Can a Higher-Order and a First-Order Theorem Prover Cooperate?

Christoph Benzmüller (Saarland University, Germany)

joint work with: Volker Sorge, Manfred Kerber (U Birmingham, UK)

Mateja Jamnik (U Cambridge, UK)

Overview: Issues of this Talk

- Computer-supported Mathematics
  - representation does matter
- Automation of Mathematical Reasoning
  - higher-order may outperform first-order in certain domains
- Automation of Higher-Order Theorem Proving (HOTP)
  - cooperation with a first-order theorem proving (FOTP) is beneficial
- Architectures supporting System Integrations
  - agent-based reasoning with OANTS
- Problem Libraries such as TPTP
  - should support alternative (higher-order) problem representations

Computer Maths: Representation Matters

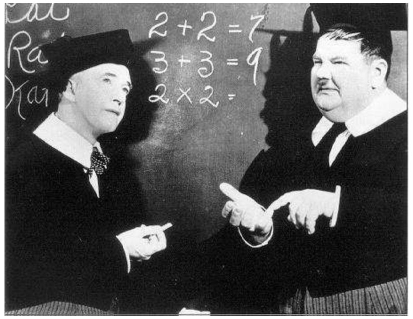
- Computer-supported Mathematics / Mathematics Assistance Systems
  - full automatization not realistic and only partly desireable
  - support for collaboration mathematician and computer is needed
  - interaction should be based on expressive languages
  - fact: maths in practice uses higher-order constructs
  - fact also: prominent proof assistents already support higher-order logic

- Example:

	textbooks	higher-order logic	first-order logic
$\mathcal{P}(A)$	$\{x \mid x \subseteq A\}$ $\mathcal{P}(\emptyset)$ is finite	$\lambda x.x \subseteq A$ $\mathrm{finite}(\mathcal{P}(\emptyset))$	$x \in \mathcal{P}(A) \Leftrightarrow x \subseteq A$ … less nice …
$\mathrm{Im}(F, A)$	$\{y \mid \exists x.x \in A \land y = F(x)\}$	$\lambda y.\exists x.x \in A \land y = F(x)$	see TPTP (terrible)

Computer Maths: Representation Matters

- Idea: Start with higher-order representations in a mathematics assistance system and combine higher-order and first-order (and propositional) reasoning (supported by tranformational mappings)



- Test Problems:
  - 45 theorems on sets, relations, and functions
  - taken from the TPTP domain "SET"
  - also used in paper on Saturate system [GanzingerStuber-IJCAR-04]
  - we added some problems that cannot be solved by any FOTP

- Conciseness of Higher-Order Representations:
  - 45 problem formulations (required defintions + theorems) fit on 1,5 page
  - not possible in first-order without λ-abstraction

Computer Maths: Representation Matters

- Examples of Basic Definitions on Sets and Relations

$\_ \in \_$	$:=$	$\lambda x, A.[A x]$
$\emptyset$	$:=$	$[\lambda x.\bot]$
$\_ \cap \_$	$:=$	$\lambda A, B.[\lambda x.x \in A \wedge x \in B]$
$\_ \cup \_$	$:=$	$\lambda A, B.[\lambda x.x \in A \vee x \in B]$
$\_ \setminus \_$	$:=$	$\lambda A, B.[\lambda x.x \in A \vee x \notin B]$
$\mathrm{Meets}(\_,\_)$	$:=$	$\lambda A, B.[\exists x.x \in A \wedge x \in B]$
$\mathrm{Pair}(\_,\_)$	$:=$	$\lambda x, y.[\lambda u, v.u = x \wedge v = y]$
$\_ \times \_$	$:=$	$\lambda A, B.[\lambda u, v.u \in A \wedge v \in B]$
$\mathrm{Subrel}(\_,\_)$	$:=$	$\lambda R, Q.[\forall x, y.R x y \Rightarrow Q x y]$
$\mathrm{IsRelOn}(\_,\_,\_)$	$:=$	$\lambda R, A, B.[\forall x, y.R x y \Rightarrow x \in A \wedge y \in B]$
$\mathrm{RestrictRDom}(\_,\_)$	$:=$	$\lambda R, A, B.[\lambda x, y.x \in A \wedge R x y]$

Display in UI as
$$A \times B = \{(u, v) | u \in A \wedge v \in B\}$$

...

- Examples of the Test Problems

SET171 + 3	.67	$\forall X, Y, Z.X \cup (Y \cap Z) = (X \cup Y) \cap (X \cup Z)$
SET611 + 3	.44	$\forall X, Y.(X \cap Y = \emptyset) \Leftrightarrow (X \setminus Y = X)$
SET624 + 3	.67	$\forall X, Y, Z.\mathrm{Meets}(X, Y \cap Z) \Leftrightarrow \mathrm{Meets}(X, Y) \vee \mathrm{Meets}(X, Z)$
SET646 + 3	.56	$\forall x, y.\mathrm{Subrel}(\mathrm{Pair}(x, y), (\lambda u.\top) \times (\lambda v.\top))$
SET670 + 3	1.0	$\forall Z, R, X, Y.\mathrm{IsRelOn}(R, X, Y) \Rightarrow \mathrm{IsRelOn}(\mathrm{RestrictRDom}(R, Z), Z, Y)$

...

Fairness of the Experiment

- Observation:
  - complete encodings of set theory in higher-order (comprehension via λ-abstraction, Boolean and functional extensionality, ...)

  vs.

  - incomplete and sometimes artificially tailored (useful lemmata) problem formulations in TPTP

- Example: TPTP171+3

Assumptions:	$\forall B, C, x. x \in (B \cup C) \Leftrightarrow (x \in B \vee x \in C)$	(1)	
	$\forall B, C, x. x \in (B \cap C) \Leftrightarrow (x \in B \vee x \in C)$	(2)	
	$\forall B, C. B = C \Leftrightarrow (B \subseteq C \wedge C \subseteq B)$	(3)	
	$\forall B, C. B \cup C = C \cup B$	(4)	derivable from 1,3,6
	$\forall B, C. B \cap C = C \cap B$	(5)	derivable from 2,3,6
	$\forall B, C. B \subseteq C \Leftrightarrow (\forall x. x \in B \Rightarrow x \in C)$	(6)	
	$\forall B, C. B = C \Leftrightarrow (\forall x. x \in B \Leftrightarrow x \in C)$	(7)	derivable from 3,6
Proof Goal:	$\forall X, Y, Z. X \cup (Y \cap Z) = (X \cup Y) \cap (X \cup Z)$	(8)	

- Hence: Our Comparison is Unfair

  $\rightarrow$ our higher-order problem formulations are more general and non-tailored

HOTP may outperform FOTP

- Observation not new:
  - TPS [see papers on TPS]
  - LEO [CADE-98,Benzmüller-PhD]
  - OMEGA-OANTS [KI-01]
  - others ...

- New:
  - Combination of HOTP and FOTP may even perform better

- Approach:
  - Make use of complementary strenghts of both worlds
  - Our HOTP of choice: LEO (extensional higher-order resolution)
  - Our FOTP of choice: Bliksem [Nivelle-99]
  - Our integration means of choice: ΩANTS [AIMSA-98,Sorge-PhD]

SET171+3: A Motivating Example

Problem: $\forall B, C, D. C \cup (B \cap D) = (C \cup B) \cap (C \cup D)$

$[\forall B, C, D. C \cup (B \cap D) = (C \cup B) \cap (C \cup D)]^F$

$\downarrow$ def.-expansion, cnf

$\downarrow$ B, C, D Skolem const.

$[(\lambda x. Bx \vee (Cx \wedge Dx)) = (\lambda x. (Bx \wedge Cx) \vee (Cx \wedge Dx))]^F$

$\downarrow$ unification constraint

$[(\lambda x. Bx \vee (Cx \wedge Dx)) =^? (\lambda x. (Bx \wedge Cx) \vee (Cx \wedge Dx))]$

$\downarrow$ f-extensionality

$\downarrow$ x new Skolem constant

$[(Bx \vee (Cx \wedge Dx)) =^? ((Bx \wedge Cx) \vee (Cx \wedge Dx))]$

$\downarrow$ B-extensionality

$[(Bx \vee (Cx \wedge Dx)) \Leftrightarrow ((Bx \wedge Cx) \vee (Cx \wedge Dx))]^F$

$\downarrow$ cnf, factor., subsumption

$[Bx]^F$
$[Bx]^T \vee [Cx]^T$
$[Bx]^T \vee [Dx]^T$
$[Cx]^F \vee [Dx]^F$

Propositional Problem!!!

within LEO or within FOTP? $\downarrow$ propositional reasoning

$\square$

SET624+3: Direct Mapping into FO

Problem: $\forall X, Y, Z.\mathrm{Meets}(X, Y \cap Z) \Leftrightarrow \mathrm{Meets}(X, Y) \vee \mathrm{Meets}(X, Z)$

$$[\forall X, Y, Z.\mathrm{Meets}(X, Y \cap Z) \Leftrightarrow \mathrm{Meets}(X, Y) \vee \mathrm{Meets}(X, Z)]^F$$

$\downarrow$ def.-expansion

$$[\exists x.(Bx \wedge (Cx \vee Dx)) \Leftrightarrow ((\exists x.Bx \wedge Cx) \vee (\exists x.Bx \wedge Dx)]^F$$

$\downarrow$ cnf

26 FO-like clauses

within LEO?
within FOTP?

SET646+3: No Proof Search

Problem: $\forall x_\alpha, y_\beta.\mathrm{Subrel}(\mathrm{Pair}(x, y), (\lambda u_\alpha.\top) \times (\lambda v_\beta.\top))$

$[\forall x, y.\mathrm{Subrel}(\mathrm{Pair}(x, y), (\lambda u.\top) \times (\lambda v.\top))]^F$

$\downarrow$ def.-expansion

$[\forall x, y, u, v.(u = x \wedge v = x) \Rightarrow ((\neg\bot) \wedge (\neg\bot))]^F$

$\downarrow$ cnf

...

$[\bot]^T \vee [\bot]^T = \Box$

...

SET611+3: Repeated Extensionality

Problem: $\forall A, B.(A \cap B = \emptyset) \Leftrightarrow (A \setminus B = A)$

$[\forall A, B.(A \cap B = \emptyset) \Leftrightarrow (A \setminus B = A)]^F$

$\downarrow$ def.-expansion

$[\forall A, B. \quad (\lambda x.(A x \wedge B x)) = (\lambda x.\bot)$
$\Leftrightarrow (\lambda x.(A x \wedge \neg B x)) = (\lambda x.A x)]^F$

$\downarrow$ cnf, A, B Skolem

(1) $[(\lambda x.(A x \wedge B x)) = (\lambda x.\bot)]^T \vee [(\lambda x.(A x \wedge \neg B x)) = (\lambda x.A x)]^T$
(2) $[(\lambda x.(A x \wedge B x)) =^? (\lambda x.\bot)] \vee [(\lambda x.(A x \wedge \neg B x)) =^? (\lambda x.A x)]$
$([(\lambda x.(A x \wedge B x)) = (\lambda x.\bot)]^F \vee [(\lambda x.(A x \wedge \neg B x)) = (\lambda x.A x)]^F)$

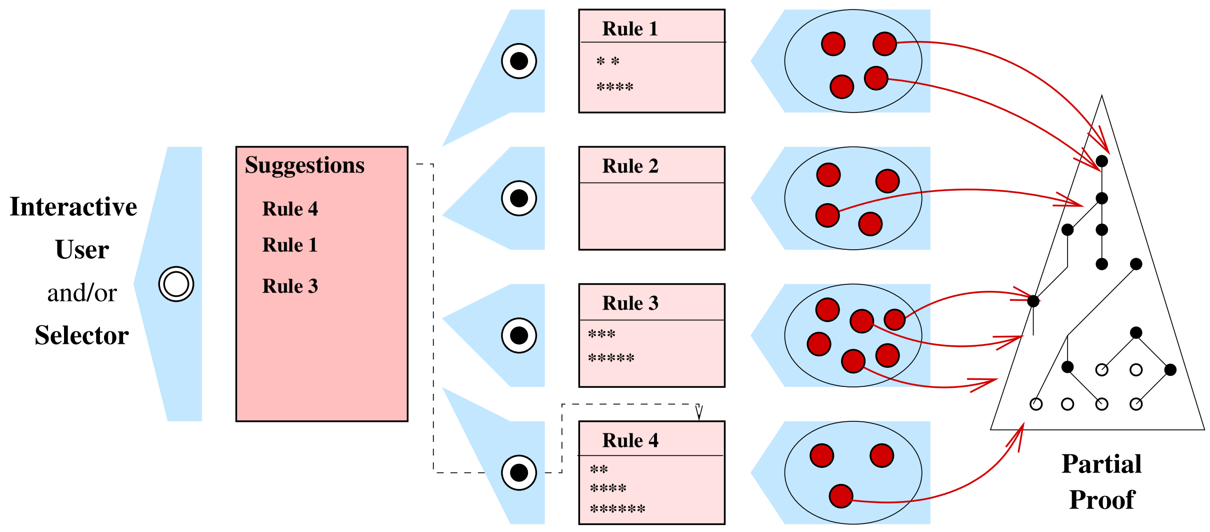
$\downarrow$ several rounds
$\downarrow$ of B&f-ext.
$\downarrow$ and cnf

inconsistent set of FO-like clauses

within LEO?
within FOTP?

HOTP-FOTP: Modeling in ΩANTS

- ΩANTS:
  - distributed suggestion mechanism for interactive theorem proving
  - blackboard architecture, supports redefinition of agents at run-time
  - automation of proof search possible [Calculemus-00]



HOTP-FOTP: Modeling in ΩANTS

OLD Solution

- $\dfrac{}{\text{HO-goal}}\ \text{LEO(LEO-params)}$
- $\dfrac{\text{Conjunction-of-FO-clauses}}{\text{HO-goal}}\ \text{LEO-with-partial-result(LEO-params)}$
- $\dfrac{}{\text{FO-goal}}\ \text{FOTP(FOTP-params)}$

NEW Solution

- $\dfrac{}{\text{HO-goal}}\ \text{LEO(LEO-params)}$
- $\dfrac{}{\text{HO-goal}}\ \text{LEO+FOTP(LEO-partial-proof,FO-clauses,FO-proof,LEO-params)}$

Experiments: Results (I)

SET	Rat.	Vampire 7	LEO			LEO + FOTP				
			Strat.	Cl.	Time	Cl.	Time	FOcl	FOtm	GenCl
014+4	.67	.01	ST	41	.16	34	6.76	19	.01	7
017+1	.56	.03	EXT	3906	57.52	25	8.54	16	.01	74
→ 066+1	1.00	--	--	--	--	26	6.80	20	10	56
067+1	.56	.04	ST	6	.02	13	.32	16	.01	12
→ 076+1	.67	.00	--	--	--	10	.47	18	.01	35
086+1	.22	.04	ST	4	.01	4	.01	N/A	N/A	N/A
→ 096+1	.56	.03	--	--	--	27	7.99	14	.01	25
143+1	.67	68.71	EIR	37	.38	33	7.93	18	.01	19
171+3	.67	108.31	EIR	36	.56	25	4.75	19	.01	20
580+3	.44	14.71	EIR	25	.19	6	2.73	8	.01	13
601+3	.22	168.40	EIR	145	2.20	55	4.96	8	.01	13
606+3	.78	62.02	EIR	21	.33	17	10.8	15	.01	5
607+3	.67	65.57	EIR	22	.31	17	7.79	15	.01	6
609+3	.89	161.78	EIR	37	.60	26	6.50	19	10	17
611+3	.44	60.20	EIR	996	12.69	72	32.14	38	.01	101
612+3	.89	113.33	EIR	41	.54	18	3.95	6	.01	7
614+3	.67	157.88	EIR	38	.46	19	4.34	16	.01	17
615+3	.67	109.01	EIR	38	.57	17	3.59	6	.01	9
→ 623+3	1.00	--	EXT	43	8.84	23	9.54	10	.01	14
624+3	.67	.04	ST	4942	34.71	54	9.61	46	.01	212
630+3	.44	60.39	EIR	11	.07	6	.08	8	10	4

Experiments: Results (II)

SET	Rat.	Vampire 7	LEO			LEO + FOTP				
			Strat.	Cl.	Time	Cl.	Time	FOcl	FOtm	GenCl
640+3	.22	70.41	EIR	2	.01	2	.01	N/A	N/A	N/A
646+3	.56	59.63	EIR	2	.01	2	.01	N/A	N/A	N/A
647+3	.56	64.21	EIR	26	.15	13	.30	13	.01	15
648+3	.56	64.22	EIR	26	.15	14	.30	13	.01	16
649+3	.33	63.77	EIR	45	.30	29	5.49	12	.01	16
651+3	.44	63.88	EIR	20	.10	11	.16	10	10	11
657+3	.22	1.44	EIR	2	.01	2	.01	N/A	N/A	N/A
669+3	.56	.34	EI	35	.22	35	.23	N/A	N/A	N/A
→ 670+3	1.00	--	EXT	15	.17	17	.36	16	.01	6
671+3	.78	218.02	EIR	78	.64	7	2.71	10	.01	14
→ 672+3	1.00	--	EXT	27	.40	30	.70	21	.01	14
673+3	.78	47.86	EIR	78	.65	14	5.66	14	.01	16
680+3	.33	.07	ST	185	.88	29	4.61	18	.01	24
683+3	.22	.06	ST	46	.20	35	8.90	18	10	24
684+3	.78	.33	ST	275	2.45	46	5.95	26	.01	47
686+3	.56	.11	ST	274	2.36	46	5.37	26	.01	46
→ 716+4	.89	--	ST	39	.45	18	3.81	18	.01	118
→ 724+4	.89	--	EXT	154	2.75	18	7.21	15	10	23
→ 741+4	1.00	--	--	--	--	--	--	--	--	--
→ 747+4	.89	--	ST	34	.46	25	1.11	18	10	10
→ 752+4	.89	--	--	--	--	50	6.60	48	.01	4363
→ 753+4	.89	--	--	--	--	15	3.07	12	10	19
764+4	.56	.02	EI	9	.05	8	.04	N/A	N/A	N/A
→ 770+4	.89	--	--	--	--	--	--	--	--	--

HOTP-FOTP: Soundness and Completeness

Soundness

- LEO's calculus is sound
- Bliksem's calculus is sound
- Crucial part:
  - transformation from FO-like clause in LEO to real FO clauses in Bliksem must preserve satisfiability
  - we use TRAMPs [Meier00] injective mapping

$$P(f(a)) \longrightarrow @^1_{\text{pred}}(P, @^1_{\text{fun}}(f, a))$$

Completeness

- LEO's calculus is Henkin complete (the implementation of LEO is not though)
- Completeness of the cooperative approach relies on the completeness of LEO

HOTP–FOTP: Problems

Generation of proof-objects

- How can we obtain a common proof object?
  - solved since Tuesday (LPAR "programming session" with Volker)

Leibniz equality (and other definitions of equality)

- Leibniz equality: $=$ can be defined as $\lambda x.\lambda y.\forall P.P(x) \Rightarrow P(y)$
- Example: $a = b \Rightarrow f(a) = f(b)$

Primitive equality	Leibniz equality
$[a = b]^T$ $[f(a) = f(b)]^F$	$[P(a)]^F \vee [P(b)]^T$ $[Q(f(a))]^T$ $[Q(f(b))]^F$
refutable in LEO and Bliksem	refutable only in LEO $P \leftarrow \lambda x.Q(f(x))$

Related Work

- Denzinger/Fuchs [IJCAI-99]:

  TECHS system

  – only cooperation of first-order systems

-

Andreas Meier [CADE-00]	Joe Hurd [CADE-02]	Jia Meng, Larry Paulson [IJCAR-04]
TRAMP, generic interface between OMEGA and FOTPs	generic interface between HOL and FOTPs	interface between Isabelle and Vampire

  – no calls to FOTP from within automated HO proof search

Summary

- Computer-supported Mathematics
  - representation does matter
- Automation of Mathematical Reasoning
  - higher-order may outperform first-order in certain domains
- Automation of Higher-Order Theorem Proving (HOTP)
  - cooperation with a first-order theorem proving (FOTP) is beneficial
- Architectures supporting System Integrations
  - agent-based reasoning with OANTS
- Problem Libraries such as TPTP
  - should support alternative (e.g. higher-order) problem representations

And Finally ...

I can fully recommend TeXmacs as scientific editor

Verified Mathematical Texts: Motivation