Journal Pre-proof

turbulence.ai: an end-to-end AI Scientist for fluid mechanics

Jingsen Feng (), Yupeng Qi () , Ran Xu (), Sandeep Pandey, Xu Chu ()

PII: S2095-0349(25)00052-2
DOI: https://doi.org/10.1016/j.taml.2025.100620
Reference: TAML 100620

To appear in: *Theoretical and Applied Mechanics Letters*

Received date: 26 May 2025
Revised date: 2 September 2025
Accepted date: 15 September 2025

Please cite this article as: Jingsen Feng (), Yupeng Qi () , Ran Xu (), Sandeep Pandey, Xu Chu (), turbulence.ai: an end-to-end AI Scientist for fluid mechanics, *Theoretical and Applied Mechanics Letters* (2025), doi: https://doi.org/10.1016/j.taml.2025.100620

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2025 Published by Elsevier Ltd on behalf of The Chinese Society of Theoretical and Applied Mechanics.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

Journal Pre-proof

Highlights

- To our best knowledge, we developed the first AI Scientist for fluid mechanics: turbulence.ai. The AI Scientist is able to generate ideas, evaluate ideas, make simulation plans, perform simulations, analyze data and write&review manuscripts.
- We also present the first pair of full-length, journal-style papers entirely drafted by an autonomous AI scientist (turbulence.ai), covering two-phase flow in porous media and flow-regime transitions in dense micro-fracture networks.
- Both generated manuscripts, specially the second one, can be published on above-averaged journals.

Journal Pre-proof

Journal Pre-proof

turbulence.ai: an end-to-end AI Scientist for fluid mechanics

Jingsen Feng (冯晶森)[a], Yupeng Qi (亓宇鹏)[b], Ran Xu (徐冉)[c], Sandeep Pandey[d], Xu Chu (初旭)[a,e,*]

[a] *Faculty of Environment, Science and Economy, University of Exeter, , Exeter, EX4 4QF, United Kingdom*
[b] *Cluster of Excellence SimTech, University of Stuttgart, , Stuttgart, , Germany*
[c] *Faculty for Aerospace Engineering and Geodesy, University of Stuttgart, , Stuttgart, , Germany*
[d] *Institute of Thermodynamics and Fluid Mechanics, Technische Universität Ilmenau, , Ilmenau, D-98684, Germany*
[e] *University of Stuttgart, , Stuttgart, , Germany*

Journal Pre-proof

Abstract

Fluid mechanics holds an almost infinite range of unsolved questions whose answers could improve energy efficiency, environmental protection and public health. Yet progress is throttled by scarce resources—human expertise, time cost and research funding. We introduce `turbulence.ai`, the first fully autonomous AI scientist for fluid mechanics, designed to lift those constraints. A multi-agent architecture unifies hypothesis generation, CFD execution and draft writing. From a single natural-language query the platform (i) formulates testable ideas, (ii) orchestrates a series of numerical experiments, (iii) interprets results and (iv) produces a draft (attached). This advance inaugurates a new chapter in fluid-mechanics research and could expand human's knowledge base by orders of magnitude.

Keywords: Artificial intelligence, Computational fluid dynamics, Multi-agent systems, Automated research, Scientific discovery

*Corresponding author. Email: x.chu@exeter.ac.uk

Journal Pre-proof

1. Introduction

Large Language Models (LLMs) have achieved remarkable progress in natural language understanding and generative reasoning [1–4]. These developments herald a new paradigm where AI systems contribute directly to scientific innovation, augmenting human expertise with automated insight generation and decision support.

Scientists from other fields have recently pursued the vision of an `AI Scientist` – an autonomous AI-driven system capable of conducting research end-to-end. Notably, Google's `AI Co-Scientist` [5] exemplifies this approach: it is a multi-agent system that autonomously generates and refines scientific hypotheses via a "generate, debate, and evolve" strategy. By orchestrating multiple specialized agents, the system can propose novel research directions and iterate on them in a manner inspired by the scientific method. In a recent demonstration focused on biomedical research, this AI co-scientist was able to identify plausible drug candidates and molecular targets; some of these AI-generated hypotheses were later validated in laboratory experiments, underscoring the system's potential to uncover real-world insights. Sakana AI introduced `The AI Scientist` [6, 7], a platform that automates the entire research pipeline from idea inception to draft writing. Sakana's system can generate original research ideas, write necessary code, execute experiments, analyze results (including creating figures), and compose a full scientific draft describing the findings. Impressively, this fully automated pipeline recently produced what is believed to be the first AI-generated research draft to pass peer review at a machine learning conference workshop. These AI Scientist efforts, while still in early stages, demonstrate the feasibility of LLM-based agents handling complex, open-ended research tasks that traditionally required extensive human intellect and labor.

In the field of fluid mechanics, researchers have begun exploring the integration of LLMs into computational fluid dynamics (CFD) [8–13]. CFD pose a formidable challenge for automation: they involve setting up intricate numerical experiments (defining geometries, boundary conditions, mesh resolutions, solver settings), running computationally intensive solvers, and interpreting high-dimensional output data. Recently, proof-of-concept studies have shown that LLMs augmented with domain knowledge can indeed navigate this complexity.

For example, Chen et al. [14] developed MetaOpenFOAM, an LLM-driven multi-agent framework tailored to automate OpenFOAM-based CFD simu-

Journal Pre-proof

Journal Pre-proof

lations. Dong et al. [8] fine-tuned Qwen2.5-7B-Instruct model on their agent `NL2FOAM`, with a custom dataset of 28716 natural language-to-OpenFOAM configuration pairs with chain-of-thought (CoT) annotation. Yang et al. [12] built a closed-loop dialogue with the LLM that can propose, refine, and physically justify turbulence models that rediscover known strategies and devise superior new ones for adverse-pressure-gradient, rotating, and rough-wall flows.

In prior work, we introduced `OpenFOAMGPT` [15, 16] as a LLM agent for the OpenFOAM CFD platform. With appropriate prompting, the `OpenFOAMGPT` agent demonstrated zero-shot capability in setting up canonical flow problems, adjusting simulation parameters (e.g. boundary conditions, turbulence models) on demand, and even translating cases between different solver frameworks – all through natural language dialogue. Building on this foundation, we developed `OpenFOAMGPT 2.0`, which introduced a more modular, multi-agent architecture to enhance robustness and reliability. In `OpenFOAMGPT 2.0`, multiple specialized agents were employed to handle pre-processing, simulation and post-processing. This design proved highly trustworthy in practice: `OpenFOAMGPT 2.0` achieved 100% reproducibility, including challenging multi-case (up to 100) parametric studies, with each run consistently producing correct and repeatable results. The framework attained a level of trustworthiness appropriate for mission-critical CFD applications. These findings marked an important step toward dependable automation in fluid simulation, aka an `AI Engineer`, demonstrating that LLM-based agents can meet the stringent accuracy and consistency requirements of scientific computing workflows.

`turbulence.ai`, the system presented in this work, aims to push this concept further: to create an AI Scientist for fluid mechanics that can autonomously conduct the entire research procedure, from problem formulation through simulation and analysis. The envisioned system is capable of formulating its own simulation plans based on a high-level research question, retrieving and adapting relevant knowledge to inform the setup, running the simulations while dynamically handling any errors or changes needed, and finally interpreting the results to yield scientific insights. `turbulence.ai` seeks to accelerate fluid mechanics investigations while maintaining the fidelity and trustworthiness that this domain demands. In the following sections, we detail the design of `turbulence.ai`, situate it in the context of the aforementioned developments, and demonstrate its capabilities on representative problems in computational fluid dynamics. Our work represents a step toward empowering researchers with an AI collaborator that can reliably tackle the complexities of fluid mechanics, thereby amplifying human creativity and productivity in fluid mechanics.

2. System architecture of `turbulence.ai`

Figure 1 illustrates the three–stage, multi–agent architecture that equips `turbulence.ai` with end-to-end autonomy: (i) Idea Generation, (ii) Simulation, and (iii) Draft Write-Up. Each stage contains specialised agents that communicate through structured messages and a shared memory, forming a closed control loop with automatic error handling and reflection.

- **Idea Generation.** A plain-language query is first handled by the *Idea-Generator Agent* that drafts a *slate* of fluid mechanics study concepts. This agent employs multiple rounds of chain-of-thought reasoning to systematically explore the problem space, drawing on a comprehensive foundation of fluid dynamics principles, numerical methods, and engineering applications. Each generated concept is refined through iterative self-reflection, where the agent critically evaluates its own proposals before finalizing the slate. Each concept comprises

  – a concise hypothesis framed in fluid mechanics terms;

  – a proposed numerical approach;

  – a tentative parameter sweep (key non-dimensional groups, operating envelopes, grid requirements);

  – an impact statement (energy yield, emission reduction, hazard mitigation, *etc.*).

  A lightweight *Idea-Critique Agent* then performs an automated literature probe (Semantic Scholar API) and scores each concept on novelty, feasibility, societal benefit, and expected computational cost. The agent filters ideas by analyzing similarity to existing publications, discarding concepts that are too closely aligned with prior work while favoring those that build meaningfully upon established knowledge. Using chain-of-thought self-evaluation, it selects the top-ranked idea and expands it into a machine-readable Simulation Plan: solver choice, mesh strategy, boundary conditions, validation metrics, and allowable

Journal Pre-proof

wall-clock budget. The critique agent additionally assigns numerical scores for interestingness, novelty, and feasibility, which are stored in an idea archive to condition future generations toward increasingly innovative directions. This plan—complete with citations and rationale—is handed off to the Simulation stage.

- **Simulation** (`OpenFOAMGPT 2.0`) The machine-readable Simulation Plan triggers a four-agent CFD pipeline that delivers fully vetted results with no manual intervention:
  - *Pre-processing Agent*: parses the plan, detects the required physics, geometry complexity, and study scope, then selects an appropriate mesh strategy—`blockMesh` for simple single-case runs or `snappyHexMesh` (with `STL` ingestion) for complex, multi-case parametric sweeps. All critical parameters (grid resolution, fluid properties, boundary conditions) are formalised and forwarded downstream.
  - *Prompt-Generation Agent*: converts each case definition into a templated natural-language prompt that constrains the solver dictionaries, numerical schemes, and run scripts; the prompts are cached in a *Prompt Pool* to enable orderly, reproducible execution of large design spaces.
  - *Core Simulation Engine*: a tripartite loop comprising **(i) Configuration Generator**, which emits complete OpenFOAM dictionaries under a deterministic, zero-temperature LLM policy enhanced by retrieval-augmented generation (RAG) that accesses validated CFD best practices and methodologies from domain-specific knowledge bases, enabling intelligent parameter selection including experience-dependent turbulence model parameters that traditionally require expert judgment. The synergy between the inherent capabilities of large language models and RAG-enhanced domain knowledge enables precise parameter selection across diverse CFD scenarios; **(ii) Executor**, which launches the run in a containerised v2406 environment and streams logs in real time; and **(iii) Error-Driven Refiner**, which captures any syntax or runtime failure, attaches the log excerpt to the original prompt, and re-invokes the LLM until the case reaches the SUCCESS state.

Journal Pre-proof

This closed loop has been verified to deliver 100 % reproducibility across >450 benchmark simulations ranging from laminar Poiseuille flow to multi-phase porous-media drainage [17].

- *Post-processing Agent*: on completion, automatically harvests fields and probes from the solver's `postProcessing` hierarchy, generates NumPy/Matplotlib scripts or ParaView VTK files as specified in the plan, and returns high-resolution plots plus derived metrics (forces, Nusselt numbers, spectrum analyses, *etc.*) back to the upper-level workflow.

Journal Pre-proof

- **Draft Write-Up.** Once a run (or batch of runs) is flagged Simulation Success, all raw fields, logs, and derived metrics are piped to a Data-Aggregation Node. Run success is determined through automated evaluation metrics including: (i) solver convergence criteria (residual reduction below prescribed thresholds), (ii) mass/energy conservation checks within acceptable tolerances (typically <1%), (iii) mesh-independence verification through grid refinement studies, and (iv) physical plausibility tests (e.g., positive pressures, realistic velocity magnitudes). Failed runs trigger automatic re-execution with adjusted parameters or alternative numerical schemes. This node auto-generates a Python script that consolidates statistics across parameter sweeps and emits publication-quality plots. Each plot, its caption, and the paragraph that references it are then reviewed by a Vision–Language Model (VLM) "AI Reviewer", which flags missing legends, mismatched captions, or duplicated figures and proposes concrete fixes.

  In parallel, a sophisticated Literature Harvester implements a citation collection and verification process. The system employs a two-stage approach for each citation round: first, a domain-specialized LLM agent analyzes the current draft content and research context to identify missing citations across seven key categories: (1) summarizing existing research, (2) supporting specific concepts or data usage, (3) enabling findings comparison, (4) highlighting research gaps, (5) crediting established methodologies, (6) backing arguments, and (7) suggesting future research directions.

  For each identified citation need, the system generates targeted search queries and interfaces directly with the Semantic Scholar API to retrieve relevant publications. The API provides access to over 200 mil-

Journal Pre-proof

lion research drafts with structured metadata including verified DOI, author information, venue details, publication years, and peer-review status. Retrieved drafts are presented to the LLM with complete bibliographic information and abstracts for intelligent selection.

The system implements robust safeguards against AI hallucinations through a strict "external-only" policy: LLMs cannot generate bibliographic information from internal knowledge but must select from API-retrieved results. Each selected draft undergoes automated citation key cleaning through accent removal and special character normalization to ensure LaTeX compatibility. The system maintains a comprehensive audit trail from initial search query through final BibTeX integration, enabling full verification of citation provenance.

Additionally, the framework includes duplicate detection mechanisms that prevent addition of citations with identical titles or content, even when retrieved under different formatting. All citations are automatically formatted as BibTeX entries and integrated into the draft's reference management system with descriptive comments explaining their relevance and intended usage context.

All assets—cleaned figures, contextual citations, and metadata—are injected into a journal-specific Draft Template and rendered in a single pass by the Draft Generator LLM. A reasoning-intensive Reflection Loop, powered by models such as o1, critiques the draft for logical flow, methodological transparency, and figure–text alignment; targeted edits are issued and re-evaluated until preset quality gates are met. However, the current writing system exhibits several notable limitations: (i) tendency toward repetitive phrasing and formulaic expressions that betray LLM authorship, (ii) insufficient critical analysis and physical interpretation of numerical results, (iii) limited ability to synthesize findings into broader theoretical frameworks, (iv) shallow discussion of experimental validation and model limitations, and (v) over-reliance on parameter sweeps without deeper mechanistic insights. These shortcomings reflect the fundamental challenge of translating computational observations into meaningful scientific understanding. This pipeline has already produced workshop-level drafts that survived blind peer review, demonstrating the practical viability of fully autonomous scientific writing.

Journal Pre-proof

Journal Pre-proof

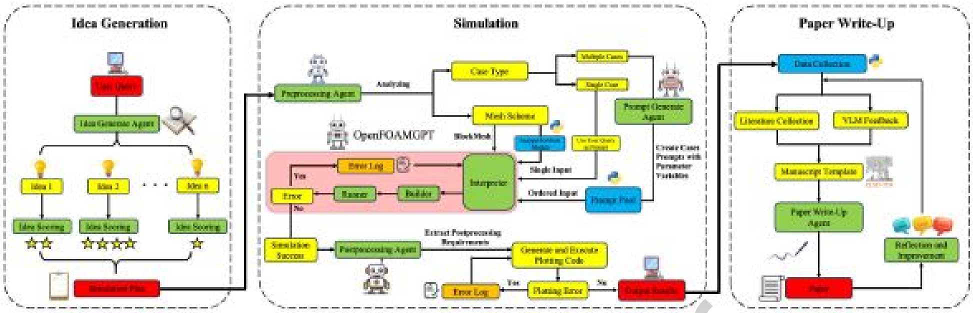

Figure 1: End-to-end architecture of `turbulence.ai`. Green boxes denote autonomous agents powered by LLMs; yellow boxes indicate transient states or artefacts; blue boxes represent reusable prompt or data pools; red boxes mark user-visible inputs/outputs. Solid arrows show the primary control flow; dashed arrows indicate feedback loops for error correction and reflection.

3. The drafts generated by `turbulence.ai`

To illustrate the systematic approach underlying autonomous scientific writing, Algorithm 1 presents the complete workflow that transforms raw experimental data into structured manuscripts. The process unfolds through five distinct phases, each orchestrated by specialized language models and external verification systems. Beginning with the ingestion of research objectives ($\mathcal{I}$) and experimental outputs ($\mathcal{D}$), the pipeline progresses through iterative citation harvesting, where up to $N_c = 30$ rounds of targeted queries against the Semantic Scholar API build a comprehensive bibliographic foundation. A vision-language model subsequently analyzes all generated figures ($\mathcal{F}$), extracting quantitative trends and physical insights that inform the narrative structure. The draft generation phase employs extended thinking mechanisms—allocating up to $\lambda = 30{,}000$ tokens for deep reasoning—to synthesize these disparate elements into coherent LaTeX source. Finally, $N_r = 3$ reflection iterations scrutinize the manuscript for technical accuracy, page-length compliance ($P_{\text{min}} = 25$, $P_{\text{max}} = 35$), and journal-specific requirements, with automated syntax correction and figure placement optimization occurring at each stage. This algorithmic pipeline ensures reproducibility while maintaining the flexibility to adapt to diverse research domains and publication venues.

Journal Pre-proof

Journal Pre-proof

Algorithm 1: Automated Paper Writing Workflow in `turbulence.ai`

Input: $\mathcal{I}$: research idea, $\mathcal{D}$: experimental data, $\mathcal{F}$: figures
Output: $\mathcal{M}$: manuscript PDF
Parameters: N_c: citation rounds, N_r: reflections, $[P_{\min}, P_{\max}]$: pages, τ: min citations, θ: relevance, λ: thinking tokens

`//Phase 1: Data Loading`
$\mathcal{I} \leftarrow$ LoadResearchIdea(); $\mathcal{S} \leftarrow$ LoadExperimentalData(); $\mathcal{F} \leftarrow$ CollectFigures(); $\mathcal{T} \leftarrow$ LoadTemplate()
`//Phase 2: Citation Collection`
$\mathcal{C} \leftarrow \emptyset$
for $i = 1$ ***to*** N_c **do**
  $q \leftarrow$ GenerateQuery($\mathcal{I}$, $\mathcal{C}$); $\mathcal{P} \leftarrow$ SemanticScholarAPI(q)
  $\mathcal{P}_{sel} \leftarrow$ SelectRelevantPapers($\mathcal{P}$, θ); $\mathcal{C} \leftarrow \mathcal{C} \cup$ CleanBibtex($\mathcal{P}_{sel}$)
  if $|\mathcal{C}| \geq \tau$ **then break**
end
`//Phase 3: Figure Analysis`
$\mathcal{V} \leftarrow \emptyset$
for ***each*** $f \in \mathcal{F}$ **do**
  $\mathcal{V}[f] \leftarrow$ VLM.AnalyzeImage(f)
end
`//Phase 4: Draft Generation`
$\rho \leftarrow$ AssemblePrompt($\mathcal{I}$, $\mathcal{S}$, $\mathcal{F}$, $\mathcal{V}$)
$\mathcal{L}_0 \leftarrow$ LLM.GenerateWithExtendedThinking(ρ, λ)
$\mathcal{L}_0 \leftarrow$ PreCheckSyntax($\mathcal{L}_0$); $\mathcal{L}_0 \leftarrow$ FixFigurePlacement($\mathcal{L}_0$)
`//Phase 5: Iterative Refinement`
for $j = 1$ ***to*** N_r **do**
  $\pi_j \leftarrow$ CompileLaTeX($\mathcal{L}_{j-1}$); $p \leftarrow$ DetectPageLength(π_j); $\mathcal{E} \leftarrow$ ValidateErrors($\mathcal{L}_{j-1}$)
  if $p \in [P_{\min}, P_{\max}]$ ***and*** $\mathcal{E} = \emptyset$ **then return** π_j
  $\phi \leftarrow$ GenerateFeedback(p, $\mathcal{E}$); $\mathcal{L}_j \leftarrow$ LLM.ReflectAndImprove($\mathcal{L}_{j-1}$, ϕ)
end
return CompileLaTeX($\mathcal{L}_{N_r}$)

Implementation details. The algorithm operates with default parameters calibrated through extensive empirical testing: $N_c = 30$ citation rounds ensure comprehensive literature coverage while avoiding redundancy, $N_r = 3$ reflection iterations balance quality improvements against computational cost, page limits $[P_{\min}, P_{\max}] = [25, 35]$ align with typical journal requirements, the minimum citation threshold $\tau = 30$ satisfies academic standards for thorough referencing, relevance threshold $\theta = 0.7$ filters spurious matches while retaining pertinent works, and the extended thinking budget $\lambda = 30{,}000$ tokens enables sophisticated reasoning about complex physical phenomena. Each phase incorporates fail-safe mechanisms: citation collection validates against the Semantic Scholar database to prevent hallucination, figure analysis employs multi-modal verification to ensure accurate interpretation, and the compilation step detects LaTeX errors through pattern matching before invoking the compiler. The entire pipeline executes deterministically given identical inputs, facilitating reproducibility and debugging.

Two full-length drafts produced without human intervention using this algorithmic framework accompany this study. The first, entitled *"Parametric Investigation of Two-Phase Displacement Efficiency in Porous Media"*, explores how wettability, inlet velocity and viscosity ratio shape immiscible displacement in a synthetic sandstone analogue. The second, *"Flow Regime Transitions in Dense Parallel Micro-fracture Systems"*, extends to a lattice of hundred-plus micro-fractures and adds an energy-budget perspective. Both drafts look and feel like conventional journal submissions, complete with abstract, theory, methods, results, discussion, conclusions and reference list. That alone is a noteworthy achievement for an AI Scientist. Yet, when read with a reviewer's eye, the work reveals a profile of strengths and shortcomings.

Overall quality. The drafts are internally coherent, adopt appropriate templates and include essentials such as grid-independence tests, solver validation references and high-resolution figures. Citations are mostly real and—spot checks suggest—correctly matched to their subjects. The prose is serviceable but repetitive; stock phrases like "systematic investigation" and "fundamental insight" recur often enough to betray an LLM origin. In short, the drafts clear the bar for technical completeness but not yet for the critical depth expected in a top-tier venue.

Porous-media study (Attachment 1). The porous-media draft impresses through its breadth: direct numerical simulations sweep contact angle from $60°$ to $150°$, velocity from 0.001–0.008 m s^{-1} and viscosity ratio from 0.1–10, while keeping all other properties fixed. The domain is a 4×2 mm^2 grain pack resolved with half a million cells, and results are clearly distilled into a regime map and an efficiency curve that peaks at neutral wettability and capillary numbers around 6×10^{-5}. The methodological section is carefully documented; mesh convergence and solver details follow accepted practice.

Journal Pre-proof

Micro-fracture study (Attachment 2). The second draft aims higher conceptually by coupling a 15-case parametric sweep to a full energy-conversion audit. A network of 110 nearly parallel fractures allows the study to identify two capillary thresholds that separate capillary-dominated, transitional and viscous regimes. Tracking kinetic, surface and dissipated energy along each trajectory is a sophisticated touch, and the numerical energy balance is reportedly conserved within two percent. Yet deficiencies similar to the first draft resurface. All simulations are still two-dimensional; the inlet velocities top out at 0.0048 m s^{-1}, which is modest for fracture flow; and key findings—such as the shift of critical capillary numbers relative to porous media—are explained only qualitatively. Several plots omit data at late times, and the discussion section offers little microscopic interpretation of the displacement morphologies.

Cross-cutting assessment of the AI scientist. Taken together, the drafts demonstrate that `turbulence.ai` can already reproduce much of the workflow an above-average researcher would follow: selecting a novel-enough topic, setting up a well-posed numerical experiment, running a credible solver, organising the outputs and writing a complete draft. Where the system is still recognisably non-human is in its critical sensibility. It excels at breadth—parameter sweeps, literature harvesting, figure generation—but struggles with depth: framing sharper hypotheses, quantifying uncertainty, flagging when a result is too context-specific, or admitting when the numerical evidence is insufficient. Those weaknesses echo the limits of present-day LLMs in reasoning about physical implication versus computational observation.

Verdict. Either draft, with targeted revision could be publishable in a fluid-mechanics or porous-media journal. From the vantage point of research quality, `turbulence.ai` has reached a high level of technical literacy but not yet the mature judgement that distinguishes robust science from plausible simulation.

4. Conclusion

Fluid mechanics holds a vast set of unresolved problems whose solutions promise major gains in energy efficiency, environmental sustainability, and public health; progress, however, is curtailed by limitations in specialised expertise, time, and research funding.

Journal Pre-proof

Journal Pre-proof

We introduced `turbulence.ai`, the first fully autonomous AI scientist for fluid mechanics, and showcased its inaugural draft. By fusing idea generation, numerical experiments and draft writing into a multi-agent system, it converts a single research question into publishable insight with no human interventions. The generated research will enrich the human's knowledge base by order of magnitudes and serve for future AI training, lowering the cost of the next and creating a compounding engine for discovery. In short, `turbulence.ai` removes the traditional bottlenecks of time, expertise and budget, opening virtually boundless research terrain; This marks a new chapter in fluid-mechanics research.

Despite these promising capabilities, `turbulence.ai` exhibits several critical limitations that must be acknowledged. Most importantly, this work and the accompanying drafts are not intended for publication but rather serve as a demonstration of AI-assisted research capabilities to support and accelerate human scientific endeavors. The system's primary value lies in its ability to assist researchers by automating routine computational tasks, generating preliminary insights, and providing starting points for deeper investigation. However, accurate assessment and interpretation of the system's outputs still require highly qualified researchers with deep domain expertise in fluid mechanics, computational methods, and physical interpretation—posing significant challenges for beginners, students, or non-specialists who may lack the necessary background to critically evaluate the generated results. While the system can produce technically competent drafts, these outputs should be regarded strictly as reference material and preliminary findings that require rigorous human review, verification through independent analysis, and substantial refinement before any research consideration. Currently, the system's scope is limited to well-established numerical methods and canonical flow problems, with limited capability for theoretical discovery and mechanistic insights. Future developments will focus on enhancing the system's ability to identify underlying physical principles, derive theoretical frameworks, and contribute to fundamental understanding rather than merely performing computational parameter studies. The goal is to evolve `turbulence.ai` from a computational assistant into a genuine partner in scientific discovery.

Declaration of Competing Interests

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

Journal Pre-proof

Journal Pre-proof

reported in this paper

Acknowledgments

XC appreciates the funding support from Royal Society (RG\R1\251236).

References

[1] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al., Gpt-4 technical report, arXiv preprint arXiv:2303.08774 (2023).

[2] A. Liu, B. Feng, B. Xue, B. Wang, B. Wu, C. Lu, C. Zhao, C. Deng, C. Zhang, C. Ruan, et al., Deepseek-v3 technical report, arXiv preprint arXiv:2412.19437 (2024).

[3] A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei, et al., Qwen2. 5 technical report, arXiv preprint arXiv:2412.15115 (2024).

[4] M. J. Buehler, Cephalo: Multi-modal vision-language models for bio-inspired materials analysis and design, Advanced Functional Materials (2024) 2409531.

[5] J. Gottweis, W.-H. Weng, A. Daryin, T. Tu, A. Palepu, P. Sirkovic, A. Myaskovsky, F. Weissenberger, K. Rong, R. Tanno, et al., Towards an ai co-scientist, arXiv preprint arXiv:2502.18864 (2025).

[6] C. Lu, C. Lu, R. T. Lange, J. Foerster, J. Clune, D. Ha, The ai scientist: Towards fully automated open-ended scientific discovery, arXiv preprint arXiv:2408.06292 (2024).

[7] Y. Yamada, R. T. Lange, C. Lu, S. Hu, C. Lu, J. Foerster, J. Clune, D. Ha, The ai scientist-v2: Workshop-level automated scientific discovery via agentic tree search, arXiv preprint arXiv:2504.08066 (2025).

[8] Z. Dong, et al., Fine-tuning a large language model for automating computational fluid dynamics simulations, arXiv preprint arXiv:2504.09602 (2025).

Journal Pre-proof

Journal Pre-proof

[9] M. Du, Y. Chen, Z. Wang, L. Nie, D. Zhang, Large language models for automatic equation discovery of nonlinear dynamics, Physics of Fluids 36 (2024).

[10] Z. Xu, L. Zhu, Training microrobots to swim by a large language model, 2024. URL: https://arxiv.org/abs/2402.00044. arXiv:2402.00044.

[11] M. Elrefaie, J. Qian, R. Wu, Q. Chen, A. Dai, F. Ahmed, Ai agents in engineering design: A multi-agent framework for aesthetic and aerodynamic car design, arXiv preprint arXiv:2503.23315 (2025).

[12] Z. Yang, et al., Large language model driven development of turbulence models, arXiv preprint arXiv:2505.01681 (2025).

[13] 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, 2024. URL: https://arxiv.org/abs/2412.08072. arXiv:2412.08072.

[14] Y. Chen, et al., Metaopenfoam: an llm-based multi-agent framework for cfd, arXiv preprint arXiv:2407.21320 (2024).

[15] S. Pandey, R. Xu, W. Wang, X. Chu, Openfoamgpt: a rag-augmented llm agent for openfoam-based computational fluid dynamics, arXiv preprint arXiv:2501.06327 (2025).

[16] W. Wang, R. Xu, J. Feng, Q. Zhang, X. Chu, A status quo investigation of large language models towards cost-effective cfd automation with openfoamgpt: Chatgpt vs. qwen vs. deepseek, arXiv preprint arXiv:2504.02888 (2025).

[17] J. Feng, R. Xu, X. Chu, Openfoamgpt 2.0: end-to-end, trustworthy automation for computational fluid dynamics, arXiv preprint arXiv:2504.19338 (2025).

Journal Pre-proof