

Dubious Witnesses and Spurious Counterexamples

UK Model Checking Days

York, September 2005

Maria Sorea

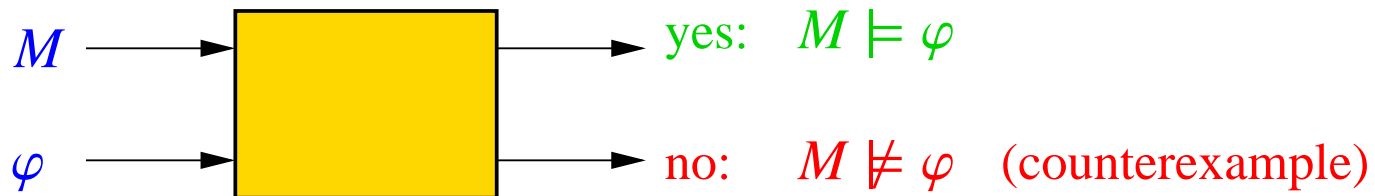
`msorea@cs.man.ac.uk`

`http://www.cs.man.ac.uk/~msorea/.`

University of Manchester

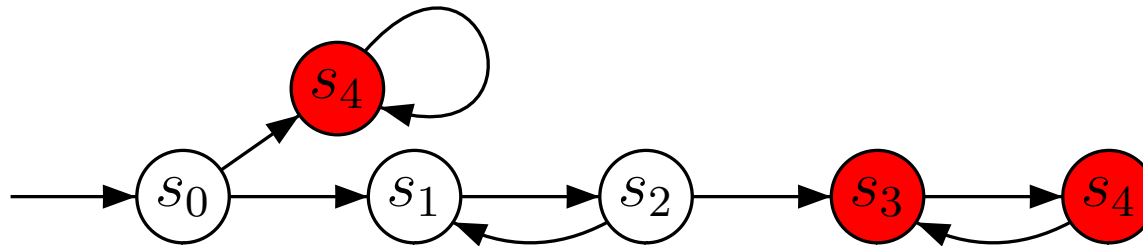
Model Checking

- Kripke structure: $M = \langle \mathbf{AP}, \mathbf{S}, \mathbf{N} \rangle$
- CTL formula: φ
- Set of initial states: $I \subseteq \mathbf{S}$
- Model-checking problem: $M, I \stackrel{?}{\models} \varphi$



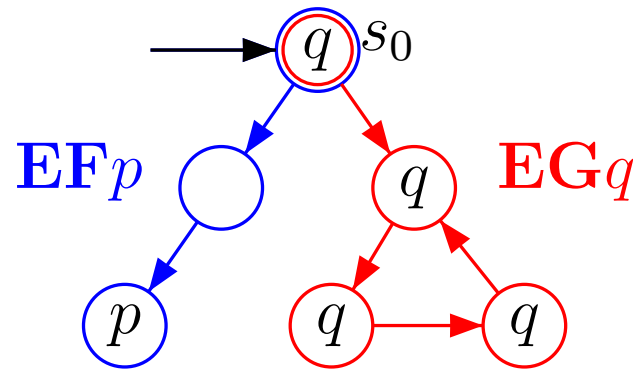
Linear Witnesses / Counterexamples

Consider the CTL property $\mathbf{EG}p$, and the Kripke structure M



- Assume: p does not hold in the *red* states
- Witness for $M, s_0 \models \mathbf{EG}p$: infinite trace (s_0, s_1, s_2, s_1)
- Counterexample for $M, s_0 \models \mathbf{AF}\neg p$
- Only for $\text{ACTL} \cap \text{LTL}$ formulas

Tree-Like Witnesses / Counterexamples



Witness for $M, s_0 \models \mathbf{EF}p \wedge \mathbf{EG}q$

Counterexample for $M, s_0 \models \mathbf{AG}\neg p \vee \mathbf{AF}\neg q$

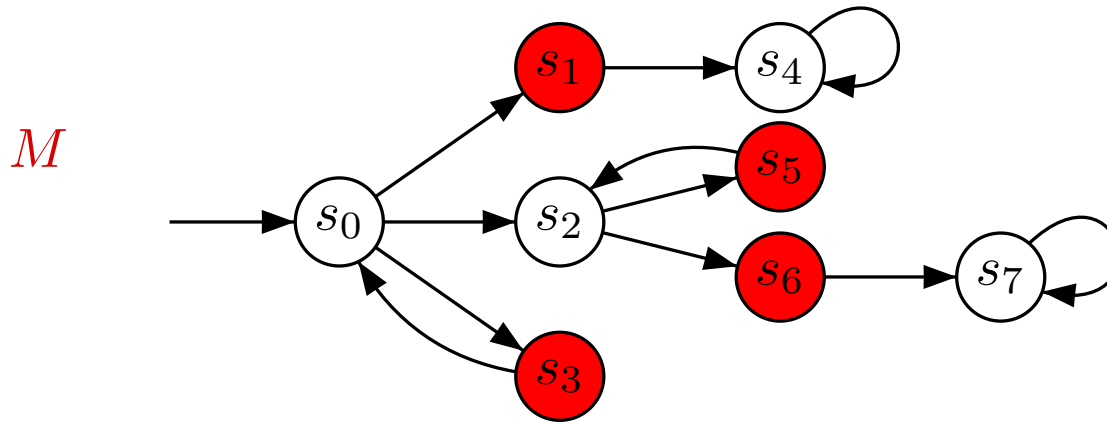
- There is a finite path to a p state
- There is an infinite path along which q is always true

Only for ACTL formulas

Symbolic Counterexample/Witnesses for CTL

[Shankar, Sorea 2003]

EG p . . . there is a path in M on which p globally holds



Assume p does not hold in the red states

Counterexample for the validity of **EG** p in M : $[X_0, X_1, X_2]$

- $X_0 = \{s_1, s_3, s_5, s_6\}$
- $X_1 = \{s_1, s_2, s_3, s_5, s_6\}$
- $X_2 = \{s_0, s_1, s_2, s_3, s_5, s_6\}$

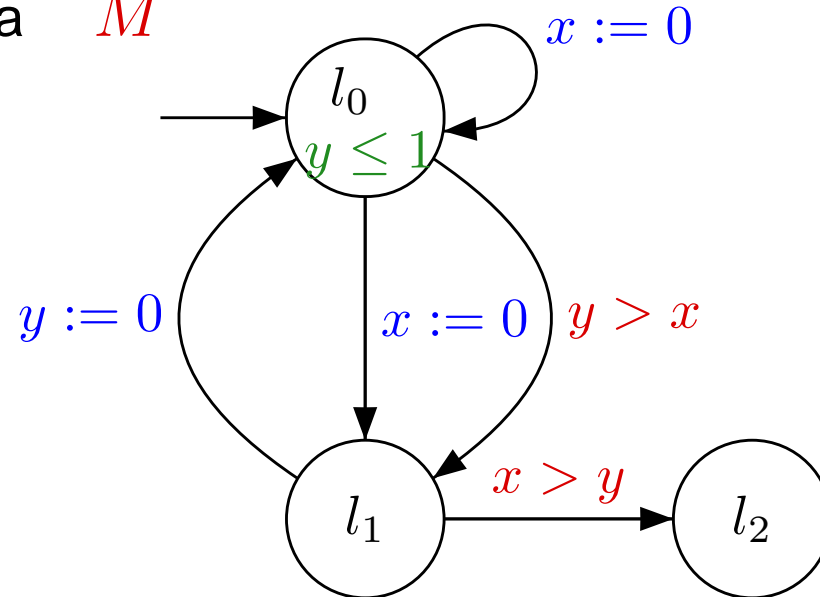
Witness for $M, s_0 \models \mathbf{AF}\neg p$

Benefits and Applications

- Positive and negative evidence for full CTL
- Symbolic form
- Explicit witnesses/counterexamples can be extracted from it
- Proof for the validity/falsity of a formula in a model
- Can be independently verified using a SAT solver
- Main applications: abstraction refinement and controller synthesis

Example: Model Checking Timed Systems

Models: Timed automata M



Specifications: CTL formulas $\varphi = \mathbf{AG}(\neg at_l_2)$

Model checking: $M \models \varphi$

Our Approach – Abstraction

Compute finite-state abstractions of M

Over- und underapproximations of M since we are interested in both \forall and \exists formulas

Underapproximation M^- has *less behavior* than M

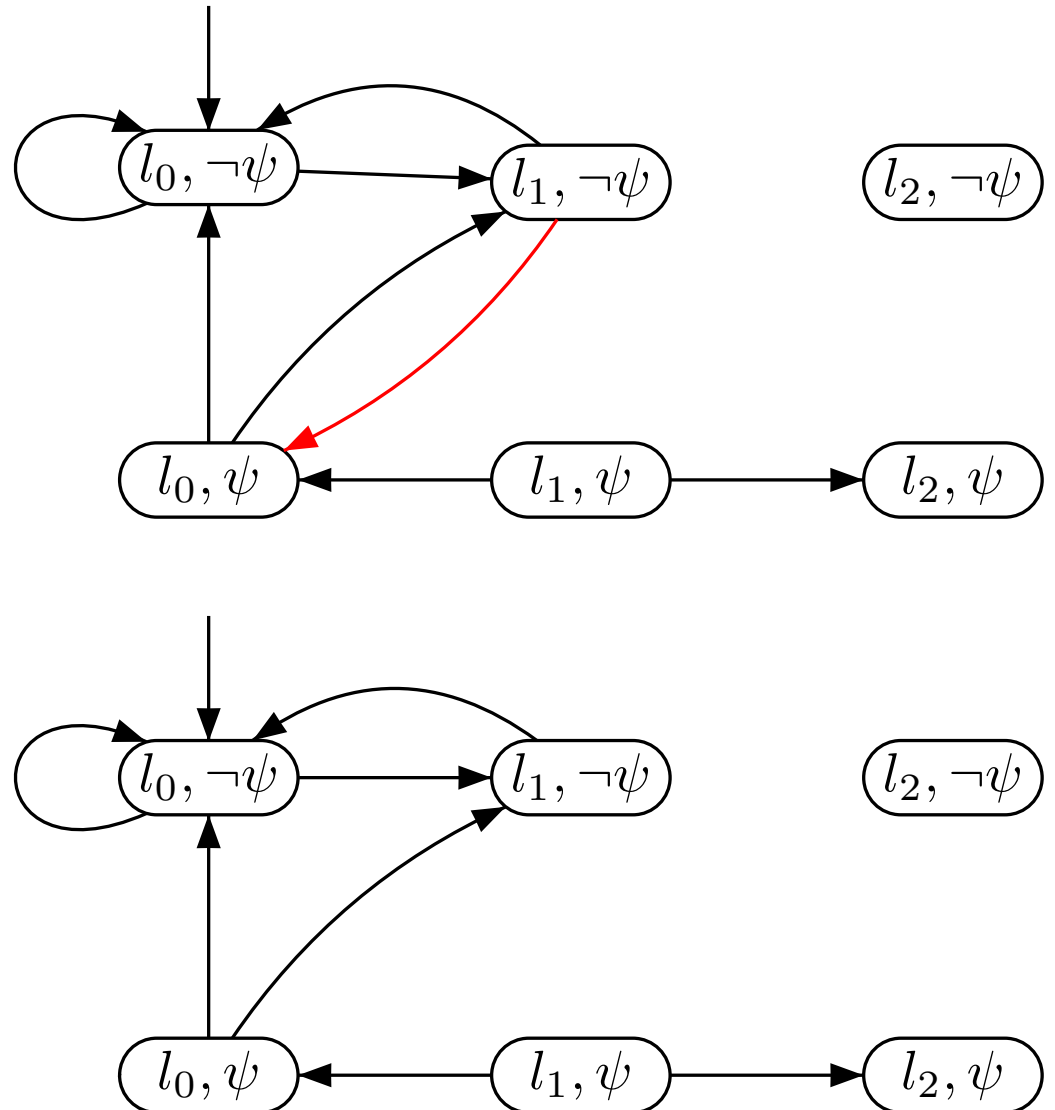
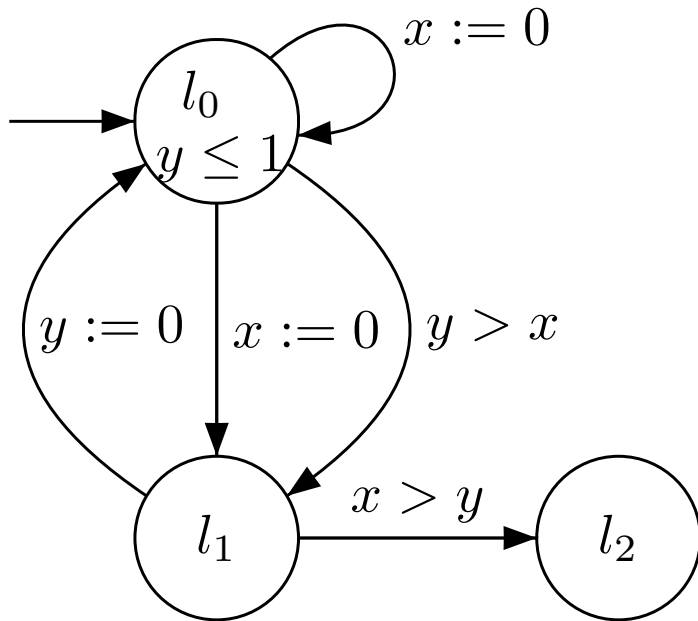
Overapproximation M^+ has *more behavior* than M

Dubious Witnesses and Spurious Counterexamples

$M^+ \models \forall \varphi \Rightarrow M \models \forall \varphi$	$M^+ \not\models \forall \varphi \Rightarrow C + R$
$M^- \models \forall \varphi \Rightarrow W + R$	$M^- \not\models \forall \varphi \Rightarrow M \not\models \forall \varphi$
$M^- \models \exists \varphi \Rightarrow M \models \exists \varphi$	$M^- \not\models \exists \varphi \Rightarrow C + R$
$M^+ \models \exists \varphi \Rightarrow W + R$	$M^+ \not\models \exists \varphi \Rightarrow M \not\models \exists \varphi$

Over ($\exists\exists$)- vs. Underapproximation ($\forall\exists$)

$\Psi = \{\psi\}$, where $\psi \equiv x > y$



Refining Approximations

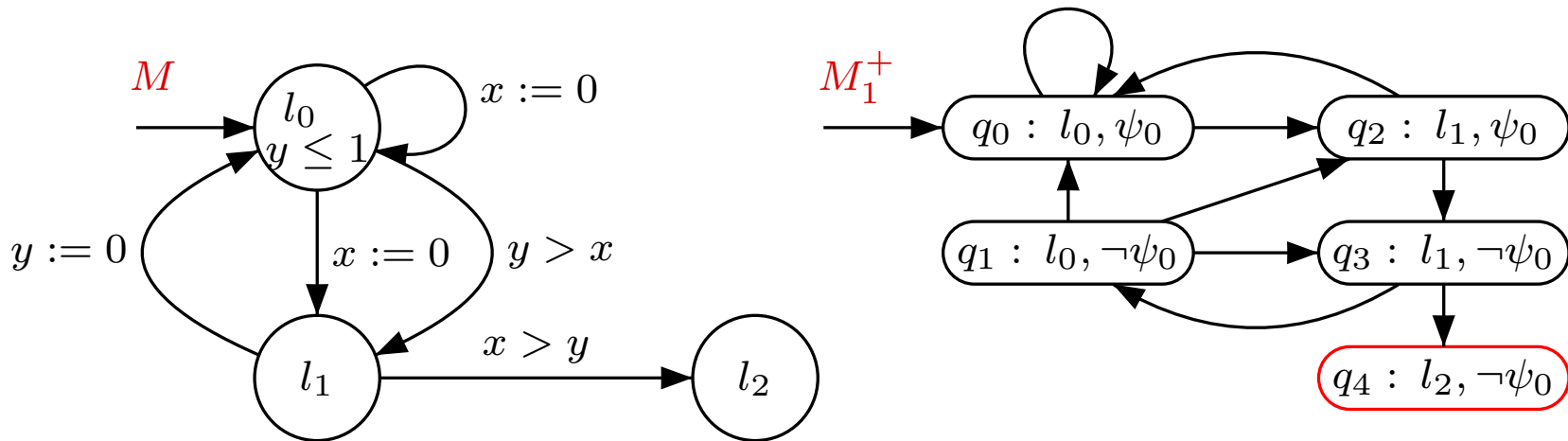
Refining Overapproximations.

- Universal formula invalid on the overapproximation can be satisfied on the system
- Goal: smaller overapproximation
- Choose new predicates to eliminate spurious counterexamples
- *preimage* computation

Refining Underapproximations.

- Universal formula valid on the underapproximation must not necessarily hold on the system
- Goal: larger underapproximation
- Choose new predicates to extend dubious witnesses
- *postcondition* computation

Example – Refining Overapproximations



Formula: $\varphi = \mathbf{AG}(\neg at_l_2)$

Initial approximation: M_1^+ with abstraction predicate $\psi_0 \equiv (x = 0)$

WMC: M_1^+, q_0, φ produces counterexample

$c = [X_0, X_1, X_2, X_3]$ with

$$X_0 = \{q_4\}, X_1 = \{q_3, q_4\},$$

$$X_2 = \{q_1, q_2, q_3, q_4\},$$

$$X_3 = \{q_0, q_1, q_2, q_3, q_4\}$$

Example – Refining Overapprox (Cont.)

Concretization of the abstract counterexample

$$\gamma(X_0) = (l_2, x > 0 \wedge y \geq 0)$$

$$\gamma(X_1) = (l_1, x > 0 \wedge y \geq 0) \cup (l_2, x > 0 \wedge y \geq 0)$$

$$\gamma(X_2) = (l_0, x > 0 \wedge y \geq 0) \cup (l_1, x \geq 0 \wedge y \geq 0) \cup (l_2, x > 0 \wedge y \geq 0)$$

$$\gamma(X_3) = (l_0, x \geq 0 \wedge y \geq 0) \cup (l_1, x \geq 0 \wedge y \geq 0) \cup (l_2, x > 0 \wedge y \geq 0)$$

Exists concrete counterexample $[X_0^c, X_1^c, X_2^c, X_3^c]$ with $X_i^c \subseteq_\nu \gamma(X_i)$?

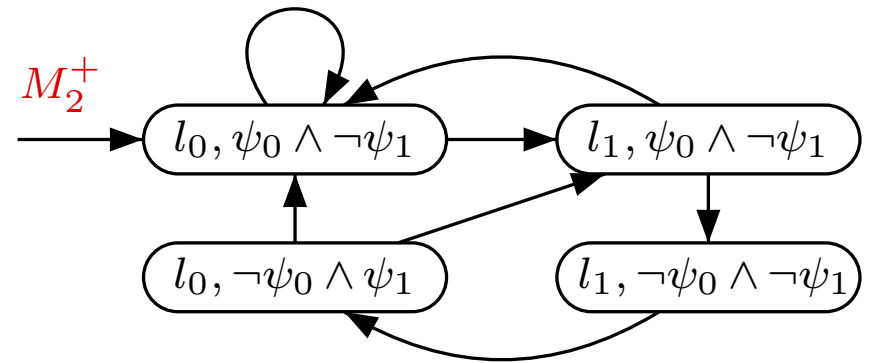
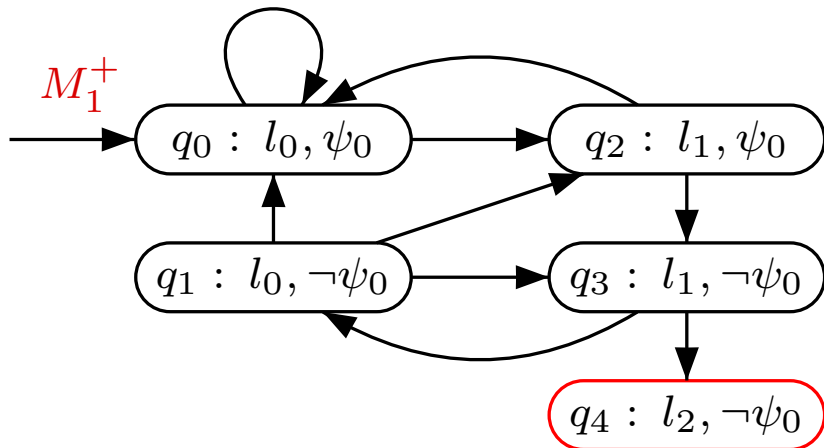
Is the following formula valid?

$$\phi = \exists X_0^c \subseteq_\nu \gamma(X_0), \dots, X_3^c \subseteq_\nu \gamma(X_3). (X_0^c \Rightarrow \text{at_}l_2) \wedge \bigwedge_{i=0}^2 (X_{i+1}^c \Rightarrow (X_i^c \vee \text{pre}(\mathbf{N})(X_i^c)))$$

No: $X_2^c \not\subseteq (X_1^c \cup \text{pre}(\mathbf{N})(X_1^c))$

Since: $X_1^c \cup \text{pre}(\mathbf{N})(X_1^c) = (l_2, x > 0 \wedge y \geq 0) \cup (l_1, x > y \geq 0) \not\subseteq l_0$

Example – Refining Overapprox (Cont.)



Choose new predicates to eliminate the transition from q_3 to q_4

preimage computation on $\gamma(q_4) = X_0^c = (l_2, x > 0 \wedge y \geq 0)$

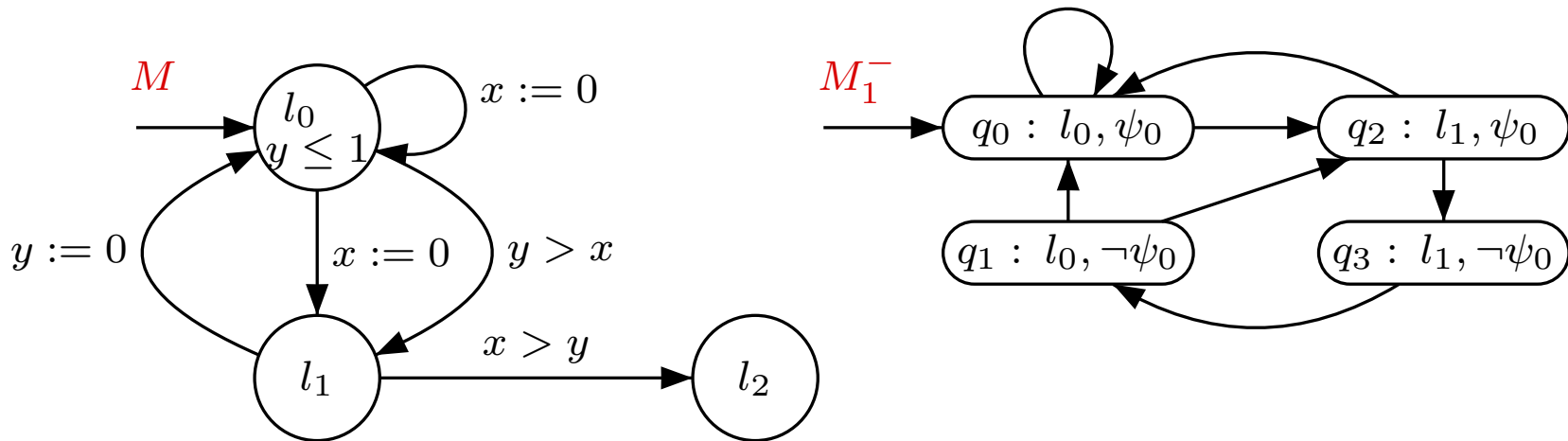
$pre(\mathbf{N})(X_0^c) = (l_1, x > y)$

$x > y$ is the new predicate ψ_1

Refine existing approximation using ψ_1

Formula $\mathbf{AG}(\neg at_l_2)$ holds now on the refined approximation M_2^+

Example – Refining Underapproximations



Formula: $\varphi = \mathbf{AG}(\neg at_l_2)$

Initial approximation: M_1^- with abstraction predicate $\psi_0 \equiv (x = 0)$

WMC: M_1^-, q_0, φ produces abstract witness

$$w = \{q_0, q_1, q_2, q_3\}$$

Example – Refining Underapprox (Cont.)

Concretization:

$$w^c = \{(l_0, x \geq 0, y \geq 0), (l_1, x \geq 0, y \geq 0)\}$$

Here w^c is not a concrete witness since the formula

$$\phi = \exists X_0^c \subseteq_\nu \gamma(X_0). (X_0^c \Rightarrow at_l_2) \wedge X_0^c \Rightarrow \widetilde{pre}(\mathbf{N})(X_0^c)$$

is not valid!

Abstract witness is dubious!

Choose new predicates to extend the existing underapproximation

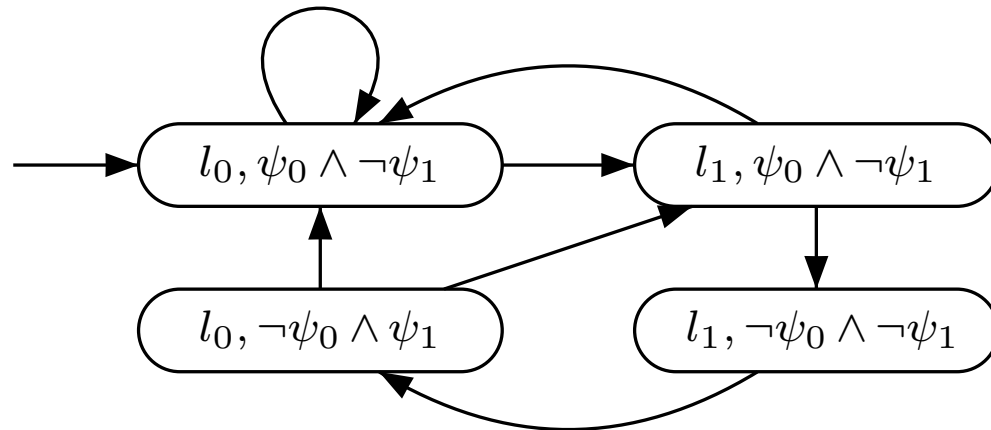
postcondition computation on $X_0^c \setminus \widetilde{pre}(\mathbf{N})(X_0^c)$

$$post(\mathbf{N})((l_1, x \geq 0, y \geq 0)) = (l_2, x > y)$$

$x > y$ is the new predicate ψ_1

Refine existing under-approximation using ψ_1

Example – Refining Underapprox (Cont.)



Abstract witness:

$$w = \{(l_0, \psi_0 \wedge \neg\psi_1), (l_0, \neg\psi_0 \wedge \psi_1), (l_1, \psi_0 \wedge \neg\psi_1), (l_1, \neg\psi_0 \wedge \neg\psi_1)\}$$

Concrete witness:

$$w^c = \{(l_0, x = 0, x \leq y), (l_0, x > 0, x > y), (l_1, x = 0, x \leq y), (l_1, x > 0, x > y)\}$$

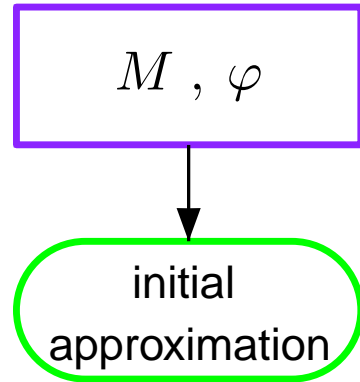
This is indeed a concrete witness!

Thus $M \models \mathbf{AG}(\neg at_l_2)$.

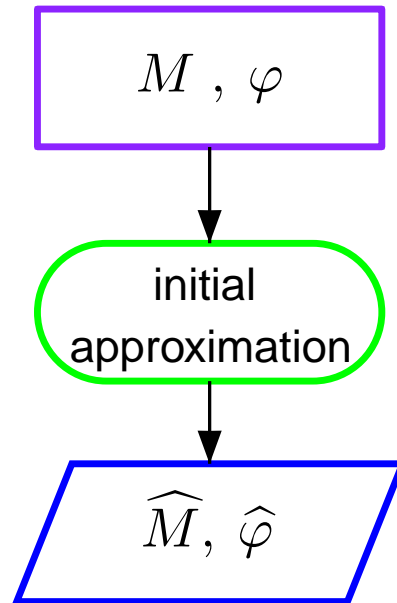
Lazy Approximation

M, φ

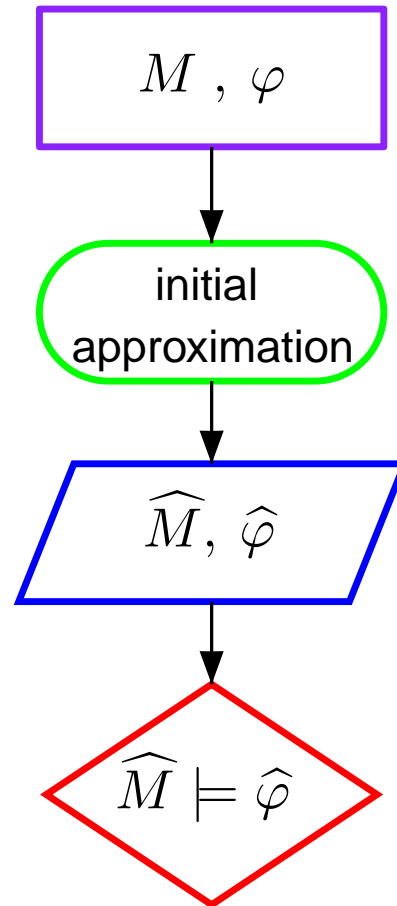
Lazy Approximation



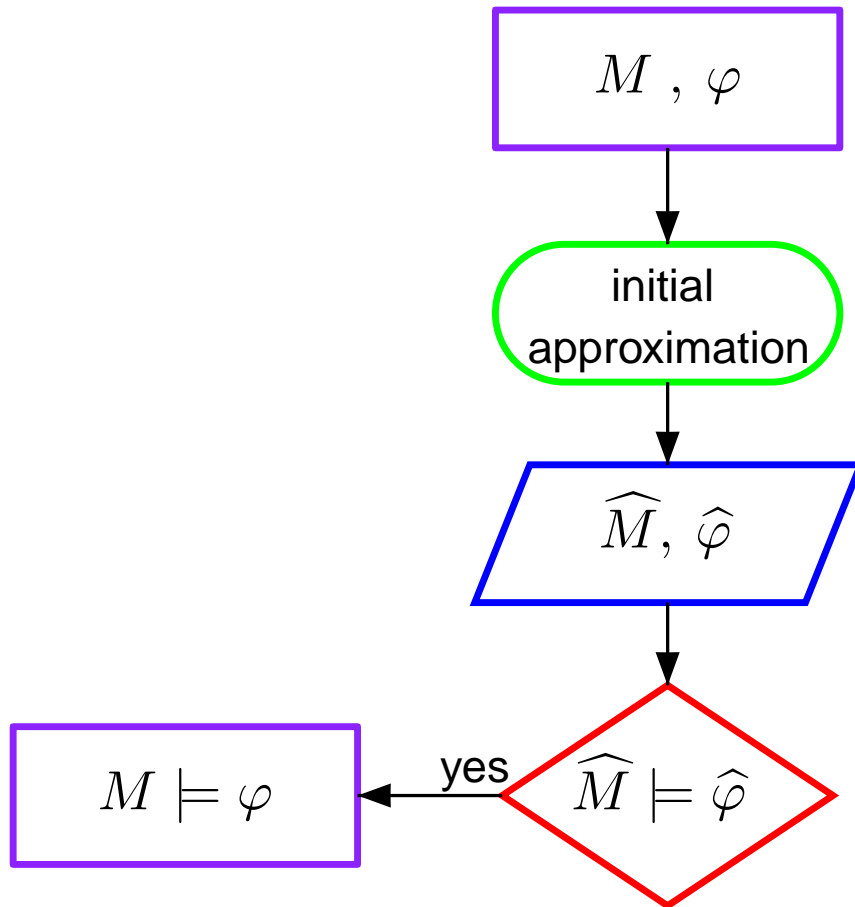
Lazy Approximation



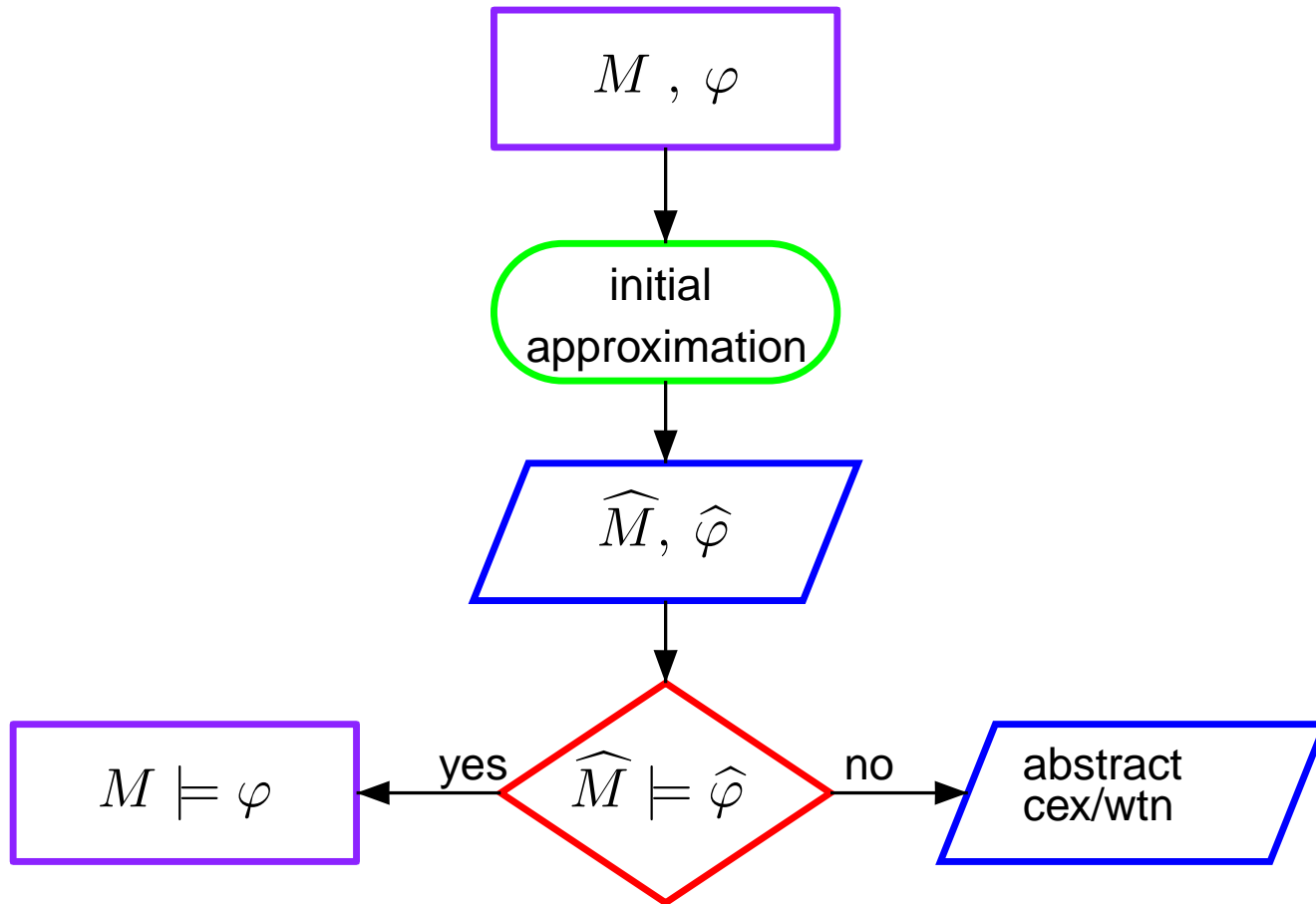
Lazy Approximation



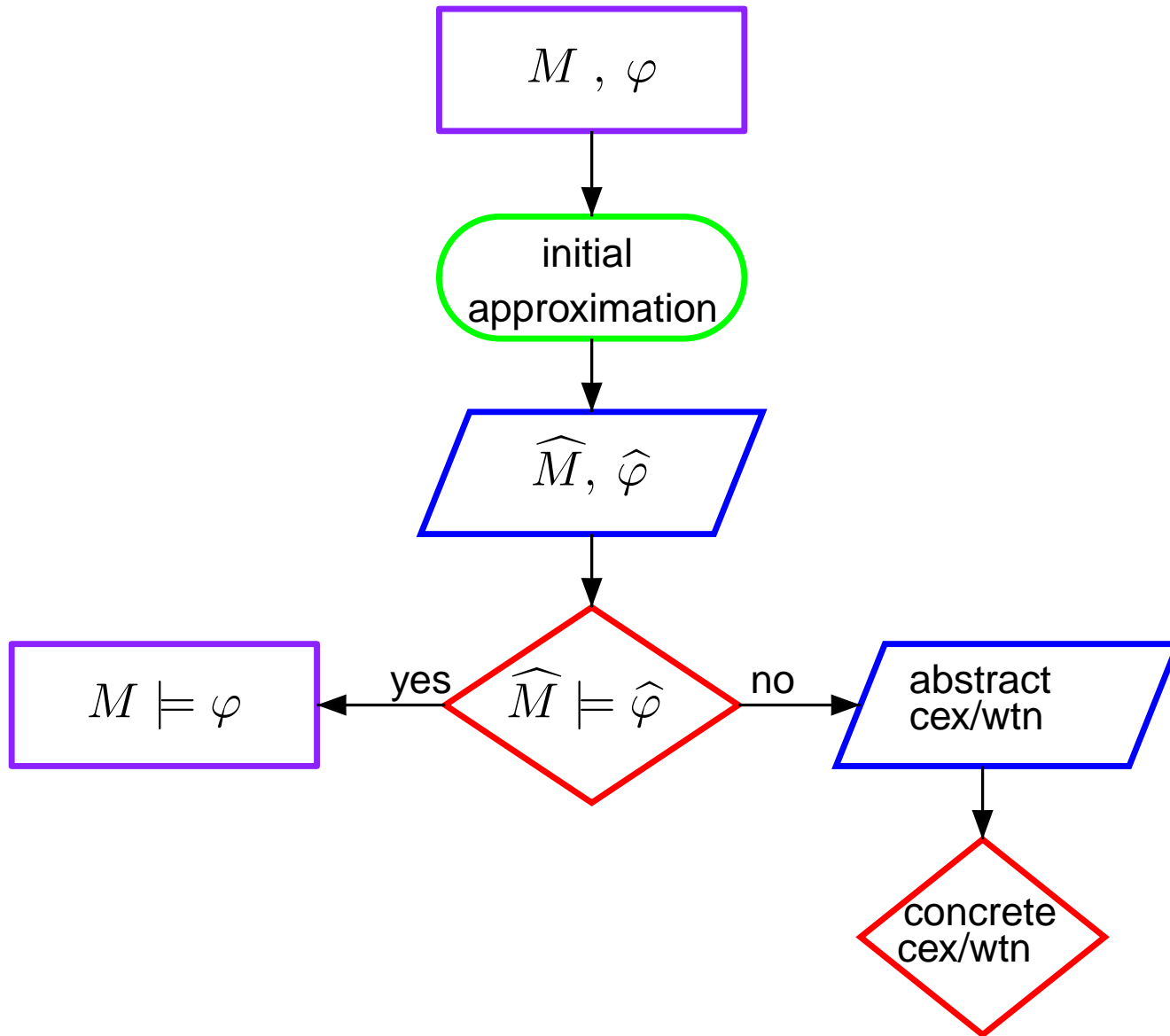
Lazy Approximation



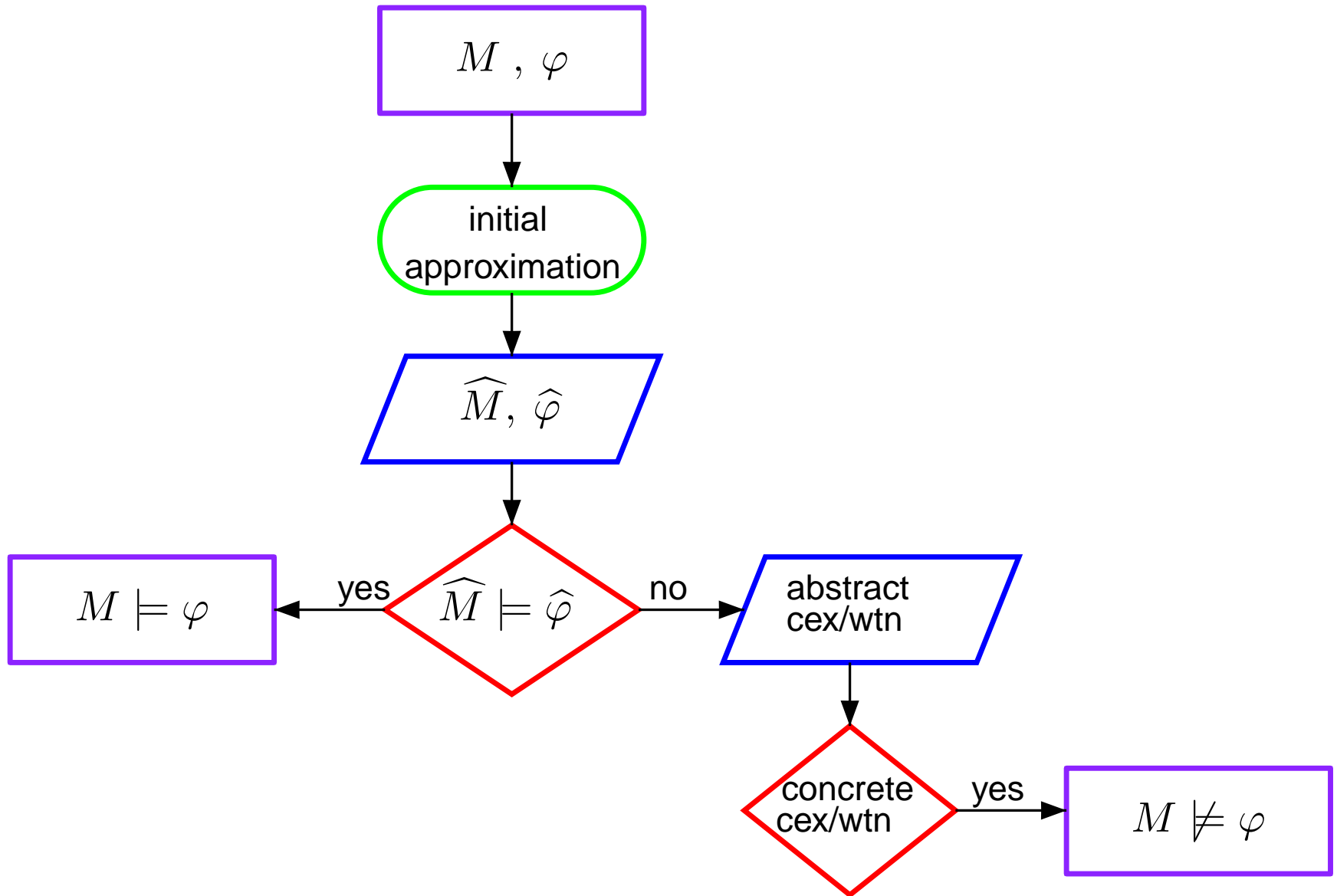
Lazy Approximation



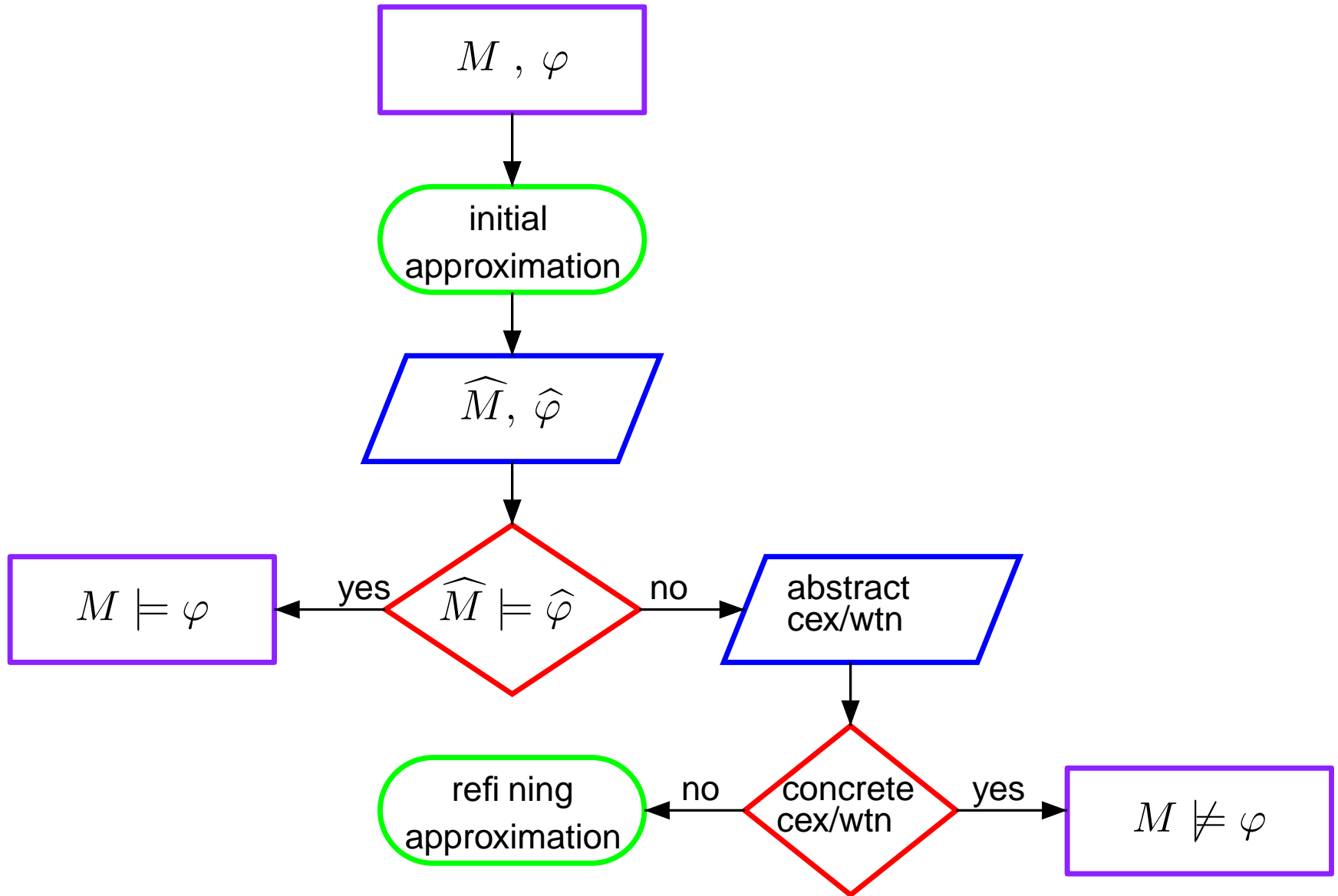
Lazy Approximation



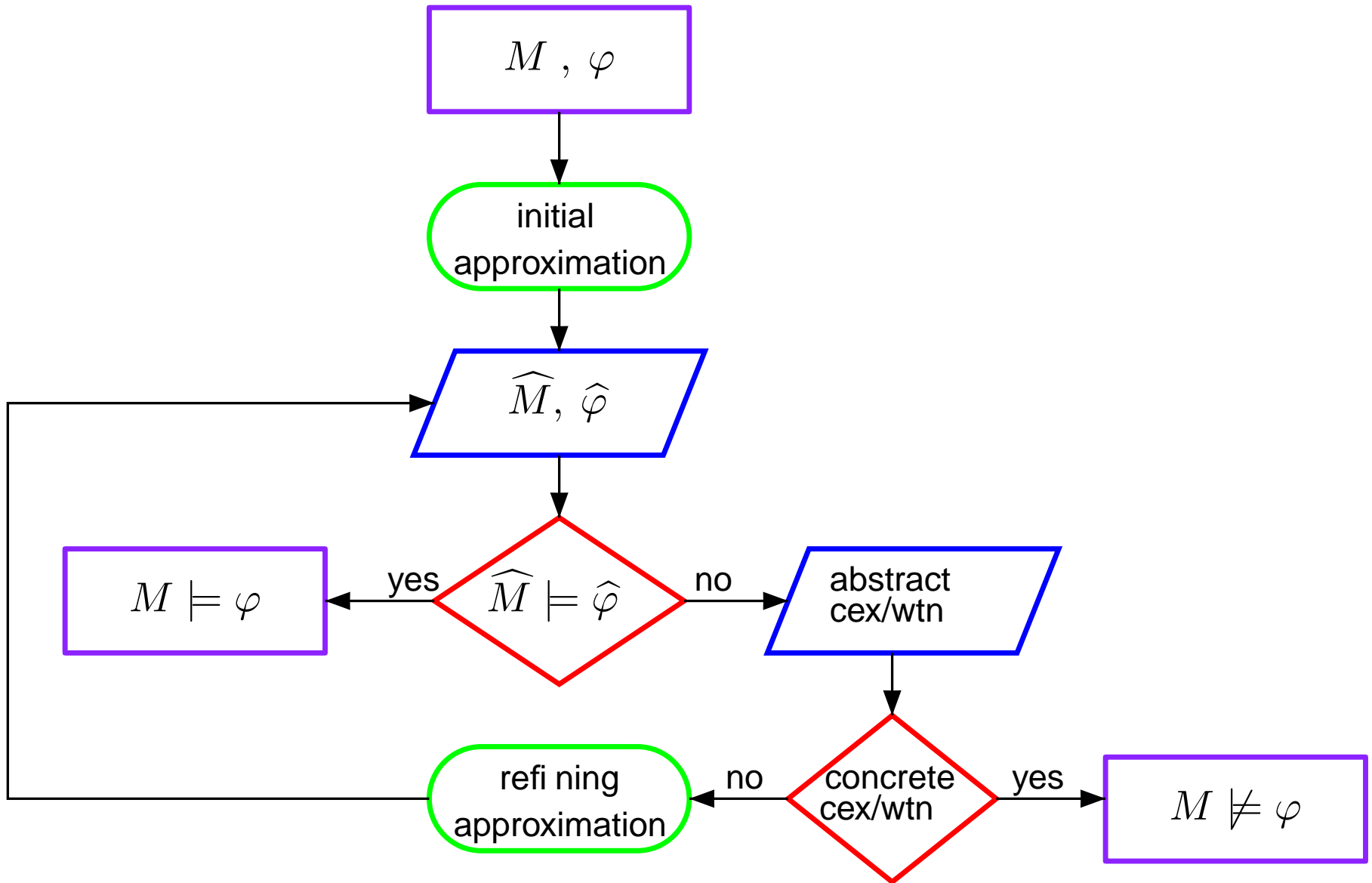
Lazy Approximation



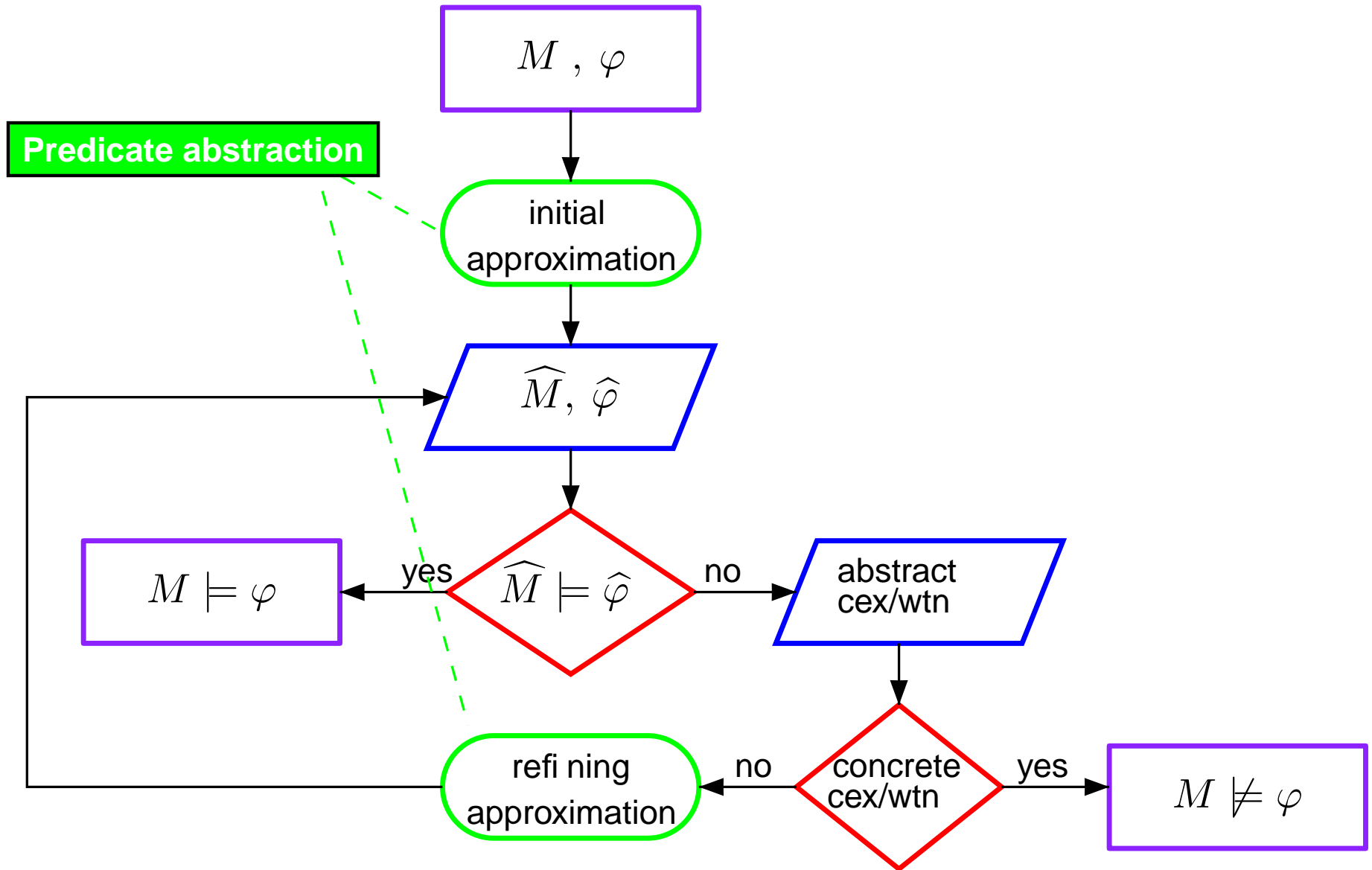
Lazy Approximation



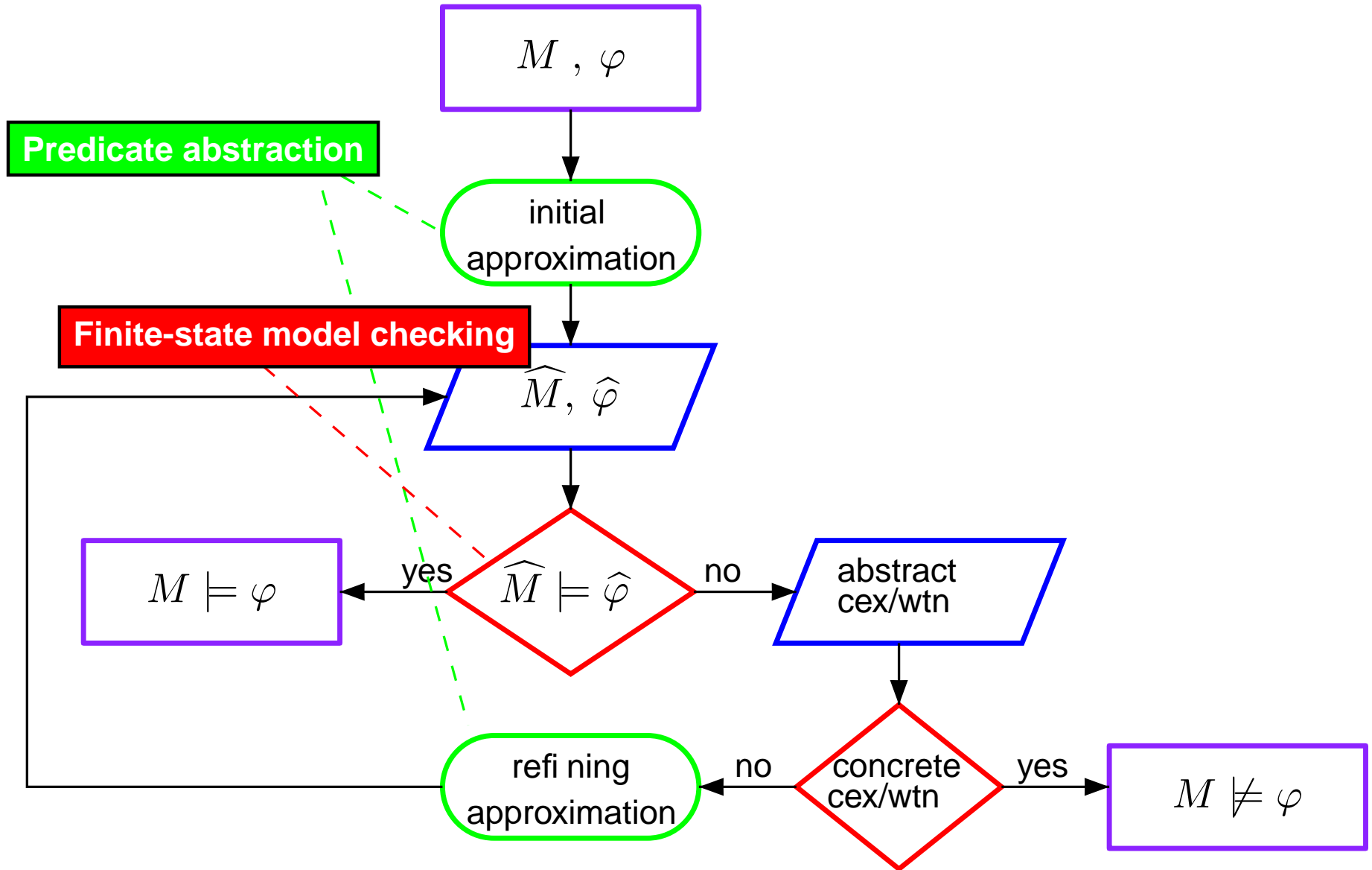
Lazy Approximation



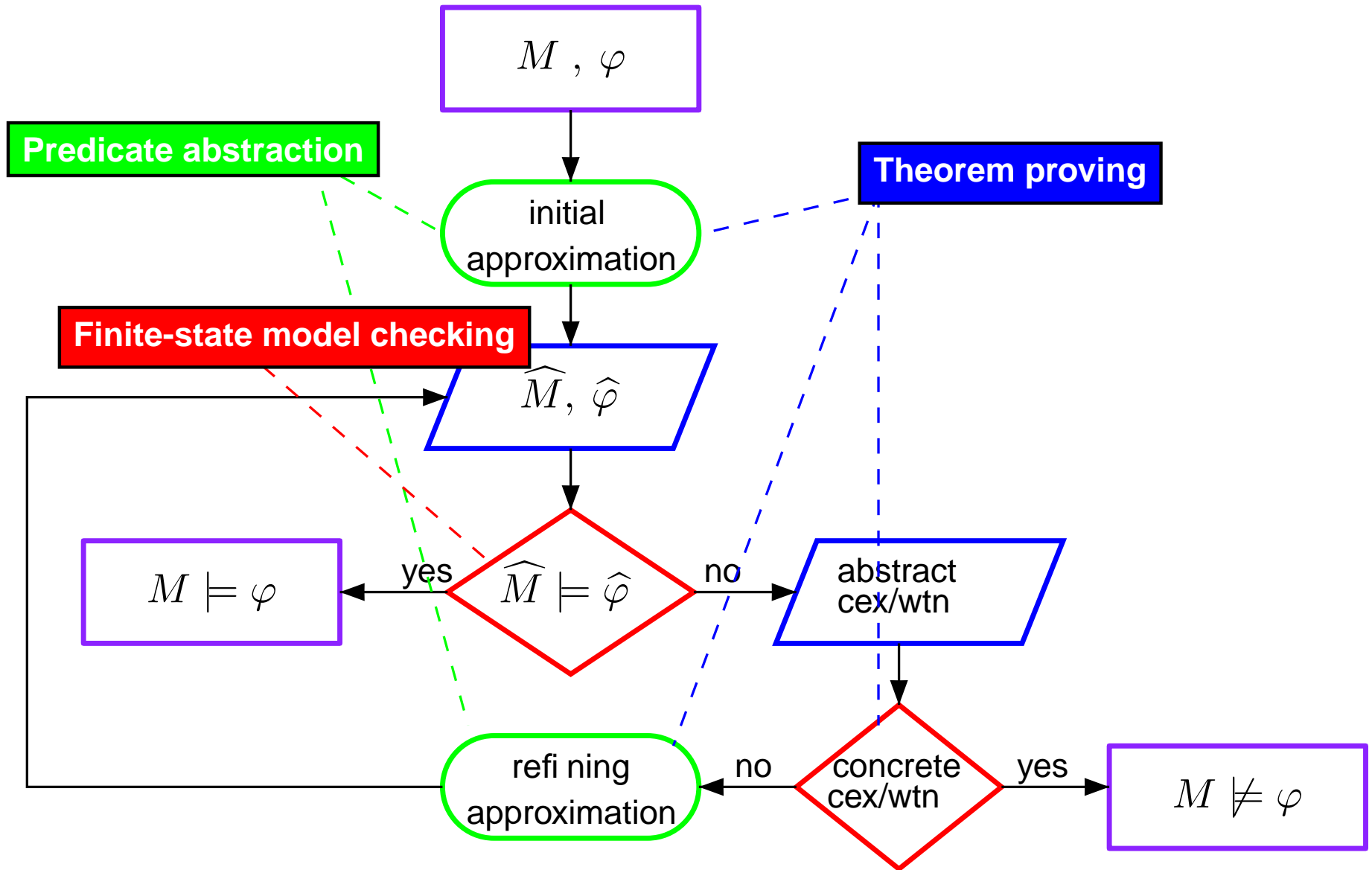
Lazy Approximation



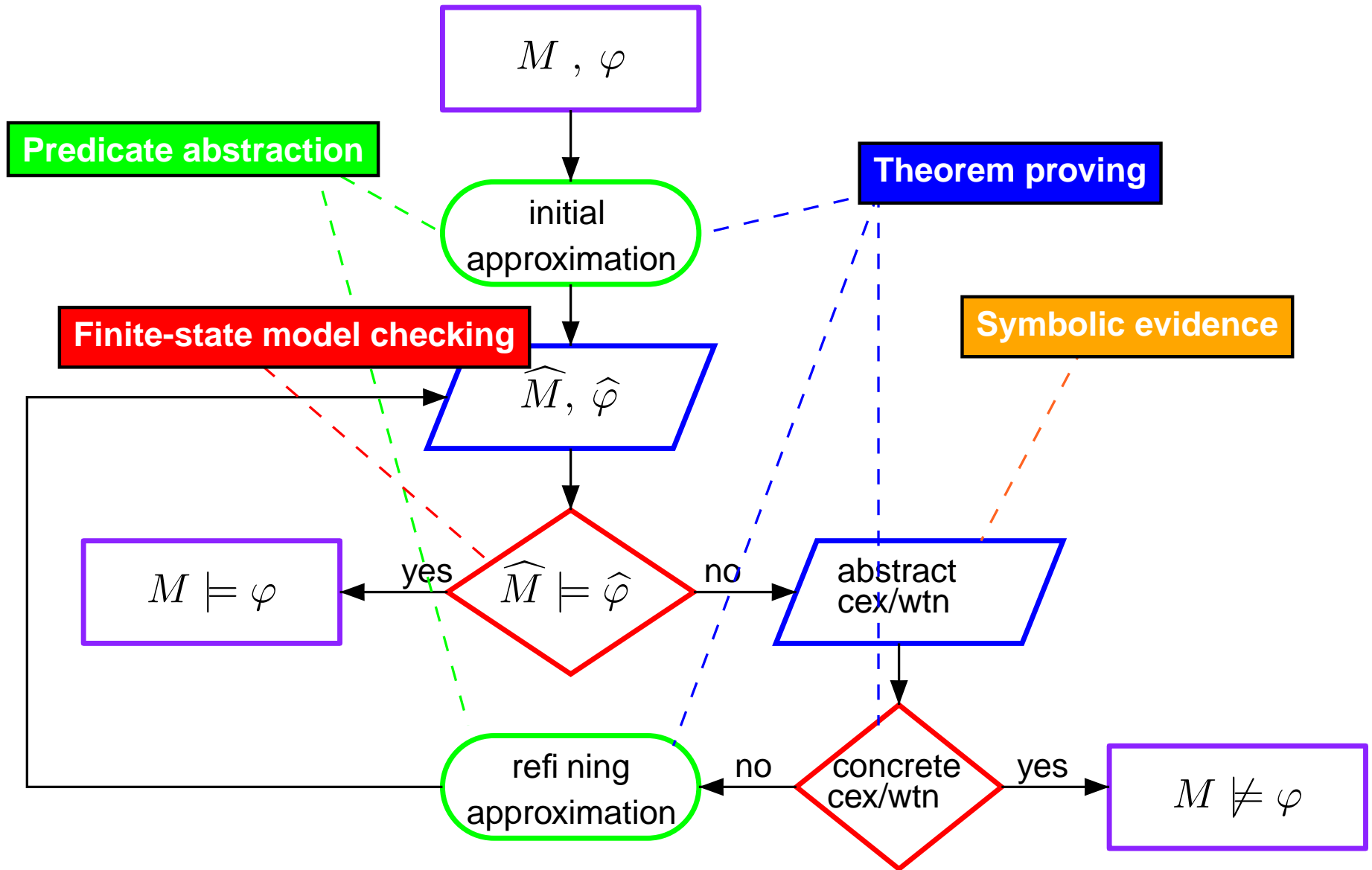
Lazy Approximation



Lazy Approximation



Lazy Approximation



Results, Conclusion

- Theorem:
 - Termination, soundness, completeness
- Typically small subsets of the basis predicates are sufficient
 - Fischer's Mutual Exclusion: 2 predicates (out of 36)
 - Train Gate Controller: 4 predicates (out of 288)
- Advantage: Applicable for larger class of problems (e.g. linear hybrid systems) \rightsquigarrow not complete anymore
- Spurious counterexamples and dubious witnesses
 - Refinement of under- und overapproximations
 - Make the refinement process converge more quickly compared to the use of linear counterexamples, as a multitude of spurious counterexamples are discarded in every refinement step
 - Lazy approximation is applicable to the full TCTL