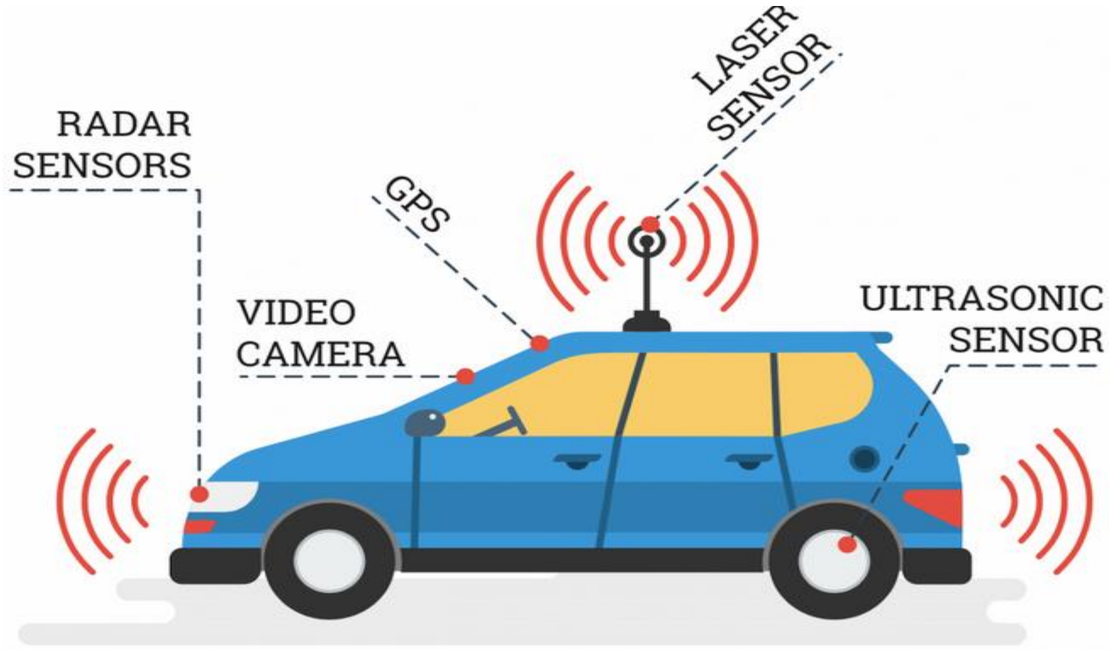
MASS 2021

A Control Policy based Driving Safety System for Autonomous Vehicles

Liuwang Kang and **Haiying Shen**
Department of Computer Science, University of Virginia

Background

- An autonomous vehicle (AV) measures its driving environment through sensors and cameras
- An AV makes real-time control decisions based on sensor measurements to ensure its driving safety



Sensors and cameras

Real-time control decision making

Background

- Control policies specify trigger conditions and control behaviors an AV should always execute when a trigger condition is satisfied
- Vehicle companies keep developing new control strategies for AVs to cover as many as driving situations

Not open for the public

Not the same for different AV types

Deriving control policies of a given AV and selecting optimal control policies help to improve driving safety

Related Work

- Some methods [CDC'18, HPCCTC'17, TCDS'19] keep developing new control strategies for a target AV to ensure its driving safety under different driving scenarios

➢The identified driving scenarios cannot cover all driving scenarios in public roads by considering highly complex driving environments

- Some methods [AR'17, TCST'17, arXiv'20] try to select control policies for a vehicle by assuming its nearby vehicle's driving state constant

➢The nearby vehicle usually has time-varying driving states in practice

Challenges

Propose a control policy-based driving safety system (Polsa) to extract control policies of a target AV and determine optimal control policies considering time-varying driving state of the AV's nearby vehicle

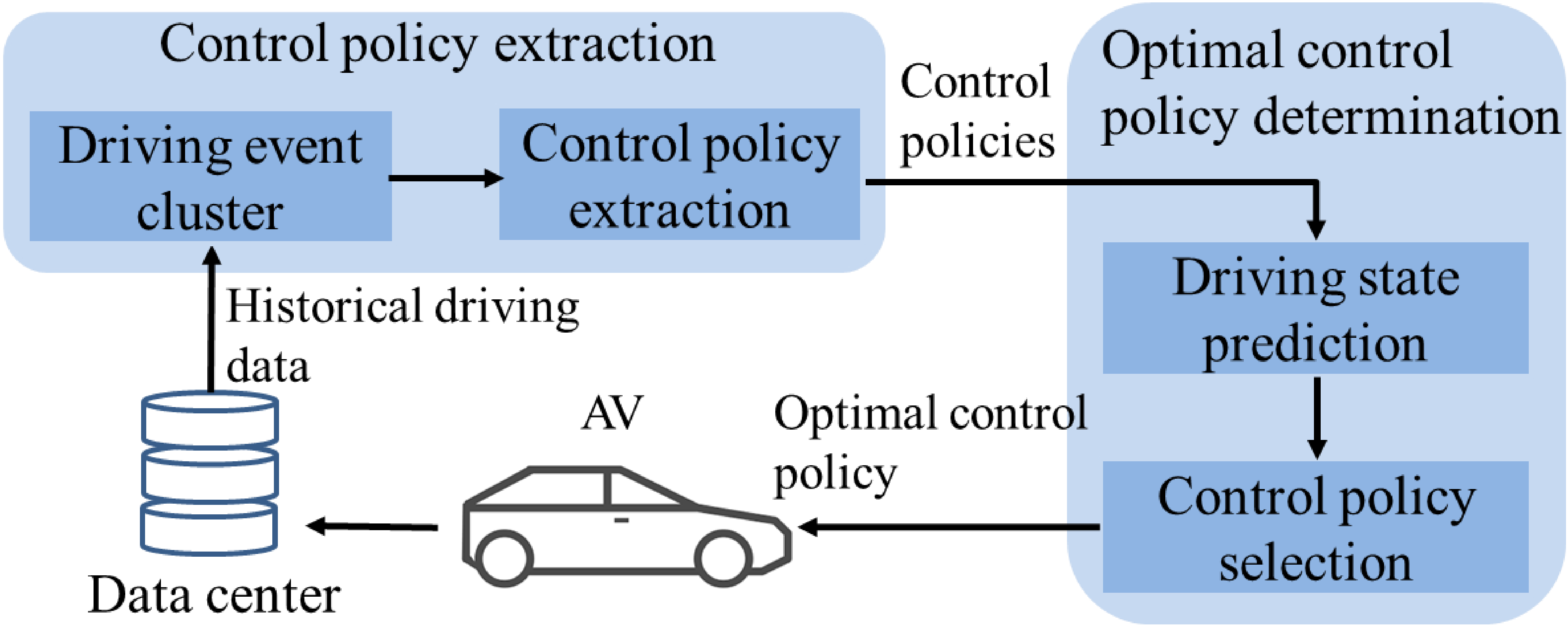
Challenge 1: How to obtain control policies of a given AV based on AV driving data?

- Control policies are not public for driving safety analysis

Challenge 2: How to choose the optimal control policy for a given driving condition?

- The nearby vehicle usually has time-varying driving state

Control Policy-based Driving Safety System



Challenge 1

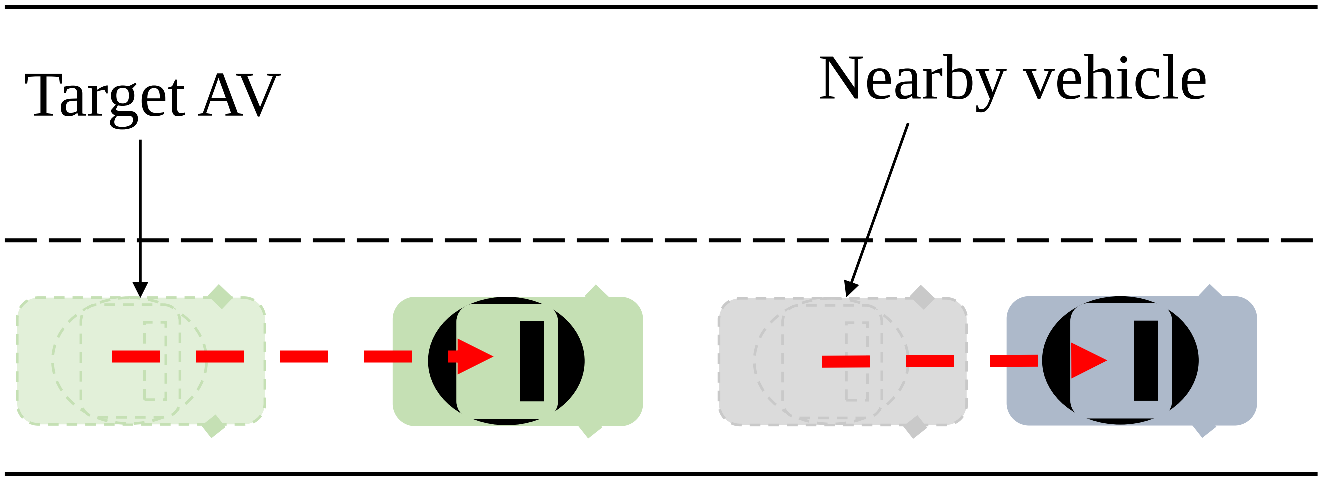
How to obtain control policies of a given AV based on AV driving data?

Control Policy Extraction from Historical Driving Data

Driving event definition

Represents a process where a target AV executes one of its control policies when driving on the road

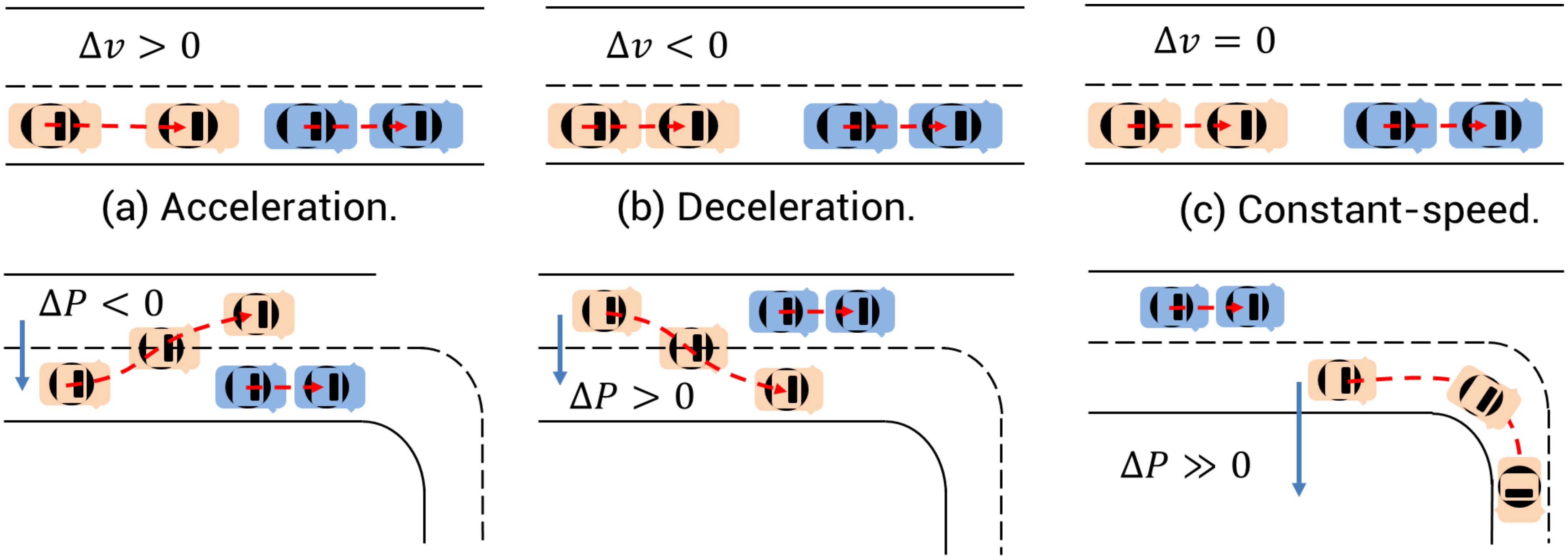
- Includes driving speed and position information of the target AV and its nearby vehicle during its control policy execution process
- Its start-time and end-time indicate specific time when a target AV starts and stops executing a control policy



Control Policy Extraction from Historical Driving Data

Driving event features

Driving events of a target AV are cause by its different control behaviors



(a) Acceleration. (b) Deceleration. (c) Constant-speed.

(d) Left-lane-change. (e) Right-lane-change. (f) Intersection-turn.

Control Policy Extraction from Historical Driving Data

Driving event clustering

- Calculate the driving event dissimilarity between two driving events

Given two driving events Y and $\bar{Y}$ with position trajectory differences ($\Delta \boldsymbol{p}_1$ and $\Delta \boldsymbol{p}_2$) and speed trajectory differences ($\Delta \boldsymbol{v}_1$ and $\Delta \boldsymbol{v}_2$), their driving event dissimilarity degree $Dis(Y, \bar{Y})$ can be calculated using the dynamic time warping method as:

$$Dis(Y, \bar{Y}) = D_p(\Delta \boldsymbol{p}_1, \Delta \boldsymbol{p}_2) + \alpha D_v(\Delta \boldsymbol{v}_1, \Delta \boldsymbol{v}_2)$$

where $D_p(\Delta \boldsymbol{p}_1, \Delta \boldsymbol{p}_2)$ and $D_v(\Delta \boldsymbol{v}_1, \Delta \boldsymbol{v}_2)$ indicate the minimum position alignment distance and the minimum speed alignment distance, respectively.

- Utilize the k-means cluster method to cluster driving events based on driving event dissimilarity calculation results

Control Policy Extraction from Historical Driving Data

Control policy extraction

For each driving event cluster, its corresponding control policy can be extracted as:

- Step 1: Obtain the control behavior type A and driving states $[v, v', p, p']$ of the target AV and its nearby vehicle
- Step 2: Derive the control policy as $\{\mathrm{f}, A\}$ for the driving event cluster

where $f(v, v', p, p')$ is described with a driving state relevant function and used to indicate the trigger condition which triggers control behavior A

Example: a target AV chooses to decelerate when the nearby vehicle decelerates and their relevant distance is less than a distance d

- Control policy: $f(v, v', p, p') = \{\Delta v' \leq 0 \;\&\; \|p' - p\| \leq d\}$ and $A = \{Deceleration\}$

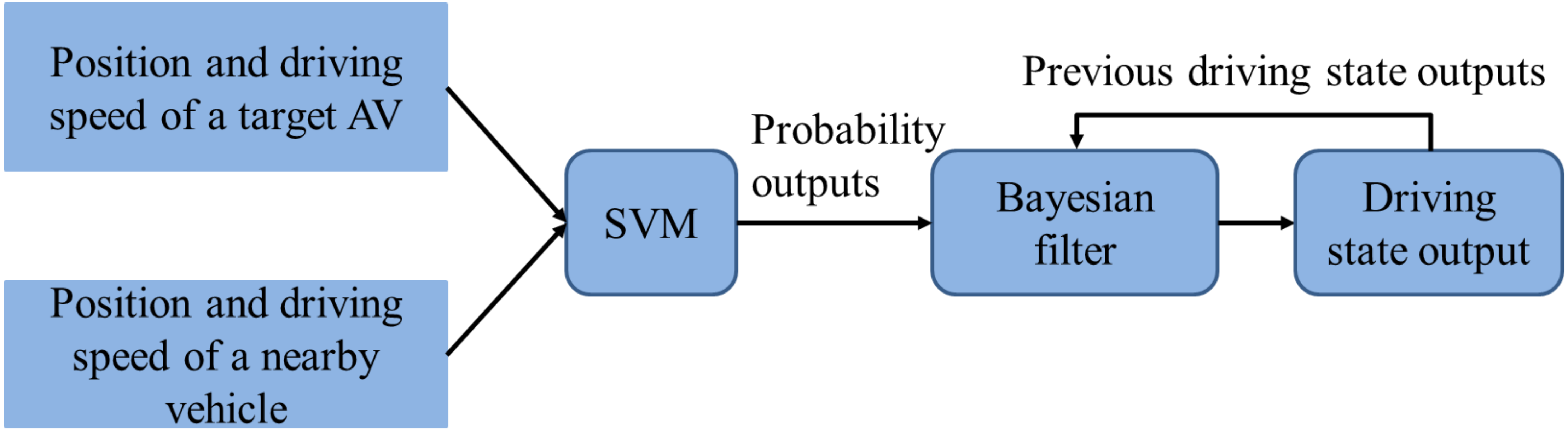
Challenge 2

How to choose the optimal control policy for a given driving condition

Optimal control policy selection

Driving state prediction of the nearby vehicle

Predict driving state of the nearby vehicle with a support vector machine and a Bayesian filter



Optimal control policy selection

Optimal control policy selection

Given driving state $\mathrm{y} = \{p, v\}$ of a target AV and driving state $y' = \{p', v'\}$ of its nearby vehicle, we form an optimal control policy problem to select an optimal control policy, which maximize total rewards in a time period Δ :

$$\arg\max_{\pi \in \Pi} \sum_{t=i}^{i+\Delta} \gamma^{t-i}\, r(y_t, y_t') P(y_t|\pi_i) P(y_t')$$

$\mathrm{r}(y_t, y_t')$ - the reward at time t and equals to $e^{\mu_1 \|p_t - p_t'\| - \mu_2 |v_t - v_t'|}$;

$P(y_t|\pi)$ - probability of driving state being y_t after selecting control policy π

$P(y_t')$ - probability of driving state being y_t' at time t

Performance Evaluation

Experiment settings

- Obtain driving data from Baidu Apollo simulation platform
- 218 driving situations for control policy extraction and 200 driving situations for optimal control policy determination evaluation

Comparison methods

- Designed-policy based driving system (DPDS) [AR’17] and reinforcement learning based driving system (RLDS) [TCST’17].

Evaluation metrics

- Control policy extraction accuracy
- Optimal control decision success rate

Performance Evaluation

Control policy extraction accuracy comparisons

Control policy extraction results on DPDS

Behavior type	# of control policies	Extraction accuracy
Acceleration	9	81%
Deceleration	9	83%
Constant-speed	21	73%
Left-lane-change	9	80%
Right-lane-change	9	83%
Intersection-turn	9	75%

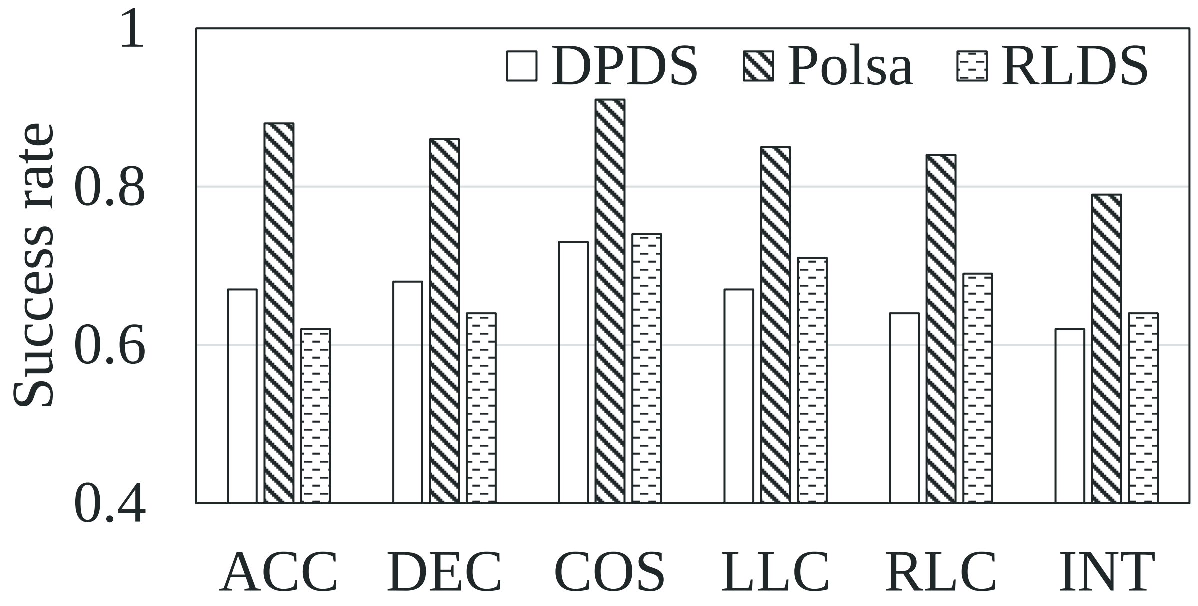
Control policy extraction results on RLDS

Behavior type	# of control policies	Extraction accuracy
Acceleration	6	83%
Deceleration	6	70%
Constant-speed	30	69%
Left-lane-change	6	67%
Right-lane-change	6	72%
Intersection-turn	6	75%

Performance Evaluation

Optimal control policy success rate comparisons

- Polsa has higher optimal control policy success rates on different driving situations
- ReDS has more stable optimal control decision success rates as multiple aggressive driving behaviors exist



Optimal control policy success rate comparisons between Polsa, DPDS and RLDS

Summary

Propose Polsa to derive control policies and select optimal control policies for an AV

- Built a control policy extraction method
- Design an optimal control policy selection method considering time-varying driving state of the nearby vehicle
- Used an industry-standard platform to verify Polsa

Future work

- Consider multiple nearby vehicle situations

Thank you!