Quantum Computation using Weighted Model Counting

Dirck van den Ende

Leiden University

Supervisors:
Dr. Alfons Laarman
Dr. Joon Hyung Lee
Dr. Henning Basold

July 24^{th}, 2025

Overview

1. Why quantum computation?
2. Why weighted model counting?
3. What is weighted model counting?
4. Quantum computation with WMC
5. Application: Potts model
6. Application: Transverse-field Ising model

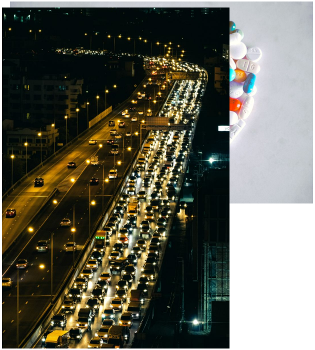
Why Quantum Computation?

Why Quantum Computation?

- Drug development

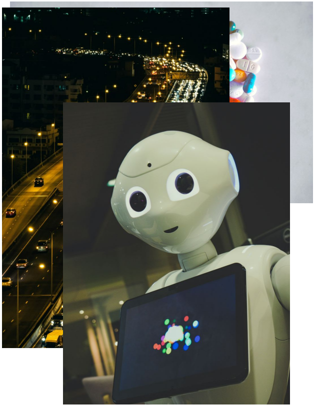
Why Quantum Computation?

- Drug development
- Traffic optimization



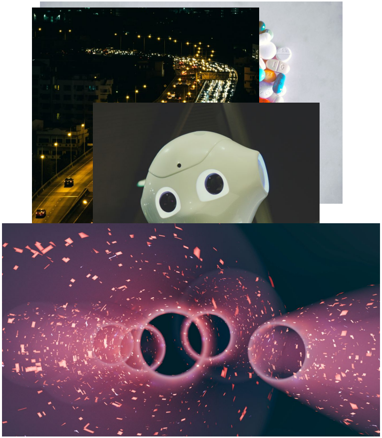
Why Quantum Computation?

- Drug development
- Traffic optimization
- Artificial intelligence



Why Quantum Computation?

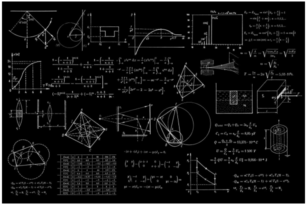
- Drug development
- Traffic optimization
- Artificial intelligence
- Quantum simulation
  - Transverse-field Ising model



Why Weighted Model Counting?

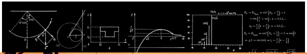
Why Weighted Model Counting?

- Statistical physics



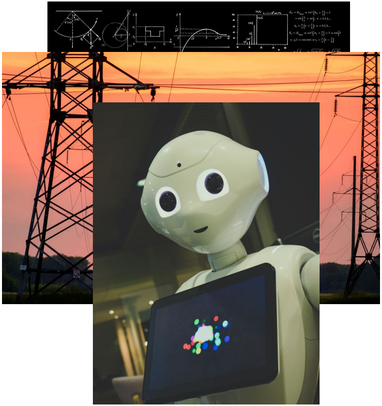
Why Weighted Model Counting?

- Statistical physics
- Critical infrastructure reliability



Why Weighted Model Counting?

- Statistical physics
- Critical infrastructure reliability
- Artificial intelligence



Why Weighted Model Counting?

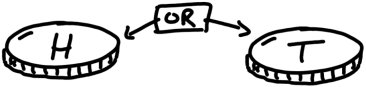
- Statistical physics
- Critical infrastructure reliability
- Artificial intelligence
- Variety of probabilistic problems

What is Weighted Model Counting?

What is Weighted Model Counting?

- Calculating the sum of weights over a set of solutions to a problem
  - Every solution has a weight
- Problems consist of two parts:
  - Boolean formula
  - Weight function

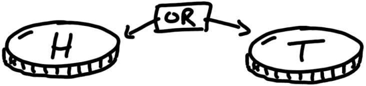
What is Weighted Model Counting?

c_1  c_2

Example

- Probability of the first coin being heads or the second being tails?

What is Weighted Model Counting?

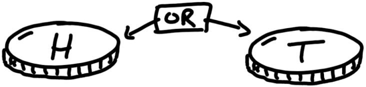
c_1  c_2

Example

- Probability of the first coin being heads or the second being tails?
- Calculated as a sum over all possibilities:

$$\begin{aligned}
&\mathbb{P}(c_1 = H) \cdot \mathbb{P}(c_2 = H) \\
&+ \mathbb{P}(c_1 = H) \cdot \mathbb{P}(c_2 = T) \\
&+ \mathbb{P}(c_1 = T) \cdot \mathbb{P}(c_2 = T) \\
= &\ 1/2 \cdot 1/2 + 1/2 \cdot 1/2 + 1/2 \cdot 1/2 \\
= &\ 3/4
\end{aligned}$$

What is Weighted Model Counting?

c_1  c_2

Example

- Model the two coins with Boolean variables c_1 and c_2
- True if the coin lands heads, false if it lands tails

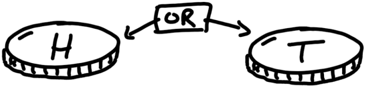
What is Weighted Model Counting?

c_1  c_2

Example

- Model the two coins with Boolean variables c_1 and c_2
- True if the coin lands heads, false if it lands tails
- **Formula:** $c_1 \vee \overline{c}_2$

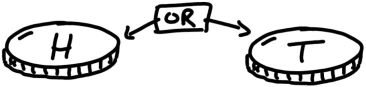
What is Weighted Model Counting?

c_1  c_2

Example

c_1	c_2	$c_1 \lor \overline{c}_2$
0	0	1
0	1	0
1	0	1
1	1	1

What is Weighted Model Counting?

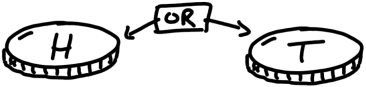
c_1  c_2

Example

c_1	c_2	$c_1 \vee \overline{c}_2$	$W(c_1)$	$W(c_2)$	W
0	0	1	1/2	1/2	1/4
0	1	0	1/2	1/2	1/4
1	0	1	1/2	1/2	1/4
1	1	1	1/2	1/2	1/4

- **Weights (probabilities):** 1/2 for every outcome of every coin

What is Weighted Model Counting?

c_1  c_2

Example

c_1	c_2	$c_1 \vee \overline{c}_2$	$W(c_1)$	$W(c_2)$	W
0	0	**1**	1/2	1/2	**1/4**
0	1	0	1/2	1/2	1/4
1	0	**1**	1/2	1/2	**1/4**
1	1	**1**	1/2	1/2	**1/4**

- **Weights (probabilities):** 1/2 for every outcome of every coin
- **Model count (total probability):** **3/4**

What is Weighted Model Counting?

c_1  c_2

Example

- Probability of two tossed coins coming up with the same side?

What is Weighted Model Counting?

c_1  =  c_2

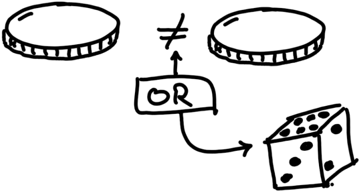
Example

- Probability of two tossed coins coming up with the same side?

c_1	c_2	$c_1 \leftrightarrow c_2$	$W(c_1)$	$W(c_2)$	W
0	0	**1**	1/2	1/2	**1/4**
0	1	0	1/2	1/2	1/4
1	0	0	1/2	1/2	1/4
1	1	**1**	1/2	1/2	**1/4**

- **Model count: 1/2**

What is Weighted Model Counting?



Example

- Probability of two coins being different, or a die rolling six?

What is Weighted Model Counting?

Example

- Probability of two coins being different, or a die rolling six?

What is Weighted Model Counting?

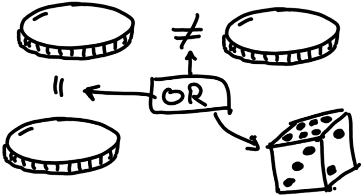
Example

- Probability of two coins being different, or a die rolling six?

c_1	c_2	d	$(c_1 \oplus c_2) \vee d$	$W(c_1)$	$W(c_2)$	$W(d)$	W
0	0	0	0	1/2	1/2	5/6	5/24
0	0	1	**1**	1/2	1/2	1/6	**1/24**
0	1	0	**1**	1/2	1/2	5/6	**5/24**
0	1	1	**1**	1/2	1/2	1/6	**1/24**
1	0	0	**1**	1/2	1/2	5/6	**5/24**
1	0	1	**1**	1/2	1/2	1/6	**1/24**
1	1	0	0	1/2	1/2	5/6	5/24
1	1	1	**1**	1/2	1/2	1/6	**1/24**

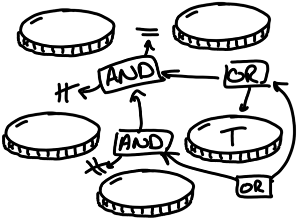
- **Model count: 7/12**

What is Weighted Model Counting?



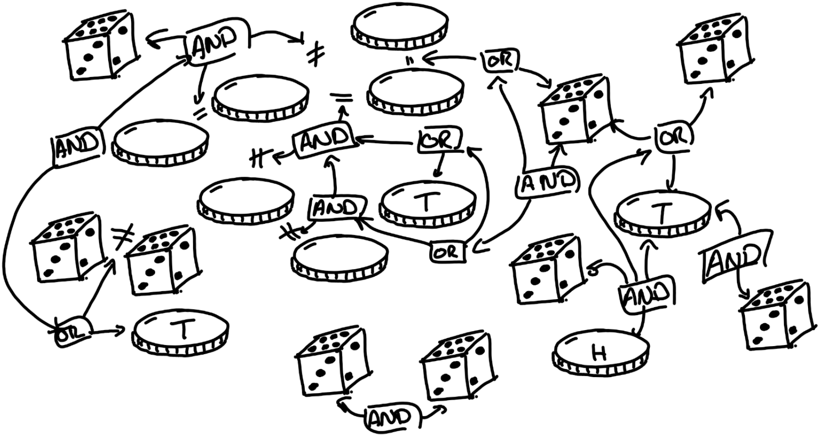
Model count: $7/12$

What is Weighted Model Counting?



Model count: $5/8$

What is Weighted Model Counting?



Model count: 5/8,957,952

What is Weighted Model Counting?

- In general, weights don't have to be probabilities
- Can be any real number

What is Weighted Model Counting?

- In general, weights don't have to be probabilities
- Can be any real number
- Formally: weighted model count of formula ϕ w.r.t. weight function W

$$\mathrm{WMC}(\phi, W) = \sum_{\tau \text{ satisfies } \phi} \prod_{v \in V} W(v, \tau(v))$$

What is Weighted Model Counting?

- In general, weights don't have to be probabilities
- Can be any real number
- Formally: weighted model count of formula ϕ w.r.t. weight function W

$$\mathrm{WMC}(\phi, W) = \sum_{\tau \text{ satisfies } \phi} \prod_{v \in V} W(v, \tau(v))$$

- Problem is #P-hard in general
- Model counters can achieve a significant speedup

Quantum Computation with WMC

Quantum Computation with WMC

- Quantum problems are just matrix operations
  - (on very large matrices)
- Encode matrices with WMC using “pointer” variables

Quantum Computation with WMC

- Quantum problems are just matrix operations
  - (on very large matrices)
- Encode matrices with WMC using “pointer” variables

Example

$$\begin{bmatrix}2\\0\end{bmatrix}\begin{array}{l}\nearrow\ \overline{x}\\[2em]\searrow\ x\end{array}$$

Quantum Computation with WMC

- Quantum problems are just matrix operations
  - (on very large matrices)
- Encode matrices with WMC using “pointer” variables

Example

$$\begin{bmatrix} 2 \\ 0 \end{bmatrix} \begin{array}{l} \nearrow \ \mathrm{WMC}(\phi \wedge \overline{x}, W) \\ \\ \searrow \ \mathrm{WMC}(\phi \wedge x, W) \end{array}$$

- **Formula:** $\phi \equiv \overline{x}$
- **Weights:** $W(\overline{x}) = W(x) = 2$

Quantum Computation with WMC

Example

- For larger vectors, use multiple variables:

$$\begin{bmatrix} 2 \\ 0 \\ 0 \\ 1 \end{bmatrix} \begin{matrix} \longrightarrow \overline{x}_1, \overline{x}_2 \\ \longrightarrow \overline{x}_1, x_2 \\ \longrightarrow x_1, \overline{x}_2 \\ \longrightarrow x_1, x_2 \end{matrix}$$

Quantum Computation with WMC

Example

- For larger vectors, use multiple variables:

$$\begin{bmatrix} 2 \\ 0 \\ 0 \\ 1 \end{bmatrix} \begin{array}{l} \longrightarrow \mathrm{WMC}(\phi \wedge \overline{x}_1 \wedge \overline{x}_2, W) \\ \longrightarrow \mathrm{WMC}(\phi \wedge \overline{x}_1 \wedge x_2, W) \\ \longrightarrow \mathrm{WMC}(\phi \wedge x_1 \wedge \overline{x}_2, W) \\ \longrightarrow \mathrm{WMC}(\phi \wedge x_1 \wedge x_2, W) \end{array}$$

- **Formula:** $\phi \equiv x_1 \leftrightarrow x_2$
- **Weights:** $W(\overline{x}_1) = 2,\ W(x_1) = 1,\ W(\overline{x}_2) = W(x_2) = 1$

Quantum Computation with WMC

Example

- Use "input and output" variables for matrices:

$$\begin{array}{ccc} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} & \begin{array}{l} \longrightarrow \overline{y} \\ \longrightarrow y \end{array} \\ \begin{array}{cc} \downarrow & \downarrow \\ \overline{x} & x \end{array} & \end{array}$$

Quantum Computation with WMC

Example

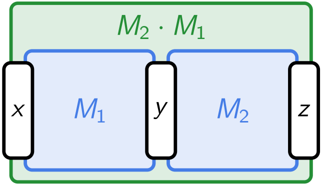
- Use "input and output" variables for matrices:

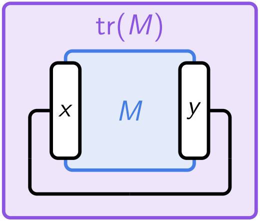
$$\begin{array}{cc} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} & \begin{array}{l} \longrightarrow \overline{y} \\ \longrightarrow y \end{array} \\ \begin{array}{cc} \downarrow & \downarrow \\ \overline{x} & x \end{array} & \end{array}$$

- **Formula:** $\phi \equiv x \oplus y$
- **Weights:** $W(\overline{x}) = W(x) = W(\overline{y}) = W(y) = 1$

Quantum Computation with WMC

- Common matrix operations can be performed on these "WMC representations"



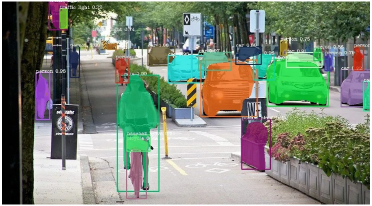


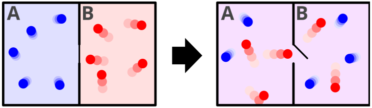
- If we need an entry from the matrix, we use a model counter

Application:
Potts Model

Potts Model

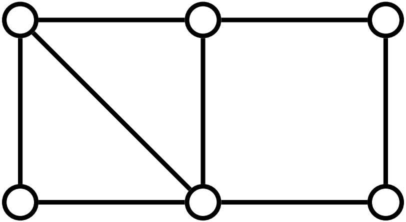
- Model of interactions between particles
- Used in image segmentation and statistical mechanics





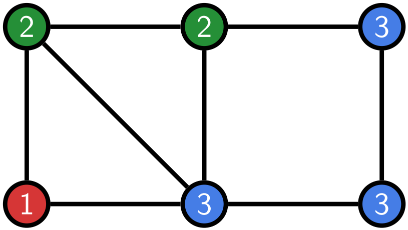
Potts Model

- Graph with interactions between "sites"



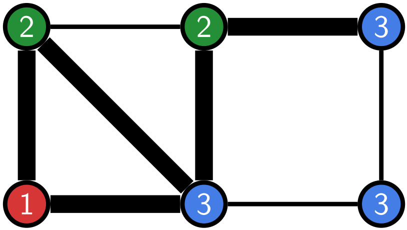
Potts Model

- Sites can be assigned a "spin"
- Number between 1 and q



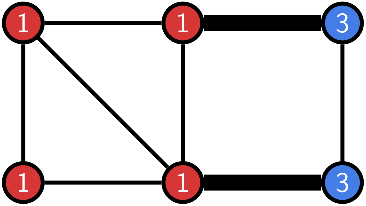
Potts Model

- Neighboring sites with different spins create “energy”

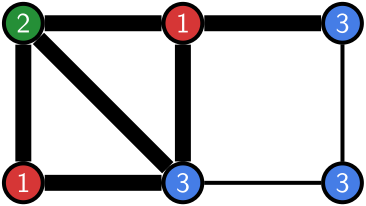


Potts Model

- Neighboring sites with different spins create “energy”



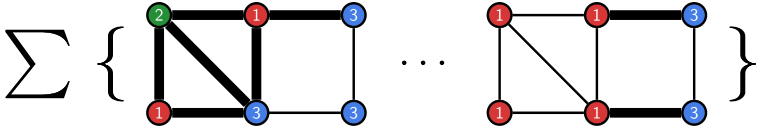
Low energy



High energy

Potts Model

- **Goal:** Calculate the partition function
- Sum over energies of all spin combinations
  - q^n different combinations
- Normalizing factor of Gibbs distribution



Potts Model

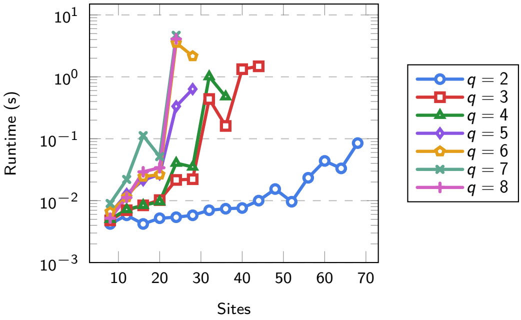
- Partition function can be written as the trace of a $q^n \times q^n$ diagonal matrix:

$$Z = \mathrm{tr}\left(\prod_{(i,j)\in E} e^{\beta J M_{ij}}\right)$$

- Use our framework by building up from smaller matrices

Potts Model

- Performance is good:
  - For small q, can calculate partition function quickly up to $n \approx 40$ sites
  - With brute force, this would take $\sim$400 million years



- In line with earlier results on the simpler Ising model

Application: Transverse-Field Ising Model

Transverse-Field Ising Model

- "Quantum version" of the Potts model
- Interactions are much more complex

Transverse-Field Ising Model

- “Quantum version” of the Potts model
- Interactions are much more complex
- Partition function is the trace of a non-diagonal matrix
  - No longer an easy sum to feed into a model counter

Transverse-Field Ising Model

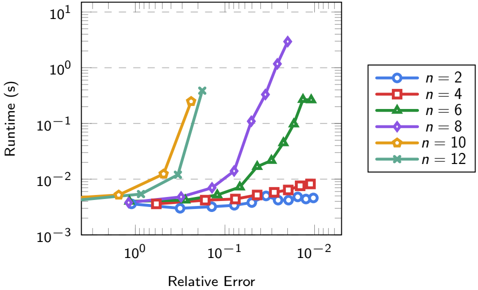
- "Quantum version" of the Potts model
- Interactions are much more complex
- Partition function is the trace of a non-diagonal matrix
  - No longer an easy sum to feed into a model counter
- Partition function can be approximated with

$$Z \approx \mathrm{tr}\left(\left[\prod_{i,j\in\Lambda} e^{J_{ij}Z_i\otimes Z_j/k}\prod_{i\in\Lambda} e^{\mu_z Z_i/k}\prod_{i\in\Lambda} e^{\mu_x X_i/k}\right]^k\right)$$

- Looks complicated, but it's just a matrix: Use the framework and we're fine

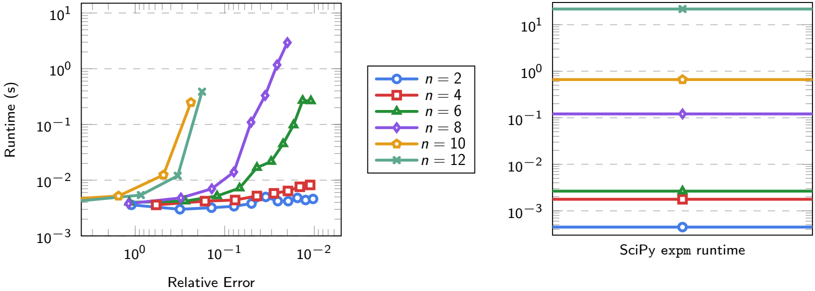
Transverse-Field Ising Model

- Good performance up to $n \approx 8$ qubits



Transverse-Field Ising Model

- Good performance up to $n \approx 8$ qubits



- Existing tools are still faster, but model counters are constantly improving!

Conclusion & Future Work

Conclusion

- We built a framework for applying WMC to problems that involve matrices
- Particularly useful in quantum applications
- Promising results on Potts and transverse-field Ising models
  - As model counters improve, our framework will become more useful

Future Work

- Use of different techniques, such as Max-WMC, could allow for finding ground states of quantum systems
- Model counters are constantly improving: This could allow for more applications
  - Effective simulation of large quantum systems
  - Modeling systems with complex interactions