Making Adequacy of Iris Satisfying

Simon Spies, MPI-SWS, Germany

4th Iris Workshop, June 2024

Making Adequacy of Iris Satisfying

Simon Spies, MPI-SWS, Germany

4th Iris Workshop, June 2024

What is Adequacy?

Adequacy extracts information about programs from Iris.

Examples

$$\text{wp } e \, \{v.\, \text{True}\} \xrightarrow{\text{adequacy}} e \text{ is safe}$$

$$\text{wp } e \, \{v.\, v = 42\} \xrightarrow{\text{adequacy}} e \text{ is safe and returns only } 42$$

$$\text{wp } e \, \{v.\, \text{False}\} \xrightarrow{\text{adequacy}} e \text{ is safe and diverges}$$

Why leave Iris?
Let's play a game!

Which entailments hold?

Example

$\vdash$ True is **true** $\vdash$ False is **false**

Step-Indexing

$\vdash \triangleright^n$ False is **???** $\vdash \exists n.\ \triangleright^n$ False is **???**

Resources with Persistency

$\vdash \{\Box\, \ell \mapsto 41\}\ !\ell\ \{w.\ w = 42\}$ is **???**

Invariants

$\boxed{\exists v.\ \ell \mapsto_{1/2} v}^{\mathcal{N}} * \boxed{\ell \mapsto_{3/4} 42}^{\mathcal{N}} \vdash \Rrightarrow_{\top}$ False is **???**

Which entailments hold?

Example

$\vdash$ True is **true** $\qquad$ $\vdash$ False is **false**

Step-Indexing

$\vdash \triangleright^n$ False is **false** $\qquad$ $\vdash \exists n.\ \triangleright^n$ False is **???**

Resources with Persistency

$\vdash \{\Box\, \ell \mapsto 41\}\ !\ell\ \{w.\ w = 42\}$ is **???**

Invariants

$\boxed{\exists v.\ \ell \mapsto_{1/2} v}^{\mathcal{N}} * \boxed{\ell \mapsto_{3/4} 42}^{\mathcal{N}} \vdash {\Rrightarrow_{\top}}$ False is **???**

Which entailments hold?

Example

$\vdash$ True is **true** $\qquad$ $\vdash$ False is **false**

Step-Indexing

$\vdash \triangleright^n$ False is **false** $\qquad$ $\vdash \exists n.\ \triangleright^n$ False is **true**

Resources with Persistency

$\vdash \{\square\, \ell \mapsto 41\}\ !\ell\ \{w.\ w = 42\}$ is **???**

Invariants

$\boxed{\exists v.\ \ell \mapsto_{1/2} v}^{\mathcal{N}} * \boxed{\ell \mapsto_{3/4} 42}^{\mathcal{N}} \vdash \, \models\!\!\Rightarrow_{\top}$ False is **???**

Which entailments hold?

Example

$\vdash$ True is **true** $\qquad\qquad$ $\vdash$ False is **false**

Step-Indexing

$\vdash \triangleright^{n}$ False is **false** $\qquad\qquad$ $\vdash \exists n.\ \triangleright^{n}$ False is **true**

Resources with Persistency

$\vdash \{\Box\, \ell \mapsto 41\}\ !\ell\ \{w.\ w = 42\}$ is **true**

Invariants

$\boxed{\exists v.\ \ell \mapsto_{1/2} v}^{\mathcal{N}} * \boxed{\ell \mapsto_{3/4} 42}^{\mathcal{N}} \vdash \models\!\!\!\Rrightarrow_{\top}$ False is **???**

Which entailments hold?

Example

$\vdash$ True is **true** $\vdash$ False is **false**

Step-Indexing

$\vdash \triangleright^n$ False is **false** $\vdash \exists n.\ \triangleright^n$ False is **true**

Resources with Persistency

$\vdash \{\Box\, \ell \mapsto 41\}\ !\ell\ \{w.\ w = 42\}$ is **true**

Invariants

$\boxed{\exists v.\ \ell \mapsto_{1/2} v}^{\mathcal{N}} * \boxed{\ell \mapsto_{3/4} 42}^{\mathcal{N}} \vdash\ \Rrightarrow_{\top}$ False is **false**

Which entailments hold?

Example

$\vdash$ True is **true** $\qquad\qquad$ $\vdash$ False is **false**

Step-Indexing

Take Home Message. Iris has many non-trivial features. By proving adequacy, we **avoid trusting Iris and ourselves.**

$\vdash \{\Box\, \ell \mapsto 41\}\ !\ell\ \{w.\ w = 42\}$ is **true**

Invariants

$\boxed{\exists v.\ \ell \mapsto_{1/2} v}^{\mathcal{N}} * \boxed{\ell \mapsto_{3/4} 42}^{\mathcal{N}} \vdash \Rrightarrow_{\top}$ False is **false**

How do we prove adequacy?

There are currently three options ...

Proving Adequacy, Option 1

Option 1. Don't! Use HeapLang Adequacy.

If you use HeapLang, you can instantiate its adequacy theorem:

Adequacy (HeapLang). Let e be an expression and σ a heap. If $\vdash$ wp $e\ \{v.\ \phi\, v\}$, then

1. (e, σ) is safe to execute and
2. all possible return values of e satisfy ϕ

Proving Adequacy, Option 2

Option 2. Use the generic adequacy theorem.

If you instantiate the standard Iris weakest precondition, you can use:

adequacy.v

```
Lemma wp_strong_adequacy_gen (hlc : has_lc) Σ Λ `{!invGpreS Σ} s es σ1 n κs t2 σ2 φ
        (num_laters_per_step : nat → nat) :
  (* WP *)
  (∀ `{Hinv : !invGS_gen hlc Σ},
      ⊢ |={⊤}=> ∃
         (stateI : state Λ → nat → list (observation Λ) → nat → iProp Σ)
         (Φs : list (val Λ → iProp Σ))
         (fork_post : val Λ → iProp Σ)
         (* Note: existentially quantifying over Iris goal! [iExists _] should
         usually work. *)
         state_interp_mono,
       let _ : irisGS_gen hlc Λ Σ := IrisG Hinv stateI fork_post num_laters_per_step
                                     state_interp_mono
       in
       stateI σ1 0 κs 0 ∗
       ([∗ list] e;Φ ∈ es;Φs, WP e @ s; ⊤ {{ Φ }}) ∗
       (∀ es' t2',
         (* es' is the final state of the initial threads, t2' the rest *)
         ⌜ t2 = es' ++ t2' ⌝ -∗
         (* es' corresponds to the initial threads *)
         ⌜ length es' = length es ⌝ -∗
         (* If this is a stuck-free triple (i.e. [s = NotStuck]), then all
         threads in [t2] are not stuck *)
         ⌜ ∀ e2, s = NotStuck → e2 ∈ t2 → not_stuck e2 σ2 ⌝ -∗
         (* The state interpretation holds for [σ2] *)
         stateI σ2 n [] (length t2') -∗
         (* If the initial threads are done, their post-condition [Φ] holds *)
         ([∗ list] e;Φ ∈ es';Φs, from_option Φ True (to_val e)) -∗
         (* For all forked-off threads that are done, their postcondition
           [fork_post] holds. *)
         ([∗ list] v ∈ omap to_val t2', fork_post v) -∗
         (* Under all these assumptions, and while opening all invariants, we
         can conclude [φ] in the logic. After opening all required invariants,
         one can use [fupd_mask_subseteq] to introduce the fancy update. *)
         |={⊤,∅}=> ⌜ φ ⌝)) →
  nsteps n (es, σ1) κs (t2, σ2) →
  (* Then we can conclude [φ] at the meta-level. *)
  φ.
```

Here is the post ϕ

that we are after.

Proving Adequacy, Option 3

Option 3. Start from scratch.

Otherwise, you can use the generic soundness result of Iris by hammering your assertion into this shape

$$\frac{\vdash \pounds m \mathrel{-\!\!*} {}^{\top}{\mid\!\!\Rrightarrow}^{\emptyset} \left({}^{\emptyset}{\mid\!\!\Rrightarrow}^{\emptyset} \triangleright {}^{\emptyset}{\mid\!\!\Rrightarrow}^{\emptyset}\right)^{n} P \qquad P \ \textit{plain}}{\vdash P}$$

Proving Adequacy, Option 3

Option 3. Start from scratch.

Otherwise, you can use the generic soundness result of Iris
by hammering your assertion into this shape

$$\frac{\vdash \pounds m \twoheadrightarrow {}^{\top}{\Rrightarrow}^{\emptyset} \left({}^{\emptyset}{\Rrightarrow}^{\emptyset} \triangleright {}^{\emptyset}{\Rrightarrow}^{\emptyset}\right)^{n} P \qquad P \textit{ plain}}{\vdash P}$$

Example: The Weakest Precondition

$$\mathsf{wp}^{\mathcal{E}}\, v\, \{\Phi\} \triangleq {\Rrightarrow}_{\mathcal{E}}\, \Phi\, v$$

$$\mathsf{wp}^{\mathcal{E}}\, e\, \{\Phi\} \triangleq \forall \sigma, n_s, \vec{\kappa}, \vec{\kappa}', n_t.\ S(\sigma, n_s, \vec{\kappa} +\!\!+ \vec{\kappa}', n_t) \twoheadrightarrow {}^{\mathcal{E}}{\Rrightarrow}^{\emptyset}\, \mathsf{red}(e, \sigma) \ast \qquad (e \notin \mathit{Val})$$

$$\forall e', \sigma', \vec{e}.\ (e, \sigma \xrightarrow{\vec{\kappa}} e', \sigma', \vec{e}) \twoheadrightarrow \pounds\,(n_{\triangleright}(n_s) + 1) \twoheadrightarrow ({\Rrightarrow}_{\emptyset} \triangleright {\Rrightarrow}_{\emptyset})^{n_{\triangleright}(n_s)+1}\, {}^{\emptyset}{\Rrightarrow}^{\mathcal{E}}$$

$$S(\sigma', n_s{+}1, \vec{\kappa}', n_t{+}|\vec{e}|) \ast \mathsf{wp}^{\mathcal{E}}\, e'\, \{\Phi\} \ast \mathop{\Huge\ast}_{e'' \in \vec{e}} \mathsf{wp}^{\top}\, e''\, \{\mathsf{True}\}$$

Proving Adequacy, Option 3

Option 3. Start from scratch.

Otherwise, you can use the generic soundness result of Iris

by hammering your assertion into this shape

$$\frac{\vdash \pounds\, m \twoheadrightarrow {}^{\top}{\Rrightarrow}^{\emptyset}\,({}^{\emptyset}{\Rrightarrow}^{\emptyset} \triangleright {}^{\emptyset}{\Rrightarrow}^{\emptyset})^{n}\, P \qquad P\ \textit{plain}}{\vdash P}$$

what if we add a third modality?

Example: ... ition

$$\mathsf{wp}^{\mathcal{E}}\, v\, \{\Phi\} = {\Rrightarrow}_{\mathcal{E}}\, \Phi\, v$$

$$\mathsf{wp}^{\mathcal{E}}\, e\, \{\Phi\} \triangleq \forall \sigma, n_s, \vec{\kappa}, \vec{\kappa}', n_t.\, S(\sigma, n_s, \vec{\kappa} +\!\!+\, \vec{\kappa}', n_t) \twoheadrightarrow {}^{\mathcal{E}}{\Rrightarrow}^{\emptyset}\, \mathsf{red}(e, \sigma) * \qquad (e \notin \mathit{Val})$$

$$\forall e', \sigma', \vec{e}.\, (e, \sigma \xrightarrow{\vec{\kappa}} e', \sigma', \vec{e}) \twoheadrightarrow \pounds\,(n_{\triangleright}(n_s) + 1) \twoheadrightarrow ({\Rrightarrow}_{\emptyset} \triangleright {\Rrightarrow}_{\emptyset})^{n_{\triangleright}(n_s)+1}\, {}^{\emptyset}{\Rrightarrow}^{\mathcal{E}}$$

$$S(\sigma', n_s{+}1, \vec{\kappa}', n_t{+}|\vec{e}|) * \mathsf{wp}^{\mathcal{E}}\, e'\, \{\Phi\} * \mathop{\huge *}_{e'' \in \vec{e}} \mathsf{wp}^{\top}\, e''\, \{\mathsf{True}\}$$

Is there a **more user-friendly** and

no monolithic lemmas to instantiate

more compositional proof strategy?

with rules for individual modalities

Satisfiability

the basic idea

Scaling Up

later credits, fancy updates, …

Models

different models for satisfiability

Satisfiability

the basic idea

Scaling Up

later credits, fancy updates, …

Models

different models for satisfiability

Let’s start simple …

Simple, Sequential Weakest Precondition

$$
\begin{aligned}
\mathsf{wp}\ v\ \{\Phi\} &\triangleq\ {\mid\!\Rrightarrow} \Phi\, v \\
\mathsf{wp}\ e\ \{\Phi\} &\triangleq \forall \sigma.\ S(\sigma) \mathrel{-\!\!*} {\mid\!\Rrightarrow} \mathsf{red}(e, \sigma) * & e \notin \mathit{Val} \\
&\qquad (\forall \sigma', e'.\ (e, \sigma) \to (e', \sigma') \mathrel{-\!\!*} {\mid\!\Rrightarrow} \triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})
\end{aligned}
$$

Let's start simple …

Simple, Sequential W

we want to get information from here

$$\mathsf{wp}\ v\ \{\Phi\} \triangleq {\models\!\!\!\Rrightarrow} \Phi\ v$$

$$\mathsf{wp}\ e\ \{\Phi\} \triangleq \forall \sigma.\ S(\sigma) \twoheadrightarrow {\models\!\!\!\Rrightarrow} \mathsf{red}(e, \sigma) * \qquad\qquad e \notin \mathit{Val}$$

$$(\forall \sigma', e'.\ (e, \sigma) \to (e', \sigma') \twoheadrightarrow {\models\!\!\!\Rrightarrow} \triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$$

Let’s start simple …

Simple, Sequential W

we want to get information from here

$$\mathsf{wp}\; v\; \{\Phi\} \triangleq\; \Rrightarrow \color{red}{\Phi\, v}$$

$$\mathsf{wp}\; e\; \{\Phi\} \triangleq \forall \sigma.\, S(\sigma) \twoheadrightarrow\; \Rrightarrow \mathsf{red}(e, \sigma) * \qquad\qquad e \notin \mathit{Val}$$
$$(\forall \sigma', e'.\, (e, \sigma) \to (e', \sigma') \twoheadrightarrow\; \Rrightarrow \triangleright S(\sigma') * \mathsf{wp}\; e'\; \{\Phi\})$$

Observation. The information we want is guarded by assumptions, assertions, and **various Iris modalities**.

The proof strategy for adequacy

Since in Iris **changes to the logical state**

decrease the step-index, update the resources

are expressed **by nesting modalities**,

$\triangleright P$, $\vert\!\Rrightarrow P$, ${}^{\mathcal{E}_1}\vert\!\Rrightarrow^{\mathcal{E}_2} P$, **wp** $e\ \{P\}$, …

we **peel off these modalities one-by-one**.

turn $\triangleright P$ into P, turn $\vert\!\Rrightarrow P$ into P, …

The Key: Satisfiability

Satisfiability

$$\mathrm{sat} : \underbrace{iProp}_{\textbf{Iris}} \rightarrow \underbrace{Prop}_{\textbf{Coq}}$$

The Key: Satisfiability

Satisfiability

$$\mathsf{sat} : \underbrace{\textit{iProp}}_{\textbf{Iris}} \to \underbrace{\textit{Prop}}_{\textbf{Coq}}$$

SAT-INTRO
$$\frac{}{\mathsf{sat}\ \mathsf{True}}$$

SAT-MONO
$$\frac{\mathsf{sat}\ P \qquad P \vdash Q}{\mathsf{sat}\ Q}$$

SAT-LATER
$$\frac{\mathsf{sat}\ (\triangleright P)}{\mathsf{sat}\ P}$$

SAT-UPD
$$\frac{\mathsf{sat}\ (\Rrightarrow P)}{\mathsf{sat}\ P}$$

SAT-ELIM
$$\frac{\mathsf{sat}\ \phi}{\phi}$$

Example: Simple Weakest Precondition

Theorem. If $\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{v.\ \phi\, v\})$ and $(e, \sigma) \rightarrow^n (v, \sigma')$, then $\phi\, v$.

Lemma 1.

$$\frac{\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\}) \qquad (e, \sigma) \rightarrow (e', \sigma')}{\mathsf{sat}(S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})}$$

Lemma 2.

$$\frac{\mathsf{sat}(S(\sigma) * \mathsf{wp}\ v\ \{w.\ \phi\, w\})}{\phi\, v}$$

Satisfiability in Action, Lemma 1

Lemma 1.

$$\frac{\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\}) \qquad (e, \sigma) \to (e', \sigma')}{\mathsf{sat}(S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})}$$

Proof

$\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\})$

Definition

$\mathsf{wp}\ e\ \{\Phi\} \triangleq \forall \sigma.\ S(\sigma) \mathrel{-\!\!*} \Rrightarrow \mathsf{red}(e, \sigma) *$
$(\forall \sigma', e'.\ (e, \sigma) \to (e', \sigma') \mathrel{-\!\!*} \Rrightarrow \triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

Rule

Satisfiability in Action, Lemma 1

Lemma 1.

$$\frac{\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\}) \qquad (e, \sigma) \rightarrow (e', \sigma')}{\mathsf{sat}(S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})}$$

Proof

$\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\})$

$\mathsf{sat}(\mid\!\Rrightarrow \mathsf{red}(e, \sigma) * (\forall \sigma', e'. \cdots))$

Definition

$\mathsf{wp}\ e\ \{\Phi\} \triangleq \forall \sigma.\ S(\sigma) \twoheadrightarrow\!\!* \mid\!\Rrightarrow \mathsf{red}(e, \sigma) *$
$(\forall \sigma', e'.\ (e, \sigma) \rightarrow (e', \sigma') \twoheadrightarrow\!\!* \mid\!\Rrightarrow \triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

Rule

SAT-MONO

$$\frac{\mathsf{sat}\ P \qquad P \vdash Q}{\mathsf{sat}\ Q}$$

Satisfiability in Action, Lemma 1

Lemma 1.

$$\frac{\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\}) \qquad (e,\sigma) \to (e',\sigma')}{\mathsf{sat}(S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})}$$

Proof

$\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\})$

$\mathsf{sat}(\Rrightarrow \mathsf{red}(e,\sigma) * (\forall \sigma', e'.\ \cdots))$

$\mathsf{sat}(\mathsf{red}(e,\sigma) * (\forall \sigma', e'.\ \cdots))$

Definition

$\mathsf{wp}\ e\ \{\Phi\} \triangleq \forall \sigma.\ S(\sigma) \twoheadrightarrow \Rrightarrow \mathsf{red}(e,\sigma) *$
$(\forall \sigma', e'.\ (e,\sigma) \to (e',\sigma') \twoheadrightarrow \Rrightarrow \triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

Rule

SAT-UPD

$$\frac{\mathsf{sat}\ (\Rrightarrow P)}{\mathsf{sat}\ P}$$

Satisfiability in Action, Lemma 1

Lemma 1.

$$\frac{\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\}) \qquad (e, \sigma) \to (e', \sigma')}{\mathsf{sat}(S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})}$$

Proof

$\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\})$

$\mathsf{sat}(\Rrightarrow \mathsf{red}(e, \sigma) * (\forall \sigma', e'.\ \cdots))$

$\mathsf{sat}(\mathsf{red}(e, \sigma) * (\forall \sigma', e'.\ \cdots))$

$\mathsf{sat}(\forall \sigma', e'. (e, \sigma) \to (e', \sigma') \mathrel{-\!\!*} \Rrightarrow S(\sigma') * \triangleright \mathsf{wp}\ e'\ \{\Phi\})$

Definition

$\mathsf{wp}\ e\ \{\Phi\} \triangleq \forall \sigma.\ S(\sigma) \mathrel{-\!\!*} \Rrightarrow \mathsf{red}(e, \sigma) *$
$(\forall \sigma', e'.\ (e, \sigma) \to (e', \sigma') \mathrel{-\!\!*} \Rrightarrow \triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

Rule

SAT-MONO

$$\frac{\mathsf{sat}\ P \qquad P \vdash Q}{\mathsf{sat}\ Q}$$

Satisfiability in Action, Lemma 1

Lemma 1.

$$\frac{\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\}) \qquad (e, \sigma) \rightarrow (e', \sigma')}{\mathsf{sat}(S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})}$$

Proof

$\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\})$

$\mathsf{sat}(|\Rrightarrow \mathsf{red}(e, \sigma) * (\forall \sigma', e'.\ \cdots))$

$\mathsf{sat}(\mathsf{red}(e, \sigma) * (\forall \sigma', e'.\ \cdots))$

$\mathsf{sat}(\forall \sigma', e'. (e, \sigma) \rightarrow (e', \sigma') \mathrel{-\!\!*} |\Rrightarrow S(\sigma') * \triangleright \mathsf{wp}\ e'\ \{\Phi\})$

$\mathsf{sat}(|\Rrightarrow \triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

Definition

$\mathsf{wp}\ e\ \{\Phi\} \triangleq \forall \sigma.\ S(\sigma) \mathrel{-\!\!*} |\Rrightarrow \mathsf{red}(e, \sigma) *$
$(\forall \sigma', e'.\ (e, \sigma) \rightarrow (e', \sigma') \mathrel{-\!\!*} |\Rrightarrow \triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

Rule

SAT-MONO

$$\frac{\mathsf{sat}\ P \qquad P \vdash Q}{\mathsf{sat}\ Q}$$

Satisfiability in Action, Lemma 1

Lemma 1.

$$\frac{\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\}) \qquad (e, \sigma) \to (e', \sigma')}{\mathsf{sat}(S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})}$$

Proof

$\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\})$

$\mathsf{sat}(\Rrightarrow \mathsf{red}(e, \sigma) * (\forall \sigma', e'. \cdots))$

$\mathsf{sat}(\mathsf{red}(e, \sigma) * (\forall \sigma', e'. \cdots))$

$\mathsf{sat}(\forall \sigma', e'. (e, \sigma) \to (e', \sigma') \mathrel{-\!\!*} \Rrightarrow S(\sigma') * \triangleright \mathsf{wp}\ e'\ \{\Phi\})$

$\mathsf{sat}(\Rrightarrow \triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

$\mathsf{sat}(\triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

Definition

$\mathsf{wp}\ e\ \{\Phi\} \triangleq \forall \sigma.\ S(\sigma) \mathrel{-\!\!*} \Rrightarrow \mathsf{red}(e, \sigma) *$

$(\forall \sigma', e'.\ (e, \sigma) \to (e', \sigma') \mathrel{-\!\!*} \Rrightarrow \triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

Rule

SAT-UPD

$$\frac{\mathsf{sat}\ (\Rrightarrow P)}{\mathsf{sat}\ P}$$

Satisfiability in Action, Lemma 1

Lemma 1.

$$\frac{\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\}) \qquad (e, \sigma) \to (e', \sigma')}{\mathsf{sat}(S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})}$$

Proof

$\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\})$

$\mathsf{sat}(\mid\!\Rrightarrow \mathsf{red}(e, \sigma) * (\forall \sigma', e'.\ \cdots))$

$\mathsf{sat}(\mathsf{red}(e, \sigma) * (\forall \sigma', e'.\ \cdots))$

$\mathsf{sat}(\forall \sigma', e'. (e, \sigma) \to (e', \sigma') -\!\!* \mid\!\Rrightarrow S(\sigma') * \triangleright \mathsf{wp}\ e'\ \{\Phi\})$

$\mathsf{sat}(\mid\!\Rrightarrow \triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

$\mathsf{sat}(\triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

$\mathsf{sat}(S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

Definition

$\mathsf{wp}\ e\ \{\Phi\} \triangleq \forall \sigma.\ S(\sigma) -\!\!* \mid\!\Rrightarrow \mathsf{red}(e, \sigma) *$
$(\forall \sigma', e'.\ (e, \sigma) \to (e', \sigma') -\!\!* \mid\!\Rrightarrow \triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

Rule

SAT-LATER

$$\frac{\mathsf{sat}\ (\triangleright P)}{\mathsf{sat}\ P}$$

Satisfiability in Action, Lemma 1

Lemma 1.

$$\frac{\mathsf{sat}(S(\sigma) * \mathsf{wp}\ e\ \{\Phi\}) \qquad (e,\sigma) \to (e',\sigma')}{\mathsf{sat}(S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})}$$

We **do not** just shift around proof effort.
The proofs are simplified by **modularity**.

Proof

$\mathsf{sat}(S(\sigma) * \mathsf{w}$ … $\mathsf{d}(e,\sigma) *$

$\mathsf{sat}(\Rrightarrow \mathsf{red}(e,$ … $\triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

$\mathsf{sat}(\mathsf{red}(e,\sigma) * (\forall \sigma', e'.\ \cdots))$

$\mathsf{sat}(\forall \sigma', e'.(e,\sigma) \to (e',\sigma') \mathrel{-\!\!*} \Rrightarrow S(\sigma') * \triangleright \mathsf{wp}\ e'\ \{\Phi\})$

$\mathsf{sat}(\Rrightarrow \triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

$\mathsf{sat}(\triangleright S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

$\mathsf{sat}(S(\sigma') * \mathsf{wp}\ e'\ \{\Phi\})$

Rule

SAT-LATER

$$\frac{\mathsf{sat}\ (\triangleright P)}{\mathsf{sat}\ P}$$

Proving adequacy,

one modality at a time.

Satisfiability

the basic idea

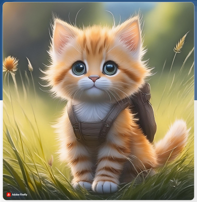
Scaling Up

later credits, fancy updates, …

Models

different models for satisfiability

Adding Frames and Resources



additional resources

credit supply, current view, ...

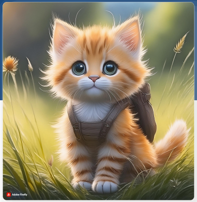
main proposition

$$\mathsf{SAT}_{F}^{[R_1;\ldots;R_n]}\ P \quad \triangleq \quad \mathsf{sat}(F * (\mathop{\Large\ast}_{i=1,\ldots,n} R_i) * P)$$

additional frame

"frame baking"

Adding Frames and Resources



additional resources

credit supply, current view, …

main proposition

$$\mathsf{SAT}_{F}^{[R_1;\ldots;R_n]}\ P \quad \triangleq \quad \mathsf{sat}(F * (\mathop{\huge *}_{i=1,\ldots,n} R_i) * P)$$

additional frame

"frame baking"

Rules

$$\frac{}{\mathsf{SAT}_{\mathsf{True}}^{[]}\ \mathsf{True}} \qquad \frac{\mathsf{SAT}_F^{\vec{R}}\ P \quad P \vdash Q}{\mathsf{SAT}_F^{\vec{R}}\ Q} \qquad \frac{\mathsf{SAT}_F^{\vec{R}}\ (\triangleright P)}{\mathsf{SAT}_F^{\vec{R}}\ P} \qquad \frac{\mathsf{SAT}_F^{\vec{R}}\ (\Rrightarrow P)}{\mathsf{SAT}_F^{\vec{R}}\ P} \qquad \frac{\mathsf{SAT}_F^{\vec{R}}\ \phi}{\phi}$$

Credit Supply Resource

$\pounds_\bullet n$ *total supply of later credits*

Rules

$$\frac{\mathsf{SAT}_F^{[\pounds_\bullet n]}\ (\Rrightarrow_{\mathsf{le}} P)}{\mathsf{SAT}_F^{[\pounds_\bullet n]}\ P}$$

$$\pounds_\bullet n * (\Rrightarrow_{\mathsf{le}} P) \vdash (\Rrightarrow \triangleright)^n \Rrightarrow \diamond \Rrightarrow (\pounds_\bullet n * P)$$

$$\frac{\mathsf{SAT}_F^{[\pounds_\bullet n]}\ (\pounds\, m \twoheadrightarrow P)}{\mathsf{SAT}_F^{[\pounds_\bullet (n+m)]}\ P}$$

$$\pounds_\bullet n \vdash \Rrightarrow (\pounds_\bullet (n+m) * \pounds\, m)$$

Fancy Updates

$$
{}^{\mathcal{E}_1}{\models\!\!\!\Rrightarrow}^{\mathcal{E}_2} \triangleq \underbrace{\text{view}\,\mathcal{E}_1}_{W * \boxed{\mathcal{E}_1}^{\gamma_{En}}} \mathrel{-\!\!*} {\models\!\!\!\Rrightarrow}_{\text{le}} \diamond (\text{view}\,\mathcal{E}_2 * P)
$$

Rules

$$
\frac{\textsf{SAT}_F^{[\pounds_\bullet\, n;\ \text{view}\,\mathcal{E}_1]}\ \left({}^{\mathcal{E}_1}{\models\!\!\!\Rrightarrow}^{\mathcal{E}_2} P\right)}{\textsf{SAT}_F^{[\pounds_\bullet\, n;\ \text{view}\,\mathcal{E}_2]}\ P}
$$

$$
\text{view}\,\mathcal{E}_1 * \left({}^{\mathcal{E}_1}{\models\!\!\!\Rrightarrow}^{\mathcal{E}_2} P\right) \vdash {\models\!\!\!\Rrightarrow}_{\text{le}} \diamond (\text{view}\,\mathcal{E}_2 * P)
$$

The Actual Weakest Precondition

$$\mathsf{wp}^{\mathcal{E}}\, v\, \{\Phi\} \triangleq \mathrel{\mid\!\Rrightarrow}_{\mathcal{E}} \Phi\, v$$

$$\begin{aligned}\mathsf{wp}^{\mathcal{E}}\, e\, \{\Phi\} \triangleq\ & \forall \sigma, n_s, \vec{\kappa}, \vec{\kappa}', n_t.\ S(\sigma, n_s, \vec{\kappa} \mathbin{+\!\!+} \vec{\kappa}', n_t) \mathrel{-\!\!*} {}^{\mathcal{E}}{\mathrel{\mid\!\Rrightarrow}}^{\emptyset}\, \mathsf{red}(e, \sigma) * \qquad (e \notin Val)\\ & \quad \forall e', \sigma', \vec{e}.\ (e, \sigma \xrightarrow{\vec{\kappa}} e', \sigma', \vec{e}) \mathrel{-\!\!*} \pounds\,(n_{\triangleright}(n_s) + 1) \mathrel{-\!\!*}\\ & \qquad (\mathrel{\mid\!\Rrightarrow}_{\emptyset} \triangleright \mathrel{\mid\!\Rrightarrow}_{\emptyset})^{n_{\triangleright}(n_s)+1}\ {}^{\emptyset}{\mathrel{\mid\!\Rrightarrow}}^{\mathcal{E}}\\ & \qquad S(\sigma', n_s{+}1, \vec{\kappa}', n_t{+}|\vec{e}|) * \mathsf{wp}^{\mathcal{E}}\, e'\, \{\Phi\} * \mathop{\text{\Large$\ast$}}_{e'' \in \vec{e}} \mathsf{wp}^{\top}\, e''\, \{\Psi\}\end{aligned}$$

Single-Step Rule

$$\frac{\mathsf{SAT}_F^{[\pounds_\bullet\, n;\ \text{view}\ \mathcal{E}]}\ \big(S(\sigma, n_s, \vec{\kappa} \mathbin{+\!\!+} \vec{\kappa}', n_t) * \mathsf{wp}^{\mathcal{E}}\, e\, \{\Phi\}\big) \qquad (e, \sigma \xrightarrow{\vec{\kappa}} e', \sigma', \vec{e})}{\mathsf{SAT}_F^{[\pounds_\bullet\, m;\ \text{view}\ \mathcal{E}]}\ \Big(S(\sigma', n_s{+}1, \vec{\kappa}', n_t{+}|\vec{e}|) * \mathsf{wp}^{\mathcal{E}}\, e'\, \{\Phi\} * \mathop{\text{\Large$\ast$}}_{e'' \in \vec{e}} \mathsf{wp}^{\top}\, e''\, \{\Psi\}\Big)\ \textit{for some } m}$$

The Actual Weakest Precondition

all of these disappear

$$\mathsf{wp}^{\mathcal{E}}\, v\, \{\Phi\} \triangleq {\Rrightarrow_{\mathcal{E}}}\, \Phi\, v$$

$$\begin{aligned}\mathsf{wp}^{\mathcal{E}}\, e\, \{\Phi\} \triangleq\ & \forall \sigma, n_s, \vec{\kappa}, \vec{\kappa}', n_t.\ S(\sigma, n_s, \vec{\kappa} \mathbin{+\!\!+} \vec{\kappa}', n_t) \mathrel{-\!\!*} {}^{\mathcal{E}}{\Rrightarrow}^{\emptyset}\, \mathsf{red}(e, \sigma) * \qquad (e \notin Val)\\ & \forall e', \sigma', \vec{e}.\ (e, \sigma \xrightarrow{\vec{\kappa}} e', \sigma', \vec{e}) \mathrel{-\!\!*} £\,(n_{\triangleright}(n_s) + 1) \mathrel{-\!\!*}\\ & ({\Rrightarrow_{\emptyset}} \triangleright {\Rrightarrow_{\emptyset}})^{n_{\triangleright}(n_s)+1}\ {}^{\emptyset}{\Rrightarrow}^{\mathcal{E}}\\ & S(\sigma', n_s{+}1, \vec{\kappa}', n_t{+}|\vec{e}|) * \mathsf{wp}^{\mathcal{E}}\, e'\, \{\Phi\} * \mathop{\Huge\ast}_{e'' \in \vec{e}} \mathsf{wp}^{\top}\, e''\, \{\Psi\}\end{aligned}$$

Single-Step Rule

$$\frac{\mathsf{SAT}_F^{[£\bullet\, n;\ \mathsf{view}\ \mathcal{E}]}\ \left(S(\sigma, n_s, \vec{\kappa} \mathbin{+\!\!+} \vec{\kappa}', n_t) * \mathsf{wp}^{\mathcal{E}}\, e\, \{\Phi\}\right) \qquad (e, \sigma \xrightarrow{\vec{\kappa}} e', \sigma', \vec{e})}{\mathsf{SAT}_F^{[£\bullet\, m;\ \mathsf{view}\ \mathcal{E}]}\ \left(S(\sigma', n_s{+}1, \vec{\kappa}', n_t{+}|\vec{e}|) * \mathsf{wp}^{\mathcal{E}}\, e'\, \{\Phi\} * \mathop{\Huge\ast}_{e'' \in \vec{e}} \mathsf{wp}^{\top}\, e''\, \{\Psi\}\right) \text{ for some } m}$$

Satisfiability

the basic idea

Scaling Up

later credits, fancy updates, …

Models

different models for satisfiability

Satisfiability, the basic model

for all step-indices n — exists a resource r

$$\mathsf{sat}\ P \triangleq \forall n.\ \exists r.\ (n, r) \in \mathcal{V} \land (n, r) \in P$$

that is valid — and satisfies P

Rules

SAT-INTRO
$$\frac{}{\mathsf{sat}\ \mathsf{True}}$$

SAT-MONO
$$\frac{\mathsf{sat}\ P \qquad P \vdash Q}{\mathsf{sat}\ Q}$$

SAT-LATER
$$\frac{\mathsf{sat}\ (\triangleright P)}{\mathsf{sat}\ P}$$

SAT-UPD
$$\frac{\mathsf{sat}\ (\Rrightarrow P)}{\mathsf{sat}\ P}$$

SAT-ELIM
$$\frac{\mathsf{sat}\ \phi}{\phi}$$

A Catch: Extracting Choices

Can we get ...

Disjunction

$$\frac{\mathsf{sat}(P \lor Q)}{\mathsf{sat}\,P \lor \mathsf{sat}\,Q}$$

Existential Quantification

$$\frac{\mathsf{sat}(\exists x : X.\ P\ x)}{\exists x.\ \mathsf{sat}(P\ x)}$$

A Catch: Extracting Choices

Can we get ...

Disjunction

$$\frac{\mathsf{sat}(P \lor Q)}{\mathsf{sat}\ P \lor \mathsf{sat}\ Q}$$

yes! (with classical logic)

Existential Quantification

$$\frac{\mathsf{sat}(\exists x : X.\ P\ x)}{\exists x.\ \mathsf{sat}(P\ x)}$$

depends ...

The step-indexing model matters …

Iris without Step-Indexing ✓ **existentials** ✗ **laters**

$$\text{sat } P \triangleq \quad \exists r.\ (1, r) \in \mathcal{V} \wedge (1, r) \in P$$

Standard Iris ✗ **existentials** (only finite) ✓ **laters**

$$\text{sat } P \triangleq \forall n.\ \exists r.\ (n, r) \in \mathcal{V} \wedge (n, r) \in P$$

Transfinite Iris ✓ **existentials** (below *iProp*) ✓ **laters**

$$\text{sat } P \triangleq \forall \alpha.\ \exists r.\ (\alpha, r) \in \mathcal{V} \wedge (\alpha, r) \in P$$

Already applied in …

Transfinite Iris
safety and liveness

Later Credits
safety with later credits

Simuliris/Velliris
termination-preserving refinement

DimSum
stateful language wrappers

Melocoton
safety with angelic non-determinism

If you like this kind of work, come talk to us!

…and soon your project?

`iris/satisfiable-demo`

Initially Allocating Global Resources

For $\underbrace{\textbf{allocating global resources}}_{\text{invariants, later credits, heaps, \ldots}}$, we use

$$\frac{\mathsf{sat}(P) \qquad a \in \overline{\mathcal{V}}}{\exists \gamma.\ \mathsf{sat}(\boxed{a}^{\gamma} * P)}$$

Note. In standard Iris, this property requires a different model. See the accompanying demo for more details.