The CERN Quantum Technology Initiative

Alberto Di Meglio – CERN openlab Head
Sofia Vallecorsa – CERN openlab AI and Quantum Research
CERN OPENLAB’S MISSION

Our recipe for success

Evaluate and test state-of-the-art technologies in a challenging environment and improve them in collaboration with industry.

Communicate results, demonstrate impact, and reach new audiences.

Collaborate and exchange ideas with other communities to create knowledge and innovation.

Train the next generation of engineers/researchers, promote education and cultural exchanges.
Joint R&D Projects (Phase VI)

- Data Acquisition (LHCb, CMS, Dune, IT-CF)
- Code modernization (EP-SFT, LHC Exp., OPL)
- Cloud infrastructure (IT-CM)
- Networks (IT-CS)
- Data Storage (IT-ST, IT-DB, EP-DT)
- Data Analytics, Machine Learning (many)
- Control Systems (BE-ICS)
- Software Defined Networks, Security

- High-bandwidth fabrics, accelerated platforms for data acquisition
- Fast simulation, data quality monitoring, anomaly detection, physics data reduction, benchmarking/scalability, systems biology and large-scale multi-disciplinary platforms
- Predictive/proactive maintenance and operations

CERN QTI and QC Highlights
CERN openlab has organized a kick-off event of its Quantum Computing initiative on **November 5th-6th, 2018**

[https://indico.cern.ch/event/719844/](https://indico.cern.ch/event/719844/)

> 400 registered participants from the HEP physics community, companies and worldwide research laboratories and beyond

**Goals:**

Create a database of QC projects to foster **collaborations** between interested **user groups, CERN openlab and industry**

Continue to seek **opportunities** to support QC projects

Propose ways of scaling up the QC activities in 2020
Quantum collaborations

- D-Wave: Quantum Annealing
- IBM: Q-System Transmon (SC Gates), Sycamore Transmon (SC Gates)
- Google: Hybrid Classic-Quantum devices (SC Gates)
- Rigetti: Tangle Lake (SC Gates, spin gates)
- Microsoft, PsiQ, Atos: Topological qubits, Silicon photonics quantum computing, Specialized hardware simulators
- System integrators, algorithms, tools
- CERN QTI and QC Highlights
**CERN Unique Expertise and Activities**

### Computing

- **qGAN**
- **Real data**

### Sensing

- **Simulation**
- **Reconstruction**
- **Classification**

**BASE phase-sensitive measurement of spin allowing very precise magnetic field drift measurements**

https://doi.org/10.1140/epjst/e2015-02607-4

**ISOLTRAP Mass-Spec**

https://doi.org/10.1088/1361-6471/aa5a20

### Communications

- **openQKD Repeater node in the CERN Data Centre**

### Theory

- **Quantum Field Theory**

**https://cds.cern.ch/record/2703396**

**Lattice QCD**

Many pilot projects already started as part of the CERN openlab quantum programme (https://openlab.cern/quantum)
Quantum Computing Objectives

Define CERN role as part of broader QT development initiatives

Computing Science

• Potential of QC and QC+AI
• Build skills and experience on simulators
  • Build a distributed quantum simulation platform
• Design/implement PoCs for HEP workloads
  • Well understood problems are perfect playground to understand quantum algorithms
• Compare/extend on real devices

Engineering

• Collaborate in the definition/development of the software engineering stack for QC
• Collaborate in the development/integration of APIs and user interfaces to access QC systems
• Collaboration on engineering aspects of QC installation:
  • Cryogenics, material science, …
Quantum Sensing Objectives

Interferometric systems
- Beams of long-lived neutral atoms
- Matter or optical interferometers

Novel types of entanglement
- Three-photon decays and three-photon entanglement
- Multi-photon entanglement and quantum communication applications
- Entanglement of annihilation gamma-rays

Quantum clocks

Single atom/ion systems
- Rydberg atoms/antiprotonic atoms
- Detection of quantum state changes through annihilation
- Surface material studies and “antihydrogen reflectometry”
- Ion traps for Rydberg atoms, precision measurements of antiproton/positrons
- Detection of electron and ion fluxes at levels below current thresholds

Ultra-sensitive calorimeters based on energy-transition detection amplified by coherence effects

Ultra-low energy radiation detectors
CERN Quantum Technology Initiative

Strategy

Coordination

R&D

Capacity building

Joint HEP R&D Programme

CERN Management

QT Advisory Board (Member States)

Sensing, Detectors R&D

Computing & Engineering

Communication

Simulation, Information Processing

Academic Programmes / Industrial Collaborations / Knowledge Transfer

CERN QTI and QC Highlights
Example QC projects @ CERN openlab

- Quantum **Generative Adversarial Networks** for detector simulation
- Quantum **Random Number Generators** tests and integration
- Quantum **Graph Neural Networks** for particle trajectory reconstruction
- Quantum **Support Vector Machines** for signal/background classification (Higgs, SUSY, ..)
- Workload optimization via quantum **Reinforcement Learning**
- Quantum **Homomorphic Encryption**

Quantum Machine Learning

QML introduces quantum algorithms as part of a larger implementation

**Fully quantum** or **hybrid** classical-quantum
Quantum or Classical **input data**

How do we construct Quantum Neural Networks?
Direct association between **neurons and qubits**
Encode information into **amplitudes** of a quantum state

How do we represent learning rules?
Need **association rule** between NN activation patterns and pure quantum states

How do we address data loading?
**Quantum state preparation**
**Direct access** through qRAM?

Advantage?
**Representational** power
Possible to train on **large datasets** by only loading a **small number of samples**?
Quantum Generative Models

Classical Generative Models can replace Monte Carlo simulation

**3DGAN: Generative Adversarial Networks** prototype for calorimeter simulation

Detector output interpreted as a 3D image.

Quantum Generative Models might have **larger representational power**

**Quantum GAN** investigations:

Down sample 3DGAN use case to **manageable number of pixels**

Use **compressed data representation** in quantum states.

- **Qubits** or **Continuous variables**
- Different **hybrid** classical-quantum combinations

3DGAN generator

https://doi.org/10.1051/epjconf/201921402010
**Hybrid Classical-Quantum GAN**

*IBM qGAN can load probability distributions in quantum states*

**Simplify dataset**: extract 1D energy profiles from 3DGAN images

- $2^n$ classical pixels expressed by $n$ qubits
- Probability of getting state $|k\rangle = \text{(Relative) Energy at pixel } k$

Train a hybrid classical-quantum GAN to generate few-pixels image

- **Classical Discriminator (pyTorch)**: 512 nodes + Leaky ReLU $\rightarrow$ 216 nodes + Leaky ReLU $\rightarrow$ single-node + sigmoid
- **Quantum Generator (Qiskit)**: 3 $R_y$ layers

---

[Diagram and code snippets related to quantum circuits and probability distributions.]
qGAN for single electron in an electromagnetic calorimeter

Low resolution 2D energy profile

Increase generator and discriminator depth (6 qubits – 36 pixels)
- Realistic 2D images
- Stable training losses

IBM qGAN is not a real generative model
- It uses the adversarial training approach to embeds the image probability distribution in a quantum circuit
- Need a way to sample single images
Extending the qGAN model

Collaboration with Cambridge Quantum Computing

Two-steps quantum generator to learn the average distribution and sample images from it

- Ry variational form implemented using qiskit & \textit{t|ket}
- Uniform parameter initialization
- AMSGRAD optimizer with lr = $10^{-4}$, lr2 = $10^{-3}$

Classical discriminator (pyTorch) 4 nodes $\rightarrow$ 512 nodes $\rightarrow$ 256 nodes $\rightarrow$ 1 node

- Leaky ReLu between hidden layers + sigmoid
- AMSGRAD optimizer + Gradient penalty for stability and convergence
First images

Training data: 20,000 samples classified into 4 classes via K-means clustering

Can generate all four image classes!

WORK IN PROGRESS

One-to-one correspondence

- Image0 → Set0
- Image1 → Set3
- Image2 → Set1
- Image3 → Set2
Continuous Variables qGAN

Image data encoded as a displaced state $|x\rangle = \bigotimes_{i=1}^{N} |x_i\rangle = \bigotimes_{i=1}^{N} D_i(x_i) |0\rangle$

StrawberryFields for network implementation. Pennylane for autodifferentiation

Fully Quantum model:

Hybrid model:

CV architecture allows the use of non-linear activation function

Two backends for photonic QC simulation:

Fock:
- Can use non-gaussian gates
- Approximate state
- Cutoff dimension limited by computing resources

Gaussian:
- Precise simulation (exact Wigner function)
- Faster compared to Fock
- Cannot simulate non-Gaussian gate
  - Replace non-linear activation function with squeezing gate
CV classifier NN

Test quantum discriminator network \( n=3 \) depth =3

Compare 1D energy profile to flat distribution

**Fock**

<table>
<thead>
<tr>
<th>Prediction outcome</th>
<th>p</th>
<th>n</th>
</tr>
</thead>
<tbody>
<tr>
<td>actual value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p'</td>
<td>1.0</td>
<td>0.0</td>
</tr>
<tr>
<td>n'</td>
<td>0.036</td>
<td>0.964</td>
</tr>
</tbody>
</table>

**Gaussian**

<table>
<thead>
<tr>
<th>Prediction outcome</th>
<th>p</th>
<th>n</th>
</tr>
</thead>
<tbody>
<tr>
<td>actual value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p'</td>
<td>0.984</td>
<td>0.016</td>
</tr>
<tr>
<td>n'</td>
<td>0.039</td>
<td>0.961</td>
</tr>
</tbody>
</table>
Preliminary Results CV qGAN

**Hybrid**

- Fully quantum approach doesn't fully converge
- Use **non-linear activation** function (kerr gate) for the generator
- Generator is probably not strong enough
  - Try image clustering approach

**Fully Quantum**

- Training time is time-consuming and highly limited by computing power
  - Multiprocessing with 50 cores yields
    - 160 min/epoch (quantum)
    - 80 min/epoch (hybrid)
Charged particle tracking

**Hep.TrkX** project introduces **Graph Neural Networks** for particle trajectory reconstruction

- Data as a *graph of connected hits*
- Connect plausibly-related hits using geometric constraints
- Full event embedding requires **large graphs** (~10^5 nodes)

HepTrkX GNN is a cascade of **Input**, Edge and Node Networks

- **Edge network** outputs edge features, using the start and end nodes
- **Node network** classifies nodes using all connected nodes features on the previous and next layers

*arxiv:1810.06111*
GNN for particle tracking

HEP.TrkX GNN Scores:
- Purity: 99.5%
- Efficiency: 98.7%
- Overall Accuracy: 99.5% with 0.5 threshold

Quantum: Gate Level implementation
A Quantum Classifier

*Edge and Node NN as Tree Tensor Networks*

Node 0 \((r_0, \phi_0, z_0)\)

Node 1 \((r_1, \phi_1, z_1)\)

There are 11 parameters (shown in red boxes) to optimize.

Ry: Y Rotation on the Bloch Sphere

Input Encoding

Output Measurement

TTN Circuit
Quantum Networks

Quantum Edge Network

Increase the size of **hidden dimension** by increasing the number of qubits.

Node Features

Hidden Features

Edge Information
Quantum Networks

Quantum Node Network

A circuit is setup for each possible neighbor. 2 independent circuits are required for this example.

Recurrent Iterations

Multiple neighbours for nodes

Node of Interest

CERN QTI and QC Highlights
Training Results of the QGNN

Comparison to Simple Classical Networks (2 epochs)

Training set: 1400 subgraphs, Validation set: 200 subgraphs, ADAM optimiser, binary cross entropy, lr = 0.01, shots =1000. Hidden Dimension Size = 1. Classical Networks have x100 learning rate.

Simple experiments with Classical Networks show the potential for the Quantum Network.
Conclusions

CERN Quantum Technology Initiative established
- Assess the potential of QC in the time scale of the High Luminosity LHC runs and beyond
- Build skills for future programmes
- Provide a thematic HEP focus for international collaborations

CERN openlab has started investigating opportunities in QC and across QC and other relevant fields (AI, HPC)
- Foster collaborations between scientists and industry
- Major focus on education and skills development

While still not a ready for prime-time production, Quantum Computing holds the promise to herald a revolution in ICT

Initial results are very exciting
- Quantum Machine Learning particularly promising
Thanks!

Questions?
## CV basics

<table>
<thead>
<tr>
<th>Qubit</th>
<th>CV</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Fundamental unit</strong></td>
<td>Qumode ( {</td>
</tr>
<tr>
<td>(</td>
<td>i</td>
</tr>
<tr>
<td>(</td>
<td>\psi\rangle = \alpha</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Relevant Operators</th>
<th>Pauli operators (\hat{\sigma}_x, \hat{\sigma}_y, \hat{\sigma}_z)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quadrature operator (\hat{x}, \hat{p})</td>
<td>Phase shift, Rotation,</td>
</tr>
<tr>
<td>Mode operators (\hat{a}, \hat{a}^\dagger)</td>
<td>Hadamard (Single Qubit),</td>
</tr>
<tr>
<td>Common gates</td>
<td>CNOT (Two Qubit)</td>
</tr>
<tr>
<td>Displacement, Rotation,</td>
<td></td>
</tr>
<tr>
<td>Squeezing (Single Qubit),</td>
<td></td>
</tr>
<tr>
<td>BeamSplitter (Two Qubit)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Measurement</th>
<th>Pauli basis measurements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Homodyne (</td>
<td>x_\phi\rangle \langle x_\phi</td>
</tr>
<tr>
<td>Heterodyne (\frac{1}{\sqrt{2}}</td>
<td>\alpha\rangle \langle \alpha</td>
</tr>
<tr>
<td>Photon counting (</td>
<td>n\rangle \langle n</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type</th>
<th>Gate</th>
<th>Unitary form</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Gaussian</strong></td>
<td>Displacement</td>
<td>(D(\alpha) = \exp(\alpha \hat{a}^\dagger - \alpha^* \hat{a}))</td>
</tr>
<tr>
<td></td>
<td>Rotation</td>
<td>(R(\phi) = \exp(i \phi \hat{a}))</td>
</tr>
<tr>
<td></td>
<td>Squeezing</td>
<td>(S(r) = \exp(\frac{1}{2}(r^* \hat{a}^\dagger \hat{a}^2 - r \hat{a}^2 \hat{a}^\dagger)))</td>
</tr>
<tr>
<td></td>
<td>Beam Splitter</td>
<td>(BS_{ij}(\theta, \phi) = \exp\left(\theta(e^{i\phi} \hat{a}_i^\dagger \hat{a}_j - e^{-i\phi} \hat{a}_i \hat{a}_j^\dagger)\right))</td>
</tr>
<tr>
<td><strong>None-Gaussian</strong></td>
<td>Kerr</td>
<td>(K(\kappa) = \exp(i \kappa \hat{n}^2))</td>
</tr>
<tr>
<td></td>
<td>Cubic</td>
<td>(V(\gamma) = \exp(i \frac{\gamma}{3} \hat{x}^3))</td>
</tr>
</tbody>
</table>
Quantum Classifiers

Hierarchical quantum classifiers

Edward Grant$^{1,2}$, Marcello Benedetti$^{1,3}$, Shuxiang Cao$^{4,5}$, Andrew Hallam$^{6,7}$, Joshua Lockhart$^1$, Vid Stojicic$^8$, Andrew G. Green$^6$ and Simone Severini$^1$

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Units</th>
<th>Rotations</th>
<th>Is &gt; 4</th>
<th>Is even</th>
<th>0 or 1</th>
<th>2 or 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>TTN</td>
<td>Simple</td>
<td>Real</td>
<td>65.59 ± 0.57</td>
<td>72.17 ± 0.89</td>
<td>92.12 ± 2.17</td>
<td>68.07 ± 2.42</td>
</tr>
<tr>
<td>TTN</td>
<td>General</td>
<td>Real</td>
<td>74.89 ± 0.95</td>
<td>83.13 ± 1.08</td>
<td>99.79 ± 0.02</td>
<td>97.64 ± 1.60</td>
</tr>
<tr>
<td>MERA</td>
<td>General</td>
<td>Real</td>
<td>75.20 ± 1.51</td>
<td>82.63 ± 1.19</td>
<td>99.84 ± 0.06</td>
<td>98.02 ± 1.40</td>
</tr>
<tr>
<td>Hybrid</td>
<td>General</td>
<td>Real</td>
<td>76.30 ± 1.04</td>
<td>83.53 ± 0.21</td>
<td>99.07 ± 0.02</td>
<td>98.07 ± 1.46</td>
</tr>
<tr>
<td>TTN</td>
<td>Simple</td>
<td>Complex</td>
<td>70.90 ± 0.73</td>
<td>80.12 ± 0.64</td>
<td>99.37 ± 0.12</td>
<td>94.09 ± 3.37</td>
</tr>
<tr>
<td>TTN</td>
<td>General</td>
<td>Complex</td>
<td>77.56 ± 0.45</td>
<td>83.53 ± 0.69</td>
<td>99.77 ± 0.02</td>
<td>97.63 ± 1.48</td>
</tr>
<tr>
<td>MERA</td>
<td>General</td>
<td>Complex</td>
<td>79.10 ± 0.90</td>
<td>84.85 ± 0.20</td>
<td>99.74 ± 0.02</td>
<td>98.86 ± 0.07</td>
</tr>
<tr>
<td>Hybrid</td>
<td>General</td>
<td>Complex</td>
<td>78.36 ± 0.45</td>
<td>84.38 ± 0.28</td>
<td>99.78 ± 0.02</td>
<td>98.46 ± 0.19</td>
</tr>
<tr>
<td>Logistic</td>
<td>N/A</td>
<td>N/A</td>
<td>70.70 ± 0.01</td>
<td>81.72 ± 0.01</td>
<td>99.53 ± 0.01</td>
<td>96.17 ± 0.01</td>
</tr>
</tbody>
</table>

Mean test accuracy and one standard deviation are reported for TTN, MERA, and hybrid classifiers with five different random initial parameter settings using two different types of unitary parametrization. Hybrid classifiers consist of pre-training a TTN classifier and then transforming it into a MERA classifier by training additional unitaries. Bold values indicate the best result for each classifier task.
Training the Network

Run the circuit \( N \) times.

Averaging the measurement outcomes gives a probability of being an edge.

Calculate an error from the ground truth data.

Calculate the gradients of the parameters using parameter shift rule.

Update the parameters:

\[ \theta = \theta - \eta \nabla_{\theta} J(\theta) \]

Repeat!
Plotting the Data

In Cylindrical Coordinates

Blue: After preprocessing with Hep.TrkX methods

Red: Ground Truth

1/16 of an event
Workload Optimization

Optimize Grid data placement

Focus on **LHC Computing Grid sites** used by the **ALICE** experiment

A **complex** and highly non-linear environment

- **70 computing centres** in 40 countries
- **150,000 CPU cores** and **120 PB of storage**
- **~140,000 jobs** running 24 x 7 x 365

**Data** is distributed according to some “reasonable” heuristic, not optimal

Optimisation via Reinforcement Learning

1 year funding from EU through the ATTRACT initiative
Quantum Reinforcement Learning

Quantum Boltzman Machine on quantum Annealer

Auto-Encoder + LSTM to reproduce I/O throughput

4% prediction accuracy is an extraordinary result for the simulation of such complex dynamic systems

Development of the optimisation tool as Quantum Boltzman Machine (QBM)

Quantum implementation on the D-Wave Quantum Annealer and minimisation of the QBM free energy (arxiv.org:1706.00074).