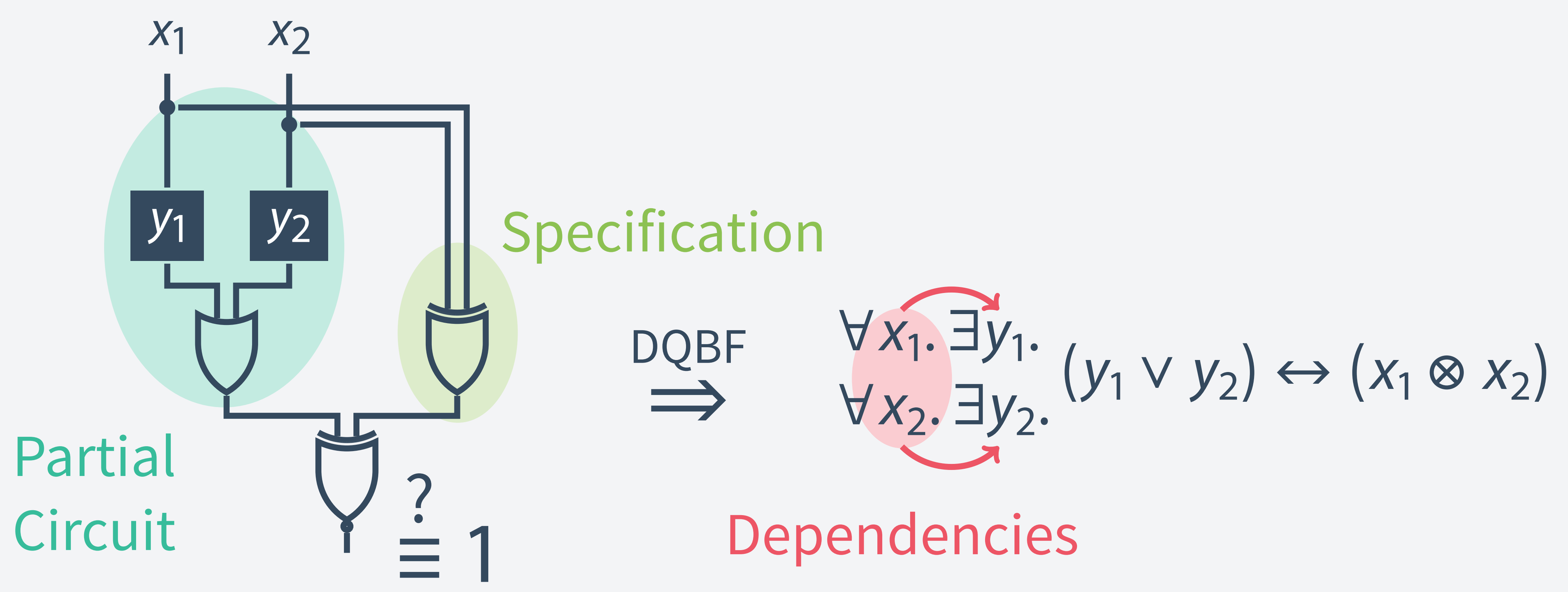
Bernd Finkbeiner
Leander Tentrup
Reactive Systems Group
Saarland University

Verifying Partial Designs with Partial Observation

Partial Equivalence Checking & DQBF
1

The Partial Equivalence Checking (PEC) problem is to decide whether there exists a combinatorial circuit for each black box in the partial design such that the completed circuit becomes equivalent to its specification.

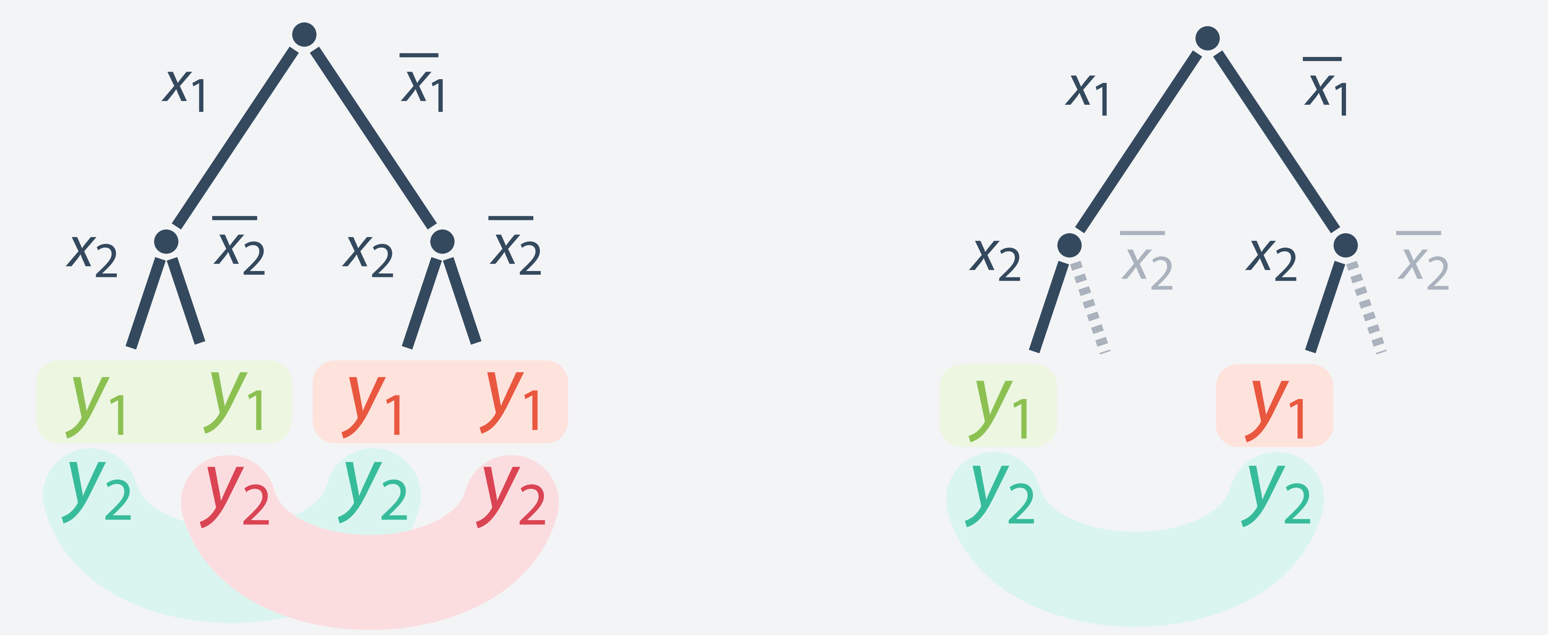


The decision problem is NEXPTIME-complete and there exists a polynomial reduction to the satisfiability problem of Dependency Quantified Boolean Formulas (DQBF) which extend QBF by Henkin quantifiers. Henkin quantifiers allow for non-linear dependencies between the existentially quantified variables.
We focus on the refutation of DQBF formulas because this corresponds to the identification of errors in the PEC problem.

Consistency: Property between Assignment-Paths
2

The assignments of y on two paths P_1 and P_2 are *consistent* if

- the assignments of y are the same ($y(P_1) = y(P_2)$) or
- the assignments of the dependencies of y on P_1 and P_2 are different.



Consistency in the full assignment tree

Consistency on the paths x_1x_2 and $\overline{x}_1x_2$

With the notion of consistency, one can characterize the existence of models, i.e., satisfying assignments of existentially quantified variables on the whole assignment-tree, as well as the existence of *partial models*, that are satisfying assignments over a subset of path assignments.
Although the existence of partial models does not imply the existence of a model, the non-existence of a partial model is a refutation witness.

DQBF Bounded Unsatisfiability [FT14]
3

A set of paths $\mathcal{P}$ can already rule out the existence of a satisfying assignment: If there is no *consistent* satisfying assignment on $\mathcal{P}$, then there is no consistent satisfying assignment for the original formula.

Definition (k-bounded unsatisfiable)

For $k \geq 1$, a DQBF formula Φ is *k-bounded unsatisfiable* if there exists a set of paths $\mathcal{P}$ with $|\mathcal{P}| \leq k$ such that there does not exist a consistent satisfying assignment over $\mathcal{P}$.

Theorem

A DQBF formula Φ is unsatisfiable if and only if it is k-bounded unsatisfiable for some $k \geq 1$.

QBF Encoding of DQBF Bounded Unsatisfiability
4

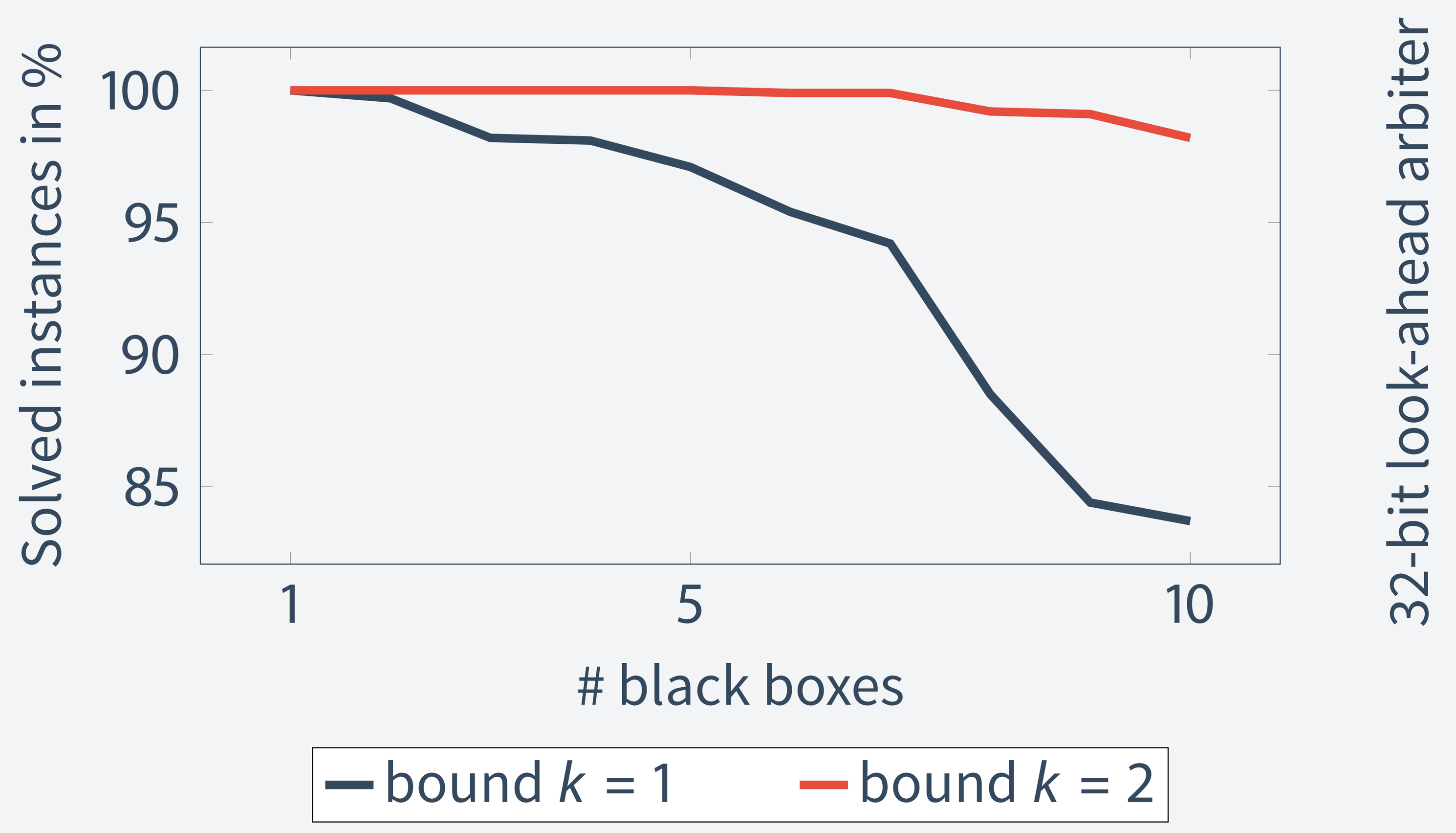
Given DQBF formula Φ and bound $k \geq 1$, **bunsat(Φ, k)** encodes a QBF query that asserts that there exist k paths such that for every *consistent* assignment of the existential variables at least one path violates the matrix.
Example for $\Phi = \forall x_1, x_2. \exists_{\{x_1\}} y_1. \exists_{\{x_2\}} y_2. \varphi$:

$$\mathbf{bunsat}(\Phi, k=2) = \exists x_1^1, x_1^2, \quad x_2^1, x_2^2. \quad \forall y_1^1, y_1^2, \quad y_2^1, y_2^2. \bigwedge_{i \in \{1,2\}} \text{consistent}(y_i^1, y_i^2) \rightarrow \neg\varphi^1 \vee \neg\varphi^2$$

$$\mathbf{bunsat}(\Phi, k=3) = \exists x_1^1, x_1^2, x_1^3, \quad x_2^1, x_2^2, x_2^3. \quad \forall y_1^1, y_1^2, y_1^3, \quad y_2^1, y_2^2, y_2^3. \bigwedge_{i \in \{1,2\}} \text{consistent}(y_i^1, y_i^2) \wedge \text{consistent}(y_i^1, y_i^3) \wedge \text{consistent}(y_i^2, y_i^3) \rightarrow \neg\varphi^1 \vee \neg\varphi^2 \vee \neg\varphi^3$$

➤ Increasing the number of paths k leads to incremental QBF formulas

Experimental Results
5



Incremental Bounded Unsatisfiability
6

- Work in progress, prototype implementation written in Python
- Look-ahead arbiter PEC benchmark family [FT14], with 20 black boxes
- Compare incremental variant (implemented using the incremental QBF solver DepQBF) to iterative monolithic variants (bunsat and bunsat+preprocessing) that use DepQBF and Bloqqer

timeout 1h	total	**bunsat+incr**	bunsat	bunsat+prepr
solved instances	100	66	61	65

- bunsat+incr solves 5 instances which no other variant could solve
- QBF preprocessing crucial for average solving time

Publications

[FT14] Bernd Finkbeiner and Leander Tentrup. "Fast DQBF Refutation". In: *SAT 2014*. Vol. 8561. LNCS. Springer, 2014, pp. 243–251.

Acknowledgements

This work was partially supported by the German Research Foundation (DFG) as part of SFB/TR 14 AVACS.

Contact

Leander Tentrup
Campus E1 3, Room 532
66123 Saarbrücken, Germany

tentrup@cs.uni-saarland.de
+49 681 302 5476

Reactive Systems

UNIVERSITÄT DES SAARLANDES