Quantization for opaque predicate location

On-going work

A. Gonzalvez[1] F. Dagnat[2] C. Fontaine[3]

[1]Univ Rennes, CNRS, IRISA, Rennes, France

[2]IMT Atlantique, Lab-STICC, Brest, France

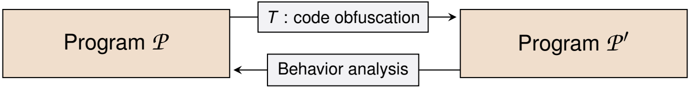
[3]Univ Paris-Saclay, CNRS, ENS Paris-Saclay, LMF, Gif-sur-Yvette, France

CAPSULE

IRISA

WG Formal Methods for Security, March 2023

Working context (1/3)



Software protection type: **opaque predicate**

Attack software protection $=$ **location problem** and deobfuscation problem

Working context (2/3)

Opaque predicates [CTL 98]

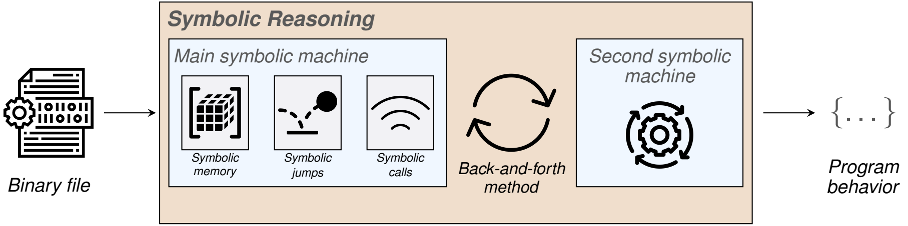
A predicate `P` is **opaque** if it has a property `r` which is known *a priori* to the obfuscator, but which is *difficult* for the deobfuscator to deduce.

Examples of opaque predicates

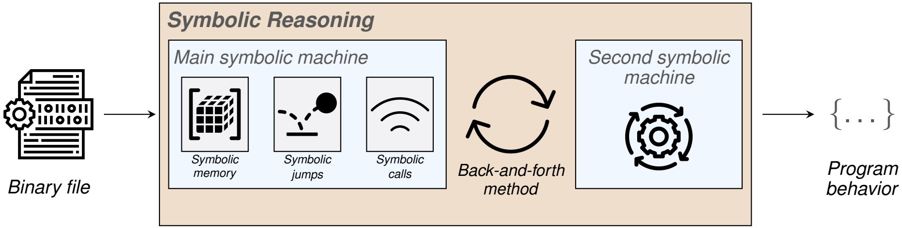
$$x * (x + 1) == 0 \; mod \; 2 \tag{1}$$

$$x + y == (x \vee y) + 2 * (x \wedge y) \tag{2}$$

Working context (3/3)

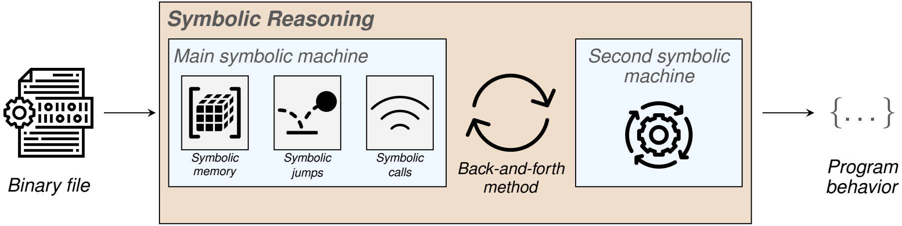


Working context (3/3)



Hard problem: Automatic location of opaque predicate during symbolic reasoning

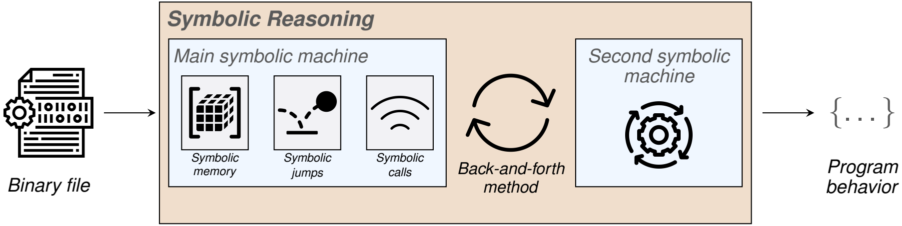
Working context (3/3)



Hard problem: Automatic location of opaque predicate during symbolic reasoning

State-of-the-art for *opaque predicate location*: defined **a priori** ☺
(heuristics, pattern-matching, algebraic methods, . . .)

Working context (3/3)



Hard problem: Automatic location of opaque predicate during symbolic reasoning

State-of-the-art for *opaque predicate location*: defined **a priori** ☺
(heuristics, pattern-matching, algebraic methods, . . .)

The *general case* is not clearly covered! ☹

Research question

How can we explicitly find the position of an opaque predicate in a binary?

SMT-solving: in one slide

Transfers information back-and-forth
between $\mathcal{T}$-solver and SAT-solver

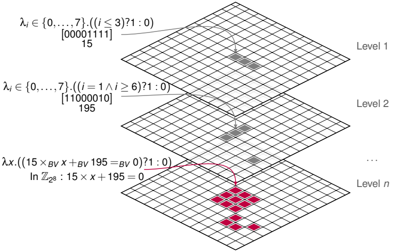
SMT-solving: in one slide

Transfers information back-and-forth between $\mathcal{T}$-solver and *SAT*-solver

In practice

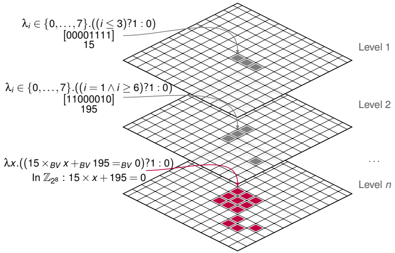
- accepts a query φ defined over a decidable theory $\mathcal{T}$
- runs an *effective interpretation*
- returns the status of φ (*sat* or *unsat*)
- returns optionally:
  - a *model* $\mathcal{M}$ (when sat)
  - a *proof* $\mathcal{P}$ (when unsat)

Bit-level precision for symbolic reasoning



$\rightarrow$ *Logic* and *fixed-size bitvectors* ($\mathcal{BV}$)
$\rightarrow$ *Stable* formulas help to catch classes of *stable theories and unstable theories*

Bit-level precision for symbolic reasoning



$\rightarrow$ *Logic* and *fixed-size bitvectors* ($\mathcal{BV}$)
$\rightarrow$ *Stable* formulas help to catch classes of *stable theories and unstable theories*

Model biinterpretable [M 13]

Two structures *s.t.*:

- each *interpretable* in the other
- the composition of the interpretations is definable

Example of model biinterpretables

Infinite finitely generated structures

The complexity of a Model: in one slide

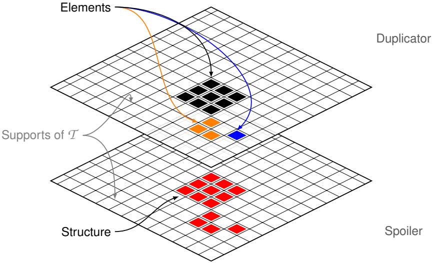
The back-and-forth games

Player 1 (or **S**poiler) challenges by providing a side and an element c.

Player 2 (or **D**uplicator) has to provide an element d on the other side that behaves similarly on the previous level.

The number of possible *moves* is defined by an initial (countable) **ordinal** α.

At each round, Player 1 picks an ordinal smaller than the previous one.



SMT-solving: example with an opaque predicate

Begin:

With bit-level precision, φ: "if EXP: $x * (x+1) == 0 \bmod 2$ is `always true` ?"

Events:

Effects:

Conclusion:

¶The finite cyclic group $\mathbb{Z}_2$ viewed as the multiplicative group of the ring $\mathbb{Z}/4\mathbb{Z}$.

†The free group F_2 with the set of symbols $\{a, b\}$ and the set of reduced words in $\{a, a^{-1}, b, b^{-1}\}$.

‡because *least support* of F_2 doesn't exist.

§[GN73][Morlay76][Millar78][G93]

SMT-solving: example with an opaque predicate

Begin:

With bit-level precision, φ: "if EXP: $x * (x+1) == 0 \bmod 2$ is `always true` ?"

Events:

- Fresh variables: biinterpretables in $\mathbb{Z}_2$¶ and F_2†
- $\mathbb{Z}_2$ interpretability: *Decidable*
- F_2 interpretability: *Computable but not Decidable* ‡
- The model construction of EXP: *an infinite finitely generated structure*§

Effects:

Conclusion:

¶The finite cyclic group $\mathbb{Z}_2$ viewed as the multiplicative group of the ring $\mathbb{Z}/4\mathbb{Z}$.

†The free group F_2 with the set of symbols $\{a, b\}$ and the set of reduced words in $\{a, a^{-1}, b, b^{-1}\}$.

‡because *least support* of F_2 doesn't exist.

§[GN73][Morlay76][Millar78][G93]

SMT-solving: example with an opaque predicate

Begin:

With bit-level precision, φ: "if EXP: $x * (x+1) == 0 \bmod 2$ is `always true` ?"

Events:

- Fresh variables: biinterpretables in $\mathbb{Z}_2$[¶] and F_2[†]
- $\mathbb{Z}_2$ interpretability: *Decidable*
- F_2 interpretability: *Computable but not Decidable* [‡]
- The model construction of EXP: *an infinite finitely generated structure*[§]

Effects:

Depending on strategy:

- **UNSAT** with an *empty proof* (*e.g.* Boolector)
- or, infinite loop for model construction (*e.g.* Z3)

Conclusion:

[¶] The finite cyclic group $\mathbb{Z}_2$ viewed as the multiplicative group of the ring $\mathbb{Z}/4\mathbb{Z}$.

[†] The free group F_2 with the set of symbols $\{a, b\}$ and the set of reduced words in $\{a, a^{-1}, b, b^{-1}\}$.

[‡] because *least support* of F_2 doesn't exist.

[§] [GN73][Morlay76][Millar78][G93]

SMT-solving: example with an opaque predicate

Begin:

With bit-level precision, φ: "if EXP: $x*(x+1) == 0 \bmod 2$ is `always true` ?"

Events:

- Fresh variables: biinterpretables in $\mathbb{Z}_2$[¶] and F_2[†]
- $\mathbb{Z}_2$ interpretability: *Decidable*
- F_2 interpretability: *Computable but not Decidable* [‡]
- The model construction of EXP: *an infinite finitely generated structure*[§]

Effects:

Depending on strategy:

- **UNSAT** with an *empty proof* (*e.g.* Boolector)
- or, infinite loop for model construction (*e.g.* Z3)

Conclusion:

EXP may be an opaque predicate.

[¶] The finite cyclic group $\mathbb{Z}_2$ viewed as the multiplicative group of the ring $\mathbb{Z}/4\mathbb{Z}$.
[†] The free group F_2 with the set of symbols $\{a, b\}$ and the set of reduced words in $\{a, a^{-1}, b, b^{-1}\}$.
[‡] because *least support* of F_2 doesn't exist.
[§] [GN73][Morlay76][Millar78][G93]

Asymptotic behavior of a model $\mathcal{M}$

General case: computation of the number of moves α is an **open problem**

	Polynomial	Spectral Gap	Intermediate	Exponential
Model behaviors	EI	$\times$	EI *or* UT	EI *or* UT
Asymptotic limits	$[1; c^k[$	$\times$	$[2^{k^{1+\varepsilon}}; 2^{p(k)}[$	$2^{p(k)}$

EI: *Effective interpretation*; UT: *Unstable theory*; $c \in [1;2^{1/5}]$; k: *swap number*; p: *polynomial of degree* ≥ 2

Table: Asymptotic recovery measurements for homogeneous structures

	Polynomial	Spectral Gap	Intermediate	Exponential
Groups examples	FG, VNG	$\times$	Grigorchuk	F_2
Asymptotic limits	$[1; R^d[$	$\times$	$[v_{min}; \exp(R)[$	$\exp(R)$

FG: *Finite groups*; VNG: *Virtual Nilpotent groups*; Grigorchuk: *Grigorchuk groups*; F_2: *Free group* F_2; $R \in \mathbb{N}$; $d \in \mathbb{N}$; v_{min}: *exp*$(R^{0.76\cdots})$

Table: Asymptotic recovery measurements for groups structures

Example with Z3 and EXP: $x * (x+1) == 0 \bmod 2$

Practical example

Time measurements of the computation each elementary equivalence of 4 consecutive propositional clauses, and compute the slope (*Effective growth rate*).

Example with Z3 and EXP: $x * (x+1) == 0 \bmod 2$

Practical example

Time measurements of the computation each elementary equivalence of 4 consecutive propositional clauses, and compute the slope (*Effective growth rate*).
Max value measured (without unit): 16.7 $\rightarrow$ *Exponential behavior*

Example with Z3 and EXP: $x * (x + 1) == 0 \bmod 2$

Practical example

Time measurements of the computation each elementary equivalence of 4 consecutive propositional clauses, and compute the slope (*Effective growth rate*).
Max value measured (without unit): 16.7 $\rightarrow$ *Exponential behavior*

$\Rightarrow$ Potentially an opaque predicate

Research question

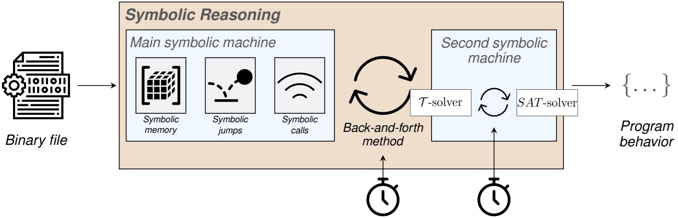
How can we explicitly find the position of an opaque predicate in a binary?

Research question

How can we explicitly find the position of an opaque predicate in a binary?

Answer:
With a dynamic complexity assessments of each model and tracing instructions / SMT-query

Opaque predicate location: overview



Preliminary results

- Some asymptotic limits are known
- First manual measures done

On-going steps

- To get more realistic examples
- Automation of dynamic measurements
- *Modification of DSE for location*

Future steps

- Finer-grained measurements ideas
- To sort and to quantify possible opaque predicate behaviors over a theory

Conclusion

A research work at the intersection of many disciplines:
Program execution, Symbolic reasoning, Formal methods, Game theory, Model theory, Algebraic structures

Cat-and-Mouse game between obfuscator people and deobfuscator people.

Thank you for your attention !