Recent Developments in Theory Combination: Datatypes, Synergies, and Unicorns

Yoni Zohar
Bar-Ilan University

Based on joint work with:

Ying Sheng, Jane Lange, Guilherme Toledo, Andrew Reynolds, Christophe Ringeissen, Pascal Fontaine, Clark Barrett, Cesare Tinelli

FroCoS 2023

Recent Developments in Theory Combination: Datatypes, Synergies, and Unicorns



Recent Developments in **Theory Combination**: Datatypes, Synergies, and Unicorns



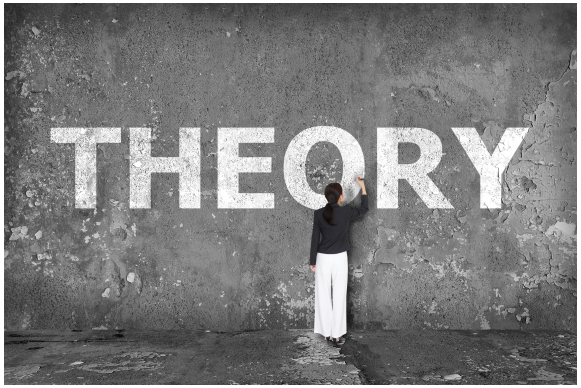
Recent Developments in **Theory** Combination: Datatypes, Synergies, and Unicorns



Theories

Disclaimer

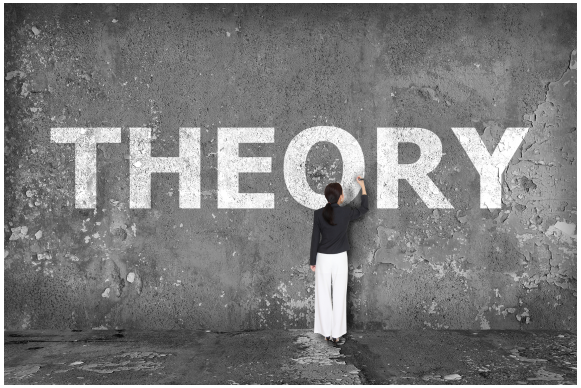
- Many-sorted first-order logic
  - Sorts
  - Sorted variables and constnts
  - Sorted / typed function and predicate symbols
  - Connectives and quantifiers
- A theory is a set of axioms
- A theory is a class of structures
- Dual notions



Theories

Disclaimer

- Many-sorted first-order logic
  - Sorts
  - Sorted variables and constnts
  - Sorted / typed function and predicate symbols
  - Connectives and quantifiers
- A theory is a set of axioms
- A theory is a class of structures
- Dual notions



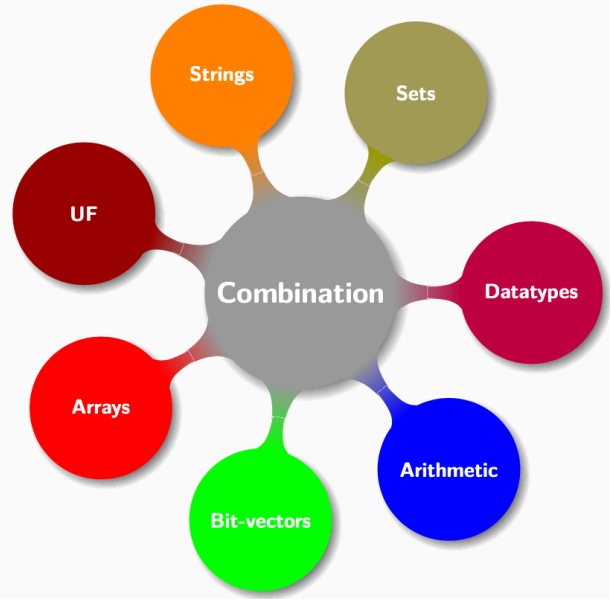
Recent Developments in Theory **Combination**: Datatypes, Synergies, and Unicorns



Theory Combination

What is This About?

- Given specialized theory solvers
- How to split the problem?
- How to combine the results?

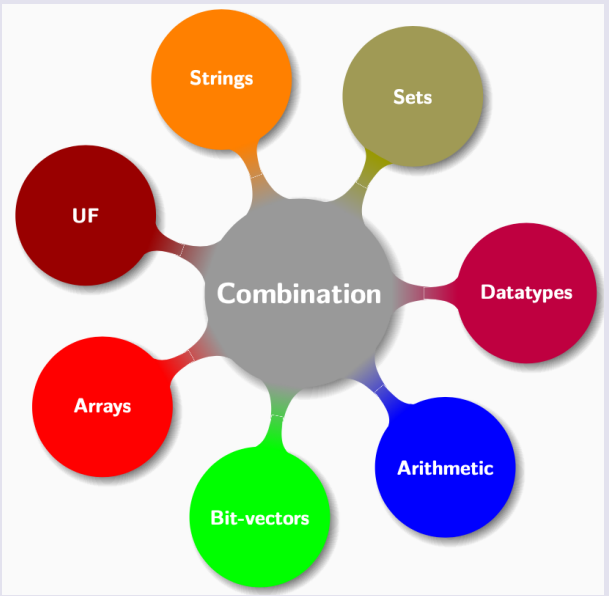



```
(set-logic UFDTNIA)
(declare-sort T@$TypeName 0)
(declare-sort |T@[Int]$TypeValue| 0)
(declare-datatypes ((T@$TypeValue 0)(T@$TypeValueArray 0)) (((($BooleanType ) ($IntegerType ) ($Ad
(declare-sort |T@[Int]$Value| 0)
(declare-datatypes ((T@$Value 0)(T@$ValueArray 0)) ((($Boolean (|b#$Boolean| Bool) ) ($Integer (|
(declare-sort |T@[$TypeValueArray,Int]Bool| 0)
(declare-sort |T@[$TypeValueArray,Int]$Value| 0)
(declare-datatypes ((T@$Memory 0)) ((($Memory (|domain#$Memory| |T@[$TypeValueArray,Int]Bool|) (|
(declare-datatypes ((T@$Location 0)) ((($Global (|ts#$Global| T@$TypeValueArray) (|a#$Global| Int
(declare-sort |T@[Int]Int| 0)
(declare-datatypes ((T@$Path 0)) ((($Path (|p#$Path| |T@[Int]Int|) (|size#$Path| Int) ) ) ))
```

Example: Lists with A Single Element

$a_0, a_1, a_2 \neq nil$ $a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$ $tail(a_0) = tail(a_1) = tail(a_2) = nil$	a_0 a_1 a_2
$0 \leq head(a_0), head(a_1), head(a_2) \leq 1$	$0 \mid 1$ a_0 $0 \mid 1$ a_1 $0 \mid 1$ a_2

1. This formula is UNSAT
2. How can we find this out...
3. Without developing a new solver

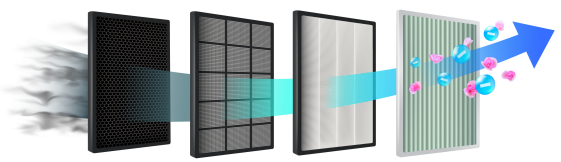


First Step: Purification

DT Solver	Arith Solver
$a_0, a_1, a_2 \neq nil$ $a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$ $tail(a_0) = tail(a_1) = tail(a_2) = nil$	

But where does the fourth formula go?

$$0 \leq head(a_0), head(a_1), head(a_2) \leq 1$$



First Step: Purification

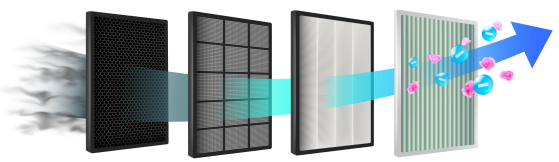
DT Solver	Arith Solver
$a_0, a_1, a_2 \neq nil$ $a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$ $tail(a_0) = tail(a_1) = tail(a_2) = nil$	

But where does the fourth formula go?

$$0 \leq head(a_0), head(a_1), head(a_2) \leq 1$$

Replace with the following two formulas:

1. $x = head(a_0), y = head(a_1), z = head(a_2)$
2. $0 \leq x, y, z \leq 1$



First Step: Purification

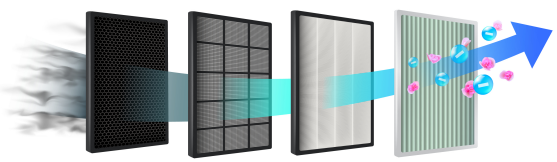
DT Solver	Arith Solver
$a_0, a_1, a_2 \neq nil$ $a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$ $tail(a_0) = tail(a_1) = tail(a_2) = nil$ $x = head(a_0), y = head(a_1), z = head(a_2)$	$0 \leq x, y, z \leq 1$

But where does the fourth formula go?

$$0 \leq head(a_0), head(a_1), head(a_2) \leq 1$$

Replace with the following two formulas:

1. $x = head(a_0), y = head(a_1), z = head(a_2)$
2. $0 \leq x, y, z \leq 1$



Second Step: Combination

Naive Method

$A \wedge B$ is $(T_1 \oplus T_2)$-SAT

$\Leftrightarrow$

A is T_1-SAT and B is T_2-SAT.

DT Solver	Arith Solver
$a_0, a_1, a_2 \neq nil$ $a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$ $tail(a_0) = tail(a_1) = tail(a_2) = nil$ $x = head(a_0), y = head(a_1), z = head(a_2)$	$0 \leq x, y, z \leq 1$
SAT	SAT

The formlua is UNSAT

Method says SAT



Second Step: Combination

Nelson-Oppen Method [Nelson& Oppen 1979]

$A \wedge B$ is $(T_1 \oplus T_2)$-SAT

$\Leftrightarrow$

$\exists \delta$ over $FV(A) \cap FV(B)$ s.t. $A \wedge \delta$ is T_1-SAT and $B \wedge \delta$ is T_2-SAT.

DT Solver	Arith Solver
$a_0, a_1, a_2 \neq nil$ $a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$ $tail(a_0) = tail(a_1) = tail(a_2) = nil$ $x = head(a_0), y = head(a_1), z = head(a_2)$ $\delta : x = y = z$	$0 \leq x, y, z \leq 1$ $\delta : x = y = z$
UNSAT	SAT

Second Step: Combination

Nelson-Oppen Method [Nelson& Oppen 1979]

$A \wedge B$ is $(T_1 \oplus T_2)$-SAT

$\Leftrightarrow$

$\exists \delta$ over $FV(A) \cap FV(B)$ s.t. $A \wedge \delta$ is T_1-SAT and $B \wedge \delta$ is T_2-SAT.

DT Solver	Arith Solver
$a_0, a_1, a_2 \neq nil$	$0 \leq x, y, z \leq 1$
$a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$	
$tail(a_0) = tail(a_1) = tail(a_2) = nil$	
$x = head(a_0), y = head(a_1), z = head(a_2)$	
$\delta : x = y \neq z$	$\delta : x = y \neq z$
UNSAT	SAT

Second Step: Combination

Nelson-Oppen Method [Nelson& Oppen 1979]

$A \wedge B$ is $(T_1 \oplus T_2)$-SAT

$\Leftrightarrow$

$\exists \delta$ over $FV(A) \cap FV(B)$ s.t. $A \wedge \delta$ is T_1-SAT and $B \wedge \delta$ is T_2-SAT.

DT Solver	Arith Solver
$a_0, a_1, a_2 \neq nil$	$0 \leq x, y, z \leq 1$
$a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$	
$tail(a_0) = tail(a_1) = tail(a_2) = nil$	
$x = head(a_0), y = head(a_1), z = head(a_2)$	
$\delta : x \neq y = z$	$\delta : x \neq y = z$
UNSAT	SAT

Second Step: Combination

Nelson-Oppen Method [Nelson& Oppen 1979]

$$A \wedge B \text{ is } (T_1 \oplus T_2)\text{-SAT}$$
$$\Leftrightarrow$$
$$\exists \delta \text{ over } FV(A) \cap FV(B) \text{ s.t. } \quad A \wedge \delta \text{ is } T_1\text{-SAT and } B \wedge \delta \text{ is } T_2\text{-SAT.}$$

DT Solver	Arith Solver
$a_0, a_1, a_2 \neq nil$	$0 \leq x, y, z \leq 1$
$a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$	
$tail(a_0) = tail(a_1) = tail(a_2) = nil$	
$x = head(a_0), y = head(a_1), z = head(a_2)$	
$\delta : x \neq y \neq z$	$\delta : x \neq y \neq z$
SAT	UNSAT

The formlua is UNSAT Method says UNSAT



Second Step: Combination

Nelson-Oppen Method [Nelson& Oppen 1979]

$$A \wedge B \text{ is } (T_1 \oplus T_2)\text{-SAT}$$
$$\Leftrightarrow$$
$$\exists \delta \text{ over } FV(A) \cap FV(B) \text{ s.t. } \quad A \wedge \delta \text{ is } T_1\text{-SAT and } B \wedge \delta \text{ is } T_2\text{-SAT.}$$

- Nelson-Oppen works for lists of integers
- What about lists of bit-vectors?

Second Step: Combination

Nelson-Oppen Method [Nelson& Oppen 1979]

$$A \wedge B \text{ is } (T_1 \oplus T_2)\text{-SAT}$$
$$\Leftrightarrow$$
$$\exists \delta \text{ over } FV(A) \cap FV(B) \text{ s.t. } \quad A \wedge \delta \text{ is } T_1\text{-SAT and } B \wedge \delta \text{ is } T_2\text{-SAT.}$$

DT Solver	BV[1] Solver
$a_0, a_1, a_2 \neq nil$ $a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$ $tail(a_0) = tail(a_1) = tail(a_2) = nil$ No shared variables Trivial δ	$TRUE$ No shared variables Trivial δ
SAT	SAT

The formlua is UNSAT



Method says SAT

Second Step: Combination

Polite Combination [Ranise et al. 2005]

$$A \wedge B \text{ is } (T_1 \oplus T_2)\text{-SAT}$$
$$\Leftrightarrow$$
$$\exists \delta \text{ over } \boxed{FV(B)} \text{ of shared sorts s.t.}$$
$$A \wedge \delta \text{ is } T_1\text{-SAT and } \boxed{\text{wit(B)}} \wedge \delta \text{ is } T_2\text{-SAT.}$$

DT Solver	BV[1] Solver
$wit(a_0, a_1, a_2 \neq nil)$	$TRUE$
$wit(a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2)$	
$wit(tail(a_0) = tail(a_1) = tail(a_2) = nil)$	
$\delta =?$	$\delta =?$

Second Step: Combination

Polite Combination [Ranise et al. 2005]

$A \wedge B$ is $(T_1 \oplus T_2)$-SAT

$\Leftrightarrow$

$\exists \delta$ over $\boxed{FV(B)}$ of shared sorts s.t.

$A \wedge \delta$ is T_1-SAT and $\boxed{\text{wit(B)}} \wedge \delta$ is T_2-SAT.

DT Solver	BV[1] Solver
$a_0, a_1, a_2 \neq nil$ $a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$ $\bigwedge_{i=0}^{2} a_i = cons(v_i, nil)$ $\delta =?$	$TRUE$ $\delta =?$

Second Step: Combination

Polite Combination [Ranise et al. 2005]

$$A \wedge B \text{ is } (T_1 \oplus T_2)\text{-SAT}$$

$$\Leftrightarrow$$

$$\exists \delta \text{ over } \boxed{FV(B)} \text{ of shared sorts s.t.}$$

$$A \wedge \delta \text{ is } T_1\text{-SAT and } \boxed{\text{wit(B)}} \wedge \delta \text{ is } T_2\text{-SAT.}$$

DT Solver	BV[1] Solver
$a_0, a_1, a_2 \neq nil$ $a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$ $\bigwedge_{i=0}^{2} a_i = cons(v_i, nil)$ $\delta : v_0 = v_1 = v_2$	$TRUE$ $\delta : v_0 = v_1 = v_2$
UNSAT	SAT

Second Step: Combination

Polite Combination [Ranise et al. 2005]

$$A \wedge B \text{ is } (T_1 \oplus T_2)\text{-SAT}$$
$$\Leftrightarrow$$
$$\exists \delta \text{ over } \boxed{FV(B)} \text{ of shared sorts s.t.}$$
$$A \wedge \delta \text{ is } T_1\text{-SAT and } \boxed{\text{wit(B)}} \wedge \delta \text{ is } T_2\text{-SAT.}$$

DT Solver	BV[1] Solver
$a_0, a_1, a_2 \neq nil$ $a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$ $\bigwedge_{i=0}^{2} a_i = cons(v_i, nil)$ $\delta : v_0 = v_1 \neq v_2$	$TRUE$ $\delta : v_0 = v_1 \neq v_2$
UNSAT	SAT

Second Step: Combination

Polite Combination [Ranise et al. 2005]

$$A \wedge B \text{ is } (T_1 \oplus T_2)\text{-SAT}$$

$$\Leftrightarrow$$

$$\exists \delta \text{ over } \boxed{FV(B)} \text{ of shared sorts s.t.}$$

$$A \wedge \delta \text{ is } T_1\text{-SAT and } \boxed{\text{wit(B)}} \wedge \delta \text{ is } T_2\text{-SAT.}$$

DT Solver	BV[1] Solver
$a_0, a_1, a_2 \neq nil$ $a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$ $\bigwedge_{i=0}^{2} a_i = cons(v_i, nil)$ $\delta : v_0 \neq v_1 = v_2$	$TRUE$ $\delta : v_0 \neq v_1 = v_2$
UNSAT	SAT

Second Step: Combination

Polite Combination [Ranise et al. 2005]

$A \wedge B$ is $(T_1 \oplus T_2)$-SAT

$\Leftrightarrow$

$\exists \delta$ over $\boxed{FV(B)}$ of shared sorts s.t.

$A \wedge \delta$ is T_1-SAT and $\boxed{\text{wit(B)}} \wedge \delta$ is T_2-SAT.

DT Solver	BV[1] Solver
$a_0, a_1, a_2 \neq nil$ $a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$ $\bigwedge_{i=0}^{2} a_i = cons(v_i, nil)$ $\delta : v_0 \neq v_1 \neq v_2$	$TRUE$ $\delta : v_0 \neq v_1 \neq v_2$
SAT	UNSAT

The formlua is UNSAT Method says UNSAT



Combination Methods – Summary

$A \wedge B$ is $(T_1 \oplus T_2)$-SAT

$\Leftrightarrow$

Naive Combination

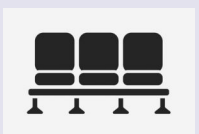
A is T_1-SAT and B is T_2-SAT



Nelson-Oppen Combination

$\exists\ \delta$ over $FV(A) \cap FV(B)$ s.t.

$A \wedge \delta$ is T_1-SAT and $B \wedge \delta$ is T_2-SAT



Polite Combination

$\exists\ \delta$ over $FV(B)$ s.t.

$A \wedge \delta$ is T_1-SAT

and $wit(B) \wedge \delta$ is T_2-SAT



Combination Methods – Summary

$A \wedge B$ is $(T_1 \oplus T_2)$-SAT

$\Leftrightarrow$

Naive Combination

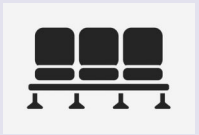
A is T_1-SAT and B is T_2-SAT



Nelson-Oppen Combination

$\exists\ \delta$ over $FV(A) \cap FV(B)$ s.t.
$A \wedge \delta$ is T_1-SAT and $B \wedge \delta$ is T_2-SAT

T_1 and T_2 are stably infinite



Polite Combination

$\exists\ \delta$ over $FV(B)$ s.t.
$A \wedge \delta$ is T_1-SAT
and $wit(B) \wedge \delta$ is T_2-SAT

T_2 is strongly polite



Cardinalities of Models

Stable Infiniteness

T is stably infinite w.r.t. S if every T-SAT formula is T-SAT by a structure in which all domains of S are infinite.



Smoothness

T is smooth w.r.t. S if for every:

- T-SAT formula A and a T-model of it $\mathcal{I}$
- κ from S to carinalities with $\kappa(\sigma) \geq |\sigma^{\mathcal{I}}|$

There is a T-model $\mathcal{J}$ of A with $|\sigma^{\mathcal{J}}| = \kappa(\sigma)$



Example

- Lists are smooth w.r.t. element sort
- BV is not smooth w.r.t. $BV[4]$



Cardinalities of Models

Finite Witnessibility

T is finitely witnessable w.r.t. S if there exists a function *wit* such that:

- A is T-equivalent to $\exists \overline{w}.wit(A)$, $w = FV(wit(A)) \setminus FV(A)$;
- If $wit(A)$ is T-SAT then it is T-SAT by a model $\mathcal{I}$ with $\sigma^{\mathcal{I}} = FV_{\sigma}(wit(A))^{\mathcal{I}}$ for every $\sigma \in S$.



Cardinalities of Models

Finite Witnessibility

T is finitely witnessable w.r.t. S if there exists a function *wit* such that:

- A is T-equivalent to $\exists \overline{w}.wit(A)$, $w = FV(wit(A)) \setminus FV(A)$;
- If $wit(A)$ is T-SAT then it is T-SAT by a model $\mathcal{I}$ with $\sigma^{\mathcal{I}} = FV_{\sigma}(wit(A))^{\mathcal{I}}$ for every $\sigma \in S$.



Formula	Model
$A: \ head(a_0) \neq head(a_1)$	$elem^{\mathcal{I}} = \emptyset$ – ☹
$wit(A): \ a_0 = cons(e_0, a_0') \wedge$ $a_1 = cons(e_0, a_1') \wedge$ $e_0 \neq e_1$	$elem^{\mathcal{I}} = \{e_0^{\mathcal{I}}, e_1^{\mathcal{I}}\}$ – ☺

Politeness

Definition

A theory is polite if it is smooth + finitely witnessble.

Theorem? [Ranise et al. 2005]

Polite combination is correct for polite theories.

Theorem! [Jovanovic & Barrett 2010]

Polite combination is correct for strongly polite theories.

Combining Data Structures with Nonstably Infinite Theories Using Many-Sorted Logic*

Silvio Ranise[1], Christophe Ringeissen[1], and Calogero G. Zarba[2]

[1] LORIA and INRIA-Lorraine
[2] University of New Mexico

Polite Theories Revisited*

Dejan Jovanović and Clark Barrett

New York University
{dejan,barrett}@cs.nyu.edu

Strong Finite Witnessibility

Strong Finite Witnessibility

T is strongly finitely witnessable w.r.t. a set S of sorts if there exists a function *wit* such that:

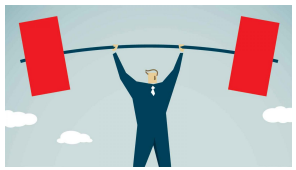
- A is T-equivalent to $\exists \overline{w}\, wit(A)$, $w = FV(wit(A)) \setminus FV(A)$;
- If $wit(A) \wedge \delta$ is T-SAT then it is T-sat by a model with $\sigma^{\mathcal{I}} = FV_{\sigma}(wit(A) \wedge \delta)^{\mathcal{I}}$ for every $\sigma \in S$, for all arrangements δ over S.

Definition

T is strongly polite if it is smooth and strongly finitely witnessable

Question

Politeness = Strong Politeness ?



Recent Developments in Theory Combination: Datatypes, Synergies, and Unicorns



Recent Developments in Theory Combination: **Datatypes**, Synergies, and Unicorns



Datatypes

Datatypes

- Models DTs such as tuples, lists and trees
- Term algebras

Example: Lists

- Constructors: $\mathit{cons}(e, \ell)$, nil
- Selectors: $\mathit{head}(x)$, $\mathit{tail}(x)$
- Testers: $\mathit{is_nil}(x)$, $\mathit{is_cons}(x)$



Axioms

$$\forall x_1, \ldots, x_n.\ is_C(C(x_1, \ldots, x_n)) \approx \mathsf{true}$$

$$\forall x_1, \ldots, x_n.\ is_{C'}(C(x_1, \ldots, x_n)) \approx \mathsf{false}$$

$$\forall x_1, \ldots, x_n.\ S_C^{(i)}(C(x_1, \ldots, x_n)) \approx x_i \qquad \text{for all } i = 1, \ldots, n$$

$$\forall x_1, \ldots, x_n.\ S_{C'}^{(i)}(C(x_1, \ldots, x_n)) \approx t_{C'}^i \qquad \text{for all } i = 1, \ldots, n'$$

Correctness in cvc5

- Correctness of algorithm depends on strong politeness
- This was never proved
- No combination bug was found
- But still – A proof of correctness is desirable



What it Boils Down To

1. Having Enough Variables

- $tail(x) \neq tail(y)$ is SAT
- No model satisfies $elem^{\mathcal{I}} = FV_{elem}(tail(x) \neq tail(y)) = \emptyset$
- Solution: witness function

2. Minimizing Models

- Given a model, create a minimal one
- New model should respect all equalities and disequalities
- Solution: finite witnessability

Generalizing

Solution should hold for all DTs



Additive Witnesses

Politeness vs. Strong Politeness

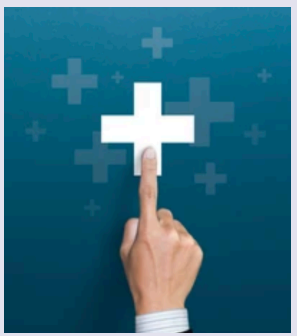
- Proving strong politeness is harder
- Idea: a sufficient condition in which polite $\rightarrow$ strongly polite
- Condition: additivity

Additiity

A function f is additive if

$$f(f(\phi) \wedge \delta) \equiv f(\phi) \wedge \delta$$

. . . under reasonable assumptions on δ. . .



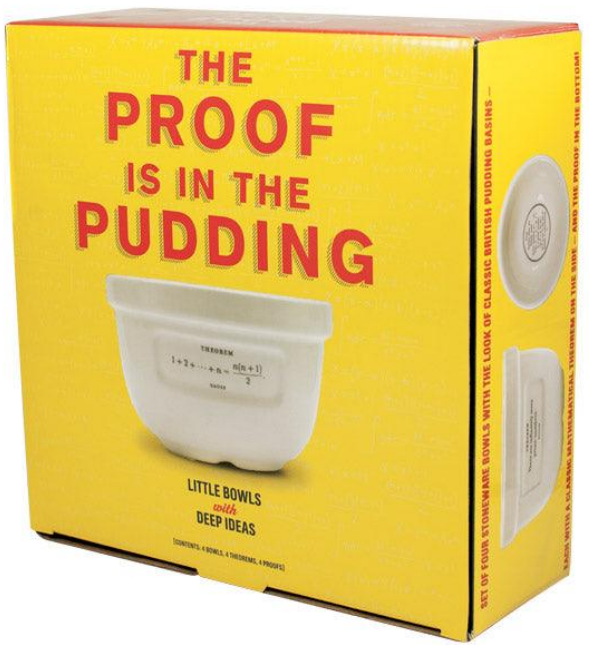
Proposition

If a witness is additive, then it is a strong witness.

Proof ideas

Prove politeness of datatype theory:

- Smoothness
  - Relatively routine
- Additive witness function
- Correctness: model construction for finite witnessability



Witness function

Definition

Extend from the original formula as follows:

- For $y = s_{c,i}(x)$:
  add $x = c(..., y, ...) \vee (\bigvee_{d \neq c} x = d(...))$
- For $is_c(x)$: add $x = c(...)$
- For $\neg is_c(x)$: add $\bigvee_{d \neq c} x = d(...)$

Additivity

- This witness function is indeed finitely witnessable.
- But strongly?



Witness function

Definition

Extend from the original formula as follows:

- For $y = s_{c,i}(x)$: if no constructor already specified to x, add $x = c(..., y, ...) \vee (\bigvee_{d \neq c} x = d(...))$
- For $is_c(x)$: add $x = c(...)$ if it does not exist
- For $\neg is_c(x)$: add $\bigvee_{d \neq c} x = d(...)$ if no such literal with $d \neq c$ exists

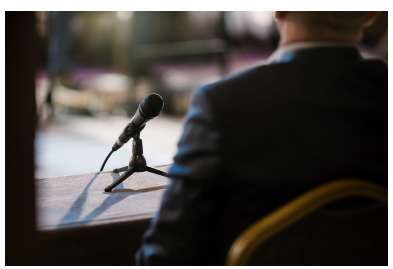
Additivity

- This witness function is indeed finitely witnessable.
- But strongly?
- Turn it into an additive function.



Witness function – Example

- $y = s_{c,i}(x) \mapsto x = c(..., y, ...) \vee (\bigvee_{d \neq c} x = d(...))$
- $is_c(x) \mapsto x = c(...)$
- $\neg is_c(x) \mapsto \bigvee_{d \neq c} x = d(...)$



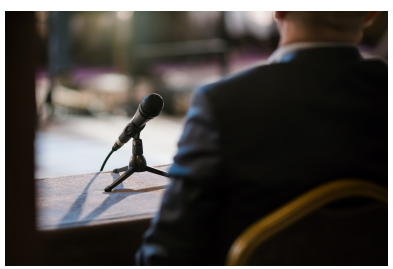
Witness function – Example

- $y = s_{c,i}(x) \mapsto x = c(..., y, ...) \vee (\bigvee_{d \neq c} x = d(...))$
- $is_c(x) \mapsto x = c(...)$
- $\neg is_c(x) \mapsto \bigvee_{d \neq c} x = d(...)$

Example

$y = tail(x)$

Add: $x = cons(e, y) \vee x = nil$



Correctness

Additivity

Almost by construction.

Witnessability Requirement

- Equivalence
- "small model"

Key Point

- $x \neq y$ maybe thanks to lost elements
- Instead, "pump" surviving elements"
- $[1, 2, 3] \neq [1, 2, 4] \mapsto [1, 2, 1, 1] \neq [1, 2, 1, 1, 1]$
- Number of pumps can be inferred from:
  - The formula
  - The original model



Datatypes – Summary

- Useful theory in SMT
- Well-understood decision procedures
- Correctness relies on strong politeness, which was not proved
- (No bug was found)

- We proved that it is indeed strongly polite
- Proof "eliminates" selectors and testers
- Two key ideas:
  - Pumping elements
  - Using the given model



Recent Developments in Theory Combination: Datatypes, Synergies, and Unicorns



Recent Developments in Theory Combination: Datatypes, Synergies, and **Unicorns**



Combination of Properties

Earlier we left open the question of:

Politeness vs. Strong Politeness



Separation

Theorem

- *Politeness* $\neq$ *Strong Politeness*
- *In mono-sorted empty signatures:*
  - *Politeness* $=$ *Strong Politeness*
  - *Finite Witnessability* $\neq$ *Strong Finite Witnessability*

Separation

Theorem

- *Politeness* $\neq$ *Strong Politeness*
- *In mono-sorted empty signatures:*
  - *Politeness* $=$ *Strong Politeness*
  - *Finite Witnessability* $\neq$ *Strong Finite Witnessability*

The Theory $T_{2,3}$

- Two sorts σ_1, σ_2, no symbols except $=$
- $\mathcal{I}$ is in $T_{2,3}$ iff at least one of the following holds:
  - $|\sigma_1^{\mathcal{I}}| = 2$ and $|\sigma_2^{\mathcal{I}}| = \infty$
  - $|\sigma_1^{\mathcal{I}}|, |\sigma_2^{\mathcal{I}}| \geq 3$
- $T_{2,3}$ is polite.
  - Smooth
  - $wit(A) = A \wedge \bigwedge_{i=1}^{3} x_i = x_i \wedge \bigwedge_{i=1}^{3} y_i = y_i$
- Every witness is not strong

Separation

Where Did We Find $T_{2,3}$?

Many-Sorted Equivalence of Shiny and Strongly Polite Theories

Filipe Casal[1,2] · **João Rasga**[1,2]

As an example of a theory stably finite according to [14] (but not stably finite according to the notion proposed in Definition 6), with minimal models with infinite cardinalities, consider a two-sorted theory that accepts all models $\mathcal{A}$ with cardinalities such that

$$\text{either } |A_1| \geq 2 \text{ and } |A_2| = \infty \text{ or } |A_1| \geq 3 \text{ and } |A_2| \geq 3 \ .$$

Alternative Route

- Casal & Rasga studied strong politeness and shininess
- Another proof can be obtained by [Casal & Rasga] + [Ranise et al.]
- Such a proof would go through shiny theories
- A direct proof is preferable

Mono Sorted Empty Signatures

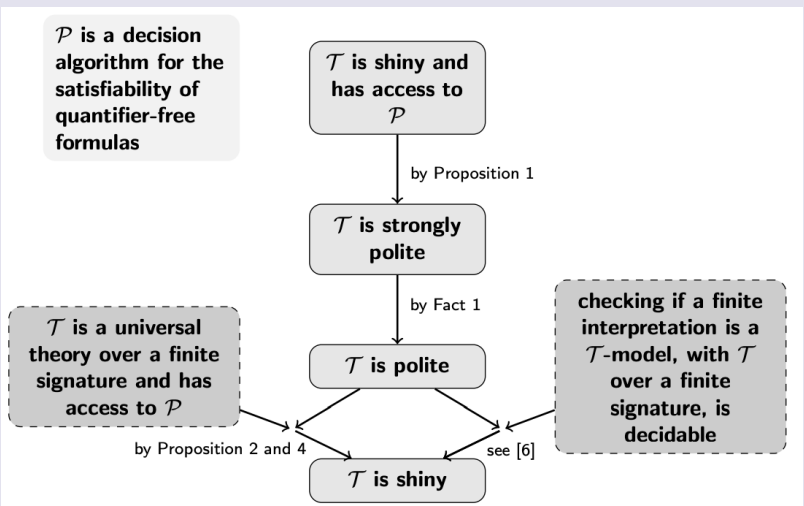
Theorem

For empty mono-sorted signatures: politeness = strong politeness.

Proof

- smoothness + finite witnessability → upward closure
- empty signatures can only describe sets of arrangements

Alternative Route [Casal&Rasga 2013,Ranise et al. 2005]



Revisiting the Equivalence of Shininess and Politeness

Filipe Casal[1] and João Rasga[2]

Finite Witnessability

Theorem

For empty mono-sorted signatures:
finite witnessability $\neq$ *strong finite witnessability*

The theory T_{Even}

all structures with even or infinite number of elements.

Lemma

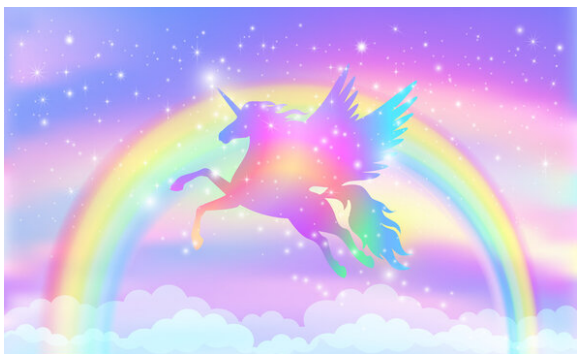
- T_{Even} is finitely witnesssable
- There is no strong witness
- Notice: T_{Even} is not smooth

Alternative Route?

- No alternative route through shiny theories
- Shiny theories are smooth

A Year Ago

OK nice, but that's it? . . .



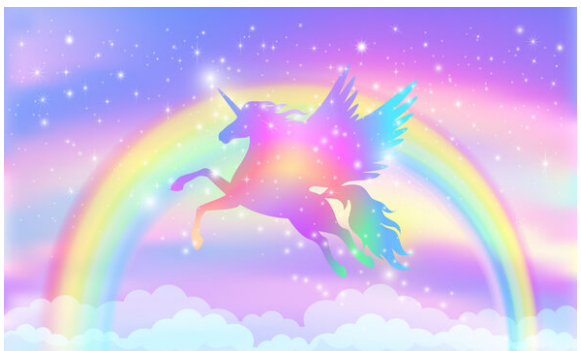
OK nice, but...

Separation

- Politeness != Strong Politeness
- Finite Witnessability != Strong Finite Witnessability
- On empty one-sorted signatures: Politeness = Strong Politeness
- But we can do so much more...

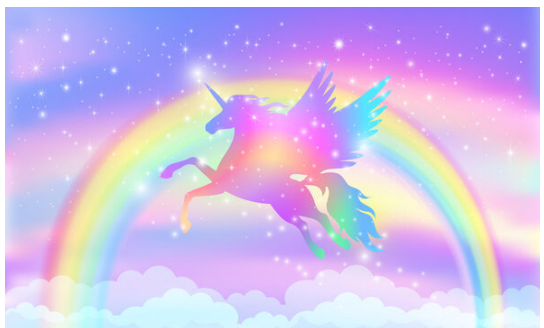
Guilherme's Idea

- Let's really understand what's going on
- Is there a ____ theory that is not _____
- Given a set of properties: is there a theory with these properties?



An Initial Table

Stably-infinite	Smooth	S.F. witnessable	Example
T	T	T	$T_{\geq n}$
		F	T_{∞}
	F	T	
		F	$T_{n,\infty}$
F	F	T	$T_{\leq n}$
		F	$T_{m,n}$





Full Table

					Empty		Non-empty		
SI	**SM**	**FW**	**SW**	**CV**	One-sorted	Many-sorted	One-sorted	Many-sorted	N^o
T	T	T	T	T	$\mathcal{T}_{\geq n}$	$(\mathcal{T}_{\geq n})^2$	$(\mathcal{T}_{\geq n})_s$	$((\mathcal{T}_{\geq n})^2)_s$	1
				F	Theorem 8		$(\mathcal{T}_{\geq n})_\vee$	$((\mathcal{T}_{\geq n})^2)_\vee$	2
			F	T	Theorem 4	$\mathcal{T}_{2,3}$	$\mathcal{T}_f$	$(\mathcal{T}_f)_s$	3
				F		Theorem 8	$\mathcal{T}_f^s$	$(\mathcal{T}_{2,3})_\vee$	4
		F	T	T	Theorem 3				5
				F					6
			F	T	$\mathcal{T}_\infty$	$(\mathcal{T}_\infty)^2$	$(\mathcal{T}_\infty)_s$	$((\mathcal{T}_\infty)^2)_s$	7
				F	Theorem 8		$(\mathcal{T}_\infty)_\vee$	$((\mathcal{T}_\infty)^2)_\vee$	8
	F	T	T	T	Theorem 10	Unicorn Theories	Theorem 10	Unicorn Theories	9
				F		Theorem 8			10
			F	T	$\mathcal{T}_{even}^\infty$	$(\mathcal{T}_{even}^\infty)^2$	$(\mathcal{T}_{even}^\infty)_s$	$((\mathcal{T}_{even}^\infty)^2)_s$	11
				F	Theorem 8		$(\mathcal{T}_{even}^\infty)_\vee$	$((\mathcal{T}_{even}^\infty)^2)_\vee$	12
		F	T	T	Theorem 3				13
				F					14
			F	T	$\mathcal{T}_{n,\infty}$	$(\mathcal{T}_{n,\infty})^2$	$(\mathcal{T}_{n,\infty})_s$	$((\mathcal{T}_{n,\infty})^2)_s$	15
				F	Theorem 8		$(\mathcal{T}_{n,\infty})_\vee$	$((\mathcal{T}_{n,\infty})^2)_\vee$	16
F	T	T	T	T	Theorem 2				17
				F					18
			F	T					19
				F					20
		F	T	T	Theorems 2 and 3				21
				F					22
			F	T	Theorem 2				23
				F					24
	F	T	T	T	$\mathcal{T}_{\leq 1}$	$(\mathcal{T}_{\leq 1})^2$	$(\mathcal{T}_{\leq 1})_s$	$((\mathcal{T}_{\leq 1})^2)_s$	25
				F	$\mathcal{T}_{\leq n}$	$(\mathcal{T}_{\leq n})^2$	$(\mathcal{T}_{\leq n})_s$	$((\mathcal{T}_{\leq n})^2)_s$	26
			F	T	Theorem 11	$\mathcal{T}_1^{odd}$	$\mathcal{T}_{odd}^{\neq}$	$(\mathcal{T}_1^{odd})_s$	27
				F	$\mathcal{T}_{m,n}$	$(\mathcal{T}_{m,n})^2$	$(\mathcal{T}_{m,n})_s$	$((\mathcal{T}_{m,n})^2)_s$	28
		F	T	T	Theorem 3				29
				F					30
			F	T	Theorem 9	$\mathcal{T}_1^\infty$	$\mathcal{T}_{1,\infty}^{\neq}$	$(\mathcal{T}_1^\infty)_s$	31
				F		$\mathcal{T}_2^\infty$	$\mathcal{T}_{2,\infty}^{\neq}$	$(\mathcal{T}_2^\infty)_s$	32

Full Table

					Empty		Non-empty		
SI	**SM**	**FW**	**SW**	**CV**	One-sorted	Many-sorted	One-sorted	Many-sorted	N^o
T	T	T	T	T	$\mathcal{T}_{\geq n}$	$(\mathcal{T}_{\geq n})^2$	$(\mathcal{T}_{\geq n})_s$	$((\mathcal{T}_{\geq n})^2)_s$	1
				F	Theorem 8		$(\mathcal{T}_{\geq n})_\vee$	$((\mathcal{T}_{\geq n})^2)_\vee$	2
			F	T	Theorem 4	$\mathcal{T}_{2,3}$	$\mathcal{T}_f$	$(\mathcal{T}_f)_s$	3
				F		Theorem 8	$\mathcal{T}_f^s$	$(\mathcal{T}_{2,3})_\vee$	4
		F	T	T	Theorem 3				5
				F					6
			F	T	$\mathcal{T}_\infty$	$(\mathcal{T}_\infty)^2$	$(\mathcal{T}_\infty)_s$	$((\mathcal{T}_\infty)^2)_s$	7
				F	Theorem 8		$(\mathcal{T}_\infty)_\vee$	$((\mathcal{T}_\infty)^2)_\vee$	8
	F	T	T	T	Theorem 10	Unicorn Theories	Theorem 10	Unicorn Theories	9
				F		Theorem 8			10
			F	T	$\mathcal{T}_{even}^\infty$	$(\mathcal{T}_{even}^\infty)^2$	$(\mathcal{T}_{even}^\infty)_s$	$((\mathcal{T}_{even}^\infty)^2)_s$	11
				F	Theorem 8		$(\mathcal{T}_{even}^\infty)_\vee$	$((\mathcal{T}_{even}^\infty)^2)_\vee$	12
		F	T	T	Theorem 3				13
				F					14
			F	T	$\mathcal{T}_{n,\infty}$	$(\mathcal{T}_{n,\infty})^2$	$(\mathcal{T}_{n,\infty})_s$	$((\mathcal{T}_{n,\infty})^2)_s$	15
				F	Theorem 8		$(\mathcal{T}_{n,\infty})_\vee$	$((\mathcal{T}_{n,\infty})^2)_\vee$	16
F	T	T	T	T	Theorem 2				17
				F					18
			F	T					19
				F					20
		F	T	T	Theorems 2 and 3				21
				F					22
			F	T	Theorem 2				23
				F					24
	F	T	T	T	$\mathcal{T}_{\leq 1}$	$(\mathcal{T}_{\leq 1})^2$	$(\mathcal{T}_{\leq 1})_s$	$((\mathcal{T}_{\leq 1})^2)_s$	25
				F	$\mathcal{T}_{\leq n}$	$(\mathcal{T}_{\leq n})^2$	$(\mathcal{T}_{\leq n})_s$	$((\mathcal{T}_{\leq n})^2)_s$	26
			F	T	Theorem 11	$\mathcal{T}_1^{odd}$	$\mathcal{T}_{odd}^{\neq}$	$(\mathcal{T}_1^{odd})_s$	27
				F	$\mathcal{T}_{m,n}$	$(\mathcal{T}_{m,n})^2$	$(\mathcal{T}_{m,n})_s$	$((\mathcal{T}_{m,n})^2)_s$	28
		F	T	T	Theorem 3				29
				F					30
			F	T	Theorem 9	$\mathcal{T}_1^\infty$	$\mathcal{T}_{1,\infty}^{\neq}$	$(\mathcal{T}_1^\infty)_s$	31
				F		$\mathcal{T}_2^\infty$	$\mathcal{T}_{2,\infty}^{\neq}$	$(\mathcal{T}_2^\infty)_s$	32

- Convexity (CV): if $\phi \to \bigvee_{i=1}^{n} x_i = y_i$ then $\phi \to x_i = y_i$

Full Table

					Empty		Non-empty		
SI	**SM**	**FW**	**SW**	**CV**	One-sorted	Many-sorted	One-sorted	Many-sorted	N^o
T	T	T	T	T	$\mathcal{T}_{\geq n}$	$(\mathcal{T}_{\geq n})^2$	$(\mathcal{T}_{\geq n})_s$	$((\mathcal{T}_{\geq n})^2)_s$	1
				F	Theorem 8		$(\mathcal{T}_{\geq n})_\vee$	$((\mathcal{T}_{\geq n})^2)_\vee$	2
			F	T	Theorem 4	$\mathcal{T}_{2,3}$	$\mathcal{T}_f$	$(\mathcal{T}_f)_s$	3
				F		Theorem 8	$\mathcal{T}_f^s$	$(\mathcal{T}_{2,3})_\vee$	4
		F	T	T	Theorem 3				5
				F					6
			F	T	$\mathcal{T}_\infty$	$(\mathcal{T}_\infty)^2$	$(\mathcal{T}_\infty)_s$	$((\mathcal{T}_\infty)^2)_s$	7
				F	Theorem 8		$(\mathcal{T}_\infty)_\vee$	$((\mathcal{T}_\infty)^2)_\vee$	8
	F	T	T	T	Theorem 10	Unicorn Theories	Theorem 10	Unicorn Theories	9
				F		Theorem 8			10
			F	T	$\mathcal{T}_{even}^\infty$	$(\mathcal{T}_{even}^\infty)^2$	$(\mathcal{T}_{even}^\infty)_s$	$((\mathcal{T}_{even}^\infty)^2)_s$	11
				F	Theorem 8		$(\mathcal{T}_{even}^\infty)_\vee$	$((\mathcal{T}_{even}^\infty)^2)_\vee$	12
		F	T	T	Theorem 3				13
				F					14
			F	T	$\mathcal{T}_{n,\infty}$	$(\mathcal{T}_{n,\infty})^2$	$(\mathcal{T}_{n,\infty})_s$	$((\mathcal{T}_{n,\infty})^2)_s$	15
				F	Theorem 8		$(\mathcal{T}_{n,\infty})_\vee$	$((\mathcal{T}_{n,\infty})^2)_\vee$	16
F	T	T	T	T	Theorem 2				17
				F					18
			F	T					19
				F					20
		F	T	T	Theorems 2 and 3				21
				F					22
			F	T	Theorem 2				23
				F					24
	F	T	T	T	$\mathcal{T}_{\leq 1}$	$(\mathcal{T}_{\leq 1})^2$	$(\mathcal{T}_{\leq 1})_s$	$((\mathcal{T}_{\leq 1})^2)_s$	25
				F	$\mathcal{T}_{\leq n}$	$(\mathcal{T}_{\leq n})^2$	$(\mathcal{T}_{\leq n})_s$	$((\mathcal{T}_{\leq n})^2)_s$	26
			F	T	Theorem 11	$\mathcal{T}_1^{odd}$	$\mathcal{T}_{odd}^{\neq}$	$(\mathcal{T}_1^{odd})_s$	27
				F	$\mathcal{T}_{m,n}$	$(\mathcal{T}_{m,n})^2$	$(\mathcal{T}_{m,n})_s$	$((\mathcal{T}_{m,n})^2)_s$	28
		F	T	T	Theorem 3				29
				F					30
			F	T	Theorem 9	$\mathcal{T}_1^\infty$	$\mathcal{T}_{1,\infty}^{\neq}$	$(\mathcal{T}_1^\infty)_s$	31
				F		$\mathcal{T}_2^\infty$	$\mathcal{T}_{2,\infty}^{\neq}$	$(\mathcal{T}_2^\infty)_s$	32

- 5 Propertoes

Full Table

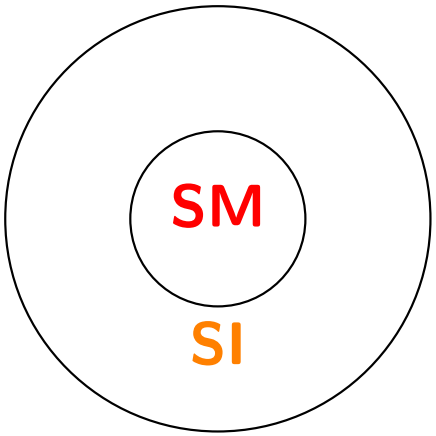
					Empty		Non-empty		
SI	**SM**	**FW**	**SW**	**CV**	One-sorted	Many-sorted	One-sorted	Many-sorted	N^o
T	T	T	T	T	$\mathcal{T}_{\geq n}$	$(\mathcal{T}_{\geq n})^2$	$(\mathcal{T}_{\geq n})_s$	$((\mathcal{T}_{\geq n})^2)_s$	1
				F	Theorem 8		$(\mathcal{T}_{\geq n})_\vee$	$((\mathcal{T}_{\geq n})^2)_\vee$	2
			F	T	Theorem 4	$\mathcal{T}_{2,3}$	$\mathcal{T}_f$	$(\mathcal{T}_f)_s$	3
				F		Theorem 8	$\mathcal{T}_f^s$	$(\mathcal{T}_{2,3})_\vee$	4
		F	T	T	Theorem 3				5
				F					6
			F	T	$\mathcal{T}_\infty$	$(\mathcal{T}_\infty)^2$	$(\mathcal{T}_\infty)_s$	$((\mathcal{T}_\infty)^2)_s$	7
				F	Theorem 8		$(\mathcal{T}_\infty)_\vee$	$((\mathcal{T}_\infty)^2)_\vee$	8
	F	T	T	T	Theorem 10	Unicorn Theories	Theorem 10	Unicorn Theories	9
				F		Theorem 8			10
			F	T	$\mathcal{T}_{even}^\infty$	$(\mathcal{T}_{even}^\infty)^2$	$(\mathcal{T}_{even}^\infty)_s$	$((\mathcal{T}_{even}^\infty)^2)_s$	11
				F	Theorem 8		$(\mathcal{T}_{even}^\infty)_\vee$	$((\mathcal{T}_{even}^\infty)^2)_\vee$	12
		F	T	T	Theorem 3				13
				F					14
			F	T	$\mathcal{T}_{n,\infty}$	$(\mathcal{T}_{n,\infty})^2$	$(\mathcal{T}_{n,\infty})_s$	$((\mathcal{T}_{n,\infty})^2)_s$	15
				F	Theorem 8		$(\mathcal{T}_{n,\infty})_\vee$	$((\mathcal{T}_{n,\infty})^2)_\vee$	16
F	T	T	T	T	Theorem 2				17
				F					18
			F	T					19
				F					20
		F	T	T	Theorems 2 and 3				21
				F					22
			F	T	Theorem 2				23
				F					24
	F	T	T	T	$\mathcal{T}_{\leq 1}$	$(\mathcal{T}_{\leq 1})^2$	$(\mathcal{T}_{\leq 1})_s$	$((\mathcal{T}_{\leq 1})^2)_s$	25
				F	$\mathcal{T}_{\leq n}$	$(\mathcal{T}_{\leq n})^2$	$(\mathcal{T}_{\leq n})_s$	$((\mathcal{T}_{\leq n})^2)_s$	26
			F	T	Theorem 11	$\mathcal{T}_1^{odd}$	$\mathcal{T}_{odd}^{\neq}$	$(\mathcal{T}_1^{odd})_s$	27
				F	$\mathcal{T}_{m,n}$	$(\mathcal{T}_{m,n})^2$	$(\mathcal{T}_{m,n})_s$	$((\mathcal{T}_{m,n})^2)_s$	28
		F	T	T	Theorem 3				29
				F					30
			F	T	Theorem 9	$\mathcal{T}_1^\infty$	$\mathcal{T}_{1,\infty}^{\neq}$	$(\mathcal{T}_1^\infty)_s$	31
				F		$\mathcal{T}_2^\infty$	$\mathcal{T}_{2,\infty}^{\neq}$	$(\mathcal{T}_2^\infty)_s$	32

- Next session: two more properties!

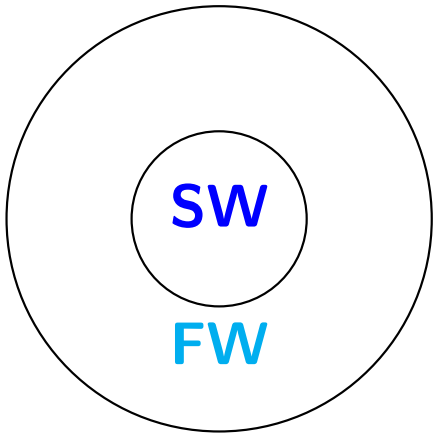
What Does The Table Mean?

- Red Background: No theories with these properties
- White Background: We have examples
  - Almost always...

Negative results

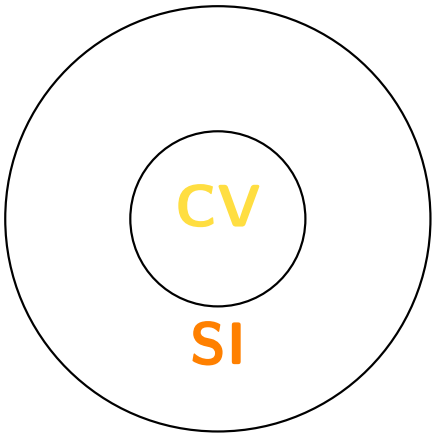


Any signatures.

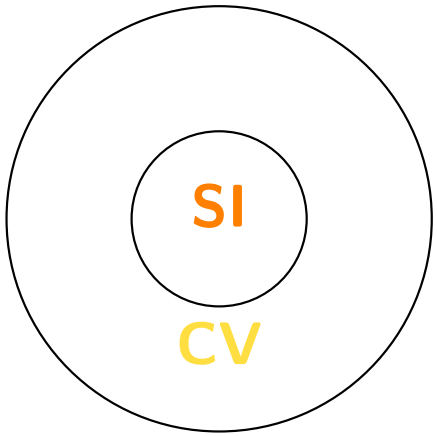


Any signatures.

Negative results

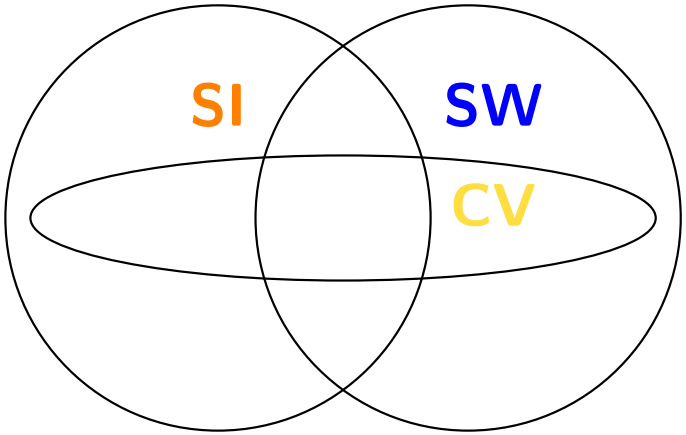


Non-trivial models only.

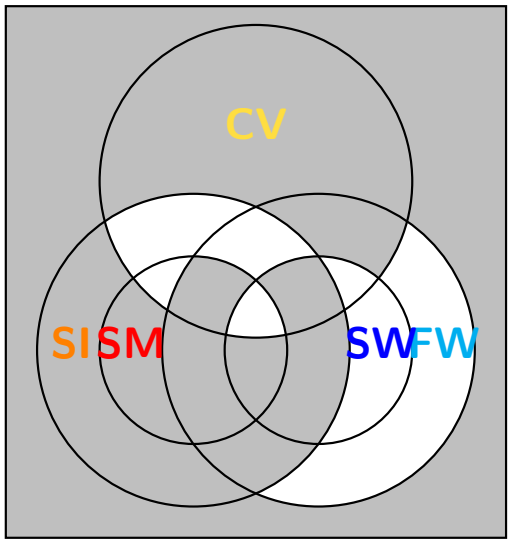


Empty signatures only.

Negative results



One-sorted, empty signatures only.

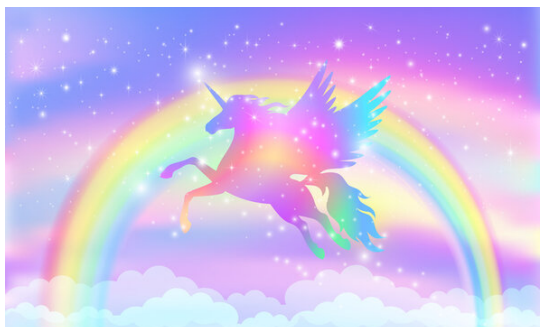


One-sorted, empty signatures only.

Unicorn theories

Theorem 10	Unicorn Theories Theorem 8	Theorem 10	Unicorn Theories	9 10

- Unicorn theories= **SI** + **SW** + ¬**SM**
- We don't believe they exist...
- But still don't have a proof of that
- In progress





The Most Difficult Case (that we found...)

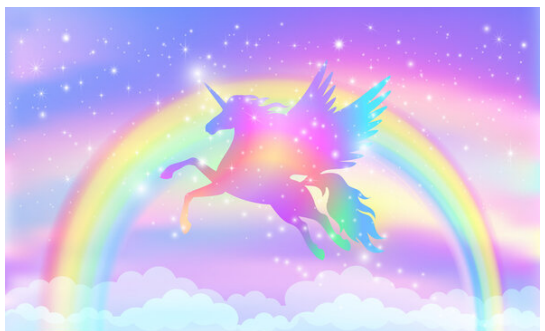
From Two Sorts to One

- polite != strongly polite using $\mathcal{T}_{2,3}$
- $\mathcal{T}_{2,3}$ is two-sorted
- Is there a one-sorted separation?
- Found such an example – involving non-computable functions

Empty		Non-empty		
One-sorted	Many-sorted	One-sorted	Many-sorted	N^o
$\mathcal{T}_{\geq n}$	$(\mathcal{T}_{\geq n})^2$	$(\mathcal{T}_{\geq n})_s$	$((\mathcal{T}_{\geq n})^2)_s$	1
Theorem 8		$(\mathcal{T}_{\geq n})_\vee$	$((\mathcal{T}_{\geq n})^2)_\vee$	2
Theorem 4	$\mathcal{T}_{2,3}$	$\mathcal{T}_f$	$(\mathcal{T}_f)_s$	3

Combination Properties – Summary

- We studied all possible combination of combination properties
- Some easy, some hard
- Unicorn Theories





Recent Developments in Theory Combination: Datatypes, Synergies, and Unicorns

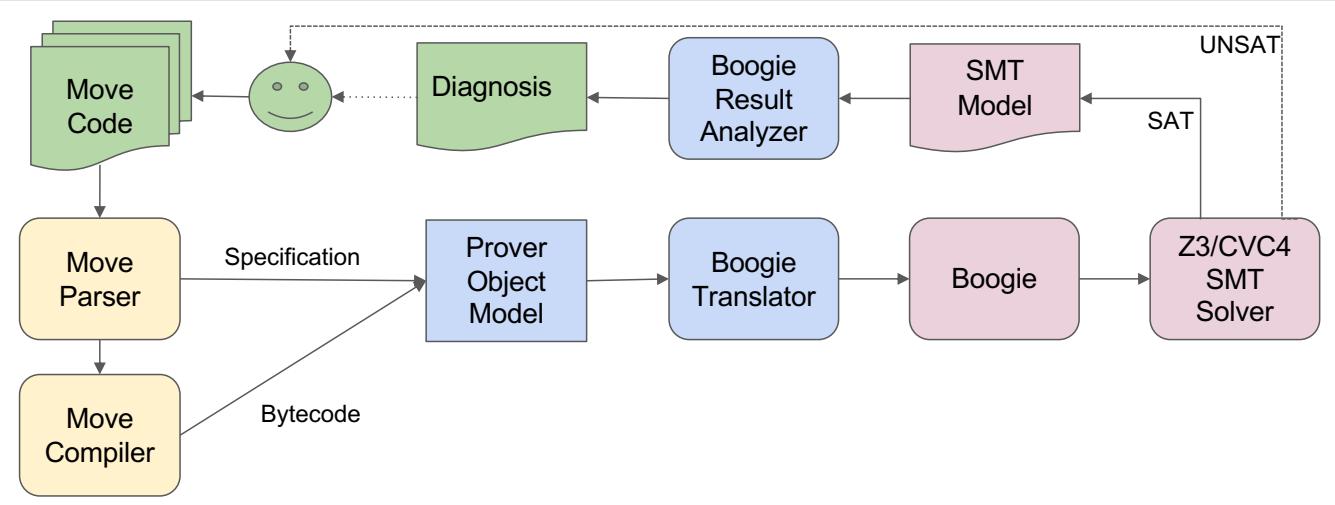


Recent Developments in Theory Combination: Datatypes, **Synergies**, and Unicorns

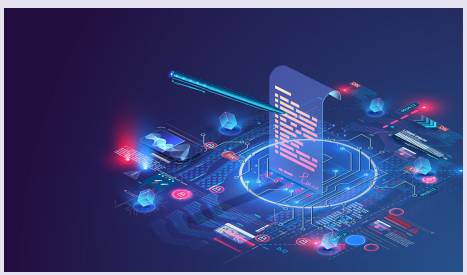


Motivation

The Move Prover [Zhong et al. 2020]

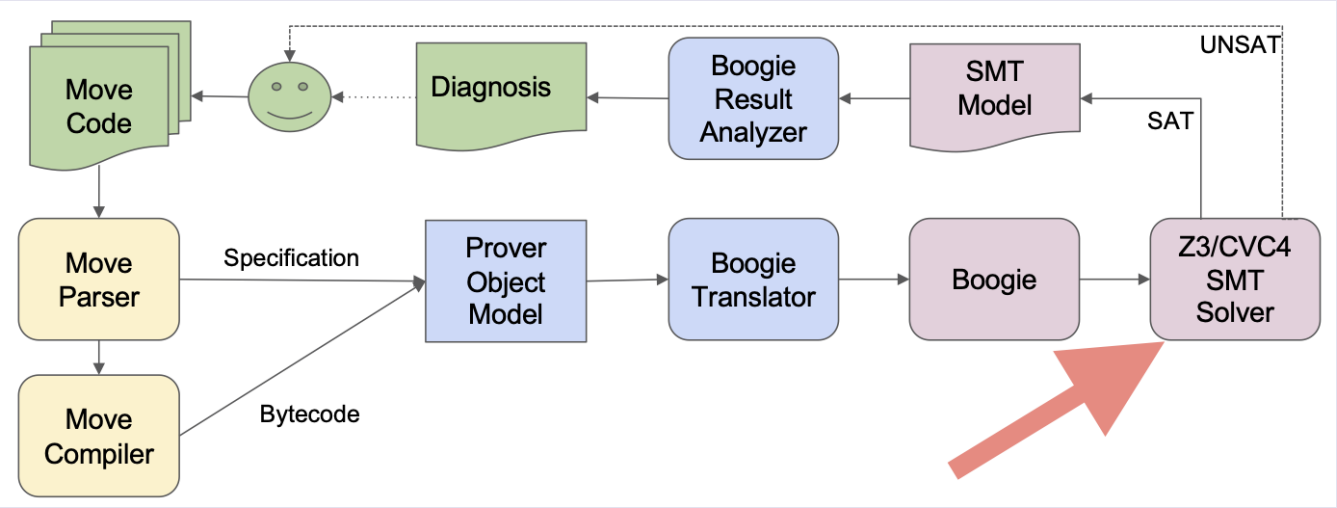


- Formal verification tool for smart contracts
- Inspires many projects in cvc5:
  - sequences, (nested) datatypes
  - quantifiers
  - theory combination

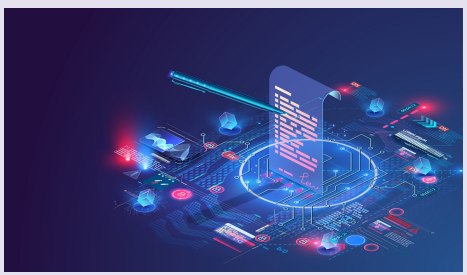


Motivation

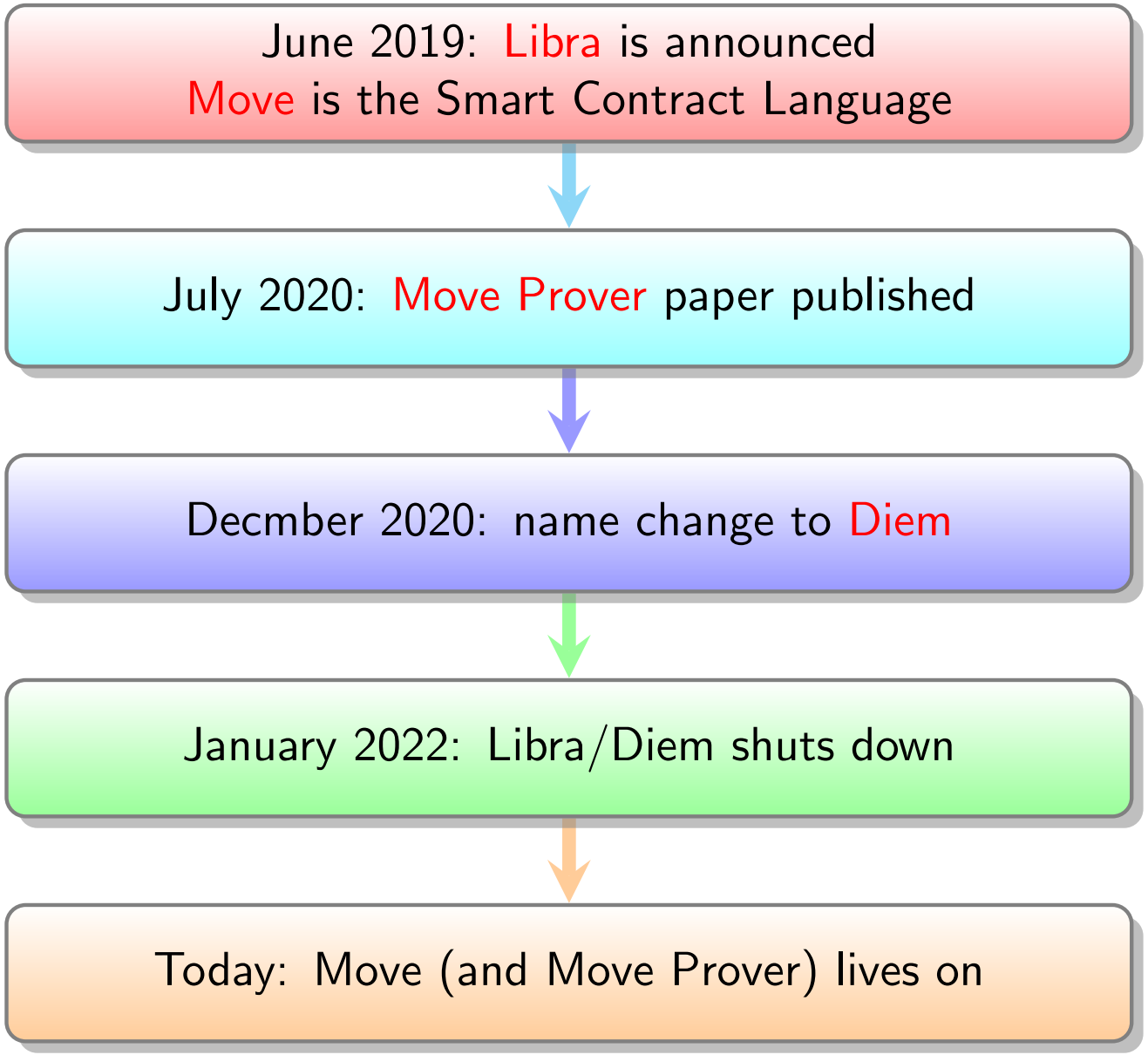
The Move Prover [Zhong et al. 2020]



- Formal verification tool for smart contracts
- Inspires many projects in cvc5:
  - sequences, (nested) datatypes
  - quantifiers
  - theory combination



More Background



This Work $\longrightarrow$

From Smart Contracts to Theory Combination



- Improving theory combination for Move Prover benchmarks
- Potential: reduce number of variables in arrangements
- Reasoning about arrangements is exponential

From Smart Contracts to Theory Combination



- Improving theory combination for Move Prover benchmarks
- Potential: reduce number of variables in arrangements
- Reasoning about arrangements is exponential

Combination Methods – Summary

Let S be the set of shared sorts.

$A \wedge B$ is $(T_1 \oplus T_2)$-SAT

$\Leftrightarrow$

Nelson-Oppen Combination

$\exists\ \delta$ over $FV(A) \cap FV(B)$ s.t.
$A \wedge \delta$ is T_1-SAT and $B \wedge \delta$ is T_2-SAT

T_1 and T_2 are stably infinite w.r.t. S

Polite Combination

$\exists\ \delta$ over $FV(B)$ s.t.
$A \wedge \delta$ is T_1-SAT and $wit(B) \wedge \delta$ is T_2-SAT

T_2 is strongly polite w.r.t. S

Combination Methods – Summary

Let S be the set of shared sorts.

$A \wedge B$ is $(T_1 \oplus T_2)$-SAT

$\Leftrightarrow$

Nelson-Oppen Combination

$\exists\ \delta$ over $FV(A) \cap FV(B)$ s.t.
$A \wedge \delta$ is T_1-SAT and $B \wedge \delta$ is T_2-SAT

T_1 and T_2 are stably infinite w.r.t. S

Polite Combination

$\exists\ \delta$ over $FV(B)$ s.t.
$A \wedge \delta$ is T_1-SAT and $wit(B) \wedge \delta$ is T_2-SAT

T_2 is strongly polite w.r.t. S

Hybrid?

$\exists\ \delta$ over $FV_{S^{si}}(A) \cap FV_{S^{si}}(B)$, $FV_{S^{nsi}}(B)$ s.t.
$A \wedge \delta$ is T_1-SAT and $wit(B) \wedge \delta$ is T_2-SAT

T_1 is stably-infinite w.r.t. S^{si}
T_2 is strongly polite w.r.t. S
$S = S^{si} \cup S^{nsi}$

First Attempts

$$S = S^{nsi} \cup S^{si}$$

Suppose T_1 is stably-infinite w.r.t. S^{si}.

Theorem?

If T_2 is:

- strongly polite w.r.t. S^{nsi}
- stably-infinite w.r.t. S^{si}

Then:

$$A \wedge B \text{ is } (T_1 \oplus T_2)\text{-SAT}$$

$$\Leftrightarrow$$

There exists an arangement δ over $FV_{S^{si}}(A) \cap FV_{S^{si}}(B)$ and $FV_{S^{nsi}}(B)$ s.t. $A \wedge \delta$ is T_1-SAT and $wit(B) \wedge \delta$ is T_2-SAT.

First Attempts

$$S = S^{nsi} \cup S^{si}$$

Suppose T_1 is stably-infinite w.r.t. S^{si}.

Theorem

If T_2 is:

- strongly polite w.r.t. S^{nsi}
- stably-infinite w.r.t. S^{si} without changing S^{nsi}

Then:

$$A \wedge B \text{ is } (T_1 \oplus T_2)\text{-SAT}$$
$$\Leftrightarrow$$
There exists an arangement δ over $FV_{S^{si}}(A) \cap FV_{S^{si}}(B)$ and $FV_{S^{nsi}}(B)$ s.t. $A \wedge \delta$ is T_1-SAT and $wit(B) \wedge \delta$ is T_2-SAT.

First Attempts

$$S = S^{nsi} \cup S^{si}$$

Suppose T_1 is stably-infinite w.r.t. S^{si}.

Theorem

If T_2 is:

- smooth w.r.t. S^{nsi} without changing infiniteness of S^{si}
- strongly finitely witnessable w.r.t. S^{nsi}
- stably-infinite w.r.t. S^{si} without increasing S^{nsi}

Then:

$$A \wedge B \text{ is } (T_1 \oplus T_2)\text{-SAT}$$

$$\Leftrightarrow$$

There exists an arangement δ over $FV_{S^{si}}(A) \cap FV_{S^{si}}(B)$ and $FV_{S^{nsi}}(B)$ s.t. $A \wedge \delta$ is T_1-SAT and $wit(B) \wedge \delta$ is T_2-SAT.

First Attempts

$$S = S^{nsi} \cup S^{si}$$

Suppose T_1 is stably-infinite w.r.t. S^{si}.

Theorem

If T_2 is:

- strongly polite w.r.t. S^{nsi} without changing infiniteness of S^{si}
- stably-infinite w.r.t. S^{si}

Then:

$$A \wedge B \text{ is } (T_1 \oplus T_2)\text{-SAT}$$
$$\Leftrightarrow$$
There exists an arangement δ over $FV_{S^{si}}(A) \cap FV_{S^{si}}(B)$ and $FV_{S^{nsi}}(B)$ s.t. $A \wedge \delta$ is T_1-SAT and $wit(B) \wedge \delta$ is T_2-SAT.

Optimized Combination

The original conjecture follows from the general variants.

Theorem

If:

- $S^{nsi} \cup S^{si}$ *is the set of shared sorts*
- T_2 *is strongly polite w.r.t.* $S^{nsi} \cup S^{si}$
- T_1 *is stably-infinite w.r.t.* S^{si}

Then:

$$A \wedge B \text{ is } (T_1 \oplus T_2)\text{-SAT}$$
$$\Leftrightarrow$$
There exists an arangement δ *over* $FV_{S^{si}}(A) \cap FV_{S^{si}}(B)$ *and* $FV_{S^{nsi}}(B)$ *s.t.* $A \wedge \delta$ *is* T_1*-SAT and* $wit(B) \wedge \delta$ *is* T_2*-SAT.*

Example



Arith+BV	DT
$x = 5$	$a_0 = cons(x, v, nil)$
$v = 0000$	$a_1 = cons(y_1, v, a_0)$
	...
	$a_n = cons(y_n, v, a_{n-1})$

- Arith+BV is stably infinite w.r.t. Arith
- DT is strongly polite w.r.t. *Arith* and *BV*

- Polite combination: arr. over $|\{x, v, y_1, \ldots, y_n\}| = n + 2$ variables
- Optimized combination: arr. over $|\{x, v\}| = 2$ variables

A challenging Combination Benchmark

On most Move Prover benchmarks: $< 10\%$ spent on theory combination

Except 1 benchmark:

- 81s solving
- 20s combination

```
(set-logic UFDTNIA)
(declare-sort T@$TypeName 0)
(declare-sort |T@[Int]$TypeValue| 0)
(declare-datatypes ((T@$TypeValue 0)(T@$TypeValueArray 0)) (((($BooleanType ) ($IntegerType ) ($Ad
(declare-sort |T@[Int]$Value| 0)
(declare-datatypes ((T@$Value 0)(T@$ValueArray 0)) (((($Boolean (|b#$Boolean| Bool) ) ($Integer (|
(declare-sort |T@[$TypeValueArray,Int]Bool| 0)
(declare-sort |T@[$TypeValueArray,Int]$Value| 0)
(declare-datatypes ((T@$Memory 0)) ((($Memory (|domain#$Memory| |T@[$TypeValueArray,Int]Bool|) (|
(declare-datatypes ((T@$Location 0)) ((($Global (|ts#$Global| T@$TypeValueArray) (|a#$Global| Int
(declare-sort |T@[Int]Int| 0)
(declare-datatypes ((T@$Path 0)) ((($Path (|p#$Path| |T@[Int]Int|) (|size#$Path| Int) ) ) ))
```

Theories

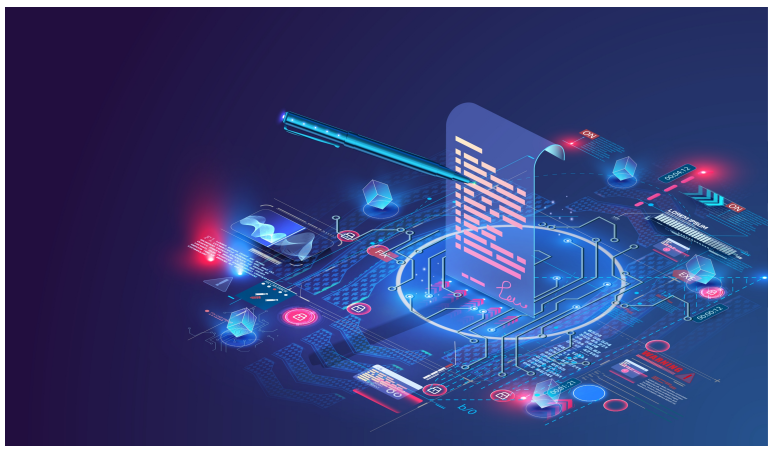
- Bool+UF+Arith: stably-infinite w.r.t. Arith
- Dataypes: strongly polite w.r.t. Bool+UF+Arith

	total (s)	comb (s)	DT	INT	UFB	shared
original	81.5	20.3	116.0	281.0	123.9	163.5
optimized	34.9	3.4	236.1	212.1	78.4	125.8

Running times (in seconds) and number of terms (in thousands)

Synergies Conclusion

- Polite Combination and Nelson Oppen are Stronger Together
- Reducing the sizes of arrangements is important
- Practical motivation and impact
- Interesting Theory

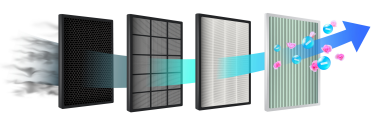


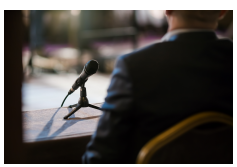
Recent Developments in Theory Combination: Datatypes, Synergies, and Unicorns

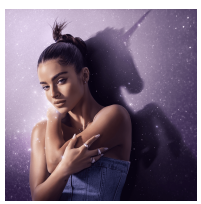
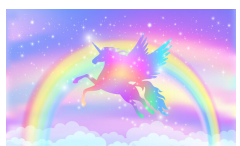


Conclusion

- Theory ombination was asleep for a while
- Renewed interest
  - These slides
  - Bridging functions (Chocron et al.)
  - CDSAT (Bonacina et al.)
  - ...
- Cool new methods
- Cool new results
  - Datatypes are polite
  - Combination of combination properties
  - Combination of combination methods

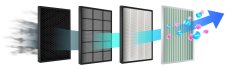




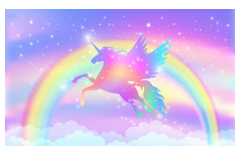



Further Work

- Chasing down unicorns
  - In Progress
- Politeness for more theories
  - Nested Datatypes
  - Sequences
  - Sets
  - ...
- Other properties
  - Stable finiteness and FMP
  - Gentle theories, shiny theories
  - ...
- Preservation of properties after combination

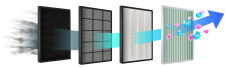




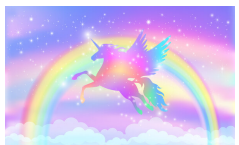



Further Work

- Chasing down unicorns
  - In Progress
- Politeness for more theories
  - Nested Datatypes
  - Sequences
  - Sets
  - ...
- Other properties
  - Stable finiteness and FMP
  - Gentle theories, shiny theories
  - ...
- Preservation of properties after combination





Thank you!