Searching QEC Codes with Differentiable Quantum Architecture Search

PEIYONG WANG

July 2021

1 Research Rationale

- We want device-tailored quantum error correction code that specially optimized for specific device topology and (unknown) noise profile;
- We lack a systematic approach for discovering new quantum error correction codes without direct knowledge of the noise profile;
- Compared to its classical counterpart, Neural Architecture Search, which often needs to search through a very large number of different NN blocks (convolution, dropout, normalization, etc), as well as different connection schemes, the search space of different quantum operations is small compared to that of neural network. Also, for quantum circuits, we don't need deal with dimension reduction and stacking features.
- There have been previous attempts, like the QVECTOR [JRO$^+$17] paper, or the paper from Simon C. Benjamin's group [XBY21].But they employed fixed ansatz structures. In [JRO$^+$17], the authors adopted a very "generallized" circuit structure, which will need a large number of parameters when the number of qubits goes up. In [XBY21], the authors randomly initialized an ansatz, train it with classical optimizers. If the results are not acceptable, another circuit is randomly initialized, and the training process is repeated, until satisfactory results are reached. In Thomas Fösel's paper [FTWM18], the authors adopted a reinforcement learning approach, a neural network agent trained to assign appropriate quantum operations to preserve quantum information.

However, to make the NN agent work with real quantum computers, a teacher network has to be printed first with quantum simulators, which will need more time and more computational resources. It is also hard to adopt RL for neural-architecture-search-like tasks;

- Differentiable quantum architecture search [ZHZY20] can be more easily optimized with gradient based classical optimizers.

2 Goal

To develop an encoding circuit for the code words of the five-qubit quantum error correction code that can produce high-fidelity encoded states under the influence of some noise profile (At first we can set the noise profile to only single-qubit depolarization channel, later we could directly adopt a noise profile from a real device).

3 Methods

3.1 Steps for DQAS

- The circuit: $U = \prod_{i=0}^{p} U_i(\vec{\theta}_i)$;
- Loss function: $L = L(U) = L(\vec{\theta})$;
- Embedding the discrete structure choices into a continuously-parameterized probabilistic model $P(\mathbf{k}, \boldsymbol{\alpha})$, $\boldsymbol{k}$ is a discrete structural parameter, if $\boldsymbol{k} = [1, 3, 1]$, then $U(\mathbf{k}) = V_1 V_3 V_1$, where V_1 and V_3 are elements in the predefined operation pool.
-
  - $\mathbf{k}$ are sampled from a probabilistic model characterized by $\boldsymbol{\alpha}$;
  - Assemble the circuit according to $\mathbf{k}$;
  - Compute the loss, then the gradient of $\boldsymbol{\theta}$ and $\boldsymbol{\alpha}$;
  - Update $\boldsymbol{\theta}$ and $\boldsymbol{\alpha}$;
  - Repeat until converges;
  - Get $\mathbf{k}^*$with highest probability from $P(\mathbf{k}, \boldsymbol{\alpha})$, and fine-tune the corresponding parameters $\boldsymbol{\theta}^*$

Algorithm 1 Differentiable Quantum Architecture Search.

Require: p as the number of components to build the circuit; operation pool with c distinct unitaries; probabilistic model and its parameters α with shape $p \times c$ initialized all to zero; resuing parameter pool θ initialized with given initializer with shape $p \times c \times l$, where l is the max number of parameters of each op in operation pool.

while not converged **do**
    Sample a batch of size K circuits from model $P(k, \alpha)$.
    Compute the objective function for each circuit in the batch in the form of Eq. (2), Eq. (4), Eq. (5) depending on different problem settings.
    Compute the gradient with respect to $\boldsymbol{\theta}$ and $\boldsymbol{\alpha}$ according to Eq. (7) and Eq. (8), respectively.
    Update $\boldsymbol{\theta}$ and $\boldsymbol{\alpha}$ using given gradient based optimizers and learning schedules.
end while
Get the circuit architecture $\boldsymbol{k}^*$ with the highest probability in $P(\boldsymbol{k}, \boldsymbol{\alpha})$; and fine tuning the circuit parameters as $\boldsymbol{\theta}^*$ associated with this circuit if necessary.
return final optimal circuit structure labeled by $\boldsymbol{k}^*$ and the associating weights $\boldsymbol{\theta}^*$.

Figure 1: DQAS algorithm from [ZHZY20]

- Gradients:
  - $\nabla_{\boldsymbol{\theta}} \mathcal{L} = \sum_{\mathbf{k} \sim P(\mathbf{k}, \boldsymbol{\alpha})} \nabla_{\boldsymbol{\theta}} \mathcal{L}(U(\mathbf{k}, \boldsymbol{\theta}))$. Not all parameters will appear in $\mathcal{L}$. $\boldsymbol{\theta}$ in shape $p \times c \times d$, where p is the total number of unitaries, c is the size of operation pools and l is the number of parameters for each unitary in the operation pool. For the missing parameters, the gradients will be set to zero.
  - $\nabla_\alpha \mathcal{L} = \sum_{k \sim P} \nabla_\alpha \ln P(\boldsymbol{k}, \boldsymbol{\alpha}) L(U(\boldsymbol{k}, \boldsymbol{\theta})) - \sum_{\boldsymbol{k} \sim P} L(U(\boldsymbol{k}, \boldsymbol{\theta})) \sum_{\boldsymbol{k} \sim P} \nabla_{\boldsymbol{\alpha}} \ln P(\boldsymbol{k}, \boldsymbol{\alpha})$ (*This still needs more looking into.*)
  - For normalized probability distributions, $\langle \nabla_{\boldsymbol{\alpha}} \ln P \rangle_P = 0$, then we simply only need to focus on the first term.
- Probabilistic model: $P(\boldsymbol{k}, \boldsymbol{\alpha}) = \prod_i^p p(k_i, \boldsymbol{\alpha_i})$, where $p(k_i = j, \boldsymbol{\alpha_i}) = \frac{e^{\alpha_{ij}}}{\sum_k e^{\alpha_{ik}}}$, where $k_i = j$ means that we picked $U_i = V_j$, $\boldsymbol{\alpha}$ has shape $p \times c$. The gradient of this particular probabilistic model can be determined analytically: $\nabla_{\alpha_{ij}} \ln P(k_i = m) = -P(k_i = m) + \delta_{jm}$;

- Possible improvements:
  - Multiple starts;
  - Parameter prethermalization;
  - Early stopping;
  - Top-k grid search or beam search;
  - Baseline for score function estimators: $L = L - \bar{L}$, where $\bar{L}$ is the average of objective from the last evaluated batch;
  - Random noise on parameter θ;
  - Regularization and penalty terms.

3.2 For Our Task

- Operation pool: {U3 on each qubit, CU3 on adjacent qubits};
- Same probability model;
- Cost function: Encode seven states, six Pauli eigenstates plus T state
  - The average overlap between the encoded states and 5-qubit code words;
  - Recoverable quantum information in [FTWM18].

3.3 Possible Obstacles

- The assumption of the probabilistic model, which indicates that the distributions of the operations in each layer are independent from each other, may not suitable for our circuit;
- Gradient calculation of the probabilistic model may be tricky during implementation.

4 Expected Results

The possible results, from the first cost function, would be an encoding circuit for the five-qubit error correction code that can produce high-fidelity encoded states under noise.

We could also start from start from smaller circuits, like the 3-qubit bit-flip code and only X errors.

5 Other Possible Approach

It is still possible to use RL to search for the target quantum circuits, like the one-shot quantum architecture search in [DHY^{+}20]. (*Still under investigation*)

References

[DHY+20] Yuxuan Du, Tao Huang, Shan You, Min-Hsiu Hsieh, and Dacheng Tao. Quantum circuit architecture search: error mitigation and trainability enhancement for variational quantum solvers, 2020.

[FTWM18] Thomas Fösel, Petru Tighineanu, Talitha Weiss, and Florian Marquardt. Reinforcement learning with neural networks for quantum feedback. *Phys. Rev. X*, 8:031084, Sep 2018.

[JRO+17] Peter D. Johnson, Jonathan Romero, Jonathan Olson, Yudong Cao, and Alán Aspuru-Guzik. Qvector: an algorithm for device-tailored quantum error correction, 2017.

[XBY21] Xiaosi Xu, Simon C. Benjamin, and Xiao Yuan. Variational circuit compiler for quantum error correction. *Phys. Rev. Applied*, 15:034068, Mar 2021.

[ZHZY20] Shi-Xin Zhang, Chang-Yu Hsieh, Shengyu Zhang, and Hong Yao. Differentiable quantum architecture search, 2020.