

# TCCM lectures – Advanced Computational Techniques

Peter Reinhardt

Laboratoire de Chimie Théorique, Sorbonne Université, 75252 Paris CEDEX 05,

[Peter.Reinhardt@upmc.fr](mailto:Peter.Reinhardt@upmc.fr)

# Overview of computational methods

- Hartree-Fock to start with ... or Hückel?
- Multiconfigurational SCF
- Perturbation theory for electron correlation
- CISD and derivates (CEPA, ACPF)
- Coupled-Cluster Theory
- Full CI
- Density-Functional methods
- Atoms-In-Molecules, Non-Covalent-Interactions etc
- Quantum Monte-Carlo methods
- Quantum chemistry for periodic systems
- Molecular dynamics, potential surfaces, finite-temperature methods ...

What are the bottle-necks? What are sources of errors?

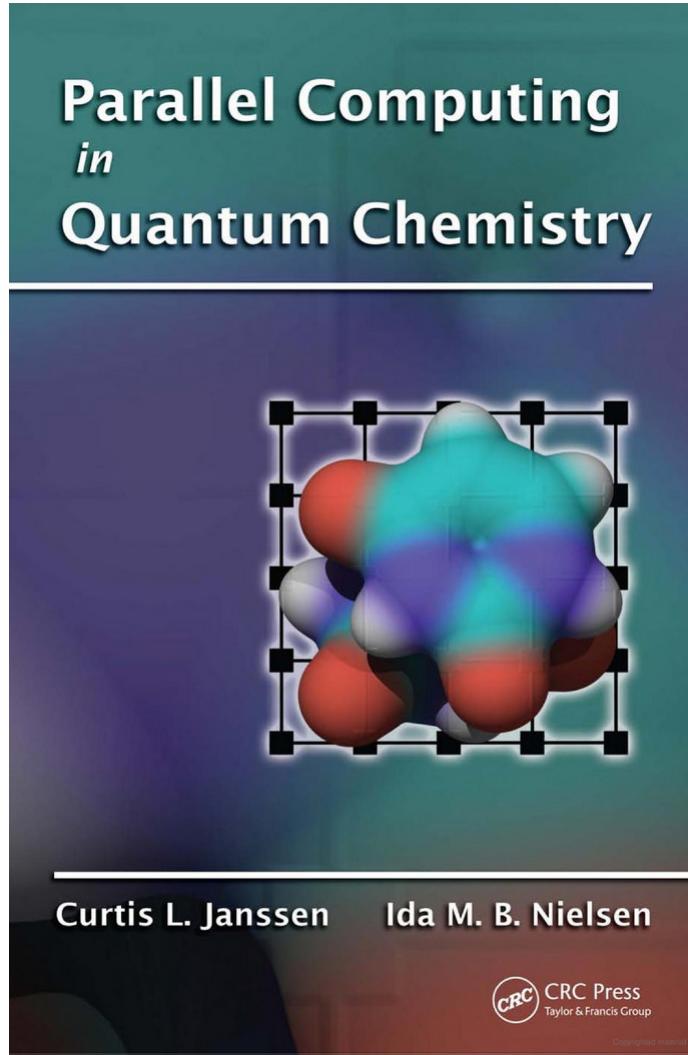
# Overview of computational methods

## Ingredients

- Atom-centered basis sets
- Precalculated integrals for one- and two-electron operators
- Storage and diagonalization of operator matrices
- Storage and matrix elements between excited determinants
- Matrix and vector operations in high dimensions
- Four-index integral transformation  $(\alpha\beta|\gamma\delta) \rightarrow (ij|kl)$
- Scaling well beyond  $N$  or  $N \log N$
- Random numbers
- Algebra in the complex plane (periodic systems)
- Numerical integration on grids
- Numerical interpolation in multidimensional spaces

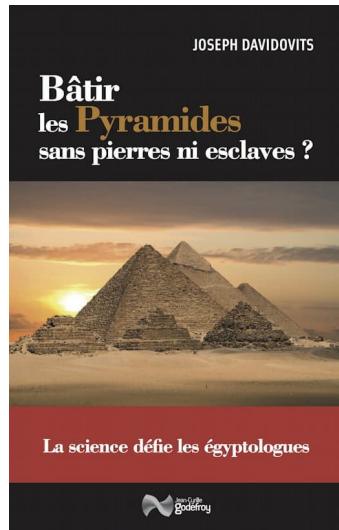
# Overview of computational methods

What can be parallelized on 100 processors, on 10 000 processors?



# Overview of computational methods

- Calculation of integrals?  $N^4 \longrightarrow 100$  procs
- Fock matrix elements?  $N^2$ , 10 procs
- Double excitations?  $n_o^2 N_v^2$
- Matrix elements between double excitations?  $N^6$ , 100 procs, but ...
- Matrix elements between triple excitations?  $N^8$
- Strange situation .... known already in ancient Egypt, 4500 years ago



Way out: Localized orbitals or Density-Functional Theory !

# Overview of computational methods

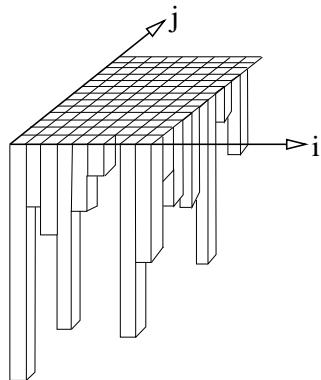
Integral-driven or Configuration-driven ?

$$\langle \Phi_{ij}^{ab} | \hat{H} | \Phi_{ij}^{cd} \rangle = (ac|bd) - (ad|bc)$$

Loop over configurations  $\Phi_{ij}^{ab}$  or over integrals  $(ab|cd)$ .

- In which category are less elements?
- CISD: many orbitals, but only 2 electrons → integral-driven
- CAS-CI: a few orbitals, but many configurations → configuration-driven

One may rearrange configurations with common indexes:



The matrix elements within each 1D “rod” need evaluation.

# Hash tables and bisection

- Bi-electronic integrals  $(ij|kl)$ , many are small or even zero, no need to store them.
- Canonical ordering:  $i \leq j, k \leq l, i \leq k$ , if  $i = k$  then  $j \leq l$  to avoid double storage
- Schwartz inequality:  $(ij|kl) \leq \sqrt{(ij|ij)(kl|kl)}$
- Calculate first the  $N^2$  integrals  $(ij|ij)$
- Maximum number of different integrals:

$$\frac{1}{8}N(N+1)(N(N+1)+2) = \frac{1}{8}N(N^3 + 2N^2 + 3N + 2)$$

- Exact address in memory is a complicated polynomial of 4th order in the first index
- Integrals may be cast into index classes, occ–virt, all different or not
- All integrals have to fit into memory.
- Storage with indexes or in an order?
- How to find a needed integral  $(ij|kl)$  rapidly?

# Hash tables and bisection

## Hash tables

- Integrals are stored with their indexes in a list
- Allocate a hash table ( $N$ , depth)
- For each integral create a number from the 4 indexes as a key
- Store the index of the integral at this place
- If a second index combination generates the same number, it is stored with a second index
- Rapid access  $\longleftrightarrow$  memory needs for the hash table
- No need for integral ordering, or definition of classes
- How to generate the key?
  - $\lambda(ijkl) = l + \alpha(k + \alpha(j + \alpha i))$  with a given  $\alpha$
  - position =  $\text{mod}(\lambda, N) + 1$
  - store in the last non-occupied depth
- 3rd-order polynomial = 3 multiplications + 3 additions

# Hash tables and bisection

Typical situation

```
NBAS =           46  NOCC =           4  NVIRT =           42
updating the hash table
      deepest hashing =           4

statistics of the hash table

    0      626804
    1      73162
    2      2360
    3       52
    4        1
    5        0
    6        0
    7        0
    8        0
    9        0

read in total      78042  integrals in core

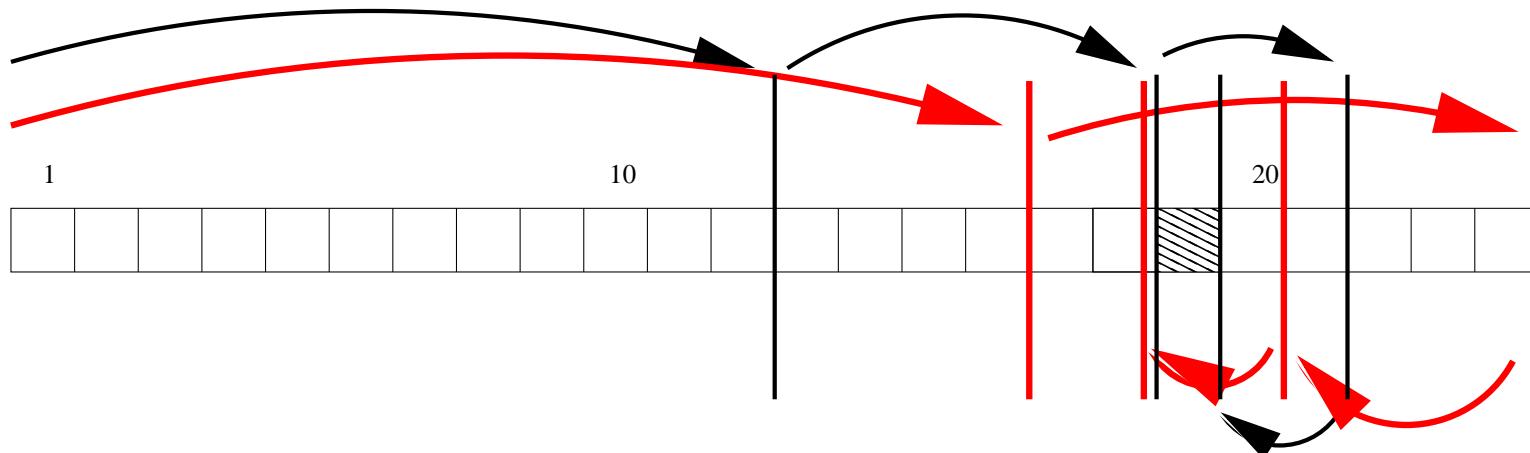
NUMBER OF INTEGRALS IN THE DIFFERENT CLASSES:
  TYPE      N1          N2
            (AABC)      (ABCD)
-----
  0000      22          0
  000V     415         84
  0V0V    1322        2713
  00VV    1373        1337
  0VVV    5157        37506
  VVVV   28113         0
```

For  $N = 46$  we would need 4 477 456 instead of the minimal 584 821 places.  
And we have “only” 78 042 relevant integrals to store (13 %).

# Hash tables and bisection

## Bisection

- Order data lexically
- Use for instance heapsort, no additional memory needed
- No need for additional tables neither
- Regroup integrals for reducing the search amplitudes, e.g. wrt to 1st index
- Start at  $N/2$ , look where your data should be, divide interval by 2 etc.
- For  $2^n$  data an item is found in  $n$  steps.
- Need for  $n$  steps as well to find that an integral is not present



# Full CI – why not?

- We have a set of molecular orbitals  $\{\phi_i(\vec{r})\}$  and determinants  $\{\Phi_I\}$
- Full CI means that we run over all orbitals and all determinants
- Look for the lowest eigenvalue of the matrix  $H_{IJ} = \langle \Phi_I | \hat{H} | \Phi_J \rangle$
- What is the action of  $\hat{H}$  on a given wavefunction  $\Psi = \sum_J c_J \Phi_J$ ?
- We write  $\hat{H}|\Psi\rangle = \sum_I c'_I |\Phi_I\rangle = \sum_J c_J \hat{H}|\Phi_J\rangle$  or

$$c'_I = \sum_J c_J \langle \Phi_I | \hat{H} | \Phi_J \rangle$$

- The matrix elements of  $\hat{H}$  may be written as

$$c'_I = \sum_{tu} \tilde{h}_{tu} \sum_J c_J A_{tu}^{IJ} + \frac{1}{2} \sum_{tuvx} (tu|vx) \left[ \sum_J c_J \left( \sum_K A_{tu}^{IK} A_{vx}^{KJ} \right) - \delta_{uv} A_{tx}^{IJ} \right]$$

- Generator matrix elements  $A_{tu}^{IJ} = \langle \Phi_I | \hat{E}_{tu} | \Phi_J \rangle$  (destroy an electron in orbital  $\phi_u$  and create one in  $\phi_t$ , very sparse matrix!)

# Full CI – why not?

- Auxiliary matrices

$$D_{tu}^K = \sum_J c_J A_{tu}^{KJ}$$

$$E_{tu}^K = \sum_{vx} (tu|vx) D_{vx}^K$$

- Final expression

$$c'_I = \sum_{tu} \left\{ \left( \tilde{h}_{tu} - \frac{1}{2} \sum_r (tr|ru) \right) D_{tu}^I + \frac{1}{2} \sum_K A_{tu}^{IK} E_{tu}^K \right\}$$

- Matrices  $D$  and  $E$  with 3 indices, matrix  $A$  very sparse and precalculated
- Ready for iterative solution

$$|\Psi\rangle \rightarrow \hat{H}|\Psi\rangle \rightarrow |Q\rangle = (\hat{H} - \langle \Psi | \hat{H} | \Psi \rangle) |\Psi\rangle \rightarrow \dots$$

# Perturbation theory

$$E_0^{(1)} = \langle \Phi_0 | V | \Phi_0 \rangle = \langle 0 | \hat{V} | 0 \rangle = \langle 0 | \hat{H} - \hat{H}_0 | 0 \rangle$$

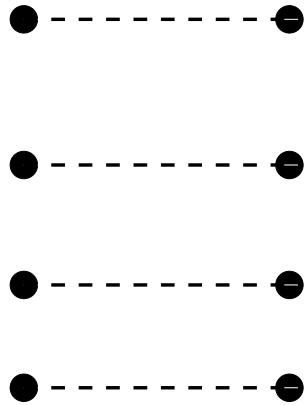
$$E_0^{(2)} = \langle \Phi_0 | V | \Psi^{(1)} \rangle = \sum_{k \neq 0} \langle 0 | \hat{V} \frac{|k\rangle\langle k|}{E_0^{(0)} - E_k^{(0)}} \hat{V} | 0 \rangle$$

$$\begin{aligned} E_0^{(3)} &= \langle \Phi_0 | V | \Psi^{(2)} \rangle \\ &= \sum_{k,l \neq 0} \langle 0 | \hat{V} \frac{|k\rangle\langle k|}{E_0^{(0)} - E_k^{(0)}} \hat{V} \frac{|l\rangle\langle l|}{E_0^{(0)} - E_l^{(0)}} \hat{V} | 0 \rangle \\ &\quad - E_0^{(1)} \sum_{k \neq 0} \left( \frac{\langle 0 | V | k \rangle}{E_0^{(0)} - E_k^{(0)}} \right)^2 \end{aligned}$$

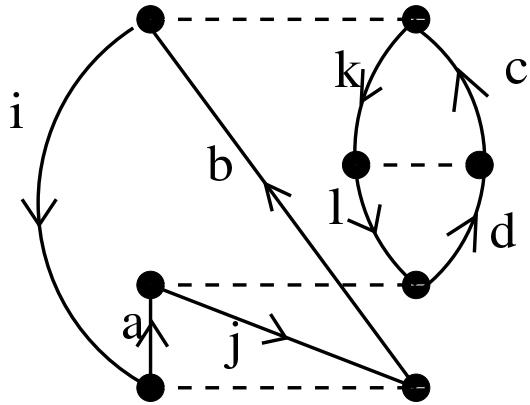
There is a systematic structure in the equations ...

# Perturbation theory

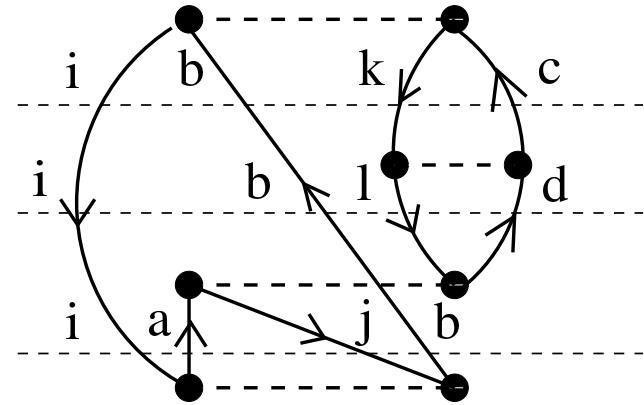
Graphical approach: a 4-th order diagram as example



(1)



(2)

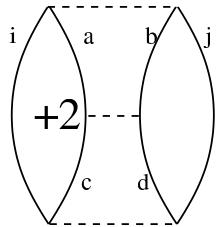


(3)

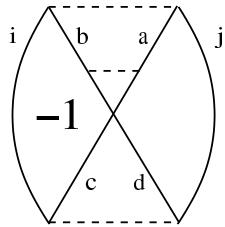
$$\sum_{ijkl} \sum_{abcd} (-1)^{2+4} 2^2 \frac{(ib|kc)(kl|cd)(ja|ld)(ia|jb)}{(\epsilon_i + \epsilon_k - \epsilon_b - \epsilon_c)(\epsilon_i + \epsilon_l - \epsilon_b - \epsilon_d)(\epsilon_i + \epsilon_j - \epsilon_a - \epsilon_b)}$$

# Perturbation theory

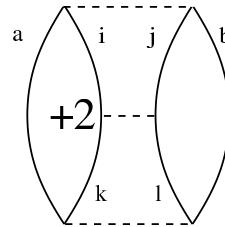
All third-order diagrams



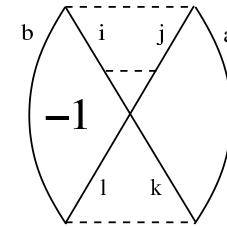
(1)



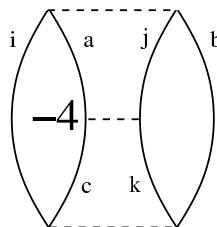
(2)



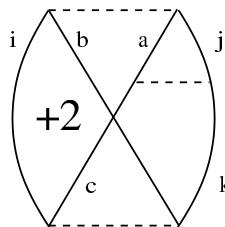
(3)



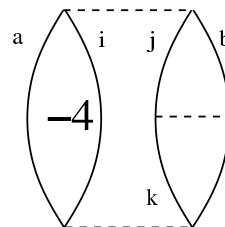
(4)



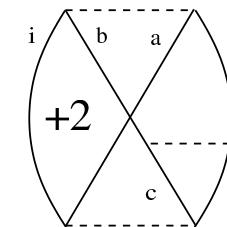
(5)



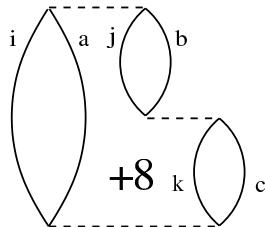
(6)



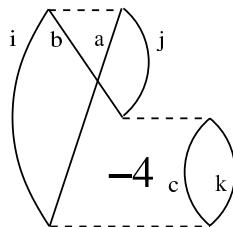
(7)



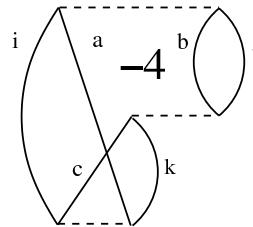
(8)



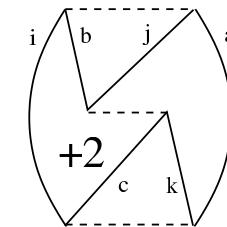
(9)



(10)



(11)



(12)

# Density Functional Theory

Integration grids needed for numerical integration of the functionals

- Spherical around atoms
- Logarithmic radial grids
- Space-filling between atoms



Chemical Physics Letters  
Volume 209, Issues 5–6, 16 July 1993, Pages 506-512



---

A standard grid for density functional calculations

Peter M.W Gill<sup>1</sup>✉, Benny G Johnson, John A Pople

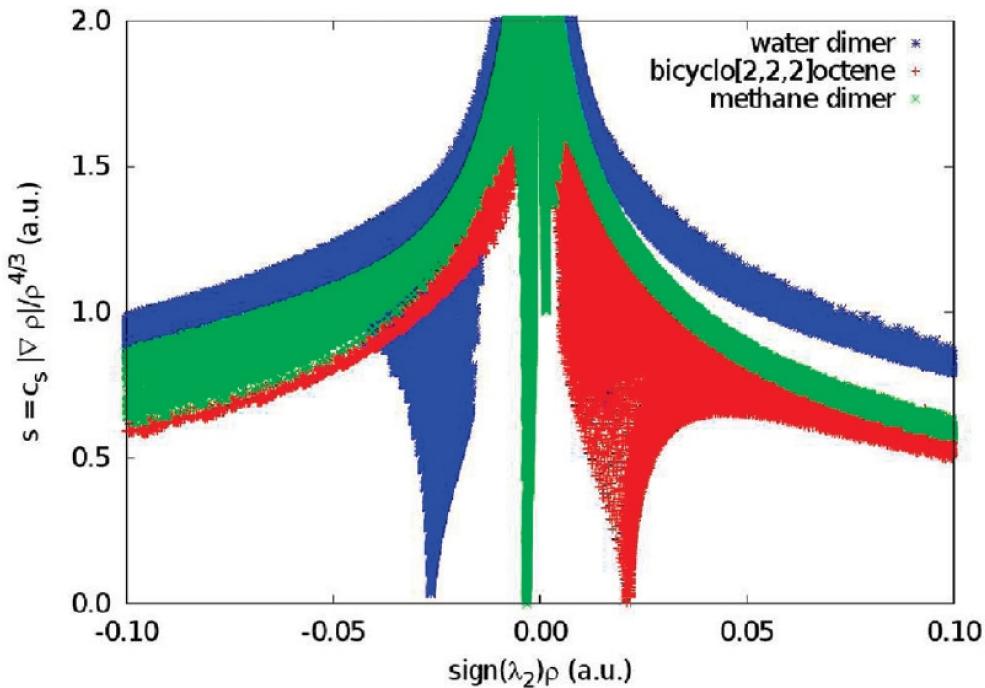
Department of Chemistry, Carnegie Mellon University, Pittsburgh, PA 15213,  
USA

# AIM, ELF, NCI etc

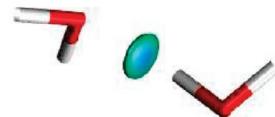
Evaluation of orbitals, densities etc on grids

- Domain boundaries difficult to localize in space
- Description of hypersurfaces, e.g.  $\Delta\rho = 0$
- Huge data volumes, however good compressibility
- Data should be plotted in 3D: .cube format

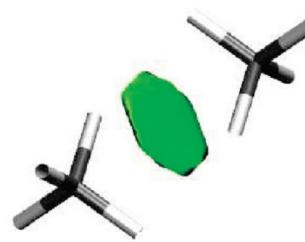
# AIM, ELF, NCI etc



Repulsive



H bond



Dispersion

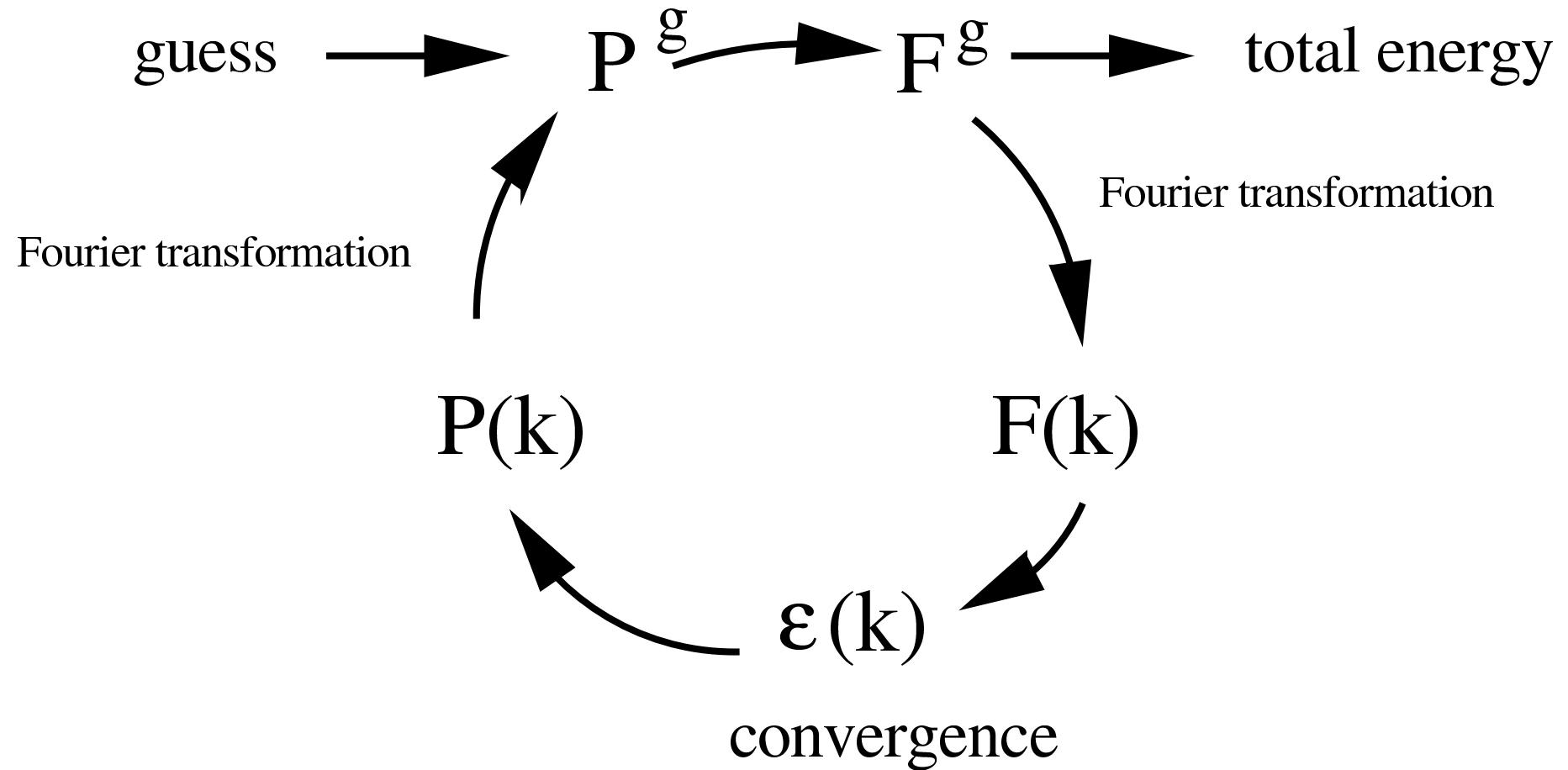
# Quantum Chemistry for periodic systems

Infinite summations over unit cells

- Electrostatic interactions
  - nuclear-nuclear: repulsive
  - nuclear-electron: attractive
  - electron-electron: repulsive

has to sum to a finite value per unit cell
- Exchange interactions have to converge on their own
- additional index for quantities needed: cell vector
- Fourier transform for  $k$ -space: complex numbers via  $e^{ik \cdot g}$
- construct Fock matrix in real space, transform to  $k$ -space, diagonalize, transform density matrix back to real space
- parallel computation by symmetry of crystals and  $k$  points

# Quantum Chemistry for periodic systems



SCF-scheme for a Hartree-Fock program for periodic systems (Pisani/Del Re 1967)

# Molecular dynamics

Classical dynamics with an ab-initio potential surface

- Initial geometry of  $N$  atoms in space, initial positions and momenta given
- Potential surface in  $3N - 6$  dimensions given
- Calculate forces on atoms as  $\vec{F}_i = -\vec{\nabla}_i V(\vec{r}_1, \dots, \vec{r}_N)$
- Calculate new positions and momenta after an acceleration period of  $\Delta t$
- Recalculate forces, recalculate displacements etc

Sources of “errors”

- Approximate potential surfaces
- Neglect of intrinsic quantum effects
- Discrete time steps
- Integration algorithms (Euler, Runge-Kutta, Cash-Karp, Verlet, etc)
- Limited number of trajectories of limited length in time for statistical treatment

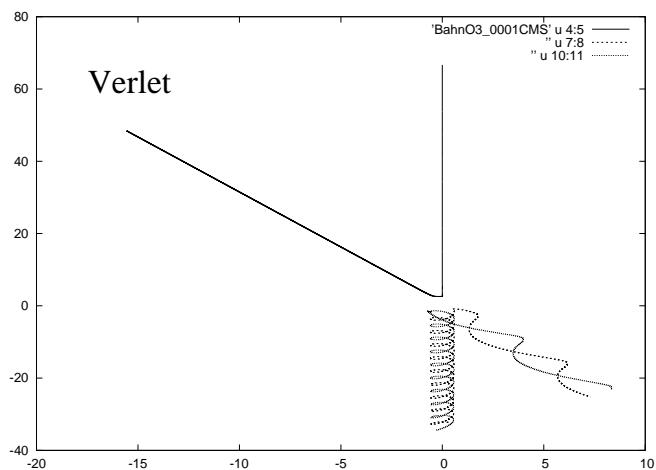
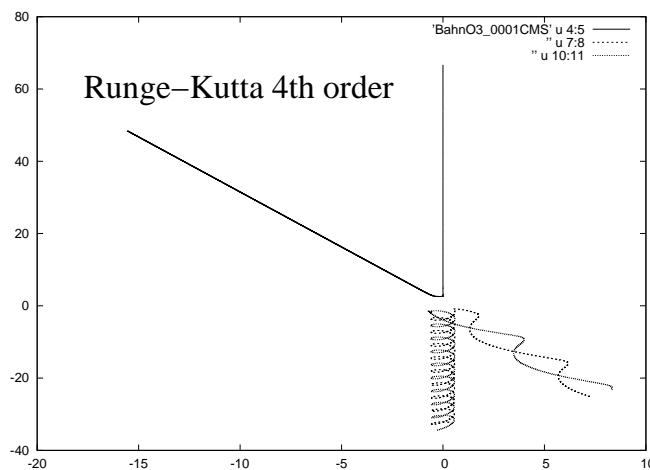
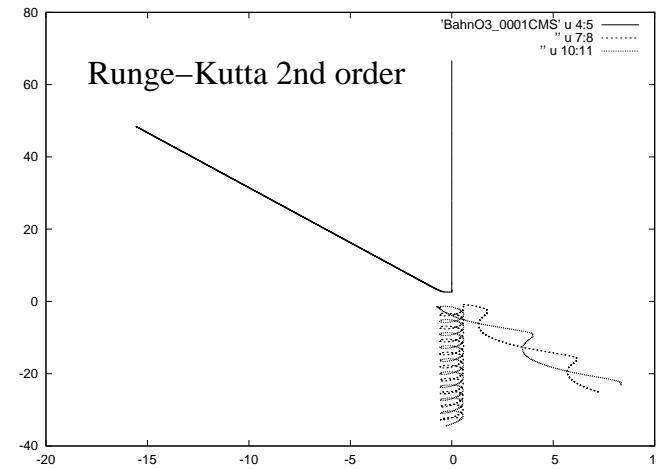
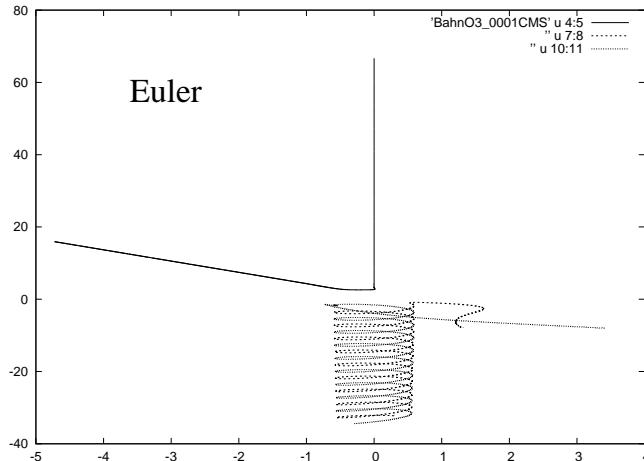
# Molecular dynamics

## Control mechanisms

- Correct total energy after each time step via kinetic energy
- Watch total angular momentum as conserved quantity of movement
- Try different time steps

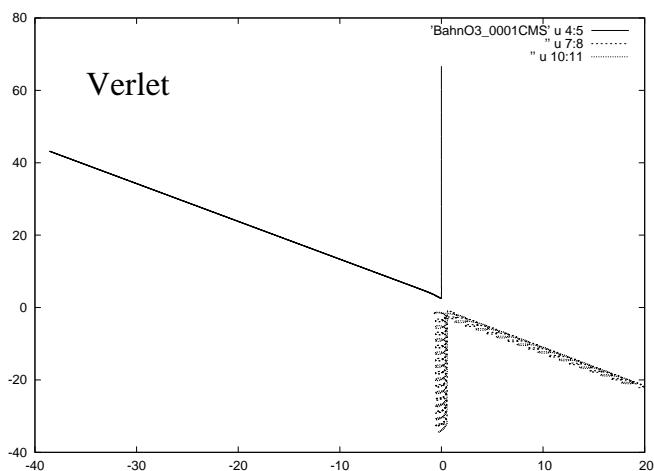
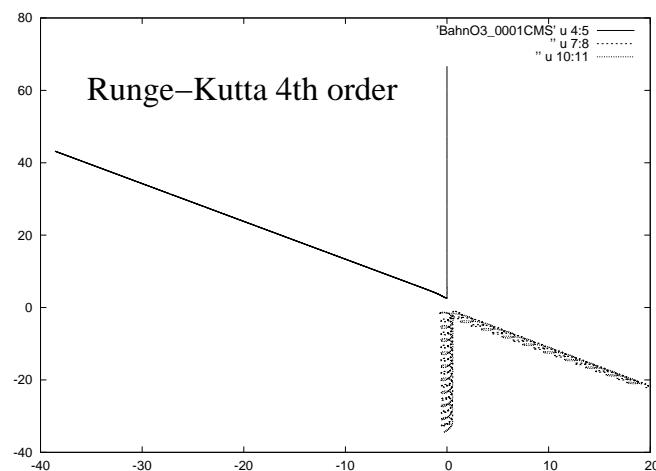
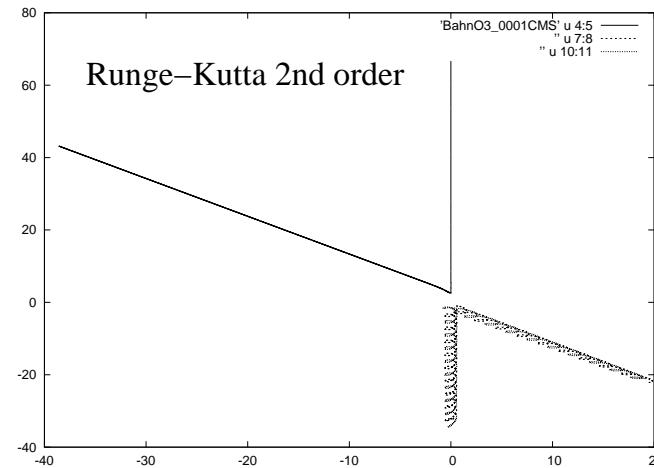
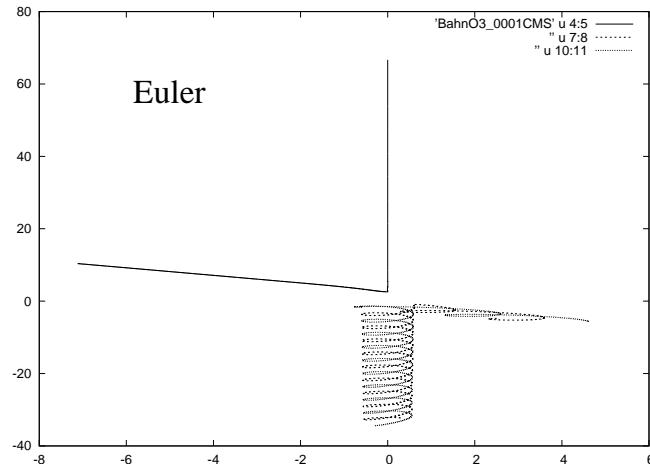
# Molecular dynamics

O + O<sub>2</sub> scattering – Schinke surface



# Molecular dynamics

O + O<sub>2</sub> scattering – Tyutereev surface



# Integrals and Determinants

MP2 and the 4-index transformation

- Sparse matrices
- Canonical orbitals and localized orbitals
- Orbital-invariant MP2
- The 4-index transformation
- Laplace-transformed MP2

# Sparse matrices

- Storage of an  $n \times n$  matrix (CRS = compressed row storage): three 1D arrays, one holding the number of non-zero elements in a row, one with the column indexes, and one with the matrix elements.
- Matrix-matrix operations, matrix-vector operations:

```
do i = 1, n
    y(i) = 0
    do j = row_ptr(i), row_ptr(i+1) - 1
        y(i) = y(i) + val(j) * x(col_ind(j))
    end do
end do
```

- Atomic orbitals are local, molecular orbitals are not
- The Hamilton matrix in the base of determinants is not really sparse neither
- Spin-free generator matrices  $\langle \Phi_I | \hat{E}_{ij} | \Phi_J \rangle$  are VERY sparse, and matrix elements are 2, 1, 0, or  $-1$ .

$$\hat{E}_{ij} = \hat{a}_{i,\alpha}^\dagger \hat{a}_{j,\alpha} + \hat{a}_{i,\beta}^\dagger \hat{a}_{j,\beta}$$

- See for instance the package `librsb-dev` on Ubuntu

# Sparse matrices

Example

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 3 \\ 0 & 1 & 2 & 0 & 0 \\ 1 & 0 & 2 & 0 & 7 \\ 0 & 0 & 0 & 4 & 0 \\ 1 & 2 & 3 & 0 & 3 \end{pmatrix}$$

The three 1D arrays:

$$R = (1, 3, 5, 8, 9, 12)$$

$$C = (1, 5, 2, 3, 1, 3, 5, 4, 1, 2, 3, 5)$$

$$V = (1, 3, 1, 2, 1, 2, 7, 4, 1, 2, 3, 3)$$

for example  $R(2) = 3$ , i.e. the 2nd row starts with the 3rd element of C (=2nd column with value 1)

# Sparse matrices

Encoding a matrix needs two passages:

1. run over the  $N$  rows and count the  $n$  non-zero elements.  $R(1) = 1$ ,  $R(N + 1) = n$ .  $R(i)$  is the number of non-zero elements found at the beginning of row  $i$ .
2. allocate the array  $C$  and  $V$ ; run again over the matrix elements and store the column index of the non-zero matrix elements.

Write then on the file:

1. Number of rows  $N$
2. the  $R$  array
3. the  $C$  array
4. the  $V$  array

For reading the first number allows to allocate the  $R$  array, and reading then the  $R$  array allows for allocating the  $C$  and the  $V$  array, all in one single read.

# Sparse matrices

The generator matrices

$$\langle \Phi_I | \hat{E}_{pq} | \Phi_J \rangle = \langle \Phi_I | \hat{a}_{p,\alpha}^\dagger \hat{a}_{q,\alpha} + \hat{a}_{p,\beta}^\dagger \hat{a}_{q,\beta} | \Phi_J \rangle$$

- Orbital  $q$  (and/or  $\bar{q}$ ) must be occupied in  $|\Phi_I\rangle$ , and orbital  $p$  (and/or  $\bar{p}$ ) empty
- From  $|\Phi_I\rangle$  the generator may produce either one or two determinants  $|\Phi_J\rangle$ , multiplied with  $\pm 1$  or  $\pm 2$
- Store in one array  $A(I, p, q)$  if there is a result
- Store in a second array  $S(I, p, q)$  whether there is a second possible result
- Store the results  $J$  in an array  $R(2, I, p, q)$  and the multiplicators  $M(2, I, p, q)$

How to calculate now  $c_I = \sum_J c_J \langle \Phi_I | \hat{h} | \Phi_J \rangle$  ?

# Sparse matrices

$$c_I = \sum_J c_J \sum_{pq} h_{pq} \langle \Phi_I | \hat{E}_{pq} | \Phi_J \rangle = \sum_{pq} h_{pq} \sum_J c_J \langle \Phi_I | \hat{E}_{pq} | \Phi_J \rangle$$

```
do p,q = 1,nbas
  calculate  $h_{pq}$ 
  do I=1,Ndets
    if (A(I,p,q) then
      find first J
      get the multiplicator
      add  $c_J * h_{pq} * m$  to  $c_I$ 
    if (S(I,p,q) then
      find possible second J
      get the multiplicator
      add  $c_J * h_{pq} * m$  to  $c_I$ 
  close the if's and the do's
```

# Sparse matrices

1%

2%

5%

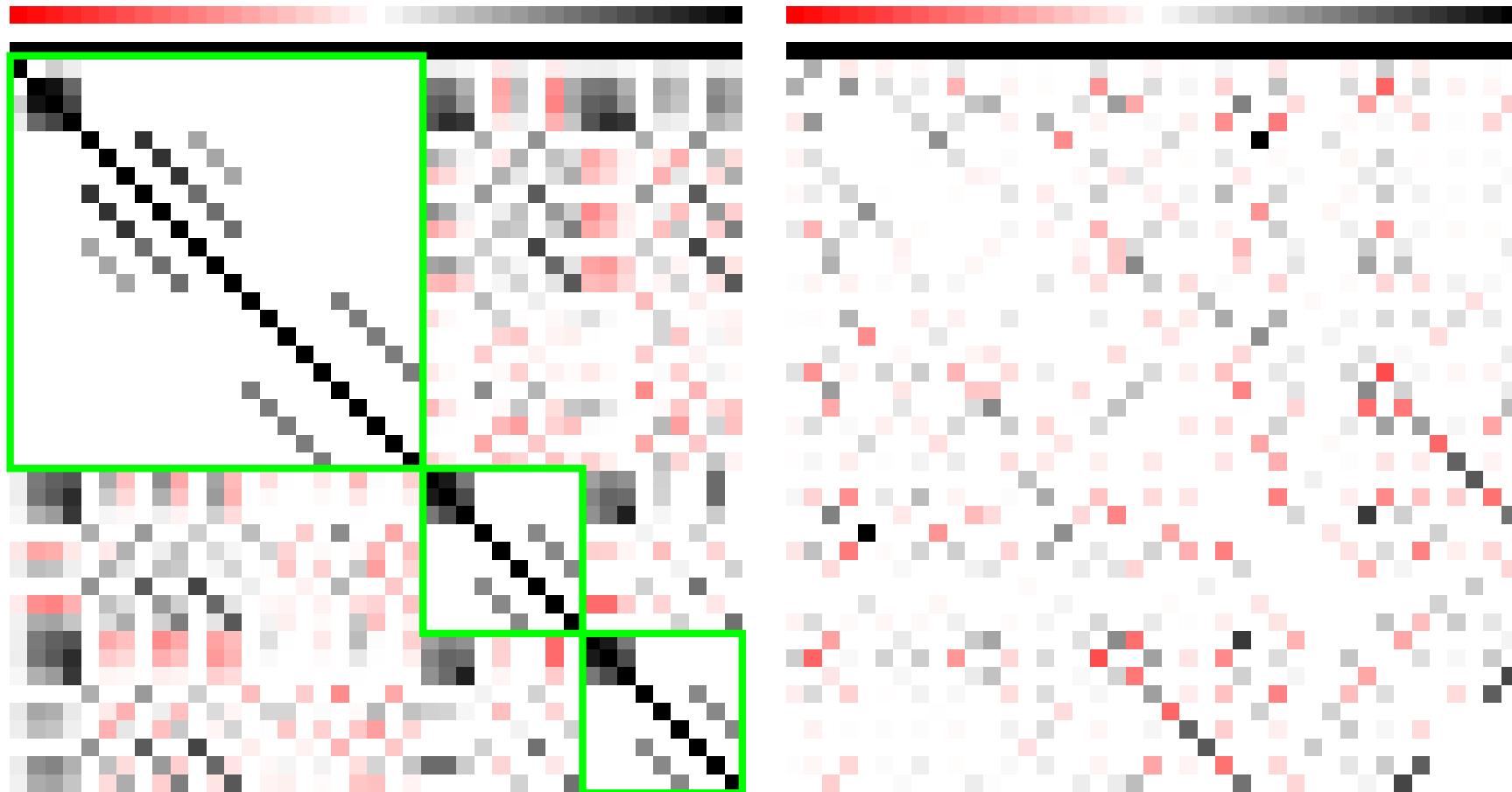
10%

25%

50%

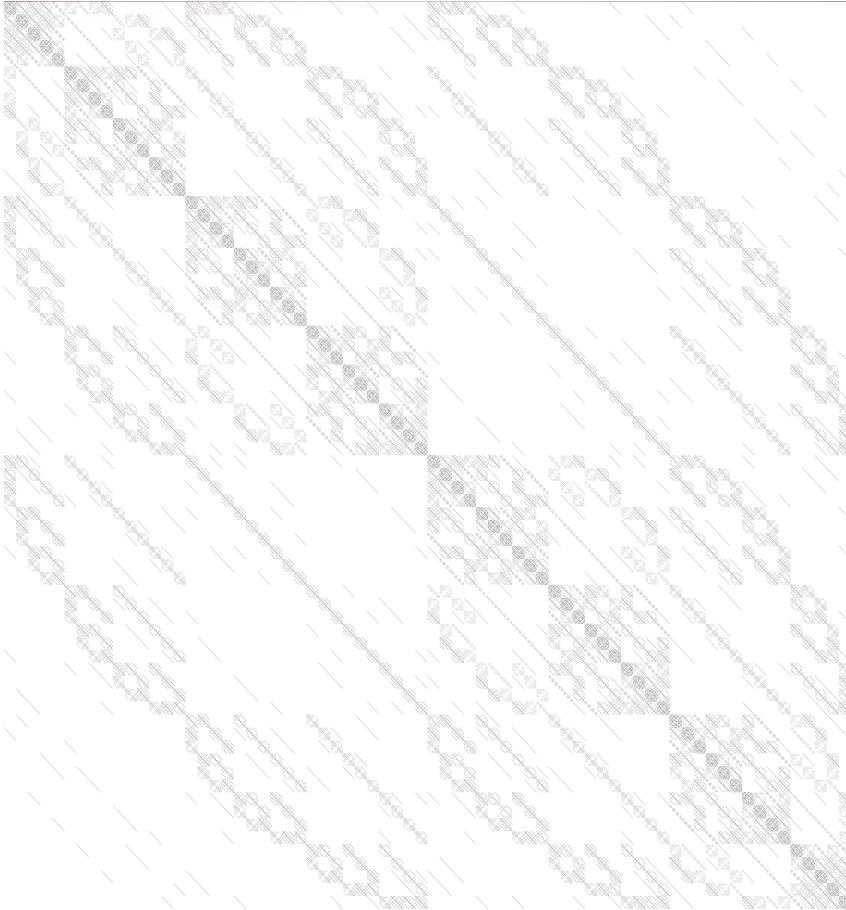
# Sparse matrices

Water molecule: overlap (AOs), Fock matrix (MOs), DZP



# Sparse matrices

Water molecule: Hamilton Matrix FCI (8 electrons / 8 orbitals,  $\approx 5000$  dets.)



# Canonical orbitals versus localized orbitals

- Diagonalization of  $F$  = delocalization of orbitals
- $N^2$  integrals ( $ii|jj$ ) scale as  $1/N \longrightarrow E \sim N$
- Localized orbitals:  $N$  integrals ( $ii|ii$ ) remain constant
- However: Fock matrix is not any more diagonal
- Need for orbital-invariant methods (CISD, CEPA–0, ACPF, Coupled-Cluster, Full CI)

Perturbation theory: define  $\hat{H}_0$  and  $\hat{V}$

$$\hat{H} = \sum_i F_{ii} \hat{a}_i^\dagger \hat{a}_i + \sum_i \sum_{j \neq i} F_{ij} \hat{a}_i^\dagger \hat{a}_j + \text{bielectronic operator}$$

- Sum additional series to infinity or approximate
- Include  $\sum_i \sum_{j \neq i} F_{ij} \hat{a}_i^\dagger \hat{a}_j$  in  $\hat{H}_0$ : system of linear equations to be solved

# Canonical orbitals versus localized orbitals

Orbital invariant MP2:

- Schrödinger equation  $(\hat{H}_0 + \lambda \hat{V}) \sum_i \lambda^i \Psi^{(i)} = \sum_j \lambda^j E^{(j)} \sum_k \lambda^k \Psi^{(k)}$
- First order wavefunction  $\Psi^{(1)} = \sum_I c_I^{(1)} \Phi_I$ , projection on  $\Phi_K$

$$\sum_I c_I^{(1)} \langle \Phi_K | \hat{H}_0 - E_0^{(0)} | \Phi_I \rangle + \langle \Phi_K | \hat{V} | \Phi_0 \rangle = 0$$

Observation of Hylleraas 1930:

- Define a functional

$$h = 2 \langle \Psi^{(1)} | \hat{V} | \Psi^{(0)} \rangle + \langle \Psi^{(1)} | \hat{H}_0 - E_0^{(0)} | \Psi^{(1)} \rangle$$

- Put derivatives to zero (minimize  $h$ )

$$\frac{\partial h}{\partial c_K} = 2 \langle \Phi_K | \hat{V} | \Phi_0 \rangle + \sum_I 2 c_I \langle \Phi_K | \hat{H}_0 - E_0^{(0)} | \Phi_I \rangle = 0$$

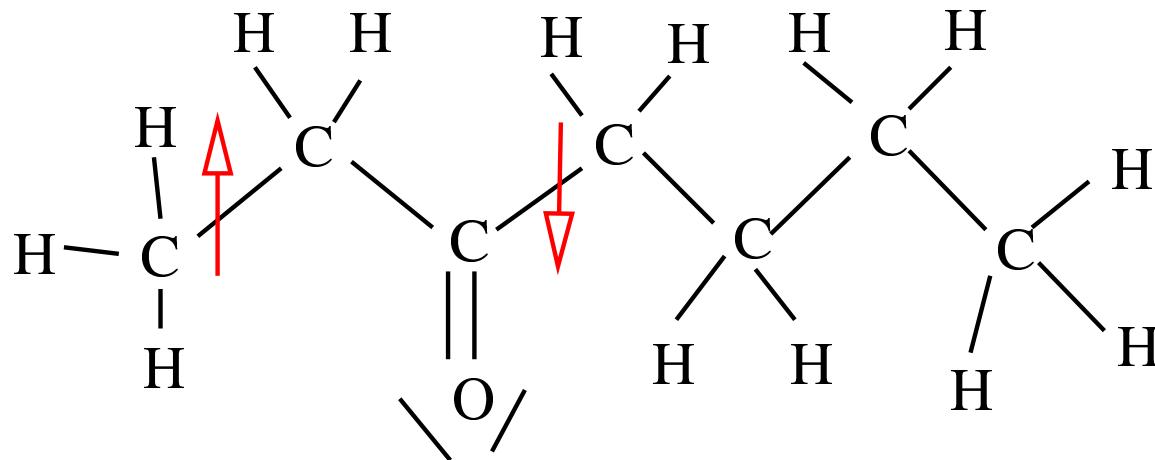
- Insert into functional:  $h = \langle \Psi^{(1)} | \hat{V} | \Psi^{(0)} \rangle = E_0^{(2)}$

# Canonical orbitals versus localized orbitals

- System of linear equations

$$\langle \Phi_K | \hat{H} | \Phi_0 \rangle + \sum_{c \in \text{virt}} (c_{ij}^{cb} F_{ac} + c_{ij}^{ac} F_{bc}) - \sum_{k \in \text{occ}} (c_{kj}^{ab} F_{ik} + c_{ik}^{ab} F_{jk}) = 0$$

- Evaluate integrals and Fock matrix in any set of Hartree-Fock orbitals
- Energy will be always the same as the coefficients minimize a functional.
- Formulas more complicated than standard MP2, but hope for linear scaling



# Matrix-dressing techniques

Start with CI of Singles and Doubles

$$(\mathbf{H} - \mathbf{E}_0) \cdot \Psi = E_{\text{corr.}} \cdot \Psi$$

- 1st row/column:  $\langle \Phi_0 | \hat{H} | \Phi_I \rangle$ ,  $\Phi_0 = \Phi_{\text{HF}}$ ,  $\Phi_I$ = di-excited determinant
- Add effects of higher excitations to the diagonal:  $H_{ii} \rightarrow H_{ii} + \Delta_i$
- $\Delta_i = E_{\text{corr}}$ : CEPA–0 (Coupled electron pair approximation)
- $\Delta_i = E_{\text{corr}} (1 - 2/n)$ : ACPF (Averaged Coupled Pair Functional,  $n$  = number of correlated electrons)
- $\Delta_i = E_{\text{corr}} - \sum_{J \in \text{EPV}} \langle \Phi_I | \hat{H} | \Phi_{I+J} \rangle$ : CEPA–2, CEPA–3, Full CEPA=(SC)<sup>2</sup>CI
- As  $\langle \Phi_I | \hat{H} | \Phi_{I+J} \rangle = \langle \Phi_0 | \hat{H} | \Phi_I \rangle$ , no additional cost
- Restores size-consistence of the CISD equations

# Matrix-dressing techniques

Coupled-Cluster equations

$$\langle \Phi_I | \hat{H} \left( 1 + \hat{T}_2 + \frac{1}{2} \hat{T}_2^2 \right) | \Phi_0 \rangle = (E_{\text{HF}} + E_{\text{corr}}) c_I$$

with  $\hat{T}_2 \hat{T}_2 |\Phi_0\rangle = 2 \sum_{klcd} (c_{ij}^{ab} * c_{kl}^{cd}) |\Phi_{ijkl}^{abcd}\rangle$  and

$$\begin{aligned} c_{ij}^{ab} * c_{kl}^{cd} &= c_{ij}^{ab} c_{kl}^{cd} - \langle c_{ij}^{ab} * c_{kl}^{cd} \rangle \\ &= c_{ij}^{ab} c_{kl}^{cd} - c_{ik}^{ab} c_{jl}^{cd} + c_{il}^{ab} c_{jk}^{cd} - c_{ij}^{ac} c_{kl}^{bd} + c_{ik}^{ac} c_{jl}^{bd} - c_{il}^{ac} c_{jk}^{bd} \\ &\quad + c_{ij}^{ad} c_{kl}^{bc} - c_{ik}^{ad} c_{jl}^{bc} + c_{il}^{ad} c_{jk}^{bc} + c_{ij}^{cd} c_{kl}^{ab} - c_{ik}^{cd} c_{jl}^{ab} + c_{il}^{cd} c_{jk}^{ab} \\ &\quad - c_{ij}^{bd} c_{kl}^{ac} + c_{ik}^{bd} c_{jl}^{ac} - c_{il}^{bd} c_{jk}^{ac} + c_{ij}^{bc} c_{kl}^{ad} - c_{ik}^{bc} c_{jl}^{ad} + c_{il}^{bc} c_{jk}^{ad} \end{aligned}$$

- Rearrange as

$$(H_{0I} - \sum_J H_{0J} \langle c_I * c_J \rangle) + \langle \Phi_I | \hat{H} - E_{\text{HF}} | \Phi_I \rangle c_I + \sum_{J \neq I} H_{IJ} c_J = 0$$

- Dressing of the first column, scales as  $N^8 (ijkl abcd)$

# Four-index transformation

$$(ij|kl) = \sum_{\alpha} \sum_{\beta} \sum_{\gamma} \sum_{\delta} c_{\alpha i} c_{\beta j} c_{\gamma k} c_{\delta l} (\alpha\beta|\gamma\delta)$$

- Scales in this form as  $N^8$
- Transform index after index, store intermediates, reduce scaling to  $N^5$
- Sort integrals, one index couple  $\alpha\beta$  on one file/processor
- Distribute — transform — assemble — distribute — transform — assemble

( $\alpha\beta$ )	1	2	3	4	5	
( $\gamma\delta$ )						

( $kl$ )	1	2	3	4	5	
( $\alpha\beta$ )						

- Only non-zero integrals are stored
- In localized orbitals at the limit linear scaling.

# Four-index transformation

$$(ij|kl) = \sum_{\alpha} \sum_{\beta} \sum_{\gamma} \sum_{\delta} c_{\alpha i} c_{\beta j} c_{\gamma k} c_{\delta l} (\alpha \beta | \gamma \delta)$$

Alternative: Laplace transformation for perturbation theory (*M. Häser*,  
Theor.Chim.Acta, **87** (1993) 1147)

$$E^{(2)} = -\frac{1}{4} \sum_{ijab} \frac{\langle ab || ij \rangle^2}{\epsilon_a + \epsilon_b - \epsilon_i - \epsilon_j} \quad (1a)$$

$$= -\frac{1}{4} \int_0^\infty dt \sum_{ijab} \langle ab || ij \rangle^2 e^{-(\epsilon_a + \epsilon_b - \epsilon_i - \epsilon_j)t} \quad (1b)$$

$$= \int_0^\infty e^{(2)}(t) dt, \quad (1c)$$

May be used with atomic orbitals!

# Exercise

Verify that the index function

$$\begin{aligned} p(i, j, k, l; n) &= (k-1)(n-1) - \frac{(k-1)(k-2)}{2} + l \\ &\quad + \frac{n(n+1)}{2}(j-1) - \frac{j(j-1)}{2} \\ &\quad + (i-1) \left( n \left( \frac{n(n-1)}{2} + 1 \right) - j(n-1) - 1 \right) \\ &\quad + \frac{(i-1)(i-2)}{2} \left( j - \frac{n(3n-7)}{2} - 4 \right) \\ &\quad + \frac{(i-1)(i-2)(i-3)}{2}(n-2) - \frac{(i-1)(i-2)(i-3)(i-4)}{8} \end{aligned}$$

produces indeed the position of a given integral  $(ij|kl)$  in a complete list of bielectronic integrals in canonical ordering, i.e.

- $0 < i, j, k, l \leq n$
- $i \leq j, k \leq l, i \leq k$
- if  $i = k$  then  $j \leq l$

# Exercise

Alternative formulation

$$\begin{aligned} p(i, j, k, l; n) = & \frac{1}{8} [-i^4 + i^3(4n+2) + i^2(4j - 2n(3n+5) - 3) + \\ & + 2i(2n+1)(-2j + n(n+3) + 1) \\ & - 4(j^2 - j(n(n+3) + 1) + n(n(n+3) - 2k) + \\ & (k-1)k - 2l + 4n)] \end{aligned}$$

You may consider the expression as a polynomial in  $i$ , to be evaluated as

$$P_4(i) = a_0 + i(a_1 + i(a_2 + i(a_3 + i a_4)))$$