID, mean) are able to further reduce the error to 3.0 m and 3.1 m, as shown in Fig. 7.3 (d) and (e), respectively.

In Fig. 7.4, we plot the mean localization error of using the abovementioned NLOS mitigation methods versus the percentage of NLOS range estimates for the network in Fig. 6.9, averaged over 100 range estimation noise and NLOS bias realizations. For each tested percentage of NLOS range estimates, the corresponding number of NLOS range estimates are randomly selected from all the available range estimates in the network and are kept unchanged over different noise and NLOS bias realizations. The range estimation noise and the NLOS bias parameters remain the same as before. From Fig. 7.4, it is clearly demonstrated that without NLOS mitigation, IPPM suffers significantly performance degradation as the percentage of NLOS range estimates increases. In particular, in the pure LOS scenario, i.e., the percentage of NLOS range estimates is equal to zero, the IPPM has a mean localization error of 2.3 m, which increases to 12.8 m in the presence of 40% NLOS range estimates and 17.3 m in the presence of 80% NLOS range estimates. On the other hand, all three proposed NLOS mitigation methods are capable of reducing the mean localization error. For example, with only the knowledge of NLOS identification, IPPM-NM(ID) is able to reduce the mean localization error to 3.6 m in the presence of 40% NLOS range estimates and 7.4 m in the presence of 80% NLOS range estimates, respectively. IPPM(ID, mean) is able to further provide 0.5 m and 2.3 m error reduction in the two cases, respectively, compared to IPPM-NM(ID). IPPM(ID, min), on the other hand, offers the best performance in the sense that the mean localization error is reduced to 2.6 m and 4.7 m in the presence of 40% and 80% NLOS range estimates, respectively. In other words, IPPM-NM(ID, min) offers the best protection against NLOS propagation such that even with 80% NLOS range estimates, the localization error is only increased by 2.4 m compared to the pure LOS scenario, which is only 16% of the communication range $R = 15$ m. We believe this can be attributed to the fact that the minimum NLOS bias represents more accurate knowledge However, compared to the mean NLOS bias. Subtracting the mininum NLOS bias from NLOS range estimate is more trustworthy than subtracting the mean NLOS bias, since the actual NLOS bias may

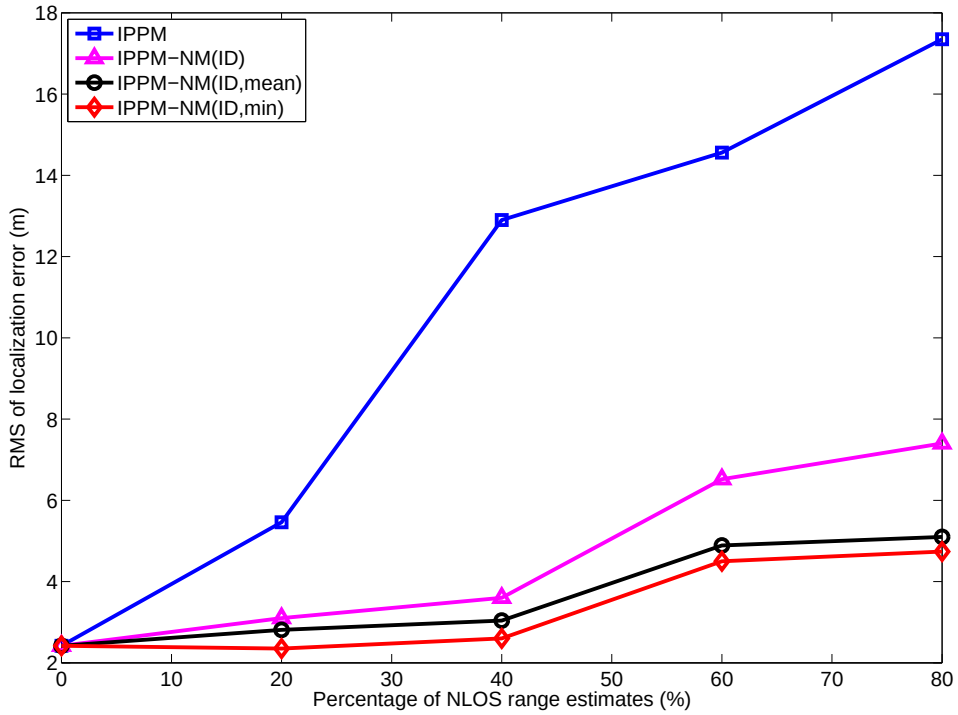
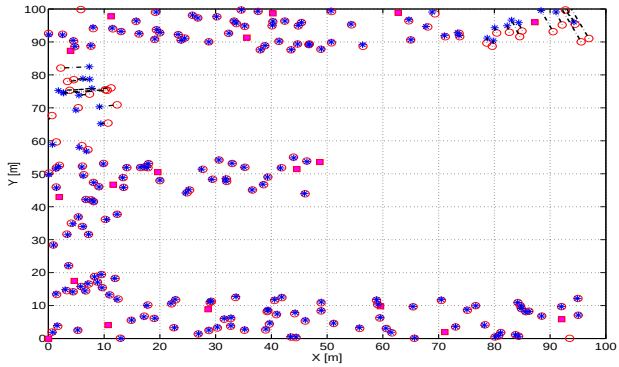


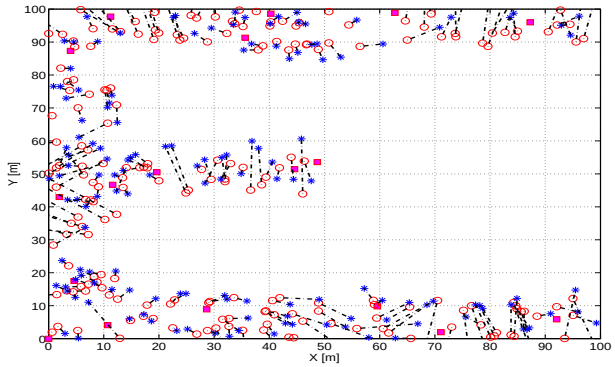
Figure 7.4: Mean localization error (averaged over 100 noise and NLOS bias realizations) versus the percentage of NLOS range estimates, when using our proposed NLOS mitigation methods to localize the network in Fig. 6.9, with NLOS bias uniformly distributed between $B_{\min} = 4\,\text{m}$ and $B_{\max} = 8\,\text{m}$.

be smaller or larger than the mean NLOS bias. We emphasize that these simulation results are based on the perfect knowledge about NLOS identification, the minimum NLOS bias and the mean NLOS bias.

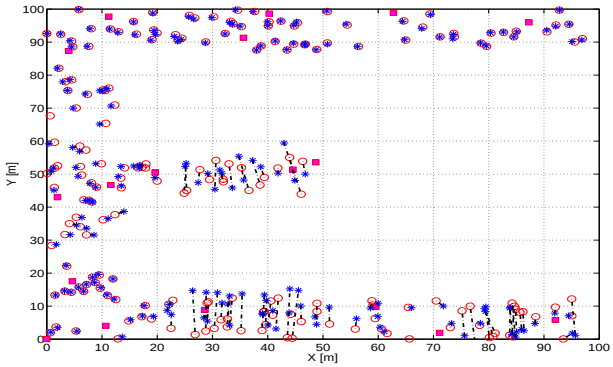
In Figs. 7.5 and 7.6, we give similar results for NLOS mitigation methods, applying on the E-shape and ring-shape networks in Fig. 6.13 and Fig. 6.17, respectively, with 40% NLOS range estimates, for one particular noise and NLOS bias realization. Other simulation parameters are the same as Fig. 7.1. We can observe that similar performance improvement can be achieved by using the proposed NLOS mitigation methods, which further demonstrates the efficacy of our proposed NLOS mitigation methods in different network topologies.



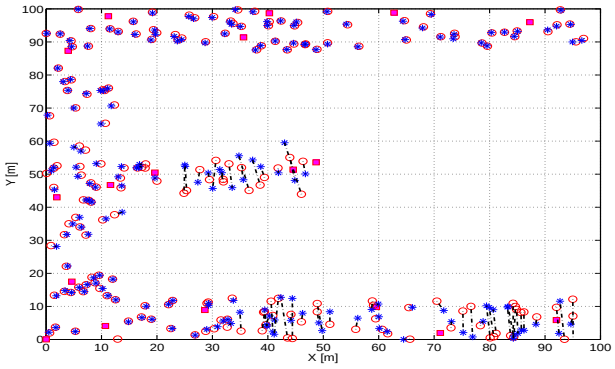
(a) IPPM in pure LOS, $\Omega = 2.2\,\mathrm{m}$



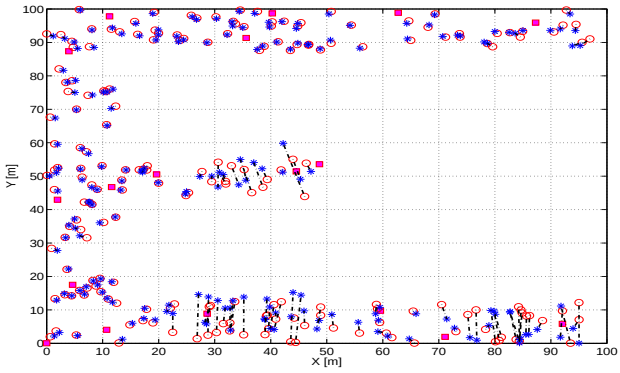
(b) IPPM, 40% NLOS, $\Omega = 7.3\mathrm{m}$



(c) IPPM-NM(ID), 40% NLOS, $\Omega = 4.2\mathrm{m}$

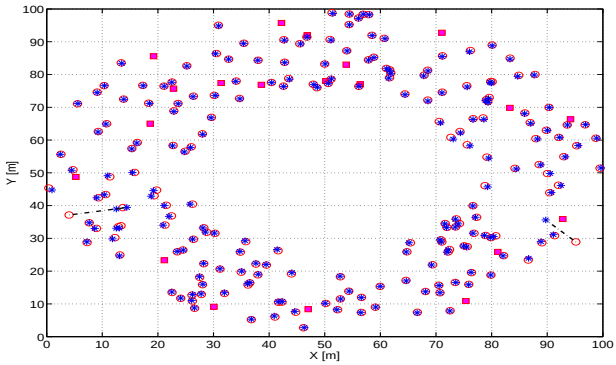


(d) IPPM-NM(ID, min), 40% NLOS, $\Omega = 3.7\mathrm{m}$

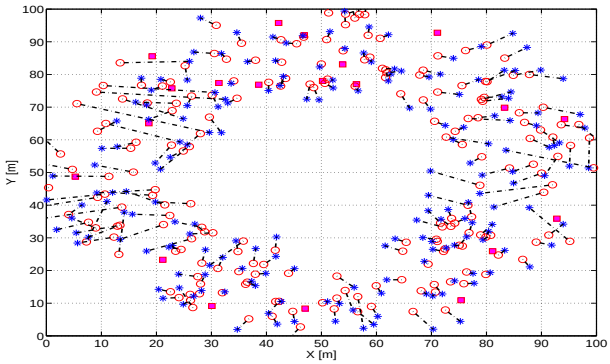


(e) IPPM-NM(ID,mean), 40% NLOS, $\Omega = 4.2\mathrm{m}$

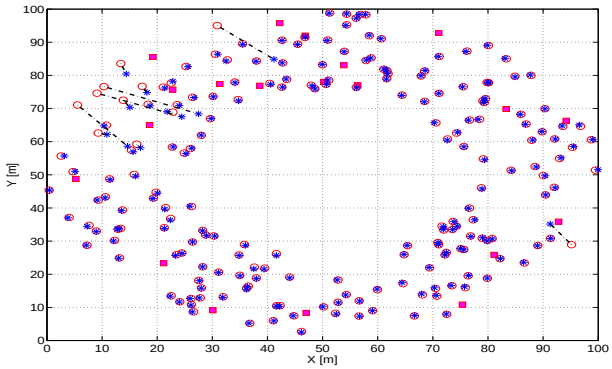
Figure 7.5: Using our proposed NLOS mitigation methods to localize the network in Fig. 6.13 for one noise and 40% NLOS bias realization. The dashed lines quantize the localization errors and Ω is the network-average localization error.



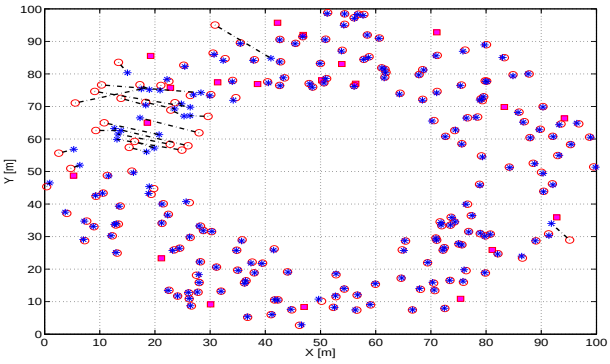
(a) IPPM in pure LOS, $\Omega = 1.0\,\mathrm{m}$



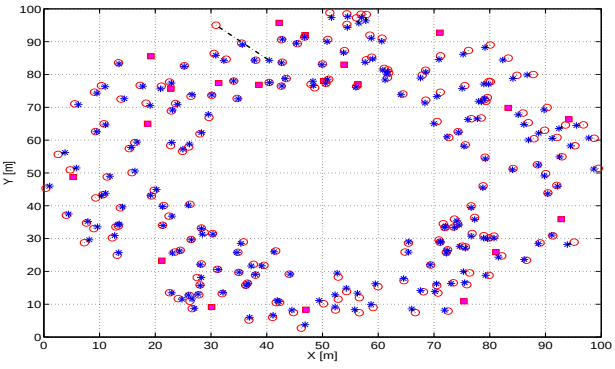
(b) IPPM, 40% NLOS, $\Omega = 6.1\mathrm{m}$



(c) IPPM-NM(ID), 40% NLOS, $\Omega = 2.4\mathrm{m}$



(d) IPPM-NM(ID, min), 40% NLOS, $\Omega = 3.1\mathrm{m}$



(e) IPPM-NM(ID,mean), 40% NLOS, $\Omega = 1.3\mathrm{m}$

Figure 7.6: Using our proposed NLOS mitigation methods to localize the network in Fig. 6.17 for one noise and 40% NLOS bias realization. The dashed lines quantize the localization errors and Ω is the network-average localization error.

7.4.2 Imperfect NLOS identification

Now, we investigate the impact of imperfect knowledge about the NLOS conditions on the performance of our proposed NLOS mitigation methods. Specifically, we first examine the effect of inaccurate NLOS identification. Later, we will examine the effect of imperfect estimation of the minimum or the mean NLOS bias. For the NLOS identification, as demonstrated in [49], using the received signal statistics such as RMS delay spread, the NLOS identification accuracy can be as high as about 90%. Therefore, in our simulation, we investigate two cases, i.e., 5% and 20% NLOS identification error, where the randomly-generated erroneous identifications can happen to either LOS or NLOS range estimates. In other words, even if all range estimates are LOS, i.e., in pure LOS scenario, there may still be some range estimates erroneously identified as NLOS range estimates. With erroneous NLOS identification, our proposed NLOS mitigation method will have some performance loss.

This is illustrated in Fig. 7.7, where there exists 5% NLOS identification error. Except for NLOS identification error, all other simulation parameters are the same as in Fig. 7.4. As we can see, all three NLOS mitigation methods have some performance loss in the presence of 5% NLOS identification error. For example, in the presence of 60% NLOS range estimates, the localization error of IPPM-NM(ID) increases by 0.7 m, while those of IPPM-NM(ID, min) and IPPM-NM(ID, mean) increase by 1.2 m and 0.5 m, respectively. With a smaller percentage of NLOS range estimates, the impact of imperfect NLOS identification is not significant, e.g., with 20% NLOS range estimates, the three methods perform almost the same as if there were no NLOS identification error. One interesting thing to note is that in the pure LOS scenario, i.e., 0% NLOS range estimates, because of the NLOS identification error, IPPM-NM(ID), IPPM-NM(ID, min) and the IPPM-NM(ID, mean) all perform worse than IPPM, meaning that in such cases, doing NLOS mitigation based on inaccurate knowledge hurts the performance. Another observation is that with NLOS identification error, IPPM-NM(ID, mean) suffers less performance loss than IPPM(ID, min) when the percentage of NLOS range estimates is high, e.g., at 80% NLOS range estimates, its resulting localization error is smaller than the IPPM-NM(ID, min). This can be explained by the fact, without a perfect NLOS identification, using the more aggressive approach of subtracting the mean NLOS bias from identified NLOS range estimates as in IPPM(ID, mean) on average can offset some performance loss resulted from erroneously subtracting the minimum NLOS bias from some LOS range estimates (misidentified as NLOS range estimates) as will happen in IPPM-NM(ID, min). Nevertheless, both IPPM-NM(ID, min) and IPPM-NM(ID, mean)

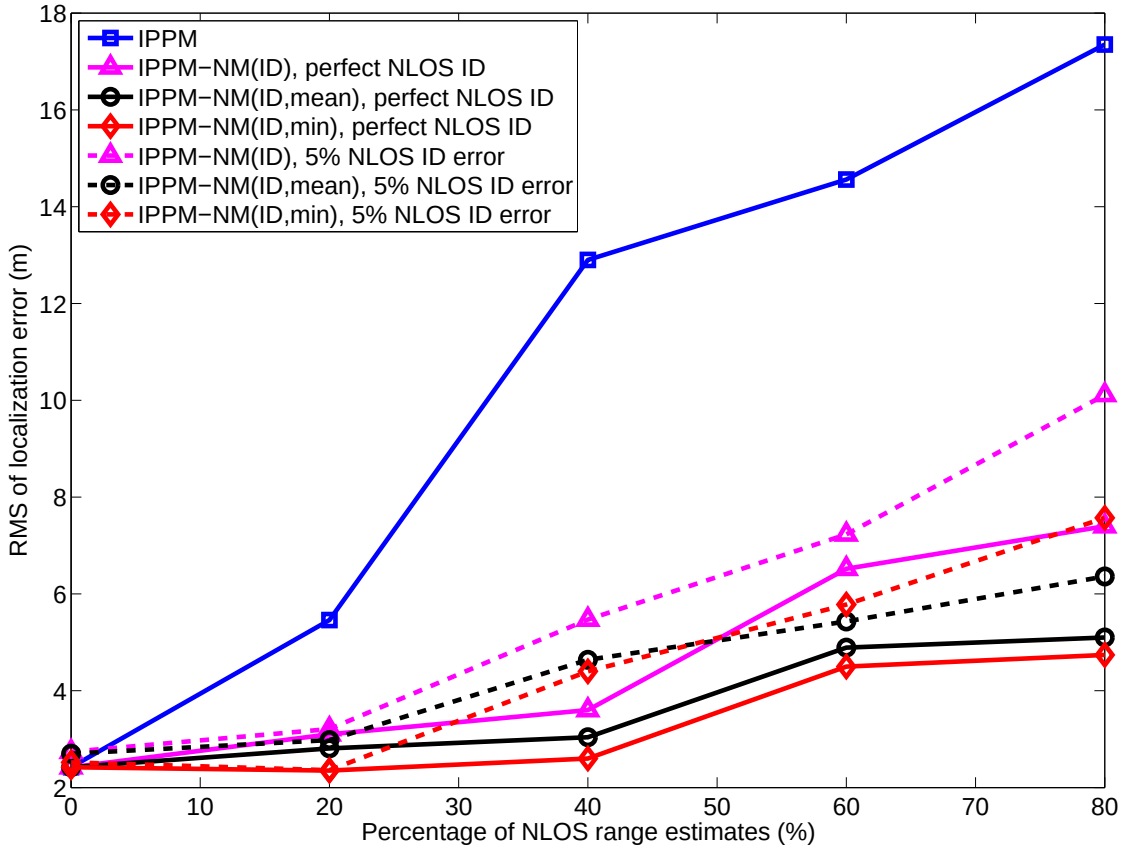


Figure 7.7: Mean localization error (averaged over 100 noise and NLOS bias realizations) versus the percentage of NLOS range estimates with 5% NLOS identification error, when using our proposed NLOS mitigation methods to localize the network in Fig. 6.9, with NLOS bias uniformly distributed between $B_{\min} = 4\,\text{m}$ and $B_{\max} = 8\,\text{m}$.

perform better than IPPM-NM(ID), even with NLOS identification error when there indeed exist NLOS range estimates. More importantly, all three NLOS mitigation methods still outperform IPPM, i.e., without any NLOS mitigation, even with only 20% NLOS range estimates.

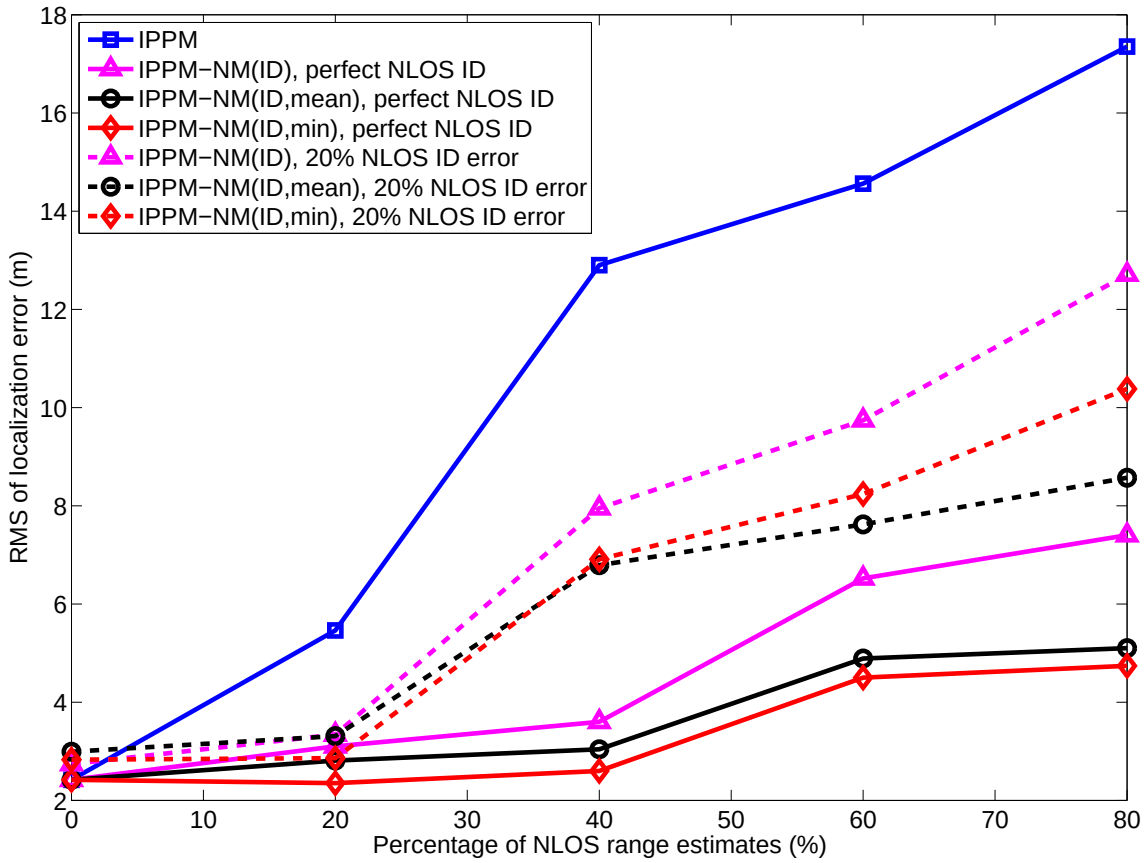


Figure 7.8: Mean localization error (averaged over 100 noise and NLOS bias realizations) versus the percentage of NLOS range estimates with 20% NLOS identification error, when using our proposed NLOS mitigation methods to localize the network in Fig. 6.9, with NLOS bias uniformly distributed between $B_{\min} = 4\,\text{m}$ and $B_{\max} = 8\,\text{m}$.

In Fig. 7.8, we simulate the case where there is 20% NLOS identification error, which represents very bad NLOS identification. All other simulation parameters remain the same as Fig. 7.7. Compared to Fig. 7.7, we easily observe that with more NLOS identification error, the performances of all three NLOS mitigation methods degrade and the degradation becomes more prominent when the percentage of NLOS range estimates increases. For example, with 80% NLOS range estimates and 20% NLOS identification error, the localization errors of IPPM-NM(ID), IPPM-NM(ID, min) and IPPM-NM(ID, mean) are 12.7 m, 10.3 m and 8.6 m, respectively, larger than the localization error of 10.1 m, 7.6 m and 7.4 m of the three methods, respectively, in the presence of only 5% NLOS as in Fig. 7.7. Again, in Fig. 7.8, we observe that in the pure LOS scenario, the three NLOS mitigation methods with NLOS identification error perform worse than IPPM, where IPPM-NM(ID) suffers the least

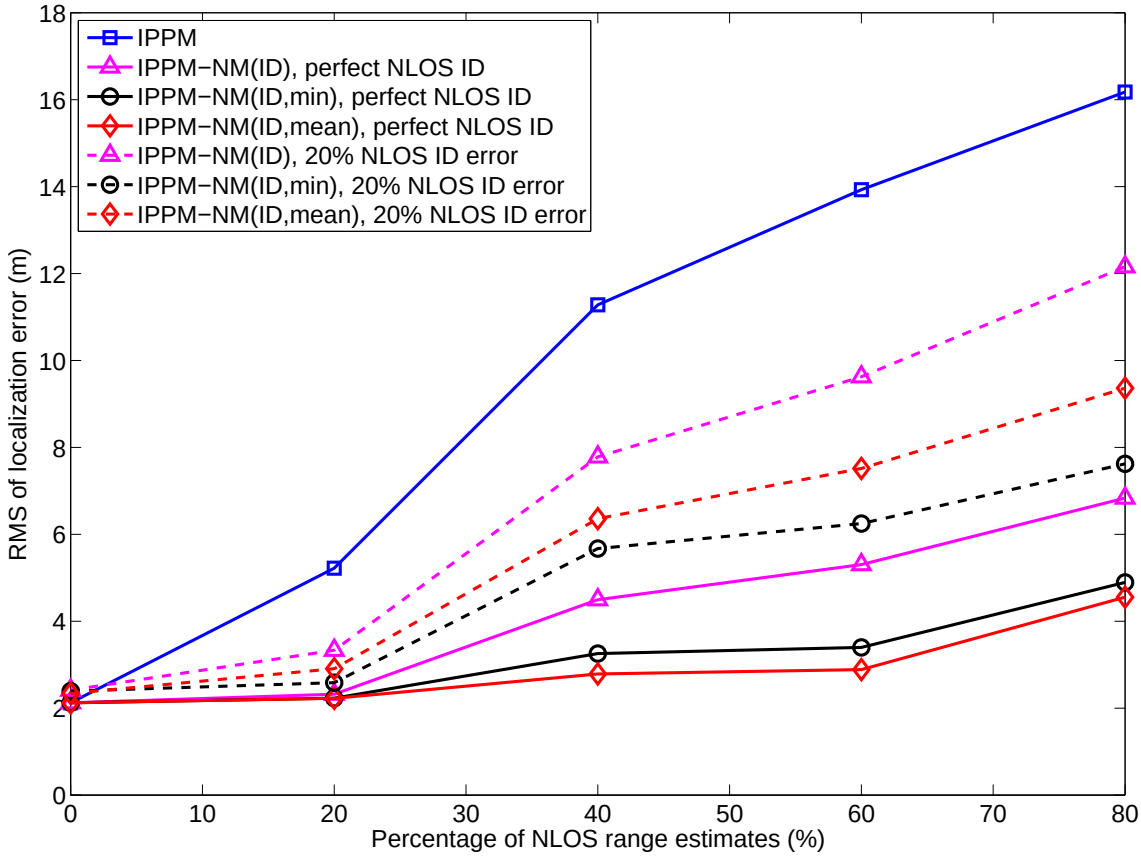


Figure 7.9: Simulation results similar to Fig. 7.8, for the E-shape network in Fig. 6.13

performance loss. In addition, IPPM-NM(ID, min) initially performs better than IPPM-NM(ID, mean) when the percentage of NLOS range estimates is low, but performs worse than IPPM-NM(ID, mean) as the percentage of NLOS range estimates is larger than 40%. Nevertheless, even with 20% NLOS identification error, all three NLOS mitigation methods still provide significant performance improvement over IPPM. An example is that with 80% NLOS range estimates and 20% NLOS identification error, even the simplest IPPM-NM(ID) method is able to reduce the localization error from 17.5 m down to 12.7 m.

In Figs. 7.9 and 7.10, we give the corresponding results for the E-shape and the ring-shape networks as shown in Fig. 6.13 and Fig. 6.17, respectively. As we can see, the performances of all three NLOS mitigation methods demonstrate a similar trend as in Fig. 7.8. For example, in the E-shape network and at 80% NLOS range estimates, IPPM suffers significant performance degradation, leading to a mean localization error of 16 m. However, IPPM-NM(ID), even with 20% NLOS identification error, is able to reduce the mean localization

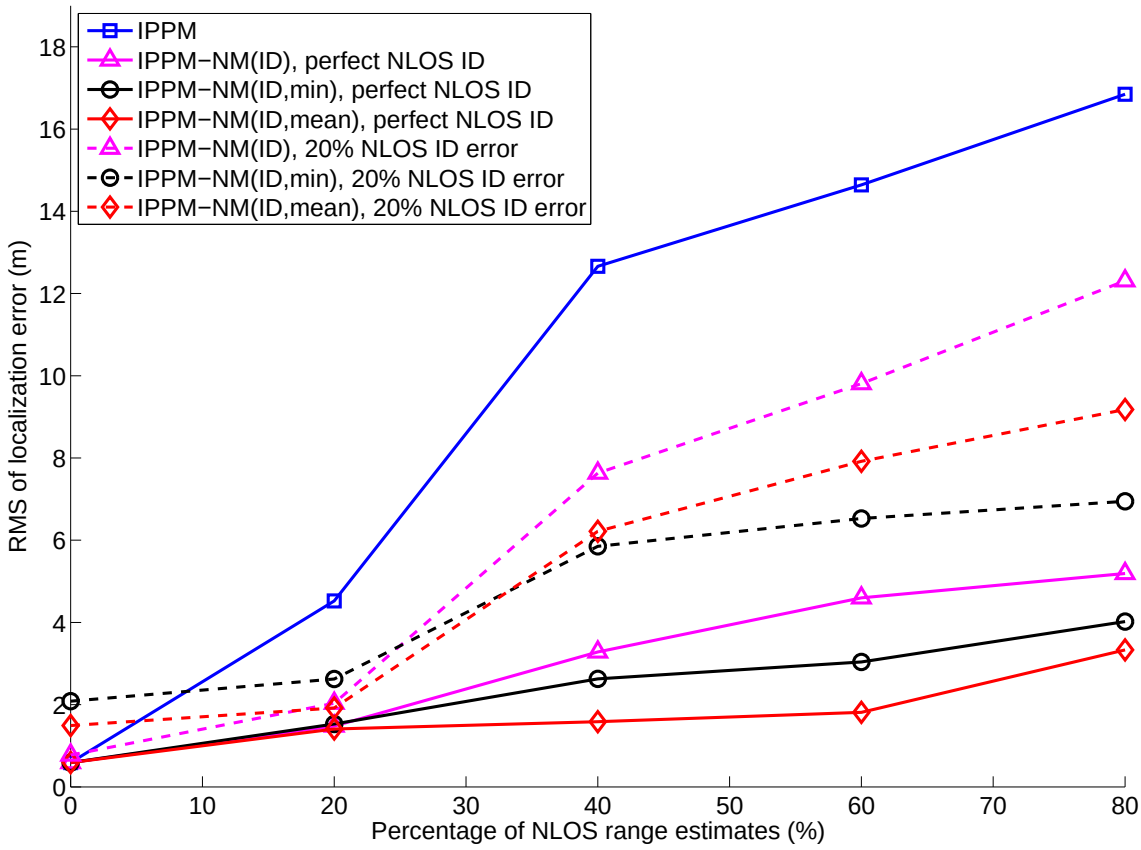


Figure 7.10: Simulation results similar to Fig. 7.8, for the ring-shape network in Fig. 6.17

error down to 12 m. With 20% NLOS identification error, IPPM-NM(ID, mean) gives the best performance, providing a mean localization error of only 7.8 m. Similar observations can be made for the ring-shape network. The results in these two figures suggest that our proposed NLOS mitigation methods are especially effective in irregularly shaped networks.

7.4.3 Imperfect NLOS identification plus inaccurate estimate of min and mean NLOS bias

The above are based on the accurate estimates of the mean and the minimum NLOS bias. In practice, these can be obtained by measurement campaign or estimated based on received signal statistics. In any case, these values can be inaccurate. Now we examine the impact of erroneous estimate of the minimum and the mean NLOS bias on our proposed NLOS

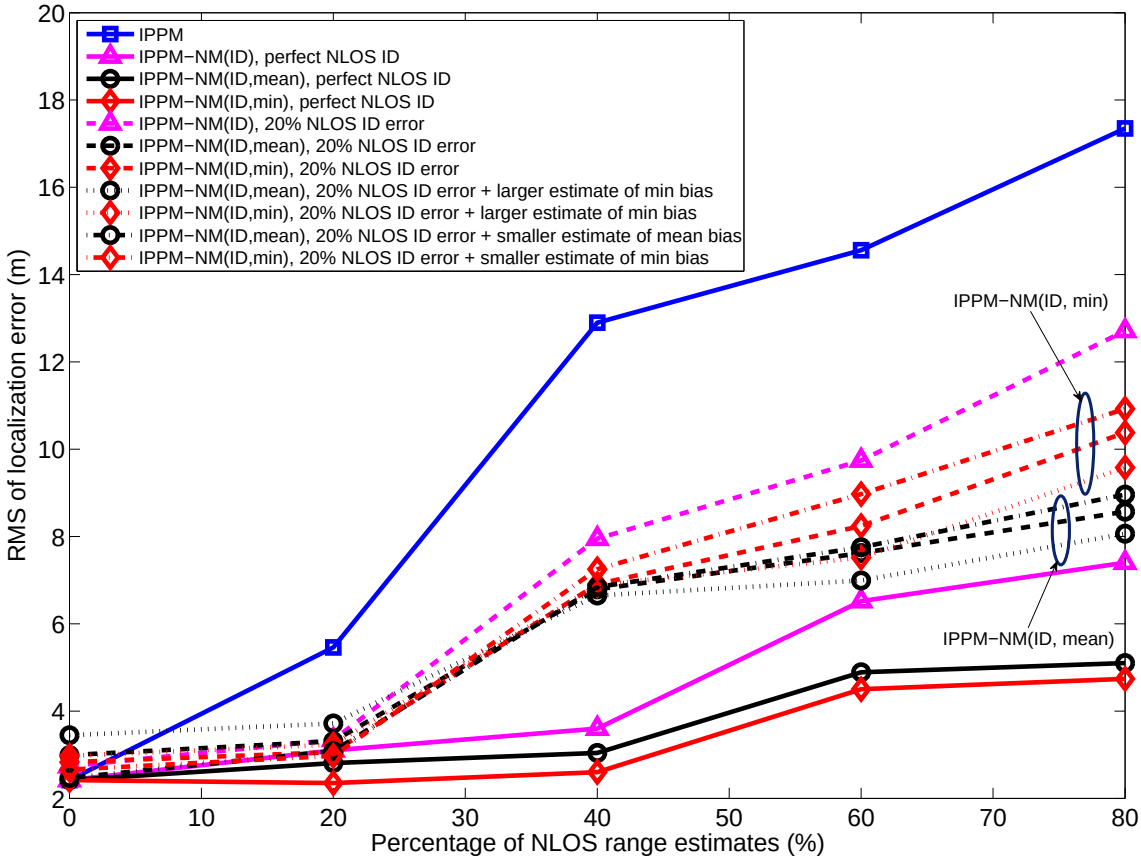


Figure 7.11: Simulation results of applying the proposed NLOS mitigation techniques to Fig. 6.9, in the presence of 20% NLOS identification error and larger or smaller estimate of the mean and the minimum NLOS bias.

mitigation techniques. In particular, we present simulation results of our proposed NLOS mitigation techniques in two cases: (1) the estimated values of the mean and the minimum NLOS bias are larger than their true values; (2) the estimated values of the mean and the minimum NLOS bias are larger than their true values. These two cases represent different estimates regarding the severity of the NLOS conditions in the deployment area. In Fig. 7.11, we present simulation results for applying the proposed NLOS mitigation techniques to Fig. 6.9, with 20% NLOS identification error. For case (1), we use $b_L = B_{\min} + 2$ and $\bar{b} = \frac{1}{2}(B_{\min} + B_{\max}) + 2$. For case (2), we use $b_L = B_{\min} - 2$ and $\bar{b} = \frac{1}{2}(B_{\min} + B_{\max}) - 2$. All other simulation parameters remain the same as before, i.e., $B_{\min} = 4$ and $B_{\max} = 8$. For reference purposes, we also plot the results when there is no NLOS identification error as solid lines. As we can see Fig. 7.11, with a larger estimate of either the mean or the minimum NLOS bias, the performances of IPPM-NM(ID, mean) and IPPM-NM(ID, min) in

the presence of 20% NLOS identification error become worse when the percentage of NLOS range estimates is smaller. However, their performances becomes better than those with accurate estimates when the percentage of NLOS range estimates is larger. For example, with 80% NLOS range estimates, 20% NLOS identification error and no error in the estimate of the mean and the minimum NLOS bias, IPPM-NM(ID, mean) and IPPM-NM(ID, min), shown by the dashed lines, lead to mean localization errors of 8.6 m and 10.4 m, respectively. However, with larger estimate of the mean and the minimum NLOS bias, IPPM-NM(ID, mean) and IPPM-NM(ID, min), as shown by the dotted lines, lead to mean localization errors of 8.0 m and 9.6 m, respectively. On the other hand, if there is less than 20% NLOS range estimates, meaning that NLOS condition is less severe, IPPM-NM(ID, mean) and IPPM-NM(ID, min) perform worse with a larger estimate of the mean and the minimum NLOS bias estimate. If the estimates of the mean and the minimum NLOS bias are smaller than their true values, we observe the opposite performance trend. As shown by the dash-dot lines in Fig. 7.11, IPPM-NM(ID, mean) and IPPM-NM(ID, min) perform better than those with accurate estimates of the mean and the minimum NLOS bias when the percentage of NLOS range is low, i.e., less than or equal to 20%. On the other hand, IPPM-NM(ID, mean) and IPPM-NM(ID, min) perform worse if the percentage of NLOS range estimates becomes larger. For example, at 80% NLOS range estimates with 20% NLOS identification error and smaller estimates of the mean and the minimum NLOS bias, IPPM-NM(ID, mean) and IPPM-NM(ID, min) result in mean localization errors of 8.9 m and 10.9 m, larger than their mean localization errors of 8.6 m and 10.4 m, respectively, in the case when the estimates are accurate. Overall, we can see that, in the presence of NLOS identification error, when the percentage of NLOS range estimates is large, i.e., severe NLOS condition and there is imperfect NLOS identification, being more aggressive in terms of the estimates of the mean and the minimum NLOS bias is helpful. However, in the presence of NLOS identification error, if the percentage of NLOS range estimates is low, i.e., less severe NLOS condition, being more conservative is beneficial.

7.5 Summary

In this chapter, we proposed three NLOS mitigation methods based on different levels of knowledge about NLOS conditions, which are incorporated into our iterative PPM approach developed in Chapter 6. We demonstrated that the proposed NLOS mitigation methods can significantly reduce the performance degradation caused by the presence of NLOS range

estimates. In addition, we show that even with NLOS identification error as large as 20%, the simplest NLOS mitigation method, which does not require further knowledge about NLOS bias statistics, is still able to reduce the localization error almost by half, as compared to not having NLOS mitigation. Considering the computational complexity and the performance in the presence of NLOS range estimates, our iterative PPM approach combined with the proposed NLOS mitigation techniques is both practical and effective.

Chapter 8

Conclusions

Position location for wireless networks in harsh environments has been a challenging research issue and the concept of collaborative position location has been proven to have the capability to improve both location coverage and localization accuracy. However, as stated in Chapter 1, there are several research problems that have not been or only partially dealt with by existing work. We now restate them here and summarize our contributions in tackling these research problems.

8.1 Research summary

The first and very fundamental question is: What is the optimal performance of collaborative position location? In particular, the authors in [24] formulated the maximum likelihood estimator (MLE) for collaborative position location. However, a closed-form expression for the solution is not known. Existing work either resorted to numerical approximations or relaxation techniques. In both cases, the solutions obtained are not necessarily the globally optimal solution to the original MLE. In our work, we developed a branch-and-bound/reformulation linearization technique (BB/RLT) framework to solve the MLE for distance-based collaborative position location with guaranteed solution optimality. Despite the high computational complexity, the MLE solution obtained by our BB/RLT framework can be used as a theoretical performance benchmark for any collaborative position location scheme. In particular, we demonstrated that the CRLB for position location with ill-conditioned node geometry is no longer a meaningful performance benchmark, while the

MLE solution obtained by the BB/RLT framework remains a practically meaningful performance limit. In this sense, we believe our work fills a hole in the existing work on the fundamental performance limit of collaborative position location. In addition to the optimal performance, another theoretical work we did is that we derived a new CRLB based on a distance-dependent noise variance model, which tells us that having the additional knowledge of the noise variance model has the potential to improve the localization accuracy.

On the practical side, the design of a computationally efficient collaborative position location scheme with good localization accuracy across different propagation conditions and network topologies has been a challenging issue. Using the paradigm of sequential localization is scalable and computationally efficient. In fact, in our proposed CQLP approach, we adopted the the sequential localization technique. However, we demonstrated that the localization accuracy of position location schemes using sequential localization degrades quickly as the number of nodes in the network is very large, due to the famous problem of error propagation. This has motivated us as well as many other researchers to resort to concurrent position location methods to alleviate the problem of error propagation. Existing work with good localization accuracy across different network topologies has shown to be computationally very expensive and does not scale well as the number of nodes in the network increases, e.g., the SDP approach. In this work, we designed a distributed collaborative position location method based on the concept of the parallel projection method. Compared to existing methods, our proposed approach has a few advantages. First, it significantly reduces the computational complexity and has much better scalability as the number of nodes in the network and the node connectivity increase. Second, it achieves comparable and often better localization accuracy, as compared to the two best existing methods, when considering various possible network topologies and propagation conditions. Furthermore, based on multiple distinct initial solutions, our proposed method can return multiple solutions that capture the effect of propagation and node geometry on the location solution, serving as a probabilistic approximation to the possible location, which tells about the solution reliability and localizability.

Last but most importantly, existing works on collaborative position location for wireless networks have ignored the issue of non-line-of-sight (NLOS) propagation or assumed that it is a separate issue and can be handled by directly borrowing the idea of NLOS mitigation used for cellular systems. We argue that the methods designed for cellular systems in general require complicated computation or have additional requirements which generally cannot be satisfied in the applications of interest and thus render existing works unsuitable. To deal

with this, we designed a simple and effective NLOS mitigation method which handles NLOS range estimates according to the available knowledge about NLOS conditions. The proposed method is incorporated into our proposed distributed collaborative position location method and involves little additional complexity, while showing its efficacy in mitigating the adverse effect of NLOS range estimates on the localization accuracy. Even with imperfect knowledge of NLOS conditions, it still achieves considerable performance improvement compared to not using any NLOS mitigation. Our method differs from existing work in the sense that it takes NLOS mitigation into account, while designing a computationally efficient collaborative position location scheme.

Overall, our work in this dissertation makes contributions to and provides insight into both theoretical and practical aspects of collaborative position location for wireless networks in harsh environments.

8.2 Future work

In our work, we have primarily focused on the position location algorithm design. It would be beneficial to use more realistic models for range estimation noise which take practical factors such as noise correlation among nearby nodes, range estimation bias caused by the clock synchronization error or round-trip propagation time measurements. Also, in this work, we compared the running time of different position location algorithms. In fact, another possible research area is to investigate the total latency, i.e., from ranging until final localization, which incorporates the delay caused by inter-node communication. The latency performance is of practical importance to the successful operation of a position location system.

Bibliography

[1] Y. Ko and N. H. Vaidya, "Location-aided routing (LAR) in mobile ad hoc networks," *Wirless Networks*, vol. 6, pp. 307–321, Jul. 2000.

[2] B. Rao and L. Minakakis, "Evolution of mobile location-based services," *Communications of the ACM, Special Issue: Mobile commerce opportunities and challenges*, vol. 46, no. 12, pp. 61–65, 2001.

[3] B. W. Parkinson, J. J. Spilker, Jr., P. Axelrad, and P. Enge., *The global positioning system : theory and applications*. American Institute of Aeronautics and Astronautics, 1995.

[4] J. Reed, K. Krizman, B. Woerner, and T. Rappaport, "An overview of the challenges and progress in meeting the E-911 requirement for location service," *IEEE Personal Communications Magazine*, vol. 36, pp. 30 – 37, Apr. 1998.

[5] C. Y. Chong and S. P. Kumar, "Sensor networks: Evolution, opportunities, and challenges," *Proceedings of IEEE*, vol. 91, pp. 1247–1256, Aug. 2003.

[6] N. Priyantha, A. Chakraborty, and H. Balakrishnan, "The Cricket location-support system," in *Proceedings of ACM International Conference on Mobile Computing and Networking (MobiCom)*, pp. 32–43, August 2000.

[7] P. Bahl and V. N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, vol. 2, pp. 775–784, Mar. 2000.

[8] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low-cost outdoor localization for very small devices," *IEEE Personal Communications, Special Issue on Smart Spaces and Environments*, vol. 7, pp. 28–34, October 2000.

[9] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher, "Range-free localization schemes for large scale sensor networks," in *Proceedings of ACM International Conference on Mobile Computing and Networking (MobiCom)*, pp. 81–95, 2003.

[10] N. Patwari, J. Ash, S. Kyperountas, A. O. Hero III, R. Moses, and N. Correal, "Locating the nodes: cooperative localization in wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 54–69, 2005.

[11] G. Mao, B. Fidan, and B. D. O. Anderson, "Wireless sensor network localization techniques," *International Journal of Computer and Telecommunications Networking*, vol. 51, no. 10, pp. 2529–2553, 2007.

[12] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proceedings of the IEEE*, vol. 97, pp. 427–450, Feb. 2009.

[13] S. Capkun, M. Hamdi, and J. Hubaux, "GPS-free positioning in mobile ad hoc networks," in *Proceedings of Hawaii International Conference on System Sciences*, pp. 3481–3490, 2001.

[14] J. Albowicz, A. Chen, and L. Zhang, "Recursive position estimation in sensor networks," in *Proceedings of IEEE International Conference on Network Protocols*, pp. 35–41, November 2001.

[15] L. Doherty, K. Pister, and L. E. Ghaoui, "Convex position estimation in wireless sensor networks," in *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, vol. 3, pp. 1655–1663, 2001.

[16] D. Niculescu and B. Nath, "Ad hoc positioning system (APS)," in *Proceedings of IEEE GLOBECOM*, pp. 2926–2931, 2001.

[17] C. Savarese and J. Rabaey, "Robust positioning algorithms for distributed ad-hoc wireless sensor networks," in *Proceedings of USENIX Annual Technical Conference*, pp. 317–327, 2002.

[18] A. Savvides, H. Park, and M. B. Srivastava, "The bits and flops of the n-hop multilateration primitive for node localization problems," in *Proceedings of International Workshop on Sensor Networks Application*, pp. 112–121, 2002.

[19] Y. Shang, W. Ruml, Y. Zhang, and M. P. J. Fromherz, "Localization from mere connectivity," in *Proceedings of Mobile Ad Hoc Networking and Computing (MobiHoc)*, pp. 201–212, 2003.

[20] Y. Shang, W. Ruml, Y. Zhang, and M. Fromherz., "Localization from connectivity in sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 15, pp. 961–974, November 2004.

[21] X. Ji and H. Zha, "Sensor positioning in wireless ad-hoc sensor networks using multidimensional scaling," in *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, vol. 4, pp. 2652–2661, March 2004.

[22] J. A. Costa, N. Patwari, and A. O. Hero III, "Distributed weighted-multidimensional scaling for node localization in sensor networks," *ACM Transactions on Sensor Networks*, vol. 2, no. 1, pp. 39–64, 2006.

[23] R. L. Moses, D. Krishnamurthy, and R. M. Patterson, "A self-localization method for wireless sensor networks," *EURASIP Journal on Applied Signal Processing*, pp. 348–358, 2003.

[24] N. Patwari, A. O. Hero III, M. Perkins, N. S. Correal, and R. J. O'Dea, "Relative location estimation in wireless sensor networks," *IEEE Transactions on Signal Processing*, vol. 51, no. 8, pp. 2137–2148, 2003.

[25] P. Biswas and Y. Ye, "Semidefinite programming for ad hoc wireless sensor network localization," in *Proceedings of International Symposium on Information Processing in Sensor Networks (IPSN)*, pp. 46–54, 2004.

[26] P. Biswas, T.-C. Lian, T.-C. Wang, and Y. Ye, "Semidefinite programming based algorithms for sensor network localization," *ACM Transactions on Sensor Networks*, vol. 2, no. 2, pp. 188–220, 2006.

[27] Z. Wang, S. Zheng, S. Boyd, and Y. Ye, "Further relaxations of the SDP approach to sensor network localization," tech. rep., Stanford University, May 2007. available at http://www.stanford.edu/ yyye/relaxationsdp9.pdf.

[28] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements," in *Proceedings of ACM conference on Embedded Networked Sensor Systems*, pp. 50–61, 2004.

[29] D. K. Goldenberg, P. Bihler, M. Cao, J. Fang, B. D. O. Anderson, A. S. Morse, and Y. R. Yang, "Localization in sparse networks using sweeps," in *Proceedings of ACM International Conference on Mobile Computing and Networking (MobiCom)*, pp. 110–121, Sept. 2006.

[30] P. Tseng, "Second-order cone programming relaxation of sensor network localization," *SIAM Journal on Optimization*, vol. 18, no. 1, pp. 156–185, 2006.

[31] S. Srirangarajan, A. Tewfik, and Z.-Q. Luo, "Distributed sensor network localization using SOCP relaxation," *IEEE Transactions on Wireless Communications*, vol. 7, pp. 4886–4895, December 2008.

[32] J. Liu, Y. Zhang, and F. Zhao, "Robust distributed node localization with error management," in *Proceedings of ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pp. 250–261, 2006.

[33] X. Li, "Collaborative localization with received-signal strength in wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 56, pp. 3807–3817, Nov. 2007.

[34] P. Zhang and M. Martonosi, "LOCALE: Collaborative localization estimation for sparse mobile sensor networks," *Proceedings of International Conference on Information Processing in Sensor Networks (IPSN)*, pp. 195–206, April 2008.

[35] O.-H. Kwon and H.-J. Song, "Localization through map stitching in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 19, pp. 93–105, Jan. 2008.

[36] A. J. Weiss and J. S. Picard, "Network localization with biased range measurements," *IEEE Transactions on Wireless Communications*, vol. 7, pp. 298–304, Jan. 2008.

[37] D. Dardari and A. Conti, "A sub-optimal hierarchical maximum likelihood algorithm for collaborative localization in ad-hoc network," in *Proceedings of IEEE Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, pp. 425–429, Oct. 2004.

[38] A. A. Kannan, G. Mao, and B. Vucetic, "Simulated annealing based wireless sensor network localization," *Journal of Computers*, vol. 1, no. 2, pp. 15–22, 2006.

[39] P.-J. Chuang and C.-P. Wu, "An effective PSO-based node localization scheme for wireless sensor networks," in *Proceedings of International Conference on Parallel and Distributed Computing, Application and Technology*, pp. 187–194, Dec. 2008.

[40] Q. Zhang, J. Wang, C. Jin, J. Ye, C. Ma, and W. Zhang, "Genetic algorithm based wireless sensor network localization," in *Proceedings of International Conference on Natural Computation*, vol. 1, pp. 608–613, Oct. 2008.

[41] S. Venkatraman and J. Caffery, Jr., "Statistical approach to non-line-of-sight BS identification," in *Proceedings of 5th International Symposium on Wireless Personal Multimedia Communications*, vol. 1, pp. 296–300, October 2002.

[42] S. Gezici, H. Kobayashi, and H. Poor, "Non-parametric non-line-of-sight identification," in *Proceedings of IEEE 58th Vehicular Technology Conference*, vol. 4, pp. 2544–2548, October 2003.

[43] S. Venkatraman and J. Caffery, "Hybrid TOA/AOA techniques for mobile location in non-line-of-sight environments," in *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC). 2004*, vol. 1, Mar. 2004.

[44] S. Venkatraman, J. J. Caffery, and Y. H. Ryeol, "A novel TOA location algorithm using LOS range estimation for NLOS environments," *IEEE Transactions on Vehicular Technology*, vol. 53, no. 5, pp. 1515–1524, 2004.

[45] L. Cong and W. Zhuang, "Nonline-of-sight error mitigation in mobile location," *IEEE Transactions on Wirelss Communications*, vol. 4, no. 2, pp. 560–573, 2005.

[46] K. Pahlavan, F. O. Akgul, M. Heidari, A. Hatami, J. M. Elwell, and R. D. Tingley, "Indoor geolocation in the absence of direct path," *IEEE Wireless Communications*, vol. 13, pp. 50–58, Dec. 2006.

[47] Y. Qi, H. Kobayashi, and H. Suda, "Analysis of wireless geolocation in a non-line-of-sight environment," *IEEE Transactions on Wireless Communications*, vol. 5, pp. 672–681, March 2006.

[48] S. Venkatesh and R. M. Buehrer, "NLOS mitigation in UWB location-aware networks using linear programming," *IEEE Transactions on Vehicular Technology*, vol. 56, pp. 3182–3198, Sept. 2007.

[49] S. Venkatesh and R. M. Buehrer, "NLOS identification in ultra-wideband systems based on received signal statistics," *IEE Proceeding on Microwaves, Antennas and Propagation: Special Issue on Antenna Systems and Propagation for Future Wireless Communications*, vol. 1, pp. 1120–1130, Dec. 2007.

[50] T. Eren, D. K. Goldenberg, W. Whiteley, Y. R. Yang, A. S. Morse, B. D. O. Anderson, and P. N. Belhumeur, "Rigidity, computation, and randomization in network localization," in *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, vol. 4, pp. 2673–2684, Mar. 2004.

[51] D. K. Goldenberg, A. Krishnamurthy, W. C. Maness, Y. Yang, A. Young, A. S. Morse, and A. Savvides, "Network localization in partially localizable networks," in *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, vol. 1, pp. 313–326, Mar. 2005.

[52] J. Aspnes, T. Eren, D. K. Goldenberg, A. S. Morse, W. Whiteley, Y. R. Yang, B. D. O. Anderson, and P. N. Belhumeur, "A theory of network localization," *IEEE Transactions on Mobile Computing*, vol. 5, no. 12, pp. 1663–1678, 2006.

[53] A. T. Ihler, J. W. Fisher III, R. L. Moses, and A. S. Willsky, "Nonparametric belief propagation for self-localization of sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 4, pp. 809–819, 2005.

[54] D. Marinakis and G. Dudek, "Probabilistic self-localization for sensor networks," in *Proceedings of the AAAI National Conference on Artificial Intelligence*, pp. 780–785, July 2006.

[55] Y. Zhang and H. Dai, "Dynamic self-calibration in collaborative wireless networks using belief propagation with gaussian particle filtering," in *Proceedings of 41st Annual Conference on Information Sciences and Systems*, pp. 771–776, Mar. 2007.

[56] R. Huang and G. V. Záruba, "Incorporating data from multiple sensors for localizing nodes in mobile ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 6, pp. 1090–1104, September 2007.

[57] D. McCrady, L. Doyle, H. Forstrom, T. Dempsey, and M. Martorana, "Mobile ranging using low-accuracy clocks," *IEEE Transactions on Microwave Theory and Techniques*, vol. 48, pp. 951–958, Jun 2000.

[58] S. Venkatesh, *The Design and Modeling of Ultra-Wideband Position-Location Networks*. Ph.D. Thesis, Virginia Tech, 2007.

[59] J.-Y. Lee and R. A. Scholtz, "Ranging in a dense multipath environment using an UWB radio link," *IEEE Journal on Selected Areas in Communications*, vol. 20, pp. 1677–1683, December 2002.

[60] B. Alavi and K. Pahlavan, "Modeling of the distance error for indoor geolocation," in *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, vol. 1, pp. 668–672, March 2003.

[61] T. Jia and R. Buehrer, "On the optimal performance of collaborative position location," *IEEE Transactions on Wireless Communications*, vol. 9, pp. 374–383, January 2010.

[62] J. Nie, "Sum of squares method for sensor network localization," *Computational Optimization and Applications*, vol. 43, no. 2, pp. 151–179, 2009.

[63] L. Li and T. Kunz, "Cooperative node localization using nonlinear data projection," *ACM Transactions on Sensor Networks*, vol. 5, no. 1, pp. 1–26, 2009.

[64] T. Jia and R. M. Buehrer, "A collaborative quasi-linear programming framework for ad hoc sensor localization," in *Proceedings of IEEE Wireless Communications and Networking Conference*, pp. 2379–2384, April 2008.

[65] S. Severi, G. Abreu, G. Destino, and D. Dardari, "Understanding and solving flip-ambiguity in network localization via semidefinite programming," in *Proceedings of IEEE GlobeCom*, Dec. 2009.

[66] Z. Yang, Y. Liu, and X.-Y. Li, "Beyond trilateration: On the localizability of wireless ad-hoc networks," in *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, pp. 2392–2400, April 2009.

[67] S. Lederer, Y. Wang, and J. Gao, "Connectivity-based localization of large scale sensor networks with complex shape," in *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, pp. 789–797, April 2008.

[68] S. Basagni, M. Battelli, M. Iachizzi, C. Petrioli, and S. Masoud, "Limiting the propagation of localization errors in multi-hop wireless network," in *IEEE Pervasive Computing and Communications Workshops*, pp. 6–11, Mar. 2006.

[69] R. M. Buehrer, S. Venkatesh, and T. Jia, "Mitigation of the propagation of localization error using multi-hop bounding," in *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 3009–3014, Apr. 2008.

[70] J. J. Caffery, "A new approach to the geometry of TOA location," in *Proceedings of IEEE Vehicular Technology Conference*, vol. 4, pp. 1943–1949, 2000.

[71] F. Chan, H. C. So, and W.-K. Ma, "A novel subspace approach for cooperative localization in wireless sensor networks using range measurements," *IEEE Transactions on Signal Processing*, vol. 57, pp. 260–269, Jan. 2009.

[72] C. Chang and A. Sahai, "Estimation bounds for localization," in *Proceedings of IEEE Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, pp. 415–424, Oct. 2004.

[73] Y. Qi and H. Kobayashi, "Cramer-rao lower bound for geolocation in non-line-of-sight environment," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 3, pp. 2473–2476, 2002.

[74] H. L. V. Trees, *Detection, Estimation, and Modulation Theory, Part I.* Wiley-Interscience, 2001.

[75] B. Ananthasubramaniam and U. Madhow, "Cooperative localization using angle of arrival measurements in non-line-of-sight environments," in *ACM workshop on Mobile entity localization and tracking in GPS-less environments*, pp. 117–122, 2008.

[76] R. Parker and S. Valaee, "Cooperative vehicle position estimation," in *IEEE International Conference on Communications*, pp. 5837–5842, 2007.

[77] F. Mendivil, R. Shonkwiler, and M. C. Spruill, "Analysis of random restart and iterated improvement for global optimization with application to the traveling salesman problem," *Journal of Optimization Theory and Applications*, vol. 124, no. 2, pp. 407–433, 2005.

[78] I. Guvenc, C.-C. Chong, and F. Watanabe, "Analysis of a linear least-squares localization technique in LOS and NLOS environments," in *Proceedings of IEEE Vehicular Technology Conference, Spring*, pp. 1886–1890, April 2007.

[79] Y. Qi, T. Asai, H. Yoshino, and N. Nakajima, "On geolocation in ill-conditioned BS-MS layouts," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 697–700, March 2005.

[80] S. O. Dulman, A. Baggio, P. J. M. Havinga, and K. G. Langendoen, "A geometrical perspective on localization," in *Proceedings of ACM workshop on Mobile Entity Localization and Tracking in GPS-less environments*, pp. 85–90, 2008.

[81] A. O. Hero, J. A. Fessler, and M. Usman, "Exploring estimator bias-variance tradeoffs using the uniform CR bound," *IEEE Transactions on Signal Processing*, vol. 44, no. 8, pp. 2026–2041, 1996.

[82] Y. C. Eldar, "Minimum variance in biased estimation: bounds and asymptotically optimal estimators," *IEEE Transactions on Signal Processing*, vol. 52, no. 7, pp. 1915–1930, 2004.

[83] J. Liu, Y. Hou, Y. Shi, H. Sherali, and S. Kompella, "On the capacity of multiuser MIMO networks with interference," *IEEE Transactions on Wireless Communications*, vol. 7, pp. 488–494, February 2008.

[84] H. D. Sherali and W. P. Adams, *A reformulation-linearization-technique for solving discrete and continuous nonconvex problems*. Boston MA: Kluwer Academic Publishing, 1999.

[85] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming: theory and algorithms*. John Wiley & Sons, Inc., Hoboken, New Jersey, 2006.

[86] P. L. Combettes, "Signal recovery by best feasible approximation," *IEEE Transactions on Image Processing*, vol. 2, pp. 269–271, Apr. 1993.

[87] P. L. Combettes, "Inconsistent signal feasibility problems: least-squares solutions in a product space," *IEEE Transactions on Signal Processing*, vol. 42, pp. 2955–2966, Nov. 1994.

[88] M. A. Diniz-Ehrhardt and J. M. Martinez, "A parallel projection method for overdetermined nonlinear system equations," *Numerical Algorithms*, pp. 241–262, 1993.

[89] J. N. Ash and R. L. Moses, "On optimal anchor node placement in sensor localization by optimization of subspace principal angles," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2289–2292, 2008.

[90] M. Heidari, F. O. Akgul, and K. Pahlavan, "Identification of the absence of direct path in indoor localization systems," in *Proceedings of IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pp. 1–6, Sept. 2007.

[91] I. Guvenc, C.-C. Chong, F. Watanabe, and H. Inamura, "NLOS identification and weighted least-squares localization for UWB systems using multipath channel statistics," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, p. 36, 2008.

[92] C.-D. Wann and C.-S. Hsueh, "NLOS mitigation with biased kalman filters for range estimation in UWB systems," in *Proceedings of IEEE Region 10 Conference TENCON*, pp. 1–4, 2007.

[93] J. Khodjaev, Y. Park, and A. S. Malik, "Survey of NLOS identification and error mitigation problems in UWB-based positioning algorithms for dense environments," *Annals of Telecommunications*, 2008.

[94] K. Yu and Y. J. Guo, "NLOS error mitigation for mobile location estimation in wireless networks," in *Proceedings of IEEE 65th Vehicular Technology Conference*, pp. 1071–1075, April 2007.