

Infrastructure-less UWB-based Active Relative Localization

Valerio Brunacci, Alberto Dionigi, Alessio De Angelis, and Gabriele Costante

Abstract—In multi-robot systems, relative localization between platforms plays a crucial role in many tasks, such as leader following, target tracking, or cooperative maneuvering. State of the Art (SotA) approaches either rely on infrastructure-based or on infrastructure-less setups. The former typically achieve high localization accuracy but require fixed external structures. The latter provide more flexibility, however, most of the works use cameras or lidars that require Line-of-Sight (LoS) to operate. Ultra Wide Band (UWB) devices are emerging as a viable alternative to build infrastructure-less solutions that do not require LoS. These approaches directly deploy the UWB sensors on the robots. However, they require that at least one of the platforms is static, limiting the advantages of an infrastructure-less setup. In this work, we remove this constraint and introduce an active method for infrastructure-less relative localization. Our approach allows the robot to adapt its position to minimize the relative localization error of the other platform. To this aim, we first design a specialized anchor placement for the active localization task. Then, we propose a novel *UWB Relative Localization Loss* that adapts the Geometric Dilution Of Precision metric to the infrastructure-less scenario. Lastly, we leverage this loss function to train an active Deep Reinforcement Learning-based controller for UWB relative localization. An extensive simulation campaign and real-world experiments validate our method, showing up to a 60% reduction of the localization error compared to current SotA approaches.

I. INTRODUCTION

Localization is a longstanding goal in robotics and is certainly one of the most crucial components of every robotic platform. In the past decades, it has been the object of countless research and, while notable results have been achieved, it still offers important challenges.

At a high level, localization can be achieved by relying either on a fixed infrastructure or solely on onboard robot sensors. The former setup is implemented by coupling external sensors with onboard markers or receivers. This is the case of localization approaches based on Global Navigation Satellite Systems (GNSSs) in outdoor scenarios, while for indoor environments Motion Capture (MoCAP) systems [1] can be used. These solutions provide highly accurate pose estimates, however, the robot is constrained within the range of the external sensor system which, in indoor scenarios, is typically limited to a single room.

Infrastructure-less solutions, on the other hand, do not require external structure at fixed known positions. If one is interested in the localization of a single robot, pose

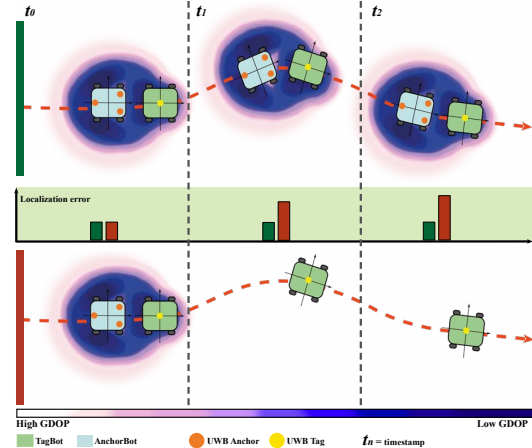


Fig. 1: Overview of the proposed approach. In a UWB-based relative localization context, the performance is affected by the geometrical distribution of the sensors in the environment. Contrary to the SotA approaches (bottom figure with red bar), we develop an active method (top figure with green bar) that exploits the movement of the robot to enhance the localization performance.

estimation can be achieved with respect to a known map, if available, or the global map and the robot position can be estimated jointly by exploiting the popular Simultaneous Localization and Mapping (SLAM) paradigm [2]. In multi-robot systems, instead, *relative localization* between platforms can be exploited to refine their respective global pose estimate, or used for tasks where platforms are mainly interested in their relative pose with respect to another robot (*e.g.*, leader following, target tracking or cooperative maneuvering).

Relative localization can be achieved with various sensor setups, among which the most popular are based on cameras, lasers, and lidars. Typically, these solutions estimate the relative pose by detecting markers mounted on the robots [1] or by inferring pose information from shape, dimensions, or texture captured by the sensors [3]. Despite the results achieved, these approaches suffer from two important limitations: i) through perception algorithms, which might be subject to errors and require tuning procedures; ii) they require Line-of-Sight (LoS) to operate, an assumption often violated due to occlusions and non-ideal environment conditions (*e.g.*, fog or absence of illumination).

Ultra Wide Band (UWB) technology is currently flourishing as a viable alternative to achieve relative localization. UWB systems are cost-effective, can be used to directly access distance information between entities, and can work in Non-Line-of-Sight (NLoS) conditions. Most of the UWB-based solutions are infrastructure-based, *i.e.*, they use fixed UWB sensors with known positions (namely the *anchors*)

All the authors are with the Department of Engineering, University of Perugia, 06125 Perugia, Italy valerio.brunacci@studenti.unipg.it, alberto.dionigi, alessio.deangelis, gabriele.costante@unipg.it. This work was supported in part by the Italian Ministry of University and Research (MUR) under Grant PRIN 2022P3JY7N, funded by the European Union - Next Generation EU.

to localize the platform equipped with another UWB device (often referred to as *tag*) [4].

Recently, UWB sensors have been also employed in infrastructure-less configurations to achieve relative localization between robots [5], [6]. In this case, also the anchors are mounted on a mobile platform that estimates the relative position of the robot with the tag UWB. However, most of the State-of-the-Art (SotA) approaches keep the robot with anchors in a fixed position, limiting the advantages of an infrastructure-less setup. Additionally, in infrastructure-less setups the tag moves outside the convex hull delimited by the anchors. In practice, this translates in a worse Geometric Dilution Of Precision (GDOP) [7], a metric that models the localization errors depending on the anchor-tag arrangement. In some configurations, particularly those where the tag is outside the anchor hull, small measurement errors might result in large relative position errors.

Driven by the aforementioned considerations, the purpose of this paper is to build an infrastructure-less UWB-based relative 2D localization approach for multi-robot systems that overcomes the drawbacks of SotA solutions. In particular, we develop an *active* localization method that enables the robot with the anchors to adapt its position to reduce the relative localization error of the tag robot, see Fig 1. This is achieved by training a Deep Reinforcement Learning (DRL) agent that controls the platform to minimize a loss function based on the GDOP metric and on a measurement model tailored to the specific problem we consider. Differently from other solutions, we do not assume that the robot with the anchors is static.

The paper is organized as follows: in Section II we provide a literature review on the problem considered. Then in Sections III we provide the preliminary background to introduce the proposed methodology, which is described in Section IV. In Section V, the experimental setup is outlined and the results obtained are discussed. Finally, in Section VI we draw conclusions.

II. RELATED WORK

In the context of multi-robot systems, robot localization can be framed as either a *self-localization* problem or a *relative localization* one, or even as a combination of both. The former, which is also typical in single robot scenarios, aims to localize the robots with respect to a global fixed reference frame (*e.g.*, a map). In practice, addressing this problem in multi-robot systems entails treating each platform as an independent robot.

Conversely, relative localization implies that the robots can estimate the position of each other in their respective body frames. This information can be used for multiple purposes: i) refining the global localization estimates of each robot; ii) localize each robot in the system with respect to a leader unit (which might perform self localization); iii) solve tasks in which global localization is not of interest, such as target tracking, or cooperative operations or maneuvers.

In the following, we focus on indoor scenarios and first provide a brief overview on infrastructure-based methodolo-

gies. Then, we focus on relative localization in infrastructure-less setups for multi-robot systems, emphasizing the motivations behind UWB-based strategies. As we exploit the active localization paradigm, we also provide a literature overview of approaches based on this paradigm. Finally, we highlight the contribution of our work.

Infrastructure-based Localization. Infrastructure-based localization (of single or multi-robot systems) in indoor environments is achieved by exploiting a fixed external structure, whose elements are static and in known positions. Both the structure and the mobile platforms can be equipped with either passive markers or active sensors, depending on the technology employed. In MoCAP systems, high-frequency cameras are used to estimate the robot's pose by tracking markers mounted on it [1]. This configuration can be reverted by mounting a camera on the robot to detect visual markers installed on the walls at known positions [3]. Different solutions employ other sensors, such as radio frequency [8], [9] and ultra-violet [10].

To overcome the LoS requirement of the aforementioned sensors, solutions based on UWB devices that work also in NLoS conditions have started to flourish [11], also in combination with Inertial Measurement Units [12], cameras, and odometric sensors [13].

Typically, in infrastructure-based scenario, a set of UWB nodes, known as *anchors*, are deployed in the corners of the room at fixed and known positions. These devices are used to measure distances from another UWB node mounted on the mobile platform, named *tag*, within the convex hull defined by the anchors. Thus, the localization area is confined within such a hull, and moving the structure into another environment requires considerable effort and calibration procedures.

UWB-based Infrastructure-less Localization. UWB technologies can also be used in infrastructure-less scenarios. In this case, also the anchors are mounted on a mobile robot, and relative localization with respect to the robot housing the tag UWB sensors can be performed. While this solution is more flexible and does not require a fixed structure, it opens important challenges. First, the precision of the pose estimates is, in general, worse than that obtained with infrastructure-based setups. Secondly, the tag devices necessarily fall outside the convex hull of the anchors (which is limited by the robot size). Thus, additional techniques need to be used to refine the pose estimate.

One of the first works that explored the infrastructure-less setup is [5]. However, it requires a complex and long initialization phase, reducing its practical usage. A viable alternative is presented in [14], although the approach proposed by the authors needs some of the UWB nodes to be static. In [6], both the requirements of an initialization phase and of static nodes are lifted. Nonetheless, they rely on additional odometry sensors and the localization performance is lower than those achieved by infrastructure-based strategies. Recently Gao et al. [15] proposed a solution solely based on UWB sensors that is capable of providing performance comparable to those of infrastructure-based approaches. In this case, the anchors are deployed on the top of one robot

and are arranged at the corners of an equilateral triangle with 1 meter sides. This arrangement provides an almost uniform GDOP distribution around the robot. However, the platform with the anchors is kept static in the experiments, which, in practice, cancels the benefits of an infrastructure-less setup.

Active Localization. Considering only stationary anchors poses severe limitations on the applicability of such methods. On the other hand, in more challenging scenarios where each platform can be moved independently, an active approach could considerably enhance the localization performance in an infrastructure-less setting. Active perception systems [16] take advantage of the robot motion in order to *actively* acquire data from the environment and obtain more useful information with respect to the specific task of interest.

To deal with this complex decision-making problem, recent studies are using Deep Reinforcement Learning approaches to train control policies suitable for active tasks [17]–[19], achieving remarkable results in terms of generalization, performance, and robustness. Among these, the problem of active localization was also considered. In [19], the authors address the global localization problem on an *occupancy-grid* map with an active approach driven by DRL. The method employs RGB images for localization and the goal of the policy is to move the robot in order to acquire the views of the environment that better disambiguate its position in the map. The method shows remarkable performance on mazes, however, several assumptions limit its applicability in real-world scenarios. These drawbacks are addressed in [20] where the authors improve the previous approach and test their policy on a real robotic platform. Recently, the focus has shifted to active SLAM [21], which unifies the localization and the mapping problems. The goal is to actively control a robot performing SLAM to reduce both the uncertainty in its localization and the representation of the map.

A. Contributions

While important results have been demonstrated by active approaches for localization, to the best of our knowledge there are no previous works that explored this strategy in UWB-based infrastructure-less relative localization problems. Due to the importance of the GDOP on the UWB-based systems localization performance, we propose a new framework to first study and design an anchor arrangement to adapt the GDOP to the relative localization task in mobile robot scenarios. Then we exploit this design through an active localization strategy. Specifically:

- We propose an infrastructure-less active relative localization method that only leverages range measurements coming from UWB-devices. Our approach is capable of reducing the relative localization error up to 60% with respect to current State-of-the-Art approaches;
- Through an extensive study on the GDOP, we (i) analyze and propose the best anchor arrangement for the considered relative active localization task, resulting in an isosceles configuration, and (ii) we design a novel loss function that combines GDOP and the measurement model of the UWB sensors;

- We use this loss function to develop a Deep Reinforcement Learning-based agent for active relative localization that controls the robot with the anchors in order to maximize the position estimation accuracy.

III. PRELIMINARY

Notation: We use lower-case symbols x to denote scalars, bold lower-case symbols \mathbf{x} to indicate vectors, and bold upper-case symbols \mathbf{X} to denote matrices.

Preliminary Definitions: In our infrastructure-less scenario, the *anchors* are mounted on a mobile robot, and their positions with respect to the platform body frame \mathcal{B}^1 , denoted as $\mathbf{p}_{a_i} = [x_{a_i}, y_{a_i}]$, are known. On the other hand, a *tag* represents the node to be localized by using distance measurements between itself and the *anchors*. The localization is *relative*, *i.e.*, we are interested in estimating the tag position with respect to the body frame of the robot with the anchors. We use $\mathbf{p}_t = [x_t, y_t]$ to denote the true relative position of the *tag*, and $\hat{\mathbf{p}}_t = [\hat{x}_t, \hat{y}_t]$ to indicate its estimated relative position using noisy distance measurements.

Geometrical Dilution of Precision - A comprehensive analytical characterization of GDOP is provided in [7]. The GDOP models the effect of geometry on the relationship between measurement error and position estimation error. Also, it is affected by the number of anchors and by the angle between anchors and tag. GDOP is a unitless metric, which describes how measurement errors affect the position estimation considering the geometry of the anchors. It is defined as the ratio between the root-mean-squared error (RMSE) of the position and that of the ranging error [22]:

$$GDOP(\hat{\mathbf{p}}_t, \mathbf{p}_t) = \frac{\sqrt{E\{(\hat{\beta} - \hat{\mu})^T(\hat{\beta} - \hat{\mu})\}}}{\sigma_{range}} \quad (1)$$

where E is the expectation operator, $\hat{\beta}$ is the position estimator used and $\hat{\mu}$ is its mean.

In the specific case of 2D relative localization with three anchors considered in this work, an alternative definition of GDOP can be formulated as follows [7]:

$$GDOP(\mathbf{p}_t) = \sqrt{\mathbf{G}_{11} + \mathbf{G}_{22}} \quad (2)$$

where

$$G = (\mathbf{H}^T \mathbf{H})^{-1}, \quad \mathbf{H} = \begin{bmatrix} \frac{x_t - x_{a_1}}{\|p_{a_1} - p_t\|} & \frac{y_t - y_{a_1}}{\|p_{a_1} - p_t\|} \\ \frac{x_t - x_{a_2}}{\|p_{a_2} - p_t\|} & \frac{y_t - y_{a_2}}{\|p_{a_2} - p_t\|} \\ \frac{x_t - x_{a_3}}{\|p_{a_3} - p_t\|} & \frac{y_t - y_{a_3}}{\|p_{a_3} - p_t\|} \end{bmatrix}$$

IV. METHODOLOGY

A. Problem Formulation

The objective of the proposed active relative localization method is to provide a robotic platform with a suitable control policy so that to obtain the best possible estimation of another robot position. In this paper, we consider an infrastructure-less context with two distinct robotic platforms: the first one, named *AnchorBot* for brevity, is

¹ We omit the letter \mathcal{B} in the variable definitions to simplify the notation since \mathcal{B} is the only considered reference frame in this work.

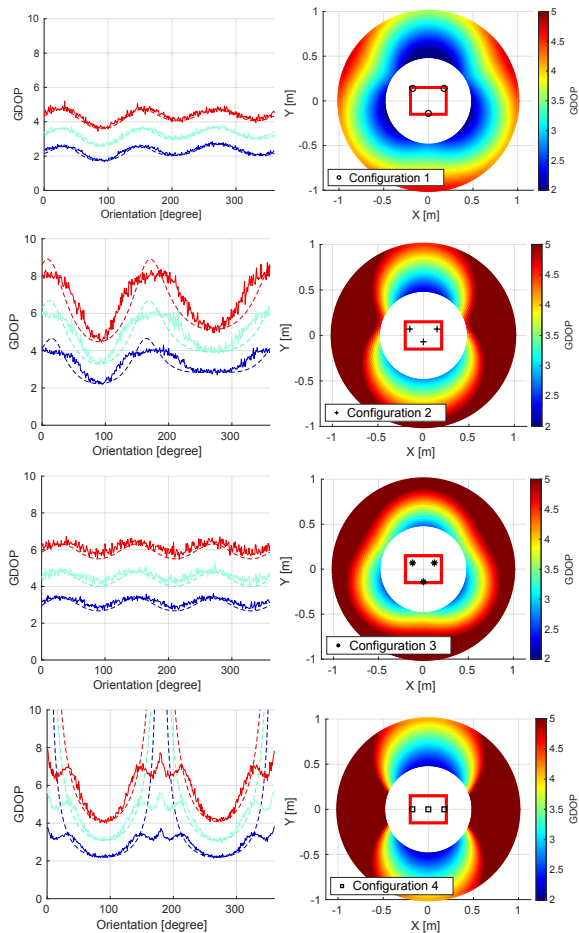


Fig. 2: Configurations from 1 to 4. On the left: simulated GDOP (continuous lines) and analytical GDOP (dashed lines) calculated on 3 circumferences of radius 50 cm (blue), 74 cm (light blue), and 1 m (red), with respect to the AnchorBot. On the right: analytical GDOP on a ring of points around the AnchorBot.

equipped with three UWB anchor nodes, and its function is to localize with respect to its body frame the second one, referred to as the *TagBot*, which is equipped with a single UWB tag. The *AnchorBot* have access only to the relative distances between the anchors and the tag, and the goal is to adapt its position in order to minimize the *TagBot* localization error.

To this aim, we firstly conduct an in-depth anchor placement analysis IV-B to identify a configuration specifically tailored for the active localization task. Subsequently, we define a novel *UWB Relative Localization Loss* IV-C that takes into account the short range measurement model of the sensors and the GDOP metric formulation. Lastly, we employ this loss to train a Deep Reinforcement Learning agent IV-D that implements the control policy for the *AnchorBot*.

B. Anchors Displacement Analysis

As shown in Fig. 3(a), the minimum of the GDOP function falls within the convex hull delimited by the anchors. In an infrastructure-less scenario, however, the sensors are mounted on the *AnchorBot* and the anchor hull area, in practice, contains the platforms itself. To prevent collisions, the *TagBot* must be positioned outside that area. Thus,

to address this issue, new anchor configurations that can achieve a reasonably low GDOP outside the anchor hull should be explored. This poses the following dilemma: should we prioritize achieving a lower minimum GDOP within a specific zone, or opt for a higher but almost uniformly distributed GDOP around the *AnchorBot*? [15] address the latter scenario: they maintain the *AnchorBot* in a stationary position, while the *TagBot* moves freely. This strategy requires keeping GDOP as low as possible at all points surrounding the *AnchorBot*. Among various anchor configurations, the equilateral triangle arrangement (as depicted in Configuration 3 in Fig. 2) comes closest to achieving this goal.

However, a uniformly distributed GDOP results in a higher minimum value (*i.e.*, 2.5) compared to configurations tailored for localizing the robot within specific areas (such as Configurations 1 and 2). Consequently, we select Configuration 1 (see Fig. 2) due to its GDOP minimum below 2. The trade-off is that the *AnchorBot*'s front region hosts this minimum, requiring precise localization of the *TagBot* in that zone. To address this, we adopt an active localization approach, utilizing an RL-based controller to maintain the *TagBot* on the designated side of the *AnchorBot*.

Monte Carlo Simulations: The theoretical GDOP values described in the previous section are also validated by simulating real-world conditions using (1). Specifically, we corrupt distance measurements with zero-mean Gaussian noise characterized by a variance of 5 cm, which is representative of a realistic level of sensor noise [23]. A Monte Carlo simulation (MCS) is performed to estimate the GDOP for each configuration considered in a circular region surrounding the *AnchorBot* (see Fig. 2). For each sampled *TagBot* location in the circular region, we compute 1000 position estimates by using a trilateration technique based on the Non-Linear Least Square (NLLS) algorithm [24]. Notably, our estimator is unbiased, ensuring that the mean of the estimator $\hat{\mu}$ aligns with the true parameter [22].

C. UWB Relative Localization Loss

As anticipated in the previous Section, we choose an anchor configuration that attains a minimum GDOP in front of the *AnchorBot*. We assign the task of controlling the platform to maintain the *TagBot* within this minimum area (thus minimizing the relative localization error) to an active control approach. To this aim, we chose to adopt a learning-based approach for the controller design and, in order to set up the optimization process, we design a loss function encoding the requirement of the task considered.

In a UWB-based localization context, directly using the GDOP as the loss function might seem a reasonable choice since, by definition, it is a performance metric for positioning. However, this is impractical since, in an infrastructure-less scenario, the GDOP minimum falls in the *AnchorBot* footprint area, as described in IV-B. Furthermore, another aspect that influences localization performance is distance. More specifically, as observed in [25], [26], UWB sensors are not able to provide a reliable measurement for distances

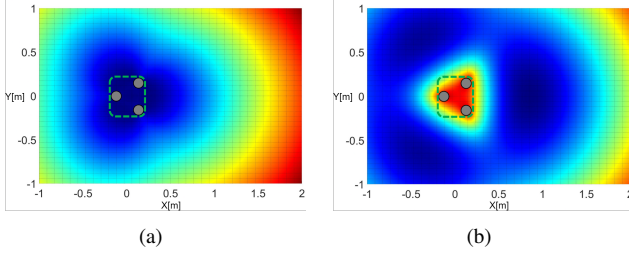


Fig. 3: Comparison between (a) the GDOP and (b) the UWB Relative Localization Loss. The grey dots are the anchors, while the dashed green line represents the footprint of the *AnchorBot*. Cold colors indicate low values, and warm ones refer to high values.

less than 10 cm, and for short-range measurements, typically between 10 and 25 cm, they exhibit greater errors in terms of measurement accuracy and variance compared to what they are able to provide for distances from 25 cm up to 5 m.

Motivated by the above considerations, we develop a novel *UWB Relative Localization Loss* $\ell(\mathbf{p}_t) : \mathbb{R}^2 \rightarrow \mathbb{R}$ that combines the GDOP metric with a suitable measurement model for the anchors and takes into consideration the UWB performance degradation for short ranges. The loss function depends on the tag position \mathbf{p}_t and is defined as:

$$\ell(\mathbf{p}_t) = GDOP(\mathbf{p}_t) + \alpha \sum_{i=0}^N f_e(\|\mathbf{p}_t - \mathbf{p}_{a_i}\|) \quad (3)$$

where \mathbf{p}_{a_i} is fixed and constant in the formula, N is the number of *anchors*, and α is a tunable hyper-parameter. The $GDOP(\mathbf{p}_t)$ term corresponds to the formulation in (2), and f_e is the analytical function of the measurement model derived from the study in [25]:

$$f_e(\|\mathbf{p}_t - \mathbf{p}_{a_i}\|) = c_1 e^{k_1 \|\mathbf{p}_t - \mathbf{p}_{a_i}\|} \quad (4)$$

where c_1 and k_1 are two parameters with value 0.51 and -3.152 1/m, respectively.

In Fig. 3(b) we show that the developed loss function $\ell(\mathbf{p}_t)$ has a suitable global minimum in front of the robot, which is unique and takes into account the performance drop in the proximity of the anchors.

D. Deep Reinforcement Learning controller

In order to develop a control policy to effectively address the Active Relative Localization task, we employ a learning-based strategy and define a Reinforcement Learning problem.

Simulation Environment - Learning a control policy directly on a real platform is not, in general, feasible due to safety and temporal constraints. Therefore, it is necessary to employ a simulation environment to train the algorithm. In this work, we model the *AnchorBot* as a unicycle-type differential drive platform [27]. The robot can move along its heading direction with linear velocity v and spin around the center point position of the vehicle with angular velocity ω . The commands v and ω are provided by the controller and correspond to the input of the robot. Moreover, the sampling instants are denoted by $k = jt_s$, in which j is the discrete-time step number and t_s is the sampling time.

On the other hand, during training, the *TagBot* moves along a sinusoidal trajectory parameterized with respect to *TagBot* starting position as follows:

$$\mathbf{p}_t(k) = \begin{bmatrix} A_x \sin(2\pi f_x k + \phi_x) \\ A_y \sin(2\pi f_y k + \phi_y) \end{bmatrix} \quad (5)$$

where (A_x, A_y) , (f_x, f_y) and (ϕ_x, ϕ_y) are respectively the amplitude, the frequency and the phase along the (x, y) axes.

DRL Strategy - In our settings, the two robots are free to move on a 2-D plane. The *AnchorBot* controlled by the DRL agent is equipped with three UWB *anchors* that respectively return their distances $\mathbf{d}(k) = [d_1(k), d_2(k), d_3(k)]$ from the *tag* mounted on the *TagBot*. The observation $\mathbf{o}(k)$ is composed of H consecutive readings of the sensors

$$\mathbf{o}(k) = [\mathbf{d}(k), \mathbf{d}(k-1), \dots, \mathbf{d}(k-H+1)] \quad (6)$$

resulting in a vector of length $3 \times H$. On the other hand, since the robot can be moved by specifying the desired linear velocity v and angular velocity ω , the action space is a continuous control command in the form of $\mathbf{u}(k) = [v(k), \omega(k)]$.

Given the large dimensionality of the state space and the continuous nature of the considered problem, employing a classical tabular RL approach is not feasible. Instead, we employ a DRL strategy, leveraging Deep Neural Network (DNN) approximators. In particular, we adopt an *asymmetric actor-critic* framework [18] and we design two DNN architectures: the *actor* (A-DNN) one learns the optimal policy $\pi(o(k))$, while the *critic* (C-DNN) counterpart is used during training phase only to evaluate such a policy.

The A-DNN consists of a Multi-Layer Perceptron (MLP) with three hidden layers, each one characterized by 256 neurons and ReLU activations. The input to the network is the flattened vector $\mathbf{o}(k)$ in (6), and its output $\mathbf{u}(k)$ contains the linear and the angular velocities needed to control the robot. Furthermore, to impose the physical constraints that come from the saturation limits of the motors, we add a tanh function to the last layer of the network in order to contain the output of the network within a reasonable range.

Through the use of an asymmetric framework, we have the flexibility to provide the C-DNN with more privileged information compared to what is accessible to the A-DNN during inference. Consequently, in order to develop a more robust criterion for policy evaluation, we define the critic observation $\mathbf{o}_c(k) = [\mathbf{p}_t(k), \theta(k)]$, where $\theta(k)$ is the *AnchorBot* orientation. The architecture of the C-DNN is an MLP composed of 3 hidden layers with 256 neurons each and ReLU activations. The network takes as input the concatenation of $\mathbf{o}_c(k)$ with $\mathbf{u}(k)$, and outputs estimation of the *action-value* function $Q_\pi(\mathbf{o}_c(k), \mathbf{u}(k))$.

Optimization Procedure - Given the continuous formulation of the problem, we decided to train the A-DNN and the C-DNN by employing the Soft Actor-Critic (SAC) [28] RL-framework, which is one of the most recent off-policy strategies. The goal of the reward signal $r(k)$ is to minimize the localization error while avoiding collisions and reducing

the control effort. To this purpose, we firstly normalize the UWB-Localization Loss so that its minimum value is zero:

$$\ell_{scaled}(\mathbf{p}_t(k)) = \frac{\ell(\mathbf{p}_t(k)) - \ell_{min}}{\ell_{max} - \ell_{min}}, \quad (7)$$

where ℓ_{min} and ℓ_{max} are the minimum and maximum values of the loss function, respectively. Subsequently, we design a first reward term that optimize (7):

$$r_l(k) = \max(0, 1 - \ell_{scaled}(\mathbf{p}_t(k))), \quad (8)$$

It can be noted that $r_l(k)$ is 1 only when the *AnchorBot* manages to keep the *TagBot* in the point that minimizes (3).

In order to also consider the control effort on the actuators, a reward penalty contribution r_u is included in the design:

$$r_u(k) = \frac{\|u(k)\|}{1 + \|u(k)\|}. \quad (9)$$

The reward function is obtained by adding up the above terms. Furthermore, to ease the learning process, we removed the local minima of (8) by considering its contribution only when the *TagBot* is in front of the *AnchorBot* within a suitable angle, which results in:

$$r(k) = \begin{cases} r_l(k) - k_u r_u(k) & x_t(k) > 0 \ \& \ |\gamma| < \frac{\pi}{4} \\ -k_c & \|p_t(k)\| < d_m \\ -k_u r_u(k) & otherwise \end{cases} \quad (10)$$

with

$$\gamma = \arctan \frac{y_t(k)}{x_t(k)} \quad (11)$$

where $k_u > 0$ is a weighting parameter that balance the performance and the control effort terms, and k_c is a large positive constant that penalize the RL agent when it does not meet the collision avoidance constraint in which d_m is the minimum distance allowed between the two robots.

V. EXPERIMENTS

A. Implementation details

Based on the study carried out in IV-B, we opt for a configuration of the anchors on the *AnchorBot* over the vertices of an isosceles triangle, whose positions with respect to the robot body frame are $(0.14m, 0.175m)$, $(0.14m, -0.175m)$ and $(-0.14m, 0m)$, respectively. Hence, in the following discussion our approach is referred to as **AUL-IS** (Active UWB-Localization - ISoscele triangle configuration). On the other hand, the tag to be localized is considered to be in the center point of the *TagBot*.

The hyper-parameter α of the UWB Relative Localization Loss (3) is a design parameter that balances the GDOP metric and the short-range measurement model terms. In order to obtain a position for the minimum that guarantees high localization performance, we set its value to 10.

For the optimization of the active controller, we use the Stable-Baselines3 [29] implementation of SAC, which we properly customized to implement our *asymmetric actor-critic* structure. The A-DNN and C-DNN have been trained by employing the Adam optimizer with a learning rate of 0.0003, a discount factor γ of 0.99, and a batch size of 256. The training phase is structured in episodes with a maximum

duration of 30 s. At the beginning of each one, the *TagBot* is randomly spawned with a uniform distribution at a distance between 60 cm and 100 cm from the *AnchorBot*. In addition, to achieve robustness against the UWB measurement noise, we corrupt the distance data with an additive zero-mean Gaussian noise with a variance 5 cm, according to [24]. During the episode, the *TagBot* remains stationary for the first 10 seconds and then starts moving along the trajectory reported in (5). Furthermore, to produce a different *TagBot* movement for each episode, we uniformly randomize the amplitude, the frequency and the phase of the trajectory in the intervals $[1, 2.5]$ m, $[0.008, 0.016]$ Hz, and $[0, 2\pi]$ rad, respectively. The episode terminates when the maximum number of steps is reached or if the *AnchorBot* violates the collision avoidance constraint.

The training process requires about 10 minutes and 1.1GB of VRAM on a workstation equipped with NVIDIA Quadro GV100. At inference time, the VRAM required for the Actor Network is about 320 kB, and the time required to compute the action is approximately 1 ms.

B. Experimental Setup

The proposed **AUL-IS** approach is tested through an extensive simulation campaign and validated with real-world experiments on a wheeled robotic platform. More specifically, we perform two distinct types of experiments. In the first one, the *TagBot* remains stationary and assumes three different positions: in front of, to the side, and behind the *AnchorBot*. The second set of tests is, instead, aimed at assessing the capability of the proposed active method to localize a moving platform. Specifically, we evaluate the generalization capabilities of the RL controller by considering trajectories with shapes different from those used for training, *i.e.*, a two-meters straight line, a circular trajectory centered in the *AchorBot*, and a one-meter side rectangular shape. To ensure a comprehensive experimental campaign, we vary the spawn distances for the *TagBot* relative to the *AnchorBot*, ranging from 70 cm up to 200 cm. For each experiment, we run a total of 1000 realizations.

As a comparison baseline, we employ the method proposed in [15]. Following the original work, the *AnchorBot* is kept stationary and the anchors are mounted on the vertices of an equilateral triangle. We refer to this method as **SUL-EQ** (Static UWB Localization - Equilateral configuration). Moreover, to highlight the advantages of our design choices, we also included as baselines (i) a stationary *AnchorBot* with our isosceles triangle configuration for the anchors (**SUL-IS**) and (ii) an active version of [15] in which the anchors have an equilateral triangle configuration, but the *AnchorBot* is controlled by our DRL active policy (**AUL-EQ**).

To evaluate the performance of the proposed approach and the baselines, we compute the Mean Localization Error (MLE) $\frac{1}{N} \sum_{t=1}^N \|\hat{\mathbf{p}}_t - \mathbf{p}_t\|$ over multiple runs of the experiments, and we adopt as comparison metrics the mean μ and standard deviation σ . Moreover, since the only available information are the three distance measurements between the *anchors* and the *tag*, we employ the NLLS method to obtain

TABLE I: Results of the simulation experiments. The mean μ and the standard deviation σ of the MLE in cm are reported.

AnchorBot - TagBot Init. Distance	Method	Static TagBot						Dynamic TagBot					
		Front Spawn		Side Spawn		Behind Spawn		Straight Line		Circle		Square	
		μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
70 cm	SUL-EQ [15]	11.0	7.8	12.9	9.6	16.0	13.9	28.2	24.9	13.3	10.4	21.6	18.2
	SUL-IS	10.3	7.1	12.9	9.7	21.6	30.7	26.0	22.8	13.5	14.1	20.2	16.9
	AUL-EQ	11.5	8.1	11.7	8.3	11.5	9.3	12.7	9.1	11.4	8.0	11.9	8.5
	AUL-IS (Our)	10.7	7.5	11.0	7.8	11.3	9.7	11.7	8.3	10.6	7.4	11.1	7.8
85 cm	SUL-EQ [15]	13.4	9.7	15.4	11.6	18.1	14.1	30.8	26.6	15.7	11.8	23.8	19.9
	SUL-IS	12.5	8.9	15.4	11.5	19.4	22.9	28.4	24.4	15.5	13.4	22.2	18.4
	AUL-EQ	11.6	8.3	12.1	8.8	12.2	9.1	12.7	9.1	11.9	8.5	12.0	8.7
	AUL-IS (Our)	10.9	7.7	11.4	8.3	11.7	9.3	11.8	8.4	11.1	7.8	11.8	8.4
100 cm	SUL-EQ [15]	15.9	11.7	18.2	13.7	20.7	15.7	33.3	28.3	18.3	13.7	26.2	21.6
	SUL-IS	14.9	10.7	18.0	13.6	20.2	19.5	31.0	26.2	17.7	14.1	24.4	20.0
	AUL-EQ	12.0	8.7	12.6	9.5	12.9	9.8	13.1	9.5	12.2	8.7	12.3	9.0
	AUL-IS (Our)	11.2	8.0	11.9	9.0	12.3	9.8	12.1	8.7	11.3	7.9	12.1	8.7
150 cm	SUL-EQ [15]	24.6	18.6	26.8	20.3	29.3	22.3	42.1	34.8	26.9	20.4	34.3	27.6
	SUL-IS	22.8	17.1	26.9	20.4	27.5	22.0	39.0	31.8	25.9	19.8	31.9	25.5
	AUL-EQ	15.0	12.2	16.2	13.7	17.0	14.4	17.2	13.7	12.8	9.4	14.5	12.1
	AUL-IS (Our)	13.9	11.2	15.4	13.1	16.0	13.7	16.0	12.6	11.8	8.5	16.0	12.6
200 cm	SUL-EQ [15]	33.2	25.2	35.6	27.1	37.9	29.0	50.7	41.0	35.6	27.1	42.8	33.9
	SUL-IS	30.8	23.1	35.6	27.1	35.3	27.4	47.0	37.7	34.3	26.2	39.7	31.3
	AUL-EQ	21.0	17.9	22.7	19.5	24.5	20.6	25.4	20.1	13.3	10.3	17.5	16.6
	AUL-IS (Our)	19.5	16.6	23.9	20.4	22.5	19.3	23.8	18.6	12.2	9.4	15.9	15.1

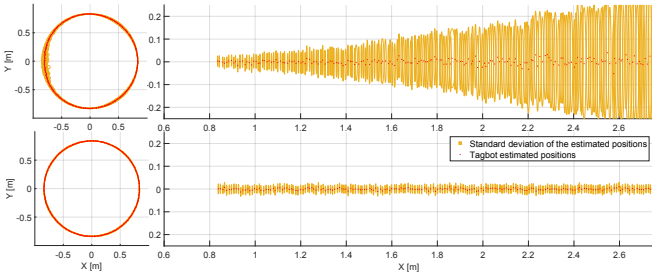


Fig. 4: Plots of the dynamic experiments for both **SUL-EQ** (top) and **AUL-IS** (bottom) are shown for the *Straight Line* and *Circle* experiments.

the position estimation $\hat{\mathbf{p}}_t$ of the *TagBot*, as for the MCS in IV-B. It is important to highlight that the proposed method is independent from the localization algorithm, since the DRL controller directly employs the raw distances coming from the UWB devices to move the robot.

C. Experimental Results

The results of the simulated experimental campaign are reported in Table I. A first key finding is that the active approaches **AUL-EQ** and **AUL-IS** outperform their static counterparts **SUL-EQ** and **SUL-IS** in every situation, achieving an error reduction up to 60% (Straight Line-150 cm). This result is a direct consequence of the control policy developed by the DRL agent, which is able to effectively move and orient the *AnchorBot* so as to maintain the *TagBot* in low GDOP areas. Moreover, it can be observed that the specialized anchor configuration exploited by **AUL-IS** can further enhance the localization performance. Clearly, as demonstrated by the static *TagBot* experiments and in accordance with the findings in [15], an equilateral triangle disposition is the best choice with a stationary *AnchorBot*. However, when we leverage an active approach that can take advantage of the *AnchorBot* movements, it is convenient to use an isosceles triangle configuration that places the minimum of the GDOP to be in front of the *AnchorBot*.

A second important aspect to emphasize is that **AUL-IS** is

able to make localization performance independent from the *TagBot* position and movements, and from the *AnchorBot* initialization. This capability is not observed in **SUL-EQ** or **SUL-IS** that do not include strategies of adaptation to GDOP conditions. For instance, if we refer to the second row of Table I (*i.e.*, *AnchorBot - TagBot* Init. distance 85 cm) and we consider both static and dynamic experiments, the difference between the worst ($\mu = 30.8$) test and the best ($\mu = 13.4$) one achieved by **SUL-EQ** is about 17.4 cm, while for **AUL-IS** this gap is less than 1 cm. A further demonstration is shown in Fig. 4. It can be observed that for **SUL-EQ**, the localization error variance increases with respect to the orientation and the distance between the two robots. On the other hand, **AUL-IS** achieves remarkable performance in all the considered scenarios and the variance of its localization error is always low and constant.

D. Real World Validation

Real-world experiments are carried out to validate the results obtained in the simulation campaign and to demonstrate the transferability of the DRL controller on a real platform. To this aim, we employ two GoPiGo3² robots and four Quorvo DWM1001-Dev UWB devices³ configured as described in V-A. The three *anchors* and the *tag* are placed on a wooden board screwed to the frame of the *AnchorBot* and the *TagBot*, respectively. Both robots are equipped with an embedded platform for the computation of the control algorithm and distance measurement acquisition.

The experiments are performed indoors, and the OptiTrack⁴ motion capture system is employed to obtain ground truth information. We test our method with either stationary and dynamic *TagBot* conditions, and in every configuration described in V-B. Moreover, without loss of generality, we considered 85 cm as the *TagBot* start distance, which was the largest possible to perform experiments within the area covered by the MoCAP system. We chose **SUL-IS** as the

² <https://www.dexterindustries.com/gopigo3/>

³ <https://www.qorvo.com/products/d/da007946>

⁴ <https://optitrack.com/>

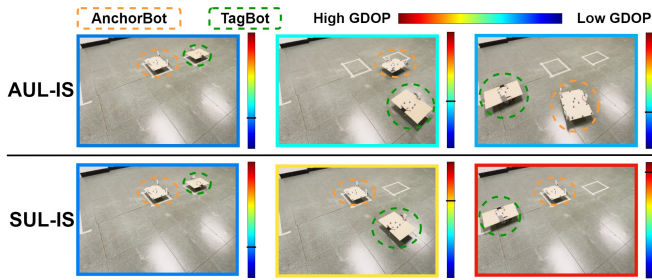


Fig. 5: Circle dynamic experiment with a radius of 85 cm in real-world. The colored outlines in the figures indicate the GDOP value for the considered timestamp according to the colorbar scale.

TABLE II: Results of the real-world experiments. The mean μ the standard deviation σ of the MLE in cm are reported.

Method	Static TagBot						Dynamic TagBot					
	Static 0°		Static 90°		Static 180°		Straight Line		Circle		Square	
	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
SUL-IS	5.6	3.7	9.0	6.3	7.6	9.7	32.5	22.2	13.2	8.8	26.6	22.9
AUL-IS (Our)	4.9	3.3	7.1	4.9	5.9	4.2	12.7	8.1	12.2	7.2	14.7	7.8

comparison baseline since in simulation it shows the best overall performance with respect to **SUL-EQ** in the 85 cm scenarios. The experiments are averaged over three different runs of about 30 seconds each, and before performing the actual trial, we carried out a calibration procedure similar to the one proposed in [23] to eliminate the bias present in the distance measurements. More specifically, we calibrate **SUL-IS** according to the specific configuration of each experiment, while for **AUL-IS** the calibration is performed only once with the *TagBot* placed in front of the *AnchorBot*. It should be noted that **SUL-IS** is favored in the comparison since it is calibrated *ad-hoc* for each configuration.

The trained DRL policy is directly deployed on the real *AnchorBot* without any fine-tuning, and the results against the static baseline are reported in Table II. The learned controller demonstrates strong generalization capabilities as it achieves remarkable localization performance when real robotic platforms and UWB sensors are involved. The numerical experiments show that **AUL-IS** outperforms **SUL-IS** in each considered scenario, confirming what we observe in the simulations experiments (see Fig. 5 for qualitative results).

VI. CONCLUSION AND FUTURE WORKS

In this work, we presented an infrastructure-less active relative localization method for multi-robot systems using UWB distance sensors. Specifically, we proposed a new anchor configuration through a GDOP analysis and introduced a novel loss function used to train a DRL-based controller to adapt the position of the *AnchorBot* and minimize the relative localization error of the *TagBot*. Extensive simulations and real-world experiments validated the effectiveness of our approach and highlighted its potential for practical deployment. In particular, the proposed strategy exhibits up to a 60% error reduction in accuracy compared to existing approaches. In future work, we will enhance our method by incorporating sensor fusion techniques to integrate our UWB-based localization approach with visual information. In addition, more challenging robotic platforms, such as Micro Aerial Vehicles (MAVs) will be considered.

REFERENCES

- [1] P. Merriault *et al.*, "A study of vicon system positioning performance," *Sensors*, vol. 17, no. 7, p. 1591, 2017.
- [2] G. Bresson *et al.*, "Simultaneous localization and mapping: A survey of current trends in autonomous driving," *IEEE Trans. Intell. Veh.*, vol. 2, pp. 194–220, 9 2017.
- [3] M. Kalaitzakis *et al.*, "Fiducial markers for pose estimation: Overview, applications and experimental comparison of the artag, apriltag, aruco and stag markers," *J. Intell. Robot.*, vol. 101, 2021.
- [4] F. Zafari *et al.*, "A survey of indoor localization systems and technologies," *IEEE Commun. Surveys Tuts.*, vol. 21, pp. 2568–2599, 2019.
- [5] X. S. Zhou *et al.*, "Robot-to-robot relative pose estimation from range measurements," *IEEE Trans. Robot.*, vol. 24, pp. 1379–1393, 2008.
- [6] Z. Cao *et al.*, "Relative localization of mobile robots with multiple ultra-wideband ranging measurements," *IEEE IROS*, pp. 5857–5863, 2021.
- [7] N. Levanon, "Lowest gdop in 2-d scenarios," *IEE Proceedings: Radar, Sonar and Navigation*, vol. 147, pp. 149–155, 2000.
- [8] M. Longhi *et al.*, "Ubiquitous flying sensor antennas: Radiofrequency identification meets micro drones," *IEEE RFID*, vol. 1, no. 4, pp. 291–299, 2017.
- [9] —, "Rfid-based localization for greenhouses monitoring using mavs," in *2018 IEEE-APS (APWC)*. IEEE, 2018, pp. 905–908.
- [10] V. Walter *et al.*, "Uvdar system for visual relative localization with application to leader-follower formations of multirotor uavs," *IEEE Robot. Autom. Lett.*, vol. 4, pp. 2637–2644, 7 2019.
- [11] W. Zhao *et al.*, "Learning-based bias correction for time difference of arrival ultra-wideband localization of resource-constrained mobile robots," *IEEE Robot. Autom. Lett.*, vol. 6, pp. 3639–3646, 4 2021.
- [12] Y. Cao *et al.*, "Accurate position tracking with a single uwb anchor," *Proceedings - IEEE ICRA*, pp. 2344–2350, 5 2020.
- [13] H. Zhang *et al.*, "A dynamic window-based uwb-odometer fusion approach for indoor positioning," *IEEE Sensors J.*, vol. 23, pp. 2922–2931, 2 2023.
- [14] Y. Cao *et al.*, "Dynamic range-only localization for multi-robot systems," *IEEE Access*, vol. 6, pp. 46 527–46 537, 8 2018.
- [15] Q. Gao *et al.*, "Infrastructure-free relative localization: System modeling, algorithm design, performance analysis, and field tests," *Authorea Preprints*, 1 2024.
- [16] R. Bajcsy, "Active perception," *Proc. IEEE*, vol. 76, no. 8, pp. 966–1005, 1988.
- [17] A. Devo *et al.*, "Enhancing continuous control of mobile robots for end-to-end visual active tracking," *Rob. Auton. Syst.*, vol. 142, p. 103799, 2021.
- [18] A. Dionigi *et al.*, "E-vat: An asymmetric end-to-end approach to visual active exploration and tracking," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4259–4266, 2022.
- [19] D. S. Chaplot *et al.*, "Active neural localization," in *International Conference on Learning Representations*, 2018.
- [20] S. K. Gottipati *et al.*, "Deep active localization," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 4394–4401, 2019.
- [21] J. A. Placed *et al.*, "A survey on active simultaneous localization and mapping: State of the art and new frontiers," *IEEE Trans. Robot.*, 2023.
- [22] Z. Sahinoglu *et al.*, *Ultra-wideband Positioning Systems*. Cambridge University Press, 9 2008.
- [23] V. Brunacci *et al.*, "Development and analysis of a uwb relative localization system," *IEEE Trans. Instrum. Meas.*, vol. 72, 2023.
- [24] —, "Fusion of uwb and magnetic ranging systems for robot positioning," *IEEE Trans. Instrum. Meas.*, vol. 73, pp. 1–12, 12 2023.
- [25] S. D. Iacono *et al.*, "Distance measurement characterization for ultra wide band indoor localization systems," *IEEE I2MTC*, 2022.
- [26] T. Polonelli *et al.*, "Performance comparison between decawave dw1000 and dw3000 in low-power double side ranging applications," *2022 IEEE SAS - Proceedings*, 2022.
- [27] M. Leomanni *et al.*, "A convex programming approach to multipoint optimal motion planning for unicycle robots," *IEEE L-CSS*, vol. 7, pp. 1688–1693, 2023.
- [28] T. Haarnoja *et al.*, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *ICML*, 2018, pp. 1861–1870.
- [29] A. Raffin *et al.*, "Stable-baselines3: Reliable reinforcement learning implementations," *Journal of Machine Learning Research*, vol. 22, no. 268, pp. 1–8, 2021.