Sensor placement determination for range-difference positioning using evolutionary multi-objective optimization

Article *in* Expert Systems with Applications · November 2015

DOI: 10.1016/j.eswa.2015.11.008

6 authors, including:

Francisco Domingo-Perez
University of Alcalá
21 PUBLICATIONS **74** CITATIONS

SEE PROFILE

David Salido-Monzú
ETH Zurich
14 PUBLICATIONS **34** CITATIONS

SEE PROFILE

Sensor placement determination for range-difference positioning using evolutionary multi-objective optimization

Francisco Domingo-Perez[a,*], Jose Luis Lazaro-Galilea[a], Andreas Wieser[b], Ernesto Martin-Gorostiza[a], David Salido-Monzu[a], Alvaro de la Llana[a]

[a] *Department of Electronics, University of Alcalá, E-28806 Alcalá de Henares, Spain*
[b] *Institute of Geodesy and Photogrammetry, ETH Zürich, 8093 Zürich, Switzerland*

Abstract

This paper focuses on the application of a multi-objective genetic algorithm to sensor deployment for indoor positioning based on range-difference measurements. In our analysis, we use four scalar performance measures as objective functions derived from the covariance matrix of the estimation, namely the trace, determinant, maximum eigenvalue and ratio of maximum and minimum eigenvalues. We run the multi-objective genetic algorithm for pairs of these objectives so as to obtain the Pareto fronts in two-dimensional graphics. The paper includes a description of the scalar performance measures under consideration, the genetic algorithm and the multi-objective optimization problem of sensor deployment for range-difference of arrival. We have applied the multi-objective optimization to an infrared positioning system as an example. We use a model of this system to obtain the covariance matrix of the range-difference measurements. We take the heteroscedasticity of the measurements of different sensors and their correlations into account. Results show the optimum placement of sensors for the different objectives (single-objective and pairs of multi-objective cases) in the case of five sensors deployed throughout a regular square area. The applied method successfully generates the Pareto fronts for the objectives and allows the designer to select the proper configuration according to selected high level criteria.

Keywords: Indoor positioning, Sensor placement, Infrared sensors, Multi-objective evolutionary optimization

*Corresponding author. Address: Dpto de Electrónica, Ctra. Madrid-Barcelona, km. 33,600, 28871 Alcalá de Henares, Spain. Tel: +34 918856904.

Email addresses: fco.domingo@depeca.uah.es (Francisco Domingo-Perez), lazaro@depeca.uah.es (Jose Luis Lazaro-Galilea), andreas.wieser@geod.baug.ethz.ch (Andreas Wieser), ernesto@depeca.uah.es (Ernesto Martin-Gorostiza), david.salido@depeca.uah.es (David Salido-Monzu), alvaro.llana@depeca.uah.es (Alvaro de la Llana)

1. Introduction

Range-based localization systems use anchor nodes (nodes with fixed and known position) and measurements like received signal strength, time of arrival or time-difference of arrival to estimate the position of a target. These measurements can be converted to geometric distances or distance-differences. Exploiting the geometry of triangles, circles or hyperbolae the actual position can then be estimated. Liu et al. (2007) provide a review of wireless indoor positioning with a comprehensive comparison of different technologies. We will subsequently consider a range-difference based positioning system using modulated infrared light (Gorostiza et al., 2011) as an example, and assume that the measurements are taken at the anchor nodes and those are the sensors. However, the methods we present in this paper can be applied to any positioning technology. The concrete choice of technology is taken into account by selecting an appropriate functional and stochastic measurement model.

It is well known that the position estimation error is affected by the measurement errors, by the geometry relating sensors and target, and by the estimation algorithm. Particularly the angle of intersection of the geometric loci corresponding to the observables (e.g., of the hyperbolae in case of range-differences) affects how the measurement uncertainties propagate to uncertainties of the estimated coordinates (e.g., Ho & Chan (1993); Kaune (2012)). All these influences are contained in the covariance matrix of the estimated coordinates which is therefore a useful starting point to assess the predicted quality or to optimize sensor placement.

The Fisher information matrix (FIM) is used to compute the covariance of maximum-likelihood estimates. The Cramér-Rao lower bound (CRLB) is computed from the inverse of FIM and provides a lower bound of the covariance that is asymptotically achievable by any unbiased estimator (Kay, 1993). Bishop et al. (2010) use FIM to analyze the sensor emitter geometry and establish which sensor configurations minimize the achievable variance by an efficient estimator. Further research was focused on generalizing this work for sensor networks comprising different types of sensors (Meng et al., 2013) and considering additionally distance-dependent ranging errors (Perez-Ramirez et al., 2013). Accuracy will improve with increasing number of sensors according to the additive property of FIM, assuming that the contribution of each sensor is properly weighted during the estimation and the measurement errors are dominated by random rather than by systematic effects. Chen et al. (2006) minimize an error bound of the linear least squares estimation and conclude that optimal sensor placements are those distributions that form simple regular shapes (triangles, squares, ...). This is a useful indication for positioning within open areas without line-of-sight (LOS) restrictions. Optimizing sensor placement for coverage and accuracy within real indoor environments with irregularly shaped floor plans and LOS obstructions requires more advanced tools and will generally lead to

different optimum network geometry for different environmental restrictions and requirements.

So far, most authors applied single-objective optimization for optimum sensor placement in relation with localization. Chaudhry et al. (2011) present a multi-objective evolutionary algorithm for solving the sensor placement problem while coping with multiple, possibly contradicting, criteria (maximize coverage and connectivity, minimize energy cost). Their approach is even applicable to situations where the number of sensors is not known beforehand.

Our approach differs from the others in that we use a multi-objective optimization that considers multiple accuracy measures instead of focusing on a single measure. We consider the problem of finding a distribution of anchor nodes which optimizes the performance of the positioning system within the entire, desired coverage area (region of interest, ROI) and takes into account various scalar performance measures derived from the CRLB. This is practically relevant when planning the installation of a positioning system e.g. in an existing building or when assessing a proposed technical solution, whose performance depends on the number and spatial distribution of the sensor nodes. Each of the performance measures, which we will introduce in Section 2, has a clear physical meaning. Employing multi-objective evolutionary optimization then allows finding a sensor placement configuration which is optimal in a practically relevant sense combining the objectives. For example, we can automatically find configurations such that the worst position error within the entire ROI is a minimum while at the same time the uncertainties of the estimated positions are as isotropic as possible (i.e., at location, the uncertainties are almost equal in all directions). The associated objective functions are usually in conflict with each other so the solution of the optimization problem is initially the so-called Pareto front, i.e., a surface in the M-dimensional domain of the performance measures where each point corresponds to the optimum constrained on the values of $M-1$ criteria. Numerical optimization yields a point cloud approximating the Pareto front. The final solution of the optimization problem is then extracted from this point cloud using additional selection criteria.

The rest of the paper continues as follows. Section 2 introduces the performance measures that we propose for sensor management. Fundamentals of multi-objective optimization and its approach by evolutionary algorithms are discussed in section 3; section 3.1 gives a detailed explanation of the genetic algorithm we use. Positioning with range-difference measurements and the evaluation of position estimates are briefly recalled in section 4, including the range-difference error modeling of the infrared link. Section 5 shows the sensor placement solutions for specific examples, obtained using the evolutionary optimization.

2. Sensor resource management

Sensor resource management in target positioning focuses on sensor selection and sensor placement to decide which sensor combination will be assigned to which target over which area. The goal is to select the best sensor configuration among the available options (sensor selection) or to place sensors into the best configuration (sensor placement) under given constraints so as to optimize a performance measure defined on the resulting target estimate statistics. The design of a resource manager depends, on the one hand, on the kind of measurements available and, on the other hand, on the performance measures chosen to be optimized (Yang et al., 2012).

We propose five scalar performance measures derived from the covariance matrix (VCM) of the estimated position. Each measure is originally defined and evaluated for a single point (i.e., an assumed, stationary target position). We can derive a related scalar performance measure for the whole ROI by repeating the evaluation of the measure for all points of a sufficiently dense grid covering the entire ROI and condensing those evaluations into a single scalar value. We will focus on the transition from individual points to a whole ROI later in this section. Table 1 shows the proposed scalar performance measures, where $\hat{\theta}$ is the estimated position of the target, $\mathbf{\Sigma}_{\hat{\theta}\hat{\theta}}$ represents its covariance matrix, and $\boldsymbol{\lambda}$ is the vector of eigenvalues of $\mathbf{\Sigma}_{\hat{\theta}\hat{\theta}}$.

Table 1

Scalar performance measures referring to individual location.

Performance measure	Expression	Practical meaning related to
Trace of VCM	$\mathrm{tr}\left(\mathbf{\Sigma}_{\hat{\theta}\hat{\theta}}\right) = \sum \boldsymbol{\lambda}$	Mean square error
Determinant of VCM	$\det\left(\mathbf{\Sigma}_{\hat{\theta}\hat{\theta}}\right) = \prod \boldsymbol{\lambda}$	Volume of the error ellipsoid
Maximum eigenvalue	$\max\left(\boldsymbol{\lambda}\right)$	Largest axis of the ellipsoid
Ratio of maximum to minimum eigenvalue	$\frac{\max(\boldsymbol{\lambda})}{\min(\boldsymbol{\lambda})}$	Isometry of uncertainty
Uncertainty in spatial direction $\mathbf{a}$	$\mathbf{a}^T \mathbf{\Sigma}_{\hat{\theta}\hat{\theta}} \mathbf{a}$	Uncertainty in the direction given by the unit vector $\mathbf{a}$

Assuming normally distributed measurements and a correct functional and stochastic model, the covariance matrix represents an ellipsoid about the estimated position, containing the true position with a certain probability. The square root of the eigenvalues of the covariance matrix are the lengths of the ellipsoid axes.

The trace, determinant, and maximum eigenvalue have been widely used as optimization criteria in the theory of design of experiments, and the designs minimizing these criteria are known as A-, D-, and E-optimal, respectively. Most researchers use D-optimality because of its advantages, in particular its in-

variance under scale changes in the parameters (Uciński, 2005). E-optimality can lead to more than a single optimum solution when the number of dimensions of the problem is greater than 2 (Yang et al., 2012). We also propose the ratio of the maximum and minimum eigenvalues as a criterion because it indicates the elongation of the error ellipsoid. When optimizing for this criterion, the resulting error ellipsoid is as spherical as possible indicating that the uncertainty is almost independent of the direction. Finally, including a measure of the uncertainty in a predefined spatial direction is useful for problems where the uncertainty is more critical in a certain direction, e.g. when targets are to be kept at a safe distance from walls along a narrow hallway.

For a given sensor configuration, any scalar from Table 1 can be interpreted as a function $O(x, y, z)$ of the target position coordinates, i.e., an individual evaluation at a specific point of an ROI. We can get the scalar measure referring to a whole ROI as mean value over the whole ROI by dividing the respective volume integral by the volume V_{ROI} of the ROI:

$$O(\mathrm{ROI}) = \frac{1}{V_{\mathrm{ROI}}} \iiint_V O(x, y, z)\, dx\, dy\, dz. \tag{1}$$

We can solve Eq. (1) numerically by evaluating the function O at P grid points, i.e. by calculating $O(p_j)$, $j = 1, 2, \ldots, P$, where p_j is the j point of the grid. This enables us to derive also other representative values for the entire ROI by choosing suitable functions of $O(p_j)$. Some examples of useful functions are the mean (Eq. (2)) or the maximum (i.e., the worst case, Eq. (3)). Additionally, the evaluated function can be weighted by location dependent weights $w_j = [0, 1]$. This location dependent weighting is useful to define areas within the ROI where we are predominantly and exclusively interested in a specific measure, e.g., minimum lateral error in an elongated narrow path and minimum MSE elsewhere. Generally, we propose the following functions for the transition from the individual points p_j to the whole ROI, represented by $\{p_1, p_2, \ldots, p_P\}$:

$$O_{\mathrm{mean}}(\mathrm{ROI}) = \frac{\sum_{j=1}^{P} \left(w_j O\left(p_j\right)\right)}{\sum_{j=1}^{P} w_j}, \tag{2}$$

$$O_{\mathrm{max}}(\mathrm{ROI}) = \max\left(\bigcup_{j=1}^{P} \left(w_j O\left(p_j\right)\right)\right). \tag{3}$$

According to the conclusion provided by Yang et al. (2012), for stationary target localization applications (individual point), the trace, determinant, and maximum eigenvalue measures lead to the same goal of placing the sensors as close to the target as possible to achieve the best positioning quality and to surround the target for most isometric uncertainty. Since each measure corresponds to a different aspect of the uncertainty, there could be multiple objectives to satisfy instead of focusing on a single measure. Yang et al. (2012) left the

multi-objective case open for future work. The present work focuses on getting the Pareto front as the result of a multi-objective optimization problem.

3. Multi-objective optimization

Multi-objective optimization involves minimizing or maximizing multiple objective functions subject to a set of constraints. The M objectives are functions of N decision variables. We need to distinguish between the space of the decision variables $\left(\mathbb{D} \subseteq \mathbb{R}^N\right)$ and that one of the objectives $\left(\mathbb{O} \subseteq \mathbb{R}^M\right)$. It is usual that these objectives are in conflict with each other and the final solution requires a tradeoff between the objectives. In this case, there is no single optimum solution but a set of solutions representing the respective optimum of one objective constrained on fixed values of the other objectives. This set, expressed in the objective domain, is the Pareto front. For each point on this Pareto front it is impossible to find a vector in the space of the decision variables improving on any of the objectives without deteriorating at least one of the others. A qualitative example (for two objectives) is shown in Fig. 1. The solutions inside the feasible region $\mathcal{F}$ are dominated by those of the Pareto front $\mathcal{P}$. A solution is said to dominate another one if each of its objective values is better - or at least none is worse - than those of the other solution. The end points of the Pareto front correspond to decision variables which globally optimize the respective objective. A solution whose objective values are both optimum is shown by $\boldsymbol{O}\left(\mathbf{x}^{\mathcal{I}}\right)$ in the figure, but it lies in the infeasible region $\mathcal{I}$, i.e., there is no set of decision variables which produces these values of the objectives.

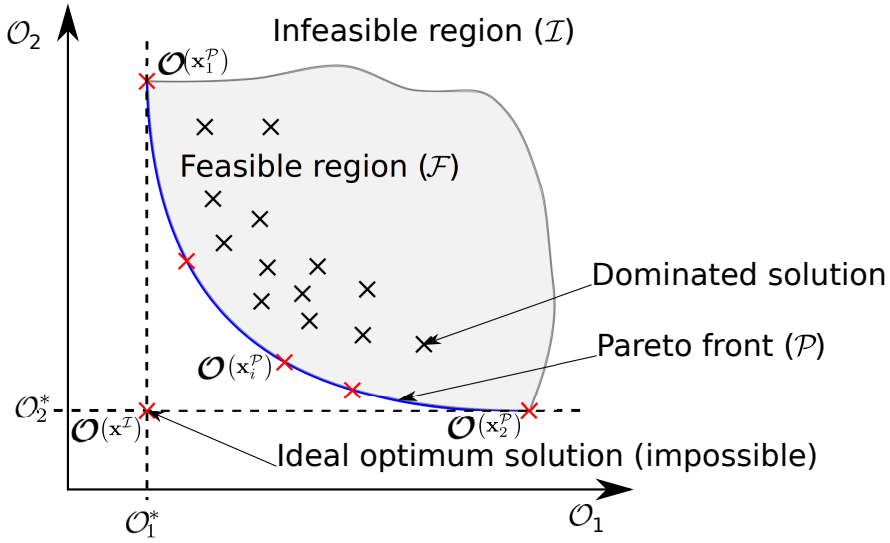


Fig. 1. Symbolic representation of solutions in the space of the objectives of a two-objective optimization problem.

More formally, the general multi-objective optimization problem has the following form:

$$\begin{aligned} &\text{Minimize} && O_m(\mathbf{x}), && m = 1, 2, \ldots, M; \\ &\text{subject to} && g_j(\mathbf{x}) \leq 0, && j = 1, 2, \ldots, J; \\ & && h_k(\mathbf{x}) = 0, && k = 1, 2, \ldots, K, \end{aligned} \tag{4}$$

where M is the number of objective functions O_m, $\mathbf{x}$ is a vector of N decision variables, g_j and h_k are constraints. If the original criterion requires maximization it is multiplied by -1 and minimized.

The values of the objective functions can be represented as points $\boldsymbol{O}_j = \boldsymbol{O}\left(\mathbf{x}_j\right) = \left[O_1\left(\mathbf{x}_j\right), O_2\left(\mathbf{x}_j\right), \ldots, O_M\left(\mathbf{x}_j\right)\right]$ in the space $\mathbb{O} \subseteq \mathbb{R}^M$ of the objectives. Different combinations of the decision variables, i.e. different points $\mathbf{x}_j$ of the decision space $\mathbb{D} \subseteq \mathbb{R}^N$, result in different vectors $\boldsymbol{O}_j$. We can sort these vectors using the product order $\leq_p$ where $\boldsymbol{O}(\mathbf{x}_a) \leq_p \boldsymbol{O}(\mathbf{x}_b)$ if and only if $O_m(\mathbf{x}_a) \leq O_m(\mathbf{x}_b), \forall m \in \{1, 2, \ldots, M\}$. If we denote with $\mathcal{F}$ the set of feasible objective vectors, i.e. those elements $\boldsymbol{O}_k \in \mathbb{O}$ for which $\exists \mathbf{x}_k \in \mathbb{D} : \boldsymbol{O}(\mathbf{x}_k) = \boldsymbol{O}_k$, and with $\mathcal{P}$ those elements of $\mathcal{F}$ which form the Pareto front, it is certain that $\mathcal{P} \subset \mathcal{F}$ and $\mathcal{P}$ can be defined as $\left\{\boldsymbol{O}\left(\mathbf{x}^{\mathcal{P}}\right) \in \mathcal{P} \mid \nexists \boldsymbol{O}\left(\mathbf{x}^{\mathcal{F}}\right) \in \mathcal{F} : \boldsymbol{O}\left(\mathbf{x}^{\mathcal{F}}\right) \leq_p \boldsymbol{O}\left(\mathbf{x}^{\mathcal{P}}\right)\right\}$. The values $\mathbf{x}^{\mathcal{P}}$ of the decision variables whose evaluation of the objective functions belongs to the Pareto front $\mathcal{P}$ are the solutions of the multi-objective optimization problem.

Selecting a single decision variable $\mathbf{x}_i^{\mathcal{P}}$ as preferred solution from the Pareto front requires additional knowledge or criteria. Common techniques of preferred solution selection include the computation of a single scalar objective function from the M different objectives (utility function approach) so that a single solution is obtained using a single-objective optimization. Another approach consists of assigning weights to the objectives along the Pareto front and selecting the proper solution according to the situation. The weights range from 0 to 1 for each objective, it is up to the designer to select appropriate weights. Finally, we could also select the point on the Pareto front which is next to the ideal optimum, i.e., has the shortest distance from $\boldsymbol{O}\left(\mathbf{x}^I\right)$ as of Fig. 1 (Deb, 2001). The main disadvantage of the utility function approach is that the design of the utility function may be arbitrary - and therefore the finally selected optimum. Basing the decision on an analysis of the Pareto front allows to better understand the tradeoffs involved. The shortest distance from the ideal optimum allows to automatically and objectively select a solution, whereas the weights approach can be useful for having a representation of the solutions with their percentage of objective fulfillment and selecting a value according to current priorities.

3.1. Evolutionary multi-objective optimization

Evolutionary algorithms are among the most popular metaheuristic algorithms for solving multi-objective optimization problems. One of the main reasons is that they can evaluate a whole population

(a population is made of individuals, each representing a decision variable vector $\mathbf{x} \in \mathbb{D}$) in a single run of the algorithm, so that it is possible to find several members of the Pareto optimal set in a single run. Additionally, they are less sensitive to the shape and continuity of the Pareto front (Coello Coello, 2006). In the sensor placement problem introduced above, a single individual consists of three coordinates (3D case) per anchor, i.e., $3n$ variables with n anchors. Given a population size of 100 individuals and five sensors a single run of the algorithm evaluates the fitness of 100 candidates, each consisting of 15 variables.

We have chosen to use the modified version of the NSGA-II algorithm (Deb et al., 2002) as implemented in the Global Optimization Toolbox™ of Matlab®. The algorithm is briefly explained in this section for convenience. The algorithm is based on the computation of how many individuals dominate each other and identifying the set of individuals which each one dominates, assigning a nondomination rank to each individual. The nondomination rank and a measure of the density of individuals surrounding a particular one (crowding distance) are used to compare individuals with respect to fitness.

The initial population (of size s_p) is randomly generated in the decision variable space and the objective functions are evaluated for each individual $\mathbf{x}_i^{(0)}$, where the superscript (0) denotes the initial generation and i takes integer values from 1 to s_p. Then the nondomination rank is assigned to each individual. Any individual whose vector of evaluated objectives is not greater than any other objective vector (in the sense of the above product order) has rank = 1. The ranked individuals are then removed and the product order comparison is performed again to assign rank = 2. The assignation continues until the whole population is ranked. Each rank number corresponds to a front and a front of individuals with a given rank will dominate all fronts associated with a greater rank. The individuals belonging to the same front are assigned a crowding distance. For each one of the objectives O_j, $j = 1, 2, \ldots, M$, the values of all individuals i within the front are normalized according to $O'_{j,i} = \frac{O_{j,i}}{1 + \max\limits_{i=1,\ldots,s_f} |O_{j,i}|}$, where s_f is the number of individuals of the front and j and i are as above. The objectives are then sorted such that $O'_{j,1} \leq O'_{j,2} \leq \cdots \leq O'_{j,s_f}$. The best and the worst individuals are given an infinite distance value. The distance of remaining individuals is $d_{c_{j,i}} = O'_{j,i+1} - O'_{j,i-1}$, where $d_{c_{j,i}}$ is the crowding distance of the i individual of a front for the objective j. This process is done for each objective and the $d_{c_{j,i}}$ values for each individuals are added to compute the final crowding distance $d_{c_i} = \sum_{j=1}^{M} d_{c_{j,i}}$ of individual i.

At the end of the above step we have the ranks and the crowding distances of the initial population. The first genetic algorithm step where a new population of size s_p is created with crossover and mutated children now takes place. The number of crossover children is $n_x = r_x s_p$, where r_x is a predefined ratio. The number of mutated children is $n_m = s_p - n_x$. We need two parents for each crossover child and only one parent for each mutated child, making the number of parents $n_p = 2n_x + n_m$. A tournament selection process chooses n_p

parents making n_p tournaments among four randomly picked individuals chosen from the current population. In case that two chosen individuals have equal rank the one with greater crowding distance is preferred to keep diversity. Crossover children are generated as weighted average of their respective parents, evaluated in the decision variable domain: $\mathbf{x}_{*_i}^{(l)} = \mathbf{x}_{i_1}^{(l)} + \boldsymbol{\rho} \circ \left(\mathbf{x}_{i_2}^{(l)} - \mathbf{x}_{i_1}^{(l)}\right)$, where $\boldsymbol{\rho}$ is a vector whose elements are random numbers between 0 and 1, acting as weights in the crossover process, $\mathbf{x}_{*_i}^{(l)}$ is the i^{th} child emerging from the l^{th} generation, $\mathbf{x}_{i_k}^{(l)}$ is the k^{th} parent of the i^{th} child, and $\circ$ denotes the entry-wise product. Expressed in the decision variables domain, each crossover child will be inside the hypercube defined by its two parents.

The adaptive feasible mutation method is used for generating n_m further children. This method uses a set of randomly generated directions together with directions parallel to the boundaries in case the decision variable is near any boundary constraint. This set is randomly permuted and the variables are moved in the first direction an initial step size, i.e., the step size is the module of the displacement vector. The feasibility of the movement is checked with the constraints. In case of feasibility the new variable is the mutated child, otherwise the individual keeps its previous value and the next direction is checked. Subsequent calls to the mutation function will test the improvement of the current generation in order to decrease the step size in case there is no improvement. A more detailed explanation of adaptive feasible mutation can be found in Kumar (2010).

At this point we have the old population (l^{th} generation) extended by the newly created crossover and mutated children. The algorithm will now create the $(l+1)^{\text{th}}$ generation by selecting s_p individuals from this extended population of the l^{th} generation. Therefore, the fitness of the children is calculated, the old population and the children are merged and the ranks and crowding distances are computed within this extended population. From the first front we extract $r_{\mathcal{P}} s_p$ individuals, where $r_{\mathcal{P}}$ indicates the fraction of individuals from the Pareto front and takes values between 0 and 1. The number of individuals to keep from remaining fronts (when existing) is chosen with a geometric progression of ratio 0.8, decreasing as long as the rank increases. The number of individuals to be retained from each front is adjusted in case there are less individuals than individuals to be retained. The individuals from each front are sorted by crowding distance and those with the smaller values are removed from being part of the next population. Since both parents and children are merged before computing rank and distance and creating the next generation, elitism is present in the sense that parents can also be part of the population of the following generation.

The steps of evaluation of objectives, ranking and derivation of a new generation are now carried out iteratively for several generations. The loop is aborted when the generation number reaches a predefined threshold G. Finally, the algorithm returns the individuals of the best front of the last generation. They are the estimate of the Pareto front.

4. Range-difference based position estimation

This section sumarizes the position estimation problem with range-difference measurements. The estimate of the position is obtained using the maximum likelihood principle and applying the Gauss-Newton method to solve the nonlinear problem. Then, we focus on our proposal and show the expressions of the measures of table 1. We formulate the multi-objective optimization problem for the example of five sensors to cover a regular square area. The objectives that we have considered include average uncertainty over the whole area and the maximum single-axis uncertainty. For simplicity we assume that we know the exact height of the emitter (0.65 m) and work in 2D further on. Finally, the last section of this part focuses on modeling the covariance matrix of the measurement for the infrared system that we use as an example of RDOA based positioning system. The synthetic measurement data we used to obtain the results of section 5 use a covariance matrix whose elements are distance-dependent and whose covariances properly reflect the correlations due to the use of a single sensor as reference.

4.1. Position estimation

Having obtained K range measurements we can pick $K-1$ sensors and difference their measurement values with the value of the remaining sensor, which acts as a reference. It is irrelevant which sensor is picked as the reference since we take into account the mathematical correlation of the measurements appropriately and can carried out outlier detection at the level of the undifferenced observations (Wieser, 2004). The $K-1$ range-difference measurements Δd can be expressed as a function of the parameters to be estimated ($\theta = [x \quad y \quad z]^T$):

$$\mathbf{r} = \boldsymbol{\mu}(\theta) + \boldsymbol{\epsilon}, \tag{5}$$

where each element of the equation is an $K-1$ column vector:

$$\mathbf{r} = \begin{pmatrix} \Delta d_{1,r} \\ \Delta d_{2,r} \\ \vdots \\ \Delta d_{K-1,r} \end{pmatrix}; \boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_{1,r} \\ \epsilon_{2,r} \\ \vdots \\ \epsilon_{K-1,r} \end{pmatrix}; \boldsymbol{\mu}(\theta) = \begin{pmatrix} \|\theta - \mathbf{x}_1\| - \|\theta - \mathbf{x}_r\| \\ \|\theta - \mathbf{x}_2\| - \|\theta - \mathbf{x}_r\| \\ \vdots \\ \|\theta - \mathbf{x}_{K-1}\| - \|\theta - \mathbf{x}_r\| \end{pmatrix}. \tag{6}$$

$\Delta d_{i,r}$ is the range-difference measurement between the i sensor and the reference, where i takes values from 1 to $K-1$. $\mathbf{x}_i$ is the coordinate vector of sensor i, $\|\cdot\|$ represents the L^2 norm and $\epsilon_{i,r}$ is the distance difference measurement error of the i and reference sensors. Assuming normal distribution, the range-difference measurement is modeled as $\mathbf{r} \sim \mathcal{N}(\boldsymbol{\mu}(\theta), \boldsymbol{\Sigma})$. Having a common reference for all the RDOA measurements implies that the reference error is present in every RDOA measurement, thus the covariance matrix $\boldsymbol{\Sigma}$ is fully

populated. The maximum likelihood estimate of θ can be obtained using the iterative Gauss-Newton method (So, 2011):

$$\hat{\theta}^{k+1} = \hat{\theta}^{k} + \left(\mathbf{J}_k^T \boldsymbol{\Sigma}^{-1} \mathbf{J}_k\right)^{-1} \mathbf{J}_k^T \boldsymbol{\Sigma}^{-1} \left(\mathbf{r} - \boldsymbol{\mu}\left(\hat{\theta}^{k}\right)\right), \tag{7}$$

with $\mathbf{J}_k = \left.\frac{\partial \boldsymbol{\mu}(\theta)}{\partial \theta}\right|_{\theta=\hat{\theta}_k}$ and reasonably chosen initial values $\hat{\theta}^{0} = \theta_0$. Note that k is the iteration number and $\mathbf{J}_k$ is the Jacobian matrix of $\boldsymbol{\mu}$, which is updated in every iteration.

$\boldsymbol{\Sigma}_{\hat{\theta}\hat{\theta}} = \left(\mathbf{J}^T \boldsymbol{\Sigma}^{-1} \mathbf{J}\right)^{-1}$ is the expression of the CRLB computed from the inverse of the FIM. Since the maximum likelihood estimator asymptotically achieves CRLB we can use the previous expression to compute the covariance of the estimation and get the performance measures of table 1. The reader can easily see that the CRLB is also part of Eq. (7), hence we can obtain the covariance of the estimation while computing the estimate of the position. In the following, we show how we obtained the values of matrix $\boldsymbol{\Sigma}$ for the infrared system used as example.

4.2. Error model of range-difference observations with infrared sensors

We use an infrared indoor positioning system using the RDOA principle to demonstrate the proposed approach to sensor placement in Section 5. Since the error model of the range-difference observations in terms of the covariance matrix $\boldsymbol{\Sigma}$ (see above) is required for solving the sensor placement problem using the previously described algorithm and objectives, we briefly summarize the functionality of this system here and derive $\boldsymbol{\Sigma}$. More details about the system can be found in Gorostiza et al. (2011).

The system achieves the RDOA principle by measuring phase differences of a modulated infrared signal emitted by a mobile target and recorded continuously by receivers (anchors) at fixed and known positions. The infrared emitter uses a wide angle IR-LED at 940 nm, intensity modulated at 8 MHz, in order to generate the measurement signal. The receivers, placed at the ceiling of the coverage area, are formed by a low level conditioning stage adapting the photocurrent generated by a wide angle silicon PIN photodiode. The outputs of the receivers are simultaneously digitized and RDOA measurements are estimated from the resulting sequences.

The optical power P_o reaching an arbitrary receiver is given by the emitted power and the solid angle of the emission covered by the receiver photodiode sensitive area as expressed in Eq. (8):

$$P_o = \frac{I_e}{d^2} \pi \left(\frac{D}{2}\right)^2 \cos^2(\phi), \tag{8}$$

where I_e is the emitted optical power per solid angle in the direction normal to the emitter surface, D is the receiver photodiode sensitive area diameter, d is the Euclidean distance between emitter and receiver and ϕ is the line-of-sight angle to the vertical direction representing the decay on both emitter and receiver effective

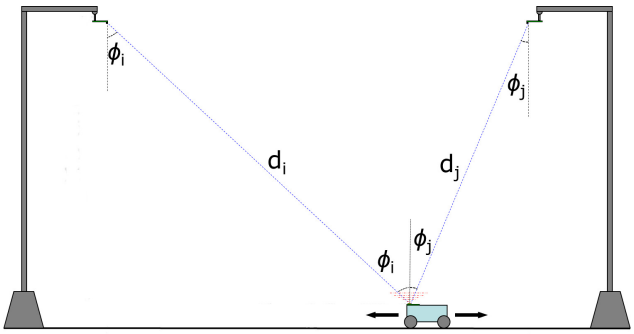


Fig. 2. Infrared emitter-sensor link.

areas due to disorientation (see Fig. 2). The received optical power generates a proportional photocurrent which is converted into a voltage and conditioned in the receivers. The output voltage V_o of an arbitrary receiver is defined as:

$$V_o = P_o \mathcal{R} G_{V/I} G_{\mathrm{VV}}, \tag{9}$$

where $\mathcal{R}$ is the responsivity of the photodiode, $G_{V/I}$ is the transimpedance amplifier gain at the modulation frequency and G_{VV} is the total voltage gain of subsequent conditioning stages.

The output SNR of a receiver, shown in Eq. (10), is defined by the output signal power and the total noise contributions of the infrared link, dominated by the noise added in the transimpedance amplifier, hence uncorrelated between different receivers.

$$\mathrm{SNR} = \frac{\frac{V_o^2}{2}}{N_0 B_n}. \tag{10}$$

N_0 is the total noise spectral density around the modulation frequency, considered to be flat given the narrow band where the final estimation is carried out, and B_n is the noise equivalent bandwidth defined by the final filtering stage around the band of interest.

The variance of each receiver single-phase estimation is inversely proportional to its output SNR, yielding a typical standard deviation σ_φ of the single-phase measurement:

$$\sigma_\varphi \simeq \frac{\gamma}{\sqrt{\mathrm{SNR}}}, \tag{11}$$

where γ is a scale factor and during simulations $\gamma = 1$. σ_φ can easily be converted into a distance standard deviation (σ_d) considering the emitted signal modulation wavelength which can be expressed with sufficient accuracy for the current purpose using the modulation frequency f_m and the approximate propagation speed $c = 3 \cdot 10^8$ m/s.

$$\sigma_d = \sigma_\varphi \frac{c}{2\pi f_m}. \tag{12}$$

Table 2 shows the numerical values of our real IR-based positioning system, as needed for Eqs. (8) to (12).

Table 2

System parameters.

	Parameter	Symbol	Value	Units
Emitter	Intensity	I_e	50	$\frac{\mathrm{W}}{\mathrm{sr}}$
	Frequency	f_m	8	MHz
Receiver	Diameter	D	5.08	mm
	Responsivity	$\mathcal{R}$	0.64	$\frac{\mathrm{A}}{\mathrm{W}}$
	I to V gain	$G_{V/I}$	$\frac{33}{\sqrt{2}}$	kΩ
	Total voltage gain	G_{VV}	100	–
	Noise equivalent bandwidth	B_n	$30 \cdot \frac{\pi}{2}$	Hz
	Noise spectral density	N_O	$1.34 \cdot 10^{-11}$	$\frac{\mathrm{W}}{\mathrm{Hz}}$

5. Numerical example

This section shows simulation results of the sensor placement problem solved with the above evolutionary algorithm. First, a single-objective optimization is shown using each of the five scalar measures introduced in section 2. We show the error ellipses to check the practical meaning mentioned at Table 1. Then, the multi-objective optimization simulation shows the point cloud of the sensor placement corresponding to a Pareto front for different combinations of two of the five objectives. We have chosen to show combinations of two objectives so that the Pareto front could be clearly represented. However, an example with three objectives has been included as well.

We have run simulations for placing five sensors over a squared area of side 3 m. We have considered an empty area without line-of-sight obstruction so that any sensor can cover almost the whole area. Thus, we do not need to take into account neither the number of sensors to deploy nor the necessity of each point in the area to be 3-covered, we will deal with both issues in future work. We have evaluated a total of uniformly spaced 121 points which have been equally considered $\left(w_j = 1\right)$, we have also included an example with location dependent weights in the multi-objective optimization. The GA has been run for 1000 generations, the values of the different parameters that were mentioned in section 3.1 appear in table 3. The population size is 15 times the number of decision variables (five sensors with three coordinates, i.e. 15 variables).

We have created an initial population placing the five sensors all over the area dividing the square in five squares of side 1.5 m. One of the squares is centered inside the original square and the remaining squares are those left by dividing the first area horizontally and vertically, the former square overlaps the other. The first individuals are generated by randomly placing each of the five sensors in a square of side 1.5 m. This constraint on the initial population has been made so that we can initially have a deployment layout approaching the optimum.

Table 3

Chosen GA parameters.

Parameter	Symbol	Value
Maximum number of generations	G	1000
Population size	s_p	225
Pareto fraction	$r_{\mathcal{P}}$	0.35
Crossover fraction	r_x	0.8

Restricting the height of the sensors to 2.80 m (since the height of the emitter was fixed to 0.65 m, the difference of the height of the sensors and the emitter is 2.15 m), the optimization problem takes the form:

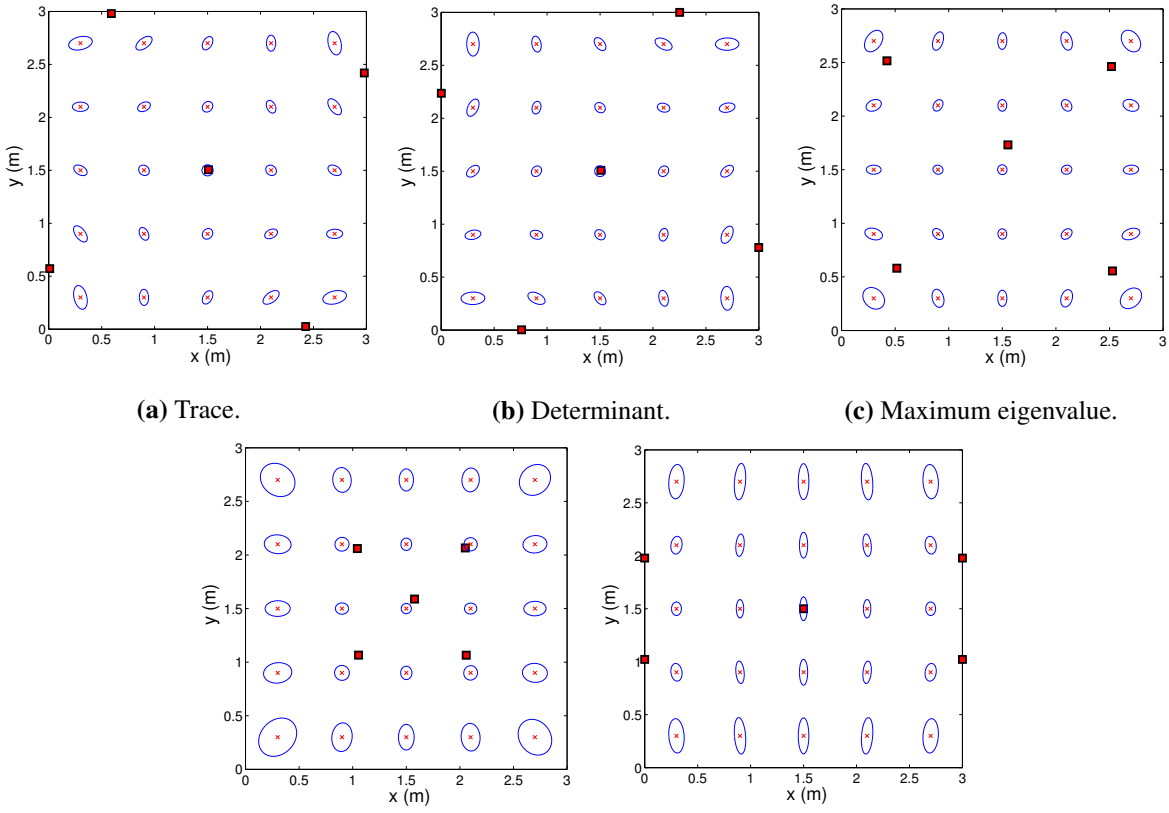
$$\begin{aligned} &\text{Minimize} && \boldsymbol{O}(\mathbf{x}); \\ &\text{subject to} && 0 \leq x_i \leq 3, \quad i = 1, 2, \ldots, 5; \\ & && 0 \leq y_i \leq 3, \quad i = 1, 2, \ldots, 5; \\ & && z_i = 2.80, \quad i = 1, 2, \ldots, 5, \end{aligned} \tag{13}$$

Where the decision variable $\mathbf{x}$ is a vector of the sensor positions $\mathbf{x} = \begin{bmatrix} x_1 & y_1 & z_1 & \ldots & x_5 & y_5 & z_5 \end{bmatrix}$ and the objective vector $\boldsymbol{O}$ will change according to the objective/s to minimize, see Table 1 and Eqs. (2) and (3).

5.1. Single-objective optimization

Fig. 3 shows the optimum sensor placement according to different single optimality criteria. The trace and determinant solutions are almost the same keeping in mind the symmetry of the considered area. It can be seen that the sensor distribution follows a square distribution plus center of mass (as showed in Chen et al. (2006)) for the five cases but the distance from the center is different in each case. Increasing the proximity to the center of the area improves the average circularity throughout the whole area, but it decreases the MSE and the volume of the error ellipsoid. The location dependent uncertainty has been analyzed with $\mathbf{a} = \begin{bmatrix} 1 & 0 \end{bmatrix}^T$, i.e., a fixed direction namely the uncertainty in the x-axis direction. The maximum eigenvalue

is evaluated from the worst case of the whole area (Eq. (3)), whereas the other objectives are evaluated as an average (Eq. (2)) over the ROI.



(a) Trace. **(b)** Determinant. **(c)** Maximum eigenvalue.

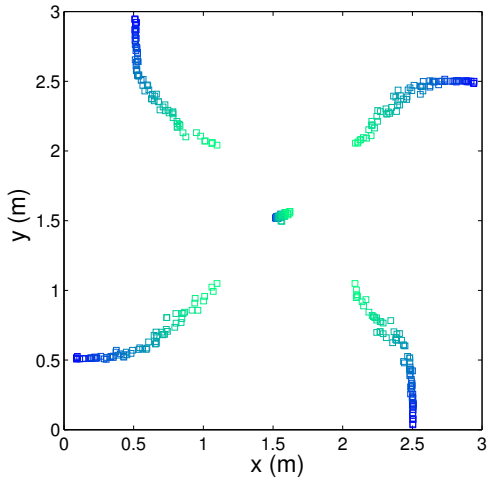
(d) Ratio of eigenvalues. **(e)** Lateral deviation.

Fig. 3. Sensor placement according to different optimality criteria. Sensors are depicted as red squares, the scaled error ellipses of a subset of the grid points are shown in blue, the subset of points appears as red crosses.

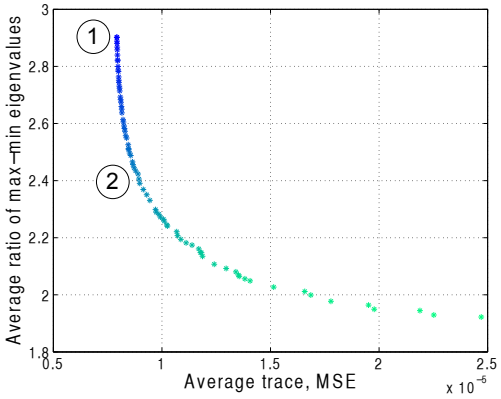
The found optimum placement slightly deviates from the expected one (perfectly symmetric). This deviation can be caused by the adjustment of the parameters in table 3. Further investigation on this topic will be carried out in the future.

5.2. Multi-objective optimization

The results of various multi-objective optimizations on pairs of criteria are shown and discussed below. Each of the figures shows potential optimum locations of the sensors on the left and the corresponding points of the Pareto front in the objective domain on the right. Figs. 4 to 8 should be read following the color in the Pareto front and locating the five points of the same color in the point cloud. We have not shown simulation results for the determinant criterion because they are similar to the ones using the trace.

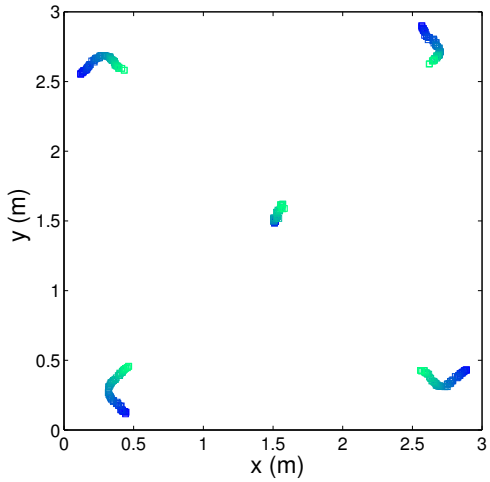


(a) Point cloud of sensor deployment.

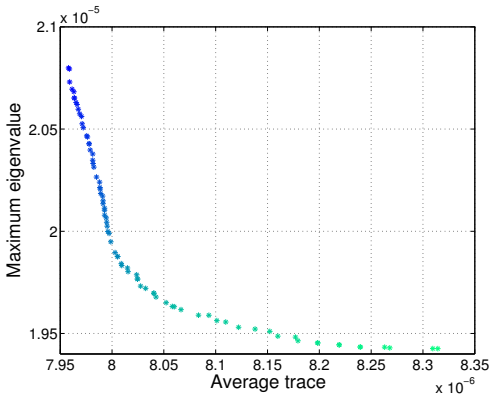


(b) Pareto front.

Fig. 4. Sensor placement and Pareto front of trace and eigenvalues ratio optimality.



(a) Point cloud of sensor deployment.

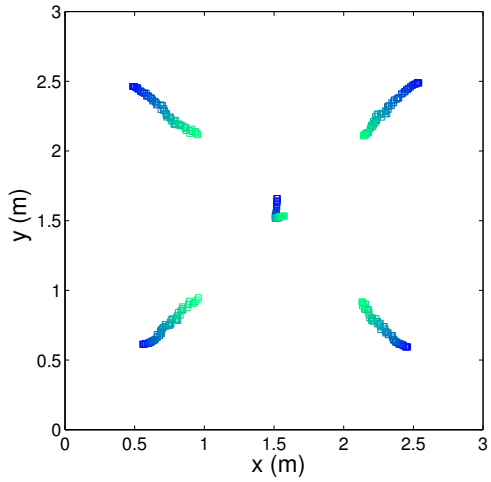


(b) Pareto front.

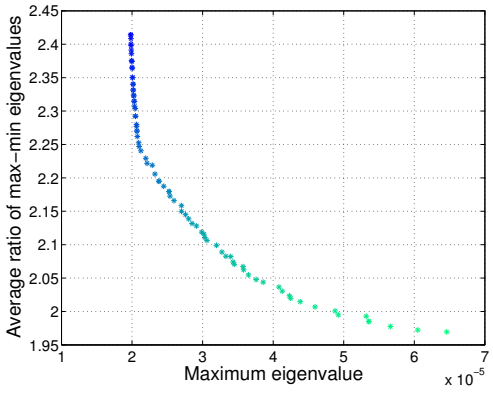
Fig. 5. Sensor placement and Pareto front of trace and maximum eigenvalue optimality.

The example of Fig. 7 has been obtained using location dependent weights. The lateral deviation objective is evaluated in a small corridor in the center, where the weights w_{lat} take the value 1 for $1.2 \leq x \leq 1.8$ and 0 otherwise. The trace is evaluated with the opposite weights, i.e. $w_{\text{tr}} = 1 - w_{\text{lat}}$.

Studying the Pareto front allows the selection of a configuration that provides the different performance measures throughout the covered area. The selection of the desired solution should be made by considering the specific case under study, e.g. if we are trying to locate a mobile in a tunnel or corridor we might not be interested in getting a circular error bound but the minimum uncertainty in a specific direction. Fig. 4

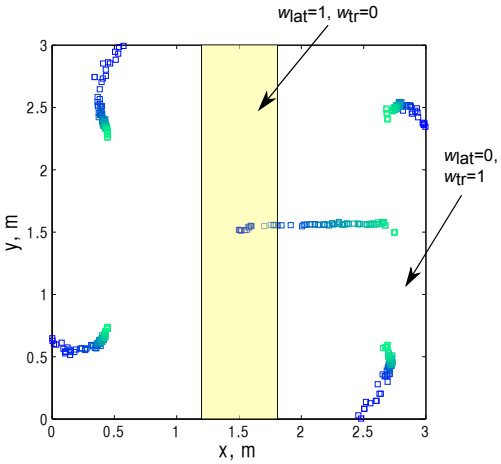


(a) Point cloud of sensor deployment.

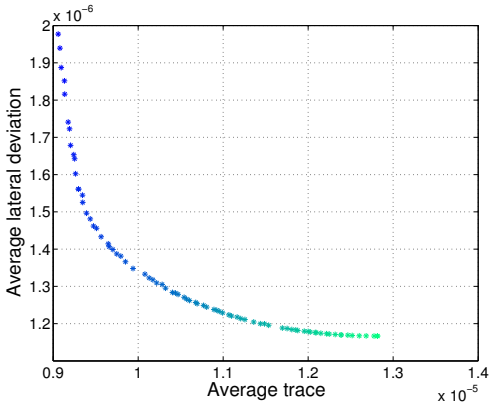


(b) Pareto front.

Fig. 6. Sensor placement and Pareto front of maximum eigenvalue and eigenvalues ratio optimality.



(a) Point cloud of sensor deployment.



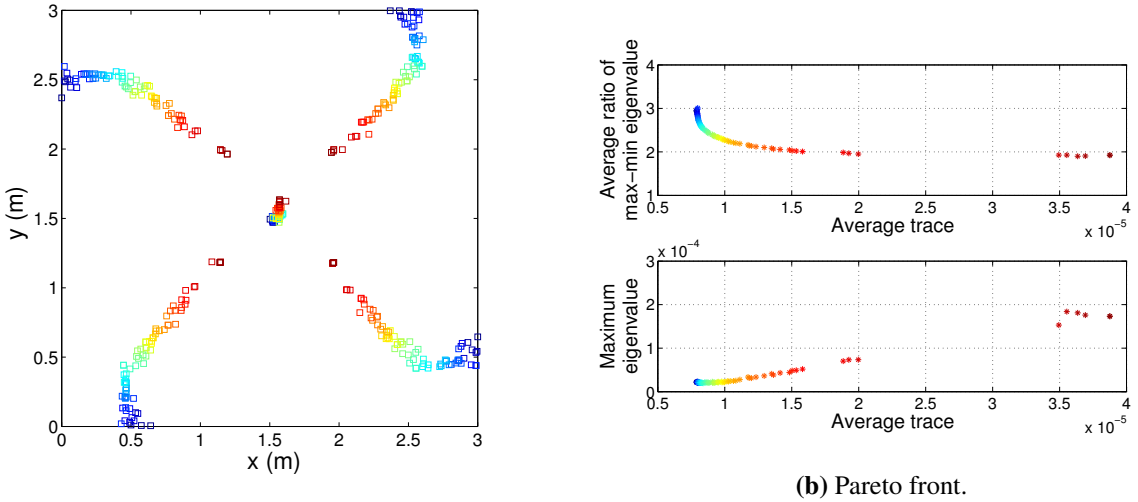
(b) Pareto front.

Fig. 7. Sensor placement and Pareto front using trace ($x < 1.2 \cup x > 1.8$) and lateral deviation ($1.2 \leq x \leq 1.8$) criteria.

shows that when minimizing the MSE over the entire area we would obtain a sensor distribution that provides highly unisotropic uncertainties (variance is three times larger in direction of maximum uncertainty than in direction of minimum uncertainty, see point 1 in Fig. 4b). However, descending along the Pareto front from the position of minimum MSE or determinant we reach with only a small deterioration of MSE a sensor distribution yielding a ratio of maximum and minimum eigenvalues of 2.4, (point 2 in Fig. 4b), i.e. we find a solution which is only very slightly less optimum with respect to MSE but much better in terms of isometry. Figs. 5 and 6 use the maximum eigenvalue for optimization, trying to reduce the maximum uncertainty in any point of the cell (the worst case). This measure avoids the potential problem of low errors in some parts

of the ROI masking greater errors in other parts of the region. Fig. 6 shows that the spread of the eigenvalues can be reduced from 2.4 to 2.25 while keeping the maximum eigenvalue almost constant.

The last simulation shows a case with three objectives.



(a) Point cloud of sensor deployment.

(b) Pareto front.

Fig. 8. Sensor placement and Pareto front of trace, eigenvalues ratio and maximum eigenvalue.

Finally, after obtaining the Pareto front and its corresponding sensor distributions we have to make the final decision and pick a single solution. This decision requires a high level knowledge of the final application and is left to the localization infrastructure designer.

6. Conclusion

This paper has shown how we can use genetic algorithms to deploy a positioning infrastructure contributing to the state-of-the-art in using more than one optimization criterion based on the covariance matrix of the estimate. This kind of optimization allows the deployment of sensors to fulfill several requirements according to specific conditions. Besides, making a weighted mean to compute the objective functions can be useful to divide the region of interest in different zones with priority levels by setting the corresponding weight values $\left(w_j\right)$. We can also use the deployment patterns that we obtained (the point clouds) for different values of the objective functions to generate trajectories for the sensors (in case they can be in motion). This approach results in a mobile positioning system whose sensors can be rearranged according to the desired criteria and the estimated position of the target.

Future work will focus on designing localization infrastructures in a general case (including occlusions due to walls, blind zones, restricting the sensor position in some places...). Using more sensors will increase

the accuracy in the area. In order to look for an optimum number of deployed sensors we need to define a metric that evaluates the difference of the objective functions as well as the drawbacks of including more sensors, i.e. the increase of computational cost, energy consumption, and cost of the whole system.

Acknowledgements

This research was supported by the Spanish Research Program through the project ESPIRA (ref. DPI2009-10143) and by the Spanish Ministry of Economy and Competitiveness through the ALCOR project (DPI2013-47347-C2-1-R). F. Domingo-Perez thanks the FPU program from the Spanish Ministry of Education, Culture and Sport (ref. BOE-A-2012-6238).

References

Bishop, A. N., Fidan, B., Anderson, B., Doğançay, K., & Pathirana, P. N. (2010). Optimality analysis of sensor-target localization geometries. *Automatica*, *46*, 479–492.

Chaudhry, S. B., Hung, V. C., Guha, R. K., & Stanley, K. O. (2011). Pareto-based evolutionary computational approach for wireless sensor placement. *Eng. Appl. Artif. Intell.*, *24*, 409–425.

Chen, Y., Francisco, J., Trappe, W., & Martin, R. (2006). A practical approach to landmark deployment for indoor localization. In *3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks (SECON 2006)* (pp. 365–373). IEEE.

Coello Coello, C. A. (2006). Evolutionary multi-objective optimization: a historical view of the field. *IEEE Comput. Intell. Mag.*, *1*, 28–36.

Deb, K. (2001). *Multi-objective Optimization Using Evolutionary Algorithms*. Chichester, UK: John Wiley & Sons.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.*, *6*, 182–197.

Gorostiza, E. M., Lázaro Galilea, J. L., Meca Meca, F. J., Salido Monzú, D., Espinosa Zapata, F., & Pallarés Puerto, L. (2011). Infrared sensor system for mobile-robot positioning in intelligent spaces. *Sensors*, *11*, 5416–5438.

Ho, K. C., & Chan, Y. T. (1993). Solution and performance analysis of geolocation by TDOA. *IEEE Trans. Aerosp. Electron. Syst.*, *29*, 1311–1322.

Kaune, R. (2012). Accuracy studies for TDOA and TOA localization. In *15th International Conference on Information Fusion (FUSION 2012)* (pp. 408–415). IEEE.

Kay, S. (1993). *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ: Prentice-Hall PTR.

Kumar, R. (2010). System and method for the use of an adaptive mutation operator in genetic algorithms. *U. S. Patent*, *7,660,773*.

Liu, H., Darabi, H., Banerjee, P., & Liu, J. (2007). Survey of wireless indoor positioning techniques and systems. *IEEE Trans. Syst. Man Cybern. Part C-Appl. Rev.*, *37*, 1067–1080.

Meng, W., Xie, L., & Xiao, W. (2013). Optimality analysis of sensor-source geometries in heterogeneous sensor networks. *IEEE Trans. Wirel. Commun.*, *12*, 1958–1967.

Perez-Ramirez, J., Borah, D., & Voelz, D. (2013). Optimal 3-d landmark placement for vehicle localization using heterogeneous sensors. *IEEE Trans. Veh. Technol.*, *62*, 2987–2999.

So, H. C. (2011). Source localization: algorithms and analysis. In R. Zekavat, & R. M. Buehrer (Eds.), *Handbook of Position Location. Theory, Practice and Advances* (pp. 25–66). Piscataway, NJ: Wiley-IEEE Press.

Uciński, D. (2005). *Optimal Measurement Methods for Distributed Parameter System Identification*. Boca Raton, FL: CRC Press.

Wieser, A. (2004). Reliability checking for GNSS baseline and network processing. *GPS Solutions*, *8*, 55–66.

Yang, C., Kaplan, L., & Blasch, E. (2012). Performance measures of covariance and information matrices in resource management for target state estimation. *IEEE Trans. Aerosp. Electron. Syst.*, *48*, 2594–2613.