Orbital optimisation

- To get a consistent answer, that does not depend on the start
- Not really very different from orthogonal case
- “Always” local - no guarantees to get absolute minimum
- Will discuss
    - Brillouin Theorem
    - Fock matrix
    - Newton-Raphson

Determinants

$$|A| = \begin{vmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{vmatrix} = A_{11} * A_{22} - A_{12} * A_{21}$$

$$|B| = \sum_{P}^{N!} (-1)^{p} P(abcd..) B_{1a} B_{2b} B_{3c} \ldots B_{Nz}$$

- A Determinant is not changed when
  - the matrix is transposed
  - a row/column multiplied by a constant is added to another
- A Determinant is linear in it's elements
- When row/columns are interchanged => sign := sign***-1**
- |ABC| = |A|.|B|.|C|
- The order of the determinant is the dimension of the matrix
- Nullity of matrix: dimension-rank (= # independent rows/columns

Orbital Division

$$\Psi = c_1 \left| \psi_c^2 \psi_d^2 \psi_{v1}^1 \psi_{v2}^2 \psi_{v3}^0 \right| +$$

$$+ c_2 \left| \psi_c^2 \psi_d^2 \psi_{v1}^1 \psi_{v2}^0 \psi_{v3}^2 \right| +$$

$$+ c_3 \left| \psi_c^2 \psi_d^2 \psi_{v1}^1 \psi_{v4}^1 \psi_{v5}^1 \right| +$$

- Core - doubly occupied in all determinants and frozen
- Doubly - doubly occupied in all determinants and optimised
- Variably - anything goes
- Empty - not occurring in any determinant

Frozen core

Always doubly occupied (space) orbitals may be projected out of other orbitals (Schmidt) without changing determinants. Then (e.g. Hosteny et al (1975))

$$E_{core} = \sum_{i \in \{\psi_i\}} \left\lfloor 2h_{ii} + \sum_{j \le i \in \{\psi_i\}} (2(ii \mid jj) - (ij \mid ji)) \right\rfloor$$

$$h_{mn} \Rightarrow h_{mn} + \sum_{i \in \{\psi_i\}} [2(mn \mid ii) - (mi \mid ni)]$$

Fock-matrix (n^4 ops)contains modified 1-electron integrals

The Generalised Brillouin Theorem

$$\Psi = |1\,\bar{1}\,2\,\bar{2}\,3|$$

infinitesimal

$$\psi_1 \rightarrow \psi_1 + \delta_{14}\psi_4$$

$$|1\,\bar{1}\,2\,\bar{2}\,3| \rightarrow |(1+\delta_{14}4)\overline{(1+\delta_{14}4)}2\,\bar{2}\,3| =$$

$$= |1\,\bar{1}\,2\,\bar{2}\,3| + \delta_{14}|4\,\bar{1}\,2\,\bar{2}\,3| + \delta_{14}|1\,\bar{4}\,2\,\bar{2}\,3| + \delta_{14}^2|4\,\bar{4}\,2\,\bar{2}\,3|$$

$$= |1\,\bar{1}\,2\,\bar{2}\,3| + \delta_{14}\left(|4\,\bar{1}\,2\,\bar{2}\,3| + |1\,\bar{4}\,2\,\bar{2}\,3|\right)$$

$$= |1\,\bar{1}\,2\,\bar{2}\,3| + \delta_{14}.C_{1\rightarrow 4}|1\,\bar{1}\,2\,\bar{2}\,3|$$

$C_{i\rightarrow j}$

Replace orbital i by orbital j, once for α spin and once for β spin in the complete function

- Unitary group generator
- retains spin
- is unnormalised
- applicable to multi-reference wave functions

Singly excited (Brillouin) state

$$\Psi_{ij} = C_{i\to j}\Psi_0 \quad \text{(VB)}$$

$$\Psi_{ij} = \left(C_{i\to j} - C_{j\to i}\right)\Psi_0$$

Unitary

$$\Psi_0(\psi_i \to \psi_i + \delta_{ij}\psi_i) = \Psi_0 + \delta_{ij}\Psi_{ij}$$

So the change in Ψ_0 is $d\Psi_0 = \delta_{ij}\Psi_{ij}$

Differentiate the energy with respect to orbital mixing $\dfrac{\langle\Psi|H|\Psi\rangle}{\langle\Psi|\Psi\rangle}$

$$\left(\text{remember}: E' = \left(\frac{t}{n}\right)' = \frac{t'n - n't}{n^2} = \frac{t' - n'\dfrac{t}{n}}{n} = t' - En'\right)$$

$$\frac{dE}{\delta_{ij}} = \frac{2}{\delta_{ij}}\langle d\Psi_0 | H - E_0 | \Psi_0\rangle = \frac{2}{\delta_{ij}}\langle \delta_{ij}\Psi_{ij} | H - E_0 | \Psi_0\rangle = 2\langle \Psi_{ij} | H - E_0 | \Psi_0\rangle$$

The Generalised Brillouin Theorem(3)

$$\left\langle \Psi_{ij} \middle| H - E_0 \middle| \Psi_0 \right\rangle = 0$$

Levy, Berthier(1968)
(without E_0)

Stationary condition for arbitrary wave function $\Psi_0 = \sum_k a_k \Phi_k$

$$\Psi_0(\psi_i \rightarrow \psi_i + \delta_{ij}\psi_i) = \Psi_0 + \delta_{ij}\Psi_{ij}$$

Make Super CI (Brillouin-state Interaction (BI))

$$\Psi_{BI} = b_0\Psi_0 + \sum b_{ij}\Psi_{ij} \quad \{solve(\mathbf{H} - E\mathbf{S})\mathbf{b} = 0\}$$

Use b_{ij} to obtain orbitals that make Ψ_0 approach Ψ_{ij} to first order

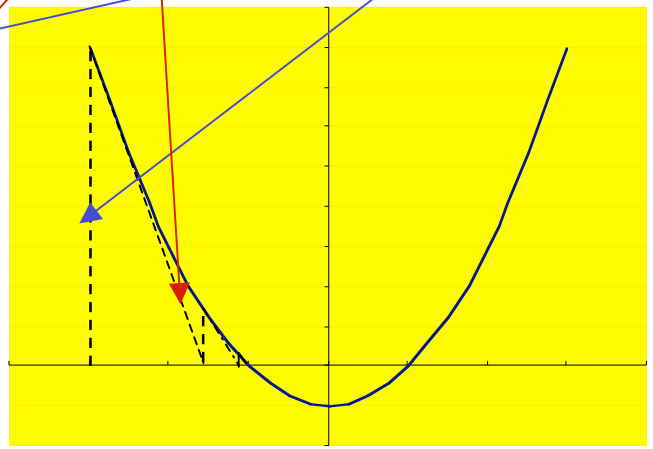
$$\psi_i \rightarrow b_0\psi_i + \sum b_{ij}\psi_j$$

← Adapt Ψ_0

Newton Raphson : $c = -\left[\nabla^2 E\right]^{-1}.\nabla E = -\mathbf{H}^{-1}\mathbf{g}$

Hessian

gradient



$$\frac{dE}{\delta_{ij}} = 2\left\langle \Psi_{ij} \middle| H - E_0 \middle| \Psi_0 \right\rangle$$

$$\frac{d^2 E}{\delta_{ij}\delta_{kl}} = 2\left[\left\langle \Psi_{ij} \middle| H - E_0 \middle| \Psi_{kl} \right\rangle + \left\langle \Psi_0 \middle| H - E_0 \middle| \Psi_{ij,kl} \right\rangle\right] -$$

$$4\left[\left\langle \Psi_0 \middle| H - E_0 \middle| \Psi_{ij} \right\rangle\left\langle \Psi_0 \middle| \Psi_{kl} \right\rangle + \left\langle \Psi_0 \middle| H - E_0 \middle| \Psi_{kl} \right\rangle\left\langle \Psi_0 \middle| \Psi_{ij} \right\rangle\right]$$

$(\mathbf{H} - E\mathbf{S})\mathbf{b} = 0$ Set b_0 (first element) to 1

$$\begin{pmatrix} E_0 - E & (\mathbf{H} - E\mathbf{S})^{\dagger}_{(1..n)0} \\ (\mathbf{H} - E\mathbf{S})_{(1..n)0} & (\mathbf{H} - E\mathbf{S})_{(1..n)(1..n)} \end{pmatrix} \begin{pmatrix} 1 \\ \mathbf{b} \end{pmatrix} = \mathbf{0}$$

$=\mathbf{g}'$

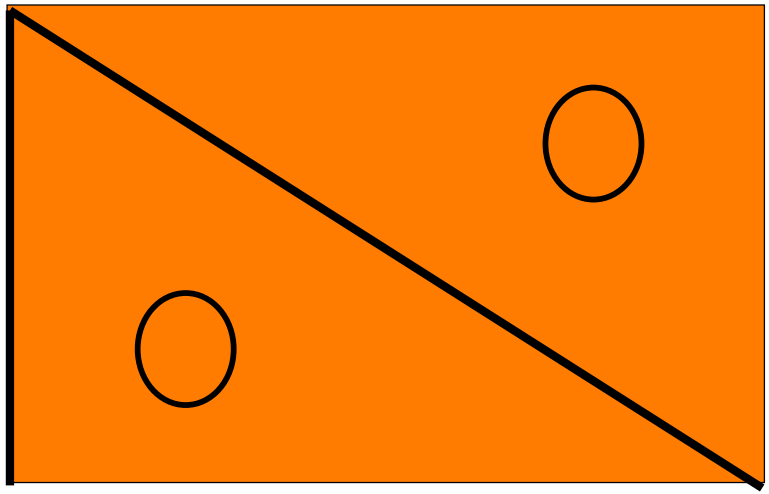
$=\mathbf{H}'+(E_0-E)\mathbf{S}$

$$\mathbf{b} = -(\mathbf{H}' + (E_0 - E)\mathbf{S})^{-1}.\mathbf{g}'$$

Approximate Hessian
no quadratic convergence

Automatic Level shift stabilises NR
(makes Hessian more positive definite)

When we ignore most of the matrix elements:



$$b_{ij} = -\frac{\langle \Psi_0 | H - E_0 | \Psi_{ij} \rangle}{\langle \Psi_{ij} | H - E_0 | \Psi_{ij} \rangle}$$

Hartree-Fock-like

Convergence stabilisation: Level Shifting → Substract from H_{11}

DIIS (GMRES) ↓

One has n pairs of iterands (orbitals ψ)
and error-vectors (gradients g)

$$g_{opt} = \sum_i^n c_i g_i \quad with \langle g | g \rangle \text{ minimal}$$

$$then \ \psi_{opt} = \sum_i^n c_i \psi_i$$

Orbital Optimisation - Fock Operator

(Closed shell Hartree Fock - type)

$$f_{ij} = \frac{1}{2}\left\langle \Psi_0 \middle| H \middle| \Psi_{ij} \right\rangle$$

Generating all Fock matrix elements

- requires 1-electron density matrix of Ψ_0
- requires n^4 operations
- only valid for orthogonal orbital sets

Then

- $\left\langle \Psi_0 \middle| H \middle| \Psi_{ij} \right\rangle = F_{ij}$
- $\left\langle \Psi_{ij} \middle| H \middle| \Psi_{ij} \right\rangle \approx \left(F_{jj} - F_{ii} \right)$

Problems

- watch # degrees of freedom
- convergence is troublesome if orbitals are very alike

Suppose we have the set $\{\psi_d, \psi_v, \phi\}$

doubly MO — variably MO — AO

Allowed mixings :

- $\psi_d \rightarrow \{\psi_v, \phi \not\subset \psi_d \cup \psi_v\}$
- $\psi_v \rightarrow \{\psi'_v, \phi \not\subset \psi_d \cup \psi_v\}$

$$P_{virtual} = \left(1 - \sum \psi_v \psi_v^\dagger - \sum \psi_d \psi_d^\dagger\right)$$

Or (# ψ_v may be > # ϕ)

diagonalise (+canonicalise) every iteration => ϕ

- $\psi_d \rightarrow \{\phi \not\subset \psi_d\}$
- $\psi_v \rightarrow \{\phi \not\subset \psi_d \cap \phi \, with \langle \phi | \psi_v \rangle \max .\}$

$$P_{virtual} = \left(1 - \sum \psi_d \psi_d^\dagger\right)$$

Example Orbital Optimisation H_2O - DZP

Fock-operator

1	-75.9934512	.09448891
2	-76.0192549	.03127837
3	-76.0296263	.01365725
4	-76.0300204	.00610260
5	-76.0301179	.00487776
6	-76.0301522	.00235531
7	-76.0301657	.00189198
8	-76.0301714	.00105559
9	-76.0301738	.00077691
10	-76.0301748	.00047629
11	-76.0301753	.00032852
12	-76.0301755	.00020987
13	-76.0301755	.00014086
14	-76.0301756	.00009158

final after 20 cycles at .04 seconds

Brillouin Interactie

it. 1 at .13 evb -76.0192549 brm 1.3E-01
it. 2 at .17 evb -76.0282700 brm 5.0E-02
it. 3 at .21 evb -76.0297771 brm 2.2E-02
it. 4 at .26 evb -76.0300936 brm 9.9E-03
it. 5 at .30 evb -76.0301585 brm 4.5E-03
it. 6 at .34 evb -76.0301720 brm 2.1E-03
it. 7 at .38 evb -76.0301749 brm 9.4E-04
it. 8 at .42 evb -76.0301755 brm 4.3E-04
it. 9 at .46 evb -76.0301756 brm 2.0E-04

final vbscf results after cycle 13 at .560 seconds

Fock-operator

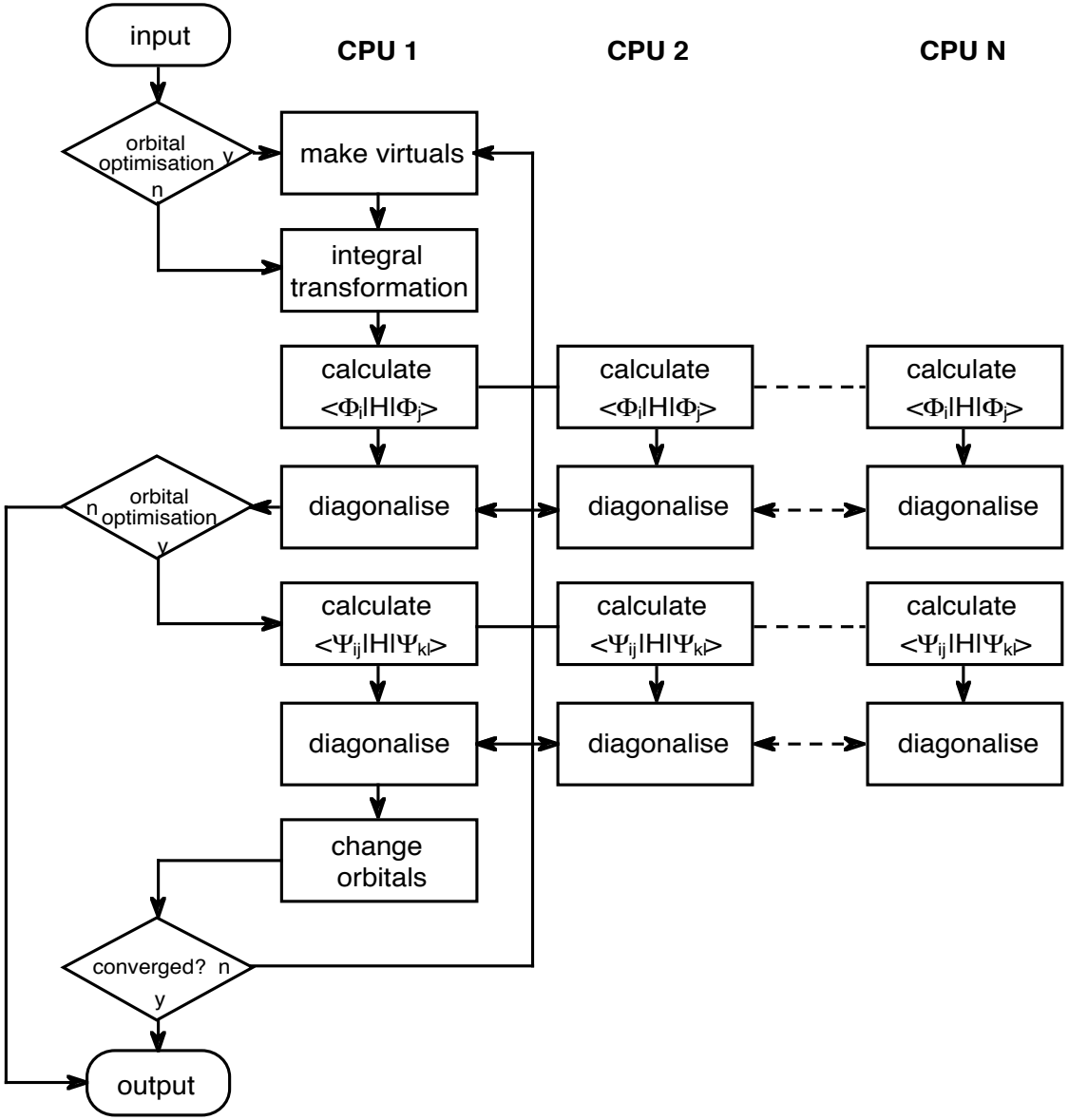
+ level shifting

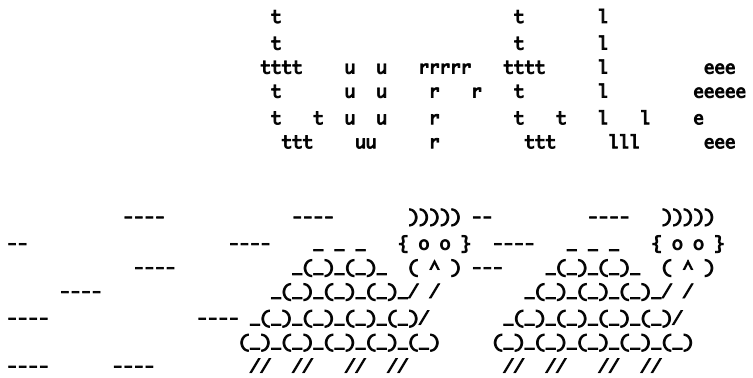
final after 16 cycles at .04 seconds

+ level shifting + diis

final after 9 cycles at .04 seconds

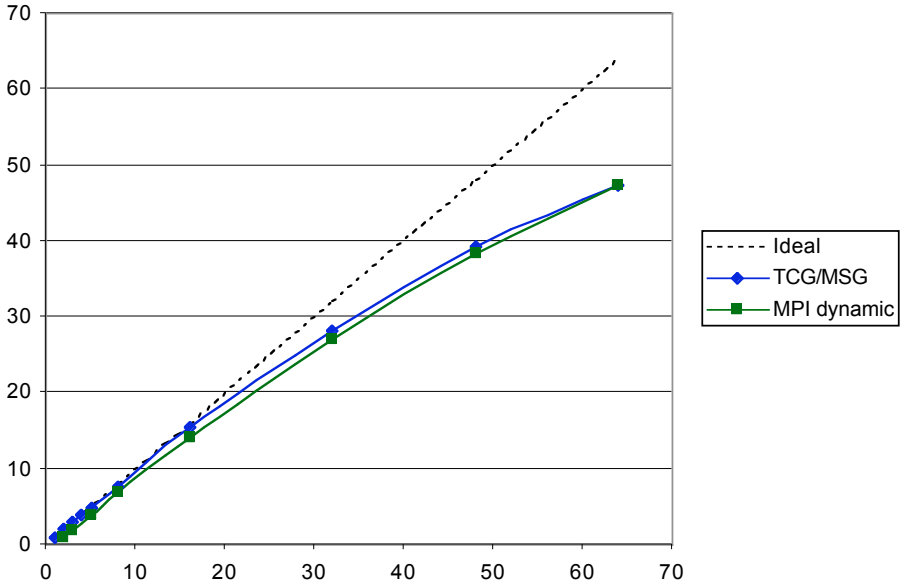
Structure of (parallel) TURTLE





TERAS

T3E



#PE	Speedup
64	54
128	96 (est)
1024	281 (est)
Parallel 99.7%	

- T3E : good communications / *slow* processor
- TURTLE: slow code (~ 100% in matrix elements)