Towards Autonomous Numerical Wind Tunnel Testing: A Domain Knowledge-Enhanced LLM-CFD Framework for Structural Aerodynamics

Baoheng Li [a], Teng Wu [b]

[a]The State University of New York at Buffalo, Buffalo, NY, USA, baohengl@buffalo.edu
[b]The State University of New York at Buffalo, Buffalo, NY, USA, tengwu@buffalo.edu

SUMMARY

Computational fluid dynamics (CFD) provides a numerical wind tunnel for structural aerodynamics, complementing physical tests with full-field information and low cost. Yet the complexity of CFD workflows makes automation essential to reduce manual effort and standardize practice. Existing CFD software and large language model (LLM) assisted CFD frameworks mainly automate numerical implementation, generally remaining at level 3–5 in level of automation (LoA) and level 1–3 in ISA-95/IEC 62264 enterprise–control system integration levels, and thus cannot autonomously formulate engineering problems, interpret design intent, or ensure code-consistent modelling. This study proposes a domain-knowledge-enhanced LLM–CFD framework with two core components. Specifically, a problem formulation module constructs code-consistent physical models from structural and wind-environment information, and a meta-decision layer evaluates scheme credibility and triggers on-demand human intervention. The framework attains LoA level 7 and ISA-95/IEC 62264 level 4, formulating an end-to-end autonomous numerical wind tunnel.

Keywords: *Computational fluid dynamics, Autonomous, Large language model, Fine-tuning, Domain knowledge*

1. INTRODUCTION

Structural aerodynamics provides the physical basis for understanding wind effects on buildings and bridges. Wind tunnel testing has offered the most reliable wind load information and forms the backbone of many design codes, but full campaigns are expensive and labor-intensive. A typical test program requires model fabrication, wind-field simulation, instrumentation, data acquisition, and extensive post-processing, often taking weeks or months and demanding close collaboration among specialists, which limits their wide-scale use in routine architectural design or iterative optimization. Computational fluid dynamics (CFD) constructs a numerical wind tunnel by solving the Navier–Stokes equations in a virtual domain that embeds the structural geometry. It can deliver full-field velocity (with needed turbulence statistics) and pressure at reduced cost and is now widely used as a complement or, in some cases, an alternative to physical testing.

However, CFD remains difficult to operate in an engineering-reliable way. Running a simulation requires deep knowledge of fluid mechanics and turbulence modelling, together with expertise in meshing, numerical schemes, boundary conditions, solver control, and verification procedures such as grid-independence studies. The workflow spans problem formulation, case set-up, monitoring, post-processing and validation, and poor choices at any step can lead to divergence or non-physical results. Consequently, high-quality CFD remains accessible only to experts, highly dependent on individual experience, and challenging to standardize or reproduce across projects. To address this issue, the community has long sought to automate parts of the CFD workflow so that non-experts can focus on engineering decisions rather than numerical details. In this context, automation is understood as the capability of a system to execute tasks and make decisions under predefined or learned logic, allowing human users to move from hands-on operation to supervisory roles. With the rapid development of artificial intelligence (AI) technologies, automation now includes not only scripted operations but also reasoning, task planning, and cross-module coordination. To rigorously evaluate how automated a CFD system is, this study adopts the level of automation (LoA) framework proposed by Vagia et al. (2016), which grades systems according to how cognitive responsibilities are shared between humans and machines, from fully manual operation to fully autonomous control. The eight LoA levels used in this work are summarized in Table 1. A complementary perspective is provided by the ISA-95/IEC 62264 enterprise–control system integration levels (Unver 2013), which classifies automation according to task scale and system scope, from local process control to production planning, as summarized in Table 2. Together, these two hierarchies form a two-dimensional grading scheme that characterizes both the intelligence of the automation and the extent of the workflow it covers. It offers an objective basis for identifying genuine advances in system maturity and for a systematical comparison framework among various methods.

Table 1: Level of automation proposed by Vagia et al. (2016).

Level	Description	Interpretation in CFD Automation
1	Manual control: computer offers no assistance.	Serve only as a numerical calculator.
2	Decision proposal stage: the computer offers some decisions to the operator. The operator is responsible to decide and execute.	Provide predefined modeling options for human selection (e.g., turbulence models), but all decisions and workflow execution are under human control.
3	Human decision select stage: the human selects one decision and the computer executes.	Execute predefined CFD workflows once the modeling choices are specified by the user.
4	Computer decision select stage: the computer selects one decision and executes with human approval.	Autonomously select modeling options and execute CFD workflows, but key decisions require human approval (e.g., turbulence model).
5	Computer execution and human information stage: the computer executes the selected decision and informs the human.	Autonomously make modeling decisions and execute complete CFD workflows without human intervention, and report results after completion.
6	Computer execution and on call human information stage: the computer executes the selected decision and informs the human only if asked.	Autonomously make modeling decisions and execute complete CFD workflows, with human involvement limited to on-demand queries.
7	Computer execution and voluntarily information stage: The computer executes the selected decision and informs the human only if it decides to inform.	Autonomously execute CFD workflows, assess credibility, and decide whether to involve human experts based on detected non-convergence, instability, or code violations.
8	Autonomous control stage: The computer does everything without human notification, except if an error that is not within the specifications occurs — in that case the computer needs to inform the operator.	Autonomously execute CFD workflows, self-correct most issues, and involve humans only for unrecoverable failures (e.g., unrealistic structural geometry).

Conventional CFD software already exhibits a certain degree of automation. Modern packages encapsulate solvers and numerical algorithms behind graphical user interfaces and scripting interfaces, and can automatically generate meshes, run solvers, and provide post-processing and visualization. Under the LoA framework, such tools correspond roughly to level 3, where the human defines the problem and selects the options, and the computer executes the prescribed actions. Under ISA-95/IEC 62264, they mainly operate at levels 2, where automation is restricted to local operations and cell- or line-level supervision within the CFD environment. Key engineering decisions, such as the choice of turbulence model, meshing strategy, and convergence criteria, remain entirely manual. As a result, conventional software provides execution-level automation but lacks the ability to interpret engineering intent.

Table 2: Enterprise–control system integration levels introduced by ISA-95/IEC 62264.

Level	Description	Interpretation in CFD Automation
0	Process control functions	Basic numerical computations (e.g., numerical solution of the governing equations), with all workflows and modeling decisions handled by the user.
1	Operations functions	CFD functions are started manually by the user within a single software tool. (e.g., meshing, solving, and post-processing)
2	Cell or line supervision functions	Automatic execution and supervision of predefined CFD workflows, while all modeling decisions remain manual.
3	Manufacturing operations management	Automatic execution CFD workflows across pre-processing, solving, and post-processing for executing predefined simulation problems.
4	Production planning, scheduling and business planning	Automatically plan CFD simulations based on design codes, coordinate multiple cases, and check results using different data sources.

Recently, large language models (LLMs) have been introduced into CFD workflows to further lower the expertise barrier (Dong et al. 2025; Pandey et al. 2025). Two main approaches have emerged. Prompt-engineering approaches use carefully designed natural-language instructions to generate input files, scripts, and log-analysis commands on demand, achieving plug-and-play automation without training. Fine-tuning approaches adapt open-source LLMs using domain data so that the model learns the structure of CFD inputs, common modelling patterns, and selected guideline-based rules. Compared to prompt engineering, fine-tuned domain-specific models can be deployed on enterprise servers or HPC clusters, offering improved data security, controllable cost, and stable behaviour, more stable performance under fixed tasks. Both strategies are used to automate the execution of CFD workflows, including pre- and post-processing and log analysis. These LLM–CFD systems generally can reach up to LoA level 5, where the computer makes and executes decisions and reports the outcome, and roughly level 3 in ISA-95/IEC 62264, where CFD simulations run predefined problems in an integrated workflow. Nevertheless, existing LLM–CFD frameworks primarily address the numerical execution of predefined problems, assuming that the physical scenario, inflow and boundary conditions, and target quantities of interest are specified a priori by human users. Consequently, the knowledge-intensive tasks of problem formulation under code constraints and engineering judgement remain largely manual.

This study proposes a domain-knowledge-enhanced LLM–CFD multi-agent framework for constructing a deployable numerical wind tunnel that enables end-to-end automation of aerodynamic assessment in structural wind engineering. Based on the existing LLM–CFD approaches, the framework consists of two tightly coupled components, namely a problem formulation module and a meta-decision layer. The overall workflow and interactions among these components, together with the execution modules, are schematically illustrated in Figure 1. The problem formulation module serves as the decision-making core. Given structural models and environmental information and operating under wind-engineering codes and best-practice literature, it autonomously constructs a computable physical problem. Specifically, the module determines computational-domain extent and mesh-resolution requirements, selects appropriate turbulence models and near-wall treatments, configures boundary conditions and scenario combinations, and defines aerodynamic performance metrics and post-processing procedures. In doing so, the system advances beyond executing user-specified settings to autonomously determining what should be simulated and the associated rationale. The meta-decision layer evaluates the credibility of automatically generated schemes and coordinating human–machine interaction. It checks each proposed case using three categories of information, namely compliance with code-based rules (e.g., domain dimensions, blockage ratio, sampling duration, and turbulence-model suitability), mesh-independence behavior assessed, and consistency with wind-tunnel databases or code-recommended design values under geometric and scenario similarity. If these checks indicate that the simulation lies within a trustworthy region supported by existing knowledge, the system proceeds autonomously to complete the simulation and post-processing. When conflicts arise, the system first attempts internal corrective actions; otherwise, the meta-decision layer triggers an on-call escalation to request expert review and suggested revisions. In this way, human experts are transformed from mandatory endpoints into on-demand supervisors. To support stable and interpretable reasoning, domain knowledge from wind-engineering guidelines, representative research, and wind-tunnel databases is encoded into machine-processable rule bases, which are used to generate training pairs with explicit reasoning chains. Supervised fine-tuning with lightweight adaptation techniques embeds this normative logic into open-source LLMs that can be deployed on enterprise servers or HPC clusters.

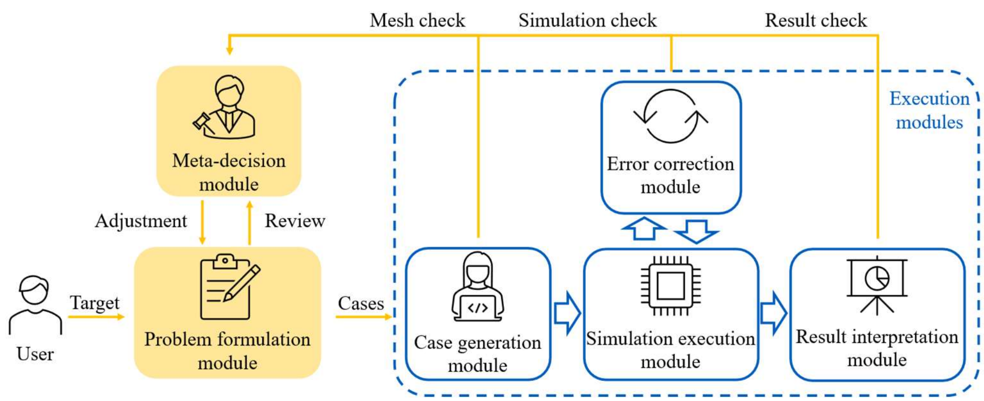

Figure 1: Domain knowledge-enhanced LLM-CFD framework workflow.

This framework reaches level 7 in the LoA, where the computer autonomously executes CFD workflows, assesses credibility, and decides whether to involve human experts based on detected non-convergence, instability, or code violations. This system extends automation to level 4 in ISA-95/IEC 62264, where code-based engineering task planning and cross-module coordination are performed. It integrates engineering requirements, physical modelling, numerical implementation, and result validation into a closed-loop workflow. Overall, the proposed domain-knowledge-enhanced LLM–CFD multi-agent framework shifts the focus from “how to run CFD” to “how to correctly define, implement, and validate CFD” simulations from an engineering perspective.

REFERENCES

Dong, Z., Lu, Z., Yang, Y., 2025. Fine-tuning a Large Language Model for Automating Computational Fluid Dynamics Simulations. Theoretical and Applied Mechanics Letters 15, 100594. https://doi.org/10.1016/j.taml.2025.100594

Pandey, S., Xu, R., Wang, W., Chu, X., 2025. OpenFOAMGPT: a RAG-Augmented LLM Agent for OpenFOAM-Based Computational Fluid Dynamics. Physics of Fluids 37, 035120. https://doi.org/10.1063/5.0257555

Unver, H.O., 2013. An ISA-95-based manufacturing intelligence system in support of lean initiatives. Int J Adv Manuf Technol 65, 853–866. https://doi.org/10.1007/s00170-012-4223-z

Vagia, M., Transeth, A.A., Fjerdingen, S.A., 2016. A literature review on the levels of automation during the years. What are the different taxonomies that have been proposed? Applied Ergonomics 53, 190–202. https://doi.org/10.1016/j.apergo.2015.09.013