F-028

The Implementation of a Proof Search System for Isabelle

何 成 † 鈴木秀男 ‡ 小林英恒 †
He Cheng Suzuki Hideo Kobayashi Hidetsune

1. Introduction

Automated reasoning is an important field in artificial intelligence. Proving mathematical formulas and verifying protocols are the two major aspects of automated reasoning.

Isabelle [1] is an interactive proof system for automated theorem proving. It has been used in a variety of domains such as formalized mathematics, hardware verification and security protocol verification.

However, writing proof scripts (proof steps) in Isabelle is not an easy work for most of users. One reason is that sometimes Isabelle will yield too many proof states to handle. Each state consists of useful information such as thN(theorem name), thCon(theorem content), step(proof step number) and sub-goals. Isabelle will not store these states after proof in order to reduce the need of too much memory. To review some states in the process of proof, users often need to prove again. That is often time-consuming because the executing speed of Isabelle now is not so fast that users often have to wait several minutes for proving one theory.

A proof search system – IsaDB (Isabelle Database) is proposed to solve the problem above.

2. Overview

IsaDB is mainly based on Isabelle and PostgreSQL database management system. Two software PHP and Apache are utilized for providing web service. All the sources could be obtained for free.

The main function of IsaDB is searching states by keywords of proof. Eight algebra theories are used as an application example in testing IsaDB. The system also supports:

1. Searching by common strings or strings containing logic symbols.
2. Printing different contents in different colors like Isabelle does.
3. Remote access by web browsers.

† Nihon University
‡ Tokyo Polytechnic University

3. Design

The idea of dealing with proof states from Isabelle can be divided into three parts: capturing, processing and using states.

- Capturing States: On one side, printed states will be cleared once Isabelle exits. Keeping them by copy-and-paste is unpractical. On the other side, many modules of Isabelle, including the state module, are realized by recursive functions which are like loops in procedural languages i.e. C language. So to extend Isabelle to export its states, it is better to modify its sources by using recursive ML functions to automatically get states while Isabelle is proving.
- Processing States: It should noticed that the states can not be directly inserted into database because they are not in right format. So it is necessary to convert them into a standard format in advance. And then all the data could be inserted very quickly by simple database commands.
- Using States: To use the states stored in the database, an efficient language which can provide interface between database and user is needed. PHP language [2] is selected in that it is a widely-used general-purpose scripting language and provides good interfaces for using many kinds of database like PostgreSQL.

4. Implementation

Corresponding to the ideas above, the implementation of IsaDB also contains three parts: state collection, database construction and interface design based on the first two.

Firstly, to get the states, Isabelle sources were modified and a new ML(Meta Language) [3] file was added to receive states from the proof process of Isabelle.

And then, a database was constructed to store these states for search. The work of the two parts is involved in programming in ML and PostgreSQL.

Finally, the main interface for search is written in PHP language. The function comprises four parts according to data flow:

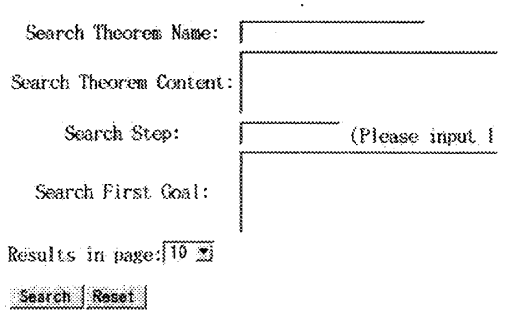

図1 The search interface of IsaDB

表1 Results of test

Item	Query	Time(s)
thN	BNTr8	0.252
thCon	$\forall x \in carrierD$	0.897
Step	100	0.004
FirGoal	$\forall x \in carrierD$	1.061
thCon; Step	$\forall x \in carrierD; 100$	0.004
FirGoal; Step	$\forall x \in carrierD; 100$	0.005

1. Input query.
2. Process query.
3. Generate search command.
4. Print results.

The search interface of IsaDB is shown in Fig.1.

The other interface to call IsaDB from Isabelle is written in Lisp language.

5. Application

To evaluate the implementation, IsaDB was applied to construct a proof database for eight files of algebra theory [4].

The theory contains 1807 lemmas (theorems) about groups, rings and modules. All 41240 proof states in proving the theory were collected and inserted into database by IsaDB.

Finally, proof state search could be done in a client's web browser which sends request to the server equipped with the database.

Part of data in this test is listed in Tab.1.

Item is the search field for keywords and FirGoal is the first goal of state. The timings were obtained from a 2.4GHz Intel PC with 128MB of RAM, and using Isabelle with Poly/ML, Apache 1.3.33, PHP 5.0.4 and PostgreSQL 8.0.1 in Linux 9.0.

The followings are some results got from the test:

1. Although the size of data inserting file is often 10-15 times of that of initial theory files, it is not a problem for IsaDB because PostgreSQL system can cope data in Gigabytes scale.
2. A search can mostly finish in one second. Search for first goal is a little slower than others because first goal is the major and biggest text object of state.
3. Advanced search is usually faster than single search. It can be seen from Tab.1 that the second searching time for first goal is reduced from 1.061 seconds to 0.005 seconds by accompanying with a step number. This owes to the step index created in PostgreSQL.

6. Conclusion

The proof search system, using several programming languages and free software, could deal with a great deal of states generated by a proof assistant Isabelle.

Our work is an attempt to apply standard database technology to automated theorem proving. All this things show that it is possible to use database technology to facilitate the work of data extraction, storing and searching in automated reasoning.

7. Related and Future Work

There is a work for constructing database for proof [5]. The main difference between the work and IsaDB is the method of state collection. At the meantime, IsaDB allows users on the internet to browse the states which have been produced by themselves or others.

There are some ways to extend this work. Other important data of Isabelle proof, for example theory content which contains type definitions and proof scripts, is also hoped to be added to the database. In addition, applying IsaDB to other theories for wider search for more states or other information will be done in the near future. The system is expected to be a bridge between database and automated reasoning.

Reference

[1] L. C. Paulson. Isabelle - A Generic Theorem Prover. Springer Verlag.

[2] 三木秀治. 初めての人のための PHP Web データベースプログラミング. MYCOM.

[3] L. C. Paulson. ML for the Working Programmer. CAMBRIDGE.

[4] Hidetsune Kobayashi, L. Chen, and H. Murao. Groups, Rings and Modules. http://afp.sourceforge.net/entries/Group-Ring-Module.shtml.

[5] 鈴木秀男, 船戸正和. 証明データベースの実装について. 日本数式処理学会 学会誌 vol.10 NO.2 2003.