A Brief Overview of Robot Navigation and Localization Techniques

Stefano Galli
Dipartimento di Elettronica e Informazione - Politecnico di Milano

10th June 2004

Abstract

This paper thakes a brief insight in some interesting solutions direct to navigate and localize a single robot, or a swarm, in known and unknown environment.

1 Introduction

The problem of navigating and localizing a robot in both known and unknown environment has been studied in many research centers. The use of sensors to directly determine position by analizing wheels or legs motion (odometry) lacks of precision, because of well known problems such as wheel slippage or mechanical shocks coming from irregular terrain.

Global Positioning System (GPS) provides a quite accurate localization over Earth surface, but it fails to direct the robot in terrains where centimeter precision is required, and obviously does not apply for indoor environments. Moreover GPS is not a solution for extraterrestrial mission (i.e. Mars exploration).

Finally, the use of landmarks and pre-set trajectories limitates the range of use of a robot to well known environments, and even a slight change in the topological disposition of landmars means for the robot to lose its guidelines.

So it seems clear that some methods are required for the robot to adapt to new terrains or to build a map of a known environment. Here we want to take a look at some solutions suggested by various researchers.

2 Overview

We will start reviewing some solutions, starting from single robot and then considering multi robot approach.

2.1 Single Robot Navigation

There are several navigation and localization techniques that we will examine here. Each one uses different sensors and has different purposes.

2.1.1 Topological Simultaneous Localization and Mapping: Toward Exact Localization Without Explicit Localization

This study, proposed by Choset and Nagatani[4], presents a low-level control law to generate a topological map. *Generalized Voronoi Graph* (GVG) is used to encode topological informations.

GVG is a one-dimensional set of curves that captures the salient topology of the robot's environment. Since it is defined in terms of distance functions, GVG is practical to be used in a sensor-based implementation. Distance can be obtained by sonar or laser sensor disposed over the perimeter of the robot.

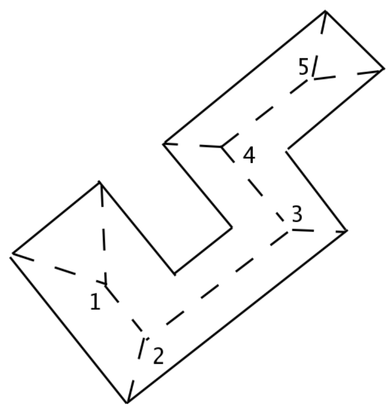

Figure 1: An example of a GVG

In the planar case, GVG edges are simply the set of points equidistant to two obstacles (see Figure 1). Using its sensors' line-of-sight readings, the robot incrementally builds the GVG: it simply moves away from the nearest obstacle, following a precise direction, until it is equidistant to two obstacles. By repeating this behaviour, the robot visits the whole environment - tipically a building - and completes the GVG, which can therefore be used to plan a path between any two locations.

Particular attention has to be posed in a procedure to determine if the robot has encountered a new meet point[1] or revisited an old one. Because of dead-reckoning errors it is not enough to compare current coordinates with the coordinates of all known meet points.

[1] A meet point is a point where GVG edges meet. When reaching a meet point the robot has to decide which branch of GVG emanating from the meet point has to be taken. In Figure 1 mett points are numbered from 1 to 5

2.1.2 Sensor Fusion for Localizing a Mobile Robot Outside of Buildings

Hassel and Hertzberg [6] worked on a way to merge data coming from different sensors to achieve a fine localization of a robot in outdoor environment. The robot was equipped with inertial sensors (encoder that register wheel movement, and a fiber-optical gyroscope) and vision sensors for pattern recognition.

Encoders were mounted on the passive wheels, so that - due to the high weight of the robot - it was impossible for both wheels to be affected by slippage. However, in case of slippage or bumps, measurement errors are corrected by data coming from the gyroscope, since it is not affected by mechanical shocks. The use of a GPS localization system in addition did not show any noticeable increase in positioning accuracy.

So in the first step of this algorithm, data coming from inertial sensors (encoders and gyroscope) are fed into a Kalman filter[2], which produces an estimated position.

This estimate is then refined by the second step. Using the vision sensors (i.e. laser scan) the robot identifies doors in the buildings around itself. Using a map of all known doors, and the relative positions of the doors coming from vision sensors, it is then trivial to pinpoint the absolute position of the robot, and use this informations for a position update in the Kalman filter.

Experimental results are very encouraging. The robot had to navigate a university campus from a starting point to a selected door, and enter it. Different paths were used: within 50 meters range, the robot never missed the door it had been given as a target. As the range increases, the number of missed doors also grows, reaching 5 over 20 for a range of 520 meters.

2.1.3 Acquisition and Propagation of Spatial Constraints Based on Qualitative Information

Ishida et al. [8] worked on a method for reconstructing qualitative positions of landmarks from qualitative information acquired by visual observation. Since this method requires a global point of view over the environment where the landmarks are located, it is appliable both on robot with omnidirectional vision sensors, and on a *distributed vision system* (DVS)[3]. The DVS approach is the one considered in this paper.

The first step of this method, is the acquisition of a qualitative spatial model by observing motion directions of a moving object form each landmark's point of view: when the projection of the moving object moves clockwise, in the

[2] The Kalman filter is a set of mathematical equations that provides an efficient computational (recursive) means to estimate the state of a process, in a way that minimizes the mean of the squared error. The filter is very powerful in several aspects: it supports estimations of past, present, and even future states, and it can do so even when the precise nature of the modeled system is unknown. [3]

[3] DVS consists in multiple vision sensors embedded in an environment, in this case every landmark is equipped with omnidirectional vision sensors.

omnidirectional sight of a vision sensor, the motion is qulitatively represented as "right", and if it moves counterclockwise, it is represented as "left".

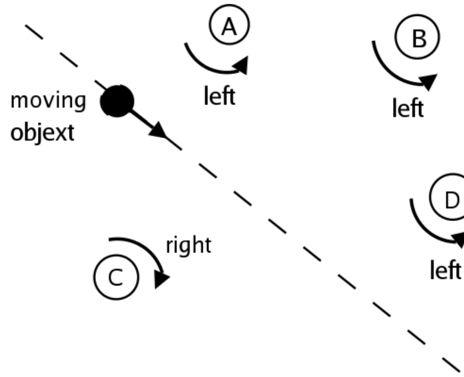

Figure 2: Observation for acquiring qualitative positions

With the observed motion directions, points are classified into a *spatially classified pair* (SCP), which consists of a pair of point sets labeled "left" and "right"; for example, the SCP consistent with Figure 2 is "{ABD},{C}".

Next step in the algorithm is the acquisition of *thee point constraints* (3PCs) from the SCP. To determine the qualitative position of the points this method considers the relative position of the target with respect to a triangle where the three landmarks are at the vertices. So, considering a SCP coming from sensor readings, it is possible to acquire a 3PCs, and knowing the region of space where the target is *not* present.

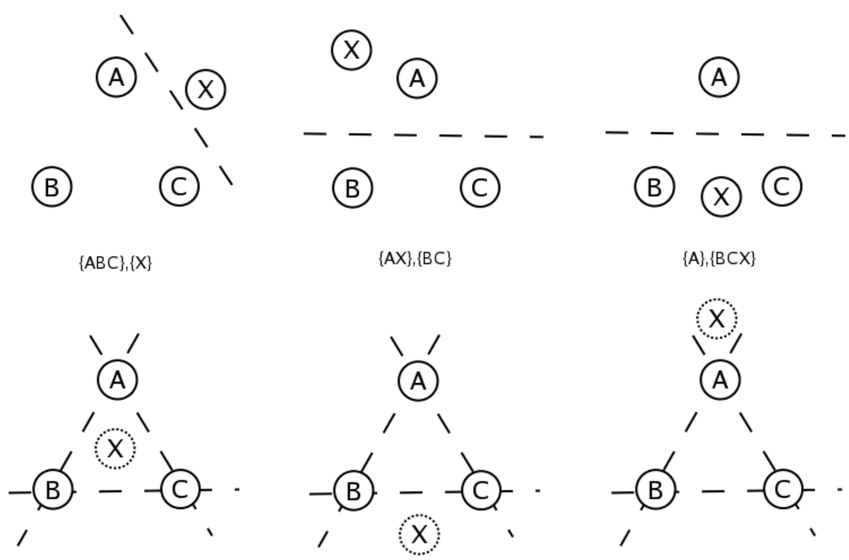

Figure 3: Three point constraints.

As seen in Figure 3 there are several possible configuration, each one eliminates a region of space. When, iterating observation, enough SPC and 3PCs are acquired, it is possible to uniquely determine the position of the target.

The last two steps of the algorithm classify the point into new SPCs based on the 3PCs, and acquire new 3PCs (this step is called "constraint propagation"), and then transform the 3PCs into the qualitative spatial model.

2.1.4 Exploiting the physics: towards Doppler-based navigation with a bat-inspired mobile robot

Carmena and Hallam [5] showed how to use Doppler-shifts for ultrasound-based navigation in mobile robots.

Since previous researches in ultrasonic sensor only considered extracting informations from Time of Flight (ToF) for the first echo only, this study took inspiration from biosonar. In particular the flight navigation system used by a specific kind of bats, classificated as CF-FM (constant frequency - frequency modulation), has been examined. The way these bats interpretate natural sensor readings is quite similar to the way mobile robots navigate in laboratories using their ultrasonic sensors.

To explore these possibilities a mobile robot (named RoBat) was built and equipped with a biomimetic sonarhead and a signal processing package whose operations, performed on the received echoes, are based upon a filterbank model of the processing performed by mammalian cochlea. RoBat was designed to use Doppler-shift informations to perform two particular tasks: convoy navigation and obstacle avoidance.

Doppler-shift gives informations about the presence and the movement of a reflecting surface (a wall, or another robot), allowing to know if it is closing or getting away with respect to the couple emitter/receiver on board the robot. After defining *max_doppler* as the maximum Doppler-shift perceable by the robot for a given velocity assuming a static reflector at 0° bearing angle, it is possible to identify three situations:

- Doppler ≤ 0: there is no reflector in the way, of ther is a moving reflector whose relative velocity with respect to the robot is zero or negative. In this case the reflector is getting away, so the robot can navigate safely within its perceptual range.
- $0 <$ Doppler $< max\_doppler$: there is either a static reflector in the way, or a moving reflector with a positive relative velocity with respect to the robot, but with a bearing angle sufficient to avoid a collision. In this case there's an object closing to the robot, but it is not in an intercept course.
- Doppler $\geq max\_doppler$: there is a moving reflector in the way with a positive relative velocity with respect to the robot, and its bearing angle

is 0 or close to 0. In this case something is closing to the robot, and it is directly in front of it, so the robot should change its path immediately to avoid a collision.

Basing on these parameters, a simple behaviour controller can be programmed to avoid an obstacle.

It is the belief of the authors that Doppler-based sensors are a good resource that is not yet been exploited in commercial ultrasonic sensors. Since Doppler-shift is proportional to the cosine of the bearing angle between the robot and the reflector, and given the dynamic nature of the world, where things move and change their position, Doppler sensors are able to provide different kind of data: Time of Flight, bearing angle and approaching/departing information.

2.1.5 A Unified Solution to Coverage and Search in Explored and Unexplored Terrains Using Indirect Control

Pirzadeh and Snyder [1] worked on an indirect control solution for autonomous robot navigation in both known and unknown environment. Indirect control means that the algorithm does not determine a destination for the robot and the calculates the path to reach it; instead the algorithm controls the robot path by placing costs along paths the robot may take.

This study considers two kind of problems: coverage (the robot passes over all parts of the taerrain that are free of obstacles) and the search problem (the robot seeks for a specified target in the terrain). Both these problems are considered for a known environment and an unknown environment.

Usually path planning algorithm have a known origin-destination pair to work on. The problems faced by this study do not take this assumption: coverage problem has the whole terrain as a destination; search problem does not have a known destination, and may even not have any.

The key idea of this solution is to divide the terrain into cells, which are units of area equal to the dimensions of the robot base. If the terrain is known then the robot has a map of all obstructed cells (cells where an obstacle is present); and if it is not, then the map can be built quite simply by using external sensors (such as proximity or distance sensors). So, having a map it is trivial to navigate in the terrain, since the problem is now discrete.

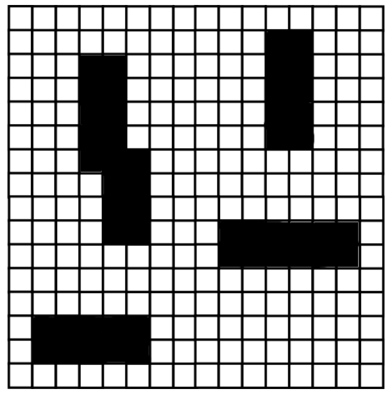

Figure 4: Discrete map

The base algorithm for the covering problem of a known terrain:

- step 1) increment the current cell cost by a constant (this prevents the robot from getting caught in endless loops, and minimizes the number of times the robot covers the same cell)
- step 2) interrogate the neighboring cells to determine the least costrly direction
- step 3) choose the direction (in case of tie, the decision is made basing on up, down, right, left priority)

As stated before the control here is indirect: the choice of the direction (step 2) is not based on the knowledge of the ultimate goal of the problem.

This base algorithm can be refined with two extensions:

- "Looking ahead in time", which associates a high cost to cells that need not to be traversed anymore, making them appear as obstacles, with the condition that cells that provide the only traverseable path between two free cells are not to be penalized
- "Looking ahead in space", which considers wether all the cells in a certain direction have been covered, and if so, penalizes that direction to prevent the robot from determining it as a possible derection

For the search problem it is simple to modify this algorithm just adding a step 0 that looks in all directions for the target, marking all cells in the direction where it is not found as covered, and preceeding towards the position of the target. Furthermore, in the case of an unknown terrain, it is necessary to add a preventive step which, using sensor readings, incrementally creates the map of free/obstructed cells.

2.2 Multi-robot navigation

Following we will see two studies covering multi-robot navigation problems.

2.2.1 A Multi-robot Approach to Stealty Navigation in the Presence of an Observer

Mataric et al. [2] proposed a method for multiple robot low-visibility navigation in presence of an observer.

It is well known that a multi-robot approach can give benefits as well as give problems. In the case of stealth navigation, a robot swarm could be difficult to manage, because of the larger profile exposed to the observer. This study, however, deals with robots sequentially navigating in the environment, so that trajectory calculated by one robot is then refined by the following. We take as assumption that the observer has omnidirectional sensing, and the environment consists of objects that can occlude the robots from the observer's sensors. Robots initially do not have a map of the environment, but the location of the goal and the observer are known.

One at the time, robots sequentially traverse the environment, building (or refining) an occupancy grid modeled by potential fields. Navigation waypoints are extracted from the grid, and used in planning a low-visibility path. These informations are passed from one robot to another, so that the successor can use them to make decisions about waypoint selection, and build a new, more precise, occupancy grid. By sharing informations each robot follows a lower visibility path than its predecessor.

Potential fields used to model the grid in this case, as usual for obstacle avoidance, contain informations about distance from the robot, the observer and the goal. To enable the robots planning a low visibility trajectory an additional field is needed, containing informations about occluded areas behind objects (with respect to the observer's position). Combining the fields robots can extract lowest-valued regions, representing obstacle-free and low-visibility regions. As a result, navigation waypoints are calculated using these data.

Further policies are considered in extracting waypoints, for example a local repellor potential field is added to prevent nearby waypoint selection.

Experimental results, coming both from simulated and real experiments, show that each traverse improves the previous in efficiency and decreased visibility, proving the real advantage of a multi-robot approach.

2.2.2 Strategis for navigation of robot swarms to be used in landmines detection

Cassinis et al. [7] studied the possibility of introducting robot swarms in demining operations. Aside from the benefit of warrant the safety of human beings during dangerous landmines detection operations, the use of multiple inexpensive robots minimizes damage due to unexpected exploding mines allowing the mission to be carried on anyway by the remaining units. To achieve these goals robots must be able to cooperate in terms of avoiding intereference with each other, providing complementary informations via different sensors, sharing the workload and dinamically reassigning tasks in case of robot failures.

One of the main advantage of using multiple small robots equipped with different sets of sensor is the possibility of merge data to locate mines that are undetectable by a single sensor, without dealing with complex and expensive multi-sensor robots. Given the drawbacks of traditional centralized control (such as high computational and communication complexity) a distributed control is more suitable, since each robot decides for itself by observing the environment and applying pre-defined control laws.

In order to control collision-free movement and contemporarily achieve a specific goal, this study examines a vectorial movement strategy. So four vectors have been identified, each one accomplishes one of the following goals:

1. avoiding obstacles, this vector can suppress all other vectors for the time necessary to move past the obstacle
2. pointing to the goal
3. maintaining position in a specific formation
4. maintaining robot direction

Using these vectors in the right combination, six strategies can be defined

- **random movement**
- **relay clustering**: robot initally move randomly, when one finds a mine it signal its position so that the others can reach it; cascade forwaring of the signal is used to ensure full coverage
- **flocking**: robots head to the same direction keeping cohesion but maintaining a certain distance between one another
- **swarming**: robots are attracted one another as the distance between them increases, so they will move in the same direction
- **formation maintenance**: robots can move in coordination, each one belonging to a specified team; the position of each robot in the formation is fixed relatively to the team centroid
- **comb movement**: similar to formation maintenance, but formation changes from line to column, passing from one goal to another one.

Experimental results proved that coordinate strategies find mines at a constant rate, while uncoordinated ones result to be more efficient if many mines are still undetected. It has also been noticed that, after a certain number, adding more robots to the swarm dows not yeld a significant reduction of required time.

References

[1] Amir Pirzadeh and Wesley Snyder. A Unified Solution to Coverage and Search in Explored and Unexplored Terrains Using Indirect Control.

[2] Ashley D. Tews, Gaurav S. Sukhtame, and Kaja J. Mataric. A Multi-robot Approach to Stealthy Navigation in the Presence of an Observer. *IEEE International Conference of Robotics and Automations*, 26 May 2004.

[3] Greg Welch and Gary Bishop. An Introduction to the Kalman Filter. *http://www.cs.unc.edu/ welch/kalman/kalman_filter/kalman.html.*

[4] Havit Choset and Keiji Nagatani. Topological Simultaneous Localization and Mapping: Toward Exact Localization Without Explicit Localization. *IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION*, 17:125–136, 2 April 2001.

[5] Jose M. Carmena and John C.T. Hallam. Exploiting the physics: towards Doppler-based navigation with a bat-inspired mobile robot.

[6] Maik Hassel and Joachim Hertzberg. Sensor Fusion for Localizing a Mobile Robot Outside of Buildings.

[7] R. Cassinis, G. Bianco, A. Cavagnini, and P. Ransenigo. Strategis for navigation of robot swarms to be used in landmines detection.

[8] Takushi Sogo, Hiroshi Ishiguro, and Toru Ishida. Acquisition and Propagation of Spatial Constraints Based on Qualitative Information. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 23(3):268–277, March 2001.