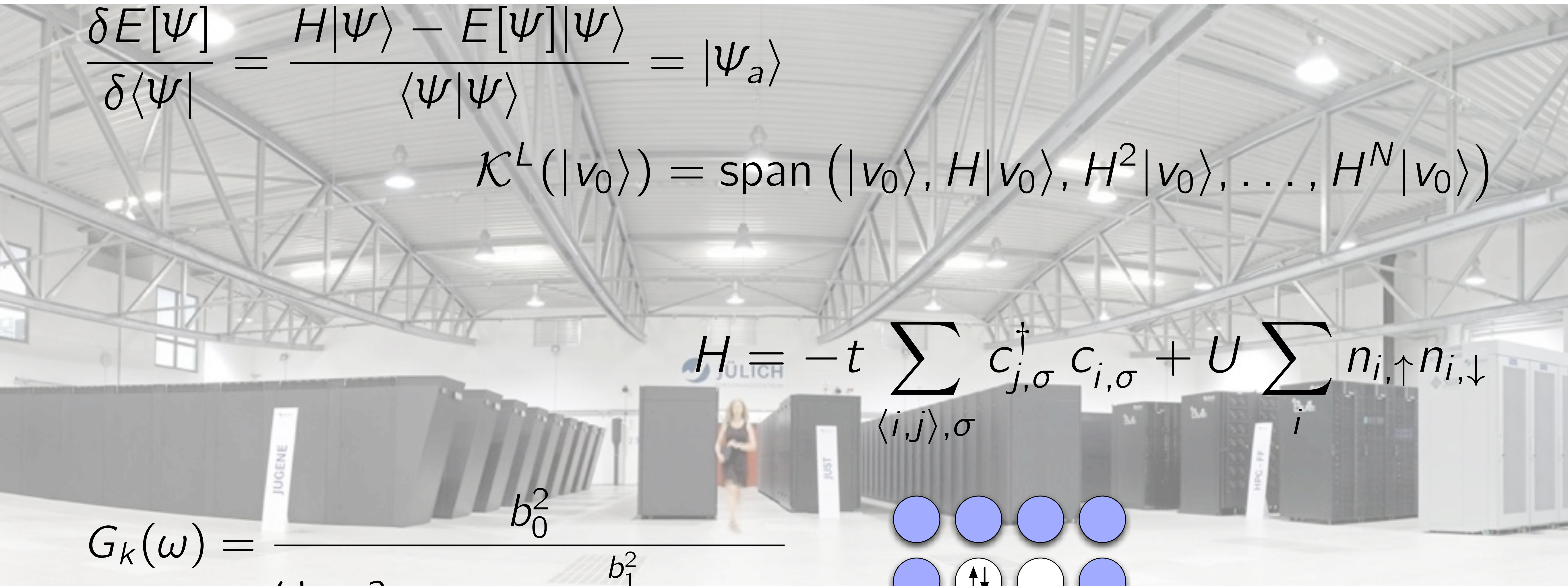
The Lanczos Method

Erik Koch – GRS Jülich

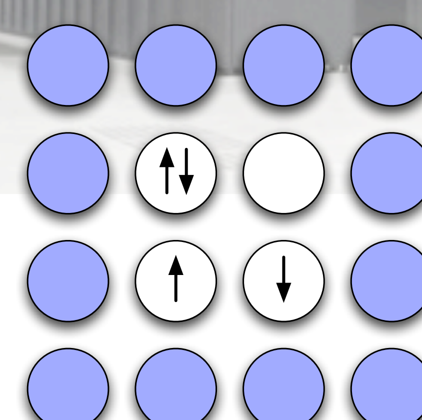


$$\frac{\delta E[\Psi]}{\delta\langle\Psi|} = \frac{H|\Psi\rangle - E[\Psi]|\Psi\rangle}{\langle\Psi|\Psi\rangle} = |\Psi_a\rangle$$

$$\mathcal{K}^L(|v_0\rangle) = \text{span}\left(|v_0\rangle, H|v_0\rangle, H^2|v_0\rangle, \ldots, H^N|v_0\rangle\right)$$

$$H = -t\sum_{\langle i,j\rangle,\sigma} c^\dagger_{j,\sigma}c_{i,\sigma} + U\sum_i n_{i,\uparrow}n_{i,\downarrow}$$

$$G_k(\omega) = \cfrac{b_0^2}{\omega - a_0 - \cfrac{b_1^2}{\omega - a_1 - \cfrac{b_2^2}{\omega - a_2 - \cfrac{b_3^2}{\omega - a_3 - \cdots}}}}$$



German Research School for Simulation Sciences

Thursday, October 6, 2011

Why Lanczos?

- ☑ numerically exact solution
- ☑ efficient for sparse Hamiltonians
- ☑ ground state (T=0) or finite (but low) temperature
- ☑ spectral function on real axis

- ☐ only finite (actually quite small) systems
  - ○ efficient parallelization to use shared memory
  - ○ optimal bath parametrization

steepest descent

energy functional

$$E[\Psi] = \frac{\langle\Psi|H|\Psi\rangle}{\langle\Psi|\Psi\rangle}$$

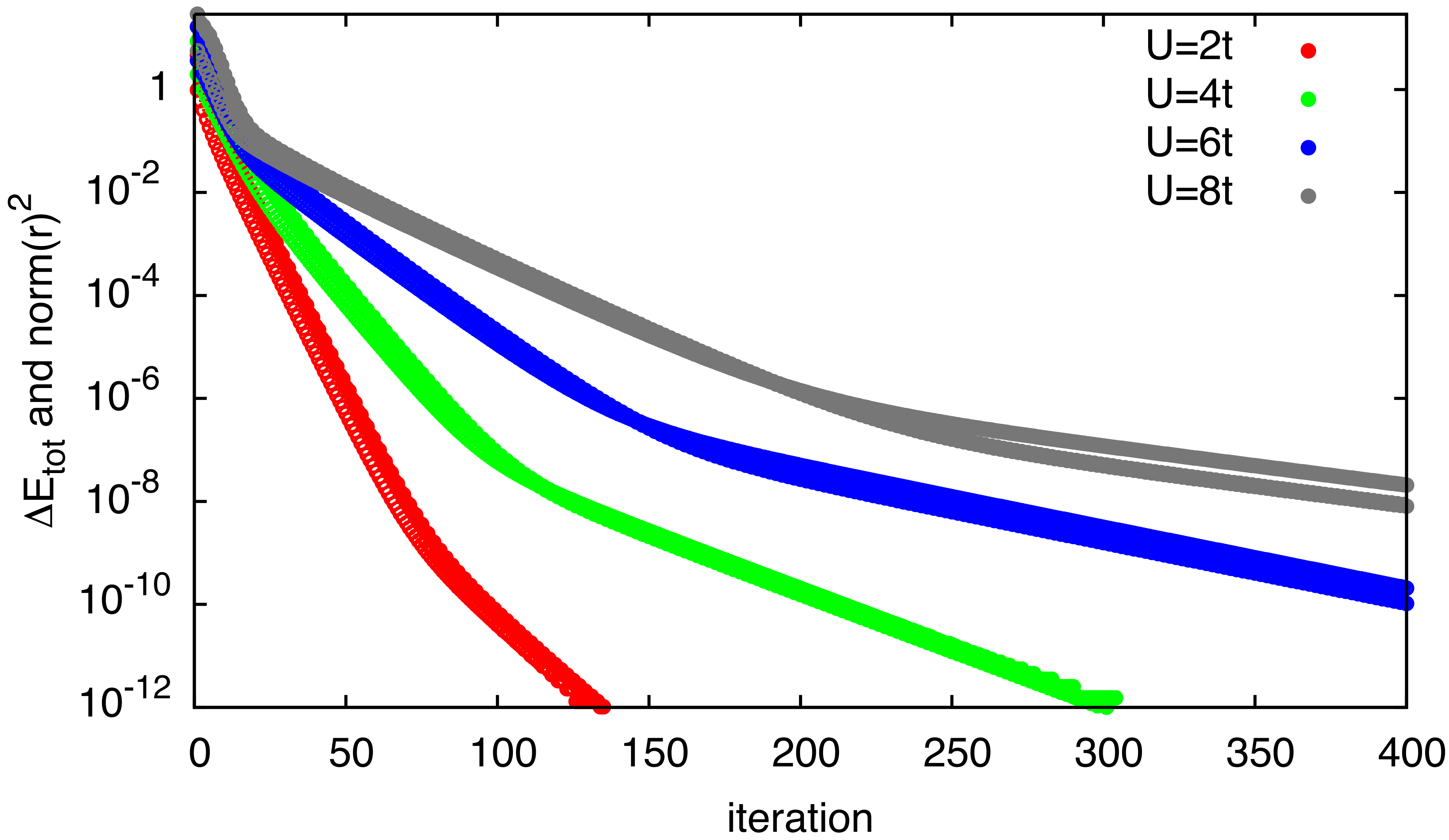
direction (in Hilbert space) of steepest ascent

$$\frac{\delta E[\Psi]}{\delta\langle\Psi|} = \frac{H|\Psi\rangle - E[\Psi]|\Psi\rangle}{\langle\Psi|\Psi\rangle} = |\Psi_a\rangle \in \mathrm{span}\,(|\Psi\rangle, H|\Psi\rangle)$$

minimize energy in $\mathrm{span}\,(|\Psi\rangle, H|\Psi\rangle)$

iterate!

convergence



Lanczos idea

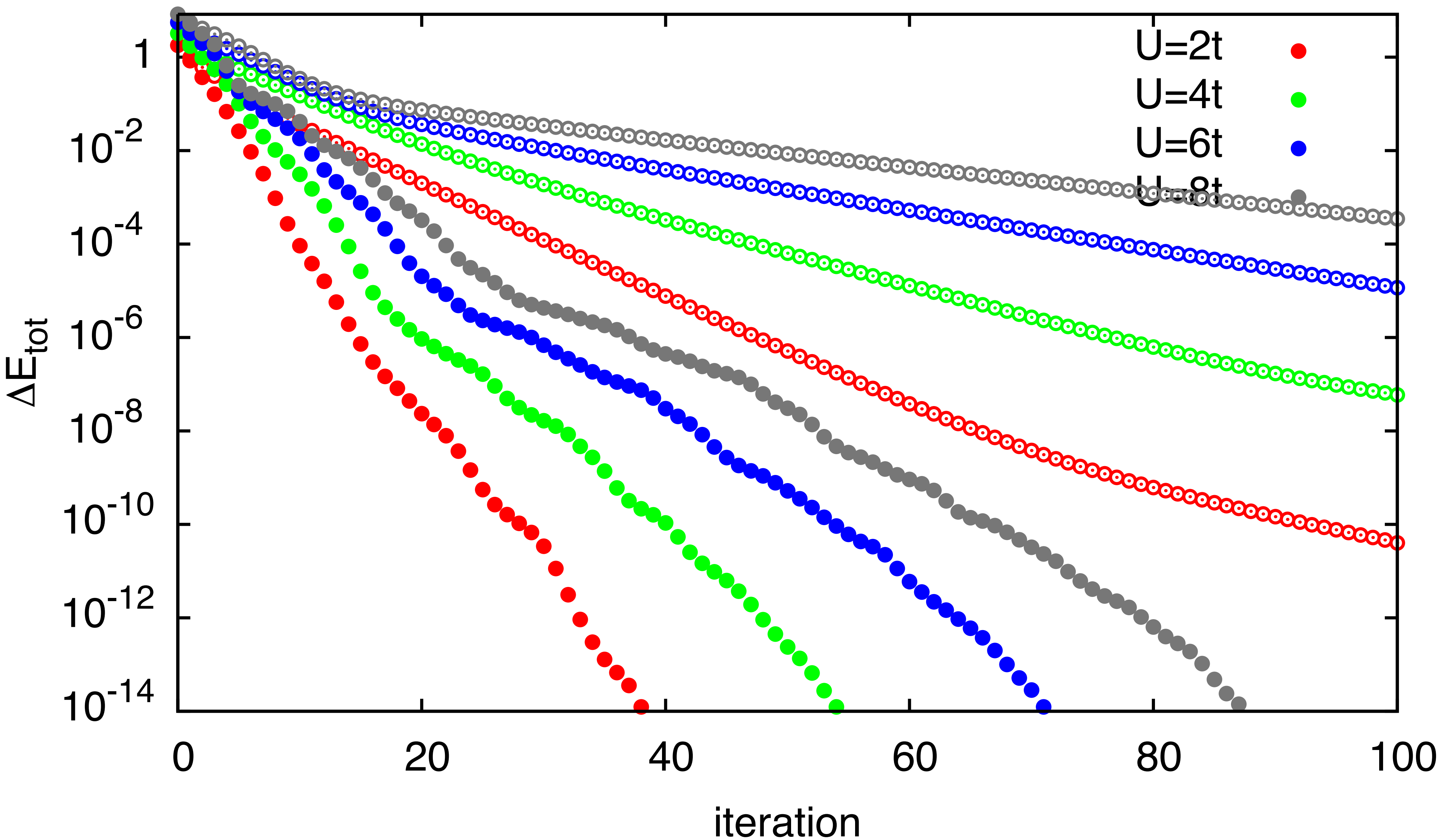
minimize on span $(|\Psi_0\rangle, H|\Psi_0\rangle)$ to obtain $|\Psi_1\rangle$
minimize on span $(|\Psi_1\rangle, H|\Psi_1\rangle) \in \text{span}\left(|\Psi_0\rangle, H|\Psi_0\rangle, H^2|\Psi_0\rangle\right)$
minimize on span $(|\Psi_2\rangle, H|\Psi_2\rangle) \in \text{span}\left(|\Psi_0\rangle, H|\Psi_0\rangle, H^2|\Psi_0, H^3|\Psi_0\rangle\right)$
etc.

instead of L-fold iterative minimization on two-dimensional subspaces
minimize energy on *L+1 dimensional* **Krylov space**

$$\mathcal{K}^L(\Psi_0\rangle) = \text{span}\left(|\Psi_0\rangle, H|\Psi_0\rangle, H^2|\Psi_0\rangle, \dots, H^L|\Psi_0\rangle\right)$$

more variational degrees of freedom ⇒ even faster convergence

convergence to ground state



Lanczos iteration

orthonormal basis in Krylov space

$$
\begin{aligned}
& |v_0\rangle \\
b_1\,|v_1\rangle &= H|v_0\rangle - a_0|v_0\rangle \\
b_2\,|v_2\rangle &= H|v_1\rangle - a_1|v_1\rangle - b_1|v_0\rangle \\
b_3\,|v_3\rangle &= H|v_2\rangle - a_2|v_2\rangle - b_2|v_1\rangle \\
&\cdots
\end{aligned}
$$

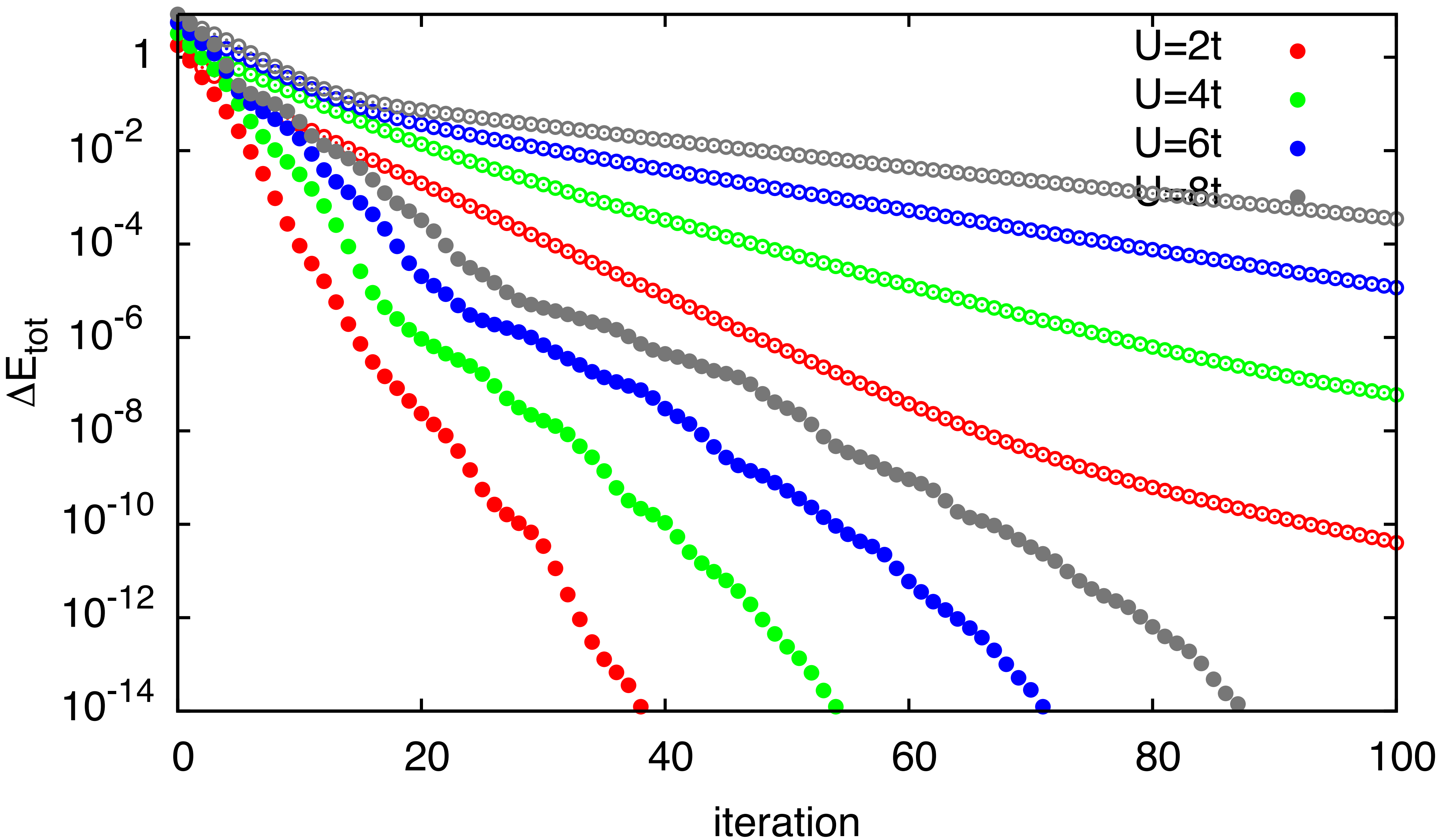
$$
H|v_n\rangle = b_n|v_{n-1}\rangle + a_n|v_n\rangle + b_{n+1}|v_{n+1}\rangle
$$

$$
H_{\mathcal{K}^L(|v_0\rangle)} = \begin{pmatrix}
a_0 & b_1 & 0 & 0 & & 0 & 0 \\
b_1 & a_1 & b_2 & 0 & \cdots & 0 & 0 \\
0 & b_2 & a_2 & b_3 & & 0 & 0 \\
0 & 0 & b_3 & a_3 & & 0 & 0 \\
 & \vdots & & & \ddots & \vdots & \\
0 & 0 & 0 & 0 & & a_{L-1} & b_L \\
0 & 0 & 0 & 0 & \cdots & b_L & a_L
\end{pmatrix}
$$

Lanczos algorithm

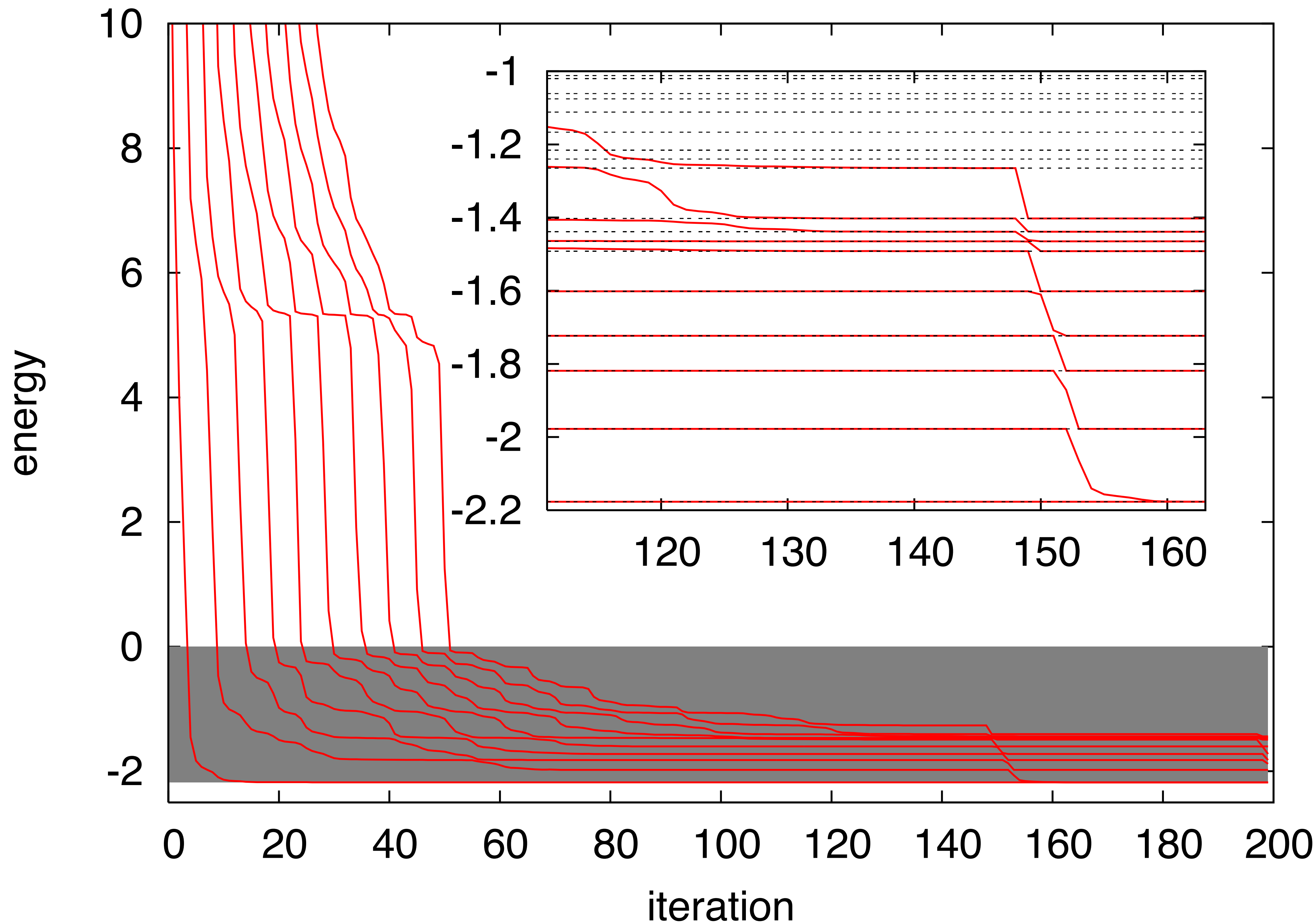
`v=init`	
`b0=norm2(v)`	not part of tridiagonal matrix
`scal(1/b0,v)`	v= $\lvert v_0 \rangle$
`w=0`	
`w=w+H*v`	w= $H\lvert v_0 \rangle$
`a[0]=dot(v,w)`	
`axpy(-a[0],v,w)`	w= $\lvert \tilde{v}_1 \rangle = H\lvert v_0 \rangle - a_0 \lvert v_0 \rangle$
`b[1]=norm2(w)`	
`for n=1,2,...`	
`  if abs(b[n])<eps then exit`	invariant subspace
`  scal(1/b[n],w)`	w= $\lvert v_n \rangle$
`  scal( -b[n],v)`	v= $-b_n \lvert v_{n-1} \rangle$
`  swap(v,w)`	
`  w=w+H*v`	w= $H\lvert v_n \rangle - b_n \lvert v_{n-1} \rangle$
`  a[n]=dot(v,w)`	a[n]= $\langle v_n \lvert H \rvert v_n \rangle - b_n \langle v_n \vert v_{n-1} \rangle$
`  axpy(-a[n],v,w)`	w= $\lvert \tilde{v}_{n+1} \rangle$
`  b[n+1]=norm2(w)`	
`  diag(a[0]..a[n], b[1]..b[n])`	getting a_{n+1} needs another $H\lvert v \rangle$
`  if converged then exit`	
`end`	

convergence to ground state



$$\frac{\check{E}_0 - E_0}{E_N - E_0} \leq \left(\frac{\tan(\arccos(\langle \check{\Psi}_0 | \Psi_0 \rangle))}{T_L\left(1 + 2\,\frac{E_1 - E_0}{E_N - E_1}\right)} \right)^2$$

overconvergence: ghost states



$$b_{n+1}|v_{n+1}\rangle = H|v_n\rangle - a_n|v_n\rangle - bn|v_{n-1}\rangle$$

resolvent / spectral function

$$G_c(z) = \left\langle \Psi_c \left| \frac{1}{z - H} \right| \Psi_c \right\rangle = \sum_{n=0}^{N} \frac{\langle \Psi_c | \Psi_n \rangle \langle \Psi_n | \Psi_c \rangle}{z - E_n}$$

$$\check{G}_c(z) = \left\langle \Psi_c \left| \frac{1}{z - \check{H}_c} \right| \Psi_c \right\rangle = \sum_{n=0}^{L} \frac{\langle \Psi_c | \check{\Psi}_n \rangle \langle \check{\Psi}_n | \Psi_c \rangle}{z - \check{E}_n}$$

$$z - \check{H}_c = \left(\begin{array}{c|cccccc} z - a_0 & -b_1 & 0 & 0 & \cdots & 0 & 0 \\ \hline -b_1 & z - a_1 & -b_2 & 0 & \cdots & 0 & 0 \\ 0 & -b_2 & z - a_2 & -b_3 & \cdots & 0 & 0 \\ 0 & 0 & -b_3 & z - a_3 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & z - a_{L-1} & -b_L \\ 0 & 0 & 0 & 0 & \cdots & -b_L & z - a_L \end{array} \right)$$

resolvent / spectral function

$$z - \check{H}_c = \begin{pmatrix} z - a_0 & {B^{(1)}}^T \\ B^{(1)} & z - \check{H}_c^{(1)} \end{pmatrix}$$

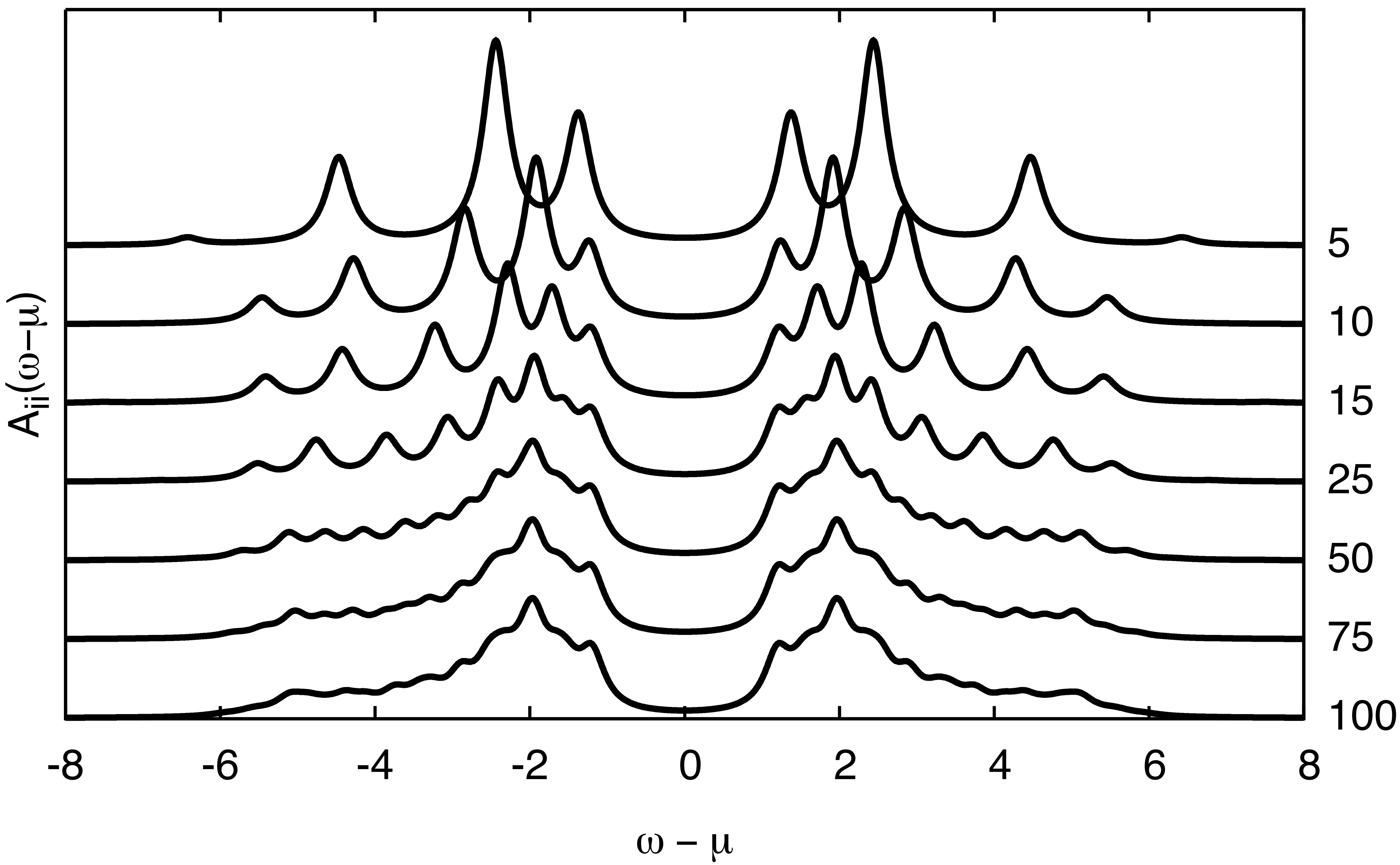
inversion by partitioning

$$\begin{aligned} \left[(z - \check{H}_c)^{-1}\right]_{00} &= \left(z - a_0 - {B^{(1)}}^T (z - \check{H}_c^{(1)})^{-1} B^{(1)}\right)^{-1} \\ &= \left(z - a_0 - b_1^2 \left[(z - \check{H}_c^{(1)})^{-1}\right]_{00}\right)^{-1} \end{aligned}$$

recursively

$$\check{G}_c(z) = \left[(z - \check{H}_c)^{-1}\right]_{00} = \cfrac{1}{z - a_0 - \cfrac{b_1^2}{z - a_1 - \cfrac{b_2^2}{z - a_2 - \cdots}}}$$

convergence: moments



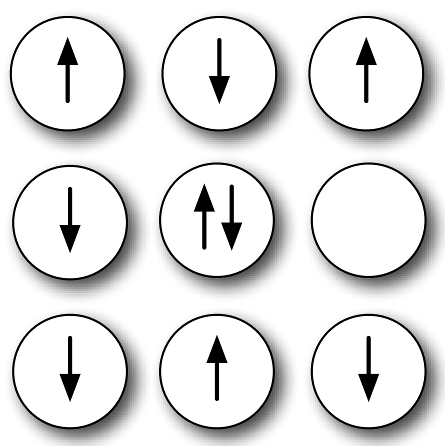
$$\int_{-\infty}^{\infty} d\omega\, \omega^m \check{A}(\omega) = \sum_{n=0}^{L} |\check{\psi}_{n,0}|^2 \check{E}_n^m = \sum_{n=0}^{L} \langle \Psi_c | \check{\Psi}_n \rangle \langle \check{\Psi}_n | \Psi_c \rangle \, \check{E}_n^m = \langle \Psi_c | \check{H}^m | \Psi_c \rangle$$

application to Hubbard model and shared-memory parallelization

dimension of many-body Hilbert space

$$H = -t \sum_{\langle i,j \rangle, \sigma} c^\dagger_{j,\sigma} c_{i,\sigma} + U \sum_i n_{i,\uparrow} n_{i,\downarrow}$$

solve finite clusters



$$\dim(H) = \binom{M}{N_\uparrow} \times \binom{M}{N_\downarrow}$$

M	$N_\uparrow$	$N_\downarrow$	dimension of Hilbert space	memory
2	1	1	4	
4	2	2	36	
6	3	3	400	
8	4	4	4 900	
10	5	5	63 504	
12	6	6	853 776	6 MB
14	7	7	11 778 624	89 MB
16	8	8	165 636 900	1 263 MB
18	9	9	2 363 904 400	18 GB
20	10	10	34 134 779 536	254 GB
22	11	11	497 634 306 624	3708 GB
24	12	12	7 312 459 672 336	53 TB

choice of basis

real space: sparse Hamiltonian

$$H = -t \sum_{\langle i,j \rangle,\sigma} c^\dagger_{j,\sigma} c_{i,\sigma} + U \sum_i n_{i,\uparrow} n_{i,\downarrow}$$

hopping only connects states of same spin
interaction diagonal (even for long-range interaction!)

k-space

$$H = \sum_{k\sigma} \varepsilon_k c^\dagger_{k\sigma} c_{k\sigma} + \frac{U}{M} \sum_{k,k',q} c^\dagger_{k\uparrow} c_{k-q,\uparrow} c^\dagger_{k'\downarrow} c_{k'+q,\downarrow}$$

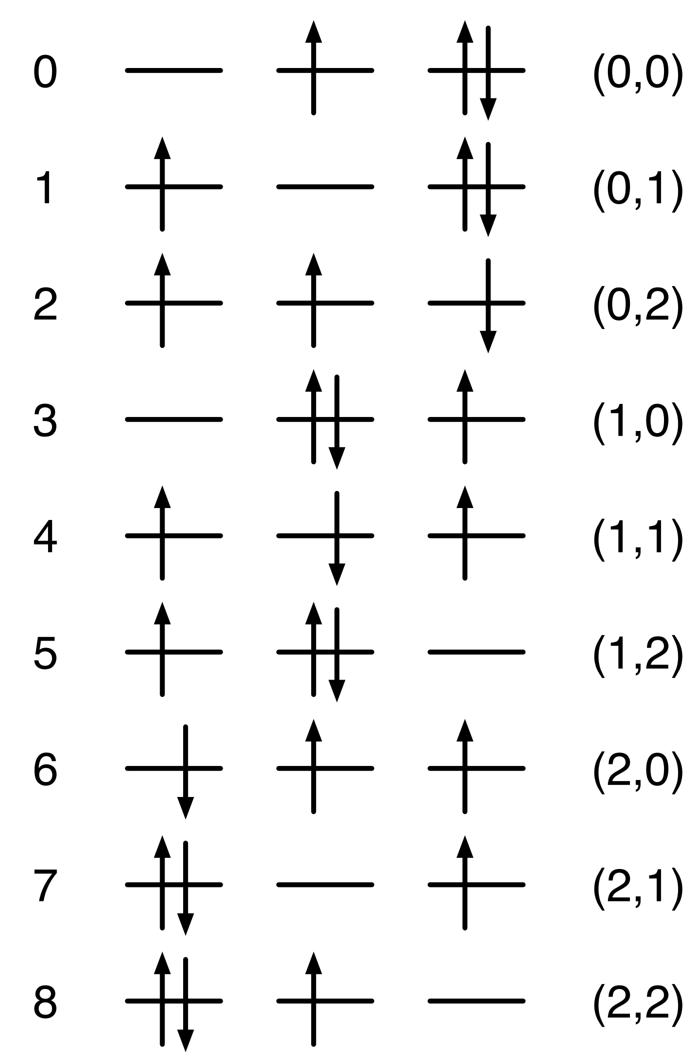
choice of basis

work with operators that create electrons in Wannier orbitals

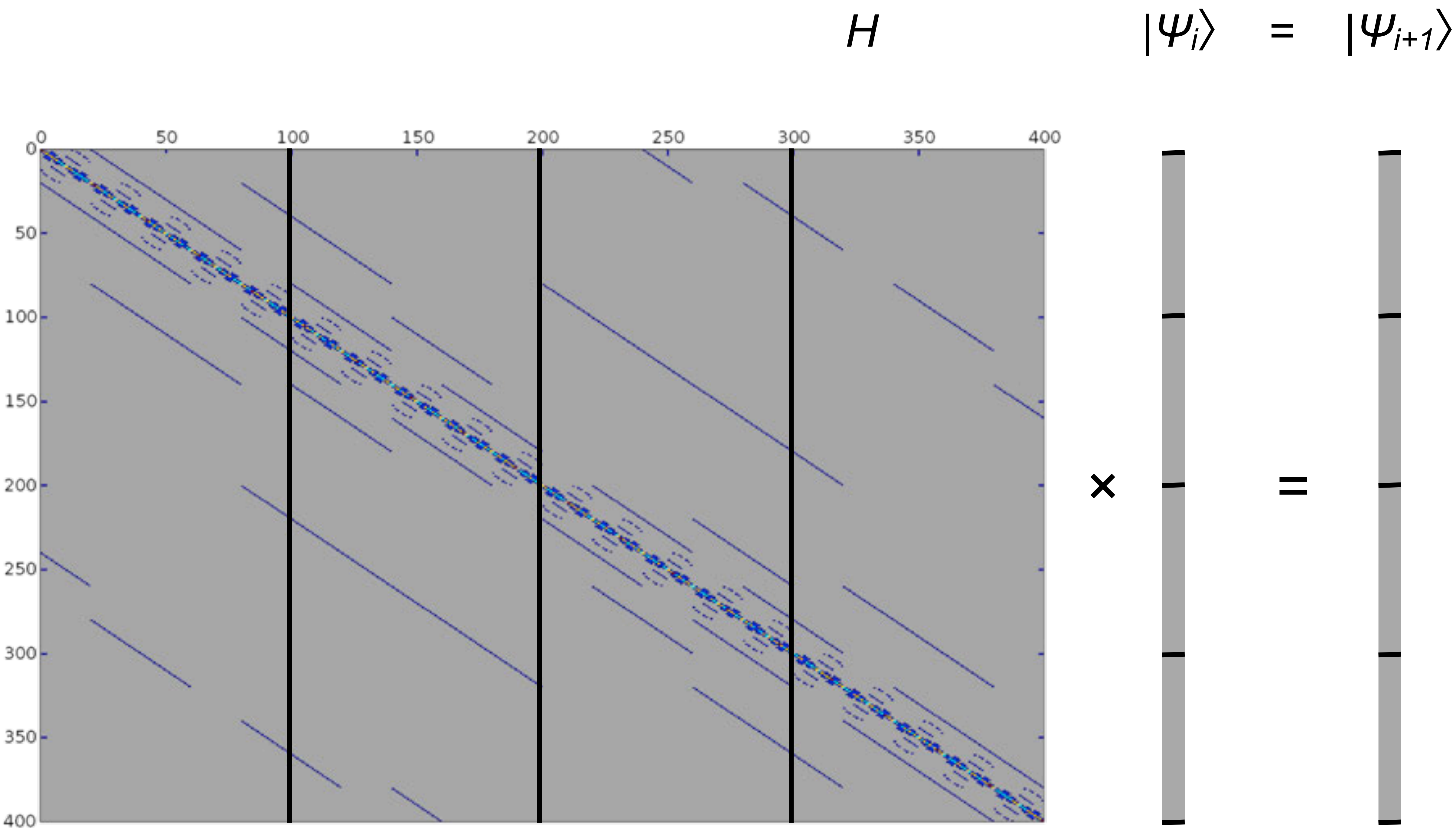
$$|\{n_{i\sigma}\}\rangle = \prod_{i=0}^{L-1} \left(c_{i\downarrow}^\dagger\right)^{n_{i\downarrow}} \left(c_{i\uparrow}^\dagger\right)^{n_{i\uparrow}} |0\rangle$$

$m_\uparrow$	bits	state	$i_\uparrow$
0	000		
1	001		
2	010		
3	011	$c_{0\uparrow}^\dagger c_{1\uparrow}^\dagger \|0\rangle$	0
4	100		
5	101	$c_{0\uparrow}^\dagger c_{2\uparrow}^\dagger \|0\rangle$	1
6	110	$c_{1\uparrow}^\dagger c_{2\uparrow}^\dagger \|0\rangle$	2
7	111		

$m_\downarrow$	bits	state	$i_\downarrow$
0	000		
1	001	$c_{0\downarrow}^\dagger \|0\rangle$	0
2	010	$c_{1\downarrow}^\dagger \|0\rangle$	1
3	011		
4	100	$c_{2\downarrow}^\dagger \|0\rangle$	2
5	101		
6	110		
7	111		



sparse matrix-vector product



sparse matrix-vector product: OpenMP

$w = w + H\,v$

$$H = \sum_{\langle ij\rangle,\sigma} t_{i,j}\, c^\dagger_{j,\sigma} c_{i,\sigma} + U \sum_i n_{i,\uparrow} n_{i,\downarrow}$$

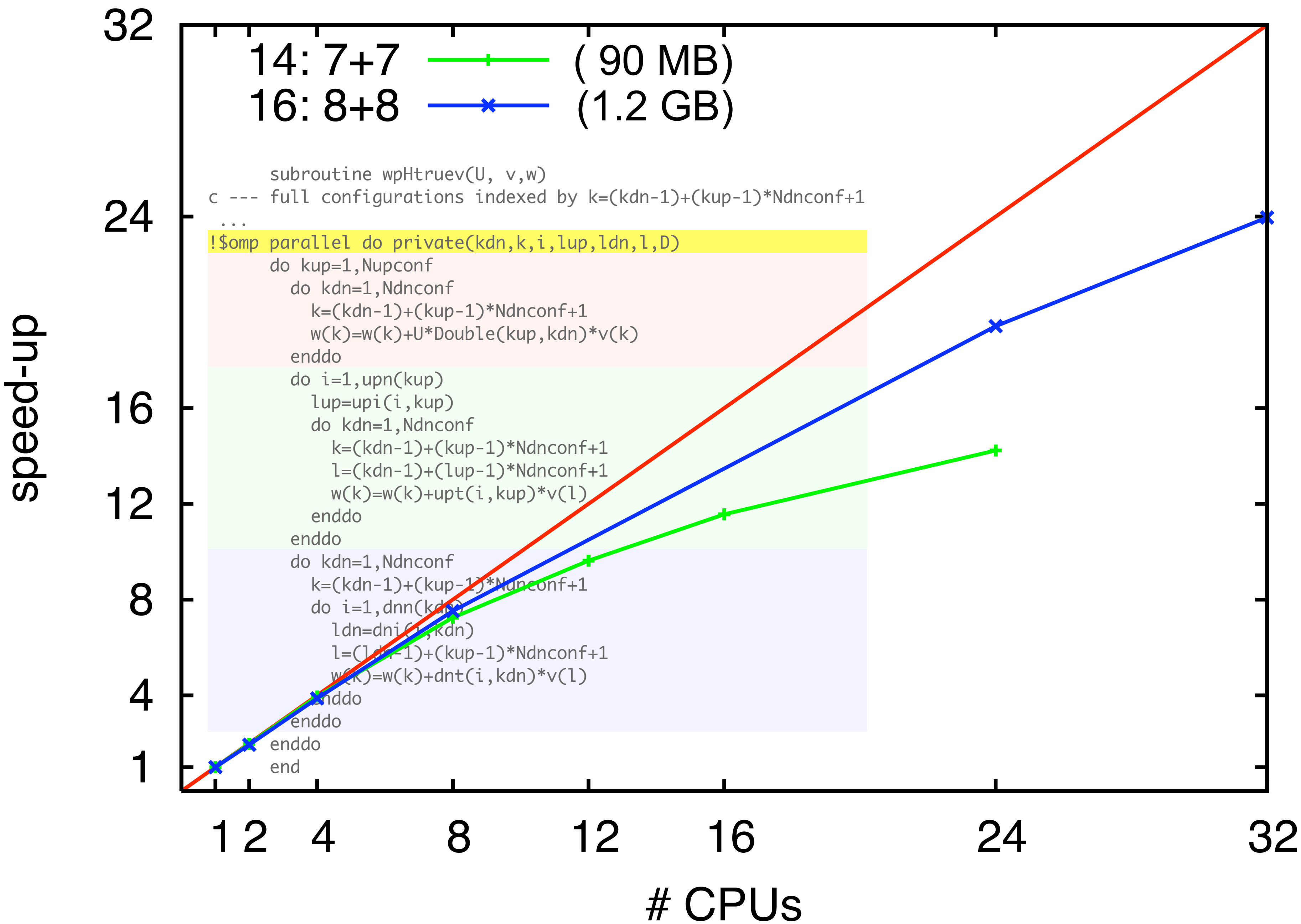
```
      subroutine wpHtruev(U, v,w)
c --- full configurations indexed by k=(kdn-1)+(kup-1)*Ndnconf+1
  ...
!$omp parallel do private(kdn,k,i,lup,ldn,l,D)
      do kup=1,Nupconf
        do kdn=1,Ndnconf
          k=(kdn-1)+(kup-1)*Ndnconf+1
          w(k)=w(k)+U*Double(kup,kdn)*v(k)
        enddo
        do i=1,upn(kup)
          lup=upi(i,kup)
          do kdn=1,Ndnconf
            k=(kdn-1)+(kup-1)*Ndnconf+1
            l=(kdn-1)+(lup-1)*Ndnconf+1
            w(k)=w(k)+upt(i,kup)*v(l)
          enddo
        enddo
        do kdn=1,Ndnconf
          k=(kdn-1)+(kup-1)*Ndnconf+1
          do i=1,dnn(kdn)
            ldn=dni(i,kdn)
            l=(ldn-1)+(kup-1)*Ndnconf+1
            w(k)=w(k)+dnt(i,kdn)*v(l)
          enddo
        enddo
      enddo
      end
```

$$U \sum_i n_{i,\uparrow} n_{i,\downarrow}$$

$$\sum_{\langle ij\rangle,\sigma=\uparrow} t_{i,j}\, c^\dagger_{j,\sigma} c_{i,\sigma}$$

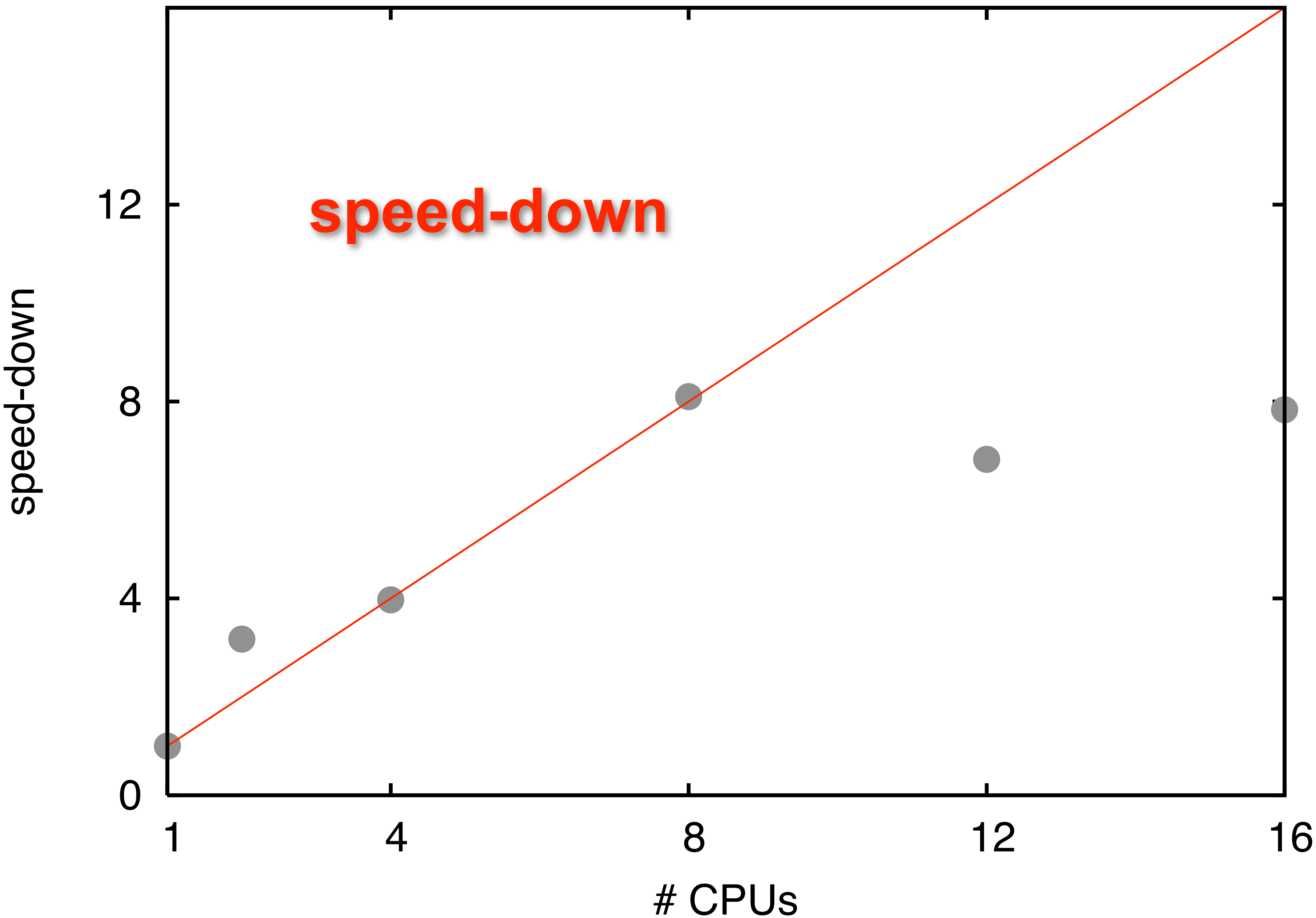
$$\sum_{\langle ij\rangle,\sigma=\downarrow} t_{i,j}\, c^\dagger_{j,\sigma} c_{i,\sigma}$$

OpenMP on Jump



distributed memory

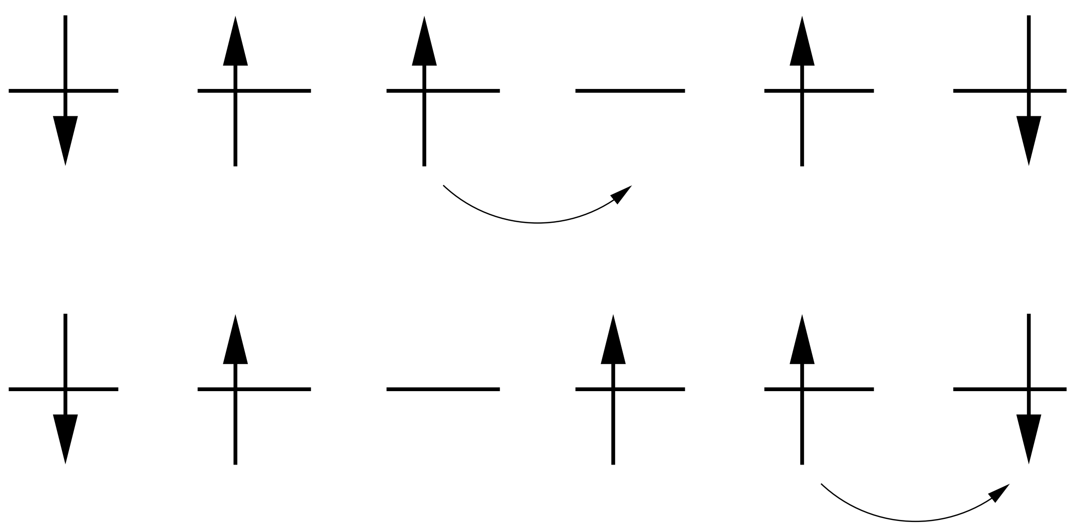
MPI-2: one-sided communication



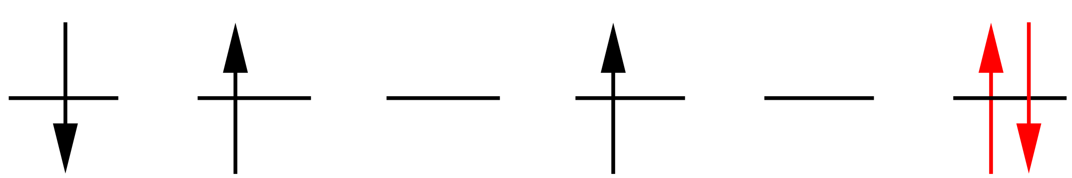
Hubbard model

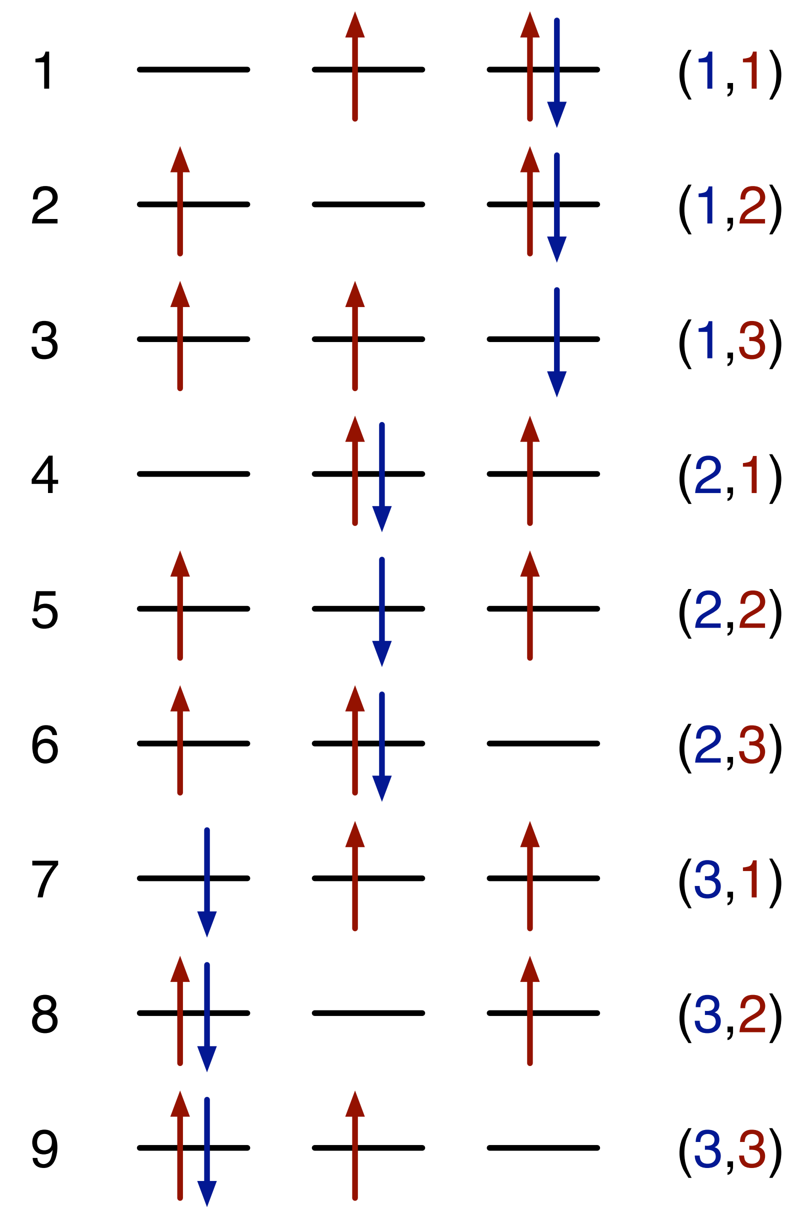
$$H = \sum_{\langle ij \rangle,\sigma} t_{i,j}\, c^{\dagger}_{j,\sigma} c_{i,\sigma} + U \sum_i n_{i,\uparrow} n_{i,\downarrow}$$

hopping: spin unchanged

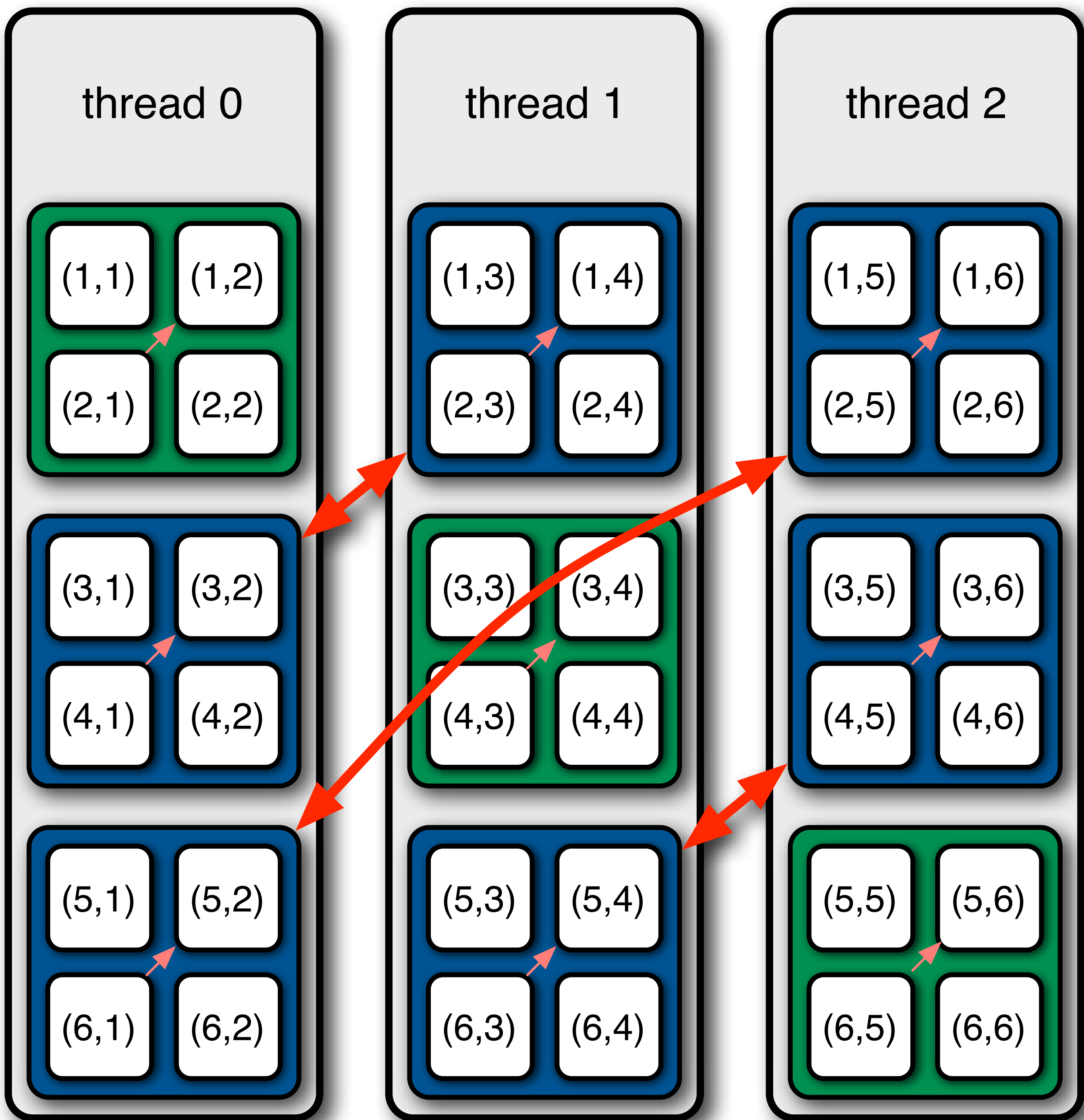


interaction diagonal





Idea: matrix transpose of $v(i_\downarrow,i_\uparrow)$



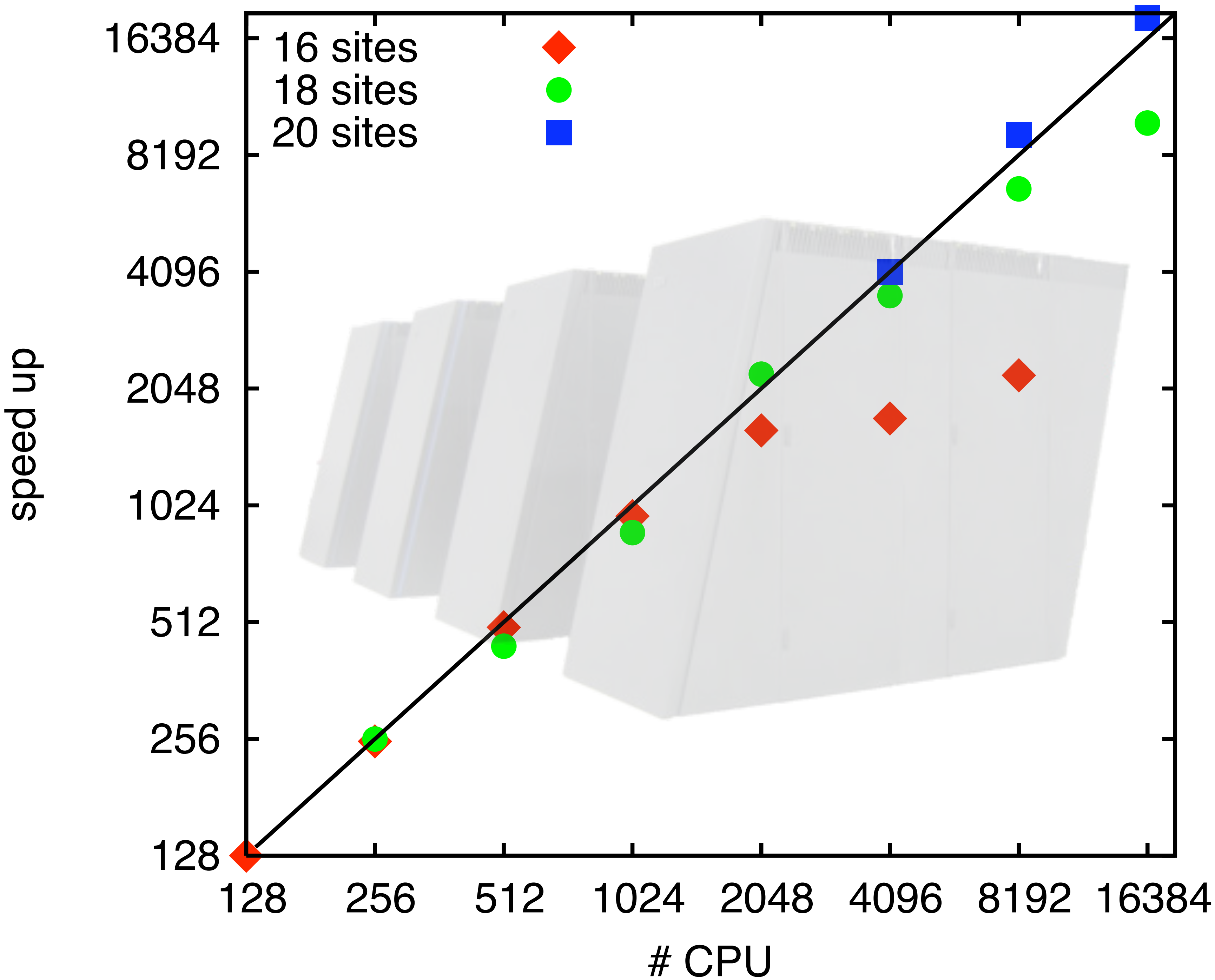
Lanczos-vector as matrix:
$v(i_\downarrow,i_\uparrow)$

before transpose: ↓-hops local
after transpose: ↑-hops local

implementation:

`MPI_alltoall` $(N_\downarrow = N_\uparrow)$
`MPI_alltoallv` $(N_\downarrow \neq N_\uparrow)$

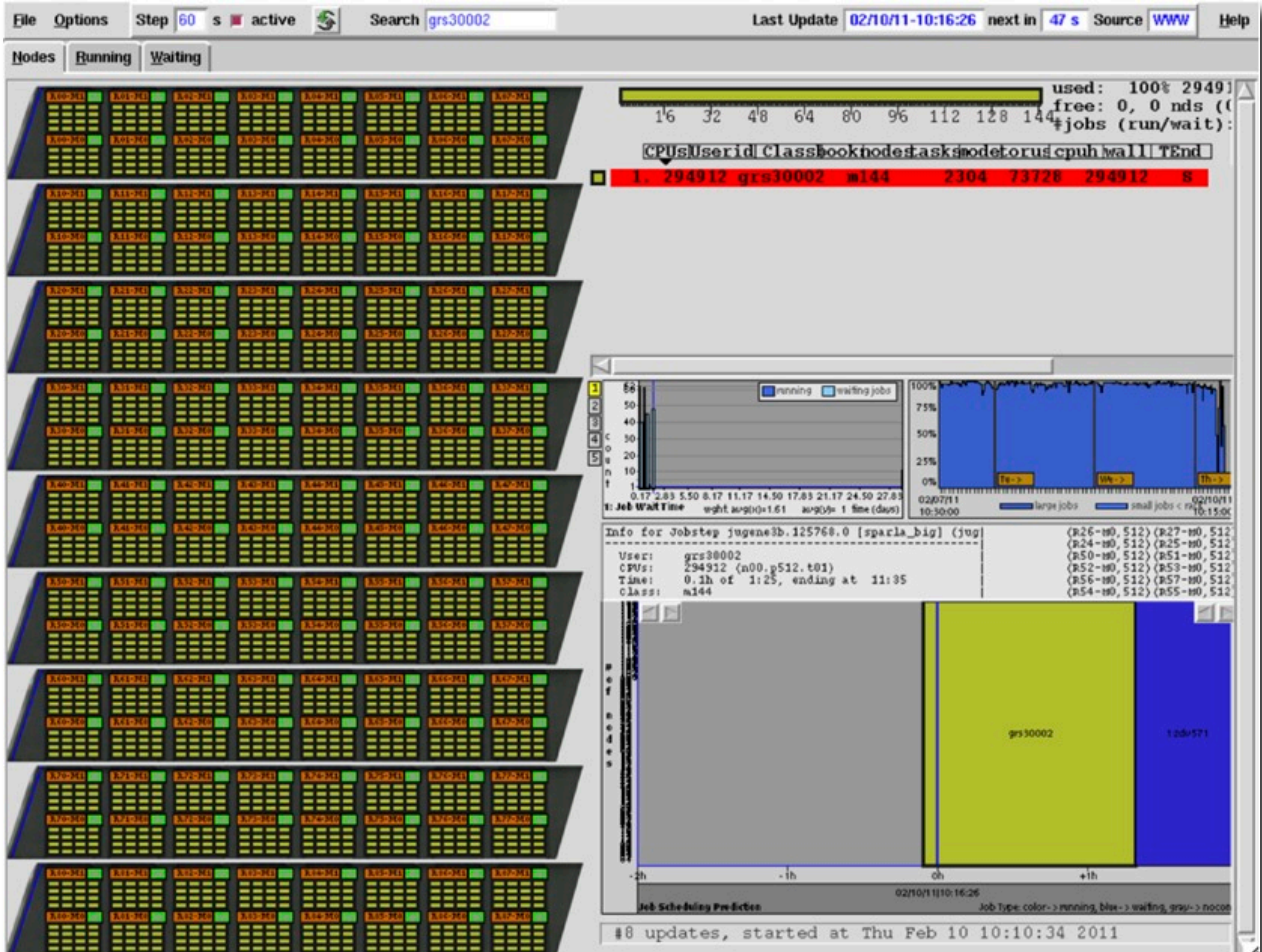
Implementation on IBM BlueGene/P



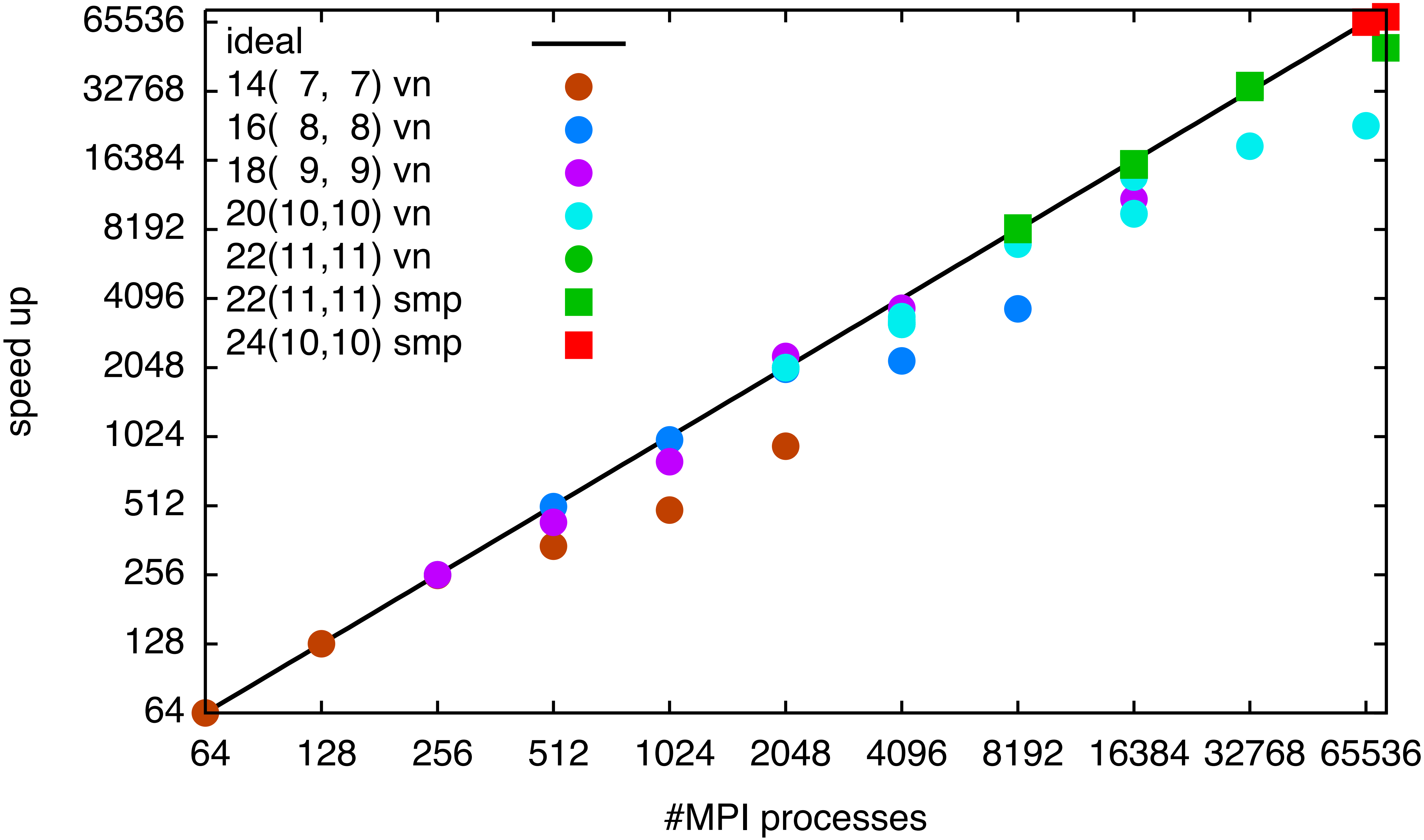
sites	memory
16	1 GB
18	18 GB
20	254 GB

Adv. Parallel Computing **15**, 601 (2008)

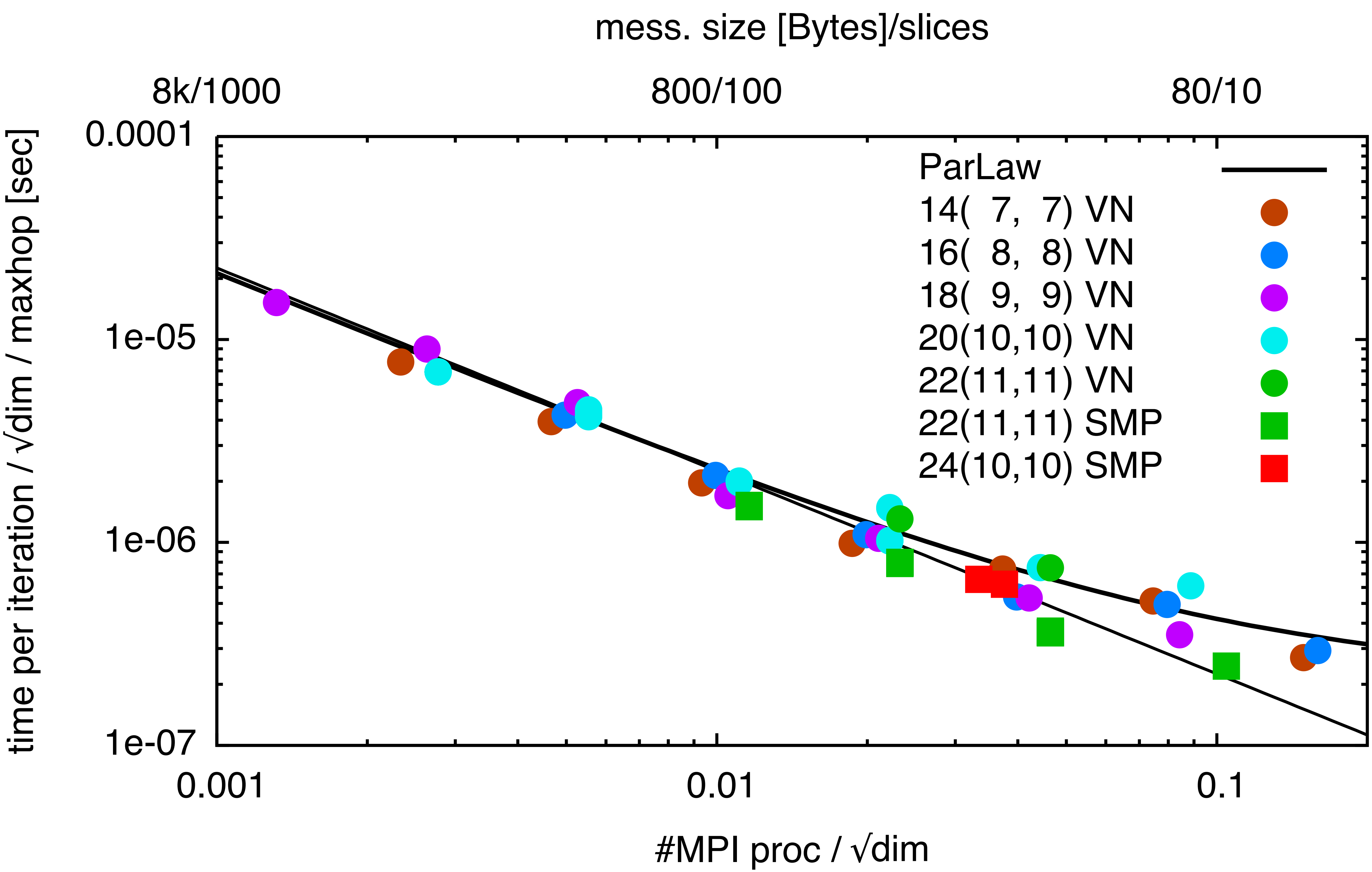
performance on full Jugene?



performance on full Jugene!



performance on full Jugene!

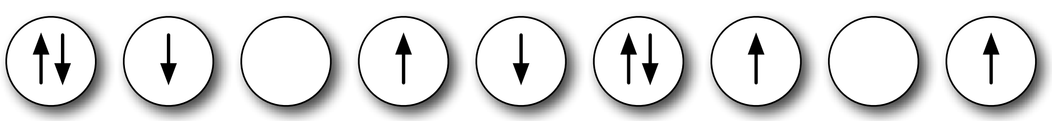


DMFT and optimal bath-parametrization

reminder: single-site DMFT

Hubbard model

$$H = -\sum_{ij\sigma} t_{ij}\, c^\dagger_{i\sigma} c_{j\sigma} + U \sum_i n_{i\uparrow} n_{i\downarrow}$$

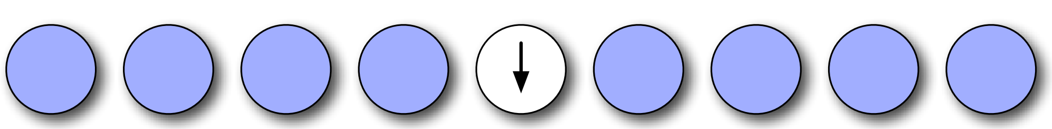


Bloch: e^{-ik} 1 e^{ik} e^{2ik} e^{3ik} e^{4ik}

$$c^\dagger_{k\sigma} = \sum e^{ikr_i} c^\dagger_{i\sigma} \Rightarrow H(\mathbf{k}) = \epsilon(\mathbf{k})$$

project to single site: $\int d\mathbf{k}\, H(\mathbf{k}) = \epsilon_0$

$$H_{\text{loc}} = \epsilon_0 + U\, n_\uparrow n_\downarrow$$



$$G_{\text{loc}}(\omega) = \int d\mathbf{k} \left(\omega - \mu - \varepsilon(\mathbf{k}) - \Sigma(\omega)\right)^{-1}$$

$$G_b^{-1}(\omega) = \Sigma(\omega) + G_{\text{loc}}^{-1}(\omega)$$

$$G_b^{-1}(\omega) \approx \omega + \mu - \epsilon_0 - \sum_l \frac{|V_l|^2}{(\omega - \varepsilon_l)}$$

$$H_{\text{And}} = H_{\text{loc}} + \sum_{l\sigma} \varepsilon_{l\sigma}\, a^\dagger_{l\sigma} a_{l\sigma} + \sum_{li,\sigma} V_{li} \left(a^\dagger_{l\sigma} c_{i\sigma} + \text{H.c.}\right)$$

$$\Sigma(\omega) = G_b^{-1}(\omega) - G_{\text{imp}}^{-1}(\omega)$$

bath parametrization

$$G_b^{-1}(\omega) = G_{\text{loc}}^{-1}(\omega) + \Sigma(\omega) = \omega + \mu - \int_{-\infty}^{\infty} d\omega' \, \frac{\Delta(\omega')}{\omega - \omega'}$$

$$G_{\text{And}}^{-1}(\omega) = \omega + \mu - \sum_{l=1}^{N_b} \frac{V_l^2}{\omega - \varepsilon_l}$$

how to determine bath parameters ε_l and V_l?

$$H_{\text{And}}^0 = \begin{pmatrix} 0 & V_1 & V_2 & V_3 & \cdots \\ V_1 & \varepsilon_1 & 0 & 0 & \\ V_2 & 0 & \varepsilon_2 & 0 & \\ V_3 & 0 & 0 & \varepsilon_3 & \\ \vdots & & & & \ddots \end{pmatrix}$$

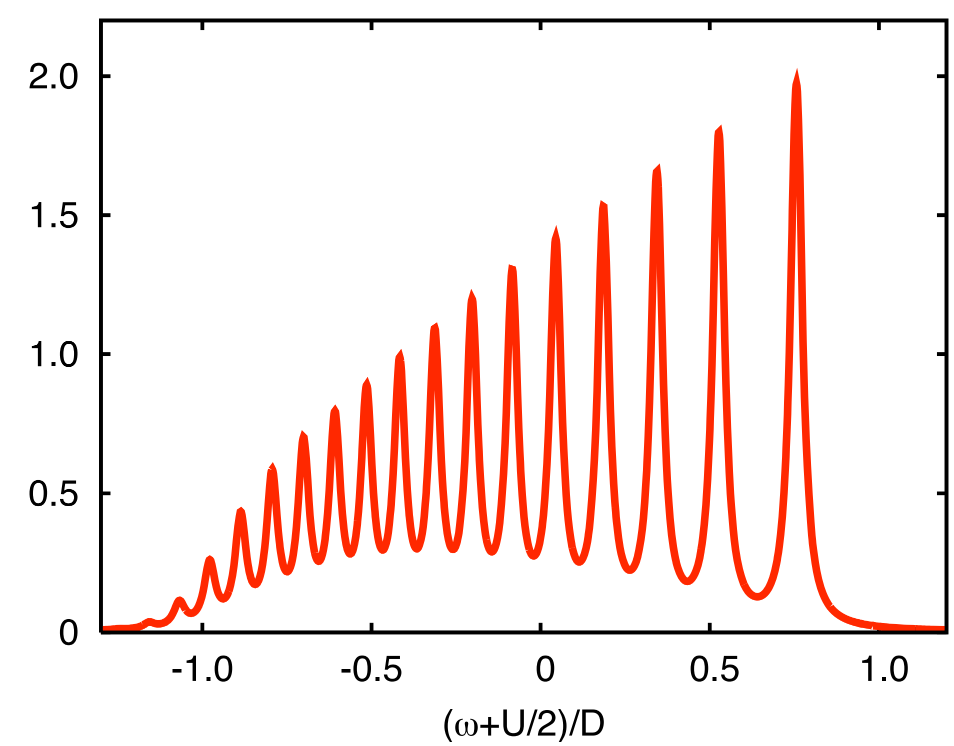
$$H_{\text{And}} = \varepsilon_0 \sum_\sigma n_\sigma + U n_\uparrow n_\downarrow + \sum_\sigma \sum_{l=1}^{N_b} \left(\varepsilon_l n_{l\sigma} + V_l \left(a_{l\sigma}^\dagger c_\sigma + c_\sigma^\dagger a_{l\sigma} \right) \right)$$

use Lanczos parameters

Bethe lattice: $$\int d\omega' \frac{\Delta(\omega')}{\omega - \omega'} = t^2 G_{\mathrm{imp}}(\omega)$$

$$t^2 G^{<}(\omega) + t^2 G^{>}(\omega) = \frac{t^2 {b_0^{<}}^2}{\omega + a_0^{<} - \frac{{b_1^{<}}^2}{\omega + a_1^{<} - \cdots}} + \frac{t^2 {b_0^{>}}^2}{\omega - a_0^{>} - \frac{{b_1^{>}}^2}{\omega - a_1^{>} - \cdots}}$$

$$H^0_{\mathrm{And}} = \begin{pmatrix} 0 & t^2 b_0^{<} & & & \cdots & t^2 b_0^{>} & & & \\ t^2 b_0^{<} & -a_0^{<} & b_1^{<} & & & & & & \\ & b_1^{<} & -a_1^{<} & b_2^{<} & & & & & \\ & & b_2^{<} & -a_2^{<} & \ddots & & & & \\ \vdots & & & \ddots & \ddots & & & & \\ t^2 b_0^{>} & & & & & a_0^{>} & b_1^{>} & & \\ & & & & & b_1^{>} & a_1^{>} & b_2^{>} & \\ & & & & & & b_2^{>} & a_2^{>} & \ddots \\ & & & & & & & \ddots & \end{pmatrix}$$



fit on imaginary axis

fictitious temperature: Matsubara frequencies

$$\chi^2(\{V_l, \varepsilon_l\}) = \sum_{n=0}^{n_{\text{max}}} w(i\omega_n) \left|\mathcal{G}^{-1}(i\omega_n) - \mathcal{G}_{\text{And}}^{-1}(i\omega_n)\right|^2$$

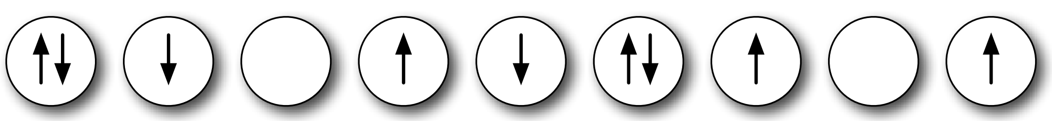
weight function $w(i\omega_n)$:

- emphasize region close to real axis
- make sum converge for $n \rightarrow \infty$ (sum rule)

reminder: single-site DMFT

Hubbard model

$$H = -\sum_{ij\sigma} t_{ij}\, c^{\dagger}_{i\sigma} c_{j\sigma} + U \sum_{i} n_{i\uparrow} n_{i\downarrow}$$

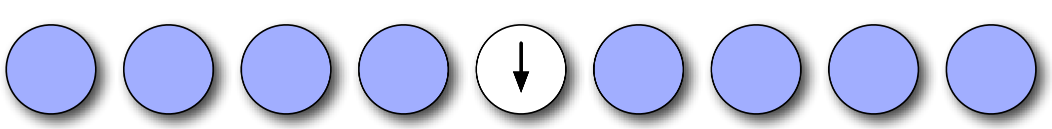


Bloch: e^{-ik} 1 e^{ik} e^{2ik} e^{3ik} e^{4ik}

$$c^{\dagger}_{k\sigma} = \sum e^{ikr_i} c^{\dagger}_{i\sigma} \Rightarrow H(\mathbf{k}) = \epsilon(\mathbf{k})$$

project to single site: $\int d\mathbf{k}\, H(\mathbf{k}) = \epsilon_0$

$$H_{\text{loc}} = \epsilon_0 + U\, n_{\uparrow} n_{\downarrow}$$



$$G_{\text{loc}}(\omega) = \int d\mathbf{k} \left(\omega - \mu - \varepsilon(\mathbf{k}) - \Sigma(\omega)\right)^{-1}$$

$$G_b^{-1}(\omega) = \Sigma(\omega) + G_{\text{loc}}^{-1}(\omega)$$

$$G_b^{-1}(\omega) \approx \omega + \mu - \epsilon_0 - \sum_l \frac{|V_l|^2}{(\omega - \varepsilon_l)}$$

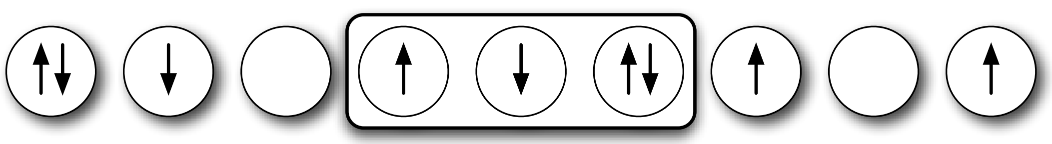
$$H_{\text{And}} = H_{\text{loc}} + \sum_{l\sigma} \varepsilon_{l\sigma}\, a^{\dagger}_{l\sigma} a_{l\sigma} + \sum_{li,\sigma} V_{li} \left(a^{\dagger}_{l\sigma} c_{i\sigma} + \text{H.c.}\right)$$

$$\Sigma(\omega) = G_b^{-1}(\omega) - G_{\text{imp}}^{-1}(\omega)$$

DMFT for clusters

Hubbard model

$$H = -\sum_{ij\sigma} t_{ij}\, c^\dagger_{i\sigma} c_{j\sigma} + U \sum_i n_{i\uparrow} n_{i\downarrow}$$

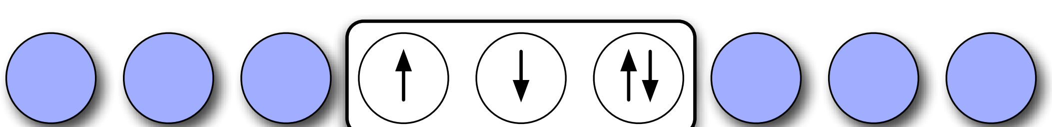


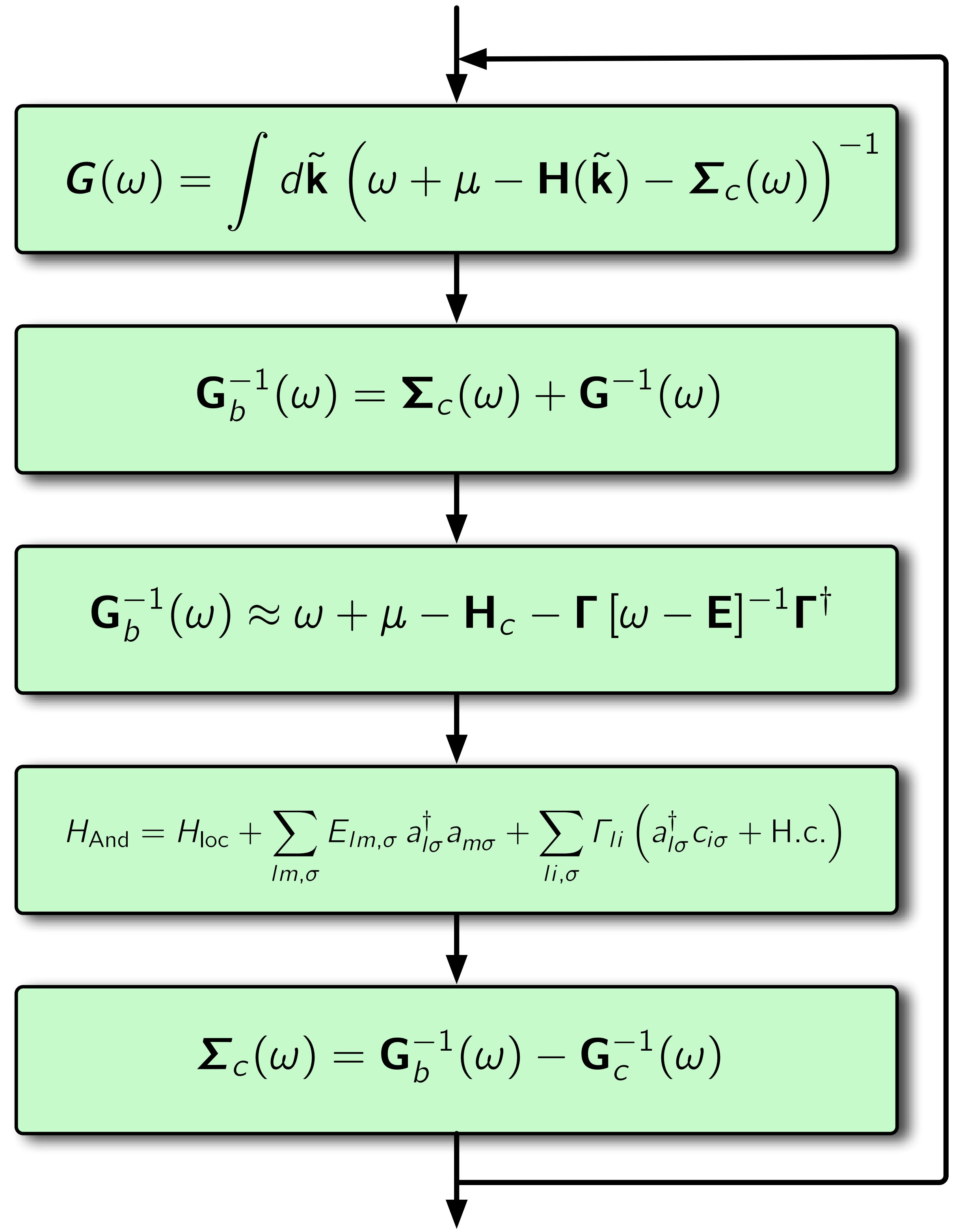
Bloch: e^{-ik} 1 e^{ik} e^{2ik} e^{3ik} e^{4ik}

$$c^\dagger_{\tilde{k}\sigma} = \sum e^{i\tilde{k}r_i} c^\dagger_{i\sigma} \Rightarrow \mathbf{H}(\tilde{\mathbf{k}})$$

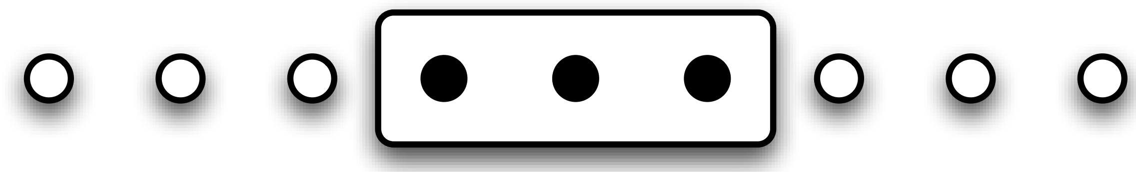
project to cluster: $\int d\tilde{\mathbf{k}}\, \mathbf{H}(\tilde{\mathbf{k}}) = \mathbf{H}_c$

$$H_{\text{loc}} = \mathbf{H}_c + U \sum_i n_{i\uparrow} n_{i\downarrow}$$





DCA 3-site cluster



$$\mathbf{H}(\tilde{k}) = -t \begin{pmatrix} 0 & e^{i\tilde{k}} & e^{-i\tilde{k}} \\ e^{-i\tilde{k}} & 0 & e^{i\tilde{k}} \\ e^{i\tilde{k}} & e^{-i\tilde{k}} & 0 \end{pmatrix}$$

$$\mathbf{H}_c = \frac{3}{2\pi}\int_{-\pi/3}^{\pi/3} d\tilde{k}\,\mathbf{H}(\tilde{k}) = -\frac{3\sqrt{3}}{2\pi}\,t \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

translation symmetry
coarse-grained Hamiltonian

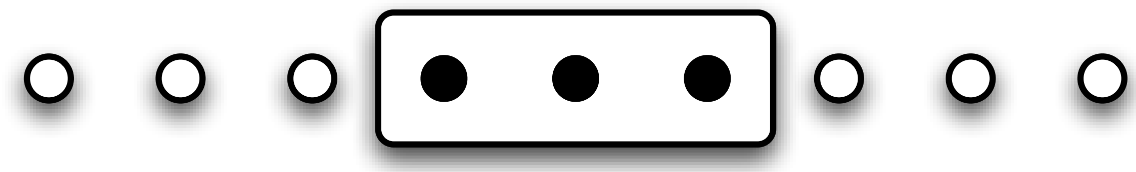
DCA

$$\mathbf{H}(\tilde{k}) = -t \begin{pmatrix} 0 & e^{i\tilde{k}} & e^{-i\tilde{k}} \\ e^{-i\tilde{k}} & 0 & e^{i\tilde{k}} \\ e^{i\tilde{k}} & e^{-i\tilde{k}} & 0 \end{pmatrix}$$

$$\mathbf{H}_c = \frac{3}{2\pi} \int_{-\pi/3}^{\pi/3} d\tilde{k}\, \mathbf{H}(\tilde{k}) = -\frac{3\sqrt{3}}{2\pi}\, t \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

translation symmetry
coarse-grained Hamiltonian

CDMFT

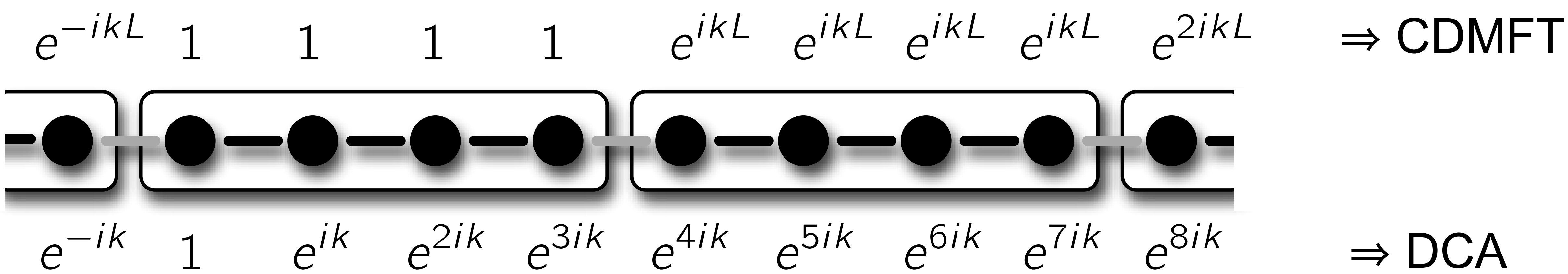


no translation symmetry
original Hamiltonian on cluster

$$\mathbf{H}(\tilde{k}) = -t \begin{pmatrix} 0 & 1 & e^{-3i\tilde{k}} \\ 1 & 0 & 1 \\ e^{3i\tilde{k}} & 1 & 0 \end{pmatrix}$$

$$\mathbf{H}_c = \frac{3}{2\pi} \int_{-\pi/3}^{\pi/3} d\tilde{k}\, \mathbf{H}(\tilde{k}) = -t \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

DCA – CDMFT



$$\tilde{c}^{\mathrm{CDMFT}}_{\mathrm{R}_i\sigma}(\tilde{\mathbf{k}}) = \sum_{\tilde{\mathrm{r}}} e^{-i\tilde{\mathrm{k}}\tilde{\mathrm{r}}}\, c_{\tilde{\mathrm{r}}+\mathrm{R}_i,\sigma}$$

$$\tilde{c}^{\mathrm{DCA}}_{\mathrm{R}_i\sigma}(\tilde{\mathbf{k}}) = \sum_{\tilde{\mathrm{r}}} e^{-i\tilde{\mathrm{k}}(\tilde{\mathrm{r}}+\mathrm{R}_i)}\, c_{\tilde{\mathrm{r}}+\mathrm{R}_i,\sigma}$$

gauge determines cluster method:

$$\tilde{c}_{\mathrm{R}_i\sigma}(\tilde{\mathbf{k}}) = \sum_{\tilde{\mathrm{r}}} e^{-i(\tilde{\mathrm{k}}\tilde{\mathrm{r}}+\varphi(\tilde{\mathrm{k}};\mathrm{R}_i))}\, c_{\tilde{\mathrm{r}}+\mathrm{R}_i,\sigma}$$

bath for cluster

$$H_{\text{And}} = H_{\text{clu}} + \sum_{lm,\sigma} E_{lm,\sigma}\, a^\dagger_{l\sigma} a_{m\sigma} + \sum_{li,\sigma} \Gamma_{li} \left(a^\dagger_{l\sigma} c_{i\sigma} + \text{H.c.} \right)$$

diagonalize bath: $\mathbf{E}\,\phi_l = \varepsilon_l \phi_l$ and define $V_{l,i} = \sum_m \Gamma_{i,m}\, \phi_{l,m}$

$$\mathbf{G}_b^{-1}(\omega) \approx \omega + \mu - \mathbf{H}_c - \sum_l \frac{\mathbf{V}_l \mathbf{V}_l^\dagger}{\omega - \varepsilon_l}$$

$$\mathbf{G}_b^{-1}(\omega) = \mathbf{\Sigma}_c(\omega) + \left(\int d\tilde{\mathbf{k}} \left(\omega + \mu - \mathbf{H}(\tilde{\mathbf{k}}) - \mathbf{\Sigma}_c(\omega) \right)^{-1} \right)^{-1}$$

expand up to $1/\omega^2$: **sum-rule**

$$\sum_l \mathbf{V}_l \mathbf{V}_l^\dagger = \int d\tilde{\mathbf{k}}\, \mathbf{H}^2(\tilde{\mathbf{k}}) - \left(\int d\tilde{\mathbf{k}}\, \mathbf{H}(\tilde{\mathbf{k}}) \right)^2$$

hybridization sum-rules: single-site

H with hopping t_n to the z_n n^{th}-nearest neighbors

$$\sum_l V_l^2 = \frac{1}{(2\pi)^d} \int_{-\pi}^{\pi} d^d\mathbf{k}\, \varepsilon_{\mathrm{k}}^2 = \sum_n z_n\, t_n^2$$

special case: Bethe lattice of coordination z with hopping $t/\sqrt{z}$

$$\sum_l V_l^2 = t^2$$

hybridization sum-rules: DCA

hybridizations diagonal in the cluster-momenta **K**:

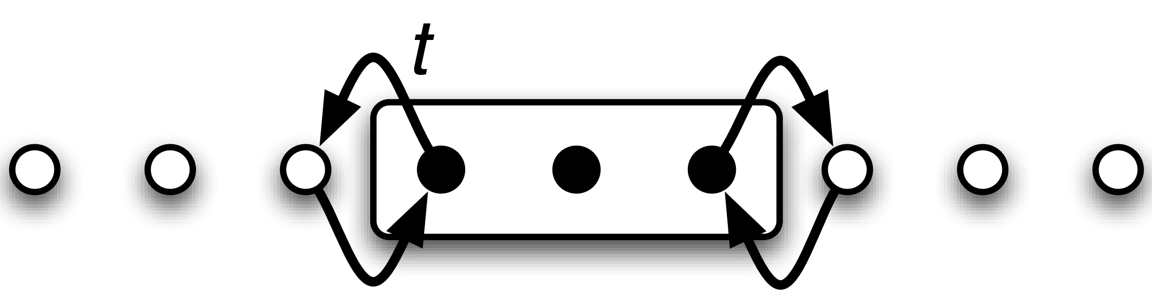
$$\sum_l |V_{l,\mathrm{K}}|^2 = \int d\tilde{\mathbf{k}}\, \varepsilon^2_{\mathrm{K}+\tilde{\mathrm{k}}} - \left(\int d\tilde{\mathbf{k}}\, \varepsilon_{\mathrm{K}+\tilde{\mathrm{k}}} \right)^2$$

all terms $V_{l,K}\, V_{l,K'}$ mixing different cluster momenta vanish

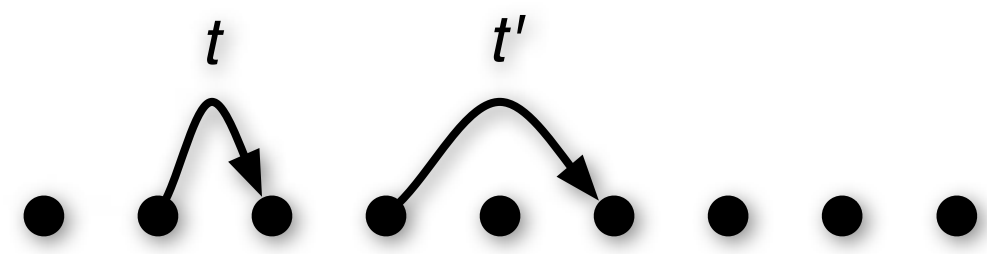
hybridization sum-rules: CDMFT

$$\mathbf{H}(\tilde{k}) = -t \begin{pmatrix} 0 & 1 & e^{-3i\tilde{k}} \\ 1 & 0 & 1 \\ e^{3i\tilde{k}} & 1 & 0 \end{pmatrix}$$

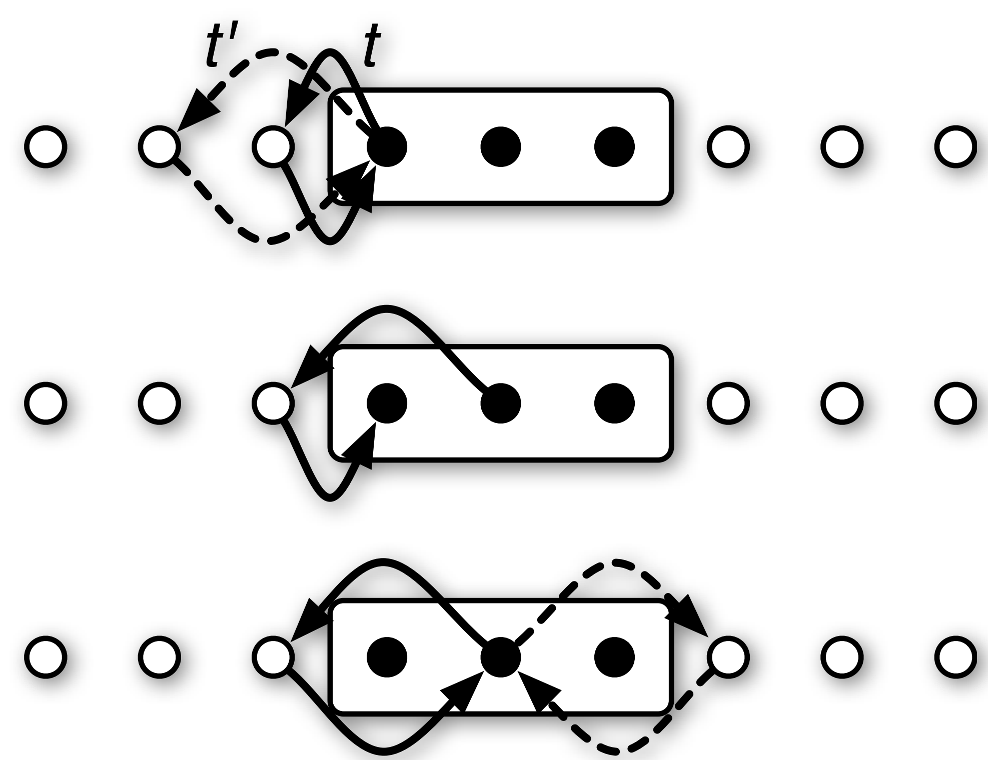
$$\sum_l \mathbf{V}_l \mathbf{V}_l^\dagger = \int d\tilde{\mathbf{k}}\, \mathbf{H}^2(\tilde{\mathbf{k}}) - \left(\int d\tilde{\mathbf{k}}\, \mathbf{H}(\tilde{\mathbf{k}}) \right)^2 = \begin{pmatrix} t^2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & t^2 \end{pmatrix}$$



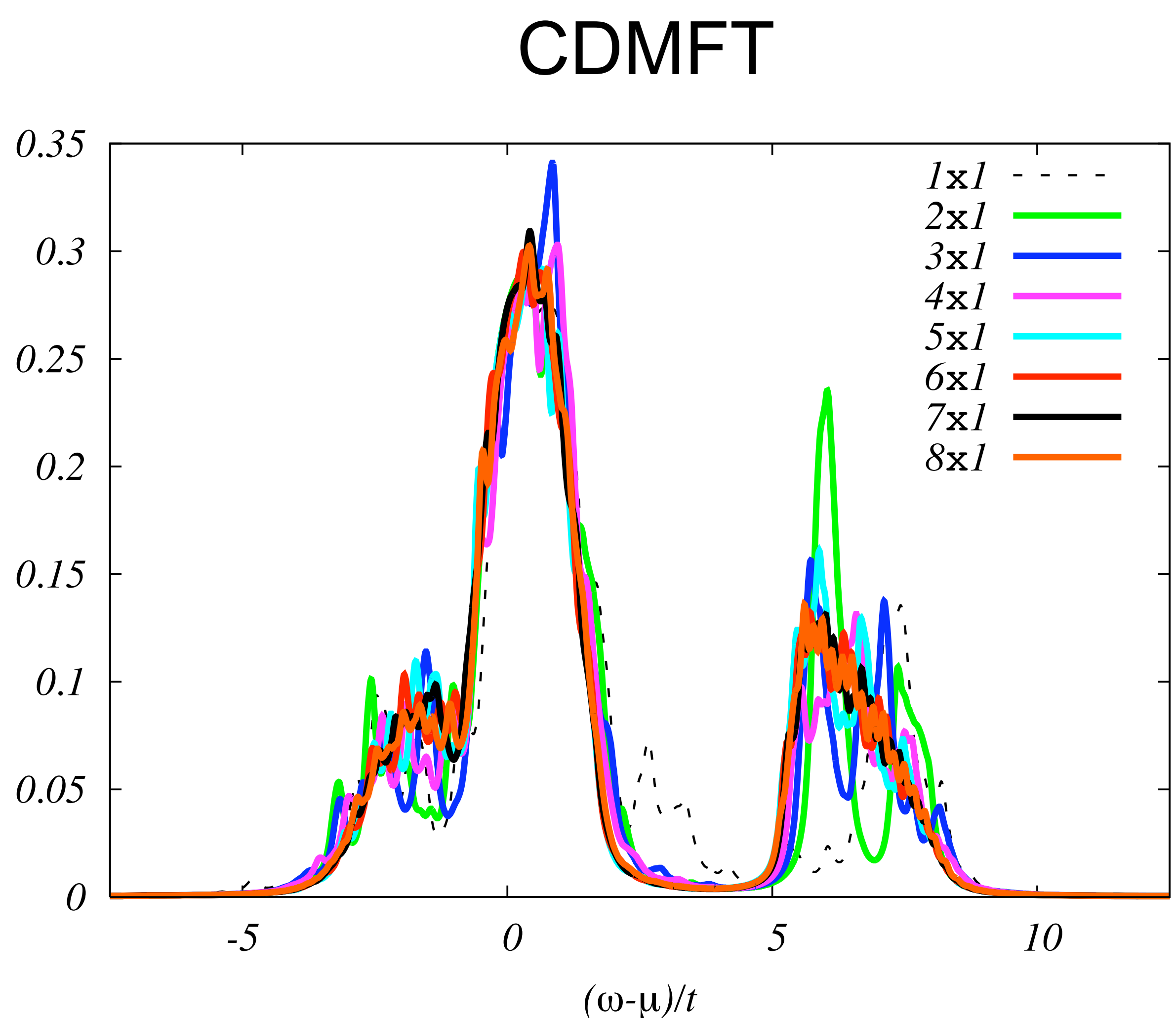
hybridization sum-rules: CDMFT

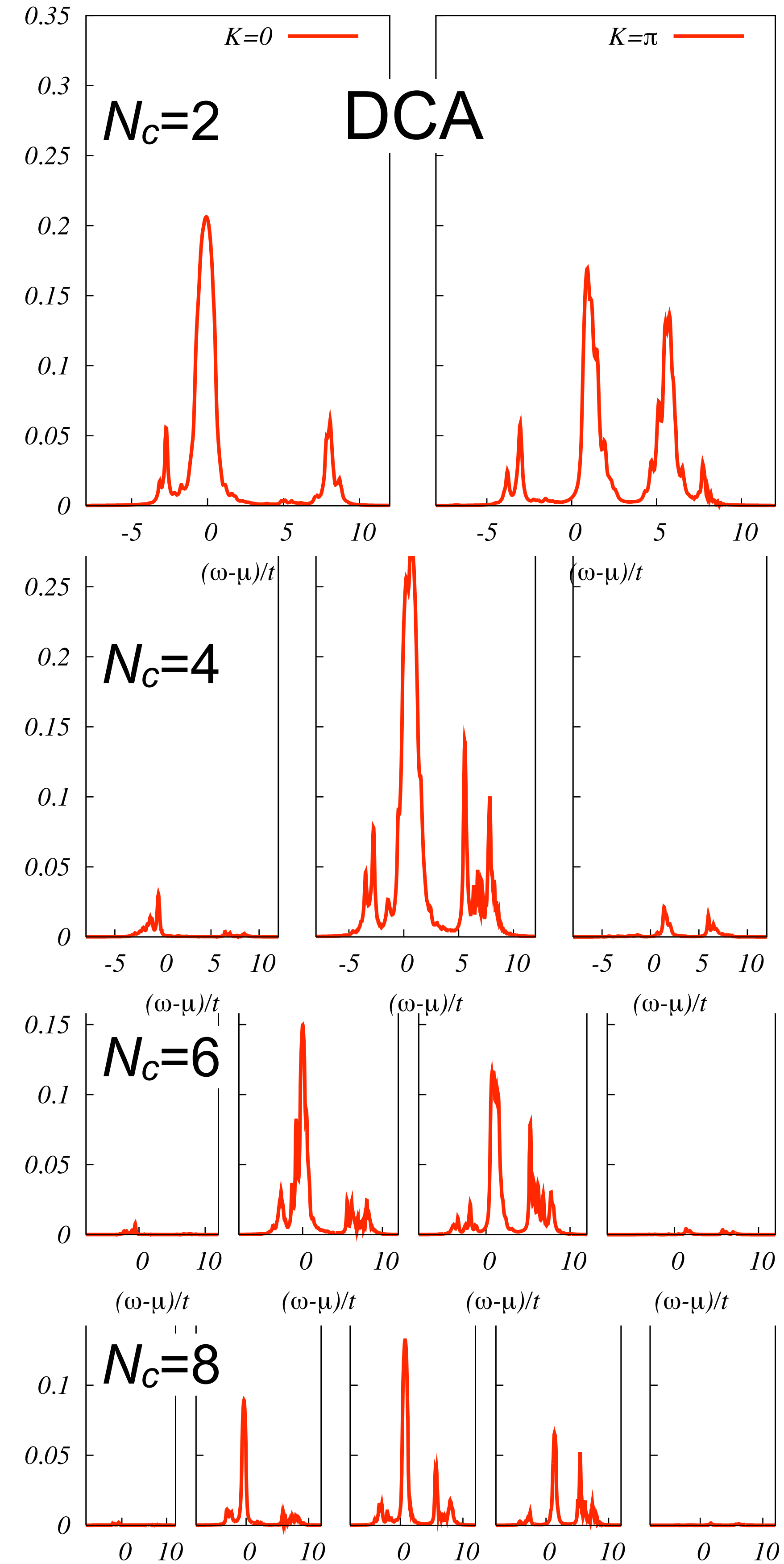


$$\left(\sum_l V_{l,i} \bar{V}_{l,j} \right) = \begin{pmatrix} t^2 + t'^2 & t\,t' & 0 \\ t\,t' & 2t'^2 & t\,t' \\ 0 & t\,t' & t^2 + t'^2 \end{pmatrix}$$



example: 1-d clusters





	CDMFT	DCA
hybridize	only surface	full cluster
strength	const.	$1/N_c^{2/d}$

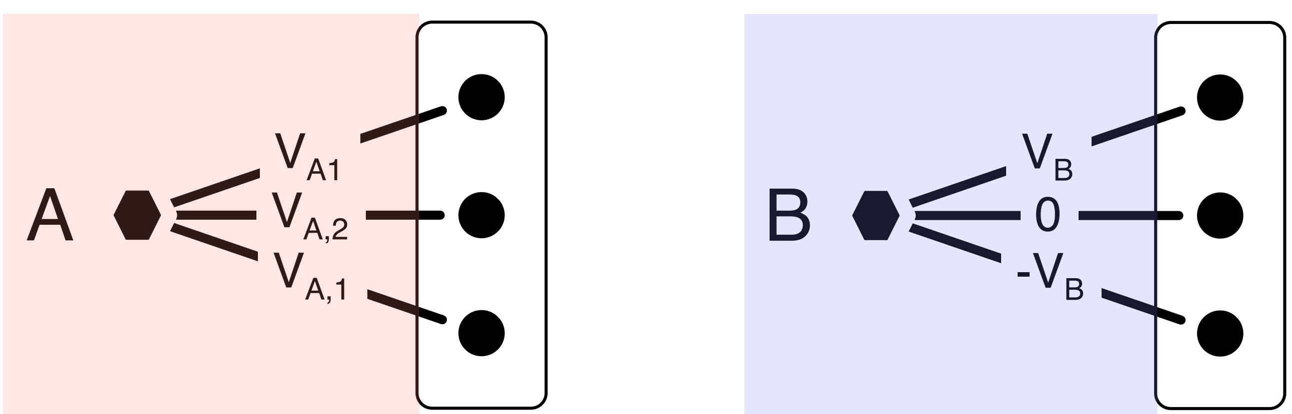
symmetry of bath

$$\mathbf{W} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 1 \\ 0 & \sqrt{2} & 0 \\ 1 & 0 & -1 \end{pmatrix}$$

irreducible representations: A (even), B (odd)

$$\mathbf{W}^\dagger \mathbf{G}_b^{-1} \mathbf{W} = \begin{pmatrix} G_{b,11}^{-1} + G_{b,13}^{-1} & \sqrt{2}\, G_{b,12}^{-1} & 0 \\ \sqrt{2}\, G_{b,21}^{-1} & G_{b,22}^{-1} & 0 \\ 0 & 0 & G_{b,11}^{-1} - G_{b,13}^{-1} \end{pmatrix}$$

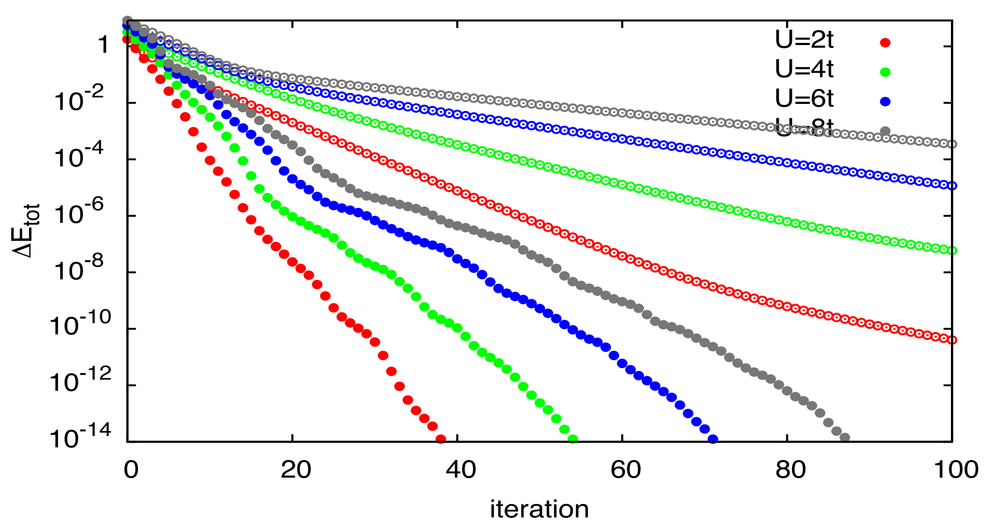
block-diagonal



summary

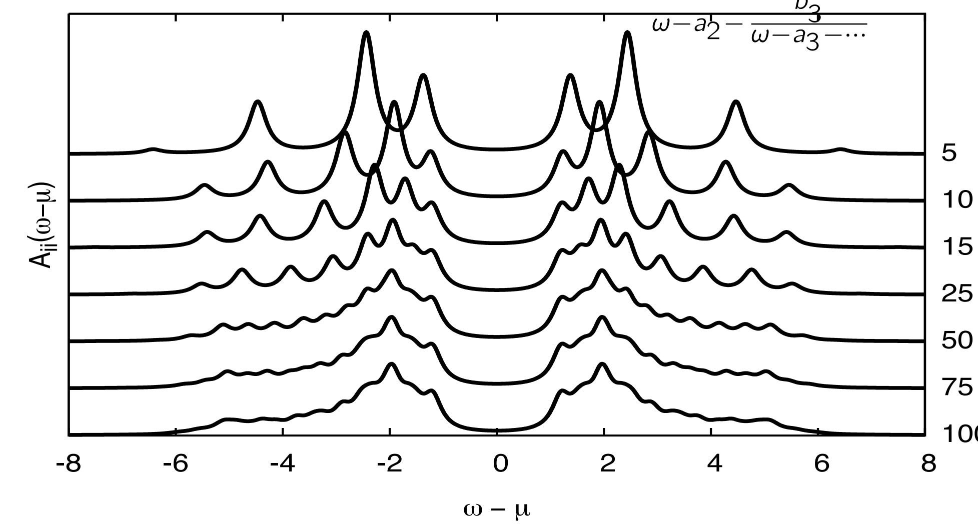
steepest descent ⇒ Krylov space

$$\frac{\delta E[\Psi]}{\delta\langle\Psi|} = \frac{H|\Psi\rangle - E[\Psi]|\Psi\rangle}{\langle\Psi|\Psi\rangle} = |\Psi_a\rangle \in \mathrm{span}\left(|\Psi\rangle, H|\Psi\rangle\right)$$

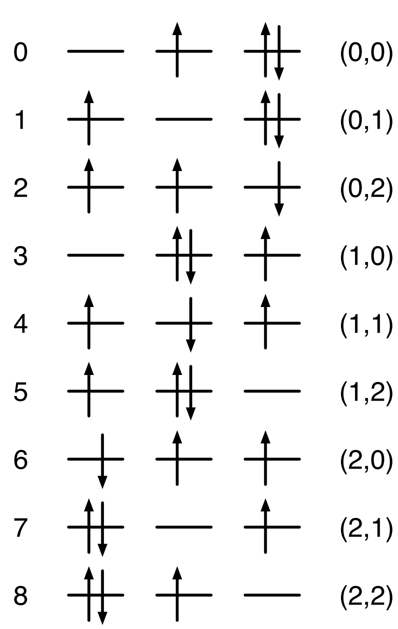


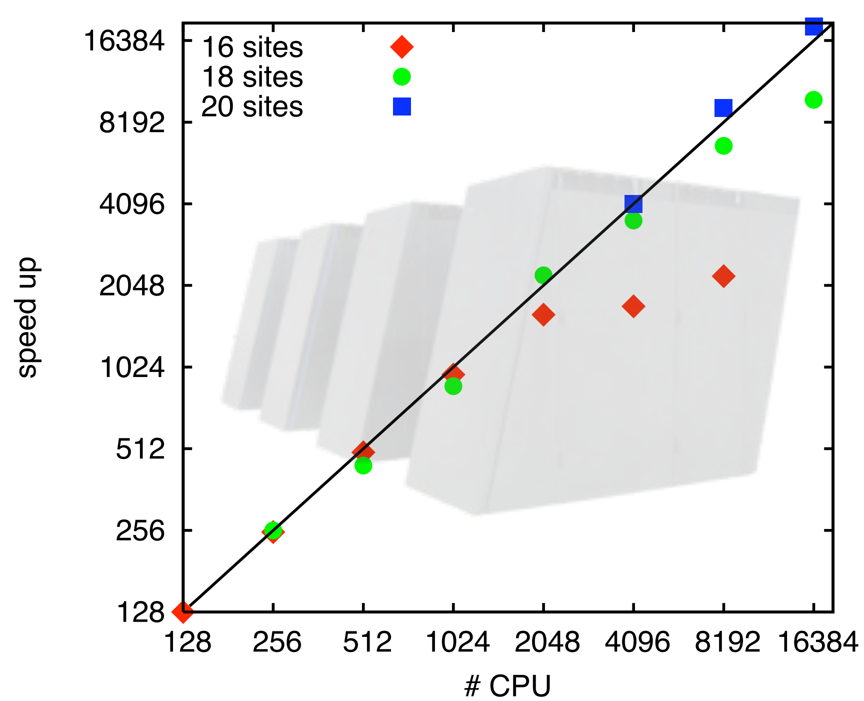
spectral function: moments

$$G_k(\omega) = \cfrac{b_0^2}{\omega - a_0 - \cfrac{b_1^2}{\omega - a_1 - \cfrac{b_2^2}{\omega - a_2 - \cfrac{b_3^2}{\omega - a_3 - \cdots}}}}$$

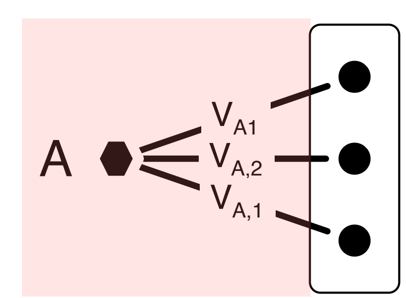


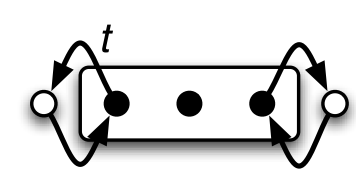
sparse Hamiltonian in Wannier representation

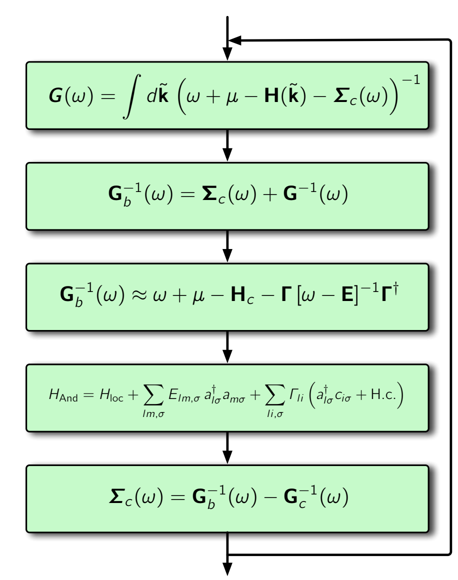




bath parametrization







$$\sum_l \mathbf{V}_l \mathbf{V}_l^\dagger = \int d\tilde{\mathbf{k}}\, \mathbf{H}^2(\tilde{\mathbf{k}}) - \left(\int d\tilde{\mathbf{k}}\, \mathbf{H}(\tilde{\mathbf{k}})\right)^2$$