800 ANNI 1222 · 2022

UNIVERSITÀ DEGLI STUDI DI PADOVA

Quantum Computing for Information Access

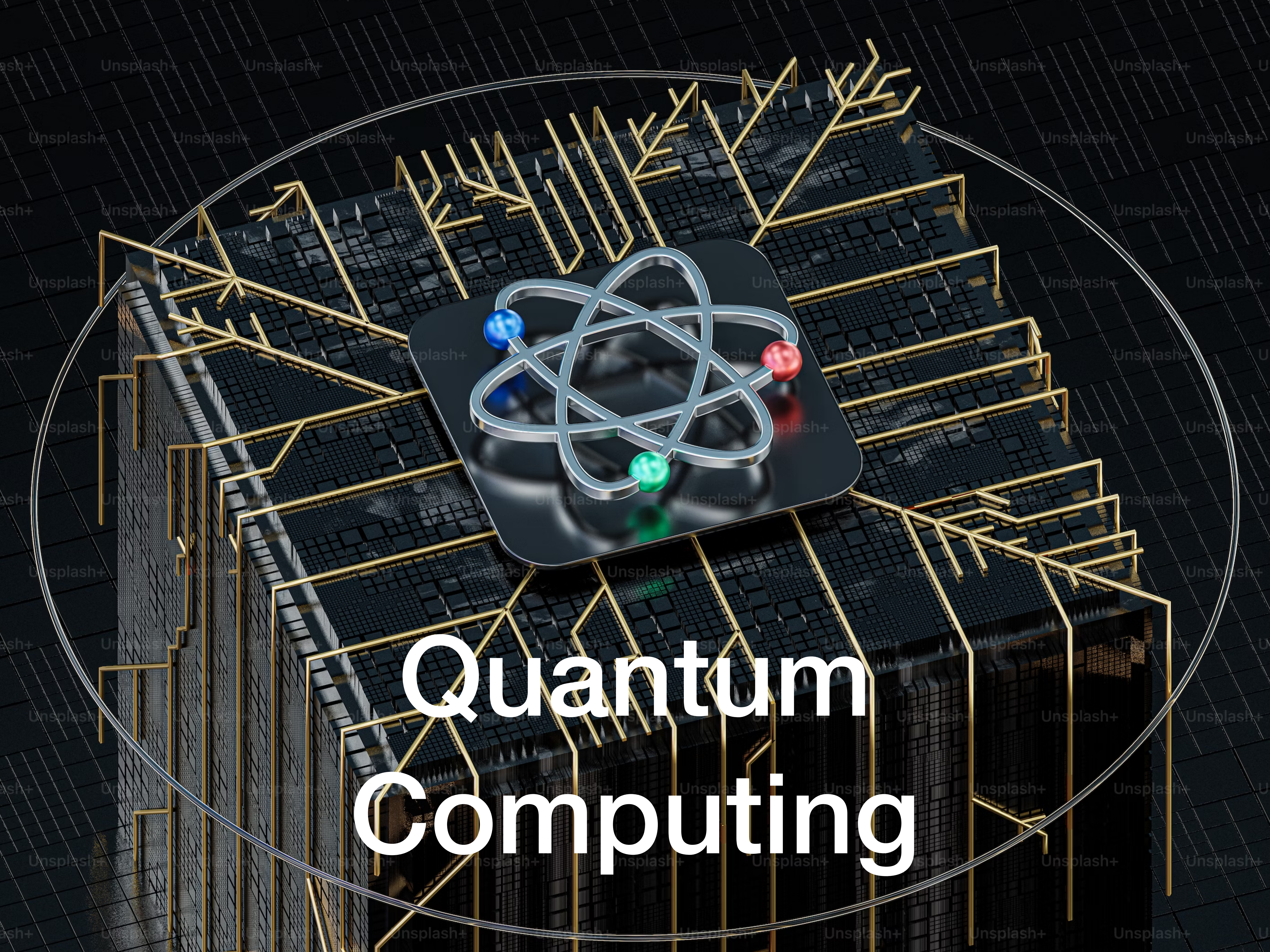
Search Engines

Master Degree in Computer Engineering
Master Degree in Data Science
Academic Year 2023/2024

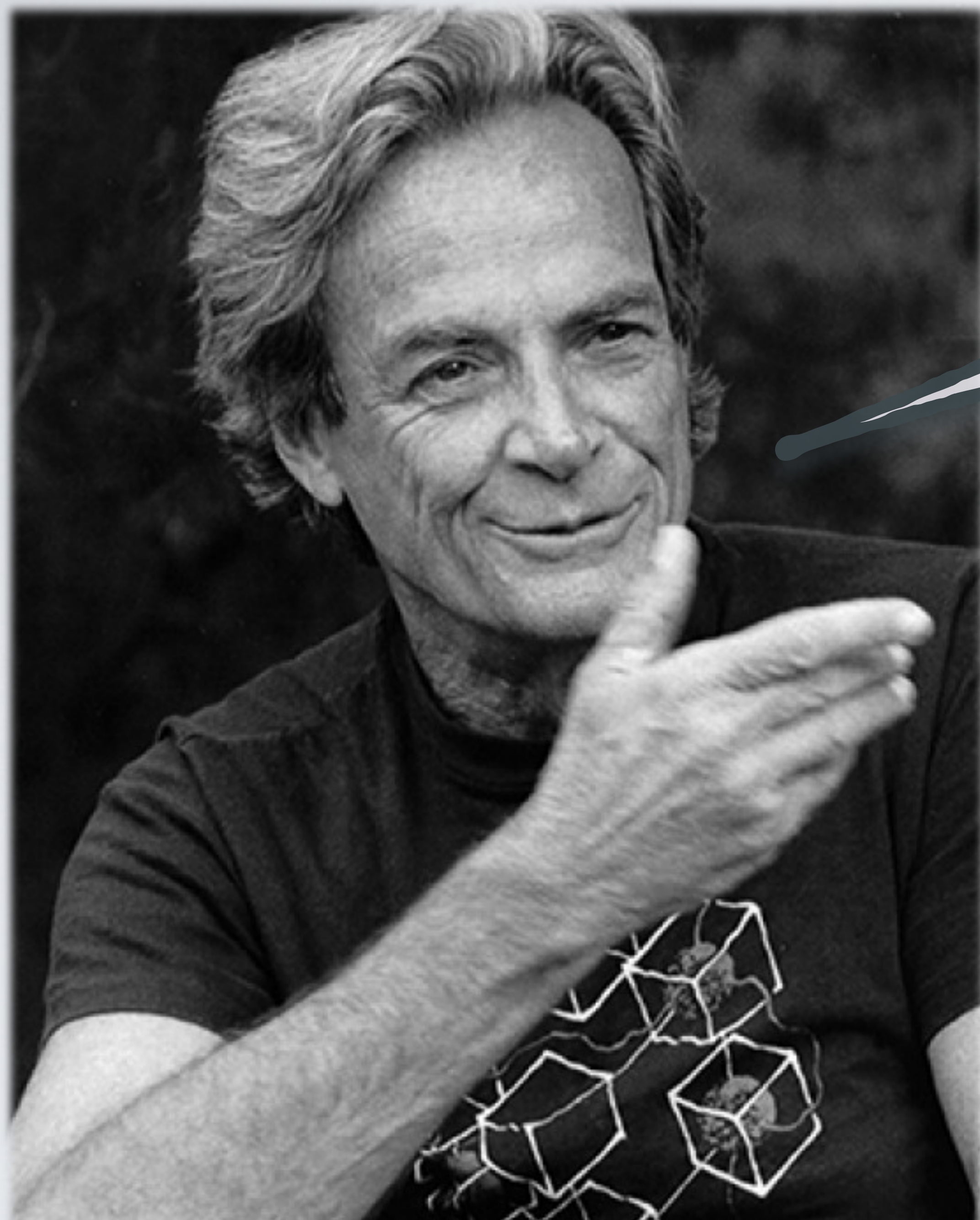
Nicola Ferro

Intelligent Interactive Information Access (IIIA) Hub
Department of Information Engineering
University of Padua

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE


Quantum
Computing

Quantum Computing



What kind of computer are we going to use to simulate physics?

- Quantum computers have the potential to solve tasks that classical computers can never solve in a practical amount of time
  - This promises to dramatically reduce both execution time and energy consumption
- In 2019, Google announced to have achieve **quantum supremacy**
  - Our Sycamore processor (**53 qbits**) takes about **200 seconds** to sample one instance of a quantum circuit a million times—our benchmarks currently indicate that the equivalent task for a state-of-the-art classical supercomputer would take **approximately 10,000 years**.

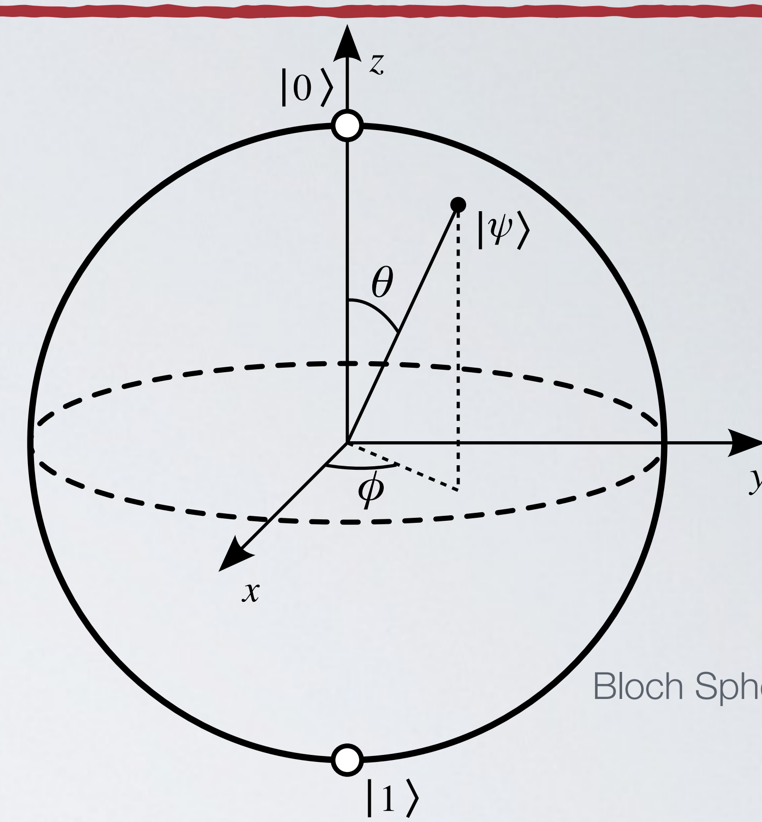
Feynman, R. P. (1982). Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6/7):467–488.
Arute, F., Arya, K., and Babbush, R. et al. (2019). Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505– 510.

Qubit: Welcome to the realm of probability

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

$$|\alpha|^2 + |\beta|^2 = 1$$
$$\alpha, \beta \in \mathbb{C}$$

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

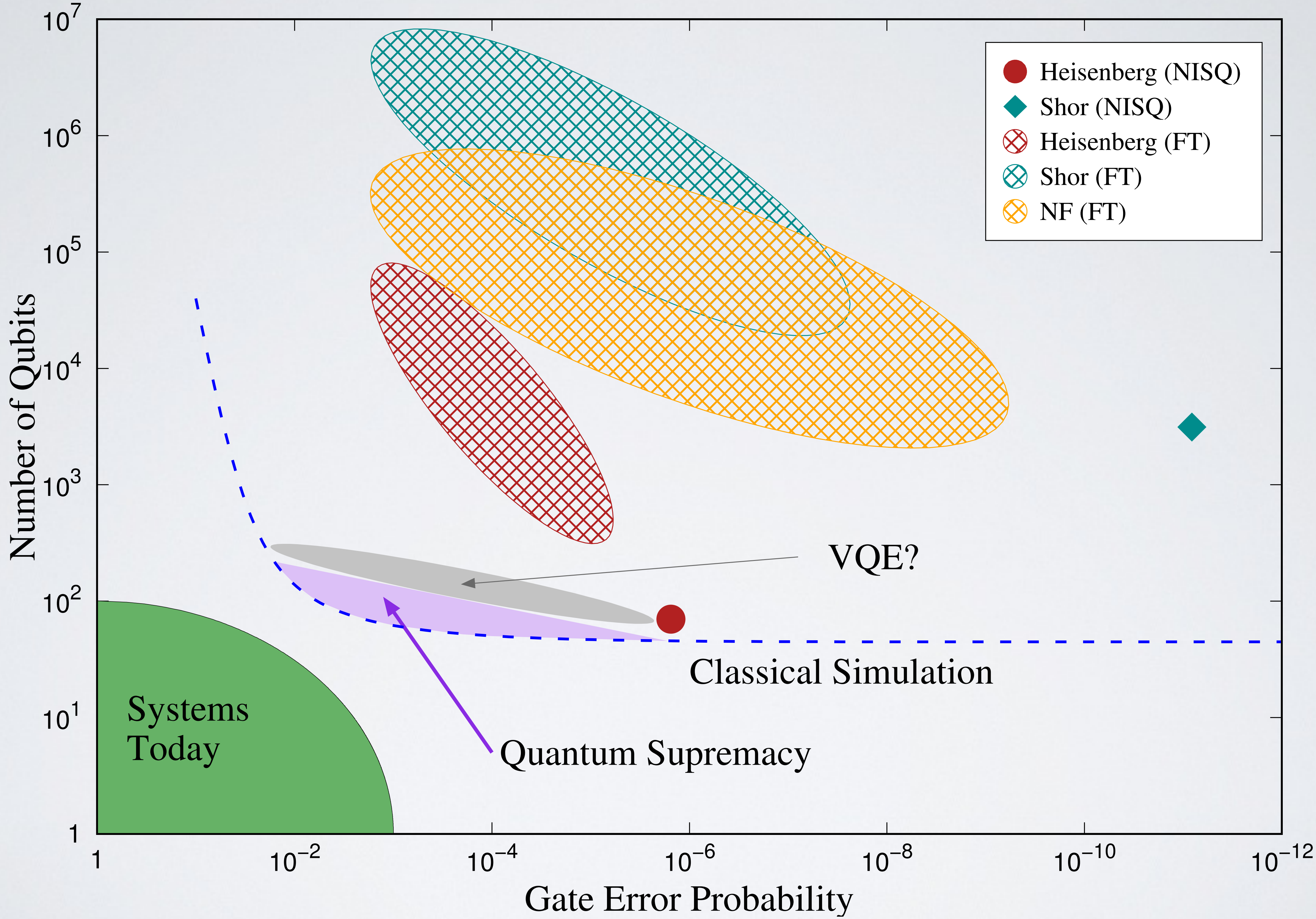


Bloch Sphere

- Quantum computing basically deals with the manipulation of **quantum systems**
  - The **state** of any quantum system is always represented by a **vector in a complex vector space**
  - Quantum algorithms are expressible as **transformations** acting on this vector space
- A **qubit** $|\psi\rangle$ is a two dimensional quantum system
  - The state of the qubit is the two dimensional complex vector $\begin{pmatrix} \alpha \\ \beta \end{pmatrix}$
  - **Measuring** a qubit (**reading out** a classical bit) will yield the classical value of either zero $|0\rangle$ with **probability** $|\alpha|^2$ or one $|1\rangle$ with **probability** $|\beta|^2$
- **Errors** happen due to decoherence, noise, …. and the bigger the system the more challenging

J., A., Adedoyin, A., Ambrosiano, J. et al. (2022). Quantum Algorithm Implementations for Beginners. *ACM Transactions on Quantum Computing (TQC)*, 3(4):18:1– 18:92.

Errors, Errors, Errors…



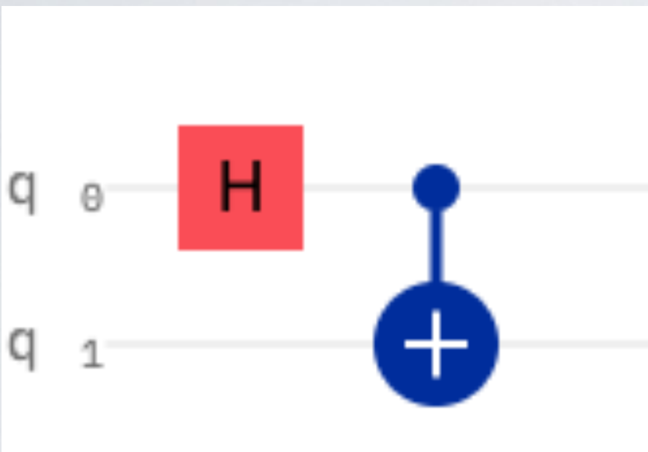
Maslov, D., Nam, Y., and Kim, J. (2019). An Outlook for Quantum Computing [Point of View]. *Proceedings of the IEEE*, 107(1):5–10

Universal Quantum Computing: The Gate Model

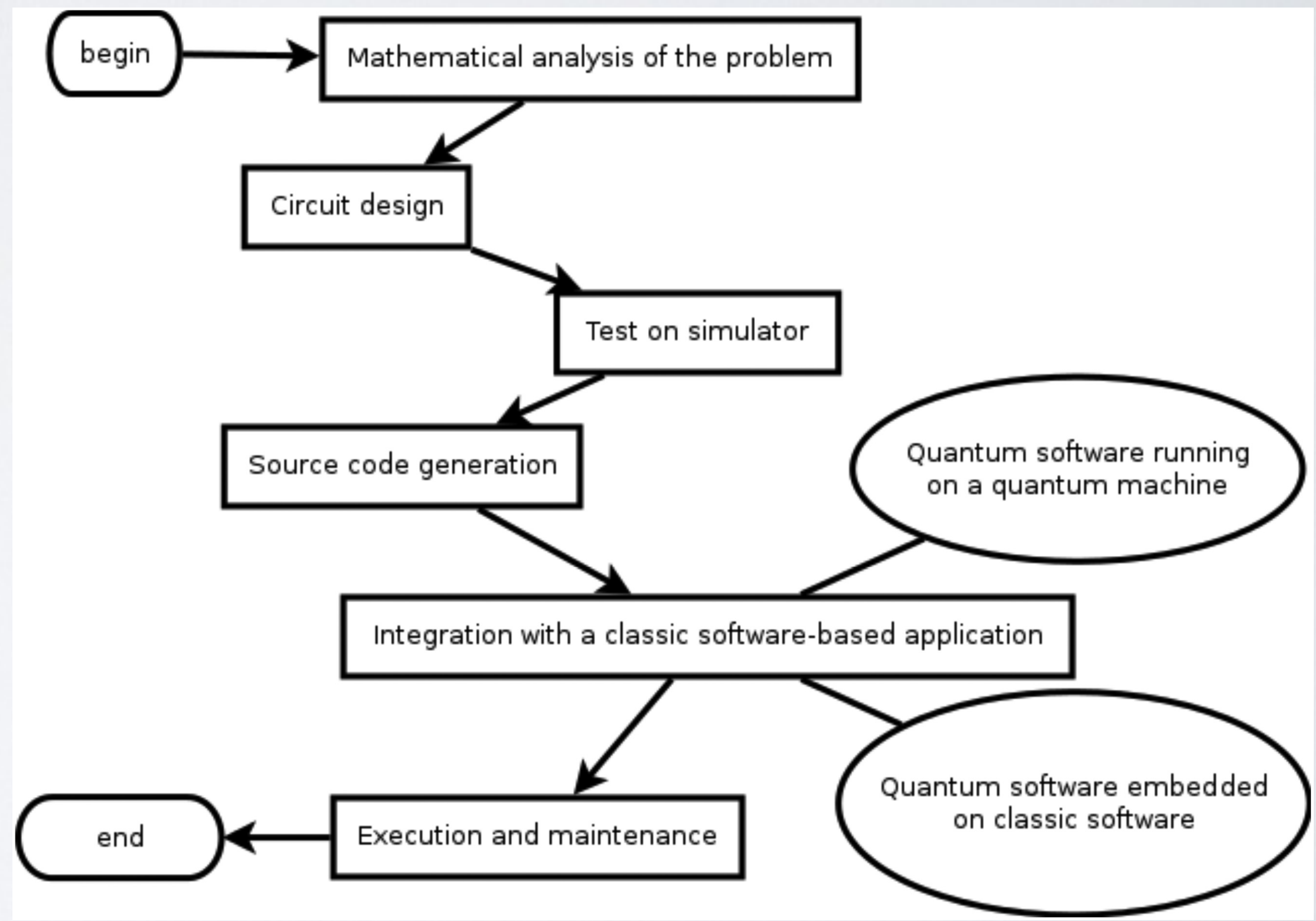
$$|\psi\rangle = \alpha\,|00\rangle + \beta\,|01\rangle + \gamma\,|10\rangle + \delta\,|11\rangle$$

$$|\alpha|^2 + |\beta|^2 + |\gamma|^2 + |\delta|^2 = 1$$

$$\Rightarrow \beta = 0, \gamma = 0$$




```
OPENQASM 2.0;
include "qelib1.inc";

qreg q[3];
creg c[3];

h q[0];
cx q[0],q[1];
```



Serrano, M. A., Cruz-Lemus, J. A., Perez-Castillo, R., and Piattini, M. (2023). Quantum Software Components and Platforms: Overview and Quality Assessment. *ACM Computing Surveys (CSUR)*, 55(8):164:1– 164:31.

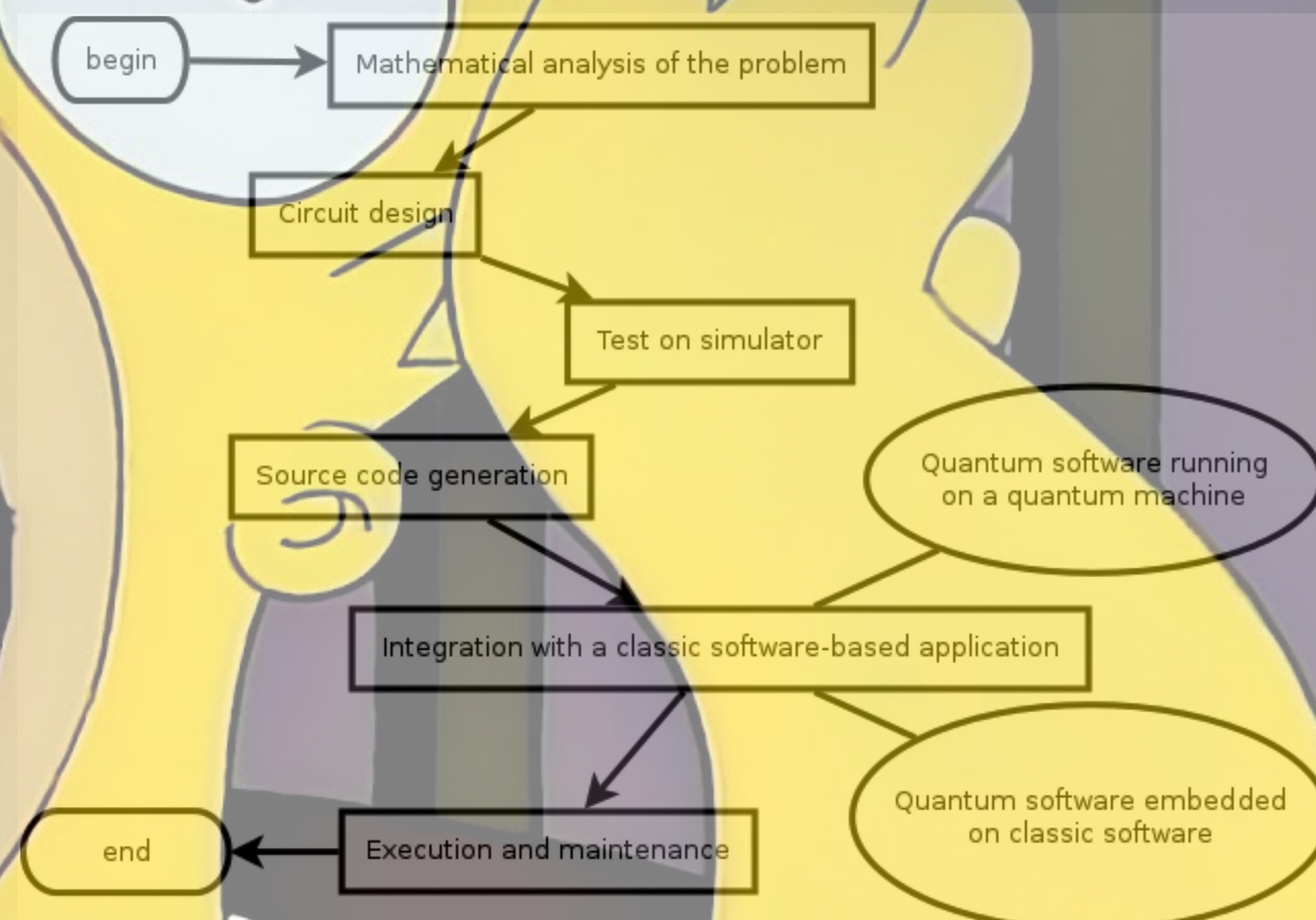
Universal Quantum Computing: The Gate Model

$$|\psi\rangle = \alpha\,|00\rangle + \beta\,|01\rangle + \gamma\,|10\rangle + \delta\,|11\rangle$$

$$|\alpha|^2 + |\beta|^2 + |\gamma|^2 + |\delta|^2 = 1$$

$$\Rightarrow \beta = 0, \gamma = 0$$


```
OPENQASM 2.0;
include "qelib1.inc";

qreg q[3];
creg c[3];

h q[0];
cx q[0],q[1];
```



Serrano, M. A., Cruz-Lemus, J. A., Perez-Castillo, R., and Piattini, M. (2023). Quantum Software Components and Platforms: Overview and Quality Assessment. *ACM Computing Surveys (CSUR)*, 55(8):164:1– 164:31.

Adiabatic Quantum Computing for Optimization

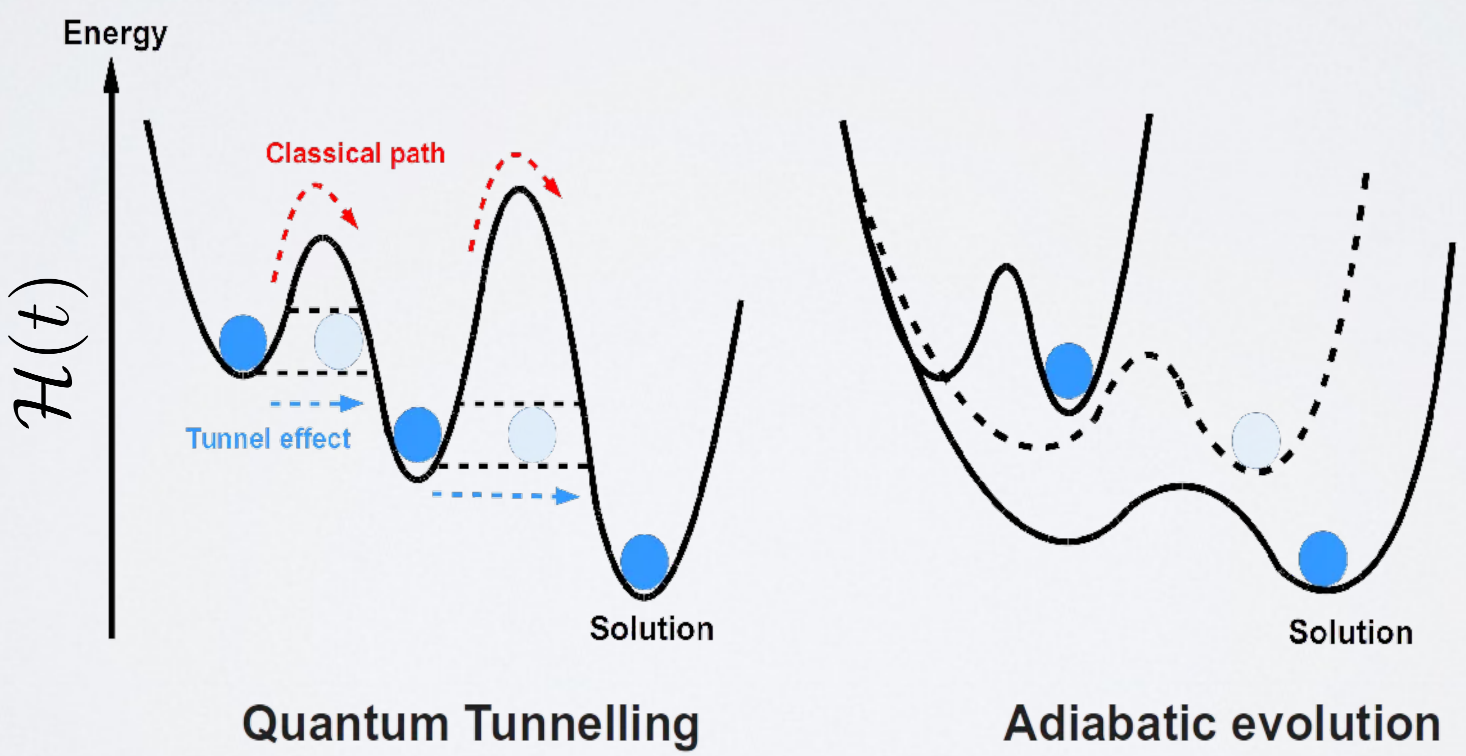
Adiabatic Theorem: a quantum system in its ground state will remain in the ground state, provided the Hamiltonian governing the dynamics changes sufficiently slowly

$$\mathcal{H}(t) = A(t)\mathcal{H}_i + B(t)\mathcal{H}_f$$
$$|\psi(t=0)\rangle \longrightarrow |\psi(t=T_a)\rangle$$

- "Sufficiently slowly" is dictated by the smallest energy difference between the **ground state** (**minimum energy**) and the first excited state exhibited during the system's evolution, also known as the **minimum gap**
- The adiabatic problem can be utilized for computation by preparing a system in the ground state of an easy to solve initial Hamiltonian $\mathcal{H}_i$ and then switching this Hamiltonian sufficiently slow to a complex Hamiltonian $\mathcal{H}_f$, which represents our **combinatorial optimization problem**
  - The system will remain in the ground state and consequently one ends up with the ground state of the complicated Hamiltonian $\mathcal{H}_f$, i.e. the solution to our combinatorial optimization problem

Quantum Annealing

- Adiabatic Quantum Computing is very difficult to obtain in practice due to the challenge of keeping the system isolated (adiabatic) and knowing T_a, i.e. what “sufficiently slowly”
- **Quantum Annealing (QA)** is a relaxation of Adiabatic Quantum Computing where T_a is determined heuristically and allows for **many more qubits** than the gate model
- Quantum **superposition**, **tunneling**, and **entanglement** allow for a direct transition between states even if there is a high energy barrier between them
  - the search algorithm can escape local minima by tunneling through energetic barriers

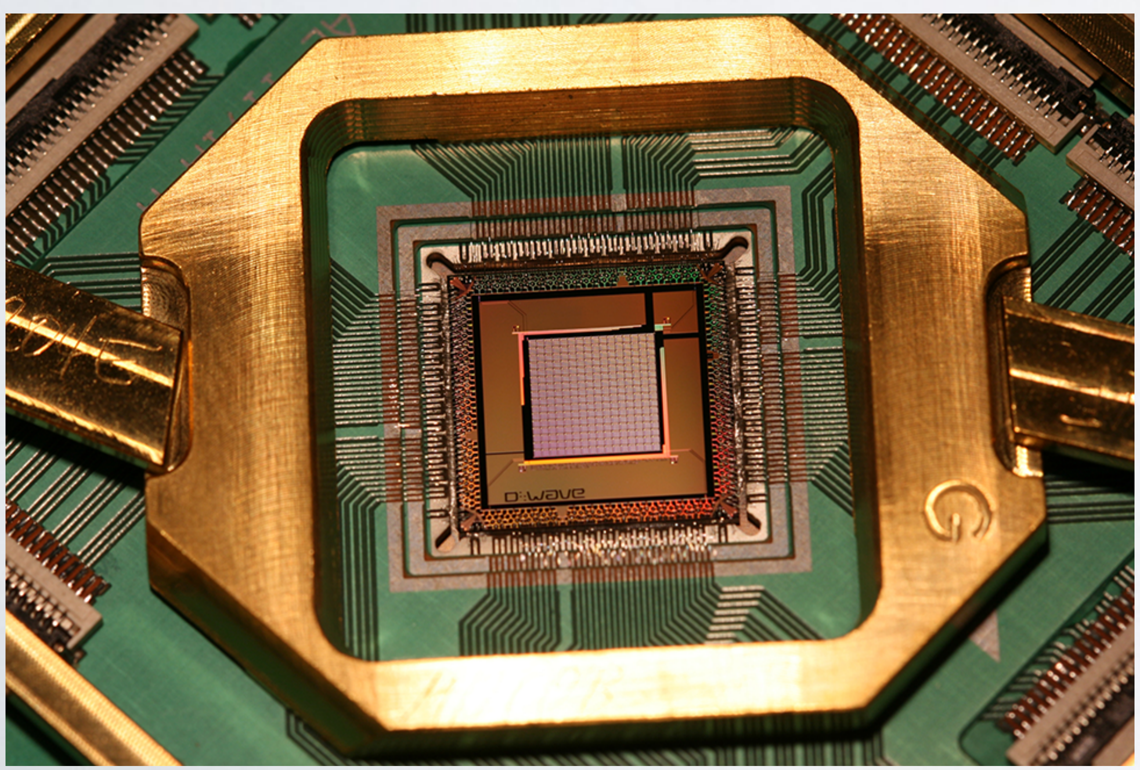


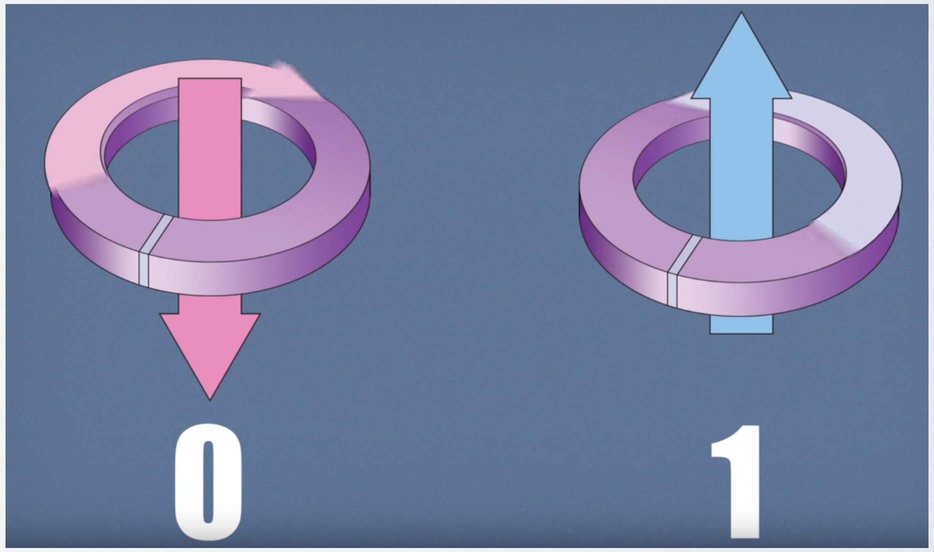
Apolloni, B., Carvalho, C., and de Falco, D. (1989). Quantum stochastic optimization. *Stochastic Processes and their Applications*, 33(2):233–244.

Apolloni, B., Cesa-Bianchi, N., and de Falco, D. (1990). A numerical implementation of quantum annealing. In Albeverio, S., Casati, G., Cattaneo, U., Merlini, D., and Moresi, R., editors, *Proc. International Conference on Stochastic Processes, Physics and Geometry*, pages 97–111. World Scientific Publishing, Singapore.

D-Wave

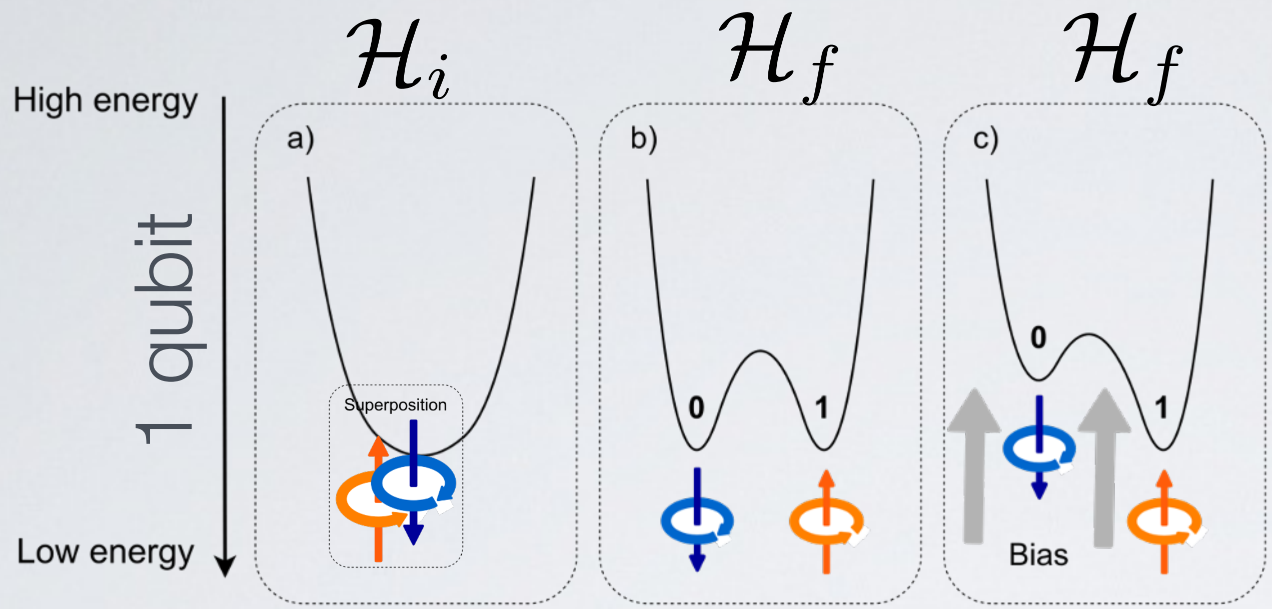
- The D-Wave **QPU (Quantum Processing Unit)** consists of **5,000 qubits** and **35,000 couplers** implemented as a lattice of tiny metal loops
  - Below temperatures of 9.2 kelvin, these loops become superconductors and exhibit quantum-mechanical effects
- Qubits are the lowest energy states of these superconducting loops, each of them having a circulating current — shown clockwise for 0 and counter clockwise for 1 — and a corresponding magnetic field

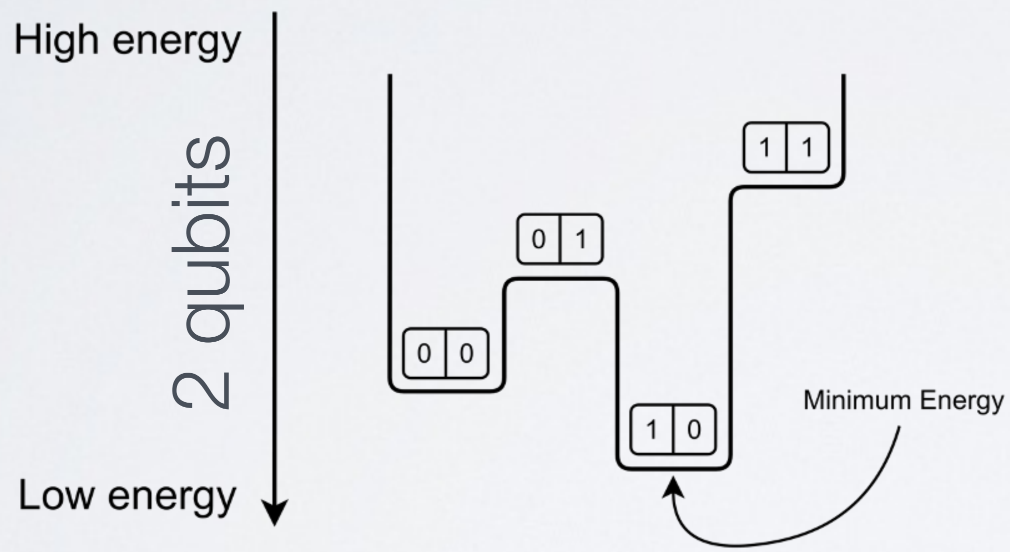




https://www.dwavesys.com/

Quantum Annealing in D-Wave





- The quantum annealing process runs, the barrier is raised, and this turns the energy diagram into what is known as a double-well potential
  - the low point of the left valley corresponds to the 0 state, and the low point of the right valley corresponds to the 1 state
  - the probability of the qubit ending in the 0 or the 1 state is equal
- The **bias**, obtained by applying an external magnetic field to the qubit, increases the probability of the qubit ending up in the lower well
- When two qubits are **entangled**, via a coupler, they can be thought of as a **single object** with **four possible states**
  - the correlation weights between coupled qubits can be programmed by setting a **coupling** strength
  - The relative energy of each state depends on the biases of qubits and the coupling between them

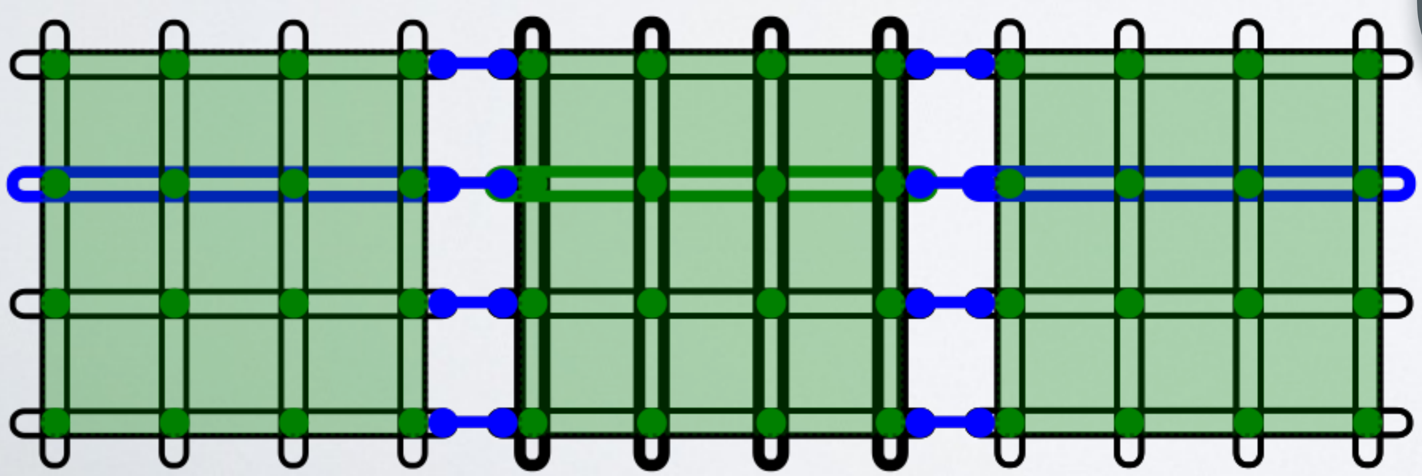
When formulating a problem, users choose values for the biases and couplers. The biases and couplings define an energy landscape, and the D-Wave quantum computer finds the minimum energy of that landscape

Quantum Annealing in D-Wave

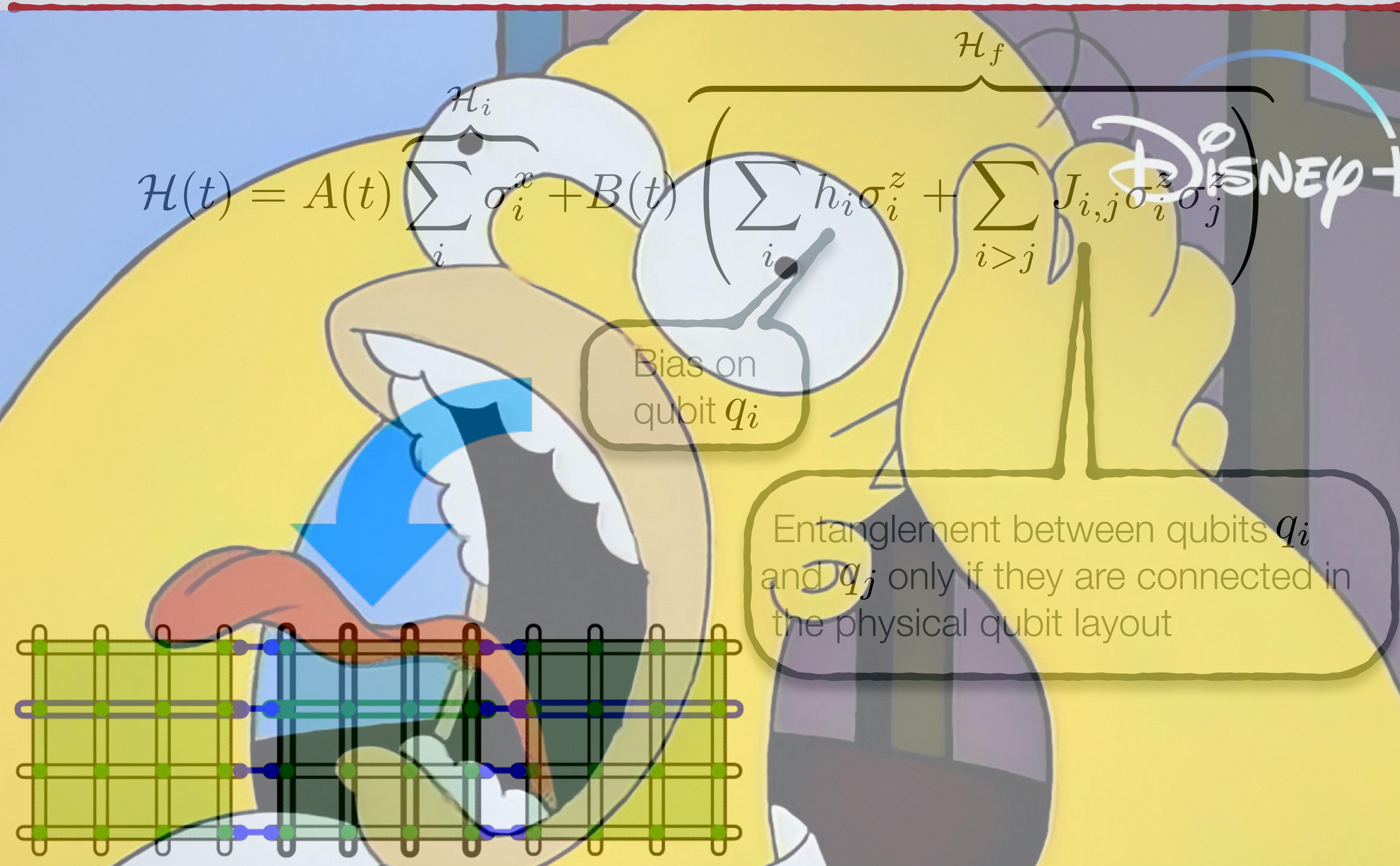
$$\mathcal{H}(t) = A(t)\overbrace{\sum_i \sigma_i^x}^{\mathcal{H}_i} + B(t)\overbrace{\left(\sum_i h_i\sigma_i^z + \sum_{i>j} J_{i,j}\sigma_i^z\sigma_j^z\right)}^{\mathcal{H}_f}$$

Bias on qubit q_i

Entanglement between qubits q_i and q_j only if they are connected in the physical qubit layout



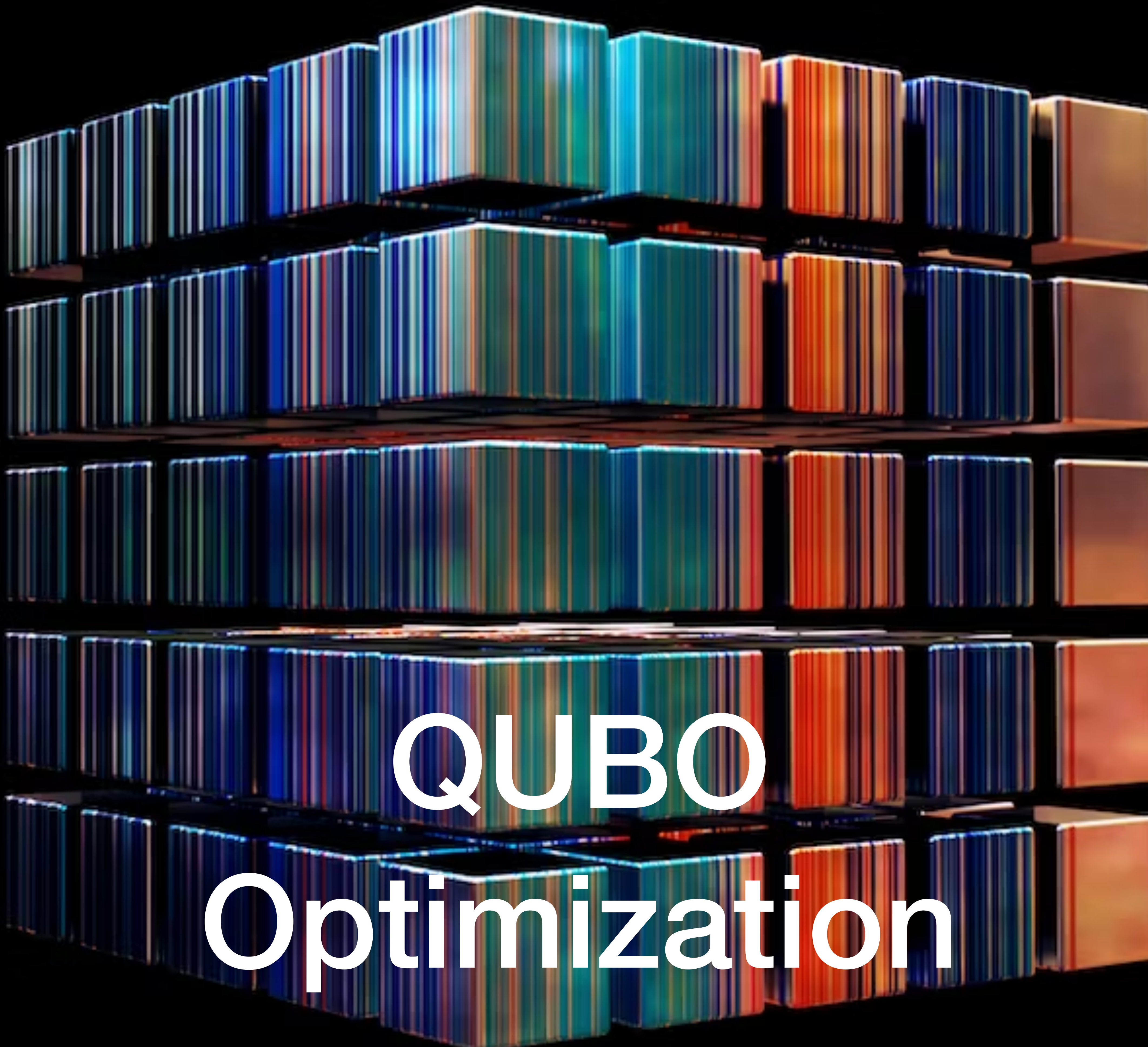
Quantum Annealing in D-Wave



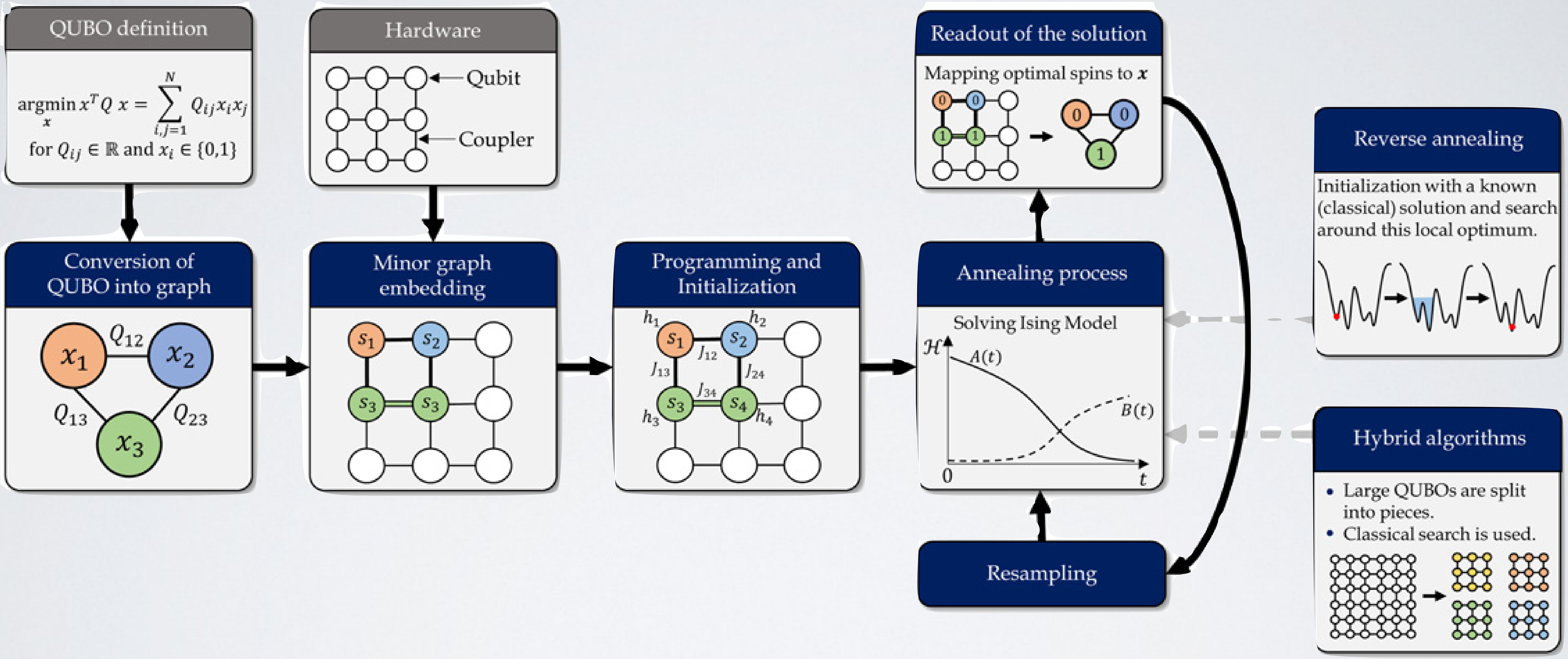
$$\mathcal{H}(t) = A(t)\overbrace{\sum_i \sigma_i^x}^{\mathcal{H}_i} + B(t)\overbrace{\left(\sum_i h_i \sigma_i^z + \sum_{i>j} J_{i,j}\sigma_i^z\sigma_j^z\right)}^{\mathcal{H}_f}$$

Bias on qubit q_i

Entanglement between qubits q_i and q_j only if they are connected in the physical qubit layout

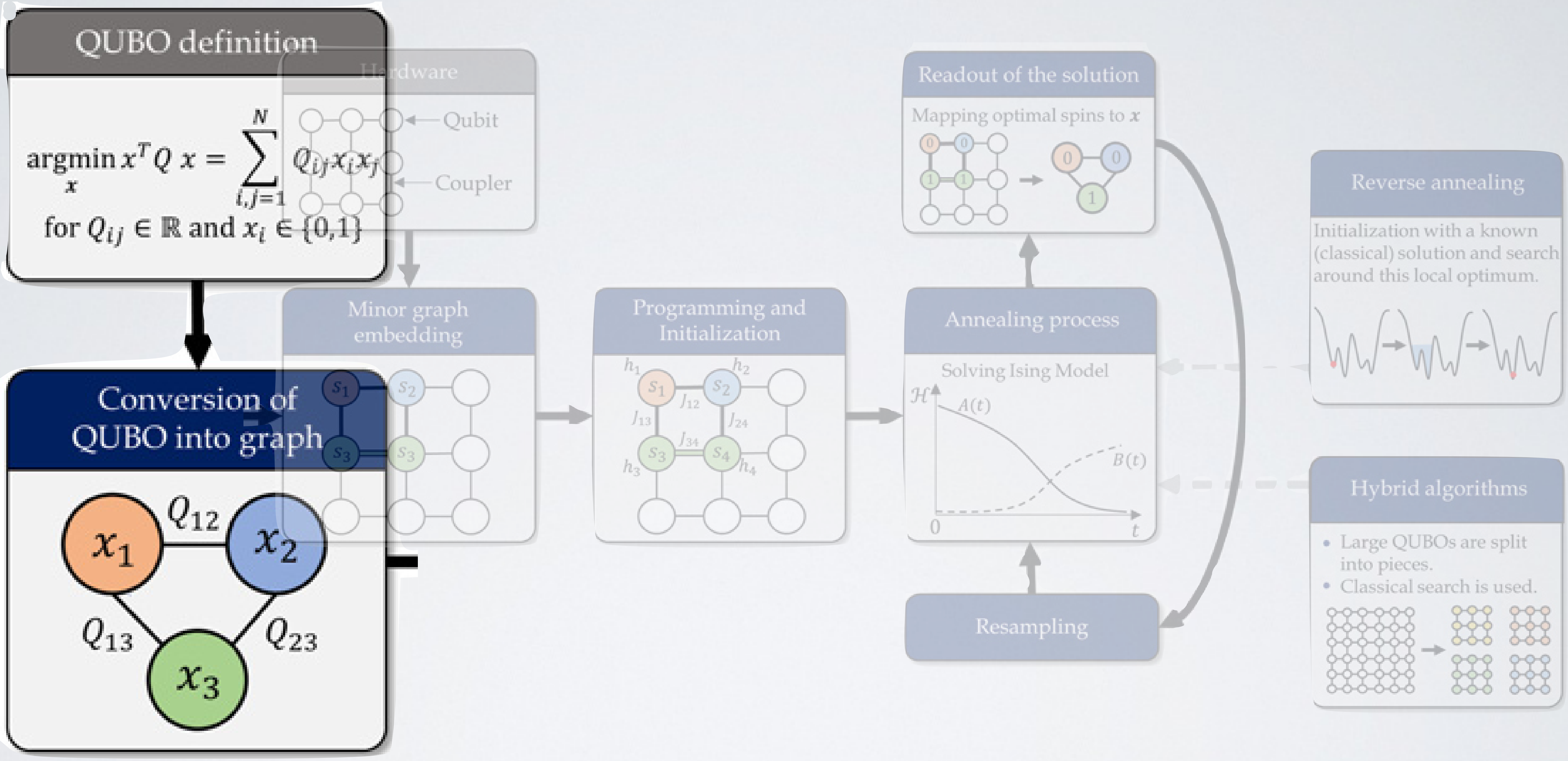

QUBO
Optimization

Quantum Annealing: QUBO at the Rescue



Yarkoni, S., Raponi, E., Bäck, T., and Schmitt, S. (2022). Quantum annealing for industry applications: introduction and review. *Reports on Progress in Physics*, 85(10):104001:1–104001:27.

Quantum Annealing: QUBO at the Rescue



Yarkoni, S., Raponi, E., Bäck, T., and Schmitt, S. (2022). Quantum annealing for industry applications: introduction and review. *Reports on Progress in Physics*, 85(10):104001:1–104001:27.

QUBO: Quadratic Unconstrained Binary Optimization

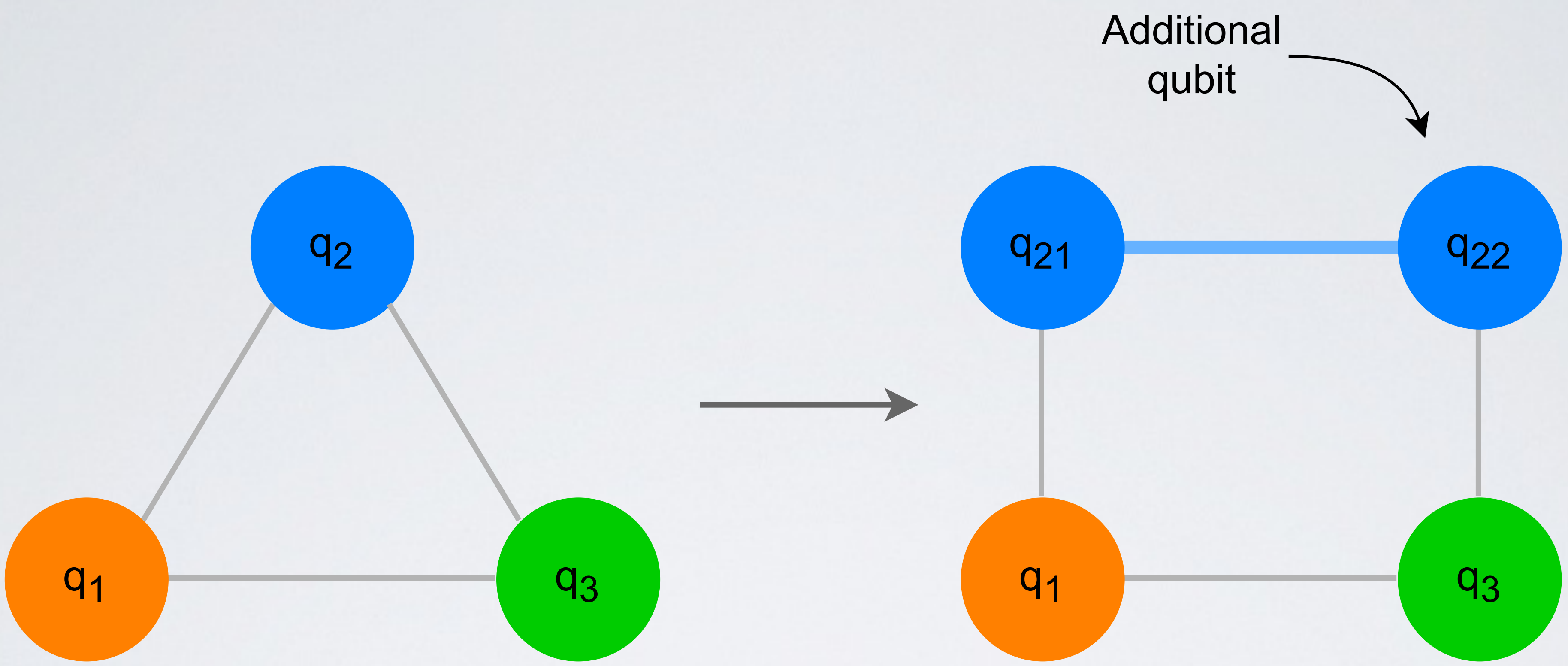
$$\arg\min_{x} x^T Q x$$

- x is a vector of binary decision variables
  - this corresponds to our qubits
- Q is a square matrix of constants
  - this corresponds to our $\mathcal{H}_f$
- Additional constraint can be specified via penalties added to $x^T Q x$

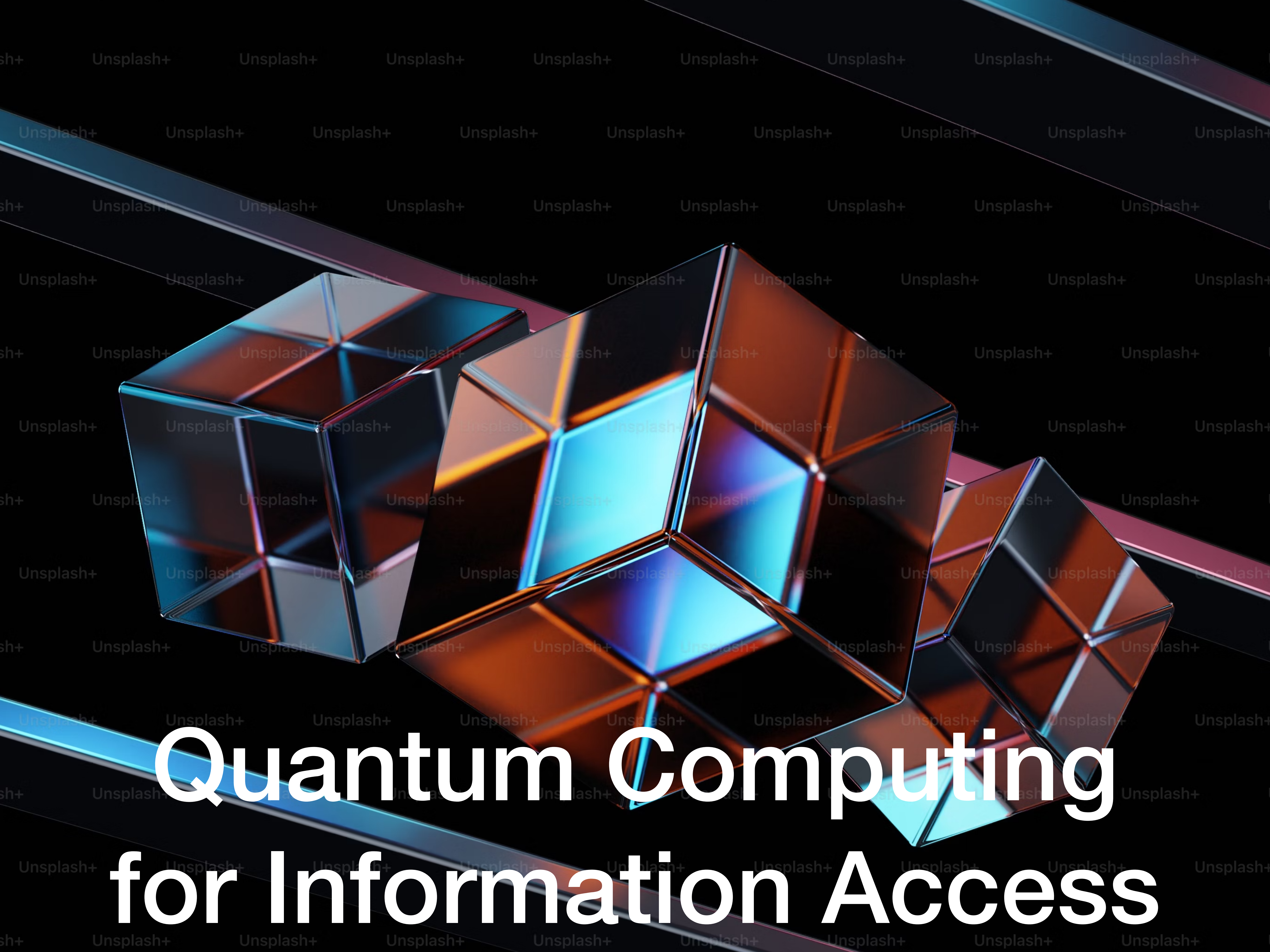
Classical constraint	Equivalent penalty
$x + y \leq 1$	$P(xy)$
$x + y \geq 1$	$P(1 - x - y + xy)$
$x + y = 1$	$P(1 - x - y + 2xy)$
$x \leq y$	$P(x - xy)$
$x_1 + x_2 + x_3 \leq 1$	$P(x_1 x_2 + x_1 x_3 + x_2 x_3)$
$x = y$	$P(x + y - 2xy)$

Glover, F., Kochenberger, G., Hennig, R., and Du, Y. (2022). Quantum bridge analytics I: a tutorial on formulating and using QUBO models. *Annals of Operations Research*, 314:141–183.

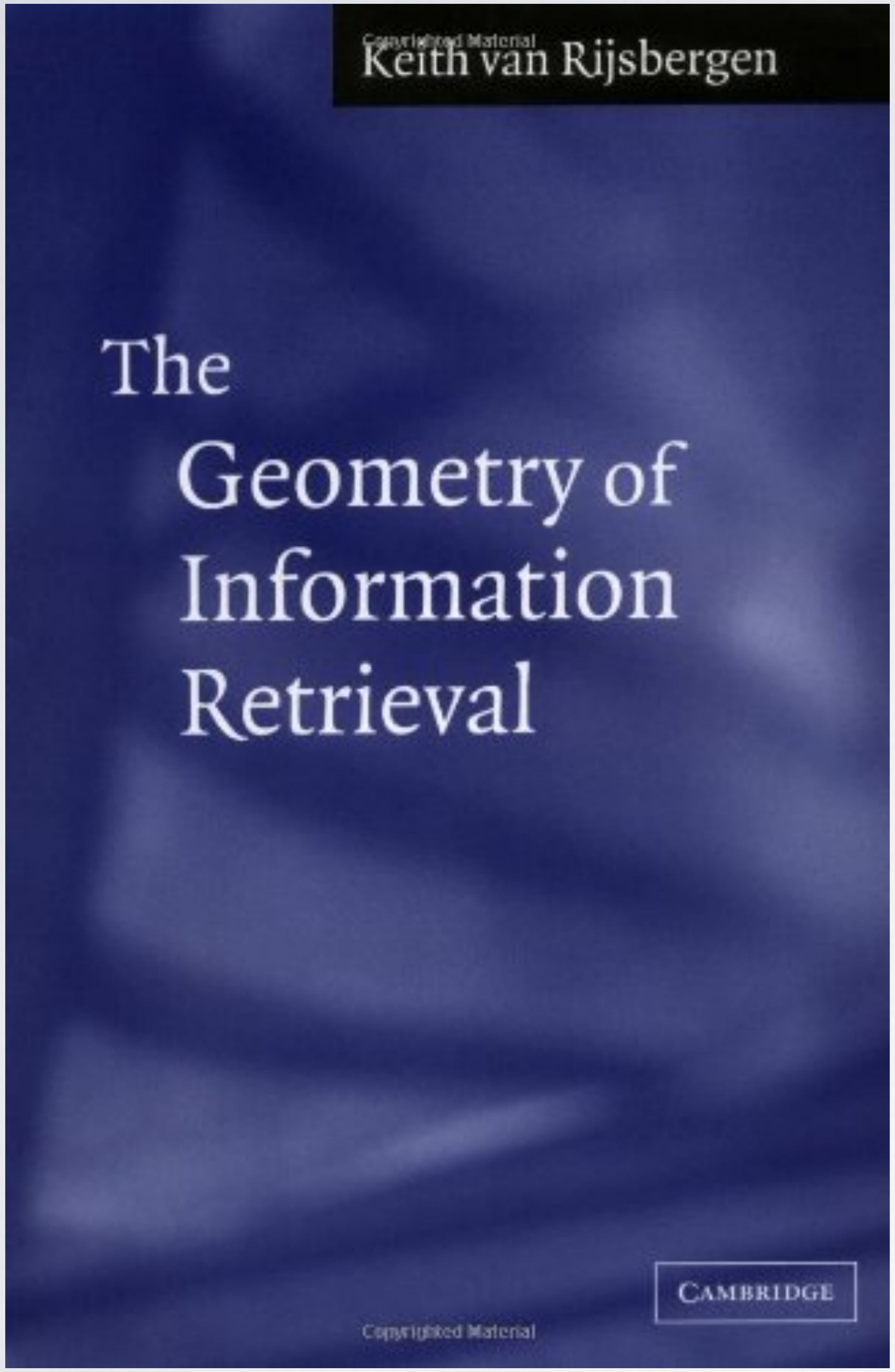
QUBO: Minor Embedding



- Suppose to have a QUBO problem with a triangle topology but the QPU uses a square topology
- We need to convert it to a square topology by adding an auxiliary variable: we add a new node (qubit) to our triangular graph and a new edge connecting the new node to its corresponding old one to represent the same variable but according to the square topology
    - in the final results they will hold the same values and the newly added one can be simply ignore
- This **minor embedding** is done by the D-Wave infrastructure using heuristic algorithms
    - for the current Pegasus topology with 5,000 qubits, the **maximum number of variables** that can be embedded in a fully connected problem is approximately **180**


Quantum Computing
for Information Access

It is Not Quantum Information Retrieval



- Quantum IR stemmed from the seminal book by C. J. “Keith” van Rijsbergen in 2004
- Quantum IR is mostly concerned with using the theory, formalism, and operators of quantum physics and exploiting them to model
  - representation and ranking
  - user interaction
- For an updated account, please, see Busemeyer and Bruza (2012); Melucci (2015); and, Uprety et al. (2021)

van Rijsbergen, C. J. (2004). *The Geometry of Information Retrieval*. Cambridge University Press, Cambridge, UK.

Busemeyer, J. R. and Bruza, P. D. (2012). *Quantum Models of Cognition and Decision*. Cambridge University Press, Cambridge, UK.

Melucci, M. (2015). *Introduction to Information Retrieval and Quantum Mechanics*, volume 35 of The Information Retrieval Series. Springer.

Uprety, S., Gkoumas, D., and Song, D. (2021). A Survey of Quantum Theory Inspired Approaches to Information Retrieval. *ACM Computing Surveys (CSUR)*, 53(5):98:1–98:39.

Quantum Annealing for PageRank

PRL **108,** 230506 (2012) PHYSICAL REVIEW LETTERS week ending 8 JUNE 2012

Adiabatic Quantum Algorithm for Search Engine Ranking

Silvano Garnerone,[1,2,5] Paolo Zanardi,[2,5] and Daniel A. Lidar[2,3,4,5]

[1]*Institute for Quantum Computing, University of Waterloo, Waterloo, ON N2L 3G1, Canada*
[2]*Department of Physics & Astronomy, University of Southern California, Los Angeles, California 90089, USA*
[3]*Department of Electrical Engineering, University of Southern California, Los Angeles, California 90089, USA*
[4]*Department of Chemistry, University of Southern California, Los Angeles, California 90089, USA*
[5]*Center for Quantum Information Science & Technology, University of Southern California, Los Angeles, California 90089, USA*

(Received 25 October 2011; published 4 June 2012)

We propose an adiabatic quantum algorithm for generating a quantum pure state encoding of the PageRank vector, the most widely used tool in ranking the relative importance of internet pages. We present extensive numerical simulations which provide evidence that this algorithm can prepare the quantum PageRank state in a time which, on average, scales polylogarithmically in the number of web pages. We argue that the main topological feature of the underlying web graph allowing for such a scaling is the out-degree distribution. The top-ranked $\log(n)$ entries of the quantum PageRank state can then be estimated with a polynomial quantum speed-up. Moreover, the quantum PageRank state can be used in "q-sampling" protocols for testing properties of distributions, which require exponentially fewer measurements than all classical schemes designed for the same task. This can be used to decide whether to run a classical update of the PageRank.

DOI: 10.1103/PhysRevLett.108.230506 PACS numbers: 03.67.Ac, 03.67.Lx, 89.20.Hh

Introduction.—Quantum mechanics provides computational resources that can be used to outperfom classical algorithms [1]. Problems for which a polynomial or exponential quantum speed-up is achievable have been sought in quantum computation since its inception, and their ranks are swelling slowly [2]. Yet, while ranking the results obtained in response to a user query is one of the most difficult tasks in searching the web [3], so far no efficient quantum algorithms have been proposed for this task [4].

Here we present an adiabatic quantum algorithm [8] which prepares a state containing the same ranking information as the PageRank vector. The latter is a central tool in data mining and information retrieval, at the heart of the success of the Google search engine [3,9–12]. The best available classical algebraic and Markov Chain Monte Carlo (MCMC) techniques used to evaluate the full PageRank vector require a time which scales as $O(n)$ and $O[n\log(n)]$, respectively, where n is the number of pages, i.e., the size of the web graph. We investigate the size of the gap of the adiabatic Hamiltonian numerically using a wide range of web-graph sizes ($n \in \{2^2, \ldots, 2^{14}\}$), and present evidence that our quantum algorithm prepares the PageRank state in a time which scales on average as $O[\text{polylog}(n)]$. We argue that while extraction of the full PageRank vector cannot in general be done more efficiently than when using the aforementioned classical algorithms, there are particular graph-topologies and specific tasks of relevance in the use of search engines for which the quantum algorithm, combined with other known quantum protocols [13–16], may provide a polynomial, or even exponential speed-up. We discuss the underlying graph structure which we believe is responsible for this potential speed-up, and provide evidence that it is the power-law distribution of the out-degree nodes that plays the key role. A proof of this fact would be very interesting.

Model of the web-graph.—The PageRank algorithm, introduced by Brin & Page [9], is probably the most prominent ranking measure using the query-independent hyperlink structure of the web. The PageRank vector is the principal eigenvector of the so-called Google matrix, which encodes the structure of the web-graph via its adjacency matrix. The humongous size of the World Wide Web (WWW), with its ever growing number of pages and links, makes the evaluation of the PageRank vector one of the most demanding computational tasks ever [12]. In practice PageRank is evaluated over real data providing the structure of the actual WWW. On the other hand the use of models of the web-graph has proved to be useful in testing new ideas concerning structure measures and dynamical properties of the web [11]. To accurately capture the WWW graph a good candidate model network should be (i) sparse (the number of edges is proportional to the number of nodes), (ii) small-world (the network diameter scales logarithmically in the size of the network), and (iii) scale-free (the in- and out-degree probability distributions obey a power law). To analyze the scaling properties of our algorithm we used two well known models of the web graph: the preferential attachment model [17], and the copying model [18]. These models are based on two different network evolution mechanisms, both of which yield sparse random graphs with small-world and scale-free (power-law) features.

We implemented a version [19] of the preferential attachment model that provides a scale-free network with $N(d) \propto d^{-3}$, where $N(d)$ is the number of nodes of degree d.

0031-9007/12/108(23)/230506(6) 230506-1

$$O\left(\frac{1}{\varepsilon^2}(\log\log n)^{b-1}(\log n)^b\right)$$

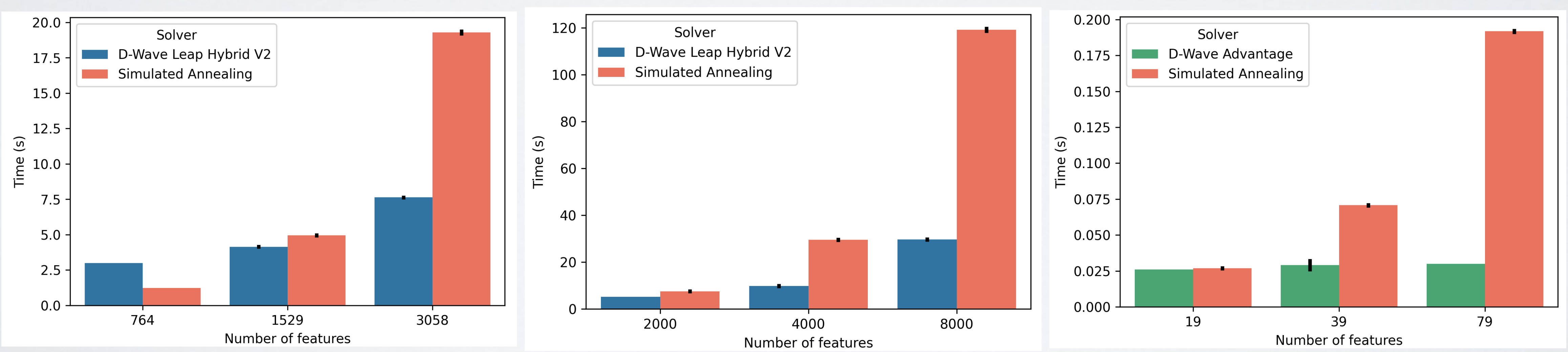
- Simulated results suggest speed-up with respect to classical implementation
  - used 10,000 nodes

Garnerone, S., Zanardi, P., and Lidar, D. A. (2012). Adiabatic Quantum Algorithm for Search Engine Ranking. *Physical Review Letters*, 108(23):230506:1–230506:6.

QA for Feature Selection in RecSys

Models	Precision	Recall	NDCG	MAP	I. Cov.	Gini
ItemKNN CBF	0.1218	0.0795	0.0856	0.0808	0.6637	0.0578
TFIDF 40%	0.0503	0.0177	0.0269	0.0337	0.1628	0.0146
TFIDF 60%	0.0793	0.0424	0.0595	0.0541	0.2251	0.0222
TFIDF 80%	0.0838	0.0477	0.0633	0.0565	0.3186	0.0381
TFIDF 95%	0.0914	0.0508	0.0665	0.0634	0.4510	0.0505
CFeCBF ItemKNN	0.0893	0.0574	0.0692	0.0598	0.5962	0.0451
CFeCBF PureSVD	0.0818	0.0547	0.0681	0.0577	0.4704	0.0309
CFeCBF RP$^3\beta$	0.0751	0.0481	0.0563	0.0499	0.6325	0.0472
CQFS trained with ItemKNN						
CQFS 20%	0.1124	0.0748	0.0788	0.0725	0.6552	0.0548
CQFS 30%	0.1163	0.0767	0.0819	0.0767	0.6558	0.0552
CQFS 40%	0.1208	0.0793	0.0853	0.0806	0.6486	0.0553
CQFS 60%	**0.1233**	**0.0802**	**0.0867**	**0.0828**	0.6531	0.0563
CQFS 80%	**0.1232**	**0.0802**	**0.0865**	**0.0824**	0.6564	0.0571
CQFS 95%	**0.1225**	**0.0798**	**0.0861**	**0.0814**	0.6631	0.0577

- Weights in the Q matrix are
  - proportional to the content-based similarity between items, only if there is collaborative similarity
  - opposite proportional to the content-based similarity if there is not collaborative similarity
  - zero otherwise



(**a**) The Movies Dataset (**b**) CiteULike-a (**c**) Xing Challenge 2017

Nembrini, R., Ferrari Dacrema, M., and Cremonesi, P. (2021). Feature Selection for Recommender Systems with Quantum Computing. *Entropy*, 23(8):970:1–970:22.

QA for Feature Selection in IR (and RecSys)

Method	Dataset	All Features F	All Features NDCG	Quantum Solver QPU N	QPU NDCG	Hybrid N	Hybrid NDCG	Traditional Solver SA N	SA NDCG	SD N	SD NDCG	TS N	TS NDCG
QUBO Correlation	OHSUMED	45	0.3882	35	0.3706	44	0.3659	44	**0.3903**	44	0.3872	26	0.3332
	MQ2007	46	0.4721	43	0.4731	39	0.4720	39	0.4738	28	0.4733	24	**0.4757**
	MQ2008	46	**0.4891**	4	0.4831	21	0.4677	21	0.4657	19	0.4854	25	0.4609
QUBO Boosting	OHSUMED	45	0.3882	9	0.3457	32	0.3632	19	0.3382	40	**0.4002**	23	0.3884
	MQ2007	46	0.4721	43	**0.4760**	37	0.4638	42	0.4632	36	0.4640	35	0.4662
	MQ2008	46	**0.4891**	8	0.4599	8	0.4736	20	0.4852	39	0.4759	34	0.4853
MIQUBO	OHSUMED	45	0.3882	11	0.3685	17	0.3750	17	**0.3942**	6	0.3755	4	0.3882
	MQ2007	46	0.4721	25	**0.4798**	34	0.4722	34	0.4685	34	0.4722	2	0.4721
	MQ2008	46	**0.4891**	1	0.4743	18	0.4791	18	0.4791	18	0.4791	18	**0.4891**

Dataset	F	QUBO-Correlation Quantum Solver QPU	QUBO-Correlation Quantum Solver Hybrid	QUBO-Correlation Traditional Solver SA	SD	TS	MIQUBO Quantum Solver QPU	MIQUBO Quantum Solver Hybrid	MIQUBO Traditional Solver SA	SD	TS
waveform-5000	40	39.0	394.4	57.9	7.8	474.6	39.2	385.9	7.5	1.0	78.6
SPECTF	44	48.4	426.3	72.3	9.3	522.4	45.1	421.7	9.5	1.2	86.7
covertype	54	130.1	491.5	17.0	1.9	99.1	127.3	809.9	17.0	1.9	99.0
spambase	57	65.9	487.4	107.1	15.7	598.9	51.5	486.4	17.6	2.0	99.1
nomao	118	100.1	507.5	323.6	54.3	620.9	98.1	512.5	46.6	7.9	101.9
tecator	124	95.6	504.3	374.2	57.7	623.5	90.2	515.1	62.3	8.8	102.3
USPS	256	-	541.1	1174.0	220.4	785.2	-	542.6	138.0	34.6	124.3
isolet	617	-	640.2	6280.7	1260.4	1735.7	-	642.4	838.1	201.6	263.8
Bioresponse	1776	-	1435.1	53978.2	10814.2	10531.4	-	1423.4	11031.8	1727.8	1632.8
SVHN_small	3072	-	3575.4	124041.7	33802.1	32440.0	-	3591.1	23332.5	5455.9	5146.8
gisette	5000	-	8666.3	80062.4	18811.6	18238.1	-	8606.4	78635.7	14210.1	14610.5
OHSUMED	45	154.1	436.8	13.5	1.7	89.4	497.9	435.1	13.9	1.5	89.2
MQ2007	46	472.5	444.5	15.6	1.9	91.5	74.0	445.0	12.8	1.5	91.2
MQ2008	46	122.7	436.7	15.0	1.6	91.7	96.4	443.1	13.0	1.5	91.3

- Weights in the Q matrix are
  - proportional to the Pearson correlation among features
  - proportional to the mutual information among features
  - proportional to the information content of a feature based on the predictions computed by a classifier using only that feature

Ferrari Dacrema, M., Moroni, F., Nembrini, R., Ferro, N., Faggioli, G., and Cremonesi, P. (2022). Towards Feature Selection for Ranking and Classification Exploiting Quantum Annealers. In Amigó, E., Castells, P., Gonzalo, J., Carterette, B. A., Culpepper, J. S., and Kazai, G., editors, *Proc. 45th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2022)*, pages 2814–2824. ACM Press, New York, USA.

QA for Information Access: Is That Difficult?

Quantum Computing for Information Access - Feature Selection.ipynb

File Edit View Insert Runtime Tools Help Last edited on November 20

Comment Share

+ Code + Text Connect

Now we will use Quantum Annealing for real. This works as follows:

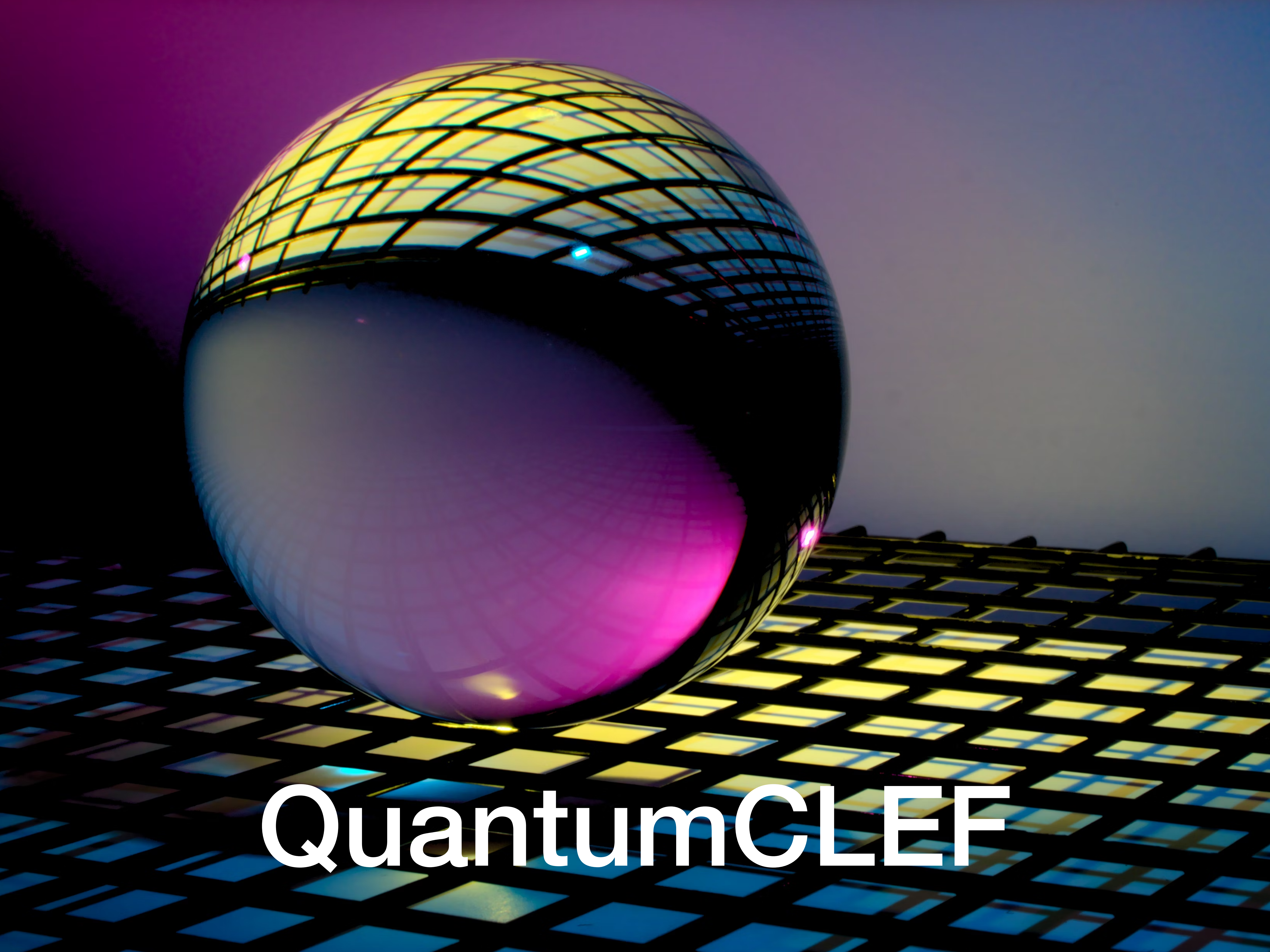
1. We need to have an API KEY which is required to have access to D-Wave's infrastructure.
2. Once we create our sampler and submit our problem, we will send it through internet. The problem will reach the D-Wave's infrastructure and will be enqued if there are currently other problems running.
3. Once it is our turn, our problem will be solved and the solution will be sent back to us.

```
[ ] %%time

    from google.colab import userdata
    token=userdata.get('API_TOKEN')

    sampler_QPU = DWaveCliqueSampler(token=token)

    response_QPU_4 = sampler_QPU.sample(kbqm,
                            label='Example - MI Feature Selection',
                            num_reads=num_reads)
```

```
CPU times: user 3.56 s, sys: 50.9 ms, total: 3.61 s
Wall time: 8.29 s
```

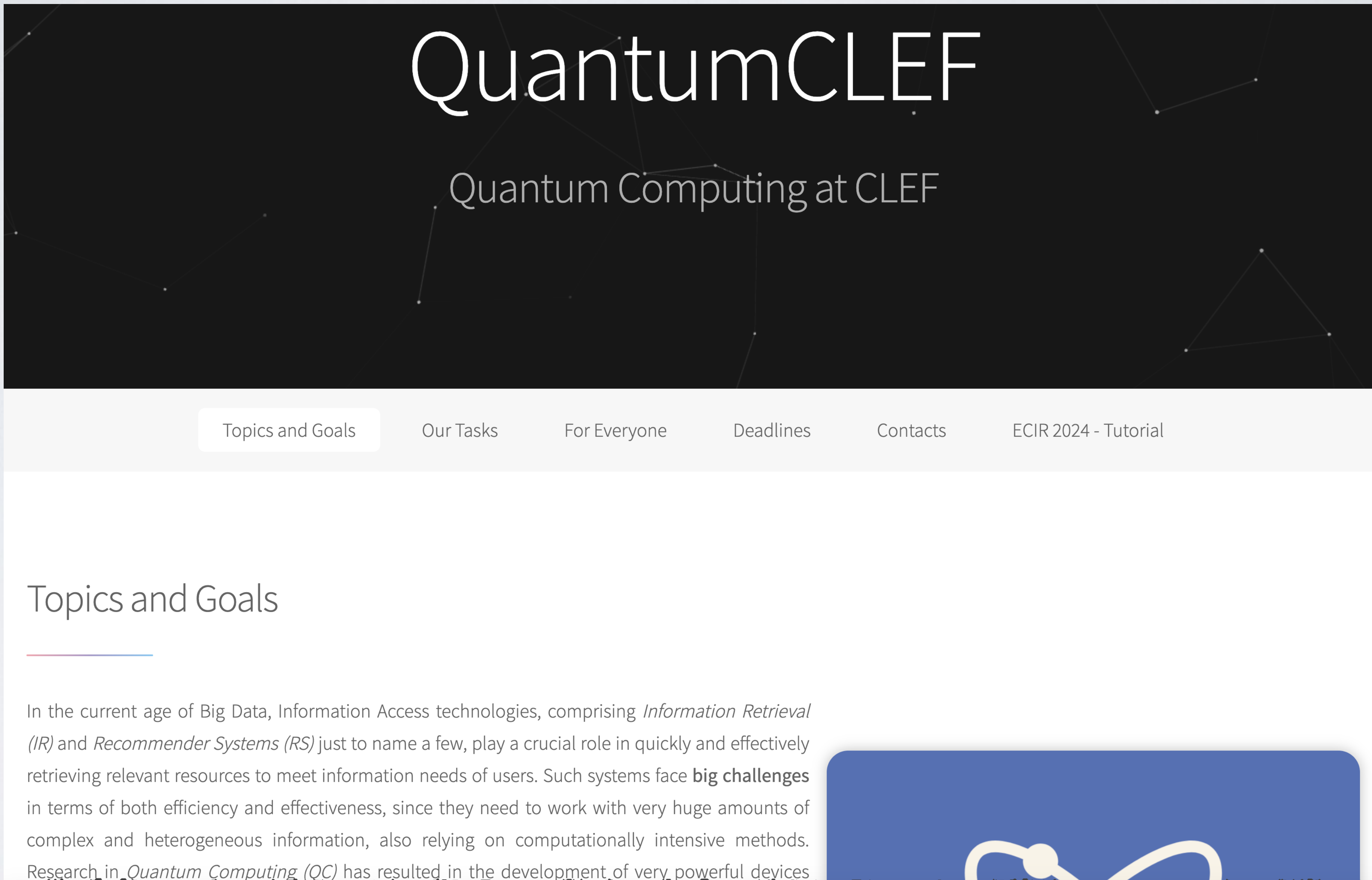
```
[ ] print_response_data(response_QPU_4.aggregate())
```

```
------------------------------------------------------------------
          Set 0          Set 1    Energy          Count
------------------------------------------------------------------
['age', 'alone', 'embarked port C', 'embarked port Q', 'embarked port S', 'fare', 'master', 'miss', 'mrs', 'rare']['cabin', 'mr', 'pclass'
['age', 'alone', 'cabin', 'embarked port C', 'embarked port Q', 'embarked port S', 'fare', 'master', 'miss', 'mrs']['mr', 'pclass', 'rare'
['age', 'alone', 'cabin', 'embarked port C', 'embarked port Q', 'embarked port S', 'fare', 'master', 'miss', 'rare']['mr', 'mrs', 'pclass'
['age', 'alone', 'cabin', 'embarked port C', 'embarked port Q', 'embarked port S', 'fare', 'master', 'mr', 'rare']['miss', 'mrs', 'pclass'
['age', 'alone', 'embarked port Q', 'embarked port S', 'fare', 'master', 'miss', 'mrs', 'pclass', 'rare']['cabin', 'embarked port C', 'mr'
['alone', 'embarked port C', 'embarked port Q', 'embarked port S', 'fare', 'master', 'miss', 'mrs', 'pclass', 'rare']['age', 'cabin', 'mr'
['age', 'alone', 'embarked port C', 'embarked port Q', 'embarked port S', 'fare', 'miss', 'mrs', 'pclass', 'rare']['cabin', 'master', 'mr'
['age', 'alone', 'embarked port C', 'embarked port Q', 'fare', 'master', 'miss', 'mrs', 'pclass', 'rare']['cabin', 'embarked port S', 'mr'
['age', 'alone', 'embarked port C', 'embarked port Q', 'embarked port S', 'fare', 'master', 'mrs', 'pclass', 'rare']['cabin', 'miss', 'mr'
['age', 'embarked port C', 'embarked port Q', 'embarked port S', 'fare', 'master', 'miss', 'mrs', 'pclass', 'rare']['alone', 'cabin', 'mr'
['age', 'alone', 'embarked port C', 'embarked port Q', 'embarked port S', 'fare', 'master', 'mr', 'pclass', 'rare']['cabin', 'miss', 'mrs'
['age', 'alone', 'embarked port C', 'embarked port Q', 'embarked port S', 'fare', 'master', 'miss', 'rare', 'sex']['cabin', 'mr', 'mrs', '
['age', 'alone', 'cabin', 'embarked port Q', 'embarked port S', 'fare', 'master', 'miss', 'pclass', 'rare']['embarked port C', 'mr', 'mrs'
```


QuantumCLEF

QuantumCLEF

https://qclef.dei.unipd.it/



Pasin, A., Ferrari Dacrema, M., Cremonesi, P., and Ferro, N. (2023). qCLEF: a Proposal to Evaluate Quantum Annealing for Information Retrieval and Recommender Systems. In Arampatzis, A., Kanoulas, E., Tsikrika, T., Vrochidis, S., Giachanou, A., Li, D., Aliannejadi, A., Vlachos, M., Faggioli, G., and Ferro, N., editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Four- teenth International Conference of the CLEF Association (CLEF 2023)*. Lecture Notes in Computer Science (LNCS) 14163, Springer, Heidelberg, Germany.

Tutorial at ECIR 2024

https://qclef.dei.unipd.it/ecir-2024-tutorial.html

Ferrari Dacrema, M., Pasin, A., Cremonesi, P., and Ferro, N. (2024). Quantum Computing for Information Retrieval and Recommender Systems. In Nazli, G., Tonellotto, N., He, Y., Lipani, A., McDonald, G., Macdon- ald, C., and Ounis, I., editors, *Advances in Information Retrieval. Proc. 46th European Conference on IR Research (ECIR 2024) – Part V*, pages 358–362. Lecture Notes in Computer Science (LNCS) 14612, Springer, Heidelberg, Germany.

Tasks 1 - Feature Selection

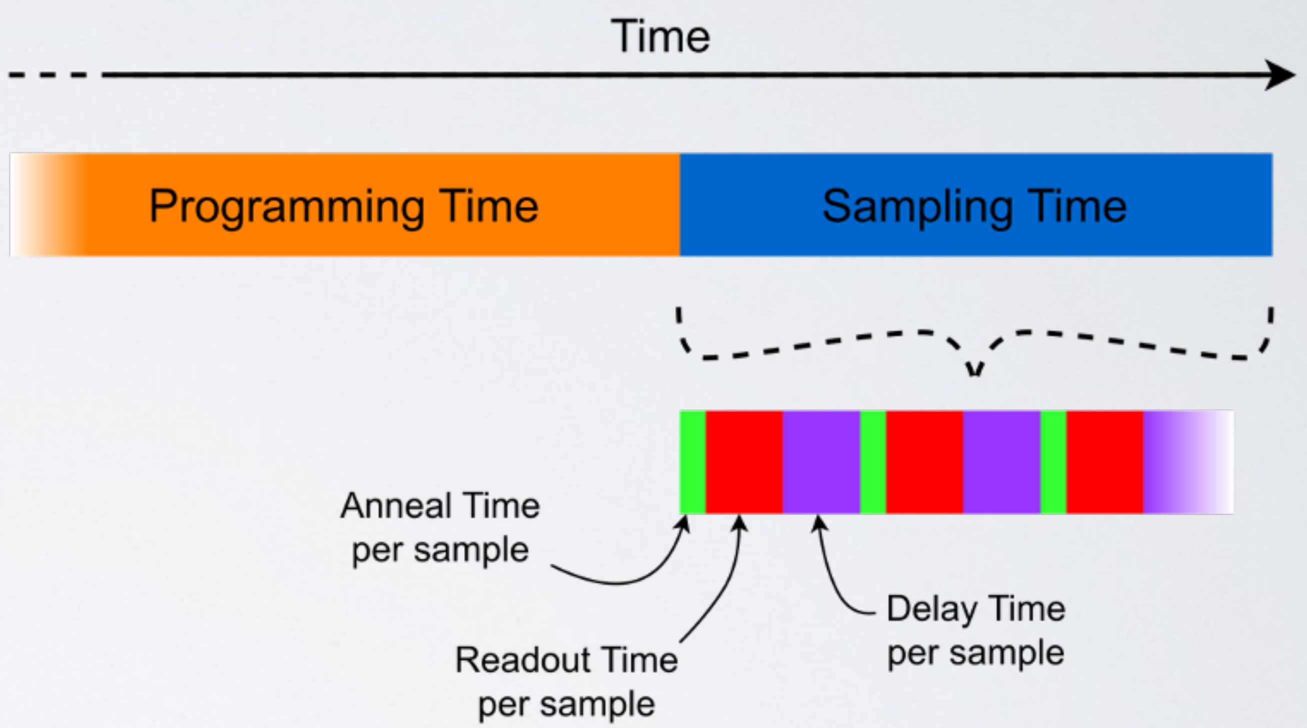
- **Quantum Feature Selection**: reducing the size of the input data to speed-up retrieval or enhancing effectiveness avoiding noisy features
- Task 1A - **Information Retrieval**
  - MQ2007 (one of the **LETOR** datasets), 46 features
  - **ISTELLA**, 220 features
  - Training of **LambdaMART** with the selected features to measure **nDCG@10**
- Task 1B - **Recommender Systems**
  - **150_ICM** (music recommendation): contains 150 features for each item
  - **500_ICM**: contains 500 features for each item
  - Training of an **Item-Based KNN** recommendation model to measure **nDCG@10**
- For each task submit runs using both Quantum Annealing (QA) and Simulated Annealing (SA)

Task 2 - Clustering in IR

- Task: obtain a list of **representative centroids** of the given dataset of embeddings (10, 25 and 50 vectors that represent the final centroids)
- **ANTIQUE** dataset in which each sentence taken from Yahoo is turned into an embedding using a transformer.
  - 6,500 sentences for training
  - 2,200 sentences for testing
- Measures
  - the **Davies-Bouldin Index** is used to measure the overall cluster quality. The index is improved (lowered) by increased separation between clusters and decreased variation within clusters.
  - **nDCG@10** is used to measure the overall retrieval quality based on a set of 50 queries.
  - Each query is transformed into its corresponding embedding, then the Cosine Similarity is used to get the closest centroid and its corresponding cluster of documents, finally all the documents belonging to that cluster are retrieved and ranked using the Cosine Similarity between the documents and the query
- Submit runs using both Quantum Annealing (QA) and Simulated Annealing (SA)

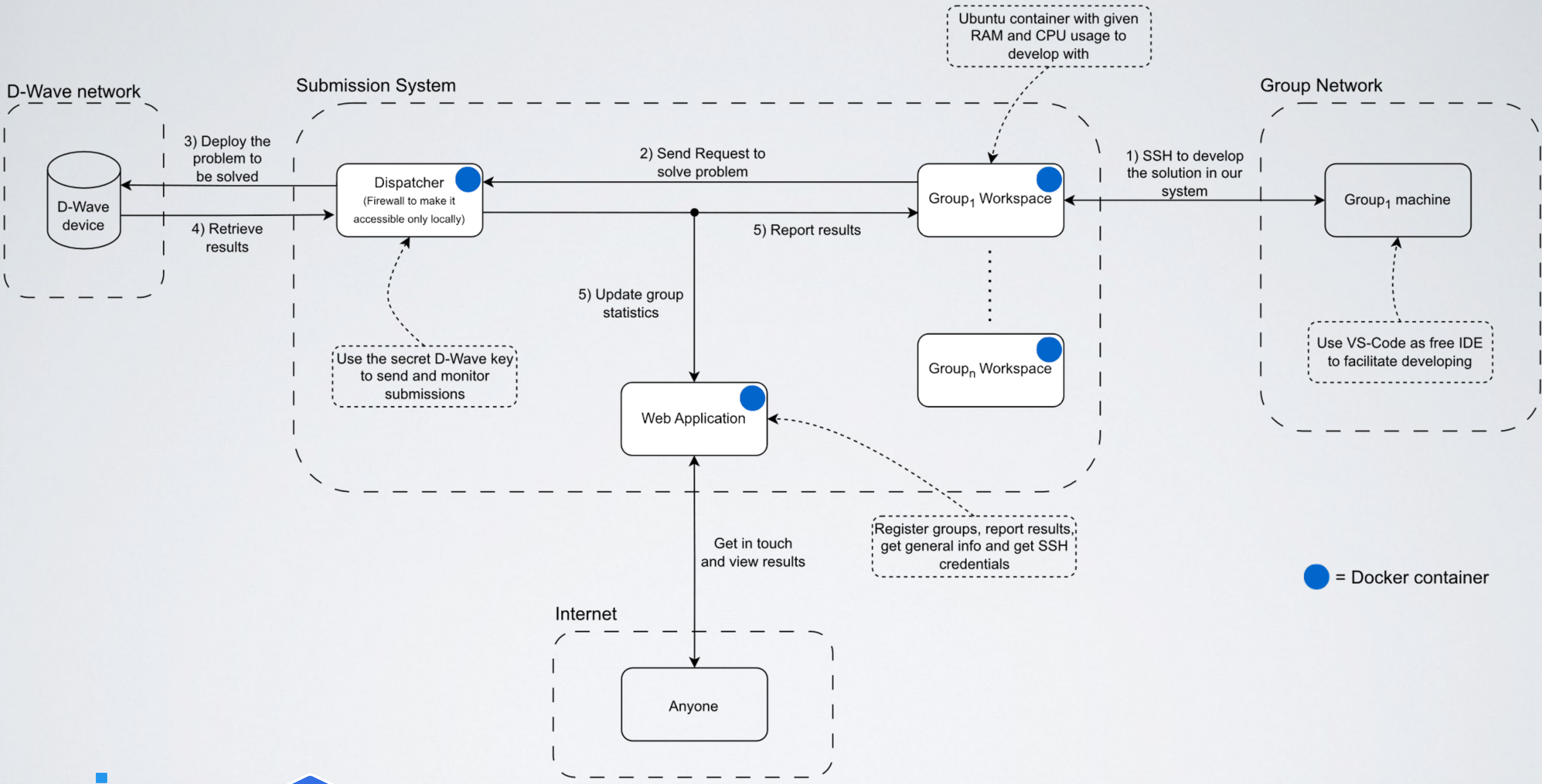
Challenges

- **Efficiency**: There is not a standard way to measure the efficiency of Quantum Annealers
  - There are several steps in the Annealing phase, each requiring a different amount of time based also on the used quantum annealer
- **Effectiveness**: Quantum Annealers do not ensure determinism and you need to sample multiple solutions to find the (hopefully) optimum
  - Can we leverage this to compare different runs by means of statistical analyses?



Resch Resch, S. and Karpuzcu, U. R. (2022). Benchmarking Quantum Computers and the Impact of Quantum Noise. *ACM Computing Surveys (CSUR)*, 54(7):142:1–142:35.

QuantumCLEF Infrastructure



docker kubernetes

QuantumCLEF Participation

- 26 groups registered for participating
  - **7 groups** submitted runs
- Submissions
  - Simulated Annealing (SA): 32
  - Quantum Annealing/Hybrid (QA): 34
- Break-down of submitted runs
  - Task 1A - Feature Selection for IR: 5 groups; 20 runs QA, 19 runs SA
  - Task 1B - Feature Selection for RecSys: 2 groups; 7 runs QA, 8 runs SA
  - Task 2 - Clustering for IR: 1 group; 7 runs QA, 5 runs SA
- Computing time
  - Simulated Annealing: ~9 hours (1.2 cores of AMD EPYC 3,6 GHz, 10 GByte RAM)
  - Quantum Annealing/Hybrid: ~5 minutes

Task 1A - LETOR, Preliminary Results

Group	nDCG@10	Annealing Time (us)	Type	Features
bit-ua	0.4410	0.2737	Q	18
bit-ua	0.4497	0.2698	Q	20
bit-ua	0.4410	1.3511	S	16
bit-ua	0.4446	3.6069	S	18
nica	0.4506	0.2741	Q	17
nica	0.4498	3.5097	S	15
ows	0.4495	0.2790	Q	25
ows	0.4506	0.2704	Q	25
ows	0.4480	0.2407	Q	25
ows	0.4475	2.8177	S	25
ows	0.4519	2.7524	S	25
ows	0.4515	2.7588	S	25
qtb	0.4299	0.3560	Q	13
qtb	0.4195	5.0000	H	10
qtb	0.4430	4.3092	H	10
qtb	0.4024	3.1741	S	10
shm2024	0.3650	0.0296	Q	5
shm2024	0.3621	0.0272	Q	5
shm2024	0.3910	0.0294	Q	5
shm2024	0.3477	0.0283	Q	5
shm2024	0.3245	0.0293	Q	5
shm2024	0.4024	0.2841	S	5
shm2024	0.3082	0.1640	S	5
shm2024	0.4249	0.1432	S	5
shm2024	0.4248	0.1467	S	5
shm2024	0.4205	0.1441	S	5
Baseline	0.4473	-	-	46

Task 1A - iSTELLA, Preliminary Results

Group	nDCG@10	Annealing Time (s)	Type	Features
bit-ua	0.6699	16.3247	S+Q	92
bit-ua	0.6905	0.5514	Q	82
bit-ua	0.6814	19.0711	S	90
bit-ua	0.7029	5.4039	S	72
bit-ua	0.7081	13.8274	S	161
nica	0.5960	0.4274	Q	15
nica	0.6211	3.9984	S	15
ows	0.6207	0.2152	Q	25
ows	0.6090	0.3936	Q	25
ows	0.6317	0.4020	Q	25
ows	0.6566	3.8747	S	25
ows	0.6541	3.7279	S	25
ows	0.6088	3.7848	S	25
Baseline	0.7146	-	-	220

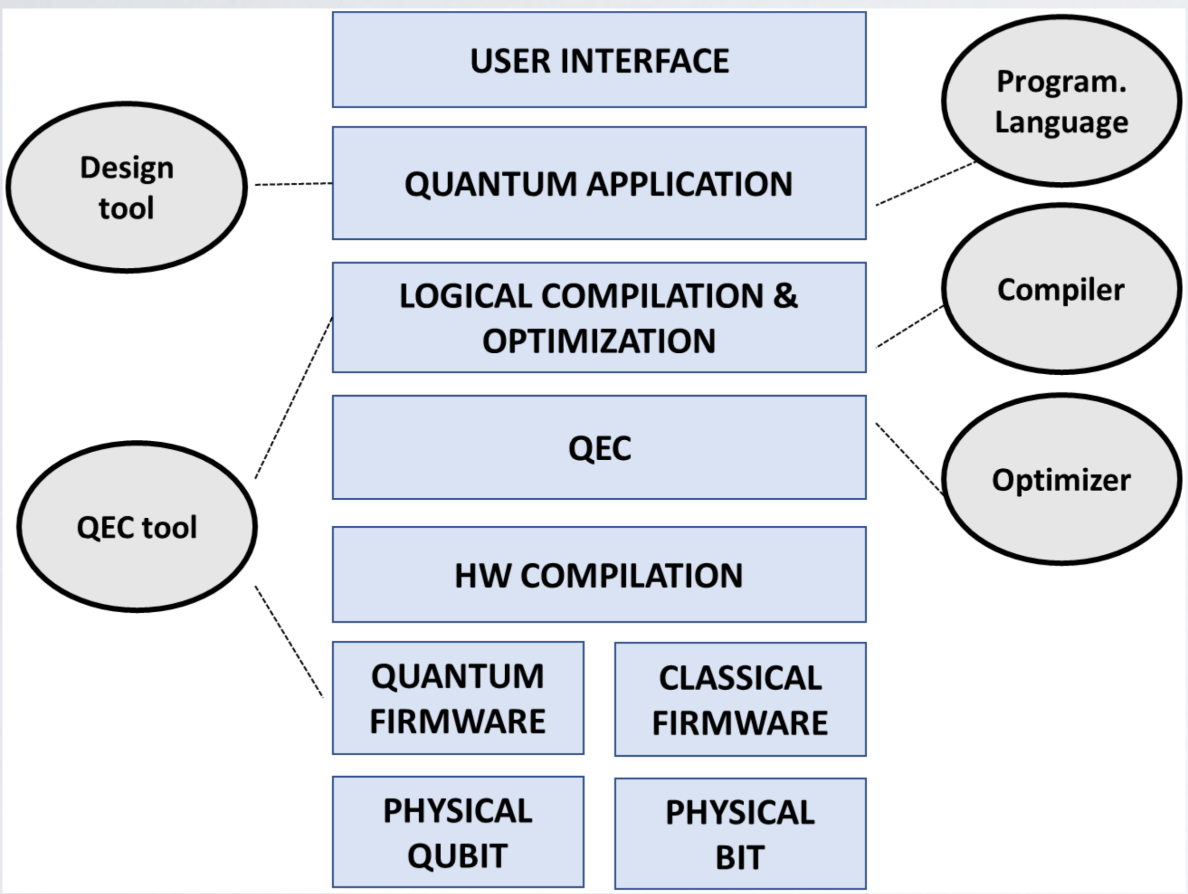
Task 2, Preliminary Results

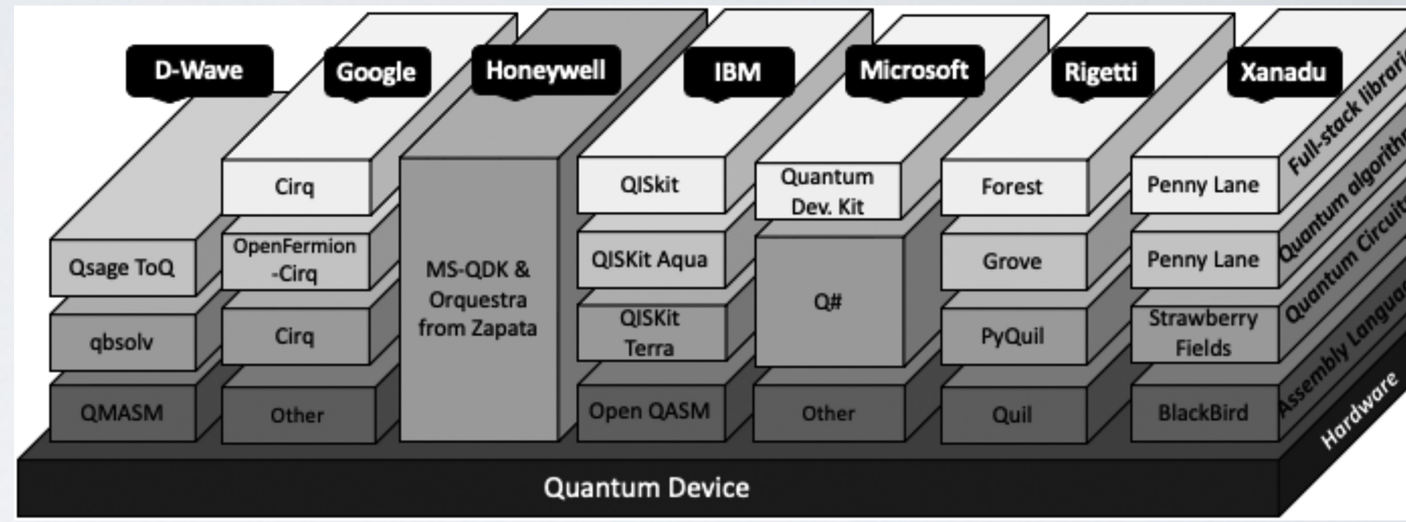
Centroids	Team	nDCG@10	DBI	Annealing Time (s)	Type
10	qIIMAS	0.5682	6.312144661	14.9934	H
10	qIIMAS	0.5622	6.684709428	535.5156	S
10	BASELINE	0.5509	7.989243805	-	-
25	qIIMAS	0.549	5.350996482	14.9946	H
25	qIIMAS	0.546	5.336858911	565.9855	S
25	BASELINE	0.5284	6.120091289	-	-
50	qIIMAS	0.5180	5.184247242	0.0185	Q
50	qIIMAS	0.5349	4.697829204	0.0667	Q
50	qIIMAS	0.5274	4.953743721	29.9954	H
50	qIIMAS	0.5564	4.803179136	14.9927	H
50	qIIMAS	0.5011	5.086784699	3.9937	H
50	qIIMAS	0.5068	4.786837347	610.0678	S
50	qIIMAS	0.5065	5.415931254	2.7928	S
50	qIIMAS	0.5310	4.811210971	2.8708	S
50	BASELINE	0.4656	5.367918023	-	-

Summary

- Quantum Computing for IR and RecSys is a largely unexplored area, yet there are potential benefits for our resource-intensive methods in terms of both efficiency and energy consumption, not to mention the fundamental research question: “Is this possible at all?”
- The Gate Model seems still very complex to be “easily” adopted in IR and RecSys while Quantum Annealing is more readily usable, also offering larger QPU
- Despite the complexity of the physics underlying QA, it can be approached also by newbie computer scientist since
    - you only need to understand how to express your problem with a QUBO formulation and much knowledge is already available for this
    - you can easily code your algorithms in plain Jupyter notebooks
- Many open challenges ahead of us
    - how to express our algorithms via a QUBO formulation
    - among alternative QUBO formulations, which are more effective and efficient?
    - QA is a technology still in its infancy and with many limitations: number of qubits, need for heuristics (minor embedding), potential errors, debate about its benefits wrt classical methods still open, …
    - How do we measure the efficiency of QA? How do we deal with the stochastic nature of this process when it comes to effectiveness?

Quantum Computing is Moving Towards Software Engineering





Requirement	D-Wave	Google	Honeywell	IBM	Microsoft	Rigetti	Xanadu
R1-agnostic					√		
R2-coexistence	√	√	√	√	√	√	√
R3-project management					√		
R4-evolution							
R5-zero defects							
R6-quality							
R7-reuse	√	√			√	√	√
R8-software governance							
R9-compatibility					√		
R10-efficiency	√	√	√	√	√	√	√
R11-reliability	√	√	√	√	√	√	√

Serrano, M. A., Cruz-Lemus, J. A., Pérez-Castillo, R., and Piattini, M. (2023). Quantum Software Components and Platforms: Overview and Quality Assessment. *ACM Computing Surveys (CSUR)*, 55(8):164:1– 164:31.

Serrano, M. A., Pérez-Castillo, R., and Piattini, M., editors (2022). *Quantum Software Engineering*. Springer International Publishing, Germany.

Thank You!