Design and Application of the BMFCP Architecture in Flight Simulation Systems

Jiaxuan Zhang[a], Runkai Ji and Guanxin Hong
School of Aeronautic Science and Engineering, Beihang University, Beijing, China

Keywords: Flight Simulation, Flight Dynamic, OOP, Software Architecture Design.

Abstract: Flight simulation plays a crucial role in aircraft conceptual design, guidance and control system development, and pilot training. To address the limitations in the architectural design of the dynamics core in traditional flight simulation systems, this study proposes a novel architecture: Boundary-Motion-Force-Coordinate-Parameter (BMFCP), based on the characteristics of flight dynamics problems and object-oriented software development techniques. The BMFCP architecture decomposes the dynamics core of flight simulation systems into three layers: the boundary layer, the motion equation layer, and the external force layer, along with two packages: the coordinate transformation package and the parameter package. Using a flight simulation system based on the BMFCP architecture, simulations of carrier-based aircraft landing and seaplane takeoff and landing processes were successfully conducted. Thanks to the design of this architecture, different flight simulation tasks can be achieved by simply modifying the code in the external force layer to simulate various aircraft. Analysis of the simulation results shows that the time-domain curves of aircraft position and attitude align with empirical observations, validating the correctness of the flight simulation system based on the BMFCP architecture.

1 INTRODUCTION

The flight process is inherently risky and uncertain, especially for newly designed aircraft and inexperienced pilots. Conducting flight missions without comprehensive aircraft performance evaluation and pilot training significantly increases the probability of aircraft accidents. Flight simulation, which uses computers or other devices to simulate aircraft motion and control in real-world environments, plays a crucial role in aircraft design and pilot training. In the field of aircraft design and controller development, (Zhang et al., 2024) highlights that conducting flight simulations during the structural design phase can evaluate the operational performance of the aircraft. Similarly, (Zhao et al., 2024) emphasizes that software-in-the-loop (SITL) simulations can validate control algorithms for aircraft. Regarding pilot training, (Caro, 1973) suggests that training time on flight simulators can replace actual flight training time, while (Thomson, 1989) points out that the degree to which simulators can replace real flight training depends on their fidelity. (Allerton, 2009) notes that compared to the 1970s, when real flights were used for training, modern simulator-based training has significantly reduced the number of training-related accidents. Additionally, (Maciejewska et al., 2024) highlights the economic advantages of simulator-based training.

In recent years, flight simulation technology has shown a rapid development trend. Firstly, the fidelity of models has always been a key focus of related research. (Milne et al., 2023) achieved accurate calculations of aeroelasticity, turbulence, atmosphere, and other effects during high-fidelity motion simulations of sounding rockets, laying the foundation for virtual sensing and digital twins in autonomous navigation and guidance. (An et al., 2022) utilized the flight dynamic model, helicopter trim, linearization, and simulation (HETLAS) system for high-fidelity motion simulations of complex-configured aircraft, effectively describing traditional helicopters, propeller-driven fixed-wing aircraft, and more complex aircraft configurations. (Rezaei and Khosravi, 2022) improved the fidelity of actuator models by conducting parametric model identification using aircraft system data. Secondly, (Dhiman et al., 2025) pointed out that artificial intelligence and data-driven technologies are gradually being applied to aircraft modeling and simulation. (Cao et al., 2022) proposed an

[a] https://orcid.org/0009-0008-2753-4885

Zhang, J., Ji, R. and Hong, G.
Design and Application of the BMFCP Architecture in Flight Simulation Systems.
DOI: 10.5220/0013637000003970
In *Proceedings of the 15th International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH 2025)*, pages 374-381
ISBN: 978-989-758-759-7; ISSN: 2184-2841

interpretable learning algorithm for aircraft systems, adaptive-SINDYc, to identify aircraft models. (Punjani and Abbeel, 2015) employed a rectified linear unit (ReLU) network model to represent helicopter dynamics.

Although recent advancements in flight simulation technology have been significant, limited attention has been paid to the architectural design of the dynamics core in flight simulation systems. A well-designed dynamics core architecture can offer reliability, reusability, readability, maintainability, and expandability advantages. An effective architecture of a dynamics core must integrate various object-oriented techniques in software design and key algorithms for flight dynamics problems. To this end, this study analyzes the establishment of aircraft motion models, key algorithms in flight simulation, and the architectural design of the dynamics core. A novel architecture termed Boundary-Motion-Force-Coordinate-Parameter (BMFCP) is proposed. Using a flight simulation system based on the BMFCP architecture, simulations of carrier-based aircraft landing and seaplane takeoff and landing processes were successfully conducted. The simulation results validate the correctness and effectiveness of the flight simulation system based on BMFCP architecture.

2 FLIGHT MOTION MODEL IN WIND FIELDS

The aircraft motion model in wind fields serves as the core model driving the flight simulation system. This section establishes both the full nonlinear motion model suitable for unsteady operating conditions and the linearized motion model suitable for steady operating conditions.

2.1 Definition of Coordinate Systems and Motion Parameters

We adopt the local tangent plane coordinate system S_{LTP} as the fixed coordinate system, with the x-axis pointing north, the y-axis pointing east, and the origin located at the mean sea level. A flight body coordinate system S_{fb} is established to describe the absolute position and attitude of the aircraft. The origin of S_{fb} is located at the aircraft's center of mass CM, with the x-axis pointing forward, the y-axis pointing to the right, and the z-axis determined by the right-hand rule pointing downward. The absolute motion parameters of the aircraft can be determined by the relative position relationship between S_{fb} and S_{LTP}. Specifically, the absolute position $\boldsymbol{r}_f$ of the aircraft is defined as the vector from point O_{LTP} to point CM as follows.

$$\boldsymbol{r}_f = \boldsymbol{r}_{CM \leftarrow O_{LTP}} \tag{1}$$

The aircraft's attitude Euler angles, including pitch angle ϕ_f, roll angle θ_f, and yaw angle ψ_f, are defined as a set of Euler angles that transform the S_{LTP} to the S_{fb} using a $z-y-x$ rotation sequence. the absolute Euler angle vector of the aircraft $\boldsymbol{\Theta}_f \in \mathbb{R}^3$ is defined as follows.

$$\boldsymbol{\Theta}_f = \begin{pmatrix} \phi_f & \theta_f & \psi_f \end{pmatrix}^\top \tag{2}$$

The aerodynamic coordinate system S_a describes the incoming airflow relative to the aircraft. The origin of this coordinate system is located at the aircraft's aerodynamic center (AC). The x-axis is aligned with the direction of the airflow velocity vector $\boldsymbol{v}_a$, pointing towards the aircraft's nose. The z-axis lies within the aircraft's symmetry plane, perpendicular to $\boldsymbol{v}_a$, and points downward. The y-axis is determined by the right-hand rule, pointing to the right side of the aircraft.

The aerodynamic angles of the aircraft, including the angle of attack α_f and the sideslip angle β_f, can be defined through the transformation relationship from S_{fb} to S_a as follows

$$\boldsymbol{S}_{fb} \xrightarrow{\boldsymbol{R}_z(-\alpha_f)} \circ \xrightarrow{\boldsymbol{R}_y(\beta_f)} \boldsymbol{S}_a \tag{3}$$

based on the above relationship, the angle of attack α_f and the sideslip angle β_f of the aircraft can be derived from the components of the airflow velocity vector $\boldsymbol{v}_a$ in S_{fb} as follows.

$$\left.\begin{aligned} \alpha_f &= \arctan\left(\frac{v_{a,zfb}}{v_{a,xfb}}\right) \\ \beta_f &= \arcsin\left(\frac{v_{a,yfb}}{|\boldsymbol{v}_a|}\right) \end{aligned}\right\} \tag{4}$$

The flight path coordinate system S_k is used to describe the aircraft's flight path and track velocity $\boldsymbol{v}_k$. The origin of this coordinate system is located at the aircraft's center of mass CM. The x-axis is aligned with the track velocity $\boldsymbol{v}_k$ and points towards the aircraft's nose. The z-axis is perpendicular to the x-axis and lies in a vertical plane, pointing downward. The y-axis is determined by the right-hand rule and points to the right side of the aircraft.

The flight path angles, including the track azimuth angle χ_f and the track inclination angle γ_f, can be defined through the transformation relationship from S_{LTP} to S_k as follows

$$\boldsymbol{S}_{LTP} \xrightarrow{\boldsymbol{R}_z(\chi_f)} \circ \xrightarrow{\boldsymbol{R}_y(\gamma_f)} \boldsymbol{S}_k \tag{5}$$

based on the above relationship, the track azimuth angle χ_f and the track inclination angle γ_f of the aircraft can be derived from the components of the track velocity vector $\boldsymbol{v}_k$ in S_{LTP} as follows.

$$\left.\begin{aligned} \gamma_f &= \arcsin\left(-\frac{v_{k,zLTP}}{|\boldsymbol{v}_k|}\right) \\ \chi_f &= \arctan\left(\frac{v_{k,yLTP}}{v_{k,xLTP}}\right) \end{aligned}\right\} \tag{6}$$

2.2 Full Nonlinear Motion Model

For a rigid body undergoing general motion, based on the general motion equations of a rigid body with respect to its center of mass, the equations of motion for an aircraft in Euler angle form are expressed as follows

$$\begin{pmatrix} \left(\frac{\mathrm{d}\boldsymbol{r}_f}{\mathrm{d}t}\right)_{fb} \\ \dot{\boldsymbol{\Theta}}_f \\ \left(\frac{\mathrm{d}\boldsymbol{v}_f}{\mathrm{d}t}\right)_{fb} \\ \left(\frac{\mathrm{d}\boldsymbol{\omega}_f}{\mathrm{d}t}\right)_{fb} \end{pmatrix} = \begin{pmatrix} (\boldsymbol{v}_f)_{fb} \\ \boldsymbol{D}_f^{-1}(\boldsymbol{\omega}_f)_{fb} \\ m_f^{-1}(\boldsymbol{F}_f)_{fb} \\ (\boldsymbol{J}_f)_{fb}^{-1}\left[(\boldsymbol{M}_{f\leftarrow O_{cm}})_{fb} - (\boldsymbol{\omega}_f)_{fb}^{\times}(\boldsymbol{J}_f)_{fb}(\boldsymbol{\omega}_f)_{fb}\right] \end{pmatrix} \tag{7}$$

where $\boldsymbol{F}_f$ represents the resultant external force acting on the aircraft; $\boldsymbol{M}_{f\leftarrow O_{cm}}$ denotes the resultant external moment about the center of mass; $(\boldsymbol{J}_f)_{fb}$ is the inertia matrix of the aircraft relative to its center of mass and S_{fb}; $\boldsymbol{D}_f$ is the transformation matrix from Euler angle rates to angular velocity.

The resultant external force $\boldsymbol{F}_f$ and moment $\boldsymbol{M}_{f\leftarrow O_{cm}}$ acting on most aircraft consist of three components: aerodynamic force, gravitational force, and thrust. The aerodynamic force $\boldsymbol{F}_{fa}$ and moment $\boldsymbol{M}_{fa\leftarrow O_{cm}}$ can be expressed as follows

$$(\boldsymbol{F}_{fa})_{fb} = \bar{q}S\boldsymbol{R}_{a\leftarrow fb}^{\top}\begin{pmatrix}-C_X & C_Y & -C_Z\end{pmatrix}^{\top} \tag{8}$$

$$\begin{aligned}(\boldsymbol{M}_{fa\leftarrow O_{cm}})_{fb} =& \bar{q}S\boldsymbol{R}_{a\leftarrow fb}^{\top}\begin{pmatrix}bC_l & \bar{c}C_m & bC_n\end{pmatrix}^{\top} \\ &+ (\boldsymbol{r}_{AC\leftarrow CM})_{fb}^{\times} \times (\boldsymbol{F}_{fa})_{fb}\end{aligned} \tag{9}$$

where $C_X, C_Y, C_Z, C_l, C_m, C_n$ represent the drag coefficient, side force coefficient, lift coefficient, rolling moment coefficient, pitching moment coefficient, and yawing moment coefficient, respectively. These parameters are functions of the aircraft's motion and aerodynamic parameters, typically obtained through wind tunnel experiments or computational fluid dynamics (CFD) methods. In simulations, these parameters are often retrieved via lookup tables. S denotes the wing area, b the wingspan, $\bar{c}$ the mean aerodynamic chord, $\bar{q}$ the dynamic pressure, and $\boldsymbol{r}_{AC\leftarrow CM}$ the vector from the aircraft's CM to AC.

The gravitational force $\boldsymbol{F}_{fg}$ acting on the aircraft, expressed in S_{fb} is given as follows

$$(\boldsymbol{F}_{fg})_{fb} = \boldsymbol{R}_{fb\leftarrow LTP}\begin{pmatrix}0 & 0 & -m_f g\end{pmatrix}^{\top} \tag{10}$$

where m_f represents the mass of the aircraft, and g denotes the gravitational acceleration.

The thrust $\boldsymbol{F}_{ft}$ generated by the aircraft engine and the additional moment $\boldsymbol{M}_{ft}$ induced by its installation can be expressed as follows

$$(\boldsymbol{F}_{ft})_b = \boldsymbol{R}_z^{\top}(\theta_{eng,ins})\begin{pmatrix}T & 0 & 0\end{pmatrix}^{\top} \tag{11}$$

$$(\boldsymbol{M}_{ft})_b = (\boldsymbol{r}_{eng\leftarrow fb})_{fb}^{\times} \times (\boldsymbol{F}_{ft})_{fb} \tag{12}$$

where $\theta_{eng,ins}$ represents the engine installation angle, T denotes the engine thrust, and $\boldsymbol{r}_{eng\leftarrow fb}$ is the position vector of the engine thrust center relative to the aircraft's center of mass.

The inputs to the aircraft dynamics system include five control variables: the left and right elevator deflections δ_{el}, δ_{er}, the aileron deflection δ_a, the rudder deflection δ_r, and the throttle setting τ. Additionally, there are three wind field disturbances, represented by the components of the wind velocity vector $\boldsymbol{v}_w$ in S_{TLP}: $u_{w,LTP}, v_{w,LTP}, w_{w,LTP}$.

The essence of full nonlinear flight simulation lies in the numerical integration of complex nonlinear differential equations. In the simulation, the continuous states are denoted as $\boldsymbol{c}$, the discrete states as $\boldsymbol{d}$, the inputs as $\boldsymbol{i}$, and the outputs as $\boldsymbol{o}$, with time t being a proper subset of the inputs $t \subset \boldsymbol{i}$. The full nonlinear flight simulation can be expressed as the following set of equations

$$\left.\begin{aligned} \dot{\boldsymbol{c}} &= f(\boldsymbol{c},\boldsymbol{d},\boldsymbol{i}) \\ \boldsymbol{d} &= g(\boldsymbol{c},\boldsymbol{i}) \\ \boldsymbol{o} &= h(\boldsymbol{c},\boldsymbol{d},\boldsymbol{i}) \end{aligned}\right\} \tag{13}$$

where $f(\boldsymbol{c},\boldsymbol{d},\boldsymbol{i})$ represents the continuous state equations, which, in the context of flight simulation, correspond to the equations of motion shown in Eq. 7. However, certain aircraft state variables, such as the angle of attack α_f, are not part of $\boldsymbol{c}$ but directly influence the computation on the right-hand side of Eq. 7. These parameters can be derived using algebraic equations $g(\boldsymbol{c},\boldsymbol{i})$ related to the continuous states $\boldsymbol{c}$ and inputs $\boldsymbol{i}$, such as Eq. 4. To provide comprehensive simulation results, the outputs $\boldsymbol{o}$ are defined as a combination of the continuous states $\boldsymbol{c}$, their derivatives $\dot{\boldsymbol{c}}$, and the discrete states $\boldsymbol{d}$.

2.3 Linearized Motion Model

When an aircraft performs steady-state motion (e.g., steady-level flight or steady descent), its motion can be well represented by a linearized model under small perturbation conditions.

We define the state variables $\Delta\boldsymbol{X} \in \mathbb{R}^9$ and input variables $\Delta\boldsymbol{U} \in \mathbb{R}^8$ for the aircraft as follows

$$\Delta\boldsymbol{X} = \begin{pmatrix} \Delta\boldsymbol{\Theta}_f \\ \Delta(\boldsymbol{v}_f)_{fb} \\ \Delta(\boldsymbol{\omega}_f)_{fb} \end{pmatrix} \tag{14}$$

$$\Delta\boldsymbol{U} = \begin{pmatrix} \Delta\delta_{el} \\ \Delta\delta_{er} \\ \Delta\delta_a \\ \Delta\delta_r \\ \Delta\tau \\ \Delta u_{w,LTP} \\ \Delta v_{w,LTP} \\ \Delta w_{w,LTP} \end{pmatrix} \tag{15}$$

where Δ represents the perturbation relative to the trim state. In the linearized motion model, we select the system output as $\Delta\boldsymbol{Y} = \Delta\boldsymbol{X}$. Additional outputs, such as the angle of attack α_f and sideslip angle β_f, can be obtained through nonlinear calculations based on the state variables $\Delta\boldsymbol{X}$ and input variables $\Delta\boldsymbol{U}$. The aircraft's position can be determined by integrating the velocity.

The linearized motion model of the aircraft can be expressed in the form of the following state-space equations

$$\begin{aligned} \Delta\dot{\boldsymbol{X}} &= \boldsymbol{A}\Delta\boldsymbol{X} + \boldsymbol{B}\Delta\boldsymbol{U} \\ \Delta\boldsymbol{Y} &= \Delta\boldsymbol{X} \end{aligned} \tag{16}$$

where $\boldsymbol{A}$ and $\boldsymbol{B}$ are the Jacobian matrices of the state-space equations, defined as follows

$$\begin{aligned} \boldsymbol{A} &= \left.\frac{\partial\dot{\boldsymbol{X}}}{\partial\boldsymbol{X}}\right|_{\boldsymbol{X}_{trim},\boldsymbol{U}_{trim}} \\ \boldsymbol{B} &= \left.\frac{\partial\dot{\boldsymbol{X}}}{\partial\boldsymbol{U}}\right|_{\boldsymbol{X}_{trim},\boldsymbol{U}_{trim}} \end{aligned} \tag{17}$$

where $\boldsymbol{X}_{trim}$ and $\boldsymbol{U}_{trim}$ represent the state variables and input variables of the aircraft in the trimmed condition, respectively.

3 BMFCP ARCHITECTURE DESIGN IN FLIGHT SIMULATION SYSTEMS

This study is based on key issues in flight dynamics and incorporates object-oriented software design techniques to propose a Boundary-Motion-Force-Coordinate-Parameter (BMFCP) architecture for the dynamics core of flight simulation systems. The BMFCP architecture is based on the Boundary-Control-Entity (BCE) three-layer architecture proposed by (Martin, 2002), with adaptive improvements tailored to flight dynamics problems. This architecture offers advantages in reliability, reusability, readability, maintainability, and extensibility.

3.1 Boundary Layer Design

In the BMFCP architecture, the Boundary Layer is responsible for interactions between the system and its actor. The design of the Boundary Layer adheres to the interface segregation principle (ISP) proposed by (Martin, 1996). Taking interaction with the Simulink system as an example, the class diagram of the boundary layer interacting with Simulink is shown in Figure 1. Where the Simulink Level-2 MATLAB S-Function is a tool in Simulink used for creating custom Simulink block interfaces using MATLAB code.

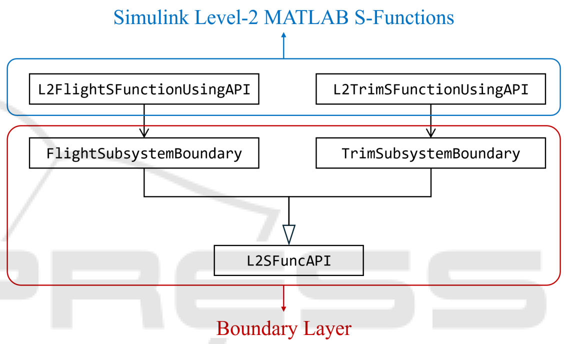


Figure 1: Class Diagram of the Boundary Layer.

3.2 Motion Equation Layer Design

The motion equation layer in the BMFCP architecture corresponds to the control layer in the BCE three-layer architecture. Since the motion equations are applicable to any aircraft under any operating conditions, this layer is the most stable and invariant in design. The class diagram of the motion equation layer is shown in Figure 2. The `CalcDerivatives` class is responsible for solving the continuous state derivatives of the aircraft, corresponding to the function $\dot{c} = f(c,d,i)$ in Eq. 13. The `UpdateDiscStates` class is used to compute the discrete states of the aircraft, corresponding to the function $d = g(c,i)$ in Eq. 13. The `DynamicEquation` class describes the aircraft's dynamic equations as shown in Eq. 7.

3.3 External Force Layer Design

The external force layer in the BMFCP architecture corresponds to the Entity Layer in the BCE three-layer architecture. Due to the varying force calculation methods for different aircraft, the design of the

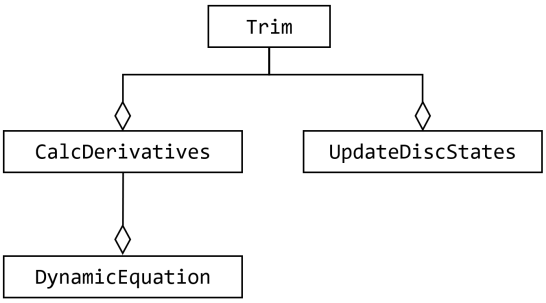


Figure 2: Class Diagram of the Motion Equation Layer.

external force layer must be interchangeable and extensible. For instance, when simulating the F-18 High Angle of Attack Research Vehicle (HARV), the class diagram of the external force layer is shown in Figure 3. The resultant force class, `ResultantForce`, comprises several sub-forces. According to the Liskov substitution principle proposed by (Liskov and Wing, 1994), the classes `AeroForceF18HARV`, `EngineF18HARV`, and `GravityF18HARV` in the external force layer can be replaced with classes describing the forces acting on other aircraft, thereby enabling simulations for different aircraft. Following the open-closed principle (OCP) proposed by (Meyer, 1988), the sub-force classes in the external force layer can be extended. For example, additional classes describing hydrodynamic forces and buoyancy can be added to simulate seaplanes, thereby expanding the simulation capabilities for such aircraft.

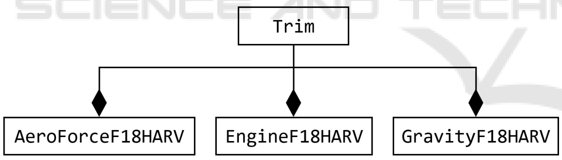


Figure 3: Class Diagram of the External Force Layer.

3.4 BMFCP Architecture Design

In addition to the three primary layers mentioned earlier, the BMFCP architecture also includes two packages: the coordinate transformation package (`CordTrans`) and the parameter package (`Parameter`). The `CordTrans` package provides frequently used coordinate transformation methods, while the `Parameter` package allows for the definition of constants in the MATLAB environment, offering the advantage of easily switching between different aircraft parameters. Since these two packages are respectively dependent on the motion equation layer and the external force layer, the overall package diagram of the BMFCP architecture is proposed in accordance with the acyclic dependencies principle (ADP) introduced by (Martin, 2002), as shown in Figure 4.

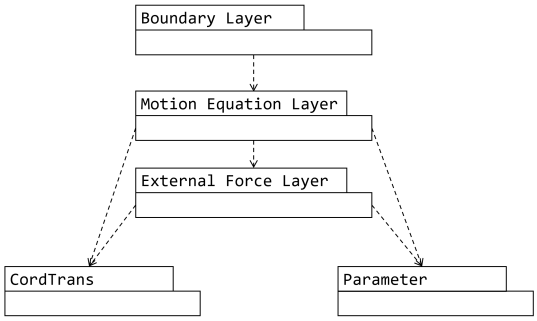


Figure 4: Architecture Package Diagram.

4 SIMULATION RESULT

To facilitate the design of carrier-based aircraft landing guidance and control laws, as well as the conceptual design of seaplanes, we conducted simulation studies on the landing process of a carrier-based aircraft and the takeoff and landing processes of a seaplane. In this section, we present the results of these simulations and analyze their outcomes.

4.1 Case Study of Carrier-Based Aircraft Landing

Due to the track inclination angle γ_f is a constant during carrier-based aircraft landing, operating condition above is a steady-state condition. Therefore, using a linearized aircraft model for simulation is appropriate. In this study, the F-18 HARV is used as the test aircraft, with its parameters sourced from (Asbury and Capone, 1995), (Iliff, 1997), and (Iliff, 1999). The aircraft is trimmed to a position where its tailhook is 50 meters below the commanded altitude to evaluate the performance of the guidance and control system. The specific trim parameters are shown in Table 1.

Table 1: Carrier-Based Flight Landing Trim Parameter.

Parameter	Trim value
Initial hook height error, h_{he0} [m]	-50
Track inclination angle, γ_f [Deg]	-4
Track speed, V_k [m/s]	80
Pitch angle, θ_f [Deg]	2.15

The aircraft's tailhook's absolute and commanded heights are shown as the blue and red lines in Figure 5, respectively. The simulation results indicate that the aircraft corrects the tailhook height error within approximately 20 seconds and effectively maintains the commanded tailhook height during the final approach phase before touchdown. These results demonstrate the effectiveness of the flight simulation system based on the BMFCP architecture when conducting simulations using linearized models.

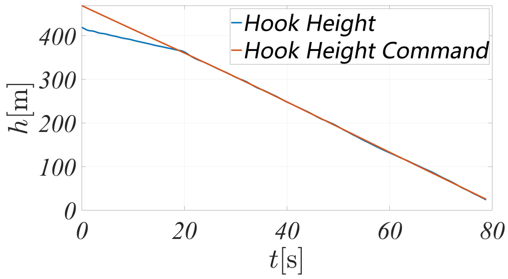


Figure 5: Carrier-Based Flight Landing Height.

4.2 Case Studies of Seaplane Takeoff and Landing

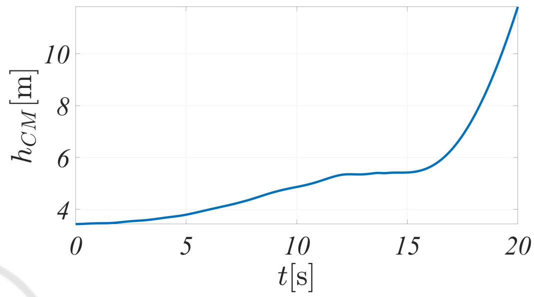
Seaplanes are aircraft capable of taking off and landing on water surfaces. Simulating the takeoff and landing processes during the preliminary design phase can facilitate rapid iterative optimization of design schemes. Due to seaplane takeoff and landing operations' highly complex, nonlinear, and unsteady nature, the aircraft's full nonlinear motion model must be employed for simulation. Thanks to the design of the external force layer in the BMFCP architecture, seaplane takeoff and landing simulation can be achieved by simply adding classes that describe hydrodynamic forces and buoyancy to this layer.

This study conducted a simulation of the takeoff process for a specific type of seaplane. The seaplane's takeoff process includes three stages: stationary floating on the water, water taxiing, and liftoff. The simulation begins with the aircraft floating stationary on the water, with the control surfaces trimmed to a state corresponding to a fixed track inclination angle γ_f during ascent. Table 2 shows the specific trim parameters.

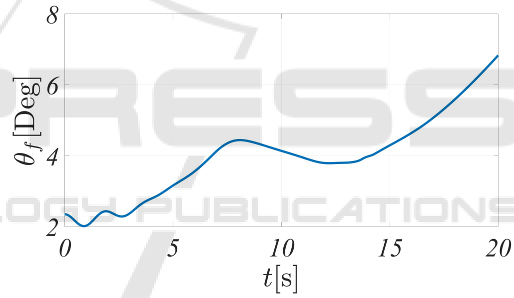
Table 2: Seaplane Takeoff Trim Parameter.

Parameter	Trim value
CM height, h_{CM} [m]	3.34
Pitch angle, θ_f [Deg]	2.34
Elevator angle, δ_e [Deg]	-5.92
Aileron angle δ_a [Deg]	0
Rudder angle δ_r [Deg]	0
Throttle	1

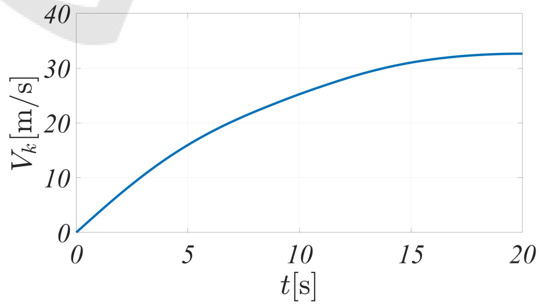
The time-domain simulation results of the seaplane takeoff process are shown in Figure 6. The results indicate that the aircraft lifts off from the water surface approximately 15 seconds after starting from rest. After liftoff, the aircraft continues to climb, reaching an altitude of $h_{CM} = 10$m with a track inclination angle of $\gamma_f \approx 0.5°$. The track velocity V_k stabilizes around 30m/s after liftoff. These results are consistent with the actual takeoff process of a seaplane, validating the effectiveness of the flight simulation system based on the BMFCP architecture when conducting simulations using the full nonlinear motion model.



(a) Takeoff CM height.



(b) Takeoff Pitch angle.



(c) Takeoff Track Speed.

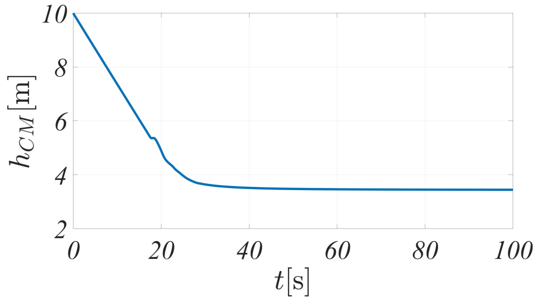
Figure 6: Time Domain Curve of Seaplane Takeoff.

This study also conducted a simulation of the landing process for a specific type of seaplane. The landing process of the seaplane includes three stages: steady descent, water touchdown, and deceleration during water taxiing. The simulation begins with the aircraft in a steady descent state, with the control surfaces and attitude trimmed to maintain a steady descent condition. The specific trim parameters are shown in Table 3.

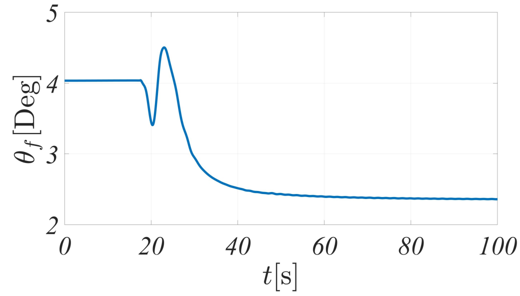
Table 3: Seaplane Landing Trim Parameter.

Parameter	Trim value
CM height, h_{CM} [m]	10
Track inclination angle, γ_f [Deg]	-0.5
Track speed, V_k [m/s]	30.3
Pitch angle, θ_f [Deg]	4.03
Angle of attack, α_f [Deg]	4.53

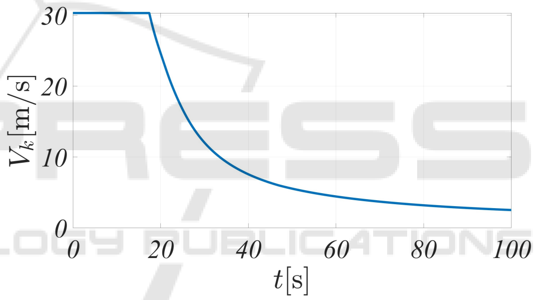
The time-domain simulation results of the seaplane landing process are shown in Figure 7. The simulation results indicate that the aircraft touches down on the water surface at approximately 17 seconds. After touchdown, the track velocity V_k decreases rapidly and reduces to approximately $2.5m/s$ at 100 seconds. The pitch angle θ_f and angle of attack α_f exhibit fluctuations of around 1° upon water impact due to hydrodynamic forces. These results are consistent with the actual landing process of a seaplane, validating the effectiveness of the flight simulation system based on the BMFCP architecture when conducting simulations using the full nonlinear motion model.



(a) Takeoff CM height.



(b) Takeoff Pitch angle.



(c) Takeoff Track Speed.

Figure 7: Time Domain Curve of Seaplane Landing.

5 CONCLUSIONS

This study, through an in-depth analysis of flight dynamics problems and their key algorithms, combined with object-oriented software design techniques, proposes a Boundary-Motion-Force-Coordinate-Parameter (BMFCP) architecture applicable to the dynamics core of flight simulation systems. This architecture enhances flight simulation systems' reliability, reusability, readability, maintainability, and extensibility. These advantages significantly reduce modification costs and failure probabilities when conducting flight simulations for different aircraft and operation conditions.

This study presents and analyzes two sets of simulation results based on the linearized and full nonlinear motion models of aircraft, focusing on the carrier-based aircraft landing process and the seaplane takeoff and landing processes. The results validate the effectiveness of the flight simulation system based on the BMFCP architecture in conducting simulations using both linearized and full nonlinear models. Furthermore, the study highlights the significant role of this simulation system in the design of aircraft guidance and control laws and overall aircraft design. Additionally, the system demonstrates potential as the dynamics core of flight simulators, with promising future pilot training applications.

REFERENCES

Allerton, D. (2009). *Principles of Flight Simulation*. Wiley.

An, J.-Y., Choi, Y.-S., Lee, I.-R., Lim, M., and Kim, C.-J. (2022). Performance analysis of a conceptual urban air mobility configuration using high-fidelity rotorcraft flight dynamic model. *Int. J. Aeronaut. Space Sci.*, 24:1491–1508.

Asbury, S. C. and Capone, F. J. (1995). Multiaxis thrust-vectoring characteristics of amodel representative of the f-18 high-alpharesearch vehicle at angles of attack from 0°to 70°. Technical report, Langley Research Center.

Cao, R., Lu, Y., and He, Z. (2022). System identification method based on interpretable machine learning for unknown aircraft dynamics. *Aerospace Science and Technology*, 126:107593.

Caro, P. W. (1973). Aircraft simulators and pilot training. *Human Factors*, 15:502–509.

Dhiman, G., Tiumentsev, A. Y., and Tiumentsev, Y. V. (2025). Neural network and hybrid methods in aircraft modeling, identification, and control problems. *Aerospace*, 12(1).

Iliff, K. W. (1997). Flight-determined subsonic longitudinalstability and control derivatives of the f-18high angle of attack research vehicle (harv)with thrust vectoring. Technical report, Dryden Flight Research Center.

Iliff, K. W. (1999). Flight-determined, subsonic, lateral-directional stability and control derivatives of the thrust-vectoring f-18 high angle of attack research vehicle (harv), and comparisons to the basic f-18 and predicted derivatives. Technical report, Dryden Flight Research Center.

Liskov, B. and Wing, J. (1994). A behavioral notion of subtyping. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 16(6):1811–1841.

Maciejewska, M., Kurzawska-Pietrowicz, P., Galant-Gołębiewska, M., Gołębiewski, M., and Jasiński, R. (2024). Ecological and cost advantage from the implementation of flight simulation training devices for pilot training. *Applied Sciences*, 14(18).

Martin, R. C. (1996). The interface segregation principle. *C++ Report*, 8(7):57–62.

Martin, R. C. (2002). *Agile Software Development: Principles, Patterns, and Practices*. Prentice Hall.

Meyer, B. (1988). *Object-Oriented Software Construction*. Prentice Hall.

Milne, B., Samson, S., Carrese, R., Gardi, A., and Sabatini, R. (2023). High-fidelity dynamics modelling for the design of a high-altitude supersonic sounding rocket. *Designs*, 7(1).

Punjani, A. and Abbeel, P. (2015). Deep learning helicopter dynamics models. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3223–3230.

Rezaei, P. and Khosravi, A. (2022). Parametric model identification of nonlinear aircraft system with actuator saturation. In *2022 International Conference on Smart Information Systems and Technologies (SIST)*, pages 1–4.

Thomson, D. R. (1989). Transfer of training from simulators to operational equipment—are simulators effective? *Journal of Educational Technology Systems*, 17(3):213–218.

Zhang, S., Tong, C., Ni, Y., Li, N., and Lin, Z. (2024). Structural design and flight simulation of firefighting and rescue uav based on coaxial dual rotors. *International Journal of Pattern Recognition and Artificial Intelligence*, 38(12):2458004.

Zhao, W., Wang, Y., Li, L., Huang, F., Zhan, H., Fu, Y., and Song, Y. (2024). Design and flight simulation verification of the dragonfly evtol aircraft. *Drones*, 8(7).