

In this thesis we deal with the problem of automatic proving (or disproving) mathematical conjectures using computer programs (usually called automated theorem provers). We address several issues that are important for a successful utilization of such programs. In Chapter 3 we examine how to store and reuse important pieces of mathematical knowledge in the form of lemmas. We investigate how this process can

be automatized, i.e. how a computer can construct and use lemmas without human guidance. The program we develop tries to shorten or to speed up the proofs of several conjectures from a common theory. It repeatedly extracts lemmas from the proofs it has already completed and uses the lemmas to improve the sets of premisses to produce more efficient proofs of the conjectures. In Chapter 4 we develop a new algorithm that tries to construct the optimal sets of premisses for proving and disproving mathematical conjectures. The algorithm semantically analyzes the conjectures and the set of premisses of the given theory to find the optimal subsets of the premisses. The algorithm uses an automated model finder to construct models that serve as counterexamples that guide the algorithm to find the optimal set of premisses. In Chapter 5 we use the algorithm to decide formulae in a wide range of modal systems. We develop an axiomatization of the modal system $S1$ that is based on equivalences. Then we use this basis to construct algebras in which we can decide the validity of modal formulae and rules in various extensions of $S1$. Finally we use the algorithm from Chapter 4 to automatically decide the validity of almost

50 formulae and rules in over 30 modal systems and to decide the relationships between the systems.