Constrained Sampling and Counting: Universal Hashing meets SAT Solving

Kuldeep S. Meel[1], Moshe Y. Vardi[1], Supratik Chakraborty[2], Daniel J. Fremont[3], Sanjit A. Seshia[3], Dror Fried[1], Alexander Ivrii[4], Sharad Malik[5]

[1]Rice University, [2]IIT Bombay, [3]University of California, Berkeley
[4]IBM Research, Haifa [5]Princeton University

Constrained Sampling and Counting

- F: CNF Formula ; R_F: Solution space of F

- Approximate Constrained Sampling

$$\forall y \in R_F,\ \frac{1}{(1+\varepsilon)|R_F|} \leq \Pr[\mathbf{UniGen}(F,\varepsilon) = y] \leq \frac{1+\varepsilon}{|R_F|}$$

- Approximate Constrained Counting

$$\Pr\left[\frac{|R_F|}{1+\varepsilon} \leq \mathrm{ApproxMC}(F,\varepsilon,\delta) \leq (1+\varepsilon)|R_F|\right] \geq 1-\delta$$

Hashing-based Techniques in Nutshell

- Key parameters:
  - Number of cells
  - Size of cell (i.e. number of solutions in a cell)
  - Size of XOR constraints

- Related Approaches
  - Strong theoretical analysis but conditioned on “right” parameters
  - Practical implementations use heuristics or “magically” choosing the right parameters

Our Contribution

- A systematic procedure to determine the right parameters

[CAV 13, CP 13, DAC 14, AAAI 14, TACAS 15, IJCAI 15, CP 15, AAAI 16]

- No gap between theoretical ideas and tool implementation
  - No user-dependent parameters
  - No heuristics without theoretical analysis

- Can handle formulas with hundreds of thousands of variables!

Peek into Runtime Performance

Generator	Normalized runtime
State-of-the-art (2012): XORSample'	50000
UniWit (2013)	5000
UniGen1 (2014)	470
UniGen2 (2015)	20
SAT Solver	1

Experiments over 200+ benchmarks