Obstacle Avoidance 101

Dariush Hasanpoor

Isfahan University Of Technology

Feb. 2016

Outline

What is a Obstacle Avoidance?

According to Wikipedia:

In robotics, obstacle avoidance is the task of satisfying some control objective subject to non-intersection or non-collision position constraints.

The Bugs!

- Many planning algorithms assume global knowledge.
- Bug algorithms assume only local knowledge of the environment and a global goal.
- The Bug algorithms[1, 2], are simple ways to overcome unexpected obstacles in the robot motion from a start point s, to a goal point g.
- The goal of the algorithms is to generate a collision free path from the s to g with the underlying principle based on *contouring the detected obstacles*.
- Bug 1 and Bug 2 assume essentially tactile sensing.

The Bugs!

- The general procedure in bugs:
  - Move towards the goal, unless an obstacle is encountered.
  - Circumnavigate the obstacle until motion toward the goal is again allowable.
- Robot is assumed to be a point with perfect positioning.
- Robot has a contact sensor which detects the obstacle boundary if it touches it.
- The robot can measure the distance $d(x, y)$ between any two points.
- The workspace is bounded.

Bug 1

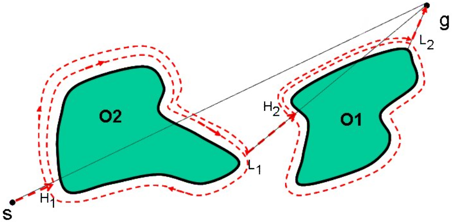


Figure 1 : Representing a situation in bug1 with two obstacles where H_1 and H_2 are the hit points and L_1 and L_2 the leave points.

Bug 2

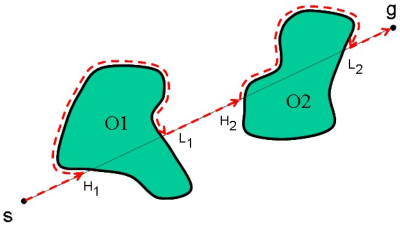
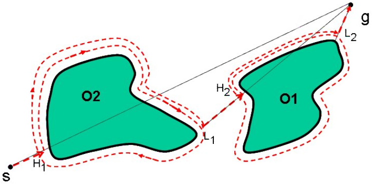


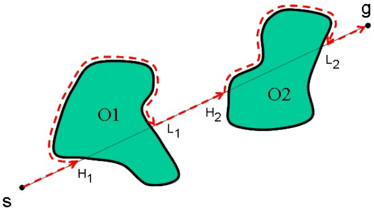
Figure 2 : Representing a situation in bug2 with two obstacles where H_1 and H_2 are the hit points and L_1 and L_2 the leave points.

Pros and Cons

Bug1 vs. Bug2



(a) The bug1 behaviour



(b) The bug2 behaviour

Figure 3 : The difference between different version of bug algorithm

Pros and Cons

Bug1 vs. Bug2

- Bug1:
  - Is an exhaustive search algorithm; it looks at all choices before commiting. therefor it's very inefficient but guarantees that the robot will reach any reachable goal.
- Bug2:
  - Is a greedy algorithm; it takes the first thing that looks better.
- In many cases, bug2 has a shorter travel time than Bug 1 algorithm and is more efficient specially in open spaces.
  - The bug1 has a more predictable performance overall.

Pros and Cons

Bug1 vs. Bug2

- None of these algorithms take robot kinematics into account which is a severe limitation, specially in the case of *non-holonomic* robots.
  - A robot is *holonomic* if all the constraints that it is subjected to are integrable into positional constraints.

The Potential Field Methods

- The robot is consider as a particle that moves immersed in a potential field generated by the goal and by the obstacles present in the environment[3].
- The goal generates an attractive potential while each obstacle generates a repulsive potential.
- The robot immersed in the potential filed is subject to the action of a force that drives it to the goal.

The Potential Field Methods

A point of view in PF

- A potential field can be viewed as an energy field and so its gradient, at each point, is a force.
- In this analogy, the robot is a positive charge, the goal is a negative charge and the obstacles are sets of positive charges.

The Potential Field Methods

- Let q represent the position of the robot, considered as a particle moving in a n-dimensional space R^n.
- The artificial potential field where the robot moves is a scalar function $U(q) : R^n \to R$:

$$\begin{aligned} U(q) &= U_{\text{att}}(q) + U_{\text{rep}}(q) \\ &= U_{\text{att}}(q) + \sum_i U_{\text{rep}_i}(q) \end{aligned} \tag{1}$$

The Potential Field Methods

- The attractive potential is chosen to be zero at the goal and to increase as the robot is far away from the goal.
- The repulsive potential, associated with each obstacle, is very high (infinity) in the close vicinity of the obstacles and decreases when the distance to the obstacle increases.

The Potential Field Methods

- The force that drives the robot is the negative gradient of the artificial potential, i.e.

$$F(q) = F_{\text{att}}(q) + F_{\text{rep}}(q) = -\nabla U_{\text{att}}(q) - \nabla U_{\text{rep}}(q) \tag{2}$$

  - This force can be considered as the velocity vector that drives the point robot.

The Potential Field Methods

The Attractive Potential

- The usual choice for the attractive potential is the standard parabolic that grows quadratically with the distance to the goal:

$$U_{\text{att}}(q) = \frac{1}{2} k_{\text{att}} d^2_{\text{goal}}(q) \tag{3}$$

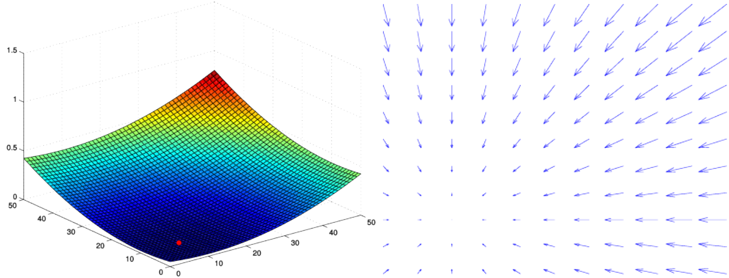
  - Where $d_{\text{goal}}(q) = \left\| q - q_{\text{goal}} \right\|$ is the Euclidean distance of the robot (considered at q), to the goal, at q_{goal} and k_{att} is a scaling factor.

$$\nabla U_{\text{att}} = k_{\text{att}}(q - q_{\text{goal}}) \tag{4}$$

$$F_{\text{att}} = -\nabla U_{\text{att}} = -k_{\text{att}}(q - q_{\text{goal}}) \tag{5}$$

The Potential Field Methods

The Attractive Potential



(a) Attractive Potential (b) Attractive Force to the goal

Figure 3 : The Potential Field Methods – the goal at (10, 10) is represented by a mark.

The Repulsive Potential

- The repulsive potential keeps the robot away from the obstacles.
- The repulsive potential is stronger when the robot is closer to the obstacle and has a decreasing influence when the robot is far away.
- Given the linear nature of the problem:

$$U_{\text{rep}} = \sum_i U_{\text{rep}_i}(q) \tag{6}$$

$$U_{\text{rep}_i} = \begin{cases} \frac{1}{2} k_{\text{obst}_i} \left(\frac{1}{d_{\text{obst}_i}(q)} - \frac{1}{d_0} \right)^2 & \text{if } d_{\text{obst}_i}(q) < d_0 \\ 0 & \text{o.w.} \end{cases} \tag{7}$$

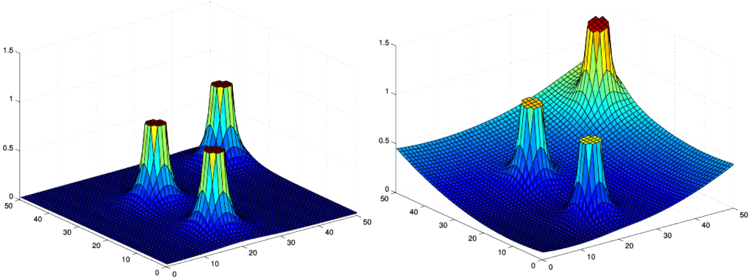
where:

- $d_{\text{obst}_i}(q)$ is the minimal distance from q to the obstacle i.
- k_{obst_i} is a scaling constant.
- d_0 is the obstacle influence threshold.

The Repulsive Potential

- The negative of the gradient of the repulsive potential, $F_{\text{rep}_i}(q) = -\nabla U_{\text{rep}_i}(q)$, is given by,

$$F_{\text{rep}_i} = \begin{cases} k_{\text{obst}_i} \left(\frac{1}{d_{\text{obst}_i}(q)} - \frac{1}{d_0} \right) \cdot \frac{q - q_{\text{obst}_i}}{d^3_{\text{obst}_i}(q)} & \text{if } d_{\text{obst}_i}(q) < d_0 \\ 0 & \text{o.w.} \end{cases} \tag{8}$$



(a) Repulsive Potential

(b) Attractive+Respulsive Potential

Figure 4 : The Potential Field Methods

The Potential Field Methods

Pros and Cons

- The potential field approach herein presented is a simple path planning.
- For a static and completely known environment, the potential can be evaluated off-line providing the velocity profile to be applied to a point robot moving in the energy field from a starting point to a goal.
- Also, the technique can be applied in an on-line version that accommodates an obstacle avoidance component.
- The robot using this approach can stuck in a local minimum:

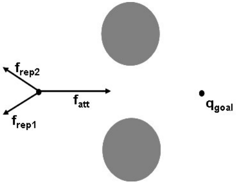


Figure 5 : Local minimum of the total potential due to environment symmetry

The Potential Field Methods

Other Versions

- In some extended version of the potential field based methods, the repulsive potential is changed in some ways[3, 4].

  ▷ The repulsive force due to an obstacle is not only considered as a function of the distance to the obstacle but also of the orientation of the robot relative to the obstacle. This is a reasonable change since the urgency of avoiding an obstacle parallel to the robot motion is clearly less than the one that arises when the robot moves directly facing the obstacle.

  ▷ The second extension of the repulsive potential does not consider the obstacles that will not closely affect the robot velocity. For example, it is irrelevant to consider the repulsive force generated by an obstacle in the back of the robot, when it is moving forward.

The Virtual Force Field

- The Virtual Force Field(VFF) method is one of earlier real-time obstacle avoidance method for fast-running vehicles[5].
- Advantage of using VFF instead of previous methods:
  - It's fast.
  - It's continuous.
  - Smooth motion of the controlled vehicle among unexpected obstacles.
  - Does not require the vehicle to stop in front of obstacles.

The Virtual Force Field

- The individual components of the VFF method are presented below.
  - ▷ The VFF method uses a two-dimensional Cartesian histogram grid C for obstacle representation.
    - Each cell (i, j) in the histogram grid holds a certainty value, $c_{i,j}$.
    - The histogram grid differs from the certainty grid in the way it is built and updated
  - ▷ Using the grid's value the VFF applies the PF idea to the histogram grid, so the probabilistic sensor information can be used efficiently to control the vehicle.

The Virtual Force Field

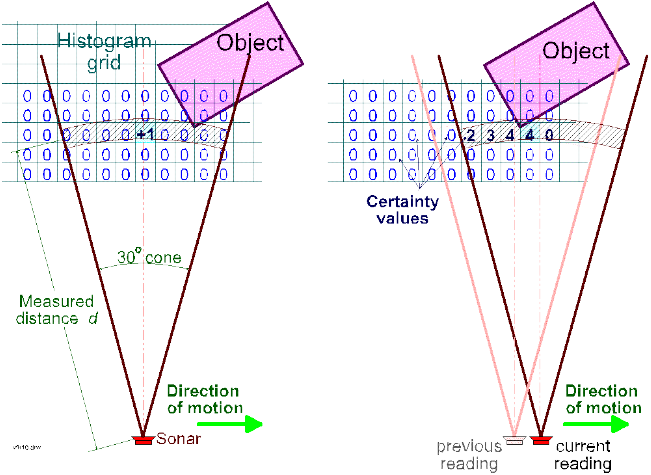
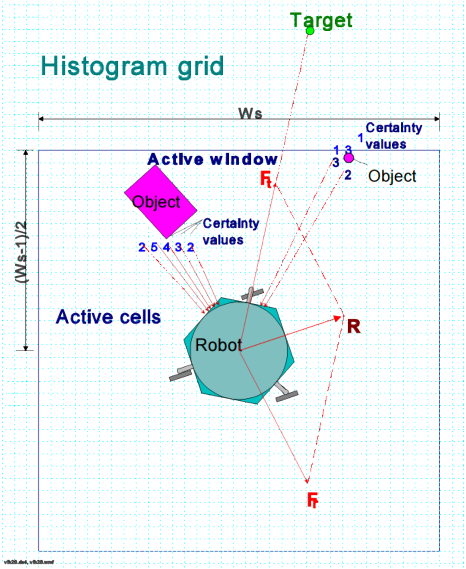


Figure 6 : Construction of the 2D Histogram grid map.

The Virtual Force Field



The Virtual Force Field

- As the robot moves around, range readings are taken and projected into the Certainty Grid, as explained.
- Simultaneously, the algorithm scans a small square window of the grid.
- The size of the window is 33×33 cells(i.e., $3.30.30$m) and its location is such that the robot is always at its center.
- Each occupied cell inside the window applies a repulsive force to the robot, "pushing" the robot away from the cell.

The Virtual Force Field

- The magnitude of this force is proportional to the cell contents, $C(i,j)$, and is inversely proportional to the square of the distance between the cell and the robot:

$$F(i,j) = \frac{F_{cr} C(i,j)}{d^2(i,j)} \left[\frac{x_t - x_0}{d(i,j)} \hat{x} + \frac{y_t - y_0}{d(i,j)} \hat{y} \right] \tag{9}$$

- where:
  - F_{cr} Force constant (repelling)
  - $d(i,j)$ Distance between cell (i,j) and the robot
  - Certainty level of cell (i,j)
  - Robot's present coordinates
  - Coordinates of cell (i,j)
- The resultant repulsive force, F_r, is the vectorial sum of the individual forces from all cells:

$$F_r = \sum_{i,j} F(i,j)$$

The Virtual Force Field

- At any time during the motion, a constant-magnitude attracting force, F_t, pulls the robot toward the target.
- F_t is generated by the target point t, whose coordinates are known to the robot. The target-attracting force F_t is given by:

$$F_t = F_{cr}\left[\frac{x_t - x_0}{d(t)}\hat{x} + \frac{y_t - y_0}{d(t)}\hat{y}\right] \tag{10}$$

- The vectorial sum of all forces, repulsive from occupied cells and attractive from the target position, produces a resultant force vector R:

$$R = F_t + F_r \tag{11}$$

The Virtual Force Field

Pros and Cons

- Some of VFF limitation and disadvantages are[6]:
  - Force-based obstacle-avoidance methods do not allow the robot to pass through narrow passages.
  - Instability of motion when traveling within narrow corridors.
- So *The Vector Field Histogram*(VFH) was introduced to improve the VFF method.
  - VFH control results in smooth motion of the controlled vehicle among densely cluttered and unexpected obstacles.
  - A VFH controlled vehicle can easily enter narrow passages and can travel in narrow corridors at high speeds and without oscillations.

The Vector Field Histogram

- The VFH method uses a two-dimensional Cartesian Histogram Grid for the representation of obstacles(like VFF method)[6].
- The VFH method employs a two-stage data reduction technique, in which three levels of data representation can be distinguished.
  - level 1) The highest level holds the detailed description of the robot's environment.
  - level 2) At the intermediate level, a Polar Histogram H is constructed around the robot's momentary center.
  - level 3) The lowest level of data representation is the output of the VFH algorithm.

The Vector Field Histogram

- Level 2) At the intermediate level, a Polar Histogram H is constructed around the robot's momentary center:
  - H comprises n angular sectors of width α
    - α may be chosen arbitrarily but must be such that $n = \frac{360}{\alpha} \in \mathbb{N}^{+}$
  - A transformation maps C^* into H resulting in each sector k holding a value h_k which represents the polar obstacle density in the direction k.

The Vector Field Histogram

- Level 3) The lowest level of data representation is the output of the VFH algorithm:
  - The reference values for the drive and steer controllers of the vehicle.

The Vector Field Histogram

Stage 1/2 of data reduction

$$\beta_{i,j} = tg^{-1}\left(\frac{y_j - y_0}{x_j - x_0}\right) \qquad (12)$$

$$m_{i,j} = (C^*)^2(a - bd_{i,j}) \qquad (13)$$

- Where:
  - a, b Positive constants.
  - $d_{i,j}$ Distance between active cell (i,j) and the VCP.
  - $c^*_{i,j}$ Certainty value of active cell (i,j).
  - $m_{i,j}$ Magnitude of the obstacle vector at cell (i,j).
  - x_0, y_0 Present coordinates of the VCP.
  - x_i, y_j Coordinates of active cell (i,j).
  - $\beta_{i,j}$ Direction from active cell (i,j) to the VCP.
- a, b are choosen such that $a - bd_{\max} = 0$ where $d_{\max} = \sqrt{2}(w_s - 1)/2$ is the distance between the farthest *active cell* and the VCP.

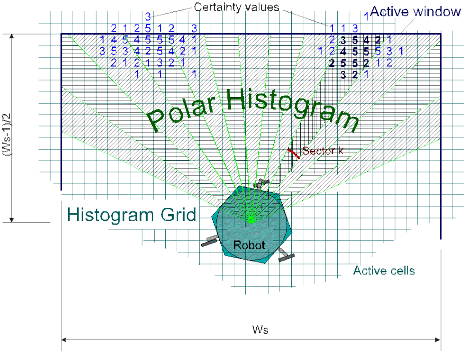
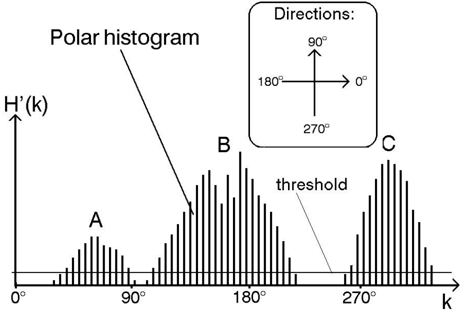


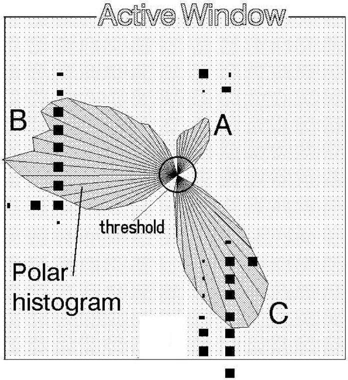
Figure 6 : Mapping active cells into sectors of the Polar Histogram H.

The Vector Field Histogram

Stage 2/2 of data reduction



(a) 1D polar histogram of obstacle occupancy around the robot.



(b) Polar histogram shown in polar form overlapped with C^*.

The Vector Field Histogram

Stage 2/2 of data reduction

- From the set of wide valleys, VHF chooses the one that minimizes a cost function that accounts for:
  - The alignment of the robot to the target,
  - The difference between the robot current direction and the goal direction,
  - The difference between the robot previously selected direction and the new robot direction.

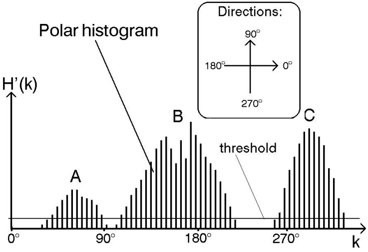


Figure 7 : 1D polar histogram of obstacle occupancy around the robot.

The Vector Field Histogram

Pros and Cons

- The Vector Field Histogram overcomes some of the limitations exhibited by the potential field methods.
  - The influence of bad sensor measurements is minimized because sensorial data is averaged out onto an histogram grid that is further processed.
  - Instability in travelling down a corridor, present when using the potential field method, is eliminated.
  - In the VFH there is no repulsive nor attractive forces and thus the robot cannot be trapped in a local minima.

Conclusion

- In this presentation we have reviewed the following obstacle avoidance algorithms:
  - Bug Algorithms
  - The Potential Field Methods
  - The Virtual Force Field
  - The Vector Field Histogram

Thank You!

References

- V. Lumelsky and T. Skewis, "Incorporating range sensing in the robot navigation function," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 20, no. 5, pp. 1058–1069, 1990.
- V. J. Lumelsky and A. A. Stepanov, "Path-planning strategies for a point mobile automaton moving amidst unknown obstacles of arbitrary shape," in *Autonomous robot vehicles*. Springer, 1990, pp. 363–390.
- O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *The international journal of robotics research*, vol. 5, no. 1, pp. 90–98, 1986.
- M. Khatib and R. Chatila, "An extended potential field approach for mobile robot sensor-based motions," in *Proc. International Conference on Intelligent Autonomous Systems (IAS4)*, 1995.
- J. Borenstein and Y. Koren, "Real-time obstacle avoidance for fast mobile robots," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 19, no. 5, pp. 1179–1187, 1989.
- ——, "The vector field histogram-fast obstacle avoidance for mobile robots," *Robotics and Automation, IEEE Transactions on*, vol. 7, no. 3, pp. 278–288, 1991.
- ——, "Real-time obstacle avoidance for fast mobile robots in cluttered environments," in *Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on*. IEEE, 1990, pp. 572–577.