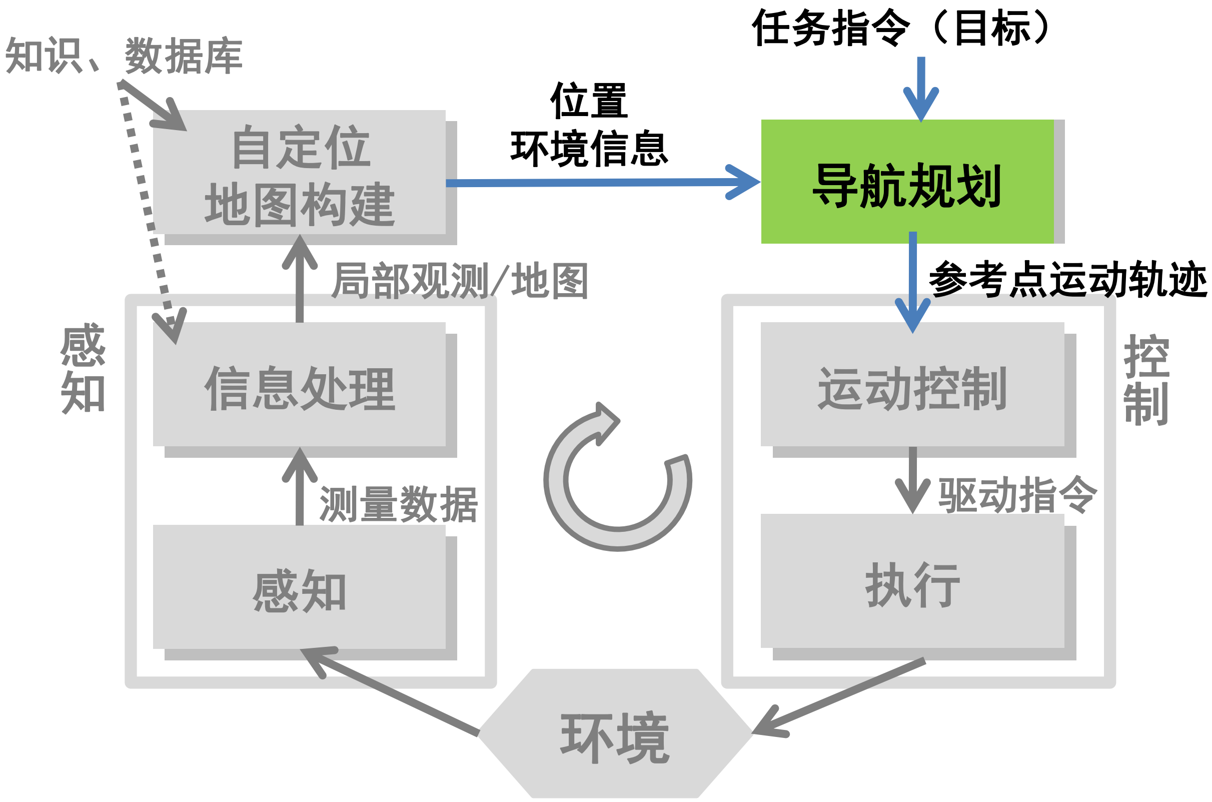
第五讲 避障规划

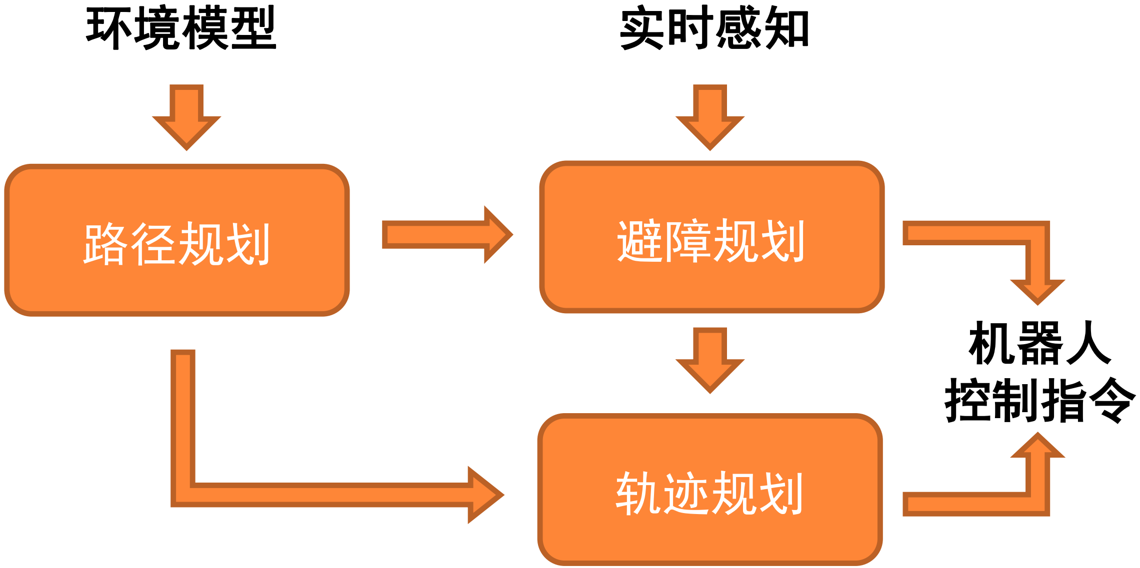
王越

浙江大学 控制科学与工程学院

自主移动机器人一般架构



路径规划、避障规划、轨迹规划三者关系

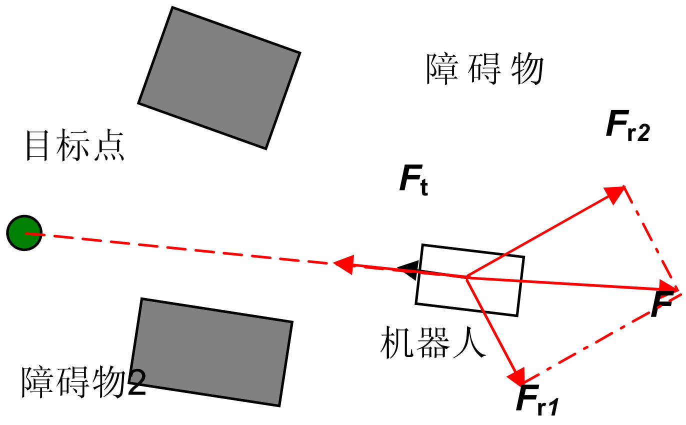


避障规划

- 根据所得到的实时传感器测量信息，规划/调整路径/轨迹，以避免发生碰撞，也称为反应式避障
- 主要方法：
  - 向量势直方图法
  - 动态窗口法

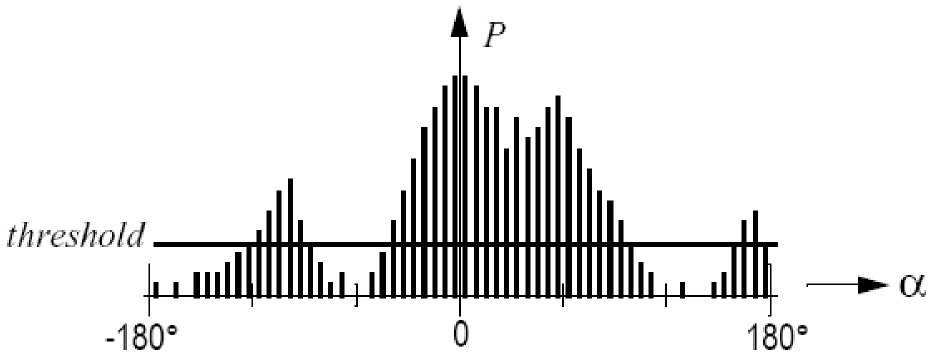
向量势直方图法（VFH: Vector Field Histogram）

- 针对问题：势场法容易陷入局部最优，导致存在振荡、难以通过窄通道

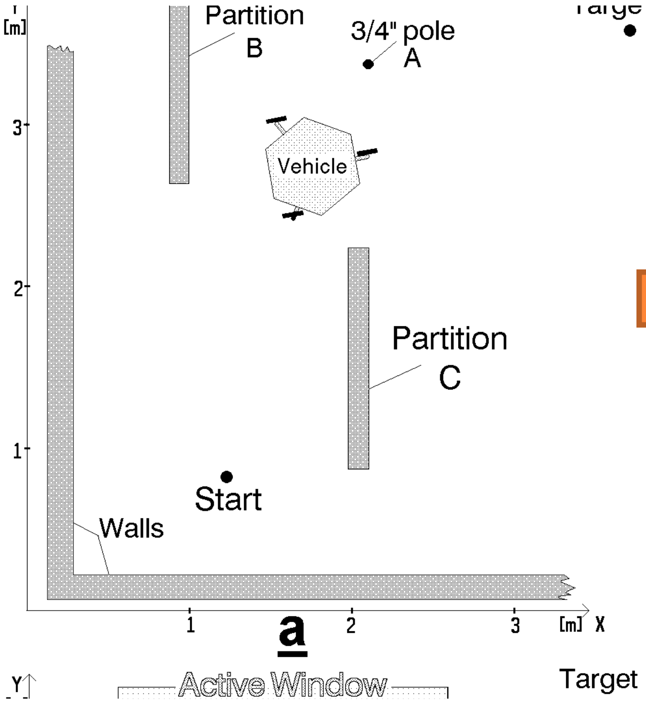


向量势直方图法（VFH: Vector Field Histogram）

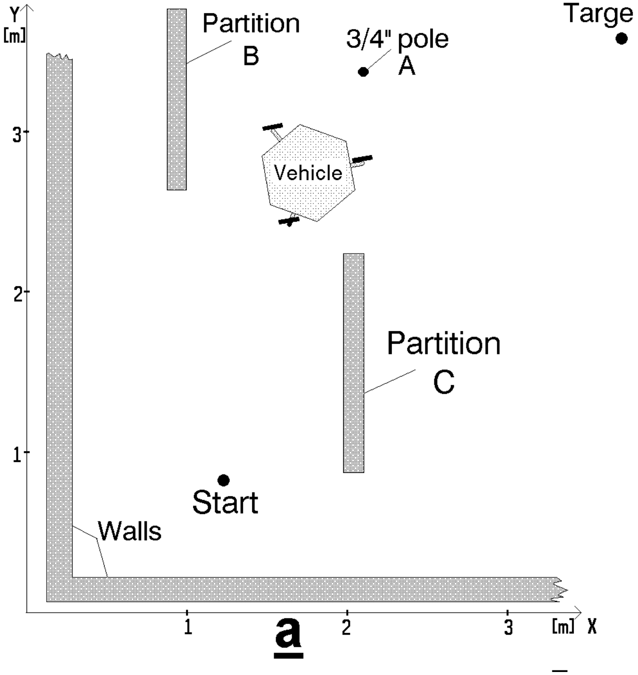
- 1991年美国密歇根大学的Johann Borenstein等提出
- 基本思想：考虑到势场法仅用推斥势来表示障碍物，从而丢失了局部障碍物分布的详细信息，提出根据环境详细栅格地图构建机器人坐标系下障碍物概率直方图，根据概率直方图评估选择最优运动方向



向量势直方图法

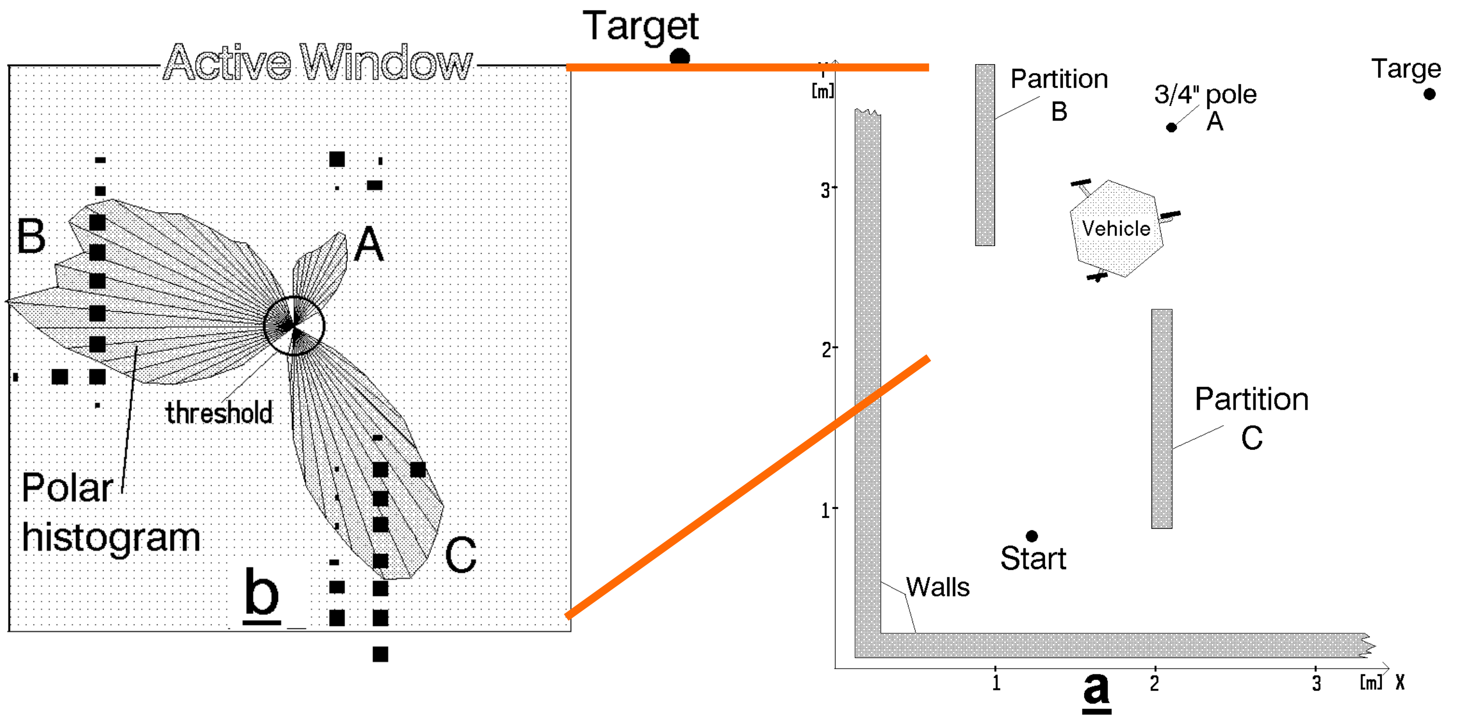


[m]
3
2
1
Partition
B
3/4" pole
A
Targe
Vehicle
Partition
C
Start
Walls
1
a
2
3
[m] X
Y
Active Window
Target



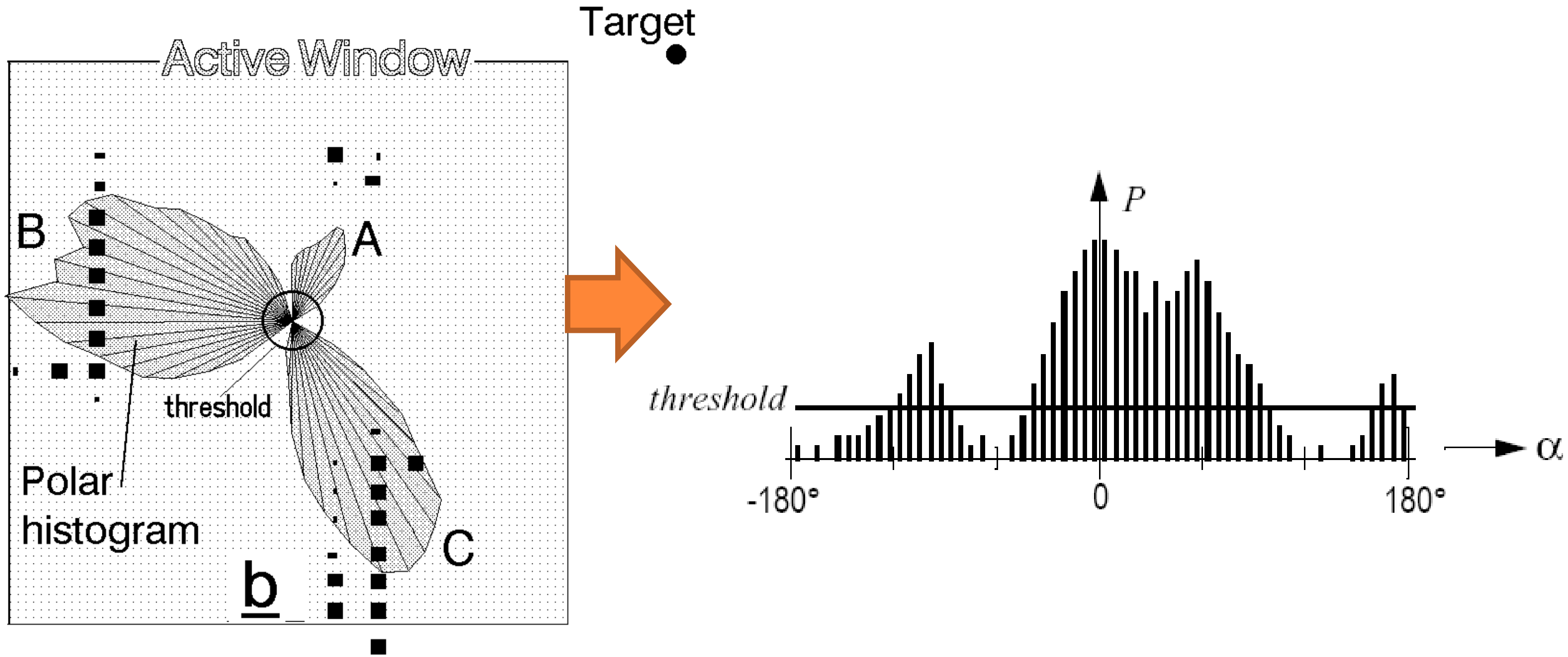
Y
[m]
3
2
1
Partition
B
3/4" pole
A
Targe
Vehicle
Partition
C
Start
Walls
1
a
2
3
[m] X

向量势直方图法



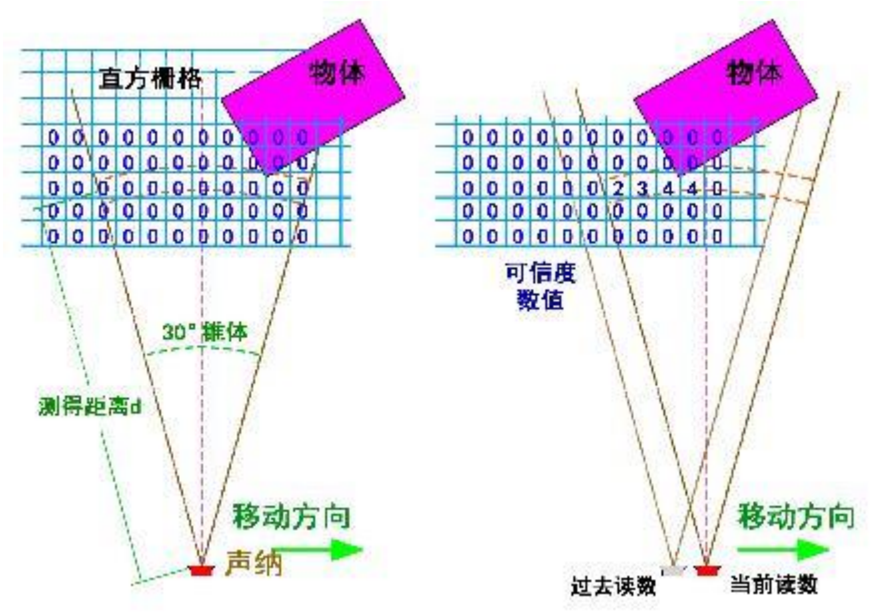
Target
Active Window
B
A
threshold
Polar
histogram
C
b
Partition
B
3/4" pole
A
Targe
Vehicle
Partition
C
Start
Walls
[m]
3
2
1
1
2
3
[m] X
a

向量势直方图法


Target
Active Window
B
A
threshold
Polar
histogram
b
C
P
threshold
α
-180°
0
180°

向量势直方图法实现步骤

1. 构建并维护机器人周围环境的局部栅格地图



直接根据距离传感器检测数据将相关栅格被占值加1

向量势直方图法实现步骤

2. 为每个栅格计算其障碍物向量，距离机器人越近向量越大

向量方向 $\beta_{i,j} = tan^{-1}\dfrac{y_i - y_0}{x_i - x_0}$

向量大小 $m_{i,j} = \left(c_{i,j}^*\right)^2(a - bd_{i,j})$

(x_0, y_0) 为机器人位置

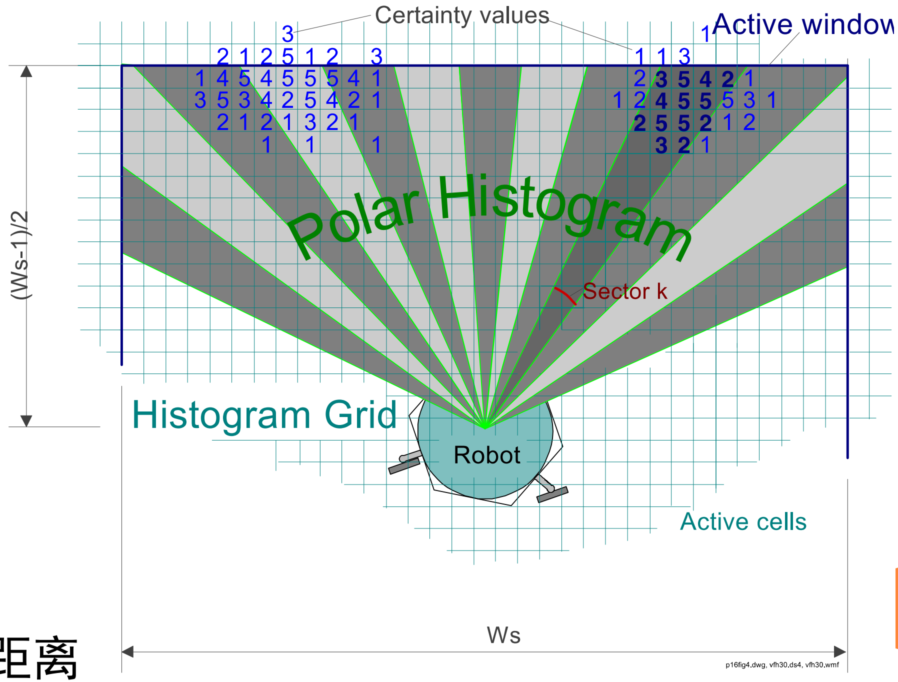
(x_i, y_i) 为单元位置

$c_{i,j}^*$ 为单元栅格值

$d_{i,j}$ 为单元与机器人之间距离

a, b 为正常数 $a - bd_{max} = 0$

d_{max} 最远活跃单元与机器人的距离



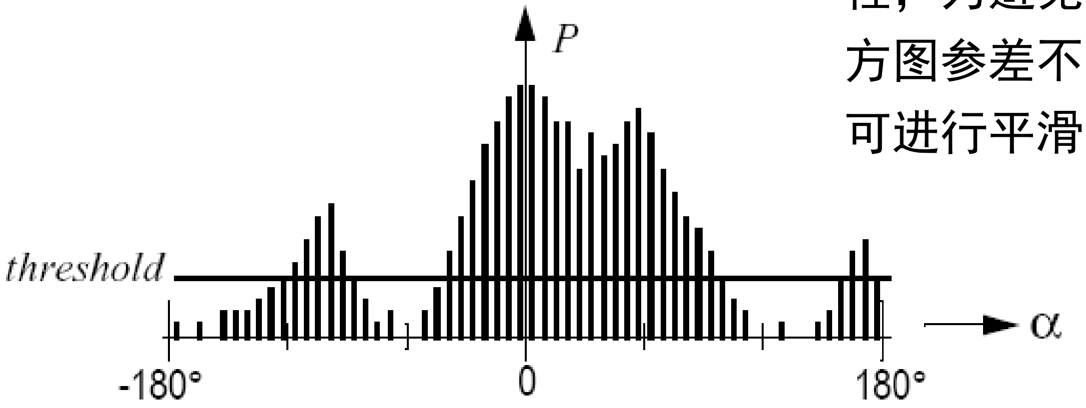
向量势直方图法

3. 转换为极坐标下的障碍物概率直方图

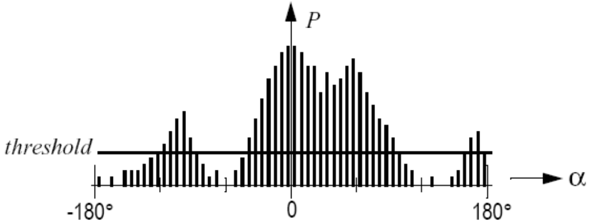
按分辨率α将0-360度分为n个扇区，单元所属扇区为$k = int\left(\frac{\beta_{i,j}}{\alpha}\right)$，极障碍密度为

$$\boldsymbol{h_k = \sum m_{i,j}}$$

由于直方栅格地图的离散特性，为避免一维极坐标系直方图参差不齐影响方向选择，可进行平滑



向量势直方图法



4. 根据直方图，识别所有可以让机器人通过的通道，然后对每个通道计算成本，选择具有最低成本的通道，得到导航方向

成本计算示例：

$$G = a \cdot target\_direction + b \cdot wheel\_orientation + c \cdot previous\_direction$$

target_direction : 路径与目标之间的对齐量

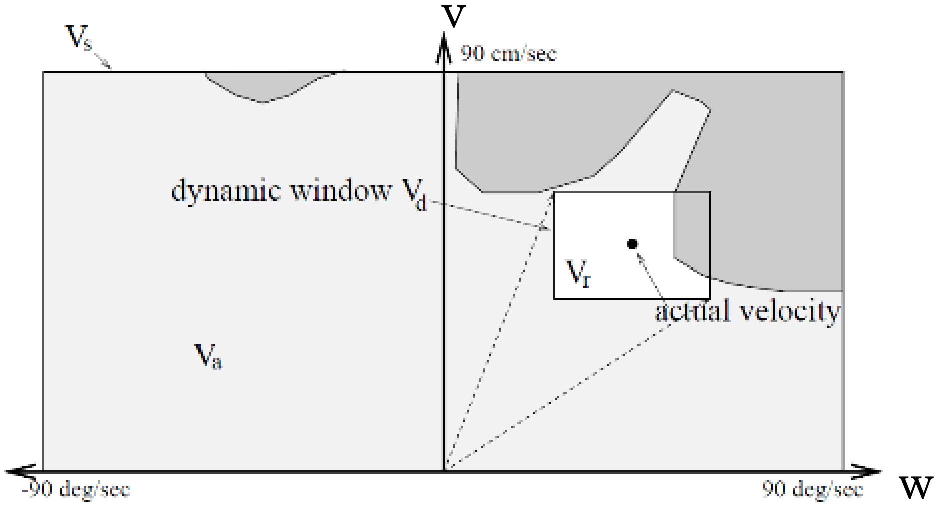
wheel_orientation：新方向和当前机器人方向的差异量

Previous_direction: 原来选择方向和新方向之间的差异量

通过a,b,c进行权重调节，这三个量都是角度量，因此不需要做归一化

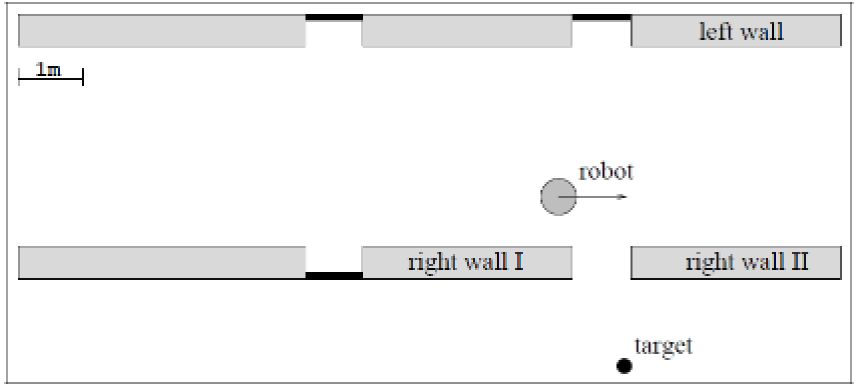
动态窗口法(DWA, Dynamic Window Algorithm)

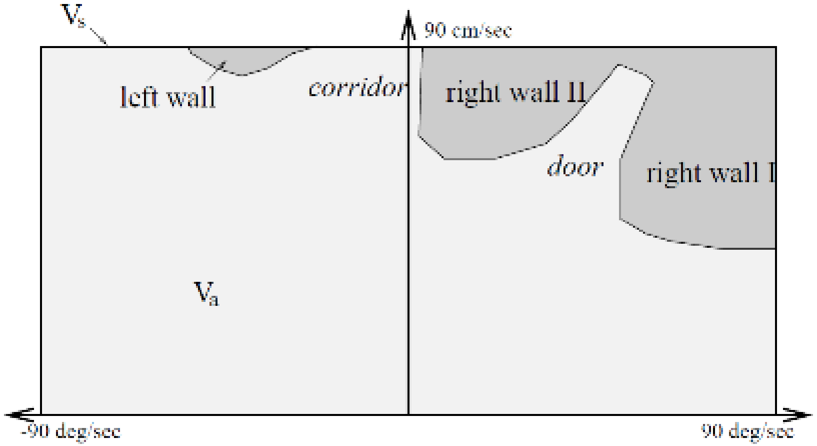
- 1997年由德国波恩大学Dieter Fox, Wolfram Burgard和美国卡耐基梅隆大学Sebastian Thrun提出
- 基本思想：在速度空间中搜索适当的平移速度和旋转速度指令 (v,w)



动态窗口法

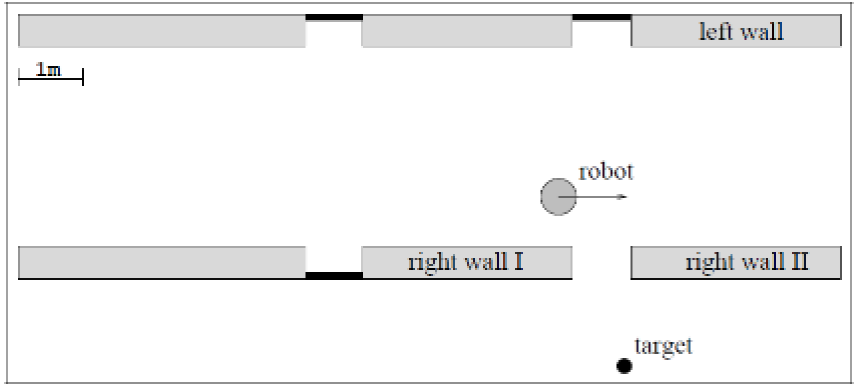
- 从几何空间搜索转化为速度空间搜索

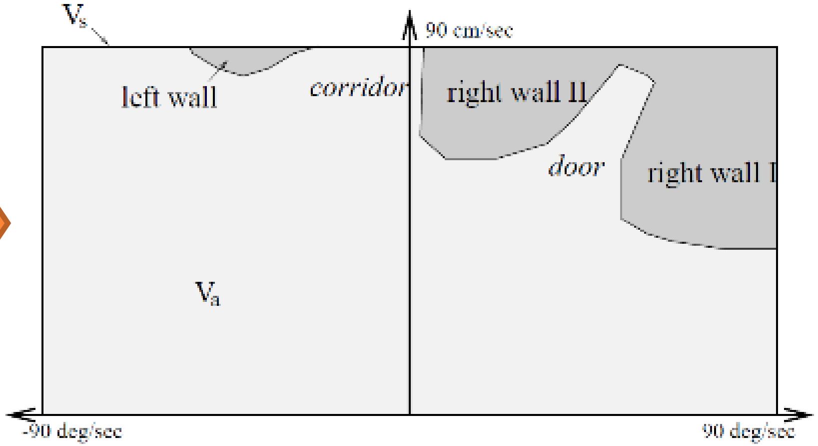




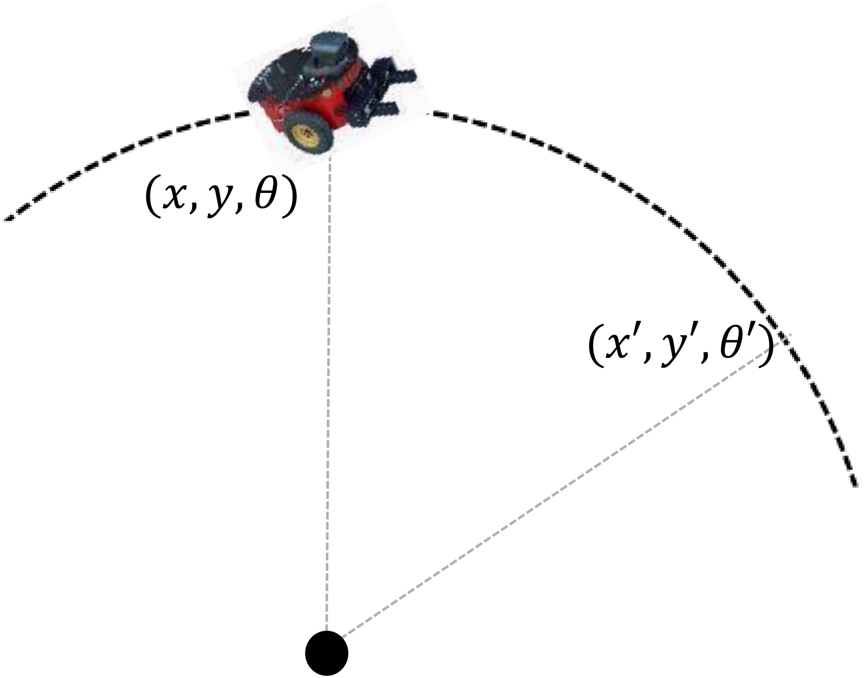
实现步骤

(1) 基于速度控制运动模型，构建可行的速度空间





机器人的速度控制运动模型

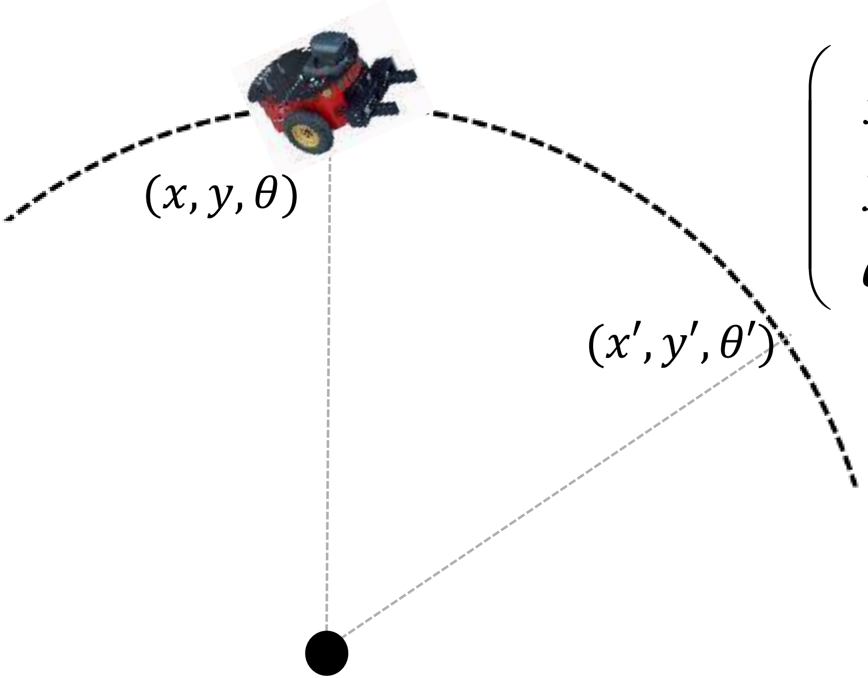


假设没有噪声，控制时间间隔为 Δt

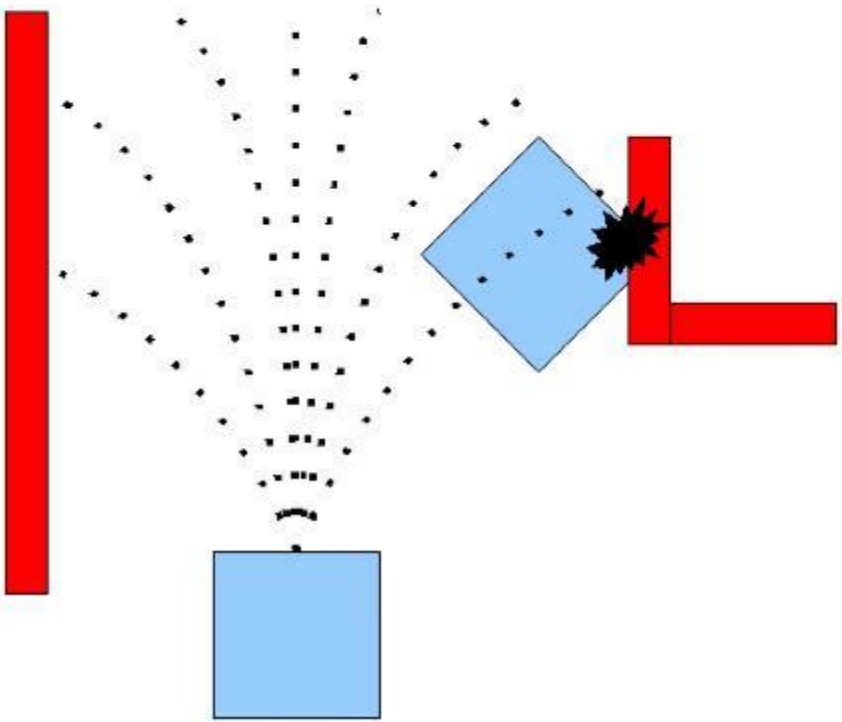
时间间隔内机器人速度和角速度保持不变，则机器人绕着半径为r的圆周运动

$$r = \left|\frac{v}{\omega}\right|$$

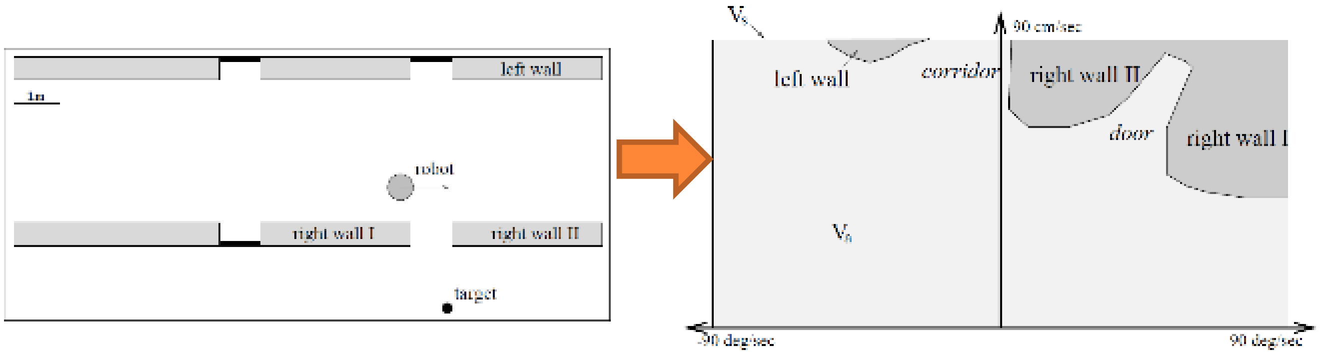
机器人的速度控制运动模型



$$\begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = \begin{pmatrix} x - \frac{v}{\omega}\sin(\theta) + \frac{v}{\omega}\sin(\theta + \omega\Delta t) \\ y + \frac{v}{\omega}\cos(\theta) - \frac{v}{\omega}\cos(\theta + \omega\Delta t) \\ \theta + \omega\Delta t \end{pmatrix}$$



不同的速度指令 (v,w)会得到不同的运动半径，同样的时间间隔到达不同的终止位置。有些位置是安全的，有些会与障碍物发生碰撞



可以让机器人停止不与障碍物相碰的可行速度集合

$$V_a = \left\{ (v,\omega) \mid v \le \sqrt{2 \cdot dist(v,\omega) \cdot \dot{v}_b} \wedge \omega \le \sqrt{2 \cdot dist(v,\omega) \cdot \dot{\omega}_b} \right\}$$

$dist(v,\omega)$ 表示速度配置 (v,ω) 所对应圆弧上最近障碍物的距离

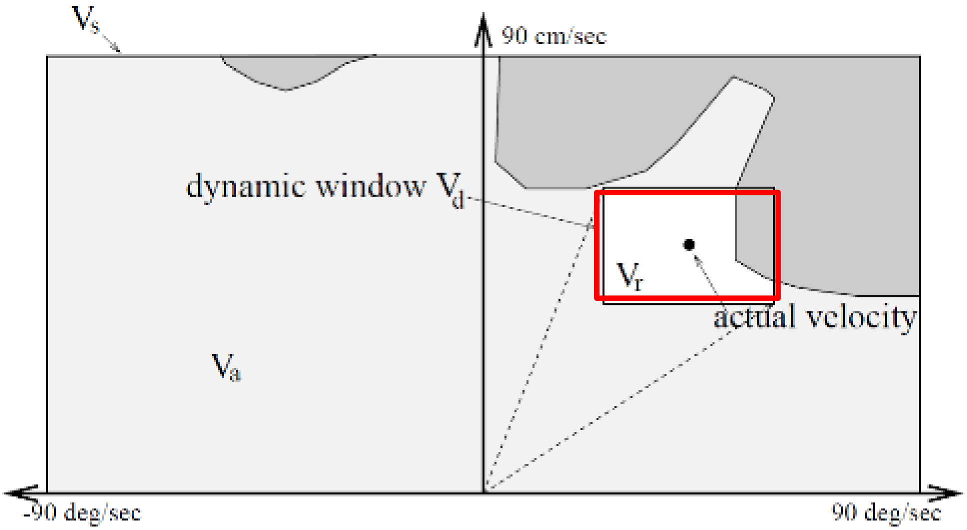
$$s = \frac{1}{2}at^2 = \frac{v^2}{2a}$$

$\dot{v}_b$ 刹车平移加速度

$\dot{\omega}_b$ 刹车旋转加速度

实现步骤

(2) 考虑到机器人在运动过程中最大加速度的约束，在当前速度配置处以固定的小时间间隔开一个速度窗口空间



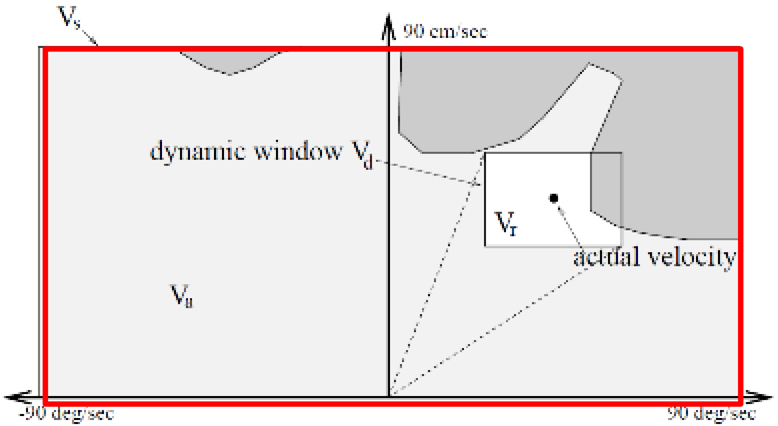
实现步骤

(2) 考虑到机器人在运动过程中最大加速度的约束，在当前速度配置处以固定的小时间间隔开一个速度窗口空间

$$V_d = \{(v,\omega) \mid v \in [v_l, v_h] \wedge \omega \in [\omega_l, \omega_h]\}$$

$$\begin{cases} v_l = v_a - a_{v\max} \times \Delta t \\ v_h = v_a + a_{v\max} \times \Delta t \\ \omega_l = \omega_a - a_{\omega\max} \times \Delta t \\ \omega_l = \omega_a - a_{\omega\max} \times \Delta t \end{cases}$$

实现步骤



(3)结合机器人速度约束，获得可行速度空间为

$$V_r = V_a \cap V_d \cap V_s$$

$$V_a = \left\{ (v,\omega) \mid v \le \sqrt{2 \cdot dist(v,\omega) \cdot \dot{v}_b} \wedge \omega \le \sqrt{2 \cdot dist(v,\omega) \cdot \dot{\omega}_b} \right\}$$

$$V_d = \{(v,\omega) \mid v \in [v_l, v_h] \wedge \omega \in [\omega_l, \omega_h]\}$$

$$V_s = \{(v,\omega) \mid v \in [-v_{\max}, v_{\max}] \wedge \omega \in [-\omega_{\max}, \omega_{\max}]\}$$

实现步骤

(4) 在可行速度空间中选择最优的速度控制指令

$$evaluation(v,\omega)=\alpha\cdot heading(v,\omega)+\beta\cdot dist(v,\omega)+\gamma\cdot velocity(v,\omega)$$

$$\alpha+\beta+\gamma=1(\alpha\geq 0,\beta\geq 0,\gamma\geq 0)$$

$heading(v,\omega)$ 朝向目标点 ： 保证机器人朝目标点运动

$dist(v,\omega)$ 远离障碍物 ： 保证机器人避开障碍物，安全不碰撞

$velocity(v,\omega)$ 速度最大化 ： 保证机器人以最大速度运动

注意各项的归一化处理

动态窗口法

- 存在问题：
  - 根据单步信息数据计算期望速度，在评估选择速度时不考虑速度和路径平滑，容易导致机器人运动存在震动和轨迹扭动问题
  - 参数较多，实际实现依赖工程经验，难以适应各种情况

End !