Politeness and Stable Infiniteness: Stronger Together

Ying Sheng[1] Yoni Zohar[1] Christophe Ringeissen[2]
Andrew Reynolds[3] Clark Barrett[1] Cesare Tinelli[3]

Stanford University

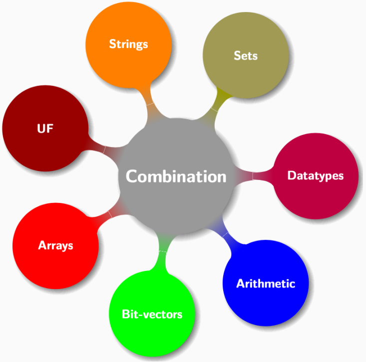
Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France

The University of Iowa

What is This About?

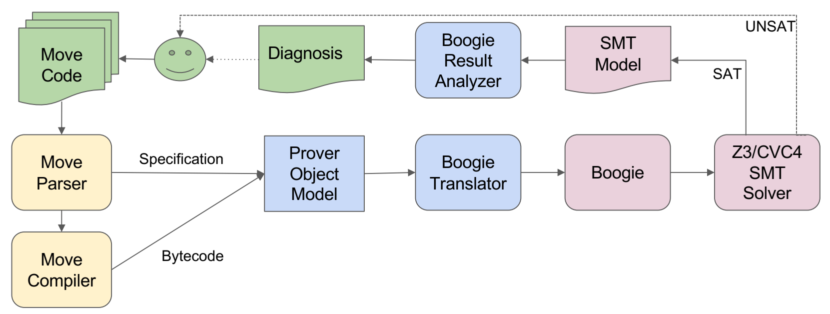
Theory Combination

- specialized theory solvers
- how to split the problem?
- how to combine the results?



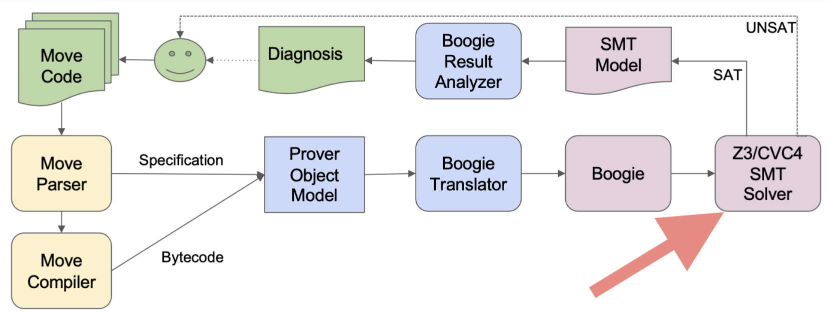

```
(set-logic UFDTNIA)
(declare-sort T@$TypeName 0)
(declare-sort |T@[Int]$TypeValue| 0)
(declare-datatypes ((T@$TypeValue 0)(T@$TypeValueArray 0)) ((($BooleanType ) ($IntegerType ) ($Ad
(declare-sort |T@[Int]$Value| 0)
(declare-datatypes ((T@$Value 0)(T@$ValueArray 0)) ((($Boolean (|b#$Boolean| Bool) ) ($Integer (|
(declare-sort |T@[$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) ) ) ))
```

The Move Prover [Zhong et al. 2020]



- Formal verification tool for Move smart contracts in the Diem blockchain
- Inspires many current projects in CVC4:
  - sequences
  - (nested) datatypes
  - quantifiers
  - **theory combination**

The Move Prover [Zhong et al. 2020]



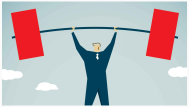
- Formal verification tool for Move smart contracts in the Diem blockchain
- Inspires many current projects in CVC4:
  - sequences
  - (nested) datatypes
  - quantifiers
  - **theory combination**

Background:
Theory Combination

Motivation:
Smart Contracts Verification

Result 1:
Politeness $\neq$ Strong Politeness

Result 2:
Politeness $\oplus$ Stable-infiniteness



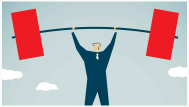
Overview

Background:
Theory Combination

Motivation:
Smart Contracts Verification

Result 1:
Politeness $\neq$ Strong Politeness

Result 2:
Politeness $\oplus$ Stable-infiniteness



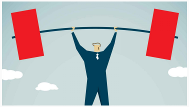
Overview

Background:
Theory Combination

Motivation:
Smart Contracts Verification

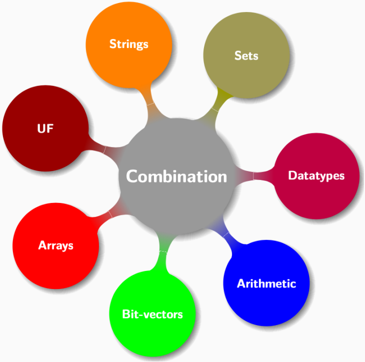
Result 1:
Politeness $\neq$ Strong Politeness

Result 2:
Politeness $\oplus$ Stable-infiniteness



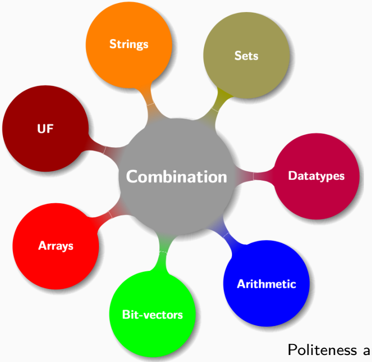

arity
equality
formula
theory
signature
sort
literal
countable
class
structure
interpretation
Disjoint
Function
free
symbol
axiomatize
predicate
variable
sentence
SMT
empty
order
first
sentences
semantics
extension
LIB
relation
quantifier
Satisfaction
terms
equivalent
domain
standard
identity
Term
logic

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



Example: Lists with A Single Element

$a_0, a_1, a_2 \neq nil$ $a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$ $tail(a_0) = tail(a_1) = tail(a_2) = nil$	$\boxed{\ }$ $\boxed{\ }$ $\boxed{\ }$ a_0 a_1 a_2
$0 \leq x, y, z \leq 1$ $x = head(a_0), y = head(a_1), z = head(a_2)$	$\boxed{0 \mid 1}$ $\boxed{0 \mid 1}$ $\boxed{0 \mid 1}$ a_0 a_1 a_2



Naive Method

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

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**

☹

Missing **arrangement** δ: $=$ / $\neq$ between variables

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
$a_0, a_1, a_2 \neq nil$	$0 \leq x, y, z \leq 1$
$a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$	
$tail(a_0) = tail(a_1) = tail(a_2) = nil$	
$x = head(a_0), y = head(a_1), z = head(a_2)$	
$\delta : x \neq y \neq z$	$\delta : \neg(x \neq y \neq z)$

The formlua is **UNSAT** Method says **UNSAT**

☺

Combination Methods

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?

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
$a_0, a_1, a_2 \neq nil$ $a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$ $tail(a_0) = tail(a_1) = tail(a_2) = nil$	$TRUE$
SAT	SAT

The formlua is **UNSAT** Method says **SAT**

☹

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
$wit(a_0, a_1, a_2 \neq nil)$	$TRUE$
$wit(a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2)$	
$wit(tail(a_0) = tail(a_1) = tail(a_2) = nil)$	

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
$a_0, a_1, a_2 \neq nil$ $a_0 \neq a_1, a_1 \neq a_2, a_0 \neq a_2$ $\bigwedge_{i=0}^{2} a_i = cons(vi, nil)$	$TRUE$
δ: $v_0 \neq v_1 \neq v_2$	δ: $\neg(v_0 \neq v_1 \neq v_2)$

The formlua is **UNSAT** Method says **UNSAT**

☺

Combination Methods – Summary

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

<u>Naive Combination</u>

A is T_1-SAT and B is T_2-SAT

No shared variables

<u>Nelson-Oppen Combination</u>

$\exists\ \delta$ 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**

<u>Polite Combination</u>

$\exists\ \delta$ 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**

Let S be a set of sorts.

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 $\mathcal{I}$
- mapping κ from S to carinalities with $\kappa(\sigma) \geq |\sigma^{\mathcal{I}}|$

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

- Smoothness $\Rightarrow$ Stable Infiniteness
- Lists are smooth w.r.t. element sort
- BV is not smooth w.r.t. $BV[4]$

Let S be a set of sorts.

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

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** w.r.t. S if it is smooth and strongly finitely witnessable w.r.t. S.

Question

Politeness = Strong Politeness ?

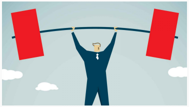
Overview

Background:
Theory Combination

Motivation:
Smart Contracts Verification

Result 1:
Politeness $\neq$ Strong Politeness

Result 2:
Politeness $\oplus$ Stable-infiniteness



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 $|\sigma_2^{\mathcal{I}}| = \infty$
  - $|\sigma_1^{\mathcal{I}}|, |\sigma_2^{\mathcal{I}}| \geq 3$

lemma

- $T_{2,3}$ is polite.
- $wit(A) = A \wedge \bigwedge_{i=1}^{3} x_i = x_i \wedge \bigwedge_{i=1}^{3} y_i = y_i$
- Every witness is not a strong witness.

First Contribution

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
- Our proof is direct

Theorem

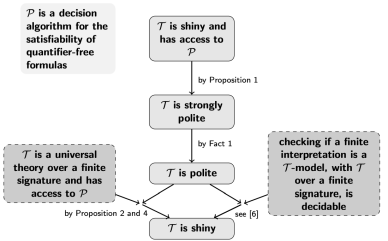
For empty mono-sorted signatures: politeness = strong politeness.

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

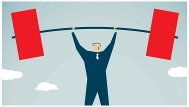
Overview

Background:
Theory Combination

Motivation:
Smart Contracts Verification

Result 1:
Politeness $\neq$ Strong Politeness

Result 2:
Politeness $\oplus$ Stable-infiniteness



Importance

Why Bother With Polite Theories?

- We plan to prove politeness for more theories
  - Datatypes (done)
  - Nested Datatypes
  - Sequences

Negative Reason

- Proving strong politeness is harder
- Now we know that sometimes the additional effort is not for nothing

Positive Reason

- Politeness can be used to prove strong politeness
- For datatypes, we did as follows [Sheng et al. 2020]:
  - Introduced a sufficient for equivalence of strong and ordinary politeness
  - Proved politeness + sufficient condition
  - Concluded strong politeness

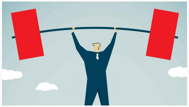
Overview

Background:
Theory Combination

Motivation:
Smart Contracts Verification

Result 1:
Politeness $\neq$ Strong Politeness

Result 2:
Politeness $\oplus$ Stable-infiniteness



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

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

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:

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

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.

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

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:

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

$$\Leftrightarrow$$

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

Example

$\boxed{5 \mid 0000}_{a_0} \leftarrow \boxed{y_1 \mid 0000}_{a_1} \leftarrow \ldots \leftarrow \boxed{y_n \mid 0000}_{a_n}$

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

- Arith+BV is stably infinite w.r.t. Arith
- DT is strongly polite w.r.t. the elemnt sorts, denoted *Arith* and *BV*

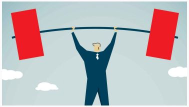
- Polite combination: arrangments over $\{x, v, y_1, \ldots, y_n\}$ – $n + 2$ variables
- Optimized combination: arrangements over $\{x, v\}$ – 2 variables

Background:
Theory Combination

Motivation:
Smart Contracts Verification

Result 1:
Politeness $\neq$ Strong Politeness

Result 2:
Politeness $\oplus$ Stable-infiniteness



A challenging Combination Benchmark

On most benchmarks from Move Prover:

$< 10\%$ and $< 10s$ time spent on theory combination (300s timeout)

Except 1 benchmark:

- 81s solving
- 20s combination (24%)

```
(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)

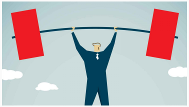
Conclusion

We Have Seen

- Politeness $\neq$ Strong Politeness
- Synergy of two combination methods

What's next?

- Sufficient conditions for equivalence
- Combination method based on Politeness
- Implementation: Optimize for other theories
- Evaluation: Collect and create more benchmarks



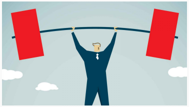
Conclusion

We Have Seen

- Politeness $\neq$ Strong Politeness
- Synergy of two combination methods

What's next?

- Sufficient conditions for equivalence
- Combination method based on Politeness
- Implementation: Optimize for other theories
- Evaluation: Collect and create more benchmarks


Thank You !