The Efficiency of Automated Theorem Proving by Translation to Less Expressive Logics

Negin Arhami and Geoff Sutcliffe

Department of Computer Science
University of Miami

July 23rd, 2014

UNIVERSITY
OF MIAMI

Table of Contents

Automated Theorem Proving (ATP)

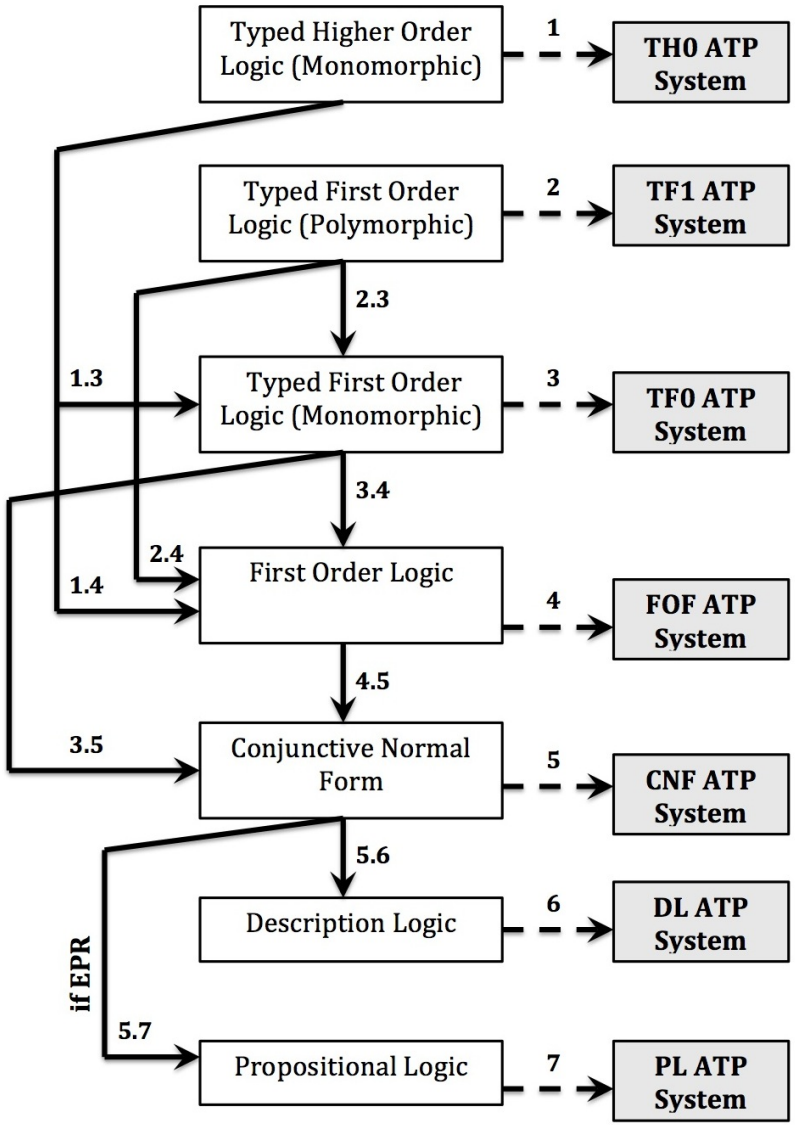
Developing automatic techniques and computer programs for checking whether the conjecture of a logic problem is a logical consequence

ATP system

A program that automatically checks whether the conjecture a problem is a logical consequence

Translator

A program that soundly translates a problem in one logic to another


Typed Higher Order Logic (Monomorphic)
1
TH0 ATP System
Typed First Order Logic (Polymorphic)
2
TF1 ATP System
2.3
1.3
Typed First Order Logic (Monomorphic)
3
TF0 ATP System
3.4
2.4
1.4
First Order Logic
4
FOF ATP System
4.5
3.5
Conjunctive Normal Form
5
CNF ATP System
5.6
Description Logic
6
DL ATP System
if EPR
5.7
Propositional Logic
7
PL ATP System

Thousands of Problems for Theorem Provers (TPTP)

- Comprehensive library of test problems for ATP systems and translators
- Software tools that facilitate using the ATP systems and translators
- General guidelines for evaluating and comparing ATP systems and translators
- TPTP syntaxes for logics

Sources of This Research

- TPTP library of sample problems
- ATP systems in SystemOnTPTP
- Translators in SystemB4TPTP

Related Conferences and Competitions

Conference on Automated Deduction (CADE)

International Joint Conference on Automated Reasoning (IJCAR)

CADE ATP System Competition (CASC)

- Evaluates the performance of ATP systems
  - Number of problems solved within a time limit
  - Average runtime for problems solved
- Main basis for selecting the ATP systems used in the experiments in this research
- Same evaluation scheme used in this research

Table of Contents

Processes and Tools

Label	Action			Tool
1	TH0	$\Rightarrow$	Proof	Isabelle-HOT 2013
1.3	TH0	$\Rightarrow$	TF0	Isabelle-2TF0 2013
1.4	TH0	$\Rightarrow$	FOF	Isabelle-2FOF 2013
2	TF1	$\Rightarrow$	Proof	Alt-Ergo 0.94
2.3	TF1	$\Rightarrow$	TF0	Why3-TF0 0.71
2.4	TF1	$\Rightarrow$	FOF	Why3-FOF 0.71
3	TF0	$\Rightarrow$	Proof	Princess 120604
3.4	TF0	$\Rightarrow$	FOF	Monotonox-2FOF e3c1636
3.5	TF0	$\Rightarrow$	CNF	Monotonox-2CNF e3c1636
4	FOF	$\Rightarrow$	Proof	Vampire 3.0
4.5	FOF	$\Rightarrow$	CNF	ECNF 1.8
5	CNF	$\Rightarrow$	Proof	Vampire 3.0 or iProver 1.0
5.6	CNF	$\Rightarrow$	DL	Saffron 1.0
5.7	CNF	$\Rightarrow$	PL	EGround 1.8
6	DL	$\Rightarrow$	Proof	HermiT 1.3.8
7	PL	$\Rightarrow$	Proof	MiniSat 2.2.0

Table of Contents

Saffron, a Translator for CNF to DL

Motivation

No translator was available from CNF to DL, which sits between EPR and Propositional Logic in terms of expressivity

Table : CNF and DL Equivalent Symbols

CNF	DL
Constants	Individuals
Predicates with one argument	Classes
Predicates with two arguments	Roles

Implementations

DL-able Clause

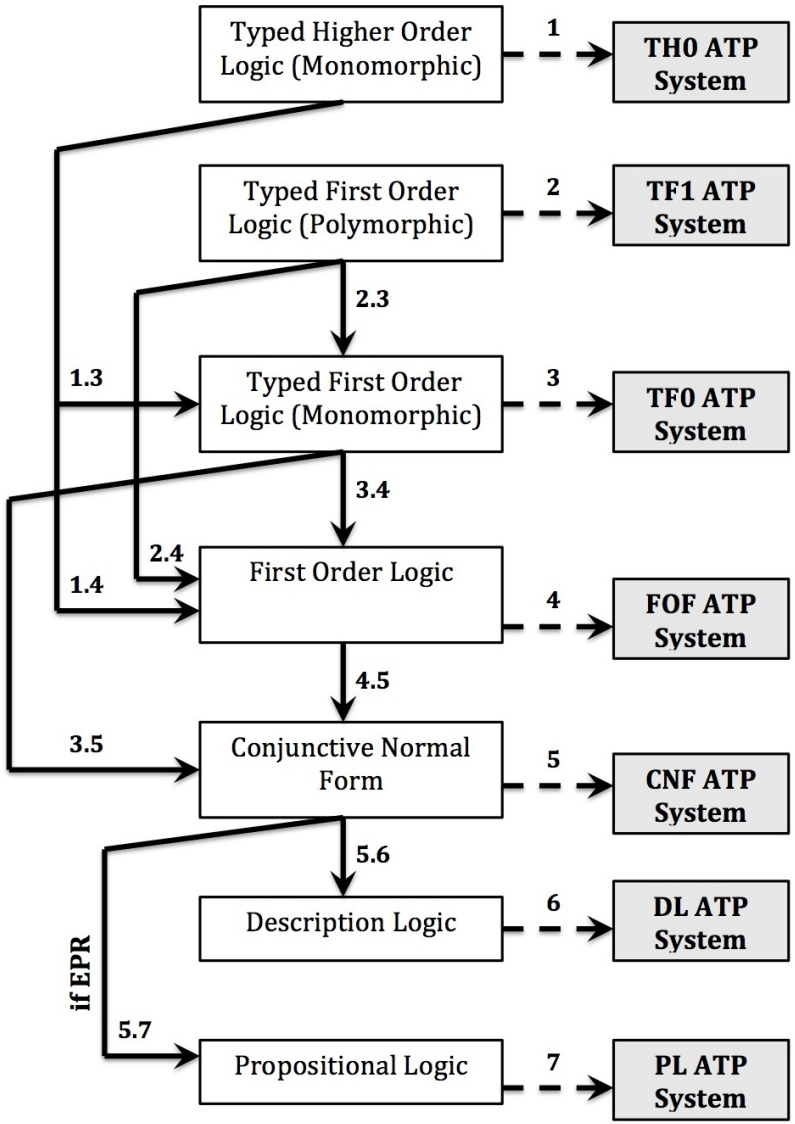
Clause with only constants and unary and binary predicates
Clause with a equivalent semantic in DL

Saffron

- Implemented in Prolog
- Translates clause by clause
- Each clause describes a characteristic of an individual, a class, a role, or the default class `Thing`.
  - Extracts all the constants, unary predicates, and binary predicates, and variables
  - Determines the form of the clause depending on these sets, and the polarities of the literals
- A problem might be fully or partially translated.

Table of Contents

Overview of The Experiment



Conjunctive Normal Form

Paths from 150 CNF Problems from CASC-J6

Path	Tools	#	Solved	Time
CNF	Vampire	150	123 (82%)	2.2
CNF.DL	Saffron▷HermiT	150	0 (0%)	0.0

Paths from 262 CNF Completely DL-able Problems

Path	Tools	#	Solved	Time
CNF	iProver	262	129 (49%)	2.2
CNF.PL	EGround▷MiniSAT	165	102 (61%)	0.0
CNF.DL	Saffron▷HermiT	262	69 (26%)	0.1

Paths from 262 CNF Completely DL-able Problems in Parallel

CPUs	Path	Unique	Solved	Time
1	CNF	129	129	2.2
2	CNF.DL	46	175	0.6

First Order Form

Paths from 500 FOF Problems

Paths	Tools	#	Solved	Time
FOF	Vampire	500	396 (79%)	3.1
FOF.CNF	ECNF▷(Vampire or iProver)	445	294 (58%)	1.5
FOF.CNF.DL	ECNF▷Saffron▷HermiT	441	20 (4%)	0.5
FOF.CNF.PL	ECNF▷EGround▷MiniSAT	13	13 (100%)	0.1

Paths from 500 FOF Problems in Parallel

CPUs	Path	Unique	Solved	Time
1	FOF	396	396	3.0
2	FOF.CNF	16	412	2.4
3	FOF.CNF.DL	13	425	2.1

Typed First Order Form - monomorphic

Paths from 97 TF0 Problems

Paths	Tools	#	Solved	Time
TF0.FOF	Monotonox-2FOF▷Vampire	88	36 (40%)	1.4
TF0.FOF.CNF	Monotonox-2FOF▷ECNF ▷Vampire	88	31 (35%)	1.0
TF0.CNF	Monotonox-2CNF▷Vampire	96	24 (25%)	1.8
TF0	Princess	97	12 (12%)	12.0
TF0.FOF.CNF.DL	Monotonox-2FOF▷ECNF ▷Saffron▷HermiT	10	5 (50%)	0.2
TF0.CNF.DL	Monotonox-2CNF▷Saffron ▷HermiT	15	2 (13%)	0.2
TF0.CNF.PL	Monotonox-2CNF▷EGround ▷MiniSAT	3	0 (0%)	0.0
TF0.FOF.CNF.PL	Monotonox-2FOF▷ECNF ▷EGround▷MiniSAT	3	0 (0%)	0.0

Paths from Typed First Order Form - monomorphic in Parallel

Paths from 97 TF0 Problems in Parallel

CPUs	Path	Unique	Solved	Time
1	TF0.FOF	36	36	1.4
2	TF0.FOF.CNF.DL	5	41	1.2
3	TF0	2	43	1.3
4	TF0.CNF.DL	1	44	1.3

Typed First Order Form - polymorphic

Paths	Tools	#	Solved	Time
TF1.FOF.CNF	Why3-FOF▷ECNF▷Vampire	987	348 (35%)	0.8
TF1.TF0.FOF	Why3-TF0▷Monotonox-2FOF ▷Vampire	957	329 (34%)	0.9
TF1.FOF	Why3-FOF▷Vampire	987	317 (32%)	0.8
TF1	Alt-Ergo	987	312 (31%)	1.0
TF1.TF0.CNF	Why3-TF0▷Monotonox-2CNF ▷Vampire	957	276 (28%)	1.7
TF1.TF0	Why3-TF0▷Princess	987	33 (3%)	12.6
TF1.TF0. FOF.CNF	Why3-TF0▷Monotonox-2FOF ▷ECNF ▷Vampire	48	16 (33%)	0.8
TF1.TF0. FOF.CNF.DL	Why3-TF0▷Monotonox-2FOF ▷ECNF ▷Saffron▷HermiT	957	2 (0%)	0.1
TF1.TF0.CNF. DL	Why3-TF0▷Monotonox-2CNF ▷Saffron▷HermiT	957	2 (0%)	0.5
TF1.FOF. CNF.DL	Why3-FOF▷ECNF ▷Saffron▷HermiT	957	0 (0%)	0.0
TF1.⋆.CNF.PL	⋆▷EGround▷MiniSAT	0	0 (0%)	0.0

Paths from Typed First Order Form - polymorphic in Parallel

Paths from 987 TF1 Problems in Parallel

CPUs	Path	Unique	Solved	Time
1	TF1.FOF.CNF	348	348	0.8
2	TF1	23	371	0.6
3	TF1.TF0.FOF	9	380	0.5
4	TF1.FOF	5	385	0.5

Typed Higher Order Form - monomorphic

Paths from 200 TH0 Problems

Paths	Tools	#	Solved	Time
TH0.FOF	Isabelle-2FOF▷Vampire	196	78 (39%)	9.0
TH0	Isabelle-HOT	200	78 (39%)	11.9
TH0.TF0.FOF	Isabelle-2TF0▷Isabelle-2FOF▷Vampire	192	77 (40%)	11.4
TH0.TF0.CNF	Isabelle-2TF0▷Monotonox-2CNF▷Vampire	194	52 (26%)	5.7
TH0.TF0	Isabelle-2TF0▷Princess	196	24 (12%)	14.9
TH0.TF0.FOF.CNF	Isabelle-2TF0▷Isabelle-2FOF▷ECNF ▷Vampire	192	0 (0%)	0.0
TH0.FOF.CNF	Isabelle-2FOF▷ECNF▷Vampire	193	0 (0%)	0.0
TH0.⋆.CNF.PL	⋆▷EGround▷MiniSAT	0	0 (0%)	0.0
TH0.⋆.CNF.DL	⋆▷Saffron▷HermiT	0	0 (0%)	0.0

Paths from Typed Higher Order Form - monomorphic in Parallel

Paths from 200 TH0 Problems in Parallel

CPUs	Path	Unique	Solved	Time
1	TH0.FOF	78	78	8.8
2	TH0	33	111	9.4
3	TH0.TF0.FOF	2	113	9.4
4	TH0.TF0.CNF	1	114	8.1

Table of Contents

Contributions and Conclusions

- Experiment
  - More TF0 and TF1 problems solved by translation
  - Generally more problems solved by translation to FOF or CNF
  - More CNF (with only DL-able clauses), FOF and TH0 problems solved through paths in parallel
- Saffron
  - A new tool offering new possibilities for reasoning by translation to DL
  - Significant numbers of uniquely solved CNF, FOF, and TF0 problems through DL-based paths
  - Saffron and HermiT combined into an ATP system called SafHer, now available in the SystemOnTPTP interface
  - Possibility that DL might provide a valuable alternative to using a ATP system such as iProver for solving EPR problems in the HWV domain of the TPTP

Future Work

- Use of other ATP systems, to determine the extent to which these results are dependent on the particular ATP systems used
- Use of other ATP systems might result in different unique solutions, which would it possible to effectively use more CPUs for alternative processing options.

Thank You