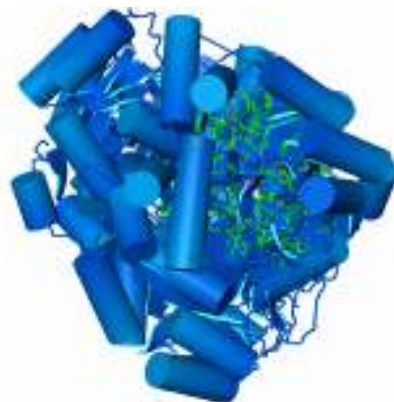
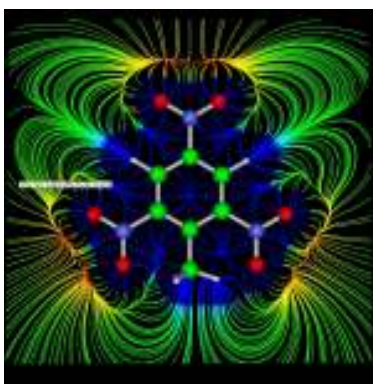


# Quantum Chemistry with GAMESS-UK

## Session 2

### More Advanced Options



<http://www.cfs.dl.ac.uk/tutorials/gamess-uk.2007/L7.pdf>

Martyn F. Guest

CCLRC Daresbury Laboratory & Cardiff University

[m.f.guest@dl.ac.uk](mailto:m.f.guest@dl.ac.uk)

# Contents

## 1. Analysing the Wavefunction

- ✘ Properties
- ✘ Graphics
- ✘ Potential Derived Charges

## 2. Density Functional Theory (DFT)

- ✘ Coulomb fitting basis sets
- ✘ Gradient corrections
- ✘ Quadrature selection

## 3. Transition States

## 4. ECPs and Valence Only Calculations

## 5. Relativistic Calculations (ZORA)

## 6. Post Hartree Fock Calculations

- ✘ MP2, MP3
- ✘ MCSCF / CASSCF
- ✘ Direct-CI
- ✘ Conventional Selection-CI
- ✘ CCSD and CCSD(T)

## 7. Solvation (DRF)

## 8. Other RUNTYPEs

## 9. Multiple RUNTYPE specification

## 10. Other Analysis Codes (NBO and AIMPAC)

## 11. Functionality and Performance of the Parallel Code

## 12. Trouble Shooting

- ✘ Linear Dependence
- ✘ SCF Convergence
- ✘ Geometry optimisation and Transition States

# Analysing the Wavefunction

GAMESS-UK includes a variety of tools for analysing wavefunctions, driven by the RUNTYPE ANALYSE directive. It is now possible to:

1. Calculate a variety of **1-electron properties**.
2. Generate a **localised orbital** representation of an SCF wavefunction using either the dipole-centroid technique (Foster and Boys), or an overlap-based criterion (Pipek and Mezey).
3. Provide graphical analysis of wavefunctions, through the generation of contour and perspective plots which depict: the **electron density** associated with one or more MOs; the **amplitude** of a MO; a **comparison of the density distribution** in two or more molecular systems; the interaction energy between a molecular distribution and a hypothetical point charge, generating the so-called **electrostatic potential plot**. Both contour and perspective plots may be generated.
4. Perform a distributed multipole analysis (DMA) of an SCF wavefunction.
5. Perform a more detailed Mulliken analysis, including both bond and orbital properties.
6. Morokuma Energy Decomposition analysis.
7. ***Graphical Analysis using the CCP1 GUI***.
8. Interfaces to other popular analysis packages, including
  - ❑ NBO
  - ❑ AIMPAC

# Analysis Options – 1. Properties

## One Electron Properties and Operator Numbers

- RUNTYPE ANALYSE, with explicit specification of the required one-electron properties.
- A simplified mechanism for property evaluation can be requested through presenting the data line

### PROPERTY ATOMS

after RUNTYPE and SCFTYPE specification. This will result in the default wavefunction analysis conducted after RUNTYPE processing being augmented with the computation of certain **one-electron properties**.

1	Potential
2	Diamagnetic Shielding
3	Electric Field
4	Electric Field Gradient
5	Dipole Moment
6	Quadrupole Moment
7	Diamagnetic Susceptibility
8	Second Moment
9	Octupole Moment
10	Third Moment
11	Third Moment (combined)
12	Hexadecapole Moment
13,14	Fourth Moment (even,odd)
15	Overlap
16,17	Planar, Line Charge Density
18	Charge Density
19	Isotropic ESR Coupling Constants
20	Anisotropic ESR Coupling Constants

## Analysis Options – 2. Graphics

- GAMESS-UK supports the graphical analysis of wavefunctions by the calculation of charge densities, molecular orbitals, atom difference densities and electrostatic potentials on a grid of points.
- The module is invoked under control of RUNTYPE ANALYSE by presence of the GRAPHICS directive.
- The following classes of directives exist
  - ⌘ Grid definition - **GDEF**
  - ⌘ Data calculation - **CALC**
  - ⌘ Surface generation - **SURF**
  - ⌘ Restoring data from the dumpfile - **RESTORE**
- The data is stored on the dumpfile using the SECTION directive.
- A PUNCH directive is required to write the data to the punchfile.

## Graphics - GDEF

- The grid definition is initiated by the GDEF directive
- The grid TYPE can be one of
  - α 2D - Rectangular 2D grid
  - α 3D - Orthogonal 3D grid
  - α SPHERE - Spherical grid
  - α CARDS - User specified points
  - α CONTOUR - Generate points on an isovalue surface
  - α WRAP - Generate points on an iso-electron-density surface
  - α ATOM - Place grid points at the nuclear positions
- SIZE specifies the edge length of the grid
- POINTS specifies the mesh density
- SECTION specifies the dumpfile section number where the grid will be stored; use PUNCH to output in formatted form.

```
...  
RUNTYPE ANALYSE  
PUNCH GRID 100  
GRAPHICS  
GDEF  
TITLE  
2D GRID 80 x 100 POINTS  
TYPE 2D  
SIZE 10 12  
POINTS 80 100  
SECTION 100  
...
```

## Graphics - CALC

- The computation of data on a predefined grid is requested by the CALC directive.
- The TYPE directive selects the property to be calculated
  - ⊠ **MO - Molecular Orbital**
  - ⊠ **DENS - electron density**
  - ⊠ **ATOM - atom difference electron density**
  - ⊠ **POTE - electrostatic potential**
  - ⊠ **GRAD DENS - gradient of the charge density**
  - ⊠ **GRAD MO - gradient of MO**
  - ⊠ **COMB - linear combination with the previous calculation**
  - ⊠ **VDW - Van der Waals function**
  - ⊠ **LVDW - log Van der Waals function**
  - ⊠ **GRAD VDW - gradient of Van der Waals function**
  - ⊠ **GRAD LVDW - gradient of Van der Waals function (sic)**
- The OCCDEF directive allows modifications to the occupation numbers

# Graphics - SURFACE

- The SURFACE directive is used to generate one or more isovalue surfaces from a 3D dataset, and calculate data at points on that surface. It is presented after directives requesting the calculation of the 3D dataset.
- Example calculates the potential at sets of grid points on the density iso-surfaces with values 0.02 and 0.04. The iso-density points are stored on sections 160 and 163, the density gradients on 161 and 164, and the potential on 162 and 165.

```
RESTART NEW
PUNCH GRID 162 165
ZMAT ANGS
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
BASIS DZ
RUNTYPE ANALYSE
GRAPHICS
GDEF
TYPE 3D
POINTS 60
SIZE 8.0
CALC
TYPE DENS
SECTION 150
SURFACE POTE 160 0.02 0.04
VECTORS 1
ENTER
```

graphics3a.in, graphics3b.in

## Analysis options – 3. Potential Derived Charges I.

- The potential derived charges module uses *electrostatic potential data calculated using the graphics module to generate least-squares fitted point charges at the nuclei.*
- The potential values can most easily be calculated using SURFACE
- The fitting procedure runs under the control of the POTFIT directive
- During the fitting constraints may be applied
  - ⌘ **CHARGE charge** - constrain charge
  - ⌘ **SYMMETRY** - constrain symmetry equivalent atoms to have the same charge
  - ⌘ **DIPOLE dx dy dz** - constrain dipole (atomic units)
  - ⌘ **CUTOFF scale** - exclude points closer than **scale** \* the covalent radius from a nucleus.

## Analysis options – 3. Potential Derived Charges II.

Generate the electron density on a 3D grid

1. Calculate the electrostatic potential at the 0.02 and 0.04 isodensity surfaces. This data is written to the dumpfile on section 172 and 175
2. Generate potential derived charges by fitting to the potential points from 1. The total charge is constrained to 0.0

```
ZMAT ANGS  
O  
H 1 1.0  
H 1 1.1 2 109.0  
END  
RUNTYPE SCF  
ENTER  
RUNTYPE ANALY  
GRAPHICS  
GDEF  
TYPE 3D  
POINTS 50  
SIZE 6  
SECTION 150
```

```
CALC  
TYPE DENS  
SECTION 151  
SURFACE POTE 170 0.02 0.04  
VECTORS 1  
ENTER  
RUNTYPE ANALY  
POTFIT 172 175 CHAR 0.0  
VECTORS 1  
ENTER
```

graphics2.in

# DFT Calculations: Quadrature Selection

Adequate *quadrature* is one that addresses 3 particular features of molecular electronic densities: (1) a cusp at the position of every atom, (2) the density decays exponentially with increasing distance to the atoms, (3) in the inter atomic region the density is smooth but can have a number of extrema due to contributions from various atoms.

## Quadrature Settings

- ✘ Quadrature partitions integrand into atom like terms (weighting scheme)
- ✘ Each term integrated in polar coordinates
- ✘ Defaults
  - MHL8SSFSCR weights
  - 15 + 10\*row# Mura-Knowles
  - 302 Lebedev

```

TITLE
H2CO DFT B3LYP REDEFINE QUADRATURE
ZMATRIX ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
BASIS 6-31G
DFT B3LYP
DFT WEIGHT BECKESCR
DFT LOG ELEMENT H 60 3.0
DFT LOG LABEL C 70 3.0
DFT LEBEDEV 86 0.5 302 1.5 194
DFT SCREEN P 1D-10 RHO 1D-10
DFT SCREEN PSI 1D-7
ENTER
  
```

dft4.in

# DFT Calculations:

## Use of the Auxiliary basis Coulomb Fit

- “Pure” density functional required
- Replace  $\rho(\vec{r}) = \sum_{j\mu\nu} C_{\mu j} C_{\nu j} \chi_{\mu}^*(\vec{r}) \chi_{\nu}(\vec{r})$
- With charge density fitted to an auxiliary basis:

$$\tilde{\rho}(\vec{r}) = \sum_{\mu} D_{\mu} \varphi_{\mu}(\vec{r})$$

- Advantage: at most 3-centre integrals needed
- Disadvantage
  - ⌘ Coulomb matrix in fitting basis inversion needed
  - ⌘ Minimal loss of accuracy
- On balance: considerable savings for medium sized molecules

*Dunlap, B. I., Connolly, J. W. D. and Sabin, J. R., 1979, J. Chem. Phys., 71, 4993-4000.*  
*Dunlap, B. I., Connolly, J. W. D. and Sabin, J. R., 1979, J. Chem. Phys., 71, 3396-3402.*

# DFT Calculations:

## Use of the Auxiliary basis Coulomb Fit II.

- Use Coulomb fit: JFIT, for energy and JFITG for gradient evaluations
- Use JFIT MEMORY to keep as many integrals as possible in memory
- Use Schwarz inequality to avoid calculating integrals with small values. SCHWARZ 6 discards all integrals less than  $10^{-6}$ .
- Use JBAS to specify fitting basis. A1-DGAUSS, A2-DGAUSS, DEMON or AHLRICHS basis sets available

```
TITLE
H2CO DFT BLYP/6-31G with J-FIT
ZMATRIX ANGSTROM
C
O 1 CO
H 1 CH 2 121.8
H 1 CH 2 121.8 3 180.0
VARIABLES
CO 1.203
CH 1.099
END
BASIS 6-31G
RUNTYPE OPTIMISE
SCFTYPE DIRECT
DFT JFIT MEMORY JFITG
DFT SCHWARZ 6
DFT JBAS A1-DGAUSS
ENTER
```

dft5.in

# Transition State Location

Three methods are available to search for a transition state on a potential Surface, each driven through SADDLE specification on the RUNTYPE directive and each relying on internal co-ordinate specification through the ZMATRIX directive.

1. the recommended method, a modification to the Cerjan and Miller 'trust-region' algorithm, is driven through the specification

**RUNTYPE SADDLE**

Performs search in internal co-ordinates, and thus requires initial ZMATRIX and VARIABLES specification of the geometry, or ZMATRIX construction from a set of cartesian co-ordinates supplied under control of GEOMETRY. The success of the method is dependent on the *quality of initial Hessian*, addressed through e.g. TYPE specifications on the VARIABLE definition lines of the ZMATRIX.

2. the second internal coordinate-driven method is that based on the hill-walking algorithm due to Simons and Jorgensen. The procedure is again reliant on a “quality” initial hessian:

**RUNTYPE SADDLE JORGENSEN**

3. the third method, less robust and requiring additional input data, is the synchronous-transit internal co-ordinate based method due to Bell and Crighton.

**RUNTYPE SADDLE**

**LSEARCH 0 4**

# Generating the Initial Hessian

A number of mechanisms are available to define or calculate the initial hessian:

1. On the VARIABLE definition lines of the ZMATRIX. Assigning a value to the diagonal element(s) of the Hessian corresponding to the variable in question is achieved by presenting the character string **HESSIAN** followed by the value to be assigned. →
2. Explicit computation of the Hessian may be requested through specification of **TYPE** on the VARIABLE definition lines. In such cases the corresponding part of the Hessian will be evaluated numerically, prior to commencing optimisation. Two settings:  
**TYPE 2** ; requests calculation of the **diagonal force constant** (an additional energy calculation)  
**TYPE 3** ; requests calculation of the **diagonal force constant and all off-diagonal elements** involving the variable (an additional energy-plus-gradient calculation for each variable).
3. Explicit computation of the entire initial Hessian, under control of **RUNTYPE HESSIAN**, and subsequent retrieval of this Hessian at the outset of the transition state search. Major benefits in practice.
4. Restoring the hessian from the dumpfile of a smaller calculation

**RUNTYPE SADDLE ED3 1**

```
OH 0.951 HESSIAN 1.1
```

```
ZMAT ANGS
```

```
C
```

```
C 1 L1
```

```
H 2 L2 1 A1
```

```
X 2 1.0 1 90.0 3 180.0
```

```
H 2 L3 4 A2 1 180.0
```

```
VARIABLES
```

```
L1 1.24054 TYPE 3
```

```
L2 1.65694 TYPE 3
```

```
L3 1.06318 TYPE 3
```

```
A1 60.3568 TYPE 3
```

```
A2 60.3568 TYPE 3
```

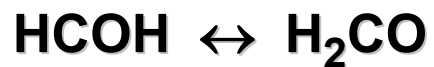
```
END
```

# The Initial Hessian: RUNTYPE HESSIAN

## Run I: Generating the Trial Hessian

```
TITLE
HCOH <-> H2CO 1A' TS
ZMAT ANGS
C
O 1 CO
H 1 CH1 2 OCH1
H 1 CH5 2 H5CO 3 180.0
VARIABLES
OCH1 56.3
CO 1.27
CH1 1.22
CH5 1.10
H5CO 115.8
END
BASIS 6-31G
RUNTYPE HESSIAN
ENTER
```

sad1a.in



```
RESTART NEW
TITLE
HCOH <-> H2CO 1A' TS
ZMAT ANGS
C
O 1 CO
H 1 CH1 2 OCH1
H 1 CH5 2 H5CO 3 180.0
VARIABLES
OCH1 56.3
CO 1.27
CH1 1.22
CH5 1.10
H5CO 115.8
END
BASIS 6-31G
RUNTYPE SADDLE FCM
ENTER
```

sad1b.in

## Run II: Restoring the Trial Hessian

## Simple Transition State Location Examples



```
TITLE
H2 + CO <-> H2CO 1A' TS
ZMAT ANGS
O
C 1 CO
X 2 1.0 1 90.0
X 2 CHH 3 ANG1 1 180.0
X 4 1.0 2 90.0 3 0.0
X 4 1.0 5 ANG2 3 0.0
H 4 XH 6 90.0 2 180.0
H 4 XH 6 90.0 2 0.0
VARIABLES
CO 1.134 TYPE 3
ANG1 43.7 TYPE 3
ANG2 57.8 TYPE 3
CHH 1.292 TYPE 3
XH 0.664 TYPE 3
END
RUNTYPE SADDLE
ENTER
```

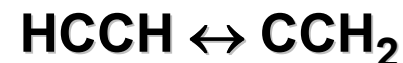
sad3.in



```
TITLE
H2CO <-> HCOH 1A' TS
ZMAT ANGS
C
O 1 CO
H 1 CH1 2 OCH1
H 1 CH5 2 H5CO 3 180.0
VARIABLES
OCH1 57.236 TYPE 3
CO 1.299456 TYPE 3
CH1 1.201293 TYPE 3
CH5 1.115436 TYPE 3
H5CO 116.882 TYPE 3
END
RUNTYPE SADDLE ED4 1
VECTORS GETQ ED4 1 1
ENTER
```

sad4a.in, sad4b.in

Other Directives:  
STEPMAX, XTOL, VALUE



```
TITLE
HCCH <-> CCH2 . RHF
ZMAT ANGS
C
C 1 L1
H 2 L2 1 A1
X 2 1.0 1 90.0 3 180.0
H 2 L3 4 A2 1 180.0
VARIABLES
L1 1.24054 TYPE 3
L2 1.65694 TYPE 3
L3 1.06318 TYPE 3
A1 60.3568 TYPE 3
A2 60.3568 TYPE 3
END
RUNTYPE SADDLE
XTOL 0.002
ENTER
```

sad2.in

# Effective Core Potentials I.

- Both non-local and local ECP calculations are available. Non-local ECPs are of limited applicability, and are of historical interest only (see manual).
- The PSEUDO directive is used to request a valence only (ECP) rather than all electron calculation.
- A local ECP calculation requires the definition of a potential that accounts for core electrons plus a basis set for the valence electrons. A number of ECP Potentials - Valence Basis pairs are provided from the literature.
- Seven build in ECPs are available: LANL, LANL2, CEP / SBKJC, CRENBL, CRENBS, STRLC, and STRSC.

Element name in the  
ECP library

```
PSEUDO ECP
S CRENBL S
C CRENBL C12
```

Atom tag as specified in  
the geometry / z-matrix

```
TITLE
H2CS - SBKJC ECP
ZMATRIX ANGSTROM
C
S 1 CS
H 1 CH 2 HCS
H 1 CH 2 HCS 3 180.0
VARIABLES
CS 1.5990897
CH 1.0775219
HCS 122.0282708
END
BASIS ECP SBKJC
PSEUDO ECP
S SBKJC S
C SBKJC C
RUNTYPE OPTIMIZE
ENTER
```

ecp0.in

Syntax: “Take the Carbon ECP from the CRENBL library and assign it to atom C12”

## Effective Core Potentials II.

Alternatively ECPs can be specified explicitly:

Maximum angular momentum and the number of electrons accounted for

The maximum angular momentum projector come first

The number of Gaussians in the projector

The power of  $r$  to multiply with the Gaussian

ecp2.in

PSEUDO ECP			
CARD NA			
2	10		
1			
1	-2.384460	0.90009	
5			
2	-10.36981	2.80357	
2	10.85699	2.61310	
2	-0.62339	0.70005	
0	6.23415	5.37232	
2	9.08374	1.11959	
5			
2	4.43943	0.78188	
2	-6.14647	0.66772	
2	1.72860	0.58063	
0	3.23971	1.29158	
2	2.53514	0.65791	

Coefficients      Exponents

## ECP Basis Sets

Seven of the standard literature basis sets may be used in performing valence-only calculations, namely;

- **LANL**, The double-zeta Hay and Wadt ECP basis sets (Na-Bi), with the outer-valence forms used for transition metals etc.
- **CEP / SBKJC** the Compact Effective Potentials due to (i) Stevens et al (Li-Ar, K-Rn, and Cundari et al (Lanthanides).
- **LANL2** - The double-zeta Hay and Wadt ECP basis sets, with the inner-valence forms used for transition metals etc. (as in Gaussian and NWChem).
- **CRENBL** - The large ECP orbital basis for use with the small core potentials due to Christiansen et al .
- **CRENBS** - The small ECP orbital basis for use with the averaged relativistic, large core ECPs due to Ermler and co-workers.
- **STRLC** - The Stuttgart relativistic, large core ECP basis sets due to Preuss et al.
- **STRSC** - The Stuttgart relativistic, small core ECP basis sets due to Preuss et al.

- See Table 5 of Part 4 of the manual for a list of the elements available for each library

- Note that six of the basis sets above are assumed to be **spherical harmonic**, and not cartesian (CEP/SBKJC basis). Their usage should thus be accompanied by the presentation of the **HARMONIC** directive.

## Relativistic Calculations (ZORA) I.

- All ZORA (**Z**ero **O**rders **R**egular **A**pproximation) effects are incorporated in the 1-electron integrals. Therefore ZORA can be used within all formalisms.
- ZORA calculations require an internal basis set to represent the relativistic kinetic energy. This basis set is generated automatically from the “normal” basis set.
- The ZORA kinetic matrix involves the inverse of the Coulomb matrix represented in the internal basis. In principle this matrix should be computed and inverted at every SCF cycle. In practice various approximations are available, the selection of which may be controlled by some of the ZORA subdirectives.
- Spin-orbit effects are only available at the Hartree-Fock level.

## Relativistic Calculations (ZORA) II.

- SCALE provides options for the ZORA orbital energy scale factors. Default is ATOM the alternative is MOLECULAR,  
`ZORA SCALE MOLECULAR`
- GAUGE provides a means for assessing gauge invariance errors e.g.,  
`GAUGE 1`  
should result in an energy shift of exactly one Hartree

```
Title  
BrF ground state ZORA  
zora  
zmatrix angstrom  
br  
f 1 brf  
variable  
brf 1.70  
end  
runtime optimize  
enter
```

zora.in

# Moller Plesset Theory: Energy Evaluation

- Invoked through SCFTYPE specification  
e.g. SCFTYPE MP2
- MP2
  - energy, gradient and hessian
  - closed shell and open shell (UHF-based only)
  - conventional and direct (closed-shells)
- MP3
  - energy, gradient
  - conventional only
- Restriction: natural orbitals and properties evaluated as part of gradient code - energy calculation gives SCF analysis

```

TITLE
H2CO - TZVP MP2/RHF
ZMATRIX ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
BASIS TZVP
SCFTYPE MP2
ENTER
  
```

mp2a.in

```

TITLE
H2CO+ - 2B2 TZVP MP2/UHF
MULT 2
CHARGE 1
ZMATRIX ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
BASIS TZVP
SCFTYPE MP2
ENTER
  
```

mp2b.in

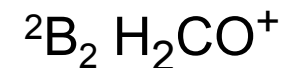
# Moller Plesset Theory: Gradient Evaluation



```
TITLE
H2CO - TZVP - MP2/RHF
ZMATRIX ANGSTROM
C
O 1 CO
H 1 CH 2 HCO
H 1 CH 2 HCO 3 180.0
VARIABLES
CO 1.203
CH 1.099
HCO 121.8
END
BASIS TZVP
RUNTYPE OPTIMIZE
PROPERTY ATOMS
SCFTYPE MP2
NATORB 20 PRINT
XTOL 0.0001
ENTER
```

mp2d.in

- Use parallel code for cases with 150+ GTOs (file usage)
- XTOL used to enhance accuracy of optimisation
- NATORB used to route natural orbitals to Dumpfile for subsequent property evaluation: PROPERTY ATOMS



```
TITLE
H2CO+ - 2B2 TZVP MP2/UHF
MULT 2
CHARGE 1
ZMATRIX ANGSTROM
C
O 1 CO
H 1 CH 2 HCO
H 1 CH 2 HCO 3 180.0
VARIABLES
CO 1.203
CH 1.099
HCO 121.8
END
BASIS TZVP
RUNTYPE OPTIMIZE
PROPERTY ATOMS
SCFTYPE MP2
NATORB 20
NATORB SPIN 21
XTOL 0.0001
ENTER
```

mp2e.in

# Moller Plesset Theory: Direct MP2 Implementations

## Serial Code

- Cost can be reduced by specifying a subset of all orbitals as the active set.
- Within this approach no derivatives are available.

## Parallel Code

- Direct implementation provides both energy and gradient functionality

```
TITLE
H2CO TZVP MP2/RHF (VALENCE)
ZMATRIX ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
BASIS TZVP
SCFTYPE DIRECT MP2
ACTIVE
3 TO 51 END
ENTER
```

mp2c.in

# MP2 Gradient Algorithms

## Parallel

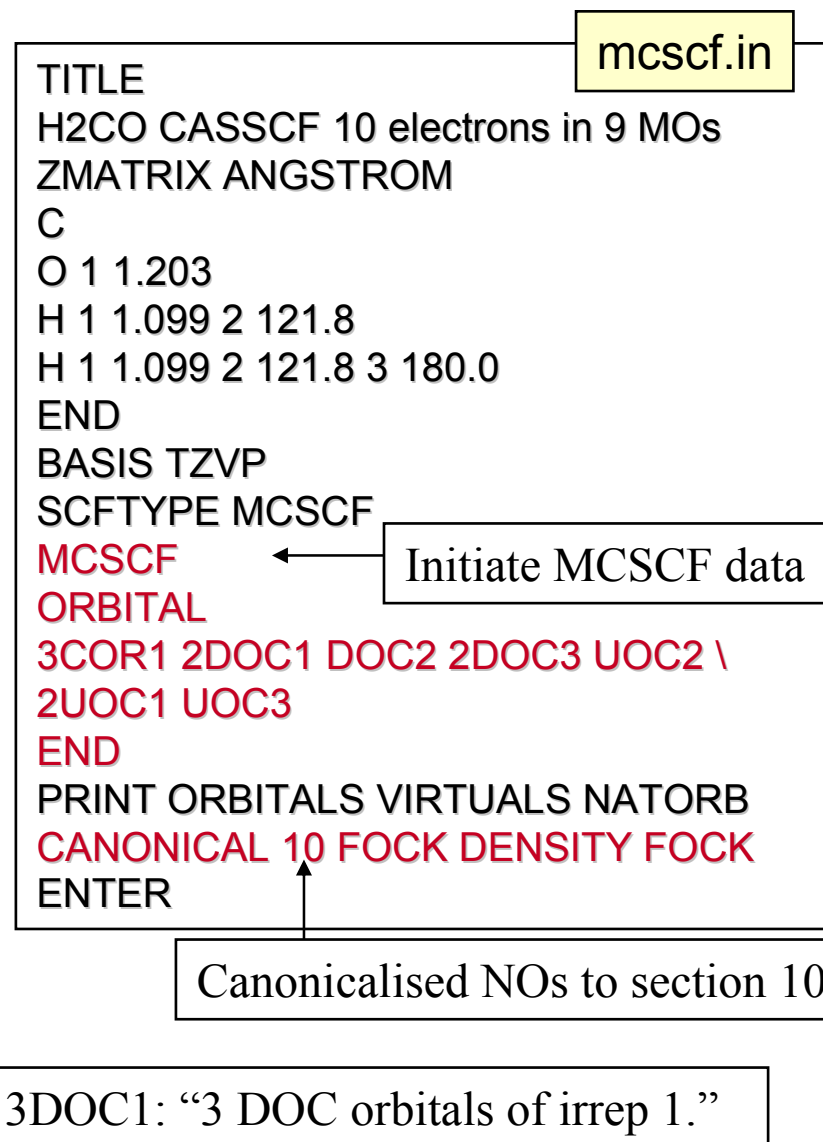
## Serial

- Conventional
  - integrals written to disk
  - read back, transformed, written out, resorted etc.
  - heavy I/O demands
- Direct/Semi-direct (Frisch, Head-Gordon & Pople, Hasse and Ahlrichs)
  - replace all/some I/O with batched integral recomputation

- Poor I/O-to-compute performance of MPPs
  - direct approach
- Current MPPs have large global memories
- Store subset of MO integrals
  - reduce number of integral recomputations
  - increase communication overhead
- Subset includes VOVO, VVOO, VOOO,
  - VVVO-class too large to store
  - compute VVVO-terms in separate step

# MCSCF and CASSCF

- Typically a two step approach
  - ✦ First calculate RHF orbitals
  - ✦ Initiate MCSCF with SCF MOs
- ORBITAL : Define initial configuration and active space by classifying SCF MOs
  - ✦ Possible orbital classifications:
    - FZC - doubly occupied and frozen throughout MCSCF.
    - COR - doubly occupied in all configurations, allowed to relax
    - DOC - doubly occupied (DOMOs)
    - ALP, BET - occupied by alpha or beta electron
    - UOC - unoccupied (VMOs)
- The DOC, ALP, BET and UOC orbitals define the active space



# Integral Transformation

We now turn our attention to the post-Hartree Fock modules within GAMESS-UK, considering initially the integral transformation routines and associated data input, and then the Direct-CI module.

Note that the transformation module acts not only as a precursor to Direct-CI, but finds more widespread usage in both OVGf and TDA Green's function calculations, and in the semi-direct Table-CI module.

Key Directives are **CORE** and **ACTIVE**:

**CORE**: factor frozen doubly occupied orbitals into the Fock Operator (F):

**ACTIVE**: specify members of the molecular orbital set which are deemed 'active' in the transformation, so that integrals of the form  $\langle ij/kl \rangle$  will be computed and output to the Transformed Integral File if all four MOs are specified using the ACTIVE directive.

tran1.in

```

TITLE
H2CO - TZVP - VALENCE-ONLY MOs
ZMATRIX ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
BASIS TZVP
RUNTYPE TRANSFORM
CORE
1 2 END
ACTIVE
3 TO 52 END
PASS 2 2
ENTER
  
```

Control size of scratch files (sort etc.)

# Direct-CI I.

- Direct-CI calculations are performed under control of RUNTYPE CI, with data input characterising the nature of the CI introduced by a data line with DIRECT in the first datafield.
- In all direct-CI calculations the user must specify the division of the MO space into an **internal** and **external** space, where the internal space comprises all orbitals that appear in any of the nominated reference functions. The latter are defined by use of the CONF directive.
- The set of NINT internal orbitals must appear first in the list of active orbitals, followed by the set of NEXT external orbitals.

```

TITLE
H2CO - 3-21G CISD DIRECT-CI
ZMATRIX ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
RUNTYPE CI
DIRECT 16 8 14
CONF
2 2 2 2 2 2 2
NATORB 12 0 PRINT
ENTER
  
```

dci0.in, dci1.in

**DIRECT NELEC NINT NEXT**

**NELEC**, number of active electrons in the CI (16 in this case)

**NINT**, the number of orbitals in the internal space (8 in this example, the doubly occupied SCF MOs)

**NEXT**, the number of orbitals in the external space (14 in this example, the SCF MOs)

# Direct-CI II.

- Each **reference function** in the CI is defined as a sequence of NINT integers specifying the orbital occupancy of each **internal orbital** in the function. This corresponds to a single data line, specified under the **CONF** directive, containing 8 data fields each set to the value 2 corresponding to the doubly occupied orbitals of the SCF reference function.
- The **NATORB** directive routes the spinfree natural orbitals (NOs) of the CI wavefunction to section 10 of the Dumpfile. The second integer refers to routing of the spin natural orbitals (set to 0 here for obvious reasons), while the PRINT keyword requests printing of the NOs
- Termination of this data is signalled by a valid Class 2 directive, such as VECTORS or ENTER.

```

TITLE
H2CO - 3-21G CISD DIRECT-CI
ZMATRIX ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
RUNTYPE CI
DIRECT 16 8 14
CONF
2 2 2 2 2 2 2 2
NATORB 12 0 PRINT
ENTER
  
```

**DIRECT NELEC NINT NEXT**

**NELEC**, 16 active electrons in the CI  
**NINT**, 8 orbitals in the internal space (the doubly occupied SCF MOs)  
**NEXT**, 14 external orbitals in the external space

# Direct-CI - $^2B_2$ H<sub>2</sub>CO<sup>+</sup>

Consider a direct-CI calculation on the  $^2B_2$  state of H<sub>2</sub>CO<sup>+</sup>, again using the SCF configuration as the sole reference function

- Of the three integers on the DIRECT data line, NINT and NEXT remain unchanged, while NELEC, the number of active electrons, is now 15.
- The occupation number of the 8th SCF MO on the CONF data line is now 1, reflecting the open shell orbital occupancy.
- The OPEN directive is now present, specified prior to the direct-CI data.
- An additional directive is required in the direct-CI data, SPIN, defining the spin multiplicity of the CI wavefunction.
- NATORB now routes the spinfree and spin NOs to sections 12 and 13 respectively of the Dumpfile.

```
TITLE
H2CO+ - 2B2 - CISD DIRECT-CI
CHARGE 1
MULT 2
ZMATRIX ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
RUNTYPE CI
OPEN 1 1
DIRECT 15 8 14
SPIN DOUBLET
CONF
2 2 2 2 2 2 1
NATORB 12 13 PRINT
ENTER
```

dci2.in

**DIRECT** NELEC NINT NEXT  
**NELEC**, 15 active electrons in the CI  
**NINT**, 8 orbitals in the internal space (the 7 doubly occupied + singly occ. SCF MOs)  
**NEXT**, 14 orbitals in the external space

# Direct-CI - Valence CI

- Now consider a valence-only CI, where we 'freeze' inner-shell orbitals, and discard certain virtual orbitals from the CI (the high-energy inner-shell complement orbitals).
- The CORE and ACTIVE directives are used to freeze both the O1s and C1s orbitals (sequence nos. 1 and 2 respectively) and to discard the corresponding inner-shell complement virtual orbitals (sequence numbers 21 and 22):
- The values of NELEC, NINT and NEXT specified on the DIRECT data line are modified to reflect the impact of CORE and ACTIVE.
- The CONF data line now comprises six integers specifying the double occupancy of the internal valence orbital set.

```

TITLE
H2CO - 3-21G CISD VALENCE-CI
ZMATRIX ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
RUNTYPE CI
CORE
1 2 END
ACTIVE
3 TO 20 END
DIRECT 12 6 12
CONF
2 2 2 2 2 2
NATORB 12 0 PRINT
ENTER
  
```

dci3.in

**DIRECT** NELEC NINT NEXT  
**NELEC**, 12 active electrons in the CI  
**NINT**, 6 orbitals in the internal space (the doubly occupied valence SCF MOs)  
**NEXT**, 12 orbitals in the external space

# Direct-CI - Multi Reference CI

Reference Function	1a <sub>1</sub>	2a <sub>1</sub>	3a <sub>1</sub>	4a <sub>1</sub>	1b <sub>2</sub>	5a <sub>1</sub>	1b <sub>1</sub>	2b <sub>2</sub>	2b <sub>1</sub>	6a <sub>1</sub>
1	2	2	2	2	2	2	2	2	0	0
2	2	2	2	2	2	2	0	2	2	0
3	2	2	2	2	2	0	2	2	0	2

dci4.in

- Assume that we wish to perform a 3-reference CI calculation for H<sub>2</sub>CO, comprising the SCF configuration, that arising from the double excitation 1b<sub>1</sub> to 2b<sub>1</sub> and that from the double excitation 5a<sub>1</sub> to 6a<sub>1</sub>
- Then the internal space must, in addition to the ground state doubly occupied SCF MOs, include the 2b<sub>1</sub> and 5a<sub>1</sub> virtual orbitals i.e., NINT=10.

## DIRECT

**NELEC**, 16 active electrons in the CI

**NINT**, 10 orbitals in the internal space (the doubly occupied SCF MOs plus the 2b<sub>1</sub> and 6a<sub>1</sub>)

**NEXT**, 12 orbitals in the external space

```

TITLE
H2CO - 3-21G CISD 3-REF CI
ZMATRIX ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
RUNTYPE CI
DIRECT 16 10 12
CONF
2 2 2 2 2 2 2 0 0
2 2 2 2 2 0 2 2 0
2 2 2 2 0 2 2 0 2
NATORB 12 0 PRINT
ENTER
  
```

# Direct-CI - Properties

- Computing the default set of one-electron properties at completion of CI processing may be readily accomplished through the addition of the PROPERTY ATOMS data line.
- Note that any such calculation requires the NATORB directive to specify the routing of the spinfree, and where relevant, the spin NOS to specified sections on the Dumpfile.
- The analysis is performed on the spinfree NOs as routed to section 12 of the Dumpfile

```
TITLE
H2CO - CISD PROPERTIES
ZMATRIX ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
RUNTYPE CI
PROPERTY ATOMS
DIRECT 16 8 14
CONF
2 2 2 2 2 2 2
NATORB 12 0 PRINT
ENTER
```

dci5.in

# Selection-driven Table-CI

Configuration selection Table-CI calculations are performed under control of RUNTYPE CI, with data input characterising the nature of the CI introduced by the data line "MRDCI DIRECT"

The Semi-direct Table-CI module comprises a set of **6 sub-modules**, which may be user-driven through input.

**TABLE**: generates an input a 'data-base' of pattern symbolic matrix elements for use in both the selection process and in solving the secular problem.

**SELECT**: performs configuration generation and selection based on a user-specified set of reference configurations and appropriate thresholds.

**CI**: provides initial pre-processing and calculation, in semi-direct fashion, of one or more CI eigenfunctions of the secular problem. Two secular problems are solved as part of the extrapolation process, one at  $T_{\min}$  and one at  $T_{\min} + T_{\text{inc}}$ .

**NATORB**: generate the spin-free natural orbitals for one or more of the calculated CI eigenvectors.

**PROP** & **TM** are optional, and may be used to further analyse one or more of the CI eigenvectors

```

TITLE
H2CO - semi-direct MRDCI 4M/1R
ZMAT ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
RUNTYPE CI
MRDCI DIRECT
TABLE
SELECT
SYMMETRY 1
SPIN 1
CNTRL 16
SINGLES ALL
CONF
0 1 2 3 4 5 13 17 18
0 1 2 3 4 5 14 17 18
0 1 2 3 4 5 13 17 19
4 13 14 18 19 1 2 3 4 5 17
END
ROOTS 1
THRESH 2 2
CI
NATORB
ENTER
  
```

mrdci0.in

# Selection-CI - Orbital Numbering and Symmetry

Reference Function	1a <sub>1</sub>	2a <sub>1</sub>	3a <sub>1</sub>	4a <sub>1</sub>	1b <sub>2</sub>	5a <sub>1</sub>	1b <sub>1</sub>	2b <sub>2</sub>	2b <sub>1</sub>	6a <sub>1</sub>	3b <sub>2</sub>
1	2	2	2	2	2	2	2	2	0	0	0
2	2	2	2	2	2	2	0	2	2	0	0
3	2	2	2	2	2	2	2	0	2	2	2
4	2	2	2	2	2	2	1	1	1	2	1

Consider initially a 4-reference CI calculation for H<sub>2</sub>CO, comprising the SCF configuration, that arising from the double excitation 1b<sub>1</sub> to 2b<sub>1</sub>, that from the double excitation 2b<sub>2</sub> to 3b<sub>2</sub> and that from the excitation 1b<sub>1</sub>2b<sub>1</sub> to (2b<sub>2</sub>3b<sub>2</sub>).

```

RUNTYPE CI
MRDCI DIRECT
TABLE
SELECT
SYMMETRY 1
SPIN 1
CNTRL 16
SINGLES ALL
CONF
0 1 2 3 4 5 13 17 18
0 1 2 3 4 5 14 17 18
0 1 2 3 4 5 13 17 19
4 13 14 18 19 1 2 3 4 5 17
END
    
```

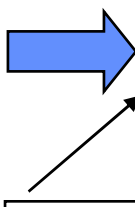
The MOs are now specified by symmetry (IRREP) ordering, the SCF output revealing:

```

=====
IRREP NO. OF SYMMETRY ADAPTED
      BASIS FUNCTIONS
=====
      1      12
      2       4
      3       6
=====
    
```

Orbital numbering

a <sub>1</sub>	1 - 12
b <sub>1</sub>	13 - 16
b <sub>2</sub>	17 - 22



No. of open shell MOs

# Selection-CI. Default Options

- In order to simplify the process of configuration specification and data preparation, the semi-direct module provides a set of default options that require little or no data input. These provide a starting point for users, and a route to more extensive studies.
- A Semi-direct Table-CI calculation is to be performed on the formaldehyde molecule.
- Default reference set comprises the SCF configuration plus, for each symmetry IRREP., the single and double excitation from the highest occupied to lowest virtual MO.

```
TITLE
H2CO - DEFAULT TABLE-CI
OPTIONS
ZMAT ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
RUNTYPE CI
MRDCI DIRECT
ENTER
```

mrDCI1.in

# Selection-CI. Expanding the Default Options

```
TITLE
H2CO - DEFAULT TABLE-CI
ZMAT ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
RUNTYPE CI
MRDCI DIRECT
ENTER
```



Full specification  
corresponding to  
the defaults

```
TITLE
H2CO - EXPLICIT DATA
ZMAT ANGSTROM
....
END
RUNTYPE CI
ACTIVE
1 TO 22 END
MRDCI DIRECT
TABLE
SELECT
CNTRL 16
SPIN 1
SYMM 1
SINGLES ALL
CONF
0 1 2 3 4 5 13 17 18
0 1 2 3 4 6 13 17 18
2 5 6 1 2 3 4 13 17 18
0 1 2 3 4 5 14 17 18
2 13 14 1 2 3 4 5 17 18
0 1 2 3 4 5 13 17 19
2 18 19 1 2 3 4 5 13 17
END
THRESH 10 10
ROOTS 1
CI
NATORB
CIVEC 1
ENTER
```

mrddci2.in

# Coupled Cluster: CCSD and CCSD(T)

- Coupled Cluster (CC) calculations are performed under control of RUNTYPE CI, with the nature of the CC defined by a data line with CCSD or CCSD(T) in the first datafield.
- Coupled Cluster treatments are limited to closed shell systems; no gradients are available at present.
- Extensive unformatted FORTRAN file utilisation (ftxx etc).
- ACTIVE and CORE directives may be used to control the final set of MOs to feature in the calculation. Then:

CCSD(T) NACT NALPHA NBETA

```
TITLE
H2CO - TZVP CCSD
ZMATRIX ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
BASIS TZVP
RUNTYPE CI
CCSD(T)
ENTER
```

ccsd0.in

```
TITLE
H2CO - TZVP valence CCSD
ZMATRIX ANGSTROM
C
O 1 1.203
H 1 1.099 2 121.8
H 1 1.099 2 121.8 3 180.0
END
BASIS TZVP
ACTIVE \ 3 TO 50 \ END
CORE \ 1 TO 2 \ END
RUNTYPE CI
CCSD(T) 48 6 6
ENTER
```

ccsd1.in

# Solvation - DRF

- The Direct Reaction Field (DRF) solvation model allows one to place a QM molecule in an environment consisting of other classical molecules and dielectric media.
- The example uses the dielectric constant of water to model the solvent.
- The Connolly surface is an access type surface.
- In contrast to other modules, DRF requires the following to be set:

**INTEGRAL HIGH**

**ADAPT OFF**

**NOSYMM**

```

title
h2o surrounded by dielectric
# connolly vdWals surface
nosymm
integral high } ← Mandatory
adapt off
geom au
  0.0      0.0      0.0 8.0 O
  1.1031   1.4325   0.0 1.0 H
  1.1031  -1.4325   0.0 1.0 H
end
basis sv 4-31g**
react ← DRF initiator
field reac scf
drftwoel disk
dielectric
dieltyp stat
solvent water
solvradi 0.0
connolly rprobe 3.64
end
end
  
```

solvation.in

## Other RUNTYPE's

- POLARISABILITY
  - ⌘ Analytic calculations of molecular polarisabilities may be conducted at both the SCF and MP2 levels. SCF frequency dependent polarisabilities are also available.
- HYPERPOLARISABILITY
  - ⌘ Analytic calculations of molecular hyperpolarisabilities may be conducted for both closed-shell SCF and open-shell RHF wavefunctions. These are third derivatives of the energy, and require preceding calculations of the first derivative wavefunctions.
- MAGNETISABILITY
  - ⌘ Analytic calculations of molecular magnetisabilities may be conducted for closed-shell SCF wavefunctions only.
- RAMAN
  - ⌘ The calculation of Raman Intensities is only available for closed-shell SCF wavefunctions

# Multiple RUNTYPE Specification I.

- All examples so far have assumed that a given run of the program is concerned with the processing associated with a single RUNTYPE.
- Also possible to issue multiple RUNTYPEs within a single run of the programme.
- Multiple RUNTYPE specification is subject to the constraint that the Class 1 directives (ZMATRIX, BASIS) appear only once in the job
- Geometry is inherited from the preceding RUNTYPE c.f. geometry optimisation ...

## \* Class 1 Directives

TITLE

ZMATRIX

BASIS

## \* Class 2 Directives (1st task)

RUNTYPE

SCFTYPE

ENTER

## \* Class 2 Directives (2nd task)

RUNTYPE

SCFTYPE

ENTER

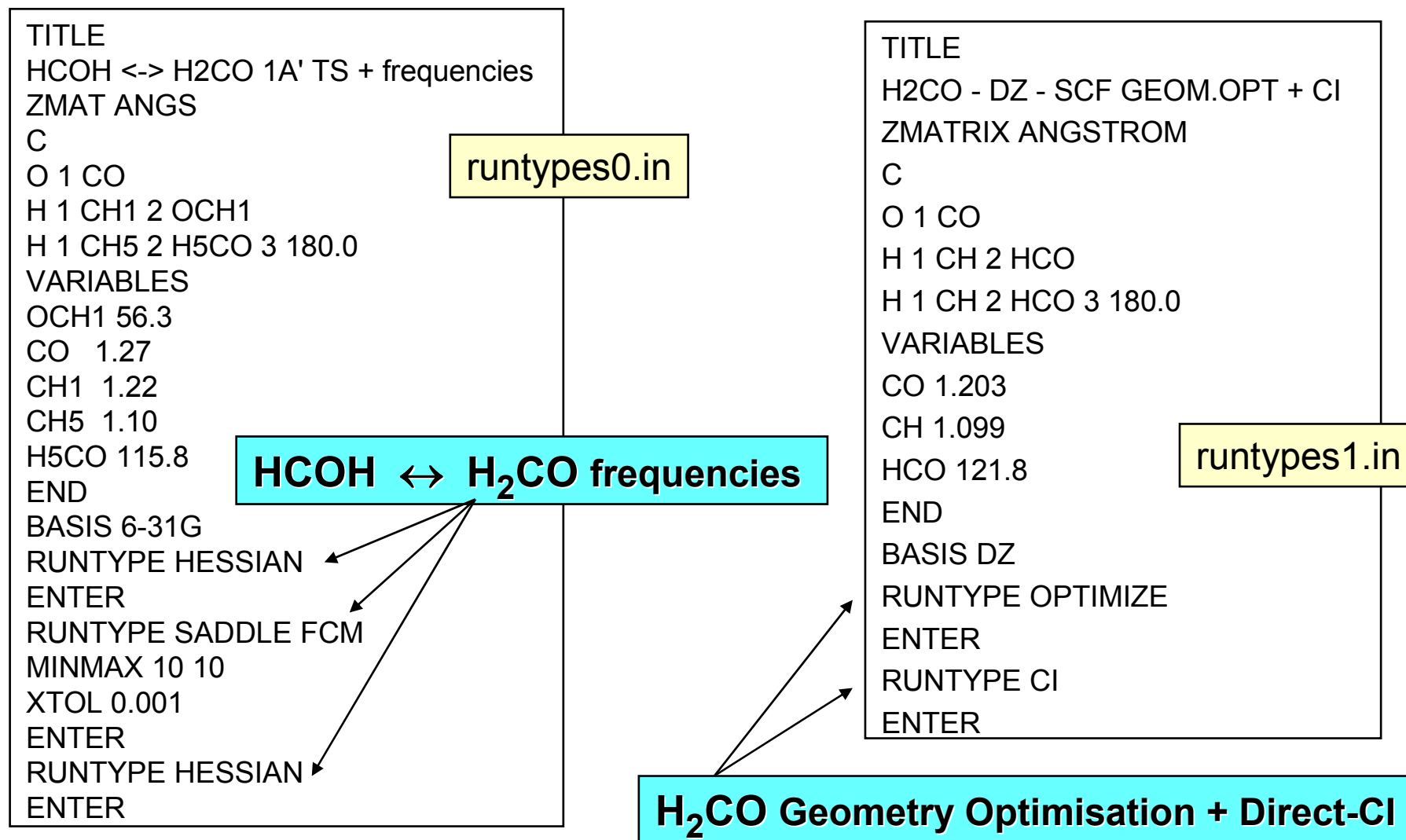
## \* Class 2 Directives (3rd task)

RUNTYPE

SCFTYPE

ENTER

# Multiple RUNTYPE Specification II.



# Other RUNTYPE's: Examples

runtypes2.in

```
TITLE
H2CO DZ MP2 polarisability
ZMATRIX ANGSTROM
C
O 1 CO
H 1 CH 2 HCO
H 1 CH 2 HCO 3 180.0
VARIABLES
CO 1.203
CH 1.099
HCO 121.8
END
BASIS DZ
RUNTYPE OPTIMISE
SCFTYPE MP2
XTOL 0.001
ENTER
RUNTYPE POLARISABILITY
SCFTYPE MP2
ENTER
```

**H<sub>2</sub>CO: MP2 Polarisability**

```
TITLE
H2CO default basis SCF Raman
ZMATRIX ANGSTROM
C
O 1 CO
H 1 CH 2 HCO
H 1 CH 2 HCO 3 180.0
VARIABLES
CO 1.203
CH 1.099
HCO 121.8
END
RUNTYPE OPTIMISE
XTOL 0.001
ENTER
RUNTYPE RAMAN
ENTER
```

runtypes3.in

**H<sub>2</sub>CO SCF Raman Intensities**

# Other Analysis Codes 1. NBO

- Both Natural Bond Orbital (NBO) and AIMPAC (Bader) analysis codes are available.
- **NBO**: code integrated into GAMESS-UK at configuration time and driven under RUNTYPE ANALYSE.
- Example illustrates multiple RUNTYPE specification
- module triggered by *nbo* directive
- unlike other modules, nbo data is presented after the enter directive
- syntax of data dependent on run-time environment (*rungames*)

```
title
Methylamine .. RHF/3-21G NBO analysis
zmatrix angstrom
C
N 1 CN
H 1 CH 2 tet
H 1 CH 2 tet 3 120.0
H 1 CH 2 tet 3 240.0
H 2 NH 1 tet 3 60.0
H 2 NH 1 tet 3 300.0
variables
CN 1.47
CH 1.09
tet 109.4712
NH 1.01
end
enter
runtype analyse
nbo
vectors 1
enter 1
nbo end
```

nbo0.in, nbo1.in

# Other Analysis Codes 2. AIMPAC

F.W. Biegler-Konig and R.F.W.Bader,  
*J. Comp. Chem. 13 (1982)*

- **AIMPAC (Version 94 Revision B):** accessed through file generation by GAMESS-UK to be subsequently input to the Bader analysis codes e.g. *proaimv* and *extreme*
- **savefile** directive specifies file

*proaimv* c4h4 c4h4v.gms

Integration input  
file, c4h4.inp

Wavefunction file  
from GAMESS-UK

```

title
C4H4 AIMPAC file generation
geometry
  0.977560  0.977560  0.977560  6  C
 -0.977560 -0.977560  0.977560  6  C
  0.977560 -0.977560 -0.977560  6  C
 -0.977560  0.977560 -0.977560  6  C
  2.138065  2.138065  2.138065  1  H
 -2.138065 -2.138065  2.138065  1  H
  2.138065 -2.138065 -2.138065  1  H
 -2.138065  2.138065 -2.138065  1  H
end
basis 6-31g*
savefile aimpac file c4h4.gms.wfn
section 1
enter
  
```

aimpac.in

# Parallel Implementation of GAMESS-UK

- Early implementation based on message passing
- Subsequent activities under HEC Facilities Agreement with support from European projects
  - ✧ **IMMP (1994-1997, part of EUROPORT)**
    - Partners: Guest, Sherwood (Daresbury) - GAMESS-UK, Baerends (Amsterdam) - ADF, Clark (Erlangen) - VAMP
    - Focus on MPP systems (e.g. Cray T3E)
    - Mapping of disk files into global memory (uses GAs)
    - First MPP MP2 algorithm
      - GA storage of transformed integrals
  - ✧ **QUASI (1998-2001)**
    - Application of QM/MM methods in Industry
    - Led by Daresbury, Partners: Catlow (RI), Thiel (MPI), BASF, ICI, Hydro
    - Focus on commodity systems, cost-effective computing in industry
      - demonstrated using Linux alpha commodity cluster at Daresbury.

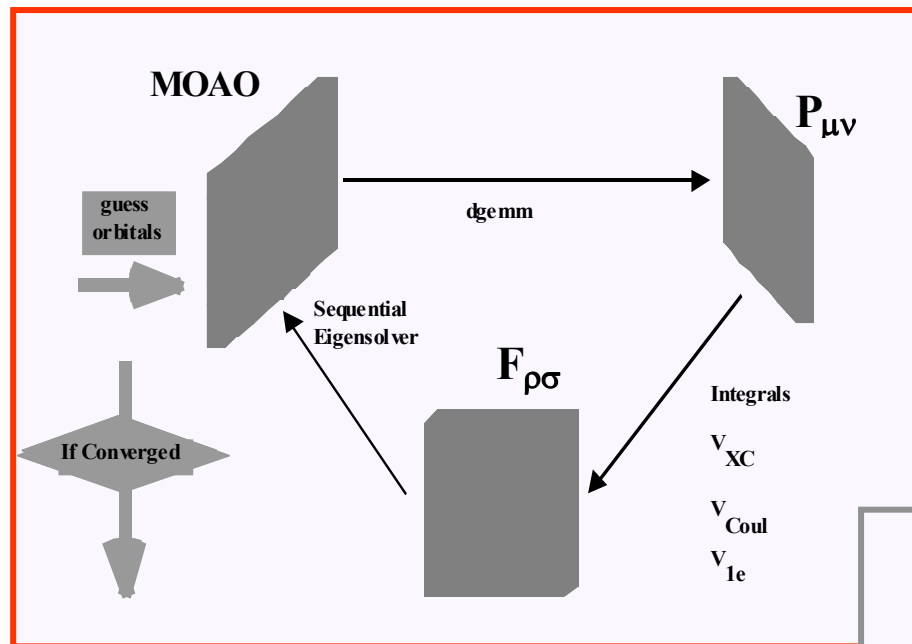
# Large-Scale Parallel *Ab-Initio* Calculations

- GAMESS-UK now has two parallelisation schemes:
  - ⌘ The traditional version based on the Global Array tools
    - **retains a lot of replicated data**
    - **limited to about 4000 atomic basis functions**
  - ⌘ More recent developments by Ian Bush (High Performance Applications Group, Daresbury) have extended the system sizes available for treatment by both GAMESS-UK (molecular systems) and CRYSTAL (periodic systems)
    - **Partial introduction of “Distributed Data” architecture...**
    - **MPI/ScaLAPACK based**

## Replicated and Distributed Data Parallel Strategies

- Replicated Data Strategy (RD)
  - ⌘ Not efficient for large parallel computers due to both system size and parallel scalability limitations.
  - ⌘ Largest problem governed by the memory available to one CPU, not that of the whole machine.
  - ⌘ Communications overhead – results of parallel OP must be re-replicated so that each CPU has a current version i.e. expensive global communication - time taken grows with the number of CPUs.
  - ⌘ RD does not fit well with many algorithms e.g. diagonalization - increasingly important stage of SCF/DFT calculations as the system size increases.
- Distributed Data Strategy (DD) - NWChem
  - ⌘ Each CPU holds only part of each of the (major) data structures, overcoming many of the drawbacks of replicated data;
    - Limit on system size is now that given by all memory on the computer,
    - global communications largely avoided as re-replication is not required,
    - many of the algorithms that can not be effectively addressed by a RD strategy fit more naturally within a distributed data one.

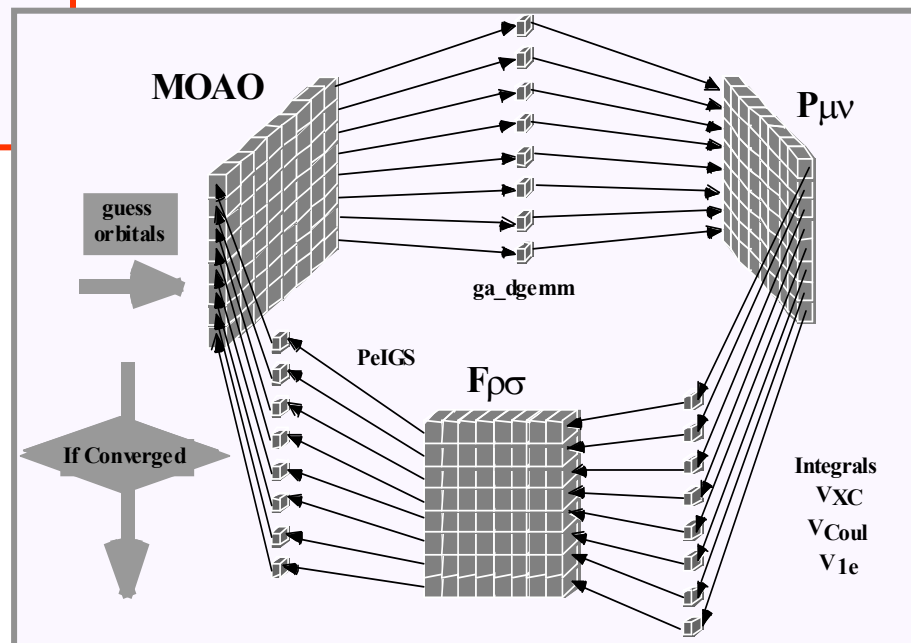
# Larger Problems - Distributed Data SCF



**Sequential**

**Distributed Data**

Pictorial representation of the iterative SCF process in (i) a sequential process, and (ii) a distributed data parallel process: **MOAO** represents the molecular orbitals, **P** the density matrix and **F** the Fock or Hamiltonian matrix



# Parallel SCF & DFT: Replicated vs Distributed Data Strategies

$$F_{\mu\nu} = H_{\mu\nu}^0 + \sum_{\lambda\sigma} P_{\lambda\sigma} [(\mu\nu | \sigma\lambda) - 1/2(\mu\lambda | \sigma\nu)]$$

**Square Matrices**      **Integrals (computed on the Fly)**

## Replicated Data e.g. GAMESS-UK

- F, H, P held on all nodes
- + Simple and easy to load balance
- Largest problem is governed by the memory available to one CPU, (not that of the whole machine)
- Harder to parallelise matrix algebra (results of parallel operation must be re-replicated so that each CPU has a current version)

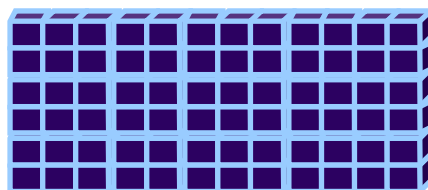
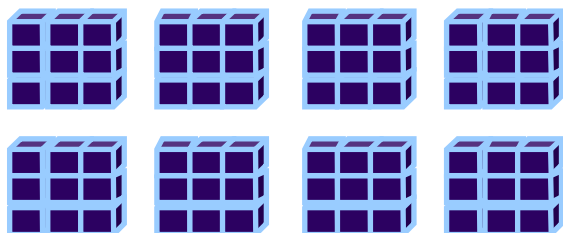
## Distributed Data e.g. NWChem

- F, H, P distributed
- + Limit on system size is now that given by all memory on the computer,
- + global communications largely avoided as re-replication is not required,
- For any integral, part of F or P is likely to be remote
  - *require many 1-sided communications (e.g. GA tools) or complex systolic strategy*
  - *problems with load balancing!*



# Global Arrays

Physically distributed data



Single, shared data structure

- Shared-memory-like model
  - Fast local access
  - NUMA aware and easy to use
  - MIMD and data-parallel modes
  - Inter-operates with MPI, ...
- BLAS and linear algebra interface
- Ported to major parallel machines
  - IBM, Cray, SGI, clusters, ...
- Originated in an HPCC project
- Used by 5 major chemistry codes, financial futures forecasting, astrophysics, computer graphics

Tools developed as part of the NWChem project at PNNL; R.J. Harrison, J. Nieplocha et al.

LAPI-based implementation on the p5-575

## Parallel Implementation of GAMESS-UK

- Current approach between the extremes of replicated and distributed data.
- Most data is replicated - when a parallel linear algebra OP is to be performed the data is copied into a global array, GA tools used to perform the operation, and the data then copied back into a replicated object.
  - ✘ SCF and DFT
    - GA Tools for caching of I/O for restart and checkpoint files
    - Storage of 3-centre 2-e integrals in DFT Jfit
    - Linear Algebra (via Parallel Linear Algebra from NWChem Project – PeIGs) - DIIS/MMOs, Inversion of Fitting matrix)
  - ✘ MP2 gradients, SCF and DFT second derivatives
    - Distribution of <vvo> and <vov> integrals via GAs
  - ✘ Direct RPA Excited States
    - Replicated data with parallelisation of direct integral evaluation
- This solves some of the problems associated with a RD strategy, but large memory requirements and the re-replication overheads remain.
- .. move to a truly distributed data model ..

# Parallel Linear Algebra

## Symmetric Eigensolver Routines

### ScaLAPACK

- drivers for solving standard and generalized dense symmetric or dense Hermitian Eigenproblems.
- PDSYEV (QR Method) (Scalapack 1.5)
- PDSYEVX (Bisection & Inverse Iteration) (Scalapack 1.5)
- PDSYEVD (D&C Method) (Scalapack (1.7))
- PDSYEVR (MRRR Method) (Future Release)
  - eXpert driver (as PDSYEVX)
  - Reduced memory overheads ( $O(N)$ )

- PDSBTDC (Block D&C Method) (Future Release)
  - Approximate full matrix by block tridiagonal
  - Performance gains at cost of accuracy

### PeIGS (PNNL)

- General symmetric and standard symmetric eigenproblems
  - PDSPEV (Bisection & Inverse Iteration)

### BFG (I. Bush)

- Block Jacobi Method on full dense symmetric matrix (+ Hermitian)

### Plapack

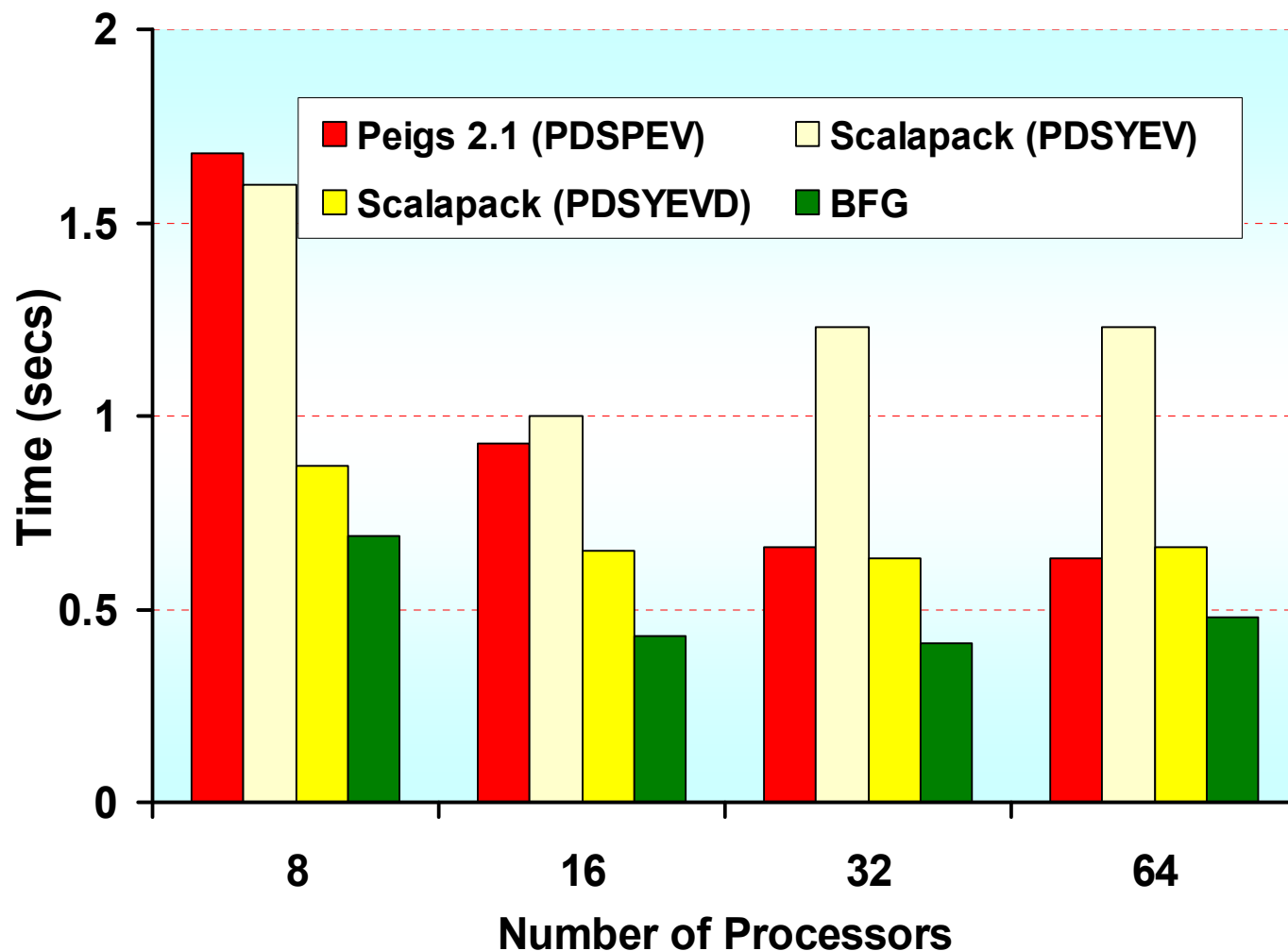
- QR method
- MRRR 'Multiple Relatively Robust Representations'

# Symmetric Eigensolver Approach

- The solution to the real symmetric Eigensolver problem usually takes place via three main steps
  - ✘ Reduction of the matrix to tri-diagonal form, typically using the Householder Reduction  $O(n^3)$
  - ✘ Solution of the real symmetric tri-diagonal Eigenproblem via one of the following methods:
    - Bisection for the Eigenvalues and inverse iteration for the Eigenvectors, up to  $O(n^3)$
    - QR algorithm, up to  $O(n^3)$
    - Divide & Conquer method (D&C), up to  $O(n^3)$
    - Multiple Relatively Robust Representations (MR3 algorithm),  $O(nk)$
  - ✘ Back substitution to find Eigenvectors for the full problem.  $O(n^3)$
- Other Methods
  - ✘ Jacobi Method  $O(n^3)$
  - ✘ Symmetric Subspace Decomposition Algorithm

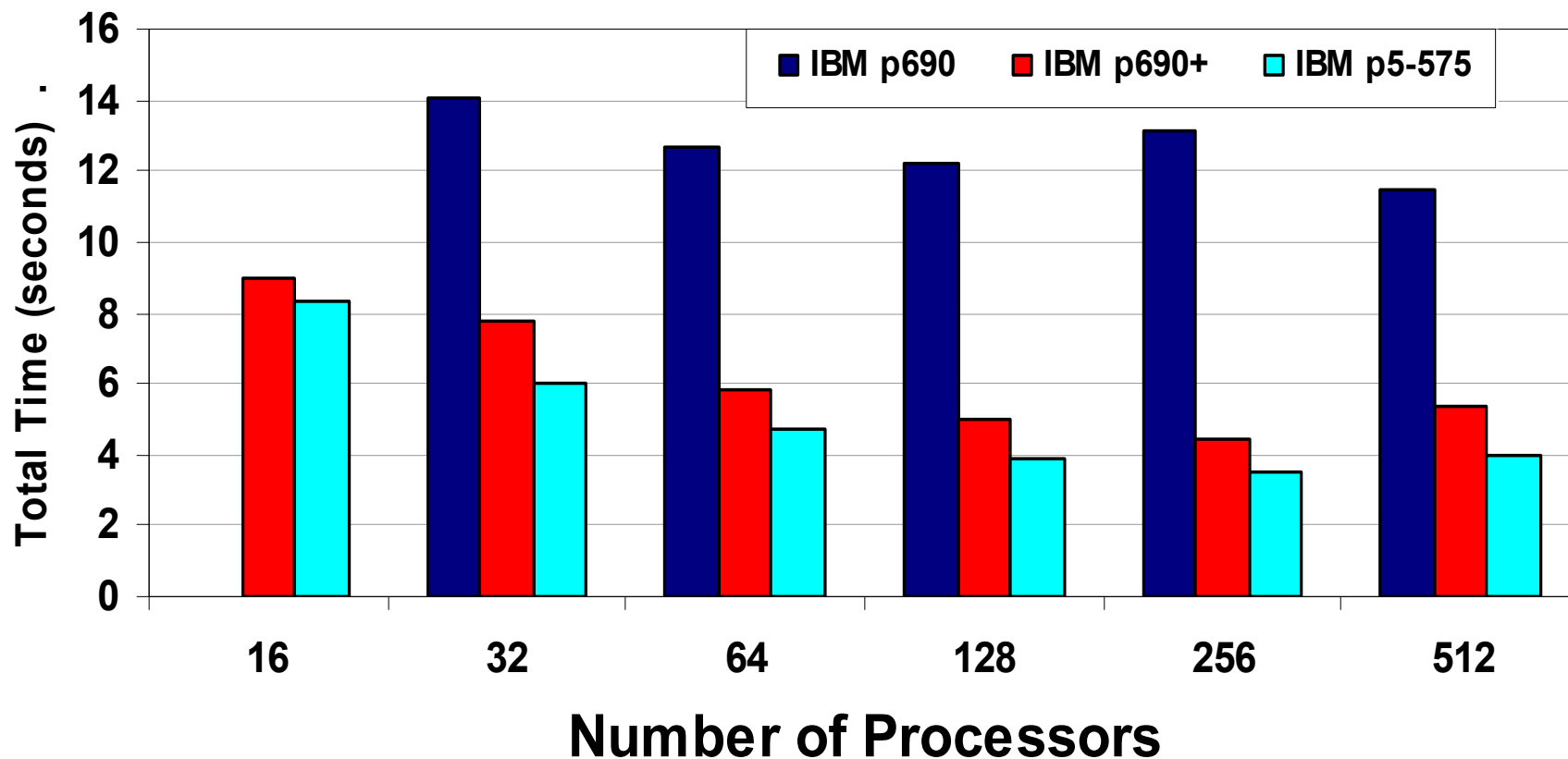
# Performance on IBM p690 - "Small" case

Fock Matrix, N = 1152



# Parallel Linear Algebra - Eigensolvers

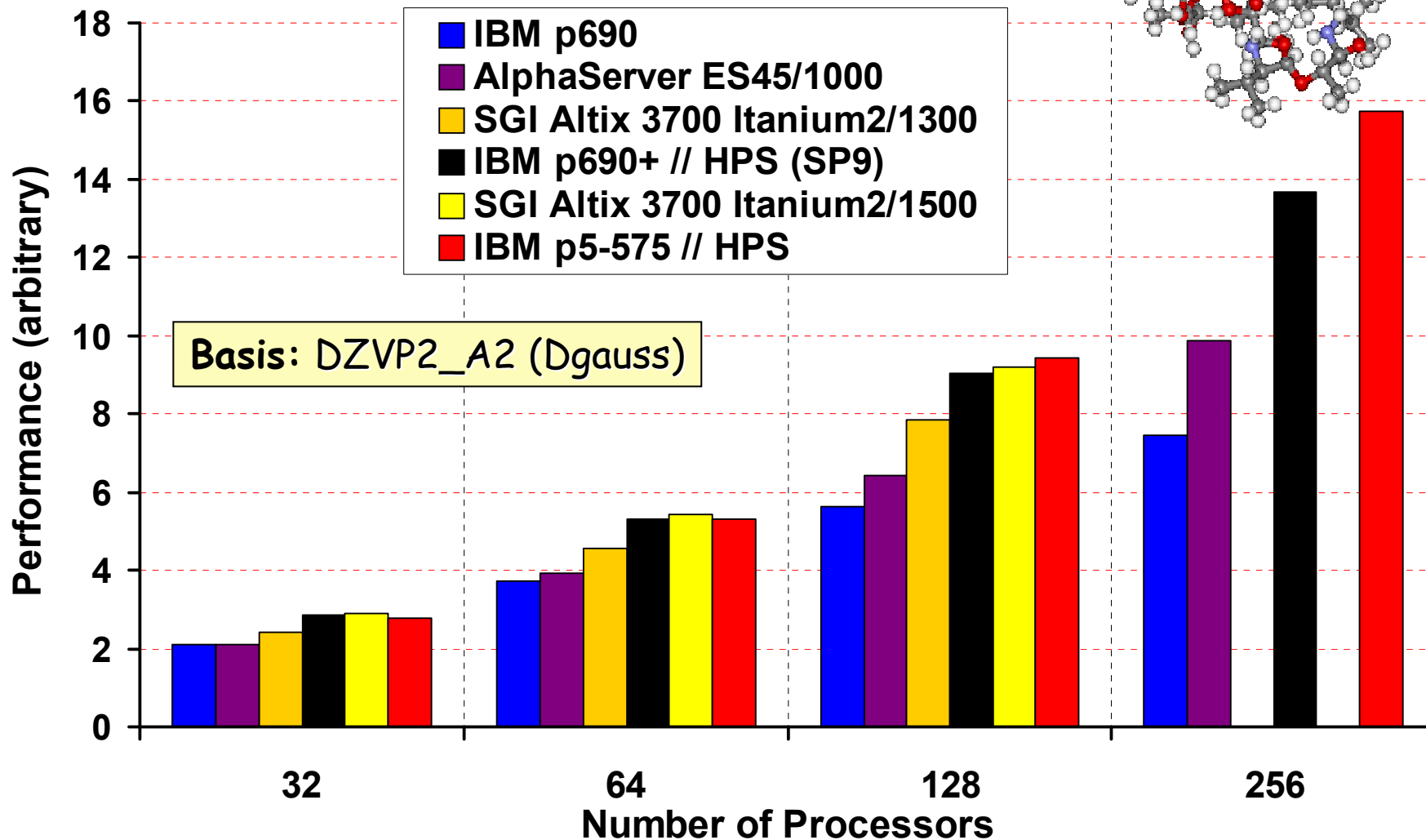
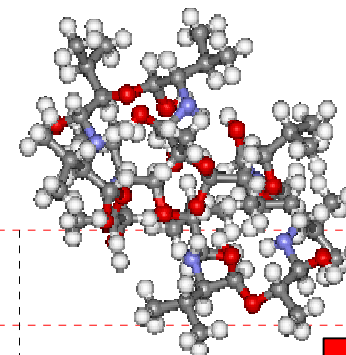
PDSYEVD Performance for Fock Matrix, (CRYSTAL), N\_basis = 3888



# Performance of the GA-based Implementation

Valinomycin, 1620 GTOs

DFT HCTH: IBM p5-575 and High-end Systems



## Auxiliary Basis Coulomb Fit (I)

The approach is based on the expansion of the charge density in an auxiliary basis of Gaussian functions

$$\rho(r) = \sum_{pq} D_{pq} |pq\rangle \approx \sum_u \left( \sum_{pq} D_{pq} C_u^{pq} \right) |u\rangle = \sum_u d_u |u\rangle$$

As suggested by Dunlap, a variational choice of the fitting coefficients  $C$  can be obtained as follows:

$$C^{pq} = V^{-1} \mathbf{b}^{pq}$$

Where  $V$  is the matrix of 2-centre 2-electron repulsion integrals in the charge density basis and  $b$  are the three centre electron repulsion integrals between the wavefunction basis set and the charge density basis.

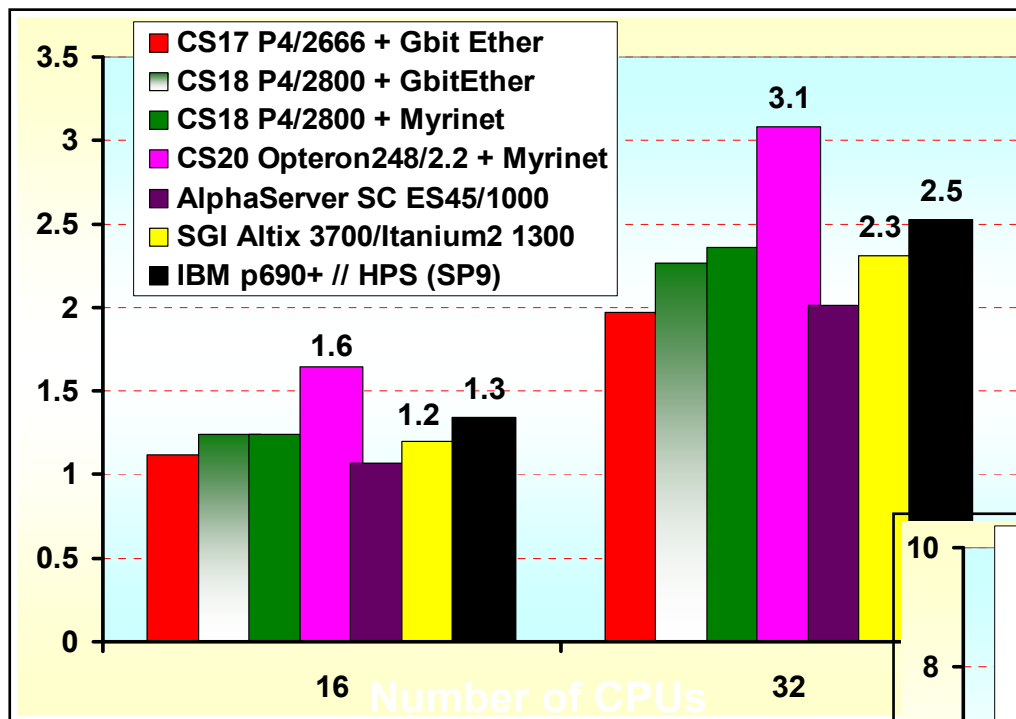
## Auxiliary Basis Coulomb Fit (ii)

- The number of 3-centre integrals is significantly smaller than the 4-centre integrals used in the conventional coulomb evaluation, but for large molecules additional screening is required.
- We make use of the Schwarz inequality

$$(pq|u) \leq \sqrt{(pq|pq)}\sqrt{(u|u)}$$

- Where p and q are AO basis functions and u are the fitting functions. Since screening is applied on a shell basis, the maximal integrals for each shell quartet are stored.
- Using this screening, and exploiting the aggregate memory of a parallel machine, it is possible to hold a significant fraction of the 3-centre integrals in core.

Performance

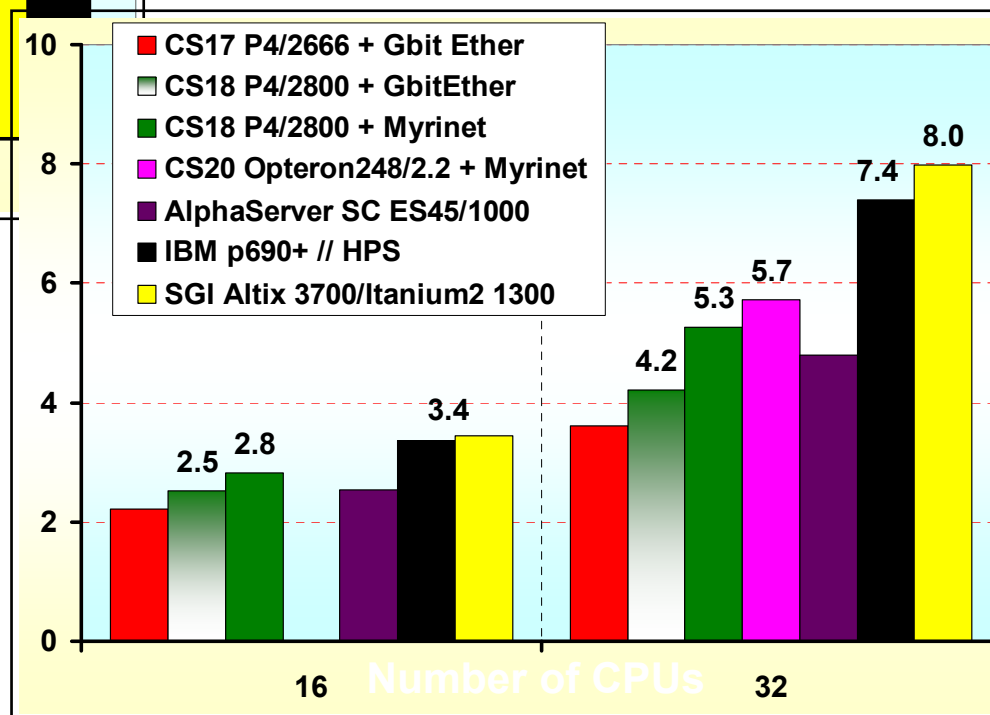


# GAMESS-UK: DFT HCTH on Valinomycin. Impact of Coulomb Fitting



$T_{32}$  SGI Altix 3700/1500 = 237 secs.

Performance



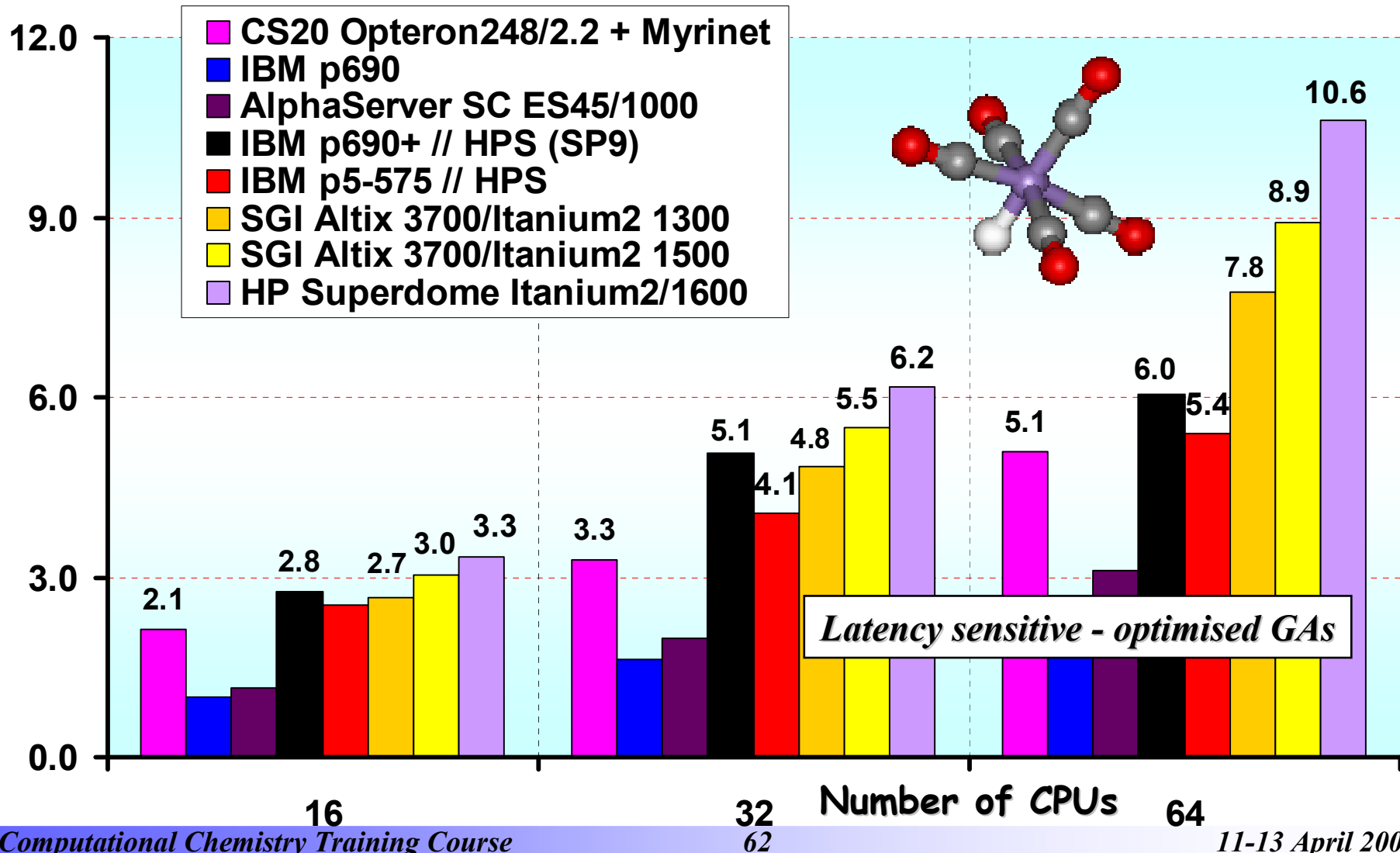
$T_{32}$  SGI Altix 3700/1500 = 780 secs.

Basis: DZV\_A2 (Dgauss)  
A1\_DFT Fit: 882/3012

# Performance of MP2 Gradient Module:

Mn(CO)<sub>5</sub>H - MP2 geometry optimisation  
BASIS: TZVP + f (217 GTOs)

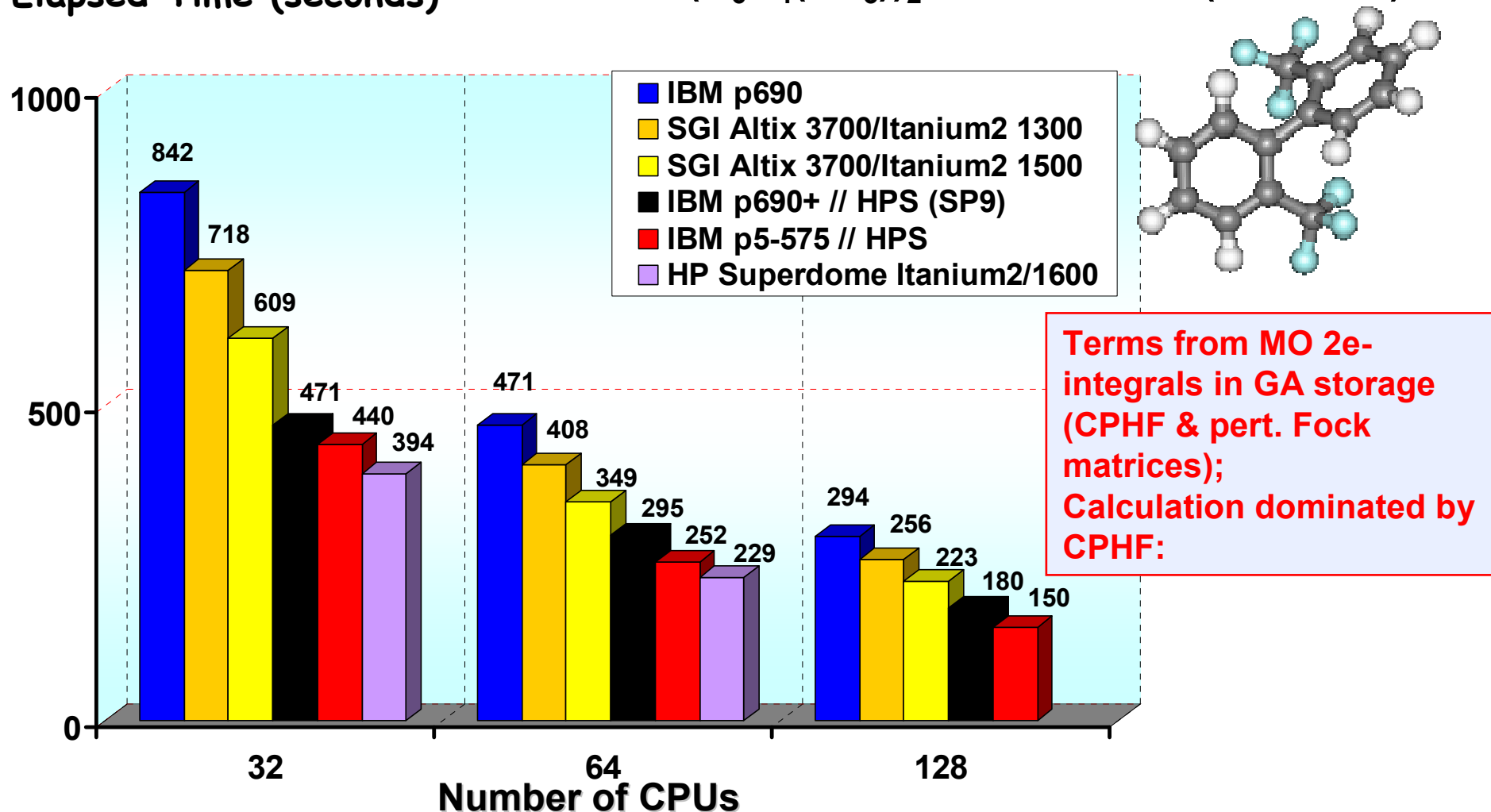
Performance



# DFT Analytic 2nd Derivatives Performance

## High-end Systems - HCTH functional

Elapsed Time (seconds) (C<sub>6</sub>H<sub>4</sub>(CF<sub>3</sub>)<sub>2</sub>): Basis 6-31G (196 GTO)



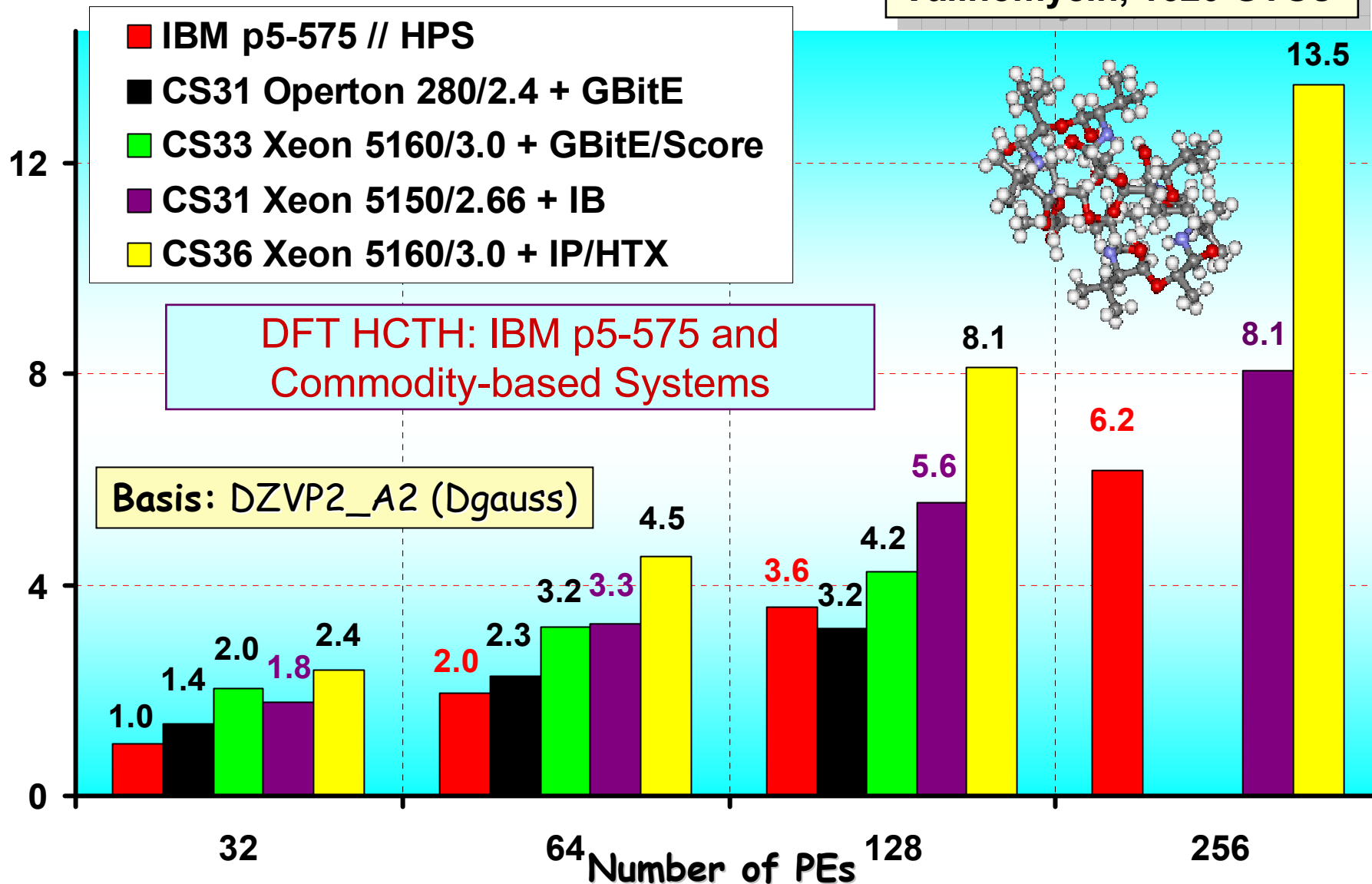
## Moving to Distributed Data. The MPI/ScaLAPACK Implementation of the GAMESS-UK SCF/DFT module

- Alternative pragmatic approach in which
  - ✘ MPI-based tools (such as ScaLAPACK) used in place of GA and LAPACK
  - ✘ All data structures except those required for the Fock matrix build are fully distributed (F, P)
- Partially distributed model chosen because, in the absence of efficient one-sided communications it is difficult to efficiently load balance a distributed Fock matrix build.
- Obvious drawback - some large replicated data structures are required.
  - ✘ These are kept to a minimum. For a closed shell HF or DFT calculation only two replicated matrices are required, one Fock and one density (doubled for UHF).
  - ✘ Further the symmetry of these matrices is used to cut down on the required memory.

# Performance of the MPI/ScaLAPACK code I.

Performance

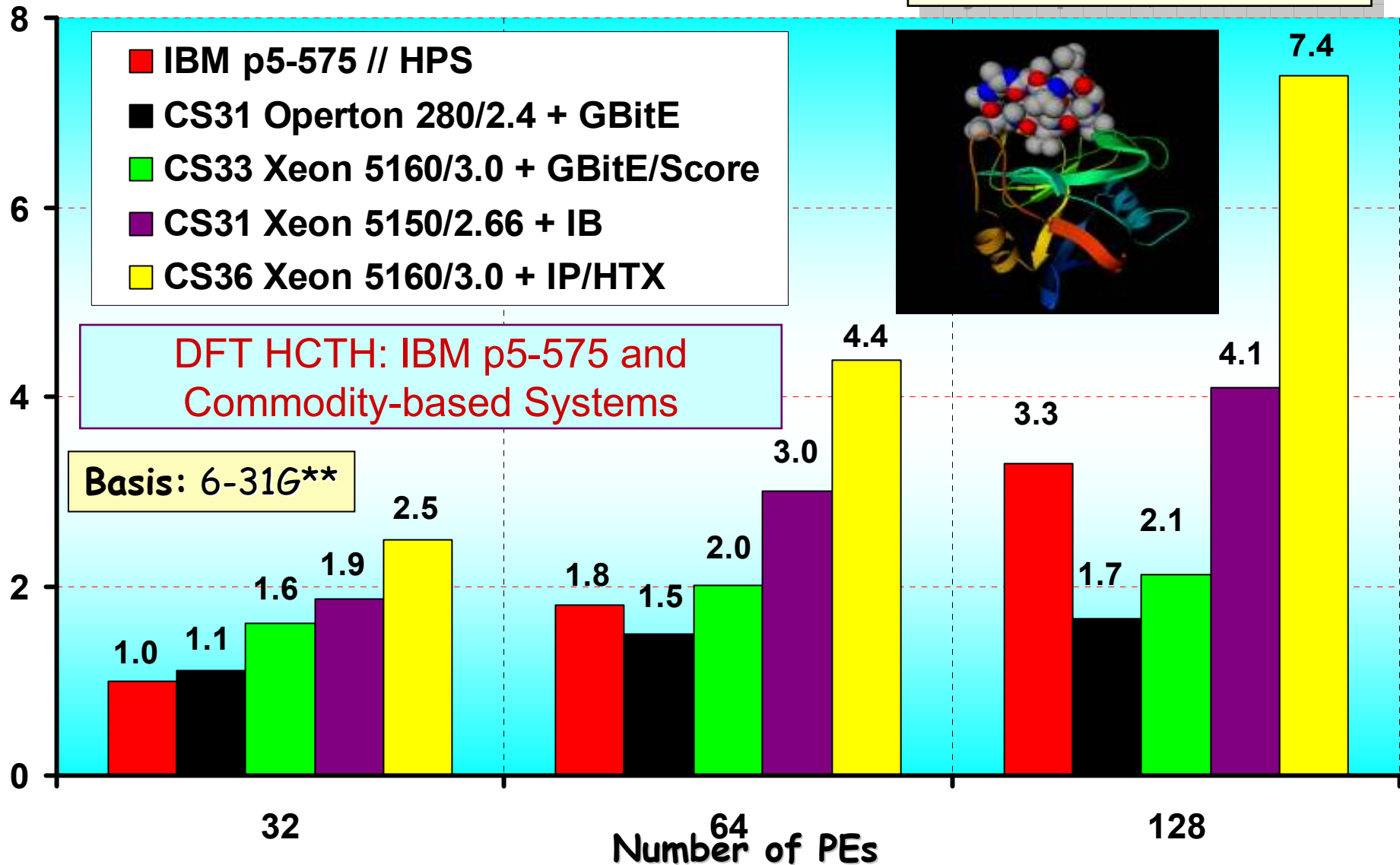
Valinomycin, 1620 GTOs



# Performance of the MPI/ScaLAPACK code II.

Performance

Cyclosporin, 1855 GTOs

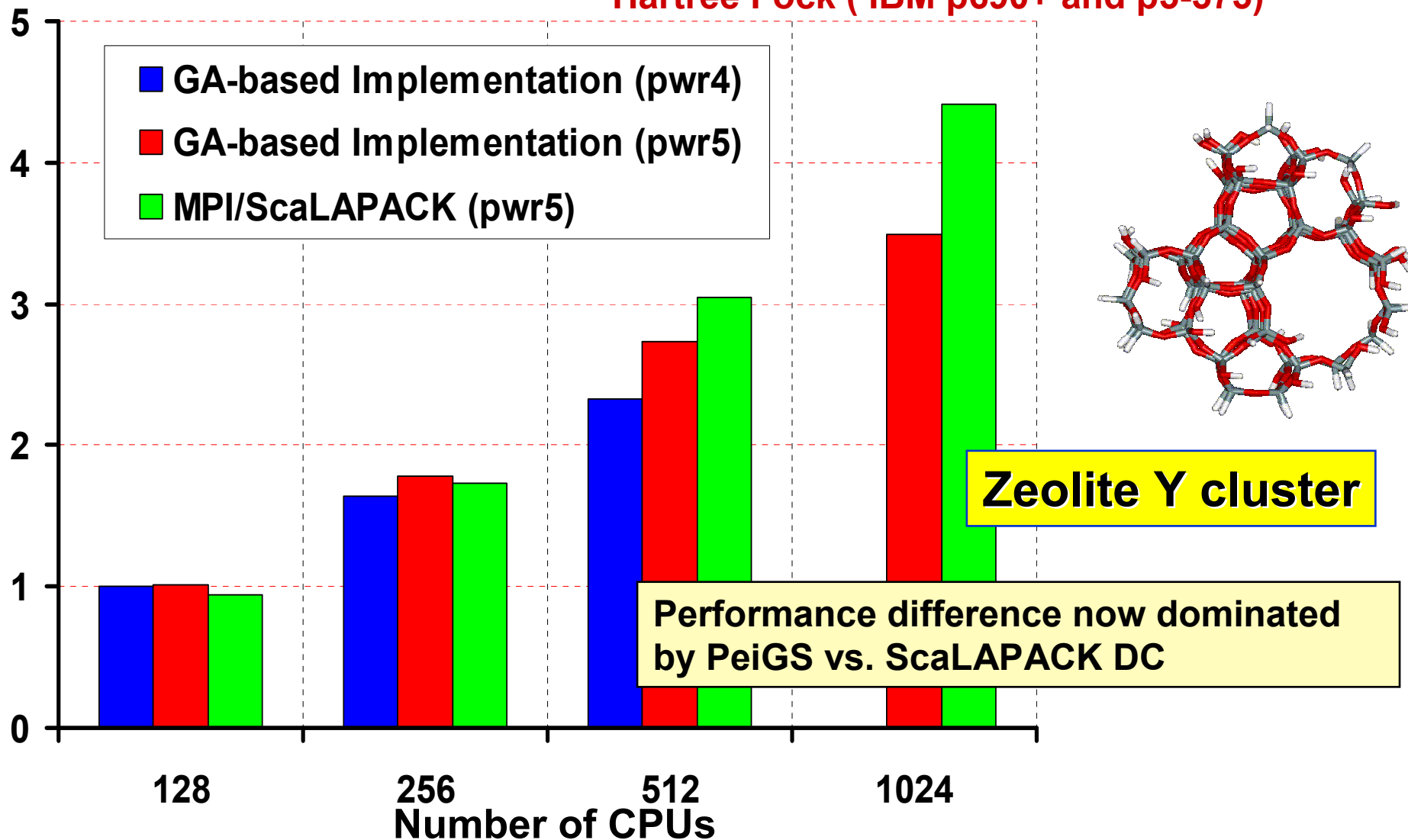


# GAMESS-UK Global Array & MPI/ScaLAPACK

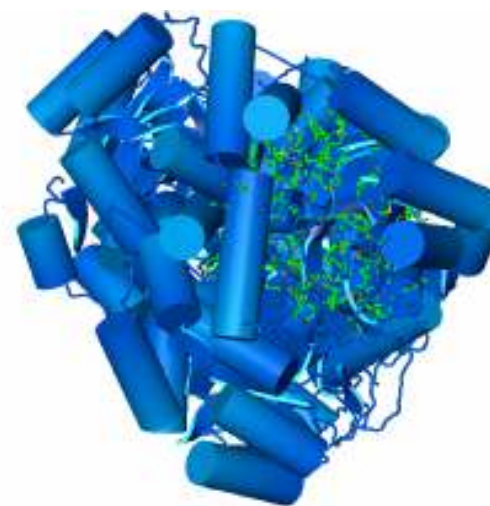
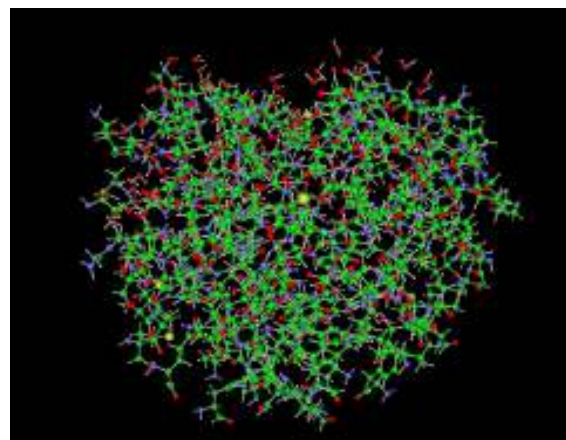
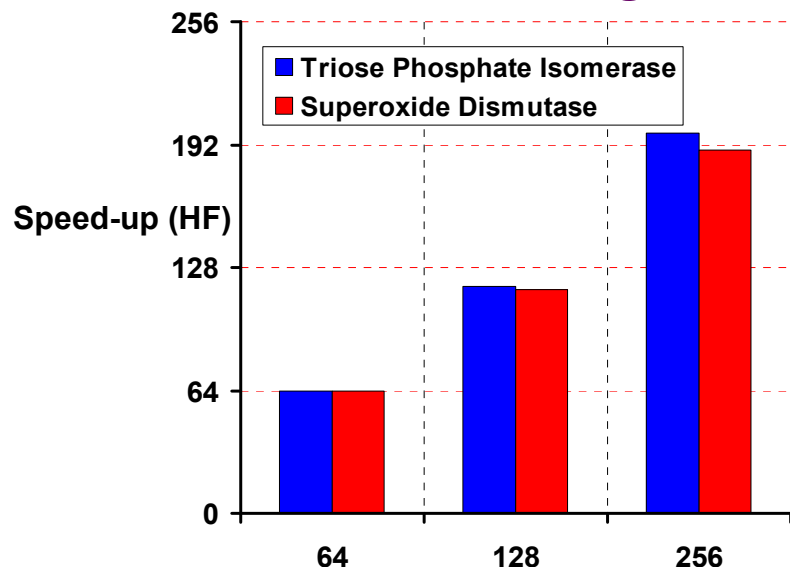
Performance

DZVP Basis (DZV\_A2) : 3975 GTOs

Hartree Fock ( IBM p690+ and p5-575)

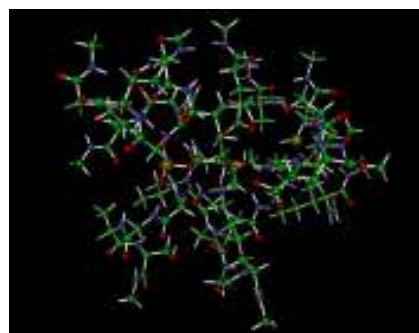
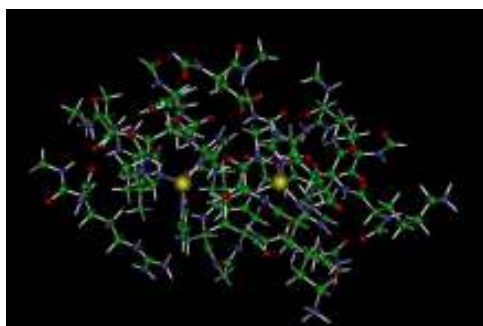


# Benchmark Systems and Target Application



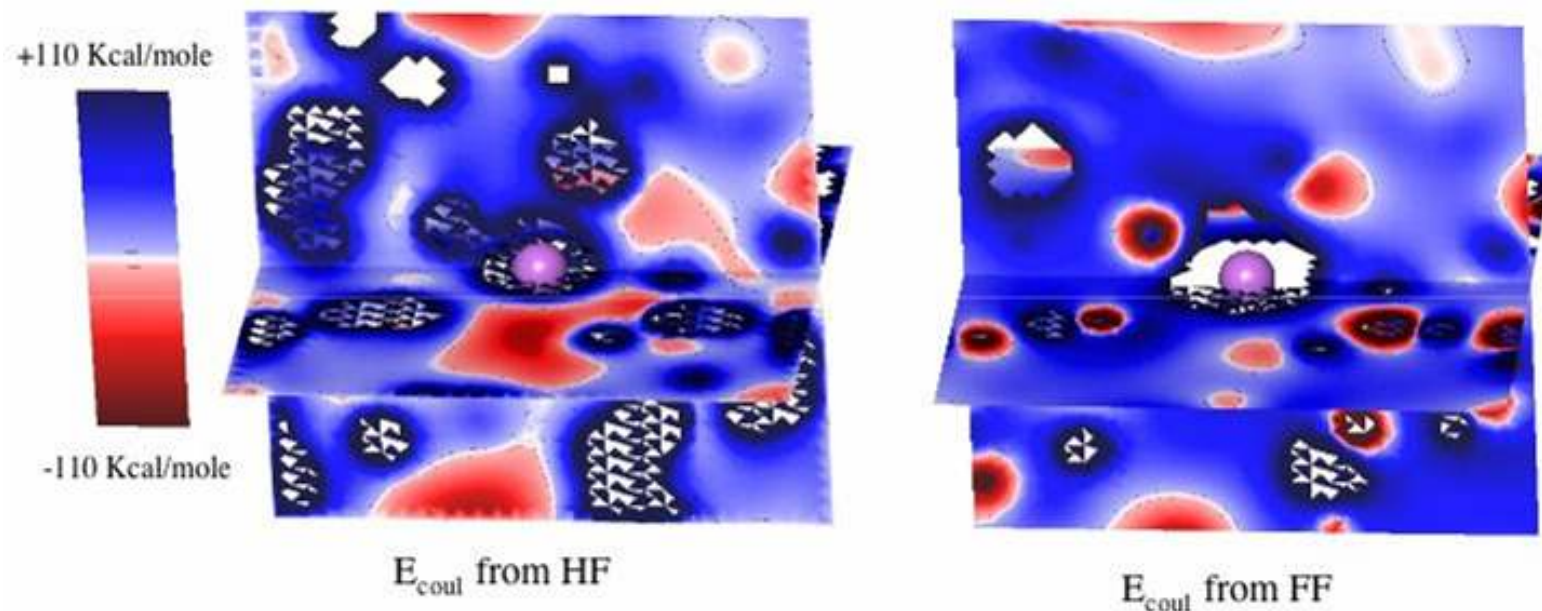
**Superoxide Dismutase (SOD)** model,  
Cu, Zn + residues within 7Å; 3870 basis  
functions at 6-31G\*

**Triose Phosphate Isomerase (TIM)**  
model, 3755 basis functions at 6-31G\*



**Isocitrate Lyase** (mixed basis set,  
22,400 basis functions) is the largest  
calculation performed to date.

# Isocitrate Lyase: Electrostatic Potentials



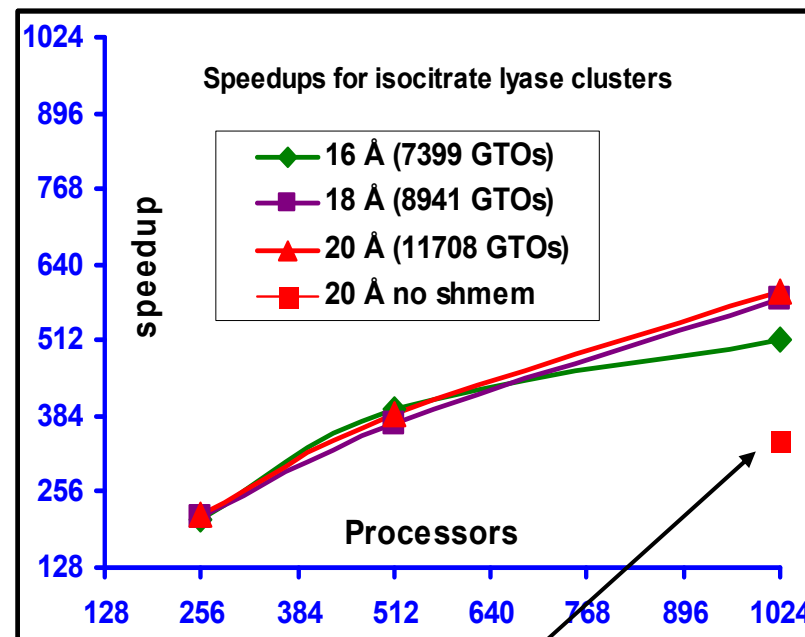
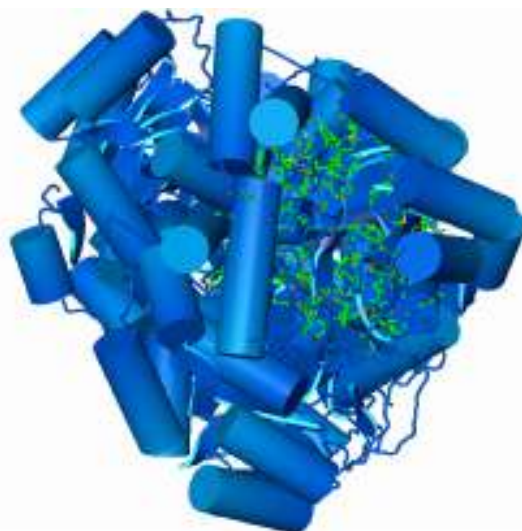
- Series of calculations forms part of a drug design study (van Lenthe, Utrecht, de Jonge & Vinkers, MolMo Services BVBA Belgium)
- Using *ab-initio* results to calibrate excluded volume and electrostatic terms in a docking scheme
- Electrostatic potential from the Hartree-Fock calculations (left) and that calculated from Force-Field charges (right), suggest that commonly used charge models may not be adequate.

# Exploiting Shared Memory

- Replicated memory Fock build is adopted to avoid large numbers of one-sided communications over the network
- Shared memory nodes offer a way to reduce memory requirement
  - ✘ Allocate one copy of the matrix *per shared memory node*
  - ✘ Exploit fast one-sided (shared memory) comms within each node
  - ✘ No additional one-sided communications between nodes
- This concept is available
  - ✘ Within Global Array-based programs using mirrored arrays
  - ✘ Within MPI programs by using standard UNIX System V Shared memory interface (library written by Ian Bush)

## GAMESS-UK: Scalability of Shared Memory code

- Sample scalability for 10 cycles of SCF for enzyme clusters on HPCx.



- F,P held in shared memory (using an optimal group size of 16 processes)
- All other data fully distributed
- Communications using MPI, distributed matrix algebra using ScaLAPACK

Shared memory use allows use of all processors on node; without it only 12 out of 32 procs can be used, reducing available performance.

## Aspects of Running the Parallel Code 1.

- Memory utilisation
  - ⌘ Default is 50 MWords per CPU on HPCx, AlphaServers and SGI Altix (c.f. 4 Mword in serial code)
    - MEMORY 50000000
- The weakness (and diversity) of I/O performance on parallel machines leads to various I/O options
  - ⌘ specified under control of GAMESS pre-directives
  - ⌘ Mapping a file into a global array
    - FILE <stream> <filename> GAFS LENGTH <length>
    - e.g. FILE ED3 MFGED3 GAFS LENGTH 70000 DELETE
  - ⌘ Mapping a file into memory:
    - FILE <stream> MEMORY LENGTH <length>
- Execution of the code may be influenced by the parallel tools used, e.g. parallel harness (TCGMSG), the Global Array (GA) Tools and the parallel diagonaliser (PeIGS).

## Aspects of Running the Parallel Code 2.

- The PARALLEL pre-directive
  - PARALLEL IOMODE
    - PARALLEL IOMODE NZO
    - PARALLEL IOMODE GAFS (Default on HPCx)
- Controlling the use of Parallel linear algebra
  - PARALLEL DIIS <size>
  - PARALLEL ORTHOG <size>
  - PARALLEL DIAG <size>
  - PARALLEL MULT2 <size>
- File Naming Conventions
  - Mainfiles - ed2, ed2001, ed2002 .... etc

PARALLEL DIIS NEVER  
PARALLEL DIIS ALWAYS

# Parallel Examples: (GA-based code)

## 1. DFT - Explicit J and J-fit on $\text{Si}_8\text{O}_{25}\text{H}_{18}$

```
rungames -p 8 -T 15 -q siosi.dft
```

```
TITLE
Si8O25H18 S-VWN DZVP2(H),DZVP(O,SI)
GEOMETRY AU NWCHEM
O 8.0 0.000000 0.000000 0.000000
SI 14.0 3.014576 0.000000 0.000000
.....
H 1.0 -7.752506 1.463380 5.771595
H 1.0 -10.346308 -2.979524 0.189468
END
BASIS
DFT H DZVP2
DFT O DZVP
DFT SI DZVP
END
SCFTYPE DIRECT
SELECT
DFT S-VWN
THRESH 4
ENTER
```

```
siosi.dft.in
```

```
TITLE
Si8O25H18 S-VWN DZVP2(H),DZVP(O,SI) JFIT
GEOMETRY AU NWCHEM
O 8.0 0.000000 0.000000 0.000000
SI 14.0 3.014576 0.000000 0.000000
.....
H 1.0 -7.752506 1.463380 5.771595
H 1.0 -10.346308 -2.979524 0.189468
END
BASIS
DFT H DZVP2
DFT O DZVP
DFT SI DZVP
END
SCFTYPE DIRECT
SELECT
DFT S-VWN
DFT JFIT MEMORY
DFT SCHWARTZ 6
DFT JBAS AHLRICH
THRESH 4
ENTER
```

```
siosi.dft.jfit.in
```

# Parallel Examples (GA-based code)

## 2. SCF Analytic 2nd Derivatives

```

TITLE
PYRIDINE 6-31G** SCF ANALYTIC 2ND
DERIVS  ZMAT ANGSTROM
N
X 1 1.0          rungamess -p 8 -T 15 -q scf2nds
X 1 1.0 2 90.
X 1 1.0 2 90. 3 90.
C 1 C4N 3 90. 2 180.
X 5 1.0 1 90. 3 0.0
X 5 1.0 1 90. 4 0.0
H 5 CH4 6 90. 1 180.
C 1 C2N 2 C2NZ 3 180.
C 1 C2N 2 C2NZ 3 0.0
C 9 C2C3 1 CCN 2 180.
C 10 C2C3 1 CCN 2 180.
H 9 C2H6 1 NCH2 2 0.0
H 10 C2H6 1 NCH2 2 0.0
H 11 C3H5 9 C2C3H 1 180.

```

```

VARIABLES
C4N  2.7729490
CH4  1.0759311
C2N  1.3209411
C2C3 1.3849059
C2H6 1.0767941
C3H5 1.0746205
C2NZ 121.1485486
CCN  123.6021799
NCH2 116.1951719
C2C3H 120.3317247
END
BASIS 6-31G**
RUNTYPE HESSIAN
SCFTYPE DIRECT scf2nds.in
ENTER

```

## Parallel Examples 3. MP2 Gradients (GA-code)

*Again, the direct-MP2 module cannot deal directly with non-abelian point groups, and the user must modify the z-matrix, in particular the atomic TAGs, to ensure that the resulting point group is a sub-group of  $D_{2h}$*

```
TITLE
SCF3 - TZVP MP2 - FORCE CONSTANTS
ZMAT ANGSTROM
SC
X 1 1.0
F 1 SCF 2 90.0
F1 1 SCF 2 90.0 3 120.0
F1 1 SCF 2 90.0 3 -120.0
VARIABLES
SCF 1.8697117 HESS .818724
END
BASIS TZVP
RUNTYPE FORCE
SCFTYPE DIRECT MP2
LEVEL 2
ENTER
```

```
TITLE
H2CO - TZVP+F+G BASIS - DIRECT-MP2
ZMATRIX ANGSTROM
C
O 1 CO
H 1 CH 2 HCO
H 1 CH 2 HCO 3 180.0
VARIABLES
CO 1.203 \ CH 1.099 \ HCO 121.8\END
BASIS
TZVP O
TZVP C
TZVP H
F C
1 1.0
F O
1.0 1.0
G C
1.0 1.0
G O
1.0 1.0
END
RUNTYPE OPTIMIZE
SCFTYPE DIRECT MP2
ENTER
```

mp2\_direct.in

# Parallel Examples: (MPI-based code)

## 1. DFT - Cyclosporin (6-31G)

```

file ed3 ed3-cyclo
parallel diag divide
title
cyclosporin (6-31g) DFT B3LYP
integ mpass
noprnt
geometry
.000000 .000000 .000000 6.0 c1
.000000 .000000 2.038032 1.0 h159
.....
-.407325 15.785158 -.310750 1.0 h132
-1.932656 12.811531 -.407583 1.0 h133
end
basis 6-31g
scftype direct

```

```

newscf
print full
Phase 1
Level 1.00
DIIS
# Switch to phase 2
next 2
Tester below 0.015
Phase 2
Level 0.00
DIIS
# Converge calculation
next 0
Tester below 1.0e-4
end
dft b3lyp
enter

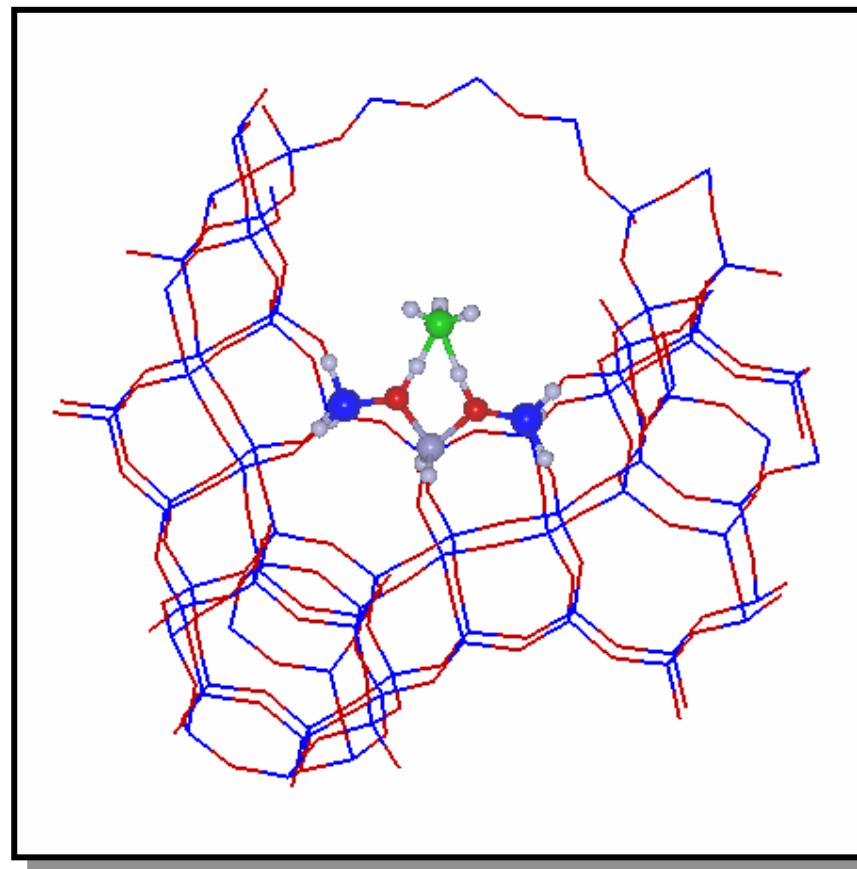
```

Fine-grain  
specification of  
convergence controls

# The QM/MM Modelling Approach

- Couple quantum mechanics and molecular mechanics approaches
- QM treatment of the active site
  - reacting centre
  - excited state processes (e.g. spectroscopy)
  - problem structures (e.g. complex transition metal centre)
- Classical MM treatment of environment
  - enzyme structure
  - zeolite framework
  - explicit solvent molecules
  - bulky organometallic ligands

<http://www.cfs.dl.ac.uk/qmmm>



[http://www.cfs.dl.ac.uk/tutorials/chemshell\\_workshop/chemshell\\_workshop.pdf](http://www.cfs.dl.ac.uk/tutorials/chemshell_workshop/chemshell_workshop.pdf)

# CHARMM + GAMESS-UK

- Developed at Daresbury in collaboration with Bernie Brooks, Eric Billings (NIH), Milan Hodoscek.
- Based on Additive Model, Electrostatic Embedding
  - ✘ Link atoms (Hydrogen)
  - ✘ Neutral charge groups can be used to choose QM region and avoid link-atom .. point charge close contacts.
- Other QM interfaces also available in CHARMM
  - ✘ Semi-empirical

M.J. Field, P.A. Bash, M. Karplus, J. Comput. Chem. 11 (1990) 700-33

- ✘ GAMESS(US)
- ✘ CADPAC

<http://www.cse.clrc.ac.uk/qcg/chmguk>

# TroubleShooting

A number of areas that might present problems for the new user, and remain as potential stumbling blocks, are outlined and developed below:

- Linearly Dependent Basis Sets
- SCF Convergence Problems
- Geometry Optimisation problems and Transition state Searches

## Linearly Dependent Basis Sets

- Smallest eigenvalue of the overlap matrix (diagnosed when generating the trial SCF MOs)

\*\*\* possible linear dependence diagnosed \*\*\*

```
the smallest eigenvalue of the overlap matrix is 0.00000005  
there is(are) 1 eigenvalue(s) less than 1.00E-05  
the number of canonical orbitals kept is 101
```

- Check for superimposed (or almost superimposed) centres.
- Even tempered basis sets? Is the scale factor of the exponents  $> 2.0$ ?
- If you are not already using HARMONIC with polarised basis sets, introduce the directive.
- Reconsider the need for diffuse exponents. Can you do without them?
- HF vs post-HF. The post-HF methods are much more sensitive to this type of problem.

# SCF Problems

The following problems may occur in SCF calculations

1. **SCF aborts at the outset of RESTART processing**; this will almost certainly have arisen from supplying a 2e-integral file that is not in the format expected. Care must be taken in understanding the workings of the SUPER directive and the use of BYPASS.
2. **SCF fails to converge, with violent oscillations in the total energy**; increase the LEVEL shift parameters, which should remove the problem.
3. **Large negative values for the energy**: the program assumes in constructing the Fock matrix that the density matrix transforms as the totally symmetric representation i.e. the wave function belongs to a non-degenerate representation of the molecular point group. If this is not the case, the energy may collapse to a meaningless value; modify the TAGs in the z-matrix to lower the effective molecular point group.
4. **SCF output reveals that the DIIS procedure has been initiated, but the energy still fluctuates, with DIIS appearing to push the energy up**; this may be indicative of trying to converge on an excited state, particularly in closed-shell calculations. If the TESTER is small, and the problem persists, it may help to suppress DIIS through the data line DIIS OFF.

## SCF Problems II.

5. The two-electron integral file is lost or corrupted, say in the middle of an optimisation; use the REGEN parameter on the RESTART directive.

6. ATTEMPTING TO RETRIEVE UNDEFINED DUMPFIL SECT message appears during vector generation in a restart job; this may have arisen from specifying an incorrect section on the VECTORS directive, but it is also indicative of the user having inadvertently omitted the RESTART directive, or incorrectly allocated the Dumpfile to the Job.

7. **Confusion over Multiple Section Specification:** When performing open shell SCF or GVB calculations, the final list of output vectors refers to the canonicalised set i.e. to the set of MOs residing in the **second** section (if specified) on the ENTER directive. Confusion may arise when restoring MOs from the first section, and assigning, for example, orbital symmetries from the SCF output.

8. **Symmetry contaminated molecular orbitals:** the SCF modules are fairly demanding in their attempts to maintain the symmetry classification of the MOs. If the orbitals appear contaminated, the cause may lie in the choice of an inappropriate starting set used to initialise the SCF.

# SCF Convergence Problems

- The SCF modules use a combination of level shifting and DIIS
- Convergence problems? Initial Steps ...
  - ✘ Check the geometry
  - ✘ It's usually the initial guess - is it sensible?
    - Vectors atoms print
  - ✘ Increase the number of iterations
    - MAXCYC 100
  - ✘ If notable oscillations in energy, increase the level shift(s)
    - LEVEL 2.0 (closed shells); LEVEL 0.3 2.0 (for GVB/open shells)
  - ✘ If calculation fails because of too few iterations, restart using MOs from the previous calculation
  - ✘ Converging to wrong state (DOMO-LUMO gap?). Use SWAP.
  - ✘ Use LOCK to fix the electronic configuration (default for open-shell).

# Common Geometry Optimisation Problems 1.

1. ZMATRIX and Point Group Problems: One of the constraints in the z-matrix formalism concerns potential problems in dealing with directly bonded angles of 180. The use of dummy X-centres in circumventing these problems is in Part 3, section 8.3. Consider below the data file for optimisation of the  $\text{CaH}_2$  species, where we are starting the optimisation with a bond angle of 150.

```
ZMAT ANGS  
CA  
H 1 CAH  
H 1 CAH 2 HCAH  
VARIABLES  
CAH 2.148 HESS 0.7  
HCAH 150.0 HESS 0.1  
END  
RUNTYPE OPTIMIZE  
ENTER
```

**This job will fail...**

geom.opt.problem1a.in

# Common Geometry Optimisation Problems 1.

... with the diagnostics shown below taken from the failing output:

```
=====
VARIABLE      VALUE      HESSIAN
=====
CAH           2.1481952 ANGS    0.699998
HCAH          181.0076613 DEGS    0.034888
ERROR ON Z-MATRIX CARD NUMBER 3
```

ANGLE ALPHA IS OUTSIDE THE VALID RANGE OF 0 TO 180 ERROR  
DETECTED: ERROR DETECTED IN CONVERTING Z-MATRIX TO  
CARTESIAN COORDINATES

In most cases this can be avoided by using the half-angle  
(XCAH) and dummy centre specification, as shown:

```
TITLE
CAH2 .. 3-21G
ZMAT ANGS
CA
X 1 1.0
H 1 CAH 2 XCAH
H 1 CAH 2 XCAH 3 180.0
VARIABLES
CAH 2.148 HESSIAN 0.7
XCAH 75.0 HESSIAN 0.1
END
RUNTYPE OPTIMIZE
ENTER
```

geom.opt.problem1b.in

## Common Geometry Optimisation Problems 2.

2. An additional constraint in the default optimisation algorithm is the inability to handle a change in point group during the optimisation pathway. This example has been chosen to illustrate this point: while the above data avoids the angle problem, it again fails because of the point group change ( $C_{2v}$  to  $D_{2h}$ ):

```
=====
                          Z-MATRIX (ANGSTROMS AND DEGREES)
CD CENT ATOM N1  LENGTH  N2  ALPHA  N3  BETA
-----
1  1  CA
2   X  1  1.000000 ( 1)
3  2  H  1  2.148105 ( 2) 2  89.998 ( 4)
4  3  H  1  2.148105 ( 3) 2  89.998 ( 5) 3  180.000 ( 6)
=====
```

```
=====
VARIABLE          VALUE          HESSIAN
=====
CAH              2.1481048 ANGS      0.699995
XCAH             89.9978901 DEGS     0.156813
```

## Common Geometry Optimisation Problems 2.

```
**** CHANGE IN POINT GROUP ****
      CNV      2  DNH      2
#####
      ERROR DETECTED
      POINT GROUP CHANGE DURING OPTIMISATION
#####
```

In such cases the optimisation may be completed by changing to the OPTXYZ algorithm, using the RESTART option to retrieve the partially converged geometry from the Dumpfile as the starting point. Note that OPTXYZ will not suffer from either ZMATRIX or Point group problems.

RESTART

TITLE\CAH2 .. 3-21G

ZMAT ANGS\CA\X 1 1.0\ H 1 CAH 2 XCAH\H 1 CAH 2 XCAH 3 180.0

VARIABLES\CAH 2.148 HESSIAN 0.7\XCAH 75.0 HESSIAN 0.1\END

RUNTYPE OPTXY

ENTER

geom.opt.problem1c.in

## Common Geometry Optimisation Problems 3.

3. Particular care should be taken when dealing with the ZMATRIX specification for **cyclic systems**. If this is specified such that a multiple bond is not explicitly defined, then the initial step taken in an optimisation with either a poor starting geometry or ill-defined Hessian can lead to unreasonable values for such a bond.

In the worst case, the bond can become so compressed (with a disastrous effect on the total energy) that linear dependence will be (correctly) diagnosed! Certainly in such cases the starting hessian should be taken from a smaller basis calculation (STO-3G will do).

4. **The optimisation appears to have converged but iterations continue**; if too small a value of XTOL is specified, the energy may not be sufficiently numerically accurate to justify using line searches. This is especially true when employing default accuracy integrals.

5. If a large step is made it is possible for the wavefunction to change state. Usually this is manifested by the **linear search returning to a previous point** on the surface, or by exceptionally large values in the hessian matrix. To avoid this reduce STEPMAX, and for a closed shell system use LOCK.

# Hands-on session

- ✎ At the login window, type the following:
  - Username: **msscXX**
  - Password: **Chem1st**
- ✎ Open a terminal:
  - Click on the **RedHat** icon on the menu bar at the bottom of the screen.
  - Select **:System Tools => Terminal**
- ✎ Software is installed in:
  - **/opt/chemistry**
- ✎ To use **rungames**:
  - **rungames** [options] *jobname*
- ✎ To run the **CCP1GUI**
  - **ccp1gui.sh**