1. Allgemeines

In this dissertation, I'm aiming to make first-order theorem proving more applicable to automated program reasoning and verification. Specifically, we want to improve the *automation of inductive reasoning over loops and recursive functions with first-order theorem provers*. The integral part of this work is automating inductive reasoning which is famously hard as verifying the correctness of any formula containing un- or semi-decidable theories such as integer arithmetic is an semi-decidable problem in mathematics. Thus, mathematically speaking verification heavily depends upon finding either decidable fragments of logic, or, as we do in this work, find ways to make our approaches applicable to a wide set of problems even in the sight of undecidability of the underlying logical framework.
We achieve this goal through multiple works expanding on inductive reasoning for trace logic, invariant generation as well as investigating functional programming paradigms and provide a proof of concept of proving functional sorting algorithms correct.

2. Ergebnisse

In the course of the netidee stipend, three scientific publications were created of which one is still under peer review. All of the works below further the use of first-order theorem provers to showcase its efficiency and possibilities of usage when it comes to automated software verification. They are listed below:

(1) Bhayat, A., Georgiou, P., Eisenhofer, C., Kovács, L., & Reger, G. (2022, September). Lemmaless induction in trace logic. In *International Conference on Intelligent Computer Mathematics—CICM 2022* (pp. 191-208). Cham: Springer International Publishing.

(2) Georgiou, P., Gleiss, B., Bhayat, A., Rawson, M., Kovács, L., & Reger, G. (2022, October). The RAPID Software Verification Framework. In *Conference on Formal Methods in Computer-aided Design—FMCAD 2022* (p. 255).

(3) Georgiou, P., Hajdu, M., Kovács, L. (2023, July). Sorting without Sorts. In *EasyChair Preprint no. 10632—EasyChair 2023* (currently under review at FASE 2024)

3. Geplante weiterführende Aktivitäten

In terms of my PhD thesis, my contributions towards automating induction in first-order theorem provers for software verification will be summarised in a coherent thesis amplifying the motivation for the work during and prior to my netidee scholarship.

4. Anregungen für Weiterführung durch Dritte

An immediate continuation of the work lies in building a framework for functional programming languages/ algorithms based on our most recent work, similar to the RAPID framework for while-based languages.
A more general open question in automated theorem proving is how to find counterexamples. One of the major drawbacks of using FOL provers is that they mostly check validity of formulae but, due to semi-decidability of first-order logic with (undecidable) theories, rarely offer the possibility to produce counterexamples to validity. They simply diverge and run until the process times out.