# Module 5: Principles of Behavior Planning

## Lesson 1: Behavior Planning

内容
- Define a behavior planning system.
- The standard input and output of a behavior planner.
- State machines.

Behavior Planning
- A behavior planning system plans the set of high level driving actions, or maneuvers to safely achieve the driving mission under various driving situations.
- Behavior planner considers.
    - Rules of the road.
    - Static objects around the vehicle.
    - Dynamic objects around the vehicle.
- Planned path must be safe and efficient.
- The behavior planners should also be able to deal with inputs that are both inaccurate, corrupted by measurement noise, and incorrect, affected by perception errors such as false positive detections and false negative detections.

Driving Maneuvers
- Track speed.
    - maintain current speed of the road.
- Follow leader.
    - match the speed of the leading vehicle and maintain a safe distance.
- Decelerate to stop.
    - begin decelerating and stop before a given space.
    - A stop point exists in the ego vehicle’s lane within the planning horizon.
- Stop.
    - remain stopped in the current position.
    - 例：when the vehicle stops at a stop sign, it should stay stopped for at least 3 seconds.
- Merge.
    - join or switch onto a new drive lane.

Output of Behavior Planner（大事）
- Driving maneuver to be executed.
- Set of constraints which must be obeyed by the planned trajectory of the self driving car.
    - Ideal path.
    - Speed limit.
    - Lane boundaries.
    - Stop locations.
    - Set of interest vehicles.
        - These dynamic objects may be important due to proximity or estimated future path.
    - つまりbehaviorは全然pathを出さない？ただlocal plannerのinputsを用意する？多分planned pathというのは、ideal pathだ。もちろんこの段階で既に環境を考えた。なぜなら、考えないと、behaviorを出せないでしょう！結局behavior plannerの出力はまだ分かっていない。
        - 復習：Behavior planner considers rules of the road, static and dynamic objects around the vehicle.
        - Behavior plannerはplanned pathを出さないと思うけど、merge以外のpath (ideal path)は大体自車線中心線なので、mergeの時のみideal pathを修正する必要がある。なかなか

behavior plannerが出すのは、planned pathという期間指令ではなく、ただの瞬時指令だ。例えば、mergeの時、merge pathは出すの?
- Constrain the local planning problem.

Input Requirements（大事、膨大）
- High definition road map.
- Mission path.
  - Dijkstra, A*, road network.
- Localization information.
- Perception information.
  - All observed dynamic objects.
    - current state.
    - predicted paths.
    - collision points.
    - time to collision.
      - しかもcollision pointsやtime to collisionの計算全部simulation-basedじゃない!
  - All observed static objects and their respective states.
    - road signs.
    - parked vehicles.
    - construction cones.
    - traffic lights.
      - with an indication of their state.
  - Occupancy grid.
    - defining the safe areas to execute maneuvers.

Finite State Machines：Simulinkだ。
- Each state is a driving maneuver.
  - Each state has associated with it an entry action, which is the action taken when a state is first entered.
  - Entry actions involve setting the necessary constraint outputs to accompany the behavior decision.
    - entry actionは例えば目標速度とか。
- Transitions.
- Transition conditions.
- Entry actions are modification to the constraints.

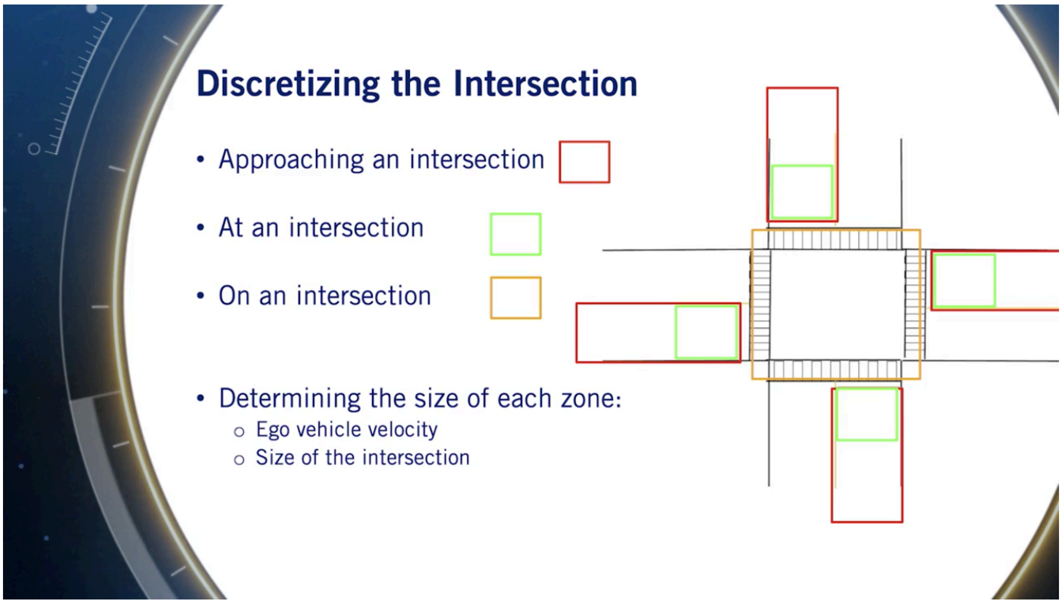
Advantages of Finite State Machines in Behavior Planning
- Limiting number of rule checks.
- Rule become more targeted and simple.
- Implementation of the behavior planner becomes simpler.
- 課題：（大事）
  - as the number of states increases, the complexity of defining all possible transitions and their conditions explodes.
  - There's also no explicit way to handle uncertainty and errors in the input data.
  - These challenges mean that the finite-state machine approach tends to run into difficulties, as we approach full level-5 autonomy.

単語
- Proximity, 近接: Proximity to a place or person is nearness to that place or person.
- Cone, 円錐「えんすい」: A cone is a shape with a circular base ending in a point at the top.

# Lesson 2: Handling an Intersection Scenario Without Dynamic Objects

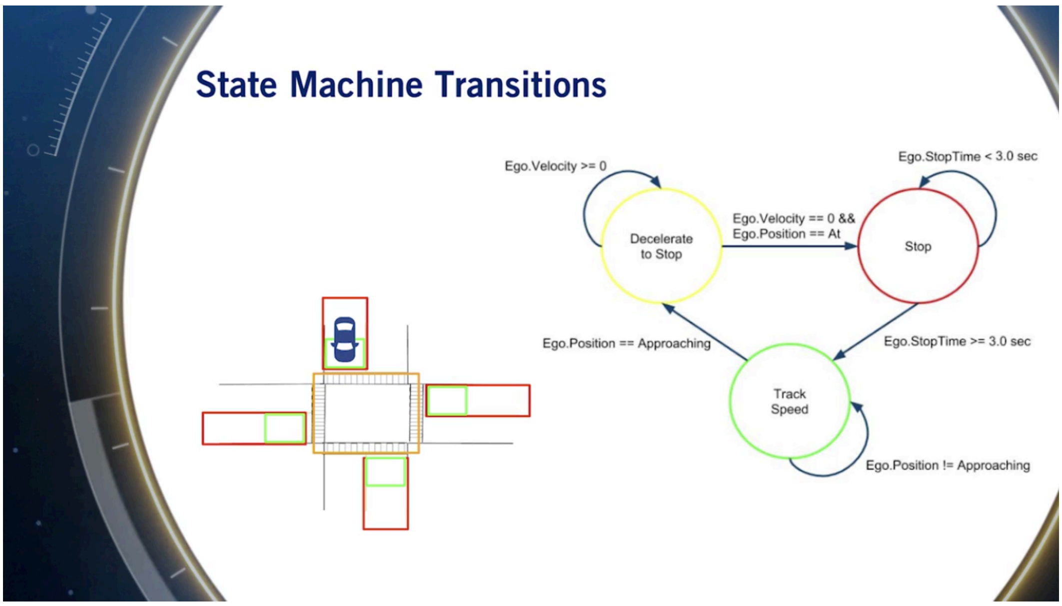
内容

- Identify the intersection scenario that will be handled.
- Discuss the discretization of the environment that will be used.
- The states required to complete the scenario.
- Create the state transitions and state outputs required to safely and effectively complete the scenario.
- Highlighting testing procedures to confirm a correct and accurate system.



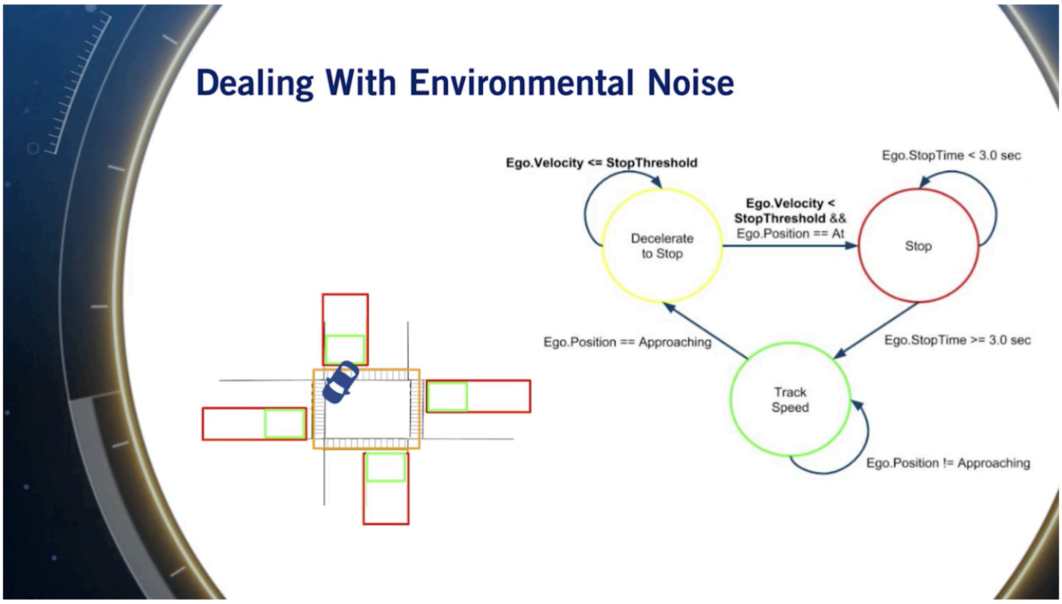

Discretizing the Intersection
- Approaching Zone.
- At Zone.
- On Zone.
- Determining the size of each zone.
  - Ego vehicle velocity.
  - Size of the intersection.

State Machine States
- Track Speed.
  - Traditionally, this is the maneuver given before entering any region of the intersection, or after entering the on the intersection zone safely.
  - Entry action: speed limit.



- Decelerate to Stop.
  - Entry action: stop point location.
- Stop.
  - Entry action: start a timer to wait for a fixed amount of time before proceeding through the intersection.

State Machine Transitions
- Throughout this process, it is vitally important that we understand how we, the human expert, analyze the scenario, and what the specific capabilities of the resulting behavior planner are.
- These need to be captured in the operational design domain definition, and we need to ensure that we create a complete state machine able to handle every possible case that can arise for the given scenarios.




**EGO.VELOCITY>=STOP_THRESHOLDのはず**

Dealing with Environmental Noise
- One particular issue that has a big impact on the performance of our state machine is the issue of noise in the inputs.
- The state transition conditions defined above are exact, and rely on the vehicle reaching the stop point and achieving a zero velocity exactly.
- Even with no other dynamic objects to detect, the localization estimates of the vehicle state may contain noise and not satisfy these conditions exactly.
- 対策：noise threshold hyperparameters.
  - Small threshold value allowing speeds close to zero to be accepted as stopped.

Behavior Planning Testing（大事）
- Code based tests.
  - Incapable of confirming if the state transitions are correct, or if the states are capable of handling all of the situations in a given scenario.
- Simulation tests.
  - Performed in a simulated environment like Carla.
  - つまりADSTの仕事をCarlaでやるのが可能だ！なぜADSTでやるのは意味少ない？ Behavior Plannerのinputsはそもそも複雑だ！例えばPerception stackを使えること、Localizationを使えること、HD Mapを使えることなどなど！
  - Able to confirm if the state machine transitions and state coverage are correct.

    - The number of tests performed in the simulation should be representative of all possible situations which can be seen when driving the scenario to catch any edge cases which programmers might have missed.
    - Many times selecting a representative set of tests is not trivial, especially as the complexity of the scenarios increases.
- Private track tests.
    - once confident that the state machine performs as intended in simulation.
    - This type of testing tests specific scenarios which are hard to confirm exactly in simulation, such as parameter tuning and noise, and errors in the perception output in a real environment.
        - つまりSimulationで現実的なノイズやエラーを模擬しにくい。特にPerceptionでしょう。いいえ、そもそも実車実験は必須だ、Simulationで何も模擬できても。
- Limited scoped close supervision road tests.

# Lesson 3: Handling an Intersection Scenario with Dynamic Objects

内容
- Include dynamic objects as part of the state machine.
- Develop an understanding of the complexities and edge cases when dealing with dynamic objects.

Review - Interaction with Dynamic Objects
- Distance to dynamic object.
    - distance to the center of any dynamic object.
- Distance to collision point.
- Time to collision (TTC).

State Machine States
- Track Speed, Decelerate to Stop, Stop.
- Follow Leader.
    - The safe distance is speed dependent and both safe speed and safe distance are updated as entry actions on every iteration of the behavior planner.

State Machine Transitions - Track Speed
- Another vehicle appears or enters the lane in front of the ego vehicle.前車が入り込んだ。
    - Follow check.
        - A distance check.
        - A same lane check.
            - For simplicity, check if the lead vehicle is within the lane limits and if the heading of the dynamic object is within a given threshold to that of the ego vehicle.

State Machine Transitions - Follow Leader
- The lead vehicle might exit the lane before the ego vehicle is approaching or at the intersection. 前車が車線変更しちゃった。
- The lead vehicle drives out of the current lane onto the intersection. 前車が交差点に入った。

State Machine Transitions - Decelerate to Stop
- A vehicle is able to pull in front of the ego vehicle at an intersection, either from a driveway or an overtaking maneuver.自車が減速中、前方車が入り込んだ。
    - follow check, throughout Decelerate to Stop.
        - the distance check considers whether the lead vehicle distance to collision is less than the stop point distance.

State Machine Transitions - Stop
- Turn left, any vehicle approaching from the left, right, or any oncoming vehicle, must clear the on intersection zone, before the ego vehicle can proceed.

- Go straight, only vehicles approaching from the left or right need to clear the intersection.
- Turn right, only vehicles approaching from the left need to clear the intersection.

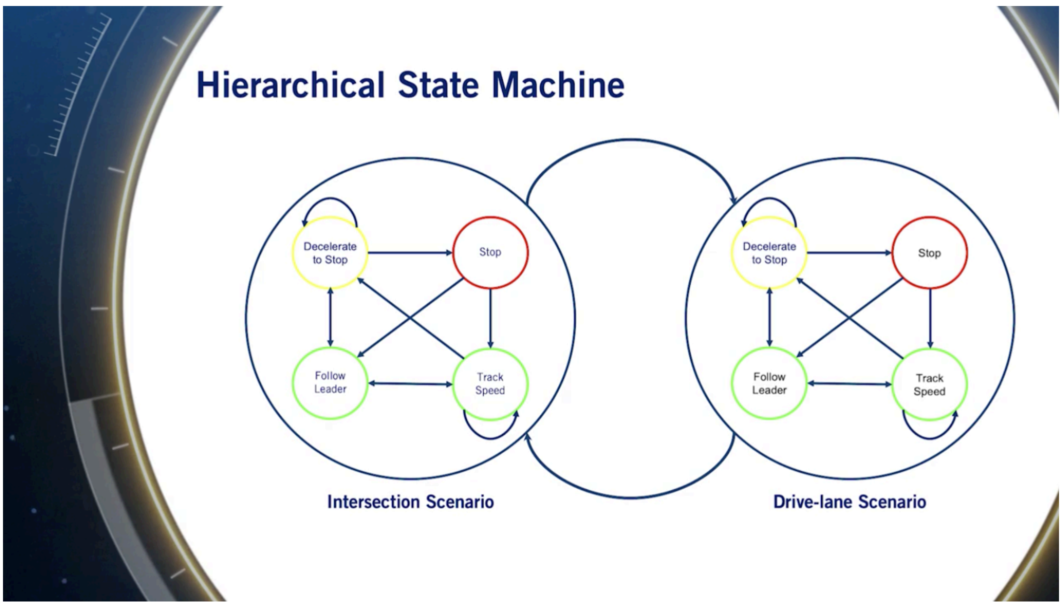
Dynamic Object Edge Cases Not Handled
- Assumption: All dynamic obstacles obey rules of the road.
- Not always the case!
- 例１： 対向車が急に中央分離帯を超えて自車線に入った。（対向車は逆行車に避けたかった。。。）Waymo 2018。ビデオ： https://techcrunch.com/2018/05/04/waymo-van-involved-in-serious-collision-in-arizona/
- 例２： An aggressive driver racing through an intersection even as the ego vehicle has begun driving through it.
- 例３： A vehicle fails to stop at the intersection.
- 例４： When a vehicle is parked in close proximity to the intersection.
  - If this vehicle is not tagged as parked, so that it might be treated as a static instead of dynamic object, our behavior planner may get stuck in a deadlock state waiting for this parked vehicle to move.面白い例！
- One of the primary objectives of behaviors safety assessment and testing, is to uncover as many variations of unexpected behaviors as possible, so that they also can be detected, categorized, and incorporated into the behavior planning design process.

単語
- Driveway, 私道「しどう」: A driveway is a piece of hard ground that leads from the road to the front of a house, garage, or other building.
- Overtake, 追い越す: If you overtake, or overtake a vehicle or a person that is ahead of you and moving in the same direction, you pass them.
- Rectilinear: in, moving in, or characterized by a straight line or lines.

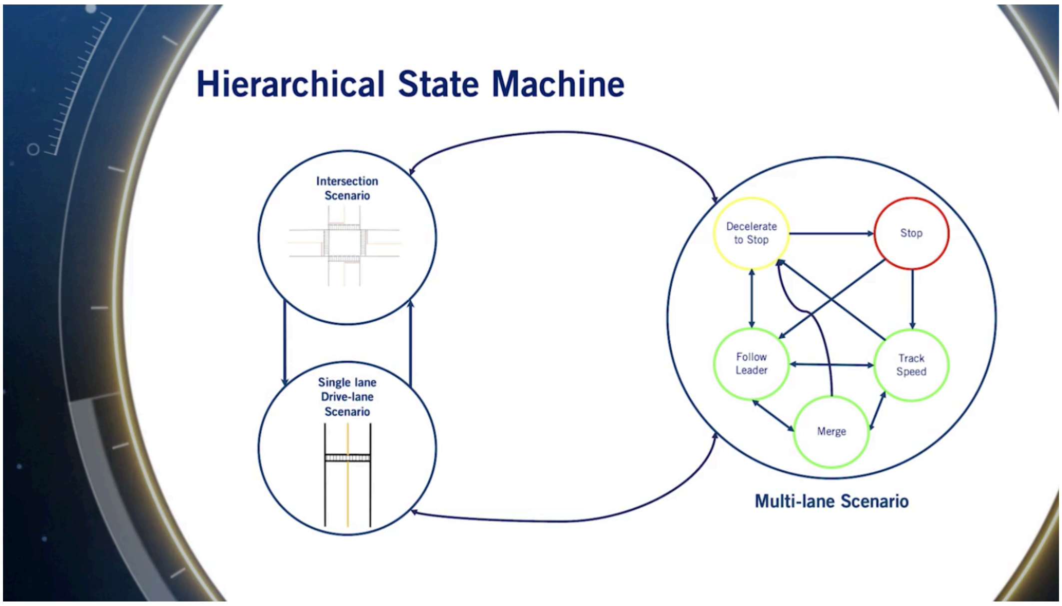
# Lesson 4: Handling Multiple Scenarios

内容
- Develop a larger overarching state machine which includes multiple scenarios.
- Develop a method to switch between driving scenarios.

Multiple Scenariosの弱い案： Single State Machine
- Single state machine method.
  - Add transitions.
  - Add additional transition conditions.
- Issues with single state machine method.
  - Rule explosion.



- Increase in computational time.
- Complicated to create and maintain.
- Essentially, as the complexity increases, we lose all the advantages of using a finite state machine.




Multiple State Machine
- Represent each high level scenario as a single state.
- Create much simpler transitions in between the high level scenario states.
- 質問：how do we switch between scenarios when processing the sub-state maneuvers?
    - Entry and Exit Transitions.
        - introduce transitions to key maneuver sub-states out of the current scenario super-state.
        - 例えばIntersection.
            - まずkey exit stateを決める。
                - The only way we're able to exit an intersection is while in the Track Speed or Follow Leader states after having passed the intersection.
        - With this method of transition, able to also maintain maneuvers between scenario switches.
            - In this case, Track Speed in the intersection super-state will connect to the Track Speed of the next scenario that we enter.（大事）

Hierarchical State Machine - Advantages and Disadvantages
- Advantages.
    - Decrease in computational time.
    - Simpler to create and maintain.
- Disadvantages.
    - Rule Explosion.
        - Repetition of many rules in the low level state machines.

僕の直感だけど、State Machineのやり方はあんまり優れていない。入力情報の豊かさと合わないし、制御の選択肢の少なさとも合わない気がする。人の設計に依存しすぎる? 自動運転車の学習力は?

単語
- Overarching, 包括的「ほうかつ」: You use overarching to indicate that you are talking about something that includes or affects everything or everyone.

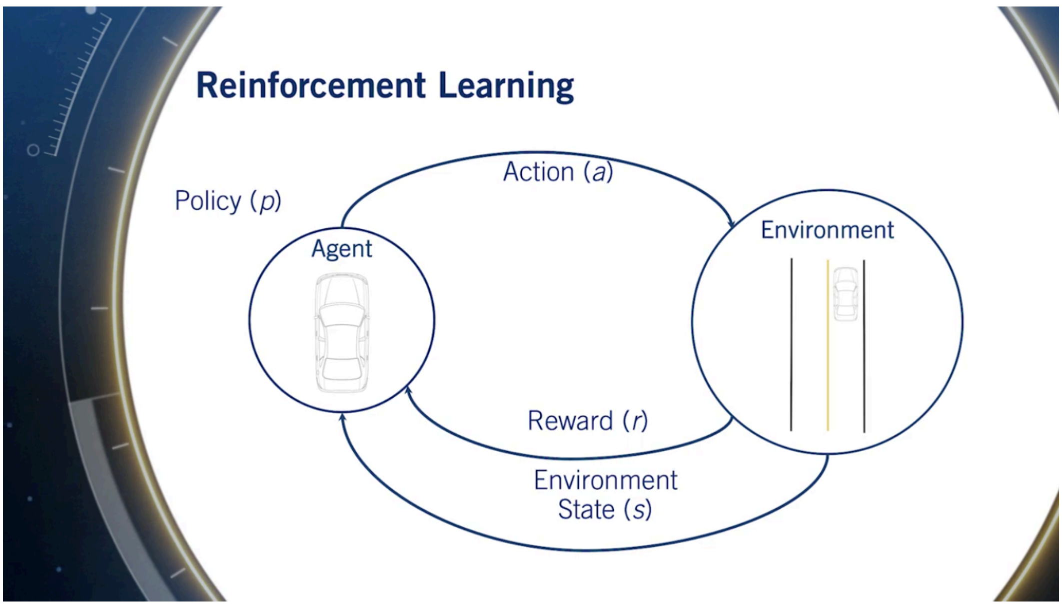
# Lesson 5: Advanced Methods for Behavior Planning

内容
- Issues with the state machine based behavior planner.
- Identify the open areas of research in behavior planning.

State Machine Behavior Planning Issues（大事）
- Rule-explosion when dealing with Complex Scenarios.
  - Developing a full level 4 or 5 capable vehicle is almost impossible with finite state machines.
- Dealing with a Noisy Environment.
- Hyperparameter Tuning.
- Incapable of dealing with Unencountered Scenarios.
  - Due to the program nature of this approach it is very likely that there will arise a situation in which the programmed logic of the system will react in an incorrect or unintended manner.
  - State machine based approaches are classified as experts systems, systems which have been designed and developed by human experts.

Rule-based Behavior Planner
- Hierarchy of rules.
  - Safety critical.
  - Defensive driving.
  - Ride comfort.
  - Nominal behaviors.
- Reduced need for duplication.
  - Rules can apply throughout ODD.
- Suffer from same challenges as finite state machines.
  - Common to all expert system designs.
- Rule based systems have to very carefully develop rules in such a way that they do not negatively impact each other, or lead to unintended outcomes when multiple rules are activated at the same time.

Fuzzy Logic
- 例：vehicle following behavior.
  - While previously we set a parametrized distance which divided the space into “Follow the Vehicle” or “Do not Follow the Vehicle”.
  - With a Fuzzy system we’re able to have a continuous space over which different rules can be applied.
- Fuzzy systemの課題：While Fuzzy based rule systems are able to deal with the environmental noise of a system to a great degree than traditional discreet systems, both rule-explosion and hyperparameter tuning remain issues with Fuzzy systems.



- In fact, Fuzzy systems can result in rule-explosion to an even greater degree, as even more logic is required to handle the fuzzy set of inputs.

Reinforcement Learning（大事）
- By learning in a simulated environment, no real world repercussions occurred during the many failures experienced during learning.
- The agent is attempting to learn a policy $p$, which maps a given environment $s$, to a given action, $a$.
- 例：vehicle following.
  - Our actions may be explicitly defined as behaviors to execute, or can define behaviors implicitly through selection of desired accelerations and turn rates.
  - The environment can be represented by a continuous set of variables telling us relevant information, such as distance to all objects and the mission path.
  - The reward function could be based on an optimal following distance.
  - Give the highest reward at the preferred distance, penalizing getting too close more heavily than being too far away.
- Because of the extremely large variety of scenarios and inputs that an autonomous vehicle can encounter, direct reinforcement learning for behavior planning is unlikely to succeed.
- Adaptations.
  - Hierarchical Reinforcement Learning.
    - Divide the problem into low level policies in the maneuver space and high level policies the scenarios.
    - Similar to the hierarchical finite state machine.
  - Model-based Reinforcement Learning.
    - policy $p$以外、Environmentについてのモデル(a model of the current environment around the agent)も勉強。
    - 例えば、include a model of the movement of dynamic objects.
    - If the agent understands the movement patterns of the dynamic object, it can create more effective plans through the environment.

Reinforcement Learning Issues
- Many simulation environments used to learn the policies required for autonomous driving are overly simplified.
  - And due to their simplicity, the policies learned may not be transferable to real world environments.
  - Overly realistic simulators lead to the issue of severe computational requirements.
    - これはあんまり課題じゃないでしょう。ふさわしいでしょう。
- Safety.（大事）
  - While there are techniques within reinforcement learning that attempt to ensure safety constraints along the trajectories created by the reinforcement learner.
  - There is still no way to perform rigorous safety assessment of a learned system, as they are mostly black boxes in terms of the way in which decisions are made.

Machine Learning（大事）
- Many of them try to learn from human driving actions.
- Inverse Reinforcement Learning.
  - Rather than trying to obtain a policy given a reward function, the approach is to use human driving data as the policy, and attempt to learn the reward function used by humans.
  - Once the reward function is learned, the algorithm can then execute driving maneuvers similarly to a human driver.
- End-to-End Approaches.
  - Input: raw sensor data.
  - Output: throttle, brake, and steering commands.
  - By learning from human driving commands in an imitation learning approach.
  - つまり、出力は人間運転手がinputの環境を見て決めたthrottle, brake, steering。
  - Nvidia.

- Behavior Planning remains one of the toughest bottlenecks in achieving real world level 5 autonomy.

論文：[2014] A Behavioral Planning Framework for Autonomous Driving.
https://ri.cmu.edu/pub_files/2014/6/IV2014-Junqing-Final.pdf