V. PIERLOT, and M. VAN DROOGENBROECK. **BeAMS: a Beacon based Angle Measurement Sensor for mobile robot positioning**. In *IEEE Transactions on Robotics*, 30(3):533-549, June 2014. http://dx.doi.org/10.1109/TRO.2013.2293834


# BeAMS: a Beacon based Angle Measurement Sensor for mobile robot positioning

Vincent Pierlot and Marc Van Droogenbroeck *Members, IEEE*
INTELSIG, Laboratory for Signal and Image Exploitation
Montefiore Institute, University of Liège, Belgium

***Abstract*—Positioning is a fundamental issue in mobile robot applications, and it can be achieved in multiple ways. Among these methods, triangulation based on angle measurements is widely used, robust, accurate, and flexible. This paper presents BeAMS, a new active beacon-based angle measurement system used for mobile robot positioning.**

**BeAMS introduces several major innovations. One innovation is the use of a unique unsynchronized channel with On-Off Keying modulated infrared signals to measure angles and to identify the beacons. We also introduce a new mechanism to measure angles: our system detects a beacon when it enters and leaves an angular window. We show that the estimator resulting from the center of this angular window provides an unbiased estimate of the beacon angle. A theoretical framework for a thorough performance analysis of BeAMS is provided. We establish the upper bound of the variance and validate this bound through experiments and simulations; the overall error measure of BeAMS is lower than $0.24\,deg$ for an acquisition rate of $10\,Hz$.**

**In conclusion, BeAMS is a low power, flexible, and robust solution for angle measurement, and a reliable component for robot positioning.**

***Index Terms*—Angle measurement, beacons, infrared detector, mobile robot, robot sensing system, positioning, triangulation.**

## I. INTRODUCTION

### A. Robot positioning

A mobile robot that evolves in its environment cannot navigate or execute its actions correctly without some form of positioning; therefore positioning is a crucial issue in mobile robot applications. Some fundamental papers such as [5], [6], [8], [38] discuss robot positioning. In particular, Betke and Gurvits [5] and Esteves *et al.* [15] highlight that sensory feedback is essential in order to position the robot in its environment. Some surveys (see [7], [12], [18], [28]) discuss several techniques used for positioning: odometry, inertial navigation, magnetic compasses, active beacons, natural landmark navigation, map-based positioning, and vision-based positioning. We can identify two main families encompassing these methods: (1) relative positioning (or dead-reckoning), and (2) global positioning (or reference-based). Techniques belonging to the first family mainly operate by odometry, which consists of counting the number of wheel revolutions (*e.g.* with optical encoders) and integrating them to compute the offset from a known position; inertial navigation (based on gyroscopes or accelerometers) is less used because of its poor accuracy [7]. Relative positioning based on odometry is very accurate for small offsets, but can lead to an increasing drift resulting from the unbounded accumulation of errors over time (due to the integration step, uncertainty about the wheelbase, wheel slippage, etc). A global positioning system is thus generally required to recalibrate the position of the robot periodically. On the other hand, global positioning systems are known to be less accurate than odometry, and this is why both methods are essential and complementary to each other [2], [5]. These two informations are generally combined in a Kalman filter or other data fusion algorithm [16], [22].

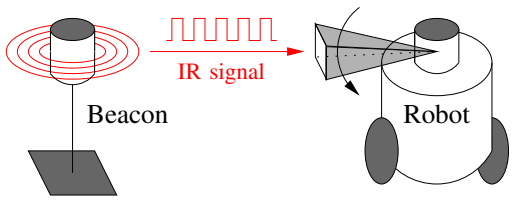



**Figure 1:** Presentation of BeAMS, a new angle measurement system for mobile robots based on two principles: (1) beacons send On-Off Keying coded infrared signals, and (2) the receiver on the robot turns at constant speed to measure angles of beacons. These angles can be combined to compute the robot position

### B. Positioning based on beacons

Most global positioning techniques rely on beacons, whose locations are known, and perform positioning by triangulation or trilateration. In the context of positioning, a beacon is a discernible object in the environment, which may be natural or artificial, passive or active; our system, BeAMS, uses active beacons, and its principle is illustrated in Figure 1. Triangulation is the process of determining the location of a point by measuring angles from that point to known locations (beacons) (see Figure 2). This contrasts with the trilateration technique which measures distances from the point to known locations. Because of its robustness, accuracy, and flexibility, triangulation with active beacons is widely used for robots [38]. Another advantage of triangulation versus trilateration is that the robot can compute its orientation (heading) in addition to its location [15], [17], [34], which can be as important as the robot position for most applications.

The description of an algorithm that uses angle measurements to compute a position or to navigate can be found in many papers, but is not our focus. Triangulation methods using three angle measurements can be found in [10], [14], [15], [17], [27], [34], and methods using more than three angle measurements are described in [5], [38]. More than three angles can be used to increase precision in some pathological

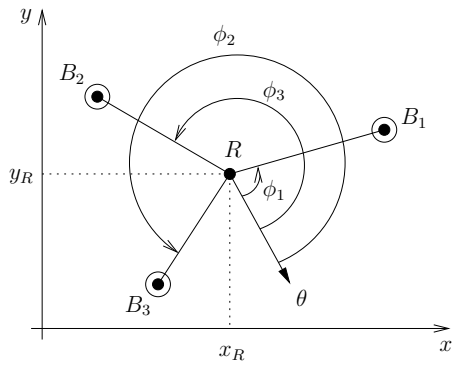



**Figure 2:** Triangulation setup in the 2D plane. $R$ denotes the robot. $B_i$ are the beacons. $\phi_i$ are the angle measurements for $B_i$, relative to the robot orientation $\theta$. A triangulation algorithm computes the robot position and orientation based on these angles (three or more).

geometrical setups or to deal with harsh environments where some beacons might be out of sight of the sensor [5], [26], [38], [42]. Note that algorithms dealing with multiple beacons can also work with only three beacons. For our application, we developed a new triangulation algorithm, described in [34].

This paper describes a new angle measurement system, independent of any positioning algorithm. In practice however, angle measurement systems are developed almost exclusively for positioning, as the process of triangulation requires angle measurements. This explains why most authors only evaluate positioning algorithms or present complete systems (hardware and software). As a consequence, it is rare that authors evaluate the performance of the underlying angle measurement system, and there is a lot of confusion about the evaluation criteria (accuracy, precision, resolution, etc). But the quality of positioning depends on the relative configuration of the robot and beacons, and it is essential to evaluate angle measurement separately to improve complete systems. For that reason, we want to evaluate BeAMS directly, and not through a positioning algorithm. In the following, we propose a complete framework to evaluate our system, including two important criteria: the precision (variance), and the accuracy (bias).

The hardware of BeAMS has already been briefly introduced in one of our previous paper [31]. A summary of our statistical study on codes and some experiments were presented in another paper [32]. In this paper, we provide an extended description of the hardware and explain all the design parameters and trades-off. We also extend the statistical analysis, discuss the relevance of several comparison criteria, provide a detailed evaluation of the performance of our system, and compare it to other similar systems.

The paper is organized as follows. Section II presents some of the numerous angle measurement systems developed for robot positioning. The hardware of our new angle measurement system is described in Section III. The angle measurement principles are explained in Section IV. We discuss the parameters and trades-off involved in our system in Section V, and discuss a practical system deployment. A theoretical model, useful for evaluating the performance of our system is detailed in Section VI. Then, in Section VII, we compare the model to simulation data and real measurements. Finally, Section VIII concludes the paper.

## II. RELATED WORK

As explained by Borenstein *et al.* [7], no universal indoor positioning system exists, contrasting with the widespread use of GPS for outdoor applications. Some surveys on indoor positioning systems may be found in [6], [7], [18], [28]. Technologies used in these systems may be as varied as lasers, IR, Ultrasound, RF including RFID, WLAN, Bluetooth and UWB, magnetism, vision-based, and even audible-sound. In this study, we concentrate mostly on angle measurement systems, although some of the systems include other forms of measurement, such as range. Hereafter, we present a selection of popular commercial systems and then some "home-made" systems found in the literature, all based on beacons.

### A. Commercial systems

Most commercial systems are described by Borenstein *et al.* in [6], and also by Zalama *et al.* in [42] (NAMCO *LaserNet*, DBIR *LaserNav*, TRC *Beacon Navigation System*, SSIM *RobotSense*, MTI Research *CONAC*, SPSi *Odyssey*, *LS6* from Guidance Control Systems Ltd., NDC *LazerWay*, SICK *NAV200*). Almost all of these systems use an on-board rotating laser beam sweeping the horizontal plane to illuminate retro-reflective beacons. The horizontal sweeping is generally performed with a fixed laser emitter and receiver combined with a $45\,deg$ tilt mirror mounted on a motor. The angular position of the motor is given by an angular encoder attached to the motor shaft. The beacons are generally simple passive retro-reflectors reflecting the light back to the sensor on the mobile robot. Systems using passive retro-reflectors cannot differentiate between beacons, which makes the task of positioning harder. Furthermore a positioning algorithm working with indistinguishable beacons needs an initial position in order to work properly [21], [42]. In addition, if beacons are not identifiable, the algorithm could fail in the following two cases: at the wake-up (robot start up or reboot) or when the robot is kidnapped (*i.e.* displaced).

To overcome these issues, some systems use variants such as bar-coded reflective tapes to identify the beacons (for example *LaserNav*, as used by Loevsky and Shimshoni [26], or robot *HILARE* [3]). Another technique for identifying beacons consists of using networked active beacons with an additional communication channel (typically an RF channel). When they are hit by the rotating laser, the beacons communicate to compute the angles between them and send the angles back to the mobile through an RF link (MTI Research *CONAC*). The difficulty with this system is the setup of the networked beacons. The SPSi *Odyssey* system (used by Beliveau *et al.* [4]) is different, since it can position a mobile in 3D. The beacons are laser transmitters and the receiver is located on the mobile robot. This system is not able to compute the heading of the mobile (unlike on-board angle measuring systems), except while the system is moving (as in the case of the GPS). Moreover, the field of view of the emitters is limited to $120\,deg$ horizontally and $30\,deg$ vertically, which

makes the positioning possible within a limited volume of space. Nowadays, positioning systems have a full $360\,deg$ coverage, except for the *Odyssey* and the older *LaserNet* ($90\,deg$) systems.

It turns out that most commercial systems use rotating lasers combined with retro-reflective beacons. They generally have a good accuracy and working range, but they cannot differentiate between beacons, apart from the *LaserNav* system, which is no longer manufactured. Finally, they are expensive and take up too much space, which makes them inappropriate for small educational robots. We will now describe home-made systems found in the literature.

### B. Non-commercial systems

*1) Rotating lasers:* One particular famous non-commercial system is the *Imperial College Beacon Navigation System* [35]. The principle involved in this system appears to be exactly the same as for the *CONAC* system. This system uses a rotating laser and networked active beacons connected to a base station that sends position back to the mobile via an RF channel. The main drawback of these systems is the wiring and setup of the beacons. To overcome this issue, a more recent system, similar to the *Imperial College Beacon Navigation System* and *CONAC,* is presented by Zalama *et al.* [42]. It uses an on-board rotating laser and active beacons that send their identifier back to the mobile with some RF coded pulses when they are hit by the rotating laser. The beacons are totally independent and stand-alone (no network, communication cables or base station), which makes the setup easier.

Even if these systems solve the beacon identification problem, there is an open issue: how do such systems behave when multiple robots use the same setup of beacons? A beacon would send its identifier back to all robots even if only one of them has hit that beacon, causing false angle measurements to the other robots. So we guess that these systems are inadequate for being used by multiple robots simultaneously.

*2) Static receivers:* In general, the $360\,deg$ horizontal field of view is covered by a single receiver combined with a rotating system. However, it is possible to cover the whole horizontal plane without mechanical part, as explained hereafter. The first type of static sensor system uses multiple static receivers uniformly distributed on the perimeter of a circle. These systems measure the angles to the beacons by simply "looking" at which receiver receives the signal from a beacon. Since more than one receiver can receive the same signal, an interpolation can be performed to improve the angular position of a beacon, as highlighted by Gutierrez *et al.* [19] and Roberts *et al.* [37]. These systems generally also derive a distance to the beacons. For example, some of these systems [19], [23], [36], [37] use the infrared received signal strength to compute the distance to the beacon, in addition to the bearing information. In [20], Hernandez *et al.* compute the distance using the aperture angle of the received signal (time taken to sweep the receiver). In [13], Durst *et al.* use Nintendo Wii cameras instead of infrared receivers to localize and identify the different beacons. Lee *et al.* [25], and Arai and Sekiai [1] use infrared light from beacons and measure the incident angle of the infrared light with two fixed photodiodes and a specialized circuit. Another similar idea consists in the use of only one static receiver or laser emitter. The $360\,deg$ field of view is obtained by the rotation of the robot itself, which is expected to move to see the beacons [29], [40]. The main drawback is that the position update rate depends on the robot movements and is generally low compared to other systems.

These systems have the benefit of being small, lightweight, and simple (no moving part). Unfortunately, it turns out that these systems are less accurate ($5 \rightarrow 10\,deg$) than rotating sensors ($0.05 \rightarrow 0.5\,deg$), and that the accuracy of the angles depends on the number of sensors. They are often used by swarm of robots for relative positioning and communication, but not for precise global positioning.

*3) Panoramic cameras for detecting beacons:* The second type of static sensor systems uses panoramic cameras to measure angles or distances. A common way to measure angles with a static camera and without moving parts is to transform it into an omni-directional camera via a catadioptric mirror, fish-eye lens, or a reflecting ball, as proposed by Betke and Gurvits [5]. With this configuration, a $360\,deg$ horizontal field of view of the scene is taken in one image. The angular positions of the beacons are computed through image processing by searching the beacon patterns whithin a circular region of the image. Jang *et al.* [22] base their system on the same principle with only one beacon, but they also compute the distance to that beacon.

One distinctive feature of panoramic cameras is that angles to beacons are measured at the same time, in one image. This can be an advantage if the positioning algorithm uses a triangulation technique directly. This advantage is useless if the angles are fed into an Extended Kalman Filter, which can take advantage of one angle at a time. Panoramic cameras also need a more complicated image processing algorithm, and they depend highly on lightning conditions. Finally, like the multiple static receivers, they are less accurate than rotating systems.

*4) Most closely related systems:* One of the oldest systems, related to ours, is described by McGillem and Rappaport in [27]. That system is made up of beacons emitting infrared modulated signals and a rotating infrared detector mounted on a turntable to measure the angles to the beacons. Another recent and similar system is proposed by Brkic *et al.* in [9]. This system relies on infrared beacons and a rotating receiver; a brushless DC motor with rotary transformer overcomes the problem of contact-less power supply, and ensures signal transfer. Unfortunately, no information about motor control, infrared codes, or angle calculation is provided in that paper. Finally, the accuracy of the system is given in terms of distance errors on the moving area, and no information about the accuracy of measured angles is given. Kemppainen *et al.* [24] also describe a system similar to ours for multi-robot spatial coordination, the system being used for inter-robot relative positioning, not global positioning. The difference with the previous systems is that the infrared emitting beacons are located onto the robots themselves, instead of being at fixed locations. In addition to the angle measurement, the system estimates the range by the received signal strength. Using

| **Commercial systems** | **performance** | **acquisition rate** | **working range** | **size** | **weight** | **cost** |
|---|---|---|---|---|---|---|
| SICK Nav200 | $0.1\,deg$ | $8\,Hz$ | $28\,m$ | $(12 \times 18 \times 18)\,cm^3$ | $3.3\,kg$ | $\$5,000$ |
| GCS LS6 | a fraction of a degree | $8\,Hz$ | $40\,m$ | not available | $2.8\,kg$ | $\$3,400$ |
| NDC LazerWay | $0.1\,deg$ | $6\,Hz$ | $70\,m$ | $(19 \times 19 \times 16)\,cm^3$ | $2.7\,kg$ | not available |
| DBIR LaserNav | $0.03\,deg$ | $10\,Hz$ | $30\,m$ | $(30 \times 30 \times 38)\,cm^3$ | $4.4\,kg$ | $\$8,000$ |
| TRC Beacon Nav. Sys. | $0.125\,deg$ | $1\,Hz$ | $24\,m$ | $(10 \times 10 \times 10)\,cm^3$ | not available | $\$11,000$ |
| SSIM RobotSense | $0.17\,deg$ | $10-40\,Hz$ | $30\,m$ | not available | not available | $\$12,800$ |
| **Prototypes** | **performance** | **acquisition rate** | **working range** | **size** | **weight** | **cost** |
| Zalama *et al.* [42] | $0.6\,deg$ max. (@ $10\,Hz$) | $1-10\,Hz$ | $40-50\,m$ | $(6 \times 11 \times 19)\,cm^3$ | $0.4\,kg$ | $\$600^{**}$ |
| Kemppainen *et al.* [24] | $1.5\,deg$ | not available | $3\,m$ | not available | not available | not available |
| Brkic *et al.* [9] | $5-10\,cm$ $(2\,deg)$ | $10\,Hz$ | not available | $(8 \times 8 \times 8)\,cm^3$ | not available | not available |
| BeAMS | $0.24\,deg^{\dagger}$ | $10\,Hz$ | $6\,m^{\star}$ | $(8 \times 8 \times 6)\,cm^3$ | $0.2\,kg$ | $\$350^{**}$ |

†This value includes the variance (precision), and the bias (accuracy) of the measures (see Section VII-F).
⋆The prototype is optimized for that range value, typical for the EUROBOT contest. In Section V-B, we explain how to increase the working range up to $36\,m$.
**The cost is calculated for the hardware components only (one sensor and three beacons).

**Table I:** Comparison of different angle measurement systems.

the bearing and the range, a robot can compute the relative position of all other robots.

These systems (emitting beacons, and rotating receiver) are able to identify the beacons while using only one communication channel (the beacon signal itself). Due to the nature of this unidirectional channel, multiple receivers (robots) can receive the signals from the same beacons at the same time without disturbing each other (like for the GPS system). But, unlike reflective tape, beacons have to be powered up.

### C. Summary

There is a large variety of angle measurement systems. Some systems do not identify the beacons, and others require more than one communication channel. Some systems cannot position multiple robot simultaneously. If we compare the values found in the literature, it turns out that rotating sensors are more accurate than fixed sensors, but have the disadvantage that information and power have to be transmitted to the sensors, if these are located on the turning part of the system. A fixed sensor can be used, if combined with a mirror and a motor to sweep the horizontal plane and cover a $360\,deg$ field of view. With a mirror, the light rays are redirected to cross the rotation center of the turning system. In general, the mirror is mounted on a hollow gear, which is driven by the motor through a gear or belt, allowing light rays to pass and reach the sensor. This solution tends to make the mechanical part of rotating systems more complicated and cumbersome. It turns out that the most flexible solutions are rotating lasers with passive bar-coded reflective tapes or active emitting beacons with a rotating receiver. This last solution requires to power-up the beacons. A comparative table of some commercial and home-made systems and their characteristics is provided in Table I. For some systems, implementation characteristics are missing or incomplete, such as the working distance, power consumption, dimensions, etc. Finally, evaluation criteria such as precision (variance), accuracy (bias) and resolution (number of steps for one turn) are sometimes confused during the performance analysis. Also, some systems are evaluated through a positioning algorithm, and therefore it is a hard task to evaluate the quality of the underlying angle measurements to compare systems.

The comparative table shows that BeAMS is small and lightweight. In addition, BeAMS proposes a new mechanism for angle measurement and uses an unsynchronized channel with coded signals to identify beacons.

## III. HARDWARE DESCRIPTION OF A NEW SYSTEM

While there are many angle measurement systems, none of them was suited for our application, as explained hereafter. Our first motivation for this work was to create a new system for the EUROBOT contest[1], which imposes many constraints. For the positioning part, the most important constraints are: (1) the available volume for the hardware on the robot is limited to $(8 \times 8 \times 8)\,cm^3$, (2) home-made laser systems are prohibited except if they are manufactured and kept in their house cases. The EUROBOT contest is a harsh environment for robot position. Firstly, as collisions and shocks are numerous, the knowledge of beacons IDs is an advantage to be robust to the wake-up or kidnapped issues. Secondly, the environment is polluted by many sources of noise including infrared, lasers, radio waves, and ultrasound signals. Also the lightning conditions are very bad and there are lots of shiny or reflective surfaces. Finally, more than one robot per team may evolve on the field.

Considering all these constraints, the system has to identify the beacons, use coded signals, and allow multiple robots. Commercial system were unsuitable because of their sizes, their high price, and because they cannot identify the beacons. Home-made laser systems are prohibited. Static receivers do not provide the accuracy needed for this contest. Finally we wanted to use a triangulation based positioning to estimate the robot heading precisely (this is important since the heading is highly downgraded by odometry). So we designed an angle measurement system based on beacons emitting infrared coded signal and a rotating receiver. Note that, to our knowledge, there is only one very similar system, designed by Brkic *et al.* [9], also for the EUROBOT contest. But, according to the authors, their system is not accurate enough to position the robot (see Table I).

[1]http://www.eurobot.org/

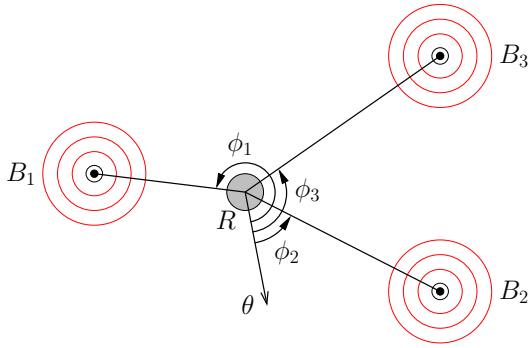



**Figure 3:** Schematic top view representation of the system. The system is composed of: (1) several active beacons $B_i$ emitting infrared light in the horizontal plane, and (2) a sensor located on the robot $R$. The aim of the sensor is to measure the azimuthal angles $\phi_i$ of the beacons in the robot reference determined by $\theta$.

BeAMS is original but it has the same limitations as any other optical system, as explained in [7], [28], [42]. First, a line of sight between beacons and sensor has to be maintained for the system to work. Also, the reflections of beacon signals on shiny surfaces can lead to false detections. Finally, the sensor could be blinded by direct sunlight (causing the SNR to decrease). This has the effect of reducing the working range in outdoor conditions.

### A. Architecture of BeAMS

The hardware of BeAMS consists of a sensor located on the robot, and several active beacons emitting infrared light, located at know positions. This configuration is represented in Figure 3. As illustrated in this figure, the aim of the sensor and processing unit is to determine the identifier of each beacon $i$, as well as its azimuthal angle $\phi_i$, in the robot reference, whose orientation is given by $\theta$. The sensor is composed of an infrared receiver/demodulator and a motor. The beacons are infrared LEDs whose signal is modulated. To achieve the angle measurements, the infrared receiver is combined with the motor turning at a constant speed. One of the key elements of our system is that the receiver sweeps the horizontal plane at constant speed, and that the relationship between the angle and time is very precise. In order to identify beacons and to increase robustness against noise, each beacon sends out a unique On-Off Keying (OOK) amplitude modulated signal over a $455\,\mathrm{kHz}$ carrier frequency. Furthermore, BeAMS only requires one infrared communication channel; there is no synchronization channel between the beacons and the robot, which allows multiple robots to share the same system. Finally, the mechanical part of the system is kept as simple as possible (motor only), with no gear system or belt, thanks to the hollow shaft, and no optical encoder for the motor control or angle measurement is needed. These features make BeAMS a small, low power, flexible, and tractable solution for robot positioning. BeAMS has been continually improved since its inception and has been used successfully in the EUROBOT contest for the last four years. In the next sections, we describe the hardware components of our system.

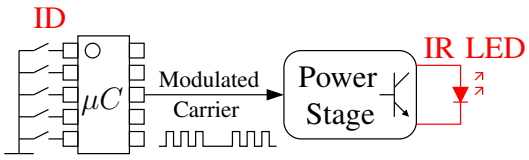



**Figure 4:** Architecture of a beacon. The central element of a beacon is an infrared LED. A PIC microcontroller ($\mu C$) generates the appropriate signal to drive the IR LEDs through the power stage. Each beacon emits its own unique IR signal continuously so that the receiver can determine the beacon identifier (ID).

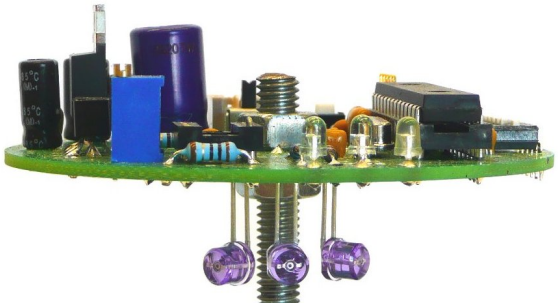


**Figure 5:** Picture of a beacon. The main part of a beacon is made by IR LEDs, which are located under the printed circuit board, parallel to the moving area and directed towards the center of the moving area.

### B. Description of the beacons

The core of a beacon is composed of IR LEDs (SFH485P); they emit signals in a plane parallel to the moving area. These LEDs have a large emission beam so that a small number of LEDs per beacon can cover the whole area. A PIC microcontroller generates the appropriate signal to drive the IR LEDs through the power stage. Each beacon continuously emits its own unique IR signal so that the receiver can determine the beacon's identifier (ID). Figure 4 represents the schematic architecture of a beacon and Figure 5 shows a picture of a beacon. The power consumption is $100\,mA$ at $9\,V$, for a working distance of up to $6\,m$.

### C. Description of the sensor

As shown in Figure 6, the sensor is composed of a mini stepper motor, a $12\,mm$ convergent lens, a small front surface mirror with a $45\,deg$ tilt, a polycarbonate light guide placed in the center of the motor shaft (which has been drilled for this purpose), an IR receiver (a TSOP7000 from VISHAY) and an optical switch used to calibrate the zero angle reference $\theta$ (see Section IV-B). The lens and mirror are placed on a "turret", which is fixed to the motor shaft. The receiver is fixed to the bottom of the motor, just under the light guide. This configuration allows IR signals from a beacon to reach the fixed receiver through the entirely passive "rotating turret" and light guide. As a result, the receiver can virtually turn at the same speed as the turret. By introducing this original disposition of optical elements into our system, the system behaves as if the receiver is turning without the mechanical constraints and inconvenience. Finally, a PIC microcontroller is used to drive the motor through its controller and to decode the output

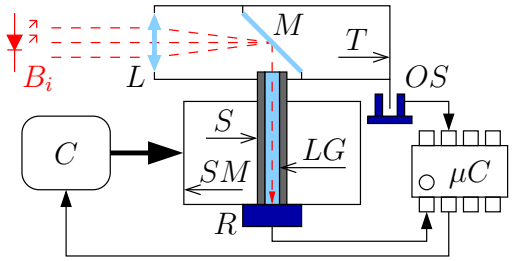



**Figure 6:** Schematic representation of the receiver. $B_i$ is a beacon emitting IR light, $L$ is the lens, $M$ is the mirror, $LG$ is the light guide, $R$ is the receiver, $SM$ is the stepper motor, $S$ is the hollow shaft, $T$ is the turret, $C$ is the motor controller, $\mu C$ is the PIC microcontroller, and $OS$ is the optical switch.

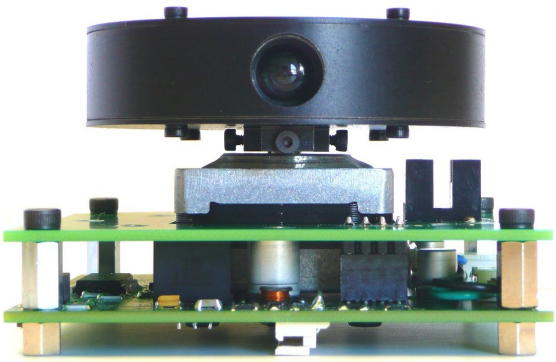


**Figure 7:** Picture of the sensor with the rotating turret in black (top) and the lens, the stepper motor (middle), the optical switch (middle right), and the electronic card (bottom).

of the receiver. Figure 7 shows a picture of the sensor. The entire sensor weights $195\,g$, and the power consumption is $47\,mA$ at $9\,V$. Now that the hardware elements have been presented, we detail some elements of the system: software architecture, motor control, angle measurement principle, and infrared codes.

## IV. MEASUREMENT PRINCIPLES OF BEAMS

### A. Software description

The software building blocks of BeAMS are drawn in Figure 8. The key principle of the software is to use a common timer to drive the stepper motor at a constant speed, and to capture the receiver output edges. The receiver output is connected to a capture module in the microcontroller. On a falling or a rising edge of the receiver output, this module latches (captures) the actual timer value to a register that may be read later by the software. This allows us to associate a time to each incoming event (falling or rising edge). And as the value of the timer is perfectly linked to the motor angular position, the association of an incoming receiver event to an angular position is as accurate as possible. The captured values serve to compute the angular position of the beacons and their IDs.

### B. Stepper motor control

The stepper motor is driven in an open loop via an input square signal to advance the motor step by step. The stepper

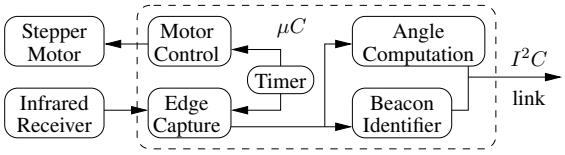



**Figure 8:** Software organization of BeAMS. $\mu C$ is the PIC microcontroller. A common timer is used to drive the stepper motor at a constant speed and to capture the edges of the receiver output. These captured values are used to compute the angular positions of the beacons as well as the beacon identifiers.

motor has 200 real steps and is driven in a half-step mode via its controller, which turns the number of steps into 400. The frequency of the step signal controls the rotation speed of the motor and is derived from the common timer. Since the timer is 156 times faster than the step signal, we achieve a sub-step time resolution so that the number of "virtual" steps is $400 \times 156 = 62400$ exactly. The motor turns at a constant speed $\omega$ and the angular position of the turret/receiver $\phi$ is thus proportional to the value of this timer. Whereas the motor is controlled step by step, the rotation is assumed to be continuous due to the high inertia of the turret compared to the motor dynamics. Since the motor turns at a constant speed, the common timer value can be seen as a linear interpolation of the motor position between two real steps of the motor.

This kind of control in open loop with a stepper motor is possible since the torque is constant and only depends on the turret inertia and motor dynamics, which are known in advance. The advantage of this approach is that we do not need a complicated control loop or expensive rotary encoder in order to detect the position of the turret with precision. Indeed, the common timer acts as a rotary encoder, and the position of the turret can be obtained by reading the value of the timer. As explained earlier, there are 62400 "virtual" steps of the motor. The angular resolution is thus given by $^{360}/_{62400} = 0.00577\,deg$ $(0.1\,mrad)$. The timer clock runs at $625000\,Hz$, to give an angular speed of $^{625000}/_{62400} = 10.016\,^{\text{turn}}/_{\text{s}}$. Since the motor can start from or stop at any angular position, an optical switch (denoted $OS$ in Figure 6) is used to calibrate the zero angle reference $\theta$ by reading the timer value when the turret passes through the switch.

### C. Angle measurement mechanism

Let us denote by $\phi$ the current angular position of the turret/receiver, relatively to $\theta$. As the turret turns at a constant speed $\omega$, the angular position $\phi$ is directly proportional to time

$$\phi(t) = \omega\, t. \tag{1}$$

As a result, we can talk about either time or angular position indifferently. For convenience, we prefer to refer to angles instead of time units.

The TSOP7000 is a miniaturized IR receiver that acts as an OOK demodulator of a 455 kHz carrier frequency. The input is a modulated signal whose carrier wave is multiplied by the "0" or "1" binary message. The receiver outputs a value "1" when it detects the carrier wave and "0" otherwise. No other information is given by the receiver. By design, the

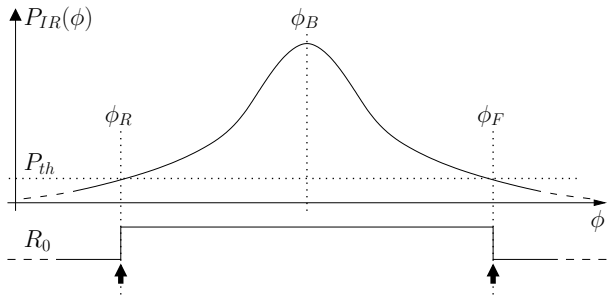



**Figure 9:** The upper curve $P_{IR}(\phi)$ is the expected infrared power collected at the receiver while the turret is turning. $R_0$ represents the receiver output for a non modulated infrared carrier wave (pure 455 kHz sine wave). The black arrows represent the measured values respectively for $\phi_R$ to the left (first *Rising* edge) and for $\phi_F$ to the right (last *Falling* edge).

receiver combined with the optical components has a narrow field of view and, consequently, the amount of infrared power collected at the receiver, denoted by $P_{IR}(\phi)$, depends on the angle. This power also depends on the power emitted by the beacon, and the distance between the beacon and the receiver. Note also that the exact shape of $P_{IR}(\phi)$ depends on the hardware, that is the receiver, optical components, and the geometry of the turret. It is impossible to derive the precise power curve from the specifications, because we only have access to the demodulated signal, and no information about power is available. Therefore, we make some basic assumptions regarding the shape of $P_{IR}(\phi)$; the resulting expected curve $P_{IR}(\phi)$ is shown in Figure 9. The exact shape of this curve does not have much importance in this study but is assumed to increase from a minimum to a maximum and then to decrease from this maximum to the minimum. In the following theoretical developments, we make three important assumptions about the curve and the detection process itself:

1) The maximum coincides with an angle, which is the angular position of the beacon, denoted $\phi_B$ (*i.e.* the angle we want to measure). As a result, for any angle $\phi$

$$P_{IR}(\phi) \leq P_{IR}(\phi_B). \tag{2}$$

2) The curve is supposed to be symmetric around the maximum since the turret and all optical components are symmetric. This means that

$$P_{IR}(\phi_B - \phi) = P_{IR}(\phi_B + \phi). \tag{3}$$

3) Finally, we assume that the receiver reacts to $0 \rightarrow 1$ and $1 \rightarrow 0$ transitions at the same infrared power threshold $P_{th}$, respectively at angles $\phi_R$ and $\phi_F$

$$P_{IR}(\phi_R) = P_{IR}(\phi_F) = P_{th}. \tag{4}$$

This hypothesis will be discussed later, in Section VII-F.

From equations (3) and (4), we derive that $\phi_B - \phi_R = \phi_F - \phi_B$ and that

$$\phi_B = \frac{\phi_R + \phi_F}{2}. \tag{5}$$

This equation expresses an important innovation that has two benefits: (1) we derive the angle of the beacon not from the maximum power, but from two angle measurements that take the narrow receiver optical field of view into account, and (2) by measuring an angular window (that is two angles) instead of one angle, it is possible to analyze the temporal evolution of the signal inside this window to determine the code of the beacon (or any other kind of useful information). Note that the angular window, defined as $\phi_F - \phi_R$, depends on the received IR power. It increases if the received power increases, and decreases if the received power decreases.

First, we assume that the beacons send a non modulated IR signal, that is a pure 455 kHz sine wave and explain the measurement principle for one beacon; the principle is the same for any number of beacons. While the turret is turning, the receiver begins to "see" the IR signal from that beacon when the power threshold $P_{th}$ is crossed upwards ($0 \rightarrow 1$ transition). The receiver continues to receive that signal until $P_{th}$ is crossed downwards ($1 \rightarrow 0$ transition). The receiver output is depicted as $R_0$ in Figure 9. At these transitions, the capture module latches values for $\phi_R$ and $\phi_F$. The angular position of the beacon is then computed after equation (5).

### D. Beacon identifier and infrared codes

The convenient assumption of continuous IR signals used in the previous section is not realistic because (1) we would not be able to distinguish between the different beacons, and (2) it is essential to establish the beacon ID (especially in a very noisy environment like the EUROBOT contest where other IR sources may exist).

In BeAMS, each beacon emits its own code over the 455 kHz carrier wave; this emission is continuous so as to avoid having any form of synchronization between the beacons and the receiver. As a result, each beacon signal is a periodic signal whose period corresponds to a particular code defining the beacon ID. The design of these codes is subject to several constraints related to (1) the receiver characteristics, (2) the loop emission, (3) the desired precision, (4) the system's immunity against noise, and (5) the number of beacons. We elaborate on these constraints below:

**1. Receiver.** The TSOP7000 requires that the burst length (presence of carrier wave) be chosen between 22 and 500 $\mu s$, the maximum sensitivity being reached with 14 carrier wave periods ($^{14}/_{455000} = 30.8\,\mu s$). The gap time between two bursts (lack of carrier wave) should be at least 26 $\mu s$.

**2. Loop emission.** Because of our willingness to avoid a synchronization between beacons and the receiver, we must ensure that the periodic emission of a code does not introduce ambiguities. For example [0101] is equivalent to [1010] when sent in a loop. Thus any rotation of any code on itself must be different from another code.

**3. Precision.** The lack of synchronization between beacons and the receiver introduces a certain amount of imprecision. Indeed, the first received IR pulse may be preceded by a gap time corresponding to a zero symbol. This affects the estimation of $\phi_R$. The same phenomenon occurs for $\phi_F$. A fairly obvious and intuitive design rule would say that we have to reduce the duration of zeros, as well as their frequency of appearance. Therefore, we forbid two or more consecutive zeros, and the duration of one zero (the gap time) must be

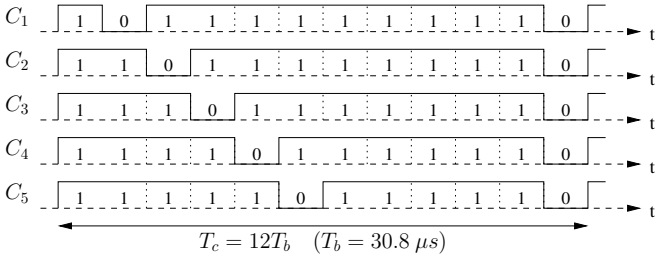



**Figure 10:** Temporal representation of the $C_1$, $C_2$, $C_3$, $C_4$ and $C_5$ codes. These codes are repeated continuously and multiply the 455 kHz carrier wave to compose the complete IR signal.

reduced as much as possible (see Section VI) for further explanations about the error due to the gap time).

**4. Immunity.** The codes should contain enough redundancy to be robust against noise or irrelevant IR signals.

**5. Number of beacons.** The codes should be long enough to handle a few beacons, but as short as possible to be seen many times in the angular window associated to a beacon, thus improving the robustness of the decoding.

All these constraints lead us to propose this family of codes:

$$C_i = [1^i\, 0^1\, 1^{2N_c - i}\, 0^1] \qquad i = 1, \ldots, N_c \tag{6}$$

where $N_c$ denotes the number of codes in the system. The duration of a bit is set to $T_b = 30.8\,\mu s$ since this value maximizes the receiver sensitivity, while respecting the minimum gap time. Although not mandatory, the duration of a one symbol has been chosen to be equal to the duration of a zero. This is to simplify the implementation of the beacons and to ease the decoding process. The gap time is the same for all codes and corresponds to the duration of a unique zero symbol. The second half part of the codes can be seen as a checksum, since it makes the number of ones constant ($2N_c$). In our current implementation, we have $N_c = 5$ codes because this is appropriate for our application. Figure 10 shows the temporal representation of the codes for $N_c = 5$. Note that any code meeting the second requirement (differentiable under loop emission) would work to identify the beacons. However, they may not meet the third requirement if the zero symbols appear in random patterns. Indeed, a thorough analysis about the error introduced by the gap time (see Section VI) shows that this error increases with the frequency of zero symbols and with the square of the zero duration. Moreover, a simulator (see Section VII-C) has been created to validate this result. This simulator helped us to compare codes *w.r.t.* the error they generate. From our experience, the codes presented in expression (6) are the best ones that meet all our requirements, but we have no formal proof of it.

The angle measurement principle still operates exactly as in Section IV-C even if the IR carrier wave is modulated by the codes. Since there are gap times in the IR signal of a beacon, there are more than two edges in the received signal. The intermediate edges are used to determine the beacon ID, by analyzing the durations of burst lengths and gap times. But the first and last edges of the received signal always correspond to our measurements of $\phi_R$ and $\phi_F$. These two edges are isolated from all other edges due to a timeout strategy, which

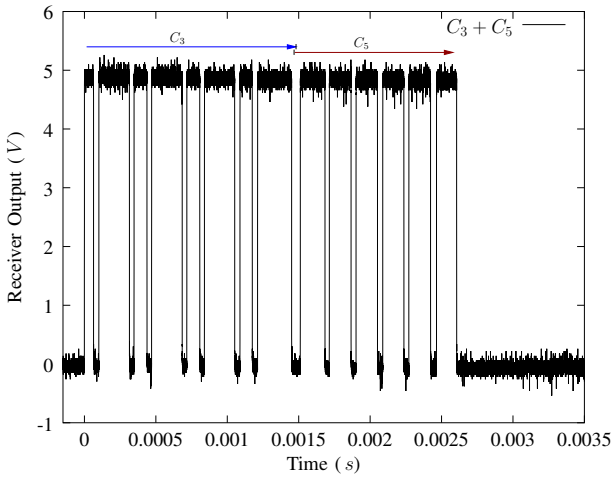



**Figure 11:** Image of an oscilloscope screen displaying the receiver output voltage. In this experimental setup, two beacons emit the codes 3 and 5, respectively. The beacons and the receiver are placed so that the signals slightly overlap. Despite that, it is possible to distinguish both codes in the signal.

relies on the fact that the separation time (or angle) between two different beacons is much greater than the separation time between consecutive edges in a code. Actually, the separation time is set to four bit durations, which corresponds to a separation angle equal to $0.44\,deg$.

Note that, for the proper functioning of the system, it is important that the receiver collects infrared light from one beacon at a time. To do this, some additional optical components are used to limit the field of view of the receiver to a narrow value (a few degrees). However, despite the narrow field of view, and the timeout strategy to separate beacons, two or more beacons might appear in the same angular window if they are close enough (from an angular point of view). When this situation occurs, the demodulated signal is composed of codes from the different beacons, and they appear in the same order as the turret turns and sees the different beacons. This means that the timeout strategy is not able to cut the different signals. Also, at the transition points, the signal could be composed of burst or gap durations that do not correspond to any code. The decoding algorithm simply adds (in counters) the number of different codes it sees, as well as bad durations. Therefore, the receiver is capable to differentiate between a pure signal from one beacon, or a compound signal from several beacons. The system can then decide to keep or reject a compound signal; this capability to check the consistency of codes is an important advantage of BeAMS. To illustrate this capability, we provide the image displayed by an oscilloscope showing the receiver output voltage for two beacons emitting the codes 3 and 5, respectively (see Figure 11). In this figure, the receiver can easily detect that two beacon signals overlap slightly.

## V. PRACTICAL CONSIDERATIONS

### A. *Parameters and trades-off*

The design of BeAMS implies many parameters and some trades-off that need to be explained. First of all, we had to

choose a turning speed (or acquisition rate). Lots of commercial or non commercial systems works at $10\,Hz$, which seems sufficient for a robot moving at moderate speed. Also, the accuracy of these systems is more likely to be $0.1\,deg$, to get a reasonable accuracy on the final position. Our system uses an OOK modulation that leads to an error due to the gap times ($T_0$). Our statistical analysis, as well as our simulator confirm this result. However, it is easy to show that the maximum absolute angular error is given by $\phi_0 = \omega\, T_0$, since the turret has turned by this angle during a gap time. This equation represents the most important trade-off: for a given receiver, increasing the rotation speed would increase the error on measured angles. We decided to choose the smallest $T_0$, and afterwards the maximum turning speed according to the maximum error accepted. In our case, the TSOP7000 was the only receiver providing the minimum gap time satisfying the pair of parameters $\phi_0$ and $\omega$. Then the optical field of view has been tuned with optical components to be narrower, but large enough to receive some bits/codes from one beacon in the angular window, for this turning speed. Typically, we receive a minimum of 20 bits ($\sim$ 2 codes) at the maximum range, which corresponds to the smallest angular window. So, for a given receiver, this maximum working distance depends on the emitted power combined with the size of the lens, and the minimum number of bits we need to identify the beacons. These parameters have been chosen to meet the EUROBOT rules. The lens/focal distance has been chosen to hold in the allowed volume. Then the emitted power has been tuned to reach the maximum distance possible on the moving area. Indeed the system works until $6\,m$, which is greater than required.

### B. System deployment

BeAMS was designed for the EUROBOT contest. However, the system could be used in any other application involving angle measurements based positioning. Two parameters are important to use BeAMS in another context: (1) the working distance, and (2) the number of beacons.

Obviously, the covered area is determined by the maximum working distance. The current version of BeAMS reaches $6\,m$ with a small lens ($12\,mm$) and usual LEDs. This distance can be increased either by increasing the size of the lens, or by rising the emitted power. In our application, the size of the lens is limited since the available volume is limited. The emitted power can be increased either by choosing other IR LEDs or by increasing the number of LEDs. For example, multiplying the emitted power $P_e$ by four, and the surface of the lens $S$ by nine would multiply the working distance $d$ by six, as $d$ is proportional to $\sqrt{SP_e}$. With our prototype, we would reach a distance of $36\,m$, which is comparable to commercial systems.

Then we have to consider the number of beacons. Although Figures 2 and 3 represent the system with three beacons, it is important to note that the sensor can measure angles for any number of beacons, three being the minimum number to achieve correct positioning. We chose a code family that allows 5 beacons because it was sufficient for our application. With the same code family, we can go up to 9 beacons since we are limited by the maximum burst length permitted by the receiver. However, we can use any other codes respecting the constraints of the receiver, as explained in Section IV-D. The number of codes is limited by the minimum number of bits we receive at the maximum working distance. This minimum number of bits received in the time window is fixed by the optical field of view combined with the rotation speed, as expressed by equation (1). As explained previously, there is a trade-off. Increasing the rotation speed decreases the time window, and subsequently the number of bits and the number of possible codes. In our application, the minimum number of bits is more or less 20, at the maximum range with a turning speed of $10\,{}^{\mathrm{turn}}/_{\mathrm{s}}$. As explained in Section IV-D, we use a checksum and we want to see the code many times in the time window. From a practical point of view, 10 of these 20 bits could be used to code the beacons, without jeopardizing the noise robustness. This allows for a maximum of 1024 beacons. But then we have to consider the beacon spacing and multiple beacon detection issue. As explained in Section IV-D, the signals from different beacons could appear in the same time window if they are too close (from an angular point of view). This has the effect of corrupting the angle measurements of those beacons. Fortunately, the system is able to detect these pathological cases and it then rejects measurements due to code collisions, unlike laser systems with retro-reflective stripes. In a practical application, code collisions occur when the beacons are almost aligned (a few degrees of angular separation). However, beacons are placed against walls or in corners, where it is unlikely to find the robot. Some papers discuss the issue of beacon placement. Algorithms to find the best place of a minimum number of beacons to meet a given criterion, most often a minimum positioning error, are proposed in [11], [39], [41]. For BeAMS, there is only one additional constraint: anywhere on the moving area, the receiver has to find at least 3 beacons, not aligned with any other beacon. Finally, note that the decision to choice a passive sensor and active beacons was guided by the EUROBOT contest. It is possible to swap some design choices, for example which elements are passive or active, for other specific industrial setups.

## VI. CODE STATISTICS AND ERROR ANALYSIS

Now that the system has been presented, we concentrate on the errors that affect angle measurements. One can identify two kinds of noise in BeAMS: (1) the natural noise, and (2) the artificial noise. Like for all other systems, BeAMS is affected by the natural noise, due to the receiver output jitter, rotation jitter, electronic noise, other infrared signals, etc. In addition to the natural noise, BeAMS is affected by another kind of noise due to the codes and the use of an OOK modulation. In order to identify the different beacons, we decided that each beacon has to send its own coded signal. Unfortunately, this strategy produces errors when no signal is sent, that is during an OFF period of the sequence, as explained in Section IV-D. In the following explanation, this is interpreted as an additional noise due to the OOK modulation mechanism. But, unlike the natural noise, the artificial noise can be controlled, and it is

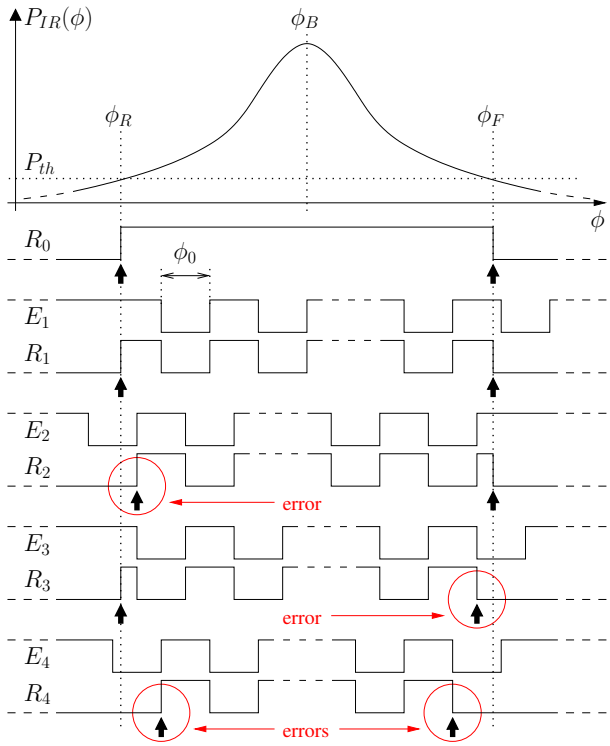



**Figure 12:** The upper curve $P_{IR}(\phi)$ is the infrared power collected at the receiver while the turret is turning. $E_i$ are examples of emitted signals from the beacons. $R_i$ are the corresponding received signals at the receiver output. $R_0$ is the special case corresponding to the non modulated infrared carrier wave (no OFF periods). The black arrows represent the measured values respectively for $\Phi_r$ to the left (first *Rising* edge) and for $\Phi_f$ to the right (last *Falling* edge). The encircled arrows emphasize errors made on $\Phi_r$ or $\Phi_f$.

important to evaluate the level of artificial error to guarantee the usability of BeAMS in real conditions. Therefore, we first focus on the artificial noise to evaluate the error made on measured angles resulting from the coding of beacon signals. The natural noise is discussed in the next section. Because of the OFF periods in the codes, there are no means to access the true values of $\phi_R$ and $\phi_F$. Therefore, we consider random variables instead, denoted by $\Phi_r$ and $\Phi_f$. According to (5), we propose the following estimator $\Phi_b$ for the true beacon angular position $\phi_B$:

$$\Phi_b = \frac{\Phi_r + \Phi_f}{2}. \tag{7}$$

### A. Description of the errors

As the receiver captures an OOK amplitude modulated signal, it can only detect the presence of the carrier wave (denoted by a 1 or ON period) or the absence of the carrier wave (denoted by a 0 or OFF period). Let us now examine the influence of the OFF periods on the first rising and last falling edges. As illustrated in Figure 12, if a beacon emits a 1 when it enters into the angular window, there is no error on $\Phi_r$. However, if a beacon emits a 0 when it enters the angular window, there is an error on $\Phi_r$ because the receiver misses the actual $0 \rightarrow 1$ transition. In fact the transition occurs later ($\Phi_r \geq \phi_R$), at the next 1. The same consideration applies to $\Phi_f$, except that the $1 \rightarrow 0$ transition could occur sooner ($\Phi_f \leq \phi_F$). All these specific situations are illustrated in Figure 12. We first represent the output of the receiver for a non modulated carrier wave, $R_0$. In that case, there are no errors in the transition times because the beacon sends out a continuous 1 symbol. The four other cases represent the output of the receiver for four different situations using an arbitrary code (we use here a simpler code than ours for the purpose of illustration, but this does not change the conclusions). The first case, corresponding to the received signal $R_1$, does not induce any error because $P_{th}$ is crossed upwards and downwards when the beacon emits a 1 symbol. The second case ($R_2$) generates an error on $\Phi_r$ only. The third case ($R_3$) generates an error on $\Phi_f$ only, and the fourth case ($R_4$) generates an error on both $\Phi_r$ and $\Phi_f$. From Figure 12, one can see that the receiver output $R_i$ is the logical AND between $E_i$ and $R_0$. Of course, this an ideal behavior of a practical receiver, and this hypothesis will be discussed later.

Assume now that the OFF periods of a sequence all have the same duration, denoted by $T_0$ (this is our choice by design). Because the motor rotates at a constant speed, an OFF period is then equivalent to an OFF angle called $\phi_0$. The worst case for estimating $\Phi_r$ occurs when an OFF period starts at an angle $\phi = \phi_R$, delaying the next transition to an angle $\phi_R + \phi_0$. The same reasoning applies to $\Phi_f$ when an OFF period begins at an angle $\phi = \phi_F - \phi_0$. In both cases, the maximum absolute error on $\Phi_r$ or $\Phi_f$ is equal to $\phi_0$. These are the worst cases but there are many combinations of these two errors. In the following sections, we establish the probability density functions (*PDFs*) of the random variables $\Phi_r$ and $\Phi_f$, and derive characteristics of the estimator $\Phi_b$.

### B. Notations

- $N_0$, $N_1$ are the number of 0's or 1's in a code, respectively.
- $p_0$, $p_1$ are the probabilities of obtaining a 0 or a 1 respectively at the IR power threshold (rising or falling edge), that is their frequencies. By definition we have $p_0 = {}^{N_0}/_{N_0+N_1}$, $p_1 = {}^{N_1}/_{N_0+N_1}$, and $p_0 + p_1 = 1$.
- $T_0$ is the OFF period (duration of a 0) in a code. The only assumption is that the OFF periods of a code must all have the same duration.
- $\phi_0$ is the OFF angle. It corresponds to the angular displacement of the turret during the OFF period $T_0$:

$$\phi_0 = \omega\, T_0. \tag{8}$$

  In our application, we have: $N_1 = 10$, $N_0 = 2$, $p_0 = {}^1/_6$, $\omega = 10.016\,{}^{\text{turn}}/_{\text{s}}$, $T_0 = 30.8\,\mu s$, and $\phi_0 = 0.111\,deg$, for each code.
- The Uniform *PDF* is defined as

$$U_{(a,b)}(x) = \begin{cases} \frac{1}{b-a} & \text{if } a \leq x \leq b, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

### C. Probability density function of $\Phi_r$ and $\Phi_f$

Errors on $\Phi_r$ originate if a beacon emits a 0 symbol while entering the angular window. Assuming time stationarity and

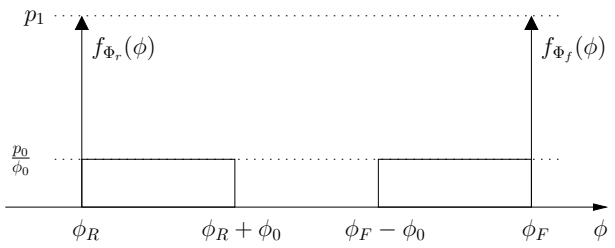



**Figure 13:** *PDFs* of $\Phi_r$ (left) and $\Phi_f$ (right).

as there is no synchronization between the beacons and the receiver, $p_1$ is the probability of determining the correct angle $\phi_R$ as the measured value for $\Phi_r$, when the beacon enters the angular window. When the beacon emits a 0, the value measured for $\Phi_r$ is not correct; we then assume that its value is uniformly distributed between $\phi_R$ and $\phi_R+\phi_0$. Therefore, if we define $\delta(x)$ as the DIRAC delta function, then the *PDF* of $\Phi_r$ is given by the following mixture of *PDFs*

$$f_{\Phi_r}(\phi) = p_1\delta(\phi-\phi_R) + p_0\, U_{(\phi_R,\phi_R+\phi_0)}(\phi), \tag{10}$$

for $\phi \in [-\pi,\, \pi)$. After some calculations, we obtain the mean and variance of $\Phi_r$ :

$$\mu_{\Phi_r} = \phi_R + p_0\frac{\phi_0}{2}, \tag{11}$$

$$\sigma^2_{\Phi_r} = p_0\frac{\phi_0^2}{3} - p_0^2\frac{\phi_0^2}{4}. \tag{12}$$

Because the configuration is symmetric when the beacon exits the angular window, a similar result yields for $\Phi_f$

$$f_{\Phi_f}(\phi) = p_1\delta(\phi-\phi_F) + p_0\, U_{(\phi_F-\phi_0,\phi_F)}(\phi), \tag{13}$$

for $\phi \in [-\pi,\, \pi)$. The mean and variance of $\Phi_f$ are

$$\mu_{\Phi_f} = \phi_F - p_0\frac{\phi_0}{2}, \tag{14}$$

$$\sigma^2_{\Phi_f} = p_0\frac{\phi_0^2}{3} - p_0^2\frac{\phi_0^2}{4}. \tag{15}$$

The *PDFs* of $\Phi_r$ and $\Phi_f$ are shown in Figure 13. Their expectations have a bias given by $\pm p_0\frac{\phi_0}{2}$ respectively (see equations (11) and (14)), and their variances are equal.

### D. Characterization of the estimator $\Phi_b$

In order to estimate the quality of the beacon angle estimator $\Phi_b$, we need to evaluate some statistics of $\Phi_b$, such as its mean and variance.

For two random variables $X$ and $Y$, we know that $E\{X+Y\} = E\{X\} + E\{Y\}$ (see [30, page 152]). Therefore, according to (11) and (14), the mean of $\Phi_b$ is given by

$$\mu_{\Phi_b} = \frac{E\{\Phi_r\} + E\{\Phi_f\}}{2} = \frac{\phi_R+\phi_F}{2} = \phi_B. \tag{16}$$

The mean of $\Phi_b$ is thus unbiased, despite the fact that both the entering angle $\Phi_r$ and the leaving angle $\Phi_f$ estimators are biased. This justifies the construction of a symmetric receiver and the choice of that particular estimator.

Let us now derive the variance of $\Phi_b$ (see [30, page 155])

$$\sigma^2_{\Phi_b} = \frac{\sigma^2_{\Phi_r} + \sigma^2_{\Phi_f} + 2C\{\Phi_r,\, \Phi_f\}}{4}, \tag{17}$$

where $C\{\Phi_r, \Phi_f\}$ denotes the covariance of $\Phi_r$ and $\Phi_f$. If $\Phi_r$ and $\Phi_f$ are uncorrelated, then we have [30, page 155]

$$\sigma^2_{\Phi_b} = \frac{\sigma^2_{\Phi_r} + \sigma^2_{\Phi_f}}{4} = \frac{\sigma^2_{\Phi_r}}{2} = \frac{\sigma^2_{\Phi_f}}{2}, \tag{18}$$

as $\sigma^2_{\Phi_r} = \sigma^2_{\Phi_f}$. However, the non correlation or independence of $\Phi_r$ and $\Phi_f$ are questionable in our case, as explained hereafter. As we mentioned earlier, four situations are possible during the angular window defined by $\phi_R$ and $\phi_F$: (1) no error is encountered, (2) an error occurs for $\Phi_r$ only, (3) an error occurs for $\Phi_f$ only, or (4) an error occurs for both angles. But, depending on the rotating speed and the code, it is possible to find particular values for the angular window $\phi_F - \phi_R$ for which it is impossible to have an error on $\Phi_r$ and $\Phi_f$ simultaneously (because the codes are deterministic and not random, and the durations between OFF periods are fixed and known). Note that, regardless of the relationship between $\Phi_r$ and $\Phi_f$, the mean of $\Phi_b$ is always given by equation (16) and $\Phi_b$ remains unbiased. In [32], we have established an upper bound on $C\{\Phi_r,\, \Phi_f\}$, but which over estimates the variance of $\Phi_b$. Here, we provide a more accurate result. The covariance can be expanded as [30, page 152]

$$C\{\Phi_r,\, \Phi_f\} = E\{\Phi_r,\, \Phi_f\} - E\{\Phi_r\}\, E\{\Phi_f\} \tag{19}$$

where $E\{\Phi_r,\, \Phi_f\}$ is the joint expectation of $\Phi_r$ and $\Phi_f$. So, in order to compute this covariance, we should express the joint *PDF* of $\Phi_r$ and $\Phi_f$ for all possibilities, depending of the angular window and the codes. It can be shown (but this is beyond the scope of this paper[2]) that the highest value of the variance occurs when no error is possible on $\Phi_r$ and $\Phi_f$ simultaneously (an error on $\Phi_r$ is not balanced by an error on $\Phi_f$, or vice versa). In that case, the joint *PDF* is given by

$$\begin{aligned} f_{\Phi_r\Phi_f}(\phi_r,\phi_f) &= (p_1-p_0)\,\delta(\phi_r-\phi_R)\,\delta(\phi_f-\phi_F) \\ &+ p_0\delta(\phi_r-\phi_R)\, U_{(\phi_F-\phi_0,\phi_F)}(\phi_f) \\ &+ p_0\delta(\phi_f-\phi_F)\, U_{(\phi_R,\phi_R+\phi_0)}(\phi_r) \end{aligned} \tag{20}$$

and the joint expectation can be computed as

$$E\{\Phi_r,\, \Phi_f\} = \phi_R\phi_F + p_0(\phi_F-\phi_R)\frac{\phi_0}{2}. \tag{21}$$

The substitution of $E\{\Phi_r,\, \Phi_f\}$ by its value into equation (19) yields $C\{\Phi_r,\, \Phi_f\} = p_0^2\frac{\phi_0^2}{4}$. This result combined with equation (17) finally yields the upper bound of the variance of $\Phi_b$:

$$\max \sigma^2_{\Phi_b} = p_0\frac{\phi_0^2}{6}. \tag{22}$$

For the parameter values of BeAMS, this variance is 14 % larger than the one given by equation (18), when $\Phi_r$ and $\Phi_f$ are supposed to be uncorrelated.

As expected, the variance is related to the presence of OFF periods in the codes. More precisely, the variance is proportional to the probability of having a zero $p_0$, and to the square of the OFF angle $\phi_0$. It is equal to zero if and only if there is no OFF period in the codes. So, this expression establishes that $p_0$ and $\phi_0$ should be kept as small as possible

[2]The complete demonstration is presented in a technical report [33], available at http://hdl.handle.net/2268/144734

to minimize the effects of the OOK modulation. Note that this variance is an upper bound, since it represents the worst case (no error on $\Phi_r$ and $\Phi_f$ simultaneously), and that this upper bound is the same for all codes (since they all have the same $p_0$ and $\phi_0$ by design). In the next section, we discuss these results and compare them to simulated values and to experimental data.

## VII. SIMULATED AND EXPERIMENTAL RESULTS

The goal of this section is to provide an error measure for BeAMS. In particular, we want to provide values for the precision (variance) and for the accuracy (bias) of the measured angles.

Our study of the related work has shown that the terms precision, accuracy, and even resolution are sometimes confused. This is unfortunate because the knowledge of these characteristics are useful for the data fusion algorithms to take measures into account properly, *w.r.t.* other measurements. They are also useful to compare systems. Also, some authors characterize their angle measurement systems through a positioning algorithm, and expresses quality results in meters.

It is a well known fact that a positioning process based on angles, regardless of its implementation, depends on the relative configuration of beacons and the robot [10], [14], [15]. So, we believe that an angle measurement system should not be evaluated through a positioning algorithm, unless a common procedure is described and used by everyone. Moreover, this evaluation procedure is difficult to implement in practice, and it adds errors due to the setup, especially errors on measurements and on the real location of beacons [26].

In the previous section, we have provided the upper bound for the additional variance on $\Phi_b$ due to the codes, and showed that the estimator is unbiased. But, in a practical situation, we have to take into account the natural (noise) variance of the system by taking real measurements. This noise is inherent in the hardware, even for a non modulated carrier wave (with no OFF periods). This noise originates from the quartz jitter, rotation jitter, etc, and, to a larger extent, from the receiver jitter at the $0 \rightarrow 1$ and $1 \rightarrow 0$ transitions. From a theoretical point of view, it is acceptable to consider that both noises are independent and, therefore, that the total noise is the sum of the natural noise and $\sigma^2_{\Phi_b}$, the power of additional noise induced by the OOK modulation.

The purpose of this section is fourfold: (1) analyze the impact of the code (via the $p_0$ and $\phi_0$ parameters) on the variance of $\Phi_b$, (2) validate the upper bound on the variance of $\Phi_b$, (3) verify if the artificial noise is independent of the natural noise, and (4) provide values for the precision and accuracy. In order to complete these analyses, simulations and measurements are performed with one beacon for several codes and angular windows.

### A. Adding codes for tests only

We have shown that the upper bound of $\sigma^2_{\Phi_b}$ depends on $p_0$ and $\phi_0$. By design, the codes all have the same $p_0$ and $\phi_0$. This is an important advantage because this implies that the additional variance is not related to any particular code. As a consequence, we have to create other codes to observe the influence of $p_0$ and $\phi_0$. Also, in order to measure the natural variance, we have to use a special code with no OFF period. So, the codes used for testing purposes are the constant code $C_k$ (no OFF period), the real $C_5$ code ("111110111110"), and two variations of code $C_5$ with increasing OFF durations ("11111001111100", "1111100011111000"). These four codes have a zero symbol probability $p_0$ respectively equal to 0, $1/6$, $2/7$, and $3/8$, and an OFF angle $\phi_0$ respectively equal to 0, 0.111, 0.222, and $0.333\,deg$. These variations have been chosen to emphasize the noise due to the OOK modulation, with increasing $p_0$ and $\phi_0$. In the following section, these four codes are referred to, respectively, as $C_k$, $C_5$, $C_{5b}$, and $C_{5c}$. Note that $C_k$, $C_{5b}$, and $C_{5c}$ are used for experiments only, but we do not use them in practice.

### B. Modifying the angular window

In the last section, we have established the upper bound of $\sigma^2_{\Phi_b}$, but we did not provide values of the angular window for which this bound is reached. To establish the mean and variance of $\Phi_r$ and $\Phi_f$, as well as the mean of $\Phi_b$, we have assumed the temporal stationarity. This is acceptable as long as the time required for the turret to complete one revolution is not an exact multiple of $T_0$, which is a choice by design. But the variance of $\Phi_b$ over the plane is not uniform. In fact, $\sigma^2_{\Phi_b}$ is invariant to the angle, but it depends on the distance between the beacon and the receiver. We therefore have to understand the relationship between this distance and $\sigma^2_{\Phi_b}$ by moving a beacon along the radius of a circle centered on the receiver. In practice, the value of the angular window $\phi_F - \phi_R$ depends on the received power and the threshold (see Figure 9). So, for a constant threshold, the angular window decreases if the power curve goes down (less received power), and increases if the power curve goes up (more received power). Then, for a given receiver and optical components, the angular window only depends on the received power. There are two practical ways to modify the received power (or angular window): (1) change the distance between the beacon and the receiver, or (2) change the power emitted by the beacon. But, in any case, the receiver only has access to the angular window via the demodulated signal and it is thus not capable of detecting whether the distance or the emitted power have been modified.

In a practical situation, the emitted power of the beacons is expected to be constant, while the distance can change. So we could measure the variance for all possible distances in the working range. However the experiment would be extremely tedious and time-consuming since the number of distances should be huge to appreciate the variations in the measurements. So, we choose to modify the emitted power, for a fixed working distance of $1\,m$. For each code, a hundred different emitted power values were taken in the $4\,mW$ to $150\,mW$ power range to obtain an approximately linear increase of the angular window. These power values and angular windows are values that correspond to distances ranging from $1\,m$ to $6\,m$. Finally, 1000 angle measurements are taken for each code and power value to compute the mean and variance of $\Phi_b$. As explained earlier, the receiver is not capable of detecting

| | | Theory | Simulations | Experiments |
|---|---|---|---|---|
| $\Phi_w$ bias ($deg$) | $C_k$ | 0 | 0 | 0 |
| | $C_5$ | $-1.85\,10^{-2}$ | $-1.78\,10^{-2}$ | $-4.85\,10^{-2}$ |
| | $C_{5b}$ | $-6.35\,10^{-2}$ | $-6.33\,10^{-2}$ | $-9.53\,10^{-2}$ |
| | $C_{5c}$ | $-1.25\,10^{-1}$ | $-1.22\,10^{-1}$ | $-1.56\,10^{-1}$ |
| $\Phi_b$ variance ($deg^2$) | $C_k$ | 0 | 0 | $3.93\,10^{-3}$ |
| | $C_5$ | $3.43\,10^{-4}$ | $3.36\,10^{-4}$ | $5.49\,10^{-3}$ |
| | $C_{5b}$ | $2.35\,10^{-3}$ | $2.35\,10^{-3}$ | $1.00\,10^{-2}$ |
| | $C_{5c}$ | $6.94\,10^{-3}$ | $6.80\,10^{-3}$ | $1.82\,10^{-2}$ |

**Table II:** Comparison of the theoretical, simulated, and experimental values for the biases of $\Phi_w$, and the maximum variances of $\Phi_b$, for the different codes.

whether the distance or the emitted power have been modified and, as a consequence, all the following graphics are plotted with respect to the angular window.

The angular window has to be estimated from the measurements. To be more precise and formal, the true angular window $\phi_W$ is defined as $\phi_F - \phi_R$ (see Figure 12). Therefore, we propose an estimator of the angular window given by

$$\Phi_w = \Phi_f - \Phi_r. \tag{23}$$

The mean of $\Phi_w$ is equal to

$$\mu_{\Phi_w} = E\{\Phi_f\} - E\{\Phi_r\} = \phi_W - p_0\phi_0. \tag{24}$$

The mean of $\Phi_w$ has a bias given by $-p_0\phi_0$. But, as the bias of $C_k$ is null since $p_0 = 0$, it is possible to derive $\phi_W$ by taking $\mu_{\Phi_w}$ for the constant code $C_k$.

### C. Simulator

In order to evaluate our theory about the code statistics, we developed a simulator. The goal is to validate the upper bound on the variance of $\Phi_b$, as well as the bias of the measured angular window $\Phi_w$. The four parameters considered by the simulator are the angular window, the code (symbols and durations), the turret period, and the number of turns. The codes, the turret period, and the number of turns are known precisely in our experiments. So, in order to compare the simulated results with the measurements, the angular windows are chosen in the same range as the real values. The simulated angular windows and variances are presented in Figure 14. Simulations confirm our theoretical results as bounds on variances (maximum of the curves) correspond to predicted bounds computed with equation (22). Also, the angular windows have a bias corresponding to values predicted by equation (24) (the numerical values are given in Table II). The simulator confirms our theoretical results about the variance added by the codes. However, the simulator does not take into account the natural noise of the system. So we have to use the real system to measure this natural noise. The results are presented in the next section.

### D. Experiments

The angular windows for each code are shown in the top plot of Figure 15 and the values of the biases are given in Table II. As expected, the curves are linear with the angular window. But the biases observed for the angular windows are

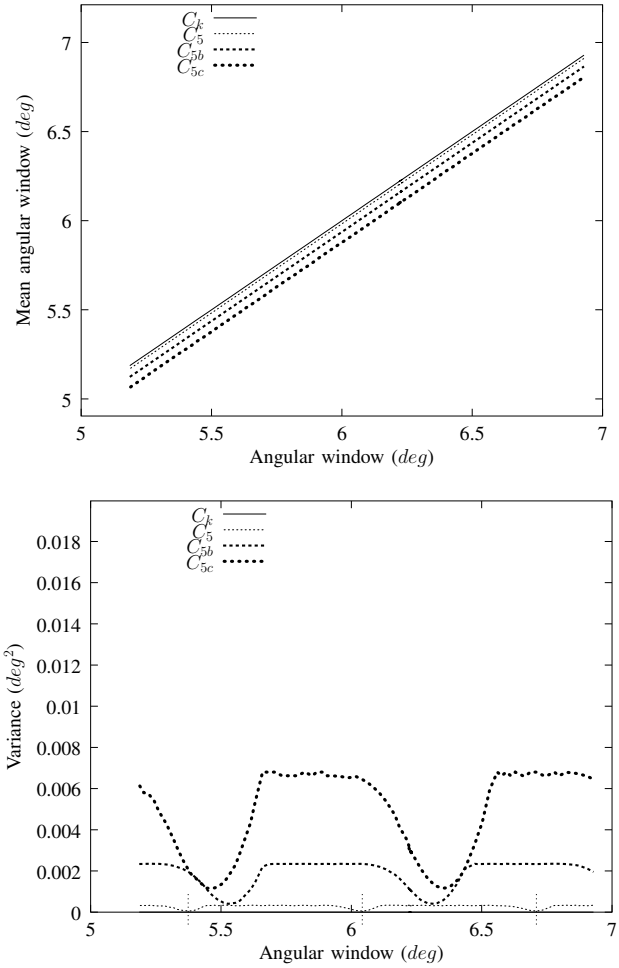



**Figure 14:** Results of simulations: values for the mean of the angular window $\Phi_w$ (top), and for the variance of the beacon angular position $\Phi_b$ (bottom).

larger (in absolute value) than the theoretical biases. On the other hand, they increase with $p_0$ and $\phi_0$, and the increments between the experimental biases are consistent with the theory.

The variances of the measurements for $\Phi_b$ are shown in the bottom plot of Figure 15. One can observe that the variance increases with $p_0$ and $\phi_0$, respectively for $C_k$ (the lowest), $C_5$, $C_{5b}$, and $C_{5c}$ (the highest), for all angular windows. One also sees large variations, especially for $C_{5b}$ and $C_{5c}$. This indicates that a dependency between $\Phi_r$ and $\Phi_f$ in function of the angular window exists. Examining the variances of $\Phi_r$ and $\Phi_f$ separately helps in this analysis (see Figure 16). Whereas $\Phi_r$ and $\Phi_f$ variances are quasi linear with respect to the angular window, $\Phi_b$ is not linear despite the fact that the estimator $\Phi_b$ is a linear function of the estimators $\Phi_r$ and $\Phi_f$. This confirms that a statistical relationship exists between $\Phi_r$ and $\Phi_f$.

Of course, there is a difference with the simulations since the measurements include the natural noise of the system and, as a result, the variances of the measurements are higher than the simulations. If the artificial noise due to the codes was independent of the natural noise, the measured variances could be obtained by adding the natural noise (measured with $C_k$) to the simulated variances. However, this is not the case

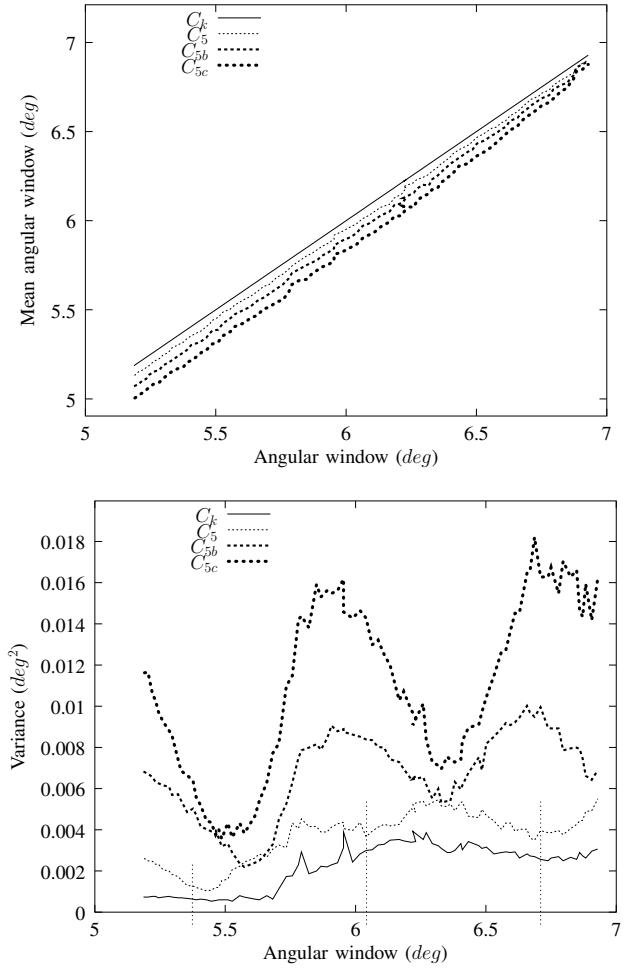



**Figure 15:** Results of measurements: values for the mean of the angular window $\Phi_w$ (top), and for the variance of the beacon angular position $\Phi_b$ (bottom).

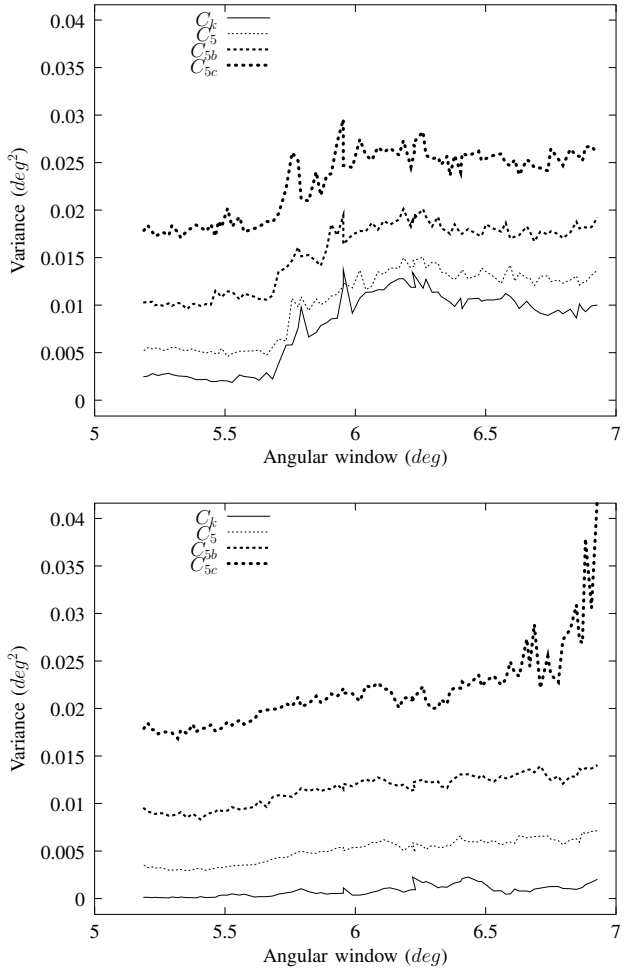



**Figure 16:** Results of measurements: variance of the beacon angular rising $\Phi_r$ (top) and falling $\Phi_f$ (bottom) edges.

since the variances obtained with this hypothesis (not shown here) still remain lower (but close) than the real variances. This result indicates that the natural and artificial noises are not independent, and that the real noise is higher than their sum. But, despite this discrepancy, the general shape of the simulated and experimental curves is similar. In particular, the large variations in the curves, the locations of the extrema, as well as their relative distances match our experiments perfectly (compare bottom plots of Figure 14 and Figure 15). Finally, note that we are interested in finding the variance of the measured angles in BeAMS, in the whole working range. The maximum of the curve measured for $C_5$ yields a variance equal to $5.49\,10^{-3}\,deg^2$, or equivalently a standard deviation equal to $7.41\,10^{-2}\,deg$.

### *E. Discussions of the experiments*

Our simulator provides values for the variances of $\Phi_b$ and biases of the angular window that match that of our theoretical model. However, they are some discrepancies between the experimental results and the theoretical bound. Amongst these discrepancies, the hypothesis that the natural variance is independent of the variance added by a code, as implemented in the simulator, is most subject to questioning. The reason for this is as follows. A detailed analysis of the receiver hardware shows the presence of an "Automatic Gain Control" (AGC) loop between the input and the demodulator. Typically, the gain is set to a high value when no signal is present for a "long time", resulting in a very noisy first transition ($\Phi_r$ in our case). This gain then decreases over time, resulting in sharper transitions (especially the last one, $\Phi_f$ in our case). This characteristic is clearly identifiable from the variances of $\Phi_r$ and $\Phi_f$ for a non modulated signal $C_k$ (see Figure 16). It appears that the gain value depends on the past values of the received signal and the duration of the OFF periods, and this produces a non constant natural variance over time. So, we have to consider this effect in tightening the agreement between theoretical and practical results. But it is no small task to consider this effect because it relates to the hardware.

Also, we have to consider another effect of the receiver hardware. In Section VI-A, we supposed that the received signal $R_i$ could be modeled has the logical AND between the emitted signal $E_i$ and $R_0$. However, it is not sure that a short leading or tailing burst (shorter than a bit) could trigger the receiver (see for example $R_2$, and $R_3$ in Figure 12). This has the effect of virtually increasing the OFF period

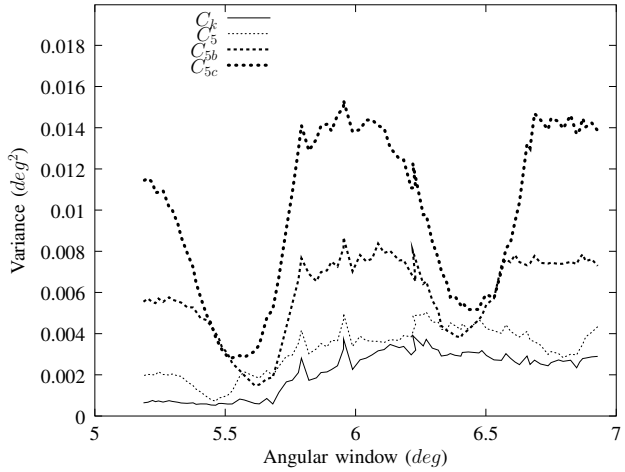



**Figure 17:** Results of simulations: values for the variance of the beacon angular position $\Phi_b$ by taking into account the natural noise and the "AND hypothesis".

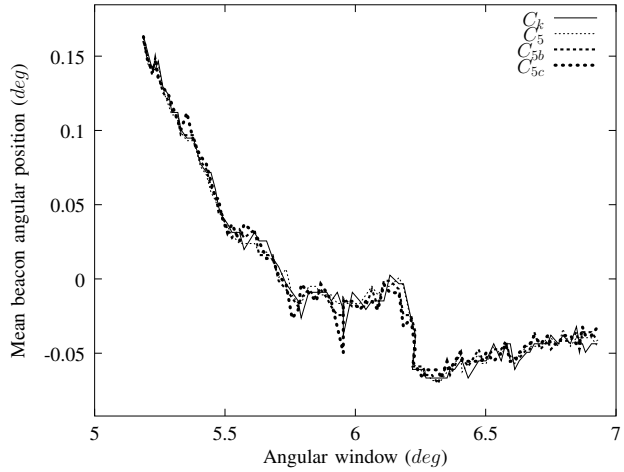



**Figure 18:** Variation in the mean beacon angular positions versus the angular window.

by a quantity equal to the minimum burst duration required ($56\,\%$ of $T_0$ in our case). This effect ("AND hypothesis") has been implemented in our simulator. Figure 17 presents the simulations obtained by taking into account the natural noise (from the measures of $C_k$), as well as the "AND hypothesis". With these modifications, the simulations are remarkably closer to the experiments (compare Figure 17 with bottom of Figure 15). From a practical point of view, it means that we have to modify the actual values of $T_0$ ($\phi_0$) and $p_0$ in order to use equation (22) adequately.

### *F. Measuring the accuracy*

In the previous sections, we have analyzed the variance of the measures of BeAMS. This variance represents the precision of the system. But we also want to evaluate the accuracy, that is the difference between the mean of the measures, and the actual (true) value of the beacon angular position. In contrast with the ease of measuring the precision of the system, the accuracy is very difficult to measure in practice (because it would require a very precise optical setup). However, there is one way to determine the accuracy for BeAMS. In order to do that, we note that the position computed by any triangulation algorithm depends only on the difference between pairs of angles [15], [34], whereas the orientation depends directly on the angles. So, a constant bias in the angles does not affect the position, but only the orientation, which has to be calibrated with the robot heading, anyway. It means that a problem arises if the biases are not the same for all angles. To verify this in BeAMS, we can plot the mean of the angle measurements versus the angular window (see Figure 18). This figure clearly shows some fluctuations of the mean angle in function of the angular window (or the received power); this indicates that there is a non constant bias in the measures, in the whole working range. This bias also has its origin by the presence of the AGC in the receiver. Biases in the measurements are due to delayed response times at the receiver. These delays are subject to the natural noise, and we have shown earlier that the natural noise depends on the code. It means that the *PDFs* of the noises at the beginning and at the end of the angular window are different, and so are their expectations, explaining that the bias changes with the angular window (or the received power). We can observe that the fluctuations of the bias occur within a range of $0.23\,deg$ (see Figure 18), and that this variation does not depend on the code. This value is our measure for the accuracy.

To be comprehensive, we present a comparison of different angle measurement systems in Table I. Only the rotating systems like BeAMS are presented. As explained earlier, it is difficult to compare systems because the performance criterion and test condition, as well as the available information are very different, or sometimes missing. Some authors mention absolute maximum error values or RMS values, while others provide standard deviation values. Most of the time, authors ignore the notion of the accuracy, and sometimes the angular resolution is expressed in the terms of the precision or of the accuracy. Also, some authors express the performance of the positioning algorithm in meters, and nothing is said about angles (*e.g.* for the system of Brkic *et al.*, we guess an angular precision of $2\,deg$). For BeAMS, we have provided values either for the variance and for the accuracy. But, if we have to provide a single error value, we can combine both measures using this result: $\sqrt{var + bias^2} = 0.24\,deg$. Table I shows that BeAMS has a better performance than other prototypes and is close to state of the art commercial systems (except for the DBIR LaserNav which is no longer manufactured). It should be noted that the performance of BeAMS are mainly downgraded by the accuracy, despite the low value of the variance. But, the performance is still good and it is sufficient in our application. However, for a large deployment of the system, we would recommend the design and use of a new fitted receiver, in which we can fully control the AGC, in order to cancel the bias and reach the highest performance (limited by the variance only).

Until now, we have assumed that the sensor (robot) does not move during measurement, and of course, this is an unrealistic situation. In a practical situation, there are additional errors due to robot motion, vibrations, shocks, uneven floors, etc.

Unlike the last ones (which cannot be predicted), the robot motion is controlled by the robot and can be taken into account by the positioning algorithm. The robot motion can be decomposed into a translation and a rotation, and it appears that the rotation is responsible for the most part of the error [26]. It should be noted that all rotating systems are subjects to this error since the robot motion has the effect of modifying the real turning speed of the sensor, and that the measurement principle relies on a constant speed. Taking into account the robot motion should be done, especially rotations because it can lead to high improvements in the computed position [26]. However, applying the corrections to the angles due to the robot motion is the task of the positioning algorithm. Nevertheless, the characteristics when the robot does not move (precision, accuracy) are still useful for data fusion algorithms, and should be the only ones to compare different systems.

## VIII. CONCLUSIONS

In this paper, we have presented BeAMS, a new beacon based angle measurement sensor used for mobile robot positioning, with an acquisition rate of $10\,Hz$. The entire sensor is contained in a $(8 \times 8 \times 6)\,cm^3$ volume. BeAMS innovates on many points. A simple infrared receiver/demodulator is the main sensor for the angle measurements; the beacons are common infrared LEDs. Furthermore, the system only requires one infrared communication channel; there is no synchronization channel between the beacons and the robot, and our system does not need an optical encoder for the motor control or angle measurement. The beacons are unequivocally identified so that the robot can compute its position without having to maintain an estimation of its position. Finally, the mechanical part of the system is kept as simple as possible (motor only, no gear system or belt) due to the hollow shaft.

In this paper, we have provided a theoretical framework to analyze the errors on the measured angles via the coding scheme used for the beacons, resulting from the use of an On-Off Keying modulation mechanism. A statistical estimator for the angular localization of a beacon has been proposed, and it was demonstrated that this estimator is unbiased and that its variance is upper bounded by $p_0 \frac{\phi_0^2}{6}$. This variance represents the power noise due to the OOK modulation. The variance increases with the square of the OFF angle $\phi_0$ (the angle corresponding to the OFF duration in a code) and with the proportion of zero symbols in a code $p_0$. This study has also justified some practical choices made in BeAMS, in particular: (1) the building of a symmetric optical part, (2) the reduction of $p_0$ versus $p_1$, and (3) the reduction of $T_0$ (or $\phi_0$).

In the last part of the paper, we have provided simulated and experimental results for the variance of the beacon position due to the OOK modulation. It appears that the results of the simulator are coherent with our theoretical results, but not entirely with the experimental results. Experimental results enlighten that the natural variance of the system depends on the code used because of the Automatic Gain Control loop of the receiver, which is responsible for a small mismatch between the experimental results and the theoretical bounds. In a practical situation, we would want to limit the variance added by the codes compared to the natural variance of the system. The theoretical bound computed in this paper as well as the simulator may help in this purpose. Note that our system achieves a low variance level on angles. The experimental values encountered in our system for the standard deviation of $\Phi_b$ range from $0.023$ to $0.063\,deg$ without codes and from $0.032$ to $0.074\,deg$ with codes, meaning that the noise added by our codes is small compared to the natural noise. The accuracy has also been evaluated to $0.23\,deg$. If we combine the variance and the bias, the final error measure is evaluated to $0.24\,deg$. It appears that the bias, caused by the AGC, is responsible for the biggest part of the error. We think that the use of a fitted receiver, in which we have full control of the AGC, would result in a far more precise system. All these features make BeAMS a small, low power, flexible, and tractable solution for robot positioning. BeAMS has now been used successfully during the EUROBOT contest for four years.

In this paper, we present a new angle measurement system designed in particular for applications with mobile robots. However, some ideas are general enough to be applicable to other sensors and systems. For example, the simplified mechanical part (hollow shaft and control without optical encoder) is usable for any rotating system. Similarly, the angular window principle can be applied to laser systems, and is not restricted to infrared systems. In addition, the theoretical framework can model any system that estimates the mean of two events of windowed OOK modulated signals. Finally, we provide an evaluation procedure valid for any angle measurement system.

## REFERENCES

[1] Y. Arai and M. Sekiai, "Absolute position measurement system for mobile robot based on incident angle detection of infrared light," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 1, Las Vegas, NV, USA, October 2003, pp. 986–991.

[2] A. Argyros, K. Bekris, and S. Orphanoudakis, "Robot homing based on corner tracking in a sequence of panoramic images," in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, Hawaii, USA, December 2001, pp. 3–10.

[3] G. Bauzil, M. Briot, and P. Ribes, "A navigation subsystem using ultrasonic sensors for the mobile robot HILARE," in *First International Conference on Robot Vision and Sensory Controls*, Stratford-upon-Avon, U.K, April 1981, pp. 47–58 & 681–698.

[4] Y. Beliveau, J. Fithian, and M. Deisenroth, "Autonomous vehicle navigation with real-time 3D laser based positioning for construction," *Automation in Construction*, vol. 5, no. 4, pp. 261–272, October 1996.

[5] M. Betke and L. Gurvits, "Mobile robot localization using landmarks," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 2, pp. 251–263, April 1997.

[6] J. Borenstein, H. Everett, and L. Feng, "Where am I? Systems and methods for mobile robot positioning," University of Michigan, Tech. Rep., March 1996.

[7] J. Borenstein, H. Everett, L. Feng, and D. Wehe, "Mobile robot positioning - sensors and techniques," *Journal of Robotic Systems*, vol. 14, no. 4, pp. 231–249, April 1997.

[8] K. Briechle and U. Hanebeck, "Localization of a mobile robot using relative bearing measurements," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 1, pp. 36–44, February 2004.

[9] M. Brkić, M. Lukić, J. Bajić, B. Dakić, and M. Vukadinović, "Hardware realization of autonomous robot localization system," in *International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, Opatija, Croatia, May 2012, pp. 146–150.

[10] C. Cohen and F. Koss, "A comprehensive study of three object triangulation," in *Mobile Robots VII*, vol. 1831. Boston, MA, USA: Society of Photo-Instrumentation Engineers, November 1992, pp. 95–106.

[11] E. Demaine, A. López-Ortiz, and J. Munro, "Robot localization without depth perception," in *Proceedings of Scandinavian Workshop on Algorithm Theory (SWAT)*, ser. Lecture Notes in Computer Science, vol. 2368. Springer, July 2002, pp. 177–194.

[12] G. Demetriou, "A survey of sensors for localization of unmanned ground vehicles (UGVs)," in *International Conference on Artificial Intelligence (ICAI)*, vol. 2. Las Vegas, NV, USA: CSREA Press, June 2006, pp. 659–668.

[13] V. Durst, D. Hagel, J. Vander, M. Blaich, and O. Bittel, "Designing an omni-directional infrared sensor and beacon system for the eurobot competition," in *Research and Education in Robotics - EUROBOT 2011*, vol. 161. Prague, Czech Republic: Springer, June 2011, pp. 102–113.

[14] A. Easton and S. Cameron, "A gaussian error model for triangulation-based pose estimation using noisy landmarks," in *IEEE Conference on Robotics, Automation and Mechatronics*, Bangkok, Thailand, June 2006, pp. 1–6.

[15] J. Esteves, A. Carvalho, and C. Couto, "Position and orientation errors in mobile robot absolute self-localization using an improved version of the generalized geometric triangulation algorithm," in *IEEE International Conference on Industrial Technology (ICIT)*, Mumbai, India, December 2006, pp. 830–835.

[16] J. Font-Llagunes and J. Batlle, "Localization of a mobile robot with omnidirectional wheels using angular kalman filtering and triangulation," in *ASME Conference on Engineering Systems Design and Analysis (ESDA)*, vol. 1, Torino, Italy, July 2006, pp. 713–720.

[17] ——, "Consistent triangulation for mobile robot localization using discontinuous angular measurements," *Robotics and Autonomous Systems*, vol. 57, no. 9, pp. 931–942, September 2009.

[18] Y. Gu, A. Lo, and I. Niemegeers, "A survey of indoor positioning systems for wireless personal networks," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 13–32, First Quarter 2009.

[19] Á. Gutiérrez, A. Campo, M. Dorigo, J. Donate, F. Monasterio-Huelin, and L. Magdalena, "Open e-puck range & bearing miniaturized board for local communication in swarm robotics," in *IEEE International Conference on Robotics and Automation (ICRA)*, Kobe, Japan, May 2009, pp. 3111–3116.

[20] S. Hernández, J. Torres, C. Morales, and L. Acosta, "A new low cost system for autonomous robot heading and position localization in a closed area," *Autonomous Robots*, vol. 15, no. 2, pp. 99–110, September 2003.

[21] H. Hu and D. Gu, "Landmark-based navigation of industrial mobile robots," *International Journal of Industry Robot*, vol. 27, no. 6, pp. 458–467, 2000.

[22] G. Jang, S. Kim, J. Kim, and I. Kweon, "Metric localization using a single artificial landmark for indoor mobile robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, August 2005, pp. 3407–3412.

[23] I. Kelly and A. Martinoli, "A scalable, on-board localisation and communication system for indoor multi-robot experiments," *Sensor Review*, vol. 24, no. 2, pp. 167–180, January 2004.

[24] A. Kemppainen, J. Haverinen, and J. Röning, "An infrared location system for relative pose estimation of robots," in *Symposium on Robot Design, Dynamics, and Control*, Warsaw, Poland, June 2006, pp. 379–386.

[25] C. Lee, Y. Chang, G. Park, J. Ryu, S.-G. Jeong, S. Park, J. Park, H. Lee, K.-S. Hong, and M. Lee, "Indoor positioning system based on incident angles of infrared emitters," in *Conference of the IEEE Industrial Electronics Society (IECON)*, vol. 3, Busan, South Korea, November 2004, pp. 2218–2222.

[26] I. Loevsky and I. Shimshoni, "Reliable and efficient landmark-based localization for mobile robots," *Robotics and Autonomous Systems*, vol. 58, no. 5, pp. 520–528, May 2010.

[27] C. McGillem and T. Rappaport, "A beacon navigation method for autonomous vehicles," *IEEE Transactions on Vehicular Technology*, vol. 38, no. 3, pp. 132–139, August 1989.

[28] K. Muthukrishnan, M. Lijding, and P. Havinga, "Towards smart surroundings: Enabling techniques and technologies for localization," in *International Workshop on Location- and Context-Awareness*, ser. Lecture Notes in Computer Science, T. Strang and C. Linnhoff-Popien, Eds., vol. 3479, Oberpfaffenhofen, Germany, May 2005, pp. 350–362.

[29] T. Nishizawa, A. Ohya, and S. Yuta, "An implementation of on-board position estimation for a mobile robot," in *IEEE International Conference on Robotics and Automation (ICRA)*, vol. 1, Nagoya, Japan, May 1995, pp. 395–400.

[30] A. Papoulis, *Probability, random variables, and stochastic processes*. McGraw-Hill, February 1991.

[31] V. Pierlot and M. Van Droogenbroeck, "A simple and low cost angle measurement system for mobile robot positioning," in *Workshop on Circuits, Systems and Signal Processing (ProRISC)*, Veldhoven, The Netherlands, November 2009, pp. 251–254.

[32] ——, "Analysis of a robot positioning system based on a rotating receiver, beacons, and coded signals," in *Proceedings of the European Signal Processing Conference (EUSIPCO)*, Barcelona, Spain, August 2011, pp. 1766–1770.

[33] ——, "Statistical analysis of modulated codes for robot positioning – application to BeAMS," University of Liège, Belgium, Tech. Rep., May 2013. [Online]. Available: http://hdl.handle.net/2268/144734

[34] V. Pierlot, M. Van Droogenbroeck, and M. Urbin-Choffray, "A new three object triangulation algorithm based on the power center of three circles," in *Research and Education in Robotics (EUROBOT)*, ser. Communications in Computer and Information Science, vol. 161. Springer, 2011, pp. 248–262.

[35] K. Premi and C. Besant, "A review of various vehicle guidance techiques that can be used by mobile robots or AGVs," in *2nd International Conference on Automated Guided Vehicle Systems*, Stuttgart, Germany, June 1983.

[36] J. Pugh, X. Raemy, C. Favre, R. Falconi, and A. Martinoli, "A fast on-board relative positioning module for multirobot systems," *IEEE/ASME Transactions on Mechatronics*, vol. 14, no. 2, pp. 151–162, April 2009.

[37] J. Roberts, T. Stirling, J.-C. Zufferey, and D. Floreano, "2.5D infrared range and bearing system for collective robotics," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, St. Louis, MO, USA, October 2009, pp. 3659–3664.

[38] I. Shimshoni, "On mobile robot localization from landmark bearings," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 6, pp. 971–976, December 2002.

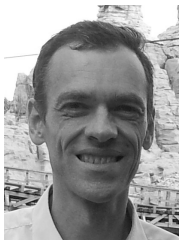
[39] D. Sinriech and S. Shoval, "Landmark configuration for absolute positioning of autonomous vehicles," *IIE Transactions*, vol. 32, no. 7, pp. 613–624, July 2000.

[40] H. Sobreira, A. Moreira, and J. Esteves, "Low cost self-localization system with two beacons," in *International Conference on Mobile Robots and Competitions (ROBOTICA)*, Leiria, Portugal, March 2010, pp. 73–77.

[41] O. Tekdas and V. Isler, "Sensor placement for triangulation-based localization," *IEEE Transactions on Automation Science and Engineering*, vol. 7, no. 3, pp. 681–685, July 2010.

[42] E. Zalama, S. Dominguez, J. Gómez, and J. Perán, "Microcontroller based system for 2D localisation," *Mechatronics*, vol. 15, no. 9, pp. 1109–1126, November 2005.

**Vincent Pierlot** was born in Liège, Belgium. He received the Electrical Engineering Degree, and the Ph.D. degree from the University of Liège in, respectively, 2006 and 2013. He is the architect of a new positioning system used during the EUROBOT contest for several years. His research interests are mainly focused on electronics, embedded systems, motion analysis, positioning, triangulation, and robotics.



**Marc Van Droogenbroeck** received the degree in Electrical Engineering and the Ph.D. degree from the University of Louvain (UCL, Belgium) in 1990 and 1994 respectively. During his Ph.D. he spent two years at the Center of Mathematical Morphology (CMM) of the School of Mines of Paris. In April 1994, he joined the New Development Department of Belgacom. He was appointed as the Head of the Belgian Delegation in the ISO/MPEG Committee and as a representative to the World Wide Web Consortium for two years. In 2003, he was a visiting Scientist at the CSIRO, Australia. Since 1998, M. Van Droogenbroeck has been a member of the Faculty of Applied Sciences at the University of Liège, where he is currently a professor. His current interests are computer vision, 3D vision, real time processing, motion analysis, gait analysis, telecommunications, positioning, and robotics.