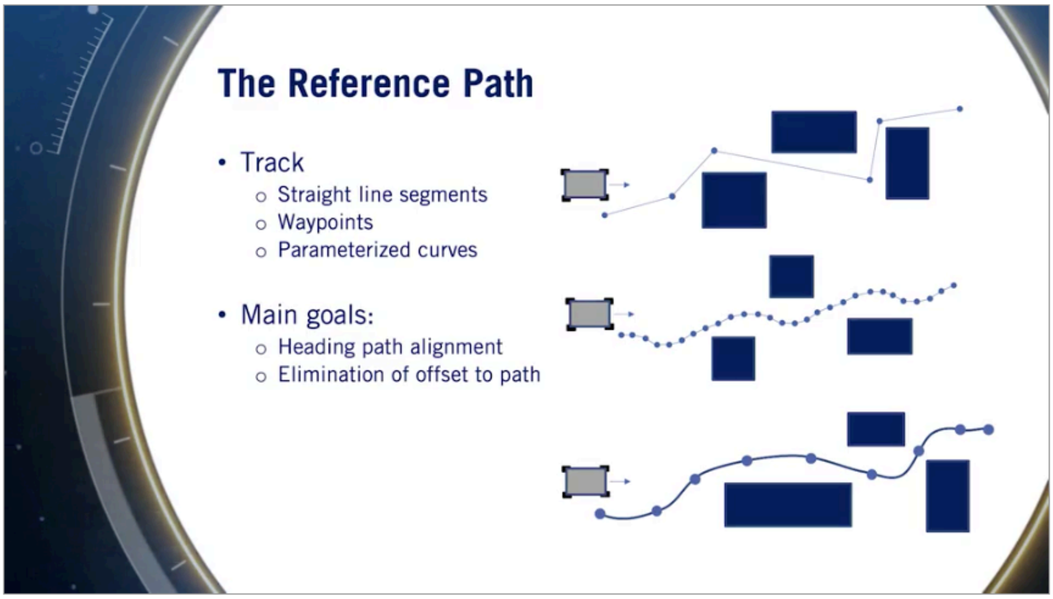
Lesson 1: Introduction to Lateral Vehicle Control

内容
- Define different types of reference path
  - The reference path is a fundamental interface between the planning system and the lateral controller.
- Compute heading and crosstrack errors relative to those reference paths

Lateral Controlのタスク
- Select a control design strategy that drives errors to zero while still satisfying steering angle limits.
- Add dynamic considerations to manage forces and moments acting on vehicle
  - Consider the dynamic limitations of vehicle and desired ride characteristics such as maximum lateral acceleration and minimum jerk.
  - Control command must be cognizant of the available tire forces and not exceed the capabilities of the vehicle when correcting for tracking errors.



The Reference Pathタイプ
- Straight line segments
  - 課題：The path includes heading discontinuities, which make precise tracking a challenge with a steered vehicle.
- Waypoints
  - tightly spaced.
  - This spacing is usually fixed in terms of distance or travel time.
  - The relative position of the waypoints can be restricted to satisfy an approximate curvature constraint.
  - Waypoints can be directly constructed from state estimates or GPS waypoints collected in earlier runs of a particular route.
- Sequence of continuous parameterized curves
  - Can be either drawn from a fixed set of motion primitives or can be identified through optimization during planning.
  - These curves provide the benefit of continuously varying motion, and can be constructed to have smooth derivatives to aid in the consistency of error and error rate calculations.
  - OpenDriveのgeometryはこのタイプらしい。しかし、geometryタグはroad reference lineの定義だけで、pathはこの3種類のどれも当たらないだろう。

- For each of these path definitions, the direction of travel along the path is also provided, which can be encoded with the point ordering or curve ordering.

Two Types of Control Design

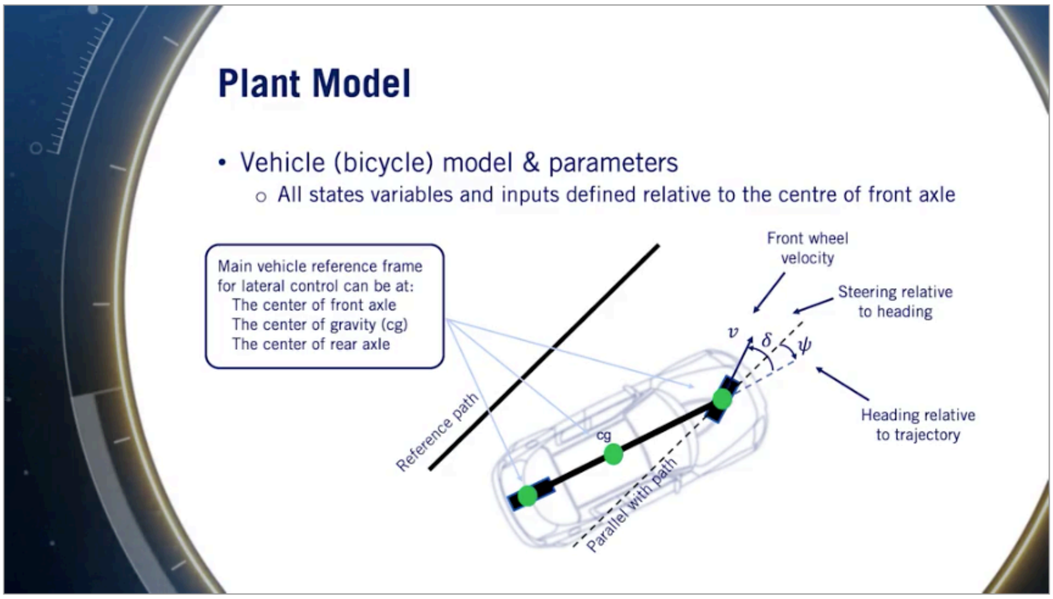
Geometric Controllers

- Rely on the geometry and coordinates of the desired path and the kinematic models of the vehicle.
- Two types: Pure pursuit (carrot following) and Stanley

Dynamic Controllers

- The most popular advanced controller in this category is the model predictive controller or MPC, which performs a finite horizon optimization to identify the control command to apply.
- MPCがよく使われる理由：its ability to handle a wide variety of constraints and to identify optimized solutions that consider more than just the current errors(??).
- Other control systems: Sliding mode, feedback linearization.

Vehicle (bicycle) model & parameters



- The bicycle model is a suitable control oriented model of a four-wheel vehicle, where the front left and right wheels are combined into a single steerable wheel, and the rear left and right wheels are combined together in a single drive wheel.

Heading Error

- Component of velocity perpendicular to trajectory divided by ICR radius
- $\dot{\psi}_{des}(t) - \dot{\psi}(t) = \dfrac{v_f(t) sin \delta(t)}{L}$ 角度ではなく、角速度だよ。
  - It is a principal measure of how well the vehicle aligned with and moving in the direction of the desired path.
  - The rate of heading error $\dot{\psi}(t)$ helps us understand how the heading error evolves over time, and can be computed from the kinematic bicycle model equations.
  - Here we present the rate of heading error relative to the front axle, as will be used in the Stanley controller.
  - For straight line segments, the desired heading rate of change is zero, and it can be removed.
  - $\dot{\psi}(t) = \dfrac{-v_f(t) sin \delta(t)}{L}$

- pathがcurveの場合ちょっと複雑：As it is not immediately clear where the reference point on the curved path should lie.
- To compute the minimum distance to a curved path defined by a spline: Robust and efficient computation of the closest point on a spline curve. http://homepage.divms.uiowa.edu/~kearney/pubs/CurvesAndSurfaces_ClosestPoint.pdf

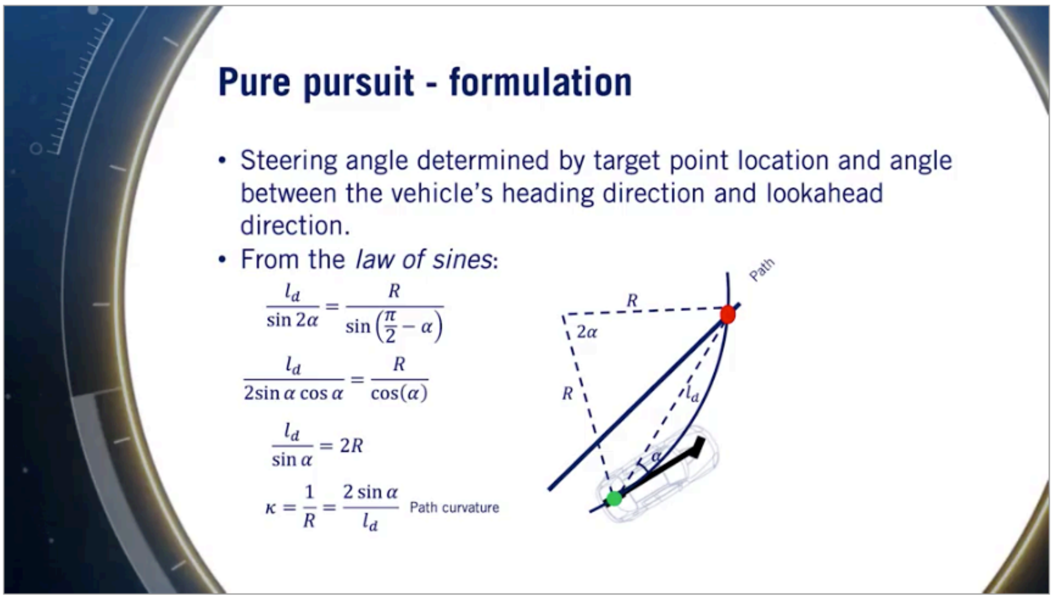
Crosstrack Error
- offset error.
- The crosstrack error is the distance between the reference point on the vehicle and the closest point on the desired path.
  - Distance from center of front axle to the closest point on path.
- The rate of change of the crosstrack error can be calculated by extracting the lateral component of the forward velocity.
  - Rate of change of crosstrack error
  - $\dot{e}(t) = v_f(t) sin(\psi(t) - \delta(t))$
  - From this equation, as the velocity increases, the crosstrack error changes more quickly, meaning that smaller steering angles are needed to correct for the same size crosstrack errors.

Lateral Control of an Autonomous Vehicle. https://repository.lboro.ac.uk/articles/Lateral_control_of_an_autonomous_vehicle/9225788/1

Lesson 2: Geometric Lateral Control - Pure Pursuit

内容
- Define the concept of a geometric path tracking controller
  - rely on kinematic vehicle model for selecting steering commands.
- Develop a pure pursuit controller for path tracking

Geometric Path Tracking Controllerの定義
- Any controller that tracks a reference path using only the geometry of the vehicle kinematics and the reference path.
- In the case of self-driving cars, a geometric path tracking controller is a type of lateral controller that ignores dynamic forces on the vehicles and assumes the no-slip condition holds at the wheels.
- One of the most popular classes of path tracking in robotics and autonomous vehicle.
  - Exploits geometric relationship between the vehicle and the path resulting in compact control law solutions to the path tracking problem.



- Use of reference point on path to measure error of the vehicle, can be ahead of the vehicle.
  - A reference point along the desired path, which can be the same reference point used to compute heading and crosstrack errors, or it can be a look-ahead point some distance in front of the vehicle along the path. Pure Pursuitはlook-ahead reference pointを使う。Stanleyはthe same reference point used to compute heading and crosstrack errorsを使う。
- 課題：its performance suffers when the vehicle motion does not match the no-slip assumption, as is the case in aggressive vehicle maneuvers with high lateral acceleration.
  - When the vehicle is operating in the linear tire region and a tire is not saturated, geometric path tracking controllers can work very well. “tire not saturated”はどういう意味?

Pure Pursuitのやり方
- The core idea is that a reference point can be placed on the path a fixed distance ahead of the vehicle, and the steering commands needed to intersect with this point using a constant steering angle can be computed.
- As the vehicle turns towards the path to follow this curve, the point continues to move forward, reducing the steering angle and gently bringing the vehicle towards the path.
- Formulation: steering angleの計算
  - In this method, the center of the rear axle is used as the reference point on the vehicle.
  - l_d : the line that connects the center of the rear axle to the target reference point as a line of fixed distance l_d (look ahead distance).
  - α : the angle between the vehicle’s body heading and the look-ahead line.
  - arc: this arc is the part of the ICR circle that covers the angle of 2α.
  - $\dfrac{l_d}{sin2\alpha} = \dfrac{R}{sin(\frac{\pi}{2} - \alpha)}$
  - つまり$\dfrac{l_d}{2sin\alpha cos\alpha} = \dfrac{R}{cos\alpha}$
  - つまり$\dfrac{l_d}{sin\alpha} = 2R$
  - path curvature: $K = \dfrac{1}{R} = \dfrac{2sin\alpha}{l_d}$
  - steering angle needed: $tan\delta = \dfrac{L}{R}$
    - $\delta = tan^{-1}\dfrac{L}{R} = tan^{-1}KL = tan^{-1}(\dfrac{2Lsin\alpha}{l_d})$
    - Recall from Module 4 Lesson 2 The Kinematic Bicycle Model, an intersection of two lines defines the instantaneous center of rotation of a bicycle model
      - Perpendicular to the center of the vehicle rear axle.
      - Perpendicular to the center of the vehicle front axle.
- Formulation: crosstrack errorの計算
  - e : the distance between the heading vector and the target point.
  - $sin\alpha = \dfrac{e}{l_d}$
  - combined with $K = \dfrac{2sin\alpha}{l_d}$
    - $K = \dfrac{2}{l_d^2}e$: the curvature of the path created by the pure pursuit controller is proportional to the crosstrack error at the look-ahead reference point.
    - As the error increases, so does the curvature, bringing the vehicle back to the path more aggressively.

- This equation demonstrates that the pure pursuit controller works in a manner similar to proportional control to correct crosstrack error using path curvature as the output of the controller.
- The proportional gain $\frac{2}{l_d^2}$ can (must?) be tuned at different speeds (the l_d being assigned as a function of vehicle speed)
  - 弱点
    - fixed l_dだと、the selected steering angle would be the same regardless of whether the vehicle is going 10 km/h or 100km/h, leading to very different lateral accelerations.
    - A controller tuned for high-speed would be far too sluggish at low speed, and one tuned for low speed would be dangerously aggressive at high speeds.
  - 解決
    - vary the look-ahead distance l_d based on the speed of the vehicle.
    - define the look-ahead distance to increase proportional to the vehicle forward speed.
    - $l_d = K_{dd}v_f$
    - combined with $\delta = tan^{-1}(\frac{2Lsin\alpha}{l_d})$
      - $\delta = tan^{-1}(\frac{2Lsin\alpha}{K_{dd}v_f})$, $K = \frac{2}{l_d^2}e$
      - The controller selects the steering angle that will form an arc to the look-ahead reference point, and adjusts this look-ahead point to be further away the faster the vehicle is traveling.

Automatic Steering Methods for Autonomous Automobile Path Tracking. CMUだから重要
https://www.ri.cmu.edu/pub_files/2009/2/Automatic_Steering_Methods_for_Autonomous_Automobile_Path_Tracking.pdf

Lesson 3: Geometric Lateral Control - Stanley

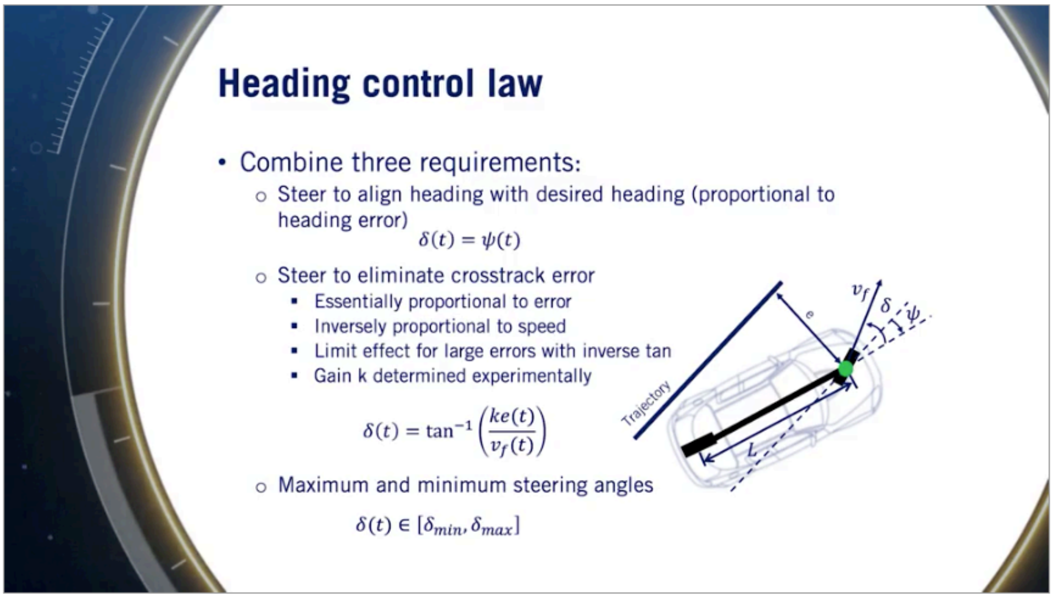
This controller was used by the Stanford racing team to win the second DARPA Grand Challenge event.
Aeryon Labs, Clearpath Robotics, and Renesas Electronics America
Waterloo Robotics Team
Waterloo Autonomous Vehicles Laboratory
Aeronautics and Astronautics

内容
- Derive the Stanley path tracking controller.
- Analyze the evolution of its steering commands for small and large heading and crosstrack errors.
- Evaluate performance in the form of convergence to the desired path from arbitrary starting points.

Stanley Controller Approach: Pure Pursuitと比べる変更点
1. Use the center of the front axle as a reference point.
   - The main concept that went into the creation of the Stanley controller was that a change in the reference position could lead to different, possibly more desirable properties of the control. vehicle reference point.
   - Dr. Hoffman was seeking a control law with global convergence to the path and predictable decay of the errors that would be independent of vehicle speed. “global convergence”はどういう意味?
2. Consider both heading alignment and crosstrack error without a look-ahead distance, but directly at the reference point.

- つまりLook at both the error in heading and the error in position relative to the closest point on the path.

3. Cap its outputs to fall within the limits of the maximum steering angle.



Heading Control Law

1. Steer to align heading with desired heading (proportional to heading error): $\delta(t) = \psi(t)$
2. Steer to eliminate crosstrack error: $\delta(t) = tan^{-1}(\dfrac{ke(t)}{v_f(t)})$
    1. Essentially proportional to error
    2. Inversely proportional to speed
    3. Limit effect for large errors with inverse tan
    4. Gain k determined experimentally
3. Maximum and minimum steering angles: $\delta(t) \in [\delta_{min}, \delta_{max}]$

- The independent penalization of heading and crosstrack errors and the elimination of the look-ahead distance make this a different approach from pure pursuit.

4. Stanley Control Law: $\delta(t) = \psi(t) + tan^{-1}(\dfrac{ke(t)}{v_f(t)})$, $\delta(t) \in [\delta_{min}, \delta_{max}]$

- For large heading error, steer in opposite direction
    - The larger the heading error, the larger the steering correction
    - Fixed at limit beyond maximum steering angle, assuming no crosstrack error.
- For large positive crosstrack error
    - $tan^{-1}(\dfrac{ke(t)}{v_f(t)}) \approx \dfrac{\pi}{2} \rightarrow \delta(t) \approx \psi(t) + \dfrac{\pi}{2}$
        - This large value clamps the steering command to the maximum and the vehicle turns towards the path.
        - The effect of this term is to increase the heading error in the opposite direction, and so the steering command will drop to 0 once the heading error reaches $-\dfrac{\pi}{2}$.
            - The vehicle then proceeds straight to the path until the crosstrack error decreases.
            - At this point, the heading term starts correcting the alignment with the path again and ultimately, the vehicle starts to track the path more closely. つまりcrosstrackエラーが大きすぎると、headingの補正は全然無くなって、車が最大steering angleで一生懸命path

に戻りたい！その後、crosstrackエラーが下がって、そんなに一生懸命戻る必要がない時、またheadingの補正が考えられる。

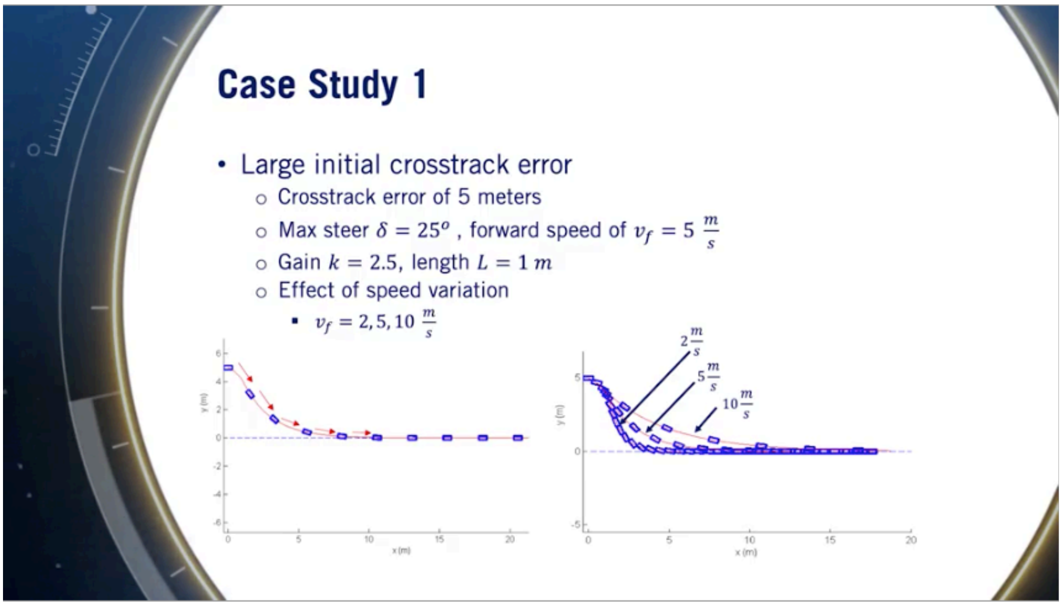
  - “large positive crosstrack error”, つまり自車が今pathの左側にいる。つまり$\frac{\pi}{2}$は時計回り?
  - As heading changes due to steering angle, the heading correction counteracts the crosstrack correction, and drives the steering angle back to zero.
    - The vehicle approaches the path, crosstrack error drops, and steering command starts to correct heading alignment.
- Convergence Characteristics.

Error Dynamics
- The error dynamics when not at maximum steering angle are:

$$\dot{e}(t) = -v_f(t) sin(\psi(t) - \delta(t)) = -v_f(t) sin(tan^{-1}(\frac{ke(t)}{v_f(t)}))$$

$$= \frac{-ke(t)}{\sqrt{1 + (\frac{ke(t)}{v_f})^2}}$$

  - 簡単な説明：$\tan\theta = \frac{x}{y} = \frac{\frac{ke(t)}{v_f(t)}}{1}, \sin\theta = \frac{x}{\sqrt{x^2+y^2}} = \frac{\frac{ke(t)}{v_f(t)}}{\sqrt{1+(\frac{ke(t)}{v_f(t)})^2}}$
- For small crosstrack errors, leads to exponential decay characteristics
  - We can simplify the denominator of this expression by assuming the quadratic term is negligible: $\dot{e}(t) \approx -ke(t)$
  - The crosstrack error evolution follows a first-order differential equation, and the solution for this ODE is an exponential.
  - Since k is positive, the error decays exponentially to 0.
  - The most interesting aspect of this investigation is that the decay rate is completely independent of the speed.
    - So faster vehicles will travel farther while converging to the path, but will converge to the path at the same time as slower moving vehicles.



Case Study 1: Large Initial Crosstrack Error

Stanley Controllerはまさに障害物回避で使うべきControllerだ！

解説

- Simulation図 1
  - The large initial error leads to a large steering command that quickly turns the vehicle towards the path.
  - The heading error and crosstrack error terms then reach an equilibrium, and the vehicle continues in a straight line towards the path.
  - As the crosstrack error decreases, the exponential decay to the path becomes visible.
- Simulation図 2
  - Run the same simulation at different forward velocities.
  - In all cases, the turn towards the path, straight line progress and then exponential decay to the path are visible.
  - The higher the speed, the further the car travels before reaching the path.
  - But the final convergence for small crosstrack errors takes the same amount of time in each case.

Case Study 2: Large Initial Heading Error

解説：

- First, the steering command is up against its limit as the heading error is corrected.
- Then as the crosstrack error starts to grow, the steering commands continue to correct the heading of the car beyond the alignment with the path.
- Finally, the car enters the exponential convergence segment as before.
- In fact, it comes with a global stability proof, meaning that no matter what the initial conditions, the controller will guide the car back to its path.
- In practice however, the Stanley controller is still a geometric path tracking controller, and as such does not consider many different aspects of real self-driving car.
  - For example, it does not consider noisy measurements, actuator dynamics or tire force effects, all of which can cause undesirable ride characteristics during maneuvers.

Adjustment: Low Speed Operation

背景：During low-speed operation, the pure pursuit and Stanley controllers can behave quite aggressively when confronted with noisy velocity estimates.

- Inverse speed can cause numerical instability.
  - Errors in low speed estimates tend to get amplified in the steering command.
  - This leads to wild swings in the steering wheel, which is not desirable for rider comfort.
- Add softening constant to controller: $\delta(t) = \psi(t) + tan^{-1}(\dfrac{ke(t)}{k_s + v_f(t)})$
  - Ensures the denominator always has a minimum value.
  - This softening constant can be tuned in the field.

Adjustment: Extra Damping on Heading

背景：At higher speeds, the issue that steering commands need to vary slowly to ensure lateral forces are not excessive.

- Even with this scaling on speeds, Stanley's response was overly aggressive at high speeds, and so a damping term on heading rate was also added.
- This essentially converts the heading error control portion to a PD controller, and the same idea can be applied to the pure pursuit control of curvature as well.

Adjustment: Steer into Constant Radius Curves

背景：For curved paths with high curvature, the controller fails to track them well as the reference dynamics were not considered in the derivation of the geometric controllers.

- Improves tracking on curves by adding a feedforward term on heading. つまりpathのheading。

結論：

- With these modifications, the Stanley controller becomes a useful tool for moderate driving tasks as long as the vehicle avoids exiting the linear tire region.

- Define paths that are safe to track in the 4th course.
- Add further enhancements that improve the controllers' real-world performance.

Stanley Controller論文：Autonomous Automobile Trajectory Tracking for Off-Road Driving: Controller Design, Experimental Validation and Racing:
http://ai.stanford.edu/~gabeh/papers/hoffmann_stanley_control07.pdf

Lesson 4: Advanced Steering Control - MPC, Model Predictive Control

内容
- Describe the MPC architecture and the concept of receding horizon control.
- Formulate an MPC optimization problem for both linear and nonlinear models.
- Apply MPC to joint longitudinal and lateral vehicle control.

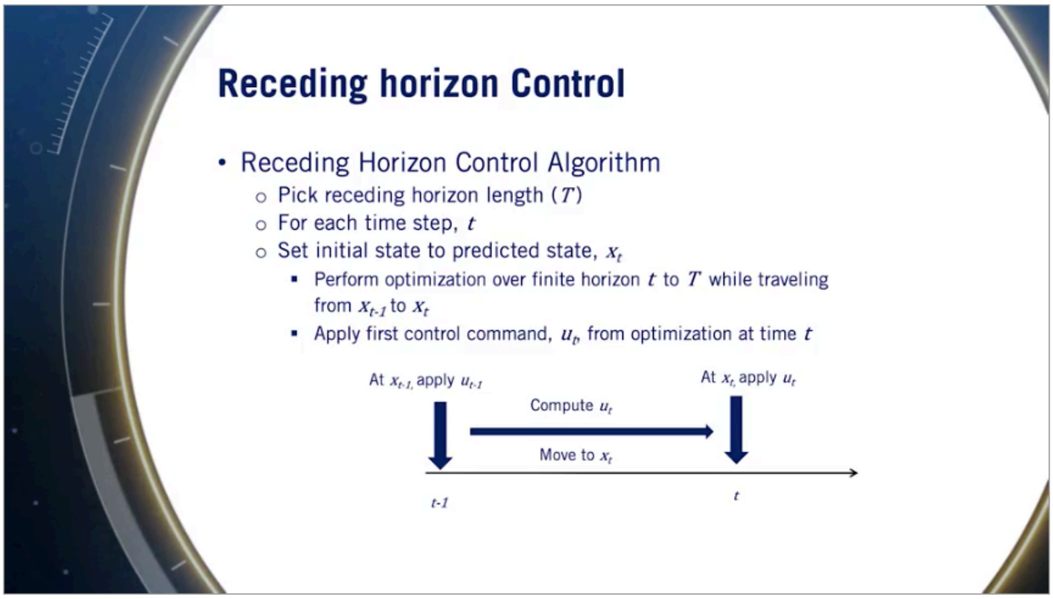
Model Predictive Controlの定義
- Numerically solving an optimization problem at each time step.
  - Because solving an optimization problem at each time step can take time, MPC was originally applied to slow processes such as industrial chemical processing.
  - However, the ever-improving performance of today's computing hardware has made MPC a viable approach even on embedded hardware.
  - More and more automotive applications are turning to MPC as a way to improve performance and expand operating range for a suite of different embedded controllers, from traction control, and stability control, to emission reduction, and idle speed control.
  - Longitudinal and lateral control for autonomous vehicles is another extremely suitable application for MPC.
- Receding horizon approach.
  - Model Predictive Control is often interchangeably referred to as Receding Horizon Control, since the controller generates an actuator signal based on a fixed finite length horizon at each time-step which receives as time moves forward.

Advantages of MPC
- Straightforward formulation.
- Explicitly handles constraints.
  - Requiring the definition of an objective function and relevant constraints that are then optimized using well-established solvers.
  - The states and control signals can be constrained to stay within safe operating bounds and controls can be selected to maximize multiple objectives simultaneously.
  - Both hard constraints and soft penalties can be employed, leading to a rich set of solutions for constrained control problems.
  - As many automotive subsystems have rigid actuator constraints and diverse performance objectives, MPC has emerged as a major tool for vehicle control.
- Applicable to linear or nonlinear models.
  - Meaning that we can use the same approach even as our models change or improve over time.

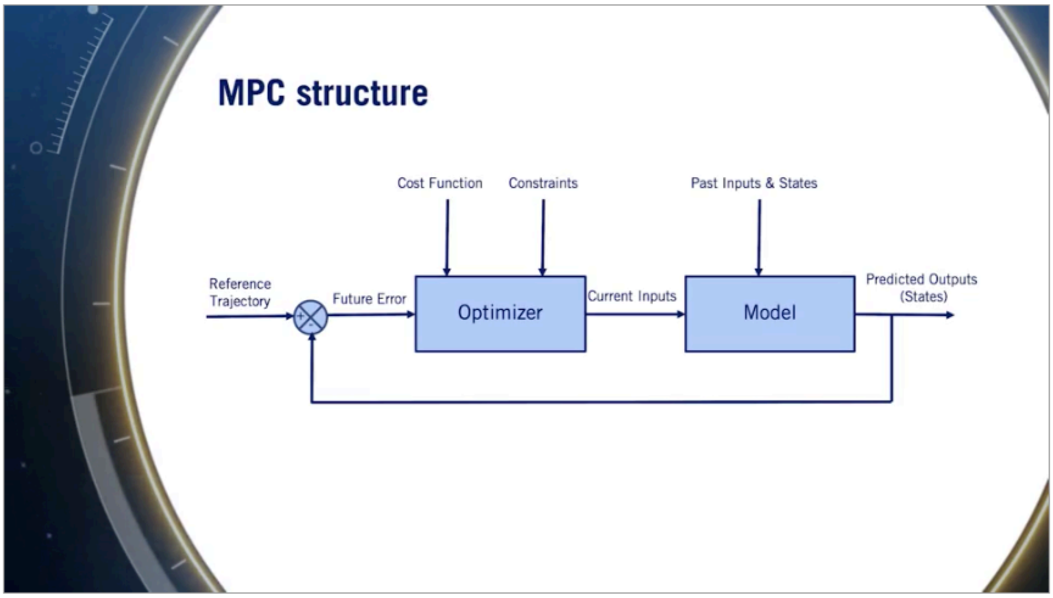
Disadvantages of MPC
- Computationally expensive.
  - It is certainly possible to create optimization formulations that are too expensive to compute at the high update rates required for smooth vehicle control.
  - Careful implementation is needed to avoid overloading processors.

Receding Horizon Control
タスク：
Receding Horizon Control solves a fixed size optimization at each time-step, which identifies optimal control inputs to apply from the current time to the end of the horizon based on the objectives' constraints and current state of the vehicle. "horizon"はどういう意味?

課題：
Because optimization can take some amount of time, the state of the vehicle when starting the optimization, will be different from the state of vehicle when completing the optimization.
対策：
As a result, must use a predicted state in the optimization for the time at which the control input will actually be applied.



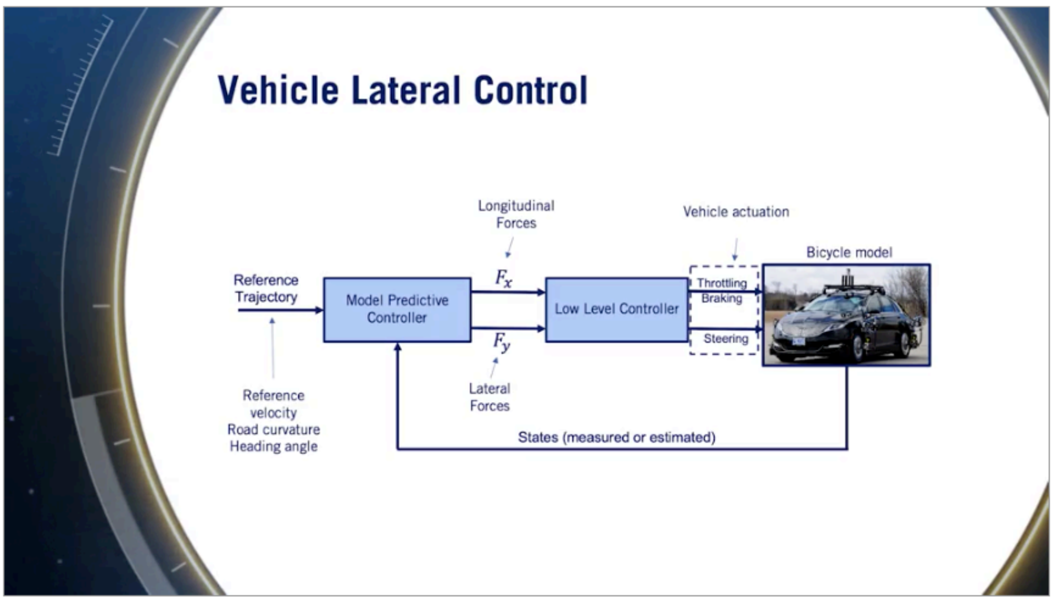
Receding Horizon Control Algorithm
- Perform optimization over finite horizon t to T while traveling from x_{t-1} to x_t. Tの意味はまだ分かっていない。
  - Using the control input identified in the previous optimization.
  - Although we won't exactly arrive at the predicted state at time t due to disturbances, we do expect to be reasonably close if the time interval is short.



- ~~つまりu_t (optimal control input)を計算するためのインプットは：u_{t-1}とx_t。しかし、x_tはまだ分からないので、predicted x_tを使う。~~
- 違う、u_tの計算はu_{t-1}がいらない。x_tの予測中のみu_{t-1}はいる。
- u_tの計算はpredicted x_tとreference trajectoryのerrorから計算する！（Receding Horizon Controlの計算プロセスはMPC Structureの図から完全に理解できる）
- また、x_{t-1}からx_tまで移動中、u_tを計算する。そうすると、車がx_tに到達する時、u_tも計算済み。実際に到達したx_tとpredicted x_tは完全に一緒ではないが、大体一緒。
- tに、車がu_tを使って、自分を制御する。繰り返し。
- 1点質問x_tを予測する時、x_{t-1}は真値? x_tの予測は、x_{t-1}とu_{t-1}があれば、OK。

MPC Structure
- The model then outputs predicted states at the next time-step, which are compared to the reference trajectory and passed into the optimizer as the future or predicted error.
- The optimizer also receives updated constraints and the cost function to use, which can be fixed in advance or varied based on changing operating modes.

Linear MPC formulation
- Linear time-invariant discrete time model: $x_{t+1} = Ax_t + Bu_t$
- MPC seeks to find control policy U: $U = \{u_{t|t}, u_{t+1|t}, u_{t+2|t}, \ldots\}$
- Object function - regulation: $J(x(t), U) = \sum_{j=t}^{t+T-1} x_{j|t}^T Q x_{j|t} + u_{j|t}^T R u_{j|t}$
  - If all the states are to be driven to zero（つまり最適化は最小化）, the objective function or cost function when we minimize, can be defined as: with quadratic error on both deviations of the state from zero and on non-zero control inputs（0からの偏差を罰する）.
  - This is similar to the optimization problem of optimal control theory and trades off control performance and input aggressiveness.
  - the matrices Q and R can be selected to achieve a particular type of response.
- Object function - tracking: $\delta_{x_{j|t}} = x_{j|t,des} - x_{j|t}$, $J(x(t), U) = \sum_{j=t}^{t+T-1} \delta_{x_{j|t}}^T Q \delta_{x_{j|t}} + u_{j|t}^T R u_{j|t}$
- This is a famous optimization formulation and has a closed form solution, the Linear Quadratic Regulator or LQR.
  - The LQR solution defines a control gain matrix K, which can be computed from the A and B matrices of the state-space model and the Q and R matrices of the cost function.



- Full state feedback: $u_t = -Kx_t$

(Non)Linear MPC formulation
- Contained (non)linear finite horizon discrete time case

$$min J(x(t), U) = \sum_{j=t}^{t+T} C(x_{j|t}, u_{j|t})$$

$$\begin{aligned}
s.t.\ \ & x_{j+1|t} = f(x_{j|t}, u_{j|t}), && t \le j \le t+T-1 : \text{non-linear dynamic models of motion} \\
& x_{min} \le x_{j+1|t} \le x_{max}, && t \le j \le t+T-1 : \text{state bounds} \\
& u_{min} \le u_{j|t} \le u_{max}, && t \le j \le t+T-1 : \text{input bounds} \\
& g(x_{j|t}, u_{j|t}) \le 0, && t \le j \le t+T-1 : \text{inequality constraints} \\
& h(x_{j|t}, u_{j|t}) = 0, && t \le j \le t+T-1 : \text{equality contraints}
\end{aligned}$$

- No closed form solution, must be solved numerically.
  - Even the kinematic bicycle model falls into this category.
  - So almost all MPC controllers for autonomous driving will be solved numerically.

Vehicle Lateral Control （MPCを使うlateral control）
解説
- Include the conversion from the tire forces (F_x, F_y) to throttle, brake, and steering commands as a low level controller inside the loop.
- The inputs to the MPC block are the reference trajectory, which include the reference path and velocity, as well as the vehicle states at each time step.
- The output of the MPC block are the lateral and longitudinal forces needed to follow the desired trajectory.
- Finally, the actuation signals are applied to the vehicle at each time-step, and a new vehicle state is achieved closing the feedback loop.

Model Predictive Controller（流れ）
- Cost Function - Minimize
  - Deviation from desired trajectory
  - Minimization of control command magnitude
- Constraints - Subject to
  - Longitudinal and lateral dynamic models
  - Tire force limits
- Can incorporate low level controller, adding constraints for
  - Engine map
  - Full dynamic vehicle model
  - Actuator models
  - Tire force models

Predictive Active Steering Control for Autonomous Vehicle Systems:
https://ieeexplore.ieee.org/document/4162483