

Ocelot

Relational Logic in a Solver-Aided Language



James Bornholt
Emina Torlak

University of Washington

<http://ocelot.tools>

```
all d: Dir | lone d.~contents  
becomes  
(all ([d Dir])  
  (lone (join d (~ contents))))
```

```
(define X (join d (~ contents)))  
(println X)  
> (join d (~ contents))
```


Supports **synthesis** of relational expressions, and combination with other constraints.

Ocelot is a **DSL for relational logic, embedded in the Rosette solver-aided language**

Analysis backend is SMT (Z3)

Ocelot can synthesize and debug large memory model specifications

[Bornholt & Torlak, PLDI 2017]

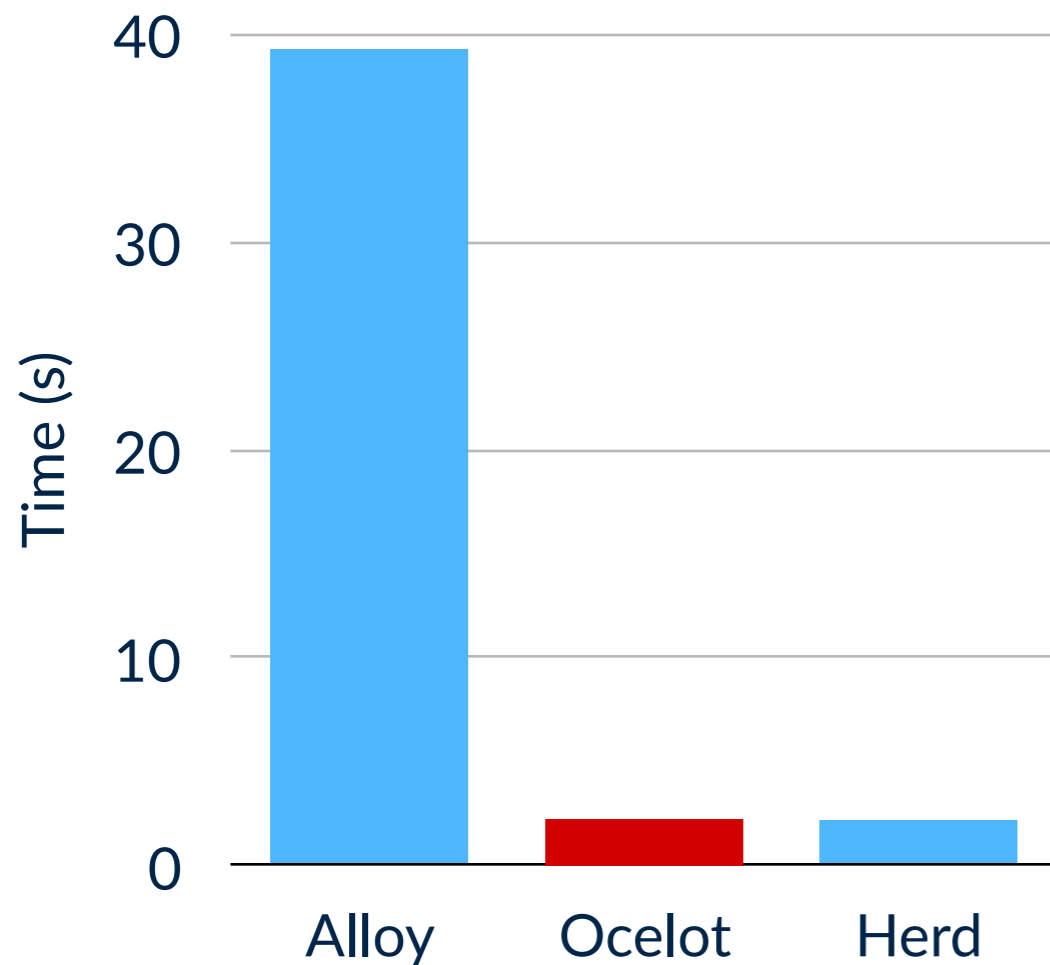
| | | <u>Synthesis</u> | <u>Ambiguity</u> |
|----------------|--|--|-----------------------------------|
| PowerPC | 768 tests [Alglave et al, CAV'10] | ✓ 12 seconds Search space: 2^{1406} | 9 new tests sync, lwsync, etc. |
| x86 | 10 tests  | ✓ 2 seconds Search space: 2^{624} | 4 new tests mfence, xchg |

Not equivalent to published model!

Not equivalent to TSO!

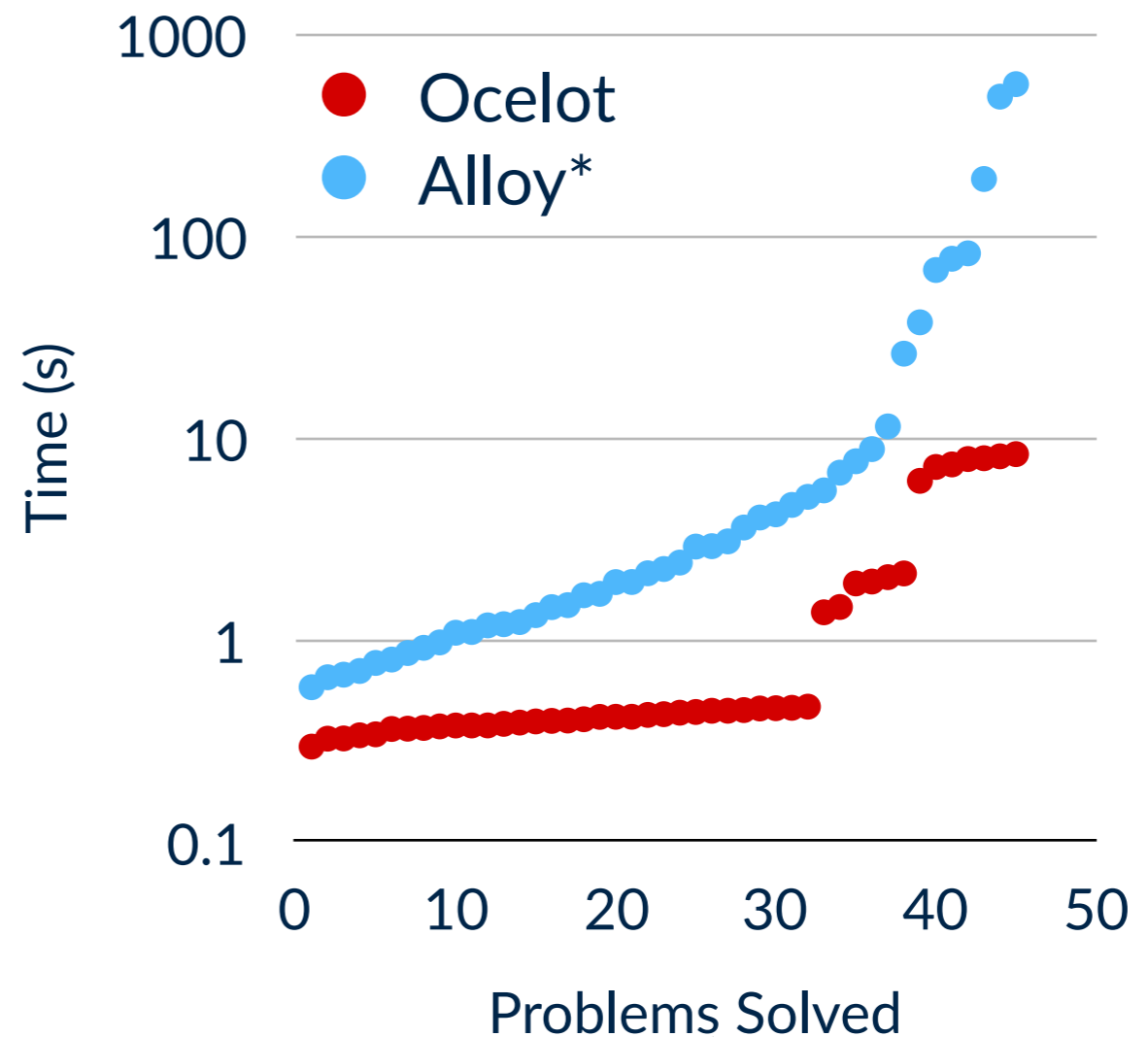
Ocelot is fast at both verification and higher-order queries

Verification



As fast as a custom memory model decision procedure

Equivalence



Average 22× faster than Alloy*

Ocelot is a DSL for
relational logic,
embedded in the
Rosette solver-aided
language



<http://ocelot.tools>