Planning: Part 1
Classical Planning

Computer Science 4766/6912

Department of Computer Science
Memorial University of Newfoundland

July 11, 2018

Planning

- Localization and mapping make little sense in isolation

Planning

- Localization and mapping make little sense in isolation
- Robots need to travel from point A to point B to achieve some task

Planning

- Localization and mapping make little sense in isolation
- Robots need to travel from point A to point B to achieve some task
- Path planning heavily studied for robot manipulators (i.e. arms)

Planning

- Localization and mapping make little sense in isolation
- Robots need to travel from point A to point B to achieve some task
- Path planning heavily studied for robot manipulators (i.e. arms)
- Manipulators usually operate at very high speeds, thus serious consideration of dynamics is required

Planning

- Localization and mapping make little sense in isolation
- Robots need to travel from point A to point B to achieve some task
- Path planning heavily studied for robot manipulators (i.e. arms)
- Manipulators usually operate at very high speeds, thus serious consideration of dynamics is required
- The situation with mobile robots is simpler:

Planning

- Localization and mapping make little sense in isolation
- Robots need to travel from point A to point B to achieve some task
- Path planning heavily studied for robot manipulators (i.e. arms)
- Manipulators usually operate at very high speeds, thus serious consideration of dynamics is required
- The situation with mobile robots is simpler:
  - Mobile robots operate at lower speeds, thus dynamics are usually not considered

Planning

- Localization and mapping make little sense in isolation
- Robots need to travel from point A to point B to achieve some task
- Path planning heavily studied for robot manipulators (i.e. arms)
- Manipulators usually operate at very high speeds, thus serious consideration of dynamics is required
- The situation with mobile robots is simpler:
  - Mobile robots operate at lower speeds, thus dynamics are usually not considered
  - Mobile robots have much lower degrees-of-freedom

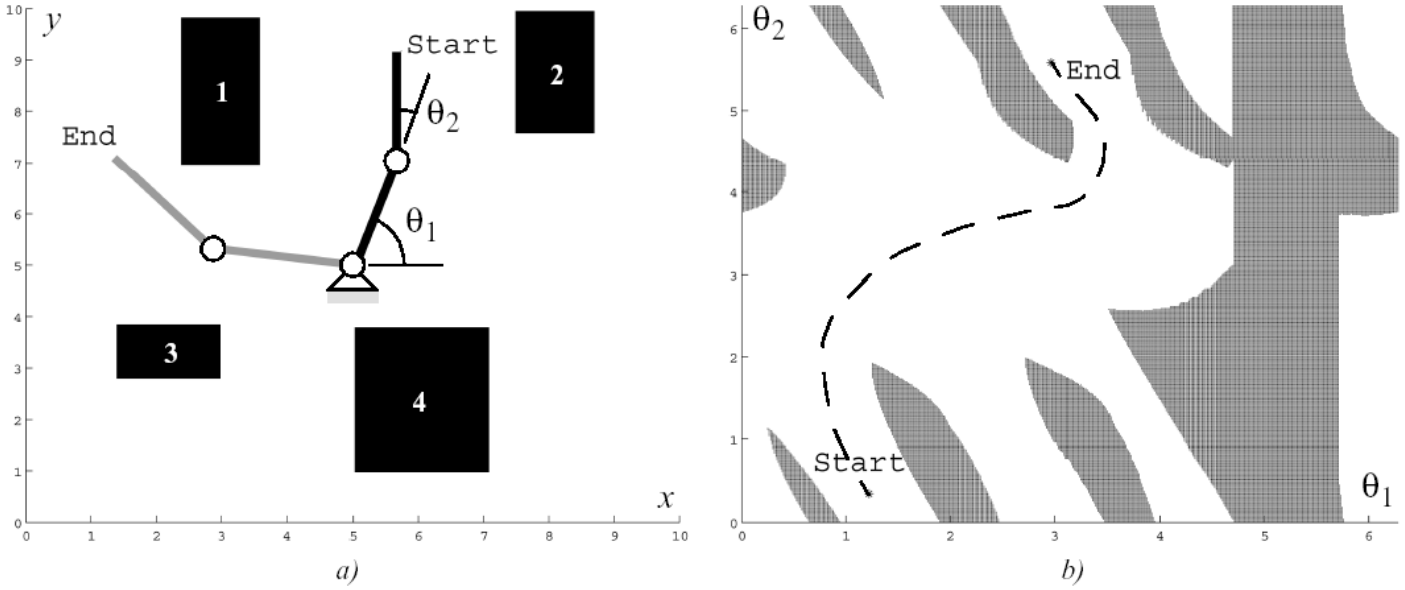
Consider a robot arm with k degrees-of-freedom

Consider a robot arm with k degrees-of-freedom. We can represent every possible configuration of the robot as a point in $k-$dimensional space

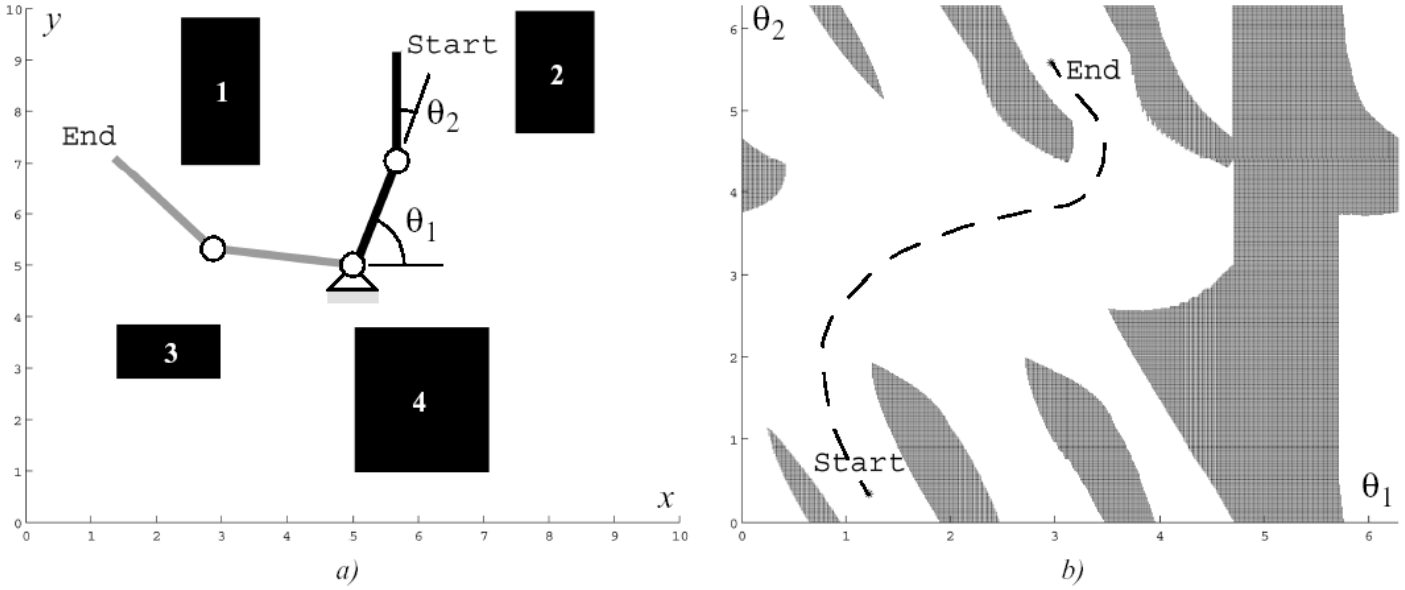
Consider a robot arm with k degrees-of-freedom. We can represent every possible configuration of the robot as a point in $k-$dimensional space. This space is known as **configuration space**

Consider a robot arm with k degrees-of-freedom. We can represent every possible configuration of the robot as a point in $k-$dimensional space. This space is known as **configuration space**. The arm pictured below has $k = 2$

Consider a robot arm with k degrees-of-freedom. We can represent every possible configuration of the robot as a point in $k-$dimensional space. This space is known as **configuration space**. The arm pictured below has $k = 2$,



Consider a robot arm with k degrees-of-freedom. We can represent every possible configuration of the robot as a point in $k-$dimensional space. This space is known as **configuration space**. The arm pictured below has $k = 2$,



Shaded positions in configuration space indicate that the robot would intersect objects in its workspace.

- For mobile robots operating in the plane, configuration space is just the space of possible x, y, θ positions

- For mobile robots operating in the plane, configuration space is just the space of possible x, y, θ positions
- It is often assumed that the robot is holonomic and can be represented as a 2-D point

- For mobile robots operating in the plane, configuration space is just the space of possible x, y, θ positions
- It is often assumed that the robot is holonomic and can be represented as a 2-D point
  - A differential-drive robot can follow the same path as a holonomic robot (however, its trajectory may differ)

- For mobile robots operating in the plane, configuration space is just the space of possible x, y, θ positions
- It is often assumed that the robot is holonomic and can be represented as a 2-D point
  - A differential-drive robot can follow the same path as a holonomic robot (however, its trajectory may differ)
  - We can account for the reduction of the robot to a point by *inflating* all obstacles by the robot's actual radius

- For mobile robots operating in the plane, configuration space is just the space of possible x, y, θ positions
- It is often assumed that the robot is holonomic and can be represented as a 2-D point
  - A differential-drive robot can follow the same path as a holonomic robot (however, its trajectory may differ)
  - We can account for the reduction of the robot to a point by *inflating* all obstacles by the robot's actual radius
- We will briefly describe three general approaches to planning:

- For mobile robots operating in the plane, configuration space is just the space of possible x, y, θ positions
- It is often assumed that the robot is holonomic and can be represented as a 2-D point
  - A differential-drive robot can follow the same path as a holonomic robot (however, its trajectory may differ)
  - We can account for the reduction of the robot to a point by *inflating* all obstacles by the robot's actual radius
- We will briefly describe three general approaches to planning:
  1. Road maps: Identify a set of routes within the free space

- For mobile robots operating in the plane, configuration space is just the space of possible x, y, θ positions
- It is often assumed that the robot is holonomic and can be represented as a 2-D point
  - A differential-drive robot can follow the same path as a holonomic robot (however, its trajectory may differ)
  - We can account for the reduction of the robot to a point by *inflating* all obstacles by the robot's actual radius
- We will briefly describe three general approaches to planning:
  1. Road maps: Identify a set of routes within the free space
  2. Cell decomposition: Discriminate between free and occupied cells

- For mobile robots operating in the plane, configuration space is just the space of possible x, y, θ positions
- It is often assumed that the robot is holonomic and can be represented as a 2-D point
  - A differential-drive robot can follow the same path as a holonomic robot (however, its trajectory may differ)
  - We can account for the reduction of the robot to a point by *inflating* all obstacles by the robot's actual radius
- We will briefly describe three general approaches to planning:
  1. Road maps: Identify a set of routes within the free space
  2. Cell decomposition: Discriminate between free and occupied cells
  3. Potential fields: A potential function attracts the robot to the goal, while repelling it from obstacles

- For mobile robots operating in the plane, configuration space is just the space of possible x, y, θ positions
- It is often assumed that the robot is holonomic and can be represented as a 2-D point
  - A differential-drive robot can follow the same path as a holonomic robot (however, its trajectory may differ)
  - We can account for the reduction of the robot to a point by *inflating* all obstacles by the robot's actual radius
- We will briefly describe three general approaches to planning:
  1. Road maps: Identify a set of routes within the free space
  2. Cell decomposition: Discriminate between free and occupied cells
  3. Potential fields: A potential function attracts the robot to the goal, while repelling it from obstacles
- The first two actually just describe how to decompose space into a graph. Once the graph is obtained, a shortest path algorithm (e.g. Dijkstra, A*) is applied.

Road Maps

Describe the robot's free space as a network of lines and/or curves

Road Maps

Describe the robot's free space as a network of lines and/or curves. Path planning can then be achieved by determining the start and end points and applying standard algorithms from graph theory (with appropriate weights)

Road Maps

Describe the robot's free space as a network of lines and/or curves. Path planning can then be achieved by determining the start and end points and applying standard algorithms from graph theory (with appropriate weights).

If our map is composed of polygonal obstacles we can apply a **visibility graph**

Road Maps

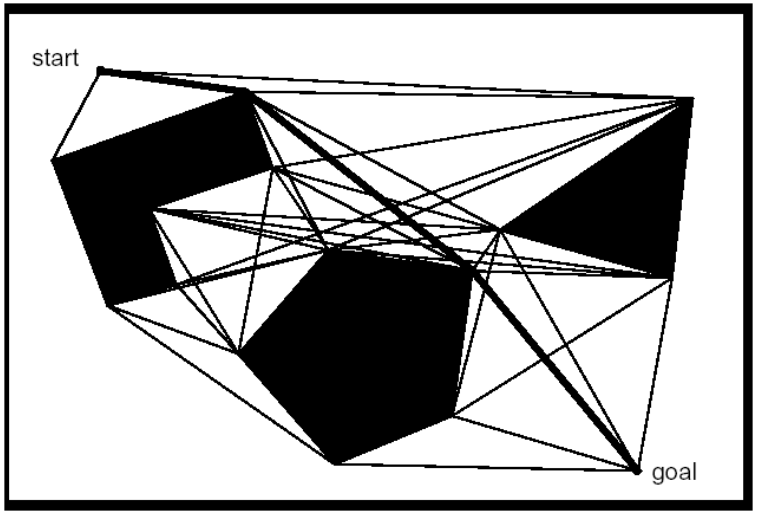
Describe the robot's free space as a network of lines and/or curves. Path planning can then be achieved by determining the start and end points and applying standard algorithms from graph theory (with appropriate weights).

If our map is composed of polygonal obstacles we can apply a **visibility graph**. A visibility graph consists of the set of edges obtained by joining all pairs of vertices that can see each other (including the start and goal vertices).

Road Maps

Describe the robot's free space as a network of lines and/or curves. Path planning can then be achieved by determining the start and end points and applying standard algorithms from graph theory (with appropriate weights).

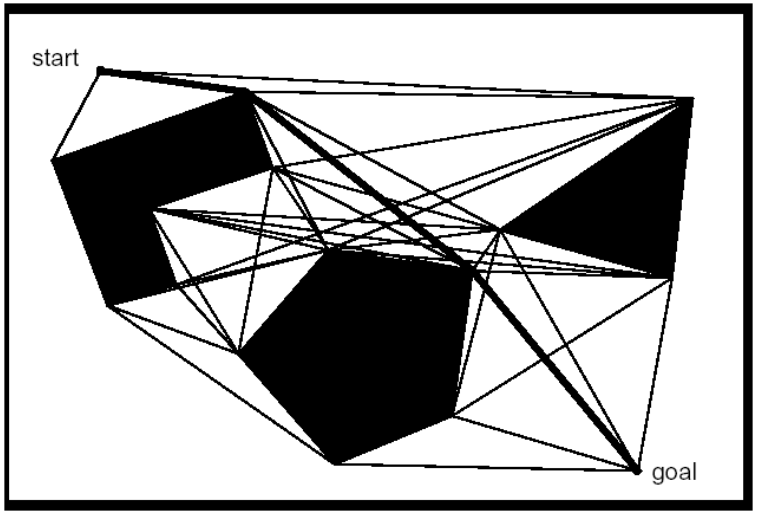
If our map is composed of polygonal obstacles we can apply a **visibility graph**. A visibility graph consists of the set of edges obtained by joining all pairs of vertices that can see each other (including the start and goal vertices).



Road Maps

Describe the robot's free space as a network of lines and/or curves. Path planning can then be achieved by determining the start and end points and applying standard algorithms from graph theory (with appropriate weights).

If our map is composed of polygonal obstacles we can apply a **visibility graph**. A visibility graph consists of the set of edges obtained by joining all pairs of vertices that can see each other (including the start and goal vertices).



Visibility graphs are easy to implement and generate optimal (shortest possible length) paths

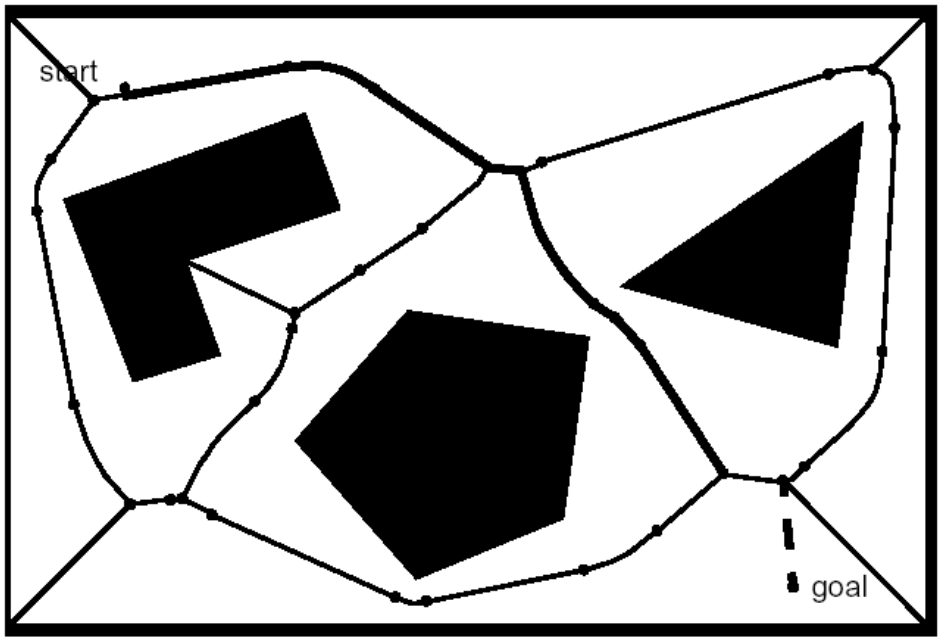
Visibility graphs are easy to implement and generate optimal (shortest possible length) paths. However, these paths skirt the edges of obstacles, possibly jeopordizing the robot.

Visibility graphs are easy to implement and generate optimal (shortest possible length) paths. However, these paths skirt the edges of obstacles, possibly jeopordizing the robot.

A **generalized Voronoi diagram** (GVD) consists of all points in free space which are equidistant to the two closest obstacles.

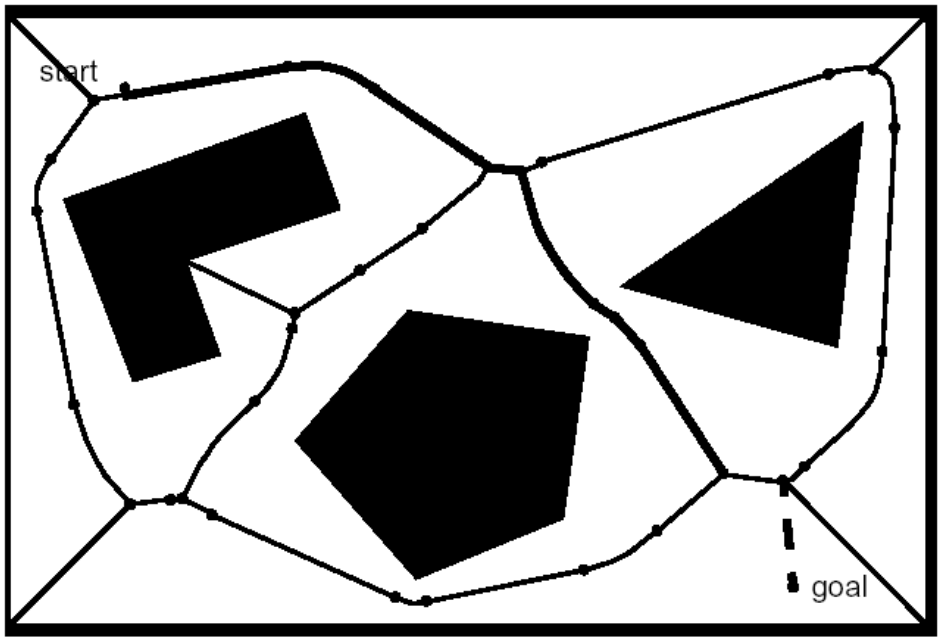
Visibility graphs are easy to implement and generate optimal (shortest possible length) paths. However, these paths skirt the edges of obstacles, possibly jeopordizing the robot.

A **generalized Voronoi diagram** (GVD) consists of all points in free space which are equidistant to the two closest obstacles.



Visibility graphs are easy to implement and generate optimal (shortest possible length) paths. However, these paths skirt the edges of obstacles, possibly jeopordizing the robot.

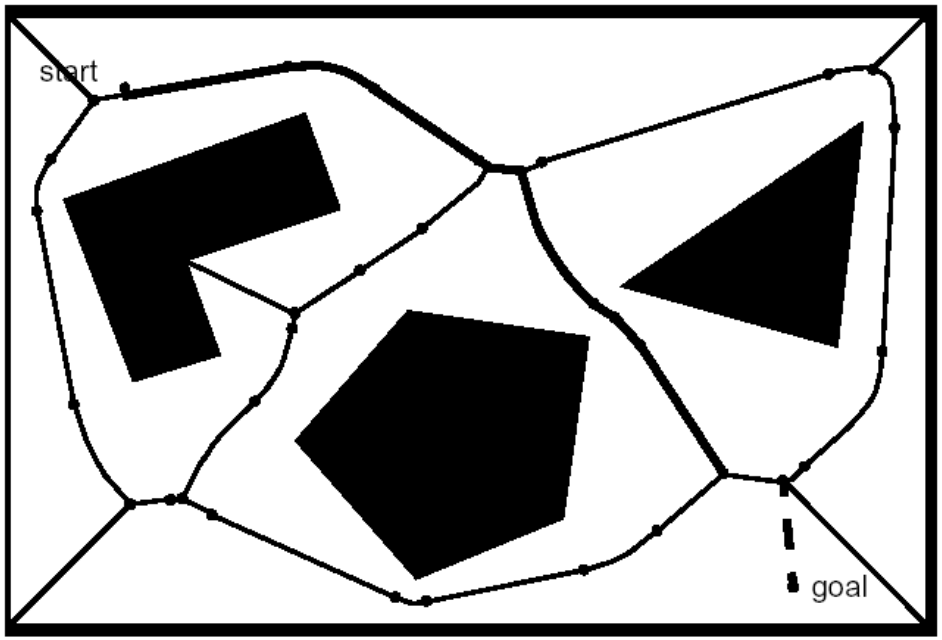
A **generalized Voronoi diagram** (GVD) consists of all points in free space which are equidistant to the two closest obstacles.



Paths are safer, but longer, than those of visibility graphs

Visibility graphs are easy to implement and generate optimal (shortest possible length) paths. However, these paths skirt the edges of obstacles, possibly jeopordizing the robot.

A **generalized Voronoi diagram** (GVD) consists of all points in free space which are equidistant to the two closest obstacles.



Paths are safer, but longer, than those of visibility graphs. A robot not on the GVD can easily join it by moving away from the nearest obstacle until the GVD is reached.

Cell Decomposition

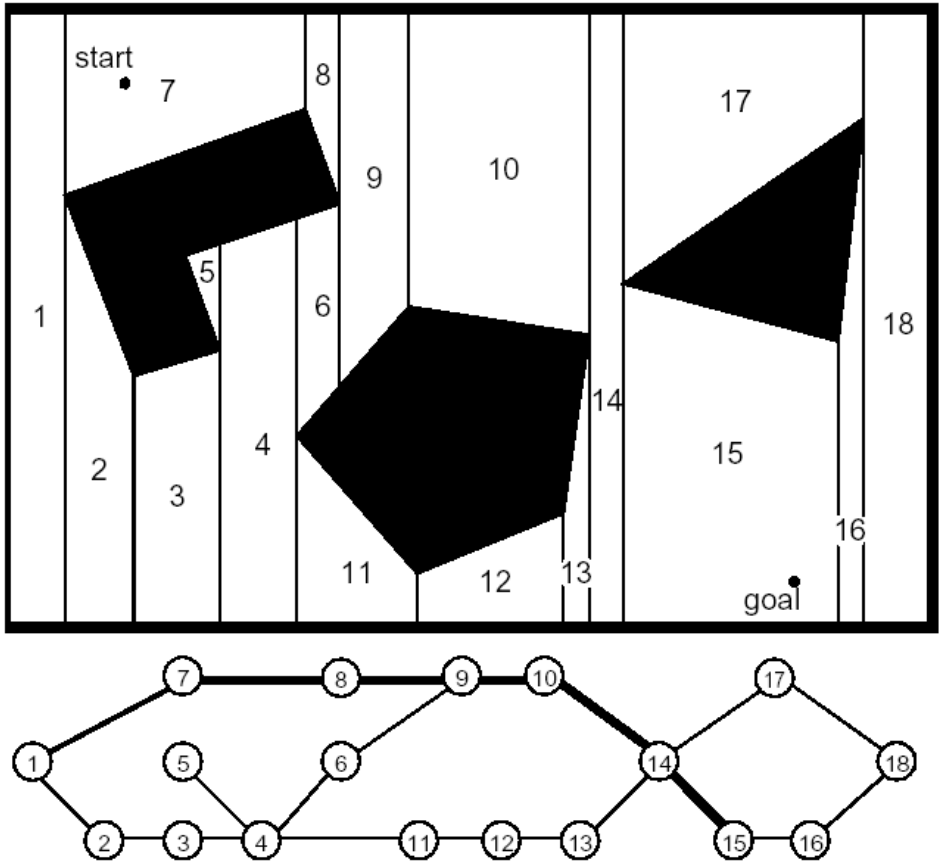
Divide the configuration space into free and occupied cells

Cell Decomposition

Divide the configuration space into free and occupied cells. Then form a **connectivity graph** which describes adjacency relationships between free cells.

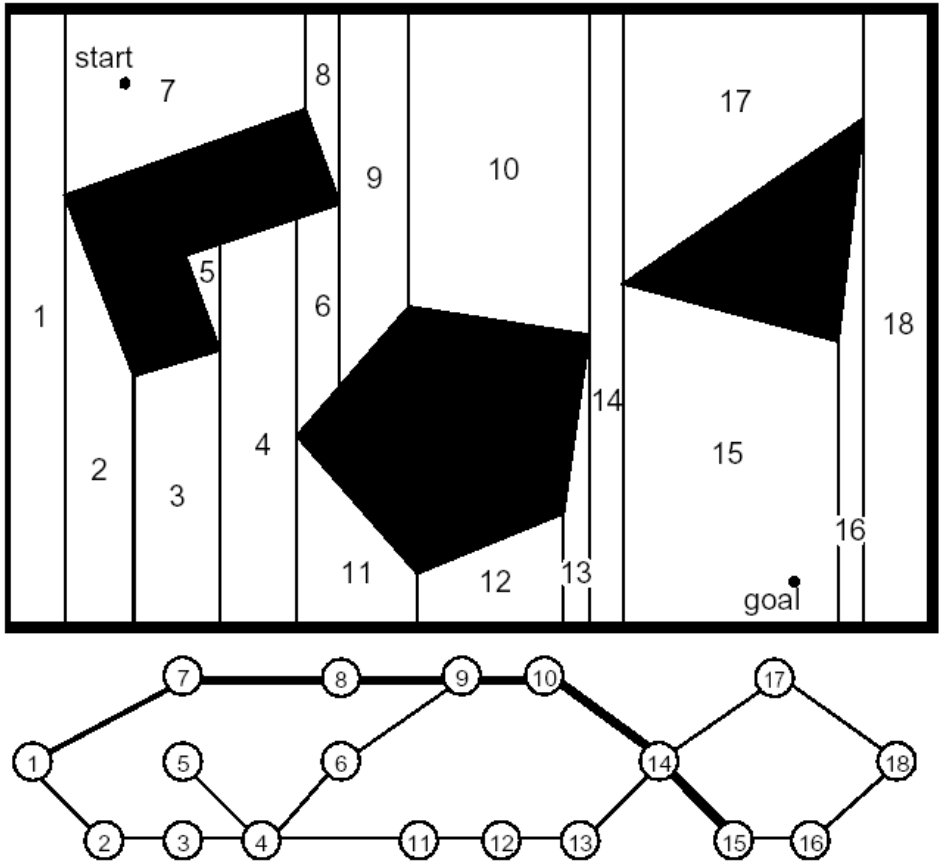
Cell Decomposition

Divide the configuration space into free and occupied cells. Then form a **connectivity graph** which describes adjacency relationships between free cells.



Cell Decomposition

Divide the configuration space into free and occupied cells. Then form a **connectivity graph** which describes adjacency relationships between free cells.



Potential Fields

Define attractive and repulsive potential fields over the configuration space

Potential Fields

Define attractive and repulsive potential fields over the configuration space. The robot is attracted to a goal by defining an attractive potential function that is minimized at the goal

Potential Fields

Define attractive and repulsive potential fields over the configuration space. The robot is attracted to a goal by defining an attractive potential function that is minimized at the goal. Similarly, define repulsive functions that are maximized at obstacle positions

Potential Fields

Define attractive and repulsive potential fields over the configuration space. The robot is attracted to a goal by defining an attractive potential function that is minimized at the goal. Similarly, define repulsive functions that are maximized at obstacle positions. All potential functions are added to form the potential field $U(x_t)$

Potential Fields

Define attractive and repulsive potential fields over the configuration space. The robot is attracted to a goal by defining an attractive potential function that is minimized at the goal. Similarly, define repulsive functions that are maximized at obstacle positions. All potential functions are added to form the potential field $U(x_t)$.

The robot reaches the goal by moving along the negative of the gradient of the potential field,

Potential Fields

Define attractive and repulsive potential fields over the configuration space. The robot is attracted to a goal by defining an attractive potential function that is minimized at the goal. Similarly, define repulsive functions that are maximized at obstacle positions. All potential functions are added to form the potential field $U(x_t)$.

The robot reaches the goal by moving along the negative of the gradient of the potential field,

$$\nabla U(x_t) = -\begin{bmatrix} \frac{\partial U}{\partial x} \\ \frac{\partial U}{\partial y} \end{bmatrix}$$

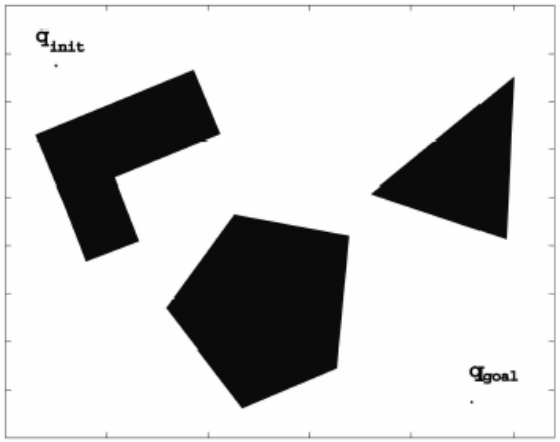
Potential Fields

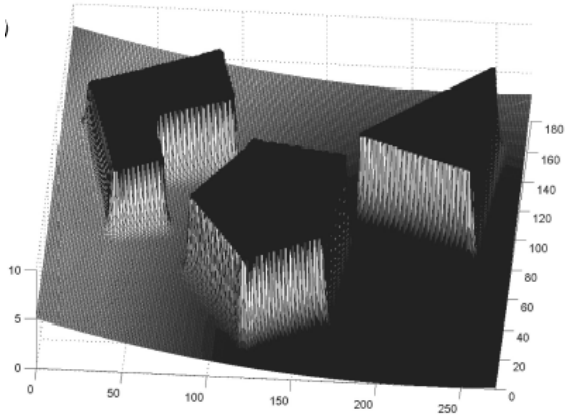
Define attractive and repulsive potential fields over the configuration space. The robot is attracted to a goal by defining an attractive potential function that is minimized at the goal. Similarly, define repulsive functions that are maximized at obstacle positions. All potential functions are added to form the potential field $U(x_t)$.

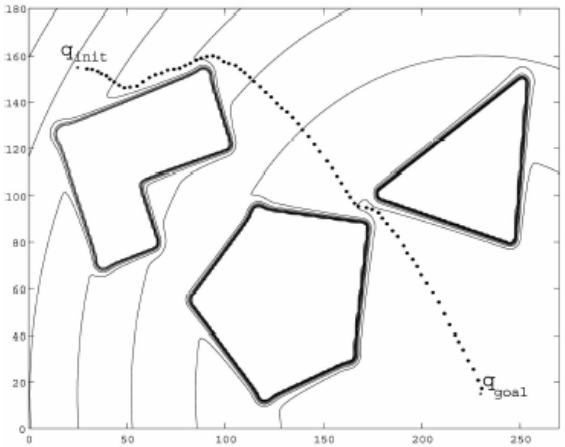
The robot reaches the goal by moving along the negative of the gradient of the potential field,

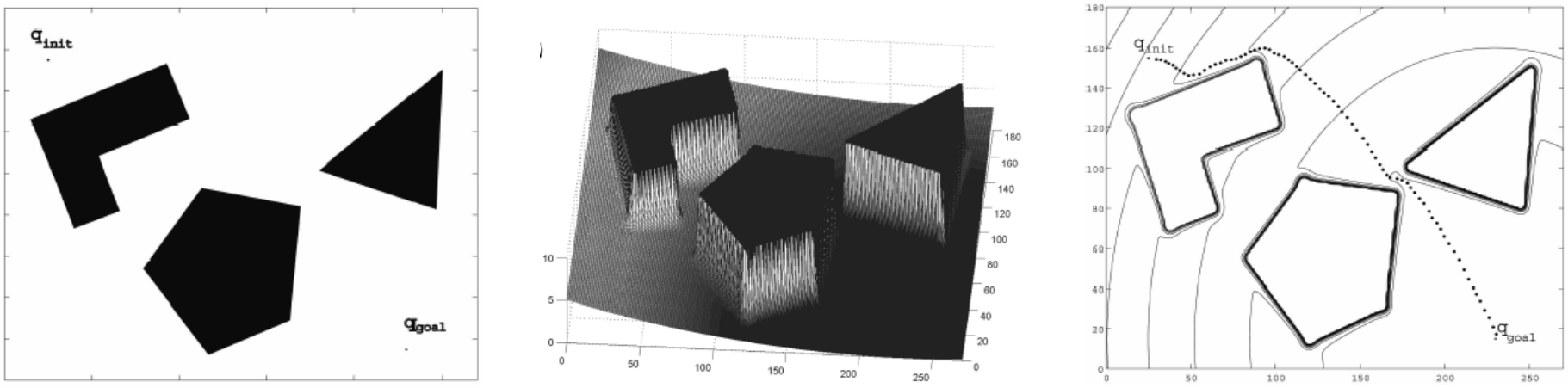
$$\nabla U(x_t) = -\begin{bmatrix} \frac{\partial U}{\partial x} \\ \frac{\partial U}{\partial y} \end{bmatrix}$$

This technique is more than just a path planning device. The resulting vectors provide a low level control law for guiding the robot.


q_{init}
q_{goal}


10
5
0
0
50
100
150
200
250
0
20
40
60
80
100
120
140
160
180


q_{init}
q_{goal}
0
20
40
60
80
100
120
140
160
180
0
50
100
150
200
250



Left: Obstacle map

Centre: Potential field

Right: Gradient contour and path followed