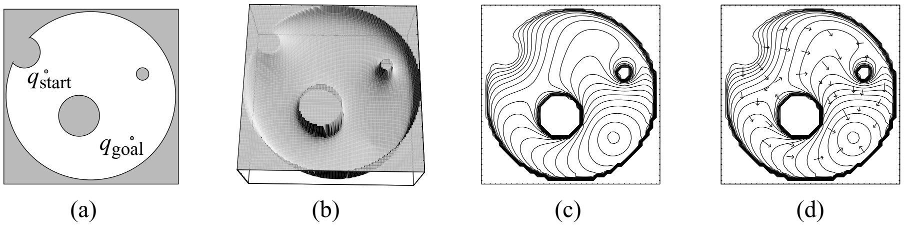


Figure 4.3 (a) A configuration space with three circular obstacles bounded by a circle. (b) Potential function energy surface. (c) Contour plot for energy surface. (d) Gradient vectors for potential function.

There are many potential functions other than the attractive/repulsive potential. Many of these potential functions are efficient to compute and can be computed online [234]. Unfortunately, they all suffer one problem—the existence of local minima not corresponding to the goal. This problem means that potential functions may lead the robot to a point which is not the goal; in other words, many potential functions do not lead to complete path planners. Two classes of approaches address this problem: the first class augments the potential field with a search-based planner, and the second defines a potential function with one local minimum, called a *navigation function* [239]. Although complete (or resolution complete), both methods require full knowledge of the configuration space prior to the planning event.

Finally, unless otherwise stated, the algorithms presented in this chapter apply to spaces of arbitrary dimension, even though the figures are drawn in two dimensions. Also, we include some discussion of implementation on a mobile base operating in the plane (i.e., a point in a two-dimensional Euclidean configuration space).

4.1 Additive Attractive/Repulsive Potential

The simplest potential function in $\mathcal{Q}_{\text{free}}$ is the attractive/repulsive potential. The intuition behind the attractive/repulsive potential is straightforward: the goal attracts the robot while the obstacles repel it. The sum of these effects draws the robot to the goal while deflecting it from obstacles. The potential function can be constructed as the sum of attractive and repulsive potentials

$$U(q) = U_{\text{att}}(q) + U_{\text{rep}}(q).$$

The Attractive Potential

There are several criteria that the potential field U_{att} should satisfy. First, U_{att} should be monotonically increasing with distance from q_{goal}. The simplest choice is the *conic potential,* measuring a scaled distance to the goal, i.e., $U(q) = \zeta d(q, q_{\text{goal}})$. The ζ is a parameter used to scale the effect of the attractive potential. The attractive gradient is $\nabla U(q) = \frac{\zeta}{d(q, q_{\text{goal}})}(q - q_{\text{goal}})$. The gradient vector points away from the goal with magnitude ζ at all points of the configuration space except the goal, where it is undefined. Starting from any point other than the goal, by following the negated gradient, a path is traced toward the goal.

When numerically implementing this method, gradient descent may have "chattering" problems since there is a discontinuity in the attractive gradient at the origin. For this reason, we would prefer a potential function that is continuously differentiable, such that the magnitude of the attractive gradient decreases as the robot approaches q_{goal}. The simplest such potential function is one that grows quadratically with the distance to q_{goal}, e.g.,

$$U_{\text{att}}(q) = \frac{1}{2}\zeta d^2(q, q_{\text{goal}}),$$

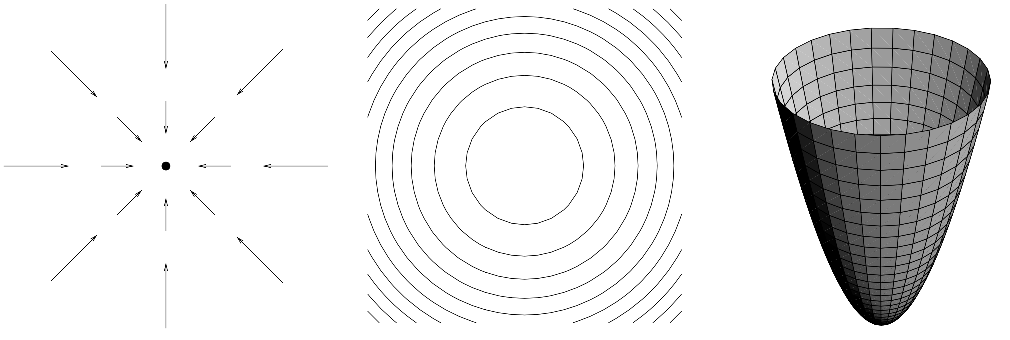
with the gradient

$$\begin{aligned} \nabla U_{\text{att}}(q) &= \nabla \left(\frac{1}{2}\zeta d^2(q, q_{\text{goal}})\right), \\ &= \frac{1}{2}\zeta \nabla d^2(q, q_{\text{goal}}), \\ &= \zeta (q - q_{\text{goal}}), \end{aligned} \tag{4.1}$$

which is a vector based at q, points away from q_{goal}, and has a magnitude proportional to the distance from q to q_{goal}. The farther away q is from q_{goal}, the bigger the magnitude of the vector. In other words, when the robot is far away from the goal, the robot quickly approaches it; when the robot is close to the goal, the robot slowly approaches it. This feature is useful for mobile robots because it reduces "overshoot" of the goal (resulting from step quantization).

In figure 4.4(a), the goal is in the center and the gradient vectors for various points are drawn. Figure 4.4(b) contains a contour plot for U_{att}; each solid circle corresponds to a set of points q where $U_{\text{att}}(q)$ is constant. Finally, figure 4.4(c) plots the graph of the attractive potential.

Note that while the gradient $\nabla U_{\text{att}}(q)$ converges linearly to zero as q approaches q_{goal} (which is a desirable property), it grows without bound as q moves away from q_{goal}. If q_{start} is far from q_{goal}, this may produce a desired velocity that is too large. For this reason, we may choose to combine the quadratic and conic potentials so that the



(a) (b) (c)

Figure 4.4 (a) Attractive gradient vector field. (b) Attractive potential isocontours. (c) Graph of the attractive potential.

conic potential attracts the robot when it is very distant from q_{goal} and the quadratic potential attracts the robot when it is near q_{goal}. Of course it is necessary that the gradient be defined at the boundary between the conic and quadratic portions. Such a field can be defined by

$$U_{\text{att}}(q) = \begin{cases} \frac{1}{2}\zeta d^2(q, q_{\text{goal}}), & d(q, q_{\text{goal}}) \leq d^*_{\text{goal}}, \\ d^*_{\text{goal}}\zeta d(q, q_{\text{goal}}) - \frac{1}{2}\zeta (d^*_{\text{goal}})^2, & d(q, q_{\text{goal}}) > d^*_{\text{goal}}. \end{cases} \tag{4.2}$$

and in this case we have

$$\nabla U_{\text{att}}(q) = \begin{cases} \zeta(q - q_{\text{goal}}), & d(q, q_{\text{goal}}) \leq d^*_{\text{goal}}, \\ \frac{d^*_{\text{goal}}\zeta(q - q_{\text{goal}})}{d(q, q_{\text{goal}})}, & d(q, q_{\text{goal}}) > d^*_{\text{goal}}, \end{cases} \tag{4.3}$$

where d^*_{goal} is the threshold distance from the goal where the planner switches between conic and quadratic potentials. The gradient is well defined at the boundary of the two fields since at the boundary where $d(q, q_{\text{goal}}) = d^*_{\text{goal}}$, the gradient of the quadratic potential is equal to the gradient of the conic potential, $\nabla U_{\text{att}}(q) = \zeta(q - q_{\text{goal}})$.

The Repulsive Potential

A repulsive potential keeps the robot away from an obstacle. The strength of the repulsive force depends upon the robot's proximity to the an obstacle. The closer the robot is to an obstacle, the stronger the repulsive force should be. Therefore, the

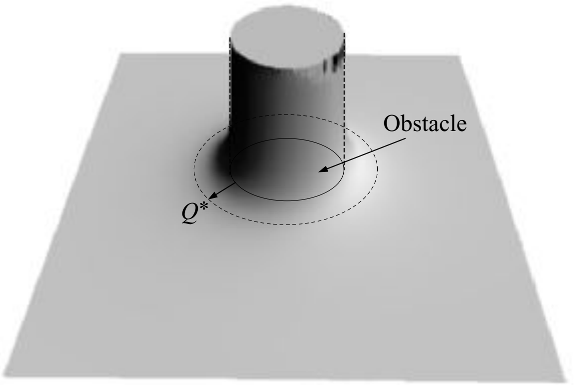


Figure 4.5 The repulsive gradient operates only in a domain near the obstacle.

repulsive potential is usually defined in terms of distance to the closest obstacle $D(q)$, i.e.,

$$U_{\text{rep}}(q) = \begin{cases} \frac{1}{2}\eta\left(\frac{1}{D(q)} - \frac{1}{Q^*}\right)^2, & D(q) \le Q^*, \\ 0, & D(q) > Q^*, \end{cases} \tag{4.4}$$

whose gradient is

$$\nabla U_{\text{rep}}(q) = \begin{cases} \eta\left(\frac{1}{Q^*} - \frac{1}{D(q)}\right)\frac{1}{D^2(q)}\nabla D(q), & D(q) \le Q^*, \\ 0, & D(q) > Q^*, \end{cases} \tag{4.5}$$

where the $Q^* \in \mathbb{R}$ factor allows the robot to ignore obstacles sufficiently far away from it and the η can be viewed as a gain on the repulsive gradient. These scalars are usually determined by trial and error. (See figure 4.5.)

When numerically implementing this solution, a path may form that oscillates around points that are two-way equidistant from obstacles, i.e., points where D is nonsmooth. To avoid these oscillations, instead of defining the repulsive potential function in terms of distance to the *closest* obstacle, the repulsive potential function is redefined in terms of distances to *individual* obstacles where $d_i(q)$ is the distance to obstacle $\mathcal{QO}_i$, i.e.,

$$d_i(q) = \min_{c \in \mathcal{QO}_i} d(q, c). \tag{4.6}$$

Note that the min operator returns the smallest $d(q, c)$ for all points c in $\mathcal{QO}_i$.

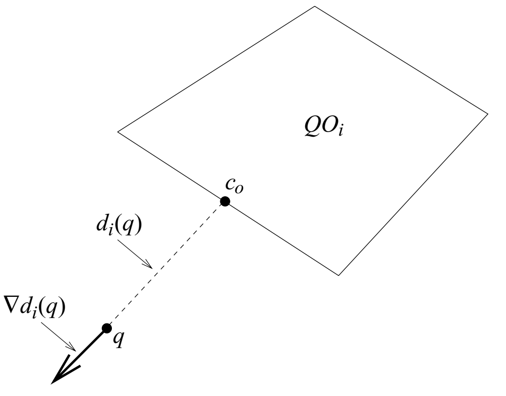


Figure 4.6 The distance between x and $\mathcal{QO}_i$ is the distance to the closest point on $\mathcal{QO}_i$. The gradient is a unit vector pointing away from the nearest point.

It can be shown for convex obstacles $\mathcal{QO}_i$ where c is the closest point to x that the gradient of $d_i(q)$ is

$$\nabla d_i(q) = \frac{q - c}{d(q, c)}. \tag{4.7}$$

The vector $\nabla d_i(q)$ describes the direction that maximally increases the distance to $\mathcal{QO}_i$ from q (figure 4.6).

Now, each obstacle has its own potential function,

$$U_{\text{rep}_i}(q) = \begin{cases} \frac{1}{2}\eta \left(\frac{1}{d_i(q)} - \frac{1}{Q_i^*}\right)^2, & \text{if } d_i(q) \leq Q_i^*, \\ 0, & \text{if } d_i(q) > Q_i^*, \end{cases}$$

where Q_i^* defines the size of the domain of influence for obstacle $\mathcal{QO}_i$. Then $U_{\text{rep}}(q) = \sum_{i=1}^{n} U_{\text{rep}_i}(q)$. Assuming that there are only convex obstacles or nonconvex ones can be decomposed into convex pieces, oscillations do not occur because the planner does not have radical changes in the closest point anymore.

4.2 Gradient Descent

Gradient descent is a well-known approach to optimization problems. The idea is simple. Starting at the initial configuration, take a small step in the direction opposite the gradient. This gives a new configuration, and the process is repeated until the gradient is zero. More formally, we can define a gradient descent algorithm (algorithm 4).