# An Approximate Skolem Function Counter

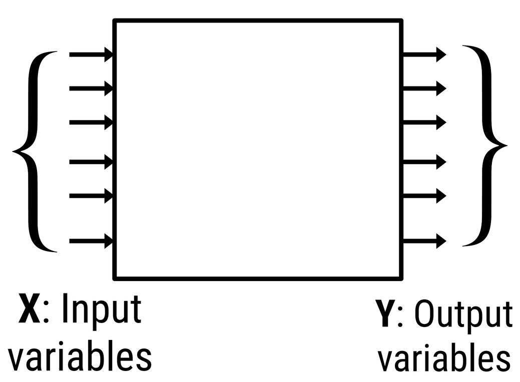

**Arijit Shaw**
Chennai Mathematical Institute
IAI, TCG-CREST, Kolkata
**Brendan Juba**
Washington University at St. Louis
**Kuldeep S. Meel**
University of Toronto


## WHAT ARE SKOLEM FUNCTIONS?

**Specification** states the relation between input and output

Given a specification F(X,Y), there can be multiple functions G satisfying the specification

Our Problem Statement: Given a specification, **count** the number of functions satisfying the specification




Also known as **Skolem** function G:

$\exists Y F(X,Y) \equiv F(X, G(X))$

## EXAMPLE

The function encode factorization of a four bit number

$X = Y_1 \times Y_2 \quad Y_1 > 1; Y_2 > 1$

| $f_1$ | $f_2$ |
|---|---|
| $12 \rightarrow 4 \times 3$ | $12 \rightarrow 6 \times 2$ |

## WHY COUNT FUNCTIONS?

Gives insight into specification, useful in :

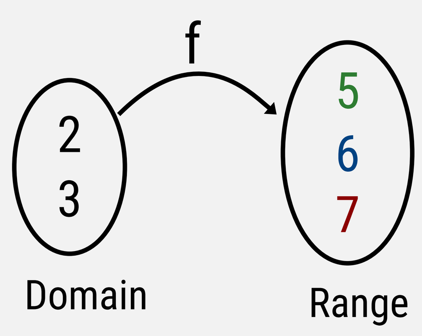
- Specification Engineering
- Understanding diversity of specification
- Evaluation of a random Skolem function

## IS IT HARD TO COUNT FUNCTIONS?

- Theoretically, #P-hard
- Even synthesizing one function is hard
- Approaches in model counting won't work
  - knowledge compilation
  - hashing-based approach

  } Harder than getting one solution

## OBSERVATION ON COUNTING FUNCTIONS




Specification says,
2 can map to only 7 and 6
3 can map to any element

$$\#\text{Functions} = \prod_{\sigma \in \text{Domain}} |\text{range}(\sigma)|$$

$$\log(\#\text{Functions}) = \sum_{\sigma \in \text{Domain}} \log(|\text{range}(\sigma)|)$$

## TECHNIQUE

### The SkolemFC Algorithm

$$\text{thres} \leftarrow 4 \ln \left(\frac{2}{\delta}\right) \frac{(1+\varepsilon)}{\varepsilon^2}$$

↓

Sample $\sigma \in S_2$

Use Almost-uniform sampler

↓

Calculate $c = |\text{range}(\sigma)|$

Use approximate model counter $F(X,Y) \wedge X = \sigma$

↓

$$\text{sum} = \text{sum} + \frac{\log(c)}{|Y|}$$

repeat while sum < thres

↓

$$\text{Est} = |S_2| \frac{\text{thres}}{\#\text{iterations}}$$

## KEY IDEA

$$\sum_{\sigma \in S_2} \log(|\text{range}(\sigma)|)$$

$$= |S_2| \sum_{\sigma \in S_1} \frac{1}{|S_2|} \log(|\text{range}(\sigma)|)$$

$$= |S_2| \mathbb{E}\left[\log(|\text{range}(\sigma)|)\right]$$

To approximate the estimate, use Monte-Carlo estimation

**We can count without even looking at one of the elements.**

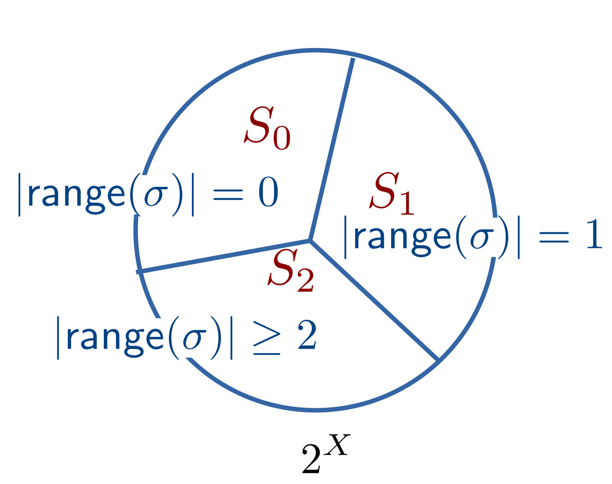
## GUARANTEES

Let s be the (log) Skolem count, then SkolemFC returns Est

$$\Pr[s(1-\varepsilon) \leq \text{Est} \leq s(1+\varepsilon)] > (1-\delta)$$

In worst case, it takes $\tilde{\mathcal{O}}(|Y|)$ many SAT oracle calls.

## PROVING GUARANTEES



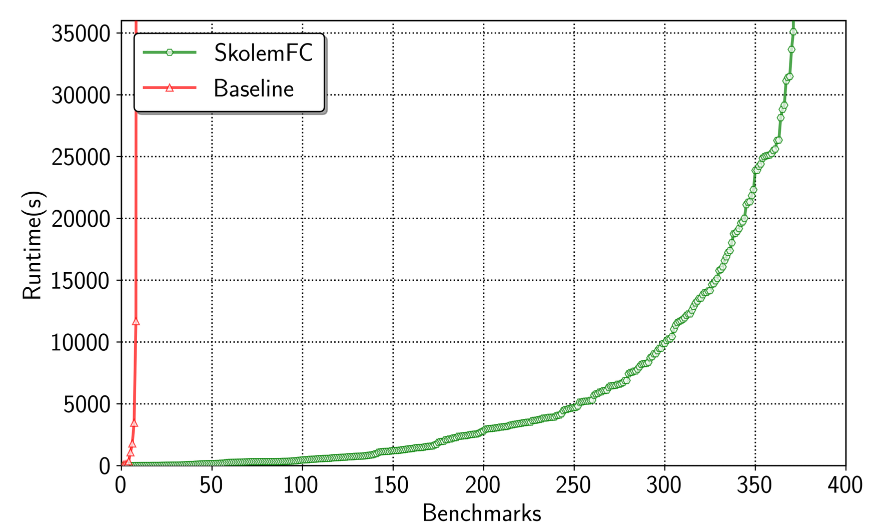

Monte Carlo estimation takes $\mathcal{O}\left(\frac{t}{\mu}\right)$ many samples and gives estimation with PAC guarantees

Limit $\mu$ from going to zero by taking samples from S2,

Using: $G(X,Y,Y') := F(X,Y) \wedge F(X,Y') \wedge (Y \neq Y')$

Each counting/sampling done in log(n) many SAT oracle call

## PERFORMANCE IN PRACTICE




609 Benchmarks, used in evaluating synthesis tools
10 hrs/ instance

**Instances solved**:

| | |
|---|---|
| SkolemFC | 375 |
| Baseline | 8 |


github.com/meelgroup/skolemfc

arijits@cmi.ac.in