

Guidelines for the Numerical Optimization Project

*commitments to be sent by email to `leo.simpson@imtek.uni-freiburg.de`
before **June the 23rd, 2026, at 23:59***

Prof. Dr. Moritz Diehl, Léo Simpson

As explained at the beginning of the course, you have the possibility of doing a project to get an additional 3 ECTS. The project consists in the formulation and implementation of a self-chosen problem or algorithm of Numerical Optimization, resulting in a documented computer code, a project report, and a public presentation. In the following section, you will receive some information regarding the scope of the project and our expectations.

If you are interested in doing a project, please **let us know via email before June the 23rd, 2026, at 23:59**. This is a **necessary condition** for being allowed to do the project. The email should contain the title of your project, the corresponding authors, and a short description of your project (the latter would be, for example, if you choose to work on an application-oriented project, a description (in words) of your objective, constraints, and decision variables). Note that you will not be required to stick to your description, but you should have spent some thought on it already.

You will be able to ask project-related questions during the Q&A sessions on June the 30th, and July the 14th. There, we will help you shape your ideas into feasible projects.

1. Make sure that you are **properly registered** for the project as required by your examination office (registration for the Studien- and/or Prüfungsleistung (SL/PL)).
2. The project can be done in **groups of up to three** students. If you would like to form a group but do not have a group mate yet, you can use the forum on Ilias to get in contact with each other.
3. The project topic can be one of the three following kinds:
 - **Algorithm-oriented project:** You choose a scheme for the numerical solution of optimization problems. The focus is on the implementation of the scheme and an investigation of its performance using several test problems/benchmarks. They should illustrate the properties of the algorithm, but they do not necessarily have a real-world interpretation.
 - **Application-oriented project:** You formulate and solve a self-chosen optimization problem. The focus should be on the mathematical description of your problem (the modeling, formulations), its numerical solution (including the choice of the optimization solver) and the interpretation of the results.
 - **Class of problem-oriented:** You choose a specific class of optimization problems that you want to analyze. This is at the border between the two previous types of projects. There, you should discuss the two aspects of the problem: introduce (at least) one method to solve numerically solve these problems and introduce (at least) one possible application of this class of problems.
4. On **July the 21th, 2026**, a **short presentation** (probably 5 min)¹ will be given by the author(s) to the teacher and the class. The slides can be based on material taken from the report and may contain additional content, e.g., videos, if required. Even though there is some time left to finish the report, you should already present the final results.
5. After the presentation, you have some time to update your results based on the feedback. The **deadline** for the submission of the written report and the code is:

¹The exact time will be announced once the number of projects is known.

6. The **project grade** is based on the form and content of the report, the originality and quality of the results, and the clarity of the code.
7. Your main output will be the **written report**.
 - The report should be submitted as a PDF file, written in English, and approximately 5 pages long.
 - You can consider using \LaTeX^2 . You can consider using the official IEEE template for conference papers.
 - The report must be a new and self-written document. The report must be solely written by the author(s). You can use LLM tools (e.g. chatGPT) to help you with the \LaTeX syntax or to check grammar and spelling, but we *strongly* discourage you from using them to generate some parts of the report. It should be based on your creativity and your own understanding of the project.
 - The report must include a short, interesting title, the name(s) of the author(s), and an abstract. The content should be clearly structured in sections. It should start with an introduction and conclude with a short summary and critical discussion of the results.
 - The report should contain at least one (self-made) sketch of the modeled system or implemented algorithm.
 - Figures and tables should have a short caption and be referenced in the text properly, e.g. “the results are shown in Fig. 1”. Use the Latex commands `\caption`, `\label` and `\ref`.
 - Plots must contain physical units and axis descriptions.
 - The report must cite all external sources as references at the end and other people’s contributions must be acknowledged. Using other people’s ideas and help is allowed, even encouraged. However, not citing or acknowledging them properly is fraud.
 - If you use academic open source software to obtain your results, it is also usually a request by the developers that you acknowledge their work by citing the corresponding papers, e.g. “The problem was formulated via the Matlab interface of the symbolic framework CasADi [1] and solved using the interior point method based solver IPOPT [2].”
 - Mathematical/physical variables shall consist of one letter only and be printed in italics. This is automatic in Latex, e.g., a_i as `$\$a_i\$$` . Physical units and sub- or superscripts that refer to words are in normal roman letters (use `\mathrm` when in Latex `mathmode`, e.g. x_{initial} as `$\$x_{\mathrm{initial}}\$$` or $\frac{\text{kg}}{\text{m}^3}$ as `$\$\frac{\mathrm{kg}}{\mathrm{m}^3}\$$`). For example, write $m = 5 \text{ kg}$ and not $m = 5kg$ or $m = 5\text{kg}$.

References

- [1] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl. CasADi – A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 11(1):1–36, 2019.
- [2] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.

²If you have not learned \LaTeX yet, see this report as an opportunity. It will certainly pay off for your master’s thesis.