Flexible Re-enactment of Proofs*

Matthias Fuchs
Center for Learning Systems and Applications (LSA)
Fachbereich Informatik, Universität Kaiserslautern
Postfach 3049, 67653 Kaiserslautern, Germany
E-mail: fuchs@informatik.uni-kl.de

Abstract. We present a method for making use of past proof experience called flexible re-enactment (FR). FR is actually a search-guiding heuristic that uses past proof experience to create a search bias. Given a proof \( P \) of a problem solved previously that is assumed to be similar to the current problem \( A \), FR searches for \( \overline{P} \) in the "neighborhood" of \( P \) in order to find a proof of \( A \).

This heuristic use of past experience has certain advantages that make FR quite profitable and give it a wide range of applicability. Experimental studies substantiate and illustrate this claim.

1 Introduction

Automated deduction is essentially a search problem that gives rise to potentially infinite search spaces because of general undecidability. Despite these unfavorable conditions state-of-the-art theorem provers have gained a remarkable level of performance mainly due to (problem-specific) search-guiding heuristics and advanced implementation techniques. Nevertheless theorem provers can hardly rival a mathematician when it comes to proving "challenging" theorems. The main reason for this fact is also a major shortcoming of theorem provers: Unlike humans, theorem provers lack the ability to learn. Learning, however, is a key ability in any form of human problem solving, in particular in theorem proving.

The necessity to equip theorem provers with learning capabilities has been recognized quite early. But learning for theorem proving causes much more difficulties than learning in other areas of artificial intelligence because the premise that "small changes of the problem cause small changes of its solution" is not fulfilled at all. In theorem proving tiny variations of a problem specification can result in significant changes of the solution (proof). This circumstance complicates matters substantially.

Learning methods based on analogous proof transformation (e.g., [3, 2, 14, 18]) basically attempt to transform a source proof of a source problem solved previously into a target proof of a given target problem to be solved. Mostly, the mainly deterministic transformation procedure centers on some kind of analogy mapping obtained by comparing source and target problem. The transformation

* This work was supported by the Deutsche Forschungsgemeinschaft (DFG).
may include abstraction and planning steps. Occasionally, a little search may be involved in order to patch failures of the transformation procedure (e.g., [2]).

The prevailing determinism has the advantage that analogous proof transformation can be quite fast. The downside, however, is that source and target problem have to be very similar so that the source proof can be transformed into a target proof mainly with deterministic actions. For this reason, methods based on analogous proof transformation are preferably applied to inductive theorem proving. There, the inherent proof structures provide a suitable platform for such methods. However, for theorem proving in domains without any (recognizable) structure, these methods appear to be inappropriate.

An alternative approach to learning for theorem proving is to incorporate problem-solving experience into the search-guiding heuristic (e.g., [20, 23, 8, 11, 6]). That is, when solving a target problem, the source problem is in some way exploited by the search-guiding heuristic. The theorem prover still conducts a search, but the heuristic is biased towards a certain area of the search space depending on the source proof and on the way it is utilized by the heuristic. This approach has the advantage that the demands on similarity between source and target problem do not have to be as high as for methods based on proof transformation. Naturally we incur the usual overhead of search caused by exploring areas of the search space that do not contribute to the proof eventually found.

In this paper we investigate a method for incorporating past proof experience into a search-guiding heuristic called flexible re-enactment (FR). FR basically attempts to re-enact the source proof via search, if possible. Flexibility is achieved by also searching in the “neighborhood” of the source proof or by using some “standard” (non-learning) heuristic if the trace of the source proof is (temporarily) lost. Both re-enactment and flexibility are combined in a monolithic structure that allows for shifting smoothly between flexibility and re-enactment depending on the search space encountered.

FR was first introduced in [11]. In this paper we present a systematic experimental evaluation of FR that illustrates its performance and range of applicability. Furthermore, we counter a frequent and unjustified criticism of FR, namely to be a costly disguise for adding the source proof (in the form of lemmas) to the axiomatization of the target problem.

2 Theorem Proving

Theorem provers can attempt to accomplish a task in various ways. We focus here on so-called saturation-based theorem provers. This type of theorem prover is very common and is employed by provers based on the resolution method (e.g., [4]) or the Knuth-Bendix completion procedure (e.g., [1]). The principle working method of such a prover is to infer facts by applying given rules of inference, starting with a given set $Ax$ of axioms, until the goal $\lambda G$ (the theorem to be proved) can be shown to be a logical consequence of $Ax$. A proof problem $A$ is hence specified by $A = (Ax, \lambda G)$. 