

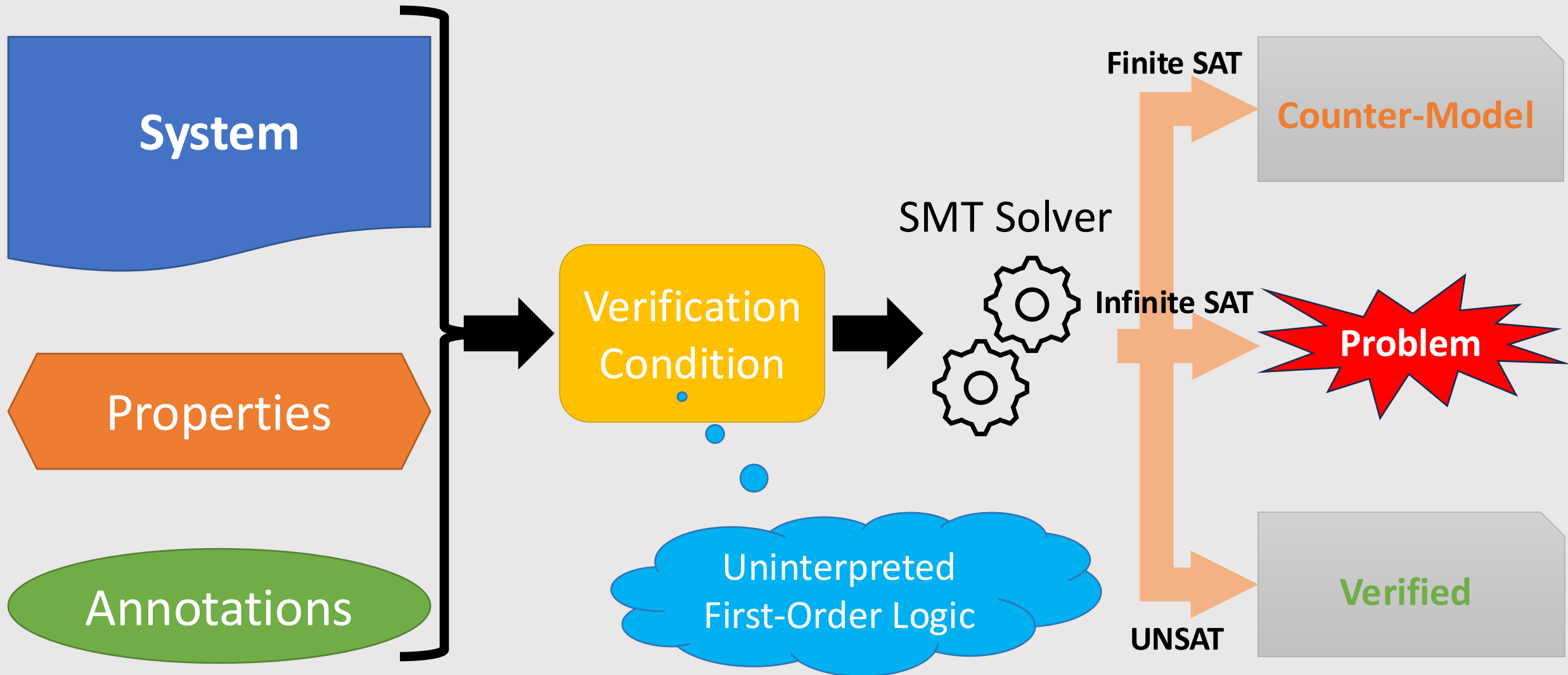
An Infinite Needle in a Finite Haystack

Finding Infinite Counter-Models in Deductive Verification

Neta Elad, Oded Padon, Sharon Shoham

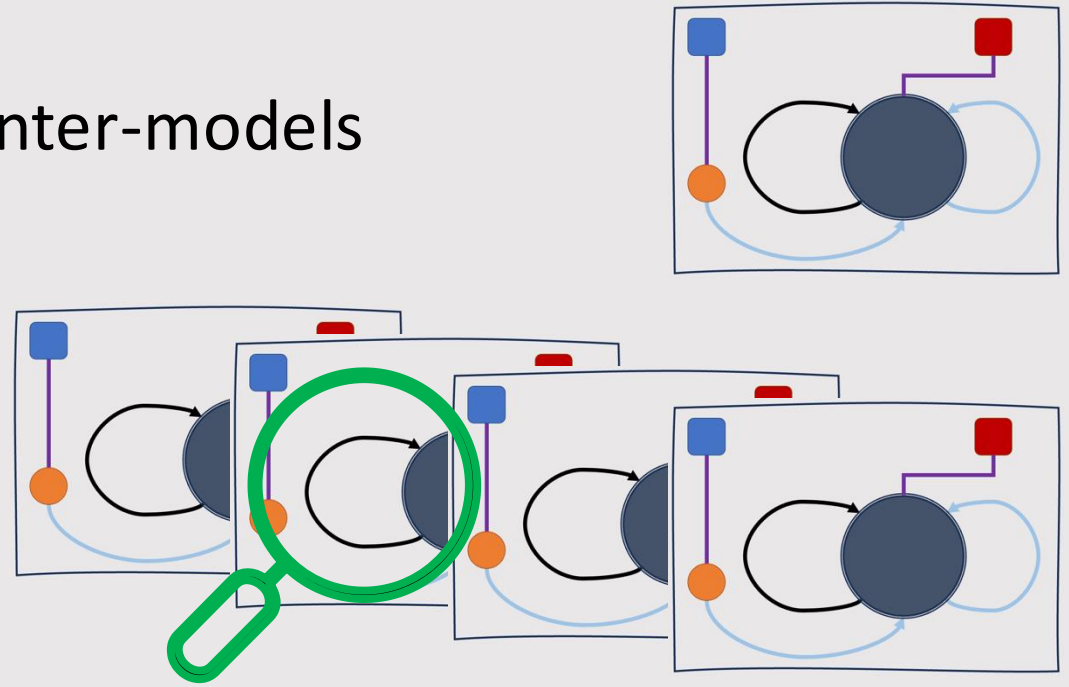


SMT-Based Deductive Verification



This Work

- Finite representation of infinite counter-models
 - Enables simple model-checking
- Efficient search procedure
- Decidability result
 - Fragment of formulas, for which we can always find an infinite model, or disprove its existence



Motivation: Infinite Objects from Abstractions

Distributed Protocols

Rounds (natural numbers)

Linked Lists

Nodes in the heap

TOY Example: Simplified VC from Paxos

Round

Value

Total Order Abstraction " $<$ "

- Anti-reflexive: $\forall R. R \not< R$
- Transitive: $\forall R_1, R_2, R_3. R_1 < R_2 \wedge R_2 < R_3 \rightarrow R_1 < R_3$
- Total: $\forall R_1, R_2. R_1 \neq R_2 \rightarrow R_1 < R_2 \vee R_2 < R_1$

$\forall R, V. \text{proposal}(R, V) \rightarrow \text{safe}(V) \vee \exists R'. R' < R \wedge \text{proposal}(R', V)$

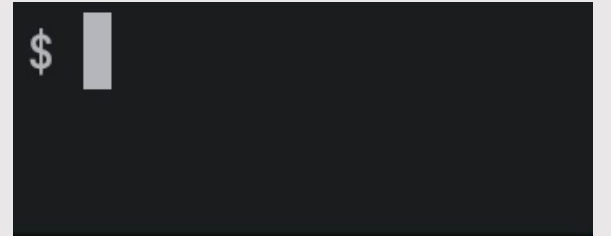
$\forall V_1, V_2. \text{safe}(V_1) \wedge \text{safe}(V_2) \rightarrow V_1 = V_2$

$\text{proposal}(r_1, v_1) \wedge \text{proposal}(r_2, v_2) \wedge v_1 \neq v_2$

TOY

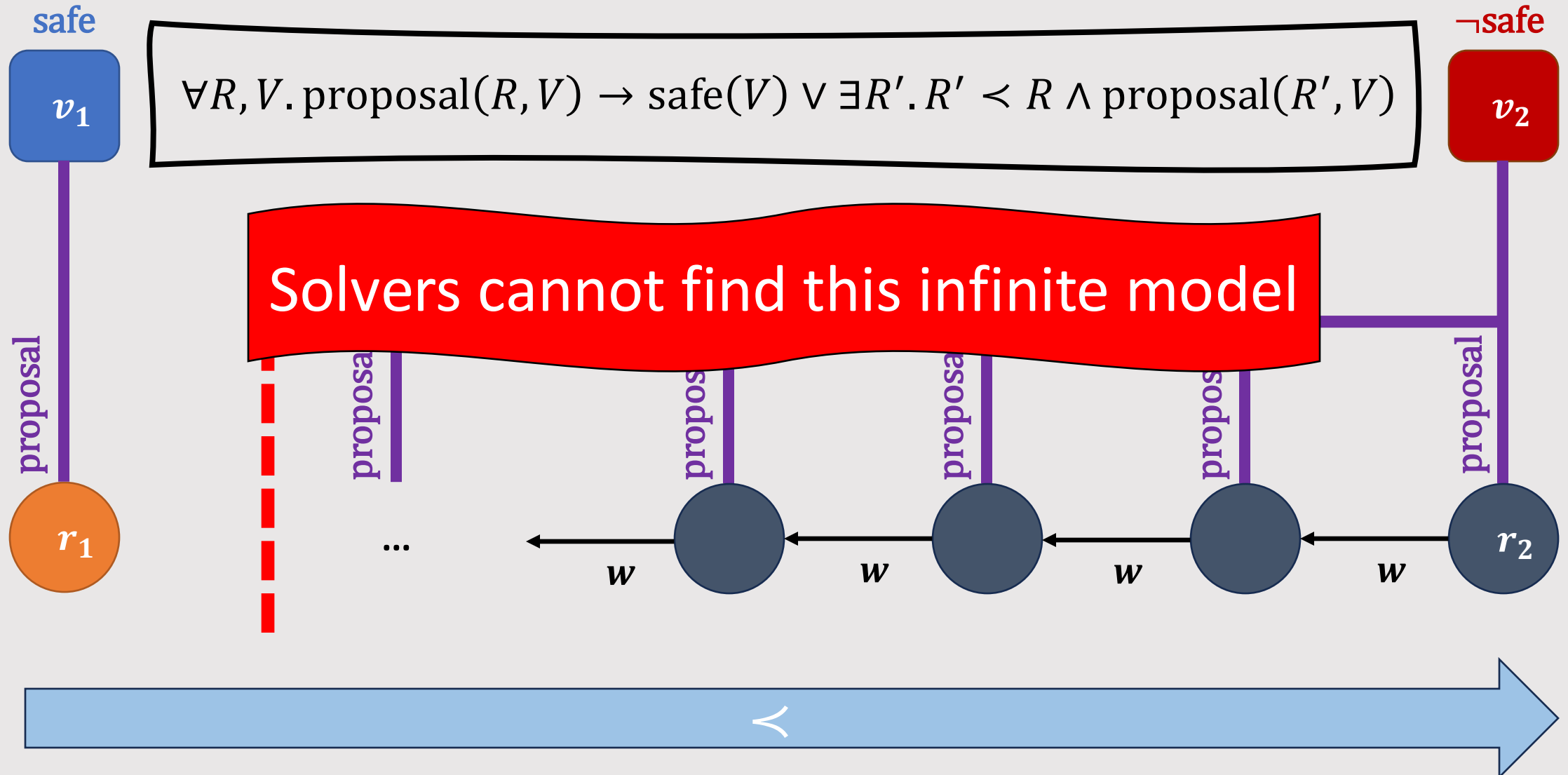
Example: Simplified VC from Paxos

- Existing tools are unable to verify
- Bad luck due to quantifier alternation or matching loops?
- **No**, there is an *infinite* counterexample



$\forall R, V. \dots \exists R'. R' < R \dots$

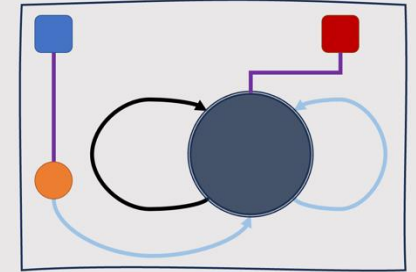
Infinite Counterexample



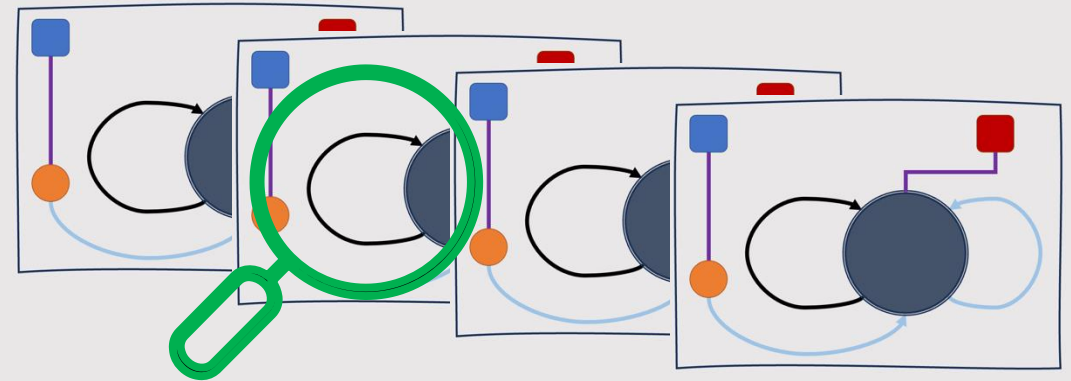
This Work



- Finite representation of infinite counter-models
 - Enables simple model-checking



- Efficient search procedure

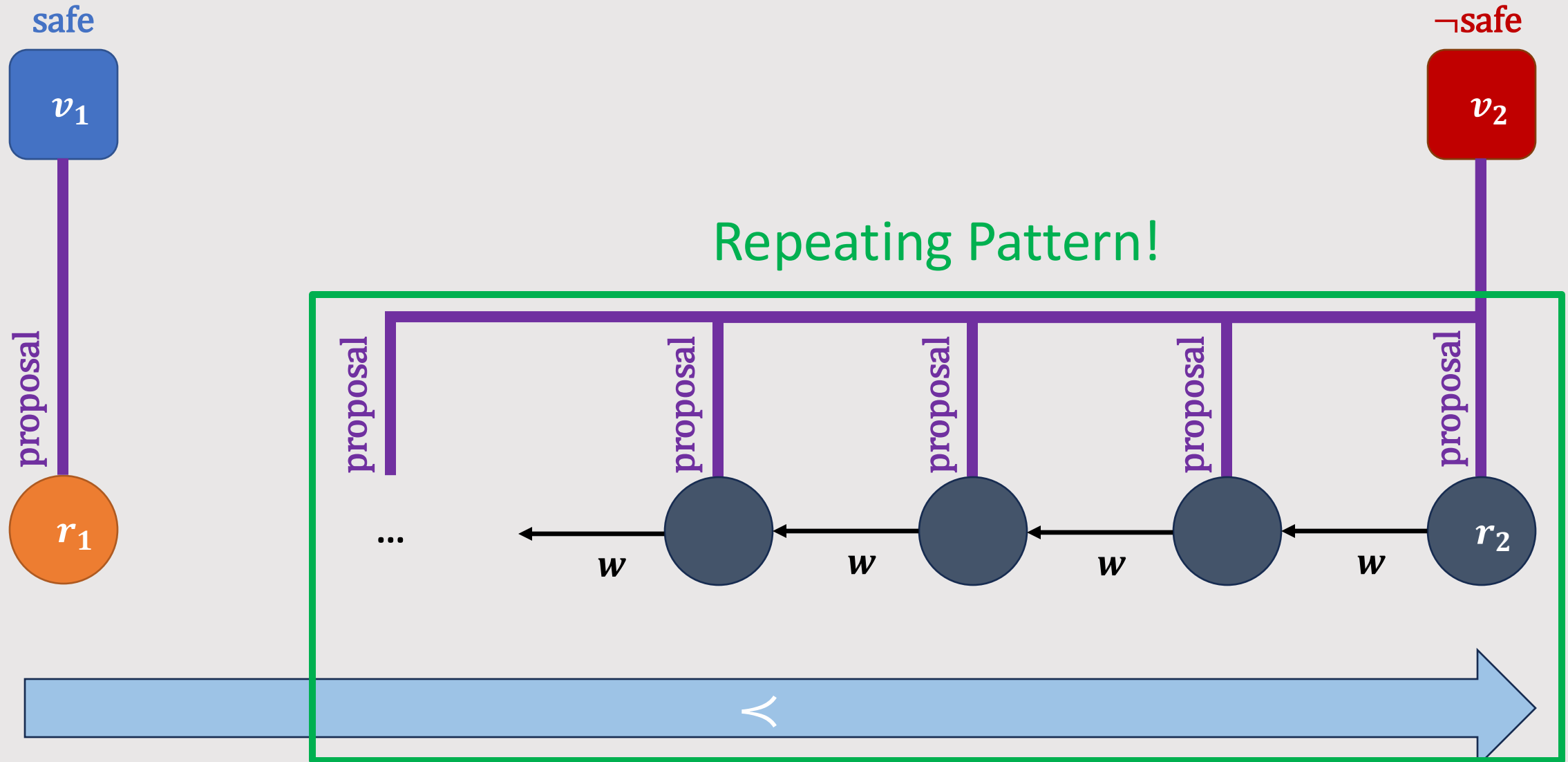


- Decidability result

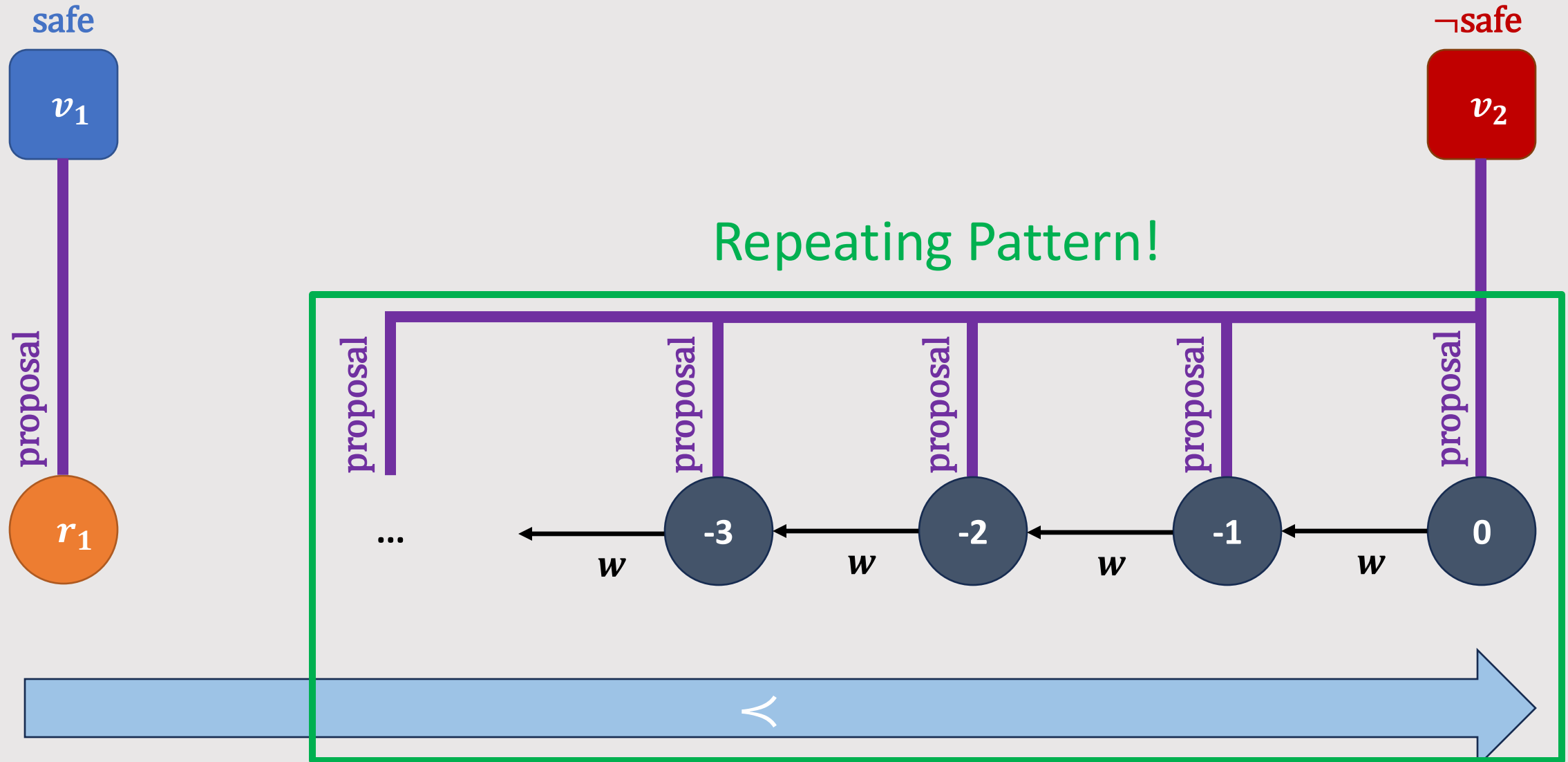
- Fragment of formulas, for which we can always find an infinite model, or disprove its existence



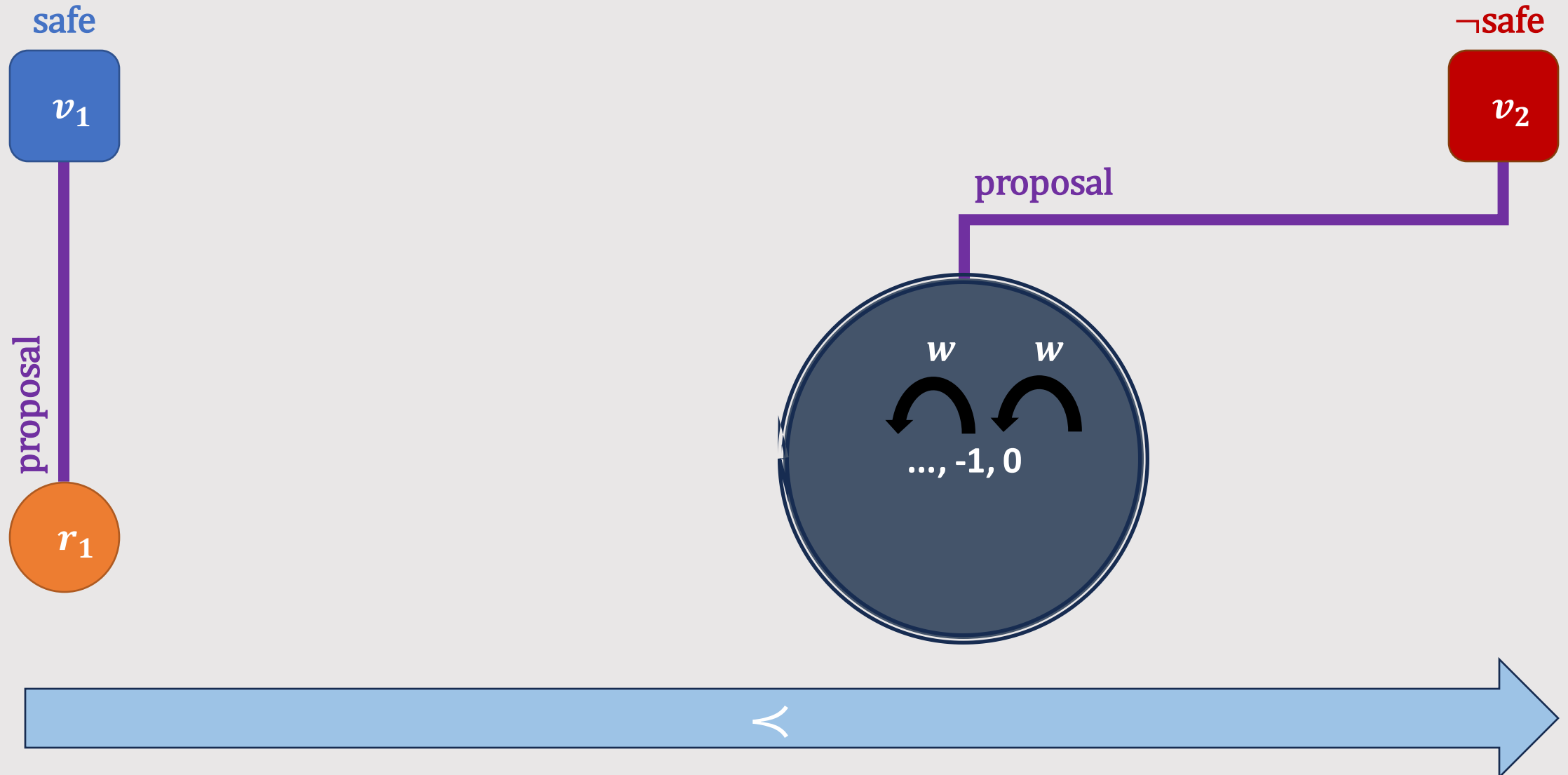
Infinite Counterexample



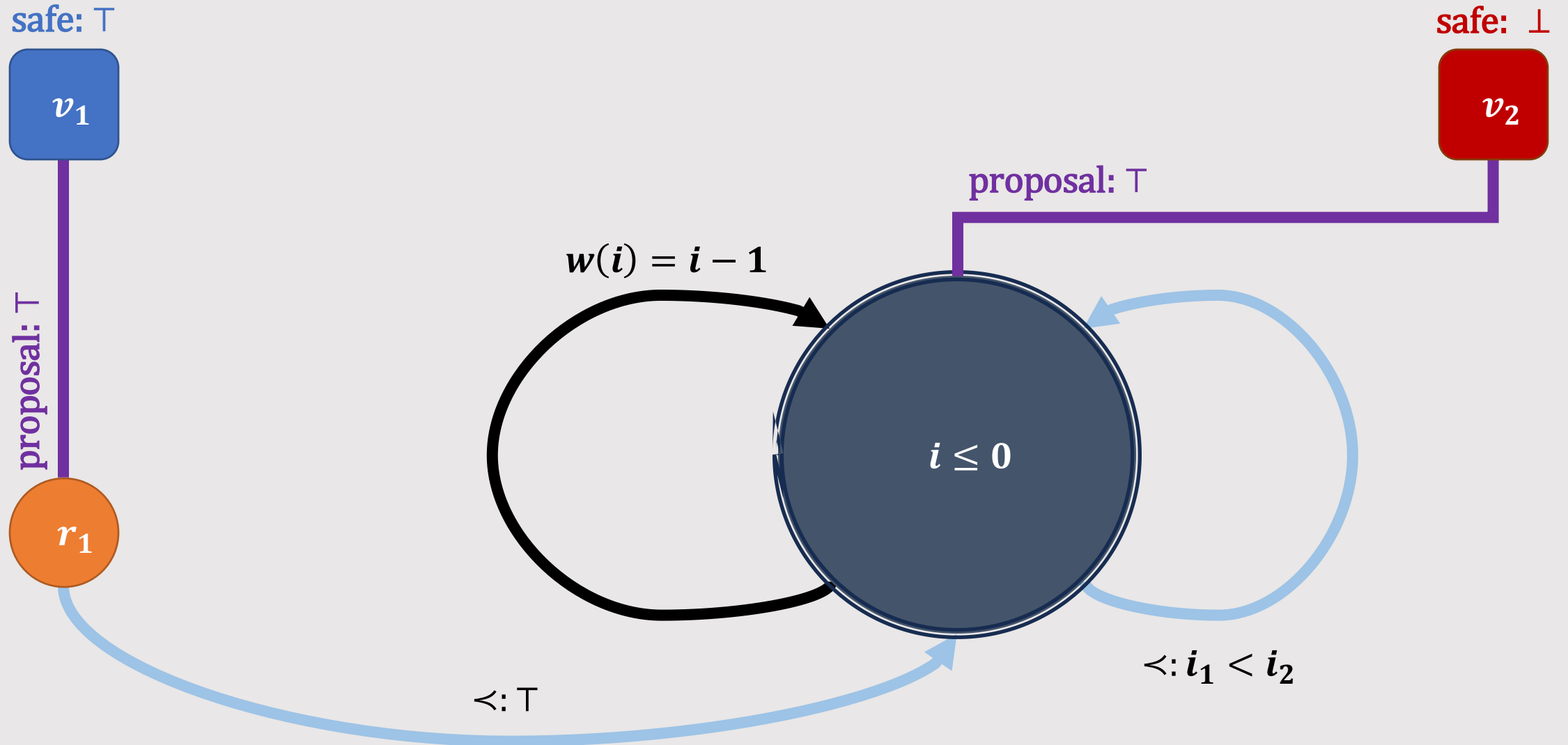
Infinite Counterexample



Infinite Counterexample



Symbolic Models



Symbolic Models

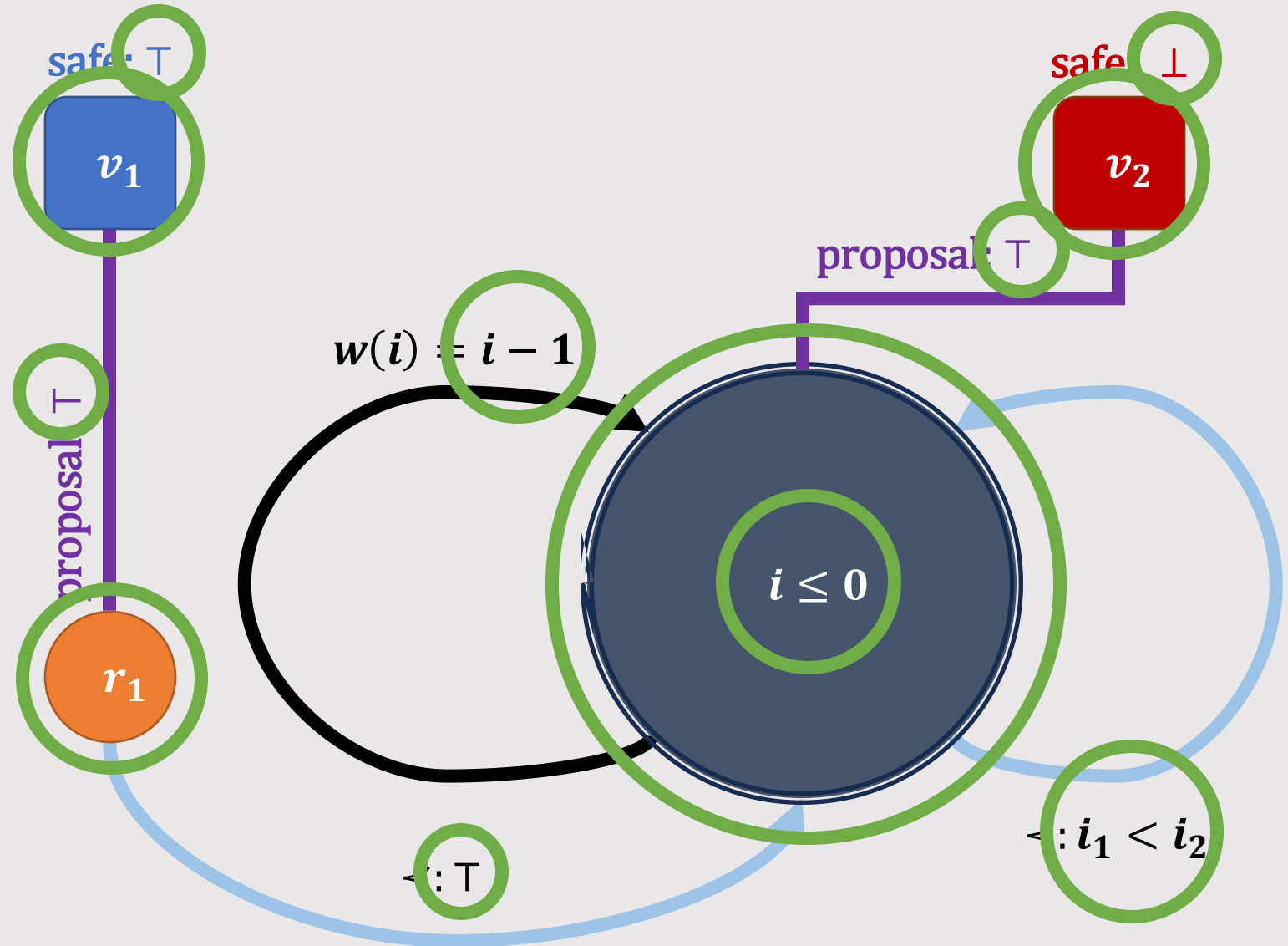
Symbolic

Linear Integer Arithmetic

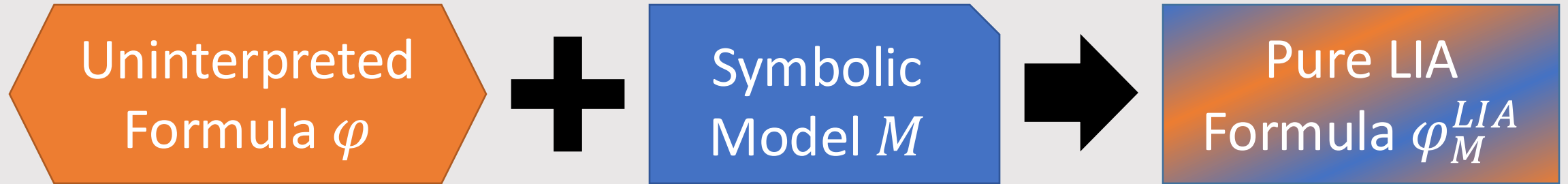
Bound Formulas

Function Terms

Relation Formulas



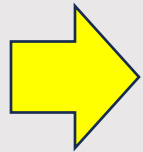
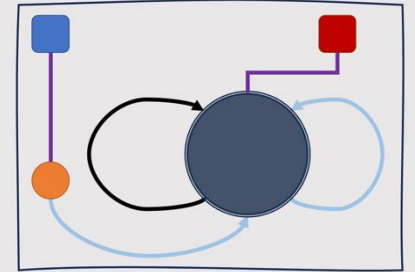
Symbolic Models – Model Checking



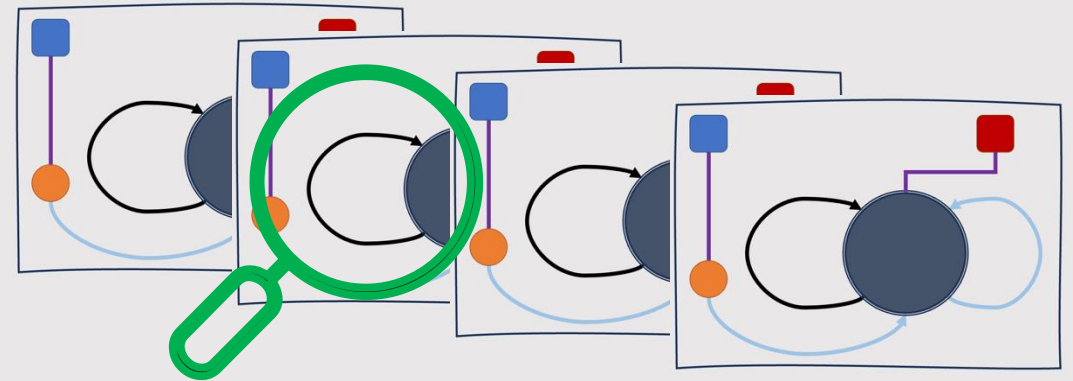
$$M \models \varphi \iff \models \varphi_M^{LIA}$$

This Work

- Finite representation of infinite counter-models
 - Enables simple model-checking



- Efficient search procedure



- Decidability result

- Fragment of formulas, for which we can always find an infinite model, or disprove its existence



Search Space: Templates

Template

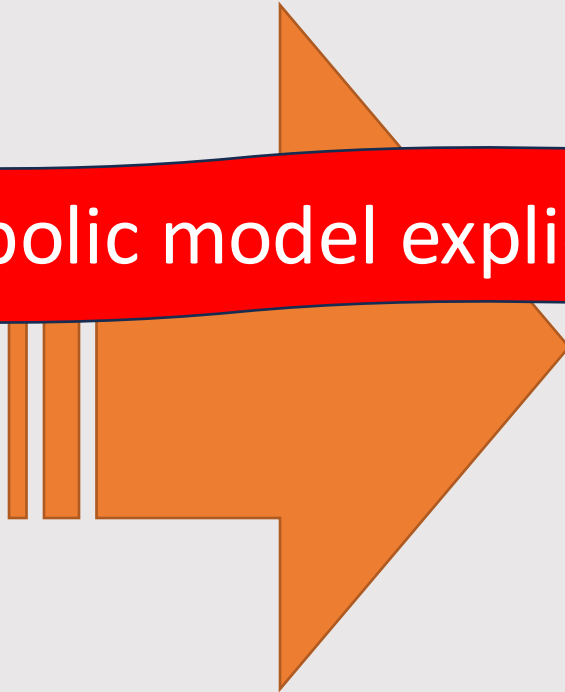
{ Symbolic
Domains }

{ Function
Terms }

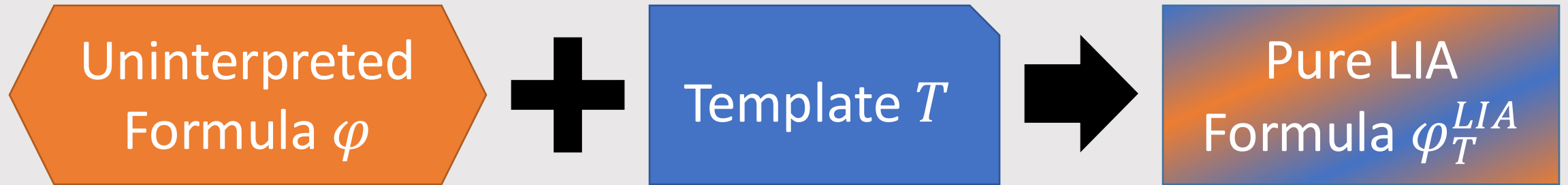
{ Relation
Formulas }

Checking every symbolic model explicitly is impractical

Symbolic
Models



Symbolic² Search Procedure



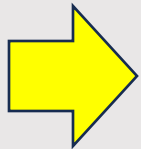
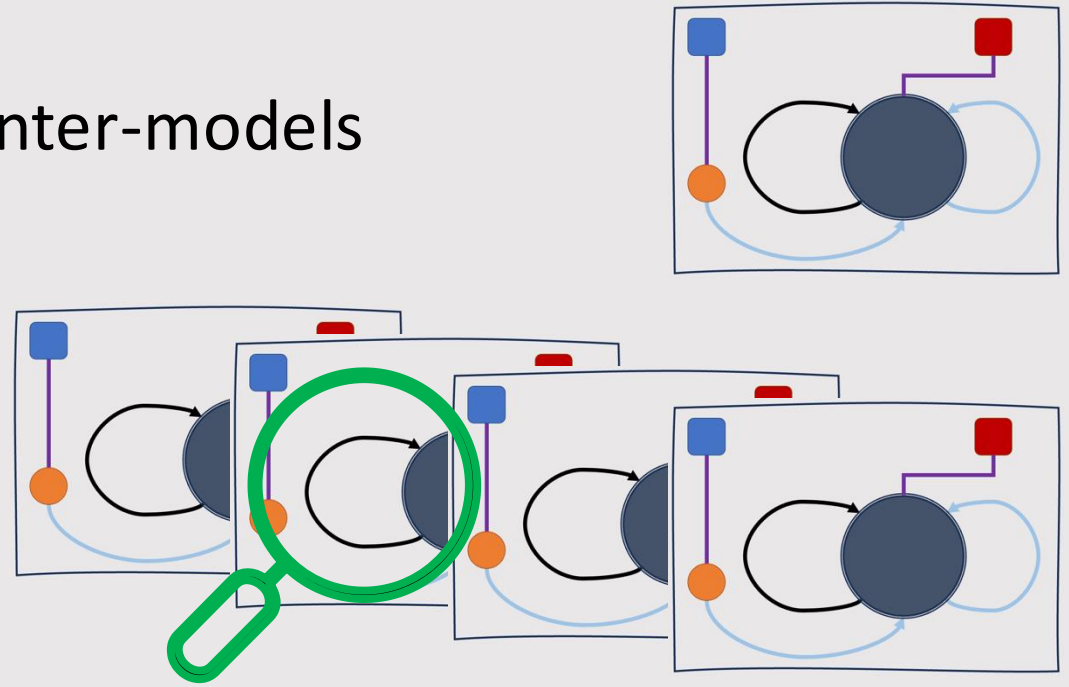
$$M_v \in T \models \varphi \quad \Leftrightarrow \quad v \models \varphi_T^{LIA}$$

Symbolic² Search Procedure

- Which formulas and terms to consider?
- Which symbolic domains to check?
- Are we guaranteed to have a satisfying symbolic model?

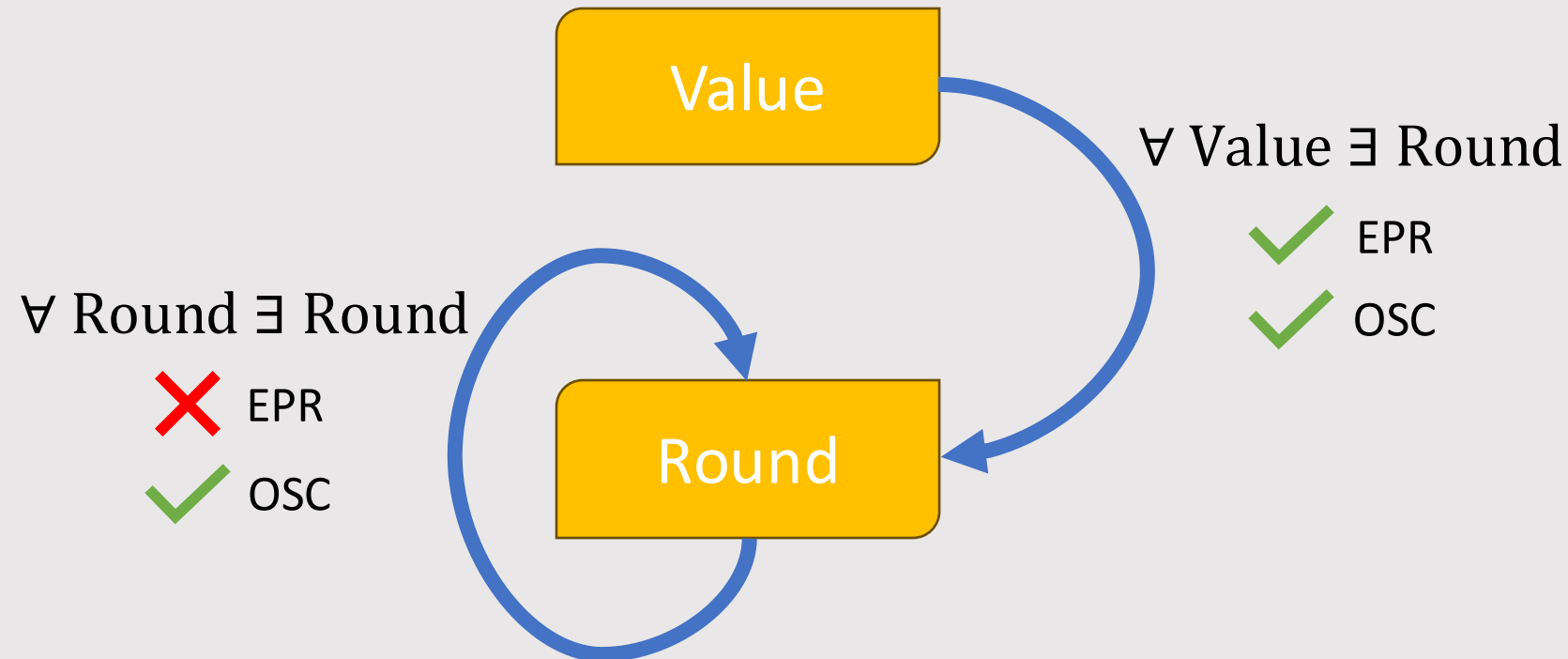
This Work

- Finite representation of infinite counter-models
 - Enables simple model-checking
- Efficient search procedure
- Decidability result
 - Fragment of formulas, for which we can always find an infinite model, or disprove its existence



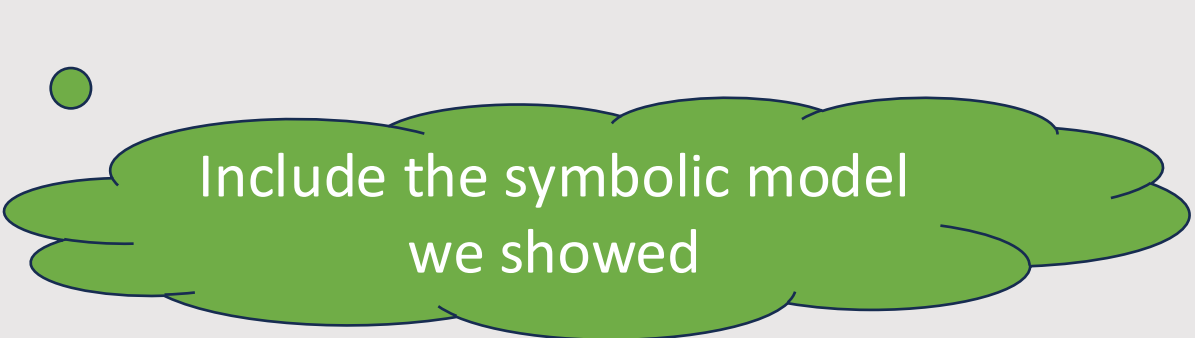
OSC – New Decidable Fragment

- New fragment of FOL, “*Ordered Self-Cycle*” (*OSC*)
 - *Paxos Simplified VC in OSC*
- OSC extends Effectively PRpositional (EPR) fragment



OSC – New Decidable Fragment

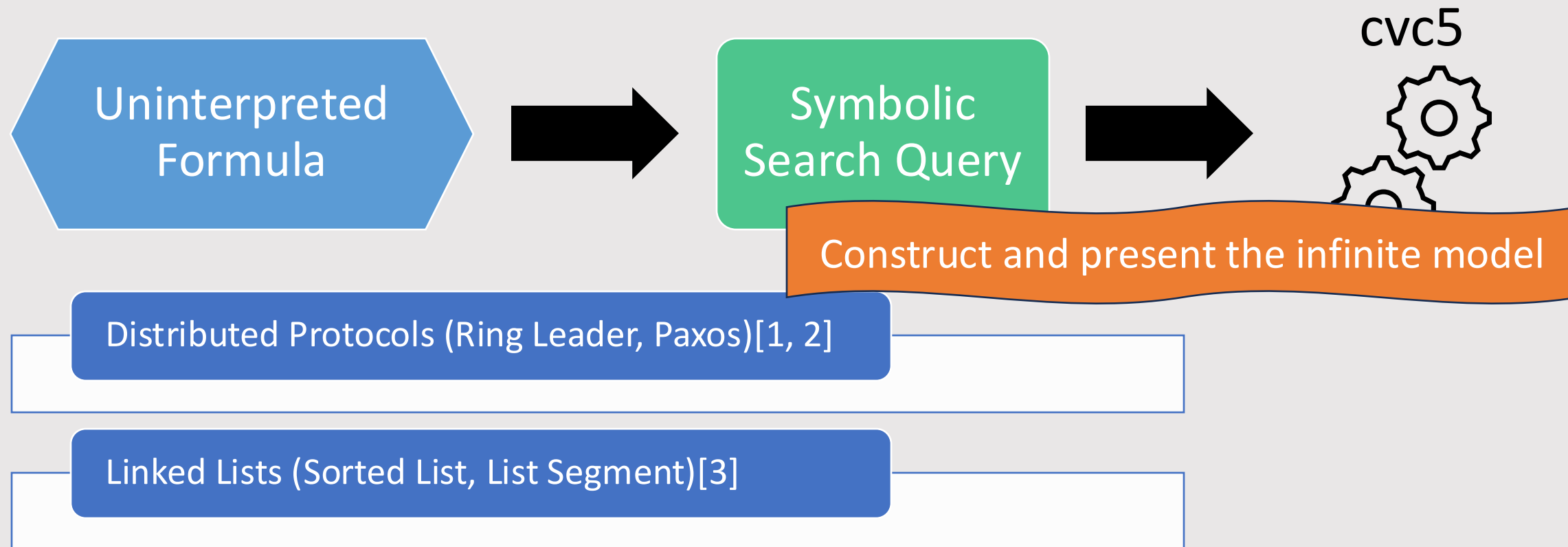
- New fragment of FOL, “*Ordered Self-Cycle*” (OSC)
 - *Paxos Simplified VC in OSC*
- OSC extends Effectively PRpositional (EPR) fragment
- Every satisfiable formula in OSC has a symbolic model
- “Small symbolic model property”
 - Bounding size of candidate symbolic models
 - Fixed sets of possible relation & bound formulas and function terms



Include the symbolic model
we showed



Evaluation – FEST (Find and Evaluate Symbolic structures via Templates)



[1] “Towards an Automatic Proof of Lamport's Paxos”. Goel et al, FMCAD 2021

[2] “Ivy: Safety Verification by Interactive Generalization”. Padon et al, PLDI 2016

[3] “Foundations for Natural Proofs and Quantifier Instantiation”. Löding et al, POPL 2018



Evaluation

Example	Infinite Sort Size regular/summary	Other Sorts Sizes	Time (s)
Echo Machine	1/1	2	< 1
Voting Protocol	2/1	3, 2, 2	~ 45
Simple Paxos	2/1	2, 2, 2	~ 12
Implicit Paxos	2/1	3, 2, 2	~ 133
Paxos	2/1	3, 3, 3	~ 80
Flexible Paxos	2/1	3, 3, 1, 1	~ 28
Ring Leader	0/1	-	< 1
Line Leader	2/1	-	~ 2

Example	Infinite Sort Size regular/summary	Time (s)
List Length	1/1	< 1
Seg. Const	1/1	< 1
Seg. Var	1/1	< 1
Seg. Order	2/1	< 1
Seg. Reverse	1/1	< 1
DL	1/1	< 1
DL Length	1/1	< 1
DL Seg.	1/1	< 1
Reverse List	2/1	< 1
Sorted Length	1/1	< 1
Sorted	1/1	< 1
Sorted Seg.	1/1	< 1
Sorted Max	1/1	< 1

Conclusion

- Symbolic Models and model-checking
 - Using LIA as an underlining language
- Templates and Symbolic² Search
 - Encoding infinitely many models in a single query
- Decidable Fragment OSC
 - Superset of EPR
- Future Work
 - More application domains
 - Other underlining theories
 - “Larger” infinite models

