Position and Orientation Errors in Mobile Robot Absolute Self-Localization Using an Improved Version of the Generalized Geometric Triangulation Algorithm

João Sena Esteves
Department of Industrial Electronics
University of Minho
4800-058 Guimarães, PORTUGAL
sena@dei.uminho.pt

Adriano Carvalho
Dept. of Electrical Engineering and Computers
Faculty of Engineering, University of Porto
4200-465 Porto, PORTUGAL

Carlos Couto
Department of Industrial Electronics
University of Minho
4800-058 Guimarães, PORTUGAL

Abstract **– Triangulation with active beacons is widely used in the absolute localization of mobile robots. The original Generalized Geometric Triangulation algorithm suffers only from the restrictions that are common to all algorithms that perform self-localization through triangulation. But it is unable to compute position and orientation when the robot is over the segment of the line that goes by beacons 1 and 2 whose origin is beacon 1 and does not contain beacon 2. An improved version of the algorithm allows self-localization even when the robot is over that line segment. Simulations results suggest that a robot is able to localize itself, with small position and orientation errors, over a wide region of the plane, if measurement uncertainty is small enough.**

I. INTRODUCTION

Localization is the process of finding both position and orientation of a vehicle in a given referential system [1], [2], [3], [4], [5], [6], [7], [8], [9], [10] Triangulation with active beacons is a robust, accurate, flexible and widely used method of absolute localization [2], [11].

Self-localization through triangulation is based on the measurement of the bearings of the robot relatively to beacons placed in known positions. When navigating on a plane, three distinguishable beacons are required - and usually enough - for the robot to localize itself. In Fig.1, λ_{12} is the oriented angle "seen" by the robot between beacons 1 and 2. It defines an arc between these beacons, which is a set of possible positions of the robot [12]. An additional arc between beacons 1 and 3 is defined by λ_{31}. The robot is in the intersection of the two arcs. Several algorithms of self-localization through triangulation are described in [1], [2], [3], [6], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22] and [23].

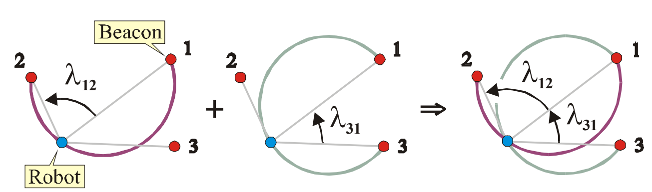


Fig. 1. Self-localization through triangulation.

Two restrictions are common to all algorithms that perform self-localization through triangulation [2], [3]:

1. The robot must "see" at least three distinguishable beacons to localize itself in a plane. All areas of the plane with less than three visible beacons are unsuitable for robot localization;
2. Localization is not possible if the robot is over the circumference defined by three non-collinear beacons (the intersection of the arcs shown in Fig.1 is another arc, not a point) or over the line defined by three collinear beacons.

The Geometric Triangulation algorithm described in [18] uses three distinguishable beacons that must be ordered in a particular way. According to the authors of that paper, *"the algorithm works consistently only when the robot is within the triangle formed by the three landmarks*[1]*."*

The Generalized Geometric Triangulation algorithm [3] does not require beacon ordering and suffers only from the two restrictions that are common to all algorithms that perform self-localization through triangulation. But it is unable to compute position and orientation when the robot is over the segment of the line that goes by beacons 1 and 2 whose origin is beacon 1 and does not contain beacon 2. It was assumed that, when a beacon becomes between the robot and another beacon, the closest beacon hides the farther one or else the goniometer is not able of simultaneously detecting more than one beacon. Any of those situations prevents self-localization. However, the impediment is due to the technology used, not triangulation itself.

Section II describes briefly an improved version of Generalized Geometric Triangulation algorithm [1], [2]. It works over the segment of the line that goes by beacons 1 and 2 whose origin is beacon 1 and does not contain beacon 2. Position and orientation errors are defined on Section III. On Section IV, simulation results obtained with different

[1] Beacons are also called *landmarks* by some authors.

1-4244-0726-5/06/$20.00 '2006 IEEE

measurement uncertainties show the distribution of errors through the navigation plane. Conclusions are presented in Section V.

II. The Improved Generalized Geometric Triangulation Algorithm

The Generalized Geometric Triangulation algorithm uses (Fig.2) three distinguishable beacons, randomly labeled 1, 2 and 3, with known positions (x_1, y_1), (x_2, y_2) and (x_3, y_3).

L_{12} is the distance between beacons 1 and 2. L_{31} is the distance between beacons 1 and 3. L_1 is the distance between the robot and beacon 1. In order to determine its position (x_R, y_R) and orientation θ_R, the robot measures – in counterclockwise fashion – the angles λ_1, λ_2 and λ_3, which are the beacon orientations relative to the robot heading.

Line 14 of the algorithm was not present in the original version of the algorithm. It is required only when the robot is over the segment of the line that goes by beacons 1 and 2 whose origin is beacon 1 and does not contain beacon 2.

III. Position and Orientation Errors

In general, the *computed position* does not coincide with the *true position*[2] of the robot and the *computed orientation* also does not coincide with its *true orientation*. There are a *position error* and an *orientation error*, which – in this paper – have the following definitions:

- the *position error* ΔP_R (Fig.3) is the distance between the *computed position* P_{Rc} and the *true position* P_R;
- the *orientation error* $\Delta\theta_R$ is the modulus of the difference between the *computed orientation* θ_{Rc} and the *true orientation.*

Measurement errors constitute the main source of position and orientation errors, which magnitude also depends on the position of the robot relatively to the beacons.

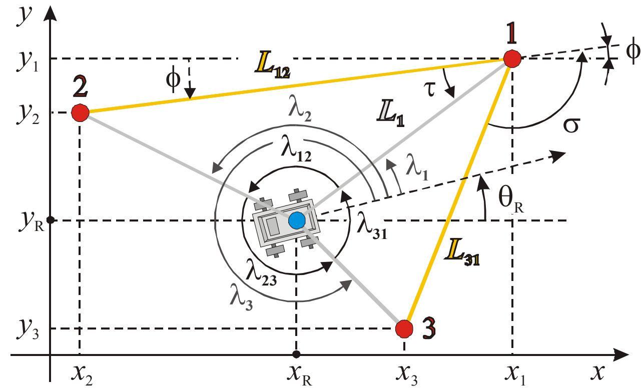
Self-localization through triangulation is not possible when the robot is over the circumference defined by three non-collinear beacons or the line defined by three collinear beacons.

In the Generalized Geometric Triangulation algorithm, this restriction appears as an impossibility to compute τ due to a 0/0 indetermination in the expression

$$\tau = \tan^{-1}\left[\frac{\sin\lambda_{12}\cdot\left(L_{12}\cdot\sin\lambda_{31}-L_{31}\cdot\sin\gamma\right)}{L_{31}\cdot\sin\lambda_{12}\cdot\cos\gamma-L_{12}\cdot\cos\lambda_{12}\cdot\sin\lambda_{31}}\right], \quad (1)$$

where

$$\gamma = \sigma - \lambda_{31}. \quad (2)$$



<u>Generalized Geometric Triangulation algorithm:</u>

1. If there are less than three visible beacons available, then return a warning message and stop.
2. $\lambda_{12} = \lambda_2 - \lambda_1$
3. If $\lambda_1 > \lambda_2$ then $\lambda_{12} = 360^\circ + (\lambda_2 - \lambda_1)$
4. $\lambda_{31} = \lambda_1 - \lambda_3$
5. If $\lambda_3 > \lambda_1$ then $\lambda_{31} = 360^\circ + (\lambda_1 - \lambda_3)$
6. Compute L_{12} from known positions of beacons 1 and 2.
7. Compute L_{31} from known positions of beacons 1 and 3.
8. Let ϕ be an oriented angle such that $-180^\circ < \phi \le 180^\circ$. Its origin side is the image of the positive x semi-axis that results from the translation associated with the vector which origin is (0, 0) and ends on beacon 1. The extremity side is the part of the straight line defined by beacons 1 and 2 which origin is beacon 1 and does not go by beacon 2.
9. Let σ be an oriented angle such that $-180^\circ < \sigma \le 180^\circ$. Its origin side is the straight line segment that joins beacons 1 and 3. The extremity side is the part of the straight line defined by beacons 1 and 2 which origin is beacon 1 and does not go by beacon 2.
10. $\gamma = \sigma - \lambda_{31}$
11. $\tau = \tan^{-1}\left[\frac{\sin\lambda_{12}\cdot(L_{12}\cdot\sin\lambda_{31}-L_{31}\cdot\sin\gamma)}{L_{31}\cdot\sin\lambda_{12}\cdot\cos\gamma-L_{12}\cdot\cos\lambda_{12}\cdot\sin\lambda_{31}}\right]$
12. If $\begin{cases}\lambda_{12} < 180^\circ \\ \tau < 0^\circ\end{cases}$ then $\tau = \tau + 180^\circ$
13. If $\begin{cases}\lambda_{12} > 180^\circ \\ \tau > 0^\circ\end{cases}$ then $\tau = \tau - 180^\circ$
14. If $\tau = 0^\circ \wedge \left[\begin{cases}\sigma > 0^\circ \\ \lambda_{31} > 180^\circ\end{cases} \vee \begin{cases}\sigma < 0^\circ \\ \lambda_{31} < 180^\circ\end{cases}\right]$ then $\tau = 180^\circ$
15. If $|\sin\lambda_{12}| > |\sin\lambda_{31}|$ then $L_1 = \frac{L_{12}\cdot\sin(\tau+\lambda_{12})}{\sin\lambda_{12}}$
16. else $L_1 = \frac{L_{31}\cdot\sin(\tau+\sigma-\lambda_{31})}{\sin\lambda_{31}}$
17. $x_R = x_1 - L_1\cdot\cos(\phi+\tau)$
18. $y_R = y_1 - L_1\cdot\sin(\phi+\tau)$
19. $\theta_R = \varphi + \tau - \lambda_1$
20. If $\theta_R \le -180^\circ$ then $\theta_R = \theta_R + 360^\circ$
21. If $\theta_R > 180^\circ$ then $\theta_R = \theta_R - 360^\circ$

Fig. 2. Generalized Geometric Triangulation.

[2] The position of a robot with non-negligible dimensions is the position of one of its points.

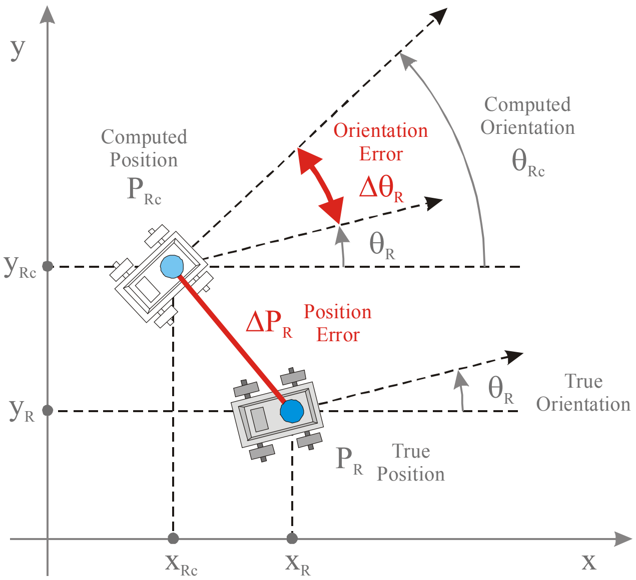


Fig. 3. Position error and orientation error.

The two arcs that form the circumference shown in Fig.4 correspond to supplementary λ_{12} angles [24]. Applying this to both λ_{12} and λ_{31} over the circumference defined by three non-collinear beacons ordered in counter-clockwise fashion results in the angles shown in Fig.5, inside gray boxes. Fig.5 also shows that

$$L_{12} \cdot \sin\delta = L_{31} \cdot \sin(\sigma - \delta) \quad . \tag{3}$$

The three possible sets of λ_{12} and λ_{31} occurring over the circumference, used together with (3), lead to the same result, which causes a 0/0 indetermination in (1):

$$\begin{cases} \sin\lambda_{12} \cdot \left[L_{12} \cdot \sin\lambda_{31} - L_{31} \cdot \sin(\sigma - \lambda_{31})\right] = 0 \\ L_{31} \cdot \sin\lambda_{12} \cdot \cos(\sigma - \lambda_{31}) - L_{12} \cdot \cos\lambda_{12} \cdot \sin\lambda_{31} = 0 \end{cases} \tag{4}$$

An analogous analysis made to non-collinear beacons ordered in clockwise fashion leads to the same conclusion. If the robot is over the line defined by three collinear beacons, each of the angles λ_{12} and λ_{31} has a value of 0º or 180º. So, $\sin\lambda_{12}$ and $\sin\lambda_{31}$ are both zero, which leads to a 0/0 indetermination in (1).

Even if computing errors are negligible, due to errors on angle measurements this indetermination may not happen. However, it is expectable that errors on the computed value of τ cause large position and orientation errors when the robot is over the circumference defined by three non-collinear beacons or the line defined by three collinear beacons.

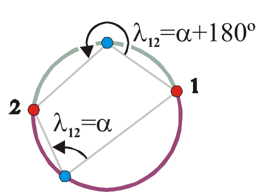


Fig. 4. λ_{12} over a circumference that goes by beacons 1 and 2.

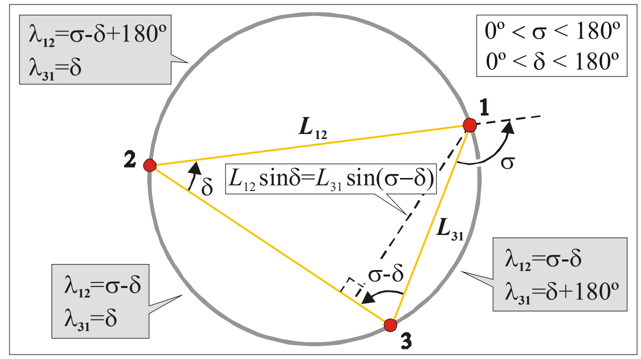


Fig. 5. λ_{12} and λ_{31} over the circumference defined by three non-collinear beacons ordered in counter-clockwise fashion.

IV. Simulations Results

In order to verify the distribution of position and orientation errors through the navigation plane, two sets of tests were performed in a simulation environment. The code was written in Java 2. It was used the *Java 2 SDK, Standard Edition* (version 1.3 for *Windows*), upgraded with *Java 3D* (version 1.2.1 Beta, for *Win32/DirectX*), on a personal computer equipped with a *Intel Pentium III* (995MHz) processor and running *Windows XP* (version 5.1.2600). Graphics were plotted with *Matlab* (version 5.2).

Simulations are performed in a square shaped area of the navigation plane. Results from the first set of tests (Fig.6) show position and orientation errors occurring close to the beacons, which are about half the length of the square side away from each other. Results from the second set of tests (Fig.7) show position and orientation errors occurring far from the beacons. All beacons are now close to the center of the square and the distance between them is about 1/100 of the square side length.

In each test, three beacons labeled 1, 2 and 3 are placed in known positions of a Cartesian plane. Beacon positions are printed close to the results of each test. A robot is placed at the origin of the referential system. Its orientation is arbitrarily set to a value between -180º and 180º. Then, a four-step sequence is performed:

I. Angles λ_1, λ_2 and λ_3 are computed from *a priori* known beacons and robot positions;
II. Angles λ_1, λ_2 and λ_3 are rounded to 2, 1 or 0 decimal places, simulating the outputs of a digital angle-measuring device with resolution ρ equal to 0,01º, 0,1º or 1º, respectively (measurement uncertainty $\pm\Delta\lambda$ equal to ±0,005º, ±0,05º or ±0,5, respectively).
III. *A priori* known beacons positions and the rounded values of λ_1, λ_2 and λ_3 are used as inputs of the Generalized Geometric Triangulation algorithm, which computes both position and orientation of the robot.
IV. Position and orientation errors are computed from *a priori* known robot position and orientation and their computed values.

The four steps are repeated for robot positions covering a 100 x 100 square. Position increments of 0.1 are made in both x and y directions. In each point, robot orientation is arbitrarily set to a value between -180º and 180º. Position and orientation errors obtained in each position are plotted in 2D and 3D graphics (Fig.6 and Fig.7). The z-axes of 3D position error graphics are labeled in the same length units used in x-axes and y-axes. The z-axes of 3D orientation error graphics are labeled in degrees. In order to emphasize the smallest errors occurring in the analyzed square, upper limits to the visualized errors are set in both 2D and 3D views. Only position and orientation errors in regions very close to the circumference defined by the three beacons are too large to be plotted.

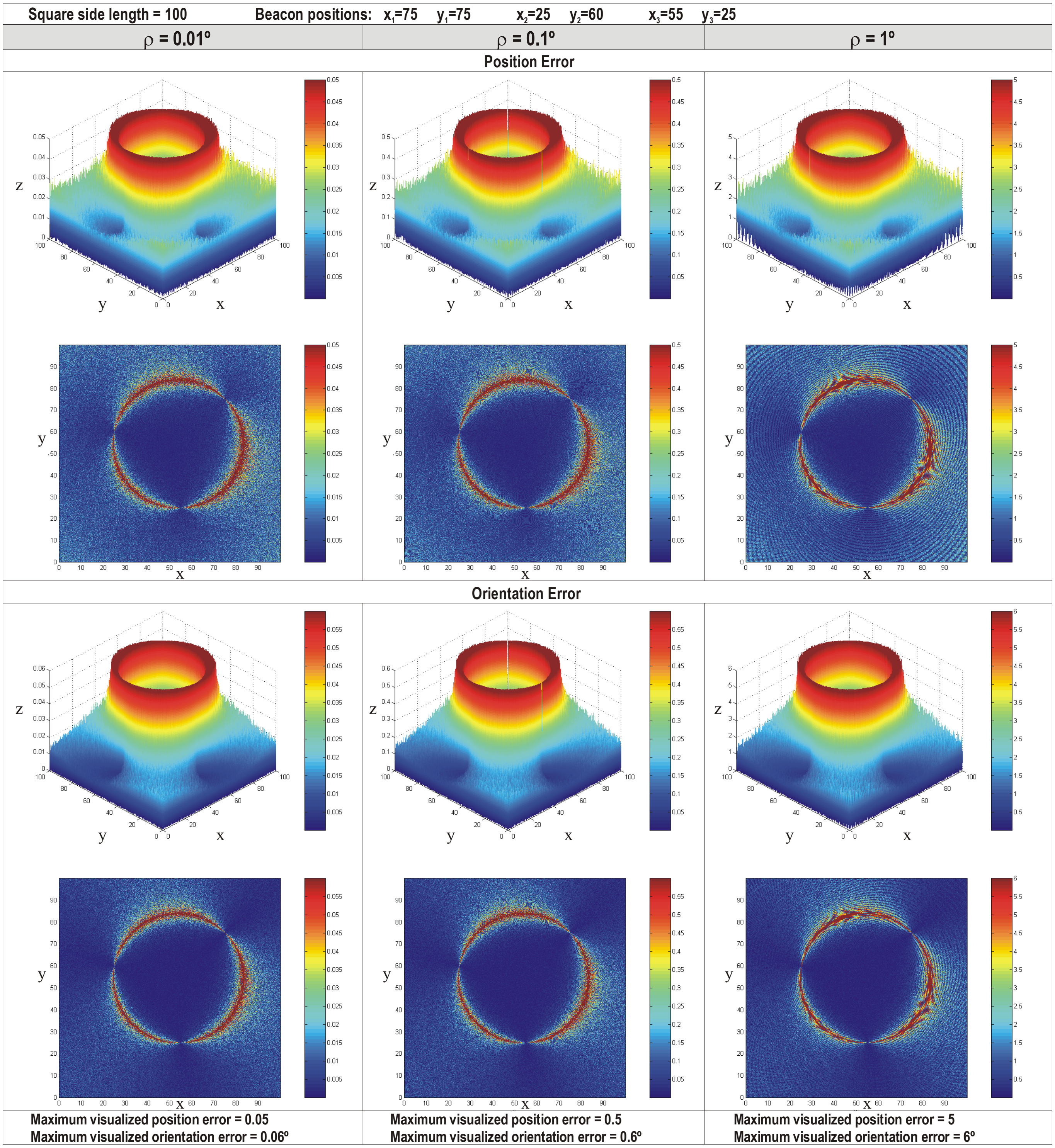


Fig. 6. Position and orientation errors close to the beacons.

Some properties of the obtained position and orientation errors are the following:

- They agree with the analysis made in Section III;
- They are small inside the triangle formed by three non-collinear beacons;
- They increase significantly as the robot approaches the circumference defined by three non-collinear beacons;
- They decay abruptly as the robot drives away from this circumference in a radial direction and remain small in its surroundings;
- They grow again, more slowly, as the robot drives away from the beacons;
- They increase about ten times each time $\Delta\lambda$ is multiplied by ten.

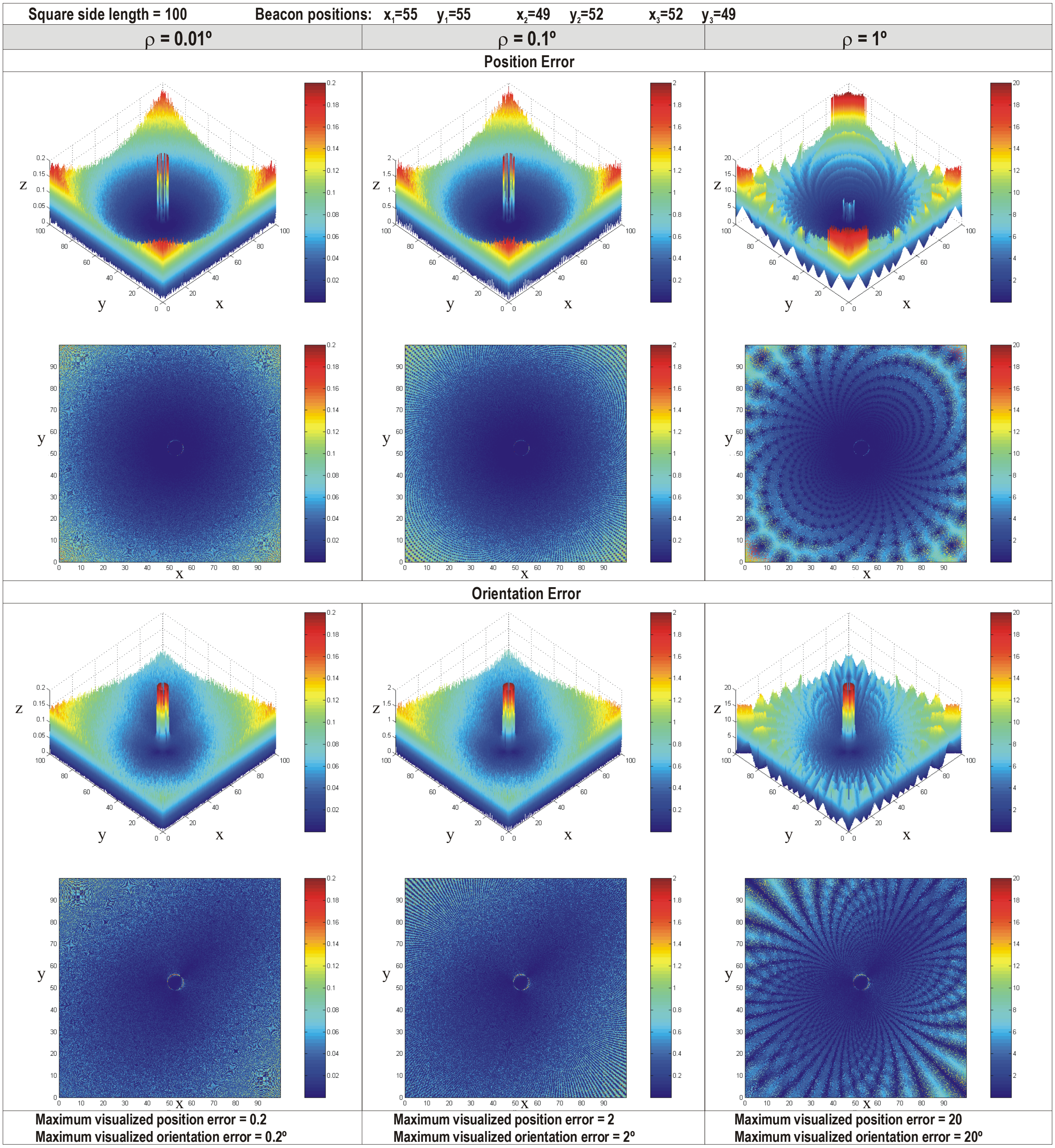


Fig. 7. Position and orientation errors far from the beacons.

An important issue is to make sure that computations are performed with enough significant digits to avoid deterioration of the results due to round-off errors. To accomplish this, simulations were performed using double precision (64 bits) in all four steps. Then, they were repeated but, this time, using single precision (32 bits) on step III. For $\Delta\lambda$ ranging from 0.01º to 1º it is not possible to distinguish the results obtained in the two sets of simulations. This shows that position and orientation errors are due only to the errors added to input angles in step II, not to round-off errors in computations. Graphics shown in this paper result from simulations entirely performed using double precision.

V. Conclusions

An improved version of Generalized Geometric Triangulation algorithm was used in two sets of tests, performed in a simulation environment in order to verify the distribution of position and orientation errors through the navigation plane.

Simulations results show that measurement errors and the position of the robot relatively to the beacons affect strongly the magnitude of position and orientation errors. These errors have the following properties:

- They agree with the analysis previously made;
- They are small inside the triangle formed by three non-collinear beacons;
- They increase significantly as the robot approaches the circumference defined by three non-collinear beacons;
- They decay abruptly as the robot drives away from this circumference in a radial direction and remain small in its surroundings;
- They grow again, more slowly, as the robot drives away from the beacons;
- They increase about ten times each time $\Delta\lambda$ is multiplied by ten.

Results suggest that, if $\Delta\lambda$ is small enough, the robot is able to localize itself, with small position and orientation uncertainties, over a wide region of the plane. They also suggest the need to provide the algorithm with a way of detecting points of the navigation plane that are unsuitable for robot localization due to inability of the algorithm to compute a solution or excessive position and orientation uncertainties.

References

[1] João Sena Esteves, Adriano Carvalho and Carlos Couto, *An improved version of the Generalized Geometric Triangulation algorithm;* II SELASI – II European-Latin-American Workshop on Engineering Systems, Faculdade de Engenharia da Universidade do Porto, Porto, Portugal, June 21-23, 2006.

[2] João Sena Esteves, Metodologia de Autolocalização Absoluta em Ambientes Quase-Estruturados (PhD Thesis), University of Minho, Portugal, July 2005.

[3] João Sena Esteves, Adriano Carvalho and Carlos Couto, *Generalized Geometric Triangulation algorithm for mobile robot absolute self-localization;* ISIE 2003 – 2003 IEEE International Symposium on Industrial Electronics, Rio de Janeiro, Brazil, June 9–12, 2003.

[4] P. J. McKerrow, *Introduction to Robotics,* Addison-Wesley Publishing Company, Inc., 1991.

[5] John J. Leonard and Hugh F. Durrant-White, *Directed Sonar Sensing for Mobile Robot Navigation,* Kluwer Academic Publishers, 1992.

[6] J. Borenstein, H. R. Everett and L. Feng, *Where am I? Sensors and Methods for Mobile Robot Positioning* (Technical Report), The University of Michigan, 1996.

[7] Cyril Cauchois, Eric Brassart, Bruno Marhic and Cyril Drocourt, *An absolute localization method using synthetic panoramic image base,* Proceedings of the OMNIVIS'02 - Third Workshop on Omnidirectional Vision, p. 128-135, June 2, 2002.

[8] Jens-Steffen Gutmann, *Markov-Kalman localization for mobile robots,* ICPR'02 – 16th International Conference on Pattern Recognition, Vol. 2, Quebec, Canada, August 11 – 15, 2002.

[9] T. Venet, T. Capitaine, M. Hamzaoui, and F. Fazzino, *One active beacon for an indoor absolute localization of a mobile vehicle,* Proceedings of the 2002 IEEE International Conference on Robotics and Automation, p. 1-6, Washington, DC, May 2002.

[10] K. Briechle, and U. D. Hanebeck, *Localization of a mobile robot using relative bearing measurements,* IEEE Transactions on Robotics and Automation, Vol. 20, Nº. 1, p. 36-44, February 2004.

[11] NDC Automation Inc., *Advantage Lazerway™! - Laser Guidance vs. Inertial Guidance* (Technology Brief), USA (980129), 1998.

[12] Paulo Ventura Araújo, *Curso de Geometria,* Gradiva, 1998.

[13] Heinrich Dorrie, *100 Great Problems of Elementary Mathematics: Their History and Solution* (trad. David Antin), Dover Publications, New York, 1965.

[14] Lélis Espartel, *Curso de Topografia (7ª ed.),* Editora Globo, 1980.

[15] Raymond E. Davis, Francis S. Foote, James M. Anderson and Edward M. Mikhail, *Surveying: Theory and Practice (6ª ed.),* McGraw-Hill, 1981.

[16] Francisco Domínguez García-Tejero, *Topografia Abreviada (5ª ed.),* Dossat, Madrid, 1981.

[17] Clare D. Mcgillem and Theodore S. Rappaport, *A beacon navigation method for autonomous vehicles,* IEEE Transactions on Vehicular Technology, Vol. 38, nº 3, August 1989.

[18] Charles Cohen and Frank V. Koss, *A comprehensive study of three object triangulation,* mobile robots, VII SPIE Vol. 1831, 1992.

[19] H. R.Everett, *Sensors for Mobile Robots,* A K Peters, June 1995.

[20] O. Fuentes *et al.*, *Mobile Robotics 1994* (Technical Report 588), The University of Rochester Computer Science Department, Rochester, New York 14627, June 1995.

[21] E. Stella *et al., An autonomous system for indoor structured environment,* Intelligent Autonomous Systems, U. Rembold *et al.* (Eds.), IOS Press, 1995.

[22] Margrit Betke and Leonid Gurvits, *Mobile robot localization using landmarks,* IEEE Transactions on Robotics and Automation, Vol. 13, Nº 2, April 1997.

[23] Junhong Ji, Giovanni Indiveri, Paul-Gerhard Ploeger and Ansgar Bredenfeld, *An omni-vision based self-localization method for soccer robot,* Proceedings of the IEEE IV2003, Intelligent Vehicles Symposium, Columbus, Ohio, USA, June 9 – 11, 2003.

[24] David Wells, *Dicionário de Geometria Curiosa,* Colecção O Prazer da Matemática, Gradiva, 1998.