CFDAgent with a general kinematics module: driving biological flapping flight simulation

Citation for the original published paper (version of record):

Xu, Z., Yao, H., Wang, S. et al (2026). CFDAgent with a general kinematics module: driving biological flapping flight simulation. Theoretical and Applied Mechanics Letters, 16(5).
http://dx.doi.org/10.1016/j.taml.2026.100734

(article starts on next page)

Theoretical and Applied Mechanics Letters 16 (2026) 100734

Contents lists available at ScienceDirect

Theoretical and Applied Mechanics Letters

journal homepage: www.elsevier.com/locate/taml

Research Article

CFDAgent with a general kinematics module: Driving biological flapping flight simulation

Zhaoyue Xu [a,*], Hua-Dong Yao [a], Shizhao Wang [b,c], Guowei He [b,c]

[a] Department of Mechanical Engineering, Chalmers University of Technology, Gothenburg 41296, Sweden
[b] The State Key Laboratory of Nonlinear Mechanics, Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China
[c] School of Engineering Sciences, University of Chinese Academy of Sciences, Beijing 100049, China

ARTICLE INFO

Keywords:
Large language model agent
Computational fluid dynamics
Immersed boundary method
Kinematics module

ABSTRACT

Automating moving-boundary computational fluid dynamics (CFD), particularly for simulations of biological flapping flight, requires complex coordination across geometry, meshing, kinematics, and postprocessing. We present an enhanced CFDAgent, a large language model (LLM)-orchestrated platform that executes the entire workflow end-to-end through natural-language dialogue, spanning geometry intake, solver execution, and iterative figure generation. The agent features a general kinematics module that directly converts natural-language motion specifications or measured trajectories into the periodic body motion incorporated into the solver. We validate the platform's accuracy and versatility against three distinct benchmarks: a rotating sphere, a flapping plate, and a slow-flying Pallas's long-tongued bat (*Glossophaga soricina*) with deforming wings. Furthermore, we apply the workflow to investigate the aerodynamic effects of hindwing damage in migrating painted lady butterflies (*Vanessa cardui*). The simulations show that increasing hindwing trailing-edge erosion alters the wake vortex structures, resulting in progressively greater aerodynamic lift loss.

1. Introduction

Large language models (LLMs) have achieved strong performance on natural-language tasks including code generation, mathematical reasoning, and few-shot transfer [1]. Computational fluid dynamics (CFD) is more difficult to address directly with LLMs alone: a complete simulation requires coordinated handling of geometry preparation, mesh generation, numerical scheme selection, boundary conditions, and postprocessing, each of which requires domain knowledge that standalone LLMs cannot reliably apply based on natural-language prompts alone [2,3]. Earlier data-driven approaches [4], such as neural-network surrogates trained on parametric studies [5], accelerate specific subproblems but do not replace the full simulation pipeline.

Recent LLM-driven agent frameworks address this difficulty by coupling LLMs with established CFD solvers, using retrieval, structured prompting, or fine-tuning to provide solver-specific knowledge. OpenFOAMGPT [6] and its cost-optimized successor [7] use retrieval to compile OpenFOAM cases from natural-language prompts; FoamGPT [8] fine-tunes the underlying LLM on curated OpenFOAM tutorial sets; LLM-PSO [9] applies in-context learning to evolutionary shape optimization; domain-specific agents have been developed for turbomachinery design [10], urban canopy flows [11], and simulation automation via the Model Context Protocol [12]; and a review-and-revision workflow has been proposed for general scientific-computing tasks [13]. Recent reviews summarize the field [14]. Existing LLM-CFD agents have mainly been demonstrated for stationary geometries and generally lack a kinematics interface for prescribed moving or deforming boundaries. Body-fitted moving-boundary simulations require mesh deformation, remeshing, or moving-interface coupling, whose reliable automation through dialogue remains difficult and may limit setup fidelity.

Biological flight poses more difficult problems for agent frameworks because the body moves. The lift-generating mechanisms of insects, birds, and bats [15] are unsteady and vortex-dominated, but vary widely in geometric and kinematic complexity. Simplified flapping-plate models isolate these mechanisms and show that lift is sensitive to kinematic phasing, spanwise oscillation, and gap geometry [16–18]. Real wings add further complexity: coupled forewing–hindwing strokes in butterflies produce thrust and underlie the evolution of distinct flight modes [19], and slow-flying bats sustain a leading-edge vortex (LEV) on a dynamically deforming membrane wing [20,21]. Natural wing damage and molt gaps further alter the wing geometry itself, adding another layer of complexity to the analysis [22,23]. Moving-boundary CFD is typically addressed through either body-fitted meshes (overset, sliding, or deforming) that track the body [24], or immersed boundary (IB) methods [25,26] that keep a fixed Cartesian Eulerian grid and represent the body as a Lagrangian surface enforcing time-dependent no-slip

* Corresponding author.
E-mail address: zhaoyue@chalmers.se (Z. Xu).

https://doi.org/10.1016/j.taml.2026.100734
Received 4 June 2026; Received in revised form 11 August 2026; Accepted 16 August 2026
Available online 4 September 2026

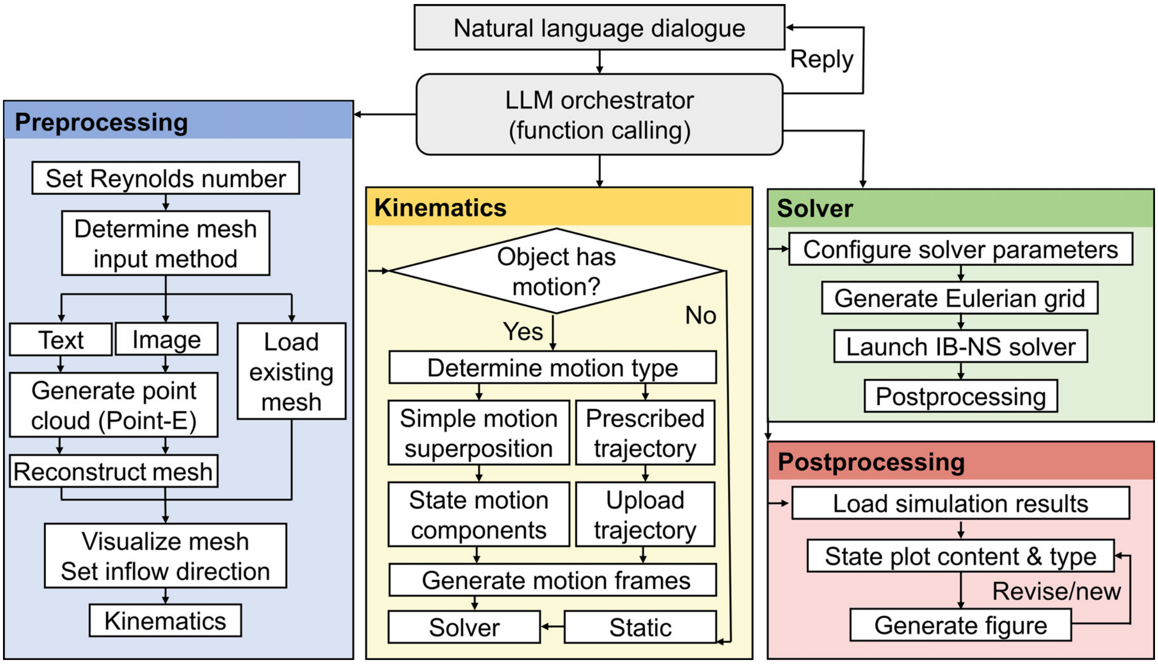

Fig. 1. Architecture of CFDAgent. The user interacts with CFDAgent through natural-language dialogue, while an LLM orchestrator dispatches tasks to the individual modules via function calling. The preprocessing module sets the Reynolds number and inflow direction, and acquires the body geometry through one of three routes: Point-E-based text-to-3D generation, Point-E-based image-to-3D generation, or direct mesh upload. The kinematics module bypasses motion generation for stationary bodies and produces a single-period Lagrangian motion buffer for moving bodies from either a superposition of analytical motion components or a prescribed trajectory. The solver module configures and launches the IB–Navier–Stokes (IB–NS) solver. The postprocessing module supports figure generation and iterative refinement.

conditions. The key advantage of the IB method for dialogue-driven moving-boundary simulations is the geometric decoupling of the moving Lagrangian surface from the fixed Eulerian grid. Consequently, the agent can readily define and revise kinematic models by updating only the Lagrangian boundary data, leaving the Eulerian grid and solver kernel unchanged and avoiding repeated body-fitted mesh generation.

The previous CFDAgent prototype [3] automated preprocessing, execution of the IB solver, and postprocessing for flows around stationary bodies. The present work extends the framework to prescribed moving and deforming boundaries by introducing a general kinematics module. The module converts either analytical motion components specified in natural language or measured trajectories into a reusable single-period Lagrangian motion buffer. The IB solver reads the buffer cyclically, allowing prescribed motions to be defined and revised without recompiling the solver. The extended framework is validated using a transversely rotating sphere, tandem flapping plates, and Pallas's long-tongued bat with measured wing trajectories, and is then applied to investigate the aerodynamic effects of progressive hindwing damage in a painted lady butterfly. All four cases are configured and executed through the dialogue-driven workflow. The current scope is limited to prescribed boundary kinematics in incompressible flows and does not yet include two-way fluid–structure interaction.

2. Methods

2.1. Overview: CFDAgent architecture

CFDAgent is an LLM-orchestrated CFD platform organized into four modules (Fig. 1). A natural-language dialogue with the user is processed by a large language model acting as a conversational orchestrator. The CFDAgent workflow is not tied to a specific LLM backbone, thus allowing different large language models to be used. The preprocessing module first sets the Reynolds number *Re*, then acquires the body geometry through one of three routes: text-to-3D or image-to-3D generation through Point-E [27] followed by mesh reconstruction, or direct upload of an existing surface mesh. The preprocessing module finally sets the freestream inflow direction relative to the body. The kinematics module builds body motions described either as a superposition of harmonic translation and rotation components in natural language, or as a Fourier-series fit of measured joint trajectories. For the special case of a stationary body with no prescribed motion, the module bypasses motion generation. The output of the kinematics module is a single-period Lagrangian motion buffer (a sequence of surface meshes spanning one motion period) that the solver reads cyclically, as detailed in Section 2.2. The solver module configures the parameters, generates the Cartesian Eulerian grid, and launches an in-house MPI-parallel finite-volume code that solves the three-dimensional unsteady incompressible NS equations. Before each launch, the complete simulation setup is presented to the user for final confirmation, and the solver is started only after explicit approval. In the IB method [26], the Eulerian velocity is interpolated to the Lagrangian mesh, whose nodal positions are supplied by the buffer. The body force on the Lagrangian mesh is computed based on the no-slip and no-penetration boundary conditions, then spread back to the Eulerian grid via a discrete delta function. It is added as a source term in the NS momentum equation to compute the flow. The postprocessing module loads the simulation outputs and produces publication-grade figures through the same dialogue interface, supporting both iterative refinement of an existing plot and the creation of new plots.

A previous prototype of CFDAgent [3] dispatched modules through prompt-based tag parsing. The present version replaces this with the LLM's native *function-calling* interface. Each module provides the LLM with a catalog of permitted operations, each specified by a name, description, and typed parameter schema. At each turn, the LLM may either reply in ordinary language or emit one or more structured calls that identify an operation and supply its arguments. CFDAgent then interprets each call and executes the corresponding application logic. This yields more robust dispatch, eliminates fragile string parsing, and

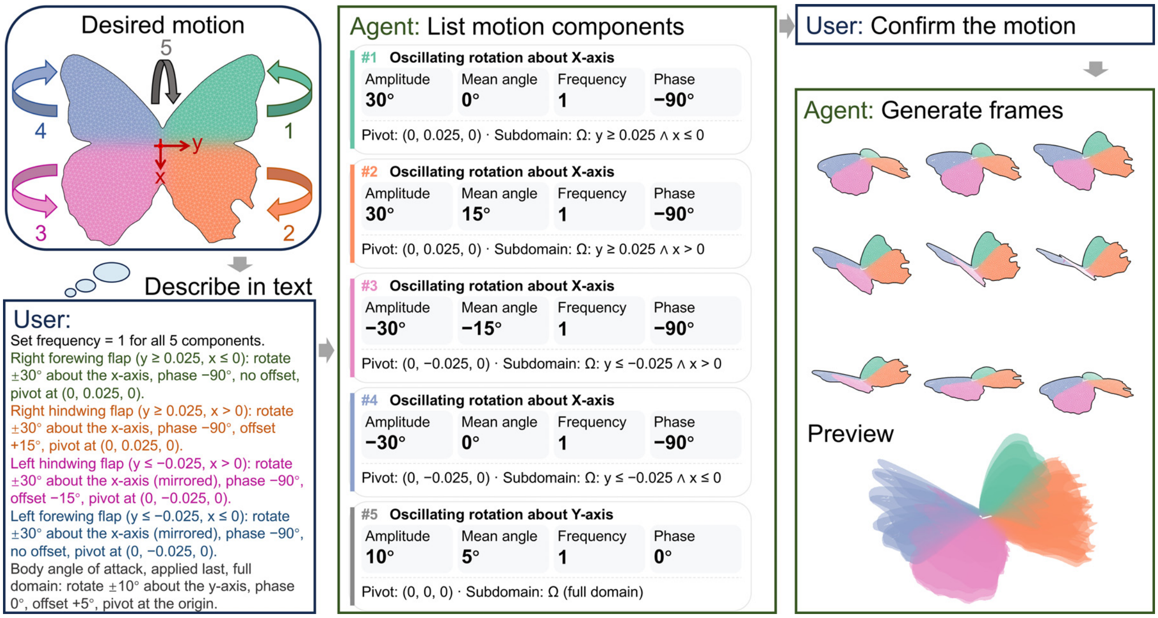

Fig. 2. Workflow of the kinematics module for a painted lady butterfly, with the prescribed motion specified in terms of five components. Given the reference surface mesh, the user describes the desired motion in natural language, listing the five components with their pivots and subdomains. The agent parses the description into structured motion components (amplitude, mean angle, frequency, phase, pivot, and subdomain). After user confirmation, the agent samples a sparse subset of frames from the single-period Lagrangian motion buffer and renders an animated preview in the user interface (UI).

supports multi-turn negotiation within each module-specific dialogue. For example, the user can interactively adjust the kinematic parameters through dialogue. All four cases reported in Section 3 were carried out end-to-end through dialogue, from the natural-language specification of the body and its motion, through the launch of the solver, to the generation of all figures presented.

2.2. Kinematics module

The kinematics module translates a dialogue-described body motion into a single-period Lagrangian motion buffer for the IB solver. The buffer comprises a sequence of surface meshes spanning one motion period, as illustrated in the agent architecture in Fig. 1 and the workflow in Fig. 2. For a moving body, the user has two options. The first option is a natural-language superposition of analytical components, including sinusoidal translation, sinusoidal rotation, and constant-rate rotation, which are defined along user-specified axes. For this option, the parsed motion-component parameters are displayed for user inspection and revision through dialogue. The parameters must be explicitly confirmed before the buffer is generated. The other option is to input a file containing measured joint trajectories that CFDAgent fits using a Fourier series. Both options generate the buffer $\boldsymbol{X}_{\text{body}}$, which comprises $N_T = \lceil T/\Delta t \rceil$ Lagrangian surface meshes spanning one motion period T. The user-supplied timestep Δt is automatically rounded to $\Delta t' = T/N_T$ so that $T/\Delta t'$ is exactly an integer and the buffer wraps with zero phase drift. The IB solver indexes the buffer cyclically so that the required motion-buffer storage is independent of the number of simulated cycles. With $\Delta t \sim 10^{-3}$ and $T \sim 1$, this yields $N_T \sim 10^3$ frames per case. A sparse 60-frame subset is additionally exported from the completed buffer for the UI motion preview. After inspecting the preview, the user either confirms the buffer and proceeds to solver setup or revises the motion specification and component parameters as needed and regenerates the buffer. This pure-data handoff (1) decouples motion design from solver compilation, (2) keeps motion storage independent of run length, (3) supports visual verification and revision before launch, and (4) confines any LLM error to an inspectable artifact rather than executable code.

Table 1
Comparison of the mean drag and lift coefficients for the transversely rotating sphere at $Re = 100$ and $\Omega^* = 1$. Methods: IB-FVM, immersed-boundary finite-volume method; FC, Fourier–Chebyshev spectral collocation; and BF-FVM, body-fitted finite-volume method.

Source	Method	$\overline{C_d}$	$\overline{C_l}$
Present study (CFDAgent, $h = D/40$)	IB-FVM	1.39	0.74
Giacobello et al. (2009) [28]	FC	1.38	0.72
Kim (2009) [29]	IB-FVM	1.38	0.72
Niazmand & Renksizbulut (2003) [30]	BF-FVM	1.25	0.58

3. Results

3.1. Transversely rotating sphere

A fundamental test of the kinematics module is the prescribed rigid-body rotation of a closed surface. We adopt the canonical transversely rotating sphere benchmark of Giacobello et al. [28] at the Reynolds number $Re = U_\infty D/\nu = 100$ and the dimensionless rotation rate $\Omega^* = \Omega D/(2U_\infty) = 1$, where D is the sphere diameter, U_∞ the freestream velocity, ν the kinematic viscosity, and Ω the prescribed angular speed. The sphere is centered at the origin in a freestream and rotates about the $\hat{y}$ axis, generating a steady lift in the $+\hat{z}$ direction that is associated with the Magnus effect. All variables in this study are nondimensionalized; in particular, the Q-criterion scalar Q and the fluid-vorticity vector $\boldsymbol{\omega}$ are nondimensionalized using the characteristic length and the freestream velocity. At this Reynolds number the flow is laminar and steady, allowing direct comparison of time-averaged drag and lift coefficients against three independent published simulations [28–30].

We generate the Lagrangian mesh using a prompt “sphere”. The kinematics module emits a single-period Lagrangian motion buffer in which the sphere mesh rotates about $\hat{y}$. The flow is computed using the IB–NS solver on a uniform Cartesian Eulerian grid with spacing $h = D/40$, and the simulation is advanced to a dimensionless time of $tU_\infty/D = 20$. The time-averaged values reported in Table 1 are $\overline{C_d} = 1.39$ and $\overline{C_l} = 0.74$. The present values differ from the results of both Giaco-

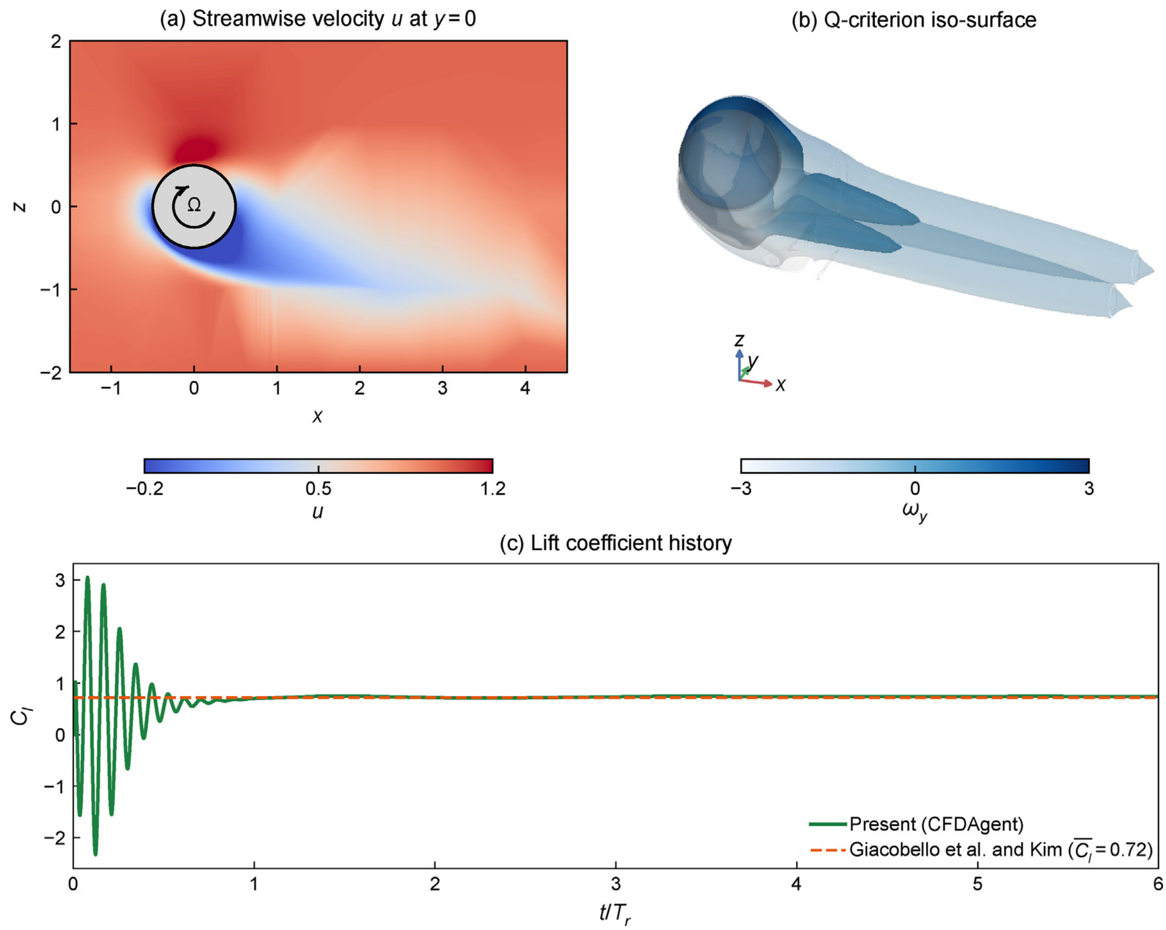

Fig. 3. Transversely rotating sphere at $Re = 100$ and $\Omega^* = 1$: (a) streamwise velocity u on the plane $y = 0$, with the sphere shown in gray, outlined in black, and rotating clockwise when viewed from $-\hat{y}$; (b) Q-criterion isosurfaces at $Q = 0.15$ and $Q = 0.8$, colored by the y-component of vorticity ω_y, showing the steady double-threaded wake; and (c) time history of the lift coefficient C_l over the first six rotational periods after the impulsive start. The rotational period is defined as $T_r = 2\pi/\Omega$, and the dashed line denotes the reference mean lift coefficient $\overline{C_l} = 0.72$ reported by Giacobello et al. [28] and Kim [29].

bello et al. [28] and Kim [29] by approximately 0.7% in $\overline{C_d}$ and 2.8% in $\overline{C_l}$, indicating close quantitative agreement. The larger discrepancy compared to Niazmand & Renksizbulut [30] is attributable to a simplified surface-velocity boundary condition in their work, as noted by Giacobello et al. [28].

Figure 3(a) plots the streamwise velocity u on the plane $y = 0$. The characteristic asymmetric distribution is evident. At the top of the sphere ($z = +D/2$), the rigid-body surface velocity is aligned with the freestream. This accelerates the boundary-layer flow above the sphere to achieve $u > 1.1$. At the bottom ($z = -D/2$), the surface moves toward $-\hat{x}$ and the counter-aligned shear decelerates the flow, producing reversed streamwise velocity.

Figure 3(b) renders the Q-criterion isosurfaces of 0.15 and 0.8, colored by the y-component of vorticity, ω_y. The wake is steady and exhibits the characteristic double-threaded structure. Figure 3(c) shows the time history of the lift coefficient over the first six rotational periods following the impulsive start. The rotational period is defined as $T_r = 2\pi/\Omega$. The dashed line denotes the mean lift coefficient $\overline{C_l} = 0.72$ reported in both reference studies [28,29]. After the initial transient, C_l settles near the reference value and exhibits only small residual fluctuations over the remainder of the simulation. Together, the quantitative agreement of the lift and drag coefficients with the reference values (Table 1) and the qualitative reproduction of the characteristic wake topology demonstrate that the CFDAgent framework accurately reproduces the benchmark results of prescribed rigid-body rotation, indicating the framework's potential for engineering simulations involving rotating bodies [31–34].

3.2. *Tandem flapping plates*

Before applying CFDAgent to the complex wing geometries of real flying animals, we validate the complete kinematics pipeline using a published flapping benchmark with simple analytical kinematics. We adopt the butterfly-inspired model of Chen et al. [17]. The forewing ($x \in [-0.5, -0.025]$) and hindwing ($x \in [0.025, 0.5]$) are represented by two tandem rectangular flat plates that together span an overall chord $c = 1$ and are separated by a spanwise slot of width $0.05c$. The forewing–hindwing pair is mirrored across the body centerline to form the opposite wing, and a central body strip of width $0.05c$ represents the body. The prescribed harmonic wing motion is defined by a common forewing–hindwing elevation angle $\Psi(t) = -\Psi_m \cos(2\pi f t)$ and a time-dependent body angle of attack $\mathrm{AoA}(t) = \mathrm{AoA}_0 + \mathrm{AoA}_m \sin(2\pi f t)$. The parameters are $\Psi_m = 40°$, $\mathrm{AoA}_0 = \mathrm{AoA}_m = 15°$, the Strouhal number $St = 0.3$, and the Reynolds number $Re = 300$. The dihedral angle is set to $\beta = 0$.

Following the methodology in Section 2.2, a natural-language description of the harmonic motion is parsed to generate a single-period Lagrangian motion buffer. The computational domain and numerical settings match those used by Chen et al. [17]: the domain is $[-6c, 12c] \times [-9c, 9c] \times [-9c, 9c]$, the Eulerian grid is locally refined to a minimum spacing of $h = c/40$, and the time step is $\Delta t = 0.01$. The solver advances the flow solution for three full wingbeats from rest, and the results are evaluated entirely in the postprocessing module. The cycle-averaged lift coefficient over the third wingbeat is $\overline{C_l} = 0.24$, agreeing with the value of 0.24 reported by Chen et al. [17]. In Fig. 4(a), the present curve agrees

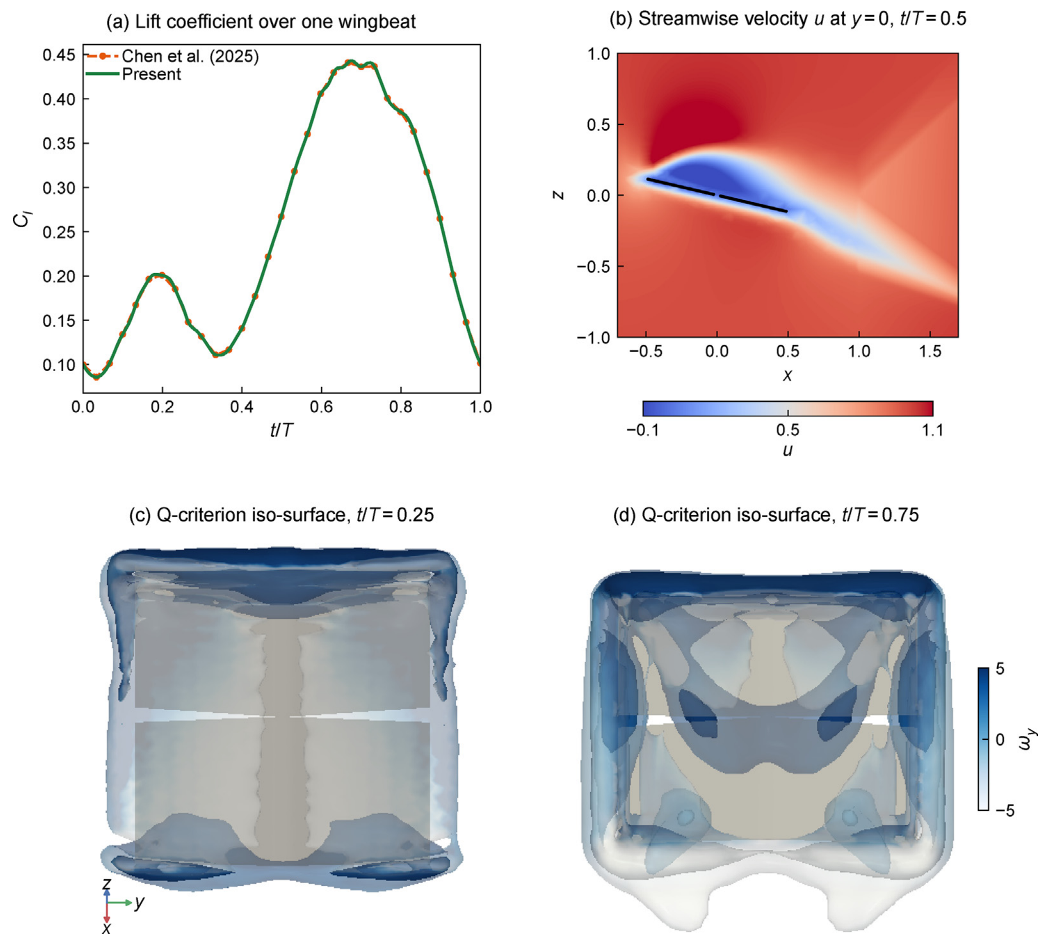

Fig. 4. Validation against the butterfly-inspired tandem-plate model of Chen et al. [17]: (a) lift-coefficient history over the third wingbeat, with the present result shown by the solid line and the reference data by the dashed line; (b) streamwise velocity u in the plane $y = 0$ at $t/T = 0.5$, with the body shown as a black line; and (c,d) Q-criterion isosurfaces at $Q = 3.0$ and $Q = 10.0$, colored by ω_y, at $t/T = 0.25$ (mid-upstroke) and $t/T = 0.75$ (mid-downstroke), respectively. All quantities are nondimensional, and the common viewing orientation of panels (c) and (d) is indicated by the coordinate triad in panel (c).

closely with the reference data reported by Chen et al. [17] throughout the cycle, validating the kinematics pipeline and its coupling to the solver.

The time-resolved $C_l(t/T)$ trace in Fig. 4(a) reproduces the characteristic double-peaked structure reported by Chen et al. [17]. A smaller upstroke peak ($C_l \approx 0.21$ at $t/T \approx 0.2$) is followed by the main downstroke peak ($C_l \approx 0.44$ at $t/T \approx 0.7$). The streamwise velocity contours in the body-symmetry plane $y = 0$ at $t/T = 0.5$, shown in Fig. 4(b), reveal the flow around the inclined body. The body, shown as a black line, is oriented at the instantaneous angle of attack of approximately 15°. A high-velocity region appears above the leading edge, while an inclined low-velocity region extends downstream into the wake.

Figure 4(c) and (d) show Q-criterion isosurfaces at $Q = 3.0$ and $Q = 10.0$, colored by ω_y, at two opposite phases of the stroke cycle. At mid-upstroke ($t/T = 0.25$), when the angle of attack reaches its maximum value of approximately 30° (Fig. 4(c)), a strong LEV remains attached to the wing surface, accompanied by a pair of coherent tip-vortex structures. At mid-downstroke ($t/T = 0.75$), when the angle of attack is approximately 0° (Fig. 4(d)), the wing is nearly parallel to the freestream; the LEV appears detached from the surface and displaced downstream, while the tip-vortex structures alter but persist. The LEV and tip-vortex structures at these two phases are qualitatively consistent with those reported by Chen et al. [17]. Overall, the present results agree with the reference benchmark in both the lift history and vortex topology.

3.3. *Painted lady butterfly with progressive hindwing damage*

The painted lady butterfly (*V. cardui*) undertakes long-distance seasonal migrations, and field observations have documented frequent damage of the hindwings [23]. The aerodynamic effects of such naturally occurring damage have not yet been quantified. We use CFDAgent to address this gap by analyzing an intact baseline geometry and three representative damaged geometries classified as mild, moderate, and severe. In the three damaged geometries, damage is confined to the right hindwing. The removed area, normalized by the planform area of the intact right hindwing, is approximately 10%, 30%, and 60% for the mild, moderate, and severe cases, respectively.

For each case, we upload the wing mesh and specify the wingbeat motion in natural language. The prescription defines the forewing elevation angle Ψ_f, the hindwing elevation angle Ψ_h, and the body angle of attack AoA as harmonic functions of dimensionless time, with the parameter values following Chen et al. [17]. The resulting single-period Lagrangian motion buffer is sent to the IB–NS solver. The solver outputs are then postprocessed entirely through CFDAgent's dialogue interface to generate the force-plot and flow-visualization panels in Fig. 5.

Figure 5 (a) plots the lift coefficient over the last wingbeat cycle for the four cases. All curves retain a similar temporal shape, with a single peak near mid-downstroke, but the peak amplitude decreases progressively as damage increases. The cycle-averaged lift coefficient $\overline{C_l}$ in Fig. 5(b) drops from 0.15 for the intact case to 0.14, 0.13, and 0.11 for the mild, moderate, and severe cases, respectively, corresponding to lift

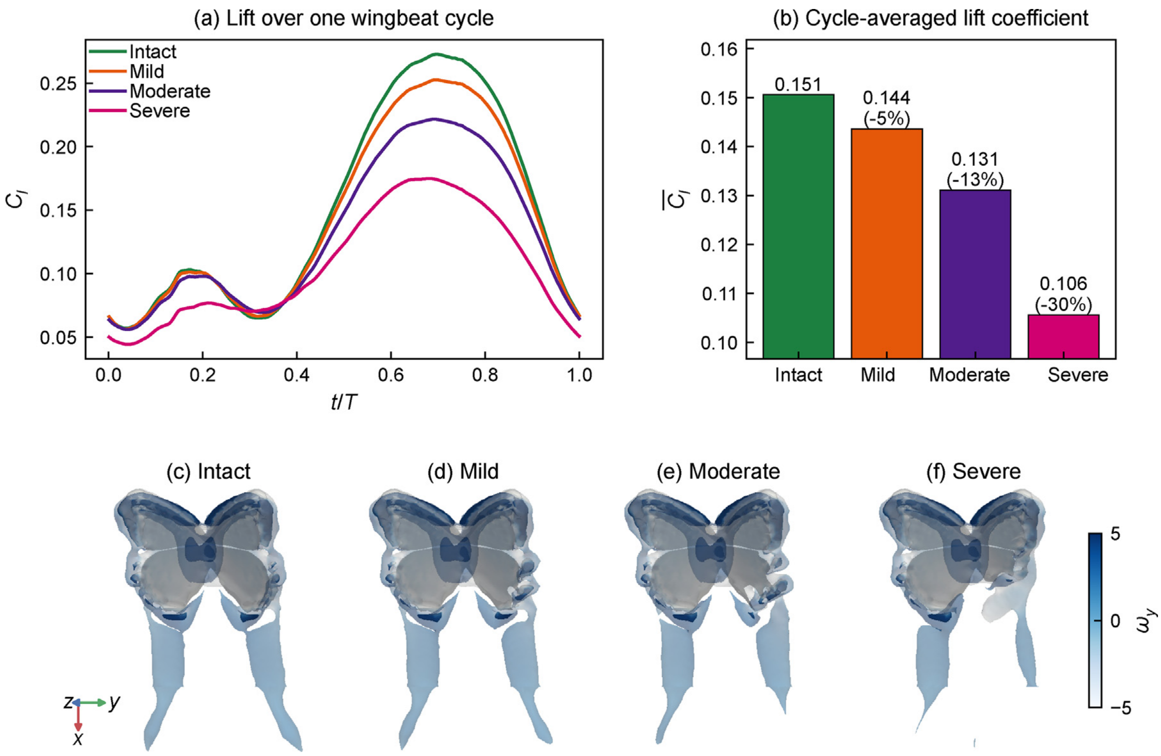

Fig. 5. Painted lady butterfly (*V. cardui*) with progressive right-hindwing damage: (a) lift-coefficient histories during the last wingbeat for the intact, mild, moderate, and severe cases; (b) the corresponding cycle-averaged lift coefficients; and (c–f) top views of the *Q*-criterion isosurfaces, colored by ω_y for all cases with the coordinate triad in panel (c).

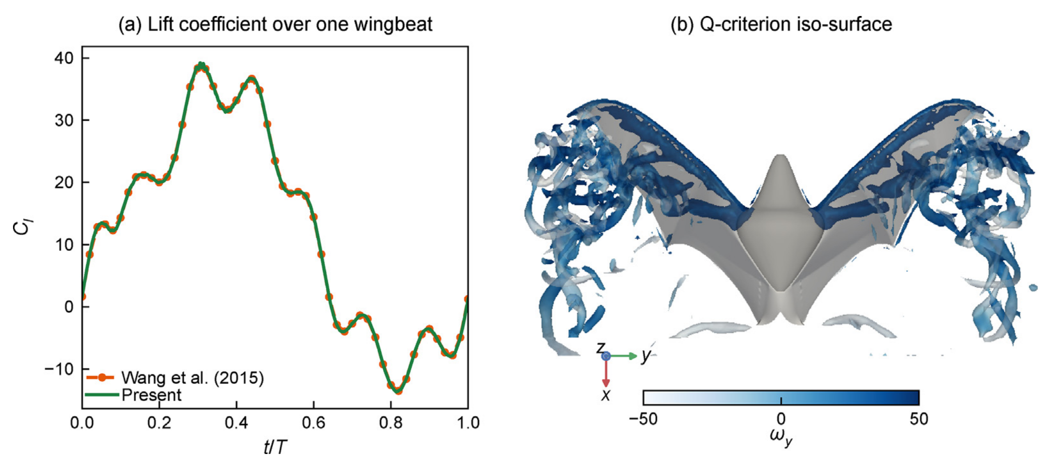

Fig. 6. Pallas's long-tongued bat in forward flight: (a) lift coefficient over one wingbeat cycle; (b) *Q*-criterion isosurface at the downstroke.

reductions of 5%, 13%, and 30% based on the unrounded values. Panels (c)–(f) of Fig. 5 show top-down views of the *Q*-criterion isosurfaces colored by ω_y. In the intact case, the vortical structures are approximately symmetric about the body centerline, with elongated streamwise wake structures extending downstream on both sides. Mild damage introduces only localized changes on the damaged side, whereas moderate and severe damage progressively reduce the extent and coherence of the vortical structures downstream of the damaged right hindwing, resulting in an increasingly asymmetric wake. This progressive loss of coherence is consistent with a weakening of the interaction between the tip vortices and the secondary vortices shed from the trailing edge, thereby reducing the associated lift augmentation. The trend is consistent with field reports that mildly damaged painted lady butterflies continue to migrate successfully, whereas extensive hindwing-area loss is expected to impair flight performance and may reduce survival [23].

3.4. Pallas's long-tongued bat in flapping flight

Bats are the only vertebrate fliers with highly articulated membrane wings. The wing consists of an elastic membrane supported by a multi-jointed skeletal framework comprising the shoulder, elbow, wrist, and digits. Unlike the butterfly cases, this case uses the measured-trajectory option of the kinematics module. Following previous research [21,35], we simulate Pallas's long-tongued bat in slow flight.

We upload the bat surface mesh and measured trajectories [35] of the wingtip, wrist, and fifth-digit. The simulation settings are the same as those of Wang et al. [21], with CFL < 0.7 and $Re = 300$. Following Wang et al. [21], we nondimensionalize the lift as $C_l(t) = L(t)/[\rho U_\infty^2 S_{\text{avg}}/2]$. Here, $L(t)$ is the total lift acting on the bat, U_∞ is the forward-flight velocity, and S_{avg} is the wing area averaged over one flapping period.

Figure 6(a) plots the lift coefficient over one wingbeat cycle. The present result agrees closely with the data reported by Wang et al.

[21]. The trace shows a strong positive peak $C_l \approx 38$ at mid-downstroke, a negative excursion $C_l \approx -13$ at mid-upstroke, and a cycle-averaged $\overline{C_l} = 12.7$. Figure 6(b) renders the Q-criterion isosurface of $Q = 400$, colored by ω_y, revealing that the LEV acts as the primary lift-generating mechanism [20].

4. Conclusions

CFDAgent with a general kinematics module has been presented as an LLM-orchestrated CFD platform built upon two design elements. The first is a kinematics module that converts natural-language motion descriptions or measured trajectories into a single-period Lagrangian motion buffer. The second is an end-to-end dialogue-driven workflow that spans geometry intake, solver launch, and publication-quality figure generation. This workflow uses the LLM's function-calling interface, replacing the prompt-based tag parsing used in earlier prototypes. The framework was validated using three benchmark cases: a transversely rotating sphere, the butterfly-inspired tandem-plate model with prescribed motion, and the Pallas's long-tongued bat case with uploaded measured trajectories. The agreement with the corresponding reference results demonstrates the accuracy and versatility of the current CFDAgent pipeline. The framework was further applied to a painted lady butterfly with progressive hindwing damage. The results show that increasing hindwing damage progressively reduces lift and produces a more asymmetric wake, demonstrating the applicability of CFDAgent to realistic biological flapping-flight problems. The current framework still relies on an adequate computational mesh together with complete, accurate user specifications. Two-way fluid–structure interaction, whereby aerodynamic loads drive structural deformation that feeds back into the flow, is not yet supported.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Zhaoyue Xu: Writing – original draft, Visualization, Validation, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Hua-Dong Yao:** Writing – review & editing, Writing – original draft, Validation, Supervision, Resources, Project administration, Formal analysis, Conceptualization. **Shizhao Wang:** Writing – review & editing, Visualization, Validation, Supervision, Software, Resources, Project administration, Funding acquisition, Formal analysis, Data curation. **Guowei He:** Writing – review & editing, Project administration, Funding acquisition, Formal analysis, Conceptualization.

Acknowledgments

This work was supported by the Swedish Transport Administration (Trafikverket) through the research project "Digital aerodynamics prediction and Stability safety for enhancing Wingsail integrity under instantaneous dynamic ship Motions (SafeWinds)" (Project ID: TRV2024/98491).

References

[1] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, et al., Language models are few-shot learners, in: Advances in Neural Information Processing Systems, volume 33, 2020, pp. 1877–1901.
[2] S. Wolfram, Can AI Solve Science?, Stephen Wolfram Writings, 2024. https://writings.stephenwolfram.com/2024/03/can-ai-solve-science
[3] Z. Xu, L. Wang, C. Wang, Y. Chen, Q. Luo, H.-D. Yao, S. Wang, G. He, CFDAgent: a language-guided, zero-shot multi-agent system for complex flow simulation, Phys. Fluids 37 (2025) 117124.
[4] K. Duraisamy, G. Iaccarino, H. Xiao, Turbulence modeling in the age of data, Annu. Rev. Fluid Mech. 51 (2019) 357–377.
[5] Z. Xu, X. Zhang, S. Wang, G. He, Artificial neural network based response surface for data-driven dimensional analysis, J. Comput. Phys. 459 (2022) 111145.
[6] S. Pandey, R. Xu, W. Wang, X. Chu, OpenFOAMGPT: a retrieval-augmented large language model (LLM) agent for openFOAM-based computational fluid dynamics, Phys. Fluids 37 (2025) 035120.
[7] W. Wang, R. Xu, J. Feng, Q. Zhang, S. Pandey, X. Chu, A status quo investigation of large-language models for cost-effective computational fluid dynamics automation with openFOAMGPT, Theor. Appl. Mech. Lett. 15 (2025) 100623.
[8] L. Yue, P. Xu, T. Zhang, N. Somasekharan, S. Pan, FoamGPT: fine-tuning large language model for agentic automation of CFD simulations with openFOAM, Machine Learning and the Physical Sciences Workshop, NeurIPS, 2025.
[9] X. Zhang, Z. Xu, G. Zhu, C.M.J. Tay, Y. Cui, B.C. Khoo, L. Zhu, Using large language models for parametric shape optimization, Phys. Fluids 37 (2025) 083601.
[10] J. Du, Y. Wu, P. Zhao, Y. Liu, M. Zhang, X. Xu, X. Zhang, TurboAgent: an LLM-driven autonomous multi-agent framework for turbomachinery aerodynamic design, 2026, https://arxiv.org/abs/2604.06747
[11] D.Z. Peng, B. Liu, W. Jiang, H. Peng, C. Huang, T.K.T. Tse, J. Zhang, Vortex state transitions in deep street canyons enabled by an automated large language model workflow, Phys. Fluids 38 (2026) 055119.
[12] Z. Dong, S. Du, Z. Lu, Y. Yang, CFD-copilot: leveraging domain-adapted large language model and model context protocol to enhance simulation automation, Chin. J. Aeronaut. (2026) 104321.
[13] A. Cheng, L. Zhang, G. He, Re4: scientific computing agent with rewriting, resolution, review and revision, in: International Conference on Learning Representations, 2026.
[14] P.-Z. Ma, G. Gai, J.-K. Li, Z.-H. Luo, L.T. Zhu, Large language models for automating computational fluid dynamics (CFD): from predictive modeling and optimization to execution scheduling, Smart Chem. Eng. 2 (1) (2026) 3.
[15] H. Liu, S. Wang, T. Liu, Vortices and forces in biological flight: insects, birds, and bats, Annu. Rev. Fluid Mech. 56 (2024) 147–170.
[16] C. Wang, Z. Xu, X. Zhang, S. Wang, Optimal reduced frequency for the power efficiency of a flat plate gliding with spanwise oscillations, Phys. Fluids 33 (2021) 111908.
[17] Y. Chen, Y. Liu, S. Wang, Streamwise ram effect and tip vortex enhance the lift of a butterfly-inspired flapping wing, J. Fluid Mech. 1005 (2025) A13.
[18] C. Wang, Z. Xu, S. Wang, G. He, Gap vortices compensate for the lifting-surface loss of a moult-gapped flapping wing, J. Fluid Mech. 1019 (2025) A19.
[19] C.L. Roy, D. Amadori, S. Charberet, J. Windt, F.T. Muijres, V. Llaurens, V. Debat, Adaptive evolution of flight in *Morpho*butterflies, Science 374 (2021) 1158–1162.
[20] F.T. Muijres, L.C. Johansson, R. Barfield, M. Wolf, G.R. Spedding, A. Hedenström, Leading-edge vortex improves lift in slow-flying bats, Science 319 (2008) 1250–1253.
[21] S. Wang, X. Zhang, G. He, T. Liu, Lift enhancement by bats' dynamically changing wingspan, J. R. Soc. Interface 12 (2015) 20150821.
[22] A. Hedenström, Effects of wing damage and moult gaps on vertebrate flight performance, J. Exp. Biol. 226 (2023) jeb227355.
[23] R. Korkmaz, H. Rajabi, S. Eshghi, S.N. Gorb, T.H. Büscher, The frequency of wing damage in a migrating butterfly, Insect Sci. 30 (2023) 1507–1517.
[24] C.W. Hirt, A.A. Amsden, J.L. Cook, An arbitrary Lagrangian–Eulerian computing method for all flow speeds, J. Comput. Phys. 14 (1974) 227–253.
[25] M. Uhlmann, An immersed boundary method with direct forcing for the simulation of particulate flows, J. Comput. Phys. 209 (2005) 448–476.
[26] S. Wang, X. Zhang, An immersed boundary method based on discrete stream function formulation for two- and three-dimensional incompressible flows, J. Comput. Phys. 230 (2011) 3479–3499.
[27] A. Nichol, H. Jun, P. Dhariwal, P. Mishkin, M. Chen, Point-e: a system for generating 3D point clouds from complex prompts, 2022, https://arxiv.org/abs/2212.08751
[28] M. Giacobello, A. Ooi, S. Balachandar, Wake structure of a transversely rotating sphere at moderate Reynolds numbers, J. Fluid Mech. 621 (2009) 103–130.
[29] D. Kim, Laminar flow past a sphere rotating in the transverse direction, J. Mech. Sci. Technol. 23 (2009) 578–589.
[30] H. Niazmand, M. Renksizbulut, Surface effects on transient three-dimensional flows around rotating spheres at moderate Reynolds numbers, Comput. Fluids 32 (2003) 1405–1433.
[31] M. Yin, R. Ji, J. Zheng, K. Sun, J. Zhang, Y. Cheng, Y. Zhang, R. Reabroy, Energy-harvesting dynamics analysis of a vertical-axis tidal turbine considering multi--degree-of-freedom motion interaction effects using the lattice Boltzmann method, Renew. Energy 266 (2026) 125632.
[32] R. Ji, M. Wu, J. Zheng, K. Sun, J. Zhang, M.-A. Xue, R. Reabroy, Y. Zhang, Lagrangian dynamic large-eddy simulation of the performance of a horizontal-axis tidal turbine with an actuator-line method, Phys. Fluids 37 (2025) 075169.
[33] M. Yin, R. Ji, R. Zhu, S. Xu, K. Sun, J. Zhang, Y. Zhang, R. Reabroy, Performance and wake prediction of a ducted tidal stream turbine in yaw misalignment using the lattice Boltzmann method, Energy Convers. Manag. 347 (2026) 120574.
[34] Y. Ma, Z. Xu, H.D. Yao, Region-aware hybrid LSTM-GRU modeling for wake flows induced by offshore wind-turbine foundations, Results Eng. 29 (2026) 109518.
[35] M. Wolf, L.C. Johansson, R. von Busse, Y. Winter, A. Hedenström, Kinematics of flight and the relationship to the vortex wake of a Pallas' long tongued bat (*Glossophaga soricina*), J. Exp. Biol. 213 (2010) 2142–2153.