Part II
Automated Theorem Proving for Hybrid Systems

Overview After having developed formal specification logics and proof calculi for specifying and verifying safety-critical properties of hybrid systems in Part I, we now turn to practical and algorithmic implementation questions. In this part, we focus on the practical aspects of implementing the proof calculi from Part I. The calculi in Part I have already been designed for the needs of automated theorem proving, most notably with the free-variable and Skolemisation techniques from Chap. 2 and the compositional proof calculi from Part I. Immediate implementations of the proof calculi from Part I in automated theorem provers can prove examples of medium complexity directly. Yet, more complex case studies still require additional algorithmic techniques for achieving high-degree automation and good scalability properties. In Chap. 5, we refine the calculi from Part I to tableau procedures and present proof strategies that navigate among their nondeterminisms to help overcome the complexity issues of integrating real quantifier elimination as a decision procedure for real arithmetic.

In Chap. 6 we introduce the "differential invariants as fixed points" paradigm. We refine the differential induction techniques from Chap. 3 to a fully automatic verification algorithm for computing the required discrete and differential invariants of a hybrid system locally in a logic-based fixed-point loop.

The algorithmic refinement techniques developed in this part of the book add better automation to the proof approach from Part I. These algorithms are crucial for automating the formal verification of properties of complex hybrid systems like the ones we consider in Part III.