On the Optimal Performance of Collaborative Position Location

Tao Jia, *Student Member, IEEE,* and R. Michael Buehrer, *Senior Member, IEEE*

***Abstract*—In this paper, we investigate the optimal performance of collaborative position location. In particular, we develop a branch-and-bound (BB) solution search strategy, coupled with the reformulation linearization technique (RLT), to solve the maximum likelihood estimation (MLE) problem for collaborative position location, which is in general a nonlinear and nonconvex optimization problem. Compared with existing work which has only approximately solved the MLE problem, our approach is guaranteed to produce the $(1-\epsilon)$-optimal solution to the MLE for arbitrarily small ϵ. With a guaranteed optimal solution to the MLE, we show that for some node geometries in non-collaborative position location, which can be viewed as a special case of collaborative position location, the Cramer-Rao lower bound (CRLB) for an unbiased estimator is no longer a meaningful performance benchmark. We demonstrate that the time-of-arrival (TOA) based MLE is in general a biased estimator and it sometimes has a mean square error (MSE) smaller than the CRLB, and thus can serve as a more practical performance benchmark. Finally, we compare the MLE with some existing position location schemes and demonstrate that it also serves as a good performance benchmark for collaborative position location.**

***Index Terms*—Collaborative position location, maximum likelihood estimation (MLE), branch-and-bound (BB), reformulation linearization technique (RLT), Cramer-Rao lower bound (CRLB), linear least-squares (LLS), mean square error (MSE).**

I. Introduction

ACCURATE location information is of great importance in future commercial and public safety wireless systems. In conventional infrastructure-based localization systems, wireless devices, such as GPS receivers or cell phones, derive their locations by making measurements to GPS satellites or cellular base stations as in E-911. On the other hand, for the large-scale deployment of wireless sensor networks (WSN), sensor self-localization, with the help of a few sensors at known locations, is a pre-requisite for the subsequent sensing task. In such scenarios, it is most likely that some, if not all, wireless devices or sensors (referred to as unlocalized nodes or simply, nodes), do not have enough connections to reference locations, namely anchors, to determine their locations. Hence, to estimate their locations, unlocalized nodes have to utilize not only the available measurements to anchors, but also the measurements among themselves. This concept has been commonly referred to as *collaborative* or cooperative position location [1], [2], as opposed to *non-collaborative* position location where an unlocalized node estimates its location by only using range or angle estimates with respect to anchors.

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 [3] 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 [2]. For instance, the authors in [3] formulated the (centralized) maximum likelihood estimate for distance-based localization, where inter-node distance can be derived from either time-of-arrival (TOA) or received signal strength (RSS) measurements. In [4]-[7], several other centralized approaches have been developed for collaborative position location. On the other hand, distributed algorithms, e.g., [8]-[10], have been proposed to address the scalability and computational complexity issues associated with centralized approaches. Some other algorithmic developments can be found in [11]-[16] and a more detailed survey can be found in [2]. Note that, using angle-of-arrival (AOA) as either a substitute [17] or supplemental measurement [18], and filtering techniques based on time-varying measurements [19] 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 [3], 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 [6] and [7] 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 [11] presented the MLE for collaborative localization using RSS measurements.

Manuscript received June 9, 2009; revised September 18, 2009 and November 1, 2009; accepted November 2, 2009. The associate editor coordinating the review of this paper and approving it for publication was D. Tarchi.

The authors are with the Mobile and Portable Radio Research Group (MPRG), Wireless@Virginia Tech, The Bradley Department of Electrical and Computer Engineering, Blacksburg, VA, 24061 (e-mail: taojia@vt.edu).

Digital Object Identifier 10.1109/TWC.2010.01.090869

However, the MLE is only evaluated via a grid-based solution search strategy, and thus is not guaranteed to be optimal. On the other hand, metaheuristic algorithms, (e.g., simulated annealing), are proposed to address the solution optimality issue in [13]-[14]. Nevertheless, it is known that these methods cannot guarantee that the solution is in fact optimal.

In this paper, 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 [29], to numerically solve the MLE associated with collaborative position location, as given in [3]. 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 unbiased estimator (despite its widespread use for this problem) sometimes is not a meaningful performance benchmark (this was also observed in [23], [24]). 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.

The rest of this paper is organized as follows. In Section II, we briefly describe the network as well as measurement models and the MLE problem formulation. In Section III, we elaborate on how the BB/RLT is applied to our problem and the mathematical development of the necessary solution procedure. We then present numerical validation as well as simulation results in Section IV. Concluding remarks are given in Section V.

II. Problem Formulation

We consider a two-dimensional (2D) square network with an area size of $(\mathcal{L}\times\mathcal{L})\,\mathrm{m}^2$, consisting of m anchors at known locations and n unlocalized nodes whose locations are to be estimated. The true locations of the unlocalized nodes are denoted by $\Theta = [\theta_1^T, \theta_2^T, \ldots, \theta_n^T]$ and the known locations of the anchors are denoted by $\mathbf{A} = [\theta_{n+1}^T, \theta_{n+2}^T, \ldots, \theta_{n+m}^T]$, where $(\cdot)^T$ is the matrix transpose operation and $\theta_i = [x_i\ y_i]^T$ is the 2D coordinate of the ith node, for $i = 1, 2, \ldots, n+m$. 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 are neighbors to each other, and thus can communicate and obtain a (noisy) range estimate r_{ij} of their true distance d_{ij}. In practice, this can be achieved by measuring RSS, TOA or the round-trip signal propagation time between the two nodes. Further, we assume range estimates are symmetric, i.e., $r_{ij} = r_{ji},\ \forall i, j$. The task of position location is then to obtain an estimate of Θ, given the set of inter-node range estimates and the set of anchor locations $\mathbf{A}$.

We consider TOA-based range or distance estimates corrupted by independent zero-mean Gaussian random noise. The range estimate between the ith and jth nodes can be modeled as

$$r_{ij} = d_{ij} + n_{ij}, \tag{1}$$

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, 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 a future work. Based on this model, the MLE of collaborative position location can be formulated as

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

where $\mathcal{H}(i) = \{q \,|\, q = 1, 2, \ldots, n+m,\ d_{iq} \le R\}$ 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[1]. 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 (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 (2). Finally, as can be easily seen, the MLE problem given in (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 (2), several alternatives have been developed. For example, in the case of non-collaborative position location (where (2) can be broken into n separate minimization problems), the authors in [20] proposed the linear least-squares (LLS) location estimator by linearizing the range estimates prior to location estimation. For collaborative position location, the authors in [6] resorted to a semi-definite programming (SDP) relaxation to the MLE problem in (2)

[1]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 [3], while we always include σ_{ij}^2 into $f(\Theta)$. Note that this does not alter the optimal solution procedure.

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 (2). To facilitate the comparison between the MLE and other existing methods, we define the following metrics. First, we term the unitless[2] objective value of the MLE, i.e., $f(\Theta)$, as the *residual* of Θ. Second, 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

$$\Omega \triangleq \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left[(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2\right]}, \tag{3}$$

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.,

$$\overline{\Omega} \triangleq \sqrt{\frac{1}{n}\sum_{i=1}^{n}\mathbf{E}\left\{(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2\right\}}, \tag{4}$$

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

III. 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 (2) and the necessary algorithm components. Finally, we discuss its computational complexity and propose a few convergence speed-up techniques.

A. BB/RLT Algorithm

The BB/RLT algorithm is outlined in [29] and was applied to finding the capacity of a multiuser MIMO system with interference in [28]. In general, for a nonlinear and nonconvex programming problem as in (2), gradient-search-based methods 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 [29]. 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. 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 LB $> (1-\epsilon)$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 [29]. The generic BB/RLT algorithm for a minimization problem is outlined in Fig. 1. In the following, we describe how we construct the LP relaxation and apply the BB/RLT solution procedure to the MLE problem.

[2]Note that the objective function is unitless since the square error is normalized by the noise variance.

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.

Fig. 1. Generic branch-and-bound/reformulation linearization technique (BB/RLT) solution procedure.

B. Reformulation and Linearization of the MLE

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

$$Z_{ij} = \sqrt{W_{ij}} \tag{5}$$

$$W_{ij} = (x_i - x_j)^2 + (y_i - y_j)^2, \tag{6}$$

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

$$\ell(Z,W) = \sum_{i=1}^{n}\sum_{\substack{j=i+1 \\ j\in\mathcal{H}(i)}}^{n+m}\frac{1}{\sigma_{ij}^2}\left(r_{ij}^2 - 2\,r_{ij}Z_{ij} + W_{ij}\right). \tag{7}$$

where $Z = \{Z_{ij}\}$ and $W = \{W_{ij}\}$ are the sets of existing Z_{ij} and W_{ij}, respectively.

Second, we need to linearize the constraint given by the square root function in (5). 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

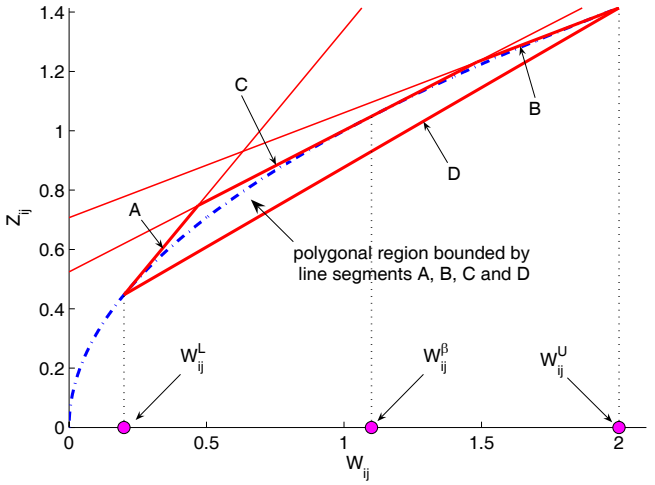

Fig. 2. Polygonal bounding region of $Z_{ij} = \sqrt{W_{ij}}$ for $W_{ij}^L = 0.2$ and $W_{ij}^U = 2$. The four tangential lines define a polyhedral bounding region for the square root function between W_{ij}^L and W_{ij}^U.

denoted by W_{ij}^β, i.e.,

$$W_{ij}^\beta = \frac{1}{2}(W_{ij}^L + W_{ij}^U), \tag{8}$$

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,

$$-W_{ij} + 2\sqrt{W_{ij}^L}\, Z_{ij} \leq W_{ij}^L, \tag{9}$$

$$-W_{ij} + 2\sqrt{W_{ij}^U}\, Z_{ij} \leq W_{ij}^U, \tag{10}$$

$$-W_{ij} + 2\sqrt{W_{ij}^\beta}\, Z_{ij} \leq W_{ij}^\beta, \tag{11}$$

$$W_{ij} - \left(\sqrt{W_{ij}^L} + \sqrt{W_{ij}^U}\right) Z_{ij} \leq -\sqrt{W_{ij}^L W_{ij}^U}. \tag{12}$$

An example of such a polygonal bounding region, for $W_{ij}^L = 0.2$ and $W_{ij}^U = 2$, is illustrated in Fig. 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 tangential line is properly defined as in (11). 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 [28]. 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 (6), which is related to the estimated distance between the ith and jth nodes. For that, we define

$$U_i = x_i^2, \tag{13}$$

$$V_i = y_i^2, \tag{14}$$

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

$$S_{ij} = x_i\, x_j, \tag{15}$$

$$T_{ij} = y_i\, y_j, \tag{16}$$

Substituting (13)-(16) into (6), 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:

$$W_{ij} - U_i + 2\, S_{ij} - U_j - V_i + 2\, T_{ij} - V_j = 0\,; \tag{17}$$

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

$$W_{ij} - U_i + 2x_i x_j - x_j^2 - V_i + 2y_i y_j - y_j^2 = 0. \tag{18}$$

With (17) and (18), we now linearize the quadratic constraints defined by (13)-(16). 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,

$$x_i - x_i^L \geq 0;\ x_i^U - x_i \geq 0; \tag{19}$$

Adopting the RLT [29], we can derive the following so-called bounding-factor constraints,

$$x_i^2 - 2x_i^L x_i \geq -(x_i^L)^2 \tag{20}$$

$$x_i^2 - 2x_i^U x_i \geq -(x_i^U)^2 \tag{21}$$

$$x_i^2 - (x_i^L + x_i^U)x_i \leq -x_i^L x_i^U\,. \tag{22}$$

Now, substituting (13) into (20)-(22), we obtain three linear constraints involving U_i and x_i, which essentially define a triangular bounding region for (13) 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,

$$x_i - x_i^L \geq 0;\ x_i^U - x_i \geq 0; \tag{23}$$

$$x_j - x_j^L \geq 0;\ x_j^U - x_j \geq 0. \tag{24}$$

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

$$x_i x_j - x_i^L x_j - x_j^L x_i \geq -x_i^L x_j^L \tag{25}$$

$$-x_i x_j + x_i^L x_j + x_j^U x_i \geq x_i^L x_j^U \tag{26}$$

$$-x_i x_j + x_i^U x_j + x_j^L x_i \geq x_i^U x_j^L \tag{27}$$

$$x_i x_j - x_i^U x_j - x_j^U x_i \geq -x_i^U x_j^U\,. \tag{28}$$

Substituting (15) into (25)-(28), we obtain four linear constraints involving S_{ij}, x_i and x_j, which essentially define a polyhedral bounding region for (15) when $x_i^L \leq x_i \leq x_i^U$ and $x_j^L \leq x_j \leq x_j^U$. Similar bounding-factor constraints can be derived for T_{ij} via y_i and y_j.

By combining all the aforementioned linear constraints with the linearized objective function $\ell(Z, W)$, the LP relaxation to the original MLE problem in (2) is given by

$$\underline{\phi}^* = \arg\min_{\phi}\, \ell(Z, W) \tag{29}$$

$$= \arg\min_{\phi} \sum_{i=1}^{n} \sum_{\substack{j=i+1 \\ j\in\mathcal{H}(i)}}^{n+m} \frac{1}{\sigma_{ij}^2}(r_{ij} - 2\, r_{ij} Z_{ij} + W_{ij})$$

s.t.: polygonal bounding constraints on (Z_{ij}, W_{ij}) (9)-(12),
equality constraints in (17) and (18),
linear constraints on U_i derived from (20)-(22),
linear constraints on V_i, similar to U_i,
linear constraints on S_{ij} derived from (25)-(28),
linear constraints on T_{ij}, similar to S_{ij}.

In (29), 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.

C. 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. 1, once the initial bounding interval for Θ is given, we can easily evaluate the lower and upper bounds for W_{ij} according to (6) 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 (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 (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 [29]. In addition, as mentioned in [29], 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 the 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.

D. 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 (29) are calculated as

$$N_{\text{var}} = 4n + 2\sum_{i=1}^{n} N^{(i)} + 2\sum_{i=1}^{n} N_u^{(i)} \quad (30)$$

$$N_{\text{con}} = 10n + 5\sum_{i=1}^{n} N^{(i)} + 8\sum_{i=1}^{n} N_u^{(i)} \quad (31)$$

where n is the total number of unlocalized nodes and

$$N^{(i)} = \text{card}\{q \,|\, q \in \mathcal{H}(i), i+1 \le q \le n+m\} \quad (32)$$

$$N_u^{(i)} = \text{card}\{q \,|\, q \in \mathcal{H}(i), i+1 \le q \le n\} \quad (33)$$

denote the total numbers 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 $\text{card}\{\cdot\}$ 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 we will demonstrate in Section IV, 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 (29). 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 (5). 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

$$W_{ij}^{\beta_p} = (W_{ij}^{\beta_{p-1}} + W_{ij}^L)/2, \quad p = 1, 2, \ldots, k_1. \quad (34)$$

where $W_{ij}^{\beta_0} \stackrel{\triangle}{=} W_{ij}^U$. Then we define the corresponding k_1 tangential lines at these k_1 points. This will form a refined polygonal bounding region for (5), and thus reduce the search space for Z and W. An example with $k_1 = 3$, $W_{ij}^L = 0.1$ and $W_{ij}^U = 200$ is shown in Fig. 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

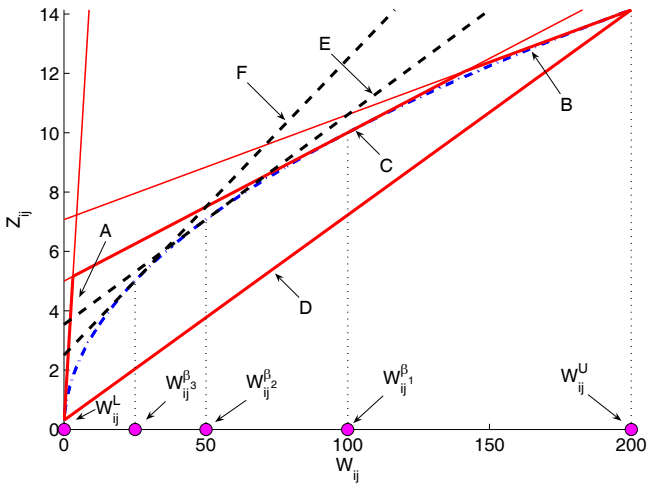

Fig. 3. Improved polygonal bounding region of $Z_{ij} = \sqrt{W_{ij}}$ for $W_{ij}^L = 0.1$ and $W_{ij}^U = 200$. The two additional tangential lines (dash) E and F refine the bounding region and reduce the search space around the small value region of W_{ij}.

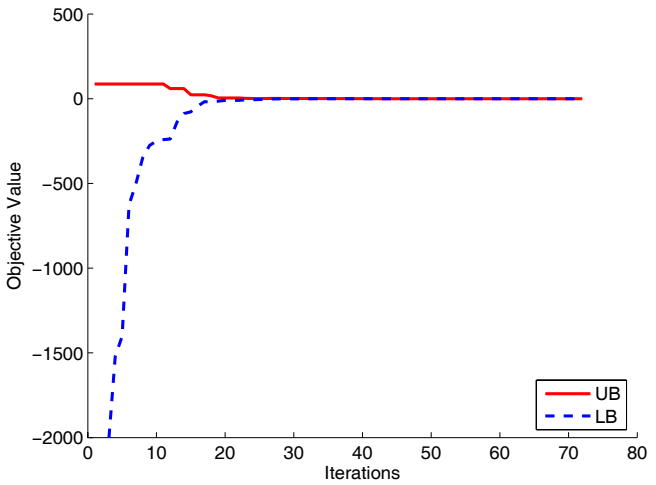

Fig. 4. The convergence of the UB and LB on the MLE objective value for noise-free case, with 3 anchors at (5, 5), (10, 15) and (15, 5), and unlocalized node at (3, 5). The final objective value is found to be 1.17×10^{-6} and the solution is (2.999, 4.999).

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 in order for the algorithm to converge. As we observe 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.

IV. Numerical Results

In this section, we give simulation results using the BB/RLT algorithm to solve (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 variance of 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 [20] as well as the Cramer-Rao lower bound (CRLB) on localization error [3]. For collaborative localization, we compare our method with the semi-definite programming (SDP) approach [25] and the gradient-search-based solution to the MLE.

First, we present the numerical validation of our BB/RLT formulation to the MLE problem. In Fig. 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, 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 found to be below -2500, the LB quickly converges to reasonable values near zero after about 15 iterations. In fact, the power of the BB/RLT algorithm is even better 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). In the absence of noise, we observed in our simulations that gradient-search-based methods sometimes return the wrong solution of (5, 3) depending on the initialization, due to the so-called flip ambiguity [22], which is indeed a local optimum with an objective value very close to that of the true global optimum. On the other hand, the BB/RLT is numerically guaranteed to find the optimal solution around (3, 5).

In Figs. 5 and 6, 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. For the LLS estimator, we always use the first anchor as the reference to linearize the range estimates [20]. In Fig. 5, we plot the mean localization errors of the MLE and the

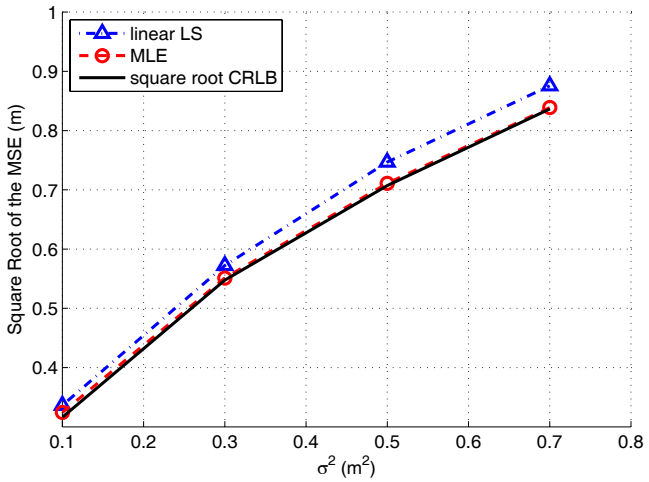

Fig. 5. Mean localization errors of the LLS and the MLE versus σ^2 with good geometry, where the CRLB serves as a good benchmark. Four anchors are placed at four corners (0 0), (0 40), (40 0), (40 40) and the unlocalized node is at (20 20).

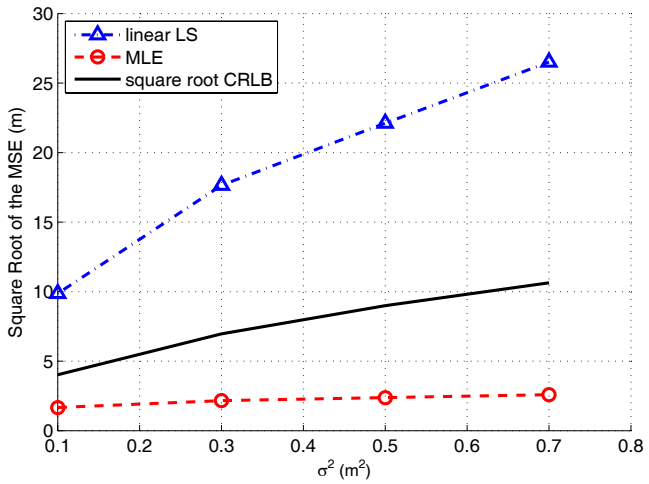

Fig. 6. Mean localization errors of the LLS and the MLE versus σ^2 with bad geometry, where the CRLB is no longer a meaningful benchmark. Four almost-collinear anchors are placed at (0 0), (10 11), (28 29), (40 40) and the unlocalized node is at (20 20).

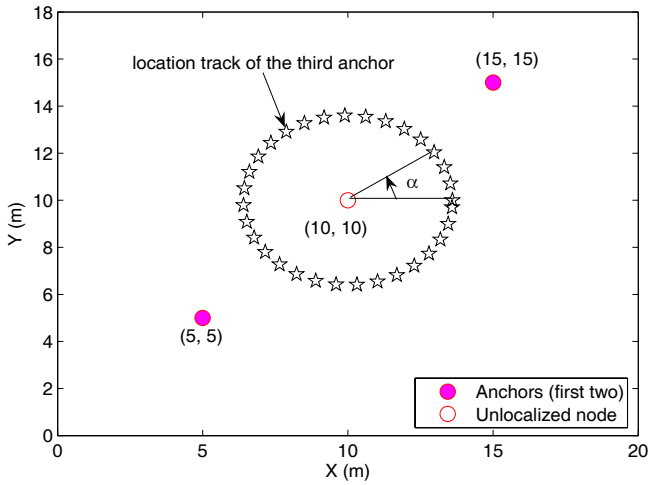

Fig. 7. Node geometry to demonstrate the bias, MSE, estimator variance of the MLE. The pentagons represent the testing locations of the third anchor, starting from $\alpha = 0^\circ$ to 360°.

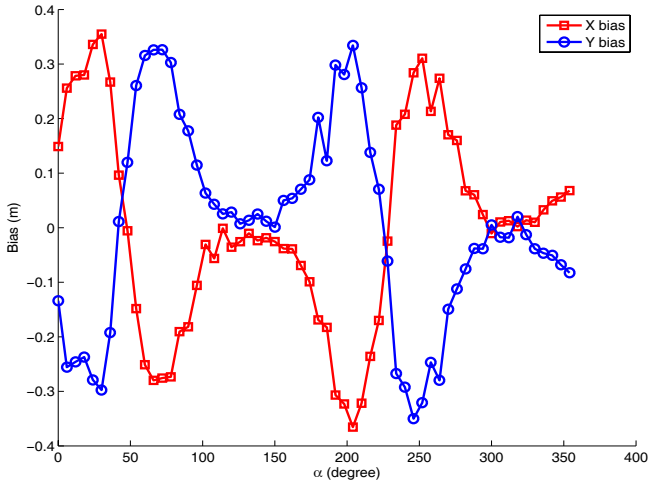

Fig. 8. Biases on x and y of the MLE with respect to α, for the node setup in Fig. 7.

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. 6, 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 suprisingly, however, the MLE performs even better than the CRLB. Despite being counter-intuitive, this has partly been observed in [23], [24]. 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 [23] 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 (2) is in general a *biased* estimator. It is worth mentioning that the authors in [3] 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 [26], [27], this observation is easily justified.

In Figs. 7-9, 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

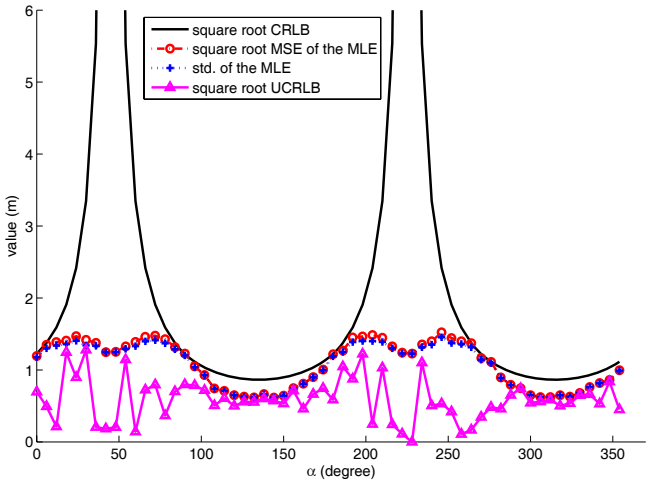

Fig. 9. The comparison between the CRLB and the MSE of the MLE, as well as the comparison between the uniform CRLB (UCRLB) and the standard deviation of the MLE, with respect to α, for the node setup in Fig. 7.

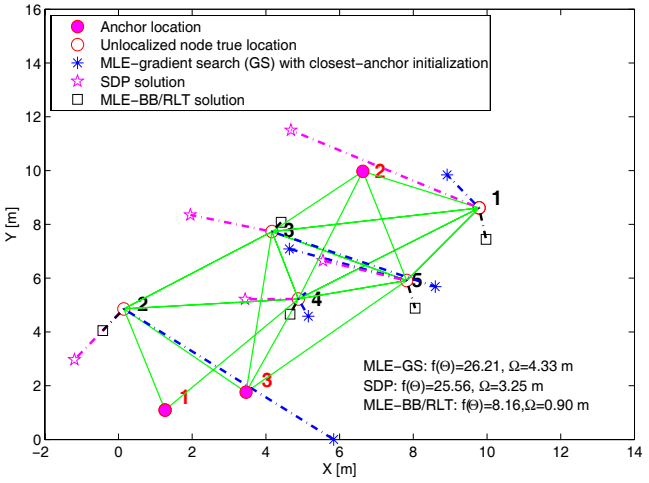

Fig. 10. The MLE solution computed by the BB/RLT and the gradient-search-based method, as well as SDP solution, for a particular network and noise realization. The noise variance is $\sigma^2 = 1\,\mathrm{m}^2$

addition, we apply the theoretical framework for the uniform CRLB (UCRLB) proposed in [27] 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 [27] as the measure of the bias matrix norm to calculate the UCRLB. In Fig. 7, 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 pentagons. This allows us to sweep through good and bad geometry conditions to see the effect that geometry plays. In Fig. 8, we present the estimation biases of x and y with respect to α, obtained from 1000 noise realizations. We can see that the TOA-based MLE in this case is clearly biased. In Fig. 9, we present the square root of the CRLB and the mean localization error of the MLE, which is essentially 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., $\alpha = 45^\circ$ and 225°. Also included in Fig. 9 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

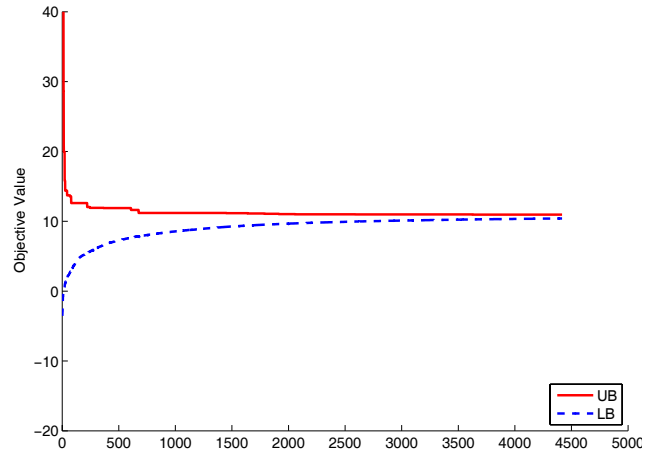

Fig. 11. The convergence of the UB and LB on the MLE objective value for one noise realization for the network in Fig. 10.

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 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. 5 to Fig. 9, 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, which is robust to both range estimation noise and more importantly, node geometry.

In Fig. 10, we present the result of 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.,

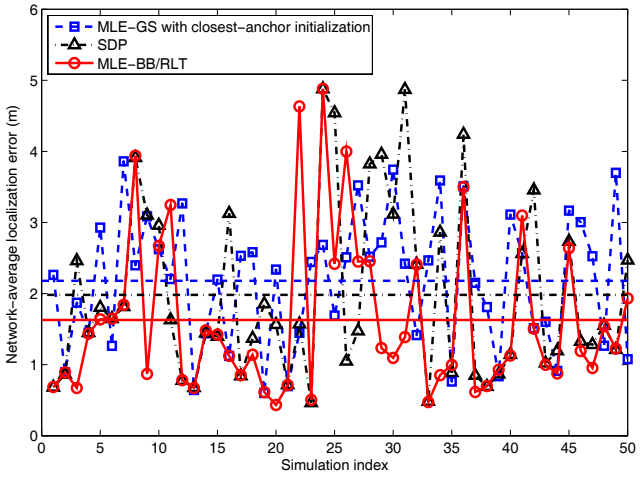

Fig. 12. The network-average localization error with respect to 50 different noise realizations for the network in Fig. 10. The three additional horizontal lines represent the mean network-average localization error for the three methods, respectively.

$\sigma_{ij}^2 = \sigma^2 = 1\,\mathrm{m}^2$. The solid (green) lines represent the connectivity and the dash-dot lines quantize the amount of the localization error. For comparison purposes, we also presented the solution of the SDP approach [25] as well as the MLE solution computed using the classical gradient-search-based method. For the gradient-search-based method, we used the standard *lsqnonlin* function in Matlab to solve (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 [25]. 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 gradient-search-based solution. In fact, the network-average localization error of the BB/RLT solution is $0.9\,\mathrm{m}$ (averaged over 5 nodes), which is smaller than the $3.25\,\mathrm{m}$ of the solution using the SDP approach and $4.33\,\mathrm{m}$ of the solution obtained using the gradient-search-based method. It is observed for this particular network geometry and range estimation noise realization, using the gradient-search-based method to solve the MLE leads to the flip ambiguities which are essentially local instead of the global optima, while the BB/RLT algorithm is guaranteed to find the global optimum to the MLE, leading to a much smaller localization error. In Fig. 11, 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, due to a larger search space.

In Fig. 12, 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. 10. The three additional horizontal lines represent the mean network-average localization errors, which are $2.17\,\mathrm{m}$, $1.98\,\mathrm{m}$ and $1.63\,\mathrm{m}$ for the MLE using gradient search, 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 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. 5 and 6, 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. 10, 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 connecity, 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.

V. Conclusion and Future Work

In this paper, 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 more efficient suboptimal and ideally distributed position location algorithms, which can approach the MLE performance.

Acknowledgements

This research was supported by the National Science Foundation under Grants 0515019 and 0802112. The authors would like to thank Jia (Kevin) Liu, Dr. Thomas Hou and Dr. Hanif D. Sherali for their help and valuable discussions.

References

[1] 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 Process. Mag.*, vol. 22, no. 4, pp. 54-69, 2005.

[2] G. Mao, B. Fidan, and B. D. O. Anderson, "Wireless sensor network localization techniques," *International J. Computer Telecommun. Netw.*, vol. 51, no. 10, pp. 2529-2553, 2007.

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

[4] L. Doherty, K. Pister, and L. E. Ghaoui, "Convex position estimation in wireless sensor networks," in *Proc. IEEE Infocom*, 2001.

[5] Y. Shang, W. Ruml, Y. Zhang, and M. P. J. Fromherz, "Localization from mere connectivity," in *Proc. MobiHoc*, pp. 201-212, 2003.

[6] P. Biswas and Y. Ye, "Semidefinite programming for ad hoc wireless sensor network localization," in *International Symp. Inf. Process. Sensor Netw.*, pp. 46-54, 2004.

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

[8] D. Niculescu and B. Nath, "Ad hoc positioning system (APS)," in *Proc. IEEE GLOBECOM*, pp. 2926-2931, 2001.

[9] C. Savarese and J. Rabaey, "Robust positioning algorithms for distributed ad-hoc wireless sensor networks," in *Proc. USENIX Annual Technical Conf.*, pp. 317-327, 2002.

[10] A. Savvides, H. Park, and M. B. Srivastava, "The bits and flops of the n-hop multilateration primitive for node localization problems," in *Proc. Int. Workshop Sensor Netw. Application*, pp. 112-121, 2002.

[11] D. Dardari and A. Conti, "A sub-optimal hierarchical maximum likelihood algorithm for collaborative localization in ad-hoc network," in *Proc. IEEE SECON 2004*, pp. 425-429, Oct. 2004.

[12] J. Liu, Y. Zhang, and F. Zhao, "Robust distributed node localization with error management," in *MobiHoc*, pp. 250-261, 2006.

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

[14] P.-J. Chuang and C.-P. Wu, "An effective PSO-based node localization scheme for wireless sensor networks," in *Proc. Int. Conf. Parallel Distributed Computing, Appl. Tech.*, pp. 187-194, Dec. 2008.

[15] A. J. Weiss and J. S. Picard, "Network localization with biased range measurements," *IEEE Trans. Wireless Commun.*, vol. 7, pp. 298-304, Jan. 2008.

[16] F. Chan, H. C. So, and W.-K. Ma, "A novel subspace approach for cooperative localization in wireless sensor networks using range measurements," *IEEE Trans. Signal Process.*, vol. 57, pp. 260-269, Jan. 2009.

[17] B. Ananthasubramaniam and U. Madhow, "Cooperative localization using angle of arrival measurements in non-line-of-sight environments," in *Proc. ACM Workshop Mobile Entity Localization Tracking GPS-less Environments*, pp. 117-122, 2008.

[18] R. Huang and G. V. Záruba, "Incorporating data from multiple sensors for localizing nodes in mobile ad hoc networks," *IEEE Trans. Mobile Computing*, vol. 6, pp. 1090-1104, Sept. 2007.

[19] R. Parker and S. Valaee, "Cooperative vehicle position estimation," in *Proc. IEEE International Conf. Commun.*, pp. 5837-5842, 2007.

[20] J. J. Caffery, "A new approach to the geometry of TOA location," in *Proc. IEEE VTC*, vol. 4, pp. 1943-1949, 2000.

[21] I. Guvenc, C.-C. Chong, and F. Watanabe, "Analysis of a linear least-squares localization technique in LOS and NLOS environments," in *Proc. IEEE VTC, Spring*, pp. 1886-1890, Apr. 2007.

[22] 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 *Proc. INFOCOM 2004.*, vol. 4, pp. 2673-2684, Mar. 2004.

[23] Y. Qi, T. Asai, H. Yoshino, and N. Nakajima, "On geolocation in ill-conditioned BS-MS layouts," in *Proc. IEEE International Conf. Acoustics, Speech, Signal Processing*, vol. 5, pp. 697-700, Mar. 2005.

[24] S. O. Dulman, A. Baggio, P. J. M. Havinga, and K. G. Langendoen, "A geometrical perspective on localization," in *Proc. ACM Workshop Mobile Entity Localization Tracking GPS-less Environments*, pp. 85-90, 2008.

[25] Z. Wang, S. Zheng, S. Boyd, and Y. Ye, "Further relaxations of the SDP approach to sensor network localization," technical report, Stanford University, May 2007. [Online]. Available: http://www.stanford.edu/ yyye/relaxationsdp9.pdf.

[26] A. O. Hero, J. A. Fessler, and M. Usman, "Exploring estimator bias-variance tradeoffs using the uniform CR bound," *IEEE Trans. Signal Process.*, vol. 44, no. 8, pp. 2026-2041, 1996.

[27] Y. C. Eldar, "Minimum variance in biased estimation: bounds and asymptotically optimal estimators," *IEEE Trans. Signal Process.*, vol. 52, no. 7, pp. 1915-1930, 2004.

[28] J. Liu, Y. Hou, Y. Shi, H. Sherali, and S. Kompella, "On the capacity of multiuser MIMO networks with interference," *IEEE Trans. Wireless Commun.*, vol. 7, pp. 488-494, Feb. 2008.

[29] H. D. Sherali and W. P. Adams, *A Reformulation-Linearization-Technique for Solving Discrete and Continuous Nonconvex Problems*. Boston, MA: Kluwer Academic Publishing, 1999.

Tao Jia received the B.Eng. degree in Electrical Engineering from University of Science and Technology of China (USTC), Hefei, in 2003, and the M.A.Sc. degree in Electrical Engineering from Simon Fraser University, Burnaby, Canada, in 2006.

He is now with the Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University (Virginia Tech), Blacksburg, VA, where he is working for a Ph.D. degree in the Mobile and Portable Radio Research Group (MPRG). His research interests include algorithm development for position location in wireless sensor networks, physical layer design and performance analysis for UWB communication systems, resource allocation for multi-user multi-antenna systems.

Dr. R. Michael Buehrer joined Virginia Tech from Bell Labs as an Assistant Professor with the Bradley Department of Electrical Engineering in 2001. He is currently an Associate Professor and is member of *Wireless @ Virginia Tech*, a comprehensive research group focusing on wireless communications. His current research interests include dynamic spectrum sharing, cognitive radio, communication theory, Multiple Input Multiple Output (MIMO) communications, intelligent antenna techniques, position location networks, Ultra Wideband, spread spectrum, interference avoidance, and propagation modeling. His work has been funded by the National Science Foundation, the Defense Advanced Research Projects Agency, Office of Naval Research, and several industrial sponsors.

Dr. Buehrer has co-authored over 30 journal and 75 conference papers and holds 11 patents in the area of wireless communications. He is currently a Senior Member of IEEE, and an Associate Editor for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and IEEE TRANSACTIONS ON COMMUNICATIONS. He was formerly an associate editor for IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGIES, IEEE TRANSACTIONS ON SIGNAL PROCESSING, and IEEE TRANSACTIONS ON EDUCATION. In 2003 he was named Outstanding New Assistant Professor by the Virginia Tech College of Engineering.