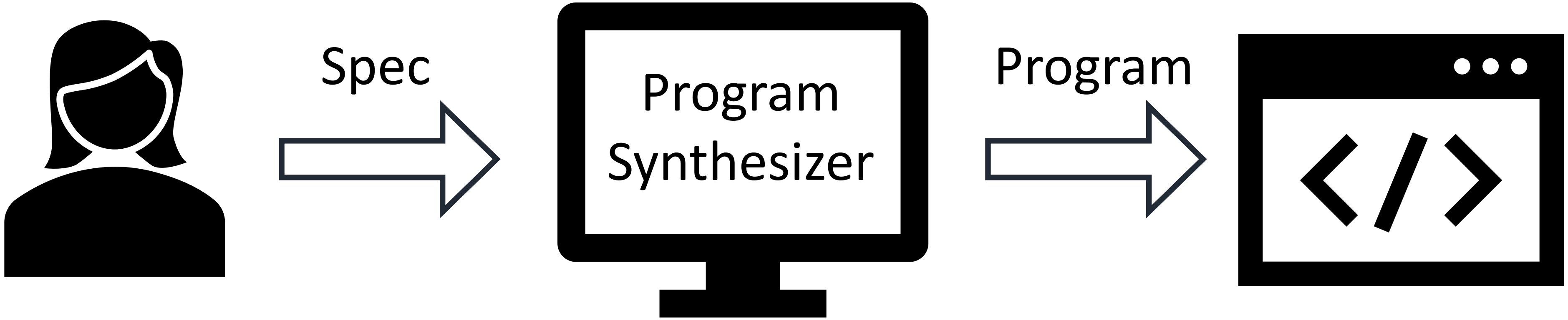
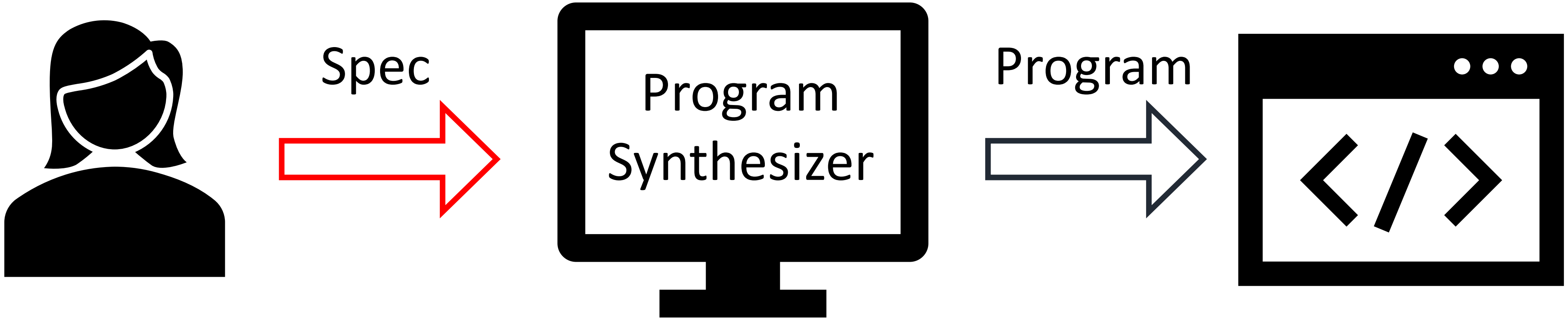
Semantics-Guided Synthesis

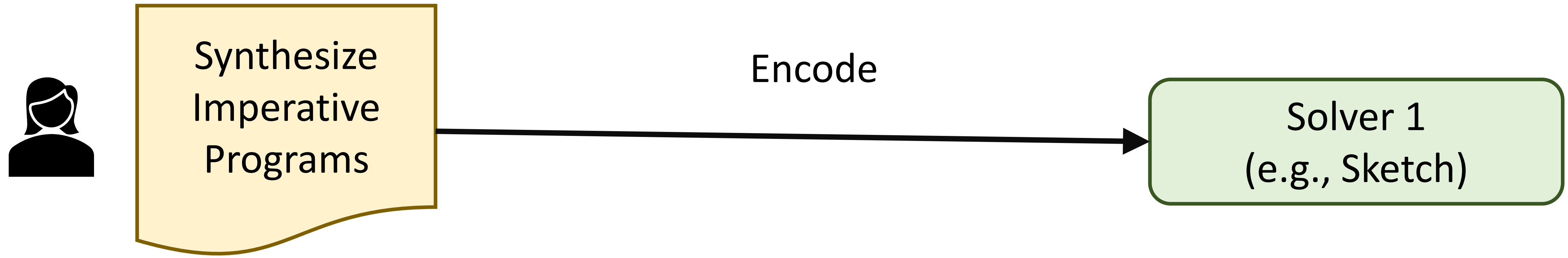
Jinwoo Kim, Qinheping Hu, Loris D'Antoni, and Thomas Reps
University of Wisconsin-Madison

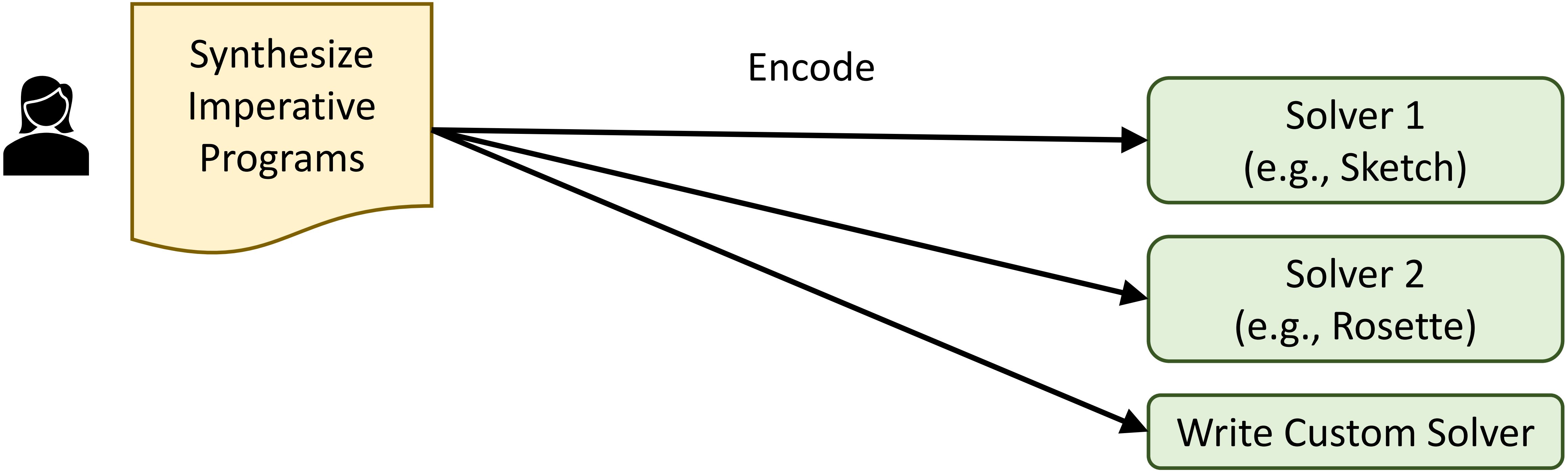
This is a Paper on Program Synthesis…

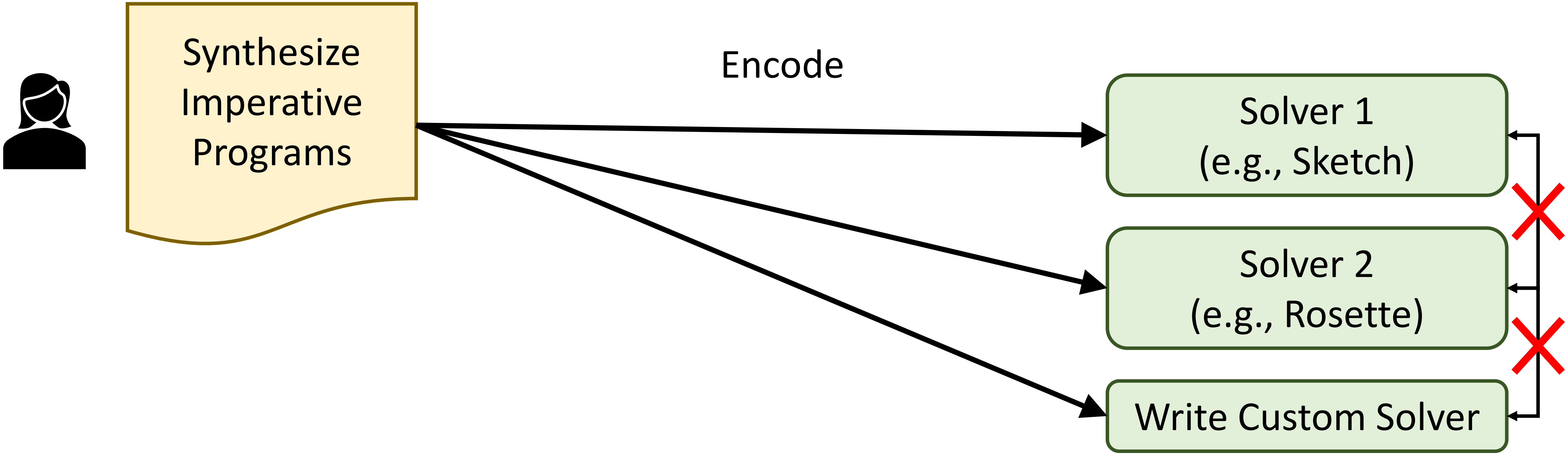


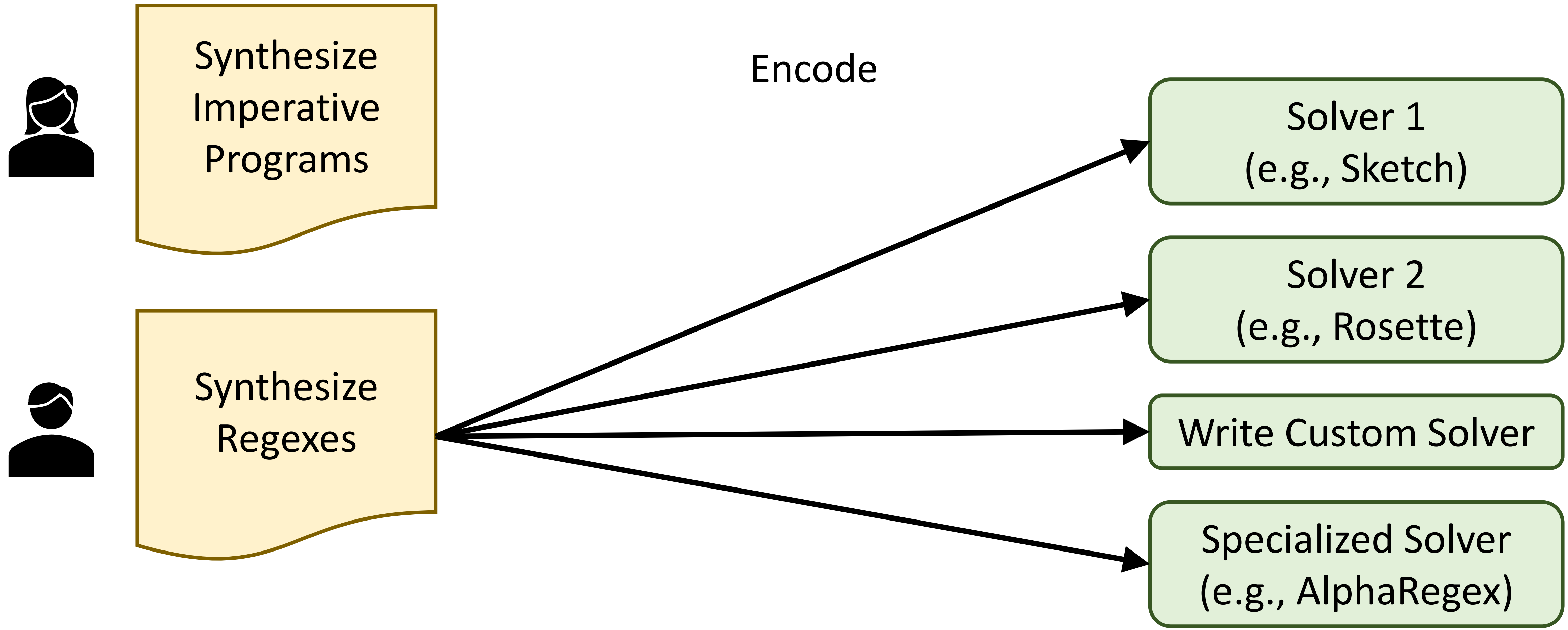
This is a Paper on Program Synthesis…

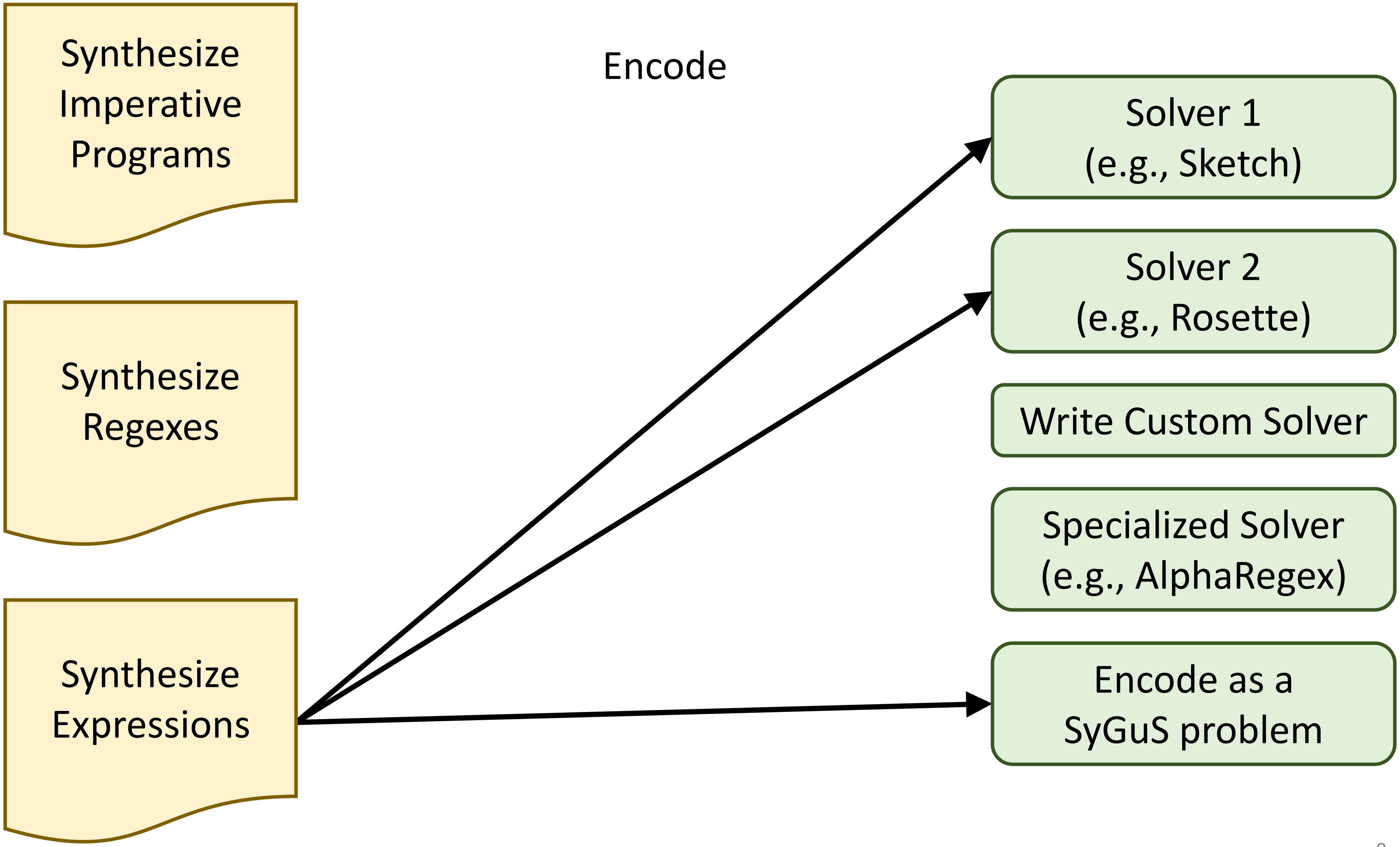


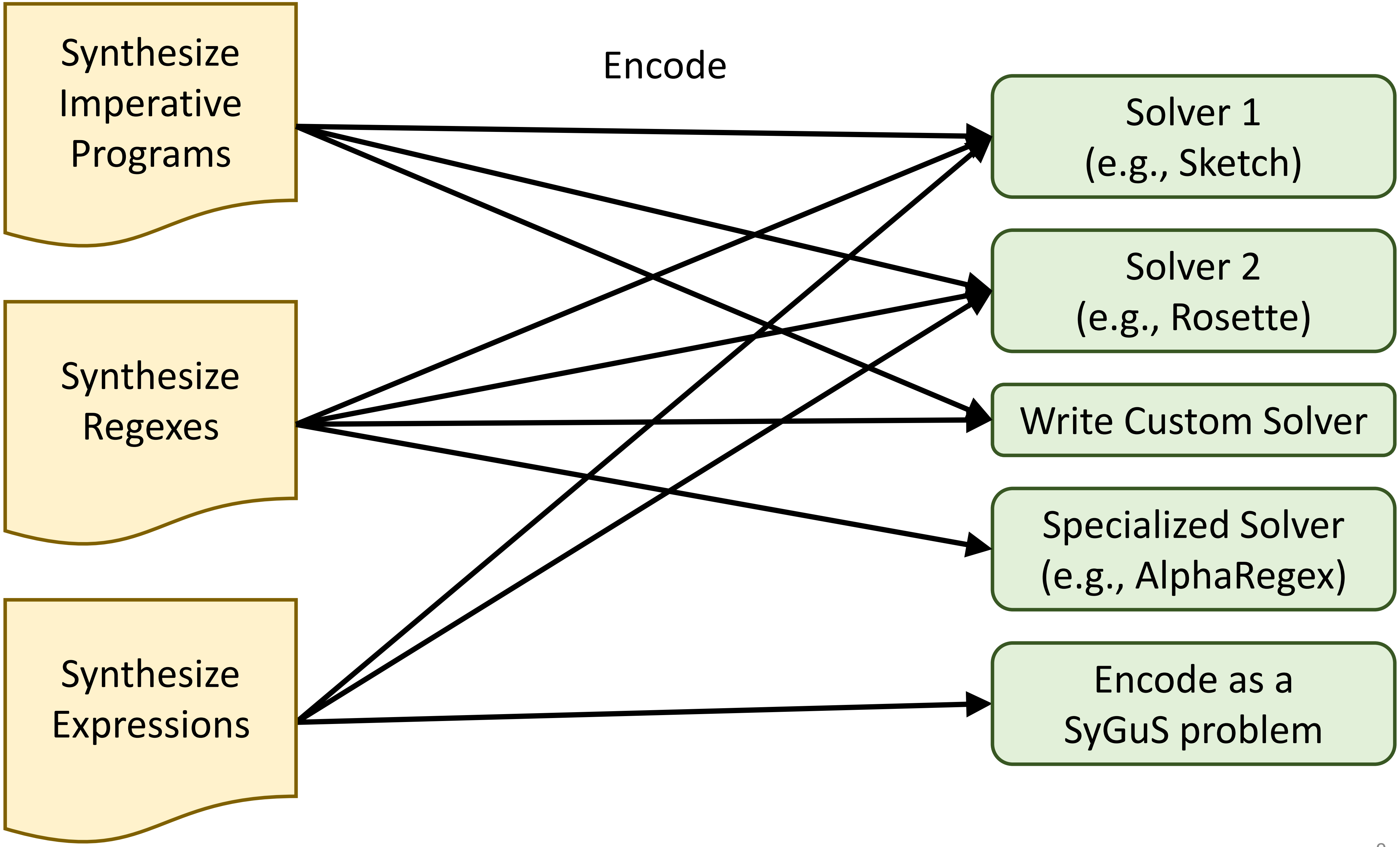

Synthesize Imperative Programs
Encode
Solver 1 (e.g., Sketch)


Synthesize Imperative Programs
Encode
Solver 1 (e.g., Sketch)
Solver 2 (e.g., Rosette)
Write Custom Solver

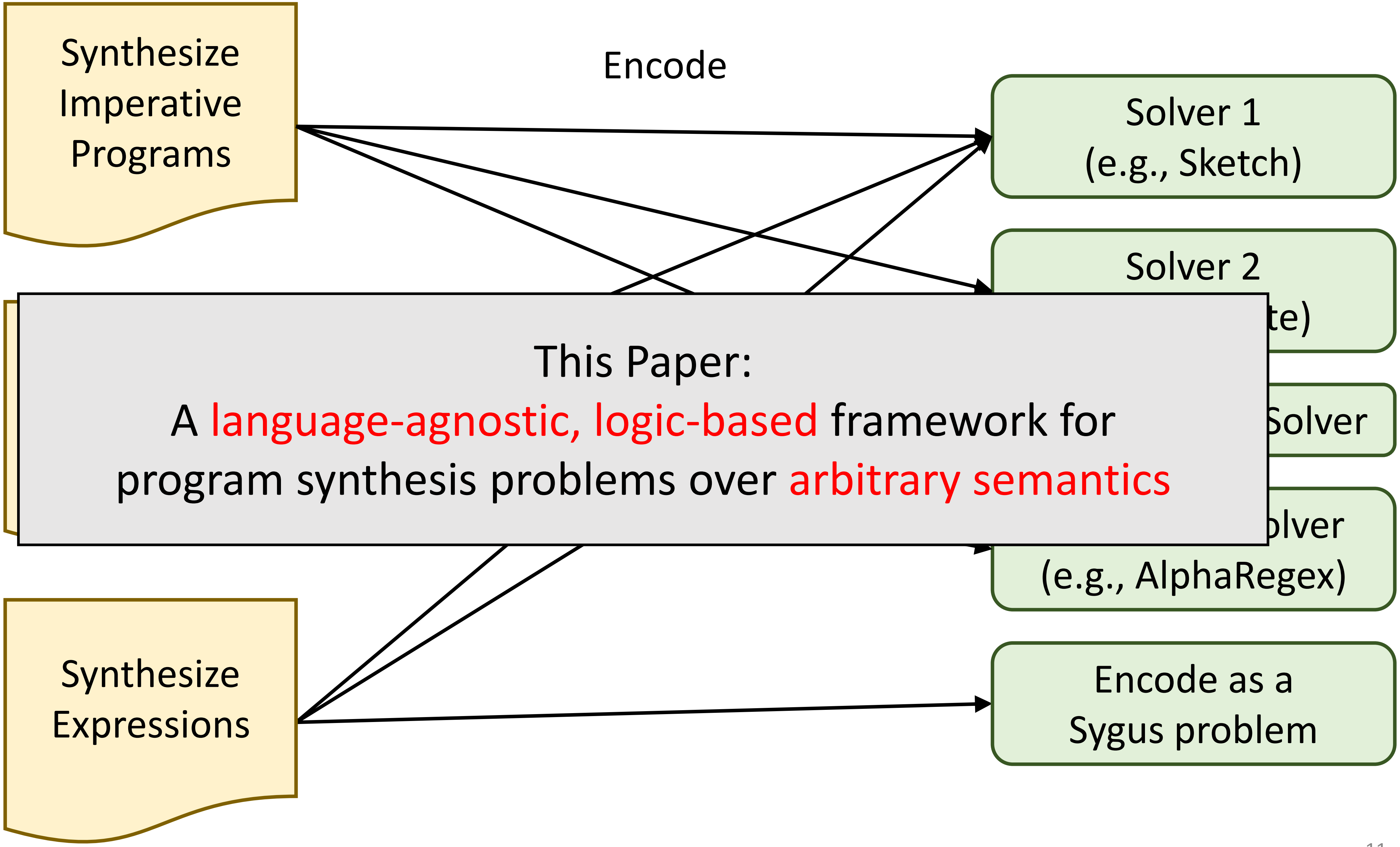

Synthesize Imperative Programs
Encode
Solver 1
(e.g., Sketch)
Solver 2
(e.g., Rosette)
Write Custom Solver


Synthesize Imperative Programs
Encode
Solver 1 (e.g., Sketch)
Synthesize Regexes
Solver 2 (e.g., Rosette)
Write Custom Solver
Specialized Solver (e.g., AlphaRegex)


Synthesize Imperative Programs
Synthesize Regexes
Synthesize Expressions
Encode
Solver 1 (e.g., Sketch)
Solver 2 (e.g., Rosette)
Write Custom Solver
Specialized Solver (e.g., AlphaRegex)
Encode as a SyGuS problem

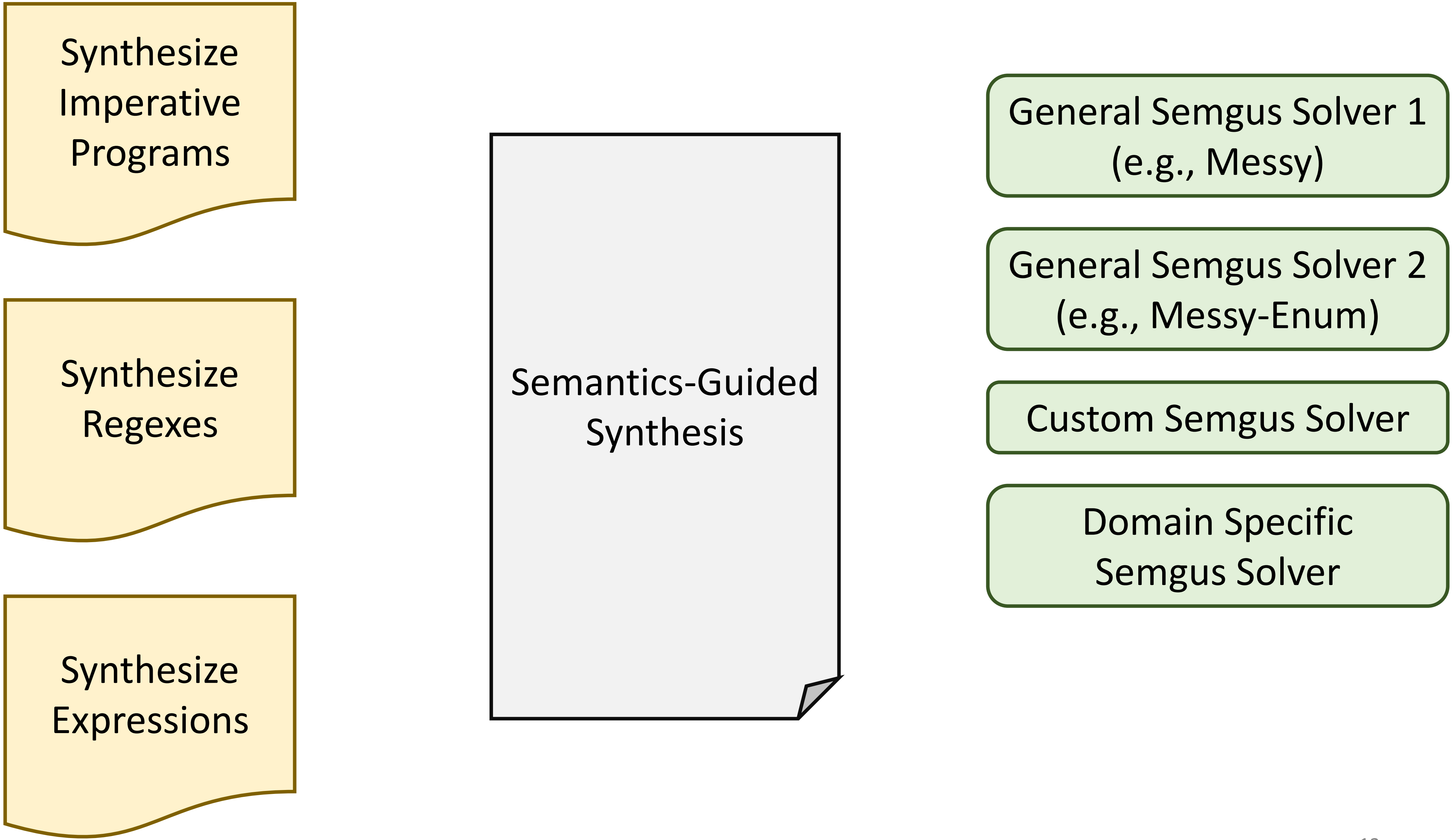

Synthesize Imperative Programs
Synthesize Regexes
Synthesize Expressions
Encode
Solver 1 (e.g., Sketch)
Solver 2 (e.g., Rosette)
Write Custom Solver
Specialized Solver (e.g., AlphaRegex)
Encode as a SyGuS problem

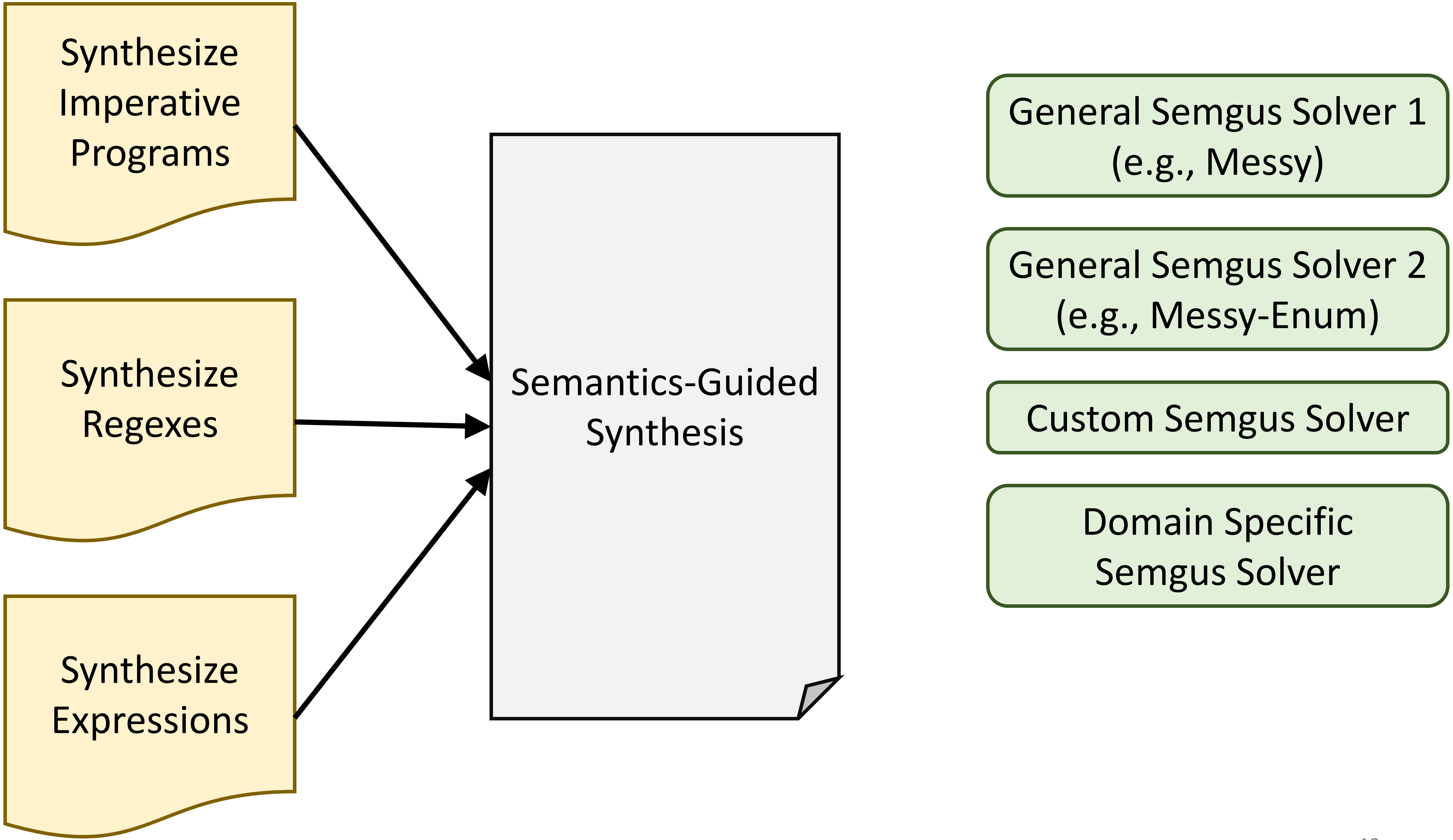
1. Semantics-Guided Synthesis
2. Solving a Semgus Problem
3. Alternative Semantics for Semgus
4. Result Highlights

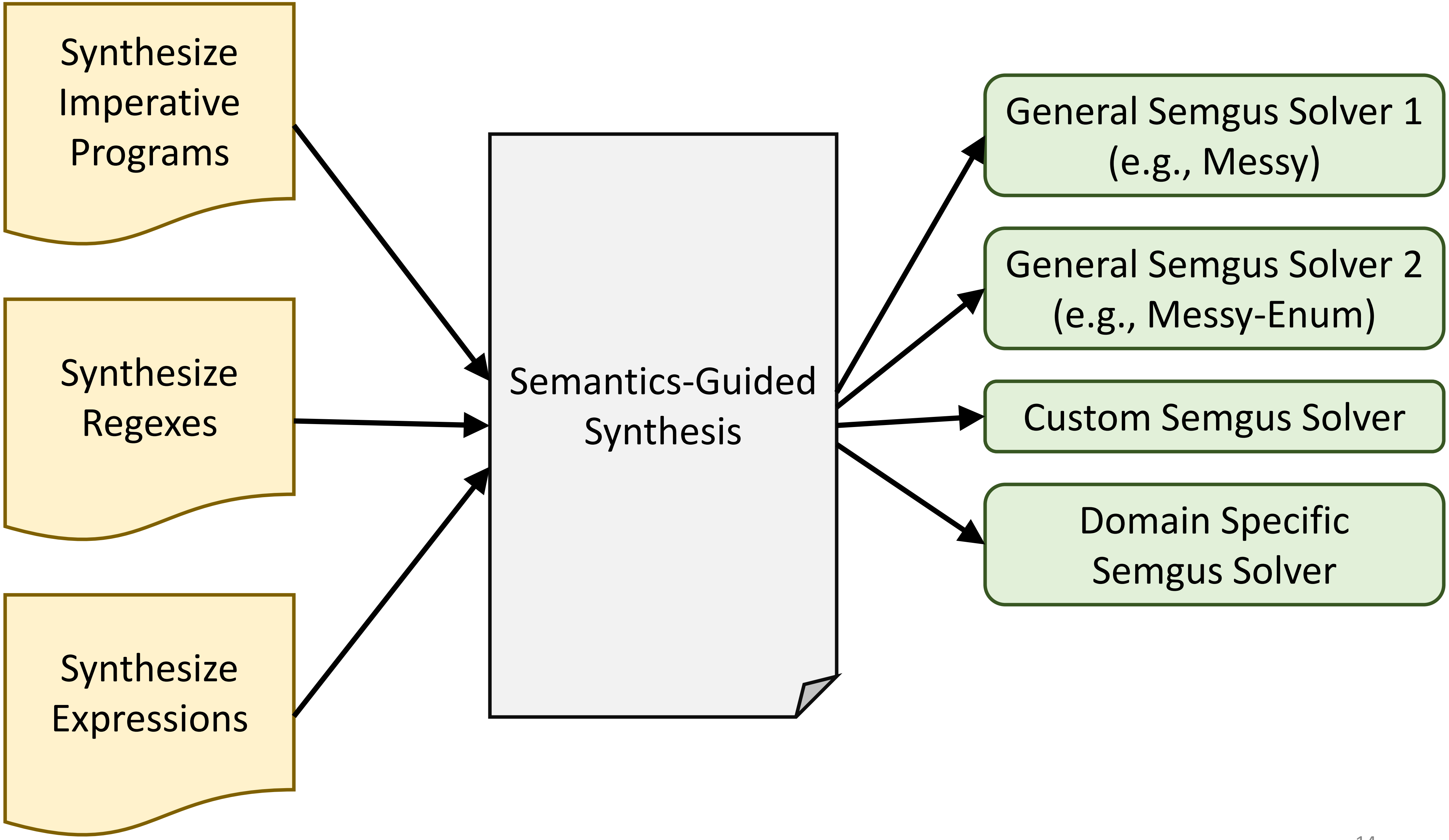


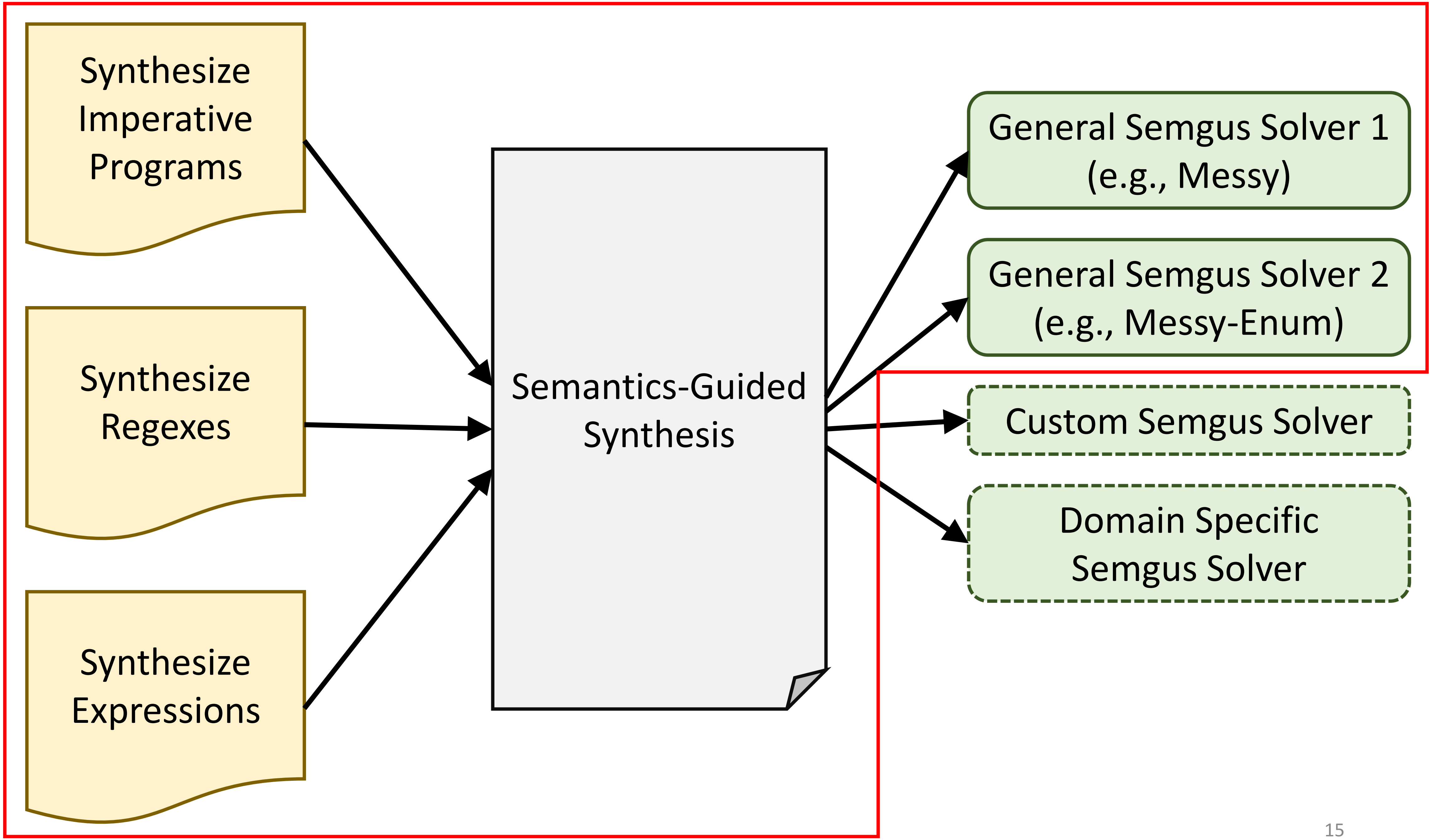
This Paper:

A language-agnostic, logic-based framework for program synthesis problems over arbitrary semantics


Synthesize Imperative Programs
Synthesize Regexes
Synthesize Expressions
Semantics-Guided Synthesis
General Semgus Solver 1 (e.g., Messy)
General Semgus Solver 2 (e.g., Messy-Enum)
Custom Semgus Solver
Domain Specific Semgus Solver


Synthesize Imperative Programs
Synthesize Regexes
Synthesize Expressions
Semantics-Guided Synthesis
General Semgus Solver 1 (e.g., Messy)
General Semgus Solver 2 (e.g., Messy-Enum)
Custom Semgus Solver
Domain Specific Semgus Solver


Synthesize Imperative Programs
Synthesize Regexes
Synthesize Expressions
Semantics-Guided Synthesis
General Semgus Solver 1 (e.g., Messy)
General Semgus Solver 2 (e.g., Messy-Enum)
Custom Semgus Solver
Domain Specific Semgus Solver


Synthesize Imperative Programs
Synthesize Regexes
Synthesize Expressions
Semantics-Guided Synthesis
General Semgus Solver 1 (e.g., Messy)
General Semgus Solver 2 (e.g., Messy-Enum)
Custom Semgus Solver
Domain Specific Semgus Solver

Semantics-Guided Synthesis Problem

Language Syntax

Regular Tree Grammar

Language Semantics

Constrained Horn Clauses

Problem Specification

Constraints / Examples

Semantics-Guided Synthesis Problem

Language Syntax

Regular Tree Grammar

Language Semantics

Constrained Horn Clauses

Problem Specification

Constraints / Examples

Semantics-Guided Synthesis Problem

Language Syntax	Language Semantics	Problem Specification
Regular Tree Grammar	Constrained Horn Clauses	Constraints / Examples

$$E ::= x \mid y \mid If\ B\ then\ E\ else\ E$$
$$B ::= E < E \mid B \land B$$

Semantics-Guided Synthesis Problem

Language Syntax

Regular Tree Grammar

Language Semantics

Constrained Horn Clauses

Problem Specification

Constraints / Examples

$$R ::= \epsilon \mid \Sigma \mid R \cdot R \mid (R \mid R) \mid R^{*}$$

Semantics-Guided Synthesis Problem

Language Syntax	Language Semantics	Problem Specification
Regular Tree Grammar	Constrained Horn Clauses	Constraints / Examples

$$
\begin{aligned}
&Start ::= While\ B\ do\ S\\
&B ::= E < E\\
&E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)\\
&S ::= S; S \mid x := E \mid y := E
\end{aligned}
$$

Semantics-Guided Synthesis Problem

Language Syntax

Regular Tree Grammar

Language Semantics

Constrained Horn Clauses

Problem Specification

Constraints / Examples

$$
\begin{aligned}
&Start ::= While\ B\ do\ S\\
&B ::= E < E\\
&E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)\\
&S ::= S; S \mid x := E \mid y := E
\end{aligned}
$$

Semantics-Guided Synthesis Problem

Language Syntax

Regular Tree Grammar

Language Semantics

Constrained Horn Clauses

Problem Specification

Constraints / Examples

$$
\begin{aligned}
&Start ::= While\ B\ do\ S \\
&B ::= E < E \\
&E ::= x \mid y \mid E \,\&\, E \mid (E \mid E) \\
&S ::= S;S \mid x \coloneqq E \mid y \coloneqq E
\end{aligned}
$$

$$Start ::= While\ B\ do\ S$$
$$B ::= E < E$$
$$E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)$$
$$\boxed{S ::= S; S \mid x := E \mid y := E}$$

$Start ::= While\ B\ do\ S$

$B ::= E < E$

$E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)$

$\boxed{S ::= S; S \mid x := E \mid y := E}$ → S: Every term in S has type $State \rightarrow State$

$Start ::= While\ B\ do\ S$

$B ::= E < E$

$E\ \ ::= x \mid y \mid E\ \&\ E \mid (E \mid E)$

$\boxed{S ::= S; S \mid x \coloneqq E \mid y \coloneqq E}$ → S: Every term in S has type $State \rightarrow State$

$$[\![\, x \coloneqq x\&y \,]\!] \begin{pmatrix} x & y \\ 1, & 0 \end{pmatrix} = \begin{pmatrix} x & y \\ 0, & 0 \end{pmatrix}$$

$Start ::= While\ B\ do\ S$

$B ::= E < E$

$E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)$

$\boxed{S ::= S; S \mid x := E \mid y := E}$

S: Every term in S has type $State \rightarrow State$

$$[\![\, x := x\&y \,]\!]\binom{x\quad y}{1, 0} = \binom{x\quad y}{0, 0}$$

Sem_S: $State \times Term \times State \rightarrow Boolean$

Relation Sem_S: Captures the semantics of all terms $t_S \in L(S)$

$Start ::= While\ B\ do\ S$

$B ::= E < E$

$E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)$

$\boxed{S ::= S; S \mid x := E \mid y := E}$ → S: Every term in S has type $State \rightarrow State$

$$[\![x := x\&y]\!] \begin{pmatrix} x & y \\ 1, & 0 \end{pmatrix} = \begin{pmatrix} x & y \\ 0, & 0 \end{pmatrix}$$

$$\boxed{Sem_S\left(\begin{pmatrix} x & y \\ 1, & 0 \end{pmatrix}, x := x\&y, \begin{pmatrix} x & y \\ 0, & 0 \end{pmatrix}\right) = True}$$

$Sem_S: State \times Term \times State \rightarrow Boolean$

Relation Sem_S: Captures the semantics of all terms $t_S \in L(S)$

$$Start ::= While\ B\ do\ S$$
$$B ::= E < E$$
$$E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)$$
$$S ::= S; S \mid x \coloneqq E \mid y \coloneqq E$$

S: Every term in S has type $State \rightarrow State$

$$[\![\, x \coloneqq x\&y \,]\!] \begin{pmatrix} x & y \\ 1, & 0 \end{pmatrix} = \begin{pmatrix} x & y \\ 0, & 0 \end{pmatrix}$$

$$Sem_S\left(\begin{pmatrix} x & y \\ 1, & 0 \end{pmatrix}, x \coloneqq x\&y, \begin{pmatrix} x & y \\ 0, & 0 \end{pmatrix}\right) = True$$

Sem_S: $State \times Term \times State \rightarrow Boolean$

Relation Sem_S: Captures the semantics of all terms $t_S \in L(S)$

$$Sem_S(\Gamma, t_S, \Gamma') \Leftrightarrow [\![\, t_S \,]\!](\Gamma) = \Gamma'$$

t_s returns Γ' when executed on Γ

$$S ::= x \coloneqq E$$

$$S ::= x \coloneqq E$$

$$\frac{\llbracket\, t_e \,\rrbracket(\Gamma) = v \qquad \Gamma_1 = \Gamma[x \mapsto v]}{\llbracket\, x \coloneqq t_e \,\rrbracket(\Gamma) = \Gamma_1}$$

$$S ::= x \coloneqq E$$

$$\frac{[\![\, t_e \,]\!](\Gamma) = v \quad \Gamma_1 = \Gamma[x \mapsto v]}{[\![\, x \coloneqq t_e \,]\!](\Gamma) = \Gamma_1}$$

$$\boxed{\frac{Sem_E(\Gamma, t_e, v) \quad \Gamma_1 = \Gamma[x \mapsto v]}{Sem_S(\Gamma, x \coloneqq t_e, \Gamma_1)}}$$

$S ::= x \coloneqq E$

$$\frac{[\![\, t_e \,]\!](\Gamma) = v \quad \Gamma_1 = \Gamma[x \mapsto v]}{[\![\, x \coloneqq t_e \,]\!](\Gamma) = \Gamma_1}$$

$$\frac{Sem_E(\Gamma, t_e, v) \quad \Gamma_1 = \Gamma[x \mapsto v]}{Sem_S(\Gamma, x \coloneqq t_e, \Gamma_1)}$$

$$\Updownarrow$$

$$\forall \Gamma, \Gamma_1, t_e, v.\ \overbrace{Sem_E(\Gamma, t_e, v)}^{\text{Relation Part}} \land \underbrace{\Gamma_1 = \Gamma[x \mapsto v]}_{\text{First-Order Constraint}} \Rightarrow \overbrace{Sem_S(\Gamma, x \coloneqq t_e, \Gamma_1)}^{\text{Formula Head}}$$

$S ::= x := E$

$$\frac{[\![t_e]\!](\Gamma) = v \quad \Gamma_1 = \Gamma[x \mapsto v]}{[\![x := t_e]\!](\Gamma) = \Gamma_1}$$

$$\frac{Sem_E(\Gamma, t_e, v) \quad \Gamma_1 = \Gamma[x \mapsto v]}{\boxed{Sem_S(\Gamma, x := t_e, \Gamma_1)}}$$

$$\Updownarrow$$

$$\forall \Gamma, \Gamma_1, t_e, v.\ \overbrace{Sem_E(\Gamma, t_e, v)}^{\text{Relation Part}} \land \underbrace{\Gamma_1 = \Gamma[x \mapsto v]}_{\text{First-Order Constraint}} \Rightarrow \boxed{\overbrace{Sem_S(\Gamma, x := t_e, \Gamma_1)}^{\text{Formula Head}}}$$

$$S ::= x \coloneqq E$$

$$\frac{[\![t_e]\!](\Gamma) = v \quad \Gamma_1 = \Gamma[x \mapsto v]}{[\![x \coloneqq t_e]\!](\Gamma) = \Gamma_1}$$

$$\boxed{\frac{Sem_E(\Gamma, t_e, v) \quad \Gamma_1 = \Gamma[x \mapsto v]}{}}\; Sem_S(\Gamma, x \coloneqq t_e, \Gamma_1)$$

$$\Updownarrow$$

$$\forall \Gamma, \Gamma_1, t_e, v.\ \boxed{\overbrace{Sem_E(\Gamma, t_e, v)}^{\text{Relation Part}} \wedge \underbrace{\Gamma_1 = \Gamma[x \mapsto v]}_{\text{First-Order Constraint}}} \Rightarrow \overbrace{Sem_S(\Gamma, x \coloneqq t_e, \Gamma_1)}^{\text{Formula Head}}$$

$$Start ::= While\ B\ do\ S$$

$$\frac{Sem_B(\Gamma, t_b, v) \quad v = True \quad Sem_S(\Gamma, t_s, \Gamma_1) \quad Sem_{Start}(\Gamma_1, While\ t_b\ do\ t_s, \Gamma_2)}{Sem_{Start}(\Gamma, While\ t_b\ do\ t_s, \Gamma_2)}$$

$$\frac{Sem_B(\Gamma, t_b, v) \quad v = False}{Sem_{Start}(\Gamma, While\ t_b\ do\ t_s, \Gamma)}$$

$Start ::= While\ B\ do\ S$

$$\boxed{\frac{Sem_B(\Gamma, t_b, v) \quad v = True \quad Sem_S(\Gamma, t_s, \Gamma_1) \quad Sem_{Start}(\Gamma_1, While\ t_b\ do\ t_s, \Gamma_2)}{Sem_{Start}(\Gamma, While\ t_b\ do\ t_s, \Gamma_2)}}$$

$$\frac{Sem_B(\Gamma, t_b, v) \quad v = False}{Sem_{Start}(\Gamma, While\ t_b\ do\ t_s, \Gamma)}$$

$$Start ::= While\ B\ do\ S$$

$$\frac{Sem_B(\Gamma, t_b, v) \quad v = True \quad Sem_S(\Gamma, t_s, \Gamma_1) \quad Sem_{Start}(\Gamma_1, While\ t_b\ do\ t_s, \Gamma_2)}{Sem_{Start}(\Gamma, While\ t_b\ do\ t_s, \Gamma_2)}$$

$$\boxed{\frac{Sem_B(\Gamma, t_b, v) \quad v = False}{Sem_{Start}(\Gamma, While\ t_b\ do\ t_s, \Gamma)}}$$

Semantics-Guided Synthesis Problem

Language Syntax

Regular Tree Grammar

Language Semantics

Constrained Horn Clauses

Problem Specification

Constraints / Examples

$$
\begin{aligned}
&Start ::= While\ B\ do\ S\\
&B ::= E < E\\
&E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)\\
&S ::= S; S \mid x \coloneqq E \mid y \coloneqq E
\end{aligned}
$$

$$
\frac{Sem_E(\Gamma, t_e, v)\ \ \Gamma_1 = \Gamma[x \mapsto v]}{Sem_S(\Gamma, x \coloneqq t_e, \Gamma_1)}
$$

Semantics-Guided Synthesis Problem

Language Syntax

Regular Tree Grammar

Language Semantics

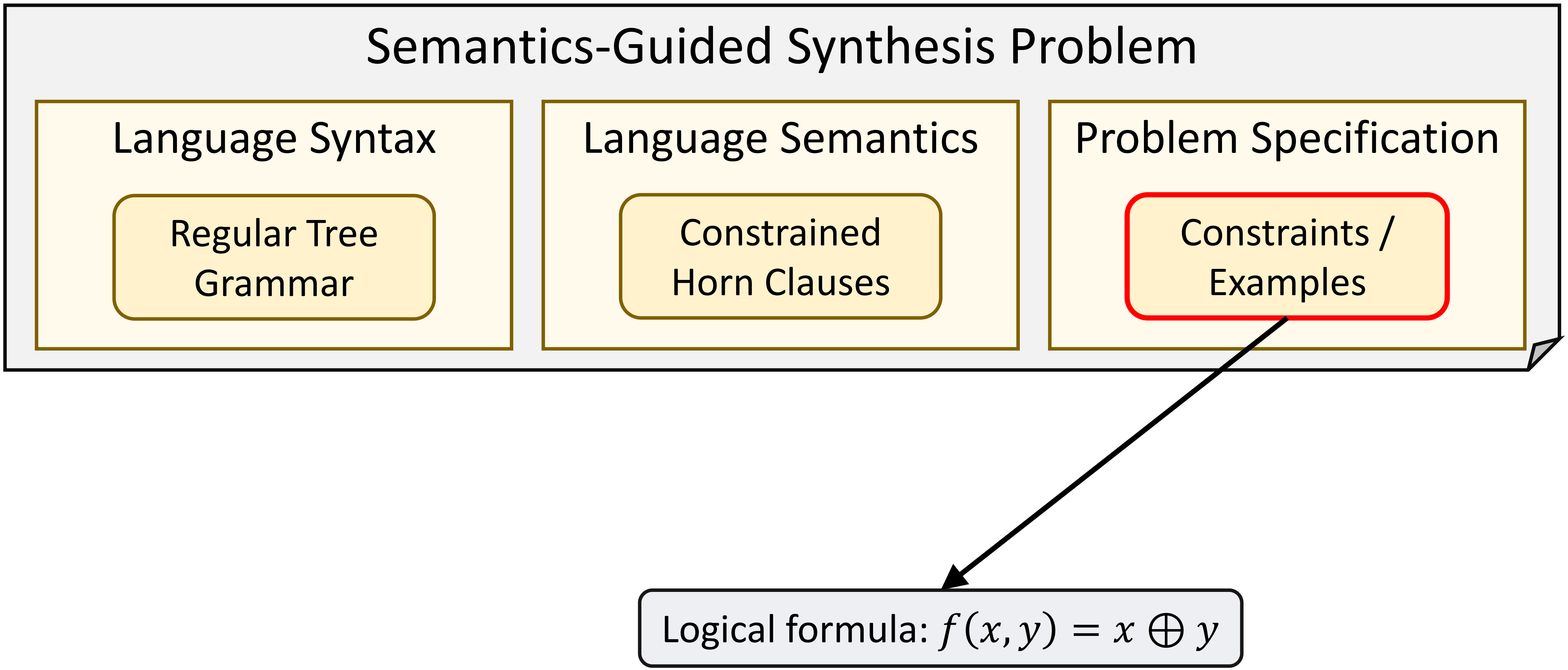
Constrained Horn Clauses

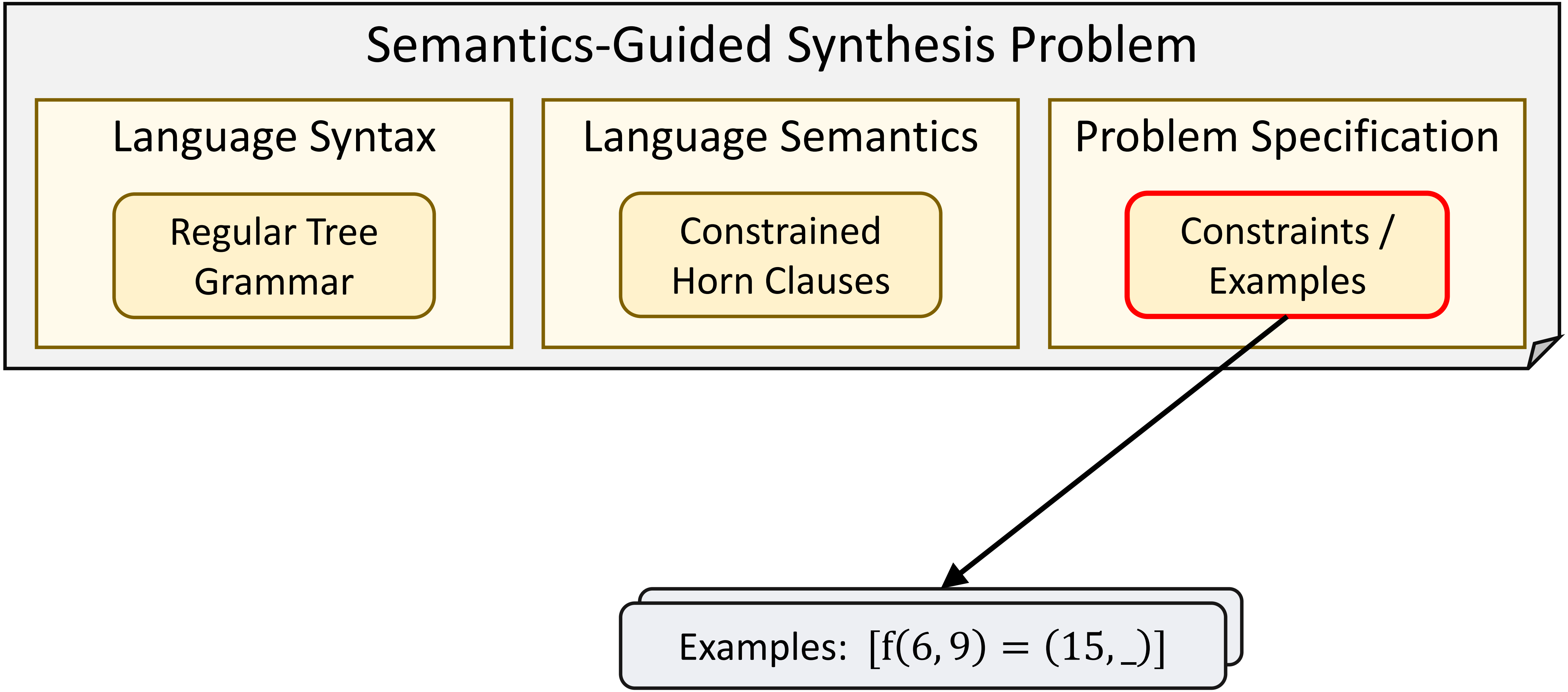
Problem Specification

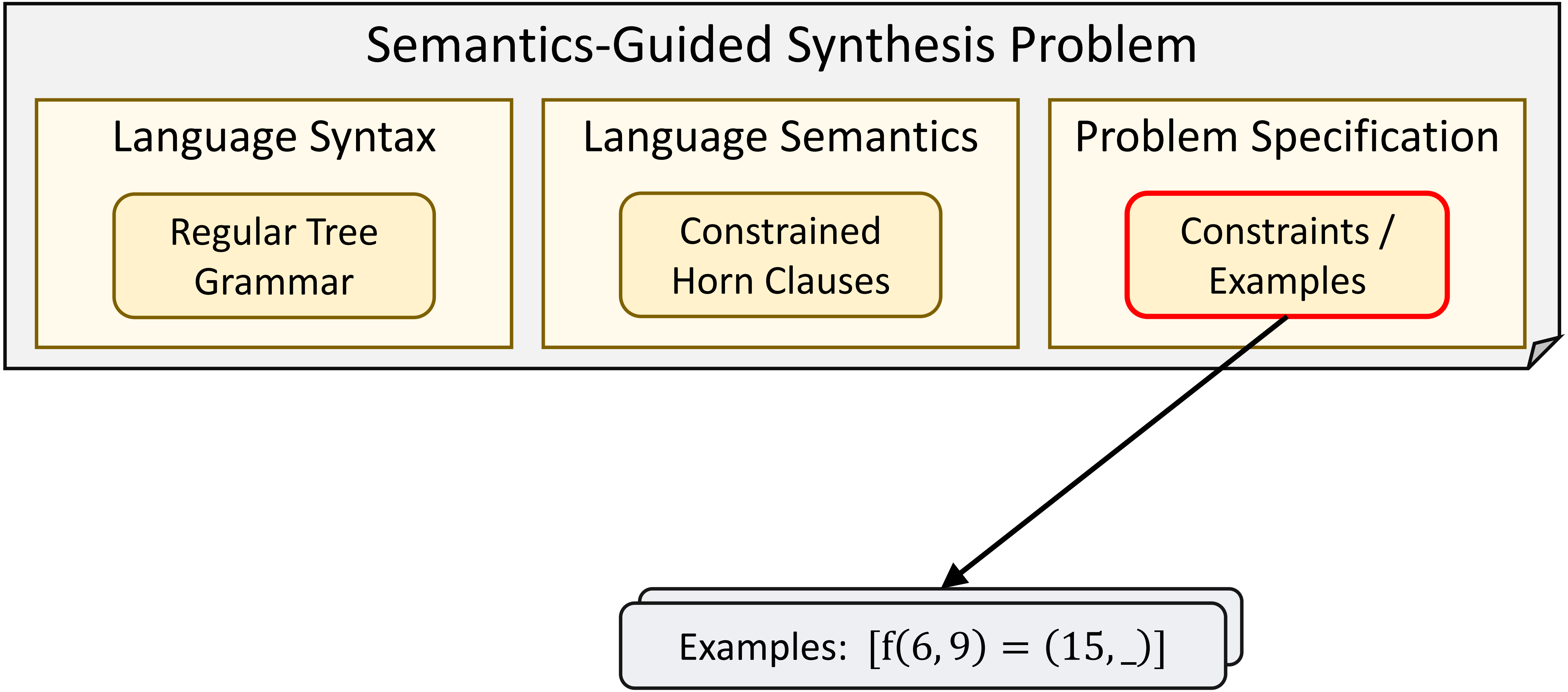
Constraints / Examples

$$
\begin{aligned}
&Start ::= While\ B\ do\ S \\
&B ::= E < E \\
&E ::= x \mid y \mid E \,\&\, E \mid (E \mid E) \\
&S ::= S; S \mid x := E \mid y := E
\end{aligned}
$$

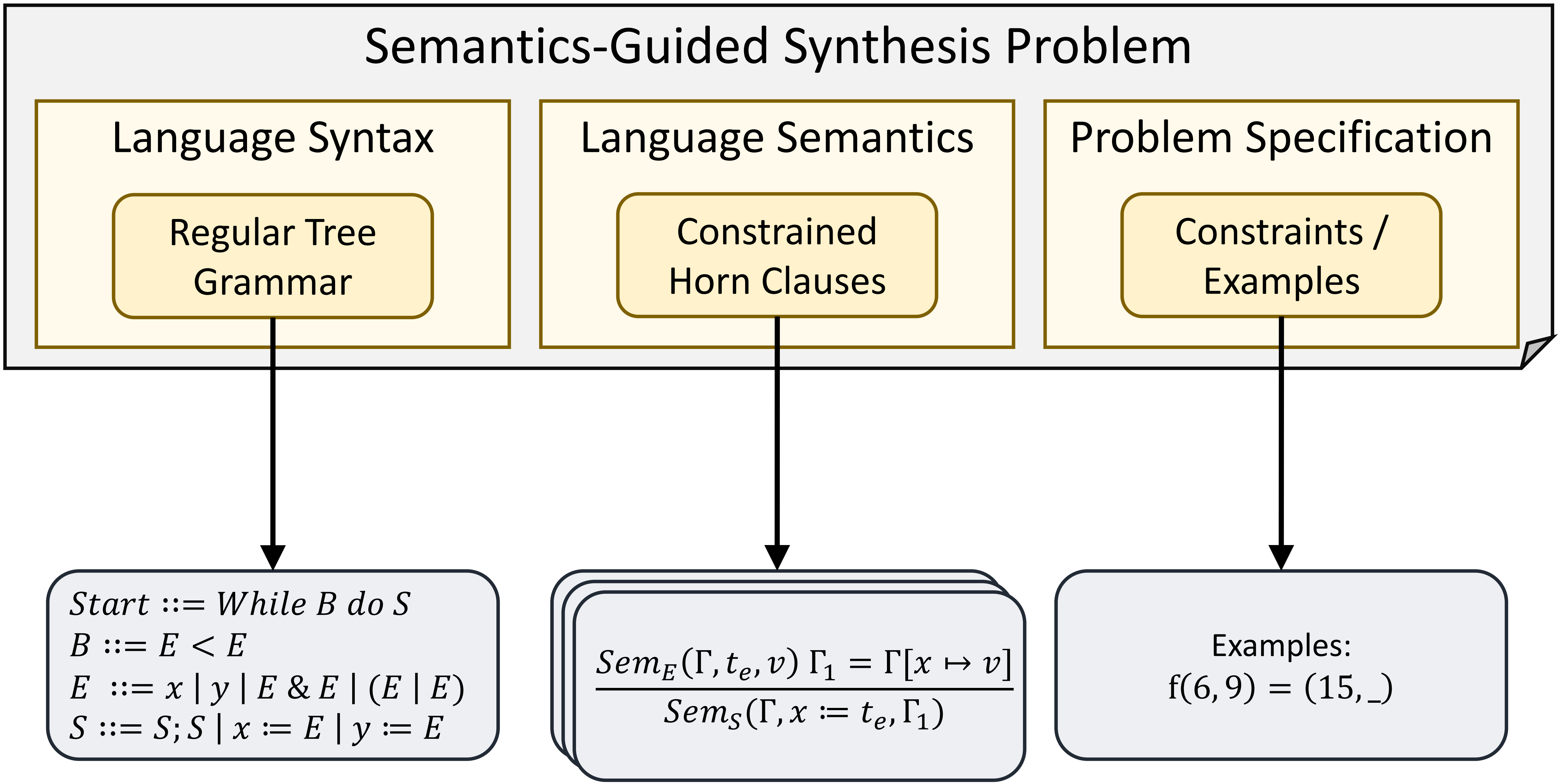
$$\frac{Sem_E(\Gamma, t_e, v)\ \ \Gamma_1 = \Gamma[x \mapsto v]}{Sem_S(\Gamma, x := t_e, \Gamma_1)}$$


Semantics-Guided Synthesis Problem
Language Syntax
Regular Tree Grammar
Language Semantics
Constrained Horn Clauses
Problem Specification
Constraints / Examples
Logical formula: $f(x, y) = x \oplus y$


Semantics-Guided Synthesis Problem
Language Syntax
Regular Tree Grammar
Language Semantics
Constrained Horn Clauses
Problem Specification
Constraints / Examples
Examples: $[\mathrm{f}(6,9) = (15, \_)]$



CEGIS: We can always generate more counterexamples!


Semantics-Guided Synthesis Problem
Language Syntax
Regular Tree Grammar
Language Semantics
Constrained Horn Clauses
Problem Specification
Constraints / Examples
$Start ::= While\ B\ do\ S$
$B ::= E < E$
$E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)$
$S ::= S; S \mid x := E \mid y := E$
$\dfrac{Sem_E(\Gamma, t_e, v)\ \Gamma_1 = \Gamma[x \mapsto v]}{Sem_S(\Gamma, x := t_e, \Gamma_1)}$
Examples:
$f(6, 9) = (15, \_)$

1. Semantics-Guided Synthesis
2. Solving a Semgus Problem
3. Alternative Semantics for Semgus
4. Result Highlights

Semantics-Guided Synthesis Problem

Language Syntax

Regular Tree Grammar

$$
\begin{aligned}
Start &::= While\ B\ do\ S \\
B &::= E < E \\
E &::= x \mid y \mid E \,\&\, E \mid (E \mid E) \\
S &::= S; S \mid x \coloneqq E \mid y \coloneqq E
\end{aligned}
$$

Language Semantics

Constrained Horn Clauses

$$\frac{Sem_E(\Gamma, t_e, v)\ \Gamma_1 = \Gamma[x \mapsto v]}{Sem_S(\Gamma, x \coloneqq t_e, \Gamma_1)}$$

Problem Specification

Constraints / Examples

Examples:

$f(6, 9) = (15, \_)$

Semantics-Guided Synthesis Problem

Language Syntax

Regular Tree Grammar

$Start ::= While\ B\ do\ S$
$B ::= E < E$
$E ::= x \mid y \mid E\ \&\ E \mid (E \mid E)$
$S ::= S; S \mid x := E \mid y := E$

Language Semantics

Constrained Horn Clauses

$$\frac{Sem_E(\Gamma, t_e, v)\quad \Gamma_1 = \Gamma[x \mapsto v]}{Sem_S(\Gamma, x := t_e, \Gamma_1)}$$

Problem Specification

Constraints / Examples

Examples:
$f(6, 9) = (15, \_)$

$$\frac{\boxed{t \in L(Start)}\quad Sem_{Start}\big((6,9), t, (15, \_)\big)}{Realizable}$$

Solving a Semgus problem ⇒ Solving a query over CHCs!

Semantics-Guided Synthesis Problem

Language Syntax

Regular Tree Grammar

$$
\begin{aligned}
&Start ::= While\ B\ do\ S \\
&B ::= E < E \\
&E ::= x \mid y \mid E\ \&\ E \mid (E \mid E) \\
&S ::= S; S \mid x \coloneqq E \mid y \coloneqq E
\end{aligned}
$$

Language Semantics

Constrained Horn Clauses

$$\frac{Sem_E(\Gamma, t_e, v)\quad \Gamma_1 = \Gamma[x \mapsto v]}{Sem_S(\Gamma, x \coloneqq t_e, \Gamma_1)}$$

Problem Specification

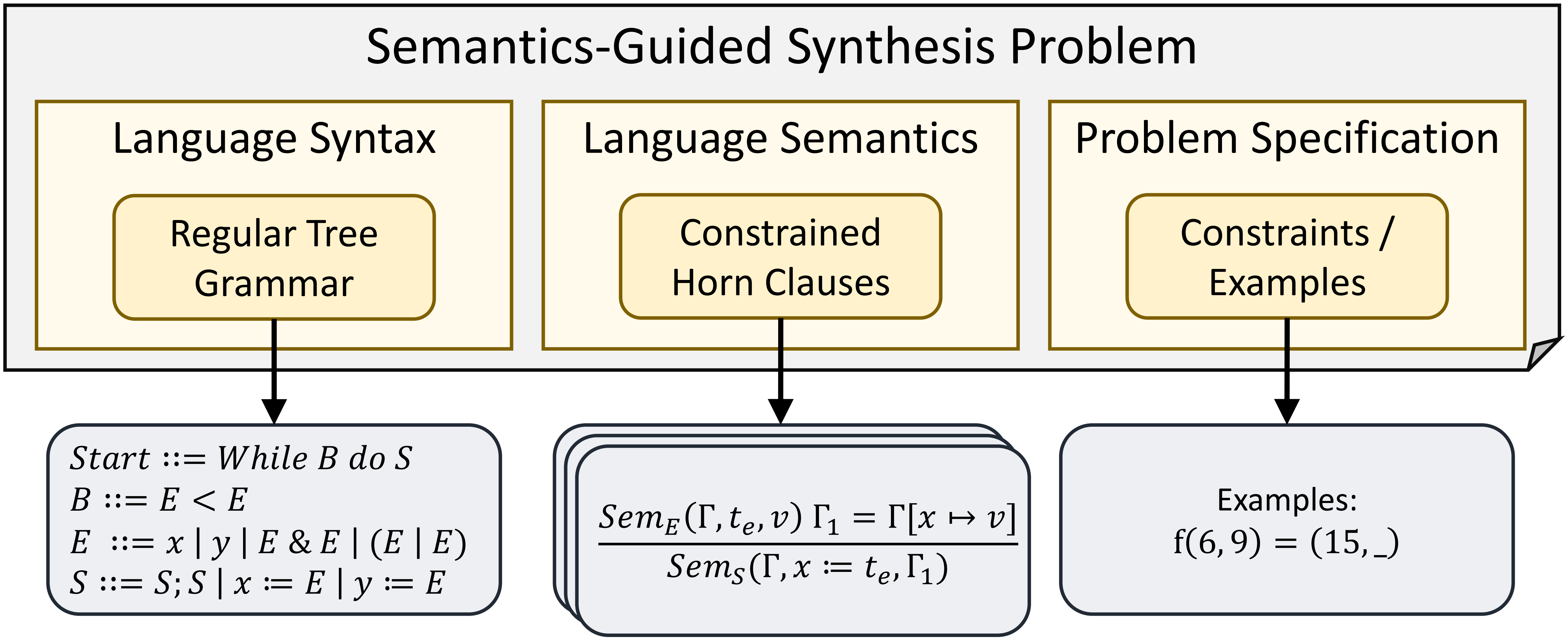
Constraints / Examples

Examples:

$f(6, 9) = (15, \_)$

$$\frac{t \in L(Start) \qquad \boxed{Sem_{Start}}\big((6,9), t, (15, \_)\big)}{Realizable}$$

Solving a Semgus problem ⇒ Solving a query over CHCs!



$$\frac{\boxed{Syn_{Start}(t)} \quad Sem_{Start}\big((6,9), t, (15, \_)\big)}{Realizable}$$

Solving a Semgus problem ⇒ Solving a query over CHCs!

$$S ::= S; S$$

$S ::= S; S$

Given $t_1 \in L(S)$ and $t_2 \in L(S)$:

$t_1; t_2 \in L(S)$

$S ::= S; S$

Given $t_1 \in L(S)$ and $t_2 \in L(S)$:
$t_1; t_2 \in L(S)$

$$\boxed{\begin{gathered} Syn_S : Term \rightarrow Boolean \\ Syn_S(t) \Leftrightarrow t \in L(S) \end{gathered}}$$

$$S ::= S; S$$

Given $t_1 \in L(S)$ and $t_2 \in L(S)$:
$t_1; t_2 \in L(S)$

$$Syn_S: Term \to Boolean$$
$$Syn_S(t) \Leftrightarrow t \in L(S)$$

$$\boxed{\frac{Syn_S(t_1) \quad Syn_S(t_2)}{Syn_S(t_1; t_2)}}$$

Syntax:

$Start ::= While\ B\ do\ S$

$B ::= E < E$

$E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)$

$S ::= S; S \mid x := E \mid y := E$

Semantics:

$$\frac{Sem_E(\Gamma, t_e, v) \quad \Gamma_1 = \Gamma[x \mapsto v]}{Sem_S(\Gamma, x := t_e, \Gamma_1)}$$

$\dots$

$$\frac{\boxed{Syn_{Start}(t)} \quad Sem_{Start}\big((6,9), t, (15, \_)\big)}{Realizable}$$

Specification: $[\mathrm{f}(6,9) = (15, \_)]$

Syntax:

$Start ::= While\ B\ do\ S$

$B ::= E < E$

$E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)$

$S ::= S; S \mid x := E \mid y := E$

Semantics:

$$\frac{Sem_E(\Gamma, t_e, v) \quad \Gamma_1 = \Gamma[x \mapsto v]}{Sem_S(\Gamma, x := t_e, \Gamma_1)}$$

$$\cdots$$

$$\frac{Syn_{Start}(t) \quad \boxed{Sem_{Start}}\big((6,9), t, (15, \_)\big)}{Realizable}$$

Specification: $[\mathrm{f}(6,9) = (15, \_)]$

Syntax:

$Start ::= While\ B\ do\ S$

$B ::= E < E$

$E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)$

$S ::= S; S \mid x := E \mid y := E$

Semantics:

$$\frac{Sem_E(\Gamma, t_e, v) \quad \Gamma_1 = \Gamma[x \mapsto v]}{Sem_S(\Gamma, x := t_e, \Gamma_1)}$$

$$\ldots$$

$$\frac{Syn_{Start}(t) \quad Sem_{Start}\boxed{((6,9), t, (15, \_))}}{Realizable}$$

Specification: $[\mathrm{f}(6,9) = (15, \_)]$

Program Synthesis: Proving *Realizable*

$$\frac{Syn_{Start}(t) \quad Sem_{Start}\big((6,9), t, (15,\_)\big)}{Realizable}$$

Specification: $[\mathrm{f}(6,9) = (15,\_)]$
Valid solution $t = While\ x < y\ do\ x := x \mid y$

Program Synthesis: Proving $\boldsymbol{Realizable}$

$$\frac{Syn_{Start}(t) \quad Sem_{Start}\big((6,9), t, (15,\_)\big)}{Realizable}$$

Specification: $[\mathrm{f}(6,9) = (15,\_)]$

Valid solution $t = While\ x < y\ do\ x := x \mid y$

Program Synthesis: Proving *Realizable*

$$\frac{Syn_B(x < y) \quad Syn_S(x \coloneqq x \mid y)}{Syn_{Start}(t)}$$

$$\uparrow$$

$$\frac{Syn_{Start}(t) \quad Sem_{Start}\big((6,9), t, (15, \_)\big)}{Realizable}$$

Specification: $[\mathrm{f}(6,9) = (15, \_)]$
Valid solution $t = While\ x < y\ do\ x \coloneqq x \mid y$

Program Synthesis: Proving *Realizable*

$$\frac{Syn_B(x < y) \quad Syn_S(x := x \mid y)}{Syn_{Start}(t)} \longleftarrow \boxed{\frac{Syn_B(b) \quad Syn_S(s)}{Syn_{Start}(While\ b\ do\ s)}}$$

$$\uparrow$$

$$\frac{Syn_{Start}(t) \quad Sem_{Start}\big((6,9), t, (15, \_)\big)}{Realizable}$$

Specification: $[\mathrm{f}(6,9) = (15, \_)]$
Valid solution $t = While\ x < y\ do\ x := x \mid y$

Program Synthesis: Proving *Realizable*

$$\frac{Syn_B(E) \quad Syn_S(E)}{Syn_B(E < E)}$$

$$\cfrac{\cfrac{Syn_E(x) \quad Syn_E(y)}{Syn_B(x < y)} \quad Syn_S(x := x \mid y)}{Syn_{Start}(t)}$$

$$\frac{Syn_{Start}(t) \quad Sem_{Start}((6,9), t, (15, \_))}{Realizable}$$

Specification: $[\mathrm{f}(6,9) = (15, \_)]$
Valid solution $t = While\ x < y\ do\ x := x \mid y$

Program Synthesis: Proving *Realizable*

$$\dfrac{\dfrac{Syn_E(x) \quad Syn_E(y)}{Syn_B(x < y)} \qquad \dfrac{\dfrac{Syn_E(x) \quad Syn_E(y)}{Syn_E(x \mid y)}}{Syn_S(x := x \mid y)}}{Syn_{Start}(t)}$$

$$\uparrow$$

$$\dfrac{Syn_{Start}(t) \quad Sem_{Start}\big((6,9), t, (15, \_)\big)}{Realizable}$$

Specification: $[\mathrm{f}(6,9) = (15, \_)]$

Valid solution $t = While\ x < y\ do\ x := x \mid y$

Program Synthesis: Proving *Realizable*

$$\frac{}{Sem_{Start}((6,9), t, (15, \_))}$$

$$\uparrow$$

$$\frac{Syn_{Start}(t) \quad Sem_{Start}((6,9), t, (15, \_))}{Realizable}$$

Specification: $[\mathrm{f}(6,9) = (15, \_)]$
Valid solution $t = While\ x < y\ do\ x := x \mid y$

Program Synthesis: Proving *Realizable*

$$\frac{Sem_B(\Gamma, t_b, v) \quad v = True \quad Sem_S(\Gamma, t_s, \Gamma_1) \quad Sem_{Start}(\Gamma_1, While\ t_b\ do\ t_s, \Gamma_2)}{Sem_{Start}(\Gamma, While\ t_b\ do\ t_s, \Gamma_2)}$$

$$\frac{Sem_B\big((6,9), x < y, True\big) \quad Sem_S\big((6,9), x \coloneqq x \mid y, (15,9)\big) \quad \cdots}{Sem_{Start}\big((6,9), t, (15, \_)\big)}$$

$$\frac{Syn_{Start}(t) \quad Sem_{Start}\big((6,9), t, (15, \_)\big)}{Realizable}$$

Specification: $[f(6,9) = (15, \_)]$

Valid solution $t = While\ x < y\ do\ x \coloneqq x \mid y$

Program Synthesis: Proving *Realizable*

$$\dfrac{\dfrac{Sem_E\big((6,9),x,6\big) \quad Sem_E\big((6,9),y,9\big)}{Sem_B\big((6,9),x<y,True\big)} \qquad \dfrac{\cdots}{Sem_S\big((6,9),x \coloneqq x \mid y,(15,9)\big)} \qquad \cdots}{Sem_{Start}\big((6,9),t,(15,\_)\big)}$$

$\uparrow$

$$\dfrac{Syn_{Start}(t) \quad Sem_{Start}\big((6,9),t,(15,\_)\big)}{Realizable}$$

Specification: $[\mathrm{f}(6,9) = (15,\_)]$
Valid solution $t = While\ x < y\ do\ x \coloneqq x \mid y$

Program Synthesis: Proving *Realizable*

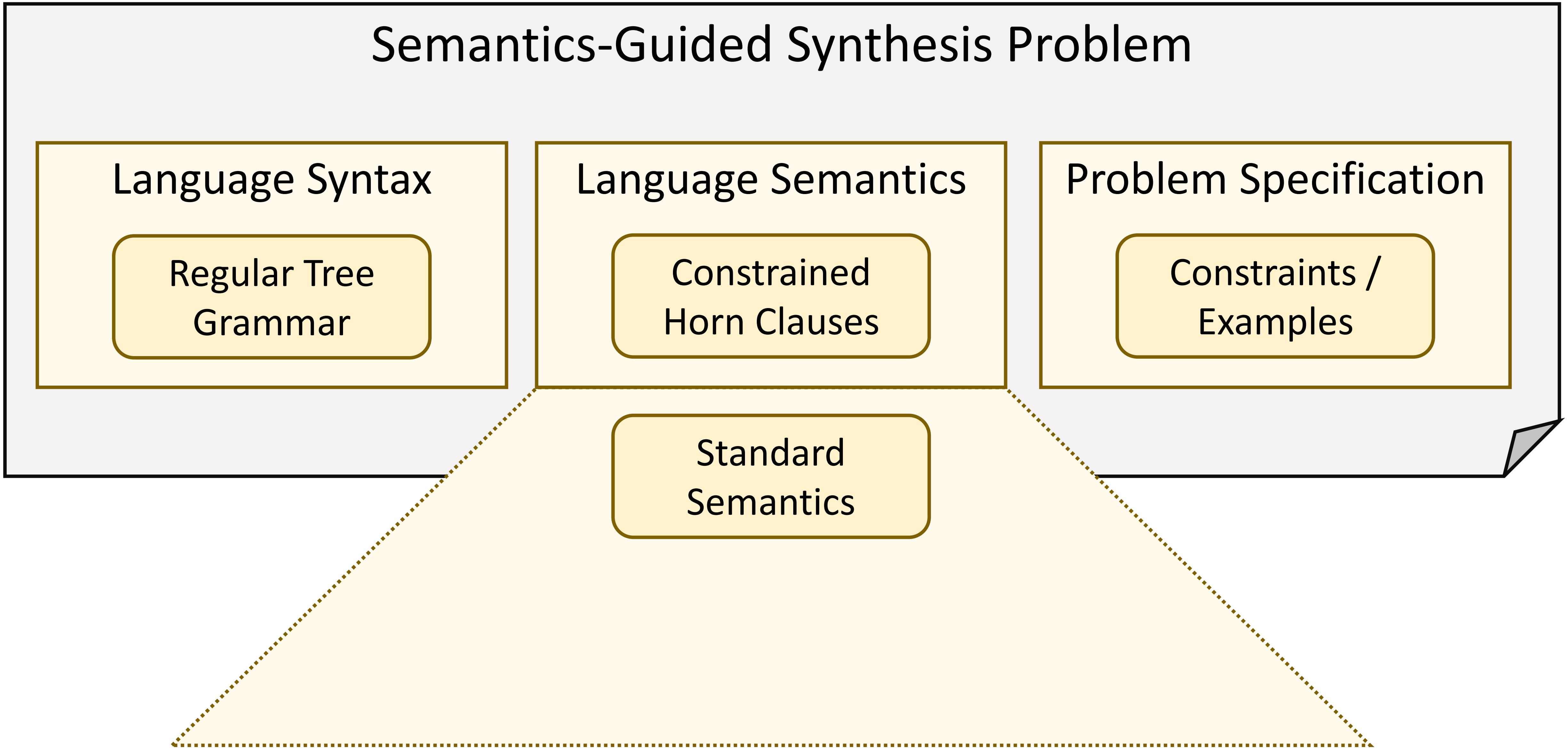
Proof tree contains term t: Program synthesized!

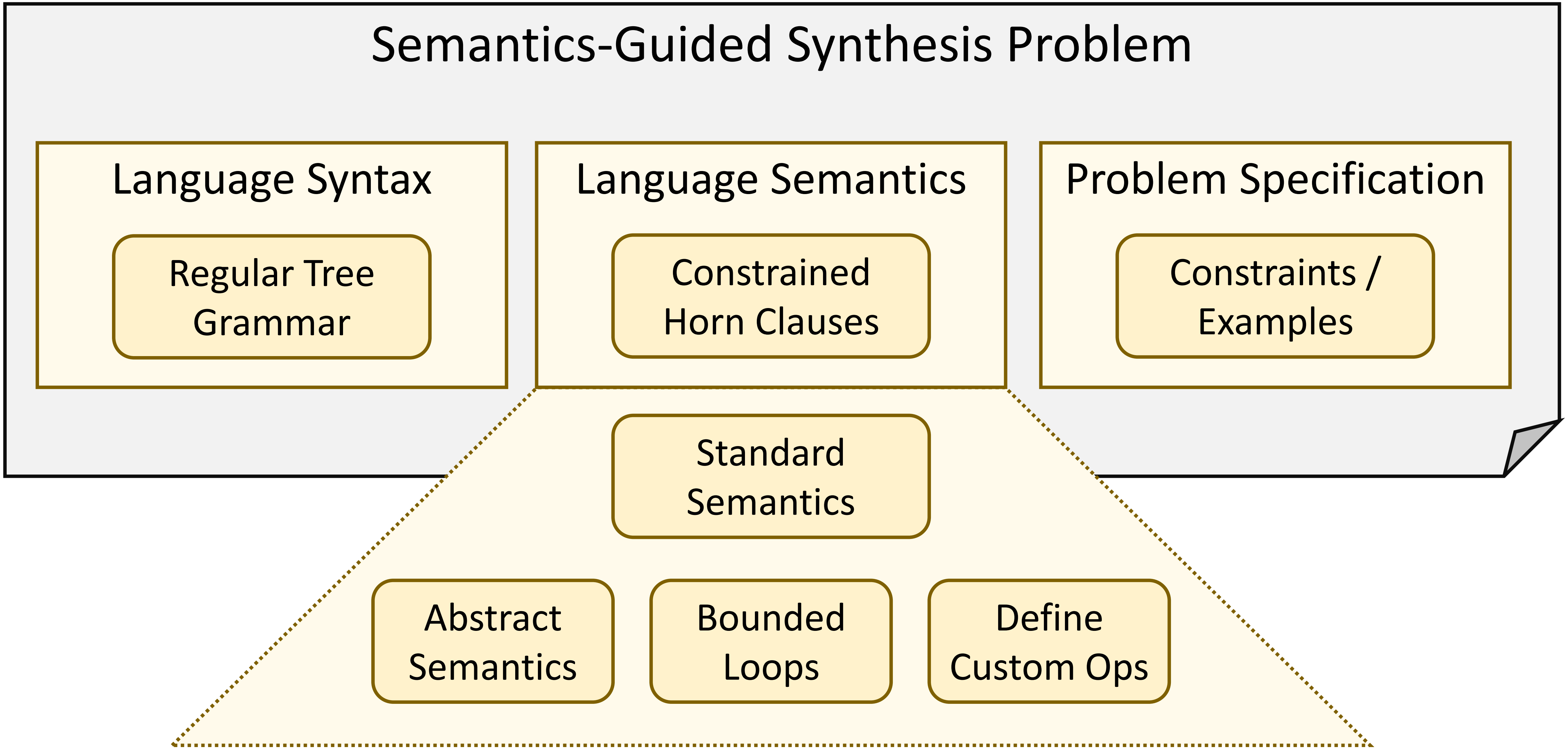
$$\dfrac{\dfrac{Sem_E\big((6,9),x,6\big) \quad Sem_E\big((6,9),y,9\big)}{Sem_B\big((6,9),x<y,True\big)} \quad \dfrac{\cdots}{Sem_S((6,9),x := x \mid y,(15{,}9))} \quad \cdots}{Sem_{Start}\big((6,9),t,(15,\_)\big)}$$

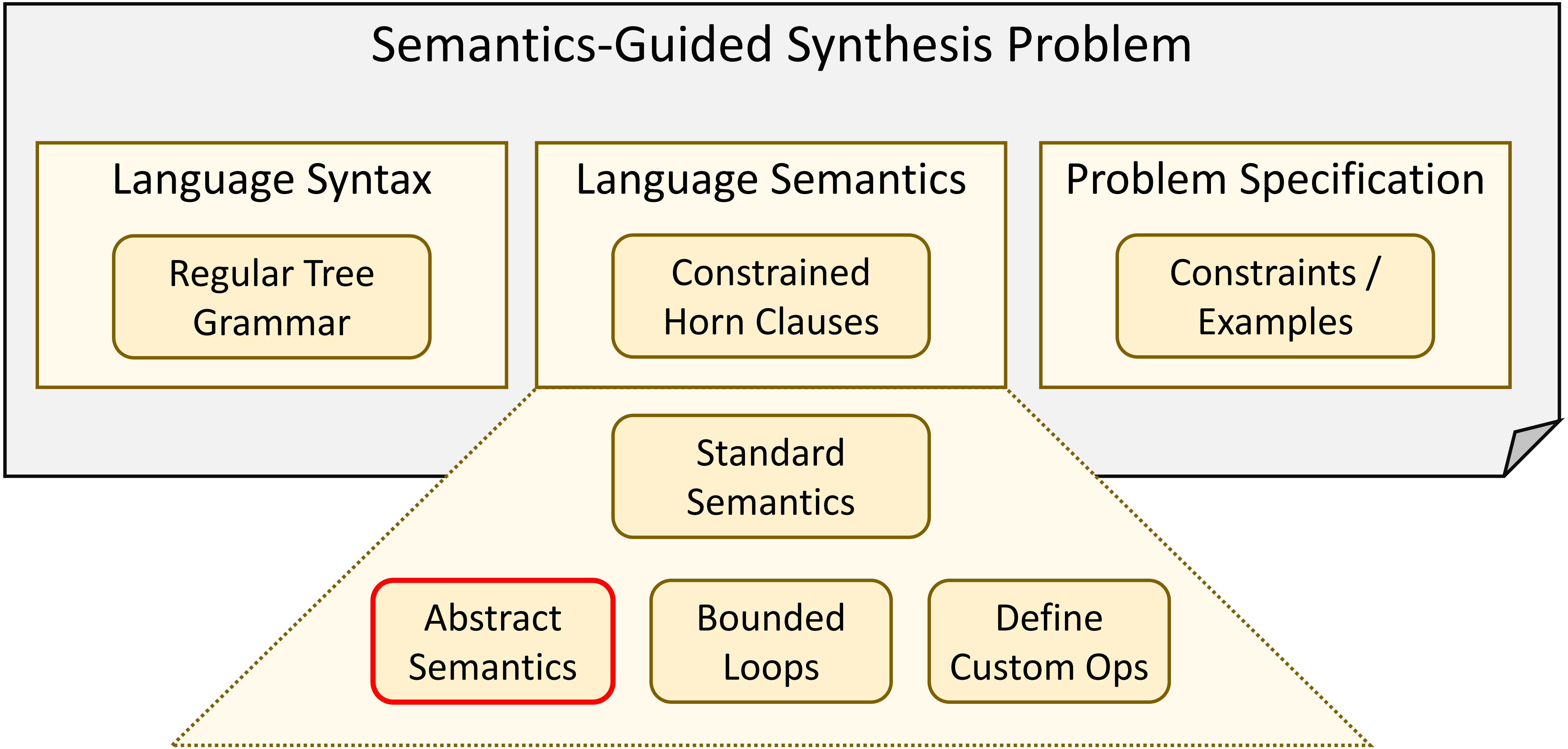
$$\dfrac{Syn_{Start}(t) \quad Sem_{Start}\big((6,9),t,(15,\_)\big)}{Realizable}$$

Specification: $[\mathrm{f}(6,9) = (15,\_)]$
Valid solution $t = While\ x < y\ do\ x := x \mid y$

1. Semantics-Guided Synthesis
2. Solving a Semgus Problem
3. $\boxed{\text{Alternative Semantics for Semgus}}$
4. Result Highlights


Semantics-Guided Synthesis Problem
Language Syntax
Regular Tree Grammar
Language Semantics
Constrained Horn Clauses
Problem Specification
Constraints / Examples
Standard Semantics


Semantics-Guided Synthesis Problem
Language Syntax
Regular Tree Grammar
Language Semantics
Constrained Horn Clauses
Problem Specification
Constraints / Examples
Standard Semantics
Abstract Semantics
Bounded Loops
Define Custom Ops



Wang, et al. POPL17

$x \,\&\, y$

$$\frac{[\![\, x \,]\!](\Gamma) = v_x \quad [\![\, y \,]\!](\Gamma) = v_y \quad v = v_x \,\&\, v_y}{[\![\, x\&y \,]\!](\Gamma) = v}$$

$x \,\&\, y$

$$\frac{⟦\, x \,⟧(\Gamma) = v_x \quad ⟦\, y \,⟧(\Gamma) = v_y \quad v = v_x \,\&\, v_y}{⟦\, x\&y \,⟧(\Gamma) = v}$$

Abstract Domain: Consider only first bit

$x \,\&\, y$

$$\frac{[\![\, x \,]\!](\Gamma) = v_x \quad [\![\, y \,]\!](\Gamma) = v_y \quad v = v_x \,\&\, v_y}{[\![\, x\&y \,]\!](\Gamma) = v}$$

Abstract Domain: Consider only first bit

$$\frac{[\![\, x \,]\!](\Gamma) = b_x \quad [\![\, y \,]\!](\Gamma) = b_y \quad b = if\ (b_x = \top \vee b_y = \top)\ \top\ else\ b_x \,\&\, b_y}{[\![\, x\&y \,]\!](\Gamma) = b}$$

$x \,\&\, y$

$$\frac{[\![\, x \,]\!](\Gamma) = v_x \quad [\![\, y \,]\!](\Gamma) = v_y \quad v = v_x \,\&\, v_y}{[\![\, x\&y \,]\!](\Gamma) = v}$$

Abstract Domain: Consider only first bit

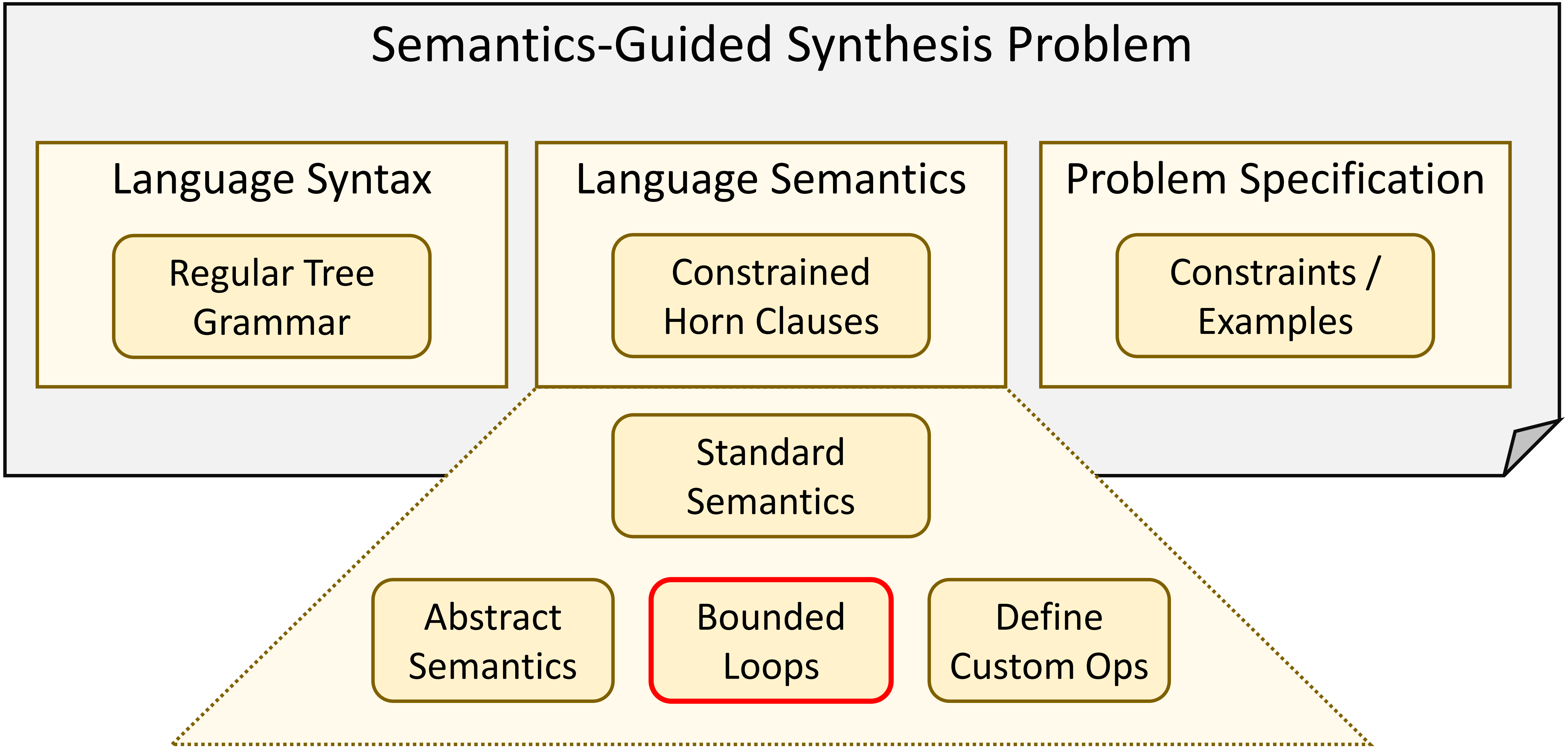
$$\frac{Sem_E(\Gamma, x, b_x) \quad Sem_E(\Gamma, y, b_y) \quad b = if\ (b_x = \top \vee b_y = \top)\ \top\ else\ b_x \,\&\, b_y}{Sem_S(\Gamma, x\&y, b)}$$

$x \,\&\, y$

$$\frac{[\![\, x \,]\!](\Gamma) = v_x \quad [\![\, y \,]\!](\Gamma) = v_y \quad v = v_x \,\&\, v_y}{[\![\, x\&y \,]\!](\Gamma) = v}$$

Problem S has no solution using abstract semantics $\Rightarrow$
S has no solution using standard semantics

$$\frac{Sem_E(\Gamma, x, b_x) \quad Sem_E(\Gamma, y, b_y) \quad b = if\ (b_x = \top \vee b_y = \top)\ \top\ else\ b_x \,\&\, b_y}{Sem_S(\Gamma, x\&y, b)}$$



Sketch, Rosette

$While\ b\ do\ s$

$$\frac{[\![\, b \,]\!](\Gamma) = True \quad [\![\, s \,]\!](\Gamma) = \Gamma_1 \quad [\![\, While\ b\ do\ s \,]\!](\Gamma_1) = \Gamma_2}{[\![\, While\ b\ do\ s \,]\!](\Gamma) = \Gamma_2}$$

$While\ b\ do\ s$

$$\frac{\llbracket\, b\, \rrbracket(\Gamma) = True \qquad \llbracket\, s\, \rrbracket(\Gamma) = \Gamma_1 \qquad \llbracket\, While\ b\ do\ s\, \rrbracket(\Gamma_1) = \Gamma_2}{\llbracket\, While\ b\ do\ s\, \rrbracket(\Gamma) = \Gamma_2}$$

Bound Loops to k iterations

$While\ b\ do\ s$

$$\frac{[\![b]\!](\Gamma) = True \quad [\![s]\!](\Gamma) = \Gamma_1 \quad [\![While\ b\ do\ s]\!](\Gamma_1) = \Gamma_2}{[\![While\ b\ do\ s]\!](\Gamma) = \Gamma_2}$$

Bound Loops to k iterations

$$\frac{[\![b]\!](\Gamma) = True \quad k > 0 \quad [\![s]\!](\Gamma) = \Gamma_1 \quad [\![While\ b\ do\ s]\!](\Gamma_1, k-1) = \Gamma_2}{[\![While\ b\ do\ s]\!](\Gamma, k) = \Gamma_2}$$

$While\ b\ do\ s$

$$\frac{[\![b]\!](\Gamma) = True \quad [\![s]\!](\Gamma) = \Gamma_1 \quad [\![While\ b\ do\ s]\!](\Gamma_1) = \Gamma_2}{[\![While\ b\ do\ s]\!](\Gamma) = \Gamma_2}$$

Bound Loops to k iterations

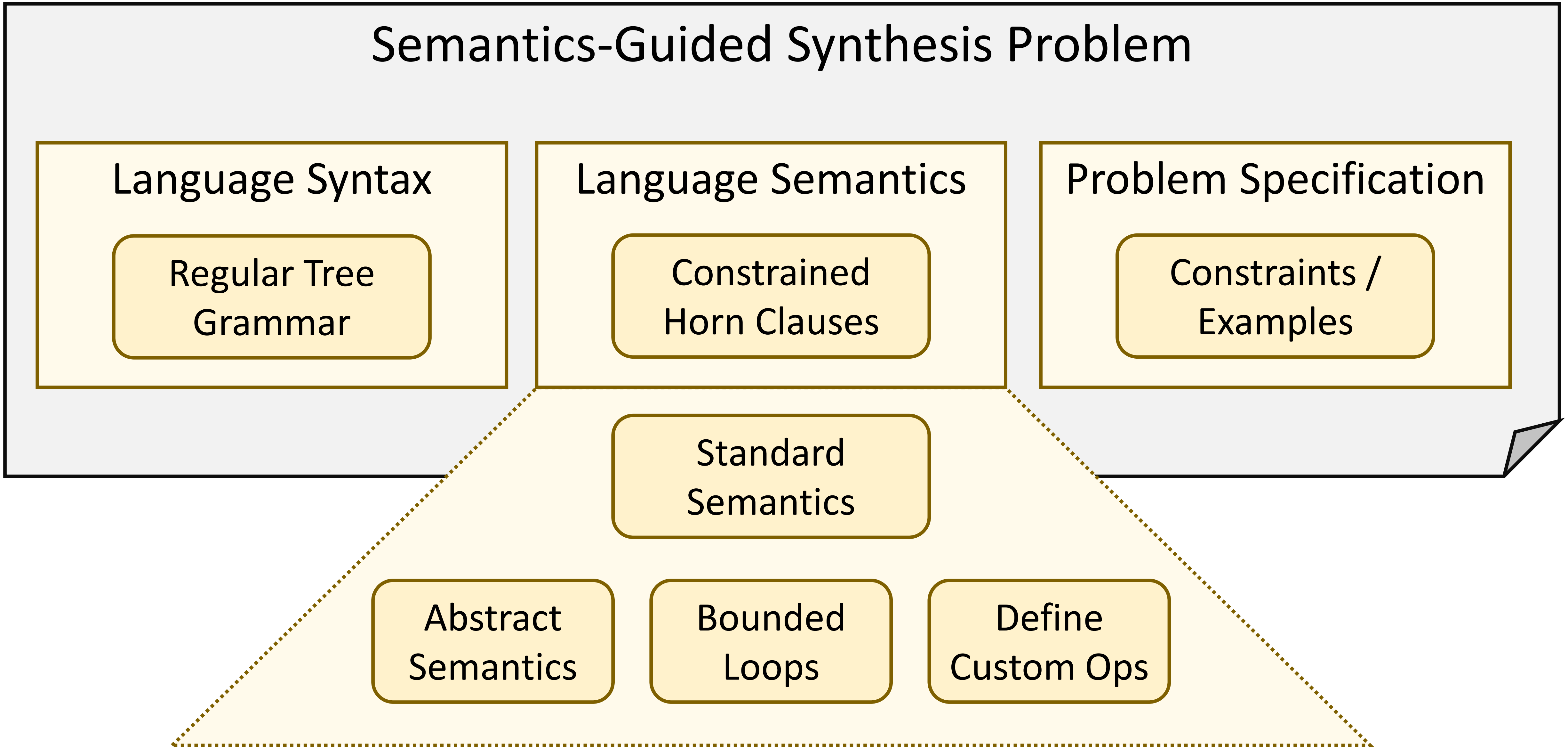
$$\frac{Sem_B(\Gamma, b, True) \quad k > 0 \quad Sem_S(\Gamma, s, \Gamma_1) \quad Sem_S(\Gamma_1, While\ b\ do\ s, \Gamma_2, k-1)}{Sem_S(\Gamma, While\ b\ do\ s, \Gamma_2, k)}$$

$While\ b\ do\ s$

$$\frac{[\![\, b \,]\!](\Gamma) = True \qquad [\![\, s \,]\!](\Gamma) = \Gamma_1 \qquad [\![\, While\ b\ do\ s \,]\!](\Gamma_1) = \Gamma_2}{[\![\, While\ b\ do\ s \,]\!](\Gamma) = \Gamma_2}$$

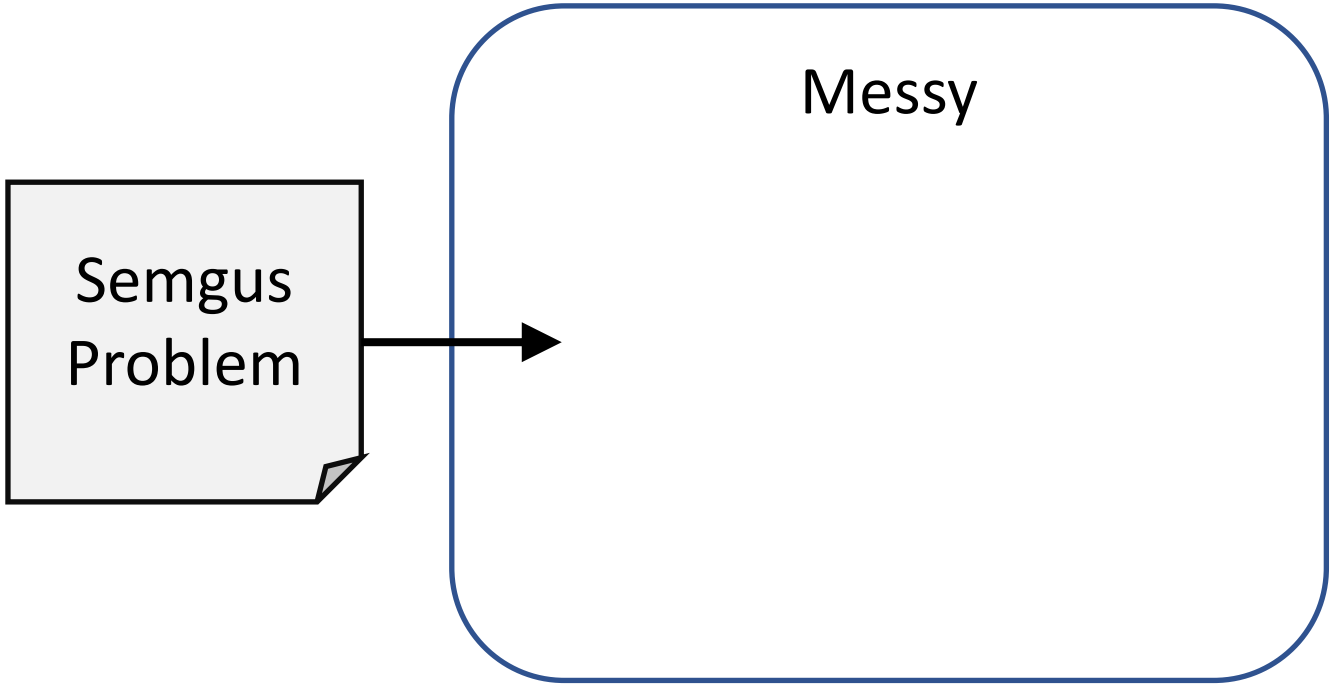
Term t is a solution using bounded loops $\Rightarrow$
t is a solution using standard semantics

$$\frac{Sem_B(\Gamma, b, True\,) \qquad k > 0 \qquad Sem_S(\Gamma, s, \Gamma_1) \qquad Sem_S(\Gamma_1, While\ b\ do\ s, \Gamma_2, k-1)}{Sem_S(\Gamma, While\ b\ do\ s, \Gamma_2, k)}$$

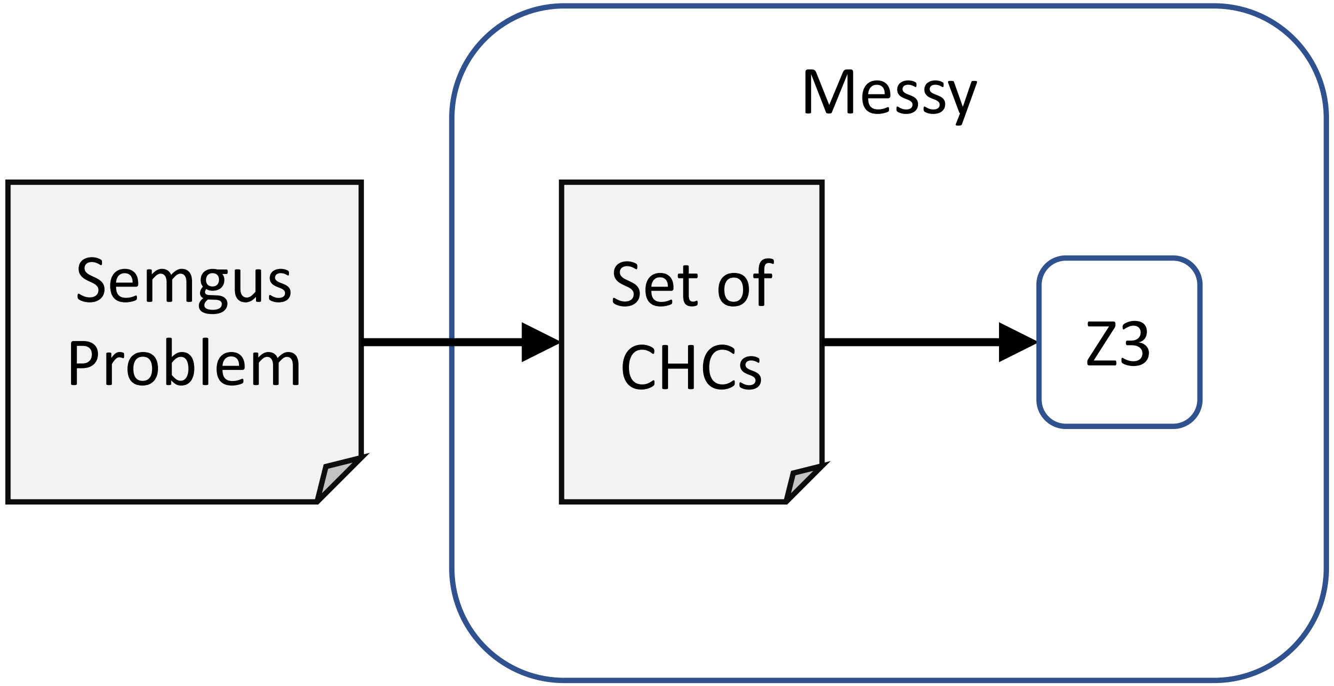

Semantics-Guided Synthesis Problem
Language Syntax
Regular Tree Grammar
Language Semantics
Constrained Horn Clauses
Problem Specification
Constraints / Examples
Standard Semantics
Abstract Semantics
Bounded Loops
Define Custom Ops

1. Semantics-Guided Synthesis
2. Solving a Semgus Problem
3. Alternative Semantics for Semgus
4. $\boxed{\text{Result Highlights}}$

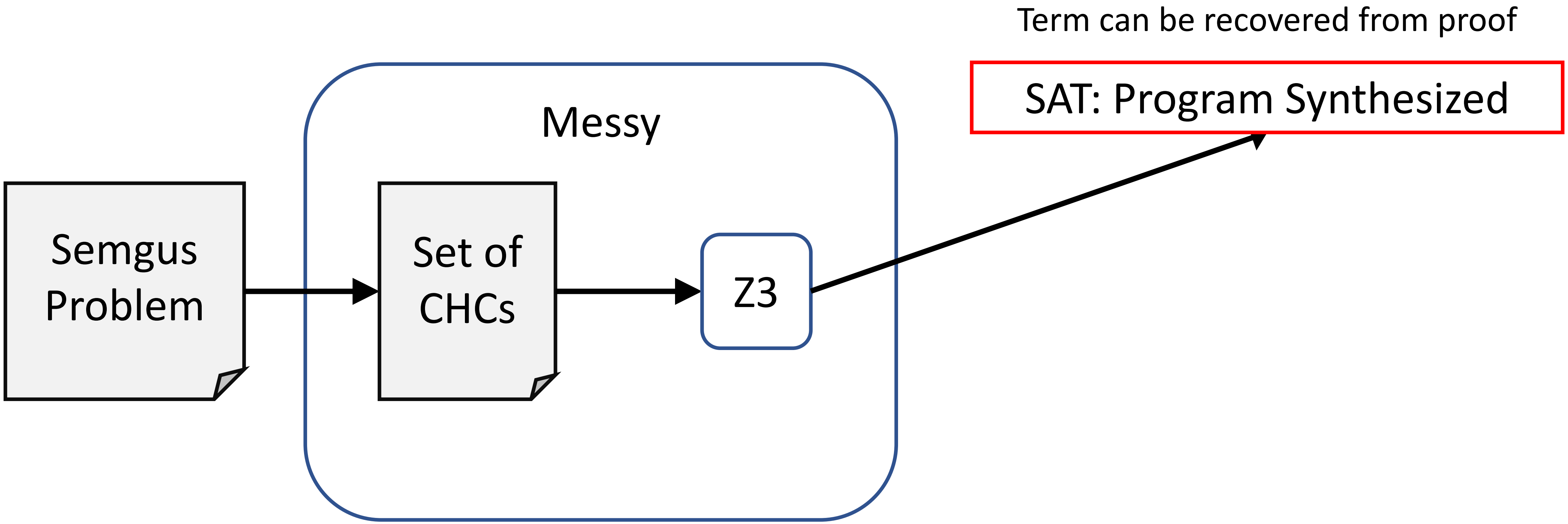
The Architecture of Messy



The Architecture of Messy

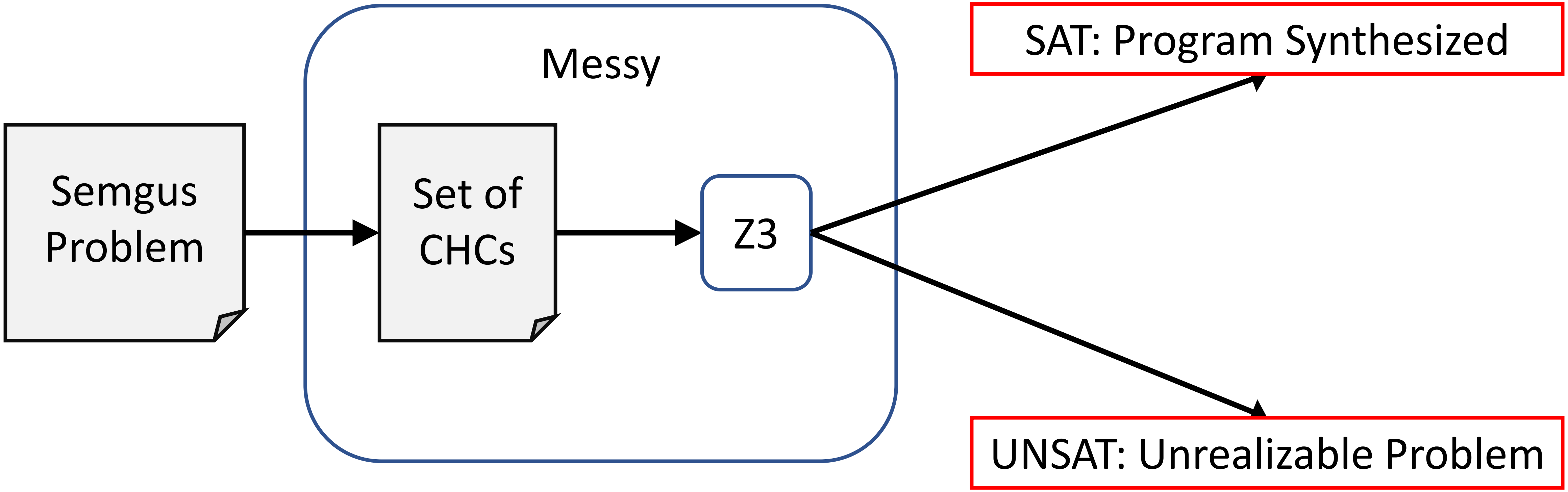


The Architecture of Messy



The Architecture of Messy

Term can be recovered from proof



Unrealizability:
When a synthesis problem has no solution within the given syntax

Benchmarks

60 LIA SyGuS Benchmarks

- From SyGuS-Comp

132 Unrealizable LIA Benchmarks

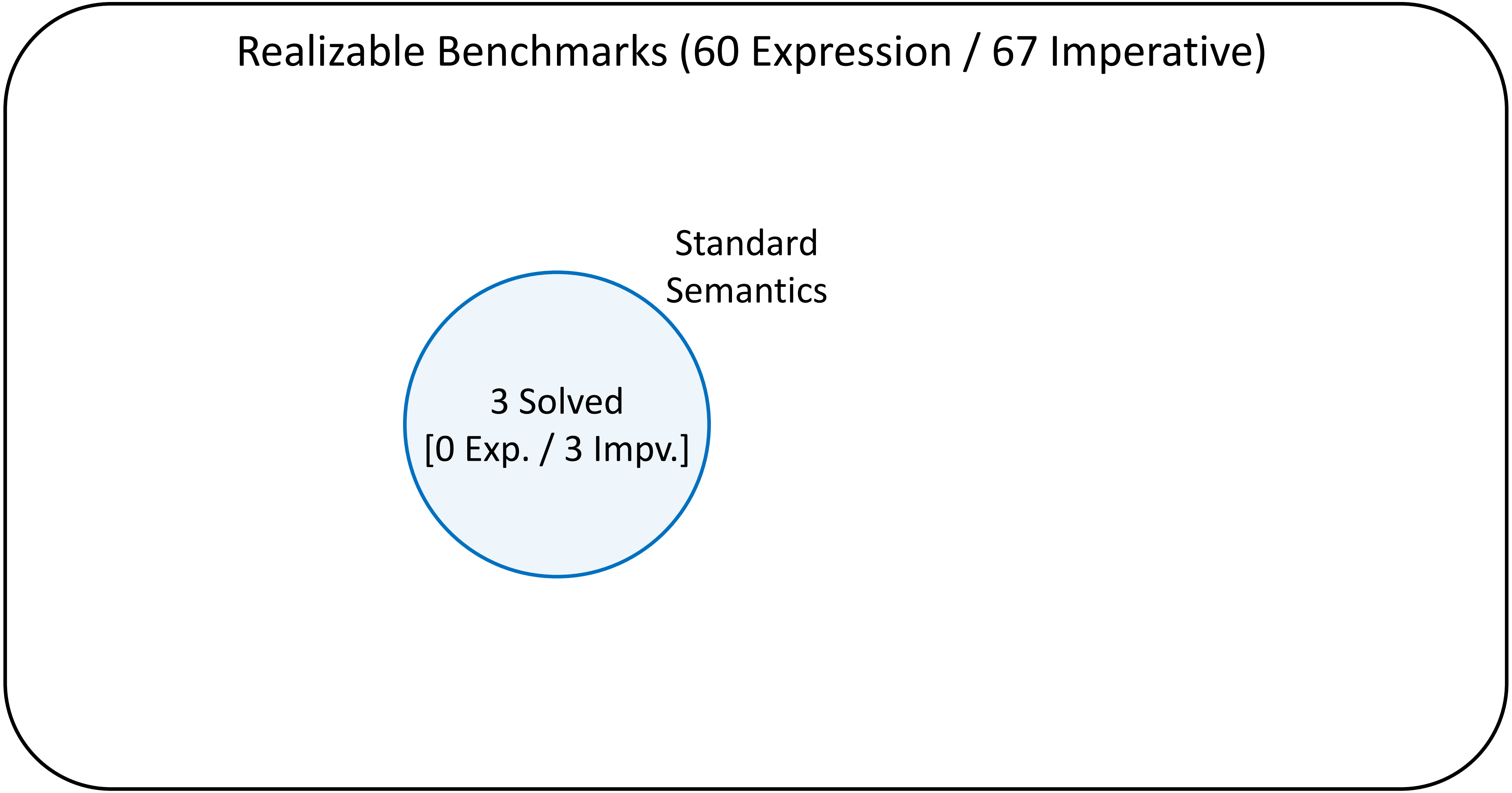
- From Hu, et al. CAV2019

67 Realizable Imperative Benchmarks

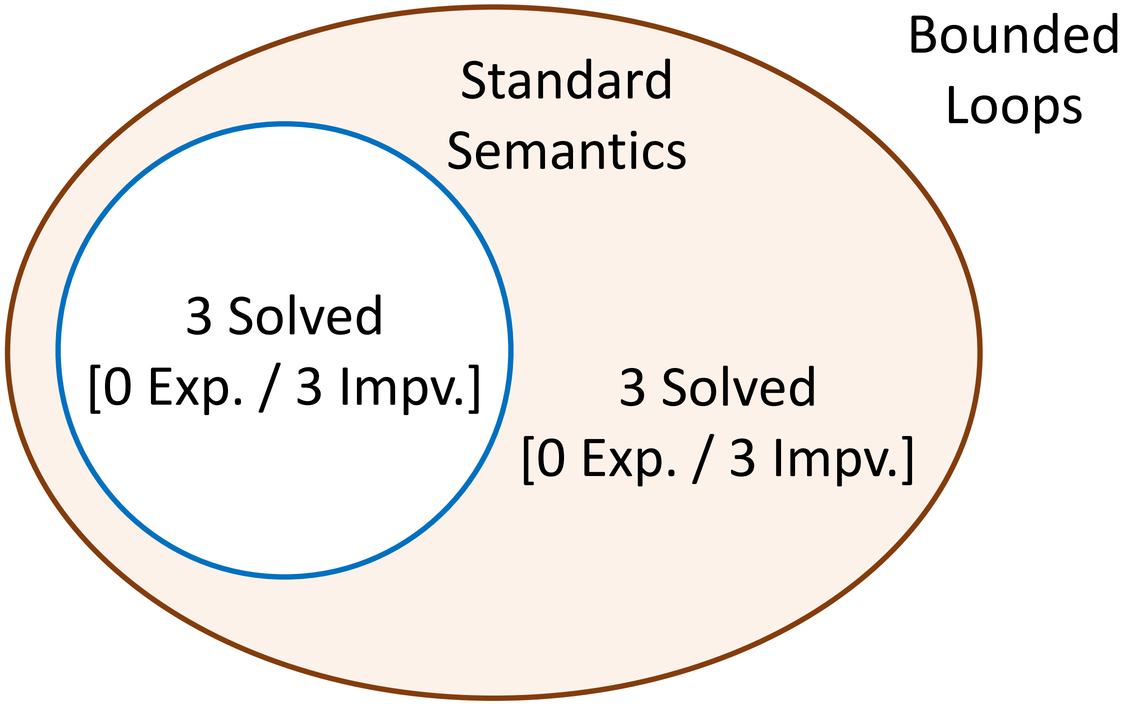
- Some written by hand, some from So, et al. SAS2017

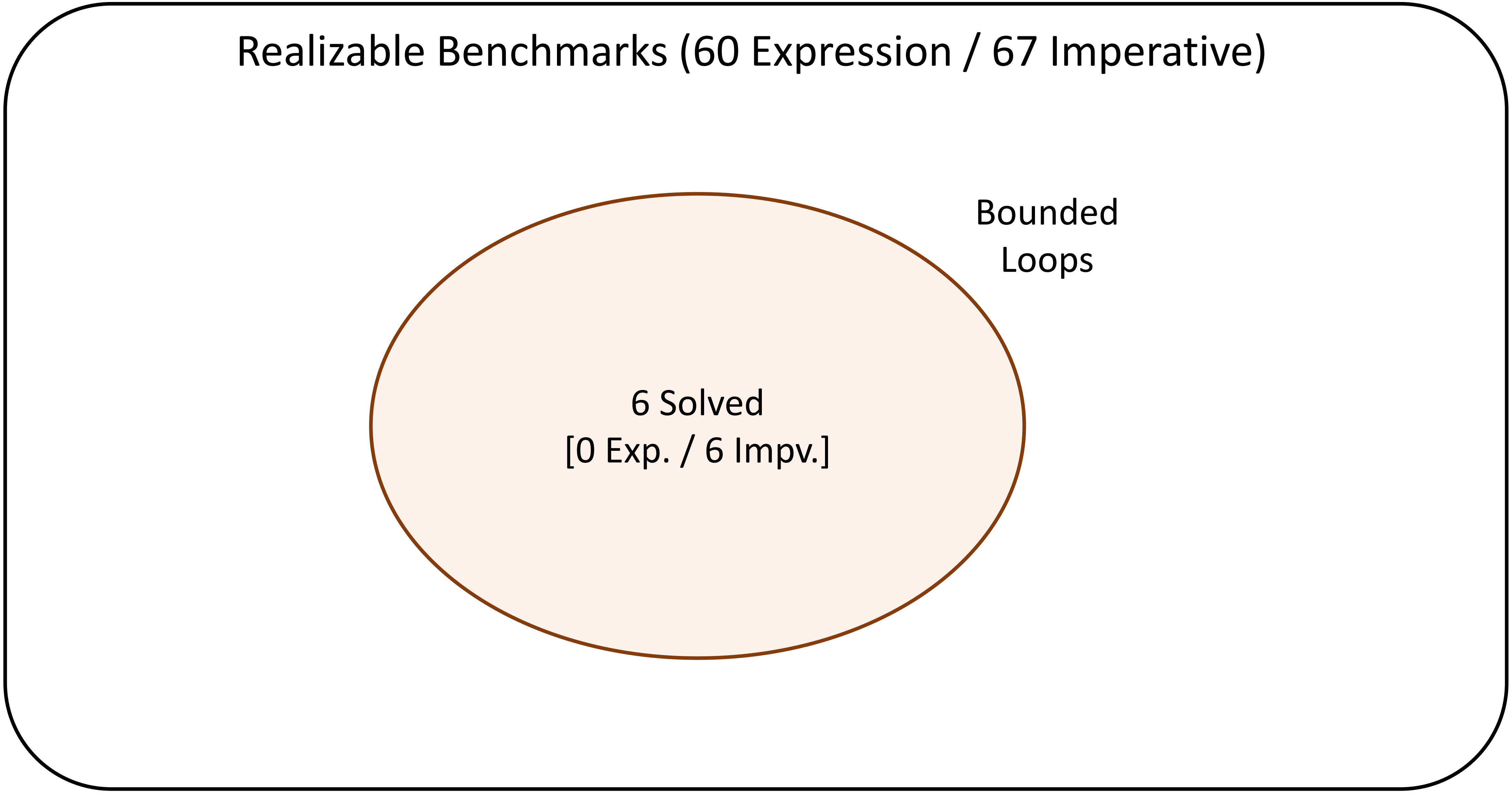
222 Unrealizable Imperative Benchmarks

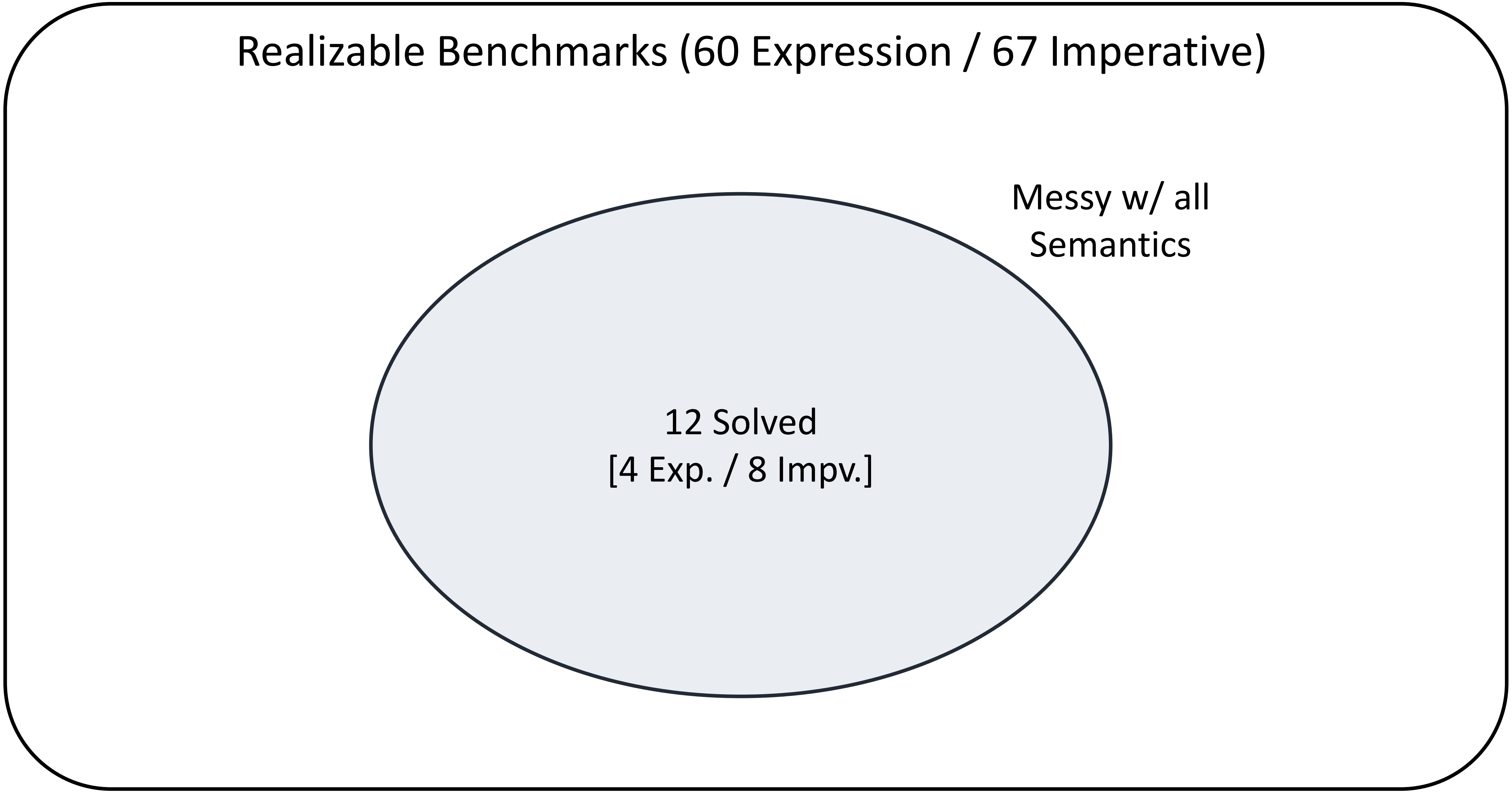
- Handwritten variants of the 67 realizable benchmarks

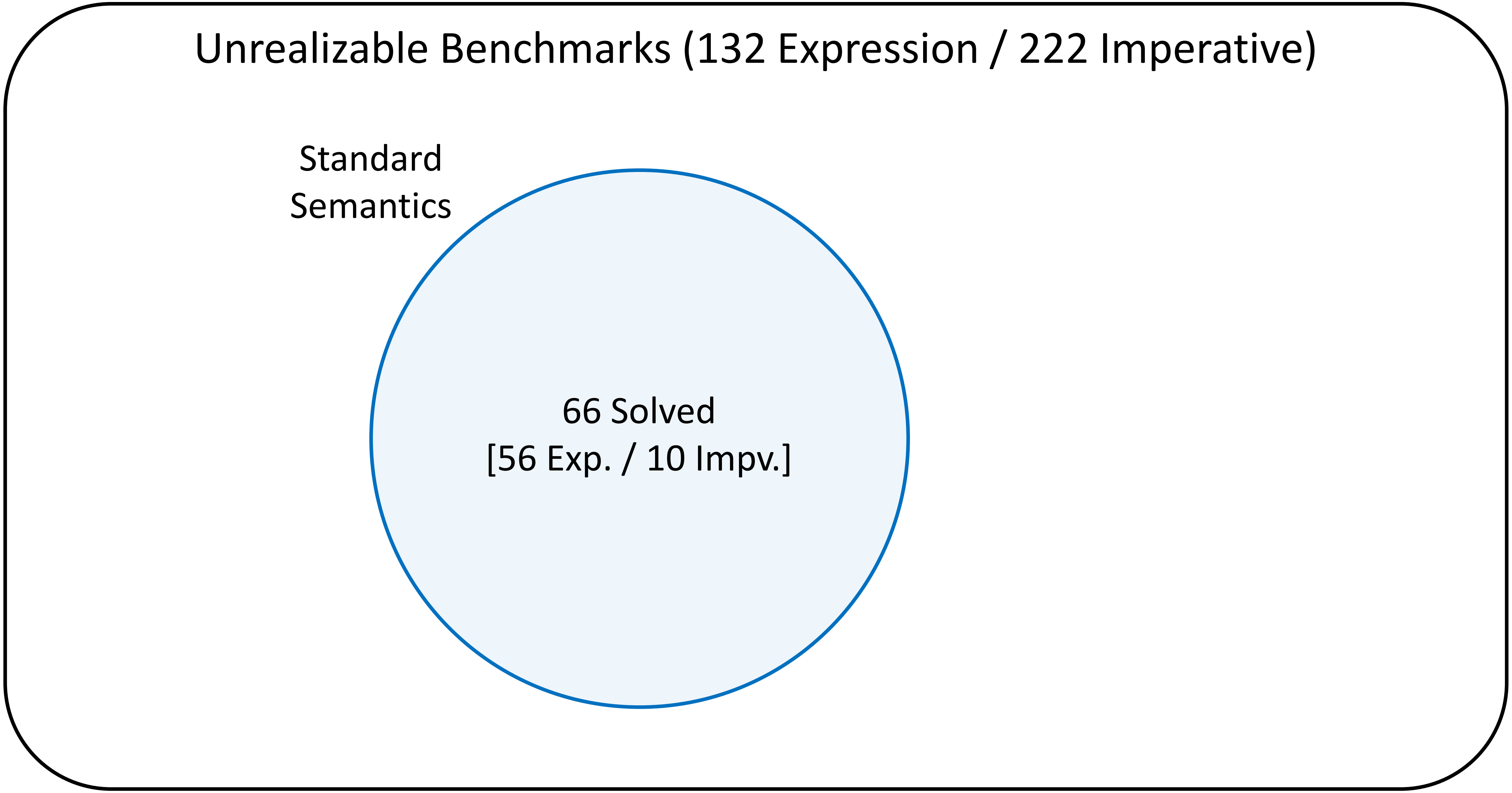

Realizable Benchmarks (60 Expression / 67 Imperative)
Standard
Semantics
3 Solved
[0 Exp. / 3 Impv.]

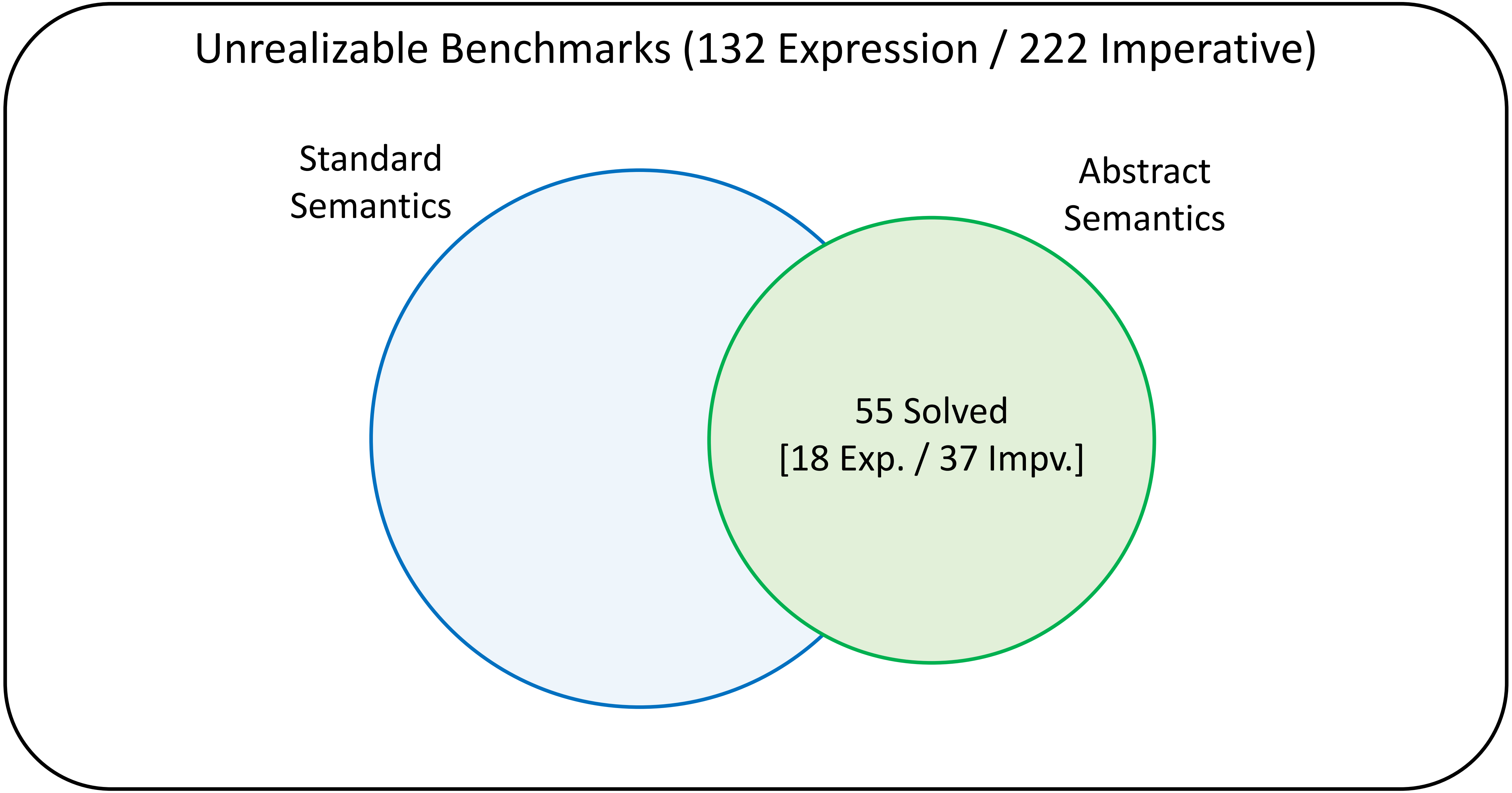
Realizable Benchmarks (60 Expression / 67 Imperative)



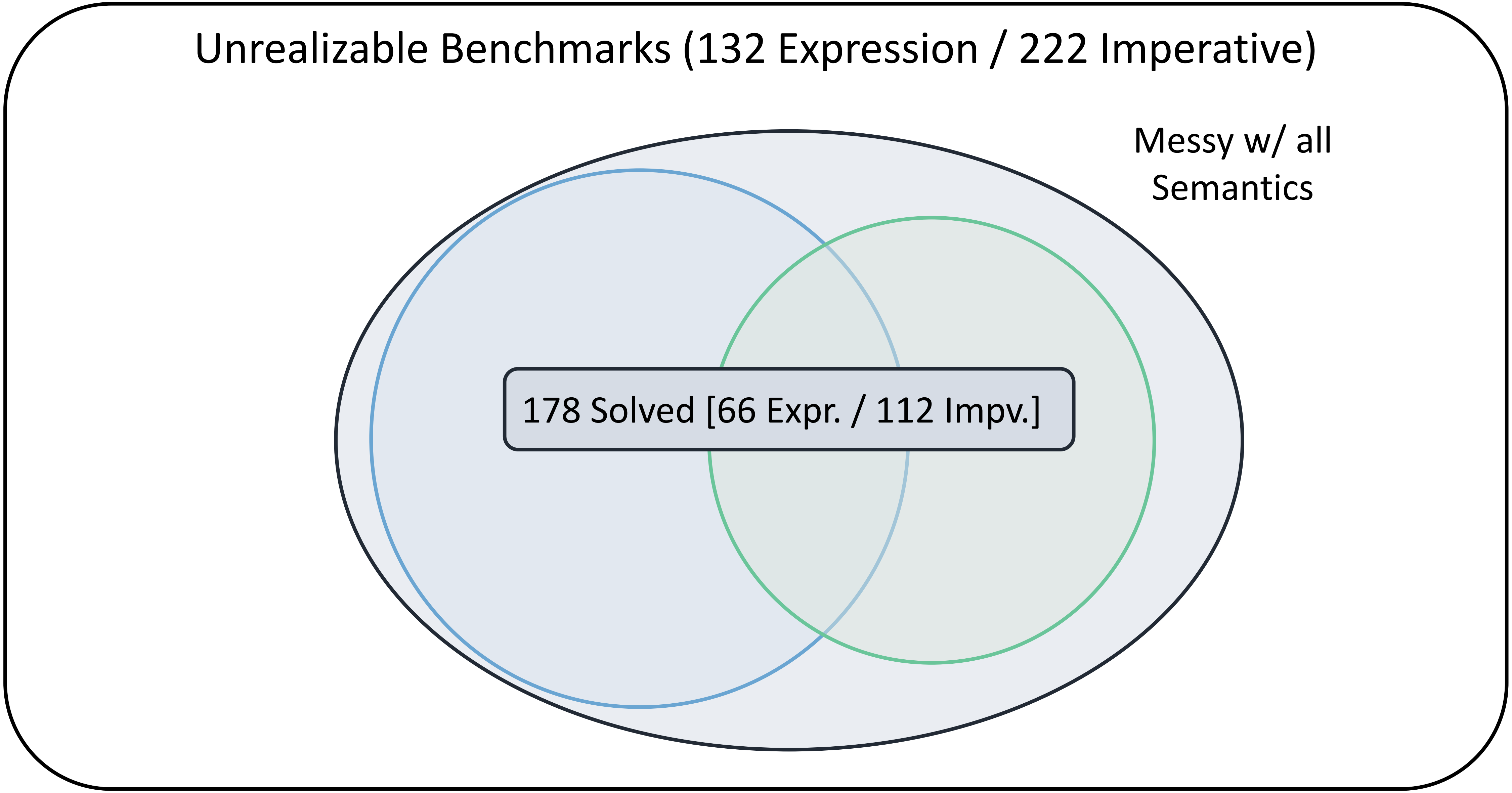

Realizable Benchmarks (60 Expression / 67 Imperative)
Bounded
Loops
6 Solved
[0 Exp. / 6 Impv.]


Realizable Benchmarks (60 Expression / 67 Imperative)
Messy w/ all
Semantics
12 Solved
[4 Exp. / 8 Impv.]

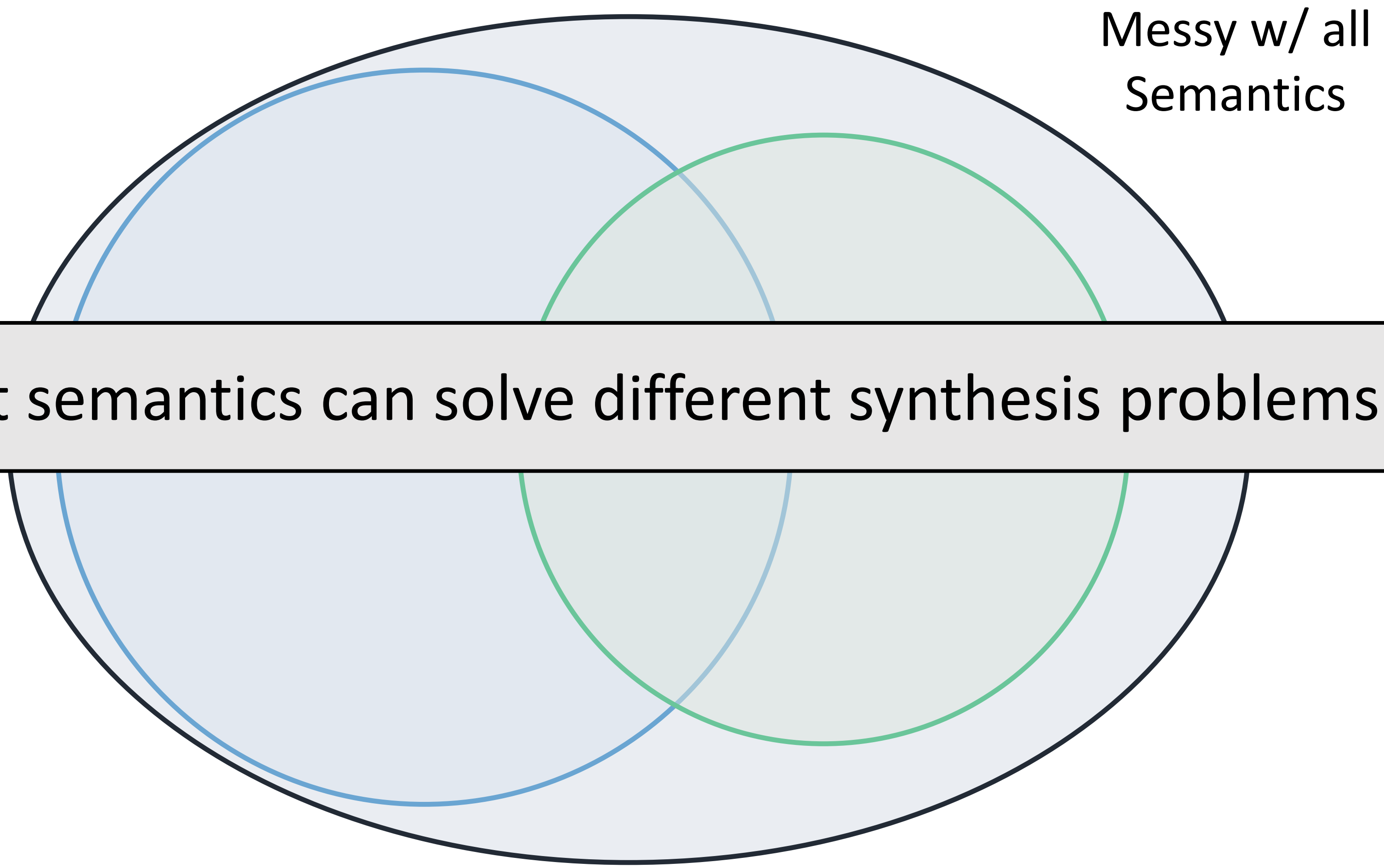

Unrealizable Benchmarks (132 Expression / 222 Imperative)
Standard
Semantics
66 Solved
[56 Exp. / 10 Impv.]


Unrealizable Benchmarks (132 Expression / 222 Imperative)
Standard Semantics
Abstract Semantics
55 Solved
[18 Exp. / 37 Impv.]

Unrealizable Benchmarks (132 Expression / 222 Imperative)


Unrealizable Benchmarks (132 Expression / 222 Imperative)
Messy w/ all
Semantics
178 Solved [66 Expr. / 112 Impv.]

Unrealizable Benchmarks (132 Expression / 222 Imperative)



Different semantics can solve different synthesis problems!

Semantics-Guided Synthesis Problem

Language Syntax

Regular Tree Grammar

Language Semantics

Constrained Horn Clauses

Problem Specification

Constraints / Examples

$$
\begin{aligned}
&Start ::= While\ B\ do\ S\\
&B ::= E < E\\
&E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)\\
&S ::= S; S \mid x \coloneqq E \mid y \coloneqq E
\end{aligned}
$$

$$\frac{Sem_E(\Gamma, t_e, v)\ \ \Gamma_1 = \Gamma[x \mapsto v]}{Sem_S(\Gamma, x \coloneqq t_e, \Gamma_1)}$$

Examples:

$f(6,9) = (15, \_)$

$\boxed{f(14,15) = (1, \_)}$

One more example!

$$\frac{Syn_{Start}(t) \quad Sem_{Start}((6,9), t, (15, y_1)) \quad \boxed{Sem_{Start}((14,15), t, (1, \_))}}{Realizable}$$

$$\frac{Syn_{Start}(t) \quad Sem_{Start}\big((6,9), t, (15,\_)\big) \quad Sem_{Start}\big((14,15), t, (1,\_)\big)}{Realizable}$$

Specification: $[\mathrm{f}(6,9) = (15,\_), \mathrm{f}(14,15) = (1,\_)]$

Syntax:

$Start ::= While\ B\ do\ S$

$B ::= E < E$

$E ::= \boxed{x \mid y \mid E \,\&\, E \mid (E \mid E)}$

$S ::= S; S \mid x := E \mid y := E$

$$\frac{Syn_{Start}(t) \quad Sem_{Start}\big((6,9), t, (15, \_)\big) \quad Sem_{Start}\big((14,15), t, (1, \_)\big)}{Realizable}$$

Specification: $[\mathrm{f}(6,9) = (15, \_), \mathrm{f}(14,15) = (1, \_)]$

Syntax:

$Start ::= While\ B\ do\ S$

$B ::= E < E$

$E ::= \boxed{x \mid y \mid E \,\&\, E \mid (E \mid E)}$

$S ::= S; S \mid x := E \mid y := E$

$$\frac{Syn_{Start}(t) \quad Sem_{Start}((6,9), t, (15, \_)) \quad \boxed{Sem_{Start}((14,15), t, (1, \_))}}{Realizable}$$

Specification: $[f(6,9) = (15, \_), f(14,15) = (1, \_)]$

Syntax:

$Start ::= While\ B\ do\ S$

$B ::= E < E$

$E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)$

$S ::= S; S \mid x \coloneqq E \mid y \coloneqq E$

Sem_{Start} satisfies the following lemma:

$$\boxed{\forall t, x', y'.\ Sem_{Start}\big((14, 15), t, (x', y')\big) \Rightarrow x' \,\&\, 4 = 4}$$

$$\frac{Syn_{Start}(t) \quad Sem_{Start}\big((6, 9), t, (15, \_)\big) \quad Sem_{Start}\big((14, 15), t, (1, \_)\big)}{Realizable}$$

Specification: $[\mathrm{f}(6, 9) = (15, \_), \mathrm{f}(14, 15) = (1, \_)]$

Syntax:

$Start ::= While\ B\ do\ S$

$B ::= E < E$

$E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)$

$S ::= S; S \mid x \coloneqq E \mid y \coloneqq E$

Sem_{Start} satisfies the following lemma:

$\forall t, x', y'.\ Sem_{Start}\big((14, 15), t, (x', y')\big) \Rightarrow x' \,\&\, 4 = 4$

$\boxed{\begin{array}{l} 14 = 1{\color{green}1}10_b \\ 15 = 1{\color{green}1}11_b \end{array}}$

$$\frac{Syn_{Start}(t) \quad Sem_{Start}\big((6, 9), t, (15, \_)\big) \quad Sem_{Start}\big((14, 15), t, (1, \_)\big)}{Realizable}$$

Specification: $[\mathrm{f}(6, 9) = (15, \_), \mathrm{f}(14, 15) = (1, \_)]$

Syntax:

$Start ::= While\ B\ do\ S$

$B ::= E < E$

$E ::= x \mid y \mid E \,\&\, E \mid (E \mid E)$

$S ::= S; S \mid x \coloneqq E \mid y \coloneqq E$

Sem_{Start} satisfies the following lemma:

$\forall t, x', y'.\ Sem_{Start}\big((14, 15), t, (x', y')\big) \Rightarrow x' \,\&\, 4 = 4$

For the desired output, 1 & 4 $\neq$ 4: Problem is unrealizable!

$$\frac{Syn_{Start}(t) \quad Sem_{Start}\big((6, 9), t, (15, \_)\big) \quad Sem_{Start}\big((14, 15), t, (1, \_)\big)}{Realizable}$$

Specification: $[\mathrm{f}(6, 9) = (15, \_), \mathrm{f}(14, 15) = (1, \_)]$

Semantics-Guided Synthesis Problem

Syntax:
Regular Tree Grammar

Semantics:
Constrained Horn Clauses

Specification:
Constraints / Examples

Highlights

- Defining the Semgus Framework
- CHC-based solving for Semgus problems
- Alternative semantics in synthesis
- Unrealizability for imperative Programs!

What's Next?

- More benchmarks for Semgus
- More Semgus solvers
- Practical issues: Efficiency, Scalability
- Theoretical questions: Can we further rely on semantics to guide synthesis?