

# 量子-HPCハイブリッドプラットフォームに於ける 量子機械学習の検証

トヨタ自動車（株）・先端材料技術部 菅 義訓、（株）グリッド・エンジニアリング第1部 斯波 廣大

実現したい姿

- ・究極のハイ・パフォーマンス・コンピューティング(HPC)
- ・3本柱でコンピューティング革新、CAE超高度化

## ①化学

→ 超高精度の電子状態予測、設計戦力化

## ②最適化(CAE)

→ 最速&最良の求解ソリューション、技術定着化

## ③量子AI

→ 既存ツールを超える学習能力、使い分け確立



クルマ(モビリティ)  
、まち(社会)  
を瞬時にシミュレート

テーマ構成

- ・後工程(お客様)のうれしさ

- 実験レス、無駄ゼロ、究極のデジタルツイン

'25 '30 '35 '40

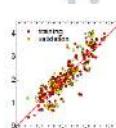
化学



シミュレータ検証 → 実機(PreFTQC) → 実機(FTQC)

究極のHPC

最適化



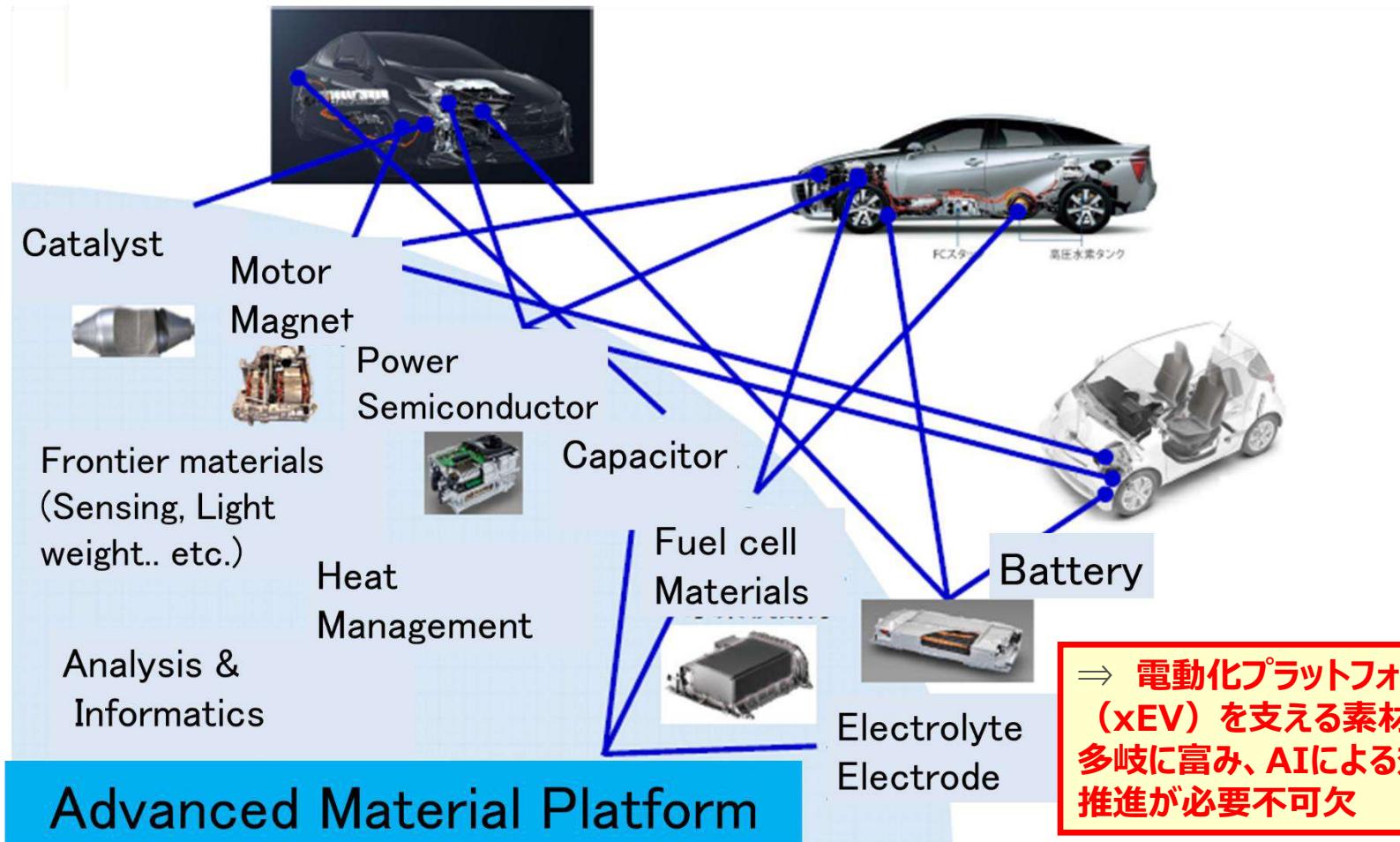
既存手法を超越 → 解析活用 → 設計ツールへ

量子AI

適用対象ベンチマーク → 現場活用 → 標準ツール化  
古典／量子の使い分け確立 → 標準ツール化

現状の立ち位置と克服シナリオ



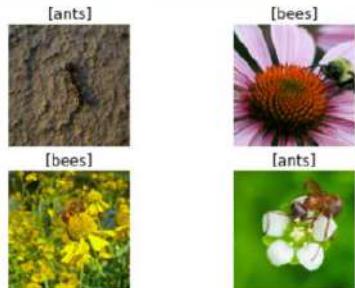
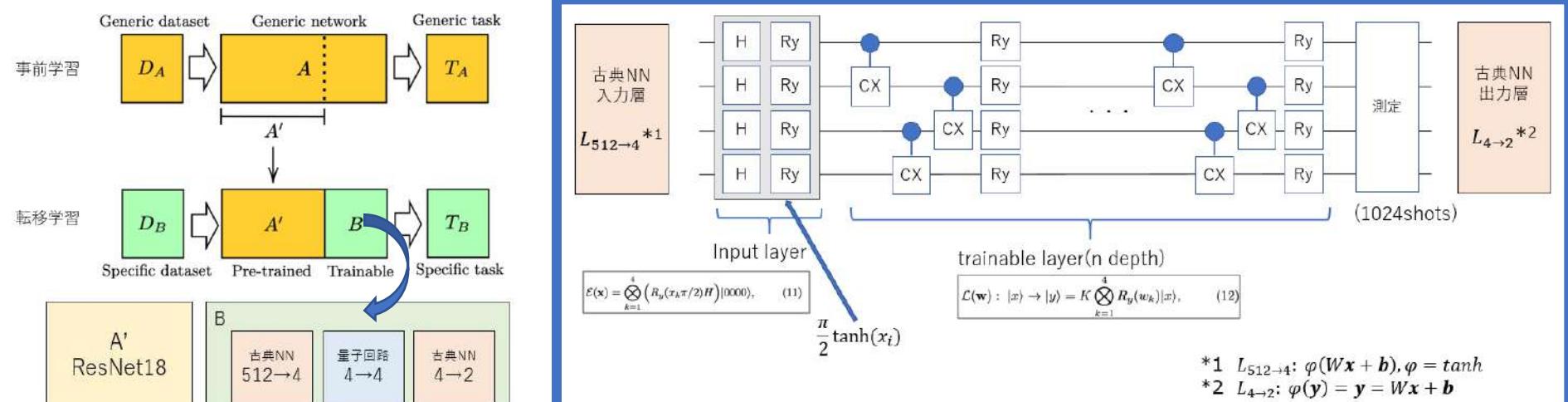


# ユースケース調査対象

## 課題タスク設定(古典NN+量子QCLでの転移学習表現(トレス量子回路))

A : B = 古典 : 量子、量子 : 古典、量子 : 量子の3パターンで転移学習を行う

転移学習では、事前学習済みネットワークAの最終層の一部を除いたA'を用いる



既存NNに対して、量子機械学習モデルのアドオンで得られる“うれしさ”を検証

⇒ 古典だけでは実現し得ない、広大な特徴空間による表現力向上を狙う

Ref.) Andrea Mari, Thomas R. Bromley, Josh Izaac, Maria Schuld, and Nathan Killoran, Quantum **4**, 340 (2020).

# Transfer learning in hybrid classical-quantum neural networks

Andrea Mari, Thomas R. Bromley, Josh Izaac, Maria Schuld, and Nathan Killoran, Quantum 4, 340 (2020).

## Transfer learning in hybrid classical-quantum neural networks

Andrea Mari, Thomas R. Bromley, Josh Izaac, Maria Schuld, and Nathan Killoran

Xanadu, 777 Bay Street, Toronto, Ontario, Canada.

We extend the concept of transfer learning, widely applied in modern machine learning algorithms, to the emerging context of hybrid neural networks composed of classical and quantum elements. We propose different implementations of hybrid transfer learning, but we focus mainly on the paradigm in which a pre-trained classical network is modified and augmented by a final variational quantum circuit. This approach is particularly attractive in the current era of intermediate-scale quantum technology since it allows to optimally pre-process high dimensional data (e.g., images) with any state-of-the-art classical network and to embed a select set of highly informative features into a quantum processor. We present several proof-of-concept examples of the convenient application of quantum transfer learning for image recognition and quantum state classification. We use the cross-platform software library PennyLane to exper-

[int-ph] 8 Oct 2020

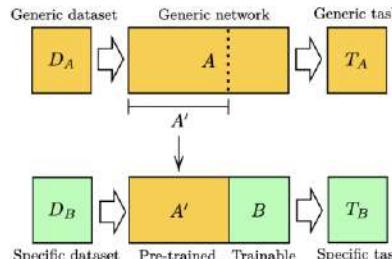
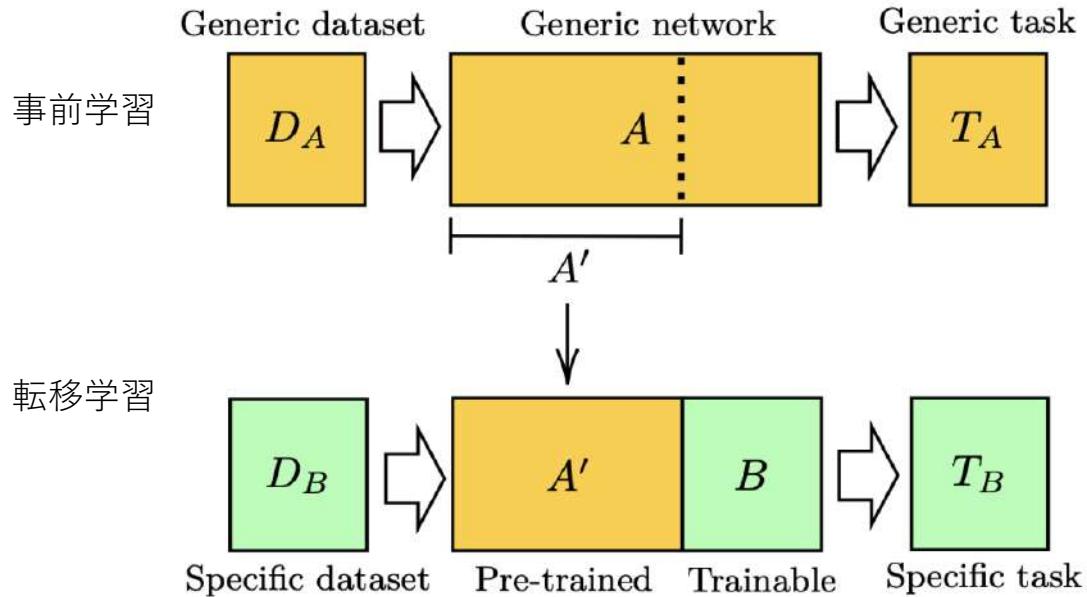


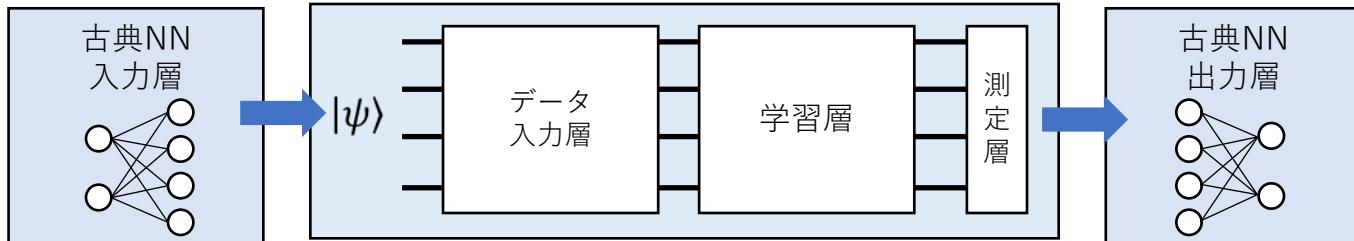
Figure 1: General representation of the *transfer learning* method, where each of the neural networks  $A$  and  $B$  can be either classical or quantum. Network  $A$  is pre-trained on a dataset  $D_A$  and for a task  $T_A$ . A reduced network  $A'$ , obtained by removing some of the final layers of  $A$ , is used as a fixed feature extractor. The second network  $B$ , usually much smaller than  $A'$ , is optimized on the specific dataset  $D_B$  and for the specific task  $T_B$ .

事前に学習済みの古典NN(ResNet)と量子層を含む学習層を組み合わせて、転移学習を実施

古典機械学習をPytorch、量子計算をQiskitやPennyLaneで実装

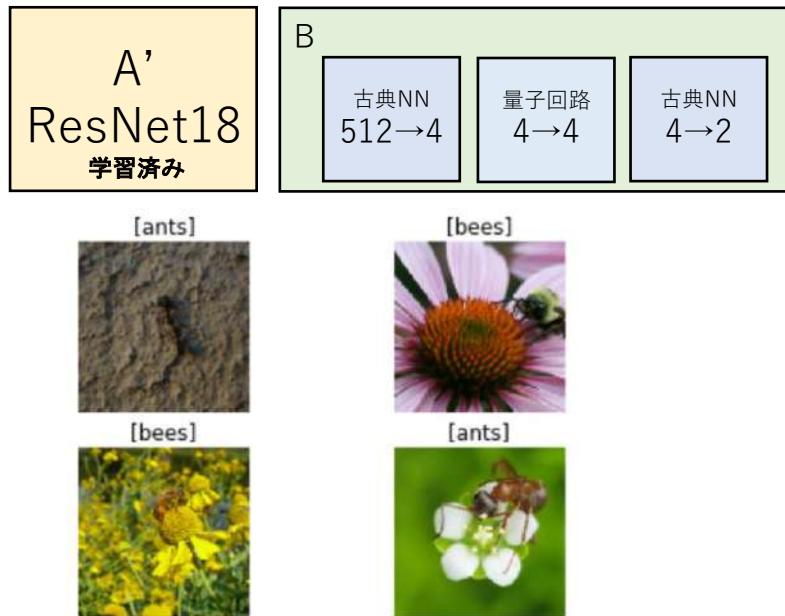


学習層B



CQ転移学習1の結果を再現

アリとハチの画像分類



### 再現結果

QPU	Accuracy
Simulator (PennyLane)	0.9513
古典NN (4→4)	0.9568
古典NN (16→16)	0.9627
IBMQ Hardware	<b>0. 9608 (on kawasaki)</b>

### 論文の結果

QPU	Accuracy
Simulator	0.967
ibmqx4	0.95
Aspen-4-4Q-A	0.80

ドレス量子回路の計算では、論文の結果に概ね近い値を再現できた  
 古典NNに置き換えた場合は、ドレス量子回路の結果よりも良くなり、ノード数を増やすとさらに向上することがわかった

以下の2種類の計算において、  
上記の計算環境設定の違いを検証した  
すべてPytorch (GPU) + PennyLane Lightning GPU  
(cuQuantum)

- Qubit = [5, 10, 15]
- Depth = [10, 15, 20, 25]

検証する計算環境設定は以下の4パターン

- GPU: 1枚, memory: 64GB
- GPU: 1枚, memory: 90GB
- GPU: 2枚, memory: 64GB
- GPU: 2枚, memory: 90GB

**Interactive Apps**

- Development
- Desktop
- Jupyter
- MATLAB (BYOL)
- RStudio
- Terminal
- VSCode
- Profiler
- NVIDIA Profiler
- Quantum
- Jupyter-qc**
- Viewer
- AVS/Express
- C-Tools
- Gnuplot

**Jupyter-qc**

This app will launch a JupyterLab with Quantum Software.

**Queue**

gpu1

**Number of GPUs**

1

Only effective when using gpu1/gpu2. Maximum 3 hours on gpu1, 24 hours on gpu2.

**Memory (GB)**

64

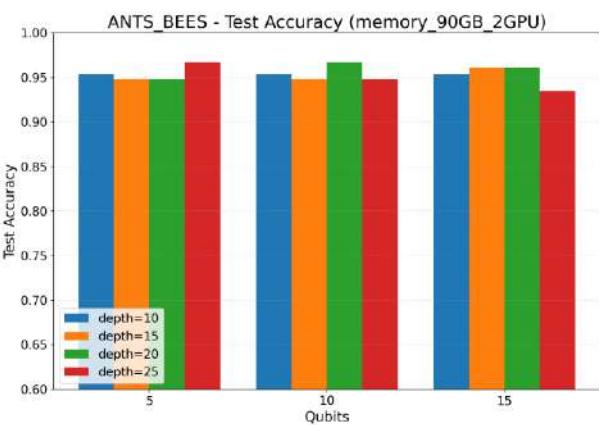
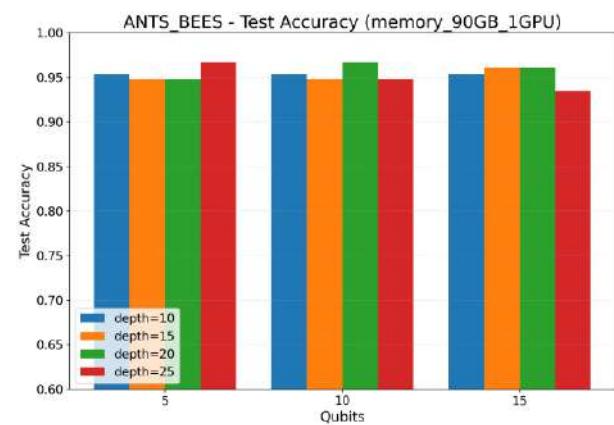
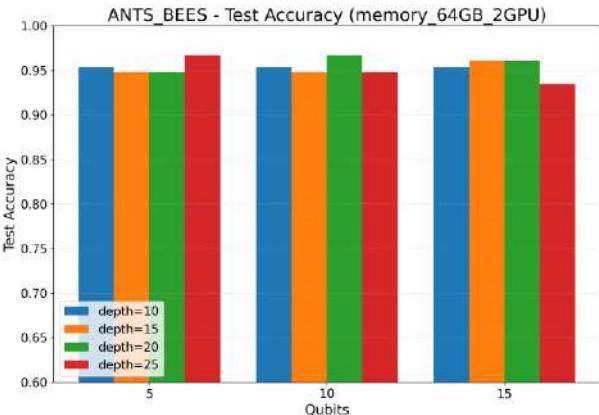
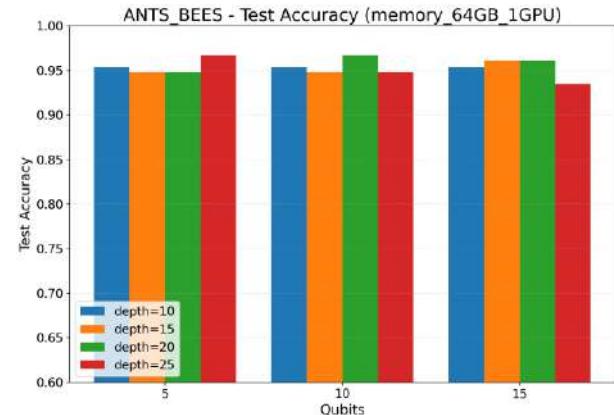
Slurm option: --mem=XXX MAX:180GB/gpu\_node

**Maximum run time (hours)**

1

**Launch**

\* The Jupyter-qc session data for this session can be accessed under the [data root directory](#).



どの結果も精度が95%前後  
5~15qubitsでは  
あまり違いがない

メモリサイズやGPU枚数による違いもない

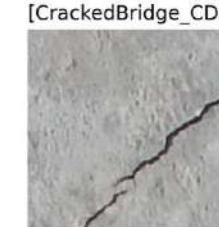
アリとハチの分類タスクだけでなく、コンクリートのひび割れ分類タスクも合わせて実施

Jupyter-qc環境で計算を実施し、GPUは2枚、memoryは90GBとした  
また、すべてPytorch (GPU) + PennyLane Lightning GPU (cuQuantum)で計算を実施した

アリとハチ

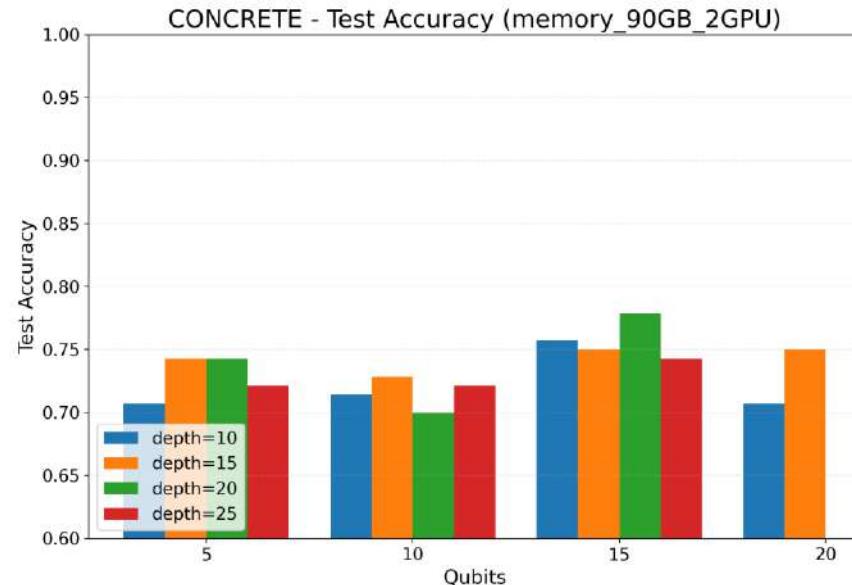
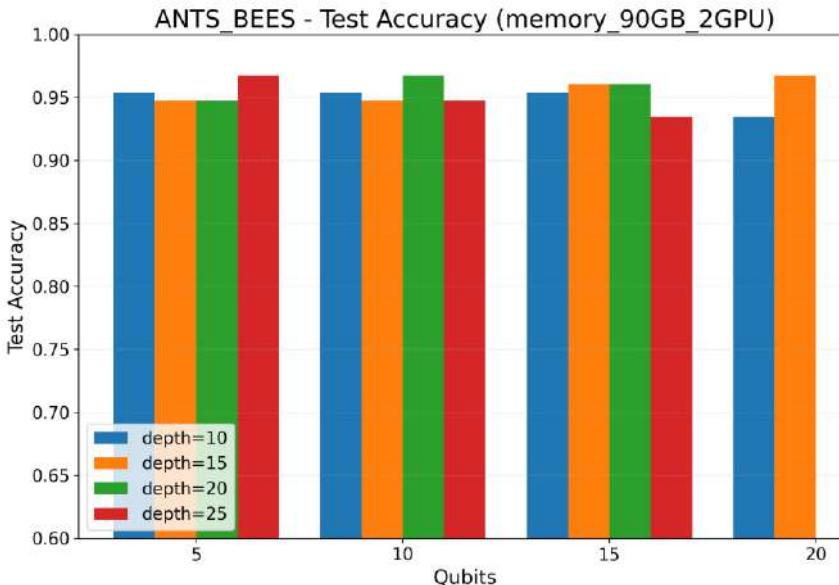


コンクリートのひび割れ



# 学習データセットの違いによる性能検証

アリとハチの分類に比べて、コンクリートのひび割れ分類はテスト精度が低い  
量子ビット数を増加させると、テスト精度がやや上昇する  
20qubitにおける回路の深さが20と25の場合は未検証



# まとめ

- Jupyter-qc環境で、量子転移学習アルゴリズムを用いて画像分類タスクの学習とテストを実施した
- GPUを用いた計算では、現時点で量子ビット数が20、量子回路の深さが15まで検証することができた
- (プリポスト環境での試行では、計算時間やメモリを考慮すると、上記程度が問題サイズの限界)

## 今後の展望

- 計算環境を富岳の計算ノードに変更し、問題サイズの検証を同様に実施する
- GPUクラスタでは再現できない大規模な問題サイズを検証し、富岳x量子ハードの有用性を検証する