6. Intelligent transportation: Smart intersections

金力 Li Jin
li.jin@sjtu.edu.cn
上海交通大学密西根学院
Shanghai Jiao Tong University UM Joint Institute

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

Outline

- Technological basis
  - Connected and autonomous vehicles
  - Vehicle platooning
- Simplified formulation
  - Modeling
  - Decision making
  - Final project option 1
- State-of-the-art formulation
  - Modeling
  - Decision making
  - String stability

Outline

- Background
  - Signalized & unsignalized intersections
  - Connected & autonomous vehicles
  - Vehicle-to-infrastructure connectivity
- Trajectory planning
  - For a single vehicle
  - For multiple vehicles
- Vehicle sequencing
  - Modeling & formulation
  - Optimization

Signalized intersection

Signalized intersection: Fixed cycle design

- Data:
  - Traffic demand in each direction
  - Saturation rate & response time
- Decision variables
  - Green ratio/time in each direction
- Constraint
  - Safety (no simultaneous greens)
  - Technical constraint (switching frequency)
- Objective
  - Ensure bounded waiting time #
  - Minimize average waiting time

Signalized intersection: Adaptive cycles

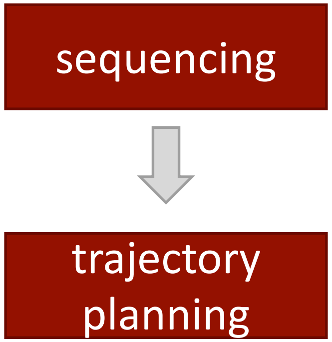
- Data:
  - Saturation rate & response time
  - Real-time traffic state on each lane
- Decision variables
  - Signaling in the next decision period
  - Or, policy for signaling
- Constraint
  - Safety (no simultaneous greens)
  - Technical constraint (switching frequency)
- Objective
  - Ensure bounded waiting time
  - Minimize expected waiting time

Unsignalized intersection

- Typically, vehicles are supposed to stop as they arrive at the intersection.
- Then, vehicles cross according to convention or rule.
- Could be chaotic…
- http://heze.dzwww.com/qx/yc/201908/t20190810_17039691.htm

Hierarchical control system

- Upper level: sequencing
  - A centralized controller (e.g. RSU) determines the sequence of CAVs
  - Sequencing leads to time windows for each CAV to cross
- Lower level: trajectory planning
  - A CAV plans its trajectory to ensure crossing during the allocated time window (absorbing delay en route)
  - Vehicle following or coordination needed

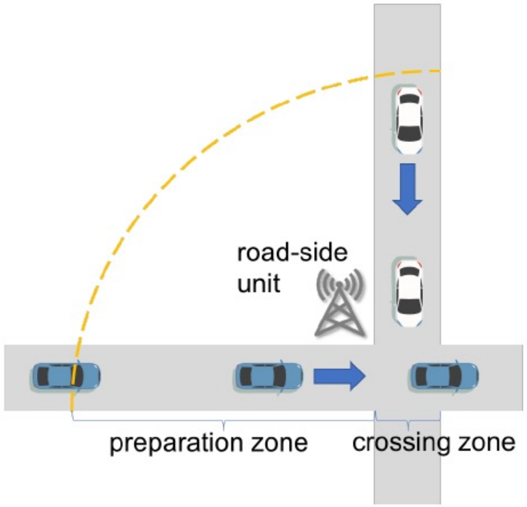


Outline

- Background
  - Signalized & unsignalized intersections
  - Connected & autonomous vehicles
  - Vehicle-to-infrastructure connectivity
- Trajectory planning
  - For a single vehicle
  - For multiple vehicles
- Vehicle sequencing
  - Modeling & formulation
  - Optimization

Trajectory for a single vehicle

- We have seen the term “reference trajectory” for multiple times.
- How do we generate that?
- Consider a vehicle that enters the preparation zone at time $t = 0$.
- Initial speed $v[0] \in \mathbb{R}_{\geq 0}$.
- Suppose that the RSU requires it to enter the crossing zone at time t_1.
- Let $u[t]$ be the acceleration.



Trajectory for a single vehicle

- At each time step, a cost of $u^2[t]$ is induced.
- You can interpret this cost as a measure for comfort.
- Less acceleration/deceleration -> higher comfort.
- Hence, we need to determine

$$u[0], u[1], \dots, u[t_1]$$

  such that

$$\sum_{t=0}^{t_1} u^2[t]$$

  is minimized.
- Note: this is open-loop decision making, since we are planning the reference trajectory.

Trajectory for a single vehicle

Data:

- Specified time for crossing t_1
- Initial position $x[0] = 0$ and initial speed $v[0] = v_0$.

Decision variable (essentially):

- Time series of acceleration $u[t]$, $t = 0,1,2, \dots, t_1 - 1$.

Constraint:

- Kinematics:

$$x[t+1] = x[t] + v[t]\delta, v[t+1] = v[t] + u[t]\delta.$$

- Crossing on time: $x[t_1] = L$.

Objective: minimize $\sum_{t=0}^{t_1} u^2[t]$

Formulation of optimization problem

These four elements are essential for every optimization or optimal control problem:

Data:

- Information based on which you make decisions.

Decision variables:

- Quantity that you can select; influences outcome

Constraints:

- Restrictions on allowable decisions.

Objective function:

- Evaluation of various outcomes

Trajectory for a single vehicle

- Trajectory planning in standard optimization presentation:

$$\min \sum_{t=0}^{t_1} u^2[t]$$

$$\text{s.t.}\quad x[t+1] = x[t] + v[t]\delta,\ \ t = 0,1,\dots,t_1 - 1,$$

$$v[t+1] = v[t] + u[t]\delta,\ \ t = 0,1,\dots,t_1 - 1,$$

$$x[t_1] = L.$$

- **Note 1**: never forget the range for t as a generic time.
- **Note 2**: strictly speaking, $x[t]$ and $v[t]$, in addition to $u[t]$, are not data and are also “decision variables” in the above formulation in a technical sense. However, $x[t]$ and $v[t]$ are essentially determined by $u[t]$.

Control-theoretic formulation

- In the above formulation, $\{u[t];\ \forall t\}$ are all independent decision variables.
- Alternatively, we can make the following restriction:
$$u[t] = \mu(x[t], v[t]).$$
- That is, instead of selecting $u[t]$ independently, we force them to be dependent of the state.
- The function $\mu: \mathbb{R}^2 \to \mathbb{R}$ is actually a control policy.
- Hence, the formulation on the previous slide is said to be policy-free.
- Compared with the policy-based formulation.
- Tradeoff between computation load and performance.

Optimal control problem

- Consider a linear system
$$\begin{bmatrix} x[t+1] \\ v[t+1] \end{bmatrix} = \begin{bmatrix} 1 & \delta \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x[t] \\ v[t] \end{bmatrix} + \begin{bmatrix} 0 \\ \delta \end{bmatrix} u[t].$$
- Cost over one time step = $u^2[t]$.
- Initial condition $0, v_0$.
- Cumulative cost $J_t(x_0, v_0) = \sum_{\tau=0}^{t} u^2[\tau]$.
- Objective: find $u[t]$ for all t to minimize cumulative cost.
- This is called linear quadratic regulation (LQR):

1. Linear dynamics,
2. Quadratic objective function.

Optimal control problem

- In CT:

$$\begin{bmatrix}\dot{x}\\ \dot{v}\end{bmatrix} = \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}\begin{bmatrix}x\\ v\end{bmatrix} + \begin{bmatrix}0 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}0\\ u\end{bmatrix}$$

- u = acceleration (control input)
- Boundary conditions: $x(0) = 0, x(t_1) = L, v(0) = v_0$.
- Instantaneous cost $L(u) = u^2$.
- Cumulative cost

$$J_t = \int_{\tau=0}^{t} u^2(\tau)d\tau\,.$$

- Also called linear quadratic regulation.

Generic formulation

- Consider an LTI system
$$\dot{x} = Ax + Bu$$
- Suppose that we want to drive the system from the initial state $x(0)$ to the target state $x(T) = 0$
- Instantaneous cost
$$L(t) = \frac{1}{2}x^T(t)Qx(t) + \frac{1}{2}u^T(t)Ru(t)$$
- The cost induced during the control process is given by
$$\frac{1}{2}x^T(T)Qx(T) + \int_{s=0}^{T} L(s)ds$$
- Linear feedback $u = -Kx$

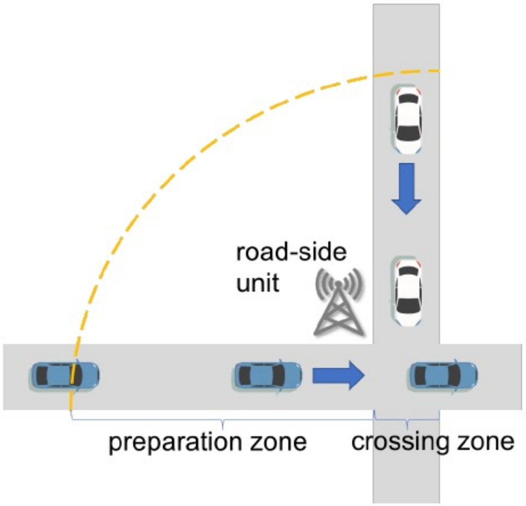
[Not required] LQR optimal control

- Linear-quadratic regulator (LQR)
- Design task: obtain the optimal gain matrix K for $u = -Kx$
- Conclusion:

1. Let P be the solution matrix to the matrix Riccati equation
$$PA + A^T P + Q - PBR^{-1}B^T P = 0.$$
1. Then the optimal K is given by
$$K = -R^{-1}B^T P.$$

- Design task: pick Q and R (i.e. cost function)
- Reference: Zhou, K., Doyle, J. C., & Glover, K. (1996). *Robust and optimal control* (Vol. 40, p. 146). New Jersey: Prentice hall.

Trajectory for multiple vehicles



- Consider a collection of vehicles

$$1,2,\dots,n.$$

- Each vehicle arrive at time

$$s_1, s_2, \dots, s_n.$$

- Crossing times $t_1, t_2, \dots, t_n$.
- Initial speeds $\phi_1, \phi_2, \dots, \phi_n$.
- For each vehicle i, we need to select the time series of acceleration $u_i[t]$ for $s_i \le t < t_i$.
- Similar to the single-vehicle, problem, but with significant differences in terms of formulation and presentation!

Trajectory for multiple vehicles

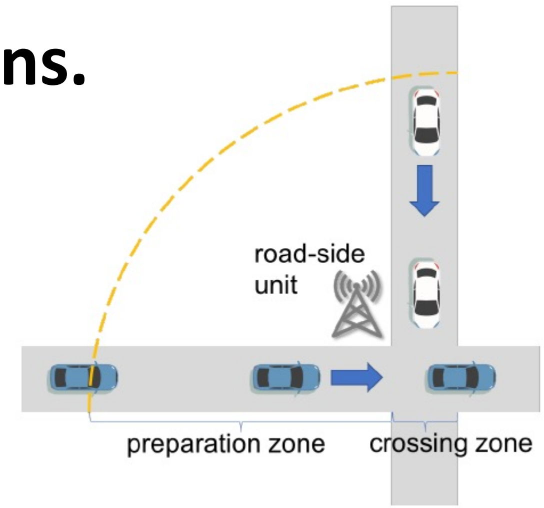
First complication 1: multiple directions.

- For ease of presentation, consider two orthogonal orbits only.
- Only keeping straight; no turning.
- Label these two directions as 1 & 2.
- So, we need to adjust the notations:
- For direction $k \in \{1,2\}$, there are n_k vehicles with

  Arrival times: $s_1^k, s_2^k, \dots, s_{n_k}^k$;

  Crossing times: $t_1^k, t_2^k, \dots, t_{n_k}^k$;

  Initial speeds: $\phi_1^k, \phi_2^k, \dots, \phi_{n_k}^k$.



Trajectory for multiple vehicles

First complication 2: safe distance.

- For vehicles from the same direction

$$x_{i-1}^k[t] - x_i^k[t] \geq d + hv_i^k[t],$$

for all $i = 1,2,\ldots,n_k$ and for $k = 1,2,$

for all t: $s_i^k \leq t \leq t_{i-1}^k$.

- For vehicles on different orbits,

$$\left|x_i^1[t] - x_j^2[t]\right| \geq d'$$

for all $1 \leq i \leq n_1$ and $1 \leq j \leq n_2$,

for all t: $\max\{t_i^1 - T, t_j^2 - T\} \leq t \leq \min\{t_i^1, t_j^2\}$.

- Need to know how to write the index conditions!

Trajectory for multiple vehicles

$$
\begin{aligned}
\text{min}\quad & \textstyle\sum_{t=0}^{T}\sum_k\sum_i \left(u_i^k[t]\right)^2 \\
\text{s.t.}\quad & x_i^k[t+1] = x_i^k[t] + v_i^k[t]\delta, \forall i, \forall t, \forall k, \\
& v_i^k[t+1] = v_i^k[t] + u_i^k[t]\delta, \forall i, \forall t, \forall k, \\
& x_{i-1}^k[t] - x_i^k[t] \geq d + hv_i^k[t], \forall i, \forall t, \forall k, \\
& \left|x_i^1[t] - x_j^2[t]\right| \geq d', \forall i, \forall j, \forall t, \\
& 0 \leq v_i^k[t] \leq \bar{v},\ -\bar{a} \leq u_i^k[t] \leq \bar{a}, \forall i, \forall t, \forall k, \\
& x_i^k\left[s_i^k\right] = 0,\ v_i^k\left[s_i^k\right] = \phi_i^k, \forall i, \forall k, \\
& x_i^k\left[t_i^k\right] = L, \forall i, \forall k.
\end{aligned}
$$

Size of problem

- In the above formulation, $\{u_i^k[t]; \forall i, \forall k, \forall t\}$ are all independent decision variables.
- If there are 100 vehicles and if every vehicle spends 50 time units (e.g. sec) in the system, then we have 5000 decision variables.
- Technically, $v_i^k[t]$ and $x_i^k[t]$ are also decision variables.
- Hence, the above formulation will involve 15,000 decision variables.
- Hence, this formulation is inefficient.
- In practice, this may not be problematic, since this is decision making on a longer time scale.

Outline

- Background
  - Signalized & unsignalized intersections
  - Connected & autonomous vehicles
  - Vehicle-to-infrastructure connectivity
- Trajectory planning
  - For a single vehicle
  - For multiple vehicles
- Vehicle sequencing
  - Modeling & formulation
  - Optimization

Hierarchical coordination

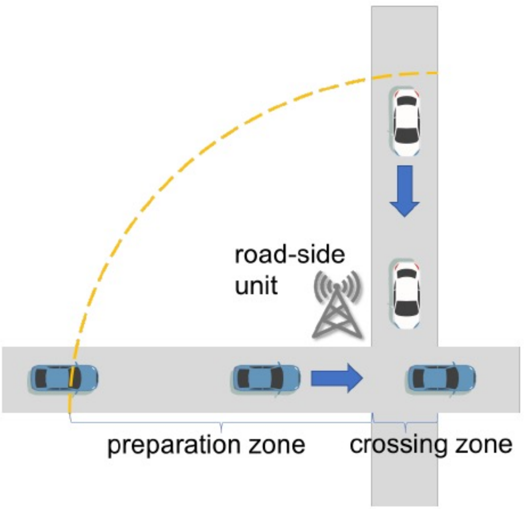
- So, how do we obtain t_i^k in the first place?
- This results from a higher-level decision-making mechanism:

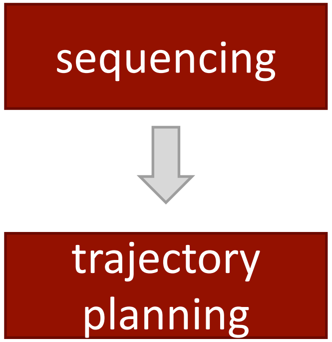
vehicle sequencing.

- A hierarchical decision-making framework
  - Upper level: scheduling/sequencing
  - Lower level: trajectory planning
- Such a hierarchical framework makes practical sense:
  - A centralized controller determines vehicle sequencing
  - Then, each vehicle determines its own trajectory to fulfill the designated sequencing

Sequencing problem

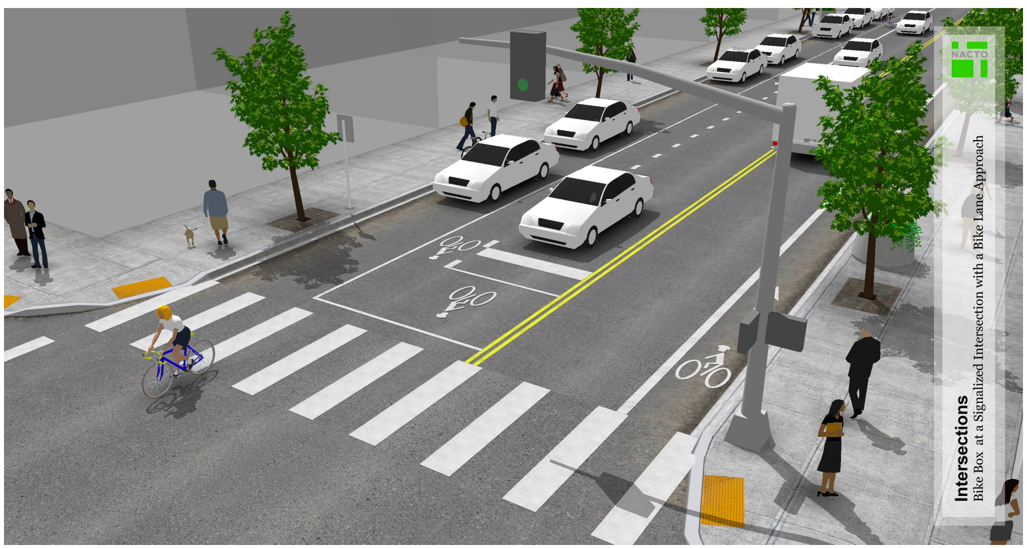
- Consider two CAVs (labeled 1 & 2) consecutively crossing the intersection
- Suppose CAV i crosses the intersection at time t_i
- What constraints are imposed on the crossing times?
  - If vehicle i enters the control zone at time s_i, it cannot cross until $s_i + \Delta$, where Δ is the nominal traverse time.
  - The crossing times t_1, t_2 should be staggered to meet safety condition.

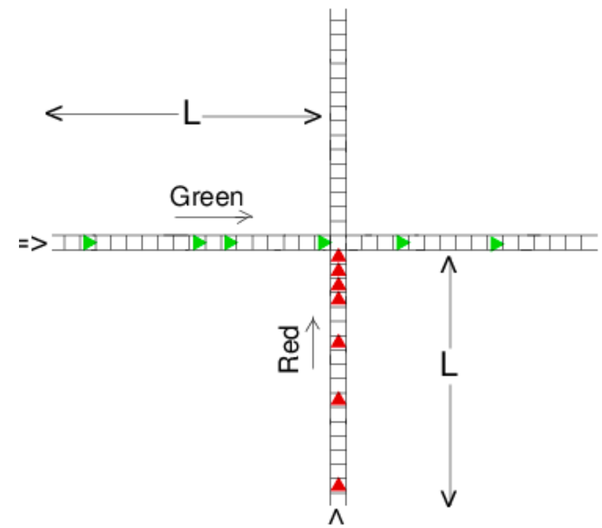




Sequencing problem

- Hence, we can forget about trajectories and only consider the arrival (entering) and departure (crossing) times.
- Scenario: a symmetric double one-way intersection
- No traffic signal; based on vehicle coordination.





Sequencing problem

- Since two directions are symmetric, we assume identical nominal traverse time Δ for all vehicles.
- Thus, we can "shift" the time axis a bit and proceed as if $\Delta = 0$ for all vehicles.
- Hence, we can formulate the sequencing problem as follows:

1. Data: arrival times s_i^k and minimal headway θ_{ij};
2. Decision variable: departure times t_i^k;
3. Constraints: FIFO in one direction, safety;
4. Objective: min total travel time.

Minimal headway

- If a class-j vehicle (vehicle 2) crosses the intersection after a class-i vehicle (vehicle 1), the headway therebetween should be no less than θ_{ij}.
- How to determine safe headway θ?
- Simple case:
  - Vehicle 1 & 2 are from the same direction
  - Minimal headway: as in platooning
- More complex case #
  - Vehicle 1 & 2 on intersecting orbits: safety constraint
  - Vehicle 1 & 2 on non-intersecting orbits: no constraint
- In our case, $\theta_{11} = \theta_{22} < \theta_{12} = \theta_{21}$.

First-in-first-out (FIFO)

- We assume that vehicles from the same direction cross on a FIFO basis.
- That is, no overtaking.
- Vehicles from different directions can also be forced to cross on a FIFO basis.
- But that will be more restrictive and questionable in terms of performance (i.e., travel time or fuel) improvement.
- However, FIFO is considered to be more fair than alternative sequencing policies.

Optimization formulation

- Suppose that we need to sequence $n_1 + n_2$ vehicles.
- Then we can determine the departure times as follows:

$$\min \sum_k \sum_i (t_i^k - s_i^k)$$

$$\text{s.t.} \quad t_i^k - s_i^k \geq 0, \forall i, \forall k,$$

$$t_i^k - t_{i-1}^k \geq \theta_{11}, \forall i, \text{(note that } \theta_{11} = \theta_{22}\text{)}$$

$$\left| t_i^1 - t_j^2 \right| \geq \theta_{12}, \forall i, j, \text{(note that } \theta_{12} = \theta_{21}\text{)}$$

- This formulation looks perfect but has two major drawbacks:

1. How can you obtain arrival times s_i^k?
2. What if the $(n_1 + n_2 + 1)$th vehicle arrives?

Problem formulation

- We now present a control-theoretic formulation.
- Two one-way orthogonal orbits without turning.
- State: $X_k(t) =$ # of CAVs waiting in direction k
- At each time step, a CAV arrives in direction k with probability $p_k \in [0,1]$
- At each time step, the intersection can discharge at most one CAV
  - Same-direction headway $\theta_{11} = \theta_{22} = 1$ [time step]
  - Orthogonal-direction headway $\theta_{12} = \theta_{21} = 2$ [time steps] (very fake number)
  - We can use an auxiliary dummy variable to formulate it

System dynamics

- A bit complex; fasten you seatbelt…
- System state $X(t) = \left[X_1[t], X_2[t]\right]^T \in \mathbb{Z}_{\geq 0}^2$
  - $X_k[t]$ = # of CAVs waiting in direction k
- The above is not enough, we need an auxiliary state $Y = \{0,1,2\}$, which is the "previous vehicle class"
  - $Y[t] = k$ if a class-k vehicle was discharged at t
  - $Y[t] = 0$ if no vehicle was discharged at t
- Action $A[t] \in \{1,2\}$
  - $A[t] = k$ essentially (but not exactly) means direction k is being discharged at time t,
  - If $X_1[t] = X_2[t] = 0$, $A[t]$ has no impact.

Queuing mechanism

- Let $\Delta X_k[t]$ be the # of vehicles arriving in direction k over one time step.
- Typically we cannot deterministically predict ΔX_k.
- Instead, we consider

$$\Delta X_k[t] = \begin{cases} 1 & \text{with probability } p_k, \\ 0 & \text{with probability } 1 - p_k. \end{cases}$$

- $X_k[t+1] =$

$$\begin{cases} (X_k[t] + \Delta X_k[t] - 1)_+ & \text{if } A[t] = k \text{ and } if\ Y[t] = k \text{ or } 0 \\ X_k(t) + \Delta X_k[t] & \text{otherwise} \end{cases}$$

- $(\cdot)_+$ represents the positive part of a function

$$(\xi)_+ = \begin{cases} \xi & \xi \geq 0 \\ 0 & o.w. \end{cases}$$

Stochastic process

- That is, $X_k[t]$ cannot be deterministically predicted.
- Instead, we characterize its distribution.
- In other words, given $X_k[t]$, we do not talk about $X_k[t+1]$, but we talk about
$$\Pr\{X_k[t+1] = x | X[t], A[t]\}, x \in \mathbb{Z}_{\geq 0}.$$
- The process $\{X[t]; t > 0\}$ (which takes values from $\mathbb{Z}_{\geq 0}^2$) is said to be a stochastic process.
- For stochastic processes, instead of considering the evolution of $X[t]$ directly, we consider the evolution of the distribution of $X[t]$.
- You can simulate with random number generators.

Discharging mechanism

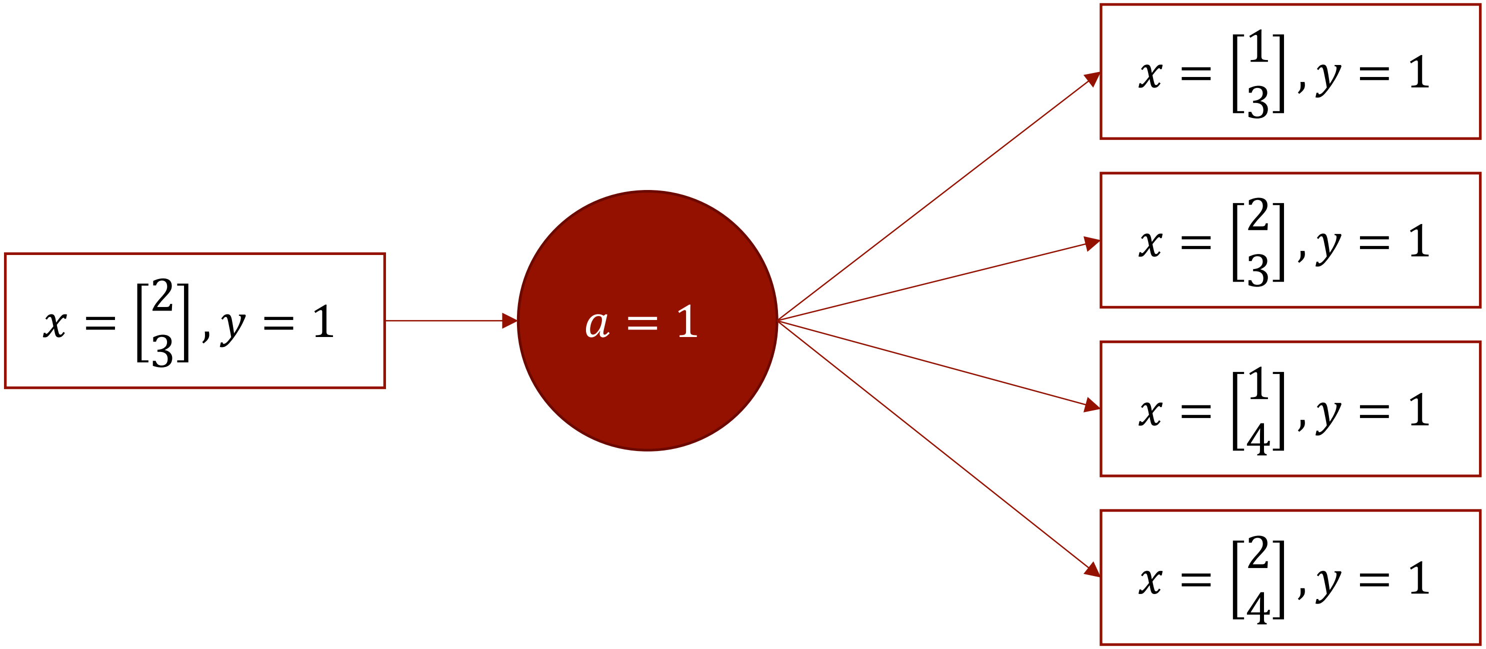
- If we discharged a class-1 vehicle at time t, we can immediately discharge another class-1 vehicle at time $t+1$:
$$Y[t]=1, Y[t+1]=1.$$
- If we discharged a class-1 vehicle at time t, we cannot discharge a class-2 vehicle until time $t+2$.
$$Y[t]=1, Y[t+1]=0, Y[t+2]=2.$$
- If no vehicle is discharged at time t, we can discharge a vehicle of either class at time $t+1$
  - Case 1: empty intersection
  - Case 2: switching over

Transition probabilities

- $p(x', y'|x, y, a)$ = probability that $X[t+1] = x', Y[t+1] = y'$ conditional on $X[t] = x, Y[t] = y, A[t] = a$.
- Notational convention:
  - Capital letter = random variables: X, Y, A
  - Lower-case letter = numbers x, y, a
  - CDF $F_X(x)$, PMF $p_X(x)$, PDF $f_X(x)$
  - Avoid writing "$\Pr\{x = 1\}$" or "$f(X)$ is increasing in X"
- Suppose that $X(t) = [2,3]^T$, $Y(t) = 1$, and $A(t) = 1$.
  - Pr{$X(t+1) = *, Y(t+1) = *$}=?
  - Pr{[1,3],1}=(1-p1)(1-p2), Pr{[2,3],1}=p1(1-p2), Pr{[1,4],1}=(1-p1)p2, Pr{[2,4],1}=p1p2

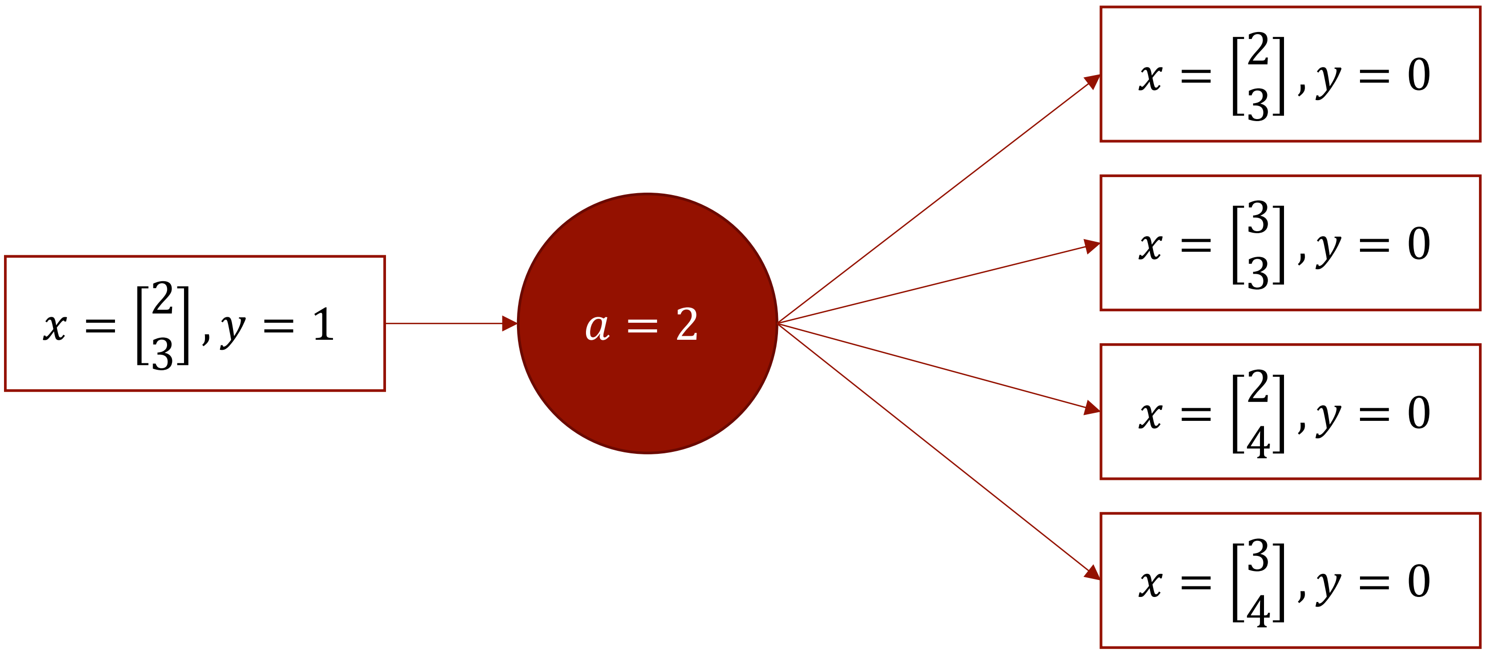
Transition probabilities

Suppose that $X(t) = [2,3]^T$, $Y(t) = 1$, and $A(t) = 1$.



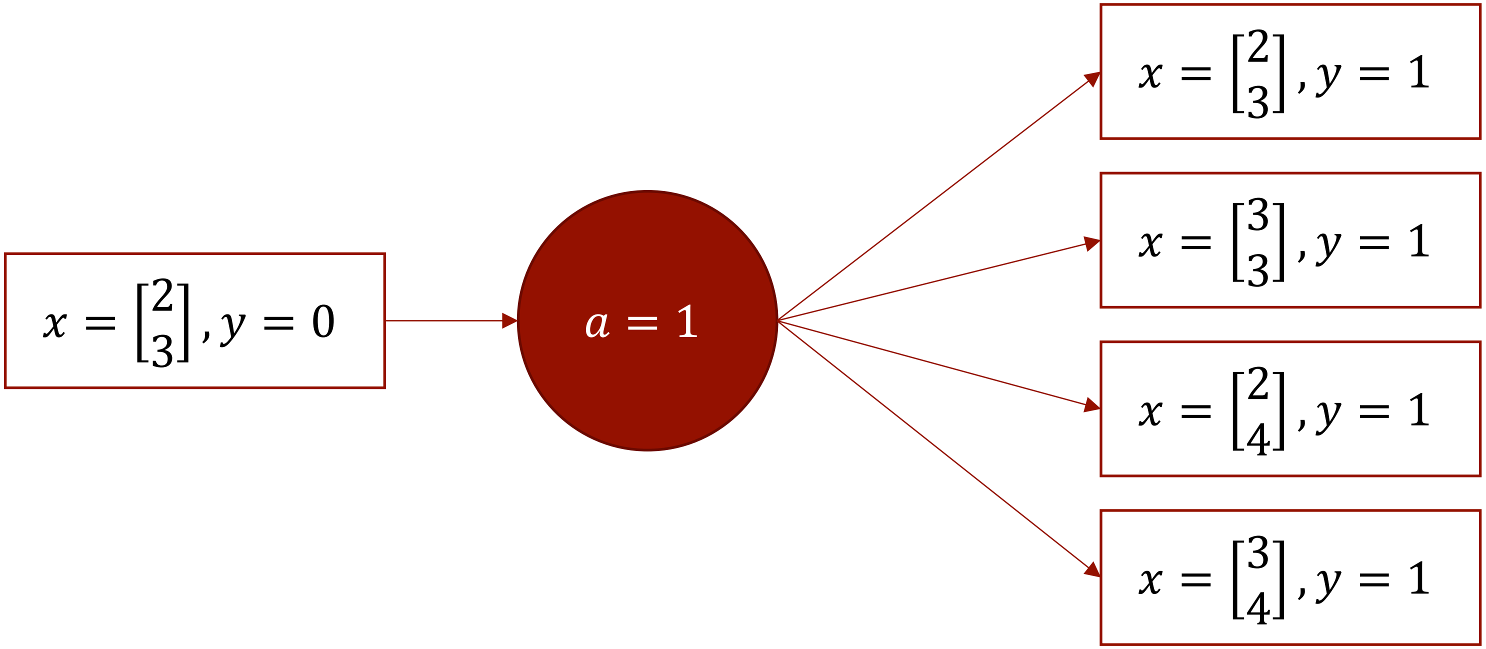
Transition probabilities

Suppose that $X(t) = [2,3]^T$, $Y(t) = 1$, and $A(t) = 2$.



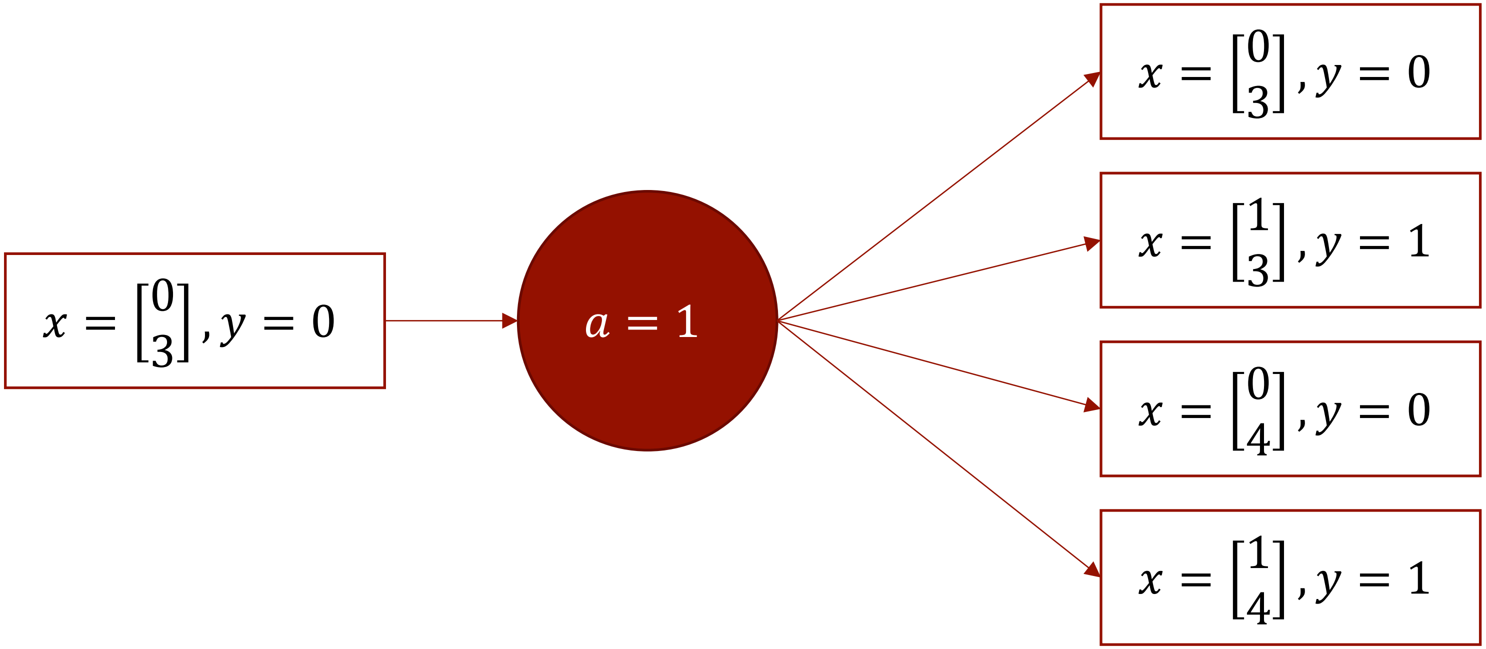
Transition probabilities

Suppose that $X(t) = [2,3]^T$, $Y(t) = 0$, and $A(t) = 1$.



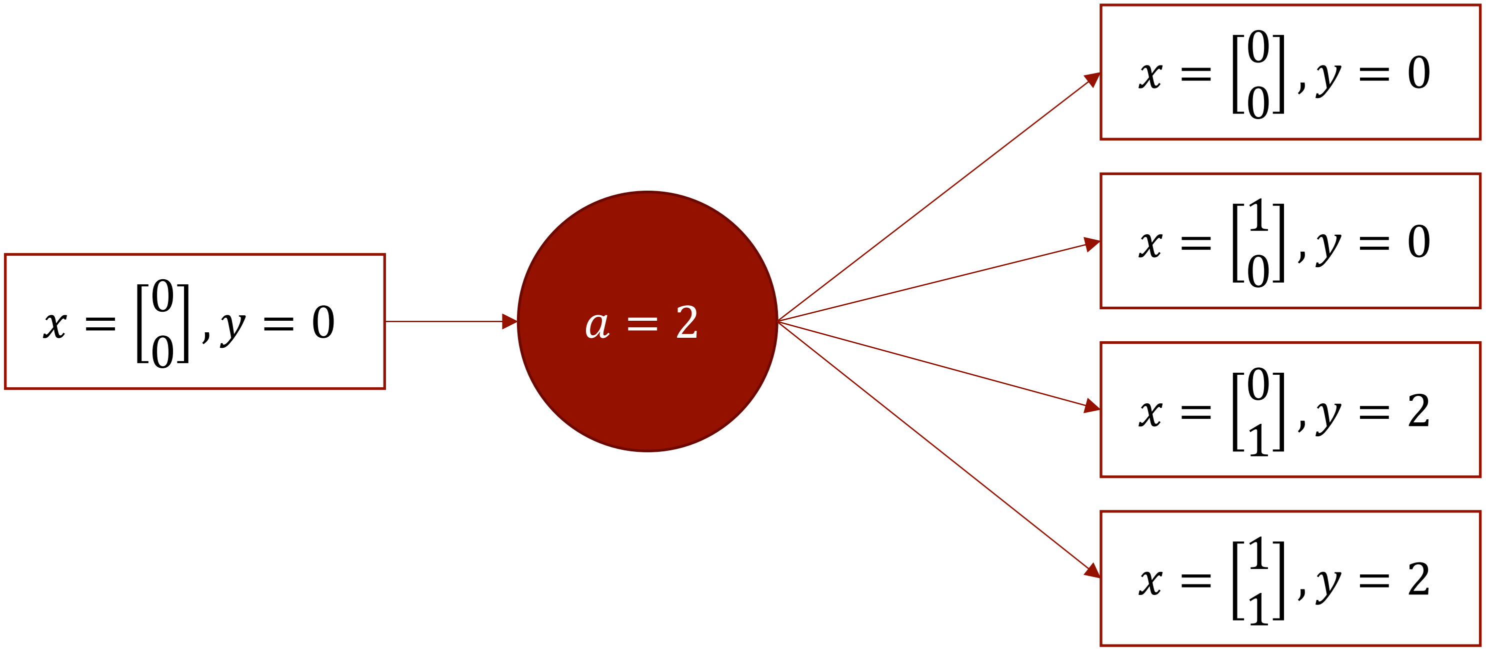
Transition probabilities

Suppose that $X(t) = [0,3]^T$, $Y(t) = 0$, and $A(t) = 1$.



Transition probabilities

Suppose that $X(t) = [0,0]^T$, $Y(t) = 0$, and $A(t) = 2$.



A quick note on dynamic programming

The process $\{X(t), Y(t); t = 0,1,2, \dots\}$ is a discrete-time, discrete-state Markov process.

- Markov processes
- Markov decision processes
- Agent-environment interface
- Reference: Sutton, Richard S., and Andrew G. Barto. Reinforcement learning: An introduction. MIT Press, 2018. http://incompleteideas.net/book/RLbook2020.pdf (Optional)

Markov process

- Stochastic process: random variables evolving over time
  - Flip a coin
  - Count vehicles
  - Measure power demand
- Mathematically, we use a time-varying random variable S_t to describe a stochastic process
- $\Pr\{S_t = s | S_{t-1}, \dots S_1, S_0\}$
- Markov process: the distribution of S_t only depends on S_{t-1} and does not depend on $S_0, \dots, S_{t-2}$
- $\Pr\{S_t = s | S_{t-1}, \dots S_1, S_0\} = \Pr\{S_t | S_{t-1}\}$

Андре́й А. Ма́рков
Andrey A. Markov
安德烈·A·马尔可夫
1856-1922

Agent-environment interface

- Markov decision process: at each time, we can take some action that affects the evolution of the stochastic process
- Agent-environment loop in a MDP

- MDP trajectory:
  - Time sequence t = 0, 1, 2, …
  - State: $S_t \in \mathcal{S}$
  - Action: $A_t \in \mathcal{A}(s)$
  - Reward: $R_t \in \mathcal{R}$
  - Trajectory: $S_0, A_0, R_1, S_1, A_1, R_2, S_2, A_2, \ldots$

Dynamics

- We use function p to describe dynamics of MPD:
$$p(s',r|s,a) := \Pr\{S_t = s', R_t = r | S_{t-1} = s, A_{t-1} = a\}$$
- Reward may or may not be random
- For intersection control, we can set
$$r = -|x|.$$
- With a slight abuse of notation, state-transition probabilities
$$p(s'|s,a) := \Pr\{S_t = s' | S_{t-1} = s, A_{t-1} = a\}$$
$$= \sum_{r \in \mathcal{R}} p(s',r|s,a)$$

Reward

- Expected reward

$$r(s,a) \coloneqq \mathrm{E}\{R_t | S_{t-1} = s, A_{t-1} = a\}$$
$$= \sum_{r \in \mathcal{R}} r \sum_{s' \in \mathcal{S}} p(s', r | s, a)$$

- Three-argument function

$$r(s,a,s') \coloneqq \mathrm{E}\{R_t = r | S_{t-1} = s, A_{t-1} = a, S_t = s'\}$$
$$= \sum_{r \in \mathcal{R}} r \frac{p(s', r | s, a)}{p(s' | s, a)}$$

MDP formulation for vehicle sequencing

- State $S(t) = \big(X(t), Y(t)\big) \in \mathbb{Z}_{\geq 0}^2 \times \{0,1,2\}$
- Action $A(t) \in \{1,2\}$
- Dynamics (transition probabilities)
$$p(s'|s,a) = p(x', y'|x, y, a)$$
- Reward $R(t) = -\|X(t)\|_1 = -X_1(t) - X_2(t)$
- Return $G(t) = \sum_{s=t}^{\infty} \gamma^{s-t} R(s)$ (why discounted?)
- The control problem is to find a policy
$$\mu: \mathbb{Z}_{\geq 0}^2 \times \{0,1,2\} \to \{1,2\}$$
$$\mu: (x, y) \mapsto a$$
that maximizes the expected return.

Sequencing policy

- First in first out (FIFO)
  - Fair, easy
- Minimal switch-over (MSO)
  - Efficient, but maybe unfair
- Longer queue first (LQF)
  - Fairer
- Two metrics for evaluation
  - Throughput: maximal demand that the intersection can accommodate
  - Waiting time: Queuing delay experienced by vehicles

Summary

- Background
  - Signalized & unsignalized intersections
  - Connected & autonomous vehicles
  - Vehicle-to-infrastructure connectivity
- Trajectory planning
  - For a single vehicle
  - For multiple vehicles
- Vehicle sequencing
  - Modeling & formulation
  - Optimization