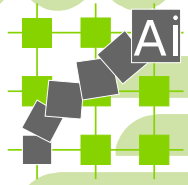
Adaptive Aggregation Based Domain Decomposition Multigrid for the Lattice Wilson Dirac Operator: Multilevel Results

Andreas Frommer, Karsten Kahl, Stefan Krieg, Björn Leder and **Matthias Rottmann**

Bergische Universität Wuppertal

July 29, 2013



Algebraic Multigrid: DD-αAMG

Task: Solution of (Clover) Wilson Dirac systems

$$D(a, m_0)\psi = \eta$$

Problem: $D(a, m_0)$ ill-conditioned for $a \to 0,\ m_0 \to m_{crit}$

Idea: Adaptive Algebraic Multigrid Approach

Two-grid error propagator for ν steps of post-smoothing

$$E_{2g}^{(\nu)} = (1 - MD)^{\nu}(1 - PD_c^{-1}P^{\dagger}D),\ D_c := P^{\dagger}DP$$

Multigrid extension by introducing recursive construction for D_c

To Do: Define interpolation P and smoother M

DD-αAMG [ArXiv:1303.1377,1307.6101]

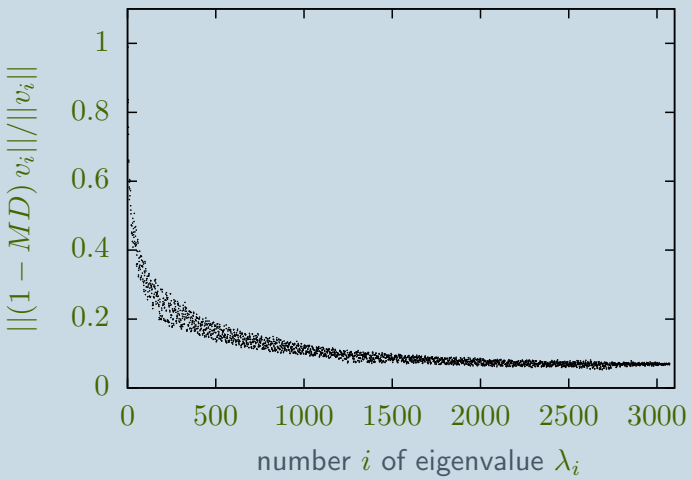
- M: Schwarz Alternating Procedure (SAP)
  [Hermann Schwarz 1870; Martin Lüscher 2003]
- P: Aggregation Based Interpolation
  [Brannick, Clark et al. 2010]

The Algebraic Multigrid Principle

Smoother: $1 - MD$

- Effective on high modes
- low modes remain

SAP



$$Dv_i = \lambda_i v_i \quad \text{with} \quad |\lambda_1| \leq \ldots \leq |\lambda_{3072}|$$

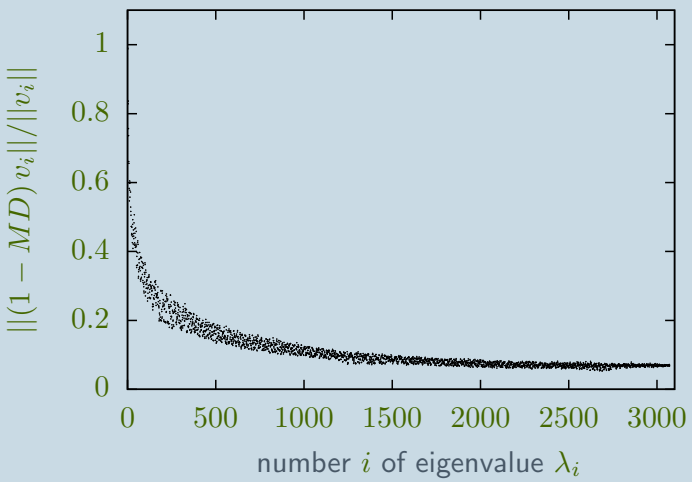
The Algebraic Multigrid Principle

Coarse-grid correction: $1 - PD_c^{-1}P^\dagger D$

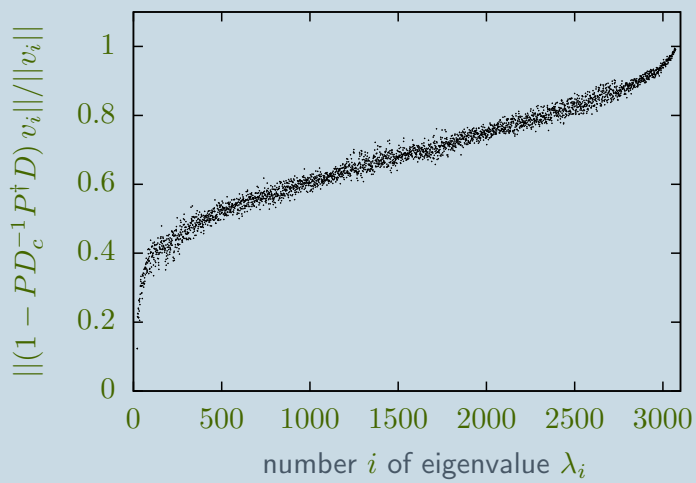
- Low modes built into interpolation P

  $\Rightarrow$ Effective on low modes

SAP



Aggregation + Low Modes



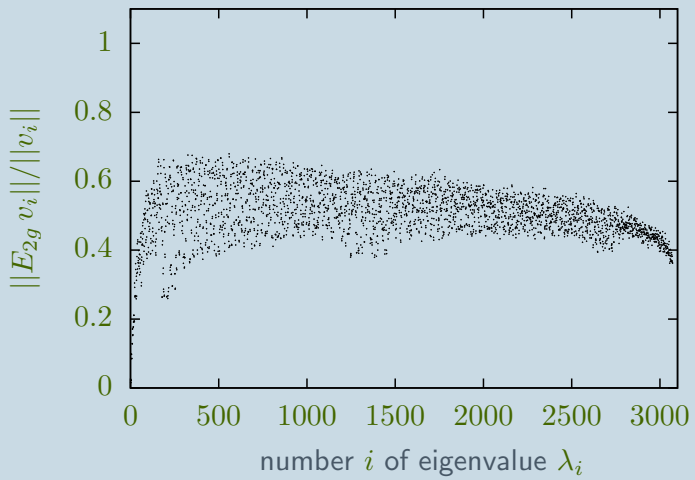
$$Dv_i = \lambda_i v_i \quad \text{with} \quad |\lambda_1| \leq \ldots \leq |\lambda_{3072}|$$

The Algebraic Multigrid Principle

Two-grid method: $E_{2g} = (1 - MD)(1 - PD_c^{-1}P^\dagger D)$

- Complementarity of smoother and coarse-grid correction
- Effective on all modes!

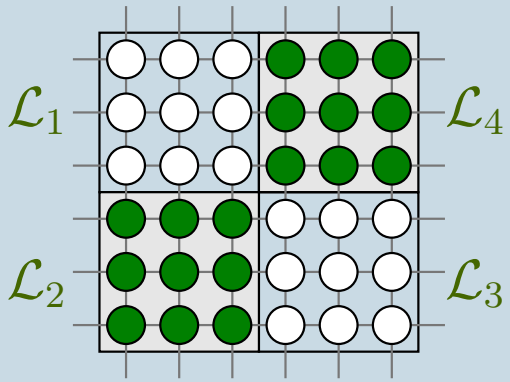
DD-αAMG



$Dv_i = \lambda_i v_i$ with $|\lambda_1| \leq \ldots \leq |\lambda_{3072}|$

SAP: Schwarz Alternating Procedure

Two color decomposition of $\mathcal{L}$



- canonical injections
  $$\mathcal{I}_{\mathcal{L}_i} : \mathcal{L}_i \to \mathcal{L}$$
- block restrictions
  $$D_{\mathcal{L}_i} = \mathcal{I}^\dagger_{\mathcal{L}_i} D \mathcal{I}_{\mathcal{L}_i}$$
- block inverses
  $$B_{\mathcal{L}_i} = \mathcal{I}_{\mathcal{L}_i} D^{-1}_{\mathcal{L}_i} \mathcal{I}^\dagger_{\mathcal{L}_i}$$

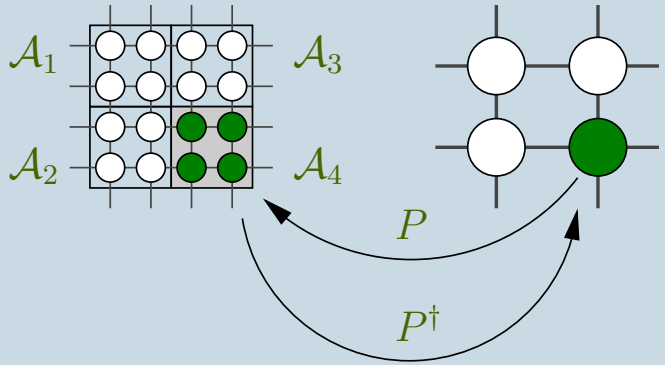
Algorithm SAP

1: in: ψ, η, ν – out: ψ
2: **for** $k = 1$ to ν **do**
3: $\quad r \leftarrow \eta - D\psi$
4: $\quad$ **for all** *green* $\mathcal{L}_i$ **do**
5: $\quad\quad \psi \leftarrow \psi + B_{\mathcal{L}_i} r$
6: $\quad$ **end for**
7: $\quad r \leftarrow \eta - D\psi$
8: $\quad$ **for all** *white* $\mathcal{L}_i$ **do**
9: $\quad\quad \psi \leftarrow \psi + B_{\mathcal{L}_i} r$
10: $\quad$ **end for**
11: **end for**

Aggregation Based Interpolation

Construction:

- Define aggregates: domain decomposition $\mathcal{A}_1, ..., \mathcal{A}_s$



- Calculate test vectors $w_1, ..., w_N$ [ArXiv:1303.1377,1307.6101]
- Decompose test vectors over aggregates $\mathcal{A}_1, ..., \mathcal{A}_s$

$$(w_1, \ldots, w_N) = \left|\begin{matrix}\\\\\\\\\end{matrix}\right| = \begin{matrix}\mathcal{A}_1\\\mathcal{A}_2\\\vdots\\\mathcal{A}_s\end{matrix} \rightarrow P = \begin{pmatrix}\mathcal{A}_1 & & \\ & \mathcal{A}_2 & \\ & & \ddots \\ & & \mathcal{A}_s\end{pmatrix}$$

Numerical Results: Configurations

id	lattice size $N_t \times N_s^3$	pion mass m_π [MeV]	CGNR iterations	shift m_0	clover term c_{sw}	provided by
1	48×48^3	135	53,932	-0.09933	1.00000	BMW-c [1,2]
2	64×64^3	135	84,207	-0.05294	1.00000	BMW-c [1,2]
3	128×64^3	270	45,804	-0.34262	1.75150	CLS [3,4]
4	128×64^3	190	88,479	-0.33485	1.90952	CLS [3,4]

[1]: S. Durr et al., Lattice QCD at the Physical Point: Simulation and Analysis Details, JHEP, 08(2011)148.

[2]: S. Durr et al., Lattice QCD at the Physical Point: Light Quark Masses, Phys. Lett. B701, (2011), pp. 265-268.

[3]: CLS, Coordinated Lattice Simulation. `https://twiki.cern.ch/twiki/bin/view/CLS/`.

[4]: P. Fritzsch et al., The Strange Quark Mass and Lambda Parameter of Two Flavor QCD, Nucl. Phys., B865 (2012), pp. 397-429.

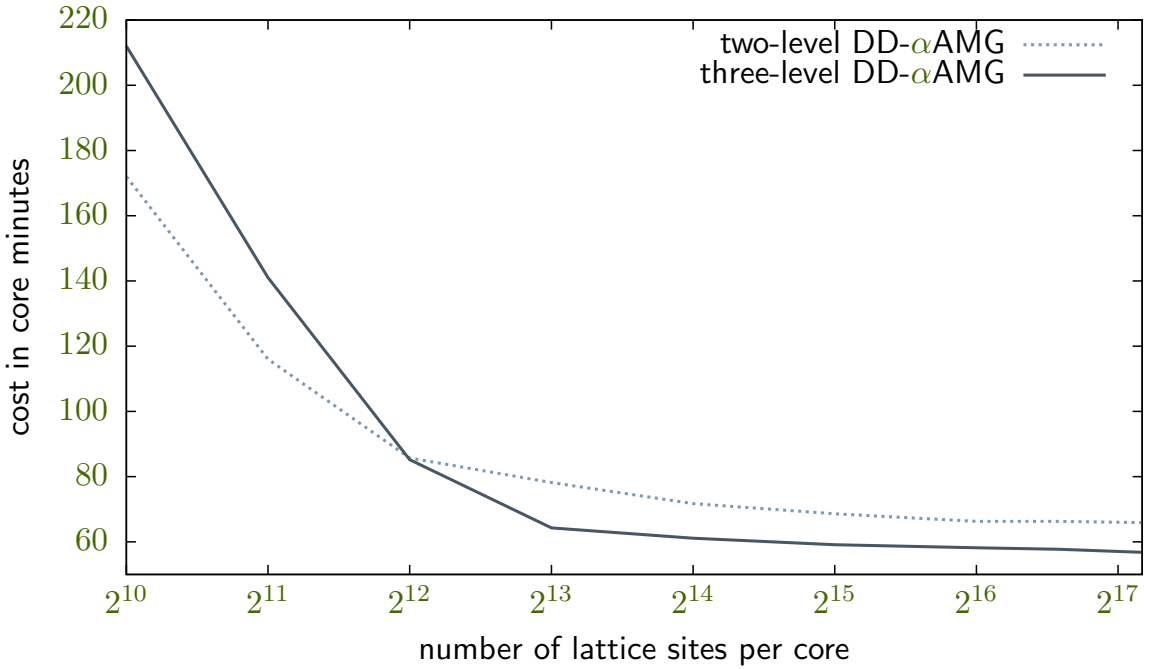
Numerical Results: Parameters

	parameter		default
setup	number of iterations	n_{inv}	6
	number of test vectors	N	20
	size of lattice-blocks for aggregates on level 1		4^4
	size of lattice-blocks for aggregates on level $\ell > 1$		2^4
	coarse system relative residual tolerance (stopping criterion for the coarse system)$^{(*)}$	ϵ	$5 \cdot 10^{-2}$
solver	restart length of FGMRES	n_{kv}	10
	relative residual tolerance (stopping criterion)	tol	10^{-10}
	number of post-smoothing steps$^{(*)}$	ν	5
	size of lattice-blocks in SAP$^{(*)}$		2^4
	number of Minimal Residual (MR) iterations to solve the local systems in SAP$^{(*)}$		3
K-cycle	maximal length$^{(*)}$		5
	maximal restarts$^{(*)}$		2
	relative residual tolerance (stopping criterion)$^{(*)}$		10^{-1}

$(*)$: same in solver and setup

Estimation of the Sweet Spot: 2 and 3 Levels

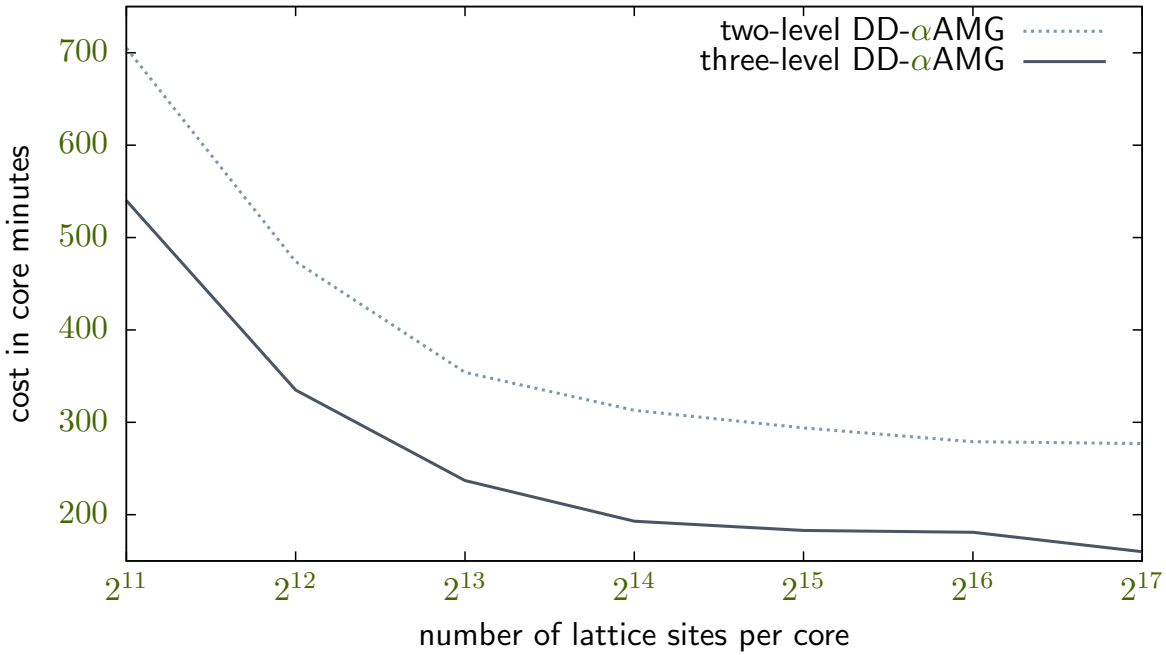
Configuration 1: 48×48^3, $m_\pi = 135$ MeV, smeared



- 36 to $5{,}184$ cores on JUROPA@JSC
- Plot ranges from nproc limit (left) to memory limit (right)
- 3 levels perform slightly better for more than 8^4 sites/core
- 2 levels perform better for less than 8^4 sites/core
- Sweet spot at 36 cores/$147{,}456$ lattice sites per core

Estimation of the Sweet Spot: 2 and 3 Levels

Configuration 2: 64×64^3, $m_\pi = 135$ MeV, smeared



- 128 to 8,192 cores on JUROPA@JSC
- 3 levels perform better in all cases
- Speed up factor becomes larger for more sites/less cores
- Sweet spot at 128 cores/2^{17} lattice sites per core

Comparison of 2, 3 & 4 Level DD-αAMG

	configuration	1	2	3	4
	lattice size	48×48^3	64×64^3	128×64^3	128×64^3
	pion mass m_π	135 MeV	135 MeV	270 MeV	190 MeV
two levels	setup time	316s	736s	630s	701s
	solve time	48.6s	130s	113s	141s
three levels	setup time	374s	744s	719s	948s
	solve time	42.6s	75.2s	74.0s	79.0s
four levels	setup time	–	806s	755s	1,004s
	solve time	–	79.8s	75.7s	79.1s
	processes	81	128	256	256
local lattice	level 1	16×16^3	32×16^3	32×16^3	32×16^3
	level 2	4×4^3	8×4^3	8×4^3	8×4^3
	level 3	2×2^3	4×2^3	4×2^3	4×2^3
	level 4	–	2×1^3	2×1^3	2×1^3

- Conf. 1 slight speed up by using 3rd level
- Conf. 2, 3, 4 significant speed up by using 3rd level
- No use for 4th level yet

Comparison with CGNR for different cost measures

Configuration 2: 64×64^3, $m_\pi = 135$ MeV, smeared

	three-level DD-αAMG	two-level DD-αAMG	CGNR
processes	128	128	8,192
solve time	75.2s	130s	816s
consumed core minutes	160	277	111,446
consumed Mflop per site	1.64	2.97	364
Gflop/s per core	2.86	2.99	0.91

CGNR vs. 3 Level DD-αAMG

- $697\times$ consumed core minutes
- $222\times$ Mflop per lattice site

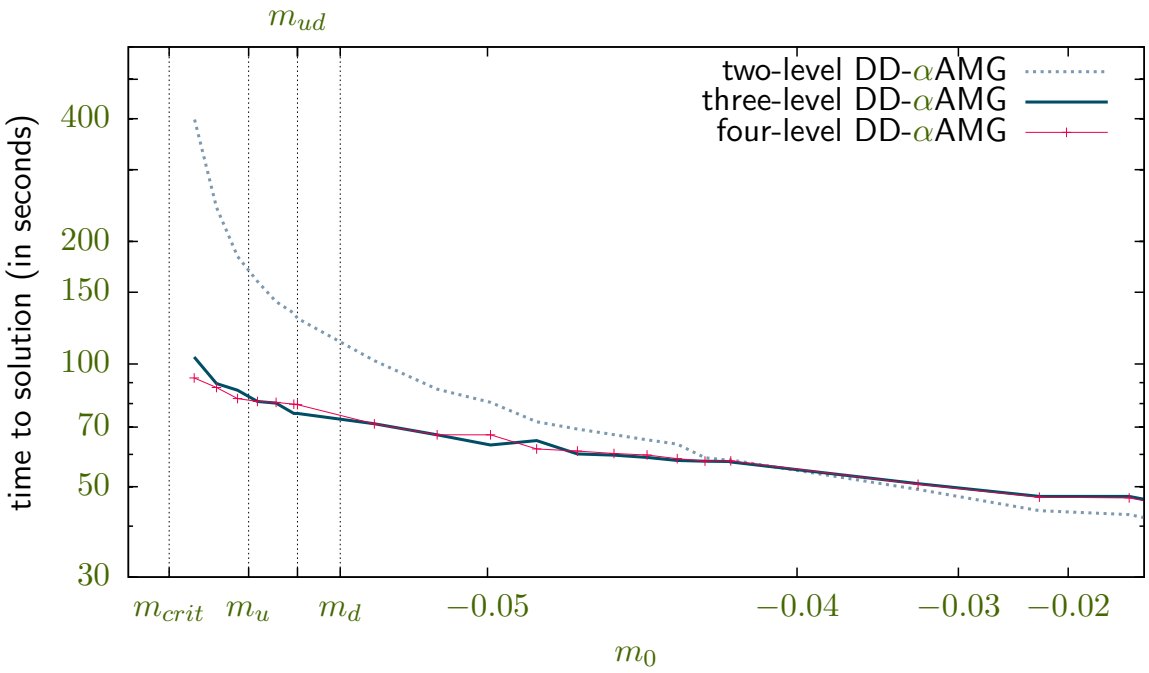
2 Level DD-αAMG vs. 3 Level DD-αAMG

- $1.73\times$ consumed core minutes
- $1.81\times$ Mflop per lattice site

Noteworthy 2.86 Gflop/s per core (12% peak) for `C-MPI-Code`

Scaling with the Mass Parameter: 2, 3 & 4 Levels

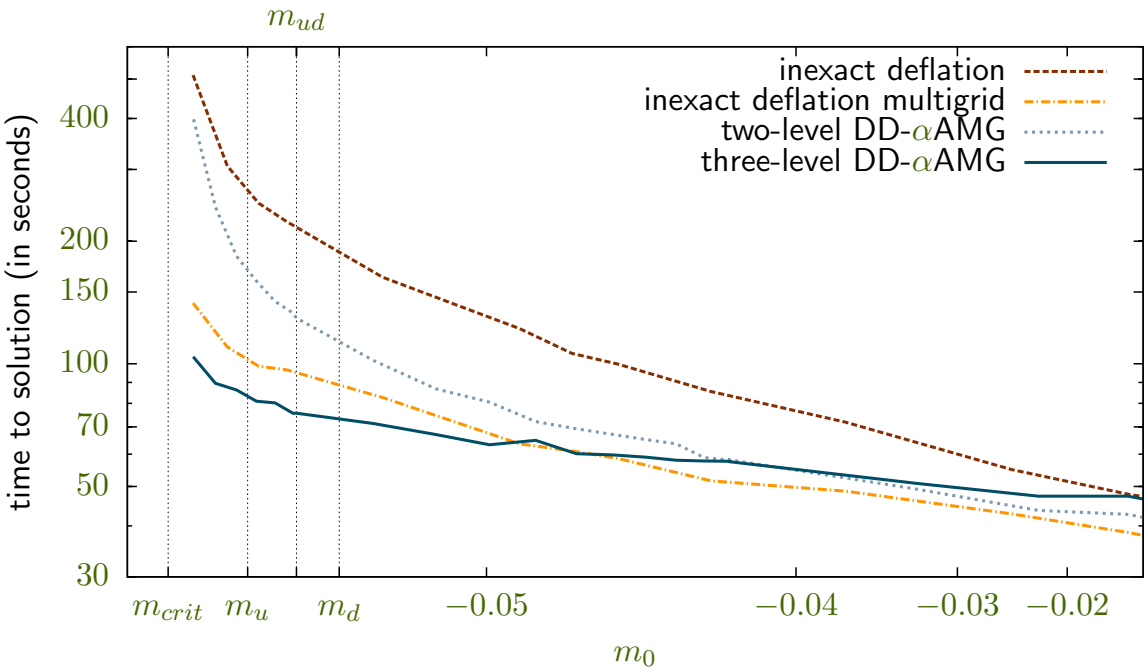
Configuration 2: 64×64^3, $m_\pi = 135$ MeV, smeared



- 3 & 4 levels clearly show improved scaling
- Speed up factor roughly 2 at m_u
- 4 levels best beyond m_u
- Consider using 4th level for future calculations

2 & 3 Level DD-αAMG, Old & New Inexact Deflation

Configuration 2: 64×64^3, $m_\pi = 135$ MeV, smeared



- 32 test vectors for inex. defl. multigrid[OpenQCD 1.2]
- Inex. defl. multigrid scales better than classical inex. defl.[DD-HMC 1.2.2] and 2 level DD-αAMG
- 3 level DD-αAMG shows best scaling behavior
- 3 levels perfom best in range of m_u and m_d

Summary & Outlook

Summary:

3 level DD-αAMG shows

- pay off for large lattice sizes and lighter pion masses
- up to factor 2 speed up over 2 level DD-αAMG
- great potential for future calculations

2 level DD-αAMG still better for calculations with

- heavier pion masses or smaller lattice sizes
- large number of processes

Outlook:

- Improve setup procedure
- Optimized versions in simulation codes of our collaborators (within SFB TR 55)

Acknowledgments

All results computed on JUROPA at
Jülich Supercomputing Centre (JSC)

Funded by Deutsche Forschungsgemeinschaft (DFG),
Transregional Collaborative Research Centre 55
(SFB TR 55)

All configurations provided by BMW-c & CLS