

IBM Mentorship Program 2022

#2 VQE optimization with dynamic shot
scheduling

Showcase : June 2022

VQE optimization with dynamic shot scheduling

In current near term quantum era, variational algorithms (like VQE) can be useful, but:

- Shots are expensive in a real quantum computer
- Noise is a problem

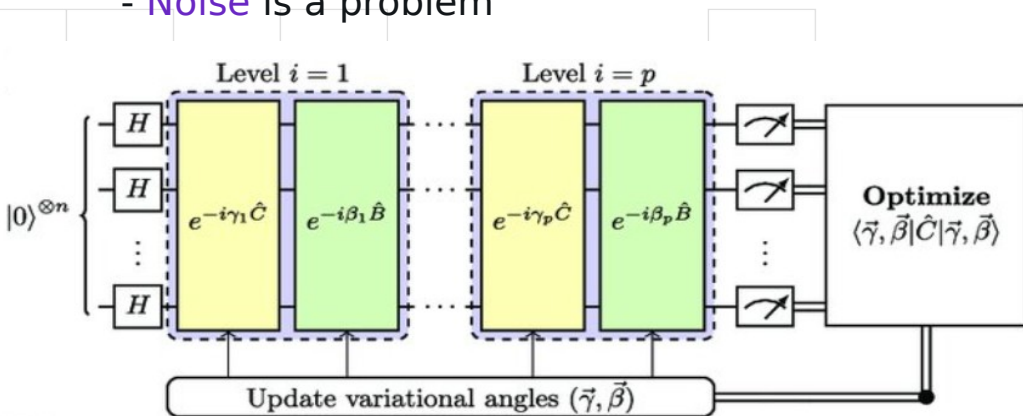
Approaches to maximize performance:

- Ansatz
- Cost function
- Gradient and other optimization methods
- Adaptive number of shots

We will implement adaptive number of shots method

⇒ More noise, more shots

Kubler et al. [1], cites adaptive shots is the most efficient optimizer, looking at the total number of shots



Team:

Mentee: Arnaldo Gunzi



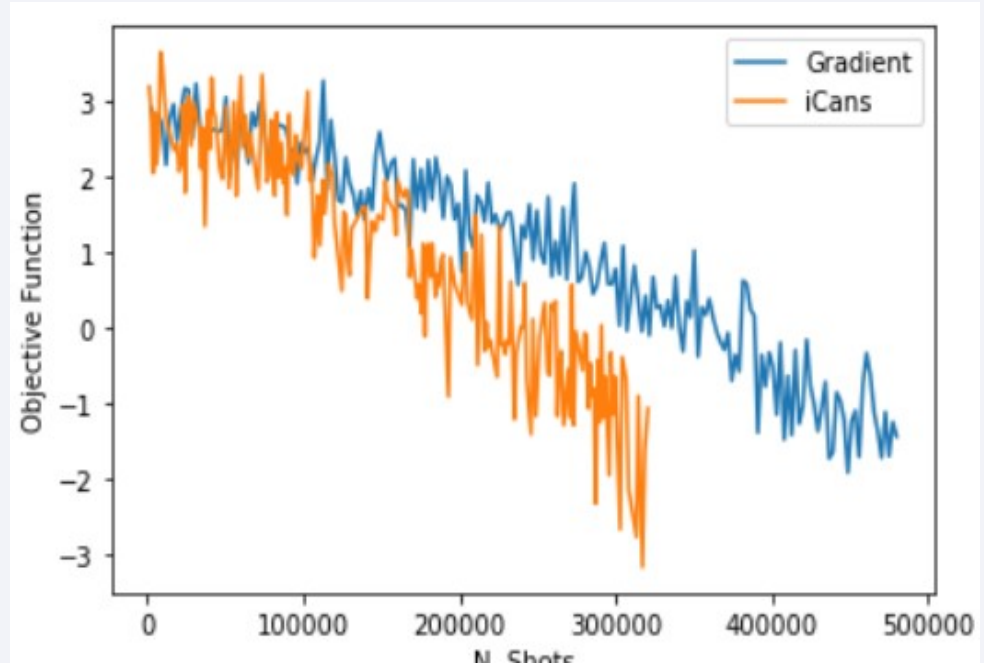
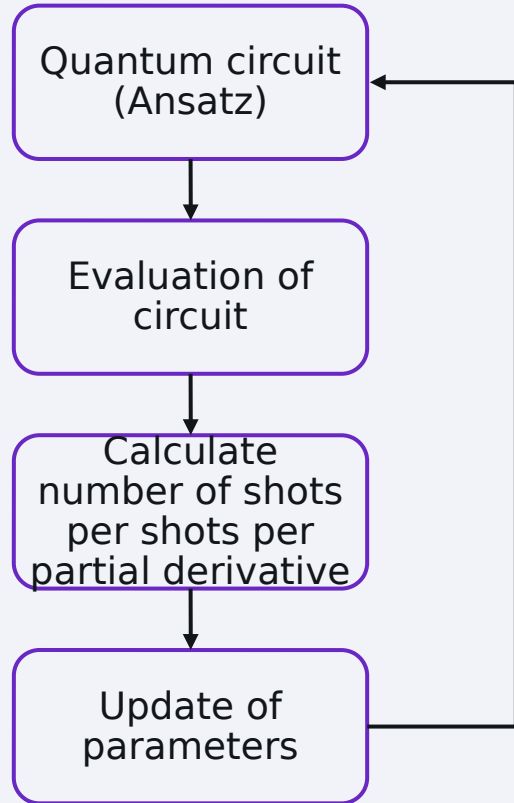
Mentor: Julien Gacon



Deliverables:

- iCANS as new subclass of VQE in Qiskit
- Tutorial on the new optimizer

Fluxogram of the algorithm and results



We did tests comparing ICANS to normal gradient and SPSA, and it behaved accordingly to the paper in references

Current status


Qiskit / qiskit-terra Public 3

<> Code Issues 622 Pull requests 193 Actions Projects 11 Security Insights

Implement the ICANS algorithm #8155

Draft asgunzi wants to merge 12 commits into `Qiskit:main` from `asgunzi:icans`

Conversation 0 Commits 12 Checks 1 Files changed 8

 asgunzi commented 5 minutes ago

Summary

Implement the ICANS algorithm as a sub-class of VQE.

<https://arxiv.org/abs/1909.09083>

Details and comments

The ICANS optimizer differently select number of shots according to the variance in the measurements. It reduces the number of shots required for minimization. This is part of Qiskit mentorship program.

TO DO:

- Tests
- Release note



Attachments

iCANs algorithm

Input: Learning rate α , starting point θ_0 , min number of shots per estimation s_{\min} , number of shots that can be used in total N , Lipschitz constant L , running average constant μ , bias for gradient norm b

- 1: initialize: $\theta \leftarrow \theta_0$, $s_{\text{tot}} \leftarrow 0$, $\mathbf{s} \leftarrow (s_{\min}, \dots, s_{\min})^T$, $\chi' \leftarrow (0, \dots, 0)^T$, $\xi' \leftarrow (0, \dots, 0)^T$, $k \leftarrow 0$
- 2: **while** $s_{\text{tot}} < N$ **do**
- 3: $\mathbf{g}, \mathbf{S} \leftarrow i\text{Evaluate}(\theta, \mathbf{s})$
- 4: $s_{\text{tot}} \leftarrow s_{\text{tot}} + 2 \sum_i s_i$
- 5: $\xi'_\ell \leftarrow \mu \xi'_\ell + (1 - \mu) \mathbf{S}_\ell$
- 6: $\chi'_\ell \leftarrow \mu \chi'_\ell + (1 - \mu) \mathbf{g}_\ell$
- 7: $\xi_\ell \leftarrow \xi'_\ell / (1 - \mu^{k+1})$
- 8: $\chi_\ell \leftarrow \chi'_\ell / (1 - \mu^{k+1})$

```
for  $i \in [1, \dots, d]$  do
  if iCANS1 then
     $\theta_i \leftarrow \theta_i - \alpha \mathbf{g}_i$ 
  else if iCANS2 then
    if  $\alpha \leq \frac{g_i^2}{L(g_i^2 + S_i/s_i + b\mu^k)}$  then
       $\theta_i \leftarrow \theta_i - \alpha \mathbf{g}_i$ 
    else
       $\alpha' \leftarrow \frac{g_i^2}{L(g_i^2 + S_i/s_i + b\mu^k)}$ 
       $\theta_i \leftarrow \theta_i - \alpha' \mathbf{g}_i$ 
    end if
  end if
   $s_i \leftarrow \left\lceil \frac{2L\alpha}{2-L\alpha} \frac{\xi_i}{\chi_i^2 + b\mu^k} \right\rceil$ 
   $\gamma_i \leftarrow \frac{1}{s_i} \left[ \left( \alpha - \frac{L\alpha^2}{2} \right) \chi_i^2 - \frac{L\alpha^2}{2s_i} \xi_i \right]$ 
end for
 $s_{\max} \leftarrow s_{\arg \max_i \gamma_i}$ 
 $\mathbf{s} \leftarrow \text{clip}(\mathbf{s}, s_{\min}, s_{\max})$ 
 $k \leftarrow k + 1$ 
```

References

[VQE optimization with dynamic shot scheduling · Issue #2 · qiskit-advocate/qamp-spring-22 · GitHub](#)

An Adaptive Optimizer for Measurement-Frugal Variational Algorithms

[Jonas M. Kübler](#), [Andrew Arrasmith](#), [Lukasz Cincio](#), [Patrick J. Coles](#)

[\[1909.09083\] An Adaptive Optimizer for Measurement-Frugal Variational Algorithms \(arxiv.org\)](#)

Qiskit tutorials:

<https://qiskit.org/documentation/tutorials.html#algorithms>

<https://qiskit.org/textbook/ch-applications/vqe-molecules.html>

VQE code:

https://github.com/Qiskit/qiskit-terra/blob/main/qiskit/algorithms/minimum_eigen_solvers/vqe.py

Monitoring VQE convergence: https://qiskit.org/documentation/tutorials/algorithms/02_vqe_convergence.html

Rosalin: <https://arxiv.org/pdf/2004.06252.pdf>

PennyLane tutorial: https://pennylane.ai/qml/demos/tutorial_rosalin.html