GenAI Usage Documentation Group 12

AI Usage Strategy

In line with the course guidelines, our primary goal is to use GenAI as an enhancement to our own reasoning and coding abilities rather than a substitute so that we can achieve results that previously may not have been possible while increasing our knowledge and abilities.

- Primary Tools: Use GitHub Copilot (via UvA access) for real-time coding and Gemini for conceptual clarification and high-level project brainstorming.
    - Other tools such as Claude code, ChatGPT etc. can also be used for similar purposes but care must be taken to use tools for problems they are optimised for e.g. Claude for coding but perhaps not higher level brain-storming
- Targeted Activities: Focus usage on debugging complex Lattice Boltzmann Method (LBM) code, refining F1 turbulence model design, and clarifying theoretical concepts from lectures.
- Validation Protocol: Every AI-generated code snippet or derivation must be manually validated. AI is notoriously unreliable and prone to hallucinations and other issues, failing to validate AI generated material is a critical error.
    - Utilise multiple AI tools in an adversarial manner, comparing results generated by each and using them to critique each other's work.

Prompting Strategy

1. Define the persona and context
   Clearly specify the AI tools role to improve the relevance of the output.
   Example: “Act as a Computational Science Professor specializing in Fluid Dynamics. I am working on an LBM simulation for F1 ground effect turbulence…”
   Note that this works better with some Generative AI tools than other such as Gemini or ChatGPT
2. Specify Goals and Constrains
   Provide clear objectives and technical constraints to minimise hallucinations.
   Example: “...Explain the stability criteria for the D2Q9 lattice model. Provide the explanation in bullet points and highlight the most important variables.”
3. Iterative Refinement and Reasoning
   Ask the tool to explain its logic to ensure you understand the underlying principles.
   Example: “Explain the reasoning behind this specific implementation of the bounce-back boundary condition. Are there alternative approaches for curved surfaces?”
4. Provide context when refining.
   Do not simply state that there are errors, rather explain what the error is, where it is and what you think may be causing it to allow the tool to faster and more efficiently solve the problem.

Review of using AI:

The team most certainly greatly benefitted from the use of AI in two major ways. The first was its use to break down theoretical concepts such as turbulence, what a Reynold’s number actually

means etc. This was very useful as not all members had experience with these concepts before and the area in which this project was situated is relatively niche. Additionally, AI greatly expedited the speed of code generation but not necessarily the quality. The use of AI allowed the team to very quickly develop working prototypes which had advantages and disadvantages. On one hand, very quickly getting something working allowed for refinement from early on but on the other hand, not actually developing those prototypes ourselves meant that solving issues and building upon what was previously built was very difficult and would occasionally require beginning from scratch to ensure understanding.

The team noticed that the following were essential when utilising AI:

1. Always be very specific with what is required: AI has the tendency to go completely overboard which can be confusing and very unproductive.
2. Critically evaluate everything: AI may tell you that everything is working fine but that is not usually the case.
3. Investigate all results: Ensure that results generated from programs developed with AI assistance align with theoretical results, ideally under different parameter settings.
4. Keep your goals small with each prompt: Do not ask an AI tool to do everything all at once as this often seriously degrades results.

Documenting Requirements:

- Tool used e.g. Copilot, Gemini
- Prompting strategy followed (see above)
- Specific Prompt
- Output generated (size and type)
- What was used, adopted or discarded and reasons.
- Observations about tools usefulness and effectiveness.

Example prompts:

Ride height sweep simulation error with NaN values (Claude)

Prompt - When running this code, specifically the ride height parameter sweep, I encountered an error on the 14th run (see error message below), can you analyze my code and see what may be causing it. I think there may be a couple of errors:

1. Nx and Ny are too small (I hope this is not the case as increasing them further drastically increases run time)
2. The mask geometry is not evenly moved up by an increase in the ride height i.e. certain parts of the car move upwards but not all of them meaning peices overlap and create weird geometries.
3. Bounding box/area of analysis does not change with the increase in ride height meaning an error occurs because the simulation is analysis "nothing" or Nx and Ny are too small for the size of the geometry.

Output - Text

Notes - Really great analysis of what is happening, didn't actually manage to find the issue but went through all of the possible issues I provided and would explain to me why none of them would be the error.

Development of Project Workflow (Gemini)
Prompt -You are a complex systems lecturer who is assisting students with a project in which they are required to model/simulate some type of complex phenomena within any application domain. The students have decided to investigate turbulent flow in ground effect for F1 cars. Based on the uploaded project plan, please suggest a workflow that can be followed by the group in terms of code development.
Output generated = Text
Notes = Given to Claude to receive feedback but did not seem particularly technical.

Evaluation of Project Workflow (Claude)
Prompt - We have generated a potential workflow based on a project planning document. Please suggest improvements or important changes to the proposed workflow based on this document.
Output = Text
Notes = Given to Gemini to receive feedback. More technical and gave us a clearer idea of what we needed to do.

Code Steps (Claude)
Prompt - Based on the modified workflow, please identify major coding steps that need to be completed.
Output - Text
Notes - Generated function headings which were very useful but results seemed to contain a lot of bloat/features that were never previously mentioned while skipping over some basic steps that we all knew would be necessary.

Phase 1 Implementation (Claude)
Prompt - Based on the provided workflow, please assist with phase 1. Specifically, please help generate the MVP.
Output - Multiple files (.py, .MD)
Notes - Generated the whole project even when explicitly asked to generate only the MVP - Very unwieldy.

Photo Generation for Presentation (Claude)
Prompt - Please help edit this image for inclusion in a PowerPoint presentation (theme of black and white). Please remove the actual simulation data and keep only the reference lines and convert it to black and white
Output - PNG

Notes - Required further refinement to increase text size, line width etc. but not a bad alternative to Photoshop or a similar tool when the user lacks experience.