Neural-network approaches for preconditioning the Dirac equation in lattice QCD

Simon Pfahler

University of Regensburg

May 26, 2026

Background

Hopping expansion

Problem statement

Bigger picture

- Lattice QCD calculations are expensive
- One of the most expensive parts is solving the Dirac equation
- Preconditioners can accelerate this

Preconditioners

- State of the art: algebraic multigrid (AMG)
- But AMG needs costly setup
- Can this setup be avoided?
- **Goal**: Preconditioner competitive with AMG, but with very little (or even no) setup

Lattice Definitions 1

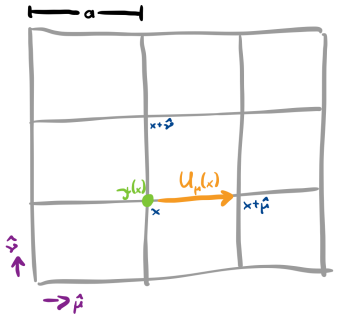
- We simulate QCD on a discrete space-time lattice
  $\rightarrow$ fermion fields $\psi \in \mathbb{C}^{V \times 4 \times 3}$
  $\rightarrow$ gauge fields $U \in \mathrm{SU}(3)^{4 \times V}$
- Gauge transformations $\Omega \in \mathrm{SU}(3)^V$ transform the fields as

$$\psi(x) \rightarrow \Omega(x)\psi(x)$$
$$U_\mu(x) \rightarrow \Omega(x)U_\mu(x)\Omega^\dagger(x+\hat{\mu})$$

- This leads to parallel-transport operators

$$H_\mu\psi(x) = U_\mu^\dagger(x-\hat{\mu})\psi(x-\hat{\mu})$$

These transform as $H_\mu\psi(x) \rightarrow \Omega(x)H_\mu\psi(x)$



Lattice Definitions 2

Dirac equation: $D\psi = \phi$
We use the Wilson(-clover) Dirac operator

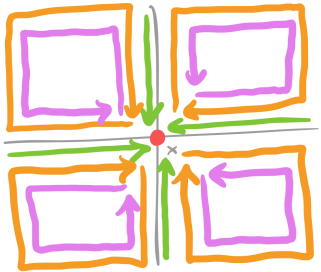
$$D_{\mathrm{WC}} = D_{\mathrm{W}} - \frac{c_{\mathrm{sw}}}{4} \sum_{\mu,\nu} \sigma_{\mu\nu} F_{\mu\nu}$$

$$D_{\mathrm{W}} = (m+4) - \sum_{\mu} \left(\frac{\mathbb{1} - \gamma_\mu}{2} H_{-\mu} + \frac{\mathbb{1} + \gamma_\mu}{2} H_{+\mu} \right)$$

$$F_{\mu\nu} = \frac{1}{8}(Q_\mu\nu - Q_\nu\mu)$$

$$Q_{\mu\nu} = P_{+\nu,+\mu} + P_{-\mu,+\nu} + P_{-\nu,-\mu} + P_{+\mu,-\nu}$$

$$P_{\pm\mu,\pm\nu} = H_{\mp\nu} H_{\mp\mu} H_{\pm\nu} H_{\pm\mu}$$



action of D_{WC} on $\psi(x)$

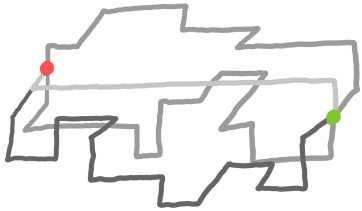
Hopping expansion

- We can write $D_{\mathrm{W}} = (m+4)(\mathbb{1} - \kappa H)$ with hopping parameter $\kappa = \frac{1}{2(m+4)}$
- For $\kappa < \frac{1}{8}$, the inverse is of D_{W} is given by

$$D_{\mathrm{W}}^{-1} = \frac{1}{m+4}(\mathbb{1} - \kappa H)^{-1} = \frac{1}{m+4} \sum_{n=0}^{\infty} \kappa^n H^n$$

- Can we learn systematic hopping-style expansions using Machine Learning?

  $\Rightarrow$ Yes, let's see how!



Machine Learning

Hopping expansion

Network Architecture

Dense linear (L) layer

- n spin-color fields $\varphi \in \mathbb{C}^{L_x \times L_y \times L_z \times L_t \times N_c \times N_s}$
- build m linear combinations of the input fields, with spin matrices $W_{ij} \in \mathbb{C}^{N_s \times N_s}$ as weights:

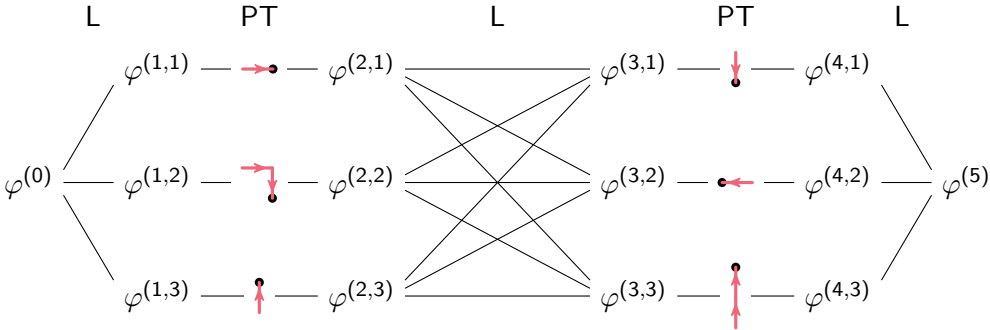
$$\psi_i(x) \overset{\mathsf{L}}{=} \sum_{j=1}^{n} W_{ij} \varphi_j(x)$$

Parallel Transport (PT) layer

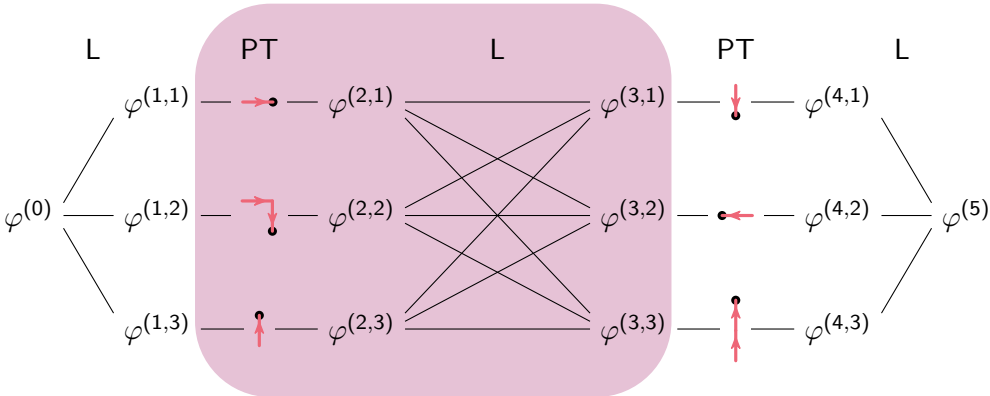
- n spin-color fields $\varphi \in \mathbb{C}^{L_x \times L_y \times L_z \times L_t \times N_c \times N_s}$
- apply a different parallel transport $p^{(i)}$ to every input field:

$$\psi_i(x) \overset{\mathsf{PT}}{=} T_{p^{(i)}} \varphi_i(x)$$

Network Architecture



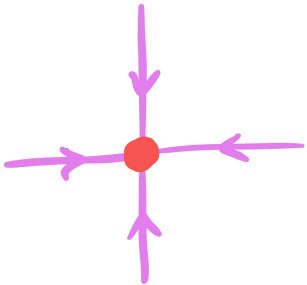
Network Architecture



Choice of PT paths

The most natural choice of PT paths is only nearest-neighbor hops:

$$P = \{\mathbb{1}\} \cup \left\{H_p | p \in \{\pm 1, \pm 2, \pm 3, \pm 4\}\right\}$$



Score function

- To train our network, we need a score function
- Ideally, we want to optimize the condition number of MD,

$$\kappa(MD) = \frac{|\lambda_{\text{max}}|}{|\lambda_{\text{min}}|}$$

  $\rightarrow$ Expensive and unstable ☹

- Instead, we use the surrogate cost function

$$C = \|MD\varphi - \varphi\|^2$$

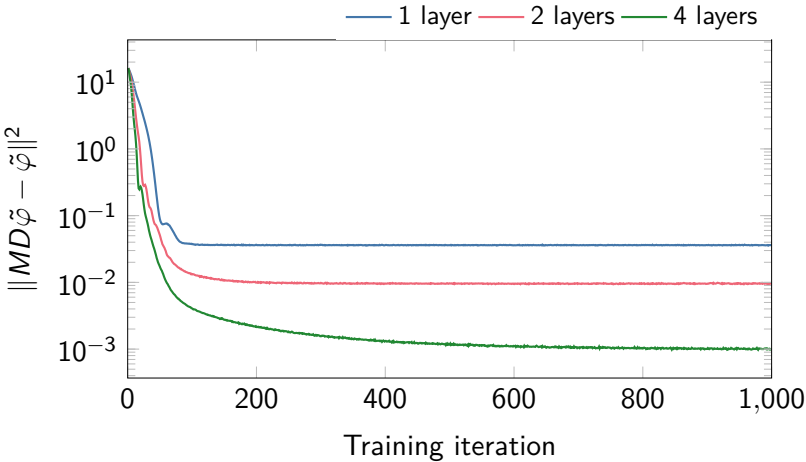
  with random fermion fields φ

Results

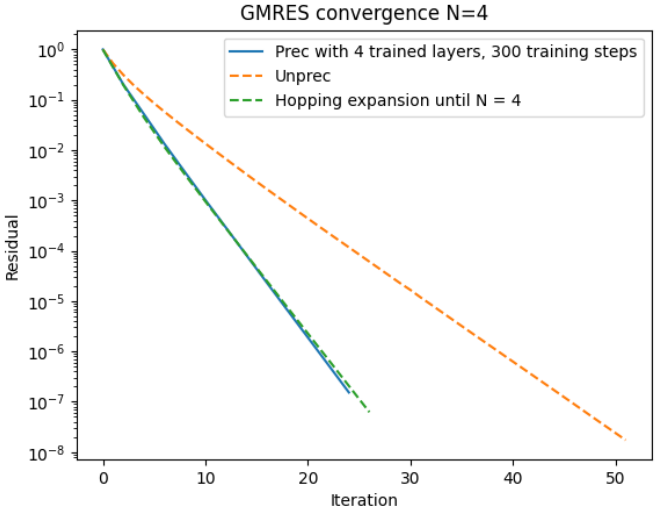
Hopping expansion

Thanks to
Maren Käfferlein!

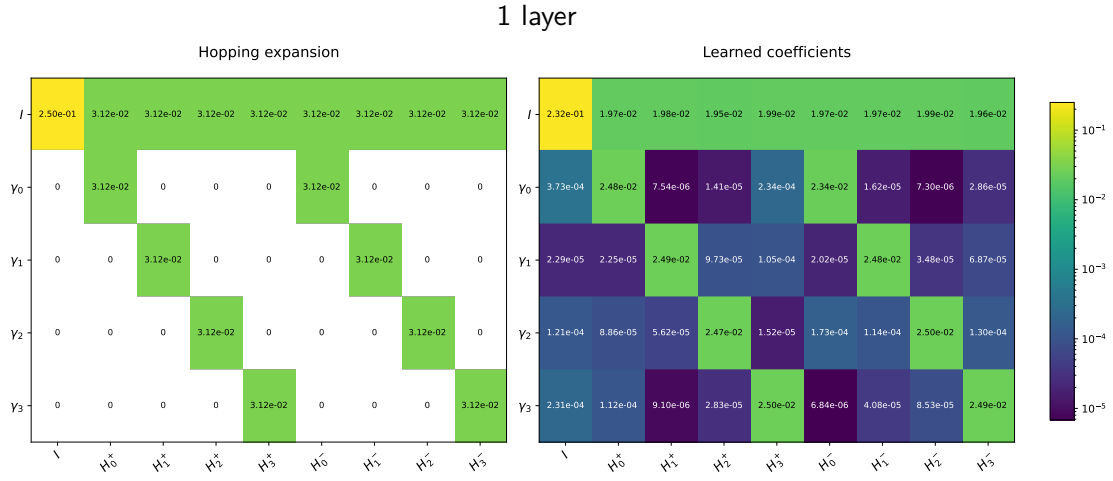
Training


1 layer
2 layers
4 layers
$\|MD\tilde{\varphi} - \tilde{\varphi}\|^2$
10^1
10^0
10^{-1}
10^{-2}
10^{-3}
0
200
400
600
800
1,000
Training iteration

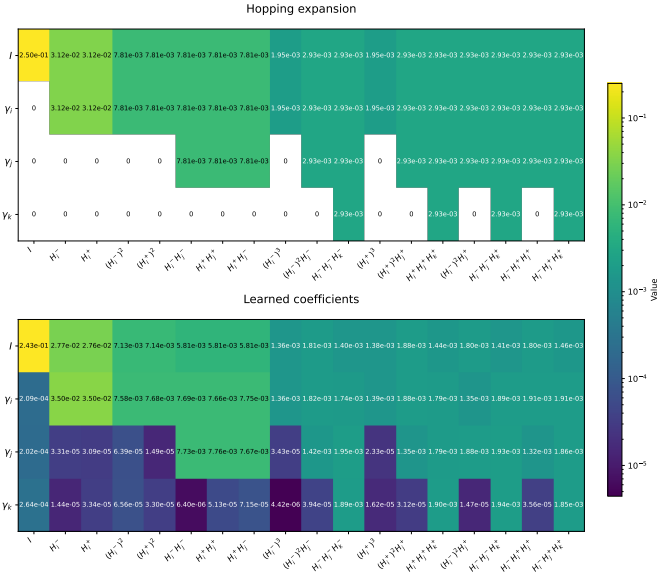
Test solves


GMRES convergence N=4
Prec with 4 trained layers, 300 training steps
Unprec
Hopping expansion until N = 4
Residual
Iteration
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{-7}
10^{-8}
0
10
20
30
40
50

Analytical structure



Analytical structure



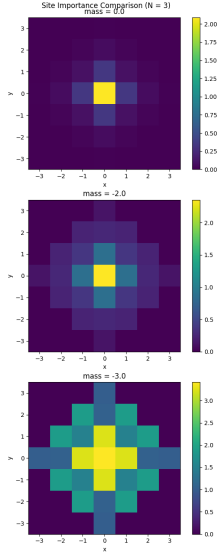
3 layers

Background

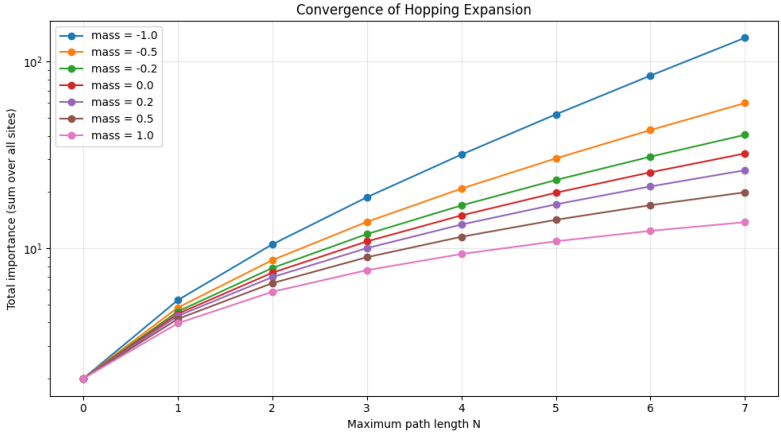
Critical slowing down

Critical slowing down

- For $\kappa = \frac{1}{2(m+4)} < \frac{1}{8}$, the hopping expansion does not converge anymore, as long paths dominate
- Near $\kappa \approx \frac{1}{8}$, we need an increasing amount of layers to build a good approximation of D^{-1}
- How can we make our networks more efficient in this regime?



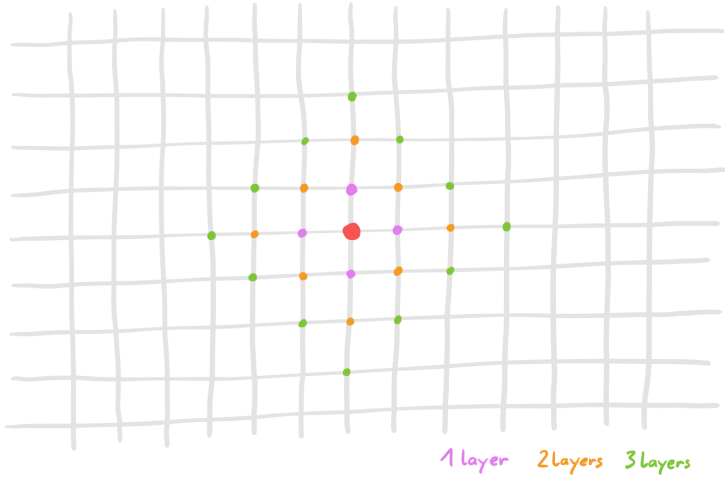
Critical slowing down



Machine Learning

Critical slowing down

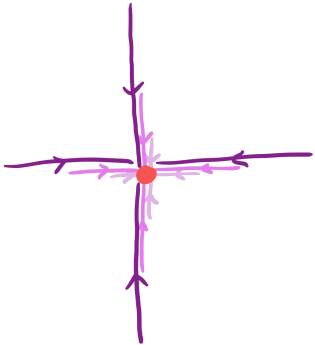
PT Paths 1



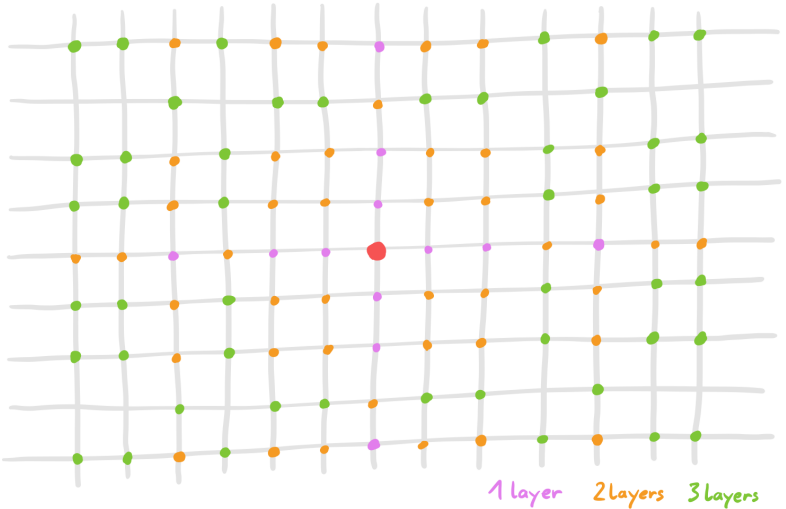
PT Paths 2

To allow for longer paths without increasing the number of paths too much, we choose

$$P = \{\mathbb{1}\} \cup \left\{ H_p^n | p \in \{\pm 1, \pm 2, \pm 3, \pm 4\}, \right.$$
$$\left. n \in \{2^0, 2^2, \ldots, L_p/2\} \right\}$$



PT Paths 3



Score function

- **Standard choice**:

$$C = \|MD\varphi - \varphi\|^2$$

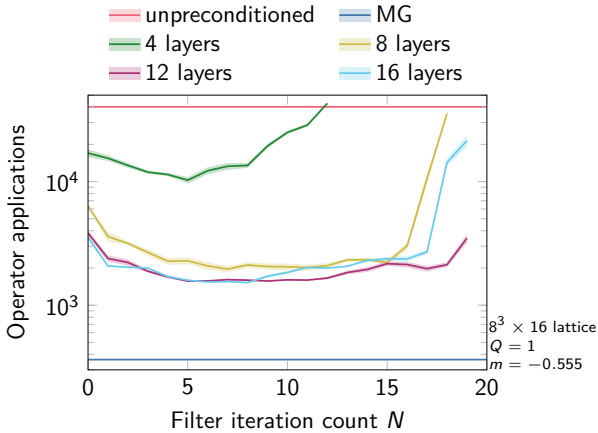
with random field φ

- **Filtered cost function**:

$$C = \|MD\tilde{\varphi} - \tilde{\varphi}\|^2$$

where $\tilde{\varphi}$ is obtained through N iterations of GMRES for $D\varphi = 0$ with random initial guess

$\Rightarrow$ Stronger emphasis on the low modes!



Results

Critical slowing down

Gauge configurations

$8^3 \times 16$ ensemble[1]:

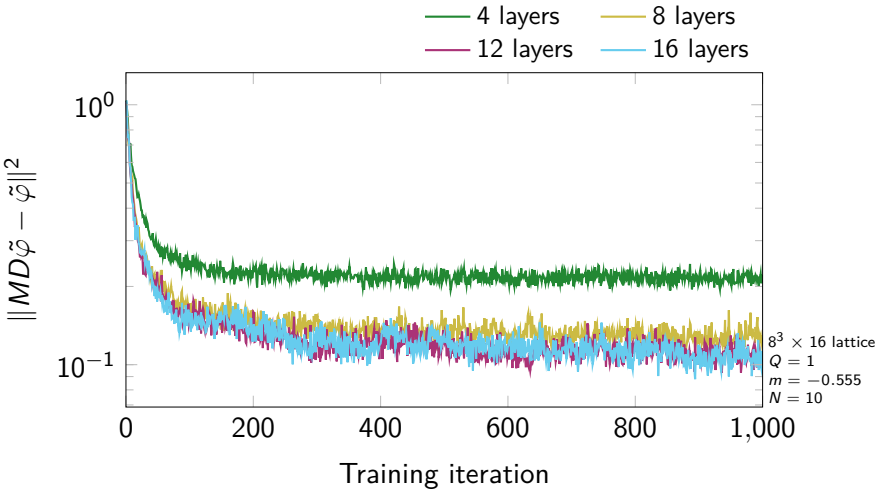
- Wilson action
- quenched, heat bath
- Wilson-clover Dirac operator
- configurations with topological charge $|Q| \in \{0, 1, 2\}$

$16^3 \times 32$ ensemble[2]:

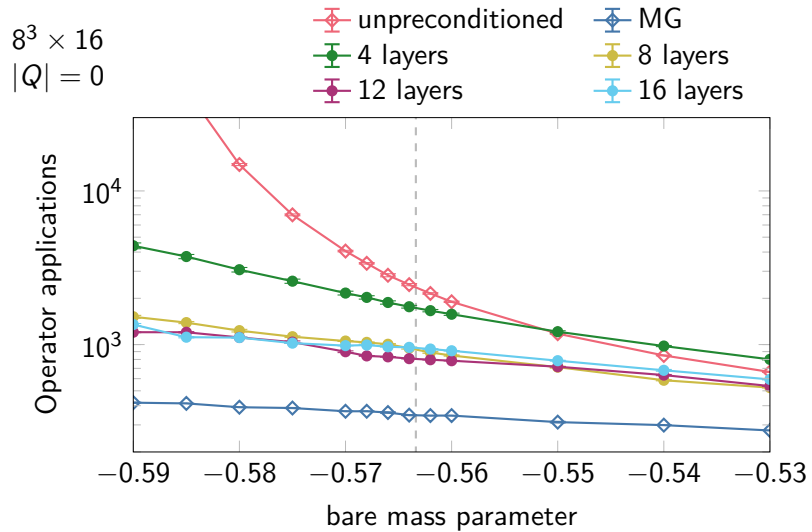
- Wilson action
- quenched, HMC
- Wilson-clover Dirac operator
- configurations with topological charge $|Q| \in \{0, 4\}$

[1] doi.org/10.5281/zenodo.15018324
[2] doi.org/10.5281/zenodo.17494829

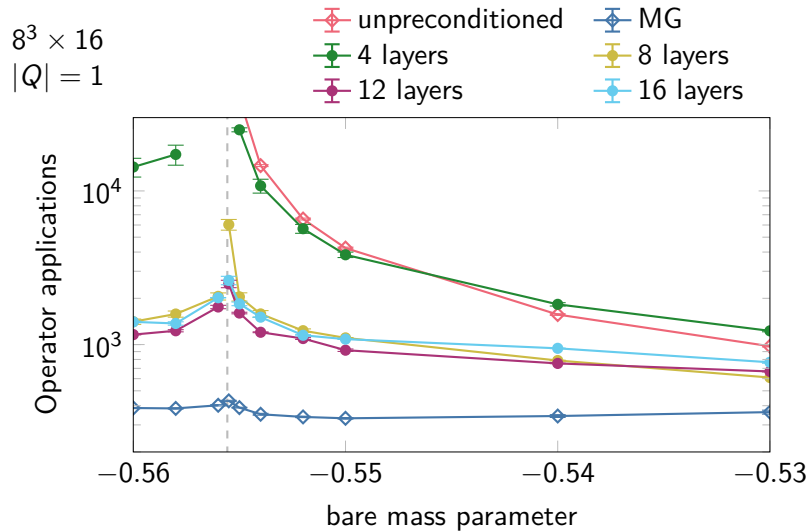
Training cost



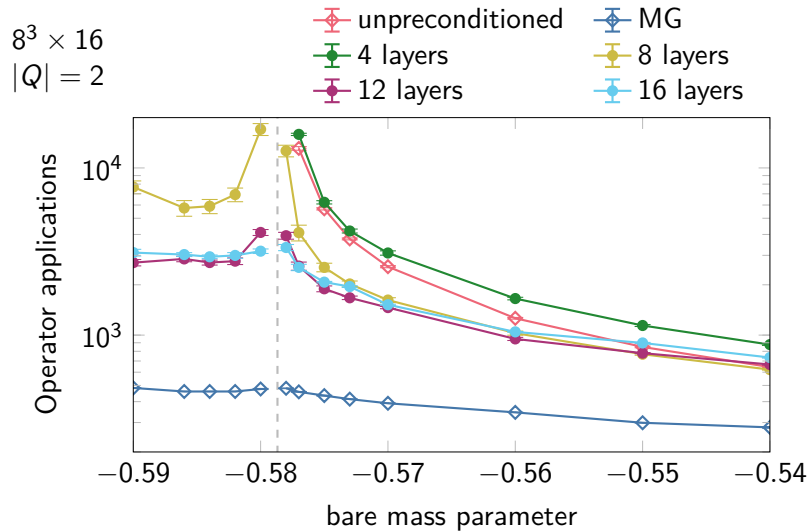
Critical slowing down



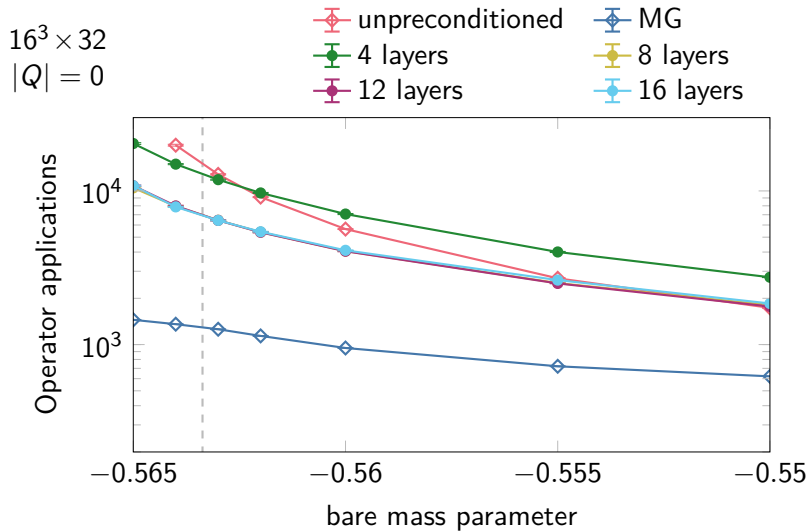
Critical slowing down



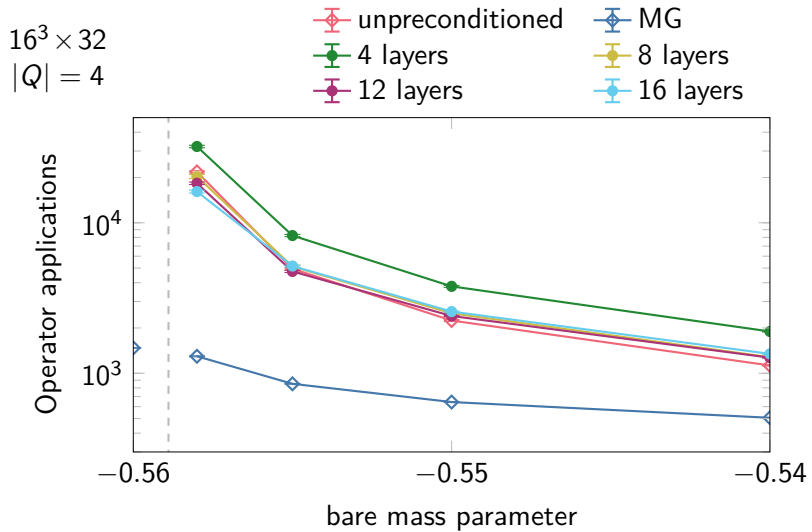
Critical slowing down



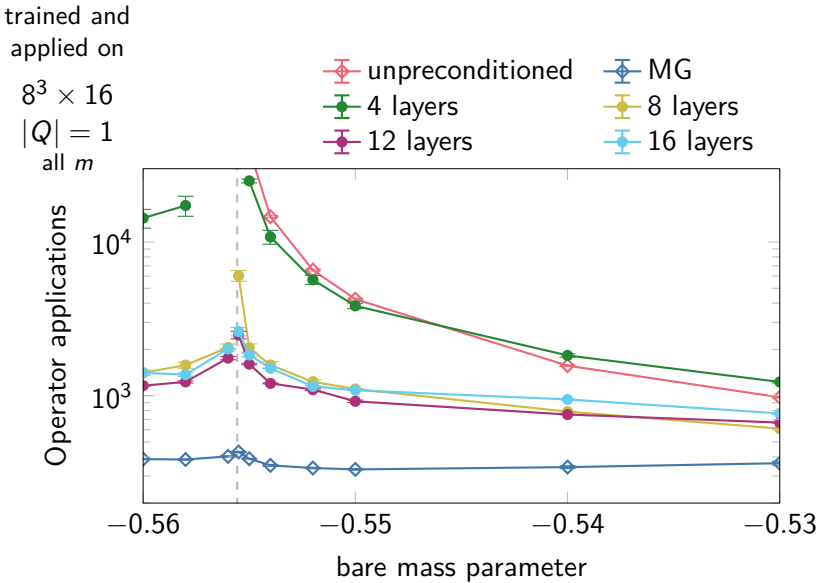
Critical slowing down



Critical slowing down



Transfer

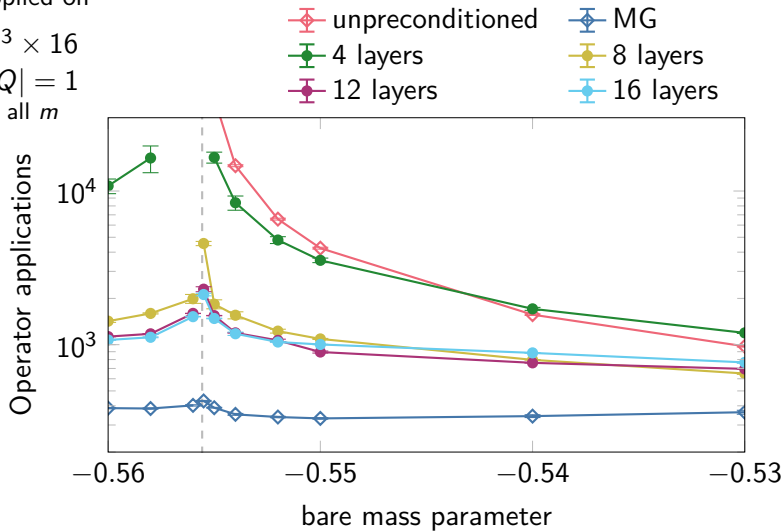

trained and
applied on
$8^3 \times 16$
$|Q| = 1$
all m
unpreconditioned
4 layers
12 layers
MG
8 layers
16 layers
Operator applications
10^4
10^3
−0.56
−0.55
−0.54
−0.53
bare mass parameter

Transfer

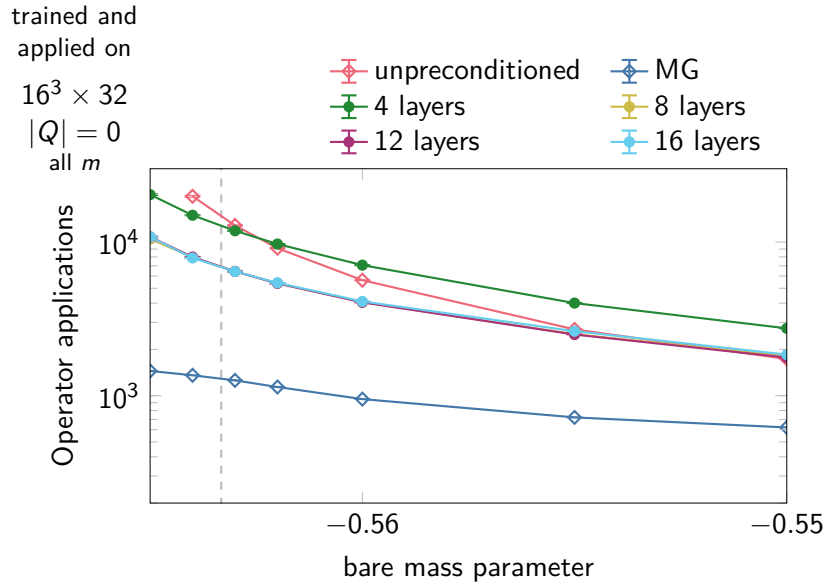
trained on
$8^3 \times 16$
$|Q| = 0$
$m = -0.56$

$\rightarrow$

applied on
$8^3 \times 16$
$|Q| = 1$
all m

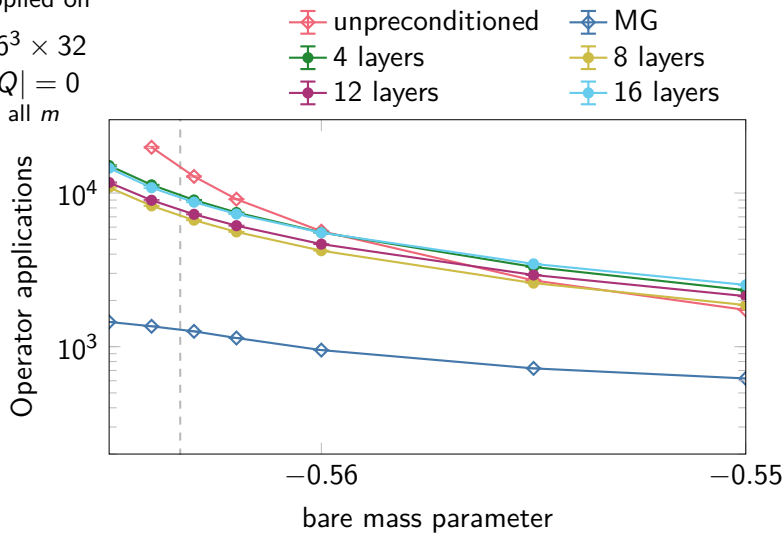


Transfer

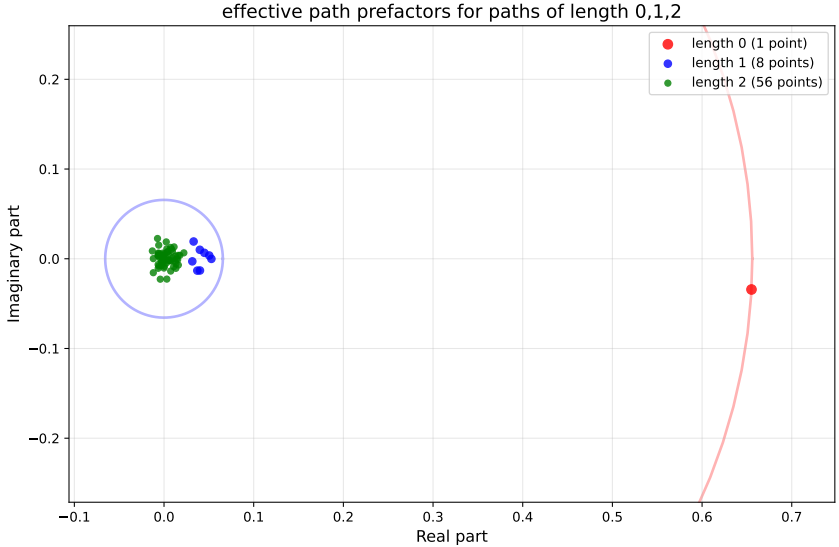

trained and
applied on
$16^3 \times 32$
$|Q| = 0$
all m
unpreconditioned
4 layers
12 layers
MG
8 layers
16 layers
Operator applications
10^4
10^3
−0.56
−0.55
bare mass parameter

Transfer

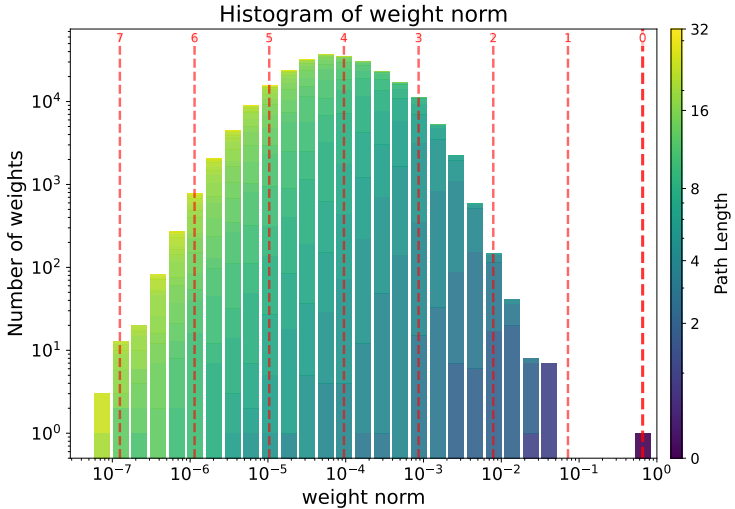
trained on $8^3 \times 16$, $|Q| = 0$, $m = -0.56$ $\rightarrow$ applied on $16^3 \times 32$, $|Q| = 0$, all m



Weights analysis



Weights analysis



Weights look systematic, but not like hopping expansion!

$\rightarrow$ new analytic formula for D_{WC}^{-1}?

Conclusion

Conclusion

Goals

- ☐ Build a preconditioner using ML that mitigates critical slowing down
- ☐ Learn a general analytic formula $\rightarrow$ No costly setup/trivial transfer
- ☐ Handle topological modes nicely

Slides and code:
simon-pfahler.github.io

Conclusion

Goals

- ☑ Build a preconditioner using ML that mitigates critical slowing down
- ☐ Learn a general analytic formula → No costly setup/trivial transfer
- ☐ Handle topological modes nicely

Slides and code:
simon-pfahler.github.io

Conclusion

Goals

- ☑ Build a preconditioner using ML that mitigates critical slowing down
- ☑ Learn a general analytic formula
  → No costly setup/trivial transfer
- ☐ Handle topological modes nicely

Slides and code:
simon-pfahler.github.io

Conclusion

Goals

- ☑ Build a preconditioner using ML that mitigates critical slowing down
- ☑ Learn a general analytic formula
  → No costly setup/trivial transfer
- ☒ Handle topological modes nicely

Slides and code:
simon-pfahler.github.io

Conclusion

Goals

- ☑ Build a preconditioner using ML that mitigates critical slowing down
- ☑ Learn a general analytic formula → No costly setup/trivial transfer
- ☒ Handle topological modes nicely

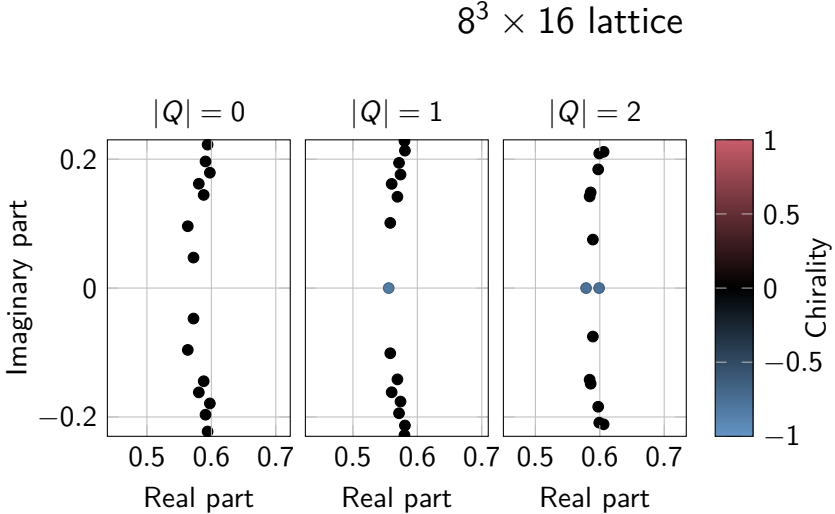
Next steps

- Extract the analytic expansion the models learned
- Understand why the architecture fails for topological modes
- Explore models to generate representative near-null space vectors economically

Slides and code:
simon-pfahler.github.io

Bonus Slides

Spectra


$8^3 \times 16$ lattice
$|Q| = 0$
$|Q| = 1$
$|Q| = 2$
Imaginary part
0.2
0
−0.2
0.5
0.6
0.7
Real part
Chirality
1
0.5
0
−0.5
−1

Spectra

$16^3 \times 32$ lattice