Sparse Hashing for Scalable Approximate Model Counting: When Theory and Practice Finally Meet

Kuldeep S. Meel

School of Computing, National University of Singapore

Joint work with S. Akshay

Corresponding publication: LICS-20

Model Counting

- Given
  - Boolean variables $X_1, X_2, \cdots X_n$
  - Formula F over $X_1, X_2, \cdots X_n$
- $\mathrm{Sol}(F) = \{$ solutions of F $\}$

Model Counting

- Given
  - Boolean variables $X_1, X_2, \cdots X_n$
  - Formula F over $X_1, X_2, \cdots X_n$
- $\mathsf{Sol}(F) = \{$ solutions of F $\}$
- Model Counting: Determine $|\mathsf{Sol}(F)|$

Model Counting

- Given
  - Boolean variables $X_1, X_2, \cdots X_n$
  - Formula F over $X_1, X_2, \cdots X_n$
- $\mathsf{Sol}(F) = \{$ solutions of F $\}$
- Model Counting: Determine $|\mathsf{Sol}(F)|$
- Given $F := (X_1 \vee X_2)$

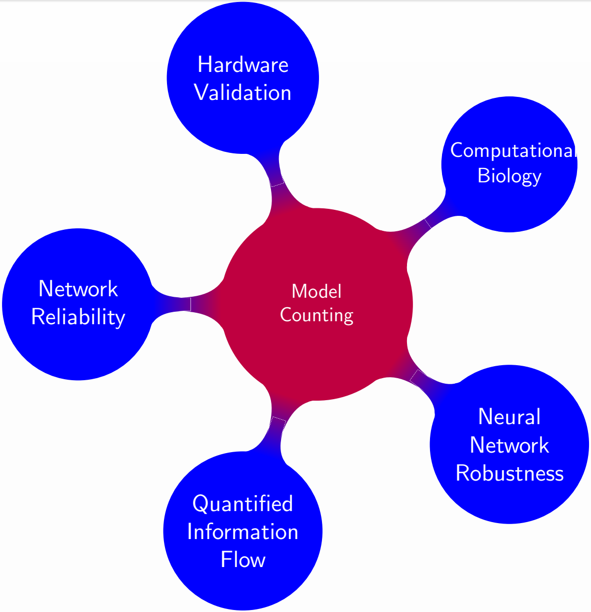
Model Counting

- Given
  - Boolean variables $X_1, X_2, \cdots X_n$
  - Formula F over $X_1, X_2, \cdots X_n$
- $\mathsf{Sol}(F) = \{$ solutions of F $\}$
- Model Counting: Determine $|\mathsf{Sol}(F)|$
- Given $F := (X_1 \vee X_2)$
- $\mathsf{Sol}(F) = \{(0,1),(1,0),(1,1)\}$

Model Counting

- Given
  - Boolean variables $X_1, X_2, \cdots X_n$
  - Formula F over $X_1, X_2, \cdots X_n$
- $\mathrm{Sol}(F) = \{$ solutions of F $\}$
- Model Counting: Determine $|\mathrm{Sol}(F)|$
- Given $F := (X_1 \vee X_2)$
- $\mathrm{Sol}(F) = \{(0,1),(1,0),(1,1)\}$
- $|\mathrm{Sol}(F)| = 3$

Applications across Computer Science



Different Shades of Approximation

- Probabilistic $(1+\varepsilon)$-Approximation

$$\Pr\left[\frac{|\mathsf{Sol}(F)|}{1+\varepsilon} \leq \mathit{ApproxCount}(F,\varepsilon,\delta) \leq |\mathsf{Sol}(F)|(1+\varepsilon)\right] \geq 1-\delta$$

Different Shades of Approximation

- Probabilistic $(1+\varepsilon)$-Approximation

$$\Pr\left[\frac{|\mathsf{Sol}(F)|}{1+\varepsilon} \le \mathit{ApproxCount}(F,\varepsilon,\delta) \le |\mathsf{Sol}(F)|(1+\varepsilon)\right] \ge 1-\delta$$

- Constant Factor Approximation: $(4,\delta)$

$$\Pr\left[\frac{|\mathsf{Sol}(F)|}{4} \le \mathit{ConstantCount}(F,\delta) \le 4\cdot|\mathsf{Sol}(F)|\right] \ge 1-\delta$$

Different Shades of Approximation

- Probabilistic $(1+\varepsilon)$-Approximation

$$\Pr\left[\frac{|\mathsf{Sol}(F)|}{1+\varepsilon} \leq ApproxCount(F,\varepsilon,\delta) \leq |\mathsf{Sol}(F)|(1+\varepsilon)\right] \geq 1-\delta$$

- Constant Factor Approximation: $(4,\delta)$

$$\Pr\left[\frac{|\mathsf{Sol}(F)|}{4} \leq ConstantCount(F,\delta) \leq 4\cdot|\mathsf{Sol}(F)|\right] \geq 1-\delta$$

- From 4 to 2-factor
  Let $G = F_1 \wedge F_2$ (i.e., two identical copies of F)

$$\frac{|\mathsf{Sol}(G)|}{4} \leq C \leq 4\cdot|\mathsf{Sol}(G)| \implies \frac{|\mathsf{Sol}(F)|}{2} \leq \sqrt{C} \leq 2\cdot|\mathsf{Sol}(F)|$$

Different Shades of Approximation

- Probabilistic $(1+\varepsilon)$-Approximation

$$\Pr\left[\frac{|\mathsf{Sol}(F)|}{1+\varepsilon} \leq ApproxCount(F,\varepsilon,\delta) \leq |\mathsf{Sol}(F)|(1+\varepsilon)\right] \geq 1-\delta$$

- Constant Factor Approximation: $(4,\delta)$

$$\Pr\left[\frac{|\mathsf{Sol}(F)|}{4} \leq ConstantCount(F,\delta) \leq 4\cdot|\mathsf{Sol}(F)|\right] \geq 1-\delta$$

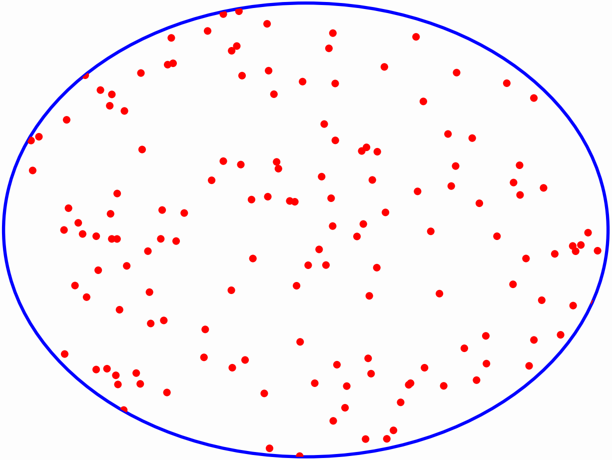
- From 4 to 2-factor
  Let $G = F_1 \wedge F_2$ (i.e., two identical copies of F)

$$\frac{|\mathsf{Sol}(G)|}{4} \leq C \leq 4\cdot|\mathsf{Sol}(G)| \implies \frac{|\mathsf{Sol}(F)|}{2} \leq \sqrt{C} \leq 2\cdot|\mathsf{Sol}(F)|$$

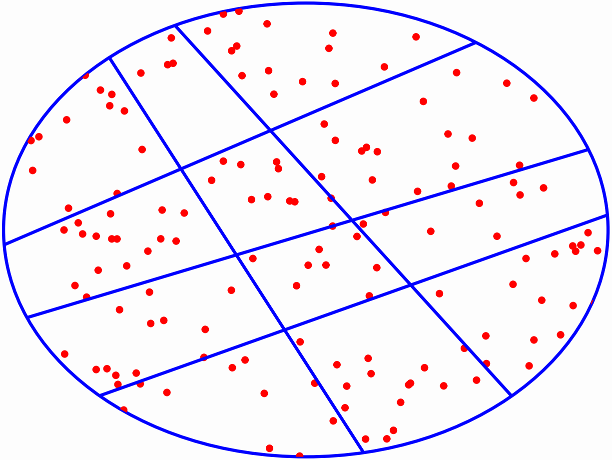
- From 4 to $(1+\varepsilon)$-factor
  Construct $G = F_1 \wedge F_2 \ldots F_{\frac{1}{\varepsilon}}$ And then we can take $\frac{1}{\varepsilon}$-root

The Rise of Hashing-based Approach: Promise of Scalability and Guarantees
(S83,GSS06,GHSS07,CMV13b,EGSS13b,CMV14,CDR15,CMV16,ZCSE16,AD16
KM18,ATD18,SM19,ABM20,SGM20)

As Simple as Counting Dots

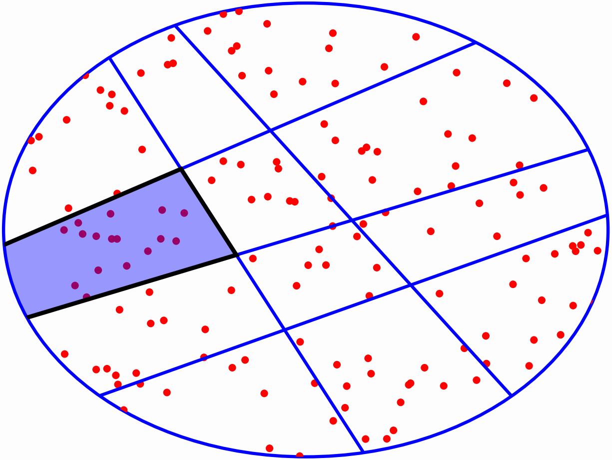


As Simple as Counting Dots



As Simple as Counting Dots

Pick a random cell



Estimate = Number of solutions in a cell × Number of cells

Challenges

Challenge 1 What is exactly a *small cell* ?

Challenges

Challenge 1 What is exactly a *small cell* ?

Challenge 2 How to partition into roughly equal small cells of solutions without knowing the distribution of solutions?

Challenge 3 How many cells?

Challenges

Challenge 1 What is exactly a *small cell* ?

- A cell is small cell if it has $\approx$ thresh solutions.
- Two choices for thresh.
  - thresh $= \mathrm{constant} \rightarrow$ 4-factor approximation
  - thresh $= \mathcal{O}(\frac{1}{\varepsilon^2})$ gives $(1+\varepsilon)$-approximation directly

Challenges

Challenge 1 What is exactly a *small cell* ?

- A cell is small cell if it has $\approx$ thresh solutions.
- Two choices for thresh.
  - thresh $= \mathrm{constant} \rightarrow$ 4-factor approximation
  - thresh $= \mathcal{O}(\frac{1}{\varepsilon^2})$ gives $(1+\varepsilon)$-approximation directly
- Z_m be the number of solutions in a randomly chosen cell ; we are interested in cases $\mathsf{E}[Z_m] \geq 1$

Challenges

Challenge 1 What is exactly a *small cell* ?

- A cell is small cell if it has $\approx$ thresh solutions.
- Two choices for thresh.
  - thresh $= \mathrm{constant} \rightarrow$ 4-factor approximation
  - thresh $= \mathcal{O}(\frac{1}{\varepsilon^2})$ gives $(1+\varepsilon)$-approximation directly
- Z_m be the number of solutions in a randomly chosen cell ; we are interested in cases $\mathsf{E}[Z_m] \geq 1$
- For thresh $= \mathcal{O}(\frac{1}{\varepsilon^2})$, we need
  dispersion index: $\frac{\sigma^2[Z_m]}{(\mathsf{E}[Z_m])} \leq \mathrm{some\ constant}$
- For thresh $= \mathrm{constant}$, sufficient to have
  coefficient of variation: $\frac{\sigma^2[Z_m]}{(\mathsf{E}[Z_m])^2} \leq \mathrm{some\ constant}$

Challenges

Challenge 1 What is exactly a *small cell* ?

- A cell is small cell if it has $\approx$ thresh solutions.
- Two choices for thresh.
  - thresh $=$ constant $\rightarrow$ 4-factor approximation
  - thresh $= \mathcal{O}(\frac{1}{\varepsilon^2})$ gives $(1+\varepsilon)$-approximation directly
- Z_m be the number of solutions in a randomly chosen cell ; we are interested in cases $\mathsf{E}[Z_m] \geq 1$
- For thresh $= \mathcal{O}(\frac{1}{\varepsilon^2})$, we need

  dispersion index: $\frac{\sigma^2[Z_m]}{(\mathsf{E}[Z_m])} \leq$ some constant
- For thresh $=$ constant, sufficient to have

  coefficient of variation: $\frac{\sigma^2[Z_m]}{(\mathsf{E}[Z_m])^2} \leq$ some constant

Techniques based on thresh $= \mathcal{O}(\frac{1}{\varepsilon^2})$ such as ApproxMC scale significantly better than those based on thresh $=$ constant.

Challenges

Challenge 1 What is exactly a *small cell* ?

Challenge 2 How to partition into roughly equal small cells of solutions without knowing the distribution of solutions?

Challenges

Challenge 1 What is exactly a *small cell* ?

Challenge 2 How to partition into roughly equal small cells of solutions without knowing the distribution of solutions?

- Designing function h : assignments $\rightarrow$ cells (hashing)
- Solutions in a cell α: $\mathsf{Sol}(F) \cap \{y \mid h(y) = \alpha\}$

Challenges

Challenge 1 What is exactly a *small cell* ?

Challenge 2 How to partition into roughly equal small cells of solutions without knowing the distribution of solutions?

- Designing function h : assignments $\rightarrow$ cells (hashing)
- Solutions in a cell α: $\mathrm{Sol}(F) \cap \{y \mid h(y) = \alpha\}$
- Choose h randomly from a specially constructed large family H of hash functions

Carter and Wegman 1977

Pairwise Independent Hashing

- Variables: $X_1, X_2, \cdots X_n$
- To construct $h : \{0,1\}^n \rightarrow \{0,1\}^m$, choose m random XORs
- Pick every X_i with prob. $\frac{1}{2}$ and XOR them
  - $X_1 \oplus X_3 \oplus X_6 \cdots \oplus X_{n-2}$
  - Expected size of each XOR: $\frac{n}{2}$

Pairwise Independent Hashing

- Variables: $X_1, X_2, \cdots X_n$
- To construct $h : \{0,1\}^n \rightarrow \{0,1\}^m$, choose m random XORs
- Pick every X_i with prob. $\frac{1}{2}$ and XOR them
  - $X_1 \oplus X_3 \oplus X_6 \cdots \oplus X_{n-2}$
  - Expected size of each XOR: $\frac{n}{2}$
- To choose $\alpha \in \{0,1\}^m$, set every XOR equation to 0 or 1 randomly

$$X_1 \oplus X_3 \oplus X_6 \cdots \oplus X_{n-2} = 0 \qquad (Q_1)$$
$$X_2 \oplus X_5 \oplus X_6 \cdots \oplus X_{n-1} = 1 \qquad (Q_2)$$
$$\cdots \qquad (\cdots)$$
$$X_1 \oplus X_2 \oplus X_5 \cdots \oplus X_{n-2} = 1 \qquad (Q_m)$$

- Solutions in a cell: $F \wedge Q_1 \cdots \wedge Q_m$

The Performance Bottleneck: SAT Calls

- Variables: $X_1, X_2, \cdots X_n$
- Set of XORs

$$X_1 \oplus X_3 \oplus X_6 \cdots \oplus X_{n-2} = 0 \quad (Q_1)$$
$$X_2 \oplus X_5 \oplus X_6 \cdots \oplus X_{n-1} = 1 \quad (Q_2)$$
$$\cdots \quad (\cdots)$$
$$X_1 \oplus X_2 \oplus X_5 \cdots \oplus X_{n-2} = 1 \quad (Q_m)$$

- Solutions in a cell: $F \wedge Q_1 \cdots \wedge Q_m$

The Performance Bottleneck: SAT Calls

- Variables: $X_1, X_2, \cdots X_n$
- Set of XORs

$$
\begin{aligned}
X_1 \oplus X_3 \oplus X_6 \cdots \oplus X_{n-2} &= 0 && (Q_1)\\
X_2 \oplus X_5 \oplus X_6 \cdots \oplus X_{n-1} &= 1 && (Q_2)\\
\cdots & && (\cdots)\\
X_1 \oplus X_2 \oplus X_5 \cdots \oplus X_{n-2} &= 1 && (Q_m)
\end{aligned}
$$

- Solutions in a cell: $F \wedge Q_1 \cdots \wedge Q_m$
- Performance of state of the art SAT solvers degrade with increase in the size of XORs (SAT Solvers != SAT oracles)

The Hope of Short XORs

- View the set of XORs as Matrices: $AX = b$ where $\cdot = \wedge$ and $+ = \oplus$
  - A is 0-1 matrix of size $m \times n$
  - b is 0-1 matrix of size $m \times 1$
- If we pick every variable X_i with probability p .
  - Expected Size of each XOR: np
- $\Pr[\sigma_1 \text{is in Cell}] = \Pr[A\sigma_1 = b] = \frac{1}{2^m}$
  - $\mathsf{E}[Z_m] = \sum_{\sigma \in \mathsf{Sol}(F)} \Pr[\sigma_1 \text{is in Cell}] = \frac{|\mathsf{Sol}(F)|}{2^m}$
- Now,

$$\begin{aligned}
\Pr[\sigma_1 \text{ and } \sigma_2 \text{ are in Cell}] &= \Pr[A\sigma_1 = b = A\sigma_2] \\
&= \Pr[A\sigma_1 = b] \Pr[A(\sigma_2 - \sigma_1) = 0] \\
&= \frac{1}{2^m} \left(\frac{1}{2} + \frac{(1-2p)^w}{2} \right)^m
\end{aligned}$$

- $\sigma^2[Z_m] \leq \mathsf{E}[Z_m] + \sum_{\sigma_1 \in \mathsf{Sol}(F)} \sum_{\substack{\sigma_2 \in \mathsf{Sol}(F) \\ w = d(\sigma_1, \sigma_2)}} r(w, m)$
  - where, $r(w, m) = \frac{1}{2^m} \left(\left(\frac{1}{2} + \frac{(1-2p)^w}{2} \right)^m - \frac{1}{2^m} \right)$
- For $p = \frac{1}{2}$, we have $\frac{\sigma^2[Z_m]}{\mathsf{E}[Z_m]} \leq 1$

The First Decade

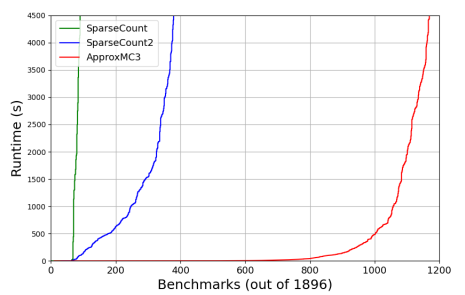
- The first decade (GSS07,EGSS14,ZCSE16,AD17,ATD18)
  - $\sum_{\sigma_1 \in \mathsf{Sol}(F)} \sum_{\substack{\sigma_2 \in \mathsf{Sol}(F) \\ w = d(\sigma_1, \sigma_2)}} r(w, m) \leq \sum_{\sigma_1 \in \mathsf{Sol}(F)} \sum_{w=0}^{n} \binom{n}{w} r(w, m)$
  - $\binom{n}{w}$ grows very fast with n, so can't upper bound $\frac{\sigma^2[Z_m]}{\mathsf{E}[Z_m]}$ by a constant.

The First Decade

- The first decade (GSS07,EGSS14,ZCSE16,AD17,ATD18)
  - $\sum_{\sigma_1 \in \mathsf{Sol}(F)} \sum_{\substack{\sigma_2 \in \mathsf{Sol}(F) \\ w = d(\sigma_1, \sigma_2)}} r(w, m) \leq \sum_{\sigma_1 \in \mathsf{Sol}(F)} \sum_{w=0}^{n} \binom{n}{w} r(w, m)$
  - $\binom{n}{w}$ grows very fast with n, so can't upper bound $\frac{\sigma^2[Z_m]}{\mathsf{E}[Z_m]}$ by a constant.
  - But, $\frac{\sigma^2[Z_m]}{(\mathsf{E}[Z_m])^2} \leq 1$ for $p = \mathcal{O}(\frac{\log m}{m})$ (ZCSE16,AD17,ATD18)

The First Decade

- The first decade (GSS07,EGSS14,ZCSE16,AD17,ATD18)
  - $\sum_{\sigma_1 \in \mathrm{Sol}(F)} \sum_{\substack{\sigma_2 \in \mathrm{Sol}(F) \\ w = d(\sigma_1, \sigma_2)}} r(w, m) \leq \sum_{\sigma_1 \in \mathrm{Sol}(F)} \sum_{w=0}^{n} \binom{n}{w} r(w, m)$
  - $\binom{n}{w}$ grows very fast with n, so can't upper bound $\frac{\sigma^2[Z_m]}{\mathsf{E}[Z_m]}$ by a constant.
  - But, $\frac{\sigma^2[Z_m]}{(\mathsf{E}[Z_m])^2} \leq 1$ for $p = \mathcal{O}(\frac{\log m}{m})$ (ZCSE16,AD17,ATD18)
  - The weak bounds lead to significant slowdown: typically 100× to 1000× factor of slowdown! (ADM20)



The Power of Isoperimetric Inequalities

- $$\sum_{\sigma_1 \in \mathrm{Sol}(F)} \sum_{\substack{\sigma_2 \in \mathrm{Sol}(F) \\ w = d(\sigma_1, \sigma_2)}} r(w, m) = \sum_{w=1}^{n} C_F(w) r(w, m)$$
- $C_F(w) = |\{\sigma_1, \sigma_2 \in \mathrm{Sol}(F) \mid d(\sigma_1, \sigma_2) = w\}|$
- Question What is the maximum value of $C_F(1)$?

The Power of Isoperimetric Inequalities

- $\sum_{\sigma_1 \in \mathrm{Sol}(F)} \sum_{\substack{\sigma_2 \in \mathrm{Sol}(F) \\ w = d(\sigma_1, \sigma_2)}} r(w,m) = \sum_{w=1}^{n} C_F(w) r(w,m)$
- $C_F(w) = |\{\sigma_1, \sigma_2 \in \mathrm{Sol}(F) \mid d(\sigma_1, \sigma_2) = w\}|$
- Question What is the maximum value of $C_F(1)$?
- Well, $C_F(1) \le |\mathrm{Sol}(F)| \binom{n}{1}$
- Suppose $n = 3$ and $|\mathrm{Sol}(F)| = 3$
- Possibilities: $\{(0,0,0), (1,0,0), (0,1,0), (0,0,1)\}$

The Power of Isoperimetric Inequalities

- $\sum_{\sigma_1 \in \text{Sol}(F)} \sum_{\substack{\sigma_2 \in \text{Sol}(F) \\ w = d(\sigma_1, \sigma_2)}} r(w, m) = \sum_{w=1}^{n} C_F(w) r(w, m)$
- $C_F(w) = |\{\sigma_1, \sigma_2 \in \text{Sol}(F) \mid d(\sigma_1, \sigma_2) = w\}|$
- Question What is the maximum value of $C_F(1)$?
- Well, $C_F(1) \leq |\text{Sol}(F)| \binom{n}{1}$
- Suppose $n = 3$ and $|\text{Sol}(F)| = 3$
- Possibilities: $\{(0,0,0), (1,0,0), (0,1,0), (0,0,1)\}$

Theorem (Harper's Theorem (1962))

$C_F(1) \leq |\text{Sol}(F)| \binom{\ell}{1}$ *where* $\ell = \log |\text{Sol}(F)|$

The Power of Isoperimetric Inequalities

Lemma (Rashtchian and Raynaud 2019)

$$\sum_{w=1}^{n} C_F(w) \leq \sum_{w=1}^{n} \binom{8e\sqrt{n \cdot \ell}}{w} \text{ where } \ell = \log|\mathrm{Sol}(F)|$$

The Power of Isoperimetric Inequalities

Lemma (Rashtchian and Raynaud 2019)

$$\sum_{w=1}^{n} C_F(w) \leq \sum_{w=1}^{n} \binom{8e\sqrt{n\cdot\ell}}{w} \textit{ where } \ell = \log|\mathrm{Sol}(F)|$$

What about $\sum_{w=1}^{n} C_F(w) r(w,m)$?

The Power of Isoperimetric Inequalities

Lemma (Rashtchian and Raynaud 2019)

$$\sum_{w=1}^{n} C_F(w) \leq \sum_{w=1}^{n} \binom{8e\sqrt{n\cdot\ell}}{w} \text{ where } \ell = \log|\mathrm{Sol}(F)|$$

What about $\sum_{w=1}^{n} C_F(w) r(w,m)$?

Lemma

$$\sum_{w=1}^{n} C_F(w) r(w,m) \leq \sum_{w=1}^{n} \binom{8e\sqrt{n\cdot\ell}}{w} r(w,m) \text{ where } \ell = \log|\mathrm{Sol}(F)|$$

- Improvement from $\binom{n}{w}$ to $\binom{8e\sqrt{n\cdot\ell}}{w}$
- $\frac{\binom{n}{w}}{\binom{8e\sqrt{n\cdot\ell}}{w}} \approx \left(\frac{n}{\ell}\right)^{\frac{w}{2}}$

From Linear to Logarithmic Size XORs

Theorem (Informal)

For all q, k, $|\mathsf{Sol}(F)| \leq k \cdot 2^m$, $p = \mathcal{O}(\frac{\log m}{m})$ *we have*

$$\frac{\sigma^2[Z_m]}{\mathsf{E}[Z_m]} \leq q(a\ constant)$$

Recall, average size of XORs: $n \cdot p$
Improvement of p from $\frac{m/2}{m}$ *to* $\frac{\log m}{m}$

From Linear to Logarithmic Size XORs

Theorem (Informal)

For all q, k, $|\mathsf{Sol}(F)| \leq k \cdot 2^m$, $p = \mathcal{O}(\frac{\log m}{m})$ *we have*

$$\frac{\sigma^2[Z_m]}{\mathsf{E}[Z_m]} \leq q(\text{a constant})$$

Recall, average size of XORs: $n \cdot p$
Improvement of p *from* $\frac{m/2}{m}$ *to* $\frac{\log m}{m}$

Challenge: No meaningful bounds on $|\mathsf{Sol}(F)|$

How many cells?

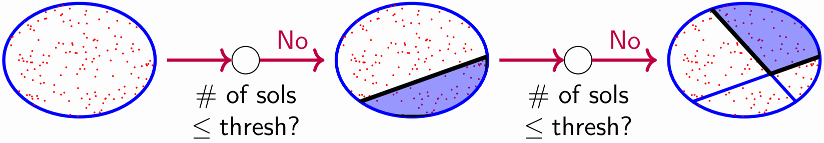
- We want to partition into 2^{m^*} cells such that $2^{m^*} = \frac{|\mathrm{Sol}(F)|}{\mathrm{thresh}}$

How many cells?

- We want to partition into 2^{m^*} cells such that $2^{m^*} = \frac{|\mathsf{Sol}(F)|}{\mathsf{thresh}}$
  - Check for every $m = 0, 1, \cdots n$ if the number of solutions $\leq$ thresh

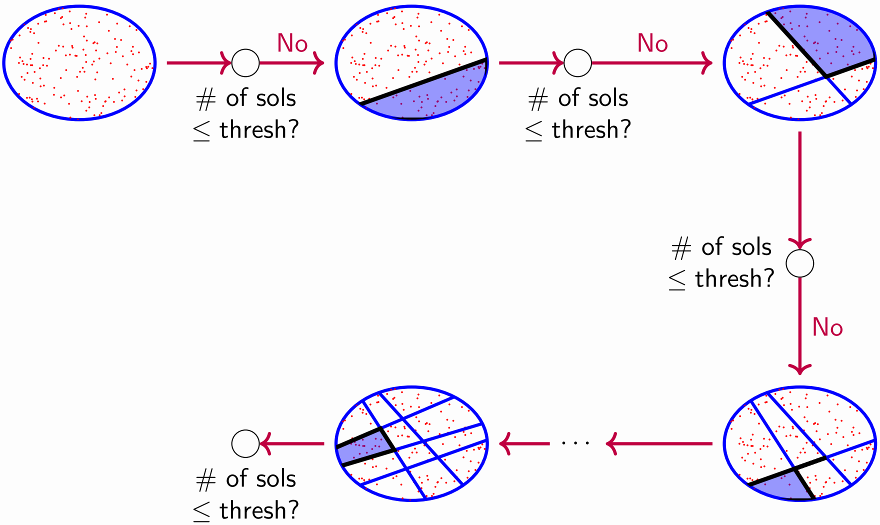
How many cells?

- We want to partition into 2^{m^*} cells such that $2^{m^*} = \frac{|\mathsf{Sol}(F)|}{\mathsf{thresh}}$
  - Check for every $m = 0, 1, \cdots n$ if the number of solutions $\leq$ thresh



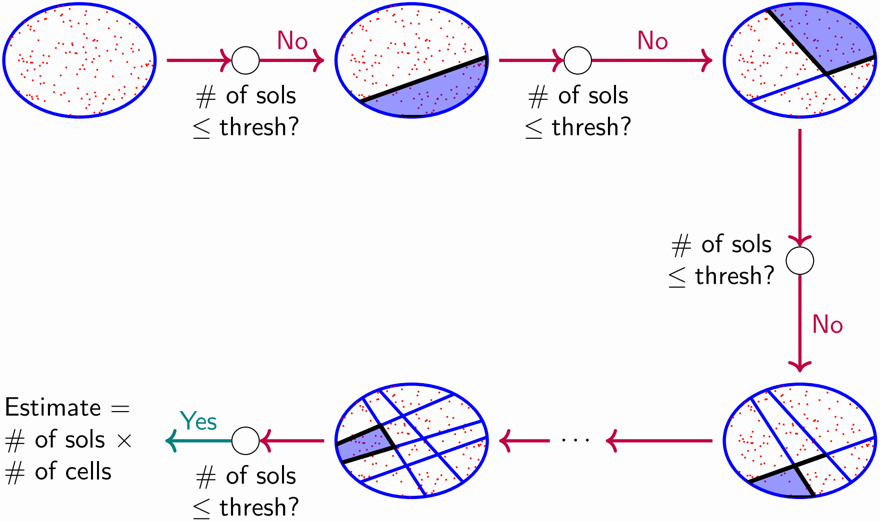
How many cells?

- We want to partition into 2^{m^*} cells such that $2^{m^*} = \frac{|\mathsf{Sol}(F)|}{\mathsf{thresh}}$
  - Check for every $m = 0, 1, \cdots n$ if the number of solutions $\leq$ thresh



How many cells?

- We want to partition into 2^{m^*} cells such that $2^{m^*} = \frac{|\mathsf{Sol}(F)|}{\mathsf{thresh}}$
  - Check for every $m = 0, 1, \cdots n$ if the number of solutions $\leq$ thresh



The Secrets of Hashing-based Techniques

Challenge How do we obtain meaningful bounds on $|\mathsf{Sol}(F)|$?

Solution : We do not need to!

Key Insight : When adding m-th XOR, theoretical analysis only requires $\frac{\sigma^2[Z_m]}{\mathsf{E}[Z_m]} \leq q$ whenever $|\mathsf{Sol}(F)| \leq \mathsf{thresh} \cdot 2^m$

The Secrets of Hashing-based Techniques

Challenge How do we obtain meaningful bounds on $|\mathsf{Sol}(F)|$?

Solution : We do not need to!

Key Insight : When adding m-th XOR, theoretical analysis only requires $\frac{\sigma^2[Z_m]}{\mathsf{E}[Z_m]} \leq q$ whenever $|\mathsf{Sol}(F)| \leq \mathsf{thresh} \cdot 2^m$

- Suppose m-th XOR is added with p_m and $p_1 \geq p_2 \cdots \geq p_m$
- $\sigma^2[Z_m] \leq \mathsf{E}[Z_m] + \sum_{\sigma_1 \in \mathsf{Sol}(F)} \sum_{\substack{\sigma_2 \in \mathsf{Sol}(F) \\ w = d(\sigma_1, \sigma_2)}} r(w, m)$

$$
\begin{aligned}
r(w, m) &= \frac{1}{2^m}\left(\prod_{i=1}^{m}\left(\frac{1}{2} + \frac{(1-2p_i)^w}{2}\right) - \frac{1}{2^m}\right) \\
&\leq \frac{1}{2^m}\left(\prod_{i=1}^{m}\left(\frac{1}{2} + \frac{(1-2p_m)^w}{2}\right)^m - \frac{1}{2^m}\right)
\end{aligned}
$$

The Secrets of Hashing-based Techniques

Challenge How do we obtain meaningful bounds on $|\mathsf{Sol}(F)|$?

Solution : We do not need to!

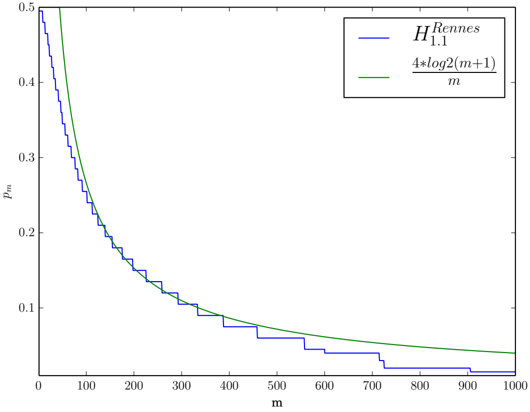
Key Insight : When adding m-th XOR, theoretical analysis only requires $\frac{\sigma^2[Z_m]}{\mathsf{E}[Z_m]} \leq q$ whenever $|\mathsf{Sol}(F)| \leq \mathsf{thresh} \cdot 2^m$

- Suppose m-th XOR is added with p_m and $p_1 \geq p_2 \cdots \geq p_m$
- $\sigma^2[Z_m] \leq \mathsf{E}[Z_m] + \sum_{\sigma_1 \in \mathsf{Sol}(F)} \sum_{\substack{\sigma_2 \in \mathsf{Sol}(F) \\ w = d(\sigma_1, \sigma_2)}} r(w, m)$

$$r(w,m) = \frac{1}{2^m}\left(\prod_{i=1}^{m}\left(\frac{1}{2} + \frac{(1-2p_i)^w}{2}\right) - \frac{1}{2^m}\right)$$
$$\leq \frac{1}{2^m}\left(\prod_{i=1}^{m}\left(\frac{1}{2} + \frac{(1-2p_m)^w}{2}\right)^m - \frac{1}{2^m}\right)$$

- Add m-th XOR with $p_m = \mathcal{O}(\frac{\log m}{m})$

Sparse Hash Functions



$H_{1.1}^{Rennes}$: Sparse hash functions that guarantee $q = 1.1$

Experimental Evaluation

Benchmark	Vars	$\log_2$(Count)	ApproxMC4	ApproxMC5	Speedup
03B-4	27966	28.55	983.72	1548.96	0.64
squaring23	710	23.11	0.66	1.21	0.55
case144	765	82.07	102.65	202.06	0.51
modexp8-4-6	83953	32.13	788.23	920.34	0.86
min-28s	3933	459.23	48.63	35.83	1.36
s9234a_7_4	6313	246.0	4.77	2.45	1.95
min-8	1545	284.78	8.86	4.59	1.93
s13207a_7_4	9386	699.0	34.94	17.05	2.05
min-16	3065	539.88	33.67	16.61	2.03
90-15-4-q	1065	839.25	273.1	135.75	2.01
s35932_15_7	17918	1761.0	–	72.32	–
s38417_3_2	25528	1663.02	–	71.04	–
75-10-8-q	460	360.13	–	4850.28	–
90-15-8-q	1065	840.0	–	3717.05	–

Remember; thresh $= \mathcal{O}(\frac{\sigma^2[Z_m]}{\mathsf{E}[Z_m]} \cdot \frac{1}{\varepsilon^2})$

$\frac{\sigma^2[Z_m]}{\mathsf{E}[Z_m]} \leq 1$ for 2-wise independent; $\frac{\sigma^2[Z_m]}{\mathsf{E}[Z_m]} \leq q = 1.1$ for $H_{1.1}^{Rennes}$.

The first sparse XOR-based scheme to achieve speedup without loss of theoretical gurantees

Conclusion

- Hashing-based techniques employ random XORs, and promise theoretical guarantees and scalability
- The runtime of SAT solvers depend on the size of XORs
- Meaningful bounds on $\frac{\sigma^2[Z_m]}{\mathsf{E}[Z_m]}$ via Isoperimetric inequalities.
- The first sparse XOR scheme to attain speedup improvement without loss of theoretical guarantees
- Future Directions:
  - Theoretical Lower bounds on the sparsity of XORs
  - Algorithmic Achieving speedup without slow down for any instance
  - System Design of Sparse XOR-based XOR solving modules
- Open-source Tool: `https://github.com/meelgroup/approxmc`