Let's try to understand (part of) Iris

Willem Penninckx

The Paper

Iris: Monoids and Invariants as an Orthogonal Basis for Concurrent Reasoning

Ralf Jung, David Swasey, Filip Sieczkowski, Kasper Svendsen, Aaron Turon, Lars Birkedal, and Derek Dreyer

DISCLAIMER

I'm not an expert

Concurrency is about shared state

Situation	**Shared state**	**Verify this**
Shared memory	Memory	No secret overwrites, Counter only increases
Message-passing	Network	Protocol
Input/output	Filesystems, Humans, ...	Protocol

How to verify when there's concurrency?

> “ ”
>
> Monoids and invariants are all you need
>
> – Iris

Invariant:
assertion about shared state

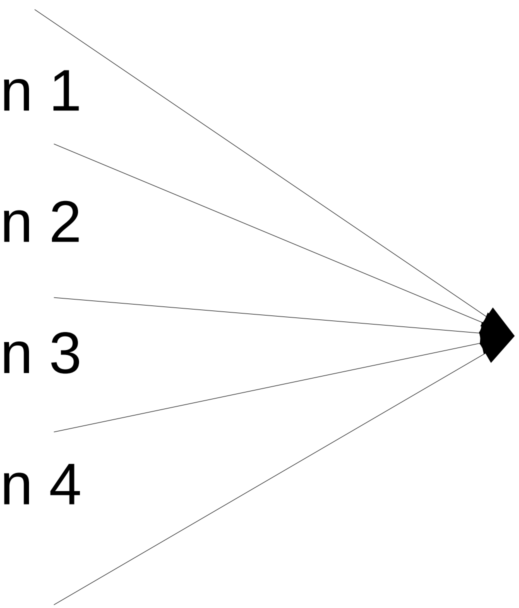
Thread 1:

Atomic operation 1

Atomic operation 2

Atomic operation 3

Atomic operation 4



Assertion holds

(Iris-style) Monoid

“error” element

Commutative binary operator

$$\mathrm{Nat} = (\mathbb{N} \cup \{\bot\}, \bot, 0, +_{\mathbb{N}})$$

Name of
the monoid

Set

$\mathrm{Neutral\ element}(n +_{\mathbb{N}} 0 = n)$

“Case study”: Verification + concurrency + heap

Proglang:
v = malloc()
v1 = !v2
v1 := v2
v1 = v2

Attempt #1

Invariant, e.g.: $\exists h \in \text{Heaps}. \lfloor h \rfloor$

GLOBAL physical state is h

$$M_{\text{h}} = (\text{Heaps} \cup \{\bot\}, \bot, \text{emptyheap}, \cdot_{\text{h}})$$

$$\{h : \text{Locations} \rightharpoonup \text{Values}\}$$

$$\{1 \mapsto 2\} \cdot_h \{7 \mapsto 8\} = \{1 \mapsto 2, 7 \mapsto 8\}$$

$$\{1 \mapsto 2\} \cdot_h \bot = \bot$$

$$\{1 \mapsto 2\} \cdot_h \{1 \mapsto 2\} = \bot$$

Thread 1:

v1 = malloc()

v1 := 7

$\exists h \in \mathrm{Heaps}. \lfloor h \rfloor$

"$\mathit{v1} \mapsto 7$" ?

“partial knowledge” in monoid

$M_a = ($

Set $|M_a|$

$, \perp, \ldots, \cdot_a)$

“partial knowledge” in monoid

Local partial knowledge

$$M_a = (\{\perp\} \cup \{(g, l) \mid \ldots$$

$$\ldots$$

$$\ldots$$

$$, \perp, \ldots, \cdot_a)$$

Global knowledge

“partial knowledge” in monoid

Local partial knowledge

$$M_a = (\{\bot\} \cup \{(g, l) \mid \begin{array}{l} g \in \mathrm{Heaps} \cup \{\epsilon_g\} \\ l \in \mathrm{Heaps} \\ \cdots \end{array}$$

No global knowledge

$$, \bot, \ldots, \cdot_a)$$

Global knowledge

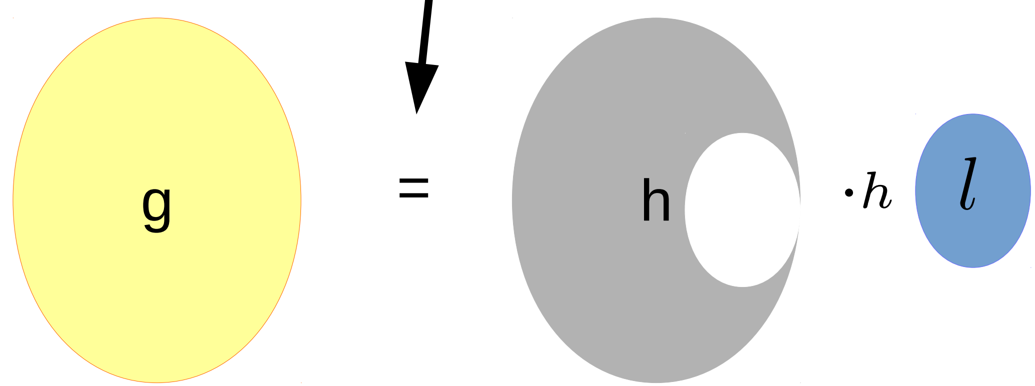
“partial knowledge” in monoid

Local partial knowledge

$$M_a = (\{\perp\} \cup \{(g, l) \mid g \in \text{Heaps} \cup \{\epsilon_g\}$$
$$l \in \text{Heaps}$$
$$g \neq \epsilon_g \rightarrow \exists h.\ g = h \cdot_h l\}$$
$$, \perp, \ldots, \cdot_a)$$

No global knowledge

Global knowledge



“partial knowledge” in monoid

$$M_a = (\{\bot\} \cup \{(g,l)|\; \begin{array}{l} g \in \mathrm{Heaps} \cup \{\epsilon_g\} \\ l \in \mathrm{Heaps} \\ g \neq \epsilon_g \to \exists h.\; g = h \cdot_h l\} \end{array}$$

$$, \bot, \ldots, \cdot_a)$$

Exercise: what does this mean?

$$(\epsilon_g, \mathrm{emptyheap})$$

$$(\epsilon_g, (1 \mapsto 2, 102 \mapsto 7))$$

$$((1 \mapsto 2, 102 \mapsto 7), (1 \mapsto 2, 102 \mapsto 7))$$

$$(\mathrm{emtyheap}, \mathrm{emtyheap})$$

$$M_a = (\{\bot\} \cup \{(g,l) |\; g \in \text{Heaps} \cup \{\epsilon_g\}$$
$$l \in \text{Heaps}$$
$$g \neq \epsilon_g \to \exists h.\; g = h \cdot_h l\}$$
$$, \bot, \ldots, \cdot_a)$$

$$(\epsilon_g, l_1) \cdot_a (g, l_2) = (g, l_1 \cdot_h l_2) \textbf{ if } (g, l_1 \cdot_h l_2) \in |M_a| \backslash \{\bot\}$$
$$m_1 \cdot_a m_2 = \bot \textbf{ other cases}$$

Note: in paper composition
Is just pointwise
(so (\eps, l1) . (\eps l2) is not always \bot)

Exercise: what is the neutral element?

Thread 1:

v1 = malloc()

$\exists h \in \mathrm{Heaps}. \lfloor h \rfloor$

v1 := 7

“$v1 \mapsto 7$” ?

Thread 1:

v1 = malloc()

v1 := 7

$\exists h \in \text{Heaps}. \boxed{(h, \text{emptyheap})} * \lfloor h \rfloor$

$\cancel{\exists h \in \text{Heaps}. \lfloor h \rfloor}$

$\boxed{(\epsilon_g, \{\mathit{v1} \mapsto 7\})}$

Link with physical?

Combined:

$$\boxed{(\epsilon_g, \{\mathit{v1} \mapsto 7\})} * \exists h \in \text{Heaps}. \boxed{(h, \text{emptyheap})} * \lfloor h \rfloor$$

$$= \exists h \in \text{Heaps}. \boxed{(\epsilon_g, \{\mathit{v1} \mapsto 7\}) \cdot_a (h, \text{emptyheap})} * \lfloor h \rfloor$$

$$= \exists h \in \text{Heaps}. \boxed{(h, \{\mathit{v1} \mapsto 7\})} * \lfloor h \rfloor$$

Know $\mathit{v1} \mapsto 7$ in physical state!

Let's prove

$$\left\{\boxed{(\epsilon_g, \{v1 \mapsto 0\})}\right\} \text{ v1 := 7 } \left\{\boxed{(\epsilon_g, \{v1 \mapsto 7\})}\right\}_{\{\iota\}}$$

Our invariant holds

Strategy

- Open invariant
- Combine thread's ghost state with invar's
  - $\boxed{\mathrm{m}_1} * \boxed{\mathrm{m}_2} = \boxed{\mathrm{m}_1 . m_2}$ → Know $a \mapsto \_$ in physical state!
- Do physical update
  - $$\frac{}{\{\lfloor h[a \mapsto v_2] \rfloor\} \quad a := v_2 \; \{\lfloor h[a \mapsto v_2] \rfloor\}}$$
- Do ghost update
- Split thread's ghost state and invar's
- Close invariant

$$\left\{ \boxed{(\epsilon_g, \{\mathit{v1} \mapsto 0\})} \right\}$$

$$\left\{ \boxed{(\epsilon_g, \{\mathit{v1} \mapsto 0\})} * \exists h \in \mathrm{Heaps}. \boxed{(h, \mathrm{emptyheap})} * \lfloor h \rfloor \right\}$$

$$\left\{ \exists h \in \mathrm{Heaps}. \boxed{(\epsilon_g, \{\mathit{v1} \mapsto 0\}) \cdot_a (h, \mathrm{emptyheap})} * \lfloor h \rfloor \right\}$$

$$\left\{ \exists h \in \mathrm{Heaps}. \boxed{(h, \{\mathit{v1} \mapsto 0\})} * \lfloor h \rfloor \right\}$$

$$\left\{ \boxed{(h'[v1 \mapsto 0], \{\mathit{v1} \mapsto 0\})} * \lfloor h'[v1 \mapsto 0] \rfloor \right\}$$

frame

Phys. Upd

$$\left\{ \lfloor h'[v1 \mapsto 0] \rfloor \right\}$$

v1 := 7

$$\left\{ \lfloor h'[v1 \mapsto 7] \rfloor \right\}$$

$$\left\{ \boxed{(h'[v1 \mapsto 0], \{\mathit{v1} \mapsto 0\})} * \lfloor h'[v1 \mapsto 7] \rfloor \right\}$$

Need to update ghost state to close invar

$$\left\{\boxed{(\mathrm{h}'[v1 \mapsto 0], \{v1 \mapsto 0\})} * \lfloor h'[v1 \mapsto 7] \rfloor\right\}$$

???

$$\left\{\boxed{(\mathrm{h}'[v1 \mapsto 7], \{v1 \mapsto 7\})} * \lfloor h'[v1 \mapsto 7] \rfloor\right\}$$

$$\left\{\boxed{(\epsilon_g, \{v1 \mapsto 7\})} * \exists h \in \mathrm{Heaps}. \boxed{(h, \mathrm{emptyheap})} * \lfloor h \rfloor\right\}$$

$$\left\{\boxed{(\epsilon_g, \{v1 \mapsto 7\})}\right\}$$

Allowed if “does not harm other threads”

“Does not harm other threads”

$(\epsilon_g, \mathrm{emptyheap}) \rightsquigarrow \{(\epsilon_g, \{72 \mapsto 0\})\}$?

No: other thread might have e.g. $(\epsilon_g, \{72 \mapsto 123\})$

$$(\{72 \mapsto 12, 1 \mapsto 3\}, \{72 \mapsto 12\}) \rightsquigarrow \{(\{72 \mapsto 0, 1 \mapsto 3\}, \{72 \mapsto 0\})\}\ ?$$

Yes: cell update

$$a \rightsquigarrow \{b\} \iff \forall f : a.f \neq \bot \Rightarrow b.f \neq \bot$$

Increase-only counter

$$M_c = (\{\perp\} \cup \{(g,l) |\ \begin{array}{l} g \in \mathbb{N} \cup \{\epsilon_c\} \\ l \in \mathbb{N} \\ g \neq \epsilon_g \rightarrow l \leq g\} \end{array}$$

$$, \perp, \ldots, \cdot_c)$$

$$(\epsilon_c, l_1) \cdot_a (g, l_2) = (g, \min(l_1, l_2)) \textbf{ if } (g, \min(l_1, l_2)) \in |M_c| \backslash \{\perp\}$$

$$m_1 \cdot_c m_2 = \perp \textbf{ other cases}$$

Wrapping up

- Monoids
- Physical assertion
- Ghost assertion
- Invariants
- $\leadsto$

Teaser Episode 3

- Can I model I/O in Iris? (Willem)
- Logical Atomicity (Amin)