

# Freeway Problem

陈天垚

PPCA

# 问题重述

## 高速公路规则

- 所有车都靠右行驶
- 当前车变道的時候才会向左变道

# 问题重述

## 高速公路规则

- 所有车都靠右行驶
- 当你要超车的时候才能向左换道

# 问题重述

## 高速公路规则

- 所有车都靠右行驶
- 当你要超车的时候才能向左换道

# 问题重述

## 高速公路规则

- 所有车都靠右行驶
- 当你要超车的时候才能向左换道

# 问题重述

- 如何建模模拟在固定规则下高速公路车流情况？
- 如何评价一个公路规则的优劣？
- 还有更好的规则吗？
- 向右行使的国家的规则能够直接镜像对称到向左行驶的国家吗？

# 问题重述

- 如何建模模拟在固定规则下高速公路车流情况？
- 如何评价一个公路规则的优劣？
- 还有更好的规则吗？
- 向右行使的国家的规则能够直接镜像对称到向左行驶的国家吗？

# 问题重述

- 如何建模模拟在固定规则下高速公路车流情况？
- 如何评价一个公路规则的优劣？
- 还有更好的规则吗？
- 向右行使的国家的规则能够直接镜像对称到向左行驶的国家吗？



# 问题重述

- 如何建模模拟在固定规则下高速公路车流情况？
- 如何评价一个公路规则的优劣？
- 还有更好的规则吗？
- 向右行使的国家的规则能够直接镜像对称到向左行驶的国家吗？

# 问题重述

- 如何建模模拟在固定规则下高速公路车流情况？
- 如何评价一个公路规则的优劣？
- 还有更好的规则吗？
- 向右行使的国家的规则能够直接镜像对称到向左行驶的国家吗？

# 元胞自动机

- 一堆节点和一个时间轴
- 节点和节点之间有拓扑关系
- 每个时刻，每个节点观察周围节点的状态，调整自己在下一个时刻的状态
- 举例：细胞游戏、经济系统。

# 元胞自动机

- 一堆节点和一个时间轴
- 节点和节点之间有拓扑关系
- 每个时刻，每个节点观察周围节点的状态，调整自己在下一个时刻的状态
- 举例：细胞游戏、经济系统。

# 元胞自动机

- 一堆节点和一个时间轴
- 节点和节点之间有拓扑关系
- 每个时刻，每个节点观察周围节点的状态，调整自己在下一个时刻的状态
- 举例：细胞游戏、经济系统。

# 元胞自动机

- 一堆节点和一个时间轴
- 节点和节点之间有拓扑关系
- 每个时刻，每个节点观察周围节点的状态，调整自己在下一个时刻的状态
- 举例：细胞游戏、经济系统。

# 元胞自动机

- 一堆节点和一个时间轴
- 节点和节点之间有拓扑关系
- 每个时刻，每个节点观察周围节点的状态，调整自己在下一个时刻的状态
- 举例：细胞游戏、经济系统。

# 模型解析

## 总述

Step 1. 调整速度

Step 2. 调整位置

Step 3. 换道判断



# 模型解析

## 总述

Step 1. 调整速度

Step 2. 调整位置

Step 3. 换道判断

# 模型解析

## 总述

Step 1. 调整速度

Step 2. 调整位置

Step 3. 换道判断

# 模型解析

## 总述

Step 1. 调整速度

Step 2. 调整位置

Step 3. 换道判断

# 模型解析

## 调整速度

Step 1. 如果速度还没有达到预期，那么就加一格速度

Step 2. 如果按照一定概率  $p_{slow}$  减一格速度

Step 3. 如果与前面车的距离太近，那么就减到安全速度

# 模型解析

## 调整速度

Step 1. 如果速度还没有达到预期，那么就加一格速度

Step 2. 如果按照一定概率  $p_{slow}$  减一格速度

Step 3. 如果与前面车的距离太近，那么就减到安全速度

# 模型解析

## 调整速度

Step 1. 如果速度还没有达到预期，那么就加一格速度

Step 2. 如果按照一定概率  $p_{slow}$  减一格速度

Step 3. 如果与前面车的距离太近，那么就减到安全速度

# 模型解析

## 调整速度

Step 1. 如果速度还没有达到预期，那么就加一格速度

Step 2. 如果按照一定概率  $p_{slow}$  减一格速度

Step 3. 如果与前面车的距离太近，那么就减到安全速度

# 模型解析

## 调整位置

- 位置 + 速度（乘单位时间间隔）



# 模型解析

## 调整位置

- 位置 + 速度（乘单位时间间隔）

# 模型解析

## 换道判断——向左换道

Condition 1. 速度还没有达到预期

Condition 2. 左前方的空间比前面的空间大

Condition 3. 与左后方的车距大于左后方的车速

- 符合以上条件后，按照一定概率  $p_{left}$  进行换道

# 模型解析

## 换道判断——向左换道

Condition 1. 速度还没有达到预期

Condition 2. 左前方的空间比前面的空间大

Condition 3. 与左后方的车距大于左后方的车速

- 符合以上条件后，按照一定概率  $p_{left}$  进行换道

# 模型解析

## 换道判断——向左换道

Condition 1. 速度还没有达到预期

Condition 2. 左前方的空间比前面的空间大

Condition 3. 与左后方的车距大于左后方的车速

- 符合以上条件后，按照一定概率  $p_{left}$  进行换道

# 模型解析

## 换道判断——向左换道

Condition 1. 速度还没有达到预期

Condition 2. 左前方的空间比前面的空间大

Condition 3. 与左后方的车距大于左后方的车速

- 符合以上条件后，按照一定概率  $p_{left}$  进行换道

# 模型解析

## 换道判断——向左换道

Condition 1. 速度还没有达到预期

Condition 2. 左前方的空间比前面的空间大

Condition 3. 与左后方的车距大于左后方的车速

- 符合以上条件后，按照一定概率  $p_{left}$  进行换道

# 模型解析

## 换道判断——向右换道

Condition 1. 没有向左换道

Condition 2. 右前方的空间比前面的空间大

Condition 3. 与右后方的车距大于右后方的车速

- 符合以上条件后，按照一定概率  $p_{right}$  进行换道

# 模型解析

## 换道判断——向右换道

Condition 1. 没有向左换道

Condition 2. 右前方的空间比前面的空间大

Condition 3. 与右后方的车距大于右后方的车速

- 符合以上条件后，按照一定概率  $p_{right}$  进行换道



# 模型解析

## 换道判断——向右换道

Condition 1. 没有向左换道

Condition 2. 右前方的空间比前面的空间大

Condition 3. 与右后方的车距大于右后方的车速

- 符合以上条件后，按照一定概率  $p_{right}$  进行换道

# 模型解析

## 换道判断——向右换道

Condition 1. 没有向左换道

Condition 2. 右前方的空间比前面的空间大

Condition 3. 与右后方的车距大于右后方的车速

- 符合以上条件后，按照一定概率  $p_{right}$  进行换道

# 模型解析

## 换道判断——向右换道

Condition 1. 没有向左换道

Condition 2. 右前方的空间比前面的空间大

Condition 3. 与右后方的车距大于右后方的车速

- 符合以上条件后，按照一定概率  $p_{right}$  进行换道

# 实现细节

## 优化搜索范围

- 搜索六个方向汽车，稀疏图的话范围太大
- 考虑到搜索范围在车速之外就没有用了
- 设定一个值 *out\_of\_sight*，表示视线范围，稍微比最大车速大一些
- 在 *out\_of\_sight* 范围之内搜索即可

# 实现细节

## 优化搜索范围

- 搜索六个方向汽车，稀疏图的话范围太大
- 考虑到搜索范围在车速之外就没有用了
- 设定一个值 *out\_of\_sight*，表示视线范围，稍微比最大车速大一些
- 在 *out\_of\_sight* 范围之内搜索即可

# 实现细节

## 优化搜索范围

- 搜索六个方向汽车，稀疏图的话范围太大
- 考虑到搜索范围在车速之外就没有用了
- 设定一个值 *out\_of\_sight*，表示视线范围，稍微比最大车速大一些
- 在 *out\_of\_sight* 范围之内搜索即可

# 实现细节

## 优化搜索范围

- 搜索六个方向汽车，稀疏图的话范围太大
- 考虑到搜索范围在车速之外就没有用了
- 设定一个值 *out\_of\_sight*，表示视线范围，稍微比最大车速大一些
- 在 *out\_of\_sight* 范围之内搜索即可

# 实现细节

## 优化搜索范围

- 搜索六个方向汽车，稀疏图的话范围太大
- 考虑到搜索范围在车速之外就没有用了
- 设定一个值 *out\_of\_sight*，表示视线范围，稍微比最大车速大一些
- 在 *out\_of\_sight* 范围之内搜索即可



# 效果

- 详见程序