Size Does Matter: Two Certified Abstractions for Disproving Entailment between Separation Logic Formulas

François Bobot[⋆] Clément Hurlin[↯] Alexander Summers[✠]

[⋆]INRIA Saclay – Île-de-France, France

[↯] INRIA Sophia Antipolis – Méditerranée, France
Twente University, The Netherlands

[✠]Imperial College London

January 8th 2009

ParSec

Parallelism and Security

Motivation

- Disprove entailment between formulas

↳ I.e. to prove $A \not\models B$

- A and B are separation logic formulas.

Motivation

- Disprove entailment between formulas

↳ I.e. to prove $A \not\models B$

- A and B are separation logic formulas.

Technique:

- By discriminating models of A and B

Separation Logic: $a \overset{\pi}{\mapsto} v$

$a \overset{\pi}{\mapsto} v$ (called "points-to predicate") has a dual meaning:

- Address a contains value v.
- Permission π to access address a.

π is a *fraction* in $(0,1]$:

- 1 is the permission to write access a location.
- Any $0 < \pi < 1$ is the permission to read-only access a location.

Separation Logic: $\star$

$A \star B$ is the *separating conjunction*:

- Permissions to access heap A and heap B
- $A \star A$ does not imply A (no weakening).
- But A does not imply $A \star A$ (no copying).
- $\star$ separates permissions.

Separation Logic: $\star$

$A \star B$ is the *separating conjunction*:

- Permissions to access heap A and heap B
- $A \star A$ does not imply A (no weakening).
- But A does not imply $A \star A$ (no copying).
- $\star$ separates permissions.

Last item means:

- $a \neq b$: $a \overset{1}{\mapsto} \_ \star b \overset{1}{\mapsto} \_$ ✓
- But: $a \overset{1}{\mapsto} \_ \star a \overset{\pi}{\mapsto} \_$ ×

Separation Logic: $\star$

Two axioms:

$$a \overset{\pi}{\mapsto} v \Rightarrow a \overset{\frac{\pi}{2}}{\mapsto} v \star a \overset{\frac{\pi}{2}}{\mapsto} v \qquad \text{(Split)}$$

$$a \overset{\frac{\pi}{2}}{\mapsto} v \star a \overset{\frac{\pi}{2}}{\mapsto} v \Rightarrow a \overset{\pi}{\mapsto} v \qquad \text{(Merge)}$$

Separation Logic: $-\star$

$A -\star B$ is the *linear implication* (or "*baguette magique*"):

- Reads "consume A yielding B" or "trade A and receive B"
- $A \star (A -\star B)$ implies B

Semantics: $\mathscr{M} \models A$

- Models $\mathscr{M}$ are lists of couples of an address and a permission.

↳ An example model is $(245, \frac{1}{2}) :: (246, 1) :: (245, \frac{1}{3}) :: []$.

Semantics: $\mathscr{M} \models A$

- Models $\mathscr{M}$ are lists of couples of an address and a permission.

↳ An example model is $(245, \frac{1}{2}) :: (246, 1) :: (245, \frac{1}{3}) :: []$.

$$\mathscr{M} \models a \overset{\pi}{\mapsto} \_ \quad \text{iff} \quad \mathscr{M} = (a, \pi) :: []$$

$$\mathscr{M} \models A \star B \quad \text{iff} \quad \exists \mathscr{M}_A, \mathscr{M}_B, \mathscr{M} = \mathscr{M}_A \uplus \mathscr{M}_B, \text{and } \mathscr{M}_A \models A \text{ and } \mathscr{M}_B \models B$$

$$\mathscr{M} \models A \mathrel{-\!\!\star} B \quad \text{iff} \quad \forall \mathscr{M}_A, \mathscr{M}_A \models A \text{ and } \mathscr{M}_A \cap \mathscr{M} = \emptyset \text{ implies } \mathscr{M}_A \uplus \mathscr{M} \models A \star B$$

Semantics: $\mathscr{M} \models A$

$$\mathscr{M} \models a \overset{\pi}{\mapsto} \_ \quad \text{iff} \quad \mathscr{M} = (a, \pi)$$

$$\mathscr{M} \models A \star B \quad \text{iff} \quad \exists \mathscr{M}_A, \mathscr{M}_B, \mathscr{M} = \mathscr{M}_A \uplus \mathscr{M}_B, \text{and } \mathscr{M}_A \models A \text{ and } \mathscr{M}_B \models B$$

$$\mathscr{M} \models A \mathrel{-\!\star} B \quad \text{iff} \quad \forall \mathscr{M}_A, \mathscr{M}_A \models A \text{ and } \mathscr{M}_A \cap \mathscr{M} = \emptyset \text{ implies } \mathscr{M}_A \uplus \mathscr{M} \models A \star B$$

$$\mathscr{M} \models A \wedge B \quad \text{iff} \quad \mathscr{M} \models A \text{ and } \mathscr{M} \models B$$

$$\mathscr{M} \models A \vee B \quad \text{iff} \quad \mathscr{M} \models A \text{ or } \mathscr{M} \models B$$

Disproving Technique

Soundness of the proof system:

$$\boxed{A \vdash B \text{ implies } (\forall \mathscr{M}, \mathscr{M} \models A \rightarrow \mathscr{M} \models B)}$$

Disproving Technique

Soundness of the proof system:

$$A \vdash B \text{ implies } (\forall \mathscr{M}, \mathscr{M} \models A \rightarrow \mathscr{M} \models B)$$

Contraposition:

$$(\exists \mathscr{M}, \mathscr{M} \models A \wedge \neg \mathscr{M} \models B) \text{ implies } A \nvdash B$$

Goal of this work:

- Take A and B and prove that $A \nvdash B$
- By discriminating models of A and B

Disproving Technique

Contraposition:

$$(\exists \mathscr{M}, \mathscr{M} \models A \land \neg \mathscr{M} \models B) \text{ implies } A \not\models B$$

Objective:

$$\text{Find } \mathscr{M} \text{ such that } \mathscr{M} \models A \text{ and } \neg \mathscr{M} \models B$$

Disproving Technique

Objective:

$$\text{Find } \mathscr{M} \text{ such that } \mathscr{M} \models A \text{ and } \neg \mathscr{M} \models B$$

To do that:

- We compute bounds on the size of models.
- $\mathsf{max} : \text{Formula} \rightarrow \mathbb{S}$ ($\mathbb{S}$ is the set of sizes)
- $\mathsf{min} : \text{Formula} \rightarrow \mathbb{S}$
- $\mathsf{size} : \text{Model} \rightarrow \mathbb{S}$

Properties of max and min:

$$\forall \mathscr{M}, \mathscr{M} \models A \text{ implies } \mathsf{min}(A) \leqslant \mathsf{size}(\mathscr{M}) \leqslant \mathsf{max}(A)$$

Disproving Technique

$$(\exists \mathscr{M}, \mathscr{M} \models A \wedge \neg \mathscr{M} \models B) \text{ implies } A \not\models B$$

$$\forall \mathscr{M}, \mathscr{M} \models A \text{ implies } \mathsf{min}(A) \leqslant \mathsf{size}(\mathscr{M}) \leqslant \mathsf{max}(A)$$

$$\downarrow$$

$$\mathsf{max}(A) < \mathsf{min}(B) \text{ implies } A \not\models B$$

Disproving Technique

$$(\exists \mathscr{M}, \mathscr{M} \models A \wedge \neg \mathscr{M} \models B) \text{ implies } A \not\models B$$

$$\forall \mathscr{M}, \mathscr{M} \models A \text{ implies } \mathsf{min}(A) \leqslant \mathsf{size}(\mathscr{M}) \leqslant \mathsf{max}(A)$$

$$\downarrow$$

$$\mathsf{max}(A) < \mathsf{min}(B) \text{ implies } A \not\models B$$

Defining size (1)

- $\mathsf{size}(\mathscr{M}) \triangleq$ sum of $\mathscr{M}$'s permissions
- size: Model $\rightarrow \mathbb{Q}$

Defining size (1)

- $\mathsf{size}(\mathscr{M}) \stackrel{\Delta}{=}$ sum of $\mathscr{M}$'s permissions
- size: Model $\to \mathbb{Q}$

$$\mathsf{size}((245, \tfrac{1}{2}) :: (246, 1) :: (245, \tfrac{1}{3}) :: []) = \tfrac{1}{2} + 1 + \tfrac{1}{3} = \tfrac{11}{6}$$

Defining max/min (1)

$\mathsf{max}(\_ \overset{\pi}{\mapsto} \_) = \pi$ $\qquad$ $\mathsf{min}(\_ \overset{\pi}{\mapsto} \_) = \pi$

$\mathsf{max}(A \star B) = \mathsf{max}(A) +_{\mathbb{Q}} \mathsf{max}(B)$ $\qquad$ $\mathsf{min}(A \star B) = \mathsf{min}(A) +_{\mathbb{Q}} \mathsf{min}(B)$

$$\mathscr{M} \models a \overset{\pi}{\mapsto} \_ \quad \text{iff} \quad \mathscr{M} = (a, \pi)$$

$$\mathscr{M} \models A \star B \quad \text{iff} \quad \exists \mathscr{M}_A, \mathscr{M}_B, \mathscr{M} = \mathscr{M}_A \uplus \mathscr{M}_B, \mathscr{M}_A \models A \text{ and } \mathscr{M}_B \models B$$

Defining max/min (1)

$$\mathsf{max}(\_ \overset{\pi}{\mapsto} \_) = \pi \qquad \mathsf{min}(\_ \overset{\pi}{\mapsto} \_) = \pi$$

$$\mathsf{max}(A \star B) = \mathsf{max}(A) +_{\mathbb{Q}} \mathsf{max}(B) \qquad \mathsf{min}(A \star B) = \mathsf{min}(A) +_{\mathbb{Q}} \mathsf{min}(B)$$

$$\mathsf{max}(A \mathrel{-\!\star} B) = \mathsf{max}(B) -_{\mathbb{Q}} \mathsf{min}(A) \qquad \mathsf{min}(A \mathrel{-\!\star} B) = \mathsf{min}(B) -_{\mathbb{Q}} \mathsf{max}(A)$$

$$\mathscr{M} \models o \overset{\pi}{\mapsto} \_ \quad \text{iff} \quad \mathscr{M} = (o, \pi)$$

$$\mathscr{M} \models A \star B \quad \text{iff} \quad \exists \mathscr{M}_A, \mathscr{M}_B, \mathscr{M} = \mathscr{M}_A \uplus \mathscr{M}_B, \mathscr{M}_A \models A \text{ and } \mathscr{M}_B \models B$$

$$\mathscr{M} \models A \mathrel{-\!\star} B \quad \text{iff} \quad \forall \mathscr{M}_A, \mathscr{M}_A \models A \text{ and } \mathscr{M}_A \cap \mathscr{M} = \emptyset \text{ implies } \mathscr{M}_A \uplus \mathscr{M} \models A \star B$$

Defining max/min (1)

$$\begin{aligned} \mathsf{max}(A \wedge B) &= \mathsf{min}_{\mathbb{Q}}(\mathsf{max}(A), \mathsf{max}(B)) & \mathsf{min}(A \wedge B) &= \mathsf{max}_{\mathbb{Q}}(\mathsf{min}(A), \mathsf{min}(B)) \\ \mathsf{max}(A \vee B) &= \mathsf{max}_{\mathbb{Q}}(\mathsf{max}(A), \mathsf{max}(B)) & \mathsf{min}(A \vee B) &= \mathsf{min}_{\mathbb{Q}}(\mathsf{min}(A), \mathsf{min}(B)) \end{aligned}$$

$$\begin{aligned} \mathscr{M} \models A \wedge B \quad &\text{iff} \quad \mathscr{M} \models A \text{ and } \mathscr{M} \models B \\ \mathscr{M} \models A \vee B \quad &\text{iff} \quad \mathscr{M} \models A \text{ or } \mathscr{M} \models B \end{aligned}$$

Demo

Demo

$$0 \xmapsto{\frac{1}{2}} \_ \star 0 \xmapsto{\frac{1}{4}} \_ \overset{?}{\vdash} 0 \xmapsto{1} \_$$

$$0 \xmapsto{\frac{1}{2}} \_ \star 0 \xmapsto{\frac{1}{4}} \_ \star 2 \xmapsto{\frac{1}{4}} \_ \star 3 \xmapsto{1} \_ \overset{?}{\vdash} \big((0 \xmapsto{1} \_ \star 1 \xmapsto{\frac{1}{2}} \_) \wedge \big(1 \xmapsto{\frac{1}{2}} \_ \star 0 \xmapsto{1} \_)\big) \star 3 \xmapsto{1} \_$$

Refinement and Extension

Previously:

- Whole heap abstraction

↳ $\mathsf{size}((245, \frac{1}{2}) :: (246, 1) :: (245, \frac{1}{3}) :: []) = \frac{1}{2} + 1 + \frac{1}{3} = \frac{11}{6}$

↳ Information on different addresses is lost.

Refinement and Extension

Previously:

- Whole heap abstraction

↳ $\mathsf{size}((245, \frac{1}{2}) :: (246, 1) :: (245, \frac{1}{3}) :: []) = \frac{1}{2} + 1 + \frac{1}{3} = \frac{11}{6}$

↳ Information on different addresses is lost.

Next slides:

- Per address abstraction.
- Pure formulas

↳ Semantics of pure formulas is permission-independent.

Per Address Abstraction

Previously:

- $\mathsf{max} : \text{Formula} \rightarrow \mathbb{Q}$
- $\mathsf{min} : \text{Formula} \rightarrow \mathbb{Q}$
- $\mathsf{max}(A) < \mathsf{min}(B)$ where $<$ is on $\mathbb{Q}$.

Now:

- $\mathsf{max} : \text{Formula} \rightarrow \text{Model}$
- $\mathsf{min} : \text{Formula} \rightarrow \text{Model}$
- $\mathsf{max}(A) < \mathsf{min}(B)$ where $<$ is on Model.

Defining max/min (2)

Previously:

$\mathsf{max}(\_ \overset{\pi}{\mapsto} \_) = \pi$ $\mathsf{min}(\_ \overset{\pi}{\mapsto} \_) = \pi$

$\mathsf{max}(A \star B) = \mathsf{max}(A) +_{\mathbb{Q}} \mathsf{max}(B)$ $\mathsf{min}(A \star B) = \mathsf{min}(A) +_{\mathbb{Q}} \mathsf{min}(B)$

Now:

$\mathsf{max}(a \overset{\pi}{\mapsto} \_) = (a, \pi) :: []$ $\mathsf{min}(a \overset{\pi}{\mapsto} \_) = (a, \pi) :: []$

$\mathsf{max}(A \star B) = \mathsf{max}(A) \,@\, \mathsf{max}(B)$ $\mathsf{min}(A \star B) = \mathsf{min}(A) \,@\, \mathsf{min}(B)$

Defining max/min (2)

Previously:

$$\mathsf{max}(A \wedge B) = \mathsf{min}_{\mathbb{Q}}(\mathsf{max}(A), \mathsf{max}(B)) \qquad \mathsf{min}(A \wedge B) = \mathsf{max}_{\mathbb{Q}}(\mathsf{min}(A), \mathsf{min}(B))$$
$$\mathsf{max}(A \vee B) = \mathsf{max}_{\mathbb{Q}}(\mathsf{max}(A), \mathsf{max}(B)) \qquad \mathsf{min}(A \vee B) = \mathsf{min}_{\mathbb{Q}}(\mathsf{min}(A), \mathsf{min}(B))$$

Now:

$$\mathsf{max}(A \wedge B) = \mathsf{min}_{\mathscr{M}}(\mathsf{max}(A), \mathsf{max}(B)) \qquad \mathsf{min}(A \wedge B) = \mathsf{max}_{\mathscr{M}}(\mathsf{min}(A), \mathsf{min}(B))$$
$$\mathsf{max}(A \vee B) = \mathsf{max}_{\mathscr{M}}(\mathsf{max}(A), \mathsf{max}(B)) \qquad \mathsf{min}(A \vee B) = \mathsf{min}_{\mathscr{M}}(\mathsf{min}(A), \mathsf{min}(B))$$

- $\mathsf{max}_{\mathscr{M}}$: Per address maximum
- $\mathsf{min}_{\mathscr{M}}$: Per address minimum

Defining max/min (2)

$$\mathsf{max}(A \wedge B) = \mathsf{min}_{\mathscr{M}}(\mathsf{max}(A), \mathsf{max}(B)) \qquad \mathsf{min}(A \wedge B) = \mathsf{max}_{\mathscr{M}}(\mathsf{min}(A), \mathsf{min}(B))$$

$$\mathsf{max}(A \vee B) = \mathsf{max}_{\mathscr{M}}(\mathsf{max}(A), \mathsf{max}(B)) \qquad \mathsf{min}(A \vee B) = \mathsf{min}_{\mathscr{M}}(\mathsf{min}(A), \mathsf{min}(B))$$

- $\mathsf{max}_{\mathscr{M}}$: Per address maximum
- $\mathsf{min}_{\mathscr{M}}$: Per address minimum

$$\mathsf{max}\left((245, \tfrac{1}{2}) :: (245, \tfrac{1}{2}) :: [] \quad , \quad (245, \tfrac{1}{2}) :: (246, 1) :: [] \right)$$

$$=$$

$$(245, \tfrac{1}{2}) :: (245, \tfrac{1}{2}) :: (246, 1) :: []$$

Pure Formulas

Pure formulas include:

- Address comparison: $a = a'$, $a \neq a'$.

↳ With arithmetic: $a + a' = b$.

- …

Pure Formulas

Pure formulas include:

- Address comparison: $a = a'$, $a \neq a'$.

↳ With arithmetic: $a + a' = b$.

- …

Semantics of a pure formula A^p:

$$\mathcal{M} \models A^p \quad \text{iff} \quad \mathsf{oracle}(A^p)$$

↳ No size constraint on $\mathcal{M}$

Pure Formulas

$$\mathscr{M} \models A^p \quad \text{iff} \quad \mathsf{oracle}(A^p)$$

- ↳ No size constraint on $\mathscr{M}$
- ↳ We add $\top$ in max/min's range.
- ↳ $\mathsf{max}(A) = \top$: A's models cannot be max-bounded.

$$\mathsf{max}(A^p) = \top \qquad \mathsf{min}(A^p) = []$$

$\top$ Does Not Harm Bounding Too Much

- A^p a subformula of B does not imply $\mathsf{max}(B) = \top$ (see case $\wedge$).

$$\mathsf{max}(A \star B) \quad = \quad \begin{cases} \top & \text{iff } A = \top \text{ or } B = \top \\ \mathsf{max}(A) @ \mathsf{max}(B) & \text{otherwise} \end{cases}$$

$$\mathsf{max}(A \wedge B) \quad = \quad \begin{cases} \top & \text{iff } A = \top \text{ and } B = \top \\ \mathsf{max}(A) & \text{if } B = \top \\ \mathsf{max}(B) & \text{if } A = \top \\ \mathsf{min}_{\mathscr{M}}(\mathsf{max}(A), \mathsf{max}(B)) & \text{otherwise} \end{cases}$$

Conclusion

- Lightweight method for disproving entailment for an undecidable fragment of separation logic
- Two different abstractions of different precision
- Certified with Coq

Conclusion

- Lightweight method for disproving entailment for an undecidable fragment of separation logic
- Two different abstractions of different precision
- Certified with Coq

- Deal with fractional permissions (this talk)
- Deal with counting permissions (work in progress)

Future Work

1. Unified model of permissions (fractional + counting)
2. Intuitionistic flavor of separation logic
3. Extend the mechanical proof to quantifiers
4. Abstraction mechanisms (Parkinson's abstract predicates)