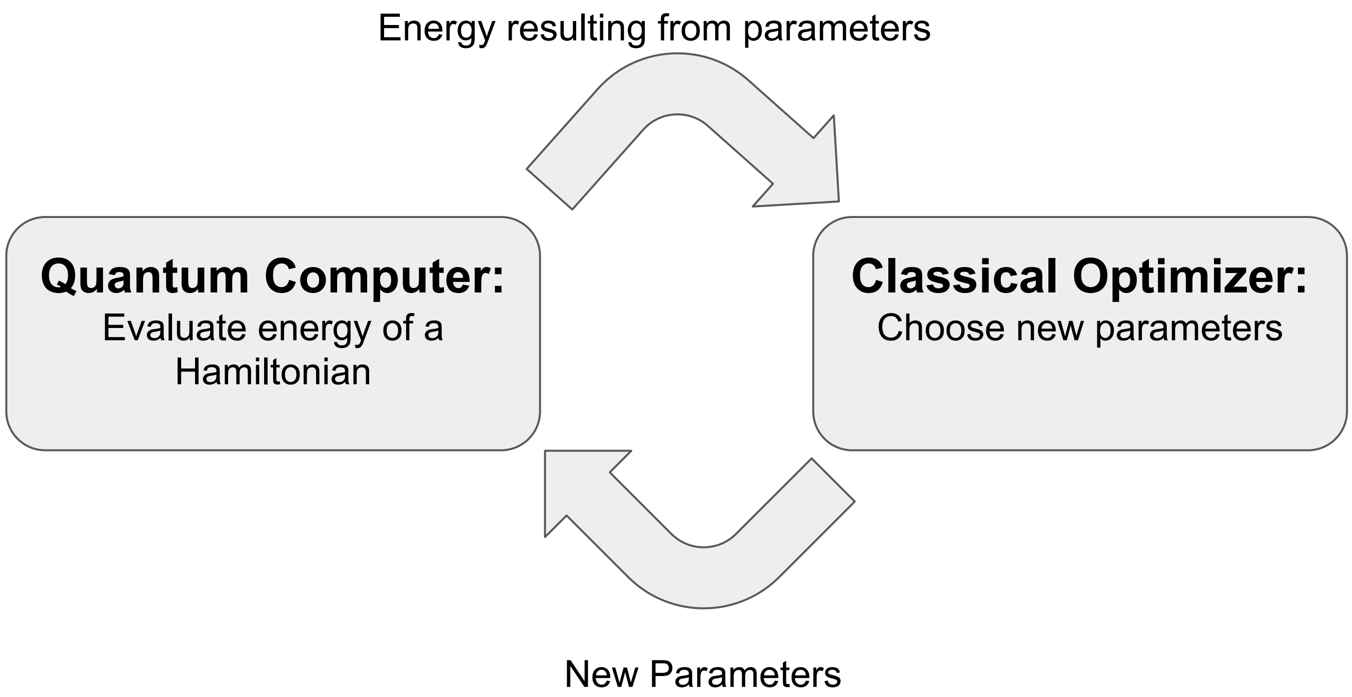
Generating Quantum Natural Finite Difference

#21

VQE: A hybrid quantum algorithm



Existing Aqua Optimizers

Qiskit / qiskit-aqua

Used by 83 | Watch 32 | Star 284 | Fork 208

Code | Issues 20 | Pull requests 17 | Actions | Projects 1 | Security | Insights

Branch: master | qiskit-aqua / qiskit / aqua / components / optimizers /

Create new file | Upload files | Find file | History

woodsp-ibm Update aqgd.py ... | Latest commit `fce19de` 7 days ago

..		
nlopts	additional docstring fixes	3 months ago
__init__.py	Complete PyLint on aqua folder	3 months ago
adam_amsgrad.py	Complete PyLint on aqua folder	3 months ago
aqgd.py	Update aqgd.py	7 days ago
cg.py	Complete PyLint on aqua folder	3 months ago
cobyla.py	Complete PyLint on aqua folder	3 months ago
l_bfgs_b.py	Complete PyLint on aqua folder	3 months ago
nelder_mead.py	Complete PyLint on aqua folder	3 months ago
optimizer.py	Complete PyLint on aqua folder	3 months ago
p_bfgs.py	Complete PyLint on aqua folder	3 months ago
powell.py	Complete PyLint on aqua folder	3 months ago
slsqp.py	Complete PyLint on aqua folder	3 months ago
spsa.py	Complete PyLint on aqua folder	3 months ago
tnc.py	Complete PyLint on aqua folder	3 months ago

Singular Points In Parameter space (BAD)

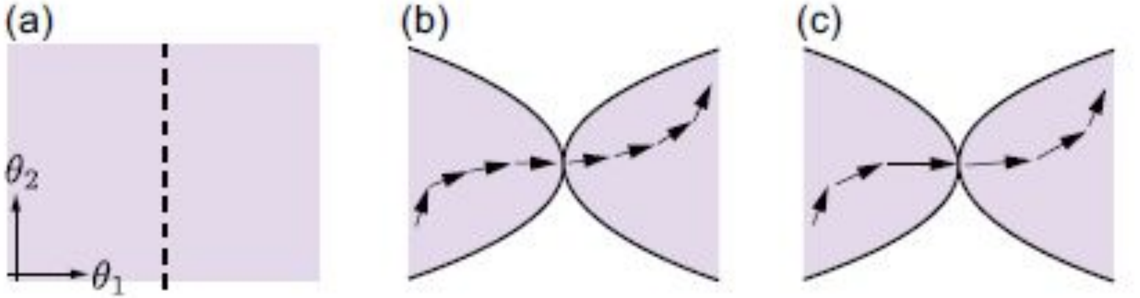


FIG. 1: Idea of natural gradient. (a) Parameter space; the dotted line represents the set of parameters where $f(\theta_1, \theta_2)$ takes the same value for all θ_2. (b) Trajectory of the parameters with Euclidean metric. (c) Trajectory of the parameters with non-Euclidean metric.

Taken from “On the natural gradient for variational quantum eigensolver - Naoki Yamamoto”

Quantum Natural Gradient (QNG) Optimizers

Pros

- Moves past singular points fast.
- Tailor made for quantum computers.

Cons

- Require 3d circuit evaluations per optimization step.
    - d is the number of parameters.

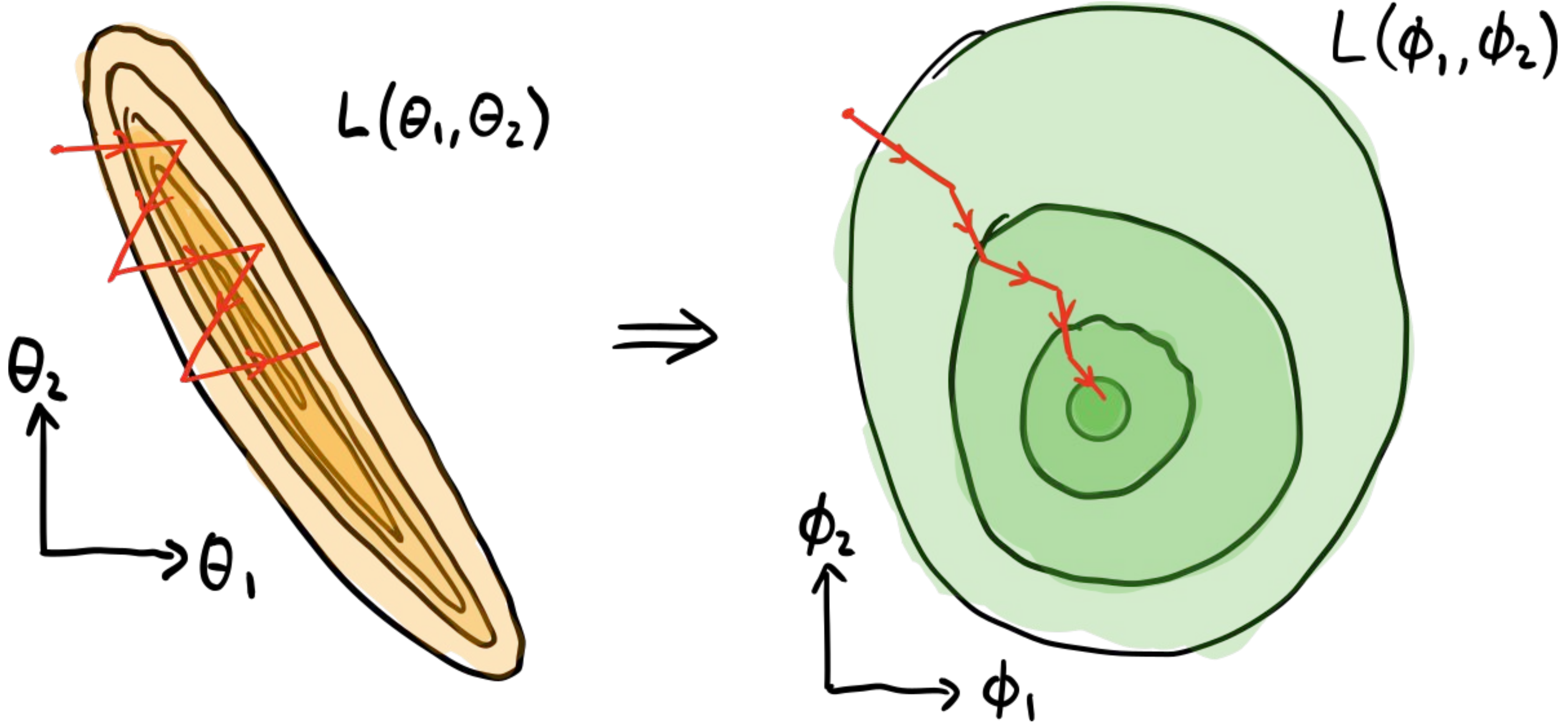
Extending the QNG

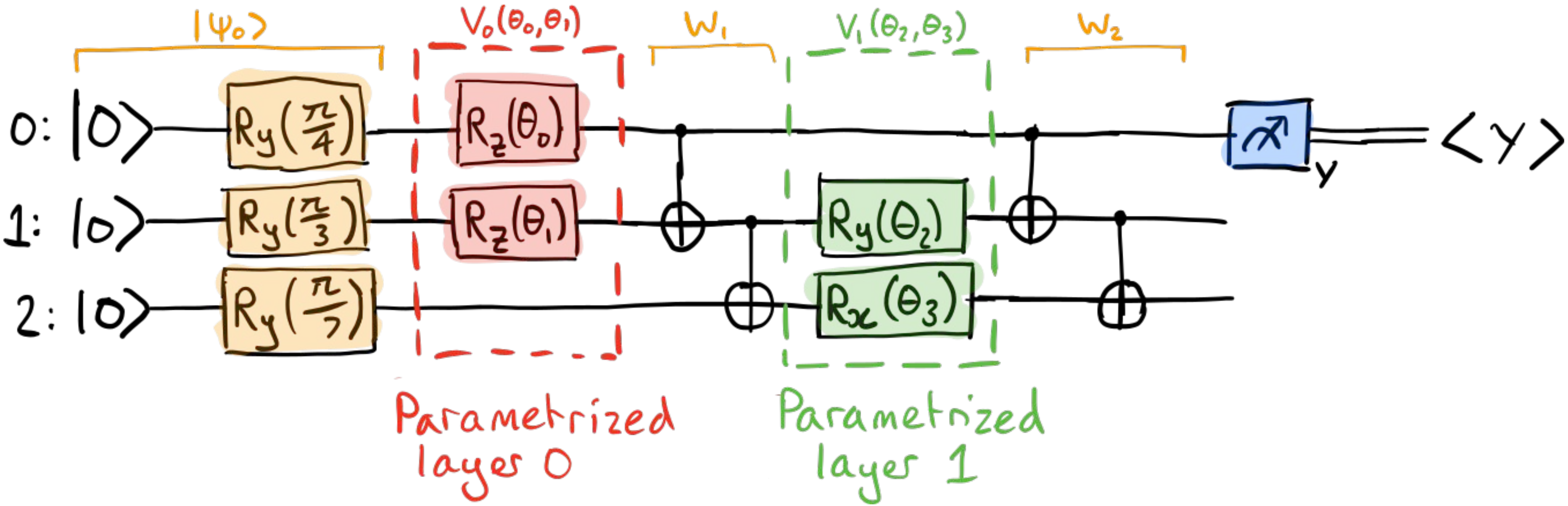
- QNG algorithm uses analytic gradients:
  - (requires 2d circuit evaluations to get those gradients)
- OUR optimizer: Quantum Natural Finite Differences:
  - Does not need analytic gradients

<u>Our approach needs only d evaluations! Rather than 3d</u>

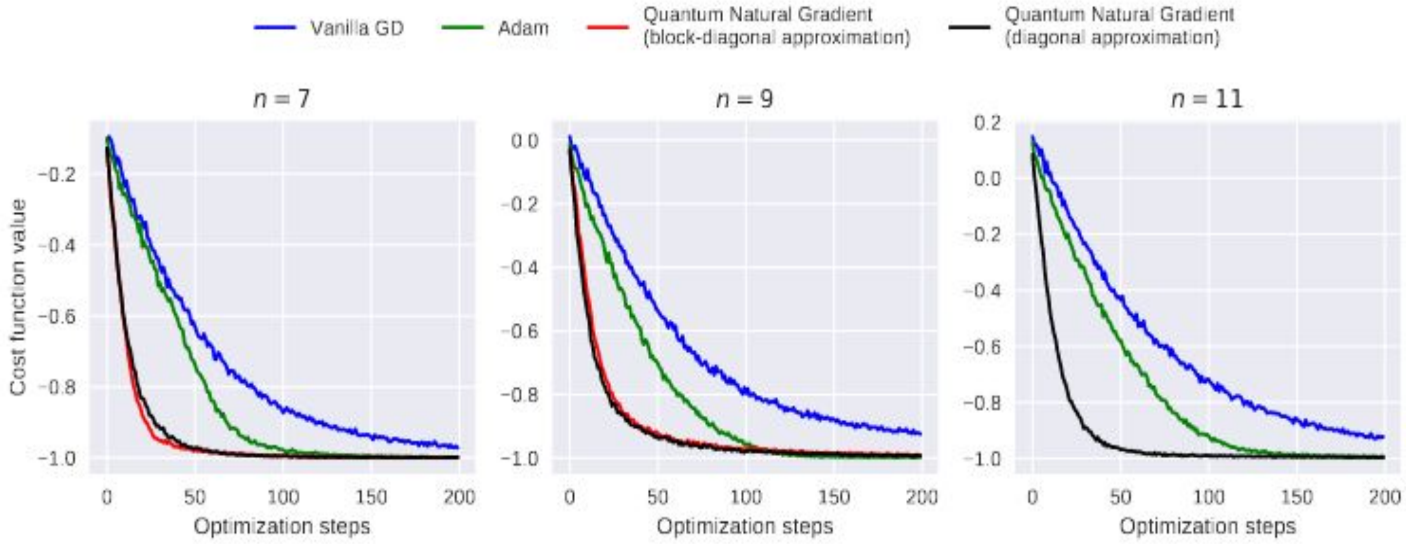
(d = number of parameters in our variational circuit)

	SPSA	QNFD	QNG
Error Robustness	Good!	Good!	Good!
Circuits run per optimization step	2	d+2	3d


$L(\theta_1, \theta_2)$
θ_2
θ_1
$\Longrightarrow$
$L(\phi_1, \phi_2)$
ϕ_2
ϕ_1

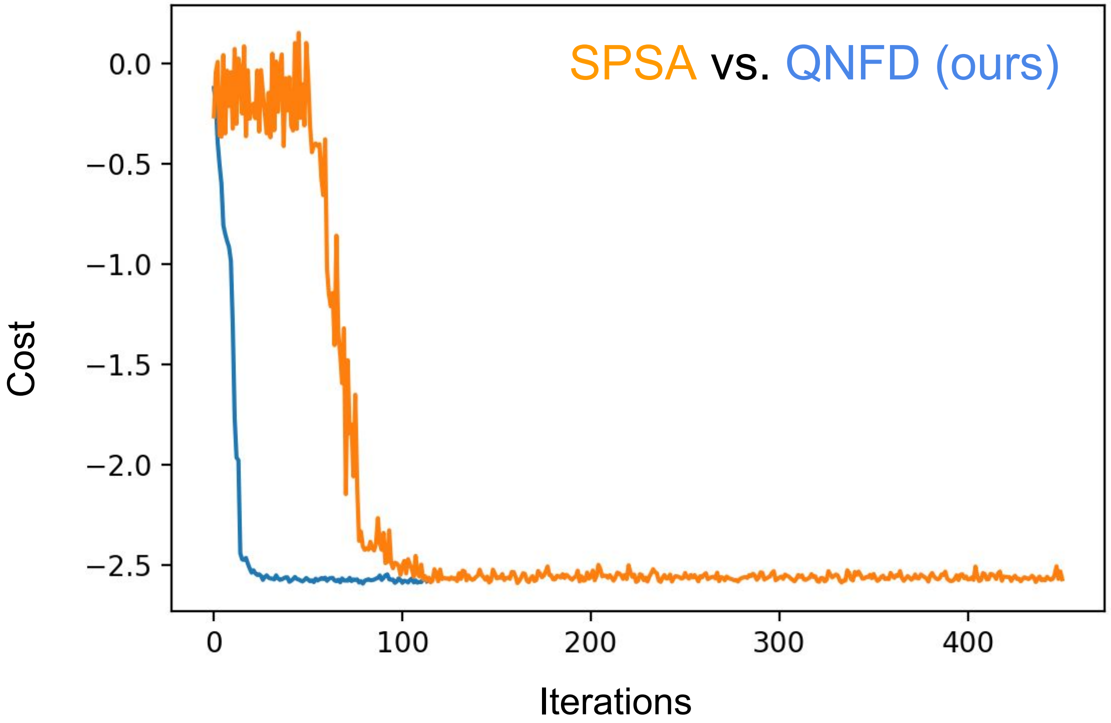


|ψ₀⟩
V₀(θ₀,θ₁)
W₁
V₁(θ₂,θ₃)
W₂
0: |0⟩
1: |0⟩
2: |0⟩
Ry(π/4)
Ry(π/3)
Ry(π/7)
Rz(θ₀)
Rz(θ₁)
Ry(θ₂)
Rx(θ₃)
y
⟨y⟩
Parametrized layer 0
Parametrized layer 1



This shows how the QNG converges faster than the rest optimizers.

Figure 4 :


SPSA vs. QNFD (ours)
Cost
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0
100
200
300
400
Iterations

APPLICATIONS

The possibility to train quantum algorithms (VQE) using gradient descent

In near-term noisy devices it may be of interest to study the relevant geometry for density matrices.

References

[1]. James Stokes, Josh Izaac, Nathan Killoran, Giuseppe Carleo. “Quantum Natural Gradient.” arXiv:1909.02108, 2019.