Reducing Residual mass of Domain-Wall Fermions using Machine Learning

Shunsuke Yasunaga[1,2]
Kenta Yoshimura[1,2], Akio Tomiya[2,3,4], and Yuki Nagai[5]
[1]Institute of Science Tokyo
[2]RIKEN R-CCS
[3]Tokyo Woman's Christian University
[4]Kyoto University
[5]The University of Tokyo

Frontiers of Lattice Fermions, YITP Kyoto, 2026/6/29–7/3

Research overview

Scope of this talk

This talk is about **chiral symmetry in lattice QCD**:
domain-wall fermions are used as chirally symmetric discretization for QCD.

Main issue

At finite fifth-dimensional extent L_5, the overlap fermion is approximated, and a small residual chiral-symmetry violation remains.

We set the fifth-dimensional parameters with machine learning.

Domain-wall fermions at finite L_5

Domain-wall fermions realize light four-dimensional quarks on the two boundaries of a five-dimensional system.

- In the $L_5 \to \infty$ limit, the boundary operator becomes an overlap fermion and satisfies the Ginsparg–Wilson relation.
- The remaining violation is measured by the residual mass m_{res}.

Practical question: can optimized parameters reduce m_{res} without increasing L_5?

Fifth dimensional matrix

$$S_5 = \sum_{s,t} \bar{\Psi}_s \left(D_5\right)_{st} \Psi_t$$

$$(D_5)_{st} = \begin{pmatrix} D_+^{(1)} & D_-^{(1)} P_- & 0 & \cdots & 0 & 0 & -m_q D_-^{(1)} P_+ \\ D_-^{(2)} P_+ & D_+^{(2)} & D_-^{(2)} P_- & \cdots & 0 & 0 & 0 \\ 0 & D_-^{(3)} P_+ & D_+^{(3)} & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & D_+^{(L_5-2)} & D_-^{(L_5-2)} P_- & 0 \\ 0 & 0 & 0 & \cdots & D_-^{(L_5-1)} P_+ & D_+^{(L_5-1)} & D_-^{(L_5-1)} P_- \\ -m_q D_-^{(L_5)} P_- & 0 & 0 & \cdots & 0 & D_-^{(L_5)} P_+ & D_+^{(L_5)} \end{pmatrix}_{st}$$

$$D_+^{(s)} = b_s D_{\text{Wilson}}(-M) + 1$$

$$D_-^{(s)} = c_s D_{\text{Wilson}}(-M) - 1$$

Domain-wall to overlap fermion

Boundary four-dimensional operator is defined quarks fields. At finite L_5,

$$D_4 = \frac{1+m_q}{2} + \frac{1-m_q}{2}\gamma_5\, \varepsilon_{L_5}(H),$$

where the finite-L_5 sign-function approximation is

$$\varepsilon_{L_5}(H) = \frac{1 - \prod_{s=1}^{L_5} T_s(H)}{1 + \prod_{s=1}^{L_5} T_s(H)}, \qquad T_s(H) = \frac{1-H_s}{1+H_s}, \qquad H_s = \gamma_5 \frac{(b_s + c_s) D_{\text{Wilson}}}{2 + (b_s - c_s) D_{\text{Wilson}}}.$$

$$L_5 \to \infty: \qquad \varepsilon_{L_5}(H) \to \mathrm{sgn}(H), \qquad D_4 \to D_{\mathrm{ov}}.$$

Parameters b_s, c_s tune the chiral symmetry.

Previous choices of b_s and c_s

$$H_s = \gamma_5 \frac{(b_s + c_s) D_{\text{Wilson}}}{2 + (b_s - c_s) D_{\text{Wilson}}}.$$

"Shamir"

$b_s = 1, \quad c_s = 0$

Fixed coefficients.

Optimal DWF

$b_s + c_s = \omega_s$

ω_s is determined by Zolotarev approx.

Möbius DWF

$b_s = b, \quad c_s = c$

independent on fifth dimension.

This work

$\{b_s, c_s\}$ are

trainable by machine learning.

Shamir, Nucl. Phys. B 406, 90 (1993).
Chen and Chiu, Phys. Rev. D 86, 094508 (2012).
Brower, Neff, and Orginos, Comput. Phys. Commun. 220, 1 (2017).

Machine learning as parameter optimization

We "optimize" parameters $\boldsymbol{\theta} = \{b_1, \ldots, b_{L_5}, c_1, \ldots, c_{L_5}\}$ by machine learning

Present method: gradient-based optimization

Starting from an initial parameter set, we update

$$\boldsymbol{\theta}^{(n+1)} = \boldsymbol{\theta}^{(n)} - \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}.$$

- The gradient is calculated from the domain-wall operator.
- We use Adam to stabilize the stochastic updates.
- No neural network is used at this stage.

Loss function: residual mass

The training target is the residual mass evaluated with noise vectors,

$$m_{\text{res}}(\boldsymbol{\theta}) = \frac{\mathrm{Re}\langle \mathrm{Tr}\, D_4^{-1}(\boldsymbol{\theta})\rangle}{\langle \mathrm{Tr}(D_4^{\dagger} D_4(\boldsymbol{\theta}))^{-1}\rangle} - m_q.$$

Y.-C. Chen and T.-W. Chiu, Phys. Rev. D **86**, 094508 (2012).

The traces are estimated stochastically,

$$\mathrm{Tr}\,\mathcal{O} \simeq \frac{1}{N_\eta}\sum_{k=1}^{N_\eta} \eta_k^{\dagger}\mathcal{O}\eta_k.$$

Loss function

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2}\left[m_{\mathrm{res}}(\boldsymbol{\theta})\right]^2.$$

Gradient of the loss function

For $r_s = b_s$ or c_s,

$$\frac{\partial \mathcal{L}}{\partial r_s} = m_{\rm res} \frac{\partial m_{\rm res}}{\partial r_s}.$$

The required inverse-operator derivative is

$$\frac{\partial D_5^{-1}}{\partial r_s} = -D_5^{-1} \frac{\partial D_5}{\partial r_s} D_5^{-1}.$$

For the fifth-dimensional coefficients,

$$\frac{\partial (D_5)_{t,u}}{\partial b_s} = \delta_{s,t}\delta_{t,u} D_{\textsf{Wilson}}, \qquad \frac{\partial (D_5)_{t,u}}{\partial c_s} = \delta_{s,t} F_{t,u} D_{\textsf{Wilson}}.$$

Optimization workflow

1. Prepare gauge configuration U
2. Procede normal HMC on single conf.
3. Calculate the gradient of objective
4. Update $\{b_s, c_s\}$ with Adam optimizer
5. Repeat Steps 1-4
6. Measure the $m_{\text{res}}(t)$ plateau value

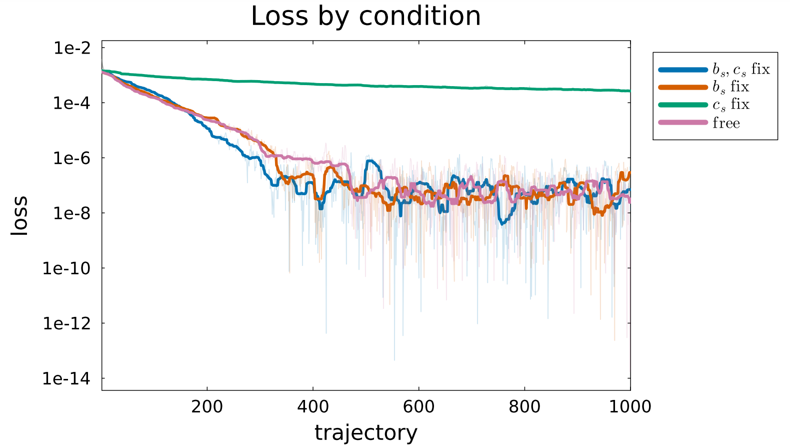
$$m_{\text{res}}(t) = \frac{\sum_{\boldsymbol{x}} \langle J_{5q}(\boldsymbol{x}, t) P(0) \rangle}{\sum_{\boldsymbol{x}} \langle P(\boldsymbol{x}, t) P(0) \rangle}.$$

T. Blum et al., Phys. Rev. D **69**, 074502 (2004).

Numerical setup

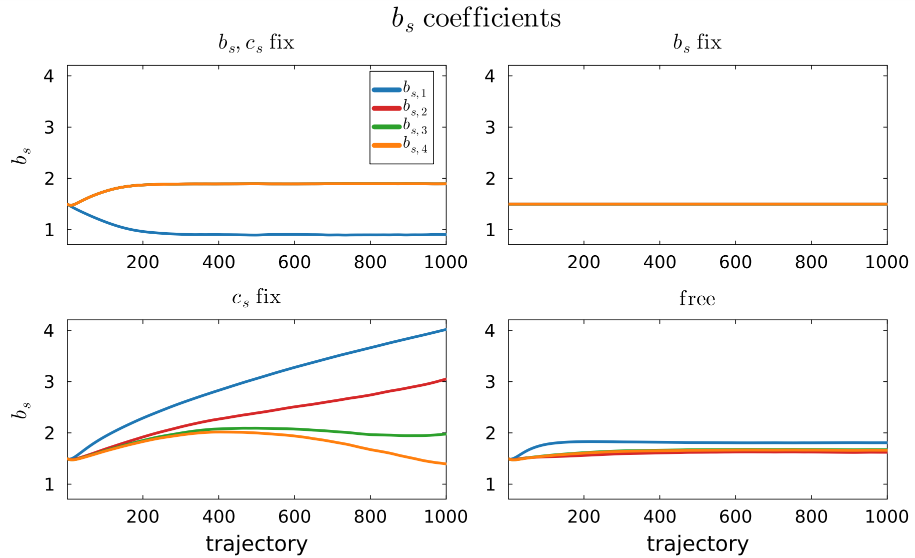
- Lattice: $16^4 \times 8$
- $\beta = 6.0$
- $m_q = 0.1$
- $M = 1.5$
- `LatticeQCD.jl`, Julia Language
- GPU calculation

Loss history: noise-vector optimization

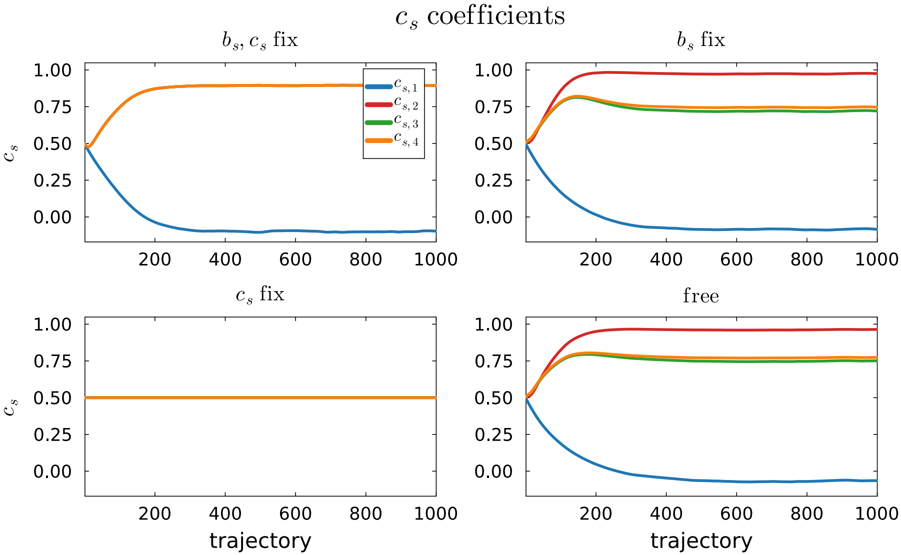


$b_s - c_s$ fixed reaches the same stochastic noise floor as the free case, while c_s fixed converges much more slowly.

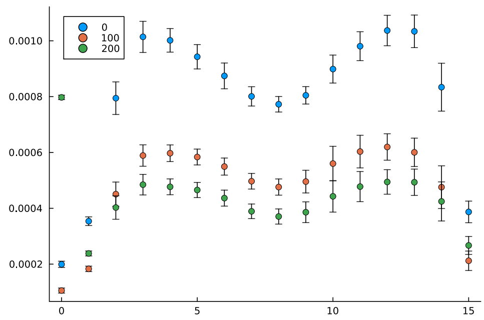
History of b_s coefficients



History of c_s coefficients



Plateau measurement after learning



Plateau check with optimized parameters.

The residual mass is extracted from the axial Ward–Takahashi identity:

$$m_{\text{res}}(t) = \frac{\sum_{\boldsymbol{x}} \langle J_{5q}(\boldsymbol{x}, t) P(\mathbf{0}, 0) \rangle}{\sum_{\boldsymbol{x}} \langle P(\boldsymbol{x}, t) P(\mathbf{0}, 0) \rangle}.$$

Current update

After the noise-vector optimization, we fix the obtained parameters $\boldsymbol{\theta}_\star$ and check the plateau on $16^4 \times L_5 = 8$.

Summary and outlook

Goal: Reduce residual chiral-symmetry breaking of domain-wall fermions at fixed L_5.

Method: Use noise vectors to estimate the residual mass, calculate its gradient, and update $\{b_s, c_s\}$ with Adam after each HMC trajectory.

Check: After learning, fix the obtained parameters and validate them using the plateau of $m_{\mathrm{res}}(t)$.

Outlook: Treat b_s and c_s also as solver-preconditioning parameters and optimize several objectives at once.

Issue: Continuum limit with fixed parameters.

Toward domain-wall fermions that are both more chiral and faster to simulate.

Backup: Code implementation

The calculation is implemented in `LatticeQCD.jl`.

- Generalized domain-wall operator
- Global and local residual-mass measurements
- Adam update with `Optimisers.jl`
- Current test size: $16^4 \times L_5 = 8$
- GPU calculation on Miyabi-G
- MPI/GPU kernels

`https://github.com/JuliaQCD`

Backup: Is fifth-dimensional reflection symmetry mandatory?

For the rational approximation itself

A product of transfer matrices can be defined without imposing $b_s = b_{L_5+1-s}$ and $c_s = c_{L_5+1-s}$.

For the standard dynamical five-dimensional formulation

Reflection symmetry is a practical structural constraint because it preserves the usual $\gamma_5 R_5$-Hermiticity and the corresponding convenient HMC formulation.

In the present optimization, we therefore impose the symmetry rather than treating all layers as independent.

Backup: Parameter count under reflection symmetry

For even L_5, the constraints

$$b_s = b_{L_5+1-s}, \qquad c_s = c_{L_5+1-s}$$

reduce the number of independent parameters from $2L_5$ to L_5.

For $L_5 = 8$,

$$\boldsymbol{\theta} = \{b_1, b_2, b_3, b_4, c_1, c_2, c_3, c_4\}.$$

This reduction acts as an inductive bias and improves the stability of stochastic optimization.