A Nearly Optimal Landmark Deployment for Indoor Localisation with Limited Sensing

Valerio Magnago, Luigi Palopoli, Roberto Passerone
Dep. of Information Engineering and Computer Science
University of Trento, Trento, Italy
E–mail: {valerio.magnago,luigi.palopoli,roberto.passerone}@unitn.it

Daniele Fontanelli, David Macii
Dep. of Industrial Engineering
University of Trento, Trento, Italy
E–mail: {daniele.fontanelli, david.macii}@unitn.it

***Abstract*—Indoor applications based on vehicular robotics require accurate, reliable and efficient localisation. In the absence of a GPS signal, an increasingly popular solution is based on fusing information from a dead reckoning system that utilises on-board sensors with absolute position data extracted from the environment. In the application considered in this paper, the information on absolute position is given by visual landmarks deployed on the floor of the environment considered. This solution is inexpensive and provably reliable as long as the landmarks are sufficiently dense. On the other hand, a massive presence of landmark has high deployment and maintenance costs. In this paper, we build on the knowledge of a large number of trajectories (collected from environment observation) and seek the optimal placement that guarantees a localisation accuracy better than a specified value with a minimal number of landmarks. After formulating the problem, we analyse its complexity and describe an efficient greedy placement algorithm. Finally, the proposed approach is validated in realistic use cases.**

Keywords*—Indoor localisation, position tracking, landmark placement, optimisation.

I. Introduction

Indoor positioning is a well-known research topic, extensively studied over the last few years, and with a wide range of possible applications. Two important examples are assisted living (AAL), robotics and customer guidance in public spaces. While the research in this field is strongly driven by the requirements of smart consumer devices, particularly using wireless techniques based on time-of-flight (ToF) and/or radio signal strength intensity (RSSI) measurements [1]. Since Global Navigation Satellite Systems (GNSS) are clearly impractical [2], a one-size-fit-all solution for indoor localisation and positioning does not exist [3]. At the moment, the most flexible solutions rely on multi-sensor data fusion algorithms [4], [5], [6]; particularly those combining ego-motion relative (e.g. dead reckoning) techniques with distance and heading values measured with respect to "anchor nodes", "tags", "markers" or "landmarks" having known coordinates in a given "absolute" reference frame.

A common problem with this type of approaches concerns the definition of criteria to place such landmarks in the environment, in such a way as to secure a good localisation accuracy avoiding over-design of the infrastructure to be deployed. This can be considered as a subclass of the landmark selection problem addressed in the literature of robotics using online [7], [8] or offline [9], [10] approaches. As pointed out in [11], the offline approach corresponds to the landmark deployment problem considered in this paper. In many research works, the problem of landmark placement is addressed only heuristically, i.e. through common-sense approaches depending on the specific features of the experimental setup considered [12]. The problem becomes very challenging when an optimal deployment is considered. The accuracy and the detection area of the sensors employed, the trajectories of the target to be tracked and the geometry of the environment makes the problem NP-complete [11]. In spite of this, a reliable indoor positioning system has to ensure results with an adequate level of confidence in all conditions [5]. The minimum uncertainty is achieved when an absolute reference, e.g. a landmark, is detected by the sensor at any time [13]. In general, if a target accuracy greater than the minimum achievable has to be guaranteed, a cost index should be suitably defined, for example, using the conditional mutual information [11], the vehicle robot belief [14] or a function of the a-priori covariance matrix in a Kalman filter [15], which is close to the metric chosen in this work. Of course, since the ultimate goal is to minimise the number of deployed landmarks detected using a sensor with a limited detection area, the data about absolute position and orientation are intrinsically intermittent: the robot moves using dead reckoning until it detects a landmark.

Another important feature of optimal landmark placement is related to the availability of the target trajectories. Solutions that work without any knowledge of the target trajectories give effective guarantees, but may be over-conservative in real scenarios [9], [13]. On the other hand, the trajectory knowledge may be stochastic [11], [16] or deterministic [15], [14], as the case considered in this paper.

In this paper, starting from the general result reported in [13], the optimisation problem is cast on a discrete set assuming that the sensor detection area is limited and anisotropic. In addition, the optimisation procedure takes into account the whole set of trajectories available in the environment. In particular, in our case, the trajectories are either generated by robotic planners or result from direct observations. This idea is quite different with respect to other approaches [15], [14], and it is beneficial because it paves the way to the application of a computationally light greedy algorithm. The results of this algorithm are also compared with the lower bound given by the solution of an optimal relaxed problem. The greedy solution is remarkably effective, especially in large environments. Notice that a greedy solution has been presented also in [11]. However in that case a single trajectory is considered on a continuous

: 978-1-5090-6299-7/17/$31.00 © 2017 European Union

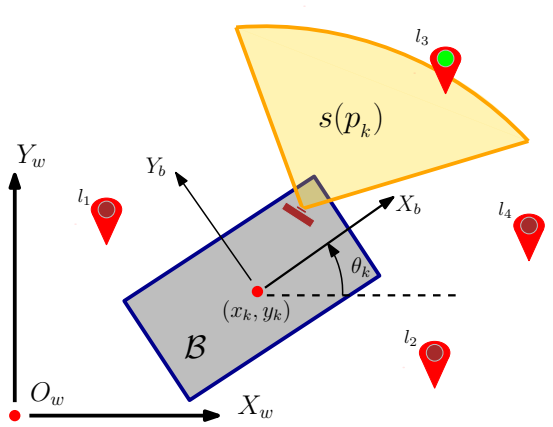


Fig. 1. Mechanical platform represented as a generic rigid body $\mathcal{B}$ moving on the $X_w \times Y_w$ plane with an attached reference frame $\langle B \rangle$. Landmarks l_1, l_2, l_3 and l_4 are also represented. In particular l_3 is inside the SDA $s(p_k)$.

solution set, which implies a numeric approximation with Monte Carlo methods.

The rest of the paper is structured as follows. Section II provides an overview of background models and formalises the problem properly. Section III describes the optimal placement strategy. Section IV shows how the proposed approach could be used in a specific case study: the context of the EU project ACANTO. Section V reports some meaningful simulation results. Finally, Section VI concludes the paper and outlines future work.

II. Models and Problem Formulation

This section presents the reference model adopted for localisation as well as the formalisation of the optimal problem. To let the discussion be more general, we will not make any specific assumption on the estimation algorithm adopted for localisation as long as it is able to estimate the uncertainty associated with the state of the target, as it typically happens with classic position estimation algorithms such as Extended Kalman Filters, H_∞ filters or particle filters [15], [14].

A. Platform Model

Our approach can be applied to a generic platform moving on a horizontal plane in an indoor environment, using both relative data for dead reckoning and landmark measures for absolute references [5]. Examples of the two classes are wheels encoders or IMU for endogenous relative sensing systems and laser scanners, cameras or RFID readers for exogenous absolute readings.

The platform localisation is defined within a fixed right-handed world reference frame $\langle W \rangle = \{O_w, X_w, Y_w, Z_w\}$, as shown in Fig. 1. The mechanical platform is regarded as a rigid body $\mathcal{B}$ moving on the $X_w \times Y_w$ plane. Denoting by t_s the sampling period of the onboard sensors, the generalised coordinates at time kt_s are given by $p_k = [x_k, y_k, \theta_k]^T$, where (x_k, y_k) are the coordinates of the origin of the frame $\langle \mathcal{B} \rangle = \{O_b, X_b, Y_b, Z_b\}$ attached to the rigid body, while θ_k is the angle between X_b and X_w, as depicted in Fig. 1. In this paper, the general class of drift–less, input–affine mechanical platforms are considered, which covers the majority of the wheeled vehicles commonly in use in indoor environments. The kinematic model can then be represented with a discrete–time system as

$$\begin{cases} p_{k+1} = p_k + G_k(p_k)(u_k + \epsilon_k) \\ z_k = h(p_k) + \eta_k \end{cases} \quad (1)$$

where u_k is the piece-wise input vector of the system between $(k-1)t_s$ and kt_s, ϵ_k is the zero mean input uncertainty term, and $G_k(p_k)$ is the generic input vector field. Furthermore, z_k is the vector of measurement data collected at time kt_s, $h(p_k)$ denotes a generic nonlinear output function of the state and η_k is the vector of the zero-mean uncertainty contributions. If the position of the robot is estimated by integrating the endogenous measurements only (dead reckoning), the accumulation of the random noise ϵ_k unavoidably causes large position and orientation uncertainty after a while.

As stated before, the platform is assumed to be equipped with sensors detecting artificial landmarks placed at known positions in $\langle W \rangle$. We assume that the sensor detection area (SDA), denoted as $s(p_k)$ for a particular position p_k, is limited in both range and angular aperture, as depicted in Fig. 1. We denote the space reachable by the platform inside the environment as $\mathcal{Q} \subseteq \mathbb{R}^2 \times [0, 2\pi)$, assuming that $p_k \in \mathcal{Q}$ $\forall k$. Furthermore, we denote with $\mathcal{D}$ the detectable area, i.e., the points that are in the SDA from at least one position p_k:

$$\mathcal{D} = \left\{ (x, y) \in \mathbb{R}^2 \mid \exists p_k \in \mathcal{Q}, (x, y) \in s(p_k) \right\},$$

and with $\mathcal{L}_p \subseteq \mathcal{D}$ the area in which it is possible to place landmarks.

B. Problem Formulation

The objective of the proposed solution is to minimise the number of artificial landmarks to be deployed in the environment of interest in order to meet a given maximum localisation uncertainty $\xi(p_k)$. This scalar value can be a function of the actual platform position to guarantee application-oriented constraints. For instance, for safe navigation, positions closer to walls need a smaller target uncertainty. If $P_k \in \mathbb{R}^{3\times3}$ denotes the covariance matrix of the localisation error associated with p_k at time kt_s, the actual localisation uncertainty can be regarded as a scalar function of P_k, i.e., $f(P_k)$. We assume that whenever a landmark is detected, the uncertainty of the platform is set equal to the measurement uncertainty associated with the system used for landmark detection (so no coherent fusion is assumed, which results in a precautionary intake), i.e. $f(P_k) = g(R)$, where R is the covariance matrix of the position measurements based on landmark detection and $g(\cdot)$ is a scalar function homologous to $f(\cdot)$. Of course, if we have just a single type of sensors for landmark detection and landmarks are just sporadically detected then $f(P_k) \geq g(R)$, $\forall k$.

To design an effective solver and to ease the practical deployment, we limit the positions in which it is possible to place a landmark to a finite set $\mathcal{L}_f \subseteq \mathcal{L}_p$, where $\mathcal{L}_f$ should be chosen not to artificially constrain the solution. For this reason, the finite set $\mathcal{L}_f$ should be such that we can still reach the minimum possible target uncertainty, i.e., $\xi(p_k) = g(R)$, $\forall p_k \in \mathcal{Q}$. In other words, we require that for every position

p_k there is at least one possible landmark position in its SDA. Formally,

$$\mathcal{L}_f \cap s(p_k) \neq \emptyset, \forall p_k \in \mathcal{Q}.$$

In fact, placing a landmark in each location $\mathcal{L}_f$ would guarantee $f(P_k) = \xi(p_k) = g(R)$, $\forall k$. Moreover, even if not strictly needed, the number of finite locations, i.e., the cardinality $|\mathcal{L}_f|$ of $\mathcal{L}_f$, should be as small as possible in order to reduce the search space. In our previous work [13], which assumes that the platform is equipped with a vision system, we found a closed-form geometric solution to this minimisation problem, expressed as follows:

Problem 1: Given $\mathcal{Q}$ and $s(\cdot)$, find

$$\mathcal{L}_f = \arg\min_{\mathcal{L}_x} \mathcal{L}_x \text{ s.t.} \quad \forall p_k \in \mathcal{Q},\ \mathcal{L}_x \cap s(p_k) \neq \emptyset \wedge \mathcal{L}_x \subseteq \mathcal{L}_p.$$

To give deterministic guarantees about the target uncertainty $\xi(p_k)$, some information on the platform trajectories is needed. Trivially, at least one landmark should be detected along each trajectory to avoid unbounded uncertainty growth due to dead reckoning. Usually, if autonomous vehicles are considered, the set of trajectories are finite and well defined. However, if the mechanical platform is driven by a human being (e.g. in the case of robotic trolleys in factory floors or robotic walkers for seniors as in the European project ACANTO [17], [18]), observations about the typical trajectories in the indoor environment are needed. In this paper, we will refer to $\mathcal{T}$ as the set of all the available trajectories, where $T_i \in \mathcal{T}$ refers to the i–th trajectory. We are now in a position to clearly state the problem at hand.

Problem 2: Given $\mathcal{Q}$, $\mathcal{L}_f$, $\mathcal{T}$ and $\xi(p_k) \geq g(R)$, $\forall p_k \in \mathcal{Q}$, find:

$$\mathcal{L} = \arg\min_{\mathcal{L}_x} |\mathcal{L}_x| \text{ s.t.} \quad \mathcal{L}_x \subseteq \mathcal{L}_f, \quad \forall i\ T_i \in \mathcal{T}, \forall k\ p_k \in T_i,\ f(P_k) \leq \xi(p_k).$$

The problem is well-posed since a solution always exists by definition, i.e. $\mathcal{L} = \mathcal{L}_f$.

The set of available trajectories $\mathcal{T}$ can be conveniently represented using the set $\mathcal{L}_f$. Indeed, $\forall i, k$, $\exists p_k \in T_i : \mathcal{S}_{i,k} = s(p_k) \cap \mathcal{L}_f \neq \emptyset$. This way, the continuous trajectory T_i can be represented with a *quantised* trajectory $\mathcal{S}_{i,k}$ induced by $\mathcal{L}_f$. Notice that the mapping between $p_k \in T_i$ and $\mathcal{S}_{i,k}$ is *not* bijective, i.e. multiple landmarks can be potentially in view from the same platform position p_k.

III. Optimal Landmark Placement

In this section we discuss how to solve Problem 2 by casting it into a binary programming problem, which can be tackled with different solution strategies.

A. CNF Problem Representation

To represent the problem, we associate to each possible landmark location $l_i \in \mathcal{L}_f$ a boolean variable a_i, such that

$$a_i = \begin{cases} 1, & \text{if a landmark is placed in } l_i, \\ 0, & \text{otherwise.} \end{cases}$$

Thus, a landmark deployment corresponds to an assignment to the boolean variables. The objective is to find a *least* assignment, i.e., an assignment such that the minimum number of variables is assigned the value 1, which satisfies the uncertainty constraints. We model the constraints by identifying all the partial assignments to the variables that lead to a violation. Consider a position $q_s \in T_i$, and assume $f(P_s) = g(R)$, i.e., the minimum uncertainty in our setting. We simulate the trajectory and compute the evolution of $P_{s+1}, P_{s+2}, \dots$ along T_i. At the same time, we keep track of the landmark positions $\mathcal{S}_{i,j}$ in view along the simulated path. If at time $k+1 > s$, $f(P_{k+1}) > \xi(p_{k+1})$, then we have a violation. In order to avoid it, at least one landmark must be present in one of the positions $\cup_{j=s}^{k} \mathcal{S}_{i,j}$ in view. This condition can be expressed as follows, i.e.

$$\omega_{i,s} = \bigvee_{j=s}^{k} \mathcal{S}_{i,j},$$

where, with a slight abuse of notation, the boolean variables associated with the landmark positions are denoted with $\mathcal{S}_{i,j}$. Clearly, a landmark deployment $\mathcal{L}$ that does not satisfy $\omega_{i,s}$ cannot be a solution of Problem 2, since between p_k and p_{k+1} the uncertainty constraint would be violated. We can repeat this analysis for all starting positions and all trajectories, and collect the clauses in a set Ω. For the problem to be satisfied, it is necessary and sufficient that all the generated clauses evaluate to true. Thus, the function

$$\varphi(a_i, \dots, a_n) = \bigwedge \Omega = \bigwedge_{i,s} \omega_{i,s}$$

evaluates to true for all and only those assignments to the boolean variables $a_1, \dots, a_n$ which correspond to a correct deployment. Given its form, φ is expressed in Conjunctive Normal Form (CNF). For example, with reference to Figure 2, from $q_4 \in T_3$, the platform sees 5 landmarks before $f(P_{13}) > \xi(q_{13})$. The set of landmarks in view is given by $\bigvee_{j=4}^{13} \mathcal{S}_{3,j} = \{l_1, l_3, l_5, l_7, l_{10}\}$, the corresponding clause $\omega_{3,4}$ is:

$$\omega_{3,4} = a_1 \vee a_3 \vee a_5 \vee a_7 \vee a_{10}.$$

Since the clauses represent a disjoint operation, a cardinality reduction of the set Ω is convenient. For example, for the following two clauses

$$\omega_{3,4} = a_1 \vee a_3 \vee a_5 \vee a_7 \vee a_{10},$$
$$\omega_{3,5} = a_3 \vee a_5 \vee a_7 \vee a_{10},$$

we have that $\omega_{3,5} = 1 \Rightarrow \omega_{3,4} = 1$ but $\omega_{3,4} = 1 \not\Rightarrow \omega_{3,5} = 1$. Thus, only $\omega_{3,5}$ is of relevance for the placement, while $\omega_{3,4}$ can be safely removed and hence reduce the complexity.

A compact representation of Ω is given by a coverage matrix whose columns are the possible landmarks locations $l_i \in \mathcal{L}_f$ and rows are the clauses $\omega_{i,s}$. The entry in position (r, c) of such a matrix has 1 if the r-th clause is satisfied by the c-th landmark, or 0 otherwise. An example is shown in Table I.

B. Optimal Placement

As discussed, to optimise the placement we need to find the least satisfying assignment, i.e., an assignment to the variables $a_1, \dots, a_n$ such that φ is true and the least number of variables

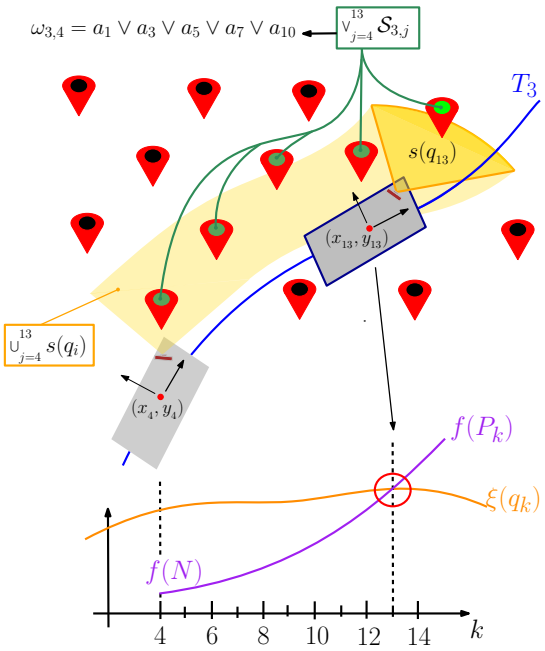


Fig. 2. Example of uncertainty growth along a sample trajectory.

TABLE I. COVERAGE MATRIX EXPRESSING THE CLAUSE AS DISJUNCTION OF BOOLEAN VARIABLES: $\omega_{2,3} = a_1 \vee a_2 \vee a_8 \vee a_9$; $\omega_{4,1} = a_2 \vee a_3 \vee a_6$; $\omega_{3,2} = a_2 \vee a_4$; $\omega_{3,4} = a_1 \vee a_3 \vee a_5 \vee a_7 \vee a_{10}$; $\omega_{3,5} = a_3 \vee a_5 \vee a_7 \vee a_{10}$.

	1	2	3	4	5	6	7	8	9	10
$\omega_{2,3}$	1	1	0	0	0	0	0	1	1	0
$\omega_{4,1}$	0	1	1	0	0	1	0	0	0	0
$\omega_{3,2}$	0	1	0	1	0	0	0	0	0	0
$\omega_{3,4}$	1	0	1	0	1	0	1	0	0	1
$\omega_{3,5}$	0	0	1	0	1	0	1	0	0	1

is assigned value 1. There are several ways to formally solve this problem. One approach is to cast it as a logic optimisation problem, and look for a minimum term cover of φ. Observe that the conjunction of the true variables of a satisfying assignment is an implicant of φ. For instance, let $I = \{i_1, \ldots, i_t\}$ be the indices of the true variables of a satisfying assignment. Then, the product term $a_{i_1} \cdot a_{i_2} \cdots a_{i_t}$ logically implies φ, that is, the product term "covers" some of the ones of φ. A *minimal* deployment (i.e., one in which no landmark can be removed without violating the constraints) corresponds to a *prime* implicant of φ. The *minimum* deployment is therefore the *largest* prime implicant.

We thus use a logic optimisation program to find a minimum 2-level cover of φ. Each term of the resulting cover corresponds to a minimal deployment, and we choose the one with the least number of variables. This approach has the advantage that it provides several alternative solutions, corresponding to the various terms of the cover. In our experiments we have used the SIS optimisation software [19]. While this strategy gives us the best solution, the downside lies in its computational complexity, which is exponential in the number of variables and in the number of prime implicants. Our experiments show that the method is practical only in the case of deployments of a limited size. For instance, a layout with 37 locations and 10 constraints is solved in less than a second on a 3.2 GHz Intel Xeon PC with 4 GB of RAM, but already results in almost 8,000 minimal solutions, with the best ones (around 1,000 solutions) using just 4 landmarks. The extension of the same problem to 52 locations and 15 constraints increases the computation time to over 7 minutes, and almost 250,000 minimal solutions; 23 of them use 4 landmarks, while the largest minimal solutions rely on 11 landmarks. Therefore, this approach is impractical for larger deployments.

Alternatively, the problem can be rephrased as a constrained boolean optimisation, i.e.,

$$\min \sum_i a_i, \quad \text{subject to} \quad \forall i, \forall s, \quad \omega_{i,s} > 0$$

Even if the computational complexity of the problem is still exponential, one can solve the continuous relaxation of the same problem, which is polynomial. Of course, since in this case the variables may take any value between 0 and 1, the solution of the problem in general will be infeasible. Despite this, the relaxed optimal solution (that henceforth will be denoted with $\mathcal{L}_r$) provides a lower bound to the number of landmarks which are required to satisfy the constraints. In the following, we will use the result of this approach to evaluate the performance of the greedy placement algorithm.

C. Greedy Placement

The greedy algorithm for landmark placement leads to a good approximation of the optimal solution within a negligible computation time. It is based on the greedy heuristic for submodular functions described in [20]. In practice, we start with the coverage matrix A_0, computed as described previously, where the columns are ordered with a decreasing number of elements equal to 1. With reference to Table I, the first column will be l_2, then l_3 and so on. A landmark is placed in the position corresponding to the first column, i.e., the one satisfying the greatest number of clauses. The corresponding satisfied clauses (the matrix rows) are then removed from the matrix, together with the first column, and the matrix is reordered. With reference to Table I, l_2 is added to $\mathcal{L}_g$ and the first three rows are removed. A new matrix A_1 is obtained, and the procedure starts over. The procedure ends when there are no more clauses to meet, i.e., when the matrix is empty. For the case of Table I, the procedure may end with $\mathcal{L}_g = \{l_2, l_5\}$ or with $\mathcal{L}_g = \{l_2, l_3\}$, namely when at most two landmarks are placed. As shown in Section V, despite its simplicity, the greedy solution $\mathcal{L}_g$ turns out to be very effective when compared to the (infeasible) lower bound $\mathcal{L}_r$. As a final remark, we notice that the lower the number of boolean variables shared between clauses, the more the greedy suboptimal solution approaches the best possible one.

IV. CASE STUDIES

This section presents the robotic platform to be localised and the uncertainty function $f(P_k)$ chosen to run meaningful simulations in a realistic environment.

A. Reference Platform

The reference platform is the *FriWalk* (Fig. 3), a service robot developed in the European project ACANTO [17] and

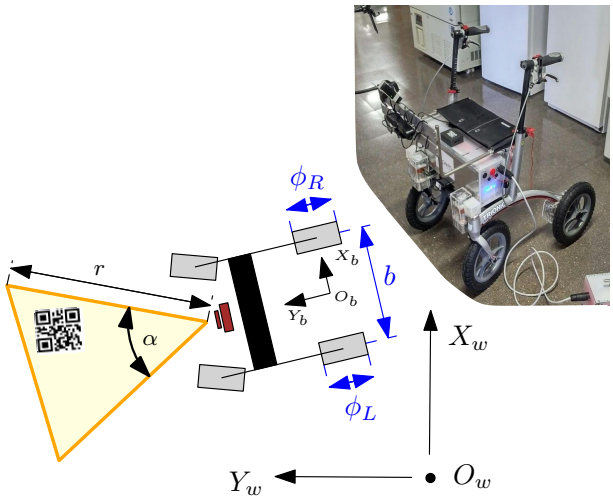


Fig. 3. The *FriWalk* schematic representation and SDA.

provided with cognitive [18], [21] and guidance functions [22], [23]. The *FriWalk* is based on a standard commercial walker[1], it is equipped with relative encoders on the rear wheels and has a front monocular camera used to detect Quick Response (QR) codes placed on the floor. The *FriWalk* follows a unicycle-like dynamics [18]. The robot planar coordinates (x_k, y_k) correspond to the mid-point of the rear wheels axle. This point coincides with the origin of the body frame O_b with the X_b axis pointing forward, as depicted in Fig. 3. With reference to Fig. 1, the robot generalised coordinates are $p_k = [x_k, y_k, \theta_k]^T$, The camera measures the relative position and orientation of the walker with respect to the QR codes, i.e., the visual landmarks to be placed. The main parameters of the SDA (which in this case coincides with the camera field of view) are the camera range r and its aperture angle α, as shown in Fig. 3. Once r and α are known, the set $\mathcal{L}_f$ can be analytically determined [13]. By knowing the position of each landmark in the environment, a measure of the entire state p_k is given with covariance R. Recall that in this paper we do not consider an estimator that coherently fuses the available measures (thus decreasing the localisation uncertainty), as for example in [5], [15].

To model the uncertainty growth when no landmark is detected (i.e. when just the rear encoders are used for odometry), variables δ_k^r and δ_k^l are used to express the angular displacements of the right and left wheels, respectively, in the time interval $[kt_s, (k+1)t_s]$. As a consequence, the right (or left) wheel linear displacement in one sampling period are given by $\frac{\phi_r}{2}\delta_k^r$ (or $\frac{\phi_l}{2}\delta_k^l$), where ϕ_r and ϕ_l are the wheel diameters. With respect to the general model (1) and recalling that the vehicle is a unicycle-like vehicle, we have

$$G_k(p_k) = \begin{bmatrix} \frac{\phi_r}{4}\cos\theta_k & \frac{\phi_l}{4}\cos\theta_k \\ \frac{\phi_r}{4}\sin\theta_k & \frac{\phi_l}{4}\sin\theta_k \\ \frac{\phi_r}{2b} & -\frac{\phi_l}{2b} \end{bmatrix}, \tag{2}$$

where b is the rear inter-axle length (Fig. 3). Thus, the system inputs can be expressed as $u_k = [\delta_r^k, \delta_l^k]^T$. The additive input noise ϵ_k is distributed according to a stationary Gaussian process with a 2×2 diagonal covariance matrix E whose diagonal elements σ_r^2 and σ_l^2 are the noise variances due to the finite resolution and tick reading errors of either encoder. The measurement function is instead $h(p_k) = p_k + \eta_k$, where η_k is the zero-mean normally distributed measurement uncertainty vector. If we assume that the measurement uncertainty contributions are uncorrelated, the corresponding covariance matrix is $R = \text{diag}(\sigma_x^2, \sigma_y^2, \sigma_\theta^2)$.

Before modelling how the localisation error grows when no landmarks are in the SDA, we further consider the uncertainties affecting vehicle parameters, here collected in the vector $\lambda = [\phi_R, \phi_L, b]^T$. For this constant, but possibly uncertain parameters, we assume Gaussian uncorrelated distributions, collected in the 3×3 diagonal covariance matrix C whose entries are $\sigma_{\phi_r}^2$, $\sigma_{\phi_l}^2$ and σ_b^2. By defining $q_k = [p_k^T, \lambda^T]^T$ and recalling (2), the final model is

$$q_{k+1} = q_k + G_k^\star(q_k)(u_k + \epsilon_k),$$

where $G_k^\star(q_k) = [G_k(q_k)^T, 0]^T$. The uncertainty growth

$$Q_{k+1} = \mathrm{E}\left\{(q_{k+1} - \mathrm{E}\{q_{k+1}\})(q_{k+1} - \mathrm{E}\{q_{k+1}\})^T\right\},$$

where $\mathrm{E}\{\cdot\}$ is the expectation operator, results from the linearisation of model (2) around the estimated state, in accordance with the so-called law of propagation of uncertainty for the multivariate case [24]. Thus, assuming that ϵ_k is uncorrelated from q_k $\forall k$, it follows that

$$Q_{k+1} \approx \left(I + \frac{\partial G_k^\star(q_k)u_k}{\partial q_k}\right) Q_k \left(I + \frac{\partial G_k^\star(q_k)u_k}{\partial q_k}\right)^T + G_k^\star(q_k)\, E\, G_k^\star(q_k)^T.$$

Notice that P_{k+1}, i.e., the localisation error covariance matrix, is the upper 3×3 matrix of Q_{k+1}. So, at the beginning of each simulation we set

$$Q_0 = \begin{bmatrix} R & 0 \\ 0 & C \end{bmatrix}.$$

B. Target Uncertainty

Since $P_k \in \mathbb{R}^{3\times3}$ is the covariance matrix of the localisation error associated with p_k at time kt_s, in our experiments the actual localisation uncertainty metric is given by

$$f(P_k) = \max \text{Eig}\,(P_k^{x,y}), \tag{3}$$

where $P_k^{x,y}$ refers to the upper 2×2 matrix of P_k, i.e., the localisation errors along X_w and Y_w, respectively, and the operator $\text{Eig}(M)$ returns the eigenvalues of matrix M. With this choice, a conservative assumption is made since the ellipsoid is approximated by the circumscribing circle (as in [15]). Finally, notice that, since the output function just returns p_k, then $g(\cdot)$ is the same function as $f(\cdot)$.

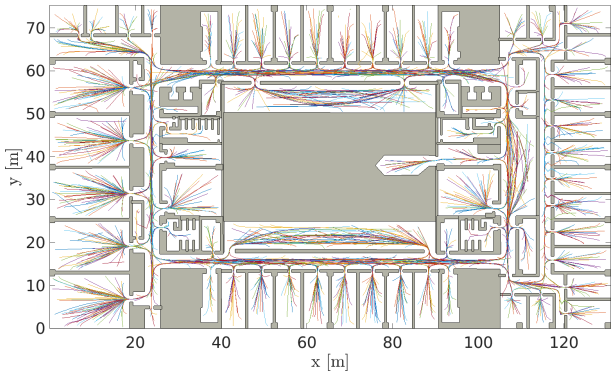
V. Simulation Results

This section presents the simulation results in different scenarios. Throughout this section, the model is the *FriWalk* with the parameters reported in Table II. For the sake of brevity, only the results with a constant target uncertainty are reported, i.e., $\xi(p_k)$ is constant for all p_k.

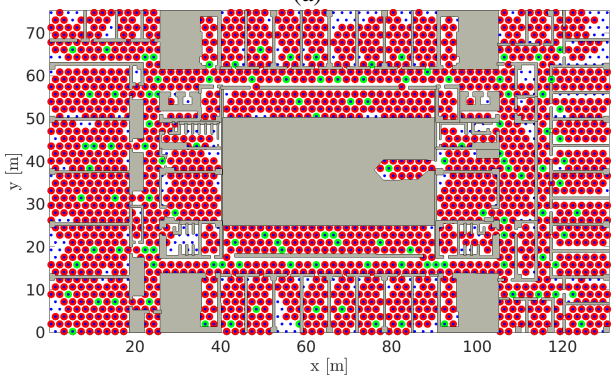
[1]Trionic Walker 12er

TABLE II. NUMERICAL VALUES ADOPTED IN THE SIMULATIONS, DERIVED FROM THE *FriWalk*

ϕ_R	150 mm	ϕ_L	150 mm	b	800 mm	t_s	10 ms
σ_r	4 mrad	σ_l	4 mrad	r	4 m	α	$\pi/3$ rad
σ_x	50 mm	σ_y	50 mm	σ_θ	$\frac{5\pi}{180}$ rad	σ_{ϕ_r}	5 mm
σ_{ϕ_l}	5 mm	σ_b	10 mm	$\xi(p_k)$	0.64 m^2		



(a)



(b)

Fig. 4. DISI scenario for vehicle trajectories generated with the chosen planner for robots. (a) 800 paths considered for the landmark placement. (b) potential QR codes locations (dots), QR codes locations detected from at least one trajectory (circled dots) and QR deployment with the greedy algorithm (green circled dots).

A. Realistic Environment

The realistic environment chosen for simulation purposes is the Department of Information Engineering and Computer Science (DISI) of the University of Trento. The *FriWalk* trajectories are generated using the path planner described in [25], which is conceived for robots moving in known structured environments. In this case, the set of trajectories is quite repetitive and regular, and robots moving in the corridor are likely to follow the same path, as clearly visible in Figure 4-(a). The regularity of the paths increases the number of shared boolean variables between the clauses, making this a very challenging situation for the greedy algorithm. If we assume to use a visual sensor with the values of r and α as reported in Table II, the potential positions of QR codes determined as described in [13] amounts to $|\mathcal{L}_f| = 2085$. Such positions are represented with blue dots in Figure 4-(b). Considering 800 different paths, randomly generated by the path planner and depicted in Figure 4-(a), 1889 potential QR code landmarks are observed at least once in at least one trajectory. The positions of these landmarks are highlighted with circled dots in Figure 4-(b). By solving the relaxed optimisation problem,

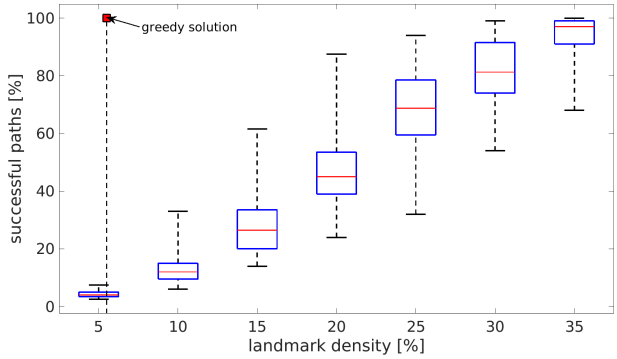


Fig. 5. Percentage of path satisfying the maximum uncertainty limit $\xi(p_k)$ (vertical axis) against the percentage of QR landmarks randomly placed with respect to $|\mathcal{L}_f|$ (horizontal axis) for the DISI scenario reported in Figure 4. The vertical thick line corresponds to the greedy solution, while the square on top recall that no path violates $\xi(p_k)$.

assuming $a_i \in [0,1] \subset \Re$ (see Section III-B), the overall optimal number of landmarks is $m_b = \sum_i a_i = 92.6$, which is also a lower bound for the optimal solution. To obtain a feasible deployment from this optimal infeasible solution, we first arrange the values of a_i in descending order. Then we place a landmark in the positions with the highest value (saturating a_i to 1), and then we continue to add landmarks in $\mathcal{L}_r$ until all the clauses are satisfied. In this way, the total number of landmarks is $M_b = |\mathcal{L}_r| = 133$, which is an upper bound of the optimal solution. The greedy algorithm instead leads to the selection of $|\mathcal{L}_g| = 115$ QR codes, i.e. which is included between M_b and m_b bounds. Such landmarks are represented with green circled dots in Figure 4-(b).

Notice that even if the number of trajectories and of potential landmark locations is quite large, the computation time of the greedy algorithm implemented in Matlab and running on a 3.50 GHz Intel Core i7 with 8 GB of RAM is about 15 minutes. In addition, we compared the greedy solution with the result of a naive approach in which different amounts of QR codes are randomly selected from $\mathcal{L}_f$. In particular, between 5% and 35% of possible landmark positions have been chosen repeatedly (i.e. 50 times) with the same probability. For each random placement the percentage of paths satisfying the maximum uncertainty limit $\xi(p_k)$ has been estimated. The results are summarised in Figure 5. The boxes define the 25-th and 75-th percentile, while the whiskers corresponds to the maximum and minimum value. The thick vertical line corresponds to the percentage of QR landmarks placed by the greedy algorithm for which all the paths meet the given uncertainty constraint, i.e. $\xi(p_k) = 0.64$ m^2. It is worth noticing how the greedy solution outperforms the naive random choice. The localisation uncertainty obtained with greedy and random placement over 800 trajectories are summarised in Table III, where the maximum, the average and the standard deviation of (3) are reported. Observe that the greedy algorithm ensures a very good accuracy, even if only 5.5% of QR codes is used (see the thick line in Figure 5).

The results of landmark deployment for more realistic, i.e. human-like, trajectories is reported in Figure 6. 800 human-like trajectories in a corridor have been synthesised using the Headed Social Force Model (HSFM) [26]. This model emulates the motion of human beings moving in shared spaces

TABLE III. MAXIMUM, AVERAGE AND STANDARD DEVIATION OF LOCALISATION UNCERTAINTY (3) FOR RANDOM AND THE GREEDY PLACEMENT, RESPECTIVELY. ALL SIMULATION RESULTS REFER TO THE REALISTIC SCENARIO SHOWN IN FIGURE 4.

	Random deployment densities					
	5%	25%	45%	65%	85%	greedy (6%)
max [m]	35	5.3	1.6	0.8	0.4	0.79
mean [m]	2.8	0.3	0.1	0.07	0.06	0.14
std [m]	4.4	0.5	0.1	0.05	0.1	0.10

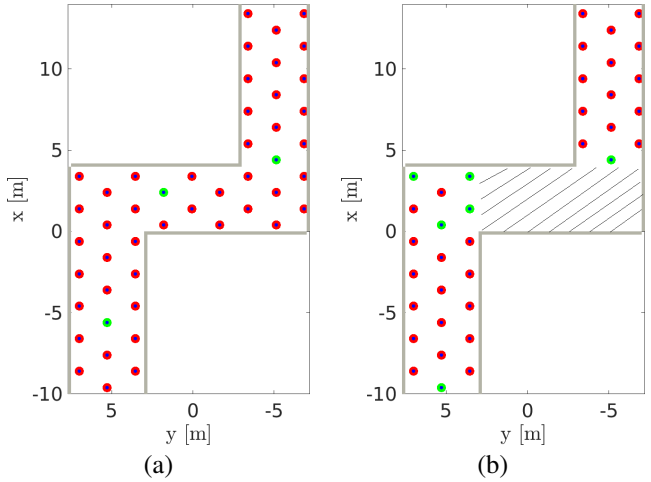


Fig. 6. Corridor scenario for trajectories generated with the HSFM [26]. QR codes locations (dots), QR codes locations detected from at least one trajectory (circled dots) and QR deployment with the greedy algorithm (green circled dots) are reported. (a) Deployment without placement constraints and (b) deployment considering an area where the QR codes cannot be placed.

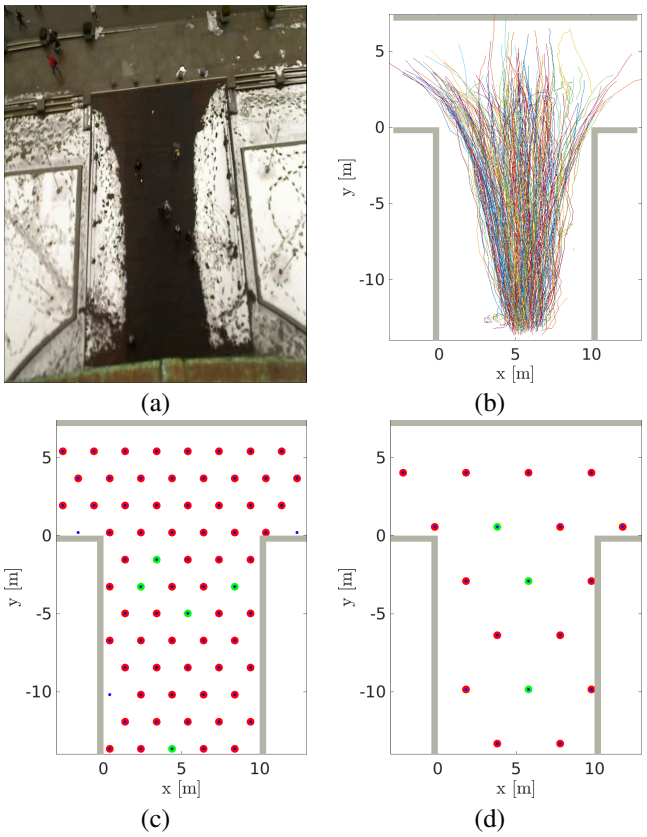


Fig. 7. Simulation on actual data. (a) ETH Zurich building entrance. (b) measured paths [27]. (c) deployment of 10 QR landmarks for the greedy algorithm. (d) deployment of 6 QR landmarks for the greedy algorithm when the SDA range doubles.

and obeys to the kinematic model that falls in the generic representation of (1). For this case, we report in Figure 6 two different landmark deployments, both based on the results of [13], which gives $|\mathcal{L}_f| = 48$ possible landmark positions if landmark deployment is unconstrained, reported in Figure 6-(a) with blue dots, and instead $|\mathcal{L}_f| = 36$ where there is an area where QR codes cannot be placed, shown in Figure 6-(b), again in blue dots. Notice that in the latter case the landmark placement in [13] cannot be strictly satisfied in the QR forbidden region. For the unconstrained deployment, the upper and lower bounds to the optimal number of deployed landmarks are $M_b = 3$ and $m_b = 3$, respectively. The proposed greedy placement algorithm returns a solution with $|\mathcal{L}_g| = 3$ QR codes, i.e. green dots in Figure 6-(a). On the contrary, for the constrained scenario, the greedy solution places $|\mathcal{L}_g| = 6$ QR codes with bounds $M_b = 6$ and $m_b = 5$.

For the empirical validation of both the placements of Figure 6, we simulate 200 additional and independent paths considering multiple persons moving simultaneously in the corridor, hence no knowledge of the trajectory is available upfront. In both cases, the localisation accuracy based on the greedy placement meets the given uncertainty constraint $\xi(p_k)$ with 99.5% probability.

B. Real trajectories

As a further validation of the proposed solution in a context similar to the applicative scenario of the ACANTO project [17], 360 paths captured at the entrance of the ETH Zurich building (see Figure 7-(a)) have been used to test the performance of landmark greedy placement [27]. Again, the applicability of model (1) is substantiated by [26]. Hence, we can safely assume that each user drives a *FriWalk*. Figure 7-(b) shows 288 paths extracted from the video footage. With the SDA parameters defined in Table II, we have $|\mathcal{L}_f| = 72$ possible QR locations (blue dots in Figure 7-(c)). In this case, $M_b = 10$ and $m_b = 3.7$, respectively. The greedy algorithm selects $|\mathcal{L}_g| = 5$ landmarks. Using the remaining 72 paths of the available data set, we found the uncertainty constraint $\xi(p_k)$ is met with 98.5% probability. Consider that the larger the SDA, the lower $|\mathcal{L}_f|$ and the more $|\mathcal{L}_g| \to |\mathcal{L}_f|$. For instance, Figure 7-(d) reports the placement results when the SDA range is two times larger than in the previous cases (i.e. $r = 8$ m). In this case, $|\mathcal{L}_f| = 18$, $M_b = m_b = 3$, and the solution of the greedy algorithm converges to $|\mathcal{L}_g| = 3$, as well. Moreover, all the remaining 72 paths meet the uncertainty constraint $\xi(p_k)$. Similar results can be achieved if the growth rate of dead reckoning uncertainty increases. This behaviour suggests that the solutions of greedy and naive random placements become closer and closer (as shown in Figure 5), depending on the ratio between the SDA dimension and the growth rate of dead reckoning uncertainty.

VI. CONCLUSIONS

In this paper, we have addressed the problem of the optimal placement of a minimum number of visual landmarks while ensuring that indoor localisation accuracy meets specified

boundaries. We have cast the problem into the framework of logic synthesis and shown a greedy solution that delivers good performance in realistic use cases.

There are several open points that deserve future investigations. The trajectories collected from surveillance cameras do not have the same level of importance; some of them are frequently taken and some are not: treating the two types of trajectories could be inefficient or could jeopardise the system performance There are two possibilities to approach the problem. The first one is deterministic: we can organise the trajectories in a group of "core trajectories" that have to be covered, while other groups of optional trajectories can be covered on a best effort basis. The extension of the algorithm to this case is currently under way. A different approach is stochastic and it relies on a Markovian chain model described in previous papers [16]. Markovian motion models for the target are potentially more powerful in terms of descriptive power than a mere enumeration of trajectories, but are computationally difficult to treat. The greedy heuristic presented in this paper could help evade the curse of dimensionality of Markov models. Regarding the algorithm solving the placement problem, we are exploring alternative encoding schemes, which could take advantage of the monotonicity of the boolean function, and which use satisfiability-based methods.

ACKNOWLEDGMENT

The activities described in this paper have received funding from the European Union Horizon 2020 Research and Innovation Programme - Societal Challenge 1 (DG CONNECT/H) under grant agreement no. 643644 for the project ACANTO - A CyberphysicAl social NeTwOrk using robot friends. The Authors would like to thank Eng. P. Tomasin for his valuable help in the planner implementation.

REFERENCES

[1] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Trans. Syst. Man Cybern. C, Appl. Rev.*, vol. 37, no. 6, pp. 1067–1080, Nov. 2007.

[2] D. Dardari, P. Closas, and P. M. Djuri, "Indoor tracking: Theory, methods, and technologies," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 4, pp. 1263–1278, Apr. 2015.

[3] L. Mainetti, L. Patrono, and I. Sergi, "A survey on indoor positioning systems," in *Proc. International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, Split, Croatia, Sep. 2014.

[4] A. Colombo, D. Fontanelli, D. Macii, and L. Palopoli, "Flexible indoor localization and tracking based on a wearable platform and sensor data fusion," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 4, pp. 864–876, Apr. 2014.

[5] P. Nazemzadeh, F. Moro, D. Fontanelli, D. Macii, and L. Palopoli, "Indoor positioning of a robotic walking assistant for large public environments," *IEEE Trans. Instrum. Meas.*, vol. 64, no. 11, pp. 2965–2976, Nov. 2015.

[6] D. Ayllón, H. A. Sánchez-Hevia, R. Gil-Pita, M. U. Manso, and M. R. Zurera, "Indoor blind localization of smartphones by means of sensor data fusion," *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 4, pp. 783–794, Apr. 2016.

[7] H. Strasdat, C. Stachniss, and W. Burgard, "Which landmark is useful? Learning selection policies for navigation in unknown environments," in *IEEE Intl. Conf. on Robotics and Automation*. IEEE, 2009.

[8] S. Thrun, "Finding landmarks for mobile robot navigation," in *IEEE Intl. Conf. on Robotics and Automation*, vol. 2. IEEE, 1998.

[9] P. Sala, R. Sim, A. Shokoufandeh, and S. Dickinson, "Landmark selection for vision-based navigation," *IEEE Trans. Robot.*, vol. 22, no. 2, pp. 334–349, Apr. 2006.

[10] L. H. Erickson and S. M. LaValle, *An art gallery approach to ensuring that landmarks are distinguishable*. MIT Press, 2012, pp. 81–88.

[11] M. Beinhofer, J. Müller, and W. Burgard, "Near-optimal landmark selection for mobile robot navigation," in *IEEE Intl. Conf. on Robotics and Automation*. IEEE, 2011, pp. 4744–4749.

[12] A. A. Khaliq, F. Pecora, and A. Saffiotti, "Inexpensive, reliable and localization-free navigation using an RFID floor," in *European Conf. on Mobile Robots (ECMR)*, Lincoln, United Kingdom, Sep. 2015.

[13] P. Nazemzadeh, D. Fontanelli, and D. Macii, "Optimal Placement of Landmarks for Indoor Localization using Sensors with a Limited Range," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Madrid, Spain, Oct. 2016, pp. 1–8.

[14] M. P. Vitus and C. J. Tomlin, "Sensor placement for improved robotic navigation," *Robotics: Science and Systems VI*, p. 217, 2011.

[15] M. Beinhofer, J. Müller, and W. Burgard, "Effective landmark placement for accurate and reliable mobile robot navigation," *ROBOT. AUTON. SYST.*, vol. 61, no. 10, pp. 1060–1069, Oct. 2013.

[16] F. Zenatti, D. Fontanelli, L. Palopoli, D. Macii, and P. Nazemzadeh, "Optimal Placement of Passive Sensors for Robot Localisation," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and System*. Daejeon, South Korea: IEEE/RSJ, Oct. 2016, pp. 4586–4593.

[17] "ACANTO: A CyberphysicAl social NeTwOrk using robot friends," http://www.ict-acanto.eu/acanto, February 2015, EU Project.

[18] L. Palopoli et al., "Navigation Assistance and Guidance of Older Adults across Complex Public Spaces: the DALi Approach," *Intelligent Service Robotics*, vol. 8, no. 2, pp. 77–92, 2015.

[19] E. Sentovich et al., "Sis: A system for sequential circuit synthesis," EECS Department, University of California, Berkeley, Tech. Rep. UCB/ERL M92/41, 1992.

[20] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions—i," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.

[21] P. Bevilacqua, M. Frego, E. Bertolazzi, D. Fontanelli, L. Palopoli, and F. Biral, "Path Planning maximising Human Comfort for Assistive Robots," in *IEEE Conference on Control Applications (CCA)*. Buenos Aires, Argentina: IEEE, Sept. 2016, pp. 1421–1427.

[22] F. Moro, A. D. Angeli, D. Fontanelli, R. Passerone, D. Prattichizzo, L. Rizzon, S. Scheggi, S. Targher, and L. Palopoli, "Sensory stimulation for human guidance in robot walkers: A comparison between haptic and acoustic solutions," in *IEEE International Smart Cities Conference (ISC2)*, Trento, Italy, Sept. 2016, pp. 1–6.

[23] M. Andreetto, S. Divan, D. Fontanelli, and L. Palopoli, "Passive Robotic Walker Path Following with Bang-Bang Hybrid Control Paradigm," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and System*. Daejeon, South Korea: IEEE/RSJ, Oct. 2016, pp. 1054–1060.

[24] ISO/IEC Guide 98-3:2008, *Uncertainty of measurement – Part 3: Guide to the expression of uncertainty in measurement (GUM:1995)*, Jan. 2008.

[25] S. Quinlan and O. Khatib, "Elastic bands: connecting path planning and control," in *[1993] Proceedings IEEE International Conference on Robotics and Automation*, May 1993, pp. 802–807 vol.2.

[26] F. Farina, D. Fontanelli, A. Garulli, A. Giannitrapani, and D. Prattichizzo, "Walking Ahead: The Headed Social Force Model," *PLOS ONE*, vol. 12, no. 1, pp. 1–23, 01 2017.

[27] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in *IEEE 12th Intl. Conf. on Computer Vision*, Sep. 2009, pp. 261–268.