Recent Research Progresses in Rotorcraft Flight Dynamics and Autonomous Flight Control at KKU

2024. 02
Prof. Chang-Joo Kim (Konkuk University, Seoul, Korea)

Recent Research Progresses in Rotorcraft Flight Dynamics and Autonomous Flight Control at KKU

Part 1: Rotorcraft Flight Dynamics

2024. 02

Prof. Chang-Joo Kim (Konkuk University, Seoul, Korea)

Contents

Flight Dynamic Model (HETLAS)

Objectives of HETLAS

HETLAS: Helicopter Trim, Linearization, And Simulation

Primary Tool for Rotorcraft Design/Development using Following Functions and Applications

Flight Dynamic Analysis
- Trim
- Linearization
- Simulation

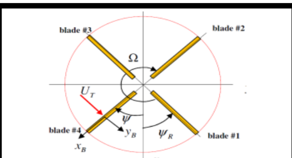
Applications for MTE Analysis / FCS Development
- Inverse Simulation
- Nonlinear Optimal Control Analysis
- Design/Evaluation of Flight Control Laws

Evolution of Rotorcraft Development

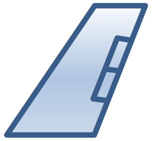
1st Generation ('45~'60) R-5 / H-21	2nd Gen.('60~'75) CH-47A,UH-1, CH-53	3rd Gen. ('75~'90) UH-60A AH-64A	3+Gen.('90~'15) V-22, UH-60M, AH-64E CH-47-II, A/UH-1Y/Z	4th Gen.('15~'20) RAH-66 CH-53K	5th Gen.('30~) FVL, MUX FUAS
• <100 knots • Reciprocating EG • Mechanical FCS • Wooden Blade • Low Survivability	• <130 knots • Turboshaft EG • Mechanical AFCS • Metal Blade • Passive Survivability	• >150 knots • Power Increased • Mechanical AFCS • Composite Blade • Active • Crashworthiness	• >150 knots • High Efficiency • FBW FCS • Composite Blade • Sensor Fusion • Low RF/Noise	• >170 knots • Extreme Efficiency • Digital FBW FCS • Composite Blade/ Structure • Active Crashworthy	•>200 knots •Long End./Range •Ind. Blade Control •Compounding •Adaptive FCS •Autonomous FCS

Flight Dynamic Model (HETLAS)

Modeling Concept: Component-Based / Physical-Law-Based Approach



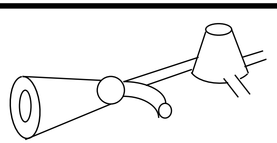
Rotor



Wing/Control surf.

Fuselage

Propeller/Ducted



Power plant

Flight control system

External forces and moments

Force Summation

$$X = X_{Rotor} + X_{Wing} + X_{Fus} + \cdots$$
$$Y = Y_{Rotor} + Y_{Wing} + Y_{Fus} + \cdots$$
$$Z = Z_{Rotor} + Z_{Wing} + Z_{Fus} + \cdots$$

Gravity forces

Atmospheric conditions
Turbulence model
Wind shear model
Etc.

Moment Summation

$$L = L_{Rotor} + L_{Wing} + L_{Fus} + \cdots$$
$$M = M_{Rotor} + M_{Wing} + M_{Fus} + \cdots$$
$$N = N_{Rotor} + N_{Wing} + N_{Fus} + \cdots$$

Flight dynamic equations

Linear motion equations

$$\dot{u} + (wq - vr) + g\sin\theta = X/m$$
$$\dot{v} + (ur - wp) + g\sin\phi\cos\theta = Y/m$$
$$\dot{w} + (vp - uq) + g\cos\phi\cos\theta = Z/m$$

Attitude kinematics

- Euler attitude angles
- Quaternion
- Finite rotation angles

Position kinematics

Angular motion equations

$$I_{xx}\dot{p} - (I_{yy} - I_{zz})qr - I_{xz}(\dot{r} + pq) = L$$
$$I_{yy}\dot{q} + (I_{xx} - I_{zz})pr + I_{xz}(p^2 - r^2) = M$$
$$I_{zz}\dot{r} - (I_{xx} - I_{yy})pq - I_{xz}(\dot{p} + qr) = N$$

Generalized Rotor Model

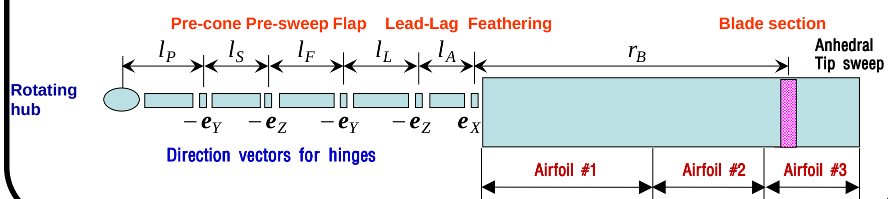
Unified Rotor Models

Dynamics
- Rotor
- or Propeller

Aerodynamics
- Blade element method
- Dynamic inflow

Rotor-components modeling requirements

- Any configurations of the rotor/propeller can be handled.
- Flapping and lagging motions are independently adopted (Ex. No dynamics: ABC rotor blade)
- Interference among rotors using empirical data.
- Both Pitch controls and RPM control are selectable.
- Number of blades and airfoils is not limited.
- Input data of each blade are received from external files
- Various inflow models
- High-fidelity rotor modeling techniques
- General rotor orientations
- General directions of rotor rotation (CW, CCW)



Diversity in Rotor Configurations is reflected in selecting Requirements for Rotor Model

Rotor Type		Dynamics	Control	Location	Orientation
Propeller		No dynamics	collective or RPM	Front (Pull) or Rear (Pusher)	90 deg FWD tilt
Rotor	Conventional main rotor	Flap / Lag / RPM	Collective 2 cyclic pitches	Top/center	Vertical (reference) Small FWD tilt
Rotor	Conventional tail rotor	Flap / RPM (MR dependent)	collective	Rear	±90 deg sideward tilt with small cant angle
Rotor	Gimbal/Teetering main rotor	Flap Gimbal	Collective 2 cyclic pitches	Top/center	Vertical (reference) Small FWD tilt
Rotor	Gimbal/Teetering tail rotor	Flap Gimbal	Collective	Rear	±90 deg sideward tilt with small cant angle
Rotor	ABC (Advanced Blade Concept)	No dynamics	Collective 2 cyclic pitches	Top/center	Vertical Small FWD tilt
Rotor	Ducted	No dynamics	Collective (thrust vectoring)	Design dependent	Design dependent

Flight Dynamic Model (HETLAS)

Generalized Wing Model (for Rotorcrafts with Lift Compounding)

Unified Wing Models

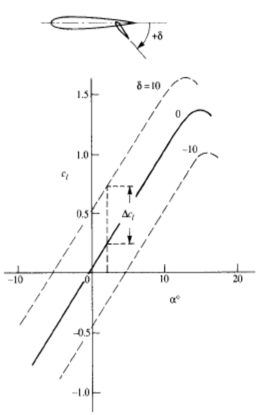
Wing + Control Surface

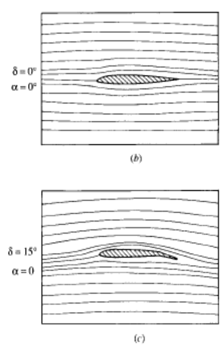
- Strip Theory
- Airfoil aerodynamic data Table Lookup
- Lifting Line Theory
- Biot-Savart Law
- Lift Increment Estimation

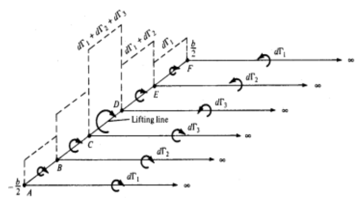
Diversity in Wing Configurations is reflected in selecting Requirements for Wing Model

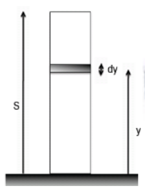
Wing-components modelling requirements

- Orientation of each wing can be defined with respect to the reference starboard main wing
- Many control surfaces can be allocated to the wing, some of which have the right or reversed deflection angles
- Airfoil can have the convectional orientation or the reversed one





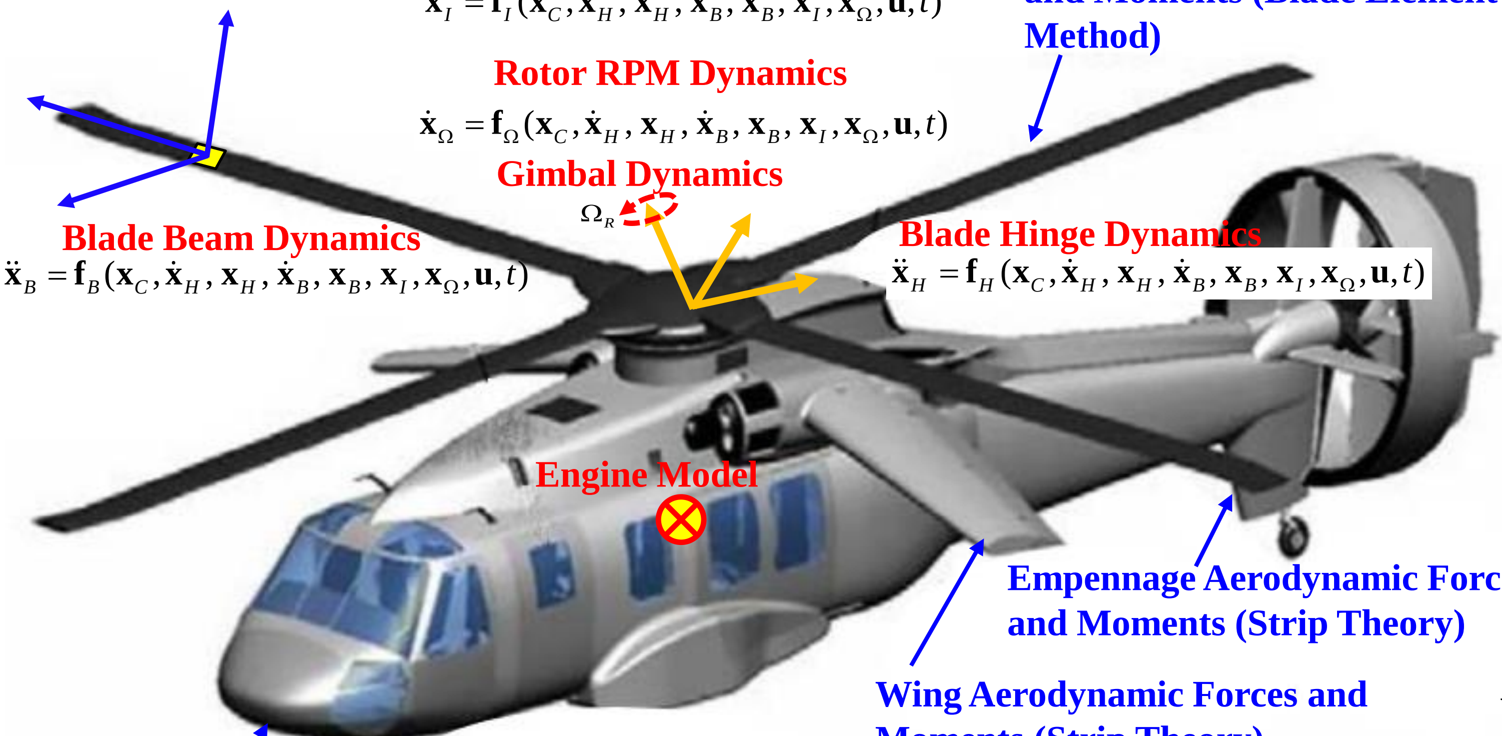




Wing Type	Control surfaces	Location	Orientation
Conventional main wing (starboard)	Aileron / flap (2 control surfaces)	Fuselage center	Horizontal with dihedral, sweep, and twist distribution along the span
Conventional main wing (port)	Aileron (reversely coupled) Flap (rightly coupled)	Fuselage center	Symmetric in x-z plane with respect to starboard side main wing
Conventional horizontal stabilizer	Elevator (1 control surfaces, independent)	Rear fuselage	Same as the starboard main wing but airfoil may be upside down orientation with specified attachment angle
Conventional vertical stabilizer	Rudder (1 control surfaces, independent)	Rear fuselage	±90 deg upward tilt from the reference airfoil may be upside down orientation with specified attachment angle
Wing with end plate	As specified	As specified	As specified
Others	As specified	As specified	As specified

Flight Dynamic Model (HETLAS)

Composition of Future Flight Dynamic Model (including Elastic Beam Model)



Rotor Inflow Dynamics

$$\dot{\mathbf{x}}_I = \mathbf{f}_I(\mathbf{x}_C, \dot{\mathbf{x}}_H, \mathbf{x}_H, \dot{\mathbf{x}}_B, \mathbf{x}_B, \mathbf{x}_I, \mathbf{x}_\Omega, \mathbf{u}, t)$$

Rotor RPM Dynamics

$$\dot{\mathbf{x}}_\Omega = \mathbf{f}_\Omega(\mathbf{x}_C, \dot{\mathbf{x}}_H, \mathbf{x}_H, \dot{\mathbf{x}}_B, \mathbf{x}_B, \mathbf{x}_I, \mathbf{x}_\Omega, \mathbf{u}, t)$$

Gimbal Dynamics

Ω_R

Blade Aerodynamic Forces and Moments (Blade Element Method)

Blade Beam Dynamics

$$\ddot{\mathbf{x}}_B = \mathbf{f}_B(\mathbf{x}_C, \dot{\mathbf{x}}_H, \mathbf{x}_H, \dot{\mathbf{x}}_B, \mathbf{x}_B, \mathbf{x}_I, \mathbf{x}_\Omega, \mathbf{u}, t)$$

Blade Hinge Dynamics

$$\ddot{\mathbf{x}}_H = \mathbf{f}_H(\mathbf{x}_C, \dot{\mathbf{x}}_H, \mathbf{x}_H, \dot{\mathbf{x}}_B, \mathbf{x}_B, \mathbf{x}_I, \mathbf{x}_\Omega, \mathbf{u}, t)$$

Engine Model

Empennage Aerodynamic Forces and Moments (Strip Theory)

Wing Aerodynamic Forces and Moments (Strip Theory)

Aircraft Rigid-Body Dynamics

$$\dot{\mathbf{x}}_C = \mathbf{f}_C(\mathbf{x}_C, \dot{\mathbf{x}}_H, \mathbf{x}_H, \dot{\mathbf{x}}_B, \mathbf{x}_B, \mathbf{x}_I, \mathbf{x}_\Omega, \mathbf{u}, t)$$

Fuselage Aerodynamic Forces and Moments (Tabulated Aerodynamic Coefficients)

$$\boxed{\begin{aligned} \ddot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \\ \mathbf{x} &= (\mathbf{x}_C^T, \dot{\mathbf{x}}_H^T, \mathbf{x}_H^T, \dot{\mathbf{x}}_B^T, \mathbf{x}_B^T, \mathbf{x}_I^T, \mathbf{x}_\Omega^T) \end{aligned}}$$

$$\mathbf{x}_C = \begin{pmatrix} u \\ v \\ w \\ p \\ q \\ r \\ \phi \\ \theta \\ \psi \end{pmatrix} \quad \mathbf{x}_H = \begin{pmatrix} \beta \\ \zeta \end{pmatrix} \quad \mathbf{x}_B = \begin{pmatrix} \mathbf{u}_B \\ \boldsymbol{\theta}_B \end{pmatrix} \quad \mathbf{x}_I = \begin{pmatrix} \mathbf{v}_i \\ \boldsymbol{\chi}_i \end{pmatrix} \quad \mathbf{x}_\Omega = \Omega$$

$$\mathbf{u}_B = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} \quad \mathbf{v}_i = \begin{pmatrix} v_{i0} \\ v_{i1C} \\ v_{i1S} \end{pmatrix}$$

$$\boldsymbol{\theta}_B = \begin{pmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{pmatrix} \quad \boldsymbol{\chi}_i = \begin{pmatrix} \chi_0 \\ \chi_{1C} \\ \chi_{01S} \\ \kappa_w \end{pmatrix}$$

Flight Dynamic Analysis Model (HETLAS)

Trim Analysis Model

Trim Flight Category (for Trim Kinematical Equations)

- Rectilinear Flight : hover, vertical flight, side and rearward Flight, forward flight with sideslip and climb angle
- Turning Flight : coordinated/uncoordinated turn with flight path angle (Helical Turn)
- Auto-rotational Descent
- Bank-zero Trim (for Pilot's Attributes)
- Pull-up (instantaneous)
- Push-over (instantaneous)

Trim Methodology

- Harmonic Balance Method
- Periodic-Trimming Algorithm (PTA)
- Partial Periodic Trim Algorithm (PPTA)

Trim Equation (NAEs) Solvers

- Standard Newton Methods
- Quasi-Newton Methods
  - ✓ Broyden's good method
  - ✓ Broyden's bad method
  - ✓ Greensradt's 1st and 2nd method
  - ✓ Thomas optimal method
  - ✓ Martinez's column-updating method
  - ✓ Etc.

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{B}_k^{-1}\mathbf{f}(\mathbf{x}_k), \quad \mathbf{B}_k \approx \mathbf{J}(\mathbf{x}_k) = \frac{d\mathbf{f}(\mathbf{x}_k)}{d\mathbf{x}}$$

Linearization Analysis Model

Numerical Jacobean approximation using the Finite Difference Formula

Motion equations $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t)$ Trim solution $\boxed{\mathbf{f}_{Trim} = \mathbf{f}_{Trim}(\mathbf{x}_T, \mathbf{u}_T) = \mathbf{0}}$

Derivation of Linear Model @ Trim Conditions $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$

$$\nabla_{x_j}\mathbf{f}(\mathbf{x},\mathbf{u}) = \frac{\mathbf{f}(x_1,\cdots,x_j+\Delta x_j,\cdots,x_n,\mathbf{u}) - \mathbf{f}(x_1,\cdots,x_j-\Delta x_j,\cdots,x_n,\mathbf{u})}{2\Delta x_j}, \quad \mathbf{x},\mathbf{f} \in R^n, \quad \mathbf{u} \in R^m$$

$$\nabla_{\mathbf{x}}\mathbf{f}(\mathbf{x},\mathbf{u}) = \left(\nabla_{x_1}\mathbf{f}(\mathbf{x},\mathbf{u}),\cdots,\nabla_{x_n}\mathbf{f}(\mathbf{x},\mathbf{u})\right) = \mathbf{A} \in R^{n\times n}$$

$$\nabla_{\mathbf{u}}\mathbf{f}(\mathbf{x},\mathbf{u}) = \left(\nabla_{u_1}\mathbf{f}(\mathbf{x},\mathbf{u}),\cdots,\nabla_{u_m}\mathbf{f}(\mathbf{x},\mathbf{u})\right) = \mathbf{B} \in R^{n\times m}$$

Reduced Order Model : Low-Order Equivalent (LOE) Model

Truncation method by ignoring the inter-axis coupling

$$\begin{pmatrix}\dot{\mathbf{x}}_1\\ \dot{\mathbf{x}}_2\end{pmatrix} = \begin{pmatrix}\mathbf{A}_{11} & \mathbf{A}_{12}\\ \mathbf{A}_{21} & \mathbf{A}_{22}\end{pmatrix}\begin{pmatrix}\mathbf{x}_1\\ \mathbf{x}_2\end{pmatrix} + \begin{pmatrix}\mathbf{B}_{11} & \mathbf{B}_{12}\\ \mathbf{B}_{21} & \mathbf{B}_{22}\end{pmatrix}\begin{pmatrix}\mathbf{u}_1\\ \mathbf{u}_2\end{pmatrix} \rightarrow \boxed{\begin{pmatrix}\dot{\mathbf{x}}_1\\ \dot{\mathbf{x}}_2\end{pmatrix} \cong \begin{pmatrix}\mathbf{A}_{11} & 0\\ 0 & \mathbf{A}_{22}\end{pmatrix}\begin{pmatrix}\mathbf{x}_1\\ \mathbf{x}_2\end{pmatrix} + \begin{pmatrix}\mathbf{B}_{11} & 0\\ 0 & \mathbf{B}_{22}\end{pmatrix}\begin{pmatrix}\mathbf{u}_1\\ \mathbf{u}_2\end{pmatrix}}$$

Residualization (time-scale separation) method

$$\begin{pmatrix}\dot{\mathbf{x}}_1\\ 0\end{pmatrix} = \begin{pmatrix}\mathbf{A}_{11} & \mathbf{A}_{12}\\ \mathbf{A}_{21} & \mathbf{A}_{22}\end{pmatrix}\begin{pmatrix}\mathbf{x}_1\\ \mathbf{x}_2\end{pmatrix} + \begin{pmatrix}\mathbf{B}_1\\ \mathbf{B}_2\end{pmatrix}\mathbf{u}$$

$$\rightarrow \dot{\mathbf{x}}_1 = \left\{\mathbf{A}_{11} - \mathbf{A}_{12}(\mathbf{A}_{22})^{-1}\mathbf{A}_{21}\right\}\mathbf{x}_1 + \left\{\mathbf{B}_1 - \mathbf{A}_{12}(\mathbf{A}_{22})^{-1}\mathbf{B}_2\right\}\mathbf{u}$$

$$\boxed{\begin{aligned}\mathbf{x}_1 &= \mathbf{x}_R\\ \mathbf{x}_2 &= (\mathbf{x}_F, \mathbf{x}_L, \mathbf{x}_I, \mathbf{x}_\Omega)^T\end{aligned}}$$

→ Residualization method is better suit for rotorcrafts due to high inter-axis coupling

Simulation Analysis Model

Standard Explicit Time Integrator

- RTAM-3 : 3rd order Real-Time Adams-Moulton integrator
- RK-4 : 4th order Runge-Kutta time integrator
- RKF-45 : 5th order Runge-Kutta time integrator with step size control

Standard Implicit Time Integrator

- Crank-Nicolson Algorithm : 2nd Order
- Backward Difference Method : 3rd/ 4th Order Algorithm

Pseudo Spectral (PS) Time Integrator coupled with Piccard Method

Motion equations $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t), \quad \mathbf{x}(0) = \mathbf{x}_o$

Nonlinear Algebraic Equations (NAEs)

$$0 = \mathbf{x}_0 + \frac{h}{2}\sum_{k=0}^{k=N} I_{jk}\mathbf{f}_k - \mathbf{x}_j \quad (j = 1, 2, \cdots, N) \leftarrow \mathbf{x}_j = \mathbf{x}_0 + \frac{h}{2}\sum_{k=0}^{k=N} I_{jk}\mathbf{f}_k$$

Applications of Piccard Fixed Point Iterative Method

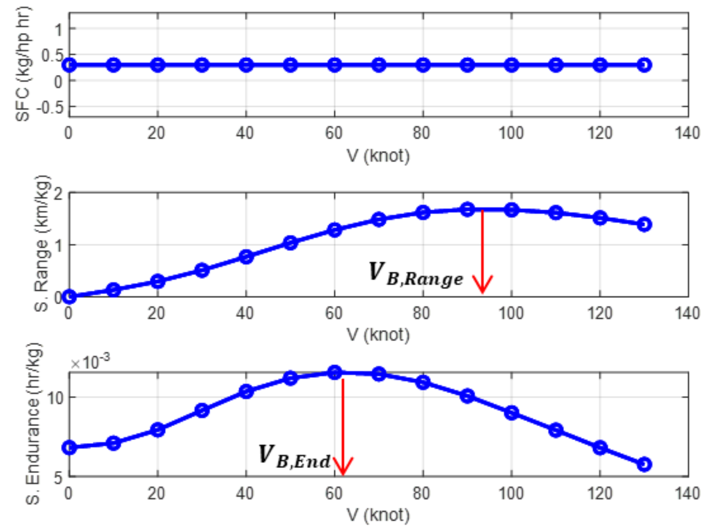
$$\mathbf{x}_j^{(iter+1)} = \mathbf{x}_0 + \frac{h}{2}\sum_{k=0}^{k=N} I_{jk}\mathbf{f}_k^{(iter)} \leftarrow \mathbf{f}_k^{(iter)} = \mathbf{f}(\mathbf{x}_k^{(iter)}, t_k)$$

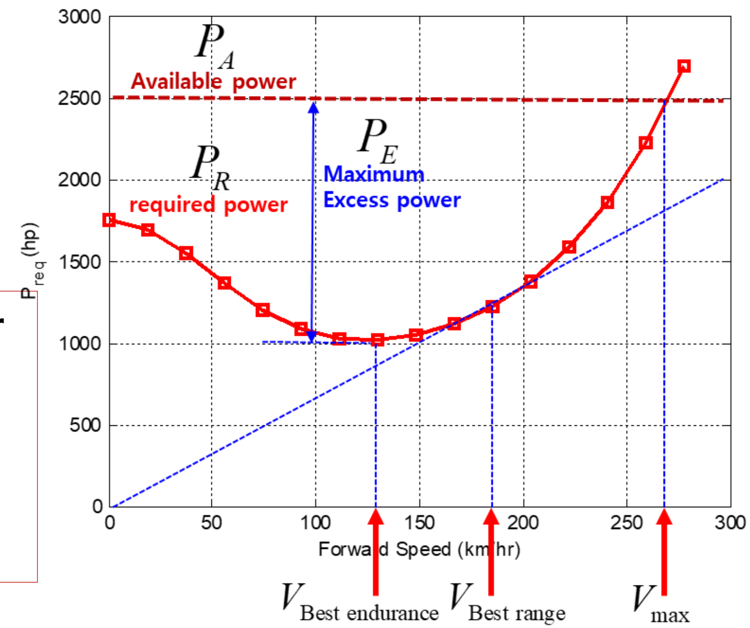
Flight Dynamic Analysis Model (HETLAS)

Point Performance (Fuel Independent) Analysis Model

- **Hovering & Vertical Flight Performance**
  - OGE Hovering Limits @MCP, TOP
  - IGE Hovering Limits @MCP, TOP
  - Max. Vertical Climb Rate @MCP, TOP

- **Forward Flight Performance**
  - Max. Climb Rate @MCP
  - OEI Service Ceiling @MCP
  - Max. Cruise Speed @ MCP
  - Never Exceed Speed Limits Vne
  - Flight Envelope
  - Max. Load Factor
  - Service Ceiling (max. RoC<100 ft/min)
  - Absolute Ceiling (max. RoC= 0 ft/min)





Engine Power	MCP: Maximum Continuous Power TOP: Take-Off Power
Engine Failure	AEO: All Engine Inoperative OEI : One Engine Inoperative
Ground Effect	IGE : In-Ground Effect OGE: Out-of-Ground Effect

Flight Dynamic Analysis Model (HETLAS)

Point Performance (Fuel Independent) Analysis Model

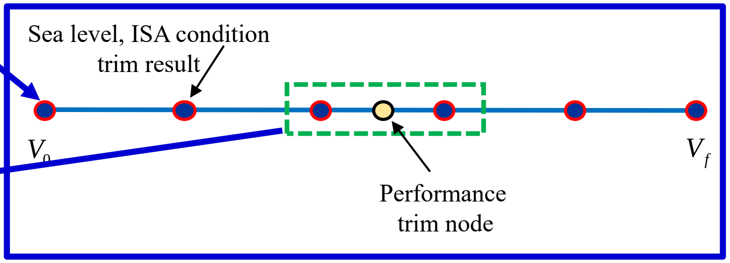
❑ Computer-Model Procedures for Point Performance Analysis

```
Subroutine Point Performance Analysis Routine                          I: Input, O: Output
|
|       Call Aircraft Technical Data Preprocessing
|       Contain
|       |       Call Aerodynamic Table Reading
|       |       Call Aircraft Configuration Data
|       |       Call Analysis Parameter Setting
|       |       Call Engine Data Preprocessing
|       End Contain
|
|       Call Standard Day Performance Analysis (I: Analysis Option, O: Trim Results)
|       Contain
|       |       Call Atmospheric condition calculation (I: Analysis Option, O: Sea Level Condition)
|       |       Call Trim Analysis (I: Sea Level Condition, O: Trim Results)
|       End Contain
|
|       Call Performance Routine Main (I: Analysis Option)
|       Contain
|       |       Do I = 1, Number of Temperatures
|       |       |       Do J=1, Number of Weight
|       |       |       |       Do K=1, Number of Altitudes
|       |       |       |       |       Call Atmospheric condition calculation
|       |       |       |       |
|       |       |       |       |       Do L=1, Number of Velocities
|       |       |       |       |       |       Call Trim Mover    (I: Swept Flight condition)
|       |       |       |       |       |                          (O: Near Trim Node)
|       |       |       |       |       |       Call Trim Analysis (I: Near Trim Node)
|       |       |       |       |       |                          (O: Trim Results)
|       |       |       |       |       |       Call Engine Module (I: Trim Results, O: SFC)
|       |       |       |       |       |       Call Performance Analysis (I: SFC)
|       |       |       |       |       |               (O: Performance for L-th velocity)
|       |       |       |       |       End Do
|       |       |       |       |
|       |       |       |       |       Call Lagrange Interpolation   (I: Performance data {1: L})
|       |       |       |       |                                     (O: Interpolated Function)
|       |       |       |       |       Call Find Maximum Parameter (I: Interpolated Function)
|       |       |       |       |                                     (O: Maximum Values)
|       |       |       |       End Do
|       |       |       End Do
|       |       End Do
|       End Contain
|
End Subroutine
```

1. Aircraft Data Processing & Analysis Option Selection

2. Standard Day Performance Analysis for Initial Guess (Generate Initial Values and Save it for 'Trim Mover')

Trim Mover Functions



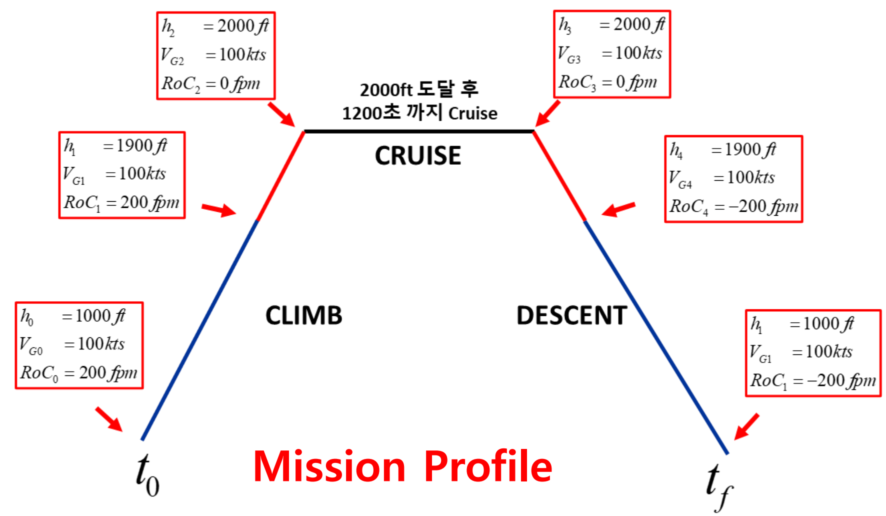
3. Performance Analysis with WAT Sweep (WAT: Weight, Altitude, Temperature)

Trim Mover Functions are developed and implemented for Robustness in Analysis

Flight Dynamic Analysis Model (HETLAS)

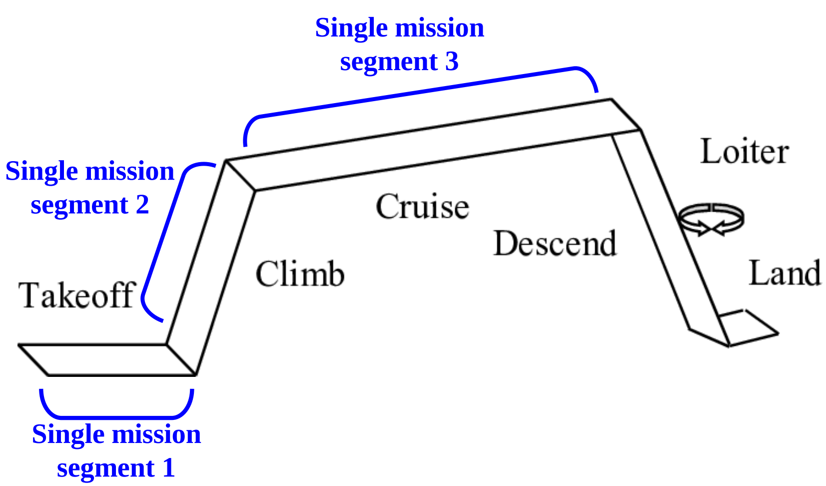
Mission Performance (Fuel Dependent) Analysis Model: Mission Segments

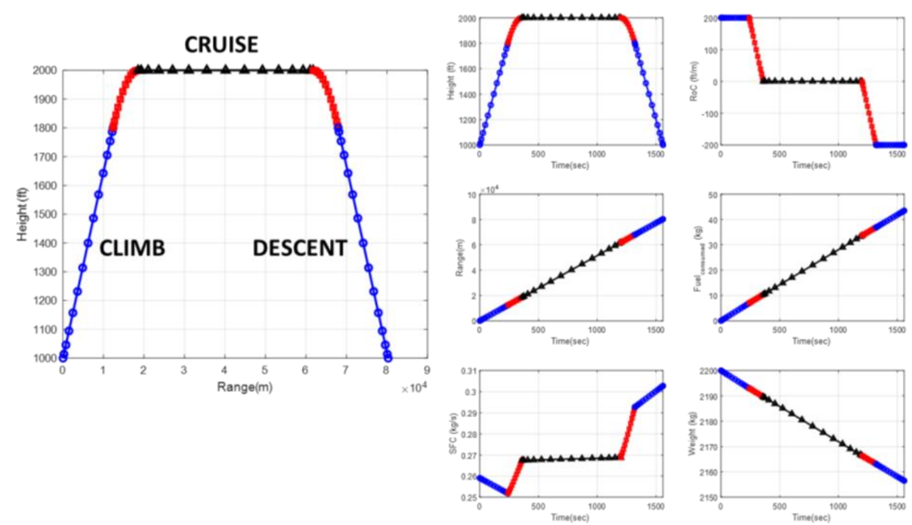
- Range
- Endurance
- Payload-Range Performance
- Payload-Endurance Performance
- Max. range with zero payload



Mission Profile

Mission Profile (Example)





Mission Performance Analysis Results

Mission Performance (Fuel Dependent) Analysis Model: Mission Segments

- **Definition of Mission Segments using Way-point Data**

$$\left\{\left(t_j, h_j, V_{G,j}, V_{ROC,j}\right)\right\}_{j=0}^{j=N_{WP}}$$

Data for height, ground speed, and rate of climb

- **Trajectory Generation using spline interpolation of** h, V_G, V_{ROC}

- **Time integration along the generated trajectory to get converged solutions of coupled mission-performance equations using PS-integrator**

$$\boxed{\begin{aligned} \frac{dm}{dt} &= -SFC \times P \\ \frac{dh}{dt} &= V_{RoC} \\ \frac{dR}{dt} &= |V_G| \end{aligned}} \longrightarrow \boxed{\begin{aligned} m(t) &= m(t_0) - \int_{t_0}^{t_f} (SFC \times P) dt \\ h(t) &= h(t_0) + \int_{t_0}^{t_f} V_{RoC} dt \\ R(t) &= R(t_0) + \int_{t_0}^{t_f} |V_G| dt \end{aligned}}$$

Flight Dynamic Analysis Model (HETLAS)

KU 建國大學校 KONKUK UNIV.

Mission Performance (Fuel Dependent) Analysis Model: Mission Segments

❑ Computer-Model Procedures of Mission Performance Analysis

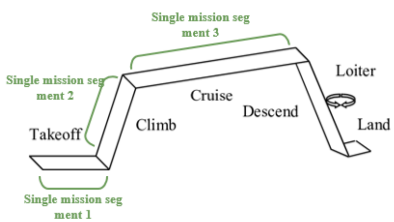
```
Subroutine Mission Performance Analysis Routine                                I: Input, O: Output
|
|       Call Aircraft Technical Data Preprocessing
|       Contain
|       |       Call Aerodynamic Table Reading
|       |       Call Aircraft Configuration Data
|       |       Call Analysis Parameter Setting
|       |       Call Engine Data Preprocessing
|       End Contain
|
|       Call Mission Segment Definition        (I: Number of Segment)
|                                              (O: Mission Segment {1: Number of Segment})
|
|       Do I=1, Number of Segment
|       |       If I=1 Then Weight initialization
|       |       Else initialize with previous segment's final node
|       |
|       |       Call Gaussian Quadrature Waypoint Generator    (I: Mission Segments {I})
|       |                                                      (O: Waypoints for Mission Segment {I})
|       |       Call linear interpolation      (I: Velocity, Height of initial and final node)
|       |                                      (O: Waypoint information for Mission Segment {I})
|       |
|       |
|       |       Do J=1, Number of Max Iteration
|       |       |       Call Trim Analysis (I: Waypoint (WP) information, O: Trim Results)
|       |       |       Call Engine Module (I: Trim Results, O: SFC)
|       |       |
|       |       |       Do K=2, Number of Waypoint in Segment
|       |       |       |       Call Waypoint Performance Calculation (I: Trim Results, SFC)
|       |       |       |               (O: Derivatives of weight, height, range for J-th waypoint)
|       |       |       End Do
|       |       |
|       |       |       Call Pseudo Spectral integrator (I: Derivatives of weight, height, range |Segment
|       |       |                       (O: Updated weight, height, range for all WP)
|       |       |
|       |       |       Do K=2, Number of Waypoint in Segment
|       |       |       |       Error = Error + 2 norm of weight, rate of climb, velocity different)
|       |       |       End Do
|       |       |
|       |       |       If Error < Tolerance Then Stop Current Segment Analysis, Go to Next Segmen
|       |       End Do
|       |
|       End Do
End Subroutine
```

Performance Analysis for Entire Mission Profiles

1. Aircraft Data Processing & Analysis Option Selection

2. Define Mission Profiles and Divide into Segments

3. Mission Segment Generation for Actual Analysis



$$\phi_j(\tau) = \frac{g_{LGL}(\tau)}{(\tau - \tau_j)g'_{LGL}(\tau_j)}$$

4. Performance Analysis for i-th Segment

$$\mathbf{x}(\tau) \approx \sum_{k=0}^{N} \phi_k(\tau)\mathbf{x}(\tau_k)$$

$$\mathbf{f}(\mathbf{x}(t),t) \approx \sum_{k=1}^{N} \phi_k(\tau)\mathbf{f}(\mathbf{x}_k, t_k)$$

$$\mathbf{x} = [W_F \quad h \quad R]^T$$

$$\mathbf{x}_j^{i+1} = \mathbf{x}_0 + \frac{t_f - t_0}{2} \sum_{k=1}^{N} I_{j,k}\mathbf{f}_k^i$$

$$\mathbf{f}_k^i = \mathbf{f}(\mathbf{x}_k^i)$$

Iterative Pseudo-Spectral Integrator

Validation of Flight Dynamic Model (HETLAS)

Validation Examples

❑ Validation of HETLAS: Example Rotorcrafts

Reference Helicopter

Bo-105

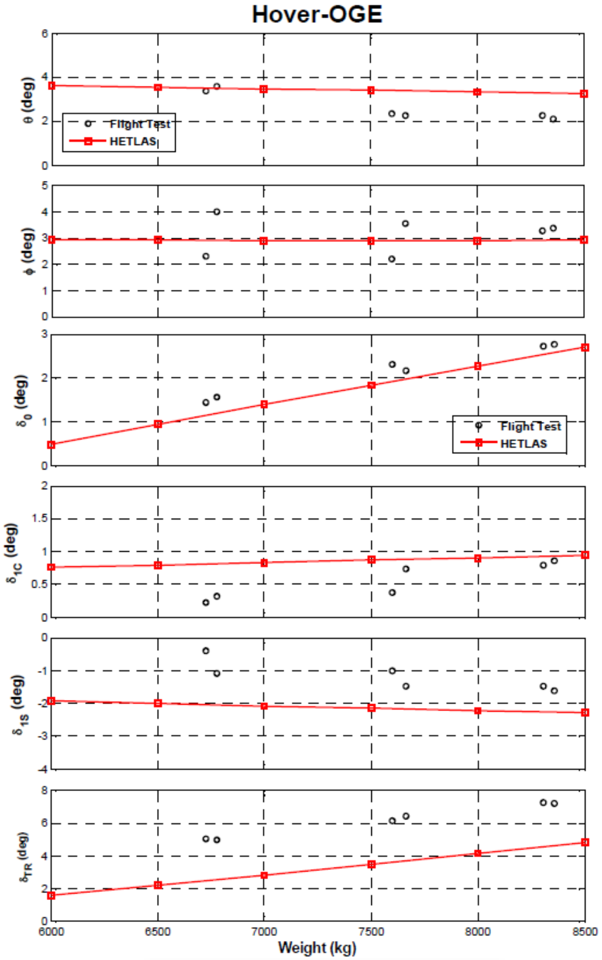
V&V: Comparison with Flight Test
Criteria: 1) FAA AC-120-63
2) GENHEL (Sikorsky 社)
3) Boeing

V&V: Using Ref. (Flight test/Analysis)
Criteria: FAA AC-120-63
Ref.: 1) AGARD GARTEUR Report
2) Published Papers

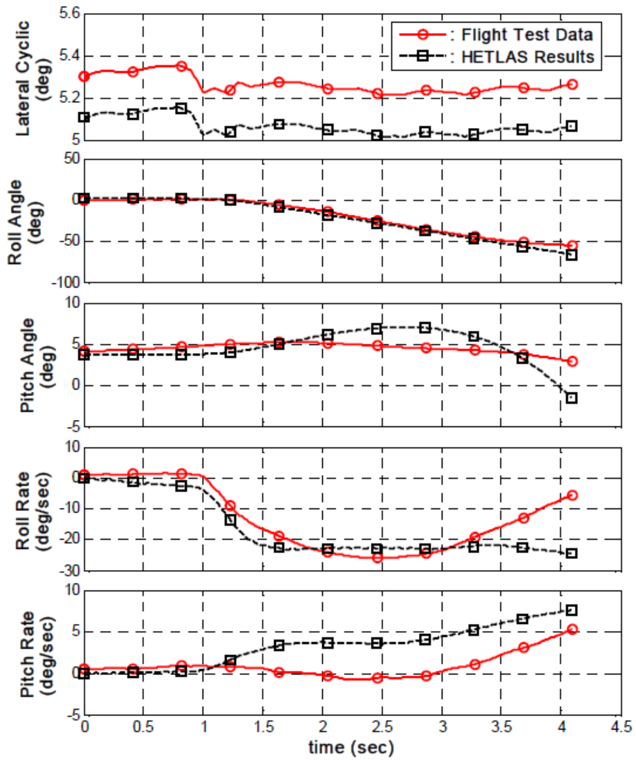
Validation of Flight Dynamic Model (HETLAS)

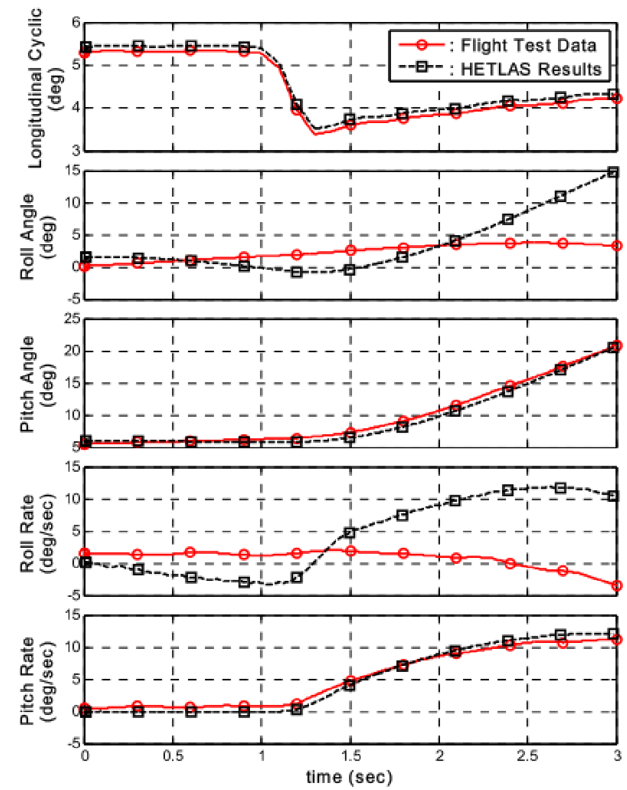
Comparison of Trim and Control Response for Reference Helicopter

C.-J. Kim, C.-D. Yang, C. Kim, Model-Fidelity Validation of the Helicopter Flight Dynamic Analysis Program, HETLAS, 213 (2019) 210–213.



Trim Analysis



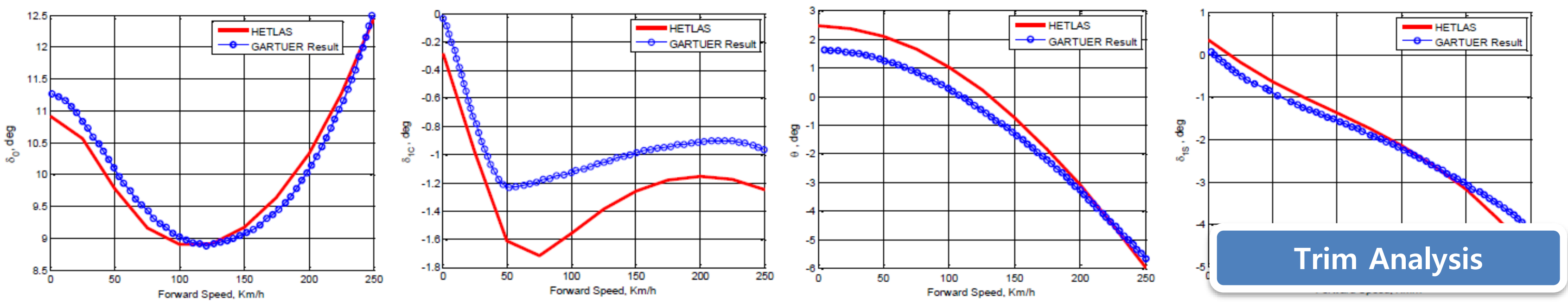


Simulation Results

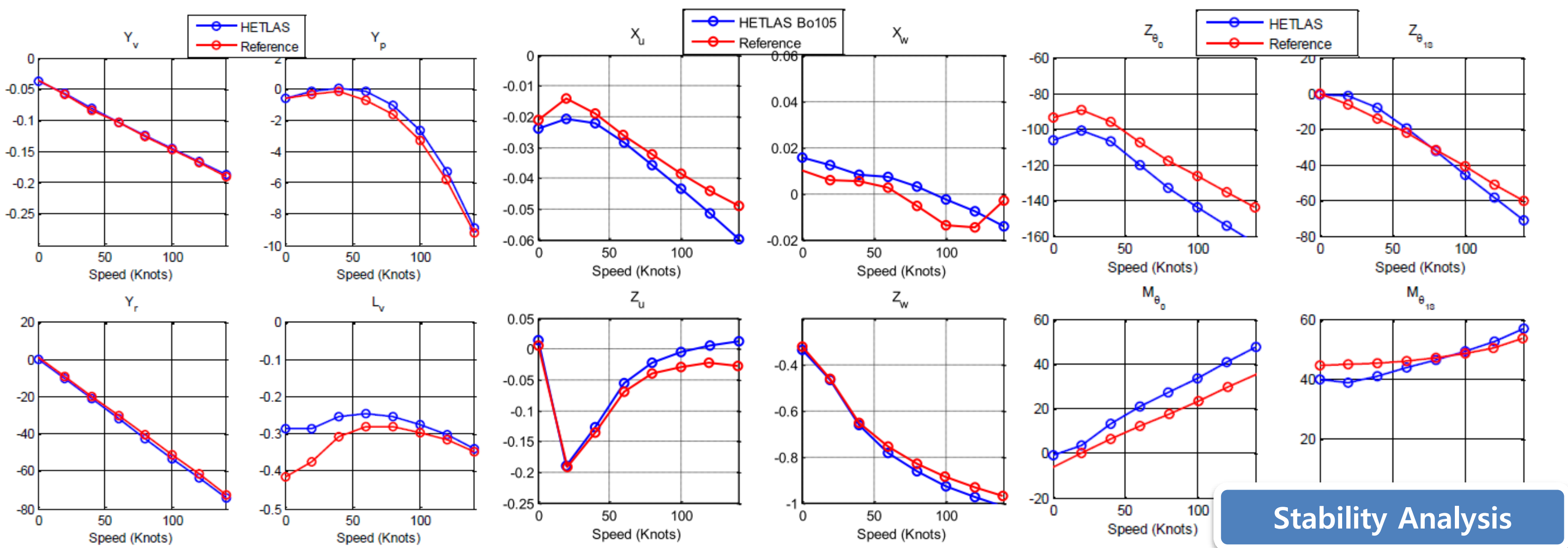
Validation of Flight Dynamic Model (HETLAS)

Comparison of Trim Results for Bo-105

C.-J. Kim, K.-C. Shin, C. Yang, I.-J. Cho, C.-D. and Yun, Y.-H., Kim, C.-J., Shin, K.-C., Yang, I.-J. Cho, Interface features of flight dynamic analysis program, HETLAS, for the development of helicopter FBW system, in: 1st Asian Australian Rotorcraft Forum and Exhibition 2012, 2012: pp. 12–15.



Trim Analysis

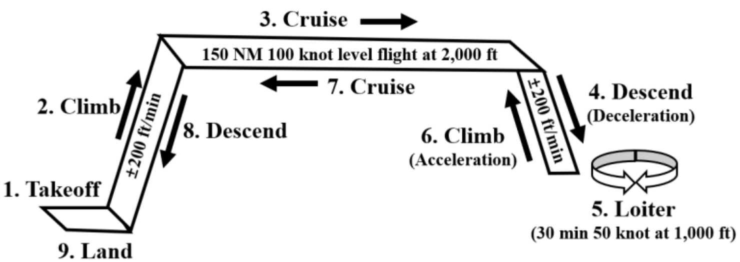


Stability Analysis

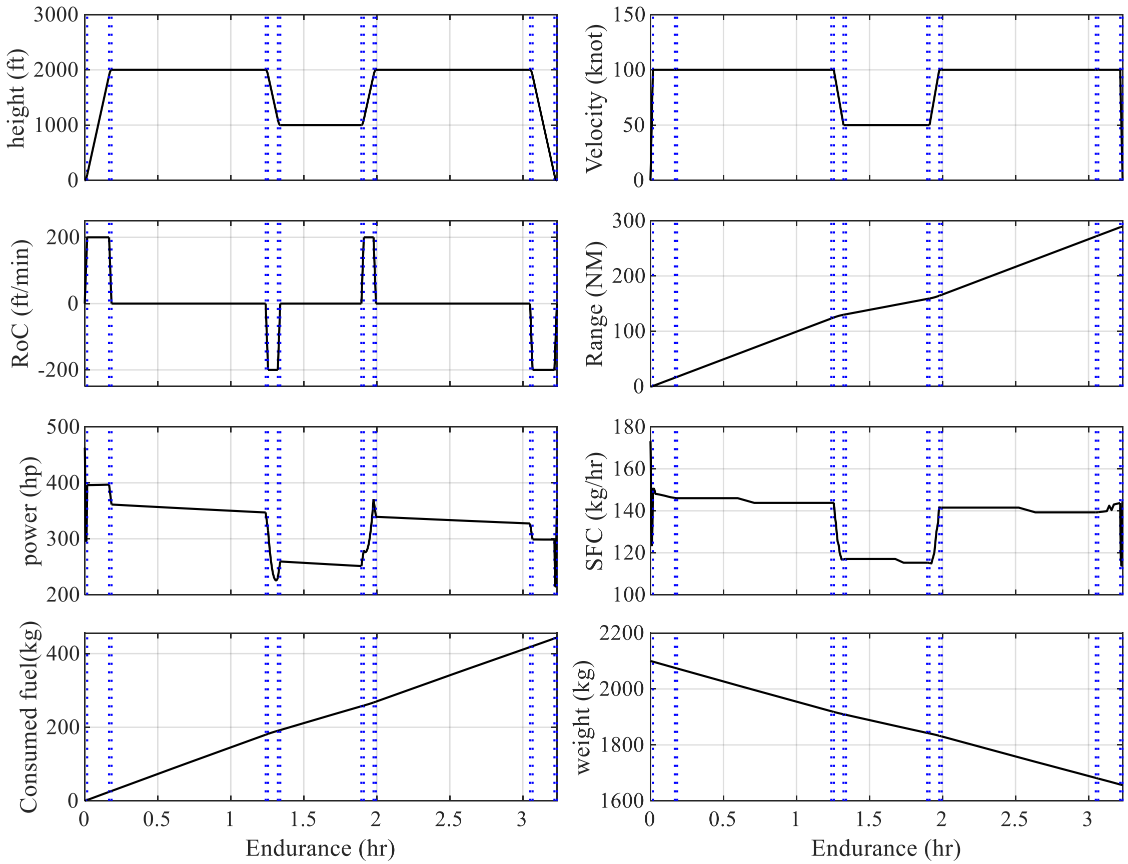
Validation of Flight Dynamic Model (HETLAS)

Comparison of Mission-Performance Analysis Results for Bo-105

J. An, Y.-S. Choi, I.-R. Lee, M. Lim, and C.-J. Kim, "Performance Analysis of a Conceptual Urban Air Mobility Configuration Using High-Fidelity Rotorcraft Flight Dynamic Model," International Journal of Aeronautical and Space Sciences, Jul. 2023, doi: 10.1007/s42405-023-00610-7.



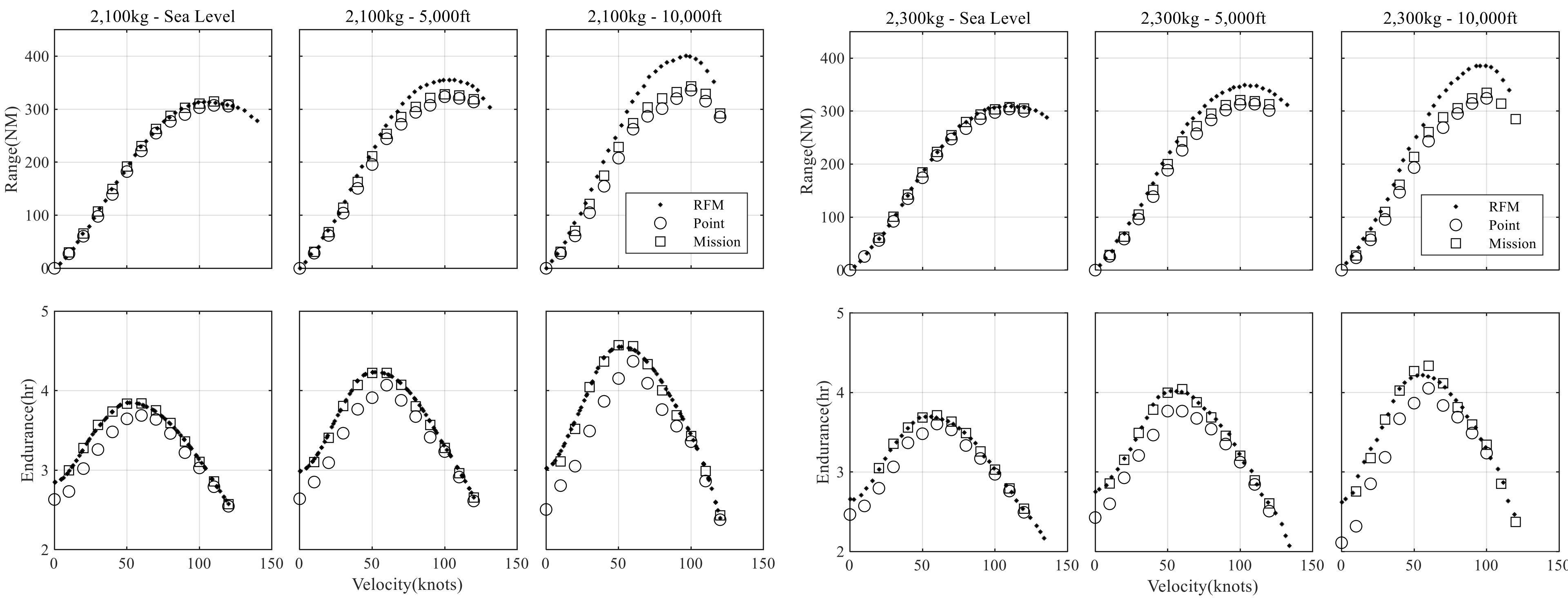
Mission Profile for Analysis



Validation of Flight Dynamic Model (HETLAS)

Comparison of Mission-Performance Analysis Results for Bo-105

J. An, Y.-S. Choi, I.-R. Lee, M. Lim, and C.-J. Kim, “Performance Analysis of a Conceptual Urban Air Mobility Configuration Using High-Fidelity Rotorcraft Flight Dynamic Model,” International Journal of Aeronautical and Space Sciences, Jul. 2023, doi: 10.1007/s42405-023-00610-7.



Endurance and Range Prediction

Validation of Flight Dynamic Model (HETLAS)

Validation of Bo-105 Model based on AC 120-63 - Helicopter Simulator Qualification

Table. AC 120 63 – Tolerance of trimmed flight control position and handling qualities.

Test	Tolerance	Comment
Level flight Performance and Trimmed Flight Control Position	Torque : ±3.0% Pitch Attitude : ±3.0° Control Position : ±5.0%	Forward Flight, Level (C, D)
Longitudinal Handing Qualities : Control Response	Pitch Rate : ±5.0% or ±2.0°/sec Pitch Attitude Change : ±10.0% or ±1.5°	Collective & Longitudinal , Level (B, C, D)
Lateral Handing Qualities : Control Response	Roll Rate : ±10.0% or ±3.0°/sec Roll Attitude Change : ±10.0% or ±3.0°	Level (B, C, D)
Directional Handing Qualities : Control Response	Yaw Rate : ±10.0% or ±2.0°/sec Yaw Attitude Change : ±10.0% or ±2.0°	Level (B, C, D)

Validation of Flight Dynamic Model (HETLAS)

Validation of Bo-105 Model based on AC 120-63 - Helicopter Simulator Qualification

[Bo-105 Data from :Padfield, Gareth D, Helicopter flight dynamics: the theory and application of flying qualities and simulation modelling, John Wiley & Sons, 2008]

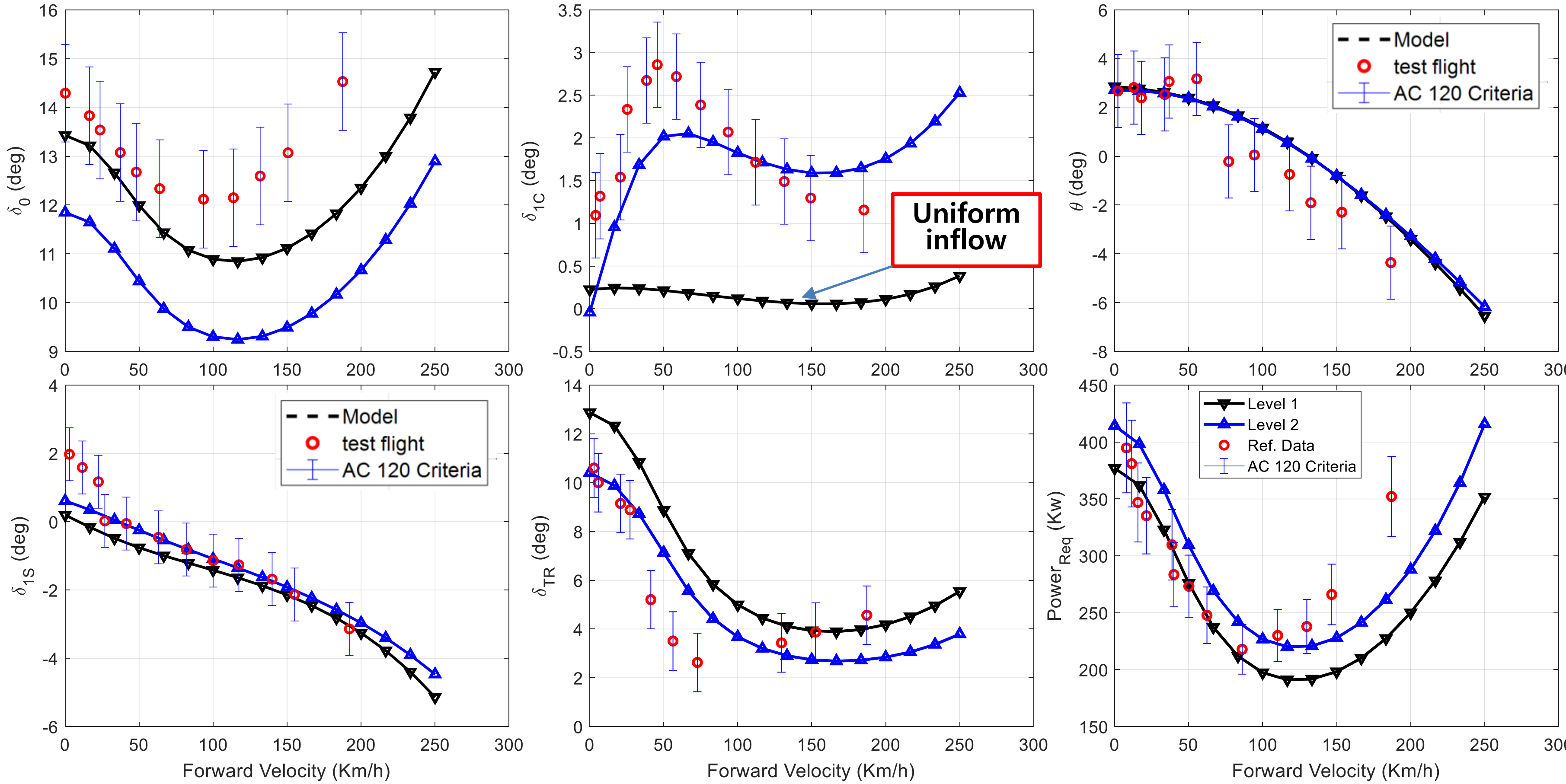


Fig. Forward flight trim result of BO-105 dynamic model

Validation of Flight Dynamic Model (HETLAS)

Validation of Bo-105 Model based on AC 120-63 - Helicopter Simulator Qualification

[Bo-105 Data from :Padfield, Gareth D, Helicopter flight dynamics: the theory and application of flying qualities and simulation modelling, John Wiley & Sons, 2008]

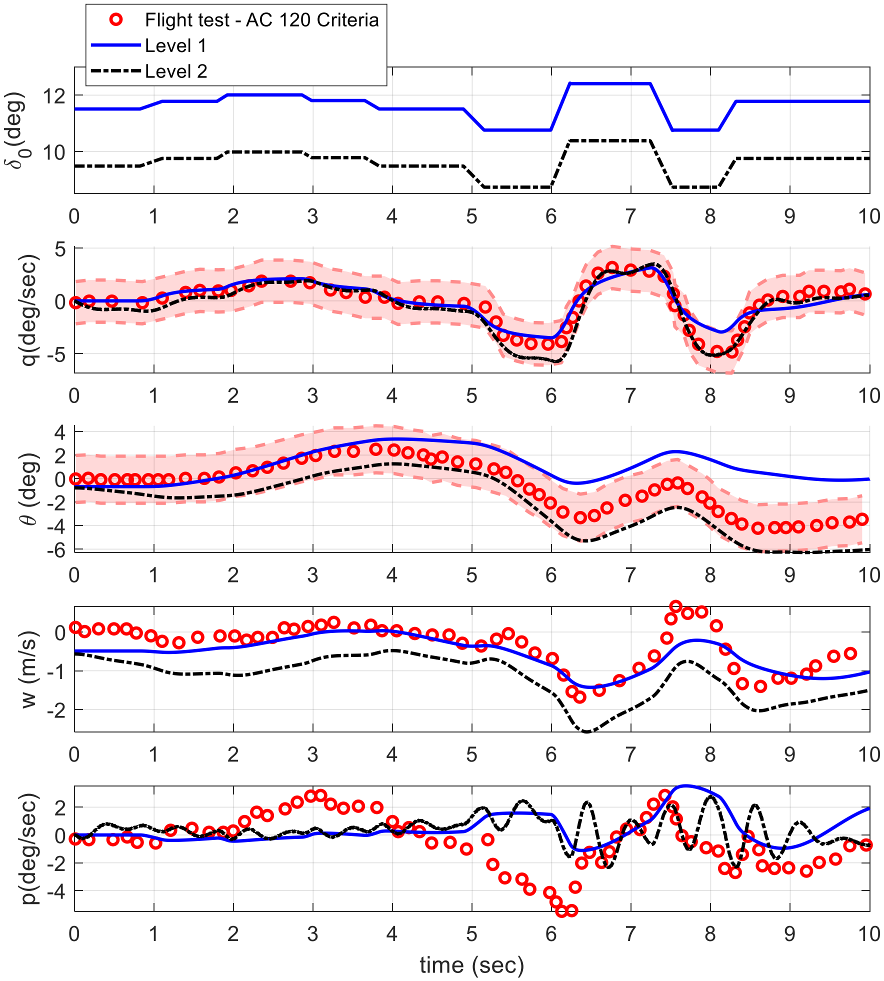


Fig. 80knot – collective input 3211 response

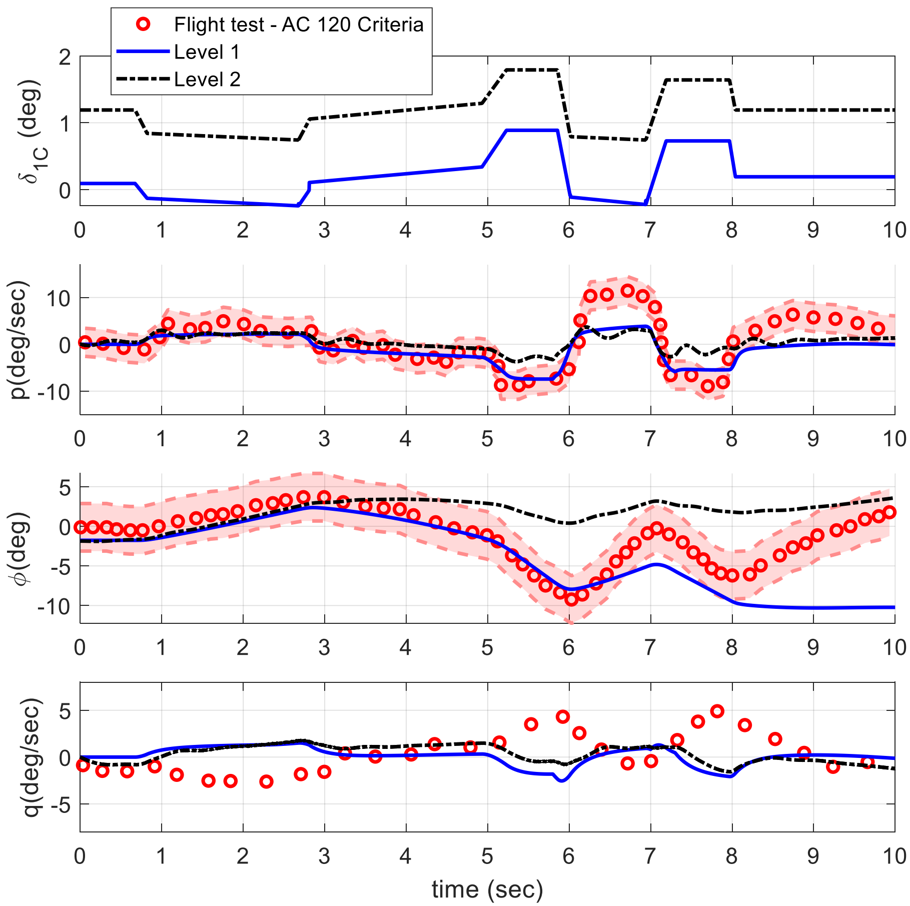


Fig. 80knot – Lateral input 3211 response

Validation of Flight Dynamic Model (HETLAS)

Validation of Bo-105 Model based on AC 120-63 - Helicopter Simulator Qualification

[Bo-105 Data from :Padfield, Gareth D, Helicopter flight dynamics: the theory and application of flying qualities and simulation modelling, John Wiley & Sons, 2008]

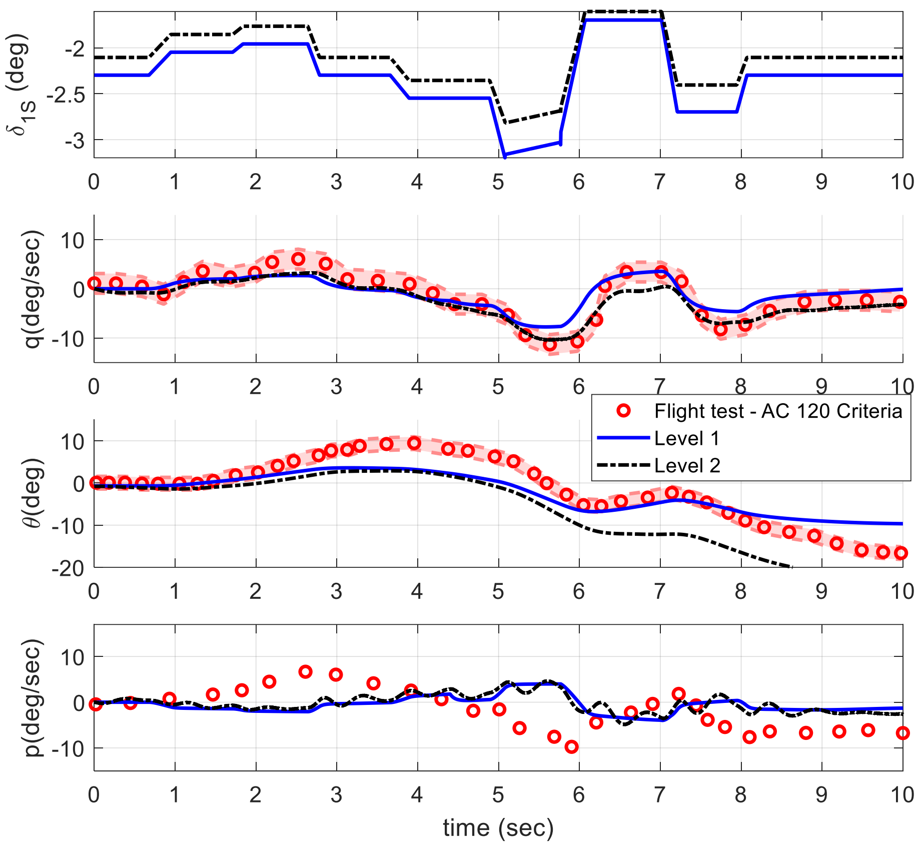


Fig. 80knot – Longitudinal input 3211 response

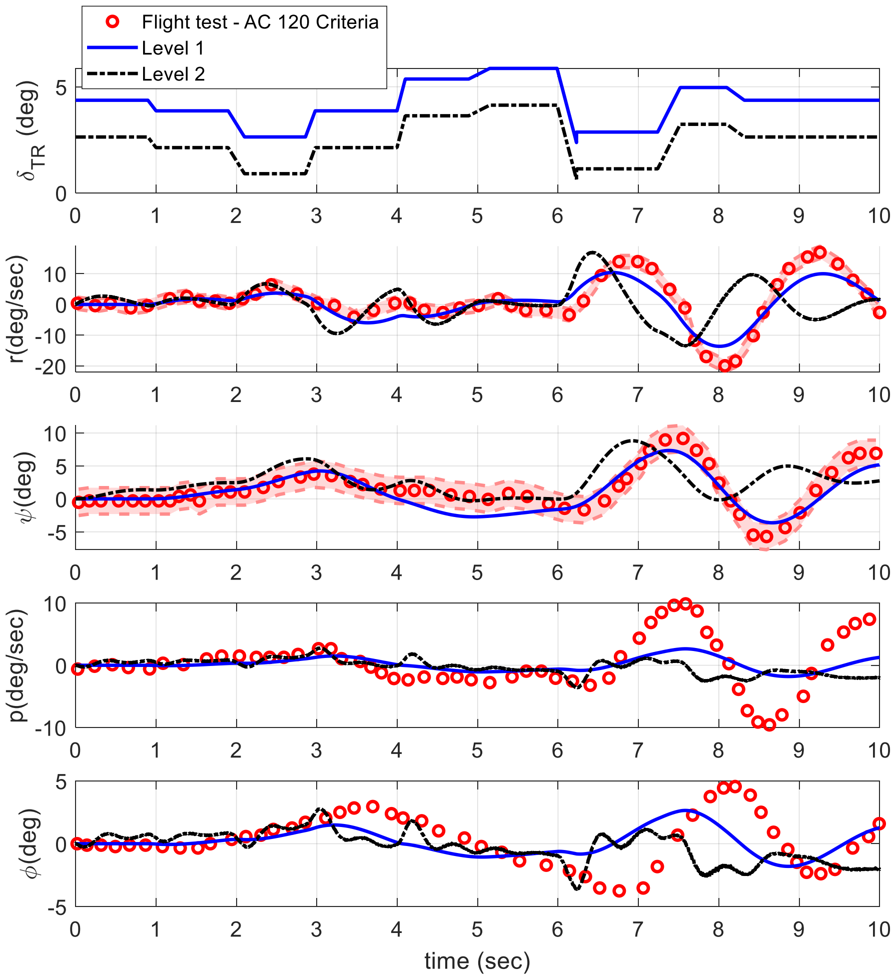
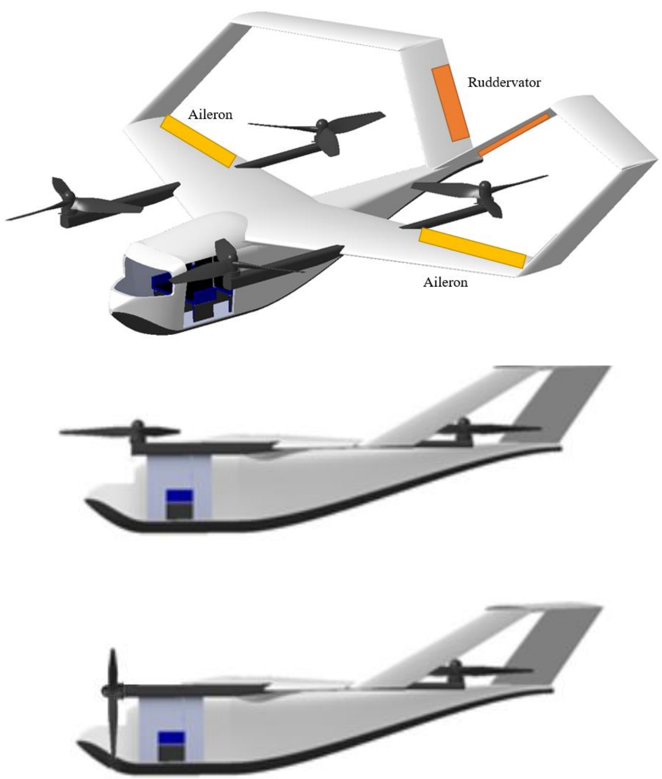


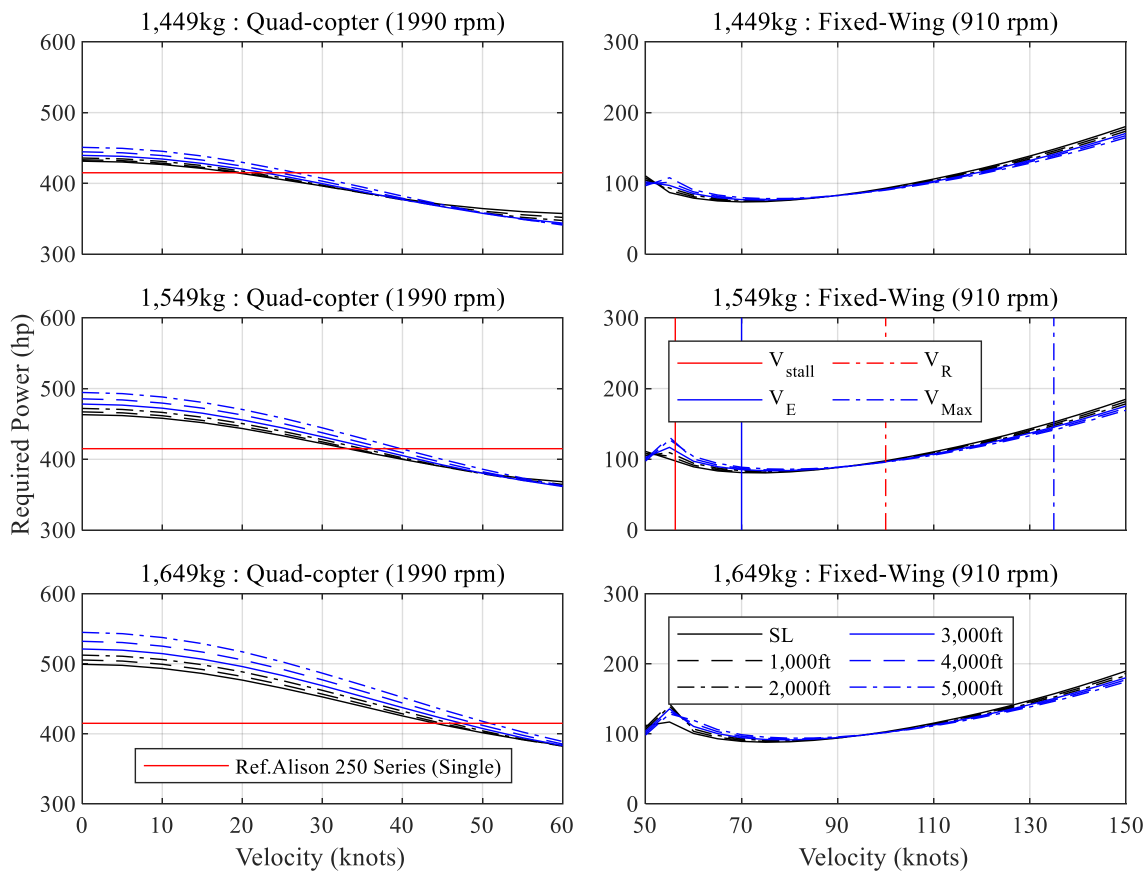
Fig. 80knot – tail collective input 3211 response

Validation of Flight Dynamic Model (HETLAS)

Application to KP-1 UAM (Urban Air Mobility) Model

J. An, Y.-S. Choi, I.-R. Lee, M. Lim, and C.-J. Kim, “Performance Analysis of a Conceptual Urban Air Mobility Configuration Using High-Fidelity Rotorcraft Flight Dynamic Model,” International Journal of Aeronautical and Space Sciences, Jul. 2023, doi: 10.1007/s42405-023-00610-7.





Quad-copter Mode

Fixed-Wing Mode

KU 建國大學校 KONKUK UNIV.

Contents

Importance of MTE Analysis

Definition and Verification Methods of Mission-Task-Elements

- MTEs provide a basis for an overall assessment of the rotorcraft's ability to perform certain critical tasks.
- One mission requires many of different flight tasks (MTEs)
- Mission success highly depends on the rotorcraft's performance for each MTEs
- ADS-33E-PRF defines 23 MTEs for Rotorcraft Handling Qualities Requirements

ADS-33E-PRF : Table XIV. Requirements/verification matrix

PARAGRAPH NO.	REQUIREMENT	VERIFICATION METHOD/EVENT				
		SFR	PDR	CDR	FFR	SVR
3.3	Hover and Low Speed					
3.3.1	Equilibrium Characteristics		A	A	A	F
3.3.2	Small-Amplitude Pitch (Roll) Attitude	A	A	A	A	F
3.11	Mission-Task-Elements			S	S	F

Methods of Verification:

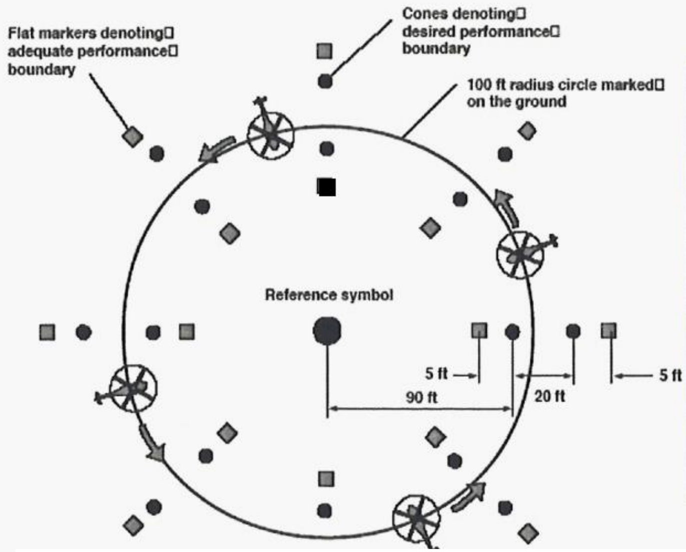
A – Analysis
S - Piloted Simulation
F - Flight Test
T - Testing, miscellaneous

Events:

SFR - System Functional Review
PDR - Preliminary Design Review
CDR - Critical Design Review
FFR - First Flight Readiness Review
SVR - System Verification Review

Importance of MTE Analysis

Example MTE: Piroutte in Test Guide for ADS-33E-PRF

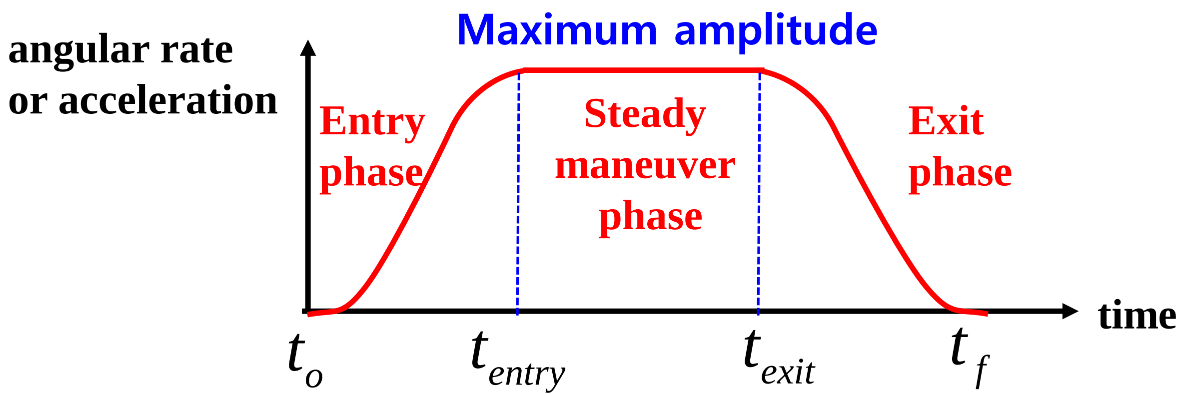


Performance – Pirouette

	GVE	DVE
DESIRED PERFORMANCE		
• Maintain a selected reference point on the rotorcraft within ±X ft of the circumference of the circle.	10 ft	10 ft
• Maintain altitude within ±X ft:	3 ft	4 ft
• Maintain heading so that the nose of the rotorcraft points at the center of the circle within ±X deg:	10 deg	10 deg
• Complete the circle and arrive back over the starting point within:	45 sec	60 sec
• Achieve a stabilized hover (within desired hover reference point) within X seconds after returning to the starting point.	5 sec	10 sec
• Maintain the stabilized hover for X sec	5 sec	5 sec

Importance of MTE Analysis

Maneuver Phases in MTEs and Rotorcraft Maneuverability/Agility



- **Maneuver Aggressiveness** is defined by entry/exit times and the maximum amplitude

$$\Delta t_{entry} = t_{entry} - t_o \qquad \dot{\phi}_{\max}, \dot{\theta}_{\max}, \dot{\psi}_{\max}$$

$$\Delta t_{exit} = t_f - t_{exit} \qquad \mathbf{a}_{\max}$$

- **Maneuverability** is evaluated with **Agility**, which is defined with both
  - Maneuver **Aggressiveness** and Maneuver **Precision**
- **Maneuverability** is directly affected by the quantitative Handling-Qualities requirements which are defined in Para. 3.3~3.10 in ADS-33E PRF

Thus, MTE Analysis allows both direct evaluation of Rotorcraft maneuverability and indirect evaluation of quantitative (objective) requirements of ADS-33E PRF

Methodologies for MTE Analysis

Two General Approaches: Inverse Simulation / Nonlinear Optimal Control Analysis

(1) Inverse Simulation Approach

- Requires Accurate Prescription of Trajectory for a Specific MTE
- Only Applicable to Aircraft Maneuvers in Normal Operating States (no Engine Failure)
- Most of Available Algorithms suffer from Numerical Stability Problems
- You can refer to following papers for Historical Overview and Theoretical Details

[1] Thomson, D.G., and Bradley, R., "Inverse simulation as a tool for flight dynamics research—Principles and applications," Progress in Aerospace Sciences, Vol. 42, 2006, pp. 174–210.

[2] Lu,L., Murray-Smith, D.J., and Thomson, D.G., "Issues of numerical accuracy and stability in inverse simulation," Simulation Modelling Practice and Theory, Vol. 16, 2008, pp. 1350–1364.

[3] Thomson, Douglas G.; Bradley, Roy, "Mathematical Definition of Helicopter Maneuvers," Journal of the American Helicopter Society, Volume 42, Number 4, 1 October 1997, pp. 307-309.

[4] R. Celi, "Optimization-Based Inverse Simulation of a Helicopter Slalom Maneuver," Journal of Guidance, Control, and Dynamics, Vol. 23, No. 2, 2000, pp. 289-297

[5] Giulio Avanzini, Guido de Matteis, and Luciano M. de Socio. "Two-Timescale-Integration Method for Inverse Simulation", Journal of Guidance, Control, and Dynamics, Vol. 22, No. 3 (1999), pp. 395-401.

[6] R.A. Hess, C. Gao, S.H. Wang, "A generalized technique for inverse simulation applied to aircraft manoeuvres," J. Guidance, Control Dynamics 14 (1991) 920–926.

[7] Murray-Smith, D.J., "The inverse simulation approach: a focused review of methods and applications," Mathematics and Computers in Simulation, Vol. 53, 2000, pp. 239–247.

Methodologies for MTE Analysis

Two General Approaches: Inverse Simulation / Nonlinear Optimal Control Analysis

(2) Nonlinear Optimal Control Theory (NOCP: Nonlinear Optimal Control Problem)

- Adopt Trajectory Tracking Control Law when Trajectory is prescribed
- Applicable to Rotorcraft Maneuvers under Failures such as Engine Malfunction
- Extremely High Computing Time is required
- No methods are available at Present time for applications using Rotorcraft Math Models with Rotor and Inflow Dynamics due to Large KKT (Karush-Kuhn-Tucker) System in Direct Methods and the extremely poor robustness with Indirect Methods

NOCP Solver

Indirect Method

- Euler-Largange Equations
- TPBVP(Two-Point Boundary Value Problem)
- Single Shooting Method
- Multiple-Shooting Method

Direct Method

- Apply Transcription Method to get NLP
- Nonlinear Programming Problem (NLP)
- Sequential Quadratic Programming Algorithm to solve NLP

KU 建國大學校 KONKUK UNIV.

Contents

Kinematically Exact Inverse Simulation Technique

Recent Research on Rotorcraft Inverse Simulation Techniques at KKU: PIST & KEIST

2019. Chang-Joo Kim, Do Hyeon Lee, and Sung Wook Hur, "Efficient and Robust Inverse Simulation Techniques Using Pseudo-Spectral Integrator with Applications to Rotorcraft Aggressive Maneuver Analyses," International Journal of Aeronautical and Space Sciences, March 2019.

2020 Chang-Joo Kim, Seong Han Lee, and Sung Wook Hur, "Kinematically Exact Inverse Simulation Techniques with Applications to Rotorcraft Aggressive-Maneuver Analyses," International Journal of Aeronautical and Space Sciences, March 2020.

Problem Definition of General Inverse Simulation Problem to Find Control

Motion equations

$$\dot{\mathbf{v}} = \mathbf{f}/m - \boldsymbol{\omega} \times \mathbf{v}$$

$$\dot{\boldsymbol{\omega}} = \mathbf{J}^{-1}\left\{\mathbf{m} - \boldsymbol{\omega} \times (\mathbf{J}\boldsymbol{\omega})\right\}$$

$$\mathbf{v} = \begin{pmatrix} u \\ v \\ w \end{pmatrix}, \quad \boldsymbol{\omega} = \begin{pmatrix} p \\ q \\ r \end{pmatrix}, \quad \boldsymbol{\varphi} = \begin{pmatrix} \phi \\ \theta \\ \psi \end{pmatrix}, \quad \mathbf{r} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

Prescribed Trajectory: Typically by Position Vector and Heading Angle

$$\mathbf{J} = \begin{pmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & I_{zz} \end{pmatrix}, \quad \mathbf{f} = \begin{pmatrix} f_x \\ f_y \\ f_z \end{pmatrix}, \quad \mathbf{m} = \begin{pmatrix} m_x \\ m_y \\ m_z \end{pmatrix}$$

$$\mathbf{p} = \left(\mathbf{r}^p, \psi^p\right)$$

Inverse Simulation Problem: Find Flight Control to track the Prescribed Path

Kinematically Exact Motion Equations for Inverse Simulation in Inertial Frame

Using angular kinematics and navigation equations, we can get new form of motion equations

$$\boldsymbol{\omega} = \mathbf{T}\dot{\boldsymbol{\varphi}} \qquad \dot{\boldsymbol{\omega}} = \dot{\mathbf{T}}\dot{\boldsymbol{\varphi}} + \mathbf{T}\ddot{\boldsymbol{\varphi}}$$

$$\mathbf{v} = \mathbf{C}\dot{\mathbf{r}} \qquad \dot{\mathbf{v}} = \dot{\mathbf{C}}\dot{\mathbf{r}} + \mathbf{C}\ddot{\mathbf{r}}$$

$$\boxed{\begin{aligned} \ddot{\mathbf{r}} &= \mathbf{C}^{-1}\left\{\mathbf{f}/m - (\mathbf{T}\dot{\boldsymbol{\varphi}})\times(\mathbf{C}\dot{\mathbf{r}}) - \dot{\mathbf{C}}\dot{\mathbf{r}}\right\} \\ \ddot{\boldsymbol{\varphi}} &= \mathbf{T}^{-1}\left[\mathbf{J}^{-1}\left\{\mathbf{m} - (\mathbf{T}\dot{\boldsymbol{\varphi}})\times(\mathbf{J}\mathbf{T}\dot{\boldsymbol{\varphi}})\right\} - \dot{\mathbf{T}}\dot{\boldsymbol{\varphi}}\right] \end{aligned}}$$

Using the prescribed trajectory information $(\mathbf{r}^p, \dot{\mathbf{r}}^p, \ddot{\mathbf{r}}^p, \psi^p, \dot{\psi}^p, \ddot{\psi}^p)$

We can get kinematically exact motion equations in DAE (Differential-Algebraic-Equation) form

$$\boxed{\begin{aligned} \ddot{\mathbf{x}} &= \bar{\mathbf{m}}(\mathbf{x},\dot{\mathbf{x}},\mathbf{u},t) \in R^2 \\ \mathbf{0} &= \bar{\mathbf{f}}(\mathbf{x},\dot{\mathbf{x}},\mathbf{u},t) - \ddot{\mathbf{r}}^p \in R^3 \\ 0 &= \bar{m}(\mathbf{x},\dot{\mathbf{x}},\mathbf{u},t) - \ddot{\psi}^p \in R \end{aligned}}$$

: Two Ordinary differential equations

: Nonlinear algebraic equations

: Nonlinear algebraic equations

where $\mathbf{x} = \begin{pmatrix} \phi \\ \theta \end{pmatrix}$ $\quad \mathbf{u} = \begin{pmatrix} \delta_0 \\ \delta_{1C} \\ \delta_{1C} \\ \delta_{TR} \end{pmatrix}$

Control Equations from 2nd and 3rd equations represents a Index 1 DAE system

$$\begin{pmatrix} \dfrac{\partial \bar{\mathbf{f}}}{\partial \mathbf{u}} \\ \dfrac{\partial \bar{m}}{\partial \mathbf{u}} \end{pmatrix} \dot{\mathbf{u}} = -\begin{pmatrix} \dfrac{\partial \bar{\mathbf{f}}}{\partial \mathbf{x}}\dot{\mathbf{x}} + \dfrac{\partial \bar{\mathbf{f}}}{\partial \dot{\mathbf{x}}}\bar{\mathbf{m}} + \dfrac{\partial \bar{\mathbf{f}}}{\partial t} - \dddot{\mathbf{r}}^p \\ \dfrac{\partial \bar{m}}{\partial \mathbf{x}}\dot{\mathbf{x}} + \dfrac{\partial \bar{m}}{\partial \dot{\mathbf{x}}}\bar{\mathbf{m}} + \dfrac{\partial \bar{m}}{\partial t} - \dddot{\psi}^p \end{pmatrix}$$

since the leading matrix is nonsingular in general rotorcraft flight dynamics

Solution using Pseudo-spectral (PS) time integrator and Quasi-Newton Method

$\ddot{\mathbf{x}} = \bar{\mathbf{m}}(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{u}, t) \in R^2$: Ordinary differential equations (ODEs)

$\mathbf{0} = \bar{\mathbf{f}}(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{u}, t) - \ddot{\mathbf{r}}^p \in R^3$: Nonlinear algebraic equations (NAEs)

$0 = \bar{m}(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{u}, t) - \ddot{\psi}^p \in R$: Nonlinear algebraic equations (NAEs)

Direct Application of PS time integrator to 2nd order ODEs with Piccard Method

$$\dot{\mathbf{x}}_j^{(iter+1)} = \dot{\mathbf{x}}_0 + \frac{t_f - t_0}{2} \sum_{k=0}^{k=N} I_{jk} \bar{\mathbf{m}}_k^{(iter)} \quad , \quad (j = 1, 2, \cdots, N)$$

$$\mathbf{x}_j^{(iter+1)} = \mathbf{x}_0 + \frac{t_f - t_0}{2} \sum_{k=0}^{k=N} I_{jk} \dot{\mathbf{x}}_j^{(iter)}$$

Quasi-Newton Method for NAEs

$$\mathbf{g}_{1,j} = \bar{\mathbf{f}}_j - \ddot{\mathbf{r}}_j^p = \mathbf{0}$$

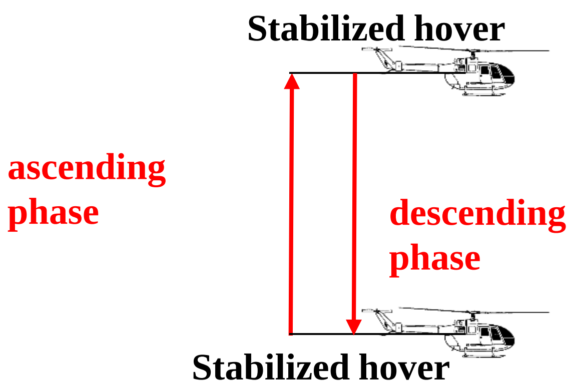
$$g_{2,j} = \bar{m}_j - \ddot{\psi}_j^p = 0$$

To find unknowns
$$\mathbf{x} = \begin{pmatrix} \phi \\ \theta \end{pmatrix} \quad \mathbf{u} = \begin{pmatrix} \delta_0 \\ \delta_{1C} \\ \delta_{1C} \\ \delta_{TR} \end{pmatrix}$$

You can refer to Reference 2020 for a detailed implementation in computer model.

Kinematically Exact Inverse Simulation Technique

Application to Bop-up MTE



$$v_1 = v_2 = 0 \text{ knots}$$
$$h_1 = h_2 = 35 \text{ ft}$$
$$t_1 = 4 \text{ sec}, \; t_2 = 6 \text{ sec}$$

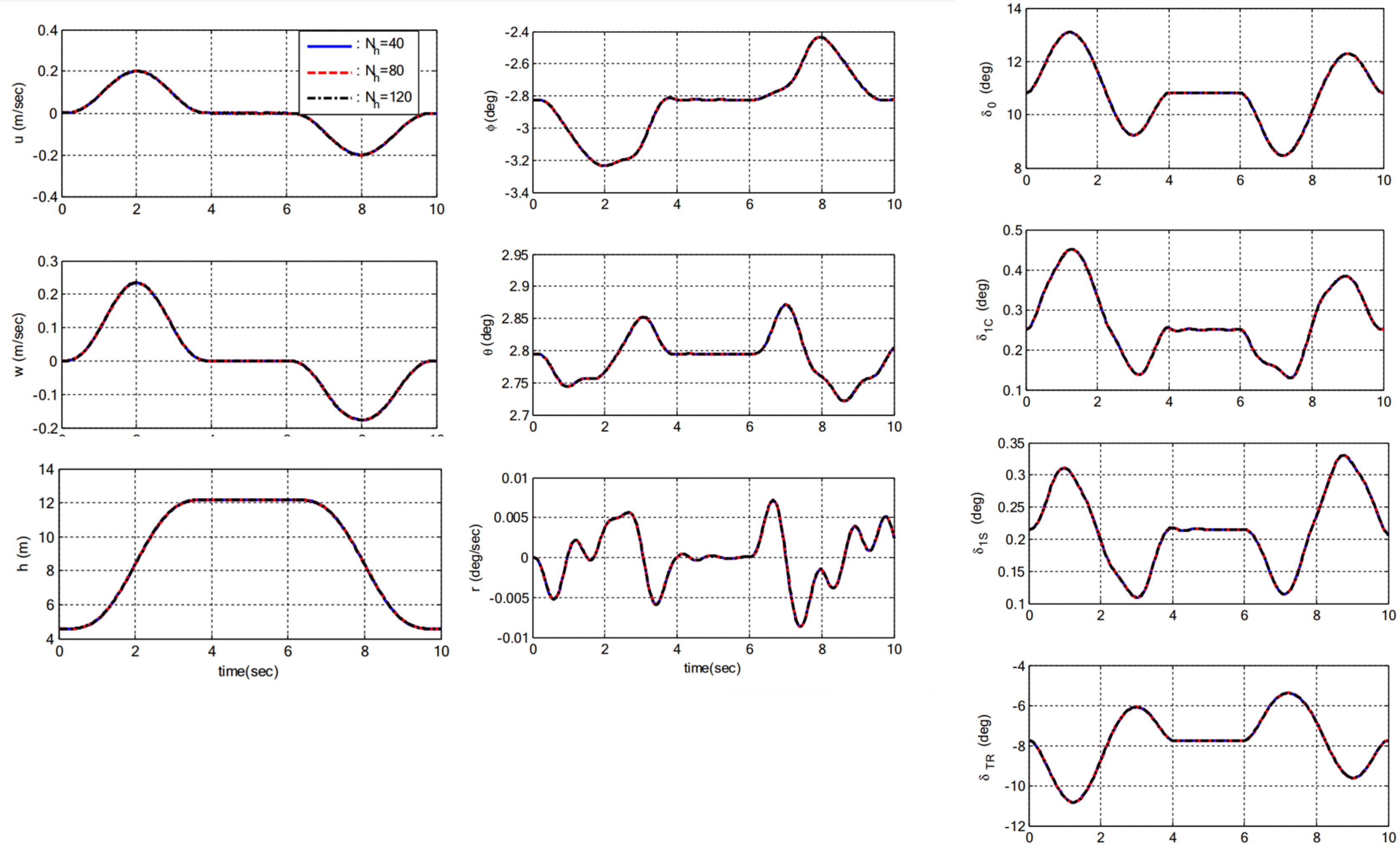
$$v_0 = v_f = 0 \text{ knots}$$
$$h_0 = h_f = 15 \text{ ft}$$
$$t_0 = 0 \text{ sec}, \; t_f = 10 \text{ sec}$$

Case	N	N_h	Δt_{avg} (s)	CPU time (Intel i7)/It_{avg}	
				KEIST formulation	Conventional formulation
1	4	20	0.1250	Failed at first segment	
2	4	40	0.0625	203/8.6	223/10.7
3	4	60	0.0417	141/7.7	170/9.8
4	4	80	0.0313	147/7.4	183/8.1
5	4	100	0.0250	155/6.5	201/8.1
6	4	120	0.0208	186/7.2	211/7.7

N = number of quadrature points
Nh = number of time horizon segments

Kinematically Exact Inverse Simulation Technique

Application to Bop-up MTE



Kinematically Exact Inverse Simulation Technique

Application to Helical Turn MTE

$t_f = 66$ sec

$h_f = 400$ ft

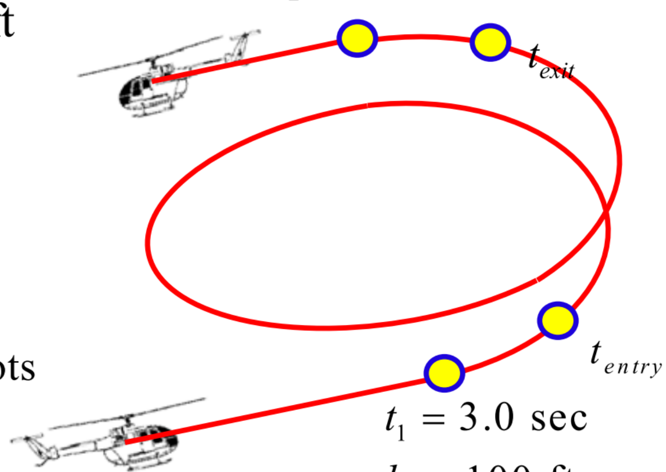
$t_2 = 63$ sec

$h_2 = 400$ ft

$t_0 = 0$ sec

$h_0 = 100$ ft

$v_0 = 60$ knots



$t_1 = 3.0$ sec

$h_1 = 100$ ft

$t_{entry} = t_1 + 0.2\Delta t_{12}$

$t_{exit} = t_1 + 0.8\Delta t_{12}$

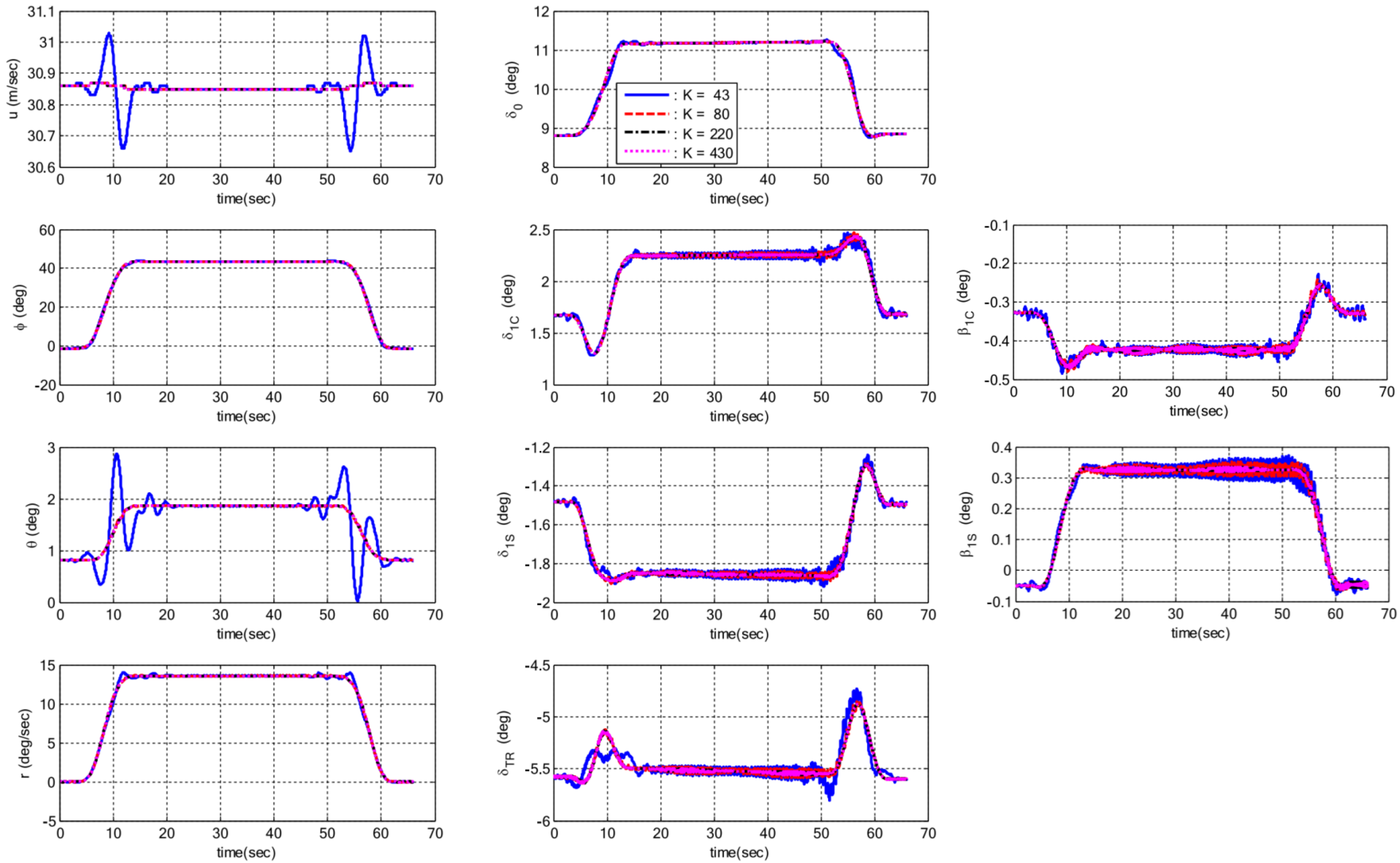
K = number of waypoint data

Case	K_0	K_{entry}	K_{steady}	K_{exit}	K_f	$K = K_{total}$
1	5	4	25	4	5	43
2	5	10	50	10	5	80
3	5	30	150	30	5	220
4	5	60	300	60	5	430

KU 建國大學校 KONKUK UNIV.

Kinematically Exact Inverse Simulation Technique

Application to Helical Turn MTE



KU 建國大學校 KONKUK UNIV.

Contents

Publications on Nonlinear Optimal Control Approaches to Rotorcraft MTE Analysis

[1] CJ Kim, J Lee, YH Byun, and YH Yu, "Nonlinear Optimal Control Analysis of Helicopter Maneuver Problems Using the Indirect Method," Transactions of the Japan Society for Aeronautical and Space Sciences, 2008.

[2] Chang-Joo Kim, Sang Kyung Sung, Soo Hyung Park, Sung-Nam Jung and Kwanjung Yee, "Selection of Rotorcraft Models for Application to Optimal Control Problems," Journal of Guidance Control and Dynamics, Vol. 31, No. 5, September–October 2008

[3] Chang-Joo Kim, Chang-Deok Yang, Seung-Ho Kim, and Changjeon Hwang, "The Analysis of Helicopter Maneuvering Flight Using the Indirect Method - Part II. Applicability of High Fidelity Helicopter Models," Journal of the Korean Society for Aeronautical & Space Sciences 36(1), 2008

[4] Chang-Joo Kim, Chang-Deok Yang, Seung-Ho Kim, and Changjeon Hwang, "Analysis of Helicopter Maneuvering Flight Using the Indirect Method - Part I. Optimal Control Formulation and Numerical Methods," Journal of the Korean Society for Aeronautical & Space SciencesJanuary 2008.

[5] Min-Jae Kim, Ji-Seung Hong, and Chang-Joo Kim, "Finding Optimal Controls for Helicopter Maneuvers Using the Direct Multiple-Shooting Method," International Journal of Aeronautical and Space Sciences, March 2010.

[6] Chang-Joo Kim, Sangkyung Sung, Soo Hyung Park, et al., "Numerical Time-Scale Separation for Rotorcraft Nonlinear Optimal Control Analyses," Journal of Guidance Control and Dynamics. 2014, Vol.37, No.2, p.658.

[7] Kim C-J, Sung SK, "A comparative study of transcription techniques for nonlinear optimal control problems using a pseudo-spectral method," International Journal of Aeronautical and Space Sciences, Vol.16, No.2, pp264–277, 2015

[8] Jun-young An, Chang-Joo Kim, Sungwook Hur, and Seong han Lee, "Category A Takeoff and Landing Trajectory Optimization for Transport Category Rotorcraft Certification," Journal of Institute of Control Robotics and Systems, December 2019

[9] Yong Hyeon Nam, Chang-Joo Kim, Seong Han Lee, and Yi Young Kwak, "Direct Dynamic-Simulation Approach to Trajectory Optimization for Rotorcraft Category-A Maneuver Procedures," International Journal of Aeronautical and Space Sciences, Vol.22, pp.648~662, November 2021

Direct Dynamic Simulation Approach (DDSA) to NOCP

Nonlinear Optimal Control Problem (NOCP)

$$
\begin{aligned}
&\min_{\mathbf{x},\mathbf{u}} J(\mathbf{x},\mathbf{u},t) = \phi(\mathbf{x}(t_0),t_0,\mathbf{x}(t_f),t_f) + \int_{t_0}^{t_f} f_{obj}(\mathbf{x}(t),\mathbf{u}(t),t)dt \\
&\text{subject to} \quad \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u},t) \\
&\qquad\qquad\quad\;\; \mathbf{h}(\mathbf{x}) = \mathbf{0} \\
&\qquad\qquad\quad\;\; \mathbf{g}(\mathbf{x}) \le 0
\end{aligned}
$$

J : total cost function

ϕ : cost function for Initial and final conditions

f_{obj} : integral cost function

t_0 : initial time

t_f : final time

$\mathbf{h}$: equality constraint function

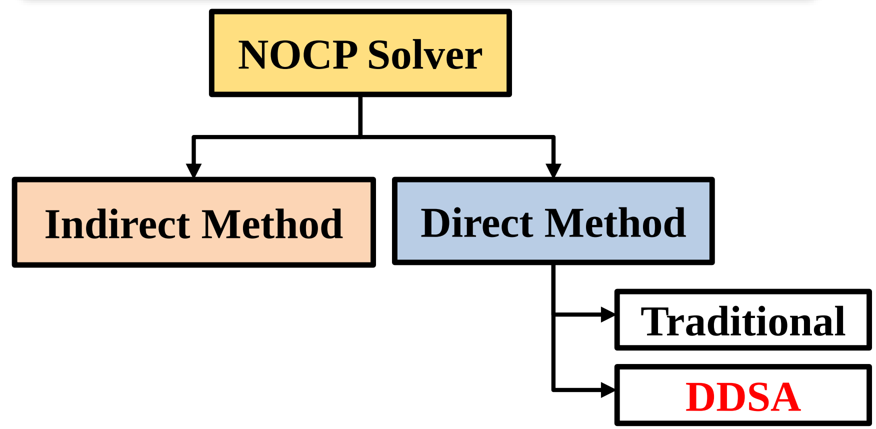
$\mathbf{g}$: inequality constraint function

$\mathbf{x}$: system states

$\mathbf{u}$: system control

$\mathbf{f}$: system forcing function

Two Typical Methods for NOCP Solution



[Remark] Direct Method typically has much higher robustness than Indirect Method

Direct Dynamic Simulation Approach (DDSA) to NOCP

Typical Procedures in Direct Method

NOCP

$$\min \quad J(\mathbf{x},\mathbf{u},t) = \phi(\mathbf{x}_0,t_0,\mathbf{x}_f,t_f) + \int_{t_0}^{t_f} f_{obj}(\mathbf{x},\mathbf{u},t)dt$$

$$\text{s.t.} \quad \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x},\mathbf{u},t), \quad t \in [t_0,t_f], \quad \mathbf{x} \in R^n, \quad \mathbf{u} \in R^m$$

$$\mathbf{h}(\mathbf{x}_0,t_0,\mathbf{x}_f,t_f) = 0, \quad \mathbf{h} \in R^{L_e}$$

$$\mathbf{g}(\mathbf{x},\mathbf{u},t) \le 0, \quad t \in [t_0,t_f], \quad \mathbf{g} \in R^{L_i}$$

Transcription of NOCP

Parametrization of both Controls & States

(Pseudo Spectral Method Using LGL Points)

NLP

$$\min \ f(\mathbf{x}) \qquad x \in R^n$$

$$s.t. \ h_j(\mathbf{x}) = 0 \quad \text{for } \ j \in E = \{1,2,\cdots,m\}$$

$$g_j(\mathbf{x}) \le 0 \quad \text{for } \ j \in I = \{1,2,\cdots,p\}$$

NLP Solving Algorithm for KKT (Karush-Kuhn-Tucker) System

Sequential Quadratic Programming

rSQP

The transcription (Discretization) intends to convert NOCP into NLP (Nonlinear Programming Problem) by applying time integrator over all computational time nodes like

$$J = \phi(\mathbf{x}_0,t_0,\mathbf{x}_f,t_f) + \frac{t_f - t_0}{2}\sum_{j=0}^{j=N} w_j f_{obj}(\mathbf{x}_j,\mathbf{u}_j,t_j)$$

$$\mathbf{x}_j = \mathbf{x}_0 + \frac{t_f - t_0}{2}\sum_{j=0}^{j=N} I_{jk}\mathbf{f}(\mathbf{x}_j,\mathbf{u}_j,t_j)$$

Thus, the system dynamics are converted into equality constraints at NLP. In addition, the NLP solver must compute unknowns design variables consisting of system states and controls at all time nodes.

$$\left\{\mathbf{x}_j,\mathbf{u}_j\right\}_{j=0}^{j=N}$$

Direct Dynamic Simulation Approach (DDSA) to NOCP

Conventional direct methods suffer from serious curse of dimensionality when using a high-fidelity rotorcraft math model due to

- Rotor dynamics (even for flap and lead-lag dynamics in rigid-blade models
- Inflow dynamics

Since the discretization of these dynamics typically require over 36 time nodes per one rotor revolution to obtain accurate time integrations of dynamical equations. Thus, the size of KKT systems is dramatically increased as the time horizon of NOCP is increased

Two Baseline concepts in developing DDSA

(1) The system states are uniquely determined by the control inputs. Thus, the states are possibly excluded from the design variables in NLP during the transcription process. In addition, the system dynamics are simply satisfied using an accurate time integrator

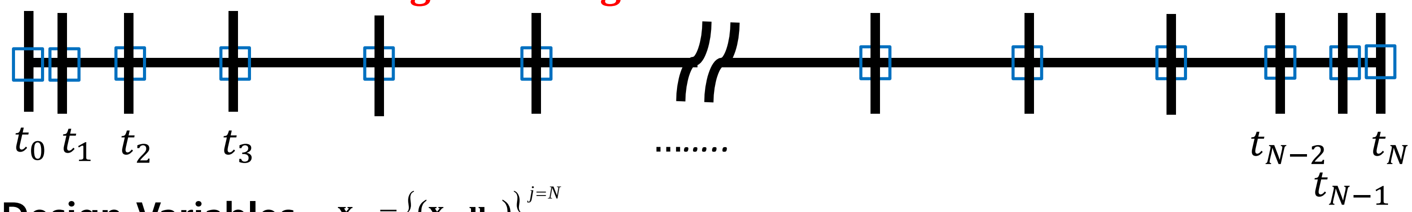
(2) Computational efficiency can be enhanced through the control parametrization using Hermit spline interpolation.

Thus, the KKT system can be derived only for system controls, which can dramatically reduced the number of both design variables and constraint functions.

Direct Dynamic Simulation Approach (DDSA) to NOCP

Comparison of Pseudo-Spectral transcription methods : Traditional method vs DDSA

Traditional method over a single time segment

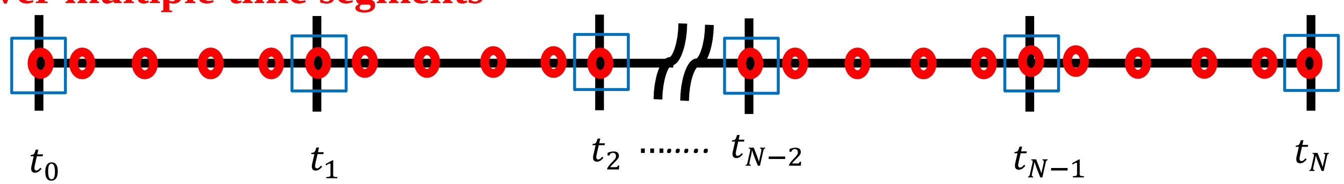


Design Variables $\mathbf{x}_D = \{(\mathbf{x}_j, \mathbf{u}_j)\}_{j=0}^{j=N}$

$\mathbf{x}_0$ $\mathbf{x}_1$ $\mathbf{x}_2$ $\mathbf{x}_3$ .... $\mathbf{x}_{N-2}$ $\mathbf{x}_{N-1}$ $\mathbf{x}_N$

$\mathbf{u}_0$ $\mathbf{u}_1$ $\mathbf{u}_2$ $\mathbf{u}_3$ .... $\mathbf{u}_{N-2}$ $\mathbf{u}_{N-1}$ $\mathbf{u}_N$

DDSA over multiple time segments



Design Variables $\mathbf{x}_D = \{\mathbf{u}_j\}_{j=0}^{j=N}$

$\mathbf{x}_0$

$\mathbf{u}_0$ $\mathbf{u}_{0,1}$ $\mathbf{u}_{0,2}$ $\mathbf{u}_{0,3}$ $\mathbf{u}_{0,4}$ $\mathbf{u}_1$ $\mathbf{u}_{1,1}$ $\mathbf{u}_{1,2}$ $\mathbf{u}_{1,3}$ $\mathbf{u}_{1,4}$ $\mathbf{u}_2$ .... $\mathbf{u}_{N-2}$ $\mathbf{u}_{N-2,1}$ $\mathbf{u}_{N-2,2}$ $\mathbf{u}_{N-2,3}$ $\mathbf{u}_{N-2,4}$ $\mathbf{u}_{N-1}$ $\mathbf{u}_{N-1,1}$ $\mathbf{u}_{N-1,2}$ $\mathbf{u}_{N-1,3}$ $\mathbf{u}_{N-1,4}$ $\mathbf{u}_N$

Integrated State Variables

$\mathbf{x}_1$ $\mathbf{x}_2$ .... $\mathbf{x}_{N-2}$ $\mathbf{x}_{N-1}$ $\mathbf{x}_N$

Direct Dynamic Simulation Approach (DDSA) to NOCP

Control Interpolation using Hermit Spline Interpolation for DDSA

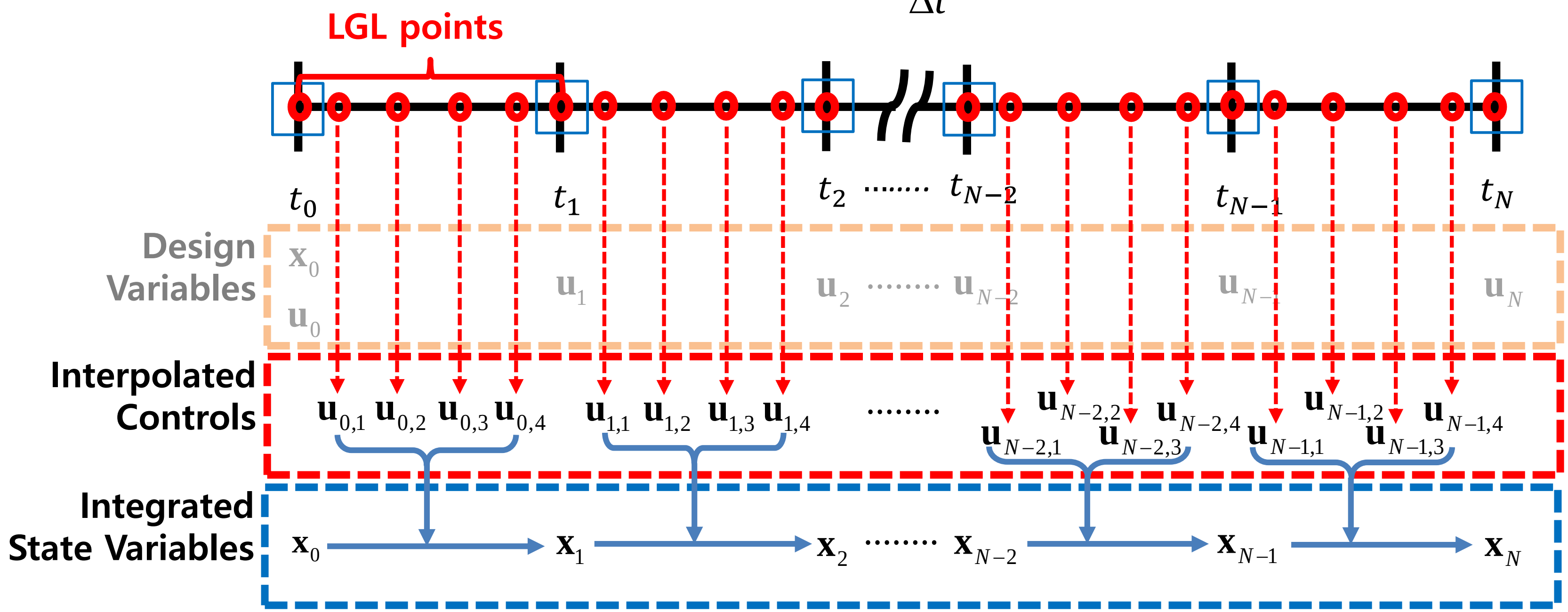
Interpolation of control inputs over $t \in [t_j, t_{j+1}]$

$$\boxed{\mathbf{u}(\tau) = \sum_{l=0}^{l=L} \left(\Delta t_j\right)^l \left\{\alpha_l(\tau)\mathbf{u}_j^{(l)} + \beta_l(\tau)\mathbf{u}_{j+1}^{(l)}\right\}}$$

$$\alpha_k^{(j)}(0) = \delta_{jk} \quad \alpha_k^{(j)}(1) = 0$$
$$\beta_k^{(j)}(0) = 0, \quad \beta_k^{(j)}(1) = \delta_{jk}$$

with $\delta_{jk} = \begin{cases} 1 & (j = k) \\ 0 & (j \neq k) \end{cases}$

$$\tau = \frac{t - t_j}{\Delta t} \in [0,1], \quad \Delta t_j = t_{j+1} - t_j$$



You can refer to Reference 2021[9] for a detailed implementation in computer model.

Direct Dynamic Simulation Approach (DDSA) to NOCP

Comparison of Computational Efficiency for Simple Problem : Traditional method vs DDSA

NOCP: Soft lunar landing of a spacecraft

$$
\begin{aligned}
\min \quad & J = \int_0^{t_f} u\,dt \\
s.t. \quad & \dot{x}_1 = x_2, \\
& \dot{x}_2 = -1.5 + u \\
& x_1(0) = 10, x_2(0) = -2 \\
& x_1(t_f) = 0, x_2(t_f) = 0 \\
& 0 \le u \le 3
\end{aligned}
$$

Exact solution

$$
u = \begin{cases} 0 & t < t_s^* \\ 3 & t_s^* < t \end{cases}, \quad \left(t_s^* = \frac{t_f^*}{2} + \frac{v_0}{3} \right)
$$

Computational Nodes

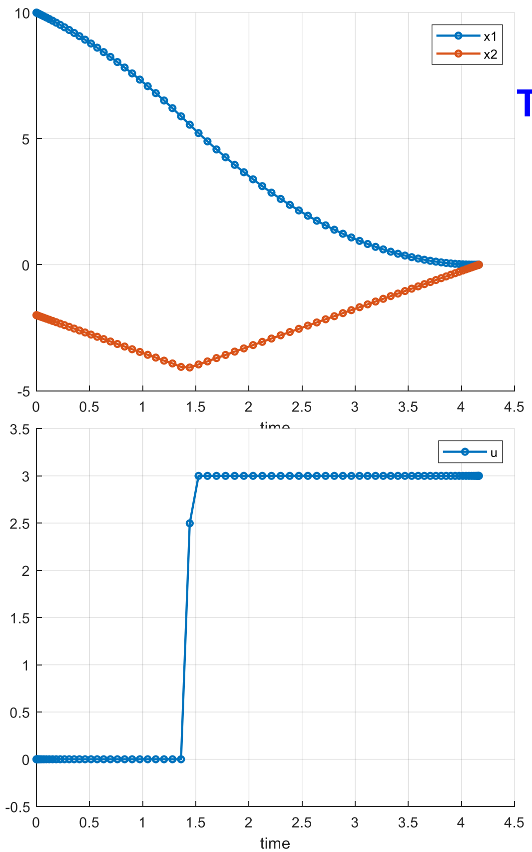
	PS Method	DDSA
Computational Nodes	-	4
Collocation Nodes	76	26
Size of the KKT system	536	234

Ratio of KKT System Sizes: 2.29

Direct Dynamic Simulation Approach (DDSA) to NOCP

Comparison of Computational Efficiency for Simple Problem : Traditional method vs DDSA

Pseudo Spectral Method



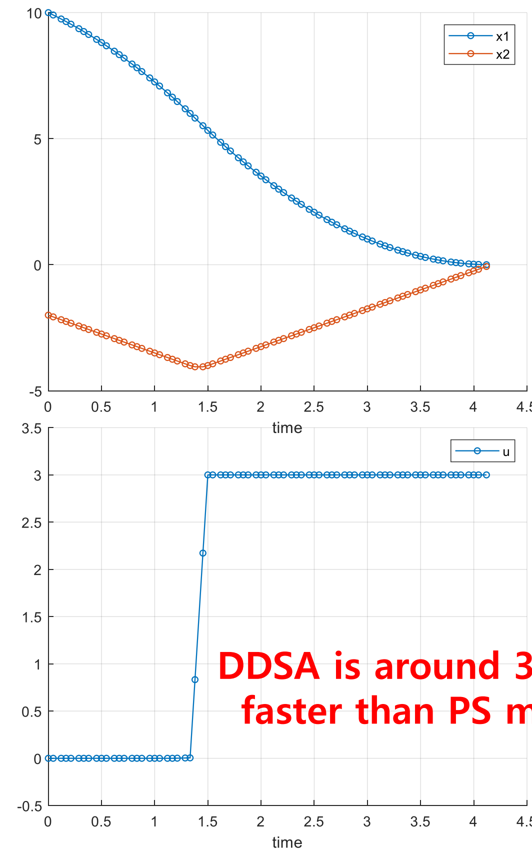
Cost Function: 0.8246E+1
CPU Time : 601.0 [sec]

Termination Condition

ε_{KKT} : $0.1E-5$

$\varepsilon_{\text{constraints}}$: $0.1E-9$

Direct Dynamic Simulation Approach



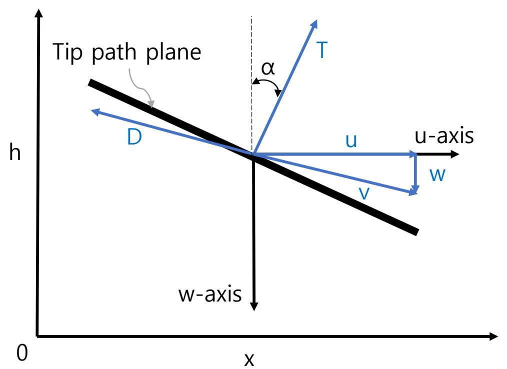
DDSA is around 30 times faster than PS method

Cost Function: 0.8246E+1
CPU Time : 22.0 [sec]

Direct Dynamic Simulation Approach (DDSA) to NOCP

Applications of DDSA using Point-Mass Model

Motion equations



$$\mathbf{x} = \begin{bmatrix} u \\ w \\ x \\ h \\ P \\ \Omega \\ C_T \\ \alpha \end{bmatrix} \qquad \mathbf{u} = \begin{bmatrix} \dot{C}_T \\ \dot{\alpha} \end{bmatrix}$$

Fig. Rotorcraft point mass model.

- **Motion equation & RPM dynamics**

$$m\dot{u} = C_T \rho \left(\pi R^2\right)\left(\Omega R\right)^2 \sin\alpha - \frac{1}{2}\rho f_e u\sqrt{u^2+w^2}$$

$$m\dot{w} = mg - C_T \rho \left(\pi R^2\right)\left(\Omega R\right)^2 \cos\alpha - \frac{1}{2}\rho f_e w\sqrt{u^2+w^2}$$

$$I_R \Omega\dot{\Omega} = P_S - \frac{1}{\eta}\rho\left(\pi R^2\right)\left(\Omega R\right)^3 C_P$$

- **Kinematics**

$$\dot{x} = u$$

$$\dot{h} = -w$$

- **Control equation**

$$\dot{C}_T = u_1$$

$$\dot{\alpha} = u_2$$

- **Power supply model**

$$\dot{P}_{fail} = -\frac{1}{\tau_P} P_S$$

$$\begin{cases} \dot{P}_{avail} = 0 & (t \le Pilot\ Delay) \\ \dot{P}_{avail} = \dfrac{1}{\tau_P}\left(P_{avail\_ref} - P_{avail}\right) & (t > Pilot\ Delay) \\ P_{avail\_ref} = \min\{P_{ref} - G(\Omega - \Omega_{ref}), P_{OEI}\} & \end{cases}$$

Model parameter

parameter	OH-58A	UH-60
Image		
Engine model	Single engine	Twin engines
f_e (equivalent flat plate area)	1.2077m^2	2.7871 m^2
m (helicopter mass)	1360.8 kg	7484.27 kg
R (main rotor radius)	5.3736 m	8.1777 m
σ (solidity)	0.048	0.0821
C_{d0} (drag coef.)	0.0087	0.012
a (lift curve)	5.73	5.73
I_R (main rotor MOI)	911.10 kg.m^3	9572.07 kg.m^3
H_R (main rotor height)	2.0 m	5.13 m
Ω_0 (ref, rotor RPM)	353 RPM	257.1 RPM
g (gravity coef.)	9.836 kg.m/s^2	9.836 kg.m/s^2
P_{ref} (ref. power)	354hp	-
P_{OEI} (OEI power)	-	1656hp
τ_P (time const.)	1.3	1.5

[Ref : Harris, Michael J., "Analytical Determination of a Helicopter Height Velocity Diagram" (2018). *Theses and Dissertations*. 1770.]
[Ref. : Robert T.N. Chen, Yiyuan Zhao, "Optimal Trajectories for the Helicopter in One-Engine-Inoperative Terminal-Area Operation," NASA/TM-96-110400, 1996]

Applications of DDSA using Point-Mass Model

Autorotational Landing Problem : NOCP Problem to minimize Touchdown speed

- Objective function

$$J = q_{fw} w_f + \frac{1}{2} Q_{fw} w_f^2 + \frac{1}{2} Q_{fu} u_f^2$$

$$where \quad q_{fw} = 40,\ Q_{fw} = 80,\ Q_{fu} = 80$$

- Initial/final constraints

$$\mathbf{x}_0 = \mathbf{x}_{trim}$$

$$h_f = 0$$

$$0m/s \le u_f \le 7.3152 m/s$$

$$0m/s \le w_f \le 0.9144 m/s$$

$$-10.0\deg \le \alpha_f \le 3.65\deg$$

- Global inequality constraint

$$w \le 9.144 m/s$$

$$0m \le h$$

$$0 \le C_T \le 0.15\sigma$$

$$-30\deg \le \alpha \le 30\deg$$

$$-25\sigma \deg/\sec \le \dot{C}_T \le 25\sigma \deg/\sec$$

$$-16\deg/\sec \le \dot{\alpha} \le 16\deg/\sec$$

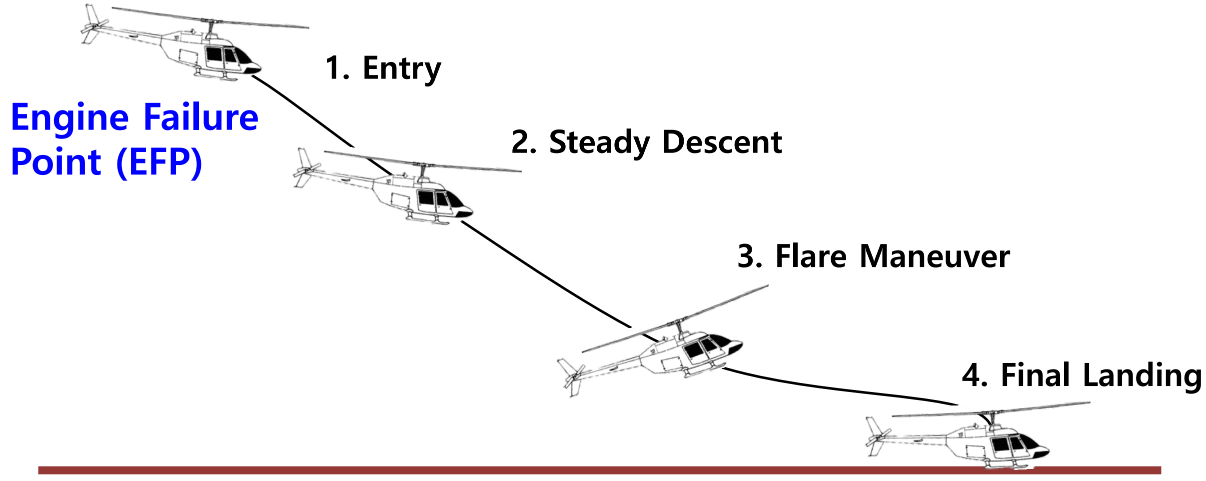


Fig. 7 Trajectory of autorotation procedure.

Aim of each phase

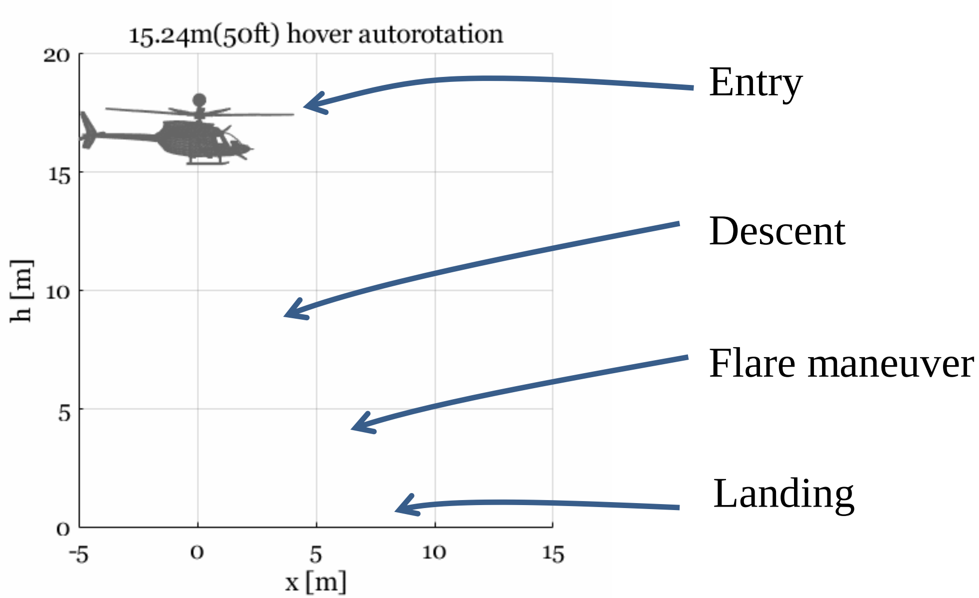
1. Entry : Recover 100% RPM while Stabilizing the aircraft
2. Steady Descent : increase the translation kinetic energy as much as possible.
3. Flare maneuver : reduce the speed and sink rate by increasing the collective pitch and by tilting rotor disc backward
4. Final landing : safe landing while keeping the attitude suitable for landing

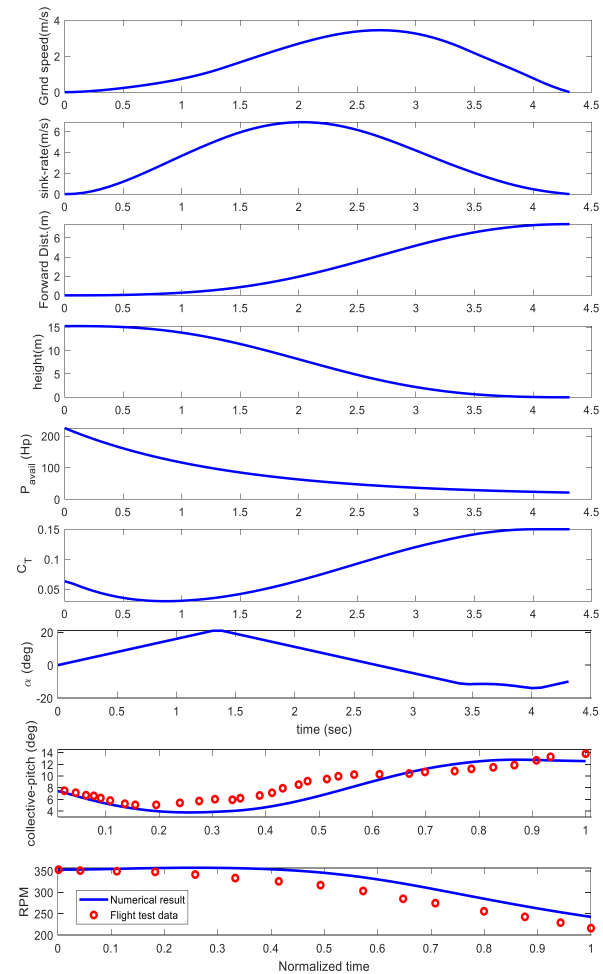
[Ref. : Edward N. Bachelder, Bimal L.Aponso,"An Autorotation Flight Director for Helicopter Training," the American Helicopter Society 59th Annual Forum proceedings, Phoenix, Arizona, May 6–8, 2003.]

Direct Dynamic Simulation Approach (DDSA) to NOCP

Applications of DDSA using Point-Mass Model

Autorotational Landing Problem : OH-58A, at low altitude hover point





Comparison of numerical results with flight test data.

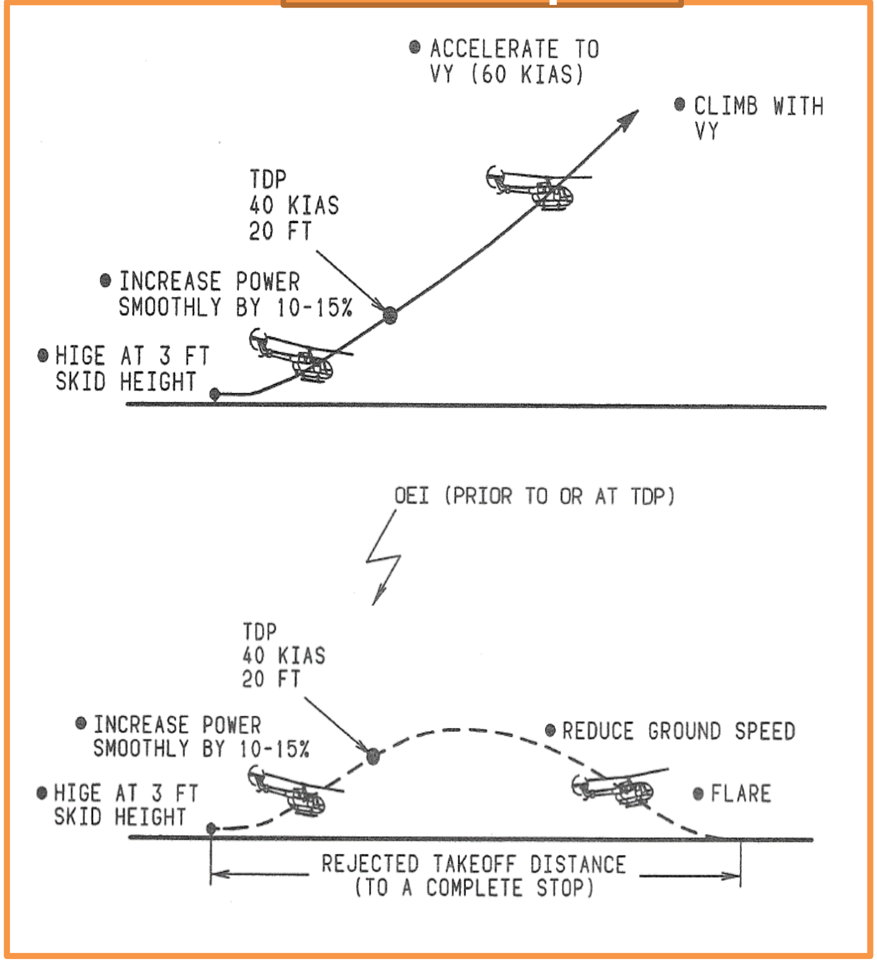
	Numerical result	Flight test data
Gross weight	1360.777 kg	1382.55 kg
Wind condition	0 knots	<3 knots
Flight time	4.3107 sec	8.1 sec
Max. sink rate	6.8796 m/s	6.096 m/s
Vertical Speed at T.D	0 m/s	0 m/s
Rotor RPM at T.D	242RPM	217 RPM
Min. Collective pitch	3.7665 deg	5.05 deg
Collective pitch at T.D	12.5375 deg	14.8 deg
Computation time	18.60 sec	.

[Ref. of flight test data: L. W. Dooley and R. D. Yeary, “Flight Test Evaluation of the High Inertia Rotor System,” Technical report, U.S. Army Research and Technology Laboratories (AVRADCOM), 1979]

Applications of DDSA using Point-Mass Model

Rejected Take-Off (RTO) Procedure after One Engine Failure [Bo-105 Flight Manual]

Clear heliport



Elevated heliport

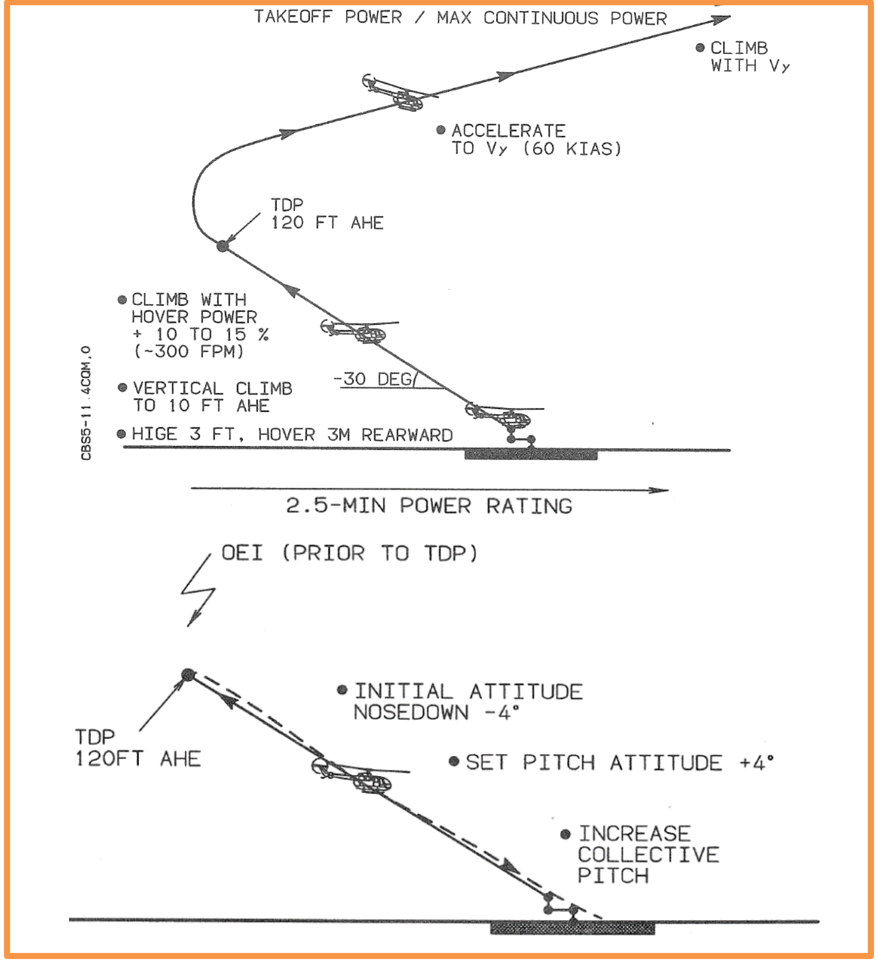


Fig. Trajectory of normal take-off procedure (Up) and RTO procedure (Down).

Direct Dynamic Simulation Approach (DDSA) to NOCP

Applications of DDSA using Point-Mass Model

RTO (Rejected Take-Off) Performance : Clear Heliport, 1sec Pilot delay, V=40 knots

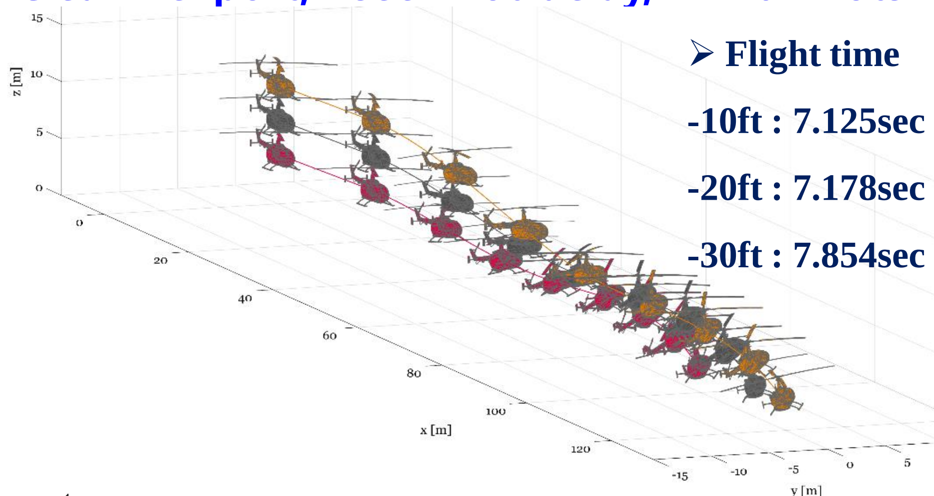
$$J = \frac{1}{2}\mathbf{x}_f^T \mathbf{Q}_f \mathbf{x}_f + \int_{t_o}^{t_f} \frac{1}{2}\mathbf{x}^T \mathbf{Q}_i \mathbf{x}\ dt$$

$$\mathbf{x}_f = [V_N, V_E, \dot{h}, \theta, \psi]^T, \mathbf{Q}_f = Dig[2,2,2,2,10]$$

$$\mathbf{x} = [p,q,r,y,\Omega,\mathbf{u}]^T, \mathbf{Q}_i = Dig[2,2,10,0.2,0.2,2]$$

$$\mathbf{u} = [\delta_0, \delta_{1c}, \delta_{1s}, \delta_{TR}]$$

$$V_{initial} = 40knot, ROC = 100\ fpm$$

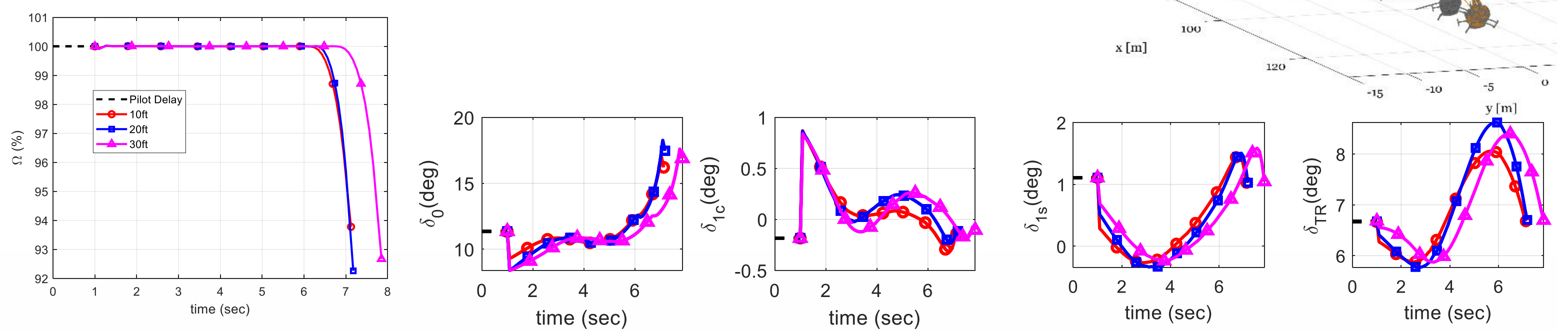


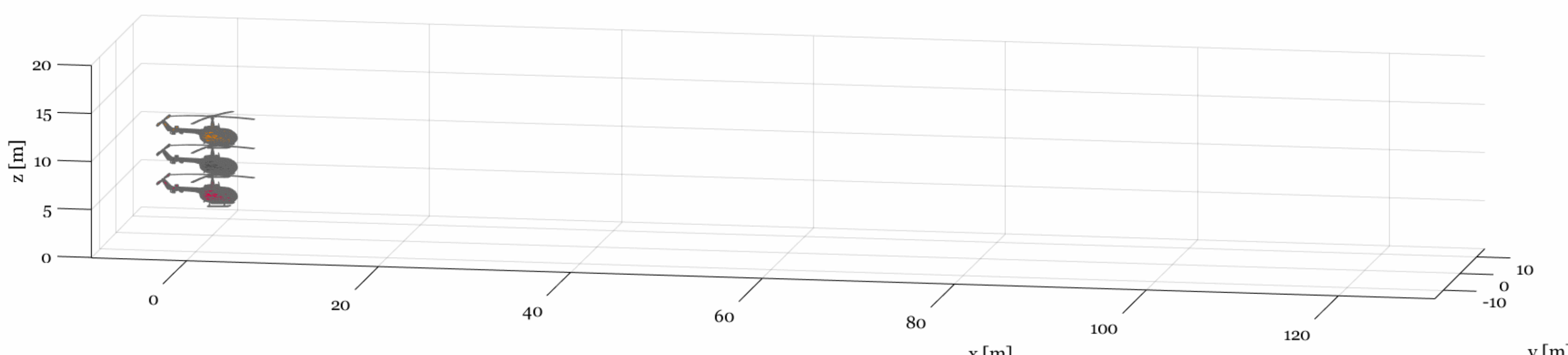
- **Flight time**

-10ft : 7.125sec

-20ft : 7.178sec

-30ft : 7.854sec





Direct Dynamic Simulation Approach (DDSA) to NOCP

Applications of DDSA using Point-Mass Model

RTO (Rejected Take-Off) Performance : Elevated Heliport, 1sec Pilot delay, height variation

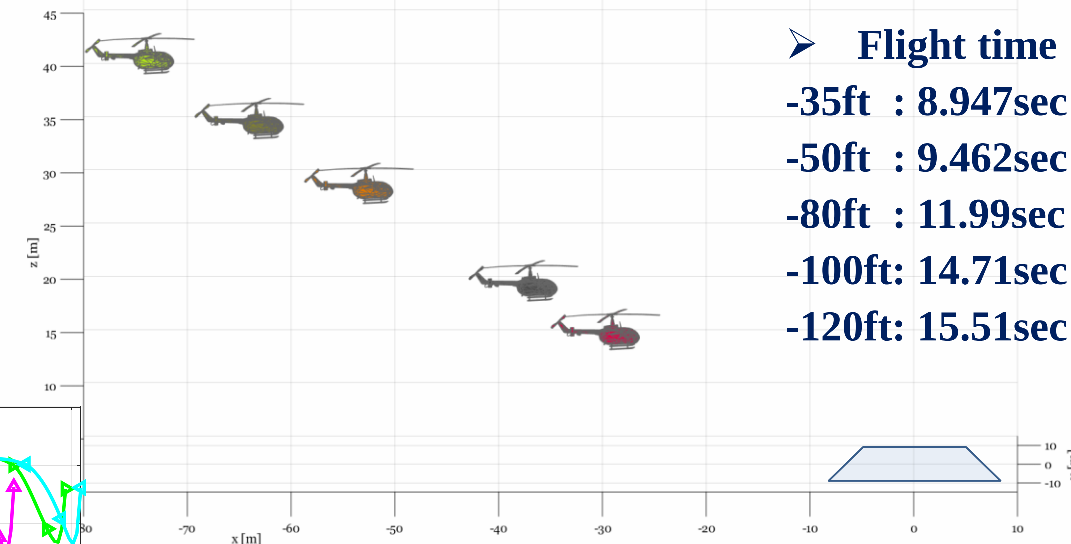
$$J = \frac{1}{2}\mathbf{x}_f^T \mathbf{Q}_f \mathbf{x}_f + \int_{t_o}^{t_f} \frac{1}{2}\mathbf{x}^T \mathbf{Q}_i \mathbf{x}\, dt$$

$$\mathbf{x}_f = [V_N, V_E, \dot{h}, \psi, x]^T, \mathbf{Q}_f = Dig[2,2,2,16,1]$$

$$\mathbf{x} = [p, q, r, y, \Omega, \mathbf{u}]^T, \mathbf{Q}_i = Dig[2,2,10,0.2,0.2,2]$$

$$\mathbf{u} = [\delta_0, \delta_{1c}, \delta_{1s}, \delta_{TR}]$$

$$V_{initial} = -5knot, ROC = 300\ fpm$$



- **Flight time**

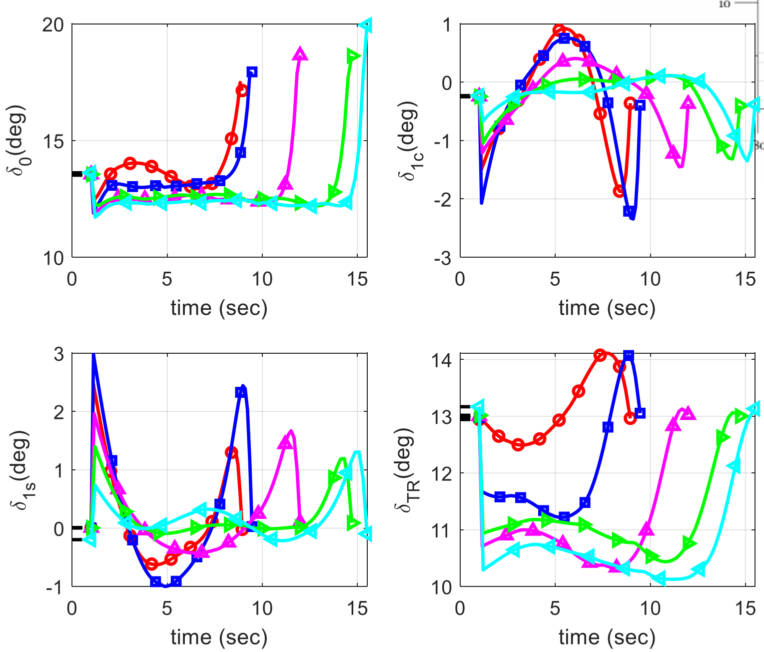
-35ft : 8.947sec

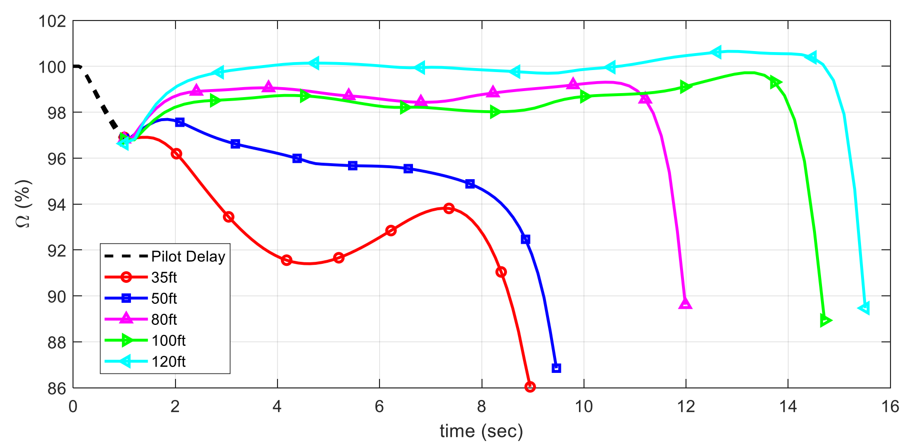
-50ft : 9.462sec

-80ft : 11.99sec

-100ft: 14.71sec

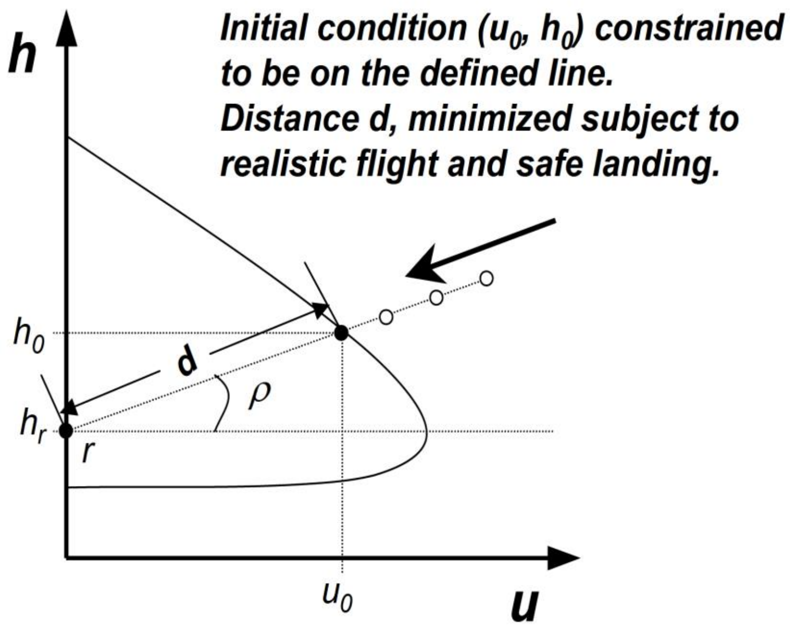
-120ft: 15.51sec





Applications of DDSA using Point-Mass Model

Height-Velocity (H-V) Diagram (Dead-Man Curve)



NOCP formulation for H-V Diagram

$\min \quad d = u_0 \cos\rho_r + (h_0 - h_r)\sin\rho_r$	Cost Function
subject to $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u},\mathbf{t})$	Motion Equations
$\mathbf{x}_0 = \mathbf{x}_{trim}$ $u_0 \sin\rho_r - (h_0 - h_r)\cos\rho_r = 0$	Initial Constraints
$w_f \le w_{\max}$ $u_f \le u_{\max}$	Final Constraints
$\alpha_{f_{\min}} \le \beta_f \le \alpha_{f_{\max}}$ $\Omega_{\min} \le \Omega(t) \le \Omega_{\max}$ $C_{T\min} \le C_T(t) \le C_{T\max}$ $\alpha_{\min} \le \alpha(t) \le \alpha_{\max}$ $-\dot{C}_{T\max} \le \dot{C}_T(t) \le \dot{C}_{T\max}$ $-\dot{\alpha}_{\max} \le \dot{\alpha}(t) \le \dot{\alpha}_{\max}$	Operational Limits

[Ref :Harris, M. J., Kunz, D. L., & Hess, J. A. (2018). Analytical Determination of a Helicopter Height-Velocity Curve. 2018 Modeling and Simulation Technologies Conference.]

Methodologies for MTE Analysis

Applications of DDSA using Point-Mass Model

Prediction H-V Diagram for OH-58A Model

NOCP formulation
(Minimize avoid zone)

$$\min J = 10.0d + \int_0^{t_f} (\Omega - \Omega_{ref})^2 dt$$

$$d = \cos\rho (u_o - u_{ref}) + \sin\rho (h_o - h_{ref})$$

$$u_{ref} = 0 knots$$

$$h_{ref} = 150\,ft$$

$$-80° \le \rho \le 80°$$

Constraints

$$h_f = 0$$

$$0\,ft/s \le u_f \le 20\,ft/s$$

$$0\,ft/s \le w_f \le 5\,ft/s$$

$$-10° \le \alpha_f \le 3.65°$$

$$u_f \le 35\,ft/s(ref)$$

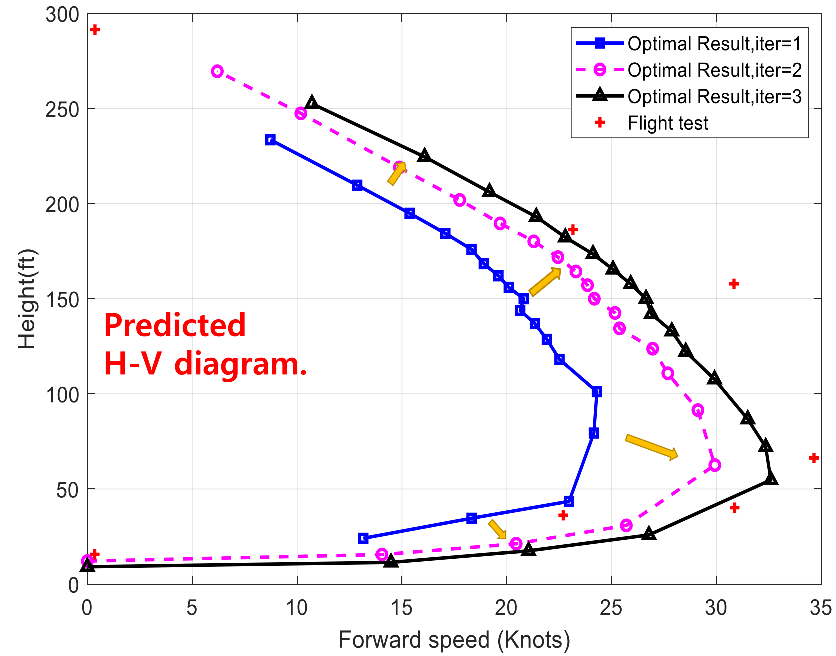
$$-30° \le \alpha \le 30°$$

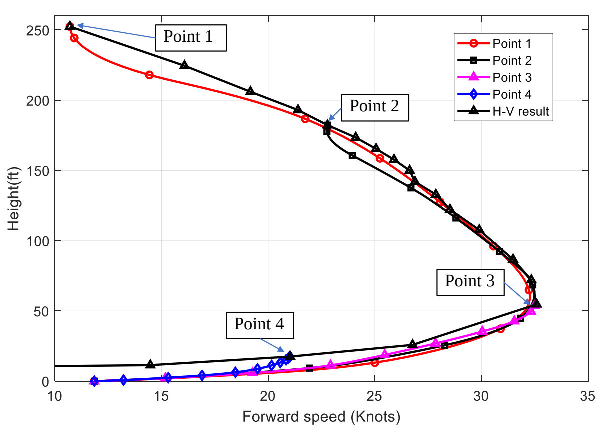
$$-16°/s \le \dot{\alpha} \le 16°/s$$

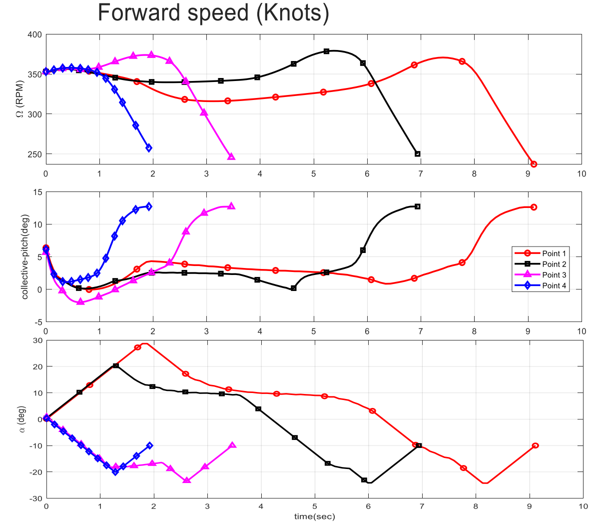
$$0.2 \le \frac{C_T}{\sigma} \le 0.15$$

$$-0.43/s \le \frac{\dot{C}_T}{\sigma} \le 0.43/s$$

$$\boxed{\sin\rho(u_o - u_{ref}) = \cos\rho(h_o - h_{ref})}$$



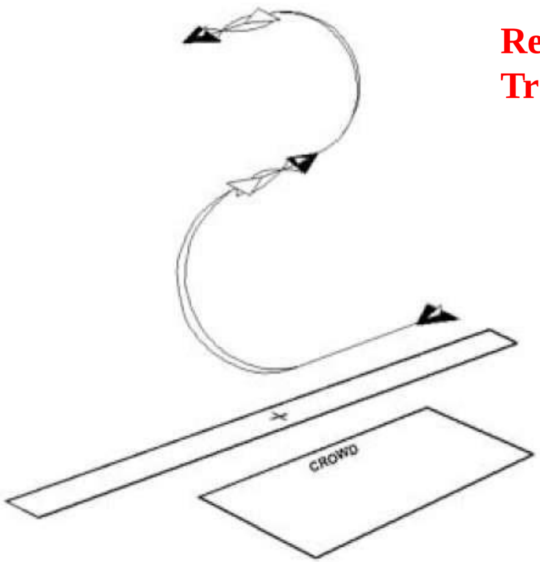




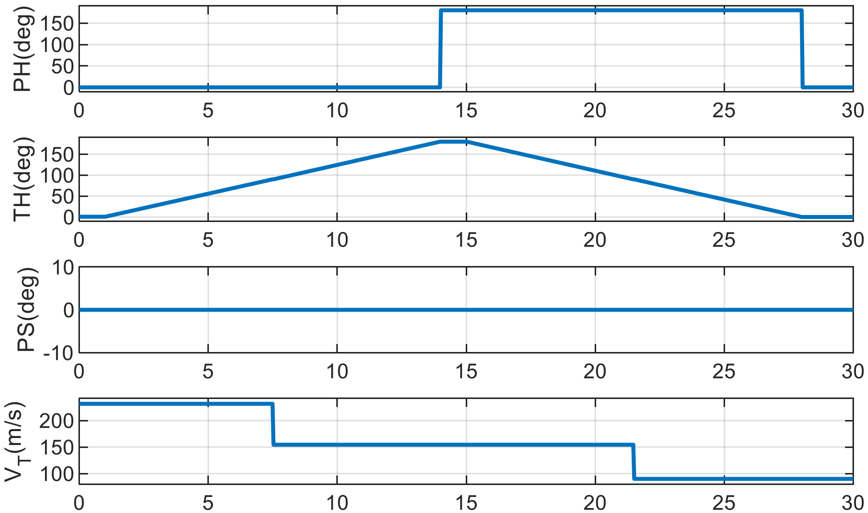
Applications of DDSA using High-Fidelity F-16 Model

Double Immelmann Turn (Ref: US Air Force Aircraft Demonstrations)

- Entry phase: 450knots Level flight
- 180 deg Heading change through Longitudinal loop maneuver
- 180 deg bank change
- Repeat above procedure
- Use 100 % throttle after entry and use throttle greater than 77% after Apex.



Reference Trajectory

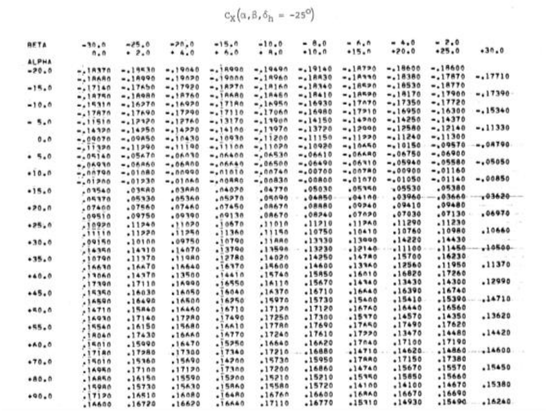


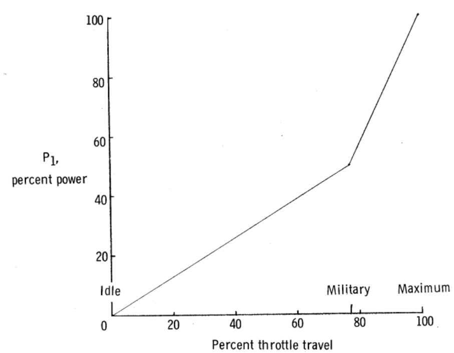
[Ref 1] Brian L. Stevens, 『*Aircraft Control and Simulation 3rd Edition*』, WILEY, November 2015
[Ref 2] Nguyen L. T, *Simulator Study of Stall/Post-Stall Characteristics of a Fighter Airplane With Relaxed Longitudinal Static Stability*, NASA Technical Paper 1538.
[Ref 3] Misawa Airbase U.S. Air Force, (2021). PACAF F-16 Demonstration Team Maneuvers Package 2021, U.S Air Force, 23 October 2014, AIR FORCE AIRCRAFT DEMONSTRATIONS (A-10, F-15, F-16, F-22)

Direct Dynamic Simulation Approach (DDSA) to NOCP

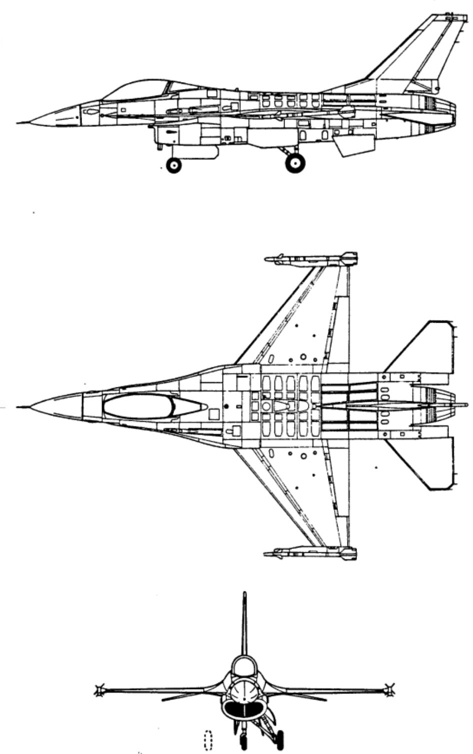
Applications of DDSA using High-Fidelity F-16 Model

High-Fidelity F-16 Model: opened at NASA Homepage





(b) Power variation with throttle position.



F-16 Configuration Data	
Weight	20,494 lb
WingSpan	32 ft 8 in
Wing area	300 ft^2
Airfoil	NACA 64A204
MTOW	42,300 lb
XCG	35.0% MGC
MAC	11.32 ft
I_{xx}	9496 Slug ft^2
I_{yy}	55814 Slug ft^2
I_{zz}	63100 Slug ft^2
I_{zx}	982 Slug ft^2
Elevator Deflection	-25 deg to 25 deg
Aileron Deflection	-21.5 deg to 21.5 deg
Rudder Deflection	-30 deg to 30 deg

[Ref 1] Brian L. Stevens, 『*Aircraft Control and Simulation 3rd Edition*』, WILEY, November 2015

[Ref 2] Nguyen L. T, *Simulator Study of Stall/Post-Stall Characteristics of a Fighter Airplane With Relaxed Longitudinal Static Stability*, NASA Technical Paper 1538.

[Ref 3] Misawa Airbase U.S. Air Force, (2021). PACAF F-16 Demonstration Team Maneuvers Package 2021, U.S Air Force, 23 October 2014, AIR FORCE AIRCRAFT DEMONSTRATIONS (A-10, F-15, F-16, F-22)

Applications of DDSA using High-Fidelity F-16 Model

NOCP Formulation of Double Immelmann Turn

min $\quad J = \frac{1}{2}\int_0^{t_f}\left[\mathbf{x}_{diff}^T(t)\mathbf{Q}(t)\mathbf{x}_{diff}(t) + \mathbf{u}_{diff}^T(t)\mathbf{R}(t)\mathbf{u}_{diff}(t)\right]dt$

s.t.

Dyanmic Constraints: $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u})$

Inequality Constraints: $\mathbf{u} \le \mathbf{u}_{\max}$, $\quad \mathbf{u} \ge \mathbf{u}_{\min}$

Equality Constraints: $\mathbf{x}(0) = \mathbf{x}_{trim}$, $\quad \mathbf{u}(0) = \mathbf{u}_{trim}$

where

t_f is fixed

$$\mathbf{x}_{diff}(t) = \mathbf{x}(t) - \mathbf{x}_{ref}(t), \qquad \mathbf{u}_{diff} = \mathbf{u}(t) - \mathbf{u}_{ref}$$

$$\mathbf{x} = (u, v, w, p, q, r, \phi, \theta, \psi, V_T, \alpha, \beta)^T$$

$$\mathbf{u} = (u_{ele}, u_{ail}, u_{rud}, u_{thr})^T$$

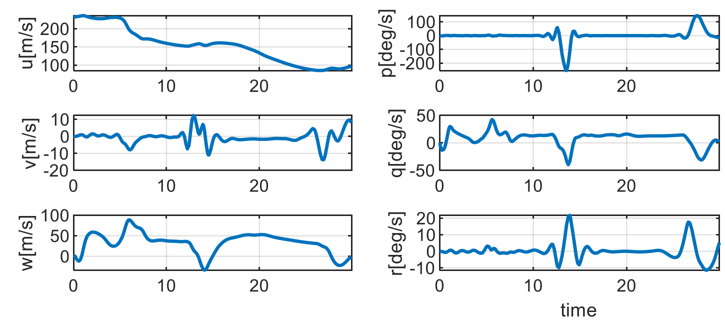
$$\mathbf{Q} = diag(0,0,0,w_p, w_q, w_r, w_\phi, w_\theta, w_\psi, w_{V_T}, 0, 0)$$

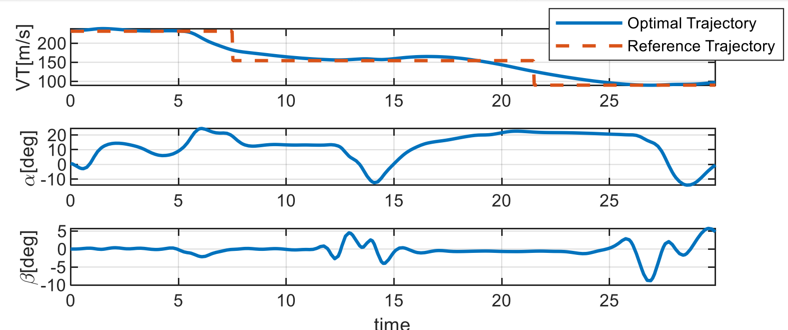
$$\mathbf{R} = diag(w_{ele}, w_{ail}, w_{rud}, w_{thr})$$

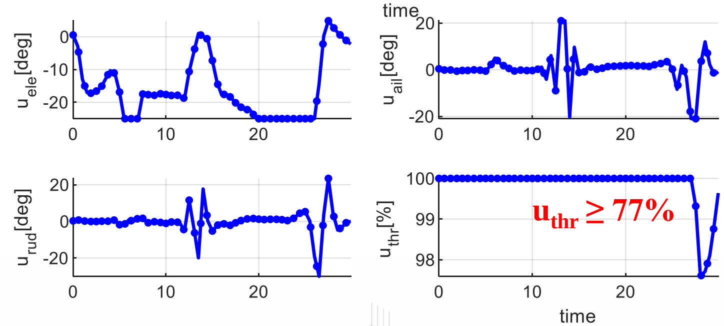
Direct Dynamic Simulation Approach (DDSA) to NOCP

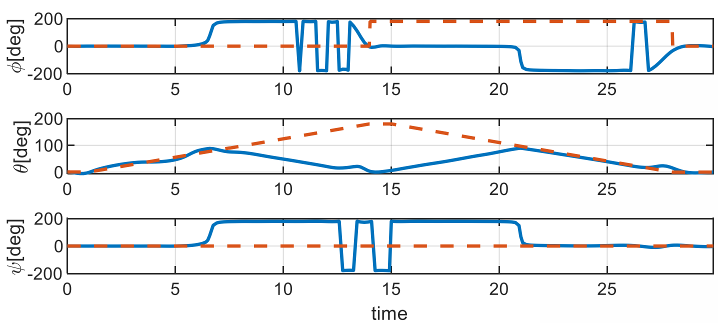
Applications of DDSA using High-Fidelity F-16 Model

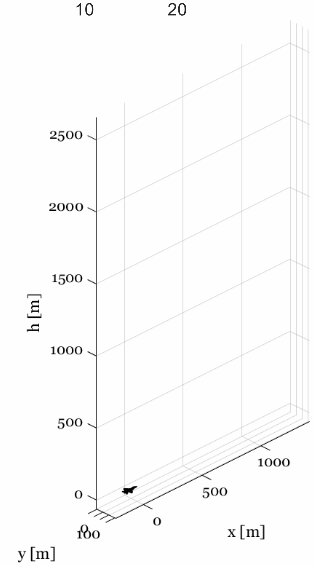
DDSA Results for Double Immelmann Turn

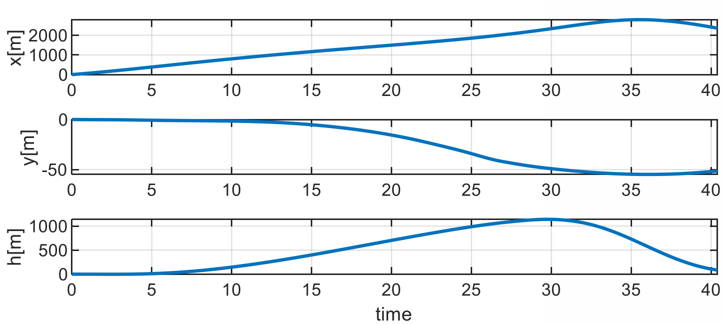












Contents

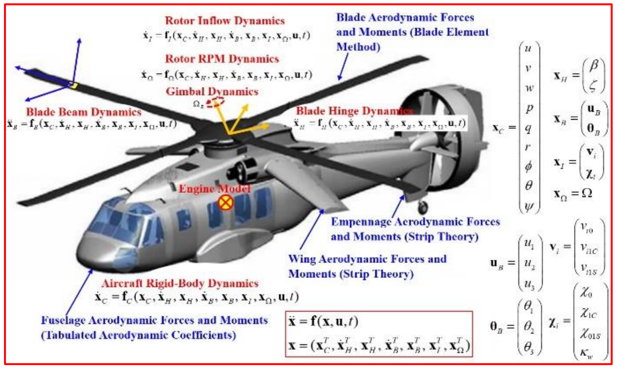
Recent Research Progresses in Rotorcraft Flight Dynamics and Autonomous Flight Control at KKU

Part 1: Rotorcraft Flight Dynamics

Summary

HETAS Math Model

- The rotor and wing models are generalized for HETLAS applications to Advanced rotorcraft configuration like the tilt-rotor aircraft and coaxial-prop rotorcraft.
- As primary functions of HETLAS, the trim, linearization and simulation routines are addressed.
- The trim mover function has been introduced for robust point and mission performance analyses.
- The coupled mission-performance-equation has been effectively solved using the pseudo-spectral integrator for the mission segment approach.
- The validation results for the fidelity of HETLAS has been presented.



- Definition of Mission Segments using Way-point Data

$$\left\{ \left(t_j, h_j, V_{G,j}, V_{ROC,j} \right) \right\}_{j=0}^{j=N_{WP}}$$ Data for height, ground speed, and rate of climb

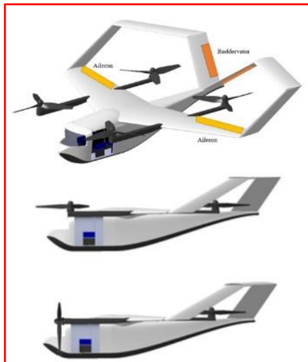
- Trajectory Generation using spline interpolation of h, V_G, V_{ROC}
- Time integration along the generated trajectory to get converged solutions of coupled mission-performance equations using PS-integrator

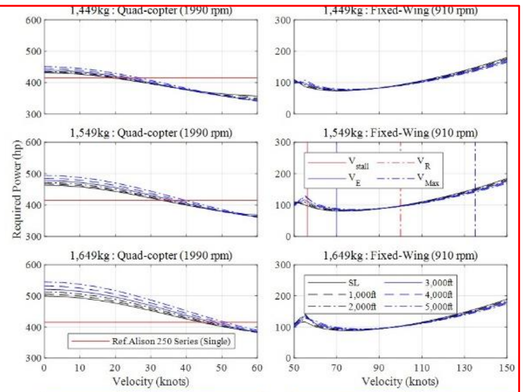
$$\frac{dm}{dt} = -SFC \times P \qquad \frac{dh}{dt} = V_{RoC} \qquad \frac{dR}{dt} = |V_G|$$

$$m(t) = m(t_0) - \int_{t_0}^{t_f} (SFC \times P)\,dt \qquad h(t) = h(t_0) + \int_{t_0}^{t_f} V_{RoC}\,dt \qquad R(t) = R(t_0) + \int_{t_0}^{t_f} |V_G|\,dt$$

- Validation of HETLAS: Example Rotorcrafts

Reference Helicopter	Bo-105
	
V&V: Comparison with Flight Test Criteria: 1) FAA AC-120-63 2) GENHEL (Sikorsky 社) 3) Boeing	V&V: Using Ref. (Flight test/Analysis) Criteria: FAA AC-120-63 Ref.: 1) AGARD GARTEUR Report 2) Published Papers





Kinematically Exact Inverse Simulation Techniques (KEIST)

- **The importance of maneuver analyses during the rotorcraft development has been emphasized.**
- **Two different approaches were introduced**

  (1) Inverse simulation approach **(2) Nonlinear optimal control approach**
- **Index 1 DAE (Differential-Algebraic Equation) systems have been derived by using**

  (1) Motion equations represented using the inertial states

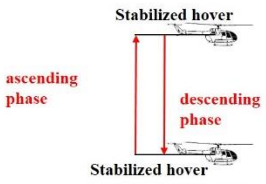
  (2) Exact trajectory information obtained using the 7-th order spline interpolation
- **KEIST has effectively solved the DAE system by using**

  (1) Quasi-Newton method for algebraic equations

  (2) the PS integrator coupled with the Piccard method
- **A series of applications showed efficiency and robustness of KEIST**

Kinematically Exact Inverse Simulation Techniques (KEIST)

Application to Pop-up MTE

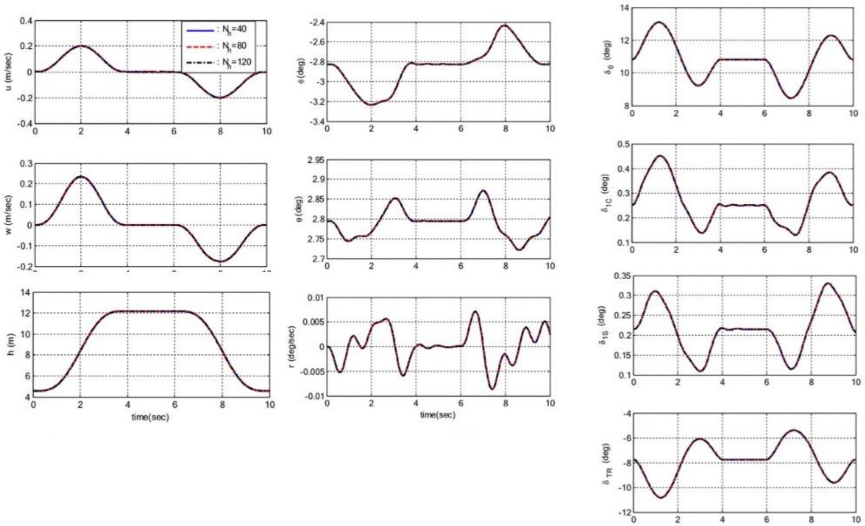


$v_1 = v_2 = 0$ knots
$h_1 = h_2 = 35$ ft
$t_1 = 4$ sec, $t_2 = 6$ sec

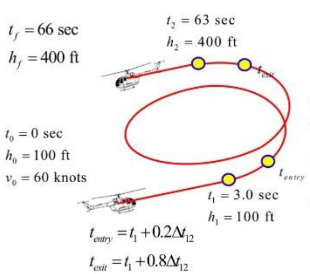
$v_0 = v_f = 0$ knots
$h_0 = h_f = 15$ ft
$t_0 = 0$ sec, $t_f = 10$ sec

Case	N	N_h	Δt_{avg} (s)	CPU time (Intel i7)/It$_{avg}$ KEIST formulation	Conventional formulation
1	4	20	0.1250	Failed at first segment	
2	4	40	0.0625	203/8.6	223/10.7
3	4	60	0.0417	141/7.7	170/9.8
4	4	80	0.0313	147/7.4	183/8.1
5	4	100	0.0250	155/6.5	201/8.1
6	4	120	0.0208	186/7.2	211/7.7

N = number of quadrature points
Nh = number of time horizon segments

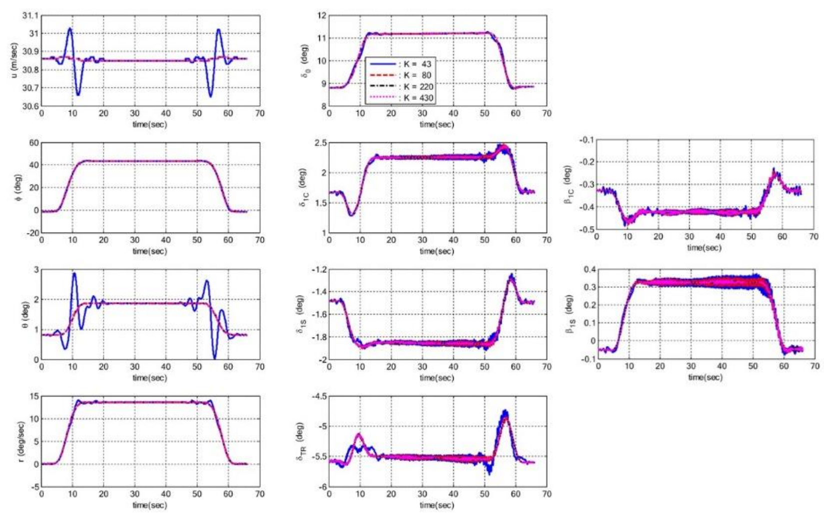


Application to Helical Turn MTE



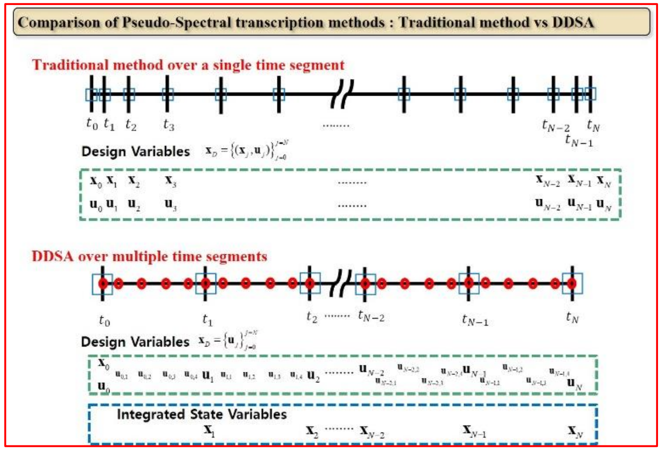
K = number of waypoint data

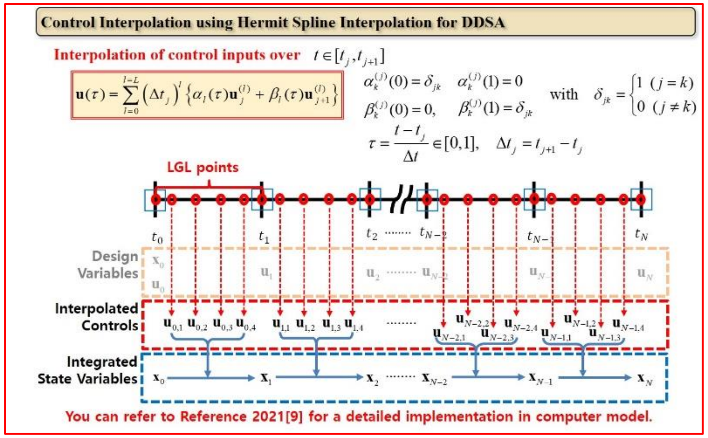
Case	K_0	K_{entry}	K_{steady}	K_{exit}	K_f	$K = K_{total}$
1	5	4	25	4	5	43
2	5	10	50	10	5	80
3	5	30	150	30	5	220
4	5	60	300	60	5	430



Direct Dynamic Simulation Approach (DDSA) to Rotorcraft Aggressive Maneuver Analysis

- **The efficient DDSA has been developed using the following two basic concepts.**
  - **(1) The system states are uniquely determined by the control inputs.**
  - **(2) Computational efficiency can be enhanced using controls interpolated with Hermit spline.**
- **The effectiveness of DDAS has been proved through a series of applications.**
  - ✓ Soft lunar landing problem of a spacecraft
  - ✓ Autorotational Landing Problem using a point-mass model
  - ✓ Rejected Take-Off (RTO) Procedure after One Engine Failure
  - ✓ Estimation of Height-Velocity (H-V) Diagram (Dead-Man Curve)
  - ✓ Double Immelmann Turn analysis using the high-fidelity F-16 model



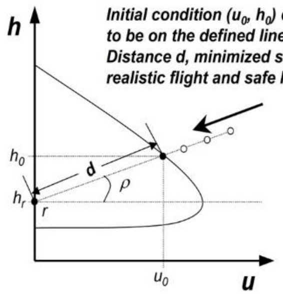


Summary of Research on Rotorcraft Flight Dynamics

Direct Dynamic Simulation Approach (DDSA) to Rotorcraft Aggressive Maneuver Analysis

Height-Velocity (H-V) Diagram (Dead-Man Curve)

Initial condition (u_0, h_0) constrained to be on the defined line. Distance d, minimized subject to realistic flight and safe landing.



NOCP formulation for H-V Diagram

min $d = u_0\cos\rho_r + (h_0 - h_r)\sin\rho_r$ — **Cost Function**

subject to

$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u},\mathbf{t})$ — **Motion Equations**

$\mathbf{x}_0 = \mathbf{x}_{trim}$

$u_0\sin\rho_r - (h_0 - h_r)\cos\rho_r = 0$ — **Initial Constraints**

$w_f \le w_{\max}$

$u_f \le u_{\max}$ — **Final Constraints**

$\alpha_{f_{\min}} \le \beta_f \le \alpha_{f_{\max}}$

$\Omega_{\min} \le \Omega(t) \le \Omega_{\max}$

$C_{T\min} \le C_T(t) \le C_{T\max}$

$\alpha_{\min} \le \alpha(t) \le \alpha_{\max}$

$-\dot{C}_{T\max} \le \dot{C}_T(t) \le \dot{C}_{T\max}$

$-\dot{\alpha}_{\max} \le \dot{\alpha}(t) \le \dot{\alpha}_{\max}$ — **Operational Limits**

[Ref :Harris, M. J., Kunz, D. L., & Hess, J. A. (2018). Analytical Determination of a Helicopter Height-Velocity Curve. 2018 Modeling and Simulation Technologies Conference.]

Prediction H-V Diagram for OH-58A Model

NOCP formulation
(Minimize avoid zone)

$$\min J = 10.0d + \int_0^{t_f}(\Omega - \Omega_{ref})^2\,dt$$

$d = \cos\rho(u_o - u_{ref}) + \sin\rho(h_o - h_{ref})$

$u_{ref} = 0knots$

$h_{ref} = 150ft$

$-80^\circ \le \rho \le 80^\circ$

Constraints

$h_f = 0$

$0ft/s \le u_f \le 20ft/s$

$0ft/s \le w_f \le 5ft/s$

$-10^\circ \le \alpha_f \le 3.65^\circ$

$u_f \le 35ft/s(ref)$

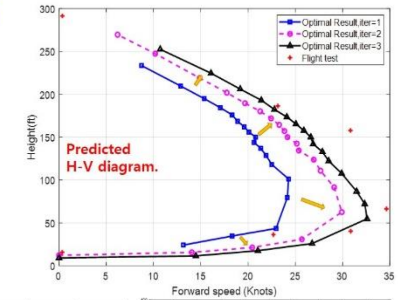
$-30^\circ \le \alpha \le 30^\circ$

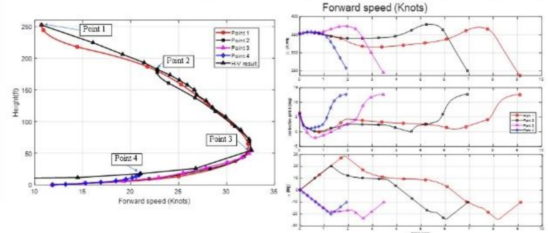
$-16^\circ/s \le \dot{\alpha} \le 16^\circ/s$

$0.2 \le \frac{C_T}{\sigma} \le 0.15$

$-0.43/s \le \frac{\dot{C}_T}{\sigma} \le 0.43/s$

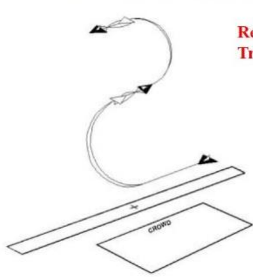
$$\sin\rho(u_o - u_{ref}) = \cos\rho(h_o - h_{ref})$$



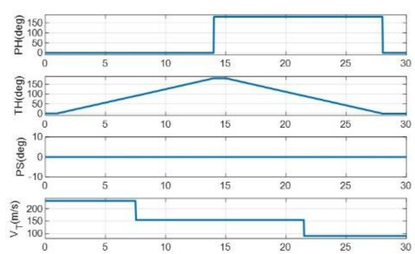


Double Immelmann Turn (Ref: US Air Force Aircraft Demonstrations)

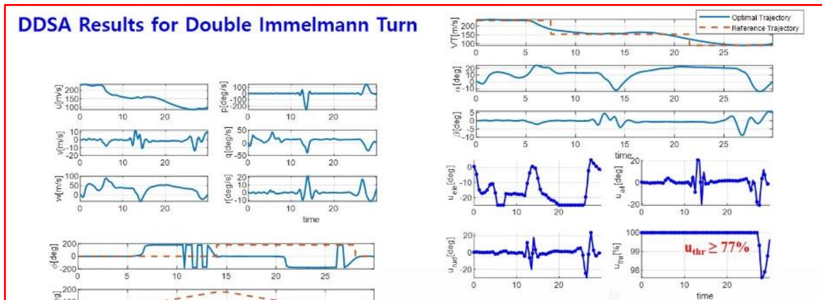
- Entry phase: 450knots Level flight
- 180 deg Heading change through Longitudinal loop maneuver
- 180 deg bank change
- Repeat above procedure
- Use 100 % throttle after entry and use throttle greater than 77% after Apex.

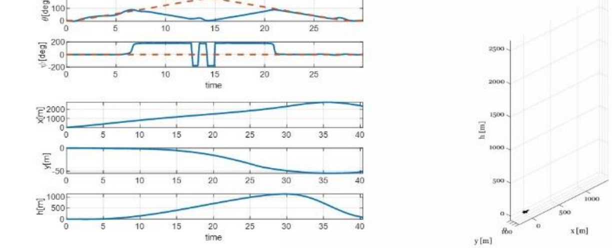


Reference Trajectory



DDSA Results for Double Immelmann Turn





End of Part 1

Thank You !!

Recent Research Progresses in Rotorcraft Flight Dynamics and Autonomous Flight Control at KKU

2024. 02

Prof. Chang-Joo Kim (Konkuk University, Seoul, Korea)

Recent Research Progresses in Rotorcraft Flight Dynamics and Autonomous Flight Control at KKU

Part 2: Rotorcraft Autonomous Flight Control System

2024. 02

Prof. Chang-Joo Kim (Konkuk University, Seoul, Korea)

Contents

Initial Motivation for Autonomous FCS Research

Questions to be Answered

What is the Autonomous FCS ?

What is the required Autonomous FCS Structure?

What is the Functional Requirements for the Autonomous FCS ?

What we have for the Design and Development of the Autonomous FCS ?

What is the Best KKU Approach to the Autonomous FCS ?

We spent around one year finding answers to these questions !!

Good References

[Ref 1] Farid Kendoul, “Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems,” Journal of Field Robotics,2012, No. 29, Vol. 2, pp 315-378.

[Ref 2] Takahashi, Marc D., et al. "Autonomous Rotorcraft Flight Control with Multilevel Pilot Interaction in Hover and Forward Flight." Journal of the American Helicopter Society 62.3 (2017): 1-13.

Initial Motivation for Autonomous FCS Research

Kendoul's Classifications of 11 Autonomy Levels (ALs)

Level	Level Descriptor	Guidance	Navigation	Control	ESI	EC	MC
10	Fully Autonomous	Human-level decision-making, accomplishment of most missions without any intervention from ES (100% ESI), cognizant of all within the operation range.	Human-like navigation capabilities for most missions, fast SA that outperforms human SA in extremely complex environments and situations.	Same or better control performance as for a piloted aircraft in the same situation and conditions.	approaching 100% ESI	extreme environment	highest complexity, all missions
9	Swarm Cognizance and Group Decision Making	Distributed strategic group planning, selection of strategic goals, mission execution with no supervisory assistance, negotiating with team members and ES.	Long track awareness of very complex environments and situations, inference and anticipation of other agents intents and strategies, high-level team SA.	Ability to choose the appropriate control architecture based on the understanding of the current situation/context and future consequences.			
8	Situational Awareness and Cognizance	Reasoning and higher level strategic decision-making, strategic mission planning, most of supervision by RUAS, choose strategic goals, cognizance.	Conscious knowledge of complex environments and situations, inference of self/others intent, anticipation of near-future events and consequences (high fidelity SA).	Ability to change or switch between different control strategies based on the understanding of the current situation/context and future consequences.	high level ESI	difficult environment	collaborative, high complexity missions
7	RT Collaborative Mission Planning	Collaborative mission planning and execution, evaluation and optimization of multi-vehicle mission performance, allocation of tactical tasks to each agent.	Combination of capabilities in levels 5 and 6 in highly complex, adversarial and uncertain environment, collaborative mid fidelity SA.	same as in previous levels (no-additional control capabilities are required)			
6	Dynamic Mission Planning	Reasoning, high-level decision making, mission driven decisions, high adaptation to mission changes, tactical task allocation, execution monitoring.	Higher-level of perception to recognize and classify detected objects/events and to infer some of their attributes, mid fidelity SA.	same as in previous levels (no-additional control capabilities are required)			
5	RT Cooperative Navigation and Path Planning	Collision avoidance, cooperative path planning and execution to meet common goals, swarm or group optimization.	Relative navigation between RUAS, cooperative perception, data sharing, collision detection, shared low fidelity SA.	Distributed or centralised flight control architectures, coordinated maneuvers.	mid level ESI	moderate environment	mid complexity, multi-functional missions
4	RT Obstacle/Event Detection and Path Planning	Hazard avoidance, RT path planning and re-planning, event driven decisions, robust response to mission changes.	Perception capabilities for obstacle, risks, target and environment changes detection, RT mapping (*optional*), low fidelity SA.	Accurate and robust 3D trajectory tracking capability is desired.			
3	Fault/Event Adaptive RUAS	Health diagnosis, limited adaptation, onboard conservative and low-level decisions, execution of pre-programmed tasks.	Most health and status sensing by the RUAS, detection of hardware and software faults.	Robust flight controller, reconfigurable or adaptive control to compensate for most failures, mission and environment changes.			
2	ESI Navigation (*e.g., Non-GPS*)	Same as in Level 1	All sensing and state estimation by the RUAS (no ES such as GPS), all perception and situation awareness by the human operator.	Same as in Level 1	low level ESI	simple environment	low level tasks
1	Automatic Flight Control	Pre-programmed or uploaded flight plans (waypoints, reference trajectories, etc.), all analyzing, planning and decision-making by ES.	Most sensing and state estimation by the RUAS, all perception and situational awareness by the human operator.	Control commands are computed by the flight control system (automatic control of the RUAS 3D pose).			
0	Remote Control	All guidance functions are performed by external systems (mainly human pilot or operator)	Sensing may be performed by the RUAS, all data is processed and analyzed by an external system (mainly human).	Control commands are given by a remote ES (mainly human pilot).	0% ESI	lowest EC	lowest MC

- **Autonomy**

  The condition or quality of being self governing

- **Autonomy Level (AL)**

  A set of progressive indices, typically numbers and/or names, identifying a UAS capability of performing autonomously assigned mission.

- **AL characteristics**

  ALs 1-4: Single Vehicle

  ALs 5-7: Multi Vehicles

  Als 8-10: High-level/Fully Autonomous

Required Functions

- **Guidance Function**

  Real-time Path Planning : Rapidly Exploring Random Trees(RRT) / PRM (Probability Road Map) (AL 4)

- **Navigation Function**

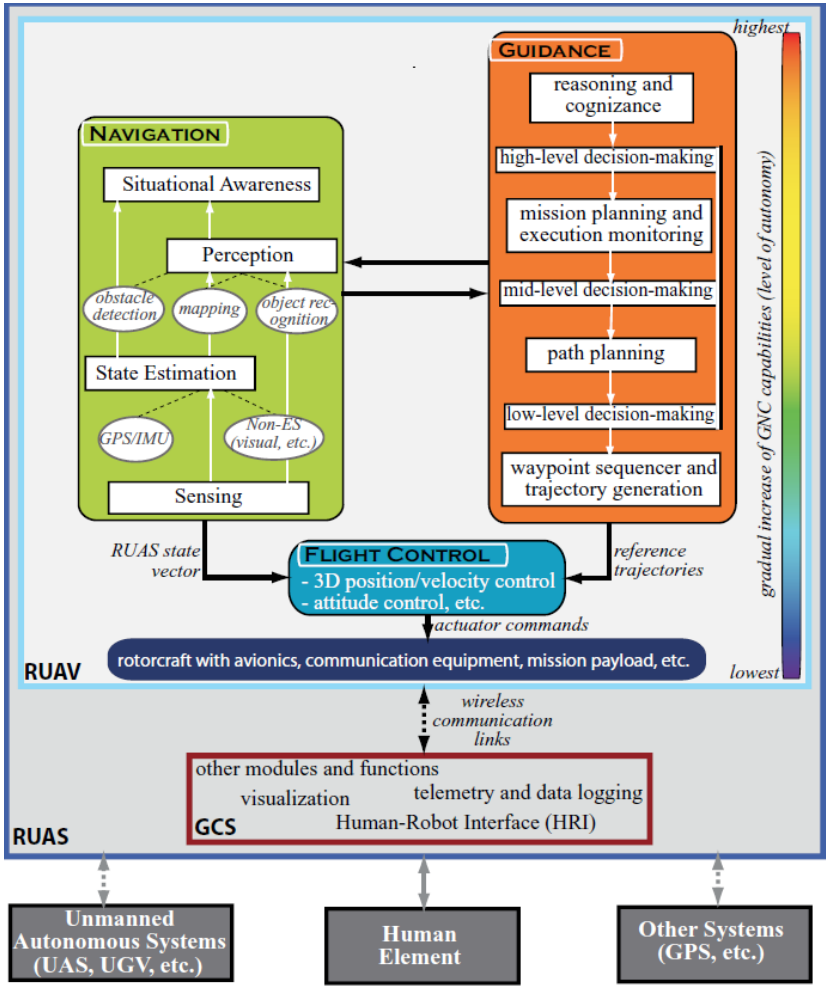
  IMU/GPS integrated with Digital map-based / Use environmental information from outside sources(AL 3~4)

- **Control Function**

  Real-time Trajectory-Tracking Control (AL 3~4)

[Ref 1] Farid Kendoul, "Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems,"

Kendoul's Proposition for UAS Autonomous FCS Structure



- **Rotorcraft Unmanned Aerial Vehicle(RUAV)**

  A powered rotorcraft that does not require an onboard crew, can operate with some degree of autonomy, and can be expendable or reusable.

- **Rotorcraft Unmanned Aerial or Aircraft System(RUAS)**

  A RUAS is a physical system that includes a RUAV, communication architecture, and a ground control station with no human element aboard any component.

- **Navigation System(NS): Perception & State Estimation**

  The process of monitoring and controlling the movement of a craft or vehicle from one place to another.

- **Guidance System(GS)**

  The "driver" of a RUAS that exercises **Mission/Path planning and decision-making functions** to achieve assigned missions or goals.

- **Autonomous Flight Control System(AFCS)**

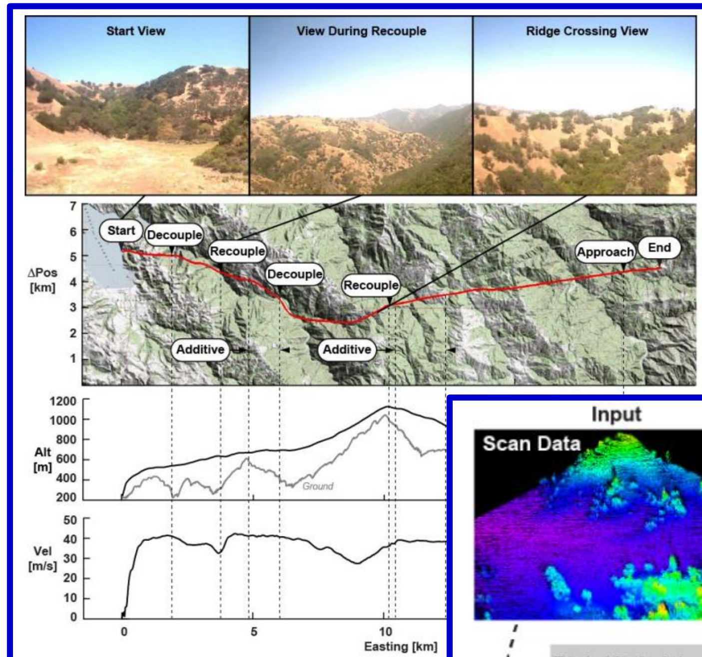
  The process of manipulating the inputs to a dynamic system to obtain a desired effect on its outputs without a human in the control loop.

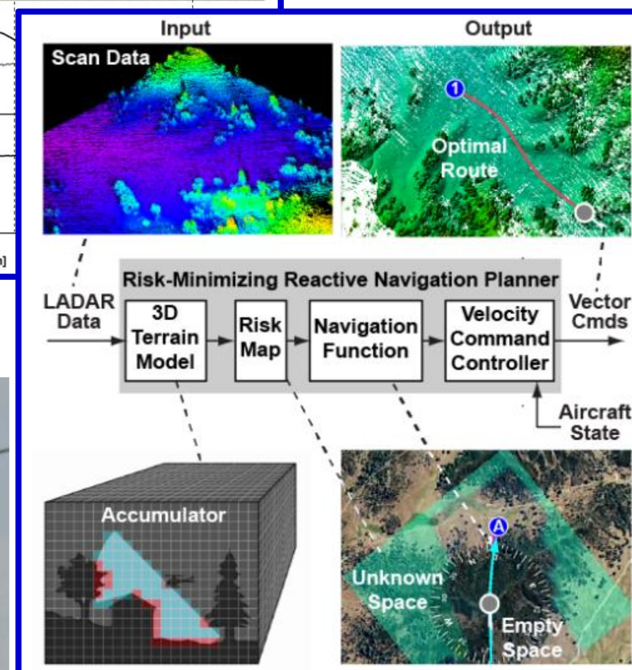
[Ref 1] Farid Kendoul, "Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems,"

Initial Motivation for Autonomous FCS Research

Autonomous FCS Structure of RASCAL JUH-60A Black Hawk (US Army)

- Multi-Level Autonomy
  - ✓ Fully Coupled Autonomous Mode
  - ✓ Additive Control Mode
  - ✓ Decoupled ACAH Mode
  - ✓ Pilot Interaction with Mode
  - ✓ Control System Design with Mode Transitions
- Mission S/W
  - ✓ Mission Manager/Operator Interface
  - ✓ Obstacle Field Navigation (OFN)
  - ✓ Safe Landing Area Determination (SLAD)
  - ✓ Path Generation
  - ✓ Vector Command
- Autonomous Flight Control S/W (AFCS)
  - ✓ Waypoint Control
  - ✓ Tracking Control
  - ✓ Inner-Loop Control

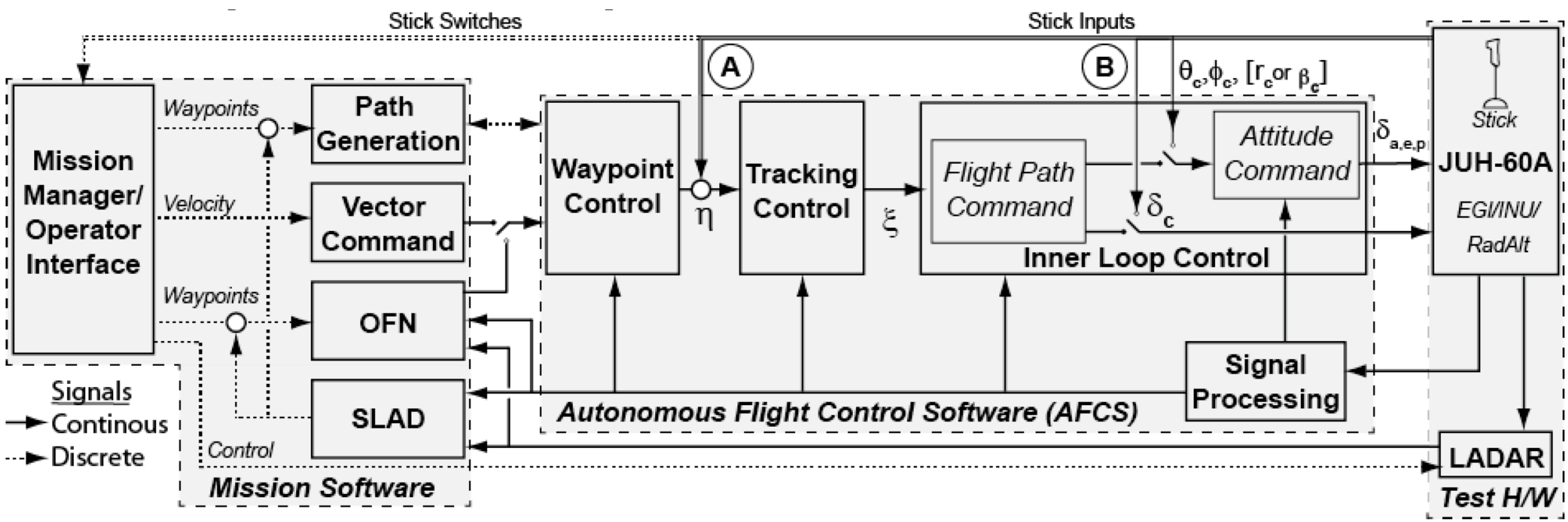




[Ref 2] Takahashi, Marc D., et al. "Autonomous Rotorcraft Flight Control with Multilevel Pilot Interaction in Hover and Forward Flight."

Initial Motivation for Autonomous FCS Research

Autonomous FCS Structure of RASCAL JUH-60A Black Hawk (US Army)



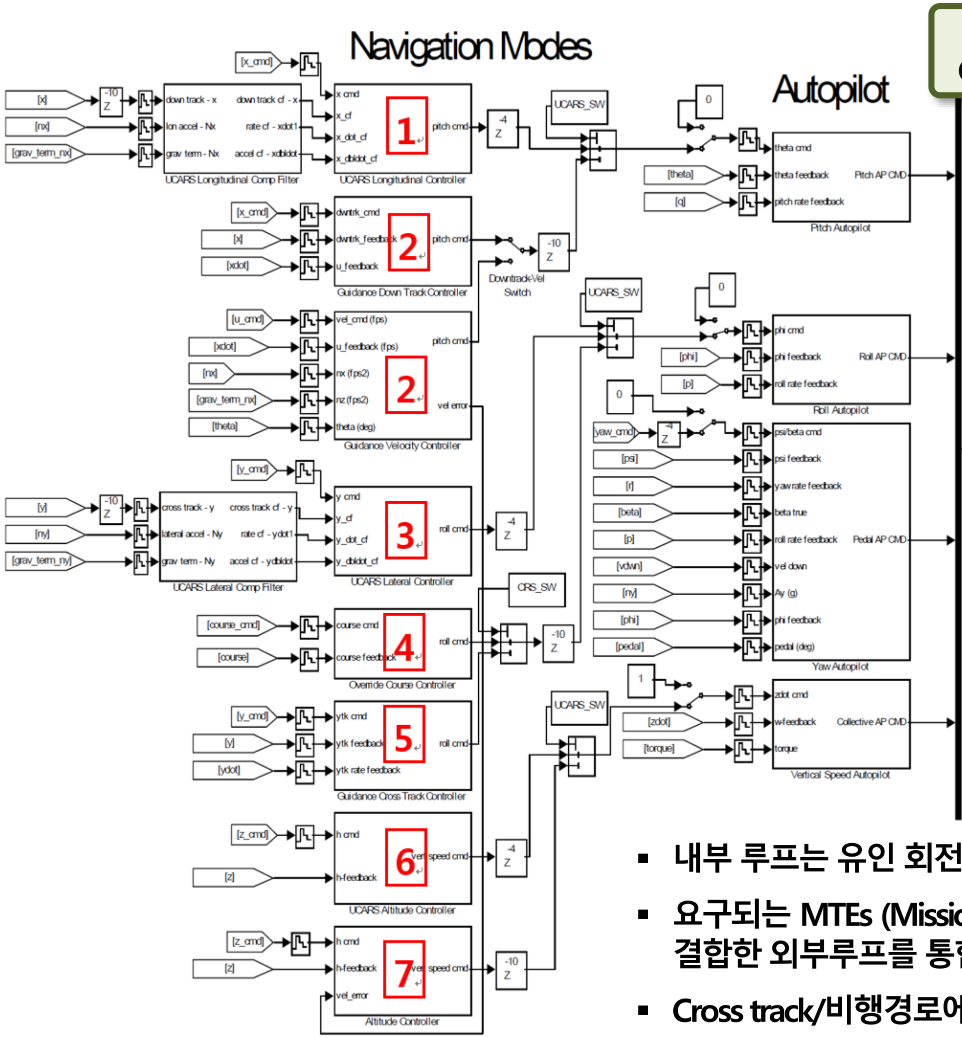
- OFN: Obstacle Field Navigation, 조종사가 지정한 목적지까지 지형/장애물 회피가 가능한 비행경로를 LADAR 를 이용생성 후 AFCS에 제공
- SLAD: Safe Landing Area Determination, 3차원 지형 정보로부터 착륙지 요구조건을 충족하는 착륙지점 결정
- Waypoint Control: 속도, heading 및 glide slope 제어. 경로점 정보 (위치, 속도, 가속도, 시간)로 부터 속도명령 생성
- Tracking control outer loop: 비행경로 추종을 위한 autopilot (AFCS)
- Tracking control inner loop: 비행경로 추종을 위한 조종응답 (command response types) 특성=ACAH, RCDH, heave RCHH)

[Ref 2] Takahashi, Marc D., et al. "Autonomous Rotorcraft Flight Control with Multilevel Pilot Interaction in Hover and Forward Flight."

Initial Motivation for Autonomous FCS Research

Autonomous FCS Structure of RASCAL JUH-60A Black Hawk (US Army)

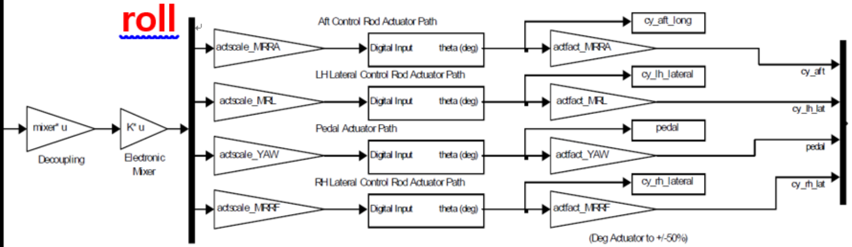
Navigation/ Guidance Modes → Autopilot → ACAH/RCDH VCVH/HCHH



pitch

Mixer algorithm

roll



yaw /turn coordination

heave

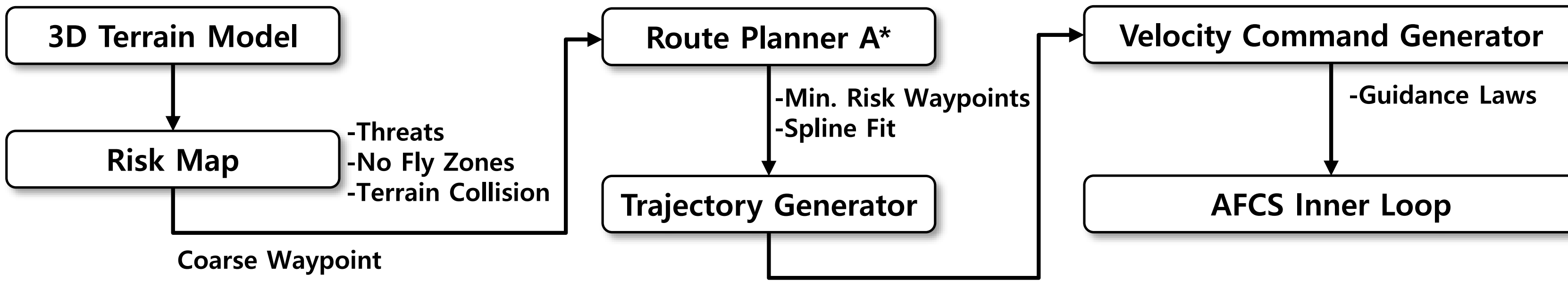
- 내부 루프는 유인 회전익기 조종응답 특성을 구현할 수 있는 구조 채용
- 요구되는 MTEs (Mission Task Elements) 를 구현할 수 있는 다양한 항법모드와 Autopilot을 결합한 외부루프를 통합하여 임무중심 비행제어기 구현
- Cross track/비행경로에 요구되는 상태변수를 임무컴퓨터에서 생성하여 외부루프에 제공

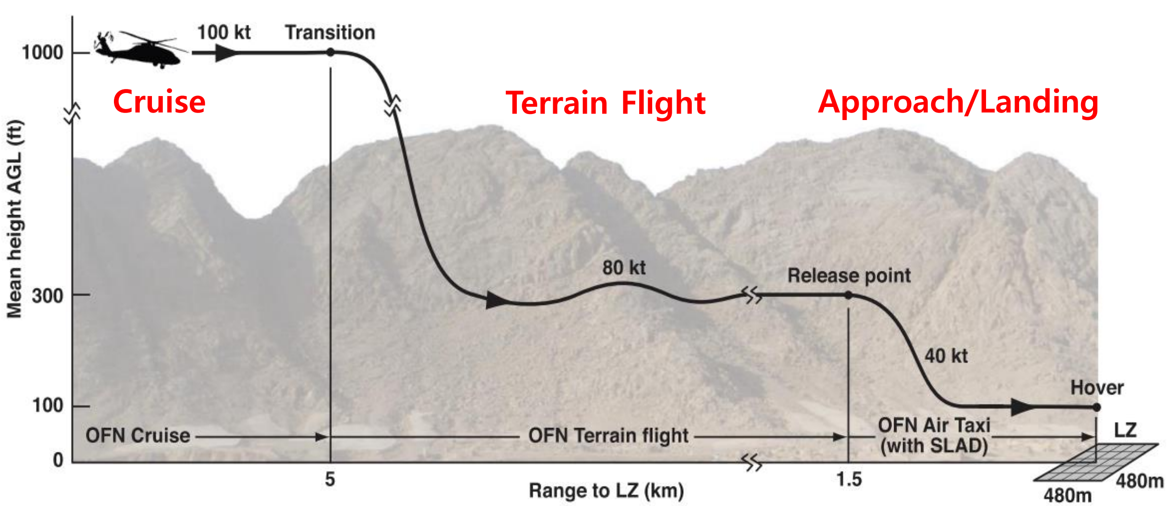
[Ref 2] Takahashi, Marc D., et al. "Autonomous Rotorcraft Flight Control with Multilevel Pilot Interaction in Hover and Forward Flight."

Initial Motivation for Autonomous FCS Research

Autonomous FCS Structure of RASCAL JUH-60A Black Hawk (US Army)

OFN (Obstacle Field Navigation)





Aircraft Parameters

maximum allowed speed (18 m/s)
maximum climb rate (2.5 m/s)
maximum descent rate (2.0 m/s)
maximum normal acceleration (2.0 m/s²)
maximum forward acceleration (0.75 m/s²)
maximum backward acceleration (0.75 m/s²)
maximum turn rate (0.262 r/sec)
width of spline corridor 4 (10 m)
horizontal obstacle clearance limit (40 m)
vertical obstacle clearance limit (30 m)

Path Plan Parameters

maximum time between replans (30 sec)
time between obstacle checks (0.5 sec)
time to update trajectories(constant, 5.0 sec)
time to update trajectories(linear, 0.5 sec)

Ref 2: Autonomous Rotorcraft Flight Control with Multilevel Pilot Interaction in Hover and Forward Flight
Ref 3. Autonomous Black Hawk in Flight: Obstacle Field Navigation and Landing-site Selection on the RASCAL JUH-60A

Initial Motivation for Autonomous FCS Research

Autonomous FCS Structure of RASCAL JUH-60A Black Hawk (US Army)

Autonomous Flight Modes

Cruise Mode

110 kts ground speed
300 m AGL
Using Waypoint Command

Automatic engagement
@ 1.5~5 km from landing zone

Terrain Flight Mode

40 kts ground speed
60 m AGL
Using Velocity Command

Automatic engagement
@ 700 m from landing zone

Approach Mode

25 kts ground speed
30 m AGL
Using Velocity Command

Autonomous FCS Structure

Waypoint Trajectory — t, r, V, Accel

Velocity Vector Command — Time sequence of V vector

Waypoint Control

Continuous trajectory position command

$$\boldsymbol{\eta}^T = (x, y, h, \psi, t)$$

Tracking Control
- Hover mode
- FWD flight mode

Aircraft reference state command

$$\boldsymbol{\xi}^T = (u, v, \dot{h}, r, \dot{\psi}, \gamma, \beta, t)$$

Inner Loop Control

Generate Control Inputs for Aircraft

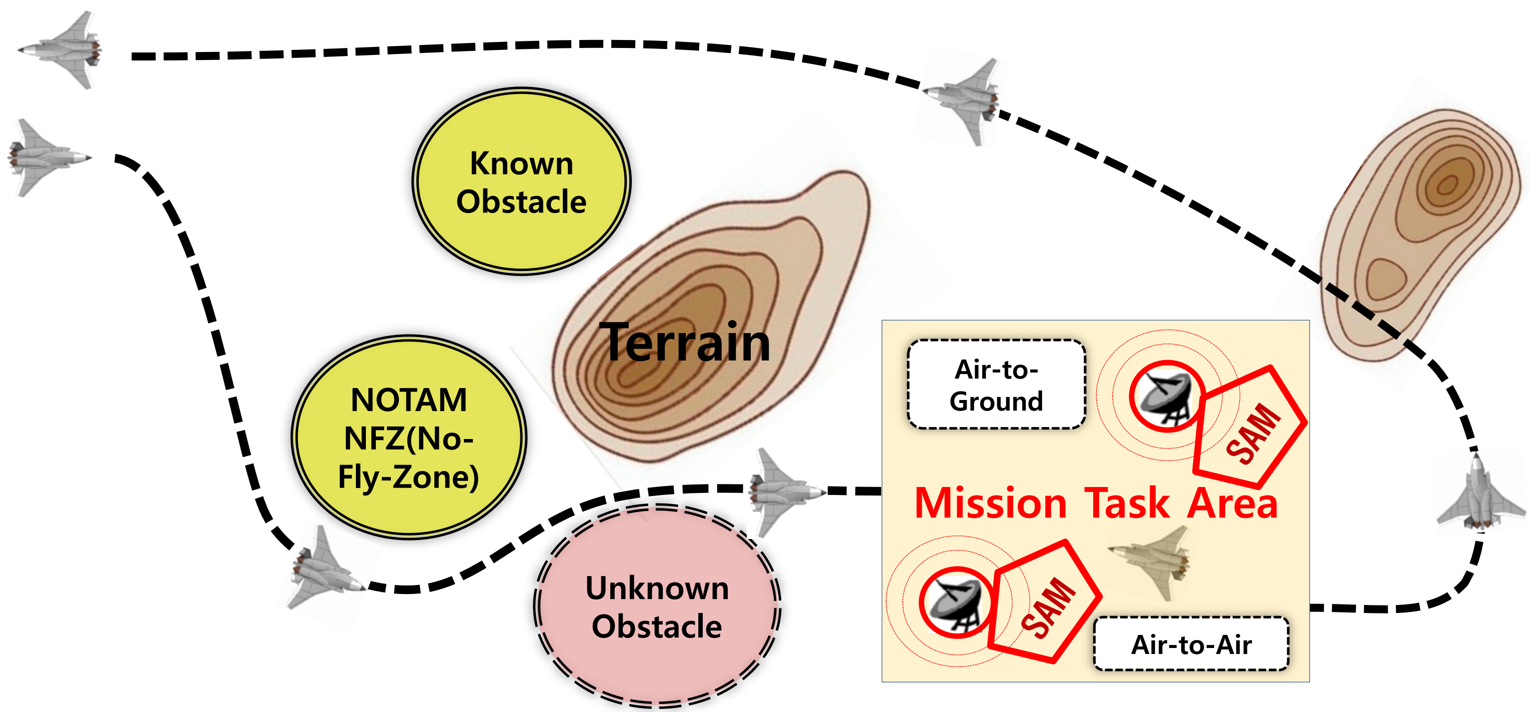
$$\boldsymbol{\delta}^T = (\delta_0, \delta_{1C}, \delta_{1S}, \delta_{TR})$$

Vehicle

Ref 2: Autonomous Rotorcraft Flight Control with Multilevel Pilot Interaction in Hover and Forward Flight
Ref 3. Autonomous Black Hawk in Flight: Obstacle Field Navigation and Landing-site Selection on the RASCAL JUH-60A
Ref 4: Development and Flight Testing of a Flight Control Law for Autonomous Operations Research on the RASCAL JUH-60A

Initial Motivation for Autonomous FCS Research

Mission Scenario Analysis for Functional Requirements : UCAV Mission



Digital Terrain / Path Planning (Waypoint Guidance Mode)

Trajectory Generator
- Shortest/Safe Path (Waypoint-based)
- Terrains / Threats /NFZ

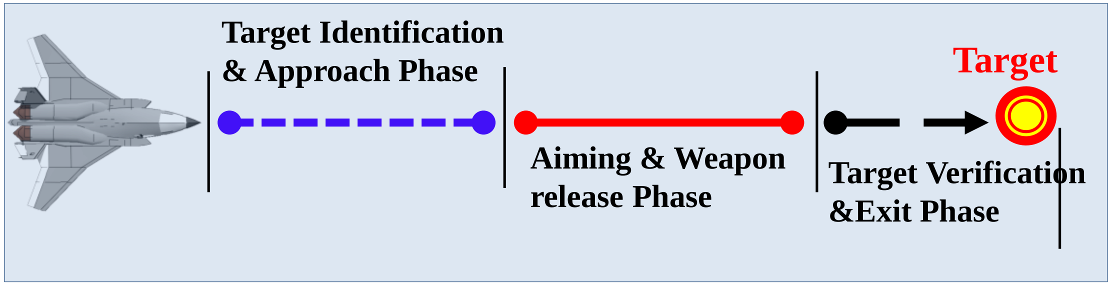
Waypoint Guidance (Path Tracking Laws)

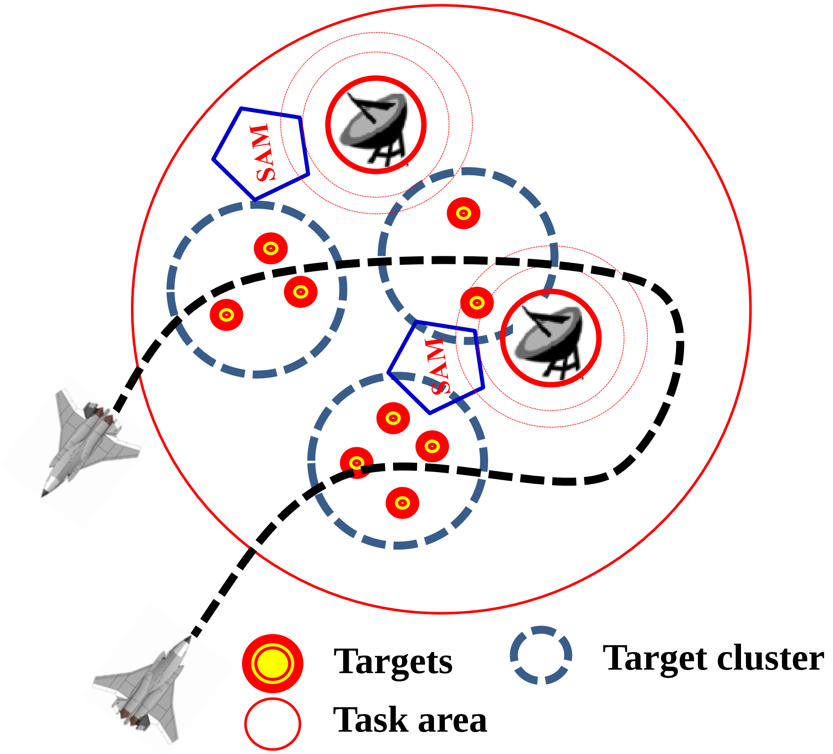
Maneuver-Trajectory Generator
- Air-to-Ground mission
- Air-to-Air mission

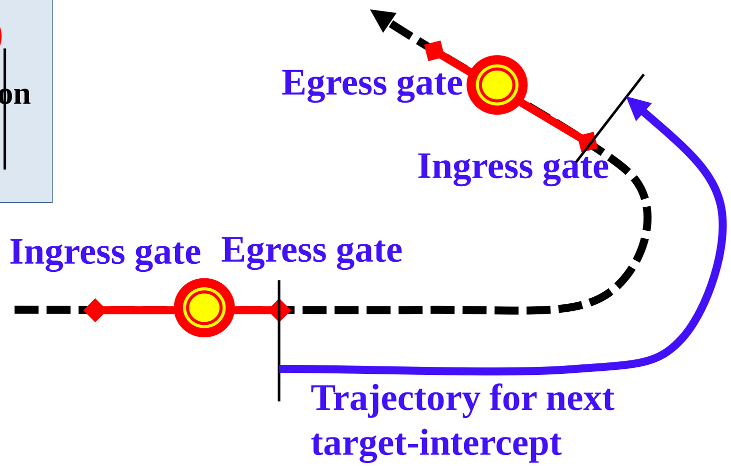
Maneuver-Trajectory Tracking Guidance

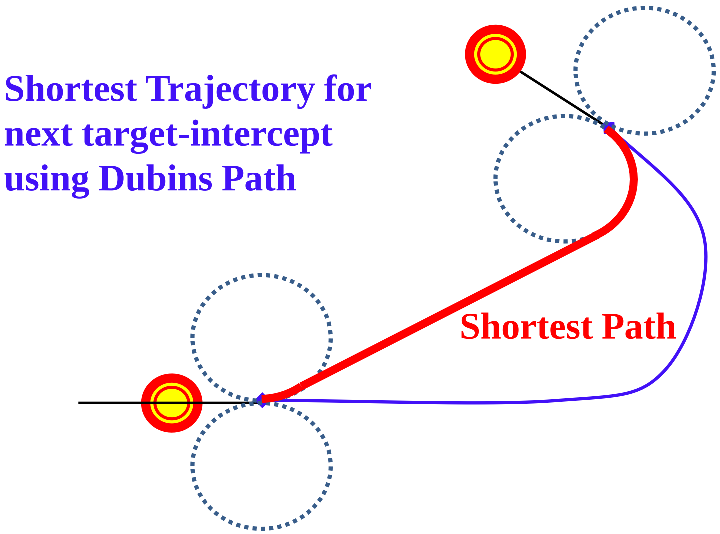
Initial Motivation for Autonomous FCS Research

Mission Scenario Analysis for Functional Requirements : Air-to-Ground





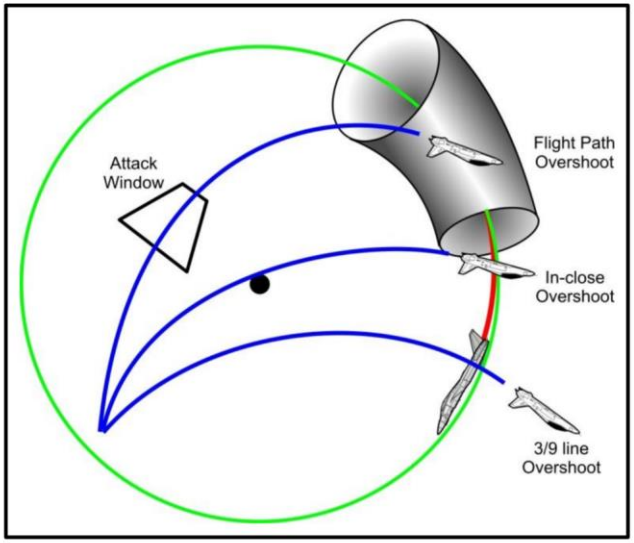


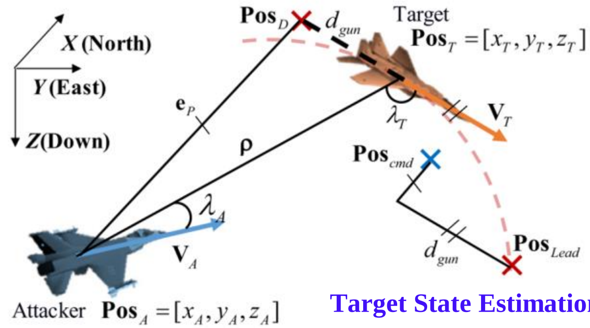


Initial Motivation for Autonomous FCS Research

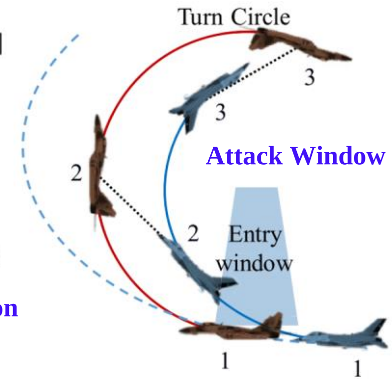
Mission Scenario Analysis for Functional Requirements : Air-to-Air

Offensive (Pursuit) Maneuvers

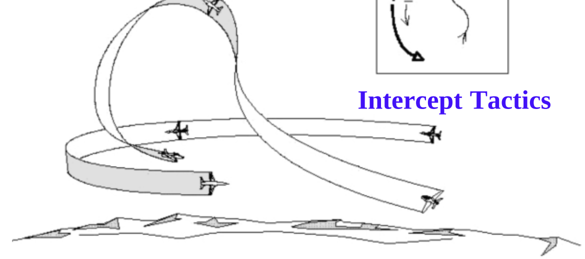


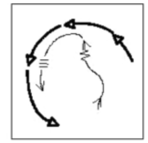


Target State Estimation

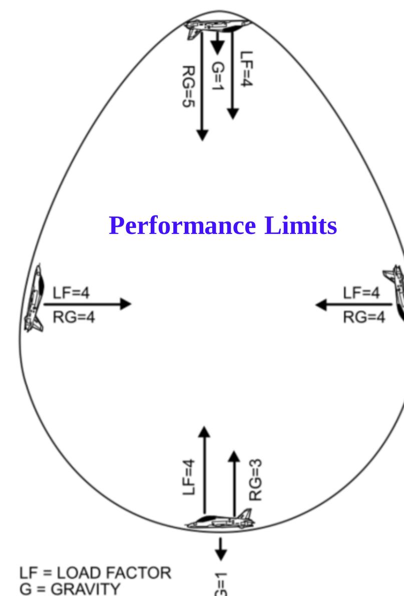


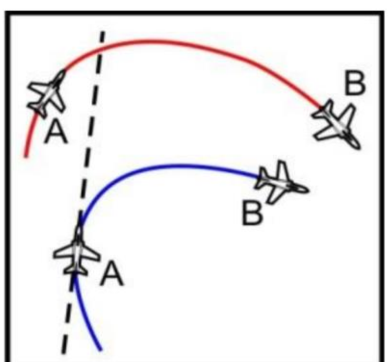
Barrel Roll



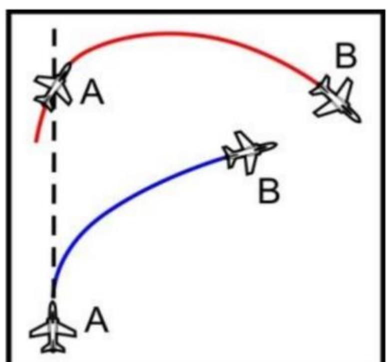


Intercept Tactics

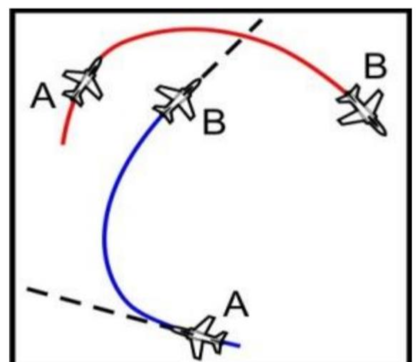




Lead Pursuit



Pure Pursuit

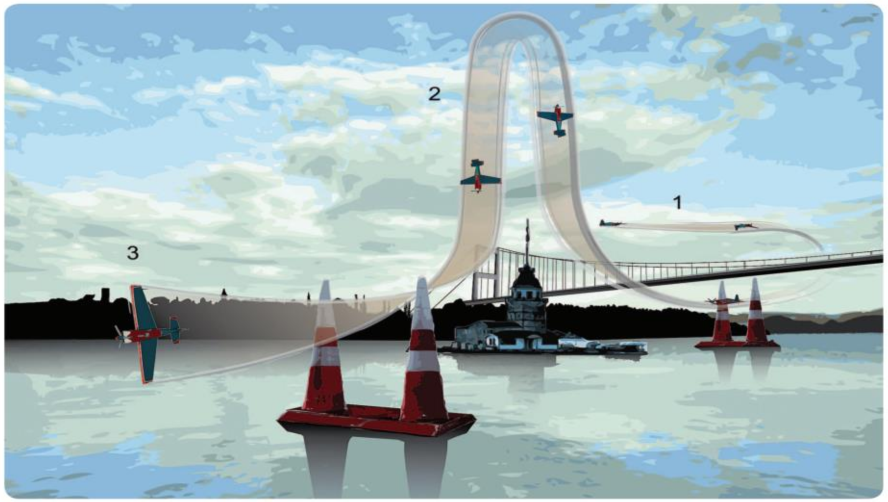


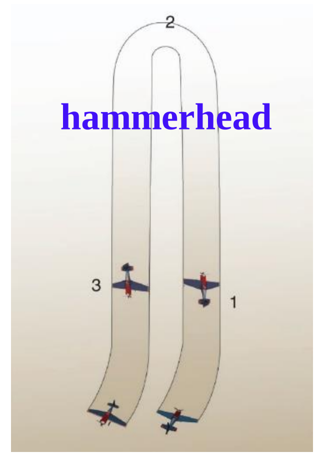
Lag Pursuit

Initial Motivation for Autonomous FCS Research

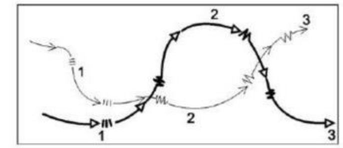
Mission Scenario Analysis for Functional Requirements : Air-to-Air

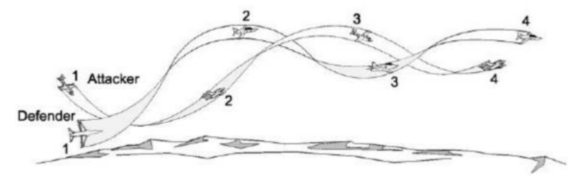
Defensive (Evasive) Maneuvers



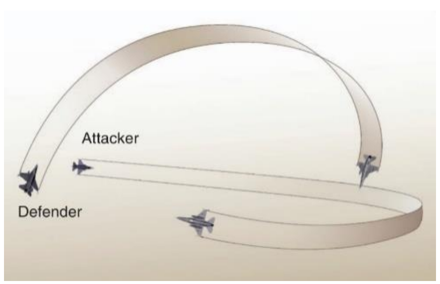


hammerhead

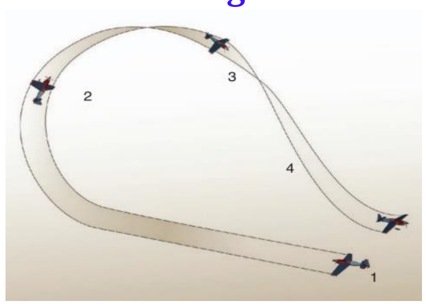


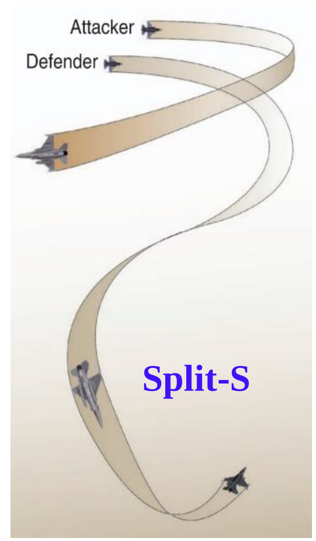


High yo-yo

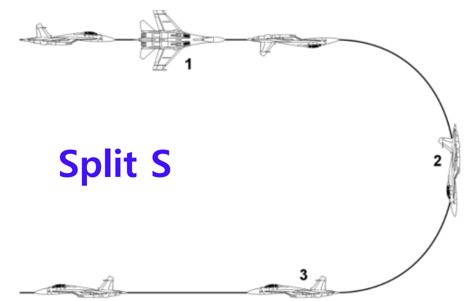


half Cuban eight

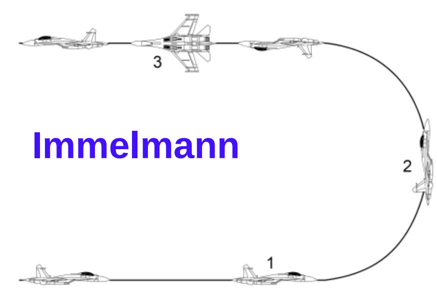




Split-S



Split S



Immelmann

Initial Motivation for Autonomous FCS Research

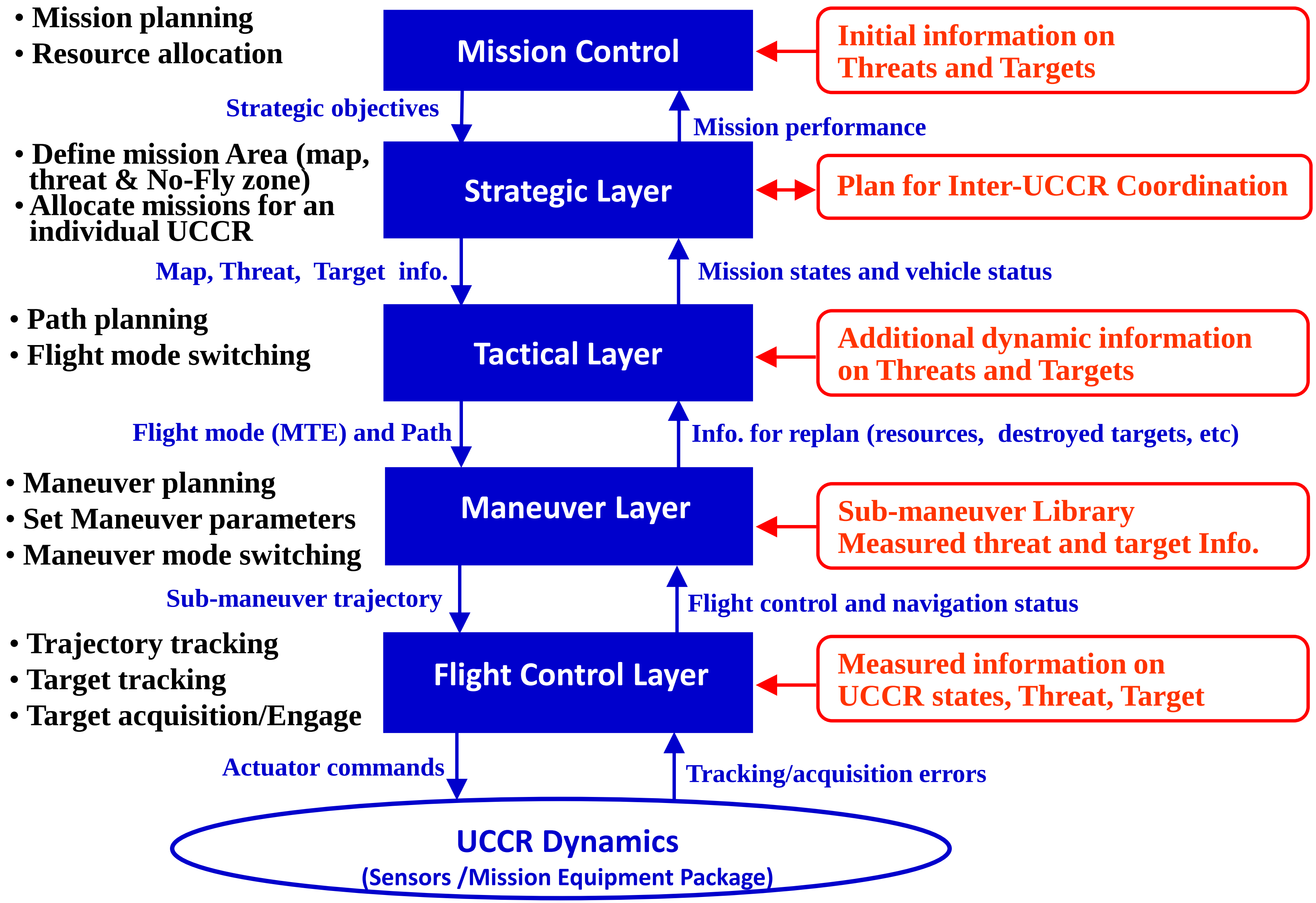
Mission Scenario Analysis for Functional Requirements : Air-to-Ground

Mission Phases	Threats / Obstacles	Terrain Masking	Trajectory	Path Constraints	Aircraft Modes
(1) Take off / Acceleration			Base	TO procedure	RW→FW
(2) Climb			Waypoint	V, RoC	FW
(3) Approach to target zone	Radar / SAM / Terrain / NFZ		Waypoint	V, RoC	FW
(4) Enter into threat aera	Radar / SAM / Terrain / NFZ		Waypoint	V, nz, RoC	FW
(5) target priority selection	Radar / SAM / Terrain / NFZ		Waypoint	V, nz, RoC	FW
(6) Ingress to target zone	Radar / SAM / Terrain / NFZ		Waypoint	V, nz, RoC	FW
(7) Maneuvers for target intercept (multi-target intercept)	Radar / SAM / Terrain / NFZ		Aggressive MTEs	Corridor for Best intercept V, nz, RoC	FW
(8) Egress from target zone	Radar / SAM / Terrain / NFZ		Waypoint	V, nz, RoC	FW
(9) Escape from threat aera	Radar / SAM / Terrain / NFZ		Waypoint	V, nz, RoC	FW
(10) Repeat (3)~(9) as required	Radar / SAM / Terrain / NFZ		Waypoint	V, nz, RoC	FW
Return to base			Waypoint	V, RoC	FW
Deceleration / Landing approach			Waypoint	V, RoD LD procedure	FW→RW
Landing			Base	LD procedure	RW

RW = Rotary Wing Mode
FW = Fixed Wing Mode

KU 建國大學校 KONKUK UNIV. Initial Motivation for Autonomous FCS Research

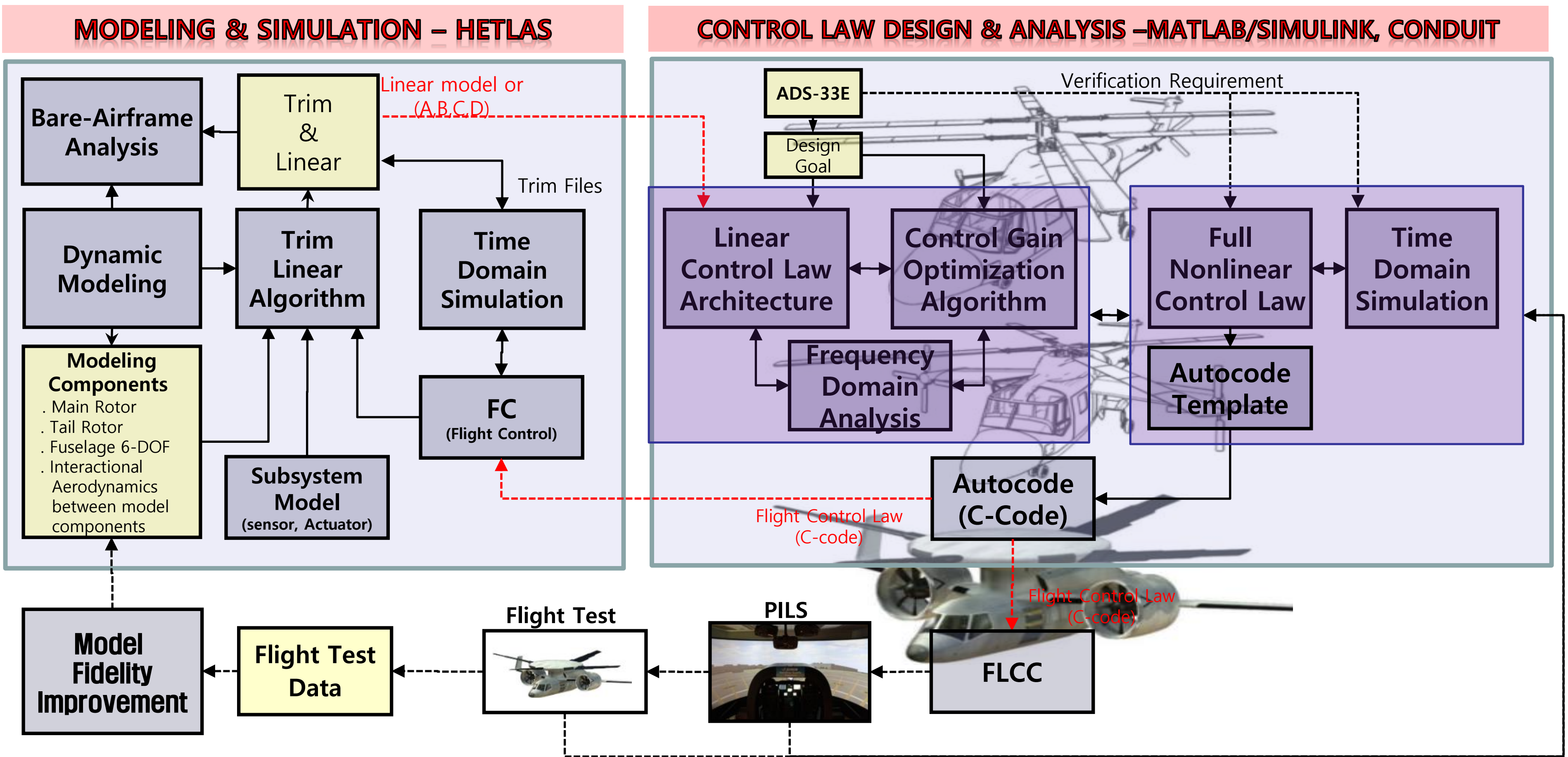
High-Level Structure and Function Requirements of Autonomous FCS



Initial Motivation for Autonomous FCS Research

Research Environments for Autonomous AFCS : What KKU Has

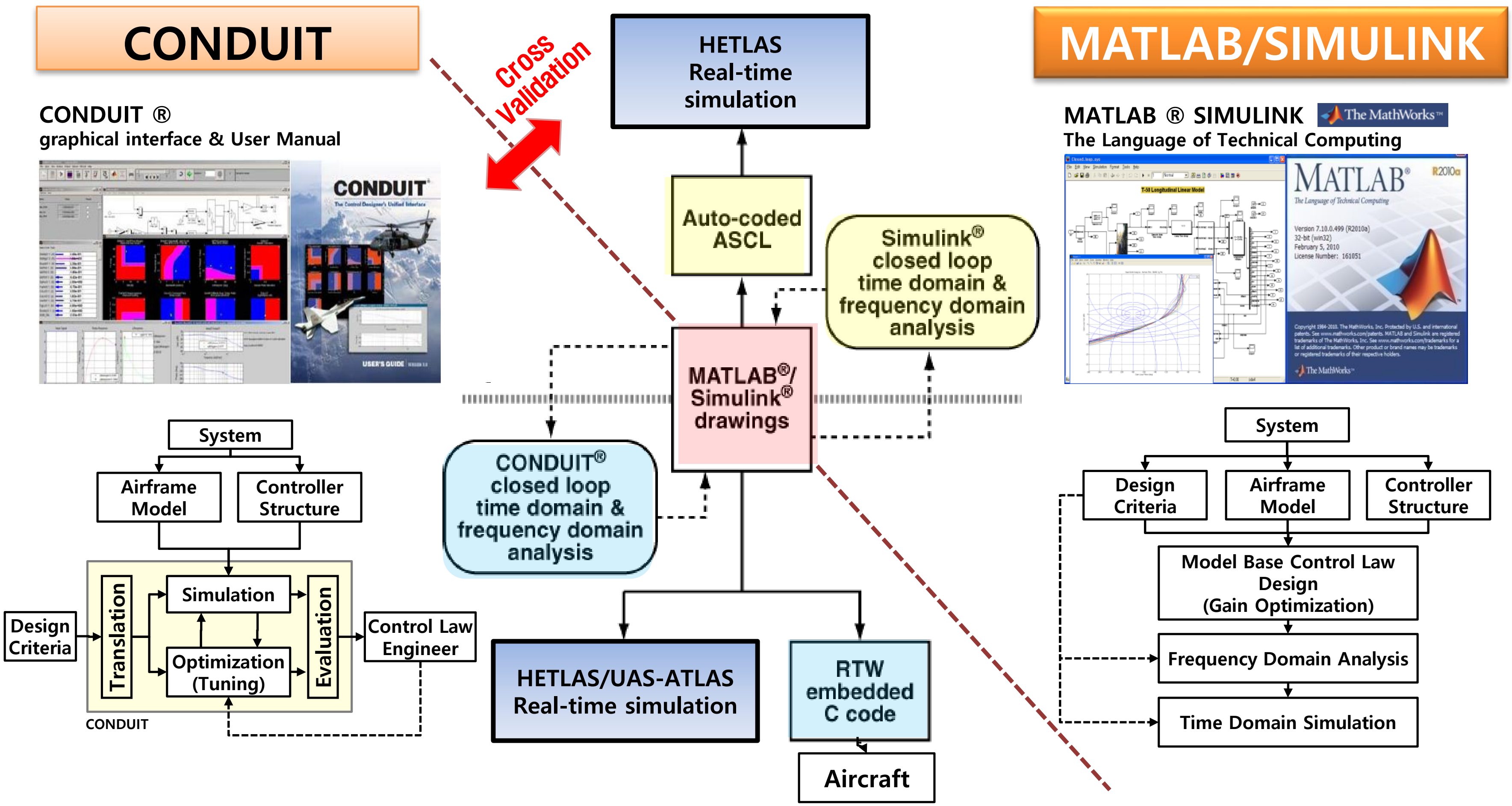
Integrated Development Environment for Advanced Flight Control System



Initial Motivation for Autonomous FCS Research

Research Environments for Autonomous AFCS : What KKU Has

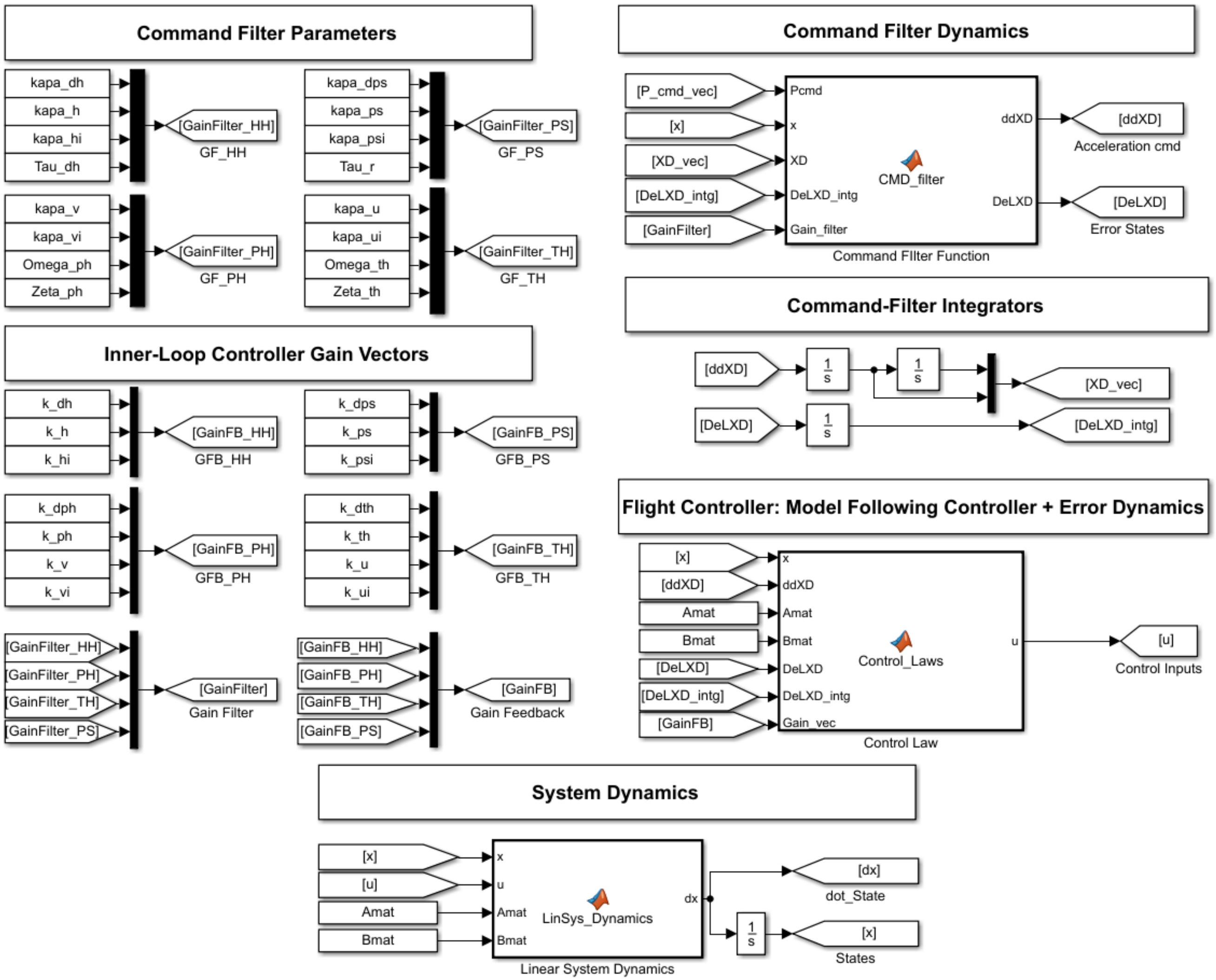
Integrated Optimal Design of Model-Following Flight Control Laws



Initial Motivation for Autonomous FCS Research

Research Environments for Autonomous AFCS : What KKU Has

Integrated Simulink Template for Design and Evaluation of Flight Control Law



Initial Motivation for Autonomous FCS Research

KKU Selection of Major Research Areas for Autonomous FCS

UAS

- Mechanics and Electronics
- Regulation and Permits
- Vehicle Autonomy
- Mission Planning
- Ground Station
- Communication and Telemetry

Vehicle Autonomy

- Task Allocation
- Obstacle Detection
- Path Planning
- Collision Avoidance
- Trajectory Generation
- Vehicle Control

Path Planning

- Cell Decomposition
- Potential Field
- Optimization

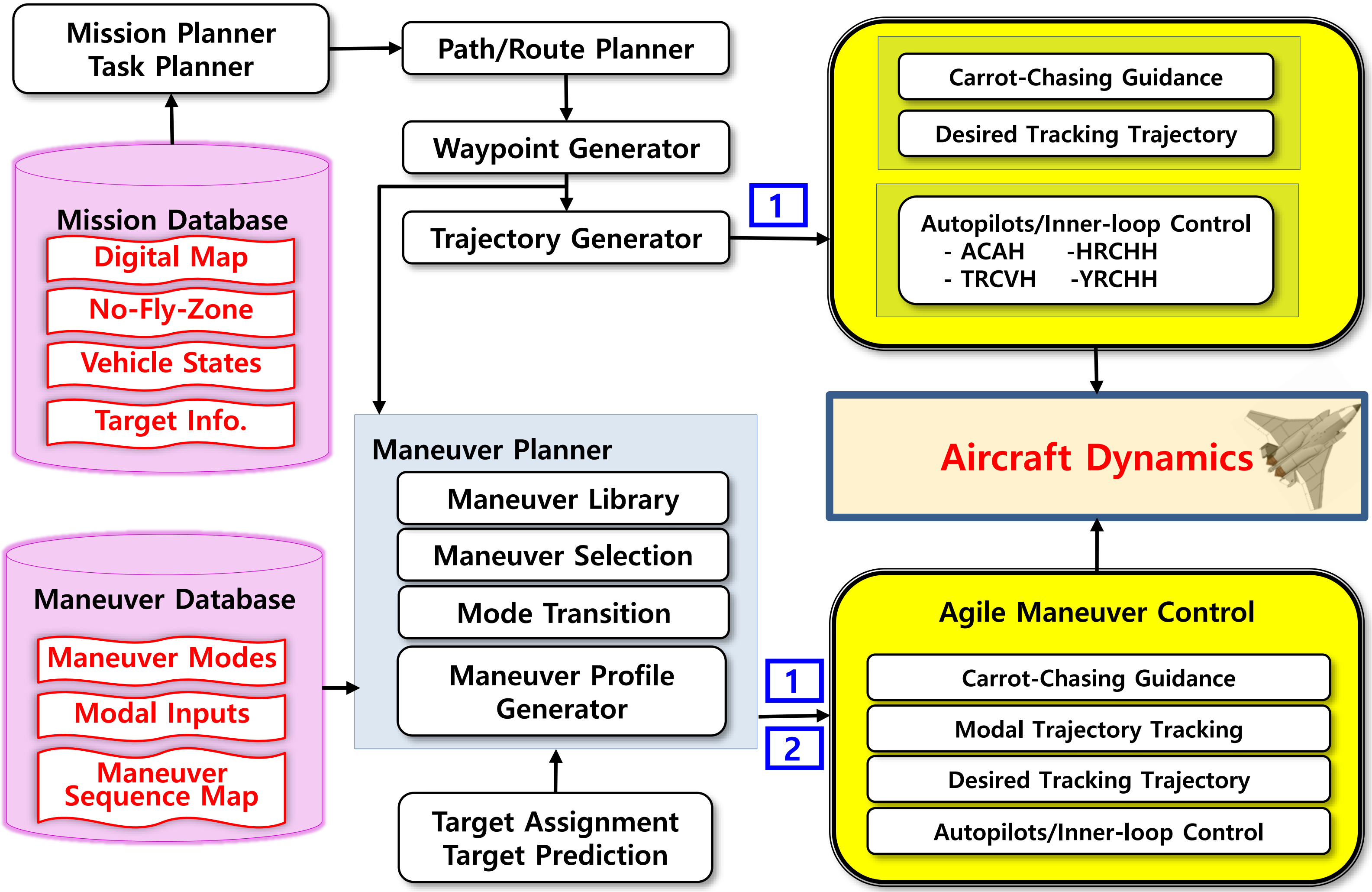
Road Map
- Visibility Graph
- Voronoi Diagram
- Probabilistic Roadmap(PRM)
- Rapidly Exploring Random tree (RRT)

Heuristic Search
- Trulla
- Dijkstra Algorithm
- A* Algorithm
- D* Lite

Selected Major Areas for KKU Research

Initial Motivation for Autonomous FCS Research

Initial Flight-Control-System Structure for Autonomous FCS



1 : Waypoint Guidance Mode

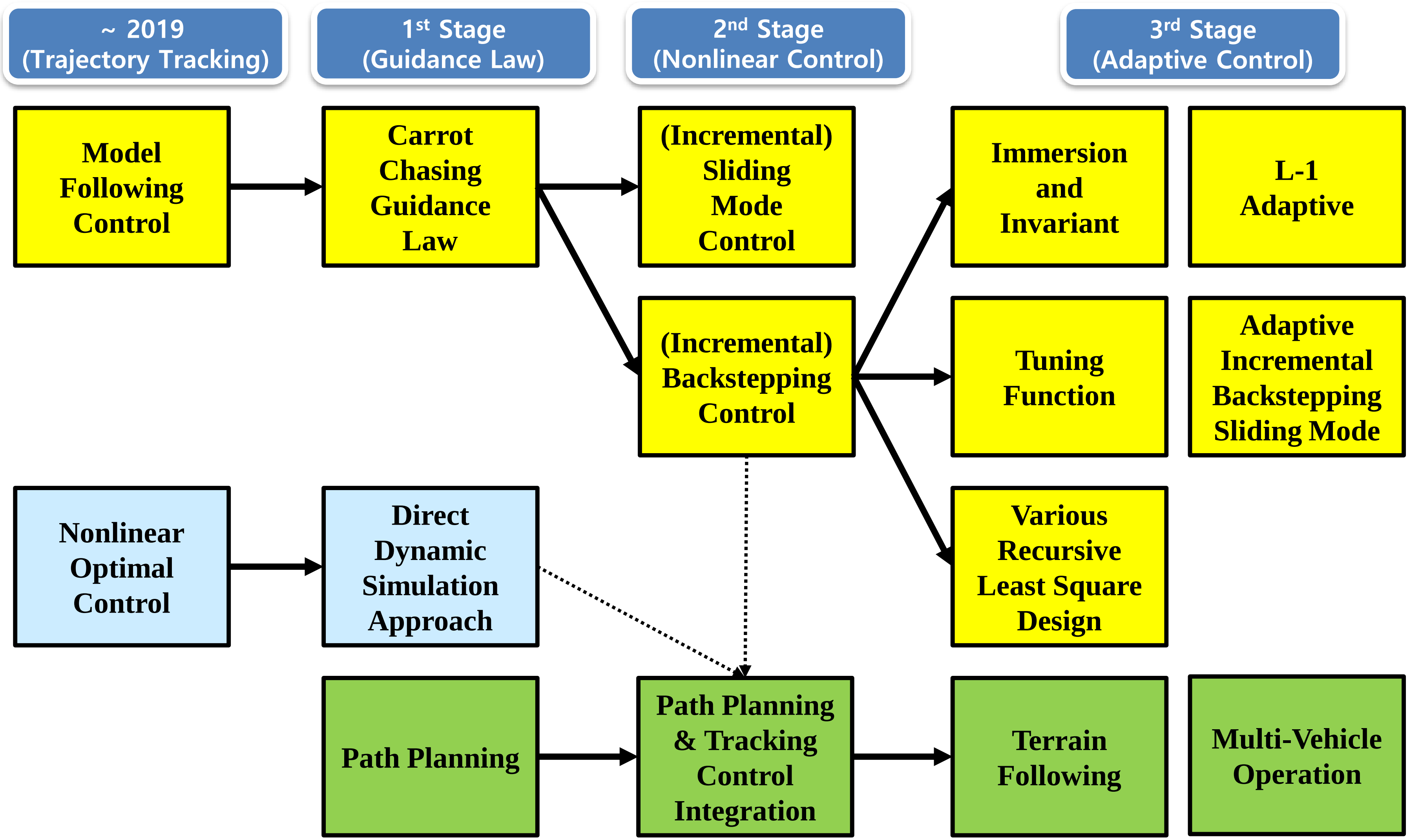
2 : Agile Tactical Maneuver Mode

KU 建國大學校 KONKUK UNIV.

Contents

Activities at Initial Stage of Autonomous FCS Research

Overall History of KKU Research Activities for Autonomous FCS

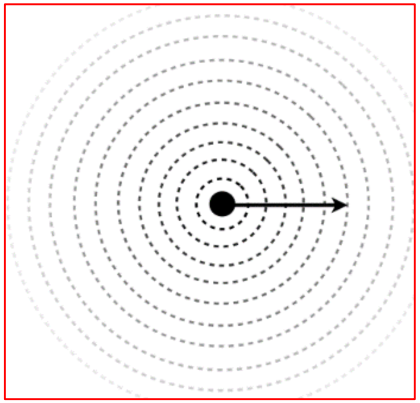


Digital Terrain Model using Radial Basis Functions

Generation of Digital Terrain with Randomly Distributed RBF

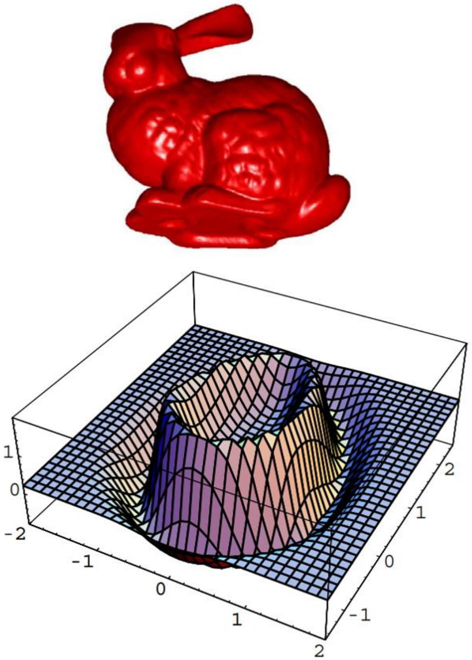
Radial Basis Function $\phi = \phi(r) \leftarrow r = \|\mathbf{r} - \mathbf{r}_0\|, \quad \mathbf{r} \in R^n, \mathbf{r}_0 \in R^n$

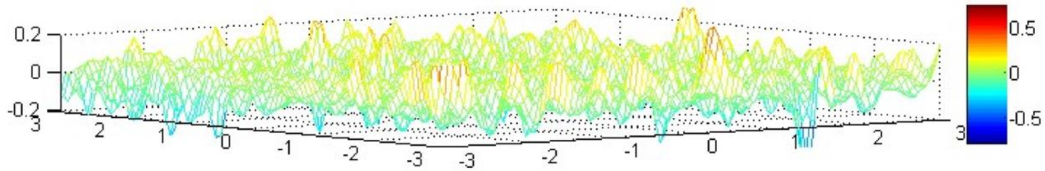
- Global RBS
- Compactly Supported RBF

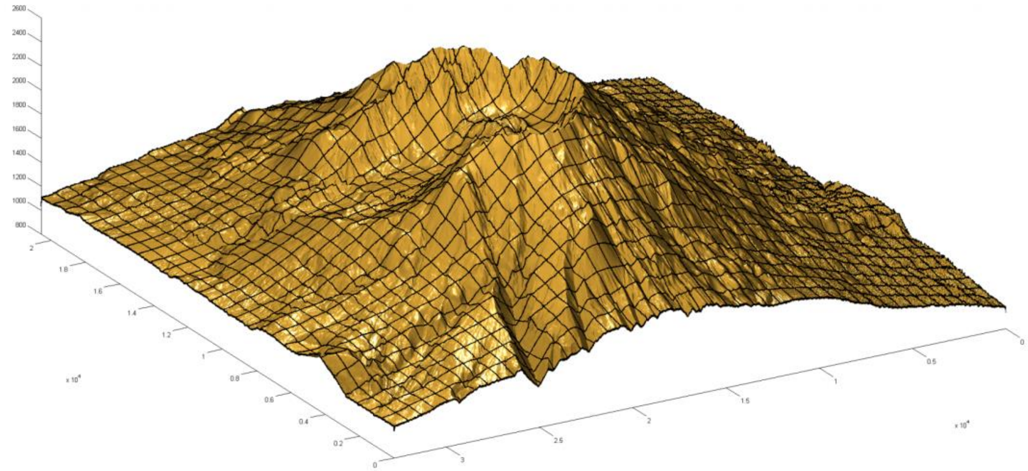


Curve Fitting Using RBF $f(\mathbf{r}) = \sum_{j=1}^{j=m} w_j \phi\left(\|\mathbf{r} - \mathbf{r}_j\|\right)$

Curve Fitting Examples



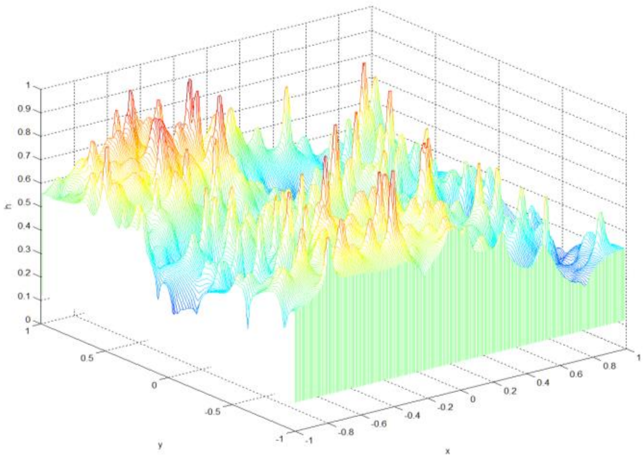




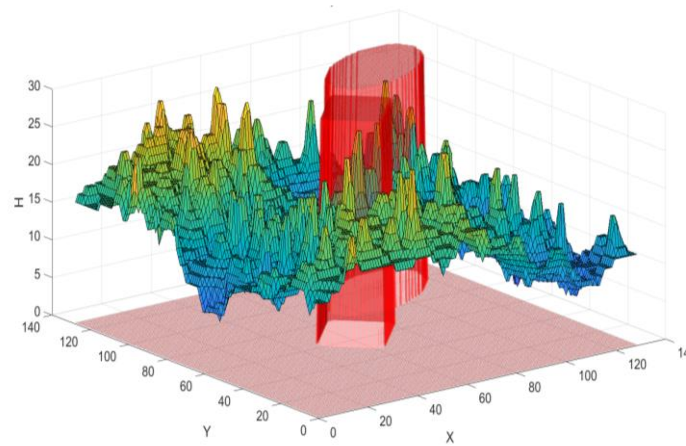
Activities at Initial Stage of Autonomous FCS Research

Digital Terrain Model using Radial Basis Functions

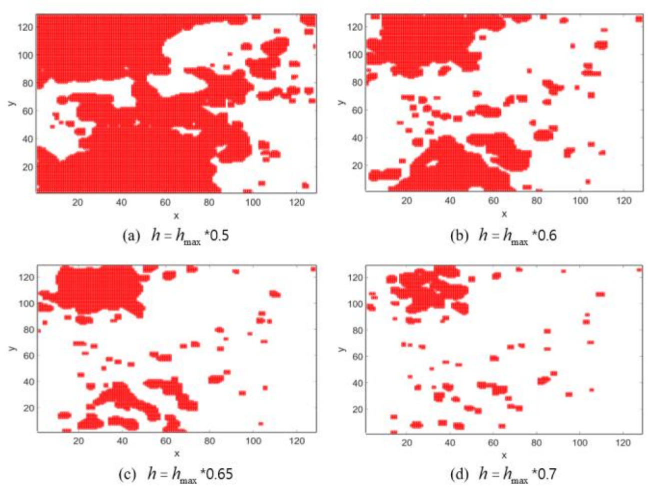
3-D Digital Terrain Model with No-Fly-Zone



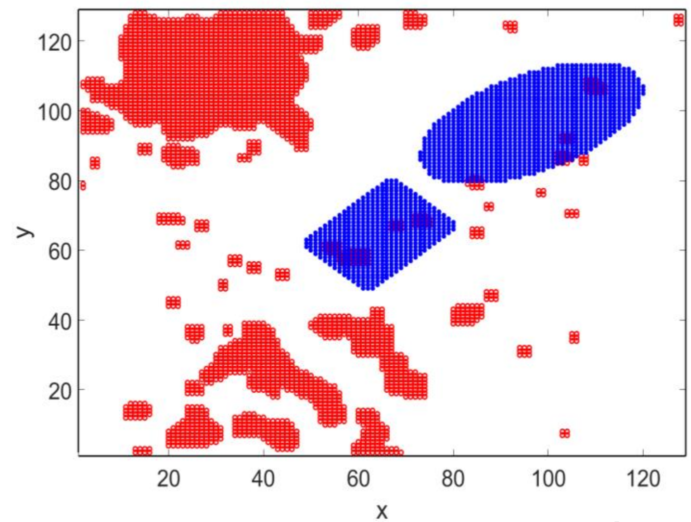
Generated Terrain



No-Fly-Zone Insertion on 3D Terrain map



2D-Plain map at given height



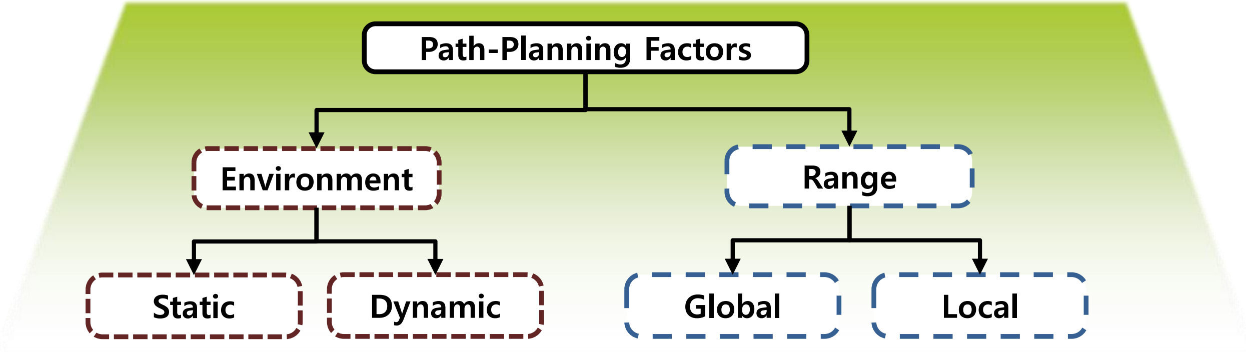
No-Fly-Zone Insertion on 2D-Plain at h=0.65*Hmax

Development of Path Planning Algorithm

Definition of Path Planning: **Find the path between the initial and final ponits without collision with terrain and obstacles with due consideration for path cost.**

Factors Affecting to Path Planning Algorithm: Environment and Planning Range

- **Static Environment** : **Time invariant. Used mainly for pre-flight path planning problems**
- **Dynamic Environment** : **Used mainly for real-time path planning with time-varying moving obstacles**
- **Global Planning** : **Path planning with the complete knowledge about entire environments**
  Used mainly for pre-flight optimal path planning problems
- **Local Planning** : **Path planning without the complete knowledge about entire environments**
  Used mainly for obstacle detection and real-time path replanning



Ref : A. Koubaa et al., "Introduction to mobile robot path planning," Stud. Comput. Intell., vol. 772, no. April, pp. 3–12, 2018.

Activities at Initial Stage of Autonomous FCS Research

Development of Path Planning Algorithm

Available 3-D Path Planning Algorithms

- 3-D Path Planning Algorithms
  - Sampling Based Algorithms
  - Math Model Based Algorithms
  - Node Based Algorithms
  - Bio-Inspired Algorithms
  - Multi-Fusion Based Algorithms

Algorithms	Time complexity	Applicable environment	Real time applicability
Sampling based algorithms	$O(\text{nlogn}) \leq T \leq O(n^2)$	Static and Dynamic(Part)	On-line
Node based algorithms	$O(n\text{logn}) \leq T \leq O(n^2)$	Static and Dynamic(Part)	On-line
Mathematic model-based algorithms	Depending on the polynomial equation	Static and Dynamic	Off-line
Bioinspired algorithms	$O(n^2) \leq T$	Static and Dynamic(Part)	Off-line
Multifusion based algorithms	$O(\text{nlogn}) \leq T$	Depending on the algorithms	On-line

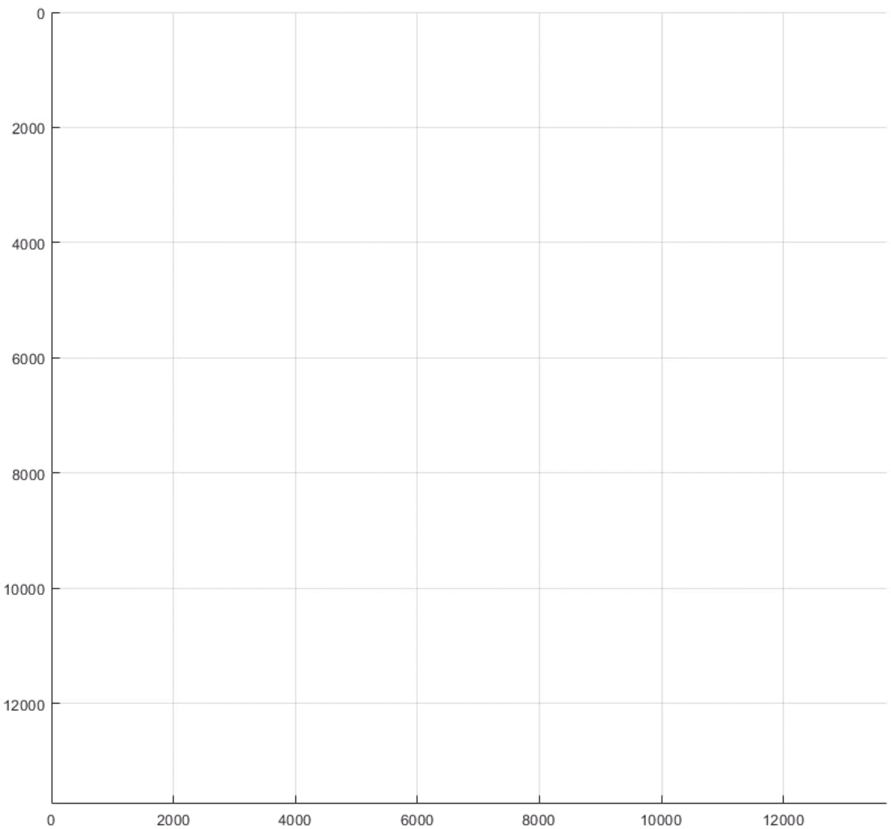
Sampling based algorithms best suit for real-time applications with less limitations

Ref : L. Yang, J. Qi, D. Song, J. Xiao, J. Han, and Y. Xia, "Survey of Robot 3D Path Planning Algorithms," Journal of Control Science and Engineering.

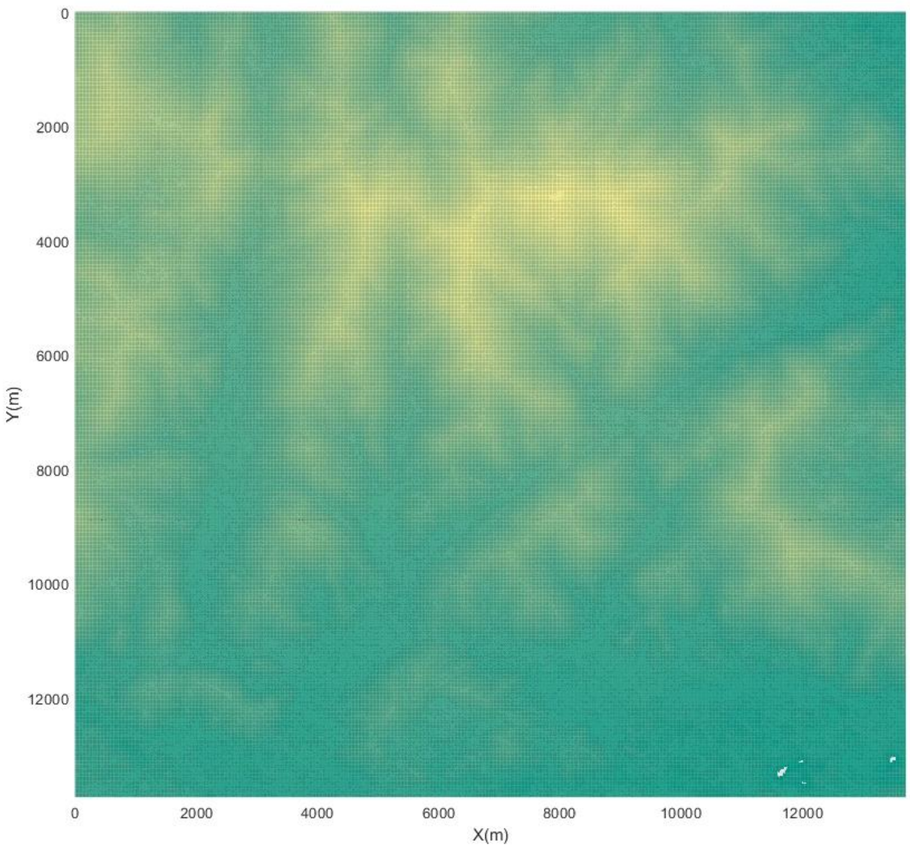
Activities at Initial Stage of Autonomous FCS Research

Development of Path Planning Algorithm

3-D Path Planning using RRT (Rapidly-exploring Random Tree) Algorithms



- Seoul (37°25'20.2" N, 127°01'21.9" E.)



Activities at Initial Stage of Autonomous FCS Research

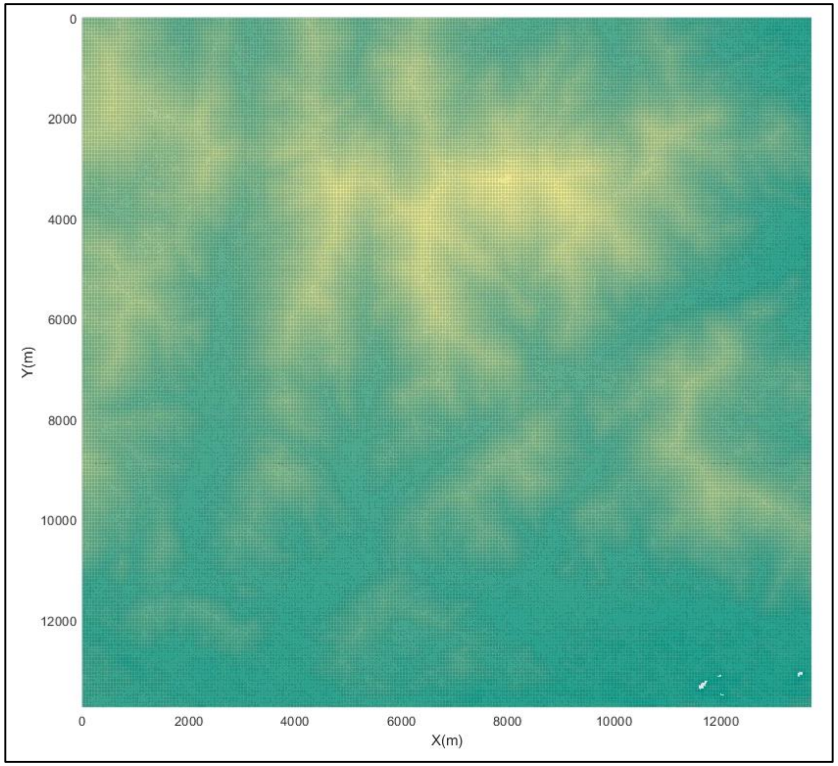
Development of Path Planning Algorithm

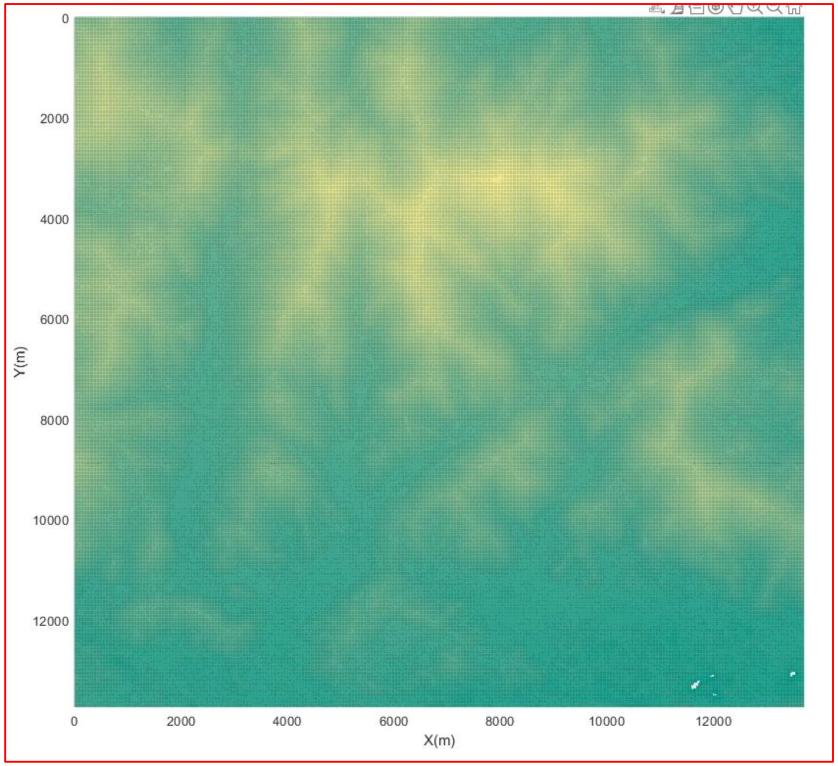
3-D Path Planning using RRT (Rapidly-exploring Random Tree) Algorithms

RRT vs RRT*

500 nodes sampling
3D Digital Terrain using Real data
$\rho_{near} = 700\text{m}$

: initial node (2162 , 12080, 300)
: goal node (11220, 1071, 300)
: Tree : Path





▲ RRT

- New Node connected with the Nearest Node
- Path is not changed after the initial path generated

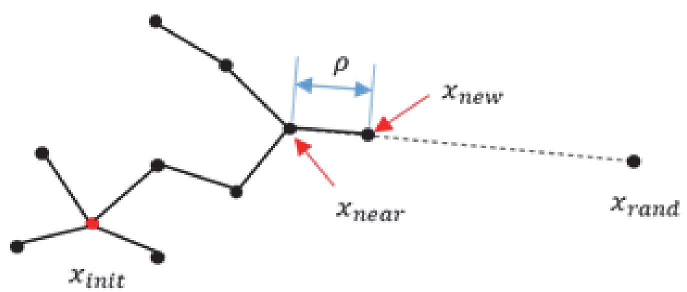
▲ RRT*

- New Node connected with the Best Node
- Tree connection changed as Node added.

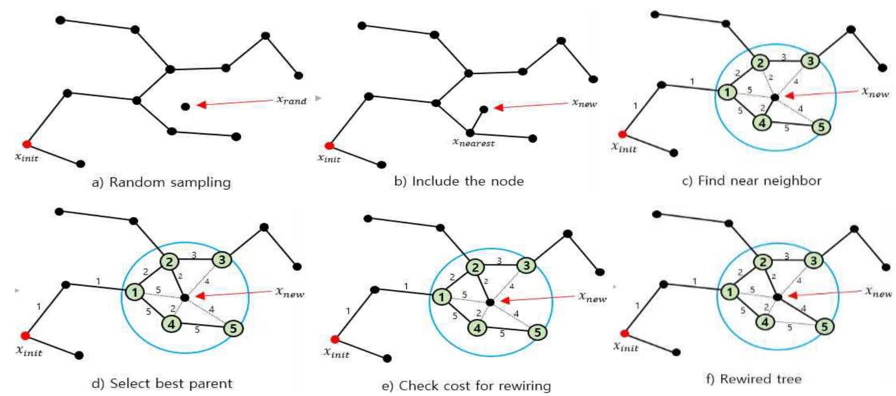
Development of Path Planning Algorithm

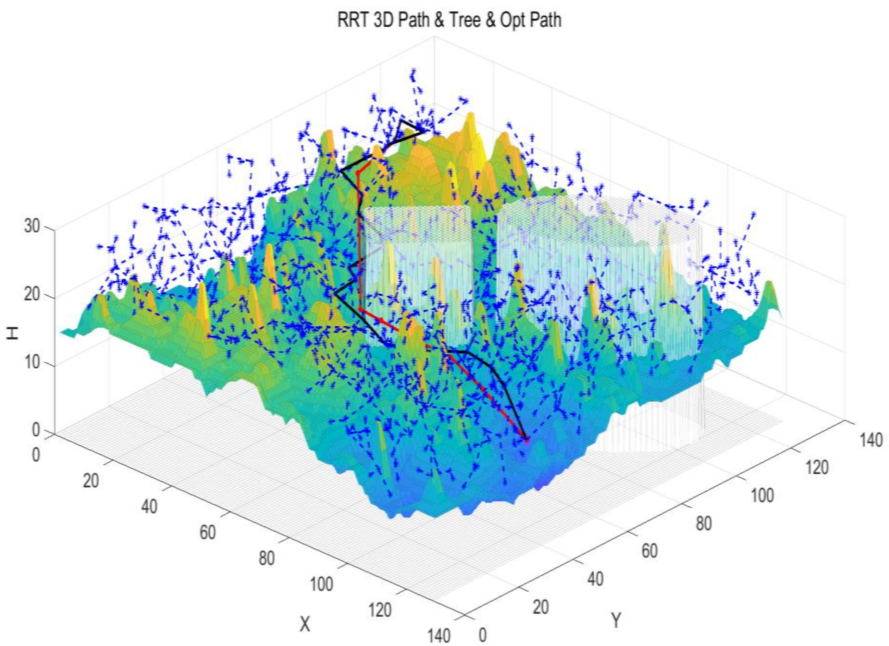
3-D Path Planning using RRT (Rapidly-exploring Random Tree) Algorithms

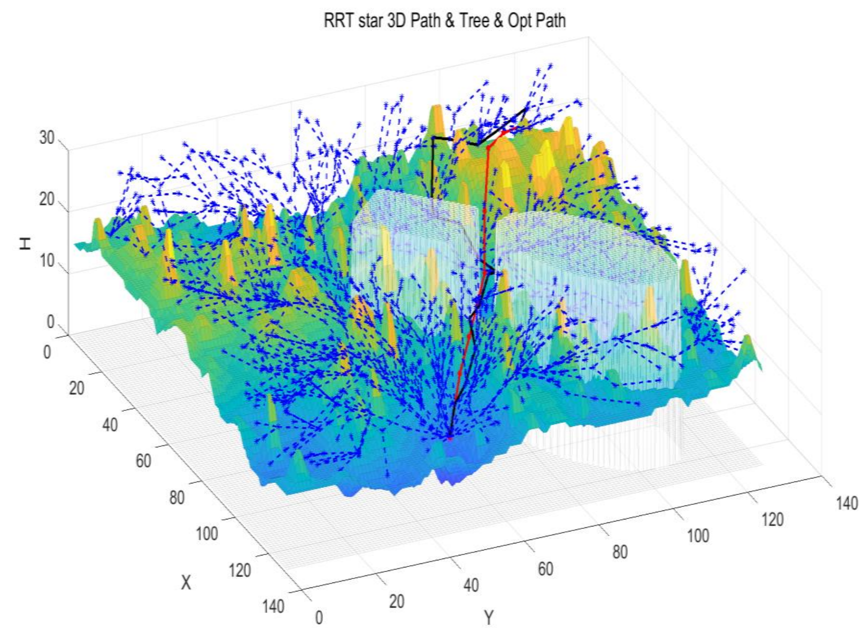
RRT



RRT*



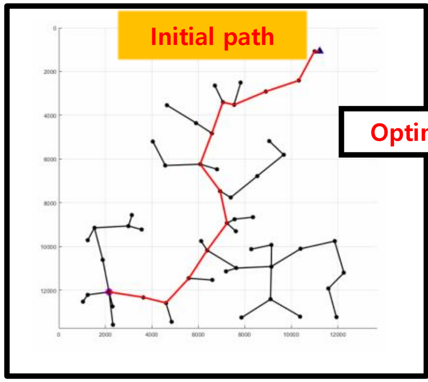




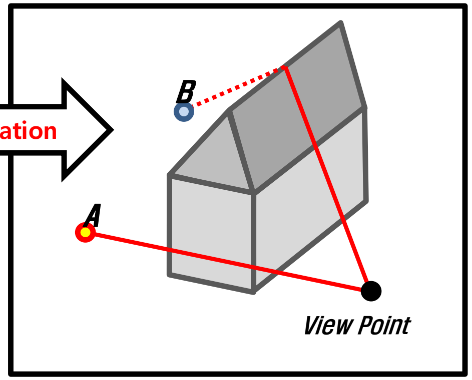
Activities at Initial Stage of Autonomous FCS Research

KU 建國大學校 KONKUK UNIV.

Development of LOS (Line-Of-Sight) Path Optimization Algorithm

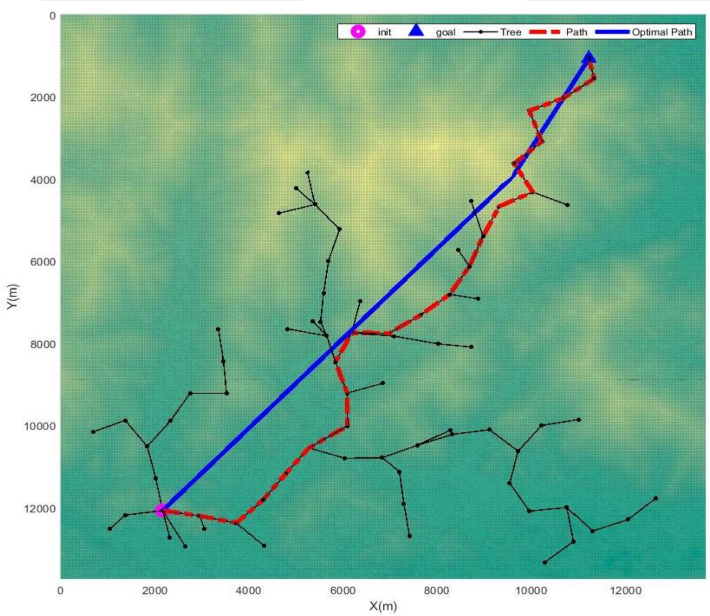


Optimization



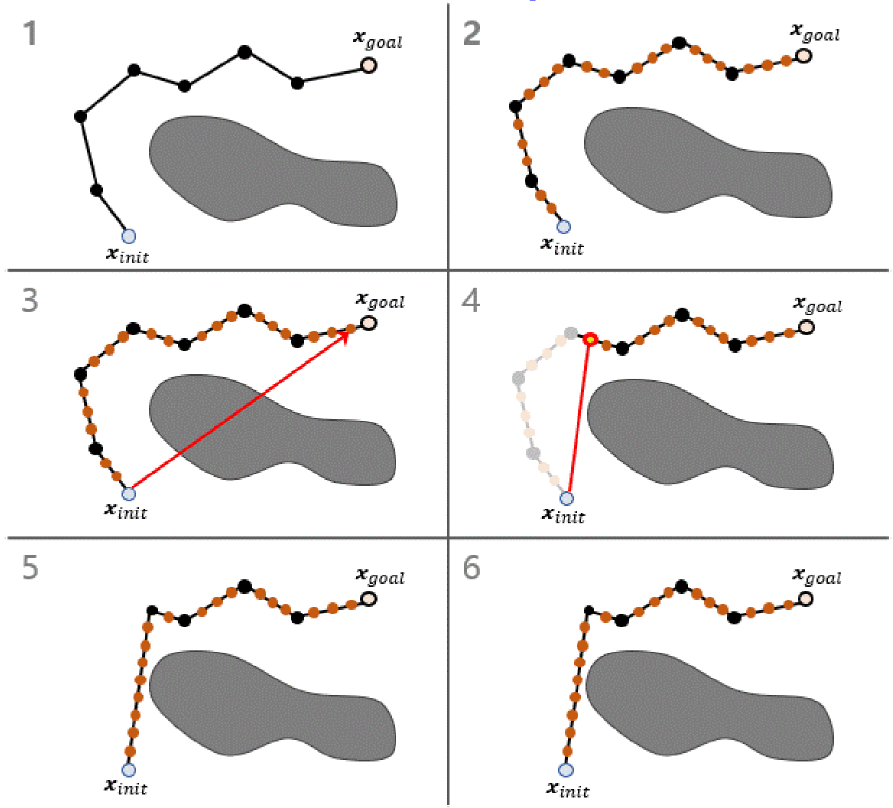
RRT*

Line-of-Sight



RRT*+LOSPO : RRT* + (Line-of-sight)

Process for LOS Path Optimization



1. Path Planning with RRT*
2. Way-point Insertion for Smooth Interpolation
3. Detect the best LOS node without collision
4. Define new path

5~6. Repeat up to the goal point to get the optimized path

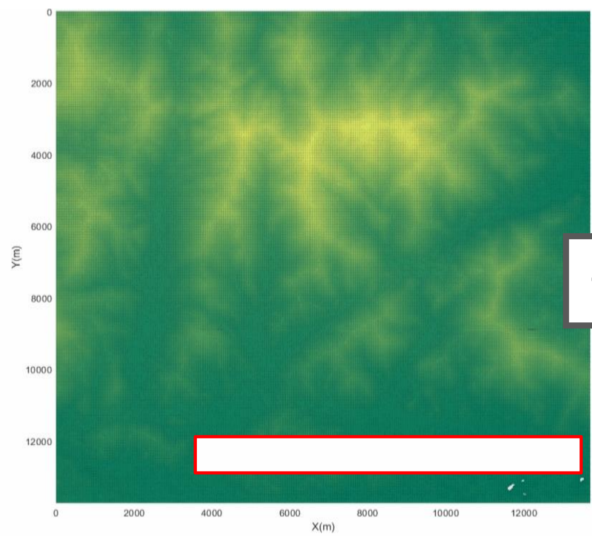
Activities at Initial Stage of Autonomous FCS Research

Development of LOS (Line-Of-Sight) Path Optimization Algorithm

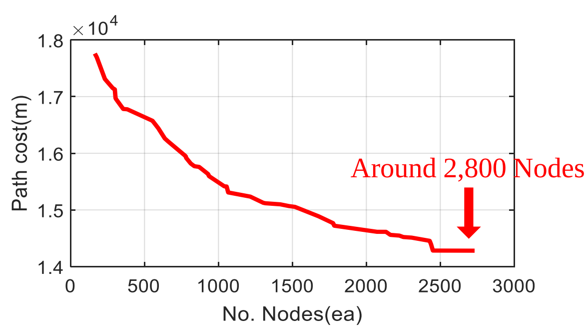
RRT* algorithm

Path Optimized when new node added

- **Large Data Storage Memory Required**
- **Slow convergence speed**



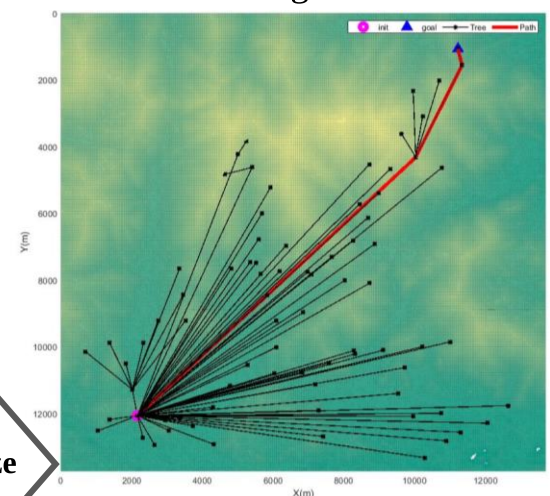
▲ **Path Optimization (RRT*)**

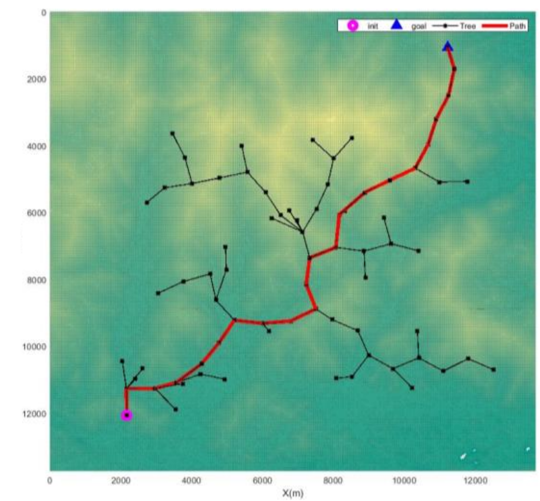


▲ Path Cost as a function of Node Number

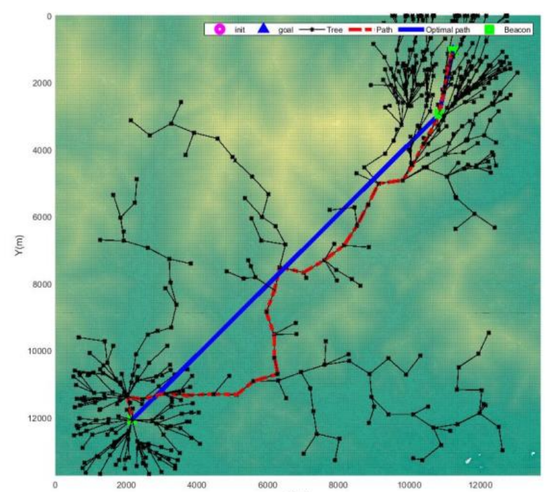
Optimize

RRT*-based algorithm

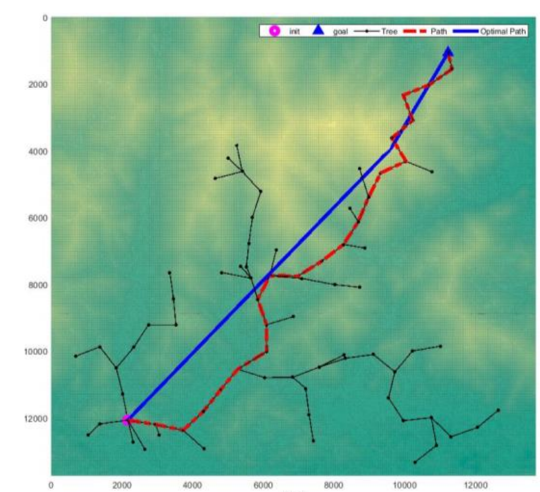




Q-RRT* : RRT* + (Best Parent selection)



RRT*-Smart : RRT* + (smart method)



RRT*+LOSPO : RRT* + (Line-of-sight)

Activities at Initial Stage of Autonomous FCS Research

Flyable Trajectory Generation using Spline Curves

Conditions for Flyable Trajectory and Its Generator

- **A flyable trajectory must pass all prescribed way points**
- **A flyable trajectory must meet the continuity conditions for position, velocity, acceleration, and even jerk vectors at each waypoint.**
- **A flyable trajectory generator must provide the useful information to check the aircraft fly-ability along the generated trajectory.**

Spline Trajectory Generator

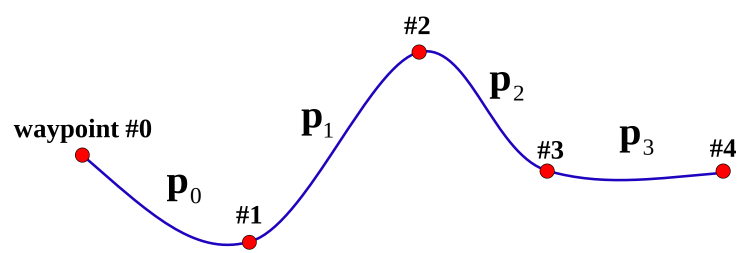
- **Waypoint data** $\left\{t_k, \mathbf{p}_k^w = \left(x_k^w, y_k^w, h_k^w, \psi_k^w\right)\right\}_{k=0}^{k=K}$
- **Spline trajectory** $\boxed{\mathbf{p}_k(\tau) = \mathbf{a}_{0k} + \mathbf{a}_{1k}\tau + \mathbf{a}_{2k}\tau^2 + \mathbf{a}_{3k}\tau^3 + \mathbf{a}_{4k}\tau^4 + \mathbf{a}_{5k}\tau^5 + \mathbf{a}_{6k}\tau^6 + \mathbf{a}_{7k}\tau^7 = \sum_{j=0}^{j=7} \mathbf{a}_{jk}\tau^j}$

$$\tau = \frac{t - t_k}{\Delta t_k} \in [0,1]$$

$$t \in [t_k, t_{k+1}]$$

$$\Delta t_k = t_{k+1} - \Delta t_k$$

$$\frac{d}{dt} = \frac{d\tau}{dt}\frac{d}{d\tau} = \frac{1}{\Delta t_k}\frac{d}{d\tau}$$



Activities at Initial Stage of Autonomous FCS Research

Flyable Trajectory Generation using Spline Curves

Spline Trajectory Generator

- Time Integration Formula

$$\boxed{\mathbf{q}(t)=\mathbf{q}_k+\int_{t_k}^{t}\mathbf{p}_k\{\tau(t)\}\,dt=\mathbf{q}_k+\Delta t_k\int_0^{\tau}\mathbf{p}_k(\tau)d\tau=\mathbf{q}_k+\Delta t_k\sum_{j=0}^{j=7}\frac{\mathbf{a}_{jk}}{j+1}\tau^{j+1}}$$

$$\dot{\mathbf{q}}(t)=\mathbf{p}_k\{\tau(t)\}$$
$$\mathbf{q}(t_k)=\mathbf{q}_k$$

- Time Derivative Formula

$$\boxed{\dot{\mathbf{p}}_k=\frac{d\mathbf{p}_k(\tau)}{dt}=\frac{1}{\Delta t_k}\mathbf{p}_k'(\tau)=\frac{1}{\Delta t_k}\sum_{j=0}^{j=7}j\mathbf{a}_{jk}\tau^{j-1}=\frac{1}{\Delta t_k}\left(\mathbf{a}_{1k}+\mathbf{a}_{2k}\tau+\mathbf{a}_{3k}\tau^2+\cdots\right)=\begin{pmatrix}\dot{\mathbf{r}}\\ \dot{\psi}\end{pmatrix}^T}$$

Information for Fly-ability Check

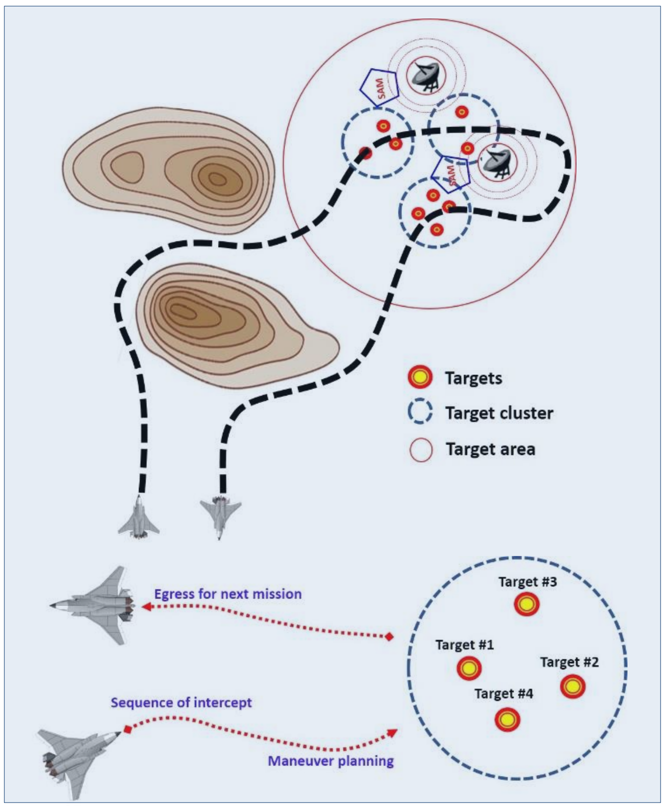
- speed $\qquad v(t)=\|\dot{\mathbf{r}}(t)\|=\sqrt{\dot{x}^2+\dot{y}^2+\dot{h}^2}$
- Load factor $\qquad n(t)=\dfrac{1}{g}\left(\dfrac{v^2}{\rho}\right)$
- Turn radius $\qquad \rho=\dfrac{\{(\dot{x})^2+(\dot{y})^2\}^{1.5}}{|\dot{x}\ddot{y}-\ddot{x}\dot{y}|}$
- Rate of climb $\qquad v_C=\dot{h}$
- Flight path angle $\qquad \gamma_C=\tan^{-1}\left(\dot{h}/\sqrt{\dot{x}^2+\dot{y}^2}\right)$

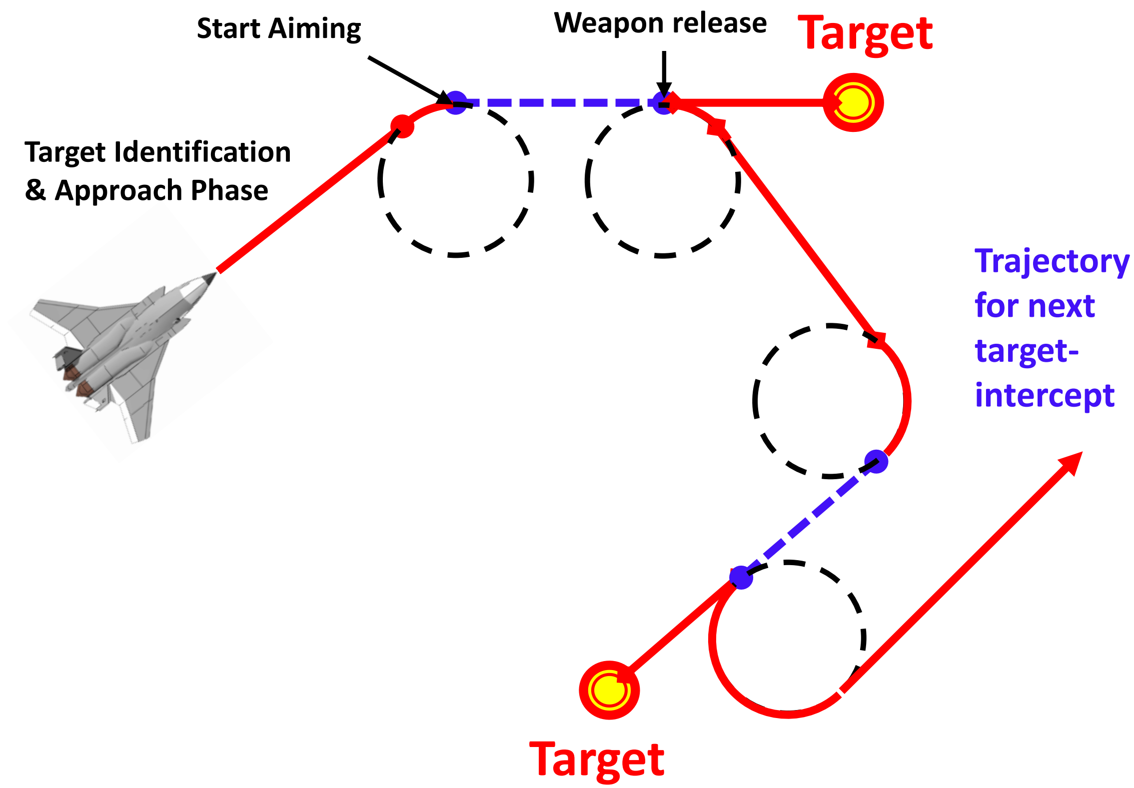
Activities at Initial Stage of Autonomous FCS Research

Flyable Trajectory Generation using Dubins Path

Conceptual Use of Dubins Path

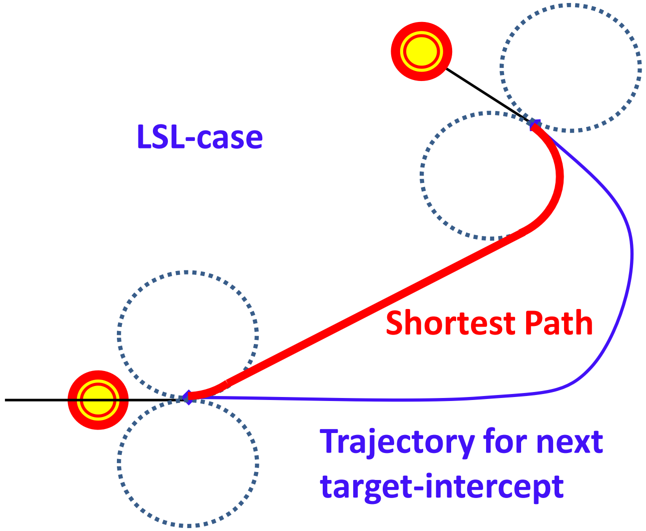
Interception of Clustered Targets

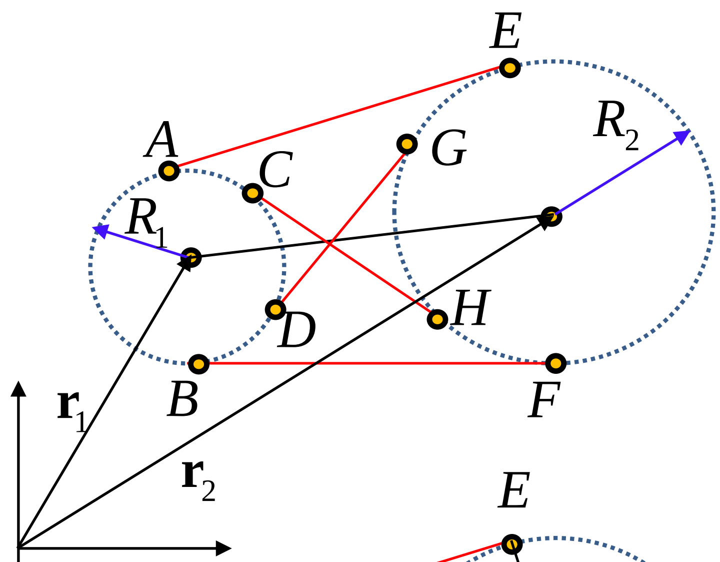


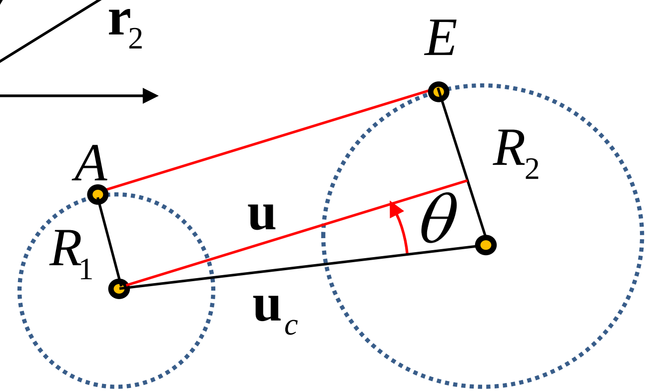


Flyable Trajectory Generation using Dubins Path

Kinematical Relations of Dubins Path for Applications in 3-D Space







$$d = \|\mathbf{r}_2 - \mathbf{r}_1\|$$

$$\mathbf{u}_c = \frac{\mathbf{r}_2 - \mathbf{r}_1}{d}$$

$$d \sin\theta = R_2 - R_1$$

$$\mathbf{u}_1 = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \mathbf{u}_c$$

$$\mathbf{n}_1 = \mathbf{k} \times \mathbf{u}_1$$

$$\mathbf{r}_A = \mathbf{r}_1 + R_1 \mathbf{n}_1$$

$$\mathbf{r}_E = \mathbf{r}_A + d\cos\theta\, \mathbf{u}_1$$

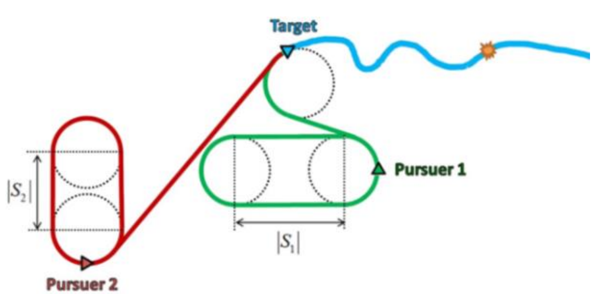
$$\mathbf{u}_2 = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \mathbf{u}_c$$

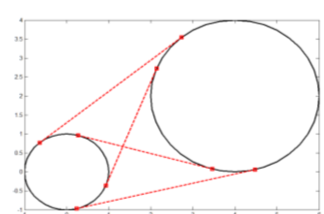
$$\mathbf{n}_2 = -\mathbf{k} \times \mathbf{u}_2$$

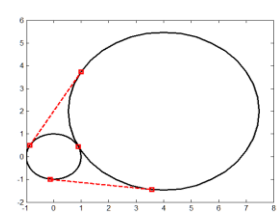
$$\mathbf{r}_B = \mathbf{r}_1 + R_1 \mathbf{n}_2$$

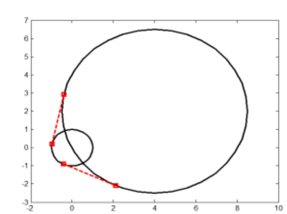
$$\mathbf{r}_F = \mathbf{r}_B + d\cos\theta\, \mathbf{u}_2$$

Path Elongation





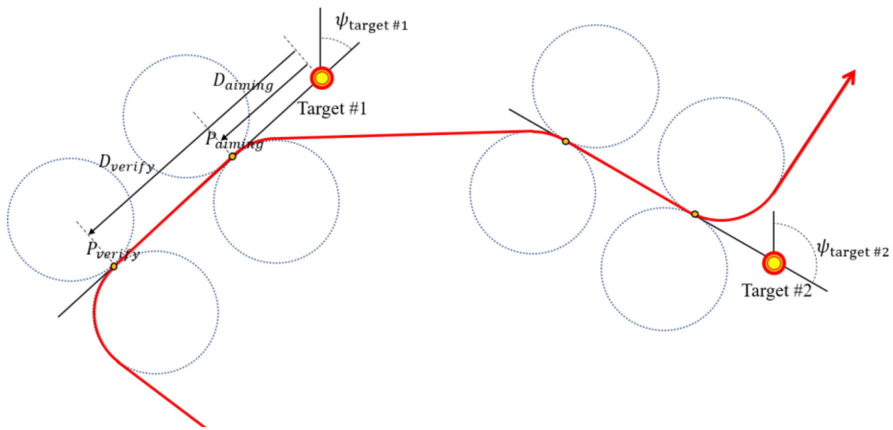




Activities at Initial Stage of Autonomous FCS Research

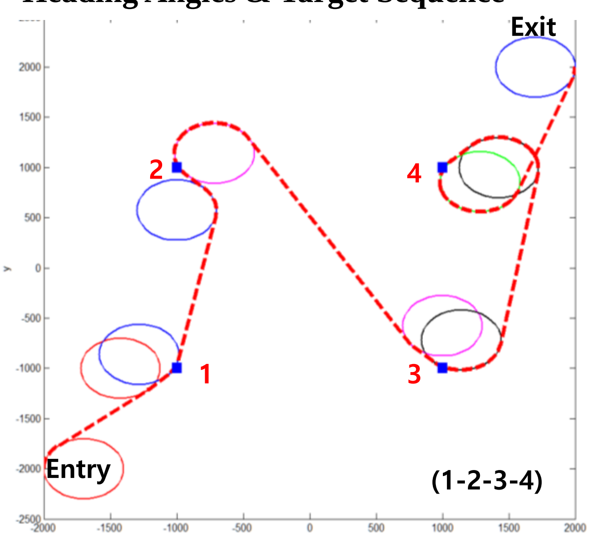
Flyable Trajectory Generation using Dubins Path

Applications to Optimal Trajectory Generation for Multi-Target-Intercept Mission

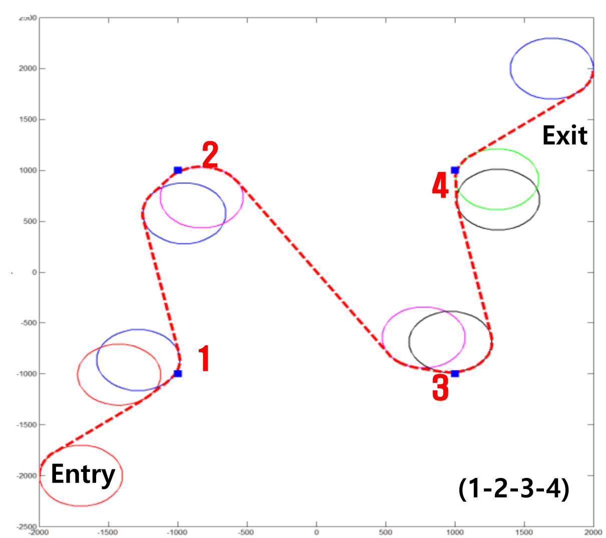


- P_{verify} : The point at which the target is identified, and the aim is stared
- D_{verify} : Distance between Target and P_{verify}
- P_{aiming} : The point at which aiming process is completed and launching the missile
- D_{aiming} : Distance between Target and P_{aiming}
- $\psi_{target\ \#n}$: n-th target heading angle

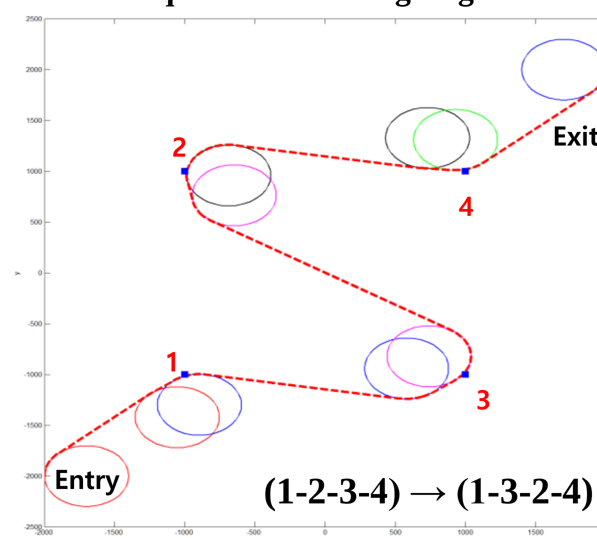
1. Dubins Path with Prescribed Intercept-Heading Angles & Target Sequence



2. Optimized Heading Angles



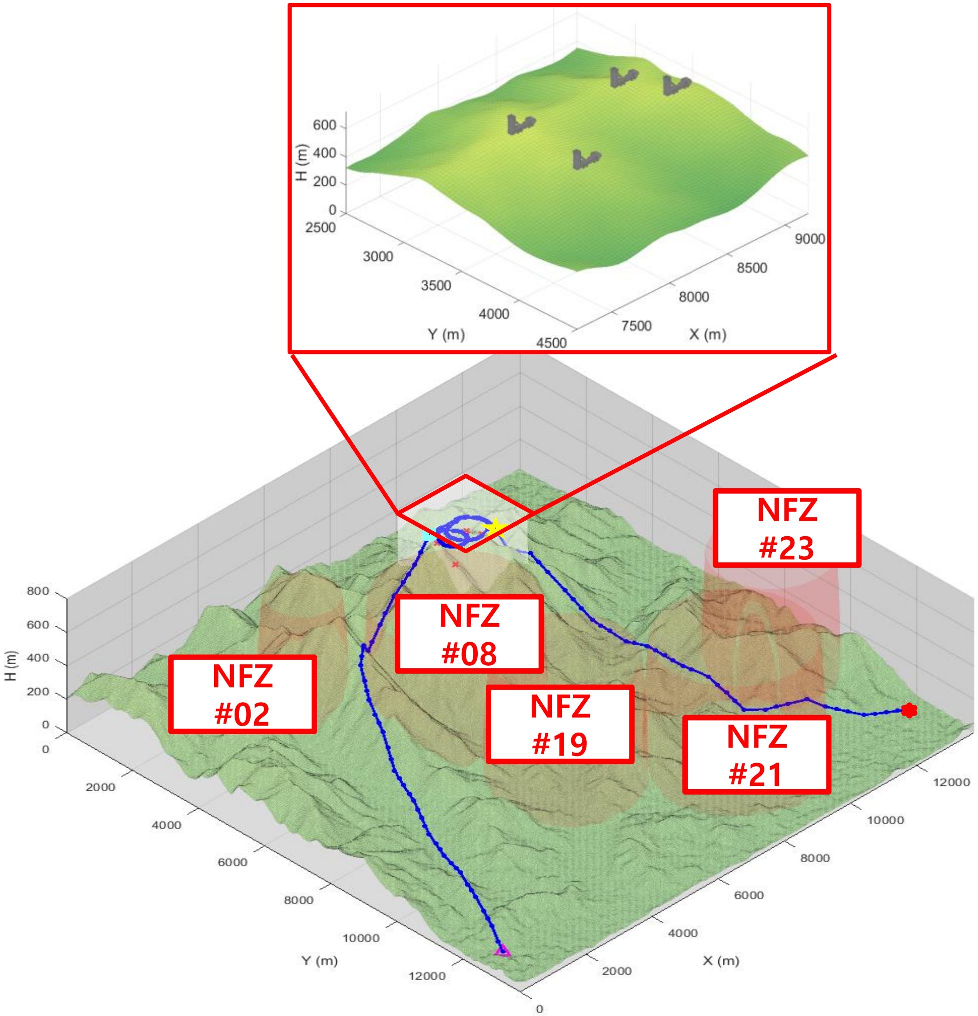
3. Optimized Target Intercept Sequence & Heading Angles

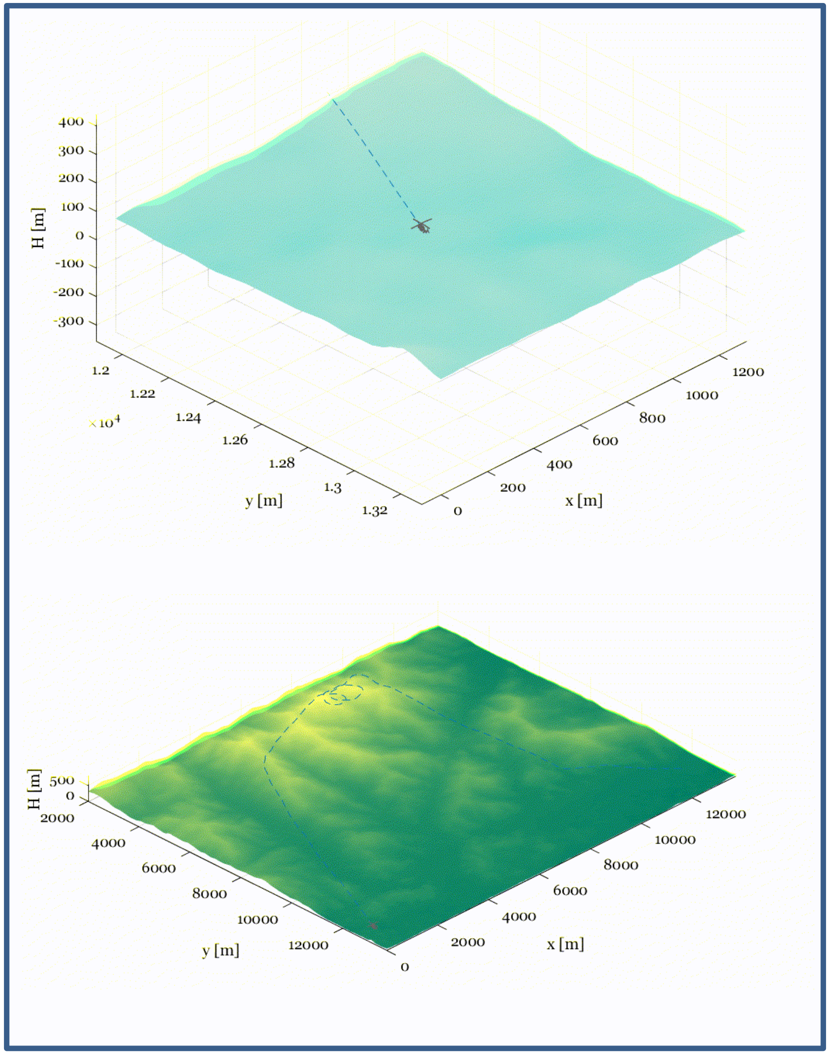


Activities at Initial Stage of Autonomous FCS Research

Flyable Trajectory Generation using Dubins Path

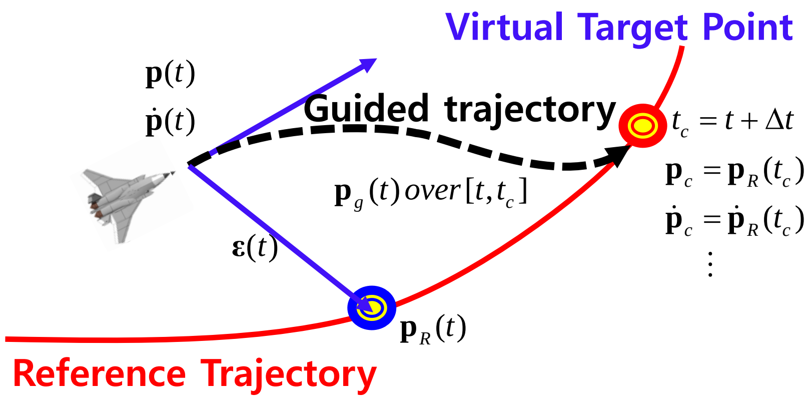
Applications to Optimal Trajectory Generation for Multi-Target-Intercept Mission





Development of Ahead-Time Based Carrot Chasing Guidance Law (CCGL)

Ahead-Time based Carrot-Chasing Algorithm



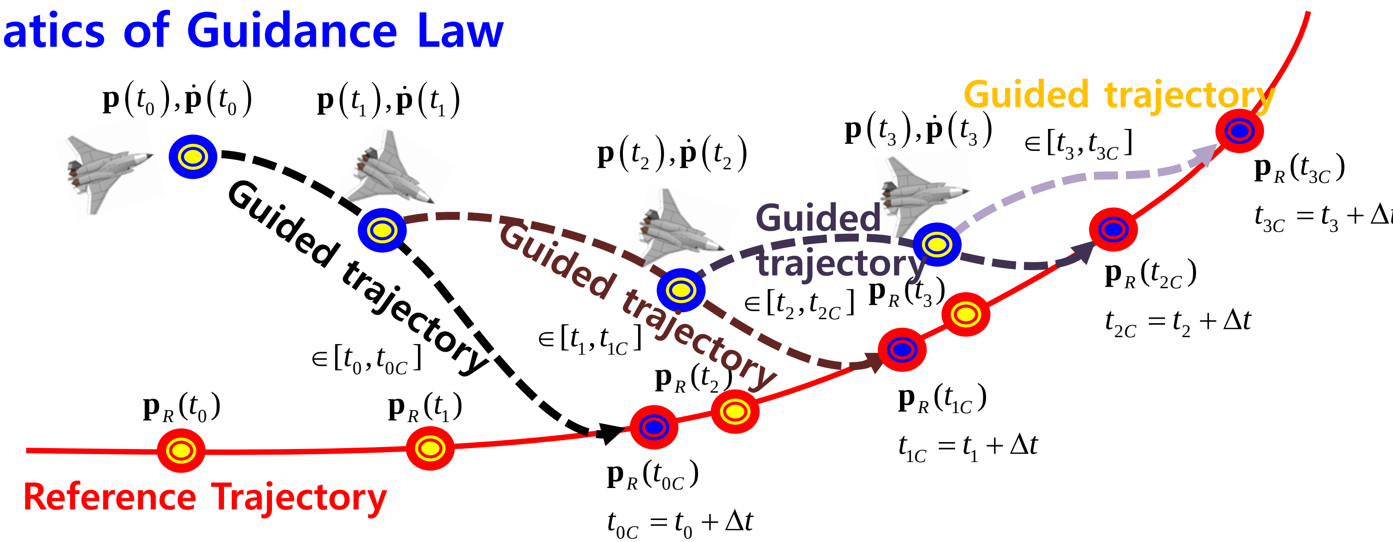
Two different trajectory are used.

1. Reference Trajectory
2. Guided Trajectory

: VTP on Reference Trajectory at time t

: VTP on Reference Trajectory at time t + Δt

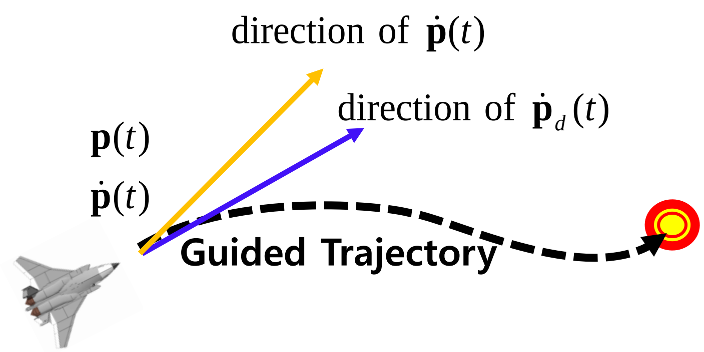
Schematics of Guidance Law



Activities at Initial Stage of Autonomous FCS Research

Development of Ahead-Time Based Carrot Chasing Guidance Law (CCGL)

Generation of Guided Trajectory using Hermit Spline Curve



$$t_c = t + \Delta t$$
$$\mathbf{p}_c = \mathbf{p}_R(t_c)$$
$$\dot{\mathbf{p}}_c = \dot{\mathbf{p}}_R(t_c)$$
$$\vdots$$

Available Aircraft States

$\mathbf{p}(t)$: Aircraft Position

$\dot{\mathbf{p}}(t)$: Velocity

Available VTP States

$\mathbf{p}_R(t_c)$: Desired Position

$\dot{\mathbf{p}}_R(t_c)$: Velocity

$\ddot{\mathbf{p}}_R(t_c)$: Acceleration

$\dddot{\mathbf{p}}_R(t_c)$: Jerk

Using the 1st , 2nd derivatives for initial (aircraft) and final (carrot) states, $\boxed{\{t_0, f_0, \dot{f}_0, \ddot{f}_0\}, \{t_f, f_f, \dot{f}_f, \ddot{f}_f\}}$

$$\boxed{f(\tau) = \sum_{m=0}^{m=5} a_m \tau^m = a_0 + a_1\tau + a_2\tau^2 + a_3\tau^3 + a_4\tau^4 + a_5\tau^5}$$

$$\boxed{\begin{aligned} f(t) &= \alpha_0 f_0 + \alpha_1 \dot{f}_0 \Delta t + \alpha_2 \ddot{f}_0 (\Delta t)^2 + \beta_0 f_0 + \beta_1 \dot{f}_0 \Delta t + \beta_2 \ddot{f}_0 (\Delta t)^2 \\ &= \left(1 - 10\tau^3 + 15\tau^4 - 6\tau^5\right) f_0 + \left(10\tau^3 - 15\tau^4 + 6\tau^5\right) f_1 \\ &\quad + \left(\tau - 6\tau^3 + 8\tau^4 - 3\tau^5\right) \dot{f}_0 \Delta t + \left(-4\tau^3 + 7\tau^4 - 3\tau^5\right) \dot{f}_1 \Delta t + \frac{1}{2}\left(\tau^2 - 3\tau^3 + 3\tau^4 - \tau^5\right) \ddot{f}_0 (\Delta t)^2 + \frac{1}{2}\left(\tau^3 - 2\tau^4 + \tau^5\right) \ddot{f}_1 (\Delta t)^2 \end{aligned}}$$

Development of Ahead-Time Based Carrot Chasing Guidance Law (CCGL)

Comparison of Acceleration Command for MFC (Model-Following-Control)

Ahead-Time based CCGL

(PID + Feedforward Control)

$$\ddot{\mathbf{p}}_{cmd}(t) = k_p(\mathbf{p}_d - \mathbf{p}) + k_d(\dot{\mathbf{p}}_d - \dot{\mathbf{p}}) + k_i\int(\mathbf{p}_d - \mathbf{p})dt + k_{ff}\ddot{\mathbf{p}}_d$$

Traditional (Ahead-distance based) CCGL

$$a_{cmd}^{lat} = k_\psi \Delta\psi + k_d d$$

$$a_{cmd}^{vert} = k_\theta(\theta_{cmd} - \theta_a) + k_h(h_t - h_a)$$

Various Options for Ahead-Time based CCGL: 8 different Guidance Laws

	Aircraft states	Target states
GL1	$\mathbf{p}(t)$	$\mathbf{p}_R(t_c)$
GL2	$\mathbf{p}(t)$	$\mathbf{p}_R(t_c), \dot{\mathbf{p}}_R(t_c)$
GL3	$\mathbf{p}(t)$	$\mathbf{p}_R(t_c), \dot{\mathbf{p}}_R(t_c), \ddot{\mathbf{p}}_R(t_c)$
GL4	$\mathbf{p}(t)$	$\mathbf{p}_R(t_c), \dot{\mathbf{p}}_R(t_c), \ddot{\mathbf{p}}_R(t_c), \dddot{\mathbf{p}}_R(t_c)$

	Aircraft states	Target states
GL5	$\mathbf{p}(t), \dot{\mathbf{p}}(t)$	$\mathbf{p}_R(t_c)$
GL6	$\mathbf{p}(t), \dot{\mathbf{p}}(t)$	$\mathbf{p}_R(t_c), \dot{\mathbf{p}}_R(t_c)$
GL7	$\mathbf{p}(t), \dot{\mathbf{p}}(t)$	$\mathbf{p}_R(t_c), \dot{\mathbf{p}}_R(t_c), \ddot{\mathbf{p}}_R(t_c)$
GL8	$\mathbf{p}(t), \dot{\mathbf{p}}(t)$	$\mathbf{p}_R(t_c), \dot{\mathbf{p}}_R(t_c), \ddot{\mathbf{p}}_R(t_c), \dddot{\mathbf{p}}_R(t_c)$

For Detailed Comparative Study on 8 Guidance Laws, you can refer to [Ref 1]

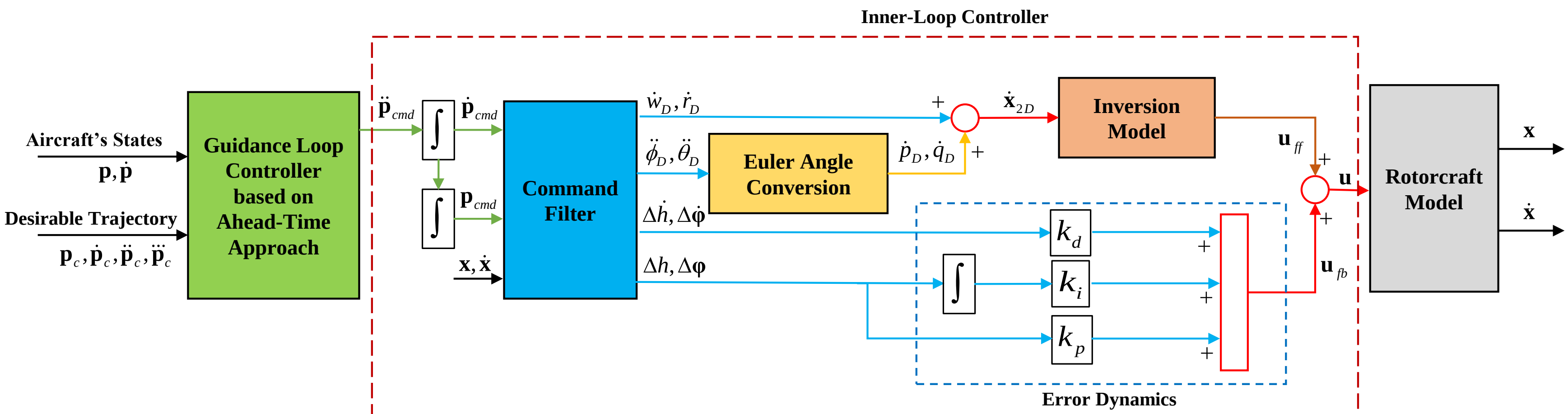
[Ref 1] Seong Han Lee, Sung Wook Hur, Yi Young Kwak, Yong Hyeon Nam, and Chang-Joo Kim, "Ahead-time Approach to Carrot-chasing Guidance Law for an Accurate Trajectory-tracking Control, "International Journal of Control, Automation and Systems 19(8) (2021) 2634-2651

Activities at Initial Stage of Autonomous FCS Research

Development of Ahead-Time Based Carrot Chasing Guidance Law (CCGL)

Model-Following-Control Structure for CCGL Implementation

- **Outer Loop** : Carrot-Chasing Guidance Law (GL1, GL3)
- **Inner Loop** : Model Following Controller (MFC)



Command Filter

$$\phi_{cmd} = \kappa_v \Delta v + \kappa_{vI} \int \Delta v dt$$

$$\theta_{cmd} = -\kappa_u \Delta u - \kappa_{uI} \int \Delta u dt$$

$$r_{cmd} = \kappa_\psi \Delta \psi + \kappa_{\dot\psi} \Delta \dot\psi + \kappa_{\psi I} \int \Delta \psi dt$$

$$\dot h_{cmd} = \dot h_{trim} + \kappa_{\dot h} \Delta \dot h + \kappa_h \Delta h + \kappa_{hI} \int \Delta h dt$$

where

$$\Delta v = v_{cmd} - v, \quad \Delta u = u_{cmd} - u, \quad \Delta \psi = \psi_{cmd} - \psi, \quad \Delta h = h_{cmd} - h$$

$$\ddot\phi_D + 2\zeta_\varphi \omega_\varphi \dot\phi_D + \omega_\varphi^2 \phi_D = \omega_\varphi^2 \phi_{cmd}, \quad \tau_r \dot r_D + r_D = r_{cmd}$$

$$\ddot\theta_D + 2\zeta_\theta \omega_\theta \dot\theta_D + \omega_\theta^2 \theta_D = \omega_\theta^2 \theta_{cmd}, \quad \tau_{\dot h} \ddot h_D + \dot h_D = \dot h_{cmd}$$

Inversion Model

$$\mathbf{u}_{ff} = \mathbf{B}_2^{-1}(\dot{\mathbf{x}}_{2D} - \mathbf{A}_{21}\mathbf{x}_1 - \mathbf{A}_{22}\mathbf{x}_2)$$

Error Dynamics

$$\mathbf{u}_{fb} = \begin{pmatrix} -k_{dh}\Delta \dot h - k_{ph}\Delta h - k_{ih}\int \Delta h dt \\ k_{d\phi}\Delta\dot\phi + k_{p\phi}\Delta\phi + k_{pv}\Delta v + k_{iv}\int \Delta v dt \\ k_{d\theta}\Delta\dot\theta + k_{p\theta}\Delta\theta - k_{pu}\Delta u - k_{iu}\int \Delta u dt \\ k_{d\psi}\Delta\dot\psi + k_{p\psi}\Delta\psi + k_{i\psi}\int \Delta\psi dt \end{pmatrix}$$

where

$$\Delta\phi = \phi_D - \phi, \quad \Delta\theta = \theta_D - \theta$$

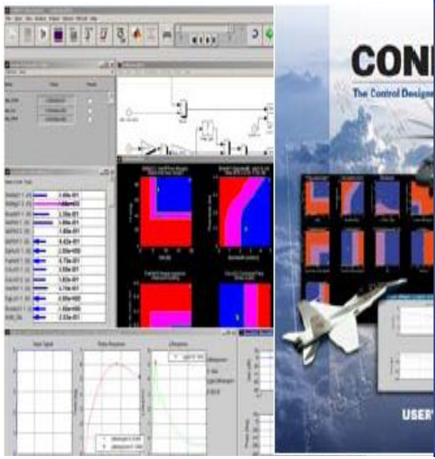
Activities at Initial Stage of Autonomous FCS Research

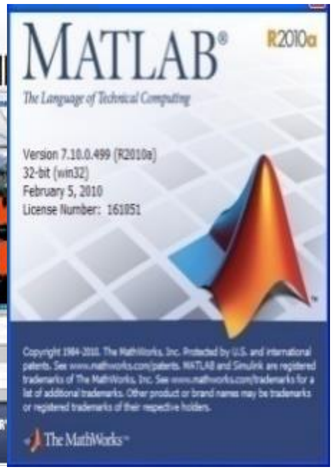
Development of Ahead-Time Based Carrot Chasing Guidance Law (CCGL)

Optimization of Controller Gains and Parameters

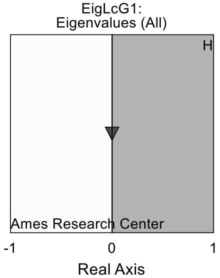
CONDUIT ®
graphical interface & User Manual

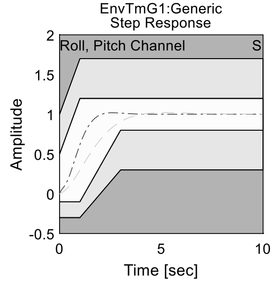
MATLAB ® SIMULINK

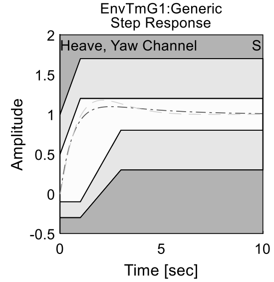




Specification	Description	Source	Channel	Constraint Type
EigLcG1	Eigenvalues (Stability)	AMES Research Center	All	Hard
EnvTmG1	Step Response	General	X,Y,Z,ψ	Soft





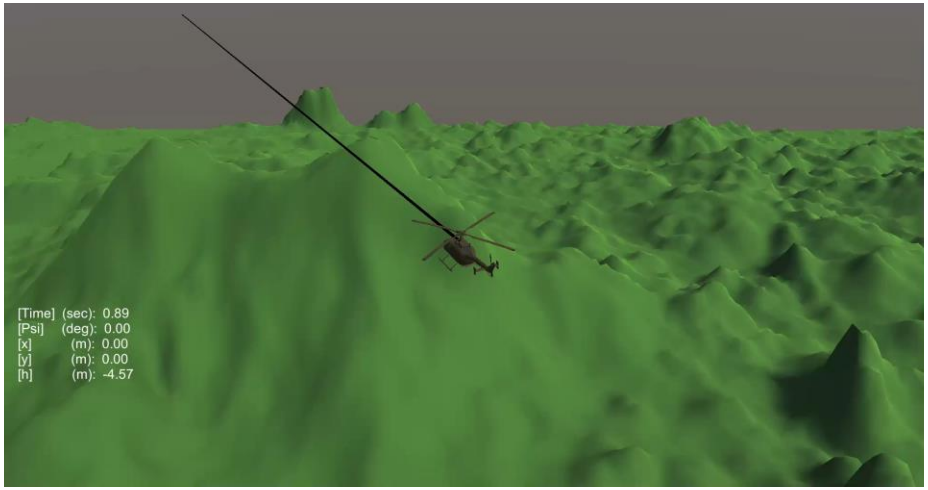


Parameters	Location	Hover	30 knots	60 knots	90knots	120 knots
$K_{\dot{h}}$	Filter (Heave)	1.263E+00	2.533E+00	1.644E+00	7.931E-01	2.383E+00
K_h	Filter (Heave)	5.988E-01	8.037E-01	6.296E-01	1.889E-01	4.488E-01
K_{hI}	Filter (Heave)	9.968E-04	9.327E-04	1.032E-03	2.032E-03	8.490E-04
$K_{\dot{\psi}}$	Filter (Yaw)	9.184E-01	5.670E-01	2.542E-01	1.079E-01	5.276E-01
K_ψ	Filter (Yaw)	5.170E-01	7.714E-01	1.017E-01	4.015E-02	5.328E-01
$K_{\psi I}$	Filter (Yaw)	8.585E-04	1.116E-03	9.864E-04	7.864E-04	8.490E-04
K_u	Filter (Pitch)	1.316E-02	1.146E-02	5.158E-02	9.397E-02	1.344E-02
K_{uI}	Filter (Pitch)	1.167E-03	9.792E-04	9.505E-04	9.505E-04	8.490E-04
K_v	Filter (Roll)	9.036E-03	1.014E-02	6.562E-02	6.220E-02	1.529E-02
K_{vI}	Filter (Roll)	8.976E-04	1.080E-03	1.056E-03	1.056E-03	8.490E-04
ω_ϕ	Filter (Roll)	3.533E+00	4.504E+00	7.121E+00	8.122E+00	5.813E+00
ω_θ	Filter (Pitch)	5.456E+00	3.704E+00	9.064E+00	7.044E+00	5.874E+00
τ_r	Filter (Yaw)	1.510E+00	1.774E+00	1.179E+00	9.628E-01	1.696E+00
$\tau_{\dot{h}}$	Filter (Heave)	1.643E+00	1.697E+00	1.770E+00	1.453E+00	3.549E+00
ζ_ϕ	Filter (Roll)	9.000E-01	9.000E-01	9.000E-01	9.000E-01	9.000E-01
ζ_θ	Filter (Pitch)	9.000E-01	9.000E-01	9.000E-01	9.000E-01	9.000E-01
$k_{d\phi}$	Feedback (Roll)	2.907E-01	2.505E-01	3.433E-01	3.778E-01	2.935E-01
$k_{p\phi}$	Feedback (Roll)	3.494E-01	3.842E-01	5.905E-01	5.282E-01	3.420E-01
k_{pv}	Feedback (Roll)	3.164E-02	2.384E-02	9.807E-05	3.729E-02	9.749E-03
k_{iv}	Feedback (Roll)	9.590E-05	1.173E-04	9.795E-05	4.130E-04	2.734E-03
$k_{d\theta}$	Feedback (Pitch)	2.192E-01	2.223E-01	1.920E-01	1.969E-01	5.648E-01
$k_{p\theta}$	Feedback (Pitch)	4.833E-01	6.027E-01	6.858E-01	7.262E-01	5.073E-01
k_{pu}	Feedback (Pitch)	6.808E-02	6.893E-02	1.268E-02	6.246E-03	1.674E-02
k_{iu}	Feedback (Pitch)	1.100E-04	1.099E-04	9.950E-05	4.369E-03	1.658E-03
$k_{d\psi}$	Feedback (Yaw)	7.530E-01	6.540E-01	9.275E-01	1.142E+00	8.872E-01
$k_{p\psi}$	Feedback (Yaw)	1.869E-01	5.087E-01	7.197E-01	6.121E-01	4.516E-01
$k_{i\psi}$	Feedback (Yaw)	1.104E-04	1.010E-04	1.048E-04	1.007E-04	1.694E-03
k_{ph}	Feedback (Heave)	4.478E-04	8.351E-05	7.528E-02	4.478E-02	2.775E-01
k_{ih}	Feedback (Heave)	0.000E+00	0.000E+00	0.000E+00	1.181E-03	4.441E-03
k_{dh}	Feedback (Heave)	6.003E-01	9.743E-01	8.052E-01	1.023E+00	8.172E-01

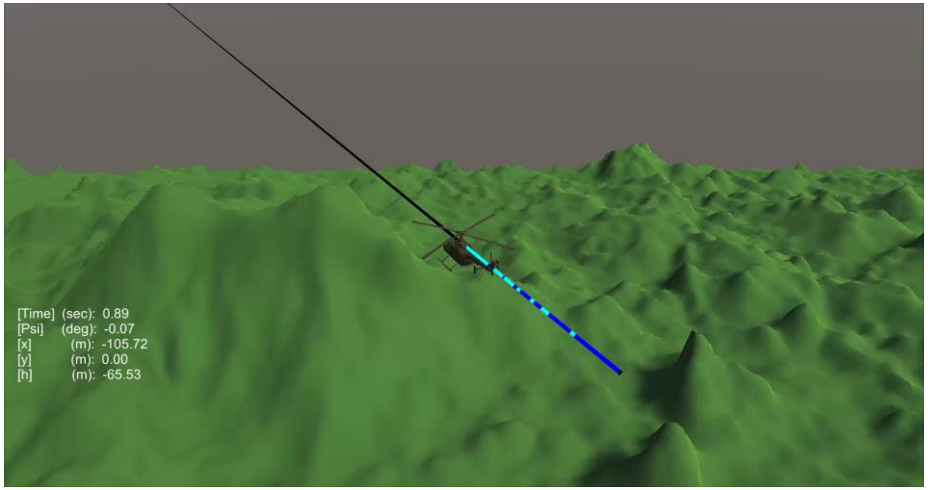
Activities at Initial Stage of Autonomous FCS Research

Development of Ahead-Time Based Carrot Chasing Guidance Law (CCGL)

Applications of CCGL to Acceleration and Deceleration Maneuvers



Acceleration Maneuver



Deceleration Maneuver

GL1 : Yellow
GL3 : Blue
GL5 : Cyan
GL6 : Red

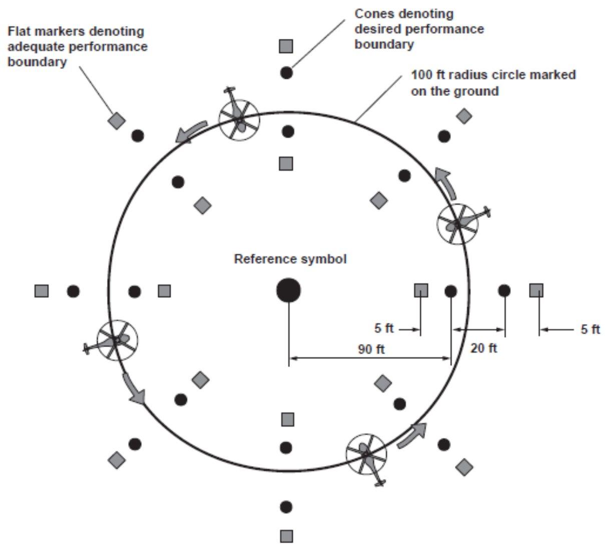
GL1 = GL5
GL3 = GL6

Activities at Initial Stage of Autonomous FCS Research

Development of Ahead-Time Based Carrot Chasing Guidance Law (CCGL)

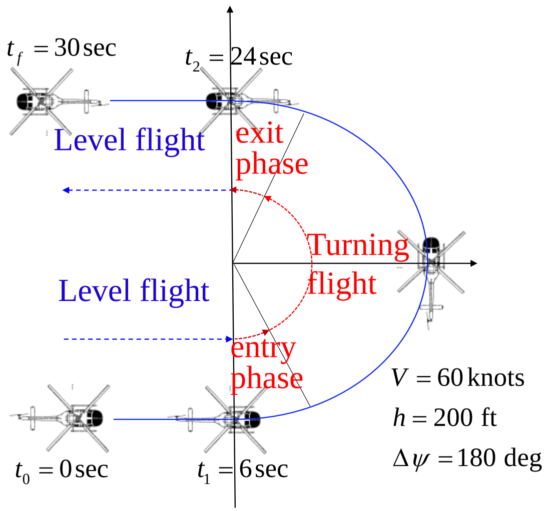
Applications of CCGL to Piroutte, Transient Turn, and Helical Turn Maneuvers

Pirouette Maneuver



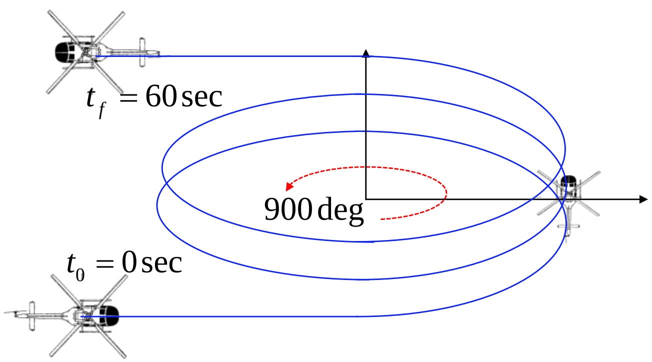
Complete the circle within 60sec
Tracking error within 15ft

Transient Turn



Lesser 120 knots velocity

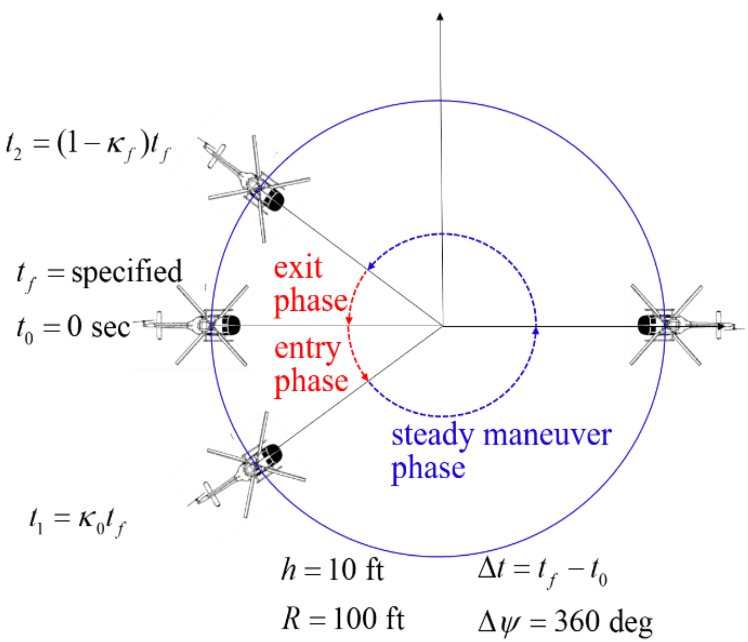
Helical Turn

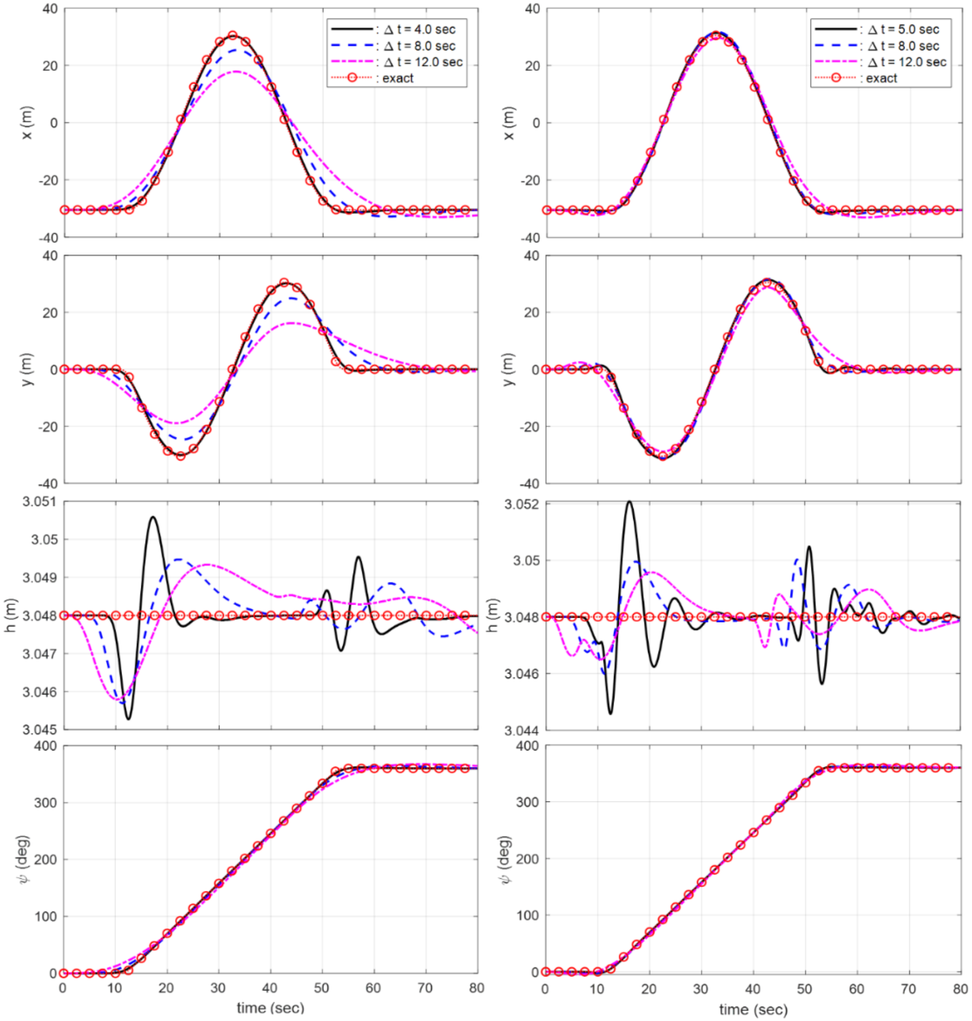


Activities at Initial Stage of Autonomous FCS Research

Development of Ahead-Time Based Carrot Chasing Guidance Law (CCGL)

Applications of CCGL to Piroutte Maneuver (MFC structure, GL1/GL3)



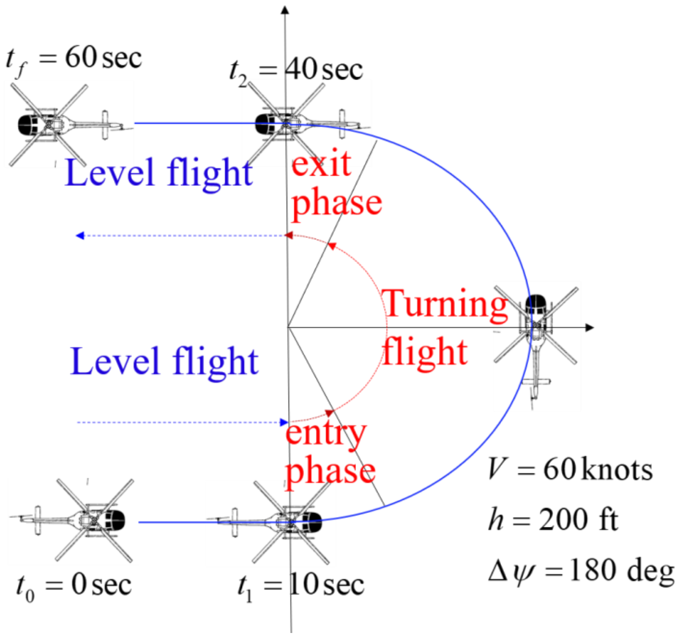


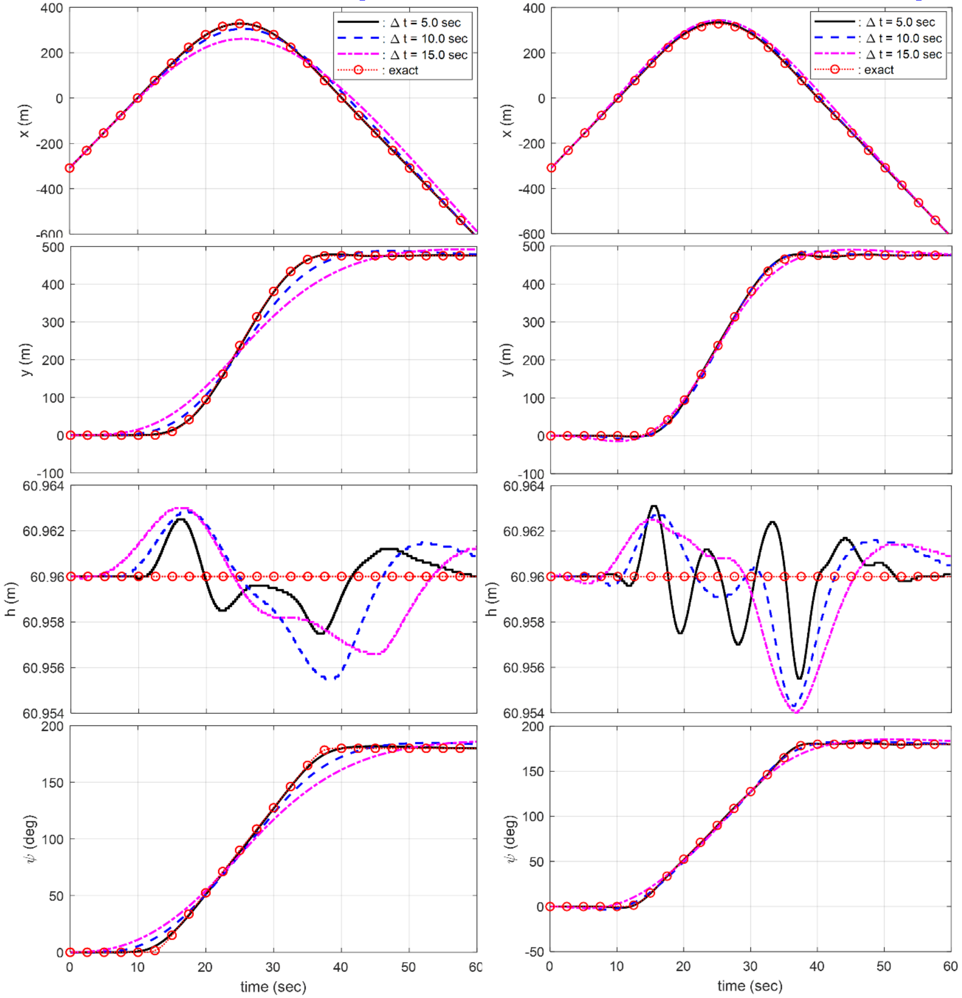
Position and Heading angle(Left:GL1, Right:GL3)

Activities at Initial Stage of Autonomous FCS Research

Development of Ahead-Time Based Carrot Chasing Guidance Law (CCGL)

Applications of CCGL to Transient Turn Maneuver (MFC structure, GL1/GL3)



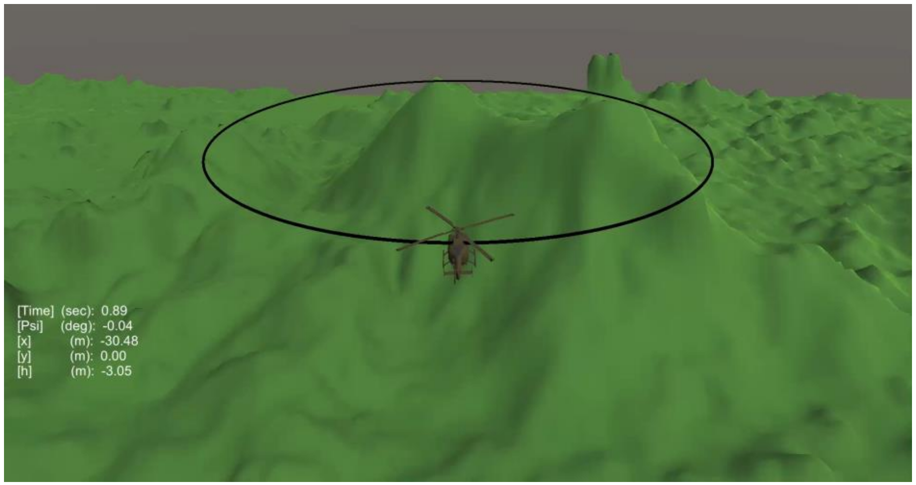


Position and Heading angle(Left:GL1, Right:GL3)

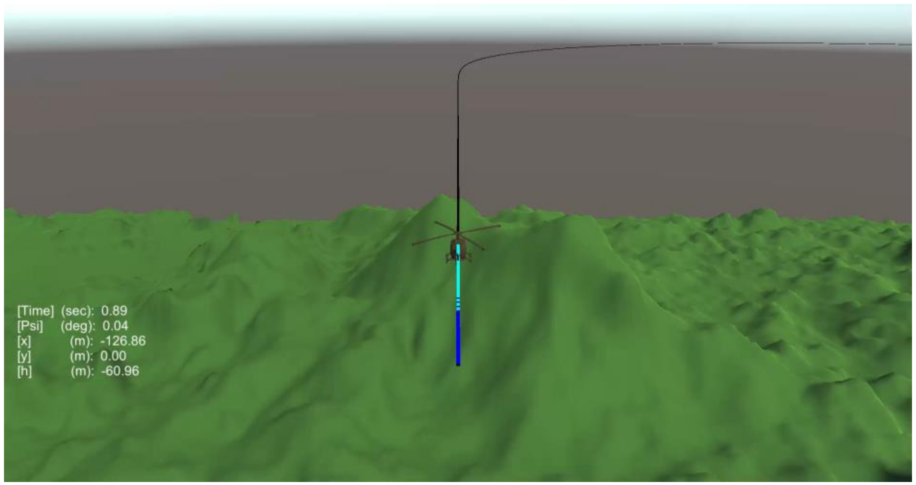
Activities at Initial Stage of Autonomous FCS Research

Development of Ahead-Time Based Carrot Chasing Guidance Law (CCGL)

Applications of CCGL to Piroutte, Transient Turn, and Helical Turn Maneuvers



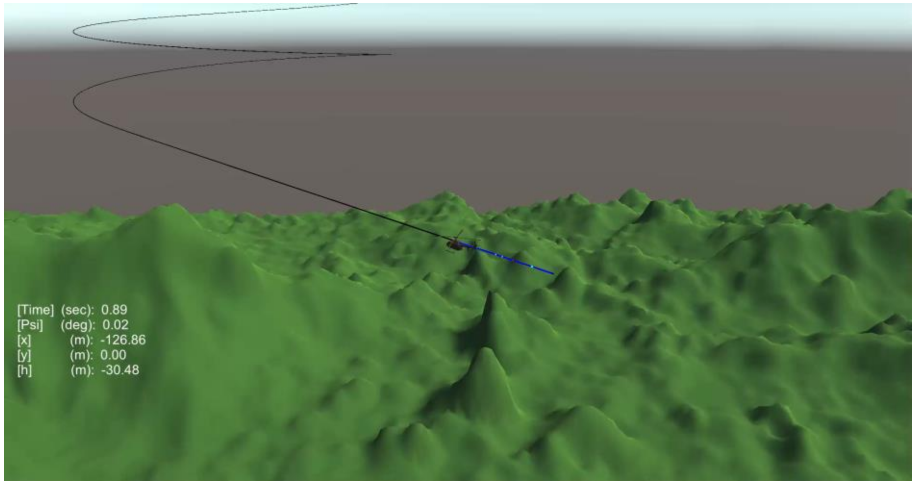
Pirouette



Transient Turn

GL1 : Yellow
GL3 : Blue
GL5 : Cyan
GL6 : Red

GL1 = GL5
GL3 = GL6



Helical Turn

Activities at Initial Stage of Autonomous FCS Research

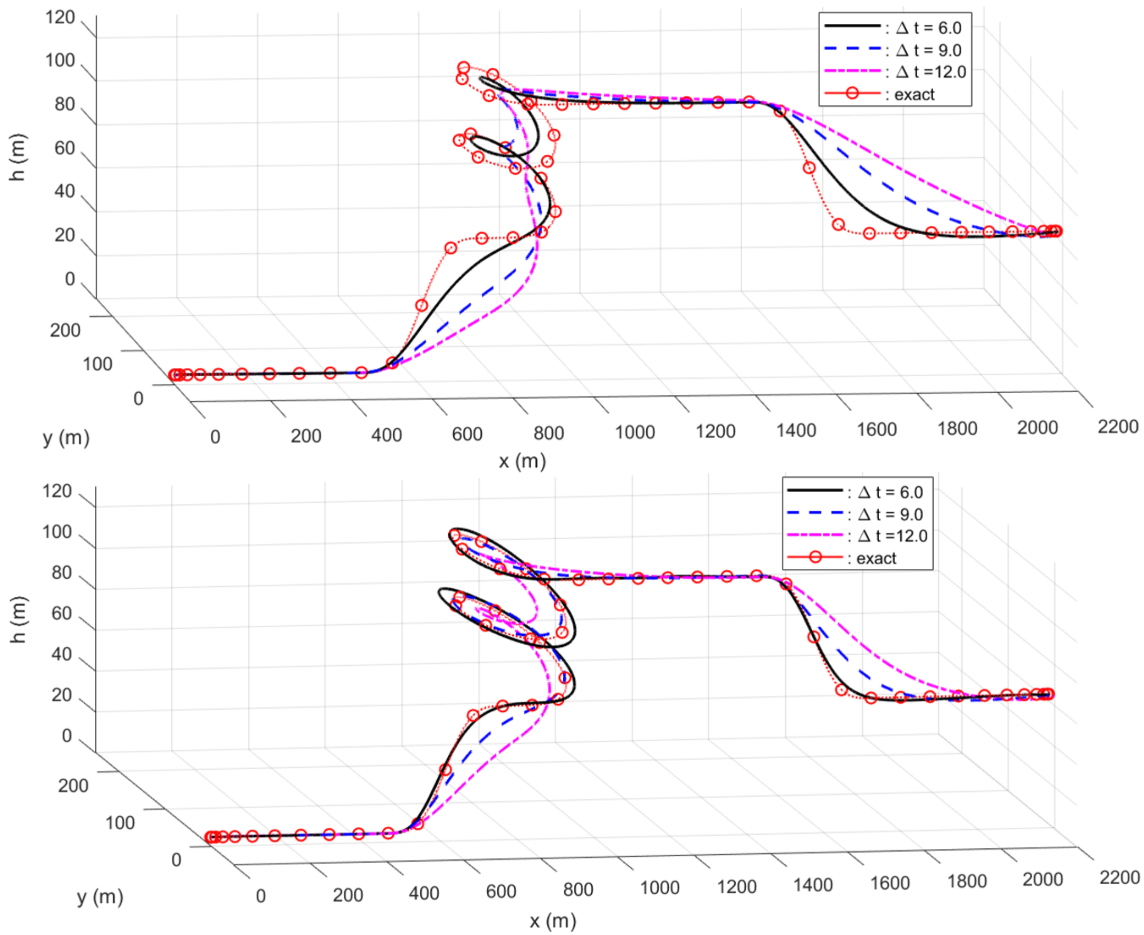
Development of Ahead-Time Based Carrot Chasing Guidance Law (CCGL)

Applications of CCGL to Composite Maneuver (MFC structure, GL1/GL3)

No.	Maneuvers	Duration	State changes during maneuver
1	Hover	15sec	Hover station-keeping at 15ft
2	Acceleration	30sec	Level acceleration from 0 knots to 60knots
3	Pop-up	10sec	Climb from 15 ft up to 215ft and recover the level f light condition
4	Helical Turn	60sec	After 720 deg heading and 200 ft altitude changes, return to level flight
5	Pop-down	10sec	Descent from 415 ft to 215ft and recover the level flight condition
6	Deceleration	30sec	Level deceleration from 60 knots to hover

Development of Ahead-Time Based Carrot Chasing Guidance Law (CCGL)

Applications of CCGL to Composite Maneuver (MFC structure, GL1/GL3)

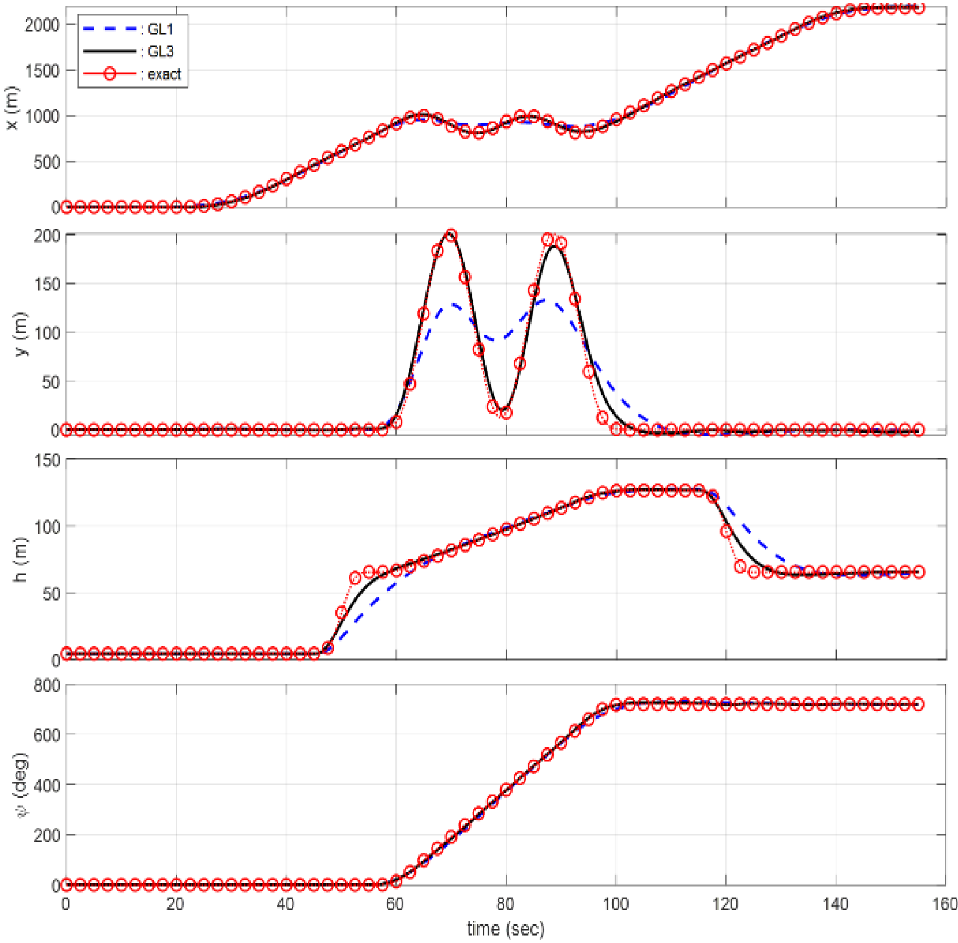


Effect of ahead time on trajectory-tracking accuracy (Upper:GL1, Lower:GL3)

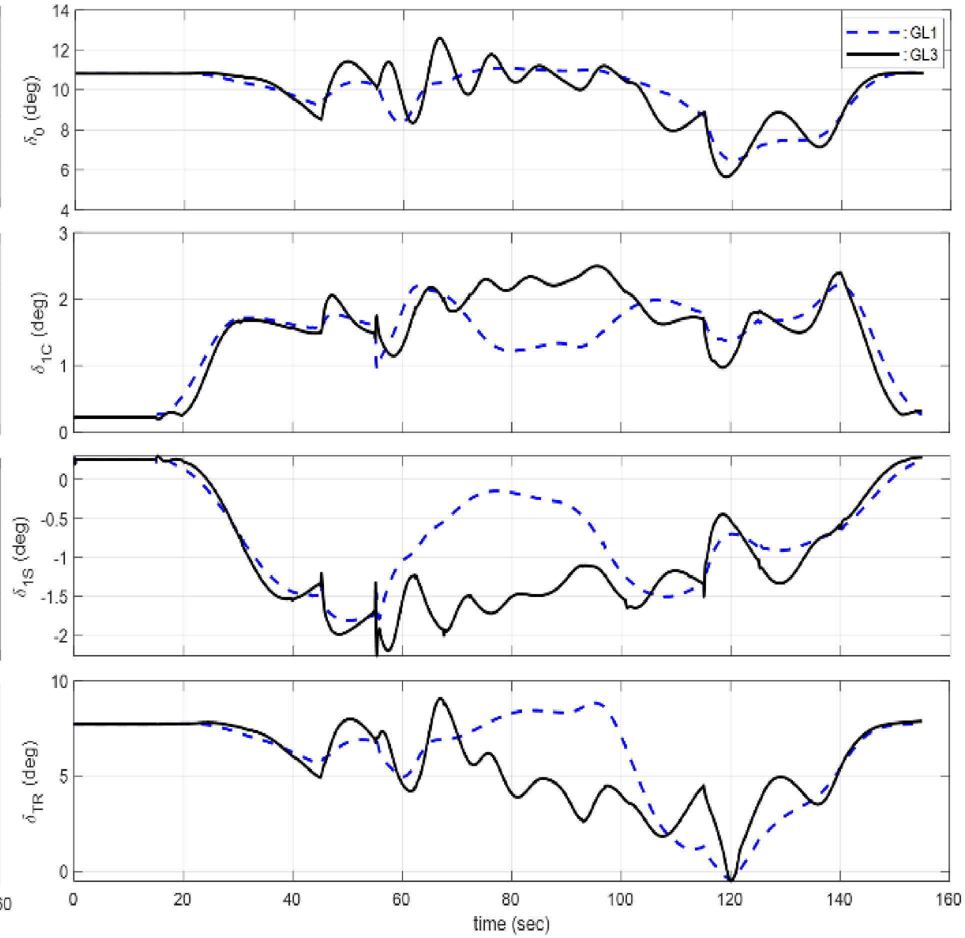
Activities at Initial Stage of Autonomous FCS Research

Development of Ahead-Time Based Carrot Chasing Guidance Law (CCGL)

Applications of CCGL to Composite Maneuver (MFC structure, GL1/GL3)



Comparison of trajectory tracking accuracy

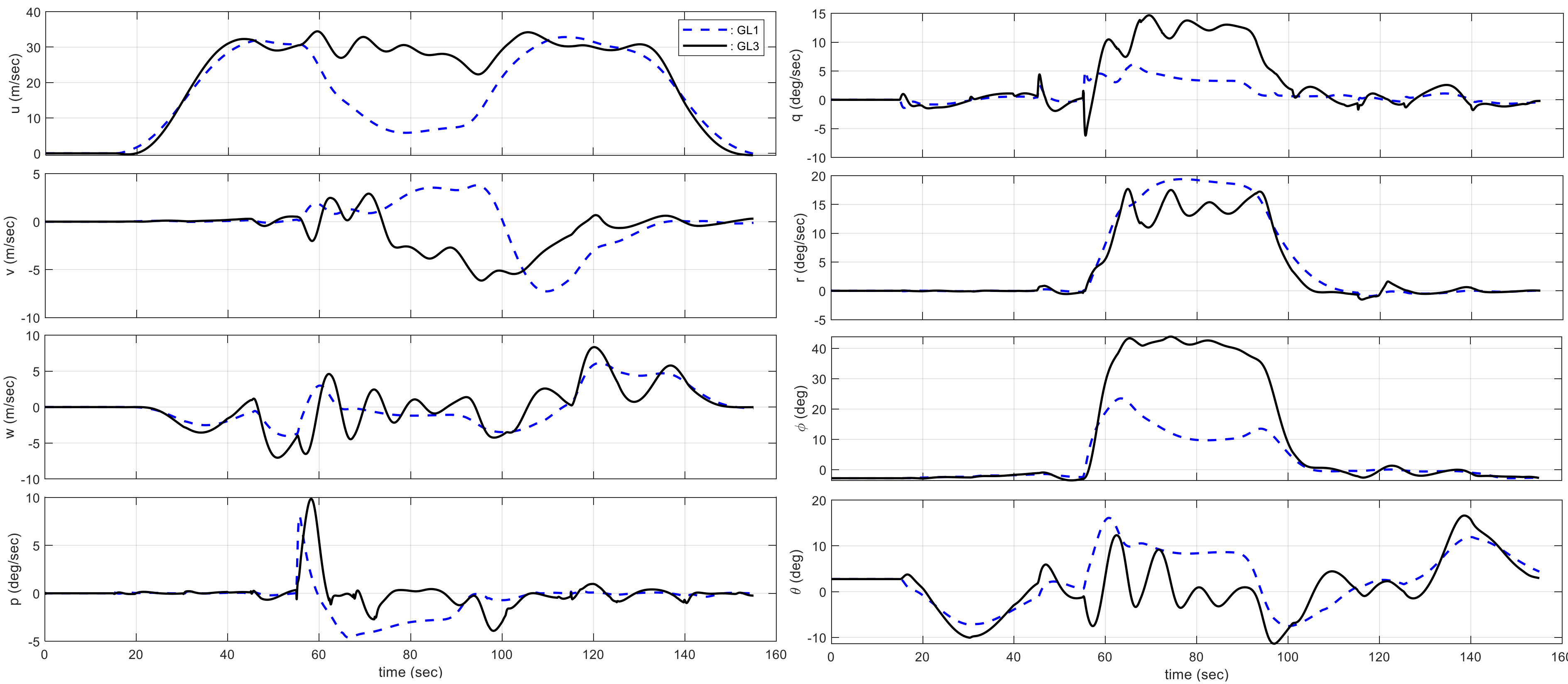


Time history of control inputs with $\Delta t = 9.0\,(\mathrm{sec})$

Activities at Initial Stage of Autonomous FCS Research

Development of Ahead-Time Based Carrot Chasing Guidance Law (CCGL)

Applications of CCGL to Composite Maneuver (MFC structure, GL1/GL3)



Aircraft states computed with $\Delta t = 9.0\,(\text{sec})$

Contents

Recent Publications

Development of Lyapunov-based Nonlinear Trajectory-Tracking Controller (Back-Stepping / Sliding-Mode Control Design)

- **Chang-Joo Kim, et al., "Adaptive Trajectory Tracking Control for Rotorcraft Using Incremental Backstepping Sliding Mode Control Strategy," International Journal of Aerospace Engineering 2021:1-15, July 2021.**
- **Chang-Joo Kim, et al., "Efficient Gain Parameter Selection Approach for Sliding Mode Control with Application to Rotorcraft Trajectory Tracking Control Design," The Proceedings of the 2021 Asia-Pacific International Symposium on Aerospace Technology (APISAT 2021), Volume 2, September 2022**
- **Chang-Joo Kim, et al., "Robust Trajectory-Tracking Control of a Rotorcraft Using Immersion-and-Invariance-Based Adaptive Backstepping Control, " International Journal of Aerospace Engineering 2022(3):1-16, July 2022.**

Development of Nonlinear Trajectory-Tracking Controller using Incremental Dynamics (Incremental Back-Stepping / Sliding-Mode Control Design)

- **Chang-Joo Kim, et al., "Guidance and control for autonomous emergency landing of the rotorcraft using the incremental backstepping controller in 3-dimensional terrain environments, "Aerospace Science and Technology 132:108051, 2022.**
- **Chang-Joo Kim, et al., "Robust Prediction of Angular Acceleration for Practical Realization of Incremental Nonlinear Trajectory-tracking Control for Aircrafts, " International Journal of Control Automation and Systems 20(4):1250-1265, April 2022.**
- **Chang-Joo Kim, et al., "A Trajectory-Tracking Controller Design of Rotorcraft Using an Adaptive Incremental-Backstepping Approach, " Aerospace 8(9):248, September 2021.**

Recent Publications

Integration of Path-Planning, Flyable Trajectory Generation, and Trajectory-Tracking Control for Mission Autonomy

- **Chang-Joo Kim, et al., "A Study on Path Planning Using Bi-Directional PQ-RRT* Algorithm and Trajectory Tracking Technique Using Incremental Backstepping Control, "The Proceedings of the 2021 Asia-Pacific International Symposium on Aerospace Technology (APISAT 2021), Volume 2, September 2022**
- **Chang-Joo Kim, et al., " A Study on Integration of Guidance System Using Real-Time PQ-RRT* Algorithm and a Trajectory Tracking Controller, " Journal of Institute of Control Robotics and Systems 28(1):75-85, January 2022.**
- **Chang-Joo Kim, et al., " An Approach to Air-to-Surface Mission Planner on 3D Environments for an Unmanned Combat Aerial Vehicle," Drones 6(1):20, January 2022**
- **Chang-Joo Kim, et al., "Integration of path planning, trajectory generation and trajectory tracking control for aircraft mission autonomy," Aerospace Science and Technology 118(1):107014, August 2021**

The Presentation will mainly focus on Incremental Back-Stepping Control (IBSC) design for brevity.

Development of IBS Trajectory-Tracking Control

Some of Claims based on Experiences

- **Mission Autonomy can be effectively achieved using Trajectory-Tracking Control.**

  Trajectory-following control : control in 3-D space (time independent) $\boxed{\mathbf{p} = (x, y, z)^T}$

  Trajectory-tracking control : control in 4-D space (exact timing is critical) $\boxed{\mathbf{p} = (x, y, z, t)^T}$

- **Flight Dynamic Model represented in the Inertial Frame is more convenient then the traditional form of Euler Equations, since desirable trajectories for Mission Autonomy are typically prescribed by the position and heading angle.** $\boxed{\mathbf{p}_d = (x, y, z, \psi, t)^T}$

Euler Equations

$$\boxed{\begin{aligned} \dot{\mathbf{v}} &= \mathbf{f}/m - \boldsymbol{\omega} \times \mathbf{v} \\ \dot{\boldsymbol{\omega}} &= \mathbf{J}^{-1}\left\{\mathbf{m} - \boldsymbol{\omega} \times (\mathbf{J}\boldsymbol{\omega})\right\} \end{aligned}} \quad \mathbf{v} = \begin{pmatrix} u \\ v \\ w \end{pmatrix}, \quad \boldsymbol{\omega} = \begin{pmatrix} p \\ q \\ r \end{pmatrix}, \quad \boldsymbol{\varphi} = \begin{pmatrix} \phi \\ \theta \\ \psi \end{pmatrix}, \quad \mathbf{r} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

Motion Equations using inertial states

$$\boxed{\begin{aligned} \ddot{\mathbf{r}} &= \mathbf{C}^{-1}\left\{\mathbf{f}/m - (\mathbf{T}\dot{\boldsymbol{\varphi}}) \times (\mathbf{C}\dot{\mathbf{r}}) - \dot{\mathbf{C}}\dot{\mathbf{r}}\right\} \\ \ddot{\boldsymbol{\varphi}} &= \mathbf{T}^{-1}\left[\mathbf{J}^{-1}\left\{\mathbf{m} - (\mathbf{T}\dot{\boldsymbol{\varphi}}) \times (\mathbf{J}\mathbf{T}\dot{\boldsymbol{\varphi}})\right\} - \dot{\mathbf{T}}\dot{\boldsymbol{\varphi}}\right] \end{aligned}}$$

Using

$$\boldsymbol{\omega} = \mathbf{T}\dot{\boldsymbol{\varphi}} \qquad \dot{\boldsymbol{\omega}} = \dot{\mathbf{T}}\dot{\boldsymbol{\varphi}} + \mathbf{T}\ddot{\boldsymbol{\varphi}}$$

$$\mathbf{v} = \mathbf{C}\dot{\mathbf{r}} \qquad \dot{\mathbf{v}} = \dot{\mathbf{C}}\dot{\mathbf{r}} + \mathbf{C}\ddot{\mathbf{r}}$$

Development of IBS Trajectory-Tracking Control

Some of Claims based on Experiences

- **Incremental Flight Dynamics are much more effective for real applications than Full Nonlinear Dynamics.**
  - It allows controller scheduling only with the control effectiveness matrix
  - The mismatched uncertainty can be removed
  - Adaptive control elements can be straightforwardly adopted

Nonlinear Dynamics

$$\ddot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \dot{\mathbf{x}}) + \mathbf{G}(\mathbf{x}, \dot{\mathbf{x}})\mathbf{u} + \mathbf{d}(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{u})$$

Nonlinear Dynamics at to

$$\ddot{\mathbf{x}}_0 = \mathbf{f}(\mathbf{x}_0, \dot{\mathbf{x}}_0) + \mathbf{G}(\mathbf{x}_0, \dot{\mathbf{x}}_0)\mathbf{u}_0 + \mathbf{d}_0(\mathbf{x}_0, \dot{\mathbf{x}}_0, \mathbf{u})$$

Nonlinear Dynamics at to+△t

$$\ddot{\mathbf{x}} \approx \left(\mathbf{f}_0 + \mathbf{G}_0\mathbf{u}_0 + \mathbf{d}_0\right) + \mathbf{G}_0\Delta\mathbf{u} + \frac{\partial \mathbf{d}}{\partial \mathbf{u}}\Delta\mathbf{u}$$

$$+ \cancel{\left(\frac{\partial \mathbf{f}_0}{\partial \mathbf{x}}\Delta\mathbf{x} + \frac{\partial \mathbf{f}_0}{\partial \dot{\mathbf{x}}}\Delta\dot{\mathbf{x}} + \frac{\partial \mathbf{d}}{\partial \mathbf{x}}\Delta\mathbf{x} + \frac{\partial \mathbf{d}}{\partial \dot{\mathbf{x}}}\Delta\dot{\mathbf{x}}\right)}$$

$$\approx \ddot{\mathbf{x}}_0 + \left(\mathbf{G}_0 + \frac{\partial \mathbf{d}}{\partial \mathbf{u}}\right)\Delta\mathbf{u}$$

Incremental Dynamics at to+△t

$$\boxed{\ddot{\mathbf{x}} \approx \ddot{\mathbf{x}}_0 + \left(\mathbf{G}_0 + \frac{\partial \mathbf{d}}{\partial \mathbf{u}}\right)\Delta\mathbf{u}}$$

Measured or Estimated linear and angular acceleration data are used (You can refer to [Ref 1])

[Ref 1] Chang-Joo Kim, et al., "Robust Prediction of Angular Acceleration for Practical Realization of Incremental Nonlinear Trajectory-tracking Control for Aircrafts, " International Journal of Control Automation and Systems 20(4):1250-1265, April 2022.

Development of IBS Trajectory-Tracking Control

Some of Claims based on Experiences

- Lyapunov-Based Control Design coupled with Incremental Dynamics is easily Certifiable by using Deterministic control effective matrices. $\boxed{\mathbf{G}_0}$

$$\boxed{\ddot{\mathbf{x}} \approx \ddot{\mathbf{x}}_0 + \left(\mathbf{G}_0 + \frac{\partial \mathbf{d}}{\partial \mathbf{u}}\right)\Delta\mathbf{u}}$$

- Slack Variables Approach to System Dynamics is extremely effective to get the non-singular square control effective matrices required for the model inversion. $\boxed{\mathbf{G}}$

$$\ddot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{u}_p) + \mathbf{G}(\mathbf{x}, \dot{\mathbf{x}})\mathbf{u} + \xi + \boxed{\mathbf{d}} \text{ Disturbance}$$

$$\mathbf{G} = \begin{pmatrix} \bar{\mathbf{G}} & \mathbf{G}_s \end{pmatrix}, \quad \xi = -\mathbf{G}_s \underbrace{\boxed{\mathbf{u}_s}}_{\text{Slack variable}}$$

$$\mathbf{x} = \begin{pmatrix} x \\ y \\ z \\ \psi \\ \phi \\ \theta \end{pmatrix} \quad \mathbf{u}_p = \begin{pmatrix} \delta_0 \\ \delta_{1c} \\ \delta_{1s} \\ \delta_{TR} \end{pmatrix} \quad \mathbf{u} = \begin{pmatrix} \mathbf{u}_p \\ \mathbf{u}_s \end{pmatrix} \quad \mathbf{G}_s = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Thus, the fully actuated system dynamics are easily obtained using slack variables.

- SAS-type functions are working well for the trajectory-tracking IBSC

$$\begin{aligned} \phi_d &= \phi_{trim}, & \dot{\phi}_d &= \ddot{\phi}_d = 0 \\ \theta_d &= \theta_{trim}, & \dot{\theta}_d &= \ddot{\theta}_d = 0 \end{aligned}$$

Thus, the prescription of trajectories for pitch and bank angles are not mandatory.

Design of IBS Trajectory-Tracking Controller

Incremental Dynamics

$$\ddot{\mathbf{x}} = \ddot{\mathbf{x}}_0 + \mathbf{G}\Delta\mathbf{u} + \Delta\xi$$

Tracking Error Dynamics

$$\mathbf{z}_1 = \mathbf{x} - \mathbf{x}_d$$

$$\mathbf{z}_2 = \dot{\mathbf{x}} - \boxed{\boldsymbol{\alpha}} \quad \text{Virtual Control}$$

$$\dot{\mathbf{z}}_1 = \mathbf{z}_2 + \boldsymbol{\alpha} - \dot{\mathbf{x}}_d$$

$$\dot{\mathbf{z}}_2 = \ddot{\mathbf{x}}_0 + \mathbf{G}\Delta\mathbf{u} + \Delta\xi - \dot{\boldsymbol{\alpha}}$$

Control Lyapunov Function CLF)

$$V = \frac{1}{2}\mathbf{z}_1^T\mathbf{Q}^{-1}\mathbf{z}_1 + \frac{1}{2}\mathbf{z}_2^T\mathbf{z}_2 + \frac{1}{2}\Delta\xi^T\boldsymbol{\Lambda}_\xi^{-1}\Delta\xi$$

$$\begin{pmatrix} \mathbf{Q} = diag(q_j)_{j=1}^{j=6} > 0 \\ \boldsymbol{\Lambda}_\xi = diag(\lambda_{\xi j})_{j=1}^{j=6} > 0 \end{pmatrix}$$

Weight Matrices for CLF

Lyapunov Stability Conditions

$$\begin{aligned} \dot{V} &= \mathbf{z}_1^T\mathbf{Q}^{-1}\dot{\mathbf{z}}_1 + \mathbf{z}_2^T\dot{\mathbf{z}}_2 + \Delta\xi^T\boldsymbol{\Lambda}_\xi^{-1}\Delta\dot{\xi} \\ &= \mathbf{z}_1^T\mathbf{Q}^{-1}(\boldsymbol{\alpha} - \dot{\mathbf{x}}_d) + \mathbf{z}_2^T(\mathbf{Q}^T\mathbf{z}_1 + \ddot{\mathbf{x}}_0 + \mathbf{G}\Delta\mathbf{u} - \dot{\boldsymbol{\alpha}}) \\ &\quad + \Delta\xi^T(\boldsymbol{\Lambda}_\xi^{-1}\Delta\dot{\xi} + \mathbf{z}_2) \le 0 \end{aligned}$$

Control Laws and Update Laws

$$-\mathbf{K}_1\mathbf{z}_1 = \mathbf{Q}^{-1}(\boldsymbol{\alpha} - \dot{\mathbf{x}}_d) \rightarrow \ \boldsymbol{\alpha} = -\mathbf{Q}\mathbf{K}_1\mathbf{z}_1 + \dot{\mathbf{x}}_d$$

$$\begin{aligned} -\mathbf{K}_2\mathbf{z}_2 &= \mathbf{Q}^{-1}\mathbf{z}_1 + \ddot{\mathbf{x}}_0 + \mathbf{G}\Delta\mathbf{u} - \dot{\boldsymbol{\alpha}} \\ &= \mathbf{Q}^{-1}\mathbf{z}_1 + \ddot{\mathbf{x}}_0 + \mathbf{G}\Delta\mathbf{u} + \mathbf{Q}\mathbf{K}_1\dot{\mathbf{z}}_1 - \ddot{\mathbf{x}}_d \end{aligned}$$

$$\boxed{\begin{aligned} &\Delta\mathbf{u} = -\mathbf{G}^{-1}\left\{(\mathbf{K}_2 + \mathbf{Q}\mathbf{K}_1)\dot{\mathbf{z}}_1 + (\mathbf{Q}^{-1} + \mathbf{K}_2\mathbf{Q}\mathbf{K}_1)\mathbf{z}_1 + \ddot{\mathbf{x}}_0 - \ddot{\mathbf{x}}_d\right\} \\ &\mathbf{u} = \mathbf{u}_0 + \Delta\mathbf{u} \end{aligned}}$$

$$\boxed{\begin{aligned} &\Delta\dot{\xi} = -\boldsymbol{\Lambda}_\xi\mathbf{z}_2 \\ &\boldsymbol{\alpha} = -\mathbf{Q}\mathbf{K}_1\mathbf{z}_1 + \dot{\mathbf{x}}_d \end{aligned}}$$

$$\begin{pmatrix} \mathbf{K}_1 = diag(k_{1j})_{j=1}^{j=6} > 0 \\ \mathbf{K}_2 = diag(k_{2j})_{j=1}^{j=6} > 0 \end{pmatrix}$$

Gain Matrices IBSC

Design of IBS Trajectory-Tracking Controller

Error Dynamics with IBS Trajectory-Tracking Control

$$\ddot{\mathbf{z}}_1 = \ddot{\mathbf{x}}_0 + \mathbf{G}\Delta\mathbf{u} + \Delta\boldsymbol{\xi} - \ddot{\mathbf{x}}_d = -(\mathbf{K}_2 + \mathbf{Q}\mathbf{K}_1)\dot{\mathbf{z}}_1 - (\mathbf{Q}^{-1} + \mathbf{K}_2\mathbf{Q}\mathbf{K}_1)\mathbf{z}_1 + \Delta\boldsymbol{\xi}$$

$$\ddot{\mathbf{z}}_1 + (\mathbf{K}_2 + \mathbf{Q}\mathbf{K}_1)\dot{\mathbf{z}}_1 + (\mathbf{Q}^{-1} + \mathbf{K}_2\mathbf{Q}\mathbf{K}_1)\mathbf{z}_1 = \Delta\boldsymbol{\xi}$$

$$\boxed{\ddot{z}_{1,j} + \left(k_{2,j} + q_j k_{1,j}\right)\dot{z}_{1,j} + \left(k_{2,j} q_j k_{1,j} + \frac{1}{q_j}\right) z_{1,j} = \Delta\xi_j, \quad (j = 1, 2, \cdots, 6)}$$

Desirable Error Dynamics and Gain Selections by specifying desirable Damping Ratio and Natural Frequency for each axis

$$k_{2j} + q_j k_{1j} = 2\zeta_j \omega_j$$

$$k_{2j} q_j k_{1j} + \frac{1}{q_j} = \omega_j^2$$

$$\frac{1}{\omega_j^2} \le q_j \le \frac{1}{\left(1-\zeta_j^2\right)\omega_j^2}$$

$$k_{1j} = \frac{1}{q_j}\left(\zeta_j\omega_j \pm \sqrt{\frac{1}{q_j} - \left(1-\zeta_j^2\right)\omega_j^2}\right)$$

$$k_{2j} = \zeta_j\omega_j \mp \sqrt{\frac{1}{q_j} - \left(1-\zeta_j^2\right)\omega_j^2}$$

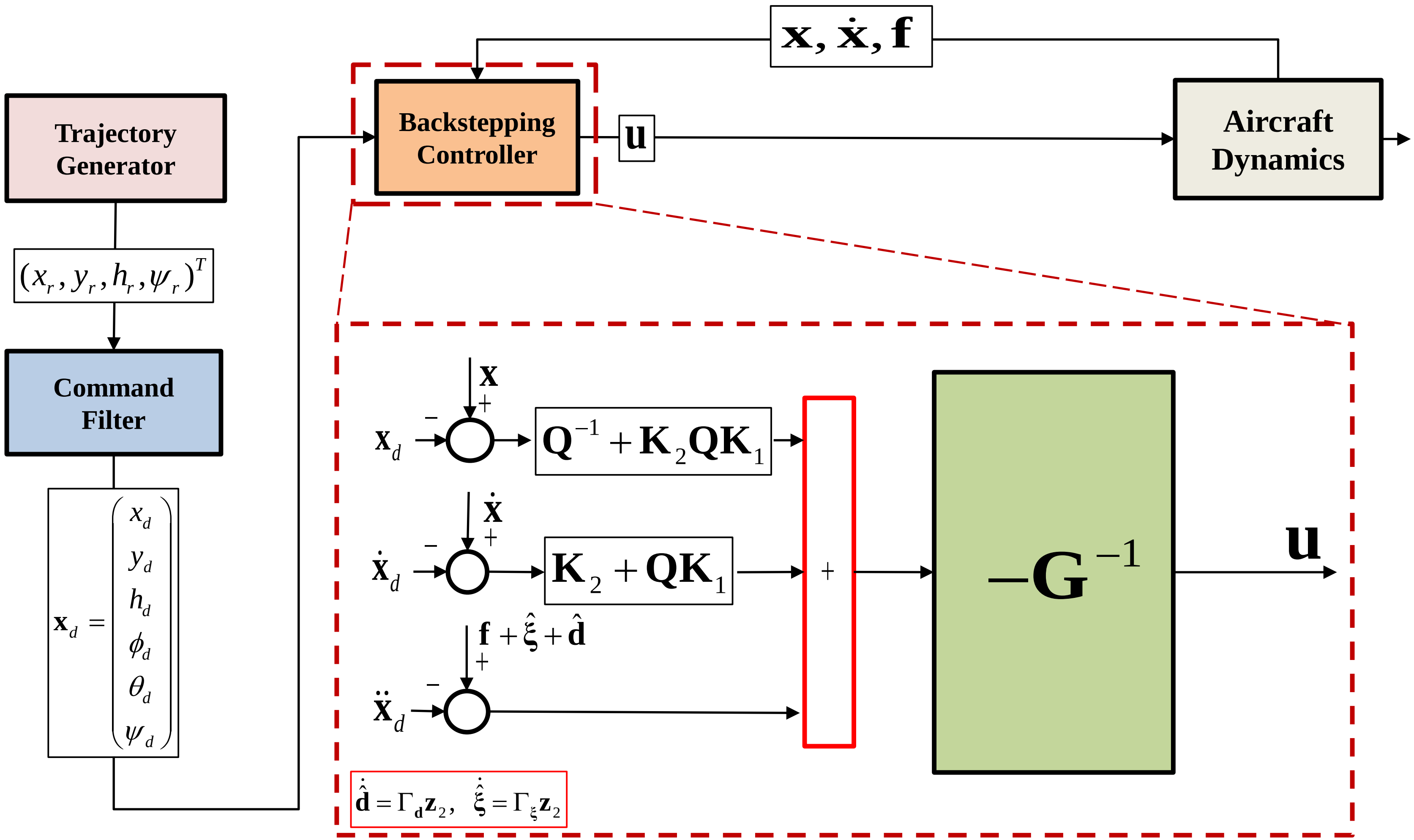
$$\zeta_j \in (0,1]$$

$$\boxed{k_{1j} = \frac{\zeta_j\omega_j}{q_j}, \quad k_{2j} = \zeta_j\omega_j, \quad q_j = \frac{1}{(1-\zeta_j^2)\omega_j^2}}$$

As a result, rigorous design works for Gain Optimization can be removed.

Design of IBS Trajectory-Tracking Controller

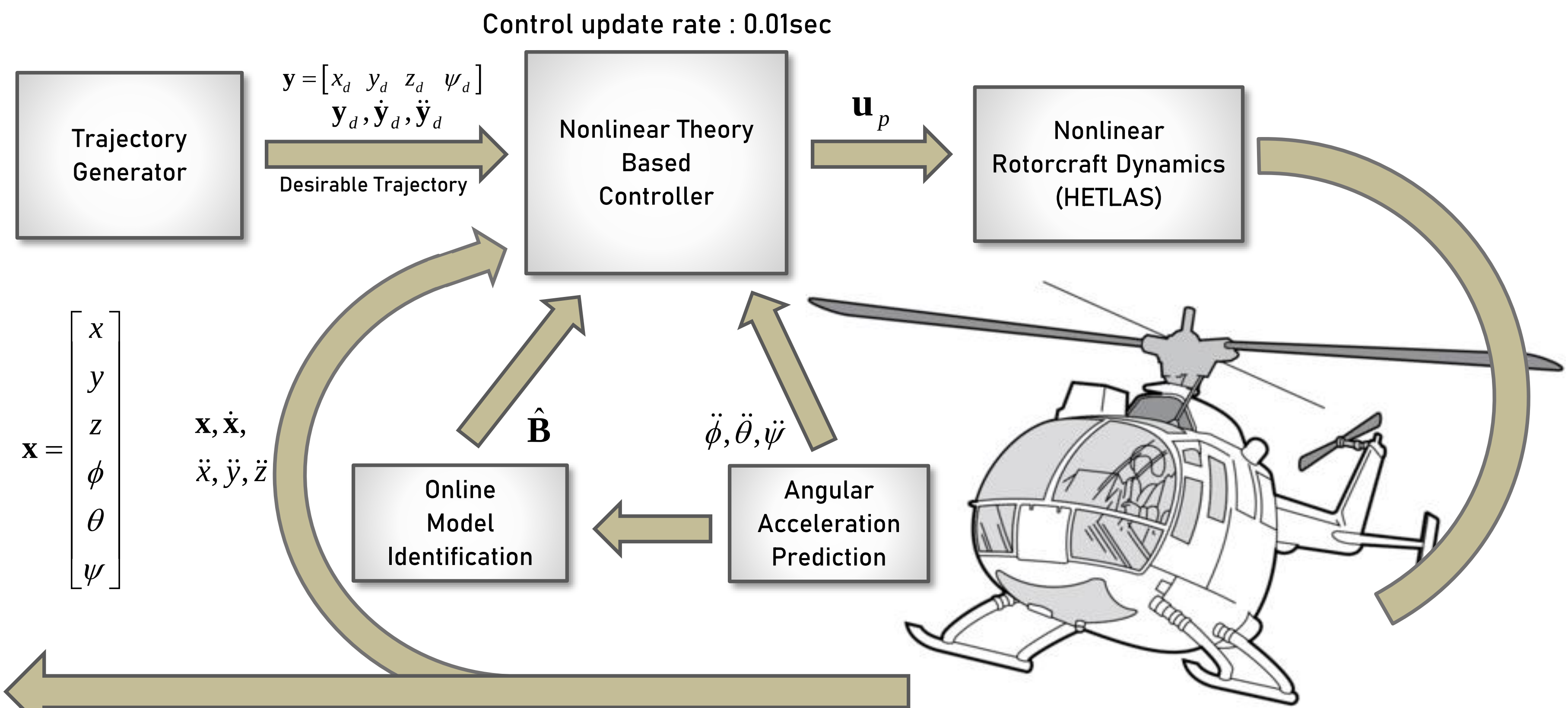
Schematics of Back-Stepping Controller with Command Filter



Development of IBS Trajectory-Tracking Control

Design of IBS Trajectory-Tracking Controller

Simulation Environment for Flight-Control-Law Validation



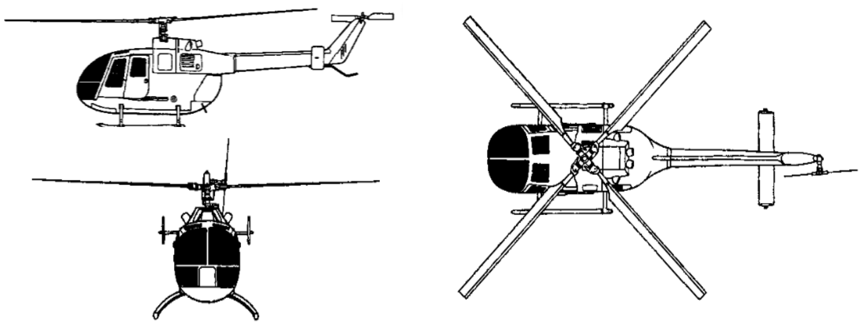
Aircraft States $\mathbf{x}, \dot{\mathbf{x}}$

*Waypoints and trajectories are assumed to be given

Simulation update rate : 0.001sec
with 4th order Runge-Kutta integrator

Validation of IBS Trajectory-Tracking Controller using Bo-105 Model

Bo-105 Helicopter



- **Model Reference**

  : Padfield, Gareth D, Helicopter flight dynamics: the theory and application of flying qualities and simulation modelling, John Wiley & Sons, 2008

Mass Properties

Helicopter Mass : 2200kg			
FIXX	1433.0 kg m^2	FIXY	0.0 kg m^2
FIYY	4973.0 kg m^2	FIYZ	0.0 kg m^2
FIZZ	4099.0 kg m^2	FIZX	660.0 kg m^2

Main Rotor Parameters

Number of Blades	4	Twist	-6.2deg
RPM	424 RPM	Lock number	5.087
Chord	0.27 m	Tilt angle	3.0deg
Radius	4.91 m	Flap hinge offset	0.746 m
lift curve slope	6.113	drag coefficient	0.0074
1st Flap moment of inertia	51.1 kg m	2nd Flap moment of inertia	231.7 kg m^2

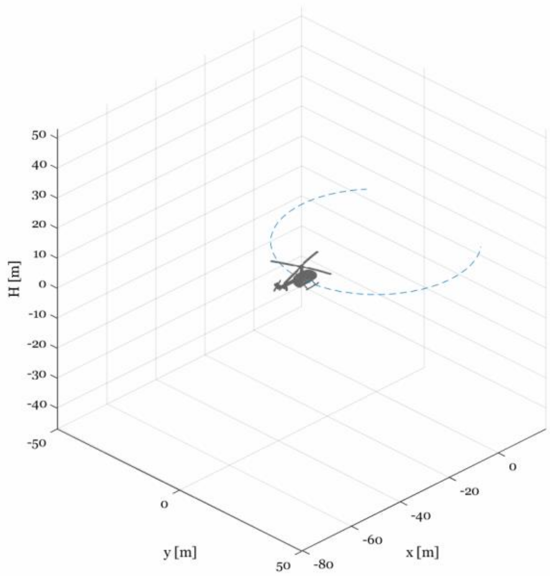
Tail Rotor Parameters

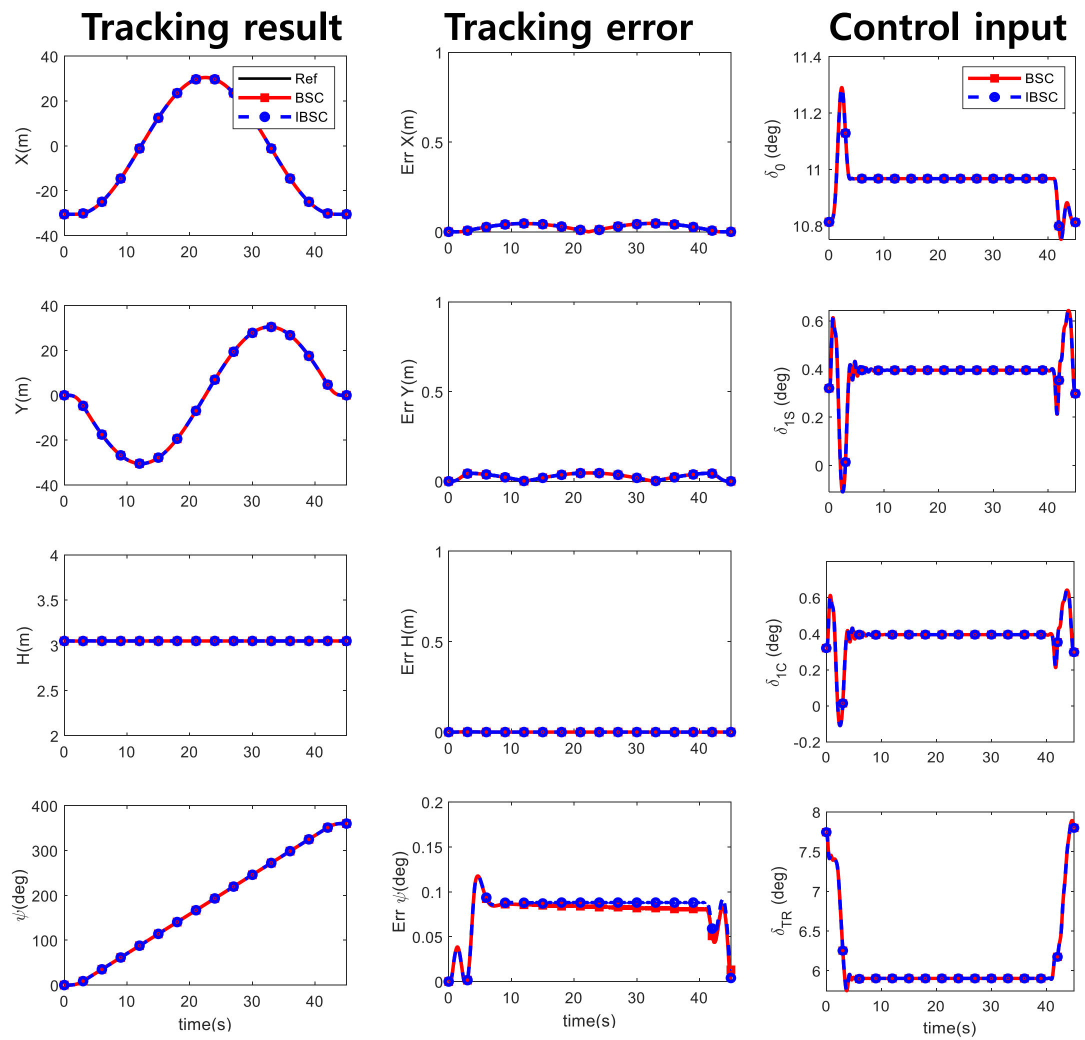
Number of Blades	2	mast height	1.72 m
RPM	233rad/sec	station	5.961m
Chord	0.179 m	Radius	0.95m
lift curve slope	4.91 m	drag coefficient	6.113
2nd Flap moment of inertia	1.06 kg m^2	1st Flap moment of inertia	0.64 kg m

Development of IBS Trajectory-Tracking Control

Validation of IBS Trajectory-Tracking Controller using Bo-105 Model

Trajectory-Tracking Control for Piroutte-Maneuver Course

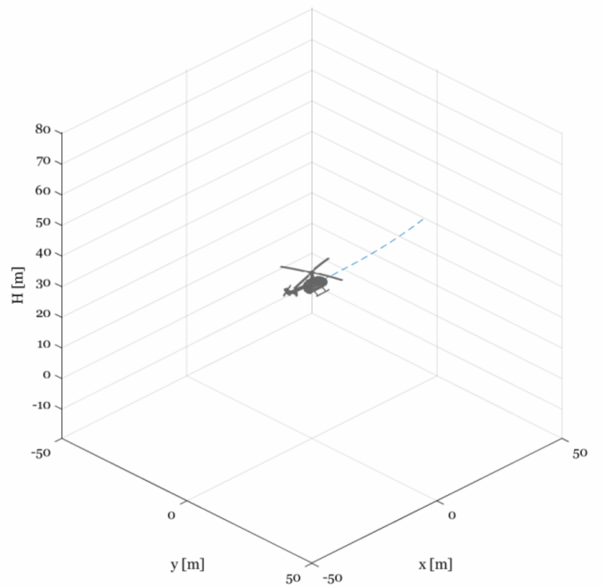


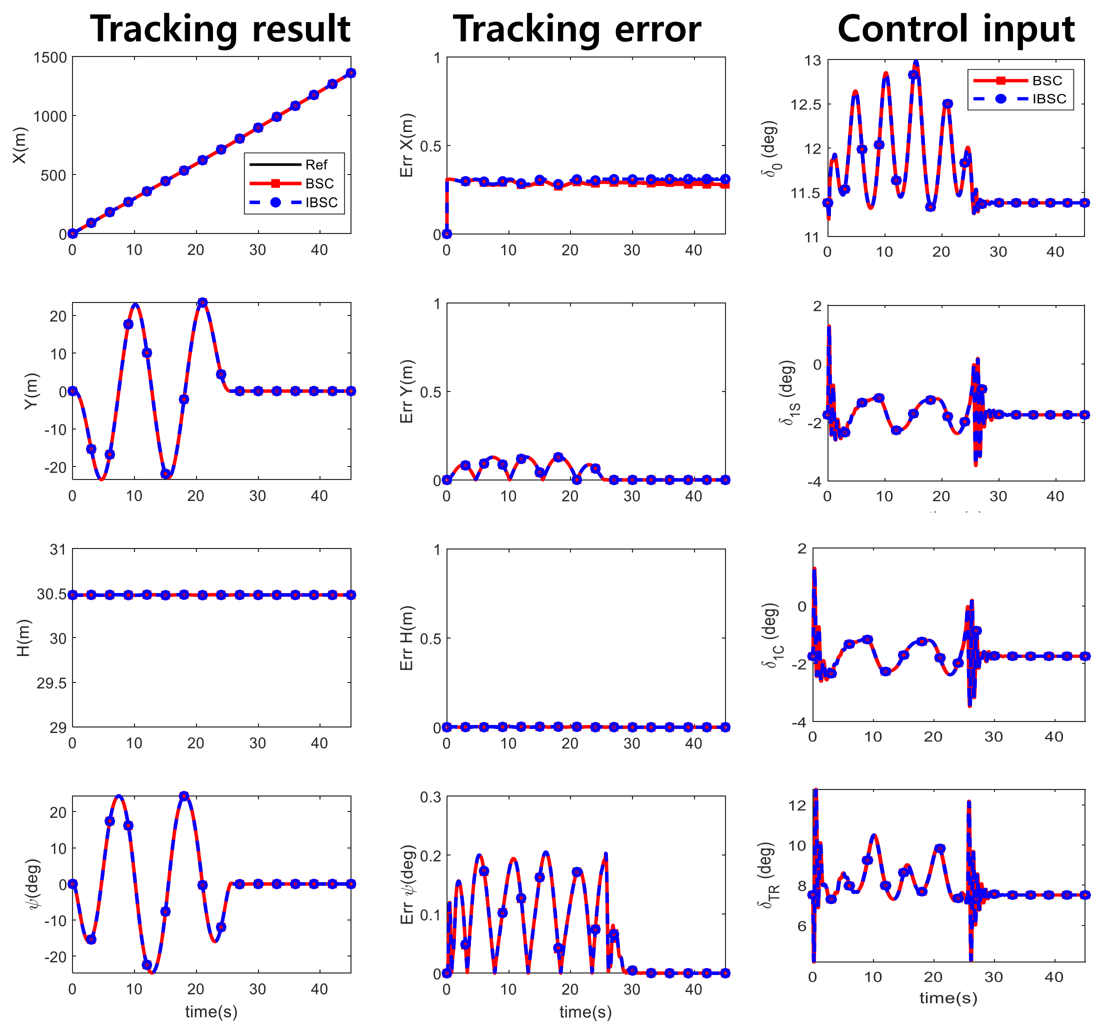


Development of IBS Trajectory-Tracking Control

Validation of IBS Trajectory-Tracking Controller using Bo-105 Model

Trajectory-Tracking Control for Slalom-Maneuver Course

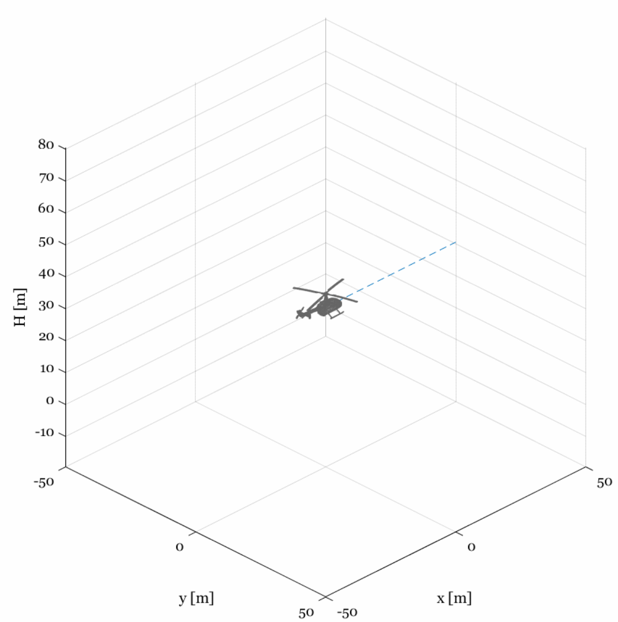




Development of IBS Trajectory-Tracking Control

Validation of IBS Trajectory-Tracking Controller using Bo-105 Model

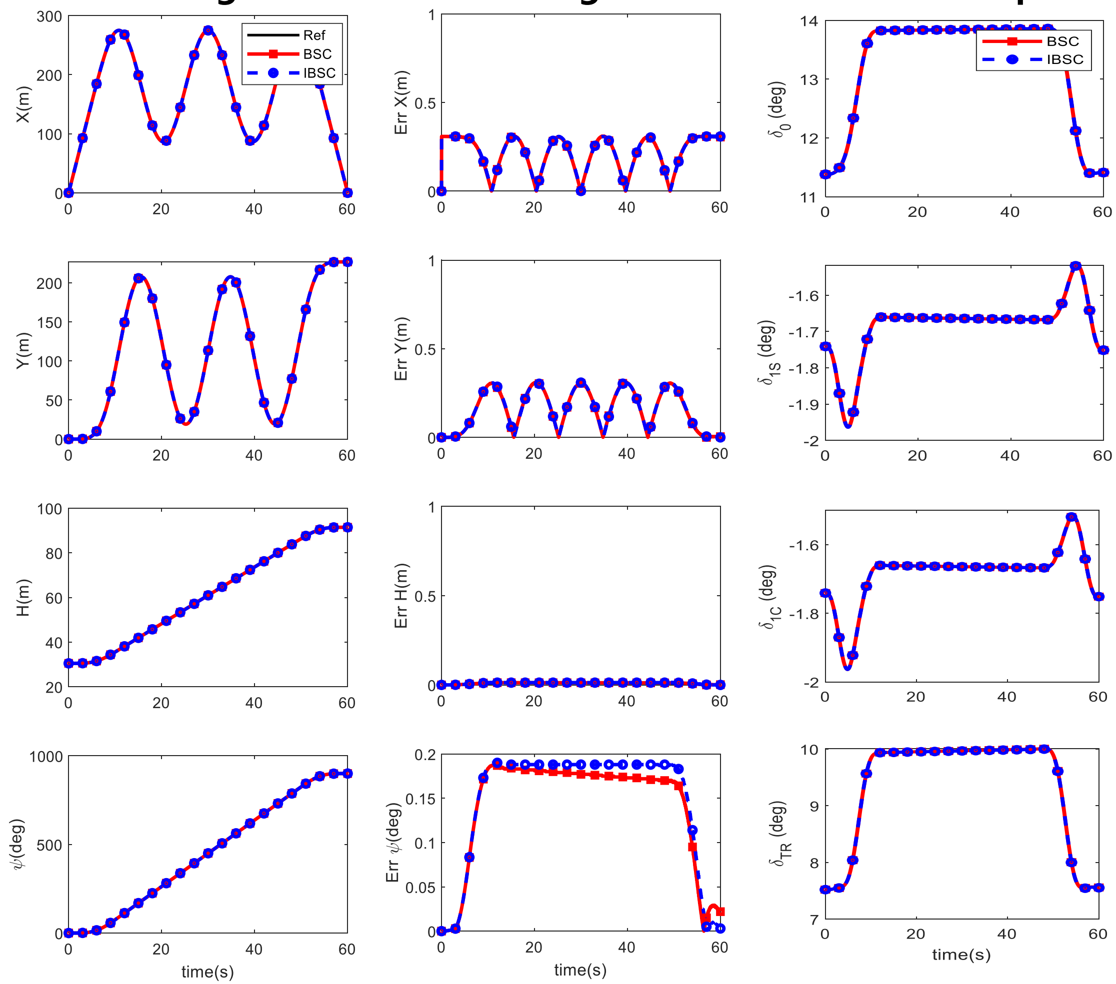
Trajectory-Tracking Control for Transient-Turn-Maneuver Course



Tracking result

Tracking error

Control input



Contents

Combined Maneuver Case

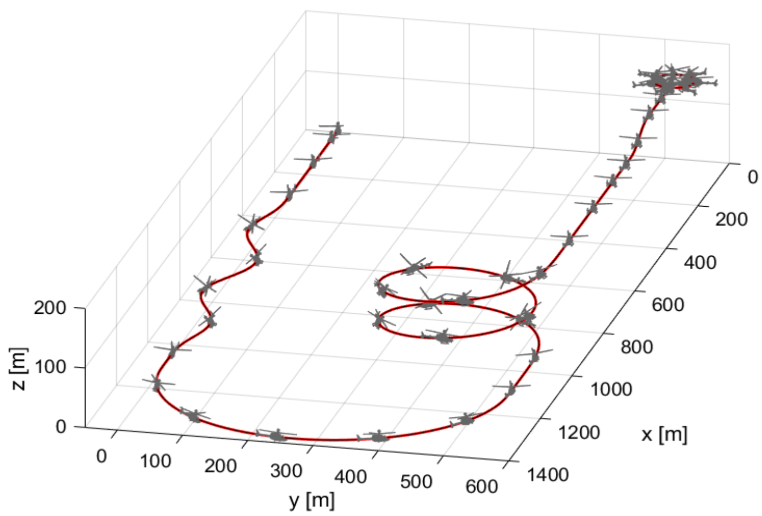
Sequence of Maneuvers

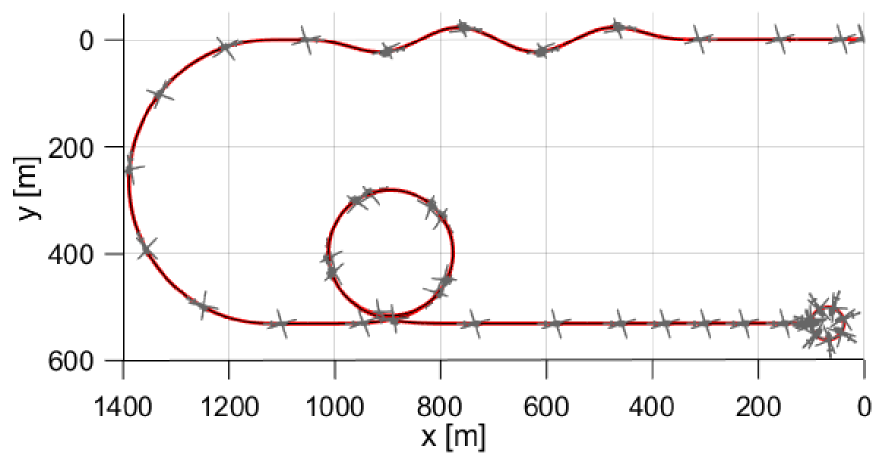
Maneuvers	Length (sec)	Velocity range (kts)	Notes
Initial Condition	0	Hover	Initial Height: 100 ft
Acceleration	0 ~ 20	0 to 60	/
Slalom	20 ~ 45	60	/
Transient Turn	45 ~ 75	60	180 deg turn
Helical Turn	75 ~ 135	60	720 deg turn
Deceleration	135 ~ 150	60 ~ 30	/
Pop up	150 ~ 160	30	100ft ascent
Deceleration	160 ~ 175	30 ~ 0	/
Pirouette	175 ~ 220	0	Radius: 100 ft

Adaptive IBSC with Least-Squares parameter estimation with direction forgetting

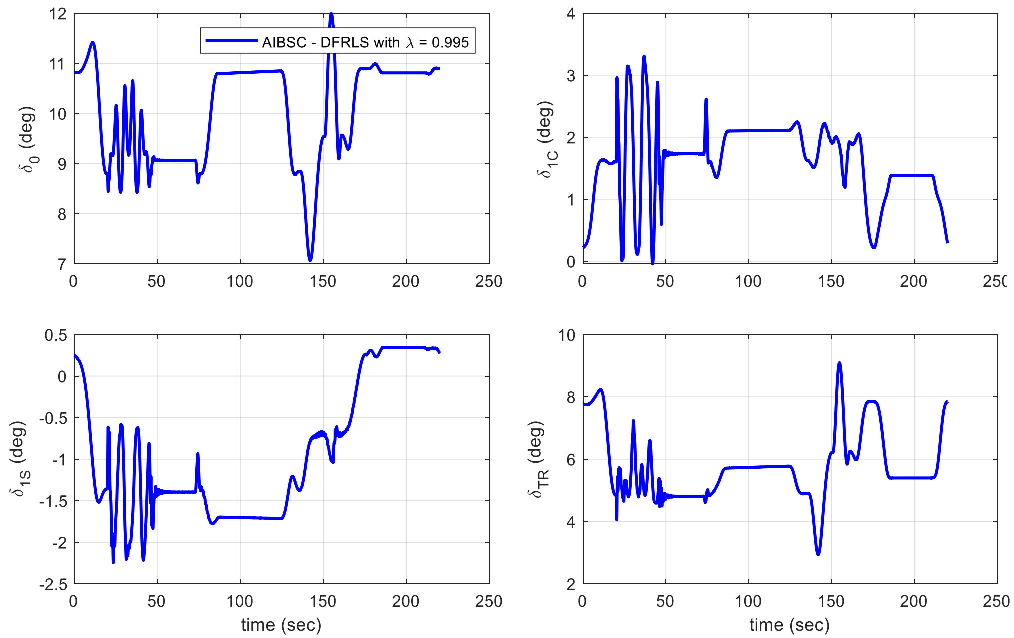
Simulation time step : 0.001sec

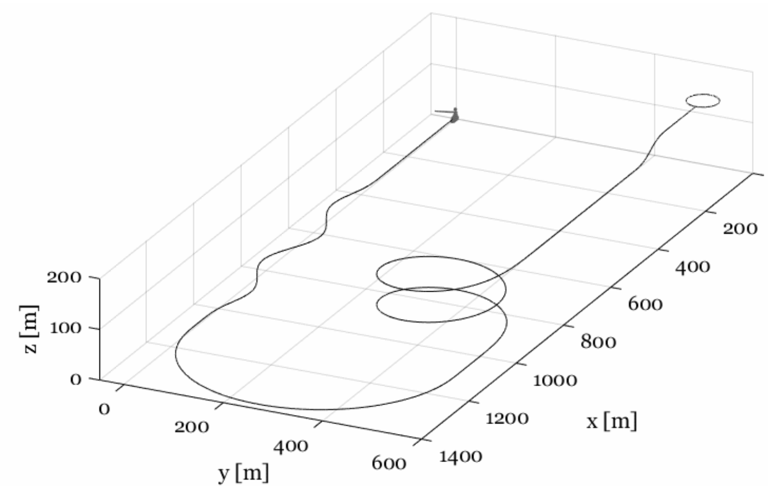
Control update rate : 0.01sec

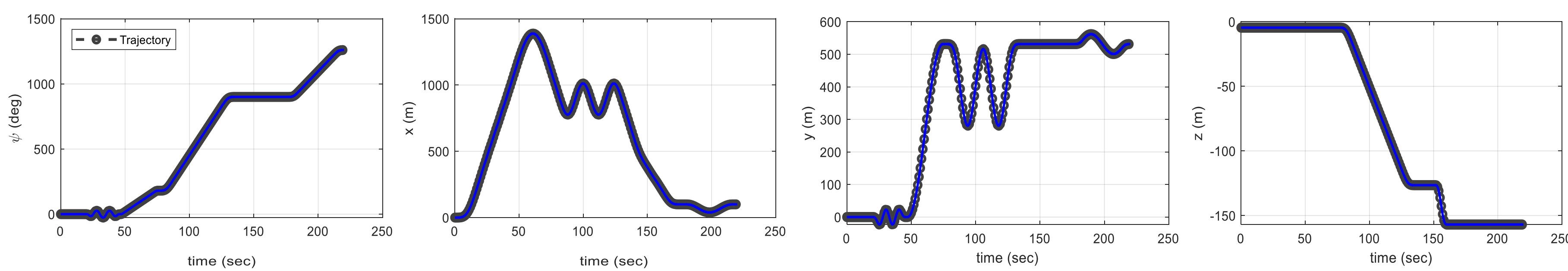




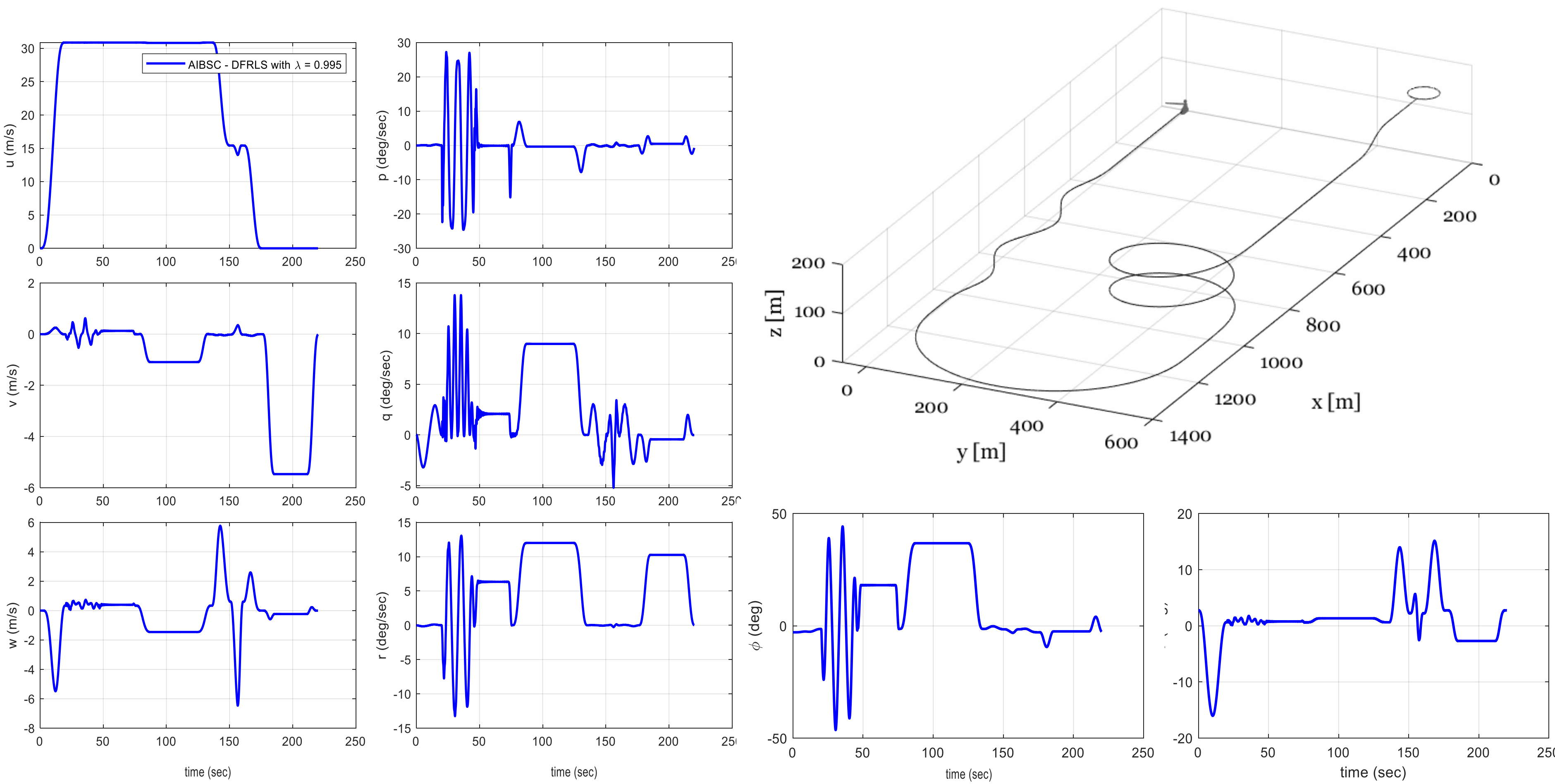
Combined Maneuver Case: Control inputs and Trajectory States





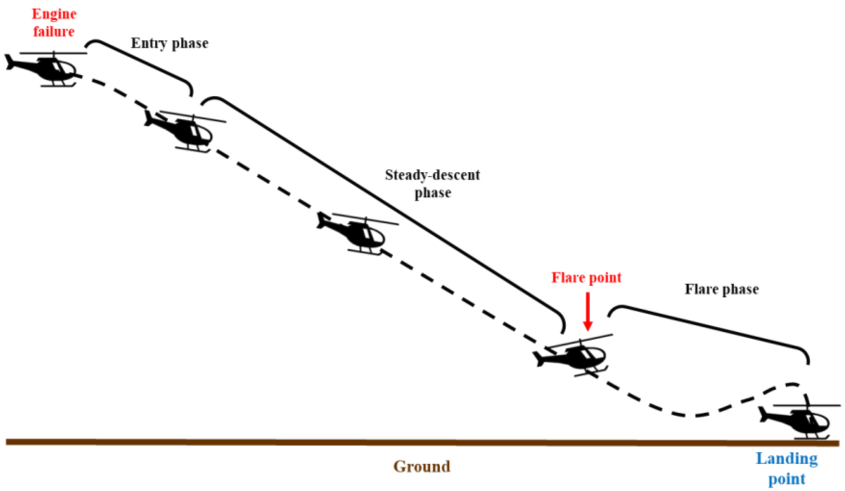


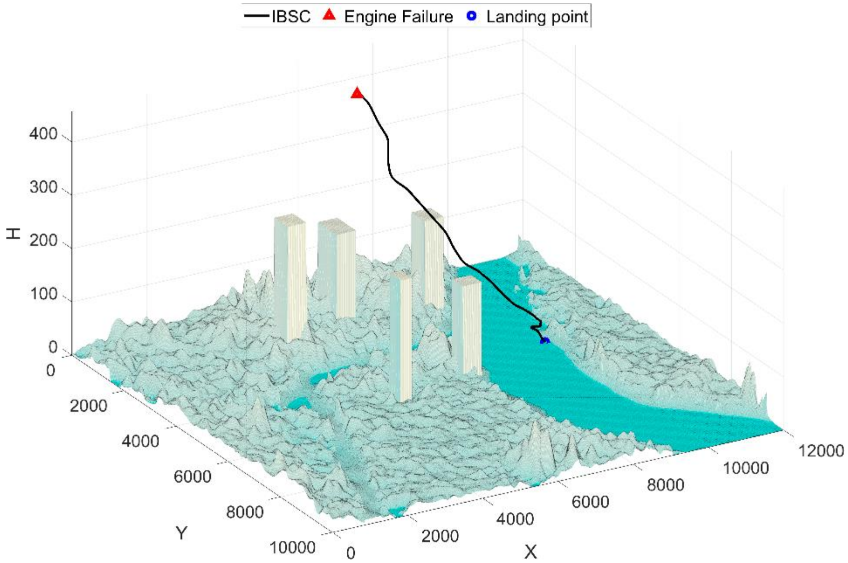
Combined Maneuver Case: Rigid-body States



Integration of Path-Plan, Trajectory Generation, and Tracking Control

Autonomous Landing after One Engine Inoperative (OEI) Condition



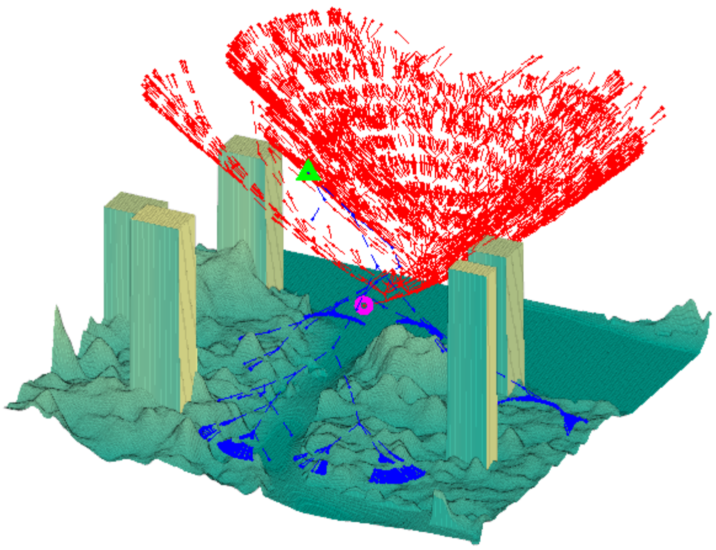


Path Planning

- Entry/Exit Phase: using NOCP solution
- Steady Decent Phase: Bi-directional RRT
  (from Entry final point to Flare initiation point)

Trajectory Generation using Spline Interpolation

Trajectory-Tracking using IBSC



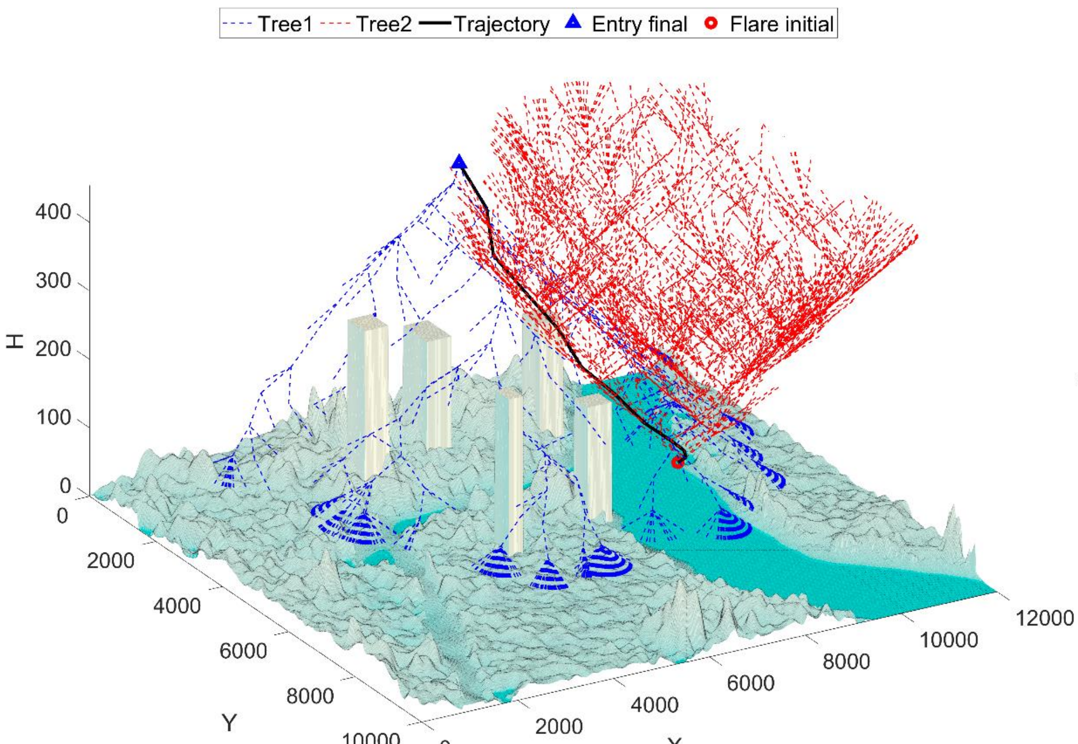
Integration of Path-Plan, Trajectory Generation, and Tracking Control

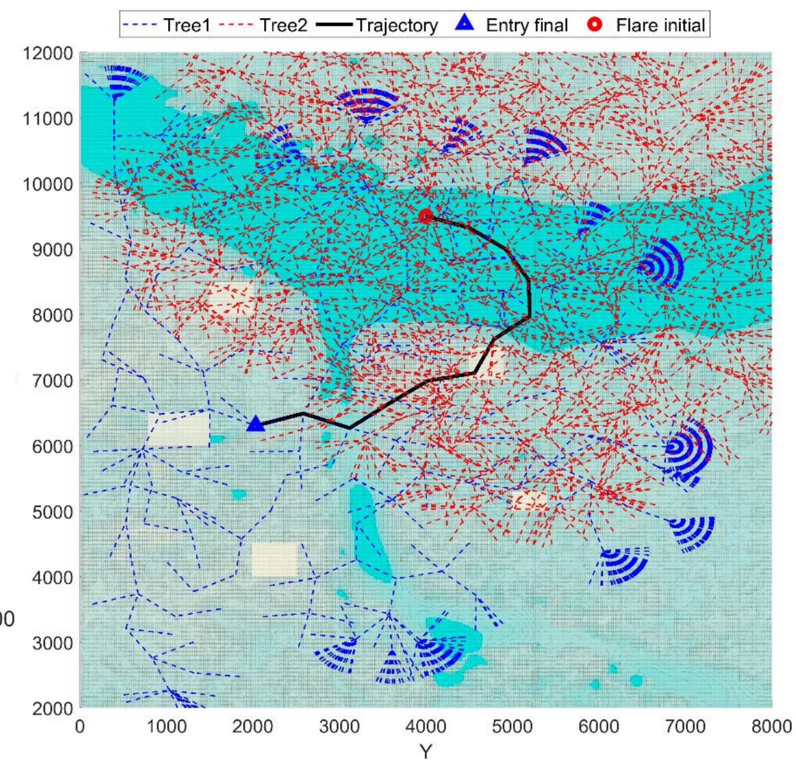
Autonomous Landing after One Engine Inoperative (OEI) Condition

Path Planning Conditions

	Conditions
Engine failure location	$x = 6200m,\ y = 2000m,\ z = 1500\,ft$
Flare initial point	$x = 9500m,\ y = 4000m,\ z = 100\,ft$
Entry trajectory	$h_{e,in} = 1500\,ft,\ \ V_{e,in} = 40knot,\ \ \dot{z}_{e,f} = 2m/s$
Flare trajectory	$h_{f,in} = 100\,ft,\ \ V_{e,in} = 40knot$
Steady-Descent trajectory	Node Generation(n) = 11

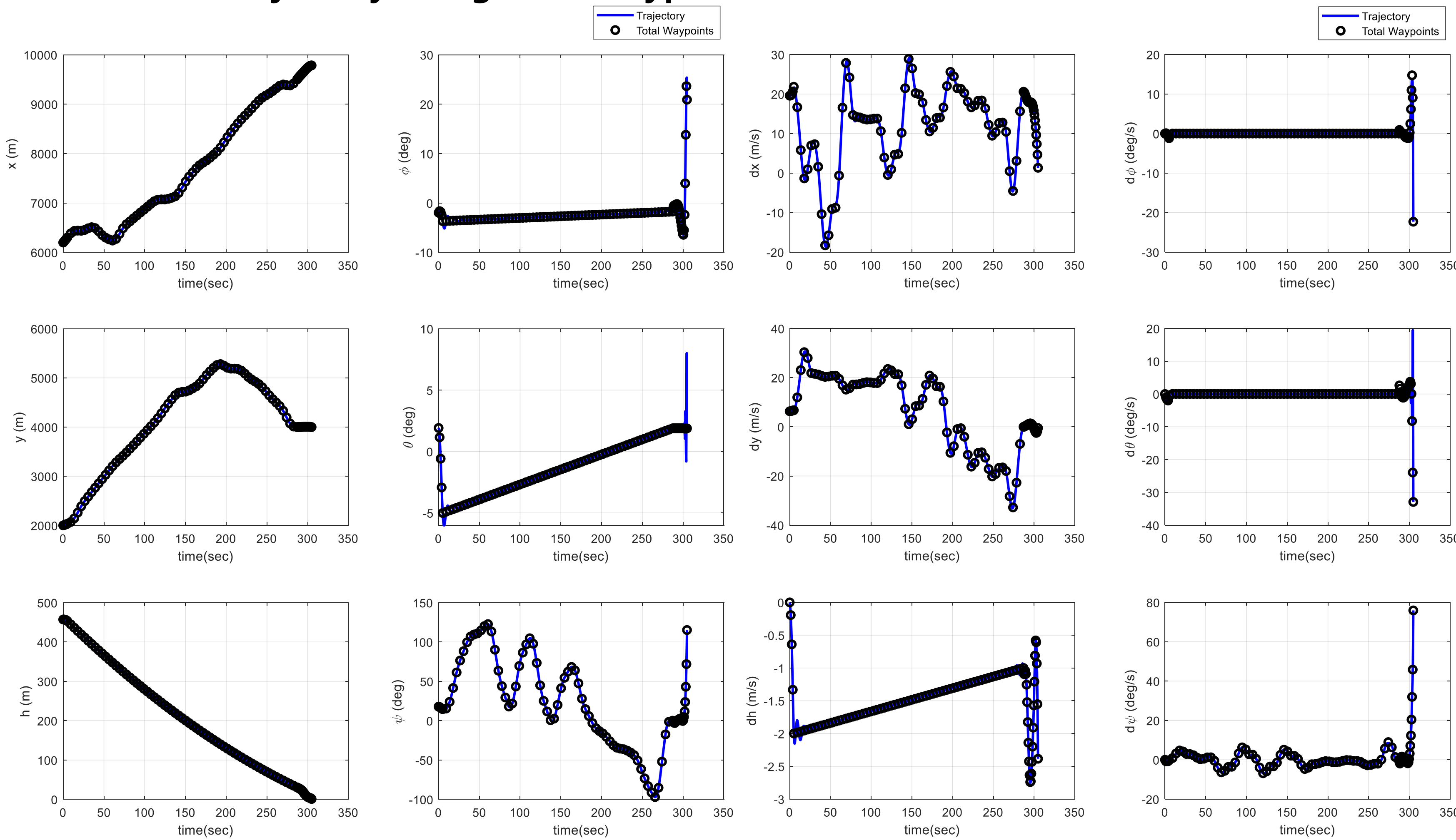
Path Planning for Steady Descent Phase : Bi-directional RRT with steady descent rate





Autonomous Landing after One Engine Inoperative (OEI) Condition

Generated Trajectory using total waypoint data

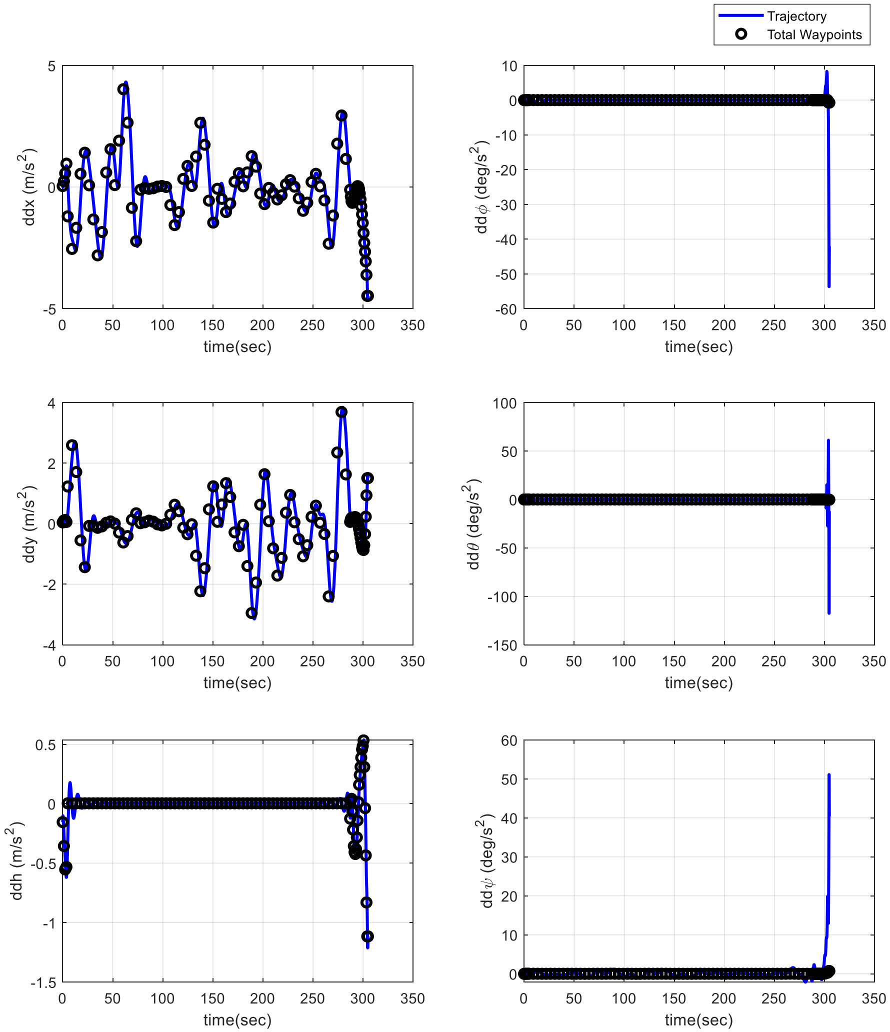


Position **Attitude** **linear velocity** **Attitude rate**

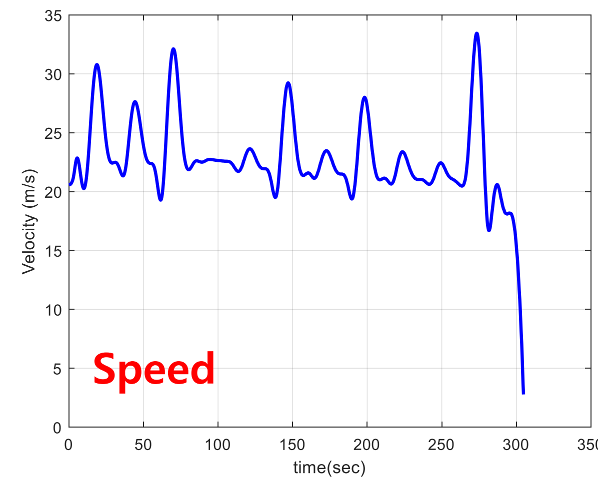
Integration of Path-Plan, Trajectory Generation, and Tracking Control

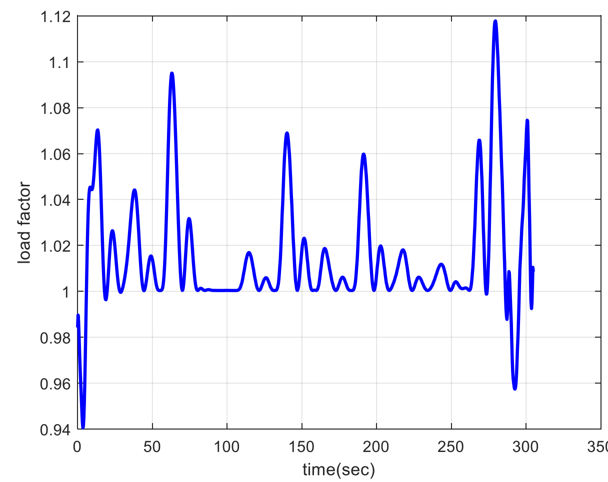
Autonomous Landing after One Engine Inoperative (OEI) Condition

Generated Trajectory using total waypoint data



Linear acceleration

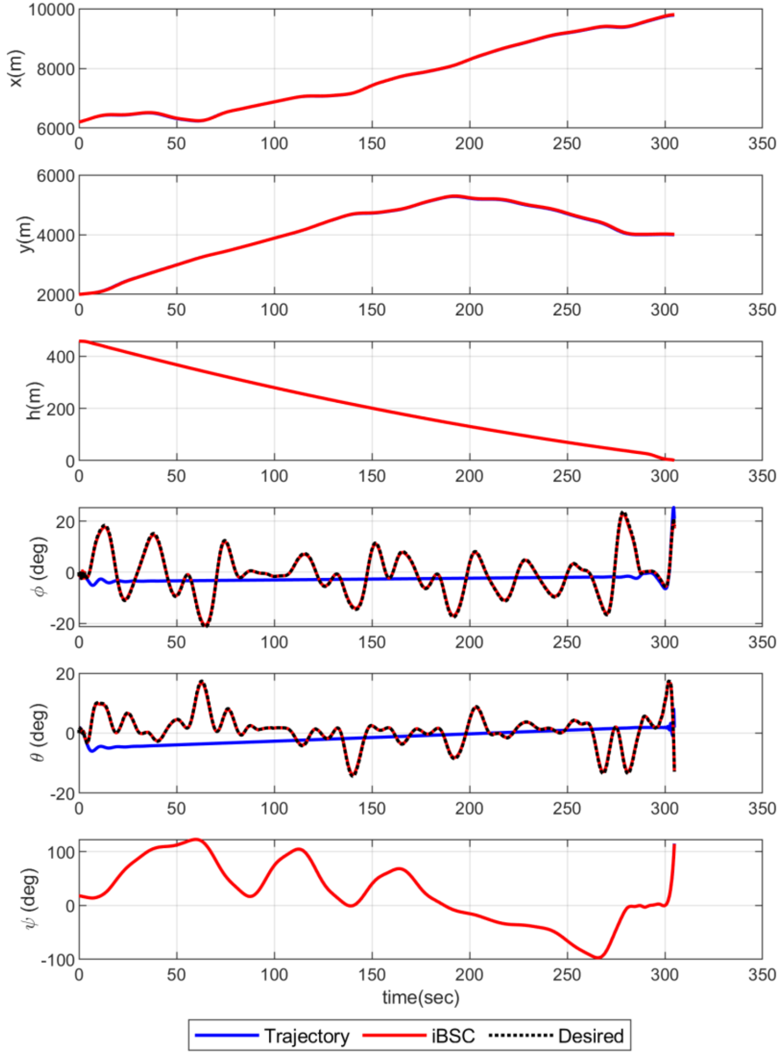




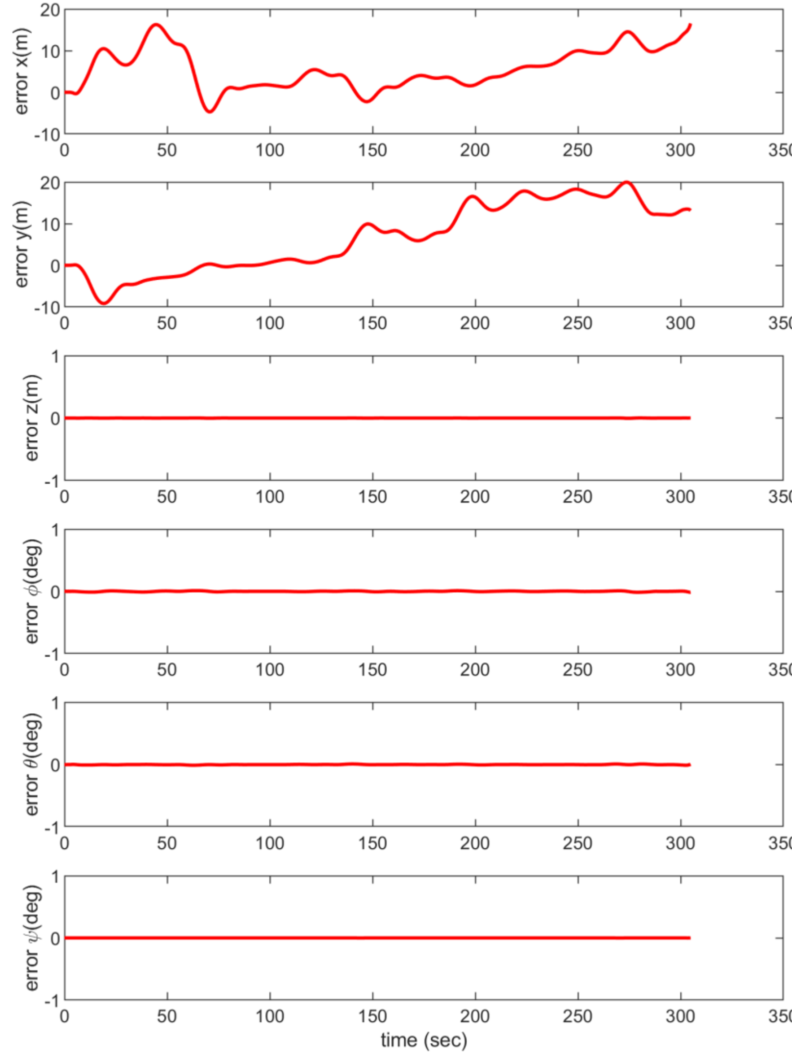
Normal Load Factor

Autonomous Landing after One Engine Inoperative (OEI) Condition

Trajectory-Tracking Control



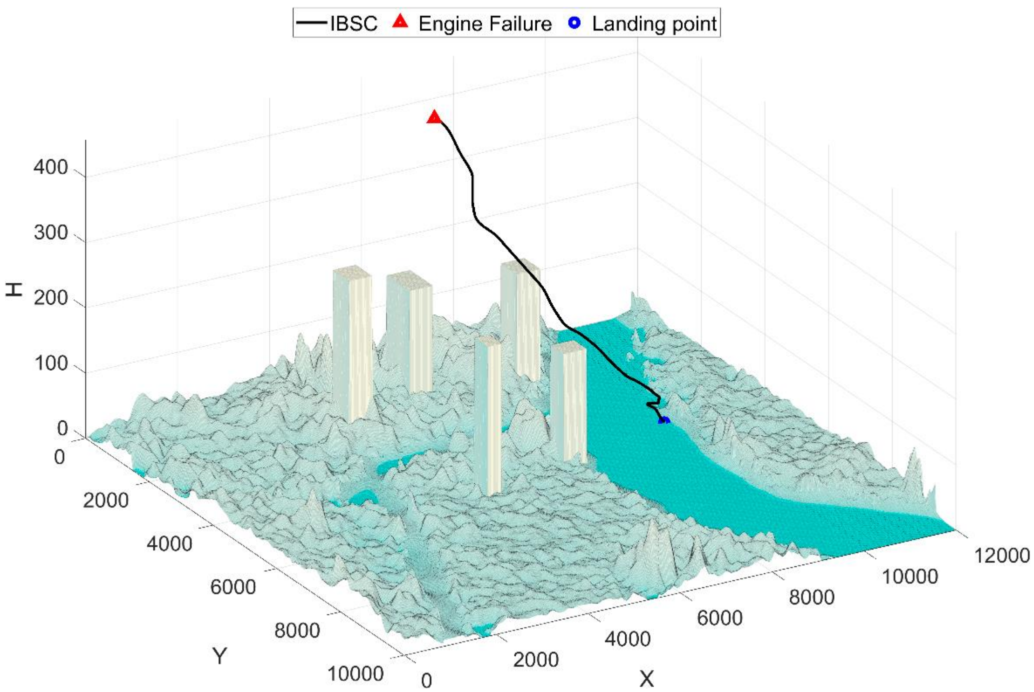
[Position and Attitude]



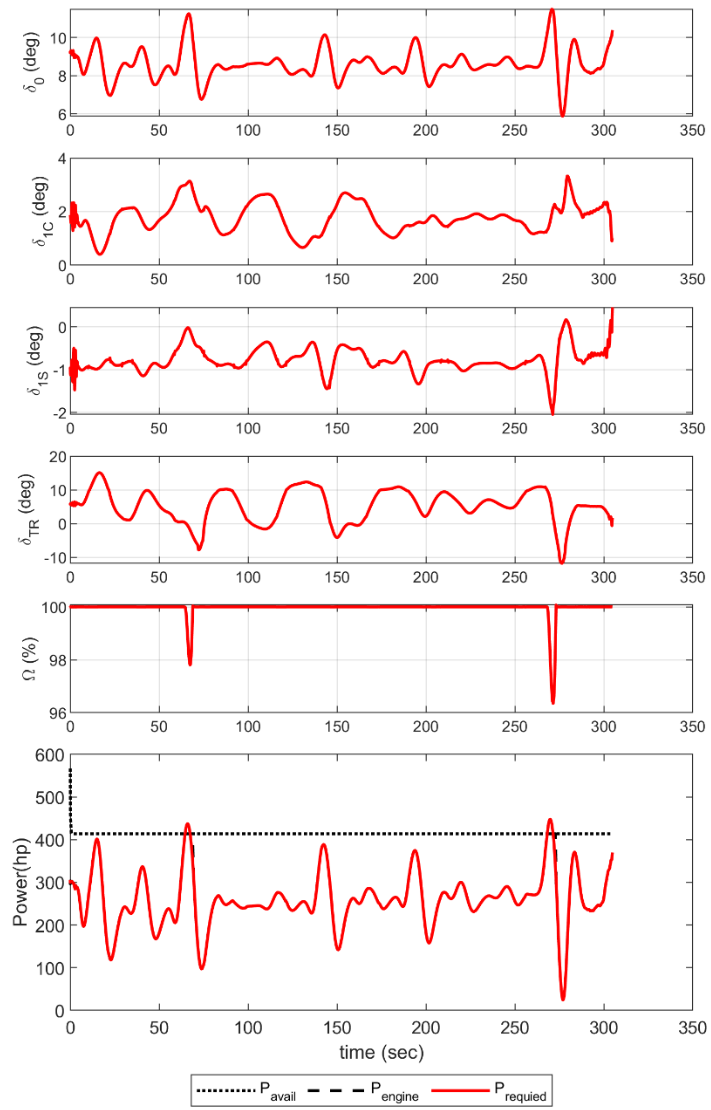
[Tracking Error]

Autonomous Landing after One Engine Inoperative (OEI) Condition

Trajectory-Tracking Control



[Tracking result with Geometric information]

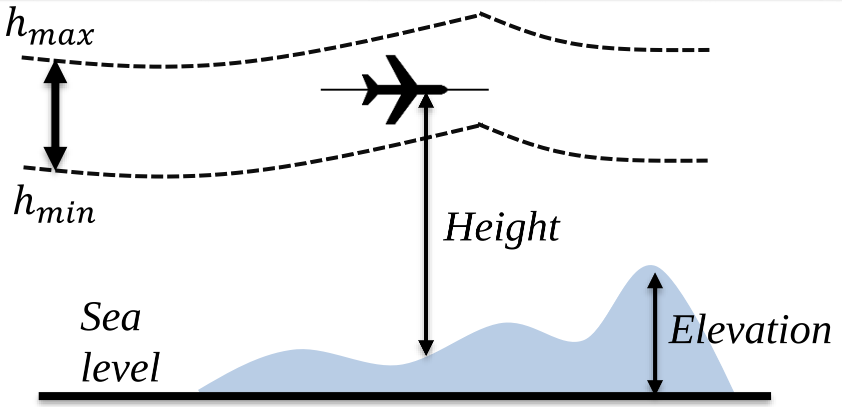


[Control, RPM, and Power]

Autonomous Terrain-Following Flight Control

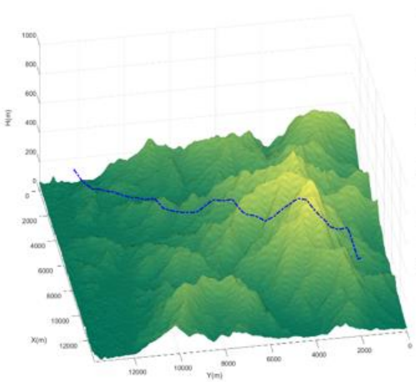
Path Planning Strategy

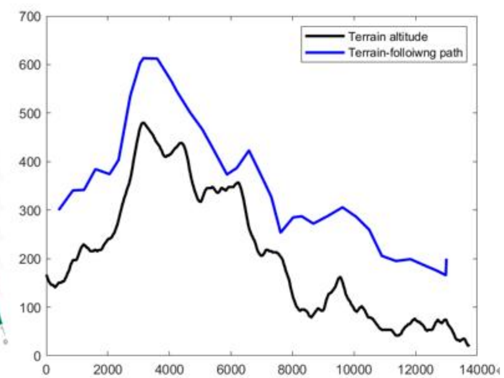
- RRT algorithm under Height clearance limits
- Real-time planning with unknown terrain information
- Re-planning when detailed terrain information becomes available
- Threat (Radar popup) cost considered



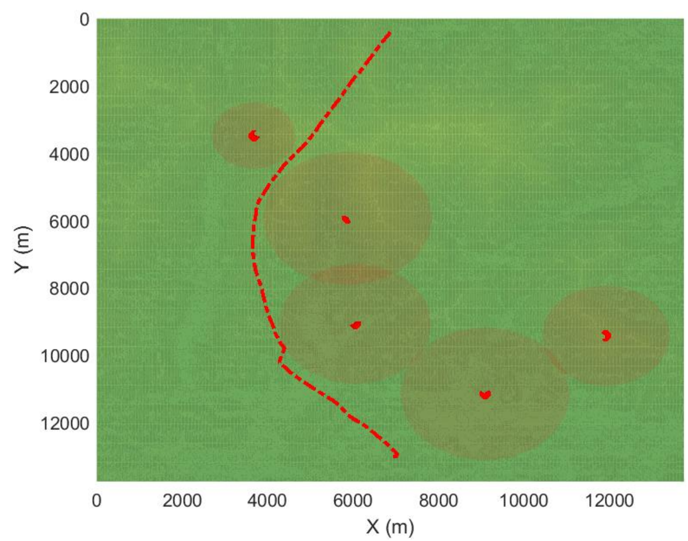
h_{max} = *Maximum clearance*

h_{min} = *Minimum clearance*





Minimum Clearance distance = 100.0m
Maximum Clearance distance = 200.0m



Autonomous Terrain-Following Flight Control

Primary Path-Planning using Low/High Resolution Terrain Information

Terrain-Following Path Planning (Unknown Terrain)

Target Point & Incomplete Terrian Information

↓

RRT Based Path Planner
(Global Planner)

↓

Path Smoothing & CLOSPO

↓

Real time Terrain Detecting
(Sensor information)

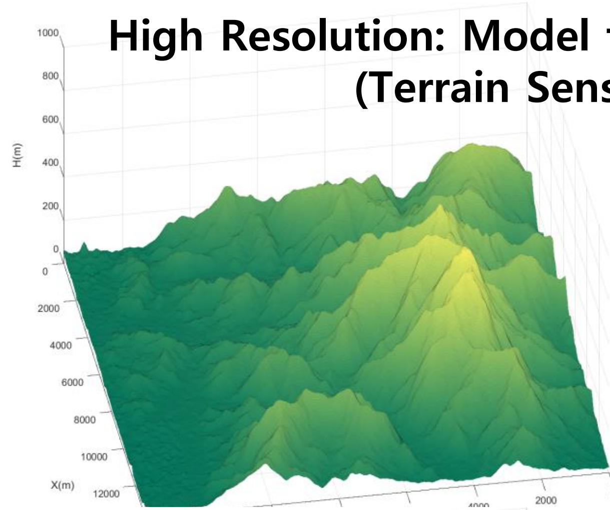
↓

Path Replanning
(Local Planner)

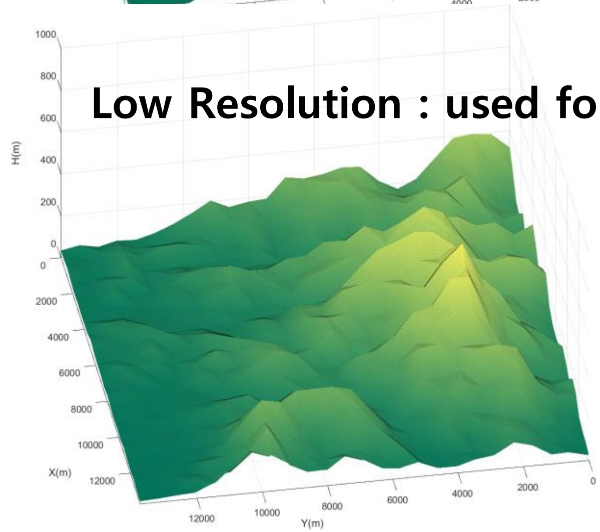
↓

Real time Terrain-Following Path

High Resolution: Model for Re-Planning (Terrain Sensor information)



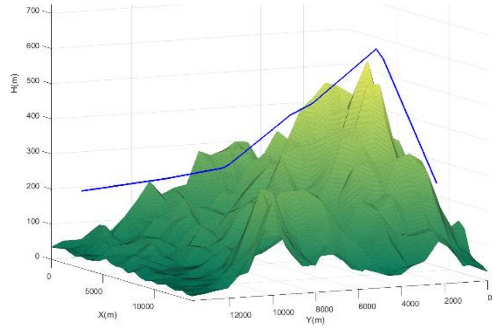
Low Resolution : used for Initial Planning

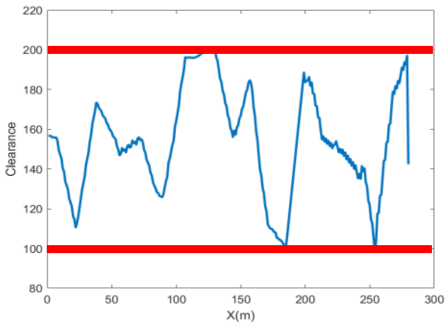


Autonomous Terrain-Following Flight Control

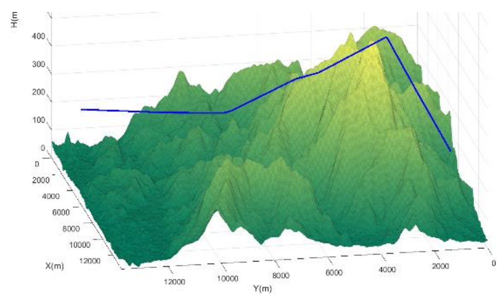
Effect of Map Resolution on Ground Clearance

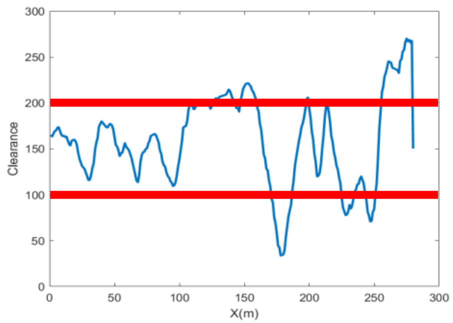
Low Resolution
Terrain Map



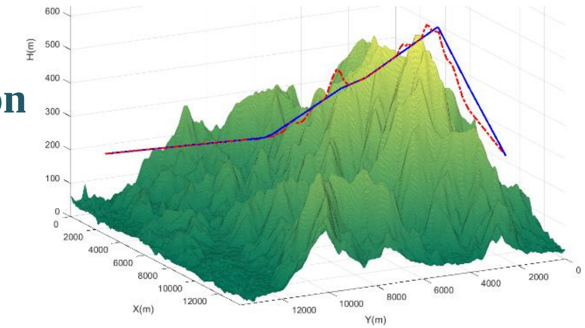


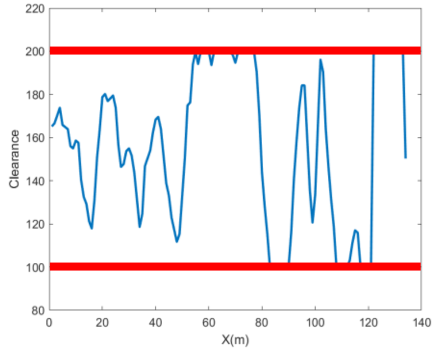
High Resolution
Terrain Map





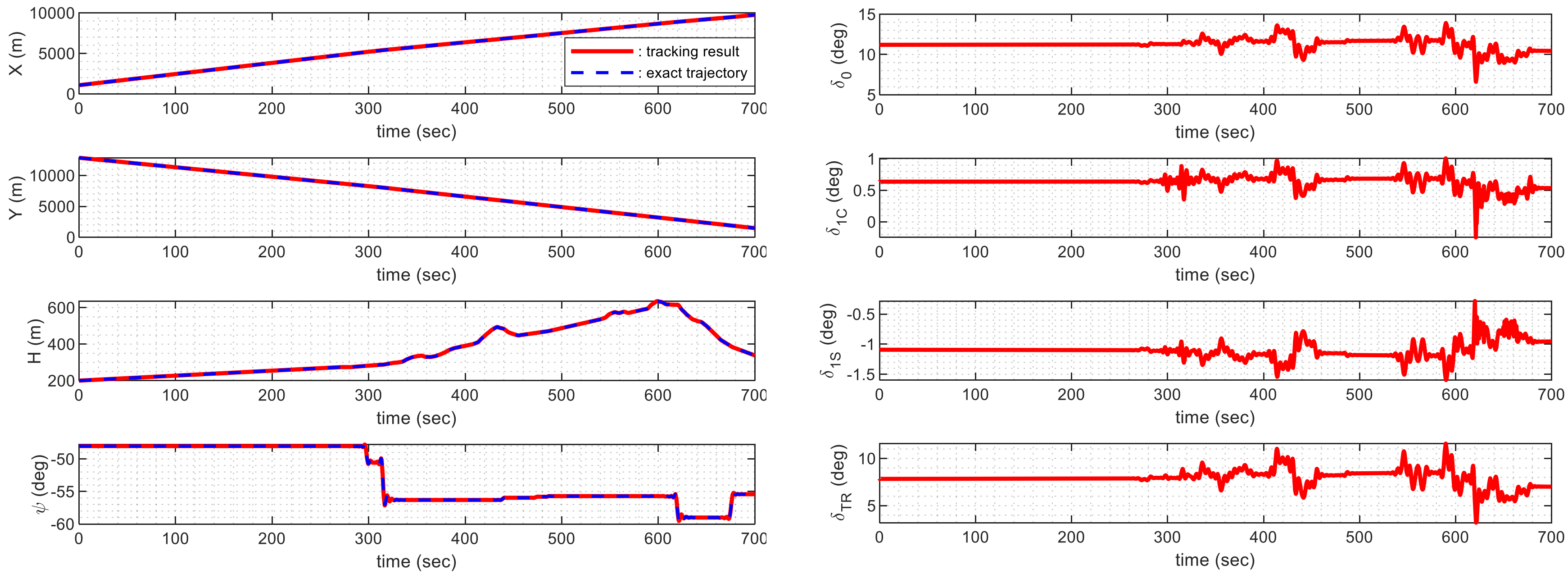
Real-Time Re-Planning with Measured Terrain Information

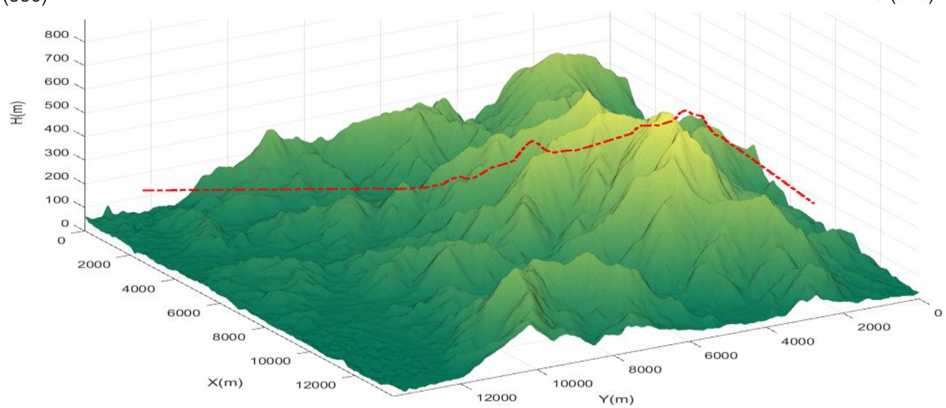




Autonomous Terrain-Following Flight Control

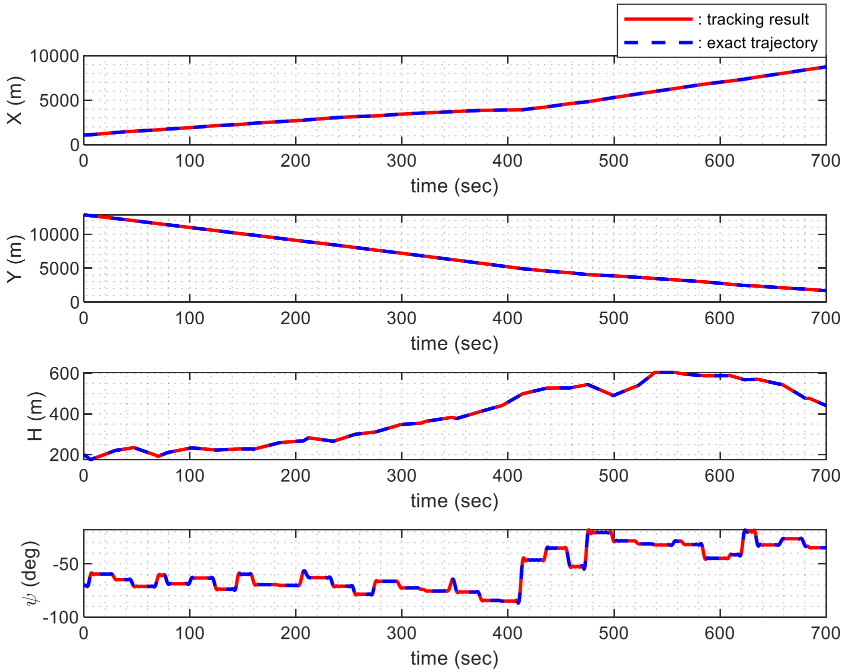
Simulation with Obstacle-free Terrain

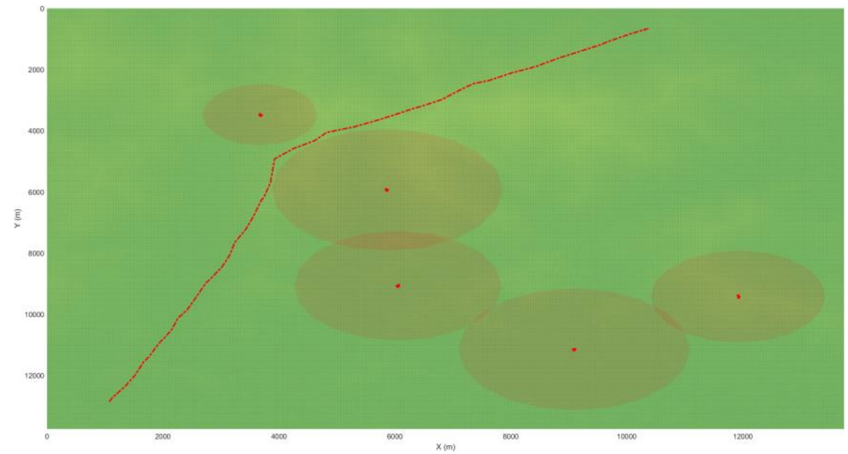


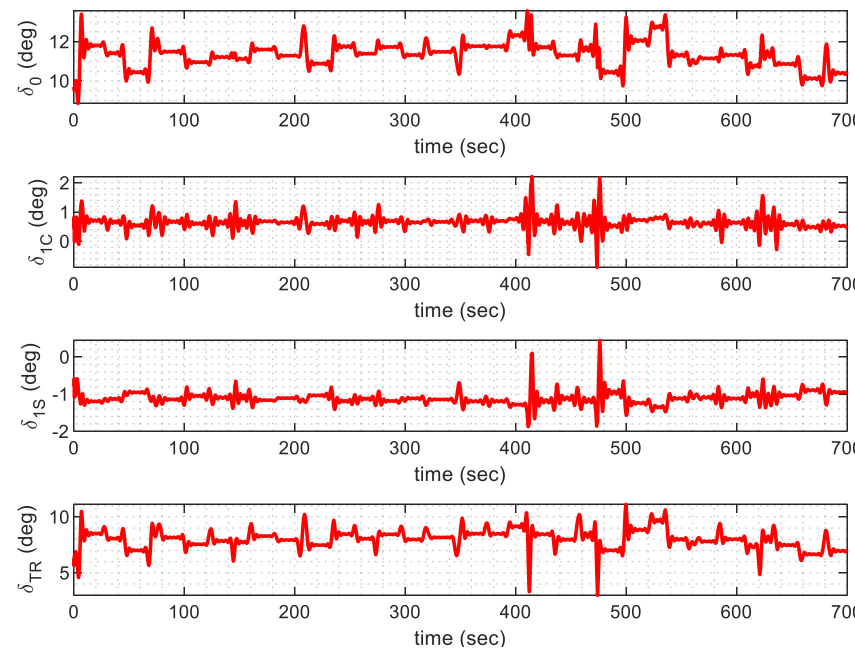


Autonomous Terrain-Following Flight Control

Simulation with Popup Radar

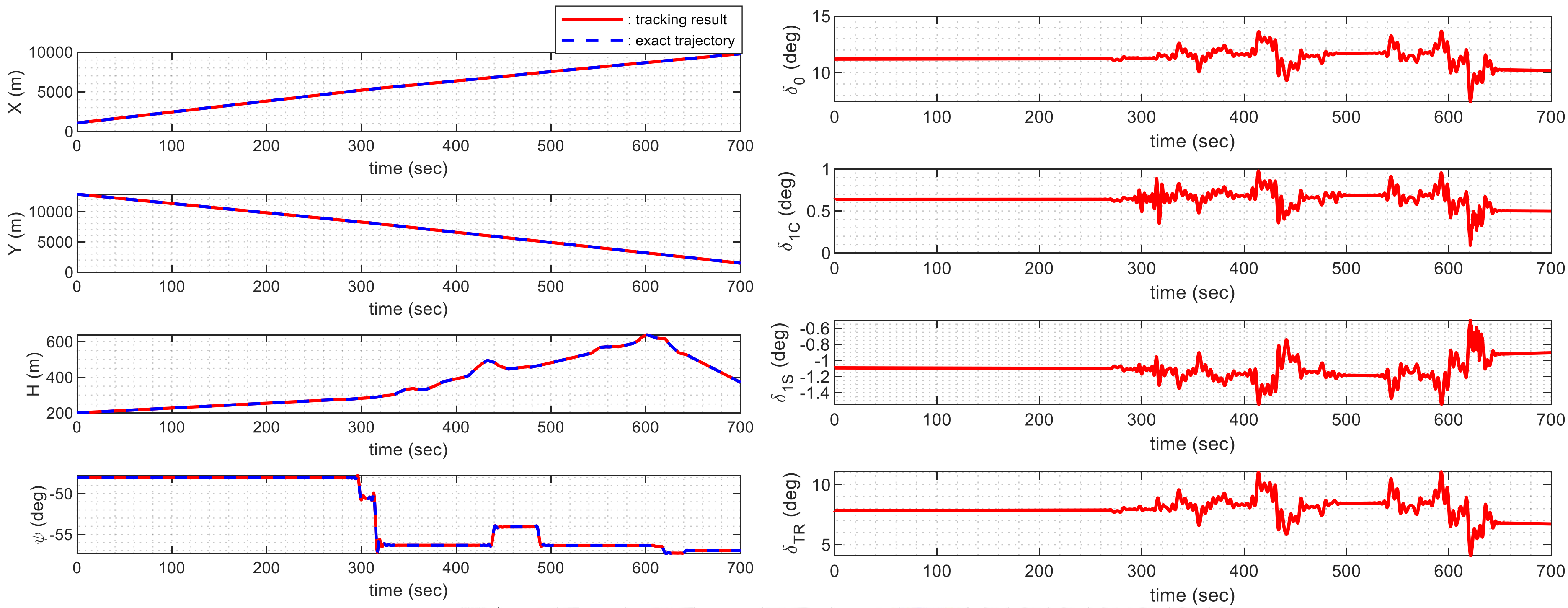


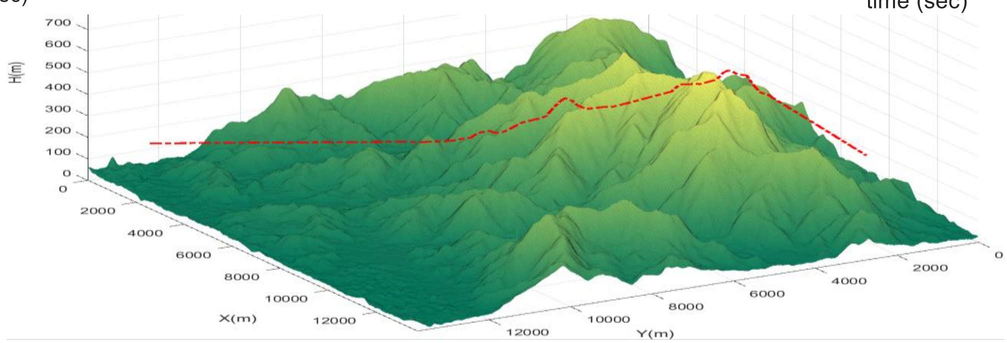




Autonomous Terrain-Following Flight Control

Simulation with Real-time Path-Planning Strategy

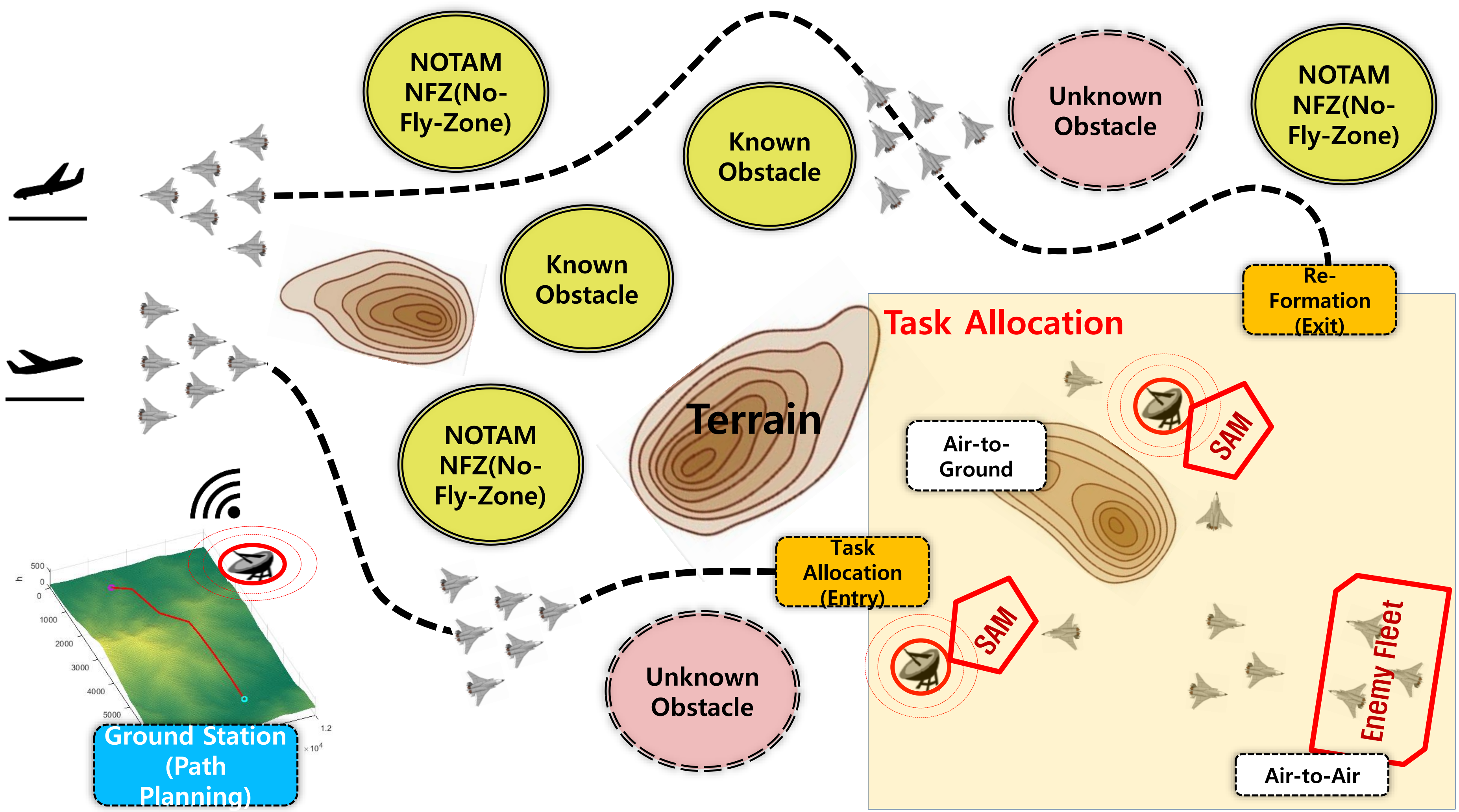




Integration of Path-Plan, Trajectory Generation, and Tracking Control

On-Going: Autonomous Multiple-RUAS Operations

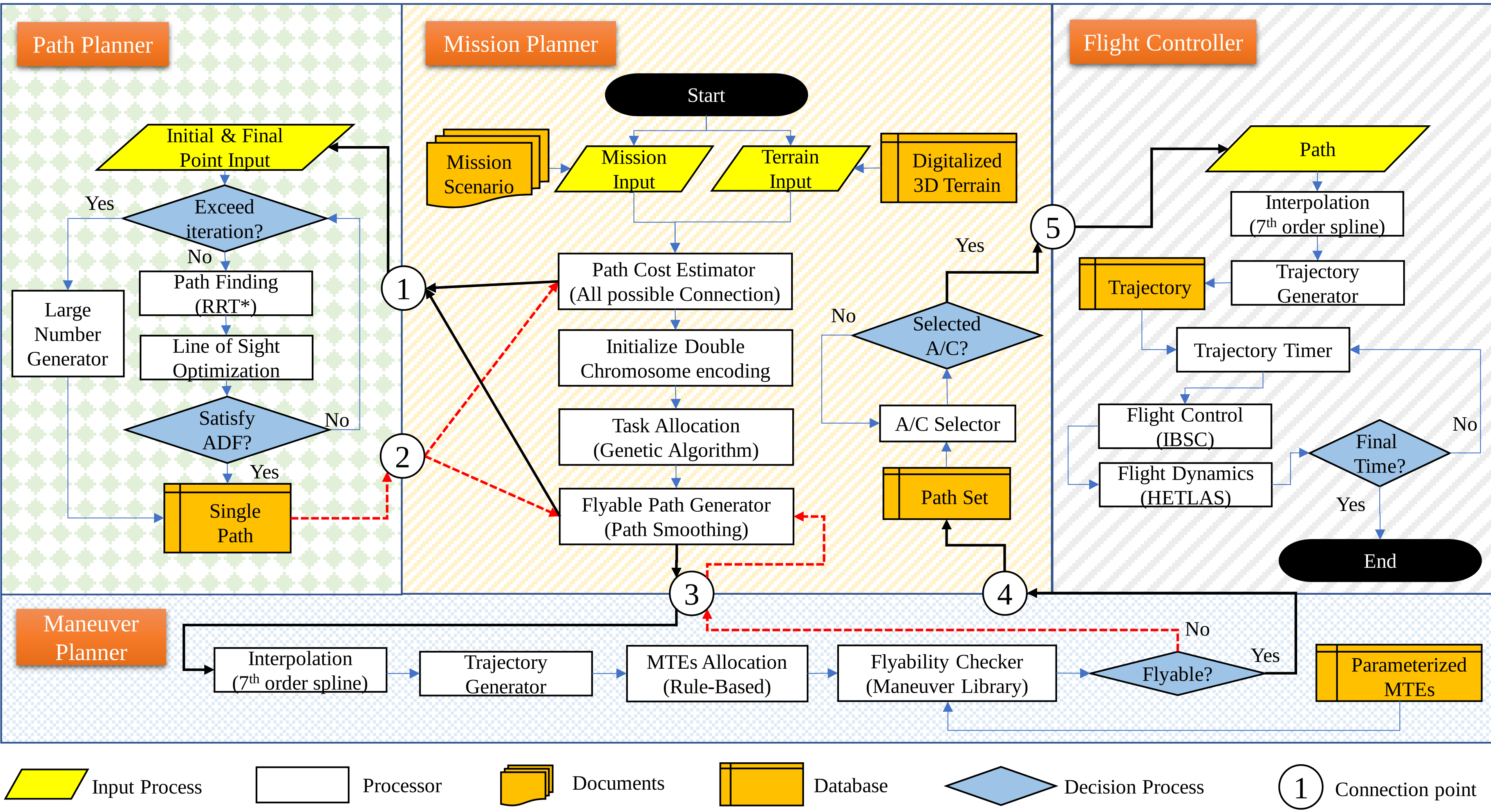
Complex and Uncertainty in Mission Environment and Scenarios for Multi-Vehicle Operation



Integration of Path-Plan, Trajectory Generation, and Tracking Control

On-Going: Autonomous Multiple-RUAS Operations

Framework for Autonomous Multi-Vehicle Operation



On-Going: Autonomous Multiple-RUAS Operations

Major Mission Planner Functions

Path-Cost Estimation

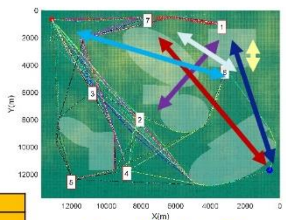
- All possible connection path between the targets are planned.
- Unconnectable case will be neglected using geometric approach
- Only cost values (ex. distance) are used for GA optimization.

No direct connection for:
Start to target 1 & 7, target 1 to 2
Target 1 to 6, Target 6 to 7 & End Point

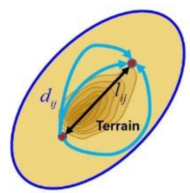
Start Point			1	2	3	4	5	6	7	End Point			
0	5	10	####	10295	13457	8493	11570	8148	####	19149	19131	19095	S.P
5	0	5	####	10284	13457	[illegible]	[illegible]	[illegible]	[illegible]	19122	19104	19116	
10	5	0	####	10280	13451	8487	11559	8151	####	19096	19093	19086	
[illegible]	[illegible]	[illegible]	0	####	12270	18407	19804	####	4349	10054	9961	9972	1
[illegible]	[illegible]	[illegible]	####	0	3326	4031	6747	8367	11023	9035	8839	8865	2
13457	13457	13451	12270	3326	0	6172	8015	11682	8023	5796	5766	5786	3
8493	8488	8487	18407	4031	6172	0	3261	9794	14104	11934	11928	11927	4
11570	11564	11559	19804	6747	8015	3261	0	12931	15660	13054	13057	13052	5
8148	8148	8151	####	8367	11682	9794	12931	0	####	[illegible]	[illegible]	[illegible]	6
####	####	####	4349	11023	8023	14104	15660	####	0	5618	5611	5617	7
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	5618	0	5	10	E.P
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	5611	5	0	5	
19095	19116	19086	9972	8865	5786	11927	13052	####	5617	10	5	0	

Start Point
Between Target
End Point
Cost of connection from point 3 to point 2

Estimated Path Cost Between the Targets



Actual Path in 3D



$d_{ij} < k_{dist} l_{ij}$

Task Allocation

- GA based algorithms are known for their robustness, fastness and others

Table 1 Comparison of common dynamic task allocation algorithms.

Algorithm	Dynamism	Robustness	Fault tolerance	Accuracy	Applicable scale	Speed
Auction algorithm	St	St	St	M	L	F
CNP	St	St	St	M	M	F
GA	St	St	St	M	M	F
PSO	St	M	M	M	L	F
ACO	St	St	St	M	L	F
Clustering algorithm	St	P	M	P	M	F

Note: St: strong; M: moderate; P: poor; L: large; F: fast.

Q. Peng, H. Wu, and R. Xue, "Review of Dynamic Task Allocation Methods for UAV Swarms Oriented to Ground Targets," Complex System Modeling and Simulation, vol. 1, no. 3, pp. 163–175, 2021, doi: 10.23919/csms.2021.0022.

- Double-Chromosome Encoding Methods are applied for task allocation

Example 1
- RUAV 3
- Chromosome I (3,1,4,2,5)
- Chromosome II (2,3)

Task Order	3	1	4	2	5
No. UAV	RUAV1		RUAV2	RUAV3	

2 3

Example 2
- 3 RUAV
- Chromosome I (3,1,4,2,5)
- Chromosome II (4,4)
- Note. No tasks are assigned to RUAV2

Task Order	3	1	4	2	5
No. UAV	RUAV1				RUAV3

4 4

Cross Over

1	3	5	2	4	6	→	4	3	5	2	1	6
2	5	4	1	6	3	→	3	5	4	1	6	2

Mutation

- **Holding** selected part

1	3	5	2	4	6	→	6	3	5	2	1	4

- **Flip** entire selected part

1	3	5	2	4	6	→	1	4	2	5	3	6

- **Exchange** selected two elements

1	3	5	2	4	6	→	1	4	5	2	3	6

- **Sliding** elements in selected part

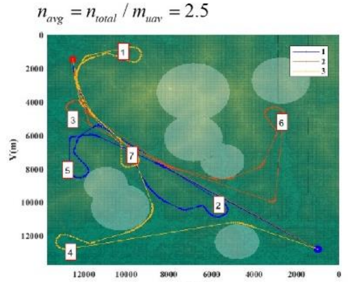
1	3	5	2	4	6	→	1	5	2	4	3	6

Task Assignment/Trajectory Generation

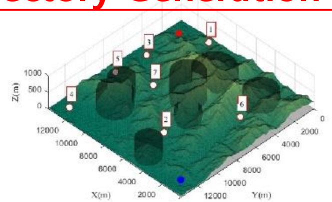
- Task allocation for: 1) **Minimize the total distance**
  2) **Evenly distributed targets**

$$\min J = \underbrace{\sum_{i=1}^{3} d_i}_{\text{total distance}} + (10^8)\underbrace{\sum_{i=1}^{3}(n_i - n_{avg})^2}_{\text{evenly distributed tasks}}$$

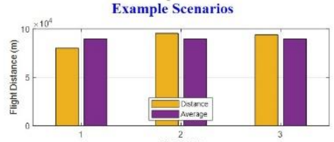
$$n_{avg} = n_{total} / m_{uav} = 2.5$$



Generated Trajectory



Example Scenarios



RUAV ID	Number of Task	Task Sequence
1	2	5-2
2	2	3-6
3	3	4-1-7

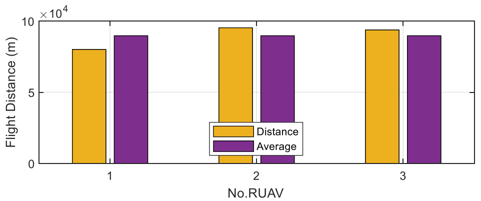
Assigned Tasks

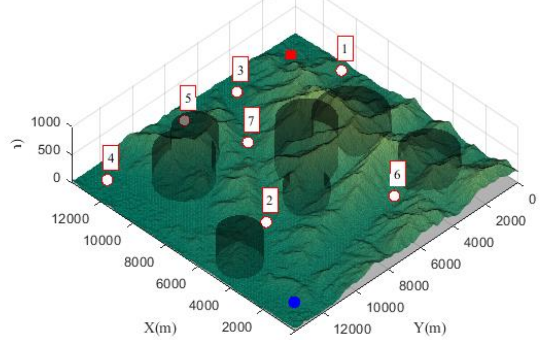
On-Going: Autonomous Multiple-RUAS Operations

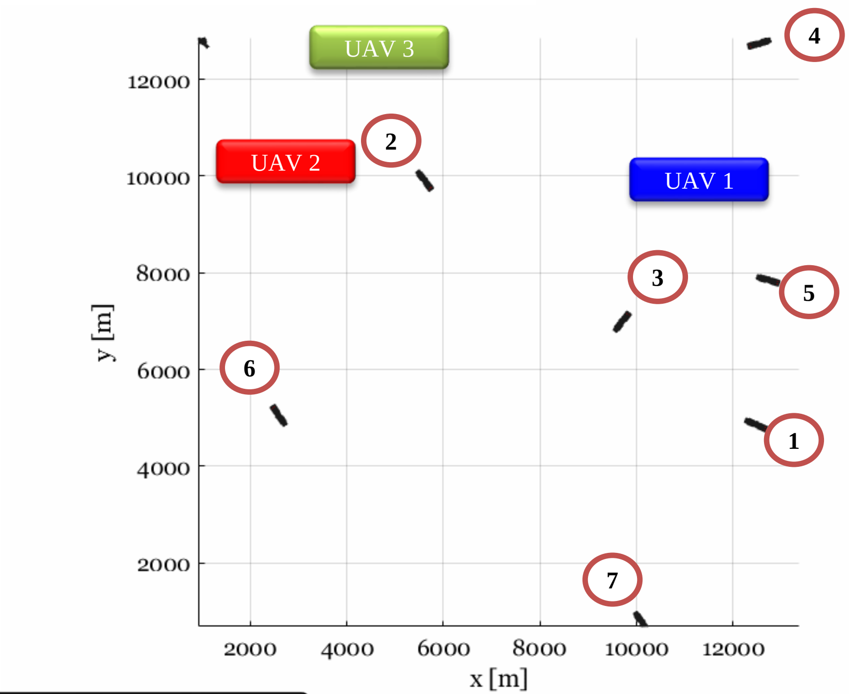
Simulation Evaluation with Trajectory-Tracking Control

$$\min J = \sum_{i=1}^{3} d_i + \left(10^8\right)\sum_{i=1}^{3}\left(n_i - n_{avg}\right)^2$$

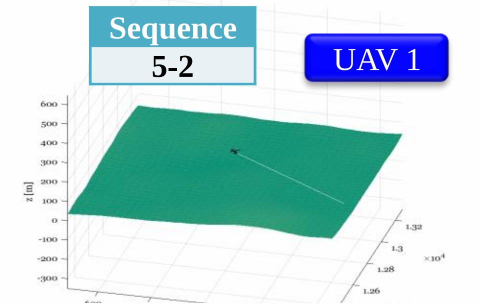
$$n_{avg} = n_{total} / m_{uav} = 2.5$$

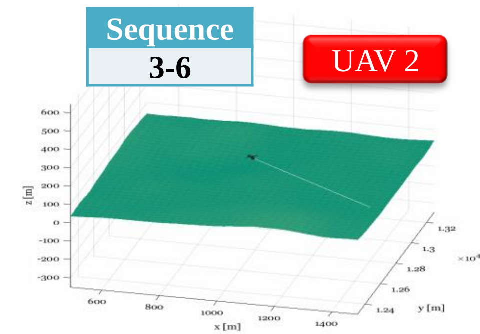


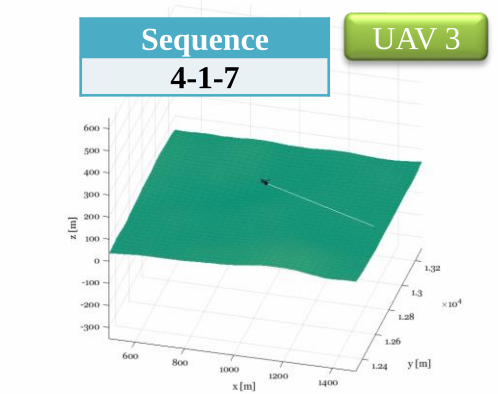




Plot without Terrain







KU 建國大學校 KONKUK UNIV.

Contents

Recent Research Progresses in Rotorcraft Flight Dynamics and Autonomous Flight Control at KKU

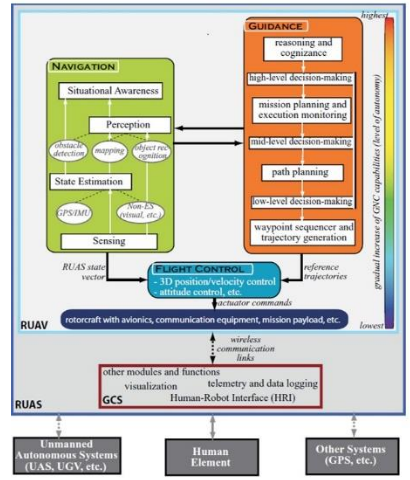
Part 2: Rotorcraft Autonomous Flight Control Systems

Summary

Summary of Part 2 : Rotorcraft Autonomous FCS

KKU Researches have been initially motivated by

- **Kendou's definition of Autonomy Level and Functional Requirements**
- **NASA's researches on RASCAL JUH-60A Black Hawk program**
- **KKU's Mission Scenario Analysis**



- **Rotorcraft Unmanned Aerial Vehicle(RUAV)**
  A powered rotorcraft that does not require an onboard crew, can operate with some degree of autonomy, and can be expendable or reusable.
- **Rotorcraft Unmanned Aerial or Aircraft System(RUAS)**
  A RUAS is a physical system that includes a RUAV, communication architecture, and a ground control station with no human element aboard any component.
- **Navigation System(NS): Perception & State Estimation**
  The process of monitoring and controlling the movement of a craft or vehicle from one place to another.
- **Guidance System(GS)**
  The "driver" of a RUAS that exercises **Mission/Path planning and decision-making functions** to achieve assigned missions or goals.
- **Autonomous Flight Control System(AFCS)**
  The process of manipulating the inputs to a dynamic system to obtain a desired effect on its outputs without a human in the control loop.

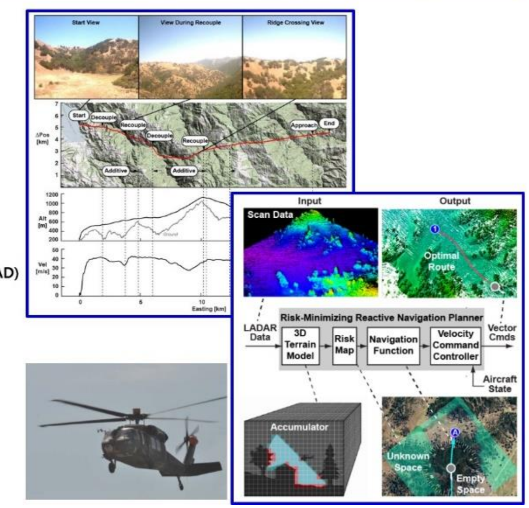
Mission Scenario Analysis for Functional Requirements : Air-to-Ground

Mission Phases	Threats / Obstacles	Terrain Masking	Trajectory	Path Constraints	Aircraft Modes
(1) Take off / Acceleration			Base	TO procedure	RW→FW
(2) Climb			Waypoint	V, RoC	FW
(3) Approach to target zone	Radar / SAM / Terrain / NFZ		Waypoint	V, RoC	FW
(4) Enter into threat aera	Radar / SAM / Terrain / NFZ		Waypoint	V, nz, RoC	FW
(5) target priority selection	Radar / SAM / Terrain / NFZ		Waypoint	V, nz, RoC	FW
(6) Ingress to target zone	Radar / SAM / Terrain / NFZ		Waypoint	V, nz, RoC	FW
(7) Maneuvers for target intercept (multi-target intercept)	Radar / SAM / Terrain / NFZ		Aggressive MTEs	Corridor for Best intercept V, nz, RoC	FW
(8) Egress from target zone	Radar / SAM / Terrain / NFZ		Waypoint	V, nz, RoC	FW
(9) Escape from threat aera	Radar / SAM / Terrain / NFZ		Waypoint	V, nz, RoC	FW
(10) Repeat (3)~(9) as required	Radar / SAM / Terrain / NFZ		Waypoint	V, nz, RoC	FW
Return to base			Waypoint	V, RoC	FW
Deceleration / Landing approach			Waypoint	V, RoD LD procedure	FW→RW
Landing			Base	LD procedure	RW

RW = Rotary Wing Mode
FW = Fixed Wing Mode

Autonomous FCS Structure of RASCAL JUH-60A Black Hawk (US Army)

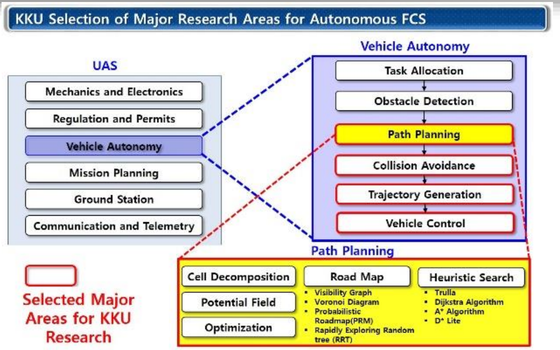
- Multi-Level Autonomy
  - ✓ Fully Coupled Autonomous Mode
  - ✓ Additive Control Mode
  - ✓ Decoupled ACAH Mode
  - ✓ Pilot Interaction with Mode
  - ✓ Control System Design with Mode Transitions
- Mission S/W
  - ✓ Mission Manager/Operator Interface
  - ✓ Obstacle Field Navigation (OFN)
  - ✓ Safe Landing Area Determination (SLAD)
  - ✓ Path Generation
  - ✓ Vector Command
- Autonomous Flight Control S/W (AFCS)
  - ✓ Waypoint Control
  - ✓ Tracking Control
  - ✓ Inner-Loop Control

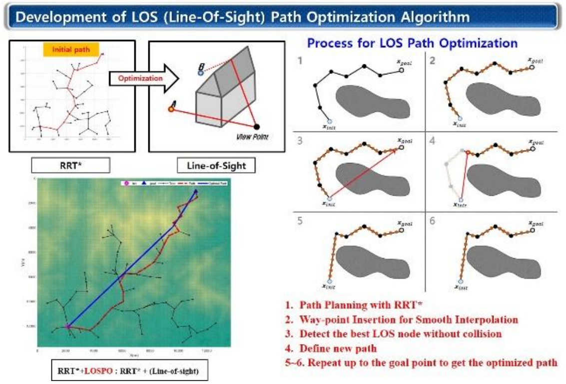


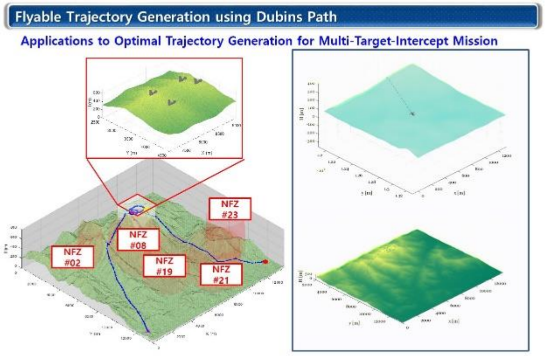
Summary of Part 2 : Rotorcraft Autonomous FCS

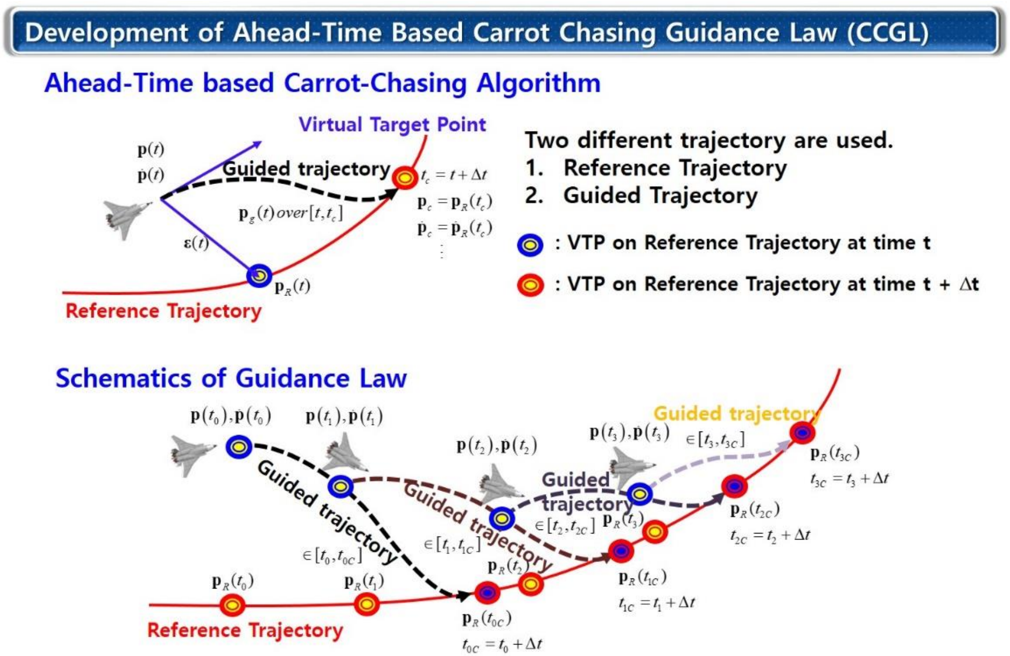
At the initial stage of Studies, KKU mainly focused on

- **Path planning based on RRT combined with Line-Of-Sight Path Optimization (LOSPO)**
- **Flyable trajectory generation avoiding ground collision**
- **Autonomous flight control laws based on the Model Following Control (MFC) framework**
- **Ahead-time based Carrot-Chasing Guidance Laws(CCGL)**









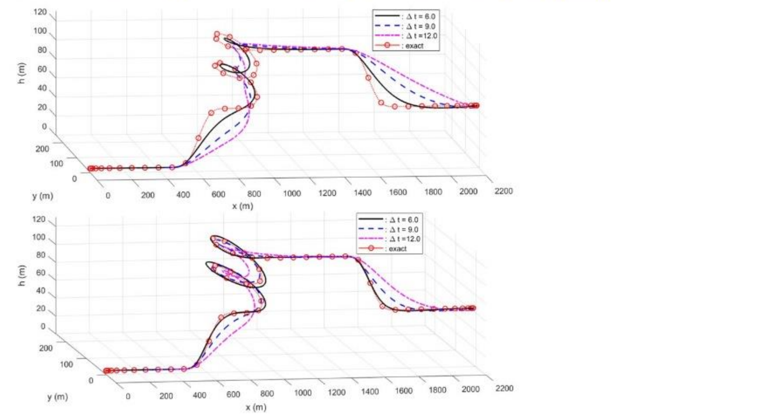
Summary of Part 2 : Rotorcraft Autonomous FCS

Effectiveness of Ahead-Time based CCGL (Carrot-Chasing Guidance Law) has been validated through its application to Autonomous guidance along the composite maneuver course.

Applications of CCGL to Composite Maneuver (MFC structure, GL1/GL3)

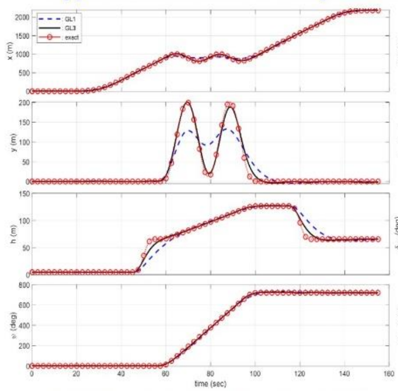
No.	Maneuvers	Duration	State changes during maneuver
1	Hover	15sec	Hover station-keeping at 15ft
2	Acceleration	30sec	Level acceleration from 0 knots to 60knots
3	Pop-up	10sec	Climb from 15 ft up to 215ft and recover the level f light condition
4	Helical Turn	60sec	After 720 deg heading and 200 ft altitude changes, return to level flight
5	Pop-down	10sec	Descent from 415 ft to 215ft and recover the level flight condition
6	Deceleration	30sec	Level deceleration from 60 knots to hover

Applications of CCGL to Composite Maneuver (MFC structure, GL1/GL3)

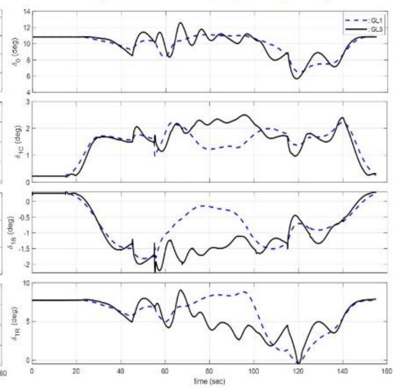


Effect of ahead time on trajectory-tracking accuracy (Upper:GL1, Lower:GL3)

Applications of CCGL to Composite Maneuver (MFC structure, GL1/GL3)

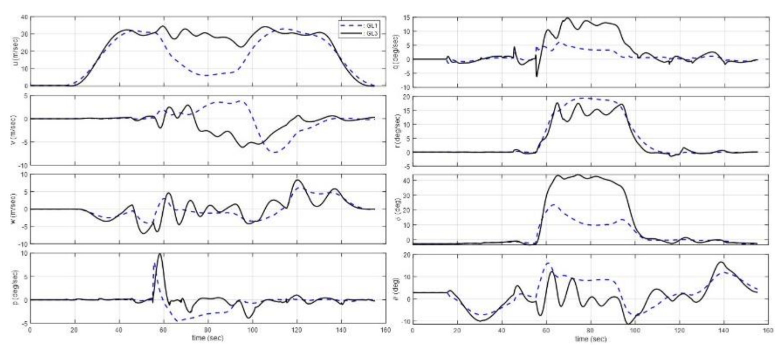


Comparison of trajectory tracking accuracy



Time history of control inputs with $\Delta t = 9.0(\mathrm{sec})$

Applications of CCGL to Composite Maneuver (MFC structure, GL1/GL3)



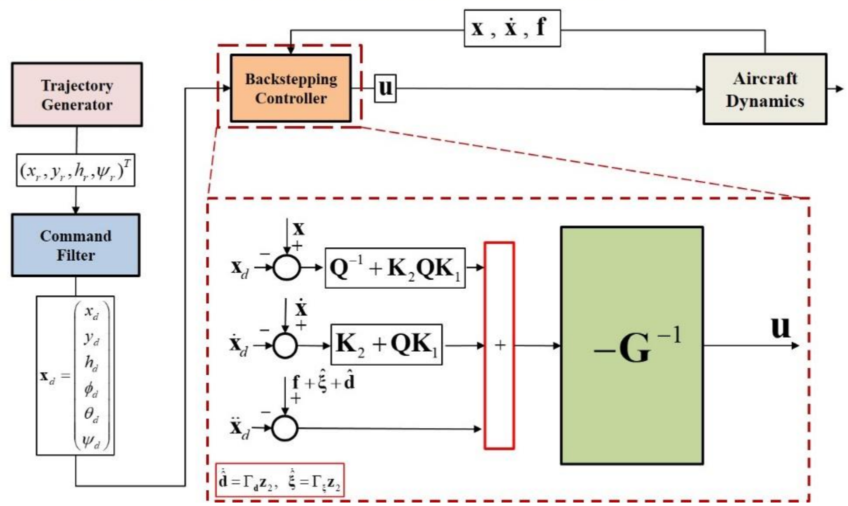
Aircraft states computed with $\Delta t = 9.0(\mathrm{sec})$

Summary of Part 2 : Rotorcraft Autonomous FCS

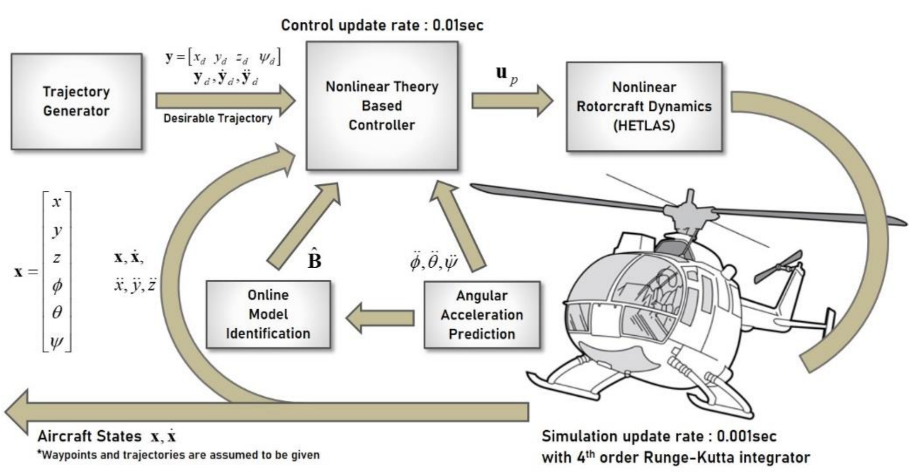
The trajectory-tracking control design, based on IBSC (Incremental Back-Stepping Control) theory, has been developed under the following Know-Hows.

- **Flight Dynamic Model represented in the Inertial Frame is more convenient.**
- **Incremental Dynamics are much more effective for real applications.**
- **Slack-Variable Approach to System Dynamics is extremely effective.**
- **SAS-type functions are working well for the trajectory-tracking IBSC.**
- **Rigorous design works for Gain Optimization can be removed.**

Schematics of Back-Stepping Controller with Command Filter



Simulation Environment for Flight-Control-Law Validation



Summary of Part 2 : Rotorcraft Autonomous FCS

Path-Planning, Flyable Trajectory Generation, and Trajectory-Tracking Control Law has been successfully integrated and validated through a series of Applications.

Combined Maneuver Case

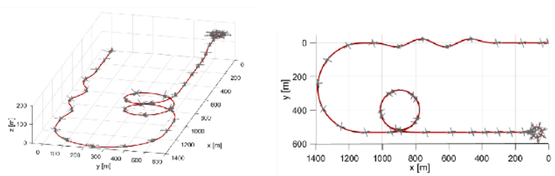
Sequence of Maneuvers

Maneuvers	Length (sec)	Velocity range (kts)	Notes
Initial Condition	0	Hover	Initial Height: 100 ft
Acceleration	0 - 20	0 to 60	/
Slalom	20 - 45	60	/
Transient Turn	45 - 75	60	180 deg turn
Helical Turn	75 - 135	60	720 deg turn
Deceleration	135 - 150	60 - 30	/
Pop up	150 - 160	30	100ft ascent
Deceleration	160 - 175	30 - 0	/
Pirouette	175 - 220	0	Radius: 100 ft

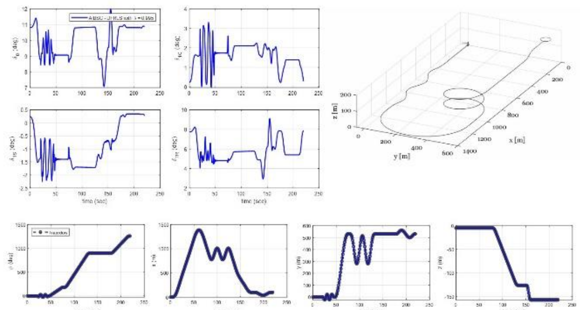
Adaptive IBSC with Least-Squares parameter estimation with direction forgetting

Simulation time step : 0.001sec

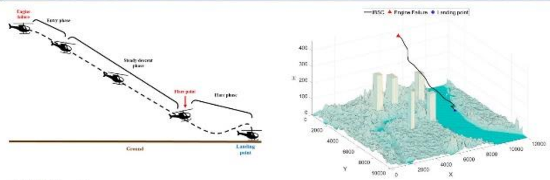
Control update rate : 0.01sec



Combined Maneuver Case: Control inputs and Trajectory States



Autonomous Landing after One Engine Inoperative (OEI) Condition



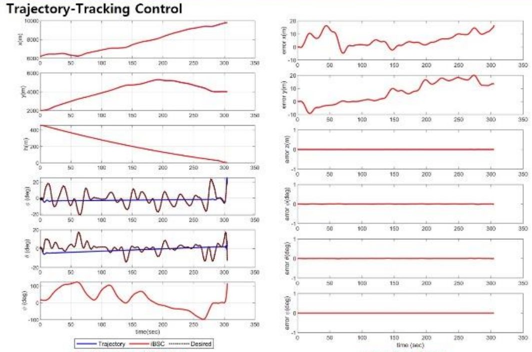
Path Planning
- Entry/Exit Phase: using NOCP solution
- Steady Decent Phase: Bi-directional RRT
  (from Entry final point to Flare initiation point)

Trajectory Generation using Spline Interpolation

Trajectory-Tracking using IBSC

Autonomous Landing after One Engine Inoperative (OEI) Condition

Trajectory-Tracking Control

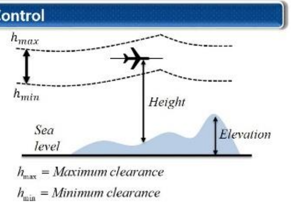


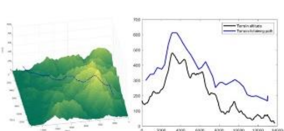
[Position and Attitude] [Tracking Error]

Autonomous Terrain-Following Flight Control

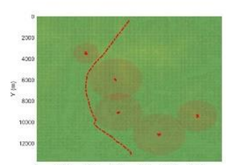
Path Planning Strategy

- RRT algorithm under Height clearance limits
- Real-time planning with unknown terrain information
- Re-planning when detailed terrain information becomes available
- Threat (Radar popup) cost considered



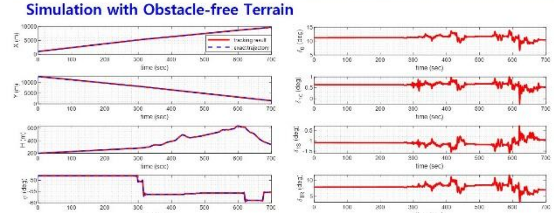


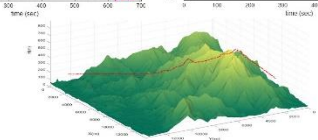
Minimum Clearance distance = 100.0m
Maximum Clearance distance = 200.0m



Autonomous Terrain-Following Flight Control

Simulation with Obstacle-free Terrain





KU 建國大學校 KONKUK UNIV.

End of Part 2

Thank You !!