Solving the Node Localization Problem in WSNs by a Two-objective Evolutionary Algorithm and Local Descent

Massimo Vecchio, Roberto López–Valcarce
Departamento de Teoría de la Señal y las Comunicaciones,
University of Vigo,
C/ Maxwell s/n, 36310 Vigo - SPAIN
Email: {massimo,valcarce}@gts.uvigo.es

Francesco Marcelloni
Dipartimento di Ingegneria dell'Informazione,
University of Pisa,
Largo Lucio Lazzarino 1, 56122 Pisa - ITALY
Email: f.marcelloni@iet.unipi.it

Abstract—Given a small percentage of nodes whose actual positions are known, the problem of estimating the locations of the remaining nodes of a wireless sensor network has attracted a large interest in the last years. The localization task is based on the noisy estimates of the distances between pairs of nodes in range of each other. The problem is particularly hard when the network connectivity is not sufficiently high, the most attractive case in real applications. In this paper, we propose to solve the localization problem by using a two–objective evolutionary algorithm which takes concurrently into account during the evolutionary process both the localization accuracy and certain topological constraints induced by the network connectivity. The solutions generated by the evolutionary algorithm are therefore refined by a gradient-based technique which further reduces the localization error. The proposed approach is tested with different network configurations and sensor setups, and compared in terms of normalized localization error with a state–of–the–art approach based on a regularized semi–definite programming technique. The results show that, in all the experiments, our approach achieves considerable accuracies, thus manifesting its effectiveness and stability, and outperforms the compared approach.

Keywords-Stochastic Optimization; Multi-objective Evolutionary Algorithms; Range Measurements;

I. INTRODUCTION

A Wireless Sensor Network (WSN) may consist of hundreds or even thousands of low–cost nodes communicating among themselves for applications like environment monitoring, precision agriculture, vehicle tracking, logistics, etc. In such applications knowledge about the location of sensor nodes may play a key role [1], [2]. Although in principle the use of a Global Positioning System (GPS) could enable such "*location awareness*", this solution is not always viable in practice. The first reason is merely economic, as the cost of GPS receivers is not negligible. The second reason is related to the power consumption of a standard GPS receiver, which is generally not affordable by battery–powered nodes. Last

but not least, a technology–related reason arises in indoor and underground WSN deployments: in these situations, in fact, communication with satellites may be compromised.

These limitations have motivated alternative approaches to the problem. Among these, *fine–grained* localization techniques arise as a flexible option. In these schemes, only a few nodes of the network (termed *anchor nodes*) are endowed with their exact positions through GPS or manual placement, while all nodes are able to estimate their distances to nearby nodes by using some measurement technique. These distance–related techniques include Received Signal Strength (RSS) measurements, Time of Arrival (ToA), Time Difference of Arrival (TDoA), etc. (for a review of these techniques the reader car refer to [1], [2]). Thus, assuming that the coordinates of anchor nodes are known, and exploiting pairwise distance measurements among the nodes, the fine–grained localization problem is to determine the positions of all non–anchor nodes. This task has proved to be rather difficult, due to the following reasons: *i)* determining the locations of the nodes from a set of pairwise distance estimates is a nonconvex optimization problem; *ii)* the measurements available to nodes are invariably corrupted by noise; and finally, *iii)* even if the distance estimates were perfectly accurate, sufficient conditions for the solution to be unique are not easily identified [3]. We will briefly discuss these issues in the following.

Assuming a statistical characterization of measurement noise, which will usually depend on the kind of measurement technique [1], Maximum Likelihood (ML) estimation is the natural approach to the localization problem. However, as previously mentioned, the ML formulation results in a multivariable nonconvex optimization problem. Three different approaches to this task can be found in the literature, namely multidimensional scaling, convex relaxation and stochastic optimization. The first approach (MDS) [4] is a *connectivity– based* technique that, in addition to distance measurements, exploits knowledge about the topology of the network; this information imposes additional constraints on the problem, since nodes within communication range of each other cannot be arbitrarily far apart. Although appealing, MDS

is not able to converge to sufficiently accurate solutions. The second approach relaxes the original nonconvex ML formulation in order to obtain a Semi–Definite Programming (SDP) or a Second–Order Cone Programming (SOCP) problems. Global solutions to these relaxed, convex problems can be then obtained with moderate computational effort [5] [6] and constitute approximate solutions to the original nonconvex problem. Since [6] has shown that the solutions obtained by SOCP relaxation are less accurate than those obtained by SDP relaxation, in this paper we will compare the performances of the proposed algorithm only against SDP. The third approach attempts to avoid local maxima of the likelihood function by resorting to global optimization methods, such as *e.g.* simulated annealing [7]. In [8] we have already shown that the proposed evolutionary method is able to outperform a state–of–the–art localization counterpart based on simulated annealing, thus we will not consider comparisons with techniques belonging to the third approach.

In the following we will describe the most accurate convex relaxation approach, which is the regularized version of the original FSDP proposed in [5]. Indeed, after having observed that FSDP could still incur significant estimation errors, the same authors proposed a regularized version (referred to as FSDPr) and a gradient–descent refinement technique in [3] and [9], respectively, in order to improve its performance. FSDPr adds a regularization term to the objective function in order to reduce the tendency of FSDP solutions to have points crowded together. This tendency occurs when projecting the high–rank SDP solution back onto the two–dimensional plane in the last step. Thus, the regularization term suggested in [3] penalizes small node separations. The main issue in FSDPr is the choice of the regularization value: if it is too low, then the effects of regularization are negligible; on the other hand, if this value is too high, then the regularization term will overwhelm the error term, making the SDP either unfeasible or yielding a solution whose points are too far apart. Although there is no way to obtain a good regularization value *a priori*, in [3] the authors proposed a heuristic method for its computation based on the solution provided by FSDP without regularization. Finally, the goal of gradient–based refinements is to improve the final estimation given by a localization algorithm. Since gradient–based methods generally do not deliver a global optimal solution when the problem is not convex, this technique can be applied as a fine–tuning phase once an approximation of the global solution has been found [3], [9]. Thus, the technique can be applied to any localization method.

In this paper we propose to tackle the localization problem by using a Multi–Objective Evolutionary Algorithm (MOEA). In particular, we adopt two objectives: the first objective, denoted *CF*, is given by the original nonconvex cost, that is, the sum of squared differences between the estimated and the corresponding measured inter–node distances. The second objective, denoted *CV*, exploits the connectivity–based *a priori* information about the network, and is especially useful in order to alleviate *localizability* issues.

The proposed approach is tested with a variety of network topologies, percentages of anchor nodes, and connectivity ranges, and compared in terms of normalized localization error with FSDPr, without and with a gradient–based refinement stage. We show that the proposed evolutionary method produces more accurate estimates of the node locations than the FSDPr. The improvement is more significant for network topologies with lower connectivity, for which the localization problem becomes more difficult. Thus, the proposed approach constitutes a good candidate for WSN applications which do not demand highly–scalable or real–time solutions but do require high localization accuracies.

The paper is organized as follows. In Section II, we present the problem formulation. Section III introduces our multi–objective evolutionary approach to the problem. The experimental results of our performance analysis are presented in Section IV. Finally, in Section V we draw some conclusions.

II. PROBLEM FORMULATION

In this section we first introduce the system model, the objective functions used in the evolutionary algorithm and the performance metric adopted in the comparison. Then we discuss certain geometrical constraints which can be defined on each non–anchor node, exploiting the *a priori* information (*i.e.* connectivity) about the network topology.

A. System model

We consider a WSN with n nodes deployed in $\mathcal{T} = [0,1] \times [0,1] \subset \mathbb{R}^2$. Among these, nodes 1 through m, with $m < n$, are anchor nodes whose coordinates $\mathbf{p}_i = (x_i, y_i) \in \mathcal{T}$, $i = 1,\ldots, m$, are known. We assume that if two sensor nodes, say i and j, are within the communication range of each other, then their inter–node distance d_{ij} can be estimated by using some measurement technique (see Section I). Distance measurements d_{ij} are modeled as

$$d_{ij} = r_{ij} + e_{ij} \tag{1}$$

where $r_{ij} = \|\mathbf{p}_i - \mathbf{p}_j\|$ is the actual distance between nodes i and j ($\|\cdot\|$ denotes the Euclidean norm). Similar to [3], [5], [7], [9], we assume that measurement errors e_{ij} follow a zero–mean Gaussian distribution with variance σ^2. It is also assumed that the random variables e_{ij} and e_{kl} are statistically independent for $(i, j) \neq (k, l)$.

A simple disk model is adopted for network connectivity: nodes i and j can communicate with each other if and only if $r_{ij} \leq R$, where R is the connectivity radius. This model is commonly used in the literature, although empirical measurements on real WSNs have shown that it is only an approximation in practice. On the other hand, different

connectivity models could be adopted by modifying the geometrical analysis in Section II-C. We refer to nodes j such that $r_{ij} \leq R$ as *first–level neighbors* of node i. Further, we refer to all nodes j which are not first–level neighbors of node i, but which share at least a first–level neighbor with node i, as *second–level neighbors* of node i. Let

$$
\begin{aligned}
N_i &= \{j \in 1 \ldots n, j \neq i : r_{ij} \leq R\} \quad (2) \\
\overline{N}_i &= \{j \in 1 \ldots n, j \neq i : r_{ij} > R\} \quad (3)
\end{aligned}
$$

be the set of the first–level neighbors of node i and its complement, respectively. We assume that sets N_i and $\overline{N}_i$ are known for all $i = 1, \ldots, n$. This is a reasonable assumption, since each node can easily determine which other nodes it can communicate with.

B. Objective functions and performance metric

Pursuing the goal of estimating the positions of the non–anchor nodes as accurately as possible, we propose to concurrently minimize two objective functions. Let $\hat{\mathbf{p}}_i = (\hat{x}_i, \hat{y}_i), i = m+1, \ldots, n$ be the estimated positions of the non–anchor nodes i. The first objective *CF* is defined as

$$
CF = \sum_{i=m+1}^{n} \left(\sum_{j \in N_i} \left(\hat{d}_{ij} - d_{ij} \right)^2 \right), \quad (4)
$$

where $\hat{d}_{ij}$ is the estimated distance between nodes i and j computed as follows:

$$
\hat{d}_{ij} = \begin{cases} \sqrt{(\hat{x}_i - x_j)^2 + (\hat{y}_i - y_j)^2}, & \text{if node } j \text{ is an anchor,} \\ \sqrt{(\hat{x}_i - \hat{x}_j)^2 + (\hat{y}_i - \hat{y}_j)^2}, & \text{otherwise.} \end{cases}
$$
$$(5)$$

Thus, *CF* is the sum of the squared differences between the distances corresponding to the candidate geometry (as given by the estimated positions $\hat{\mathbf{p}}_i, i = m+1, \ldots, n$ of the non–anchor nodes and the positions of the anchor nodes) and the measured data.

Given a set of data consisting of the set of anchor nodes and the inter–node distance measurements, a network is said to be localizable if there is only one possible geometry compatible with the data. Localizability is a fundamental problem which can be studied within the framework of *rigid graph theory*. If the network is not localizable, then multiple global minima will be present in the *CF*, with only one of them corresponding to the actual geometry of the deployment. Thus, in settings which are close to not being localizable, any localization algorithm will become extremely sensitive to these false minima of *CF*, resulting in very large localization errors [10].

The simplest effect leading to lack of localizability is the so–called *flip ambiguity* phenomenon, shown in Fig. 1. The neighbors of node i (*i.e.* nodes j, k, l and m) are almost collinear (double line in the figure), and thus, it is clear that if the location of node i is flipped with respect

to this line to the new position denoted by i', then the new geometry so obtained is almost compatible with the original inter–node distance measurements (it would be fully compatible if nodes j, k, l and m were perfectly aligned). In order to counteract this lack of localizability, connectivity considerations are helpful: by analyzing Fig. 1, one can observe that whereas the flipped position i' is within the communication range of node n (shown by the circle in the figure), the actual position i is not. If the network is sufficiently dense, one would expect false minima of *CF* to violate some connectivity constraints of this sort. The number of these violations in a candidate topology constitutes our second objective function *CV*.
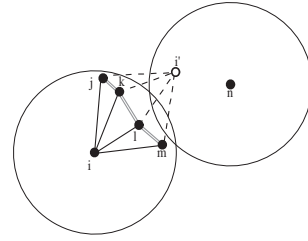

Figure 1. The flip ambiguity problem.

Formally, *CV* counts the number of connectivity constraints which are not satisfied by the candidate geometry, and is defined as

$$
CV = \sum_{i=1}^{n} \left(\sum_{j \in N_i} \delta_{ij} + \sum_{j \in \overline{N}_i} (1 - \delta_{ij}) \right), \quad (6)
$$

where $\delta_{ij} = 1$ if $\hat{d}_{ij} > R$ and 0 otherwise.

In order to evaluate the accuracy of the estimates, we consider the *normalized localization error* (NLE), defined as

$$
NLE = \frac{1}{R} \sqrt{\frac{1}{(n-m)} \sum_{i=m+1}^{n} \left(\|\mathbf{p}_i - \hat{\mathbf{p}}_i\|^2 \right)} \times 100\%. \quad (7)
$$

Thus, assuming that the estimate is unbiased, NLE can be interpreted as the ratio of the standard deviation to the connectivity radius.

C. Geometrical constraints

The connectivity ranges and the positions of the anchor nodes determine subsets of the overall search space where each single non–anchor node can be positioned. These subsets depend on the type of non–anchor node, and can be defined by means of geometrical constraints. We adopt the following classification based on the position of a non–anchor node with respect to anchor nodes:

- *Class 1 node:* a non–anchor node which is first–level neighbor to at least one anchor node.

If a node belongs to Class 1, then its position must lie within the intersection of the circles of radius R centered in the anchor nodes which it is neighbor to.

- *Class 2 node:* a non–anchor node which is second–level neighbor to at least one anchor node.

  If a node belongs to class 2, then its position must lie within the intersection of the annuli with inner and outer radii R and $2R$, respectively, centered in the anchor nodes which it is second–level neighbor to.

- *Class 3 node:* a non–anchor node which belongs to neither class 1 nor class 2.

  If a node is a member of class 3, then its position must lie outside the union of the circles of radius R centered in all anchor nodes.

It is clear that, given a non–anchor node belonging to one of the three classes, it is possible to restrict the space where it can be located. This information can be exploited both in the generation of the initial population of the MOEA and, during the evolutionary process, in the application of the mating operators. By avoiding the generation of solutions which certainly cannot be optimal (since they violate the geometrical constraints determined by the connectivity ranges and by the known anchor node positions), the execution of the evolutionary algorithm can be sped up. Furthermore, these constraints help alleviate the localizability issues and in particular the flip ambiguity effect. Indeed, this phenomenon is much more likely to occur if the candidate positions of the non–anchor nodes are not constrained within the subspace corresponding to its membership class.

III. THE OPTIMIZATION FRAMEWORK

MOEAs aim to search for optimal solutions to problems that incorporate multiple performance criteria by generating a family of equally valid solutions, where each solution tends to satisfy a criterion to a higher extent than another. Different solutions are compared with each other by using the notion of Pareto dominance [11].

We have used PAES as MOEA in our experiments. PAES probably represents the most elementary nontrivial algorithm capable of generating diverse solutions in the Pareto optimal front. Further, it is characterized by a lower computational complexity than traditional niching methods. Due to space limitation the details of the algorithm are omitted: the interested reader car refer to [12].

In our optimization framework each chromosome encodes the positions of all non–anchor nodes in the network. Thus, each chromosome consists of $n - m$ pairs of real numbers, where each pair represents the coordinates $\hat{x}$ and $\hat{y}$ of a non–anchor node. The variation range of each coordinate is bounded by the geometrical constraints described in Section II-C. We enforce compliance with these constraints in the initial population. Further, whenever mutations are applied during the evolutionary process, only mutated individuals satisfying these constraints are generated. Each

chromosome is associated with a vector of two elements, which represent the values of the two objective functions *CF* and *CV* (Eqs. (4) and (6) in Section II-B).

We have defined two mutation operators. The first mutation operator, denoted *Node Mutation* operator, performs a uniform–like mutation: the position of each non–anchor sensor node is mutated with probability $P_U = 1/(n - m)$. Positions are randomly generated within the geometrical constraints imposed on the specific node location. The second mutation operator, denoted *neighborhood mutation* operator, is applied when the first operator is not. The neighborhood mutation operator mutates, with probability $P_U = 1/(n - m)$, the position of each non–anchor sensor node within the geometrical constraints determined for the specific node, but unlike the first operator, it applies the same rigid translation, which has brought the mutated node i from the pre–mutation position to the post–mutation position, to the neighbors of i with a certain probability (denoted *rigid translation probability*). As we have already discussed in [8], this operator results to be particularly suitable for dealing with specific topological configurations.

IV. SIMULATION RESULTS

In this section we show the effectiveness of the proposed two–objective evolutionary algorithm in tackling the localization problem in WSNs. We have built different network topologies by randomly placing 200 nodes with a uniform distribution in $\mathcal{T}$. We have varied the percentage of anchor nodes to 8%, 10% and 12% (thus each topology consists of 16, 20 and 24 anchor nodes and 184, 180 and 176 non–anchor nodes, respectively). Further, we have varied the connectivity radius R in the interval $[0.11, 0.16]$ with step 0.01. The distance measurements between neighboring nodes were generated according to the model (1). We assume that these distance estimates are derived from RSS measurements, which are commonly affected by log–normal shadowing with standard deviation of the errors proportional to the actual r_{ij} [1]. Thus, the variance of e_{ij} is given by $\sigma^2 = \alpha^2 r_{ij}^2$. A value of $\alpha = 0.1$ was used in the simulations.

For each value of R, 10 random network topologies were generated. For each topology, a different scenario was built for each percentage of anchor nodes. Thus, we were able to perform an evaluation on the effects of both the different connectivity radii and the different percentages of anchor nodes on the normalized localization error. Table I shows the average values of some network indicators, namely the node degree (considering anchor and non–anchor nodes), the percentage of non–anchor nodes classified in Class 1 and Class 2 (the percentage of nodes in Class 3 can be easily deduced from the first two), the percentage of non–anchor nodes with no anchor node in their neighborhoods and the percentage of non–anchor nodes having at least 3 anchor nodes in their neighborhoods. The analysis of Table I reveals that, when the connectivity radius and the percentage

R	node degree	anchor (%)	Cl.1 (%)	Cl.2 (%)	0 anch. (%)	3(or more) anch.(%)
0.11	6.86	8	42.28	31.68	57.72	2.28
		10	50.44	32.72	49.56	3.56
		12	56.02	30.68	43.98	5.97
0.12	8.09	8	51.03	31.85	48.97	2.12
		10	59.67	29.22	40.33	4.50
		12	66.48	25.80	33.52	5.85
0.13	9.41	8	58.04	32.72	41.96	2.17
		10	65.28	29.72	34.72	5.33
		12	69.72	26.14	30.28	9.43
0.14	10.84	8	56.52	30.43	43.48	5.98
		10	67.28	25.94	32.72	8.94
		12	75.06	20.68	24.94	14.20
0.15	12.28	8	62.01	29.35	37.99	7.66
		10	70.94	24.11	29.06	12.78
		12	75.74	21.42	24.26	17.16
0.16	14.16	8	67.83	26.20	32.17	11.09
		10	74.17	22.72	25.83	17.22
		12	81.36	16.76	18.64	24.55

Parameter	Value
Size of non–dominated solution archive	20
Number of divisions of the objective space	5
Number of fitness evaluations	4×10^5
Node mutation probability	0.9
Rigid translation probability	0.3

of anchor nodes are low, the localization problem is very complex: indeed, with a connectivity radius $R = 0.11$ and 8% of anchor–nodes, only a small fraction of non–anchor nodes can rely on 3 or more anchor neighbors (2.28%), while more than the half of them (57.72%) are in communication with no anchor node. Moreover it is worth noting that, even when the connectivity radius is increased to 0.16 and the percentage of anchor nodes to 12%, the average percentage of non–anchor nodes with no anchor neighbor is not negligible (18.64%), while the average percentage of non–anchor nodes with 3 or more anchor neighbors is still significantly low (24.55%). For each scenario, 15 trials of PAES were executed, with parameter values summarized in Table II.

Once the Pareto front approximation has been generated, a solution must be chosen. In our experiments, we verified that the variation interval of *CF* for the solutions on the final Pareto front approximation is quite small. Thus, we can assume that each solution on the Pareto front can be acceptable with respect to the *CF* objective. We have validated this hypothesis with a Wilcoxon test, by selecting from each final archive the solutions characterized by the minimum value of *CV* and the minimum value of *CF* (in practice, the solutions on the extremes of the Pareto front approximation). Since no statistical difference exists

in terms of NLE among the solutions in the final Pareto front approximation, each solution can be actually selected in order to perform a comparison with the FSDPr algorithm. For the sake of brevity, we take the solution characterized by the lowest value of *CF*.

Regarding the FSDPr algorithm, in order to select the optimum value of the regularization term (λ) we have adopted the following heuristic strategy. Given a scenario, we first solve the non–regularized problem and then exploit the non–regularized solution to compute the upper bound of λ (λ^*), as suggested in [3]. λ^* is used as starting value in the main tuning loop, where the regularized problem is solved; if the current regularized solution is not feasible, then the current value of λ is divided by two and the new regularized problem is solved again, until a feasible solution is obtained or a maximum number of tries ($MT = 5$) is reached. In the latter case, the regularized solution coincides with the non–regularized one (*i.e.*, $\lambda = 0$).

In Fig. 2, we have plotted as solid lines the average NLE obtained by FSDPr and PAES algorithms versus the six values of radius R used in the experiments. Further, we have shown as dotted lines the average NLE obtained by applying the gradient refinement described in [3], [9] to the final solutions computed by the two algorithms (we denote these approaches as FSDPr+REF and PAES+REF, respectively).

The analysis of the figures highlights that the gradient refinement is able to improve the estimation only when it is already sufficiently accurate. Indeed, if we consider the solutions generated by FSDPr for the lowest connectivity radius ($R = 0.11$), we realize that the gradient method is unable to perturb the estimation out of the reached local minimum. On the contrary, when the solutions are characterised by a low NLE, the gradient method is able to improve them. In particular, the almost constant gap between the solid lines and the dotted lines for PAES suggests a stable improvement introduced by the refinement phase.

The NLEs obtained by PAES are comparable to those obtained by FSDPr+REF when the connectivity radii are sufficiently high ($R \geq 0.14$), while they are slightly lower when $R < 0.14$. Further, in all the experiments, PAES+REF considerably outperforms FSDPr+REF. For example, when the percentage of anchor nodes is 8% and the connectivity radius is 0.11, from Fig. 2(a) we can derive that PAES+REF provides estimations which are on average 36.57% more accurate than FSDPr+REF. This percentage increases to 45.75% when the connectivity radius is equal to 0.16. Similar conclusions can be deduced by analysing Figs. 2(b)- 2(c), which show the results obtained by using the 10% and 12% of anchor nodes: PAES+REF generate solutions which are 57.84% and 53.72% (61.79% and 48.87%) more accurate when the percentage of anchor nodes is equal to 10% (12%) and the connectivity radius is 0.11 and 0.16.

(a) 8% of anchor nodes

(b) 10% of anchor nodes
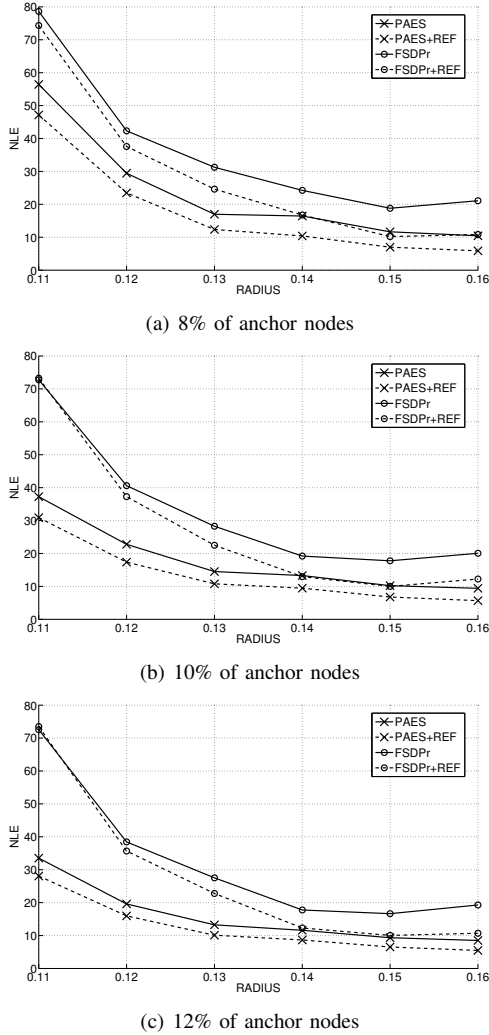

(c) 12% of anchor nodes

Figure 2. Comparison between PAES and FSDPr without (solid line) and with (dotted line) gradient refinement (*REF*) using different percentages of anchor nodes and different connectivity radii.

V. Conclusions

In this paper we have proposed a two–objective evolutionary algorithm able to accurately solve the fine–grained localization problem in WSNs. The problem is not new in the literature, since several techniques have been proposed in the last decade. The novelty of the approach relies on a better exploitation of the connectivity graph so as to define topological constraints to be used as a second objective function in a multi–objective optimization framework. The topological constraints define zones of the space where each sensor can or cannot be located, thus reducing the search space of the evolutionary algorithm and contextually the chance of ambiguously flipping node locations. Moreover we have discussed the possibility of using a standard gradient–based technique able to refine the final estimation produced by the evolutionary algorithm. We

have shown that the proposed approach is able to solve the localization problem with high accuracy for a number of different topologies, connectivity ranges and percentages of anchor nodes. Further, we have discussed how our approach outperforms the regularized version of the standard semi-definite programming technique.

References

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

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

[3] P. Biswas, T.-C. Liang, K.-C. Toh, Y. Ye, and T.-C. Wang, "Semidefinite programming approaches for sensor network localization with noisy distance measurements," *IEEE Trans. Autom. Sci. Eng.*, vol. 3, no. 4, pp. 360–371, 2006.

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

[5] P. Biswas and Y. Ye, "Semidefinite programming for ad hoc wireless sensor network localization," in *IPSN '04: Proc. of the 3rd Int. Symp. on Information Processing in Sensor Networks*, 2004, pp. 46–54.

[6] P. Tseng, "Second–order cone programming relaxation of sensor network localization," *SIAM J. on Optimization*, vol. 18, no. 1, pp. 156–185, 2007.

[7] A. A. Kannan, G. Mao, and B. Vucetic, "Simulated annealing based wireless sensor network localization with flip ambiguity mitigation," in *Proc. of the 63-rd IEEE Vehicular Technology Conference*, 2006, pp. 1022–1026.

[8] M. Vecchio, R. López-Valcarce, and F. Marcelloni, "A two-objective evolutionary approach based on topological constraints for node localization in wireless sensor networks," *Applied Soft Computing*, vol. In Press, Corrected Proof, pp. – , 2011.

[9] P. Biswas, T.-C. Lian, T.-C. Wang, and Y. Ye, "Semidefinite programming based algorithms for sensor network localization," *ACM Trans. Sen. Netw.*, vol. 2, pp. 188–220, May 2006.

[10] A. A. Kannan, B. Fidan, and G. Mao, "Analysis of flip ambiguities for robust sensor network localization," *IEEE Trans. Vehicular Technology*, vol. 59, no. 4, pp. 2057–2070, 2010.

[11] F. Marcelloni and M. Vecchio, "Enabling energy-efficient and lossy-aware data compression in wireless sensor networks by multi-objective evolutionary optimization," *Information Sciences*, vol. 180, no. 10, pp. 1924–1941, 2010.

[12] J. D. Knowles and D. W. Corne, "Approximating the nondominated front using the Pareto Archived Evolution Strategy," *IEEE Trans. Evol. Comput.*, vol. 8, no. 2, pp. 149–172, 2000.