Funded by the European Union

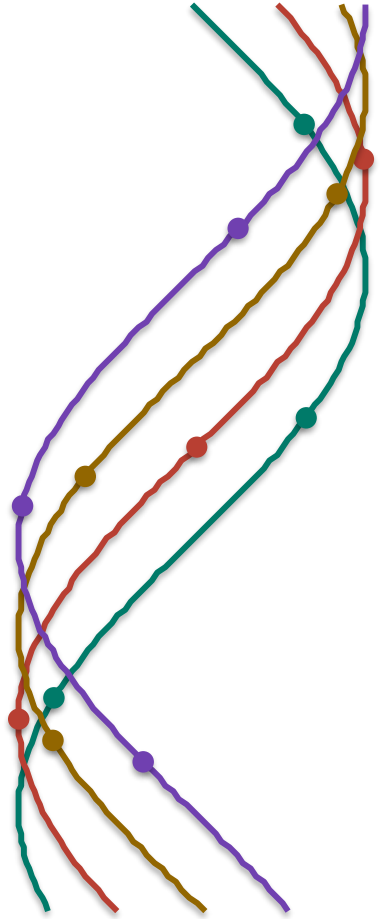
AQTIVATE

TECHNISCHE UNIVERSITÄT BERLIN

UNIVERSITÀ DEGLI STUDI DI PADOVA

BIFOLD

Bayesian methods for Variational Quantum Eigensolvers



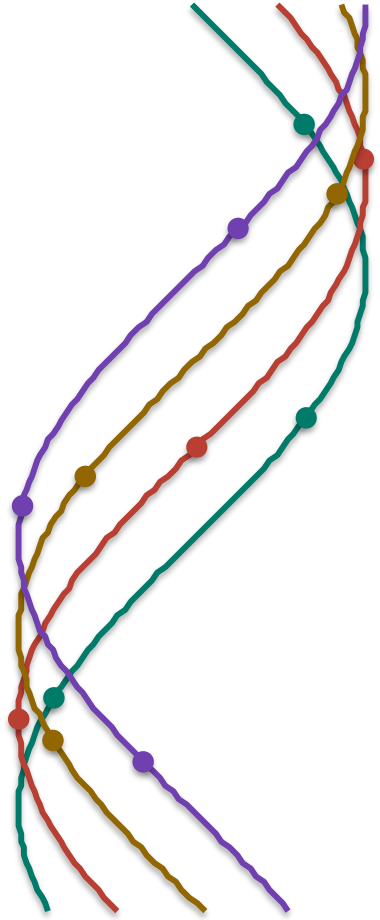
AQTIVATE CONFERENCE

29.09.2026

Samuele Pedrielli

BIFOLD · Technische Universität Berlin · Università di Padova

Context / Variational Quantum Eigensolver



Goal

Given a Hamiltonian, find its lowest-energy eigenstate.

$$H\ |\psi_0\rangle = E_0|\psi_0\rangle$$

E_0 : lowest possible energy. $|\psi_0\rangle$:corresponding ground state.

KNOWN: THE HAMILTONIAN H

It defines the problem & how energy is evaluated.

UNKNOWN: THE GROUND STATE

Its ground-state eigenvector is unknown.

Goal

Given a Hamiltonian, find its lowest-energy eigenstate.

$$H\,|\psi_0\rangle = E_0|\psi_0\rangle$$

E_0 : lowest possible energy. $|\psi_0\rangle$:corresponding ground state.

KNOWN: THE HAMILTONIAN H

It defines the problem & how energy is evaluated.

UNKNOWN: THE GROUND STATE

Its ground-state eigenvector is unknown.

VQE addresses this problem using *both* a quantum and a classical computer.

How

Prepare a parametrized state on a quantum computer:

$$|\psi(\theta)\rangle = U(\theta)|0\rangle$$

θ: adjustable parameters

Measure its energy on a on a quantum computer:

$$E(\theta) = \langle\psi(\theta)|H|\psi(\theta)\rangle \geq E_0$$

How

Prepare a parametrized state on a quantum computer:

$$|\psi(\theta)\rangle = U(\theta)|0\rangle$$

θ: adjustable parameters

Measure its energy on a on a quantum computer:

$$E(\theta) = \langle\psi(\theta)|H|\psi(\theta)\rangle \geq E_0$$

Minimize the energy on a classical computer:

$$\theta^* = \underset{\theta}{\operatorname{argmin}}\, E(\theta)$$

Prepare → measure → update θ

Finite shots give noisy,
costly energy estimates.

How

Prepare a parametrized state on a quantum computer:

$$|\psi(\theta)\rangle = U(\theta)|0\rangle$$

θ: adjustable parameters

Measure its energy on a on a quantum computer:

$$E(\theta) = \langle\psi(\theta)|H|\psi(\theta)\rangle \geq E_0$$

Minimize the energy on a classical computer:

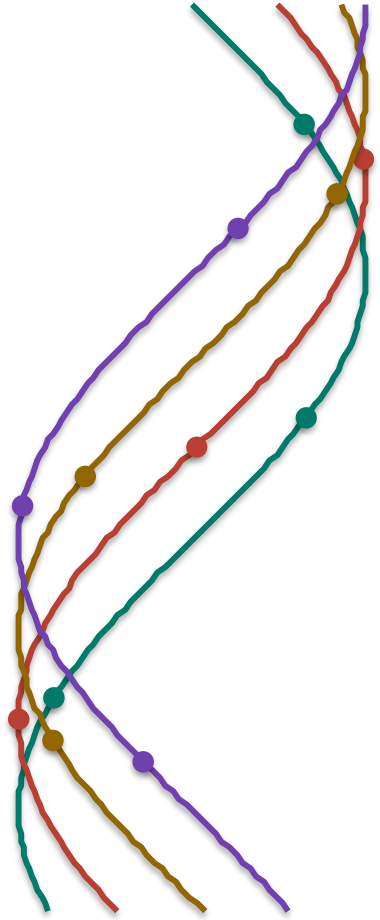
$$\theta^* = \underset{\theta}{\operatorname{argmin}}\, E(\theta)$$

Prepare → measure → update θ

Finite shots give noisy,
costly energy estimates.

We focus on the classical optimization part of the algorithm.

Context / Optimization Strategies



Two ways to optimize VQE

Sequential Minimal Optimization

$$\theta_d^* = \underset{\theta_d}{\operatorname{argmin}}\, E_d(\theta_d)$$

Look only at one parameter slice θ_d.
Minimize that slice.
Move to the next parameter θ_{d+1}.

Bias + regularization

Pedrielli et al. [2605.15813]

Gradient descent

$$\theta^* = \theta - \eta \hat{g}$$

Estimate the gradient $\hat{g}$.
Take a step set by the optimizer (ADAM).

GradCoRe

Pedrielli et al. [2502.02625]

Two ways to optimize VQE

Sequential Minimal Optimization

$$\theta_d^* = \underset{\theta_d}{\operatorname{argmin}}\, E_d(\theta_d)$$

Look only at one parameter slice θ_d.
Minimize that slice.
Move to the next parameter θ_{d+1}.

Bias + regularization

Pedrielli et al. [2605.15813]

Gradient descent

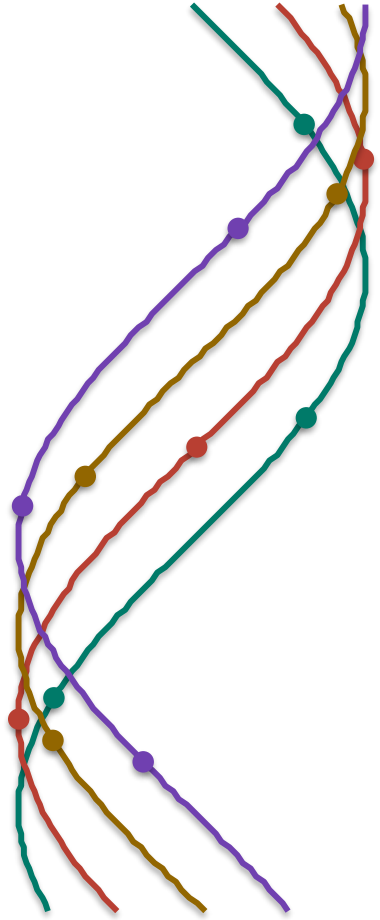
$$\theta^* = \theta - \eta \hat{g}$$

Estimate the gradient $\hat{g}$.
Take a step set by the optimizer (ADAM).

GradCoRe

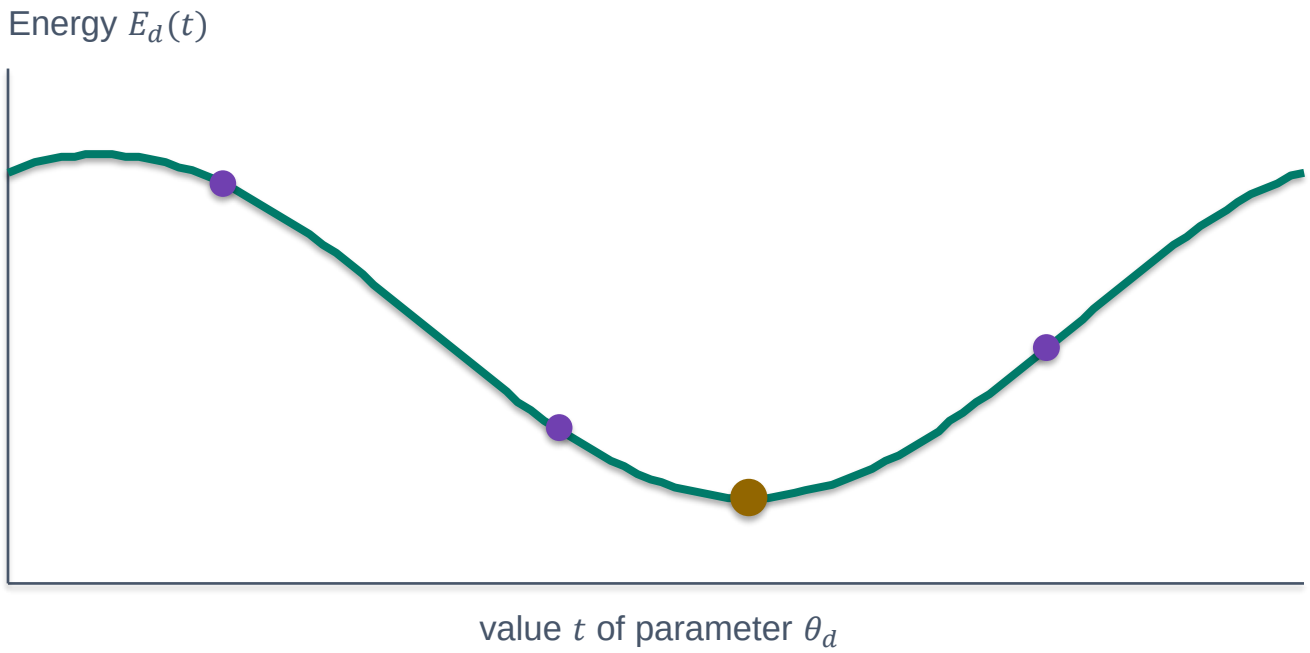
Pedrielli et al. [2502.02625]

Context / Sequential Minimal Optimization



Fix all but one angle → the energy is a sinusoid

$$E_d(t) = a + b \cdot \cos(t) + c \cdot \sin(t)$$



Nakanishi et al. [1903.12166]
Ostaszewski et al. [1905.09692]

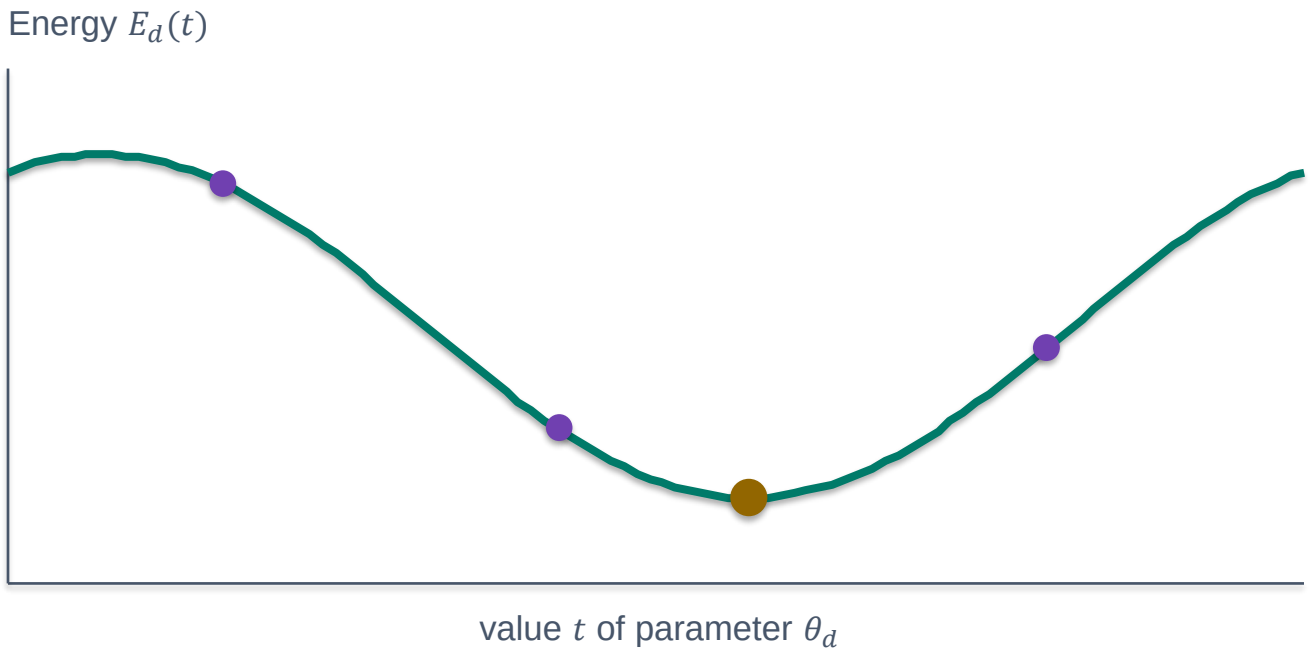
3 coefficients

3 distinct energy values

analytic parameter minimum

Fix all but one angle → the energy is a sinusoid

$$E_d(t) = a + b \cdot \cos(t) + c \cdot \sin(t)$$



Nakanishi et al. [1903.12166]
Ostaszewski et al. [1905.09692]

3 coefficients

3 distinct energy values

analytic parameter minimum

This sinusoidal slice makes exact coordinate minimization possible in the noisless case.

SMO-NFT repeatedly solves this one-dimensional problem

SMO: minimize one coordinate at a time. NFT: use the exact sinusoidal form.

NFT = Nakanishi–Fujii–Todo

1

Select parameter θ_d

All other angles stay fixed.

SMO-NFT repeatedly solves this one-dimensional problem

SMO: minimize one coordinate at a time. NFT: use the exact sinusoidal form.

NFT = Nakanishi–Fujii–Todo

1

Select parameter θ_d

All other angles stay fixed.

→

2

Fit the sinusoid

Using the energy evaluations.

→

SMO-NFT repeatedly solves this one-dimensional problem

SMO: minimize one coordinate at a time. NFT: use the exact sinusoidal form.

NFT = Nakanishi–Fujii–Todo

1

Select parameter θ_d

All other angles stay fixed.

→

2

Fit the sinusoid

Using the energy evaluations.

→

3

Minimize the slice

$\theta_d^* = \operatorname{atan2}(c, b) + \pi$

→

SMO-NFT repeatedly solves this one-dimensional problem

SMO: minimize one coordinate at a time. NFT: use the exact sinusoidal form.

NFT = Nakanishi–Fujii–Todo

1

Select parameter θ_d

All other angles stay fixed.

2

Fit the sinusoid

Using the energy evaluations.

3

Minimize the slice

$\theta_d^* = \operatorname{atan2}(c, b) + \pi$

4

Next coordinate

Repeat; a sweep visits all D.

SMO-NFT repeatedly solves this one-dimensional problem

SMO: minimize one coordinate at a time. NFT: use the exact sinusoidal form.

NFT = Nakanishi–Fujii–Todo

1

Select parameter θ_d

All other angles stay fixed.

→

2

Fit the sinusoid

Using the energy evaluations.

→

3

Minimize the slice

$\theta_d^* = \operatorname{atan2}(c, b) + \pi$

→

4

Next coordinate

Repeat; a sweep visits all D.

NFT reuse: previous fitted minimum → next center value

SMO-NFT repeatedly solves this one-dimensional problem

SMO: minimize one coordinate at a time. NFT: use the exact sinusoidal form.

NFT = Nakanishi–Fujii–Todo

1

Select parameter θ_d

All other angles stay fixed.

2

Fit the sinusoid

Using the energy evaluations.

3

Minimize the slice

$\theta_d^* = \operatorname{atan2}(c, b) + \pi$

4

Next coordinate

Repeat; a sweep visits all D.

NFT reuse: previous fitted minimum → next center value

> **In this way you need *two* energy evaluations instead of three.**

One NFT update: choose, measure, fit, move, repeat

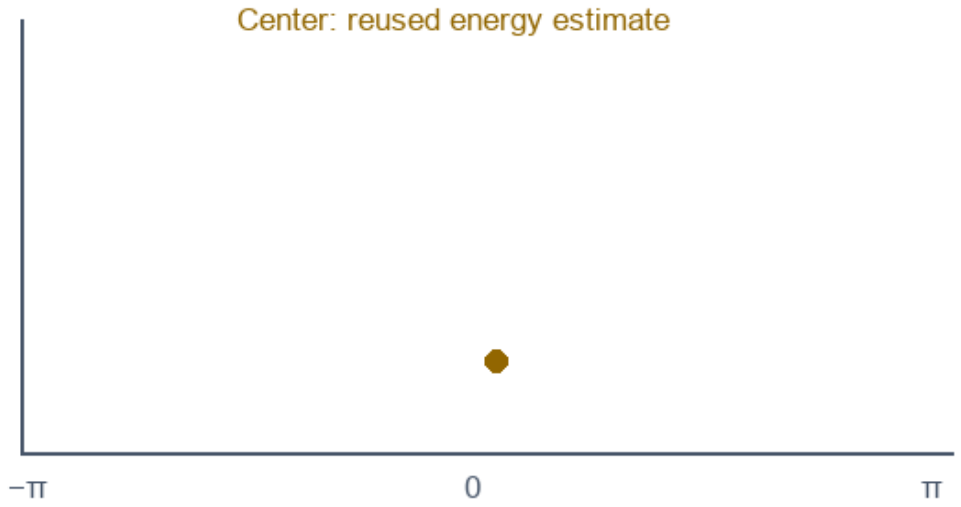
1 Start from the current point

Parameter space: one active coordinate



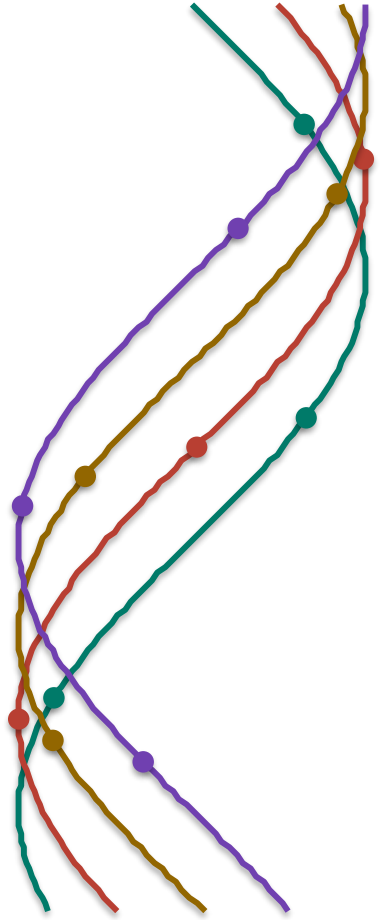
Only coordinate d is free

Energy along the active coordinate



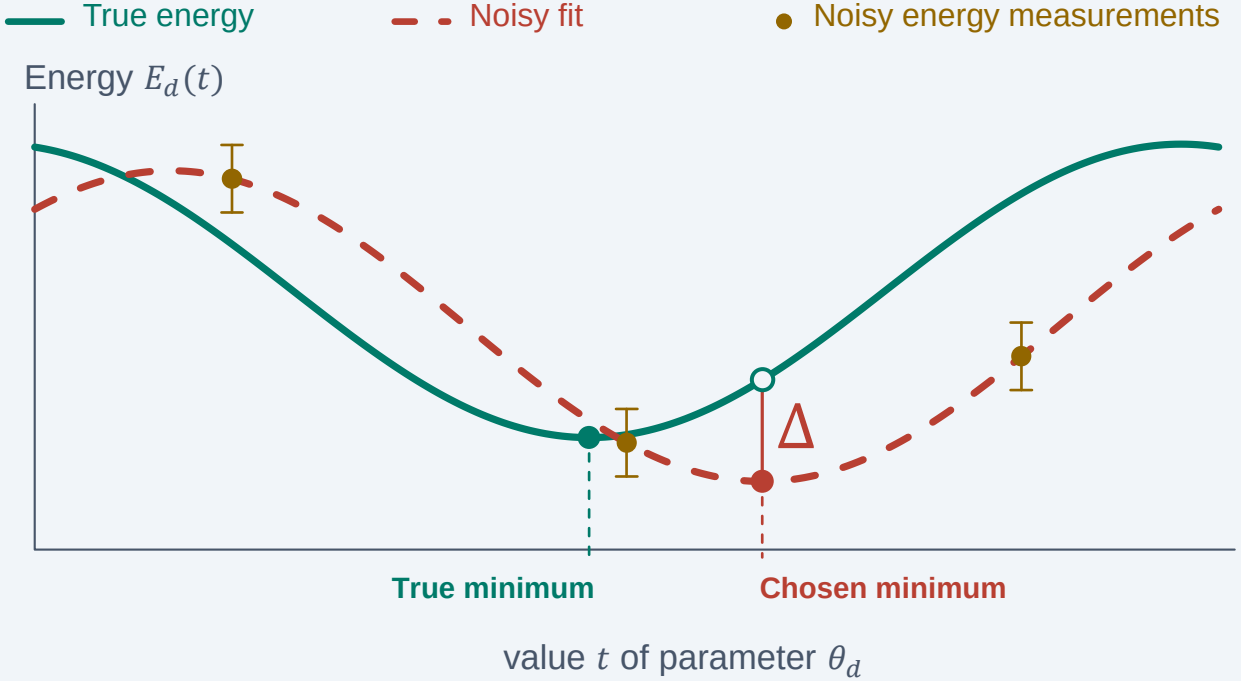
Illustrative NFT update · existing center estimate + two new energy evaluations · no benchmark data

Problem / SMO-NFT · Finite Measurements



Finite shots make the NFT update uncertain

Noisy energy samples perturb the sinusoid that SMO–NFT minimizes.

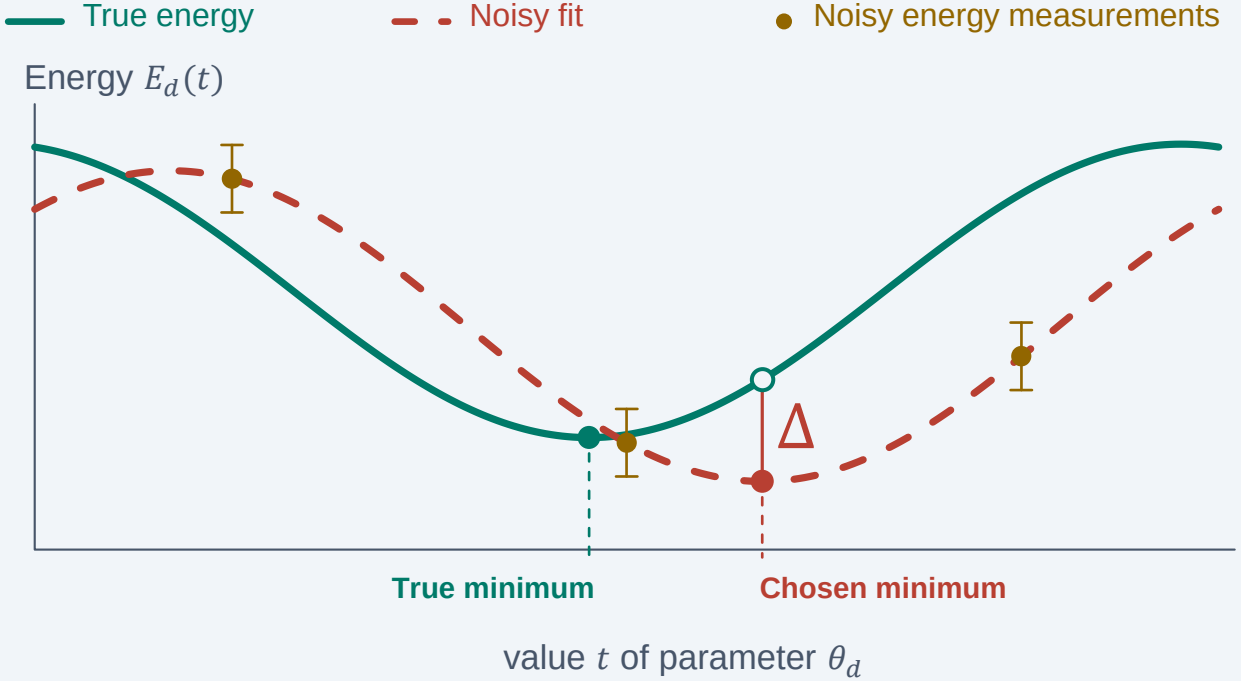


1 A noisy fit gives a noisy update

We need to come up with a resource allocation strategy that balances precision and cost

Finite shots make the NFT update uncertain

Noisy energy samples perturb the sinusoid that SMO–NFT minimizes.



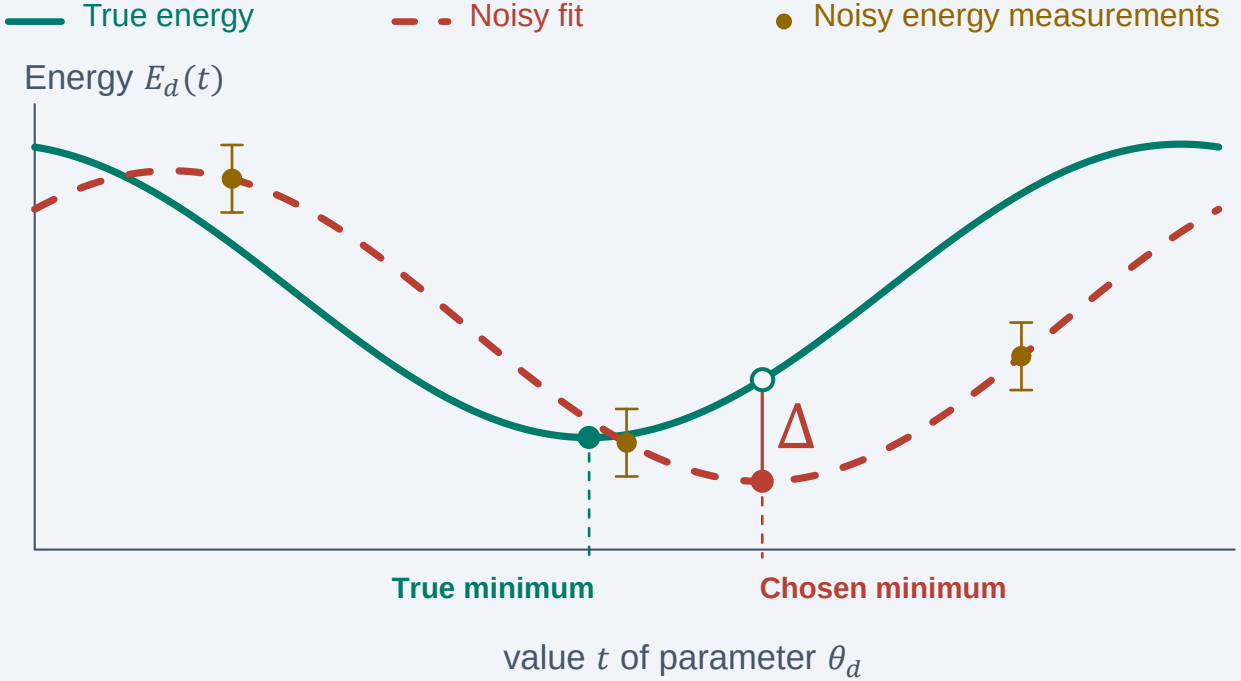
SubsCoRe **Anders et al. [2502.01704]**

1 A noisy fit gives a noisy update

We need to come up with a resource allocation strategy that balances precision and cost

Finite shots make the NFT update uncertain

Noisy energy samples perturb the sinusoid that SMO–NFT minimizes.



SubsCoRe **Anders et al. [2502.01704]**

1 A noisy fit gives a noisy update

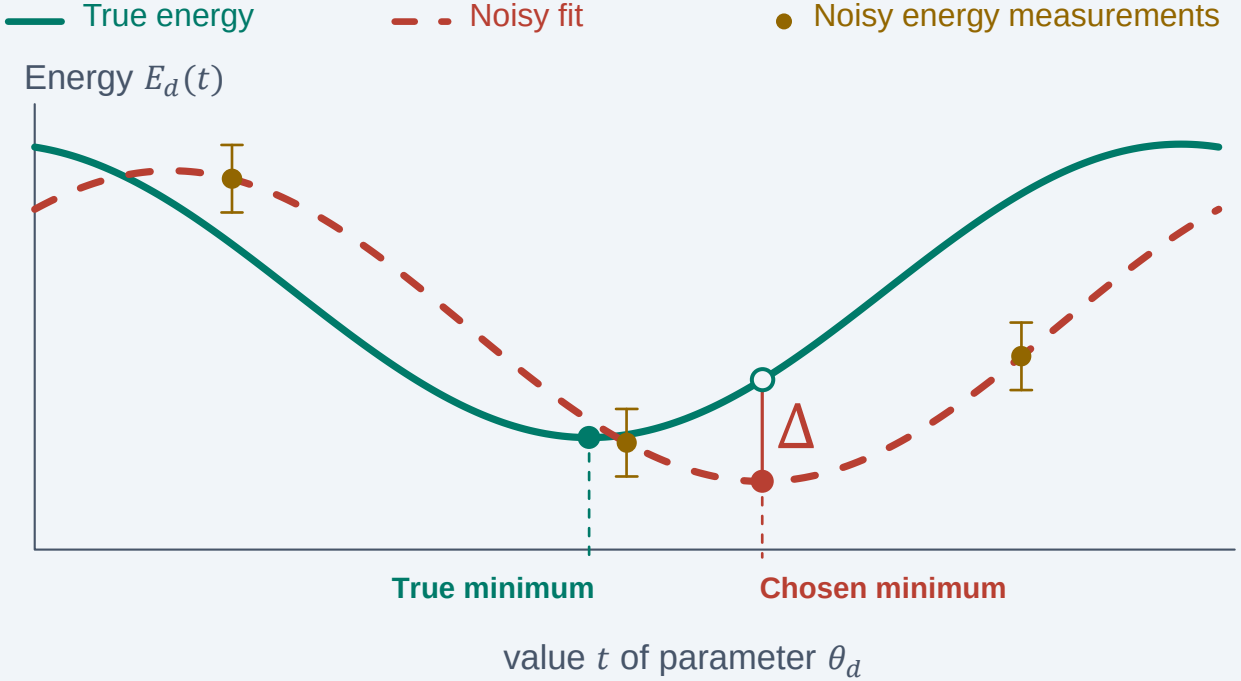
We need to come up with a resource allocation strategy that balances precision and cost

2 NFT reuse carries errors forward

The fitted minimum is too low on average. Reusing it propagates this bias.

Finite shots make the NFT update uncertain

Noisy energy samples perturb the sinusoid that SMO–NFT minimizes.



SubsCoRe **Anders et al. [2502.01704]**

1 A noisy fit gives a noisy update

We need to come up with a resource allocation strategy that balances precision and cost

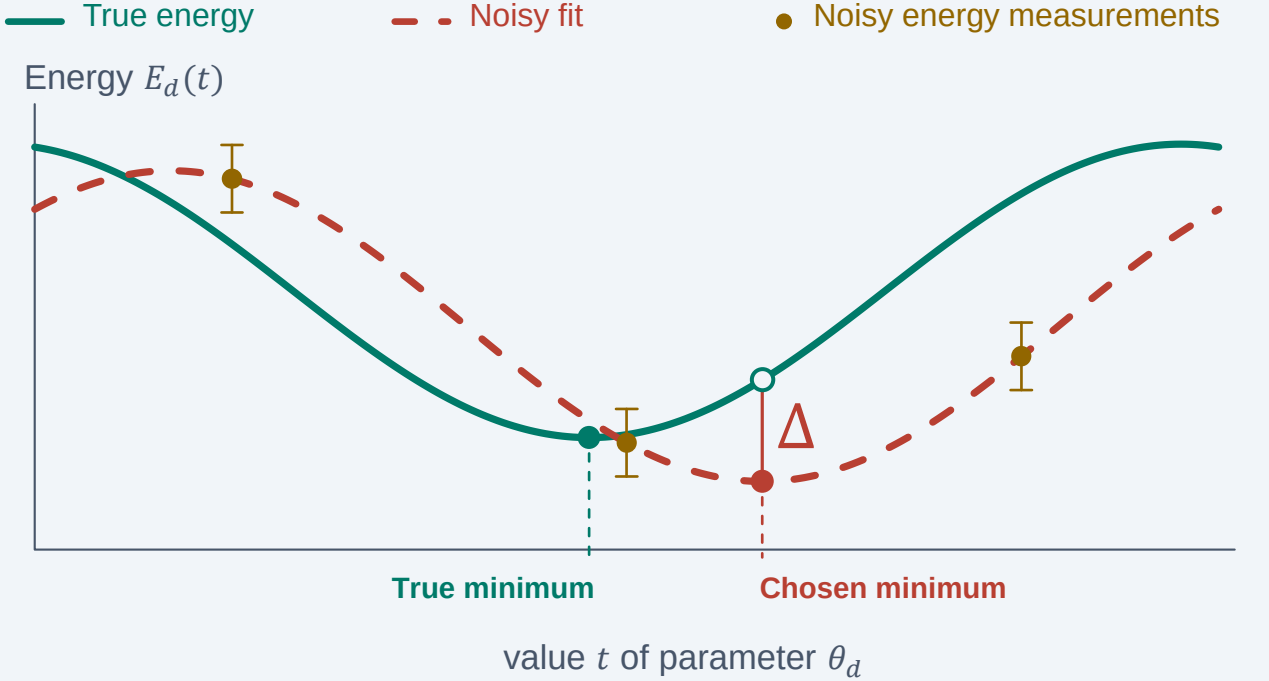
Bias **Pedrielli et al. [2605.15813]**

2 NFT reuse carries errors forward

The fitted minimum is too low on average. Reusing it propagates this bias.

Finite shots make the NFT update uncertain

Noisy energy samples perturb the sinusoid that SMO–NFT minimizes.



SubsCoRe **Anders et al. [2502.01704]**

1 A noisy fit gives a noisy update

We need to come up with a resource allocation strategy that balances precision and cost

Bias **Pedrielli et al. [2605.15813]**

2 NFT reuse carries errors forward

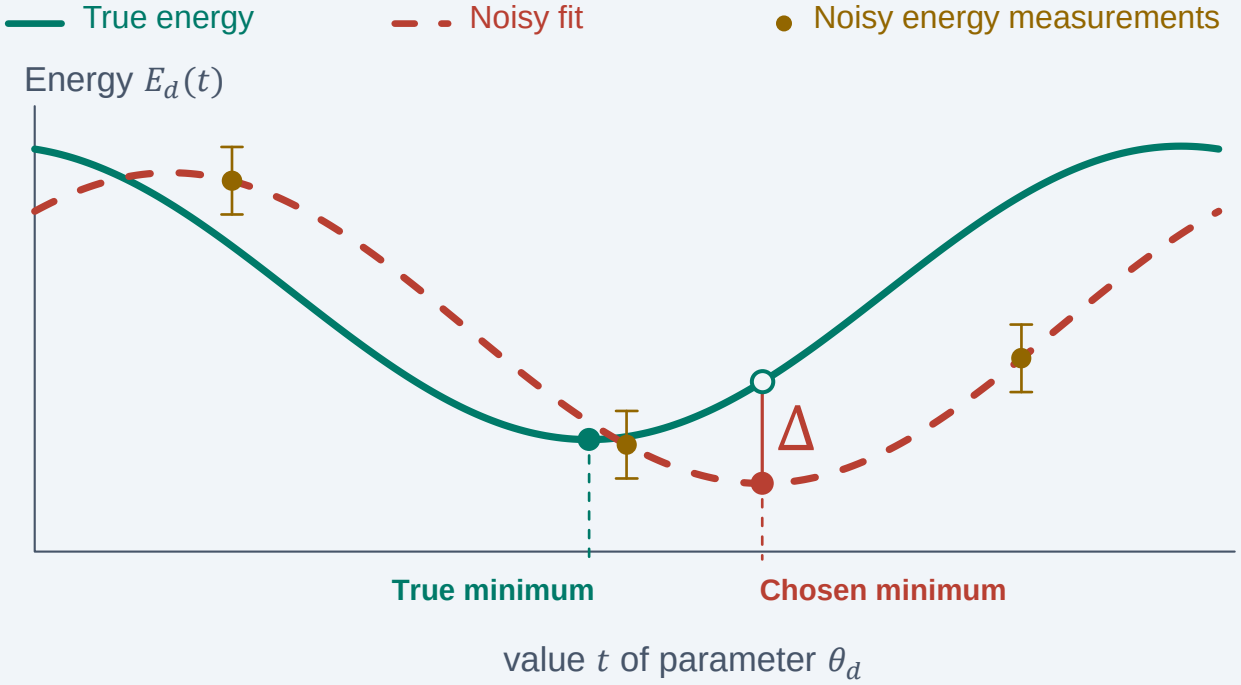
The fitted minimum is too low on average. Reusing it propagates this bias.

3 Where we measure matters

Can past updates guide the measurement shifts to estimate the minimizing angle more precisely?

Finite shots make the NFT update uncertain

Noisy energy samples perturb the sinusoid that SMO–NFT minimizes.



SubsCoRe **Anders et al. [2502.01704]**

1 A noisy fit gives a noisy update

We need to come up with a resource allocation strategy that balances precision and cost

Bias **Pedrielli et al. [2605.15813]**

2 NFT reuse carries errors forward

The fitted minimum is too low on average. Reusing it propagates this bias.

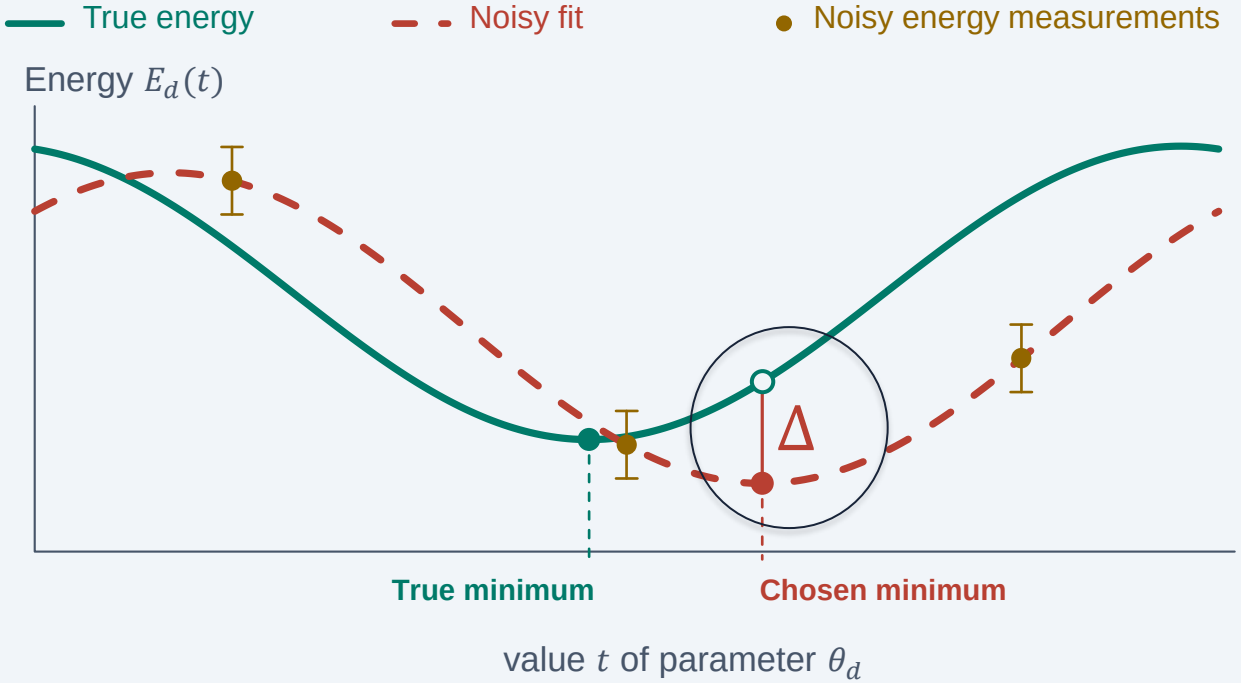
PAS **Stalschus et al. [2608.21616]**

3 Where we measure matters

Can past updates guide the measurement shifts to estimate the minimizing angle more precisely?

Finite shots make the NFT update uncertain

Noisy energy samples perturb the sinusoid that SMO–NFT minimizes.



SubsCoRe **Anders et al. [2502.01704]**

1 A noisy fit gives a noisy update

We need to come up with a resource allocation strategy that balances precision and cost

Bias **Pedrielli et al. [2605.15813]**

2 NFT reuse carries errors forward

The fitted minimum is too low on average. Reusing it propagates this bias.

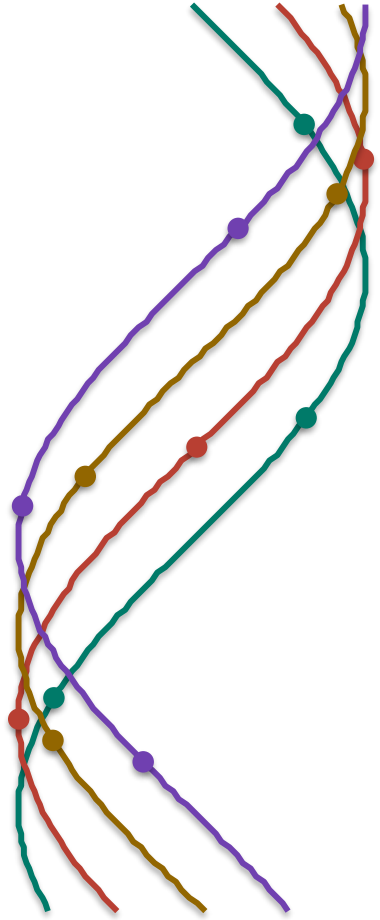
PAS **Stalschus et al. [2608.21616]**

3 Where we measure matters

Can past updates guide the measurement shifts to estimate the minimizing angle more precisely?

Bias: can we correct the energy estimate?

Our Work / SMO-NFT · Bias and Regularization

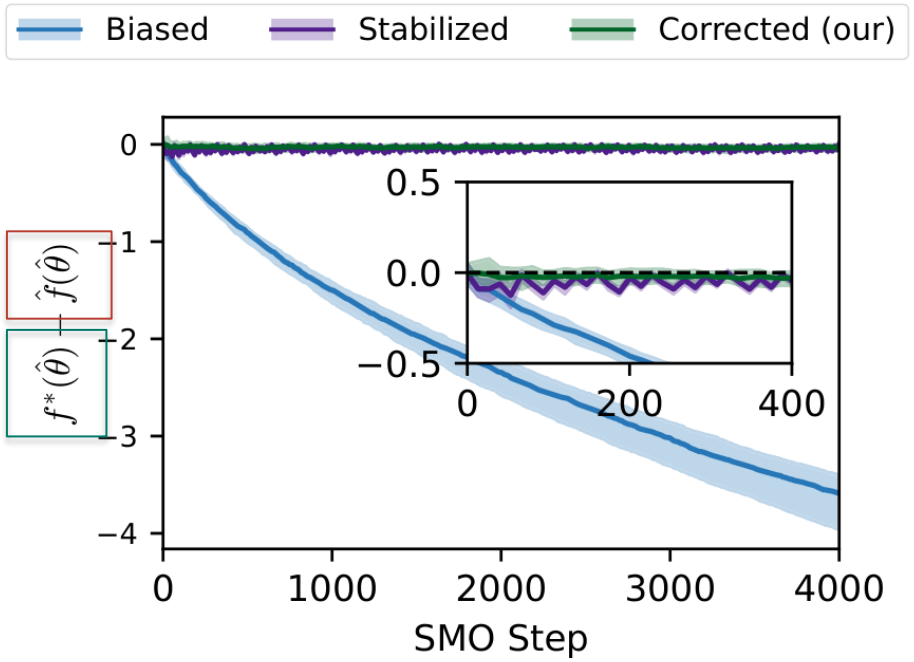


Energy reuse accumulates a systematic error

The reported energy gets lower and lower.

Estimated (finite meas.)

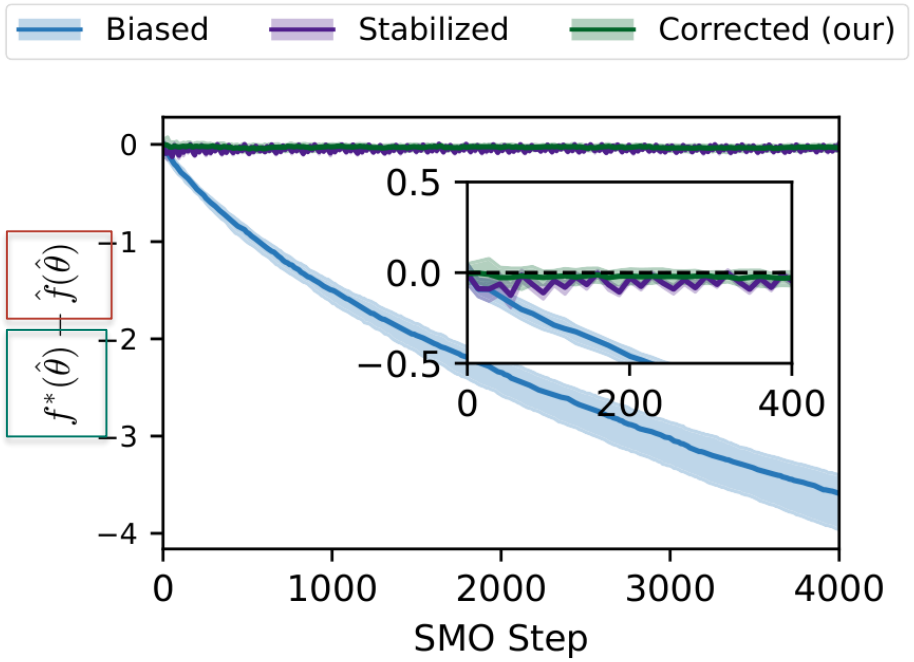
Exact (infinite meas.)



Pedrielli et al. · Bias, Fig. 1 · 100 trials

Energy reuse accumulates a systematic error

The reported energy gets lower and lower.



Estimated (finite meas.)

Exact (infinite meas.)

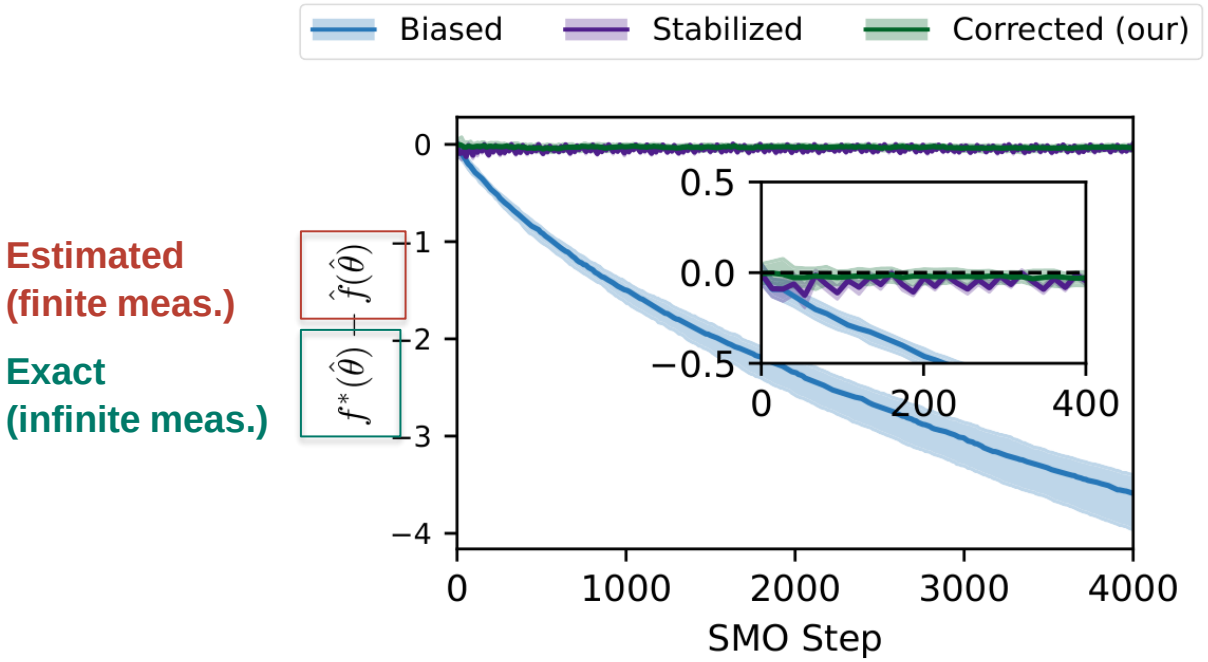
1 Predict the downward bias

Bayesian inference analysis → Δ

Pedrielli et al. · Bias, Fig. 1 · 100 trials

Energy reuse accumulates a systematic error

The reported energy gets lower and lower.



Pedrielli et al. · Bias, Fig. 1 · 100 trials

1 Predict the downward bias

Bayesian inference analysis → Δ

↓

2 Add back the predicted error

Corrected estimate

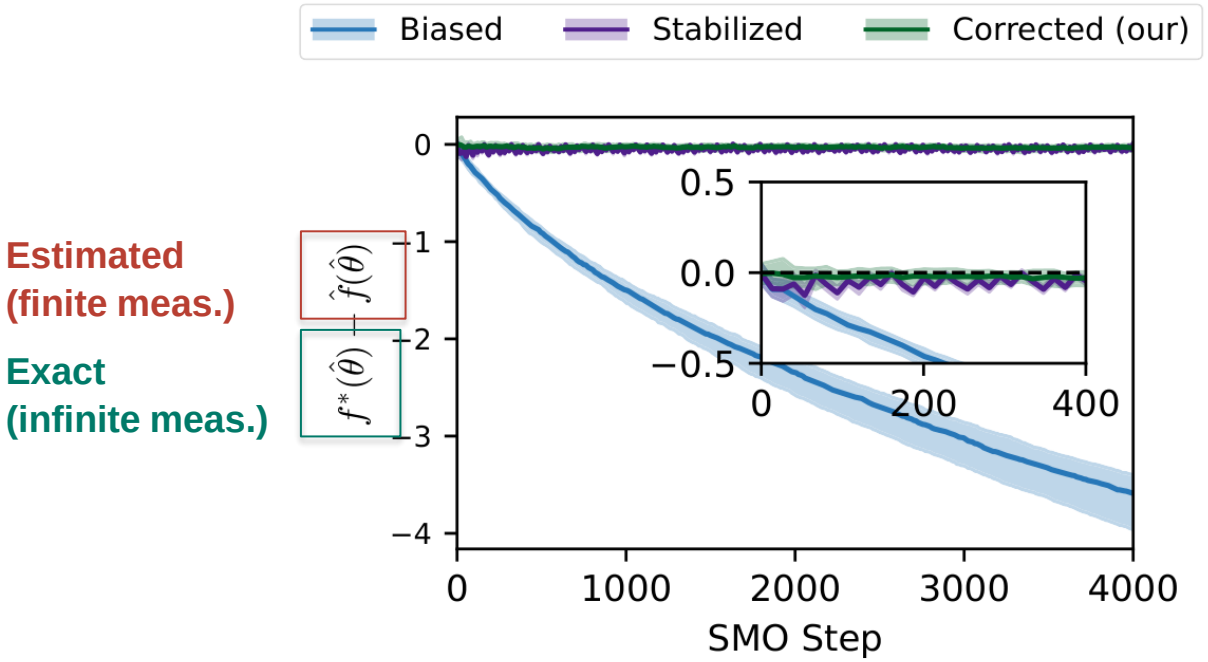
+ analytical correction Δ

Fitted minimum energy

↓

Energy reuse accumulates a systematic error

The reported energy gets lower and lower.



Pedrielli et al. · Bias, Fig. 1 · 100 trials

1 Predict the downward bias

Bayesian inference analysis → Δ

2 Add back the predicted error

Corrected estimate

+ analytical correction Δ

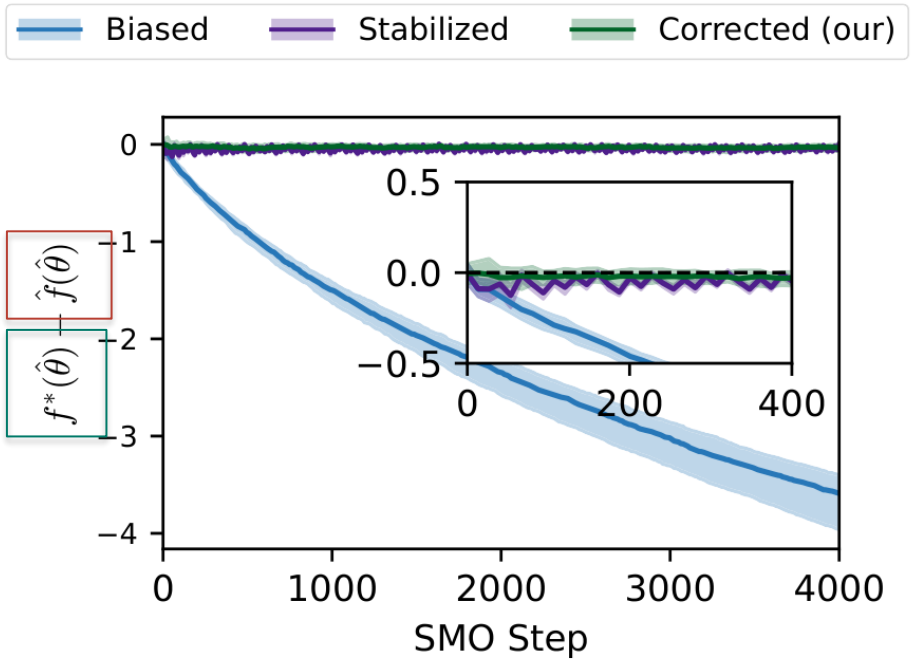
Fitted minimum energy

3 Reuse the corrected energy

Every update · no extra measurements

Energy reuse accumulates a systematic error

The reported energy gets lower and lower.



Estimated (finite meas.)

Exact (infinite meas.)

Pedrielli et al. · Bias, Fig. 1 · 100 trials

1 Predict the downward bias

Bayesian inference analysis → Δ

2 Add back the predicted error

Corrected estimate

+ analytical correction Δ

Fitted minimum energy

3 Reuse the corrected energy

Every update · no extra measurements

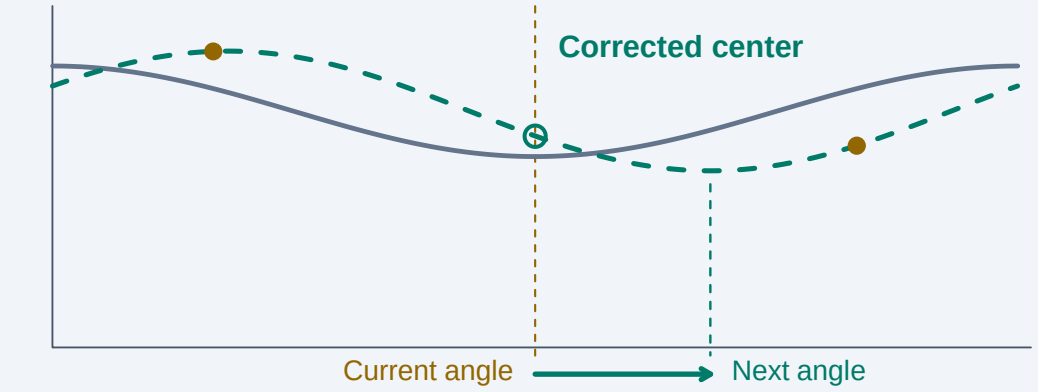
We can correct the leading bias, but this hinders performance. *Why?*

Correcting the estimate can destabilize the update

Near a flat minimum, a bias can suppress jumps driven by noise.

Correction

Use the corrected center in the fit.

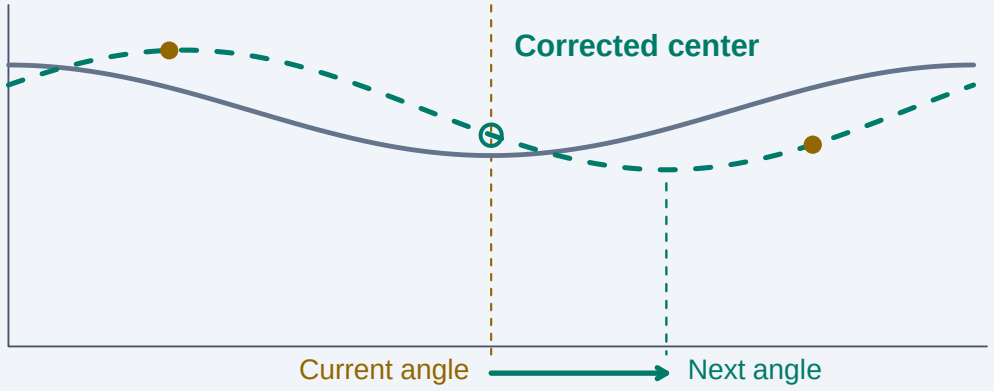


Correcting the estimate can destabilize the update

Near a flat minimum, a bias can suppress jumps driven by noise.

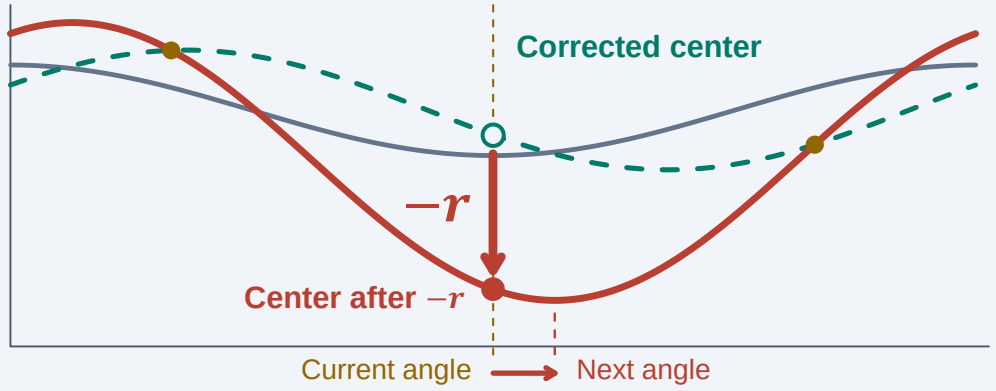
Correction

Use the corrected center in the fit.



Regularization

Lower the corrected center by r, then refit.



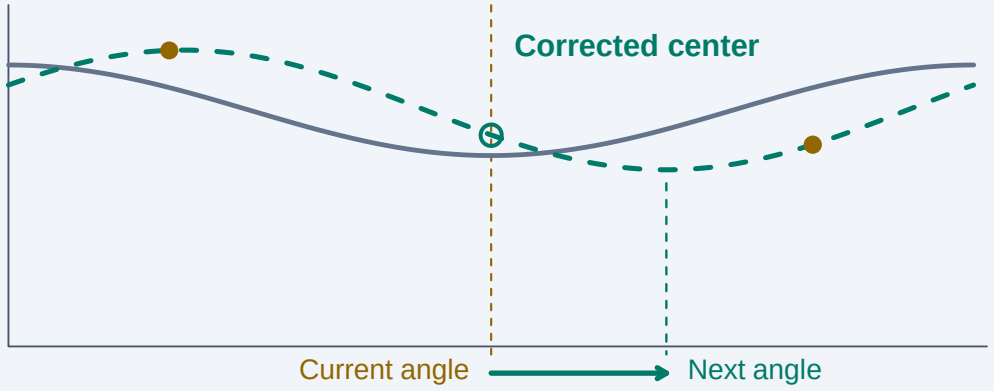
True energy Corrected energy Regularized energy

Correcting the estimate can destabilize the update

Near a flat minimum, a bias can suppress jumps driven by noise.

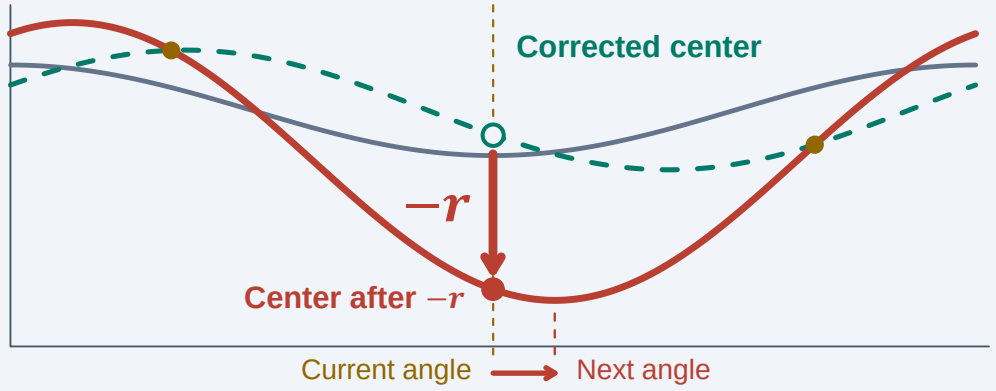
Correction

Use the corrected center in the fit.



Regularization

Lower the corrected center by r, then refit.

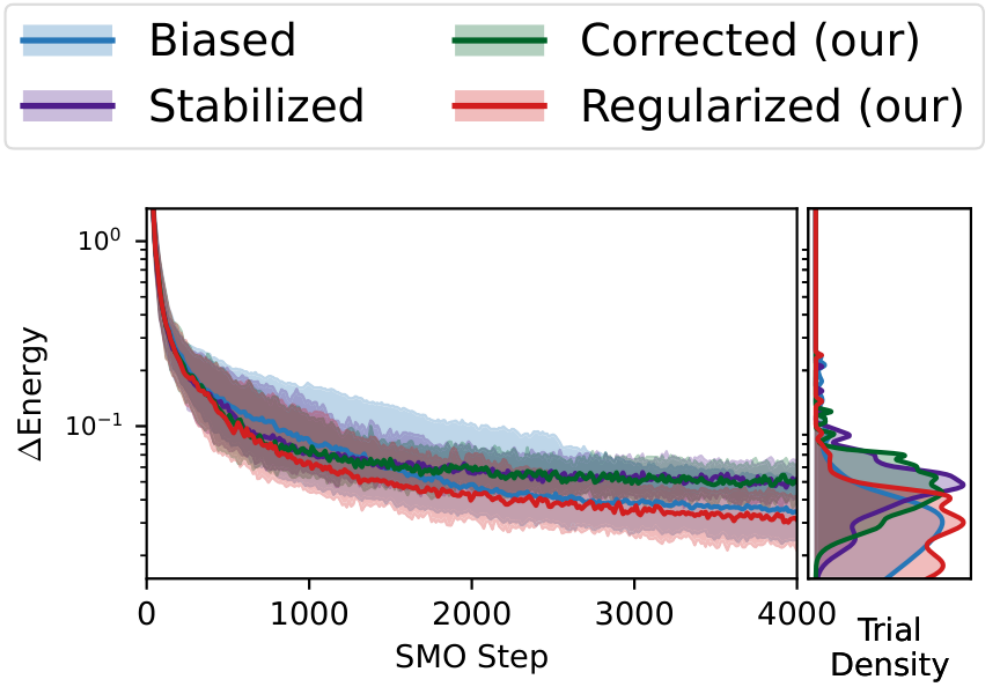


True energy — Corrected energy — Regularized energy

Replace spontaneous bias with a chosen regularizer; correct the reported energy.

Regularization improves the true energy reached

The corrected estimate alone does not produce the best optimization.



Red: regularized NFT

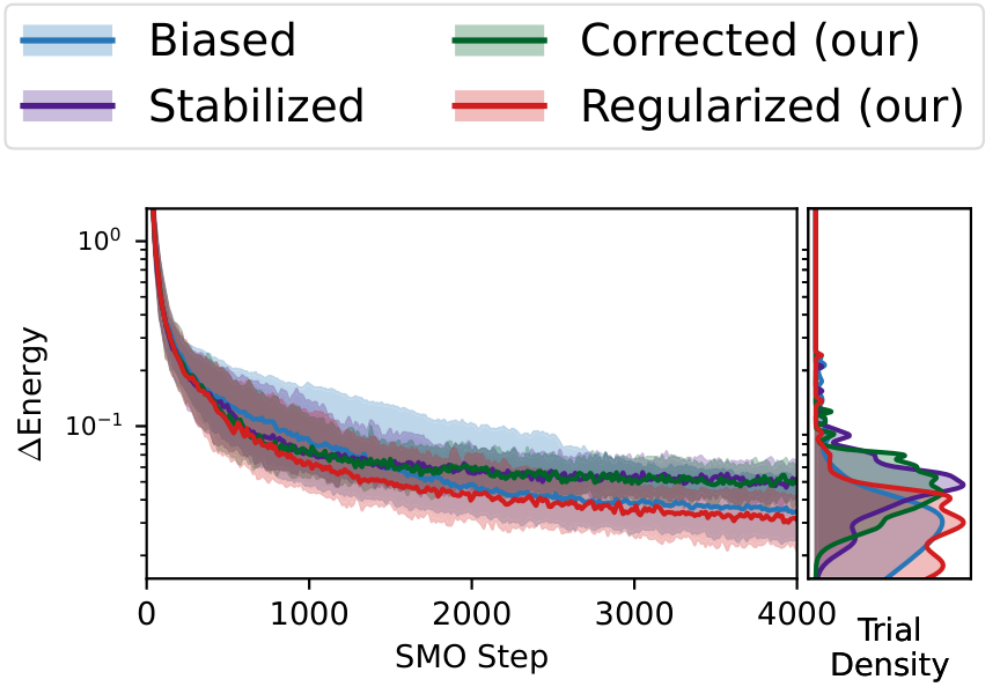
Lower true energy error than correction alone and the original NFT.

5 qubits · 3 layers
Ising model · 100 trials
900 shots / evaluation

Bias paper, Fig. 2 (energy panel) · 100 sweeps / 4,000 updates · mean and one standard deviation

Regularization improves the true energy reached

The corrected estimate alone does not produce the best optimization.



Red: regularized NFT

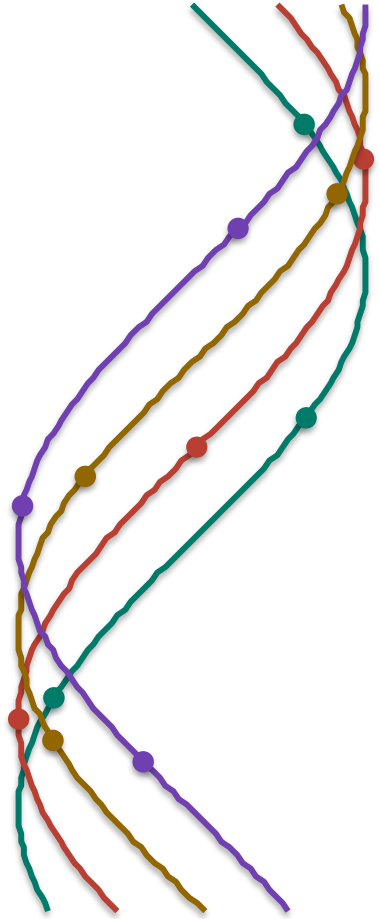
Lower true energy error than correction alone and the original NFT.

5 qubits · 3 layers
Ising model · 100 trials
900 shots / evaluation

Bias paper, Fig. 2 (energy panel) · 100 sweeps / 4,000 updates · mean and one standard deviation

The contribution: accurate energy estimation and a more stable coordinate update.

Context / Optimization Strategies



Two ways to optimize VQE

Sequential Minimal Optimization

$$\theta_d^* = \underset{\theta_d}{\operatorname{argmin}}\, E_d(\theta_d)$$

Look only at one parameter slice θ_d.
Minimize that slice.
Move to the next parameter θ_{d+1}.

Bias + regularization

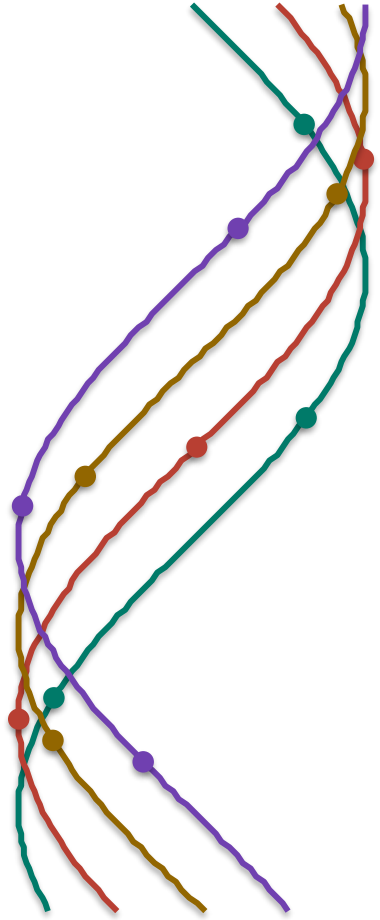
Gradient descent

$$\theta^* = \theta - \eta \hat{g}$$

Estimate the gradient $\hat{g}$.
Take a step set by the optimizer (ADAM).

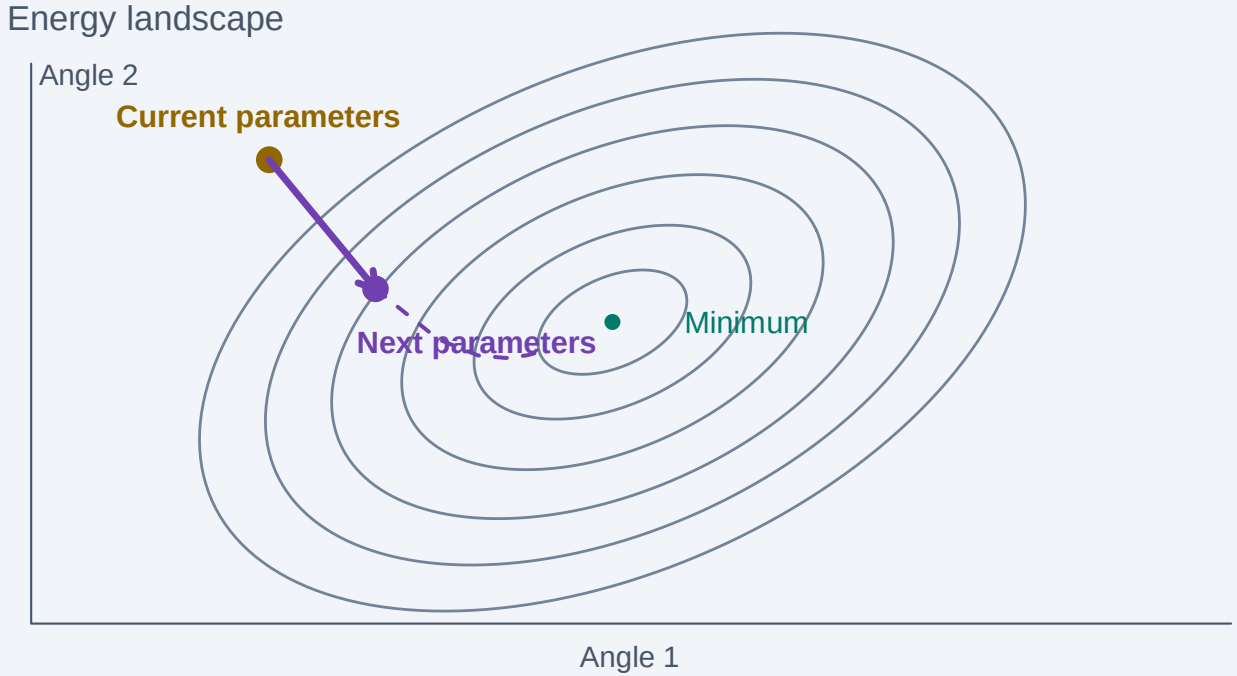
GradCoRe

Context / Gradient Descent



Gradient descent follows the local slopes

Estimate the gradient, take a controlled step in the full parameter space, then repeat.



Estimate the local slopes

One gradient component per parameter.

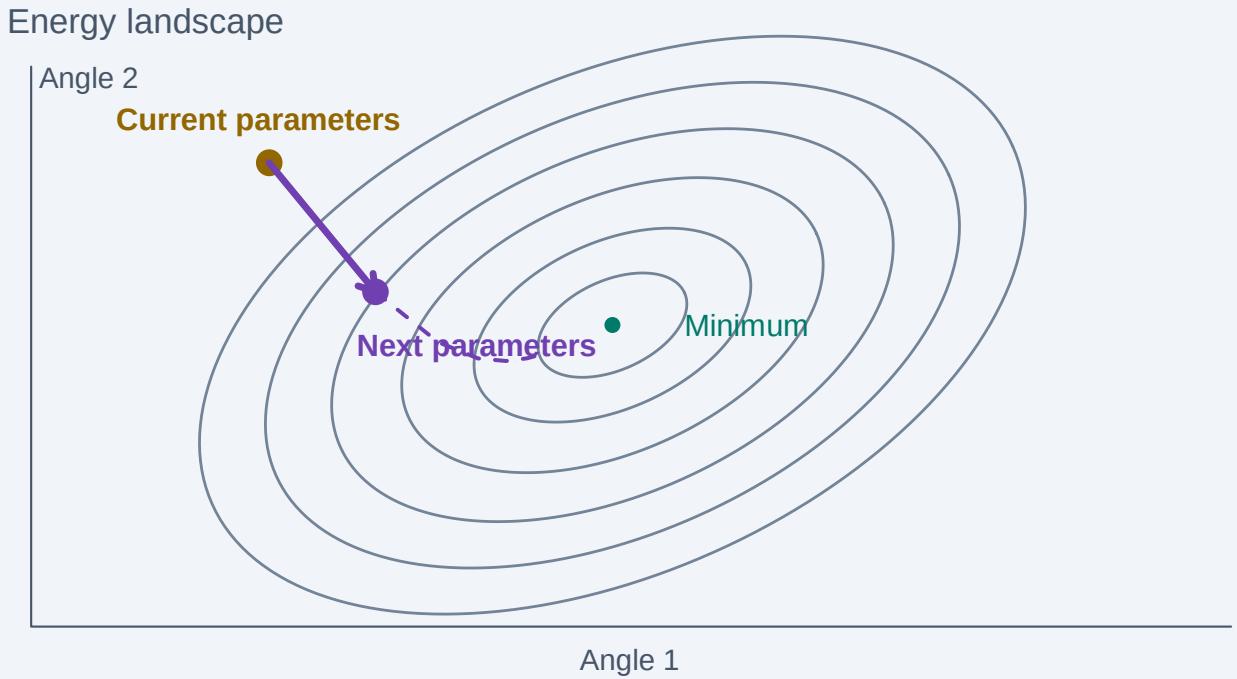
$$\theta^* = \theta - \eta \hat{g}$$

$\hat{g}$: estimated gradient
η: step size

Recompute after each update.

Gradient descent follows the local slopes

Estimate the gradient, take a controlled step in the full parameter space, then repeat.



Estimate the local slopes

One gradient component per parameter.

$$\theta^* = \theta - \eta \hat{g}$$

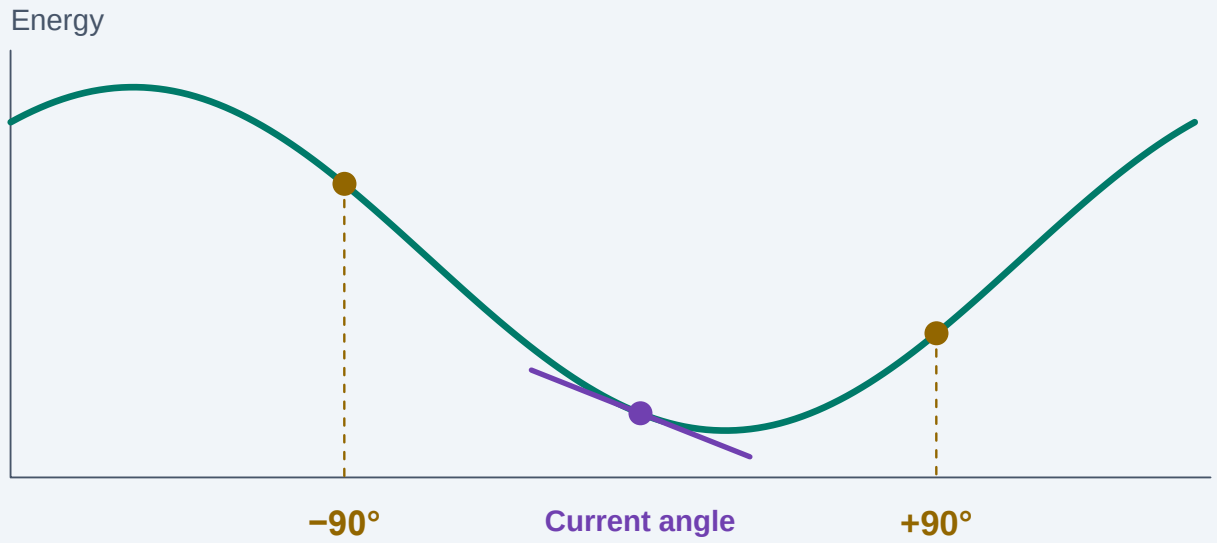
$\hat{g}$: estimated gradient
η: step size

Recompute after each update.

Use the slopes to update the parameters together, rather than minimize one fitted slice.

The parameter-shift rule turns energies into slopes

The same sinusoidal structure on which SMO-NFT is based gives an exact derivative from two shifted energies.



$$E'_d(t) = \frac{E_d\left(t + \frac{\pi}{2}\right) - E_d\left(t - \frac{\pi}{2}\right)}{2}$$

Two circuit settings

Shift one angle by −90° and +90°.

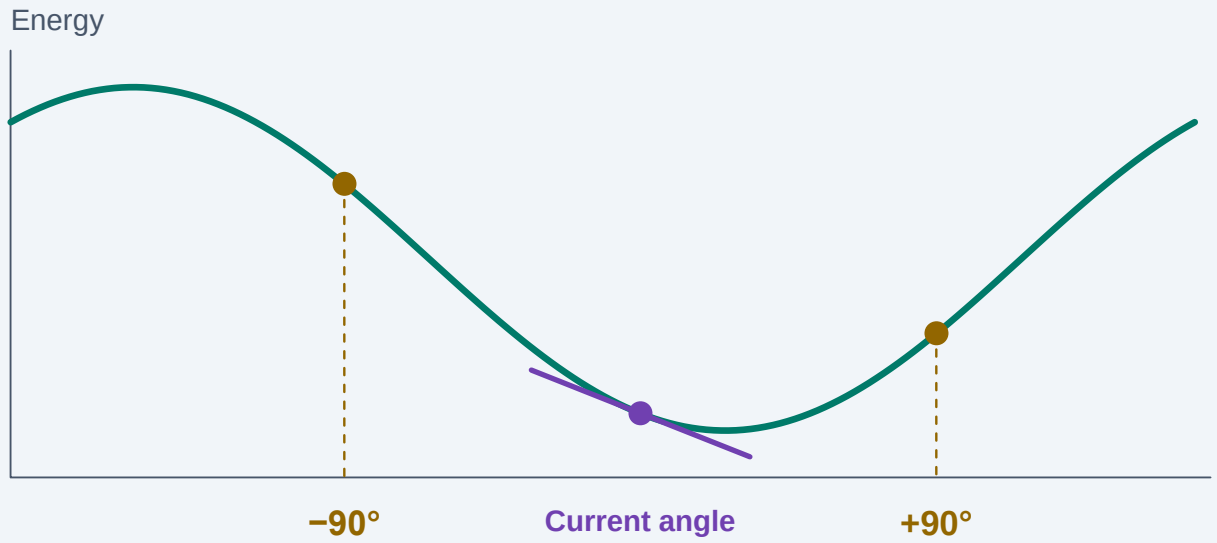
Measure the energy at each shift.

Subtract, then divide by two.

All other angles stay fixed.
Exact for the sinusoidal slice.

The parameter-shift rule turns energies into slopes

The same sinusoidal structure on which SMO-NFT is based gives an exact derivative from two shifted energies.



Two circuit settings

Shift one angle by −90° and +90°.

Measure the energy at each shift.

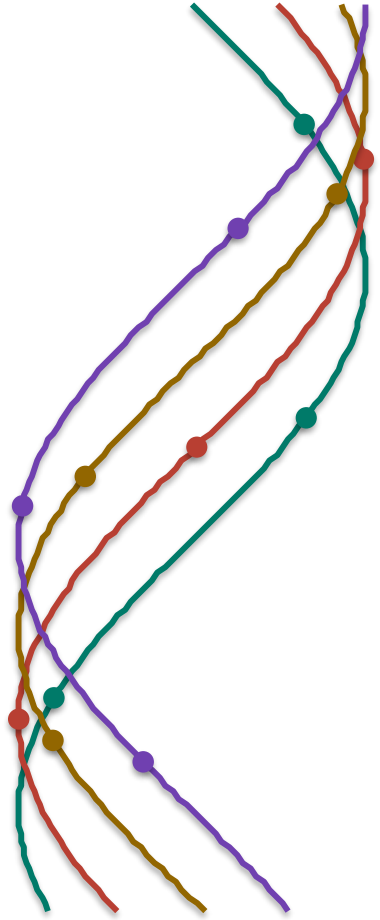
Subtract, then divide by two.

All other angles stay fixed.
Exact for the sinusoidal slice.

$$E'_d(t) = \frac{E_d\left(t + \frac{\pi}{2}\right) - E_d\left(t - \frac{\pi}{2}\right)}{2}$$

Finite shots make the gradient estimate noisy. How precise must it be for a useful step?

Our Work / Bayesian PSR and GradCoRe

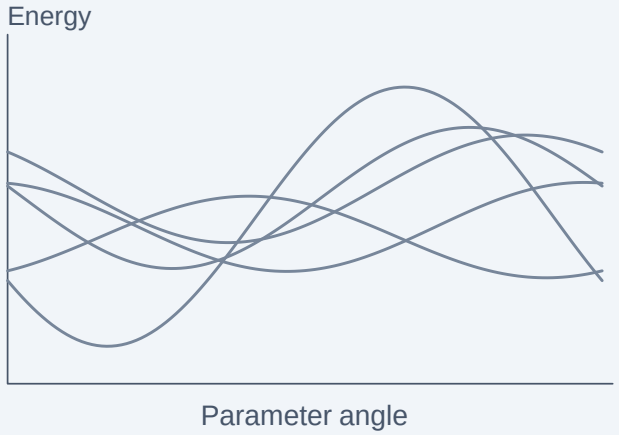


Bayesian inference: learn the gradient and its uncertainty

Gradient descent needs reliable slopes at the current parameters.

1 Start with structure

The known Fourier structure constrains the possible curves.

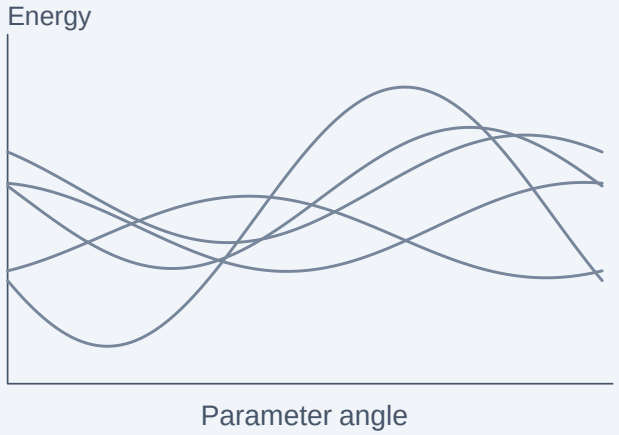


Bayesian inference: learn the gradient and its uncertainty

Gradient descent needs reliable slopes at the current parameters.

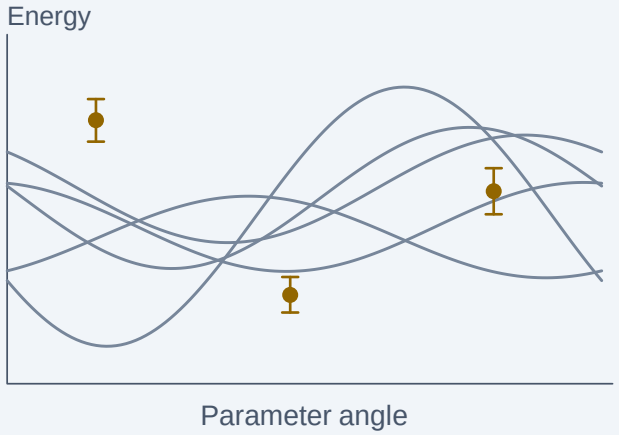
1 Start with structure

The known Fourier structure constrains the possible curves.



2 Add energy measurements

Past and new measurements refine the same GP model.

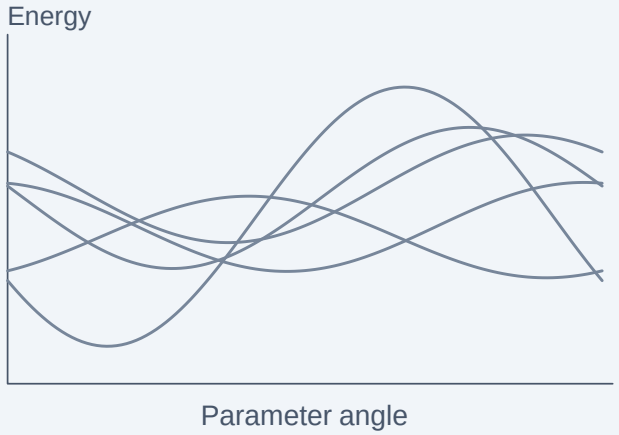


Bayesian inference: learn the gradient and its uncertainty

Gradient descent needs reliable slopes at the current parameters.

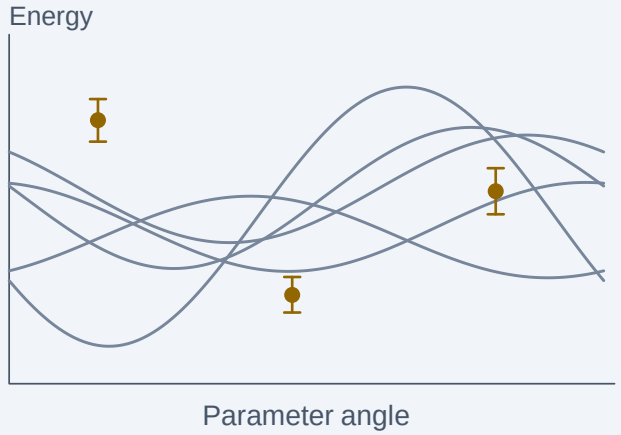
1 Start with structure

The known Fourier structure constrains the possible curves.



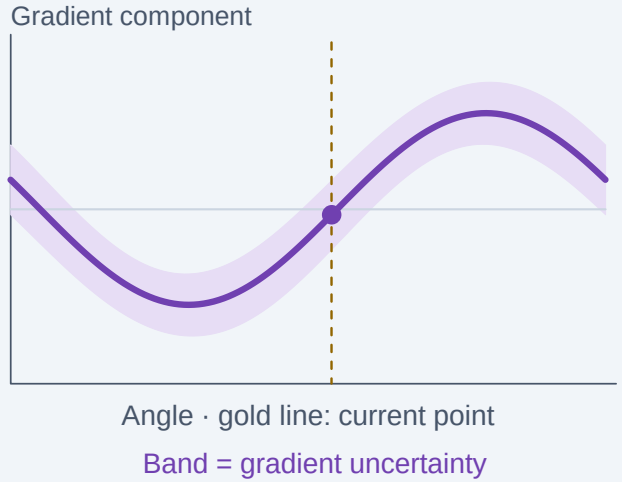
2 Add energy measurements

Past and new measurements refine the same GP model.



3 Infer the gradient

A slope estimate, plus how uncertain it remains.

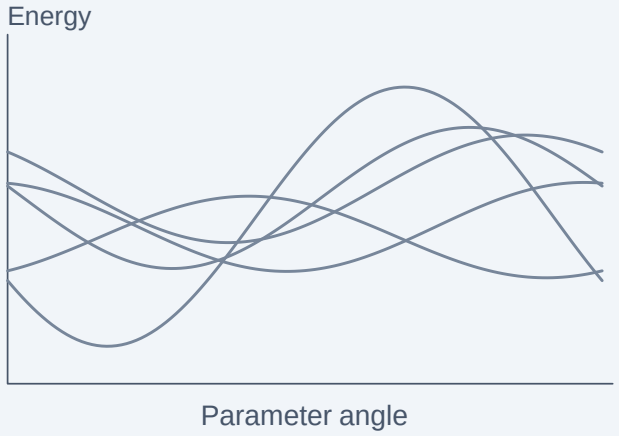


Bayesian inference: learn the gradient and its uncertainty

Gradient descent needs reliable slopes at the current parameters.

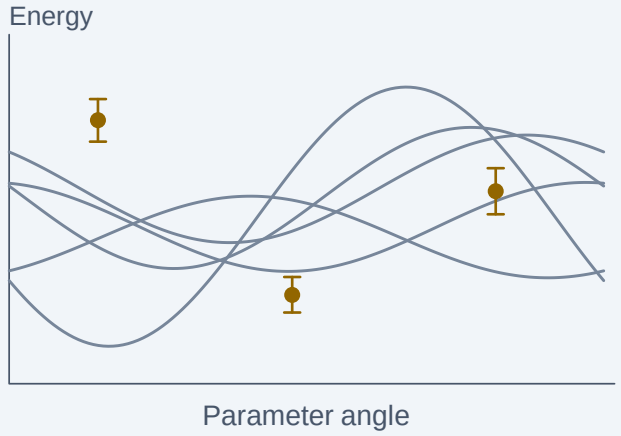
1 Start with structure

The known Fourier structure constrains the possible curves.



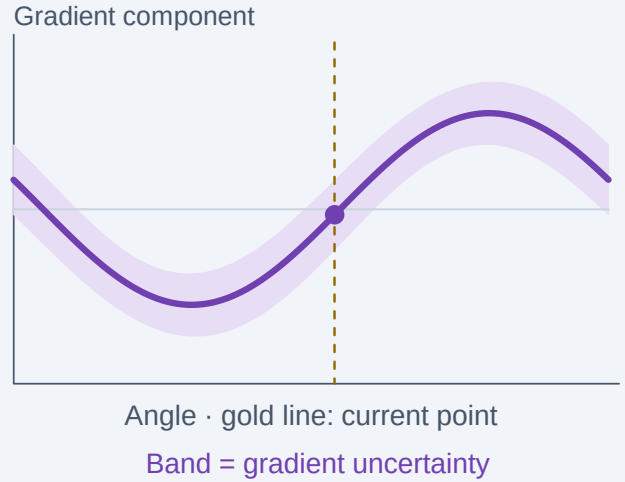
2 Add energy measurements

Past and new measurements refine the same GP model.



3 Infer the gradient

A slope estimate, plus how uncertain it remains.



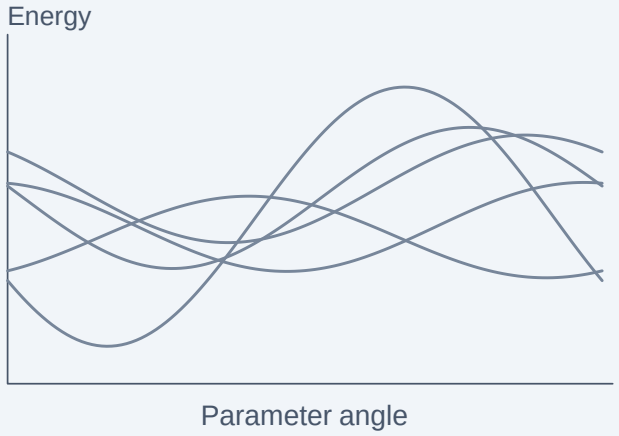
Energy observations → GP model → gradient estimate and uncertainty (Bayesian PSR).

Bayesian inference: learn the gradient and its uncertainty

Gradient descent needs reliable slopes at the current parameters.

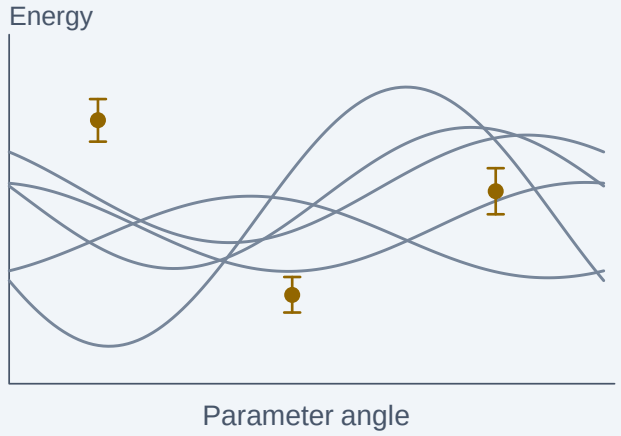
1 Start with structure

The known Fourier structure constrains the possible curves.



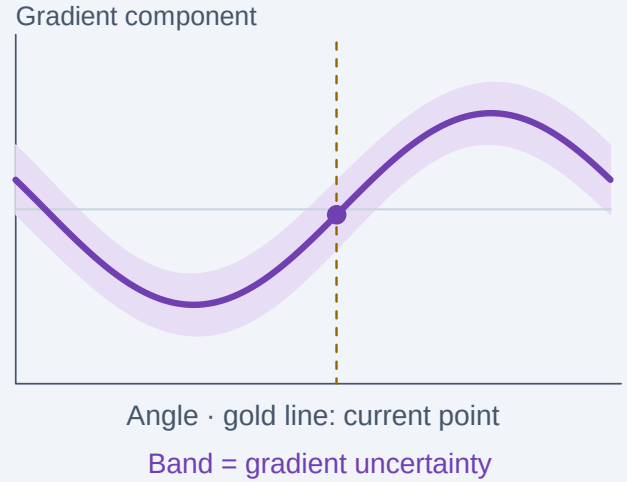
2 Add energy measurements

Past and new measurements refine the same GP model.



3 Infer the gradient

A slope estimate, plus how uncertain it remains.

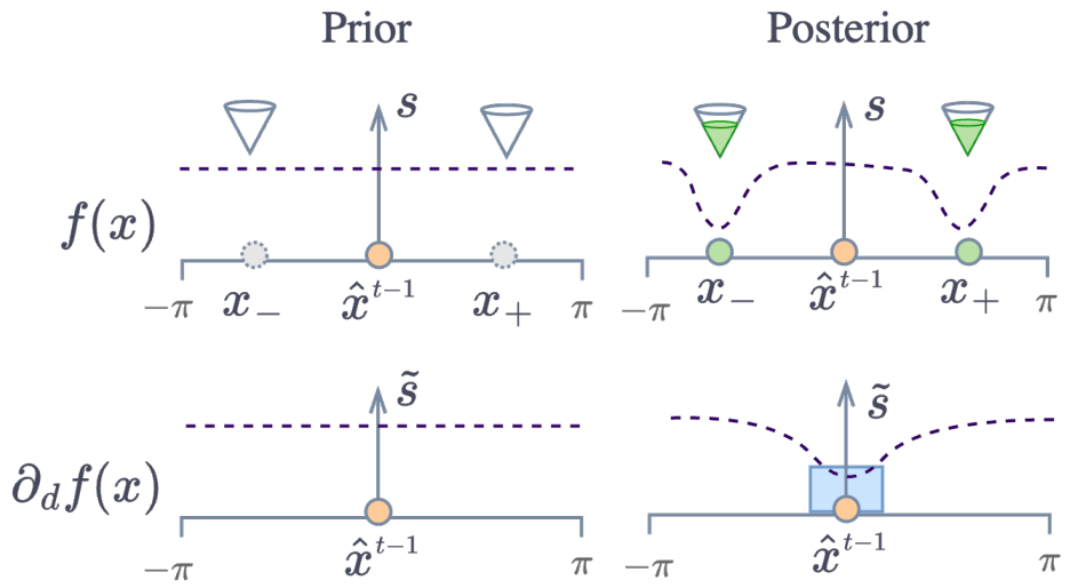


Energy observations → GP model → gradient estimate and uncertainty (Bayesian PSR).

GradCoRe leverages uncertainty in the gradient estimate to allocate resources in the next update.

GradCoRe spends shots on gradient precision

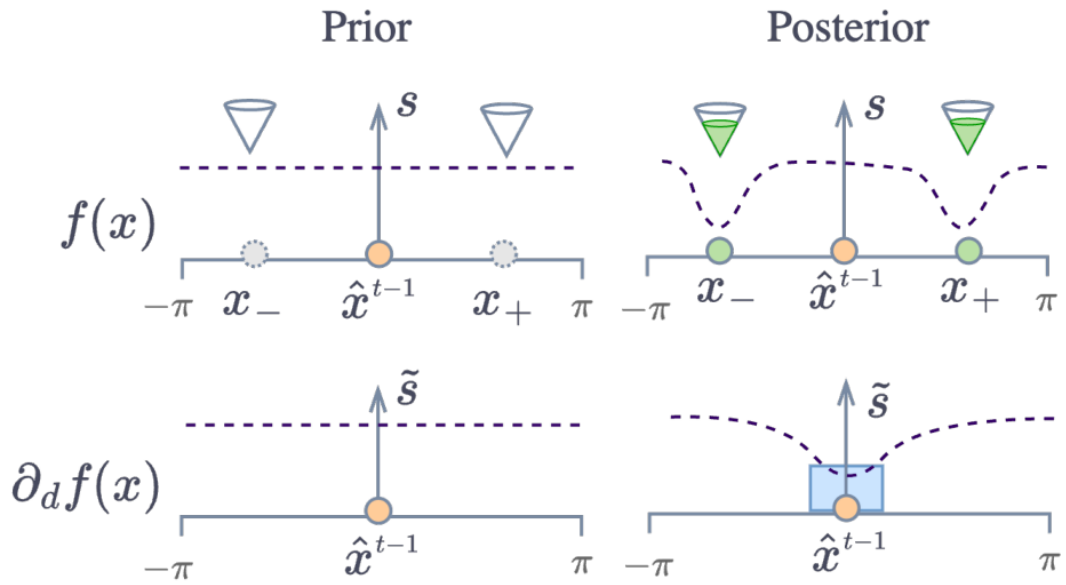
Choose a small measurement budget that makes the current gradient sufficiently certain.



Paper Fig. 1 · top: energy uncertainty · bottom: gradient uncertainty

GradCoRe spends shots on gradient precision

Choose a small measurement budget that makes the current gradient sufficiently certain.

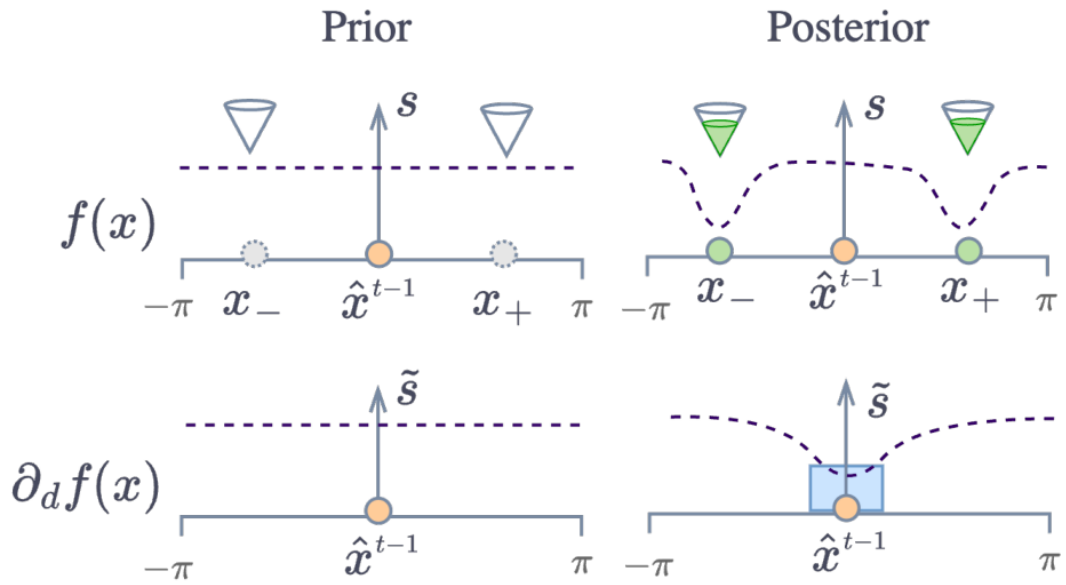


How do we choose the budget?

Paper Fig. 1 · top: energy uncertainty · bottom: gradient uncertainty

GradCoRe spends shots on gradient precision

Choose a small measurement budget that makes the current gradient sufficiently certain.



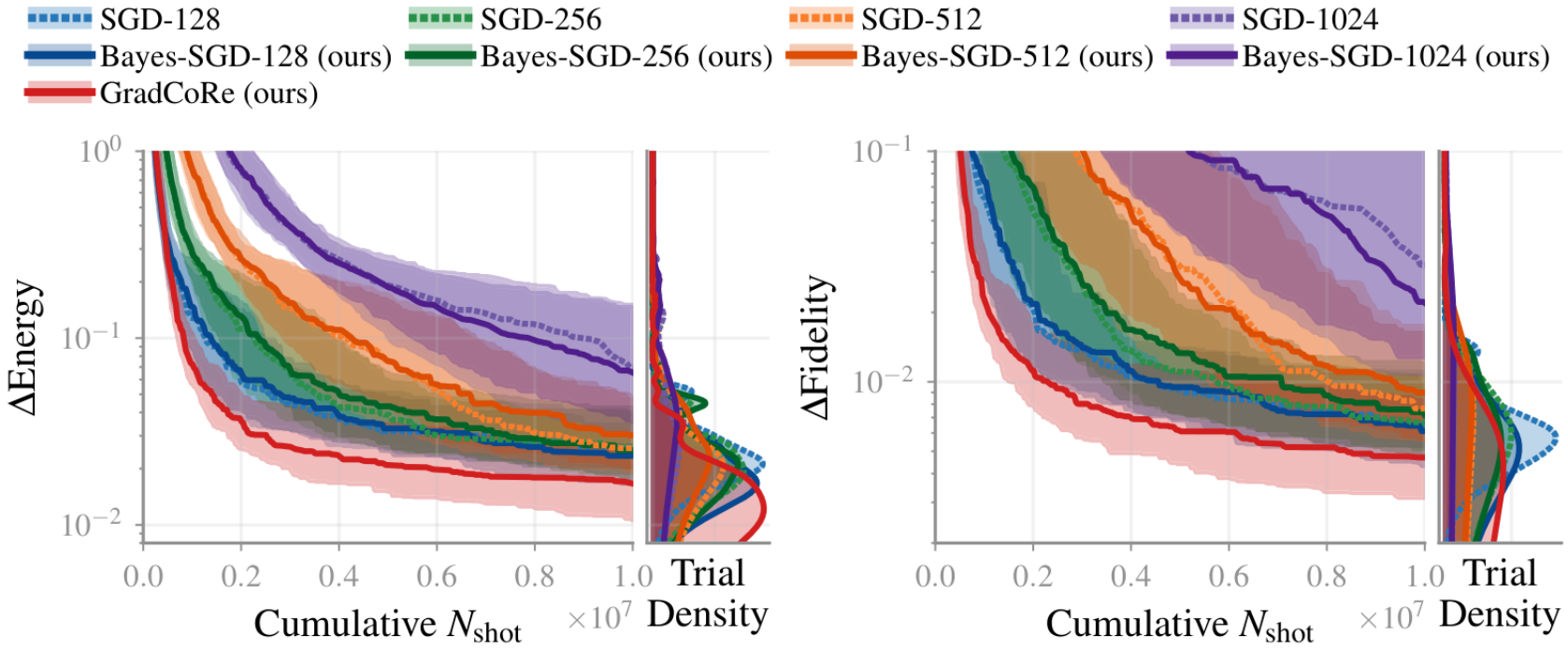
How do we choose the budget?

Paper Fig. 1 · top: energy uncertainty · bottom: gradient uncertainty

The need for precision increases as the optimization proceeds. GradCoRe calculates this and allocates.

Adaptive shots improve progress per measurement

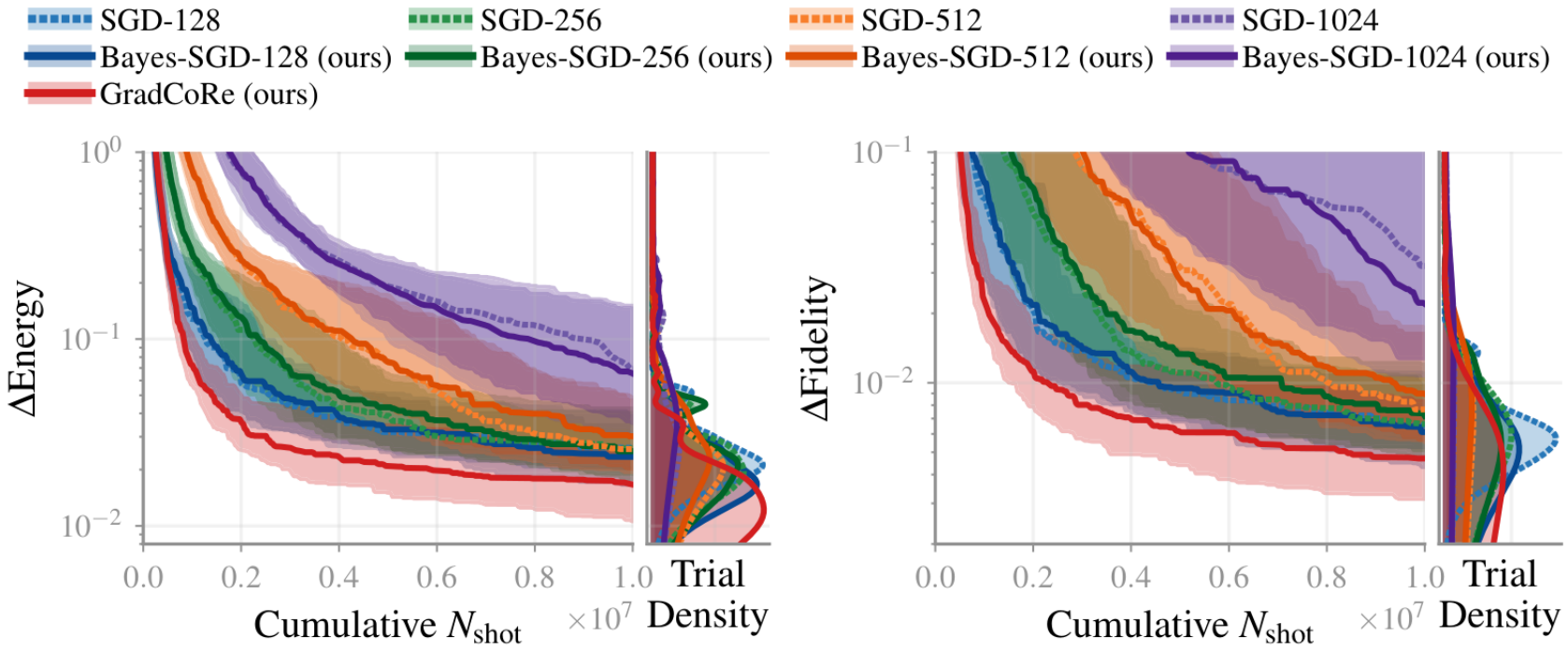
Red: GradCoRe. Other curves: fixed shot budgets with standard or Bayesian PSR.



Paper Fig. 3 · Ising model · 5 qubits / 3 layers · 100 trials

Adaptive shots improve progress per measurement

Red: GradCoRe. Other curves: fixed shot budgets with standard or Bayesian PSR.



Paper Fig. 3 · Ising model · 5 qubits / 3 layers · 100 trials

Bayes-PSR gives gradient uncertainty. GradCoRe uses it to decide how much to measure.

Two optimizers, two uses of uncertainty

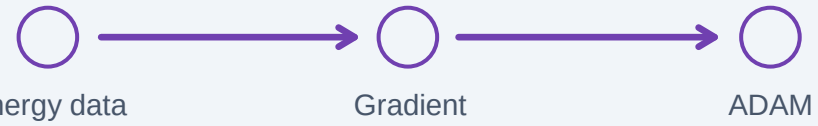
SMO–NFT · Bias



Correct and regularize the energy estimate.

Stabilize the coordinate update.

Gradient descent · GradCoRe



Infer the gradient and its uncertainty.

Adapt the measurement budget.

Two optimizers, two uses of uncertainty

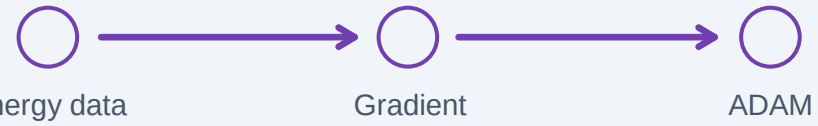
SMO–NFT · Bias



Correct and regularize the energy estimate.

Stabilize the coordinate update.

Gradient descent · GradCoRe



Infer the gradient and its uncertainty.

Adapt the measurement budget.

Thank you!