Docking and Charging System for Autonomous Mobile Robots

Riccardo Cassinis, Fabio Tampalini, Paolo Bartolini and Roberto Fedrigotti
Department of Electronics for Automation
University of Brescia
IT - Brescia
{riccardo.cassinis,fabio.tampalini}@ing.unibs.it, ilbarto@virgilio.it, roberto.fedrigotti@libero.it

***Abstract*— In this paper we describe a safe and reliable system for the autonomous batteries' charging of a mobile robot.**

The project includes an electro-mechanical part, consisting of a station into which the robot can dock and wait during the charge, and a software part, to program the robot with the essential operations it has to execute to reach the docking-charging station.

The main part of the resarch is in the docking algorithm, based on a low-cost, simple and reliable alignment method inspired by an old aid to navigation: the use of range lights.

***Index Terms*— Docking station, charging station, alignment, behaviour**

I. Introduction

The main aim of mobile robotics is to create completely autonomous robots, in the sense that they must be able to complete their tasks without any human intervention.

In order to achieve this goal a method to let the robot itself handle autonomous docking and charging problem is needed, especially when the tasks are longer than the robots' battery charge.

In our experience, we encountered this situation in a project we are working on at the Advanced Robotics Laboratory of the University of Brescia: an application for a *Pioneer 3* robot called Morgul (acronym of *MObile Robot for Guarding University Laboratories*), programmed for patrolling in a business environment. In this example, the robot should be fully autonomous over very long shifts, for instance during weekends or holidays.

The developed system takes advantage from the precision of the docking algorithm, inspired by an ancient navigation aid: the use of range lights (figure 1). As stated in [Bowditch, 2002], "*Range lights are light pairs that indicate a specific line of position when they are in line. The higher rear light is placed behind the front light. When the mariner sees the lights vertically in line, he is on the range line. If the front light appears left of the rear light, the observer is to the right of the range line; if the front appears to the right of the rear, the observer is left of the range line.*"

This short–range aid to navigation is used to indicate a safe route or the centerline of a channel and to facilitate the entrance into harbours. It's a simple, precise and historically well known method; for examples it was used in the cities of Genoa and Pisa during the medieval time.

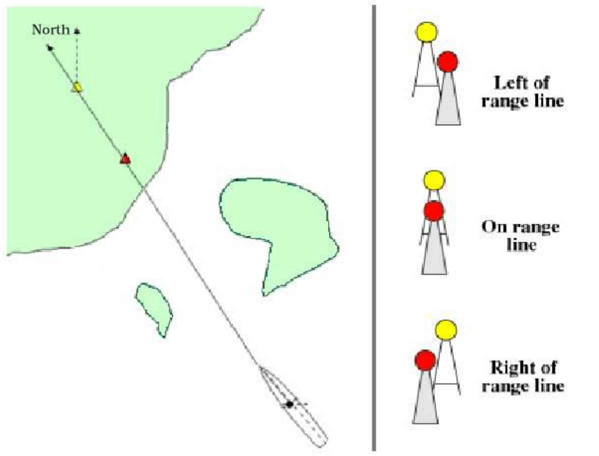

Fig. 1. Range lights are widely used in sailing, to indicate a safe route passing through channels or entering ports, avoiding dangerous obstacles.

In our system this method is exploited to provide high precision in positioning and reliability, thanks to the great simplicity of the electronics involved.

Section II deals with similar projects that have already been completed; in section III we will describe in detail the solution we realized and tested; in section IV we will present the results of our work, its capabilities and some further perspectives for our project.

II. Related work

In the last years, there have been some successful attempts in implementing the docking capability on autonomous mobile robots.

The first robots able to dock and recharge were "Tortoises", developed by Grey Walter: these robots used a light following behaviour to find their way into a hut, containing a battery charger that made contact with the robot. There was a light bulb attached to the hut that guided the robot into the charging station [Walter, 1953].

More recently, Hada and Yuta presented a battery charging system for long-term activity mobile robots, using infrared sensors and a reflective tape on the floor as target to position the robot for docking [Hada and Yuta, 1999].

Oh, Zelinsky and Taylor realized a complex docking system inspired to the aircraft landing: in this project, the robot approached the docking station thanks to a long-range infrared beacon and a sonar; when the robot was in proximity of the station, a Sick laser range finder was used to align the robot to a grid target, that had a pattern designed to distinguish it from the surrounding environment [Oh et al., 2000].

ActivMedia provides a recharging station for Pioneer robots, which requires additional complex electronics to be installed internally. Vision is used to find the charging station, which consists of a plate placed on the ground, with upward protruding spring contacts for the robot to drive over. Mating contacts are mounted under the robot, and an integrated sensor informs the robot when it is located over the charger [ActivMedia Robotics, 2003].

From the mechanical point of view, an interesting docking system is also explained in [Silverman et al., 2002], where a particular mechanism allows a high angular and displacement error during the docking process. A combination of vision and laser beacons is then deployed to perform the autonomous recharging of a Pioneer 2DX robot.

III. The Proposed Solution

The solution we propose is a complete system for the autonomous charging of the batteries of a mobile robot. The main two aspects to consider in this project are naturally the docking and the charging procedures.

As for the charging procedures, we integrated into a solid and firm structure the essential electrical circuits to supply the needed power. This structure itself supports the robot in the docking operations showing two lamps that act like active markers, helping the robot in finding the way to reach it in the way used by vessels to enter in a channel. This localization is possible thanks to the vision system mounted on the robot, constitued by a webcam and an image grabber installed on the robot's computer. The whole system is represented in figure 2.

Fig. 2. The whole system: photograph of the robot docking into the charging station.

The use of ranges is an easy, popular and reliable aid to navigation that defines a line of position, usually with great accuracy that doesn't depend on expansive shipboard equipment. As stated in the introduction, approaching the front range markers, if the two marks are exactly in range, the vessel's position is exactly along the range line. If the lower marker is to the left (right), the vessel must alter its course to the left (right) to rejoin the range.

Because of geometric considerations, the horizontal angle between the range markers seen by a vessel at a fixed distance far away from the channel centerline increases with decreasing distance. Thus, the sensitivity of the angle to side-to-side excursions increases as the vessel draws closer to the range lights (see figure 3).

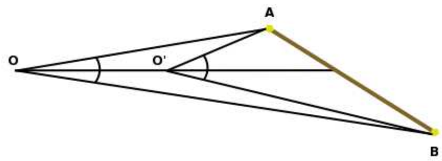

Fig. 3. The sensitivity of the angle increases as the vessel draws closer to the range lights.

Following this idea, the markers on the docking station acts like range lights for our robot.

When a low battery power is detected, the robot stops the execution of its current tasks and come in proximity of the docking station. Here the robot, thanks to the webcam (that we see attached under the blinking lamp on its top), uses the information provided by the markers' positions to calculate its displacement and to evaluate its distance to the target. After these considerations, it can then move forward or rotate to assure a safe docking to the structure.

Once entered the station, the contacts on the robot's front side exactly fit into the matching electrical contacts mounted on the fixed structure.

In the following sections we are going to describe some implementation details, explaining the considerations that led us to these choices, in particular about: the docking station, the algorithms for localize the station and the robot's behaviours to reach it.

A. The docking station

The docking station (that we see in figure 4) is a wooden box, open on a side and without the cover. The front opening is 7 cm wider than the robot, in order to allow small displacement and angular errors. The ground gives solidity to the structure and is covered of cork mainly for aesthetic reasons.

Inside the box, on the back, there are four stainless steel clasps, opportunely designed to fit with the horizontal bar terminals mounted on the robot (shown in figure 5). The clasps are attached to a wooden support, to avoid undesired short circuits. Their middle displacement is studied to guarantee the electrical connection with the matching bars

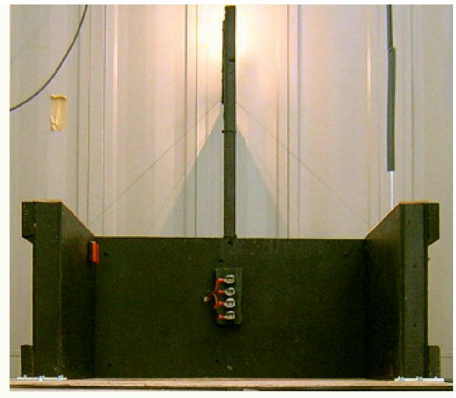

Fig. 4. Photograph of the docking station.

Fig. 5. Photograph of the electrical bar contacts mounted onto the robot.

on the robot, regardless of the way the robot enters the box.

On the two sides of the box, near the contacts, there is a system composed of a photoelectric cell and a beacon, used to detect the presence or the absence of the robot in charge.

Finally, the docking station shows two lights on the top, that act as range lights (depicted in figure 6). The distances between the markers have been calculated considering that they must guarantee a certain accuracy and that they have to be placed in a way that allows the webcam to see them both, since the robot is some meters far from the box, until the end of the docking operations.

As for the accuracy, we considered that when the markers' barycenters have the same horizontal position in an image, they are believed as in line; if then we think about the area concerning a pixel, the markers can have in fact a position different from the perfect alignment (as shown in figure 7), so we calculated the error (indentified as the value of θ) derived from this situation, varying the distance AB between the two markers.

Our purpose was to have an error (θ) less than 0.4 degrees at 3 meters of distance from the markers (OH), so, watching the results summarized in table I, we chose a distance between the markers (AB) of 45 cm.

B. Localization of Active Markers

In order to lead the robot into the docking station, the deviation regarding the optimal route of approach is needed. This information is supplied through the recognition of two

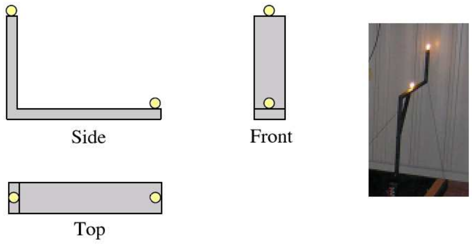

Fig. 6. The markers: configuration scheme and photograph.

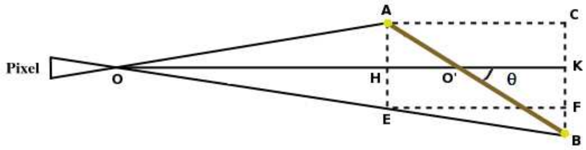

Fig. 7. The accuracy in aligment of our vision system: though points A and B are believed as in line, the real dimensions of the pixel cause an error (θ) of positioning.

active markers in the images acquired from a webcam on the robot. The markers are simply two small halogen lamps placed along the same direction but at different heights (see figure 6), in a particular configuration that makes possible to evaluate the direction from which the robot comes.

In fact, if the robot is perfectly aligned with the docking station, the lamps are vertically aligned too in the image, as shown in figure 8; otherwise, if it comes from a direction that is not the optimal one, the two lights are on different horizontal positions. An example is explained in figure 9: here the robot comes from the right and the topmost marker is on the right of the other light.

As stated above, the robot's behaviour depends on its direction and consequently the recognition of the markers' coordinates in the image assumes an important role in the

AB (cm)	θ (°)
20	0.827
22.5	0.738
25	0.666
27.5	0.608
30	0.559
32.5	0.518
35	0.482
37.5	0.451
40	0.424
42.5	0.401
45	**0.379**
47.5	0.361
50	0.343

TABLE I

The error θ decreases while the distance between the two markers (AB) raises; the data above are referred to an observer 3 meters far from the station (OH = 3m).

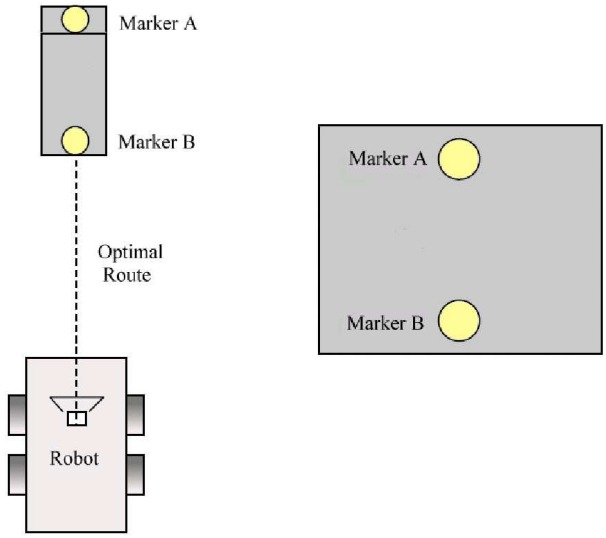

Fig. 8. Robot aligned with the docking station.

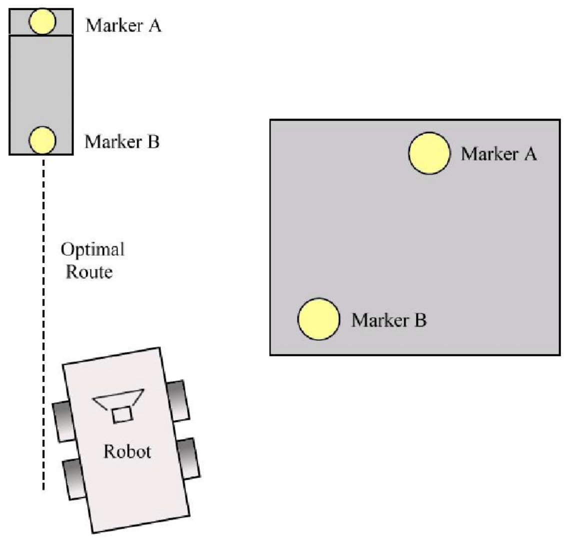

Fig. 9. Robot with a wrong guideline.

docking task. Active markers were chosen because they emit luminous energy towards the vision system and so they allow an easier recognition from the webcam through the calibration of some of the camera parameters, basically the gain and the shutter speed. The technique developed therefore takes advantage of the particular configuration of the lights in the images and in order to have a quick response, analyzes the images of the webcam in the gray scale.

Initially the software scans the lines of the image from the topleft apex moving towards right and when it encounters a pixel with value greater or equal than a threshold, then it is assumed that the first marker is found. The scan finishes when is found an entire line that doesn't contain an over-threshold pixel. In a similar way the inferior marker is searched with a new scanning that begins from the low-right apex moving left. The threshold plays a fundamental role in this algorithm, in fact a too much permissive value would erroneously consider groups of pixels not pertaining to a marker, while a too restrictive one wouldn't permit to find the markers. Moreover, an increase of the distance of the robot and the webcam from the docking station corresponds to a decrease in the number and in the value of the pixels belonging to the markers. The use of a fixed value for the threshold is not the optimal solution to the problem and so there has been implemented an algorithm to dynamically calculate it for each image. The search space is constituted by a range of values for the gray scale color initially bounded by 0 and 255, thus containing all possible values that pixels can assume. These values are updated once a loop by a binary algorithm to identify the highest valid threshold that allows to localize the two blobs representing the markers. The next step is the identification of the barycenters of the groups of pixels, using the following formulas:

$$x_b = \frac{\sum_{i=0}^{N} x_i}{N} \qquad y_b = \frac{\sum_{i=0}^{N} y_i}{N}$$

Where x_b and y_b are the coordinates of the barycenter, x_i and y_i are the coordinates of the pixels that belong to the blob and N is the number of pixels of the blob representing the marker. Finally those values are stored in a memory buffer, in order to be collected and used by the behaviour software in the Saphira environment to determinate the robot's movements.

C. Robot behaviours

The guidance system is the component that makes possible the correct docking of the robot when it is in proximity of the station with an accidental direction and with the markers in the webcam's field of vision.

In order to complete this task, the robot has to discover its direction compared with the axis of the charging station. This information is obtained from the markers' coordinates, stored in a memory buffer by the recognition algorithm, through the difference between the abscissas of the highest marker and the lowest. In fact a positive value of *distance* (refer to figure 10) means that the robot comes from the right, while in the opposite case it comes from the left. If the absolute value of *distance* is lower than the predefined DIST, the robot has a direction parallel to the axis of the station.

Otherwise, in order to catch up the axis of the recharging station following the short distance, the robot goes forward turning until the center of the two markers reaches an offset respect to the edge of the image, but maintaining always the markers in the field of vision (figure 11).

When the markers are vertically aligned, the robot is on the axis of the docking station, but yet with a wrong direction. So it turns until the abscissas of the markers are in the center of the image obtained from the webcam

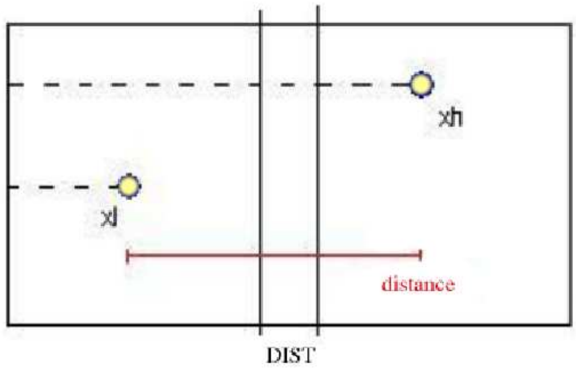

Fig. 10. *Distance* supply the direction of the robot regarding the station. The robot comes from the right so the value of *distance* is positive.

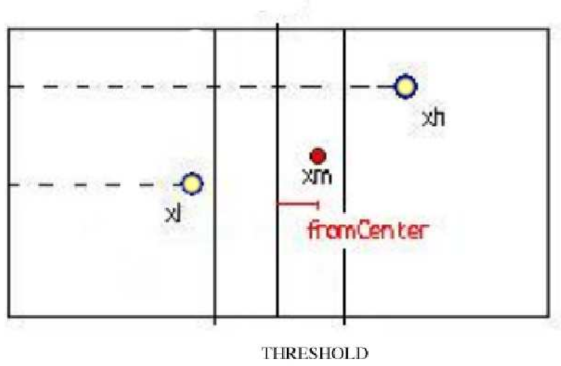

Fig. 12. *fromCenter* is lower than THRESHOLD, so the robot is on the axis of the docking station.

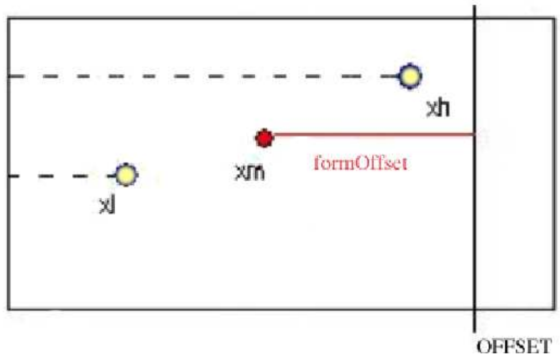

Fig. 11. To catch up the axis of the station, the robot turns until *fromOffset* is minimum.

(figure 12) and the absolute value of *distance* is lower then DIST (figure 10).

The robot has now the correct direction to enter the station, so it has only to go straight keeping the markers in this position in order to maintain its alignment. Once the robot is perfectly docked, it stops and a photoelectric cell determines the turn off of the markers and the exploitation of the docking task.

The system has been developed in the Saphira environment and is based on behaviours. So the algorithm of the approach has been divided in simpler sub-problems, based on different objective and strategy of resolution, each corresponding to a behaviour:

- catch up the axis of the station of it recharges: behaviour *SfAxAction*
- maintain itself in axis: behaviour *SfCenterAction*
- go ahead straight to enter the station: behaviour *SfForwardAction*.

Finally an action written in Colbert language loads those behaviours in the Saphira environment passing the opportune parameters and establishing their priority. The behaviour *SfAxAction* manages also the limit cases, therefore it must necessarily have the maximum priority.

IV. Conclusion and Perspectives

In conclusion, in this work it has been realized a system that allows a mobile robot to autonomously recharge its batteries when it detects a low power supply. Thanks to this system, the robot is therefore able to execute long-term tasks, because the user does not still have to take care of the energy consumption of the robot.

The functionality of the whole system is granted by the particular docking method chosen: the algorithm based on range lights gives robustness and reliability to the system, maintaining at the same time a great simplicity of the apparatus. Thanks to its plainness, the system does not require a specific instrumentation and is easily reproducible with many common materials: so this application can be realized at a very low cost, as is evident comparing our project with other existing docking-charging applications.

In spite of the low-cost, the performance of the system we built is very satisfying. After the theoretic studies that proved the high precision of the docking algorithm, we performed many tests on the real system realized. We tried to order the robot to dock into the station from many positions inside the area represented in figure 13 (a rectangle of 4m by 5m) and we gladly observed successful completions to these tasks, always letting the charging station correctly start the procedures for the batteries' recharging.

In figure 13, are represented the odometry results of three attempts at docking into the charging station (the box at the bottom), executed by the robot from the three positions labeled as *(a)*, *(b)* and *(c)*. The trajectories give the idea of the application of the docking algorithm previously studied.

We can eventually conclude that the tests performed gave satisfying results, proving that, in the right conditions, the robot is capable to dock into the station and the charging mechanism works correctly.

These conditions absolutely do not limit the functionality of the system, because they are satisfied by a sensible planning for the displacement of the docking station: it is clear that the station must be put in a place accessible

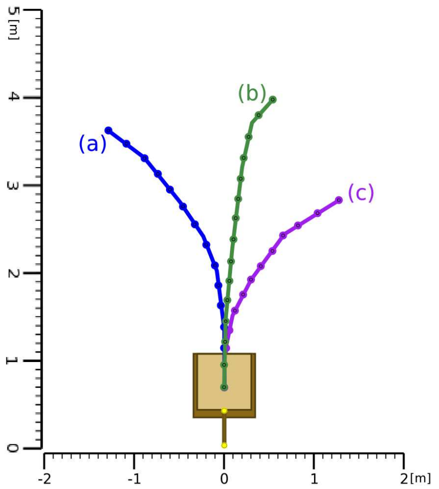

Fig. 13. Odometry traces measured during three docking tests.

by the robot, possibly without obstacles on the ground and far from bright lights that may deceive the robot in the identification of the docking station; it would not be difficult to find such a place in an indoor environment.

As for the perspectives of this project, the system we realized may be useful in a large number of mobile robotics' applications: we described a system dedicated to the autonomous batteries' recharging, but the plainness and the reliability of the docking algorithm should be exploited in many other projects, in which a robot need to dock to an object to do every kind of work of precision.

Finally, the system actually works well if the robot's starting position lets it see the docking station, that means that the robot is in the same room of the station, not too far (up to few meters) from it. Intuitively, the first extension to implement is an application to let the robot work inside a wide area, e. g. a business department, giving it the capability of coming back if necessary: this can be simply implemented by furnishing the robot with a map of the environment, or for instance by letting it remember the path covered.

As a further application, it would be interesting to extend the capabilities of our system by letting it interface with a distributed system based on the DCDT. The Device Community Development Toolkit (DCDT) is a project carried out by the Department of Electronics for Automation of the University of Brescia, that consists in a message oriented middleware to enable asynchronous exchange of data, event notification, persistence, quality of service and ease of development, to implement a communication layer among different devices, making the underlying media transparent to the developer [Cassinis et al., 2001].

If our system was able to interact in the DCDT, for instance it could be possible for the robot to notify to every other device in the community whether the robot was going to charge the batteries or if it was correctly executing the assigned tasks. Or, even, the robot could ask the community members for some information about the environment: those could be useful to come back to the docking station in case of need.

REFERENCES

[ActivMedia Robotics, 2003] ActivMedia Robotics (2003). `http://www.activrobots.com/ACCESSORIES/DXDock.html`.

[Austin and Kouzoubov, 2002] Austin, D. and Kouzoubov, K. (2002). Robust, long term navigtion of mobile robot. In *IARP Workshop on Technical Challenges for Dependable Robots in Human Environments*.

[Bowditch, 2002] Bowditch, N. (2002). The american practical navigator.

[Cassinis et al., 2001] Cassinis, R., Meriggi, P., Bonarini, A., and Matteucci, M. (2001). Device communities developement toolkit: an introduction. In *Eurobot*, pages 155–161, Lund, Sweden.

[Cassinis et al., 2003] Cassinis, R., Meriggi, P., and Panteghini, M. (2003). A very low cost distribuited localization and navigation system for mobile robot. In *1st European Conference on Mobile Robots (ECMR'03)*.

[Cassinis et al., 2005a] Cassinis, R., Tampalini, F., and Bartolini, P. (2005a). Wireless Network Issues for a Roaming Robot. Technical report, Università degli Studi di Brescia. `http://www.ing.unibs.it/~cassinis/docs/papers/05_010.pdf`.

[Cassinis et al., 2005b] Cassinis, R., Tampalini, F., and Fedrigotti, R. (2005b). Active Markers for Outdoor and Indoor Robot Localiziation. Technical report, Università degli Studi di Brescia. `http://www.ing.unibs.it/~cassinis/docs/papers/05_009.pdf`.

[Hada and Yuta, 1999] Hada, Y. and Yuta, S. (1999). Robust navigation and battery re-charging systems for long term activity of autonomous-mobile robot. In *The Ninth International Conference on Advanced Robotics*, Tokyo.

[Kouzoubov and Austin, 2002] Kouzoubov, K. and Austin, D. (2002). Autonomous recharging for mobile robotics.

[Maxim, 1997] Maxim, L. D. (1997). The nautical chart user's manual.

[Meriggi, 2002] Meriggi, P. (2002). *Processing and Communication Systems for Device Communities*. PhD thesis, Università degli Studi di Brescia, Italy.

[Oh et al., 2000] Oh, A., Zelinsky, A., and Taylor, K. (2000). Autonomous battery recharging for indoor mobile robots. In *Australian Conference on Robotics and Automation*.

[Silverman et al., 2002] Silverman, M. C., Nies, D., Jung, B., and Sukhatme, G. S. (2002). Staying alive: A docking station for autonomous robot recharging. In *Australian Conference on Robotics and Automation*, pages 1050–1055.

[Tampalini and Cassinis, 2005] Tampalini, F. and Cassinis, R. (2005). Fuzzy logic controller based on XML formatted files for behaviour–based mobile robots. Technical report, Università degli Studi di Brescia. `http://www.ing.unibs.it/~cassinis/docs/papers/05_011.pdf`.

[Walter, 1953] Walter, W. G. (1953). *The Living Brain*. W.W. Norton & Company, New York.