

# 1 Einleitung

Das Boolean Satisfiability Problem (SAT), auf Deutsch Erfüllbarkeitsproblem der Aussagenlogik, ist das erste Problem, welches durch den „Satz von Cook“ als NP-vollständig bewiesen wurde. Dadurch wurden zahlreiche Forschungen im Bereich des SAT-Problems betrieben. Mitunter wurden mehrere SAT-Solver entwickelt, wie z.B. zChaff und Mini-SAT.

Die Entwicklung von effizienten SAT-Solvern ist seit dem Jahr 2002 stetig angestiegen. Jedoch wird für die Programmierung dieser Solver hauptsächlich eine objektorientierte Programmiersprache verwendet. Im Gegensatz dazu ist der Anteil von SAT-Solvern, welche komplett in einer rein funktionalen Programmiersprache geschrieben wurden, sehr gering.

Im Rahmen dieser Bachelorarbeit wird der Frage nachgegangen, wie eine mögliche Umsetzung eines „Conflict-driven clause learning“ (CDCL) SAT-Solvers in einer rein funktionalen Programmiersprache aussehen könnte. Das Ziel dieser Arbeit ist eine erste Implementierung eines CDCL SAT-Solvers in Haskell und mögliche Vorschläge wie deren Implementierung verbessert werden kann.

Anhand einer Literaturliste werden bestehende Algorithmen untersucht, wofür verschiedene Arbeiten über CDCL und näherstehende Arbeiten betrachtet werden. Die Literaturliste wurde gewählt, um bestehende Erforschungen in diesem Fachbereich zu erhalten.

In dieser Arbeit werden zuerst die verschiedenen Programmierparadigmen und die verwendete Programmiersprache Haskell in Kapitel 2 erläutert. Daraufhin werden SAT und verschiedene SAT-Solver in Kapitel 3 vorgestellt, wobei auch ein Einblick in die SAT-Competitions gewährt wird. Danach wird der Unterschied zwischen dem CDCL-Algorithmus und dem Davis-Putnam-Logemann-Loveland-Algorithmus (DPLL) in Kapitel 4 erläutert. Das darauffolgende Kapitel 5 beinhaltet eine Beschreibung für eine erste Umsetzung eines CDCL SAT-Solvers. Ein Vergleich der Implementierung mit anderen CDCL SAT-Solvern wird in Kapitel 6 durchgeführt. Anschließend werden in Kapitel 7 mögliche Verbesserungsmöglichkeiten und Algorithmen für die Implementierung vorgestellt. Im letzten Teil der Arbeit (Kapitel 8) wird ein Fazit über die gewonnenen Erkenntnisse ge-

## *1 Einleitung*

zogen als auch ein Ausblick über Verbesserungsmöglichkeiten für eine bessere Effizienz des SAT-Solvers gegeben.

<sup>1</sup>

---

<sup>1</sup>Kapitel 7 vielleicht streichen?