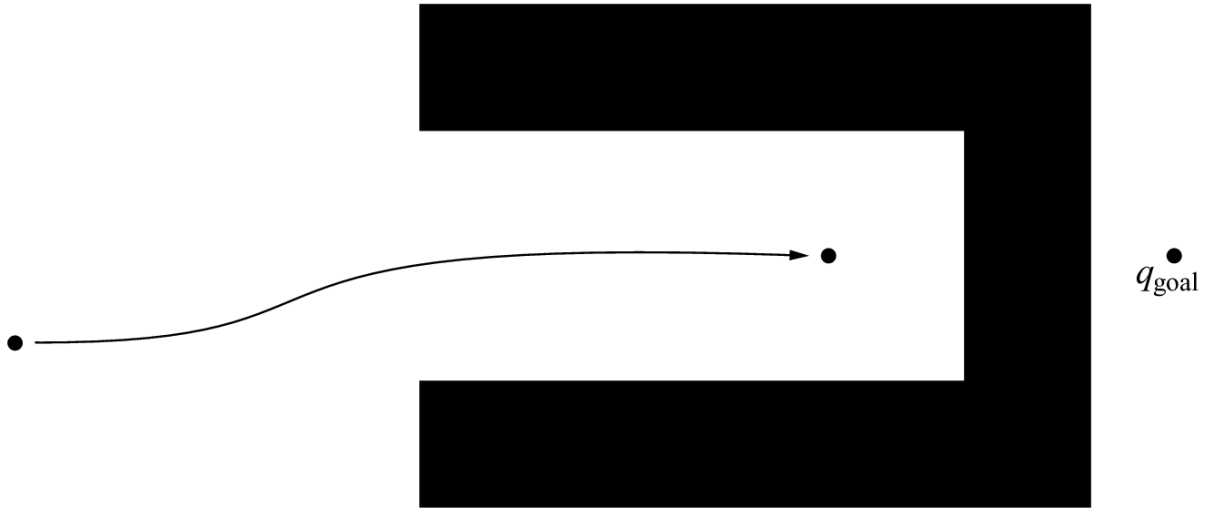
Navigation: Potential fields

Local minima

What could go wrong?

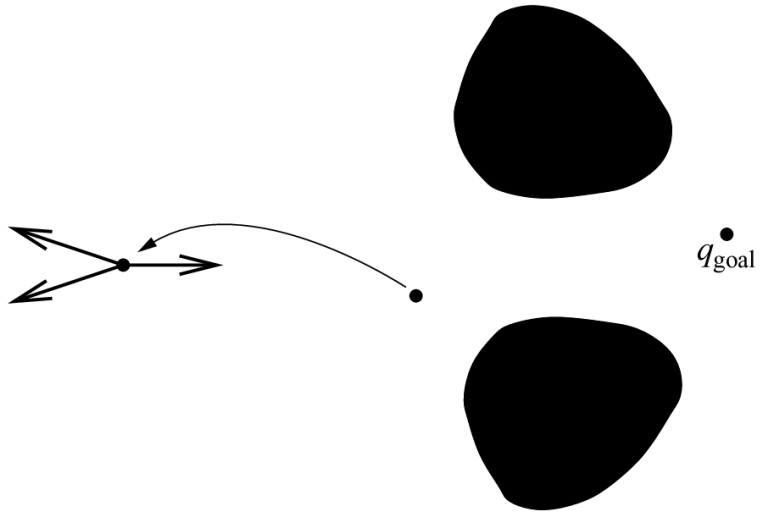
The main limitation of the potential field method is the problem of **local minima**: points other than the goal (the "global minimum") at which the gradient is zero.



The robot gets "stuck". There is **no guarantee** that the robot will reach its goal.

What could go wrong?

The main limitation of the potential field method is the problem of **local minima**: points other than the goal (the "global minimum") at which the gradient is zero.



The robot gets "stuck". There is **no guarantee** that the robot will reach its goal.

Avoiding local minima

How can a robot avoid becoming stuck in a local minimum of its potential field?

- Find a better potential function that doesn't have local minima. (A **navigation function**.)
- **Notice** that we've reached a local minimum, and try to escape. (Short-term gradient ascent; random motions; etc)

Avoiding local minima

How can a robot avoid becoming stuck in a local minimum of its potential field?

- Find a better potential function that doesn't have local minima. (A **navigation function**.)
- **Notice** that we've reached a local minimum, and try to escape. (Short-term gradient ascent; random motions; etc)

Or... choose a different navigation algorithm altogether.

Summary

We saw three classes of navigation algorithms.

- Visibility graphs
- Bug algorithms
- Potential fields

When is it possible to use each one? When is it a good idea?