

AperTO - Archivio Istituzionale Open Access dell'Università di Torino

Dynamic Multidimensional Scaling with anchors and height constraints for indoor localization of mobile nodes

This is a pre print version of the following article:

Original Citation:

Availability:

This version is available <http://hdl.handle.net/2318/1685411> since 2018-12-30T17:31:36Z

Published version:

DOI:10.1016/j.robot.2018.06.015

Terms of use:

Open Access

Anyone can freely access the full text of works made available as "Open Access". Works made available under a Creative Commons license can be used according to the terms and conditions of said license. Use of all other works requires consent of the right holder (author or publisher) if not exempted from copyright protection by the applicable law.

(Article begins on next page)

Dynamic Multidimensional Scaling with anchors and heights constraints for indoor localization of mobile nodes

Carmelo Di Franco^{a,*}, Mauro Marinoni^a, Enrico Bini^b, Giorgio C. Buttazzo^a

^a*Retis Laboratory, Scuola Superiore Sant'Anna, Italy.*

^b*Department of Computer Science, University of Turin, Italy.*

Abstract

In distance-based localization, estimating the position of a network of wireless sensors is not an easy task. The problem increases when dealing with moving nodes and cluttered indoor environments. Many algorithms have been proposed in the literature and, among them, the Multidimensional Scaling (MDS) technique gained a lot of interest due to its resilience to flips ambiguities and easiness of use. Many variants of MDS have been proposed to overcome issues such as partial connectivity or distributed computation. In this context, it is common to place some anchors nodes to help in estimating the coordinates of the network correctly. However, instead of using the anchor's positions directly during the minimization of the MDS cost function, most approaches act on the estimated coordinates at the end of the MDS computation without fully utilizing the knowledge about anchors. In this work, the classic MDS and Dynamic MDS have been reformulated to utilize the anchor's position inside the minimization function. A set of real experiments in 3D with Ultrawide-band devices show that our approach considerably improves the accuracy of localization with respect to the usual MDS techniques.

Keywords: Multidimensional Scaling, Anchors, Distance-based Localization, Mobile nodes, Indoor localization

*Corresponding author

Email address: c.difranco@santannapisa.it (Carmelo Di Franco)

1. Introduction

In a wireless sensor network, estimating the positions of the nodes is of primary importance in many distributed systems. In indoor environments, where the GPS is not available, distance-based localization is typically used to derive the node coordinates by measuring inter-nodes distances. Node coordinates can be estimated from distance measurements using different techniques, such as trilateration, multilateration, and Multidimensional Scaling (MDS). MDS [1, 2] aims at visualizing a set of objects in an n -dimensional space. It takes as input a Dissimilarity Matrix that expresses how much two objects are dissimilar along one quality and finds a set of coordinates such that the distance between each couple of objects is proportional to the value of dissimilarity. MDS has been used for plotting sets of data in many application fields, such as economics and psychology. In the last decades, MDS has also been used for localization, where the objects are the nodes, and the dissimilarity matrix contains the inter-node distances.

Using anchor nodes in wireless sensor networks significantly improved the accuracy of localization [3]. However, in the MDS formulation the notion of anchor nodes is missing since this technique was designed for plotting generic objects with qualitative characteristics. Moreover, some limitations such as partial connectivity [4, 5] and different type [6] of noise are peculiar for network localization. For this reason, many variants of MDS have been proposed in the literature. To take anchors into account, some authors proposed solutions for incorporating the known coordinates into the MDS algorithm [7]. Such variants of MDS can be distinguished into distributed and centralized approaches. In distributed solutions the computation of the algorithm is shared among the components of the network.

In centralized approaches, a common solution is to apply a roto-translation transformation after the MDS computation and superimpose the estimated coordinates over the anchors [4, 5, 8, 9]. Biaz and Ji [7] used a different method consisting of updating the anchor's positions during the minimization proce-

dure. However, contrarily to the current MDS versions, the positions of the anchor nodes should not be modified during the minimization. For all these cases, the coordinates of the anchors slightly changes due to the minimization procedure, leading to a position error that reduces the overall accuracy of the system. Also, in the case where the anchor coordinates are modified at the end of the minimization, the output is not exact since the anchor's coordinates are not entirely used for finding the best estimation.

Figure 1 shows a trivial example that highlights the drawbacks described above. In this example, the network is composed of three anchors and two nodes that are localized with the classical MDS and the use of a roto-translation applied to the output.

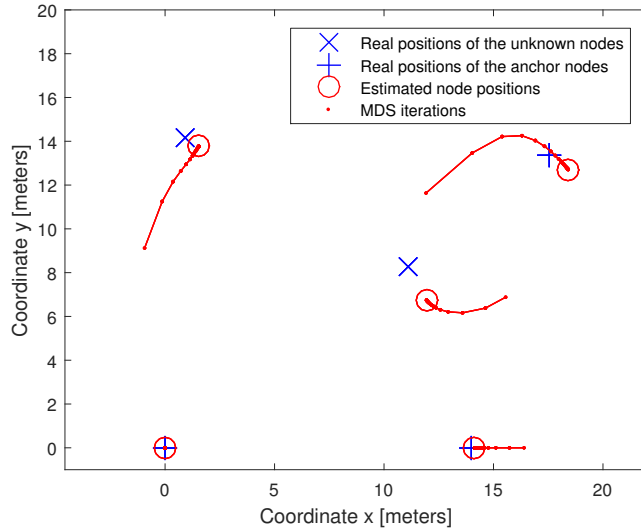


Figure 1: Estimated coordinates computed with the classical MDS. The red dots represent the coordinates during the algorithm iterations.

The figure exhibits that the classical MDS formulation does not benefit from the anchor information. Although the anchors' positions are known, their coordinates are subject to a not required minimization. Moreover, the algorithm introduces noise on the anchors' coordinates. As shown in Figure 1, the anchors' locations are not precisely reconstructed, even if such information is known a-

priori.

A distributed version of MDS, called distributed weighted Multidimensional Scaling (dwMDS), was proposed by Costa et al. [10]. They successfully included
50 the notion of anchors and were able to consider them in the MDS formulation since they split the computation on each node without modifying the anchor's coordinates. However, a distributed approach takes a lot of time to converge to a single agreed set of coordinates since all the network has to converge to a unique solution through wireless communication messages. For this reason, such
55 approach is usually suited for large static sensor networks with low computation capabilities. Instead, centralized approaches are usually preferred in small networks and in applications which consider node mobility, e.g., in the case of a small team of robots [11]. Another application case in which a centralized approach is commonly used includes indoor people tracking [3].

60 A previous work [12] proposes a theoretical generalization of the classical MDS algorithm, named MDS with Anchors (MDS-A), which uses the coordinates of some nodes (e.g., anchors) to improve the accuracy of the estimation. The approach presented in this paper extends the one proposed in [12] to the more general case of Dynamic Multidimensional Scaling (DMDS) - a technique
65 used for applications that include node mobility such as indoor people tracking - by proposing Dynamic MDS with Anchors (DMDS-A) that, similarly to MDS-A, incorporates the concept of anchors in the minimization. A set of real experiments employing Ultra-wide Band (UWB) devices has been performed to validate both MDS-A and DMDS-A. Moreover, is also proposed a modifi-
70 cation of MDS for 3D applications using the apriori knowledge of the height of the nodes. The development of such an approach has been encouraged by those applications in which nodes are attached at a fixed height (e.g., the belt of a walking person, shoes, a moving robot) and this knowledge can be used to improve the overall accuracy.

75 The rest of the paper is structured as follow: Section 2 provides an overview of the state of the art in MDS-based localization. Then, in Section 3 the addressed problem will be formalized. Section 4 will review the MDS and DMDS

mathematical formulations. In Section 5, MDS-A, DMDS-A, and the
MDS with heights constraints (MDS-Z) will be described. Experimental results
80 will be provided and discussed in Section 6. Finally, Section 7 will state the
conclusions.

2. Related work

Multidimensional Scaling (MDS) is a technique that represents a set of el-
ements in an r -dimensional space using the similarities/dissimilarities between
85 pairs of elements as distance information. There exist several variants of MDS
such as Classical MDS, Metric MDS, Non-Metric MDS, depending on the char-
acteristics of the distance information. The technique initially was meant for
visualizing a set of objects in a 2-D (or 3-D) space. Also, a particular formulation
for visualizing data with a correlation over time, called Dynamic Multidimen-
90 sional Scaling (DMDS) has been proposed by Ambrosi and Hansohm [13] in
1987. In the last decade, it has been extensively used in distance-based local-
ization for its elegant formulation, resilience to flip ambiguities, and easiness of
use. However, in order to be used in practical scenarios, many variants have
been proposed to overcome issues such as partial connectivity and node mobil-
95 ity. These techniques usually do not modify the MDS formulation but change
the algorithm input (distance measurements) or adjust its output.

Many variants have been introduced in literature, and some of them used
misleading acronyms that can be confused with the original approach. For exam-
ple, Garimella, in his master thesis [14], proposed a MDS variation that involves
100 the localization of “virtual” nodes apart from the original nodes. He named the
proposed approach with the same name of the original DMDS. Cabero et al. [3]
proposed a variant of MDS and named their approach dynamic weighted MDS
(dwMDS), which is the same acronym of the well known distributed weighted
MDS (dwMDS) proposed by Costa [10]. For the sake of clarity, in the rest of
105 the paper, the name DMDS refers to the original approach proposed by Am-
brosini and Hansohm [13]. All the other techniques will be defined as variants

or modifications of MDS.

In Xu et al. [15], the authors use DMDS for visualizing the temporal evolution of dynamic networks. Beck and Baxley [16] proposed to use DMDS as
110 a methodology to track nodes over time by exploiting odometry information. Among the other approaches that use MDS for tracking mobile nodes, Cabero et al. [3] proposed an extension of MDS that finds the embedding of people's trajectories by using a set of connectivity matrices through time. For this reason, they put particular attention in understanding how their approach behaves with
115 respect to varying connectivity. Moreover, they make an extensive use of anchors conveniently fixed at the borders of a rectangle area to track the dynamic of peoples inside the hull. In Jamâa et al. [17], the authors proposed a modified version of the classical MDS approach, i.e., coordinate are computed with eigenvalues decompositions, to introduce a cooperative mobile network tracking
120 algorithm. They changed the matrices containing the orthonormal eigenvectors and proposed two different algorithms also considering partial connectivity. In the particular context of mobile robot localization without anchors, Oliveira and Almeida [18] proposed a technique that gives confidence values to position estimates obtained by MDS at successive instants. In [19] the authors proposed
125 a modification to MDS, which includes the notion of nodes velocities to solve the ambiguities generated by two consecutive MDS outputs. However, the approach doesn't use the standard SMACOF minimization being more computationally expensive.

The presented works from literature do not try to exploit the knowledge of
130 anchor positions in the minimization but apply some filtering techniques to the MDS output. Hence, most of such extensions suffer from two drawbacks: the MDS minimization does not benefit from anchors information, and the algorithm introduces noise on the coordinates.

The different approaches that have been used to fix the problem described
135 above can be divided into two main classes: those applying a roto-translation to the MDS coordinates, and those operating in the minimization process by adding constraints or modifying the cost function. In the remaining of the

section is explained why both approaches do not solve the problem efficiently. This work refers only to centralized approaches; a distributed solution has been
140 proposed by Costa et al. [10].

2.1. Applying a roto-translation transformation

The first class of works separates the localization algorithm in two step: first, the relative coordinates are computed through MDS; then, a roto-translation transformation is applied at the end of the minimization to align the estimated
145 coordinates of the anchors to their real positions.

For example, Ji and Zha [8] align the relative positions of the nodes to their real ones by computing a shift, rotation, and reflection of the coordinates. They calculate the rotation and translation matrices by the aid of at least three nodes. Similarly, in the MDS-MAP algorithm [4], the authors transform the global map
150 to an absolute map based on the absolute positions of the anchors. They state that the complexity of this step is $O(m^3 + n)$ for m anchors.

Latsoudas and Sidiropoulos [20] avoid the problem of aligning the relative positions of the nodes to the real location of the anchors by carefully placing the anchors to form an orthogonal triangle, and the orthogonal sides of this triangle
155 are chosen as coordinate basis vectors. Then, all the projections are computed directly onto the native coordinate basis, thus, avoiding the need of applying a roto-translation. This approach has the drawback that the anchor nodes need to be placed in the environment according to a specific pattern.

Cheung and So [21] proposed an approach that employs the anchor coordinates to compute the rotation matrix. Their solution is initially similar to the
160 one proposed in this paper. However, their MDS formulation is specific to the case in which only one node has to be located through the use of m anchors.

2.2. Modifying the MDS minimization or adding constraints

Biaz and Ji [7] described a way of updating the anchor's positions during the minimization procedure. They first run the MDS step and then update the
165 anchors' location, after a check of communication constraints. Note that this

procedure is only a workaround since all nodes are treated as unknown, and the MDS minimization is also performed on the anchors. After the minimization, the position of the anchors is updated, but this also requires to adjust the distances between the anchors and the nodes to maintain consistency.

Another technique that exists in the literature on MDS consists in applying generic restrictions to the variables [22]. In particular, information regarding the anchors' locations can be included as a constraint of MDS. However, such an approach increases the space of the variables and leads to higher complexity and computational cost. Conversely, reducing the space of the variables to only the unknown nodes and consider the anchors as a constant value, not only improves the precision but also reduces the complexity of the algorithm. Such an intuition is behind the approach proposed in this paper. With respect to the related work, the proposed approach provides a mathematical formulation to estimate only the nodes coordinates. Since the anchor's coordinates are used to estimate the node positions, the two presented drawbacks do not subsist, leading to a higher accuracy in the localization.

3. Problem formulation

Consider a collection of $N = n + m$ nodes with position $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N]^T \in \mathbb{R}^{N \times r}$ where $r = 2, 3$ (2D or 3D). The first n nodes have unknown coordinates while the last m are anchors whose position is known a-priori. Each node i can estimate the distance $\hat{d}_{ij}$ from any other node j in the range of communication, and the estimated distance is affected by noise. Also, $d_{ij}(\mathbf{X}) = \|\mathbf{x}_i - \mathbf{x}_j\|$ is the Euclidean distance between node i and j .

Each node continuously estimates all the inter-node distances and exchanges this information with the other nodes on the network. At time t_k the network performs the k -th step, and each node receives all the distance pairs and stores

them in a Matrix, called Distance Matrix, that is defined as following:

$$\mathbf{D} = \begin{bmatrix} 0 & \hat{d}_{12} & \cdots & \hat{d}_{1N} \\ \hat{d}_{21} & 0 & \cdots & \hat{d}_{2N} \\ \vdots & \cdots & \ddots & \vdots \\ \hat{d}_{N1} & \cdots & \cdots & 0 \end{bmatrix}$$

190 During the interval Δt between two consecutive iterations, each node may have traveled a certain distance $\widehat{\Delta d_i} \in \mathbb{R}^+$. Nodes will measure their traveled distance through dead reckoning techniques (e.g., pedestrian step detection, odometry) and will exchange this information to the other nodes together with collected distances.

195 **Problem** (distance based-localization with mobile and anchor nodes). *At each time step k , given*

- (a) *range measurements $\hat{d}_{ij}(k)$, $| i, j \in \{1, \dots, N\}$,*
- (b) *odometry of the mobile nodes $\widehat{\Delta d_i}(k)$, $| i \in \{1, \dots, n\}$,*
- (c) *and anchor positions $\mathbf{x}_i$, $| i \in \{n+1, \dots, N\}$,*

200 *or a subset of them, determine the coordinates of the unknown nodes $\mathbf{X}(k)$.*

4. Multidimensional Scaling overview

Given a network composed of N nodes in a r -dimensional space, whose coordinates $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N]^T \in \mathbb{R}^{N \times r}$ are unknown, the algorithm recovers the coordinates of the elements by minimizing the mismatch of the following function

$$\min_{\mathbf{X}} \mathcal{S}(\mathbf{X}) = \min_{\mathbf{X}} \sum_{i < j \leq N} w_{ij} \left(\hat{d}_{ij} - d_{ij}(\mathbf{X}) \right)^2 \quad (1)$$

where w_{ij} is a weight defining the quality of the measurement $\hat{d}_{ij}$.

The objective function $\mathcal{S}$, also called stress-function, can be minimized in different ways, such as using the steepest descent approach. De Leeuw [23]
 205 proposed an iterative method that at each step minimizes a simple convex function which majorizes the complex function. This approach is called “Scaling by

MAjorizing a COmplicated Function” (SMACOF) and it was proved to perform significantly better with respect to other approaches [23], in terms of guarantees and rate of convergence.

210 The non linear least squares problem in Equation (1) is solved minimizing iteratively a convex function $\mathcal{T}(\mathbf{X}, \mathbf{Z}) \geq \mathcal{S}(\mathbf{X})$. $\mathcal{T}$ bounds $\mathcal{S}$ from above and touches the surface of $\mathcal{S}$ at point $\mathbf{Z}$. The iterative procedure is summarized in Algorithm 1.

Algorithm 1 Scaling by MAjorizing a COmplicated Function (SMACOF)

input: initial position estimate $\mathbf{X}^{(0)}$

stress function $\mathcal{S}$

majorizing function $\mathcal{T}$

repeat

$\mathbf{Z} = \mathbf{X}^{k-1}$

$\mathbf{X}^k \leftarrow \min_{\mathbf{X}} \mathcal{T}(\mathbf{X}, \mathbf{Z})$

until $\mathcal{S}(\mathbf{X}^{k-1}) - \mathcal{S}(\mathbf{X}^k) < \epsilon$

4.1. The majorization function

The stress function $\mathcal{S}$ can be expanded as follows:

$$\begin{aligned} \mathcal{S}(\mathbf{X}) &= \sum_{i < j \leq N} w_{ij} \left(\hat{d}_{ij} - d_{ij}(\mathbf{X}) \right)^2 = \\ &= \sum_{i < j} w_{ij} \hat{d}_{ij}^2 + \sum_{i < j} w_{ij} d_{ij}^2(\mathbf{X}) - 2 \sum_{i < j} w_{ij} \hat{d}_{ij} d_{ij}(\mathbf{X}). \end{aligned}$$

Note that the first term is a constant and the second term is quadratic in $\mathbf{X}$ and therefore relatively easily solved. The third term is bounded using the Cauchy-Schwarz inequality using the fact that:

$$\begin{aligned} d_{ij}(\mathbf{X}) &= \|\mathbf{x}_i - \mathbf{x}_j\| = \|\mathbf{x}_i - \mathbf{x}_j\| \frac{\|\mathbf{z}_i - \mathbf{z}_j\|}{\|\mathbf{z}_i - \mathbf{z}_j\|} \geq \\ &= \frac{(\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{z}_i - \mathbf{z}_j)}{\|\mathbf{z}_i - \mathbf{z}_j\|} \end{aligned} \tag{2}$$

215 where $\mathbf{Z} = [\mathbf{z}_1, \dots, \mathbf{z}_N]^T \in \mathbb{R}^{N \times r}$. Hence, the third term can be bounded by:

$$\sum_{i < j} w_{ij} \hat{d}_{ij} d_{ij}(\mathbf{X}) \geq \sum_{i < j} w_{ij} \hat{d}_{ij} \frac{(\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{z}_i - \mathbf{z}_j)}{\|\mathbf{z}_i - \mathbf{z}_j\|} \quad (3)$$

Thus, there is a simple quadratic function $\mathcal{T}(\mathbf{X}, \mathbf{Z})$ that majorizes the stress:

$$\begin{aligned} \mathcal{S}(\mathbf{X}) \leq \mathcal{T}(\mathbf{X}, \mathbf{Z}) &= \sum_{i < j} w_{ij} \hat{d}_{ij}^2 + \sum_{i < j} w_{ij} d_{ij}^2(\mathbf{X}) \\ &\quad - 2 \sum_{i < j} w_{ij} \hat{d}_{ij}^2 \frac{(\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{z}_i - \mathbf{z}_j)}{\|\mathbf{z}_i - \mathbf{z}_j\|} \end{aligned} \quad (4)$$

$\mathcal{T}(\mathbf{X}, \mathbf{Z})$ can be written in matrix form:

$$\mathcal{T}(\mathbf{X}, \mathbf{Z}) = \mathbf{C} + \text{tr}(\mathbf{X}^T \mathbf{V} \mathbf{X}) - 2 \text{tr}(\mathbf{X}^T \mathbf{B}(\mathbf{Z}) \mathbf{Z}) \quad (5)$$

where $\mathbf{V}$ and $\mathbf{B}(\mathbf{Z})$ are two matrices whose elements are defined as follow:

$$v_{ij} = \begin{cases} \sum_{k=1, k \neq j}^N w_{kj} & \text{if } i \neq j, \\ \sum_{k=1, k \neq j}^N v_{kj} & \text{if } i = j. \end{cases}$$

$$b_{ij} = \begin{cases} \sum_{k=1, k \neq j}^N w_{kj} \frac{\hat{d}_{ij}}{d_{ij}(\mathbf{Z})} & \text{if } i \neq j, \\ \sum_{k=1, k \neq j}^N b_{kj} & \text{if } i = j. \end{cases}$$

Thus, the minimum of the function can be computed as:

$$\mathbf{X} = \min_{\mathbf{X}} \mathcal{T}(\mathbf{X}, \mathbf{Z}) = \mathbf{V}^{-1} \mathbf{B}(\mathbf{Z}) \mathbf{Z} \quad (6)$$

4.2. Dynamic Multidimensional Scaling

DMDS is a particular formulation of MDS that consider the problem of estimating the positions not only for the current k -th step but also the p instants before it. In order to maintain invariant the MDS formulation the coordinates

220

vector is redefined as $\mathbf{X} = [\mathbf{X}^{k-p}, \dots, \mathbf{X}^k]$ variables to find having as an input a “super-dissimilarity” matrix that contains the distances measurements as follow:

$$D = \begin{bmatrix} \mathbf{D}^{k-p} & \dots & \mathbf{R}^{k-p,k} \\ \vdots & \ddots & \vdots \\ \mathbf{R}^{k,k-p} & \dots & \mathbf{D}^k \end{bmatrix}$$

where $\mathbf{R}^{k-p,k}$ contains the information of the displacement between node i at the $(k-p)$ -th step and node j at the k -th step. Note that, in the context of localization, the only measurable information is the distance $d_{ii}^{k-p,k}$ that is, in practice, the odometry of the node. Hence, $\mathbf{R}^{k-p,k}$ will be defined as follow:

$$\mathbf{R}^{k-p,k} = \begin{bmatrix} \widehat{d_{11}}^{k-p,k} & 0 & \dots & 0 \\ 0 & \widehat{d_{22}}^{k-p,k} & \dots & 0 \\ \dots & \dots & \ddots & 0 \\ 0 & \dots & \dots & \widehat{d_{nn}}^{k-p,k} \end{bmatrix}.$$

Note that, in order to discard the zeros introduced in the matrices, the corresponding weights $w_{ij}^{k-p,k}$ must be set to zero.

225 The choice of the value of p varies according to the context of the applications. For offline trajectory reconstruction, a high number of p steps ($p = 10$) should improve the accuracy, but it inevitably increases the computational complexity since it is linear with p . For online tracking, p should be set as a low values such as $p = 2$, or $p = 3$, as suggested in [16].

230 5. Proposed approach

This section describes the proposed extension to the MDS algorithm to include information on the anchors. First, is presented the improvement for the classic MDS, then is described the extension for the DMDS algorithm.

5.1. MDS with anchors constraints (MDS-A)

235 Let us consider a set of $N = n + m$ nodes where the first n nodes have unknown positions and the last m are the anchor nodes. Referring to the matrices in Equation (6), $\mathbf{X}$ and $\mathbf{Z}$ can be partitioned as follows:

$$\begin{aligned}
\mathbf{X} &= \begin{bmatrix} \mathbf{X}_u \\ \mathbf{X}_a \end{bmatrix}, \quad \mathbf{Z} = \begin{bmatrix} \mathbf{Z}_u \\ \mathbf{Z}_a \end{bmatrix}, \quad \text{with} \\
\mathbf{X}_u &= [\mathbf{x}_1, \dots, \mathbf{x}_n]^T \in \mathbb{R}^{n \times r} \\
\mathbf{X}_a &= [\mathbf{x}_{n+1}, \dots, \mathbf{x}_{n+m}]^T \in \mathbb{R}^{m \times r} \\
\mathbf{Z}_u &= [\mathbf{z}_1, \dots, \mathbf{z}_n]^T \in \mathbb{R}^{n \times r} \\
\mathbf{Z}_a &= [\mathbf{z}_{n+1}, \dots, \mathbf{z}_{n+m}]^T \in \mathbb{R}^{m \times r}.
\end{aligned}$$

Similarly, $\mathbf{V}$ and $\mathbf{B}(\mathbf{Z})$ can be partitioned in blocks as follow:

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_{11} & \mathbf{V}_{12} \\ \mathbf{V}_{12}^T & \mathbf{V}_{22} \end{bmatrix}, \quad \mathbf{B}(\mathbf{Z}) = \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} \\ \mathbf{B}_{12}^T & \mathbf{B}_{22} \end{bmatrix}, \quad (7)$$

where matrices $\mathbf{V}_{11}$ and $\mathbf{B}_{11}$ are of size $n \times n$, $\mathbf{V}_{12}, \mathbf{B}_{12}$ are $n \times m$, and $\mathbf{V}_{22}, \mathbf{B}_{22}$ are $m \times m$.

240 The following theorem provides a way for computing $\mathbf{X}_u$ as a function of the anchors coordinates $\mathbf{X}_a$.

Theorem 1. *Given the stress function $\mathcal{T}(\mathbf{X}, \mathbf{Z})$ of Equation (5), if is known the exact value of the subset $\mathbf{X}_a \in \mathbb{R}^{m \times r}$ of $\mathbf{X}$, it is possible to compute the remaining unknown values $\mathbf{X}_u$ as a function of $\mathbf{X}_a$:*

$$\mathbf{X}_u = \mathbf{V}_{11}^{-1} (\mathbf{B}_{11} \mathbf{Z}_u + \mathbf{B}_{12} \mathbf{Z}_a - \mathbf{V}_{12} \mathbf{X}_a), \quad (8)$$

with $\mathbf{V}_{11}$, $\mathbf{V}_{12}$, $\mathbf{B}_{11}$, and $\mathbf{B}_{12}$ matrix block defined in (7).

Proof. See Appendix A. □

245 The major practical benefit deriving from Theorem 1 is the possibility to extend any variant of MDS that uses the SMACOF implementation to support the notion of anchors by merely modifying the computation of $\mathbf{X}$. Note that, Equation (8) reduces to Equation (6) in the absence of anchors (i.e., $m = 0$). Hence, the approach is not limited to the a-priori knowledge on the anchors, but can straightforwardly be used also when their number varies.

By applying such an approach to the motivational example showed in Figure 1, the node coordinates can be estimated exploiting the knowledge of the anchor nodes, as shown in Figure 2. Moreover, as it can be seen, only the unknown coordinates are computed during the minimization while the anchor nodes are at their fixed position from the beginning.

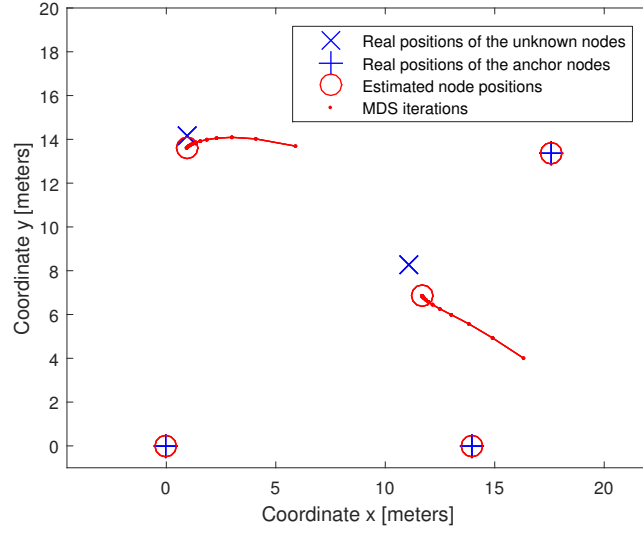


Figure 2: Estimated coordinates computed with the proposed formulation of MDS with anchors. Only the unknown node coordinates are estimated leading to a reduction of the overall error.

255 5.2. DMDS with anchors constraints (DMS-A)

This section extends the formulation of MDS with anchors to the more general DMDS technique. Similarly to the previous approach, it splits the matrices $\mathbf{X}$, $\mathbf{Z}$, $\mathbf{V}$, and $\mathbf{B}(\mathbf{Z})$ as follow:

$$\mathbf{X} = [\mathbf{X}_u^{k-p}, \mathbf{X}_a, \dots, \mathbf{X}_u^k, \mathbf{X}_a]^T, \quad \mathbf{Z} = [\mathbf{Z}_u^{k-p}, \mathbf{Z}_a, \dots, \mathbf{Z}_u^k, \mathbf{Z}_a]^T$$

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_{11} & \mathbf{V}_{12} & \mathbf{V}_{13} & \mathbf{V}_{14} & \cdots & \cdots \\ \mathbf{V}_{21} & \mathbf{V}_{22} & \mathbf{V}_{23} & \mathbf{V}_{24} & \cdots & \cdots \\ \hline \mathbf{V}_{31} & \mathbf{V}_{32} & \mathbf{V}_{33} & \mathbf{V}_{34} & \cdots & \cdots \\ \mathbf{V}_{41} & \mathbf{V}_{42} & \mathbf{V}_{43} & \mathbf{V}_{44} & \cdots & \cdots \\ \hline \vdots & \vdots & \vdots & \vdots & \ddots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}, \mathbf{B}(\mathbf{Z}) = \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} & \mathbf{B}_{13} & \mathbf{B}_{14} & \cdots & \cdots \\ \mathbf{B}_{21} & \mathbf{B}_{22} & \mathbf{B}_{23} & \mathbf{B}_{24} & \cdots & \cdots \\ \hline \mathbf{B}_{31} & \mathbf{B}_{32} & \mathbf{B}_{33} & \mathbf{B}_{34} & \cdots & \cdots \\ \mathbf{B}_{41} & \mathbf{B}_{42} & \mathbf{B}_{43} & \mathbf{B}_{44} & \cdots & \cdots \\ \hline \vdots & \vdots & \vdots & \vdots & \ddots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad (9)$$

where the dimension of the matrix blocks are defined as in (7). The problem can hence be extended to the general case. The following corollary provides a way for computing $\mathbf{X}_u^k$ at different time steps k as a function of the anchors coordinates $\mathbf{X}_a$.

Corollary 1 (of Theorem 1). *Given the stress function $\mathcal{T}(\mathbf{X}, \mathbf{Z})$ of Equation (5), if the exact values of the subset $\mathbf{X}_a \in \mathbb{R}^{m \times r}$ of $\mathbf{X}$ are known, it is possible to compute the remaining unknown values $\mathbf{X}_u^k$ at different time steps k as a function of $\mathbf{X}_a$:*

$$\mathbf{X}_u^k = \mathbf{V}_{k,k}^{-1} \left(\sum_{j=1}^p \mathbf{B}_{i,j} \mathbf{Z}_u^j + \sum_{j=1}^p (\mathbf{B}_{i,2j} - \mathbf{V}_{i,2j}) \mathbf{Z}_a - \sum_{j \neq i}^p \mathbf{V}_{i,j} \mathbf{X}_u^j \right), \text{ with } i = 2k-1 \quad (10)$$

with $\mathbf{V}_{ij}$ and $\mathbf{B}_{ij}$ matrix blocks defined in (9).

5.3. 3D localization with known heights

In indoor localization, anchor nodes are positioned in a 3D space, and they are usually placed in order to increase the coverage and accuracy of the localization. Mobile nodes can be either robotic units or tracked persons. In both cases, the height of the range sensor may be known. For example, the sensor may be attached to the belt of a person or at a specific position of the robot. In these cases, when localization is in 3D, the error in the resulting computed positions will also affect the height of the nodes that in this case are known. Hence, such

information can be used to force the algorithm to provide the correct height and reduce the final error. The following section proposes a modification of the MDS algorithm that can be used when one (or more) partial coordinates are known.

Let's consider a set of nodes $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N]^T \in \mathbb{R}^{N \times r}$ with $\mathbf{x} = [x \ y \ z]$ whose 3th component z is known. It is possible to write $\mathbf{X}$ as $\mathbf{X} = [\mathbf{X}_{xy} \ \mathbf{X}_z]$ and rewrite the trace in Eq. (5) as follows:

$$\begin{aligned} \text{tr}(\mathbf{X}^T \mathbf{V} \mathbf{X}) &= \text{tr} \left(\begin{bmatrix} \mathbf{X}_{xy} \\ \mathbf{X}_z \end{bmatrix}^T \mathbf{V} \begin{bmatrix} \mathbf{X}_{xy} & \mathbf{X}_z \end{bmatrix} \right) = \\ &= \text{tr} \left(\begin{bmatrix} \mathbf{X}_{xy}^T \mathbf{V} \mathbf{X}_{xy} & \mathbf{X}_{xy}^T \mathbf{V} \mathbf{X}_z \\ \mathbf{X}_z^T \mathbf{V} \mathbf{X}_{xy} & \mathbf{X}_z^T \mathbf{V} \mathbf{X}_z \end{bmatrix} \right) = \\ &= \text{tr}(\mathbf{X}_{xy}^T \mathbf{V} \mathbf{X}_{xy}) + \text{tr}(\mathbf{X}_z^T \mathbf{V} \mathbf{X}_z). \end{aligned}$$

Following the same decomposition on the second term, Eq. (5) can be rewritten as:

$$\mathcal{T}(\mathbf{X}_{xy}, \mathbf{Z}) = \mathbf{C}' + \text{tr}(\mathbf{X}_{xy}^T \mathbf{V} \mathbf{X}_{xy}) - 2 \text{tr}(\mathbf{X}_{xy}^T \mathbf{B}(\mathbf{Z}) \mathbf{Z}_{xy}) \quad (11)$$

where $\mathbf{C}' = \mathbf{C} + \text{tr}(\mathbf{X}_z^T \mathbf{V} \mathbf{X}_z) - 2 \text{tr}(\mathbf{X}_z^T \mathbf{B}(\mathbf{Z}) \mathbf{Z}_z)$ is constant.

The modification can be employed on any variant of MDS since it is directly applied to the SMACOF approach. The solutions obtained extending the previously proposed approaches (i.e., MDS-A and DMDS-A) are called MDS with Anchors and heights constraints (MDS-ZA) and Dynamic MDS with Anchors and heights constraints (DMDS-ZA), respectively.

6. Experimental evaluation

The two proposed formulations, MDS-A and DMDS-A, have been evaluated in comparison with the generic MDS and DMDS approaches that apply a roto-translation transformation to the coordinates estimated by the minimization.

It is worth noting that the different techniques presented in the related work mainly differ in the way they compute the roto-translation matrix. Even if different methods lead to different accuracy, the two drawbacks highlighted in Section 2 are common to all those approaches despite the specific algorithm
 295 used to align the coordinates.

To evaluate the goodness of the approaches with respect to the real coordinates the Root Mean Square Error (RMSE) between actual and estimated node positions is used as a metric, defined as:

$$\text{RMSE}(\mathbf{X}, \hat{\mathbf{X}}) := \sqrt{\frac{1}{n} \sum_{i=1}^n \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2}. \quad (12)$$

Note that, the computation of the RMSE considers only mobile nodes and explicitly excludes the anchors, even if anchors' coordinates computed with standard approaches such as roto-translation are affected by an error.

The rest of the section will first describe the hardware used and the experiment setup. Then, results for the proposed approaches (MDS-A, DMDS-A,
 300 MDS-ZA, and DMDS-A) will be provided.

6.1. Experimental platform and setup

The experiments are performed using a set of commercial devices developed from Pozyx Labs[24]. The Pozyx system is a hardware solution that provides
 305 accurate positioning and motion information, and its development kit is composed of 5 mobile devices, named tags, and 4 anchors. Each node (anchor or tag) relies on a UWB transceiver for accurate distance estimation. The accuracy is in the order of centimeters when two devices are in line of sight, or up to 2 meters with the presence of obstacles, walls or electronic interference. Tags also
 310 have a 9-axis IMU sensor that can be used to acquire motion information and can work both attached to an Arduino board or remotely triggered by other tags. In the first option, a tag can be handled through an open-source Arduino library. Figure 3a shows the Pozyx tag used as a shield for an Arduino Uno.

The 4 anchors are placed at the borders of a room with size 20×15 meters.
 315 Three tags were located at a fixed static known position and can act as anchors

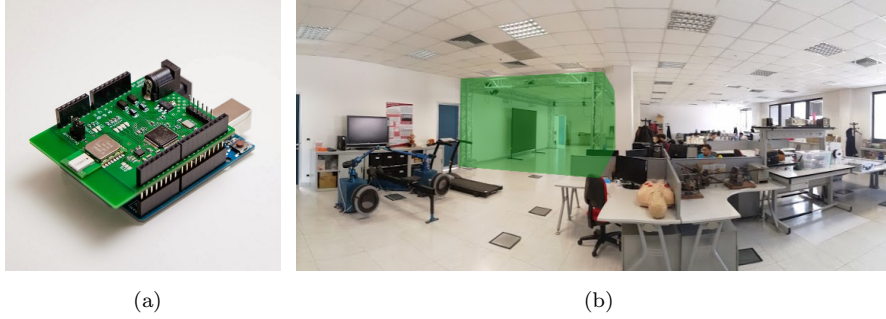


Figure 3: a) Pozyx device on top of an Arduino Board. b) Laboratory in which the real experiment has been performed. In green the Area of coverage of the the Vicon system.

(i.e., the 3D coordinates are known) or tags depending on the specific experiment. The remaining two tags were attached to the belt of two walking persons. In order to provide the ground truth for the evaluation, a Vicon System able of tracking a portion of the room of 6×4 meters has been used. Figure 3b shows the laboratory highlighting (in green) the area tracked by the Vicon system.

6.2. Experimental results

The results presented in this section are obtained from one of the experiments in which two people walking inside the area covered by the Vicon System were tracked. All the following results are computed by using the log collected in this experiment by varying the anchors setup or changing the MDS algorithm.

To better explain the trial, Figure 4 presents the snapshot of the entire experiment, showing the setup together with the ground truth and the estimation obtained with one of the approaches (i.e., DMDS-ZA) when all the seven static nodes act as anchors. The black line represents the ground truth of the first person walking along one direction back and forth, while the red line shows the ground truth of the second one performing circles around the area. The blue and magenta dots shows the coordinates estimated by DMDS-ZA of the first and second person, respectively. The black dots are the tag nodes, in this case, all acting as anchors (known 3D location).

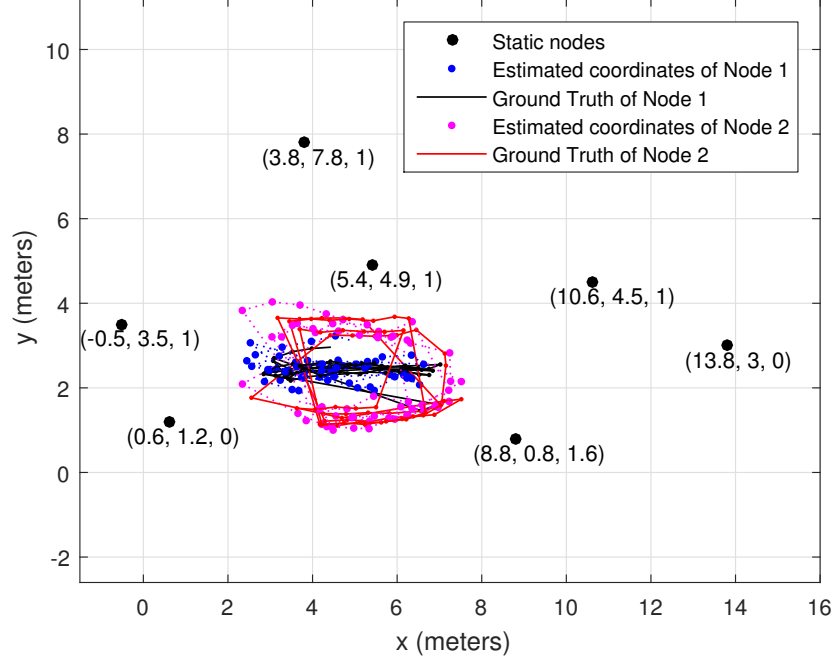


Figure 4: Snapshot of the trial. The black and red lines represent the ground truth of two walking persons and the blue and magenta dots their estimated coordinates with the proposed approach. The black dots are the tag nodes that act as anchors.

335 6.2.1. *MDS vs MDS-A*

The first experiment presents the advantages in term of accuracy obtained through the inclusion of the knowledge about anchors, and in particular the effects of the number of them. The network has $N = 9$ nodes and the number of anchors m varied from 3 to 7, while the two moving nodes acquire distance
340 information with the enabled nodes and compute their position.

The first experiment presents the advantages in term of accuracy obtained through the inclusion of the knowledge about anchors, and in particular the effects of the number of them. The network has $N = 9$ nodes and the number of anchors m varied from 3 to 7, while the two moving nodes acquire distance
345 information with the enabled nodes and compute their position. Results are presented in Figure 5, which shows the Cumulative Distribution Function (CDF)

of the RMSE error of MDS and MDS-A for $m = 3$ and $m = 7$, while the values of the CDF are showed in Table 1).

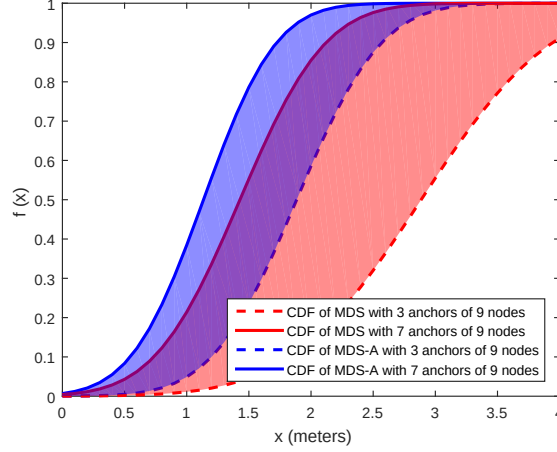


Figure 5: CDF of the RMSE error of MDS and MDS-A for $m = 3$ and $m = 7$.

As expected, MDS-A (blue lines) has a lower mean error than MDS (red lines). The benefits of the anchors are more relevant when $m = 3$ (dashed lines), but MDS has significantly higher mean error even with $m = 7$ (solid lines).

Table 1: Mean error and standard deviation of the CDFs showed in Figure 5

Approach	μ	σ
MDS with $m = 3$	2.8860	0.8258
MDS-A with $m = 3$	1.8847	0.5345
MDS with $m = 7$	1.4282	0.5406
MDS-A with $m = 7$	1.1350	0.4596

Among the reasons behind the quite high mean error, there is the presence of possible flips around the Z-plane (i.e., some nodes may have a negative z-coordinate). If the nodes heights are known, such a problem can be eliminated applying the MDS-Z extension proposed in Section 5.3.

6.2.2. MDS vs MDS-Z

In the second experiment, the same scenario of the previous experiment was repeated, but the nodes coordinates have been computed taking into account with MDS-Z and MDS-ZA, which take into account the knowledge about nodes height. The elevation of the two nodes attached to the belt of the walking persons are subjected to a variation of few centimeters and hence can be considered as fixed and known apriori. Results are shown in Figure 6, while the values of the CDF are showed in Table 2.

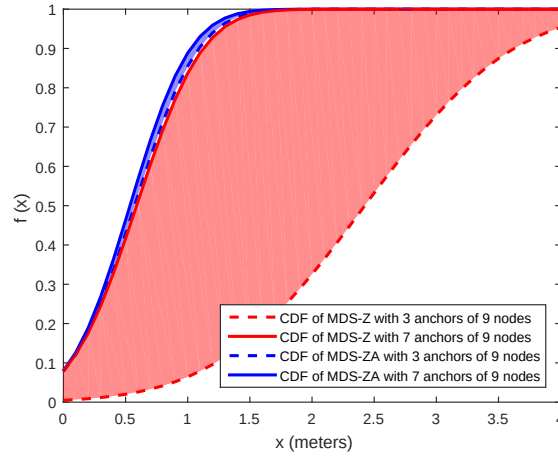


Figure 6: CDF of the RMSE error of MDS-Z and MDS-ZA for $m = 3$ and $m = 7$.

Figure 6 clearly highlights the benefits of the knowledge of the height of the nodes. MDS-ZA (blue lines) presents similar performance with $m = 3$ (dashed line) and with $m = 7$ (solid lines). Moreover, even without anchors constraints, MDS - Z (red lines) shows great improvements with a number of anchors $m = 7$. This effect is due to the z^{th} constraints forcing the minimization to align the nodes over a specific plane Z-plane. Then the roto-translation shifts the coordinates to the correct position nullifying the flips on the z-plane. Note that, even if this result is significant, MDS-Z with $m = 7$ has a CDF that is lower than MDS-ZA with $m = 3$, proving the necessity of using the anchor's constraints.

Table 2: Mean error and standard deviation of the CDFs showed in Figure 6

Approach	μ	σ
MDS-Z with $m = 3$	2.0539	0.8193
MDS-Z with $m = 3$	0.5882	0.4152
MDS-ZA with $m = 3$	0.5705	0.4010
MDS-ZA with $m = 7$	0.5358	0.3780

Figure 7 shows the norm of the error on the z-coordinate of the mobile nodes computed with MDS-A (red line) and MDS-ZA (blue line), when the number of anchors m is equal to 7.

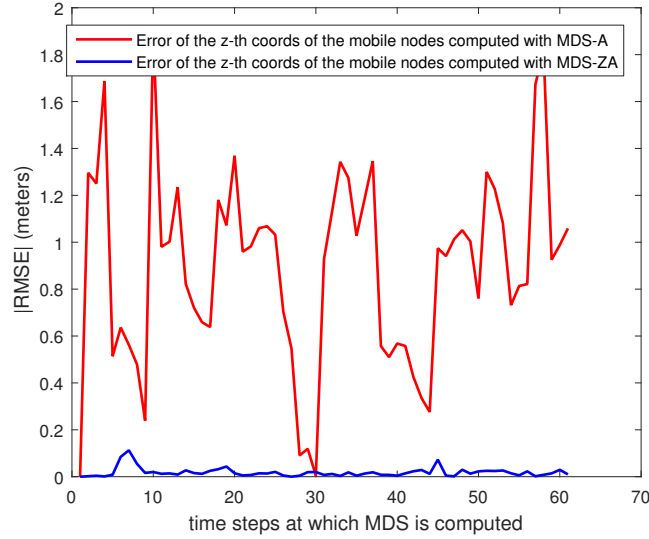


Figure 7: Norm of the error on the z-coordinate of the mobile nodes computed with MDS-A and MDS-ZA, computed with $m = 7$.

Since the nodes are attached to the belt, considering a fixed height in the approach leads to a minimal error. On the contrary, when the height is not constrained, the error can reach high values (almost up to 2 meters).

Similar results can be obtained by filtering the coordinates obtained as output. However, similarly to the idea of using the anchors' nodes directly in

the minimization, to use the z-information inside the MDS algorithm reduces the error at the minimization level, making the successive filtering even more accurate.

6.2.3. MDS-ZA vs DMDS-ZA

The last experiment analyzes the benefits of the anchors' constraints on DMDS. The nodes communicate using a TDMA mechanism with 9 slots of 0.1 seconds each. Thus, the value of Δt is equal to the resulting TDMA wheel period, that is 0.9 seconds. In order to measure the traveled distance needed to populate the Distance matrix for DMDS, a step counter algorithm has been implemented by following the approach proposed in [25]. The sampling rate of the IMU Hz is configured to 12 Hz.

Figure 8 shows the roll, pitch, and yaw measured with the Pozyx devices on the node monitoring the person walking back and forth. Also, the last sub-figure shows the step detection algorithm, where the curve is computed from the sum of all the accelerations, and the red dots are the detected steps. Implementation details can be found in [25].

In order to provide a bound on the accuracy of the DMDS approach, we also applied DMDS to the real traveled distance, measured with the Vicon System (i.e., the ground truth). The accuracy found with this approach not only can be used as a bound but avoid that a bad dead-reckoning technique will impact on the strategy lowering the performances. Figure 9 shows the results of the comparison between MDS-ZA, DMDS-ZA, and DMDS-ZA (with ground truth) and the values of the CDF are reported in Table 3.

Table 3: Mean error and standard deviation of the CDFs showed in Figure 9

Approach	μ	σ
DMDS-ZA with Gth	0.45	0.104
DMDS-ZA	0.51	0.185
MDS-ZA	0.47	0.171

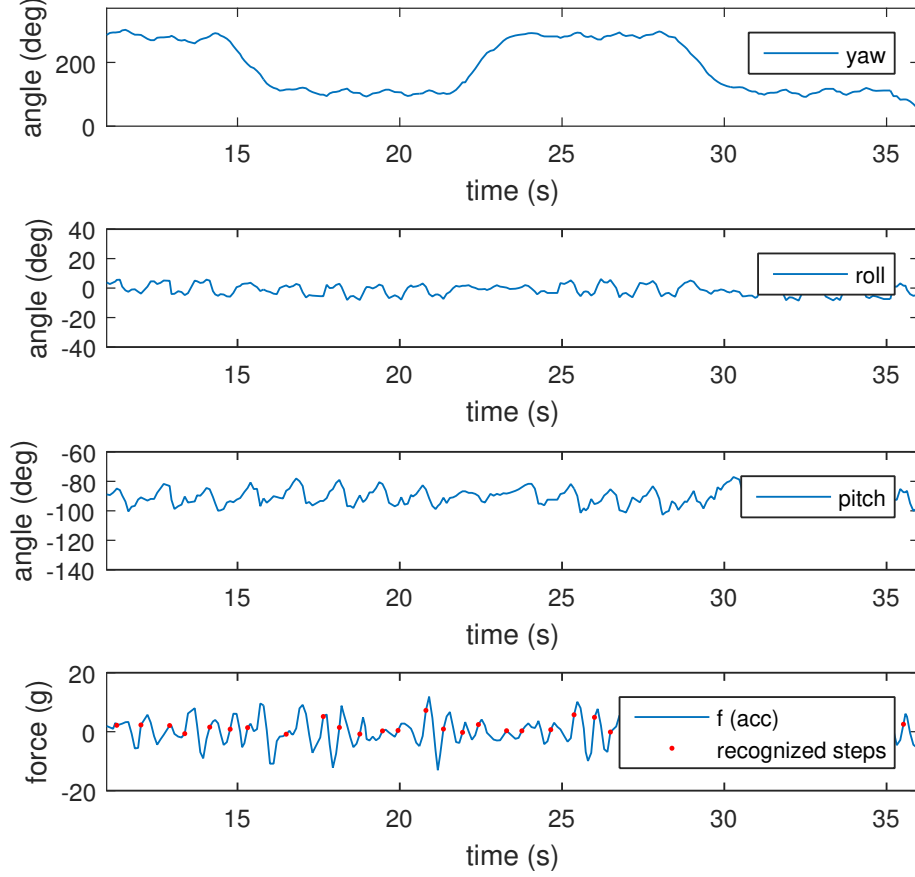


Figure 8: (From top to bottom) Yaw, roll, pitch, and step detection algorithm. Roll, pitch, and yaw are measured with the Pozyx devices at a frequency of 12 Hz. The step detection algorithm uses the linear accelerations measured by the IMU for estimating the number of steps and their frequency (red dots).

The DMDS approach with $p = 2$ has a limited impact on the minimization since only two more data are added to the Distance Matrix (i.e., the traveled distance of the two mobile nodes). However, despite the fact that the mean error is only slightly lower than MDS-ZA, the primary benefit of DMDS can be
410 seen in the very low standard deviation. Our DMDS-ZA implementation with pedestrian step detection reaches the performance of MDS-ZA with a slightly

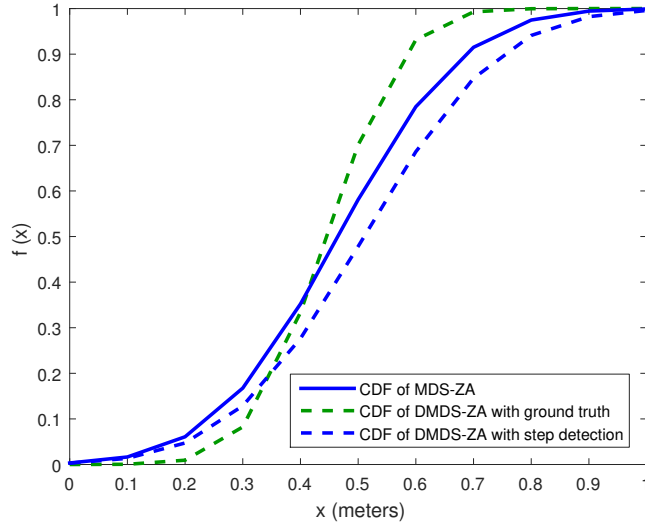


Figure 9: CDFs of MDS-ZA, DMDS-ZA using the ground truth to measure the traveled distance, and DMDS-ZA with a pedestrian step detection algorithm

higher mean error and standard deviation. Note that, the system used a low acquisition rate (caused by the Pozyx APIs) and the parameters used in the step detection were not optimal. The results of DMDS-ZA with the ground truth, instead, are encouraging, since the low standard deviation suggest that
415 the DMDS-ZA approach can be very efficient and further improve the already accurate MDS-ZA.

7. Conclusion

This work proposes a generalized formulation of the standard MDS and
420 DMDS techniques that actively consider the knowledge of the anchors coordinates in the minimization. Furthermore, MDS has also been modified to include the concept of heights constraints when this information is known. Real experiments show that the anchored versions of the algorithms (MDS-A and DMDS-A) outperform the respective MDS and DMDS approaches, showing ev-
425 ident improvements in the accuracy of estimation. Moreover, such improvements increase even more when also using the information of the node heights.

The results on DMDS-ZA show that it can improve the already accurate MDS-ZA if accurate odometry is given, making such approach promising in the context of node mobility. Other future works concern a more in-depth analysis and more experiments on the effect of a varying Δt and the use of DMDS with
430 o number of steps greater than two.

References

- [1] J. B. Kruskal, Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis, *Psychometrika* 29 (1) (1964) 1–27.
- 435 [2] I. Borg, P. Groenen, *Modern Multidimensional Scaling: Theory and Applications*, Springer Series in Statistics, Springer, 2005.
- [3] F. Cabero, Jose Maria sand De la Torre, A. Sanchez, I. Arizaga, Indoor people tracking based on dynamic weighted multidimensional scaling, in: *Proceedings of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems*, 2007, pp. 328–335.
440
- [4] Y. Shang, W. Ruml, Improved mds-based localization, in: *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, Vol. 4, IEEE, 2004, pp. 2640–2651.
- [5] Y. Shang, W. Rumi, Y. Zhang, M. Fromherz, Localization from connectivity in sensor networks, *Parallel and Distributed Systems, IEEE Transactions on* 15 (11) (2004) 961–974.
445
- [6] C. Di Franco, A. Prorok, N. Atanasov, B. Kempke, P. Dutta, V. Kumar, G. J. Pappas, Uwb network localization with non-line-of-sight range measurements, in: *Proceedings of the 16th International Conference on Information Processing in Sensor Networks*, IEEE Press, 2017, p. accepted for
450 publication.

- [7] S. Biaz, Y. Ji, Precise distributed localization algorithms for wireless networks, in: Sixth IEEE international symposium on a world of wireless mobile and multimedia networks, IEEE, 2005, pp. 388–394.
- 455 [8] X. Ji, H. Zha, Sensor positioning in wireless ad-hoc sensor networks using multidimensional scaling, in: INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, Vol. 4, IEEE, 2004, pp. 2652–2661.
- 460 [9] A. Amar, Y. Wang, G. Leus, Extending the classical multidimensional scaling algorithm given partial pairwise distance measurements, *Signal Processing Letters*, IEEE 17 (5) (2010) 473–476.
- [10] J. A. Costa, N. Patwari, A. O. Hero III, Distributed weighted-multidimensional scaling for node localization in sensor networks, *ACM Transactions on Sensor Networks (TOSN)* 2 (1) (2006) 39–64.
- 465 [11] L. Oliveira, H. Li, L. Almeida, T. E. Abrudan, Rssi-based relative localisation for mobile robots, *Ad Hoc Networks* 13 (2014) 321–335.
- [12] C. Di Franco, E. Bini, M. Marinoni, G. C. Buttazzo, Multidimensional scaling localization with anchors, in: *Autonomous Robot Systems and Competitions (ICARSC)*, 2017 IEEE International Conference on, IEEE, 2017, pp. 49–54.
- 470 [13] K. Ambrosi, J. Hansohm, *Ein Dynamischer Ansatz zur Repräsentation von Objekten*, Vol. 1986 of *Operations Research Proceedings*, Springer Berlin Heidelberg, 1987.
- [14] R. K. Garimella, W. Sheng, Dynamic localization of multiple mobile subjects in wireless ad hoc networks, in: *Intelligent Robots and Systems*, 2007. IROS 2007. IEEE/RSJ International Conference on, IEEE, 2007, pp. 2003–2008.
- 475 [15] K. S. Xu, M. Kliger, A. O. Hero III, Visualizing the temporal evolution of dynamic networks, *stress (X)* 1 (2011) 2.

- 480 [16] B. Beck, R. Baxley, Anchor free node tracking using ranges, odometry, and multidimensional scaling, in: Proceedings of the 39th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014, pp. 2209–2213.
- [17] H. Jamali-Rad, G. Leus, Dynamic multidimensional scaling for low-complexity mobile network tracking, IEEE Transactions on Signal Processing 60 (8) (2012) 4485–4491.
- 485 [18] L. Oliveira, L. Almeida, Rf-based relative position estimation in mobile ad-hoc networks with confidence regions, in: RoboCup International Symposium, 2014.
- 490 [19] C. Di Franco, A. Melani, M. Marinoni, Solving ambiguities in mds relative localization, in: Advanced Robotics (ICAR), 2015 International Conference on, IEEE, 2015, pp. 230–236.
- [20] G. Latsoudas, N. D. Sidiropoulos, A fast and effective multidimensional scaling approach for node localization in wireless sensor networks, IEEE Transactions on Signal Processing 55 (10) (2007) 5121–5127.
- 495 [21] K. W. Cheung, H.-C. So, A multidimensional scaling framework for mobile location using time-of-arrival measurements, IEEE transactions on signal processing 53 (2) (2005) 460–470.
- [22] J. De Leeuw, W. J. Heiser, Multidimensional scaling with restrictions on the configuration, Multivariate analysis 5 (1980) 501–522.
- 500 [23] J. De Leeuw, Applications of convex analysis to multidimensional scaling, Department of Statistics, UCLA.
- [24] Pozyx, Pozyx, accurate positioning (2015).
URL <https://www.pozyx.io/>
- 505 [25] L. Ilkovičová, P. Kajánek, A. Kopáček, Pedestrian indoor positioning and tracking using smartphone sensors, step detection and map matching al-

gorithm, in: International Symposium on Engineering Geodesy, 2016, pp. 20–22.

Appendix A. Proof of Theorem 1

The trace of a matrix is defined as the sum of the elements along the diagonal. Hence, the trace of the $r \times r$ matrix $\mathbf{X}^T \mathbf{V} \mathbf{X}$ of (5) is equal to

$$\text{tr}(\mathbf{X}^T \mathbf{V} \mathbf{X}) = \sum_{k=1}^r \mathbf{X}_{(k)}^T \mathbf{V} \mathbf{X}_{(k)}$$

with $\mathbf{X}_{(k)} \in \mathbb{R}^{N \times 1}$ being the k -th column vector of $\mathbf{X}$. Hence, Equation (5) can be rewrite as follows:

$$\begin{aligned} \mathcal{T}(\mathbf{X}, \mathbf{Z}) &= \mathbf{C} + \text{tr}(\mathbf{X}^T \mathbf{V} \mathbf{X}) - 2 \text{tr}(\mathbf{X}^T \mathbf{B}(\mathbf{Z}) \mathbf{Z}) \\ &= \mathbf{C} + \sum_{k=1}^r \mathbf{X}_{(k)}^T \mathbf{V} \mathbf{X}_{(k)} - 2 \sum_{k=1}^r \mathbf{X}_{(k)}^T \mathbf{B}(\mathbf{Z}) \mathbf{Z}_{(k)}. \end{aligned}$$

By partitioning in block the above expression, is possible to obtain

$$\begin{aligned} \mathcal{T}(\mathbf{X}, \mathbf{Z}) &= \mathbf{C} + \sum_{k=1}^r \mathbf{X}_{(k)}^T \mathbf{V} \mathbf{X}_{(k)} - 2 \sum_{k=1}^r \mathbf{X}_{(k)}^T \mathbf{B}(\mathbf{Z}) \mathbf{Z}_{(k)} \\ &= \mathbf{C} + \sum_{k=1}^r \left(\mathbf{X}_{u(k)}^T \mathbf{V}_{11} \mathbf{X}_{u(k)} + 2 \mathbf{X}_{u(k)}^T \mathbf{V}_{12} \mathbf{X}_{a(k)} + \mathbf{X}_{a(k)}^T \mathbf{V}_{22} \mathbf{X}_{a(k)} \right) \\ &\quad - 2 \sum_{k=1}^r \left(\mathbf{X}_{u(k)}^T \mathbf{B}_{11} \mathbf{Z}_{u(k)} + 2 \mathbf{X}_{u(k)}^T \mathbf{B}_{12} \mathbf{Z}_{a(k)} + \mathbf{X}_{a(k)}^T \mathbf{B}_{22} \mathbf{Z}_{a(k)} \right). \end{aligned} \tag{A.1}$$

By differentiating the expression of $\mathcal{T}(\mathbf{X}, \mathbf{Z})$ of Equation (A.1) with respect to the unknowns $\mathbf{X}_{u(k)}$ only, can be obtained that:

$$\frac{\partial \mathcal{T}(\mathbf{X}, \mathbf{Z})}{\partial \mathbf{X}_{u(k)}} = 2(\mathbf{V}_{11} \mathbf{X}_{u(k)} + \mathbf{V}_{12} \mathbf{X}_{a(k)} - \mathbf{B}_{11} \mathbf{Z}_{u(k)} - \mathbf{B}_{12} \mathbf{Z}_{a(k)}), \tag{A.2}$$

and by setting it equal to $\mathbf{0}$ it is possible to find the unknowns $\mathbf{X}_{u(k)}$, which are

$$\mathbf{X}_{u(k)} = \mathbf{V}_{11}^{-1} (\mathbf{B}_{11} \mathbf{Z}_{u(k)} + \mathbf{B}_{12} \mathbf{Z}_{a(k)} - \mathbf{V}_{12} \mathbf{X}_{a(k)}),$$

or, equivalently, in a matrix form

$$\mathbf{X}_u = \mathbf{V}_{11}^{-1} (\mathbf{B}_{11} \mathbf{Z}_u + \mathbf{B}_{12} \mathbf{Z}_a - \mathbf{V}_{12} \mathbf{X}_a). \tag{A.3}$$