Нелинейное моделирование и управление гибридными БЛА

Nonlinear Modeling and Control of Hybrid UAVs

Ben M. Chen

On Behalf of Unmanned Systems Research Groups

Chinese University of Hong Kong

Peng Cheng Laboratory

National University of Singapore

Moscow, Russia. September 10–12, 2020

Outline of my talk…

- Introduction
- Hybrid UAV platforms
- Dynamics modeling
- Flight control system
- Some applications
- Concluding remarks

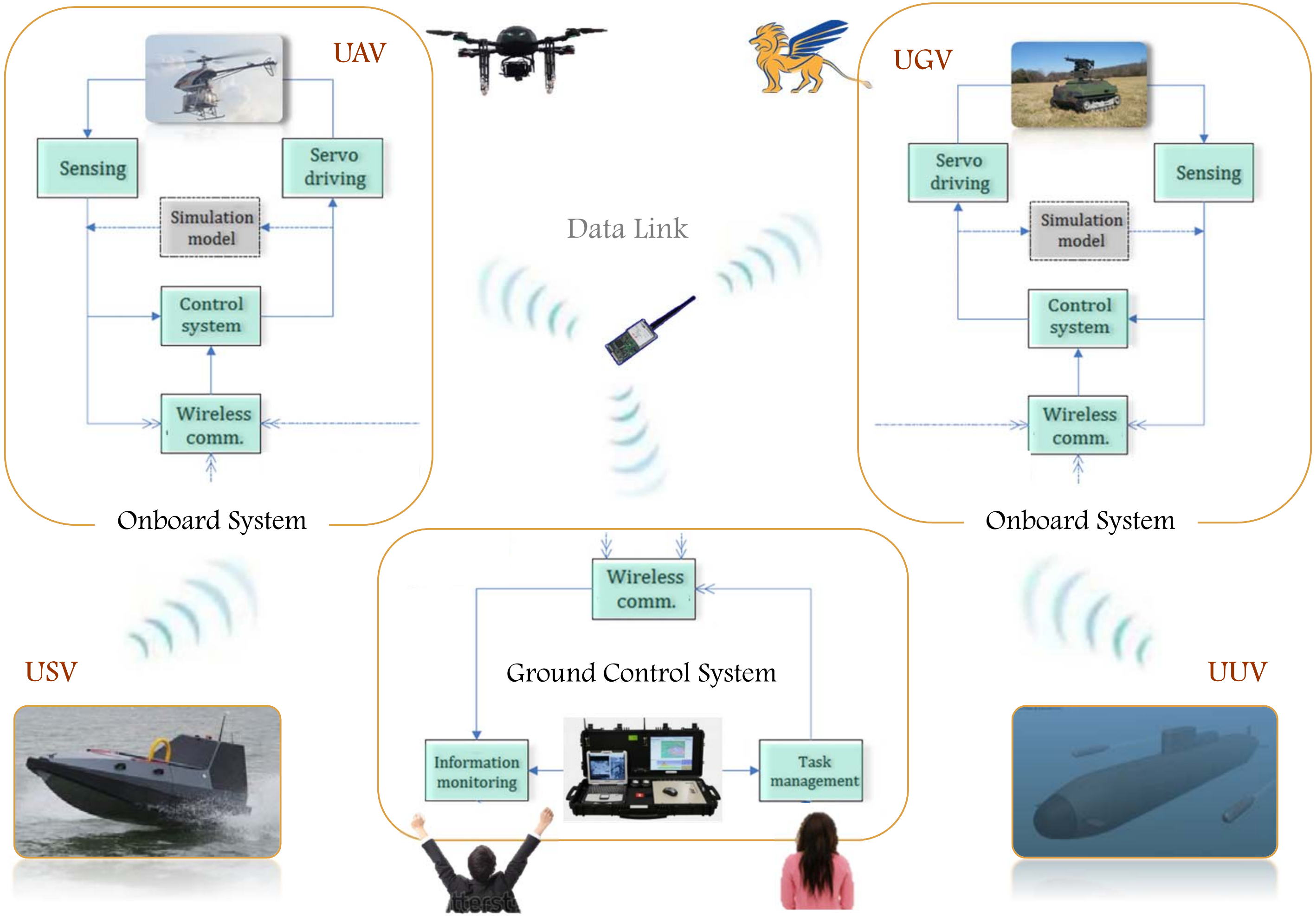
План моего выступления

Введение
Гибридные платформы БПЛА
Моделирование динамики
Система управления полетом
Некоторые приложения
Заключительные замечания

An unmanned system architecture   Архитектура беспилотной системы

Беспилотные летательные аппараты

Беспилотные наземные машины

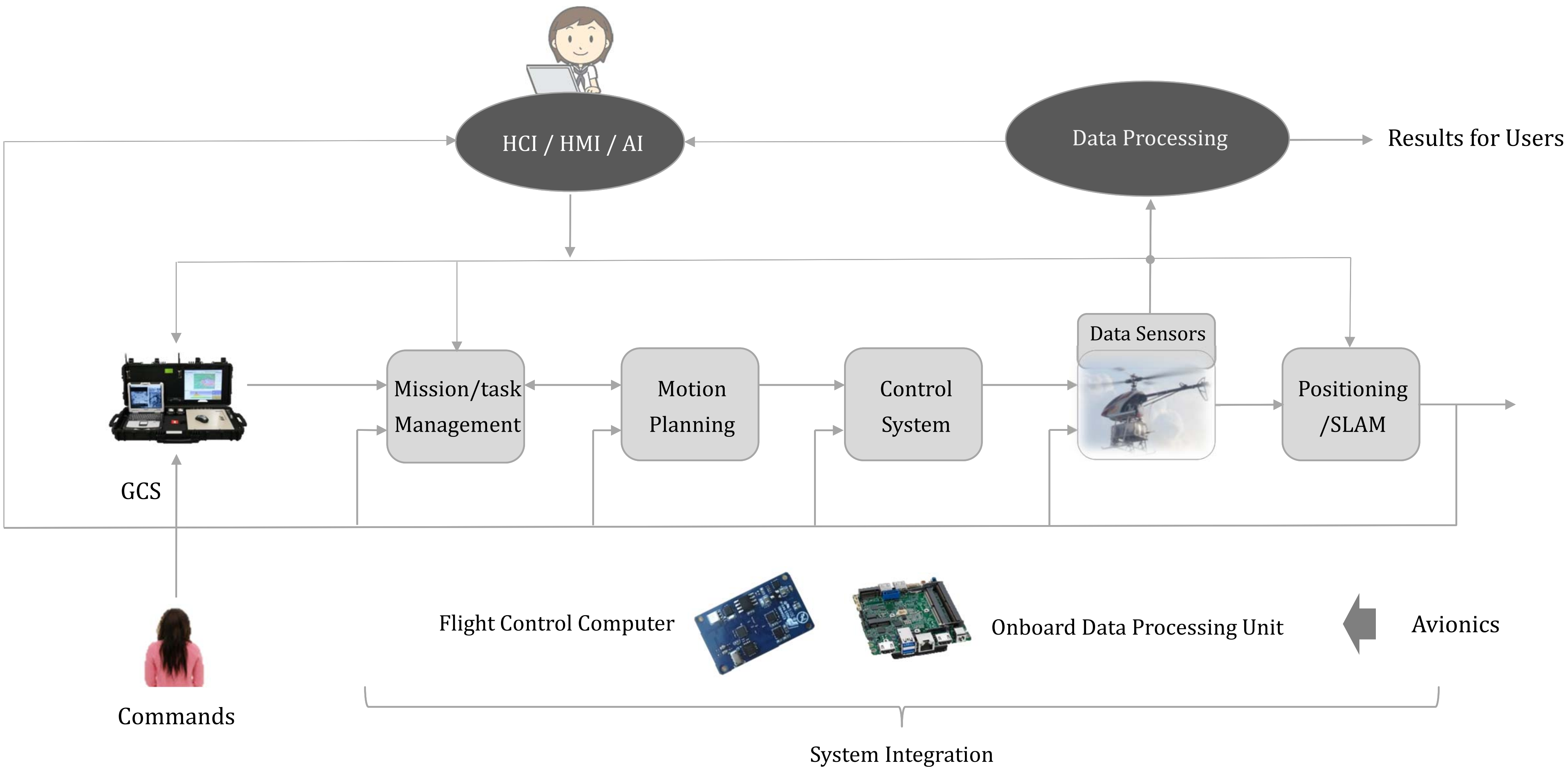


Беспилотное надводное судно

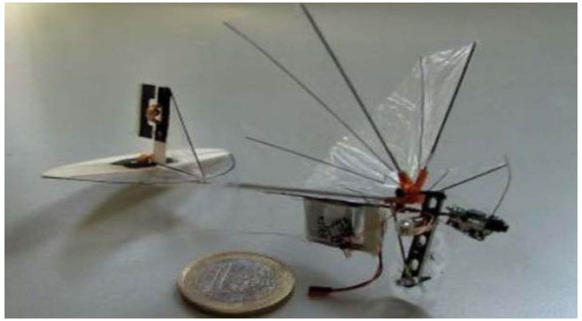
Беспилотное подводное судно

Internal framework of an intelligent autonomous UAS…

Внутренний каркас интеллектуального автономного БПЛА



Some common/uncommon drones… Некоторые обычные/необычные дроны



Why hybrid UAVs?

Почему гибридные БПЛА?

Long Range
Flight Efficiency

VTOL Maneuverability

An aircraft with VTOL and cruise flight capability

Some existing hybrid UAVs Некоторые существующие гибридные БЛА

Evolution of our hybrid UAVs…

Эволюция наших гибридных БПЛА

……?

Dynamics modeling… Моделирование динамики

- Kinematics:

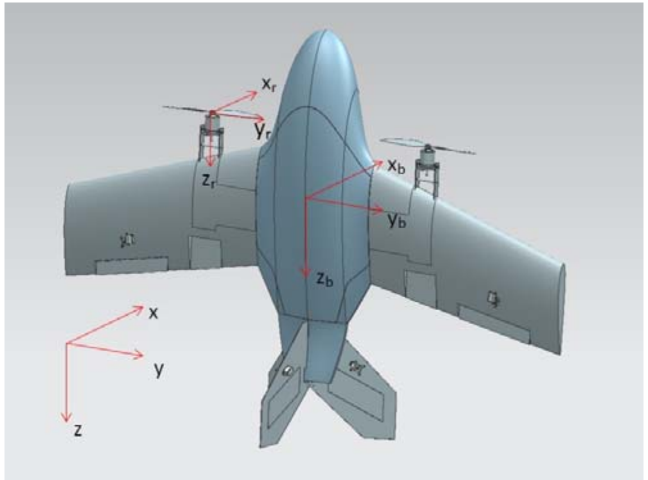
$$\dot{\mathbf{P}}_{\rm n} = \mathbf{R}_{\rm n/b}\mathbf{V}_{\rm b},$$
$$\dot{\mathbf{R}}_{\rm n/b} = \mathbf{W}\mathbf{R}_{\rm n/b},$$

- Rigid body dynamics:

$$m\dot{\mathbf{V}}_{\mathbf{b}} + \omega \times (m\mathbf{V}_{\mathbf{b}}) = \mathbf{F},$$
$$\mathbf{J}\dot{\omega} + \omega \times (\mathbf{J}\omega) = \mathbf{M}$$

- Forces and moments:

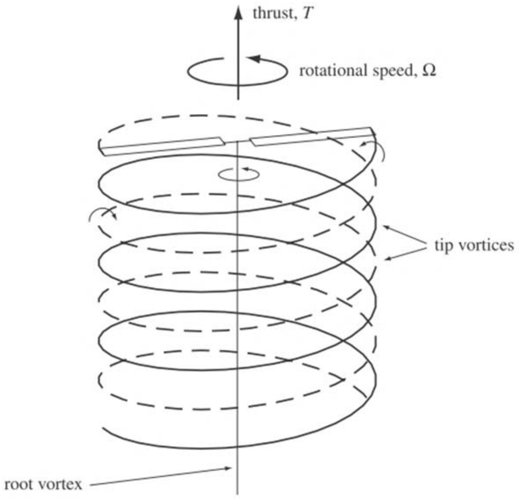
$$\mathbf{F} = \mathbf{F}_{\rm grav} + \mathbf{F}_{\rm prop} + \mathbf{F}_{\rm aero},$$
$$\mathbf{M} = \mathbf{M}_{\rm prop} + \mathbf{M}_{\rm fin} + \mathbf{M}_{\rm aero},$$



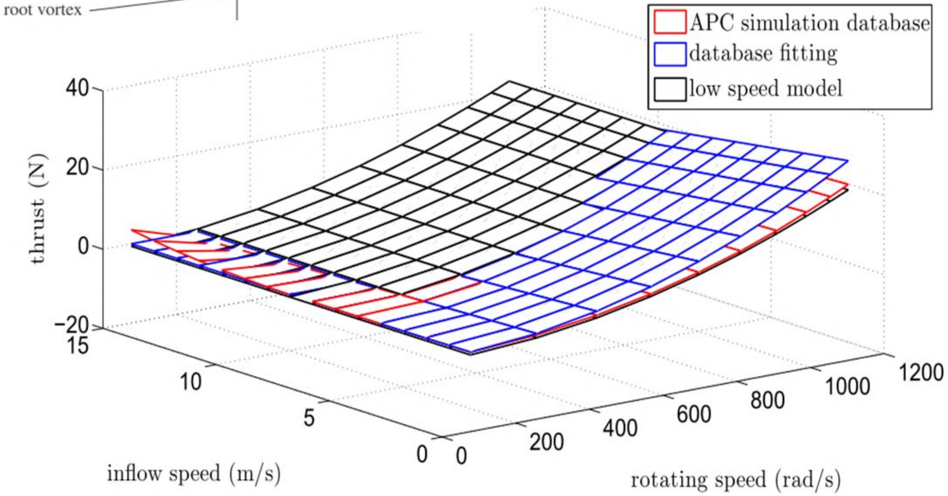
P Position vector
V Velocity vector
F Force vector
M Moment vector
R Rotation matrix
W Angular velocity tensor
J Moment of inertia matrix
b Body frame
n Local NED frame
m Mass
ω Angular velocity

Dynamics modeling (propeller dynamics)…

Experiment Setup



Моделирование динамики (динамика воздушного винта)



Simulation and
Experimental Results

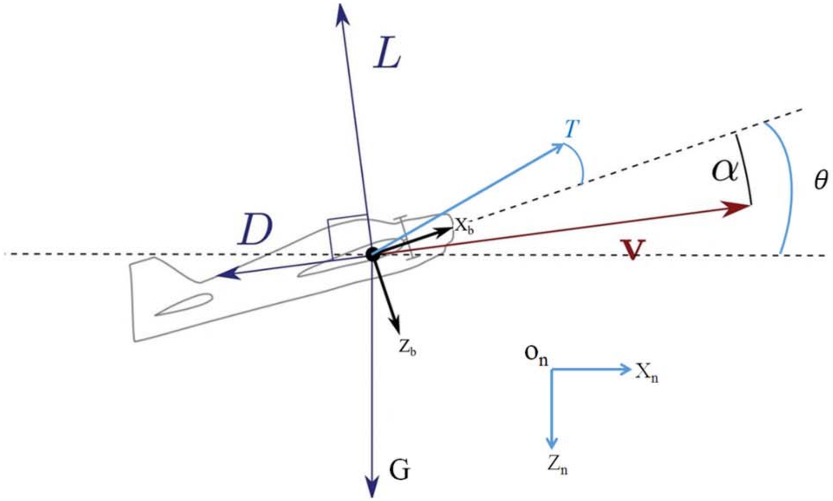
Dynamics modeling (transition mode)…

Моделирование динамики (переходный режим)

Define state as $\mathbf{x} = [\, u \quad w \quad q \quad \theta \,]^{\mathrm{T}}$, input as $\mathbf{u} = [\, T_{\mathrm{u}} \quad T_{\mathrm{w}} \quad T_{\mathrm{f}} \,]^{\mathrm{T}}$, then main dynamics:

$$\begin{aligned} \dot{u} &= \frac{1}{\mathrm{m}} F_{\mathrm{a,x}}(\boldsymbol{x}) - g\sin(\theta) - qw + \frac{1}{m} T_{\mathrm{u}} + \delta_{\mathrm{u}}(t), \\ \dot{w} &= \frac{1}{m} F_{\mathrm{a,z}}(\boldsymbol{x}) + g\cos(\theta) + qu - \frac{1}{m} T_{\mathrm{w}} + \frac{1}{m} T_{\mathrm{f}} + \delta_{\mathrm{w}}(t), \\ \dot{q} &= \frac{1}{I_{\mathrm{y}}} M_{\mathrm{a}}(\boldsymbol{x}) + \frac{l_{\mathrm{m}}}{I_{\mathrm{y}}} T_{\mathrm{w}} + \frac{l_{\mathrm{t}}}{I_{\mathrm{y}}} T_{\mathrm{f}} + \delta_{\mathrm{q}}(t), \\ \dot{\theta} &= q, \end{aligned} \qquad (1)$$

where $$\begin{bmatrix} F_{\mathrm{a,x}}(\boldsymbol{x}) \\ F_{\mathrm{a,z}}(\boldsymbol{x}) \end{bmatrix} = \begin{bmatrix} \sin\alpha & -\cos\alpha \\ -\cos\alpha & -\sin\alpha \end{bmatrix} \begin{bmatrix} L(\alpha) \\ D(\alpha) \end{bmatrix}$$



$$\begin{aligned} L(\alpha) &= \tfrac{1}{2}(u^2 + w^2)\rho A_{\mathrm{w}} C_{\mathrm{L}}(\alpha), \\ D(\alpha) &= \tfrac{1}{2}(u^2 + w^2)\rho A_{\mathrm{w}} C_{\mathrm{D}}(\alpha). \\ M_{\mathrm{a}}(\boldsymbol{x}) &= \tfrac{1}{2}(u^2 + w^2)\rho A_{\mathrm{w}} C_{\mathrm{M}}(\alpha). \end{aligned}$$

$$\alpha = \mathrm{atan2}(u/w)$$

$\delta_{\mathrm{q}}(t)$	The unknown perturbations in q dynamics
$\delta_{\mathrm{u}}(t)$	The unknown perturbations in u dynamics
$\delta_{\mathrm{w}}(t)$	The unknown perturbations in w dynamics
$C_{\mathrm{D}}(\alpha)$	Aerodynamic drag coefficient
$C_{\mathrm{L}}(\alpha)$	Aerodynamic lift coefficient
$C_{\mathrm{M}}(\alpha)$	Aerodynamic moment coefficient

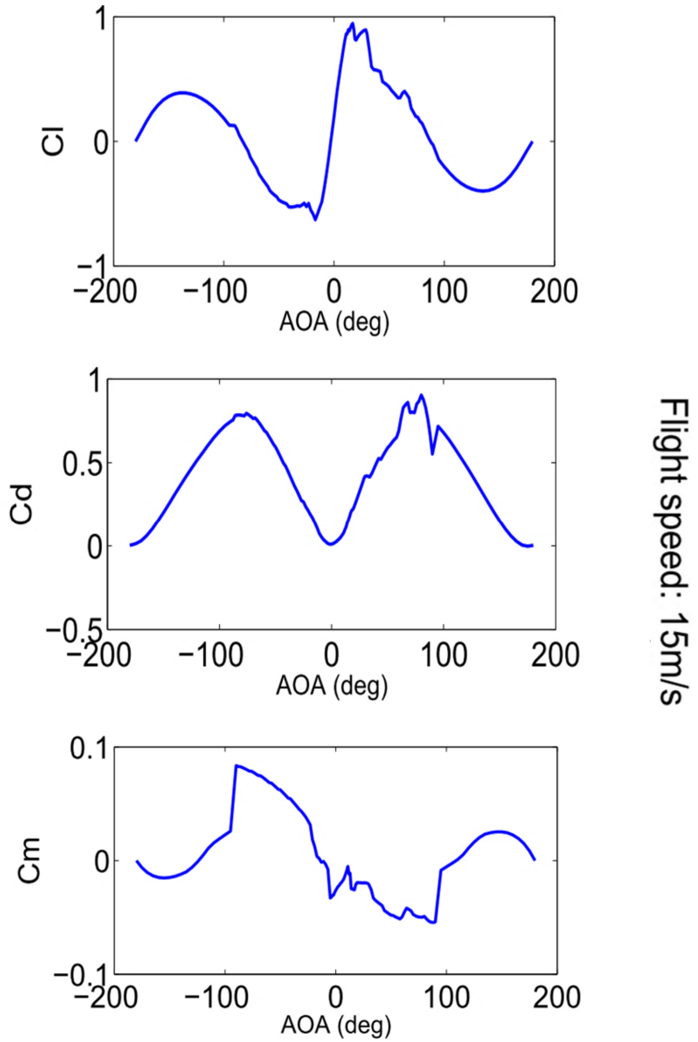
ρ	Air density
A_{w}	Surface area of the wing
g	Gravity constant
I_{y}	Moment of inertia of KH-Lion in $\boldsymbol{Y}$-direction
l_{f}	Distance from the tail fin center to the CG along $\boldsymbol{X}_b$
l_{m}	Distance from motor to the CG along $\boldsymbol{X}_b$
m	Mass of KH-Lion

γ_{v}	Vectoring thrust titling angle
γ_{f}	Tail fin control surface deflection angle
θ	Pitch angle of KH-Lion
q	Angular speed in pitch direction
T_{f}	Force generated by the tail fin control surfaces in $\boldsymbol{Z}_{\mathrm{b}}$ direction
T_{u}	Thrust decomposition in $\boldsymbol{X}_{\mathrm{b}}$-axis direction
T_{w}	Thrust decomposition in $\boldsymbol{Z}_{\mathrm{b}}$-axis direction
u	Velocity in the body frame $\boldsymbol{X}_{\mathrm{b}}$-axis direction
w	Velocity in the body frame $\boldsymbol{Z}_{\mathrm{b}}$-axis direction

Dynamics modeling (aerodynamics coefficients)…

- Dynamics model is highly nonlinear and complex
- Aerodynamics coefficients depend on speed and AOA
- High AOA dynamics difficult to measure and estimate
- Uncertainties & disturbance
- Input constraints
- State (velocity, acceleration, etc.) constraints

Моделирование динамики (аэродинамические коэффициенты)



Flight control systems (transition mode)…

Системы управления полетом (переходный режим)

- Find an optimal control law for the transition
  - Discretise the feasible state and action space
  - Find optimal action for each state in state space based on DP algorithm
  - The control law for a random state is obtained by interpolating the state in the state spaces, and sum up the corresponding actions
- Find the optimal trajectory for the transition
  - Apply the optimal control law obtained to the model from an initial condition
  - Recording the input and state trajectory for the transition
- Advantages of DP algorithm
  - The complexity of the system does not affect the algorithm complexity
  - Well handled the input and state constraints

Flight control systems (transition mode)…

Denote the system (1) as

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, \boldsymbol{u})$$

Discretising the system:

$$\boldsymbol{x}[n+1] = f_T(\boldsymbol{x}[n], \boldsymbol{u}[n])$$

To find an optimal policy:

$$\boldsymbol{u}[n] = \pi^*(\boldsymbol{x}[n])$$

which optimizing the cost function:

$$J(\boldsymbol{x}_0) = h(\boldsymbol{x}[N]) + \sum_{n=0}^{N-1} g(\boldsymbol{x}, \boldsymbol{u}, n)$$

The cost function is selected to be:

$$\begin{aligned} (\boldsymbol{x}[n]) &= (\boldsymbol{x}[n] - \boldsymbol{x}_{\mathrm{c}})^T Q_{\mathrm{f}} (\boldsymbol{x}[n] - \boldsymbol{x}_{\mathrm{c}}) \\ (\boldsymbol{x}, \boldsymbol{u}, n) &= \boldsymbol{u}[n]^T R_{\mathrm{f}} \boldsymbol{u}[n], \end{aligned}$$

x_c Nominal state of cruise flight

Q_f, R_f Weight matrix

Rewrite the cost function in recursive format:

$$\begin{aligned} J^*(\boldsymbol{x}, n) &= \min_{\boldsymbol{u}} [g(\boldsymbol{x}, \boldsymbol{u}, n) + J^*(\boldsymbol{x}[n+1], n+1)], \\ \pi^*(\boldsymbol{x}, n) &= \operatorname*{argmin}_{\boldsymbol{u}} [g(\boldsymbol{x}, \boldsymbol{u}, n) + J^*(\boldsymbol{x}[n+1], n+1)] \end{aligned}$$

The optimal policy can be found by DP algorithm:

Data: State set S, action set A
Result: For each $s \in S$, the optimal policy $\pi^*(s)$ and optimal cost $J^*(s)$

for *each state* $s \in S$ **do**
  $J^*(s) \leftarrow h(s)$
end
while $J^*(s)$ *not converged* **do**
  for *each state* s **do**
    for *each action* a **do**
      $s' \leftarrow f(s, a)$;
      do volumetric interpolation s' in S so that
      $s' = \sum_{m=1}^{16} w_m s_m,\ s_m \in S$
      $J = g(s, a) + \sum_{m=1}^{16} w_m J^*(s_m)$
      if $J < J^*(s)$ **then**
        $J^*(s) \leftarrow J$
        $\pi^* \leftarrow a$
      end
    end
  end
end

Системы управления полетом (переходный режим)

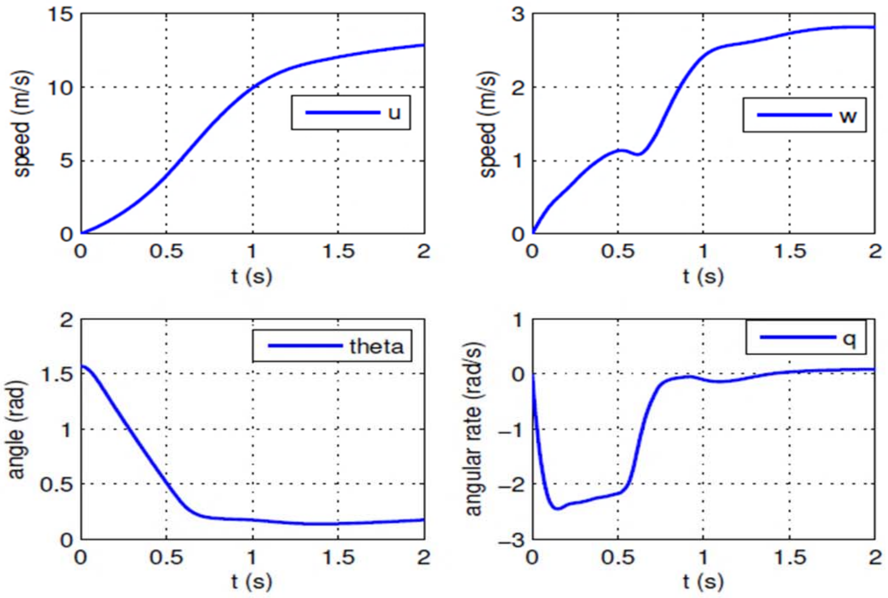
Flight control systems (transition mode)…

Системы управления полетом (переходный режим)

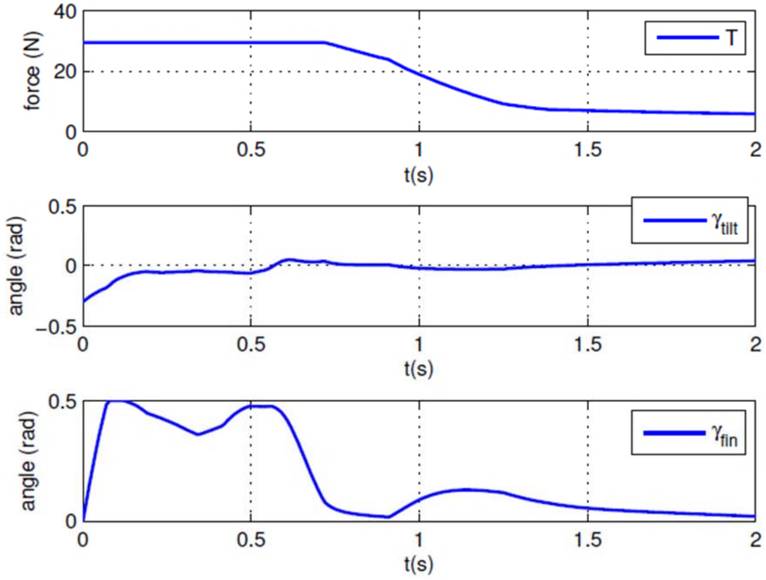
Once the $\pi^*(s)$ is obtained, the optimal control law for state x is obtained through:

1. Do volumetric interpolation so that: $\boldsymbol{x} = \sum_{m=1}^{16} w_m s_m, s_m \in S$

2. Obtain the optimal control input: $\boldsymbol{u} = \sum^{16} w_m \pi^*(s_m)$



State trajectories for forward transition



Input trajectories for forward transition

Flight control systems (transition mode)…

Системы управления полетом (переходный режим)

Once we obtain the optimal trajectory:

$$v^*(t) = (u^*(t), w^*(t), q^*(t), \theta^*(t), T_{\mathrm{u}}^*(t), T_{\mathrm{w}}^*(t), T_{\mathrm{f}}^*(t))$$

The error dynamics for tracking the trajectory is:

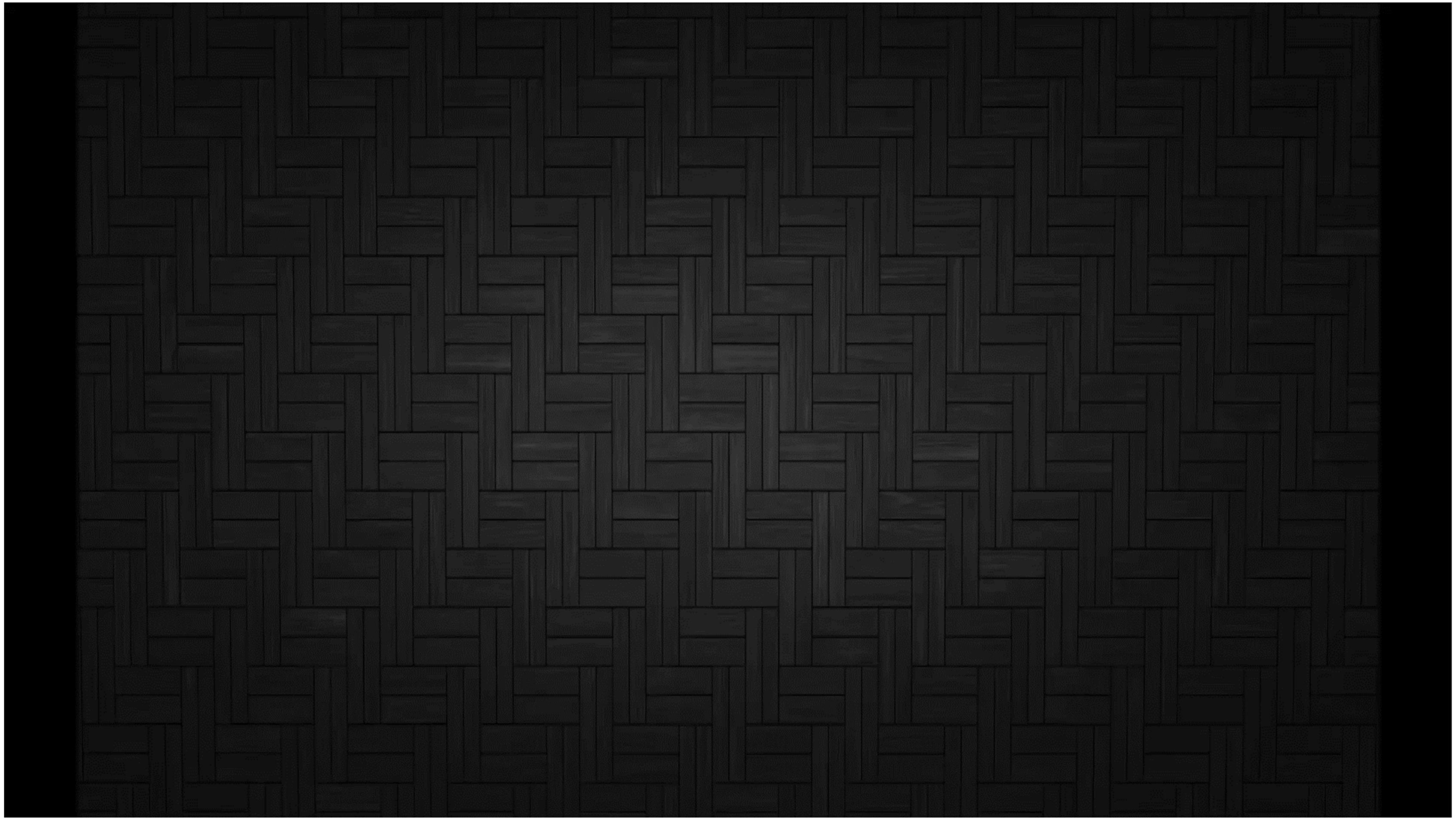
$$\begin{aligned}
\dot{\tilde{u}} &= \Psi_{\mathrm{u}}(\tilde{u}, \tilde{w}, \tilde{q}, \tilde{\theta}, t) + \tfrac{1}{m}\tilde{T}_{\mathrm{u}}, \\
\dot{\tilde{w}} &= \Psi_{\mathrm{w}}(\tilde{u}, \tilde{w}, \tilde{q}, \tilde{\theta}, t) - \tfrac{1}{m}\tilde{T}_{\mathrm{w}} + \tfrac{1}{m}\tilde{T}_{\mathrm{f}}, \\
\dot{\tilde{q}} &= \Psi_{\mathrm{q}}(\tilde{u}, \tilde{w}, \tilde{q}, \tilde{\theta}, t) + \tfrac{l_{\mathrm{m}}}{I_{\mathrm{y}}}\tilde{T}_{\mathrm{w}} + \tfrac{l_{\mathrm{t}}}{I_{\mathrm{y}}}\tilde{T}_{\mathrm{f}}, \\
\dot{\tilde{\theta}} &= \tilde{q},
\end{aligned}$$

Treat the error terms $\Psi(u), \Psi(w), \Psi(q)$ as lumped disturbances, then the nominal dynamics becomes is an LTI system and any linear control law can be designed to regulate the errors to zero.

Fully autonomous flight test…

Полностью автономные летные испытания

Auto-landing on moving platform… Автоматическая посадка на движущуюся платформу



Concluding remarks…

Заключительные замечания

We invite nonlinear systems and control experts to help tackling our problem…

ПОМОГИТЕ!

Мы приглашаем специалистов по нелинейным системам и управлению
для решения нашей проблемы.

Огромное спасибо!

Thank You!

www.bmchen.net