

Quantitative Performance Metrics for Mobile Robots Navigation

Nelson David Muñoz Ceballos*, Jaime Alejandro Valencia**
and Nelson Londoño Ospina**

**Polytechnic Jaime Isaza Cadavid, **University of Antioquia
Medellin, Colombia*

1. Introduction

Mobile robots navigation involves the design of navigation algorithms. There are several navigation methods for mobile robots and every time a new navigation algorithm is proposed, some comparisons with traditional algorithms are usually made, but there isn't set of benchmarks globally accepted to assess the performance of algorithms. Despite the wide variety of studies and research on robot navigation systems, performance measures are often neglected in this research area, which makes it difficult to make an objective performance comparison (Wong et al., 2002); in general, use of quantitative metrics is limited to measuring the length of the path or the time needed by the robot to complete the task. Additionally, the lack of consensus on how to define or measure these systems impedes rigor and prevents evaluation of progress in this field and compare its different capabilities (Evans & Messina, 2000). As the degree of complexity in robotics is increasing, becomes necessary to establish proper approaches and benchmarking procedures, mainly for two reasons: first, reliable benchmarking is necessary in order to allow the comparison of the many robotics research results seeking this way to enable their industrial application. Second, to position the robotics as a serious science is important to consider the replication of experiments, i.e., it is important to verify whether a new procedure or algorithm proposed is really a breakthrough that can be used in new applications. This would be achieved only if it is determined which are the appropriate benchmarking procedures that allow to compare the actual practical results with reference to standard accepted procedures (Eurongemsig, 2008). However, by applying navigation comparison metrics of a mobile robot, such as the length of trajectory (path), collision risk and smoothness of trajectory, using a protocol, that is in a systematic and ordered way, works on mobile robots navigation algorithms can be systematized, and this will help researchers to decide which algorithm should be implemented in the vehicle.

This chapter describes aspects related with a procedure used for the performance evaluation of mobile robots navigation algorithms. First, several performance metrics used in the navigation of mobile robots are described, then, the protocol to be followed in the performance evaluation is defined, finally, the results are presented. To illustrate the procedure, we used simulation software.

2. Performance Metrics for Robot Navigation

Robot navigation is based in sensing of world (scenario), to compute the next motion, and actuation. (Minguez, 2008) cites a concise summary of several navigation methods:

The potential field methods addressed the first sensor-based motions (Khatib, 1986; Krogh & Thorpe, 1986), the vector field histogram was the first alternative to do obstacle avoidance with uncertain sensor like ultrasounds (Borenstein & Koren, 1991), Elastic bands was the first technique combining planning and reaction schemas in a unified framework (Quinlan & Khatib, 1993), the dynamic window was the first technique to address kinematics and dynamics to carry out motion at high speeds (Fox et al., 1997), Curvature-Velocity method was a similar method developed alternatively (Simmons, 1996), Nearness diagram navigation was the first technique to address motion in troublesome scenarios (Minguez & Montano, 2004), etc

Additionally, extensions of the previous techniques were developed, for example:

Based in different potential functions (Tilove, R.B., 1990; Koren & Borenstein, 1991; Chenqing et al., 2000; Azarm & Schmith, 1994; Borenstein & Koren, 1989). Based in the vector field histogram (Ulrich & Borenstein, 1990; Ulrich & Borenstein, 2000; Borenstein & Raschke, 1992; Yang et al., 2000; Bell et al., 1994). Based in the elastic bands (Brock & Khatib, 2000; Khatib et al., 1997). Based in the dynamic window approach (Brock & Khatib, 1999; Stachniss & Burgard, 2002). Based in the nearness diagram (Minguez et al., 2004; Minguez, 2005; Marques, 2001), etc.

Also, there are hybrid methods that combine these techniques with tactical planners (Yang et al., 2000; Brock & Khatib, 1999; Minguez et al., 2001; Stachniss & Burgard, 2002; Philippsen & Siegwart, 2003).

The navigation system gives to robot the capability to move between given locations. There are several metrics that can be used to evaluate the performance of a navigation system, but none of them are able to indicate the quality of the whole system. Therefore it is necessary to use a combination of different indexes that quantify different aspects of the system. Having a good set of performance metrics is useful for: Optimizing algorithm parameters, testing navigation performance within a variety of work environments, making a quantitative comparison between algorithms, supporting algorithm development and helping with decisions about the adjustments required for a variety of aspects involved in system performance (Cielniak et al., 2005).

In autonomous navigation and obstacle avoidance, typical performance criteria are: (Minguez, 2008), (Álvarez, 1998)

1. Mission success: number of successful missions.
2. Path length: distance traveled to accomplish the task.
3. Time: time taken to accomplish the task.
4. Collisions: number of collisions per mission, per distance and per time.
5. Obstacle clearance: minimum and mean distance to the obstacles.
6. Robustness in narrow spaces: number of narrow passages successfully traversed.
- 7- Smoothness of the trajectory: relative to control effort.

This performance metrics can be classified in the following importance order:

- A- Metrics that consider the security in the trajectory or proximity to obstacles
- B- Metrics that consider the dimensions of the trajectory towards the goal
- C- Metrics that evaluate the smoothness of the trajectory

In the following, a set of different performance metrics are described.

2.1 Security metrics (Álvarez, 1998)

These metrics express the relationship between the security with which the robot travels through a trajectory, taking into account the distance between the vehicle and the obstacles in its path.

Security Metric-1 (SM1): Mean distance between vehicle and the obstacles through the entire mission measured by all the sensors; the maximum value will be produced in an obstacle free environment. If the deviation of the index from its maximum value is low, it means that the chosen route passed through obstacles free area.

Security Metric-2 (SM2): Minimum mean distance to obstacles. This is taken from the average of the lowest value of the n sensors. This index gives an idea of the risk taken through the entire mission, in terms of the proximity to an obstacle. In an obstacles free environment $SM1 = SM2$ is satisfied.

Minimum Distance (Min): Minimum distance between any sensor and any obstacle through the entire trajectory. This index measures the maximum risk taken throughout the entire mission.

2.2 Dimension metrics

The trajectory towards the goal is considered in its time and space dimensions. In general, it is assumed that an optimal trajectory towards the goal is, whenever possible, a line with minimum length and zero curvature between the initial point (x_i, y_i) and the finishing point (x_n, y_n) , covered in the minimum time.

Length of the Covered Trajectory (P_L): is the length of the entire covered trajectory by the vehicle from the initial point to the goal. For a trajectory in the x-y plane, composed of n points, and assuming the initial point as $(x_1, f(x_1))$ and the goal as $(x_n, f(x_n))$, P_L can be calculated as:

$$P_L = \sum_{i=1}^{n-1} \sqrt{(x_{i+1} - x_i)^2 + (f(x_{i+1}) - f(x_i))^2} \quad (1)$$

Where $(x_i, f(x_i))$, $i = 1, 2, \dots, n$ are the n points of the trajectory in cartesian coordinates (Guo & Wang, 2003).

The length of a trajectory given by $y = f(x)$, in the x-y plane between the points $(a, f(a))$ and $(b, f(b))$, can also be calculated as (Selekwa et al., 2004)

$$P_{L_{approx}} \cong \int_a^b \sqrt{1 + (f'(x_i))^2} dx \quad (2)$$

Mean distance to the goal (Mgd): This metric can be applied to robots capable of following reference trajectories. An important aspect when determining the quality of the robot navigation system is the ability to follow a trajectory that aims to reach a goal, so, to evaluate the quality of the execution of the trajectory, the mean distance between the vehicle and goal is analyzed. The difference is more significant if the covered distance is shorter (Rosenblatt, 1997). The mean distance to the goal is defined by the square of the proximity to

the goal distance l_n , integrated across the length of the trajectory and normalized by the total number of points n :

$$l_n = \min\left(\forall n\left(\sqrt{(x_i - x_n)^2 + (f(x_i) - f(x_n))^2}\right)\right) \quad (3)$$

$$Mgd = \frac{\int_0^l l_n^2 ds}{n} \quad (4)$$

Control Periods (LeM): It is the amount of control periods. This metric relates to the number of decisions taken by the planner to reach the goal. If the robot moves with constant lineal speed (v), this gives an idea of the time needed to complete the mission (Álvarez, 1998).

2.3 Smoothness metrics

The smoothness of a trajectory shows the consistency between the decision-action relationship taken by the navigation system, and also, the ability to anticipate and to respond to events with sufficient speed (Rosenblatt, 1997). The smoothness in the way a trajectory is generated is a measure of the energy and time requirements for the movement; a smooth trajectory allows translates into energy and time savings (Dongqing, 2006). Additionally a smooth trajectory is also beneficial to the mechanical structure of the vehicle. Bending Energy (B_E): This is a function of the curvature, k , used to evaluate the smoothness of the robot's movement. For curves in the x-y plane, the curvature, k , at any point $(x_i, f(x_i))$ across a trajectory is given by:

$$k(x_i, f(x_i)) = \frac{f''(x_i)}{(1 + (f'(x_i))^2)^{\frac{3}{2}}} \quad (5)$$

The bending energy can be understood as the energy needed to bend a rod to the desired shape (Aguirre & Gonzales, 2000). B_E can be calculated as the sum of the squares of the curvature at each point of the line $k(x_i, f(x_i))$, along the length of the line L . So, the bending energy of a robot trajectory is given by:

$$B_E = \frac{1}{n} \sum_{i=1}^n k^2(x_i, f(x_i)) \quad (6)$$

Where $k(x_i, f(x_i))$ is the curvature at each point of the robot trajectory and n is the number of points in the trajectory.

The value of B_E is an average and does not show with clarity enough that some trajectories are longer than others. Therefore, TB_E can be used instead; this metric takes into account the smoothness and length of the trajectory simultaneously.

TB_E is defined by

$$TB_E = \int_a^b k^2(x) dx \quad (7)$$

And numerically,

$$TB_E = \sum_{i=1}^n k^2(x_i, f(x_i)) \quad (8)$$

In a straighter trajectory, the values B_E and TB_E will be lower, which is desirable since the energy requirement is increased according to the increase in the curvature of the trajectory. Smoothness of Curvature (*Smoo*): is defined by the square of the change in the curvature k of the trajectory of a vehicle with respect to the time, integrating along the length of the trajectory and normalized by the total time t (Rosenblatt, 1997).

$$Smoo = \frac{\int_0^l \left(\frac{dk}{dt} \right)^2 ds}{t} \quad (9)$$

3. Simulation Framework

In this section it's describe a simulation framework used for evaluation of mobile robot navigation algorithms. Simulation is one of the most important tools in robotic development. It enables the evaluation of different alternatives during the design phase of robot systems and may therefore lead to more general solutions. Also, the simulation supports the process of software development by providing a replacement for robots that are currently not available (e.g. broken or used by another person) or not able to endure long running experiments (e.g. learning tasks). Finally, it is much easier to build scenarios using a simulator. The execution of robot programs inside a simulator offers the possibility to perform an easier and faster debugging phase before the first real experiment. (Calisi et al, 2008)

The simulation framework is compound for a Giraa_02 mobile robot model (see Fig 1a), and a 6m x 4m structured environment with static obstacles (see Fig. 2).

The Giraa_02 robot has a cylindrical structure of 30cm diameter and approximately 20cm height. It has 8 ultrasound and 8 infrared sensors distributed equally around the robot's circumference, for the simulation, only 8 infrared sensors were taken into account, these have a range of $d= 26.5\text{cm}$ and $a= 15$ degree detection cone; figure 1b. The vehicle has an odometric system and a differential locomotion system (Muñoz et al., 2006).

Data acquisition in the mobile robot, which occurs during each control period, consists of the current position of the robot and its orientation (x_i, y_i, θ_i) . The eight (8) proximity sensors are also read, the maximum reading being 26.5 cm, so that, if the robot spends n control periods reaching the goal, there is an array of $n \times 11$, and n sampling points per 11 pieces of data (3 coordinates and 8 sensors).

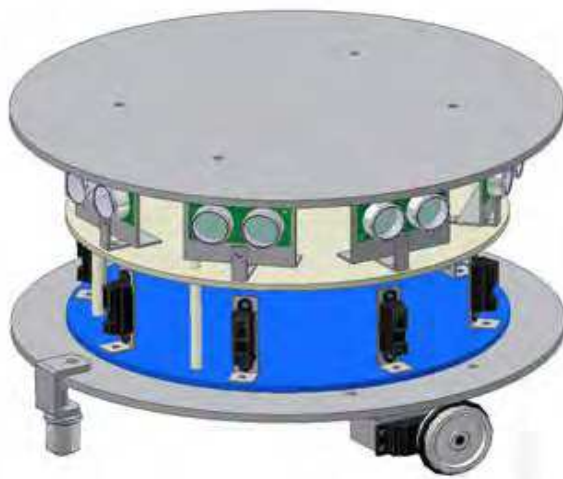


Fig. 1a. Mobile Robot Giraa_02

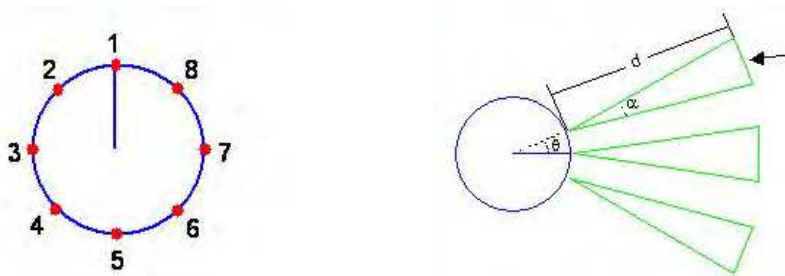


Fig. 1b. Infrared Sensor Configuration

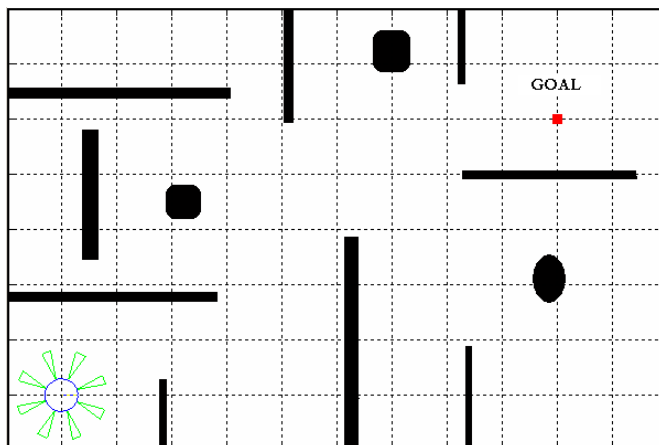


Fig. 2. Test Scenario for Mobile Robot Navigation

4. Definition of the evaluation protocol

In this section it describe a protocol used for evaluation of mobile robot navigation algorithms, and it's explains the proposed benchmark for navigation algorithms.

An evaluation protocol a systematic and ordered way to make the test. As a navigation benchmark, it's common to choose a "towards a goal" mission; the objective is to execute a navigation mission from a starting point to a final point (goal).

An order of importance for evaluating the navigation characteristics can be established as follows:

1. The mean distance between the vehicle and the obstacles during the trajectory
2. The distance covered by the vehicle between the starting point and the goal
3. The time needed to complete the mission
4. The smoothness of the trajectory

The first point considers the security of the trajectory and measures the risk taken by the robot in its movement towards the goal. The second and third points measure aspects related to the planning of the trajectory and the fourth point considers the quality of the trajectory according to the energy and time required for the movement. These characteristics can be analyzed using the following set of performance metrics:

1. SM1, SM2 and Min are proposed for evaluating security.
2. PL and LeM are proposed for evaluating the trajectory
3. TB_E is proposed for evaluating the smoothness of the trajectory.

For general purposes, only one metric is required for each one of the 3 categories described in section 2, but the use of various metrics helps to improve the analysis. In this case, the indexes were selected according to the capabilities of the GIRAA_02 mobile robot, considering the information provided by its data acquisition system; the readings from all the sensors are available for each point of the path, allowing the calculation of SM1, SM2, and Min. The Mgd index does not apply in this navigation mission since it applies when a trajectory is followed; TB_E is proposed because it analyses the smoothness and length of the path. Also, this metric is numerically simpler and more precise, making it easier to calculate than the other metrics.

5. Tests and Results (Muñoz et al, 2007)

The control algorithms provide basic capabilities for the mobile robot, such as the ability to evade obstacles and to generate a trajectory towards a goal. In this case, the framework is used for the evaluation of a navigation mission between two points (towards a goal). For this mission, it can use several navigation methods, like mentioned in section 2. As example, we go to compare the performance of two methods: potential field and AFREB.

5.1 Control Algorithm 1

This is a reactive algorithm based on a potential field method (Khatib, 1986; Krogh & Thorpe, 1986), which produces two different behaviors: first, goal attraction, and second, obstacles repulsion (keep away from objects). The planning of the movement consists in the

proper combination of both behaviors in such a way that the robot reaches the goal without collisions (See fig 3.). This combination is achieved using a vector sum (Latombe, 1996).

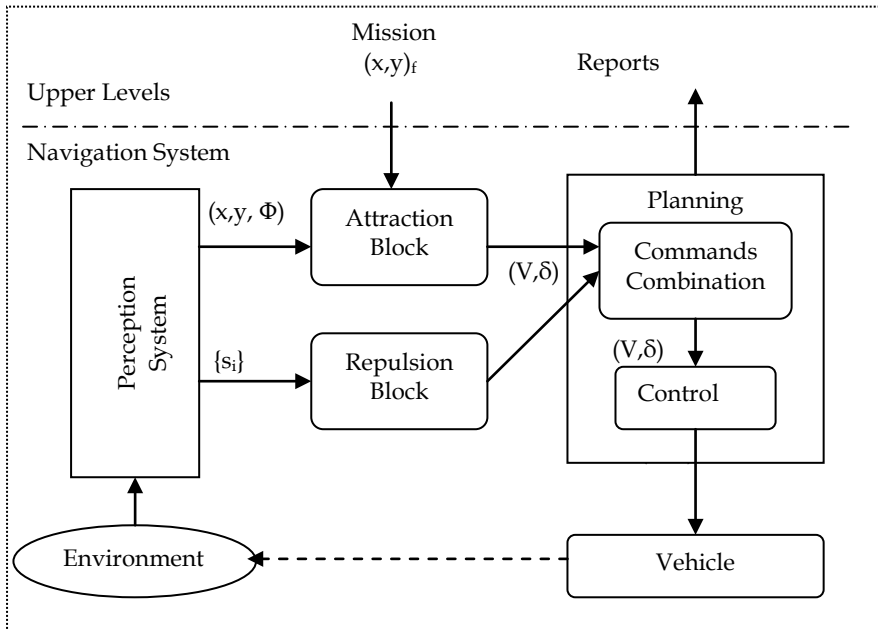


Fig. 3. Diagram of navigation based in potential field method

Where

(x, y, Φ) : current position and orientation

$(x, y)_f$: final goal position

S_i : sensors information (length measurement sensors)

(V, δ) : velocity and orientation angle commands

(v, w) : linear and angular velocity

5.2 Control Algorithm 2

This control algorithm is based on reactive behaviors, denominated AFREB "adaptive fusion of reactive behaviors" (Zalzala & Morris, 1996), (Gachet et al, 1994). By using a neural net, an appropriate combination of the behaviors can be achieved, so that the system is able to perform tasks, such as navigation towards a goal, while evading obstacles in its path. The AFREB control architecture is depicted in Fig 4, basically consists of the following modules: behavioral fusion, fusion supervisor, behavior primitives (1, 2, ..., n), and executor.

A primitive behavior can be characterized by a temporal sequence of appropriate values for linear velocity $v(i)$, and curvature $k(i)$ which cause the robot to exhibit the prespecified response to sensorial information. Thus it is defined the output of a primitive behavior $c(i)$ as a vector:

$$c(i) = (v(i), k(i))^T \quad (10)$$

Where

The variable i denotes the i -th cycle of the robot controller

(x,y, Φ) : current position and orientation

$(x,y)_f$: final goal position

$c1..cn$: Behavior primitives output

a_i : Behavior weighs (coefficients)

c : Emergent behavior (mission), linear and angular velocity vector (v,w)

The primitive behaviors implemented are:

$c1$: goal attraction

$c2$: perimeter following (contour left - CW)

$c3$: perimeter following (contour right - CCW)

$c4$: free space

$c5$: keep away (from objects)

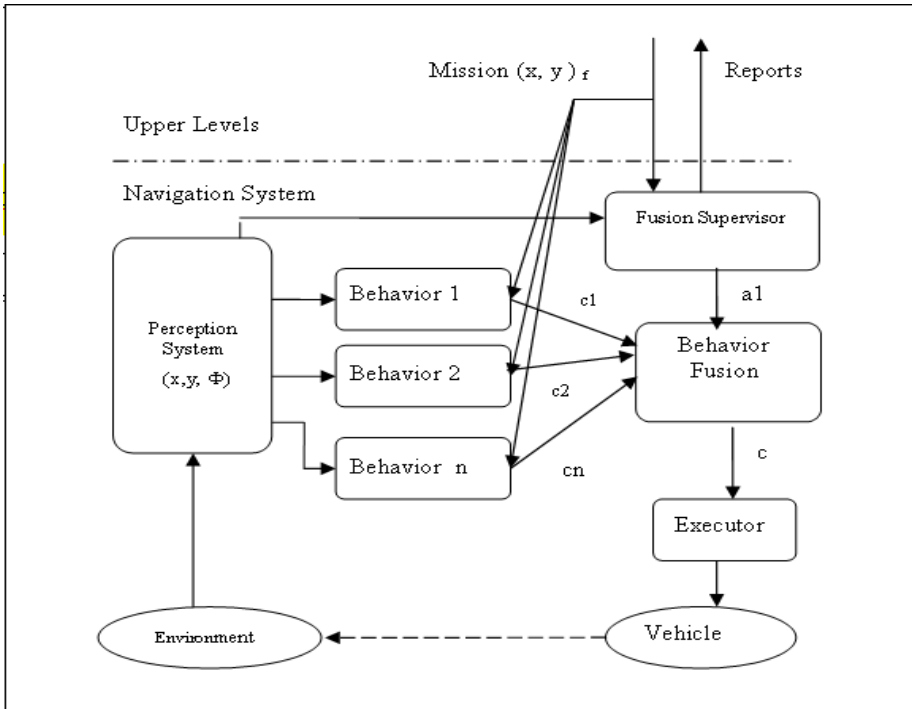


Fig. 4. Diagram of navigation based in AFREB

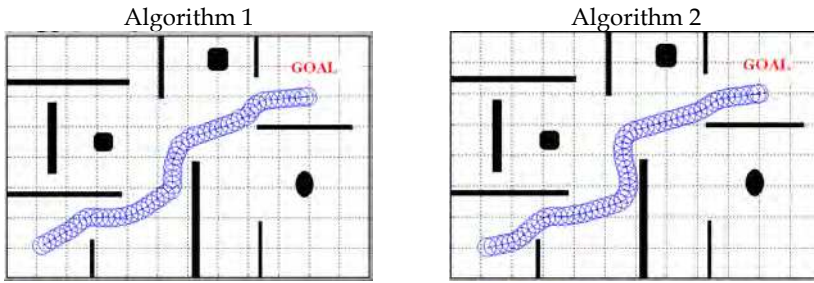
If $c1..cn$ are the output of each primitive behavior, then the output of an emergent behavior (i.e., towards a goal, other mission, etc) is:

$$c = \sum_{i=1}^N a_i c_i \tag{11}$$

Where a_i coefficients, with $0 \leq a_i \leq 1$, are found by an appropriate combination of measurement information provided by the perception system.

5.3 Results

The paths generated by the algorithms, in the scenario 1, are shown in figure 5. The table 1 summarizes the results obtained from the simulation using both control algorithms for all scenarios, according to the quality metrics proposed.



Scenario 1. Start point (50,50) Goal (500,300)

Fig. 5. Paths generated by the control algorithms

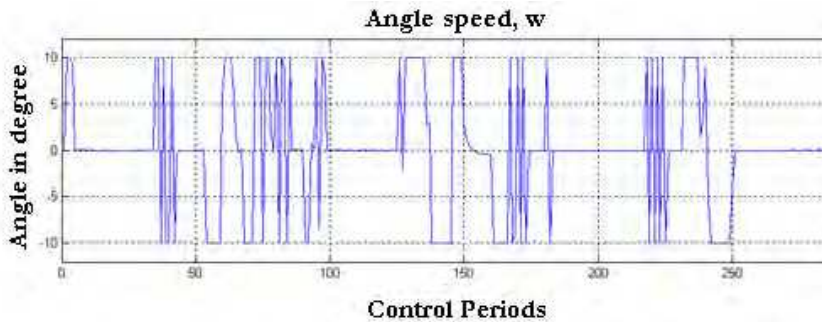


Fig. 6. Smoothness of the trajectory, change in the robot heading each control period, generated by algorithm 1.

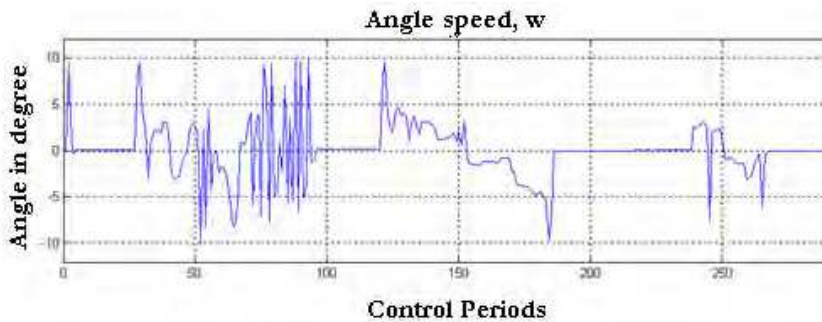


Fig. 7. Smoothness of the trajectory, change in the robot heading each control period, generated by algorithm 2

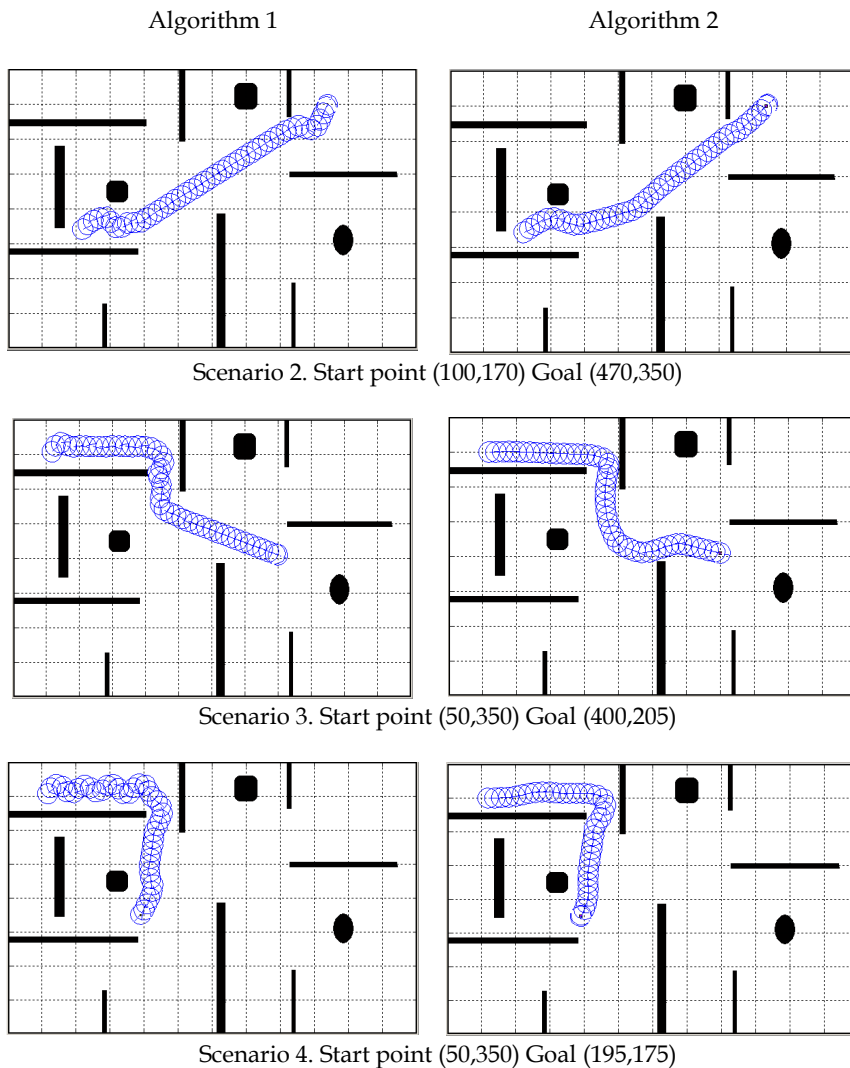


Fig. 8. Paths generated by the control algorithms in scenarios 2, 3, 4.

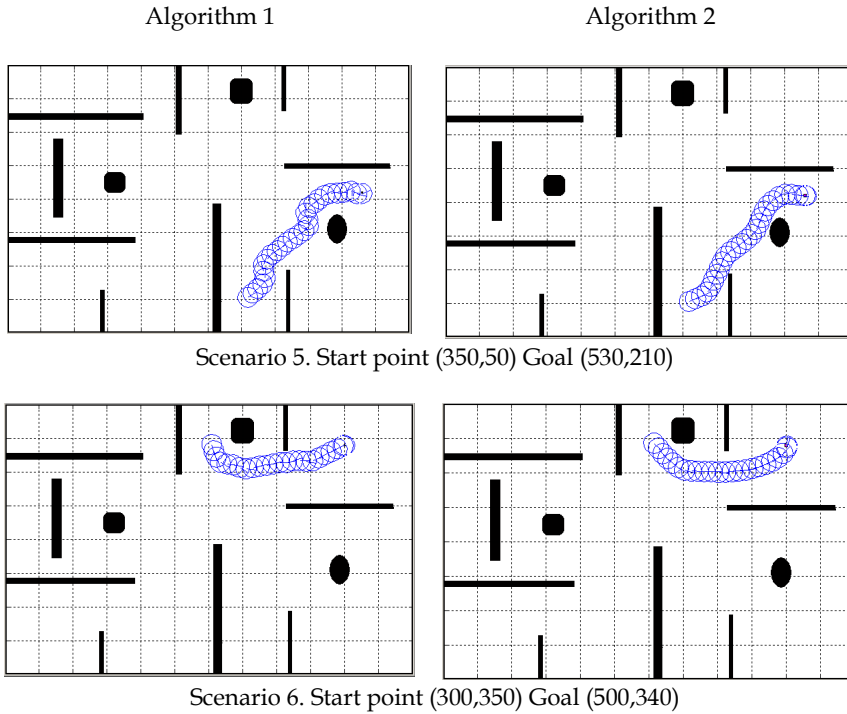


Fig. 9. Paths generated by the control algorithms in scenarios 5 and 6.

Metric	SM1 [cm]		SM2 [cm]		Min [cm]		PL [cm]		LeM		TB _E	
	Alg. 1	Alg. 2	Alg. 1	Alg. 2	Alg. 1	Alg. 2	Alg. 1	Alg. 2	Alg. 1	Alg. 2	Alg. 1	Alg. 2
1	26.1	25.6	18.3	17.3	11	7	562.7	581.9	283	292	0.2463	0.0846
2	25.9	25.7	13.0	14.0	3	7	441.8	429.9	222	216	0.2810	0.0718
3	25.4	23.9	10.0	8.9	3	3	456.7	462.9	234	235	0.5873	0.0120
4	25.0	24.4	13.0	12.4	7	3	395.7	359.9	199	181	0.4007	0.0140
5	25.9	24.9	19.4	16.4	15	3	275.8	259.9	139	131	0.1626	0.0394
6	26.0	25.9	19.7	22.6	7	11	229.9	229.9	116	116	0.1722	0.0469

Table 1. Robot performance

5.4 Analysis of results

In scenario 1, the algorithm 1 uses less control periods, and consequently takes less time to complete the mission, and covers a safer and shorter path, the figure 6 shows that algorithm 1 produces a great orientation change for each control period. Algorithm 2 covers a smoother path, the figure 7 shows a smaller change in the orientation during each control period, with consequent energy saving and less structural effort on the robot.

From table 1, it can be deduced that the difference between both algorithms in the trajectory and time taken is only 3.3% and 3.1% respectively. The robot programmed with algorithm 2 passed a minimum 7 cm from any obstacle, which is acceptable for a 30cm diameter robot; also, it showed approximately 65% less bending energy than algorithm 1. For these reasons, algorithm 2 can be considered the best choice.

The figures 8 and 9, shows other sceneries. The algorithm 1 has the tendency to generate safer trajectories, because the robot transits, on average, through zones that are farther from the obstacles, it is explained because the closer the robot is to the obstacles, the higher the repulsion potential, this way, the robot succeeds in keeping away from them. Despite Algorithm 2 is ruled by the same repulsion principle, the command that finally guides the robot, is a combination of 5 different behaviors, which reduces the role of the repulsion potential, but without having collisions. The main difference between all simulations is that Algorithm 2 generates smoother trajectories than Algorithm 1.

The wall following behavior CW and CCW present in Algorithm 2, makes the robot able to transit through narrow zones like corridors, keeping a safe distance from the obstacles and also generating smooth trajectories, which doesn't happen with Algorithm 1 as it is seen in sceneries 3 and 4. In general, Algorithm 2 exhibits a better performance, the bending energy index is always smaller than in the Algorithm 1, even in the sceneries 2, 4 and 5, generates shorter trajectories and uses less time to complete the mission, and in the scenario 6, generates a safe trajectory.

6. Conclusion

This chapter provides an analysis on several performance metrics to contrast mobile robots navigation algorithms including safety, dimension and smoothness of the trajectory. The suggested metrics are quite straight forward, however, it has been shown that they can be used together to systematize simulated or experimental studies on control algorithms for mobile robot navigation.

A very simple application example was presented. The obtained results demonstrate the need to establish a procedure that can be used to analyze and compare control algorithms using several performance metrics. This is an open topic of research. It becomes necessary to establish proper approaches and benchmarking procedures, for example, using standard framework of benchmarks for navigation algorithm and performance evaluation.

This metrics can be applied in simulated environments, but the performance metrics evaluation is more important in real environments. Many of the challenges in robot navigation come from the challenges of real environments, such an uncertainty in the sensors and the errors in odometry, and this, in general, is not considered in simulation.

7. References

- Aguirre, E. & Gonzales, A. (2000). Fuzzy Behaviors for mobile robot navigation, design, coordination and fusion. *International Journal of Approximate Reasoning*, vol. 25, 2000, pp. 255-289.
- Álvarez, J. (1998). Planificación del movimiento de vehículos autónomos basada en sensores. *Tesis doctoral*, Universidad de Oviedo, Oviedo, España, pp. 178.

- Azarm, K. & Schmith, G. (1994). Integrated mobile robot motion planning and execution in changing indoor environments. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Munchen, Germany, 1994, pp. 298-305.
- Bell, D.; Levine, S.; Koren, Y.; Jaros, L. & Borenstein, J. (1994). Design criteria for obstacle avoidance in a shared-control system. *Proceedings of RESNA Conference*, Nashville, 1994.
- Borenstein, J. & Koren, Y. (1989). Real-Time Obstacle Avoidance for Fast Mobile Robots. *Proceedings of IEEE Transactions on Systems, Man and Cybernetics*, Vol. 19, pp. 1179-1187.
- Borenstein, J. & Koren, Y. (1991). The Vector Field Histogram-Fast Obstacle Avoidance for Mobile Robots. *Proceedings of IEEE Transactions on Robotics and Automation*, pp. 278-288.
- Borenstein, J. & Raschke, U. (1992). Real-time obstacle avoidance for non-point mobile robots. *Proceedings of Fourth World Conference on Robotics Research*, Michigan, USA.
- Brock, O. & Khatib, O. (1999). Real High-Speed Navigation Using the Global Dynamic Window Approach. *Proceedings of IEEE International Conference on Robotics and Automation*, Detroit, MI, pp. 341-346
- Brock, O. & Khatib, O. (2000). Real-Time Replanning in High-Dimensional Configuration Spaces using Sets of Homotopic Paths. *Proceedings of IEEE International Conference on Robotics and Automation*, San Francisco, USA, pp. 550-555.
- Calisi, D., Iocchi, L., & Nardi, D. (2008). A unified benchmark framework for autonomous mobile robots and vehicles motion algorithms (MoVeMA benchmarks). In RSS workshop on experimental methodology and benchmarking, Zurich, Switzerland.
- Chenqing, L.; Ang, M.; Krishnan, H. & Yong, L. (2000). Virtual Obstacle Concept for Local-minimum-recovery in Potential-field Based Navigation. *Proceedings of IEEE International Conference on Robotics and Automation*, San Francisco, USA, pp. 983-989.
- Cielniak, G.; Treptow, A. & Duckett, T. (2005). Quantitative Performance Evaluation of a People Tracking System on a Mobile Robot. *Proceedings of the European Conference on Mobile Robots (ECMR)*, Ancona, Italy.
- Dongqing, S. (2006). Aerial robot navigation in cluttered urban environment, *PhD Thesis*, The Florida State University, Florida, USA, pp. 87.
- Eurongemsig, (2008). Special Interest Group on Good Experimental Methodology in Robotics European Robotics Research Network (EURON), [Online]. Available: www.heronrobots.com/EuronGEMSig/
- Evans, J. & Messina, E. (2000). Performance Metrics for Intelligent Systems, *Proceeding of the Performance Metrics for intelligent Systems Workshop*, August 2000, MD, Gaithersburg.
- Fox, D.; Burgard, W. & Thrun, S. (1997). The Dynamic Window Approach to Collision Avoidance. *Proceedings of IEEE Robotics and Automation Magazine*. Vol. 4, Num. 1, 1997.
- Gachet D.; Salichs, M. A.; Moreno, L.; Pimentel, J. R. (1994). Learning Emergent Tasks for an Autonomous Mobile Robot. *Proceedings of the International Conference on Intelligent Robots and Systems (IROS '94)*.
- Guo, Y & Wang, J. (2003). A new performance based motion planner for nonholonomic mobile robots, *Proceedings of the 3rd performance metrics for the Intelligent Systems Workshop (PerMIS'03) NIST*, September 2003, Gaithersburg, MD.

- Khatib, M.; Jaouni, H.; Chatila, R. & Laumond, J. (1997). Dynamic Path Modification for Car-Like Nonholonomic Mobile Robots. *Proceedings of IEEE International Conference on Robotics and Automation*, Alburquerque, Mexico, pp. 2920–2925.
- Khatib, O. (1986). Real-Time Obstacle Avoidance for Manipulators and Mobile Robots. *Proceedings of International Journal of Robotics Research*, Vol. 5, pp. 90-98, Spring, 1986.
- Koren, Y. & Borenstein, J. (1991). Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation. *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 2, Sacramento, CA, pp. 1398–1404.
- Krogh, B.H. & Thorpe, C.E. (1986). Integrated Path Planning and Dynamic Steering control for Autonomous Vehicles. *Proceedings of IEEE International Conference on Robotics and Automation*, San Francisco, USA, pp. 1664–1669.
- Latombe, J.C. (1996). Robot Motion Planning, *Kluwer Academic Publishers*, 4th Edition, 1996, Boston.
- Marques, C. (2001). Multi-sensor navigation for soccer robots. Master's thesis. In: *Instituto Superior Tecnico*, Portugal, 2001.
- Minguez, J. (2005). The Obstacle Restriction Method (ORM): Obstacle Avoidance in Difficult Scenarios. *Proceedings of IEEE International Conference on Intelligent Robot and Systems*, Edmonton, Canada, 2005.
- Minguez, J. (2008). Robot Obstacle Avoidance Papers Using Experiments, In: *Good experimental methodology guidelines*, Bonsignorio, F.; Hallam J., & del Pobil, A. P. (Ed), Special Interest Group on Good Experimental Methodology in Robotics European Robotics Research Network (EURON), Tech. Rep., 2008.
[Online]. Available:
www.heronrobots.com/EuronGEMSig/Downloads/GemSigGuidelinesBeta.pdf
- Minguez, J. & Montano, L. (2004). Nearness Diagram (ND) Navigation: Collision Avoidance in Troublesome Scenarios. *Proceedings of IEEE Transactions on Robotics and Automation*, Vol. 20, Num. 1, pp. 45–59.
- Minguez, J. ; Montano, L.; Simeon, N. & Alami, R. (2001). Global Nearness Diagram Navigation (GND). *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 33–39, Seoul, Korea .
- Minguez, J.; Osuna, J. & Montano, L. (2004). A Divide and Conquer Strategy to Achieve Reactive Collision Avoidance in Troublesome Scenarios. *Proceedings of IEEE International Conference on Robotics and Automation*, Minnesota, USA.
- Muñoz, N.; Andrade, C. & Londoño, N. (2006). Diseño y construcción de un robot móvil orientado a la enseñanza e investigación, *Ingeniería & Desarrollo Ed. 9*, 2006.
- Muñoz, N.; Valencia, J. & Londoño, N. (2007). Evaluation of Navigation of an Autonomous Mobile Robot, *Proceedings of the Performance Metrics for Intelligent Systems (PerMIS) Workshop*, Washington, DC, EEUU.
- Philipsen, R. & Siegwart, R. (2003). Smooth and efficient obstacle avoidance for a tour guide robot. *Proceedings of IEEE International Conference on Robotics and Automation*, Taipei, Taiwan.
- Quinlan, S. & Khatib, O. (1993). Elastic Bands: Connecting Path Planning and Control. *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 2, pp. 802–807, Atlanta, USA.

- Rosenblatt, J. (1997). DAMN: Distributed Algorithm for Mobile Navigation. PhD. Thesis, *Carnegie Mellon University Robotics Institute, Pittsburg, PA, 1997.*
- Simmons, R. (1996). The Curvature-Velocity Method for Local Obstacle Avoidance. *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 3375–3382, Minneapolis, USA.
- Selekwa, M.; Collins, E. & Combey, J. (2004). Multivalued Versus univalued Reactive Fuzzy Behavior Systems for Navigation Control of Autonomous Ground Vehicles, *Proceedings from the 17th annual Florida Conference on the Recent Advances in Robotics FCRAR2004*, May 2004.
- Stachniss, C. & Burgard, W. (2002). An Integrated Approach to Goal-directed Obstacle Avoidance under Dynamic Constraints for Dynamic Environments. *Proceedings of IEEE-RSJ International Conference on Intelligent Robots and Systems*, pp. 508–513, Switzerland, 2002.
- Tilove, R.B. (1990). Local Obstacle Avoidance for Mobile Robots Based on the Method of Artificial Potentials. *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 2, pp. 566–571, Cincinnati, OH, 1990.
- Ulrich, I. & Borenstein, J. (1990). VFH+: Reliable Obstacle Avoidance for Fast Mobile Robots. *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1572–1577.
- Ulrich, I. & Borenstein, J. (2000). VFH*: Local Obstacle Avoidance with Look-Ahead Verification. *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 2505–2511, San Francisco, USA, 2000.
- Wong, S; Middleton, L. & MacDonald, B. (2002). Performance metrics for robot coverage task, *Proceedings Australasian Conference on Robotics and Automation ACRA*, Auckland, New Zealand, 2002, pp. 7-12
- Yang, H.; Borenstein, J. & Were, D. (2000). Double-vfh: Reliable obstacle avoidance for large, non-point, omni-directional mobile robots. *Proceedings of ANS Conference on Robotics and Remote Systems*, Pittsburg, PA, 1999.
- Zalzala, A. & Morris, A. (1996). *Neural Networks for Robotic Control*, Ellis Horwood, 1996, pp.278.



Mobile Robots Navigation

Edited by Alejandra Barrera

ISBN 978-953-307-076-6

Hard cover, 666 pages

Publisher InTech

Published online 01, March, 2010

Published in print edition March, 2010

Mobile robots navigation includes different interrelated activities: (i) perception, as obtaining and interpreting sensory information; (ii) exploration, as the strategy that guides the robot to select the next direction to go; (iii) mapping, involving the construction of a spatial representation by using the sensory information perceived; (iv) localization, as the strategy to estimate the robot position within the spatial map; (v) path planning, as the strategy to find a path towards a goal location being optimal or not; and (vi) path execution, where motor actions are determined and adapted to environmental changes. The book addresses those activities by integrating results from the research work of several authors all over the world. Research cases are documented in 32 chapters organized within 7 categories next described.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Nelson David Munoz Ceballos, Jaime Alejandro Valencia and Nelson Londono Ospina (2010). Quantitative Performance Metrics for Mobile Robots Navigation, Mobile Robots Navigation, Alejandra Barrera (Ed.), ISBN: 978-953-307-076-6, InTech, Available from: <http://www.intechopen.com/books/mobile-robots-navigation/quantitative-performance-metrics-for-mobile-robots-navigation>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.