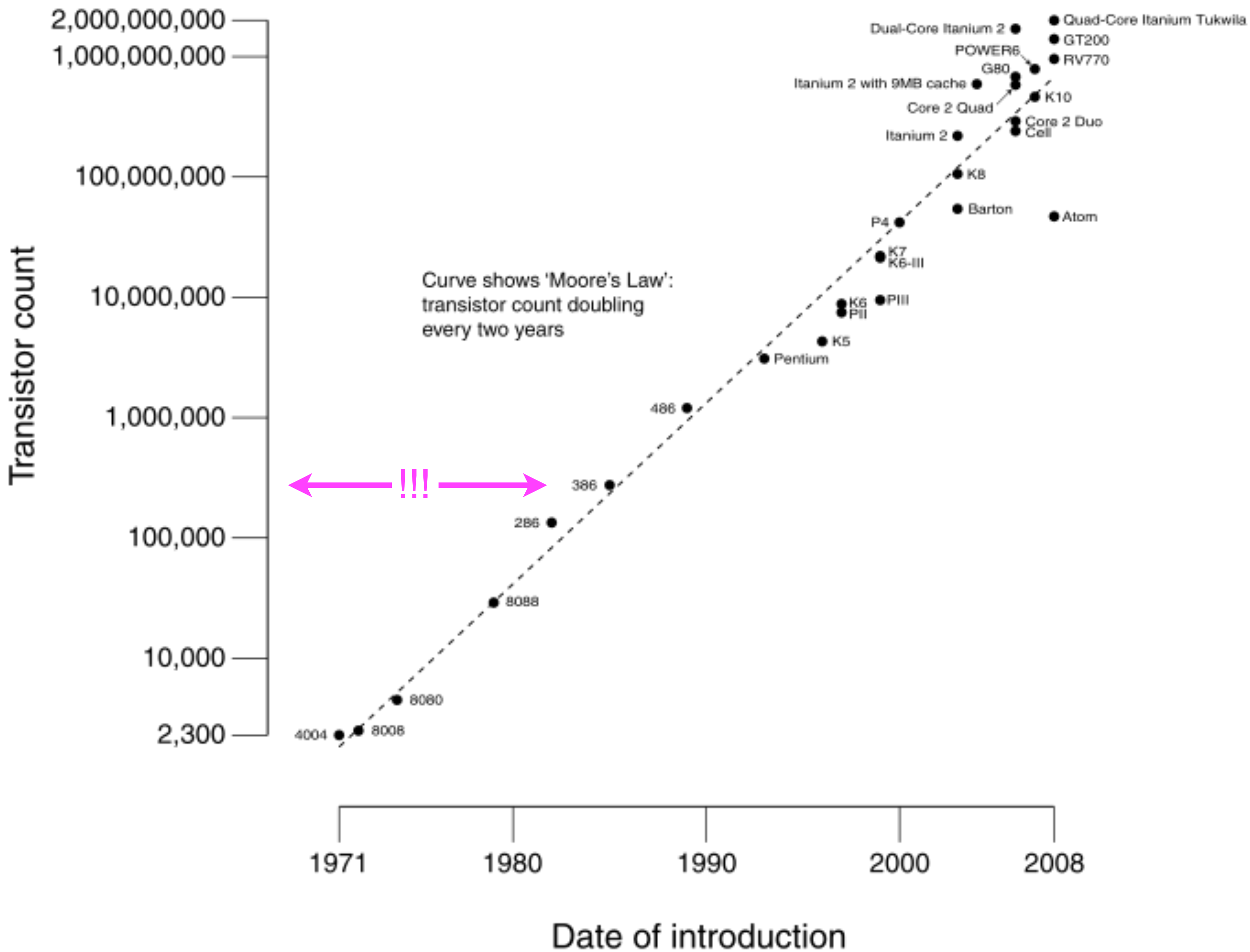
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

CSCS
Swiss National Supercomputing Centre

Electronic structure theory at the petascale and beyond

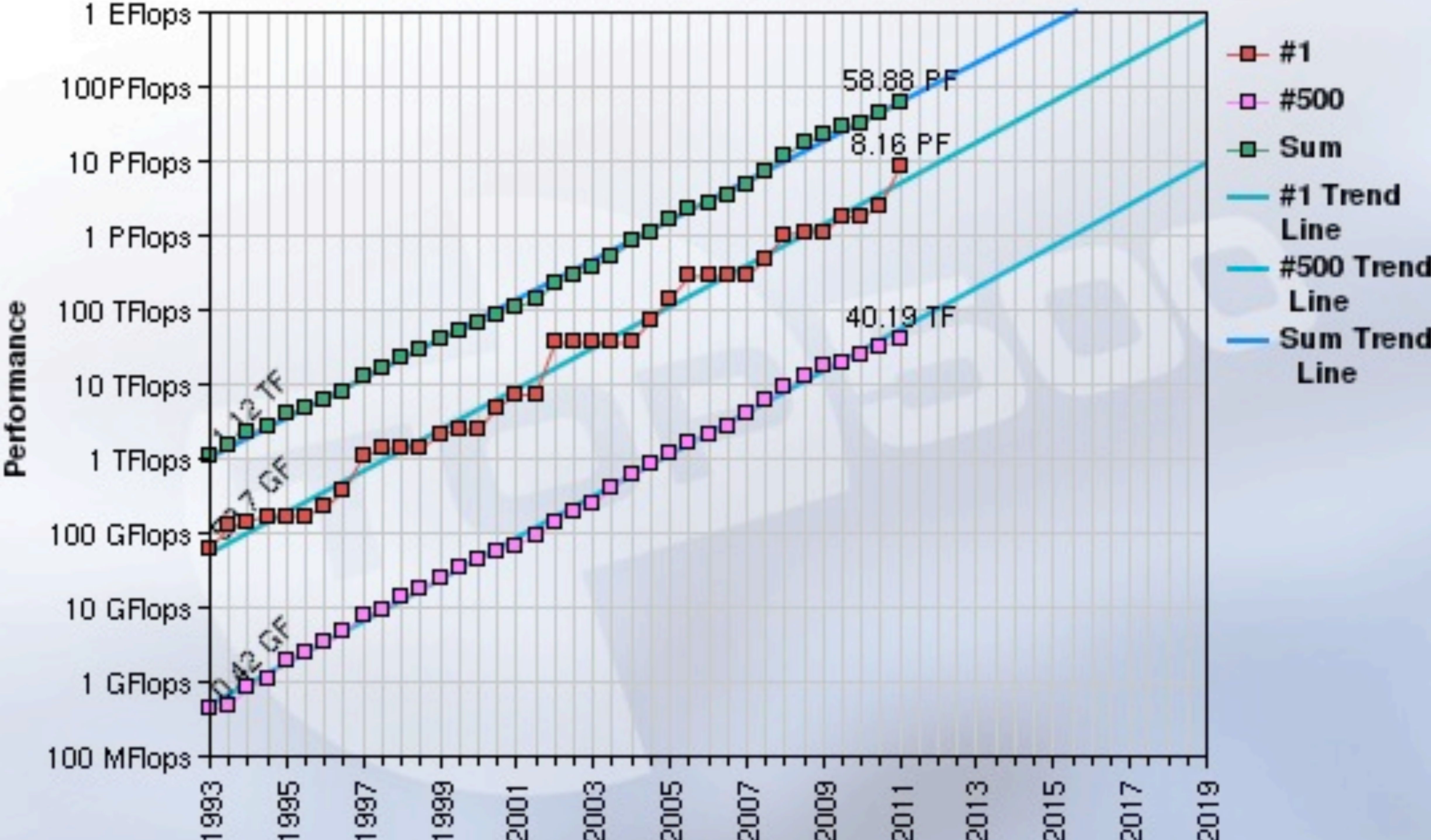
Thomas C. Schulthess

CPU Transistor Counts 1971-2008 & Moore's Law



Source: Wikipedia, the free encyclopedia

Projected Performance Development



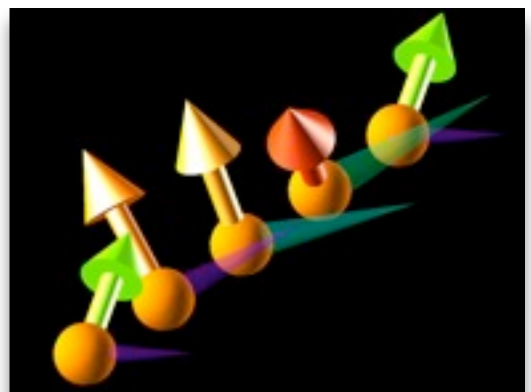
Computer performance and application performance increase ~10^3 every decade

~100 Kilowatts → ← ~5 Megawatts → ← 20-30 MW →

~1 Exaflop/s

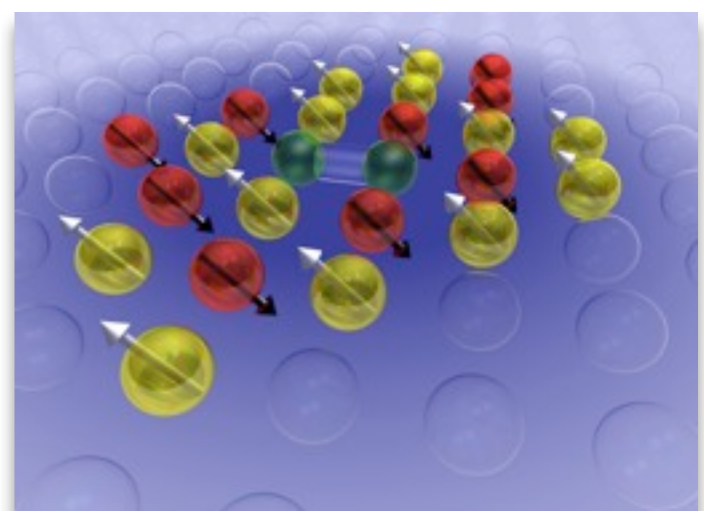
100 million or billion processing cores (!)

1 Gigaflop/s
Cray YMP
8 processors



1.02 Teraflop/s
Cray T_{3E}
1’500 processors

1.35 Petaflop/s
Cray XT5
150’000 processors



1988	1998	2008	2018
First sustained GFlop/s Gordon Bell Prize 1988	First sustained TFlop/s Gordon Bell Prize 1998	First sustained PFlop/s Gordon Bell Prize 2008	Another 1,000x increase in sustained performance

Applications running at scale on Jaguar @ ORNL (Spring 2011)

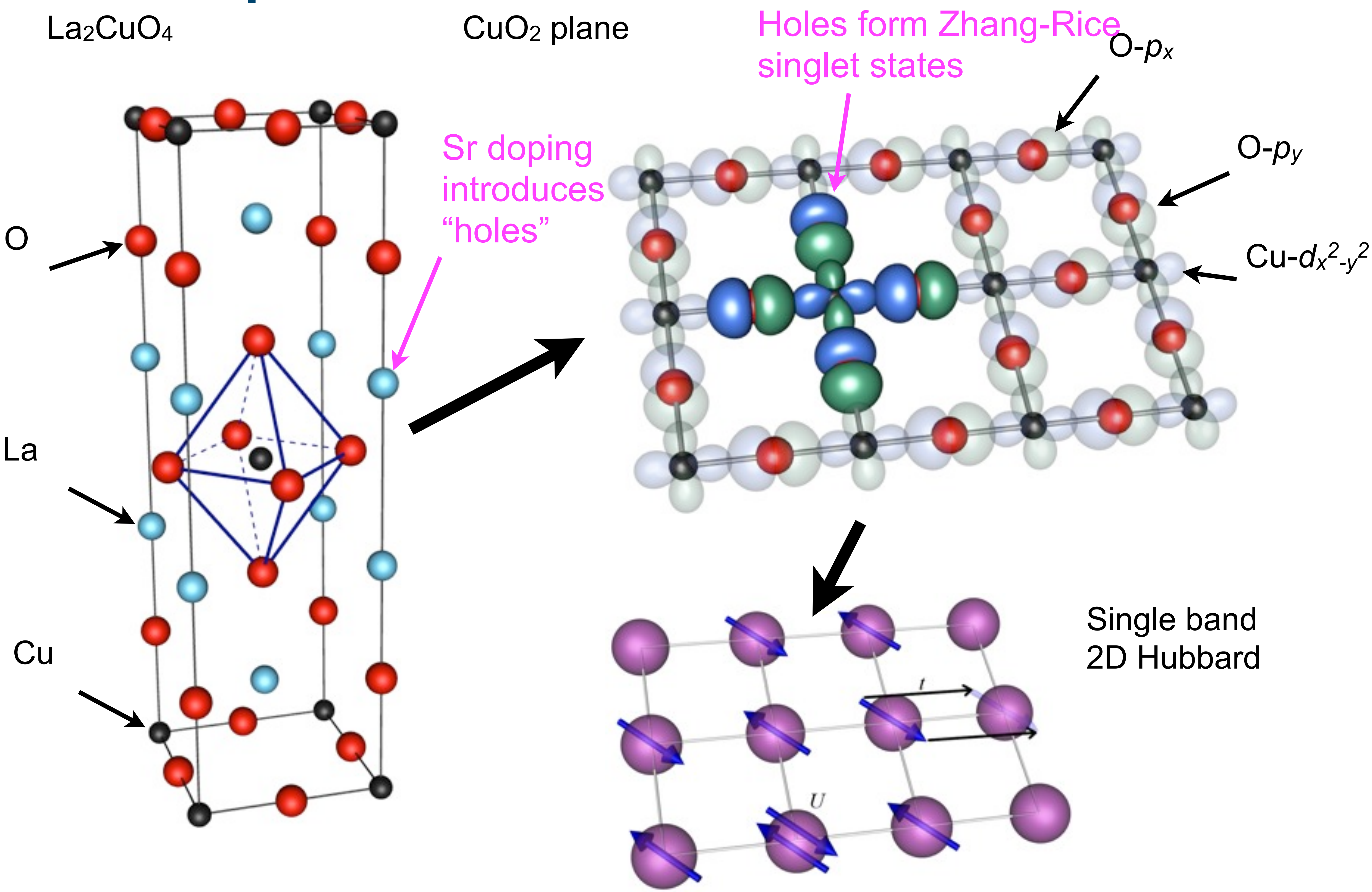
Domain area	Code name	Institution	# of cores	Performance	Notes
Materials	DCA++	ORNL	213,120	**1.9 PF**	2008 Gordon Bell Prize Winner
Materials	WL-LSMS	ORNL/ETH	223,232	**1.8 PF**	2009 Gordon Bell Prize Winner
Chemistry	NWChem	PNNL/ORNL	224,196	1.4 PF	2008 Gordon Bell Prize Finalist
Materials	DRC	ETH/UTK	186,624	1.3 PF	2010 Gordon Bell Prize Hon. Mention
Nanoscience	OMEN	Duke	222,720	> 1 PF	2010 Gordon Bell Prize Finalist
Biomedical	MoBo	GaTech	196,608	780 TF	2010 Gordon Bell Prize Winner
Chemistry	MADNESS	UT/ORNL	140,000	550 TF	
Materials	LS3DF	LBL	147,456	442 TF	2008 Gordon Bell Prize Winner
Seismology	SPECFEM3D	USA (multiple)	149,784	165 TF	2008 Gordon Bell Prize Finalist
Combustion	S3D	SNL	147,456	83 TF	
Weather	WRF	USA (multiple)	150,000	50 TF	

Outline

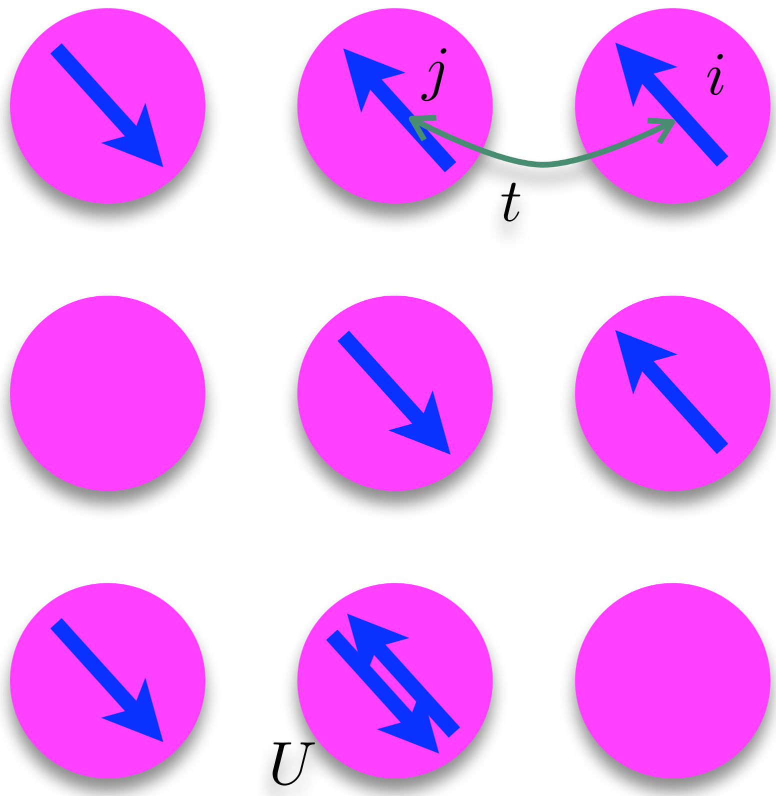
- Introduction – scale of supercomputing today
- Superconductivity and model of high T_c superconductors
  - Superconductivity and the 2D-Hubbard model
  - Quantum cluster theory & insights into the nature of superconductivity
  - DCA++ – algorithmic improvements, optimally mapping onto hardware
- A strategy to back out of the model
  - Screened Coulomb interaction within LAPW
  - Down-folded band structure and frequency dependent Hubbard U
- Conclusions
  - Recommendations for future code development
  - What the future will bring

This lecture is not just about what we can do with supercomputers – it will be mostly about how we map simulations on to computer systems

From cuprate materials to the Hubbard model

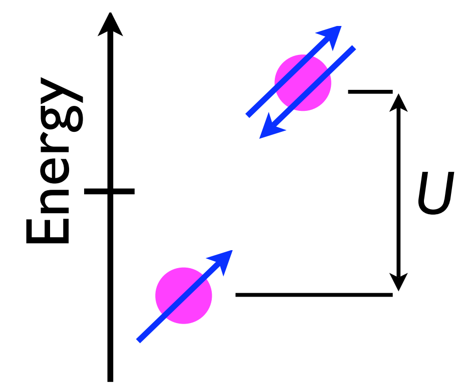


2D Hubbard model and its physics

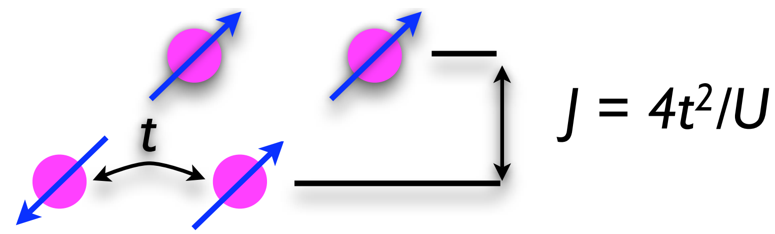


Half filling: number of carriers = number of sites

Formation of a **magnetic moment** when *U* is large enough

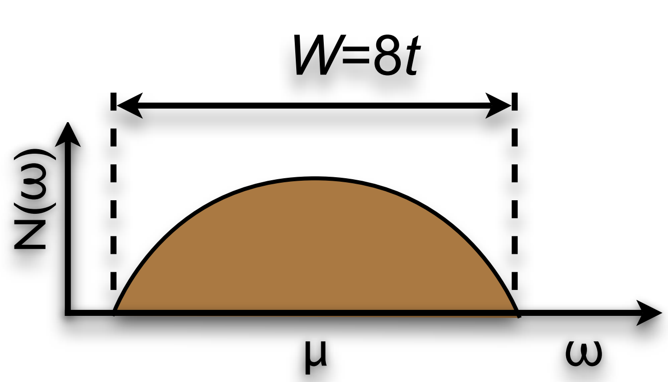


Antiferromagnetic alignment of neighboring moments



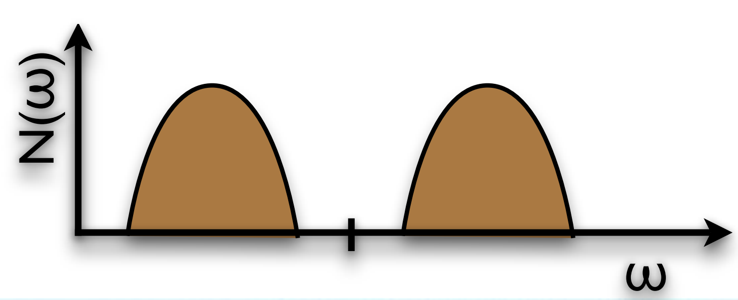
1. When *t* >> *U*:

Model describes a metal with band width W=8t

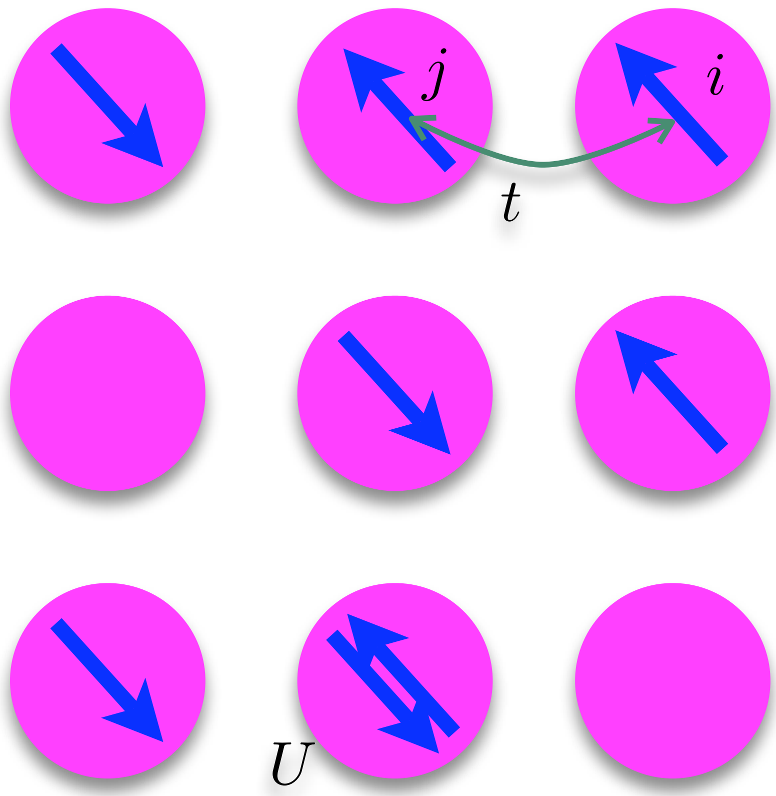


2. When *U* >> 8*t* at half filling (not doped)

Model describes a “Mott Insulator” with antiferromagnetic ground state (as seen experimentally seen in undoped cuprates)

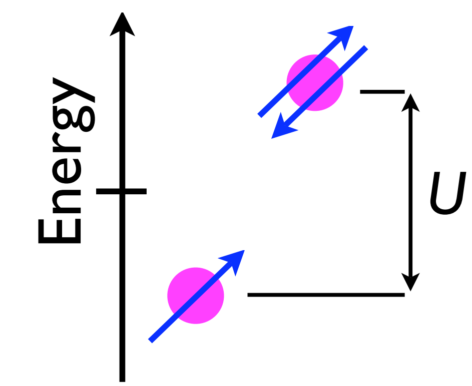


Hubbard model for the cuprates

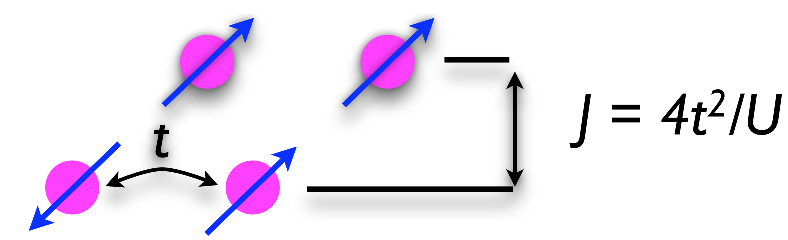


Half filling: number of carriers = number of sites

Formation of a **magnetic moment** when U is large enough



Antiferromagnetic alignment of neighboring moments



3. Parameter range relevant for superconducting cuprates

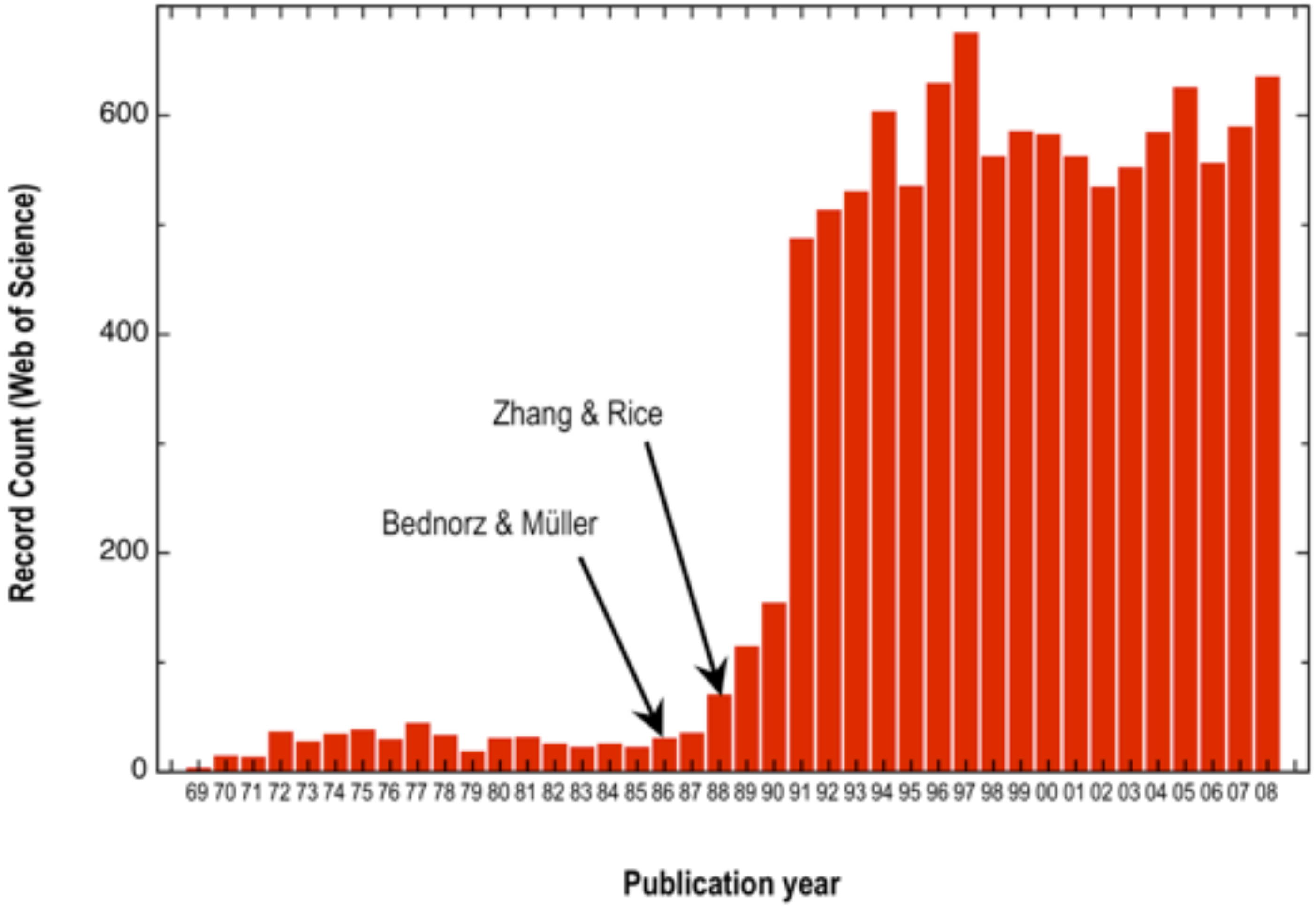
$U \approx 8t$

No simple solution!

Finite doping levels (0.05 – 0.25)

Typical values: U~10eV; t~0.9eV; J~0.2eV; (0.1eV ~ 10^3 Kelvin)

Hubbard model for the cuprates



3. Parameter range relevant for superconducting cuprates

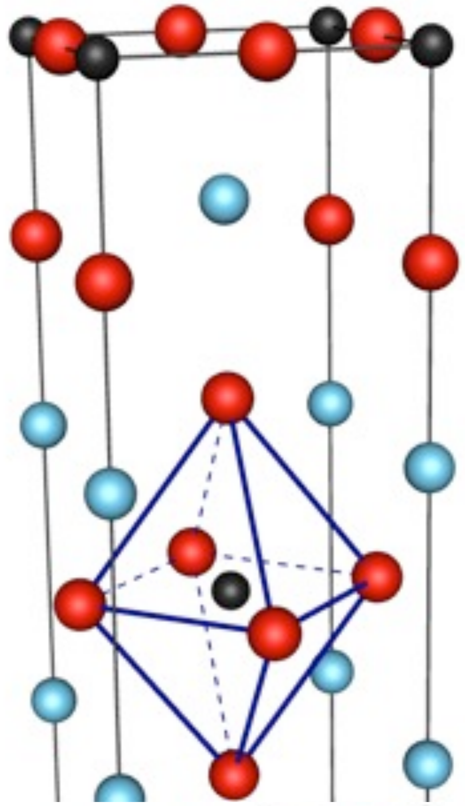
$U \approx 8t$

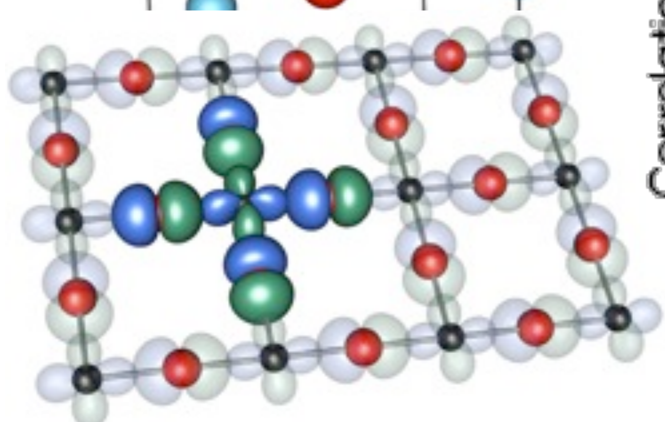
No simple solution!

Finite doping levels (0.05 – 0.25)

Typical values: U~10eV; t~0.9eV; J~0.2eV; (0.1eV ~ 10^3 Kelvin)

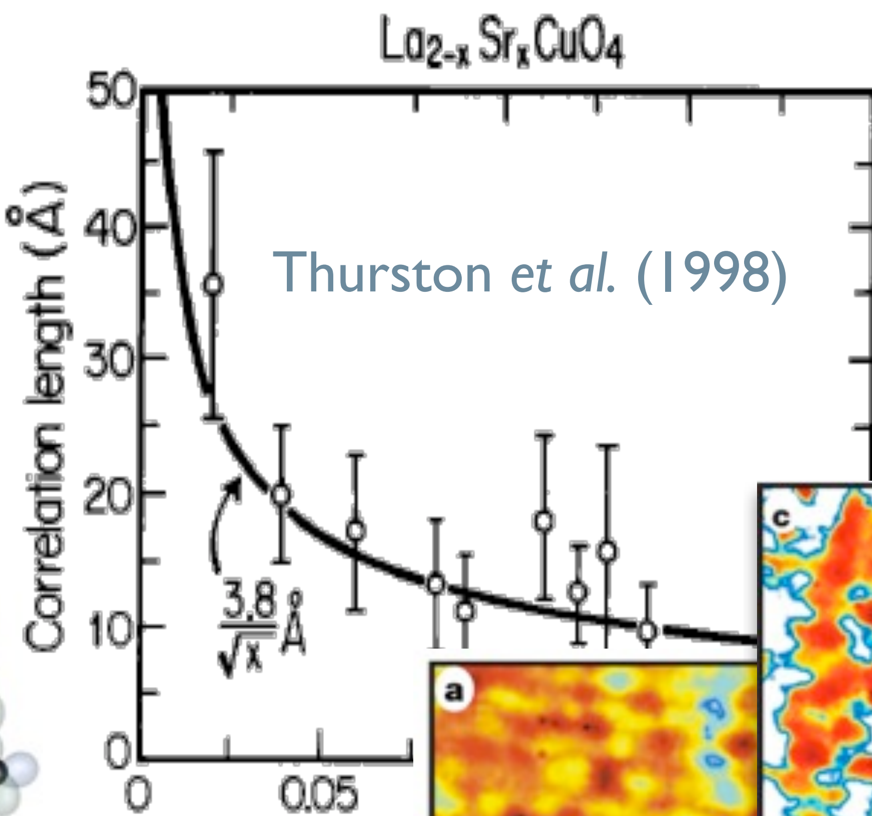
The challenge: a (quantum) multi-scale problem

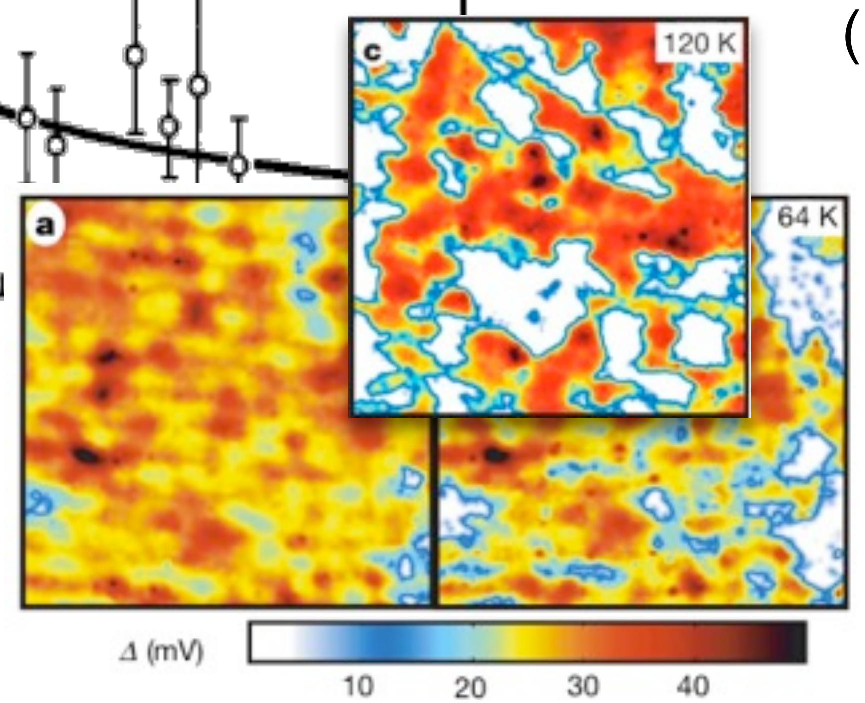




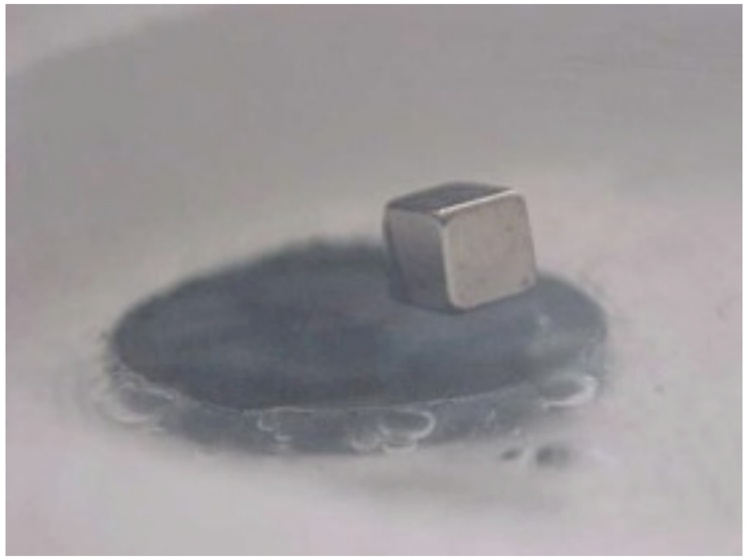
On-site Coulomb repulsion (~A)

Antiferromagnetic correlations / nano-scale gap fluctuations





Gomes et al. (2007)



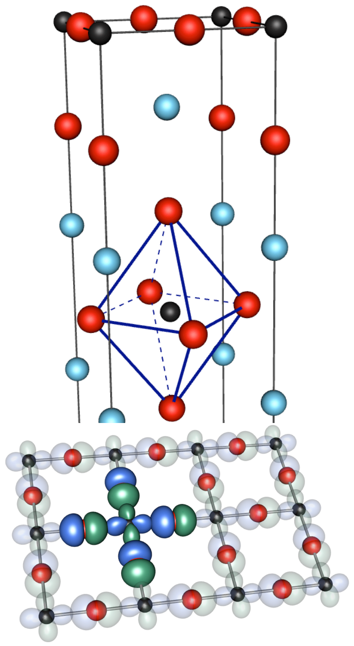
Superconductivity (macroscopic)

$N \sim 10^{23}$

complexity $\sim 4^N$

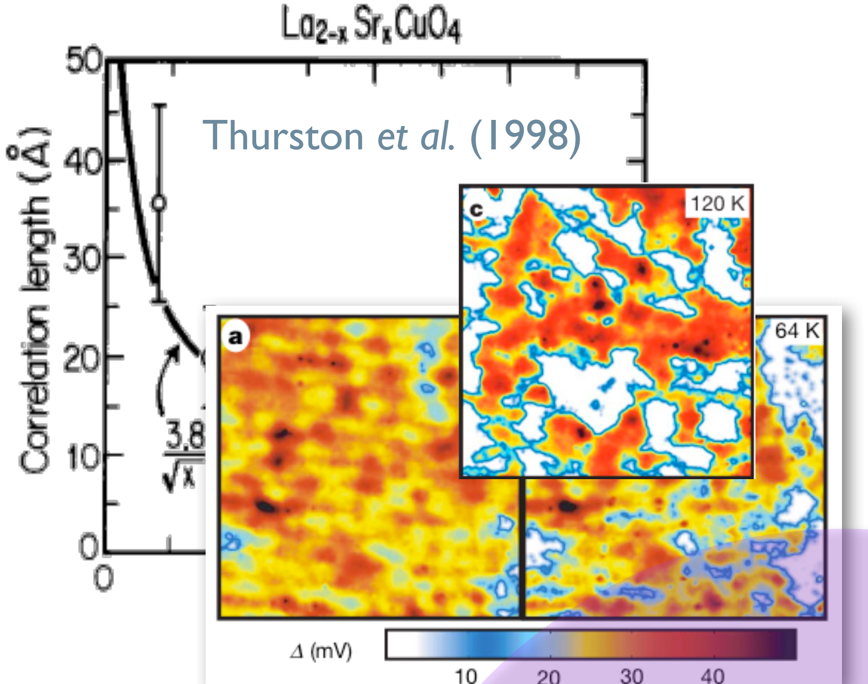
Quantum cluster theories

Maier *et al.*, Rev. Mod. Phys. '05

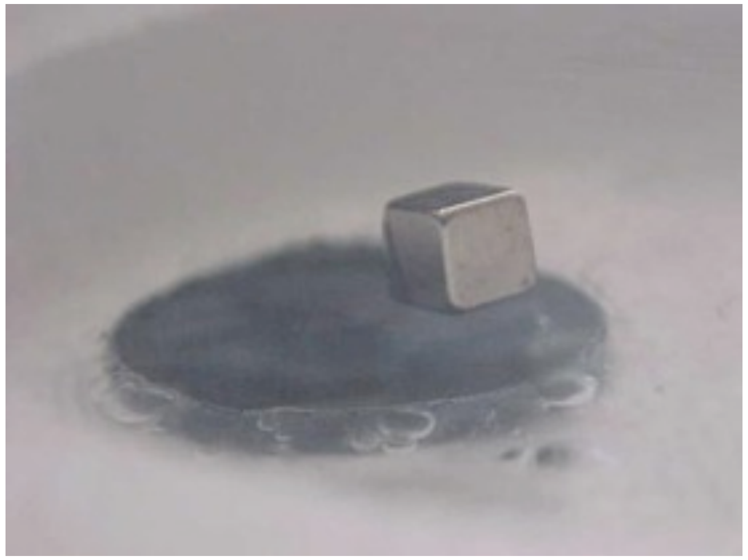


On-site Coulomb repulsion (~A)

Antiferromagnetic correlations / nano-scale gap fluctuations

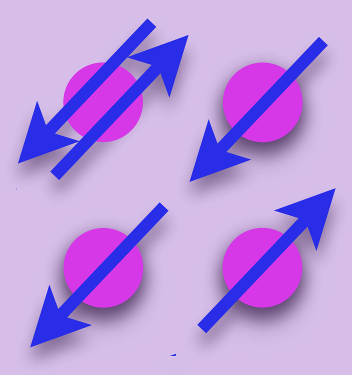


Gomes et al. (2007)



Superconductivity (macroscopic)

Explicitly treat correlations within a localized cluster



Treat macroscopic scales within mean-field

Coherently embed cluster into effective medium

Green's functions in quantum many-body theory

Noninteracting Hamiltonian & $H_0 = \left[-\frac{1}{2}\nabla^2 + V(\vec{r}) \right]$

Green's function $\left[i\frac{\partial}{\partial t} - H_0 \right] G_0(\vec{r},t,\vec{r}',t') = \delta(\vec{r}-\vec{r}')\delta(t-t)$

Fourier transform & analytic continuation: $z^{\pm} = \omega \pm i\epsilon$ $\quad G_0^{\pm}(\vec{r},z) = \left[z^{\pm} - H_0 \right]^{-1}$

Hubbard Hamiltonian $H = -t \sum_{<ij>,\sigma} c_{i\sigma}^{\dagger} c_{j\sigma} + U \sum_i n_{i\uparrow} n_{i\downarrow}$ $\quad n_{i\sigma} = c_{i\sigma}^{\dagger} c_{i\sigma}$

Hide symmetry in algebraic properties of field operators

$$c_{i\sigma}c_{j\sigma'} + c_{j\sigma'}c_{i\sigma} = 0$$

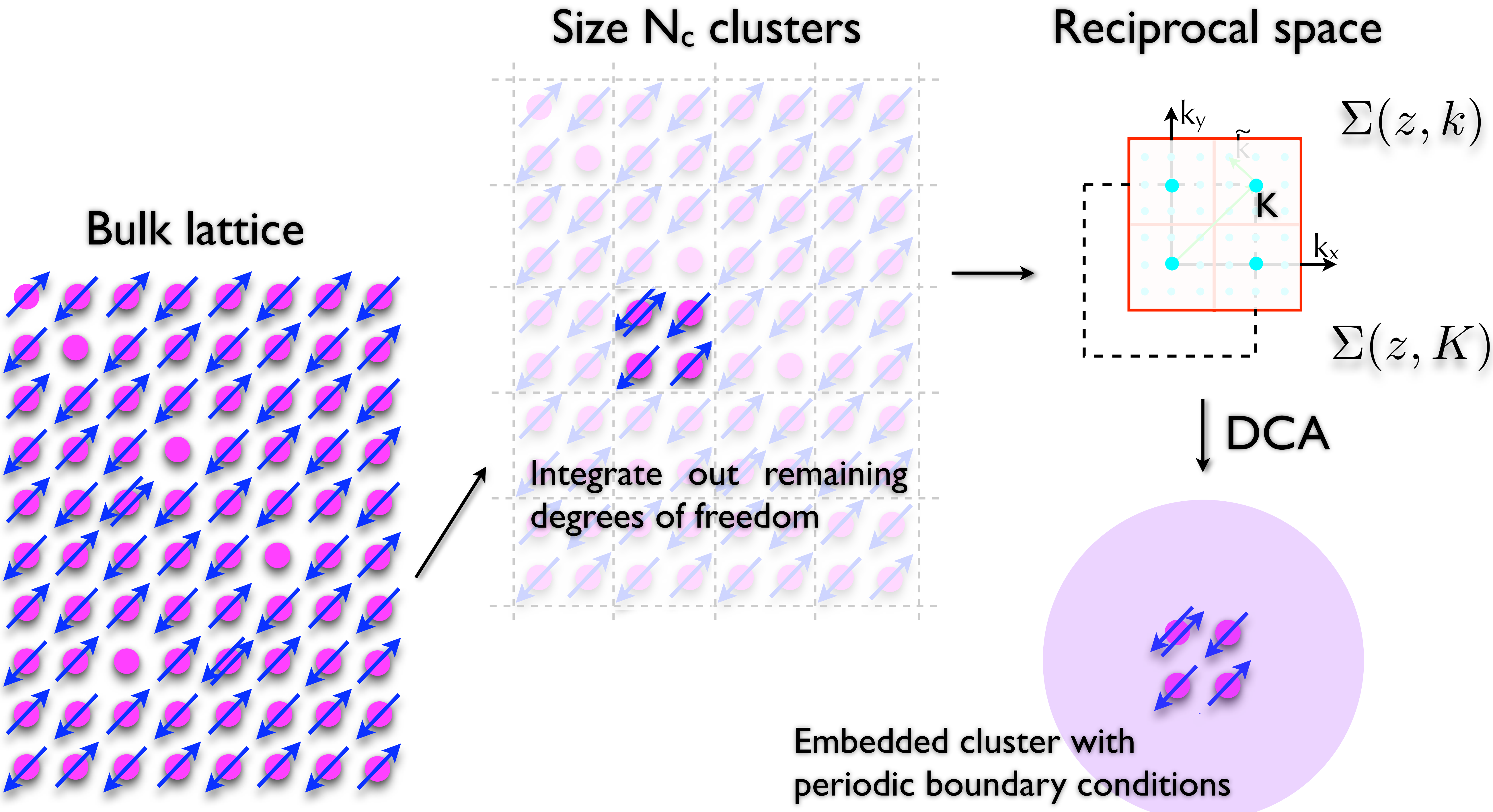
$$c_{i\sigma}c_{j\sigma'}^{\dagger} + c_{j\sigma'}^{\dagger}c_{i\sigma} = \delta_{ij}\delta_{\sigma\sigma'}$$

Green's function $G_\sigma(r_i,\tau;r_j,\tau') = -\left\langle \mathcal{T} c_{i\sigma}(\tau) c_{j\sigma}^{\dagger}(\tau') \right\rangle$

Spectral representation $G_0(k,z) = [z - \epsilon_0(k)]^{-1}$

$$G(k,z) = [z - \epsilon_0(k) - \Sigma(k,z)]^{-1}$$

Sketch of the Dynamical Cluster Approximation

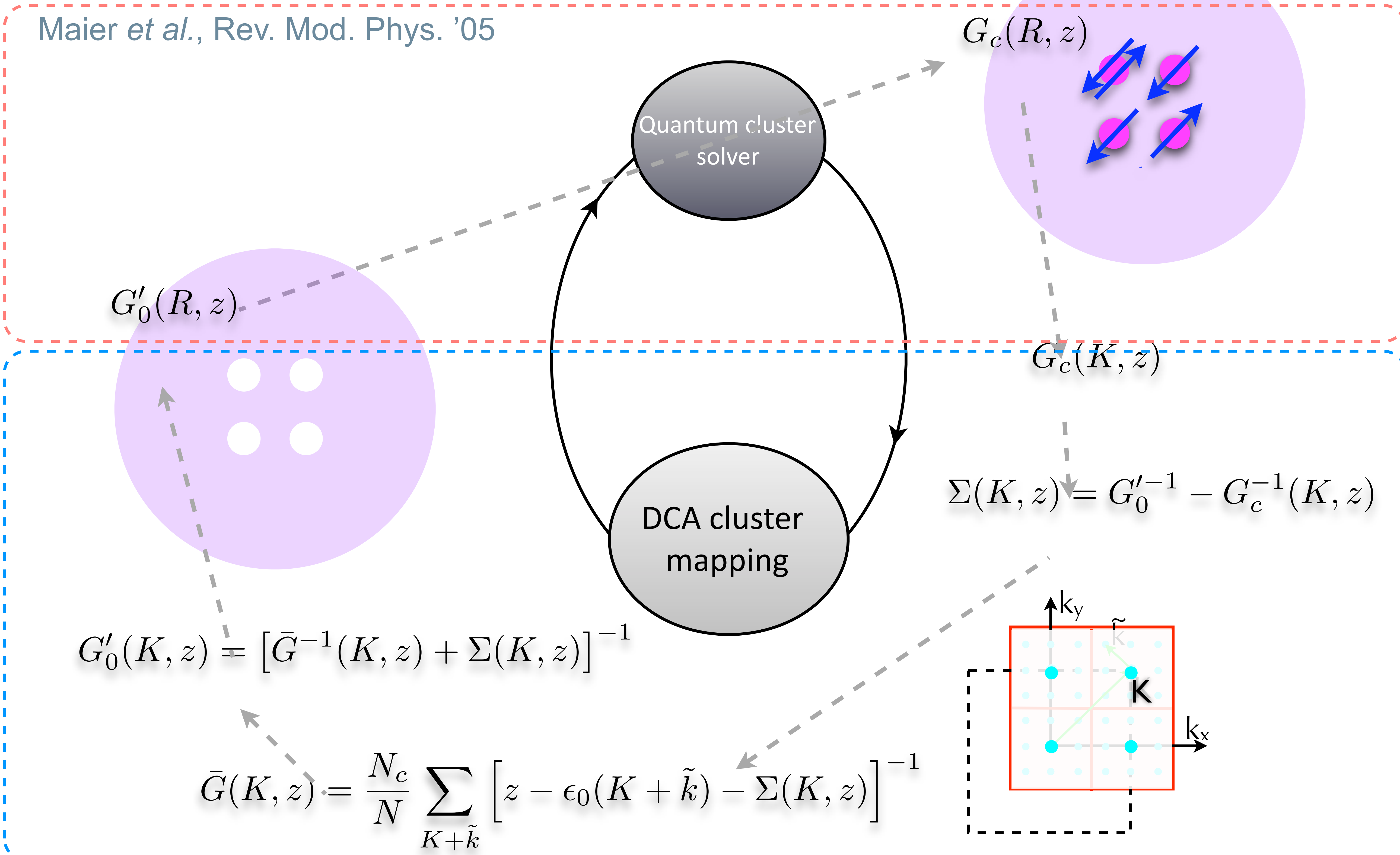


Solve many-body problem with quantum Monte Carole on cluster

➢Essential assumption: Correlations are short ranged

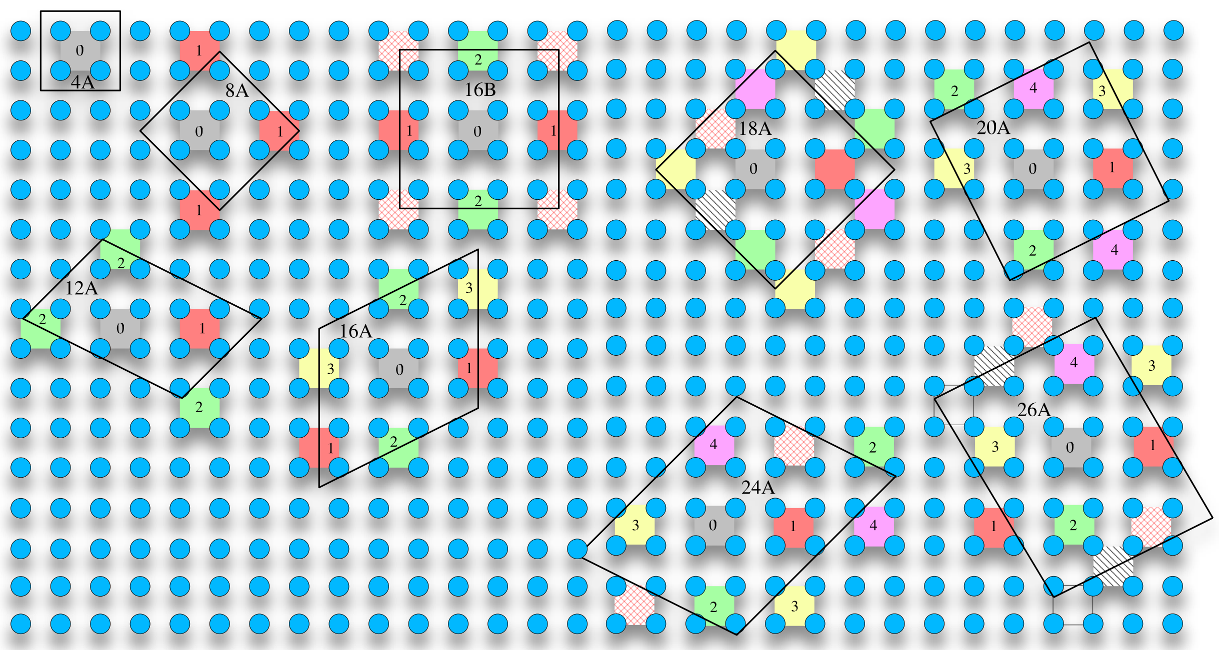
DCA method: self-consistently determine the "effective" medium

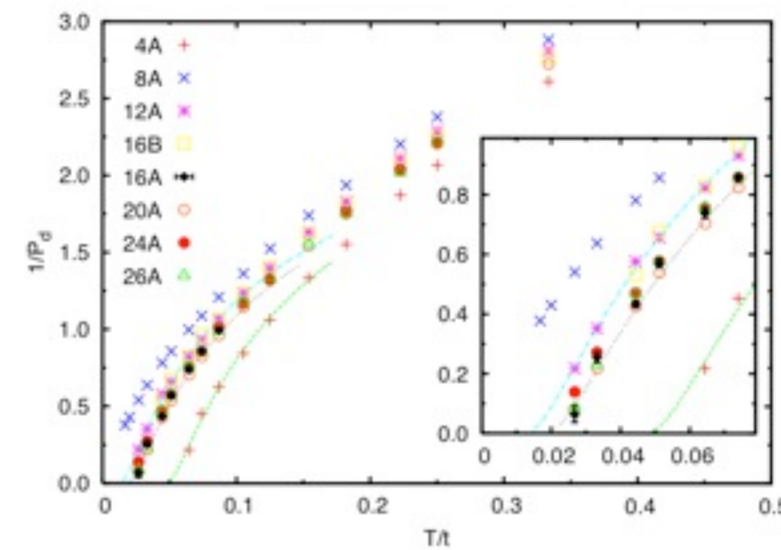
Maier *et al.*, Rev. Mod. Phys. '05



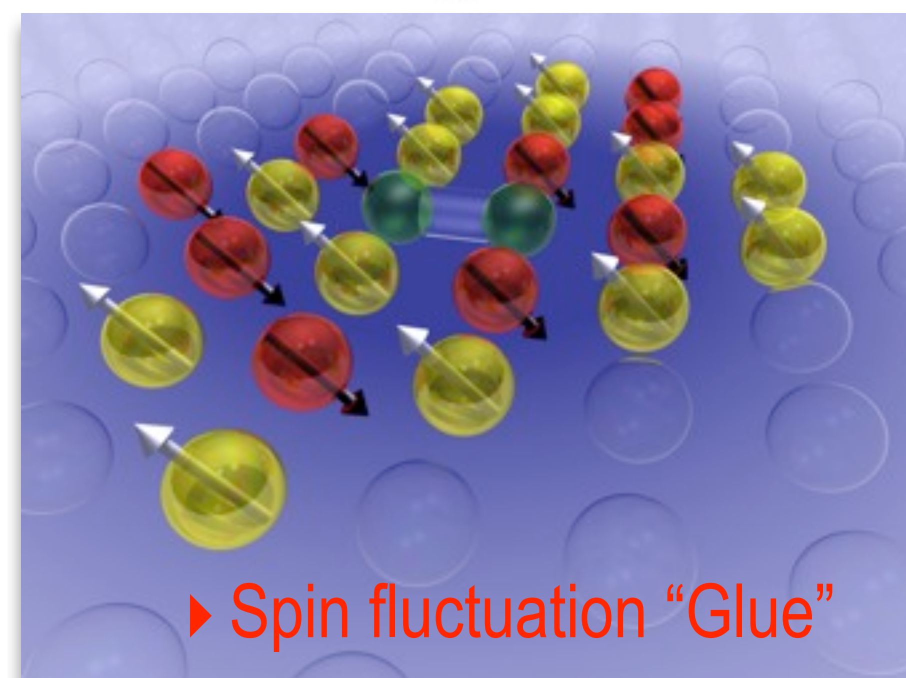
Systematic solution and analysis of the pairing mechanism in the 2D Hubbard Model

- First systematic solution demonstrates existence of a superconducting transition in 2D Hubbard model   Maier, et al., Phys. Rev. Lett. **95** 237001 (2005)



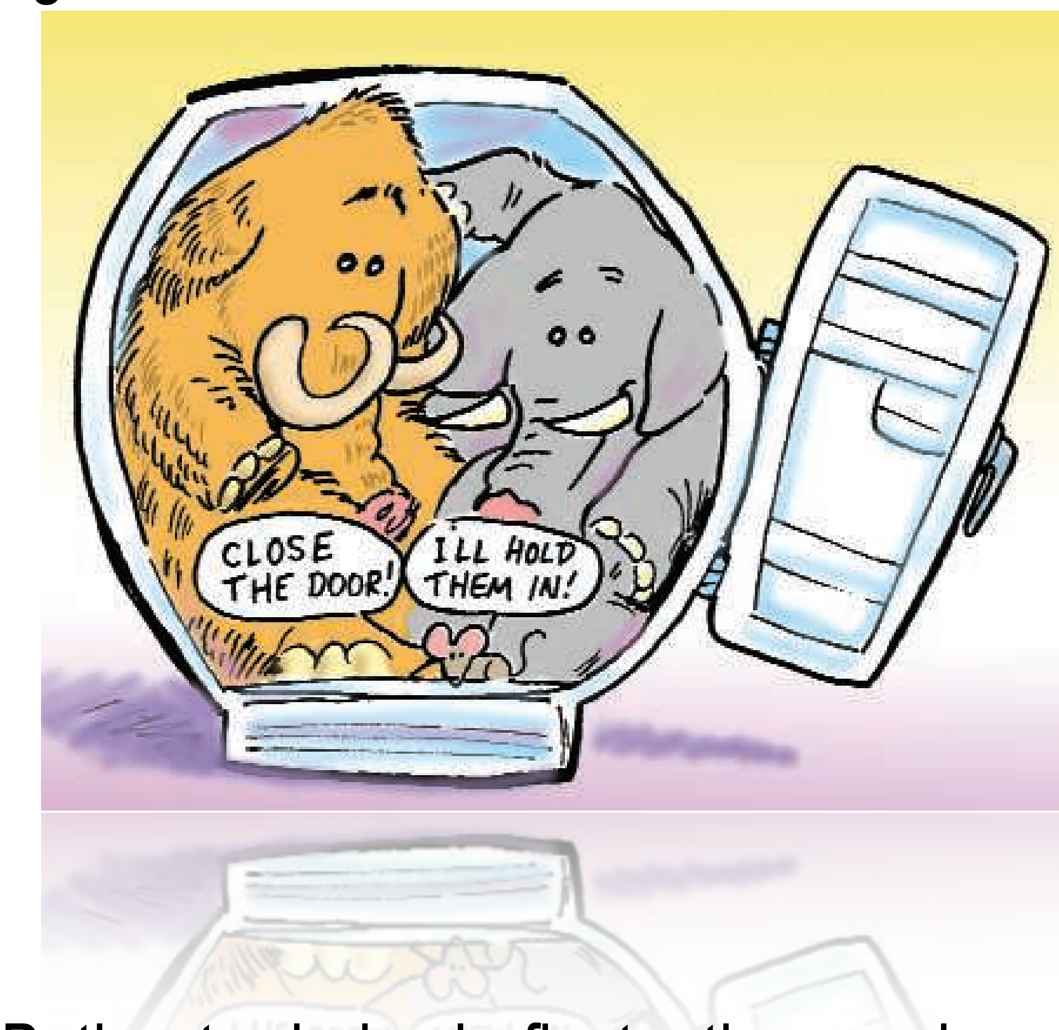


- Study the mechanism responsible for pairing in the model
  - Analyze the particle-particle vertex
  - Pairing is mediated by spin fluctuations
    Maier, et al., Phys. Rev. Lett. **96** 47005 (2006)

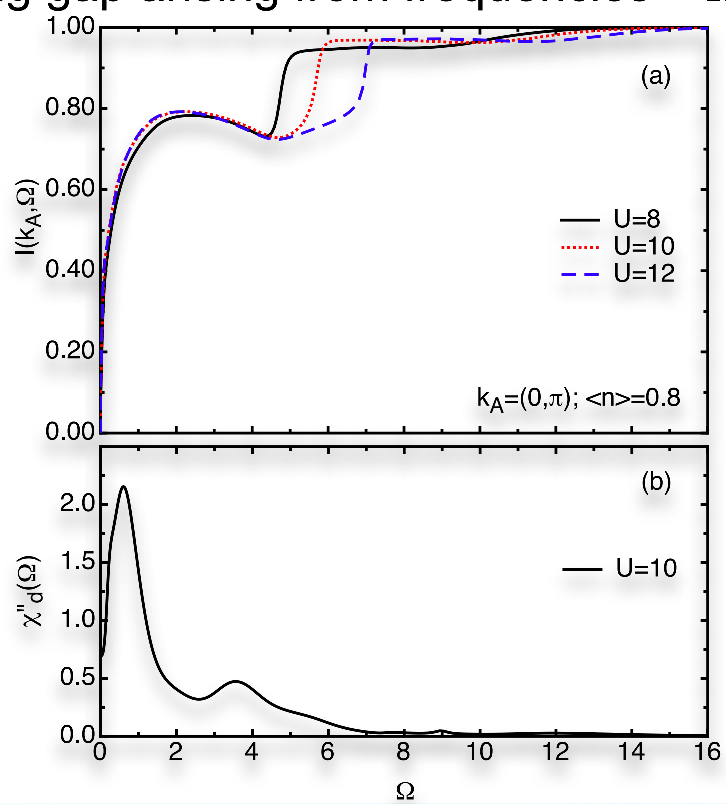


Moving toward a resolution of the debate over the pairing mechanism in the 2D Hubbard model

- “We have a mammoth (*U*) and an elephant (*J*) in our refrigerator - do we care much if there is also a mouse?”
  - P.W. Anderson, Science **316**, 1705 (2007)
  - see also www.sciencemag.org/cgi/eletters/316/5832/1705
    “Scalapino is not a glue sniffer”
- Relative importance of resonant valence bond and spin-fluctuation mechanism
  - Maier et al., Phys. Rev. Lett. **100** 237001 (2008)



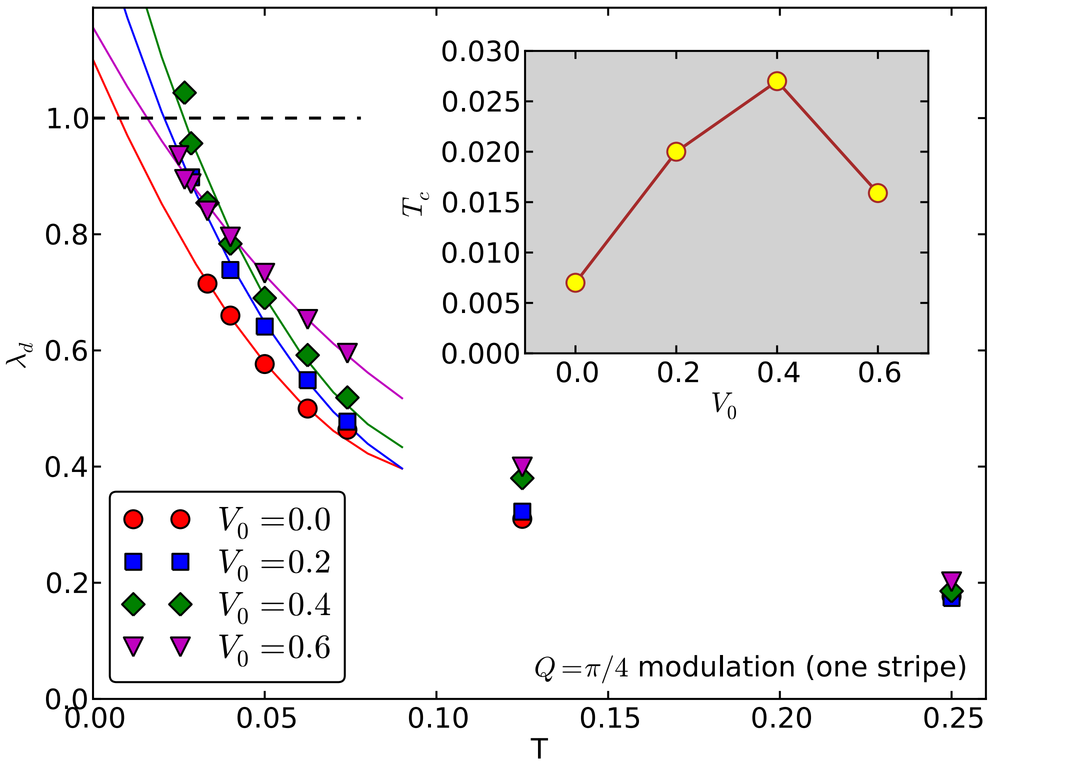
Fraction of superconducting gap arising from frequencies ≤ Ω

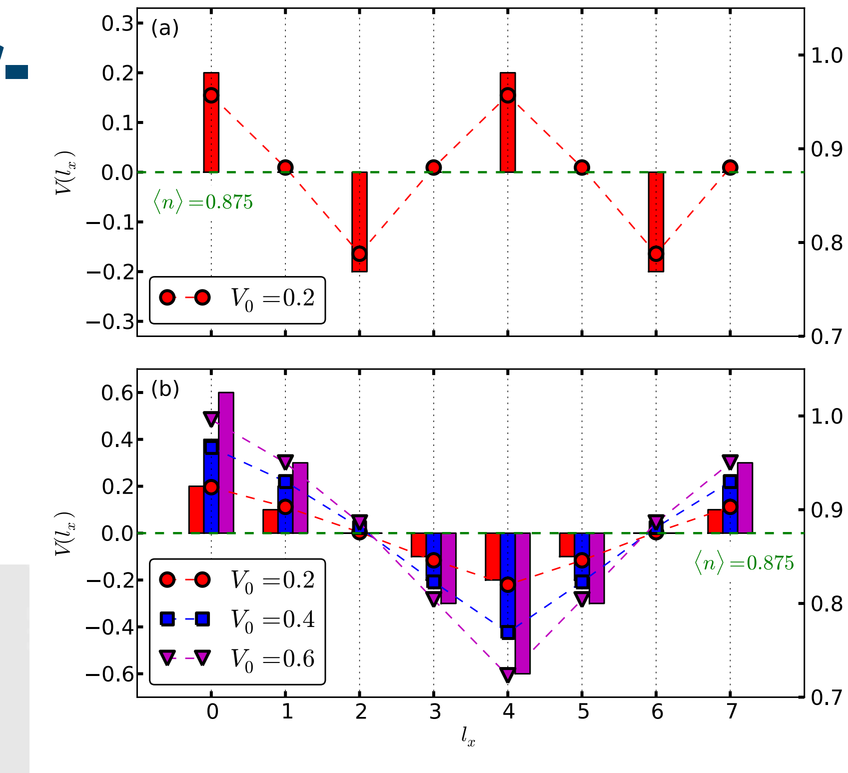


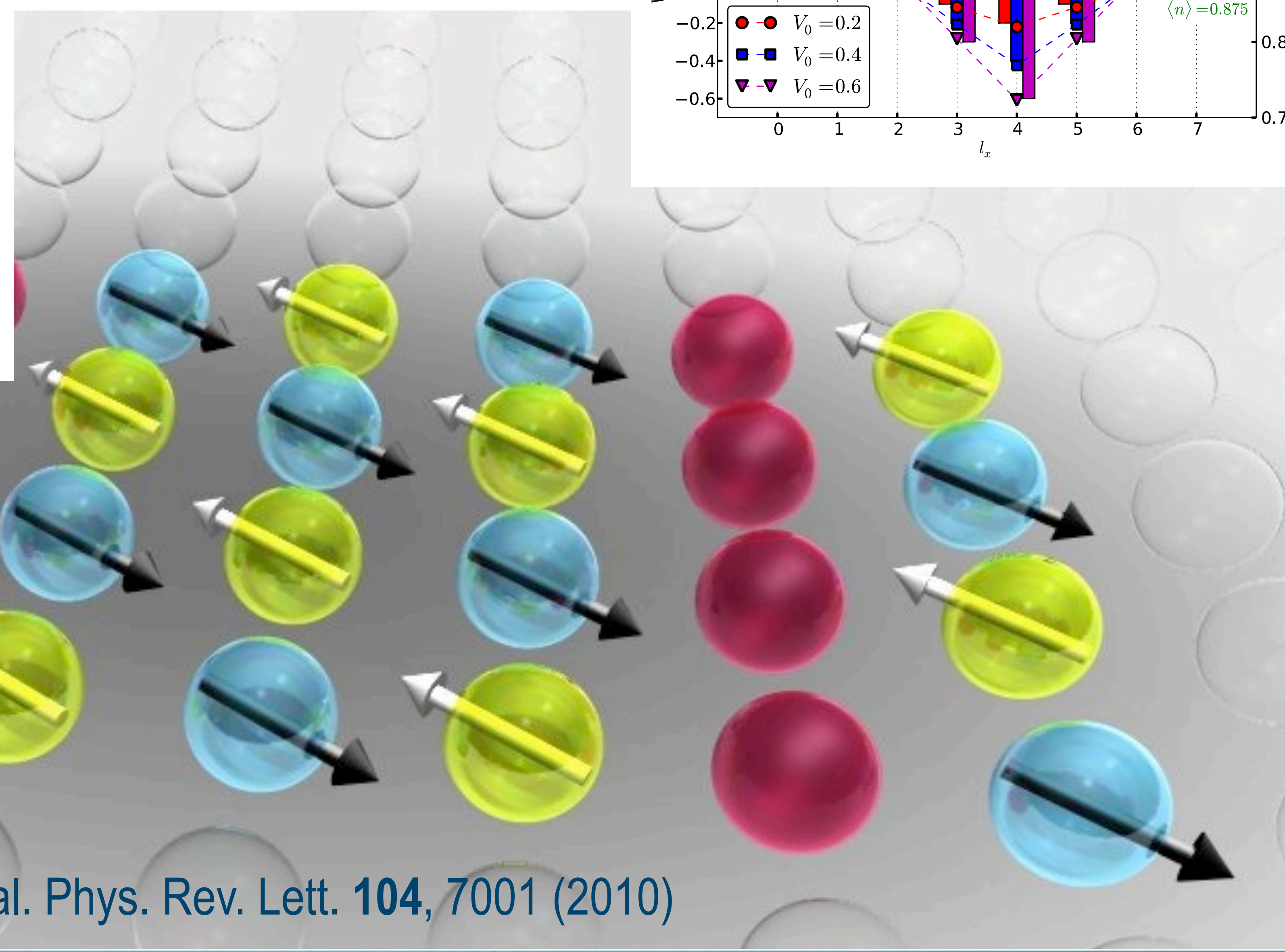
Both retarded spin-fluctuations and non-retarded exchange interaction J contribute to the pairing interaction

Dominant contribution comes from spin-fluctuations!

Nanoscale stripe modulations enhance super-conducting transition temperature







Maier, et al. Phys. Rev. Lett. **104**, 7001 (2010)

Hirsch-Fye Quantum Monte Carole (HF-QMC) for the quantum cluster solver

Hirsch & Fye, Phys. Rev. Lett. **56**, 2521 (1998)

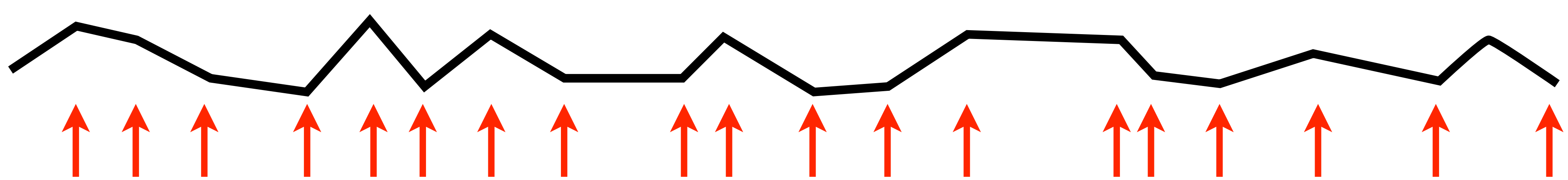
Partition function & Metropolis Monte Carlo $Z = \int e^{-E[\mathbf{x}]/k_\mathrm{B}T} d\mathbf{x}$

Acceptance criterion for Metropolis-MC move: $\min\{1, e^{E[\mathbf{x}_k]-E[\mathbf{x}_{k+1}]}\}$

Partition function & HF-QMC: $Z \sim \sum_{s_i,l} \det[\mathbf{G}_c(s_i,l)^{-1}]$

N_c (for s_i), $N_l \approx 10^2$ (for l)

Acceptance: $\min\{1, \det[\mathbf{G}_c(\{s_i,l\}_k)]/\det[\mathbf{G}_c(\{s_i,l\}_{k+1})]\}$

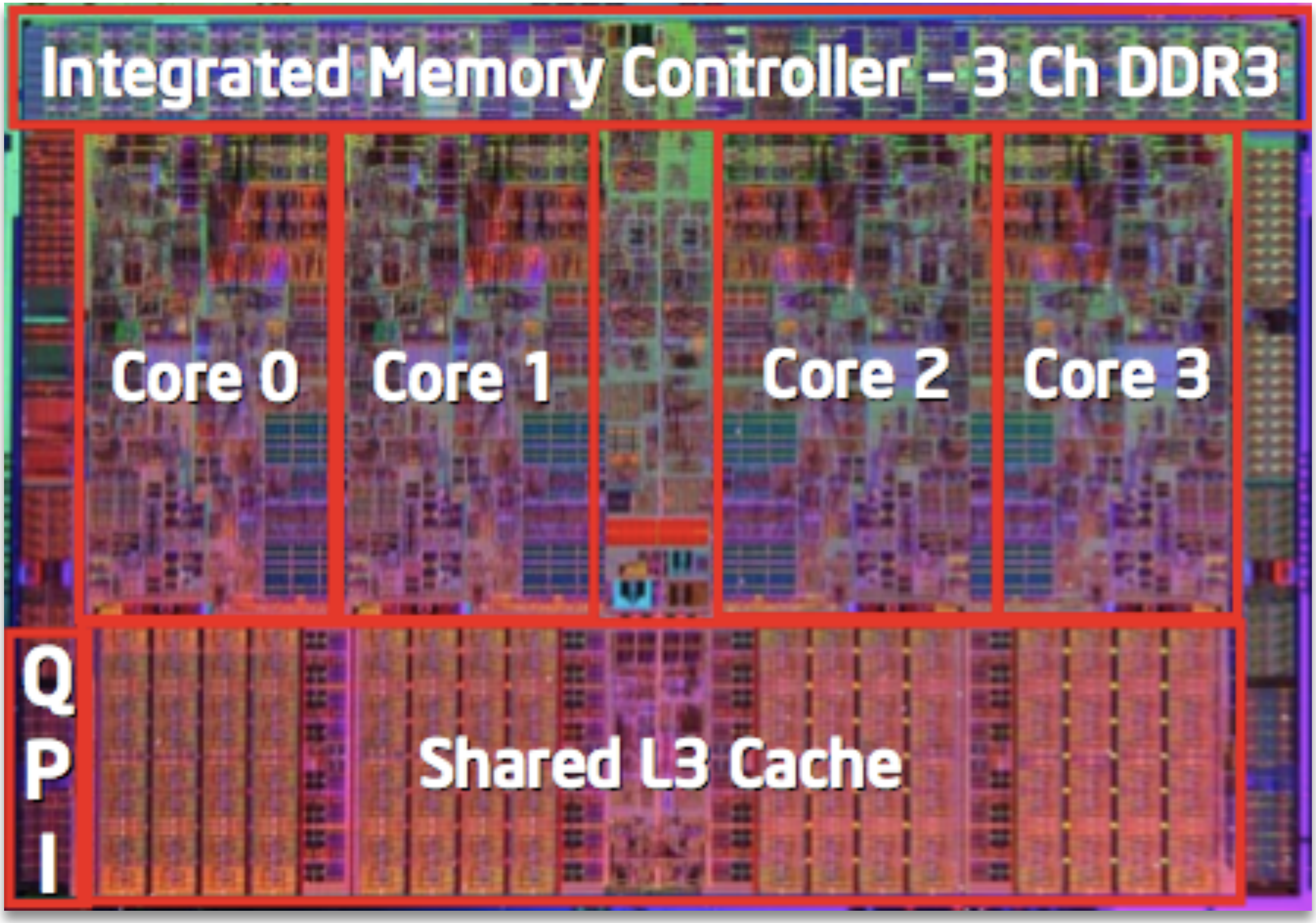


Update of accepted Green's function: matrix of dimensions $N_t \times N_t$

$$\mathbf{G}_c(\{s_i,l\}_{k+1}) = \mathbf{G}_c(\{s_i,l\}_k) + \mathbf{a}_k \times \mathbf{b}_k$$

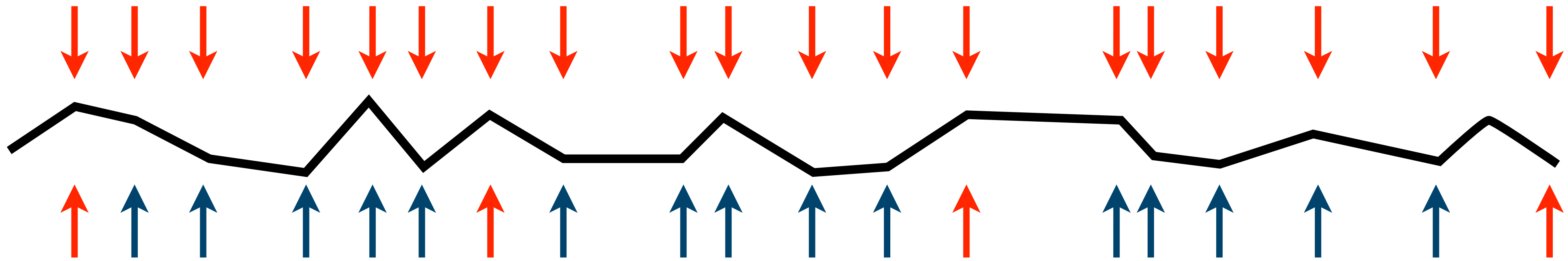
$N_t = N_c \times N_l \approx 2000$

Take advantage of many-cores / shared L3 cash?



HF-QMC with Delayed updates (or Ed updates)

$$\mathbf{G}_c(\{s_i,l\}_{k+1}) = \mathbf{G}_c(\{s_i,l\}_k) + \mathbf{a}_k \times \mathbf{b}_k^t$$



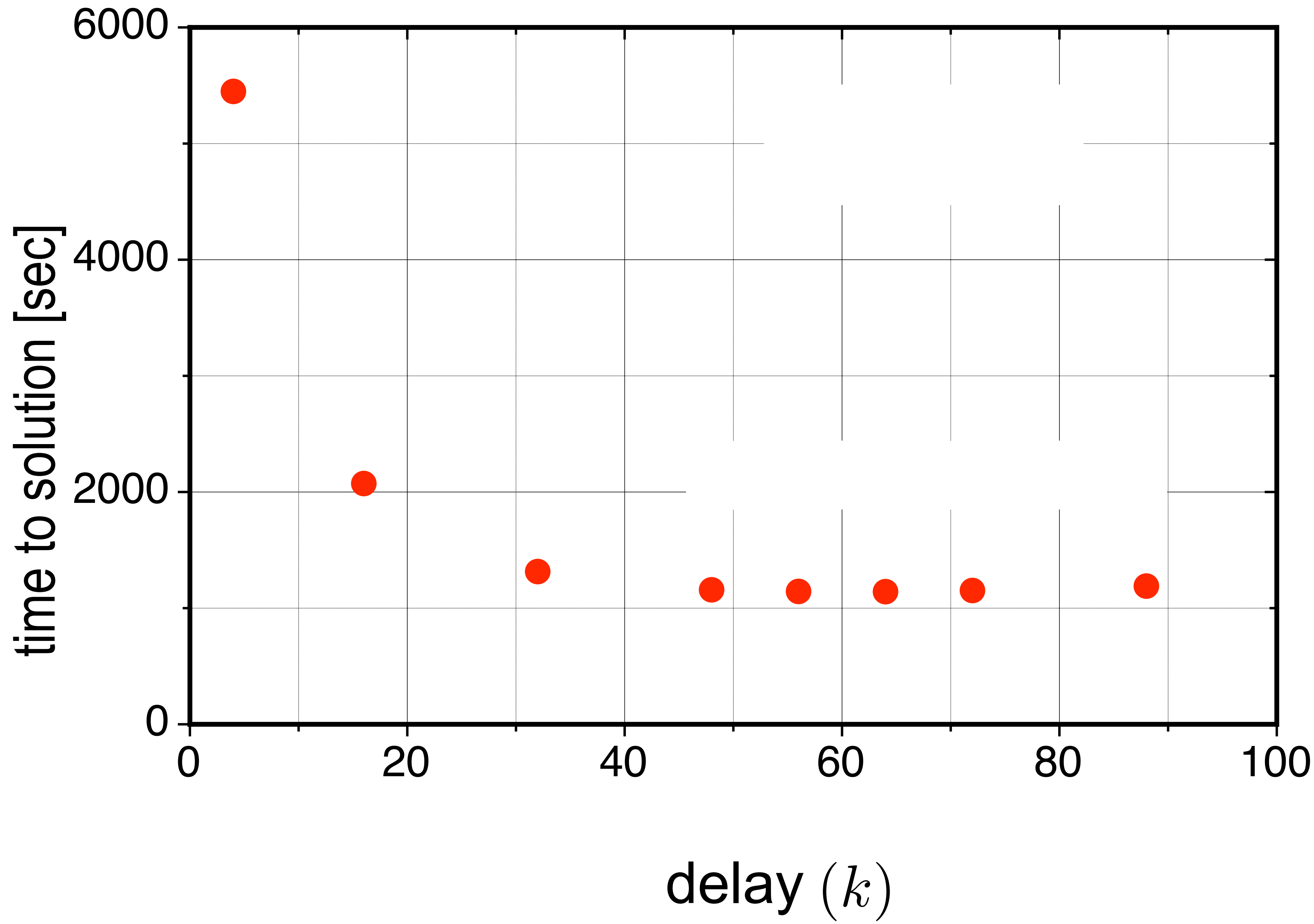
$$\mathbf{G}_c(\{s_i,l\}_{k+1}) = \mathbf{G}_c(\{s_i,l\}_0) + [\mathbf{a}_0|\mathbf{a}_1|...|\mathbf{a}_k] \times [\mathbf{b}_0|\mathbf{b}_1|...|\mathbf{b}_k]^t$$

Complexity for *k* updates remains $\mathcal{O}(kN_t^2)$

But we can replace *k* rank-1 updates with one matrix-matrix multiply plus some additional bookkeeping.

Performance improvement with delayed updates

$N_c = 16 \qquad N_l = 150 \qquad N_t = 2400$

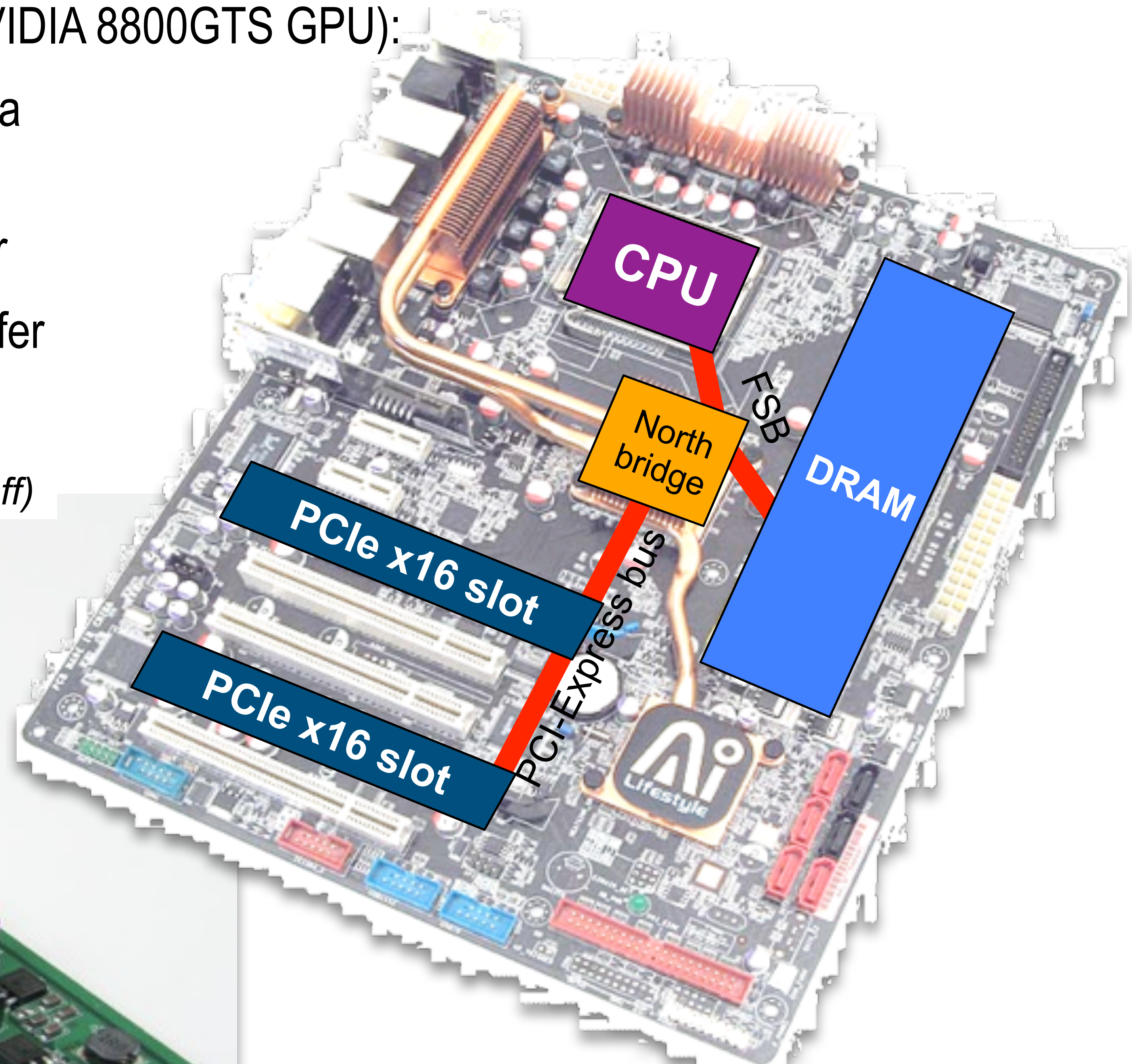


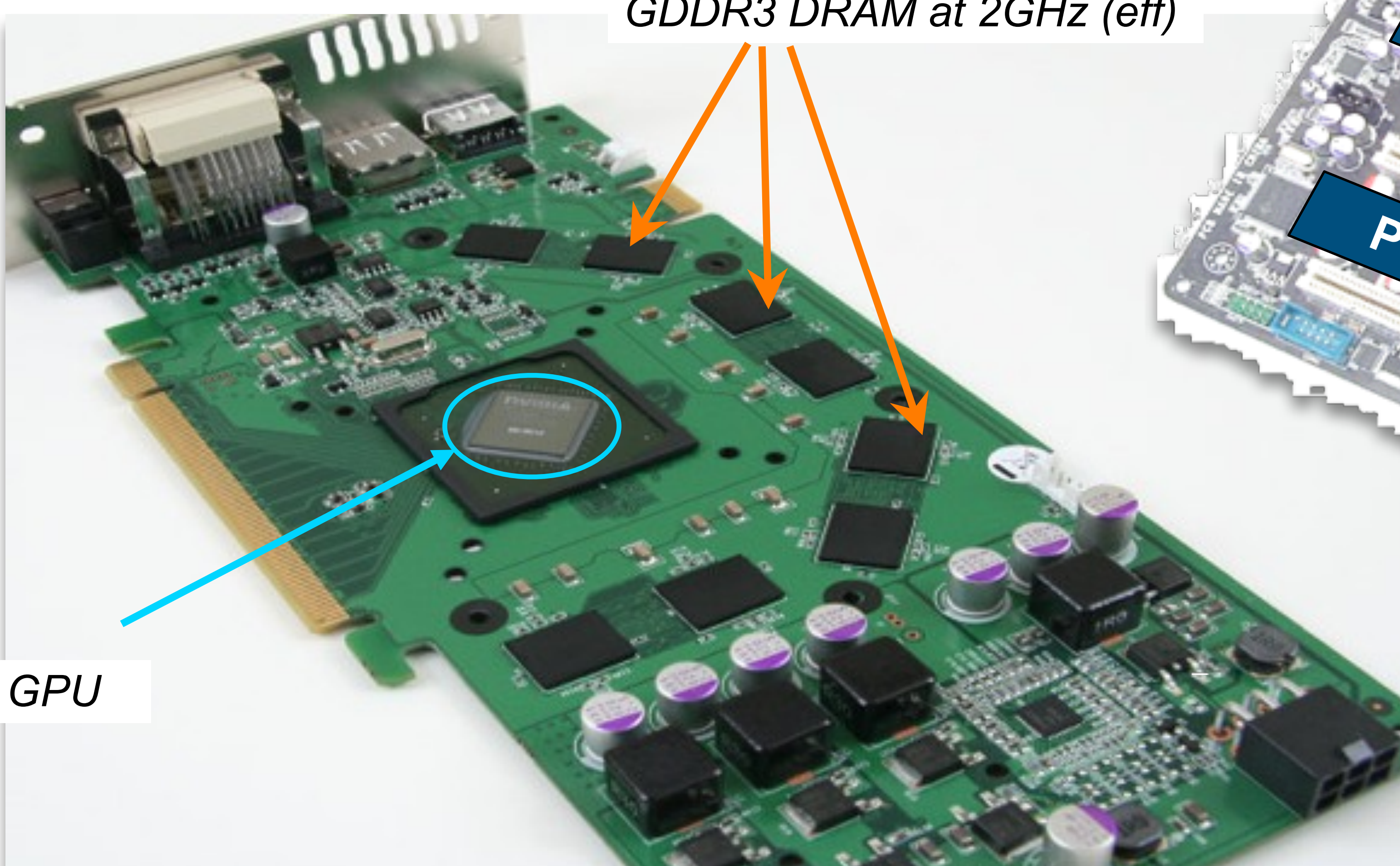
DCA++ speedup on GPU

Meredith et al., Par. Comp. **35**, 151 (2009)

Speedup of HF-QMC updates (2GHz Opteron vs. NVIDIA 8800GTS GPU):

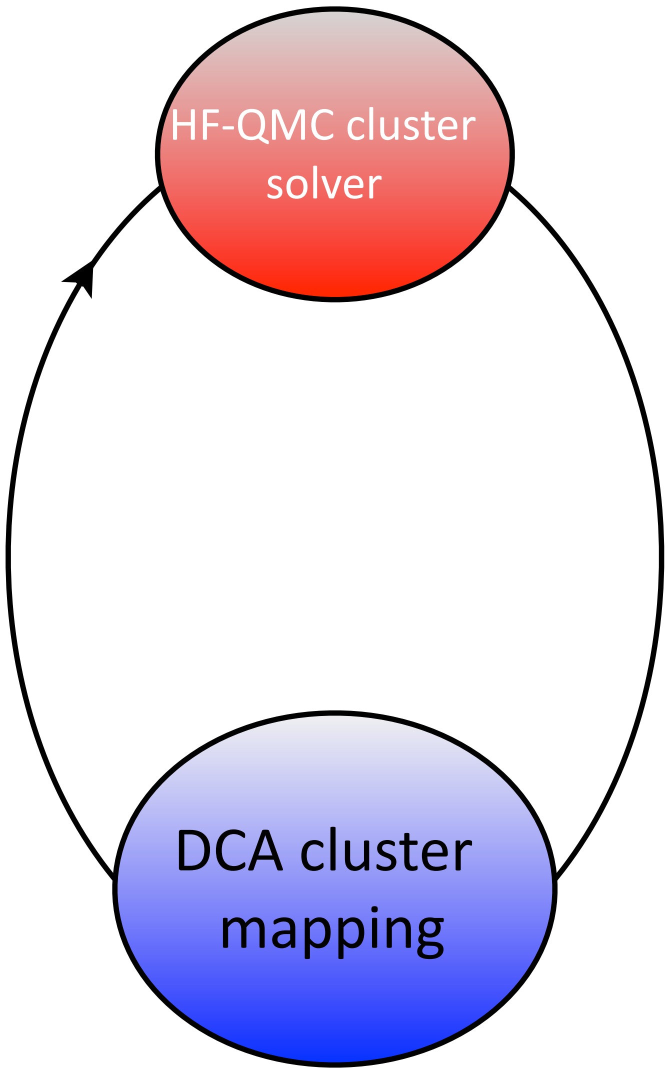
- 9x for offloading BLAS to GPU & transferring all data (completely transparent to application code)
- 13x for offloading BLAS to GPU & lazy data transfer
- 19x for full offload HF-updates & full lazy data transfer

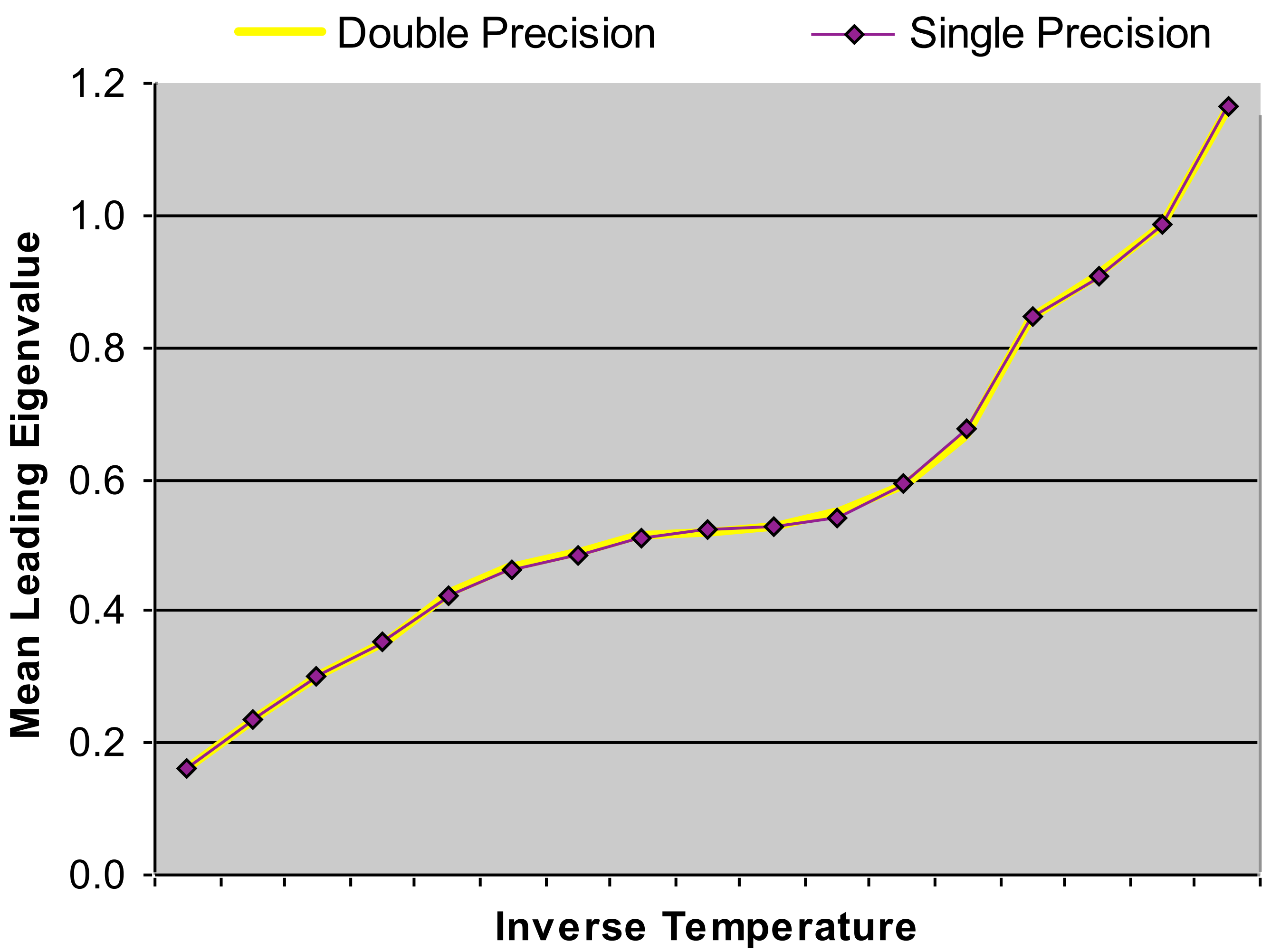




DCA++ with mixed precision

Run HF-QMC in single precision

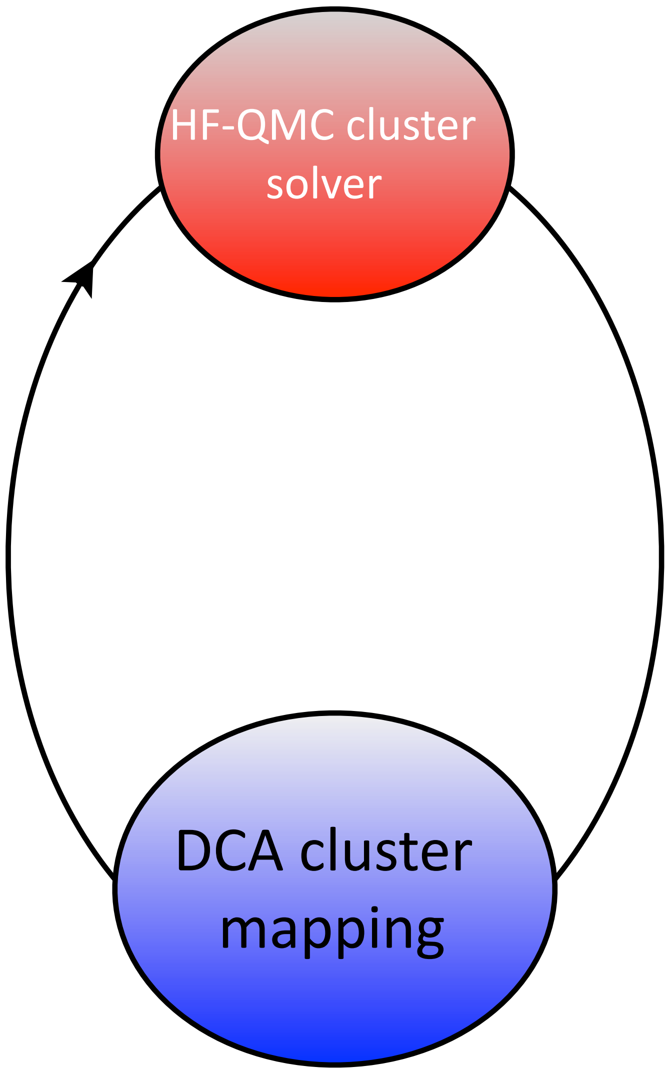




Keep the rest of the code, in particular cluster mapping in double precision

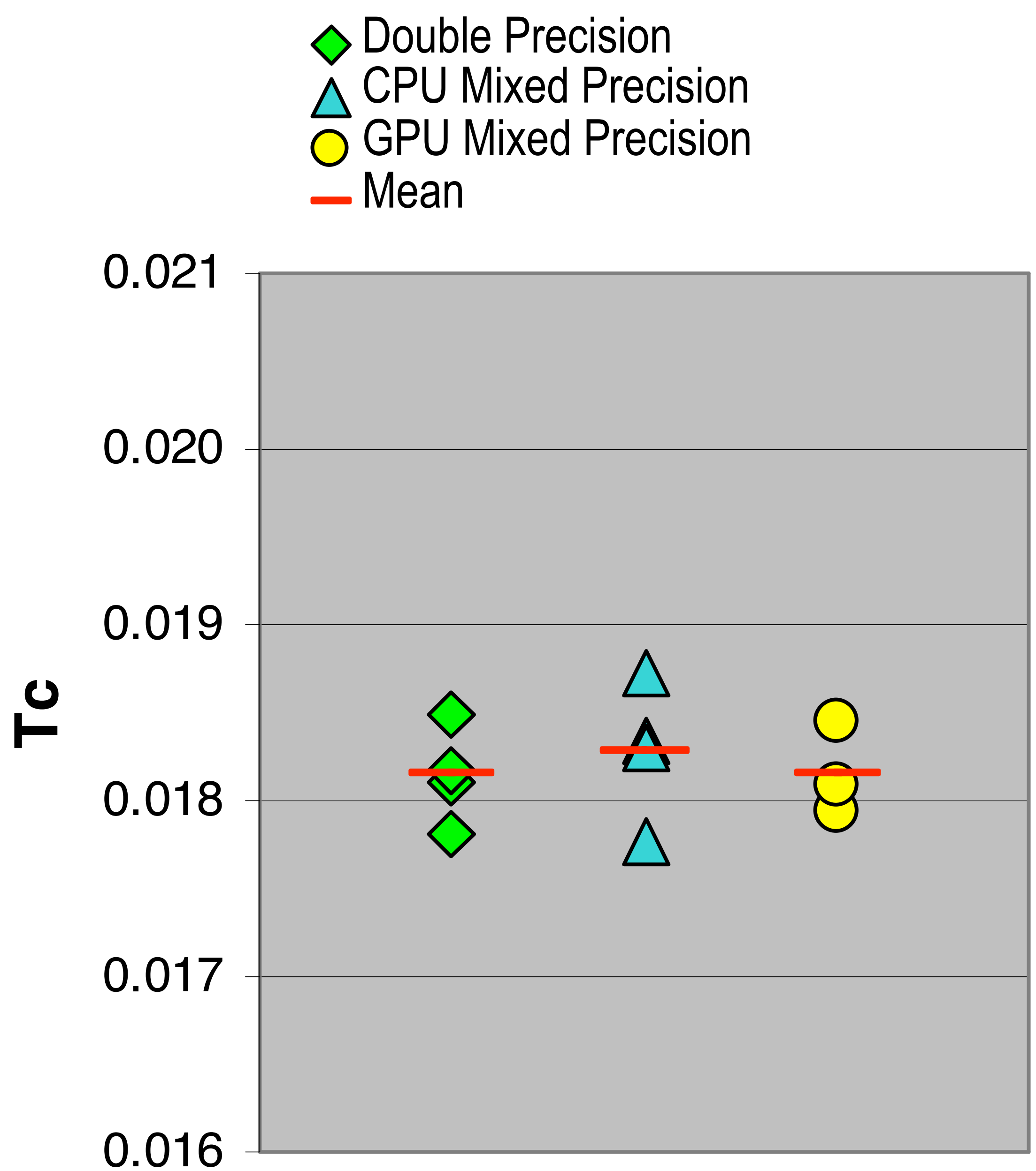
DCA++ with mixed precision

Run HF-QMC in single precision



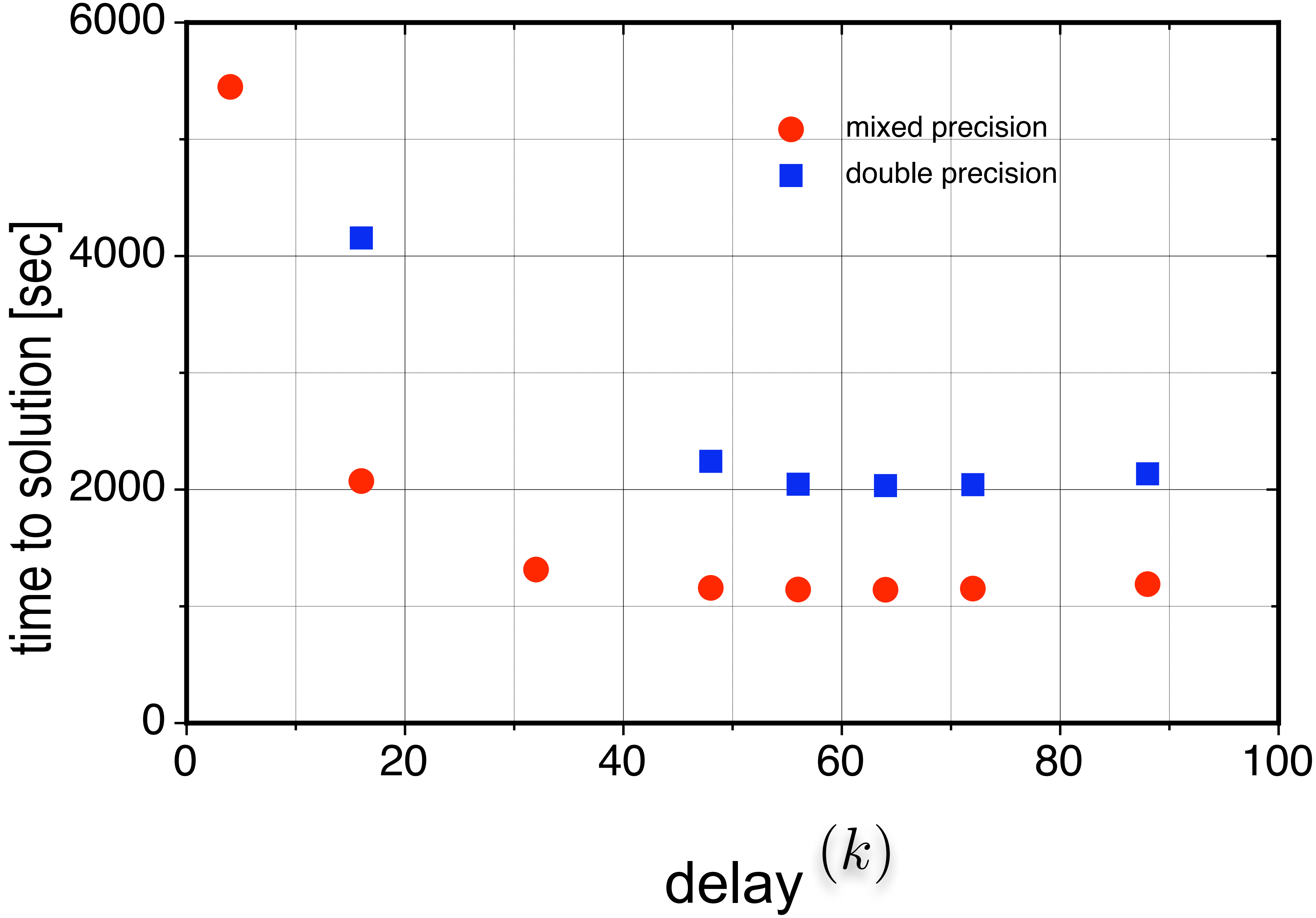
Keep the rest of the code, in particular cluster mapping in double precision

Multiple runs to compute T_c:



Performance improvement with delayed and mixed precision updates

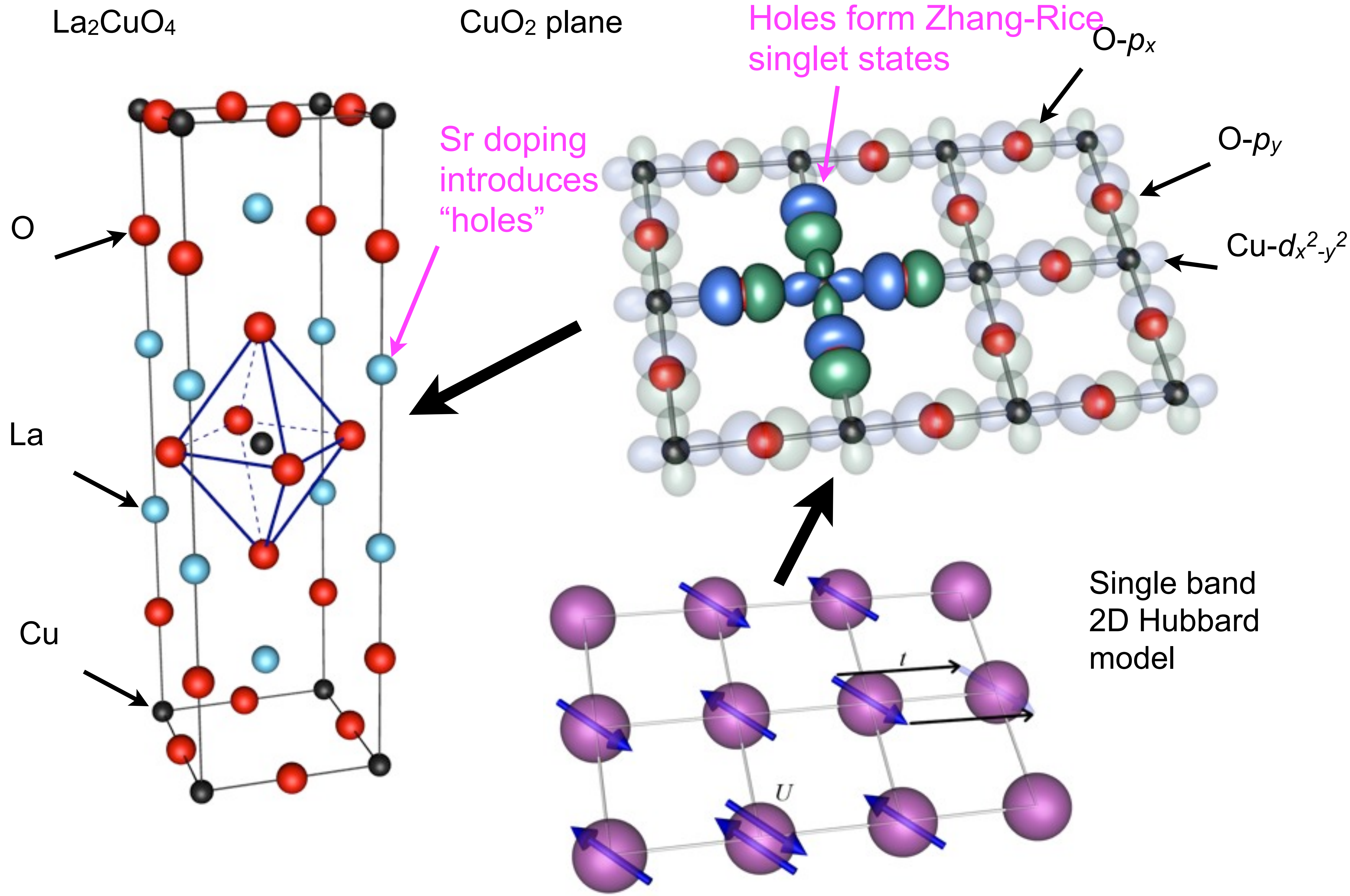
$N_c = 16 \qquad N_l = 150 \qquad N_t = 2400$



Hirsch-Fye, delayed updates, and beyond: sub-matrix updates and continuous time QMC

- J. E. Hirsch and R. M. Fye, Phys. Rev. Lett. **56**, 2521 (1986)
  - Original Hirsch-Fye algorithm with rank 1 update
- G. Alvarez et al., Proceedings of the 2008 ACM/IEEE Conference on Supercomputing
  - Hirsch-Fye algorithm with delayed updates – same complexity but with rank k update (much more efficient)
- P. K. V. V. Nukala et al., Phys. Rev. B **80**, 195111 (2009)
  - Hirsch-Fye with sub-matrix updates – reduce complexity but retain high-rank updates
- E. Gull et al., Phys. Rev. B **76**, 235123 (2007)
  - Continuous time auxiliary (CT-AUX) field QMC algorithm – much faster & more accurate/reliable than Hirsch-Fye algorithm
- E. Gull et al., Phys. Rev. B **83**, 075122 (2011)
  - CT-AUX algorithm combined with sub-matrix updates – best of all worlds: fast, accurate, reduced complexity and high-rank updates (i.e. efficient)

Making the Hubbard model materials specific (...)



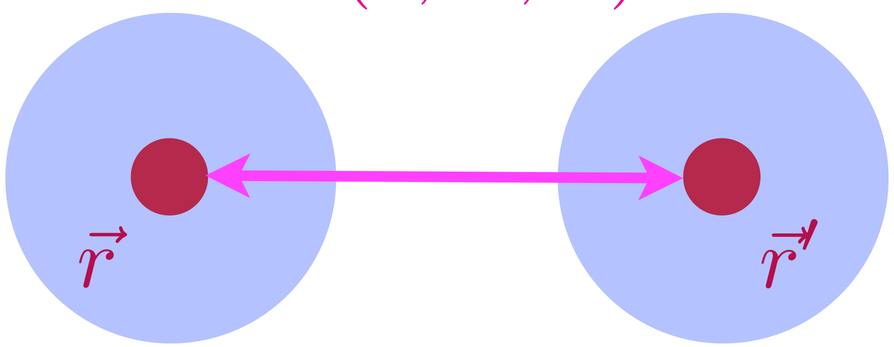
Taylor expansion of self-energy & Green's function

Bare Coulomb interaction

$$w(\vec{r},\vec{r}') = \frac{e^2}{|\vec{r}-\vec{r}'|}$$

Screened Coulomb interaction

$$W(\vec{r},\vec{r}',\omega)$$



Taylor expansion of the self-energy in terms of the screened Coulomb interaction

$$\Sigma = iGW + GWGW + \ldots$$

GW approximation: $$\Sigma(\vec{r},\vec{r}',\omega) = \frac{i}{2\pi}\int d\omega' G(\vec{r},\vec{r}',\omega+\omega')W(\vec{r},\vec{r}',\omega)$$

The challenge: $W(\vec{r},\vec{r}',\omega)$ is extremely expensive to compute

Can this be computed at scale and efficiently?

Screened Coulomb interaction from time dependent DFT or the random phase approximation

$$W_{\mathbf{G}\mathbf{G}'} = \frac{4\pi}{|\mathbf{G}+\mathbf{q}|^2}\delta_{\mathbf{G}\mathbf{G}'} + \frac{4\pi}{|\mathbf{G}+\mathbf{q}|^2}\chi_{\mathbf{G}\mathbf{G}'}(\mathbf{q},\omega)\frac{4\pi}{|\mathbf{G}'+\mathbf{q}|^2}$$

With LAPW only up to 10^3 **G** vectors

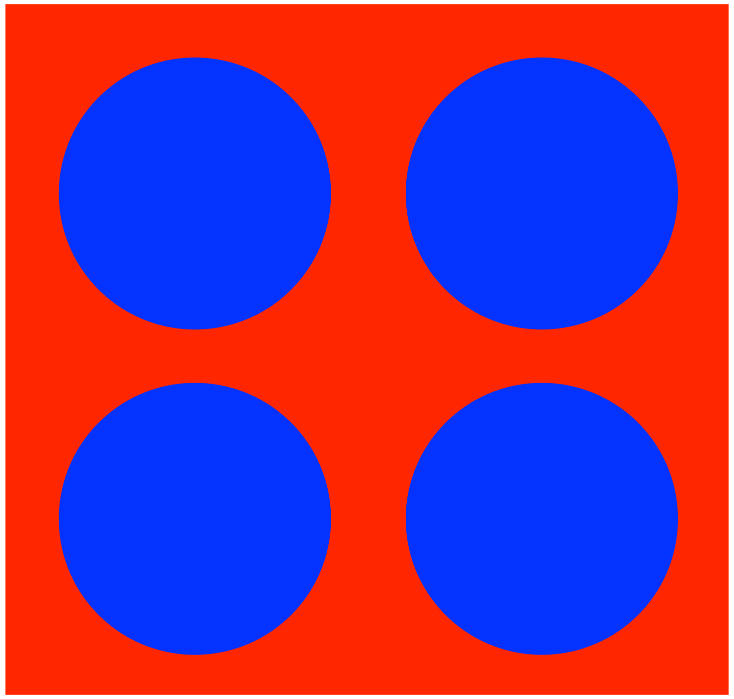
$$\chi_{\mathbf{G}\mathbf{G}'}(\mathbf{q},\omega) = \chi^{KS}_{\mathbf{G}\mathbf{G}'}(\mathbf{q},\omega) + \sum_{\mathbf{G}_1\mathbf{G}_2} \chi^{KS}_{\mathbf{G}\mathbf{G}_1}(\mathbf{q},\omega) \times \left(\frac{4\pi}{|\mathbf{G}_1+\mathbf{q}|}\delta_{\mathbf{G}_1\mathbf{G}_2} + f^{xc}_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q},\omega)\right)\chi_{\mathbf{G}_2\mathbf{G}'}(\mathbf{q},\omega)$$

$$f^{xc}[\rho_0] = \left.\frac{\delta V_{xc}[\rho]}{\delta\rho}\right|_{\rho_0} \approx 0$$ Random Phase Approximation

Block & Wannier functions, screened Hubbard-*U*

$$\psi^{\sigma}_{j\mathbf{k}}(\mathbf{r}) = \begin{cases} \sum_{lm} \sum_{\nu=1}^{N_l^\alpha} A^{\alpha,\sigma j\mathbf{k}}_{lm\nu} u^\alpha_{l\nu}(r) Y_{lm}(\hat{\mathbf{r}}) \\ \frac{1}{\sqrt{\Omega}} \sum_{\mathbf{G}} e^{i(\mathbf{G}+\mathbf{k})\mathbf{r}} C^{\sigma j\mathbf{k}}_{\mathbf{G}} \end{cases}$$

Exciting / Elk code
exciting.sourceforge.org



$$|w^{\mathbf{T}}_n\rangle = \frac{1}{N_k} \sum_{\mathbf{k}}^{BZ} e^{-i\mathbf{K}\mathbf{T}} \sum_j U^{\mathbf{k}}_{nj} |\psi_{j\mathbf{k}}\rangle$$

Screened Hubbard-*U* parameter

Miyake & Aryasetiawan, PRB **77**, 085122 (2008)

$$U^{\mathbf{T}}_{nn'}(\omega) = \frac{1}{N_k \Omega} \sum_{\mathbf{q}}^{BZ} \sum_{\mathbf{G}\mathbf{G}'} \langle w^{\mathbf{0}}_n | e^{-i(\mathbf{G}+\mathbf{q})\mathbf{r}} | w^{\mathbf{0}}_n \rangle \times$$

$$\times W_{\mathbf{G}\mathbf{G}'}(\mathbf{q},\omega) \langle w^{\mathbf{T}}_{n'} | e^{i(\mathbf{G}'+\mathbf{q})\mathbf{r}} | w^{\mathbf{T}}_{n'} \rangle$$

The computationally intensive part – lots of nested loops

$$\chi^{KS}_{\mathbf{G}\mathbf{G}'}(\mathbf{q},\omega) = \frac{1}{N_k\Omega}\sum_{\mathbf{k}}^{BZ}\sum_{jj'}\langle\psi_{j\mathbf{k}}|e^{i(\mathbf{G}+\mathbf{q})\mathbf{r}}|\psi_{j'\mathbf{k}+\mathbf{q}}\rangle\times$$

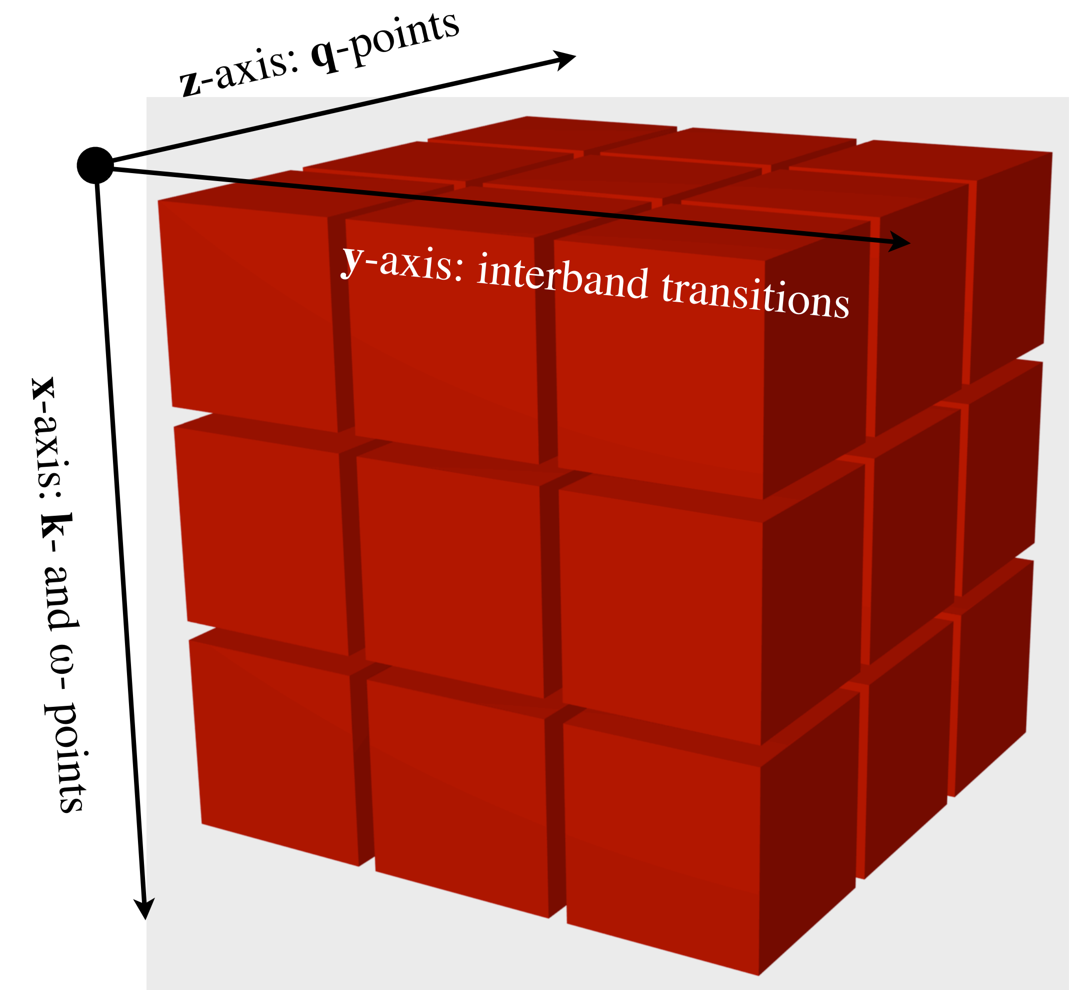
$$\times\frac{f_{j\mathbf{k}}-f_{j'\mathbf{k}+\mathbf{q}}}{\epsilon_{j\mathbf{k}}-\epsilon_{j'\mathbf{k}+\mathbf{q}}+\omega+i0^+}\langle\psi_{j'\mathbf{k}+\mathbf{q}}|e^{-i(\mathbf{G}'+\mathbf{q})\mathbf{r}}|\psi_{j\mathbf{k}}\rangle$$

$$\chi^{KS}_{\mathbf{G}\mathbf{G}'}(\mathbf{q},\omega) = \frac{1}{N_k\Omega}\sum_{\mathbf{k}}^{\mathbf{BZ}}\sum_{\beta}A^{\mathbf{k},\mathbf{q}}_{\beta\mathbf{G}}B^{\mathbf{k},\mathbf{q}}_{\beta\mathbf{G}'}(\omega)$$

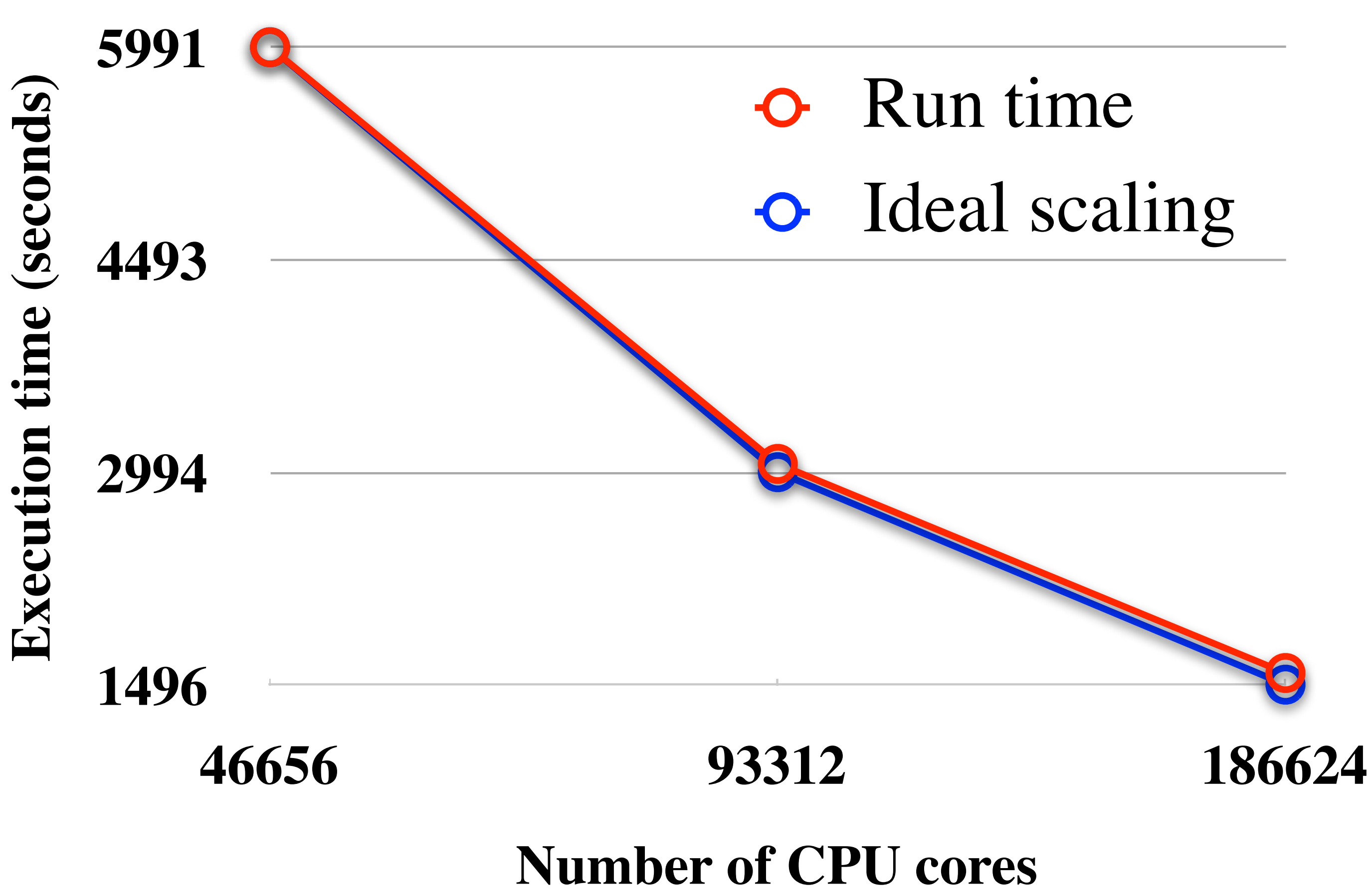
Reduce to a complex matrix multiply – BLAS3 zgemm
(code rewrite yields order of magnitude improvement in time to solution)

Parallelize with MPI-Grid

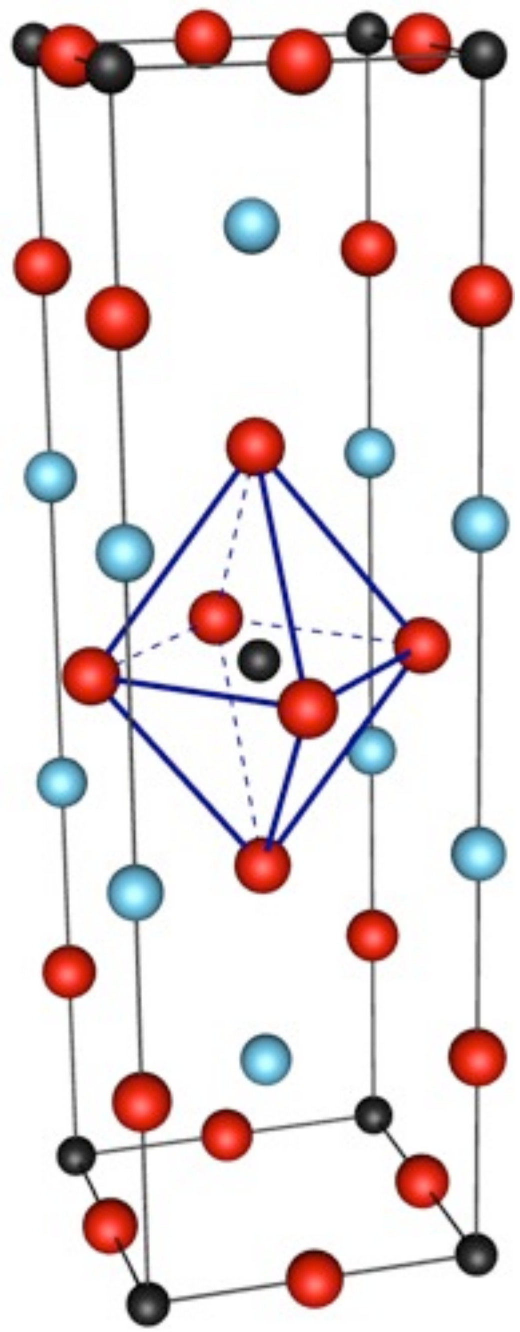
$$\chi^{KS}_{\mathbf{G}\mathbf{G}'}(\mathbf{q},\omega) = \frac{1}{N_k \Omega} \sum_{\mathbf{k}}^{\mathbf{BZ}} \sum_{\beta} A^{\mathbf{k},\mathbf{q}}_{\beta\mathbf{G}} B^{\mathbf{k},\mathbf{q}}_{\beta\mathbf{G}'}(\omega)$$

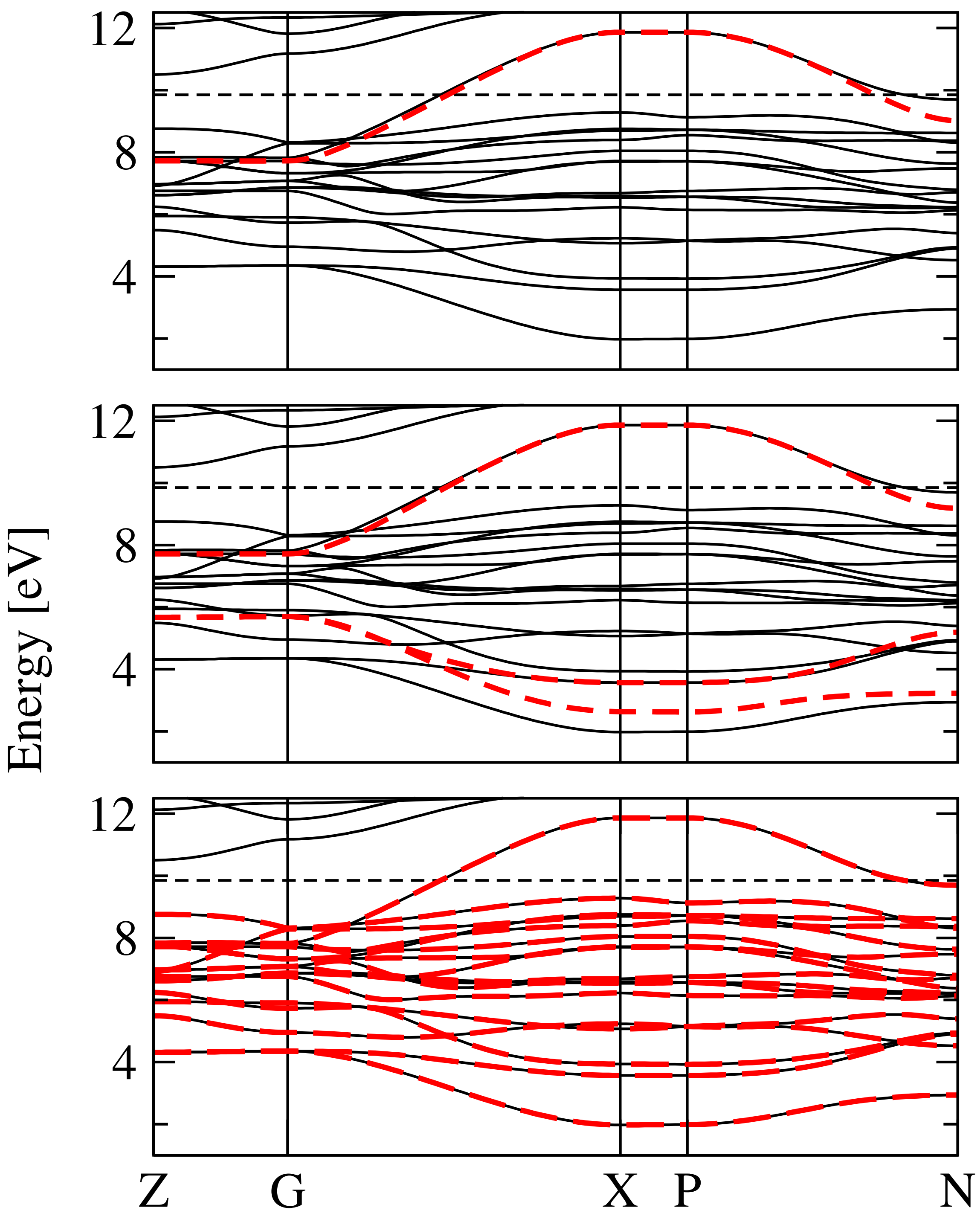


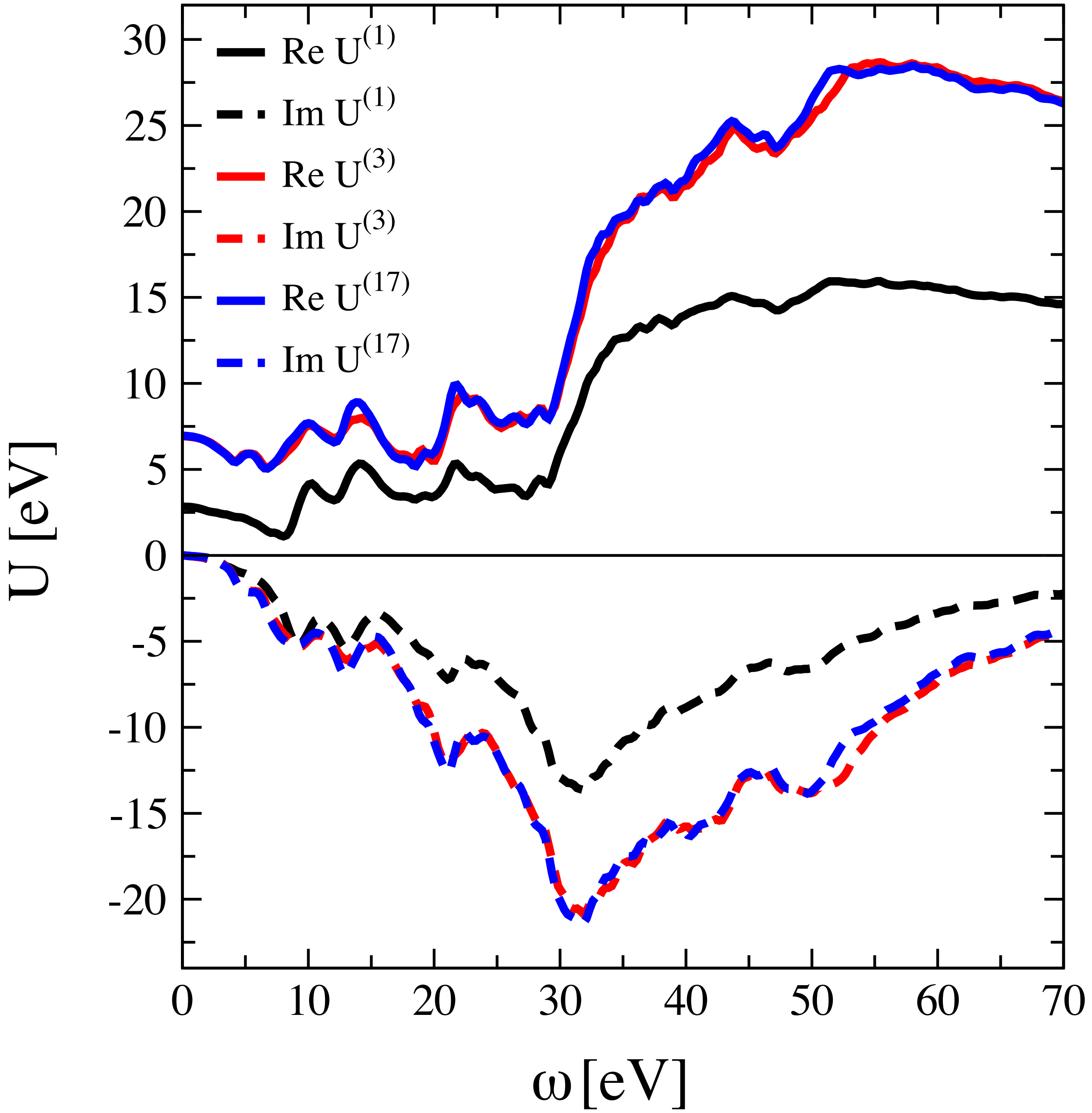
Execution time of DRC code computing W (and U) for La_2CuO_4

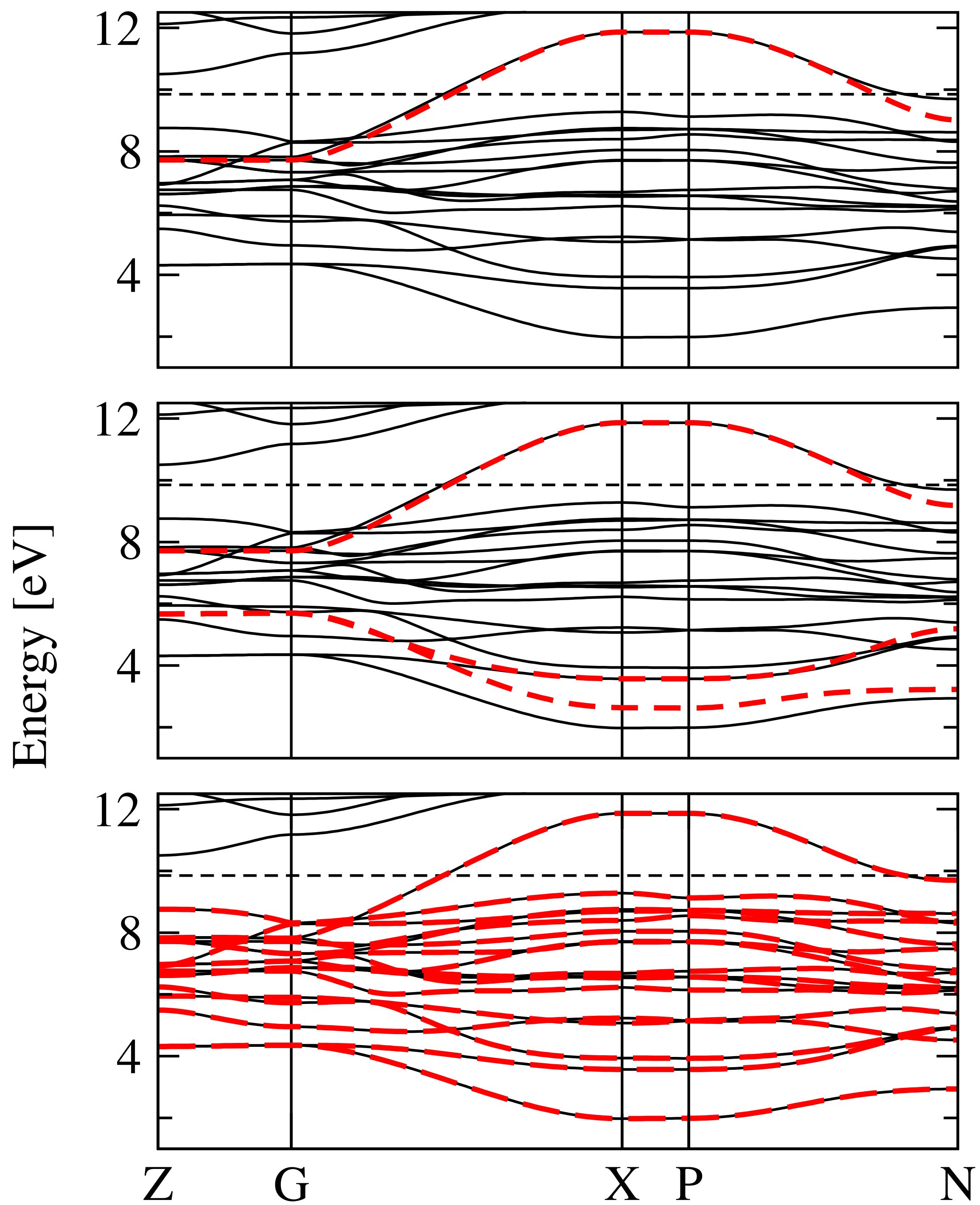


La_2CuO_4






Re $U^{(1)}$
Im $U^{(1)}$
Re $U^{(3)}$
Im $U^{(3)}$
Re $U^{(17)}$
Im $U^{(17)}$
U [eV]
ω [eV]


Energy [eV]
Z
G
X
P
N

Applications running at scale on Jaguar @ ORNL (Spring 2011)

Domain area	Code name	Institution	# of cores	Performance	Notes
Materials	DCA++	ORNL	213,120	**1.9 PF**	2008 Gordon Bell Prize Winner
Materials	WL-LSMS	ORNL/ETH	223,232	**1.8 PF**	2009 Gordon Bell Prize Winner
Chemistry	NWChem	PNNL/ORNL	224,196	1.4 PF	2008 Gordon Bell Prize Finalist
Materials	DRC	ETH/UTK	186,624	1.3 PF	2010 Gordon Bell Prize Hon. Mention
Nanoscience	OMEN	Duke	222,720	> 1 PF	2010 Gordon Bell Prize Finalist
Biomedical	MoBo	GaTech	196,608	780 TF	2010 Gordon Bell
Chemistry	MADNES				
Materials	LS3DF				
Seismology	SPECFEM				
Combustion	S3D	SNL	147,456	83 TF	
Weather	WRF	USA (multiple)	150,000	50 TF	

Behind each of these codes is a similar story of algorithmic re-engineering
> performance gains are useful on workstation and clusters as well!

Applications running at scale on Jaguar @ ORNL (Spring 2011)

Domain area	Code name	Institution	# of cores	Performance	Notes
Materials	DCA++	ORNL	213,120	**1.9 PF**	2008 Gordon Bell Prize Winner
Materials	WL-LSMS	ORNL/ETH	223,232	**1.8 PF**	2009 Gordon Bell Prize Winner
Chemistry	NWChem	PNNL/ORNL	224,196	1.4 PF	2008 Gordon Bell Prize Finalist
Materials	DRC	ETH/UTK	186,624	1.3 PF	2010 Gordon Bell Prize Hon. Mention
Nanoscience	OMEN	Duke	222,720	> 1 PF	2010 Gordon Bell Prize Finalist
Biomedical	MoBo	GaTech	196,608	780 TF	2010 Gordon Bell Prize Winner
Chemistry	MADNESS	UT/ORNL	140,000	550 TF	
Materials	LS3DF	LBL	147,456	442 TF	2008 Gordon Bell Prize Winner
Seismology	SPECFEM3D	USA (multiple)	149,784	165 TF	2008 Gordon Bell Prize Finalist
Combustion	S3D	SNL	147,456	83 TF	
Weather	WRF	USA (multiple)	150,000	50 TF	

Dynamics in COSMO-CCLM

Source: Oliver Fuhrer, MeteoSwiss

velocities

$$\frac{\partial u}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\frac{\partial E_h}{\partial \lambda} - vV_a\right\} - \dot{\zeta}\frac{\partial u}{\partial \zeta} - \frac{1}{\rho a\cos\varphi}\left(\frac{\partial p'}{\partial \lambda} - \frac{1}{\sqrt{\gamma}}\frac{\partial p_0}{\partial \lambda}\frac{\partial p'}{\partial \zeta}\right) + M_u$$

$$\frac{\partial v}{\partial t} = -\left\{\frac{1}{a}\frac{\partial E_h}{\partial \varphi} + uV_a\right\} - \dot{\zeta}\frac{\partial v}{\partial \zeta} - \frac{1}{\rho a}\left(\frac{\partial p'}{\partial \varphi} - \frac{1}{\sqrt{\gamma}}\frac{\partial p_0}{\partial \varphi}\frac{\partial p'}{\partial \zeta}\right) + M_v$$

$$\frac{\partial w}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial w}{\partial \lambda} + v\cos\varphi\frac{\partial w}{\partial \varphi}\right)\right\} - \dot{\zeta}\frac{\partial w}{\partial \zeta} + \frac{g}{\sqrt{\gamma}}\frac{\rho_0}{\rho}\frac{\partial p'}{\partial \zeta} + M_w + g\frac{\rho_0}{\rho}\left\{\frac{(T-T_0)}{T} - \frac{T_0 p'}{T p_0} + \left(\frac{R_v}{R_d} - 1\right)q^v - q^l - q^f\right\}$$

pressure

$$\frac{\partial p'}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial p'}{\partial \lambda} + v\cos\varphi\frac{\partial p'}{\partial \varphi}\right)\right\} - \dot{\zeta}\frac{\partial p'}{\partial \zeta} + g\rho_0 w - \frac{c_{pd}}{c_{vd}}pD$$

temperature

$$\frac{\partial T}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial T}{\partial \lambda} + v\cos\varphi\frac{\partial T}{\partial \varphi}\right)\right\} - \dot{\zeta}\frac{\partial T}{\partial \zeta} - \frac{1}{\rho c_{vd}}pD + Q_T$$

water

$$\frac{\partial q^v}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial q^v}{\partial \lambda} + v\cos\varphi\frac{\partial q^v}{\partial \varphi}\right)\right\} - \dot{\zeta}\frac{\partial q^v}{\partial \zeta} - (S^l + S^f) + M_{q^v}$$

$$\frac{\partial q^{l,f}}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial q^{l,f}}{\partial \lambda} + v\cos\varphi\frac{\partial q^{l,f}}{\partial \varphi}\right)\right\} - \dot{\zeta}\frac{\partial q^{l,f}}{\partial \zeta} - \frac{g}{\sqrt{\gamma}}\frac{\rho_0}{\rho}\frac{\partial P_{l,f}}{\partial \zeta} + S^{l,f} + M_{q^{l,f}}$$

turbulence

$$\frac{\partial e_t}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial e_t}{\partial \lambda} + v\cos\varphi\frac{\partial e_t}{\partial \varphi}\right)\right\} - \dot{\zeta}\frac{\partial e_t}{\partial \zeta} + K_m^v\frac{g\rho_0}{\sqrt{\gamma}}\left\{\left(\frac{\partial u}{\partial \zeta}\right)^2 + \left(\frac{\partial v}{\partial \zeta}\right)^2\right\} + \frac{g}{\rho\theta_v}F^{\theta_v} - \frac{\sqrt{2}e_t^{3/2}}{\alpha_M l} + M_{e_t}$$

Computationally this is a much simpler problem that solving Schrödinger equation!

Algorithmic motif: structured grid / finite difference stencils & tridiagonal solve

Algorithmic motifs and their arithmetic intensity

Arithmetic intensity: number of operations per word of memory transferred

Finite difference / stencil in S3D and WRF (& COSMO)

Rank-1 update in HF-QMC → **Rank-N update in DCA++**

DRC

QMR in WL-LSMS

Linpack (Top500)

Sparse linear algebra

Matrix-Vector

Vector-Vector — Fast Fourier Transforms — Dense Matrix-Matrix

BLAS1&2 — FFTW & SPIRAL — BLAS3

O(1) — **O(log N)** — **O(N)**

Supercomputers are designed for certain algorithmic motifs – which ones?

Conclusions

- Supercomputers work well for electronic structure based simulations
  - Very large numbers of atoms, accurate statistical sampling (not covered in this talk)
  - Pushing the limits in quantum many-body problem
  - Going beyond current state of the art in DFT simulations
- Efficient implementation of simulations requires algorithmic modifications
  - Computer architecture has to be considered when algorithms are developed!
  - Just porting a serial code does not lead to efficient simulations
- Improvements usually pay off at all scales, supercomputers and clusters
  - Jaguar and your laptop have similar processors
  - Improvements to both algorithms I discussed will impact efficiency of codes on your laptop as well
- When developing codes, consider
  - Modular/OO approach to manage data and complexity of code (same as before)
  - Break algorithms into a hierarchy of motifs, consider this hierarchy in implementation
  - Be prepared to change algorithms

Collaborators

- WL-LSMS: Chenggang Zhou, Markus Eisenbach, Don Nicholson (ORNL)
  Greg Brown (FSU), David Landau (UGA), Malcolm Stocks (ORNL)
- DCA++: Thomas Maier, Mike Summers, Gonzalo Alvarez, Paul Kent (ORNL)
  Peter Staar (ETH), Emanuel Gull (Columbia U.)
- DRC: Anton Koszevnikov (ETH) and Adolfo Eguiluz (U. of Tennessee)
- GPU: Jeremy Meredith and Jeff Vetter (ORNL)
- Applied math: Ed D'Azevedo and Phani Nukala (ORNL)
- Cray Inc.: Jeff Larkin and John Levesque
- NCCS: Markus Eisenbach, Don Maxwell + many others (ORNL)
- Doug Scalapino (UCSB) and Mark Jarrell (now at LSU)