

Alignments and Their Applications

Thibault Gauthier Cezary Kaliszyk Karol Pąk

Logipedia Meeting

January 22, 2019

Proof Interoperability

Proof Analysis

- Comparing, Presentation, Search...

Proof Auditing

- HOL/Import, HOL Zero, ...

Re-use and Combining

- Particularly useful if shallow

Alignments Applications

Organizing data (logical framework).

- Eliminate duplicates.
- Regroup theorems.

Importing an external library (import HOL Light).







- Shallow embedding.
- Flexible mappings.

Proving.

- Premise selection: Learning relation between similar concepts.

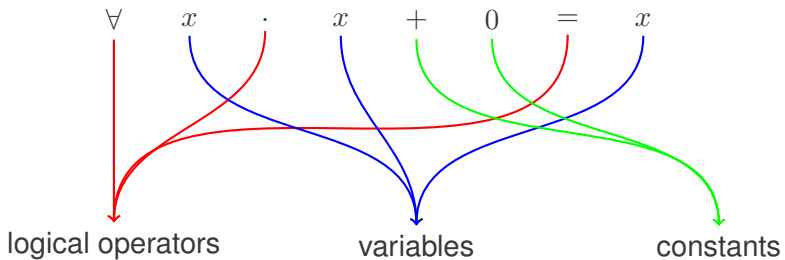
Conjecturing.

- Discovery of dual theorems.

Proof Assistant	Theorems	Constants
Mizar 	51086	9172
Coq 	23320	4841
HOL4 	16476	2247
HOL Light 	16191	820
Isabelle/HOL 	14814	1076
Matita 	1712	629

How to find alignments

- Find properties: associativity, commutativity, nilpotence, distributivity, injectivity,
- Rank pair of constants according to a score based on the number of their common properties.
- Improve scores based on already found alignments



Constant

Library 1 0
 +
 ×

Library 2 \emptyset
 \cup
 \cap

$$\forall x, y. x + y = y + x$$

Constant

Library 1 0
 +
 ×

Library 2 \emptyset
 \cup
 \cap

Commutativity: $\forall x, y. x \square y = y \square x$

	Constant	Properties
	0	
Library 1	+	$C[+]$
	\times	$C[\times]$
	\emptyset	
Library 2	\cup	$C[\cup]$
	\cap	$C[\cap]$

$$\forall x, y. x + 0 = x$$

	Constant	Properties
	0	
Library 1	+	$C[+]$
	\times	$C[\times]$
	\emptyset	
Library 2	\cup	$C[\cup]$
	\cap	$C[\cap]$

Neutral: $\forall x. x \square_1 \square_2 = x$

	Constant	Properties
Library 1	0	$N[+, 0]$
	+	$C[+] N[+, 0]$
	\times	$C[\times]$
Library 2	\emptyset	$N[\cup, \emptyset]$
	\cup	$C[\cup] N[\cup, \emptyset]$
	\cap	$C[\cap]$

$$\forall x, y. x \times 0 = 0$$

	Constant	Properties
Library 1	0	$N[+, 0]$
	+	$C[+] N[+, 0]$
	\times	$C[\times]$
Library 2	\emptyset	$N[\cup, \emptyset]$
	\cup	$C[\cup] N[\cup, \emptyset]$
	\cap	$C[\cap]$

Absorbent: $\forall x. x \sqcap_1 \sqcap_2 = \sqcap_2$

	Constant	Properties
Library 1	0	$N[+, 0] A[\times, 0]$
	+	$C[+] N[+, 0]$
	\times	$C[\times] A[\times, 0]$
Library 2	\emptyset	$N[\cup, \emptyset] A[\cap, \emptyset]$
	\cup	$C[\cup] N[\cup, \emptyset]$
	\cap	$C[\cap] A[\cap, \emptyset]$

$$\forall x, y, z. x \times (y + z) = x \times y + x \times z$$

	Constant	Properties
Library 1	0	$N[+, 0] A[\times, 0]$
	+	$C[+] N[+, 0]$
	\times	$C[\times] A[\times, 0]$
Library 2	\emptyset	$N[\cup, \emptyset] A[\cap, \emptyset]$
	\cup	$C[\cup] N[\cup, \emptyset]$
	\cap	$C[\cap] A[\cap, \emptyset]$

Distrib: $\forall x, y, z. x \sqcap_1 (y \sqcap_2 z) = (x \sqcap_1 y) \sqcap_2 (x \sqcap_1 z)$

	Constant	Properties
Library 1	0	$N[+, 0] A[\times, 0]$
	+	$C[+] N[+, 0] D[\times, +]$
	\times	$C[\times] A[\times, 0] D[\times, +]$
Library 2	\emptyset	$N[\cup, \emptyset] A[\cap, \emptyset]$
	\cup	$C[\cup] N[\cup, \emptyset] D[\cup, \cap] D[\cap, \cup]$
	\cap	$C[\cap] A[\cap, \emptyset] D[\cup, \cap] D[\cap, \cup]$

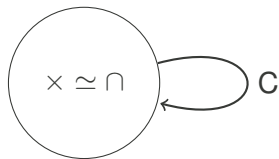
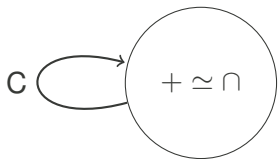
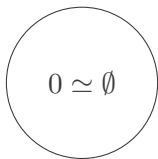
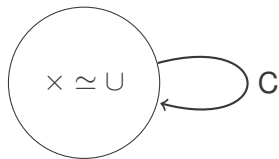
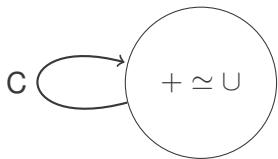
$$+ \simeq \cup$$

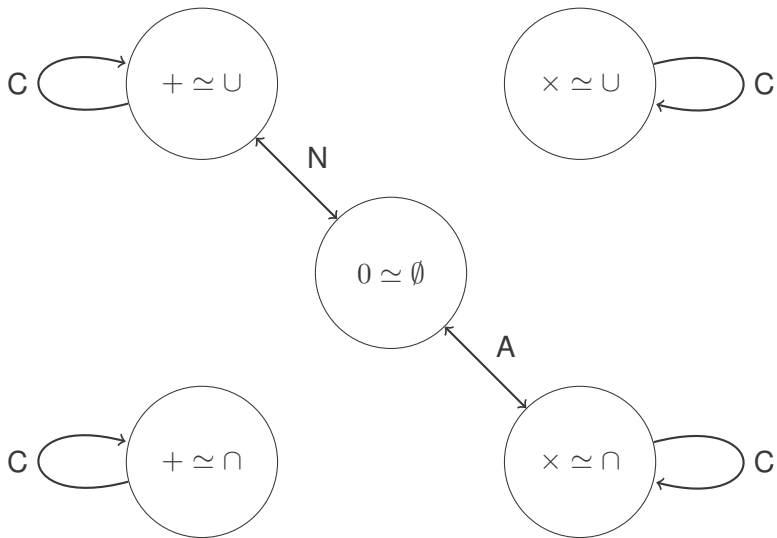
$$\times \simeq \cup$$

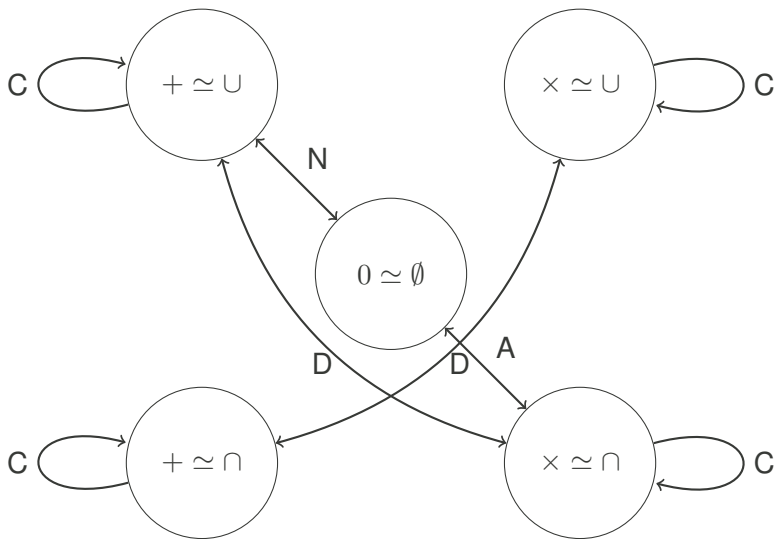
$$0 \simeq \emptyset$$

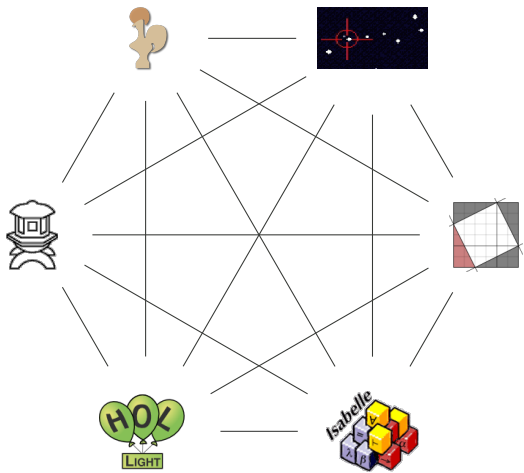
$$+ \simeq \cap$$

$$\times \simeq \cap$$















Applications

- HOLyHammer

Tested library	Proof automation	Success
		30%
	 +  \approx 	40%

- Conjecturing

Lipschitzian \rightarrow Continuous

Theorem

If there is a 2-inaccessible cardinal, then higher-order Tarski-Groethendieck is satisfiable.

Allows us to combine Isabelle/HOL and Isabelle/Mizar libraries

Isomorphism for natural numbers

fun $h2sn :: nat \Rightarrow Set$ ($h2s_{\mathbb{N}}(-)$) **where**

$h2s_{\mathbb{N}}(0::nat) =_S 0_S \mid h2s_{\mathbb{N}}(Suc(x)) =_S succ\ h2s_{\mathbb{N}}(x)$

function $s2hn :: Set \Rightarrow nat$ ($s2h_{\mathbb{N}}(-)$) **where**

$\neg x\ be\ Nat \implies s2h_{\mathbb{N}}(x) =_{\mathcal{H}}\ undefined$

$\mid s2h_{\mathbb{N}}(0_S) =_{\mathcal{H}}\ 0$

$\mid x\ be\ Nat \implies s2h_{\mathbb{N}}(succ(x)) =_{\mathcal{H}}\ Suc(s2h_{\mathbb{N}}(x))$

Preservation of Constants allows theorem translation

theorem *Nat_to_Nat*:

fixes $x::nat$ **and** $y::nat$

assumes n be *Nat* **and** m be *Nat*

shows $\text{h2s}_{\mathbb{N}}(x +_{\mathcal{H}} y) =_{\mathcal{S}} \text{h2s}_{\mathbb{N}}(x) +_{\mathcal{S}^{\mathbb{N}}} \text{h2s}_{\mathbb{N}}(y)$

$$\text{s2h}_{\mathbb{N}}(n +_{\mathcal{S}^{\mathbb{N}}} m) =_{\mathcal{H}} \text{s2h}_{\mathbb{N}}(n) +_{\mathcal{H}} \text{s2h}_{\mathbb{N}}(m)$$

$$\text{h2s}_{\mathbb{N}}(x *_{\mathcal{H}} y) =_{\mathcal{S}} \text{h2s}_{\mathbb{N}}(x) *_{\mathcal{S}^{\mathbb{N}}} \text{h2s}_{\mathbb{N}}(y)$$

$$\text{s2h}_{\mathbb{N}}(n *_{\mathcal{S}^{\mathbb{N}}} m) =_{\mathcal{H}} \text{s2h}_{\mathbb{N}}(n) *_{\mathcal{H}} \text{s2h}_{\mathbb{N}}(m)$$

$$x < y \longleftrightarrow \text{h2s}_{\mathbb{N}}(x) \subset \text{h2s}_{\mathbb{N}}(y)$$

$$n \subset m \longleftrightarrow \text{s2h}_{\mathbb{N}}(n) < \text{s2h}_{\mathbb{N}}(m)$$

$$x \text{ dvd } y \longleftrightarrow \text{h2s}_{\mathbb{N}}(x) \text{ divides } \text{h2s}_{\mathbb{N}}(y)$$

$$n \text{ divides } m \longleftrightarrow \text{s2h}_{\mathbb{N}}(n) \text{ dvd } \text{s2h}_{\mathbb{N}}(m)$$

$$\text{prime}(x) \longleftrightarrow \text{h2s}_{\mathbb{N}}(x) \text{ is prime}_{\mathcal{S}}$$

$$n \text{ is prime}_{\mathcal{S}} \longleftrightarrow \text{prime}(\text{s2h}_{\mathbb{N}}(n))$$

theorem *Bertrand*:

$\forall n::\text{Nat}. 1_{\mathcal{S}} \subset n \longrightarrow$

$(\exists p::\text{Nat}. p \text{ be prime}_{\mathcal{S}} \wedge n \subset p \wedge p \subset (2_{\mathcal{S}} *_{\mathcal{S}^{\mathbb{N}}} n))$

Beyond natural numbers

Lists

theorem *s2hL-Prop*:

assumes *p* be *FinSequence* **and** *q* be *FinSequence*
and *n* be *Nat* **and** *n* in *len p*

shows $size(s2h_L(s2h, p)) =_{\mathcal{H}} s2h_{\mathbb{N}}(len\ p)$
 $s2h_L(s2h, p \hat{\ } q) =_{\mathcal{H}} s2h_L(s2h, p) @ s2h_L(s2h, q)$
 $s2h_L(s2h, p) ! s2h_{\mathbb{N}}(n) =_{\mathcal{H}} s2h(p. (succ\ n))$

Functions

theorem *HtoSappl*:

assumes $belsoS(h2sd, s2hd, d)$ **and** $belsoS(h2sr, s2hr, r)$
shows $h2sf(s2hd, h2sr, d, f).h2sd(x) =_S h2sr(f(x))$

Algebra

Groups

Assuming an isomorphism on the carrier and operation, groups are isomorphic

Rings

mdf *int_3_def_3* (\mathbb{Z} -ring) **where**

func \mathbb{Z} -ring \rightarrow strict(doubleLoopStr) equals [#

carrier \mapsto INT;

addF \mapsto addint;

ZeroF \mapsto 0_S;

multF \mapsto multint;

OneF \mapsto 1_S]

theorem *H_Zring_to_S_Zring*:

$\text{h2s}_R(\text{s2h}_{\mathbb{Z}}, \text{h2s}_{\mathbb{Z}}, \text{INT}, \mathcal{Z}) =_S \mathbb{Z}\text{-ring}$

$\text{s2h}_R(\text{s2h}_{\mathbb{Z}}, \text{h2s}_{\mathbb{Z}}, \mathbb{Z}\text{-ring}) =_{\mathcal{H}} \mathcal{Z}$

Questions: What do we want from Dedukti?

- Proof Analysis / Presentation / Search
- Proof Translation and Auditing
- Proof Advice (supervised and unsupervised learning)
- Re-use and Combining
- ...

What techniques for this do we have / want?

- Alignments
- Proof Engineering
- Reverse Mathematics
- ...