IOWA STATE UNIVERSITY

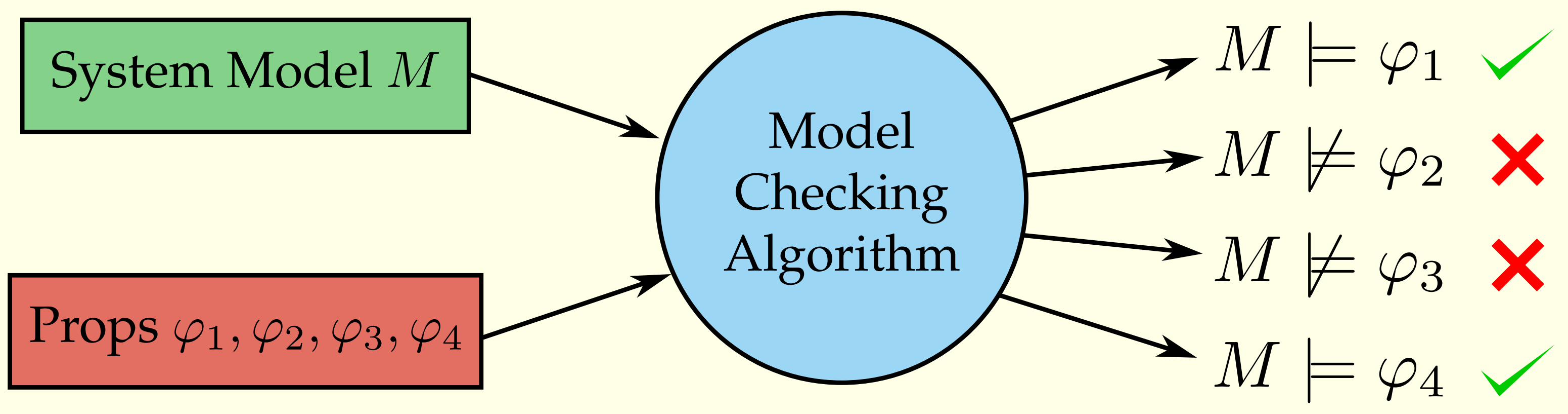
Scalable Verification of Designs with Multiple Properties

Rohit Dureja and Kristin Yvonne Rozier

Laboratory for Temporal Logic

1. Motivation

- The formal verification of complex industrial designs often entails checking a large number of properties.
  1. **Equivalence checking** compares pairwise equality of each design output: distinct property per output,
  2. **Functional verification** checks low-level assertions to high-level encompassing properties, and
  3. **Design-space exploration** via model checking verifies properties against competing system designs.


System Model M
Props $\varphi_1, \varphi_2, \varphi_3, \varphi_4$
Model Checking Algorithm
$M \models \varphi_1$ ✓
$M \not\models \varphi_2$ ✗
$M \not\models \varphi_3$ ✗
$M \models \varphi_4$ ✓

- Most research and development efforts address the problem of single-property verification, multiple properties are verified concurrently, or one-at-a-time.

Possible inter-property relationships, and shared sub-problems are typically ignored.
Opportunity to save verification resource.

2. Multiple Property Verification

- Develop efficient and scalable techniques for automatic verification of multiple properties.
  1. **Inter-property relationships** – utilize logical dependencies to minimize model-checking runs.
  2. **Information reuse** – learned state-space information is reused across various property verification tasks.
  3. **Improved orchestration** – properties with nearly identical cone-of-influence are verified concurrently.

3. Inter-Property Relationships

- Proprocess the set of properties to find pairwise logical dependencies; LTL satisfiability checking.

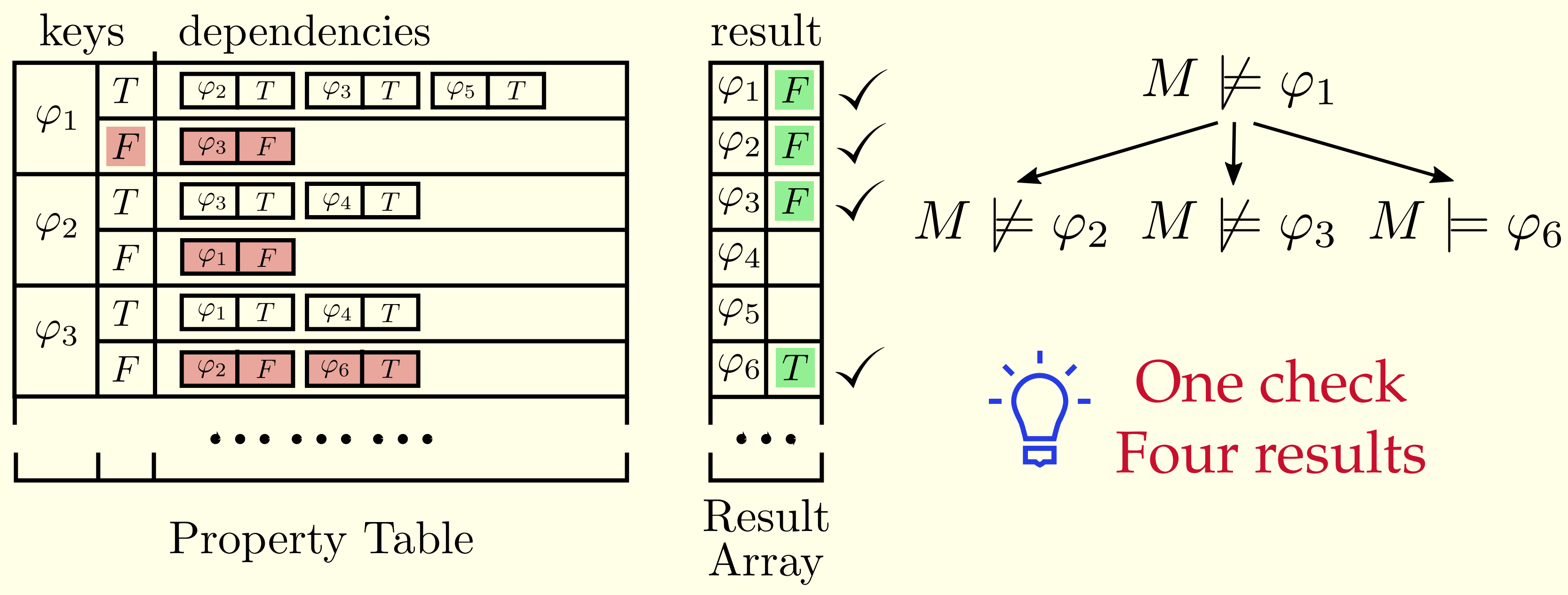
$\varphi_1 = \Box p$ $\varphi_2 = \Box(p \land q)$ $\varphi_3 = \Box(p \lor q)$

$M \models \varphi_2$ then $M \models \varphi_1$

φ_1 and φ_2 are dependent

$M \models \varphi_2$ then $M \models \varphi_3$

φ_2 and φ_3 are dependent



keys		dependencies
φ_1	T	φ_2 T, φ_3 T, φ_5 T
	F	φ_3 F
φ_2	T	φ_3 T, φ_4 T
	F	φ_1 F
φ_3	T	φ_1 T, φ_4 T
	F	φ_2 F, φ_6 T
		• • • • • • • • •

Property Table

result		
φ_1	F	✓
φ_2	F	✓
φ_3	F	✓
φ_4		
φ_5		
φ_6	T	✓
• • •		

Result Array

$M \not\models \varphi_1$
$M \not\models \varphi_2$
$M \not\models \varphi_3$
$M \models \varphi_6$

One check
Four results

Few minutes to find dependencies between properties, and <10% properties checked for each design.

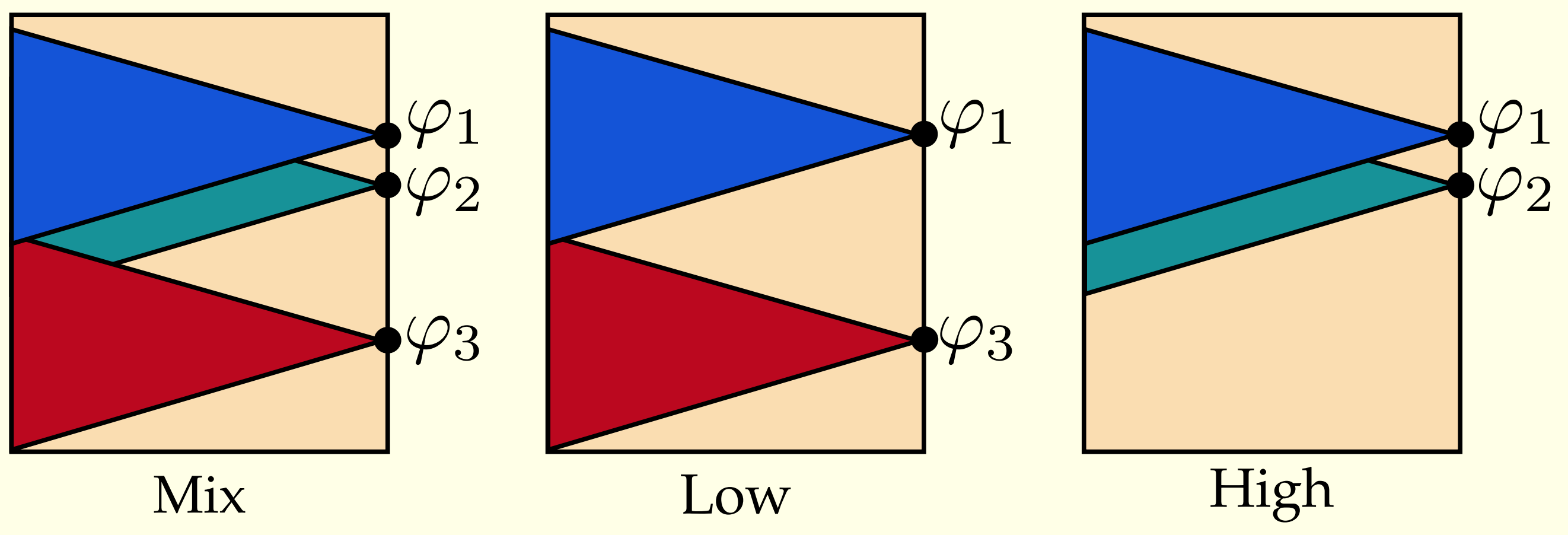
4. Information Reuse

- Sequentially check properties by reusing information; state approximations, counterexamples, and invariants.
- Stored information is repaired before reuse; add "just enough" extra information to enable reuse.

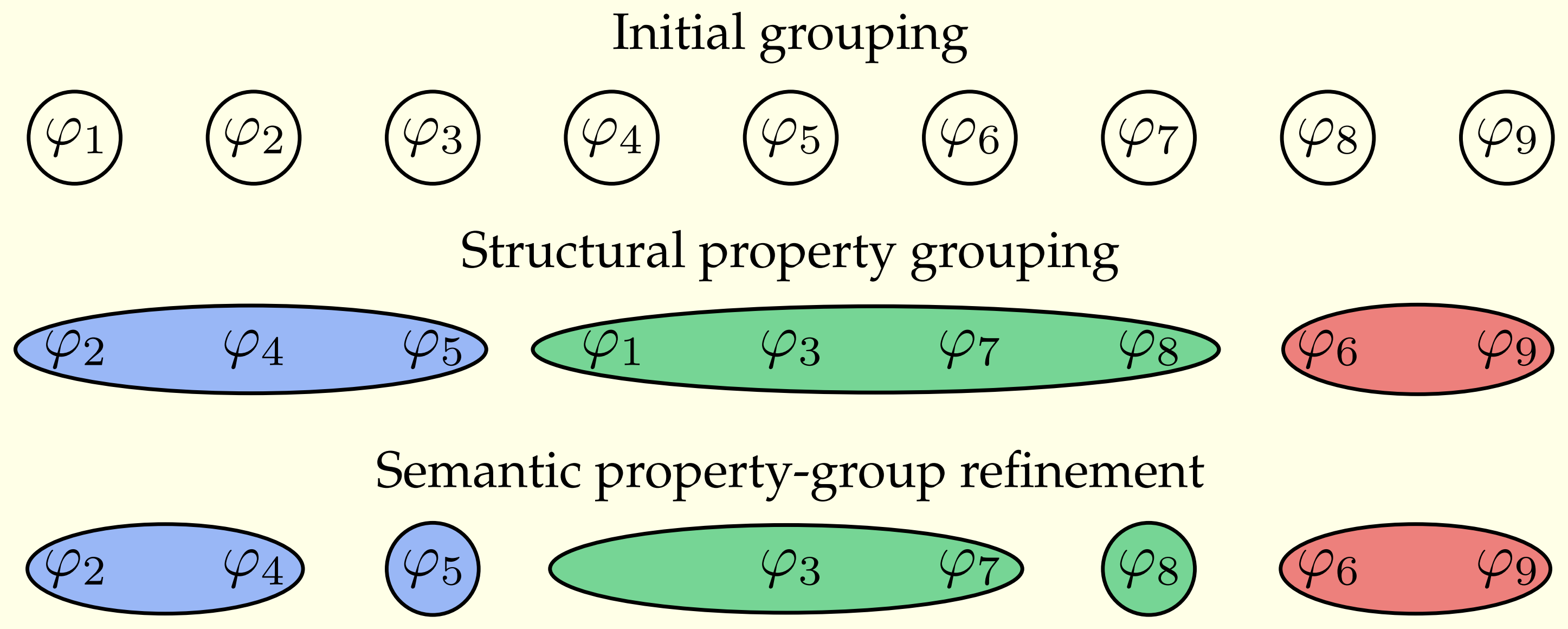
Adapt IC3/PDR for multi-property model checking by reusing frames to enable 4.5× faster verification.

5. Improved Orchestration

- Property grouping saves substantial verification resource by concurrent verification of high-affinity properties.



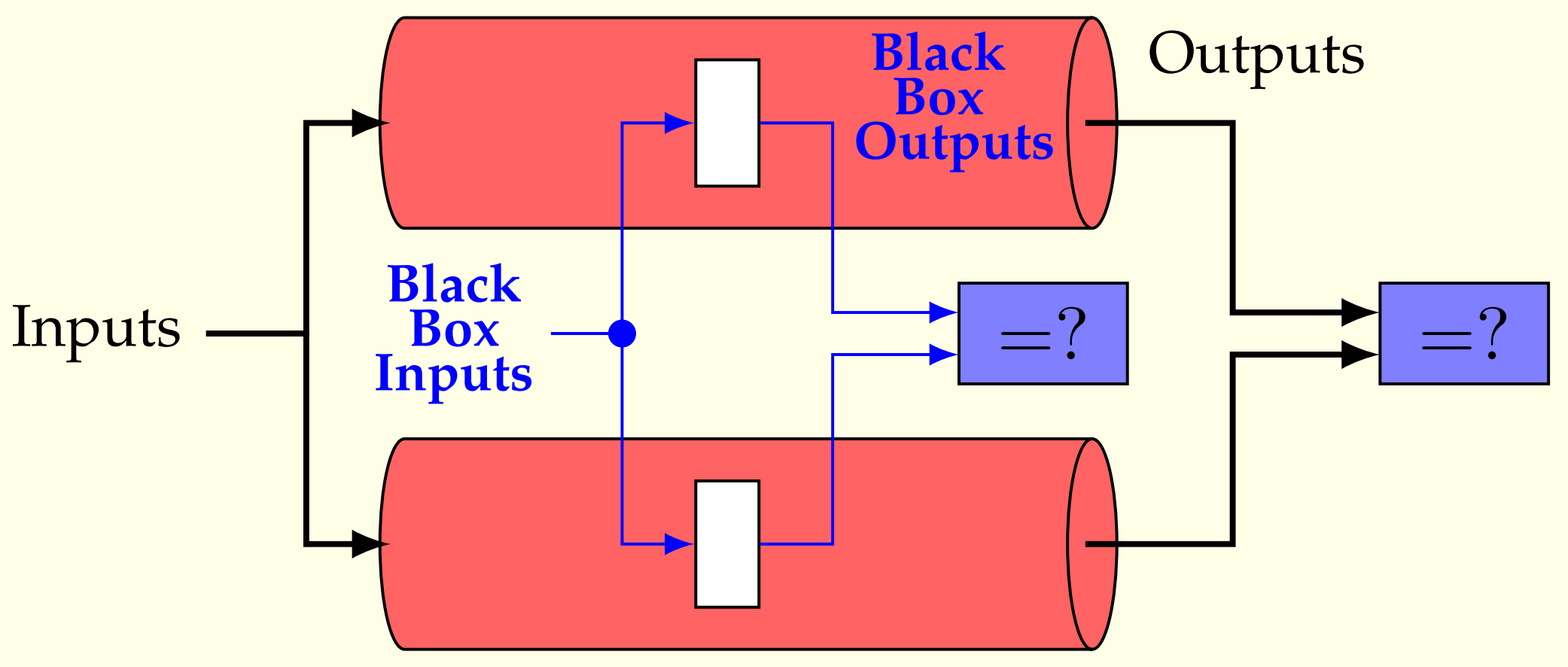
- Partition properties into provably high-affinity groups based on cone-of-influence (COI); ∼linear runtime.
- Two-level orchestration; structural property grouping followed by semantic property-group refinement.


Initial grouping
φ_1 φ_2 φ_3 φ_4 φ_5 φ_6 φ_7 φ_8 φ_9
Structural property grouping
φ_2 φ_4 φ_5
φ_1 φ_3 φ_7 φ_8
φ_6 φ_9
Semantic property-group refinement
φ_2 φ_4
φ_5
φ_3 φ_7
φ_8
φ_6 φ_9

Improved multiple property verification offering 4.8× end-to-end speedup; advance state-of-the-art localization.

6. Ongoing and Future Work

- **When to use structural vs. semantic grouping?** Difficult to discern what COI subset is relevant to what property.



- Sequential equivalence checking (SEC) is a prevalent multiple property verification application; several miters.
- Improve SEC by intelligently discharging non-inductive provable miters by improved property orchestration.

1. "FuseIC3: An Algorithm for Checking Large Design Spaces," in FMCAD, 2017.
2. "More Scalable LTL Model Checking via Discovering Design-Space Dependencies," in TACAS, 2018.
3. "Boosting Verification Scalability via Structural Grouping and Semantic Partitioning of Properties," in FMCAD, 2019.

Supported by NSF CAREER Award CNS-1664356

fmcad^{19}

http://laboratory.temporallogic.org