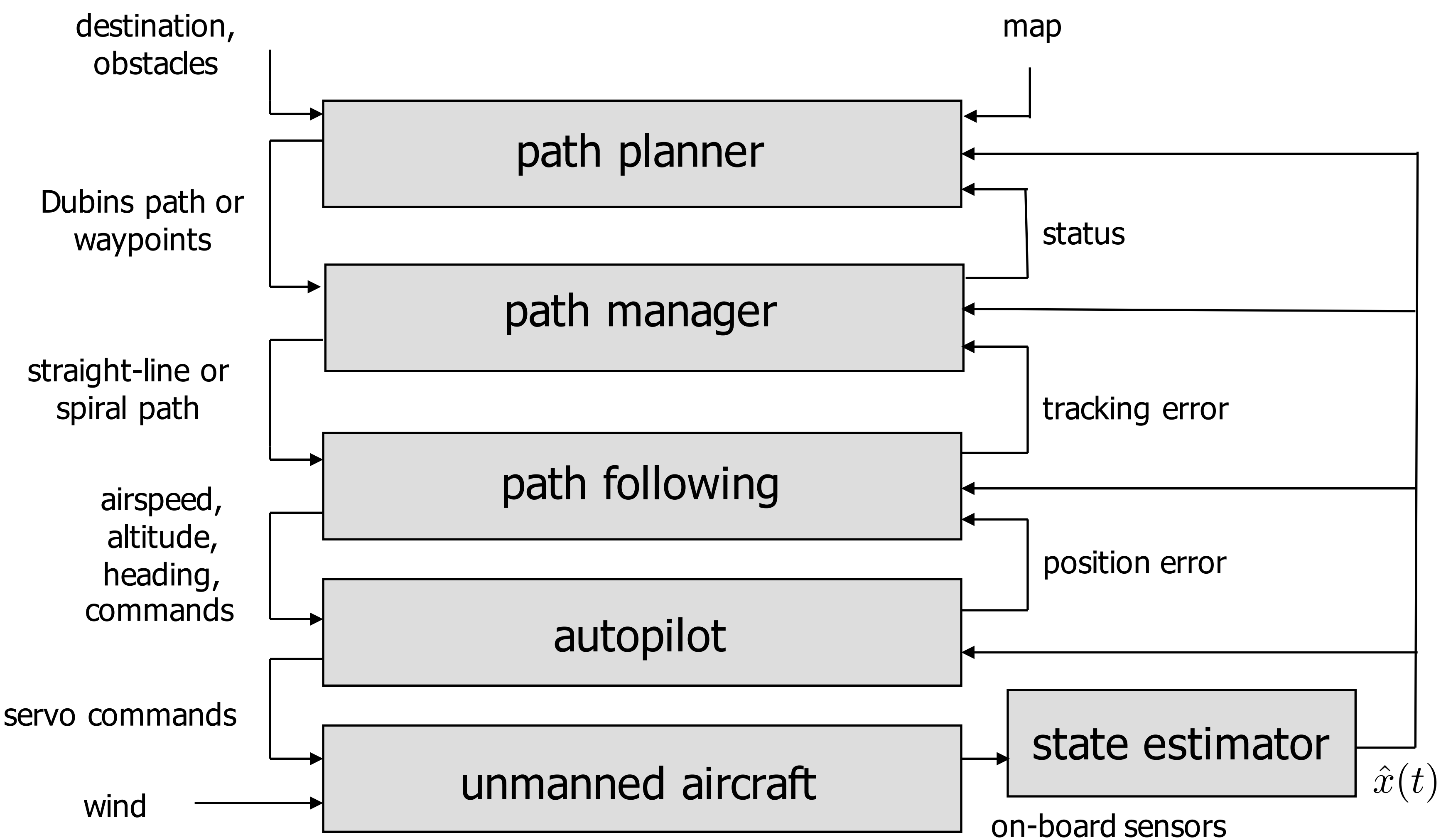
Autopilot design for small fixed wing aerial vehicles

Randy Beard
Brigham Young University

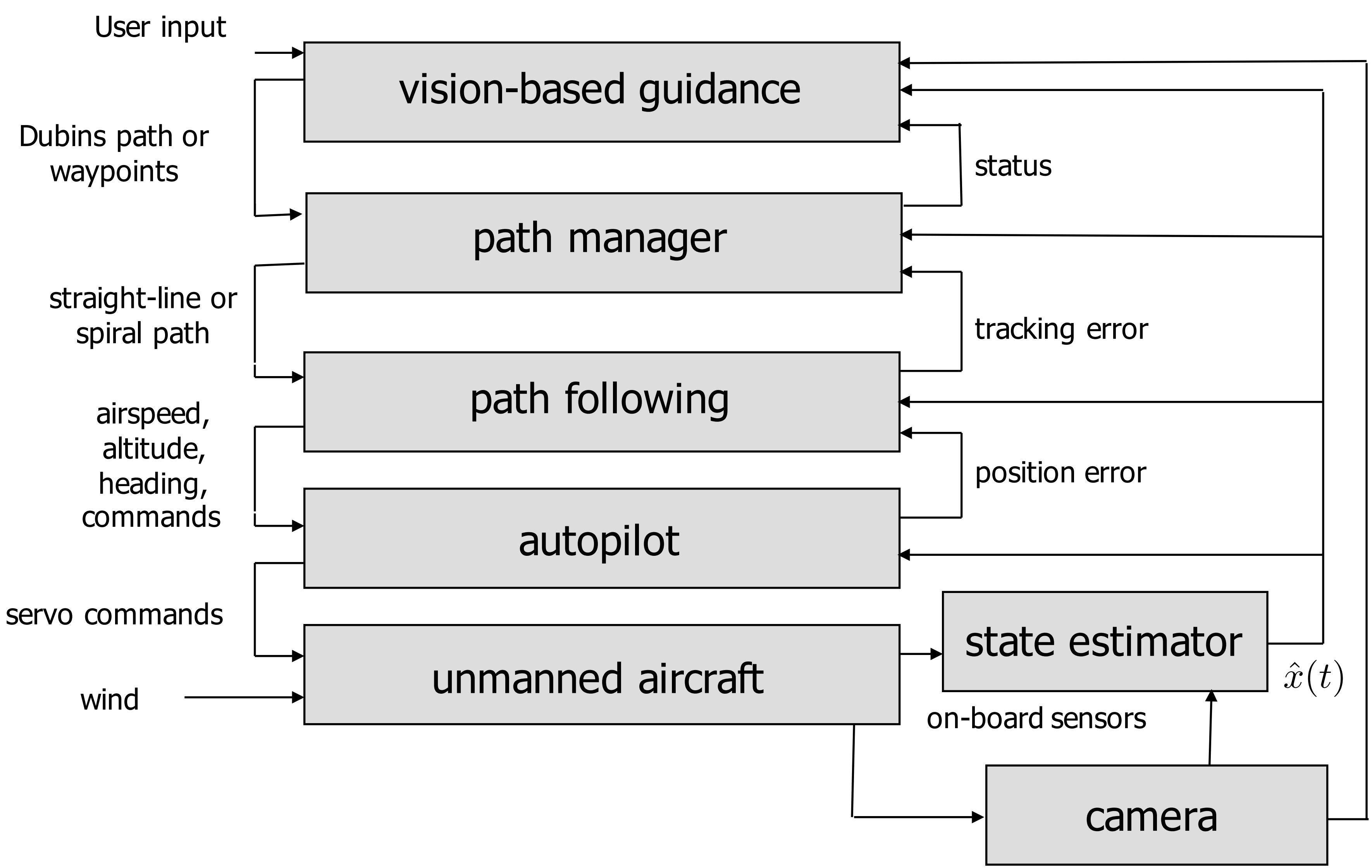
Outline

- Control architecture
- Low level autopilot loops
- Path following
- Dubins airplane paths and path management

Control Architecture



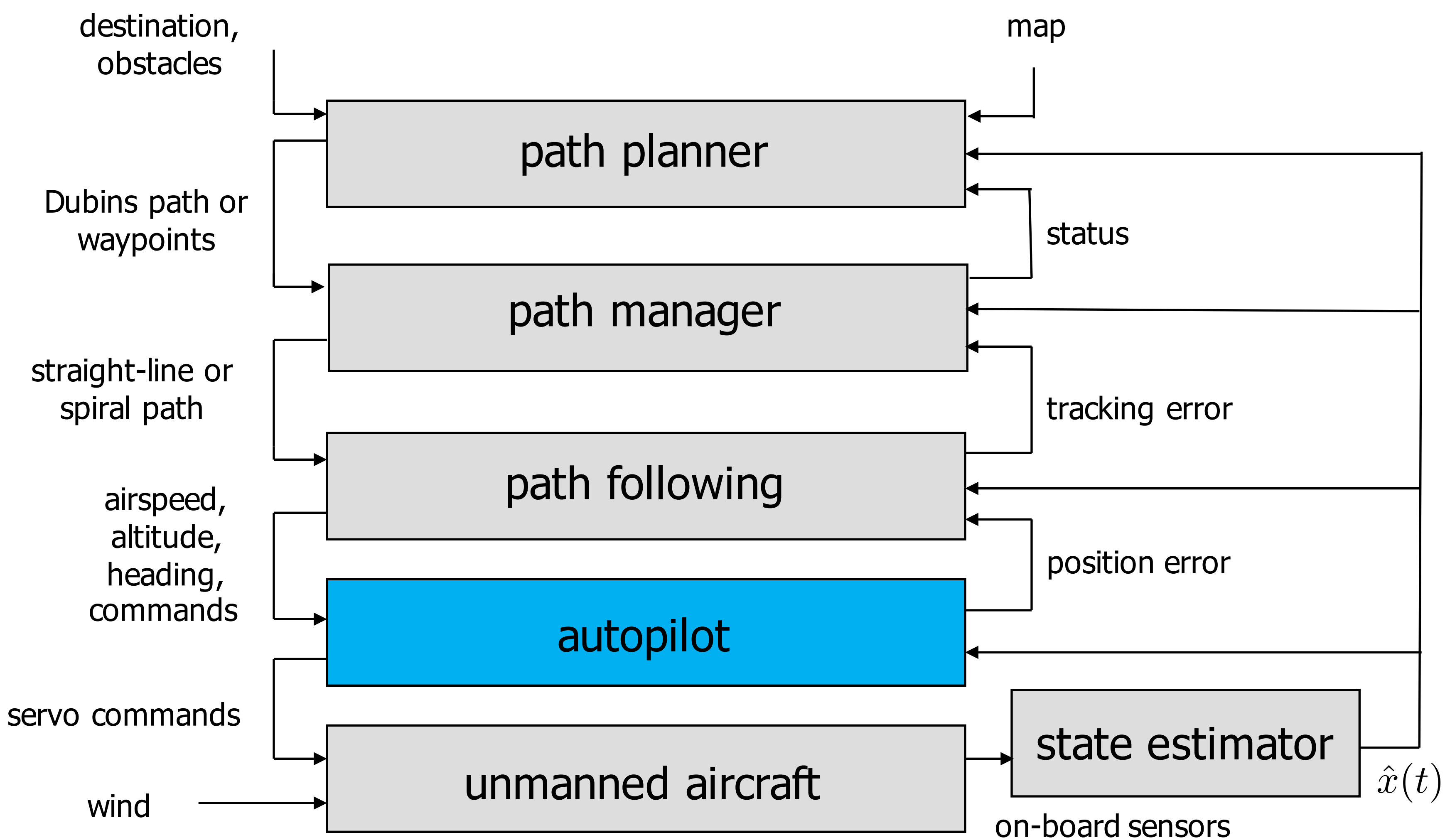
Control Architecture (2)



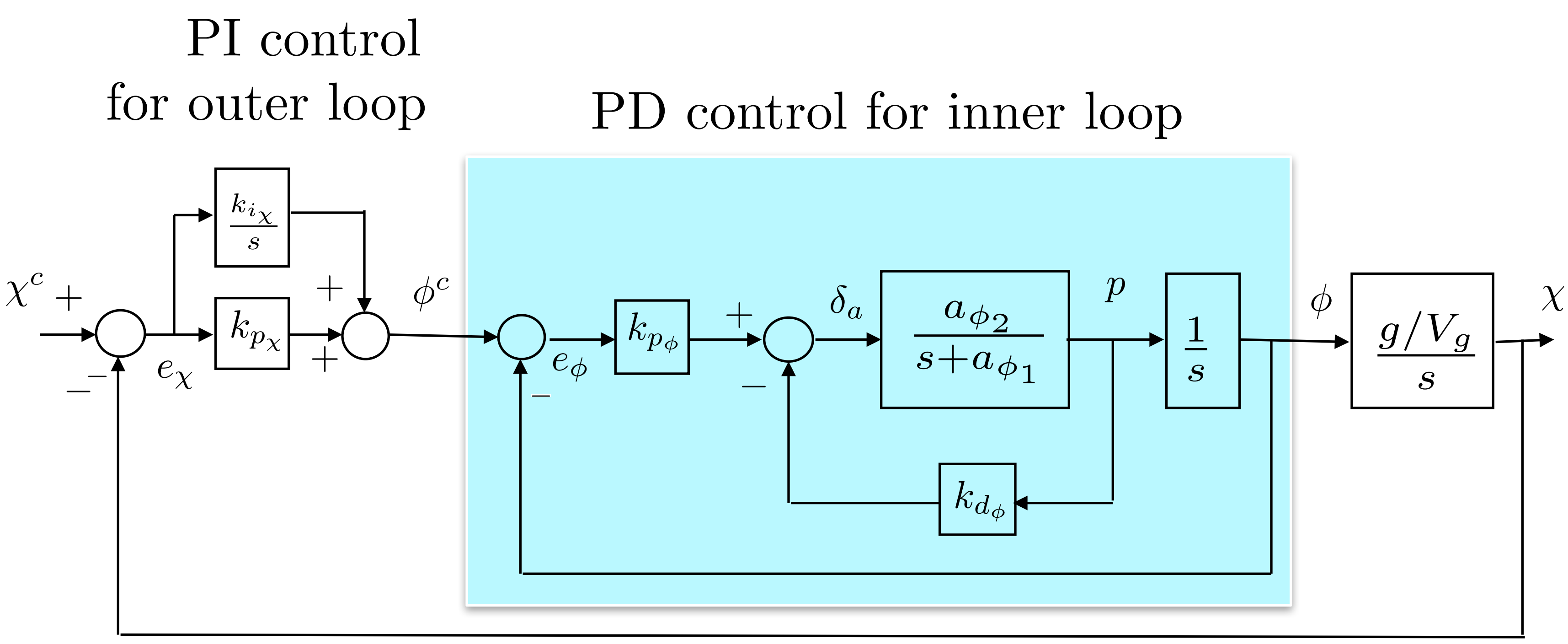
Outline

- Control architecture
- Low level autopilot loops
- Path following
- Dubins airplane paths and path management

Control Architecture

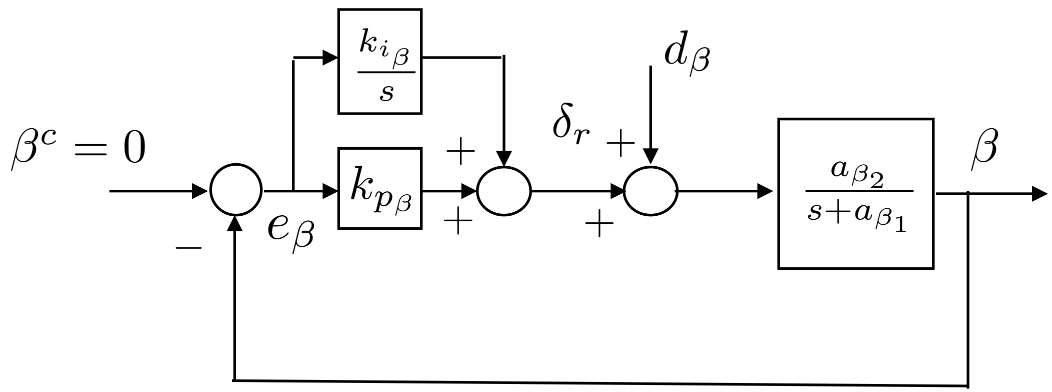


Lateral Control loop



Control course angle χ.

Sideslip Hold



If the airframe has a rudder,
then the rudder is used to zero the side slip angle β.

Requires direct measurement or estimate of β.

Lateral Autopilot – In Flight Tuning

If model is not known, and autopilot must be tuned in flight, then the following gains are tuned one at a time, in this specific order:

Inner Loop (roll attitude hold)

- k_{d_ϕ} - Increase k_{d_ϕ} until onset of instability, and then back off by 20%.
- k_{p_ϕ} - Tune k_{p_ϕ} to get acceptable step response.

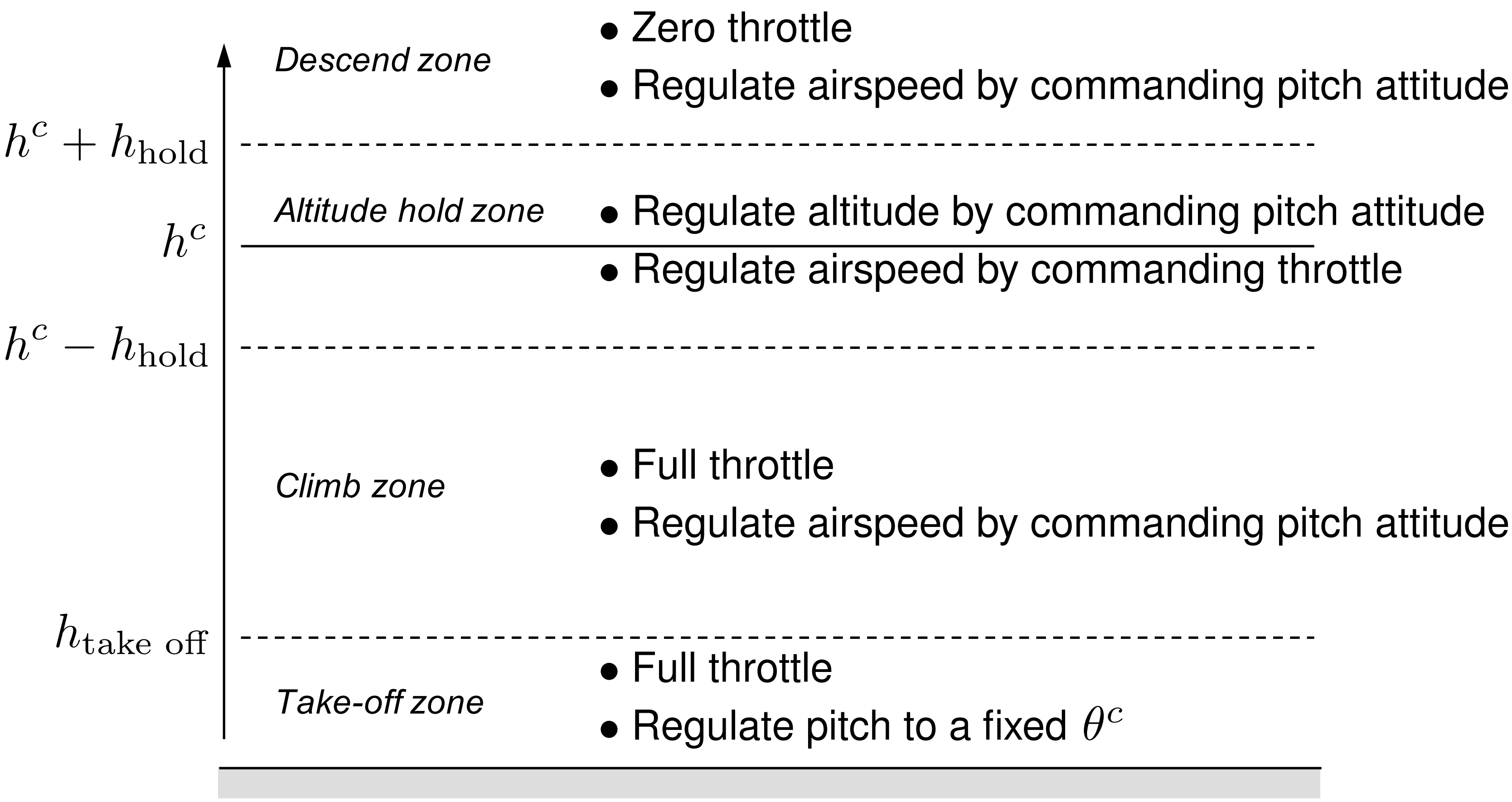
Outer Loop (course hold)

- k_{p_χ} - Tune k_{p_χ} to get acceptable step response.
- k_{i_χ} - Tune k_{i_χ} to remove steady state error.

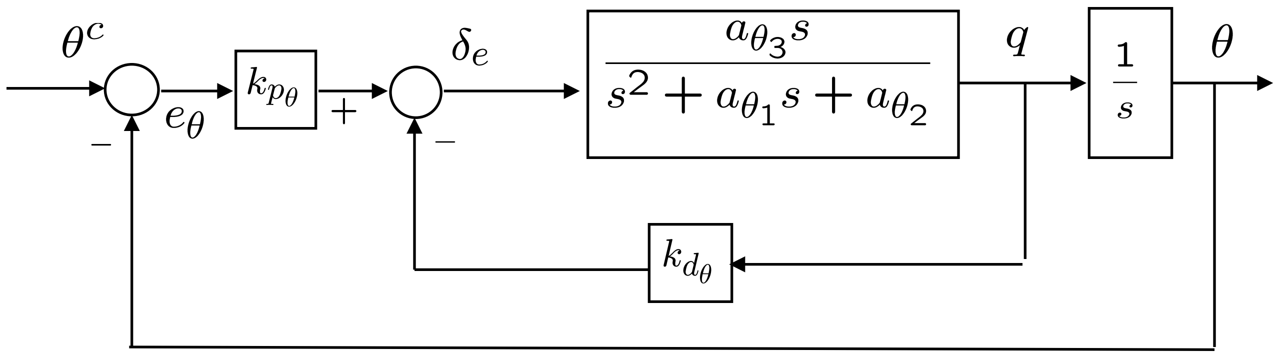
Sideslip hold (if rudder is available)

- k_{p_β} - Tune k_{p_β} to get acceptable step response.
- k_{i_β} - Tune k_{i_β} to remove steady state error.

Longitudinal Autopilot – Traditional

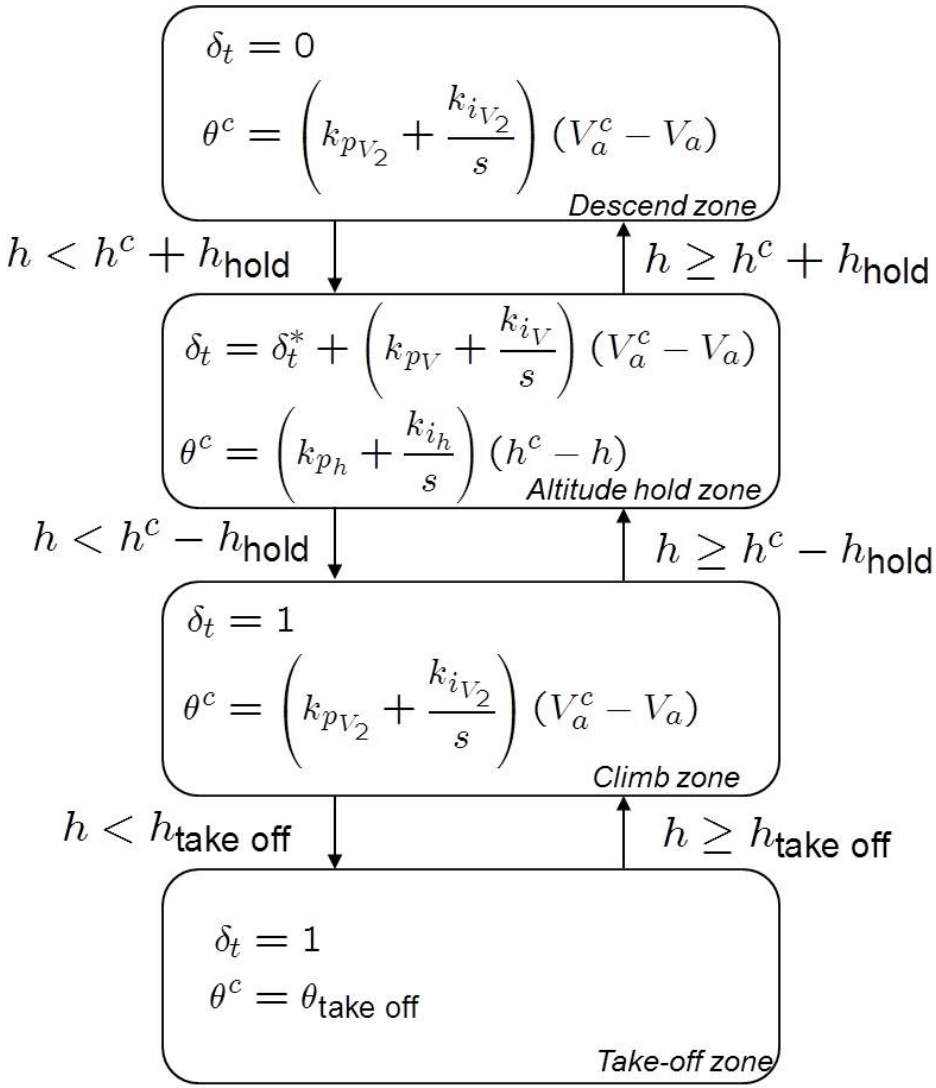


Pitch Attitude Hold



- Pitch attitude hold used for most longitudinal modes.
- Note that DC gain is not unity.
- Adding an integrator on this inner loop is not a good strategy.

Longitudinal Autopilot – Traditional



Longitudinal Autopilot – In Flight Tuning

If model is not known, and autopilot must be tuned in flight, then the following gains are tuned one at a time, in this specific order:

Inner Loop (pitch attitude hold)

- k_{d_θ} - Increase k_{d_θ} until onset of instability, and then back off by 20%.
- k_{p_θ} - Tune k_{p_θ} to get acceptable step response.

Altitude Hold Outer Loop

- k_{p_h} - Tune k_{p_h} to get acceptable step response.
- k_{i_h} - Tune k_{i_h} to remove steady state error.

Airspeed Hold Outer Loop

- $k_{p_{V_2}}$ - Tune $k_{p_{V_2}}$ to get acceptable step response.
- $k_{i_{V_2}}$ - Tune $k_{i_{V_2}}$ to remove steady state error.

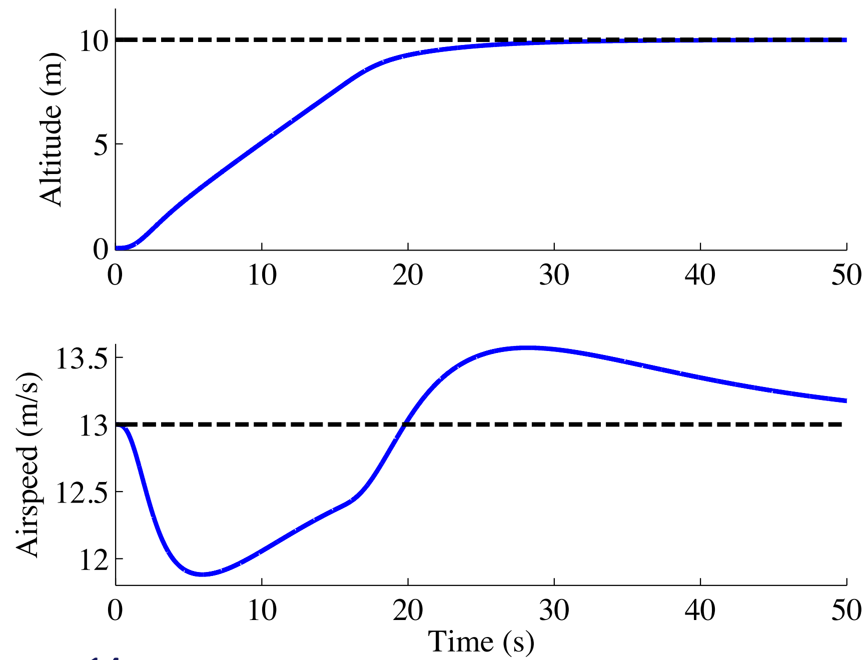
Throttle hold (inner loop)

- k_{p_V} - Tune k_{p_V} to get acceptable step response.
- k_{i_V} - Tune k_{i_V} to remove steady state error.

Airspeed and Attitude Coupling

- Traditional approach assumes longitude and lateral dynamics are decoupled
  - Altitude and airspeed does not influence course and position
  - Valid assumption

- Airspeed and altitude are decoupled
  - Invalid Assumption

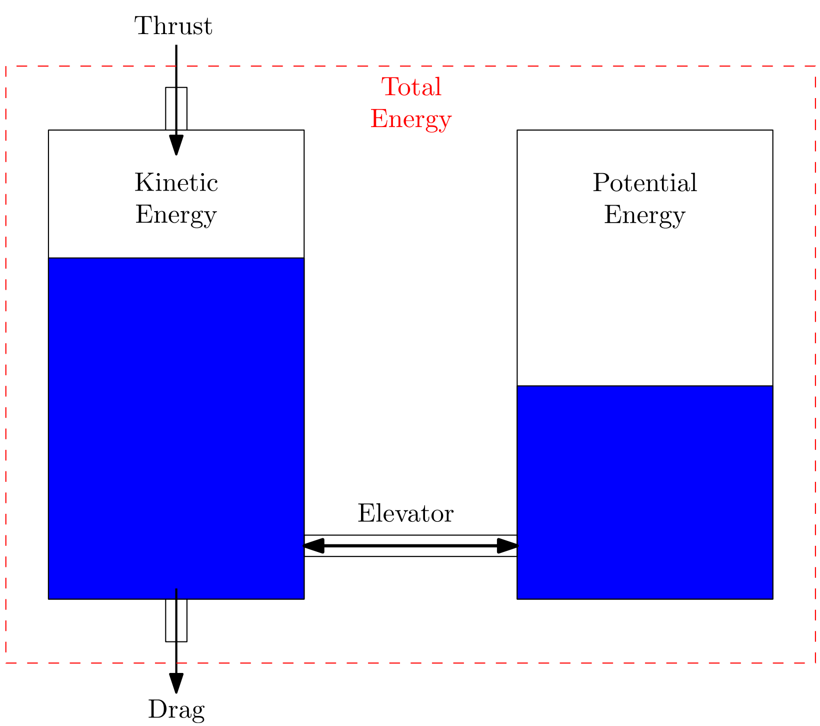


Airspeed and Attitude Coupling

		Assumed Response		**True Response**	
Pitch	Thrust	Altitude	Airspeed	Altitude	Airspeed
$\Uparrow$	-	$\Uparrow$	-	$\Uparrow$	$\Downarrow$
$\Downarrow$	-	$\Downarrow$	-	$\Downarrow$	$\Uparrow$
-	$\Uparrow$	-	$\Uparrow$	$\Uparrow$	$\Uparrow$
-	$\Downarrow$	-	$\Downarrow$	$\Downarrow$	$\Downarrow$

Total Energy Control System

- Developed in the 1980's by Antonius Lambregts
- Based on energy manipulation techniques from the 1950's
- Control the energy of the system instead of the altitude and airspeed



Total Energy Control System

- Kinetic Energy: $E_K \triangleq \frac{1}{2} m V_a^2$
- Potential Energy: $E_P \triangleq mgh$
- Total Energy: $E_T \triangleq E_P + E_K$
- Energy Difference: $E_D \triangleq E_P - E_K$

Total Energy Control System

Original TECS proposed by Lambregts is based on energy rates:

- $T^c = T_D + k_{p,t}\dot{E}_t + k_{i,t}\int_{t_0}^{t} \dot{\tilde{E}}_t \delta_\tau$
  - T_D is thrust needed to counteract drag
  - PI controller based on total energy rate

- $\theta^c = k_{p,\theta}\dot{E}_d + k_{i,\theta}\int_{t_0}^{t} \dot{\tilde{E}}_d \delta_\tau$
  - PI controller based on energy distribution rate

- Stability shown for linear systems

We will show that the performance of this scheme is less than desirable.

Total Energy Control System

Nonlinear re-derivation:

- Error Definitions

$$\tilde{E}_K = \frac{1}{2}m\left(\left(V_a^d\right)^2 - V_a^2\right)$$
$$\tilde{E}_P = mg\left(h^d - h\right)$$

- Lyapunov Function

$$V = \frac{1}{2}\tilde{E}_T^2 + \frac{1}{2}\tilde{E}_D^2$$

- Controller

$$T^c = D + \frac{\tilde{E}_T^d}{V_a} + k_T\frac{\tilde{E}_T}{V_a}$$
$$\gamma^c = \sin^{-1}\left(\frac{\dot{h}^d}{V_a} + \frac{1}{2mgV_a}\left(-k_1\tilde{E}_K + k_2\tilde{E}_P\right)\right)$$

Total Energy Control System

- Original: $$T^c = D + k_{p,t}\frac{\dot{E}_T}{mgV_a} + k_{i,t}\frac{\tilde{E}_T}{mgV_a}$$
- Nonlinear: $$T^c = D + \frac{\dot{E}_T^d}{V_a} + k_T\frac{\tilde{E}_T}{V_a}$$

Similar if $k_{p,T} = mg$ and $k_{i,T} = mgk_T$.

The nonlinear controller uses the desired energy rate.

Total Energy Control System

- Modified Original (Ardupilot):

$$\theta^c = \frac{k_{p,\theta}}{V_a mg}\left((2-k)\,\dot{E}_P - k\dot{E}_K\right) + \frac{k_{i,\theta}}{V_a mg}\tilde{E}_D$$
$$k \in [0,2]$$

- Nonlinear:

$$\gamma^c = \sin^{-1}\left(\frac{\dot{h}^d}{V_a} + \frac{1}{2mgV_a}\left(-k_1\tilde{E}_K + k_2\tilde{E}_P\right)\right)$$
$$k_1 \triangleq |k_T - k_D|$$
$$k_2 \triangleq k_T + k_D$$
$$0 < k_T \leq k_D$$

Lyapunov derivation suggests potential energy error should be weighted more than kinetic energy

Total Energy Control System

If the drag is unknown, then we can add an adaptive estimate:

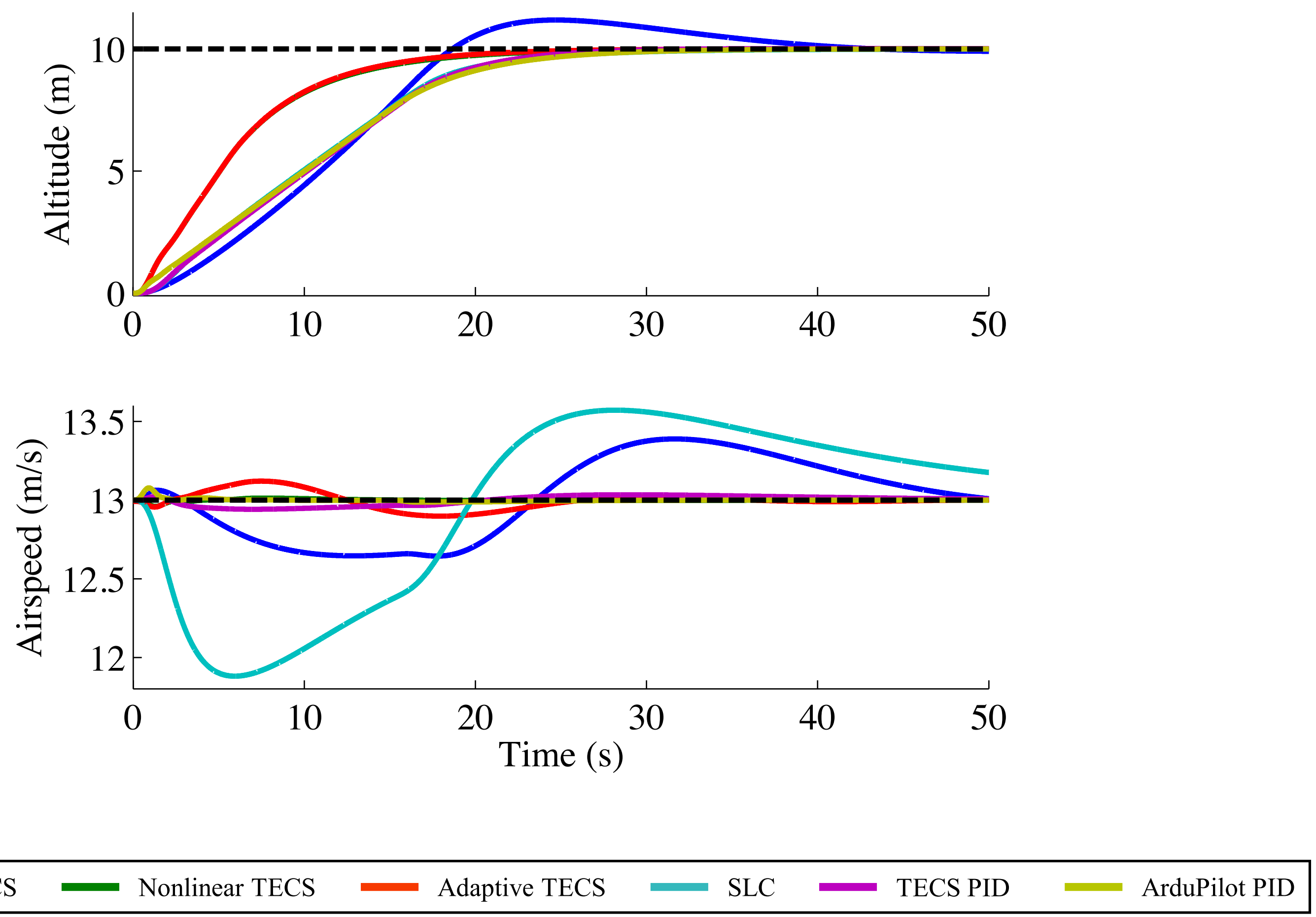
$$T^c = \hat{D} + \boldsymbol{\Phi}^\top \hat{\boldsymbol{\Psi}} + \frac{\dot{E}_T^d}{V_a} + k_T \frac{\tilde{E}_T}{V_a}$$

$$\gamma^c = \sin^{-1}\left(\frac{\dot{h}^d}{V_a} + \frac{1}{2mgV_a}\left(-k_1\tilde{E}_K + k_2\tilde{E}_P\right)\right)$$

$$\dot{\hat{\boldsymbol{\Psi}}} = \left(\Gamma_T\tilde{E}_T - \Gamma_D\tilde{E}_D\right)\boldsymbol{\Phi}V_a$$

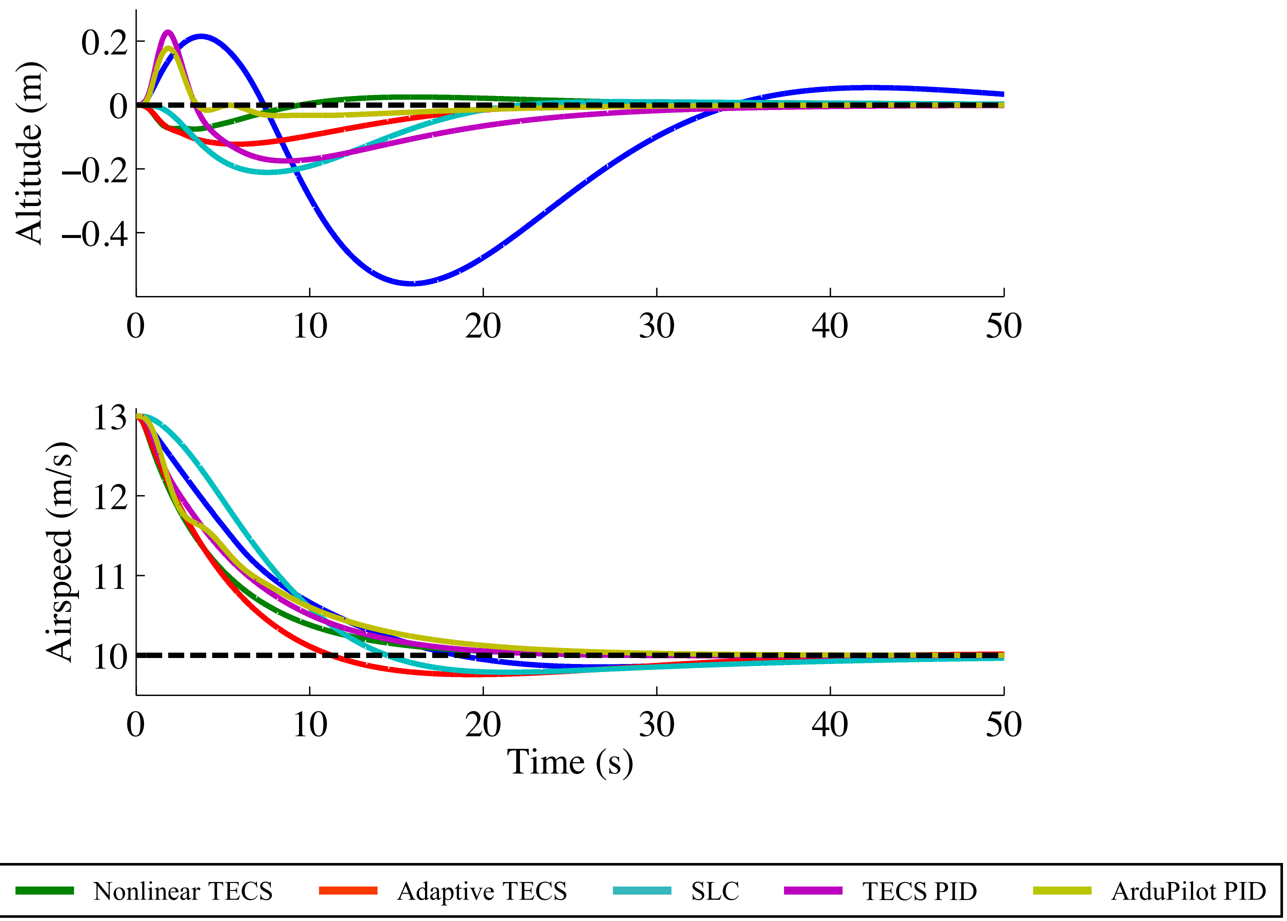
Total Energy Control System

Step in Altitude, Constant Airspeed



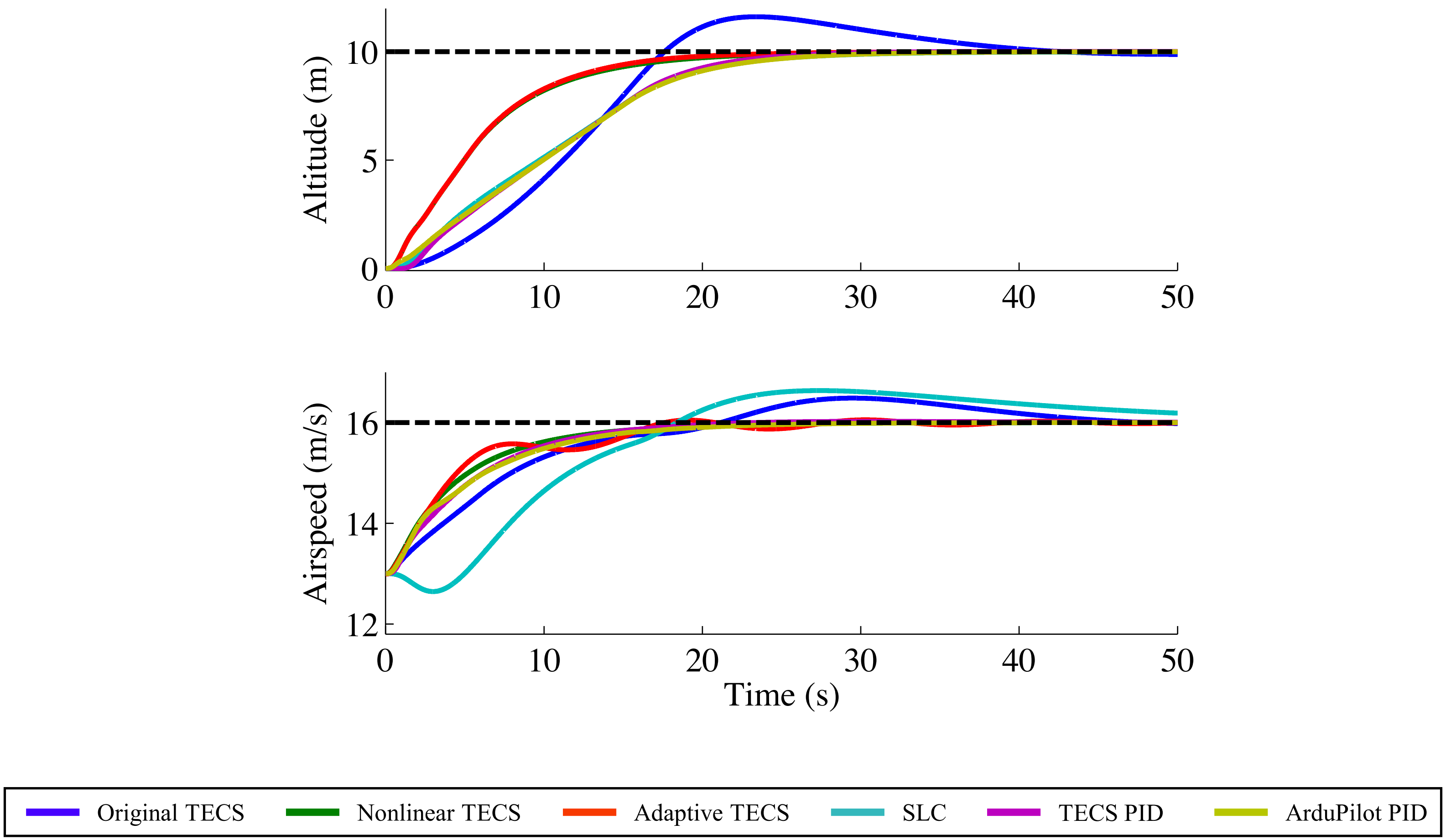
Total Energy Control System

Step in Airspeed, Constant Altitude



Total Energy Control System

Step in Altitude and Airspeed



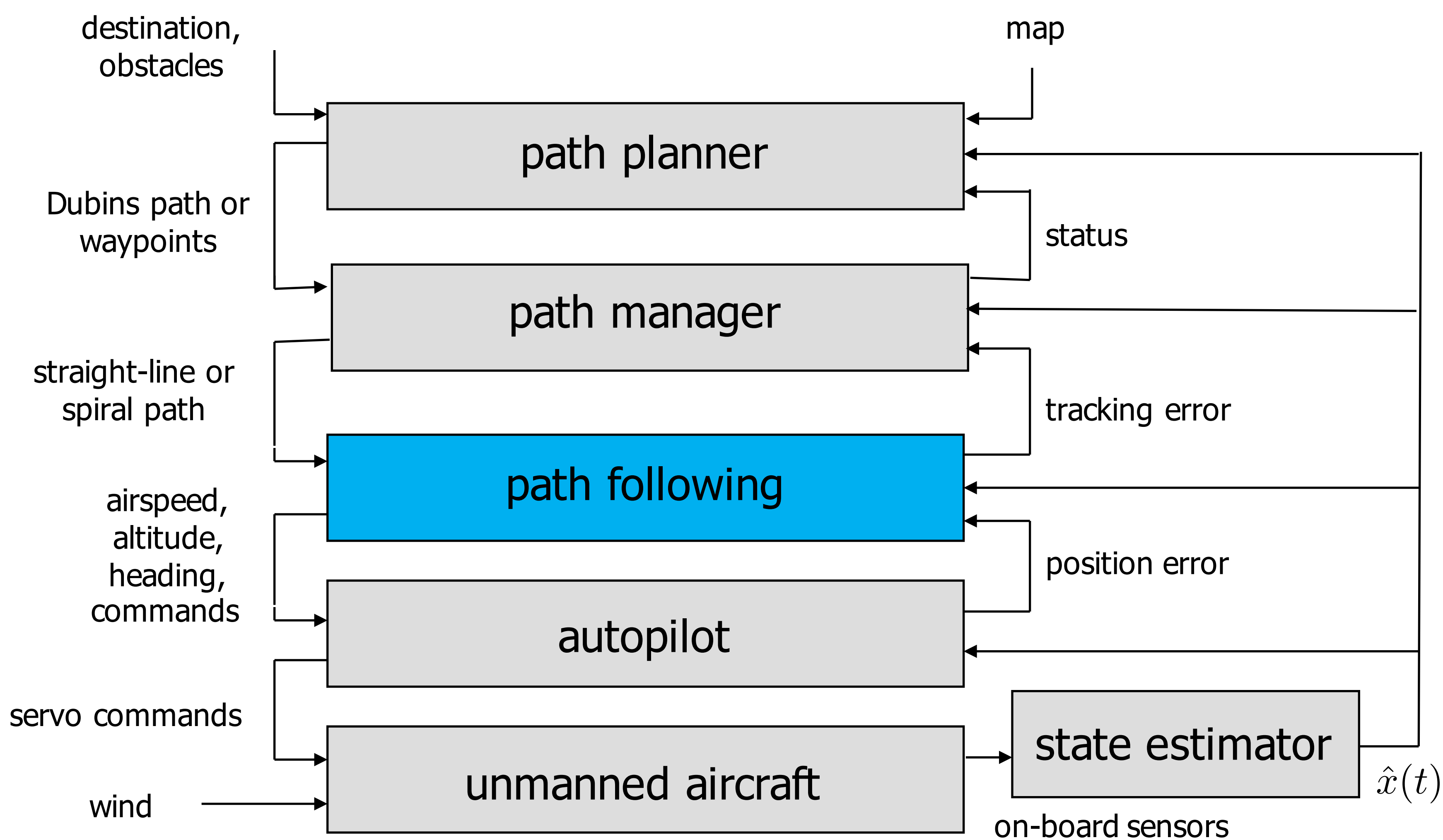
Total Energy Control System

- Observations
  - TECS seems to work better than successive loop closure.
  - Still need separate mode for take-off
  - Nonlinear TECS seems to work better, but the Ardupilot controller works very well.

Outline

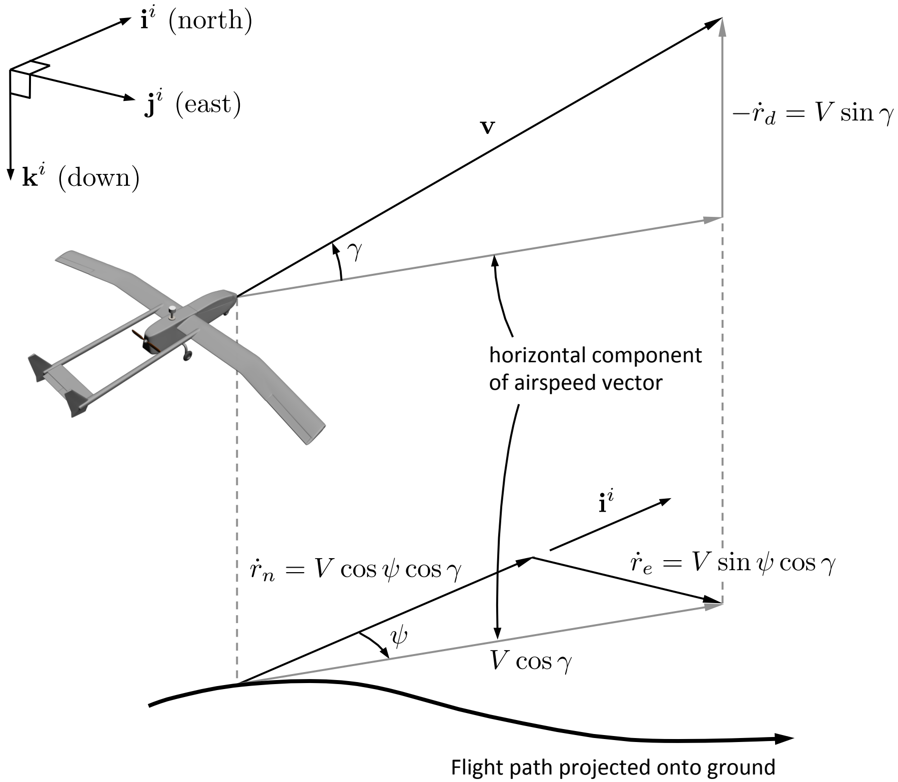
- Control architecture
- Low level autopilot loops
- Path following
- Dubins airplane paths and path management

Control Architecture



Focus on following straight lines and helices

Dubins Airplane Model



Dubins airplane model:

$$\dot{r}_n = V\cos\psi\cos\gamma^c$$
$$\dot{r}_e = V\sin\psi\cos\gamma^c$$
$$\dot{r}_d = -V\sin\gamma^c$$
$$\dot{\psi} = \frac{g}{V}\tan\phi^c.$$

with the constraint that

$$|\phi^c| \leq \bar{\phi}$$
$$|\gamma^c| \leq \bar{\gamma}.$$

3D Vector Field Path Following

Path defined as intersection of two 2D manifolds in $\mathbb{R}^3$:

$$\alpha_1(\mathbf{r}) = 0$$
$$\alpha_2(\mathbf{r}) = 0$$

The path given by the intersection is connected and one-dimensional. Define the function

$$V(\mathbf{r}) = \frac{1}{2}\alpha_1^2(\mathbf{r}) + \frac{1}{2}\alpha_2^2(\mathbf{r}).$$

Consider the velocity command

$$\mathbf{u}' = \underbrace{-K_1\frac{\partial V}{\partial \mathbf{r}}}_{\text{Forces cross track convergence}} + \underbrace{K_2\frac{\partial \alpha_1}{\partial \mathbf{r}} \times \frac{\partial \alpha_2}{\partial \mathbf{r}}}_{\text{Forces along track convergence}},$$

where $K_1 > 0$ and $K_2 > 0$.

Method based on: Goncalves, et. al., "Vector Fields for Robot Navigation Along Time-Varying Curves in n-Dimensions," IEEE Transactions on Robotics, vol. 26, pp. 647–659, 2010.

3D Vector Field Path Following

To ensure that the magnitude of the velocity vector is V, normalize $\mathbf{u}'$ as

$$\mathbf{u} = V \frac{\mathbf{u}'}{\|\mathbf{u}'\|}.$$

The commanded flight-path angle γ^c, and the desired heading angle ψ^d are therefore given by

$$\gamma^c = -\mathrm{sat}_{\bar{\gamma}}\left[\sin^{-1}\left(\frac{u_3}{V}\right)\right]$$
$$\psi^d = \mathrm{atan2}(u_2, u_1).$$

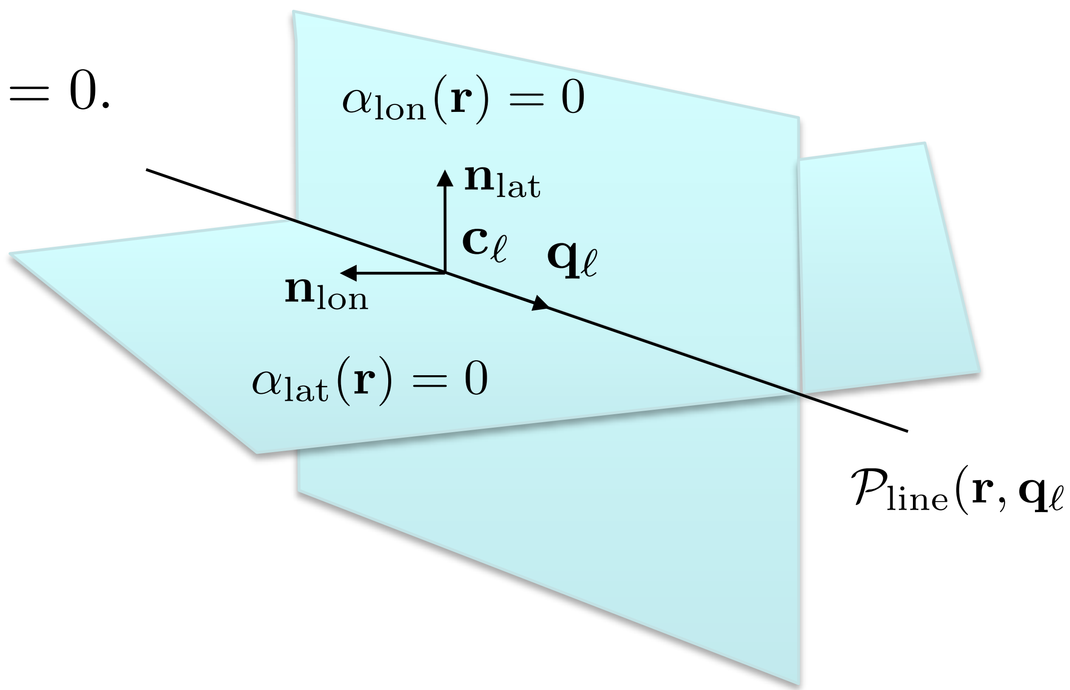
Method based on: Goncalves, et. al., “Vector Fields for Robot Navigation Along Time-Varying Curves in n-Dimensions,” IEEE Transactions on Robotics, vol. 26, pp. 647–659, 2010.

3D Vector Field Path Following

Straight-line Paths

$$\alpha_{\text{lon}}(\mathbf{r}) \stackrel{\triangle}{=} \mathbf{n}_{\text{lon}}^{\top}(\mathbf{r} - \mathbf{c}_{\ell}) = 0$$

$$\alpha_{\text{lat}}(\mathbf{r}) \stackrel{\triangle}{=} \mathbf{n}_{\text{lat}}^{\top}(\mathbf{r} - \mathbf{c}_{\ell}) = 0.$$



Parameterized by

- Start Position $\mathbf{c}_{\ell}$

Method based on: Goncalves, et. al., "Vector Fields for Robot Navigation Along Time-Varying Curves in n-Dimensions," IEEE Transactions on Robotics, vol. 26, pp. 647–659, 2010.

3D Vector Field Path Following

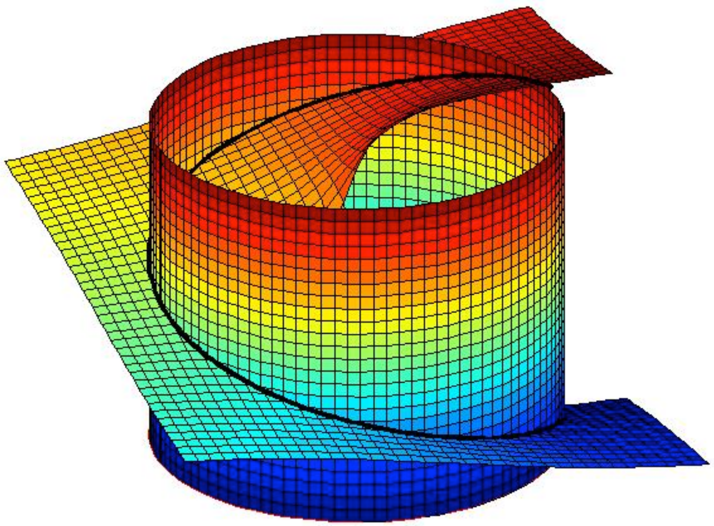
Helical Paths

$$\alpha_{\text{cyl}}(\mathbf{r}) = \left(\frac{r_n - c_n}{R_h}\right)^2 + \left(\frac{r_e - c_e}{R_h}\right)^2 - 1$$

$$\alpha_{\text{pl}}(\mathbf{r}) = \left(\frac{r_d - c_d}{R_h}\right) + \frac{\tan\gamma_h}{\lambda_h}\left(\tan^{-1}\left(\frac{r_e - c_e}{r_n - c_n}\right) - \psi_h\right).$$

Parameterized by

- Spiral Center (c_n, c_e, c_d)
- Radius R_h
- Direction λ_h
- Initial heading ψ_h

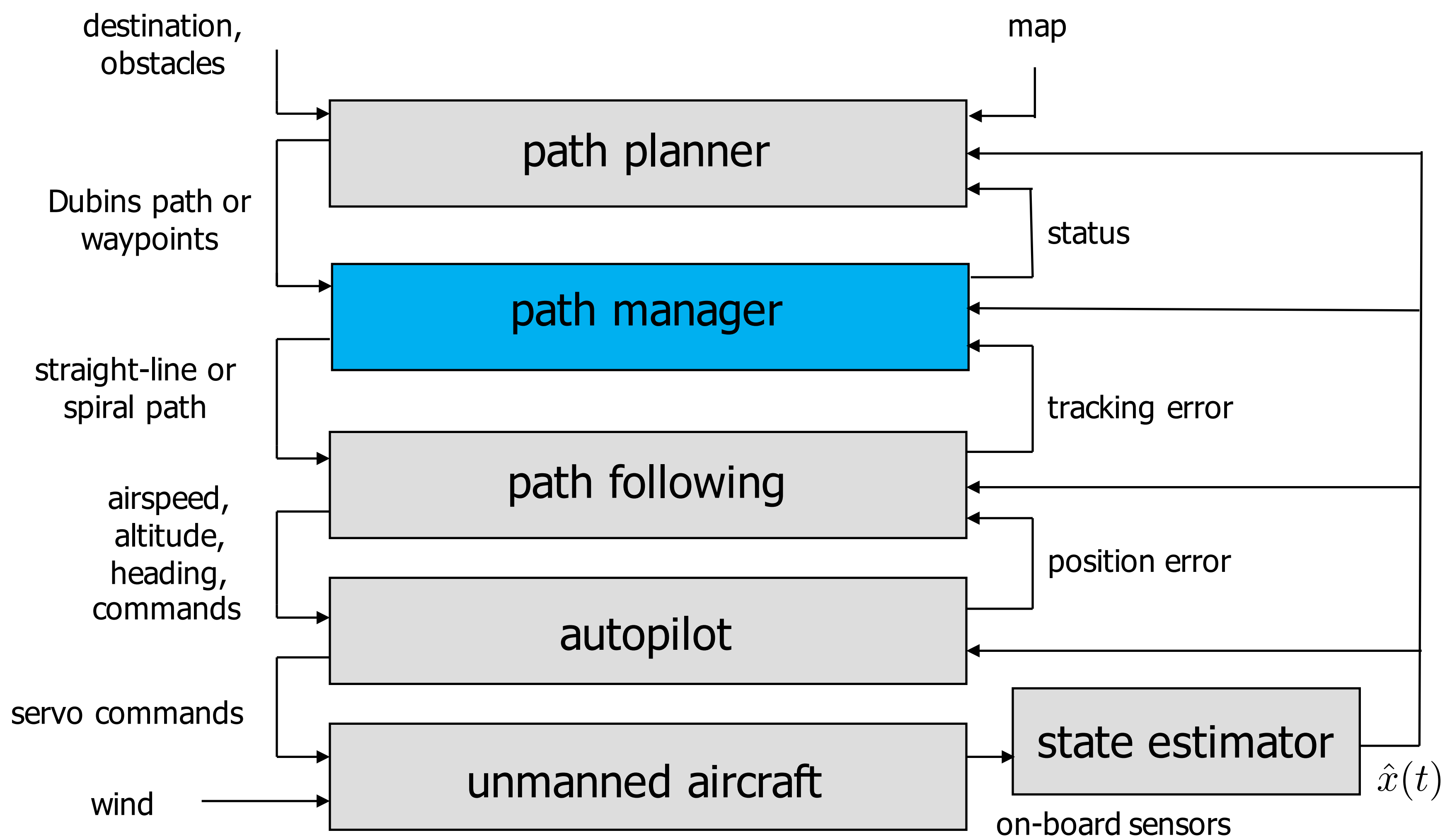


Method based on: Goncalves, et. al., "Vector Fields for Robot Navigation Along Time-Varying Curves in n-Dimensions," IEEE Transactions on Robotics, vol. 26, pp. 647–659, 2010.

Outline

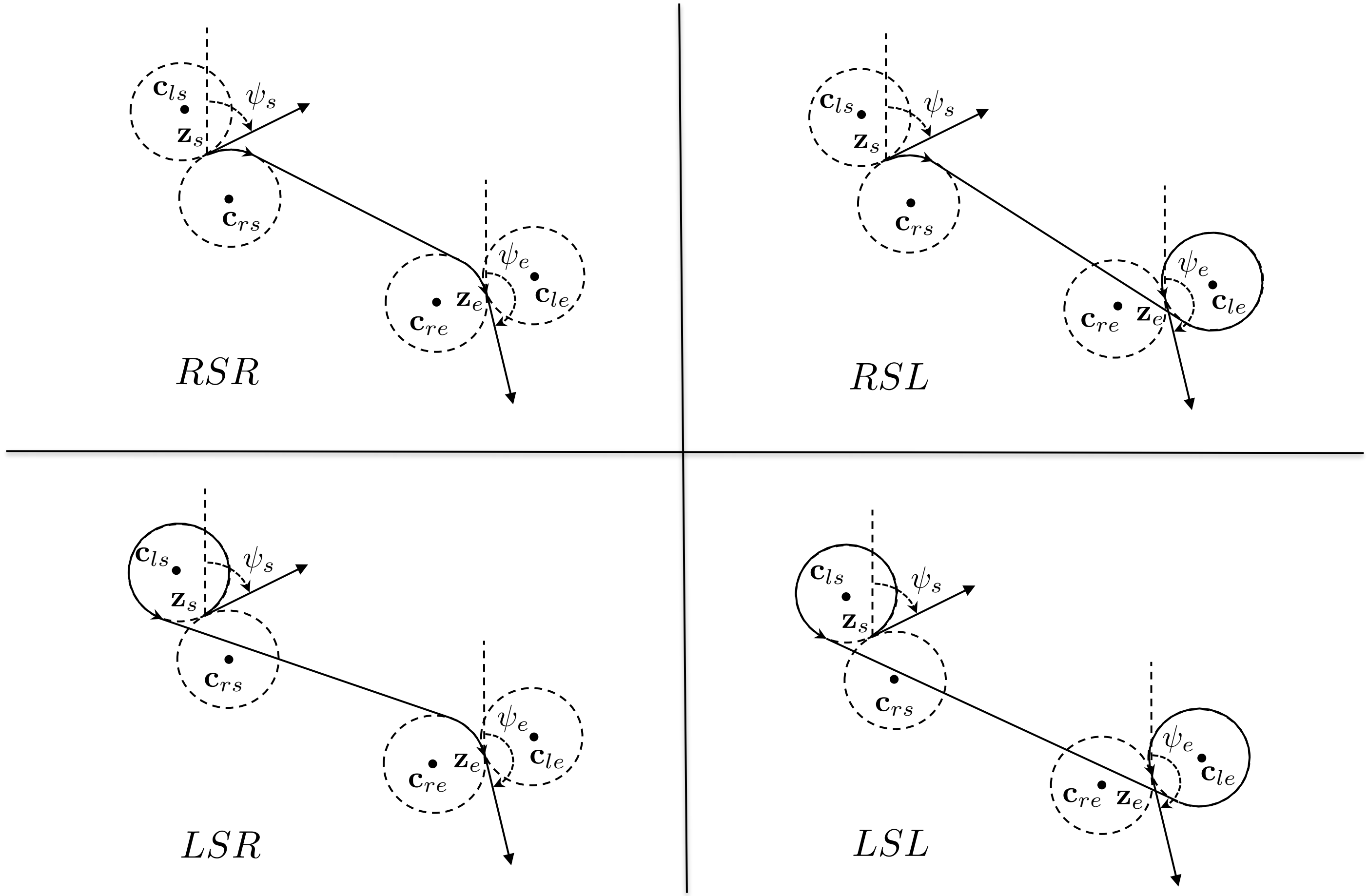
- Control architecture
- Low level autopilot loops
- Path following
- Dubins airplane paths and path management

Control Architecture



Dubins Airplane

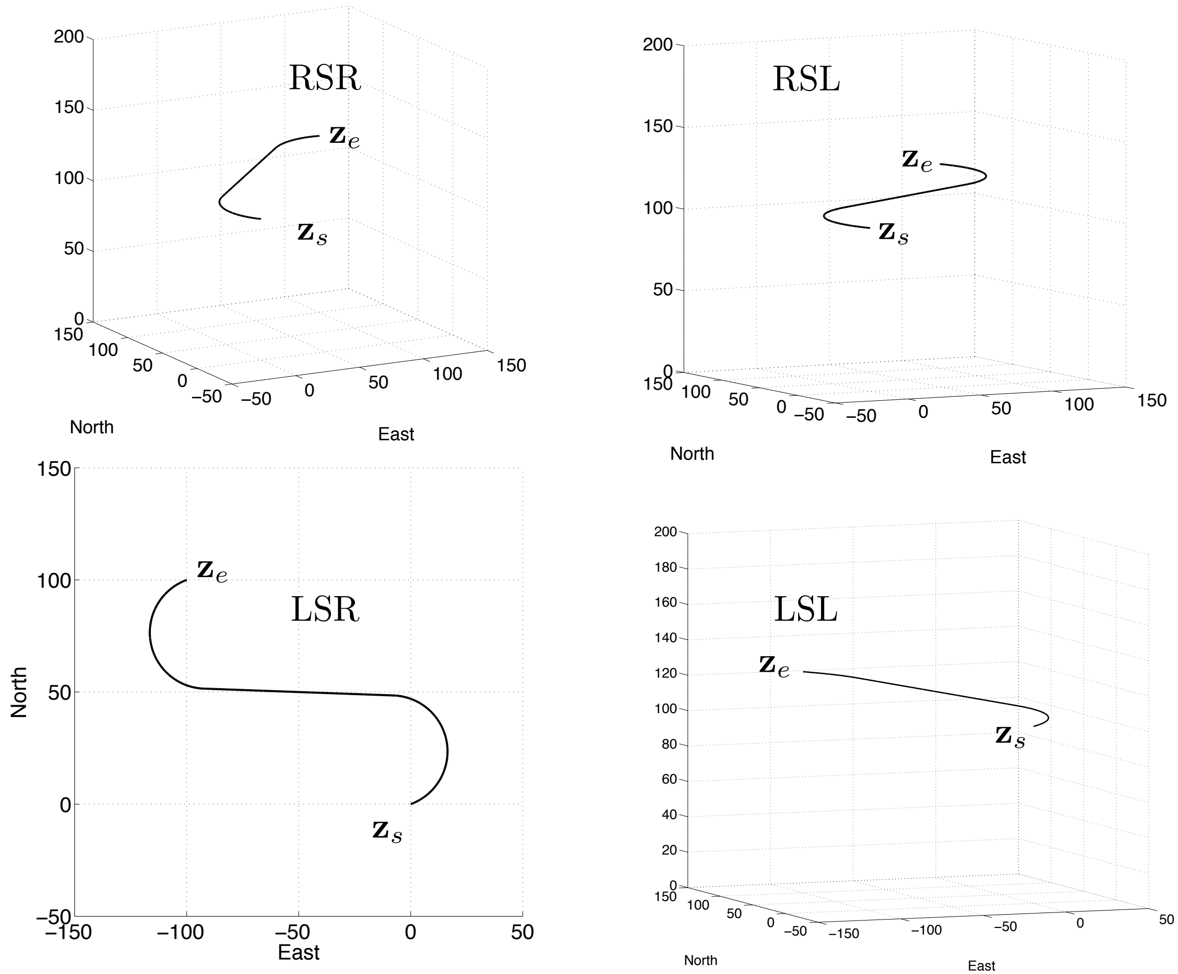
Options for Dubins car



For Dubins airplane, also need to address altitude

Dubins Airplane

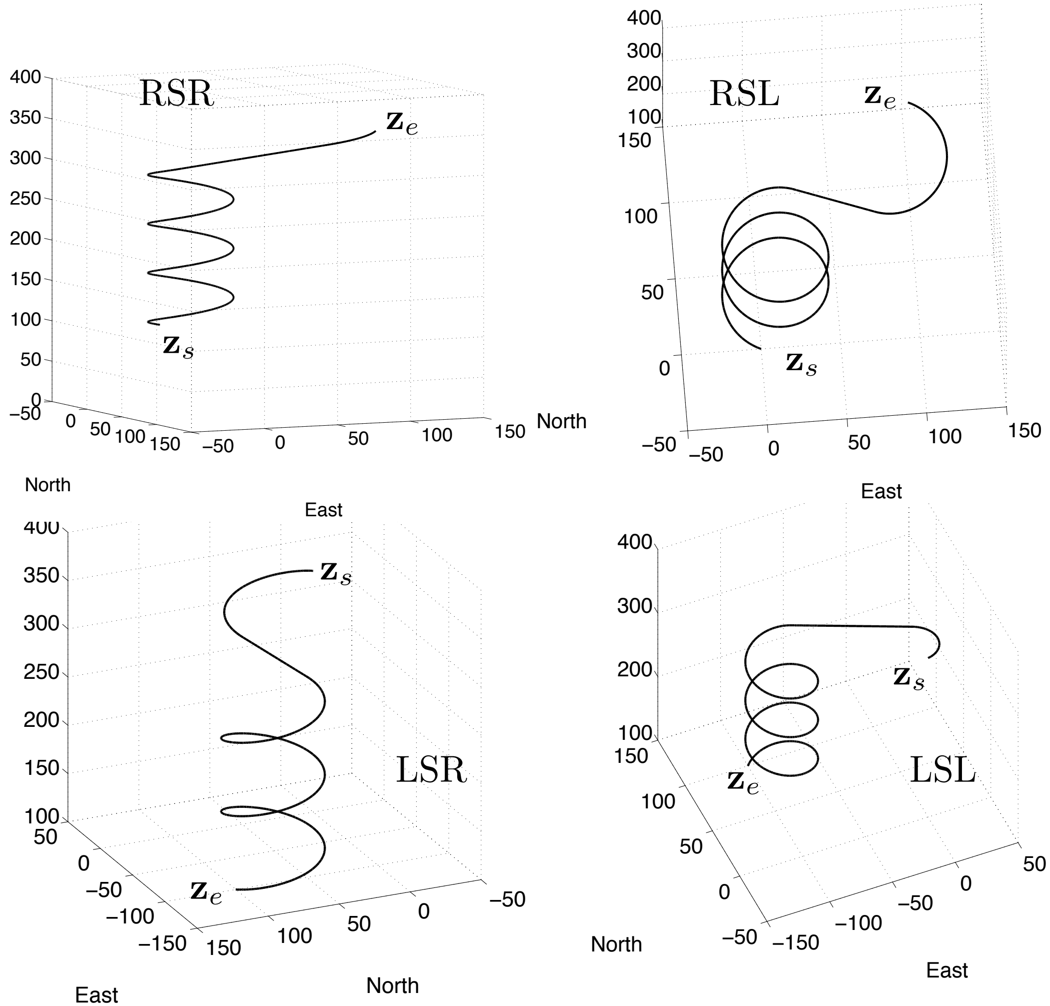
Low altitude case: $|z_{de} - z_{ds}| \leq L_{\text{car}}(R_{\min}) \tan \bar{\gamma}$,
Can achieve altitude gain without modifying Dubins car path.



Dubins Airplane

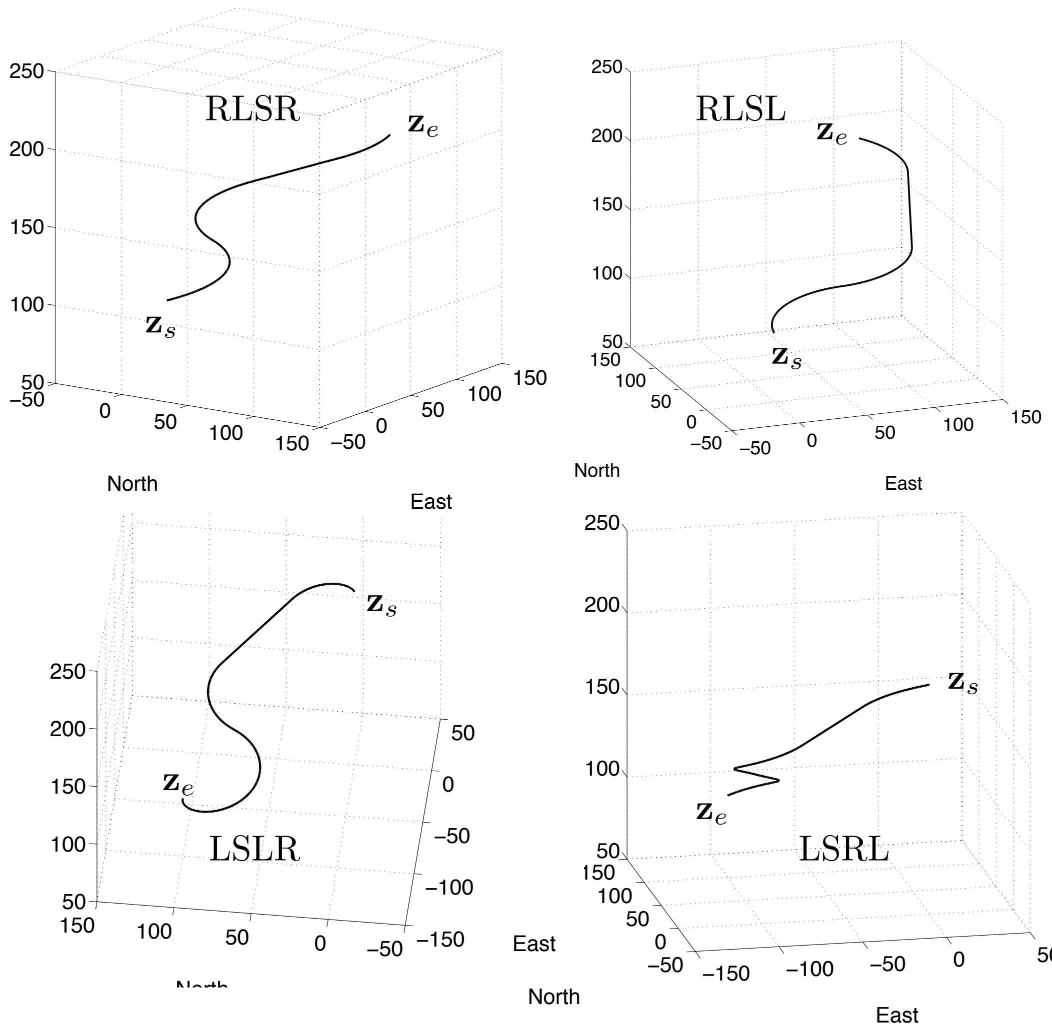
High altitude case: $|z_{de} - z_{ds}| > [L_{\text{car}}(R_{\min}) + 2\pi R_{\min}] \tan \bar{\gamma}$.

Can achieve altitude gain by spiraling at beginning or end of Dubins car path.

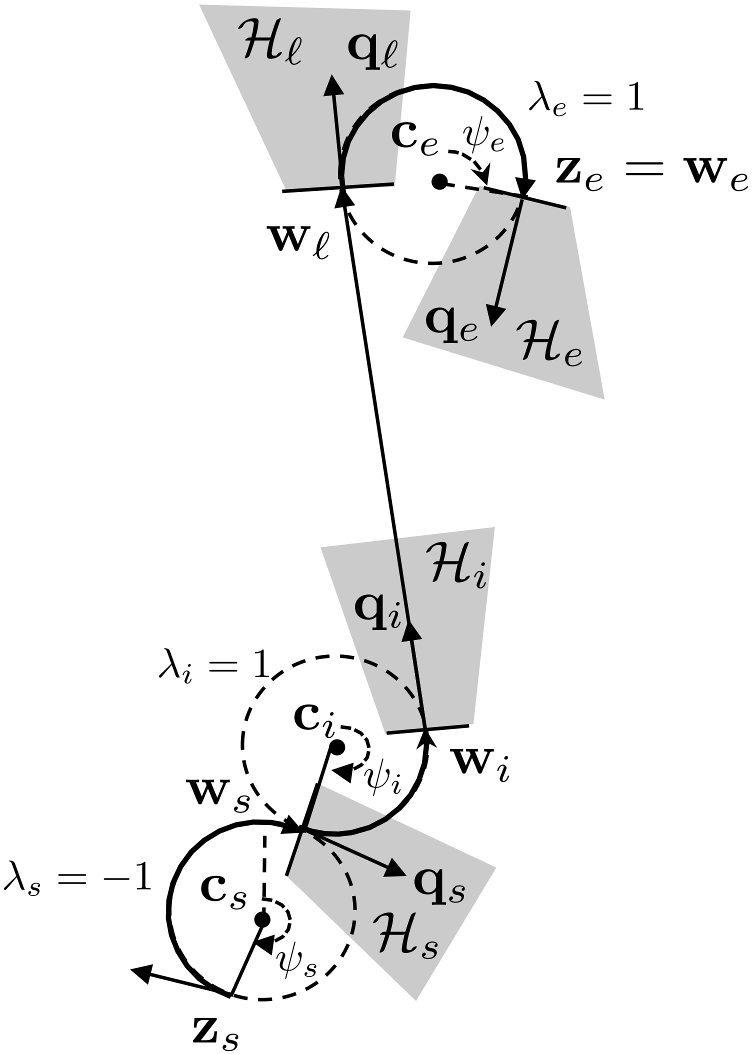


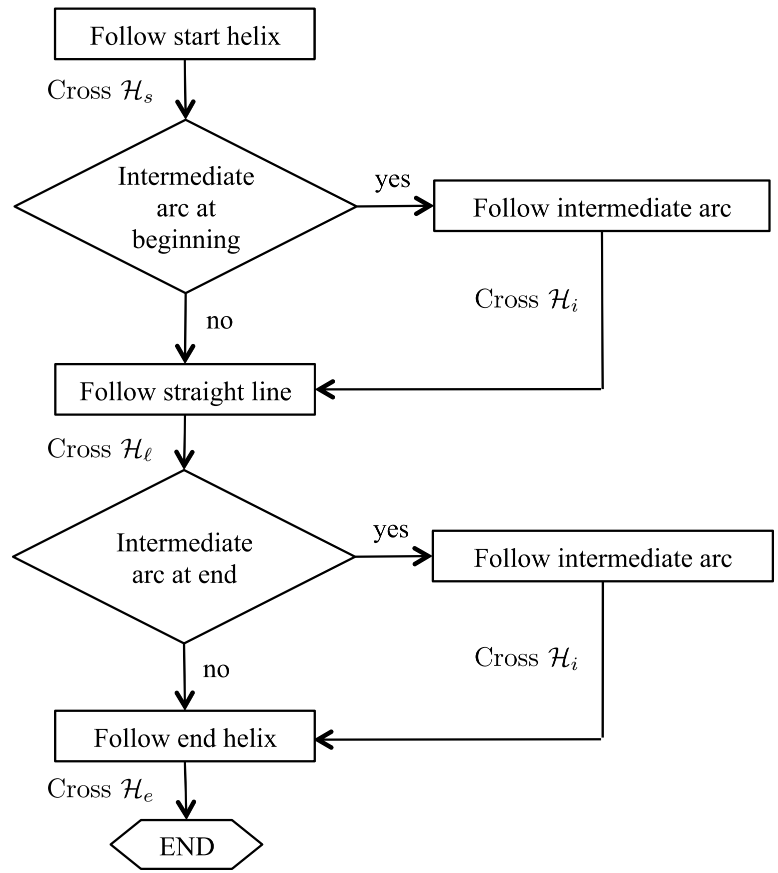
Dubins Airplane

Medium altitude case: $L_{\text{car}}(R_{\min}) \tan \bar{\gamma} < |z_{de} - z_{ds}| \leq [L_{\text{car}}(R_{\min}) + 2\pi R_{\min}] \tan \bar{\gamma}$
To achieve altitude gain, must add path deviation.



Path Management





Summary

- **Autopilot**
  - Successive Loop Closure for Lateral Control
  - Total Energy Control System for Longitudinal Control
- **Path Following**
  - 3D Vector field method
- **Path Management**
  - Dubins airplane paths
- Methods are computationally simple, and easily fit on embedded processors.
- Methods have flown extensively on small fixed wing UAS.

Questions?