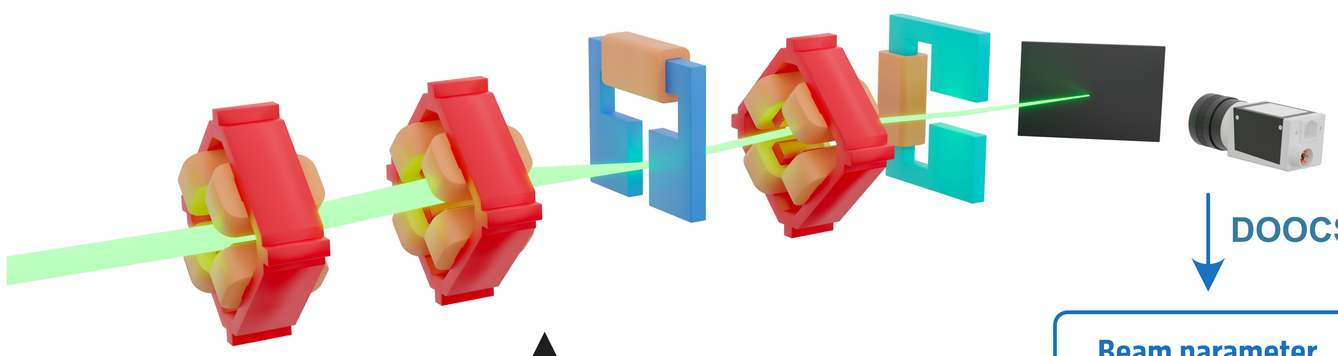
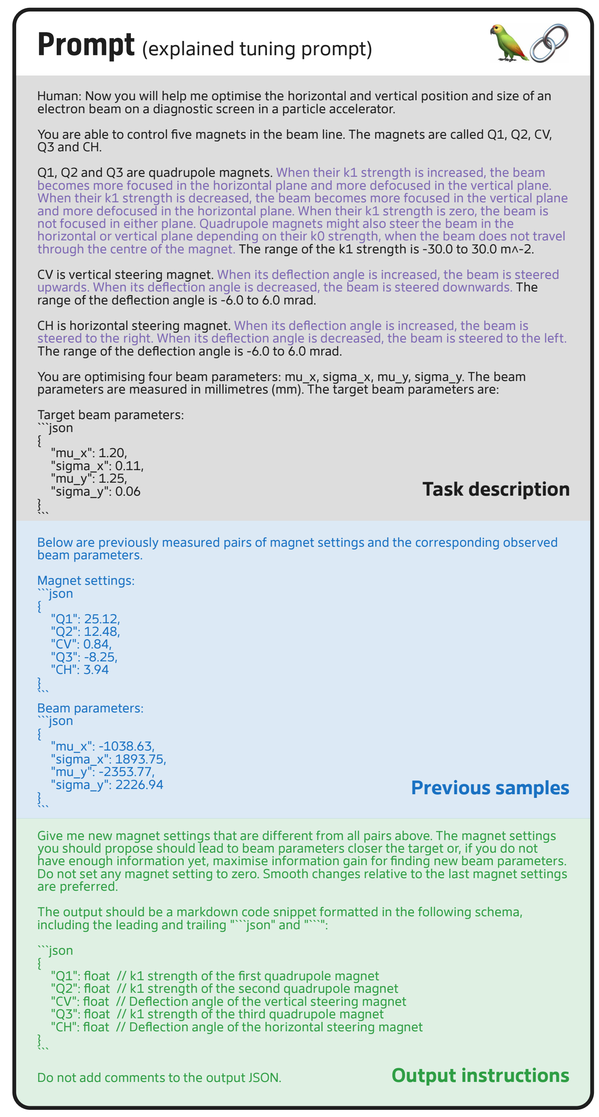
Large language models (LLMs) like GPT-4, which are really **only trained to do text completion**, can be (ab)used as optimisers to directly and **autonomously tune a particle accelerator**.





Towards Natural Language-driven Autonomous Particle Accelerator Tuning.

Jan Kaiser*, Annika Eichler, Anne Lauscher

Motivation

- Large language models (LLMs) have most recently demonstrated incredible capabilities beyond simple text completion, such as programming and problem solving, with **emergent-behaviours** appearing as models become larger and better trained.
- This raises the question: **Can LLMs tune a particle accelerator?**
- Ultimately, this could be a step towards accelerator operators defining goals for accelerator operation through natural language, with LLMs figuring out how to get there.

Implementation

- We engineer a prompt made up of three main components:
  - In the **task description**, we explain the task and define a goal.
  - We then list a history of **previously collected samples** of inputs and resulting outputs.
  - At last, we provide **instructions for the output**, asking the model to output the next magnet settings and providing format instructions for later parsing.
- We test four different **prompting strategies** following the above scheme:
  - **Tuning prompt**: Frames the task in terms of accelerator tuning, where the model is tasked to find magnet settings that result in specific target beam parameters. We assume the model knows how magnets in an accelerator work.
  - **Explained tuning prompt**: We explain in simple terms how quadrupole magnets etc. work and how they influence the beam parameters. Otherwise like the tuning prompt.
  - **Chain-of-though prompt**: Ask the model to explain its reasoning before providing the next magnet settings. Otherwise like explained tuning prompt.
  - **Optimisation prompt**: Frames the task as a simple optimisation, where the output is an objective value. The model is unaware it is tuning an accelerator.
- Implementation of prompt creation with *LangChain*. LLMs are run through the *OpenAI* API or with *Ollama* on A100 GPUs on the DESY Maxwell cluster.

Results

- Success depends on model and prompting scheme:
  - **Often successful with *GPT-4*** / *GPT-4-Turbo* and explained tuning prompt.
  - Few successes with capable open source models like *Mixtral 7x8B* and even small ones like *Mistral 7B* and *Starling LM*.
  - Some models, like *Llama 2 70B* and *Orca 2 13B*, fail because they explain a strategy instead of providing the next magnet settings. Others, like *GPT-3.5-Turbo*, simply fail to show reasonable behaviour.

Outlook

- More extensive evaluation of **large commercial models**, including *Gemini* and *Claude*, as well as inevitably **better future models** (*Gemma*? *GPT-5*?).
- Test potentially better suited prompting schemes like ***ReAct***.

***Corresponding author:** jan.kaiser@desy.de

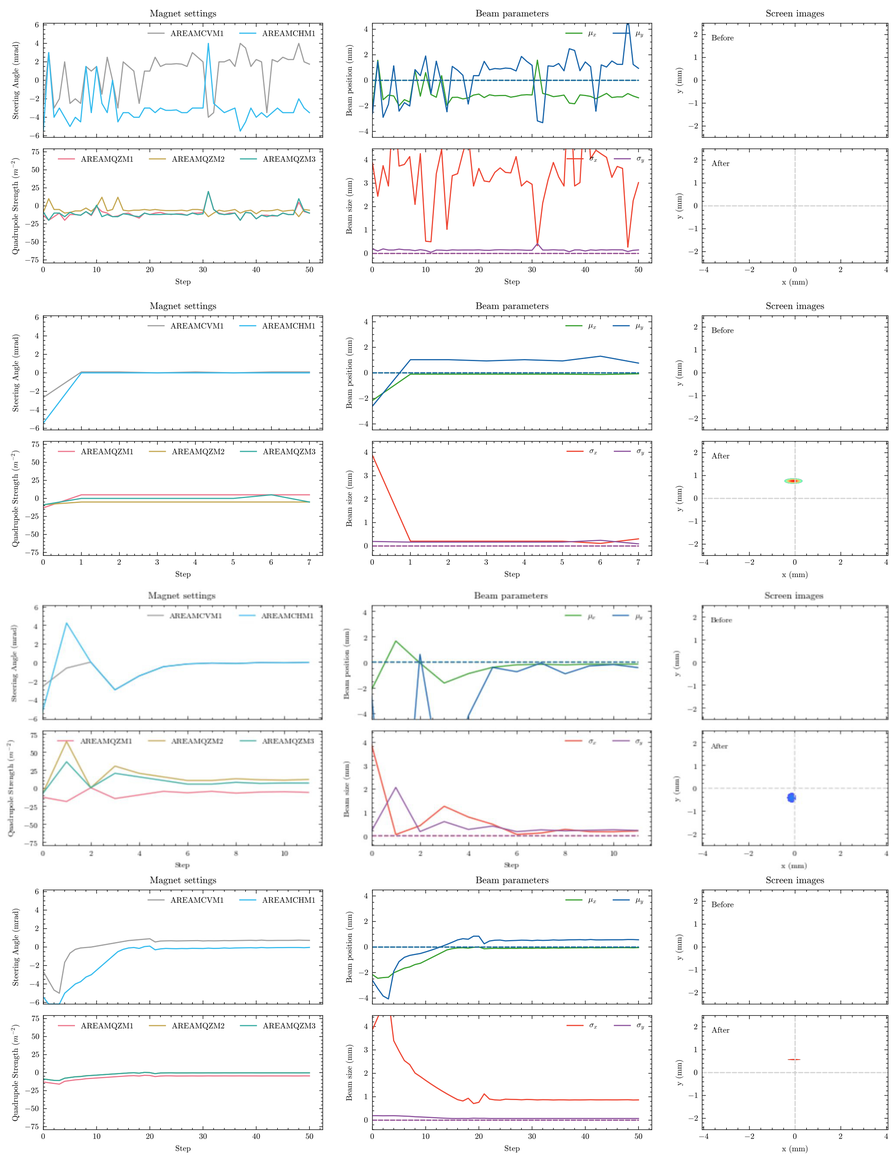
Example tuning runs by LLMs

Cannot be helped …
GPT 3.5 Turbo (chain-of-thought prompt)

Better!
Mixtral 7x8B (explained prompt)

Good, but failed at formatting once tuned.
GPT 4 (explained prompt)

Very good!
GPT 4 (optimisation prompt)



Scan me to download the poster!

Deutsches Elektronen-Synchrotron DESY
A Research Centre of the Helmholtz Association

SINBAD HIR³X Universität Hamburg DER FORSCHUNG | DER LEHRE | DER BILDUNG DESY.