MRHS multigrid solver for Wilson clover fermions

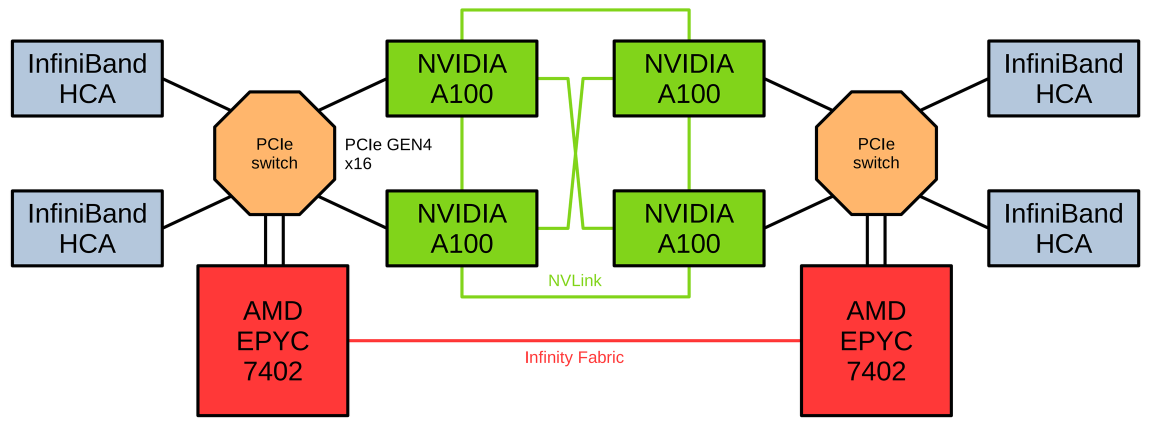
Daniel Richtmann, Nils Meyer, Tilo Wettig
Department of Physics, University of Regensburg, Germany

Multiple right-hand sides (MRHS): Motivation and Grid implementation

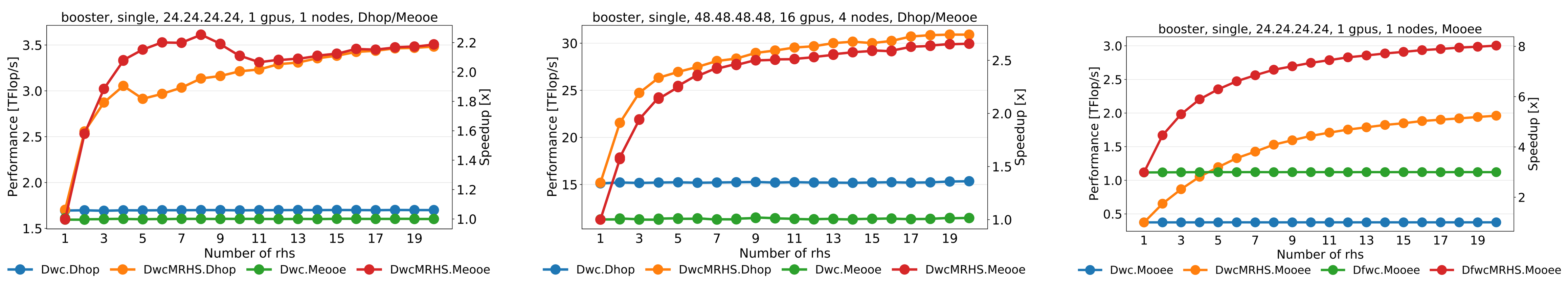
- Multigrid (MG) in conflict with ever-increasing parallelism requirements
- Small local volume on coarser levels cripples performance on GPUs already now
- Reformulate problem to process large number of d.o.f.s in parallel → solve multiple RHS simultaneously
- Benefits of MRHS:
  - More cache reuse → improved compute performance
  - Larger and fewer messages in comms → shift from latency problem towards bandwidth problem
- Grid supports native 5d fields and operations for DWF
- DWF is implicitly an MRHS system → groundwork already in place
- Idea: Formulate MRHS as a 5d problem with RHS index in fifth dimension
  - Can simply call DWF hopping term for MRHS Wilson-Clover operator
  - Program all other MG kernels in analogy to DWF operator (no data-layout transformation necessary inside solver)

JUWELS Booster architecture

- 936 compute nodes, each equipped with
  - Dual-socket AMD EPYC 7402
  - 4 × NVIDIA A100 (40 GB HBM2 each), interconnected via NVLink
  - 4 × InfiniBand HDR Host Channel Adapter (HCA), each 200 Gbit/s per direction
- Fat tree network topology within cells of 48 nodes each



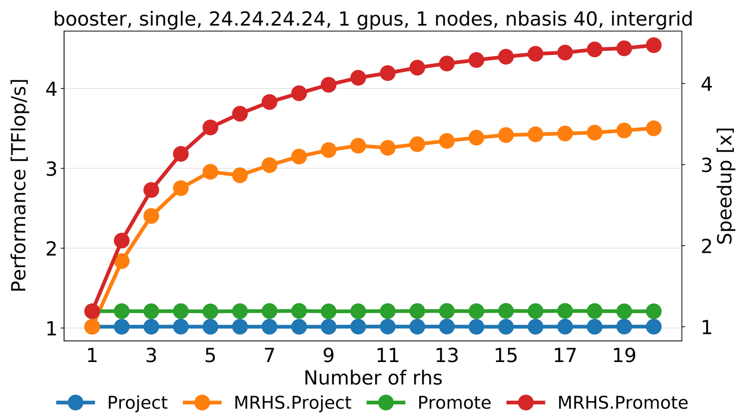
Hopping and clover term (fine level)



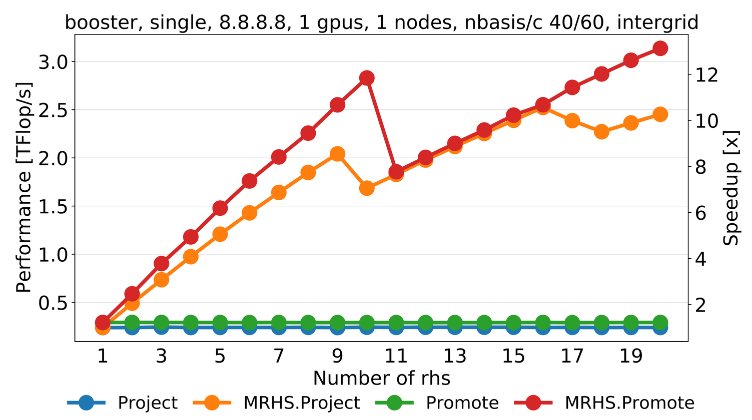
- D_{hop} = hopping term, M_{eooe} = hopping term between checkerboarded grids, M_{ooee} = clover term
- D_{wc} = Grid default clover class, D_{fwc} = faster implementation (both scale trivially with number of GPUs)

Intergrid operators for 3-level multigrid

fine to intermediate



intermediate to coarse

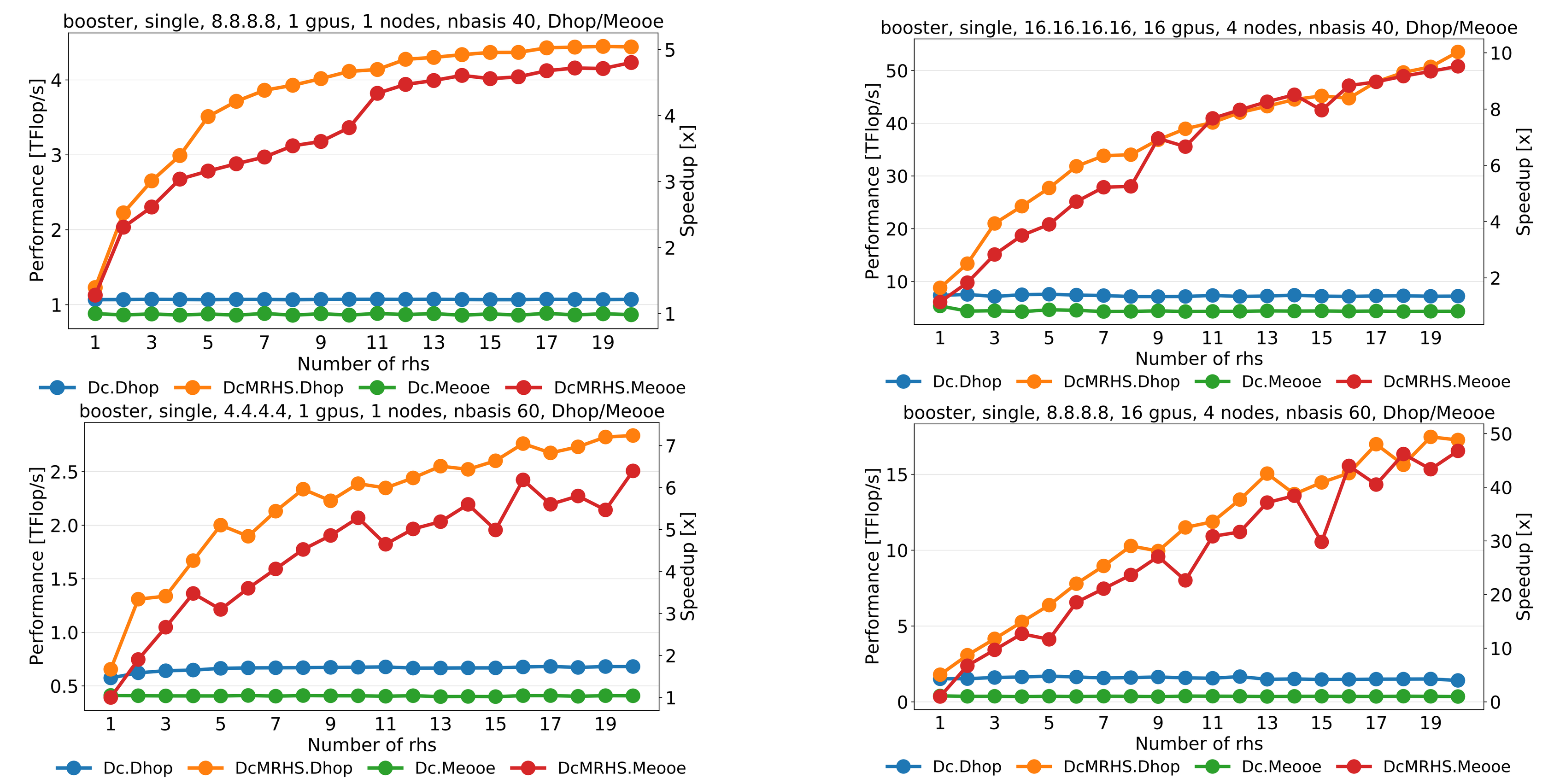


- Project: from finer to coarser level
- Promote: from coarser to finer level
- nbasis / nbasis_c = 2 × number of test vectors (cf. DD-αAMG) on fine / intermediate level

Coarse level operators D_c

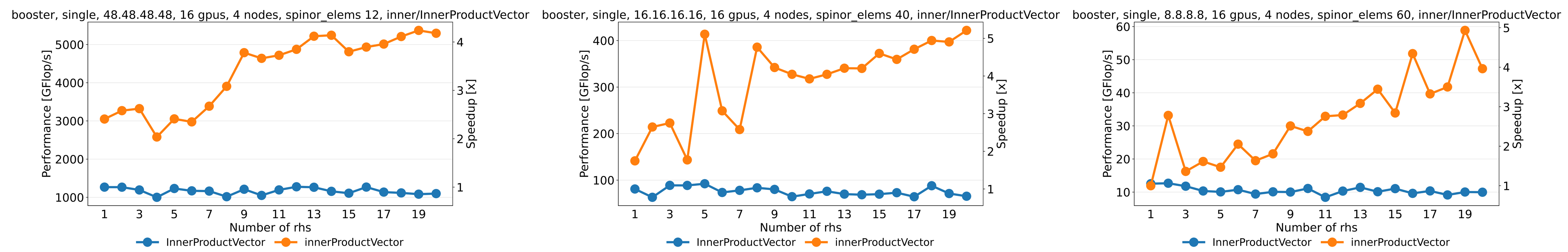
intermediate level

coarse level



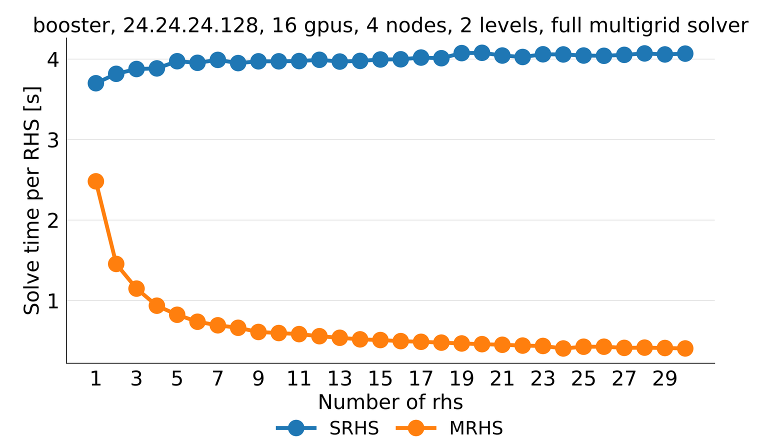
- On coarser levels, MRHS leads to larger speedups

Reductions: Inner product of two vectors



- InnerProductVector: rhs sequential inner products, innerProductVector: rhs simultaneous inner products

Overall speedup of 2-level multigrid on CLS configuration U101



Time in s for 30 RHS:		
	Fine	4.78
	Project	0.03
	Promote	0.02
	Coarse	6.73
	Rest	0.23
	Total preconditioner	11.79

- Most of wall-clock time spent on coarse level
- Speedup = 10 for 30 RHS

Summary and outlook

- Modern hardware forces us to increase parallelism
- MRHS increases parallelism straightforwardly → significant boost in performance
- No data-layout transformation necessary inside solver
- Rethink existing single-RHS algorithms?
- Outlook / Work in progress
  - More optimization efforts on inner product
  - Use eigenvector-recycling solver (GCRO-DR) on coarse level
  - Schwarz preconditioning on all MG levels to avoid communication