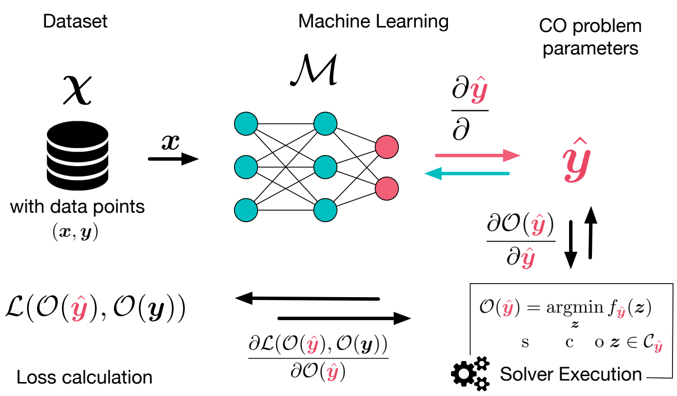
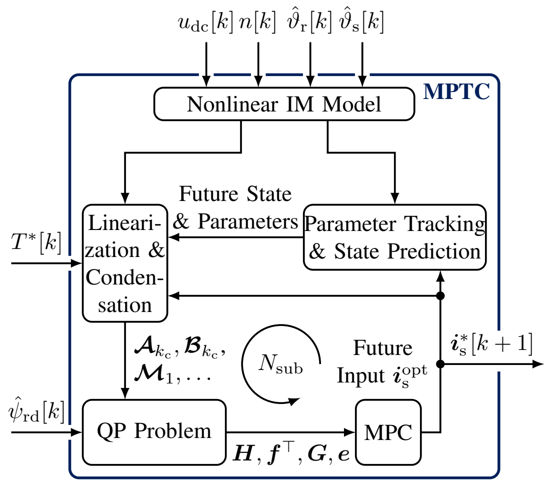
Master thesis
Learning to optimize electric drive control during runtime

Controlling electric drives often comes with various real-time optimization needs, such as choosing (steady-state) operation points (trying to minimize drive losses) [1] or model predictive control (MPC) [2] for trajectory tracking. The underlying optimization problems are often formulated as constrained quadratic programs and sometimes even as nonlinear problems. Both classes can be solved with numerical optimization methods such as interior-point methods or sequential quadratic programming. However, these methods might computationally expensive and require tuning to achieve good performance, in particular in the context of electric drives where control decisions must be made in the sub-millisecond range. Hence, this thesis should investigate the potential of machine learning algorithms which learn the optimization procedure, i.e., replace the numerical optimization solver [3]. While solving the specific task offline and just deploying the result (e.g., via a look-up table) is a typical standard approach nowadays, issues arise if the drive shows time-varying behavior (e.g., due to temperature or aging effects) motivating an online optimization need. The intended goal is to speed up the optimization process to enable more challenging control tasks in real-time.



(a) Learning to optimize (L2O) toolchain [3].



(b) MPC example for electric drives [2].

Key research questions:

- Is learning the optimizer feasible for electric drive control (calculation speed)?
- Can, or must we learn specialized optimizers for specific control tasks?
- What are the differences in terms of performance and computational complexity compared to standard offline /online approaches with established solvers?

Necessary requirements:

- Finished course work on machine learning or numerical optimization
- Ideally: finished course work on electric drives
- Solid skills in scientific programming languages (e.g., Julia, JAX, PyTorch)

WP 1: Literature research **[3 weeks]**
Scanning the scientific literature for relevant publications and patents related learning to optimize (L2O) is the first step. Also, getting familiar with typical constrained optimization problems in the drive control domain is part of this WP. Moreover, relevant (open-source) software work in the field (e.g., NeuroMANCER) should be considered. This also includes the identification of relevant keywords as part of the search strategy. Relevant work will be stored in a literature review software (e.g., JabRef) and summarized in the thesis.

WP 2: Operating point selection **[5 weeks]**
A traditional optimization problem in electric drive control is the selection of the optimal operating point. This can be formulated as a constrained optimization problem: minimize the drive losses while fulfilling the torque reference requirement as well as current and voltage limitations. This problem can be typically well-solved for a given machine and operation conditions (e.g., fixed temperature) such that standard numerical optimization solvers can be used to deliver benchmark results. Nevertheless, the goal of this WP is to model the underlying optimization problem within an end-to-end differentiable framework (e.g., PyTorch, JAX, Julia) and train a neural network to learn the optimizer. The training data should be generated by solving the optimization problem for a set of random operating points and different drive parameters. Finally, the trained optimizer should be compared against the standard numerical optimization solver in terms of performance and computational complexity.

WP 3: Optimal transient control **[5 weeks]**
Another typical optimization problem in electric drive control is the trajectory tracking problem. This can be formulated as a model predictive control (MPC) problem: minimize the torque tracking error while fulfilling the current and voltage limitations. Compared to the previous WP's static problem, this transient control problem considers the dynamical behavior of the drive plant. Hence, building an end-to-end differentiable model for this problem is more challenging as an additional ODE solver layer is required. The training data should be generated by solving the optimization problem for a set of random trajectories and different drive parameters. Finally, the trained optimizer should be compared against the standard numerical optimization solver in terms of performance and computational complexity.

WP 4: Hyperparameters **[4 weeks]**
The learned optimizer and its training toolchain have several hyperparameters that need to be tuned. The hyperparameter optimization (HPO) can be automated using available open-source toolboxes and should also focus on the complexity vs. accuracy trade-off by identifying the Pareto front between these two objectives. Depending on the computational load of this WP, the HPO can be parallelized on a high-performance computing cluster. Furthermore, the question should be addressed if the learned optimized structure should be adapted to specific control tasks or if a general optimizer can be learned for all tasks.

WP 5: Embedded deployment **[4 weeks]**
Since a special focus of this thesis is on online optimization, the trained optimizers should be deployed on an embedded system (e.g, a dSPACE rapid control prototyping platform). This requires the conversion of the previously learned optimizers and underlying models to a format that can be executed on the embedded system. The performance of the embedded optimizer should be evaluated in terms of calculation speed and accuracy based on a software-in-the-loop scenario.

WP 6: Documentation **[3 weeks]**
All work packages should be reported in a structured way within the thesis. A LaTeX template should be used for this purpose: `https://github.com/IAS-Uni-Siegen/thesis_latex_template`. Writing instructions can be found within the provided template source files. Based on the previous empirical findings, conclusions should be drawn, and future research directions should be outlined.

Gantt chart

The planned timetable is shown in the Gantt diagram below.

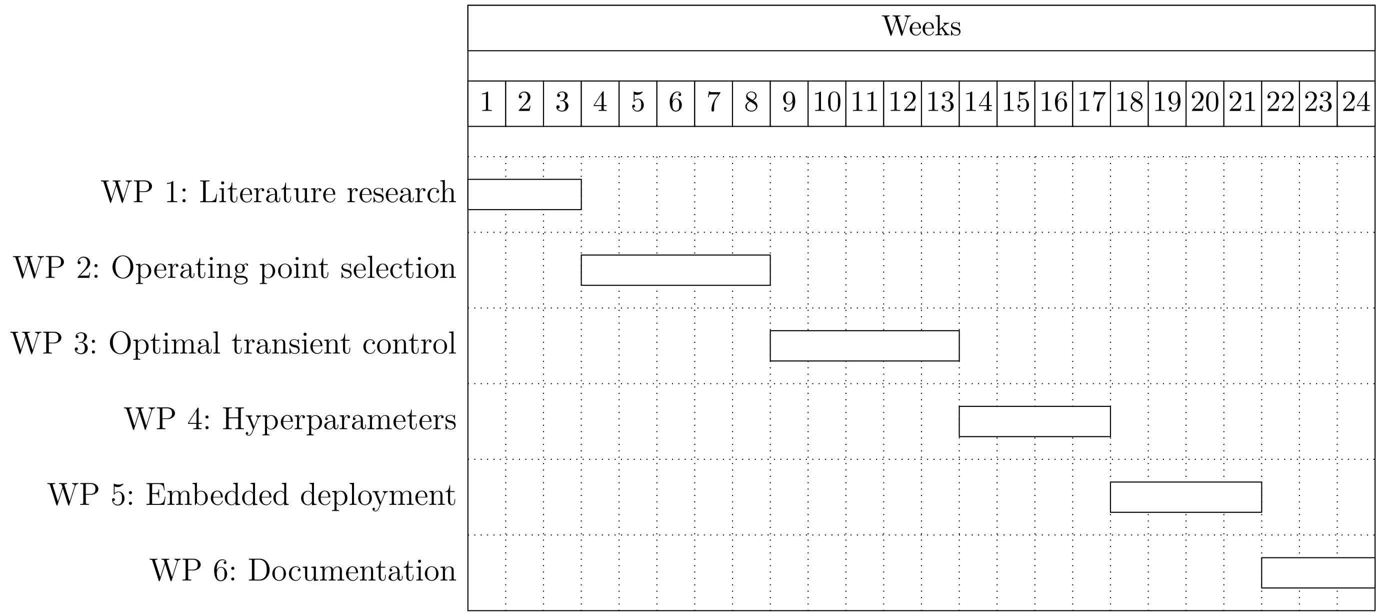


Figure 2: Gantt chart for the thesis.

References

[1] H. Eldeeb, C. M. Hackl, L. Horlbeck, and J. Kullick, “A unified theory for optimal feedforward torque control of anisotropic synchronous machines,” *International Journal of Control*, vol. 91, no. 10, pp. 2273–2302, 2018.

[2] M. Becker, M. Stender, and O. Wallscheid, “Nonlinear efficiency-optimal model predictive torque control of induction machines,” *Authorea Preprints*, 2023.

[3] J. Kotary, F. Fioretto, P. van Hentenryck, and B. Wilder, “End-to-end constrained optimization learning: A survey,” in *International Joint Conference on Artificial Intelligence (IJCAI)*, 2021, pp. 4475–4482.