

## Research Article

# Reduced-Complexity Algorithms for Indoor Map-Aware Localization Systems

Francesco Montorsi,<sup>1</sup> Fabrizio Pancaldi,<sup>2</sup> and Giorgio M. Vitetta<sup>1</sup>

<sup>1</sup>Department of Engineering “Enzo Ferrari”, University of Modena and Reggio Emilia, Via Pietro Vivarelli 10, 41125 Modena, Italy

<sup>2</sup>Department of Sciences and Methods for Engineering, University of Modena and Reggio Emilia, Via Amendola 2, 42122 Reggio Emilia, Italy

Correspondence should be addressed to Francesco Montorsi; francesco.montorsi@gmail.com

Received 26 May 2015; Revised 13 September 2015; Accepted 28 September 2015

Academic Editor: Gonzalo Seco-Granados

Copyright © 2015 Francesco Montorsi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The knowledge of environmental maps (i.e., map-awareness) can appreciably improve the accuracy of optimal methods for position estimation in indoor scenarios. This improvement, however, is achieved at the price of a significant complexity increase with respect to the case of map-unawareness, specially for large maps. This is mainly due to the fact that optimal map-aware estimation algorithms require integrating highly nonlinear functions or solving nonlinear and nonconvex constrained optimization problems. In this paper, various techniques for reducing the complexity of such estimators are developed. In particular, two novel strategies for restricting the search domain of map-aware position estimators are developed and the exploitation of state-of-the-art numerical integration and optimization methods is investigated; this leads to the development of a new family of suboptimal map-aware localization algorithms. Our numerical and experimental results evidence that the accuracy of these algorithms is very close to that offered by their optimal counterparts, despite their significantly lower computational complexity.

## 1. Introduction

Indoor localization systems have found widespread application in a number of different areas, both military and civilian [1]. In many cases, the maps of the environments where the users are supposed to lie (e.g., the floor plans of a given building or a spatial road network) are perfectly known and this form of a priori knowledge (dubbed *map-awareness* in the following) can be employed for improving localization accuracy. In particular, in the technical literature about localization systems map information is usually exploited to (a) improve the quality of position estimates generated by standard localization technologies; (b) set simple geometric constraints in the search space of *fingerprinting systems* (e.g., see [2–4]); (c) acquire an accurate knowledge of propagation of wireless signals on the basis of two-dimensional or three-dimensional *ray-tracing techniques* [5, 6]; (d) develop *map-based statistical models* for the measurements acquired by multiple anchors in a given indoor environment. It is important to point out the following:

- (i) Point (a) refers to the wide family of *map-matching algorithms* (see [7] and references therein), which includes both simple search techniques exploiting geometric and topological information provided by maps, and more refined techniques based on signal processing algorithms, such as Kalman filtering and particle filtering [8]. Distinct types of map-matching algorithms may offer a substantially different complexity-performance trade-off, but the improvement in localization accuracy is limited by the fact that map information is not exploited in the first stage of position estimation (i.e., in ranging). Note also that these algorithms have been mainly developed for transport applications (where they are employed to refine the estimated position provided by GPS or GPS integrated with *dead-reckoning* in outdoor environments). In principle, they can be also employed in indoor scenarios, provided that a proper map (e.g., a schematic representation of the possible user positions such as a Voronoi diagram) is available

for the considered environment; for instance, map-matching methods based on particle filtering have been proposed in [9–13] for localization systems operating in indoor scenarios.

- (ii) *Fingerprinting methods* (see point (b)) make a limited use of the geometrical properties of the propagation scenario, since they mainly rely on a training database. In addition, the preliminary measurement campaign necessary to generate such a database can represent a formidable task in large buildings and/or time-varying scenarios.
- (iii) *Ray-tracing techniques* (see point (c)) can certainly outperform the above-mentioned techniques in terms of accuracy; however, real-time ray-tracing is often unfeasible because of its large computational complexity and the detailed knowledge it requires about various physical properties (e.g., electrical permittivity of walls and boundary conditions) of localization scenarios.
- (iv) Point (d) refers to the so-called *map-aware statistical localization techniques*, which have been developed for *time of arrival* (TOA), *time difference of arrival* (TDOA), and *received signal strength-* (RSS-) based indoor localization systems [14, Sec. 4.11.5], [15, 16]. Such techniques rely on the availability of *map-aware statistical signal models* and are characterized by the following relevant features: (a) they are able to compensate for the *non-line-of-sight* (NLOS) *bias* (i.e., for the extra delay or extra attenuation due to propagation through obstructions, mainly walls), which represents a major source of error in indoor localization; (b) their use requires measurement campaigns substantially less time consuming than those needed for fingerprinting techniques in the same scenarios. In addition, previous work illustrated in [15, 16] has evidenced that these models can be easily combined with optimal estimation techniques to derive novel localization algorithms.

This paper represents a follow-up to [15], where new map-aware statistical models based on experimental measurements have been exploited to develop optimal *map-aware minimum mean square error* (MMSE) and *maximum a posteriori* (MAP) estimators. Our previous work has evidenced that these estimators may substantially outperform their map-unaware counterparts at the price, however, of a significant complexity increase, specially for large maps. In this paper, the problem of complexity reduction of map-aware MMSE and MAP estimators is tackled and the following novel contributions are developed:

- (1) Two novel algorithms for restricting the domain over which the agent position is searched for in map-aware estimation are devised. These algorithms, dubbed *distance-reduced domain* (DRD) and *probability-reduced domain* (PRD) in the following, can reduce the rate at which the complexity of map-aware estimators increases with map size without substantially affecting localization accuracy.

- (2) The application of specific mathematical tools (namely, *cubature rules* for numerical integration [17, 18] and *direct-search methods* [19, 20]), which can be exploited to reduce the implementation complexity of map-aware algorithms, is analysed.
- (3) Novel suboptimal map-aware algorithms resulting from the combination of the above-mentioned algorithms and mathematical tools are proposed and are compared, in terms of both *root median square error* (RMSE) and computational complexity (namely, overall number of *floating point operations* (FLOPs)), to *maximum likelihood* (ML) estimators (note that ML estimators are optimal for *map-unaware* localization and have been widely adopted for their limited computational complexity [21]).

The following is worth pointing out:

- (i) Most of the performance results illustrated in this paper mainly rely on a set of data generated according to the statistical model developed in [15], which, in turn, is based on experimental measurements. This approach, which is commonly adopted in the technical literature (e.g., see [22, 23]), is motivated by the fact that experimental databases referring to localization systems have usually a limited size; this is mainly due to the time consuming tasks required by any measurement campaign in this field. However, some performance results evaluated on the basis of our experimental database are also shown.
- (ii) In our work only RSS-based localization algorithms are considered, even if the proposed approach can be easily extended to TOA and TDOA based systems, since all rely on similar statistical signal models [15].
- (iii) Our contribution focuses on the role played by *map-awareness* in the localization of still targets in static environments (consequently, the potential improvement deriving from the knowledge of mobility models is not taken into consideration).

The remaining part of this paper is organized as follows. In Section 2, some models for indoor maps and measurements in map-aware RSS-based localization systems are illustrated. In Section 3, optimal and suboptimal localization algorithms based on the proposed models are developed, whereas in Section 4 various problems arising from their implementation are analysed and some solutions are proposed. In Section 5 various numerical results about the performance and complexity of the devised algorithms are illustrated. Finally, in Section 6, some conclusions are drawn.

*Notations.* The *probability density function* (pdf) of a random vector  $\mathbf{R}$  evaluated at the point  $\mathbf{r}$  is denoted  $f(\mathbf{r})$ ;  $\mathcal{N}(\mathbf{r}; \mathbf{m}, \mathbf{\Sigma})$  denotes the pdf of a Gaussian random vector  $\mathbf{R}$  having mean  $\mathbf{m}$  and covariance matrix  $\mathbf{\Sigma}$ , evaluated at the point  $\mathbf{r}$ ;  $\text{diag}\{\cdot\}$  denotes a square matrix having the arguments on its main diagonal and zeros elsewhere;  $|\mathcal{S}|$  denotes the cardinality of the set  $\mathcal{S}$ ;  $x_i$  denotes the  $i$ th element of the vector  $\mathbf{x}$  and  $\lfloor \mathbf{x} \rfloor$  denotes its size;  $\lfloor x \rfloor$  denotes the floor of  $x$ ;  $\text{proj}\{\mathcal{S}, \mathbf{p}\}$  denotes

the Euclidean projection of  $\mathbf{p}$  over a domain  $\mathcal{S}$ , that is, the point  $\mathbf{q} \in \mathcal{S}$  which minimises the Euclidean distance with  $\mathbf{p}$ ;  $\text{centre}\{\mathcal{S}\}$  denotes the centre of the rectangle  $\mathcal{S}$ ;  $\mathbf{e}_x$  ( $\mathbf{e}_y$ ) is the unit vector along the  $x$  ( $y$ ) axis.

## 2. Reference Scenario and Signal Model

In the following, we focus on a *two-dimensional* (2D) RSS-based localization system employing  $N_a$  devices, called *anchors*, and whose positions  $\{\mathbf{p}_i^a\}_{i=1}^{N_a}$  are known, to estimate the position  $\mathbf{p} \triangleq [x, y]^T$  of an *agent* (see Figure 1). The mathematical models defined in [15] for the map, the wireless connectivity between the agent and the anchors, and the measurements acquired for position estimation are adopted; their essential features are summarised below.

**2.1. Map Model.** A *rectangular uniform map*, having support  $\mathcal{R}$  and area  $\mathcal{A}_{\mathcal{R}}$ , is assumed in the following, since it provides a good approximation of the floor plans of typical buildings (see [15, eq. (1)]). This map consists of the union of  $N_r$  nonoverlapping rectangles  $\{\mathcal{R}_k \subset \mathbb{R}^2\}_{k=1}^{N_r}$  having their sides parallel to the axes of the adopted reference frame and representing the so-called *rectangular covering* of  $\mathcal{R}$ , so that  $\mathcal{R} = \bigcup_{k=1}^{N_r} \mathcal{R}_k$  (see Figure 1). Note that, generally speaking, the support  $\mathcal{R}$  of a rectangular map can be approximated by different coverings  $\{\mathcal{R}_k\}$  and there is no one-to-one mapping between the support  $\mathcal{R}$  of a (rectangular) map and its covering  $\{\mathcal{R}_k\}$ . However, various algorithms are already available in the technical literature to partition a generic polygon into a set of rectangles (e.g., see [24]) and, in particular, to generate a *minimal nonoverlapping covering* (MNC) [25, 26] of  $\mathcal{R}$  that is a partition consisting of a *minimal* number of rectangles for the considered domain. In the following, a MNC of  $\mathcal{R}$ , which is a partition  $\{\mathcal{R}_k^{\text{MNC}}\}$  consisting of a *minimal* number of rectangles for this domain, or a *dense nonoverlapping covering* (DNC)  $\{\mathcal{R}_k^{\text{DNC}}\}$ , which originates from splitting the MNC rectangles whose areas or side lengths exceed a given threshold, is considered, so that

$$\mathcal{R} = \bigcup_{k=1}^{N_r^{\text{MNC}}} \mathcal{R}_k^{\text{MNC}} = \bigcup_{k=1}^{N_r^{\text{DNC}}} \mathcal{R}_k^{\text{DNC}}; \quad (1)$$

here  $N_r^{\text{MNC}}$  ( $N_r^{\text{DNC}}$ ) is the number of rectangles in the selected MNC (DNC) of  $\mathcal{R}$ . As it will become clearer later, a MNC of  $\mathcal{R}$  can be very useful to reduce the computational load of some map-aware localization algorithms; some algorithms, however, perform better if a DNC is selected. In localization systems,  $\mathcal{R}$  is usually time invariant and such coverings can be computed offline resorting to a number of optimized (but complicated) algorithms (e.g., see [25, 26] and references therein); in our work, however, the simpler algorithm described in [27] is adopted, since it can handle maps containing “holes” (e.g., inaccessible areas of a building floor) and its input data can be easily extracted from the floor plans of typical buildings.

Note that our approach to map-aware localization requires not only the knowledge of a MNC (or, equivalently, a

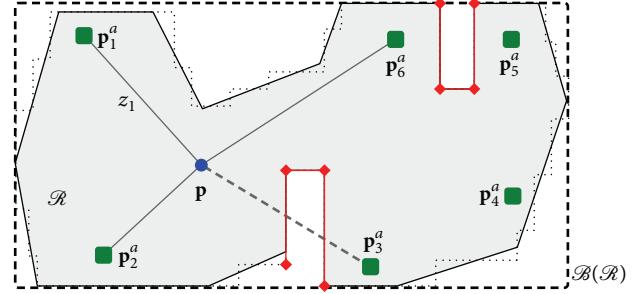


FIGURE 1: Localization system employing  $N_a = 6$  anchors (green squares) to localise a single agent (blue circle) and operating in a bounded map (whose support  $\mathcal{R}$  is identified by the grey area). The dashed line denotes the map bounding box  $\mathcal{B}(\mathcal{R})$ . LOS (NLOS) links are evidenced by continuous (dashed) lines (note that  $N_{\text{obs}} = 4$  in the considered case, since the agent is not connected to anchors 4 and 5). The dotted line represents the outer edge of a possible rectangular covering, whereas the  $N_s = 6$  segments representing the environmental obstructions are evidenced in red.

DNC) of the map support  $\mathcal{R}$ , but also that of the obstructions (e.g., walls) the map contains; this knowledge is exploited to evaluate the function  $N_o(\mathbf{p}_1, \mathbf{p}_2)$ , which represents the number of obstructions interposed between two arbitrary points  $\mathbf{p}_1$  and  $\mathbf{p}_2$  of  $\mathcal{R}$ . On the contrary, *map-unaware* localization systems, which are introduced later for comparison, rely on the knowledge of the *bounding box*  $\mathcal{B}(\mathcal{R})$  of  $\mathcal{R}$  only.

**2.2. Connectivity Model.** In our work, the coverage region  $\mathcal{R}^{(i)}$  of the  $i$ th anchor (with  $i = 1, 2, \dots, N_a$ ) is assumed to consist of a circle centred at  $\mathbf{p}_i^a$  whose radius  $d_{\text{max},i}$  depends on the transmitted power and signal propagation conditions; inside this region, if the agent is connected with the associated anchor, it acquires a single *observation*  $z_i$  (in particular, a RSS measurement) for localization purposes by exchanging wireless signals with the anchor itself [15, Sec. II.B.] (if multiple measurements are acquired by the  $i$ th anchor, they are averaged or filtered in order to generate a single observation  $z_i$ ). In practice, in the presence of harsh propagation conditions, the shape of the coverage region is likely to substantially differ from a circular shape, so that the number  $N_{\text{obs}}$  of observations available to the agent cannot be predicted theoretically; for this reason, the parameter  $N_{\text{obs}} \triangleq |\mathcal{I}|$  is defined, where  $\mathcal{I}$  is the set of indices associated with the anchors *truly connected* with the agent. It is easy to show that if  $i \in \mathcal{I}$ , then  $\mathbf{p} \in \mathcal{R}^{(i)} \cap \mathcal{R}$ ; consequently, in a *map-aware* localization system the agent position has to be searched for inside the domain  $\mathcal{R}^{(\mathcal{I})} \triangleq (\bigcap_{i \in \mathcal{I}} \mathcal{R}^{(i)}) \cap \mathcal{R}$ . In our work, the region  $\mathcal{R}^{(\mathcal{I})}$  is assumed to be well approximated by the rectangular region that consists of all the covering rectangles intersecting with the exact  $\mathcal{R}^{(\mathcal{I})}$ ; that is,

$$\mathcal{R}^{(\mathcal{I})} \simeq \bigcup_{k \in \mathcal{I}} \mathcal{R}_k, \quad (2)$$

where

$$\mathcal{K} \triangleq \{k = 1, \dots, N_r \mid \mathcal{R}_k \cap \mathcal{R}^{(i)} \neq \emptyset, \forall i \in \mathcal{I}\} \quad (3)$$

and  $\mathcal{R}_k$  belongs to a MNC or a DNC set (in the following, unless explicitly stated, the symbol  $\mathcal{R}_k(N_r)$  denotes  $\mathcal{R}_k^{\text{DNC}}(N_r^{\text{DNC}})$  or  $\mathcal{R}_k^{\text{MNC}}(N_r^{\text{MNC}})$ ). Note that applying this connectivity model to the considered localization problem results in a preliminary *domain reduction*; in fact, instead of considering the  $N_r^{\text{MNC}}$  or  $N_r^{\text{DNC}}$  rectangles covering the whole map support, all the proposed localization algorithms restrict their search domain to  $N_r^{\text{SRC}} \triangleq |\mathcal{K}|$  rectangles approximating  $\mathcal{R}^{(\mathcal{I})}$  (2). In a *map-unaware* system, instead, the agent position is expected to belong to

$$\mathcal{P}^{(\mathcal{I})} \triangleq \left( \bigcap_{i \in \mathcal{I}} \mathcal{R}^{(i)} \right) \cap \mathcal{B}(\mathcal{R}). \quad (4)$$

In this case, it is assumed that for any  $i$  the domain  $\mathcal{R}^{(i)}$  can be approximated by a square region whose centre is  $\mathbf{p}_i^a$  and whose side is  $2d_{\max,i}$ ; this implies that  $\mathcal{P}^{(\mathcal{I})}$  (4) consists of a single rectangle.

Finally, we note that the sets  $\mathcal{I}$  and  $\mathcal{K}$  in (3) and the domains  $\mathcal{R}^{(\mathcal{I})}$  (2) and  $\mathcal{P}^{(\mathcal{I})}$  (4) are all available to the agent localization algorithm immediately after all RSS observations have been acquired.

**2.3. Statistical Modelling of Observations.** We consider a wireless localization system inferring the agent position  $\mathbf{p}$  from a set of  $N_{\text{obs}}$  RSS observations  $\{z_i, i \in \mathcal{I}\}$ , with  $N_{\text{obs}} \leq N_a$ . In [15] it is shown that, given the trial agent position  $\tilde{\mathbf{p}} \in \mathcal{R}^{(\mathcal{I})}$ , the *map-aware likelihood* for the observation vector  $\mathbf{z} \triangleq [z_i, \forall i \in \mathcal{I}]^T \in \mathbb{R}^{N_{\text{obs}}}$  can be expressed as (see [15, Eq. (13)])

$$\begin{aligned} f(\mathbf{z} \mid \tilde{\mathbf{p}}) &= \prod_{i \in \mathcal{I}} \mathcal{N}(z_i; d_i(\tilde{\mathbf{p}}) + \mu_{b,i}(\tilde{\mathbf{p}}), \sigma_{b,i}^2(\tilde{\mathbf{p}}) + \sigma_{n,i}^2(\tilde{\mathbf{p}})). \end{aligned} \quad (5)$$

Here  $d_i(\tilde{\mathbf{p}}) \triangleq \|\tilde{\mathbf{p}} - \mathbf{p}_i^a\|$  is the Euclidean distance between the trial position  $\tilde{\mathbf{p}}$  and the  $i$ th anchor and

$$\begin{aligned} \mu_{b,i}(\tilde{\mathbf{p}}) &\triangleq (\mu_{b,0} + \mu_{b,m} N_o(\tilde{\mathbf{p}}, \mathbf{p}_i^a)) u_{N_o}, \\ \sigma_{b,i}(\tilde{\mathbf{p}}) &\triangleq \max\{\sigma_{b,0} - \sigma_{b,m} N_o(\tilde{\mathbf{p}}, \mathbf{p}_i^a), 0\} \end{aligned} \quad (6)$$

denote the mean and the standard deviation, respectively, of the *bias*  $b_i(\mathbf{p}) \sim \mathcal{N}(\mu_{b,i}(\mathbf{p}), \sigma_{b,i}^2(\mathbf{p}))$  affecting the  $i$ th observation  $z_i$  and originating from signal propagation inside the obstructions, and

$$\sigma_{n,i}(\tilde{\mathbf{p}}) \triangleq \sigma_{n,0} \left( \frac{d_i(\tilde{\mathbf{p}})}{d_0} \right)^{\beta_n} \quad (7)$$

represents the standard deviation of the bias-unrelated and position-dependent *noise*  $n_i(\mathbf{p}) \sim \mathcal{N}(0, \sigma_{n,i}^2(\mathbf{p}))$  affecting the same observation. Note that the likelihood function  $f(\mathbf{z} \mid \tilde{\mathbf{p}})$  (5) is influenced by the structure of the map support  $\mathcal{R}$

through the parameters  $\mu_{b,i}(\tilde{\mathbf{p}})$  and  $\sigma_{b,i}(\tilde{\mathbf{p}})$ , since both these quantities depend on  $N_o(\tilde{\mathbf{p}}, \mathbf{p}_i^a)$ .

Map-unaware systems, instead, cannot rely on the knowledge of the function  $N_o(\cdot, \cdot)$  in bias modelling. Moreover, state-of-the-art map-unaware systems often employ a LOS/NLOS detection stage before localization, whose output  $\widehat{\mathcal{I}}^{\text{NLOS}}$  ( $\widehat{\mathcal{I}}^{\text{LOS}}$ ) is the subset of  $\mathcal{I}$  containing the indices of the anchors detected to experience NLOS (LOS) conditions with the agent. Given  $\widehat{\mathcal{I}}^{\text{NLOS}}$  and the trial agent position  $\tilde{\mathbf{p}} \in \mathcal{P}^{(\mathcal{I})}$ , the *map-unaware likelihood* (see [15, Eq. (15)])

$$\begin{aligned} f(\mathbf{z}; \tilde{\mathbf{p}}) &= \prod_{i \in \mathcal{I}} \mathcal{N}(z_i; d_i(\tilde{\mathbf{p}}) + \mu_{b,i}^{\text{MU}}(\tilde{\mathbf{p}}), (\sigma_{b,i}^{\text{MU}}(\tilde{\mathbf{p}}))^2 \\ &\quad + (\sigma_{n,i}^{\text{MU}}(\tilde{\mathbf{p}}))^2) \end{aligned} \quad (8)$$

is adopted for the same observation vector  $\mathbf{z}$  considered in (5). Here  $\mu_{b,i}^{\text{MU}} \triangleq \kappa_b$  and  $\sigma_{b,i}^{\text{MU}} \triangleq \gamma_b$  for  $i \in \widehat{\mathcal{I}}^{\text{NLOS}}$  (both are zero otherwise) and  $\sigma_{n,i}^{\text{MU}}(\tilde{\mathbf{p}}) = \sigma_{n,0}^{\text{MU}}(d_i(\tilde{\mathbf{p}})/d_0)^{\beta_n^{\text{MU}}}$ .

**2.4. Calibration.** The parameters appearing in both the map-aware and map-unaware likelihood functions (see (5) and (8), resp.) need to be extracted from a set of experimental measurements, as illustrated in [15]. In our work, the performance of the developed localization algorithms has been assessed on the floors of 3 different buildings of the University of Modena and Reggio Emilia (these buildings mainly host offices and laboratories). In each scenario, 15–20 measurement sites have been identified in order to acquire RSS data referring to almost 100 independent links. Then, the method illustrated in [15] has been exploited to extract the values of the unknown parameters appearing in (5) and (8); this has produced the following values:  $\mu_{b,0} = 12.6$  m,  $\mu_{b,m} = 2.53$  m,  $\sigma_{b,0} = 7.07$  m,  $\sigma_{b,m} = 3.0$  m,  $\sigma_{n,0} = 2.47$  m,  $\beta_n = 0.21$ , and  $d_0 = 1$  m for the map-aware model (5) and the parameters  $\kappa_b = 21$  m,  $\gamma_b = 2.81$  m,  $\sigma_{n,0}^{\text{MU}} = 4.47$  m, and  $\beta_n^{\text{MU}} = 0.19$  for the map-unaware model (8) (please see [15, Table I] for further details). It is important to note the following:

- (i) In our experimental campaign, we have also found that the same values of the considered parameters can accurately describe the statistical models to be adopted for RSS measurements in different scenarios; this result is motivated by the fact that actually similar propagation conditions are experienced in distinct buildings (which can be classified as *light commercial* in standard terminology), even if the areas of the considered propagation scenarios (floor plans) are significantly different (see Section 5).
- (ii) In principle, a preliminary measurement campaign is always required to adapt models (5) and (8) to different propagation scenarios. This need is common to various localization techniques, which may require a time consuming training phase.
- (iii) Map-aware solutions require a proper preprocessing of the map to compute its MNC or DNC and to evaluate the function  $N_o(\mathbf{p}_1, \mathbf{p}_2)$  (see Section 2.1);

however, this step can be carried out offline and does not appreciably affect the overall computational complexity (see also Section 4.1).

### 3. Estimation Algorithms

In this Section, optimal map-aware (namely, MMSE and MAP) and map-unaware (namely, ML) localization algorithms are analysed first. Then, two novel heuristic (suboptimal) strategies are proposed to reduce the computational load required by optimal estimation algorithms.

**3.1. Optimal Map-Unaware Estimator.** In a map-unaware context (i.e., in the absence of prior information about the position of the agent), the ML approach can be adopted to generate the optimal estimate  $\hat{\mathbf{p}}_{\text{ML}}(\mathbf{z}) \triangleq \arg \max_{\tilde{\mathbf{p}}} \ln f(\mathbf{z}; \tilde{\mathbf{p}})$  of  $\mathbf{p}$  [28], where  $f(\mathbf{z}; \tilde{\mathbf{p}})$  is given by (8). In practice, this estimate can be evaluated as

$$\hat{\mathbf{p}}_{\text{ML}}(\mathbf{z}) = \arg \min_{\tilde{\mathbf{p}} \in \mathcal{D}(\mathcal{Z})} \left\{ \sum_{i \in \mathcal{Z}} \frac{(z_i - d_i(\tilde{\mathbf{p}}) - \mu_{b,i}^{\text{MU}})^2}{(\sigma_{n,i}^{\text{MU}}(\tilde{\mathbf{p}}))^2 + (\sigma_{b,i}^{\text{MU}})^2} + 2 \ln \prod_{i \in \mathcal{Z}} \sqrt{(\sigma_{n,i}^{\text{MU}}(\tilde{\mathbf{p}}))^2 + (\sigma_{b,i}^{\text{MU}})^2} \right\}, \quad (9)$$

where  $\mathcal{D}(\mathcal{Z})$  is a single rectangle (see Section 2.2). The last expression deserves the following comments:

- (1) The cost function appearing in (9) exhibits a discontinuous behaviour, even if it does not contain map-aware functions (e.g.,  $N_o(\cdot, \cdot)$ ), since  $\mu_{b,i}^{\text{MU}}$  and  $\sigma_{b,i}^{\text{MU}}$  are assigned a constant value or zero depending on  $\mathcal{Z}^{\text{NLOS}}$ . Consequently, in principle, optimization methods involving gradients or Hessians cannot be used, unless the cost function is, at least *approximately*, continuous.
- (2) The evaluation of  $\hat{\mathbf{p}}_{\text{ML}}(\mathbf{z})$  in (9) requires solving a constrained least squares problem with a quadratic regularization term (represented by the product appearing in the right-hand side of (9)) which penalizes trial positions  $\tilde{\mathbf{p}}$  affected by large noise and bias. Unluckily, standard optimization methods do not necessarily achieve the *global* minimum; when this occurs, large localization errors are found (note that some methods developed to mitigate the problem of local minima, like *projection onto convex sets* (POCS) and *multidimensional scaling* (MDS), cannot be straightforwardly applied in this case because of the significant complexity of our observation model).
- (3) The complexity of the cost function in (9) can be related to  $N_{\text{obs}}$  as  $\mathcal{O}(N_{\text{obs}})$ .

**3.2. Optimal Map-Aware Estimators.** In a map-aware context, a MMSE or a MAP approach can be employed in position

estimation. In particular, it is not difficult to show that the MMSE estimate  $\hat{\mathbf{p}}_{\text{MMSE}}(\mathbf{z})$  of  $\mathbf{p}$  is given by [28]

$$\hat{\mathbf{p}}_{\text{MMSE}}(\mathbf{z}) \triangleq \mathbb{E}_{\mathbf{p}} \{\mathbf{p} \mid \mathbf{z}\} = \frac{\left[ \frac{\iint_{\mathcal{R}} x f(\mathbf{z}; \mathbf{p}) d\mathbf{p}}{\iint_{\mathcal{R}} y f(\mathbf{z}; \mathbf{p}) d\mathbf{p}} \right]}{\iint_{\mathcal{R}} f(\mathbf{z}; \mathbf{p}) d\mathbf{p}}. \quad (10)$$

Substituting (5) in (10) yields, after some manipulation,

$$\hat{\mathbf{p}}_{\text{MMSE}}(\mathbf{z}) = \frac{\left[ \frac{\iint_{\mathcal{R}(\mathcal{Z})} (xh(\mathbf{z}; \mathbf{p}) / \prod_{i \in \mathcal{Z}} \sqrt{\sigma_{n,i}^2(\mathbf{p}) + \sigma_{b,i}^2(\mathbf{p})}) d\mathbf{p}}{\iint_{\mathcal{R}(\mathcal{Z})} (yh(\mathbf{z}; \mathbf{p}) / \prod_{i \in \mathcal{Z}} \sqrt{\sigma_{n,i}^2(\mathbf{p}) + \sigma_{b,i}^2(\mathbf{p})}) d\mathbf{p}} \right]}{\iint_{\mathcal{R}(\mathcal{Z})} \left( h(\mathbf{z}; \mathbf{p}) / \prod_{i \in \mathcal{Z}} \sqrt{\sigma_{n,i}^2(\mathbf{p}) + \sigma_{b,i}^2(\mathbf{p})} \right) d\mathbf{p}}, \quad (11)$$

where

$$h(\mathbf{z}, \mathbf{p}) \triangleq \exp \left[ -\frac{1}{2} \sum_{i \in \mathcal{Z}} \left( \frac{(z_i - d_i(\mathbf{p}) - \mu_{b,i}(\mathbf{p}))^2}{\sigma_{n,i}(\mathbf{p})} \right) \right]. \quad (12)$$

The last result deserves the following comments:

- (1) The MMSE estimation procedure, unlike its MAP and ML counterparts, does not involve an optimization step and, consequently, does not suffer from problem of local minima. However, it requires numerical multidimensional integration.
- (2) Our numerical results have evidenced that (a) the product  $\prod_{i \in \mathcal{Z}} \sqrt{\sigma_{n,i}^2(\mathbf{p}) + \sigma_{b,i}^2(\mathbf{p})}$  appearing in (11) exhibits small variations inside the considered map if the anchor density is approximately constant and (b) the accuracy of MMSE estimation is negligibly influenced by the presence of this term for any  $\mathbf{p} \in \mathcal{R}$  such that  $d_i(\mathbf{p}) \gg 1 \forall i \in \mathcal{Z}$  (i.e., for any agent position far from the anchors). Then, discarding this term yields the approximate (and computationally simpler) variant

$$\hat{\mathbf{p}}_{\text{MMSE}}(\mathbf{z}) \simeq \frac{\left[ \frac{\iint_{\mathcal{R}(\mathcal{Z})} x h(\mathbf{z}; \mathbf{p}) d\mathbf{p}}{\iint_{\mathcal{R}(\mathcal{Z})} y h(\mathbf{z}; \mathbf{p}) d\mathbf{p}} \right]}{\iint_{\mathcal{R}(\mathcal{Z})} h(\mathbf{z}; \mathbf{p}) d\mathbf{p}} \quad (13)$$

of the MMSE estimator.

- (3) The computational load required by the evaluation of  $N_o(\cdot, \cdot)$  is  $\mathcal{O}(N_s)$  (see Section 2.1), where  $N_s$  is the *number of segment intersection tests accomplished in processing a given observation*; consequently, the complexity required by  $h(\mathbf{z}, \mathbf{p})$  (12) is approximately  $\mathcal{O}(\bar{N}_s \cdot N_{\text{obs}})$ , where  $\bar{N}_s$  is the average of  $N_s$  evaluated over the observations collected in  $\mathbf{z}$  (note that  $\bar{N}_s$  is nonlinearly related to the shape of obstructions, the trial agent position, and the anchor positions).

An alternative to the MMSE estimator is offered by the so-called MAP estimator, which can be expressed as  $\hat{\mathbf{p}}_{\text{MAP}}(\mathbf{z}) \triangleq \arg \max_{\tilde{\mathbf{p}}} \ln f(\mathbf{z} \mid \tilde{\mathbf{p}})$  [28] or, if (5) and Bayes' rule are exploited, as

$$\hat{\mathbf{p}}_{\text{MAP}}(\mathbf{z}) = \arg \min_{\tilde{\mathbf{p}} \in \mathcal{D}(\mathcal{Z})} f_c(\tilde{\mathbf{p}}), \quad (14)$$

where

$$\begin{aligned}
f_c(\tilde{\mathbf{p}}) &\triangleq -2 \ln f(\tilde{\mathbf{p}} | \mathbf{z}) \\
&= \sum_{i \in \mathcal{I}} \frac{(z_i - d_i(\tilde{\mathbf{p}}) - \mu_{b,i}(\tilde{\mathbf{p}}))^2}{\sigma_{n,i}^2(\tilde{\mathbf{p}}) + \sigma_{b,i}^2(\tilde{\mathbf{p}})} \\
&\quad + 2 \ln \prod_{i \in \mathcal{I}} \sqrt{\sigma_{n,i}^2(\tilde{\mathbf{p}}) + \sigma_{b,i}^2(\tilde{\mathbf{p}})}
\end{aligned} \tag{15}$$

is the MAP *cost function*. This function deserves the following comments:

- (1) It is not differentiable, since it depends on  $N_o(\cdot, \cdot)$ . Consequently, its gradient/Hessian matrix cannot be evaluated analytically and, in principle, steepest descent methods cannot be used; in practice, such methods can be exploited if it is assumed that the cost function is *approximately* continuous.
- (2) For large  $N_{\text{obs}}$ , it is characterized by a better numerical stability than that of the function  $h(\mathbf{z}, \mathbf{p})$  (12). In fact, for large  $N_{\text{obs}}$ , the argument of the function  $\exp(\cdot)$  appearing in  $h(\mathbf{z}, \mathbf{p})$  becomes small, so that the resulting numerical values of  $h(\mathbf{z}, \mathbf{p})$  itself may quickly drop below machine precision. On the contrary, this problem is not experienced with  $f_c(\tilde{\mathbf{p}})$  (15), because of the presence of a natural logarithm in its expression.
- (3) Its computational complexity is similar to that of  $h(\mathbf{z}, \mathbf{p})$ , which is  $\mathcal{O}(\bar{N}_s \cdot N_{\text{obs}})$ .

**3.3. Distance-Reduced Domain Map-Aware Estimator.** The computational complexity of the MMSE and MAP estimators described above becomes unacceptable when the considered map is large. To mitigate this problem, a new technique for restricting the domain over which the agent position is searched for has been developed; it consists of the following three steps:

- (1) *Map-Unaware Estimation.* A raw estimate  $\hat{\mathbf{p}}_{\text{raw}}(\mathbf{z})$  is generated by the map-unaware estimator (9), without performing any NLOS correction (this is equivalent to assuming  $\widehat{\mathcal{L}}^{\text{NLOS}} \equiv \mathcal{L}$  and  $\widehat{\mathcal{L}}^{\text{NLOS}} = \emptyset$  in (9)).
- (2) *Domain Reduction.* A portion

$$\mathcal{R}_{\text{red}}^{(\mathcal{X})} \triangleq \bigcup_{k \in \mathcal{K}} \mathcal{R}_k \tag{16}$$

of  $\mathcal{R}$  close to  $\hat{\mathbf{p}}_{\text{raw}}$  is extracted, where  $\mathcal{K}$  is a subset of the values of the index  $k$  associated with the set of rectangles  $\{\mathcal{R}_k\}$  forming  $\mathcal{R}^{(\mathcal{X})}$  (2); in practice,  $\mathcal{K}$  is generated according to the heuristic criterion

$$\mathcal{K} \triangleq \{k \in \mathcal{K} : \|\hat{\mathbf{p}}_{\text{raw}} - \text{proj}\{\mathcal{R}_k, \hat{\mathbf{p}}_{\text{raw}}\}\| < d_{\text{th}}\}, \tag{17}$$

where  $d_{\text{th}} \triangleq \text{median}_{i \in \mathcal{I}} \{\gamma \sqrt{\sigma_{b,i}^2(\hat{\mathbf{p}}_{\text{raw}}) + \sigma_{n,i}^2(\hat{\mathbf{p}}_{\text{raw}})}\}$  is a *threshold distance*,  $\gamma$  is a fixed parameter, and  $\mathcal{K}$  is defined by (3). Note that the definition given above for  $d_{\text{th}}$  entails that only the covering rectangles within  $\gamma$  standard deviations from  $\hat{\mathbf{p}}_{\text{raw}}$  are selected.

- (3) *Map-Aware Estimation.* The final estimate  $\hat{\mathbf{p}}_{\text{DRD}}(\mathbf{z})$  is generated using either the MMSE (13) or the MAP estimator (14) under the constraint that  $\mathbf{p} \in \mathcal{R}_{\text{red}}^{(\mathcal{X})}$ ; intuitively, since the subset  $\mathcal{R}_{\text{red}}^{(\mathcal{X})}$  is smaller than  $\mathcal{R}^{(\mathcal{X})}$  (2), the computational load required by this step is lower than that required by (13) or (14).

Note that the proposed technique mitigates localization complexity by restricting the search domain  $\mathcal{R}^{(\mathcal{X})}$  on the basis of a distance criterion, in the sense that only a subset of rectangles, close to the raw estimate  $\hat{\mathbf{p}}_{\text{raw}}(\mathbf{z})$ , is taken into consideration. For this reason, in the following, this procedure is dubbed *distance-reduced domain* MMSE (DRD-MMSE) or *distance-reduced domain* (DRD-MAP), if a MMSE or a MAP estimator is employed in the last step, respectively. Note that DRD performance strongly depends on (a) the accuracy of the  $\hat{\mathbf{p}}_{\text{raw}}(\mathbf{z})$  estimate and (b) the criterion adopted in generating the set  $\mathcal{K}$  (17) (if the proposed criterion fails to select the map portion where the agent truly lies, then a large error is made in step (3), where it is assumed that  $\mathbf{p} \in \mathcal{R}_{\text{red}}^{(\mathcal{X})}$ ). In principle, other heuristic criteria (e.g., based on selecting a fixed number of rectangles close to  $\hat{\mathbf{p}}_{\text{raw}}$ ) could be adopted; however, (17) turned out to provide the best accuracy/complexity trade-off among the “distance-based” criteria we tested in our computer simulations.

**3.4. Probability-Reduced Domain Map-Aware Estimator.** An alternative to the DRD approach is based on a sort of *probabilistic criterion* and consists of the following two steps:

- (1) *Map-Aware Raw Estimation.* Similarly to step (1) of the DRD algorithm, a portion  $\mathcal{R}_{\text{red}}^{(\mathcal{X})}$  (16) of  $\mathcal{R}$  is extracted. However, in this case, the integer set  $\mathcal{K}$  is generated by taking the values of the index  $k$  associated with the  $N_{r,\text{min}}$  largest elements of the set

$$\mathcal{F} \triangleq \{\ln f(\mathbf{z} | \tilde{\mathbf{p}} = \mathbf{p}_k^c), \forall k \in \mathcal{K} \cap \mathcal{N}\}, \tag{18}$$

where  $N_{r,\text{min}}$  is a fixed parameter,  $\mathbf{p}_k^c \triangleq \text{centre}\{\mathcal{R}_k\}$ ,  $f(\mathbf{z} | \tilde{\mathbf{p}})$  is given by (5),  $\mathcal{K}$  is defined by (3), and  $\mathcal{N}$  is a proper integer set, whose target is reducing the cardinality of  $\mathcal{F}$  (and thus the overall computational load of the estimation algorithm). In practice,  $\mathcal{N} \triangleq \{k \mid \|\mathbf{p}_k^c - \mathbf{p}_i^c\| > d_{\text{th}}, \forall i \in \mathcal{N} \setminus \{k\}\}$  has been selected, where  $d_{\text{th}}$  is a fixed real and positive parameter, so that only rectangles spaced by at least  $d_{\text{th}}$  meters are considered (instead of all rectangles forming the search domain  $\mathcal{R}^{(\mathcal{X})}$  (2)). The effect of the choice expressed by (18) is to generate the reduced domain  $\mathcal{R}_{\text{red}}^{(\mathcal{X})}$  (16) by selecting the  $N_{r,\text{min}}$  rectangles that, on the basis of their centers, exhibit the largest likelihood values.

- (2) *Map-Aware Estimation.* The final estimate  $\hat{\mathbf{p}}_{\text{PRD}}(\mathbf{z})$  is evaluated exploiting either the MMSE (13) or the MAP estimator (14) under the constraint that  $\mathbf{p} \in \mathcal{R}_{\text{red}}^{(\mathcal{X})}$ .

The resulting estimation technique is dubbed *probability-reduced domain* MMSE (PRD-MMSE) or *probability-reduced*

domain MAP (PRD-MAP), if a MMSE or a MAP estimator, respectively, is employed in the last step of the procedure illustrated above. It is important to point out that this approach offers some advantages with respect to its DRD counterpart, since (a) the criterion (18) is more closely related to the optimal MAP criterion than (17) and (b) all the PRD steps rely on map-awareness, that is, on the most accurate modelling we propose. However, it should be also taken into account the fact that the first step of the PRD approach may entail a significant complexity, since the evaluation of a map-aware likelihood is required.

#### 4. Implementation of Estimation Algorithms

All the map-aware and map-unaware estimation algorithms illustrated in Section 3 require numerical integration and/or the application of optimization methods and involve nonlinear and nondifferentiable cost functions. For these reasons, their implementation raises various problems; some possible solutions are illustrated below.

**4.1. Map-Aware Implementations.** In this section, two different implementations for the evaluation of the MMSE estimate on the basis of (13) are proposed and analysed in detail. The first implementation, called MMSE<sub>c</sub>, is based on the use of the so-called *cubature formulas* (e.g., see [17, 29–31]), which approximate any multidimensional definite integral as a finite sum of terms depending on its integrand. In particular, in the following, any integration over the search domain  $\mathcal{R}^{(\mathcal{Z})}$  is expressed as the sum of  $N_r^{\text{SRC}}$  distinct integrals, each referring to a distinct rectangular domain (see (1)), and the cubature formulas illustrated in [30] are exploited for the evaluation of each of the  $N_r^{\text{SRC}}$  integrals. Then, from (13) the approximate expression

$$\widehat{\mathbf{P}}_{\text{MMSE}_c}(\mathbf{z}) \approx \frac{\left[ \sum_{k=1}^{N_r^{\text{SRC}}} \sum_{j=1}^{N_{n,k}} w_{k,j} \alpha_{(k,j)}^{(x)} h(\mathbf{z}, \alpha_{(k,j)}) \right] \left[ \sum_{k=1}^{N_r^{\text{SRC}}} \sum_{j=1}^{N_{n,k}} w_{k,j} \alpha_{(k,j)}^{(y)} h(\mathbf{z}, \alpha_{(k,j)}) \right]}{\sum_{k=1}^{N_r^{\text{SRC}}} \sum_{j=1}^{N_{n,k}} w_{k,j} h(\mathbf{z}, \alpha_{(k,j)})} \quad (19)$$

is easily inferred, where  $\alpha_{(k,j)} = [\alpha_{(k,j)}^{(x)}, \alpha_{(k,j)}^{(y)}]^T \triangleq \mathbf{c}_k + \mathbf{L}_k \alpha_{(k,j)}^{(F)}$  is the  $j$ th *node* referring to the  $k$ th rectangle  $\mathcal{R}_k \triangleq [l_k; r_k] \times [b_k; t_k]$  in  $\mathcal{R}^{(\mathcal{Z})}$  (2),  $w_{k,j} \triangleq w_{k,j}^{(F)} \det(\mathbf{L}_k)$  is the corresponding *weight*,  $\mathbf{c}_k \triangleq (1/2)[l_k + r_k; b_k + t_k]^T$ , and  $\mathbf{L}_k \triangleq (1/2)\text{diag}\{r_k - l_k, t_k - b_k\}$ , whereas  $\alpha_{(k,j)}^{(F)} \in [-1; 1] \times [-1; 1]$  and  $w_{k,j}^{(F)}$  denote the  $j$ th *node* and the  $j$ th *weight*, respectively, of the cubature formula of order  $N_{n,k}$ . It is worth mentioning that, in principle, the larger the size of  $\mathcal{R}_k$  is, the stronger the fluctuations of the integrand function to be expected over this domain are; consequently, high cubature orders (i.e., large values of  $\{N_{n,k}\}$ ) should be selected for large rectangles, at the price, however, of an increase in the computational load. In our computer simulations, the heuristic formula  $N_{n,k} = N_n(\lfloor m \mathcal{A}_{\mathcal{R}_k} + d_{\min} \rfloor)$  has been adopted for selecting the number of nodes; here,  $N_n(\cdot)$  denotes the function defined in [30] to map the degree  $d$  of a polynomial into the number of

integration nodes,  $\lfloor m \mathcal{A}_{\mathcal{R}_k} + d_{\min} \rfloor$  is the polynomial degree heuristically associated with the (nonpolynomial) integrand functions appearing in (13),  $\mathcal{A}_{\mathcal{R}_k} = \det(\mathbf{L}_k)$  is the area of  $\mathcal{R}_k$ , whereas  $m$  and  $d_{\min}$  are real parameters whose values are listed in Table 1. It is also important to point out that (a) all the sums appearing in (19) require the evaluation of the same quantities  $\{h(\mathbf{z}, \alpha^{(k,j)})\}$  and (b) MMSE<sub>c</sub> is noniterative and its approximate complexity is  $\mathcal{O}(\overline{N}_s \cdot N_{\text{obs}} \cdot N_{\text{eval}})$ , where  $N_{\text{eval}} = \overline{N}_n^x \cdot \overline{N}_n^y \cdot N_r^{\text{SRC}}$  and  $\overline{N}_n^x$  ( $\overline{N}_n^y$ ) denotes the average number of nodes selected for integrating over each of the  $N_r^{\text{SRC}}$  rectangles. Further details about our implementation of MMSE<sub>c</sub> can be found in [27, 32].

The second implementation, called MMSE<sub>r</sub>, is based on the well-known *trapezoidal* integration rule; then, from (13), the approximate formula

$$\widehat{\mathbf{P}}_{\text{MMSE}_r}(\mathbf{z}) \approx \frac{\left[ \sum_{k=1}^{N_r^{\text{SRC}}} \sum_{i=1}^{N_{n,k}^x} \sum_{j=1}^{N_{n,k}^y} w_{k,i,j} \alpha_x^{(k,i,j)} h(\mathbf{z}, \alpha^{(k,i,j)}) \right] \left[ \sum_{k=1}^{N_r^{\text{SRC}}} \sum_{i=1}^{N_{n,k}^x} \sum_{j=1}^{N_{n,k}^y} w_{k,i,j} \alpha_y^{(k,i,j)} h(\mathbf{z}, \alpha^{(k,i,j)}) \right]}{\sum_{k=1}^{N_r^{\text{SRC}}} \sum_{i=1}^{N_{n,k}^x} \sum_{j=1}^{N_{n,k}^y} w_{k,i,j} h(\mathbf{z}, \alpha^{(k,i,j)})} \quad (20)$$

is easily inferred. Here, the *nodes*  $\{\alpha^{(k,i,j)} \triangleq [l_k + (i-1)\Delta, b_k + (j-1)\Delta]^T, i = 1, 2, \dots, N_{n,k}^x, j = 1, 2, \dots, N_{n,k}^y\}$  of the  $k$ th integration formula correspond to the vertices of a regular grid extending over  $\mathcal{R}_k$  and characterized by a spacing  $\Delta$  (see Table 1), whereas the corresponding weights  $\{w_{k,i,j}\}$  are evaluated as

$$w_{k,i,j} \triangleq \begin{cases} \frac{\Delta^2}{4} & \text{for } i \in \{1, N_{n,k}^x\}, j \in \{1, N_{n,k}^y\} \\ \frac{\Delta^2}{2} & \text{for } i \in \{1, N_{n,k}^x\}, j \in \{2, \dots, N_{n,k}^y - 1\} \\ \frac{\Delta^2}{2} & \text{for } i \in \{2, \dots, N_{n,k}^x - 1\}, j \in \{1, N_{n,k}^y\} \\ \Delta^2 & \text{for } i \in \{2, \dots, N_{n,k}^x - 1\}, j \in \{2, \dots, N_{n,k}^y - 1\} \end{cases} \quad (21)$$

with  $N_{n,k}^x \triangleq \lfloor (r_k - l_k)/\Delta \rfloor + 1$  and  $N_{n,k}^y \triangleq \lfloor (t_k - b_k)/\Delta \rfloor + 1$ . It is important to point out the following: (a) similarly to (19), the three sums in (20) require the evaluation of the same quantities  $\{h(\mathbf{z}, \alpha^{(k,i,j)})\}$ ; (b) MMSE<sub>r</sub> is noniterative and its approximate complexity is  $\mathcal{O}(\overline{N}_s \cdot N_{\text{obs}} \cdot N_{\text{eval}})$ , where  $N_{\text{eval}} = \overline{N}_n^x \cdot \overline{N}_n^y \cdot N_r^{\text{SRC}}$  and  $\overline{N}_n^x$  ( $\overline{N}_n^y$ ) denotes the average number of integration points selected along the  $x$  ( $y$ ) direction.

In implementing the MAP estimator (14), the search over the space  $\mathcal{R}^{(\mathcal{Z})}$  has been turned into  $N_r$  searches over the distinct rectangles  $\{\mathcal{R}_k, k = 1, 2, \dots, N_r\}$  covering  $\mathcal{R}^{(\mathcal{Z})}$ . This choice is motivated by the fact that (a) the constraint  $\tilde{\mathbf{p}} \in \mathcal{R}^{(\mathcal{Z})}$ , unlike  $\tilde{\mathbf{p}} \in \mathcal{R}_k$ , cannot be directly formulated as a set of linear inequalities, as required by standard optimization techniques; (b) solving  $N_r$  distinct optimization problems, each involving a search domain much smaller than  $\mathcal{R}^{(\mathcal{Z})}$ , strongly reduces the probability of reaching local minima; (c) when iterative optimization algorithms are adopted over

TABLE 1: Covering and specific values selected for the implementations illustrated in Section 4.

Algorithm	Cov.	Parameters
MMSE <sub>c</sub>	MNC	$m = 1.4 \text{ m}^{-2}, d_{\min} = 5$
MMSE <sub>r</sub>	MNC	$\Delta = 0.3 \text{ m}$
MAP <sub>f</sub>	MNC	Algorithm = active-set, maxiter = $10^3$ , tolfun = tolx = $10^{-2}$
MAP <sub>g</sub>	MNC	$\Delta_1 = 2 \text{ m}, \Delta_{\text{final}} = 0.3 \text{ m}, N_{r,\min} = 3$
MAP <sub>c</sub>	MNC	$\Delta_1 = 2 \text{ m}, \Delta_{\text{final}} = 0.3 \text{ m}$
DRD-E <sub>c</sub> , DRD-P <sub>g</sub>	MNC	$\gamma = 0.6, N_{r,\min} = 5$
PRD-E <sub>c</sub> , PRD-P <sub>g</sub>	MNC	$d_{\text{th}} = 3.5 \text{ m}, N_{r,\min} = 5$
PRD-P <sub>p</sub>	DNC	$d_{\text{th}} = 3.5 \text{ m}, N_{r,\min} = 10$
ML <sub>f</sub>	—	The same as for MAP <sub>f</sub>
ML <sub>m</sub>	—	$\Delta = 10 \text{ m}$ ; others the same as for MAP <sub>f</sub>
ML <sub>c</sub>	—	$\Delta_1 = 2 \text{ m}, \Delta_{\text{final}} = 0.3 \text{ m}$

each of the rectangles  $\{\mathcal{R}_k\}$ , fast convergence can be usually achieved if their centres are selected as initial points. Note that in this case direct-search methods [19, 20] should be preferred to standard Newton-based or gradient-based methods, since the first class of methods usually admits simple implementations for constrained problems and does not require smooth cost functions. These considerations have led us to develop 3 different implementations, called MAP<sub>f</sub>, MAP<sub>g</sub>, and MAP<sub>c</sub>, of the MAP strategy (14). The first one (namely, MAP<sub>f</sub>) is summarised in Algorithm 1 and is based on the use of the MATLAB `fmincon` routine to solve the optimization problem appearing in step (2). This implementation is characterized by various adjustable parameters; their values, which are listed in Table 1, have been selected to minimise the overall computational complexity.

The second implementation (namely, MAP<sub>g</sub>) is summarised in Algorithm 2 and is based on discretisation of the search space and on the use of iterative direct-search methods (see [19, Sec. 3]). In practice, in its  $k$ th iteration, for each rectangle of the selected covering, the cost function  $f_c(\cdot)$  (15) is evaluated at the vertices of a rectangular grid characterized by spacing  $\Delta_k$  (see lines (4)-(5)); then, on the basis of the resulting values, only the most promising rectangles are saved for the next iteration (see line (8)), in which a halved step size is adopted (see line (9)). Note that both the computational complexity and the accuracy of MAP<sub>g</sub> depend on the initial and the final values of the step size (denoted by  $\Delta_1$  and  $\Delta_{\text{final}}$ , resp.), so that a proper trade-off has to be achieved in the selection of these parameters (the values we adopted are listed in Table 1).

The third implementation (namely, MAP<sub>c</sub>) is summarised in Algorithm 3 and is based on a different direct-search method, known as the *compass algorithm* [20, Sec. 8.1]. Similarly to MAP<sub>g</sub>, MAP<sub>c</sub> exploits a discretisation of the search space and an iterative reduction of the step size; however, the grid it employs is not regular. In fact, in each rectangle, the domain is explored moving from a “current trial position”  $\tilde{\mathbf{p}}$  on the basis of given displacement vectors  $\pm \mathbf{e}_x$  or  $\pm \mathbf{e}_y$  (see line (6)) in order to identify the direction along which the cost function decreases (see line (8)). Even in this case, the overall computational complexity and the

**Require:** covering  $\{\mathcal{R}_k, k \in \mathcal{K}\}$  of  $\mathcal{R}^{(X)}$  (2);  $f_c(\mathbf{p})$  (15).  
(1) **for**  $k \in \mathcal{K}$  **do**  
(2)  $\mathbf{p}_{\min,k} \leftarrow \arg \min_{\tilde{\mathbf{p}} \in \mathcal{R}_k} f_c(\tilde{\mathbf{p}})$   
(3)  $f_{\min,k} \leftarrow f_c(\mathbf{p}_{\min,k})$   
(4) **end for**  
(5)  $\hat{k} \leftarrow \arg \min_k \{f_{\min,k}\}, \hat{\mathbf{p}}_{\text{MAP}_f} \leftarrow \mathbf{p}_{\min,\hat{k}}$

ALGORITHM 1: Implementation of the MAP strategy (14) based on the `fmincon` routine (MAP<sub>f</sub>).

accuracy of this implementation depend on the initial and final values of the step size (see Table 1), which need to be carefully selected.

It is not difficult to show that the overall computational complexity of each of the MAP implementations proposed above is  $\mathcal{O}(\bar{N}_s \cdot N_{\text{obs}} \cdot N_{\text{eval}})$ , where  $N_{\text{eval}}$  denotes the overall number of times the cost function  $f_c(\cdot)$  (15) is evaluated. Unluckily,  $N_{\text{eval}}$  cannot be easily related to the other parameters of the proposed MAP implementations because of its nonlinear dependence on such parameters; consequently, a more accurate estimate of the computational burden required by the proposed implementations cannot be provided.

Let us focus now on the problem of combining the implementation of the MMSE and MAP estimators illustrated above with the DRD and PRD techniques described in Section 3.3. As far as the DRD technique is concerned, we note that all the proposed MAP/MMSE implementations can be employed in its last step. In the following, however, the implementations of DRD-MMSE and DRD-MAP techniques are always based on MMSE<sub>c</sub> and MAP<sub>g</sub>, respectively; for this reason, they are denoted by DRD-E<sub>c</sub> and DRD-P<sub>g</sub>, respectively. It is important to point out the following:

- (1) The algorithms DRD-E<sub>c</sub> and DRD-P<sub>g</sub> are not intrinsically iterative. However, they can be easily modified to include a mechanism for iteratively updating the value of the parameter  $\gamma$  in order to improve their robustness against inaccuracies in the initial setup (see Algorithm 4).



**Require:** covering  $\{\mathcal{R}_k = [l_k, r_k] \times [b_k, t_k], k \in \mathcal{K}\}$  of  $\mathcal{R}^{(\mathcal{Z})}$  (2);  $f_c(\mathbf{p})$  (15);  $\Delta_1, \Delta_{\text{final}}$  and  $N_{r,\text{min}}$  (Table 1).

- (1)  $\Delta \leftarrow \Delta_1, \widetilde{\mathcal{K}} \leftarrow \mathcal{K}$
- (2) **while**  $\Delta > \Delta_{\text{final}}$  **do**
- (3) **for**  $k \in \widetilde{\mathcal{K}}$  **do**
- (4)  $\mathcal{I} \leftarrow \{1, \dots, \lfloor (r_k - l_k)/\Delta \rfloor\}, \mathcal{J} \leftarrow \{1, \dots, \lfloor (t_k - b_k)/\Delta \rfloor\}$
- (5)  $(\hat{i}, \hat{j}) \leftarrow \arg \min_{i \in \mathcal{I}, j \in \mathcal{J}} f_c((l_k, b_k) + \Delta(i, j))$
- (6)  $\mathbf{p}_{\text{min},k} \leftarrow (l_k, b_k) + \Delta(\hat{i}, \hat{j}), f_{\text{min},k} \leftarrow f_c(\mathbf{p}_{\text{min},k})$
- (7) **end for**
- (8)  $\widetilde{\mathcal{K}} \leftarrow$  indices of the  $N_{r,\text{min}}$  smallest elements of  $\{f_{\text{min},k}\}$
- (9)  $\Delta \leftarrow \Delta/2$
- (10) **end while**
- (11)  $\hat{k} \leftarrow \arg \min_{k \in \widetilde{\mathcal{K}}} \{f_{\text{min},k}\}, \widehat{\mathbf{p}}_{\text{MAP}_g} \leftarrow \mathbf{p}_{\text{min},\hat{k}}$

ALGORITHM 2: Implementation of the MAP strategy (14) based on an iterative grid search (MAP<sub>g</sub>).

**Require:** covering  $\{\mathcal{R}_k, k \in \mathcal{K}\}$  of  $\mathcal{R}^{(\mathcal{Z})}$  (2);  $f_c(\mathbf{p})$  (15);  $\Delta_1$  and  $\Delta_{\text{final}}$  (Table 1).

- (1) **for**  $k \in \mathcal{K}$  **do**
- (2)  $\Delta \leftarrow \Delta_1$
- (3)  $\mathbf{p}_{\text{min},k} \leftarrow \text{centre}\{\mathcal{R}_k\}, f_{\text{min},k} \leftarrow f_c(\mathbf{p}_{\text{min},k})$
- (4) **while**  $\Delta > \Delta_{\text{final}}$  **do**
- (5)  $\text{bImproved} \leftarrow \text{False}$
- (6) **for**  $\mathbf{d} \in \{\mathbf{e}_x, \mathbf{e}_y, -\mathbf{e}_x, -\mathbf{e}_y\}$  **do**
- (7)  $\tilde{\mathbf{p}} \leftarrow \mathbf{p}_{\text{min},k} + \Delta \cdot \mathbf{d}$
- (8) **if**  $f_c(\tilde{\mathbf{p}}) < f_{\text{min},k}$  and  $\tilde{\mathbf{p}} \in \mathcal{R}_k$  **then**
- (9)  $f_{\text{min},k} \leftarrow f_c(\tilde{\mathbf{p}}), \mathbf{p}_{\text{min},k} \leftarrow \tilde{\mathbf{p}}$
- (10)  $\text{bImproved} \leftarrow \text{True}, \text{break}$
- (11) **end if**
- (12) **end for**
- (13)  $\Delta \leftarrow \Delta/2$  only if  $\text{bImproved} = \text{False}$
- (14) **end while**
- (15) **end for**
- (16)  $\hat{k} \leftarrow \arg \min_k \{f_{\text{min},k}\}, \widehat{\mathbf{p}}_{\text{MAP}_c} \leftarrow \mathbf{p}_{\text{min},\hat{k}}$

ALGORITHM 3: Implementation of the MAP strategy (14) based on an iterative compass search (MAP<sub>c</sub>).

**Require:** covering  $\{\mathcal{R}_k, k \in \mathcal{K}\}$  of  $\mathcal{R}^{(\mathcal{Z})}$  (2);  $\widehat{\mathbf{p}}_{\text{MMSE}_c}$  ( $\widehat{\mathbf{p}}_{\text{MAP}_g}$ ) estimator; constants  $\gamma$  and  $N_{r,\text{min}}$  (Table 1).

- (1)  $\widehat{\mathbf{p}}_{\text{raw}} \leftarrow \widehat{\mathbf{p}}_{\text{ML}_m}$ , computed assuming  $\widehat{\mathcal{Z}}^{\text{NLOS}} = \mathcal{Z}$
- (2)  $d_{\text{th}} \leftarrow \text{median}_{i \in \mathcal{Z}} \{\gamma \sqrt{\sigma_{b,i}^2(\widehat{\mathbf{p}}_{\text{raw}}(\mathbf{z})) + \sigma_{n,i}^2(\widehat{\mathbf{p}}_{\text{raw}}(\mathbf{z}))}\}$
- (3)  $d_k \leftarrow \|\widehat{\mathbf{p}}_{\text{raw}} - \text{proj}\{\mathcal{R}_k, \widehat{\mathbf{p}}_{\text{raw}}\}\|$  for  $k \in \mathcal{K}$  (see (17))
- (4) **while**  $|\widetilde{\mathcal{K}}| < N_{r,\text{min}}$  **do**
- (5)  $\widetilde{\mathcal{K}} \leftarrow \{k \mid d_k < d_{\text{th}}\}, d_{\text{th}} \leftarrow 1.5d_{\text{th}}$
- (6) **end while**
- (7)  $\widehat{\mathbf{p}}_{\text{DRD-E}_c} \leftarrow \widehat{\mathbf{p}}_{\text{MMSE}_c}(\widehat{\mathbf{p}}_{\text{DRD-P}_g} \leftarrow \widehat{\mathbf{p}}_{\text{MAP}_g})$  evaluated in the reduced map  $\mathcal{R}_{\text{red}}^{(\mathcal{Z})} = \bigcup_{k \in \widetilde{\mathcal{K}}} \mathcal{R}_k$

ALGORITHM 4: Implementation of the iterative DRD estimator (see Section 3.3) employed by DRD-E<sub>c</sub> (DRD-P<sub>g</sub>).

- (2) The complexity of DRD- $E_c$  and DRD- $P_g$  is approximately  $\mathcal{O}(N_r^{\text{SRC}} + N_{\text{obs}}(\bar{N}_s \cdot N_{\text{eval}} + N_{\text{eval,raw}}))$ , where  $N_{\text{eval}}$  denotes the overall number of evaluations of the MMSE integrand (for DRD- $E_c$ ) or that of the MAP cost function (for DRD- $P_g$ ), and  $N_{\text{eval,raw}}$  represents the overall number of evaluations of the ML cost function.

In our work, MMSE $_c$  and MAP $_g$  have been also selected for the last step of the PRD technique; the resulting implementations are denoted by PRD- $E_c$  and PRD- $P_g$ , respectively, in the following. It is worth mentioning that

- (1) both PRD- $E_c$  and PRD- $P_g$  are not iterative;
- (2) their complexity is  $\mathcal{O}(\bar{N}_s \cdot N_{\text{obs}} \cdot (N_{\text{eval,1}} + N_{\text{eval,2}}))$ , where  $N_{\text{eval,1}}$  and  $N_{\text{eval,2}}$  denote the overall number of evaluations of the MMSE integrand (for PRD- $E_c$ ) and that of the MAP cost function (for PRD- $P_g$ ) in the first step and in the second step, respectively (see Section 3.4);
- (3) the values of parameters  $d_{\text{th}}$  and  $N_{r,\text{min}}$  need to be carefully selected.

Our numerical results (see Section 5) have evidenced that whatever option for map-aware estimation is selected, the parameter  $\bar{N}_s$  (i.e., the average number of intersection tests accomplished in processing each observation) appearing in the complexity estimate of all the implementations developed in this paragraph plays a significant role in penalizing them with respect to their map-unaware counterparts. To reduce  $\bar{N}_s$ , the following two techniques have been devised:

- (1) *Segment Partitioning* (SP). This consists of an offline phase, where (a)  $\mathcal{R}$  is partitioned in multiple cells using a nonuniform rectangular grid (in a way that each anchor lies on a vertex of the grid) and (b) the obstruction segments falling inside each cell are identified. Then, when  $N_o(\mathbf{p}_i^a, \mathbf{p}_2)$  is evaluated, only the segments of the cells intersecting with the  $(\mathbf{p}_i^a, \mathbf{p}_2)$  segment are taken into account, thus reducing  $\bar{N}_s$ .
- (2) *Precomputing* (PC). This is based on an offline evaluation of the function  $N_o(\cdot, \cdot)$  over  $N_{\text{pt}}$  points (uniformly) distributed inside each rectangle of the available MNC or DNC; this requires storing  $N_{\text{pt}} \cdot \bar{N}_a \cdot N_r^{\text{MNC}}$  or  $N_{\text{pt}} \cdot \bar{N}_a \cdot N_r^{\text{DNC}}$  integers in a table, where  $\bar{N}_a$  denotes the average number of connected anchors. This technique is “memory hungry” for large maps; however, the precomputed table replaces that memorising information about the position of  $N_s$  obstruction segments and can be stored efficiently using a limited number of bits for the integer representation of the precomputed values of  $N_o(\cdot, \cdot)$ . Moreover, this memory requirement is often more than counterbalanced by the reduction of computational complexity (since  $\bar{N}_s = 0$  in this case).

In our computer simulations, the first technique has been adopted for all the map-aware estimators. The second technique, instead, has been used in the first step of PRD-MAP

estimation only; the resulting implementation is dubbed PRD- $P_p$ . This choice is motivated by the fact that this estimator can be easily adapted to the use of precomputed quantities, since it evaluates the map-aware likelihood (8) at a set of fixed points  $\{\mathbf{p}_k^c\}$  (see Section 3.4).

**4.2. Map-Unaware Implementations.** Before illustrating our implementations of the ML estimator in (9), it is important to mention that if the constraint  $\tilde{\mathbf{p}} \in \mathcal{P}(\mathcal{Z})$  was neglected, standard algorithms (e.g., the *Nelder-Mead simplex method* [33]) could be employed to solve the optimization problem it involves. Unluckily, such a constraint plays an important role, specially in large maps, since (a) minima different from the global one are substantially less likely to be contained in the domain  $\mathcal{P}(\mathcal{Z})$  than in the whole  $\mathbb{R}^2$  space and (b) the map-unaware likelihood (8) is not defined for  $\tilde{\mathbf{p}} \notin \mathcal{P}(\mathcal{Z})$ , as already explained in [15, Sec. II.C.]. For these reasons, in the 3 different implementations of the ML estimator (9) proposed below, the constraint  $\tilde{\mathbf{p}} \in \mathcal{P}(\mathcal{Z})$  is always accounted for. The first implementation, called ML $_f$ , relies on the MATLAB `fmincon` routine, which is employed to solve the optimisation problem of (9). The second one, dubbed ML $_m$ , is based on partitioning  $\mathcal{P}(\mathcal{Z})$  into  $N_{\text{sub}}$  subrectangles (whose sides have a length not exceeding a threshold  $\Delta$ ) and on running the MATLAB `fmincon` routine over each of the subrectangles; in this case, the final estimate is selected among  $N_{\text{sub}}$  candidates according to a minimum-cost criterion (note that a similar approach has been adopted in Algorithm 1). Finally, the third implementation, called ML $_c$ , employs the *compass algorithm* (see Algorithm 3) under the assumption that  $N_r = 1$  and  $\mathcal{R}_1 \equiv \mathcal{P}(\mathcal{Z})$ . Note that the complexity of each of the proposed implementations is  $\mathcal{O}(N_{\text{obs}} \cdot N_{\text{eval}})$ , where  $N_{\text{eval}}$  is the number of times the ML cost function is evaluated. The values of the parameters selected for each of them are listed in Table 1.

Finally, a diagram summarising all the implementations developed in this section is provided in Figure 2.

## 5. Simulation Results

In this section, the implementations described in Section 4 are compared in terms of accuracy and complexity. In our analysis, the three different maps, shown in Figure 3, are considered. Such maps are characterized by different sizes and obstruction densities (see Table 2), by a specific deployment of anchors (whose positions are identified by green squares in Figure 3), and by a similar number of anchors per unit area (1 anchor each  $\approx 100 \text{ m}^2$ ), so that a fair comparison to the data referring to different maps is ensured. In addition, in all cases, the anchor coverage radius (see Section 2.2) is  $d_{\text{max},i} = d_{\text{max}} = 30 \text{ m}$  (with  $i = 1, 2, \dots, N_a$ ).

The remaining part of this section is organized as follows. In Section 5.1, the method employed to estimate the computational complexity of the proposed implementations is described. In Section 5.2, the structure of the software simulator developed to assess both the accuracy and the complexity of our implementations is illustrated. Finally, in Sections 5.3

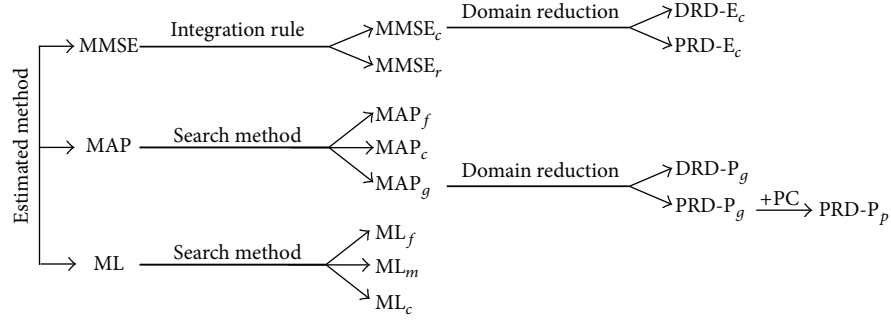


FIGURE 2: Implementations of optimal map-aware and map-unaware estimators proposed in this paper.

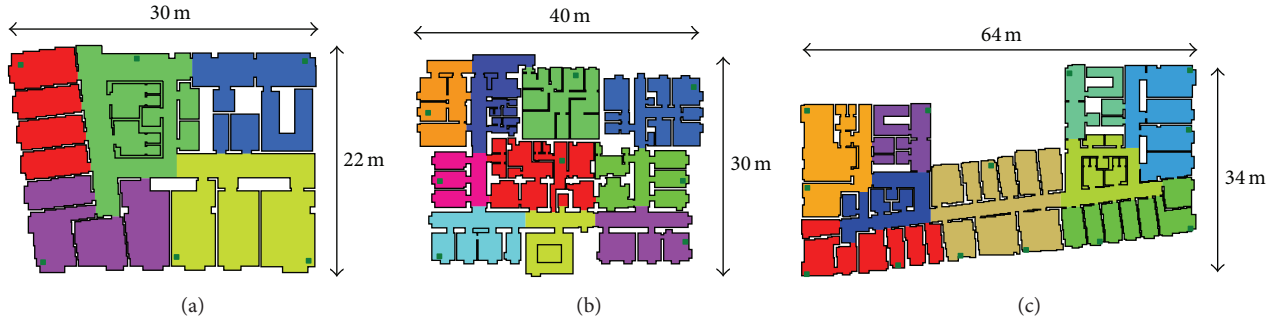


FIGURE 3: Support regions of map #1 (a), map #2 (b), and map #3 (c). Note that different scales have been used in the representation of these maps.

 TABLE 2: Values of some parameters characterizing the maps considered in Section 5 and their MNC/DNC coverings. The number of anchors ( $N_a$ ) and intersection tests ( $N_s$ ) is also provided for each map.

Map	$N_r^{\text{MNC}}$	$N_r^{\text{DNC}}$	$N_s$	$\mathcal{A}_{\mathcal{R}}$	$N_a$
1	340	632	574	540 m <sup>2</sup>	6
2	529	925	1029	800 m <sup>2</sup>	8
3	773	1461	1375	1310 m <sup>2</sup>	13

and 5.4, various numerical results based on statistical models and experimental measurements, respectively, are analysed.

**5.1. Evaluation of the Computational Complexity of Different Implementations.** Map-aware implementations are expected to outperform their map-unaware counterparts at the price of a larger computational complexity. However, in this case, an accurate assessment of computational complexity is a challenging task, mainly because of the adaptive and data-dependent nature of the proposed algorithms; for instance, the term  $N_{\text{eval}}$  (appearing in all the formulas summarised in Table 3) cannot be easily related to the size of the search domain or other implementation-specific parameters. For this reason, the computational complexity of each implementation has been assessed by estimating the average amount  $\bar{F}$  of FLOPs required in our computer simulations; this task has been accomplished under the following assumptions:

TABLE 3: Computational complexity of the implementations proposed in Section 4 for the ML, MAP, and MMSE estimators.

Algorithm	Complexity
MMSE <sub>c</sub>	$\mathcal{O}(\bar{N}_s \cdot N_{\text{obs}} \cdot N_{\text{eval}})$
MMSE <sub>r</sub>	$\mathcal{O}(\bar{N}_s \cdot N_{\text{obs}} \cdot N_{\text{eval}})$
MAP <sub>f</sub> , MAP <sub>g</sub> , MAP <sub>c</sub>	$\mathcal{O}(\bar{N}_s \cdot N_{\text{obs}} \cdot N_{\text{eval}})$
DRD-E <sub>c</sub> , DRD-P <sub>g</sub>	$\mathcal{O}(N_r^{\text{SRC}} + N_{\text{obs}}(\bar{N}_s \cdot N_{\text{eval}} + N_{\text{eval,raw}}))$
PRD-E <sub>c</sub> , PRD-P <sub>g</sub>	$\mathcal{O}(\bar{N}_s \cdot N_{\text{obs}} \cdot (N_{\text{eval,1}} + N_{\text{eval,2}}))$
ML <sub>f</sub> , ML <sub>m</sub> , ML <sub>c</sub>	$\mathcal{O}(N_{\text{obs}} \cdot N_{\text{eval}})$

- (1) Basic algorithmic steps (e.g., basic checks and assignments in Algorithms 1–4 for map-aware estimation and LOS/NLOS detection in map-unaware estimation) play a negligible role.
- (2) The computational complexity of each implementation mainly depends on the average number  $\bar{N}_{\text{eval}}$  of evaluations of integrands (in MMSE estimators) or of cost functions (in MAP and ML estimators); as already pointed out in Section 4, the complexity of such evaluations depends, in turn, on  $\bar{N}_s$  and  $\bar{N}_{\text{obs}}$ . For this reason, in our simulations, counters for  $N_{\text{eval}}$ ,  $N_s$ , and  $N_{\text{obs}}$  have been used in order to estimate  $\bar{N}_{\text{eval}}$ ,  $\bar{N}_s$ , and  $\bar{N}_{\text{obs}}$ , respectively (see Table 3).

- (3) Since distinct types of operations are characterized by different computational weights, the expression  $\bar{F} = \bar{N}_{\text{eval}} \sum_t w_t n_t$  has been adopted, where  $n_t$  is the number of FLOPs of type  $t$  performed in each evaluation of integrands or cost function and  $w_t$  is the associated computational weight. In particular, in our simulations, the weights  $\{3, 3, 3, 2, 2, 1, 0\}$  have been assigned to square roots, exponentials, logarithms, divisions, multiplications, additions, and comparisons, respectively.

Our approach is expected to provide statistically meaningful results about  $\bar{F}$  only if a large number (in practice, at least  $10^4$ ) of statistically independent measurements are available. Acquiring such a large number of experimental data would require a burdensome experimental campaign involving at least 150 measurement sites (so that  $\binom{150}{2} \approx 10^4$  independent links would be available). To circumvent this problem, the computational complexity of the devised algorithms has been assessed mainly through computer simulations, following the approach always adopted in related work (e.g., see [22, 23]). However, some numerical results based on the available measurements are also shown (see Section 5.4) to support our conclusions.

**5.2. Description of the Developed Simulator.** A software simulator has been developed (the MATLAB code used to generate the numerical results is publicly available (together with other results not included in this paper for space limitations) at <http://frm.users.sourceforge.net/publications.html>, in accordance with the philosophy of reproducible research standard [34]) to test the proposed implementations in the harsh propagation conditions characterizing typical indoor environments [15]. In each simulation run, the following steps are accomplished sequentially:

- (1) *Generation of the Agent Position.* The true agent position  $\mathbf{p}$  is drawn from the selected map according to the rule  $\mathbf{p} \sim f(\mathbf{p})$ , where  $f(\mathbf{p}) = 1/\mathcal{A}_{\mathcal{R}}$  for  $\mathbf{p} \in \mathcal{R}$  and  $f(\mathbf{p}) = 0$  elsewhere (in other words, the *uniform statistical map model* of [15] has been adopted).
- (2) *Selection of the Connected Anchors.* The  $i$ th anchor is deemed to be “connected” with the agent only if the following conditions are jointly satisfied: (a)  $N_o(\mathbf{p}, \mathbf{p}_i^a) < N_{w,\max} \triangleq 8$ , (b)  $\sigma_{n,i}^2(\mathbf{p}) + \sigma_{b,i}^2(\mathbf{p}) < \sigma_{\max}^2 \triangleq 30 \text{ m}^2$ , and (c)  $d_i(\mathbf{p}) < \bar{d}_{\max} \triangleq 27 \text{ m}$ . In addition, it is assumed that each agent-anchor link is characterized by a “failure probability”  $P_f \triangleq 0.1$  (due, e.g., to synchronization or connection problems between the agent and the anchor itself).
- (3) *Generation of the Noisy Observations.* The noisy observation vector  $\mathbf{z}$  is drawn from the map-aware likelihood (5) according to rule  $\mathbf{z} \sim f(\mathbf{z} | \bar{\mathbf{p}} = \mathbf{p})$ ; the associated sets  $\hat{\mathcal{Z}}^{\text{LOS}}$  and  $\hat{\mathcal{Z}}^{\text{NLOS}}$  are randomly generated assuming the presence of errors with probability  $P_e^{\text{NLOS}} = 0.1$  (to account for the presence of a nonideal LOS/NLOS detector).

- (4) *Position Estimation.* All the estimator implementations discussed in Section 4 are run sequentially; they are all fed by  $\mathbf{z}$  and  $\mathcal{R}^{(\mathcal{X})}$  (2) (map-aware estimators) or  $\mathcal{P}^{(\mathcal{X})}$  (4) (for map-unaware ones). Their output is either a position estimate  $\hat{\mathbf{p}}(\mathbf{z})$  or a failure indication (occurring, e.g., when optimization routines do not converge).

In our simulations, steps (1)–(4) have been repeated for at least  $N_{\text{runs}} = 10^4$  times in order to generate accurate estimates of the RMSE  $\epsilon$  (which is more robust to outliers than root mean square error) and of the mean number of FLOPs  $\bar{F}$  for each implementation.

**5.3. Accuracy and Complexity Comparison.** Figure 4 shows the accuracy (in terms of RMSE  $\epsilon$ ) of the implementations discussed in Section 4 for the 3 considered maps. These results show the following:

- (1) The accuracy achieved by the DRD-MMSE (DRD-MAP) and PRD-MMSE (DRD-MAP) estimator is similar to that of MMSE (MAP) estimators (e.g.,  $\epsilon_{\text{PRD-E}_c} \approx \epsilon_{\text{DRD-E}_c} \approx 1.06\epsilon_{\text{MMSE}_c}$  for map #3).
- (2) The implementations  $\text{MMSE}_c$  and  $\text{MMSE}_r$  provide a 10%–15% performance improvement with respect to  $\text{MAP}_f$ ,  $\text{MAP}_g$ , and  $\text{MAP}_c$  (e.g.,  $\epsilon_{\text{MAP}_g} \approx 1.16\epsilon_{\text{MMSE}_c}$  and  $\epsilon_{\text{MAP}_g} \approx 1.15\epsilon_{\text{MMSE}_r}$  for map #3).
- (3) Map-awareness significantly improves estimation accuracy (up to 250%), as already evidenced in [15] (e.g.,  $\epsilon_{\text{ML}_m} \approx 2.46\epsilon_{\text{MMSE}_c}$  and  $\epsilon_{\text{ML}_m} \approx 2.12\epsilon_{\text{MAP}_g}$  for map #3).

It is important to point out that the estimation accuracy of the proposed estimators is mainly related to (a) the average spatial density of obstruction segments (i.e., to the ratio  $N_s/\mathcal{A}_{\mathcal{R}}$ ) and (b) the average spatial density of anchors (i.e., to the ratio  $N_a/\mathcal{A}_{\mathcal{R}}$ ). Since the propagation scenario is similar in the three maps (light commercial buildings) and a similar anchor density has been adopted for the sake of a fair comparison (see Table 2), the RMSEs provided by the considered estimators are similar over the three maps and do not necessarily increase with  $\mathcal{A}_{\mathcal{R}}$  (e.g., map #3 is larger than map #2, but a lower RMSE is achieved for map #3).

Note also that Figure 4 does not unveil some important aspects of the considered estimation algorithms, which emerge, instead, from the related box plot of Figure 5 describing the distribution of the RMSE outcomes for map #3. Here, the box represents the 75th percentile, the line within the box denotes the median, the black dashed whiskers are upper adjacents, and the cross markers represent the outliers. From Figure 5, the following can be easily inferred:

- (1) The behaviour of all the estimators is characterized by the presence of multiple RMSE outliers associated with  $\epsilon > 8 \text{ m}$  as well as of upper adjacents and values of 75th percentiles extending well beyond their median values. This is due to the statistical model of Section 2, which reproduces harsh propagation conditions degrading RSS measurements in practical

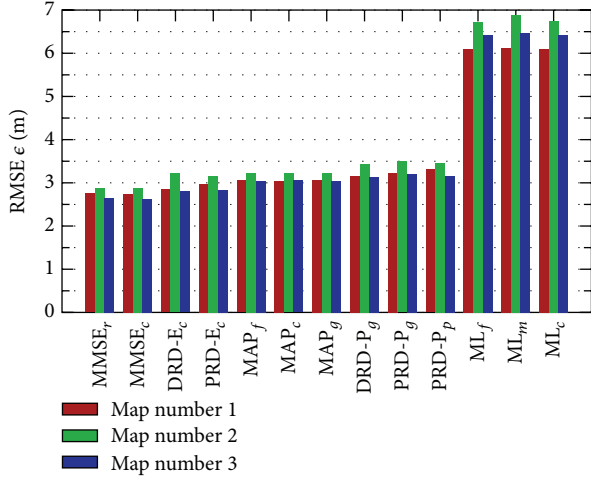


FIGURE 4: RMSE performance provided by various ML, MAP, and MMSE estimation algorithms.

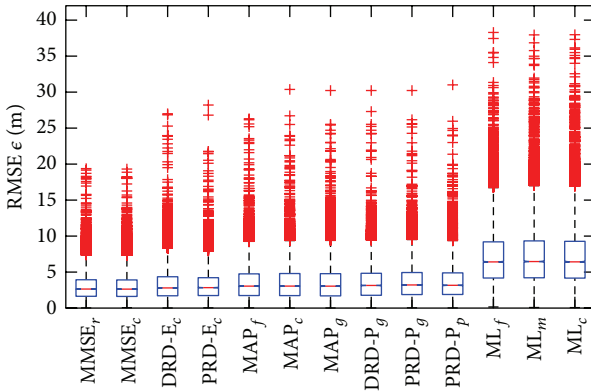


FIGURE 5: Box plot of the RMSE  $\epsilon$  provided by various ML, MAP, and MMSE estimation algorithms. Only map #3 is considered.

systems. Note that nonlinear filtering (e.g., particle filtering) could be employed to reject, at least in part, outliers when tracking moving agents.

- (2) The implementations  $MMSE_c$  and  $MMSE_r$  are characterized by the smallest dispersion of estimation errors (in fact, upper adjacents are at 7.3 m for both).

Some numerical results about the complexity of the estimators implementations are shown in Figure 6 (note that a logarithmic scale is adopted for  $\bar{F}$  since complexities range roughly from  $10^3$  to  $10^9$  FLOPs). These results lead to the following conclusions:

- (1) The MAP and MMSE estimators based on the DRD and PRD techniques are characterized by a computational complexity 4–11 times lower than that of their counterparts not employing such domain reduction techniques, especially for large maps (e.g.,  $\bar{F}_{MAP_g} \approx 3.9\bar{F}_{DRD-P_g}$  and  $\bar{F}_{MAP_g} \approx 11\bar{F}_{PRD-P_p}$  for map #3). This result can be related to the mean ratio between the areas  $\mathcal{R}_{red}^{(\mathcal{X})}$  (16) and  $\mathcal{R}^{(\mathcal{X})}$  (2), that is, to the amount

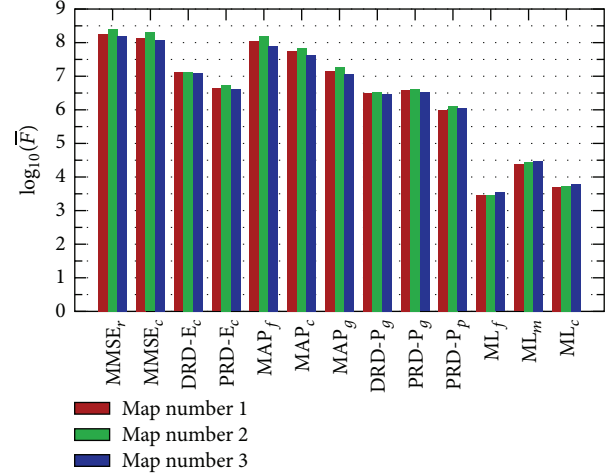


FIGURE 6: FLOP count  $\bar{F}$  referring to various ML, MAP, and MMSE estimation algorithms.

of “domain reduction.” In fact, our simulations have evidenced that, in the case of map #3, on average, the DRD and PRD-based estimators restrict their search domain to 13% and 4% only, respectively, of the whole domain  $\mathcal{R}^{(\mathcal{X})}$ . This makes the complexities of DRD and PRD estimators much closer to that of map-unaware estimators (at the price of a negligible sacrifice in accuracy).

- (2) Even if all map-aware implementations exhibit similar accuracy, they may require significantly different complexities. In fact, the parameter  $\bar{F}$  ranges from  $\bar{F}_{MMSE_r} \approx 1.6 \cdot 10^8$  (for the most demanding map-aware algorithm) to  $\bar{F}_{PRD-P_p} \approx 1.1 \cdot 10^6$  (for the most computationally efficient map-aware algorithm), if map #3 is considered. Moreover, there is a significant gap between the FLOP counts of map-aware estimators and those of their map-unaware counterparts (e.g.,  $\bar{F}_{MMSE_c} \approx 4.1 \cdot 10^3 \bar{F}_{ML_m}$  for map #3).
- (3) Similarly to the RMSE, the parameter  $\bar{F}$  is significantly influenced by the size of  $\mathcal{R}^{(\mathcal{X})}$  (2) and gets larger as the average spatial density of obstruction segments increases.
- (4) PRD-based estimators are less computationally demanding than their DRD counterparts, specially in their MMSE variants (e.g.,  $\bar{F}_{DRD-E_c} \approx 3.1\bar{F}_{PRD-E_c}$  for map #3).

Additional numerical results about the complexity of such algorithms (not shown here for space limitations) can be found in [27, 32].

**5.4. Experimental Results.** The conclusions inferred from Figures 4–6 have been also supported by additional numerical results based on a set of experimental measurements referring to map #2. In fact, in the last part of our work, the simulator described in Section 5.2 has been modified in order to replace

the computer-based generation of the noisy observation vector  $\mathbf{z}$  (see step (3) of the simulator) with a random selection of observations stored in the experimental database described in [15]. Note that such a database, like other experimental databases publicly available, contains a limited set of statistically independent measurements (referring to 116 distinct wireless links) and this unavoidably reduces the overall number of independent simulations that can be run in processing such measurements; for this reason, from a statistical point of view, the results analysed in the previous Sections 5.2 and 5.3 are characterized by a higher level of confidence than those based on experimental measurements (see our comments at the end of Section 5.1). In practice, in this case, the number of simulation runs has been reduced to  $N_{\text{runs}} = 10^3$ . The results shown in Figures 7 and 8 lead to the following conclusions:

- (1) The RMSE achieved by map-aware estimators on the basis of experimental measurements is on the order of 2 m and is about 30% smaller than that of the simulated counterpart; these results can be related to the harsh propagation environment reproduced by the statistical model (5).
- (2) The RMSE provided by map-unaware estimators on the basis of experimental measurements is close to that of its simulated counterpart; the good agreement between experimental and simulated results can be related to the availability of an ideal LOS/NLOS detector (actually, a small probability of error  $P_e^{\text{NLOS}} = 0.1$  was assumed in Section 5.2).
- (3) The computational complexity involved by the considered algorithms (both map-aware and map-unaware) is substantially the same in the case of simulated and experimental data; this means that the shape of the likelihood functions (5) and (8) does not significantly change substituting the experimental data with the observations generated through the statistical simulator.

Finally, the possibility of generalising the proposed approach to other environments deserves some comments. In fact, in this study, three distinct scenarios, characterized by different maps and areas, but by similar propagation conditions, have been taken into consideration. Our measurements have led us to the conclusion that all the measurements acquired in these scenarios can be described by the same statistical model, which is that the same parameters can be exploited by map-aware and map-unaware estimators for each of the three considered environments. This suggests that the methodology adopted in this work can be easily adapted to other buildings characterized by offices, laboratories, and open spaces, like those encountered in light commercial buildings. Nonetheless, it is important to stress that, in principle, indoor environments characterized by different propagation conditions require a dedicated calibration phase for extracting a statistical model for measurements, as already mentioned in Section 2.4. Given such a model, the proposed techniques can be easily applied.

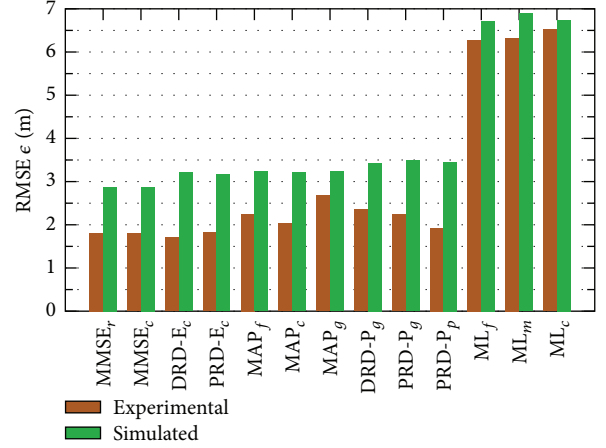


FIGURE 7: RMSE performance provided by various ML, MAP, and MMSE estimation algorithms. These results, referring to map #2, are based on measurements.

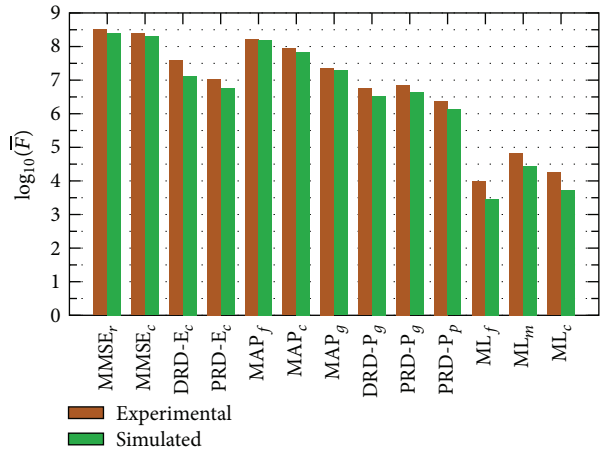


FIGURE 8: FLOP count  $\bar{F}$  for various ML, MAP, and MMSE estimation algorithms. These results, referring to map #2, are based on measurements.

## 6. Conclusions

In this paper, specific solutions to the problem of implementing optimal map-aware and map-unaware estimators for indoor localization have been developed. These solutions are based on state-of-the-art numerical integration and optimization methods as well as novel techniques for reducing the size of search domains. Their accuracy and complexity in terms of RMSE and FLOP count, respectively, have been analysed through computer simulations relying on both statistical models (validated with measurements) and experimental databases. Numerical results have evidenced that (a) map-awareness can appreciably improve RMSE performance of RSS-based localization algorithms; (b) the proposed domain reduction techniques significantly reduce the computational complexity of map-aware algorithms; (c) the best accuracy-complexity trade-off is achieved by the PRD-E<sub>p</sub> implementation, since it achieves nearly optimal RMSE performance and entails a computational burden much smaller than that of all

the other map-aware algorithms in the considered scenarios; (d) experimental results substantially confirm the simulation results in terms of computational complexity, whereas the RMSE achieved over the experimental database is about 30% smaller than that expected by simulations because of the harsh propagation environment reproduced by the adopted statistical model.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## 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 Magazine*, vol. 22, no. 4, pp. 54–69, 2005.
- [2] E. Elnahrawy, X. Li, and R. P. Martin, "The limits of localization using signal strength: a comparative study," in *Proceedings of the 1st Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON '04)*, pp. 406–414, October 2004.
- [3] S.-H. Fang, T.-N. Lin, and K.-C. Lee, "A novel algorithm for multipath fingerprinting in indoor WLAN environments," *IEEE Transactions on Wireless Communications*, vol. 7, no. 9, pp. 3579–3588, 2008.
- [4] M. Bshara, U. Orguner, F. Gustafsson, and L. V. Biesen, "Fingerprinting localization in wireless networks based on received-signal-strength measurements: a case study on WiMAX networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 1, pp. 283–294, 2010.
- [5] K. El-Kafrawy, M. Youssef, A. El-Keyi, and A. Naguib, "Propagation modeling for accurate indoor WLAN RSS-based localization," in *Proceedings of the IEEE 72nd Vehicular Technology Conference Fall (VTC-Fall '10)*, pp. 1–5, Ottawa, Canada, September 2010.
- [6] Y.-H. Jo, J.-Y. Lee, D.-H. Ha, and S.-H. Kang, "Accuracy enhancement for UWB indoor positioning using ray tracing," in *Proceedings of the IEEE/ION Position, Location, and Navigation Symposium*, pp. 565–568, San Diego, Calif, USA, April 2006.
- [7] M. A. Quddus, W. Y. Ochieng, and R. B. Noland, "Current map-matching algorithms for transport applications: state-of-the art and future research directions," *Transportation Research Part C: Emerging Technologies*, vol. 15, no. 5, pp. 312–328, 2007.
- [8] F. Gustafsson, F. Gunnarsson, N. Bergman et al., "Particle filters for positioning, navigation, and tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 425–437, 2002.
- [9] F. Evennou and F. Marx, "Advanced integration of WiFi and inertial navigation systems for indoor mobile positioning," *EURASIP Journal on Applied Signal Processing*, vol. 2006, Article ID 86706, 2006.
- [10] S. Beauregard, Widyawan, and M. Klepal, "Indoor PDR performance enhancement using minimal map information and particle filters," in *Proceedings of the IEEE/ION Position, Location and Navigation Symposium*, pp. 141–147, Monterey, Calif, USA, May 2008.
- [11] P. Davidson, J. Collin, and J. Takala, "Application of particle filters for indoor positioning using floor plans," in *Proceedings of the Ubiquitous Positioning Indoor Navigation and Location Based Service (UPINLBS '10)*, pp. 1–4, Kirkkonummi, Finland, October 2010.
- [12] A. U. Peker, O. Tosun, and T. Acarman, "Particle filter vehicle localization and map-matching using map topology," in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV '11)*, pp. 248–253, IEEE, Baden-Baden, Germany, June 2011.
- [13] S. Kaiser, M. Khider, and P. Robertson, "A maps-based angular PDF for navigation systems in indoor and outdoor environments," in *Proceedings of the IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN '11)*, pp. 1–7, IEEE, Guimarães, Portugal, September 2011.
- [14] T. S. Rappaport, *Wireless Communications: Principles and Practice*, Prentice-Hall, 2002.
- [15] F. Montorsi, F. Pancaldi, and G. M. Vitetta, "Map-aware models for indoor wireless localization systems: an experimental study," *IEEE Transactions on Wireless Communications*, vol. 13, no. 5, pp. 2850–2862, 2014.
- [16] F. Montorsi, F. Pancaldi, and G. M. Vitetta, "Map-aware RSS localization models and algorithms based on experimental data," in *Proceedings of the IEEE International Conference on Communications (ICC '13)*, pp. 5798–5803, Budapest, Hungary, June 2013.
- [17] R. Cools, "An encyclopaedia of cubature formulas," *Journal of Complexity*, vol. 19, no. 3, pp. 445–453, 2003.
- [18] A. Haegemans and R. Piessens, "Construction of cubature formulas of degree seven and nine symmetric planar regions, using orthogonal polynomials," *SIAM Journal on Numerical Analysis*, vol. 14, no. 3, pp. 492–508, 1977.
- [19] M. Powell, "Direct search algorithms for optimization calculations," *Acta Numerica*, vol. 7, pp. 287–336, 1998.
- [20] T. G. Kolda, R. M. Lewis, and V. Torczon, "Optimization by direct search: new perspectives on some classical and modern methods," *SIAM Review*, vol. 45, no. 3, pp. 385–482, 2003.
- [21] I. Güvenç and C.-C. Chong, "A survey on TOA based wireless localization and NLOS mitigation techniques," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 3, pp. 107–124, 2009.
- [22] R. Karlsson, T. Schön, and F. Gustafsson, "Complexity analysis of the marginalized particle filter," *IEEE Transactions on Signal Processing*, vol. 53, no. 11, pp. 4408–4411, 2005.
- [23] J. Yan, C. C. J. M. Tiberius, P. J. G. Teunissen, G. Bellusci, and G. J. M. Janssen, "A framework for low complexity least-squares localization with high accuracy," *IEEE Transactions on Signal Processing*, vol. 58, no. 9, pp. 4836–4847, 2010.
- [24] F. Cheng and I.-M. Lin, "Covering of polygons by rectangles," *Computer Aided Design*, vol. 21, no. 2, pp. 97–101, 1989.
- [25] W. T. Liou, J. J.-M. Tan, and R. C. T. Lee, "Minimum rectangular partition problem for simple rectilinear polygons," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 9, no. 7, pp. 720–733, 1990.
- [26] S.-Y. Wu and S. Sahni, "Fast algorithms to partition simple rectilinear polygons," *VLSI Design*, vol. 1, no. 3, pp. 193–215, 1994.
- [27] F. Montorsi, F. Pancaldi, and G. M. Vitetta, "Technical report: algorithm implementation details and extended numerical results," Tech. Rep., University of Modena and Reggio Emilia, Modena, Italy, 2013, <http://frm.users.sourceforge.net/publications.html>.
- [28] S. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, vol. 1, Prentice-Hall, 1993.
- [29] R. Cools, "Advances in multidimensional integration," *Journal of Computational and Applied Mathematics*, vol. 149, no. 1, pp. 1–12, 2002.

- [30] M. Festa and A. Sommariva, "Computing almost minimal formulas on the square," *Journal of Computational and Applied Mathematics*, vol. 236, no. 17, pp. 4296–4302, 2012.
- [31] H. Li, J. Sun, and Y. Xu, "Cubature formula and interpolation on the cubic domain," *Numerical Mathematics: Theory, Methods and Applications*, vol. 2, no. 2, pp. 119–152, 2009.
- [32] F. Montorsi, *Localization and tracking for indoor environments [Ph.D. thesis]*, University of Modena and Reggio Emilia, Modena, Italy, 2013.
- [33] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, "Convergence properties of the Nelder-Mead simplex method in low dimensions," *SIAM Journal on Optimization*, vol. 9, no. 1, pp. 112–147, 1998.
- [34] P. Vandewalle, J. Kovacević, and M. Vetterli, "Reproducible research in signal processing: what, why, and how," *IEEE Signal Processing Magazine*, vol. 26, no. 3, pp. 37–47, 2009.





**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

