Deciding Satisfiability for Overlaid Symbolic Heaps

Nicolas Peltier[1] **Quentin Petitjean**[2] **Mihaela Sighireanu**[2]

[1]Univ. Grenoble Alpes, CNRS, LIG, 38000 Grenoble France

[2]Univ. Paris-Saclay, CNRS, ENS Paris-Saclay, Laboratoire Méthodes Formelles, 91190 Gif-sur-Yvette, France

French National Research Agency project NARCO `ANR-21-CE48-0011`

FROCOS, Reykjavik, 30/09/2025

Introduction

- Separation Logic is widely used to reason about programs manipulating memory.
- SL use the connective $\star$ to compose disjoint structures and reason about them.

$$\text{Frame Rule: } \frac{\{P\}C\{Q\}}{\{P \star I\}C\{Q \star I\}}$$

- SL is an expressive program logic for data structures specified using inductive predicates:

$\mathtt{ls}(x, y)$: non-empty list

$$\begin{array}{lcl} \mathtt{ls}(x,y) & \Leftarrow & x \to (y) \\ \mathtt{ls}(x,y) & \Leftarrow & x \to (z) \star \mathtt{ls}(z,y) \end{array}$$

$\mathtt{bt}(x)$: binary tree

$$\begin{array}{lcl} \mathtt{bt}(x) & \Leftarrow & x \to () \\ \mathtt{bt}(x) & \Leftarrow & x \to (y,z) \star \mathtt{bt}(y) \star \mathtt{bt}(z) \end{array}$$

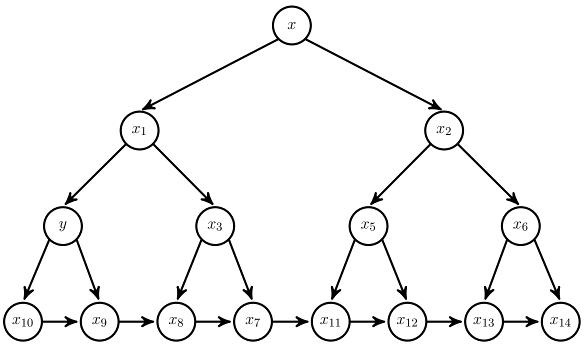

Figure. A more complex structure.

- The satisfiability problem is decidable for various fragments of SL, in particular with inductive predicates[1].
- The entailment problem is decidable in more restricted fragments of SL[2].
- General overlaid data structures raise issues for SL with inductive predicates:
  - expressivity;
  - compositional reasonning;
  - decidability of satisfibility and entailement.

[1] James Brotherston et al. **"A decision procedure for satisfiability in separation logic with inductive predicates".** In: *Proceedings of the Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*. July 2014

[2] Radu Iosif, Adam Rogalewicz, and Jiri Simacek. **"The Tree Width of Separation Logic with Recursive Definitions".** In: *Automated Deduction – CADE-24*. 2013

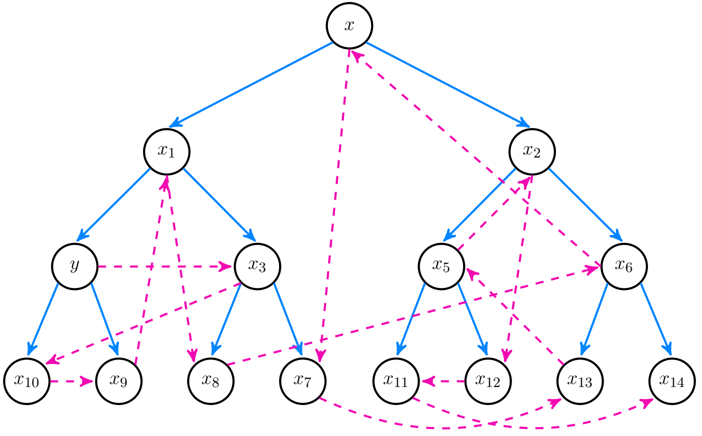

Figure. An overlaid data structure that slips out of inductive predicates.

1. Overlaid Separation Logic

Overlaid Separation Logic — Contribution

- We propose an extension of SL, Overlaid Separation Logic (OSL):
  - expressivity: capture complex data structure by still using inductive predicates;
  - allow composition reasoning due to a special overlaid separating conjunction;
  - decidability of satisfiability.
- We propose a decision procedure for the satisfiability problem for OSL.

Theorem

The satisfiability problem for OSL is decidable in NEXPTIME *if each predicate only allocates a single field.*

- Syntax:

$$\varphi := \texttt{emp} \mid x.f \to (y_1, \dots, y_d) \mid \mathrm{L} \mid \mathrm{B} \mid \mathrm{A} \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \star \varphi_2 \mid \varphi_1 \circledast \varphi_2$$
$$\mid p(x_1, \dots, x_{\#(p)-1}, \mathrm{X})$$

◇ Syntax:

$$\varphi := \texttt{emp} \mid x.f \to (y_1, \ldots, y_d) \mid \mathrm{L} \mid \mathrm{B} \mid \mathrm{A} \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \star \varphi_2 \mid \varphi_1 \circledast \varphi_2$$
$$\mid p(x_1, \ldots, x_{\#(p)-1}, \mathrm{X})$$

◇ Semantics:

- Structures $(\mathfrak{s}, \mathfrak{h}, \Sigma)$ composed of a *store* $\mathfrak{s}$ and a *heap* $\mathfrak{h}$ of domain $\mathcal{L} \times \mathcal{F}$, and a set interpretation Σ.

- Syntax:

$$\varphi := \texttt{emp} \mid x.f \to (y_1, \ldots, y_d) \mid \mathrm{L} \mid \mathrm{B} \mid \mathrm{A} \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \star \varphi_2 \mid \varphi_1 \circledast \varphi_2$$
$$\mid p(x_1, \ldots, x_{\#(p)-1}, \mathrm{X})$$

- Semantics:
  - Structures $(\mathfrak{s}, \mathfrak{h}, \Sigma)$ composed of a *store* $\mathfrak{s}$ and a *heap* $\mathfrak{h}$ of domain $\mathcal{L} \times \mathcal{F}$, and a set interpretation Σ.

◇ Syntax:

$$\varphi := \texttt{emp} \mid x.f \to (y_1, \ldots, y_d) \mid \mathrm{L} \mid \mathrm{B} \mid \mathrm{A} \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \star \varphi_2 \mid \varphi_1 \circledast \varphi_2$$
$$\mid p(x_1, \ldots, x_{\#(p)-1}, \mathrm{X})$$

◇ Semantics:

- Structures $(\mathfrak{s}, \mathfrak{h}, \Sigma)$ composed of a *store* $\mathfrak{s}$ and a *heap* $\mathfrak{h}$ of domain $\mathcal{L} \times \mathcal{F}$, and a set interpretation Σ.
- x is said allocated and $y_1, \ldots, y_d$ are said pointed-to.
- $(\mathfrak{s}, \mathfrak{h}) \models_{\mathcal{R}} (x.f \to (y_1, \ldots, y_d))$ if $\mathfrak{h} = [(\mathfrak{s}(x), f) \mapsto (\mathfrak{s}(y_1), \ldots, \mathfrak{s}(y_d))]$.

◇ Syntax:

$$\varphi := \texttt{emp} \mid x.f \to (y_1, \ldots, y_d) \mid \mathrm{L} \mid \mathrm{B} \mid \mathrm{A} \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \star \varphi_2 \mid \varphi_1 \circledast \varphi_2$$
$$\mid p(x_1, \ldots, x_{\#(p)-1}, \mathrm{X})$$

◇ Semantics:

- Structures $(\mathfrak{s}, \mathfrak{h}, \Sigma)$ composed of a *store* $\mathfrak{s}$ and a *heap* $\mathfrak{h}$ of domain $\mathcal{L} \times \mathcal{F}$, and a set interpretation Σ.
- x is said allocated and $y_1, \ldots, y_d$ are said pointed-to.
- $(\mathfrak{s}, \mathfrak{h}, \Sigma) \models_{\mathcal{R}} \varphi_1 \star \varphi_2$ if there exist $\mathfrak{h}_1, \mathfrak{h}_2$ such that $\mathfrak{h} = \mathfrak{h}_1 \cup \mathfrak{h}_2$, there is no location ℓ allocated in $\mathfrak{h}_1$ and in $\mathfrak{h}_2$, and $(\mathfrak{s}, \mathfrak{h}_i, \Sigma) \models_{\mathcal{R}} \varphi_i$.
- $(\mathfrak{s}, \mathfrak{h}, \Sigma) \models_{\mathcal{R}} \varphi_1 \circledast \varphi_2$ if there exist $\mathfrak{h}_1, \mathfrak{h}_2$ such that $\mathfrak{h} = \mathfrak{h}_1 \cup \mathfrak{h}_2$, there is no location ℓ allocated in $\mathfrak{h}_1$ and in $\mathfrak{h}_2$ with the same field, and $(\mathfrak{s}, \mathfrak{h}_i, \Sigma) \models_{\mathcal{R}} \varphi_i$.

Overlaid Separation Logic — Syntax and semantics

◇ Syntax:

$$\varphi := \texttt{emp} \mid x.f \to (y_1, \ldots, y_d) \mid \mathrm{L} \mid \mathrm{B} \mid \mathrm{A} \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \star \varphi_2 \mid \varphi_1 \circledast \varphi_2$$
$$\mid p(x_1, \ldots, x_{\#(p)-1}, \mathrm{X}) \qquad \mathrm{L} := x \approx y \mid x \not\approx y$$

◇ Semantics:

- Structures $(\mathfrak{s}, \mathfrak{h}, \Sigma)$ composed of a *store* $\mathfrak{s}$ and a *heap* $\mathfrak{h}$ of domain $\mathcal{L} \times \mathcal{F}$, and a set interpretation Σ.
- Equality constraints L over locations.

◇ Syntax:

$$\varphi := \texttt{emp} \mid x.f \to (y_1, \ldots, y_d) \mid \mathrm{L} \mid \mathrm{B} \mid \mathrm{A} \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \star \varphi_2 \mid \varphi_1 \circledast \varphi_2$$
$$\mid p(x_1, \ldots, x_{\#(p)-1}, \mathrm{X}) \qquad \mathrm{L} := x \approx y \mid x \not\approx y$$

$$\mathrm{T} := \{x\} \mid \mathrm{X} \mid \emptyset \mid \mathrm{T}_1 \sqcup \mathrm{T}_2 \mid \mathrm{T}_1 \sqcap \mathrm{T}_2 \qquad \mathrm{B} := \mathrm{T}_1 \approx \mathrm{T}_2 \mid \mathrm{T}_1 \not\approx \mathrm{T}_2 \mid \mathrm{T}_1 \sqsubseteq \mathrm{T}_2 \mid \mathrm{T}_1 \not\sqsubseteq \mathrm{T}_2$$

◇ Semantics:

- Structures $(\mathfrak{s}, \mathfrak{h}, \Sigma)$ composed of a *store* $\mathfrak{s}$ and a *heap* $\mathfrak{h}$ of domain $\mathcal{L} \times \mathcal{F}$, and a set interpretation Σ.
- Equality constraints L over locations.
- Set constraints B over set terms T, interpreted by finite sets of locations.

◇ Syntax:

$$\varphi := \mathtt{emp} \mid x.f \to (y_1,\ldots,y_d) \mid \mathrm{L} \mid \mathrm{B} \mid \mathrm{A} \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \star \varphi_2 \mid \varphi_1 \circledast \varphi_2$$
$$\mid p(x_1,\ldots,x_{\#(p)-1},\mathrm{X}) \qquad \mathrm{L} := x \approx y \mid x \not\approx y \qquad t := \mathrm{K} \mid t_1 \oplus t_2 \mid \mathrm{K} \odot t \mid |\mathrm{T}|$$
$$\mathrm{A} := t_1 \approx t_2 \mid t_1 \not\approx t_2 \mid t_1 \prec t_2 \mid t_1 \not\prec t_2 \mid \mathrm{K}\,\mathrm{div}\,t \mid \mathrm{K}\,\mathrm{ndiv}\,t$$
$$\mathrm{T} := \{x\} \mid \mathrm{X} \mid \emptyset \mid \mathrm{T}_1 \sqcup \mathrm{T}_2 \mid \mathrm{T}_1 \sqcap \mathrm{T}_2 \qquad \mathrm{B} := \mathrm{T}_1 \approx \mathrm{T}_2 \mid \mathrm{T}_1 \not\approx \mathrm{T}_2 \mid \mathrm{T}_1 \sqsubseteq \mathrm{T}_2 \mid \mathrm{T}_1 \not\sqsubseteq \mathrm{T}_2$$

◇ Semantics:

- Structures $(\mathfrak{s},\mathfrak{h},\Sigma)$ composed of a *store* $\mathfrak{s}$ and a *heap* $\mathfrak{h}$ of domain $\mathcal{L}\times\mathcal{F}$, and a set interpretation Σ.
- Equality constraints L over locations.
- Set constraints B over set terms T, interpreted by finite sets of locations.
- Arithmetic constraints A over arithmetic terms t, interpreted by integers.

◇ Syntax:

$$\varphi := \texttt{emp} \mid x.f \to (y_1,\ldots,y_d) \mid \mathrm{L} \mid \mathrm{B} \mid \mathrm{A} \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \star \varphi_2 \mid \varphi_1 \circledast \varphi_2$$
$$\mid p(x_1,\ldots,x_{\#(p)-1},\mathrm{X}) \qquad \mathrm{L} := x \approx y \mid x \not\approx y \qquad t := \mathrm{K} \mid t_1 \oplus t_2 \mid \mathrm{K} \odot t \mid |\mathrm{T}|$$
$$\mathrm{A} := t_1 \approx t_2 \mid t_1 \not\approx t_2 \mid t_1 \prec t_2 \mid t_1 \not\prec t_2 \mid \mathrm{K}\,\mathrm{div}\,t \mid \mathrm{K}\,\mathrm{ndiv}\,t$$
$$\mathrm{T} := \{x\} \mid \mathrm{X} \mid \emptyset \mid \mathrm{T}_1 \sqcup \mathrm{T}_2 \mid \mathrm{T}_1 \sqcap \mathrm{T}_2 \qquad \mathrm{B} := \mathrm{T}_1 \approx \mathrm{T}_2 \mid \mathrm{T}_1 \not\approx \mathrm{T}_2 \mid \mathrm{T}_1 \sqsubseteq \mathrm{T}_2 \mid \mathrm{T}_1 \not\sqsubseteq \mathrm{T}_2$$

◇ Semantics:

- Structures $(\mathfrak{s},\mathfrak{h},\Sigma)$ composed of a *store* $\mathfrak{s}$ and a *heap* $\mathfrak{h}$ of domain $\mathcal{L} \times \mathcal{F}$, and a set interpretation Σ.
- p is a predicate defined by a *set of inductive rules* (SID) $\mathcal{R}$ with a unique set variable X.

$$p(z_1,\ldots,z_{\#(p)-1},\mathrm{X}) \Leftarrow z_j.f \to (\vec{z}) \star \bigstar_{i=1}^{m} q_i(\vec{y}_i,\mathrm{Y}_i) \star \varphi \star (\mathrm{X} \approx \mathrm{E} \sqcup \bigsqcup_{i\in \mathrm{J}} \mathrm{Y}_i),$$

  where Y_i are pairwise distinct and distinct from X; E is either $\emptyset$ or $\{z_j\}$; $\mathrm{J} \subseteq [\![1,m]\!]$; φ is a $\star$-conjunction of equalities and disequalities.
- $(\mathfrak{s},\mathfrak{h},\Sigma) \models_{\mathcal{R}} p(x_1,\ldots,x_{\#(p)-1},\mathrm{X})$, if $(\mathfrak{s},\mathfrak{h},\Sigma) \models_{\mathcal{R}} \psi$ for some ψ such that $p(x_1,\ldots,x_{\#(p)-1},\mathrm{X}) \Leftarrow \psi$ (unfolding).

$\mathtt{bt}(x, \mathrm{Y}) \circledast \mathtt{ls}(y, \mathrm{Y})$ with:

$$\begin{aligned}
\mathtt{bt}(x, \mathrm{Y}) &\Leftarrow x.f \to () \star \mathrm{Y} \approx \{x\} \\
\mathtt{bt}(x, \mathrm{Y}) &\Leftarrow x.f \to (x_1, x_2) \star \mathtt{bt}(x_1, \mathrm{Y}_1) \\
&\quad \star \mathtt{bt}(x_2, \mathrm{Y}_2) \star \mathrm{Y} \approx \{y\} \sqcup \mathrm{Y}_1 \sqcup \mathrm{Y}_2 \\
\mathtt{ls}(y, \mathrm{Y}) &\Leftarrow y.g \to () \star \mathrm{Y} \approx \{y\} \\
\mathtt{ls}(y, \mathrm{Y}) &\Leftarrow y.g \to (y') \star \mathtt{ls}(y', \mathrm{Y}') \\
&\quad \star \mathrm{Y} \approx \{y\} \sqcup \mathrm{Y}'
\end{aligned}$$

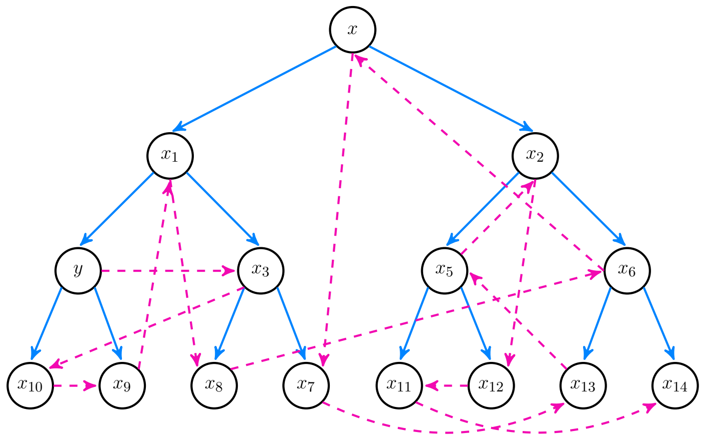

Figure. A model of $\mathtt{bt}(x, \mathrm{Y}) \circledast \mathtt{ls}(y, \mathrm{Y})$.

Overlaid Separation Logic — Examples

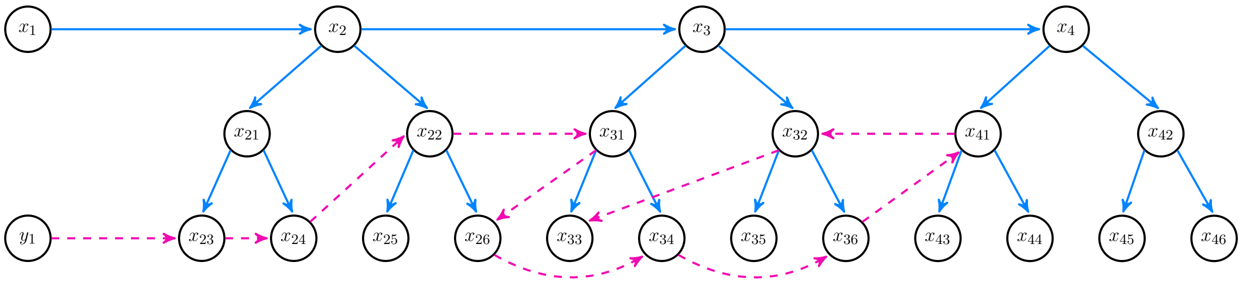

Figure. An OSL structure.

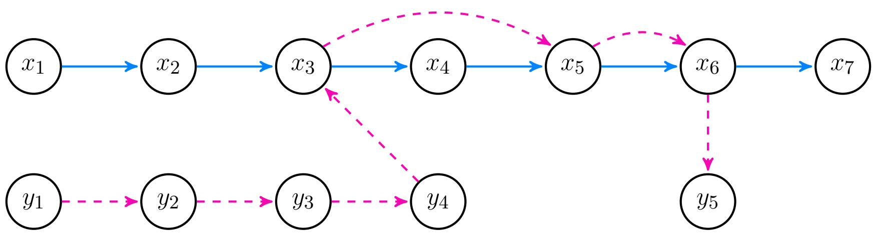

Figure. An OSL structure.

2. Decidability

Satisfiability of $(\varphi, \mathcal{R})$ is reduced to satisfiability of a BAPA formula[3].
Reduction steps:

[3] Viktor Kuncak, Huu Hai Nguyen, and Martin Rinard. **"An Algorithm for Deciding BAPA: Boolean Algebra with Presburger Arithmetic".** In: *Automated Deduction – CADE-20*. 2005

Satisfiability of $(\varphi, \mathcal{R})$ is reduced to satisfiability of a BAPA formula[3].
Reduction steps:

- First, decorate all the predicates appearing in φ and $\mathcal{R}$ to handle the spacial part, i.e, guessing and fixing:
  - the aliasing and non-aliasing relations between the location variables;
  - the set of location variables that occur in the set parameter of the predicate;
  - the set of allocated location variables.

[3] Viktor Kuncak, Huu Hai Nguyen, and Martin Rinard. **"An Algorithm for Deciding BAPA: Boolean Algebra with Presburger Arithmetic".** In: *Automated Deduction – CADE-20*. 2005

Decidability — Overview

Satisfiability of $(\varphi, \mathcal{R})$ is reduced to satisfiability of a BAPA formula[3].
Reduction steps:

- First, decorate all the predicates appearing in φ and $\mathcal{R}$ to handle the spacial part, i.e, guessing and fixing:
  - the aliasing and non-aliasing relations between the location variables;
  - the set of location variables that occur in the set parameter of the predicate;
  - the set of allocated location variables.
- Then, compute a set of rules for these decorated predicates using the rules in $\mathcal{R}$, keeping only those with coherent decorations.

[3] Viktor Kuncak, Huu Hai Nguyen, and Martin Rinard. **"An Algorithm for Deciding BAPA: Boolean Algebra with Presburger Arithmetic".** In: *Automated Deduction – CADE-20.* 2005

Decidability — Overview

Satisfiability of $(\varphi, \mathcal{R})$ is reduced to satisfiability of a BAPA formula[3].
Reduction steps:

- First, decorate all the predicates appearing in φ and $\mathcal{R}$ to handle the spacial part, i.e, guessing and fixing:
  - the aliasing and non-aliasing relations between the location variables;
  - the set of location variables that occur in the set parameter of the predicate;
  - the set of allocated location variables.
- Then, compute a set of rules for these decorated predicates using the rules in $\mathcal{R}$, keeping only those with coherent decorations.
- Next, use the decorations of the decorated pair to calculate Presburger formulæ describing the possible cardinalities of all set variables of φ.

[3] Viktor Kuncak, Huu Hai Nguyen, and Martin Rinard. **"An Algorithm for Deciding BAPA: Boolean Algebra with Presburger Arithmetic".** In: *Automated Deduction – CADE-20*. 2005

Decidability — Overview

Satisfiability of $(\varphi, \mathcal{R})$ is reduced to satisfiability of a BAPA formula[3].
Reduction steps:

- First, decorate all the predicates appearing in φ and $\mathcal{R}$ to handle the spacial part, i.e, guessing and fixing:
  - the aliasing and non-aliasing relations between the location variables;
  - the set of location variables that occur in the set parameter of the predicate;
  - the set of allocated location variables.
- Then, compute a set of rules for these decorated predicates using the rules in $\mathcal{R}$, keeping only those with coherent decorations.
- Next, use the decorations of the decorated pair to calculate Presburger formulæ describing the possible cardinalities of all set variables of φ.
- Finally, translate the guessed decorated pair into an equi-satisfiable formula in the logic BAPA by applying a recursive function on φ.

[3] Viktor Kuncak, Huu Hai Nguyen, and Martin Rinard. **"An Algorithm for Deciding BAPA: Boolean Algebra with Presburger Arithmetic".** In: *Automated Deduction – CADE-20.* 2005

Handling the spacial part with decorations.

Decidability — Decoration

Handling the spacial part with decorations.

- Decorated predicates: $p_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x_1, \ldots, x_n, \mathrm{X})$ where:
  - $i \in \mathrm{I}$ iff $x_i \in \mathrm{X}$;
  - $j \in \mathrm{J}$ iff x_j is allocated;
  - $\sim$ encodes the aliasing relation;
  - $\not\simeq$ encodes the distinguishing relation.

Decidability — Decoration

Handling the spacial part with decorations.

- Decorated predicates: $p_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x_1, \ldots, x_n, \mathrm{X})$ where:
  - $i \in \mathrm{I}$ iff $x_i \in \mathrm{X}$;
  - $j \in \mathrm{J}$ iff x_j is allocated;
  - $\sim$ encodes the aliasing relation;
  - $\not\simeq$ encodes the distinguishing relation.
- $[\mathrm{Y}]_\psi$, $alloc(\psi)$, $\equiv_\psi$, $\dot{\not=}_\psi$, are the extension of decoration to formulæ, induced by the decorations of predicates.

Decidability — Decoration

Handling the spacial part with decorations.

- Decorated predicates: $p_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x_1, \ldots, x_n, \mathrm{X})$ where:
  - $i \in \mathrm{I}$ iff $x_i \in \mathrm{X}$;
  - $j \in \mathrm{J}$ iff x_j is allocated;
  - $\sim$ encodes the aliasing relation;
  - $\not\simeq$ encodes the distinguishing relation.
- $[\mathrm{Y}]_\psi$, $alloc(\psi)$, $\equiv_\psi$, $\dot{\not=}_\psi$, are the extension of decoration to formulæ, induced by the decorations of predicates.
- Decorated rules must have coherent decorations for the right-hand and left-hand side.

Handling the spacial part with decorations.

- Decorated predicates: $p_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x_1, \ldots, x_n, \mathrm{X})$ where:
  - $i \in \mathrm{I}$ iff $x_i \in \mathrm{X}$;
  - $j \in \mathrm{J}$ iff x_j is allocated;
  - $\sim$ encodes the aliasing relation;
  - $\not\simeq$ encodes the distinguishing relation.
- $[\mathrm{Y}]_\psi$, $alloc(\psi)$, $\equiv_\psi$, $\dot{\not\equiv}_\psi$, are the extension of decoration to formulæ, induced by the decorations of predicates.
- Decorated rules must have coherent decorations for the right-hand and left-hand side.
- Consider $\varphi = \mathtt{ls}(x_1, y_1, \mathrm{X}_1) \star \mathtt{ls}(x_2, y_2, \mathrm{X}_2)$ with

$$\mathtt{ls}(x, y, \mathrm{X}) \Leftarrow x.f \to (y) \star \mathrm{X} \approx \{x\}\,,\ \mathtt{ls}(x, y, \mathrm{X}) \Leftarrow x.f \to (z) \star \mathtt{ls}(z, y, \mathrm{Y}) \star \mathrm{X} \approx \{x\} \sqcup \mathrm{Y}\,.$$

  The only decoration resulting in coherent rules is $\mathrm{I} = \{1\}$, $\mathrm{J} = \{1\}$, $\sim = \mathrm{Id}$, and $\not\simeq = \emptyset$.

Decidability — Cardinalities

We want to know the possible cardinalities of all set variables.

Decidability — Cardinalities

We want to know the possible cardinalities of all set variables.

- Let $\varphi = \mathtt{ls}^2(x, y, \mathrm{X}) \circledast \mathtt{ls}^1(x, y, \mathrm{Y}) \star \mathrm{X} = \mathrm{Y}$, with

$$\begin{aligned}
\mathtt{ls}^1(x, y, \mathrm{Y}) &\Leftarrow x.f \to (z) \star \mathrm{Y} \approx \{x\}\,, \\
\mathtt{ls}^1(x, y, \mathrm{Y}) &\Leftarrow x.f \to (z) \star \mathtt{ls}^2(z, y, \mathrm{X}) \star \mathrm{Y} \approx \{x\} \sqcup \mathrm{X}\,, \\
\mathtt{ls}^2(x, y, \mathrm{Y}) &\Leftarrow x.f \to (z) \star \mathtt{ls}^1(z, y, \mathrm{X}) \star \mathrm{X} \approx \{x\} \sqcup \mathrm{Y}\,.
\end{aligned}$$

Decidability — Cardinalities

We want to know the possible cardinalities of all set variables.

- Let $\varphi = \mathtt{ls}^2(x,y,\mathrm{X}) \circledast \mathtt{ls}^1(x,y,\mathrm{Y}) \star \mathrm{X} = \mathrm{Y}$, with

$$\mathtt{ls}^1(x,y,\mathrm{Y}) \Leftarrow x.f \to (z) \star \mathrm{Y} \approx \{x\}\,,$$
$$\mathtt{ls}^1(x,y,\mathrm{Y}) \Leftarrow x.f \to (z) \star \mathtt{ls}^2(z,y,\mathrm{X}) \star \mathrm{Y} \approx \{x\} \sqcup \mathrm{X}\,,$$
$$\mathtt{ls}^2(x,y,\mathrm{Y}) \Leftarrow x.f \to (z) \star \mathtt{ls}^1(z,y,\mathrm{X}) \star \mathrm{X} \approx \{x\} \sqcup \mathrm{Y}\,.$$

- Grammar of cardinalities: $\mathcal{G}_{\mathtt{ls}^2_{\mathrm{I},\mathrm{J},\sim,\not\approx}(x,y,\mathrm{X})} = (\mathcal{N}, \mathcal{T}, \mathcal{R}, \mathrm{N}_0)$ with $\mathcal{N} = \{\mathrm{N}_{\mathtt{ls}^2_{\mathrm{I},\mathrm{J},\sim,\not\approx}(x,y,\mathrm{X})}, \mathrm{N}_{\mathtt{ls}^1_{\mathrm{I},\mathrm{J},\sim,\not\approx}(x,y,\mathrm{Y})}\}$, $\mathcal{T} = \{1\}$, $\mathrm{N}_0 = \mathrm{N}_{\mathtt{ls}^2_{\mathrm{I},\mathrm{J},\sim,\not\approx}(x,y,\mathrm{X})}$ and $\mathcal{R}$ containing:
  - $\mathrm{N}_{\mathtt{ls}^1_{\mathrm{I},\mathrm{J},\sim,\not\approx}(x,y,\mathrm{Y})} \to 1$; $\quad \mathrm{N}_{\mathtt{ls}^1_{\mathrm{I},\mathrm{J},\sim,\not\approx}(x,y,\mathrm{Y})} \to 1\mathrm{N}_{\mathtt{ls}^2_{\mathrm{I},\mathrm{J},\sim,\not\approx}(x,y,\mathrm{X})}$;
  - $\mathrm{N}_{\mathtt{ls}^2_{\mathrm{I},\mathrm{J},\sim,\not\approx}(x,y,\mathrm{X})} \to 1\mathrm{N}_{\mathtt{ls}^1_{\mathrm{I},\mathrm{J},\sim,\not\approx}(x,y,\mathrm{Y})}$.

We want to know the possible cardinalities of all set variables.

- Let $\varphi = \mathtt{ls}^2(x, y, \mathrm{X}) \circledast \mathtt{ls}^1(x, y, \mathrm{Y}) \star \mathrm{X} = \mathrm{Y}$, with

$$\mathtt{ls}^1(x, y, \mathrm{Y}) \Leftarrow x.f \to (z) \star \mathrm{Y} \approx \{x\}\,,$$
$$\mathtt{ls}^1(x, y, \mathrm{Y}) \Leftarrow x.f \to (z) \star \mathtt{ls}^2(z, y, \mathrm{X}) \star \mathrm{Y} \approx \{x\} \sqcup \mathrm{X}\,,$$
$$\mathtt{ls}^2(x, y, \mathrm{Y}) \Leftarrow x.f \to (z) \star \mathtt{ls}^1(z, y, \mathrm{X}) \star \mathrm{X} \approx \{x\} \sqcup \mathrm{Y}\,.$$

- Grammar of cardinalities: $\mathcal{G}_{\mathtt{ls}^2_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{X})} = (\mathcal{N}, \mathcal{T}, \mathcal{R}, \mathrm{N}_0)$ with $\mathcal{N} = \{\mathrm{N}_{\mathtt{ls}^2_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{X})}, \mathrm{N}_{\mathtt{ls}^1_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{Y})}\}$, $\mathcal{T} = \{1\}$, $\mathrm{N}_0 = \mathrm{N}_{\mathtt{ls}^2_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{X})}$ and $\mathcal{R}$ containing:
  - $\mathrm{N}_{\mathtt{ls}^1_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{Y})} \to 1$; $\qquad \mathrm{N}_{\mathtt{ls}^1_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{Y})} \to 1\mathrm{N}_{\mathtt{ls}^2_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{X})}$;
  - $\mathrm{N}_{\mathtt{ls}^2_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{X})} \to 1\mathrm{N}_{\mathtt{ls}^1_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{Y})}$.
- $\mathrm{L}(\mathcal{G}_{\mathtt{ls}^2_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{X})})$ corresponds to $Sp(\mathtt{ls}^2_{\mathrm{I,J},\sim,\not\approx}(x, y, \mathrm{X}))$, the set of values of $\mathrm{card}(\Sigma(\mathrm{X}))$.

Decidability — Cardinalities

We want to know the possible cardinalities of all set variables.

- Let $\varphi = \mathtt{ls}^2(x, y, \mathrm{X}) \circledast \mathtt{ls}^1(x, y, \mathrm{Y}) \star \mathrm{X} = \mathrm{Y}$, with

$$\mathtt{ls}^1(x, y, \mathrm{Y}) \Leftarrow x.f \to (z) \star \mathrm{Y} \approx \{x\}\,,$$
$$\mathtt{ls}^1(x, y, \mathrm{Y}) \Leftarrow x.f \to (z) \star \mathtt{ls}^2(z, y, \mathrm{X}) \star \mathrm{Y} \approx \{x\} \sqcup \mathrm{X}\,,$$
$$\mathtt{ls}^2(x, y, \mathrm{Y}) \Leftarrow x.f \to (z) \star \mathtt{ls}^1(z, y, \mathrm{X}) \star \mathrm{X} \approx \{x\} \sqcup \mathrm{Y}\,.$$

- Grammar of cardinalities: $\mathcal{G}_{\mathtt{ls}^2_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{X})} = (\mathcal{N}, \mathcal{T}, \mathcal{R}, \mathrm{N}_0)$ with $\mathcal{N} = \{\mathrm{N}_{\mathtt{ls}^2_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{X})}, \mathrm{N}_{\mathtt{ls}^1_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{Y})}\}$, $\mathcal{T} = \{1\}$, $\mathrm{N}_0 = \mathrm{N}_{\mathtt{ls}^2_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{X})}$ and $\mathcal{R}$ containing:
  - $\mathrm{N}_{\mathtt{ls}^1_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{Y})} \to 1$; $\quad \mathrm{N}_{\mathtt{ls}^1_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{Y})} \to 1\mathrm{N}_{\mathtt{ls}^2_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{X})}$;
  - $\mathrm{N}_{\mathtt{ls}^2_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{X})} \to 1\mathrm{N}_{\mathtt{ls}^1_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{Y})}$.
- $\mathrm{L}(\mathcal{G}_{\mathtt{ls}^2_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{X})})$ corresponds to $Sp(\mathtt{ls}^2_{\mathrm{I,J},\sim,\not\approx}(x, y, \mathrm{X}))$, the set of values of $\mathrm{card}(\Sigma(\mathrm{X}))$.
- An existential Presburger formula $\xi_{\mathtt{ls}^2_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{X})}$, describing the Parikh image[4] of the language can be computed in in linear time[5]. It simplifies into $\exists k. i_{\mathrm{X}} = 2(k + 1)$.

---

[4] Rohit J. Parikh. **"On Context-Free Languages".** In: *J. ACM* 4 (Oct. 1966)

[5] Kumar Neeraj Verma, Helmut Seidl, and Thomas Schwentick. **"On the Complexity of Equational Horn Clauses".** In: *Automated Deduction – CADE-20*. 2005

Let $\psi = \mathtt{ls}^{f}_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x, z, \mathrm{X}) \circledast \mathtt{ls}^{f}_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(y, z, \mathrm{Y}) \star \mathrm{X} \approx \mathrm{Y}$ with $\mathrm{I}, \mathrm{J}, \sim, \not\simeq$ defined as previously.
We translate into:

$$\mathcal{C}(\psi) =$$

Let $\psi = \mathtt{ls}^{f}_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x, z, \mathrm{X}) \circledast \mathtt{ls}^{f}_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(y, z, \mathrm{Y}) \star \mathrm{X} \approx \mathrm{Y}$ with $\mathrm{I}, \mathrm{J}, \sim, \not\simeq$ defined as previously.
We translate into:

$$\mathcal{C}(\psi) = (|\mathrm{V}_x| \approx_{\mathrm{BP}} 1) \wedge (|\mathrm{V}_y| \approx_{\mathrm{BP}} 1) \wedge (|\mathrm{V}_z| \approx_{\mathrm{BP}} 1)$$

- We associate to each free variable x of ψ a fresh BAPA set variable V_x.

Let $\psi = \mathtt{ls}^{f}_{\mathrm{I,J},\sim,\not\simeq}(x, z, \mathrm{X}) \circledast \mathtt{ls}^{f}_{\mathrm{I,J},\sim,\not\simeq}(y, z, \mathrm{Y}) \star \mathrm{X} \approx \mathrm{Y}$ with $\mathrm{I}, \mathrm{J}, \sim, \not\simeq$ defined as previously.
We translate into:

$$\mathcal{C}(\psi) = (|\mathrm{V}_x| \approx_{\mathrm{BP}} 1) \wedge (|\mathrm{V}_y| \approx_{\mathrm{BP}} 1) \wedge (|\mathrm{V}_z| \approx_{\mathrm{BP}} 1)$$
$$\wedge\, (|\mathrm{X}| \approx_{\mathrm{BP}} i_{\mathrm{X}}) \wedge \xi_{\mathtt{ls}^{f}_{\mathrm{I,J},\sim,\not\simeq}(x,y,\mathrm{X})}(i_{\mathrm{X}}) \wedge (\mathrm{V}_x \sqsubseteq_{\mathrm{BP}} \mathrm{X}) \wedge (\mathrm{V}_z \sqcap_{\mathrm{BP}} \mathrm{X} \approx_{\mathrm{BP}} \emptyset)$$

- We associate to each free variable x of ψ a fresh BAPA set variable V_x.
- The translation of a decorated quantifier-free symbolic heap into a BAPA formula:
  - $\mathcal{T}(p_{\mathrm{I,J},\sim,\not\simeq}(x_1, \ldots, x_n, \mathrm{X})) = |\mathrm{X}| \approx_{\mathrm{BP}} i_{\mathrm{X}} \wedge \xi_{p_{\mathrm{I,J},\sim,\not\simeq}(x_1,\ldots,x_n,\mathrm{X})}(i_{\mathrm{X}})$
    $\wedge \left(\bigsqcup^{\mathrm{BP}}_{i\in\mathrm{I}} \mathrm{V}_{x_i}\right) \sqsubseteq_{\mathrm{BP}} \mathrm{X} \wedge \left(\bigsqcup^{\mathrm{BP}}_{x\in\{x_1,\ldots,x_n\}\setminus\{x_j \mid j\in\mathrm{I}\}} \mathrm{V}_x\right) \sqcap_{\mathrm{BP}} \mathrm{X} \approx_{\mathrm{BP}} \emptyset;$

Decidability — Translation from SL to BAPA

Let $\psi = \mathtt{ls}^{f}_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x,z,\mathrm{X}) \circledast \mathtt{ls}^{f}_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(y,z,\mathrm{Y}) \star \mathrm{X} \approx \mathrm{Y}$ with $\mathrm{I},\mathrm{J},\sim,\not\simeq$ defined as previously. We translate into:

$$\begin{aligned}\mathcal{C}(\psi) = (|\mathrm{V}_x| \approx_{\mathrm{BP}} 1) \wedge (|\mathrm{V}_y| \approx_{\mathrm{BP}} 1) \wedge (|\mathrm{V}_z| \approx_{\mathrm{BP}} 1)\\ \wedge (|\mathrm{X}| \approx_{\mathrm{BP}} i_{\mathrm{X}}) \wedge \xi_{\mathtt{ls}^{f}_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x,y,\mathrm{X})}(i_{\mathrm{X}}) \wedge (\mathrm{V}_x \sqsubseteq_{\mathrm{BP}} \mathrm{X}) \wedge (\mathrm{V}_z \sqcap_{\mathrm{BP}} \mathrm{X} \approx_{\mathrm{BP}} \emptyset)\\ \wedge (|\mathrm{Y}| \approx_{\mathrm{BP}} i_{\mathrm{Y}}) \wedge \xi_{\mathtt{ls}^{f}_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x,z,\mathrm{Y})}(i_{\mathrm{Y}}) \wedge (\mathrm{V}_y \sqsubseteq_{\mathrm{BP}} \mathrm{Y}) \wedge (\mathrm{V}_z \sqcap_{\mathrm{BP}} \mathrm{Y} \approx_{\mathrm{BP}} \emptyset)\end{aligned}$$

- We associate to each free variable x of ψ a fresh BAPA set variable V_x.
- The translation of a decorated quantifier-free symbolic heap into a BAPA formula:
  - $\mathcal{T}(p_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x_1,\ldots,x_n,\mathrm{X})) = |\mathrm{X}| \approx_{\mathrm{BP}} i_{\mathrm{X}} \wedge \xi_{p_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x_1,\ldots,x_n,\mathrm{X})}(i_{\mathrm{X}})$ $\wedge (\bigsqcup^{\mathrm{BP}}_{i\in\mathrm{I}} \mathrm{V}_{x_i}) \sqsubseteq_{\mathrm{BP}} \mathrm{X} \wedge \left(\bigsqcup^{\mathrm{BP}}_{x\in\{x_1,\ldots,x_n\}\setminus\{x_j \mid j\in\mathrm{I}\}} \mathrm{V}_x\right) \sqcap_{\mathrm{BP}} \mathrm{X} \approx_{\mathrm{BP}} \emptyset$;

Let $\psi = \mathtt{ls}^f_{\mathrm{I,J},\sim,\not\approx}(x,z,\mathrm{X}) \circledast \mathtt{ls}^f_{\mathrm{I,J},\sim,\not\approx}(y,z,\mathrm{Y}) \star \mathrm{X} \approx \mathrm{Y}$ with $\mathrm{I}, \mathrm{J}, \sim, \not\approx$ defined as previously.
We translate into:

$$
\begin{aligned}
\mathcal{C}(\psi) = {} & (|\mathrm{V}_x| \approx_{\mathrm{BP}} 1) \wedge (|\mathrm{V}_y| \approx_{\mathrm{BP}} 1) \wedge (|\mathrm{V}_z| \approx_{\mathrm{BP}} 1) \\
& \wedge (|\mathrm{X}| \approx_{\mathrm{BP}} i_{\mathrm{X}}) \wedge \xi_{\mathtt{ls}^f_{\mathrm{I,J},\sim,\not\approx}(x,y,\mathrm{X})}(i_{\mathrm{X}}) \wedge (\mathrm{V}_x \sqsubseteq_{\mathrm{BP}} \mathrm{X}) \wedge (\mathrm{V}_z \sqcap_{\mathrm{BP}} \mathrm{X} \approx_{\mathrm{BP}} \emptyset) \\
& \wedge (|\mathrm{Y}| \approx_{\mathrm{BP}} i_{\mathrm{Y}}) \wedge \xi_{\mathtt{ls}^f_{\mathrm{I,J},\sim,\not\approx}(x,z,\mathrm{Y})}(i_{\mathrm{Y}}) \wedge (\mathrm{V}_y \sqsubseteq_{\mathrm{BP}} \mathrm{Y}) \wedge (\mathrm{V}_z \sqcap_{\mathrm{BP}} \mathrm{Y} \approx_{\mathrm{BP}} \emptyset) \\
& \wedge ((\mathrm{V}_x \sqcup \mathrm{X}) \sqcap_{\mathrm{BP}} (\mathrm{V}_y \sqcup \mathrm{Y}) \approx_{\mathrm{BP}} \emptyset)
\end{aligned}
$$

- We associate to each free variable x of ψ a fresh BAPA set variable V_x.
- The translation of a decorated quantifier-free symbolic heap into a BAPA formula:
  - $\mathcal{T}(p_{\mathrm{I,J},\sim,\not\approx}(x_1,\ldots,x_n,\mathrm{X})) = |\mathrm{X}| \approx_{\mathrm{BP}} i_{\mathrm{X}} \wedge \xi_{p_{\mathrm{I,J},\sim,\not\approx}(x_1,\ldots,x_n,\mathrm{X})}(i_{\mathrm{X}})$
    $\wedge (\bigsqcup^{\mathrm{BP}}_{i\in\mathrm{I}} \mathrm{V}_{x_i}) \sqsubseteq_{\mathrm{BP}} \mathrm{X} \wedge \left(\bigsqcup^{\mathrm{BP}}_{x\in\{x_1,\ldots,x_n\}\setminus\{x_j \mid j\in\mathrm{I}\}} \mathrm{V}_x\right) \sqcap_{\mathrm{BP}} \mathrm{X} \approx_{\mathrm{BP}} \emptyset$;
  - $\mathcal{T}(\psi_1 \star \psi_2) = \mathcal{T}(\psi_1) \wedge \mathcal{T}(\psi_2) \wedge \left(\bigsqcup^{\mathrm{BP}}_{f\in\mathcal{F}} \mathcal{T}^f(\psi_1)\right) \sqcap_{\mathrm{BP}} \left(\bigsqcup^{\mathrm{BP}}_{f\in\mathcal{F}} \mathcal{T}^f(\psi_2)\right) \approx_{\mathrm{BP}} \emptyset$;
  - $\mathcal{T}(\psi_1 \circledast \psi_2) = \mathcal{T}(\psi_1) \wedge \mathcal{T}(\psi_2) \wedge \bigwedge_{f\in\mathcal{F}} \left(\mathcal{T}^f(\psi_1) \sqcap_{\mathrm{BP}} \mathcal{T}^f(\psi_2) \approx_{\mathrm{BP}} \emptyset\right)$.
  - $\mathcal{T}^f(\psi)$ is a set term denoting the set of named locations allocated by φ, for field f.

Let $\psi = \mathtt{ls}^{f}_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x, z, \mathrm{X}) \circledast \mathtt{ls}^{f}_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(y, z, \mathrm{Y}) \star \mathrm{X} \approx \mathrm{Y}$ with $\mathrm{I}, \mathrm{J}, \sim, \not\simeq$ defined as previously.
We translate into:

$$\begin{aligned}\mathcal{C}(\psi) = {} & (|\mathrm{V}_x| \approx_{\mathrm{BP}} 1) \wedge (|\mathrm{V}_y| \approx_{\mathrm{BP}} 1) \wedge (|\mathrm{V}_z| \approx_{\mathrm{BP}} 1) \\ & \wedge (|\mathrm{X}| \approx_{\mathrm{BP}} i_{\mathrm{X}}) \wedge \xi_{\mathtt{ls}^{f}_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x,y,\mathrm{X})}(i_{\mathrm{X}}) \wedge (\mathrm{V}_x \sqsubseteq_{\mathrm{BP}} \mathrm{X}) \wedge (\mathrm{V}_z \sqcap_{\mathrm{BP}} \mathrm{X} \approx_{\mathrm{BP}} \emptyset) \\ & \wedge (|\mathrm{Y}| \approx_{\mathrm{BP}} i_{\mathrm{Y}}) \wedge \xi_{\mathtt{ls}^{f}_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x,z,\mathrm{Y})}(i_{\mathrm{Y}}) \wedge (\mathrm{V}_y \sqsubseteq_{\mathrm{BP}} \mathrm{Y}) \wedge (\mathrm{V}_z \sqcap_{\mathrm{BP}} \mathrm{Y} \approx_{\mathrm{BP}} \emptyset) \\ & \wedge ((\mathrm{V}_x \sqcup \mathrm{X}) \sqcap_{\mathrm{BP}} (\mathrm{V}_y \sqcup \mathrm{Y}) \approx_{\mathrm{BP}} \emptyset) \\ & \wedge (\mathrm{X} \approx_{\mathrm{BP}} \mathrm{Y})\,.\end{aligned}$$

- We associate to each free variable x of ψ a fresh BAPA set variable V_x.
- The translation of a decorated quantifier-free symbolic heap into a BAPA formula:
  - $\mathcal{T}(p_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x_1, \ldots, x_n, \mathrm{X})) = |\mathrm{X}| \approx_{\mathrm{BP}} i_{\mathrm{X}} \wedge \xi_{p_{\mathrm{I},\mathrm{J},\sim,\not\simeq}(x_1,\ldots,x_n,\mathrm{X})}(i_{\mathrm{X}})$
    $\wedge (\bigsqcup^{\mathrm{BP}}_{i \in \mathrm{I}} \mathrm{V}_{x_i}) \sqsubseteq_{\mathrm{BP}} \mathrm{X} \wedge \left(\bigsqcup^{\mathrm{BP}}_{x \in \{x_1,\ldots,x_n\} \setminus \{x_j \mid j \in \mathrm{I}\}} \mathrm{V}_x\right) \sqcap_{\mathrm{BP}} \mathrm{X} \approx_{\mathrm{BP}} \emptyset$;
  - $\mathcal{T}(\psi_1 \star \psi_2) = \mathcal{T}(\psi_1) \wedge \mathcal{T}(\psi_2) \wedge \left(\bigsqcup^{\mathrm{BP}}_{f \in \mathcal{F}} \mathcal{T}^f(\psi_1)\right) \sqcap_{\mathrm{BP}} \left(\bigsqcup^{\mathrm{BP}}_{f \in \mathcal{F}} \mathcal{T}^f(\psi_2)\right) \approx_{\mathrm{BP}} \emptyset$;
  - $\mathcal{T}(\psi_1 \circledast \psi_2) = \mathcal{T}(\psi_1) \wedge \mathcal{T}(\psi_2) \wedge \bigwedge_{f \in \mathcal{F}} \left(\mathcal{T}^f(\psi_1) \sqcap_{\mathrm{BP}} \mathcal{T}^f(\psi_2) \approx_{\mathrm{BP}} \emptyset\right)$.
  - $\mathcal{T}^f(\psi)$ is a set term denoting the set of named locations allocated by φ, for field f.

3. Conclusion and Future Work

Conclusion and Future Work

Contributions:

- The SL extension OSL captures a wide range of overlaid data structures specified compositionally using inductively defined predicates.
- The satisfiability problem is decidable in Nexptime.

Conclusion and Future Work

Contributions:

- The SL extension OSL captures a wide range of overlaid data structures specified compositionally using inductively defined predicates.
- The satisfiability problem is decidable in Nexptime.

Some lines of **future work**:

- Explore the decidability of the entailment problem: this work is ongoing and requires additional restrictions.
- Investigate the optimality of the procedure: satisfiability is clearly Exptime-hard, but it is not clear whether Nexptime represents a tight upper bound.
- Investigate whether the systematic enumeration of all decorations could be circumvented.
- Determine whether the conditions on the inductive rules could be relaxed.

Thank You!