# CS686: Configuration Space II

Sung-Eui Yoon
(윤성의)

**Course URL:**
http://sgvr.kaist.ac.kr/~sungeui/MPA

KAIST

# Coming Schedule and Homework

- Browse recent papers (2015 ~ 2019)
  - You need to present two papers at the class
- Declare your chosen 2 papers at the KLMS by Oct-14 (Mon.)
  - First come, first served
  - Paper title, conf. name, publication year
- Student presentations will start right after the mid-term exam
  - 2 talks per each class; 25 min for each talk
  - Each presenter needs two short quiz

# Class Objectives

- Configuration space
  - Definitions and examples
  - Obstacles
  - Paths
  - Metrics

# Obstacles in the Configuration Space

- A configuration $q$ is collision-free, or **free**, if a moving object placed at $q$ does not intersect any obstacles in the workspace

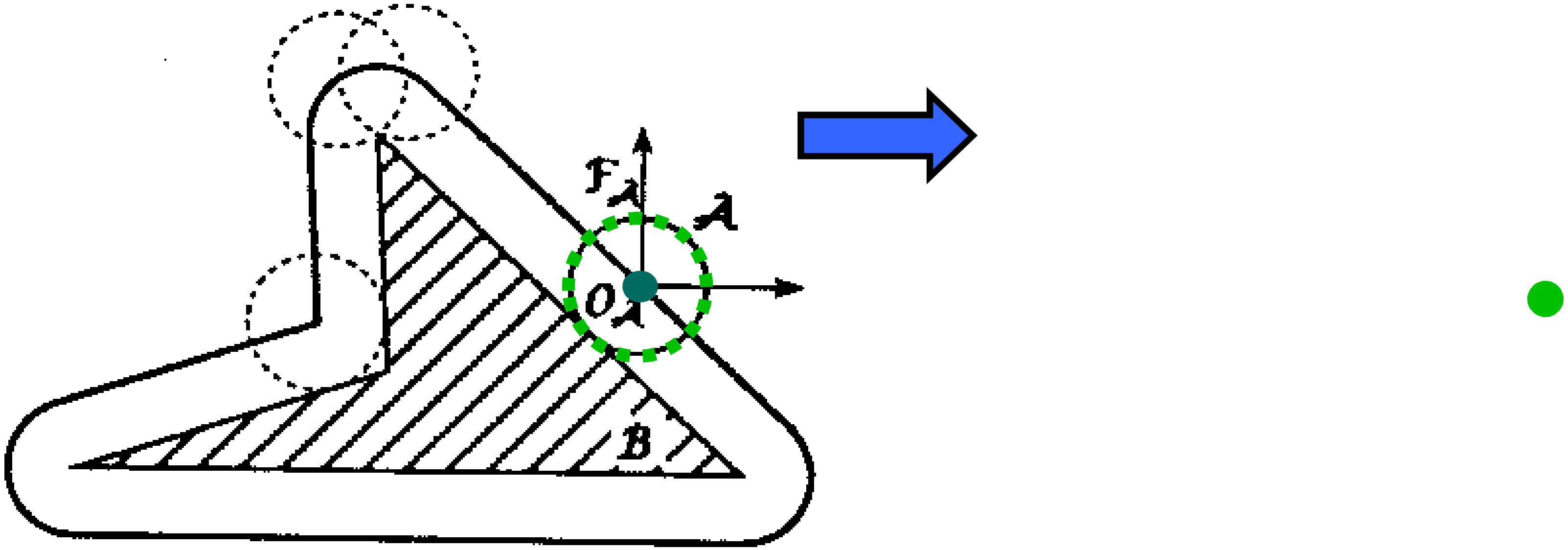
- The **free space** $F$ is the set of free configurations

- A configuration space obstacle (**C-obstacle**) is the set of configurations where the moving object collides with workspace obstacles

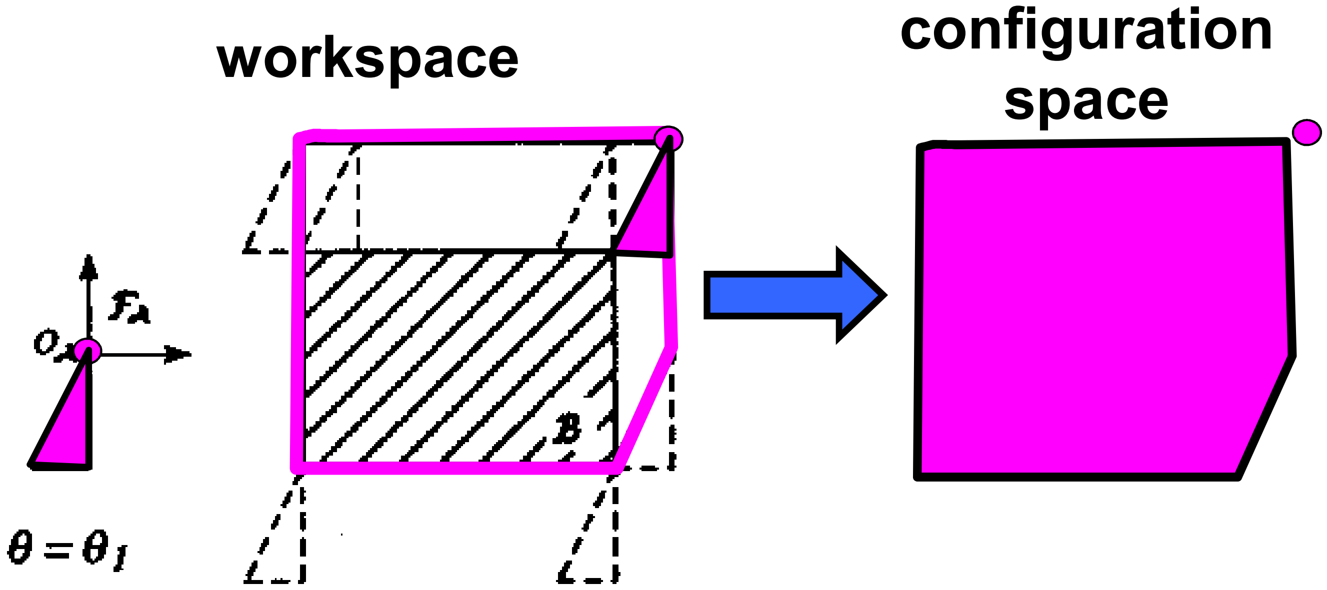
# Disc in 2-D Workspace

**workspace**

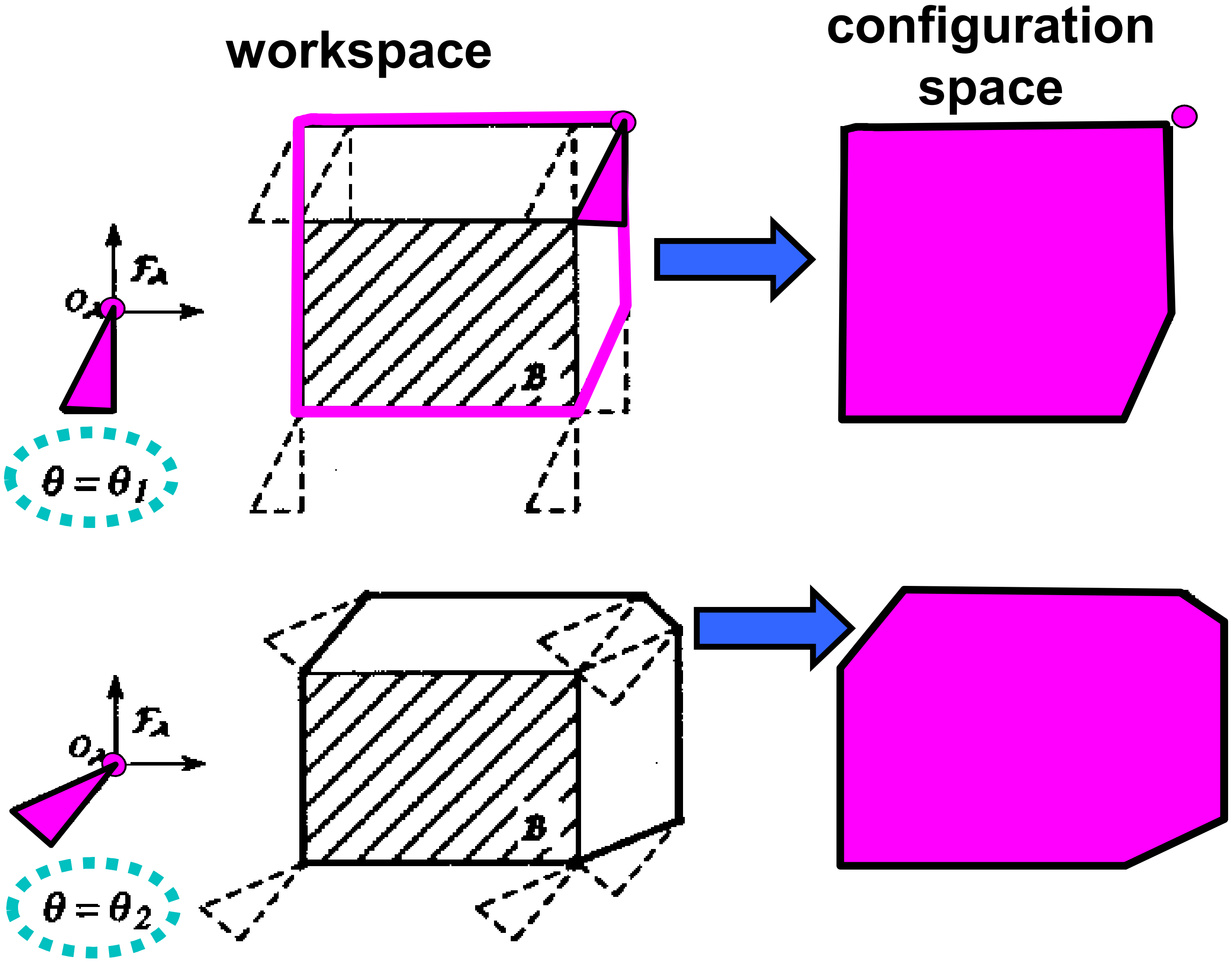
**configuration space**



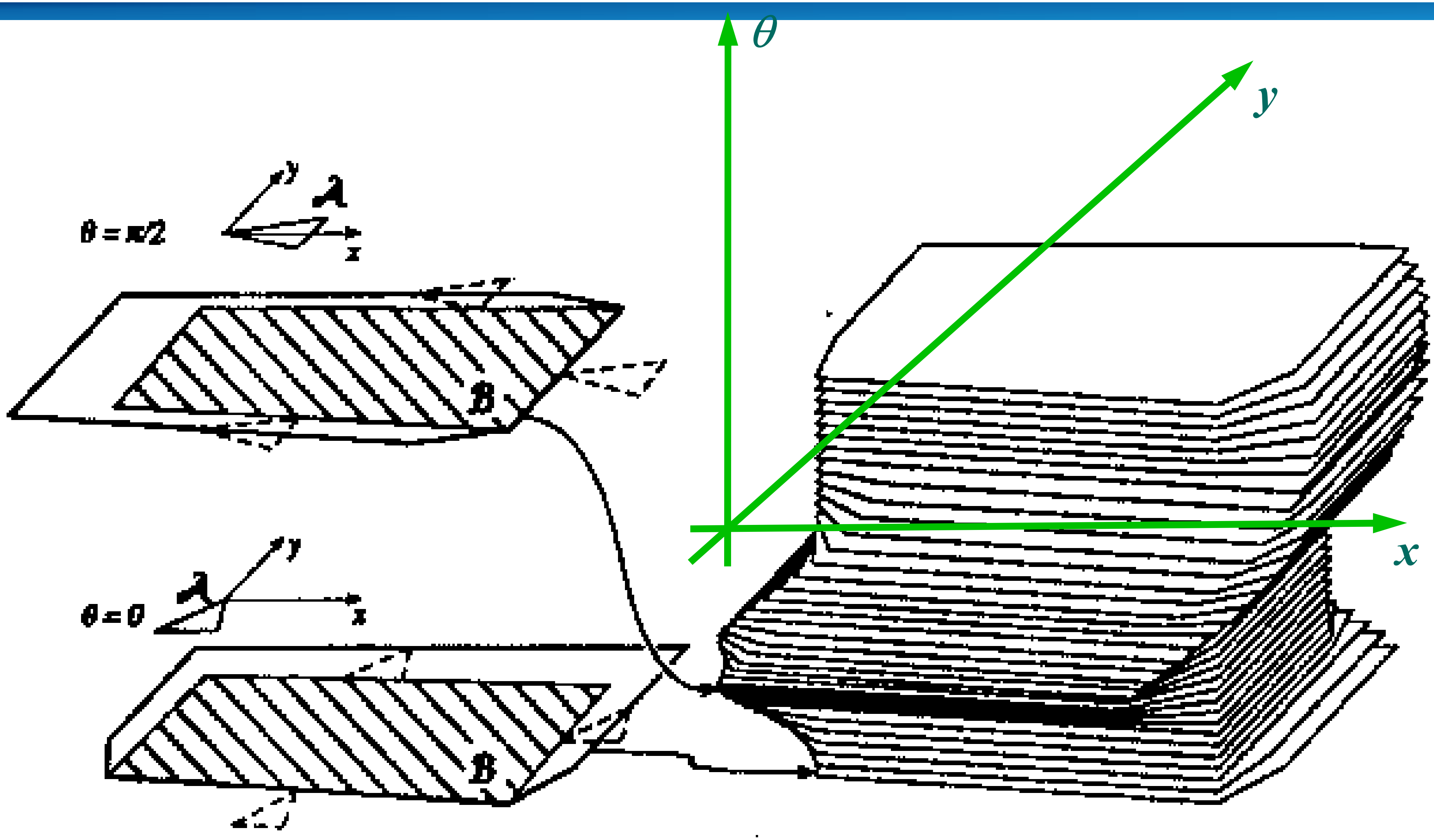
# Polygonal Robot Translating in 2-D Workspace



# Polygonal Robot Translating & Rotating in 2-D Workspace



# Polygonal Robot Translating & Rotating in 2-D Workspace



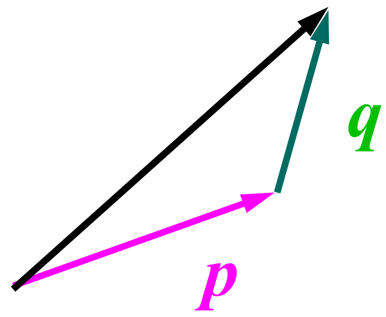
# C-Obstacle Construction

- Input:
  - Polygonal moving object translating in 2-D workspace
  - Polygonal obstacles
- Output:
  - Configuration space obstacles represented as polygons

# Minkowski Sum

- The **Minkowski sum** of two sets $P$ and $Q$, denoted by $P \oplus Q$, is defined as

$$P \oplus Q = \{ p+q \mid p \in P, q \in Q \}$$




- Similarly, the **Minkowski difference** is defined as

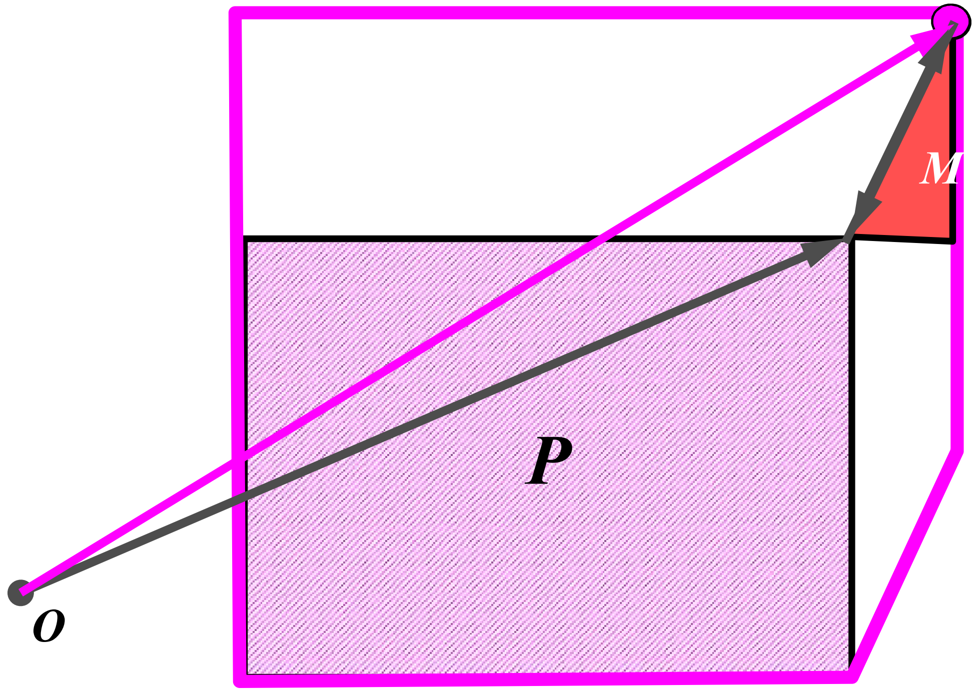
$$P \ominus Q = \{ p-q \mid p \in P, q \in Q \}$$
$$= P \oplus -Q$$

# Minkowski Sum of Convex Polygons

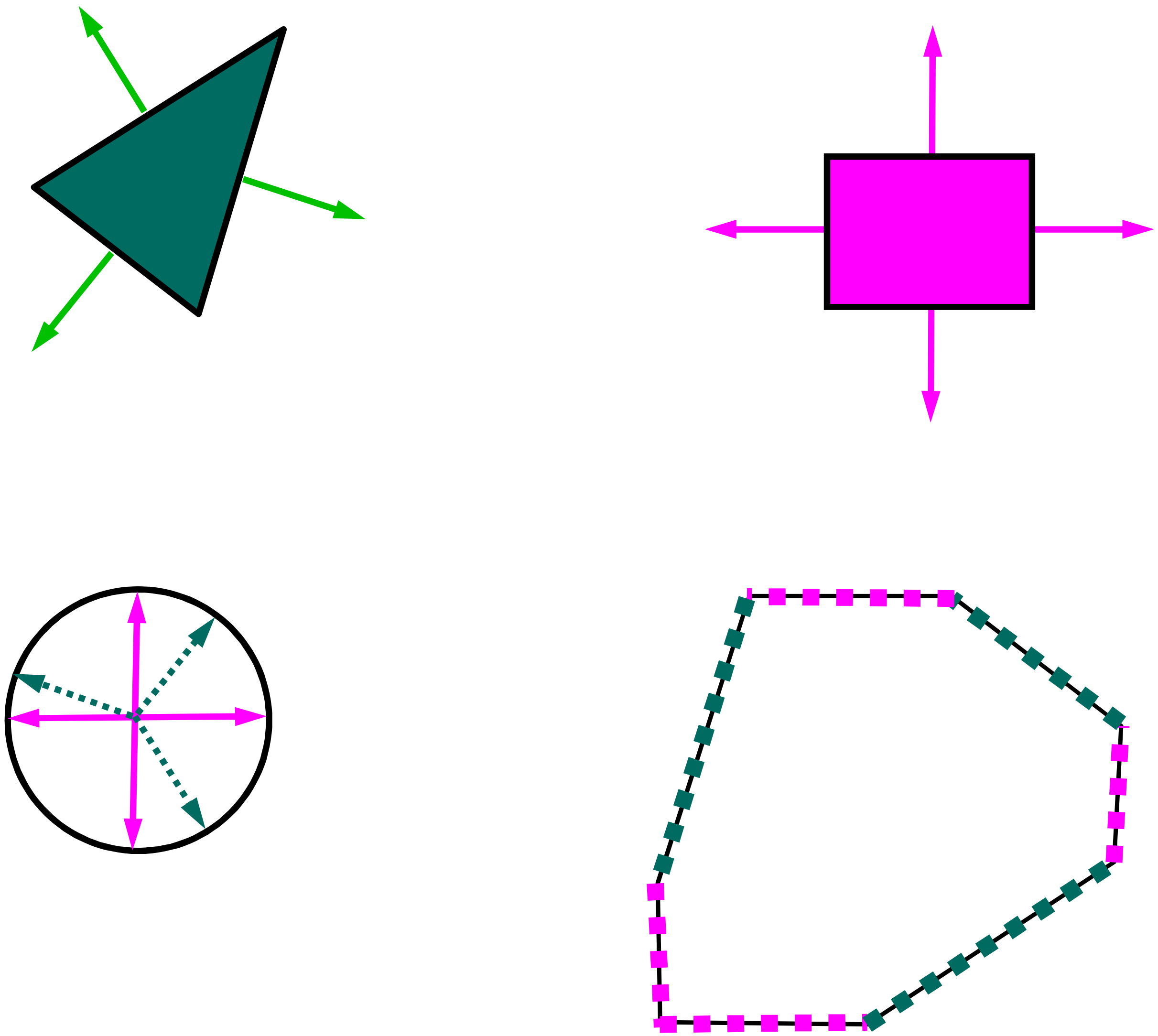
- **The Minkowski sum of two convex polygons $P$ and $Q$ of $m$ and $n$ vertices respectively is a convex polygon $P \oplus Q$ of $m + n$ vertices.**
  - **The vertices of $P \oplus Q$ are the "sums" of vertices of $P$ and $Q$.**

# Observation

- **If $P$ is an obstacle in the workspace and $M$ is a moving object. Then the C-space obstacle corresponding to P is $P \ominus M$.**



# Computing C-obstacles



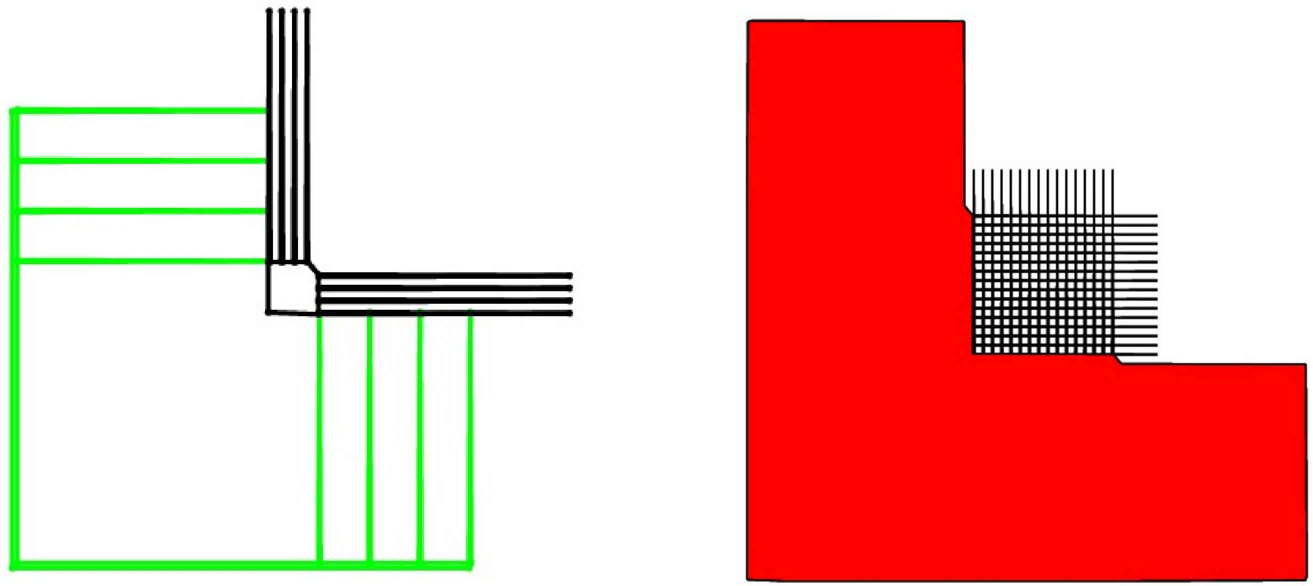
# Computational efficiency

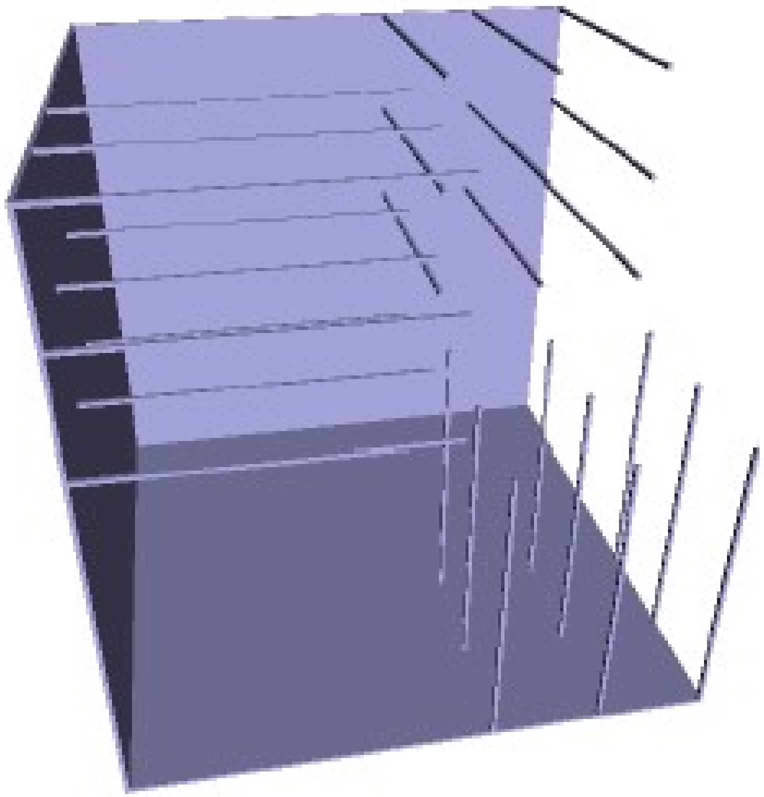
- **Running time** $O(n+m)$
- **Space** $O(n+m)$
- **Non-convex obstacles**
  - **Decompose into convex polygons (*e.g.,* triangles or trapezoids)*,* compute the Minkowski sums, and take the union**
  - **Complexity of Minkowksi sum** $O(n^2m^2)$
- **3-D workspace**
  - **Convex case:** $O(nm)$
  - **Non-convex case:** $O(n^3m^3)$

# Worst case example

- $O(n^2m^2)$ complexity

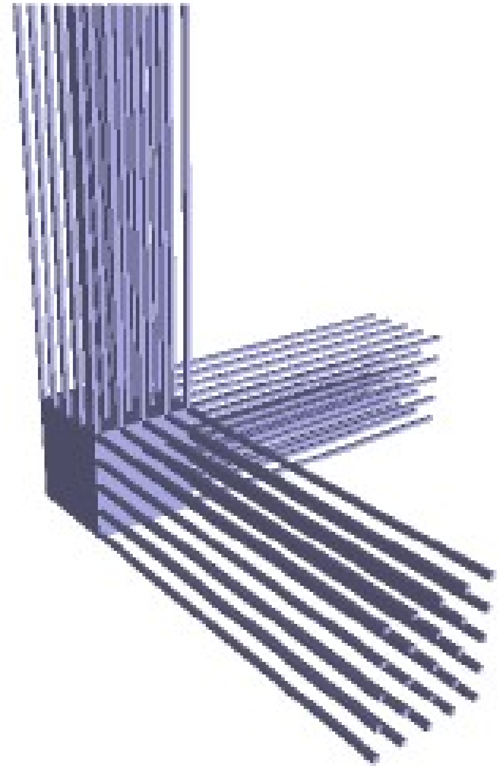
2D example
Agarwal et al. 02



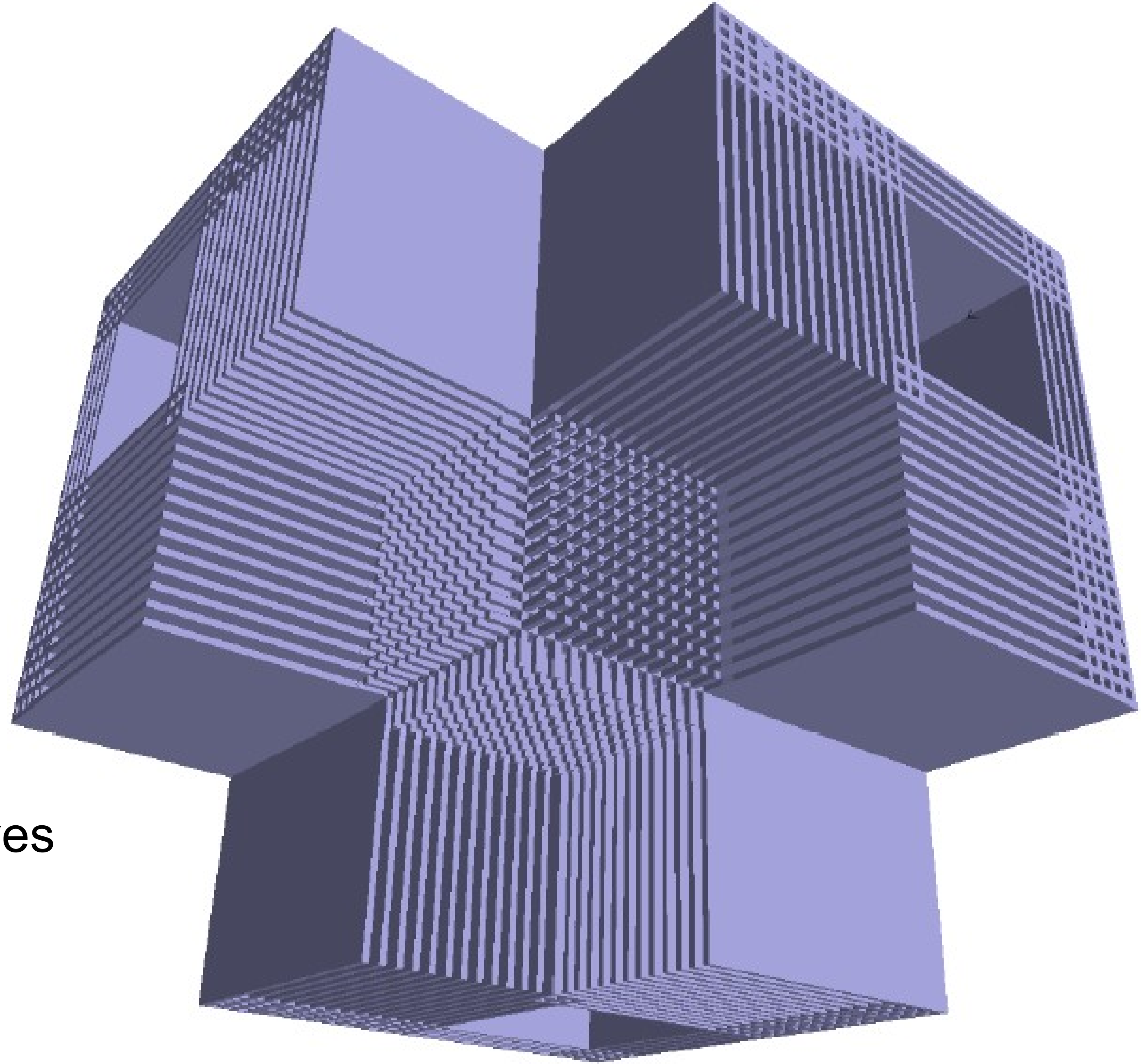


444 tris

$\oplus$



1,134 tris



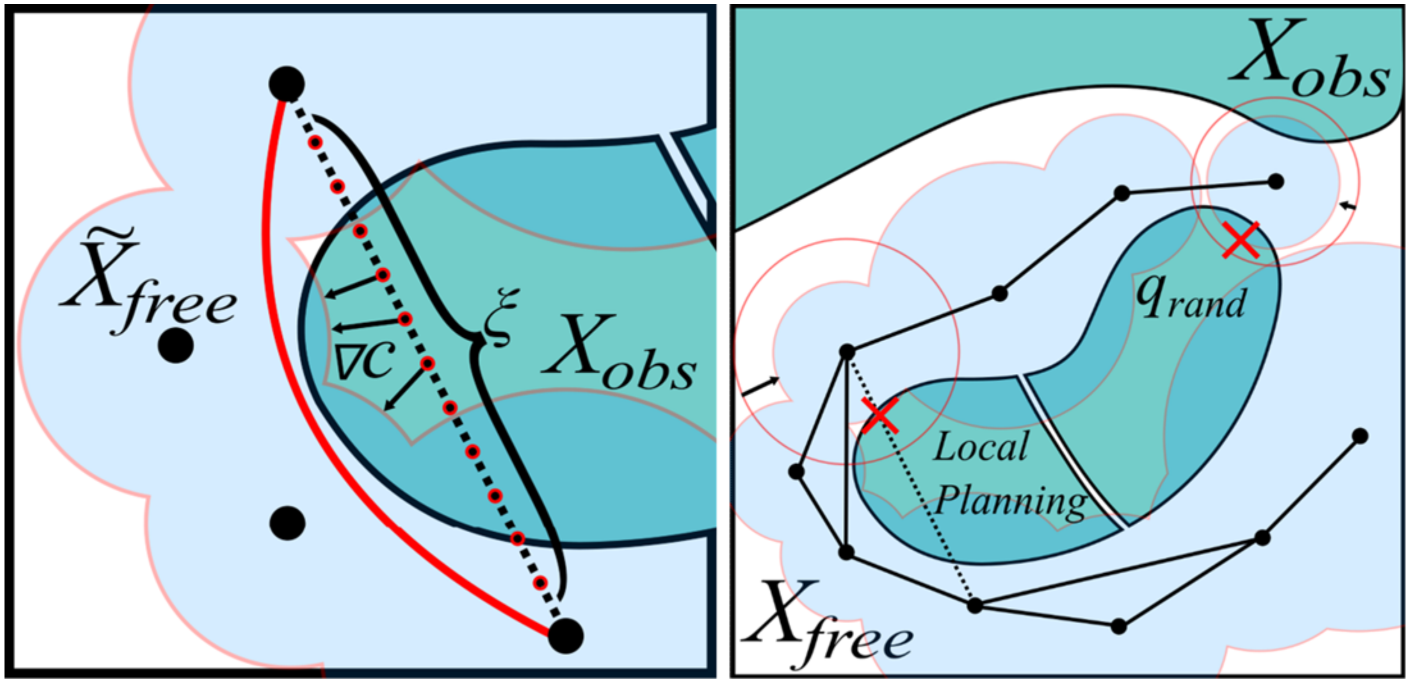
Union of
66,667 primitives

# Main Message

- **Computing the free or obstacle space in an accurate way is an expensive and non-trivial problem**
- **Lead to many sampling based methods**
  - **Locally utilize many geometric concepts developed for designing complete planners**

# Approximation of Configuration Free Space

- **Dancing PRM* : Simultaneous Planning of Sampling and Optimization with Configuration Free Space Approximation**
  - **Approximate C-Space and perform planning**
  - **Improve the quality in an iterative manner**




Video

# Sensors!

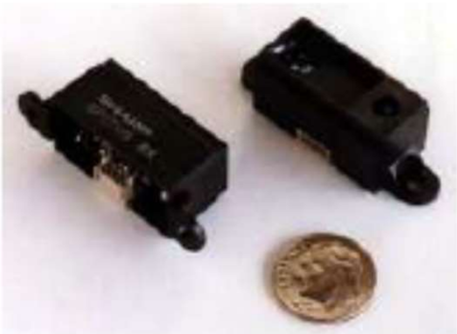
Robots' link to the external world...

sonar rangefinder

gyro

**Sensors, sensors, sensors!**
**and tracking what is sensed: world models**

compass



IR rangefinder

sonar rangefinder

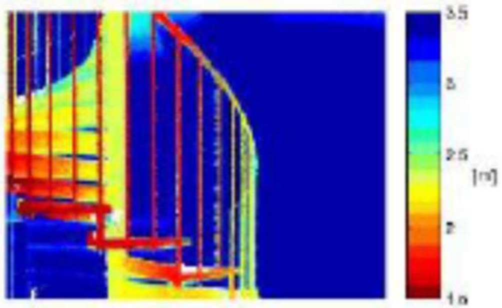
CMU cam with on-board processing

odometry…

# Laser Ranging

LIDAR

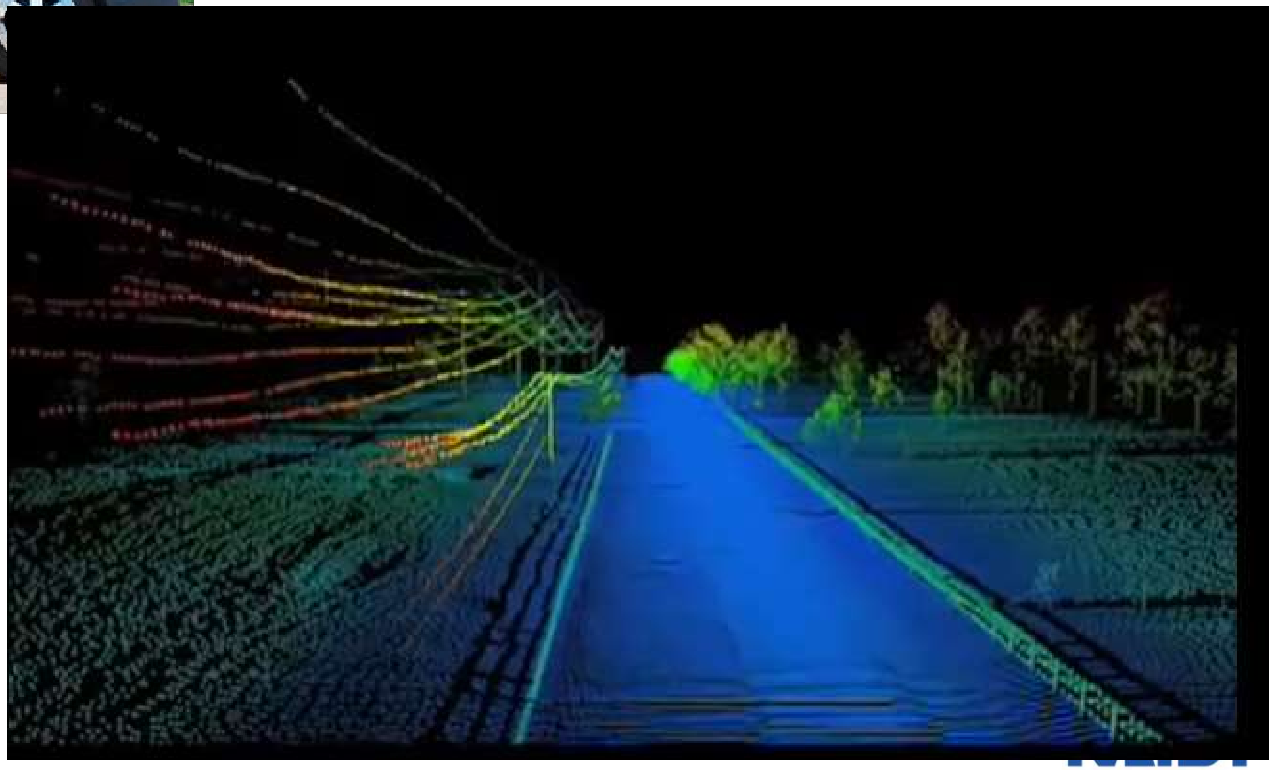
Sick Laser




LIDAR map

Sick laser range finder

# Velodyne



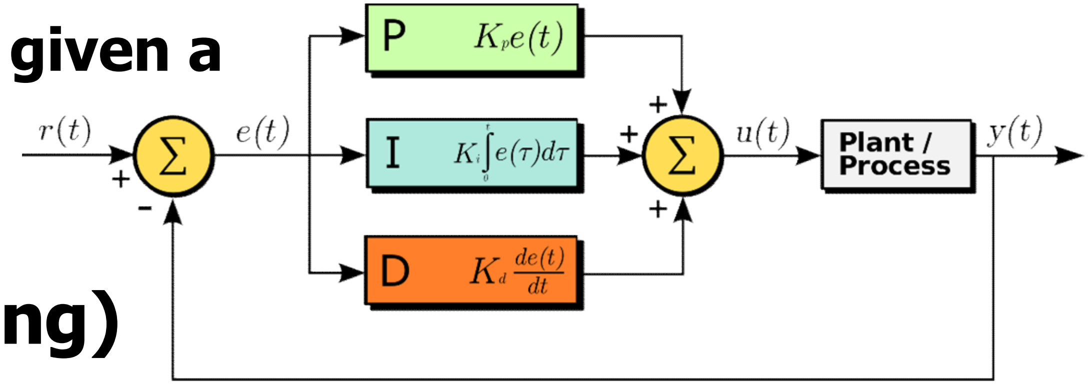
# Kinect and Xtion



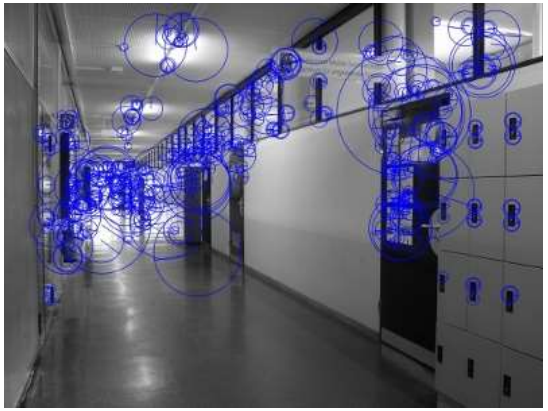
- **Kinect resolution**
  - **640×480 pixels @ 30 Hz (RGB camera)**
  - **640×480 pixels @ 30 Hz (IR depth-finding camera)**

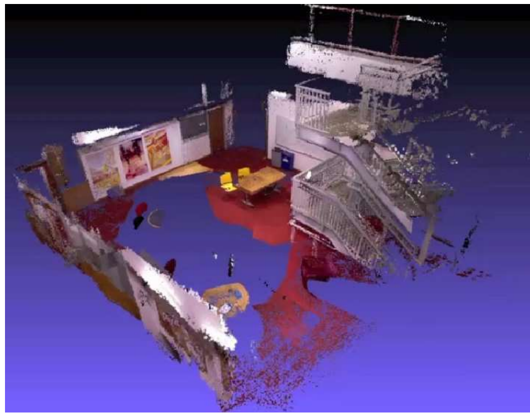
# Whole Picture

- **Sensor**
  - **Point clouds as obstacle map**
- **Control**
  - **Compute force controls given a computed path**




- **SLAM (Simultaneous Localization and Mapping)**
- **Path/motion planner**

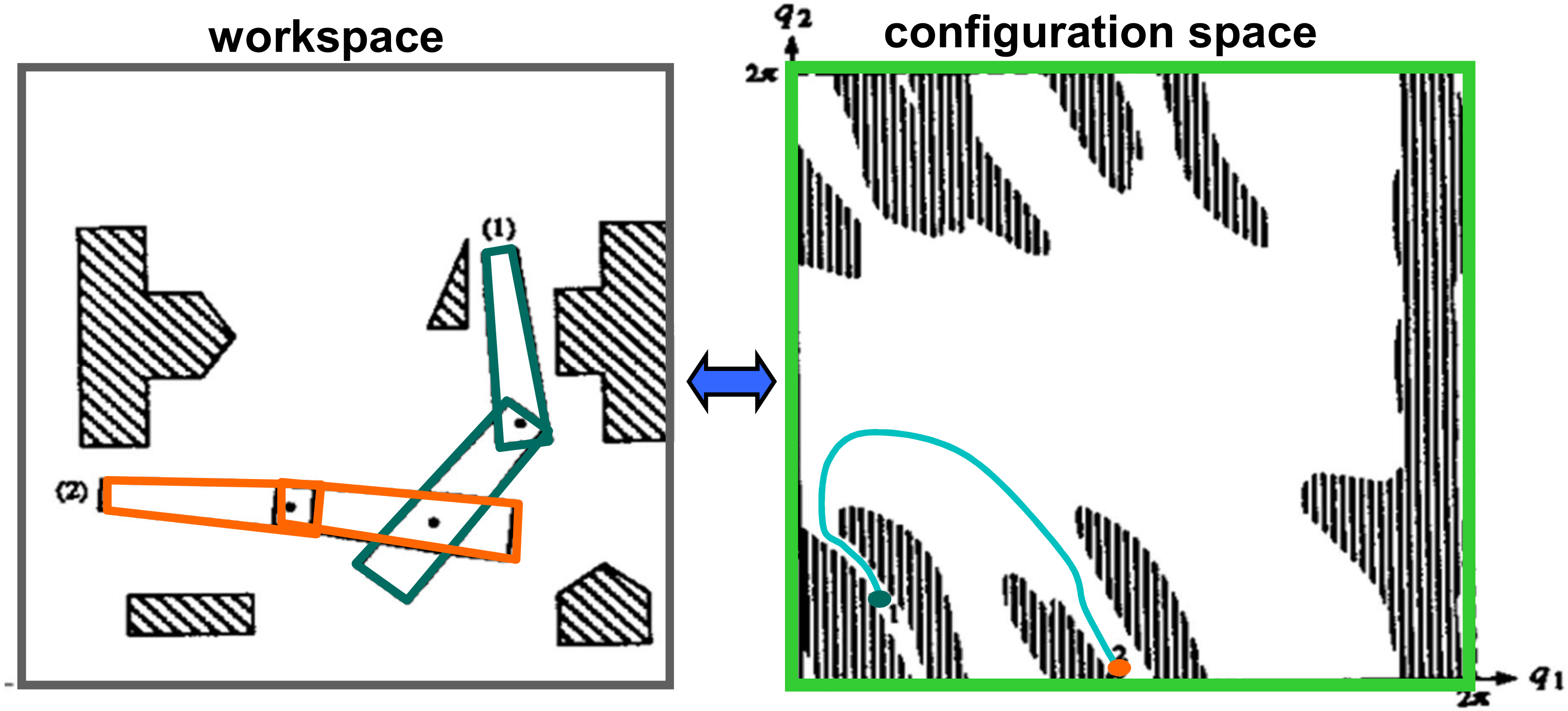




# Configuration space

- Definitions and examples
- Obstacles
- Paths
- Metrics

# Paths in the configuration space




- **A path in $C$ is a continuous curve connecting two configurations $q$ and $q'$:**

$$\tau : s \in [0,1] \rightarrow \tau(s) \in C$$

**such that $\tau(0) = q$ and $\tau(1)=q'$.**

# Constraints on paths

- A **trajectory** is a path parameterized by time:

$$\tau : t \in [0, T] \rightarrow \tau(t) \in C$$

- Constraints
  - Finite length
  - Bounded curvature
  - Smoothness
  - Minimum length
  - Minimum time
  - Minimum energy
  - …


27

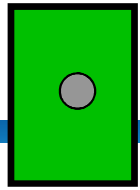
# Free Space Topology

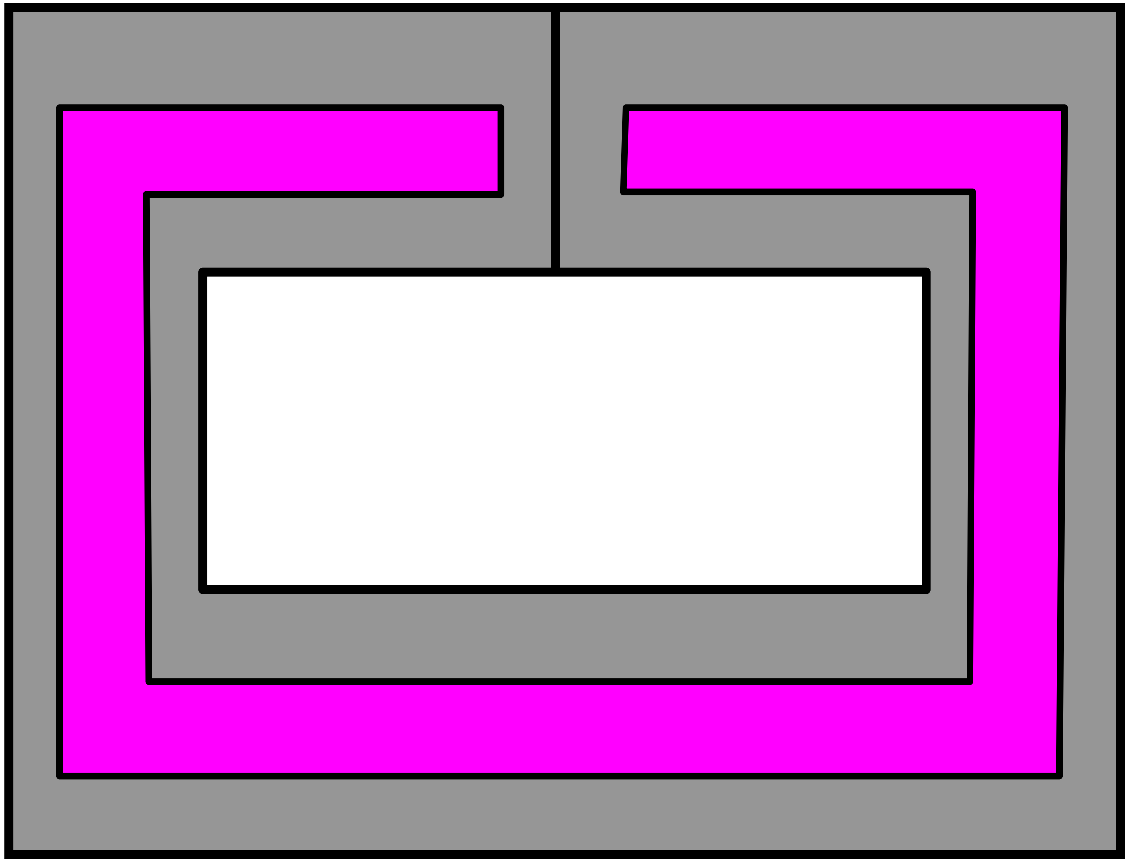
- A **free** path lies entirely in the free space $F$
  - The moving object and the obstacles are modeled as closed subsets, meaning that they contain their boundaries.
  - One can show that the C-obstacles are closed subsets of the configuration space $C$ as well
  - Consequently, **the free space $F$ is an open subset of $C$**

# Semi-Free Space

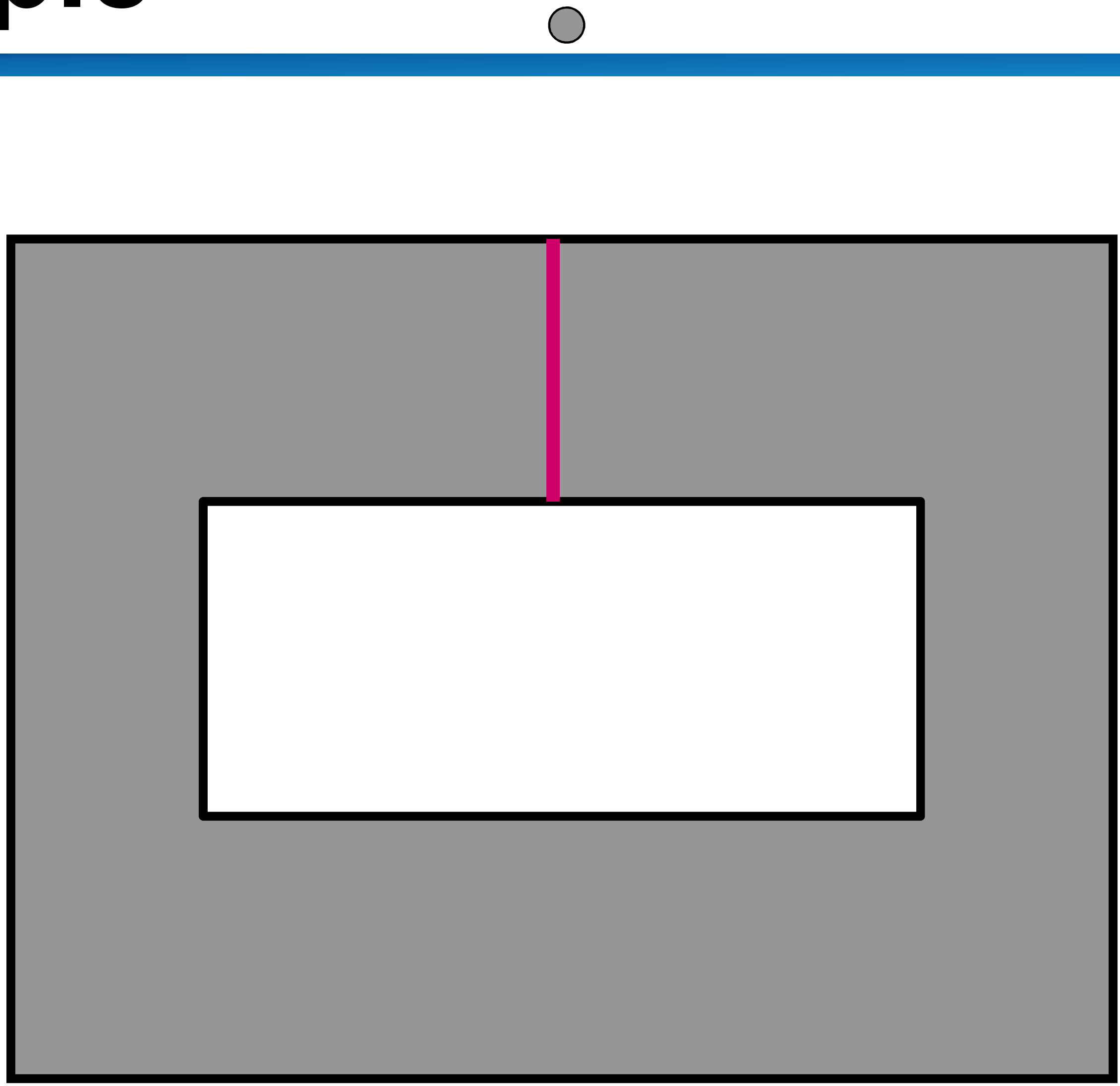
- **A configuration $q$ is semi-free if the moving object placed $q$ touches the boundary, but not the interior of obstacles.**
  - **Free, or**
  - **In contact**
- **The semi-free space is a closed subset of $C$**

# Example





# Example



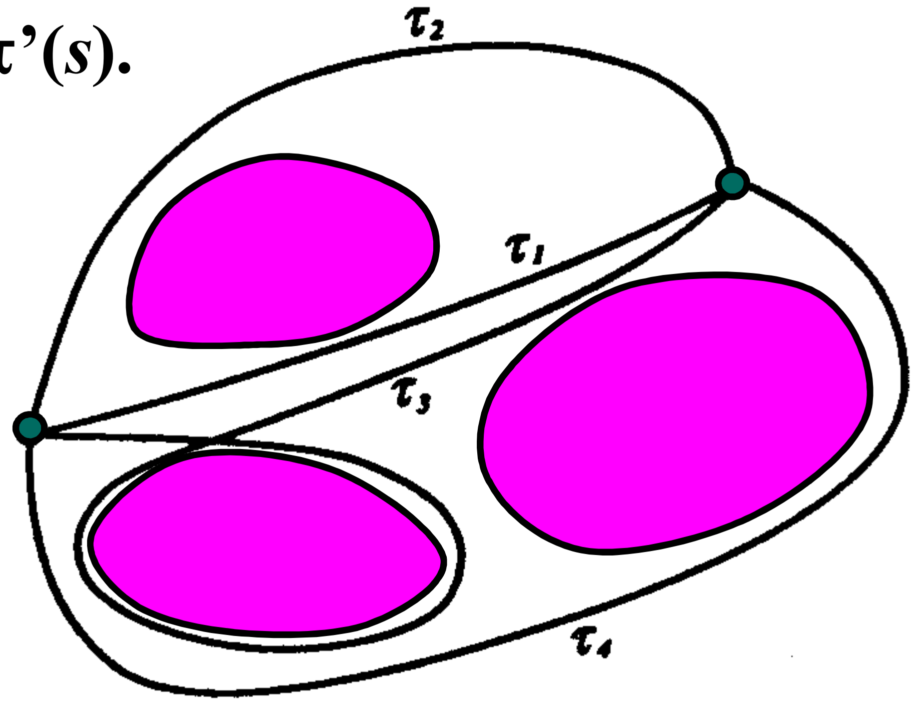
# Homotopic Paths

- **Two paths τ and τ' (that map from U to V) with the same endpoints are homotopic if one can be continuously deformed into the other:**

$$h : U \times [0,1] \rightarrow V$$

**with $h(s,0) = \tau(s)$ and $h(s,1) = \tau'(s)$.**

- **A homotopic class of paths contains all paths that are homotopic to one another**



# Connectedness of C-Space

- $C$ is **path-connected** if every two configurations can be connected by a path.
- $C$ is **simply-connected** if any two paths connecting the same endpoints are homotopic.
  Examples: $R^2$ or $R^3$
- Otherwise $C$ is multiply-connected.


KAIST

# Configuration space

- Definitions and examples
- Obstacles
- Paths
- Metrics

# Metric in Configuration Space

- **A metric or distance function *d* in a configuration space *C* is a function**

$$d : (q, q') \in C^2 \rightarrow d(q, q') \geq 0$$

**such that**

- $d(q, q') = 0$ **if and only if** $q = q'$,
- $d(q, q') = d(q', q)$,
- $d(q, q') \leq d(q, q'') + d(q'', q')$ .



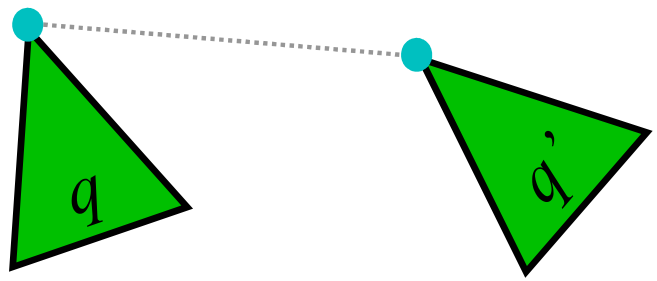
KAIST

# Example

- Robot $A$ and a point $x$ on $A$
- $x(q)$: position of $x$ in the workspace when $A$ is at configuration $q$
- A distance $d$ in $C$ is defined by

$$d(q, q') = \max_{x \in A} \| x(q) - x(q') \|,$$

where $||x - y||$ denotes the Euclidean distance between points $x$ and $y$ in the workspace.



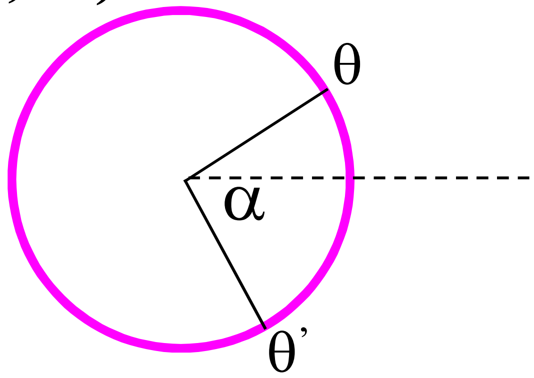
# $L_p$ Metrics

$$d(x, x') = \left( \sum_{i=1}^{n} | x_i - x_i' |^p \right)^{1/p}$$

- $L_2$: Euclidean metric
- $L_1$: Manhattan metric
- $L_\infty$: Max $(| x_i – x_i' |)$

# Examples in $R^2$ x $S^1$

- **Consider** $R^2$ **x** $S^1$
  - $q = (x, y, \theta)$, $q' = (x', y', \theta')$ **with** $\theta, \theta' \in [0,2\pi)$
  - $\alpha = \min \{ |\theta - \theta'|, 2\pi - |\theta - \theta'| \}$

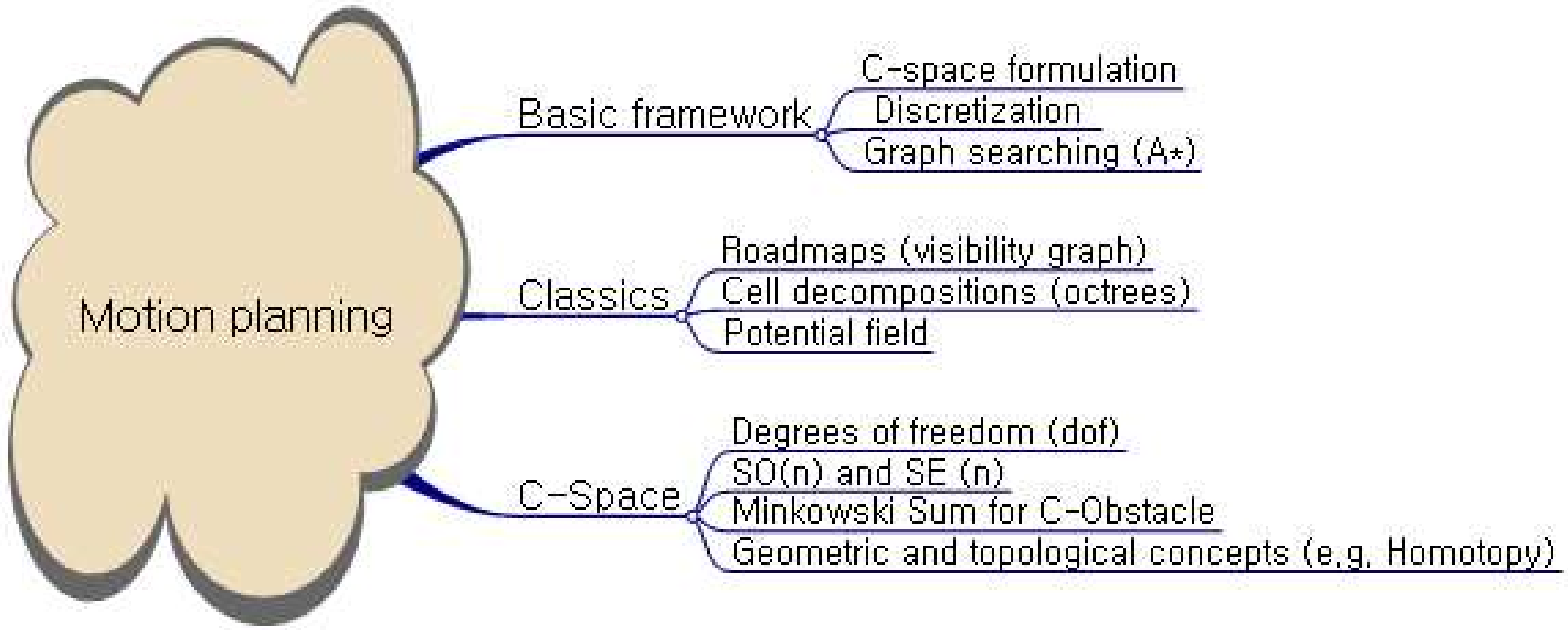



- $d(q, q') = \mathrm{sqrt}( (x-x')^2 + (y-y')^2 + \alpha^2 ) )$

# Class Objectives were:

- **Configuration space**
  - **Definitions and examples**
  - **Obstacles**
  - **Paths**
  - **Metrics**

# Summary


Motion planning
Basic framework
C-space formulation
Discretization
Graph searching (A*)
Classics
Roadmaps (visibility graph)
Cell decompositions (octrees)
Potential field
C-Space
Degrees of freedom (dof)
SO(n) and SE (n)
Minkowski Sum for C-Obstacle
Geometric and topological concepts (e.g. Homotopy)

# Next Time….

- **Collision detection and distance computation**

# Homework

- **Submit summaries of 2 ICRA/IROS/RSS/CoRL/TRO/IJRR papers**
- **Go over the next lecture slides**
- **Come up with 3 questions before the mid-term exam**