Self-driving*: Introduction, Challenges and Open Questions

Aleksandr Petiushko

Nuro

Lomonosov MSU

LOMONOSOV MOSCOW STATE UNIVERSITY

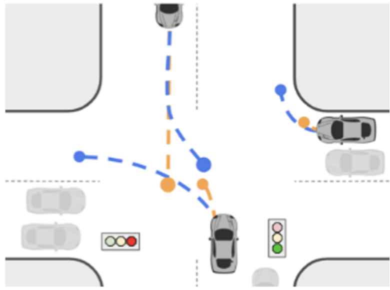
nuro

May 25th, 2023
DLS presentation

*Behavior point of view

Alex’s Intro

- **Motto:** *Standing on the shoulders of giants*
- **Approach**: to combine Academia and Industry Research
    - Academia: Ph.D., lecturer on theory of ML/DL
    - Industry: TLM, Autonomy Interaction Research -> Behavior Research



LOMONOSOV MOSCOW STATE UNIVERSITY

PhD THESIS

MIPT

LOMONOSOV MOSCOW STATE UNIVERSITY

LSI

HUAWEI

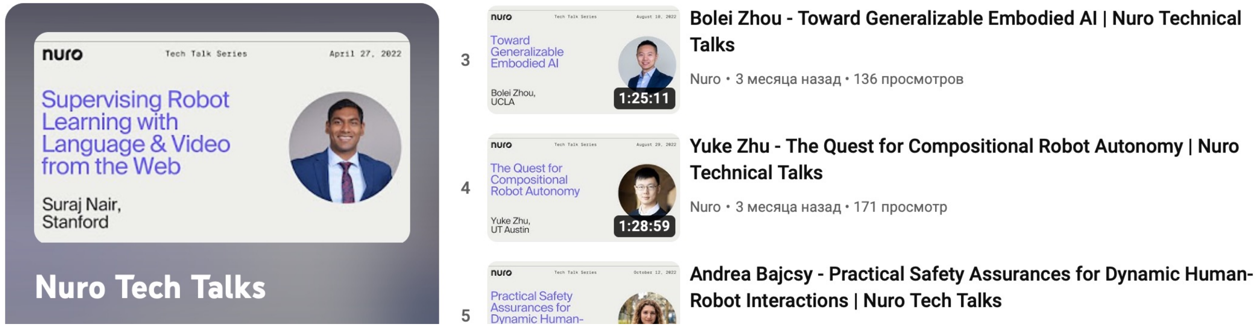
AIRI

nuro

time

Nuro’s intro

- **Motto**: *Better everyday life through robotics*
- **Approach**: to build a self-driving electric last mile delivery bot w/o any driver/passenger
  - Self-driving: ML/DL/AI/Robotics in SW
  - Electric: HW Research
  - Last mile delivery: Restriction of Operation Design Domains
  - Driverless/passenger-free: Slightly different implementation constraints (both SW and HW)



Nuro’s Tech Talks on YouTube: playlist

nuro

Three generations of custom electric vehicles.

1st

AV to receive NHTSA-approved exemption.

Seven leading brands who are trusted partners.

2

States with autonomy operations on public roads—CA & TX.

What is Autonomy Stack itself?

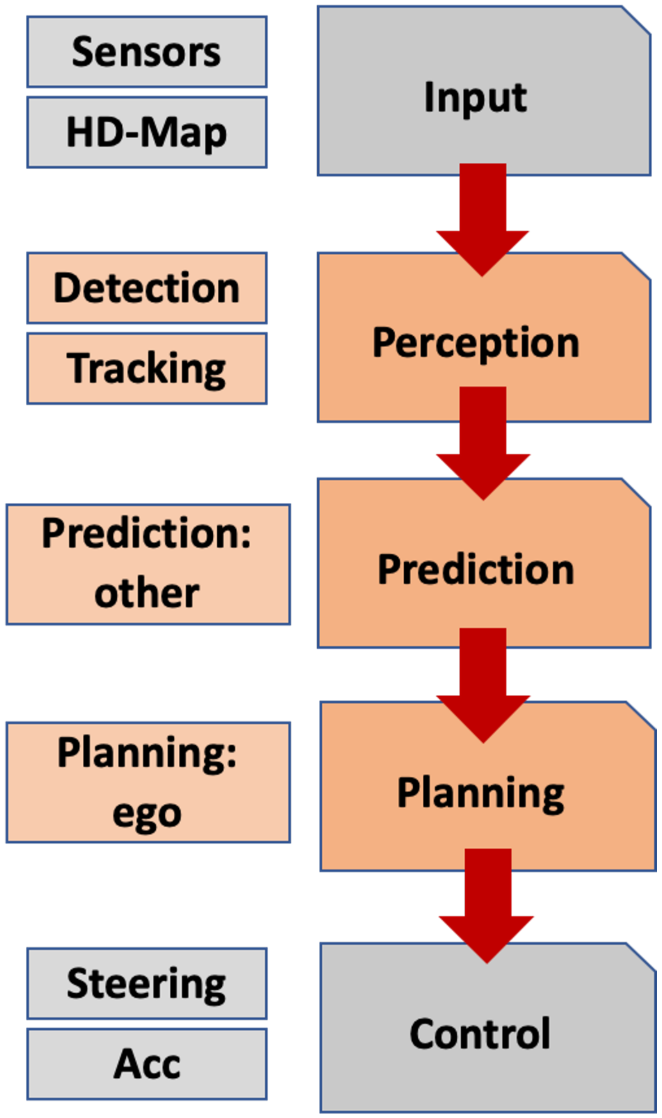
AD and SDV

- **AD** = Autonomous Driving: the *task*
- **SDV** = Self-Driving Vehicle: the *car*
- *AD* is one of the most complex and difficult tasks, both theoretically and practically

<u>Safety</u> of SDV and other agents on the road is crucial

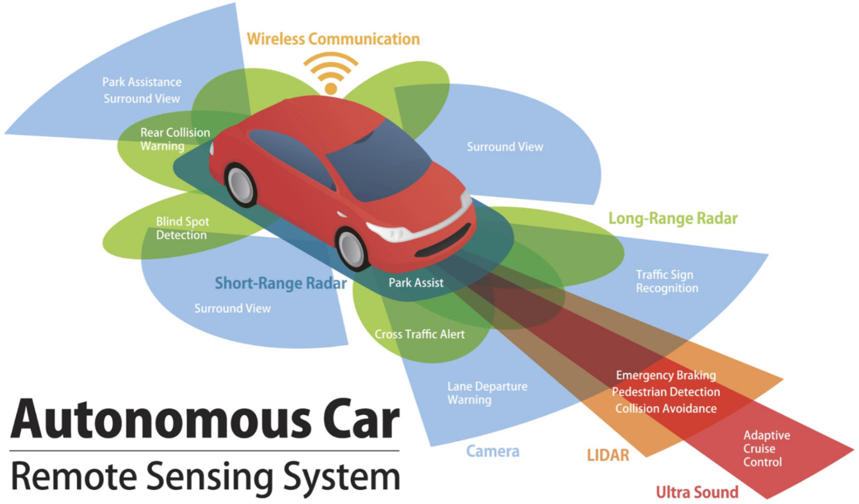
AD: ML Stack of Technologies

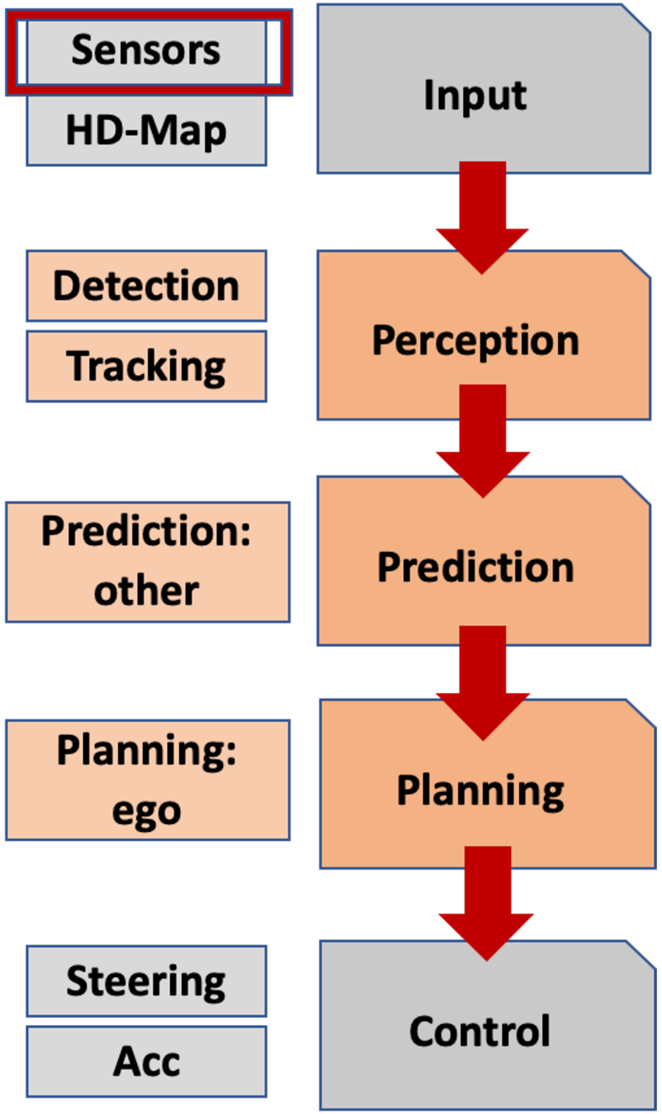
- The main **software** parts are the so-called **P**^{3}:
  - **P**erception, **P**rediction and **P**lanning
- **Hardware** parts:
  - Input: Sensors
  - Output: Control (steering, acceleration)
- High-Definition Map as the helper
  - **HD-Map** contains info about the road



SDV: Sensors

- Various **sensors** are used:
  - LIDAR
  - Radar
  - Ultra Sound
  - Cameras (*x N*)

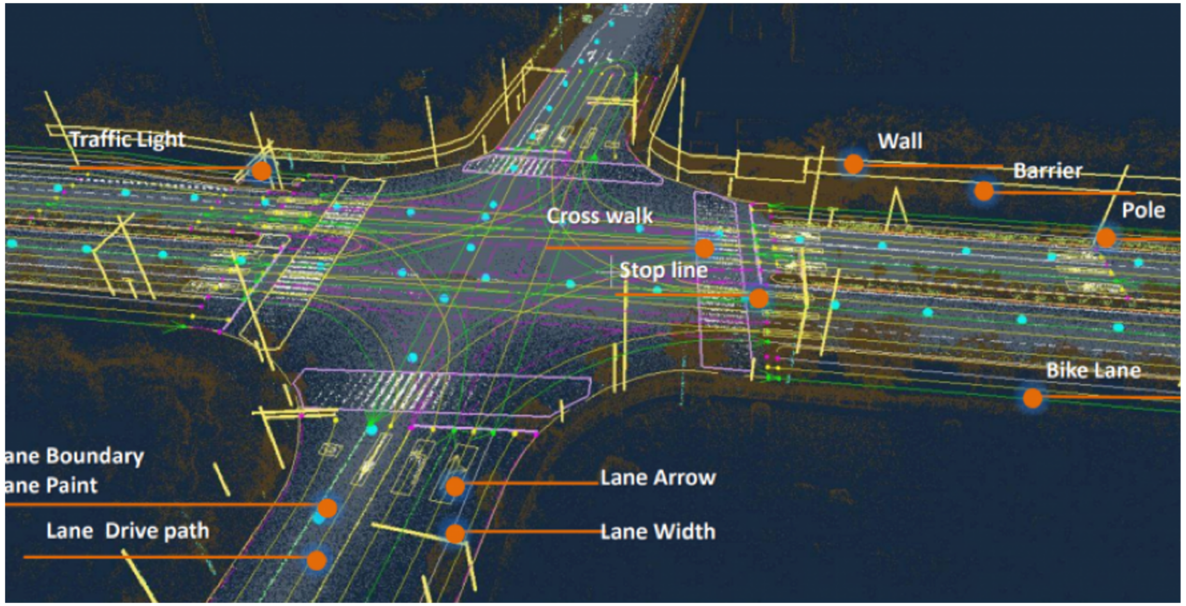


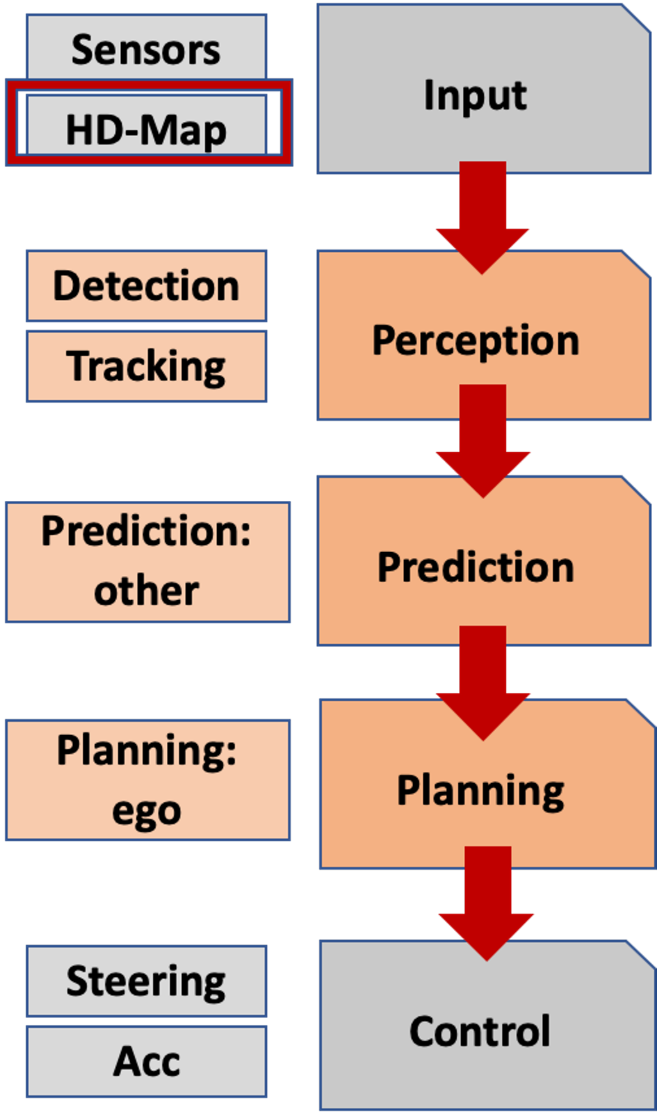


- **Problems**:
  - Expensive
  - Hard to synchronize

AD: HD-Map

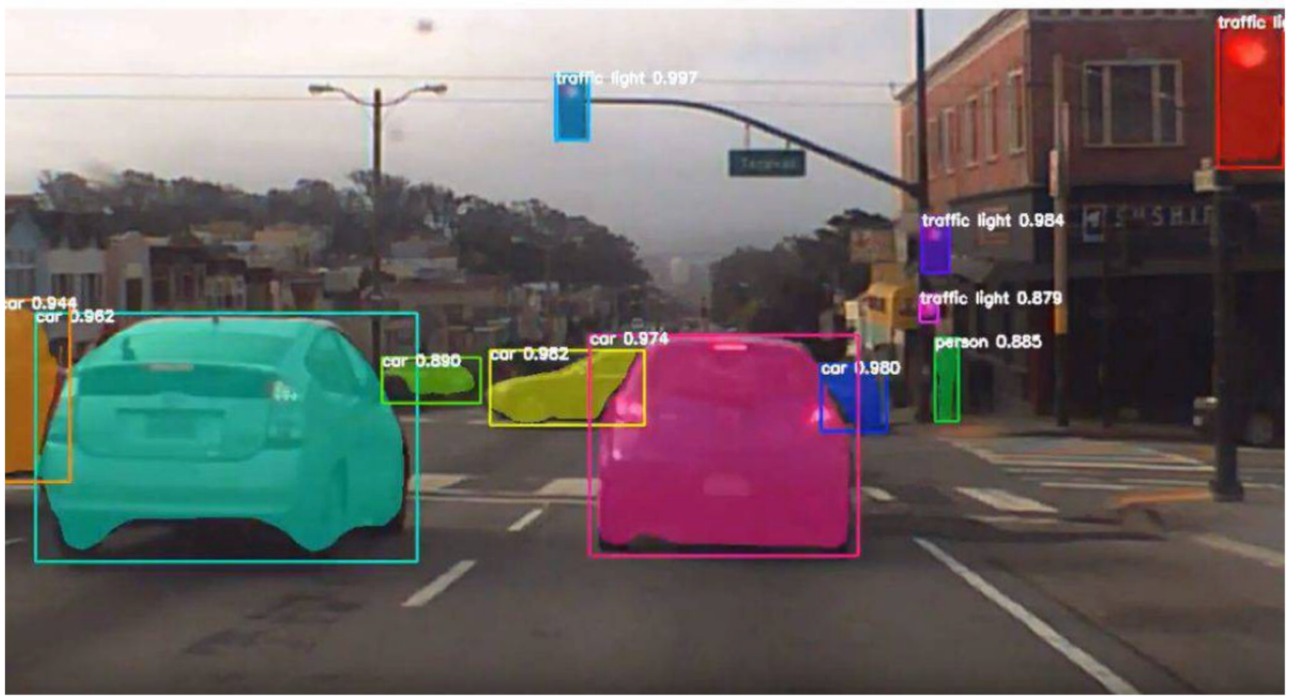
- Helpful for prediction and planning
  - Contains information about a **road**:
    - Lanes, crosswalks, traffic lights, etc.
- **Problems**:
  - Every company has its own format
  - Significant overhead

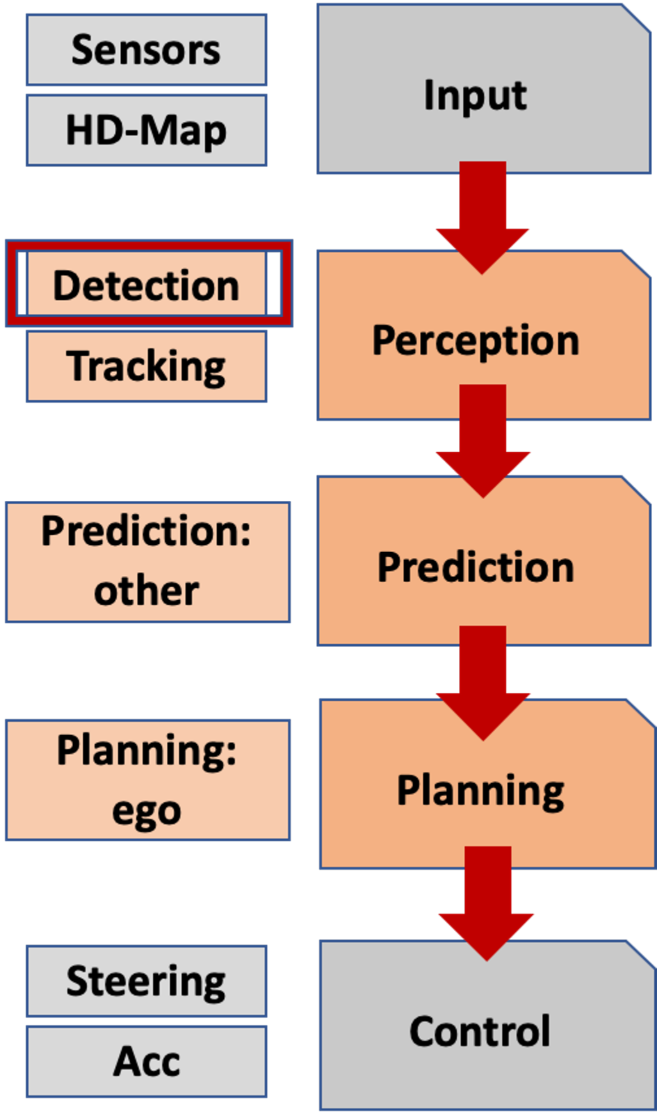




AD: Detection

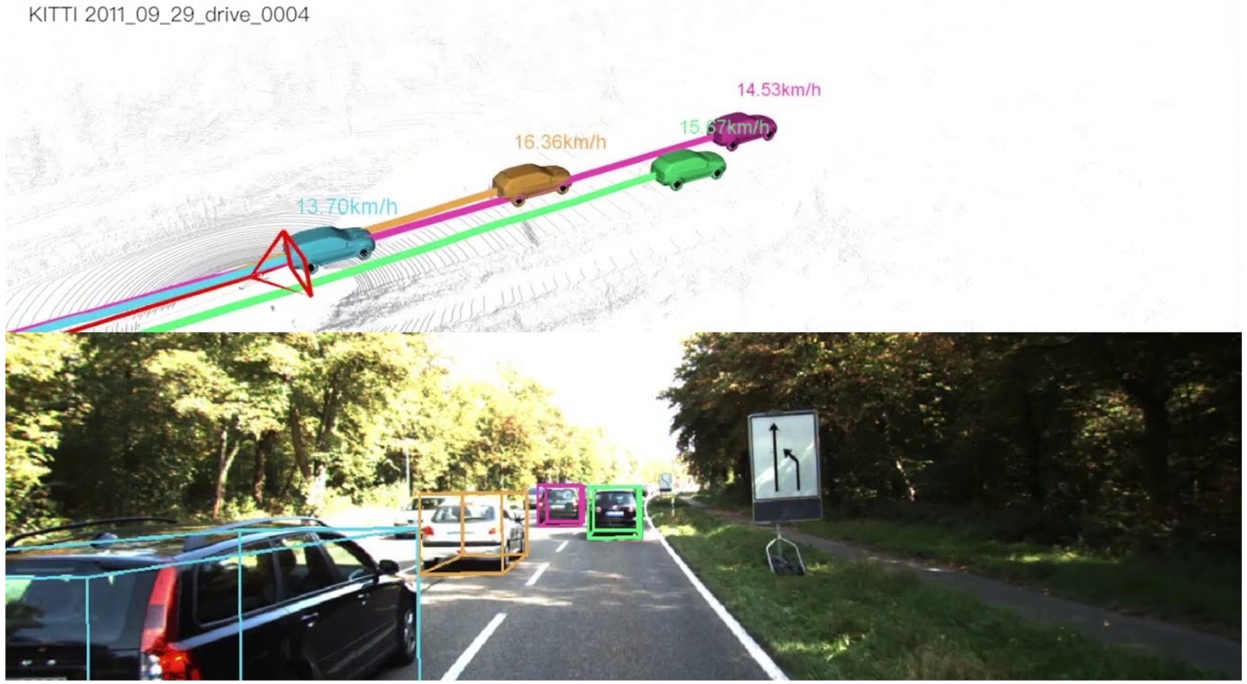
- The *first* step of the Perception part:
  - **Detection** (segmentation, depth-estimation, etc.) of the objects around
- **Problems**:
  - Long tail (small and unusual objects) and anomalies

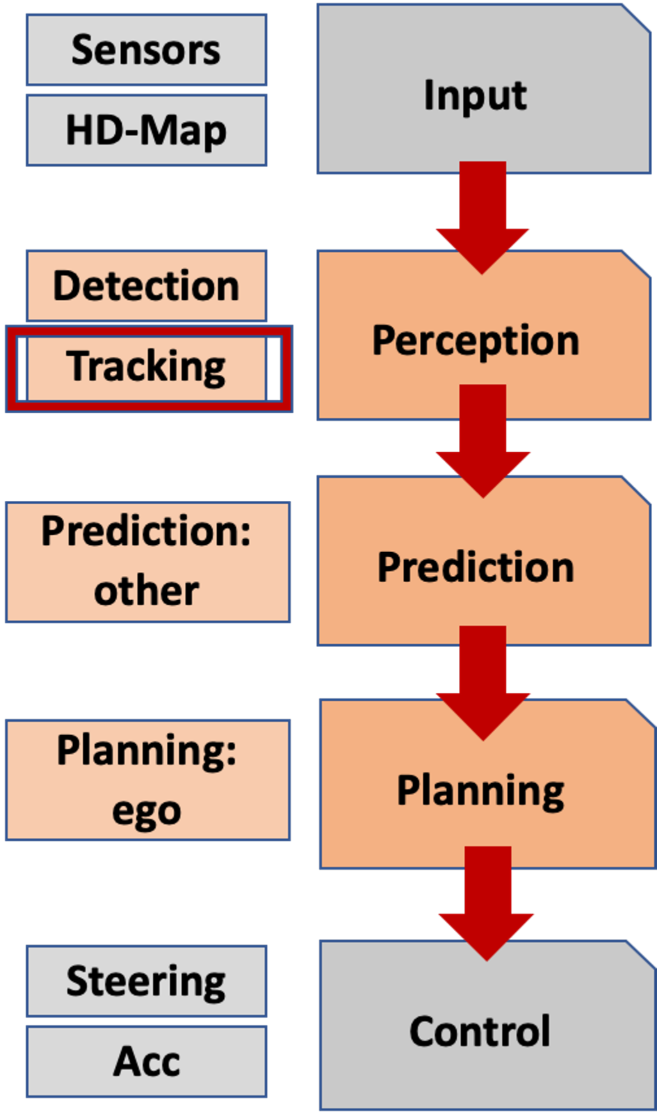




AD: Tracking

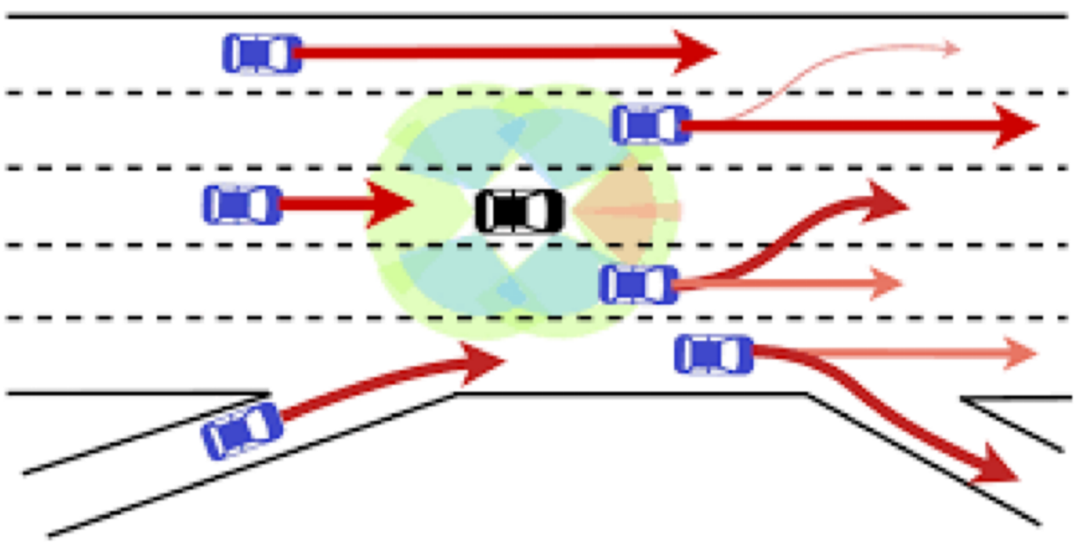
- The *second* step of the Perception part:
  - **Tracking** of the detected objects and estimation of their coordinates for the Prediction part
- **Problems**:
  - Track association of flickering objects

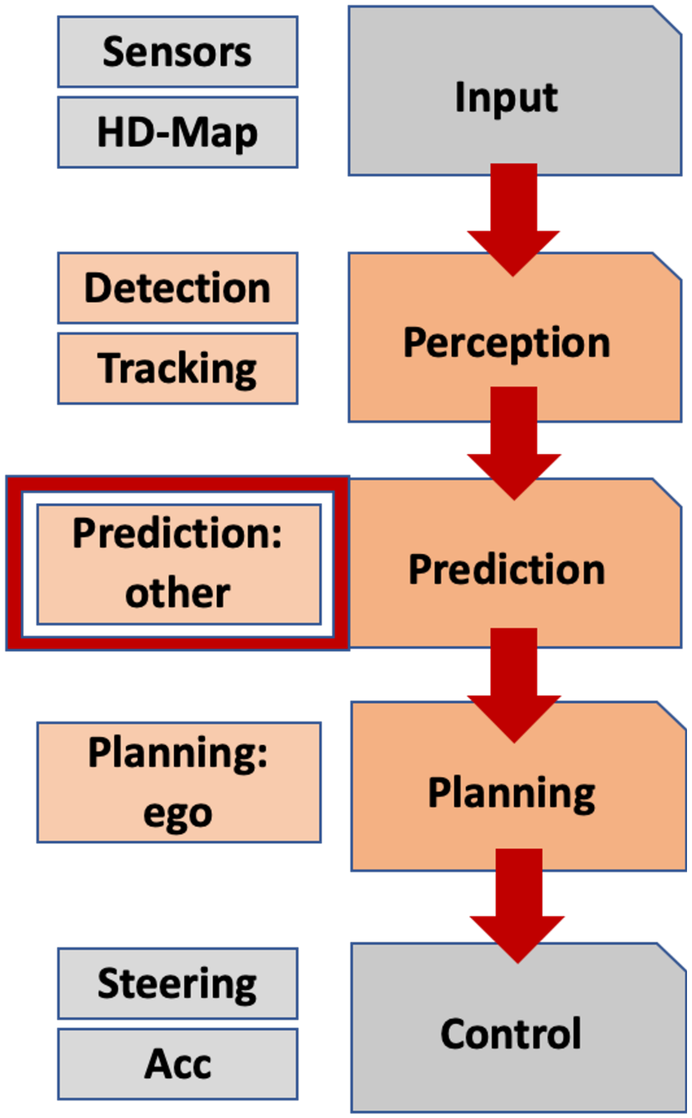




AD: Prediction

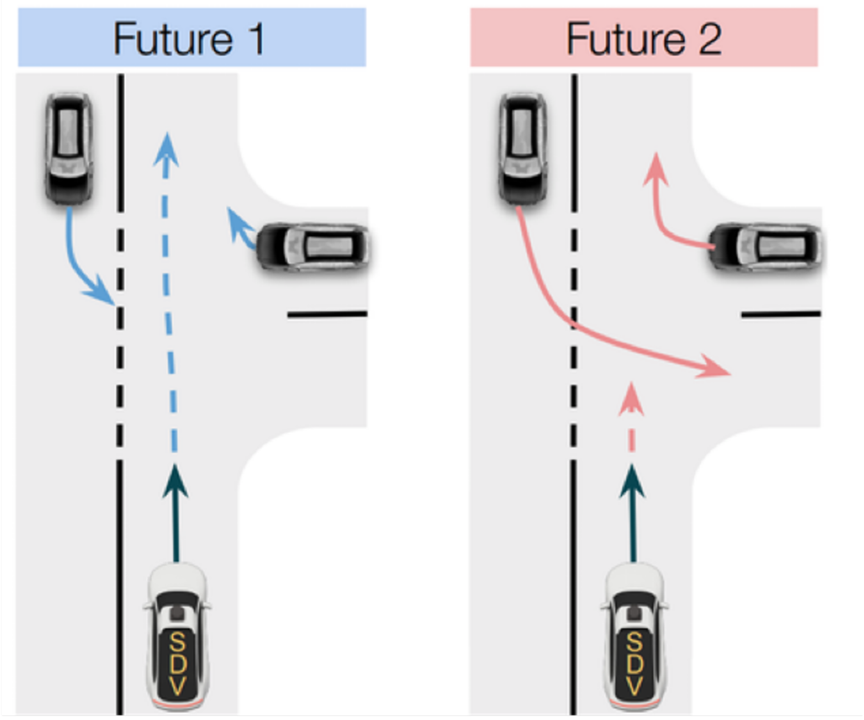
- Future trajectories **prediction** of all surrounding objects based on the *tracking history* and *HD-Map*
  - Usually, 1-10 second
- **Problems**:
  - Multi-modality for recall

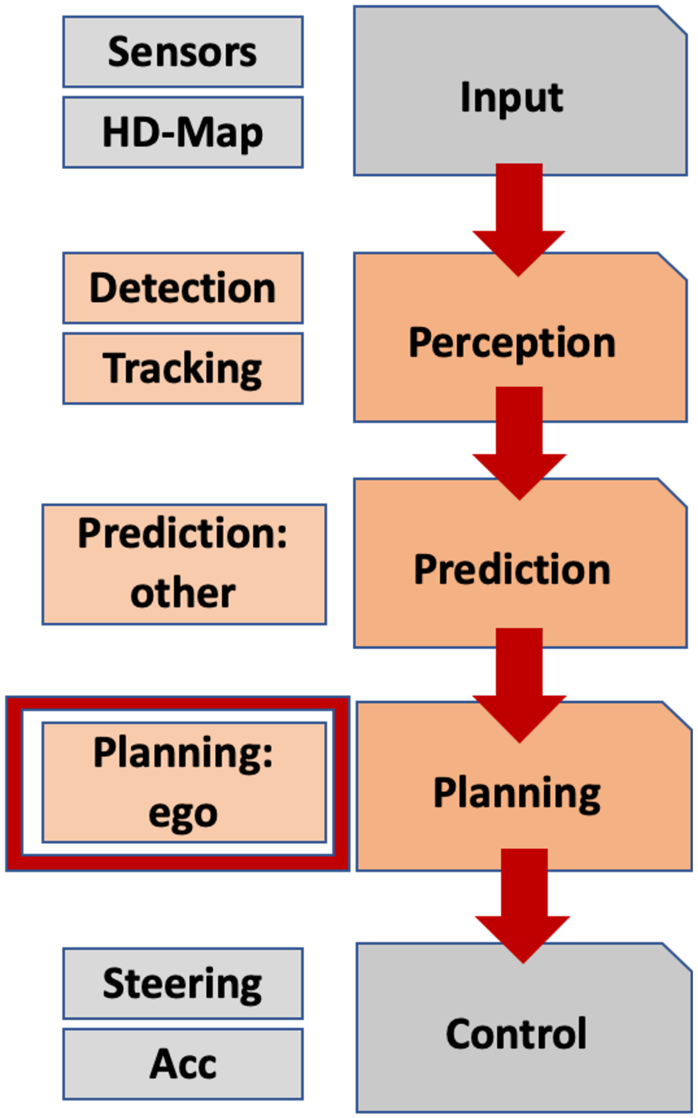




AD: Planning

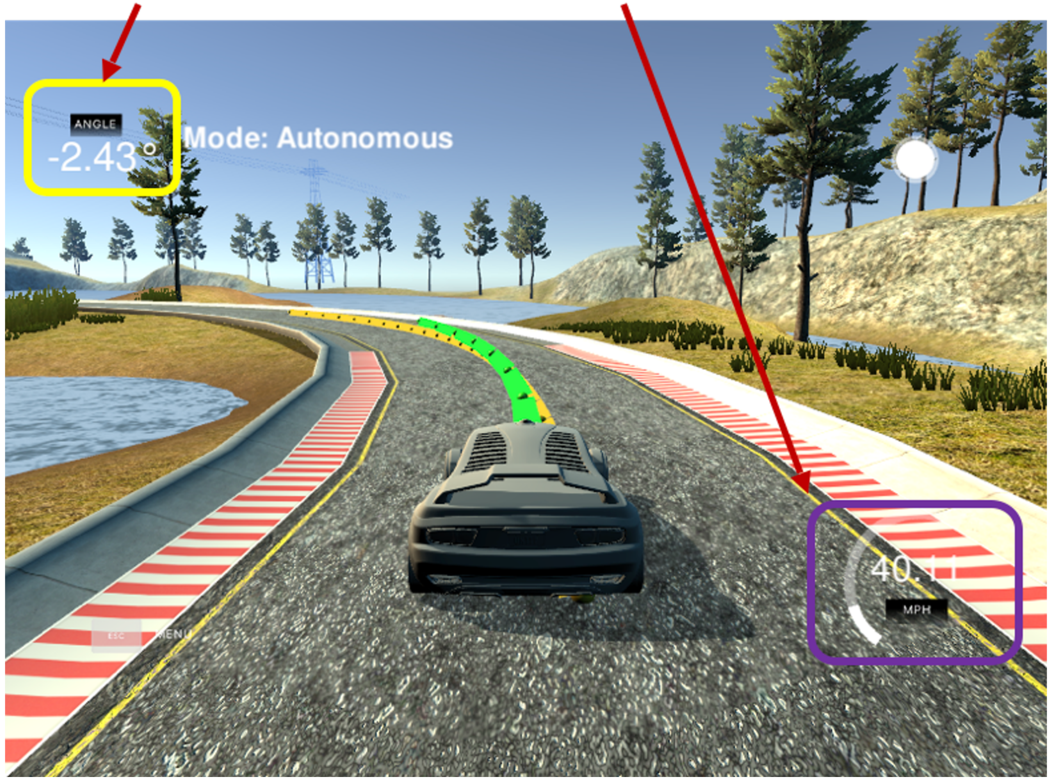
- **Planning** of SDV future actions based on the *predictions* and *HD-Map*
- **Problems**:
  - Consistent joint prediction and planning



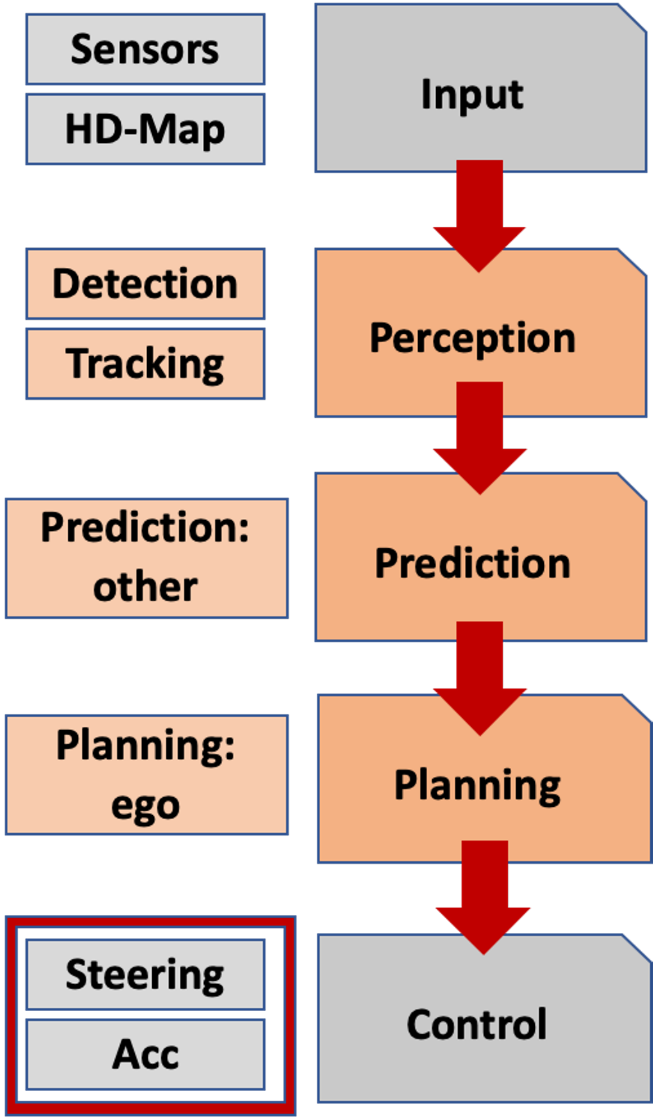


SDV: Control

- Realization and **control** of SDV actions based on *motion plan*
    - Steering control, acceleration control, etc.



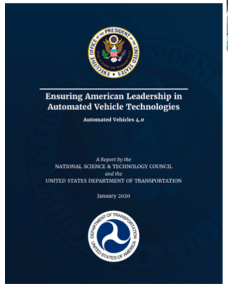
- **Problems**:
    - Dynamic and kinematic limitations



Let’s go deeper and start with regulations

US Department of Transportation

USDOT: Automated Vehicles activities



- Sep 2016 — Federal Automated Vehicles Policy: Accelerating the Next Revolution In Roadway Safety
- Sep 2017 — Automated Driving Systems 2.0: A Vision for Safety
- Oct 2018 — Automated Vehicles 3.0: Preparing for the Future of Transportation
- Jan 2020 — Automated Vehicles 4.0: Ensuring American Leadership in Automated Vehicle Technologies
- Jan 2021 — Automated Vehicles Comprehensive Plan
- 202X — YYY

Five Eras of Safety

According to [National Highway Traffic Safety Administration (NHTSA)]

1950-2000

Safety/Convenience Features

Cruise Control
Seat Belts
Antilock Brakes

2000-2010

Advanced Safety Features

Electronic Stability Control
Blind Spot Detection
Forward Collision Warning
Lane Departure Warning

2010-2016

Advanced Driver Assistance Features

Rearview Video Systems
Automatic Emergency Braking
Pedestrian Automatic Emergency Braking
Rear Automatic Emergency Braking
Rear Cross Traffic Alert
Lane Centering Assist

2016-2025

Partially Automated Safety Features

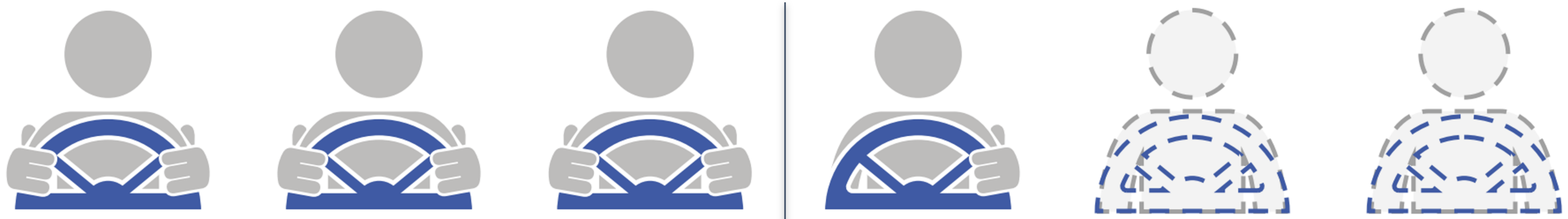
Lane Keeping Assist
Adaptive Cruise Control
Traffic Jam Assist

2025+

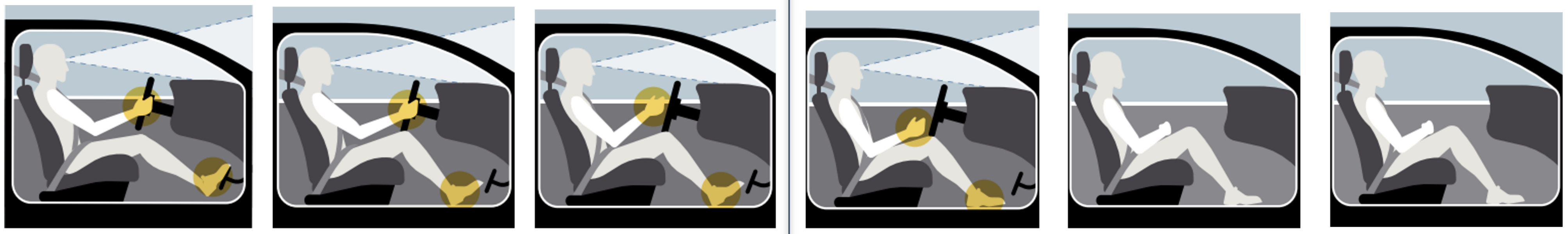
Fully Automated Safety Features

Everything?
* *probably not only above things but even more and/or wider adoption*

Levels of Automation

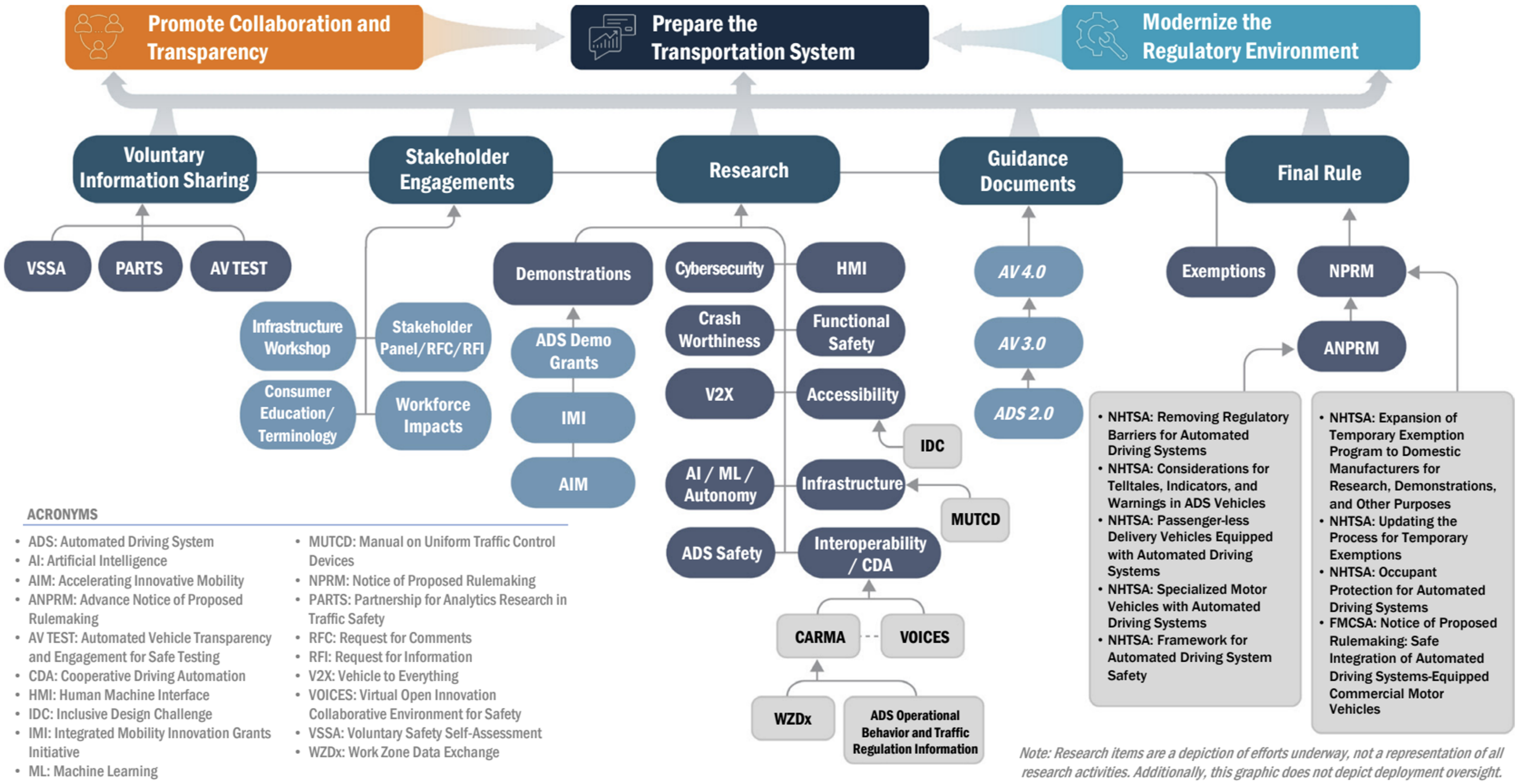


0	1	2	3	4	5
No Automation	**Driver Assistance**	**Partial Automation**	**Conditional Automation**	**High Automation**	**Full Automation**
Zero autonomy; the driver performs all driving tasks.	Vehicle is controlled by the driver, but some driving assist features may be included in the vehicle design.	Vehicle has combined automated functions, like acceleration and steering, but the driver must remain engaged with the driving task and monitor the environment at all times.	Driver is a necessity, but is not required to monitor the environment. The driver must be ready to take control of the vehicle at all times with notice.	The vehicle is capable of performing all driving functions under certain conditions. The driver may have the option to control the vehicle.	The vehicle is capable of performing all driving functions under all conditions. The driver may have the option to control the vehicle.



NHTSA: 1, 2 + SAE (Society of Automotive Engineers) J3016

AV Holistic Plan



USDOT: [Automated Vehicles Comprehensive Plan]

State Regulations

CA DMV Autonomous Vehicle Testing Permit holders

Permit Type

- Testing with a Driver
  - 50 companies
  - ✅ Nuro
- Driverless Testing
  - 7 companies
  - ✅ Nuro
- Deployment
  - 3 companies
  - ✅ Nuro

CA and NV are the only states that allow deployment and require a permit.
* And NV's process is much simpler

California Department of Motor Vehicles (CA DMV)

State Regulations: metrics

Main metrics to report:

- Collisions
- Disengagements
- Mileage (in addition to Disengagement)

California Department of Motor Vehicles (CA DMV)

Article 3.7. Testing of Autonomous Vehicles

(Effective 4/13/2022)

International Standards

- International Electrotechnical Commission
- Functional **Safety** of Electrical/Electronic/Programmable Electronic Safety-related Systems (IEC 61508)

Likelihood of occurrence

Category	Definition	Range (failures per year)
Frequent	Many times in lifetime	$> 10^{-3}$
Probable	Several times in lifetime	10^{-3} to 10^{-4}
Occasional	Once in lifetime	10^{-4} to 10^{-5}
Remote	Unlikely in lifetime	10^{-5} to 10^{-6}
Improbable	Very unlikely to occur	10^{-6} to 10^{-7}
Incredible	Cannot believe that it could occur	$< 10^{-7}$

Consequences

Category	Definition
Catastrophic	Multiple loss of life
Critical	Loss of a single life
Marginal	Major injuries to one or more persons
Negligible	Minor injuries at worst

Risk Analysis

Risk class matrix

	Consequence			
Likelihood	Catastrophic	Critical	Marginal	Negligible
Frequent	I	I	I	II
Probable	I	I	II	III
Occasional	I	II	III	III
Remote	II	III	III	IV
Improbable	III	III	IV	IV
Incredible	IV	IV	IV	IV

- **Class I**: Unacceptable in any circumstance;
- **Class II**: Undesirable: tolerable only if risk reduction is impracticable or if the costs are grossly disproportionate to the improvement gained;
- **Class III**: Tolerable if the cost of risk reduction would exceed the improvement;
- **Class IV**: Acceptable as it stands, though it may need to be monitored.

Wiki on IEC 61508

International Standards

- International Organization for Standardization
- Road vehicles – Functional **safety** (ISO 26262)

ASIL = S x E x C

		C1	C2	C3
S1	E1	QM	QM	QM
S1	E2	QM	QM	QM
S1	E3	QM	QM	ASIL A
S1	E4	QM	ASIL A	ASIL B
S2	E1	QM	QM	QM
S2	E2	QM	QM	ASIL A
S2	E3	QM	ASIL A	ASIL B
S2	E4	ASIL A	ASIL B	ASIL C
S3	E1	QM	QM	ASIL A
S3	E2	QM	ASIL A	ASIL B
S3	E3	ASIL A	ASIL B	ASIL C
S3	E4	ASIL B	ASIL C	**ASIL D**

Autonomous Driving: **ASIL D** => acceptable probability of system / component failure of one in a hundred million

Wiki on IEC 61508 and ASIL (I)

Severity Classifications (S):
- S0 No Injuries
- S1 Light to moderate injuries
- S2 Severe to life-threatening (survival probable) injuries
- S3 Life-threatening (survival uncertain) to fatal injuries

Exposure Classifications (E):
- E0 Incredibly unlikely
- E1 Very low probability (injury could happen only in rare operating conditions)
- E2 Low probability
- E3 Medium probability
- E4 High probability (injury could happen under most operating conditions)

Controllability Classifications (C):
- C0 Controllable in general
- C1 Simply controllable
- C2 Normally controllable (most drivers could act to prevent injury)
- C3 Difficult to control or uncontrollable

Safety integrity level (**SIL**)

SIL	Low demand mode: average probability of failure on demand	High demand or continuous mode: probability of dangerous failure per hour
1	$\geq 10^{-2}$ to $< 10^{-1}$	$\geq 10^{-6}$ to $< 10^{-5}$
2	$\geq 10^{-3}$ to $< 10^{-2}$	$\geq 10^{-7}$ to $< 10^{-6}$
3	$\geq 10^{-4}$ to $< 10^{-3}$	$\geq 10^{-8}$ to $< 10^{-7}$ (1 dangerous failure in 1140 years)
4	$\geq 10^{-5}$ to $< 10^{-4}$	$\geq 10^{-9}$ to $< 10^{-8}$

Automotive Safety integrity level (**ASIL**) vs SIL

Domain	Domain-Specific Safety Levels					
Automotive (ISO 26262)	QM	ASIL A	ASIL B	ASIL C	ASIL D	-
General (IEC 61508)	-	SIL-1	SIL-2		SIL-3	SIL-4

All these regulations are about physical (onroad) metrics.

How to ensure the safe & fast development cycle?

Simulators

Q: How to **safely test** the autonomous capabilities?

A: Using the **simulator**!

Main challenges:

- Sensors simulation
- **Behavior simulation**

CARLA simulator

+ NVIDIA DRIVE Sim, Deepdrive, LGSVL, SUMMIT, Flow, …
+ Internal and specific to any AV company simulators

Kaur, Prabhjot, et al. "A survey on simulators for testing self-driving cars." 2021

Simulators reliability

<u>Reliability</u> questions:

- How to guarantee the **generalization** of simulation results?
- Can we really rely on any **metrics inside** the simulation?

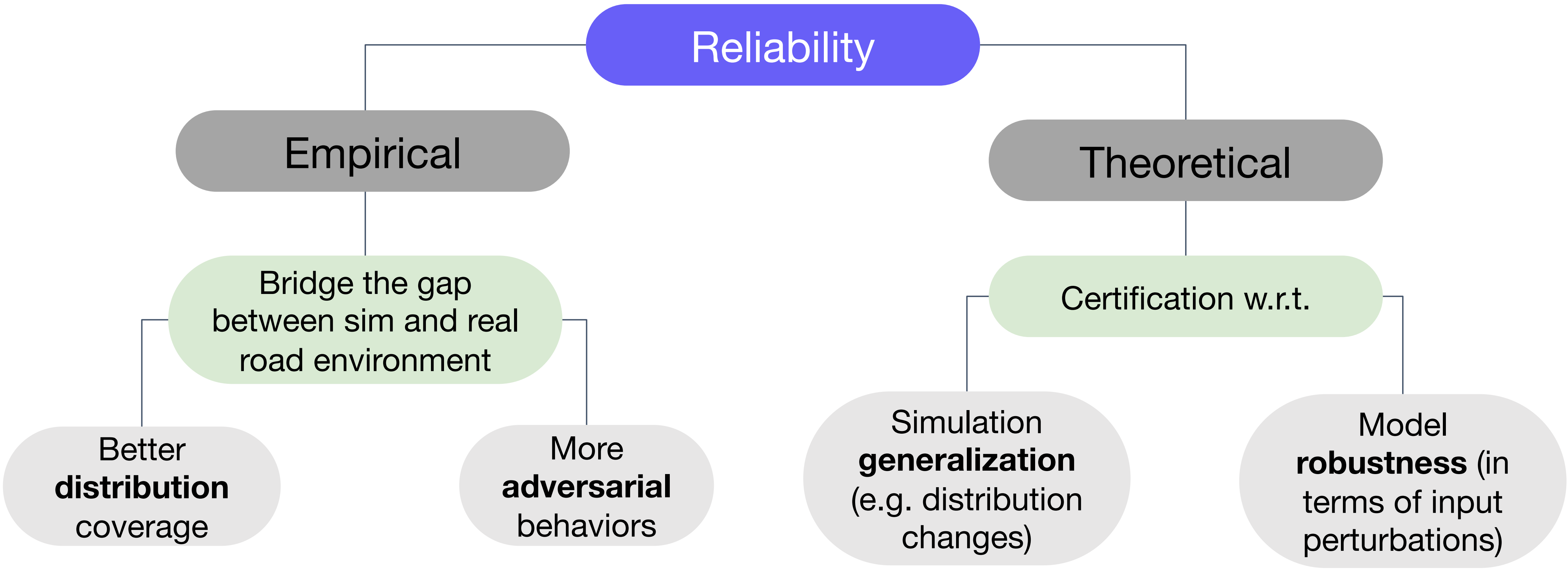
SIMULATION

REALITY

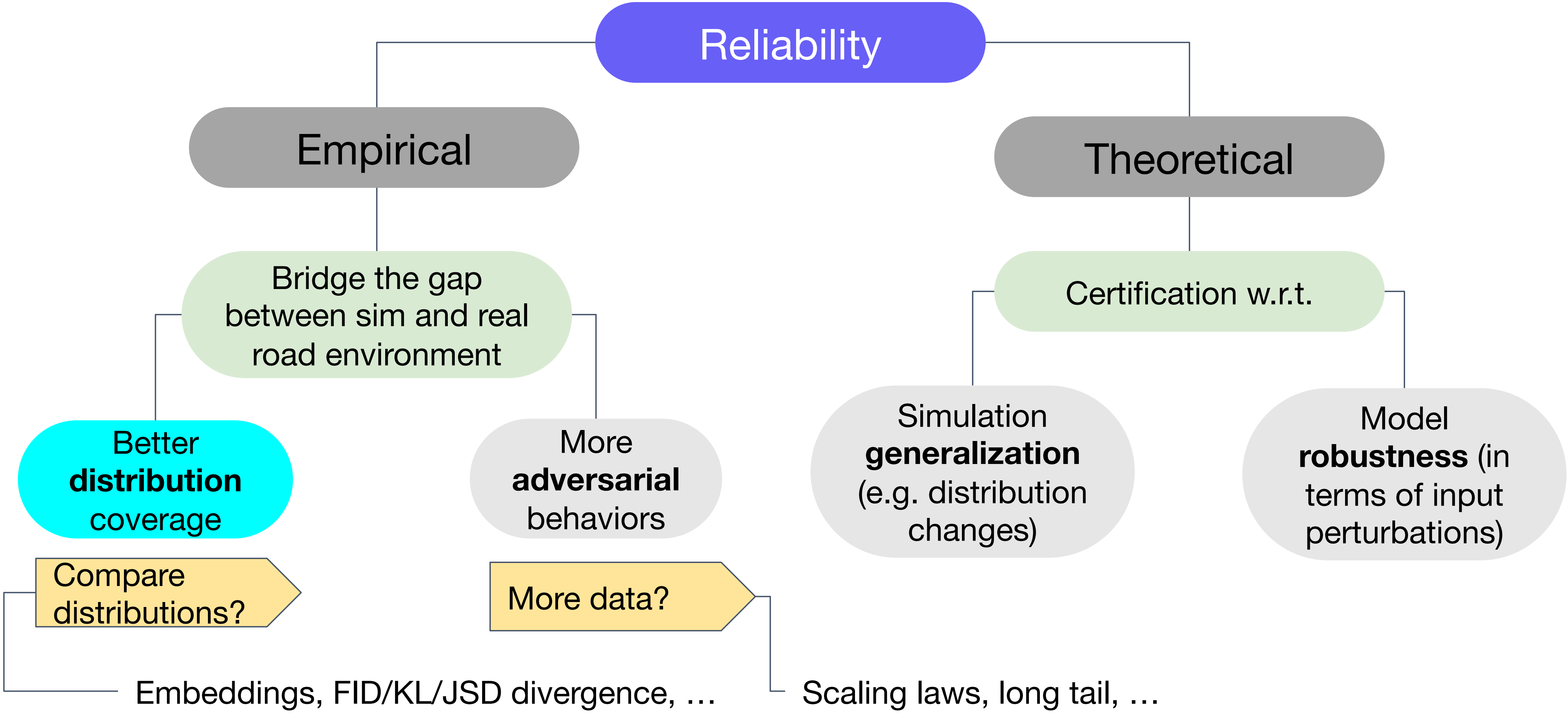
Medium.com: [Simulation vs Reality in Marketing]

Paperswithcode.com: [Domain (distribution) shift]

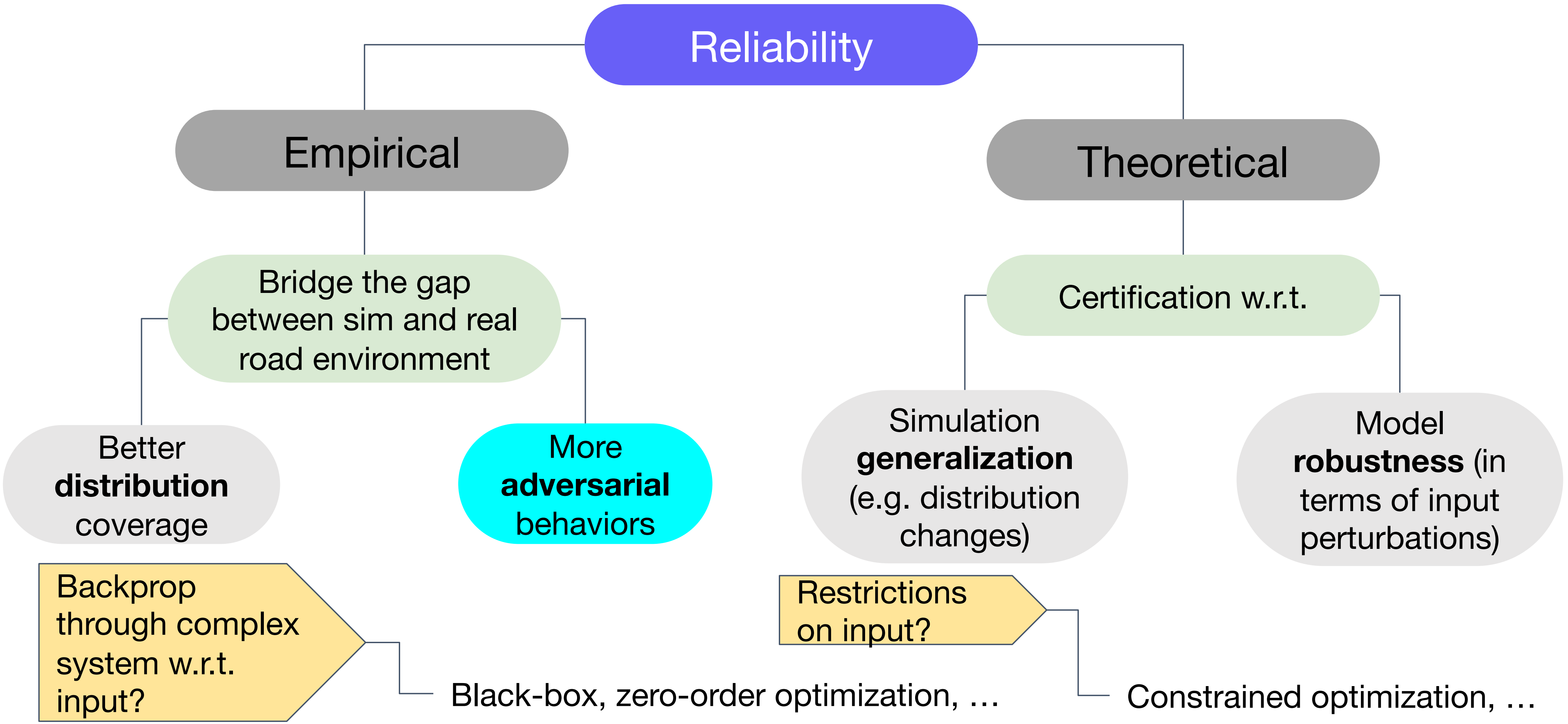
Towards Reliability



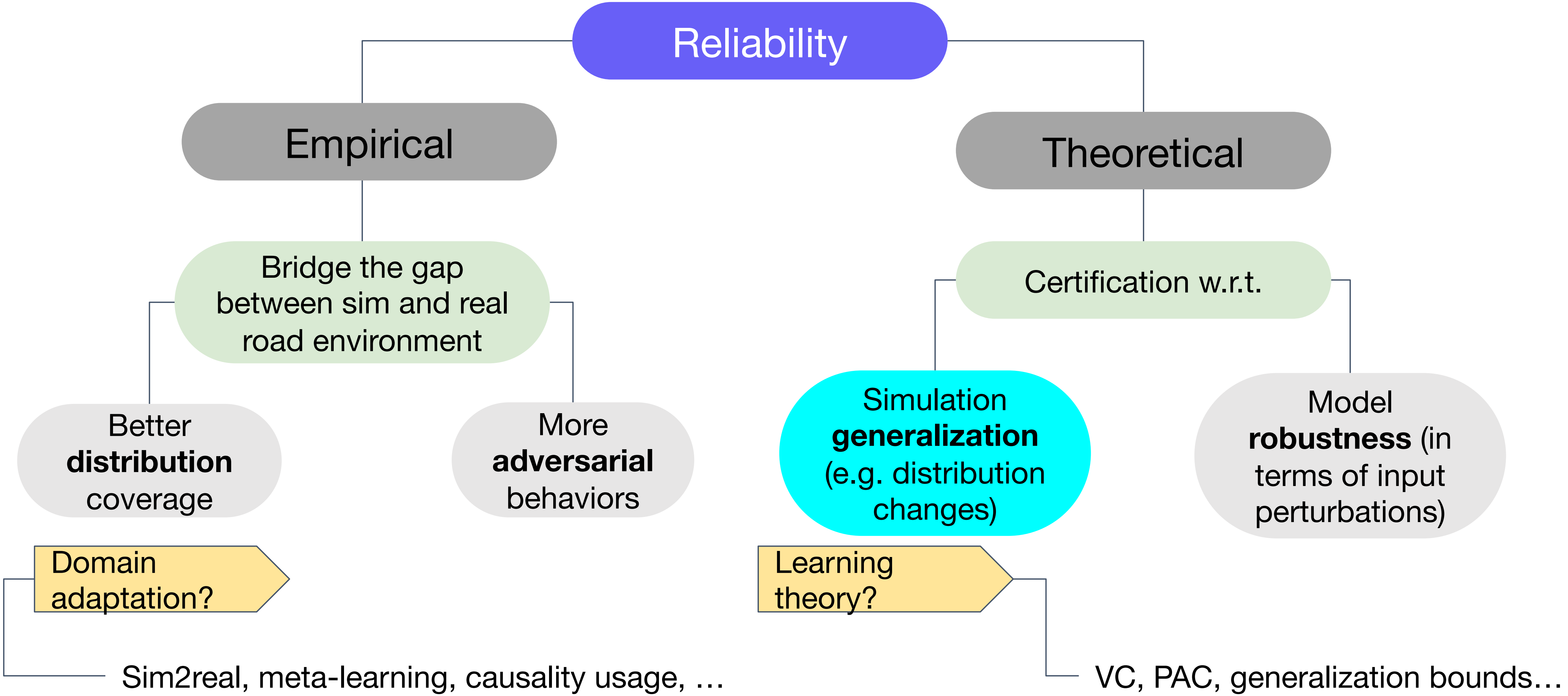
Towards Reliability



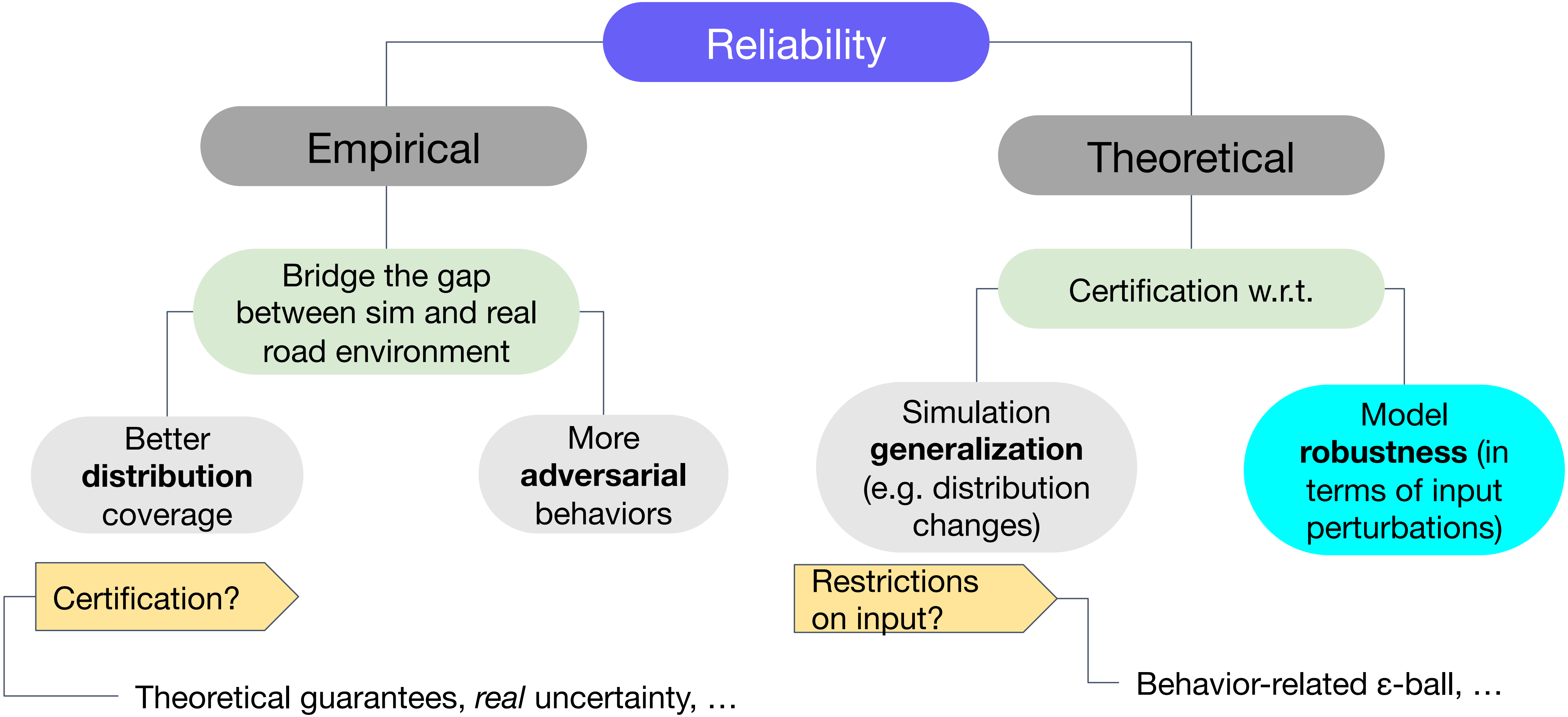
Towards Reliability



Towards Reliability



Towards Reliability

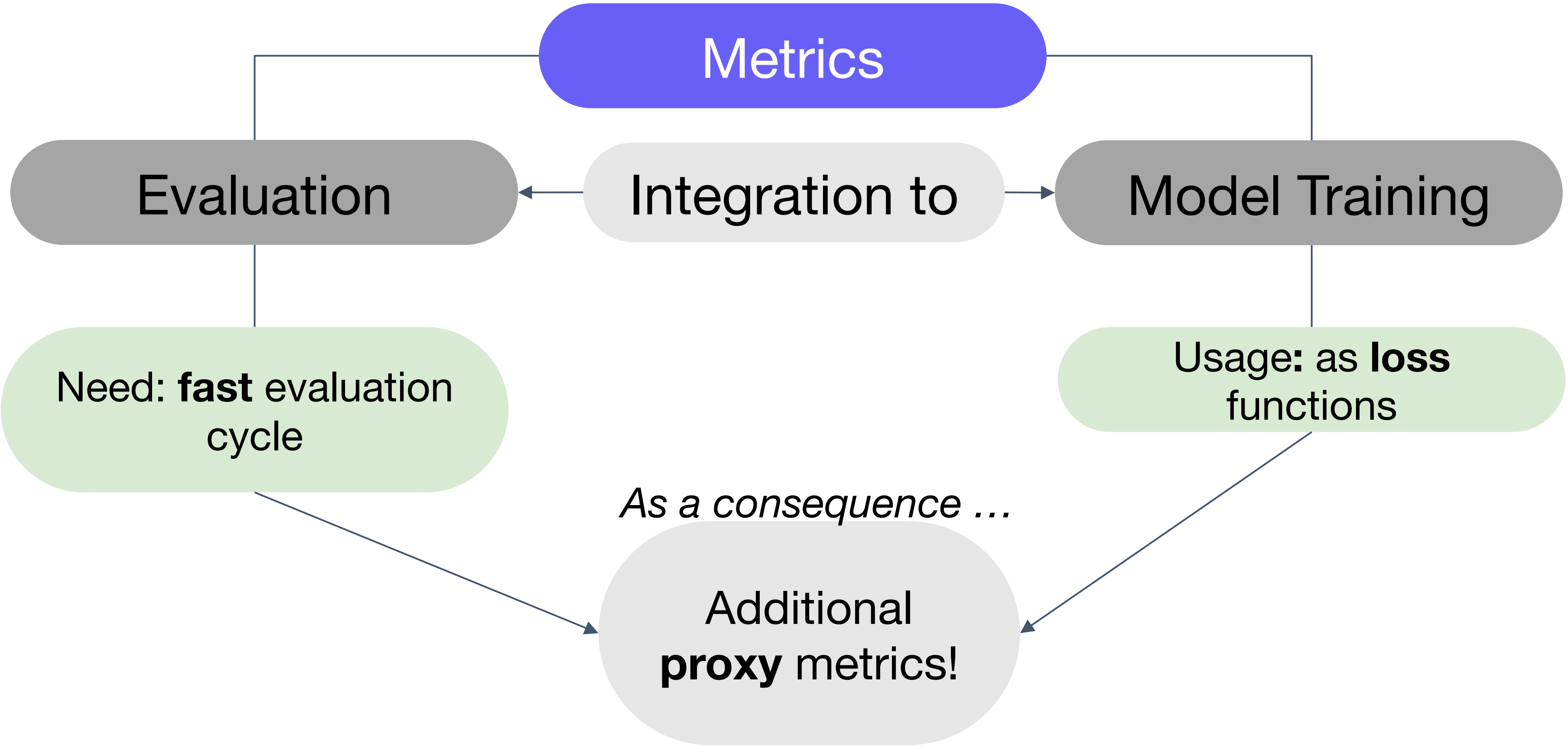


How to ensure the safe & fast development cycle?

Metrics

<u>Common</u> metrics of AV:

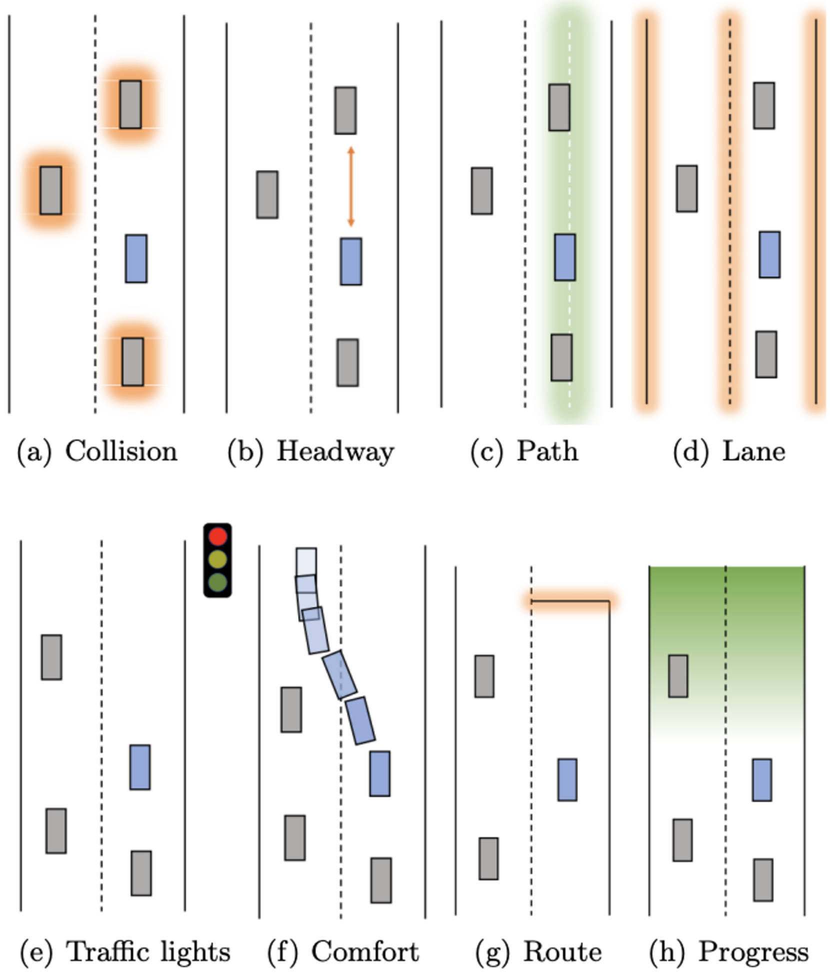
- Miles per (critical) disengagement (**MPD**, **MPCD**)
- **Inverse**: number of disengagements per thousand of miles



Metrics in the literature

<u>Proxy</u> metrics:

- Time to Collision
- Collision rate
- Off-road rate
- Off-route rate
- L2-based
- Comfort-based
    - Jerk
    - Lateral acceleration
- …



<u>Metrics</u>:

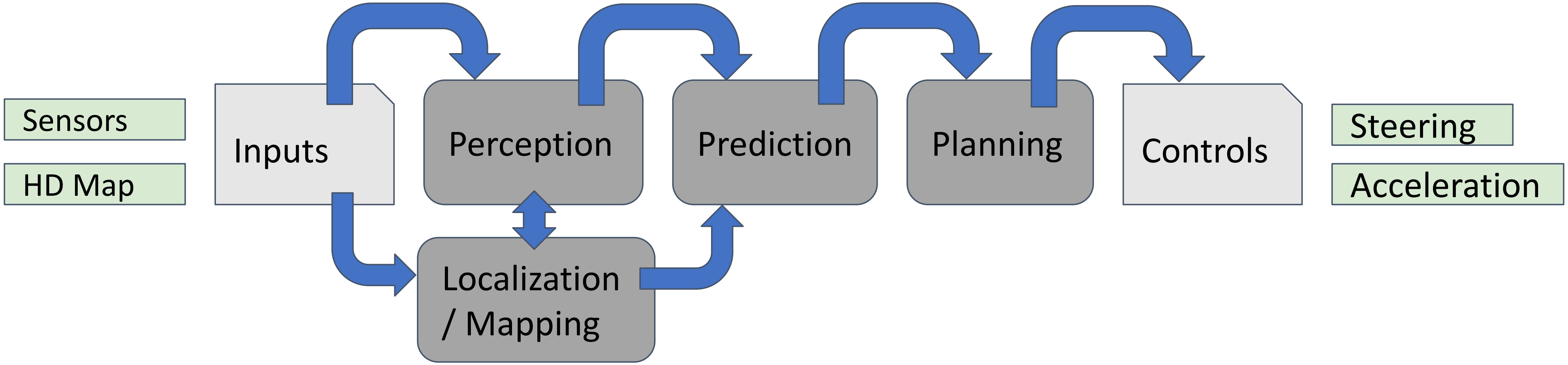
- **Open**-loop vs **Closed**-loop
    - L2-distance is not very important for closed-loop eval
- **Eval**-only vs **Train**+eval
    - The earlier to get the signal for the model, the better
- **Correlation** of MPCD/Disengagements with proxy metrics?
    - What are just regularization metrics for better train / faster eval?

Sadat, Abbas, et al. "Perceive, predict, and plan: Safe motion planning through interpretable semantic representations." 2020.

Do we really need to stick to the classical Autonomy Stack?

Stack

Classical **modular** structure



Each module:

- Has its **own** training / validation **data**
- Can be developed **independently**

Stack: unification?

Modular system being very useful still has **cons**:

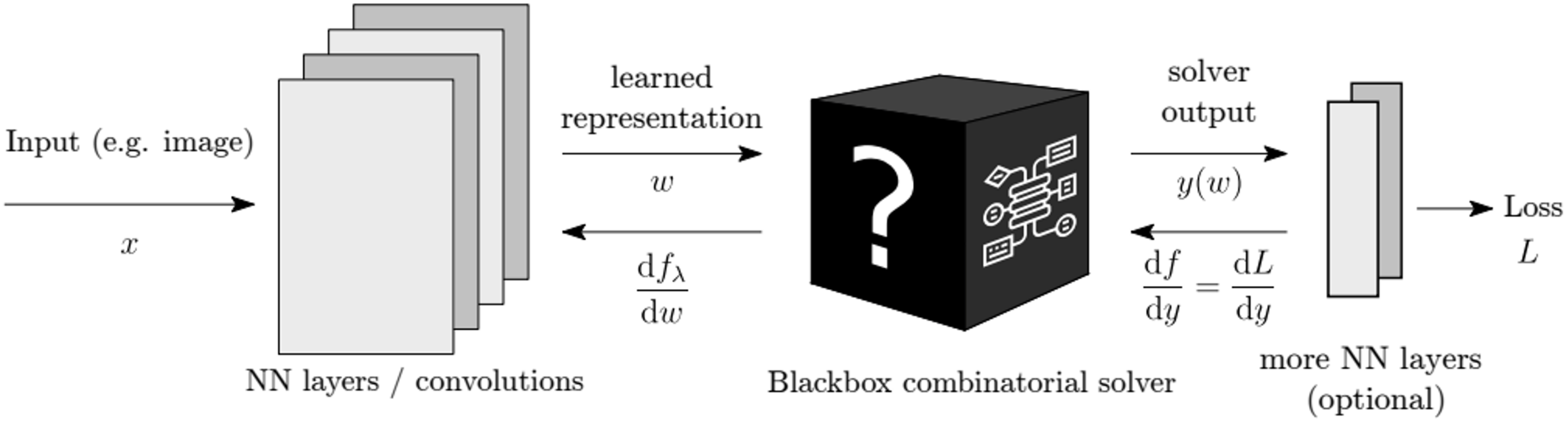
- **Sub-optimal** optimization and performance
- **Hard to propagate** uncertainty estimations

Would be **helpful**:

- To **propagate** the learning **signal** through the **whole** stack
- (Probably) **not to do end2end** approach like *Behavior Cloning* (or even *Imitation Learning*)

Is it **real**?

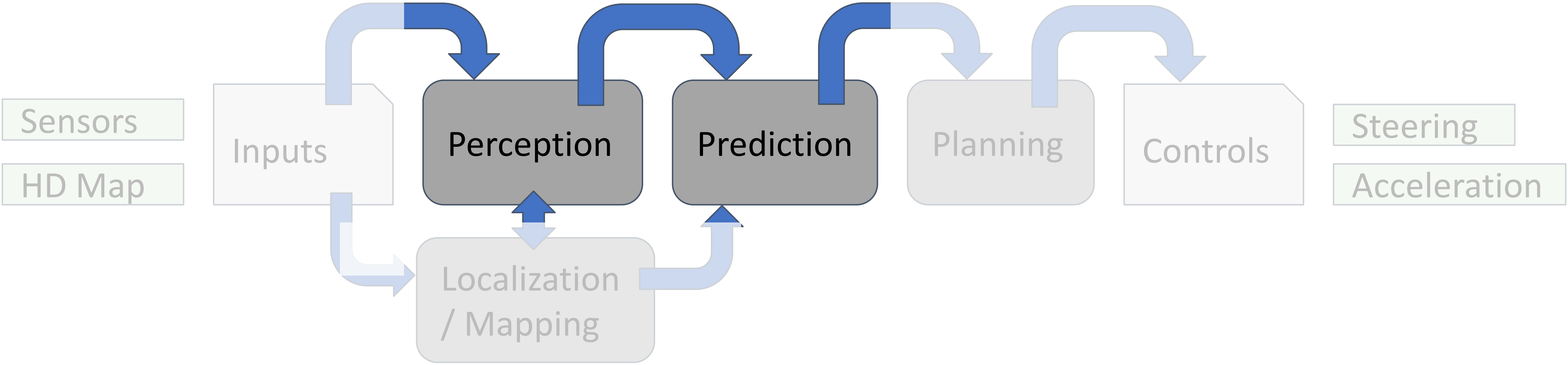
- The "**Theorem of existence**" provides the way to incorporate the non-differentiable modules into the pipeline
  - Although done for some narrow class of tasks



Vlastelica, Marin, et al. "[Differentiation of blackbox combinatorial solvers](). " 2019

Stack:
unification I

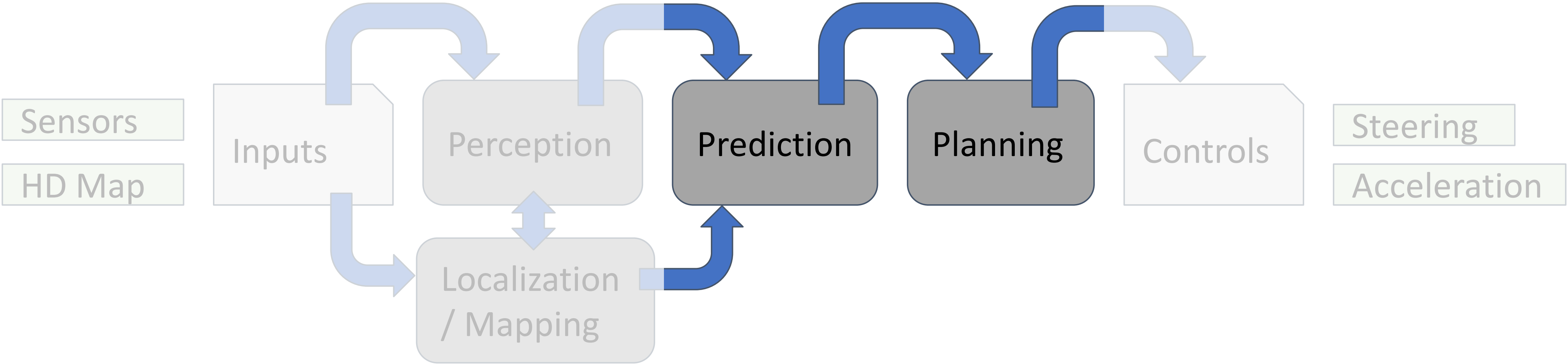
Combine: **Perception** + **Prediction**



Luo, Wenjie, et al. "*Fast and furious: Real time end-to-end 3d detection, tracking and motion forecasting with a single convolutional net*." 2018

Stack:
unification II

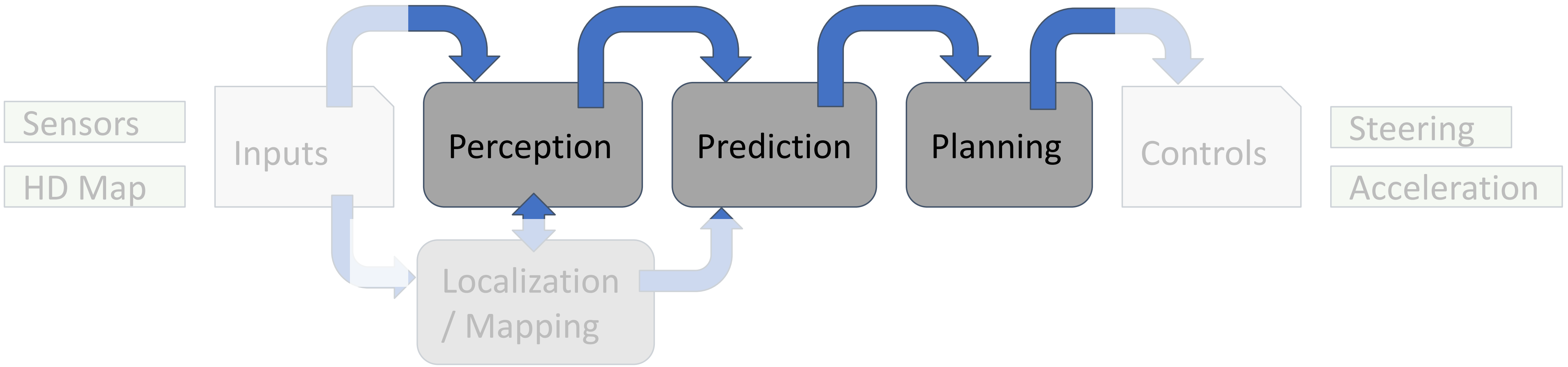
Combine: **Prediction + Planning**



Liu, Jerry, et al. "Deep structured reactive planning." 2021.

Stack: unification III

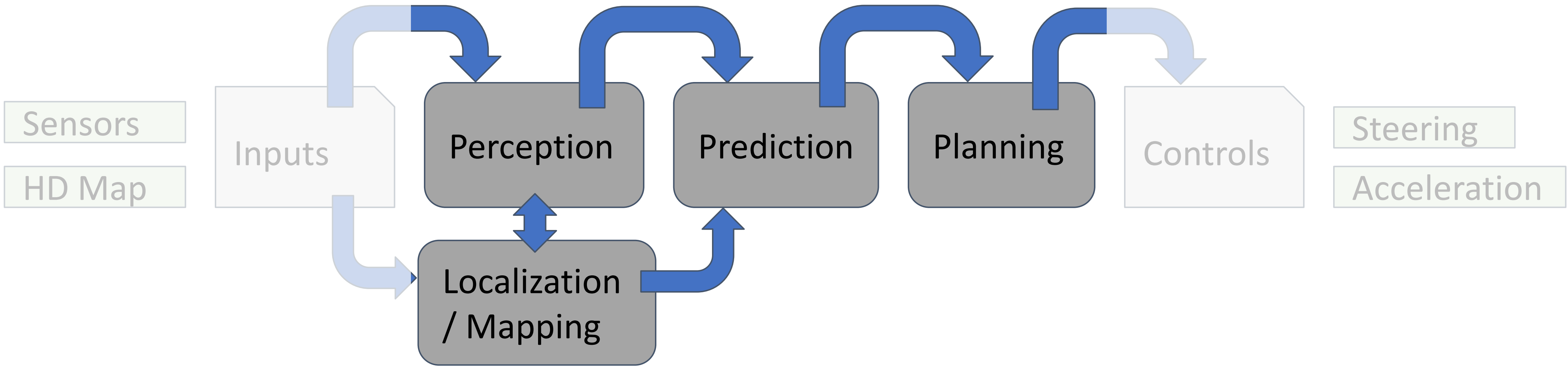
Combine: **Perception** + **Prediction + Planning**



Sadat, Abbas, et al. "Perceive, predict, and plan: Safe motion planning through interpretable semantic representations." 2020.

Stack:
unification IV

Combine: **Mapping** + **Perception** + **Prediction** + **Planning**

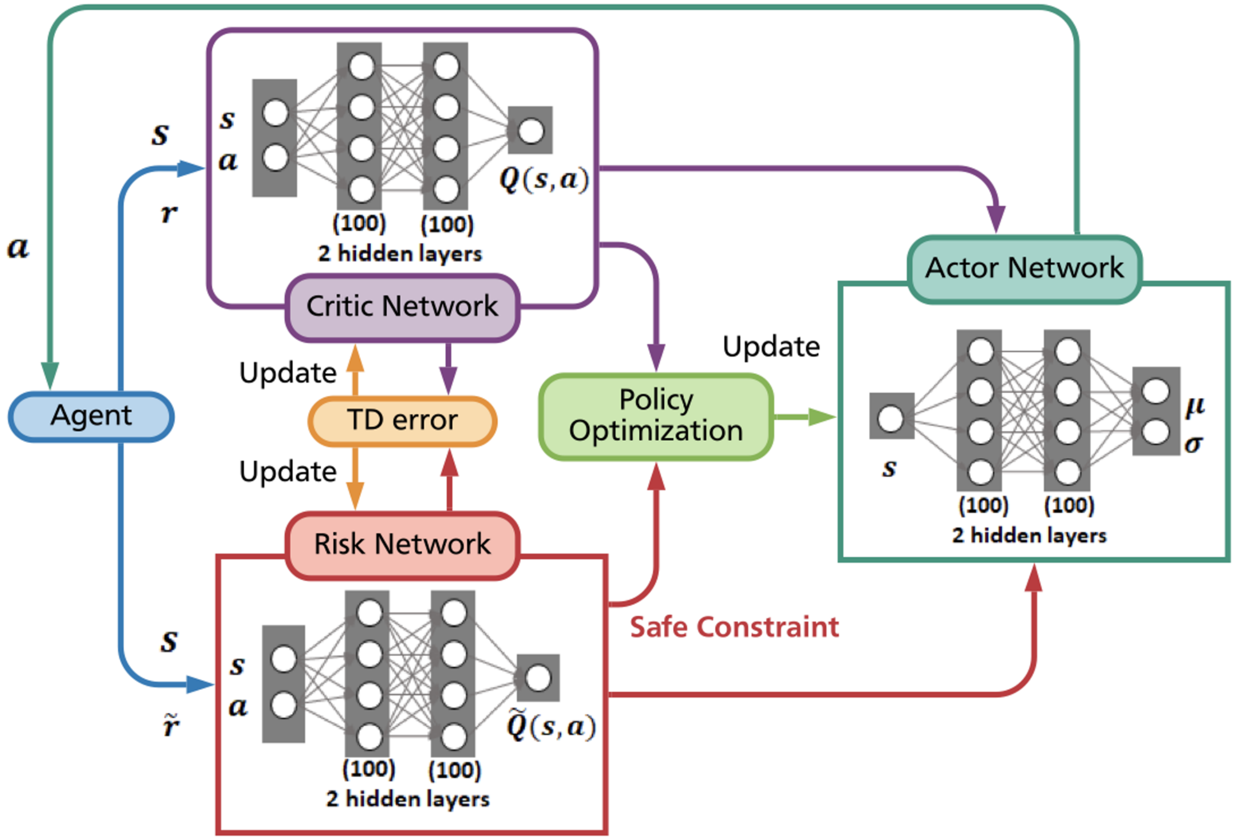


Casas, Sergio, et al. "Mp3: A unified model to map, perceive, predict and plan." 2021.

Stack and RL

Reinforcement Learning can be added for some of the modules combination

- Naturally integrates **planning**
- **State defines** the amount of input information (and the combination of modules as well)



Wen, Lu, et al. "Safe reinforcement learning for autonomous vehicles through parallel constrained policy optimization." 2020.

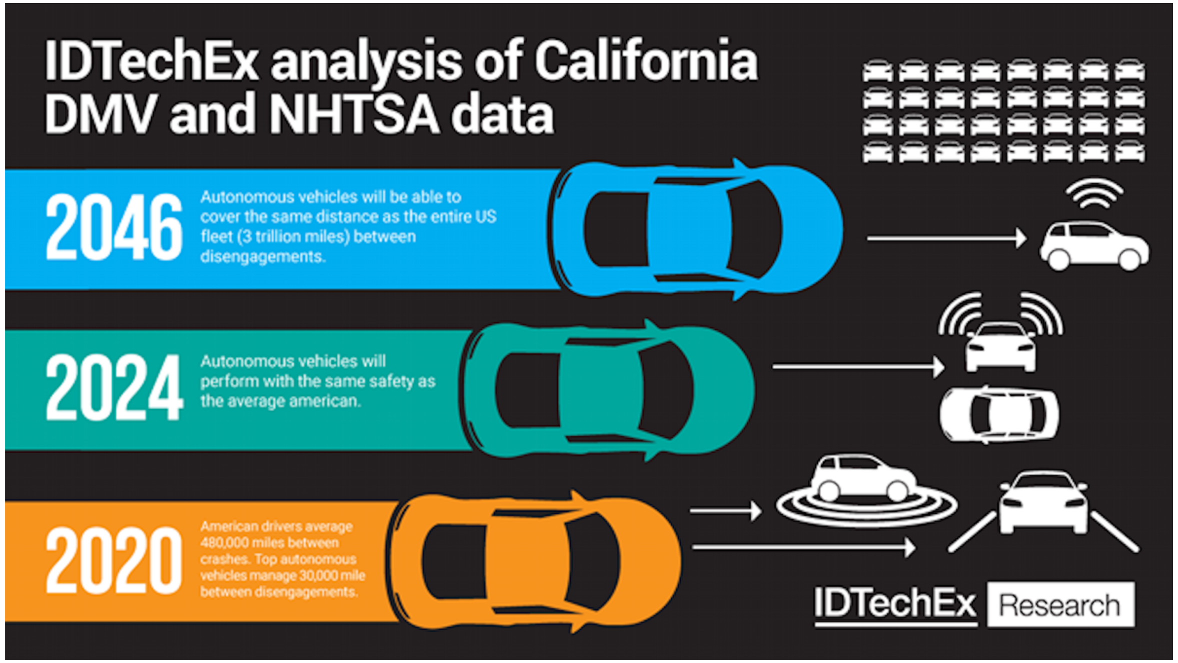
Intermediate Takeaways

- ➔ Hard to use common AV metrics for research
- ➔ Current closed-loop evaluation is still imperfect
- ➔ Need to understand what are discrepancies w.r.t. the real environments (distribution shift) and how to certify the current results (analytical guarantee)
- ➔ Eventually the technological approach can be much (or even completely) different from the classical one

Bright Future

Great **change** of paradigm:

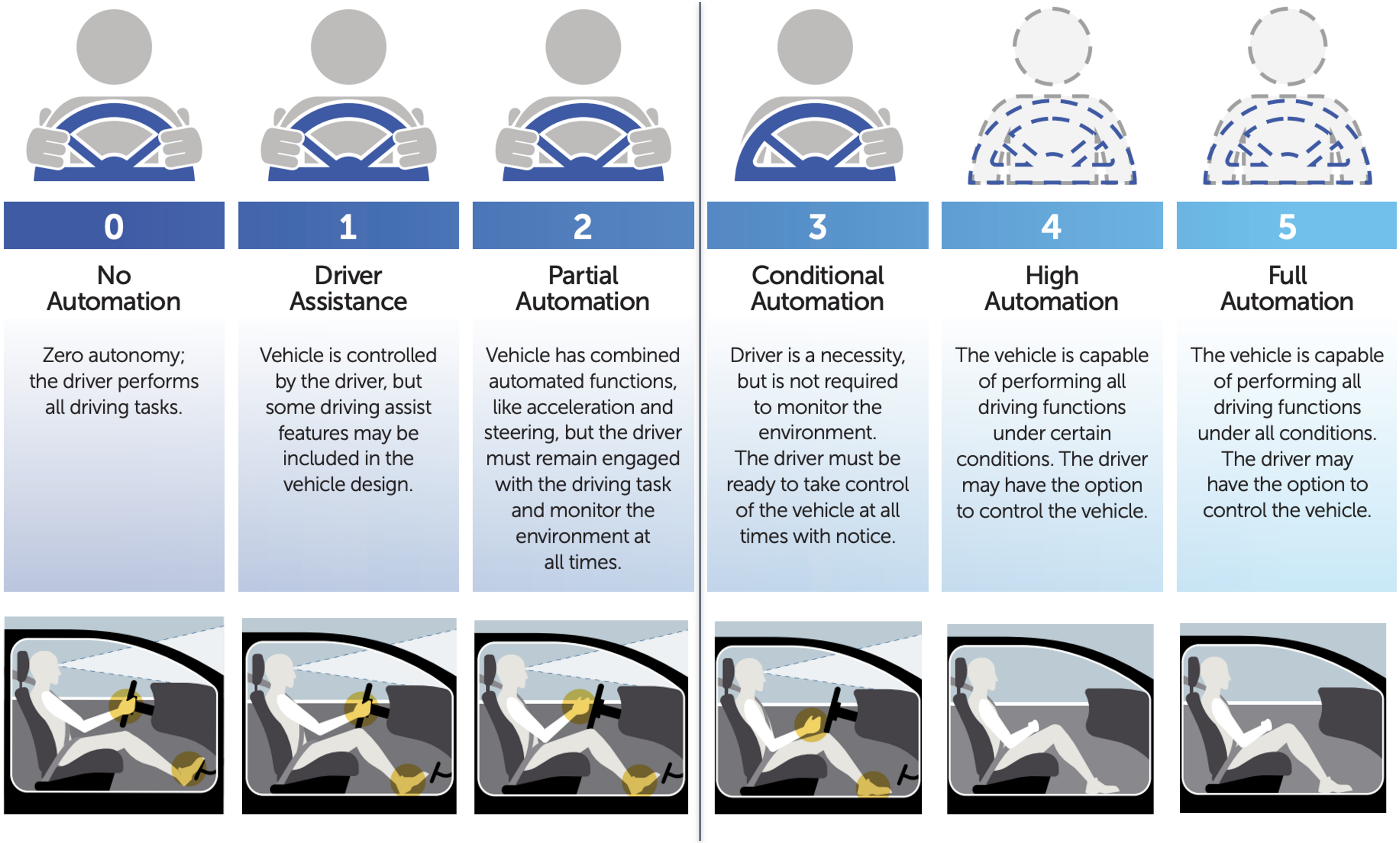
1. Be **as a human driver**:
   - **N** years?
2. Be **much better** as a human driver:
   - Is it really a jump of N→**NN** years?



Source: IDTechEx

Do we have the clear understanding / roadmap for introducing **high** Automation levels?

Levels of Automation



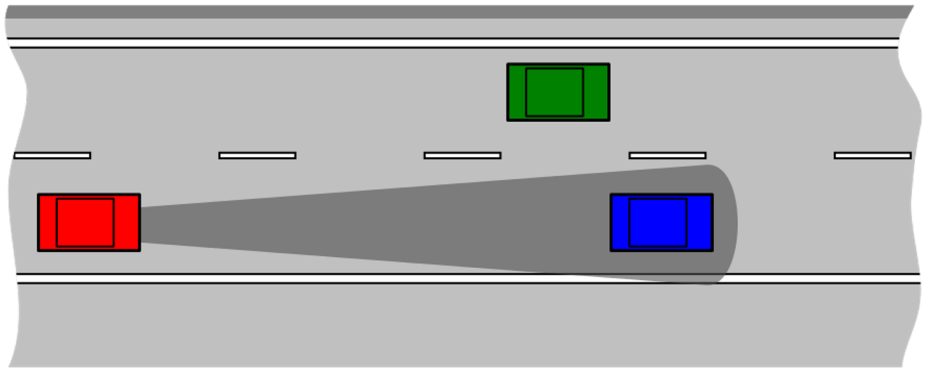
NHTSA: 1, 2 + SAE (Society of Automotive Engineers) J3016

Conditional Automation

Q: how to make **notice** for driver ***in advance***?
Is it **realistically** doable and useful?

Problem:
- Example: **collision avoidance signal**[1]
- **Time of human reaction**: 1-2 seconds[2]
- **False** positives avoidance **vs true** positives coverage

W/ and **w/o** waiting for the human **feedback**:
- **Automatic Emergency Braking**
  - Pros: greatly *reduces rear-end collisions* (by 40-50%)
  - Cons: still not ideal (have *hundreds per year accidents* caused by drivers placing too much confidence in automatic brakes)



0.7 sec -- about as fast as it gets
1.0 sec -- old standard
1.5 sec -- common use
2.0 sec -- common use
2.3 sec -- AVERAGE
2.5 sec -- used in a few states
3.0 sec -- NSC and UK Standard

Driver reaction times

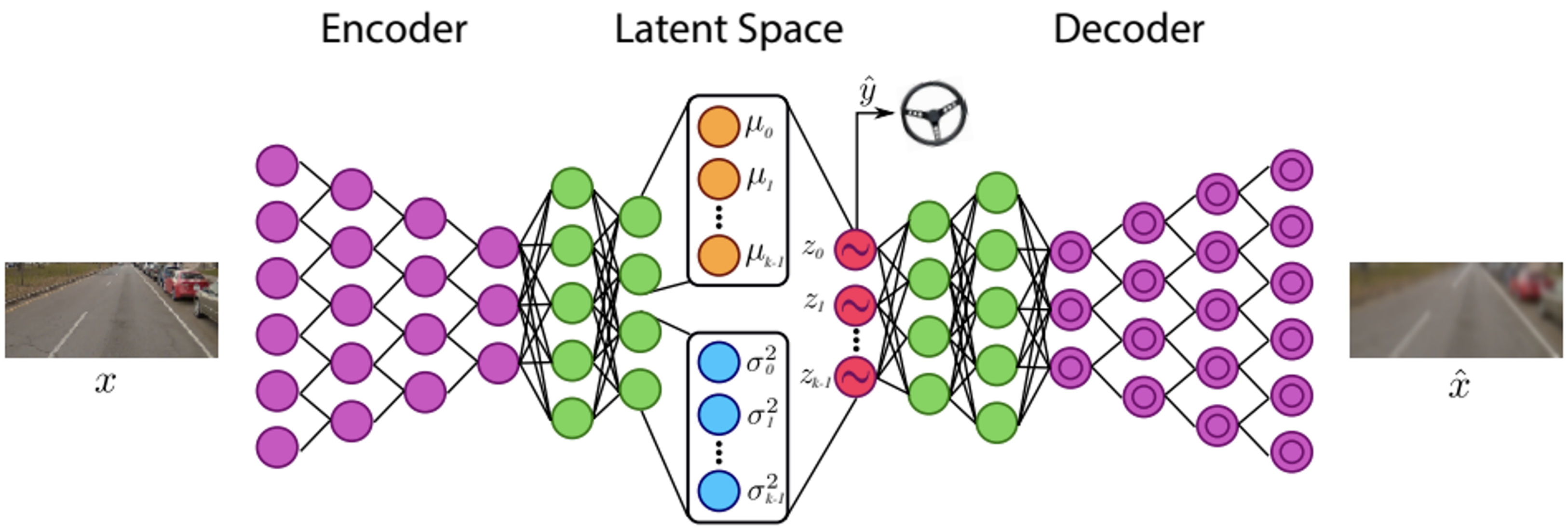
Wiki on Collision Avoidance System
McGehee, Daniel. et al. "Driver reaction time in crash avoidance research: Validation of a driving simulator study on a test track." 2000. + copradar.com

High vs Full Automation

Q: how to understand that we are **in** or **out** of our “**certain** conditions”?

Problem:
- need to understand the input **distribution shift**
- need to understand it for **every single module** inside the Autonomy Stack (e.g., Perception, Prediction, Planning, etc)

Possible **solution**:
- (Variational) **Autoencoders**[1]
  - Cons: How to behave if *OOD/Anomaly* (see “*Conditional Automation*”)?



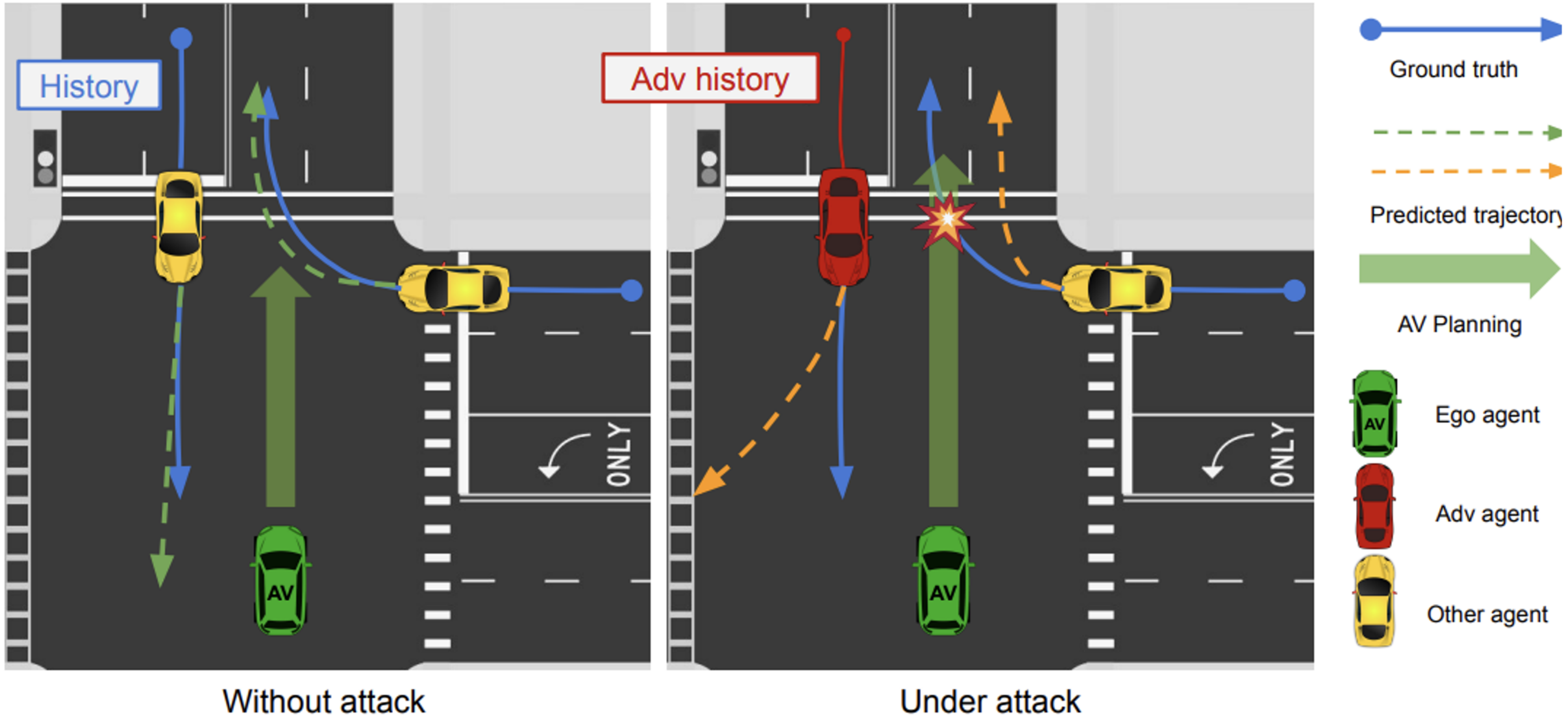
Amini, Alexander, et al. "Variational autoencoder for end-to-end control of autonomous driving with novelty detection and training de-biasing." 2018.

Full Automation

Q: how to make the model **working** for **all input** (even weird) conditions?

Problem:
- **known unknowns**: specific adversarial RL agents for the specifically designed scenario
- **unknown unknowns**: some physically plausible input providing “bad” outputs (e.g., collisions)

Possible **solutions**:
- **Adversarial RL** agents
  - Cons: *limited* by scenario generation and RL engine capabilities
- **Backpropagation**[1] w.r.t. Input
  - Cons: full-stack usually *hardly backpropagatable, constraint*s on Input



Cao, Yulong, et al. "[Advdo: Realistic adversarial attacks for trajectory prediction](#)." 2022.

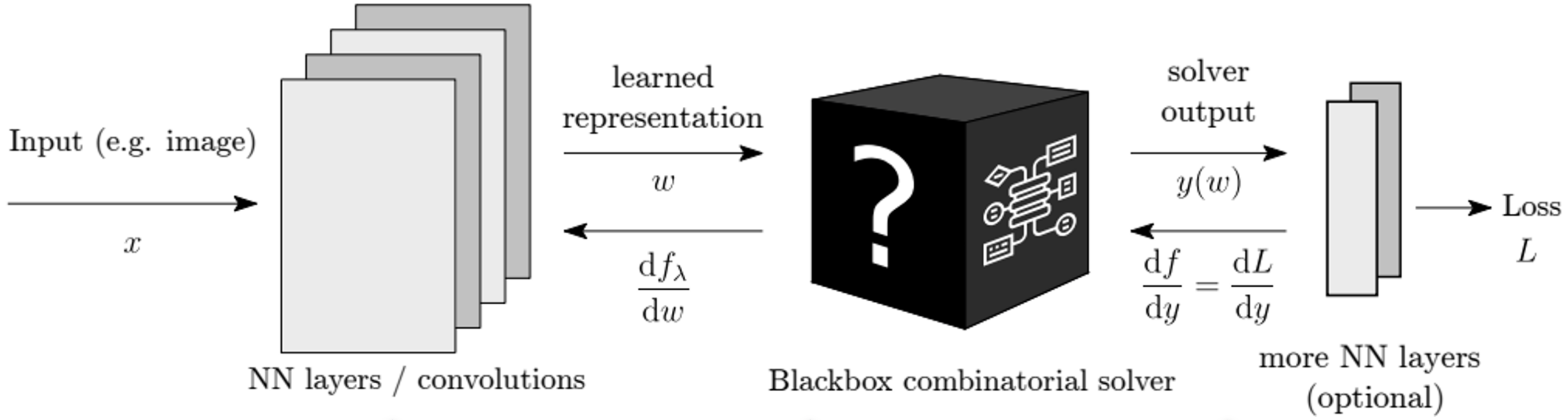
What could be the development stepping stones for reaching the self-driving?

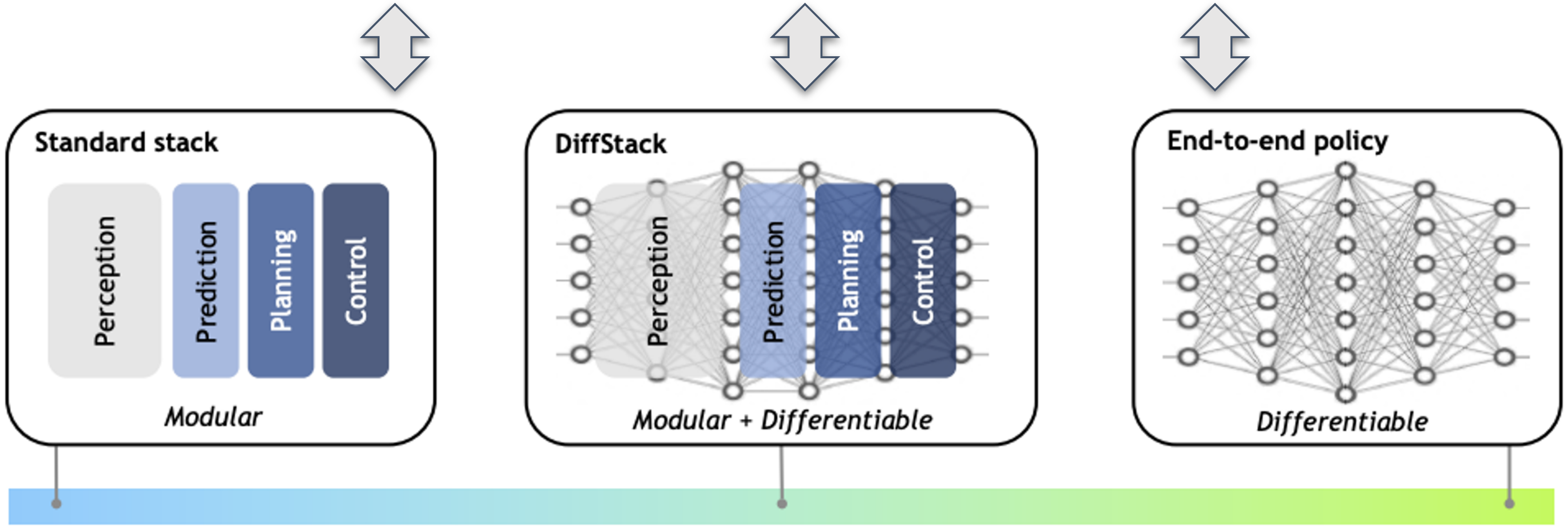
Differentiability

Q: how to propagate the learning signal (and uncertainty estimations) through the whole stack?

Problem:
- avoid **end2end** approach like *Behavior Cloning*
- **re-use** the existing modules and *expert* knowledge

Possible **solutions**:
- **Approximation** of non-differentiable modules by:
  - differentiable **wrapping**[1]
  - differentiable **approximation**[2]
  - Cons:
    - *constraints* on modules inside wrapping
    - *hard* / *slow* to approximate some existing modules (iLQR, sampling)





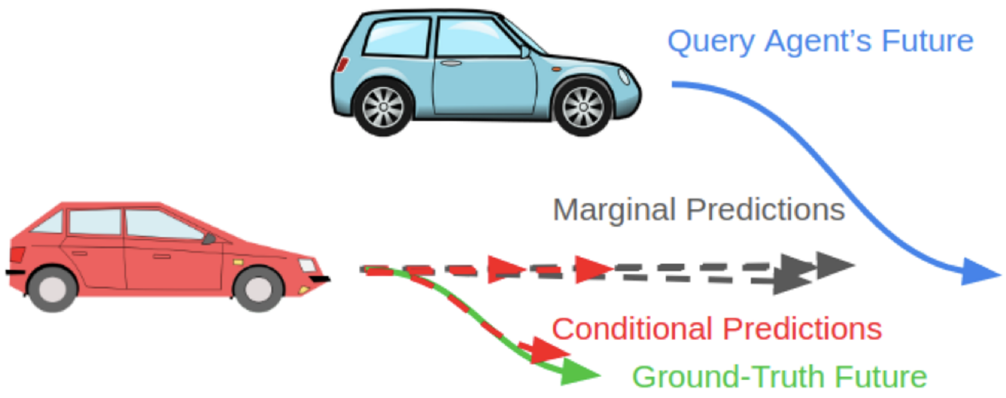
Vlastelica, Marin, et al. "[Differentiation of blackbox combinatorial solvers](). " 2019
Karkus, Peter, et al. "[DiffStack: A Differentiable and Modular Control Stack for Autonomous Vehicles.]()" 2022.

Jointness I

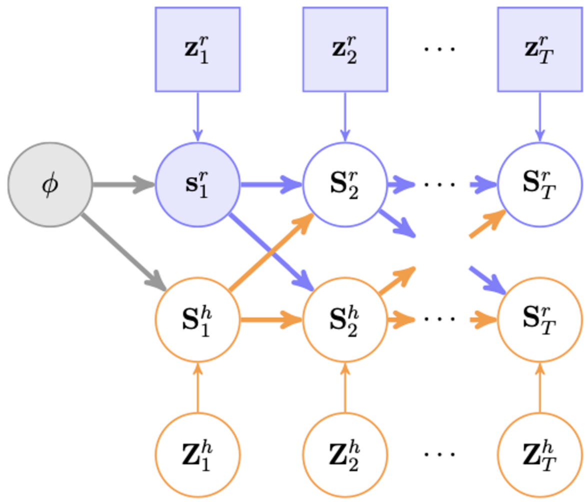
Q: how to **ensure consistency** between:
- **between prediction and planning,**
- different predictions,
and how to evaluate it?

Problem:
- **feedback loop** between the robot future and other road agents futures
- mining of **interactivity** scenes

Possible **solutions**:
- **Heuristically** (e.g., by distance) defining the interactive scenes/agents
- Conditional Behavior Prediction by the **new model input** (robot planned future)
- Conditioning in the **autoregressive** way



Conditional Behavior Prediction[1]



PRECog[2]

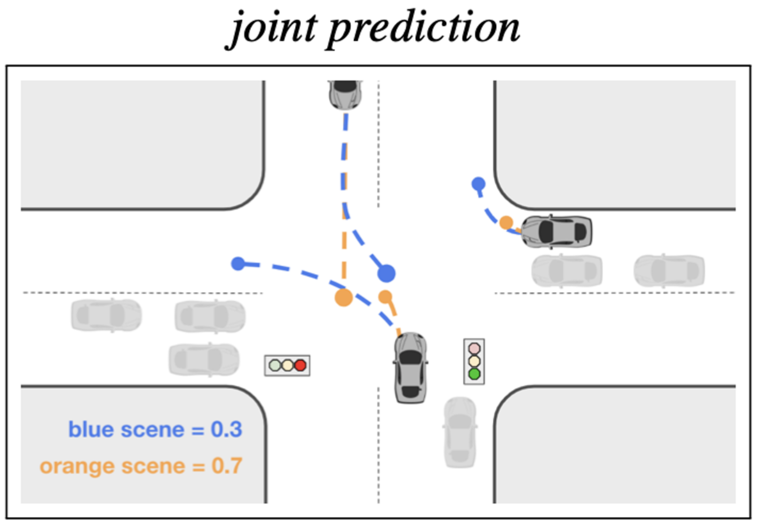
Tolstaya, Ekaterina, et al. "Identifying driver interactions via conditional behavior prediction." 2021
Rhinehart, Nicholas, et al. "Precog: Prediction conditioned on goals in visual multi-agent settings." 2019

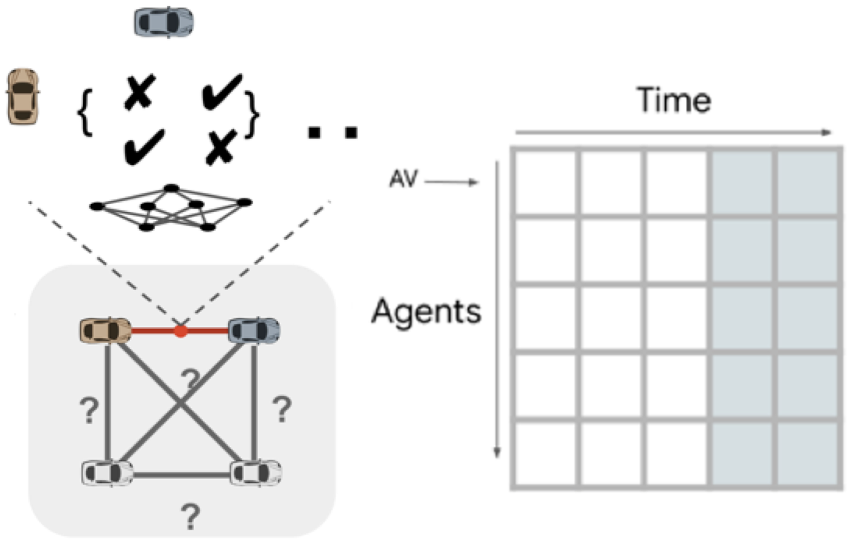
Jointness II

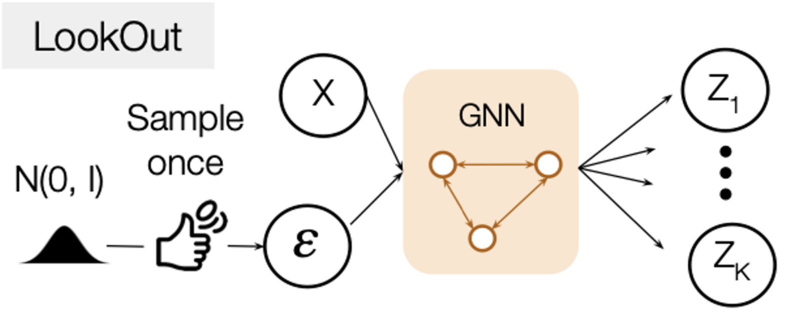
Q: how to **ensure consistency** between:
- between prediction and planning,
- **different predictions,**
and how to evaluate it?

Problem:
- working on top of **marginals** is **error-prone**
- considering all the combinations of agents leads to a **combinatorial** complexity **explosion**

Possible **solutions**:
- Different mitigations:
  - Joint pairwise by **message passing**[1]
  - Jointness by **transformer decoder**[2]
  - Jointness by the **unified latent**[3]
- These are still mitigations







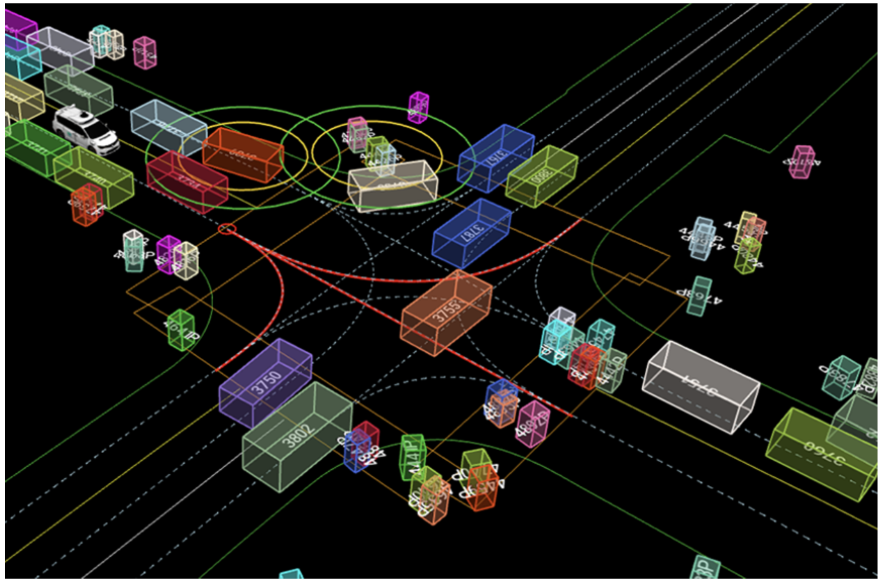
Luo, Wenjie, et al. "[JFP: Joint Future Prediction with Interactive Multi-Agent Modeling for Autonomous Driving]." 2023
Ngiam, Jiquan, et al. "[Scene Transformer: A unified architecture for predicting multiple agent trajectories]." 2021
Cui, Alexander, et al. "[Lookout: Diverse multi-future prediction and planning for self-driving.]" 2021

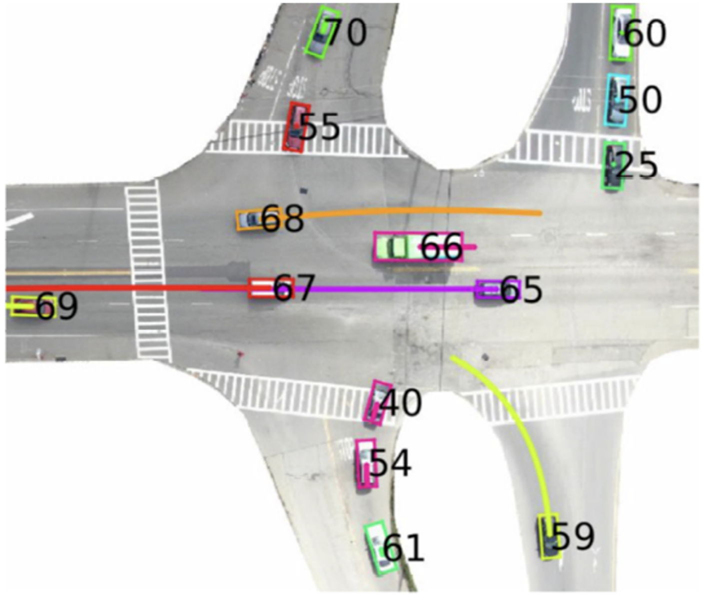
Jointness III

Q: how to **ensure consistency** between:
- between prediction and planning,
- different predictions,
and **how to evaluate it?**

Problem:
- need new **joint metrics**
- need public **datasets** and **challenges** supporting it

Possible **solutions**:
- **Scene-level** analogs of marginals
  - minSADE vs minADE
- **Waymo**[1] (pairwise joint) and **Interaction**[2] (pairwise and fully joint conditional) datasets





$$minADE = \frac{1}{l}\sum_{i=1}^{l} \min_{k} ||x_i^k - x_i^{gt}||$$

$$minSADE = \frac{1}{l}\min_{k} \sum_{i=1}^{l} ||x_{scene,i}^k - x_i^{gt}||$$

Ettinger, Scott, et al. "[Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset]." 2021
Zhan, Wei, et al. "[Interaction dataset: An international, adversarial and cooperative motion dataset in interactive driving scenarios with semantic maps]." 2019

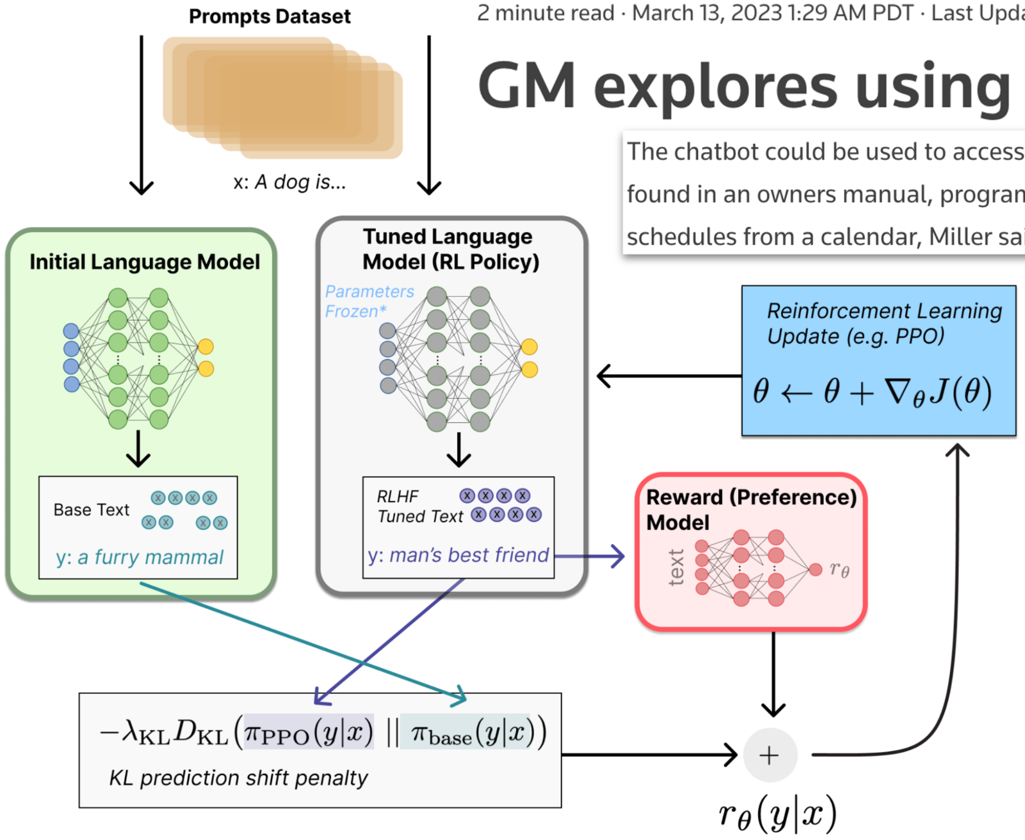
RL for AV

Q: how to **incorporate Reinforcement Learning** (RL) into the Autonomy Stack taking into account safety requirements?

Problem:
- Explicit Planning by RL is unstable / unreliable
- Hard to balance and optimize multiple safety constraints

Possible **solutions**:
- Instead of explicit Planning by RL, **fine-tuning by RL rollouts**
  - Cons: having the good model is a *chicken-egg* problem
- Usage of **Human Preference[2]** labels (RL from Human Feedback (HF)): ChatGPT[1]-like approach
  - Cons: 1) *absence* of a good *foundation* model for AD; 2) *hard* to get *lots of HF labels* for AV
- Still unknown what is the best way to **inject safety constraints** (and is it needed explicitly?)

OpenAI: [ChatGPT]
Hugginface: [RL from HF]
Reuters: [GM explores using ChatGPT in vehicles]



[3]

2 minute read · March 13, 2023 1:29 AM PDT · Last Updated 7 days ago

GM explores using ChatGPT in vehicles

The chatbot could be used to access information on how to use vehicle features normally found in an owners manual, program functions such as a garage door code or integrate schedules from a calendar, Miller said.

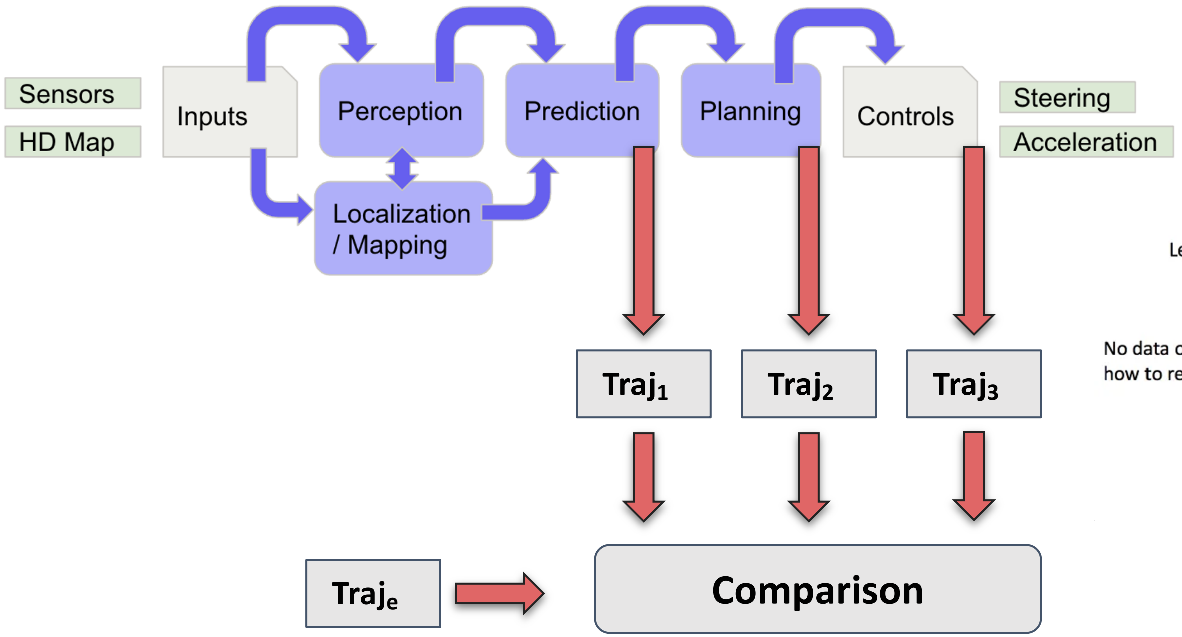
How to evaluate our progress being engineers?

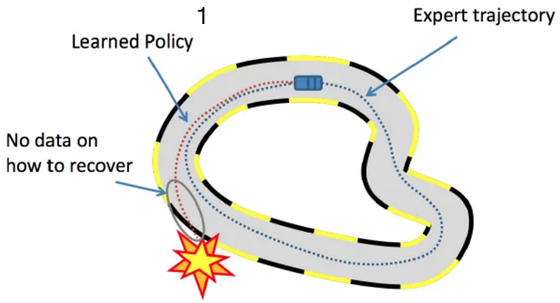
Evaluation

Q: how to make the evaluation process be **less costly** and **faster**?

Problem:
- **how** (metrics) and **where** (modular vs end2end) to evaluate?
- need in **submodular** eval?

Possible **solutions**:
- **End2end comparison** with the human expert
  - Cons: it is only Imitation Learning-like metric
- **Submodular comparison** with the human expert
  - Cons: need to produce the robot trajectory *as soon as possible*
- *Necessity* vs *sufficiency*





Medium: [Imitation Learning](), 2019

Conclusion

- ➔ Formal Automation Levels definition are not clarifying the possible approaches to reach them
- ➔ Stepping stones towards the full self-driving are reasonable but not set in stone
- ➔ Consistency in a model output is going to be a trend; but need deeper support from datasets/metrics/challenges
- ➔ Evaluation is painful
- ➔ "*ADGPT*" to the rescue?

Links

- **Introduction:** Autonomy: Introduction of ML for High School
- **Part I**: Autonomy Challenges (presentation, video)
- **Part II**: Autonomy: Open Questions

Thank you!