# Two-link Staggered Quark Smearing in QUDA

Steven Gottlieb[1], **Hwancheol Jeong**[1,†], Alexei Strelchenko[2]

[1] *Indiana University,* [2] *Fermilab*

## Abstract


- For staggered quarks, gauge covariant smearing based on the 3D lattice Laplacian needs to introduce two-link parallel transport to preserve taste properties.
- MILC code provides this two-link staggered quark smearing, but we found that it was taking an inordinate amount of time on the CPU.
- We have implemented it in QUDA. We have also improved the algorithm to reuse two-link matrices stored in the memory.


## Two-link staggered quark smearing

- 3D lattice Laplacian:

$$\nabla^2(x,y) = \sum_{\mu=1}^{3} \left[U_\mu(x)\,\delta_{x+\hat{\mu},\,y} + U_\mu^\dagger(x-\hat{\mu})\,\delta_{x-\hat{\mu},\,y}\right] - 6\,\delta_{x,y} \tag{1}$$

- Taste-preserving (two-link) 3D lattice Laplacian:

$$\nabla^2_{\text{two}}(x,y) \equiv \sum_{\mu=1}^{3} \left[\boldsymbol{V_\mu(x)}\,\delta_{x+2\hat{\mu},\,y} + \boldsymbol{V_\mu^\dagger(x-2\hat{\mu})}\,\delta_{x-2\hat{\mu},\,y}\right] - 6\,\delta_{x,y} \tag{2}$$

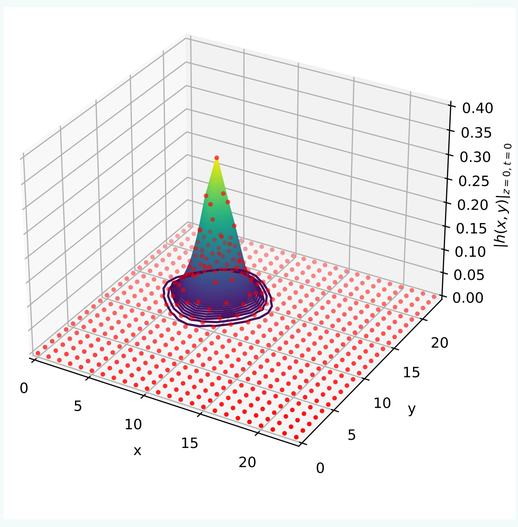
  where two-link $\boldsymbol{V_\mu(x)} \equiv U_\mu(x)U_\mu(x+\hat{\mu})$.

- Gaussian smearing for staggered quark $\psi(x)$:

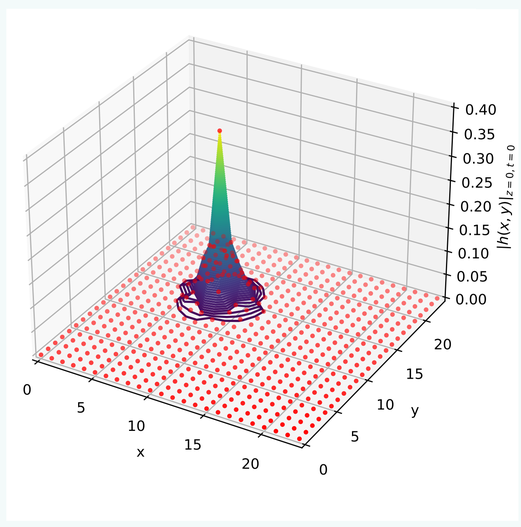
$$\widetilde{\psi} = \left(1 + \frac{\sigma}{n}\nabla^2_{\text{two}}\right)^n \psi \tag{3}$$

  where $n \in \mathbb{Z}$, $n > 0$ and $\sigma \in \mathbb{R}$ are tunable parameters.

- For $n \gg 1$, $\widetilde{\psi} \sim \exp\left(\sigma\nabla^2_{\text{two}}\right)\psi$
- For $U = 1$, $\widetilde{\psi}$ follows the **Gaussian distribution**.




Unit gauge ($U = 1$)

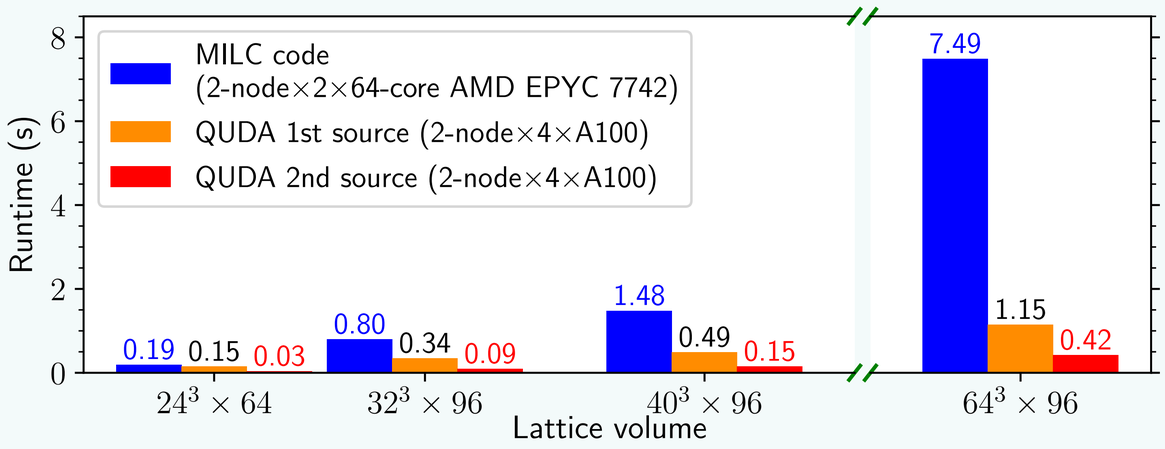



HISQ gauge

## GPU implementation & algorithm improvement

| **MILC code** | **QUDA (this work)** |
|---|---|
| CPU | GPU<br>⇒ **Higher FLOPS** |
| Computes two consecutive parallel transports every iteration<br>$(U_\mu(x)U_\mu(x+\hat{\mu})\psi(x+2\hat{\mu}))$ | **Computes two-link once** and **reuse** it for all iterations and even for different sources and sinks<br>$(V_\mu(x)\psi(x+2\hat{\mu}))$<br>⇒ **Less # of FLOPS**<br>⇒ **Less communication** |

- Performance improvement (Big Red 200, $n = 50$)




- 2nd source (and so on) reuses two-link in the memory.
- Our QUDA code on GPU is faster than the MILC code on CPU by 600∼1800%. The bigger the lattice volume, the greater the improvement.

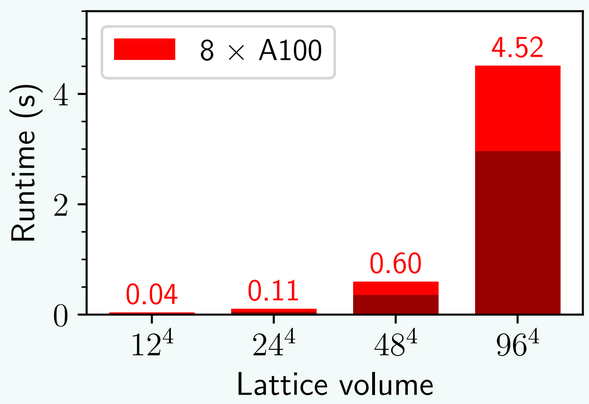
- In practice, we have many sources and sinks that can share the same two-link.

† speaker, sonchac@gmail.com

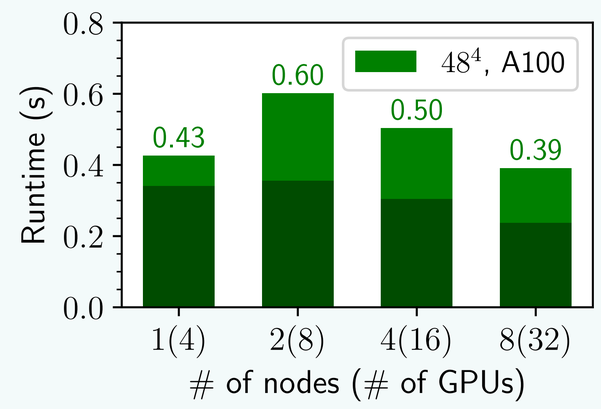
## Performance on NVIDIA GPU

- Runtime to smear three (different color) wall sources, $n = 50$
- Unshaded: two-link calculation, shaded: smearings

- Volume scalability (Big Red 200)

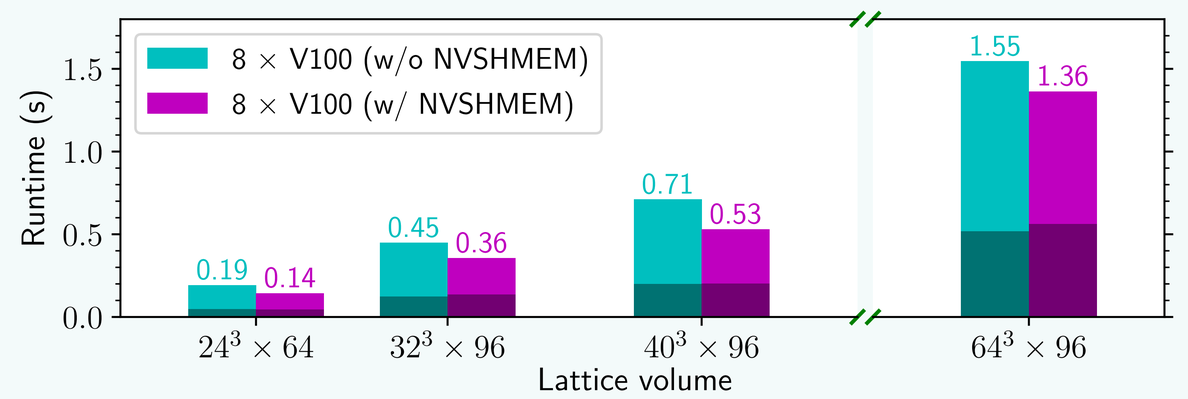



- Strong scalability (Big Red 200)




▶ Volume scalability looks good. But strong scalability is poor.

- Smearing a source with and without NVSHMEM (Summit)

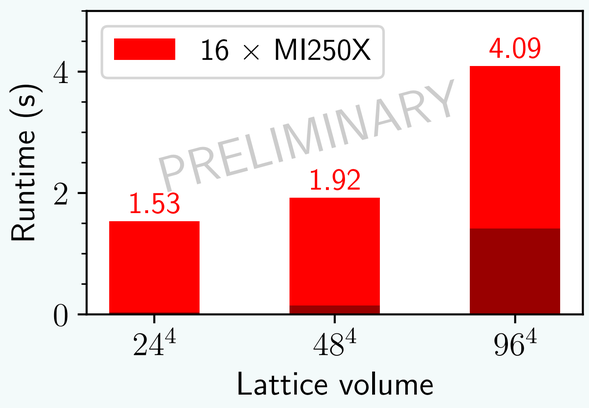



▶ NVSHMEM improves the two-link computation speed by 30∼50%.

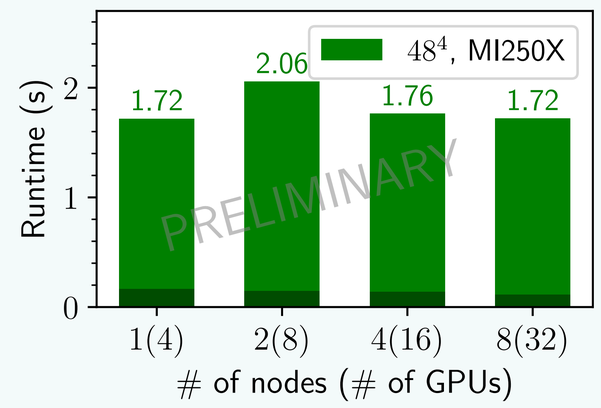
## Performance on AMD GPU

- Runtime to smear three (different color) wall sources, $n = 50$
- Unshaded: two-link calculation, shaded: smearings

- Volume scalability (Crusher)



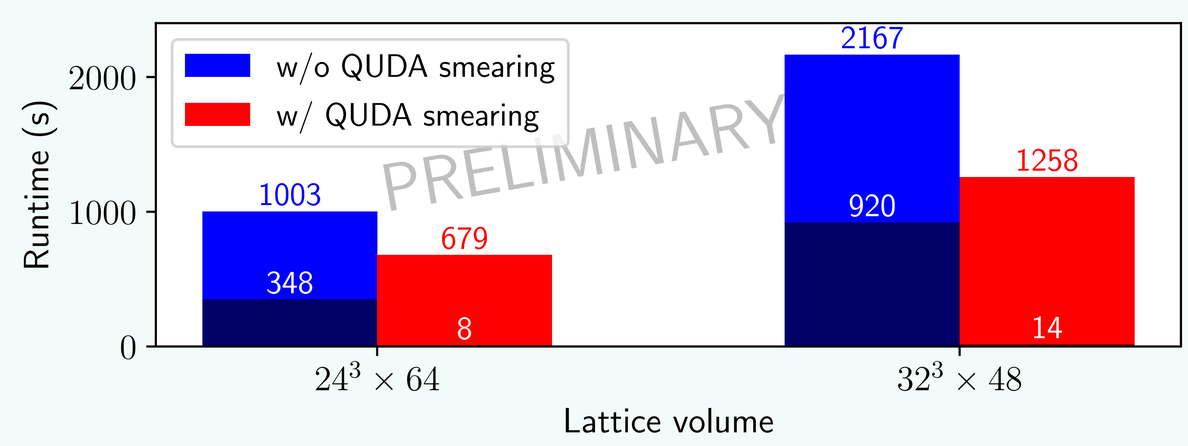

- Strong scalability (Crusher)




▶ Two-link computation is much slower than that on NVIDIA GPUs.

## [Application] Baryon correlator calculation

- 72 source and sink smearings ($n = 30$)
- Shaded: total smearing time
- Personal server similar to Cooley: 2 MPI × ( 6 OpenMP threads + 1 K80 )




▶ Total smearing time is reduced by around 400∼700%.

## Conclusion & Plan

- We have significantly reduced the cost to smear staggered quark fields.
- Gaussian smearing is no longer a bottleneck in the baryon correlator calculation.
- Need to improve the scalability and the two-link computation on AMD GPUs.
- It will be available in the develop branch of QUDA soon. (Pull request is open.)