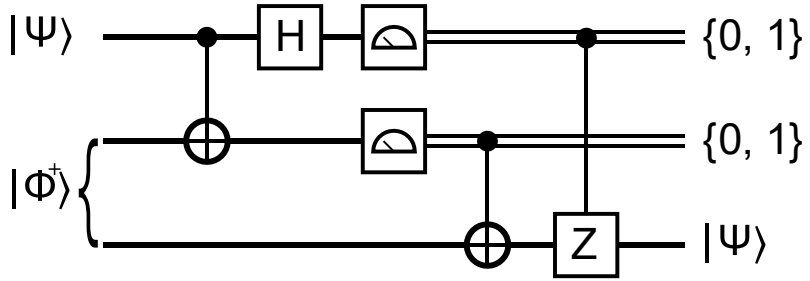
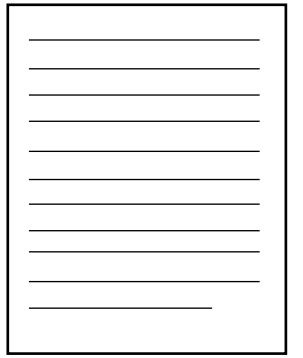
Quantum Computer simulation improving

Hernán I. de la Cruz, J. Javier Paulet, Fernando Cuartero, Fernando L. Pelayo, Jesús Andrés
ReTiCS, ESIIAB, UCLM, Albacete, Spain
{HernanIndibil.LaCruz, JoseJavier.Paulet, Jesus.Andres}@alu.uclm.es
{Fernando.Cuartero, FernandoL.Pelayo}@uclm.es





Results

1. Motivation

Quantum Simulation

Pros:

- Independent from physical setup
- Accessible
- Number of qubits

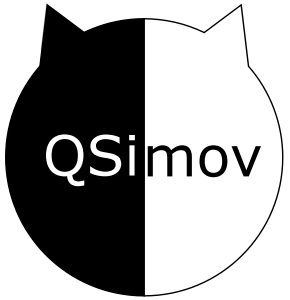
Cons:

- Exponential complexity in both time and memory
  - Currently (P≠NP has not been proven) no benefits between best classical algorithm and best quantum simulated algorithm

2. Quantum Simulator *QSimov*

I3A Instituto de Investigación en Informática de Albacete

- Core written in C
  - Parallelized with OpenMP
- User Interface Library written in Python 3
  - Easy to learn



It’s able to simulate up to 30 entangled qubits (due to *malloc* restrictions that will be removed in the next version) with little time/memory restrictions (can be run in a laptop).

2. Quantum Simulator *QSimov*

I3A Instituto de Investigación en Informática de Albacete

Deutsch-Josza circuit generator function:

```
import qsimov as qj


def DJAlgCircuit(size, U_f):
    c = qj.QCircuit("Deutsch-Josza Algorithm", ancilla=[1])

    c.addLine(*["H" for i in range(size)])
    c.addLine(U_f)
    c.addLine(*("H" for i in range(size - 1)), "I")

    c.addLine(qj.Measure([1 for i in range(size - 1)] + [0]))

    return c
```

3. Simulation problem

I3A
Instituto de Investigación en Informática de Albacete

Let *n* be the number of qubits in the system, and *s* be the number of qubits affected by the gate, with $s \leq n$.

Spatial complexity	**Time complexity**
State (vector): $\Omega(2n)$, $O(2^n)$ Gate (matrix): $\Omega(2^{2s})$, $O(2^{2n})$ Gate (QSimov): $\Theta(2^{2s})$	Apply Gate (Gate|State⟩): $\Omega(2^{2s})$, $O(2^{2n})$ Apply Gate (QSimov): $\Omega(2^s)$, $O(2^{s+n})$ Tensor Product of two states: $\Theta(2^{n_1}+2^{n_2})$

4. Conclusions and future work

I3A Instituto de Investigación en Informática de Albacete

Conclusions:

- Data structures play a key role when dealing with computationally expensive tasks as simulating a quantum computer.
- They can greatly reduce the complexity of a problem

Future work:

- A graphic front-end
- Turn QSimov into a Middle-ware between the user and a real quantum computer