Implementing ATP Systems

Unit 10: Testing and Problem Libraries

Jens Otten

University of Potsdam

Outline

Problem Libraries: Motivation

- Important questions when developing ATP systems.
  - What is its performance compared to existing ATP systems?
  - Does a new strategy really improve performance?
  - Is the ATP system correct and/or complete?
- Important questions when applying ATP systems.
  - Which ATP systems are available? Where can I get them?
  - How fast are they? How well suited for specific problem class?
- Objectives:
  - Provide large collection of problems in a standardized syntax for testing and benchmarking ATP systems.
  - Put evaluation of ATP systems onto a firm basis and make meaningful system comparisons possible.
  - Measuring progress in ATP research.

ATP Problem Libraries: Requirements

- Easy to discover and obtain; provides guidelines for its use in evaluating ATP systems.
- Well structured and documented; provides statistics about the library as a whole.
- It is easy to use; the problems are provided in an easy-to-understand format, and conversion tools to other known syntax formats are included.
- It is large enough for statistically significant testing.
- It contains problems of varying difficulty.
- It assigns each problem a unique name and provides status and difficulty rating for each problem.
- Largest problem library: TPTP library (Sutcliffe '09).

TPTP Syntax for Representing Problems

- Uniform syntax for representing problems in first-order logic.
- Example: (SYN054+1)

$\neg(\exists x(Sx \wedge Qx))$	Axiom 1	(1)
$\forall(Px \Rightarrow (Qx \vee Rx))$	Axiom 2	(2)
$\neg(\exists xPx) \Rightarrow \exists yQy$	Axiom 3	(3)
$\forall x((Qx \vee Rx) \Rightarrow Sx)$	Axiom 4	(4)
$\exists x(Px \wedge Rx)$	Conjecture	(5)

- Block: *language(name,role,formula,source,useful_info)*. *language*=`thf`|`fof`|`cnf`; *role*=`axiom`|`conjecture` (e.g.); *source* and *useful_info* are optional.
-

```
%--------------------------------------------------------------------------
% File     : SYN054+1 : TPTP v4.0.1. Released v2.0.0.
% Domain   : Syntactic
% Problem  : Pelletier Problem 24
% Status   : Theorem
% Rating   : 0.00 v2.1.0
%--------------------------------------------------------------------------
fof(pel24_1,axiom, ( ~ ( ? [X] : ( big_s(X) & big_q(X) ) ) )).
fof(pel24_2,axiom, ( ! [X] : ( big_p(X) => ( big_q(X) | big_r(X) ) ) )).
fof(pel24_3,axiom, ( ~ ( ? [X] : big_p(X) ) => ? [Y] : big_q(Y) )).
fof(pel24_4,axiom, ( ! [X] : ( ( big_q(X) | big_r(X) ) => big_s(X) ) )).
fof(pel24,conjecture, ( ? [X] : ( big_p(X) & big_r(X) ) )).
%--------------------------------------------------------------------------
```

TPTP Syntax for Representing Resolution Proofs

- Block: *language(name,role,formula,source,useful_info).*

  source= `file(`*file_name,file_info*`)`
  `inference(`*inference_name,inference_info,parents*`)`

  inference_info lists additional information; *parents* is list of the (logical) parents; variable bindings captured in `bind`/2 terms.

- ```
  %----------------------------------------------------------------------------
  fof(1, axiom,~(?[X1]:(big_s(X1)&big_q(X1))),file('SYN054+1.p',pel24_1)).
  fof(2, axiom,![X1]:(big_p(X1)=>(big_q(X1)|big_r(X1))),file('SYN054+1.p',pel24_2)).
  fof(3, axiom,(~(?[X1]:big_p(X1))=>?[X2]:big_q(X2)),file('SYN054+1.p',pel24_3)).
  fof(4, axiom,![X1]:((big_q(X1)|big_r(X1))=>big_s(X1)),file('SYN054+1.p',pel24_4)).
  fof(5, conjecture,?[X1]:(big_p(X1)&big_r(X1)),file('SYN054+1.p',pel24)).
  ...
  fof(22,negated_conjecture,![X1]:(~(big_p(X1))|~(big_r(X1))),inference(fof_nnf,[],[6])).
  fof(23,negated_conjecture,
      ![X2]:(~(big_p(X2))|~(big_r(X2))),inference(variable_rename,[],[22])).
  cnf(24,negated_conjecture,(~big_r(X1)|~big_p(X1)),inference(split_conjunct,[],[23])).

  cnf(25,plain,(big_q(X1)|~big_p(X1)),inference(csr,[],[12,24])).
  cnf(26,plain,(~big_q(X1)),inference(csr,[],[9,21])).
  cnf(27,plain,(big_p(esk1_0)),inference(sr,[],[16,26])).
  cnf(28,plain,(~big_p(X1)),inference(sr,[],[25,26])).
  cnf(29,plain,($false),inference(sr,[],[27,28])).
  %----------------------------------------------------------------------------
  ```
  ```

The TPTP Library for Classical Logic

- Web: www.tptp.org (Sutcliffe/Suttner '98).
- TPTP v5.0.0 (September 2010): 18480 problems.
- 46 problem classes (domains), e.g., ALG (general algebra, 533 problems), ARI (arithmetic, 571) COL (combinatory logic, 239), COM (computing theory, 50), CSR (commonsense reasoning, 838), GRP (algebra/groups, 1090), MGT (management, 56), NLP (natural language, 520), NUM (number theory, 1207), PUZ (puzzles, 194), SET (set theory, 1395), SWV (software verification, 1390), SYN (syntactic, 1294).
- 7634 clausal (CNF), 7137 non-clausal (FOF) problems; 74%/86% with status Unsatisfiable/Theorem (of CNF/FOF).
- Provides tptp2X tool for converting problems in the library into syntax of existing ATP systems.
- Problems are given a unique name: `DDD.NNN+V[.SSS].p` , where `DDD` is mnemonic of the domain, `NNN` is number of the problem, `V` is version number, and `SSS` is size of the instance. E.g. `SYN054+1.p` is the 54th problem in the domain SYN.

Rating and Status Information

- Rating indicates difficulty of a problem with respect to current state-of-the-art ATP systems.
- Rating defined as ratio of state-of-the-art ATP systems that do *not* solve a problem within a given time limit.
- E.g. a rating of 0.30 indicates that 30% of the state-of-the-art systems do *not* solve the problem.

- Status is, e.g., `Theorem` or `Countersatisfiable` (FOF problems), `Unsatisfiable` or `Satisfiable` (CNF problems), `Unknown` or `Open`.
- Problems with status `Unknown` or `Open` have not been solved by any state-of-the-art ATP system.
- For `Open` problems it is unknown if they are theorems or not (the abstract problem has not been solved so far).

Performance of leanCoP 1.0 on TPTP

- Tested on all 3644 FOF problems of TPTP library v3.3.0.

System	leanTAP	leanCoP	SETHEO	Otter	Prover9	E
Version	2.3	1.0	3.3	3.3	Dec-2007	0.999
Proved	375	1004	1192	1310	1677	2250
[%]	10%	28%	33%	36%	46%	62%
0s to 1s	351	787	864	987	1281	1760
1s to 10s	12	84	205	183	197	229
10s to 100s	11	74	62	106	141	192
100s to 600s	1	59	61	34	58	69
0.00...0.24	22.8%	56.2%	63.9%	72.2%	72.8%	77.7%
0.25...0.49	5.9%	26.0%	34.2%	39.7%	69.9%	84.5%
0.50...0.74	2.2%	7.1%	8.5%	3.0%	28.2%	69.1%
0.75...1.00	0.4%	0.0%	1.5%	0.7%	2.5%	18.5%

The ILTP Library for Intuitionistic Logic

- Web: `www.iltp.de` (Raths/Otten/Kreitz '05).
- ILTP v1.1.2 (January 2007): 2754 problems.
- Propositional/first-order part: 274/2550 problems.
- Provides intuitionistic status information: either `Theorem`, `Non-Theorem`, `Unsolved` or `Open`.
- Provides intuitionistic rating information (like TPTP rating).
- For rating information eight state-of-the-art systems were chosen according to their performance on the ILTP library.
- Provides converting tool and list of intuitionistic ATP systems.
- Puts evaluation of intuitionistic ATP systems onto a firm basis and makes meaningful systems comparisons possible.

Performance of ileanCoP

- Tested on all 2550 first-order problems of ILTP library v1.1.2.

System	JProver	ft_{Prolog}	ileanSeP	ileanTAP	ft_C	ileanCoP
Version	11-2005	1.23	1.0	1.17	1.23	1.0
Solved	268	299	313	364	315	690
Proved	264	299	309	334	311	610
Refuted	4	0	4	30	4	80
0 to <1 s	243	285	249	351	299	557
1 to <10 s	11	9	33	6	8	46
10 to <100 s	8	1	19	7	5	44
100 to 600 s	6	4	12	0	3	43
rated 0.0	203	193	176	203	203	203
to ≤0.7	63	99	85	127	101	154
to ≤1.0	2	7	52	34	11	340

- 258 problems could only be solved by ileanCoP.

The QMLTP Library for Modal Logic

- Web: www.iltp.de/qmltp (Raths/Otten '09).
- QMLTP v0.2 (Juli 2009): 200 problems in 7 domains.
- The TPTP syntax is extended by the modal operators "box" and "dia": $\Box F$ and $\Diamond F$ represented by "box:F" and "dia:F".
- The multi-modal operators are expressed by "box(i)" and "dia(i)" with constant i.
- Format files are used to convert problems into syntax of existing ATP systems (using tptp2X tool).
- Syntax will be changed to "#box:F" and "#dia:F"!
- First official release: Beginning of 2011.
- Partly funded by National Science Foundation (DFG) within the project "ATP in First-Order Modal Logic".

Example: Modal Syntax for Representing Problems

```
%--------------------------------------------------------------------------
% File     : SYM002+1 : QMLTP v0.2
% Domain   : Syntactic (modal)
% Problem  : Converse Barcan scheme instance
% Version  : Especial.
% English  : If it is necessary that for all x f(x), then for all x
%            necessarily f(x)
% Refs     : [Brc46] [1] R. C. Barcan. A functional calculus of first
%            order based on strict implication. Journal of Symbolic Logic
%            11:1-16, 1946.
% Source   : [Brc46]
% Names    : Instance of the converse Barcan formula
% Status: S4 cumulative : Theorem
% Rating: S4 cumulative : 0.00 v0.2
%
% Comments :
%--------------------------------------------------------------------------

fof(con,conjecture,
(( box : ( ! [X] : ( f(X) ) ) ) => ( ! [X] : ( box : ( f(X) ) ) ))).

%--------------------------------------------------------------------------
```

CASC: The ATP System Competition

- Web: www.tptp.org/CASC (Sutcliffe '10/'09/'08/...).
- Yearly competition that evaluates the performance of sound, fully automatic ATP systems.
- Several divisions, e.g.,
  - FOF: Valid first-order problems.
  - FNT: Invalid first-order problems.
  - CNF: Valid first-order problems in clausal form.
  - SAT: Satisfiable propositional problems.
  - THF: Typed higher-order problems.
  - TFA: Valid typed first-order problems with arithmetic.
- Typical between 75 and 200 problems in each division.
- Typical time limit of about 300 seconds.
- Winners in 2010 (CASC-J5) are, e.g., Vampire, E, iProver, LEO-II, Waldmeister, leanCoP-Ω.