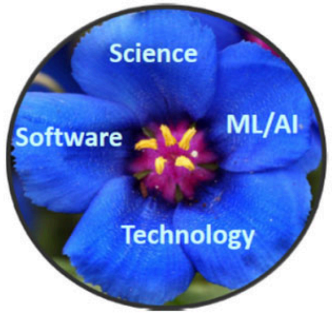
Scalable GW software for excited electrons using OpenAtom

Kavitha Chandrasekar, Eric Mikida, Eric Bohm and Laxmikant Kale
University of Illinois at Urbana-Champaign

Kayahan Saritas, Minjung Kim and Sohrab Ismail-Beigi
Yale University

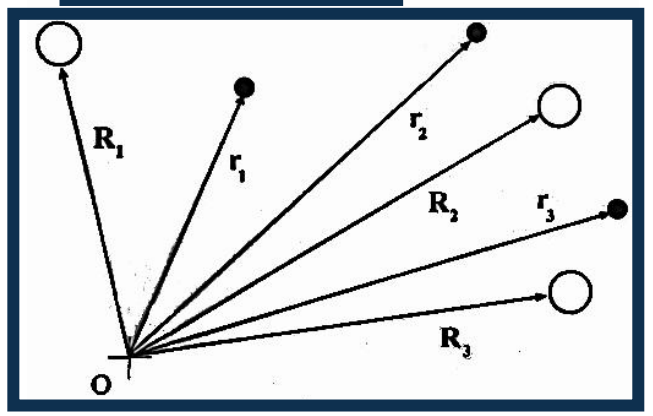
Glenn Martyna
Pimpernel Science, Software and Information Technology



Electronic structure calculations

- **Time independent Schrodinger equation for a *many-body* system**

$$i\hbar \frac{\partial}{\partial t} |\Psi(t)\rangle = \hat{H} |\Psi(t)\rangle$$



Many R_i & r_j

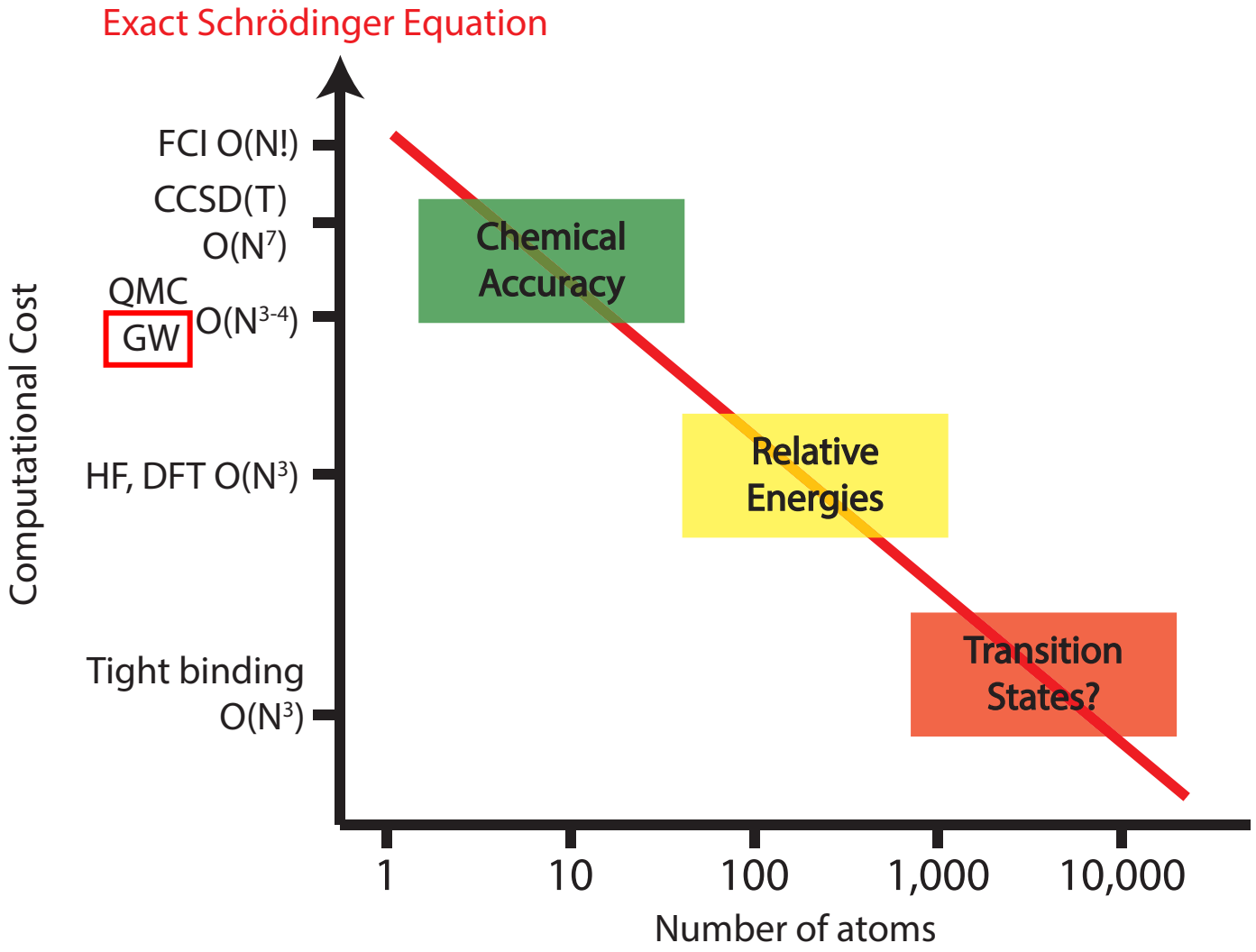
- **Density functional theory (DFT) simplifies this to one-body problem**

$$\left[-\frac{\nabla^2}{2} + V_{ion}(r) + V_H(r) + V_{xc}(r) \right] \psi_j(r) = \epsilon_j \psi_j(r)$$

$$V_{xc}(r) = \frac{\delta E_{xc}}{\delta n(r)}$$

Solve for wavefunctions $\psi_j(r)$ and energies ϵ_j

Comparison of the methods

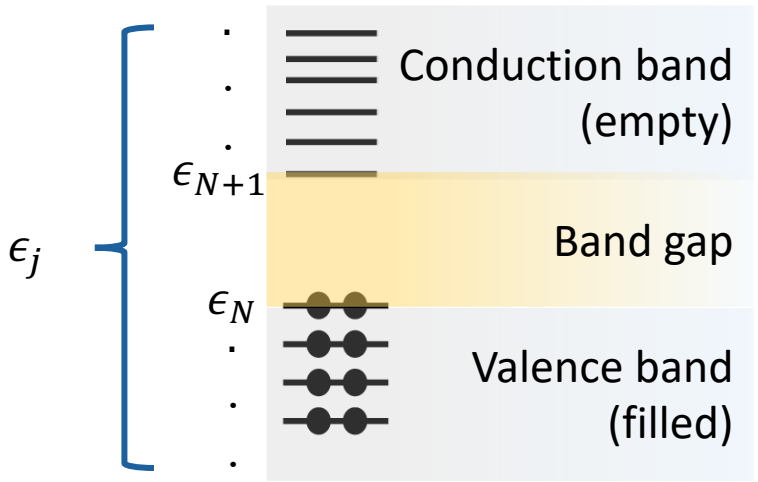


$$V_{xc}(r) = \frac{\delta E_{xc}}{\delta n(r)}$$

DFT problem with excitations

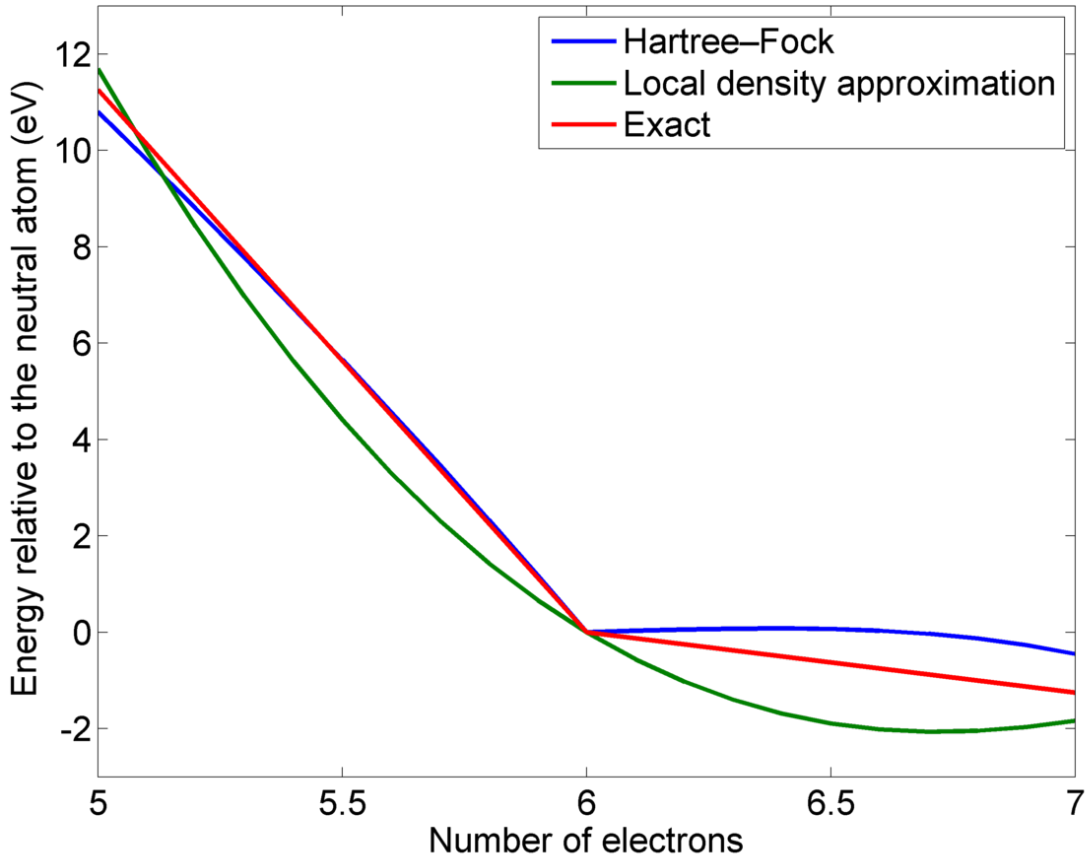
DFT: $$\left[-\frac{\nabla^2}{2} + V_{ion}(r) + V_H(r) + V_{xc}(r)\right]\psi_j(r) = \epsilon_j\psi_j(r)$$

ground state



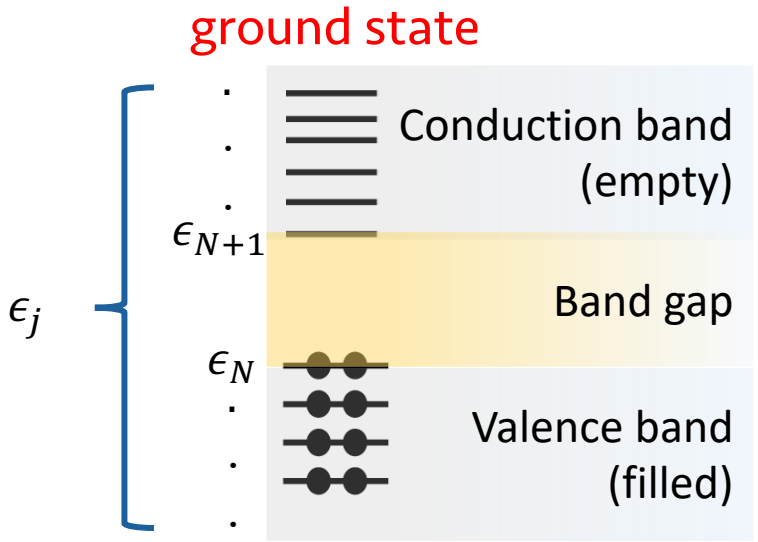
Janak's theorem

$$E_{gap} = \left.\frac{\partial E}{\partial N}\right|_{N+\delta} - \left.\frac{\partial E}{\partial N}\right|_{N-\delta} = \epsilon_{N+1} - \epsilon_N$$



DFT problem with excitations

DFT: $$\left[-\frac{\nabla^2}{2} + V_{ion}(r) + V_H(r) + V_{xc}(r)\right]\psi_j(r) = \epsilon_j\psi_j(r)$$



$$E_{gap} = \left.\frac{\partial E}{\partial N}\right|_{N+\delta} - \left.\frac{\partial E}{\partial N}\right|_{N-\delta} = \epsilon_{N+1} - \epsilon_N$$

Why band gap/excitations in a material is important?

- Metallic, semiconducting or insulating?
- Light-matter interactions in general
- A lot of engineering implications: PV, lasers, luminescence …

Band gaps (eV)

Material	DFT	GW	Expt.
Diamond	3.9	5.6*	5.48
Si	0.5	1.3*	1.17
LiCl	6.0	9.1*	9.4
$SrTiO_3$	2.0	3.4-3.8	3.25

GW method

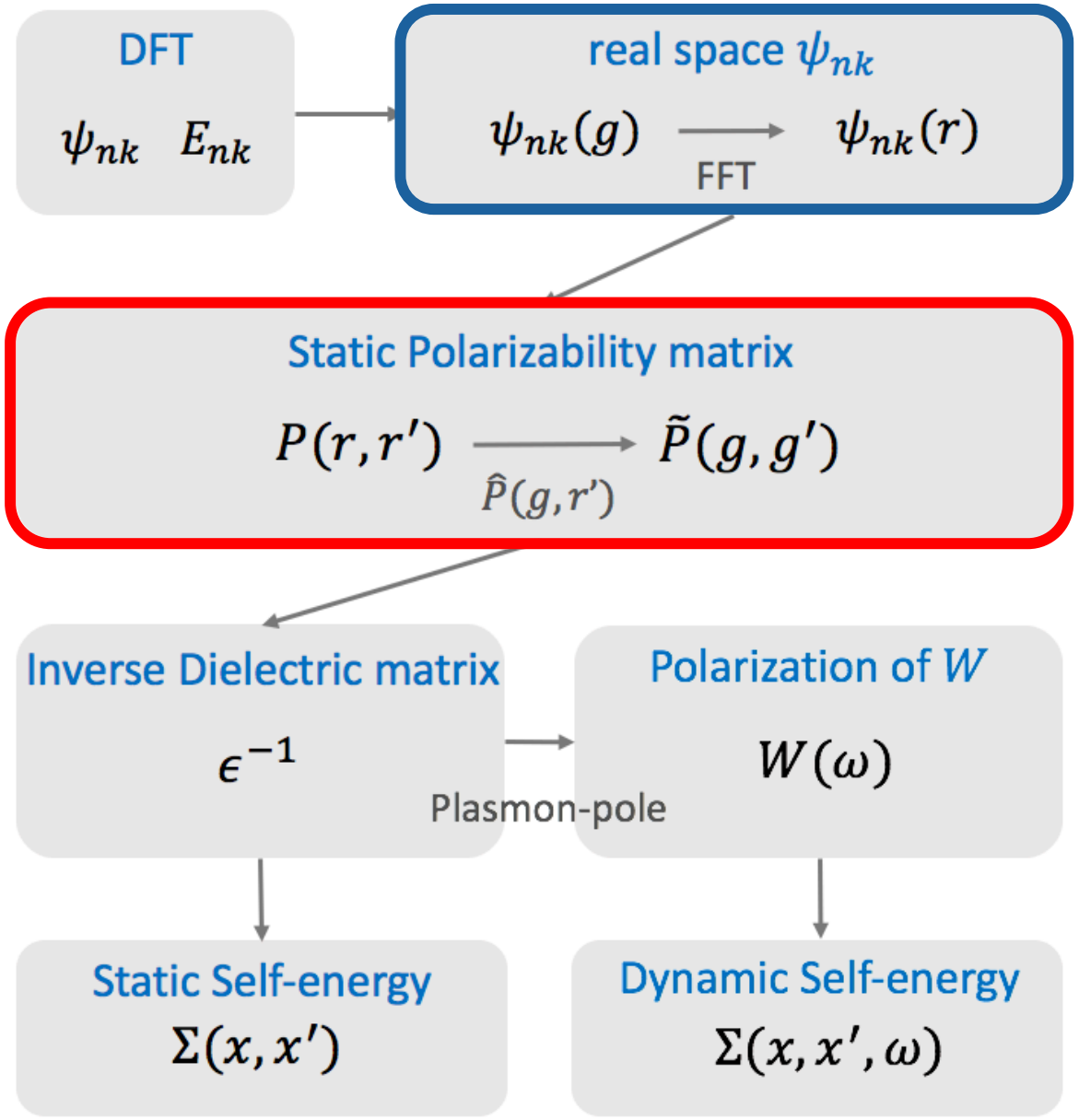
Challenges

- Memory intensive
- Much larger number of conduction bands: Huge number of FFTs
- Large and dense matrix multiplications
- Unfavorable scaling $O(N^4)$

Goal

- Efficient and highly scalable GW software
- $O(N^3)$ scaling method

What is expensive in GW?



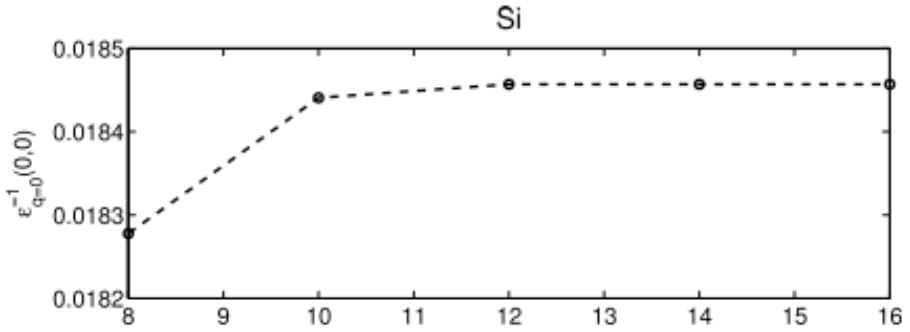
$\sim(N_c + N_v)N_r \ln N_r$

$$P(r,r') = -2 \sum_{v}^{\text{filled}} \sum_{c}^{\text{empty}} \frac{\psi_v(r)\psi_c(r)\psi_v(r')\psi_c(r')}{E_v - E_c}$$

$\sim(N_c N_v)N_r^2 = O(N^4)$

$\sim 2N_r^2 \ln N_r$

- Lots of FFTs to get $\psi_i(r)$ functions
- However, ϵ^{-1} can converge using a small r-grid



* Kim et al., (2020), Phys. Rev. B., 101, pp. 035139

O(N³) algorithm (CTSP) for P

CTSP: Complex time shredded propagator

$$P_{r,r'} = -2 \sum_{v}^{N_{occ}} \sum_{c}^{N_{unocc}} \frac{\psi^*_{r,v}\psi_{r,c}\psi^*_{r',c}\psi_{r',v}}{E_c - E_v}$$

$N_r^2 N_{unocc} N_{occ} \sim \mathbf{N^4}$

$$X_{r,r'} = \sum_{i}^{N_a} \sum_{j}^{N_b} \frac{A_{r,r'} B_{r,r'}}{w + a_i - b_j}$$

(1) Laplace transform:

$$\frac{1}{E_c - E_v} = \int_0^\infty e^{-(E_c - E_v)\tau} d\tau = \int_0^\infty e^{-E_c\tau} e^{E_v\tau} d\tau = \int_0^\infty f(\tau) e^{-\tau}\, d\tau$$

(2) Gauss-Laguerre quadrature:

$$\int_0^\infty f(\tau) e^{-\tau}\, d\tau \approx \sum_{k}^{N_q} \omega_k f(\tau_k)$$

$N_r^2 \mathbf{N_q} (N_{unocc} + N_{occ}) \sim \mathbf{N^3}$

$$N^4 \Rightarrow N^3$$

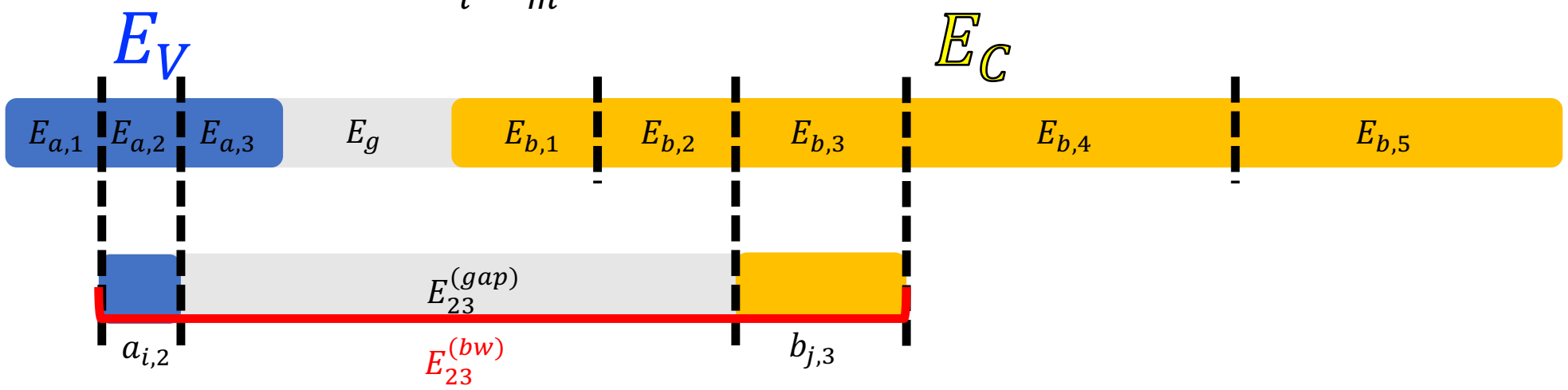
O(N³) algorithm (CTSP) for P

$$P_{r,r'} = -2 \sum_{v}^{N_{occ}} \sum_{c}^{N_{unocc}} \psi^*_{r,v} \psi_{r,c} \psi^*_{r',c} \psi_{r',v} \sum_{k}^{N_q} \omega_k \, f(\tau_k)$$

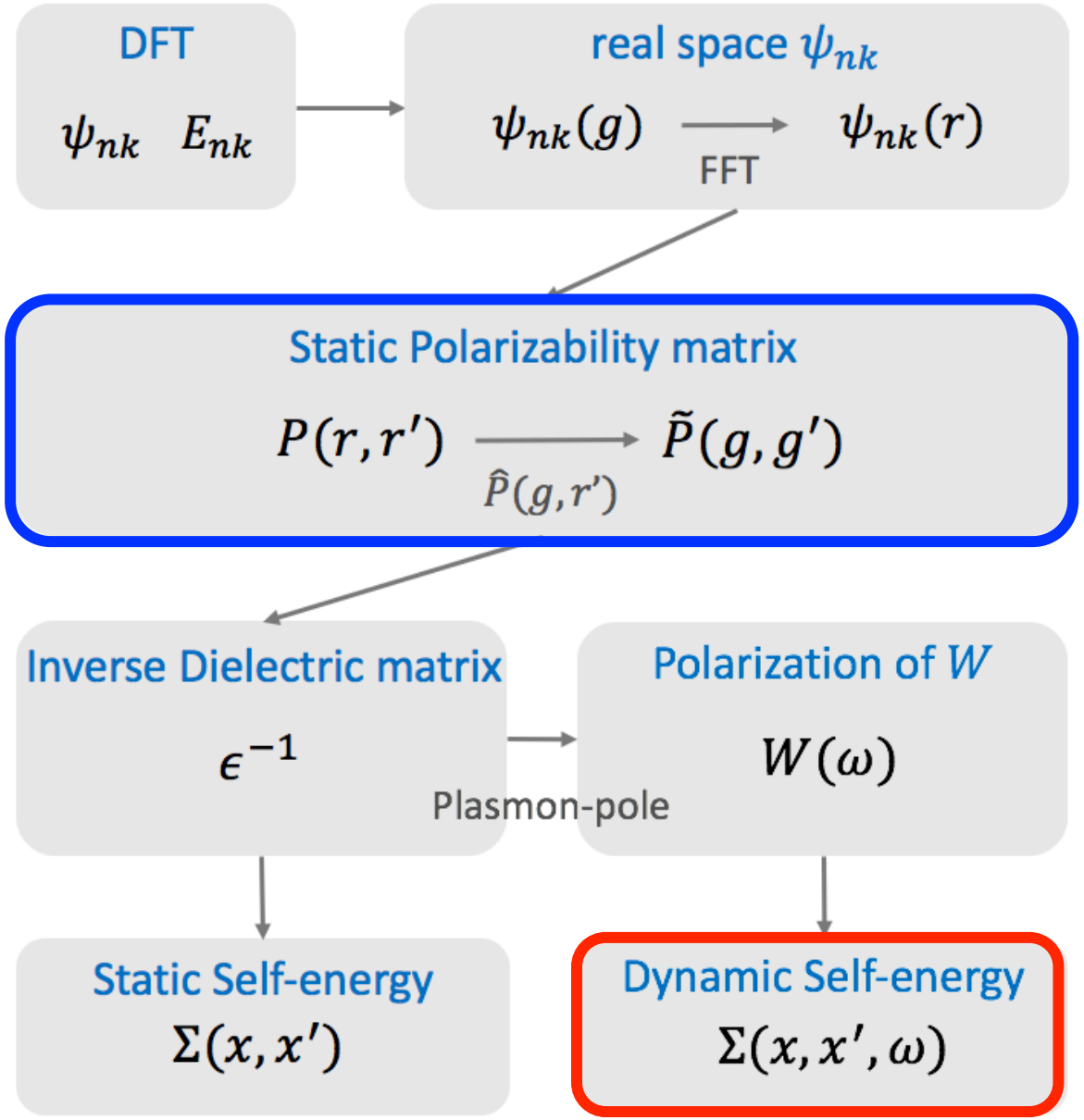
$$= \sum_{k}^{N_q} \omega_k \left[\sum_{v}^{N_{occ}} \psi_{r,v} \psi^*_{r',v} e^{E_v \tau_k} \right] \left[\sum_{c}^{N_{unocc}} \psi_{r,c} \psi^*_{r',c} \, e^{-E_c \tau_k} \right]$$

$N_q \, (N_{unocc} + N_{occ}) \, N_r^2 \sim N^3$

(3) Energy windows: $P_{r,r'} = \sum_{l}^{N_{aw}} \sum_{m}^{N_{bw}} P^{lm}_{r,r'}$



Steps for typical GW calculations



Most expensive

- Real-space P
- $O(N^3)$ method

Also expensive - $O(N^4)$

$O(N^3)$ method for self-energy

$$\Sigma^{\pm}(\omega)^{dyn}_{r,r'} = \sum_{p,n} \frac{B^p_{r,r'}\psi_{rn}\psi^*_{r'n}}{\omega - E_n \pm \omega_p}$$

$B^p_{r,r'}$: residues

ω_p: energies of the poles of $W(r)_{r,r'}$

$$X_{r,r'} = \sum_i^{N_a} \sum_j^{N_b} \frac{A_{r,r'}B_{r,r'}}{w + a_i - b_j}$$

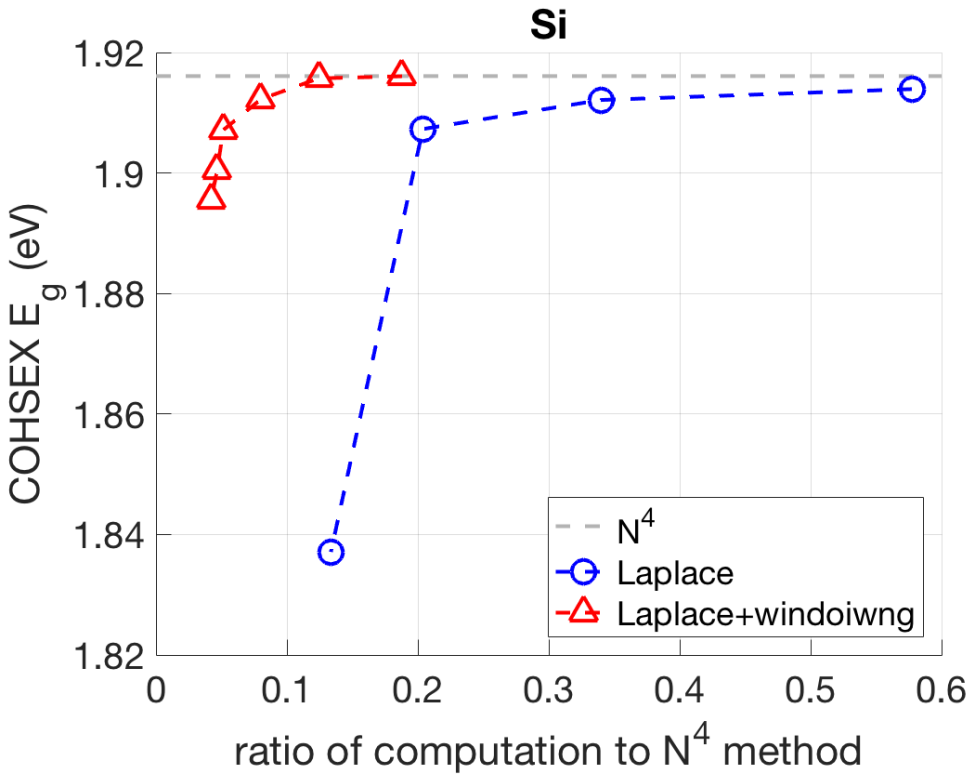
- $\omega - \epsilon_n \pm \omega_p = 0$ is possible: Gauss-Laguerre quadrature not applicable

- New quadrature is needed and was developed: Hermite-Gauss-Laguerre quadrature

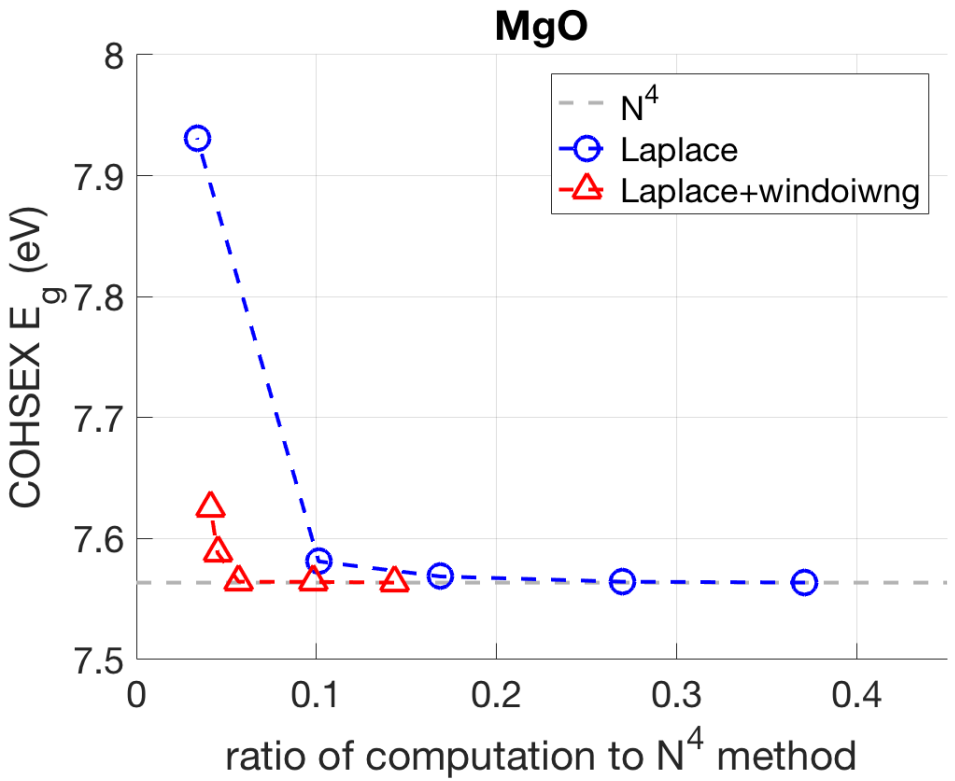
$$\frac{1}{\omega - E_n \pm \omega_p} = Im \int_0^\infty d\tau e^{-\tau - \tau^2/2} e^{i(\omega - E_n \pm \omega_p)\tau}$$

Results: Energy gap

- Si crystal (16 atoms)
- Number of bands: 399
- N_{wv}=1, N_{wc}=4



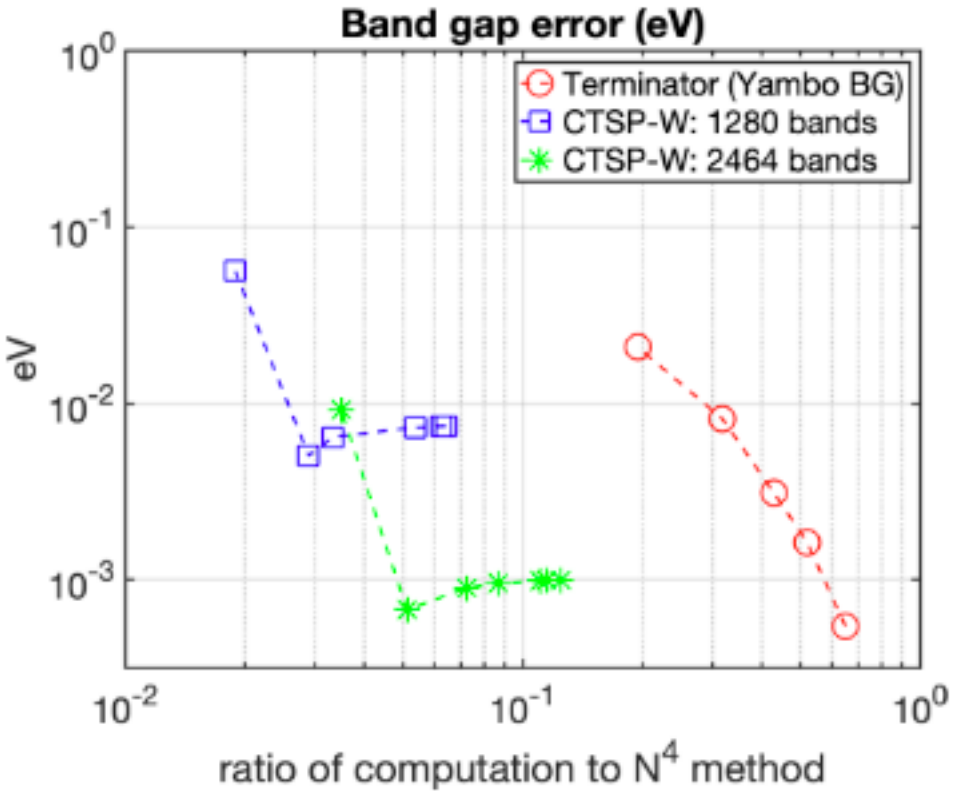
- MgO crystal (16 atoms)
- Number of bands: 433
- N_{wv}=1, N_{wc}=4



* Kim et al., (2020), Phys. Rev. B., 101, pp. 035139

Performance against other codes

- Si crystal (16 atoms)
- Number of bands: 399
- N_{pw}=15, N_{nw}=30



* Kim et al., (2019), Comput. Phys. Commun., 244, pp. 427-441

http://charm.cs.illinois.edu/OpenAtom/

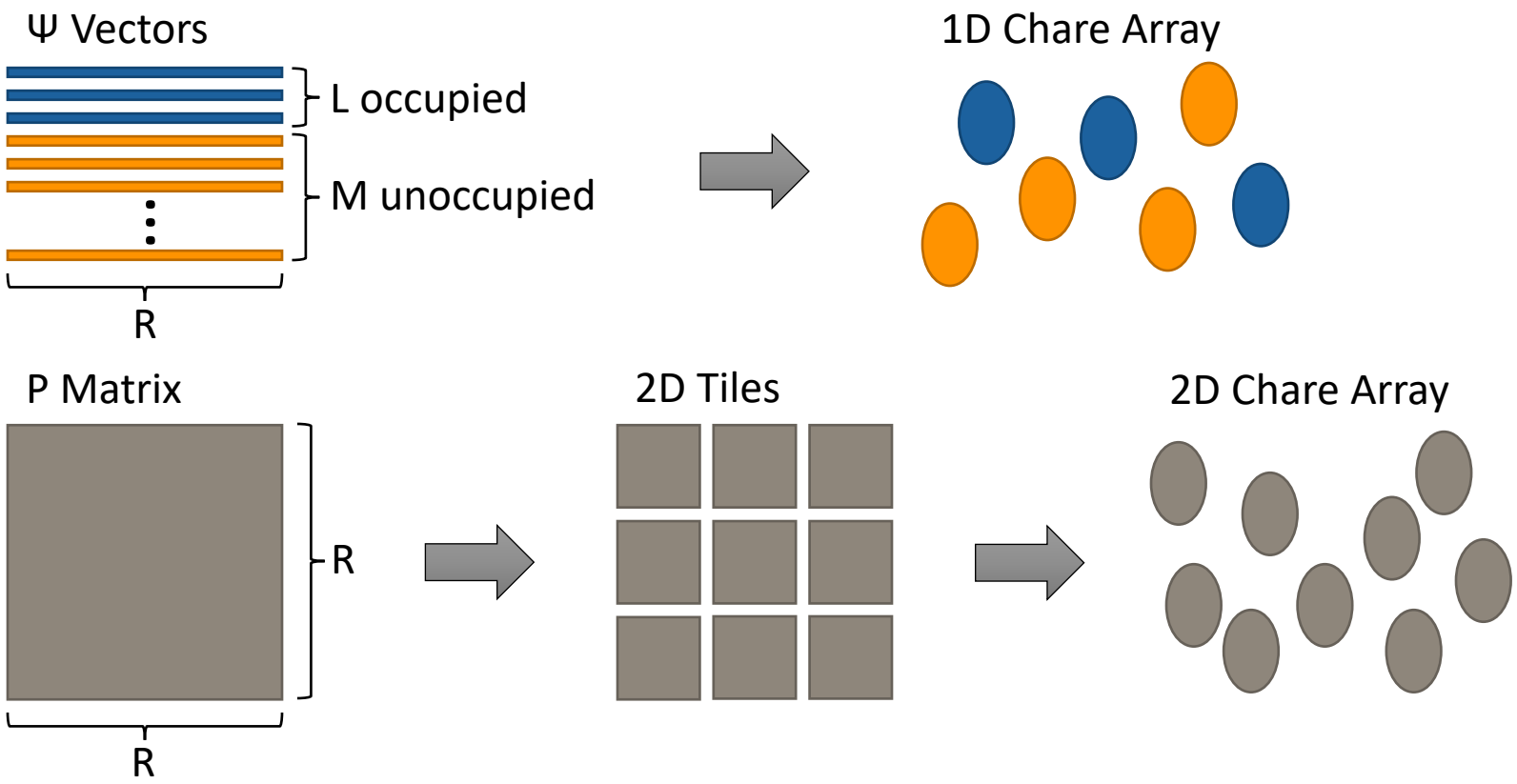
OpenAtom GW Parallel Scaling

OpenAtom Team

GW-BSE Parallelization

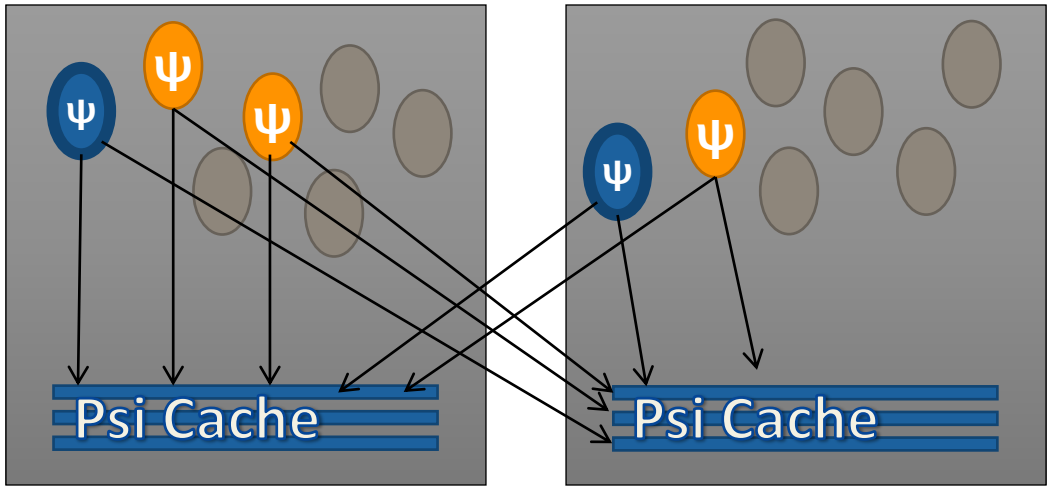
Phase		Serial	Parallel
1	Compute P in Rspace (N^4 and N^3 methods)	Complete	Complete
2	FFT P to GSpace	Complete	Complete
3	Invert epsilon	Complete	Complete
4	Plasmon pole	Complete	Future Work
5	COHSEX Self-energy	Complete	Complete
6	Dynamic Self-energy	Complete	Future Work

GW Phase-I
P Matrix Computation (N^4 and N^3 method)



Parallel Decomposition: Input state vectors

Duplicate occupied and unoccupied states on each node



Computation of Pmatrix using N^3 method

- Outer loops are windows of occupied and unoccupied states
- Most expensive computation - ρ and ρ'matrices

```
for l = 1:Nvw
   for m = 1:Ncw
      for j = 1:Nquad^lm
         calculate ρ^kqlmj
         calculate ρ'^kqlmj
         P[r,r'] += ρ^kqlmj[r,r'] x ρ'^kqlmj[r,r']
```

Computation ρ matrix (Using occupied states)

- State vectors are represented with ψ
  - Number of occupied states = L, each state has N elements
  - All occupied states can be represented as a matrix $\psi_V[1:L][1:N]$)

ρ^{kqlmj} -> Add elements of outer product of $\psi_V[1:L]$

```
for l=1:L
   for r=1:N
      for r'=1:N
         ρ^kqlmj [r,r'] += ψV [l] T[r] x ψV[l][r']
```

ρ^{kqlmj} -> Same as **ZGEMM** of all ψ_V and all ψ_V^T
ZGEMM ($\psi_V^T[1:N][1:L]$, $\psi_V[1:L][1:N]$)
(i.e matrix multiply)

```
for r=1:N
   for r'=1:N
      for l=1:L
         ρ^kqlmj [r,r'] += ψV^T[r][l] x ψV[l][r']
```

Computation ρ' matrix (Using unoccupied states)

- Number of unoccupied states = M, each state has N elements
- All unoccupied states can be represented as a matrix $\psi_C[1{:}M][1{:}N]$)

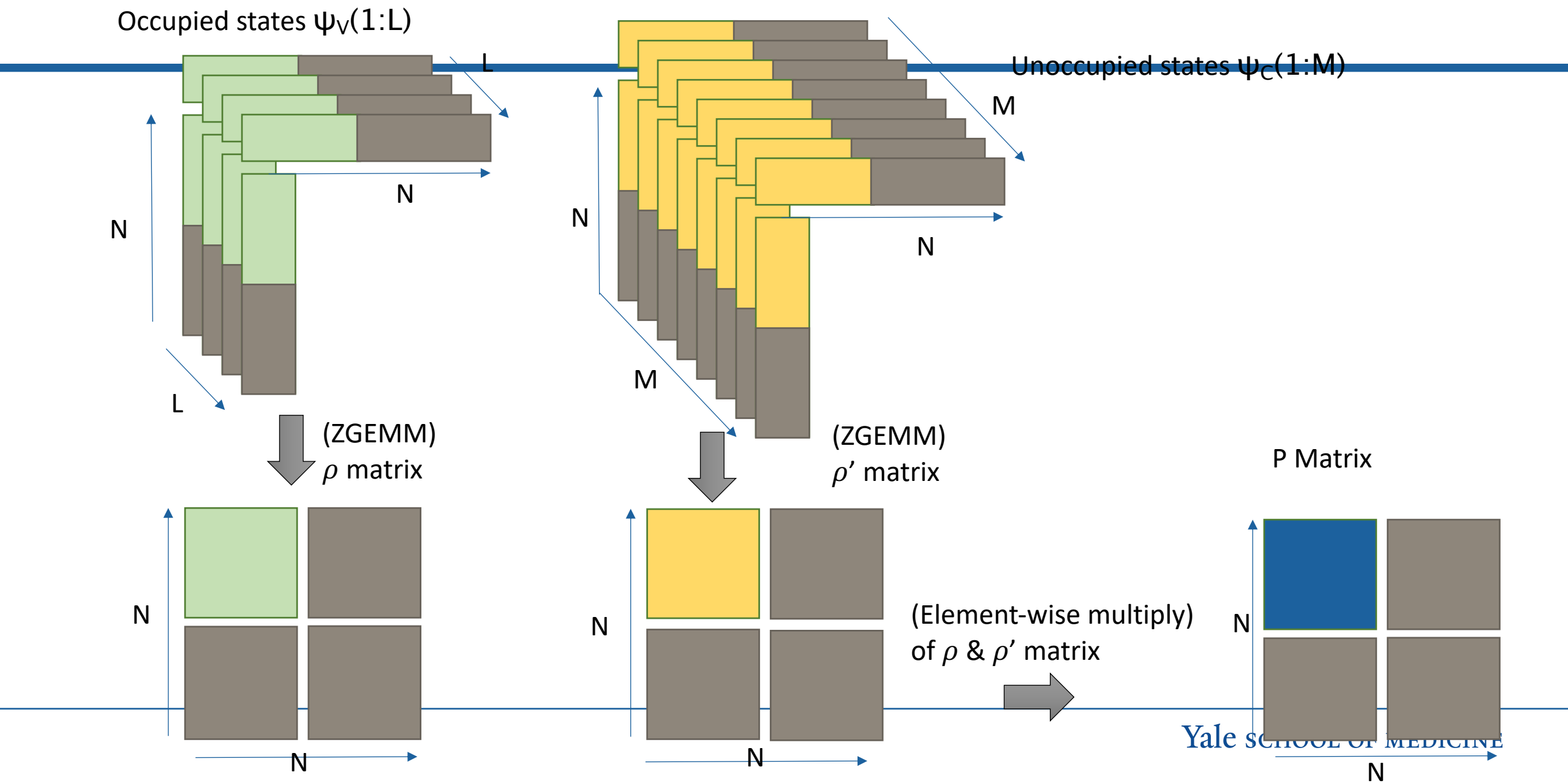
ρ^{kqlmj} -> Add elements of outer product of $\psi_C[1{:}M]$

```
for m=1:M
   for r=1:N
      for r'=1:N
         ρ'^kqlmj [r,r'] += ψC [m] T[r] x ψC[m][r']
```

ρ^{kqlmj} -> Same as **ZGEMM** of all ψ_C and all ψ_C^T
ZGEMM ($\psi_C^T[1{:}N][1{:}M]$, $\psi_C[1{:}M][1{:}N]$)
(i.e matrix multiply)

```
for r=1:N
   for r'=1:N
      for m=1:M
         ρ'^kqlmj [r,r'] += ψC^T[r][m] x ψC[m][r']
```

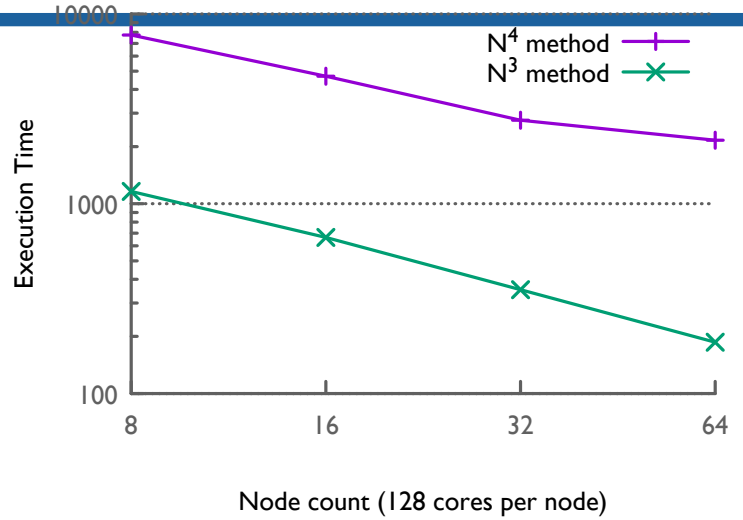
Computation of P-matrix (tiled) (N^3)



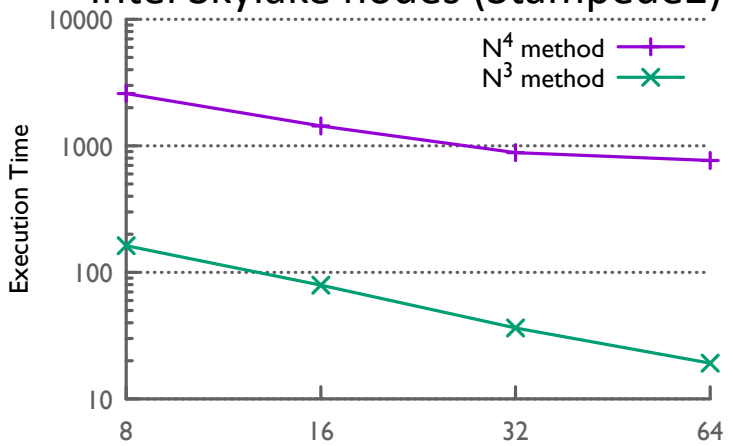
Performance of N^3 method

- N^3 method is an order faster than N^4 method for Si108 atoms dataset
  - 20k X 20k output matrix size
- Scales well on Intel KNL and SkyLake nodes
- Future scaling results for larger datasets

Intel KNL nodes (Stampede2)



Intel Skylake nodes (Stampede2)



Questions?