2013 European Control Conference (ECC)
July 17-19, 2013, Zürich, Switzerland.

Network Localization by Shadow Edges

Gabriele Oliva*, Stefano Panzieri†, Federica Pascucci† and Roberto Setola*
* University Campus Biomedico of Rome, Italy
g.oliva@unicampus.it r.setola@unicampus.it
† Dipartimento di Ingegneria, University "Roma TRE",
Via della Vasca Navale, 79, 00146, Roma, Italy.
panzieri@uniroma3.it pascucci@dia.uniroma3.it

***Abstract*— Localization is a fundamental task for sensor networks. Traditional network localization approaches allow to obtain localized networks requiring the nodes to be at least tri-connected (in 2D), i.e., the communication graph needs to be globally rigid. In this paper we exploit, besides the information on the neighbors sensed by each robot/sensor, also the information about the lack of communication among nodes. The result is a framework where the nodes need to be at least bi-connected and the communication graph has to be rigid. This is possible considering a novel typology of link, namely Shadow Edge, that accounts for the lack of communication among nodes and allows to reduce the uncertainty associated to the position of the nodes.**

I. Introduction

Location service is a fundamental building block of many emerging computing/networking paradigms. Sensor location information is of the utmost importance for both sensor data integrity and important network management issues such as coverage and data delivery. Most of the approaches proposed in literature rely on Global Navigation Satellite System (GNSS) [7]. However, GNSS is not suitable for large-scale sensor localization due to its high cost, large form factor and, least but not last, environmental constraints. GNSS, indeed, requires direct line of sight to satellites. As a consequence, it does not work indoor, underground, and underwater. Moreover, due to the well known urban-canyon problem, GNSS provides poor accuracy in large metropolitan areas.

Recently, novel schemes have been proposed to determine the locations of the nodes in a network by exploiting some special nodes (called *anchors*) which know their locations. According to this approach, localization algorithms derive the locations of sensor nodes from local measurements such as relative distance and angle estimations between neighbors [3], [4], [14]. Existing localization algorithms require either the connectivity graph [19], or the distances between neighboring sensor nodes [3], [4], [14], [20] as input. One major challenge using this approach is localization ambiguity, since multiple, different localization solutions can satisfy all the distance constraints even if they are far from each other [22].

The problem of whether a graph with given edge length constraints admits a unique embedding in the plane is studied by rigidity theory [11]. A graph is rigid in the plane if one cannot continuously deform the shape of the graph without modifying the lengths of the edges. A graph is globally rigid if it admits a unique embedding in the plane, subject to global rotations and translations. The theory of graph rigidity in 2D has been widely studied and well understood.

In graph rigidity literature, many efforts have been made to explore the combinatorial conditions for rigidity. The Laman condition [11] characterizes graphs that are generically rigid. An efficient algorithm, the pebble game [10], is able to test whether a graph is generically rigid in time $O(nm)$ where n is the number of nodes and m is the number of edges. Similarly, for global rigidity, a sufficient and necessary condition [8] based on the results in [6] has been identified by combining both redundant rigidity and 3–connectivity. Both a combinatorial characterization of globally rigid graphs and polynomial algorithms for testing such graphs are however, not trivial to apply in the development of efficient localization algorithms. Given a graph with edge lengths specified, finding a valid graph realization in $\mathbb{R}^d$ for a fixed dimension d is an NP-complete problem [18].

Recently, Jackson and Jordan [9] proved a sufficient condition based on 6 mixed connectivity, which improves a previous result of 6–connectivity by [13]. There are also some results for random geometric graphs. Assuming the unit disk model, many researchers [5], [12], [17], [21] considered critical conditions for graph connectivity. Simulation results [3] ensure that the hitting radius of global rigidity is between 3– and 6–connectivity in a probabilistic sense.

To the best of our knowledge, all the approaches proposed in literature partially exploit the topology information. They consider as good, indeed, only the information of *being connected* and completely discarded the one of *not being connected*. This papers shows how to exploit connectivity graph taking into account both the information on connectivity (i.e., the presence of an edge between two nodes) and lack of connectivity (i.e., the absence of a link). Specifically the *being connected* relationship carries information about where a node is located (i.e., *admissible regions*), while the *not being connected* one carried information about where a node cannot be located (i.e., *forbidden regions*). In this paper the problem of finding a unique realization is addressed by exploiting both the admissible and forbidden region actively. According to this approach the conventional rigidity requirement is no longer a necessary condition to find a unique realization. To this end, rigid graphs become only a subset of the total set of networks where a unique solution can be found. To demonstrate this, the notions of *shadow edges* and *shadow graph* are introduced and the realization problem is studied considering a combination of both regular and shadow edges. Furthermore, a localization algorithm is built exploiting shadow edges and shadow graph.

The paper is organized as follows: Section II provides a general overview and some preliminaries results on localization and rigid graphs; the proposed approach is detailed in Section III;some simulation results and comparison with standard localization algorithm are provided in Section IV; finally, some remarks and future work directions are collected in Section V.

II. Localization and Rigidity

In this section, the network localization problem with distance information is addressed: as explained above, it can be studied in the framework of graph theory.

To this end, we need some initial definitions, well known in the literature.

Definition 1: Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ be a *graph* with n nodes, where the set $\mathcal{V}$ denotes the nodes $v_1, \ldots, v_n$; $\mathcal{E}$ is the set of edges (v_i, v_j). The graph can be also represented by means of an *adjacency matrix* $\Gamma = \{\gamma_{ij}\}$ that is composed of non-negative entries $\gamma_{ij} = 1 \Leftrightarrow (v_i, v_j) \in \mathcal{E}$, i.e., there exists an arc that starts form node v_i and reaches node v_j (note that it is also possible to consider non-unitary weights).

The graph $\mathcal{G}$ is said to be *undirected* if $(v_j, v_i) \in \mathcal{E}$ whenever $(v_i, v_j) \in \mathcal{E}$ (the weights are $\gamma_{ij} = \gamma_{ji}$ in this case); otherwise the graph $\mathcal{G}$ is said to be *directed*.

Definition 2: Let a set of n sensors $\Sigma = \{\sigma_i\}$ embedded in $\mathbb{R}^d$, and let d_{ij} be the distances between sensors σ_i and σ_j. Suppose that the coordinates $p_i \in \mathbb{R}^d$ of small number of sensors $\sigma_j \in \Sigma_j \subseteq \Sigma$ are known, and denote Σ_j as the *kernel sensor set*. Each sensor σ_i is assumed to be located at a fixed position in $\mathbb{R}^d$ and is associated with a specific set of neighboring sensors. In this paper only symmetric neighboring relation is considered, thus a sensor σ_j is a neighbor of a sensor σ_i if and only if sensor σ_i is also a neighbor of sensor σ_j. The *localization problem* consists in finding a map $Q : \Sigma \to \mathbb{R}^d$ (where d is 2 or 3) which assigns coordinates $p_i \in \mathbb{R}^d$ to each sensor σ_i such that $||p_i - p_j|| = d_{ij}$ holds for all pairs (i, j) for which d_{ij} is given, and the assignment is consistent with any sensor coordinate assignments provided in the problem statement.

Under these conditions, a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ can be correlated with a sensor network by associating a vertex v_i of the graph with each sensor σ_i, and an edge of the graph (v_i, v_j) with each sensor pair σ_i, σ_j for which the inter-sensor distance is known.

Definition 3: A *d–dimensional framework* $(\mathcal{G}, Q)$ is a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ together with a map $Q : \mathcal{V} \to \mathbb{R}^d$. The framework is a *realization* if it results in $||p_i - p_j|| = d_{ij}$ for all pairs (σ_i, σ_j) such that $(v_i, v_j) \in \mathcal{E}$. According with this approach, the localization over Σ problem is mapped into a realization problem over $(\mathcal{G}, Q)$.

Definition 4: The network localization problem just formulated is said to be *solvable* if there is exactly one set of vectors $\{p_1, p_2, \ldots p_n\}$ is consistent with the given data $(\mathcal{G}, Q)$.

To understand the solvability of the localization problem, let us consider the notions of rigidity and global rigidity.

Definition 5: Two frameworks $(\mathcal{G}, Q)$ and $(\mathcal{G}, Q^*)$ are *equivalent* if $||p_i - p_j|| = ||p_i^* - p_j^*||$ holds for all pairs (i, j) with $(v_i, v_j) \in \mathcal{E}$.

Definition 6: Two frameworks $(\mathcal{G}, Q)$ and $(\mathcal{G}, Q^*)$ are *congruent* if $||p_i - p_j|| = ||p_i^* - p_j^*||$ holds for all pairs i, j with $v_i, v_j \in \mathcal{V}$.

This is the same as saying that $(\mathcal{G}, Q)$ can be obtained from $(\mathcal{G}, Q^*)$ by an isometry over $\mathbb{R}^d$, i.e., a combination of translations, rotations and reflections.

Definition 7: A framework $(\mathcal{G}, Q)$ is *rigid* if there exists a sufficiently small positive ϵ such that if $(\mathcal{G}, Q)$ is equivalent to $(\mathcal{G}, Q^*)$ and $||p_i - p_i^*|| < \epsilon$ for all $v_i \in \mathcal{V}$ then $(\mathcal{G}, Q)$ is congruent to $(\mathcal{G}, Q^*)$.

Intuitively, a rigid framework cannot flex.

Definition 8: A framework $(\mathcal{G}, Q)$ is *globally rigid* if every framework which is equivalent to $(\mathcal{G}, Q)$ is congruent to $(\mathcal{G}, Q)$.

Obviously, if $\mathcal{G}$ is a complete graph then the framework $(\mathcal{G}, Q)$ is necessarily globally rigid.

According to this approach, a localization problem can be solved only if the graph framework $(\mathcal{G}, Q)$ is globally rigid (i.e., otherwise the position of the sensors would not be univocally determined, since flipping some node would result in another admissible solution).

Although the realization of general globally rigid graphs is hard, a class of globally rigid graphs can be computationally efficient to realize by means of *trilateration*. Trilateration is the operation whereby a node with known distances from three $(d = 2)$ or four $(d = 3)$ other nodes, determines its own position in terms of the positions of those neighbors. An easy way to obtain a localizable network (i.e., globally rigid graph) is by iteratively adding nodes attached to at least three (four) nodes. This in the following will be referred to as *Trilateration Network Construction* (TNC) algorithm. The trilateration test can be efficiently used to localized nodes in a sensor networks.

In the following, we demonstrate that considering also information about non-neighboring nodes, the global rigidity assumption can be relaxed and a different and more effective procedure to build localizable network can be set up.

It is worth to underline that, given the graph and distance set of a globally rigid framework, there is not enough information to position the framework absolutely in $\mathbb{R}^d$. To do this requires the absolute position of at least three $(d = 2)$ or four vertices $(d = 3)$ non-collinear vertices, i.e., *kernel* nodes.

III. Localization and Shadow Edges

In the following we will assume to test the localizability of a network by testing the localizability of a node on the basis of the information provided by its neighbors.

Specifically, each node i is provided with a maximum communication radius ρ (assumed to be the same for all the agents) and is able to detect the presence of each and every node j that falls within such communication radius, obtaining also an information on the distance $d_{ij} \leq \rho$ among them (we assume $d_{ij} = d_{ji}$).

***Definition** 9:* The set of *localization options* $\mathcal{L}_i(\mathcal{G})$ for a node v_i in a graph $\mathcal{G}$, is the set of points $(x, y) \in \mathbb{R}^2$ that are admissible for the position of node v_i, given the structure of the graph $\mathcal{G}$.

***Definition** 10:* A node v_i is *localized* over a graph $\mathcal{G}$ provided that its position is univocally determined. Otherwise the node is not localized.

***Definition** 11:* A graph $\mathcal{G}$ is *localized* provided that each node $v_i \in \mathcal{V}$ is localized.

It is worth noticing that a node v_i, being able to sense its neighbors v_j within the communication range ρ, is able to obtain the distance d_{ij}; moreover, if the nodes v_j are localized, their position is known. Therefore, assuming the graph $\mathcal{G}$ to be connected the following options are possible, depending on the cardinality $|\mathcal{L}_i(\mathcal{G})|$:

- $|\mathcal{L}_i(\mathcal{G})| = 1$ if and only if node v_i is connected to at least 3 non collinear localized nodes;
- $|\mathcal{L}_i(\mathcal{G})| = 2$ if and only if node v_i is connected to 2 localized nodes;
- $|\mathcal{L}_i(\mathcal{G})| = \infty$ if node v_i is connected to 1 localized node

***Definition** 12:* A *sensing disk* $S_{i,h}$ for a node v_i is a disk or radius ρ centered in a localization option $(x_h, y_h) \in \mathcal{L}_i(\mathcal{G})$, i.e.,

$$S_{i,h} = \{(x, y) \in \mathbb{R}^2 : (x - x_h)^2 + (y - y_h)^2 \leq \rho^2\}. \quad (1)$$

***Definition** 13:* The *admissible sensing region* for a node v_i, denoted by $\mathcal{D}_i$ is the union of the sensing disks $S_{i,h}$ for each of the localization options $(x_h, y_h) \in \mathcal{L}_i(\mathcal{G})$, i.e.

$$\mathcal{D}_i = \{(x, y) \in \mathbb{R}^2 : (x - x')^2 + (y - y')^2 \leq \rho^2, (x', y') \in \mathcal{L}_i\}.$$

As discussed above, if each node is connected to at least 3 non collinear localized nodes, the network is localized. However it will be shown that, if a node v_i is such that $|\mathcal{L}_i(\mathcal{G})| = 2$ (e.g., it is connected to 2 localized nodes), it may still be possible to localize it. To this end we need to define the following edge class.

***Definition** 14:* A *shadow edge* is an edge that connects two nodes v_i and v_j if and only if one of the following holds true:

- v_i is localized over $\mathcal{G}$ and $|\mathcal{L}_j(\mathcal{G})| = 2$; $\mathcal{L}_i(\mathcal{G}) \in \mathcal{D}_j$ but $d_{ij} > \rho$;
- v_j is localized over $\mathcal{G}$ and $|\mathcal{L}_i(\mathcal{G})| = 2$; $\mathcal{L}_j(\mathcal{G}) \in \mathcal{D}_i$ but $d_{ji} > \rho$.

Such an edge can be created when node v_i is not detected by v_j but $v_i \in \mathcal{D}_j$ or vice versa. Hence, it represents the fact that node v_i has 2 localization options and one of them should not be taken into account; in fact if node v_i was in the localization option such that v_j lies in the corresponding disk, it should have sensed node v_j. Therefore node v_i it is in the other localization option. Note that the node v_j cannot lie in the intersection of the 2 disks, otherwise it would have been connected to v_i with a regular edge. Figure 1 shows an example of localization by means of a shadow edge.

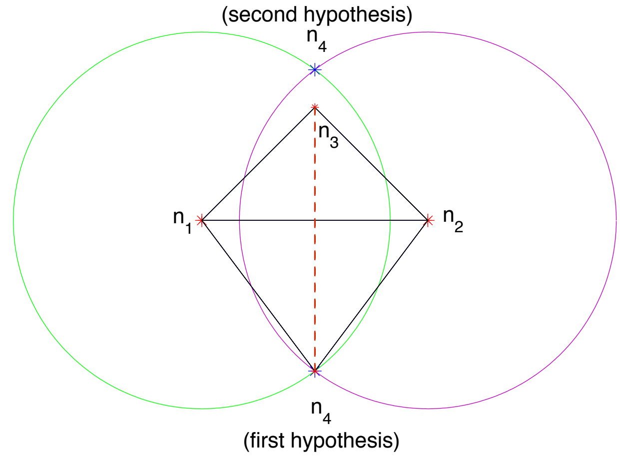

Fig. 1. Example of shadow edge: n_4 senses nodes n_1 and n_2, two hypotheses are possible for its position (i.e., the intersection of the green and purple circumferences). Since node n_3 is not detected by n_4, it is possible to exclude hypothesis 2 and node n_4 is localized by adding a shadow edge (red dotted line) between n_4 (first hypothesis) and n_3.

***Definition** 15:* A *shadow graph* is the graph

$$\mathcal{G}_s = \{\mathcal{V}, \mathcal{E}_s\}$$

where $\mathcal{V}$ is the set of nodes of graph G and $\mathcal{E}_s$ is the set of the shadow edges.

***Definition** 16:* A *extended shadow graph* is the graph

$$\mathcal{G}_e = \{\mathcal{V}, \mathcal{E} \cup \mathcal{E}_s\}$$

where $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ is the nominal graph and $\mathcal{E}_s$ is the set of the shadow edges.

In the following we will assume, for the sake of simplicity and without loss of generality, that the event of connecting a node v_i to three collinear nodes v_j, v_k, v_h either in $\mathcal{G}$ and in $\mathcal{G}_e$ is never verified.

***Definition** 17:* A *minimal shadow graph* is the graph

$$\mathcal{G}_s = \{\mathcal{V}, \mathcal{E}_s^*\}$$

where $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ is the nominal graph and $\mathcal{E}_s^*$ is a *minimal set of shadow edges* defined as follows:

$$\forall i : |\mathcal{L}_i(\mathcal{G})| = 2 \quad \exists! \text{ shadow edge } (v_i, v_j) \in \mathcal{E}_s^*.$$

Let us now provide some results on localization using such shadow edges.

***Lemma** 1:* Suppose that a graph $\mathcal{G}$ is connected and that $\mathcal{E}_s$ is a set of shadow edges. Then the nodes are localized over $\mathcal{G}_e$ if and only if the extended shadow graph $\mathcal{G}_e$ is globally rigid.

Proof: Clearly, if $\mathcal{G}_e$ is globally rigid, then $|\mathcal{L}_i(\mathcal{G}_e)| = 1$ for each node $i = 1, \dots, n$ and the nodes are localized. To prove the reversed implication, suppose that the nodes are all localized over $\mathcal{G}_e$ but $\mathcal{G}_e$ is not globally rigid: then there is at least a node i such that $|\mathcal{L}_i(\mathcal{G})| \geq 2$, but this is not possible, since the network is localized, yielding to an absurd. ■

The above Lemma, therefore, allows to infer the localization of the nodes in $\mathcal{G}$ by considering the union of the regular edges and of the shadow edges. According to such an approach, let us provide the following Theorem.

Theorem 1: Suppose that the graph $\mathcal{G}$ is connected and that $\mathcal{E}_s$ is a set of shadow edges; then the following propositions are equivalent:

1) the nodes are localized over $\mathcal{G}_e$;
2) $\max_{i=1,\dots,n} |\mathcal{L}_i(\mathcal{G})| \leq 2$ and each node $v_i \in \mathcal{G}$ such that $|\mathcal{L}_i(\mathcal{G})| = 2$ is connected to at least 2 localized nodes and there is at least a shadow edge (v_i, v_j) with $v_j \in \mathcal{V}$
3) $\mathcal{G}$ is rigid and there is a minimal shadow edge set $\mathcal{E}_s^* \subseteq \mathcal{E}_s$.

Proof: Let us first prove 1) $\Leftrightarrow$ 2). Note that, if the assumptions in 2) are verified, then each node in $\mathcal{G}_e$ is at least tri-connected and the nodes are localized. Conversely, assume that the nodes are not localized over $\mathcal{G}_e$ and that the hypotheses in 2) hold: then there is at least a node v_i that is not localized. This implies $|\mathcal{L}_i(\mathcal{G}_e)| \geq 2$, which is contrast with the hypotheses in 2). Let us now prove 2) $\Leftrightarrow$ 3). Note that, if m is the number of nodes such that $|\mathcal{L}_i(\mathcal{G})| = 2$, then choosing at least a shadow edge for each of such nodes implies $|\mathcal{E}| \geq m$. A minimal shadow edges set is such that

$$|\mathcal{E}_s^*| = \sum_{i=1}^{p} |\mathcal{L}_i| - n = m$$

hence $\mathcal{E}_s^* \subseteq \mathcal{E}_s$. Conversely, choosing a minimal set of shadow edges implies 2) if $\max_{i=1,\dots,n} |\mathcal{L}_i(\mathcal{G})| \leq 2$. Note that if $\mathcal{G}$ is not rigid, then there is at least a node v_i with $|\mathcal{L}_i(\mathcal{G})| = \infty$; conversely if there is at least a node v_i with $|\mathcal{L}_i(\mathcal{G})| = \infty$ the graph $\mathcal{G}$ is not rigid. ■

Algorithm 1: Node Localization Check

Data: Undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, node v_i
Result: success or fail

```
if |L_i(G)| ≥ 3 then
    return fail;
else
    Set c_j = ∃(v_i, v_j) ∈ E : d_ij ≤ ρ ∧ |L_j(G)| = 1;
    Set
    c_k = ∃(v_i, v_k) ∈ E : v_k ≠ v_j ∧ d_ik ≤ ρ ∧ |L_k(G)| = 1;
    Set c_h = ∃v_h ∈ V :
    d_ih > ρ ∧ |L_h(G)| = 1 ∧ L_h(G) ⊂ D_i;
    if not(c_j ∧ c_k ∧ c_h) then
        return fail;
    else
        return success;
    end
end
```

Let us now provide a criterion to iteratively test the localizability of a network. This criterion does not require the graph to be rigid.

Corollary 1: Consider a constructive localization algorithm network starting with 3 connected and localized nodes, then if a node v_i is localizable by trilateration or is such that it is connected to at least 2 non collinear preexisting nodes v_j, v_k having $|\mathcal{L}_j(\mathcal{G})| = |\mathcal{L}_k(\mathcal{G})| = 1$ and it is possible to find at least a shadow edge (v_i, v_k), then the resulting network is localized.

Proof: Note that $|\mathcal{L}_i(\mathcal{G})| \leq 2$ for all $v_i \in \mathcal{V}$ and there is a shadow edge for each node v_i such that $|\mathcal{L}_i(\mathcal{G})| = 2$; therefore, according to Theorem 1, the nodes are localized. ■

Exploiting Theorem 1 and Corollary 1 it is possible to build a localization method (Algorithm 2) based on shadow edges. The method, starting with a core of 3 localized nodes and $n - 3$ nodes, whose positions are randomly generated, tests the localizability of nodes by using Algorithm 1. The procedure is iterated until no more nodes can be localized, thus providing the maximum localizable subgraph $\mathcal{G}_l = (\mathcal{V}_l, \mathcal{E}_l)$ according to shadow edge definition. Notice that $\mathcal{N}_{\mathcal{G}_l}(v_j)$ denotes the neighbor of v_j in the localizable subgraph, *Trilaterate* performs the trilateration and *TrilaterateBySE* the trilateration based on shadow edges.

Algorithm 2: Localized Subgraph

Data: Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with 3 connected and localized nodes. Target size $n > 3$ of the graph.
Result: Position (x_j, y_j) for each node $v_i \in \mathcal{V}_l$

```
/* Initialisation                                   */
Set i = 2;
Set |V_l(i − 1)| = 2;
Set |V_l(i)| = 3;
/* Main Cycle                                       */
while |V_l(i)| > |V_l(i − 1)| do
    for j=1 to n do
        if NodeLocalizationCheck (G, v_j) then
            Set V_l = V_l ∪ v_j;
            if N_{G_l}(v_j) ≥ 3 then
                Set (x_j, y_j)= Trilaterate (G_l, v_j);
            else
                Set (x_j, y_j)= TrilaterateBySE (G_l, v_j);
            end
        end
    end
    Set i = i + 1;
    Set |V_l(i − 1)| = |V_l|;
end
```

IV. Simulation Results

Let us now provide some simulation results and a comparison between the proposed methodology and the standard trilateration algorithm to localize the nodes in a network. In the following we will consider networks where the nodes are

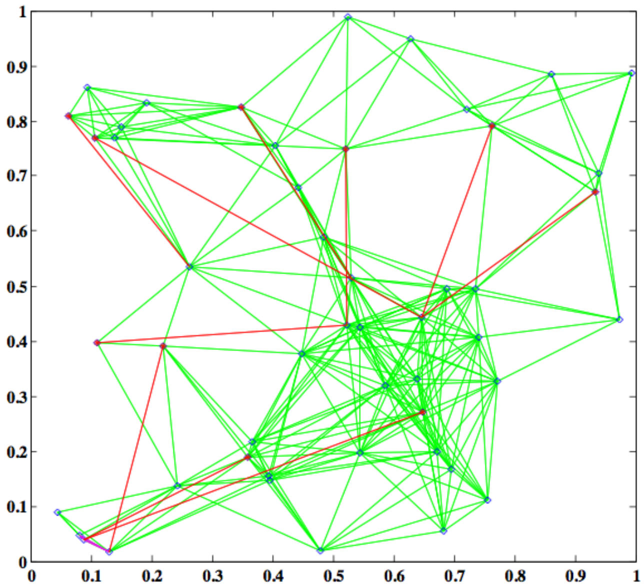

Fig. 2. Example of graph with 50 nodes. The blue nodes are localized according to trilateration after adding the shadow edges (red edges). The proposed algorithm localizes 50 nodes, the standard localization algorithm localizes 3 nodes, i.e., the anchors nodes.

embedded in the unit square $[0,1]^2 \subset \mathbb{R}^2$, i.e., for each node i, $x_i, y_i \in [0,1]$.

Let us first provide an example of execution of the procedure proposed in Algorithm 2. Figure 2 shows a graph with 50 nodes and 268 edges. During the execution of the procedure, blue nodes are localized using trilateration, while red nodes are localized by shadow edge trilateration. The proposed algorithm localizes all the nodes in the network exploiting three anchors and adding 10 shadow edges (i.e. less that the 4% of the edges). Using the same core anchors and applying only trilateration it is possible to localize only 5 nodes.

To prove the effecttiveness of the proposed approach, extensive trials have been carried out by considering random generates graphs. The networks are deployed over the unit square $[0,1]^2$, the radius of the sensing regions ρ ranges in $[0,1]$ and the size of the networks n ranges in $[0,1000]$. Each test is executed with fixed (ρ, d) and is averaged over 50 trials. The results are collected in figures 3, 4 and 5

Figure 3 shows the percentage of localized nodes obtained by Algorithm 2 with respect to total generated nodes, plotted against the distance radius ρ and the network size n. As shown by the Figure, and just as expected, for a fixed network size there is a threshold on the distance radius, and the nodes are almost all localized for valued of ρ above such a threshold. Note further that such a threshold value tends to decrease when the network size increases.

Figures 4 and 5 show, respectively, the difference and the ratio of the percentage of localized nodes obtained by Algorithm 2 and by the trilateration algorithm. Both figures highlight that the proposed algorithm is particularly effective for values of ρ below the network size-dependent threshold values identified in Figure 3. Specifically, note that the percentage of the proposed algorithm are always greater or equal than the trilateration algorithm; moreover there are up to 33.67 times more localized nodes (with $\rho = 700$ and $d = 0.07$ Algorithm 2 localizes 60.6% of the nodes, while TNC only 1.8%). Figure 6 shows that, the range of values of ρ and n where the proposed algorithm enhances the localization of nodes, is indeed the range where the shadow edges are created. Note that, in such a range, the amount of shadow edges is about $10-12\%$ of the total number of edges created.

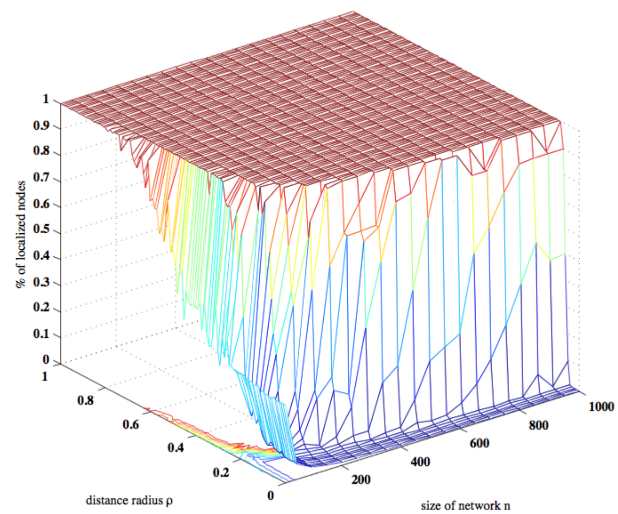

Fig. 3. Localization by shadow edges: percentage of localized nodes obtained by Algorithm 2 with respect to total generated nodes, plotted against the distance radius ρ and the network size n.

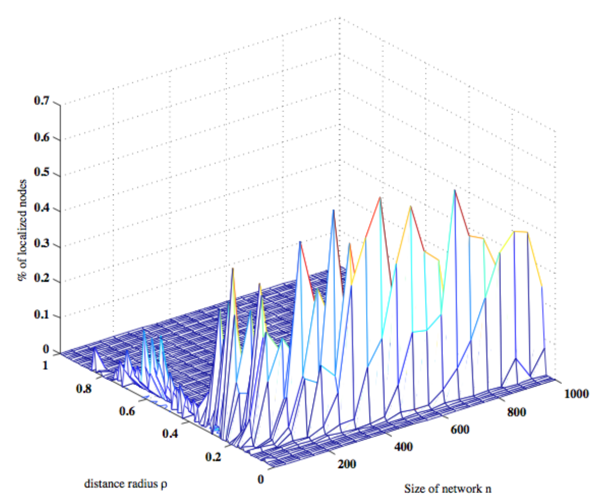

Fig. 4. Difference between the percentage of localized nodes with respect to total generated nodes, obtained by Algorithm 2 and by the trilateration algorithm, plotted against the distance radius ρ and the network size n.

V. Conclusions

In this paper a methodology for testing the localizability of a networks is proposed: the approach is less demanding than

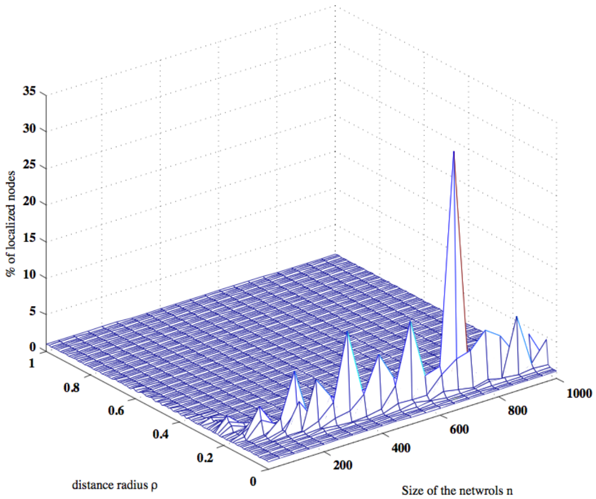

Fig. 5. Ratio between the percentage of localized nodes with respect to total generated nodes, obtained by Algorithm 2 and by the trilateration algorithm, plotted against the distance radius ρ and the network size n.

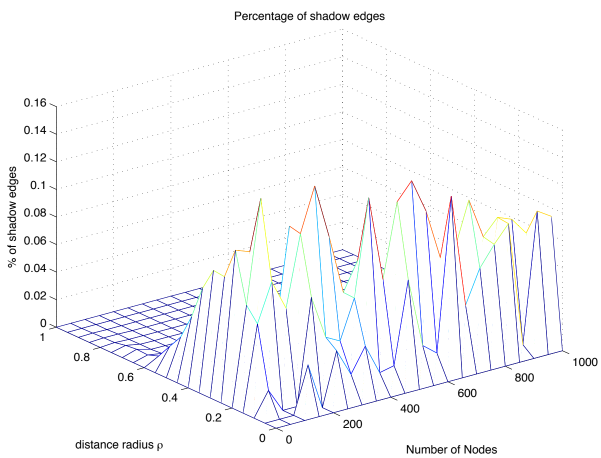

Fig. 6. Number of shadow edges created by the algorithm, plotted against the distance radius ρ and the network size d. Results are the average of 50 runs.

the trilateration algorithm. The idea is to use the information about the lack of connectivity between nodes in order to enhance the localization procedure, thus requiring (in 2D) each node to sense just 2 neighbors nodes, instead of the 3 required by the trilateration algorithm. The effectiveness of the proposed algorithm is compared with respect to the trilateration algorithm. The presented procedure is fully centralized, but author provide a decentralized version in [15], [16] according to the approach in [2]

Future work will be devoted to extend the methodology in order to provide a reasonably small set of localization options for the nodes, in the case where only a single real link is available, by considering multiple shadow edges. The possibility to localize, at least to a certain extent, a network made in such a way would allow agents to construct radio bridges (e.g., chains) for the ad-hoc communication in harsh environments (e.g., robotic swarms operating in tunnels or corridors during a fire blast). Moreover the proposed algorithm has to be tested under noisy condition, to benchmark the real effectiveness of the approach: here only measurements non corrupted by noise are considered, but in real scenario noise could heavily affect the measurements.

REFERENCES

[1] P. Biswas and Y. Ye, "Semidefinite programming for ad hoc wireless sensor network localization", In *Proc. of the 3rd International Symposium on Information Processing in Sensor Networks*, 2004.

[2] M. Carli, S. Panzieri, F. Pascucci, "A joint routing and localization algorithm for emergency scenario", *Ad Hoc Networks*, ISSN 1570-8705, 10.1016/j.adhoc.2012.09.001, 2012.

[3] 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 INFOCOM*, 2004.

[4] C. Gotsman and Y. Koren, "Distributed graph layout for sensor networks", In *Proc. of the International Symposium on Grpah Drawing*, pages September 2004.

[5] P. Gupta and P. Kumar, "Critical power for asymptotic connectivity in wireless networks", *Stochastic Analysis, Control, Optimization and Applications*, 1998.

[6] B. Hendrickson, "Conditions for unique graph realizations", *SIAM Journal of Computing*, vol. 21, pp. 65-84, 1992.

[7] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins,*"Global Positioning Systems: Theory and Practice"*, Springer, 5th edition, 2001.

[8] B. Jackson and T. Jordan, "Connected rigidity matroids and unique realizations of graphs", *Journal of Combinatorial Theory*, Series B, vol. 94, pp. 1-29, 2005.

[9] B. Jackson and T. Jordan, "A sufficient connectivity condition for generic rigidity in the plane", *Tech. Rep. Operations Research Department*, Eotvos University, Budapest, TR-2008-01, 2008.

[10] D. J. Jacobs and B. Hendrickson, "An algorithm for two-dimensional rigidity percolation: The pebble game", *Journal of Computational Physics*, vol. 137, pp. 346-365, 1997.

[11] G. Laman, "On graphs and rigidity of plane skeletal structures", *Journal of Engineering Mathematics*, vol. 4, pp. 331-340, 1970.

[12] X.-Y. Li, Y. Wang, P.-J. Wan, and C.-W. Yi, "Fault tolerant deployment and topology control for wireless ad hoc networks", in *Proc. of ACM MobiHoc*, 2003.

[13] L. Lovasz and Y. Yemini, "On generic rigidity in the plane", *SIAM Journal on Algebraic and Discrete Methods*, vol. 3, no. 1, pp. 91-98, 1982.

[14] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements", In *Proc. of the 2nd International Conference on Embedded Networked Sensor Systems*, 2004.

[15] G. Oliva, S. Panzieri, F. Pascucci, R. Setola, "Exploiting Routing Information in Wireless Sensor Networks Localization", *IEEE 2nd International Workshop on Network Science*, West Point, NY, USA, 2013.

[16] G. Oliva, S. Panzieri, F. Pascucci and R. Setola, "Simultaneous Localization and Routing in Sensor Networks using Shadow Edges", *2013 IFAC Intelligent Autonomous Vehicles Symposium* (IAV2013), Gold Coast, Australia, 26-28 June 2013.

[17] M. Penrose, "On K-connectivity for a geometric random graph", *Random Structures and Algorithms*, 1999.

[18] J. B. Saxe, "Embeddability of weighted graphs in k-space is strongly NP-hard", In *Proc. of the 17th Allerton Conference in Communications, Control and Computing*, 1979.

[19] Y. Shang, W. Ruml, Y. Zhang, and M. P. J. Fromherz, "Localization from mere connectivity", In *Proc. of ACM MobiHoc 03*, 2003.

[20] A. M.-C. So and Y. Ye, "Theory of semidefinite programming for sensor network localization". In *Proc. of ACM SIAM Symposium On Discrete Algorithms*, 2005.

[21] F. Xue and P. Kumar, "The number of neighbors needed for connectivity of wireless networks",*Wireless Networks*, vol. 10, pp. 169-181, 2004.

[22] L. Yunhao and Y. Zheng, *"Location, Localization and Localizability"*, Springer, 2011.