



清华大学
Tsinghua University

自动驾驶汽车的决策规划技术 (Part I)

李升波

清华大学·车辆与运载学院

Email: lishbo@tsinghua.edu.cn

目录

1

决策规划技术概述

2

分解式决策方案

3

集中式决策方案

决策规划技术概述

决策规划的难点



某一发展中国家的高峰时段交通状况

什么是决策规划？



决策规划系统类似驾驶员的大脑



制定当前道路的行驶策略（跟车、换道、超车、停车、转弯、掉头等），输出参考行驶轨迹



决策规划能力是自动驾驶汽车
“智能性”的核心体现

决策规划的难点

□ 复杂性 (Sophisticated)

- 交通参与者多，道路类型多，规则约束多

□ 动态性 (Dynamic)

- 行人/自行车/机动车/红绿灯均运动，自车面临的情况瞬息万变

□ 随机性 (Stochastic)

- 机动车、非机动车、自行车、行人的意图和行为难以预测

□ 博弈性 (Game-based Interactive)

- 自车与交通参与者（行人、机动车等）的行为是互相影响/制约

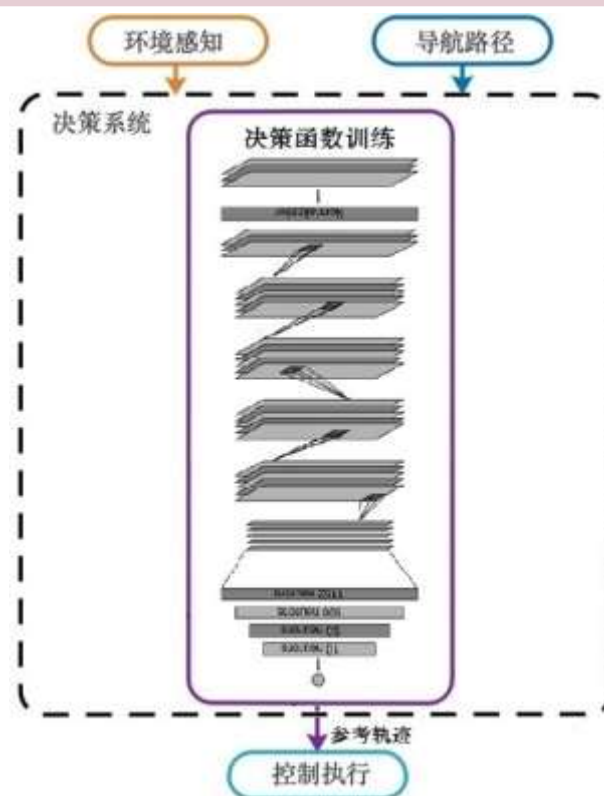
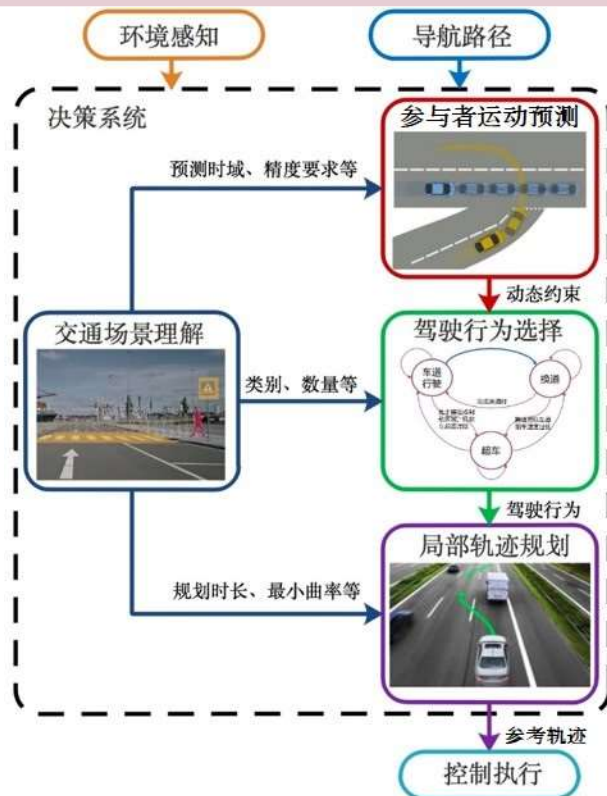


如下交通警示谚语，那一些反映了驾驶过程的博弈性？

- | | | | |
|---------------------------------------|-------------|---------------------------------------|----------------|
| <input type="checkbox"/> A | 红灯停，绿灯行 | <input type="checkbox"/> F | 黄实线，不能压导向线，要遵守 |
| <input type="checkbox"/> B | 开车守规矩，不要抢红灯 | <input type="checkbox"/> G | 违规掉头扣3分 |
| <input checked="" type="checkbox"/> C | 左转让直行，变道莫加塞 | <input checked="" type="checkbox"/> H | 打转向灯，提醒后车避让 |
| <input checked="" type="checkbox"/> D | 支线让干线，抢行起祸端 | <input checked="" type="checkbox"/> I | 闪远光灯，晃瞎对头车 |
| <input type="checkbox"/> E | 超车选直宽，蛮干不安全 | | |
- 提交

两类基本决策方案

	分解式	集中式
定义	将决策过程分解为独立的子问题，如：场景理解、运动预测、行为选择、路径规划等，每一个问题独立解决。	以环境感知结果为输入，以期望路径或执行器控制命令为输出，将决策过程视作一个不可分解的黑箱。



两类基本决策方案

	分解式	集中式
方法	排序、搜索、优化等	监督学习、强化学习等
代表	谷歌、百度、通用、福特、Tesla等	英伟达, Comma.ai, Wayve 等
优点	<ul style="list-style-type: none">✓ 问题可分解、任务可分工✓ 节省车载存储和计算资源✓ 决策代码的开发可控性好	<ul style="list-style-type: none">✓ 体系框架简洁明了✓ 环境感知信息无损失✓ 几乎无需手工标记数据
缺点	<ul style="list-style-type: none">x 感知信息存在损失x 涵盖场景\行为有限x 决策目标制定困难	<ul style="list-style-type: none">x 难以嵌入已知驾驶经验x 场景间难以互相迁移x 算法难以理解与手动改进

典型决策功能设计



- 驾驶水平：曾获美国DARPA无人驾驶挑战赛**冠军**。最高时速达190km/h。
- 测试道路：Thunderhill Raceway赛道
- 传感配置：差分GPS、激光雷达、轮速传感器、加速度传感器、陀螺仪

□ 决策功能解析

- 场景：非公路赛道、城市赛道
- 分解式决策方案
 - (1) 场景理解：支持向量机 (SVM)
 - (2) 周车预测：卡尔曼滤波、粒子滤波
 - (3) 行为选择：有限状态机、贝叶斯推断
 - (4) 轨迹规划：动态规划、多项式轨迹采样、Hybrid A*搜索

典型决策功能设计



- 驾驶水平：行驶145万英里，人工接管0.076次/1000英里（*2019年DMV报告）
- 示范运行：美国加州、亚利桑那州（600辆）
- 传感配置：GPS，激光雷达，摄像头，毫米波雷达、惯导IMU

□ 决策功能解析

- 场景：结构化城市道路+高速公路
- 分解式决策方案
 - （1）场景理解：卷积神经网络、动态博弈
 - （2）周车预测：循环神经网络
 - （3）行为选择：卷积神经网络、有限状态机
 - （4）轨迹规划：循环神经网络、Hybrid A*

典型决策功能设计



- 驾驶水平：行驶10.8万英里，人工接管0.055次/1000英里，居于榜首（*2019年DMV报告）
- 示范运行：中国长沙（45辆）
- 传感配置：GPS，激光雷达，摄像头，毫米波雷达、惯导IMU

□ 决策功能解析

- 场景：结构化城市道路+高速公路
- 分解式决策方案
 - （1）场景理解：卷积神经网络
 - （2）周车预测：卡尔曼滤波
 - （3）行为选择：有限状态机
 - （4）轨迹规划：轨迹采样、二次规划、交互式预测

典型决策功能设计



- 驾驶水平：每3.07百万英里行驶中发生一次事故；自主泊车/出库
- 传感配置：GPS、摄像头，超声波雷达，毫米波雷达

□ 决策功能解析

- 场景：结构化城市道路+高速公路
- 分解式决策方案
 - (1) 场景理解：数字神经网络
 - (2) 周车预测：粒子滤波
 - (3) 行为选择：有限状态机
 - (4) 轨迹规划：卷积神经网络

典型决策功能设计



- 驾驶水平：平均人工干预里程为11.0英里。
- 传感配置：3个摄像头

□ 决策功能解析

- 场景：稀疏交通流、结构化低速道路
- 集中式决策方案
 - 卷积神经网络、全连接神经网络

典型决策功能设计



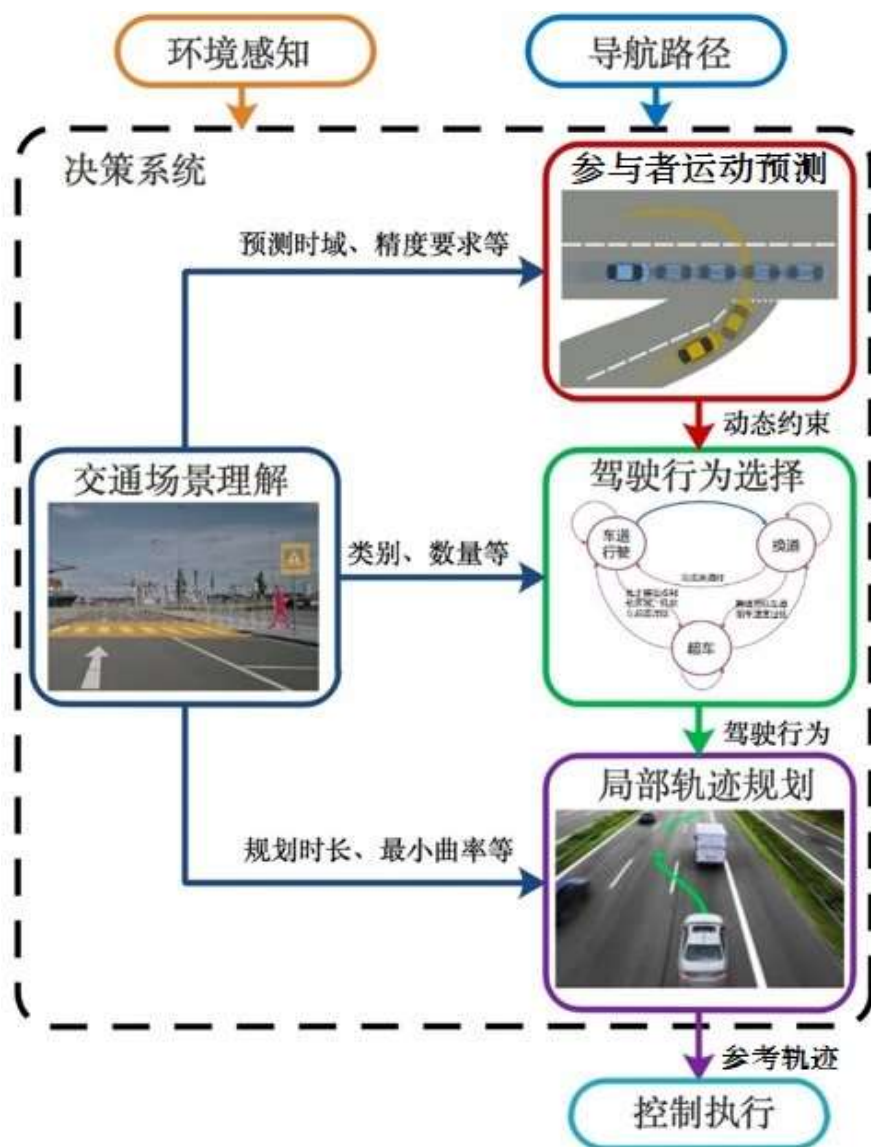
- 驾驶水平：车道保持功能，车速 $< 20\text{km/h}$ 。
- 传感配置：1个前向摄像头

□ 决策功能解析

- 场景：稀疏交通流、结构化低速道路
- 集中式决策方案
 - 强化学习（DDPG、卷积神经网络）

分解式决策规划方案

分解式决策的构架



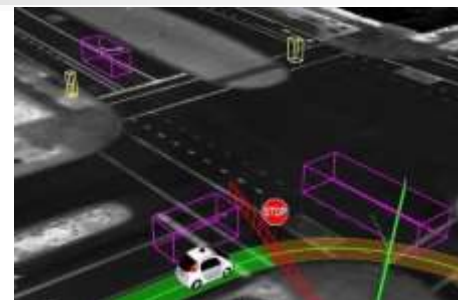
类型	学者	特点
有限状态机	Buehler M et al., 2009	简单易用, 但是扩展性差, 依赖设计人员经验
模糊逻辑	Vicente Milanés et al., 2010	
驾驶员模型	Davis L et al., 2007; Chen et al., 2017	
博弈规则	Kitta 1999; Xuegang Ban et al., 2007; Kyungwon Kang et al., 2017	
神经网络模型	MA Arain et al., 1993; Y Lin et al., 2005	依赖一定数量的驾驶员数据训练
马尔可夫决策模型	S Ulbrich, 2013	
贝叶斯模型	S brechtel et al., 2014	



分解式决策的构架

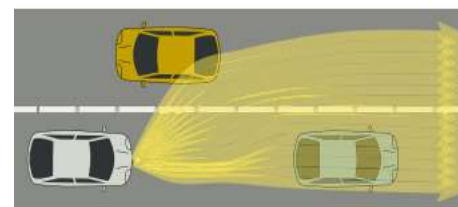
□ (a) 交通情景理解

- 对自车所处道路交通环境的认知与理解



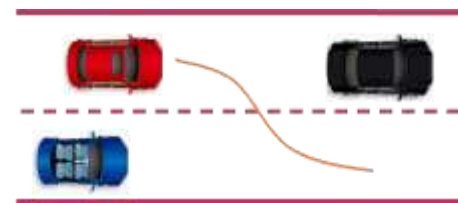
□ (b) 参与者运动预测

- 预测交通参与者的行驶意图和行驶轨迹



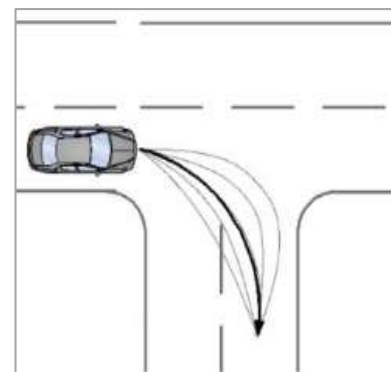
□ (c) 驾驶行为选择

- 依据安全、高效、低耗等目标，选择超车、换道、直行等某一行为模式



□ (d) 局部轨迹规划

- 规划一条可行的局部行驶轨迹，作为后续的运动控制参考



分解式决策：交通情景理解

□ 从感知到认知



■ 感知：

- What? ——行人
- Where? ——马路右侧
- How? ——不知道
- Why? ——不知道
- 识别目标的类型及姿态

■ 认知：

- What? ——行人
- Where? ——马路右侧
- How? ——横穿马路或停止不前
- Why? ——不知道
- 对目标可能的行为模式进行估计

分解式决策：交通情景理解

□ 从认知到理解



■ 认知:

- What? ——行人
- Where? ——马路右侧
- How? ——横穿马路或停止不前
- Why? ——不知道
- 对目标可能的行为模式进行估计

■ 理解:

- What? ——行人
- Where? ——马路右侧
- How? ——横穿马路或停止不前
- Why? ——行人抢行或让汽车先行
- 推断交通参与者之间的博弈关系

分解式决策：交通情景理解

□ “理解一件事情” 的困难性



■ 挖掘隐含信息

- **认知：**一个男人被一群大象追着跑
- **理解：**大选前夕，奥巴马被一群共和党竞选者追赶！

■ 推断逻辑关系

- **认知：**一辆车斜横于车道中间
- **理解：**警车在后面，驾驶员的确着急，小心这辆车！

分解式决策：交通情景理解

□ “认知” VS “理解”：环境感知的不同要求



■ 认知层的感知需求：

- 目标位置
- 行进速度
- 行进方向

■ 理解层的感知需求：

- 身高/年龄/性别
- 手部动作/步幅/步频/头部朝向
- 人行横道、地面标识、路侧标识
- 人群状态、他车行为
- 天气、光照、路面 等

分解式决策：交通情景理解

□ 情景理解的语义解析

● 感知层

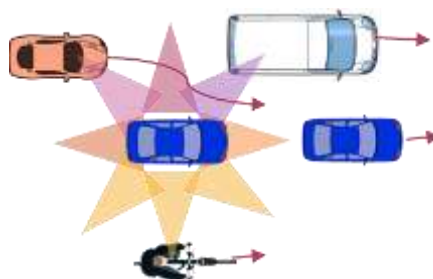
人车路的位置姿态



- 道路结构
- 行人/骑车人
- 车辆
- 信号灯
- 路侧/地面标识

● 认知层

人车路态势判断



- 场景类型判别
- 交通流密度估测
- 自车可选行为模式
- 行车风险评估

● 理解层

人车路行为的机理解析



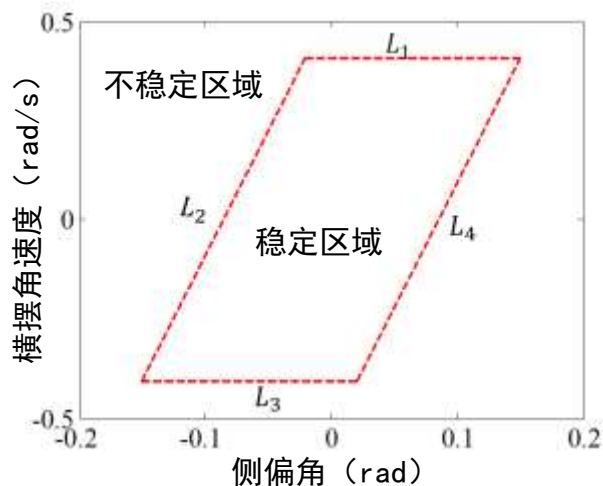
- 参与者的行为博弈
- 参与者的意图解析
- 参与者的冲突模式

认知层：行车风险评估

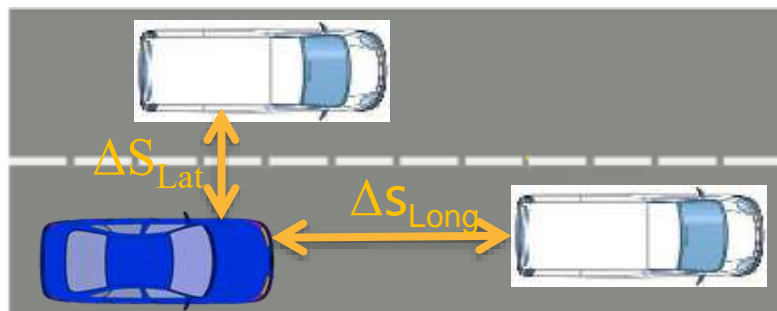
□ 典型风险评估模型



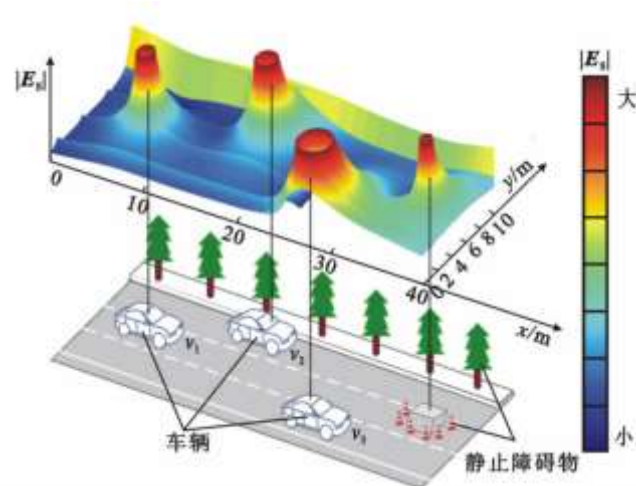
动力学失稳风险



稳定包络模型



车辆碰撞风险

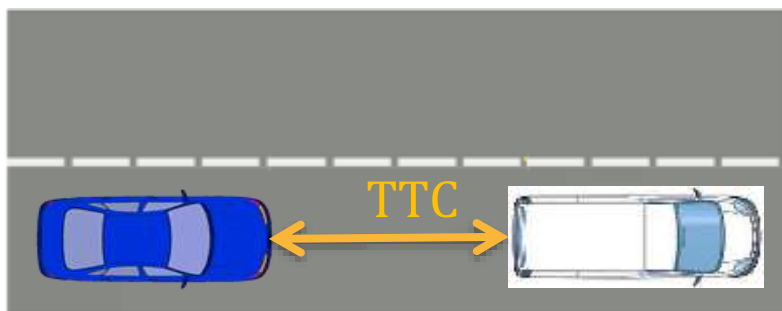


安全场模型

认知层：行车风险评估

□ 纵向跟车风险：TTC (Time To Collision)

- 后车与前车发生碰撞所需的时间



- D — 前后车相对距离
- Δv — 后车车速 - 前车车速

$$TTC = \frac{D}{\Delta v}$$

前车车速	20km/h	40km/h	60km/h
低风险	5s	5s	5s
中风险	1.1s	1.3s	1.3s
高风险	0.67s	0.8s	0.83s

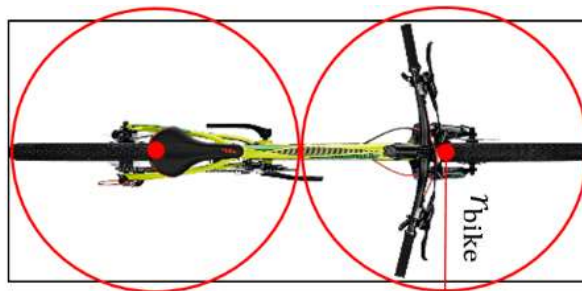
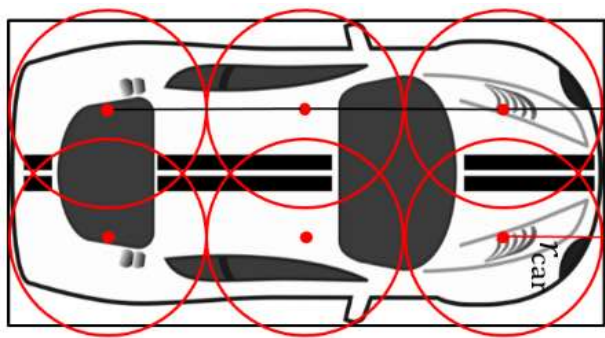
请计算一下后车车速=40km/h，车间距离是50米时，大致的纵向行车风险是多少呢？

- ☐ A 高风险
- ☐ B 中风险
- ☒ C 低风险
- ☐ D 难以计算

提交

认知层：行车风险评估

□ 车车碰撞风险：安全距离法



$$R_{\text{汽车}} = \frac{\text{车长}}{6}$$

$$R_{\text{自行车}} = \frac{\text{车长}}{2}$$

$$R_{\text{人}} = 0.5\text{m}$$

计算自车各圆圆心与目标物圆心的距离，当任一距离小于两者半径之和， $d < (R_{ego} + R_{env})$ ，即视为**碰撞事故**。

车速	20km/h	40km/h	60km/h
安全距离	3.2m	9.8m	21m

理解层：车辆博弈关系建模

□ 交通参与者博弈关系



斑马线抢行/避让



换道抢行/避让



避让救护车



无交通灯抢行/避让



有交通灯抢行/避让



顾客打车

理解层：车辆博弈关系建模

□ 零和博弈

- 各方的利益冲突，总利益/损失为“零”



□ 非零和博弈

- 各方可能存在某种共同利益

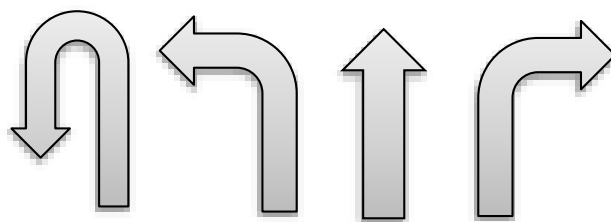


分解式决策：参与者运动预测

□ 基本任务

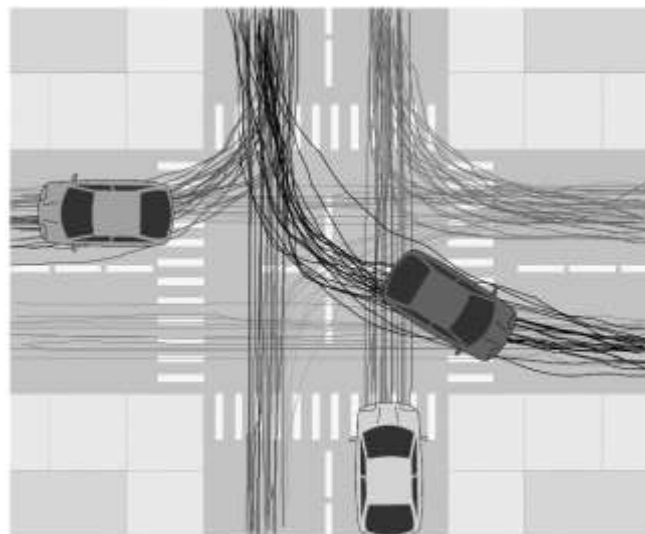
● 第一步：识别行驶意图

- 直行：自由直行、跟驰、紧急制动等
- 换道：左\右换道、自由换道、受迫换道
- 转弯、掉头、并线等



● 第二步：预测运动轨迹

- 预测未来可能出现的位置
- 预测未来每个时间的速度

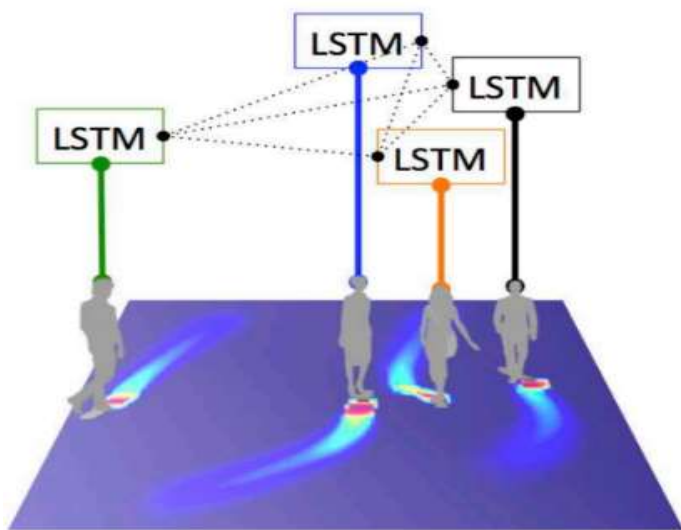


分解式决策：参与者运动预测

□ 以运动轨迹预测为例

- 待预测的对象：行人、自行车、机动车 等
- 挑战性难题：
 - 交通参与者的路权存在重叠\冲突（自行车→机动车道）
 - 使用道路的优先级难以确定（车避让人 VS 人避让车）
 - 参与者之间存在更强的交互性\博弈性

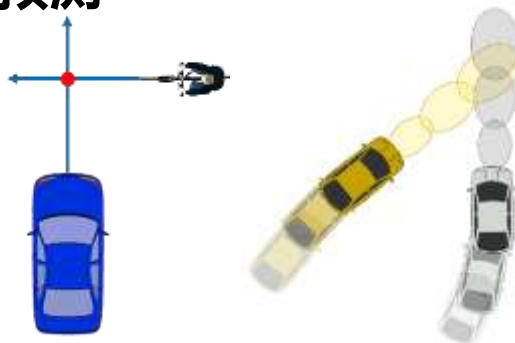
□ 行人轨迹预测方法（室内）



分解式决策：参与者运动预测

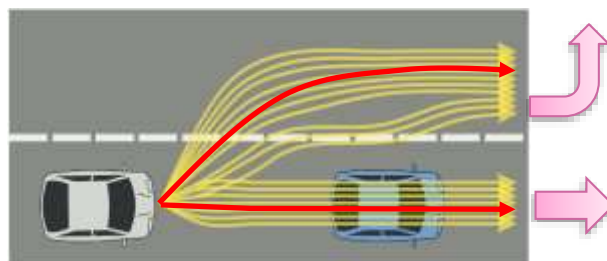
□ 开放道路的周车轨迹预测

■ 开环轨迹递推模型



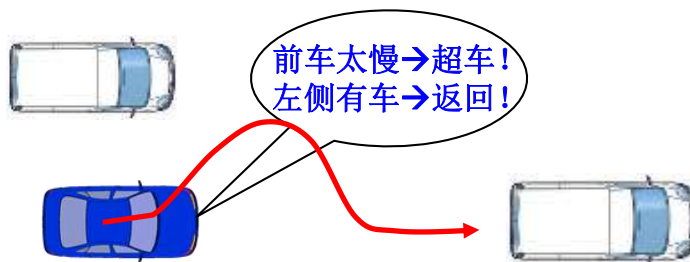
✓ 预测 $<1s$ ，适用于车辆实时控制

■ 驾驶员闭环递推模型 (+ 驾驶员行为模型)



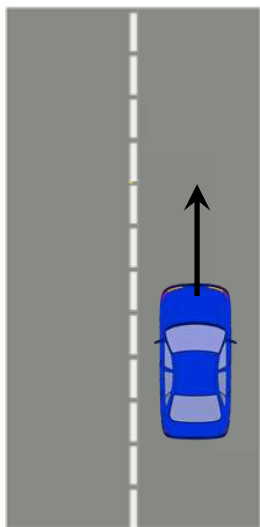
✓ 预测 $1\sim3s$ ，适用于稀疏交通流

■ 驾驶员闭环递推模型 (+ 驾驶员意图模型)



✓ 预测 $1\sim3s$ ，考虑人-人、车-车交互，适用于中等密度交通流

□ 线性递推模型（直行）



■ 恒定速度 v

$$x(t+1) = x(t) + v * \Delta t$$

■ 恒定加速度 a

$$x(t+1) = x(t) + v(t) * \Delta t$$

$$v(t+1) = v(t) + a * \Delta t$$

- $x(t)$ — 当前时刻纵向位置； $x(t+1)$ — 下一时刻纵向位置
- $v(t)$ — 当前时刻速度； $v(t+1)$ — 下一时刻速度
- Δt — 时间间隔

开环轨迹递推模型

□ 线性递推模型（转向）

- 刚性车身假设（忽略轮胎侧偏角）

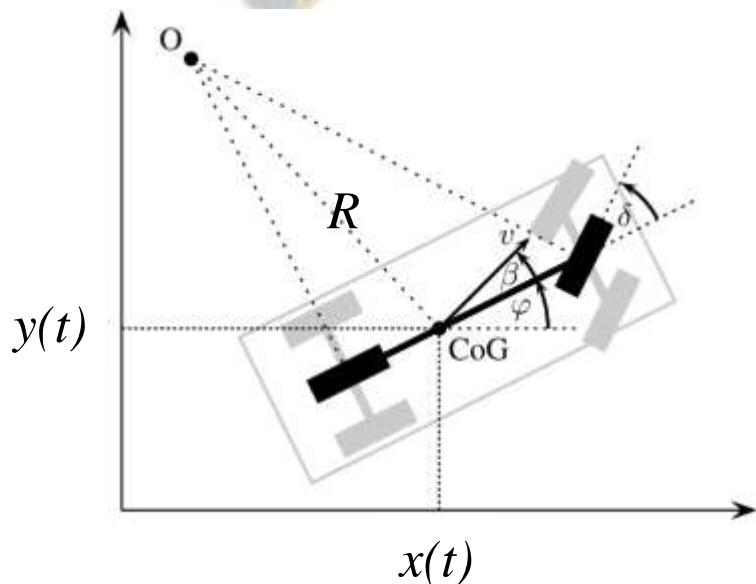


- 恒定前轮转角 δ , 和速度 v

$$x(t+1) = x(t) + v * \Delta t * \sin(\beta + \varphi(t))$$

$$y(t+1) = y(t) + v * \Delta t * \cos(\beta + \varphi(t))$$

$$\varphi(t+1) = \varphi(t) + v * \Delta t / R$$



- $y(t)$ — t 时刻侧向位置
- $\varphi(t)$ — t 时刻横摆角
- $\beta = f(\delta)$ — 质心侧偏角
- R — 质心转向半径

开环轨迹递推模型

□ 线性递推模型（转向）

- 刚性车身假设（忽略轮胎侧偏角）

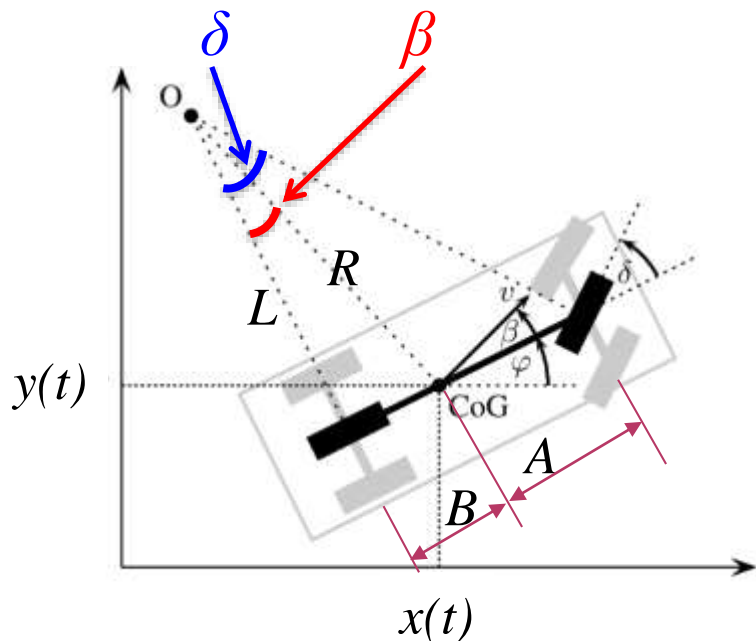
■ 前轮转角 δ 和质心侧偏角 β 的关系

$$\tan(\delta) = (A+B)/L$$

$$\tan(\beta) = B/L$$



$$\beta = \tan^{-1}(B * \tan(\delta) / (A+B))$$



- A — 质心到前轴距离
- B — 质心到后轴距离
- L — 转向中心到后轴距离

开环轨迹递推模型

□ 线性递推模型（转向）

- 刚性车身假设（忽略轮胎侧偏角）



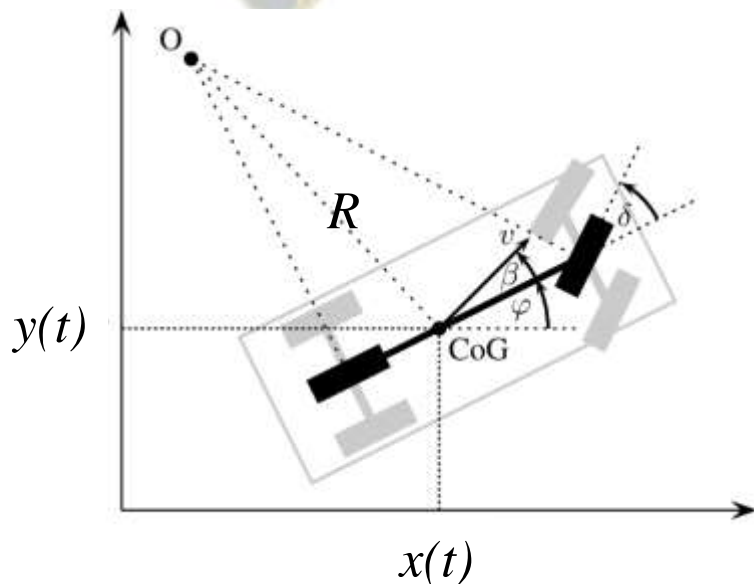
■ 恒定前轮转角 δ , 和加速度 a

$$x(t+1) = x(t) + v(t) * \Delta t * \sin(\beta + \varphi(t))$$

$$y(t+1) = y(t) + v(t) * \Delta t * \cos(\beta + \varphi(t))$$

$$\varphi(t+1) = \varphi(t) + v(t) * \Delta t / R$$

$$v(t+1) = v(t) + a * \Delta t$$



- $y(t)$ — t 时刻侧向位置
- $\varphi(t)$ — t 时刻横摆角
- $\beta = f(\delta)$ — 质心侧偏角
- R — 质心转向半径

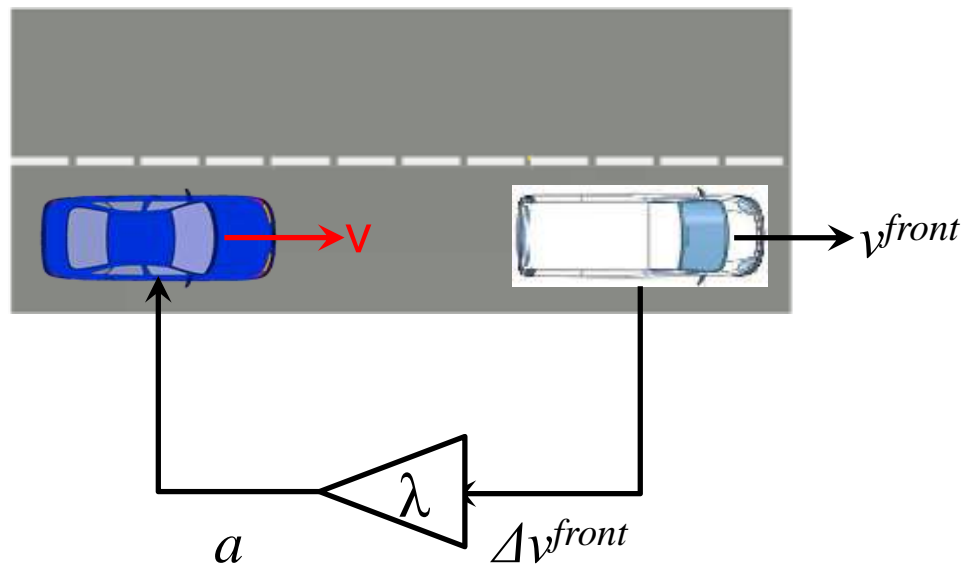
驾驶员闭环递推模型

□ 驾驶员纵向跟驰模型

- 利用车辆跟驰数据拟合的刺激-反应模型

$$a(t+T) = \lambda * \Delta v^{front}(t)$$

- a — 车辆加速度
- λ — 跟车灵敏度
- T — 驾驶员的反应时间
- $\Delta v^{front}(t)$ — 前车速度变化



驾驶员闭环递推模型

□ 驾驶员纵向跟驰模型

- 利用车辆跟驰数据拟合的刺激-反应模型

$$a(t+T) = \lambda * \Delta v^{front}(t)$$

- a — 车辆加速度
- λ — 跟车灵敏度
- T — 驾驶员的反应时间
- $\Delta v^{front}(t)$ — 前车速度变化



$$x(t+1) = x(t) + v(t) * \Delta t$$

$$v(t+1) = v(t) + a(t) * \Delta t$$

$$a(t) = \begin{cases} 0 & , t-t_0 < T \\ \lambda * \Delta v^{front}(t) & , t-t_0 \geq T \end{cases}$$

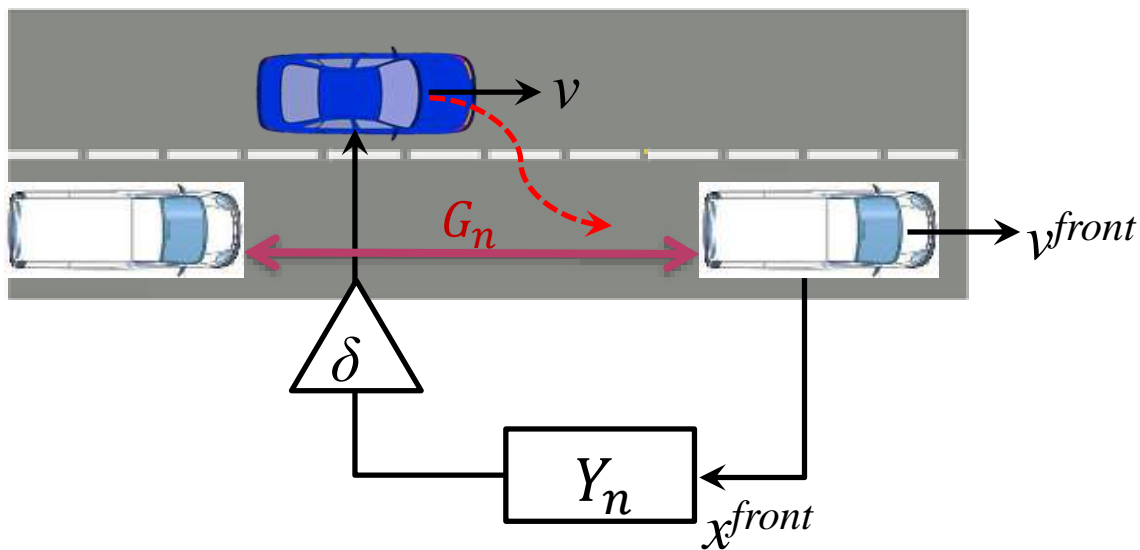
驾驶员闭环递推模型

□ 驾驶员换道跟踪模型（结合换道意图）

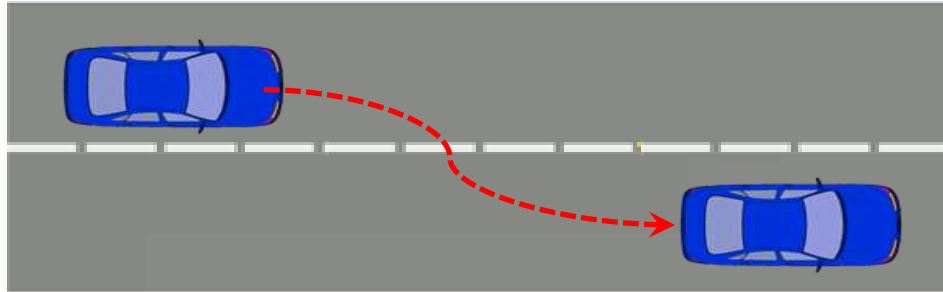
- 换道意图判定：利用目标车道的前车和后车状态，判断能否换道

$$Y_n(t) = \begin{cases} 1 & \text{if } G_n(t) \geq G_n^{er}(t) \\ 0 & \text{if } G_n(t) \leq G_n^{er}(t) \end{cases}$$

- $Y_n(t)$ 是选择指示变量，值是1间隙可接受，值是0则相反；
- $G_n(t)$ 是可用到的间隙； $G_n^{er}(t)$ 是临界间隙



□ 换道轨迹跟踪模型



- 换道轨迹:

$$y_{des}(t) = b_3 t^3 + b_2 t^2 + b_1 t + b_0$$



$$x(t+1) = x(t) + v * \Delta t * \sin(\beta(t) + \varphi(t))$$

$$y(t+1) = y(t) + v * \Delta t * \cos(\beta(t) + \varphi(t))$$

$$\varphi(t+1) = \varphi(t) + v * \Delta t / R$$

$$\beta(t) = f(\delta(t))$$

$$\delta(t) = k_{Lateral} * (y_{des}(t) - y(t))$$

侧向轨迹误差反馈

请具体计算一下 5步 纵向闭环预测模型？

$$x(t+1) = x(t) + v(t) * \Delta t$$

$$v(t+1) = v(t) + a(t) * \Delta t$$

$$a(t) = \begin{cases} 0, & t - t_0 < T \\ \lambda * \Delta v^{front}(t), & t - t_0 \geq T \end{cases}$$

初始状态: $x(0)=0, v(0)=10m/s, \Delta t=0.2 \text{ sec}, T=0.6 \text{ sec}$

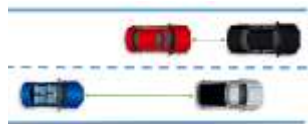
作答

正常使用主观题需2.0以上版本雨课堂

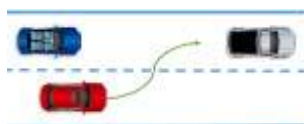
分解式决策：驾驶行为选择

□ 驾驶行为分类

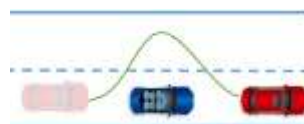
通用



直行/跟车



换道



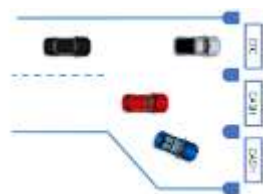
超车



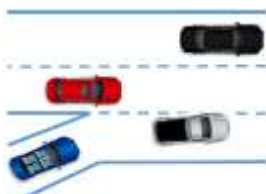
并道



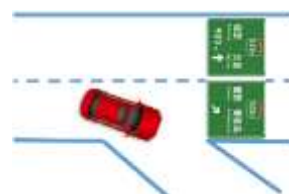
高速



过收费站



进匝道



出匝道



极端工况行车



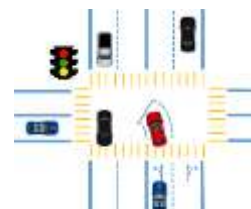
城市



避让行人



抢道



左转/右转/掉头



过环岛



分解式决策：驾驶行为选择

□ 行为选择的基本流程



例子

高速公路



自由直行、跟车
左换道、右换道

道路拥堵



自由直行、跟车
左换道、右换道

家有急事



自由直行、跟车
左换道、右换道

最终选择



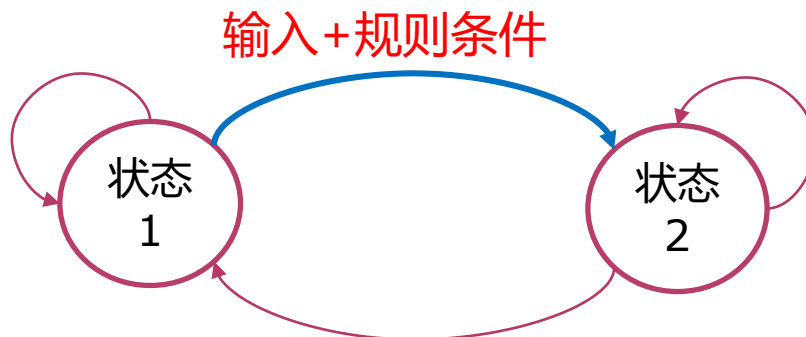
自由直行、跟车
左换道、右换道

分解式决策：驾驶行为选择示例

□ 选择方法：有限状态机

- 行为状态（例：跟车、自由直行、换道）
- 输入量（例：自车速度、前车位置、侧车位置）
- 规则条件（例：自车速度 $<80\text{km/h}$ ）

□ 有限状态机的示意图

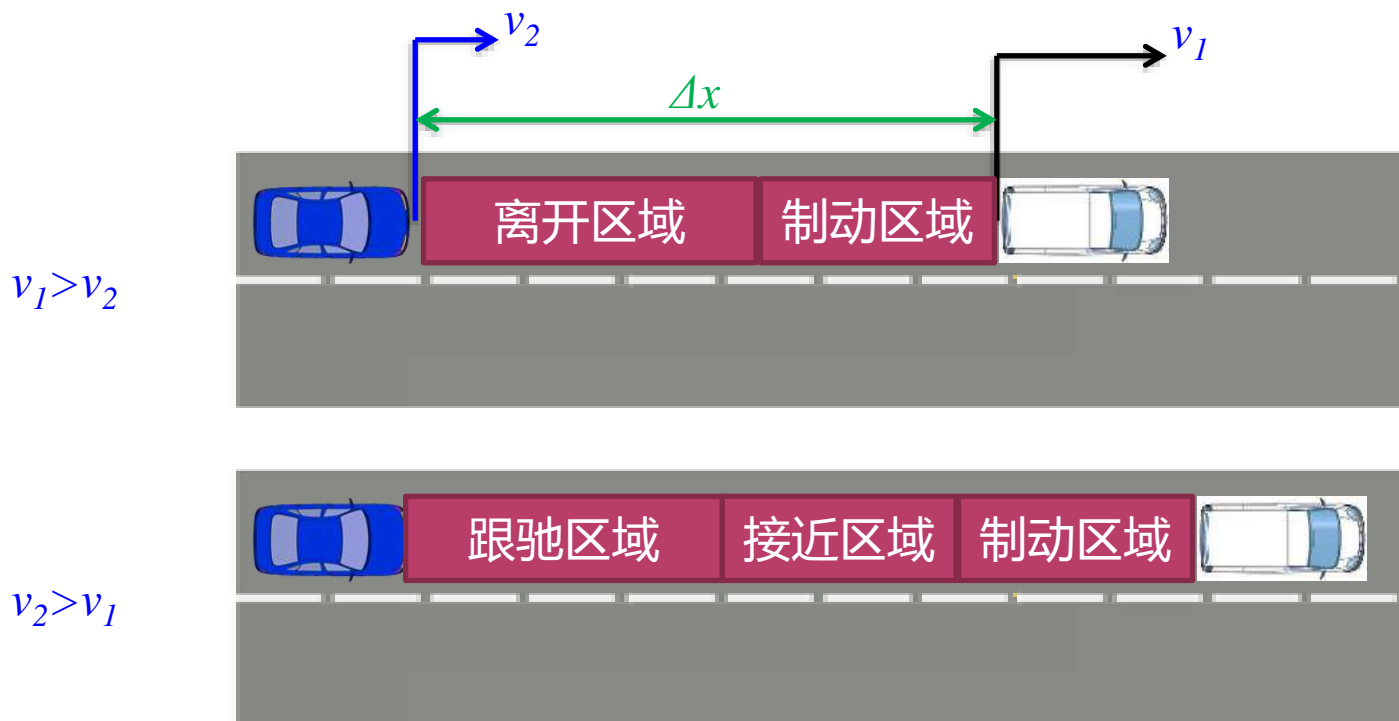


- 有限个状态数量
- 只能处于某一种状态
- 给定规则条件的状态转移

分解式决策：驾驶行为选择示例

□ 基于有限状态机的行为选择

- 道路场景：高速公路
- 行为状态：跟车、换道、超车
- 输入量：与前车的距离（ Δx ）、与前车的相对速度（ Δv ）



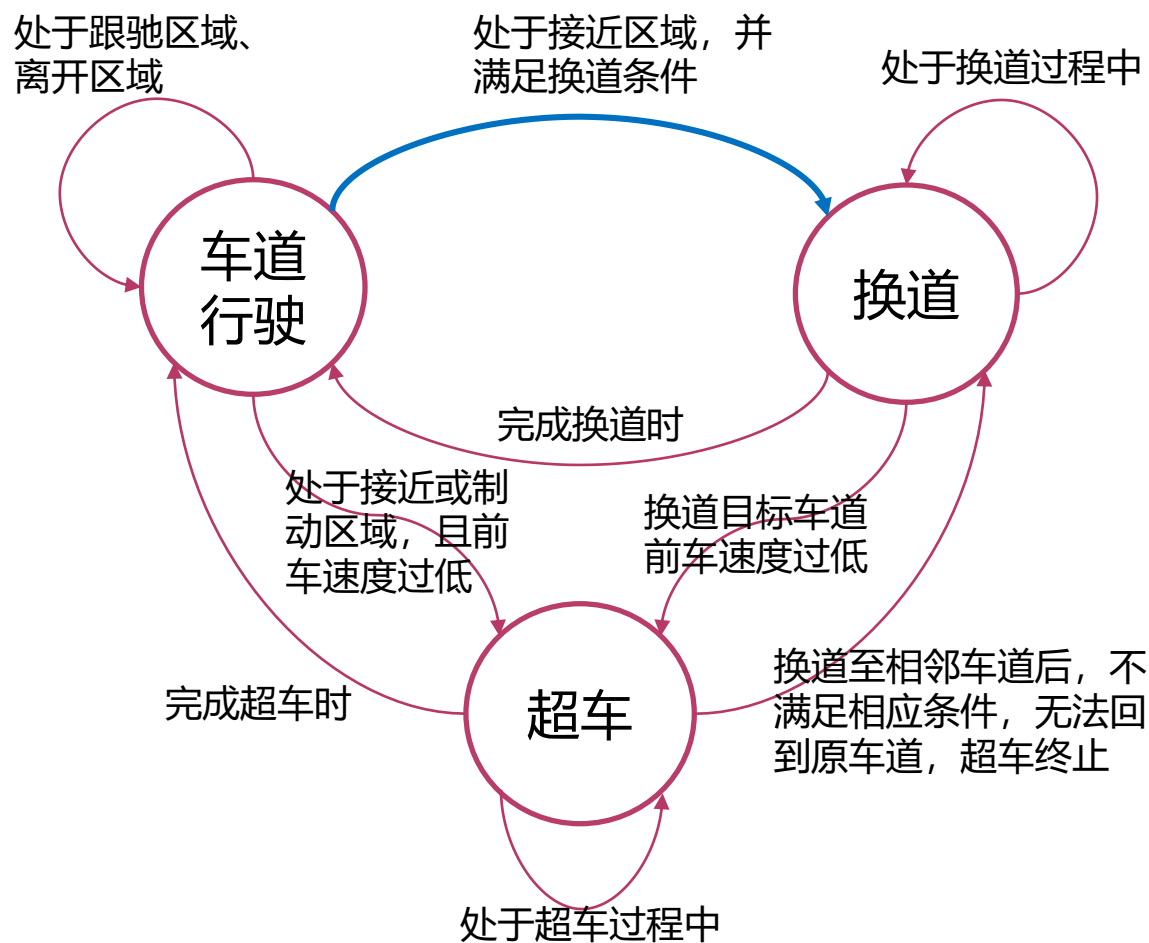
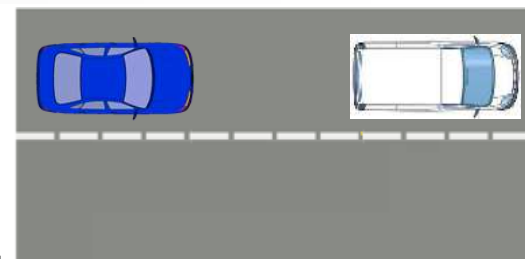
分解式决策：驾驶行为选择示例

□ 基于有限状态机的行为选择（示例）

- 主要行为选择规则：
 - 如果处于跟驰区域、离开区域，则保持车道行驶
 - 如果处于接近区域，并满足换道条件，则进行换道
 - 如果完成换道，则进行车道行驶
 - 如果处于接近或制动区域，且前车速度过低，则进行超车
 - 如果完成超车，则进行车道行驶
 - 换道过程中，如果换道目标车道前车速度过低，则进行超车
 - 超车过程中，如果无法回到原车道，则超车终止，进行换道

分解式决策：驾驶行为选择示例

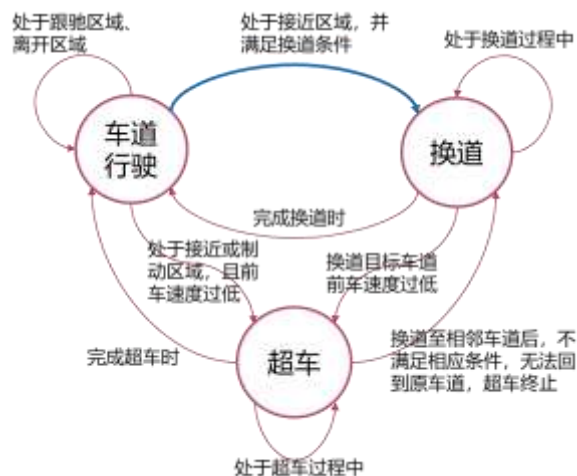
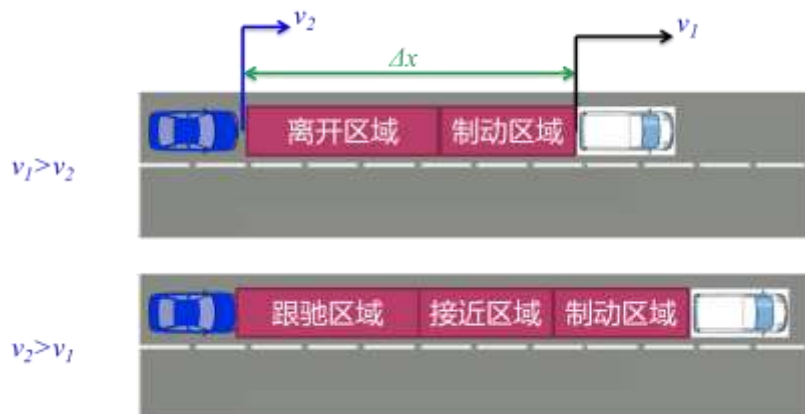
□ 基于有限状态机的行为选择（示例）



3个状态
+
9条规则

分解式决策：驾驶行为选择示例

□ 基于有限状态机的行为选择（示例）

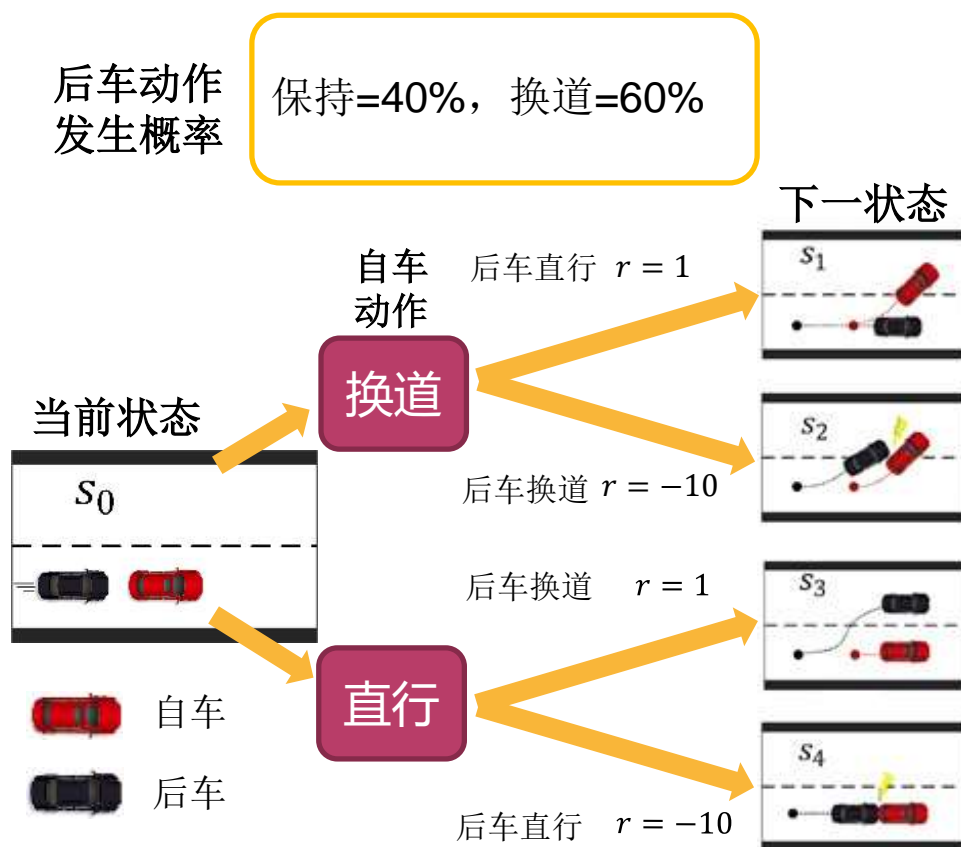


- $\Delta x \in$ 跟驰区域 or 离开区域 = 车道行驶
- 超车 + 完成超车 = 车道行驶
- 换道 + 完成换道 = 车道行驶
- $\Delta x \in$ 接近区域 + 可以换道 = 换道
- 超车 + 已至邻道 + 不能回原车道 = 换道
- 换道 + 未完成换道 = 换道
- $\Delta x \in$ 接近区域 or 制动区域 + 前车速度低 = 超车
- 换道 + 目标车道前车速度低 = 超车
- 超车 + 未完成超车 = 超车

分解式决策：驾驶行为选择示例

□ 选择方法：贝叶斯推断

- 示例：双车行驶工况
- 动作=[保持、换道]；状态=[当前车道、临侧车道]



分解式决策：驾驶行为选择示例

□ 选择方法：贝叶斯推断

- 示例：双车行驶工况
- 动作=[保持、换道]；状态=[当前车道、临侧车道]

$$p(\text{后车直行}) = 0.4 \quad r(\text{自车换道: 安全}) = +1$$

$$p(\text{后车换道}) = 0.6 \quad r(\text{自车换道: 危险}) = -10$$

$$p(\text{后车换道}) = 0.6 \quad r(\text{自车直行: 安全}) = +1$$

$$p(\text{后车直行}) = 0.4 \quad r(\text{自车直行: 危险}) = -10$$

风险

自车换道:

$$\text{cost} = 0.4 * 1 + 0.6 * (-10)$$

自车直行:

$$\text{cost} = 0.6 * 1 + 0.4 * (-10)$$

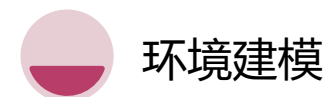
最优动作：？

分解式决策：局部轨迹规划

□ 基本概念：

- 给定起点和终点位置，连接起点位置和终点位置的序列点或曲线称之为轨迹，构成轨迹的策略称之为轨迹规划。
 - 轨迹：带时间轴信息的位置曲线
 - 路径：不带时间轴信息的轨迹

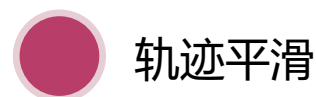
□ 一般步骤：



将物理空间抽象成计算机可处理的数学模型。

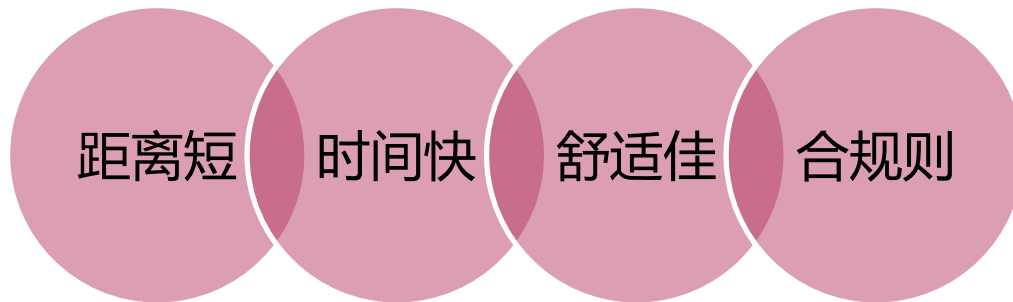


利用优化目标寻找一条最佳的可行轨迹，使预定的目标性能最优。



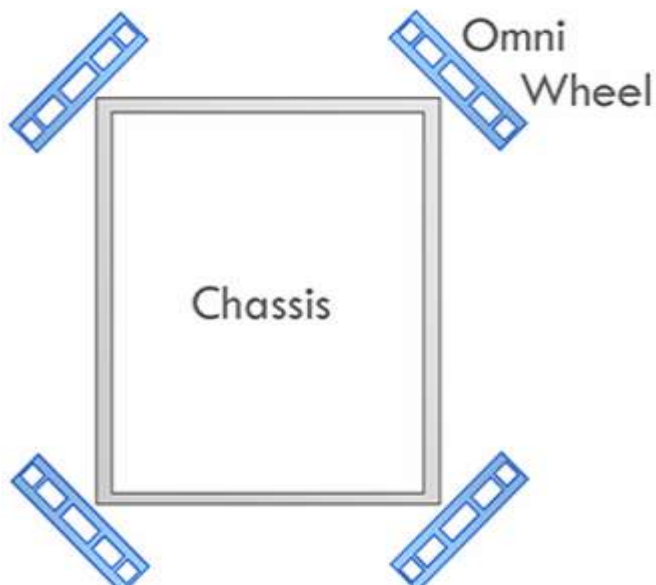
进一步平滑，使之适用于车辆跟踪控制。

□ 优化目标：

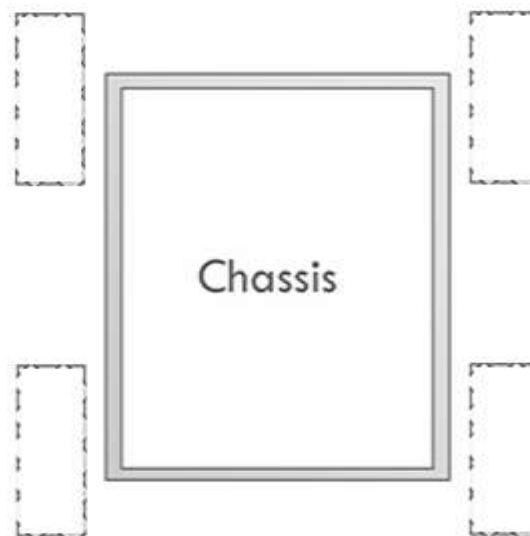


□ 智能汽车的路径规划难点

- (1) 车辆本身是受约束的动力学系统



完整式机器人
(三自由度独立运动)

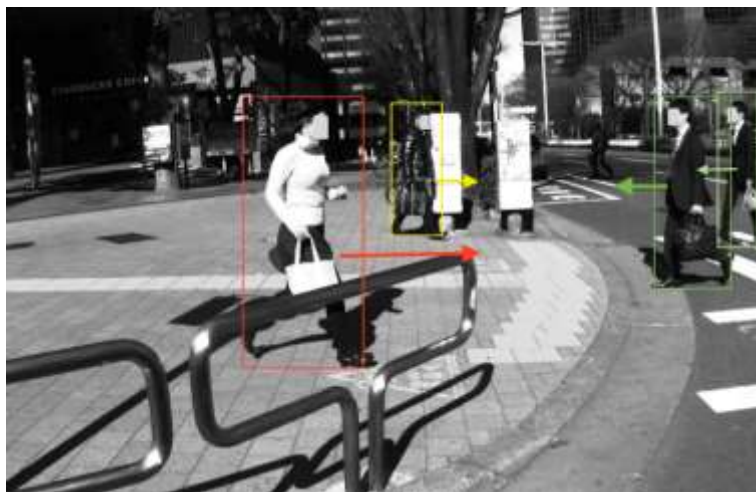


非完整式机器人
(三自由度不能独立运动)

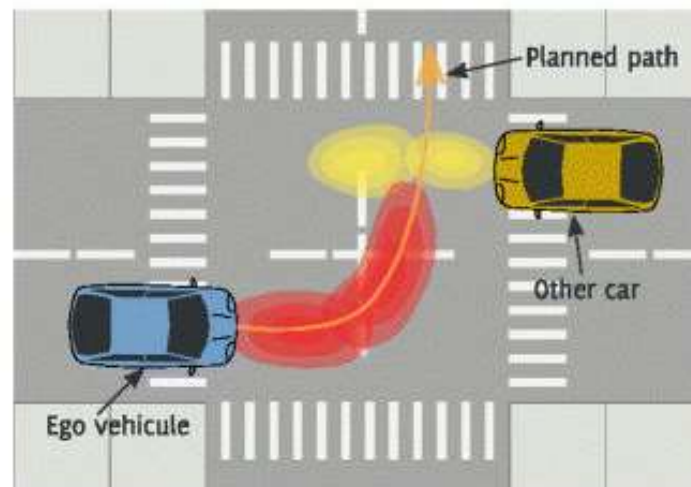
分解式决策：局部轨迹规划

□ 智能汽车的路径规划难点

- (2) 环境本身是动态的（车辆、行人）且存在随机不确定性



行人的复杂性

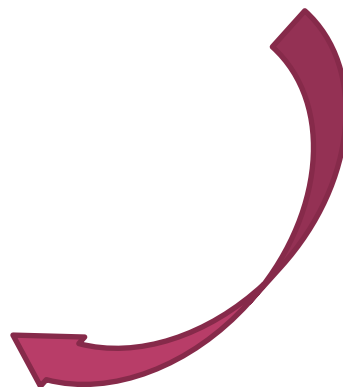
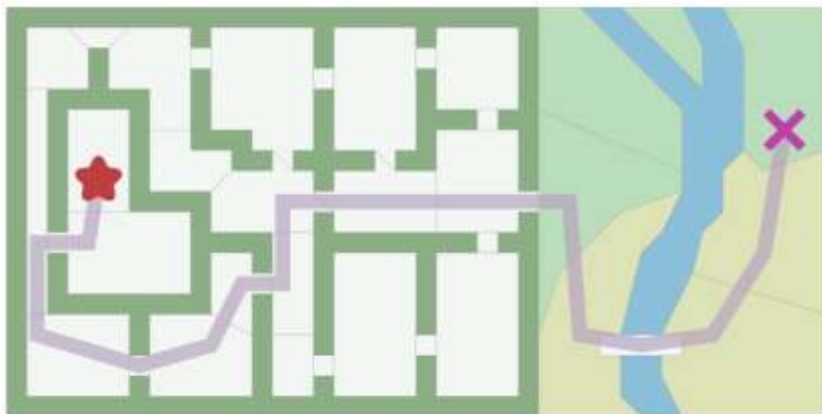
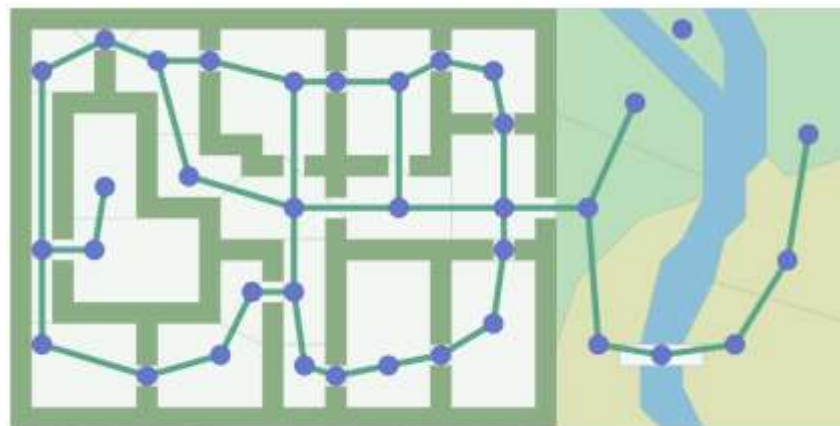


车辆运动的复杂性

分解式决策：局部轨迹规划

□ 地图网格化

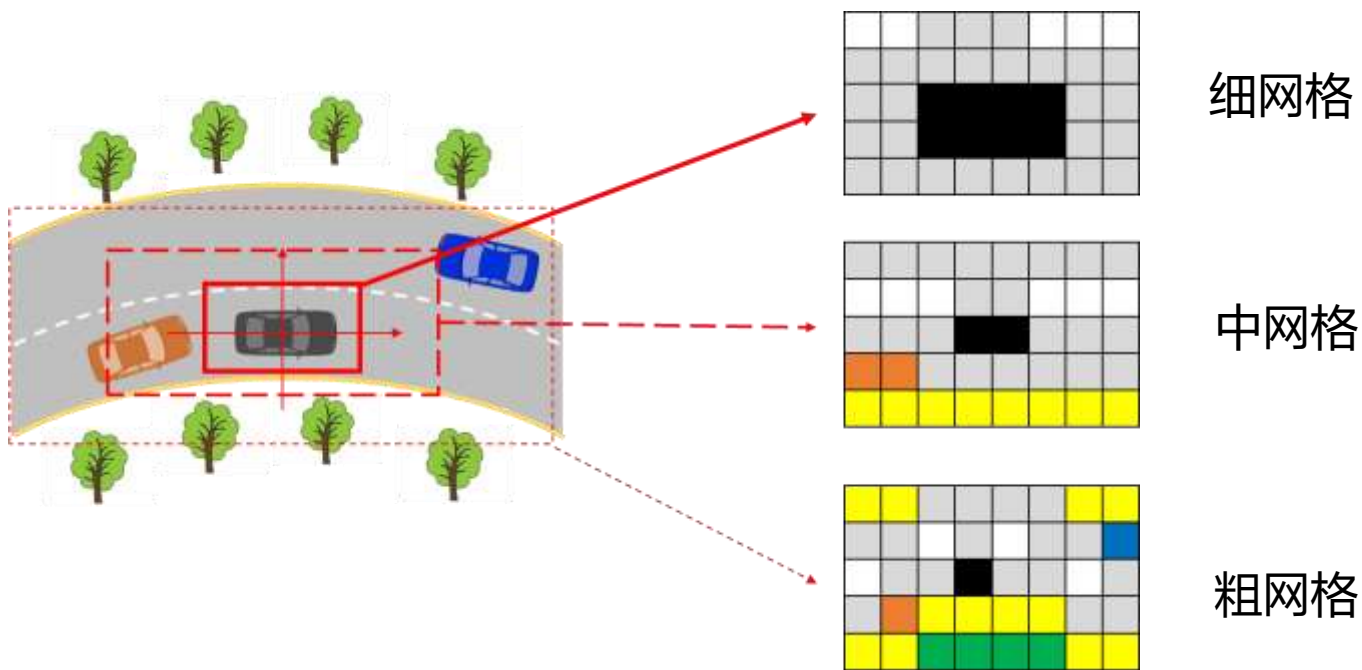
- 将路面划分成小型网格
- 网格大小根据需求调整
- 将网格节点作为不同位置点
- 将网格边线作为节点与节点间的链接路线
- 轨迹规划问题转化为图论中的最短路径问题



分解式决策：局部轨迹规划

□ 网格化难点：

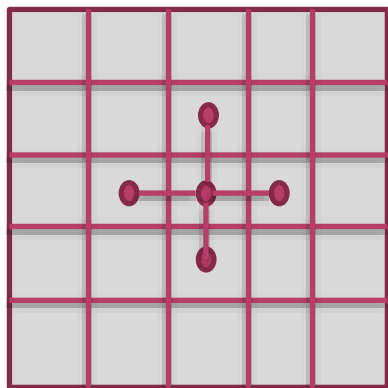
- 如何找到网格的大小，网格的大小越小，对环境的表现就越准确。
- 使用较小的网格将导致记忆空间和搜索范围的指数增长。



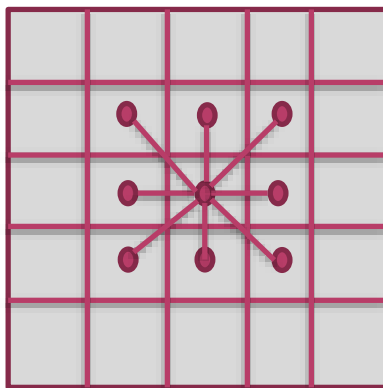
分解式决策：局部轨迹规划

□ 单元格之间的关系

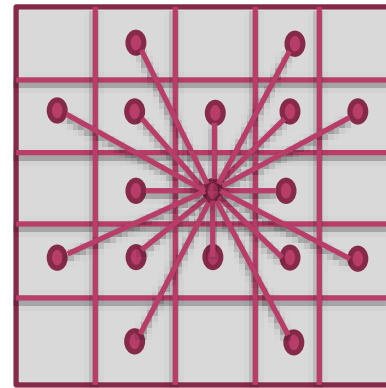
- 各单元格之间的联系分为四连接、八连接、十六连接
- 四连接
 - 表明当前栅格可以到达与之相邻的四个栅格节点
- 八连接
 - 表明当前栅格可以到达与之相邻的八个栅格节点
- 十六连接
 - 表明当前栅格可以到达与之相邻的十六个栅格节点



四连接



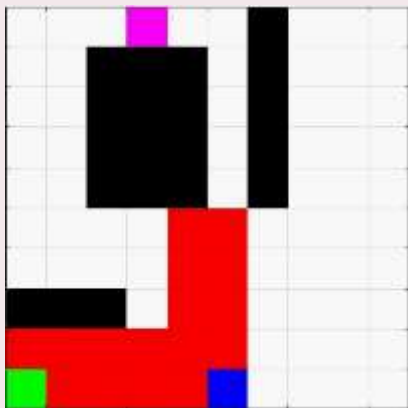
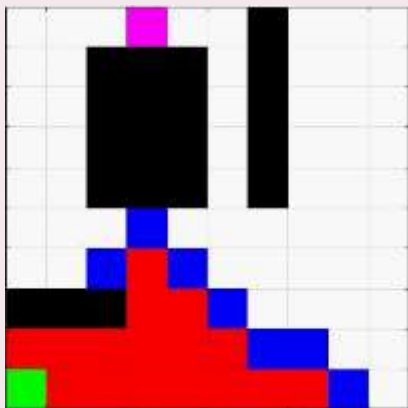
八连接



十六连接

分解式决策：局部轨迹规划

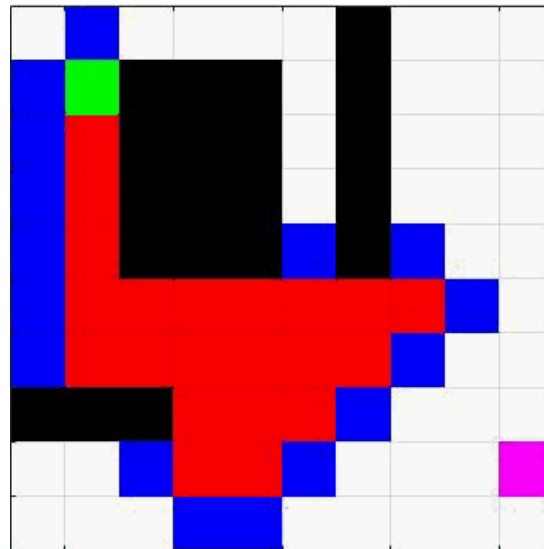
□ 基于网格化地图的搜索算法

	深度优先算法	广度优先算法
原理	从起点开始，沿一条路径一直搜索到底，可能需要重复搜索。	从起点开始，以不断扩散的方式来遍历整个地图。
搜索效率	较低	较高
适用场景	节点数目少	节点数目多
示意图		

分解式决策：局部轨迹规划

□ 启发式搜索

- 利用问题拥有的启发信息和相应启发函数具体引导搜索，以减少搜索范围、降低问题复杂度。
- 启发式广度优先算法：

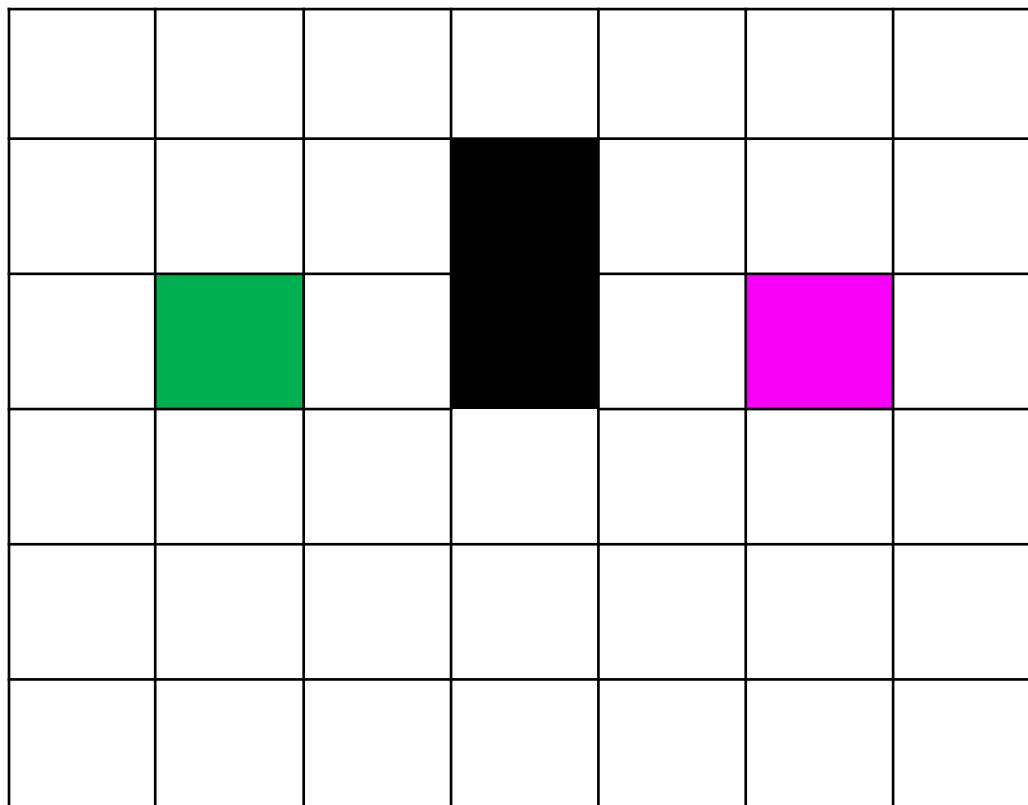


A*搜索算法

分解式决策：局部轨迹规划

□ A*搜索算法示例

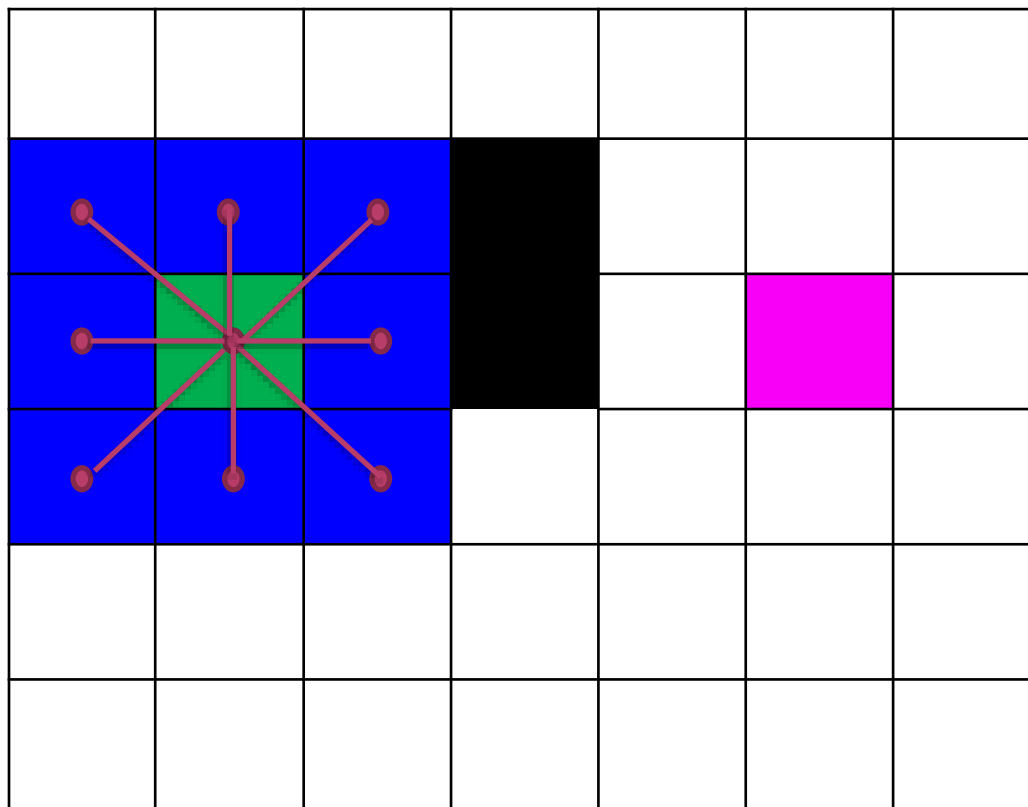
- 从**起点（绿）**，通过可行节点（白），越过**障碍物（黑）**，到达**终点（粉）**，寻找**最短轨迹**。



分解式决策：局部轨迹规划

□ A*搜索算法示例

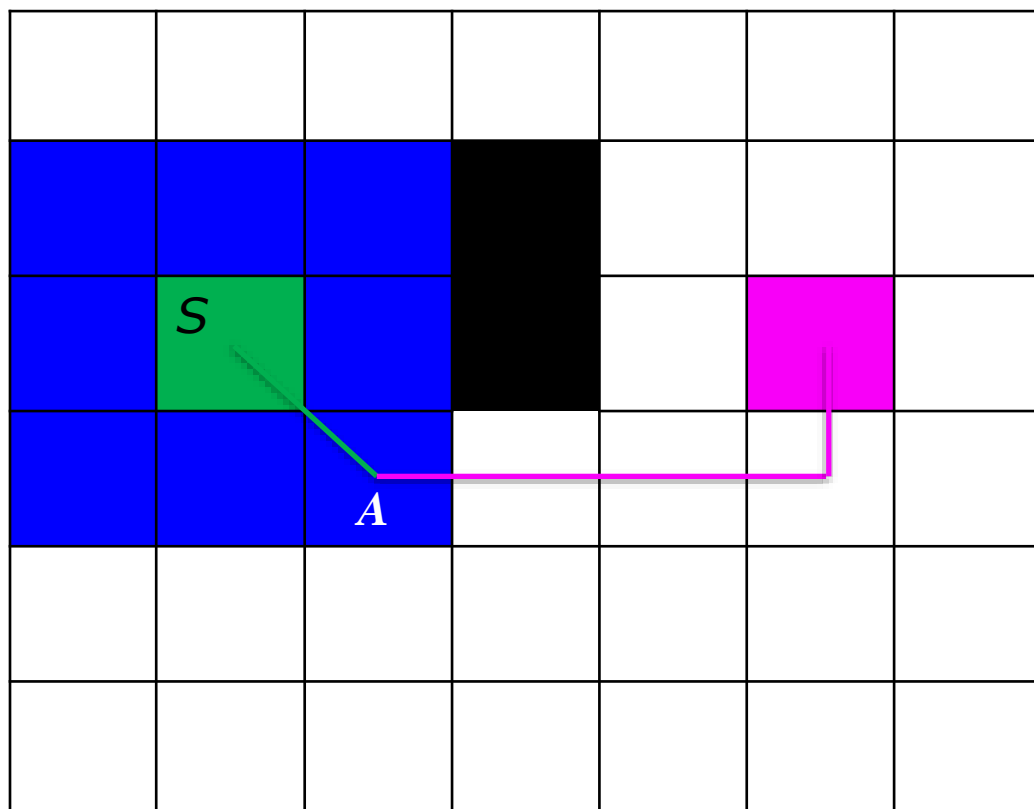
- 8连接**可行方向（棕）**，可到达**相邻节点（蓝）**。



分解式决策：局部轨迹规划

□ A*搜索算法示例

- 启发式代价函数: $f(x) = g(x) + h(x)$
 - $g(x)$ ——节点 x 到起点的 shortest 路程或轨迹路程（欧式距离）
 - $h(x)$ ——不考虑障碍物，节点 x 到终点的最短路程（曼哈顿距离）



$$g(A) = \sqrt{10^2 + 10^2} \approx 14$$

$$h(A) = 30 + 10 = 40$$



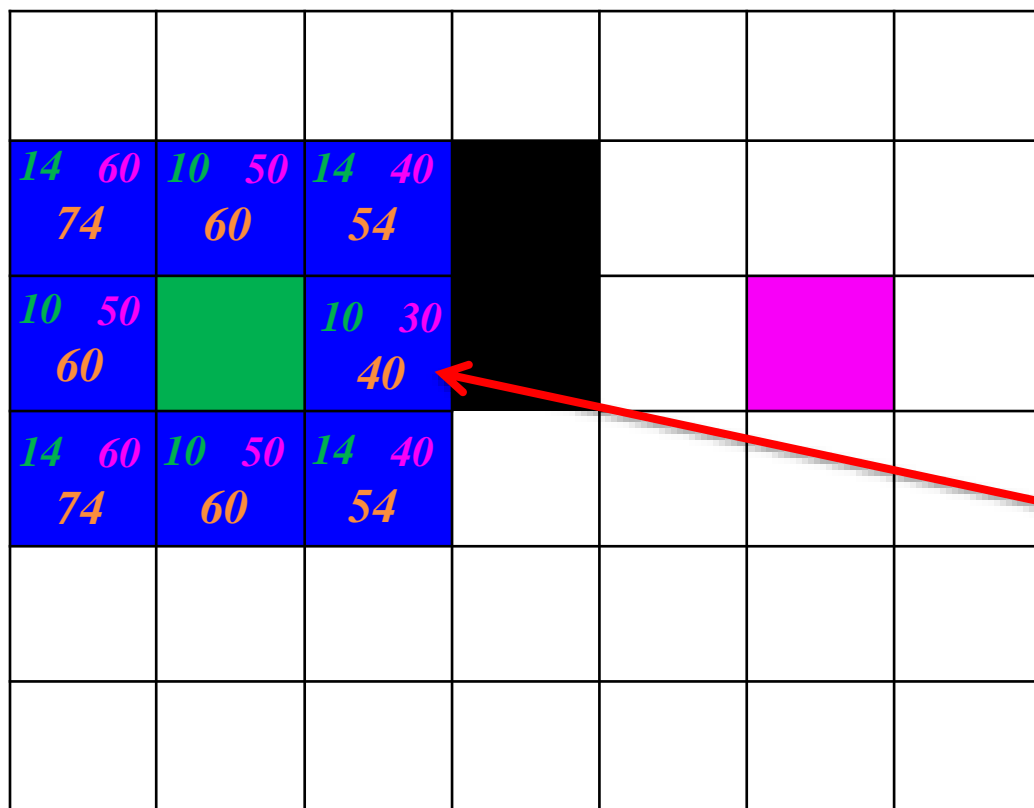
$$f(A) = g(A) + h(A) = 54$$

单元格边长=10

分解式决策：局部轨迹规划

□ A*搜索算法示例

- 启发式代价函数: $f(x) = g(x) + h(x)$
 - $g(x)$ ——节点 x 到起点的 shortest 路程或轨迹路程
 - $h(x)$ ——不考虑障碍物, 节点 x 到终点的最 short 路程



$$\min \{f(x), x \in \text{子节点}\} \\ = 40$$

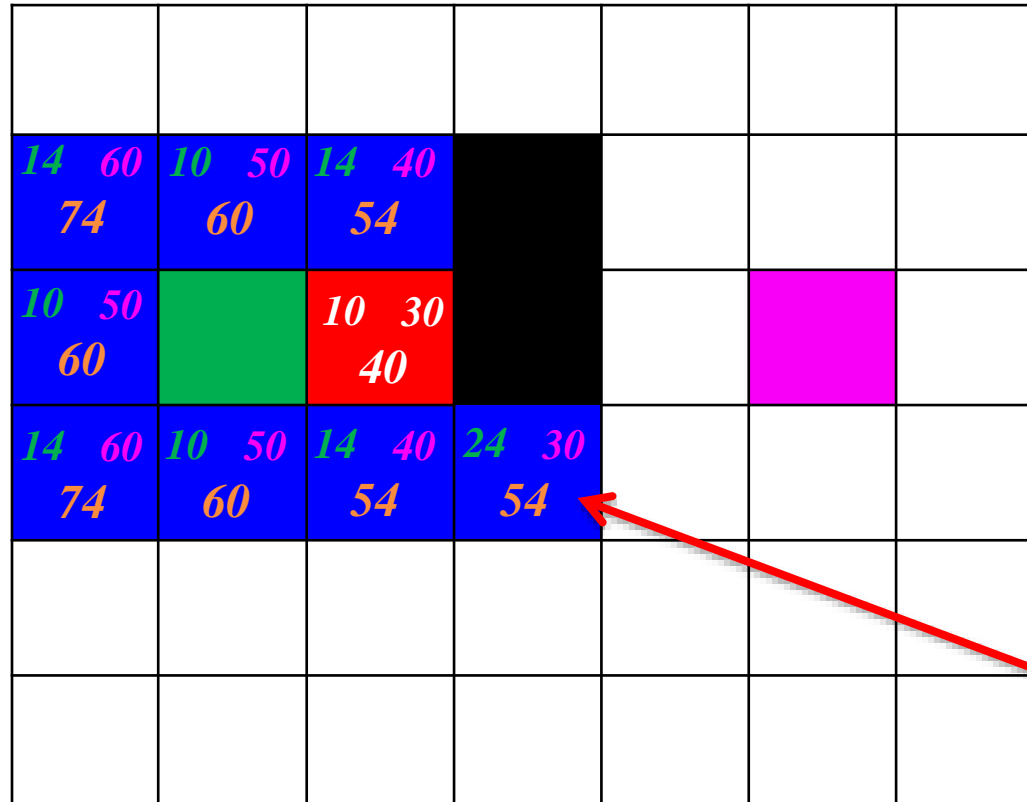


取该点为审查节点

分解式决策：局部轨迹规划

□ A*搜索算法示例

- 启发式代价函数: $f(x) = g(x) + h(x)$
 - $g(x)$ ——节点 x 到起点的 shortest 路程或轨迹路程
 - $h(x)$ ——不考虑障碍物, 节点 x 到终点的最 short 路程



添加新的子节点

↓ $g(x)$ 为轨迹路程

$$\min \{f(x), x \in \text{子节点}\} \\ = 54$$

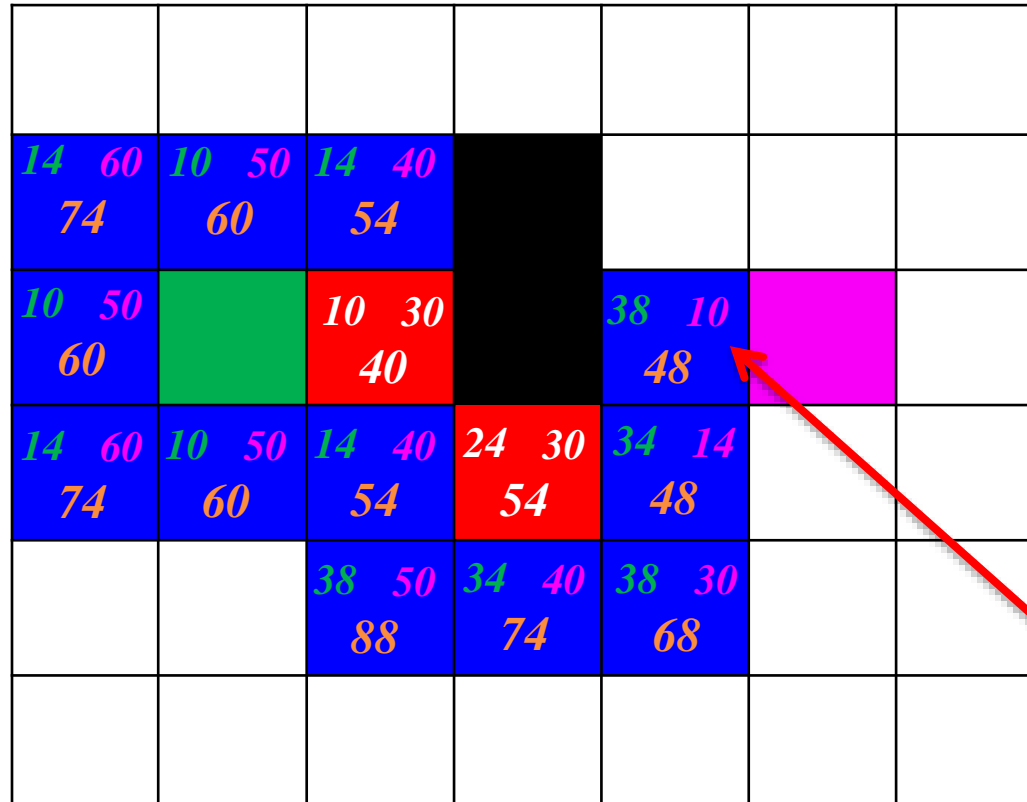
↓ $\min \{h(x)\}$

添加新的审查节点

分解式决策：局部轨迹规划

□ A*搜索算法示例

- 启发式代价函数: $f(x) = g(x) + h(x)$
 - $g(x)$ ——节点 x 到起点的 shortest 路程或轨迹路程
 - $h(x)$ ——不考虑障碍物, 节点 x 到终点的最 short 路程



添加新的子节点

↓ $g(x)$ 为轨迹路程

$$\min \{f(x), x \in \text{子节点}\} \\ = 48$$

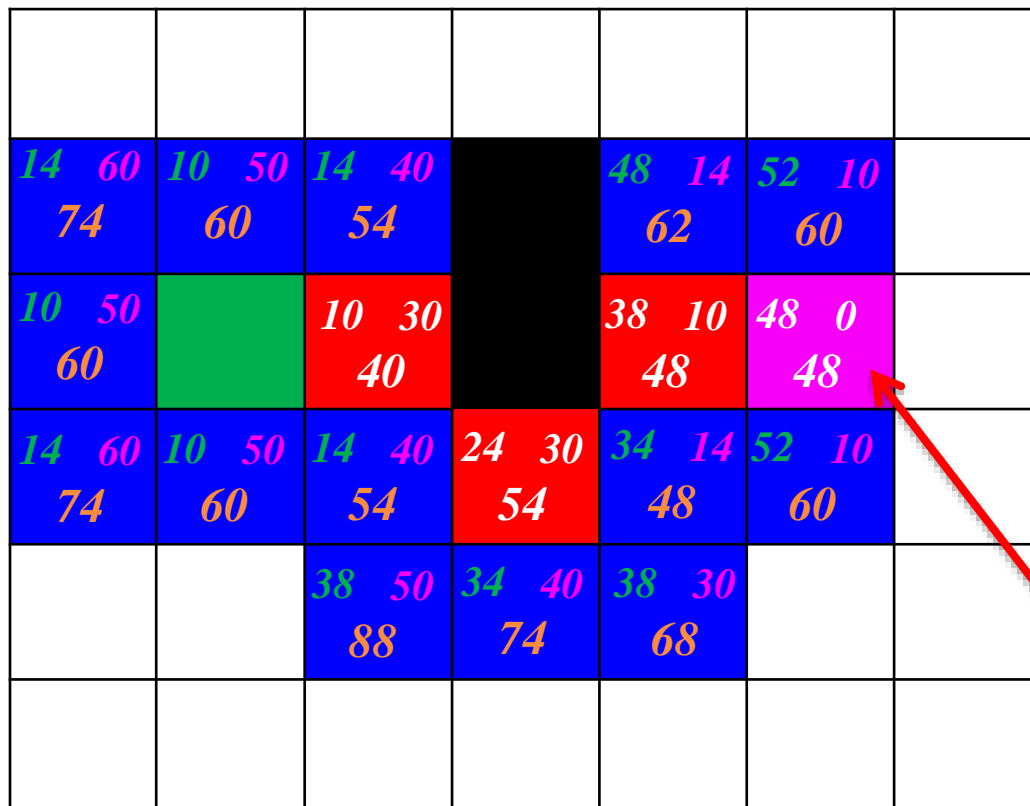
↓ $\min \{h(x)\}$

添加新的审查节点

分解式决策：局部轨迹规划

□ A*搜索算法示例

- 启发式代价函数: $f(x) = g(x) + h(x)$
 - $g(x)$ ——节点 x 到起点的 shortest 路程或轨迹路程
 - $h(x)$ ——不考虑障碍物, 节点 x 到终点的最 short 路程



添加新的子节点

↓ $g(x)$ 为轨迹路程

$$\min \{f(x), x \in \text{子节点}\} \\ = 48$$

↓ $\min \{h(x)\}$

添加新的审查节点

分解式决策：局部轨迹规划

□ A*搜索算法示例

- 启发式代价函数: $f(x) = g(x) + h(x)$
 - $g(x)$ ——节点 x 到起点的 shortest 路程或轨迹路程
 - $h(x)$ ——不考虑障碍物, 节点 x 到终点的最 short 路程

14 60 74	10 50 60	14 40 54		48 14 62	52 10 60	
10 50 60		10 30 40		38 10 48		
14 60 74	10 50 60	14 40 54	24 30 54	34 14 48	52 10 60	
		38 50 88	34 40 74	38 30 68		

新审查节点为终点



搜索结束



得到最短轨迹

本课小结

□ (a) 交通情景理解

- 感知 → 认知 → 理解

□ (b) 参与者运动预测

- 开环轨迹递推模型
- 驾驶员闭环递推模型

□ (c) 驾驶行为选择

- 任务制定 → 行为选择（有限状态机）

□ (d) 局部路径规划

- 环境建模（地图网格化）→ 轨迹搜索（A*算法）



谢谢大家!