

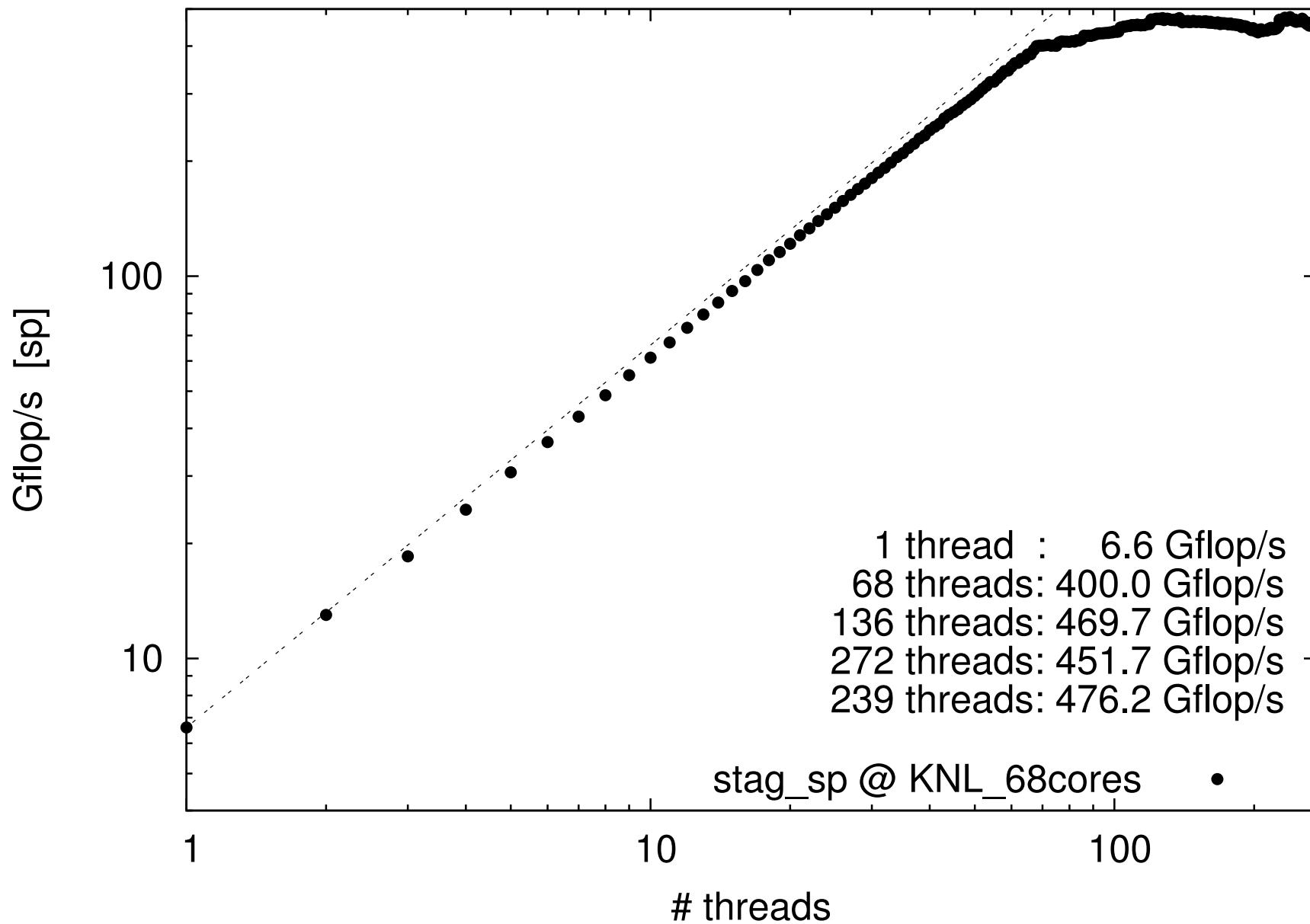
Three Dirac operators on two architectures (with one piece of code)

Stephan Dürr

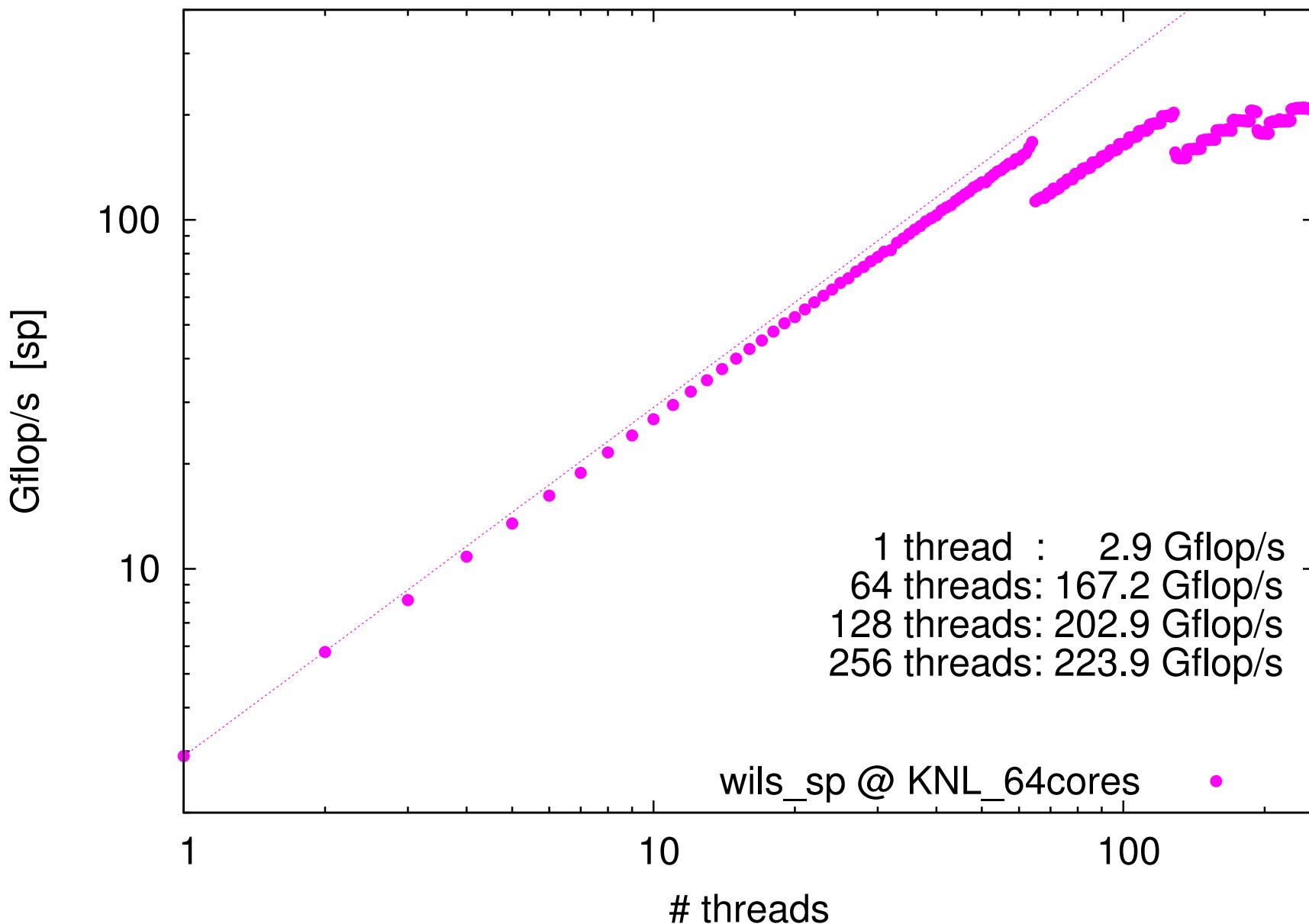


University of Wuppertal
Jülich Supercomputing Center

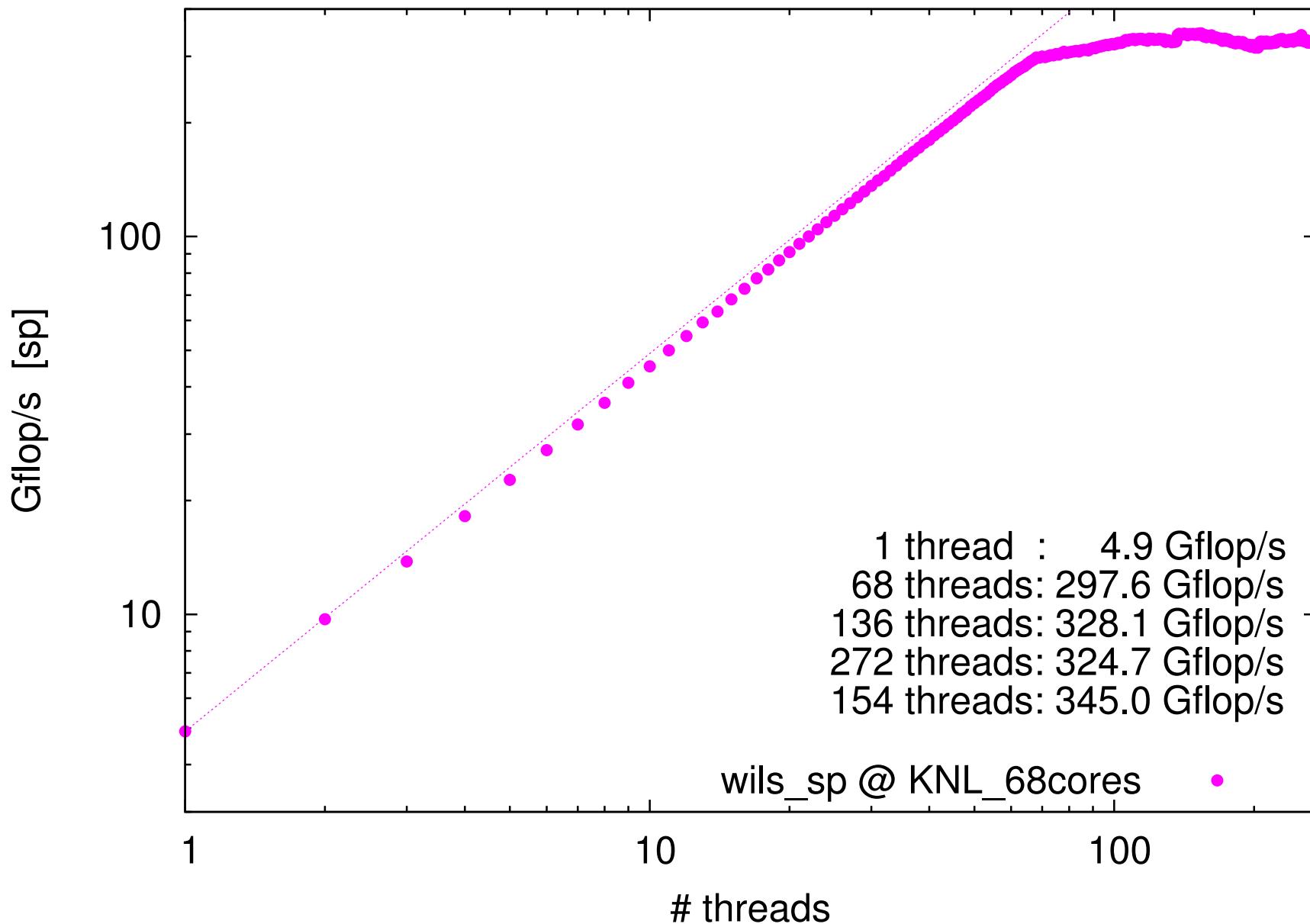
Staggered operator thread-scaling on KNL (2018 all new)



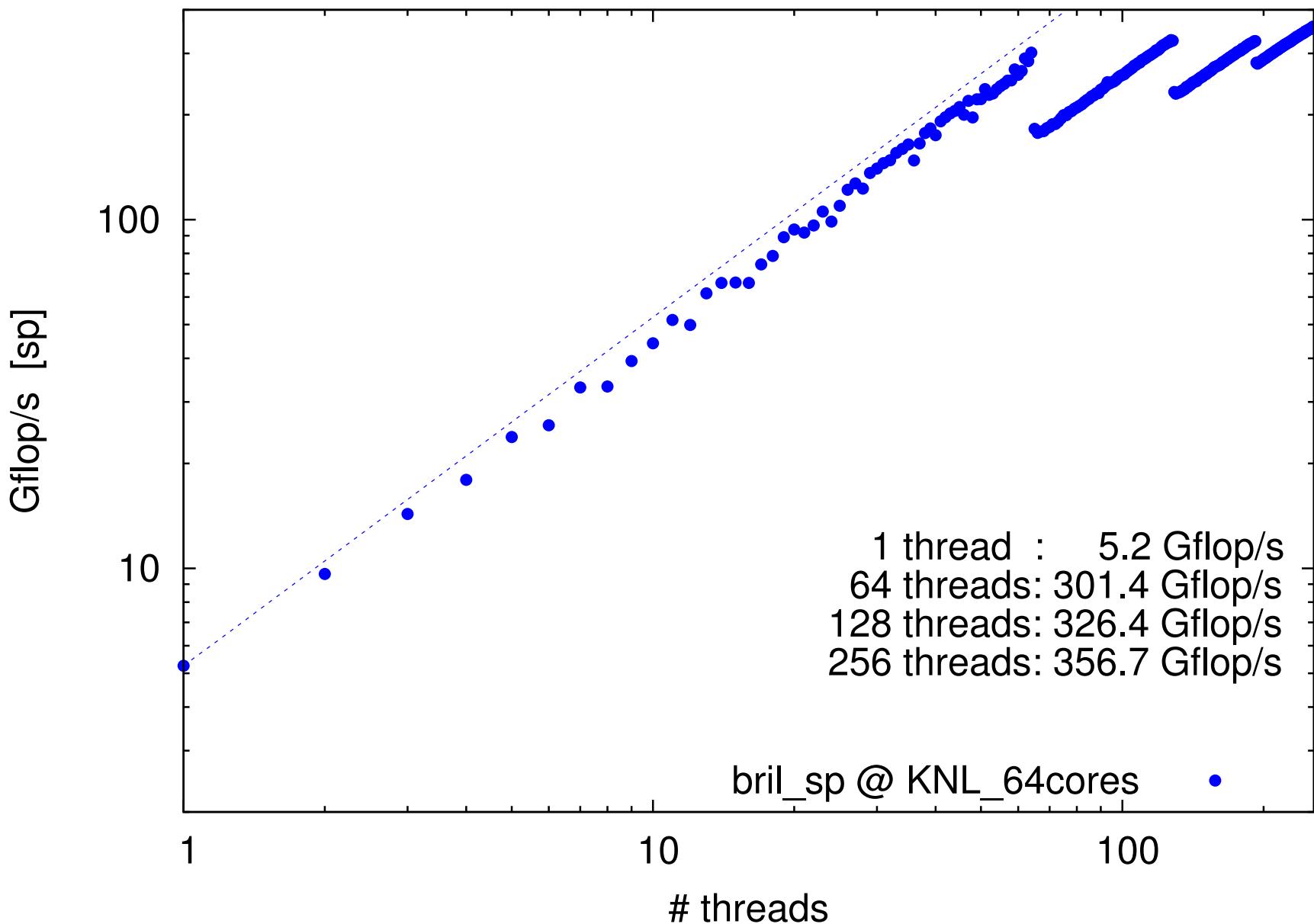
Wilson operator thread-scaling on KNL (2017 version)



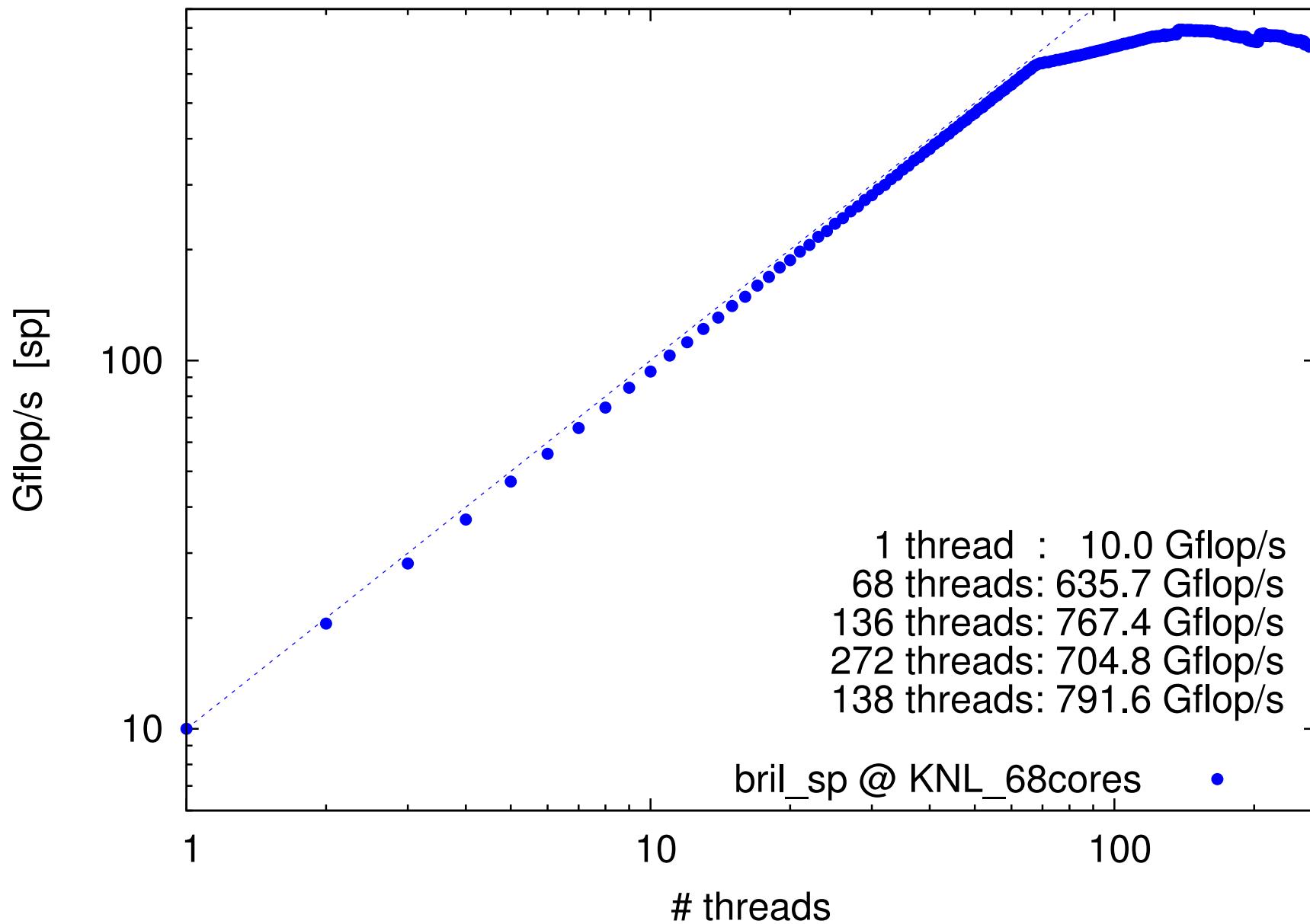
Wilson operator thread-scaling on KNL (2018 version)



Brillouin operator thread-scaling on KNL (2017 version)



Brillouin operator thread-scaling on KNL (2018 version)

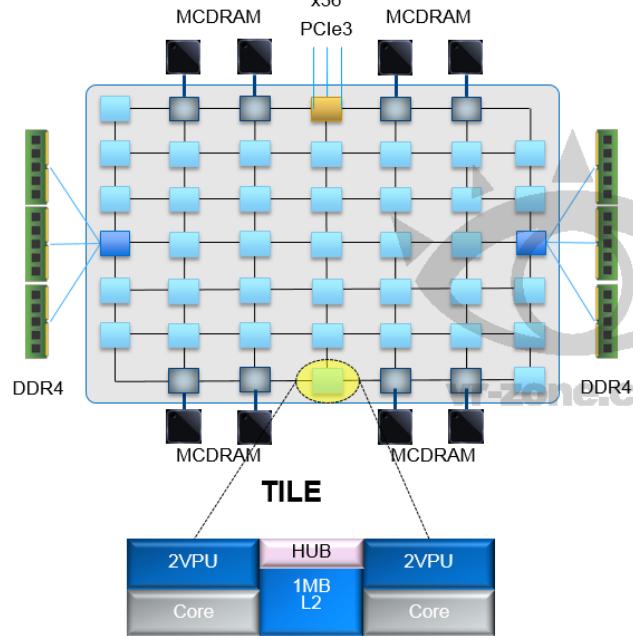


What changed ?

- more operators:
 D_{wils} , D_{bril} \longrightarrow Δ_{wils} , D_{wils} , Δ_{bril} , D_{bril} , D_{stag}
- color-spinor-rhs array layout:
 $NcNsNv$ \longrightarrow $NcNsNv$, $NsNcNv$, $NcNvNs$, $NvNcNs$, $NsNvNc$, $NvNsNc$
- accumulation variable layout:
 $NcNsNv$ \longrightarrow $NcNsNv$, $NsNcNv$, $NcNvNs$, $NvNcNs$, $NsNvNc$, $NvNsNc$
- introduce masks for `lap` and `der_i`, `der_j`, `der_k`, `der_l`
- better pragmas: `!$OMP COLLAPSE(2) SCHEDULE(dynamic)`
- better hardware: 64 \longrightarrow 68 cores, 1.3 \longrightarrow 1.4 GHz

Intel KNL: architecture overview

Knights Landing Processor Architecture



Up to 72 Intel Architecture cores based on Silvermont (Intel® Atom processor)

- Four threads/core
- Two 512b vector units/core
- Up to 3x single thread performance improvement over KNC generation

Full Intel® Xeon processor ISA compatibility through AVX-512 (except TSX)

6 channels of DDR4 2400 MHz -up to 384GB

36 lanes PCI Express® Gen 3

8/16GB of high-bandwidth on-package MCDRAM memory >500GB/sec

200W TDP

	Knights Landing (Estimate)	Knights Corner	Haswell Client	Ivy Bridge Server	Unit
Process Node	14	22	22	22	nm
Base Frequency	1.4	1.238	3.6	2.7	GHz
TDP	300	300	82	130	W
Cores	72	61	4	12	
DP FLOP/core	32	16	16	8	FLOP/cycle
DP GFLOPs/core	44.8	19.8	57.6	21.6	GFLOP/s
L1D Capacity	32	32	32	32	KB
L1D Read	128	64	64	32	B/cycle
L1D Read BW	179.2	79.2	230.4	86.4	GB/s
L2 Capacity/core	256	512	256	256	KB
L2 Read/core	64	64	64	32	B/cycle
L2 Read BW/core	89.6	79.2	230.4	86.4	GB/s
L3 Capacity/core	2	NA	2	2.5	MB
L3 Read/core	32	NA	32	32	B/cycle
L3 Read BW/core	44.8	NA	115.2	86.4	GB/s
Total DP GFLOP/s	3225.6	1208.288	230.4	259.2	GFLOP/s
Total L1 Read BW	12.902	4.833	0.922	1.037	TB/s
Total L2 Read BW	6.451	4.833	0.922	1.037	TB/s
Total L3 Read BW	3.226	NA	0.128	0.384	TB/s
Total LLC Capacity	144	30.5	8	30	MB

- `export KMP_AFFINITY=scatter`
- `numactl --membind 1 ./test_knl` to access MCDRAM
- shared variables initialized with same `NUM_THREADS` as operator ("first touch")
- L3-read 44.8 GB/s/core, sp-peak 89.6 Gflop/s/core \Rightarrow ideally 0.5 bytes/flop

Code suite guidelines

- o Fortran 2008 with stride-notation (like matlab)
- o Avoid derived data types; use 4D arrays for vectors (3D for staggered)
- o All parameters known at compile time:
 $N_c \sim 3$ colors, $N_s = 4$ spinors, $N_v \sim 12$ right-hand-sides, $N_x N_y N_z N_t$ sites
- o OpenMP pragmas for shared-memory parallelization
- o OpenMP pragmas for SIMD heuristics overwrite

• Fortran loop/slot ordering (“column major memory layout”)

Declare `complex(kind=sp)` array `V(Nc,Nc,4,Nx,Ny,Nz,Nt)` for smeared gauge field

- * `V(:,:,3,x,y,z,t)` is link in 3-direction at (x, y, z, t) ; contiguous in memory
- * `F(:,:,3,x,y,z,t)` is field-strength F_{14} at (x, y, z, t) ; contiguous in memory

Declare `complex(kind=sp)` array for vector, e.g. `vec(Nc,4,Nv,Nx*Ny*Nz*Nt)`:

- | | | |
|--|-----------------|--------------------------------|
| * <code>color(1:Nc)</code> | innermost/first | ← unroll (if possible) |
| * <code>spinor(1:4)</code> | next | ← unroll (if possible) |
| * <code>rhs-idx(1:Nv)</code> | next | ← SIMD via OMP pragma |
| * <code>position(1:Nx*Ny*Nz*Nt)</code> | outermost/last | ← distribute among OMP threads |

- Layout options for vec and site

Above layout prevents SIMD-loop over one (innermost) space-time direction.
Second best option seems SIMD-loop over $N_v \sim 12$ right-hand-sides.
Unclear which internal ordering would be best; leaves 6×6 or 2×2 choices:

Wilson vector layout:	<code>vec(Nc,04,Nv,Nx*Ny*Nz*Nt)</code> <code>vec(04,Nc,Nv,Nx*Ny*Nz*Nt)</code> <code>vec(Nc,Nv,04,Nx*Ny*Nz*Nt)</code> <code>vec(Nv,Nc,04,Nx*Ny*Nz*Nt)</code> <code>vec(04,Nv,Nc,Nx*Ny*Nz*Nt)</code> <code>vec(Nv,04,Nc,Nx*Ny*Nz*Nt)</code>
Wilson accum. variable:	<code>site(Nc,04,Nv)</code> <code>site(04,Nc,Nv)</code> <code>site(Nc,Nv,04)</code> <code>site(Nv,Nc,04)</code> <code>site(04,Nv,Nc)</code> <code>site(Nv,04,Nc)</code>
Staggered vector layout:	<code>suv(Nc,Nv,Nx*Ny*Nz*Nt)</code> <code>suv(Nv,Nc,Nx*Ny*Nz*Nt)</code>
Staggered accum. variable:	<code>site(Nc,Nv)</code> <code>site(Nv,Nc)</code>

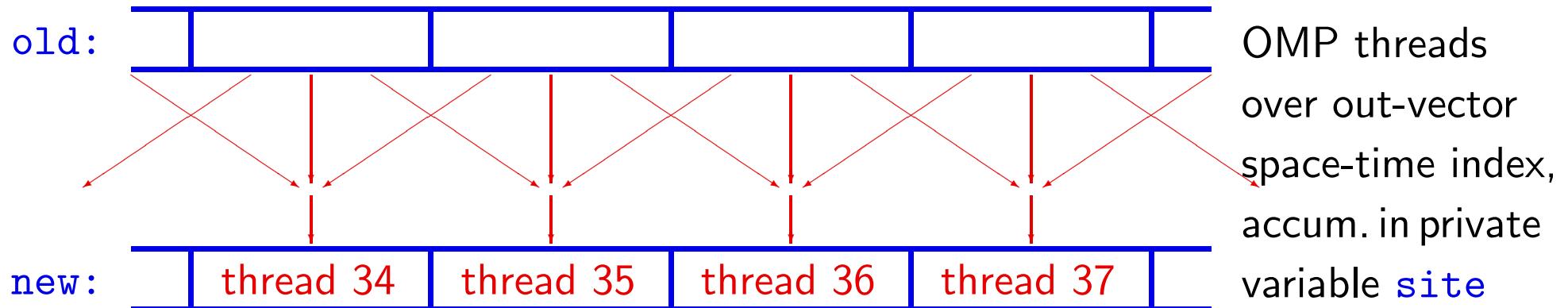
- **Timings for all varieties of app_blap_sp on KNL_68**

Times in sec per rhs for $-\frac{1}{2}\Delta^{\text{std}} + \frac{1}{2}m^2$ on $34^3 \times 68$ lattice with $N_c = 3, N_s = 4, N_v = 12$:

1	0.0178	0.0210	0.0183	0.0197	0.0183	0.0201	0.0234	0.0203	0.0206	0.0209	0.0222	0.0214
2	0.0190	0.0221	0.0192	0.0209	0.0217	0.0216	0.0210	0.0208	0.0204	0.0201	0.0205	0.0207
3	0.0182	0.0210	0.0181	0.0190	0.0209	0.0201	0.0230	0.0202	0.0213	0.0207	0.0202	0.0212
4	0.0178	0.0201	0.0208	0.0215	0.0205	0.0210	0.0245	0.0252	0.0253	0.0251	0.0250	0.0239
5	0.0188	0.0220	0.0192	0.0208	0.0217	0.0217	0.0207	0.0205	0.0200	0.0200	0.0206	0.0208
6	0.0179	0.0187	0.0191	0.0213	0.0186	0.0211	0.0238	0.0243	0.0224	0.0251	0.0235	0.0244
7	0.0202	0.0191	0.0179	0.0214	0.0207	0.0208	0.0259	0.0253	0.0221	0.0200	0.0203	0.0201
8	0.0212	0.0215	0.0179	0.0182	0.0180	0.0186	0.0252	0.0257	0.0214	0.0229	0.0214	0.0235
9	0.0205	0.0197	0.0176	0.0212	0.0201	0.0188	0.0199	0.0242	0.0230	0.0199	0.0199	0.0233
10	0.0264	0.0217	0.0267	0.0259	0.0223	0.0263	0.0244	0.0200	0.0246	0.0240	0.0205	0.0239
11	0.0193	0.0206	0.0201	0.0194	0.0175	0.0199	0.0199	0.0251	0.0198	0.0198	0.0211	0.0198
12	0.0276	0.0216	0.0266	0.0252	0.0220	0.0253	0.0251	0.0200	0.0247	0.0237	0.0202	0.0236
13	0.0233	0.0421	0.0230	0.0405	0.0415	0.0397	0.0234	0.0342	0.0236	0.0339	0.0344	0.0325
14	0.0229	0.0421	0.0234	0.0413	0.0408	0.0399	0.0235	0.0340	0.0242	0.0346	0.0335	0.0327
15	0.0184	0.0425	0.0189	0.0398	0.0422	0.0397	0.0227	0.0348	0.0254	0.0334	0.0346	0.0324
16	0.0177	0.0178	0.0183	0.0187	0.0209	0.0211	0.0245	0.0238	0.0243	0.0242	0.0254	0.0253
17	0.0229	0.0428	0.0228	0.0417	0.0404	0.0407	0.0233	0.0346	0.0235	0.0347	0.0332	0.0329
18	0.0181	0.0178	0.0186	0.0210	0.0211	0.0212	0.0241	0.0234	0.0241	0.0238	0.0255	0.0251
19	0.0190	0.0217	0.0180	0.0253	0.0234	0.0242	0.0255	0.0202	0.0216	0.0228	0.0207	0.0211
20	0.0183	0.0211	0.0184	0.0184	0.0177	0.0179	0.0241	0.0247	0.0203	0.0205	0.0233	0.0236
21	0.0190	0.0219	0.0195	0.0247	0.0235	0.0242	0.0206	0.0202	0.0203	0.0225	0.0206	0.0211
22	0.0189	0.0226	0.0202	0.0247	0.0232	0.0242	0.0204	0.0208	0.0208	0.0235	0.0209	0.0218
23	0.0183	0.0203	0.0187	0.0186	0.0178	0.0183	0.0230	0.0248	0.0207	0.0219	0.0234	0.0227
24	0.0193	0.0227	0.0203	0.0252	0.0233	0.0243	0.0203	0.0211	0.0208	0.0235	0.0209	0.0215
	0.0177	0.0178	0.0176	0.0182	0.0175	0.0179	0.0199	0.0200	0.0198	0.0198	0.0199	0.0198

col: **vec**-layout **NcNsNv, NsNcNv, NcNvNs, NvNcNs, NsNvNc, NvNsNc** at 136 (1:6) and 272 (7:12) threads
row: **site**-layout **NcNsNv, NsNcNv, NcNvNs, NvNcNs, NsNvNc, NvNsNc**, 4 loop nestings & SIMD choices
best 0.0175=355.0 Gflop/s, worst 0.0428=145.5 Gflop/s, min-over-rows almost col-independent

- Implicit avoidance of write-collisions among threads



- Best of breed ansatz

app_blap_sp: old_sp \rightarrow new_sp for any of the $6 \cdot 6 \cdot 4 = 144$ routine options.
Any of these routines can be used with 1:272 threads (mostly 68,136,272 on KNL_68).
Overall, this gives 39168 options for app_blap_sp; hard to predict which one is best.
Solution: compile all 144 routines, test-run with reasonable subset of thread options.
Overstating things, one might make a case for “machine learning” ;-)

- **Compilation (ifort \geq 17.2)**

```
ifort [...] -qopenmp -O2 -xmic-avx512 -o test_knl test_knl.f90  
ifort [...] -qopenmp -O2 -xcore-avx2 -o test_knl test_knl.f90
```

Staggered kernel details

PARAMETERS: Nx,Ny,Nz,Nt, Nc,Nv, sp,dp, i_sp,i_dp !!! known at compile time

```
!!! note: eta1=1, eta2=(-1)^(x1), eta3=(-1)^(x1+x2), eta4=(-1)^(x1+x2+x3)
!$OMP PARALLEL DO DEFAULT(private) FIRSTPRIVATE(mass) SHARED(old,new,V) NUM_THREADS(Nthr) SCHEDULE(dynamic)
do l=1,Nt; l_plu=modulo(l,Nt)+1; l_min=modulo(l-2,Nt)+1; eta4=1.000
do k=1,Nz; k_plu=modulo(k,Nz)+1; k_min=modulo(k-2,Nz)+1; eta4=-eta4; eta3=1.000
do j=1,Ny; j_plu=modulo(j,Ny)+1; j_min=modulo(j-2,Ny)+1; eta4=-eta4; eta3=-eta3; eta2=1.000
do i=1,Nx; i_plu=modulo(i,Nx)+1; i_min=modulo(i-2,Nx)+1; eta4=-eta4; eta3=-eta3; eta2=-eta2
n=sum(([i,j,k,l]-[0,1,1,1])*[1,Nx,Nx*Ny,Nx*Ny*Nz]) !!! lexicographic index
site(:, :)=cmplx(0.0,kind=sp) !!! accumulation variable
!!! direction -4 gets factor -1/2*eta4
tmp=0.5*eta4*transpose(conjg(V(:,:,:,4,i,j,k,l_min))); nsh=n+(l_min-1)*(Nz*Ny*Nx)
!$OMP SIMD
do idx=1,Nv
  forall(col=1:Nc) site(idx,col)=site(idx,col)-sum(tmp(:, :)*old(:,idx,nsh))
end do ! idx=1,Nv
!$OMP END SIMD
...
!!! add mass term and plug everything into new vector
forall(col=1:Nc,idx=1:Nv) new(col,idx,n)=site(idx,col)+mass*old(col,idx,n)
end do ! i=1,Nx
end do ! j=1,Ny
end do ! k=1,Nz
end do ! l=1,Nt
!$OMP END PARALLEL DO
```

Transfer to k-loop: $\text{eta4}=\text{float}(1-2*\text{modulo}(k,2))$, $\text{l_plu}=\dots$, $\text{l_min}=\dots$

Introduce **COLLAPSE(2)** on l-loop (after transfer is done).

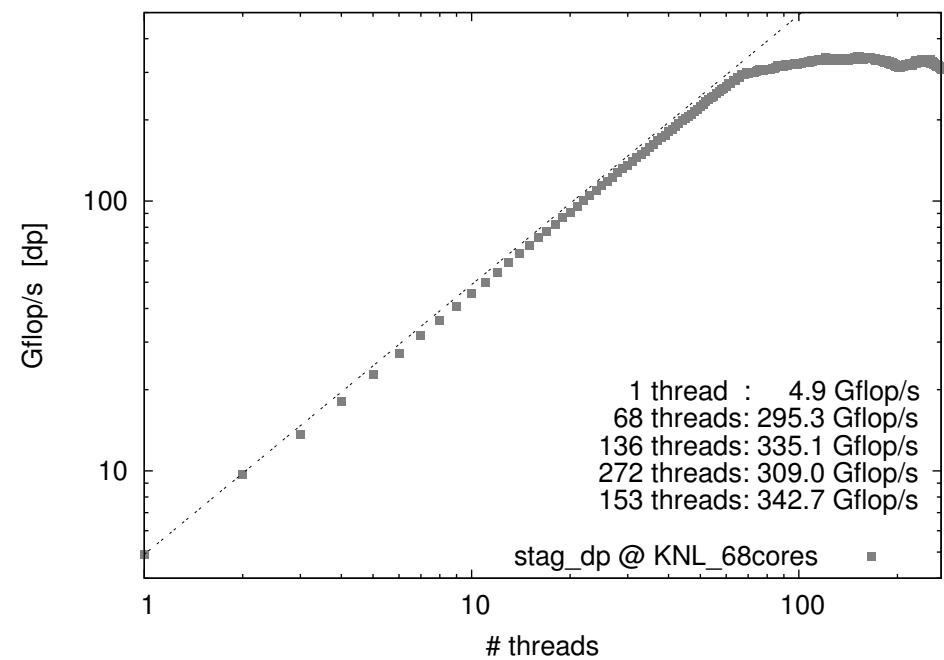
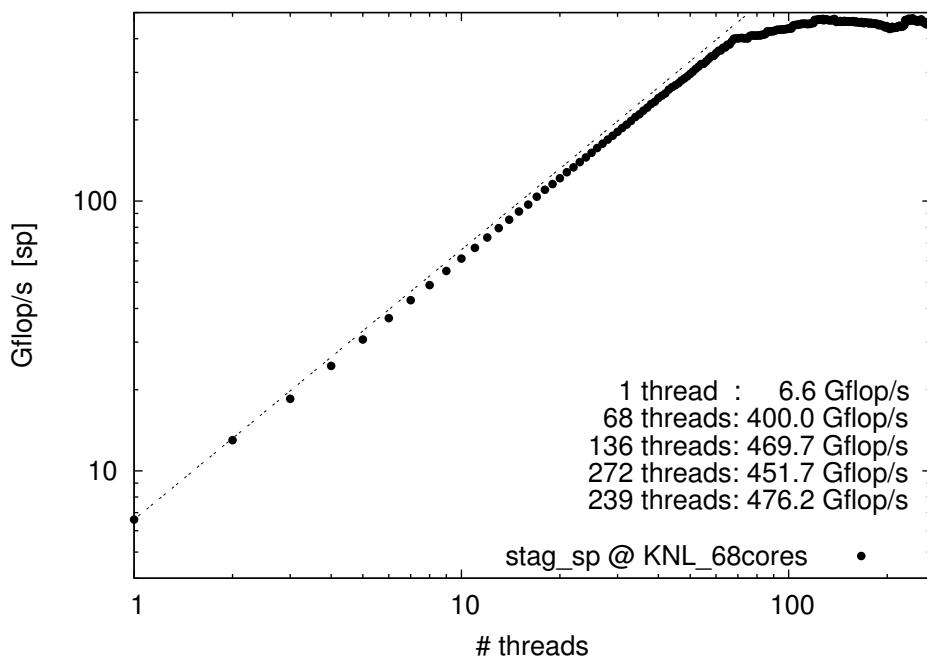
Replace $i_plu=\dots$ and siblings by 1D lookup-tables (also for Wilson and Brillouin).

Staggered operator timings

- **Staggered on KNL (68 cores)**

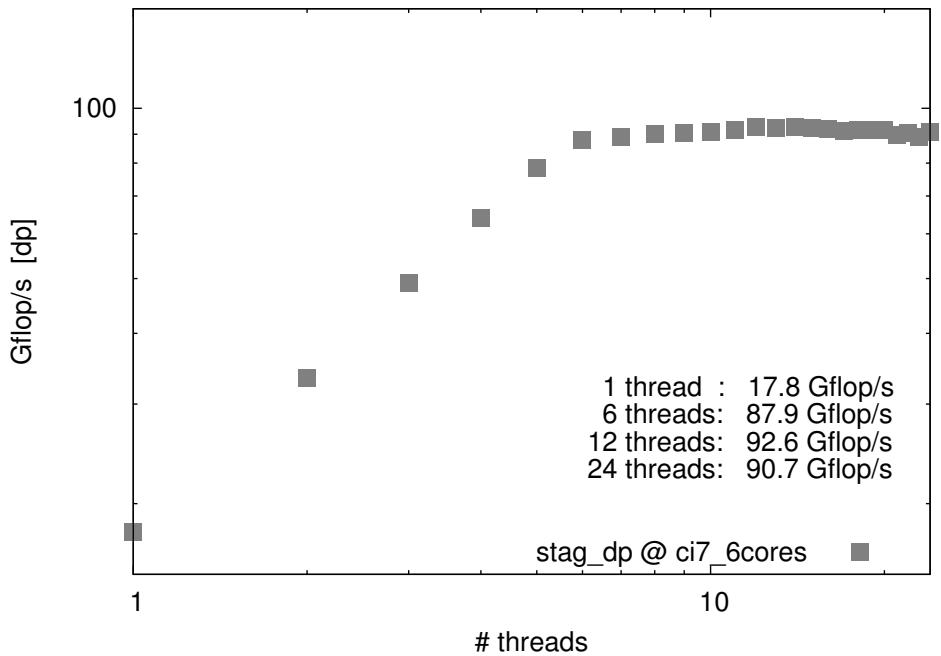
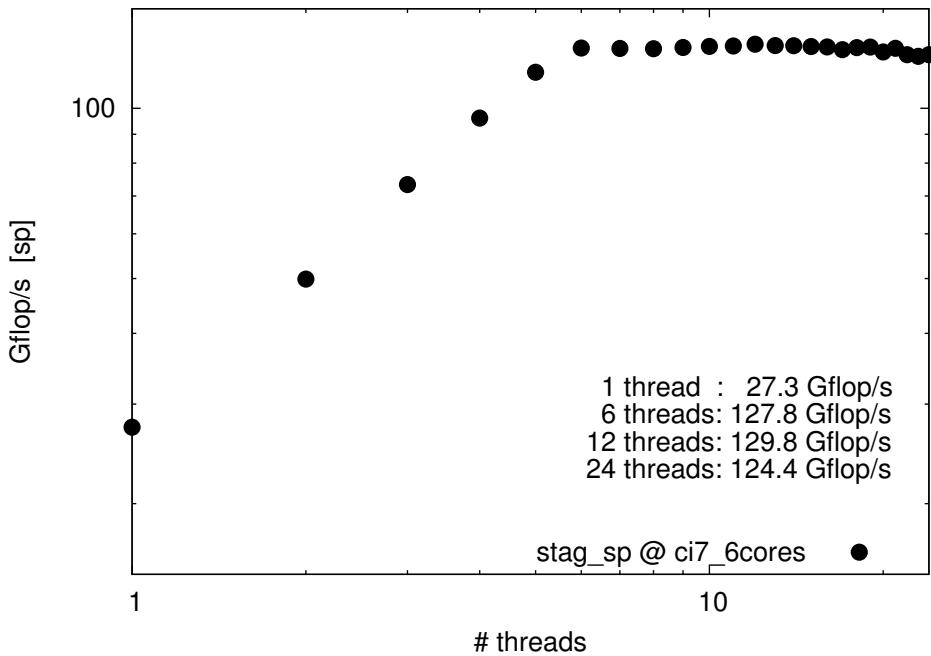
Volume $34^3 \cdot 68$ fixed, $N_c = 3$ fixed, $N_v = 4N_c$ fixed, all performances in Gflop/s:

	$N_{\text{thr}} = 1$	$N_{\text{thr}} = 68$	$N_{\text{thr}} = 136$	$N_{\text{thr}} = 272$	optimum
sp	6.6	400.0	469.7	451.7	476.2@239
dp	4.9	295.3	335.1	309.0	342.7@153



- **Staggered on BDW (6 cores, same parameters)**

	$N_{\text{thr}} = 1$	$N_{\text{thr}} = 2$	$N_{\text{thr}} = 6$	$N_{\text{thr}} = 12$	$N_{\text{thr}} = 24$	optimum
sp	27.3	49.9	127.8	129.8	124.4	129.8@12
dp	17.8	33.4	87.9	92.6	90.7	92.6@12



- **Staggered performance ratios**

KNL: 476/343 Gflop/s out of 6.1/3.05 Tflop/s mean 7.9 / 11.4% of peak [sp/dp]
 BdW: 130 / 93 Gflop/s out of 690/345 Gflop/s mean 18.8/26.9% of peak [sp/dp]

Wilson kernel details

PARAMETERS: Nx,Ny,Nz,Nt, Nc,Nv, sp,dp, i_sp,i_dp !!! known at compile time

```
!$OMP PARALLEL DO COLLAPSE(2) DEFAULT(private) FIRSTPRIVATE(mass) SHARED(old,new,V) SCHEDULE(dynamic)
do l=1,Nt
do k=1,Nz; k_plu=modulo(k,Nz)+1; k_min=modulo(k-2,Nz)+1; l_plu=modulo(l,Nt)+1; l_min=modulo(l-2,Nt)+1
do j=1,Ny; j_plu=modulo(j,Ny)+1; j_min=modulo(j-2,Ny)+1
do i=1,Nx; i_plu=modulo(i,Nx)+1; i_min=modulo(i-2,Nx)+1
n=sum(([i,j,k,l]-[0,1,1,1])*[1,Nx,Nx*Ny,Nx*Ny*Nz]) !!! lexicographic index
site(:,:,:)=cmplx(0.0,kind=sp) !!! accumulation variable
!!! direction -4 gets factor 1/2*(-I-gamma4)
tmp=0.5*transpose(conjg(V(:,:,:,4,i,j,k,l_min))); nsh=n+(l_min-1)*Nz*Ny*Nx; ...
!$OMP SIMD PRIVATE(full)
do idx=1,Nv
  forall(col=1:Nc,spi=1:4) full(col,spi)=sum(tmp(col,:)*old(:,spi,idx,nsh))
  site(:,1,idx)=site(:,1,idx)-full(:,1)-      full(:,3) !!! transpose(gamma4)=      0      0      1      0
  site(:,2,idx)=site(:,2,idx)-full(:,2)-      full(:,4) !!!                      0      0      0      1
  site(:,3,idx)=site(:,3,idx)-full(:,3)-      full(:,1) !!!                      1      0      0      0
  site(:,4,idx)=site(:,4,idx)-full(:,4)-      full(:,2) !!!                      0      1      0      0
end do ! idx=1,Nv
!$OMP END SIMD
...
!!! add mass term and plug everything into new vector
new(:,:,:,:n)=site(:,:,:,:)+(4.0+mass)*old(:,:,:,:n)
end do ! i=1,Nx
end do ! j=1,Ny
end do ! k=1,Nz
end do ! l=1,Nt
!$OMP END PARALLEL DO
```

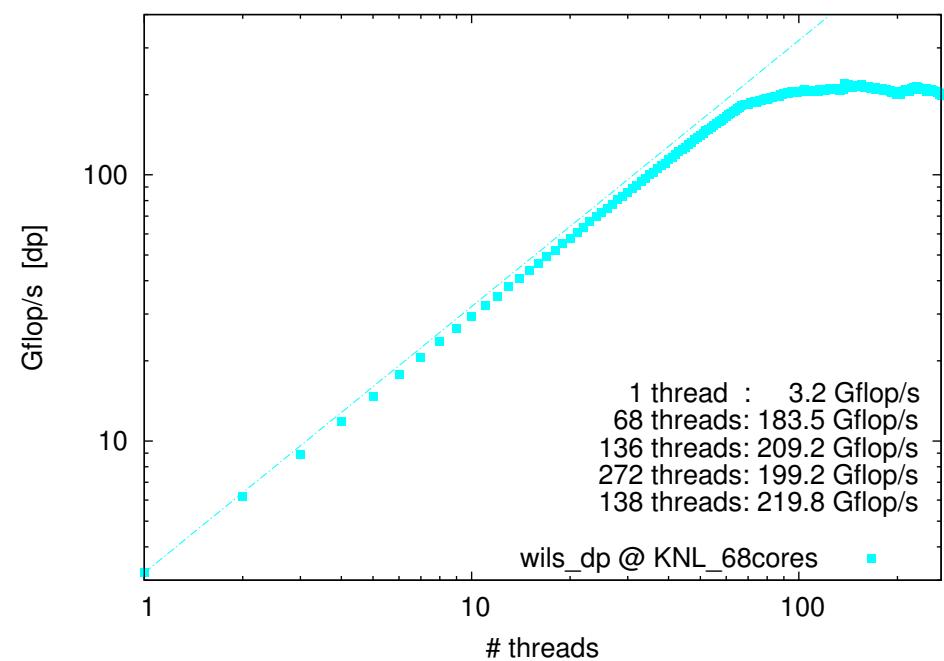
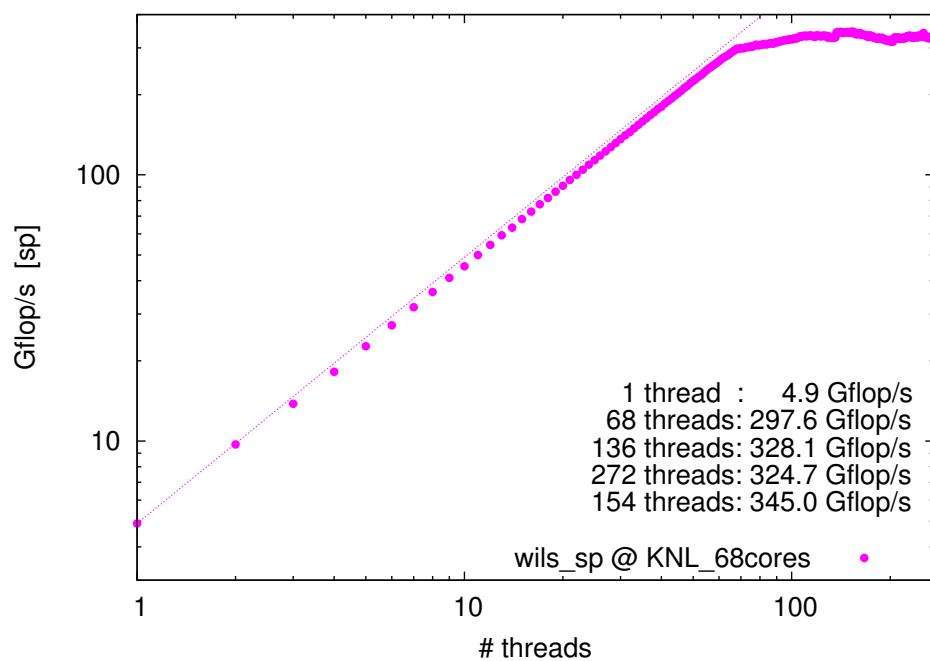
- OMP-pragmas for shared-memory parallelization over space-time indices
- OMP-pragmas for SIMD-pipelining over Nv right-hand-sides
- Employ shrink-expand trick for $\text{old}(:,:,\text{idx},\text{nsh}) \rightarrow \text{gauge} * \text{old} * \text{trsp}(\gamma)$

Wilson operator timings

- **Wilson on KNL (68 cores)**

Volume $34^3 \cdot 68$ fixed, $N_c = 3$ fixed, $N_v = 4N_c$ fixed, all performances in Gflop/s:

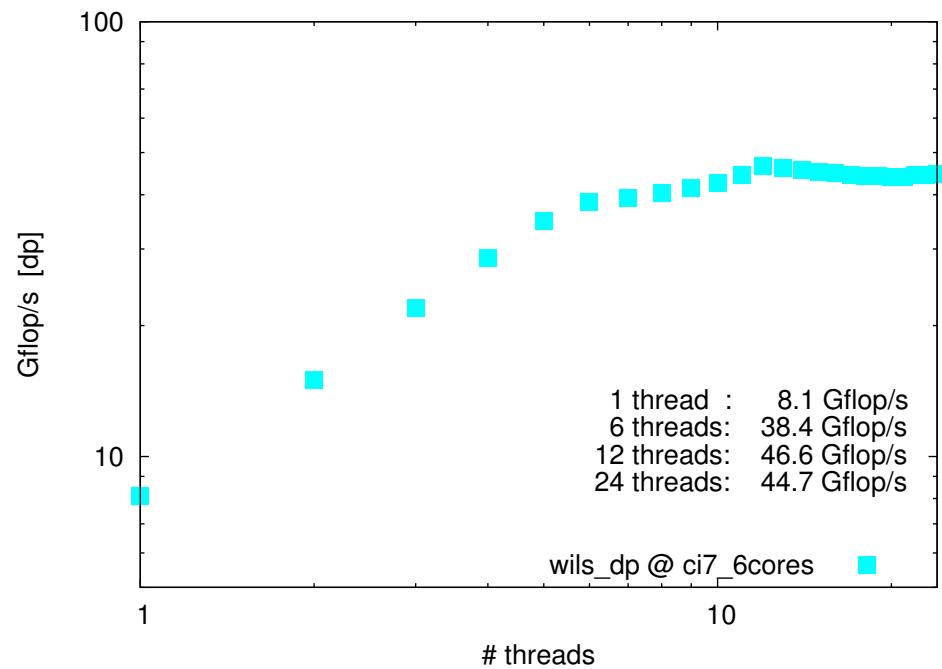
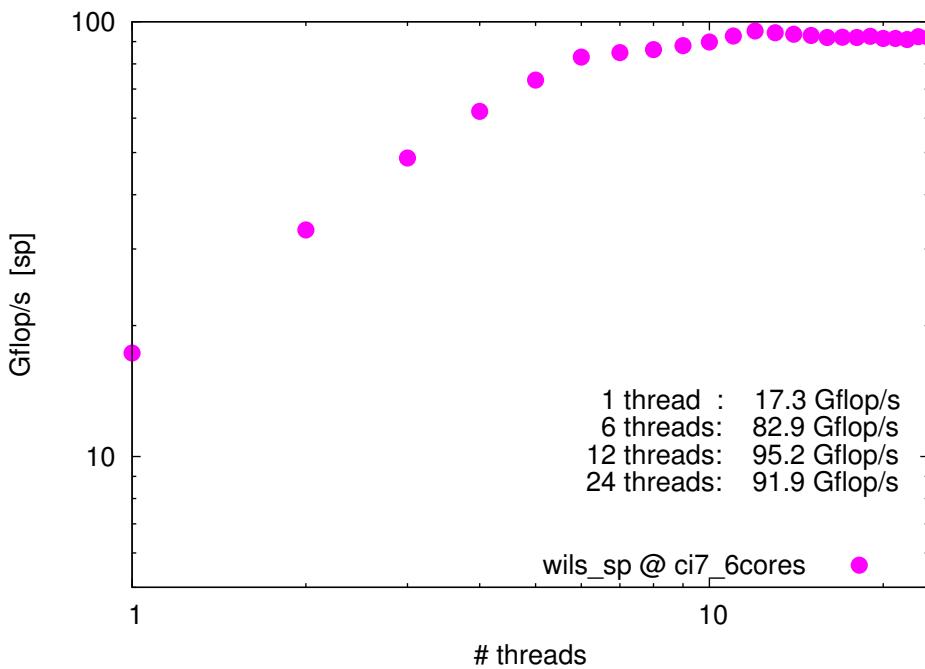
	$N_{\text{thr}} = 1$	$N_{\text{thr}} = 68$	$N_{\text{thr}} = 136$	$N_{\text{thr}} = 272$	optimum
sp	4.9	297.6	328.1	324.7	345.0@154
dp	3.2	183.5	209.2	199.2	219.8@138



Only layout subset $N_c N_s N_v$, $N_v N_c N_s$, $N_v N_s N_c$ (for vec) times ditto (for site) explored.

- Wilson on BDW (6 cores, same parameters)

	$N_{\text{thr}} = 1$	$N_{\text{thr}} = 2$	$N_{\text{thr}} = 6$	$N_{\text{thr}} = 12$	$N_{\text{thr}} = 24$	optimum
sp	17.3	33.2	82.9	95.2	91.9	95.2@12
dp	8.1	15.0	38.4	46.6	44.7	46.6@12



- Wilson performance ratios

KNL: 345/220 Gflop/s out of 6.1/3.05 Tflop/s mean 5.7 / 7.3% of peak [sp/dp]
 BdW: 95 / 47 Gflop/s out of 690/345 Gflop/s mean 13.8/13.6% of peak [sp/dp]

Brillouin operator definition

- **Standard versus Brillouin Laplacian**

$$a^2 \Delta^{\text{std}}(x, y) = -8 \delta_{x,y} + 1 \sum_{\mu} V_{\mu}(x) \delta_{x+\hat{\mu},y}$$

Brillouin Laplacian defined with $(\lambda_0, \lambda_1, \lambda_2, \lambda_3, \lambda_4) \equiv (-240, 8, 4, 2, 1)/64$:

$$\begin{aligned} a^2 \Delta^{\text{bri}}(x, y) = & \lambda_0 \delta_{x,y} + \lambda_1 \sum_{\mu} W_{\mu}(x) \delta_{x+\hat{\mu},y} \\ & + \lambda_2 \sum_{\neq(\mu,\nu)} W_{\mu+\nu}(x) \delta_{x+\hat{\mu}+\hat{\nu},y} \\ & + \lambda_3 \sum_{\neq(\mu,\nu,\rho)} W_{\mu+\nu+\rho}(x) \delta_{x+\hat{\mu}+\hat{\nu}+\hat{\rho},y} \\ & + \lambda_4 \sum_{\neq(\mu,\nu,\rho,\sigma)} W_{\mu+\nu+\rho+\sigma}(x) \delta_{x+\hat{\mu}+\hat{\nu}+\hat{\rho}+\hat{\sigma},y} \end{aligned}$$

1-hop: 8 terms, each 1! paths $W_{\mu}(x) = V_{\mu}(x)$ (smeared link, $\mu \in \{\pm 1, \pm 2, \pm 3, \pm 4\}$)

2-hop: 24 terms, each 2! paths $W_{\mu+\nu}(x) = \frac{1}{2}[V_{\mu}(x)V_{\nu}(x+\hat{\mu}) + \text{perm}]$

3-hop: 32 terms, each 3! paths $W_{\mu+\nu+\rho}(x) = \frac{1}{6}[V_{\mu}(x)V_{\nu}(x+\hat{\mu})V_{\rho}(x+\hat{\mu}+\hat{\nu}) + \text{perm}]$

4-hop: 16 terms, each 4! paths $W_{\mu+\nu+\rho+\sigma}(x) = \frac{1}{24}[V_{\mu}V_{\nu}V_{\rho}V_{\sigma} + \text{perm}]$

- **Standard versus Isotropic Derivative**

$$a\nabla_\mu^{\text{std}}(x, y) = 1 [V_\mu(x)\delta_{x+\hat{\mu}, y} - V_{-\mu}(x)\delta_{x-\hat{\mu}, y}]$$

Isotropic Derivative defined with $(\rho_1, \rho_2, \rho_3, \rho_4) \equiv (64, 16, 4, 1)/432$:

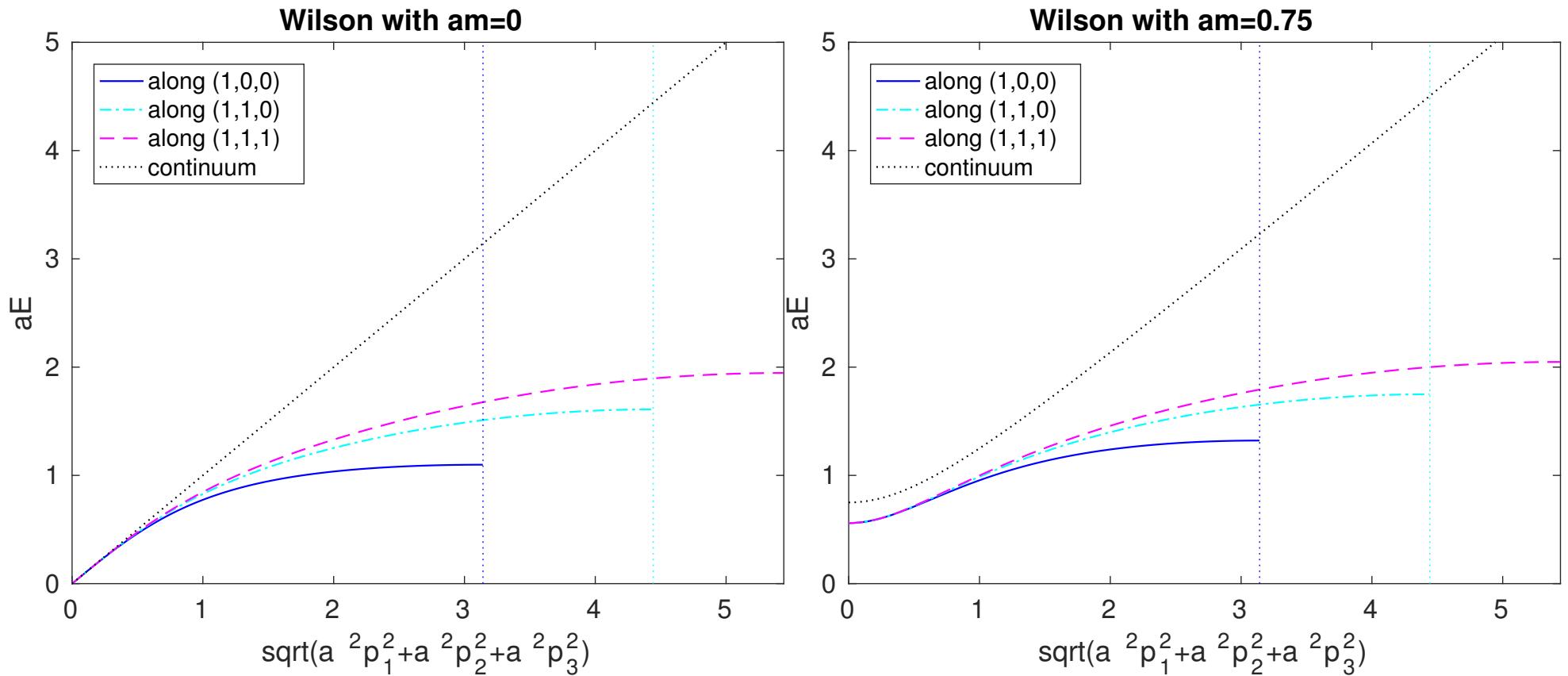
$$\begin{aligned} a\nabla_\mu^{\text{iso}}(x, y) &= \rho_1 [W_\mu(x)\delta_{x+\hat{\mu}, y} - W_{-\mu}(x)\delta_{x-\hat{\mu}, y}] \\ &+ \rho_2 \sum_{\neq(\nu; \mu)} [W_{\mu+\nu}(x)\delta_{x+\hat{\mu}+\hat{\nu}, y} - W_{-\mu+\nu}(x)\delta_{x-\hat{\mu}+\hat{\nu}, y}] \\ &+ \rho_3 \sum_{\neq(\nu, \rho; \mu)} [W_{\mu+\nu+\rho}(x)\delta_{x+\hat{\mu}+\hat{\nu}+\hat{\rho}, y} - (\mu \rightarrow -\mu)] \\ &+ \rho_4 \sum_{\neq(\nu, \rho, \sigma; \mu)} [W_{\mu+\nu+\rho}(x)\delta_{x+\hat{\mu}+\hat{\nu}+\hat{\rho}+\hat{\sigma}, y} - (\mu \rightarrow -\mu)] \end{aligned}$$

Comment: paths must be averaged in order to maintain γ_5 -hermiticity

Comment: weights $\{\lambda_i\}$ and $\{\rho_j\}$ fixed via free-field consideration (no tuning)

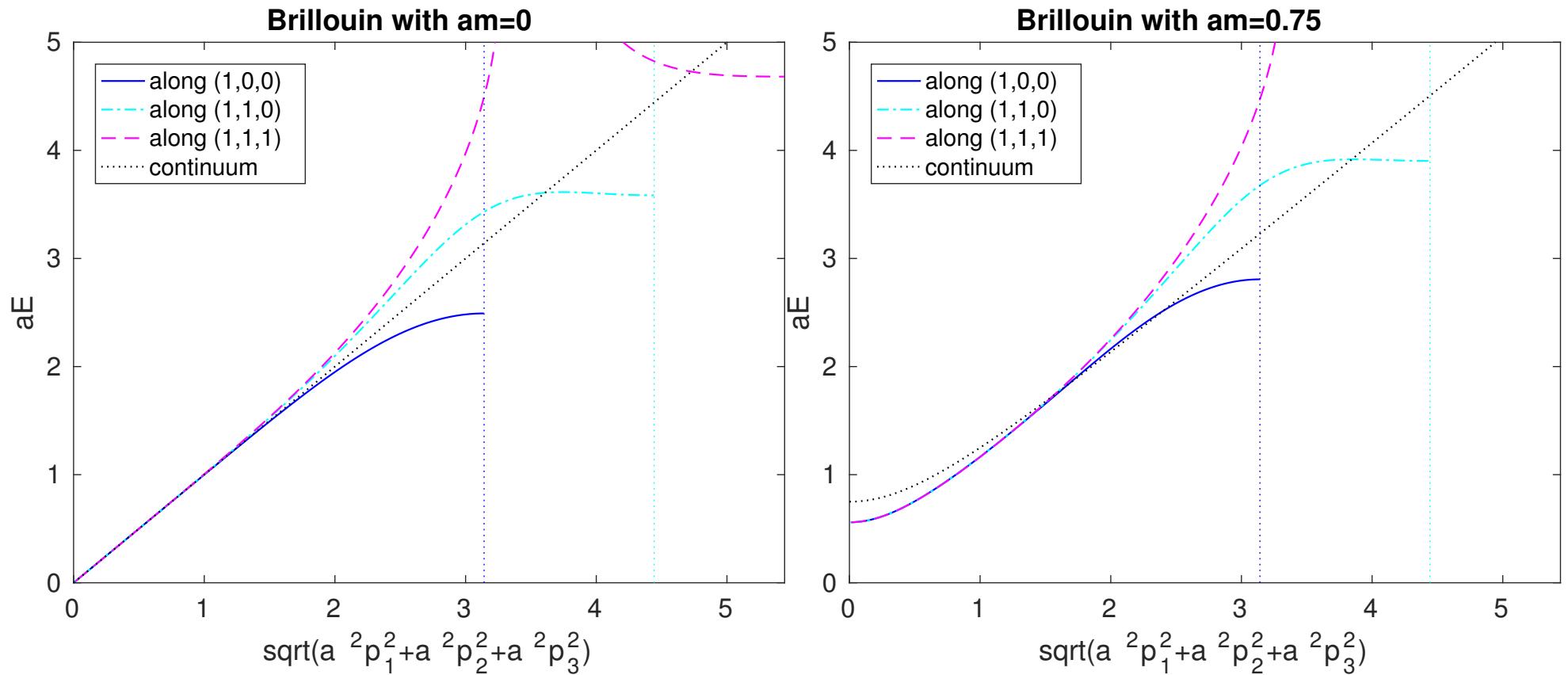
Comment: currently \mathbf{W} precomputed (40 directions, hence 10-fold memory of \mathbf{V})

- Dispersion relation for Wilson operator



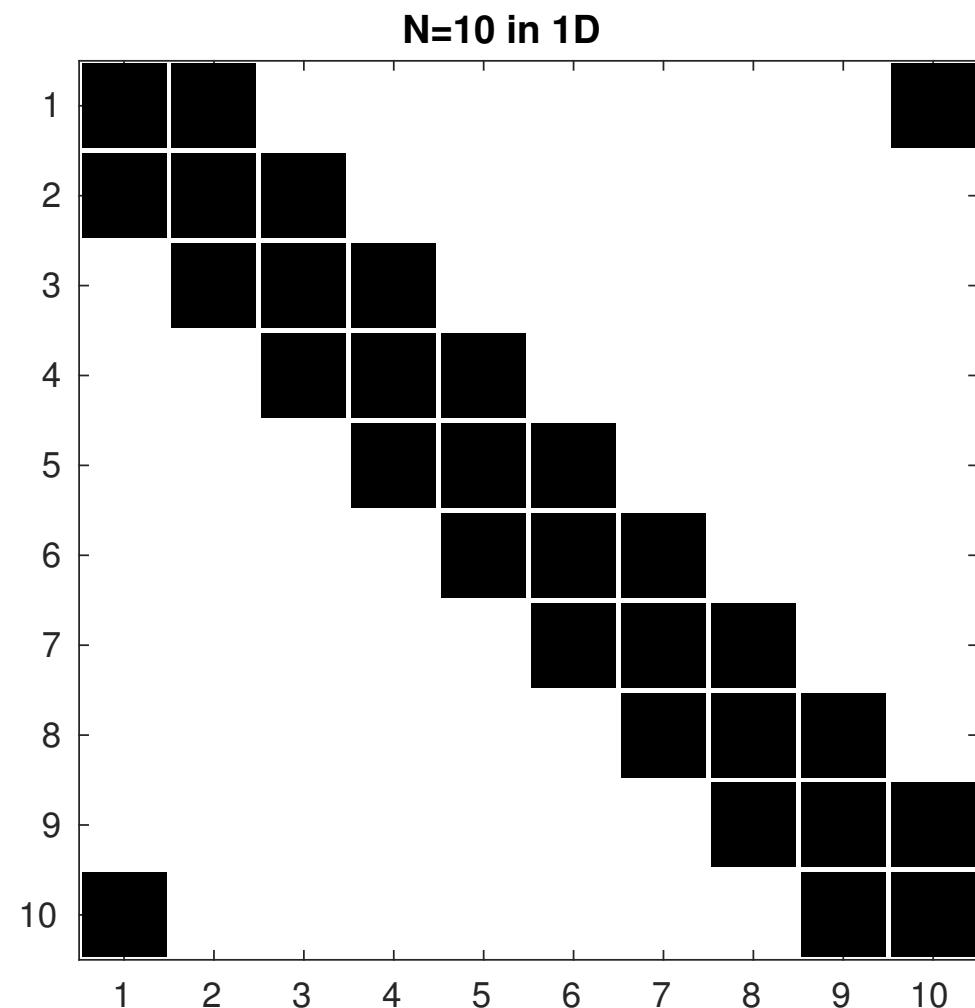
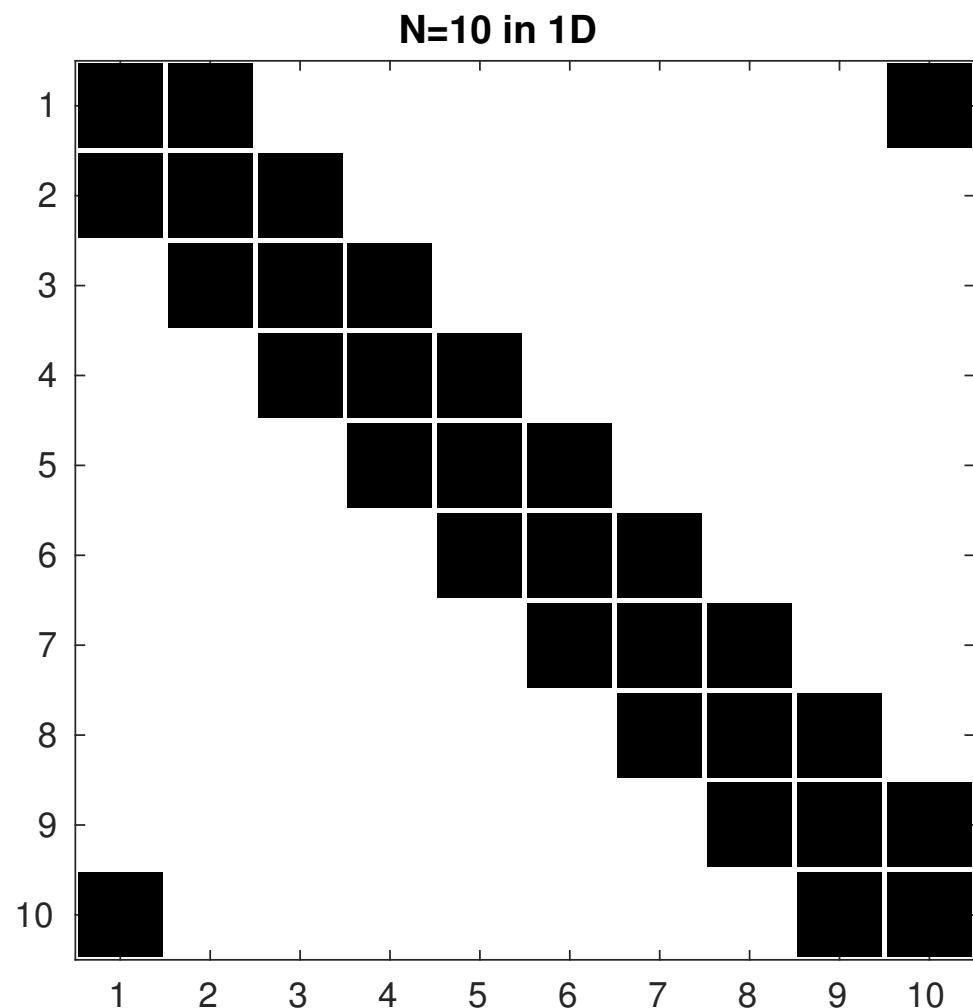
- ⊖ strong deviation from continuum for any $a|\mathbf{p}| > 1$
- ⊖ strong rotational symmetry breaking for any $a|\mathbf{p}| > 1$
- ⊖ strong effect of $am \ll 1$, even at $\mathbf{p} = 0$

- Dispersion relation for Brillouin operator



- ⊕ mild deviation from continuum up to $a|\mathbf{p}| \simeq 2$
- ⊕ mild rotational symmetry breaking up to $a|\mathbf{p}| \simeq 2$
- ⊖ strong effect of $am \ll 1$, especially at $\mathbf{p} = 0$

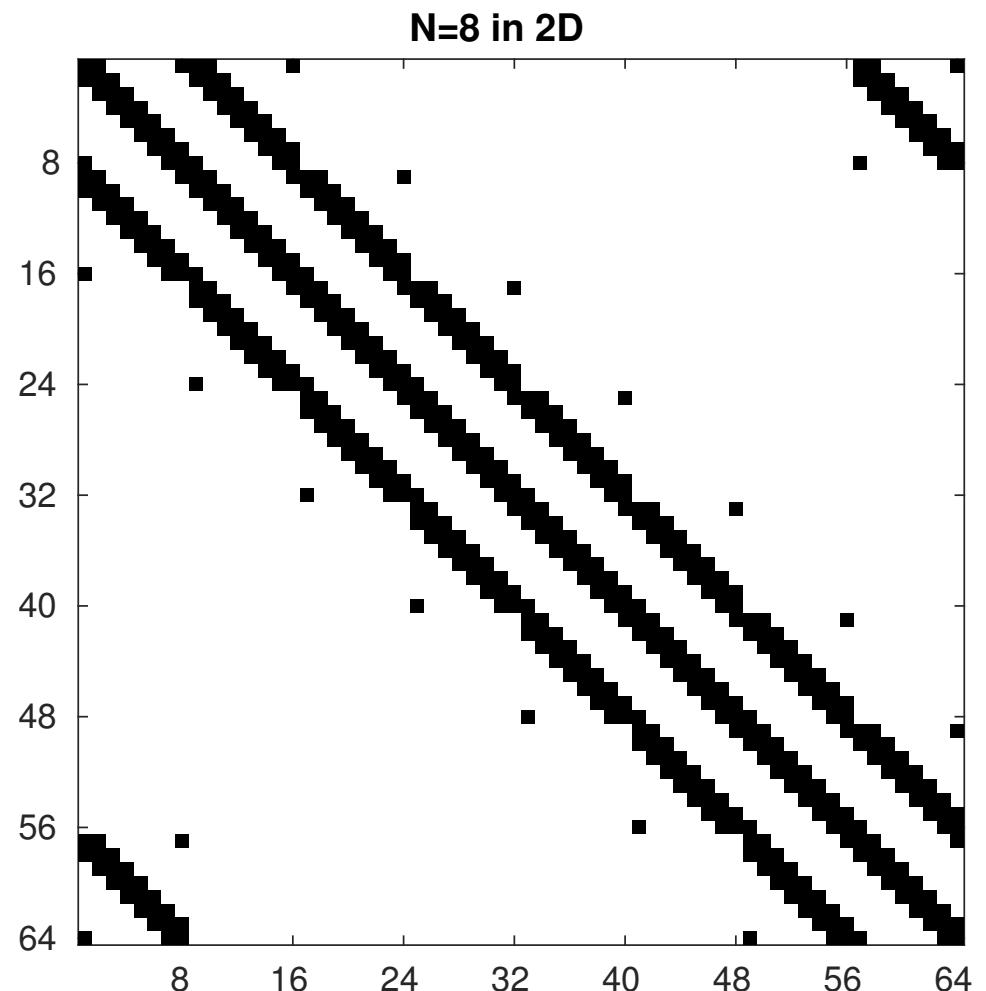
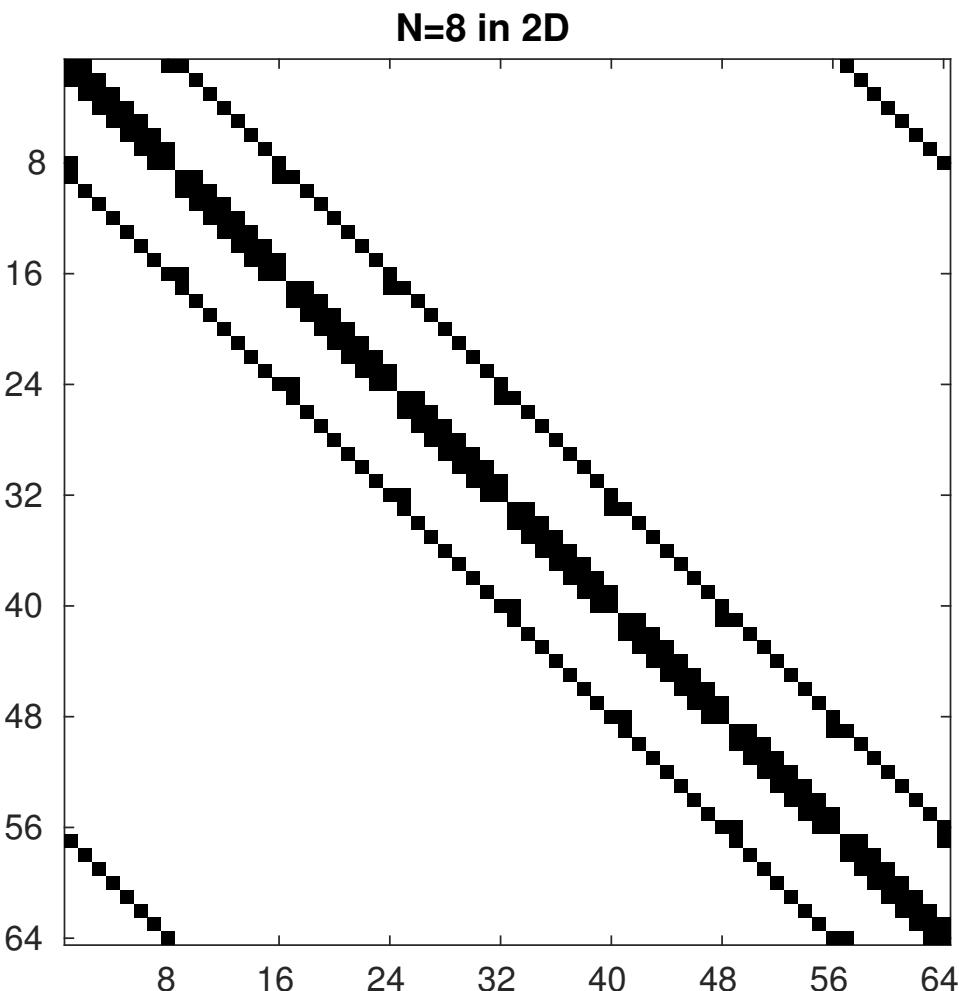
- **Visualization of PBC in 1D for Wilson and Brillouin operators**



Fixed number of non-zero entries per row/col in 1D is 3 (3) for Wilson (Brillouin).

Note: Further factors ... \otimes spinor \otimes color omitted for simplicity.

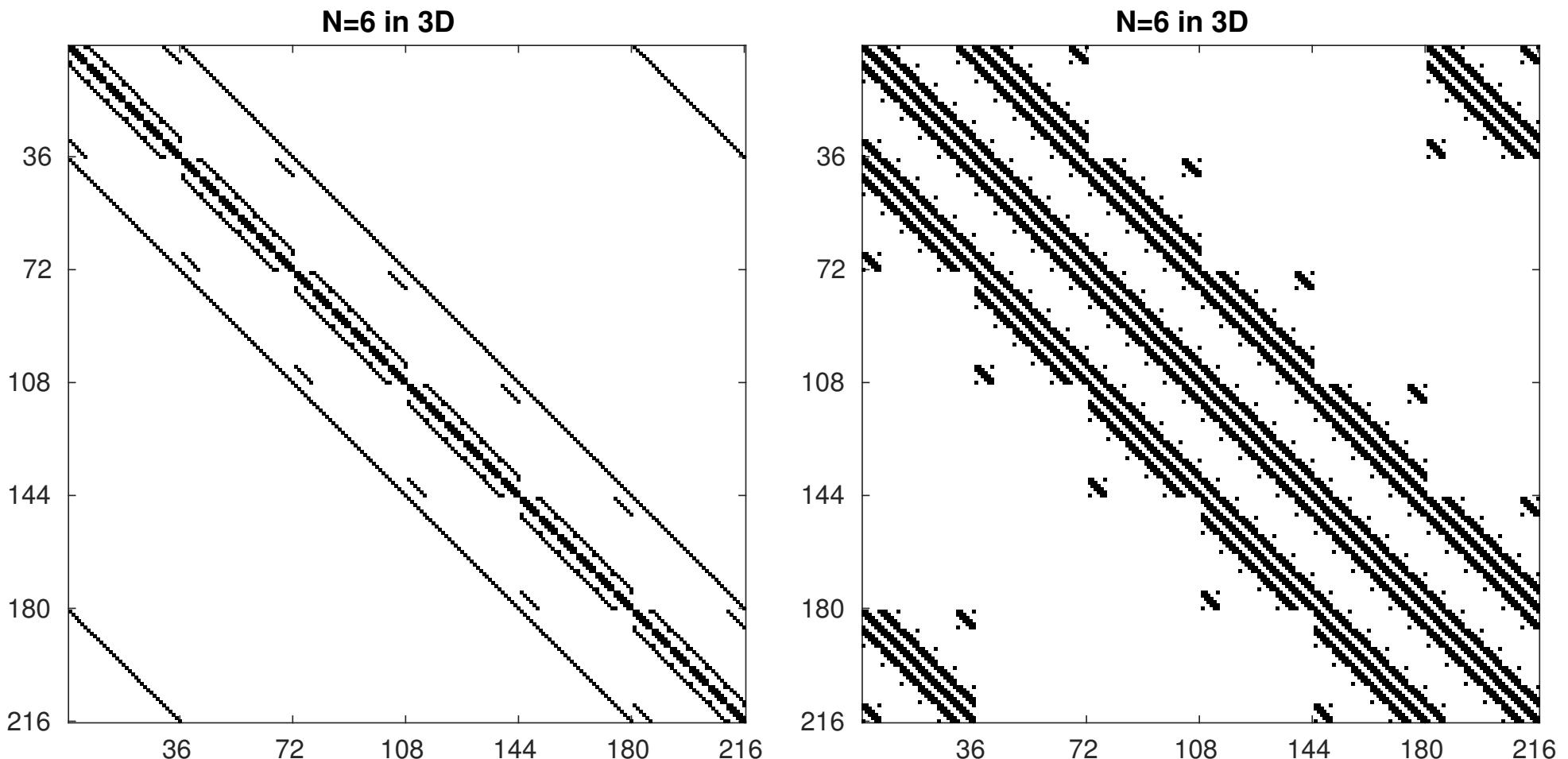
- **Visualization of PBC in 2D for Wilson and Brillouin operators**



Fixed number of non-zero entries per row/col in 2D is 5 (9) for Wilson (Brillouin).

Note: Further factors ... \otimes spinor \otimes color omitted for simplicity.

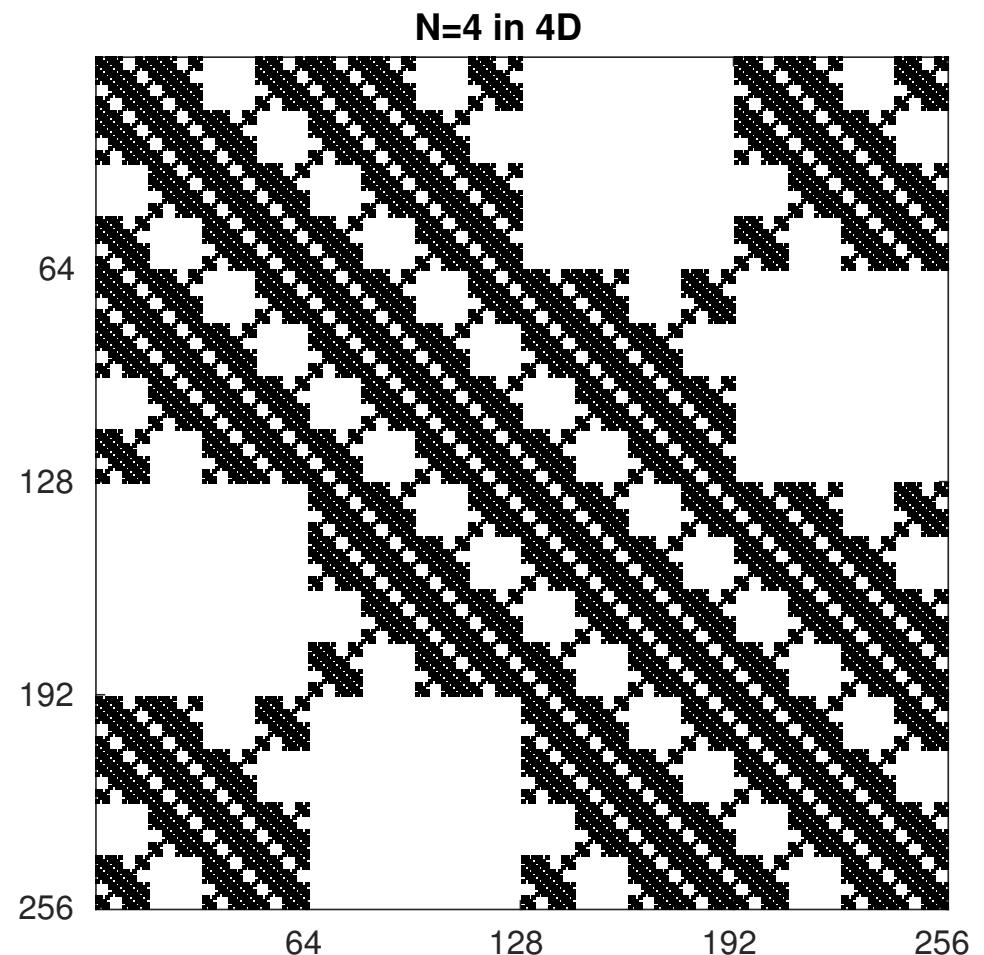
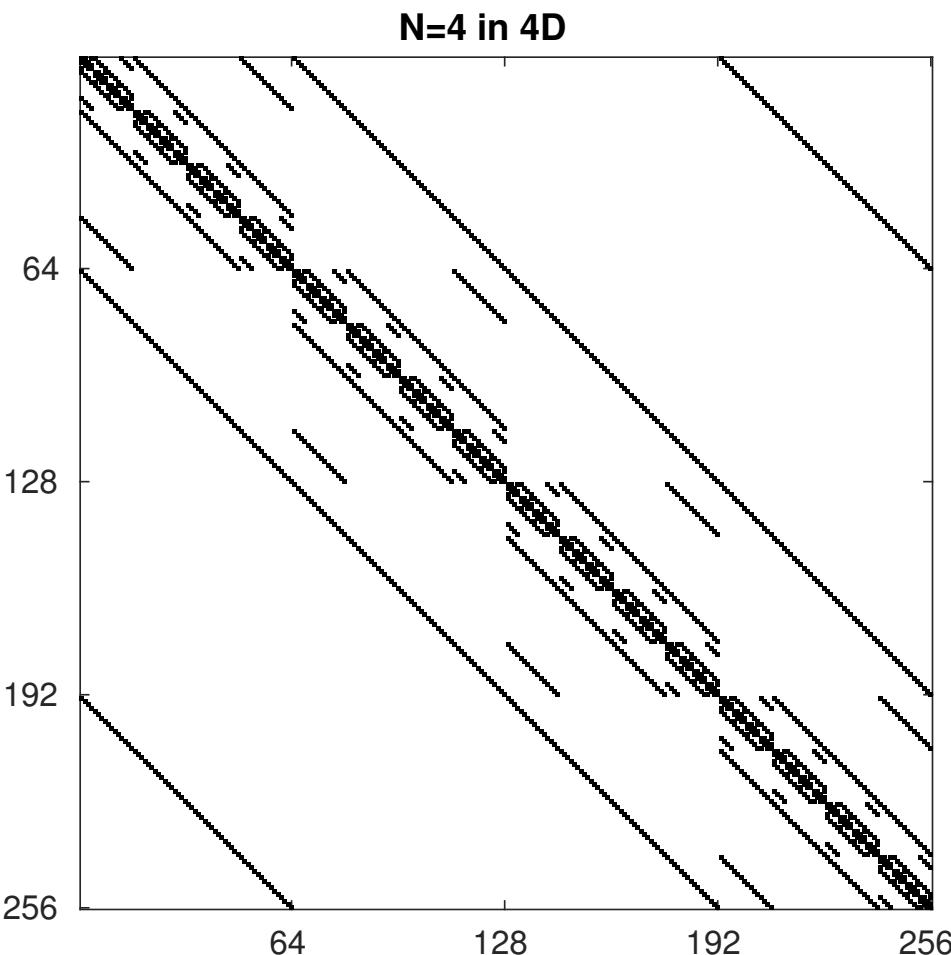
- **Visualization of PBC in 3D for Wilson and Brillouin operators**



Fixed number of non-zero entries per row/col in 3D is 7 (27) for Wilson (Brillouin).

Note: Further factors ... \otimes spinor \otimes color omitted for simplicity.

- **Visualization of PBC in 4D for Wilson and Brillouin operators**



Fixed number of non-zero entries per row/col in 4D is 9 (81) for Wilson (Brillouin).

Note: Further factors ... \otimes spinor \otimes color omitted for simplicity.

Brillouin kernel details

PARAMETERS: Nx,Ny,Nz,Nt, Nc,Nv, sp,dp, i_sp,i_dp !!! known at compile time

```
mask_der=[0.0_sp,64.0_sp,16.0_sp,4.0_sp,1.0_sp]/432.0_sp !!! note: declared as real(kind=sp),dimension(0:4)
!$OMP PARALLEL DO COLLAPSE(3) DEFAULT(private) FIRSTPRIVATE(mask_der) SHARED(old,new,W) SCHEDULE(dynamic)
do l=1,Nt
do k=1,Nz
do j=1,Ny
do i=1,Nx
n=sum(([i,j,k,l]-[0,1,1,1])*[1,Nx,Nx*Ny,Nx*Ny*Nz]) !!! lexicographic index
site(:,:,:)=cmplx(0.0,kind=sp) !!! accumulation variable
!!! visit all 81 sites within hypercube (distances 0 to 4 in taxi-driver metric)
dir=0
do go_l=-1,1; lsh=modulo(l+go_l-1,Nt)+1
do go_k=-1,1; ksh=modulo(k+go_k-1,Nz)+1
do go_j=-1,1; jsh=modulo(j+go_j-1,Ny)+1
do go_i=-1,1; ish=modulo(i+go_i-1,Nx)+1
dir=dir+1 !!! note: dir=(go_l+1)*27+(go_k+1)*9+(go_j+1)*3+go_i+2
...
end do ! go_i=-1,1
end do ! go_j=-1,1
end do ! go_k=-1,1
end do ! go_l=-1,1
!!! plug everything into new vector
do idx=1,Nv
new(:,:,:idx)=site(:,:,:idx)
end do ! idx=1,Nv
end do ! i=1,Nx
end do ! j=1,Ny
end do ! k=1,Nz
end do ! l=1,Nt
!$OMP END PARALLEL DO
```

The block ... is replaced by (chiral representation of γ -matrices, simplest version):

```
select case(dir)
  case(01:40); tmp=W(:,:,:dir,i,j,k,l)
  case( 41); tmp=color_eye() !!! note: yields Nc*Nc identity matrix
  case(42:81); tmp=conjg(transpose(W(:,:,:82-dir,ish,jsh,ksh,lsh)))
end select
absgo_ijkl=sum([go_i,go_j,go_k,go_l]**2) !!! note: absgo_ijkl=go_i**2+go_j**2+go_k**2+go_l**2
lap=0.125_sp/2**absgo_ijkl           !!! note: factor for 1/2 times Brillouin Laplacian
if (absgo_ijkl.eq.0) lap=lap-2.0_sp-mass !!! note: correction for go_i=go_j=go_k=go_l=0
der_i=go_i*mask_der(absgo_ijkl)       !!! note: factor for isotropic derivative in x-direction
der_j=go_j*mask_der(absgo_ijkl)       !!! note: factor for isotropic derivative in y-direction
der_k=go_k*mask_der(absgo_ijkl)       !!! note: factor for isotropic derivative in z-direction
der_l=go_l*mask_der(absgo_ijkl)       !!! note: factor for isotropic derivative in t-direction
nsh=sum(([lsh,ksh,jsh,ish]-[1,1,1,0])*[Nx*Ny*Nz,Nx*Ny,Nx,1])
 !$OMP SIMD PRIVATE(full)
do idx=1,Nv
  forall(col=1:Nc,spi=1:4) full(col,spi)=sum(tmp(:,col,:)*old(:,spi,idx,nsh))
  site(:,1,idx)=site(:,1,idx)-lap*full(:,1)+cmplx(-der_j,-der_i)*full(:,4)+cmplx(+der_l,-der_k)*full(:,3)
  site(:,2,idx)=site(:,2,idx)-lap*full(:,2)+cmplx(+der_j,-der_i)*full(:,3)+cmplx(+der_l,+der_k)*full(:,4)
  site(:,3,idx)=site(:,3,idx)-lap*full(:,3)+cmplx(+der_j,+der_i)*full(:,2)+cmplx(+der_l,+der_k)*full(:,1)
  site(:,4,idx)=site(:,4,idx)-lap*full(:,4)+cmplx(-der_j,+der_i)*full(:,1)+cmplx(+der_l,-der_k)*full(:,2)
end do ! idx=1,Nv
 !$OMP END SIMD
```

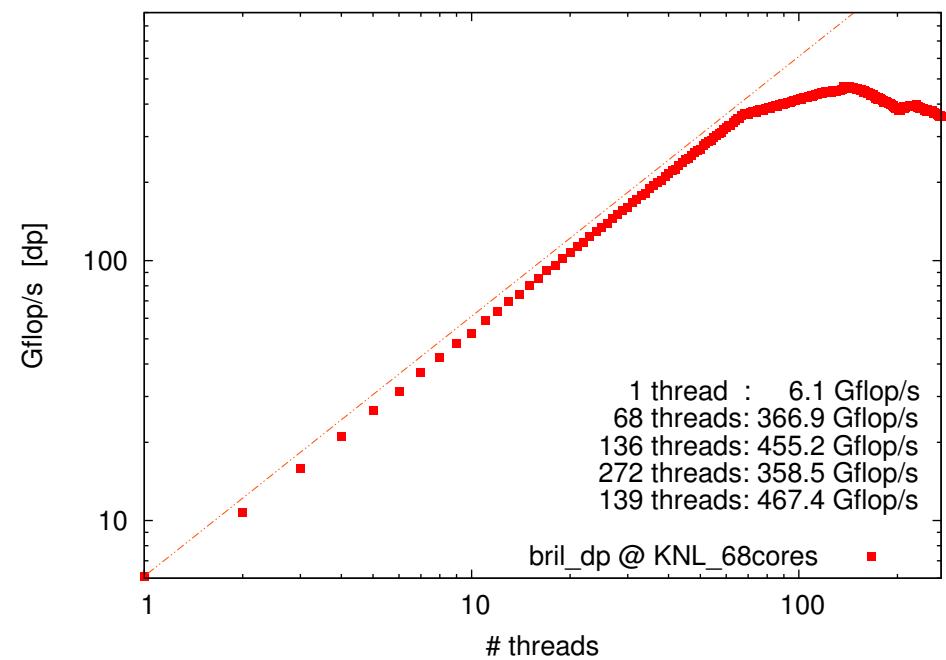
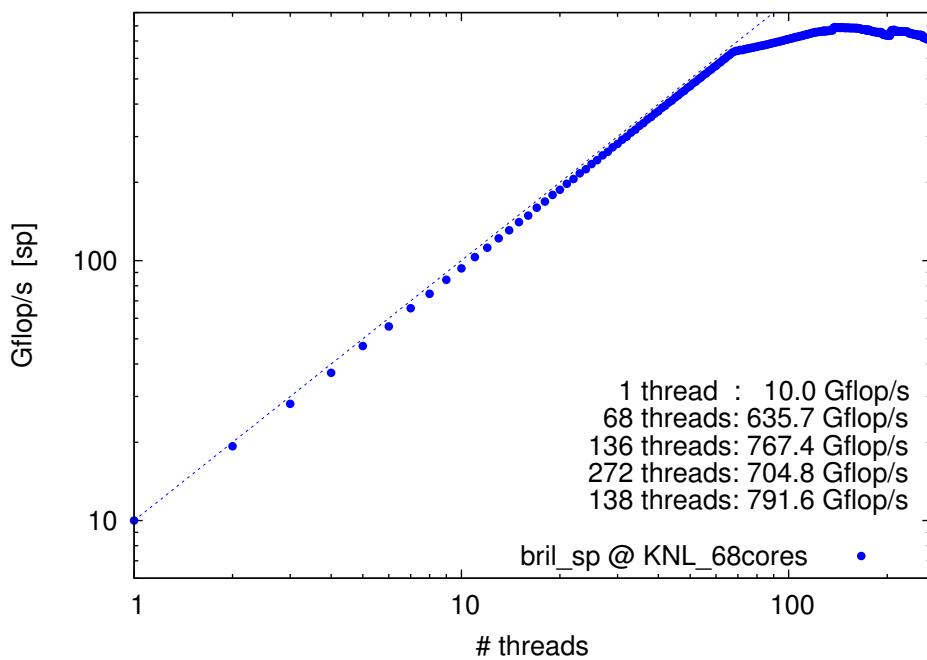
- OMP-pragmas for shared-memory parallelization over space-time indices
- OMP-pragmas for SIMD-pipelining over Nv right-hand-sides
- color-spinor multiplication in `forall(col=1:Nc,spi=1:4)` means unrolling
- implied-do in `site(:,spi,idx)=...` means unrolling

Brillouin operator timings

- **Brillouin on KNL (68 cores)**

Volume $34^3 \cdot 68$ fixed, $N_c = 3$ fixed, $N_v = 4N_c$ fixed, NvNsNc-performances in Gflop/s:

	$N_{\text{thr}} = 1$	$N_{\text{thr}} = 68$	$N_{\text{thr}} = 136$	$N_{\text{thr}} = 272$	optimum
sp	10.0	635.7	767.4	704.8	791.6@138
dp	6.1	366.9	455.2	358.5	467.4@139



Only layout subset NcNsNv, NvNcNs, NvNsNc (for vec) times ditto (for site) explored.

- **Dependence on $N_x \cdot N_y \cdot N_z \cdot N_t$ (on KNL_68)**

$N_c = 3$ fixed, $N_v = 4N_c$ fixed, $T = 2L$ scales, with sp-performance in Gflop/s:

	$L = 16$	$L = 20$	$L = 24$	$L = 32$	$L = 40$	$L = 48$
136 threads	756.9	756.4	770.2	670.7	756.6	738.1
272 threads	734.5	738.2	759.4	620.9	602.3	522.7

- **Dependence on N_v (on KNL_68)**

Volume $24^3 \cdot 48$ fixed, $N_c = 3$ fixed, with sp-performance in Gflop/s:

	$N_v = 2N_c$	$N_v = 4N_c$	$N_v = 8N_c$	$N_v = 16N_c$	$N_v = 32N_c$
136 threads	511.0	770.5	1013.9	652.4	593.3
272 threads	647.5	755.5	682.2	613.3	557.3

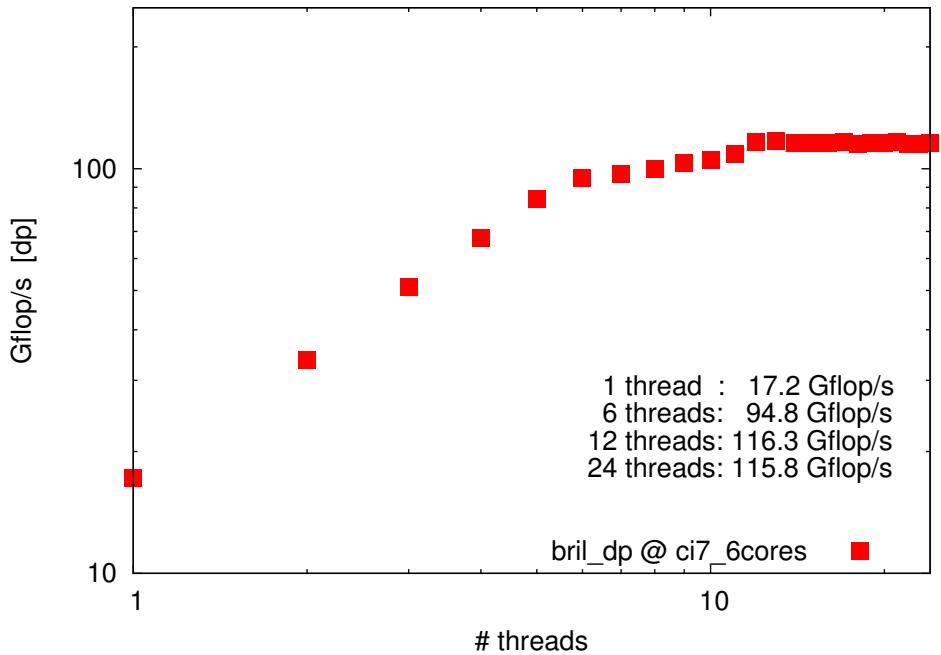
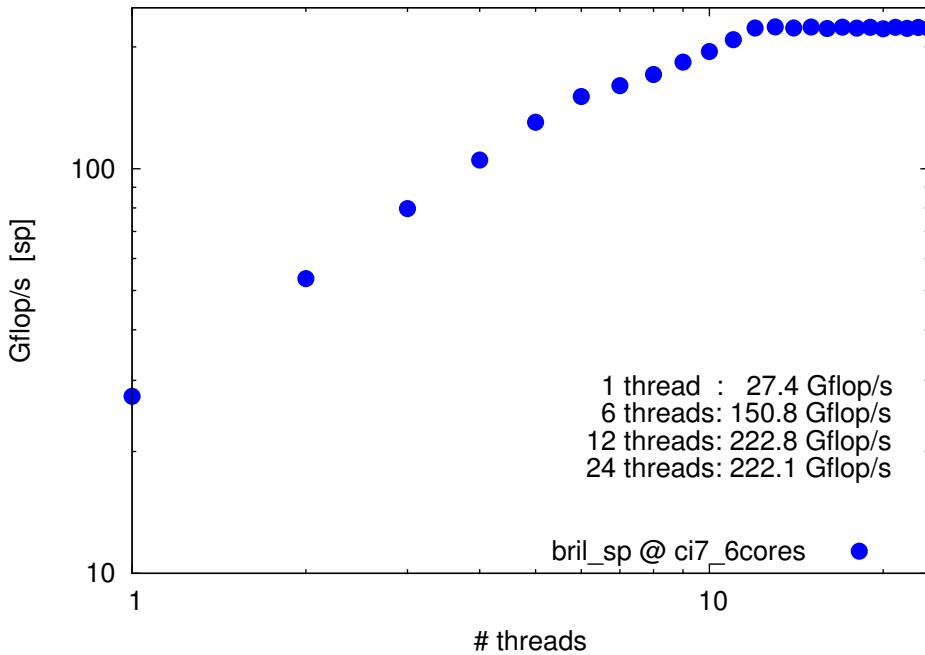
- **Dependence on N_c (on KNL_68)**

Volume $24^3 \cdot 48$ fixed, $N_v = 12$ fixed, with sp-performance in Gflop/s:

	$N_c = 2$	$N_c = 3$	$N_c = 4$	$N_c = 5$	$N_c = 6$	$N_c = 7$	$N_c = 8$
136 threads	602.0	770.4	782.0	865.0	855.8	828.4	844.9
272 threads	659.0	763.2	751.0	773.2	852.7	921.2	937.0

- Brillouin on BDW (6 cores, original parameters again)

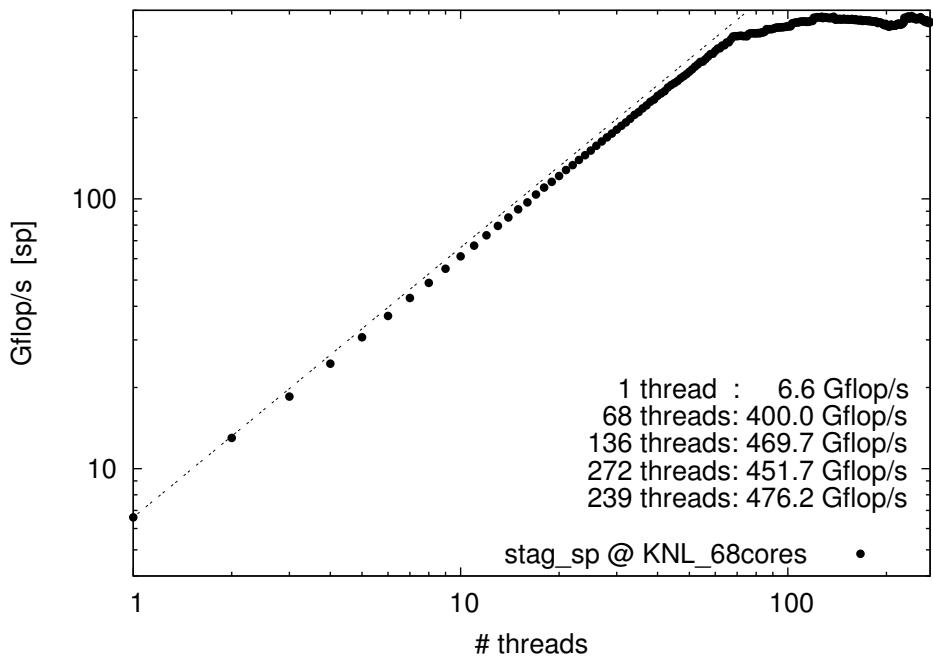
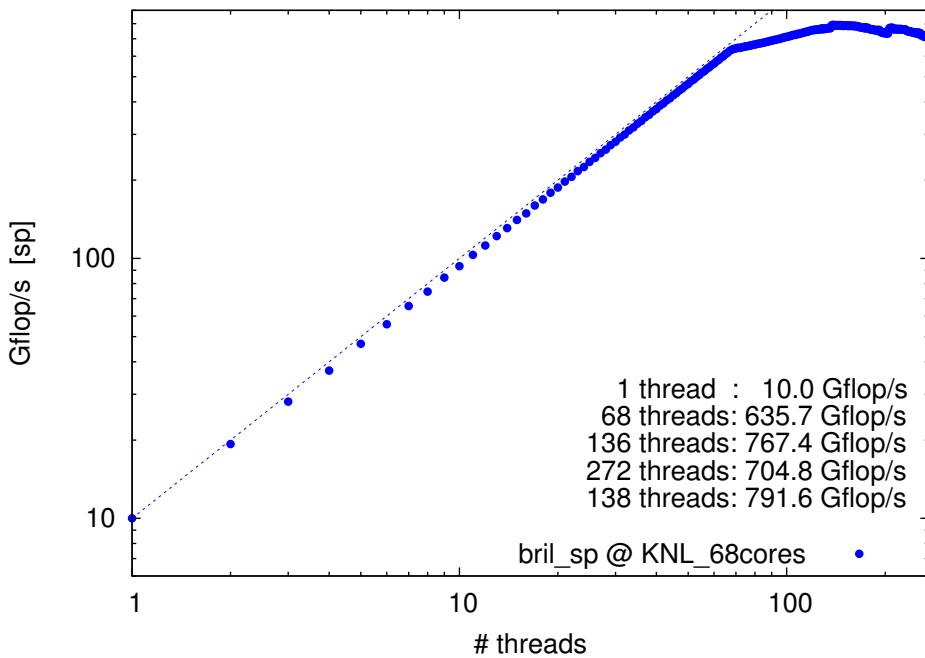
	$N_{\text{thr}} = 1$	$N_{\text{thr}} = 2$	$N_{\text{thr}} = 6$	$N_{\text{thr}} = 12$	$N_{\text{thr}} = 24$	optimum
sp	27.4	53.5	150.8	222.8	222.1	224.1@13
dp	17.2	33.6	94.8	116.3	115.8	116.8@13



- Brillouin performance ratios

KNL: 791/467 Gflop/s out of 6.1/3.05 Tflop/s mean 13.0/15.3% of peak [sp/dp]
 BdW: 224/117 Gflop/s out of 690/345 Gflop/s mean 32.5/33.9% of peak [sp/dp]

Summary

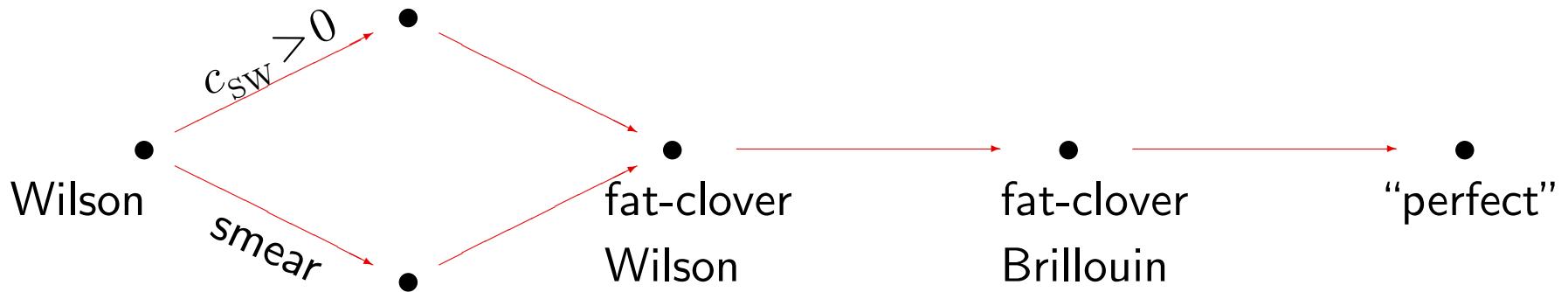


KNL: 791 Gflops [`bril_sp`] is $\sim 13.0\%$ peak, 476 Gflops [`stag_sp`] is $\sim 7.9\%$ peak
Bdw: 224 Gflops [`bril_sp`] is $\sim 32.5\%$ peak, 130 Gflops [`stag_sp`] is $\sim 18.8\%$ peak

- OpenMP parallelization of space-time loop, thread-priv. `site` avoids write-collisions
- SIMD over number of right-hand-sides (N_v), explicit unrolling of color/spinor
- KISS strategy \iff code fully portable \iff “best of breed” ansatz

BACKUP PAGES

Dirac operator roadmap (pedestrian perspective)



Wilson Dirac operator:

$$D_W(x, y) = \sum_{\mu} \gamma_{\mu} \nabla_{\mu}^{\text{std}}(x, y) - \frac{a}{2} \Delta^{\text{std}}(x, y) + m_0 \delta_{x,y} - \frac{c_{\text{SW}}}{2} \sum_{\mu < \nu} \sigma_{\mu\nu} F_{\mu\nu} \delta_{x,y}$$

Brillouin Dirac operator:

$$D_B(x, y) = \sum_{\mu} \gamma_{\mu} \nabla_{\mu}^{\text{iso}}(x, y) - \frac{a}{2} \Delta^{\text{bri}}(x, y) + m_0 \delta_{x,y} - \frac{c_{\text{SW}}}{2} \sum_{\mu < \nu} \sigma_{\mu\nu} F_{\mu\nu} \delta_{x,y}$$

$\sigma_{\mu\nu} = \frac{i}{2} [\gamma_{\mu}, \gamma_{\nu}]$, $F_{\mu\nu}$ the hermitean clover-leaf field-strength tensor, new m_0, c_{SW}

- **Wilson flop count**

For matrix-times-vector operation we must (per site):

(i) spin-project (from 4 to 2) for each direction $\rightarrow 12 \cdot 8 = 96$ flops

(ii) SU(3)-multiply (spin-reduced) for each direction $\rightarrow 6 \cdot 22 \cdot 8 = 1056$ flops

(iii) accumulate $8 + (4+m_0)1$ contributions to out-spinor $\rightarrow 24 \cdot 9 = 216$ flops

All together 1368 flops per site.

- **Brillouin flop count**

For matrix-times-vector operation we must (per site):

(i) SU(3)-multiply (spin-full) for each direction $\rightarrow 12 \cdot 22 \cdot 80 = 21120$ flops

(ii) multiply with $\text{fac_i}/\dots/\text{fac}$: $24 \cdot (54+54+54+54+81) = 7128$ flops

(iii) accumulate 81 contributions to out-spinor $\rightarrow 24 \cdot 80 = 1920$ flops

All together 30168 flops per site.

- **Mini-summary (for details see arXiv:1701.00726)**

The Brillouin-to-Wilson ratio of flops is 22.1

The Brillouin-to-Wilson ratio of memory traffic is 8.9 (next slide)

The Brillouin-to-Wilson ratio of computational intensity is 2.5.

- **Wilson memory traffic**

For matrix-times-vector operation we must (per site, with #rhs=1):

(i) read one sp-spinor for each direction $\rightarrow 24 \cdot 9 = 216$ floats

(ii) read one sp-gauge matrix V per direction $\rightarrow 18 \cdot 8 = 144$ floats

(iii) write one sp-spinor $\rightarrow 24$ floats

All together 384 floats of traffic per site, i.e. 1536 bytes in sp (1.12 bytes/flop).

With 12 rhs this changes to $216+12+24=252$ floats/site/rhs (0.737 bytes/flop).

- **Brillouin memory traffic**

For matrix-times-vector operation we must (per site, with #rhs=1):

(i) read one sp-spinor for each direction $\rightarrow 24 \cdot 81 = 1944$ floats

(ii) read one sp-gauge matrix W per direction $\rightarrow 18 \cdot 80 = 1440$ floats

(iii) write one sp-spinor $\rightarrow 24$ floats

All together 3408 floats of traffic per site, i.e. 13632 bytes in sp (0.45 bytes/flop).

With 12 rhs this changes to $1944+120+24=2088$ floats/site/rhs (0.277 bytes/flop).

- **Mini-summary (for details see arXiv:1701.00726)**

The Brillouin-to-Wilson ratio of memory traffic is $3408/384=8.9$ or $2088/252=8.3$.

Worst case scenario assumed, i.e. everything to be read afresh, i.e. nothing in cache.

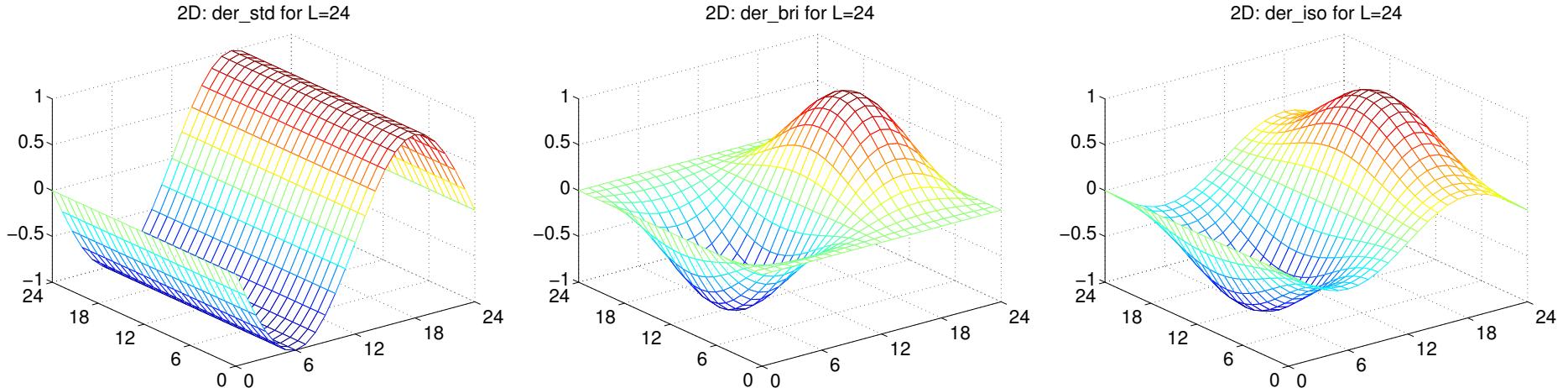
Brillouin operator details

- 3 options for (covariant) Nabla

Standard Derivative: $\hat{\nabla}_x = i \sin(k_1)$

Brillouin Derivative: $\hat{\nabla}_x = i \sin(k_1)[\cos(k_2) + 1][\cos(k_3) + 1][\cos(k_4) + 1]/8$

Isotropic Derivative: $\hat{\nabla}_x = i \sin(k_1)[\cos(k_2) + 2][\cos(k_3) + 2][\cos(k_4) + 2]/27$



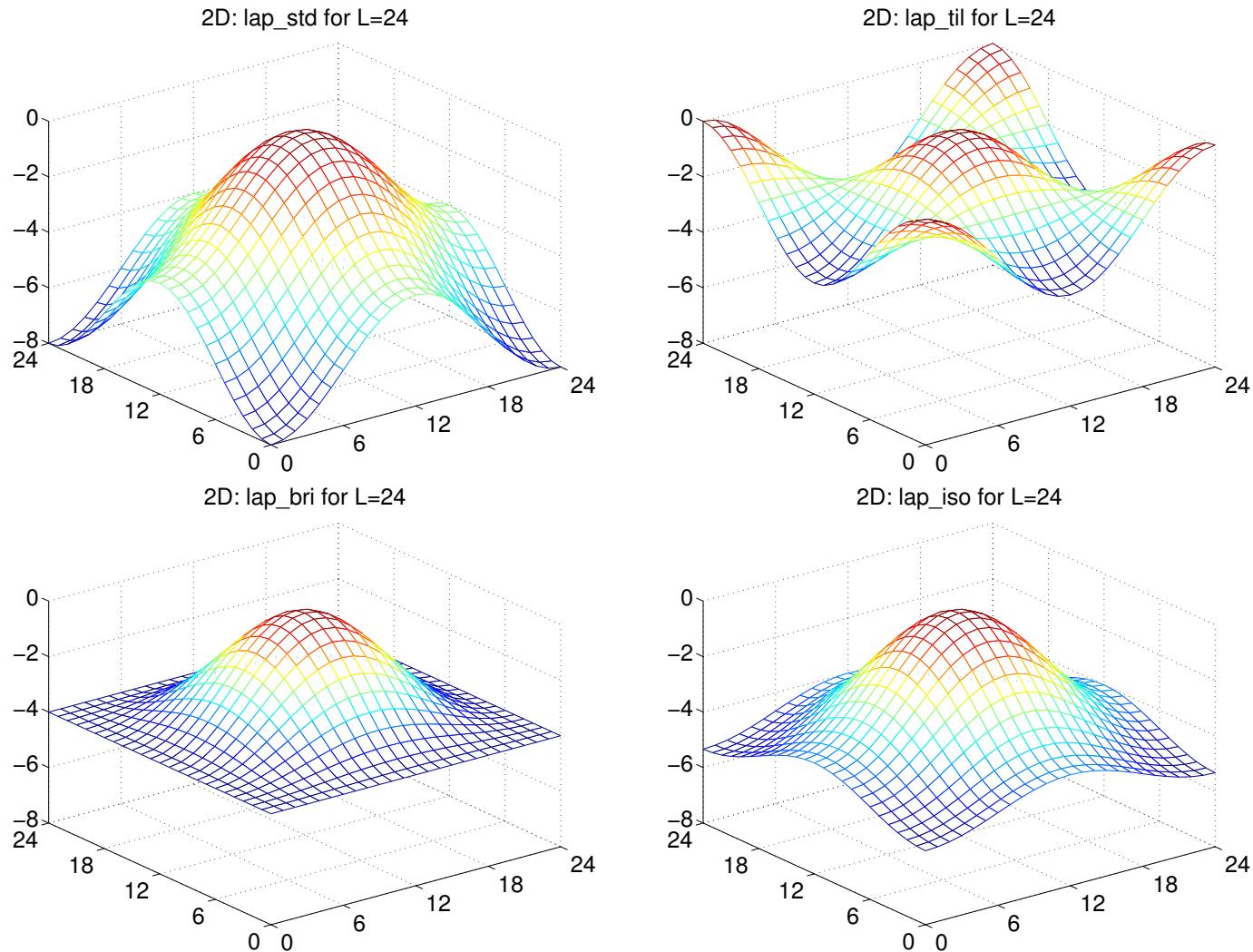
- 4 options for (covariant) Laplacian

Standard Laplacian: $\hat{\Delta} = 2 \cos(k_1) + 2 \cos(k_2) + 2 \cos(k_3) + 2 \cos(k_4) - 8$

Tilted Laplacian: $\hat{\Delta} = 2 \cos(k_1) \cos(k_2) \cos(k_3) \cos(k_4) - 2$

Brillouin Laplacian: $\hat{\Delta} = 4 \cos^2(k_1/2) \cos^2(k_2/2) \cos^2(k_3/2) \cos^2(k_4/2) - 4$

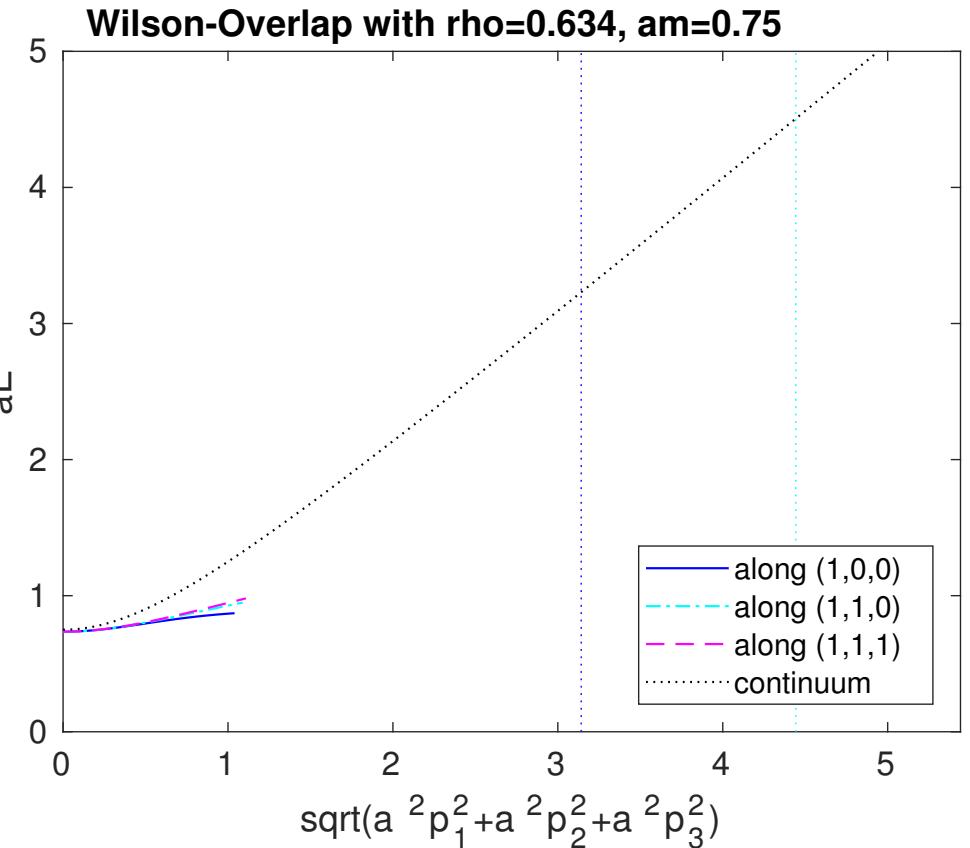
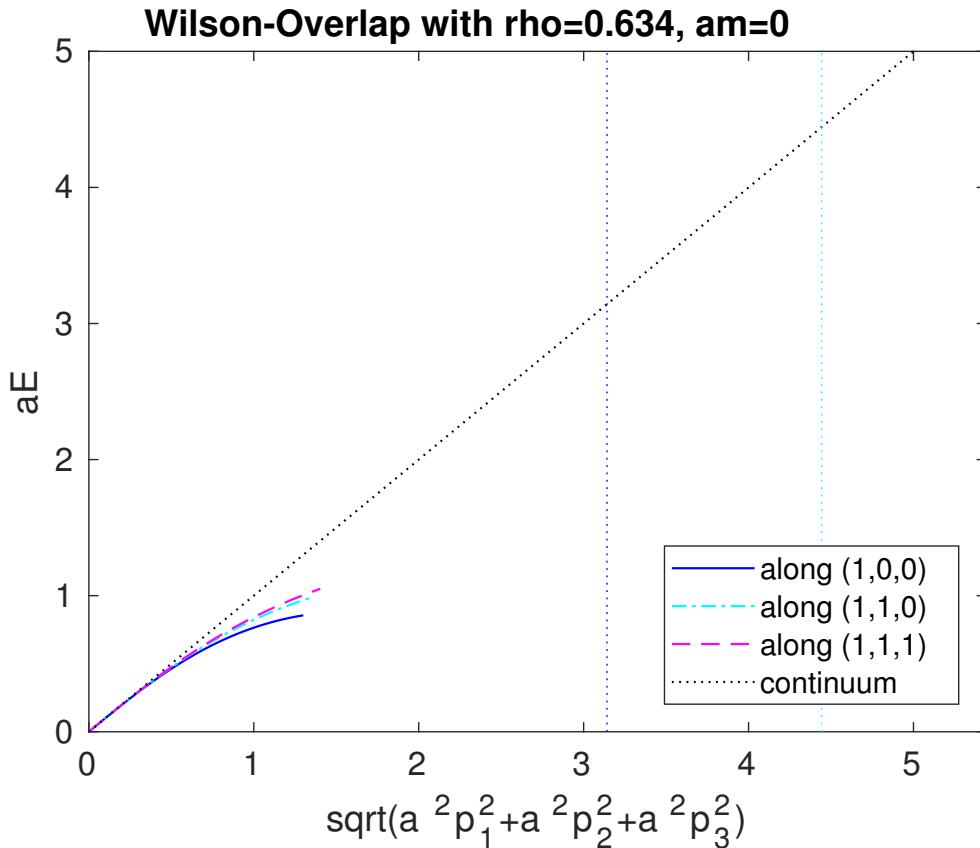
Isotropic Laplacian: $\hat{\Delta} = [2c_1c_2c_3c_4 + 7c_1c_2c_3 + \dots + 20c_1c_2 + \dots + 25c_1 + \dots - 250]/54$



● Selection Procedure

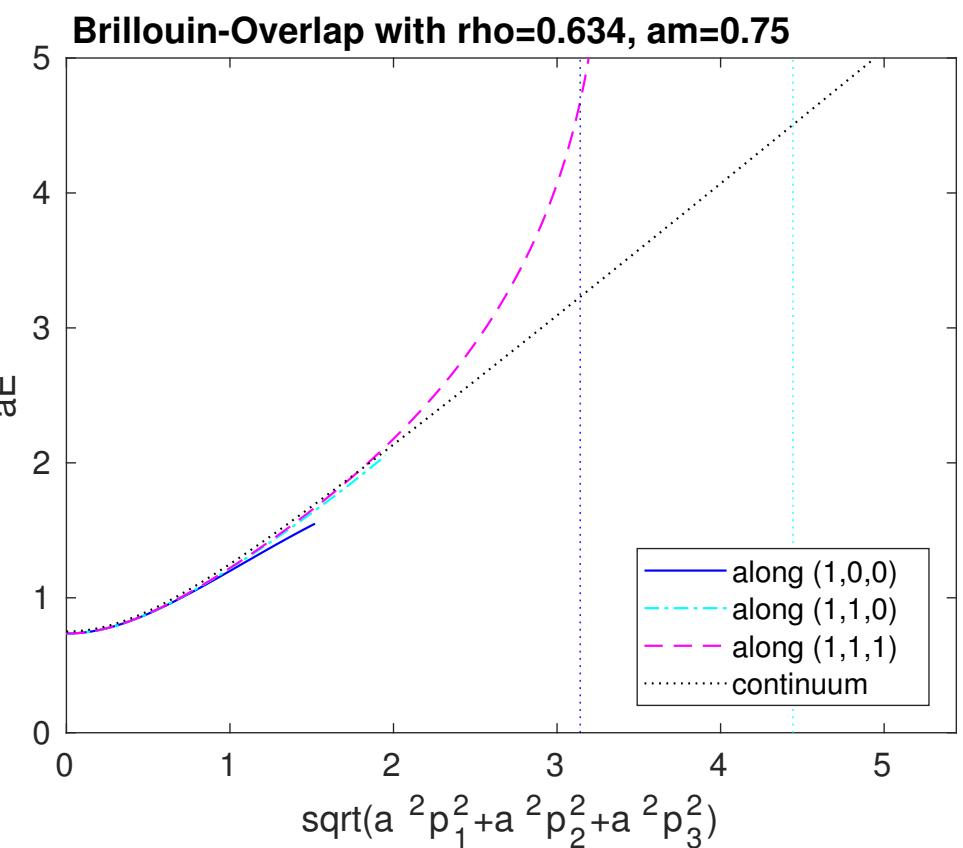
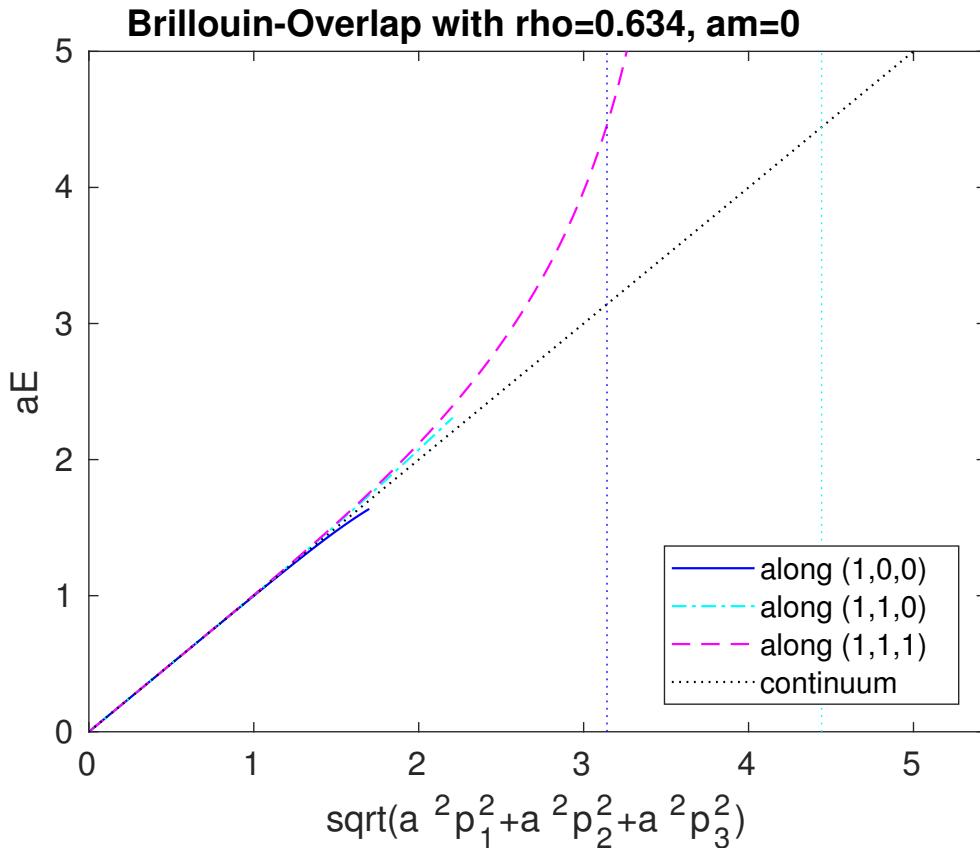
All 12 options live on $[-1 : 1]^4$ hypercube (81 sites in 4D, ultralocal). Test them systematically (eigenvalues, dispersion relation). Combination $(\Delta^{\text{bri}}, \nabla^{\text{iso}})$ prevails.

- Dispersion relation for Wilson-kernel overlap-operator



- ⊖ strong deviation from continuum for any $a|\mathbf{p}| > 1$
- ⊖ strong rotational symmetry breaking for any $a|\mathbf{p}| > 1$
- ⊕ mild effect of $am \ll 1$ for “magic” value $\rho = 0.634$

- Dispersion relation for Brillouin-kernel overlap-operator



- ⊕ mild deviation from continuum up to $a|\mathbf{p}| \simeq 1.5$
- ⊕ mild rotational symmetry breaking up to $a|\mathbf{p}| \simeq 1.5$
- ⊕ mild effect of $am \ll 1$ for “magic” value $\rho = 0.634$

- Cut-off effects for Wilson and Brillouin operators

Wilson operator:

$$\begin{aligned}
 (aE)^2 - (a\mathbf{p})^2 &= \left[(am)^2 - (am)^3 + \frac{11}{12}(am)^4 - \frac{5}{6}(am)^5 \right] \\
 &+ \left[-\frac{2}{3}(am)^2 + \frac{7}{6}(am)^3 \right] (a\mathbf{p})^2 \\
 &+ \left[-\frac{2}{3} + \frac{am}{2} \right] \left(\sum_{i<j} a^4 p_i^2 p_j^2 + \sum_i (ap_i)^4 \right) + O(a^6)
 \end{aligned}$$

Brillouin operator:

$$\begin{aligned}
 (aE)^2 - (a\mathbf{p})^2 &= \left[(am)^2 - (am)^3 + \frac{11}{12}(am)^4 - \frac{5}{6}(am)^5 \right] \\
 &+ \left[0 + \frac{1}{12}(am)^3 \right] (a\mathbf{p})^2 \\
 &+ \left[0 + \frac{am}{12} \right] \left(\sum_{i<j} a^4 p_i^2 p_j^2 + \sum_i (ap_i)^4 \right) + O(a^6)
 \end{aligned}$$

Note: Fermilab reinterpretation manages to get rid of $-(am)^3 + \dots$ part at $\mathbf{p} = 0$

- **Cut-off effects for Wilson and Brillouin overlap operators**

Overlap operator with Wilson kernel:

$$\begin{aligned}
 (aE)^2 - (a\mathbf{p})^2 &= \left[(am)^2 - \frac{2\rho^2 - 6\rho + 3}{6\rho^2} (am)^4 \right] \\
 &+ \left[-\frac{2}{3}(am)^2 + 0 \right] (a\mathbf{p})^2 \\
 &+ \left[-\frac{2}{3} + 0 \right] \left(\sum_{i < j} a^4 p_i^2 p_j^2 + \sum_i (ap_i)^4 \right) + O(a^6)
 \end{aligned}$$

Overlap operator with Brillouin kernel:

$$\begin{aligned}
 (aE)^2 - (a\mathbf{p})^2 &= \left[(am)^2 - \frac{2\rho^2 - 6\rho + 3}{6\rho^2} (am)^4 \right] \\
 &+ \left[0 + 0 \right] (a\mathbf{p})^2 \\
 &+ \left[0 + 0 \right] \left(\sum_{i < j} a^4 p_i^2 p_j^2 + \sum_i (ap_i)^4 \right) + O(a^6)
 \end{aligned}$$

Note: For details see arXiv:1701.00726