University of Udine

Speeding up Answer Set Programming by Quantum Computing

Davide della Giustina[1] Stefano Pessotto[1]
Carla Piazza[1] Riccardo Romanello[1]

[1] UNIVERSITY OF UDINE, ITALY

June 28, 2024

Our Goal

- Quantum algorithms like Shor and Grover have proven the so-called *Quantum speed-up*
- Answer Set Programming (ASP) allows to solve NP-complete problems in a relatively short amount of time
- ASP solutions are found through a visit of a search space
- Our goal is to use Quantum Computation to speed up such research

Results

- Solutions to ASP programs are called models
- The search space is made of all the possible models
- We are interested in finding only a small subset of models called *stable*
- We obtained a Quantum Algorithm to speed-up the research of such particular models

Plan of the Talk

- Quantum Computation: a crash course
- Answer Set Programming: the theoretical minimum
- Three ingredients:
  - Grover search for ASP solutions
  - Quantum Weighted Model Counting
  - Rushing and Strolling through models
- Our Algorithm

- Quantum Computing only allows reversible operations
- Quantum states are described through normalized vectors inside $\mathbb{C}^{2^n}$ for some n
- States are denoted using *bra-ket* notation

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

- A generic quantum state has the form

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$$

- Quantum states can be manipulated only using unitary matrices
- Let $U \in \mathbb{C}^{2^n \times 2^n}$. Then U is unitary if and only if $UU^\dagger = I$
- Some examples of Unitary matrices are:

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \qquad S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \qquad T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{pmatrix}$$

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Grover's Algorithm

A simple problem

- **Input**: a function $\chi : \{0,1\}^n \mapsto \{0,1\}$
- **Output**: a value $x \in \{0,1\}^n$ such that $\chi(x) = 1$

A simple problem

- **Input**: a function $\chi : \{0,1\}^n \mapsto \{0,1\}$
- **Output**: a value $x \in \{0,1\}^n$ such that $\chi(x) = 1$

A Classical Approach

- Test all the possible inputs, until you find a solution (if it exists)
- Complexity: $\mathcal{O}(N)$, where $N = 2^n$

A simple problem

- **Input**: a function $\chi : \{0,1\}^n \mapsto \{0,1\}$
- **Output**: a value $x \in \{0,1\}^n$ such that $\chi(x) = 1$

A Classical Approach

- Test all the possible inputs, until you find a solution (if it exists)
- Complexity: $\mathcal{O}(N)$, where $N = 2^n$

A Quantum Approach

- Adopt a *Quantum Oracle* $\mathcal{O}_\chi$ for the function χ
- Solves the problem with $\mathcal{O}(\sqrt{N})$ calls to the oracle

(Another) simple problem

- **Input**: a function $\chi : \{0,1\}^n \mapsto \{0,1\}$
- **Output**: $\left|\bar{X} := \{\bar{x} \in \{0,1\}^n : \chi(\bar{x}) = 1\}\right|$

(Another) simple problem

- **Input**: a function $\chi : \{0,1\}^n \mapsto \{0,1\}$
- **Output**: $\left|\bar{X} := \{\bar{x} \in \{0,1\}^n : \chi(\bar{x}) = 1\}\right|$

A Classical Approach

- Test all the possible inputs, count how many satisfy χ
- Complexity: $\theta(N)$, where $N = 2^n$

Quantum Counting

(Another) simple problem

- **Input**: a function $\chi : \{0,1\}^n \mapsto \{0,1\}$
- **Output**: $\left|\bar{X} := \{\bar{x} \in \{0,1\}^n : \chi(\bar{x}) = 1\}\right|$

A Classical Approach

- Test all the possible inputs, count how many satisfy χ
- Complexity: $\theta(N)$, where $N = 2^n$

A Quantum Approach

- The eigenvalues of $\mathcal{O}_\chi$ have the form $e^{\imath\theta}$ and $e^{\imath(2\pi-\theta)}$
- Compute such eigenvalues using phase estimation algorithm
- $|\bar{X}| := 2N \sin^2 \frac{\theta}{2}$

- **Program** $\Pi := \{p \leftarrow \neg q, q \leftarrow \neg p, r \leftarrow p, r \leftarrow q\}$
- **Interpretation:** binds some values to constants, functions and relations. For the above (ground) program, $S := \{p, r\}$ and $S' := \{p, q, r\}$ are both interpretations
- **Model:** let I be an interpretation for Π. I is a model (answer set) if it satisfies all the rules of Π.
- **Stable model:** A model I is said to be stable if it is minimal with respect to set inclusion—no smaller set $I' \subset I$ satisfies the same set of rules

- **What**: Count and find stable models of an ASP program
- **How**: Grover's algorithm with an oracle that marks stable models only
- **Type of algorithm**: Quantum, to solve a classical problem

Weighted Count of Propositional Models [3]

- **What**: Weighted Count—some solution are *better* than the other—the number of interpretations that satisfy a propositional formula (WMC)
- **How**: Tweak Quantum Counting to take into account *weights* given to solutions
- **Type of algorithm**: Quantum, to solve a classical problem

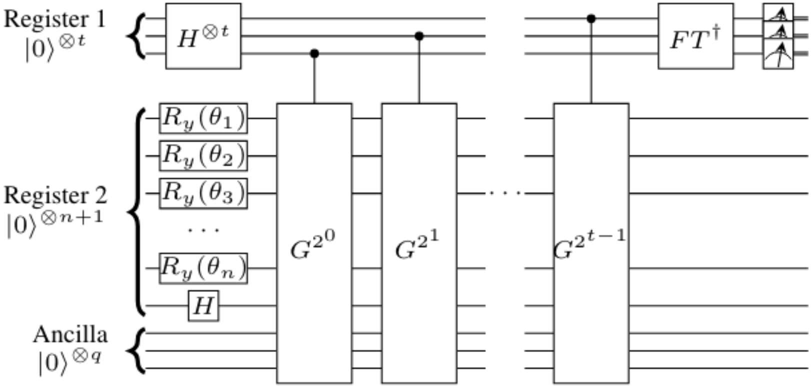

Figure 1: The Quantum Weighted Model Counting Circuit

Classically navigating Stable Models [1]

- **What**: Navigate in a fast fashion through the solution space of an ASP program to find stable models
- **How**: Introduce the notion of facets to move from one model to another.
  Moreover, a weight is assigned to each facet to guide the visit.
- **Type of algorithm**: Classical, to solve a classical problem

What are these facets?

Let $\Pi, \mathcal{AS}(\Pi)$ be an ASP program, and the set of its stable models, respectively.

- $\mathcal{BC}(\Pi) := \bigcup \mathcal{AS}(\Pi)$ the set of Π's *brave consequences*;
- $\mathcal{CC}(\Pi) := \bigcap \mathcal{AS}(\Pi)$ the set of its *cautious consequences*.

The set of *facets* of Π is then defined as follows:

$$\mathcal{F}(\Pi) := \mathcal{F}^+(\Pi) \cup \mathcal{F}^-(\Pi) \tag{1}$$

where:

$$\mathcal{F}^+(\Pi) := \mathcal{BC}(\Pi) \backslash \mathcal{CC}(\Pi) \qquad \mathcal{F}^-(\Pi) := \{\bar{\alpha} : \alpha \in \mathcal{F}^+(\Pi)\} \tag{2}$$

Let facets cook

- An interpretation S for Π satisfies an *inclusive facet* $\alpha \in \mathcal{F}^+(\Pi)$ if $\alpha \in S$.
- S satisfies an *exclusive facet* $\bar{\alpha} \in \mathcal{F}^-(\Pi)$ if $\alpha \notin S$.
- We denote by $ic(f)$ the singleton ASP program defined as follows :

$$ic(f) := \begin{cases} \{\leftarrow \neg\alpha\} & \text{if } f \text{ is an atom } \alpha \\ \{\leftarrow \alpha\} & \text{otherwise} \end{cases} \quad (3)$$

where $f \in \mathcal{F}(\Pi)$.

Navigating using facets

Let Π be an ASP program.

- The activation of a facet $f \in \mathcal{F}(\Pi)$ results in the program $\Pi' := \Pi \cup ic(f)$
- The activation of f denotes a *navigation step* from Π to Π'
- A finite sequence $\delta := \langle f_1, \ldots, f_k \rangle$ of facets defines a *route*, and denotes an ordered sequence of navigation steps from an initial program Π

Our contribution

Let Π be an ASP program. We denote with $\mathcal{AS}(\Pi)$ the set of its stable models.

- In [1], authors proposed different weight functions to guide the navigation of the solution space of Π
- One of these functions—namely $w_{\#\mathcal{AS}}$—showed very good results
- It is defined as follows:

$$w_{\#\mathcal{AS}}(f, \Pi^{\delta}) := |\mathcal{AS}(\Pi^{\delta})| - |\mathcal{AS}(\Pi^{\langle \delta, f \rangle})|$$

- Unfortunately, its computation is #coNP-*complete*

Our Contribution - This time for real

- We developed a Quantum Algorithm to compute the value of $w_{\#\mathcal{AS}}$
- Such algorithm relies on the solution to the Weighted Model Counting Problem

Technical Details

- Let χ be a boolean function that outputs 1 if and only if its input encodes a stable model of Π
- To apply WMC, we define a *weight* function w as follows:

$$w(i,1) := \begin{cases} 1 & \text{if } \exists\, j = 1,\ldots,k\ f_j = \alpha_i \\ 0 & \text{if } \exists\, j = 1,\ldots,k\ f_j = \bar{\alpha}_i \\ 1/2 & \text{otherwise} \end{cases} \qquad \forall\, i = 1,\ldots,n \tag{4}$$

  and set each $w(i,0)$ so that w is normalized.
- Suppose k is the number of facets used for the navigation. What we proved is as follows:

$$|\mathcal{AS}(\Pi^\delta)| = 2^{n-k} \cdot \mathrm{WMC}(\chi, w)$$

- Hence, by using the value computed via WMC, we can compute $w_{\#\mathcal{AS}}$ with a quadratic speed-up over any classical method

Conclusion

- We tackled the problem of speeding-up Answer Set Programming using Quantum Computation
- We focused on the task of searching for stable models of an ASP program
- We started from an existing classical method to visit the search space
- We introduced a quantum routine to compute the function $w_{\#\mathcal{AS}}$ which was proven to be very effective when adopted to guide such visit

[1] **Johannes K. Fichte, Sarah A. Gaggl, and Dominik Rusovac.**
Rushing and Strolling among Answer Sets – Navigation Made Easy.
CoRR, 12 2021.

[2] **David A. Meyer, James Pommersheim, and Jeffrey B. Remmel.**
Finding stable models via quantum computation.
In *International Workshop on Non-Monotonic Reasoning*, 01 2004.

[3] **Fabrizio Riguzzi.**
Quantum Weighted Model Counting.
In *European Conference on Artificial Intelligence*, 2019.