Outline

• Quantum computing problems
• Available hardware
• Programming SDKs, Frameworks
• Reference Architecture for cloud users
• EGI Notebook service integration
Quantum computing problems

• In traditional computing functions are created to perform some sort of calculation, which is translated to CPU code to alter the registers of the CPU and the content of the memory.

• Quantum computing follows different paradigms:
  • Quantum annealing problem solving (e.g. D-Wave)
  • Quantum circuits, based on quantum gates (e.g. IonQ),
Quantum annealing: D-Wave approach

• Instead of solving a problem programmatically, we are defining an objective function for the problem in an adequate format, where the global minimum of the function represents the best solution for the problem.
• Use the quantum hardware to find a global minimum for the objective function through sampling/solving.
Quantum circuits

• Qiskit definition: “A quantum circuit is a computational routine consisting of coherent quantum operations on quantum data, such as qubits, and concurrent real-time classical computation. It is an ordered sequence of quantum gates, measurements and resets, all of which may be conditioned on and use data from the real-time classical computation.”

• Implementations:
  • IonQ
  • Rigetti
  • Oxford Quantum Circuits
  • IBM
  • Google
Quantum hardware providers

• D-Wave:
  • Annealing systems
  • Provided through D-Wave Leap
  • Simulator available

• Amazon Braket:
  • Annealing systems: D-Wave
  • Gate-based systems: IonQ, Rigetti, Oxford Quantum Circuits
  • Simulator available for gate-based systems

• IBM:
  • Gate-based systems
  • Simulator available

• Azure:
  • Gate-based systems: IonQ, Rigetti, Oxford Quantum Circuits
  • Simulator available

• Google
Programming SDKs, Frameworks

• Providers offer their SDKs for easy access:
  • D-Wave: Ocean SDK (Python)
  • Amazon: AWS Braket SDK (Python)
  • IBM: Qiskit (Python)
  • Azure: Quantum Development Kit (Q#, Qiskit, Cirq)
  • Google: Cirq (Python)

• Frameworks enabling the adaptation of quantum paradigm:
  • PennyLane: Differentiable programming, Machine learning, Quantum chemistry
  • Qiskit: Machine learning, Finance, Optimization, Nature
  • Strawberry Fields: Graph algorithms, Machine learning, Chemistry
QuickStart reference architecture for cloud users

- **Aim:** offer a reference architecture for cloud users which allows the quick startup for experimenting with quantum computing, using cloud resources and quantum simulators/hardware
- **Implemented as a set of containers including features:**
  - Jupyter Lab for easy code editing and execution
  - Apache Spark cluster for demonstration of multiple solution possibilities
  - Prepared examples
- **Included SDKs, Frameworks:**
  - D-Wave Ocean SDK
  - Amazon Braket SDK
  - PennyLane
  - Qiskit
- **Startup possibilities:**
  - Docker Compose
  - Kubernetes
A set of examples are offered

For D-Wave resources:
- **Minimum vertex cover**
- Point clustering
- EV charger placement
- Logic gate simulation
- Quadratic Model examples

Using IBM Qiskit:
- **Deutsch-Jozsa algorithm**
Usage, exploitation possibilities

• Offered for interested ELKH Cloud users

• Experiment with manufacturing-related problems in the CO-VERSATILE project, and Hungarian TKP NKTA programs

• Implement Quantum Support Vector Machine algorithms inside the Hungarian Artificial Intelligence National Laboratory (MILAB)
EGI Notebook service integration

- A browser-based tool for interactive analysis of data using EGI storage and compute services
- Accessible through [https://notebooks.egi.eu/hub/login](https://notebooks.egi.eu/hub/login)
- Adding quantum resource access to EGI Notebooks:
  - Ongoing work
  - SDKs and frameworks offered by the quantum reference architecture will be included
  - Examples downloaded on demand
- Results:
  - Quantum resources will be accessible for EOSC users
  - Local simulators usable right from within the Notebook server started for a user
  - External access credentials are still required to compute using real quantum hardware
Interested?

• Check out the reference architecture, try it for yourself:
  
  https://git.sztaki.hu/science-cloud/reference-architectures/quantum

• If you have a fitting problem, don't hesitate to contact!

• Any feedback welcome:

  Zoltán Farkas, SZTAKI  
  zfarkas@sztaki.hu
THANK YOU FOR YOUR ATTENTION!

www.sztaki.hu