# 机器人导论：规划与集群

- Lecture

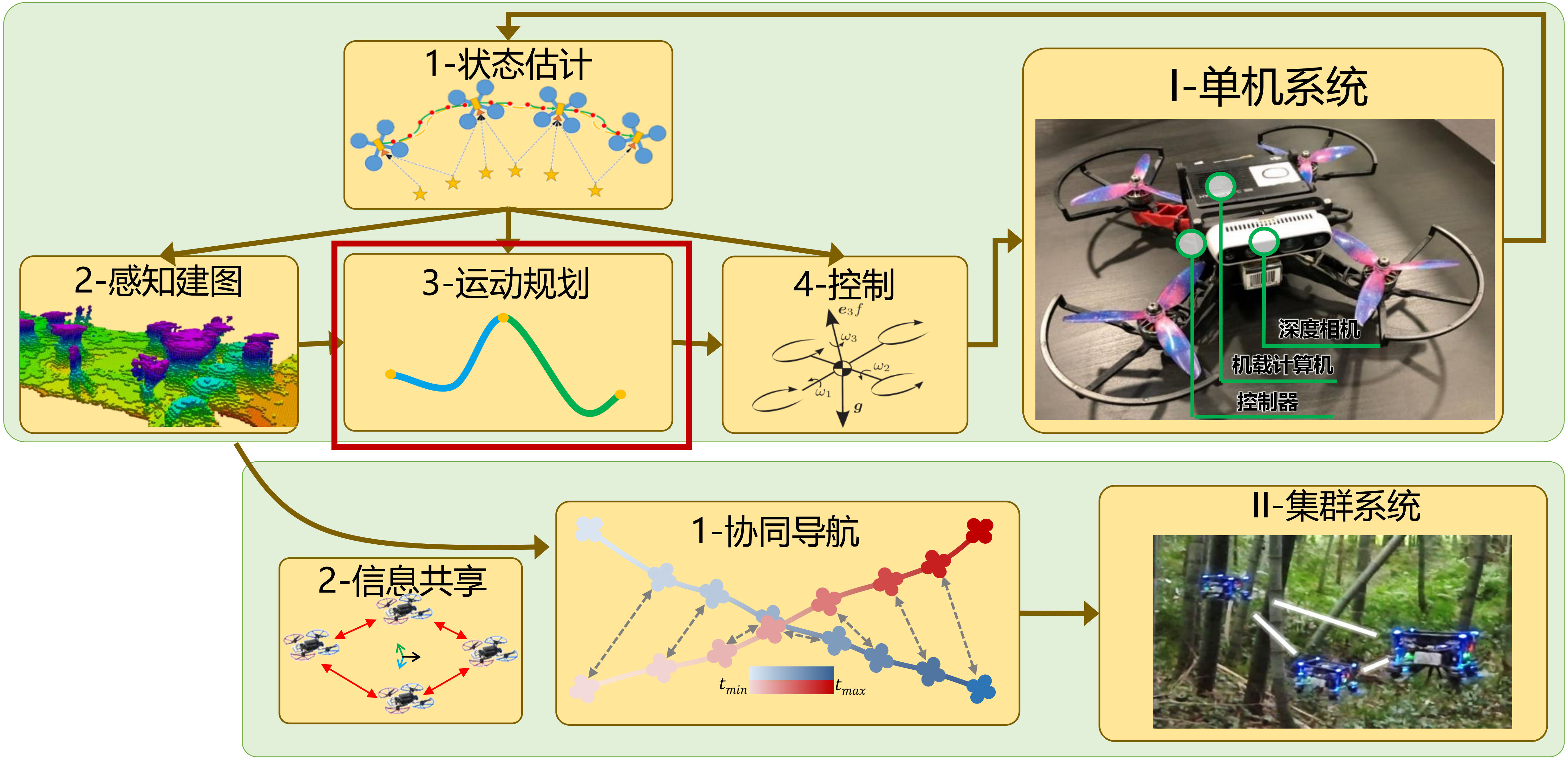
## 课程内容复习

高飞

浙江大学 控制学院

# 自主导航软件框架

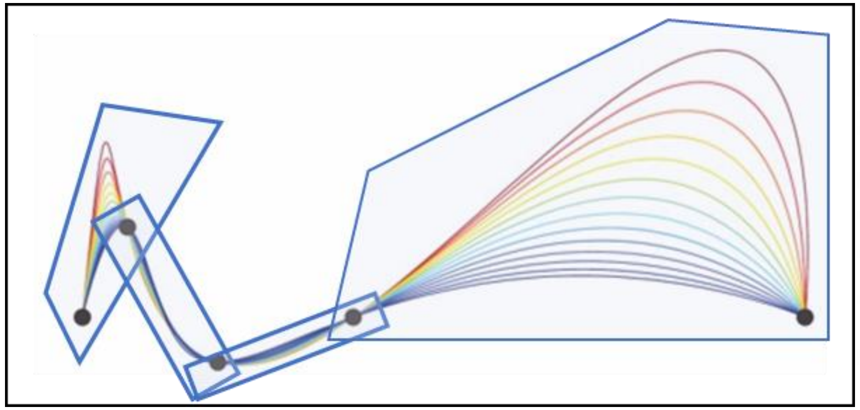


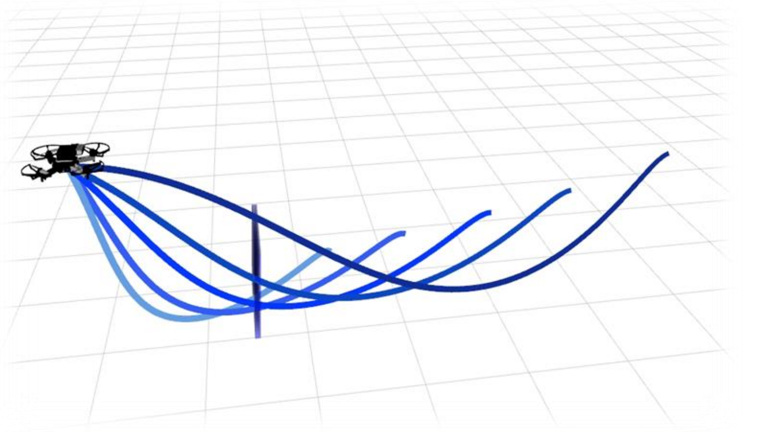

1-状态估计
I-单机系统
深度相机
机载计算机
控制器
2-感知建图
3-运动规划
4-控制
2-信息共享
1-协同导航
$t_{min}$
$t_{max}$
II-集群系统

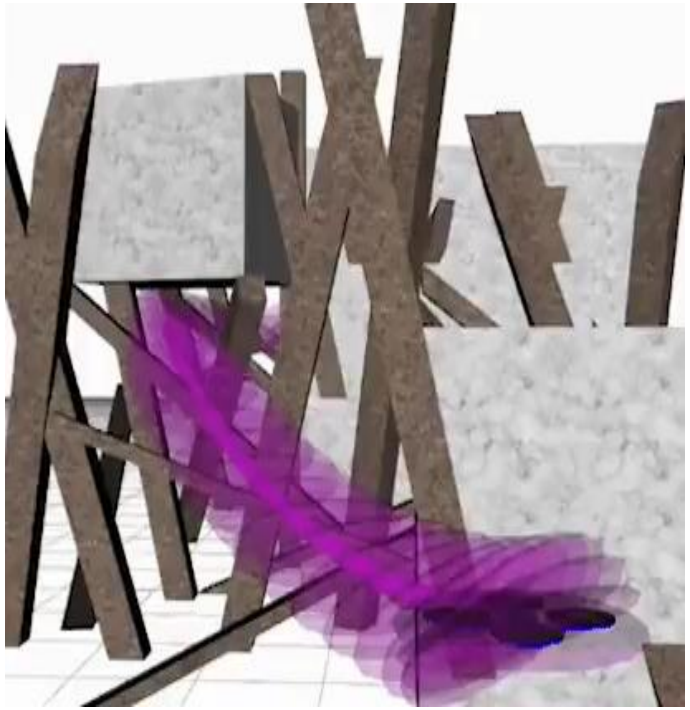
# 什么是运动规划?

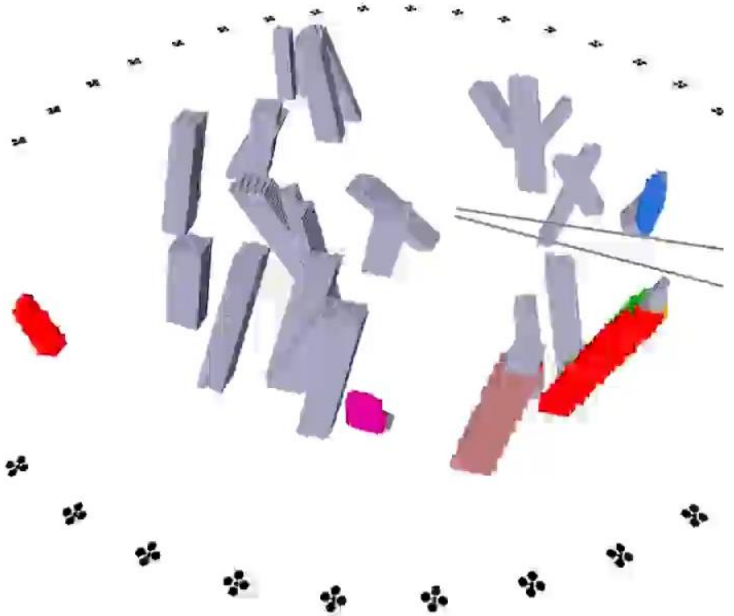


- **基本要求**
  - 安全：避免碰撞
  - 光滑性：节能、平稳
  - 动力学可行性：可执行、可控
- **通用运动规划方法**
  - 前端-路径搜索
    - 低维
    - 离散空间
    - 搜索初始安全路径
  - 后端-轨迹优化
    - 高维
    - 连续空间
    - 生成可执行轨迹



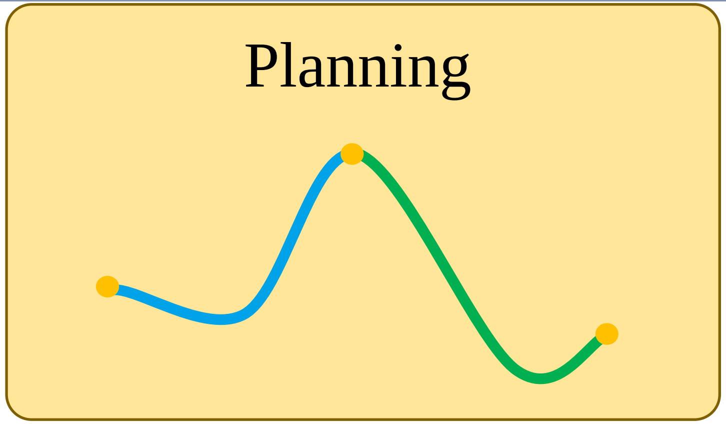


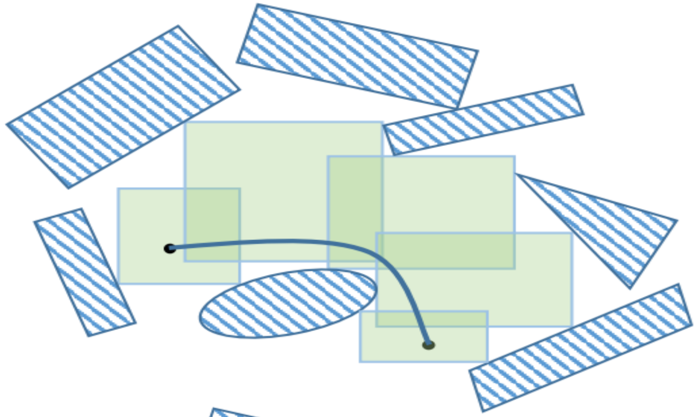


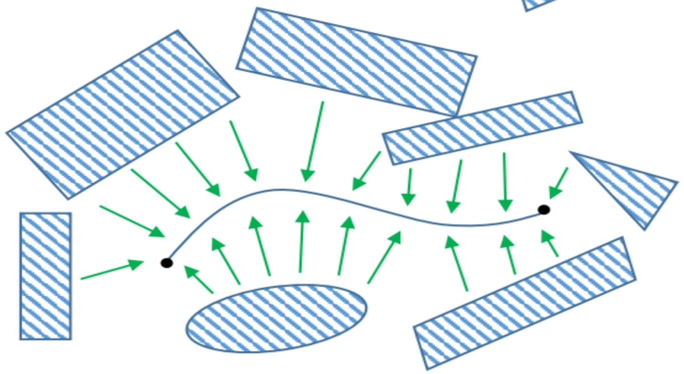


# 什么是运动规划?

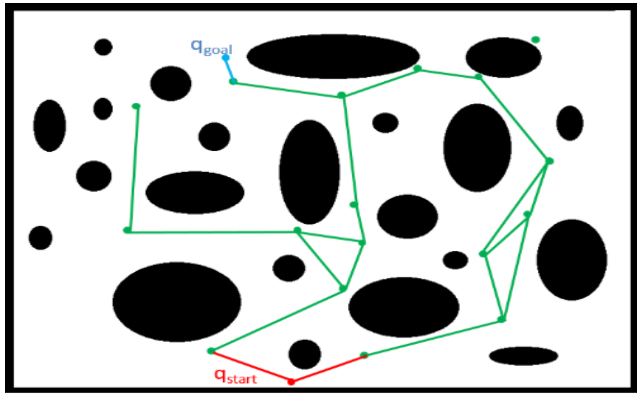




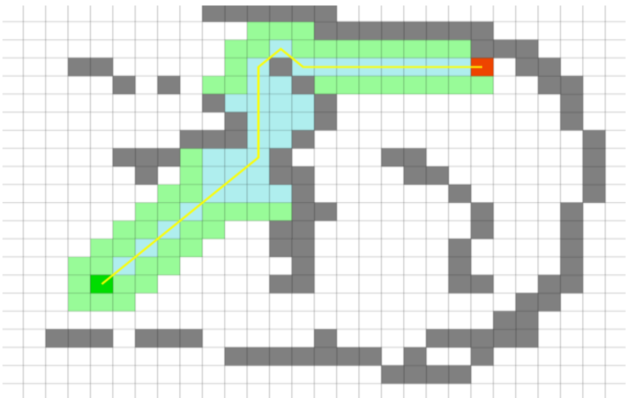





## 前端：路径搜索



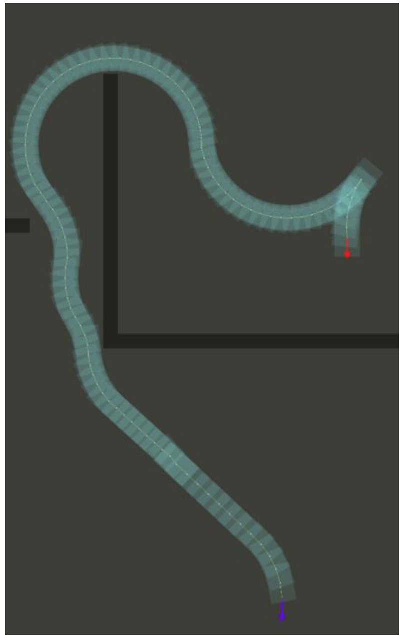

**Search-based Method**



**Sampling-based methods**

## Kinodynami

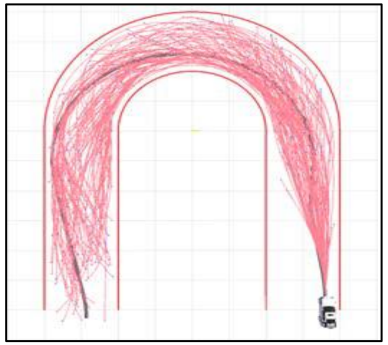
**Kinematic + Dynamic**



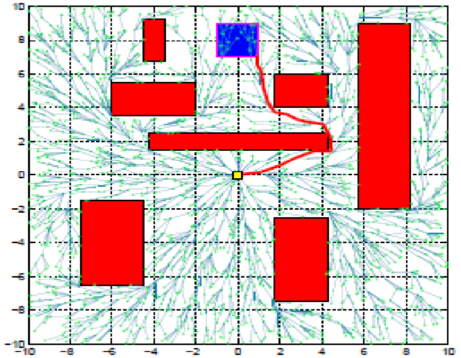
## 后端：轨迹优化

# 前端-路径搜索

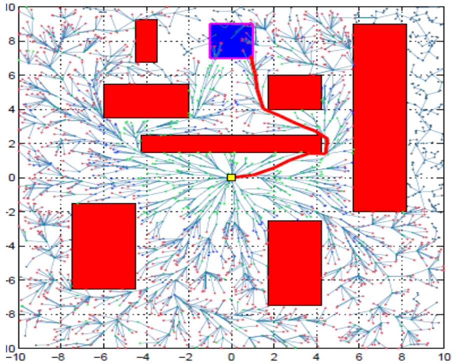
- **基于搜索的方法**
  - Graph Search Basis
  - Dijkstra and A*
  - Jump Point Search (JPS)
- **基于采样的方法**
  - Probabilistic Road Map (PRM)
  - Rapidly-exploring Random Tree (RRT)
  - Optimal Sampling-based Methods
  - Advanced Sampling-based Methods
- **满足动力学要求的路径规划**
  - State-state Boundary Value Optimal Control Problem
  - State Lattice Search
  - Kinodynamic RRT*
  - Hybrid A*



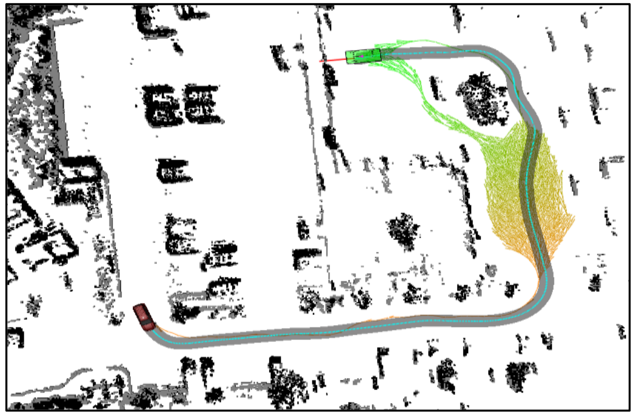
RRT



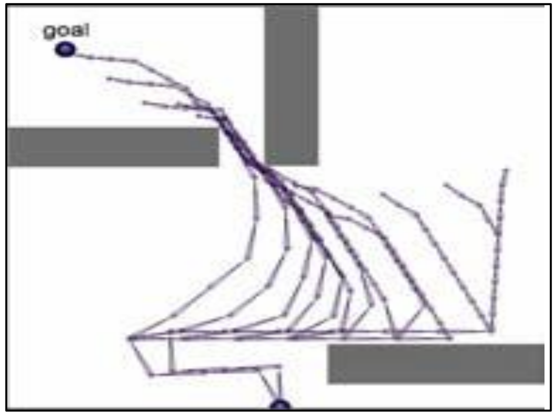
RRT*



RRT#



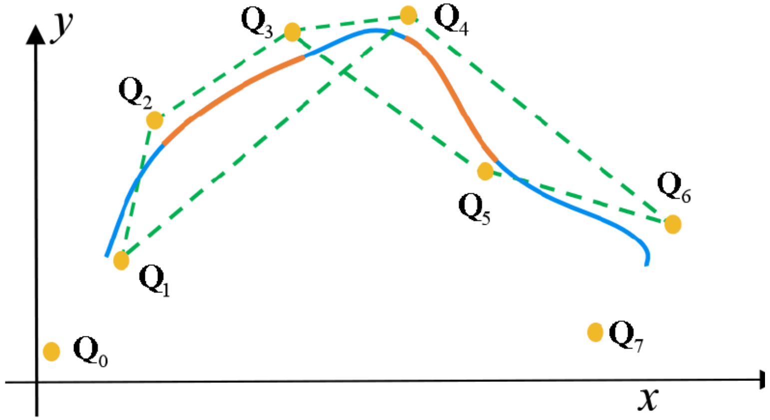
Hybrid A*



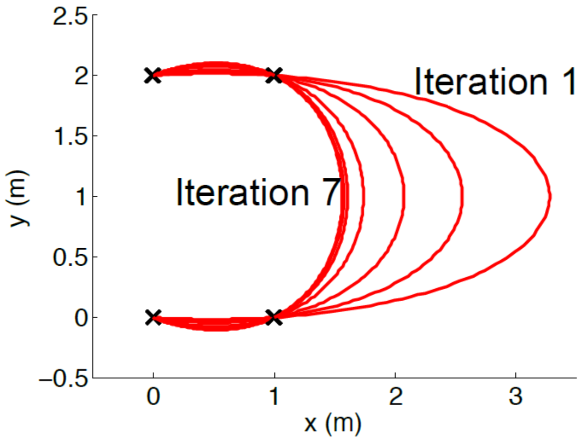

ARA*

# 后端-轨迹优化

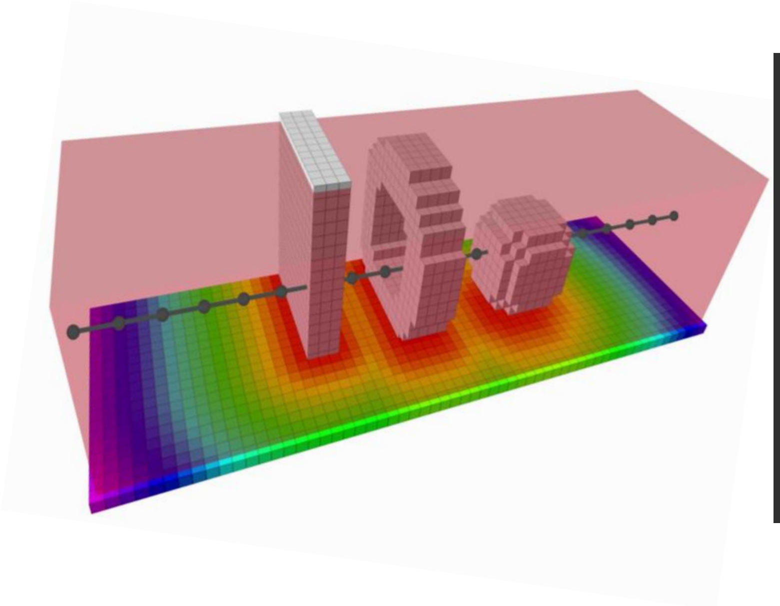
- **Minimum Snap轨迹优化**
  - ➢ Differential Flatness
  - ➢ Minimum Snap Optimization
  - ➢ Closed-form Solution to Minimum Snap
  - ➢ Time Allocation
  - ➢ Implementation in Practice
- **硬约束与软约束轨迹优化**
  - ➢ Soft Constrained Trajectory Optimization
  - ➢ Hard Constrained Trajectory Optimization

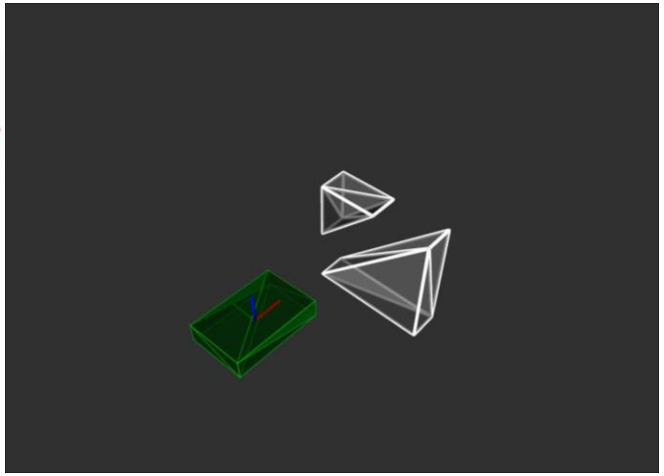



B-spline

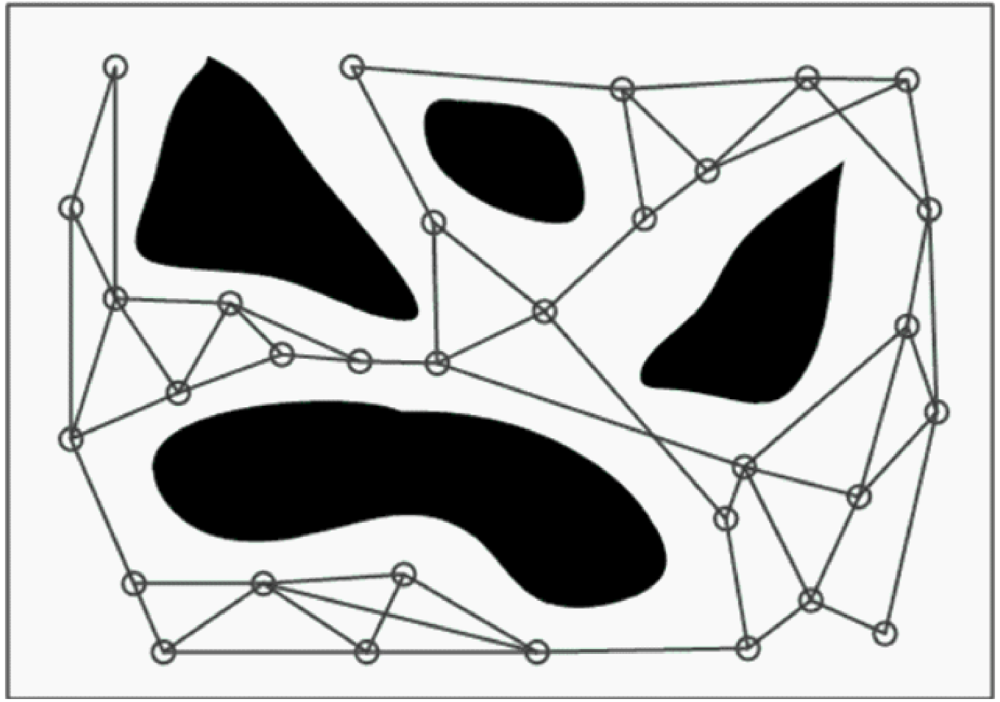


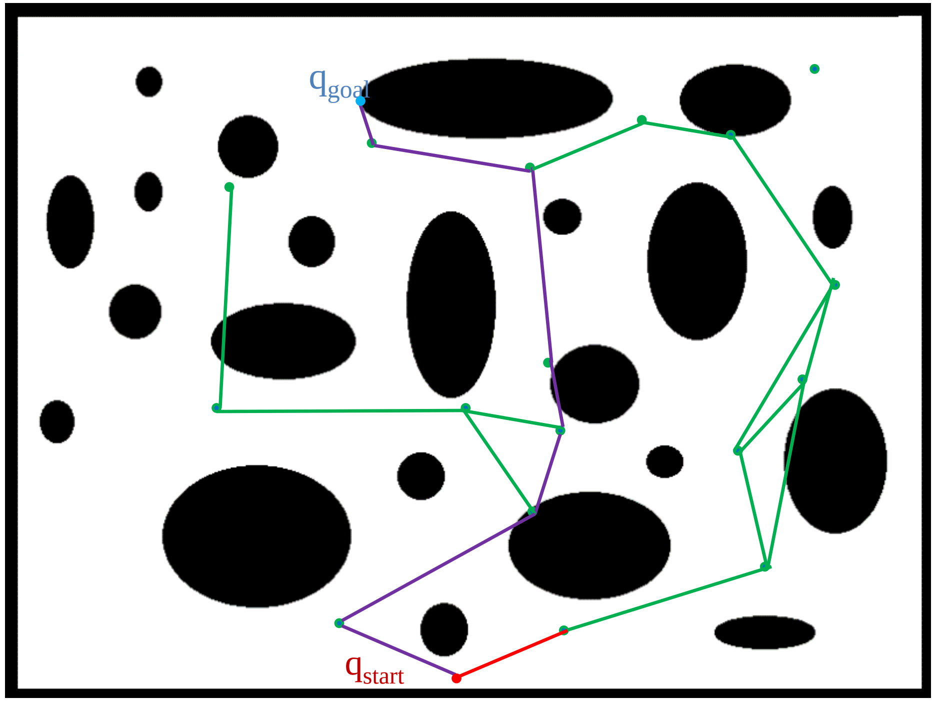

Minimum Snap





Probabilistic Roadmap (PRM)

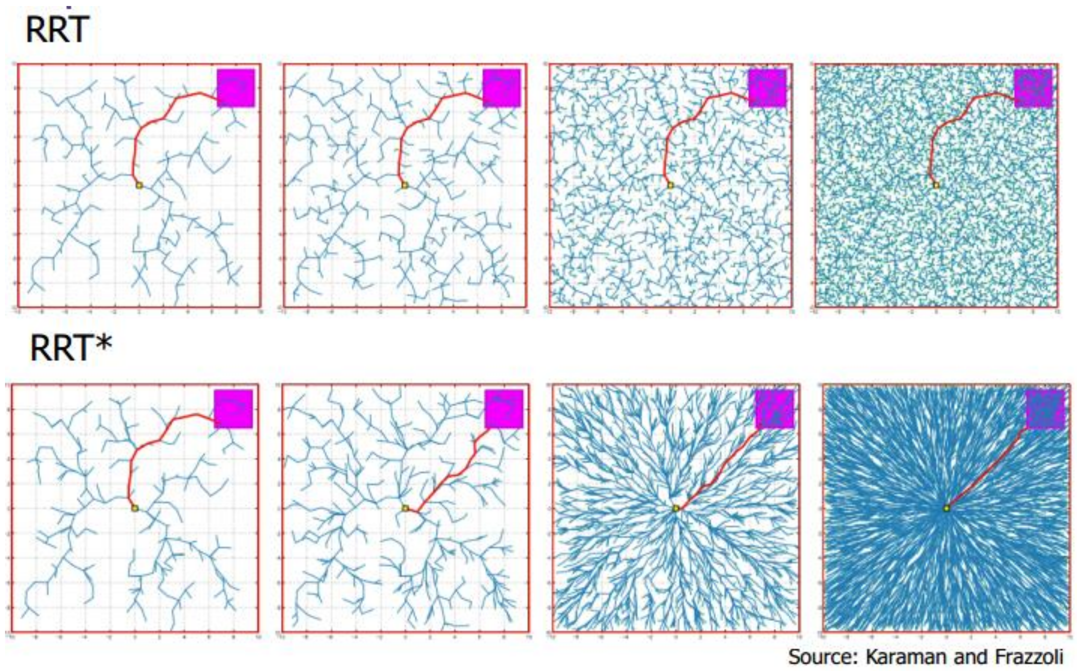


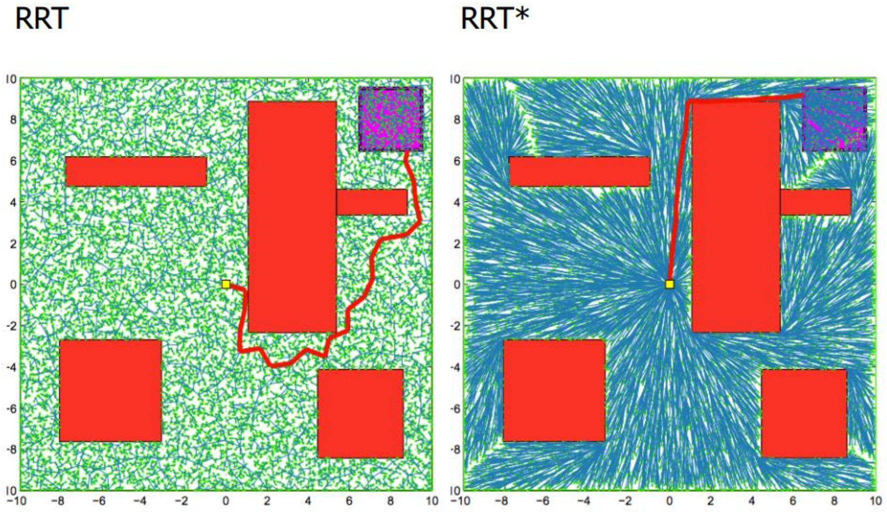



*C-space*

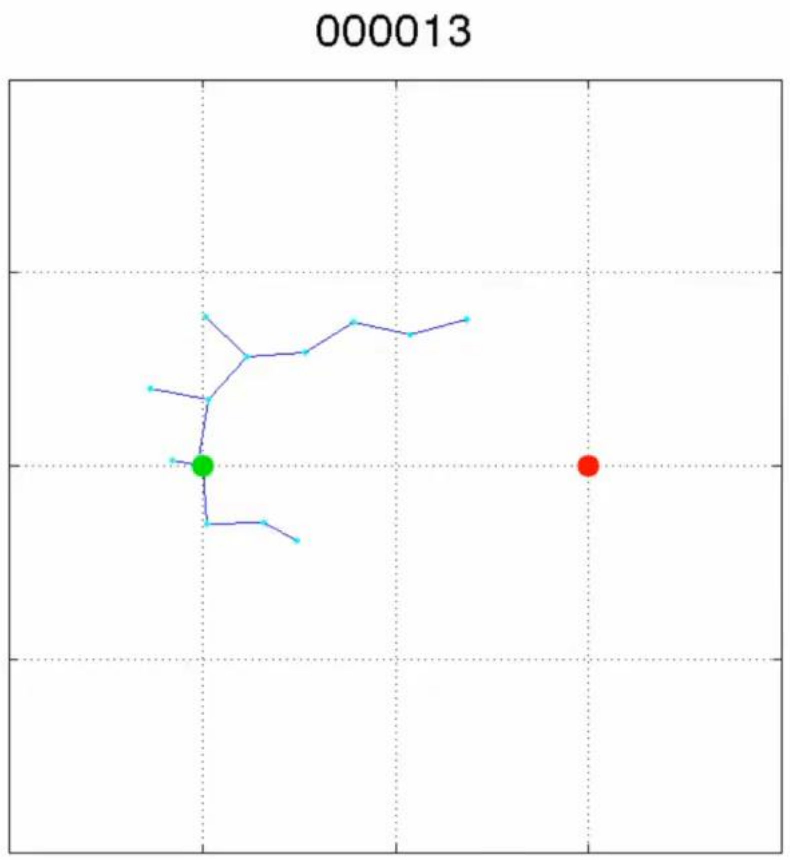
# 基于采样的方法

RRT* vs RRT



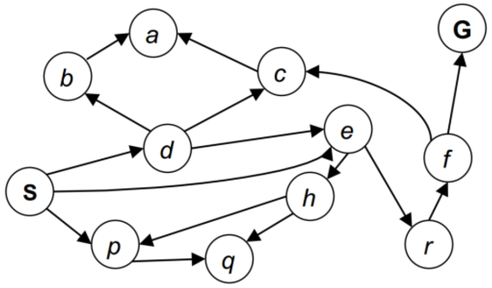



Informed RRT*

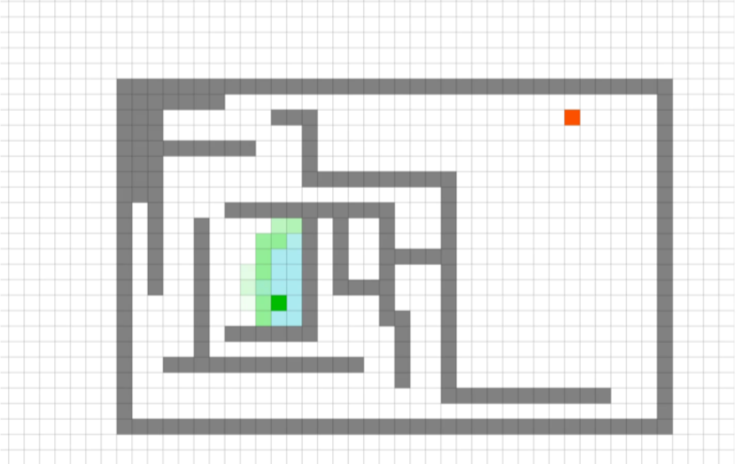


# 基于搜索的方法

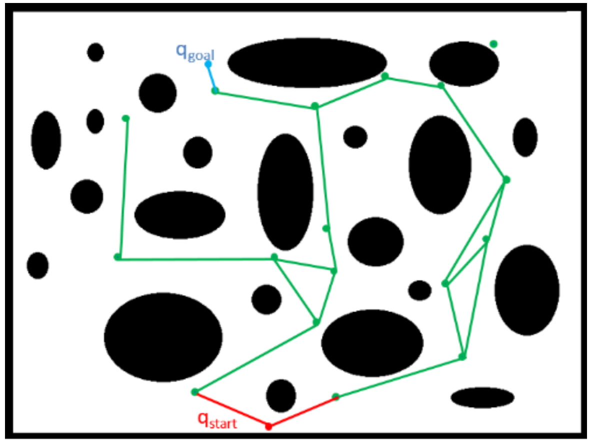
- 对于每个搜索问题，都有一个相应的状态空间图
- 图中节点之间的连通性由（有向或无向）边表示




*Ridiculously tiny search graph for a tiny search problem*



Grid-based graph: use grid as vertices and grid connections as edges



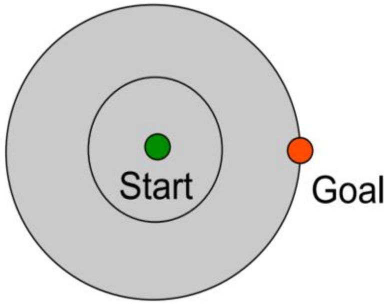

The graph generated by probabilistic roadmap (PRM)

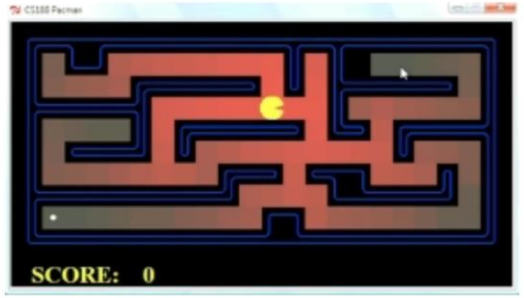
# 基于搜索的方法



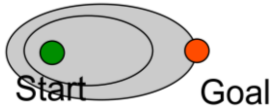
## Dijkstra’s vs. A*

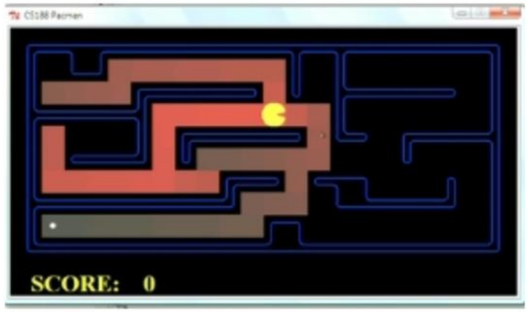
- Dijkstra算法朝各个方向探索



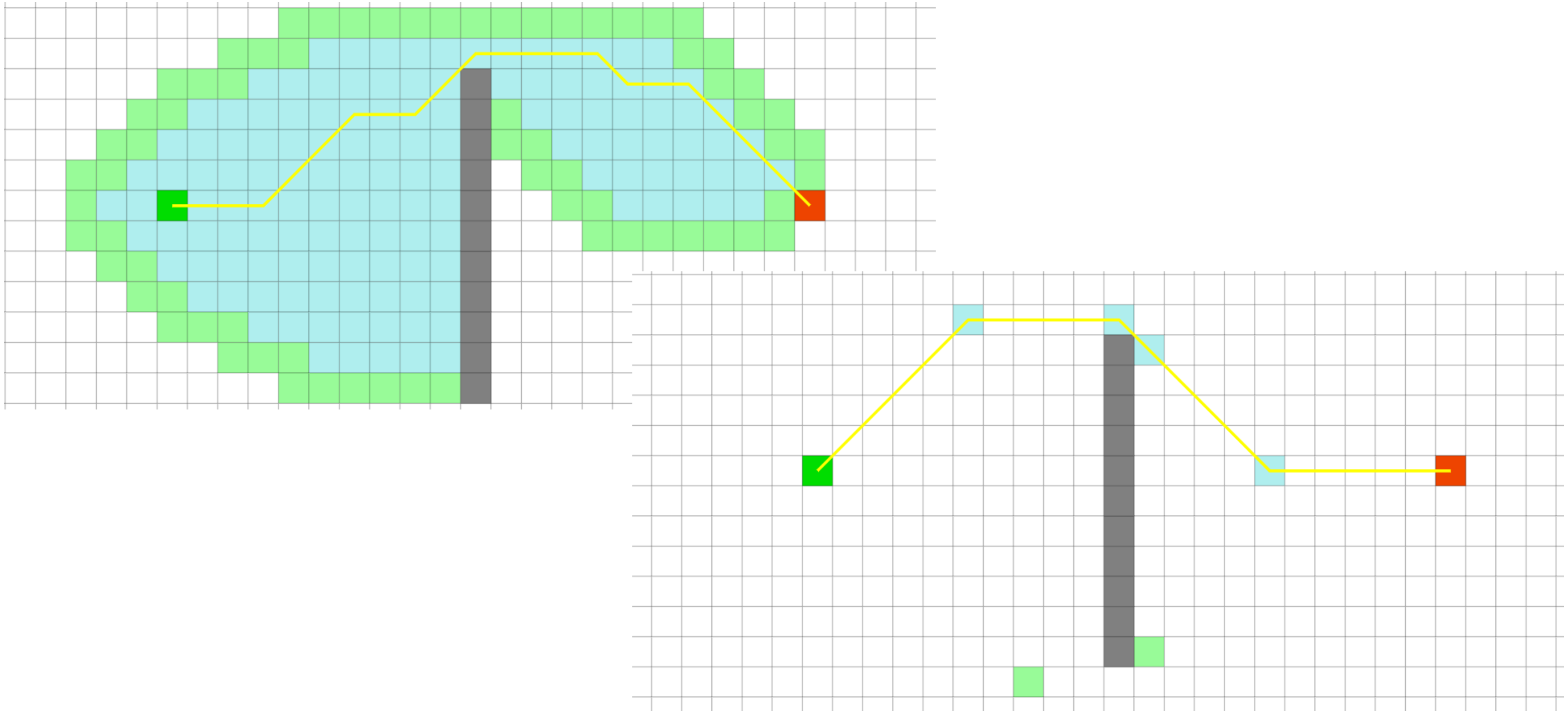




- A*算法主要朝着目标点方向探索

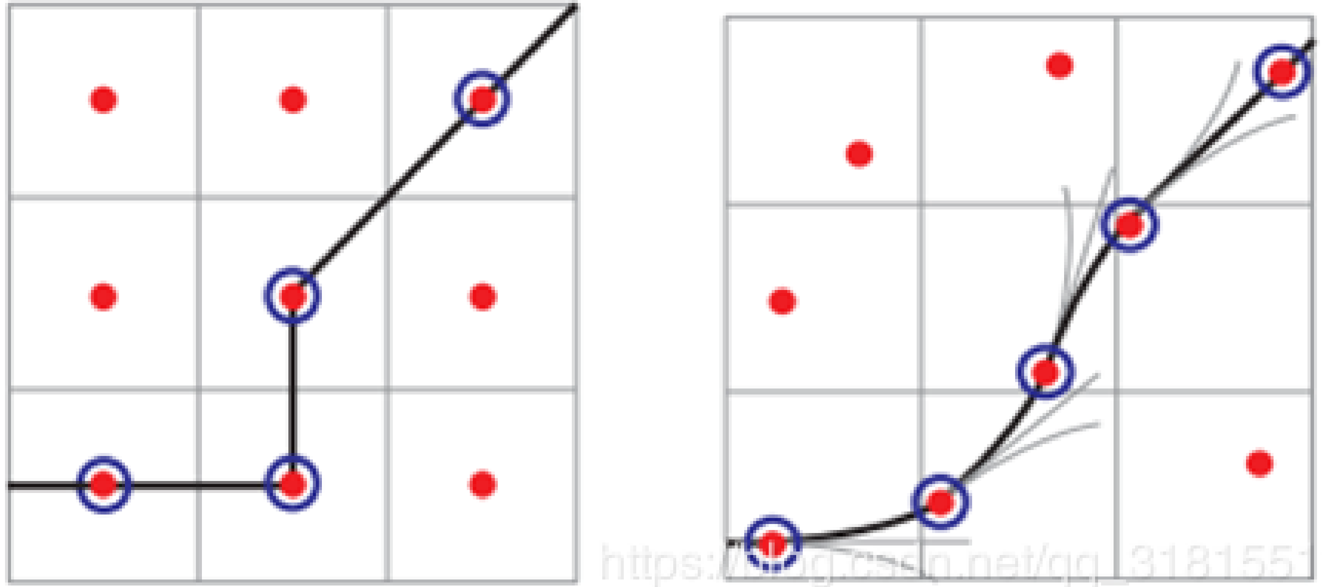


A* vs. JPS



# Hybrid A*



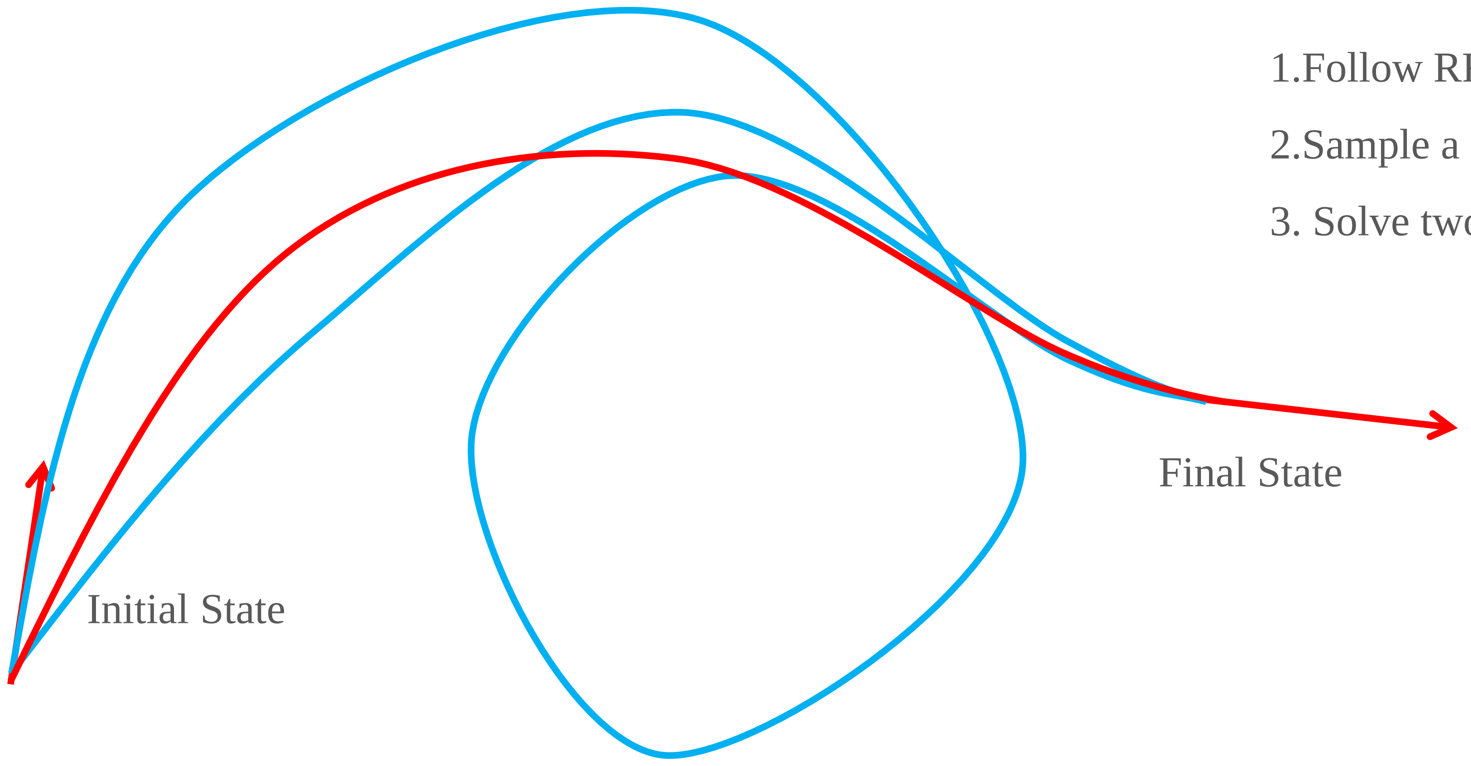
1.Follow A* algorithm

2. Forward simulate states with different discrete control inputs

3. Keep only 1 state in each grid

Discrete control

## Kinodynamic RRT*



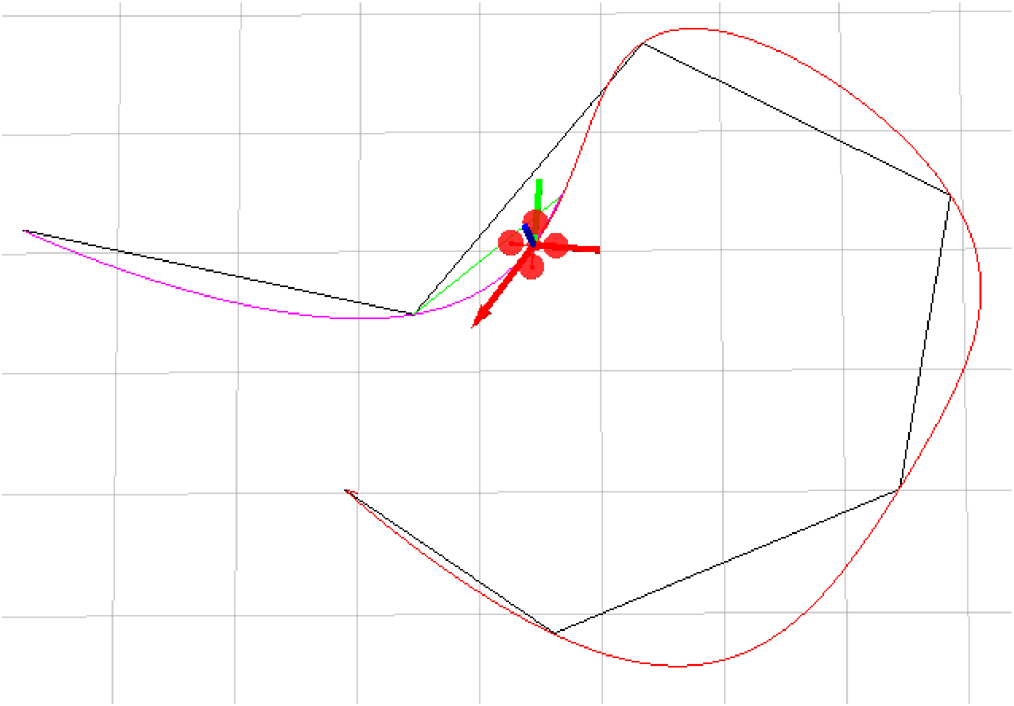

1.Follow RRT* algorithm

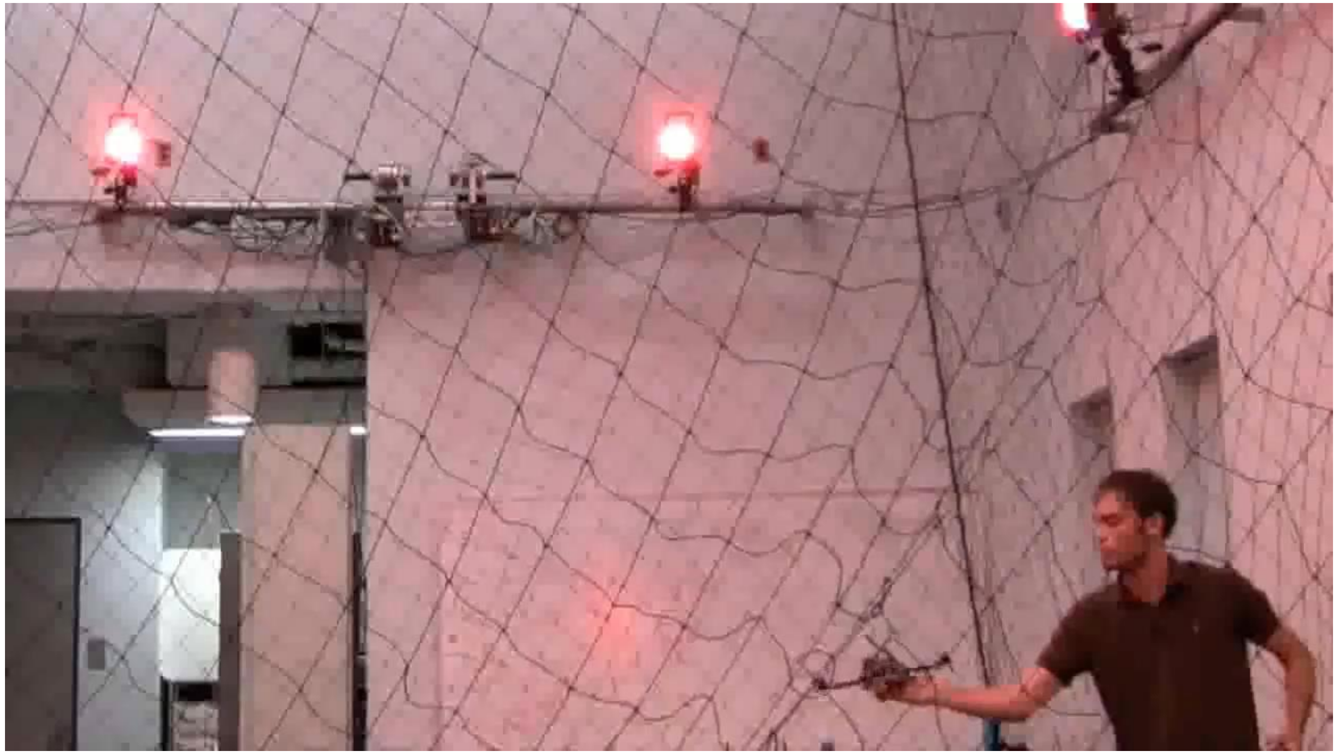
2.Sample a random state

3. Solve two state boundary optimal control problem

Discrete state

Basic Minimum-snap



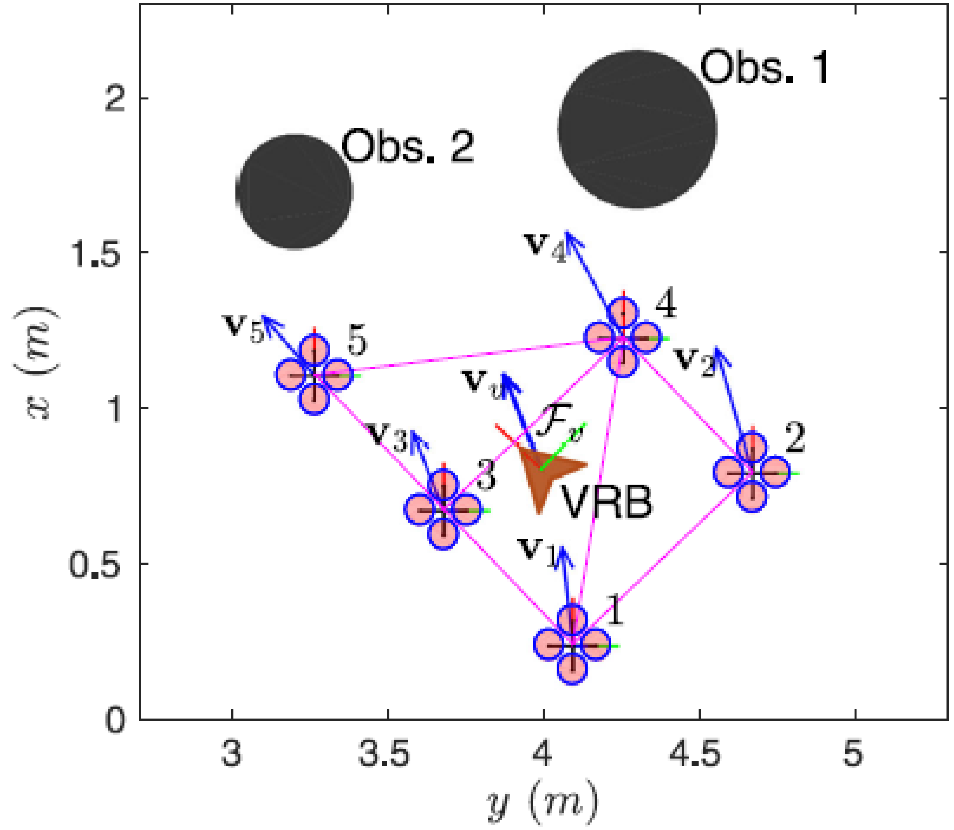


# 基于Virtual Structures的编队控制

Zhou.D, Wang.Z , ‘Agile Coordination and Assistive Collision Avoidance for Quadrotor Swarms Using Virtual Structures’, **IEEE TRO**, 2018

➢ 核心思想

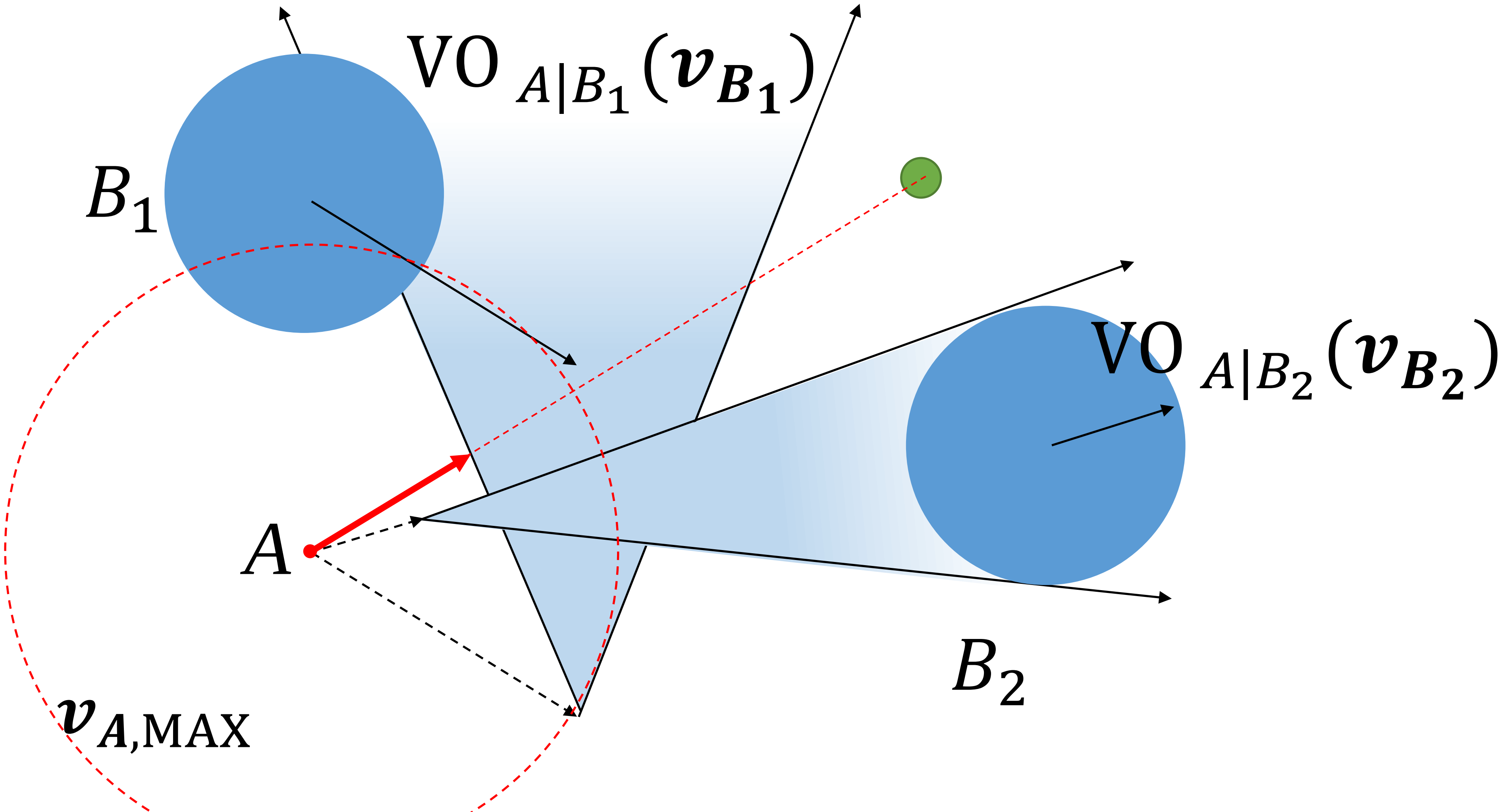
- 集群编队结构表示：将整个集群用virtual structure表示为一个世界坐标系下的整体（virtual rigid body, VRB）；
- 多目标需求：基于势场法表示集群中每架无人机编队保持、相互躲避、障碍物避障的需求；
- 控制：在VRB坐标系下统一上述各个势场得到相应的控制指令。




世界坐标系下VRB表示的集群编队

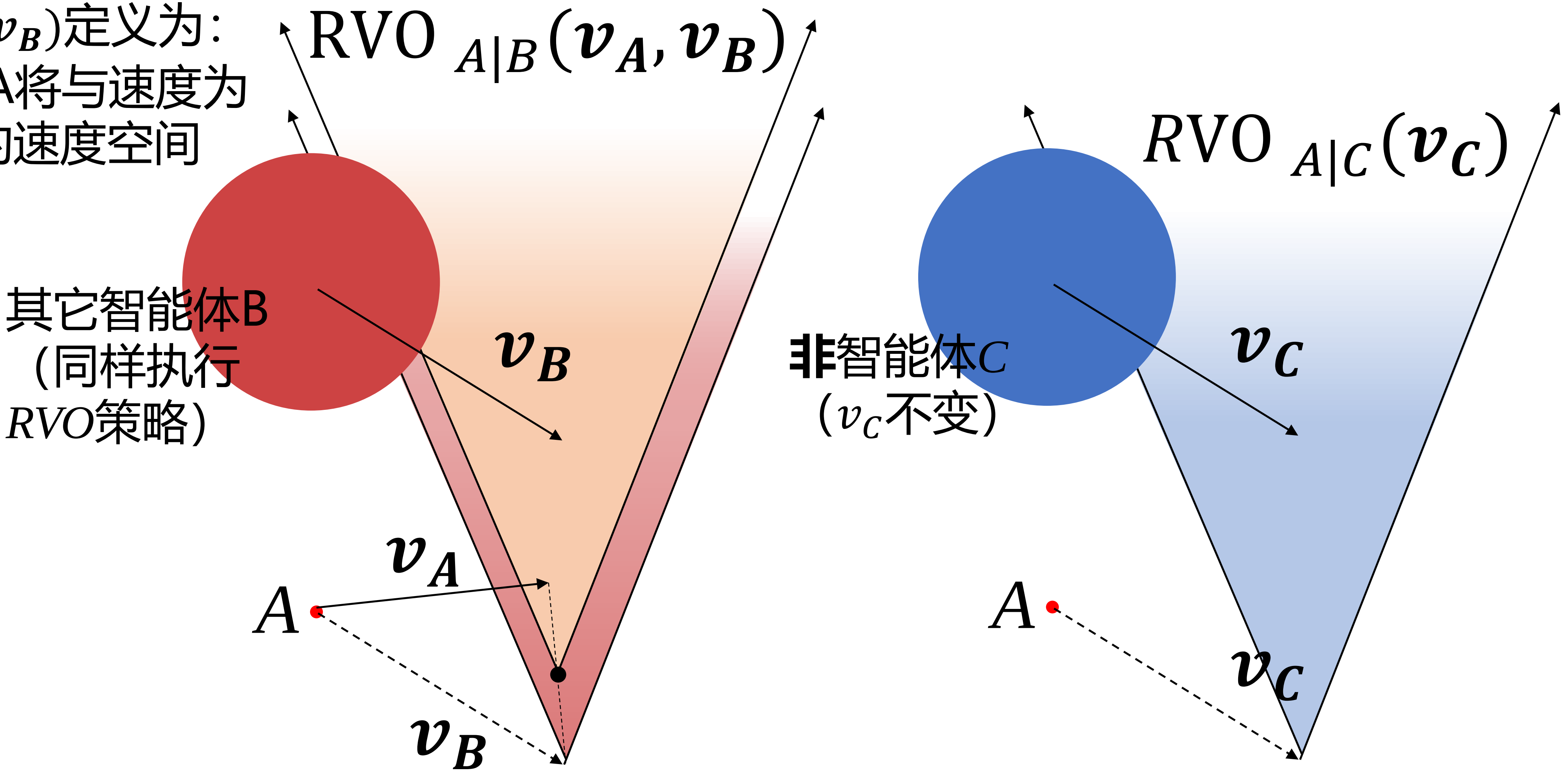
# VO (Velocity Obstacle)

针对**多个**移动障碍物的VO



# VO改进：RVO (Reciprocal Velocity Obstacle)

$RVO_{A|B}(\boldsymbol{v_A}, \boldsymbol{v_B})$定义为：
速度为$\boldsymbol{v_A}$的A将与速度为$\boldsymbol{v_B}$的B相撞的速度空间



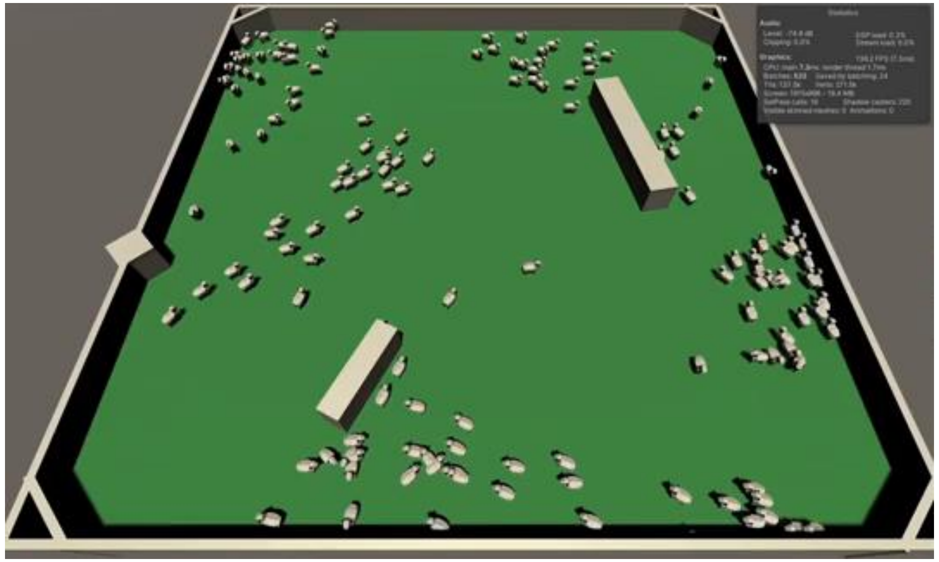
# 基于生物群落模型的集群算法（Flocking models）

**基本思想**：为实现像鸟群一样的一致飞行，每一个体的运动由三股力量（速度）决定：

- 短距离：与邻居、障碍物的排斥速度$\mathbf{v}^{rep}$，越靠近斥力越大；
- 中距离：运动对齐速度$\mathbf{v}^{frict}$，越偏离权重越大；
- 长距离：远方目标的引力$\mathbf{v}^{flock}$，一定范围内维持未定；

执行速度为三类速度的矢量

$$\mathbf{v}^{exe} = \mathbf{v}^{rep} + \mathbf{v}^{frict} + \mathbf{v}^{flock}.$$



**应用难点**：参数繁多且对参数灵敏

**解决办法**：进化算法调参[1]

[1] Vásárhelyi, Gábor, et al. "Optimized flocking of autonomous drones in confined environments." *Science Robotics* 3.20 (2018): eaat3536.

# 基于轨迹规划的集群导航