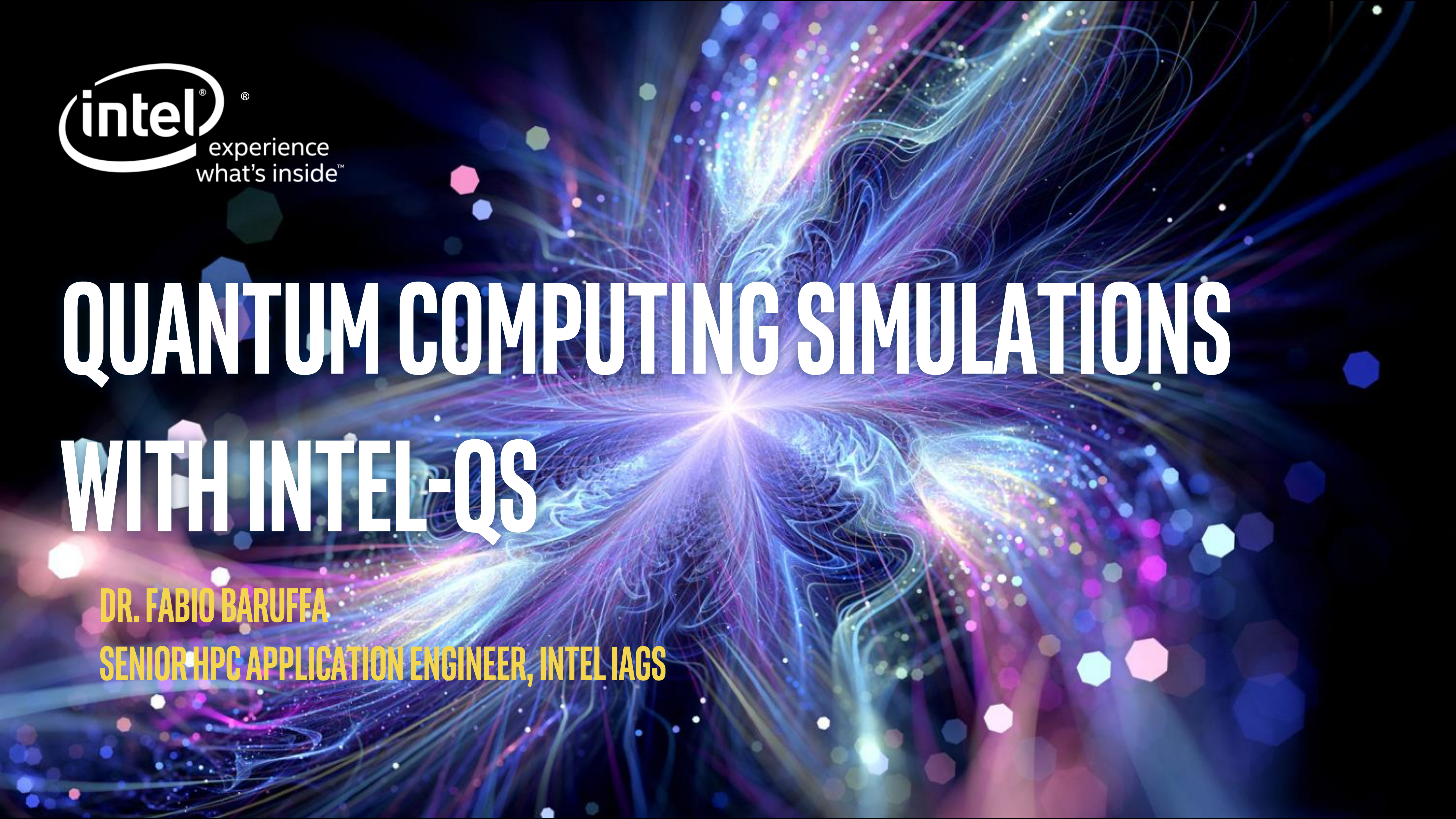


QUANTUM COMPUTING SIMULATIONS WITH INTEL-QS

DR. FABIO BARUFFA

SENIOR HPC APPLICATION ENGINEER, INTEL IAGS

QUANTUM COMPUTING: KEY CONCEPTS

Superposition

Classical Physics

Heads or Tails

Quantum Physics

Heads and Tails

Entanglement

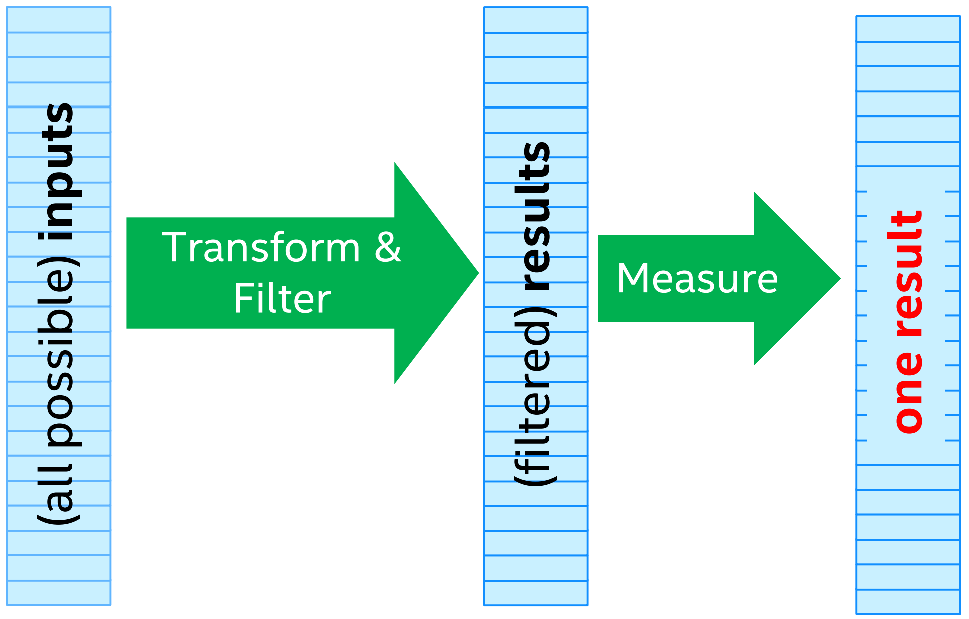
N Quantum Bits or **Qubits =** 2^N States

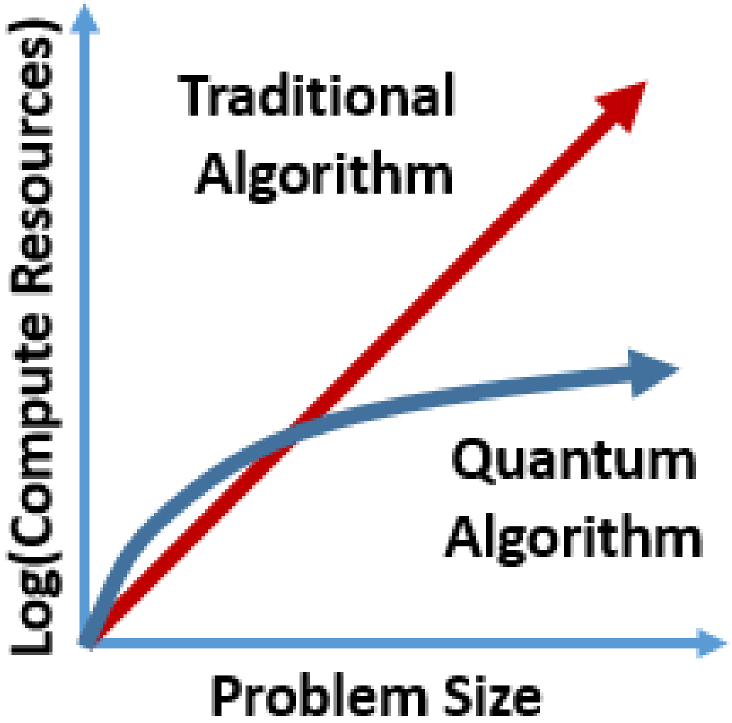
Fragility

Observation or noise causes loss of information

- 50 Entangled Qubits = more states than any possible supercomputer
- 300 Entangled Qubits = more states than atoms in the universe
- Fragility will require error correction and likely millions of qubits

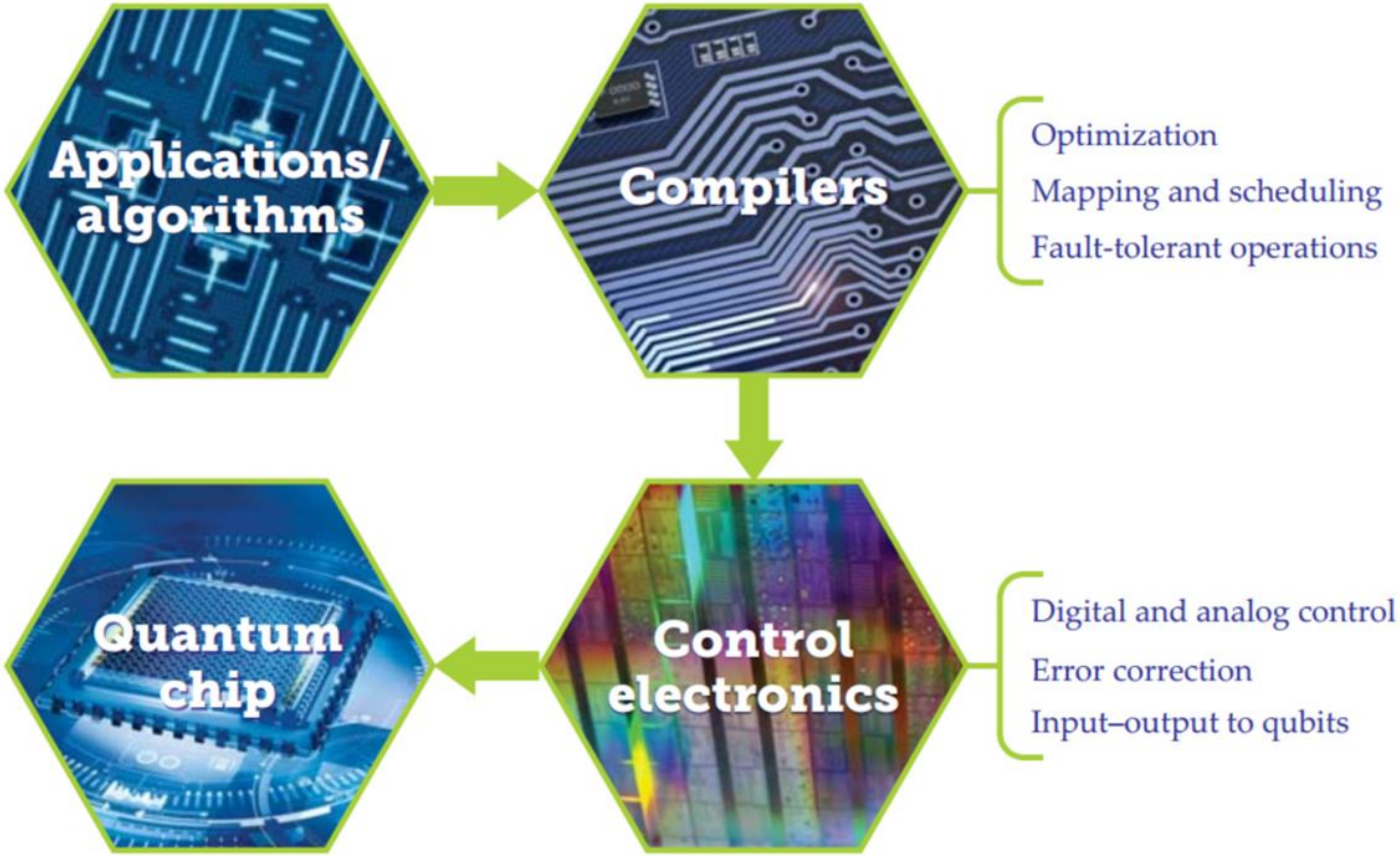
THE PROMISE OF QUANTUM COMPUTING





Exponential speedup ←→ surpassing the limits of scaling

Functionalities Necessary for a Quantum Computer



Optimization
Mapping and scheduling
Fault-tolerant operations

Digital and analog control
Error correction
Input–output to qubits

How does one create a quantum computing system that takes a quantum algorithm as input and automatically performs a computation on qubits?

A systems perspective of quantum computing: https://doi.org/10.1063/PT.3.4163

Augmenting the Traditional HPC Systems

~50+ Qubits: Proof of concept

- Computational power exceeds supercomputers
- Learning test bed for quantum “system”

~1000+ Qubits: Small problems

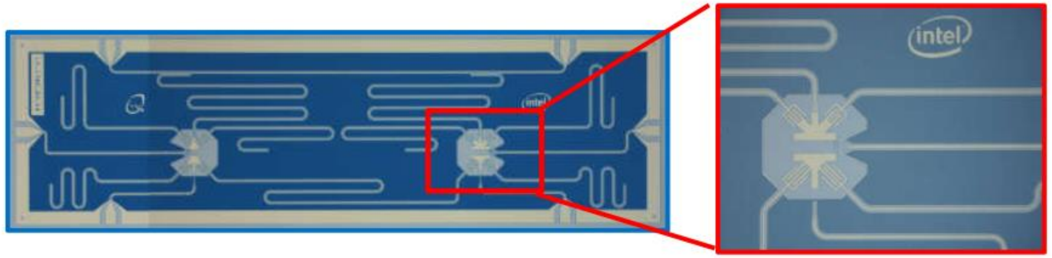
- Limited error correction
- Chemistry, material design
- Optimization

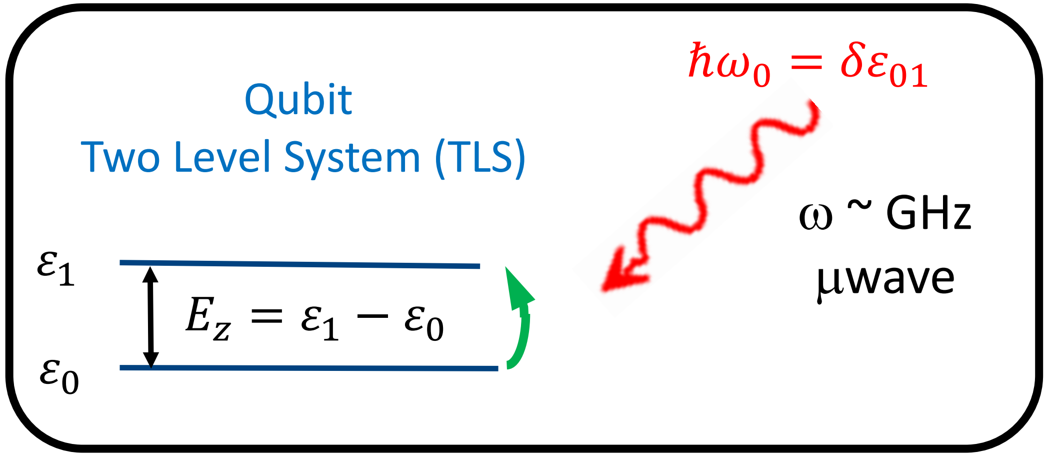
~1M+ Qubits: Commercial scale

- Fault tolerant operation
- Cryptography
- Machine Learning

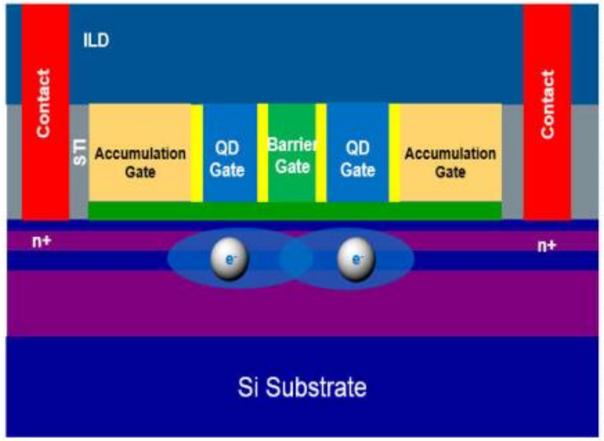
BUILDING QUBITS

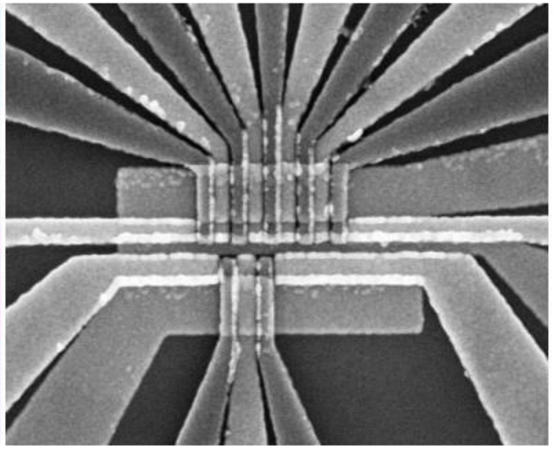
Superconducting Qubits

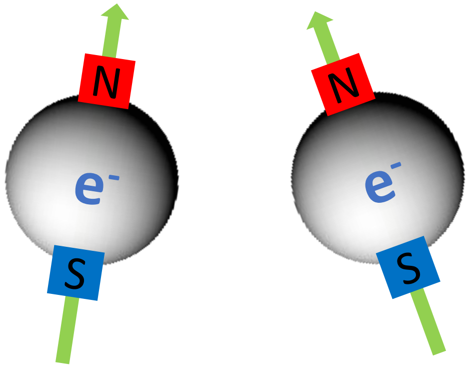



Qubit
Two Level System (TLS)
ε_1
ε_0
$E_z = \varepsilon_1 - \varepsilon_0$
$\hbar\omega_0 = \delta\varepsilon_{01}$
ω ~ GHz
μwave

Spin Qubits in Silicon



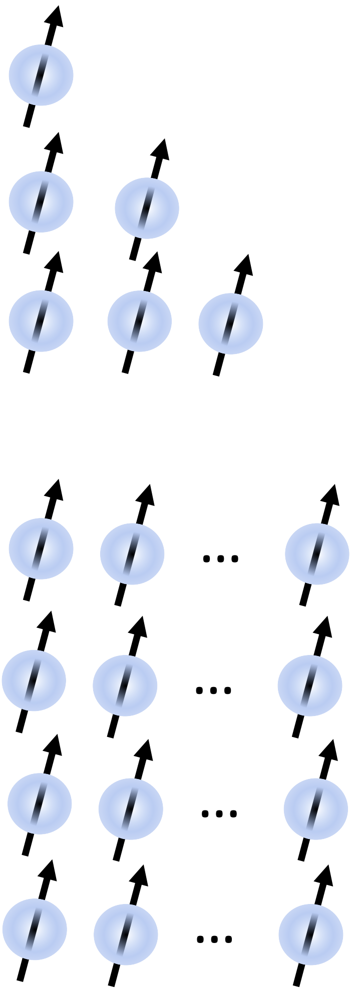




Single electron transistor

WHAT CAN WE DO NOW?
SIMULATING QUANTUM ALGORITHMS ON CURRENT SYSTEMS

THE POWER OF QUANTUM COMPUTING: EXPONENTIAL COMPLEXITY



1 $|0\rangle, |1\rangle$ → $2 = 2^1$

2 $|00\rangle, |01\rangle, |10\rangle, |11\rangle$ → $4 = 2^2$

3 $|000\rangle, |001\rangle, |010\rangle, \ldots, |111\rangle$ → $8 = 2^3$

10 $|00\ldots\rangle, |00\ldots1\rangle, \ldots, |11\ldots1\rangle$ → $1k = 2^{10}$

20 $|00\ldots\rangle, |00\ldots1\rangle, \ldots, |11\ldots1\rangle$ → $1M = 2^{20}$

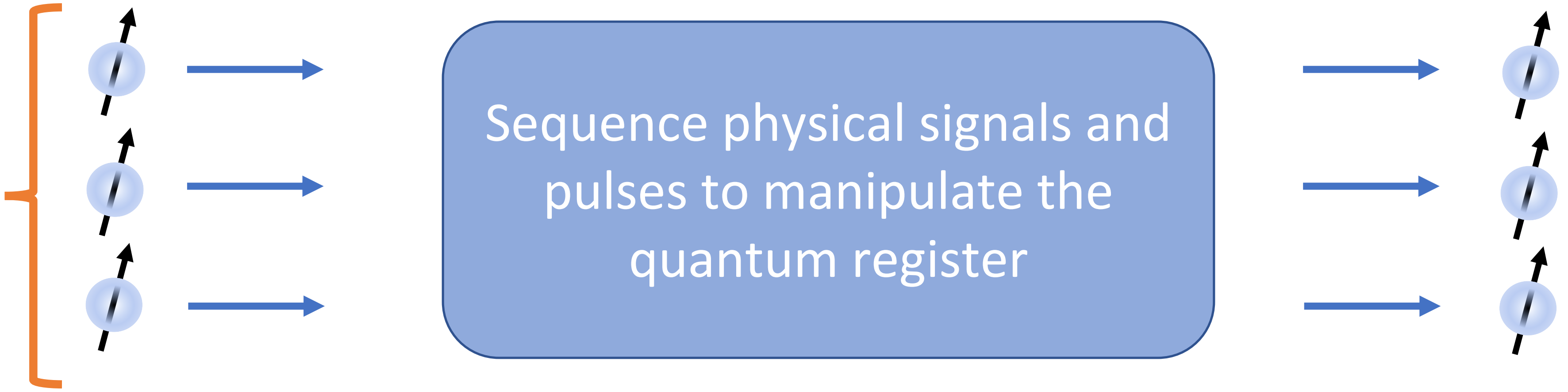
30 $|00\ldots\rangle, |00\ldots1\rangle, \ldots, |11\ldots1\rangle$ → $1G = 2^{30}$

40 $|00\ldots\rangle, |00\ldots1\rangle, \ldots, |11\ldots1\rangle$ → $1T = 2^{40}$

QUANTUM ALGORITHM

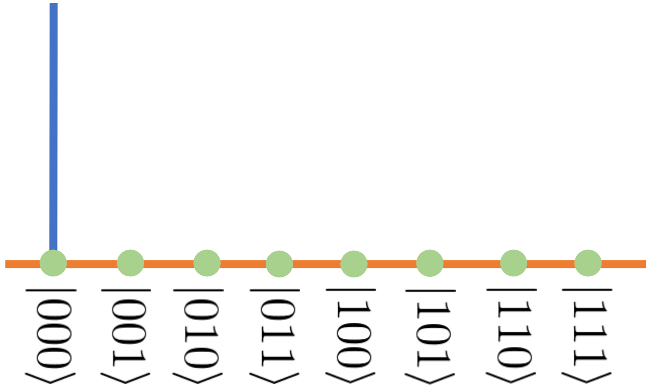
Physically

N input qubit



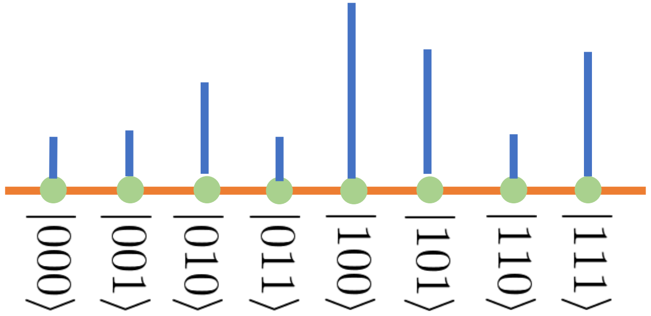
Sequence physical signals and pulses to manipulate the quantum register

Practically



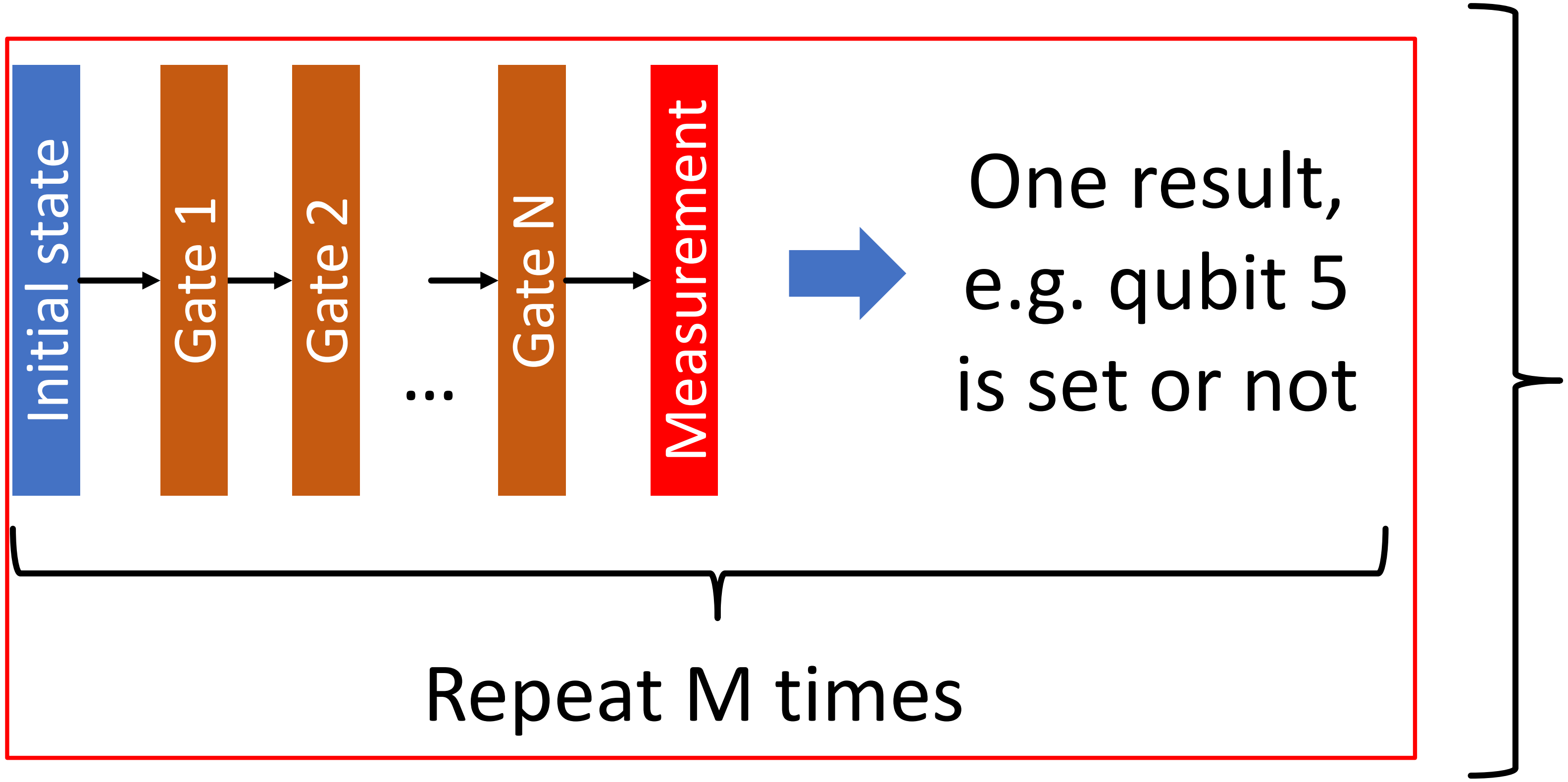
Probability distribution over the classical states

Sequence of Quantum Gates



Maximizing the probability of the good solution

QUANTUM COMPUTING – SPEED UP



One result, e.g. qubit 5 is set or not

Repeat M times

many results, e.g. qubit 5 is set In 75% of the cases

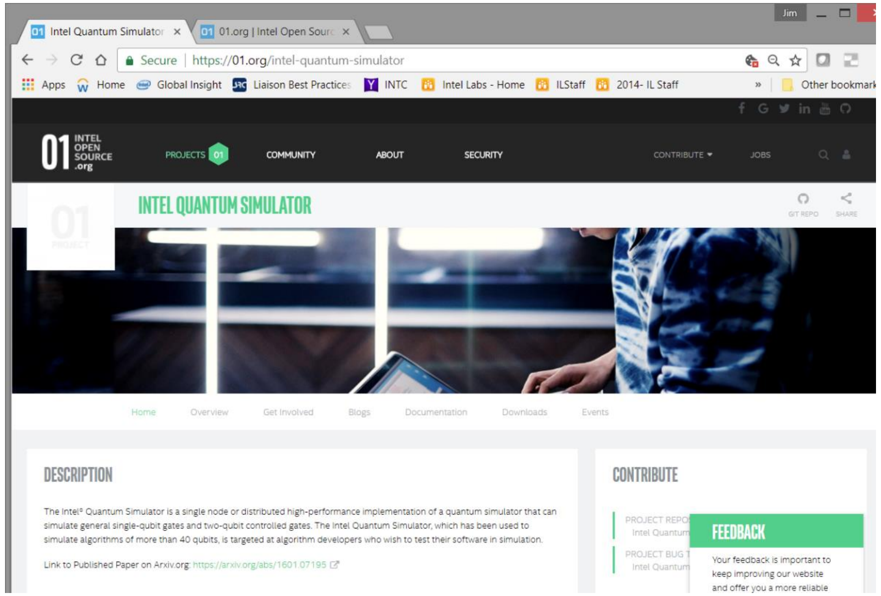
The number of gates times the number of repetitions determines the execution time, not necessarily the size of the problem!

INTEL QUANTUM SIMULATOR

https://github.com/intel/intel-QS

The **Intel® Quantum Simulator** is a single node or distributed high-performance implementation of a quantum simulator that can simulate general single-qubit gates and two-qubit controlled gates.

It is based on a complete representation of the qubit register state in terms of a distributed vectors.



DESCRIPTION

The Intel® Quantum Simulator is a single node or distributed high-performance implementation of a quantum simulator that can simulate general single-qubit gates and two-qubit controlled gates. The Intel Quantum Simulator, which has been used to simulate algorithms of more than 40 qubits, is targeted at algorithm developers who wish to test their software in simulation.

Link to Published Paper on Arxiv.org: https://arxiv.org/abs/1601.07195

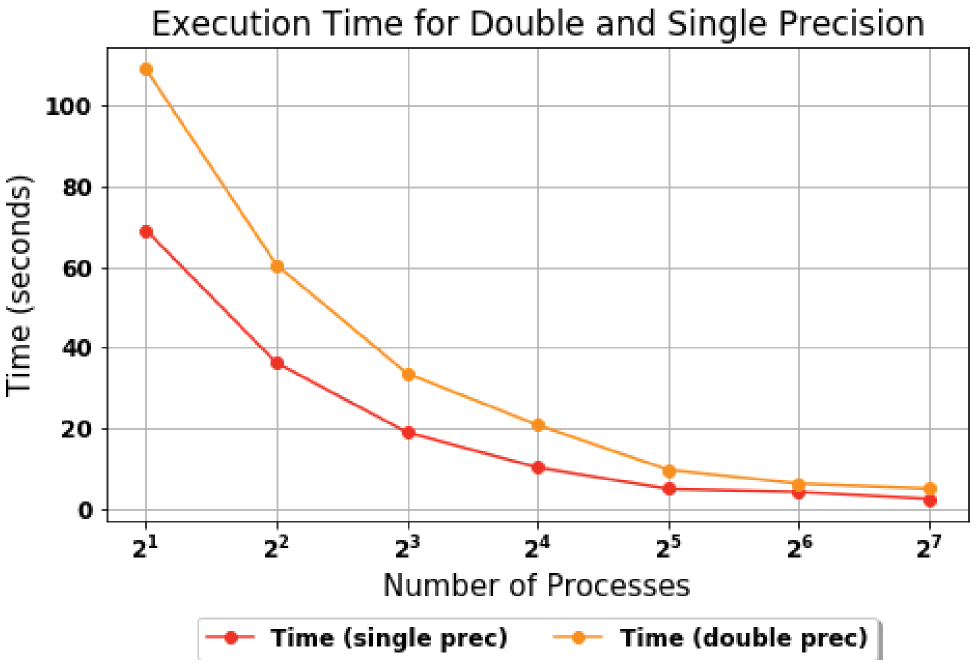
CONTRIBUTE

PROJECT REPO
Intel Quantum

PROJECT BUG T
Intel Quantum

FEEDBACK

Your feedback is important to keep improving our website and offer you a more reliable



QUANTUM NATURAL LANGUAGE PROCESSING

Distributional Compositional Semantics (DisCo) model [1,2,3]

- NLP algorithms to compute meanings of two sentences and decide if their meanings match
- Incorporates grammatical structure of sentences in a language into the analysis algorithms

Quantum advantage in memory requirement during analysis

- *Example*: word-meaning space of a corpus, if based on 2000 most common words
  - One transitive verb: ~1 GB in classical, 33 qubits in quantum
  - 10K transitive verbs: ~10 TB in classical, 47 qubits in quantum

[1]. William Zeng and Bob Coecke, "Quantum Algorithms for Compositional Natural Language Processing", Proceedings of SLPCS, 2016.

[2]. Stephen Clark, Bob Coecke and Mehrnoosh Sadrzadeh, "A Compositional Distributional Model of Meaning", Proceedings of 2nd Quantum Interaction Symposium, 2008.

[3]. Bob Coecke, Mehrnoosh Sadrzadeh and Stephen Clark, "Mathematical Foundations of a Compositional Distributional Model of Meaning", Special issue of Linguistic Analysis, 2010.

DISCO MODEL

DisCo Model

- Sentence meaning determined by combining word adjacency in text corpus + known meanings of component (basis) words
- Sentences are represented as tensor products of each individual word (nouns, verbs, etc)
- Convert graphical notation directly to quantum mechanical representation

Example

- Sentence structure: noun-verb-noun. "*John eats cake*", "*John likes dogs*".

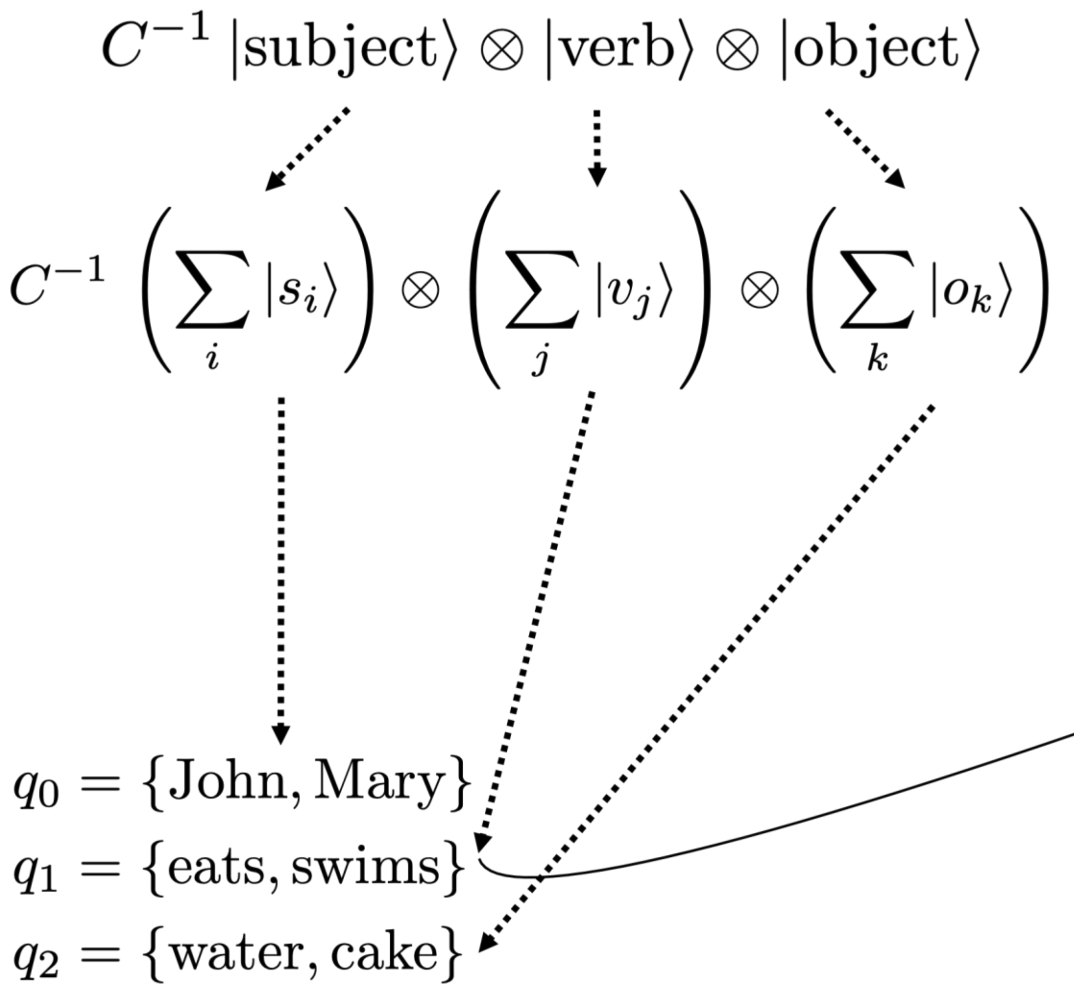
$$\text{John} \in \mathcal{N} \qquad\qquad \text{eats} \in \mathcal{N} \otimes \mathcal{S} \otimes \mathcal{N}$$

- Entire meaning space given by the tensor product:

$$\mathcal{N} \otimes (\mathcal{N} \otimes \mathcal{S} \otimes \mathcal{N}) \otimes \mathcal{N}$$

EXAMPLE REPRESENTATION

“John eats cake, Mary swims in water”



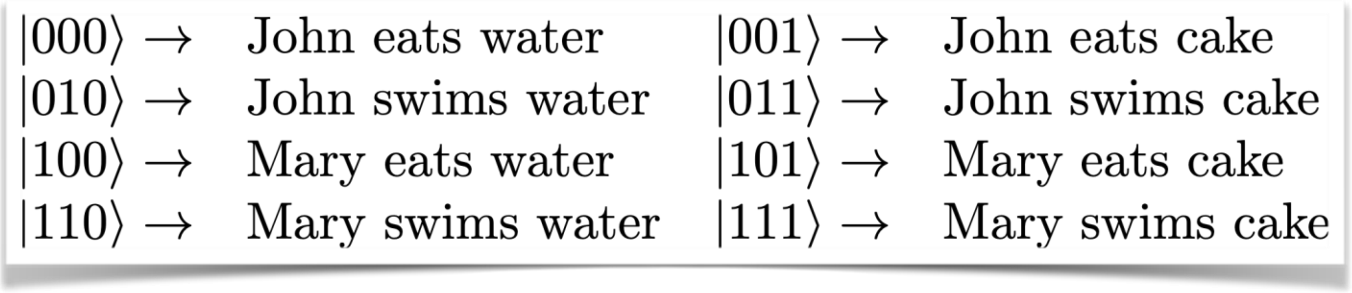
$$C^{-1}\,|\text{subject}\rangle \otimes |\text{verb}\rangle \otimes |\text{object}\rangle$$

$$C^{-1}\left(\sum_i |s_i\rangle\right) \otimes \left(\sum_j |v_j\rangle\right) \otimes \left(\sum_k |o_k\rangle\right)$$

$$q_0 = \{\text{John}, \text{Mary}\}$$

$$q_1 = \{\text{eats}, \text{swims}\}$$

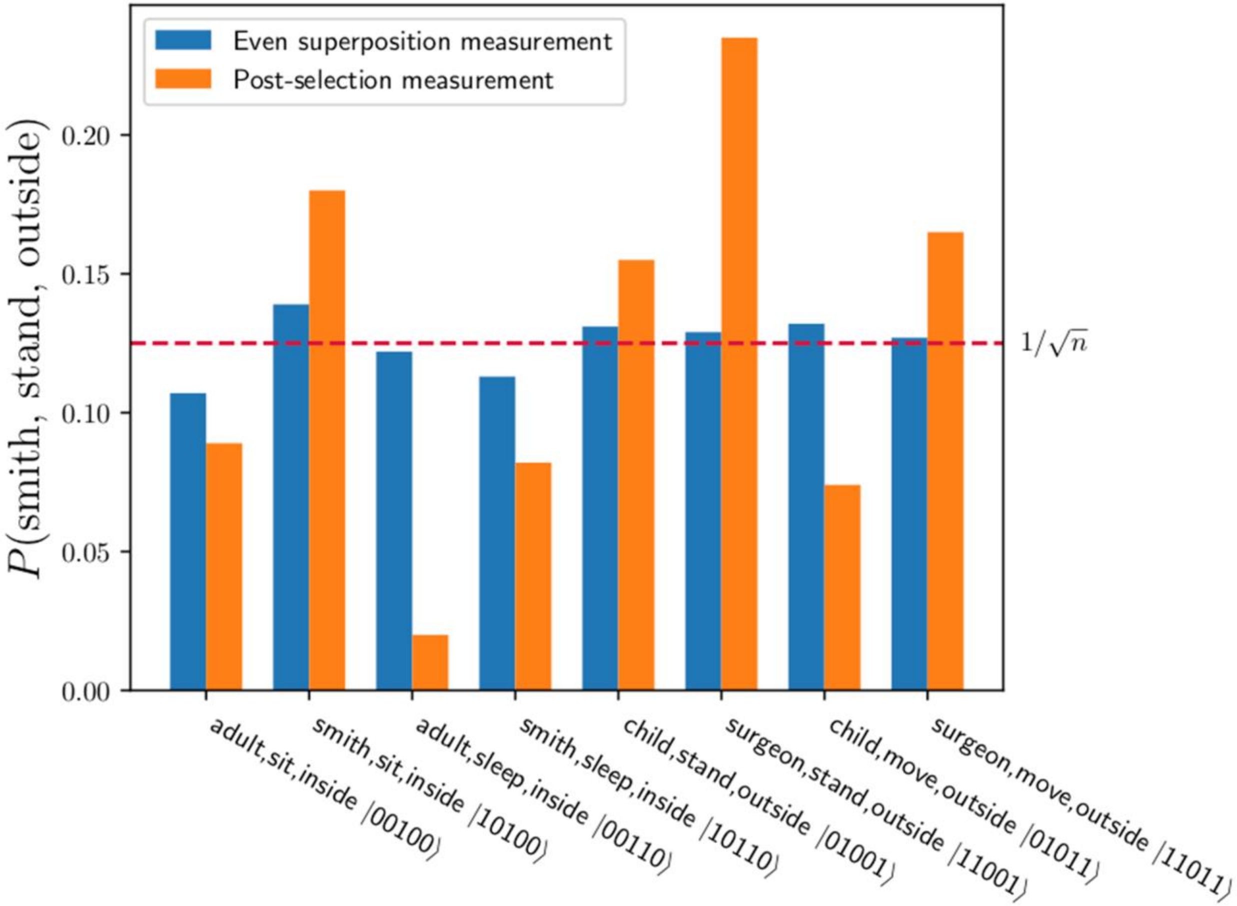
$$q_2 = \{\text{water}, \text{cake}\}$$



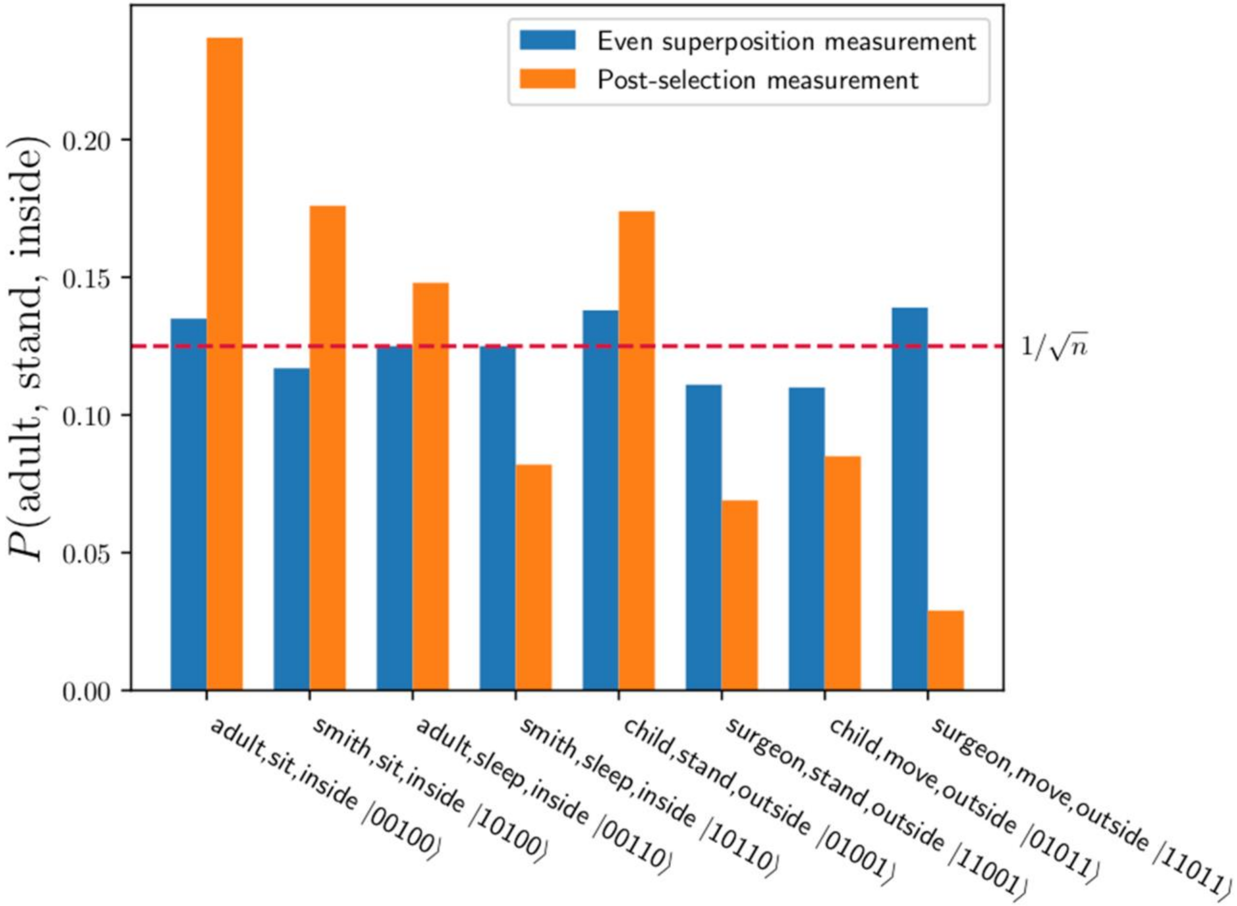
$\|000\rangle \rightarrow$	John eats water	$\|001\rangle \rightarrow$	John eats cake
$\|010\rangle \rightarrow$	John swims water	$\|011\rangle \rightarrow$	John swims cake
$\|100\rangle \rightarrow$	Mary eats water	$\|101\rangle \rightarrow$	Mary eats cake
$\|110\rangle \rightarrow$	Mary swims water	$\|111\rangle \rightarrow$	Mary swims cake

$$\begin{pmatrix} \text{John} \\ \text{Mary} \end{pmatrix} \otimes \begin{pmatrix} \text{eats} \\ \text{swims} \end{pmatrix} \otimes \begin{pmatrix} \text{water} \\ \text{cake} \end{pmatrix}$$

RESULTS



Even superposition measurement
Post-selection measurement
$P(\text{smith, stand, outside})$
0.00
0.05
0.10
0.15
0.20
$1/\sqrt{n}$
adult,sit,inside $|00100\rangle$
smith,sit,inside $|10100\rangle$
adult,sleep,inside $|00110\rangle$
smith,sleep,inside $|10110\rangle$
child,stand,outside $|01001\rangle$
surgeon,stand,outside $|11001\rangle$
child,move,outside $|01011\rangle$
surgeon,move,outside $|11011\rangle$



Even superposition measurement
Post-selection measurement
$P(\text{adult, stand, inside})$
0.00
0.05
0.10
0.15
0.20
$1/\sqrt{n}$
adult,sit,inside $|00100\rangle$
smith,sit,inside $|10100\rangle$
adult,sleep,inside $|00110\rangle$
smith,sleep,inside $|10110\rangle$
child,stand,outside $|01001\rangle$
surgeon,stand,outside $|11001\rangle$
child,move,outside $|01011\rangle$
surgeon,move,outside $|11011\rangle$

CONCLUSIONS

https://github.com/intel/intel-QS

- Having a commercial system is not enough, knowledge of ***new quantum algorithms*** and applications is required
- Prototyping and testing new Quantum Algorithms on HPC system is beneficial for learning the new field -> Intel Quantum Simulator
- The Natural Language Processing method is a good candidate prototype application to be run on a quantum computer
- We collaborate with different research partners to explore the potential of new algorithms for quantum computing

ICHEC Irish Centre for High-End Computing

CERN openlab

lrz Leibniz-Rechenzentrum der Bayerischen Akademie der Wissenschaften experience what's inside™

THANK YOU!

experience
what's inside™