

(5 December 2014)

```

*****
*                                     *
* Section 4 - Further Information *
*                                     *
*****

```

This section of the manual contains literature references and hints on how to make skillful use of GAMESS.

The following topics are covered:

<i>Computational References</i>	6
<i>Basis Set References</i>	18
<i>Spherical Harmonics</i>	30
<i>How to do RHF, ROHF, UHF, and GVB calculations</i>	31
general considerations	31
direct SCF	32
convergence accelerators	35
high spin open shell SCF (ROHF)	38
other open shell SCF cases (GVB)	41
true GVB perfect pairing runs	48
the special case of TCSCF	49
a caution about symmetry	50
<i>How to do MCSCF (and CI) calculations</i>	52
MCSCF implementation	53
orbital updates	55
CI coefficient optimization	57
determinant CI	60
CSF CI	63
starting orbitals	67
miscellaneous hints	69
MCSCF references	71
<i>Second Order Perturbation Theory</i>	74
RHF and UHF reference MP2	74
high spin ROHF reference MP2	75
GVB based MP2	77
MCSCF reference perturbation theory	78

<i>Coupled-Cluster Theory</i>	83
available computations (ground states)	86
available computations (excited states)	94
density matrices and properties	100
excited state example	104
resource requirements	105
restarts in ground-state calculations.....	110
initial guesses in excited-state calculations.....	112
eigensolvers for excited-state calculations.....	113
references and citations required in publications	115
<i>Density Functional Theory</i>	122
DFTTYP keywords	122
grid-free DFT	123
DFT with grids	124
Time Dependent Density Functional Theory (TD-DFT)	126
references for DFT	129
<i>Summary of excited state methods</i>	140
<i>Geometry Searches and Internal Coordinates</i>	142
quasi-Newton Searches	142
the nuclear Hessian	145
coordinate choices	146
the role of symmetry	151
practical matters.....	152
saddle points.....	154
mode following	156
<i>Intrinsic Reaction Coordinate Methods</i>	158
<i>Gradient Extremals</i>	163
<i>Continuum Solvation Methods</i>	169
Self Consistent Reaction Field (SCRF).....	169
Polarizable Continuum Model (PCM)	170
SVPE and SS(V)PE.	179
Conductor-like screening model (COSMO).....	184
<i>The Effective Fragment Potential Method</i>	186
terms in an EFP	187
constructing an EFP1	188
constructing an EFP2	189
current limitations	190

Further Information	4-3
practical hints for using EFPs	191
global optimization	193
QM/MM across covalent bonds	194
Simpler potentials	197
references	198
<i>The Fragment Molecular Orbital method</i>	203
Surfaces and solids	205
FMO variants	206
Effective fragment molecular orbital method (EFMO)	207
Guidelines for approximations with FMO3	207
How to perform FMO-MCSCF calculations	208
How to perform multilayer runs	209
How to mix basis sets in FMO	210
How to perform FMO/PCM calculations	210
How to perform FMO/EFP calculations	211
Geometry optimization or saddle point search for FMO	211
FMO hessian calculations	212
Molecular dynamics with FMO	212
Pair interaction energy decomposition analysis (PIEDA)	212
Excited states	213
Selective FMO	214
Frozen domain	214
IMOMM with FMO	214
Analyzing and visualizing the results	215
Parallelization of FMO runs with GDDI	216
Limitations of the FMO method in GAMESS	216
Restarts with the FMO method	217
Note on accuracy	217
FMO References	218
<i>The Cluster-in-Molecules method</i>	224
sequential and parallel execution	224
restarts	224
the cimshell script	225
CIM references	226
<i>MOPAC Calculations within GAMESS</i>	227
<i>Molecular Properties and Conversion Factors</i>	230
Polarizabilities	231

Further Information	4-4
<i>Localized Molecular Orbitals</i>	233
<i>Transition Moments and Spin-Orbit Coupling</i>	239
states	239
orbitals	240
symmetry	241
spin orbit coupling	243
input nitty-gritty	246
references	247
examples	249

For people who are newcomers to computational chemistry, it may be helpful to study an introductory book.

First, some texts about quantum chemistry:

"Ab Initio Molecular Orbital Theory"

W.J.Hehre, L.Radom, J.A.Pople, P.v.R.Schleyer
Wiley and Sons, New York, 1986

"Modern Quantum Chemistry" (now a Dover paperback)

A.Szabo, N.S.Ostlund McGraw-Hill, 1989

"Quantum Chemistry, 6th Edition"

I.N.Levine Prentice Hall, 2008

Then, a few books more focused on computation:

"Introduction to Quantum Mechanics in Chemistry"

M.A.Ratner, G.C.Schatz Prentice Hall, 2000

"Introduction to Computational Chemistry, 2nd Edition"

Frank Jensen Wiley and Sons, Chichester, 2006

"Molecular Modeling Basics"

Jan H. Jensen CRC Press, Boca Raton, 2010

Frank's book is an outstanding survey of methods, basis sets, properties, and other topics.

Jan's book is a good complement to Frank's, staying at a simpler level, using GAMESS input examples. It has an accompanying online blog,

<http://molecularmodelingbasics.blogspot.com>

Computational References

GAMESS -

M.W.Schmidt, K.K.Baldrige, J.A.Boatz, S.T.Elbert,
M.S.Gordon, J.H.Jensen, S.Koseki, N.Matsunaga,
K.A.Nguyen, S.Su, T.L.Windus, M.Dupuis, J.A.Montgomery
J.Comput.Chem. 14, 1347-1363 (1993)

M.S.Gordon, M.W.Schmidt pp 1167-1189
in "Theory and Applications of Computational Chemistry,
the first forty years" C.E.Dykstra, G.Frenking, K.S.Kim,
G.E.Scuseria (editors), Elsevier, Amsterdam, 2005.

HONDO -

These papers describes many of the algorithms in detail,
and much of these applies also to GAMESS:

"The General Atomic and Molecular Electronic Structure
System: HONDO 7.0" M.Dupuis, J.D.Watts, H.O.Villar,
G.J.B.Hurst Comput.Phys.Comm. 52, 415-425(1989)

"HONDO: A General Atomic and Molecular Electronic
Structure System" M.Dupuis, P.Mougenot, J.D.Watts,
G.J.B.Hurst, H.O.Villar in "MOTEC: Modern Techniques
in Computational Chemistry" E.Clementi, Ed.
ESCOM, Leiden, the Netherlands, 1989, pp 307-361.

"HONDO: A General Atomic and Molecular Electronic
Structure System" M.Dupuis, A.Farazdel, S.P.Karna,
S.A.Maluendes in "MOTEC: Modern Techniques in
Computational Chemistry" E.Clementi, Ed.
ESCOM, Leiden, the Netherlands, 1990, pp 277-342.

M.Dupuis, S.Chin, A.Marquez in "Relativistic and Electron
Correlation Effects in Molecules", G.Malli, Ed. Plenum
Press, NY 1994, pp 315-338.

sp integrals and gradient integrals -

inner axis sp integration is done by McMurchie/Davidson
J.A.Pople, W.J.Hehre J.Comput.Phys. 27, 161-168(1978)
H.B.Schlegel, J.Chem.Phys. 77, 3676-3681(1982)

spd integrals by rotated axis/McMurchie-Davidson

K.Ishimura, S.Nagase Theoret.Chem.Acc. 120, 185-189(2008)

McMurchie/Davidson integrals -

L.E.McMurchie, E.R.Davidson
J.Comput.Phys. 26, 218-231(1978)

spdfghi integrals -

"Numerical Integration Using Rys Polynomials"

- H.F.King and M.Dupuis J.Comput.Phys. 21,144(1976)
"Evaluation of Molecular Integrals over Gaussian
Basis Functions"
M.Dupuis,J.Rys,H.F.King J.Chem.Phys. 65,111-116(1976)
"Molecular Symmetry and Closed Shell HF Calculations"
M.Dupuis and H.F.King Int.J.Quantum Chem. 11,613(1977)
"Computation of Electron Repulsion Integrals using
the Rys Quadrature Method"
J.Rys,M.Dupuis,H.F.King J.Comput.Chem. 4,154-157(1983)

ERIC spdfg integrals -
"Recursion Formula for Electron Repulsion Integrals Over
Hermite Polynomials"
G.D.Fletcher Int.J.Quantum Chem. 106, 355-360(2006)

spdfg gradient integrals -
"Molecular Symmetry. II. Gradient of Electronic Energy
with respect to Nuclear Coordinates"
M.Dupuis and H.F.King J.Chem.Phys. 68,3998(1978)
although the implementation is much newer than this paper.

spd hessian integrals -
"Molecular Symmetry. III. Second derivatives of Electronic
Energy with respect to Nuclear Coordinates"
T.Takada, M.Dupuis, H.F.King
J.Chem.Phys. 75, 332-336 (1981)

the Q matrix, and integral transformation symmetry -
E.Hollauer, M.Dupuis J.Chem.Phys. 96, 5220 (1992)

spdfg effective core potential (ECP) integral/derivatives -
C.F.Melius, W.A.Goddard Phys.Rev.A 10,1528-1540(1974)
L.R.Kahn, P.Baybutt, D.G.Truhlar
J.Chem.Phys. 65, 3826-3853 (1976)
M.Krauss, W.J.Stevens Ann.Rev.Phys.Chem. 35, 357-385(1985)
J.Breidung, W.Thiel, A.Komornicki
Chem.Phys.Lett. 153, 76-81(1988)
B.M.Bode, M.S.Gordon J.Chem.Phys. 111, 8778-8784(1999)
See also the papers listed for SBKJC and HW basis sets.

model core potential (MCP) reviews -
S.Huzinaga Can.J.Chem. 73, 619-628(1995)
M.Klobukowski, S.Huzinaga, Y.Sakai, in Computational
Chemistry: Reviews of current trends, volume 3, pp 49-74,
edited by J.Leszczynski, World Scientific, Singapore, 1999.

Quantum fast multipole method (QFMM) -
E.O.Steinborn, K.Ruedenberg
Adv.Quantum Chem. 7, 1-81(1973)

L.Greengard "The Rapid Evaluation of Potential Fields in Particle Systems" (MIT, Cambridge, 1987)

C.H.Choi, J.Ivanic, M.S.Gordon, K.Ruedenberg

J.Chem.Phys. 111, 8825-8831(1999)

C.H.Choi, K.Ruedenberg, M.S.Gordon

J.Comput.Chem. 22, 1484-1501(2001)

C.H.Choi J.Chem.Phys. 120, 3535-3543(2004)

RHF -

C.C.J.Roothaan Rev.Mod.Phys. 23, 69-89(1951)

UHF -

J.A.Pople, R.K.Nesbet J.Chem.Phys 22, 571-572(1954)

high-spin coupled ROHF -

C.C.J.Roothaan Rev.Mod.Phys. 32, 179-185(1960)

R.McWeeny, G.Diercksen J.Chem.Phys. 49,4852-4856(1968)

M.F.Guest, V.R.Saunders Mol.Phys. 28, 819-828(1974)

J.S.Binkley, J.A.Pople, P.A.Dobosh

Mol.Phys. 28, 1423-1429(1974)

E.R.Davidson Chem.Phys.Lett. 21,565-567(1973)

K.Faegri, R.Manne Mol.Phys. 31,1037-1049(1976)

H.Hsu, E.R.Davidson, and R.M.Pitzer

J.Chem.Phys. 65,609-613(1976)

B.N.Plakhutin, E.V.Gorelik, N.N.Breslavskaya

J.Chem.Phys. 125, 204110/1-10(2006)

B.N.Plakhutin, E.R.Davidson

J.Phys.Chem.A 113, 12386-12395(2009)

E.R.Davidson, B.N.Plakhutin

J.Chem.Phys. 132, 184110/1-14(2010)

K.R.Glaesemann, M.W.Schmidt

J.Phys.Chem.A 114, 8772-8777(2010)

Constrained UHF (CUHF is equivalent to high spin ROHF) -

G.E.Scuseria, T.Tsuchimochi

J.Chem.Phys. 134, 064101/1-14(2011)

GVB and low-spin coupled ROHF -

F.W.Bobrowicz and W.A.Goddard, in Modern Theoretical Chemistry, Vol 3, H.F.Schaefer III, Ed., Chapter 4.

DFT and TD-DFT -

All appropriate references are included in the section on density functional theory included below.

MCSCF - see reference list in its own subsection below

determinant CI -

full CI (ALDET) and general CI (GENCI),

J.Ivanic, K.Ruedenberg

Theoret.Chem.Acc. 106, 339-351(2001)

occupation restricted multiple active space (ORMAS),

J.Ivanic J.Chem.Phys. 119, 9364-9376, 9377-9385(2003)

configuration state function CI (GUGA) -

B.Brooks and H.F.Schaefer J.Chem. Phys. 70,5092(1979)

B.Brooks, W.Laidig, P.Saxe, N.Handy, and H.F.Schaefer,
Physica Scripta 21, 312(1980).

CIS energy and gradient -

J.B.Foresman, M.Head-Gordon, J.A.Pople, M.J.Frisch

J.Phys.Chem. 96, 135-149(1992)

R.M.Shroll, W.D.Edwards

Int.J.Quantum Chem. 63, 1037-1049(1997)

the parallel CIS implementation in GAMESS is described in

S.P.Webb Theoret.Chem.Acc. 116, 355-372(2006)

which has a nice review of other excited state methods.

spin-flip CIS:

A.I.Krylov Chem.Phys.Lett. 338, 375(2001)

closed, unrestricted open shell 2nd order Moller-Plesset -

J.A.Pople, J.S.Binkley, R.Seeger

Int. J. Quantum Chem. S10, 1-19(1976)

M.J.Frisch, M.Head-Gordon, J.A.Pople,

Chem.Phys.Lett. 166, 275-280(1990)

C.M.Aikens, S.P.Webb, R.L.Bell, G.D.Fletcher, M.W.Schmidt,

M.S.Gordon Theoret.Chem.Acc., 110, 233-253(2003)

with the TCA "overview article" being a thorough review of
the single determinant MP2 gradient equations.

CODE=SERIAL is generally based on the CPL paper above, as
described in the HONDO references given above.

The next two document CODE=DDI for RHF and UHF,

G.D.Fletcher, M.W.Schmidt, M.S.Gordon

Adv.Chem.Phys. 110, 267-294(1999)

C.M.Aikens, M.S.Gordon

J.Phys.Chem.A, 108, 3103-3110(2004)

The next two document CODE=IMS for RHF,

K.Ishimura, P.Pulay, S.Nagase

J.Comput.Chem. 27, 407-413(2006)

K.Ishimura, P.Pulay, S.Nagase

J.Comput.Chem. 28, 2034-2042(2007)

The next documents code=RIMP2 for RHF and UHF,

M.Katouda, S.Nagase

Int.J.Quantum Chem. 109, 2121-2130(2009)

Spin Component Scaled MP2 (SCS-MP2)

S.Grimme

J.Chem.Phys. 118, 9095-9102(2003)

spin restricted open shell MP2, ZAPT energy -

T.J.Lee, D.Jayatilaka

Chem.Phys.Lett. 201, 1-10(1993)

T.J.Lee, A.P.Rendell, K.G.Dyall, D.Jayatilaka

J.Chem.Phys. 100, 7400-7409(1994)

nuclear gradients for ZAPT -

The next two document the CODE=DDI program,

G.D.Fletcher, M.S.Gordon, R.L.Bell

Theoret.Chem.Acc. 107, 57-70(2002)

C.M.Aikens, G.D.Fletcher, M.W.Schmidt, M.S.Gordon

J.Chem.Phys. 124, 014107/1-14(2006)

spin restricted open shell MP2, RMP method -

P.J.Knowles, J.S.Andrews, R.D.Amos, N.C.Handy, J.A.Pople

Chem.Phys.Lett. 186, 130-136 (1991)

W.J.Lauderdale, J.F.Stanton, J.Gauss, J.D.Watts, R.J.Bartlett

Chem.Phys.Lett. 187, 21-28(1991)

CUMP2 is equivalent to RMP2 (See CUHF reference above).

multiconfigurational quasidegenerate perturbation theory -

H.Nakano J.Chem.Phys. 99, 7983-7992(1993)

ORMAS-based multireference perturbation theory -

L.Roskop, M.S.Gordon J.Chem.Phys. 135, 044101/1-11(2012)

Coupled-Cluster -

Equation of Motion Coupled-Cluster (EOMCC) -

this is a subset of the relevant papers:

P.Piecuch, S.A.Kucharski, K.Kowalski, M.Musiał,

Comput.Phys.Comm. 149, 71-96(2002)

K.Kowalski, P.Piecuch,

J.Chem.Phys. 120, 1715-1738 (2004)

P.Piecuch, S.A.Kucharski, K.Kowalski, M.Musiał

Comput.Phys.Comm. 149, 71-96(2002).

parallel CCSD(T) program -

J.L.Bentz, R.M.Olson, M.S.Gordon, M.W.Schmidt, R.A.Kendall

Comput.Phys.Comm. 176, 589-600(2007)

R.M.Olson, J.L.Bentz, R.A.Kendall, M.W.Schmidt, M.S.Gordon

J.Comput.Theoret.Chem. 3, 1312-1328(2007)

Any publication describing the results of ground-state and/or excited-state calculations using the equation of motion coupled-cluster and/or completely renormalized EOMCCSD(T) options (CCTYP=EOM-CCSD or CR-EOM) obtained with GAMESS should reference the specific papers appearing in the printout. For more references to the primary literature for both types of coupled-cluster methods, see the section "Coupled-Cluster theory" below.

RHF/ROHF/TCSCF coupled perturbed Hartree Fock -
"Single Configuration SCF Second Derivatives on a Cray"
H.F.King, A.Komornicki in "Geometrical Derivatives of Energy Surfaces and Molecular Properties" P.Jorgensen J.Simons, Ed. D.Reidel, Dordrecht, 1986, pp 207-214.
"A parallel Distributed data CPHF algorithm for analytic Hessians" Y.Alexeev, M.W.Schmidt, T.L.Windus, M.S.Gordon J.Comput.Chem. 28, 1685-1694(2007).
Y.Osamura, Y.Yamaguchi, D.J.Fox, M.A.Vincent, H.F.Schaefer J.Mol.Struct. 103, 183-186(1983)
M.Duran, Y.Yamaguchi, H.F.Schaefer J.Phys.Chem. 92, 3070-3075(1988)
"A New Dimension to Quantum Chemistry" Y.Yamaguchi, Y.Osamura, J.D.Goddard, H.F.Schaefer Oxford Press, NY 1994

MCSCF coupled perturbed Hartree-Fock -
M.R.Hoffman, D.J.Fox, J.F.Gaw, Y.Osamura, Y.Yamauchi, R.S.Grev, G.Fitzgerald, H.F.Schaefer, P.J.Knowles, N.C.Handy J.Chem.Phys. 80, 2660-2668(1984)
the book by Osamura, Goddard, and Schaefer just mentioned.
T.J.Dudley, R.M.Olson, M.W.Schmidt, M.S.Gordon J.Comput.Chem. 27, 353-362(2006)

non-adiabatic coupling matrix element (NACME) -
J.C.Tully, chapter 5 (pp 217-267) in "Dynamics of Molecular Collisions - Part B", edited by W.H.Miller, Plenum Press, NY, 1976.
B.H.Lengsfeld, D.R.Yarkony, chapter 1 (pp. 1-71) in "State-selected and state-to-state in-molecule reaction dynamics- Part 2, theory", edited by M.Baer and C.-Y.Ng, John Wiley, NY, 1992.

harmonic vibrational analysis in Cartesian coordinates -
W.D.Gwinn J.Chem.Phys. 55,477-481(1971)

Normal coordinate decomposition analysis -
J.A.Boatz and M.S.Gordon,
J.Phys.Chem. 93, 1819-1826(1989).

Partial Hessian vibrational analysis -

H.Li, J.H.Jensen, Theoret.Chem.Acc. 107, 211-219(2002)

anharmonic vibrational spectra (VSCF) -

a review of VSCF:

R.B.Gerber, J.O.Jung

in "Computational Molecular Spectroscopy"

P.Jensen, P.R.Bunker, eds.

Wiley and Sons, Chichester, 2000, pp 365-390.

the basic method for VSCF and cc-VSCF:

G.M.Chaban, J.O.Jung, R.B.Gerber

J.Chem.Phys. 111, 1823-1829(1999)

the QFF approximation:

K.Yagi, K.Hirao, T.Taketsugu, M.W.Schmidt, M.S.Gordon

J.Chem.Phys. 121, 1383-1389(2004)

the VDPT solver:

N.Matsunaga, G.M.Chaban, R.B.Gerber

J.Chem.Phys. 117, 3541-3547(2002)

solver for larger systems:

L.Pele, B.Brauer, R.B.Gerber

Theoret.Chem.Acc. 117, 69-72(2007)

use of internal coordinates, and thermochemistry

B.Njegic, M.S.Gordon

J.Chem.Phys. 125, 224102/1-12(2006)

applications of RUNTYP=VSCF:

G.M.Chaban, J.O.Jung, R.B.Gerber

J.Phys.Chem.A 104, 2772-2779(2000)

J.Lundell, G.M.Chaban, R.B.Gerber

Chem.Phys.Lett. 331, 308-316(2000)

K.Yagi, T.Taketsugu, K.Hirao, M.S.Gordon

J.Chem.Phys. 113, 1005-1017(2000)

G.M.Chaban, R.B.Gerber, K.C.Janda

J.Phys.Chem.A 105, 8323-8332(2001)

A.T.Kowal Spectrochimica Acta A 58, 1055-1067(2002)

G.M.Chaban, S.S.Xantheas, R.B.Gerber

J.Phys.Chem.A 107, 4952-4956(2003)

G.M.Chaban J.Phys.Chem.A 108, 4551-4556(2004)

Y.Miller, G.M.Chaban, R.B.Gerber

J.Phys.Chem.A 109, 6565-6574(2005)

Y.Miller, G.M.Chaban, R.B.Gerber

Chem.Phys. 313, 213-224(2005)

C.A.Brindle, G.M.Chaban, R.B.Gerber, K.C.Janda

Phys.Chem.Chem.Phys. 7, 945-954(2005)

G.M.Chaban, R.M.Gerber

Theoret.Chem.Acc. 120, 273-279(2008)

Raman spectrum -

A.Komornicki, J.W.McIver J.Chem.Phys. 70, 2014-2016(1979)

G.B.Bacskay, S.Saebo, P.R.Taylor
Chem.Phys. 90, 215-224(1984)

static polarizabilities:

H.A.Kurtz, J.J.P.Stewart, K.M.Dieter
J.Comput.Chem. 11, 82-87 (1990)

dynamic polarizabilities:

P.Korambath, H.A.Kurtz, in "Nonlinear Optical Materials",
ACS Symposium Series 628, S.P.Karna and A.T.Yeates, Eds.
pp 133-144, Washington DC, 1996.

nuclear derivatives of dynamic polarizabilities,
and dynamic Raman and hyper-Raman:

O.Quinet, B.Champagne J.Chem.Phys. 115, 6293-6299(2001)

O.Quinet, B.Champagne B.Kirtman
J.Comput.Chem. 22, 1920-1932(2001)

O.Quinet, B.Champagne J.Chem.Phys. 117, 2481-2488(2002)

O.Quinet, B.Kirtman, B.Champagne
J.Chem.Phys. 118, 505-513(2003)

Geometry optimization and saddle point location -

J.Baker J.Comput.Chem. 7, 385-395(1986).

T.Helgaker Chem.Phys.Lett. 182, 503-510(1991).

P.Culot, G.Dive, V.H.Nguyen, J.M.Ghuysen
Theoret.Chim.Acta 82, 189-205(1992).

Dynamic Reaction Coordinate (DRC) -

J.J.P.Stewart, L.P.Davis, L.W.Burggraf,
J.Comput.Chem. 8, 1117-1123 (1987)

S.A.Maluendes, M.Dupuis, J.Chem.Phys. 93, 5902-5911(1990)

T.Taketsugu, M.S.Gordon, J.Phys.Chem. 99, 8462-8471(1995)

T.Taketsugu, M.S.Gordon, J.Phys.Chem. 99, 14597-604(1995)

T.Taketsugu, M.S.Gordon, J.Chem.Phys. 103, 10042-9(1995)

M.S.Gordon, G.Chaban, T.Taketsugu
J.Phys.Chem. 100, 11512-11525(1996)

T.Takata, T.Taketsugu, K.Hirao, M.S.Gordon
J.Chem.Phys. 109, 4281-4289(1998)

T.Taketsugu, T.Yanai, K.Hirao, M.S.Gordon
THEOCHEM 451, 163-177(1998)

Energy orbital localization -

C.Edmiston, K.Ruedenberg Rev.Mod.Phys. 35, 457-465(1963).

R.C.Raffenetti, K.Ruedenberg, C.L.Janssen, H.F.Schaefer,
Theoret.Chim.Acta 86, 149-165(1993)

Boys orbital localization -

S.F.Boys, "Quantum Science of Atoms, Molecules, and Solids"

P.O.Lowdin, Ed, Academic Press, NY, 1966, pp 253-262.

See the first paper on oriented localized orbitals if you wish to know the true origin of "Boys localization"

Population orbital localization -

J.Pipek, P.Z.Mezey J.Chem.Phys. 90, 4916(1989).

Oriented localized orbitals -

J.Ivanic, G.M.Atchity, K.Ruedenberg

Theoret.Chem.Acc. 120, 281-294(2008)

J.Ivanic, K.Ruedenberg

Theoret.Chem.Acc. 120, 295-305(2008)

Valence Virtual Orbitals (VVOS) -

W.C.Lu, C.Z.Wang, M.W.Schmidt, L.Bytautas, K.M.Ho,
K.Ruedenberg

J.Chem.Phys. 120, 2629-2637 and 2638-2651(2004)

W.C.Lu, C.Z.Wang, T.L.Chan, K.Ruedenberg, K.M.Ho
Phys.Rev.B 70, 041101-1/4(2004)

Mulliken Population Analysis -

R.S.Mulliken J.Chem.Phys. 23, 1833-1840, 1841-1846,
2338-2342, 2343-2346(1955)

so called "Lowdin Population Analysis" -

This should be described as "a Mulliken population analysis (ref M1-M4 above) based on symmetrically orthogonalized orbitals (ref L)", where reference L is

P.-O.Lowdin Adv.Chem.Phys. 5, 185-199(1970)

Lowdin populations are not invariant to rotation if the basis set used is Cartesian d,f,...:

I.Mayer, Chem.Phys.Lett. 393, 209-212(2004).

Bond orders and valences -

M.Giambiagi, M.Giambiagi, D.R.Grempe, C.D.Heymann

J.Chim.Phys. 72, 15-22(1975)

I.Mayer, Chem.Phys.Lett. 97,270-274(1983), 117,396(1985).

M.S.Giambiagi, M.Giambiagi, F.E.Jorge

Z.Naturforsch. 39a, 1259-73(1984)

I.Mayer, Theoret.Chim.Acta 67, 315-322(1985).

I.Mayer, Int.J.Quantum Chem. 29, 73-84(1986).

I.Mayer, Int.J.Quantum Chem. 29, 477-483(1986).

The same formula (apart from a factor of two) may also be seen in equation 31 of the second of these papers (the bond order formula in the 1st of these is not the same formula):

T.Okada, T.Fueno Bull.Chem.Soc.Japan 48, 2025-2032(1975)

T.Okada, T.Fueno Bull.Chem.Soc.Japan 49, 1524-1530(1976)

a review about bond orders:

I. Mayer, J.Comput.Chem. 28, 204-221(2007).

Direct SCF -

J.Almlof, K.Faegri, K.Korsell
J.Comput.Chem. 3, 385-399 (1982)
M.Haser, R.Ahlrichs
J.Comput.Chem. 10, 104-111 (1989)

DIIS (Direct Inversion in the Iterative Subspace) -
P.Pulay J.Comput.Chem. 3, 556-560(1982)

SOSCF -

G.Chaban, M.W.Schmidt, M.S.Gordon
Theor.Chem.Acc. 97, 88-95(1997)
T.H.Fischer, J.Almlof J.Phys.Chem. 96,9768-74(1992)

Modified Virtual Orbitals (MVOs) -

C.W.Bauschlicher, Jr. J.Chem.Phys. 72,880-885(1980)

Thermochemistry (RUNTYP=G3MP2) -

G3(MP2,CCSD(T)) is defined in
L.A.Curtiss, K.Ragavachari, P.C.Redfern, A.G.Baboul,
J.A.Pople Chem.Phys.Lett. 314, 101-107(1999)
based on various other G3 basis set/method papers:
L.A.Curtiss, P.C.Redfern, K.Raghavachari, V.Rassolov,
J.A.Pople J.Chem.Phys. 110, 4703-4709(1999)
L.A.Curtiss, P.C.Redfern, K.Raghavachari, V.Rassolov,
J.A.Pople J.Chem.Phys. 114, 9287-9295(2001)
L.A.Curtiss, P.C.Redfern, K.Raghavachari, V.Rassolov,
J.A.Pople J.Chem.Phys. 109,7764-7776(1998)
L.A.Curtiss, K.Ragavachari
Theoret.Chem.Acc. 108, 61-70(2002)

EVVRSP, in memory diagonalization -

S.T.Elbert Theoret.Chim.Acta 71,169-186(1987)

Davidson eigenvector method -

E.R.Davidson J.Comput.Phys. 17,87(1975)
"Matrix Eigenvector Methods" p. 95-113 in "Methods in
Computational Molecular Physics", edited by G.H.F.Diercksen
and S.Wilson, D.Reidel Publishing, Dordrecht, 1983.
M.L.Leininger, C.D.Sherrill, W.D.Allen, H.F.Schaefer,
J.Comput.Chem. 22, 1574-1589(2001)

DK (Douglas-Kroll relativistic transformation) -

M.Douglas, N.M.Kroll Ann.Phys. 82, 89-155(1974)
B.A.Hess Phys.Rev. A33, 3742-3748(1986)
G.Jansen, B.A.Hess Phys.Rev. A39, 6016-6017(1989)
T.Nakajima, K.Hirao J.Chem.Phys. 113, 7786-7789(2000)
T.Nakajima, K.Hirao Chem.Phys.Lett. 329, 511-516(2000)
W.A.DeJong, R.J.Harrison, D.A.Dixon

J.Chem.Phys. 114, 48-53(2001)
A.Wolf, M.Reiher, B.A.Hess J.Chem.Phys. 117, 9215-26(2002)
T.Nakajima, K.Hirao J.Chem.Phys. 119, 4105-4111(2003)
(and see just below for DK1 during SOC)

IOTC (Infinite-Order Two-Component) relativy correction -
M.Barysz, A.J.Sadlej J.Chem.Phys. 116, 2696-2704(2002)
M.Barysz, Progress in Theoretical Chemistry and Physics,
Kluwer Academic Publishers, 349-397(2002)
D.Kedziera, M.Barysz, A.J.Sadlej
Struct.Chem. 15, 369-377(2004)
D.Kedziera, M.Barysz, J.Chem.Phys. 121, 6719-6727(2004)
M.Barysz, L.Mentel, J.Leszczynski
J.Chem.Phys. 130, 164114/1-7(2009)

RESC (Relativistic Elimination of Small Components) -
T.Nakajima, K.Hirao Chem.Phys.Lett. 302, 383-391(1999)
T.Nakajima, T.Suzumura, K.Hirao
Chem.Phys.Lett. 304, 271(1999)
D.G.Fedorov, T.Nakajima, K.Hirao
Chem.Phys.Lett. 335, 183-187(2001)

NESC (Normalized Elimination of Small Components) -
K.G.Dyall J.Comput.Chem. 23, 786-793(2002)

Spin-orbit coupling and transition moments -
Many references can be found in the section on this topic
below.

GIAO NMR -
R.Ditchfield Mol.Phys. 27, 789-807(1974)
M.A.Freitag, B.Hillman, A.Agrawal, M.S.Gordon
J.Chem.Phys. 120, 1197-1202(2004)

Solvation models: EFP, SCRF, PCM, or COSMO.
All appropriate references are included in the sections on
these topics included below.

MOPAC 6 -
J.J.P.Stewart
J.Computer-Aided Molecular Design 4, 1-105 (1990)
References for parameters for individual atoms may be found
on the printout from your runs.

MacMolPlt -
B.M.Bode, M.S.Gordon J.Mol.Graphics Mod. 16, 133-138(1998)

quantum chemistry parallelization in GAMESS -
for SCF, see the main GAMESS paper quoted above.

- T.L.Windus, M.W.Schmidt, M.S.Gordon,
Chem.Phys.Lett. 216, 375-379(1993)
- T.L.Windus, M.W.Schmidt, M.S.Gordon,
Theoret.Chim.Acta 89, 77-88 (1994)
- T.L.Windus, M.W.Schmidt, M.S.Gordon, in "Parallel Computing
in Computational Chemistry", ACS Symposium Series 592,
Ed. by T.G.Mattson, ACS Washington, 1995, pp 16-28.
- K.K.Baldrige, M.S.Gordon, J.H.Jensen, N.Matsunaga,
M.W.Schmidt, T.L.Windus, J.A.Boatz, T.R.Cundari
ibid, pp 29-46.
- G.D.Fletcher, M.W.Schmidt, M.S.Gordon
Adv.Chem.Phys. 110, 267-294 (1999)
- H.Umeda, S.Koseki, U.Nagashima, M.W.Schmidt
J.Comput.Chem. 22, 1243-1251 (2001)
- C.H.Choi, K.Ruedenberg J.Comput.Chem. 22, 1484-1501(2001)
- D.G.Fedorov, M.S.Gordon ACS Symp.Series 828, 1-22(2002)
- H.Li, C.S.Pomelli, J.H.Jensen
Theoret.Chem.Acc. 109, 71-84(2003)
- C.M.Aikens, M.S.Gordon J.Phys.Chem.A 108, 3103-3110(2004)
- H.M.Netzloff, M.S.Gordon J.Comput.Chem. 25, 1926-1936(2004)
- T.J.Dudley, R.M.Olson, M.W.Schmidt, M.S.Gordon
J.Comput.Chem. 27, 353-362(2006)
- C.M.Aikens, G.D.Fletcher, M.W.Schmidt, M.S.Gordon
J.Chem.Phys. 124, 014107/1-14(2006)
- Y.Alexeev, M.W.Schmidt, T.L.Windus, M.S.Gordon
J.Comput.Chem. 28, 1685-1694(2007).
- R.M.Olson, J.L.Bentz, R.A.Kendall, M.W.Schmidt, M.S.Gordon
J.Comput.Theoret.Chem. 3, 1312-1328(2007)
- J.L.Bentz, R.M.Olson, M.S.Gordon, M.W.Schmidt, R.A.Kendall
Comput.Phys.Comm., 176, 589-600 (2007).
- G.D.Fletcher Mol.Phys. 105, 2971-2976(2007)

The Distributed Data Interface (DDI), which is the computer science layer underneath the parallel quantum chemistry -

- G.D.Fletcher, M.W.Schmidt, B.M.Bode, M.S.Gordon
Comput.Phys.Comm. 128, 190-200 (2000)

- R.M.Olson, M.W.Schmidt, M.S.Gordon, A.P.Rendell
Proc. of Supercomputing 2003, IEEE Computer Society.
This does not exist on paper, but can be downloaded at
http://www.sc-conference.org/sc2003/tech_papers.php

- D.G.Fedorov, R.M.Olson, K.Kitaura, M.S.Gordon, S.Koseki
J.Comput.Chem. 25, 872-880(2004).

Basis Set References

An excellent review of the relationship between the atomic basis used, and the accuracy with which various molecular properties will be computed is:

E.R.Davidson, D.Feller Chem.Rev. 86, 681-696(1986).

STO-NG	H-Ne	Ref. 1 and 2
	Na-Ar,	Ref. 2 and 3 **
	K,Ca,Ga-Kr	Ref. 4
	Rb,Sr,In-Xe	Ref. 5
	Sc-Zn,Y-Cd	Ref. 6

- 1) W.J.Hehre, R.F.Stewart, J.A.Pople
J.Chem.Phys. 51, 2657-2664(1969).
- 2) W.J.Hehre, R.Ditchfield, R.F.Stewart, J.A.Pople
J.Chem.Phys. 52, 2769-2773(1970).
- 3) M.S.Gordon, M.D.Bjorke, F.J.Marsh, M.S.Korth
J.Am.Chem.Soc. 100, 2670-2678(1978).
- ** the valence scale factors for Na-Cl are taken
from this paper, rather than the "official"
Pople values in Ref. 2.
- 4) W.J.Pietro, B.A.Levi, W.J.Hehre, R.F.Stewart,
Inorg.Chem. 19, 2225-2229(1980).
- 5) W.J.Pietro, E.S.Blurock, R.F.Hout,Jr., W.J.Hehre, D.J.
DeFrees, R.F.Stewart Inorg.Chem. 20, 3650-3654(1980).
- 6) W.J.Pietro, W.J.Hehre J.Comput.Chem. 4, 241-251(1983).

MINI/MIDI	H-Xe	Ref. 9
-----------	------	--------

- 9) "Gaussian Basis Sets for Molecular Calculations"
S.Huzinaga, J.Andzelm, M.Klobukowski, E.Radzio-Andzelm,
Y.Sakai, H.Tatewaki Elsevier, Amsterdam, 1984.
This book is referred to in certain circles as "the green
book" based on the color of its cover.

The MINI bases are three Gaussian expansions of each atomic orbital. The exponents and contraction coefficients are optimized for each element, and s and p exponents are not constrained to be equal. As a result these bases give much lower energies than does STO-3G. The valence MINI orbitals of main group elements are scaled by factors optimized by John Deisz at North Dakota State University. Transition metal MINI bases are not scaled. The MIDI bases are derived from the MINI sets by floating the outermost

primitive in each valence orbitals, and renormalizing the remaining 2 gaussians. MIDI bases are not scaled by GAMESS. The transition metal bases are taken from the lowest SCF terms in the s**1,d**n configurations.

3-21G	H-Ne	Ref. 10	(also 6-21G)
	Na-Ar	Ref. 11	(also 6-21G)
K,Ca,Ga-Kr,Rb,Sr,In-Xe		Ref. 12	
	Sc-Zn	Ref. 13	
	Y-Cd	Ref. 14	

- 10) J.S.Binkley, J.A.Pople, W.J.Hehre
J.Am.Chem.Soc. 102, 939-947(1980).
- 11) M.S.Gordon, J.S.Binkley, J.A.Pople, W.J.Pietro,
W.J.Hehre J.Am.Chem.Soc. 104, 2797-2803(1982).
- 12) K.D.Dobbs, W.J.Hehre J.Comput.Chem. 7,359-378(1986)
- 13) K.D.Dobbs, W.J.Hehre J.Comput.Chem. 8,861-879(1987)
- 14) K.D.Dobbs, W.J.Hehre J.Comput.Chem. 8,880-893(1987)

N-31G	references for	4-31G	5-31G	6-31G
	H	15	15	15
	He	23	23	23
	Li	19,24		19
	Be	20,24		20
	B	17		19
	C-F	15	16	16
	Ne	23		23
	Na-Al			22
	Si			21 **
	P-Cl	18		22
	Ar			22
	K-Kr			26

- 15) R.Ditchfield, W.J.Hehre, J.A.Pople
J.Chem.Phys. 54, 724-728(1971).
- 16) W.J.Hehre, R.Ditchfield, J.A.Pople
J.Chem.Phys. 56, 2257-2261(1972).
- 17) W.J.Hehre, J.A.Pople J.Chem.Phys. 56, 4233-4234(1972).
- 18) W.J.Hehre, W.A.Lathan J.Chem.Phys. 56,5255-5257(1972).
- 19) J.D.Dill, J.A.Pople J.Chem.Phys. 62, 2921-2923(1975).
- 20) J.S.Binkley, J.A.Pople J.Chem.Phys. 66, 879-880(1977).
- 21) M.S.Gordon Chem.Phys.Lett. 76, 163-168(1980)

** - Note that the built in 6-31G basis for Si is not that given by Pople in reference 22. The Gordon basis gives a better wavefunction, for a ROHF calculation in full atomic (Kh)

symmetry,		
6-31G	Energy	virial
Gordon	-288.828573	1.999978
Pople	-288.828405	2.000280

See the input examples for how to run in Kh.

- 22) M.M.Francl, W.J.Pietro, W.J.Hehre, J.S.Binkley,
M.S.Gordon, D.J.DeFrees, J.A.Pople
J.Chem.Phys. 77, 3654-3665(1982).
- 23) Unpublished, copied out of GAUSSIAN82.
- 24) For Li and Be, 4-31G is actually a 5-21G expansion.
- 25) V.A.Rassolov, J.A.Pople, M.A.Ratner, T.L.Windus
J.Chem.Phys. 109, 1223-1229(1998)
- 26) A.V.Mitin, J.Baker, P.Pulay
J.Chem.Phys. 118, 7775-7782(2003) - not in GAMESS.
- 27) V.A.Rassolov, M.A.Ratner, J.A.Pople, P.C.Redfern,
L.A.Curtiss J.Comput.Chem. 22, 976-984(2001).

Note that reference 27 renames basis sets published earlier as "6-31G*" in references 25 and 32. GAMESS was changed to use the 6-31G* basis sets from reference 27 for K, Ca, and Ga-Kr in September 2006. Sc-Zn remain those of ref. 25.

Extended basis sets

--> 6-311G

- 28) R.Krishnan, J.S.Binkley, R.Seeger, J.A.Pople
J.Chem.Phys. 72, 650-654(1980).

--> valence double zeta "DZV" sets:

"DH" basis - DZV for H, Li-Ne, Al-Ar

- 30) T.H.Dunning, Jr., P.J.Hay Chapter 1 in "Methods of
Electronic Structure Theory", H.F.Schaefer III, Ed.
Plenum Press, N.Y. 1977, pp 1-27.
Note that GAMESS uses inner/outer scale factors of
1.2 and 1.15 for DH's hydrogen (since at least 1983).
To get Thom's usual basis, scaled 1.2 throughout:

```
HYDROGEN  1.0  x, y, z
DH 0  1.2  1.2
```

DZV for K,Ca

- 31) J.-P.Blaudeau, M.P.McGrath, L.A.Curtiss, L.Radom
J.Chem.Phys. 107, 5016-5021(1997)
"BC" basis - DZV for Ga-Kr
- 32) R.C.Binning, Jr., L.A.Curtiss
J.Comput.Chem. 11, 1206-1216(1990)

Note, this basis set is available only by GBASIS=DZV, since it is no longer considered to be the 6-31G substitute.

--> valence triple zeta "TZV" sets:

- TZV for H,Li-Ne
- 40) T.H. Dunning, J.Chem.Phys. 55 (1971) 716-723.
TZV for Na-Ar - also known as the "MC" basis
- 41) A.D.McLean, G.S.Chandler
J.Chem.Phys. 72,5639-5648(1980).
TZV for K,Ca
- 42) A.J.H. Wachters, J.Chem.Phys. 52 (1970) 1033-1036.
(see Table VI, Contraction 3).
TZV for Sc-Zn (taken from HONDO 7)
This is Wachters' (14s9p5d) basis (ref 42) contracted
to (10s8p3d) with the following modifications
1. the most diffuse s removed;
 2. additional s spanning 3s-4s region;
 3. two additional p functions to describe the 4p;
 4. (6d) contracted to (411) from ref 43,
except for Zn where Wachter's (5d)/[41]
and Hay's diffuse d are used.
- 43) A.K. Rappe, T.A. Smedley, and W.A. Goddard III,
J.Phys.Chem. 85 (1981) 2607-2611

Valence only basis sets (ECPs and MCPs)

SBKJC ECP, these are -31G splits for main group, bigger for transition metals (available Li-Rn):

- 50) W.J.Stevens, H.Basch, M.Krauss
J.Chem.Phys. 81, 6026-6033 (1984)
- 51) W.J.Stevens, M.Krauss, H.Basch, P.G.Jasien
Can.J.Chem. 70, 612-630 (1992)
- 52) T.R.Cundari, W.J.Stevens
J.Chem.Phys. 98, 5555-5565(1993)

HW ECP, these are -21 splits (sp exponents not shared)
transition metals (not built in at present, although
they will work if you type them in):

- 53) P.J.Hay, W.R.Wadt J.Chem.Phys. 82, 270-283 (1985)
main group (available Na-Xe)
- 54) W.R.Wadt, P.J.Hay J.Chem.Phys. 82, 284-298 (1985)
see also
- 55) P.J.Hay, W.R.Wadt J.Chem.Phys. 82, 299-310 (1985)

Model core potentials (MCP):

To understand the model core potential formalism itself,
see the review articles

S.Huzinaga Can.J.Chem. 73, 619-628(1995)

M.Klobukowski, S.Huzinaga, Y.Sakai, in Computational Chemistry: Reviews of current trends, volume 3, pp 49-74, edited by J.Leszczynski, World Scientific, Singapore, 1999.

The MCP-xZP, MCP-AxZP, MCP-CxZP, MCP-ACxZP families:

- 60) Y.Sakai, E.Miyoshi, M.Klobukowski, S.Huzinaga,
"Model potentials for main group elements",
J. Chem. Phys. 106, 8084-8092 (1997).
- 61) E. Miyoshi, Y. Sakai, K. Tanaka, M. Masamura,
"Relativistic dsp-model core potentials for main group
elements in the fourth, fifth and sixth row
and their applications",
J. Mol. Struct. (THEOCHEM) 451, 73-79 (1998)
- 62) Y. Sakai, E. Miyoshi, H. Tatewaki,
"Model core potentials for the lanthanides",
J. Mol. Struct. (THEOCHEM) 451, 143-150 (1998)
- 63) E.Miyoshi, H.Mori, R.Hirayama, Y.Osanai, T.Noro,
H.Honda, M.Klobukowski
"Compact and efficient basis sets of s- and p-block
elements for model core potential method"
J.Chem.Phys. 122, 074104/1-8(2005)
- 64) M. Sekiya, T. Noro, Y. Osanai, E. Miyoshi, T. Koga,
"Relativistic Correlating Basis Sets for Lanthanide
Atoms from Ce to Lu",
J. Comput. Chem. 27, 463 (2006)
- 65) H. Anjima, S. Tsukamoto, H. Mori, H. Mine,
M. Klobukowski, E. Miyoshi,
"Revised Model Core Potentials of s-Block Elements",
J. Comput. Chem. 28, 2424-2430 (2007)
- 66) Y. Osanai, M. S. Mon, T. Noro, H. Mori,
H. Nakashima, M. Klobukowski, E. Miyoshi,
"Revised model core potentials for first-row
transition-metal atoms from Sc to Zn",
Chem. Phys. Lett. 452, 210-214 (2008)
- 67) Y. Osanai, E. Soejima, T. Noro, H. Mori, M. Ma San,
M. Klobukowski, E. Miyoshi,
"Revised model core potentials for second-row
transition metal atoms from Y to Cd",
Chem. Phys. Lett. 463, 230-234 (2008)
- 68) H. Mori, K. Ueno-Noto, Y. Osanai, T. Noro, T. Fujiwara,
M. Klobukowski, E. Miyoshi,
"Revised model core potentials for third-row
transition-metal atoms from Lu to Hg",
Chem. Phys. Lett. 476, 317-322 (2009)

the iMCP (improved model core families) are:

- 71) C.C.Lovallo, M.Klobukowski
J.Comput.Chem. 24, 1009-10015(2003)
- 72) C.C.Lovallo, M.Klobukowski

J.Comput.Chem. 25, 1206-1213(2004)

the ZFK (Zeng, Fedorov, Klobukowski) family for sp block:

72) T.Zeng, D.G.Fedorov, M. Klobukowski
J.Chem.Phys. 133, 114107/1-11 (2010)

For additional information, see also

T.Zeng, D.G.Fedorov, M. Klobukowski
J.Chem.Phys. 131, 124109/1-17 (2009)

T.Zeng, D.G.Fedorov, M.Klobukowski
J.Chem.Phys. 132, 074102/1-15 (2010)

The MCP family, built into the \$DATA group only:

75) Y.Sakai, E.Miyoshi, M.Klobukowski, S.Huzinaga,
"Model potentials for molecular calculations. I.
The sd-MP set for transition metal atoms Sc-Hg",
J. Comput. Chem. 8 (1987) 226-255.

76) Y.Sakai, E.Miyoshi, M.Klobukowski, S.Huzinaga,
"Model potentials for molecular calculations. II.
The spd-MP set for transition metal atoms Sc-Hg",
J. Comput. Chem. 8 (1987) 256-264.

77) Y.Sakai, E.Miyoshi, M.Klobukowski, S.Huzinaga,
"Model potentials for main group elements",
J. Chem. Phys. 106 (1997) 8084-8092.

78) E.Miyoshi, Y.Sakai, K.Tanaka, M.Masamura
"Relativistic dsp-Model Core Potentials for Main Group
Elements in the 4th, 5th, and 6th-Row and Applications"
J. Mol. Struct. (Theochem), 451 (1998) 73-79.

79) Y.Sakai, E.Miyoshi, H.Tatewaki
"Model Core Potentials for the Lanthanides"
J. Mol. Struct. (Theochem), 451 (1998) 143-150.

Systematic basis set families:

Polarization Consistent basis sets (PCseg-n):

The segmented contractions which are internally stored in GAMESS are described in this paper:

81) F.Jensen, J.Chem.Theory Comp. 10, 1074-1085(2014)

Papers describing the older general contractions are:

F.Jensen J.Chem.Phys. 115, 9113-9125(2001).

erratum J.Chem.Phys. 116, 3502(2002).

F.Jensen J.Chem.Phys. 116, 7372-7379(2002).

F.Jensen J.Chem.Phys. 117, 9234-9240(2002).

F.Jensen J.Chem.Phys. 118, 2459-2463(2003).

F.Jensen, T.Helgaker J.Chem.Phys. 121, 3463-3470(2004).

F.Jensen, J. Phys. Chem. A 111, 11198-11204(2007)

F.Jensen, J. Chem. Phys. 136, 114107(2012)

F.Jensen, J. Chem. Phys. 138, 014107(2013)

Correlation Consistent bases (CCn, ACCn, etc.):

The GAMESS keyword and official names for these "Dunning-style" basis sets are,

CCn=cc-pVnZ, ACCn=aug-cc-pVnZ, n=D,T,Q,5,...
 CCnC=cc-pCVnZ, ACCnC=aug-cc-pCVnZ,
 CCnWC=cc-pwCVnZ, ACCnWC=aug-cc-pwCVnZ (w="omega").
 See \$BASIS for important information about Al-Ar's bases,
 where the GAMESS keyword invokes "tight d" (n+d) sets.

Please see the Pacific Northwest National Laboratory web
 page <http://www.emsl.pnl.gov/forms/basisform.html> for
 references to these basis sets. Kirk Peterson's very
 thorough bibliography can be found at
<http://tyr0.chem.wsu.edu/~kipeters/basis-bib.html>

Sapporo (SPK) basis set family

- first, the non-relativistic valence sets,
 S1. H.Tatewaki, T.Koga J.Chem.Phys. 104, 8493(1996)
 S2. H.Tatewaki, T.Koga, H.Takashima
 Theoret.Chem.Acc. 96, 243(1997)
 S3. T.Koga, H.Tatewaki, Y.Satoh
 Theoret.Chem.Acc. 102, 105(1999)
 S4. T.Koga, S.Yamamoto, T.Shimazaki, H.Tatewaki,
 Theoret.Chem.Acc. 108, 41(2002)
 then, the relativistic valence sets,
 S6. T.Noro, M.Sekiya, T.Koga, S.L.Saito
 Chem.Phys.Lett. 481, 229-233(2009)
 core/valence relativistic and non-relativistic:
 S7. T.Noro, M.Sekiya, T.Koga (main group)
 Theoret.Chem.Acc. 131, 1124(2012)
 S8. M.Sekiya, T.Noro, T.Koga, T.Shimuzaki (lanthanides)
 Theoret.Chem.Acc. 131, 1247(2012)

Karlsruhe basis sets (group of Reinhart Ahlrichs)

- 91) A.Schaefer, H.Horn, R.Ahlrichs
 J.Chem. Phys. 97,2571 (1992).
 92) A.Schaefer, C.Huber, R.Ahlrichs
 J.Chem. Phys. 100, 5829 (1994).

Polarization exponents:

- STO-NG*
 100) J.B.Collins, P. von R. Schleyer, J.S.Binkley,
 J.A.Pople J.Chem.Phys. 64, 5142-5151(1976).

3-21G*. See also reference 12.

- 101) W.J.Pietro, M.M.Francl, W.J.Hehre, D.J.DeFrees, J.A. Pople, J.S.Binkley J.Am.Chem.Soc. 104,5039-5048(1982)

6-31G* and 6-31G**. See also reference 22 above.

- 102) P.C.Hariharan, J.A.Pople
Theoret.Chim.Acta 28, 213-222(1973)

multiple polarization, and f functions

- 103) M.J.Frisch, J.A.Pople, J.S.Binkley J.Chem.Phys.
80, 3265-3269 (1984).

Anion diffuse functions:

3-21+G, 3-21++G, etc.

- 105) T.Clark, J.Chandrasekhar, G.W.Spitznagel, P. von R. Schleyer J.Comput.Chem. 4, 294-301(1983)
106) G.W.Spitznagel, Diplomarbeit, Erlangen, 1982.

STO-NG* means d orbitals are used on third row atoms only. The original paper (ref 100) suggested $z=0.09$ for Na and Mg, and $z=0.39$ for Al-Cl. We prefer to use the same exponents as are used in 3-21G* and 6-31G*, so we know we're looking at changes in the sp basis, not the d exponent.

3-21G* means d orbitals on main group elements in the third and higher periods. Not defined for the transition metals, where there are p's already in the basis. Except for alkalis and alkali earths, the 4th and 5th row zetas are from Huzinaga, et al. (ref 9). The exponents are normally the same as for 6-31G*.

6-31G* means d orbitals on second and third row atoms. We use Mark Gordon's $z=0.395$ for Silicon, as well as his fully optimized sp basis (ref 21). This is often written 6-31G(d) today. For the first row transition metals, the * means an f function is added. The transition metal 3d 6-31G orbital is NOT of triple zeta quality, and thus is probably not very accurate.

6-31G** means the same as 6-31G*, except that p functions are added on hydrogens. This is often written 6-31G(d,p) today.

6-311G** means p orbitals on H, and d orbitals elsewhere. The exponents were derived from correlated atomic states, and so are considerably tighter than the polarizing functions used in 6-31G**, etc. This is often written 6-311G(d,p) today.

The exponents for 6-31G* for C-F are disturbing, in that each atom has exactly the same value. Dunning and Hay (ref 30) have recommended a better set of exponents for second row atoms and a slightly different value for H.

2p, 3p, 2d, 3p polarization sets are usually thought of as arising from applying splitting factors to the 1p and 1d values. For example, SPLIT2=2.0, 0.5 means to double and halve the single value. The default values for SPLIT2 and SPLIT3 are taken from reference 103, and were derived with correlation in mind. The SPLIT2 values often produce a higher (!) HF energy than the singly polarized run, because the exponents are split too widely. SPLIT2=0.4,1.4 will always lower the SCF energy (the values are the unpublished personal preference of MWS), and for SPLIT3 we might suggest 3.0,1.0,1/3.

With all this as background, we are ready to present the tables of polarization exponents that are built into GAMESS. Please note that the names associated with each column are only generally descriptive. The column marked "COMMON" is obtained from both Pople (mostly his 6-31G, but using Gordon's value for Silicon) and Huzinaga (from the "green book"). The exponents for K-Kr under "Dunning" are from Curtiss, et al., not Thom Dunning, and so on. The exponents are for d functions unless otherwise indicated.

Polarization exponents, chosen by POLAR= in \$BASIS:

	COMMON	POPN31	POPN311	DUNNING	HUZINAGA	HOND07
	-----	-----	-----	-----	-----	-----
H	1.1(p)		0.75(p)	1.0(p)	1.0(p)	1.0(p)
He	1.1(p)		0.75(p)	1.0(p)	1.0(p)	1.0(p)
Li	0.2		0.200		0.076(p)	
Be	0.4		0.255		0.164(p)	0.32
B	0.6		0.401	0.70	0.388	0.50
C	0.8		0.626	0.75	0.600	0.72
N	0.8		0.913	0.80	0.864	0.98
O	0.8		1.292	0.85	1.154	1.28
F	0.8		1.750	0.90	1.496	1.62
Ne	0.8		2.304	1.00	1.888	2.00
Na	0.175				0.061(p)	0.157
Mg	0.175				0.101(p)	0.234
Al	0.325				0.198	0.311
Si	0.395				0.262	0.388
P	0.55				0.340	0.465
S	0.65				0.421	0.542
Cl	0.75				0.514	0.619
Ar	0.85				0.617	0.696
K	0.2	0.04485		0.260	0.039(p)	
Ca	0.2	0.0502		0.229	0.059(p)	
Sc-Zn	N/A	0.8(f)	N/A	N/A	N/A	N/A
Ga	0.207	0.2289		0.141		
Ge	0.246	0.2772		0.202		
As	0.293	0.3277		0.273		
Se	0.338	0.3810		0.315		
Br	0.389	0.4366		0.338		
Kr	0.443	0.4948		0.318		
Rb	0.11				0.034(p)	
Sr	0.11				0.048(p)	

A blank means the value equals the "COMMON" column.

Common d polarization for all sets ("green book"):

In	Sn	Sb	Te	I	Xe
0.160	0.183	0.211	0.237	0.266	0.297
Tl	Pb	Bi	Po	At	Rn
0.146	0.164	0.185	0.204	0.225	0.247

see f exponents on next page...

f polarization functions, from reference 103:

Li	Be	B	C	N	O	F	Ne
0.15	0.26	0.50	0.80	1.00	1.40	1.85	2.50
Na	Mg	Al	Si	P	S	Cl	Ar
0.15	0.20	0.25	0.32	0.45	0.55	0.70	--

Anions usually require diffuse basis functions to properly represent their spatial diffuseness. The use of diffuse sp shells on atoms in the second and third rows is denoted by a + sign, also adding diffuse s functions on hydrogen is symbolized by ++. These designations can be applied to any of the Pople bases, e.g. 3-21+G, 3-21+G*, 6-31++G**. The following exponents are for L shells, except for H. For H-F, they are taken from ref 105. For Na-Cl, they are taken directly from reference 106. These values may be found in footnote 13 of reference 103. For Ga-Br, In-I, and Tl-At these were optimized for the atomic ground state anion, using ROHF with a flexible ECP basis set, by Ted Packwood at NDSU.

H						
0.0360						
Li	Be	B	C	N	O	F
0.0074	0.0207	0.0315	0.0438	0.0639	0.0845	0.1076
Na	Mg	Al	Si	P	S	Cl
0.0076	0.0146	0.0318	0.0331	0.0348	0.0405	0.0483
		Ga	Ge	As	Se	Br
		0.0205	0.0222	0.0287	0.0318	0.0376
		In	Sn	Sb	Te	I
		0.0223	0.0231	0.0259	0.0306	0.0368
		Tl	Pb	Bi	Po	At
		0.0170	0.0171	0.0215	0.0230	0.0294

Additional information about diffuse functions and also Rydberg type exponents can be found in reference 30.

The following atomic energies are UHF (RHF on 1-S states), p orbitals are not symmetry equivalent, using the default scale factors. They may be useful in picking a basis of the desired accuracy.

Atom state	ST0-2G	ST0-3G	3-21G	6-31G
H 2-S	-.454397	-.466582	-.496199	-.498233
He 1-S	-2.702157	-2.807784	-2.835680	-2.855160
Li 2-S	-7.070809	-7.315526	-7.381513	-7.431236
Be 1-S	-13.890237	-14.351880	-14.486820	-14.566764
B 2-P	-23.395284	-24.148989	-24.389762	-24.519492
C 3-P	-36.060274	-37.198393	-37.481070	-37.677837
N 4-S	-53.093007	-53.719010	-54.105390	-54.385008
O 3-P	-71.572305	-73.804150	-74.393657	-74.780310
F 2-P	-95.015084	-97.986505	-98.845009	-99.360860
Ne 1-S	-122.360485	-126.132546	-127.803825	-128.473877
Na 2-S	-155.170019	-159.797148	-160.854065	-161.841425
Mg 1-S	-191.507082	-197.185978	-198.468103	-199.595219
Al 2-P	-233.199965	-239.026471	-240.551046	-241.854186
Si 3-P	-277.506857	-285.563052	-287.344431	-288.828598
P 4-S	-327.564244	-336.944863	-339.000079	-340.689008
S 3-P	-382.375012	-393.178951	-395.551336	-397.471414
Cl 2-P	-442.206260	-454.546015	-457.276552	-459.442939
Ar 1-S	-507.249273	-521.222881	-524.342962	-526.772151

Atom state	DH	6-311G	MC	SCF limit*
H 2-S	-.498189	-.499810	--	-0.5
He 1-S	--	-2.859895	--	-2.861680
Li 2-S	-7.431736	-7.432026	--	-7.432727
Be 1-S	-14.570907	-14.571874	--	-14.573023
B 2-P	-24.526601	-24.527020	--	-24.529061
C 3-P	-37.685571	-37.686024	--	-37.688619
N 4-S	-54.397260	-54.397980	--	-54.400935
O 3-P	-74.802707	-74.802496	--	-74.809400
F 2-P	-99.395013	-99.394158	--	-99.409353
Ne 1-S	-128.522354	-128.522553	--	-128.547104
Na 2-S	--	--	-161.845587	-161.858917
Mg 1-S	--	--	-199.606558	-199.614636
Al 2-P	-241.855079	--	-241.870014	-241.876699
Si 3-P	-288.829617	--	-288.847782	-288.854380
P 4-S	-340.689043	--	-340.711346	-340.718798
S 3-P	-397.468667	--	-397.498023	-397.504910
Cl 2-P	-459.435938	--	-459.473412	-459.482088
Ar 1-S	--	--	-526.806626	-526.817528

* M.W.Schmidt and K.Ruedenberg, J.Chem.Phys. 71, 3951-3962(1979). These are ROHF energies in Kh symmetry. H-Xe can be found in Phys.Rev.A 46, 3691-3696(1992).

Spherical Harmonics

The implementation of ISPHER in \$CONTRL does not rely on using a spherical harmonic basis set, in fact the atomic basis remains the Cartesian Gaussians. Instead, certain MOs formed from particular combinations of the Cartesian Gaussians (for example, $xx+yy+zz$) are deleted from the MO space. Thus a run with ISPHER=1 will have fewer MOs than AOs. Since neither the occupied nor virtual MOs contain any admixture of $xx+yy+zz$, the resulting energy and wave-function is exactly equivalent to the use of a spherical harmonic basis.

The log file output will contain expansions of each MO in terms of 6 d's, 10 f's, and 15 g's, and the \$VEC also contains the same expansion over Cartesian Gaussians. Both the matrix in your log file and in \$VEC will contain fewer MOs than AOs, the exact number of MOs used is printed in the initial guess section of the log file. It should be possible to read such \$VEC groups into runs with different settings of ISPHER, should you choose to do so.

The advantage of this approach is that intelligence in the generation of symmetry orbitals combined with the capability to drop linearly dependent MO combinations means that the details of ISPHER are located only in the orbital optimization code, where the variational spaces are simply reduced in size to eliminate the undesired contaminant functions. This means that none of the integral routines need be modified, as the atomic basis remains the Cartesian Gaussians. The disadvantage is that AO integral files run over the Cartesian Gaussians, and thus are not reduced in size. Of course transformed MO integrals and various computations in correlated calculations are reduced in size, since the number of MOs may be greatly reduced.

Computationally, the advantages of ISPHER=1 are not limited to the reduced CPU time associated with fewer total MOs. Questions about d orbital participation as measured by Mulliken populations are cleanly addressed when the d's usage in the MOs does not contain any contamination from the s shape $xx+yy+zz$. Less obviously, the use of spherical harmonics frequently greatly reduces problems with linear dependency, that exhibit as poor SCF convergence.

How to do RHF, ROHF, UHF, and GVB calculations

general considerations

These four SCF wavefunctions are all based on Fock operator techniques, even though some GVB runs use more than one determinant. Thus all of these have an intrinsic N^4 time dependence, because they are all driven by integrals in the AO basis. This similarity makes it convenient to discuss them all together. In this section we will use the term HF to refer generically to any of these four wavefunctions, including the multi-determinate GVB-PP functions. \$SCF is the main input group for all these HF wavefunctions.

As will be discussed below, in GAMESS the term ROHF refers to high spin open shell SCF only, but other open shell coupling cases are possible using the GVB code.

Analytic gradients are implemented for every possible HF type calculation possible in GAMESS, and therefore numerical Hessians are available for each.

Analytic Hessian calculation is implemented for RHF, ROHF, and any GVB case with NPAIR=0 or NPAIR=1. Analytic Hessians are more accurate, and much more quickly computed than numerical Hessians, but require additional disk storage to perform an integral transformation, and also more physical memory.

The second order Moller-Plesset energy correction (MP2) is implemented for RHF, UHF, ROHF, and MCSCF wavefunctions. Analytic gradients may be obtained for MP2 with RHF, UHF, or ROHF reference wavefunctions, and MP2 level properties are therefore available for these, see MP2PRP in \$MP2. All other cases give properties for the SCF function.

Direct SCF is implemented for every possible HF type calculation. The direct SCF method may not be used with DEM convergence. Direct SCF may be used during energy, gradient, numerical or analytic Hessian, CI or MP2 energy correction, or localized orbital computations.

direct SCF

Normally, HF calculations proceed by evaluating a large number of two electron repulsion integrals, and storing these on a disk. This integral file is read in once during each HF iteration to form the appropriate Fock operators. In a direct HF, the integrals are not stored on disk, but are instead reevaluated during each HF iteration. Since the direct approach **always** requires more CPU time, the default for DIRSCF in \$SCF is .FALSE.

Even though direct SCF is slower, there are at least two reasons why you may want to consider using it. The first is that it may not be possible to store all of the integrals on the disk drives attached to your computer. Second, what you are really interested in is reducing the wall clock time to obtain your answer, not the CPU time. Workstations, particularly nodes with multiple CPUs and only one disk subsystem, may have modest hardware I/O capabilities. Other environments such as a mainframe shared by many users may also have very poor CPU/wall clock performance for I/O bound jobs such as conventional HF.

You can estimate the disk storage requirements for conventional HF using a P or PK file by the following formulae:

$$\begin{aligned} \text{nint} &= 1/\text{sigma} * 1/8 * N^{**4} \\ \text{Mbytes} &= \text{nint} * x / 1024^{**2} \end{aligned}$$

Here N is the total number of basis functions in your run, which you can learn from an EXETYP=CHECK run. The 1/8 accounts for permutational symmetry within the integrals. Sigma accounts for the point group symmetry, and is difficult to estimate accurately. Sigma cannot be smaller than 1, in no symmetry (C1) calculations. For benzene, sigma would be almost six, since you generate 6 C's and 6 H's by entering only 1 of each in \$DATA. For water sigma is not much larger than one, since most of the basis set is on the unique oxygen, and the C2v symmetry applies only to the H atoms. The factor x is 12 bytes per integral for basis sets smaller than 255, and 16 otherwise. Finally, since integrals that are very close to zero need not be stored on disk, the actual power dependence is not as bad as N**4, and in fact in the limit of very large molecules can be as low as N**2. Thus plugging in sigma=1 should give you an upper bound to the actual disk space needed. If the estimate exceeds your available disk storage, your only recourse is direct HF.

What are the economics of direct HF? Naively, if we assume the run takes 10 iterations to converge, we must spend 10 times more CPU time computing the integrals on each iteration. However, we do not have to waste any CPU time reading blocks of integrals from disk, or in unpacking their indices. We also do not have to waste any wall clock time waiting for a relatively slow mechanical device such as a disk to give us our data.

There are some less obvious savings too, as first noted by Almlof. First, since the density matrix is known while we are computing integrals, we can use the Schwarz inequality to avoid doing some of the integrals. In a conventional SCF this inequality is used to avoid doing small integrals. In a direct SCF it can be used to avoid doing integrals whose contribution to the Fock matrix is small (density times integral=small). Secondly, we can form the Fock matrix by calculating only its change since the previous iteration. The contributions to the change in the Fock matrix are equal to the change in the density times the integrals. Since the change in the density goes to zero as the run converges, we can use the Schwarz screening to avoid more and more integrals as the calculation progresses. The input option FDIFF in \$SCF selects formation of the Fock operator by computing only its change from iteration to iteration. The FDIFF option is not implemented for GVB since there are too many density matrices from the previous iteration to store, but is the default for direct RHF, ROHF, and UHF.

So, in our hypothetical 10 iteration case, we do not spend as much as 10 times more time in integral evaluation. Additionally, the run as a whole will not slow down by whatever factor the integral time is increased. A direct run spends no additional time summing integrals into the Fock operators, and no additional time in the Fock diagonalizations. So, generally speaking, a RHF run with 10-15 iterations will slow down by a factor of 2-4 times when run in direct mode. The energy gradient time is unchanged by direct HF, and this is a large time compared to HF energy, so geometry optimizations will be slowed down even less. This is really the converse of Amdahl's law: if you slow down only one portion of a program by a large amount, the entire program slows down by a much smaller factor.

To make this concrete, here are some times for GAMESS for a job which is a RHF energy for a SbC402NH4. These

timings were obtained an extremely long time ago, on a DECstation 3100 under Ultrix 3.1, which was running only these tests, so that the wall clock times are meaningful. This system is typical of Unix workstations in that it uses SCSI disks, and the operating system is not terribly good at disk I/O. By default GAMESS stores the integrals on disk in the form of a P supermatrix, because this will save time later in the SCF cycles. By choosing NOPK=1 in \$INTGRL, an ordinary integral file can be used, which typically contains many fewer integrals, but takes more CPU time in the SCF. Because the DECstation is not terribly good at I/O, the wall clock time for the ordinary integral file is actually less than when the supermatrix is used, even though the CPU time is longer. The run takes 13 iterations to converge, the times are in seconds.

	P supermatrix	ordinary file
# nonzero integrals	8,244,129	6,125,653
# blocks skipped	55,841	55,841
CPU time (ints)	709	636
CPU time (SCF)	1289	1472
CPU time (job total)	2123	2233
wall time (job total)	3468	3200

When the same calculation is run in direct mode (integrals are processed like an ordinary integral disk file when running direct),

	FDIFF=.TRUE.	FDIFF=.FALSE.
iteration 1:		
# nonzero integrals	6,117,416	6,117,416
# blocks skipped	60,208	60,208
iteration 13:		
# nonzero integrals	3,709,733	6,122,912
# blocks skipped	105,278	59,415
CPU time (job total)	6719	7851
wall time (job total)	6764	7886

Note that elimination of the disk I/O dramatically increases the CPU/wall efficiency. Here's the bottom line on direct HF:

$$\begin{aligned}\text{best direct CPU / best disk CPU} &= 6719/2123 = 3.2 \\ \text{best direct wall/ best disk wall} &= 6764/3200 = 2.1\end{aligned}$$

Direct SCF is slower than conventional disk SCF, but not outrageously so! From the data in the tables, we can see that the best direct method spends about $6719 - 1472 = 5247$ seconds doing integrals. This is an increase of about $5247/636 = 8.2$ in the time spent doing integrals, in a run

that does 13 iterations (13 times evaluating integrals). 8.2 is less than 13 because the run avoids all CPU charges related to I/O, and makes efficient use of the Schwarz inequality to avoid doing many of the integrals in its final iterations.

convergence accelerators

Generally speaking, the simpler the HF function, the better its convergence. In our experience, the majority of RHF and ROHF runs converge readily from GUESS=HUCKEL. UHF often takes considerably more iterations than either of these, due to the extremely common occurrence of heavy spin contamination. GVB runs typically require GUESS=MOREAD, although the Huckel guess usually works for NPAIR=0. GVB cases with NPAIR greater than one are particularly difficult.

Unfortunately, not all HF runs converge readily. The best way to improve your convergence is to provide better starting orbitals! In many cases, this means to MOREAD orbitals from some simpler HF case. For example, if you want to do a doublet ROHF, and the HUCKEL guess does not seem to converge, do this: Do an RHF on the +1 cation. RHF is typically more stable than ROHF, UHF, or GVB, and cations are usually readily convergent. Then MOREAD the cation's orbitals into the neutral calculation which you wanted to do at first.

GUESS=HUCKEL does not always start with the correct electronic configuration. It may be useful to use PRTMO in \$GUESS during a CHECK run to examine the starting orbitals, and then reorder them with NORDER if that seems appropriate.

Of course, by default GAMESS uses the convergence procedures which are usually most effective. Still, there are cases which are difficult, so the \$SCF group permits you to select several alternative methods for improving convergence. Briefly, these are

EXTRAP. This extrapolates the three previous Fock matrices, in an attempt to jump ahead a bit faster. This is the most powerful of the old-fashioned accelerators, and normally should be used at the beginning of any SCF run. When an extrapolation occurs, the counter at the left of the SCF printout is set to zero.

DAMP. This damps the oscillations between several successive Fock matrices. It may help when the energy is seen to oscillate wildly. Thinking about which orbitals should be occupied initially may be an even better way to avoid oscillatory behaviour.

SHIFT. Level shifting moves the diagonal elements of the virtual part of the Fock matrix up, in an attempt to uncouple the unoccupied orbitals from the occupied ones. At convergence, this has no effect on the orbitals, just their orbital energies, but will produce different (and hopefully better) orbitals during the iterations.

RSTRCT. This limits mixing of the occupied orbitals with the empty ones, especially the flipping of the HOMO and LUMO to produce undesired electronic configurations or states. This should be used with caution, as it makes it very easy to converge on incorrectly occupied electronic configurations, especially if DIIS is also used. If you use this, be sure to check your final orbital energies to see if they are sensible. A lower energy for an unoccupied orbital than for one of the occupied ones is a sure sign of problems.

DIIS. Direct Inversion in the Iterative Subspace is a modern method, due to Pulay, using stored error and Fock matrices from a large number of previous iterations to interpolate an improved Fock matrix. This method was developed to improve the convergence at the final stages of the SCF process, but turns out to be quite powerful at forcing convergence in the initial stages of SCF as well. By giving ETHRSH as 10.0 in \$SCF, you can practically guarantee that DIIS will be in effect from the first iteration. The default is set up to do a few iterations with conventional methods (extrapolation) before engaging DIIS. This is because DIIS can sometimes converge to solutions of the SCF equations that do not have the lowest possible energy. For example, the 3-A-2 small angle state of SiLi₂ (see M.S.Gordon and M.W.Schmidt, Chem.Phys.Lett., 132, 294-8(1986)) will readily converge with DIIS to a solution with a reasonable S*2, and an energy about 25 milliHartree above the correct answer. A SURE SIGN OF TROUBLE WITH DIIS IS WHEN THE ENERGY RISES TO ITS FINAL VALUE. However, if you obtain orbitals at one point on a PES without DIIS, the subsequent use of DIIS with MOREAD will probably not introduce any problems. Because DIIS is quite powerful, EXTRAP, DAMP, and SHIFT are all turned off once DIIS begins to work. DEM and RSTRCT will still be in use, however.

SOSCF. Approximate second-order (quasi-Newton) SCF orbital optimization. SOSCF will converge about as well as DIIS at the initial geometry, and slightly better at subsequent geometries. There's a bit less work solving the SCF equations, too. The method kicks in after the orbital gradient falls below SOGTOL. Some systems, particularly transition metals with ECP basis sets, may have Huckel orbitals for which the gradient is much larger than SOGTOL. In this case it is probably better to use DIIS instead, with a large ETHRSH, rather than increasing SOGTOL, since you may well be outside the quadratic convergence region. SOSCF does not exhibit true second order convergence since it uses an approximation to the inverse hessian. SOSCF will work for MOPAC runs, but is slower in this case. SOSCF will work for UHF, but its convergence may be better than DIIS. SOSCF will work for non-Abelian cases, but may encounter problems if the open shell is degenerate.

It should be clear that SOSCF and DIIS are the two work-horse convergers, with DAMP (and possibly SHIFT) useful in cases where the initial guess is such that these two are not engaged immediately. If you compute many different types of molecules, you will find cases where SOSCF works but DIIS does not, but also cases where DIIS works but SOSCF does not (although often both will work). If you do not obtain convergence with one of these, try the other one! If you still have problems, attempt to get better starting orbitals.

DEM. Direct energy minimization should be your last recourse. It explores the "line" between the current orbitals and those generated by a conventional change in the orbitals, looking for the minimum energy on that line. DEM should always lower the energy on every iteration, but is very time consuming, since each of the points considered on the line search requires evaluation of a Fock operator. DEM will be skipped once the density change falls below DEMCUT, as the other methods should then be able to affect final convergence. While DEM is working, RSTRCT is held to be true, regardless of the input choice for RSTRCT. Because of this, it behooves you to be sure that the initial guess is occupying the desired orbitals. DEM is available only for RHF. The implementation in GAMESS resembles that of R.Seeger and J.A.Pople, J.Chem.Phys. 65, 265-271(1976). Simultaneous use of DEM and DIIS resembles the ADEM-DIOS method of H.Sellers, Chem.Phys.Lett. 180, 461-465(1991). DEM does not work with direct SCF.

high spin open shell SCF (ROHF)

Open shell SCF calculations are performed in GAMESS by both the ROHF code and the GVB code. Note that when the GVB code is executed with no pairs, the run is NOT a true GVB run, and should be referred to in publications and discussion as a ROHF calculation. Low spin couplings are possible with the GVB program.

The ROHF module in GAMESS can handle any number of open shell electrons, provided these have a high spin coupling. For example: one open shell, doublet:

```
$CONTRL SCFTYP=ROHF MULT=2 $END
```

two open shells, triplet:

```
$CONTRL SCFTYP=ROHF MULT=3 $END
```

m open shells, high spin:

```
$CONTRL SCFTYP=ROHF MULT=m+1 $END
```

John Montgomery (who was then at United Technologies) is responsible for the ROHF implementation in GAMESS. The following discussion is due to him, dating from 1988 when his method of forming a combined Fock operator was included in GAMESS. Other choices (Euler and two "canonical" sets) were added to the table in 2009/2010.

The Fock matrix in the MO basis has the form

	closed		open		virtual
closed	F2		Fb		(Fa+Fb)/2
	-----		-----		-----
open	Fb		F1		Fa
	-----		-----		-----
virtual	(Fa+Fb)/2		Fa		F0

where Fa and Fb are the usual alpha and beta Fock matrices any UHF program produces. All ROHF methods agree on these, as they are the variational conditions that separate the doubly occupied, alpha occupied, and empty orbital spaces. The diagonal blocks can be written

$$F2 = Acc*Fa + Bcc*Fb$$

$$F1 = Aoo*Fa + Boo*Fb$$

$$F0 = Avv*Fa + Bvv*Fb$$

Some choices for the canonicalization coefficients to define the diagonal blocks are

	Acc	Bcc	Aoo	Boo	Avv	Bvv
Guest and Saunders	1/2	1/2	1/2	1/2	1/2	1/2
Roothaan single matrix	-1/2	3/2	1/2	1/2	3/2	-1/2
Davidson/1988	1/2	1/2	1	0	1	0
Binkley, Pople, Dobosh	1/2	1/2	1	0	0	1
McWeeny and Diercksen	1/3	2/3	1/3	1/3	2/3	1/3
Faegri and Manne	1/2	1/2	1	0	1/2	1/2

Here is some idea of the range of eigenvalues that result from using the various canonicalization schemes in the table above. The system is 6-31G nitrogen atom, all runs give $E = -54.3820511123$ (matching all 10 decimals):

	1s	2s	2p	"3p"	"3s"
Roothaan	-15.5514	-0.5306	-0.1774	+0.7666	+0.8704
McWeeny,Diercksen	-15.6214	-0.8745	-0.1183	+0.8984	+0.9684
Davidson	-15.6355	-0.9432	-0.5657	+0.8457	+0.9292
Guest,Saunders	-15.6355	-0.9432	-0.1774	+0.9248	+0.9880
Binkley,Pople,Dob.	-15.6355	-0.9432	-0.5657	+1.0039	+1.0469
Faegri,Manne	-15.6355	-0.9432	-0.5657	+0.9248	+0.9880
GVB (aka Euler)	-15.6355	-0.9432	-0.2828	+0.9248	+0.9880
"canonical 1"	-15.5933	-0.7370	-0.5657	+0.8457	+0.9292
"canonical 2"	-15.7065	-1.2863	+0.2109	+1.0567	+1.0861

Since the ionization potential (IP) for a 2p electron in Nitrogen is 0.53 Hartree, it is clear that most of the orbital energies above do not approximately predict this IP. Recent work by Boris Plakhutin and co-workers (see the 3 papers in the ROHF references above) leads to two sets of orbitals and eigenvalues, for the prediction of both IP and electron affinities (EA), for various ionization events, starting from a high spin ROHF state:

canonical 1 (to produce high spin final states)

A1 = remove beta e⁻ from filled space,

B1 = remove alpha e⁻ from open shell,

C1 = attach alpha e⁻ to virtual space.

canonical 2 (to produce low spin final states)

A2 = remove alpha e⁻ from filled space,

B2 = attach beta e⁻ to filled space,

C2 = attach beta e⁻ to virtual space.

Correct handling of spin requires the value of the spin S of the initial state in the 2nd canonicalization set. Two different ROHF runs are necessary to get all six EA and IP processes.

Additional discussion about ROHF orbital energies may be found in

K.R.Glaesemann, M.W.Schmidt

J.Phys.Chem.A 114, 8772-8777(2010) [available on-line]
along with a demonstration of the non-uniqueness of ROHF-based perturbation theories.

High-spin ROHF results are automatically obtained if a UHF calculation is constrained so that the occupied beta orbitals lie entirely within the occupied alpha orbital space. Such a constraint is called CUHF, and while it will appear to have different alpha/beta orbitals, and alpha/beta eigenvalues, its spin expectation value will be

$\langle S^2 \rangle = 2S+1$, and its energy will be exactly that of ROHF.

The CUHF method is given by

G.E.Scuseria, T.Tsuchimochi

J.Chem.Phys. 134, 064101/1-14(2011)

CUHF runs as a UHF calculation, solving two distinct SCF equations in the alpha and beta space, so its convergence behavior and its final eigenvalues will differ from that for any of the ROHF canonicalizations. See the section on perturbation theory in this chapter regarding CUHF's perturbation theory, CUMP2.

other open shell SCF cases (GVB)

Genuine GVB-PP "perfect pairing" runs will be discussed later in this section. First, we will consider how to do open shell SCF with the GVB part of the program, with NPAIR=0. This is an extremely powerful open shell SCF program, capable of handling low spin couplings and/or degenerate open shells, with the energy formula

$$E = 2 \sum_i F_i h_{ii} + \sum_i \sum_j \text{ALPHA}_{ij} J_{ij} + \text{BETA}_{ij} K_{ij}$$

Here F_i (meaning subscript i) is a fractional occupancy, and is 1.0 for the filled shell that is usually present below the open shell(s). A d^2 shell has $F_i=0.2$. The h, J, K are the usual one electron, coulomb, and exchange operators. For a few common cases, the F , ALPHA , BETA are internally stored in the program,

one open shell, doublet:

```
$CONTRL SCFTYP=GVB MULT=2 $END
```

```
$SCF NCO=xx NSET0=1 NO(1)=1 $END
```

two open shells, triplet:

```
$CONTRL SCFTYP=GVB MULT=3 $END
```

```
$SCF NCO=xx NSET0=2 NO(1)=1,1 $END
```

two open shells, singlet:

```
$CONTRL SCFTYP=GVB MULT=1 $END
```

```
$SCF NCO=xx NSET0=2 NO(1)=1,1 $END
```

Note that the first two cases duplicate runs which the high spin ROHF module can readily do. The last case is also an ROHF calculation, albeit for a low-spin coupling. One should note that the open shell singlet case is a variational calculation IF AND ONLY IF the two singly degenerate orbitals have different space symmetry.

A publication should refer to any run that has NPAIR=0 as being ROHF type, rather than GVB, since there are no perfect pairs in use. Note, however, that the GVB program can, if you wish, have both open shells and genuine valence bond pairs at the same time! See the following section about the use of GVB's perfect pairs. The use of open shells and pairs is illustrated in one of the GAMESS standard example inputs.

If you would like to do any cases other than those shown above, including many cases with orbital degeneracy, you must derive the coupling coefficients ALPHA and BETA, and input them with the occupancies F in the \$SCF group. Fortunately, many cases can be looked up!

Mariusz Klobukowski of the University of Alberta has shown how to obtain coupling coefficients for the GVB open shell program for many such open shell states. These can be obtained from Appendix A of the book "A General SCF Theory" by Ramon Carbo and Josep M. Riera, Springer-Verlag (1978). The rule is

- (1) $F(i) = 1/2 * \omega(i)$
- (2) $ALPHA(i) = \alpha(i)$
- (3) $BETA(i) = -\beta(i)$,

where ω , α , β are symbols used in these Tables.

The keyword NSET0 gives the number of open shells, and keyword NO gives the degeneracy of each open shell. Thus the 5-S state of carbon (from configuration s1,p3) would enter NSET0=2 and NO(1)=1,3. NCO is an easy keyword: that is the number of filled orbitals. The total number of electrons in an open shell GVB run is

$$NE = 2*NCO + \sum_i 2*F(i)*NO(i)$$

and this value will be checked against your ICHARG input.

- - - -

Specific input for all terms found in the atomic p**N configurations follow. Be sure to enter 14 digits, as these values are part of a double precision energy formula!

Values for the excited terms in the p**N configurations were extracted from C.F.Jackels and E.R.Davidson Int. J. Quantum Chem. 8, 707-714(1974), which explains the concept of averaging over equivalent determinants to enforce a symmetric density matrix, which preserves radial symmetry in the atomic orbitals. The ALPHA and BETA values can also

be extracted from Roothaan's 1960 A and B coefficients by the prescription detailed below.

```
!   p**1   2-P state
$CONTRL SCFTYP=GVB  MULT=2   $END
$SCF     NCO=xx    NSET0=1  NO=3   COUPLE=.TRUE.
          F(1)=   1.0   0.16666666666667
          ALPHA(1)= 2.0   0.33333333333333  0.00000000000000
          BETA(1)= -1.0 -0.16666666666667 -0.00000000000000 $END
```

```
!   p**2   3-P state
$CONTRL SCFTYP=GVB  MULT=3   $END
$SCF     NCO=xx    NSET0=1  NO=3   COUPLE=.TRUE.
          F(1)=   1.0   0.33333333333333
          ALPHA(1)= 2.0   0.66666666666667  0.16666666666667
          BETA(1)= -1.0 -0.33333333333333 -0.16666666666667 $END
```

For the 1-D excited state, change the open-shell parameters to ALPHA(3)=0.1 and BETA(3)= +0.03333333333333

For the 1-S excited state, change the open-shell parameters to ALPHA(3)=0.0 and BETA(3)= +0.33333333333333

```
!   p**3   4-S state
$CONTRL SCFTYP=ROHF  MULT=4   $END
which is equivalent to
$CONTRL SCFTYP=GVB  MULT=4   $END
$SCF     NCO=xx    NSET0=1  NO=3   COUPLE=.TRUE.
          F(1)=   1.0   0.50000000000000
          ALPHA(1)= 2.0   1.00000000000000  0.50000000000000
          BETA(1)= -1.0 -0.50000000000000 -0.50000000000000 $END
```

For 2-D, use ALPHA(3)= 0.4 and BETA(3)= -0.2
for 2-P, use ALPHA(3)= 0.33333333333333 and BETA(3)= 0.0

```
!   p**4   3-P state
$CONTRL SCFTYP=GVB  MULT=3   $END
$SCF     NCO=xx    NSET0=1  NO=3   COUPLE=.TRUE.
          F(1)=   1.0   0.66666666666667
          ALPHA(1)= 2.0   1.33333333333333  0.83333333333333
          BETA(1)= -1.0 -0.66666666666667 -0.50000000000000 $END
```

For 1-D, use ALPHA(3)= 0.76666666666667 and BETA(3)= -0.3
for 1-S, use ALPHA(3)= 0.66666666666667 and BETA(3)= 0.0

```
!   p**5   2-P state
$CONTRL SCFTYP=GVB  MULT=2   $END
$SCF     NCO=xx    NSET0=1  NO=3   COUPLE=.TRUE.
          F(1)=   1.0   0.83333333333333
          ALPHA(1)= 2.0   1.66666666666667  1.33333333333333
```

BETA(1)= -1.0 -0.8333333333333333 -0.666666666666667 \$END

- - - -

Coupling constants for the highest spin state(s) in d**N configurations are taken from "Handbook of Gaussian Basis Sets", R.Poirier, R.Kari, I.G.Csizmadia, Elsevier, Amsterdam, 1985.

```
!      d**1    2-D state
$CONTRL SCFTYP=GVB MULT=2 $END
$SCF     NCO=xx NSET0=1 NO=5 COUPLE=.TRUE.  F(1)=1.0,0.1
          ALPHA(1)= 2.0, 0.20, 0.00
          BETA(1)=-1.0,-0.10, 0.00 $END
```

```
!      d**2    average of 3-F and 3-P states
$CONTRL SCFTYP=GVB MULT=3 $END
$SCF     NCO=xx NSET0=1 NO=5 COUPLE=.TRUE.  F(1)=1.0,0.2
          ALPHA(1)= 2.0, 0.40, 0.05
          BETA(1)=-1.0,-0.20,-0.05 $END
```

Note: "average" means a degeneracy weighted combination of determinants with $S_z=S$: here, 2 electrons in 5 orbitals coupled as a triplet is ten determinants with $S_z=1$. The energy from these parameters is $E=[7 \times E(3-F) + 3 \times E(3-P)]/10$.

```
!      d**3    average of 4-F and 4-P states
$CONTRL SCFTYP=GVB MULT=4 $END
$SCF     NCO=xx NSET0=1 NO=5 COUPLE=.TRUE.  F(1)=1.0,0.3
          ALPHA(1)= 2.0, 0.60, 0.15
          BETA(1)=-1.0,-0.30,-0.15 $END
```

```
!      d**4    5-D state
$CONTRL SCFTYP=GVB MULT=5 $END
$SCF     NCO=xx NSET0=1 NO=5 COUPLE=.TRUE.  F(1)=1.0,0.4
          ALPHA(1)= 2.0, 0.80, 0.30
          BETA(1)=-1.0,-0.40,-0.30 $END
```

```
!      d**5    6-S state
$CONTRL SCFTYP=ROHF MULT=6 $END
```

```
!      d**6    5-D state
$CONTRL SCFTYP=GVB MULT=5 $END
$SCF     NCO=xx NSET0=1 NO=5 COUPLE=.TRUE.  F(1)=1.0,0.6
          ALPHA(1)= 2.0, 1.20, 0.70
          BETA(1)=-1.0,-0.60,-0.50 $END
```

```
!      d**7    average of 4-F and 4-P states
$CONTRL SCFTYP=GVB MULT=4 $END
$SCF     NCO=xx NSET0=1 NO=5 COUPLE=.TRUE.  F(1)=1.0,0.7
```

```
ALPHA(1)= 2.0, 1.40, 0.95
BETA(1)=-1.0,-0.70,-0.55 $END
```

```
!      d**8    average of 3-F and 3-P states
$CONTRL SCFTYP=GVB MULT=3 $END
$SCF    NCO=xx NSET0=1 NO=5 COUPLE=.TRUE.  F(1)=1.0,0.8
ALPHA(1)= 2.0, 1.60, 1.25
beta(1)=-1.0,-0.80,-0.65 $end
```

```
!      d**9    2-D state
$CONTRL SCFTYP=GVB MULT=2 $END
$SCF    NCO=xx NSET0=1 NO=5 COUPLE=.TRUE.  F(1)=1.0,0.9
ALPHA(1)= 2.0, 1.80, 1.60
BETA(1)=-1.0,-0.90,-0.80 $END
```

Note that GAMESS can do a proper calculation on the ground term for the d**2, d**3, d**7, d**8 configurations only by means of state averaged MCSCF. For d**8, use

```
$contrl scftyp=mcscf mult=3 $end
$det    group=c1 ncore=xx nact=5 nels=8
        nstate=10 wstate(1)=1,1,1,1,1,1,1,0,0,0 $end
```

to correctly average the 7 lowest roots (3-F) with no weight given to the highest three roots (3-P). Although this is done with the SCFTYP=MCSCF program, this is still a SCF calculation since only the orbitals, but not any CI coefficients, are optimized.

Dirk Andrae in Berlin has provided a great many examples of using the MCSCF program to do terms found in the p**n, d**n, and even f**n configurations:

<http://userpage.fu-berlin.de/~dandrae/openshell/opensls/opensls.html>

Type that web reference on just one line, of course.

Open shell cases such as s**1,d**n are probably most easily tackled with the state-averaged MCSCF program. The ORMAS CI code may be convenient in fixing the number of electrons found in each open shell.

If you are a true aficionado of atomic calculations, including open shell f configurations, look for Russ Pitzer's version of the famous Roothaan/Bagus ATOM-SCF program

R.M.Pitzer, Comput.Phys.Commun. 170, 239-264(2005)

- - - -

The literature contains an alternate energy formula, involving the so-called Roothaan A and B parameters:

$$E = 2 \sum_i f_i h_{ii} + \sum_i \sum_j f_i f_j [2A_{ij} J_{ij} - B_{ij} K_{ij}]$$

Comparing this to the GVB program's energy formula above, we immediately see that its fractional occupancy f is the same as GVB's F , and that

$$\text{ALPHA-}ij = +2 * f_i * f_j * A_{ij}$$

$$\text{BETA-}ij = -1 * f_i * f_j * B_{ij}$$

Let c =the closed shell orbitals, and o =a single open shell. Tables such as Roothaan's 1960 paper give only the A_{oo} and B_{oo} values, since $A_{cc} = B_{cc} = A_{co} = B_{co} = 1.0$. It is easy to convert such A, B values to all $\text{ALPHA-}cc, -co, -oo$ and $\text{BETA-}cc, -co, -oo$ values. Degenerate open shells in linear molecules follow immediately from Roothaan's 1960 table:

```
!      2-Pi from configuration pi**1:
$contrl scftyp=gvb mult=2 $end
$scf    nco=xx nseto=1 no=2 couple=.true. f(1)=1.0,0.25
        alpha(1)= 2.0, 0.50, 0.0
        beta(1)=-1.0,-0.25, 0.0 $end

!      3-Sigma-minus from configuration pi**2:
$contrl scftyp=gvb mult=3 $end
$scf    nco=xx nseto=1 no=2 couple=.true. f(1)=1.0,0.50
        alpha(1)= 2.00, 1.00, 0.50
        beta(1)=-1.00,-0.50,-0.50 $end

!      1-Delta from configuration pi**2:
$contrl scftyp=gvb mult=1 $end
$scf    nco=xx nseto=1 no=2 couple=.true. f(1)=1.0,0.50
        alpha(1)= 2.00, 1.00, 0.00
        beta(1)=-1.00,-0.50,+0.50 $end

!      1-Sigma-plus from configuration pi**2:
$contrl scftyp=gvb mult=1 $end
$scf    nco=xx nseto=1 no=2 couple=.true. f(1)=1.0,0.50
        alpha(1)= 2.00, 1.00, 0.25
        beta(1)=-1.00,-0.50, 0.00 $end

!      2-Pi from configuration pi**3:
$contrl scftyp=gvb mult=2 $end
$scf    nco=xx nseto=1 no=2 couple=.true. f(1)=1.0,0.75
        alpha(1)= 2.00, 1.50, 1.00
        beta(1)=-1.00,-0.75,-0.50 $end
```

The same inputs apply to delta^{**N} configurations: only the names of the energy terms (states) change!

It is possible to do every term arising from the s^{**1}, p^{**N} configurations. Roothaan-style A,B parameters are as follows:

Acc = Bcc = 1.0 (as always)
 Aco = Bco = 1.0, for o=s and for o=p (as always)
 Ass = Bss = 0.0, since there is no other s e- to repel
 Asp = 1.0
 Bsp = $-3K101 + 1$

App and Bpp: take from the parent term in config p^{**N} and then apply the rule turning A,B into ALPHA,BETA, for $F_c=1$, $F_s=0.5$, $F_p=N/6$. K101 should be taken from Roothaan and Bagus in Methods Comput. Phys. 2, 49(1963).

- - - -

It may be useful to obtain a "configuration average", especially whenever a specific term energy cannot be computed. Boris Plakhutin in Novosibirsk has kindly provided a recipe for the Roothaan A,B values that give the average energy of configuration 0^{**N} , meaning N electrons in some open shell 0. Let D be the degeneracy of the open shell 0:

atoms: 0=p, d, f... has D= 3, 5, 7...
 non-linear molecules: 0=e, t, g, h has D= 2, 3, 4, 5.
 The fractional occupation number $F_o = N/(2*D)$. The open shell Roothaan couplings are $A_{oo} = B_{oo} = (N-1)/(N-F)$.

For example, the case d^{**2} has $F_o=1/5$ so $A_{oo}=B_{oo}=5/9$. Using the formulae to generate all ALPHA and BETA causes the GVB program to optimize the following spin-and-space weighted average of all terms appearing in d^{**2} :

$$E_{\text{avg}} = [21xE(3-F) + 9xE(3-P) + 9xE(1-G) + 5xE(1-D) + E(1-S)]/45$$

by this input

```
$contrl scftyp=gvb mult=3 $end
$scf nco=xx nseto=1 no=5 couple=.true.
      f(1)= 1.0, 0.2
      alpha(1)= 2.00, 0.40, 0.04444444444444444444
      beta(1)=-1.00,-0.20,-0.02222222222222222222 $end
```

Note that here, as in all other runs with the GVB module, the spin in \$CONTRL is irrelevant: the energy that is optimized is determined by F, ALPHA, and BETA. In this last case, these data implicitly include both triplet and singlet determinants in the energy averaging.

Boris Plakhutin also provided a formula in case you want to average over a particular spin S within an open shell 0^{**N} with some degeneracy D implying a fractional occupancy of F. The result is

$$A_{-oo} = \frac{N^2(N-1) + FxN(N-4) + 4FxS(S+1)}{N(N^2-4F^2)}$$

and

$$B_{-oo} = \frac{4F(N-1) + N(N-4) + 4S(S+1)}{(N^2-4F^2)}$$

For the d^2 configuration, the space-and-spin degeneracy weighted average of the 3-F and 3-P terms has $A_{oo}=5/8$ and $B_{oo}=10/8$, while the average of the 1-G, 1-D, and 1-S terms has $A_{oo}=5/12$ and $B_{oo}=-10/12$. These optimize orbitals for the average

$$E_{\text{-triplet}} = [21xE(3-F) + 9xE(3-P)]/30$$

which is not quite the same as the space degeneracy average shown above, and for the average

$$E_{\text{-singlet}} = [9xE(1-G) + 5xE(1-D) + E(1-S)]/15$$

The literature reference for the overall and the spin-specific configurational averages is

B.N.Plakhutin, J. Mathematical Chem. 22, 203-233(1997)

true GVB perfect pairing runs

True GVB runs are obtained by choosing NPAIR nonzero. If you wish to have some open shell electrons in addition to the geminal pairs, you may add the pairs to the end of any of the GVB coupling cases shown above. The GVB module assumes that you have reordered your MOs into the order: NCO double occupied orbitals, NSET0 sets of open shell orbitals, and NPAIR sets of geminals (with NORDER=1 in the \$GUESS group).

Each geminal consists of two orbitals and contains two singlet coupled electrons (perfect pairing). The first MO of a geminal is probably heavily occupied (such as a bonding MO u), and the second is probably weakly occupied (such as an antibonding, correlating orbital v). If you have more than one pair, you must be careful that the initial MOs are ordered $u_1, v_1, u_2, v_2, \dots$, which is *NOT* the same order that RHF starting orbitals will be found in. Use NORDER=1 to get the correct order.

These pair wavefunctions are actually a limited form of MCSCF. GVB runs are much faster than MCSCF runs, because the natural orbital u, v form of the wavefunction permits a Fock operator based optimization. However, convergence of the GVB run is by no means assured. The same care in selecting the correlating orbitals that you would apply to

an MCSCF run must also be used for GVB runs. In particular, look at the orbital expansions when choosing the starting orbitals, and check them again after the run converges.

GVB runs will be carried out entirely in orthonormal natural u,v form, with strong orthogonality enforced on the geminals. Orthogonal orbitals will pervade your thinking in both initial orbital selection, and the entire orbital optimization phase (the CICOEF values give the weights of the u,v orbitals in each geminal). However, once the calculation is converged, the program will generate and print the nonorthogonal, generalized valence bond orbitals. These GVB orbitals are an entirely equivalent way of presenting the wavefunction, but are generated only after the fact.

Convergence of true GVB runs is by no means as certain as convergence of RHF, UHF, ROHF, or GVB with NPAIR=0. You can assist convergence by doing a preliminary RHF or ROHF calculation, and use these orbitals for GUESS=MOREAD. Few, if any, GVB runs with NPAIR non-zero will converge without using GUESS=MOREAD. Generation of MVOs during the preliminary SCF can also be advantageous. In fact, all the advice outlined for MCSCF computations below is germane, for GVB-PP is a type of MCSCF computation.

The total number of electrons in the GVB wavefunction is given by the following formula:

$$NE = 2*NCO + \sum_i 2*F(i)*NO(i) + 2*NPAIR$$

The charge is obtained by subtracting the total number of protons given in \$DATA. The multiplicity is implicit in the choice of alpha and beta constants. Note that ICHARG and MULT must be given correctly in \$CONTRL anyway, as the number of electrons from this formula is double checked against the ICHARG value.

the special case of TCSCF

The wavefunction with NSET0=0 and NPAIR=1 is called GVB-PP(1) by Goddard, two configuration SCF (TCSCF) by Schaefer or Davidson, and CAS-SCF with two electrons in two orbitals by others. Note that this is just semantics, as these are identical. This is a very important type of wavefunction, as TCSCF is the minimum acceptable treatment

for singlet biradicals. The TCSCF wavefunction can be obtained with SCFTYP=MCSCF, but it is usually much faster to use the Fock based SCFTYP=GVB. Because of its importance, the TCSCF function (together with open shells, if desired) permits analytic hessian computation.

a caution about symmetry

Caution! Some exotic calculations with the GVB program do not permit the use of symmetry. The symmetry algorithm in GAMESS was "derived assuming that the electronic charge density transforms according to the completely symmetric representation of the point group", Dupuis/King, JCP, 68, 3998(1978). This may not be true for certain open shell cases, and in fact during GVB runs, it may not be true for closed shell singlet cases!

First, consider the following correct input for the singlet-delta state of NH:

```
$CONTRL SCFTYP=GVB NOSYM=1 $END
```

```
$SCF      NCO=3 NSET0=2 NO(1)=1,1 $END
```

for the x^2y state, or for the x^2-y^2 state,

```
$CONTRL SCFTYP=GVB NOSYM=1 $END
```

```
$SCF      NCO=3 NPAIR=1 CICOEF(1)=0.707,-0.707 $END
```

Neither gives correct results, unless you enter NOSYM=1. The electronic term symbol is degenerate, a good tip off that symmetry cannot be used. However, some degenerate states can still use symmetry, because they use coupling constants averaged over all degenerate states within a single term, as is done in EXAM15 and EXAM16. Here the "state averaged SCF" leads to a charge density which is symmetric, and these runs can exploit symmetry.

Secondly, since GVB runs exploit symmetry for each of the "shells", or type of orbitals, some calculations on totally symmetric states may not be able to use symmetry. An example is CO or N₂, using a three pair GVB to treat the sigma and pi bonds. Individual configurations such as $(\sigma)^2, (\pi-x)^2, (\pi-y)^2$ do not have symmetric charge densities since neither the pi nor pi* level is completely filled. Correct answers for the sigma-plus ground states result only if you input NOSYM=1.

Problems of the type mentioned should not arise if the point group is Abelian, but will be fairly common in linear molecules. Since GAMESS cannot detect that the GVB electronic state is not totally symmetric (or averaged to at least have a totally symmetric density), it is left up

to you to decide when to input NOSYM=1. If you have any question about the use of symmetry, try it both ways. If you get the same energy, both ways, it remains valid to use symmetry to speed up your run.

And beware! Brain dead computations, such as RHF on singlet O₂, which actually is a half filled degenerate shell, violate the symmetry assumptions, and also violate nature. Use of partially filled degenerate shells always leads to very wild oscillations in the RHF energy, which is how the program tries to tell you to think first, and compute second. Configurations such as pi**2, e**1, or f2u**4 can be treated, but require GVB wavefunctions and F, ALPHA, BETA values from the sources mentioned.

How to do MCSCF (and CI) calculations

Multi-configuration self consistent field (MCSCF) wavefunctions are the most general SCF possible. MCSCF allows for a natural description of chemical processes involving the separation of electrons (bond breaking, electronic excitation, etc), which are often not well represented using the single configuration SCF methods.

MCSCF wavefunctions, as the name implies, contain more than one configuration, each of which is multiplied by a configuration interaction (CI) coefficient, optimized to determine its weight. In addition, the shapes of the orbitals used to form each of the configurations are optimized, just as in a simpler SCF, to self consistency.

Typically every different chemical problem requires that an MCSCF wavefunction be designed to treat it, on a case by case basis, by choosing an "active space". For example, one may be interested in describing the reactivity of a particular functional group, instead of elsewhere in the molecule. So, the active electrons and active orbitals will be those that are "active" on that functional group. Orbitals elsewhere in the molecule just remain doubly occupied, as for RHF. This means some attention must be paid to orbitals in order to obtain the desired results.

Procedures for the selection of configurations (which amounts to choosing the number of active electrons and active orbitals), for the two mathematical optimizations just mentioned, ways to interpret the resulting MCSCF wavefunction, and treatments for dynamical electron correlation of MCSCF wavefunctions are the focus of a review article:

"The Construction and Interpretation of MCSCF wavefunctions"

M.W.Schmidt and M.S.Gordon,
Annu.Rev.Phys.Chem. 49,233-266(1998)

One section of this article is devoted to the problem of designing the correct active space to treat your problem. Additional reading is listed at the end of this section. A much newer and very valuable review article is

"Multiconfiguration self-consistent field and multireference configuration interaction methods and applications"

P.G.Szalay, T.Muller, G.Gidofalvi, H.Lischka, R.Shepard
Chem.Rev. 112, 108-181(2012)

These pages describe a powerful and mature MCSCF program, allowing computation of the MCSCF energy, nuclear gradient, and nuclear hessian for pure states. Practical procedures for generating starting orbitals are available. Localized orbital analysis of the final active orbitals is provided. State-averaged energies and their analytic gradients can be obtained. Minimum energy crossing points and conical intersections between surfaces may be found, see RUNTYP=MEX and CONINT. Non-adiabatic coupling matrix elements (NACME) can be computed, see RUNTYP=NACME. Diabatic potential energy surfaces can be obtained, see DIABAT in \$MCSCF. If desired, spin-orbit couplings or transition dipole moments can be found, see elsewhere in this chapter. Efficient perturbative treatments of the dynamical correlation energy for all electrons, whether in active and filled orbitals, are provided. Of course, parallel computation has been enabled.

The most efficient technique implemented in GAMESS for finding the dynamic correlation energy of MCSCF is second order perturbation theory, in the variant known as MCQDPT (known as MRMP for one state). MCQDPT is discussed in a different section of this chapter.

The use of CI, probably in the form of second order CI, will be described below, en passant, during discussion of the input defining the configurations for MCSCF. Selection of a CI following any type of SCF (except UHF) is made with CITYP in the \$CONTRL group, and masterminded by \$CIINP.

MCSCF implementation

With the exception of the QUAD converger, the MCSCF program is of the type termed "unfolded two-step" by Roos. This means the orbital and CI coefficient optimizations are separated. The latter are obtained in a conventional CI diagonalization, while the former are optimized by a separate orbital improvement step.

Each MCSCF iteration (except for the JACOBI and QUAD convergers) consists of the following steps:

- 1) transformation of AO integrals to the current MO basis,
- 2) generation of the Hamiltonian matrix and optimization of the CI coefficients by a Davidson diagonalization,
- 3) generation of the first and second order density matrix,
- 4) improvement of the molecular orbitals.

During the first iteration at the first geometry, you will receive verbose output from each of these steps, but each subsequent iteration produce only a single summary line.

The CI problem in steps two and three has four options for the many electron basis, namely ALDET, ORMAS, or GENCI using determinants, or GUGA using CSFs. This choice is made with the keyword CISTEP in \$MCSCF. Much more will be said below about the differences between determinants and CSFs. The word "configuration" will be used throughout this section to refer to either determinants or CSFs, when a generic term is needed for the many-electron functions. Most people use CSF and configuration interchangeably, so please note the distinction made here.

The orbital update in step four has five options, namely FOCAS, SOSCF, FULLNR, JACOBI, and QUAD, listed here in roughly the order of their increasing mathematical sophistication, convergence characteristics, and of course, their computer resource requirements. Again, these are chosen by keywords in the \$MCSCF group. More will be said just below about the relative merits of these.

Depending on the converger chosen, the program will select the appropriate kind of integral transformation at step one. There's seldom need to try to fine tune this, but note that the \$TRANS group allows you to choose an AO integral direct transformation, with the DIRTRF flag.

The type of CI and the type of orbital converger are to some extent "mix and match". This is particularly true for the two full CI programs, GUGA or ALDET, where either produces exactly the same CI density matrices. Here is a chart of the ways to combine CI and orbital optimizers:

orbital converger	parallel run's transformation memory	CI computation via CISTEP=			
		GUGA	ALDET	GENCI	ORMAS
FOCAS	replicated	ok	ok	silly	silly
SOSCF	replicated	ok	ok	ok	ok
FULLNR	distributed	ok	ok	ok	ok
QUAD	serial	ok	xx	xx	xx
JACOBI	serial	ok	ok	ok	ok

"xx" means QUAD converger is coded only for CISTEP=GUGA.

"silly" means that this converger ignores active-active rotations, so these runs are likely to be divergent, or perhaps converge to a false solution.

"serial" means this can only run sequentially at present.

The next two sections provide more information on the two mathematical optimizations, first how the orbital shape is refined, and then the determination of CI coefficients.

orbital updates

There are presently five orbital improvement options, namely FOCAS, SOSCF, FULLNR, JACOBI, and QUAD. All but the JACOBI update run in parallel. Each converger is discussed briefly below, in order of increasing robustness. The most commonly used convergers are SOSCF and FULLNR.

The input to control the orbital update step is the \$MCSCF group, where you can pick the convergence procedure. Most of the input in this group is rather specialized, but note in particular MAXIT and ACURCY, which control the convergence behavior.

FOCAS is a first order, complete active space MCSCF optimization procedure. It is based on a novel approach due to Meier and Staemmler, using very fast but numerous microiterations to improve the convergence of what is intrinsically a first order method. Since FOCAS requires only one virtual orbital in the integral transformation to compute the Lagrangian (whose asymmetry is the orbital gradient, and must fall below ACURCY at convergence), the total MCSCF job may take less time than a second order method, even though it may require many more iterations to converge. The use of microiterations is crucial to FOCAS' ability to converge. It is important to take a great deal of care choosing the starting orbitals.

SOSCF is a method built upon the FOCAS code, which seeks to combine the speed of FOCAS with approximate second order convergence properties. Thus SOSCF is an approximate Newton-Raphson, based on a diagonal guess at the orbital hessian, and in fact has much in common with the SOSCF option in \$SCF. Its time requirements per iteration are like FOCAS, with a convergence rate better than FOCAS but not as good as true second order. Storage of only the diagonal of the orbital hessian allows the SOSCF method to be used with much larger basis sets than exact second order methods. Because SOSCF usually requires the least CPU time, disk space, and memory needs, it is the default. Good convergence by the SOSCF method requires that you prepare starting orbitals carefully, and read in all MOs in \$VEC, as providing canonicalized virtual orbitals increases the diagonal dominance of the orbital hessian. Parallel

computations are possible with SOSCF, but only to a modest number of nodes.

FULLNR means a full Newton-Raphson orbital improvement step is taken, using the exact orbital hessian. FULLNR is a robust convergence method, and normally takes the fewest iterations to converge. Computing the exact orbital hessian requires two virtual orbital indices be included in the integral transformation, making this step quite time consuming, and of course memory for storage of the orbital hessian must be available. Because both the transformation and orbital improvement steps of FULLNR are time consuming, FULLNR is not the default. You may want to try FULLNR when convergence is difficult, assuming you have already tried preparing good starting orbitals by the hints below.

The serial FULLNR code uses the augmented hessian matrix approach to solve the Newton-Raphson equations. There are two suboptions for computation of the orbital hessian: DM2 is faster, but takes more memory than TEI. The parallel implementation of FULLNR avoids explicit storage of the orbital hessian, by recomputing the product of the hessian times orbital rotation vector during the subiterations solving the Newton-Raphson problem. The partial integral transformation used to set up the FULLNR converger has been changed to use distributed memory, and will scale like the MP2 energy/gradient programs, to many nodes. Parallel FULLNR requires large MEMORY only for the CI step (if the active space is big), but always requires a large MEMDDI. The parallel FULLNR program is essentially diskless, apart from storage of converged CI vectors.

The JACOBI method uses a series of 2 by 2 orbital rotations by an angle predicted to lower the energy. This should essentially ensure convergence after sweeping through all possible orbital pairs enough times. The procedure was created to converge selected (general) determinant MCSCF functions, but of course it can be used will full lists as well in difficult cases. The JACOBI calculation will consist of a full four index transformation over all MOs before the iterations begin. Each iteration consists of

1. a small 4 index transformation over active orbitals
2. optimization of the CI vector
3. generation of the 1e- and 2e- density matrices
4. sweeps over Jacobi rotations, using MO integrals in memory to generate each rotation, with a subsequent update after each pair is rotated.
5. when sufficient energy lowering has been achieved,

begin a new iteration.

This procedure never generates the orbital Lagrangian! Unfortunately this means that at present it is not possible to compute nuclear gradients. Due to lack of a Lagrangian, ACURCY is of course irrelevant, so the convergence test is on ENGTOL.

QUAD uses a fully quadratic, or second order approach and is thus the most powerful MCSCF converger. The QUAD code is programmed only for CISTEP=GUGA. QUAD runs begin with normal unfolded FULLNR iterations, until the orbitals approach convergence sufficiently. QUAD then begins the simultaneous optimization of CI coefficients and orbitals, and convergence should be obtained in 3-4 additional MCSCF iterations. The QUAD method requires building the full electronic hessian, including orbital/orbital, orbital/CI, and CI/CI blocks, which is a rather big matrix. In principle, this is the most robust method available, but it is limited to perhaps 50-100 CSFs only, because it is a memory hog. QUAD may be helpful in converging excited electronic states, but note that you may not use state averaging with QUAD. In practice, QUAD has not received very much use compared to the unfolded convergers.

CI coefficient optimization

Determinants or configuration state functions (CSFs) may be used to form the many electron basis set. It is necessary to explain these in a bit of detail so that you can understand the advantages of each.

A determinant is a simple object: a product of spin orbitals with a given S_z quantum number, that is, the number of alpha spins and number of beta spins are a constant. Matrix elements involving determinants are correspondingly simple, but unfortunately determinants are not necessarily eigenfunctions of the S^2 operator.

To expand on this point, consider the four familiar 2e-functions which satisfy the Pauli principle. Here u , v are space orbitals, and a , b are the alpha and beta spin functions. As you know, the singlet and triplets are:

$$S1 = (uv + vu)/\sqrt{2} * (ab - ba)/\sqrt{2}$$

$$T1 = (uv - vu)/\sqrt{2} * aa$$

$$T2 = (uv - vu)/\sqrt{2} * (ab + ba)/\sqrt{2}$$

$$T3 = (uv - vu)/\sqrt{2} * bb$$

It is a simple matter to multiply out $S1$ and $T2$, and to expand the two determinants which have $S_z=0$,

$$D1 = |ua \ vb|$$

$$D2 = |va \ ub|$$

This reveals that

$$S1 = (D1+D2)/\text{sqrt}(2) \quad \text{or} \quad D1 = (S1 + T2)/\text{sqrt}(2)$$

$$T2 = (D1-D2)/\text{sqrt}(2) \quad \quad \quad D2 = (S1 - T2)/\text{sqrt}(2)$$

Thus, one must take a linear combination of determinants in order to have a wavefunction with the desired total spin.

There are two important points to note:

- a) A two by two Hamiltonian matrix over D1 and D2 has eigenfunctions with -different- spins, $S=0$ and $S=1$.
- b) use of all determinants with $S_z=0$ does allow for the construction of spin adapted states. $D1+D2$, or $D1-D2$, are -not- spin contaminated.

By itself, a determinant such as D1 is said to be "spin contaminated", being a fifty-fifty admixture of singlet and triplet. (It is curious that calculations with just one such determinant are often called "singlet UHF", when this is half triplet!). Of course, some determinants are spin adapted all by themselves, for example the spin adapted functions T1 and T3 above are single determinants, as are the closed shells

$$S2 = (uu) * (ab - ba)/\text{sqrt}(2).$$

$$S3 = (vv) * (ab - ba)/\text{sqrt}(2).$$

It is possible to perform a triplet calculation, with no singlet states present, by choosing determinants with $S_z=1$ such as T1, since then no state with $S_z=0$ exists in the determinant basis set (as is required when $S=0$). To summarize, the eigenfunctions of a Hamiltonian formed by determinants with any particular S_z will be spin states with $S=S_z$, $S=S_z+1$, $S=S_z+2$, ... but will not contain any S values smaller than S_z .

CSFs are an antisymmetrized combination of a space orbital product, and a spin adapted linear combination of simple alpha-beta products. Namely, the following CSF

$$C1 = A (uv) * (ab-ba)/\text{sqrt}(2)$$

which has a singlet spin function is identical to S1 above if you write out what the antisymmetrizer A does, and the CSFs

$$C2 = A (uv) * aa$$

$$C3 = A (uv - vu)/\text{sqrt}(2) * (ab + ba)/\text{sqrt}(2)$$

$$C4 = A (uv) * bb$$

equal T1-T3. Since the three triplet CSFs have the same energy, GAMESS works with the simpler form C2. Singlet and triplet computations using CSFs are done in separate runs, because when spin-orbit coupling is not considered, the Hamiltonian is block diagonal in a CSF basis. Technical information about the CSFs is that they use Yamanouchi-Kotani spin couplings, and matrix elements are obtained using a GUGA, or graphical unitary group approach.

Determinant and CSF are both primarily used for MCSCF wavefunctions, but can be used in CI (see CITYP in \$CONTRL). Other comparisons between the determinant and CSF implementations, as they exist in GAMESS today, are

	determinants	CSFs
parallel execution	yes	yes
direct CI	yes	no
use Abelian group symmetry	yes	yes
state average mixed spins	yes	no
first order density	yes	yes
state averaged densities	yes	yes
analytic nuclear hessian	yes	no
can form CI Lagrangian	no	yes

In nearly every circumstance the determinant CI will run faster than GUGA, so it is the default. Here are timings for N electrons in N orbitals, no symmetry used:

N in N	ALDET	GENCI	--- GUGA ---	
8	1	1	1	0
10	8	38	19	33
12	228	3122	534	2209
14	7985	--	15377	130855

The reason there are two numbers under GUGA is that the first is for writing the loops (Hamiltonian data) to disk, and the second is for the actual diagonalization. Note that the GUGA Hamiltonian time alone is greater than the entire ALDET computation! The ALDET program does not store anything on disk, and so runs at CPU/wall ratios of 1. The quality of the initial guess of the CI eigenvector in the various determinant codes is much better than in the GUGA CSF code, so the chances of converging to an incorrect excited state root is much less. Finally, determinant input is generally easier. No wonder determinants are now the default configurations!

Two of the determinant CI programs, namely ALDET or ORMAS (but not GENCI) have been changed to use replicated memory parallelism, with modest scaling. The GUGA program is parallel in its solving, but not its Hamiltonian generation, and as already noted, its H formation takes more time than the entire determinant CI (translation: use CISTEP=ALDET, not CISTEP=GUGA). The GENCI program will run as a serial bottleneck (no speedup) in parallel runs.

The next two sections describe in detail the input for specification of the configurations, either determinants or CSFs.

determinant CI

Three determinant CI codes are provided for MCSCF, one for full CI spaces (ALDET), another named the Occupation Restricted Multiple Active Spaces (ORMAS), and finally there is a program for arbitrary (selected) determinant lists (GENCI). For straight CI, but not MCSCF, there is a fourth program, the full second order CI (CITYP=FSOCI), whose purpose is MR-CISD.

ALDET is a full CI within the chosen active space. It is possible to go up to 16 electrons in 16 orbitals, if your computer has a lot of memory. ALDET is the only CISTEP for which analytic nuclear hessian is possible, and it is also the most scalable CI code (using replicated memory to store CI vectors). A sample input for ALDET is
\$DET STSYM=B1 NSTATE=3 NCORE=xx NELS=8 NACT=6 \$END
Keywords in this group actually relate to all determinant programs, and are described below.

The \$DET input group is basic to all determinant CI codes. Keywords GROUP and STSYM specify the desired spatial symmetry of the determinants. Most runs need give only the orbital and electron counts: NCORE, NACT, and NELS. The number of electrons is $2*NCORE+NELS$, and will be checked against the charge implied by ICHARG. The MULT given in \$CONTRL is used to determine the desired S_z value, by extracting S from $MULT=2S+1$, then by default $S_z=S$. If you wish to include lower spin multiplicities, which will increase the CPU time of the run, but will let you know what the energies of such states are, just input a smaller value for SZ . The states whose orbitals will be MCSCF optimized will be those having the requested MULT value, unless you choose otherwise with the PURES flag.

The remaining parameters in the \$DET group give extra control over the diagonalization process. Most are not given in normal circumstances, except NSTATE, which you may need to adjust to produce enough roots of the desired MULT value. The only important keyword which has not been discussed is the WSTATE array, giving the weights for each state in forming the first and second order density matrix elements, which drive the orbital update methods. Note that analytic gradients are available only when the WSTATE array is a unit vector, corresponding to a pure state, such as $WSTATE(1)=0,1,0$ which permits gradients of the first excited state to be computed. When used for state averaged MCSCF, WSTATE is normalized to a unit sum, thus input of

WSTATE(1)=1,1,1 really means a weight of 0.33333... for the each of the states being averaged.

ORMAS (Occupation Restricted Multiple Active Space) is a program designed to limit the size of the full CI problem, and may be useful when the number of active orbitals is 10 or higher. By dividing your total active space into multiple subspaces, and specifying a range of electrons to occupy each subspace, most of the full CI's effect can be included. ORMAS generates a full CI within each orbital subspace, taking the product of each small full CI to generate the determinant list.

Here are some ideas on how to use ORMAS, which is a very flexible CI program:

- a) single reference, arbitrary excitation level CI-X, from a closed shell reference:

```
$det   ncore=y nact=z nels=10      (y+z=entire basis)
$ormas  nspace=2 mstart(1)=y+1,y+6 mine(1)=10-x,0
                                     maxe(1)=10,x
```

This excites the 5 doubly occupied orbitals, to the desired excitation level of X.

An open shell example of CI-SD from ...22111 might be

```
$contrl mult=4
$det   ncore=y nact=z nels=7      (y+z=entire basis)
$ormas  nspace=3 mstart(1)=y+1,y+3,y+6
                                     mine(1)=2,1,0
                                     maxe(1)=4,5,2
```

No more than 2e- are allowed to be promoted from the doubly occupied or singly occupied spaces, and no more than 2 are allowed to enter the singly occupied or empty spaces.

- b) simple product of active spaces

For example, consider furan, with two active subspaces. Keeping the 5 true core and the 4 CH bonds in the core space, the sigma subspace might contain 5 ring sigma, one oxygen lone pair, and 5 ring sigma antibonds, with a total of 12 e-. The pi active space contains 5 pi orbitals and 6 e-:

```
$det   ncore=9 nact=16 nels=18
$ormas  nspace=2 mstart(1)=10,21 mine(1)=12,6
                                     maxe(1)=12,6
```

Having the minimum and maximum electron counts the same is what makes this the simple product of two separate active spaces. In other words, this is similar to the QCAS procedure of Nakano and Hirao,

but ORMAS limits only the total electron counts, not separately the numbers of alpha and beta e-, in other words all spin couplings are used.

c) flexible occupancy between active subspaces

Imagine that you are interested in excited states of formaldehyde, some of which will have Rydberg character, dominated by single excitations into diffuse orbitals. H₂CO's valence states arise from 3 orbitals, the CO pi and pi* and one oxygen lone pair. Placing the O sp lone pair and three sigma bonds into the filled space, and centering diffuse s,p,d shells on the carbon:

```
$det      ncore=6 nact=12 nels=4
$ormas    nspace=2 mstart(1)=7,10 mine(1)=3,0
                                     maxe(1)=4,1
```

This is a 4e-, 3 orbital n,pi,pi* space to describe valence states, and excites one electron into the 9 diffuse orbitals to describe Rydberg states. It is many fewer determinants than a 4e- in 12 orbital FCI.

d) RAS-like CI

The previous example is reminiscent of Roos' RAS-SCF. In fact ORMAS can do RAS-SCF, which is three spaces: the lowest space is allowed to excite only a few electrons, a middle space that is the rest, and a top space into which only a few electrons can be excited. Suppose there are 10 e-, 10 orbitals, that the bottom and top spaces involve 3 orbitals, and that a "few" means specifically 2 e-:

```
$det      ncore=20 nact=10 nels=10 $end
$ormas    nspace=3 mstart(1)=21,24,28 mine(1)=4,2,0
                                     maxe(1)=6,6,2
```

However, ORMAS can use more than 3 orbital subspaces.

e) first or second order CI.

Consider C₂H₄, with a 4 orbital active space of CC sigma, pi, pi*, and sigma*. In order to correlate the four valence CH orbitals by double excitations, an MCSCF based on \$DET, followed by SOCI based on \$CIDET and \$ORMAS, is:

```
$contrl  scfity=mcscf cityp=ormas
$mcscf   cistep=aldet
$det      ncore=6 nact=4 nels=4
$cidet    ncore=2 nact=y nels=12 (y=rest of basis)
$ormas    nspace=3 mstart(1)=3,7,11 mine(1)=6,2,0
                                     maxe(1)=8,6,2
```

which permits singles and doubles out of the CH and CC spaces, into the CC and external spaces.

ORMAS is a full CI (or several full CI's) within each orbital subspace. However, ORMAS does not generate all excitation levels between spaces (just those implied by the minimum and maximum electron counts you give). This means ORMAS MCSCF runs must optimize active-active rotations between the subspaces, and therefore you should expect better convergence from FULLNR than SOSCF.

ORMAS is sure to require orbital reordering. For the furan example just mentioned, there is no reason to expect that the RHF occupied orbitals will not have the filled sigma and pi orbitals intermingled. You must use the NORDER and IORDER keywords in \$GUESS to carefully partition starting orbitals into sigma and pi subspaces.

The selected (general) determinant list is used if CISTEP=GENCI, and the list is controlled by two input groups. The first is \$GEN, which is identical to \$DET except for inclusion of an additional keyword GLIST=INPUT. This reads the determinants (as space products) from an additional input group \$GCILST. Completely arbitrary choices for the space products may be made, but peculiar lists may lead to poor MCSCF convergence. The FOCAS converger should not be used, as it assumes full CI spaces.

If you are doing straight CI calculations, the required input for each determinant CITYP is:

ALDET needs \$CIDET

ORMAS needs \$CIDET and \$ORMAS

GENCI needs \$CIDET and \$CIGEN and probably \$GCILST

FSOCI needs \$CIDET and \$SODET

In other words, \$CIDET replaces \$DET, and \$CIGEN replaces \$GEN, but the keywords in the groups mean the same thing. The reason for different names is to allow CI calculations to follow MCSCF in the same run, without clashing input group names.

CSF CI

The GUGA-based CSF package was originally a set of different programs, so its input is spread over several input groups. The CSFs are specified by a \$CIDRT group in the case of CITYP=GUGA, and by a \$DRT group for MCSCF wavefunctions. Thus it is possible to perform an MCSCF defined by a \$DRT input (or perhaps using \$DET during the MCSCF), and follow this with a second order CI defined by a \$CIDRT group, in the same run.

The remaining input groups used by the GUGA CSFs are \$CISORT, \$GUGEM, \$GUGDIA, and \$GUGDM2 for MCSCF runs, with the latter two being the most important, and in the case of CI computations, \$GUGDM and possibly \$LAGRAN groups are relevant. Perhaps the most interesting variables outside the \$DRT/\$CIDRT group are NSTATE in \$GUGDIA to include excited states in the CI computation, IROOT in \$GUGDM to select the CI state for properties, and WSTATE in \$GUGDM2 to control which state's orbitals are optimized, and possible state-averaging.

The \$DRT and \$CIDRT groups are almost the same, with the only difference being orbitals restricted to double occupancy are called MCC in \$DRT, and FZC in \$CIDRT. Therefore the rest of this section refers only to "\$DRT".

The CSFs are specified by giving a reference CSF, together with a maximum degree of electron excitation from that single CSF. The MOs in the reference CSF are filled in the order MCC or FZC first, followed by DOC, AOS, BOS, ALP, VAL, and EXT (the Aufbau principle). AOS, BOS, and ALP are singly occupied MOs. ALP means a high spin alpha coupling, while AOS/BOS are an alpha/beta coupling to an open shell singlet. This requires the value NAOS=NBOS, and their MOs alternate. An example is

NFZC=1 NDOC=2 NAOS=2 NBOS=2 NALP=1 NVAL=3
which gives the reference CSF

FZC,DOC,DOC,AOS,BOS,AOS,BOS,ALP,VAL,VAL,VAL
This is a doublet state with five unpaired electrons. VAL orbitals are unoccupied only in the reference CSF, they will become occupied as the other CSFs are generated. This is done by giving an excitation level, either explicitly by the IEXCIT variable, or implicitly by the FORS, FOCI, or SOCI flags. One of these four keywords must be chosen, and during MCSCF runs, this is usually FORS.

Consider another simpler example, for an MCSCF run,
NMCC=3 NDOC=3 NVAL=2
which gives the reference CSF

MCC,MCC,MCC,DOC,DOC,DOC,VAL,VAL
having six electrons in five active orbitals. MCSCF calculations are usually of the Full Optimized Reaction Space (FORS) type. Some workers refer to FORS as CASSCF, complete active space SCF. These are the same, but the keyword is spelled FORS in GAMESS. In the present instance, choosing FORS=.TRUE. gives an excitation level of 4, as the 6 valence electrons have only 4 holes available for excitation. MCSCF runs typically have only a small

number of VAL orbitals. It is common to summarize this example as "six electrons in five orbitals".

The next example is a first or second order multi-reference CI wavefunction, where

```
NFZC=3 NDOC=3 NVAL=2 NEXT=-1
```

leads to the reference CSF

```
FZC,FZC,FZC,DOC,DOC,DOC,VAL,VAL,EXT,EXT,...
```

FOCI or SOCI is chosen by selecting the appropriate flag, the correct excitation level is automatically generated. Note that the -1 for NEXT causes all remaining MOs to be included in the external orbital space. One way of viewing FOCI and SOCI wavefunctions is as all singles, or all singles and doubles, from the entire MCSCF wavefunction as a reference. An equivalent way of saying this is that all CSFs with N electrons (in this case N=6) distributed in the valence orbitals in all ways (that is the FORS MCSCF wavefunction) make up the reference wavefunction. To this, FOCI adds all CSFs with N-1 electrons in active and 1 electron in external orbitals. SOCI adds all CSFs with N-2 electrons in active orbitals and 2 in external orbitals. SOCI is often prohibitively large, but is also a very accurate wavefunction. SOCI can also be performed with determinants, as CITYP=FSOCI, or CITYP=ORMAS. The latter may be the most efficient way to generate SOCI energies. For larger molecules, where SOCI is impractical, the most effective way to recover dynamic correlation energy is the multireference perturbation method.

Sometimes people use the CI package for ordinary single reference CI calculations, such as

```
NFZC=3 NDOC=5 NVAL=34
```

which means the reference RHF wavefunction is

```
FZC FZC FZC DOC DOC DOC VAL VAL ... VAL
```

and in this case NVAL is a large number conveying the total number of -virtual- orbitals into which electrons are excited. The excitation level would be given as IEXCIT=2, perhaps, to perform a SD-CI. All excitations smaller than the value of IEXCIT are automatically included in the CI. Note that NVAL's spelling was chosen to make the most sense for MCSCF calculations, and so it is a bit of a misnomer here.

Before going on, there is a quirk related to single reference CI that should be mentioned. Whenever the single reference contains unpaired electrons, such as

```
NFZC=3 NDOC=4 NALP=2 NVAL=33
```

some "extra" CSFs will be generated. The reference here can be abbreviated

2222 11 000 000 000 000 000 000 000 000 000 000 000
 Supposing IEXCIT=2, the following CSF
 2200 22 000 011 000 000 000 000 000 000 000 000 000
 will be generated and used in the CI. Most people would prefer to think of this as a quadruple excitation from the reference, but acting solely on the reasoning that no more than two electrons went into previously vacant NVAL orbitals, the GUGA CSF package decides it is a double. So, an open shell SD-CI calculation with GAMESS will not give the same result as other programs, although the result for any such calculation with these "extras" is correctly computed. Note that if you also select the INTACT option, the extra space products are eliminated, but that some of the spin couplings for the truly IEXCIT'd space products are also eliminated. Note that this kind of problem does not arise if you use ORMAS!

As was discussed above, the CSFs are automatically spin-symmetry adapted, with S implicit in the reference CSF. The spin quantum number you appear to be requesting in \$DRT (basically, $S = \text{NALP}/2$) will be checked against the value of MULT in \$CONTRL. The total number of electrons, $2*\text{NMCC}(\text{or NFZC}) + 2*\text{NDOC} + \text{NAOS} + \text{NBOS} + \text{NALP}$ will be checked against the input given for ICHARG.

The CSF package is also able to exploit spatial symmetry, which like the spin and charge, is implicitly determined by the choice of the reference CSF. The keyword GROUP in \$DRT governs the use of spatial symmetry.

The CSF program works with Abelian point groups, which are D2h and any of its subgroups. However, \$DRT allows the input of some (but not all) higher point groups. For non-Abelian groups, the program automatically assigns the orbitals to an irrep in the highest possible Abelian subgroup. For the other non-Abelian groups, you must at present select GROUP=C1. Note that when you are computing a Hessian matrix, many of the displaced geometries are asymmetric, hence you must choose C1 in \$DRT (however, be sure to use the highest symmetry possible in \$DATA!).

The symmetry of the reference CSF given in your \$DRT is one way to determine the symmetry of the CSFs which are generated. As an example, consider a molecule with Cs symmetry, and these two reference CSFs

```
...MCC...DOC DOC VAL VAL
...MCC...DOC AOS BOS VAL
```

Suppose that the 2nd and 3rd active MOs have symmetries a' and a". Both of these generate singlet wavefunctions, with

4 electrons in 4 active orbitals, but the former constructs 1-A' CSFs, while the latter generates 1-A'' CSFs. However, if the 2nd and 3rd orbitals have the same symmetry type, an identical list of CSFs is generated. The alternative is to enter the spatial symmetry with the STSYM keyword.

In cases with high point group symmetry, it may be possible to generate correct state degeneracies only by using no symmetry (GROUP=C1) when generating CSFs. As an example, consider the 2- π ground state of NO. If you use GROUP=C4V, which will be mapped into its highest Abelian subgroup C2v, the two components of the π state will be seen as belonging to different irreps, B1 and B2. The only way to ensure that both sets of CSFs are generated is to enforce no symmetry at all, so that CSFs for both components of the π level are generated. This permits state averaging (WSTATE(1)=0.5,0.5) to preserve cylindrical symmetry. It is however perfectly feasible to use C4v or D4h symmetry in \$DRT when treating sigma states.

The use of spatial symmetry decreases the number of CSFs, and thus the size of the Hamiltonian that must be computed. In molecules with high symmetry, this may lead to faster run times with the GUGA CSF code, compared to the determinant code.

starting orbitals

The first step is to partition the orbital space into core, active, and external sets, in a manner which is sensible for your chemical problem. This is a bit of an art, and the user is referred to the references quoted at the end of this section. Having decided what MCSCF to perform, you now must consider the more pedantic problem of what orbitals to begin the MCSCF calculation with.

You should always start an MCSCF run with orbitals from some other run, by means of GUESS=MOREAD. Do not expect to be able to use HUCKEL! At the start of a MCSCF problem, use orbitals from some appropriate converged SCF run. A realistic example of an MCSCF calculation is GAMESS examples 8 and 9. Once you get an MCSCF to converge, you can and should use these MCSCF MOs at other nearby geometries (MOREAD will apply an appropriate Schmidt orthogonalization).

Starting from SCF orbitals can take a little bit of care. Most of the time (but not always) the orbitals you

want to correlate will be the highest occupied orbitals in the SCF. Fairly often, however, the correlating orbitals you wish to use will not be the lowest unoccupied virtuals of the SCF. You will soon become familiar with `NORDER=1` in `$GUESS`, as reordering is needed in 50% or more cases.

The occupied and especially the virtual canonical SCF MOs are often spread out over regions of the molecule other than "where the action is". Orbitals which remedy this can be generated by two additional options at almost no CPU cost.

The best way to improve upon the SCF canonical virtual orbitals as starting MOs is to generate valence virtual orbitals (VVOs), after any RHF, ROHF, or GVB calculation. These are constructed by projection of internally stored atomic core and valence orbitals onto the SCF external orbitals, so by construction, the resulting VVOs are valence in character. See `VVOS` in `$SCF`.

An alternative choice, usually not as good as VVOs, are the modified virtual orbitals. MVOs are obtained by diagonalizing the Fock operator of a very positive ion, within the virtual orbital space only. As implemented in GAMESS, MVOs can be obtained at the end of any RHF, ROHF, or GVB run by setting `MVOQ` in `$SCF` nonzero, at the cost of a single SCF cycle. Typically, we use `MVOQ=+6`. Generating MVOs does not change any of the occupied SCF orbitals of the original neutral, but gives more valence-like LUMOs.

Another way to improve SCF starting orbitals is by a partial localization of the occupied orbitals. Typically MCSCF active orbitals are concentrated in the part of the molecule where bonds are breaking, etc. Canonical SCF MOs are normally more spread out. By choosing `LOCAL=BOYS` along with `SYMLOC=.TRUE.` in `$LOCAL`, you can get orbitals which are localized, but still retain orbital symmetry to help speed the MCSCF along. In groups with an inversion center, a `SYMLOC` Boys localization does not change the orbitals, but you can instead use `LOCAL=POP`. `LOCAL=RUEDNBRG` may also be used, but requires more machine resources, and the other two localizations are normally good enough for starting orbital purposes. Localization tends to order the orbitals fairly randomly, so be prepared to inspect them, and then reorder them appropriately.

In case VVOS are generated in the same run as the localization, the localization is also applied within the valence virtual space. The effect is to localize occupied orbitals into lone pairs and bond pairs, and in the VVOs

space, to find localized antibond pairs. When you choose the bonds and antibonds of chemical interest as the starting orbitals for the active space, convergence to the desired MCSCF wavefunction should follow.

If you take the time to design your active space sensibly, select appropriate starting orbitals from the occupied and VVO unoccupied spaces, possibly by localizing these two subspaces, and carefully inspect your converged results, you will be able to carry out MCSCF computations correctly.

Convergence of MCSCF is by no means guaranteed. Poor convergence can invariably be traced back to either a poor initial selection of orbitals, or poor design of the active space. The best advice is, before you even start:

"Look at the orbitals."

"Then, look at the orbitals again".

Later, if you have any trouble:

"Look at the orbitals some more".

Few people are able to see the orbital shapes in the LCAO matrix in a log file, and so need a visualization program. In particular, you should download a copy of MacMolPlt from

<http://code.google.com/p/wxmacmolplt>

This runs on all popular desktop operating systems (Apple, Linux, and Windows), making it easy to see your final MCSCF orbital shapes.

Even if you don't have any trouble, look at the orbitals to see if they converged to what you expected, and have reasonable occupation numbers. It is particularly useful to check the oriented localized MCSCF orbitals (see the discussion of this in the section on localized orbitals in this section for more information). MCSCF is by no means the sort of "black box" that RHF is these days, so please look very carefully at your final results.

miscellaneous hints

It is very helpful to execute a EXETYP=CHECK run before doing any MCSCF or CI run. The CHECK run will tell you the total number of configurations and check the charge and multiplicity and electronic state symmetry, based on your input. The CHECK run also lets the program feel out the memory that will be required to actually do the run. Thus the CHECK run can potentially prevent costly mistakes, or tell you when a calculation is prohibitively large.

A very common MCSCF wavefunction has 2 electrons in 2 active MOs. This is the simplest possible wavefunction describing a singlet diradical. While this function can be obtained in an MCSCF run (using NACT=2 NELS=2 or NDOC=1 NVAL=1), it can be obtained much faster by use of the GVB code, with one GVB pair. This GVB-PP(1) wavefunction is also known in the literature as two configuration SCF, or TCSCF. The two configurations of this GVB are equivalent to the three configurations used in this MCSCF, as orbital optimization in natural form (configurations 20 and 02) causes the coefficient of the 11 configuration to vanish.

If you are using a large active space (say, 12 or more orbitals), the main bottleneck in the MCSCF calculation is the formation and diagonalization of the Hamiltonian, not the integral transformation and orbital updates. Of course, since determinants are much faster than CSFs, and do not use large disk files, you should use determinants for large active spaces. In this case, you would be wise to switch to FULLNR, which will minimize the total number of iterations, and thus the number of CI calculations. Note that by selecting ITERMX=5 in \$DET or \$GEN, you can avoid fully converging the CI during each MCSCF iteration, saving a bit of time. Since each iteration's CI calculation starts with the previous iteration's result, the CI vectors will become fully converged during the MCSCF cycles. The total run time may decrease, although a few additional MCSCF iterations may be required. For small active spaces, where the CI step takes trivial time, you should use a bigger ITERMX to ensure fully converged CI states are generated every iteration.

If you choose to use ORMAS, a general determinant CI, or if you select an CSF excitation level IEXCIT smaller than that needed to generate the FORS space, you must use the SOSCF, JACOBI, or FULLNR method as these can optimize active-active rotations. Be sure to set FORS=.FALSE. in \$MCSCF when for non-full CI cases, or else very poor convergence will result. Actually, the convergence for incomplete active spaces is likely to be poorer than for full active spaces, anyway.

A good way to check the active space is to localize the orbitals, to see if they resemble the atomic orbitals which you imagined formed the bonds, antibonds, and lone pairs in the active space. The ORIENT keyword in \$LOCAL will print a density matrix analysis, showing active electron bonding and antibonding patterns (see reference 18/19 below).

- - - - -

The MCSCF technology in GAMESS is the result of some considerable programming effort: The FOCAS, serial FULLNR, and QUAD convergers were adapted from Michel Dupuis' HONDO program. The SOSCF converger was written by Galina Chaban, the parallel FULLNR converger is due to Graham Fletcher, and the JACOBI converger is due to Joe Ivanic. The GUGA CI programs were written by Bernie Brooks and others, while all determinant CI codes (ALDET, GENCI, ORMAS, and FSOCI) stem from Joe Ivanic. Analytic nuclear Hessians were programmed by Tim Dudley. The CSF-based multireference perturbation program was written by Haruyuki Nakano, with a determinant implementation provided by Joe Ivanic. Shiro Koseki and Dmitri Fedorov are responsible for the spin-orbit coupling and transition moment codes. The expertise of Klaus Ruedenberg in MCSCF wavefunctions has been the inspiration for many of these developments!

MCSCF references

There are several review articles about MCSCF listed below. Of these, the first two are a nice overview of the subject, the final 3 are more technical.

1. "The Construction and Interpretation of MCSCF wavefunctions"
M.W.Schmidt and M.S.Gordon,
Ann.Rev.Phys.Chem. 49,233-266(1998)
- 2a. "The Multiconfiguration SCF Method"
B.O.Roos, in "Methods in Computational Molecular Physics", edited by G.H.F.Diercksen and S.Wilson
D.Reidel Publishing, Dordrecht, Netherlands,
1983, pp 161-187.
- 2b. "The Multiconfiguration SCF Method"
B.O.Roos, in "Lecture Notes in Quantum Chemistry",
edited by B.O.Roos, Lecture Notes in Chemistry v58,
Springer-Verlag, Berlin, 1994, pp 177-254.
3. "Optimization and Characterization of a MCSCF State"
J.Olsen, D.L.Yeager, P.Jorgensen
Adv.Chem.Phys. 54, 1-176(1983).
4. "Matrix Formulated Direct MCSCF and Multiconfiguration Reference CI Methods"
H.-J.Werner, Adv.Chem.Phys. 69, 1-62(1987).
5. "The MCSCF Method"
R.Shepard, Adv.Chem.Phys. 69, 63-200(1987).

There is an entire section on the choice of active spaces in Reference 1. As this is a matter of great

importance, here are two alternate presentations of the design of active spaces:

6. "The CASSCF Method and its Application in Electronic Structure Calculations"
B.O.Roos, in "Advances in Chemical Physics", vol.69,
edited by K.P.Lawley, Wiley Interscience, New York,
1987, pp 339-445.
7. "Are Atoms Intrinsic to Molecular Electronic Wavefunctions?"
K.Ruedenberg, M.W.Schmidt, M.M.Gilbert, S.T.Elbert
Chem.Phys. 71, 41-49, 51-64, 65-78 (1982).

Two papers germane to the FOCAS implementation are

8. "An Efficient first-order CASSCF method based on the renormalized Fock-operator technique."
U.Meier, V.Staemmler Theor.Chim.Acta 76, 95-111(1989)
9. "Modern tools for including electron correlation in electronic structure studies"
M.Dupuis, S.Chen, A.Marquez, in "Relativistic and Electron Correlation Effects in Molecules and Solids", edited by G.L.Malli, Plenum, NY 1994

The paper germane to the the SOSCF converger is

10. "Approximate second order method for orbital optimization of SCF and MCSCF wavefunctions"
G.Chaban, M.W.Schmidt, M.S.Gordon
Theor.Chem.Acc. 97: 88-95(1997)

Two papers germane to the FULLNR converger, and two discussing implementation details are

11. "General second order MCSCF theory: A Density Matrix Directed Algorithm"
B.H.Lengsfeld, III, J.Chem.Phys. 73,382-390(1980).
12. "The use of the Augmented Matrix in MCSCF Theory"
D.R.Yarkony, Chem.Phys.Lett. 77,634-635(1981).
13. M.Dupuis, P.Mougenot, J.D.Watts, in "Modern Techniques in Theoretical Chemistry", E.Clementi, editor, ESCOM, Leiden, 1989, chapter 7.
14. "A parallel multi-configuration self-consistent field algorithm"
G.D.Fletcher, Mol.Phys. 105, 2971-2976(2007)

The paper describing the JACOBI converger is

15. "A MCSCF method for ground and excited states based on

full optimizations of successive Jacobi rotations"
J.Ivanic, K.Ruedenberg
J.Comput.Chem. 24, 1250-1262(2003)

For determinant CI codes, see

16. "Identification of deadwood in configuration spaces through general direct configuration interaction"
J.Ivanic, K.Ruedenberg
Theoret.Chem.Acc. 106, 339-351(2001)
17. "Direct configuration interaction and multi-configurational self-consistent-field method for multiple active spaces with variable occupancies. Part I.Method Part II.Applications"
J.Ivanic
J.Chem.Phys. 119, 9364-9376 and 9377-9385(2003)

For CSFs, see

18. "GUGA approach to the electron correlation problem"
B.R.Brooks, H.F.Schaefer
J.Chem.Phys. 70, 5092-5106(1979)

Orientation of localized MCSCF active orbitals for bonding analysis:

19. J.Ivanic, G.M.Atchity, K.Ruedenberg
Theoret.Chem.Acc. 120, 281-294(2008)
20. J.Ivanic, K.Ruedenberg
Theoret.Chem.Acc. 120, 295-305(2008)

Second Order Perturbation Theory

The perturbation theory techniques available in GAMESS expand to the second order energy correction only, but permit use of nearly any zeroth order SCF wavefunction. Since MP2 theory for systems well described by the chosen zeroth order reference recovers about 80-85% of the dynamical correlation energy (assuming the use of large basis sets), MP2 is often a computationally effective theory. For higher accuracy, you can instead choose the more time consuming coupled cluster theory. When using MPLEVEL=2, it is important to ensure that your system is well described at zeroth order by your choice of SCFTYP.

The input for second order perturbation calculations based on SCFTYP=RHF, UHF, or ROHF is found in \$MP2, while for SCFTYP=MCSCF, see \$MRMP.

By default, frozen core MP2 calculations are performed.

RHF and UHF reference MP2

These methods are well defined, due to the uniqueness of the Fock matrix definitions. These methods are also well understood, so there is little need to say more, except to point out an overview article on RHF or UHF MP2 gradients:

C.M.Aikens, S.P.Webb, R.L.Bell, G.D.Fletcher,
M.W.Schmidt, M.S.Gordon

Theoret.Chem.Acc. 110, 233-253(2003)

The distributed memory parallel MP2 gradient program is described in

G.D.Fletcher, M.W.Schmidt, M.S.Gordon
Adv.Chem.Phys. 110, 267-294(1999)

and that for UMP2 in

C.M.Aikens, M.S.Gordon
J.Phys.Chem.A 108, 3103-3110(2004)

One point which may not be commonly appreciated is that the density matrix for the first order wavefunction for the RHF and UHF case, which is generated during gradient runs or if properties are requested in the \$MP2 group, is of the type known as "response density", which differs from the more usual "expectation value density". The eigenvalues of the response density matrix (which are the occupation numbers of the MP2 natural orbitals) can therefore be greater than 2 for frozen core orbitals, or even negative

values for the highest 'virtual' orbitals. The sum is of course exactly the total number of electrons. We have seen values outside the range 0-2 in several cases when the single configuration HF wavefunction is not an appropriate description of the system, and thus these occupancies may serve as a guide to the wisdom of using a HF reference:

M.S.Gordon, M.W.Schmidt, G.M.Chaban, K.R.Glaesemann,
W.J.Stevens, C.Gonzalez J.Chem.Phys. 110,4199-4207(1999)

high spin ROHF reference MP2

There are a number of open shell perturbation theories described in the literature. It is important to note that these methods give different results for the second order energy correction, reflecting ambiguities in the selection of the zeroth order Hamiltonian and in defining the ROHF Fock matrices. See

K.R.Glaesemann, M.W.Schmidt

J.Phys.Chem.A 114, 8772-8777(2010)

for a figure showing 4 different ROHF-based perturbation theory potentials, which are highly parallel, but have different total energy values.

Two of the perturbation theories mentioned below, RMP and ZAPT, are available in GAMESS using SCFTYP=ROHF (see OSPT in \$MP2). Nuclear gradients can be obtained for ZAPT. The OPT1 results can be generated using MPLEVEL=2 with SCFTYP=MCSCF, using an active space where every orbital is singly occupied with the highest MULT possible (which is single-determinant).

One theory is known as RMP, which it should be pointed out, is entirely equivalent to the ROHF-MBPT2 method and to the CUMP2 method. The perturbation theory is as UHF-like as possible, and can be chosen in GAMESS by selection of OSPT=RMP. The second order energy is defined by

1. P.J.Knowles, J.S.Andrews, R.D.Amos, N.C.Handy,
J.A.Pople Chem.Phys.Lett. 186, 130-136(1991)
2. W.J.Lauderdale, J.F.Stanton, J.Gauss, J.D.Watts,
R.J.Bartlett Chem.Phys.Lett. 187, 21-28(1991).

The submission dates are in inverse order of publication dates, and -both- papers should be cited when using this method. Here we will refer to the method as RMP in keeping with much of the literature. The RMP method diagonalizes the alpha and beta Fock matrices separately, so their occupied-occupied and virtual-virtual blocks are canonicalized. This generates two distinct orbital sets, whose double excitation contributions are processed by the

usual UHF MP2 program, but an additional energy term from single excitations is required. The recent CUHF method converges a UHF calculation directly to these semi-canonical orbitals, rather than generating them after ROHF converges. The CUMP2 perturbation theory is thus identical to methods described above:

3. G.E.Scuseria, T.Tsuchimochi

J.Chem.Phys. 134, 064101/1-14(2011)

CUMP2's input is SCFTYP=UHF, MPLEVEL=2, CUHF=.TRUE. rather than SCFTYP=ROHF, MPLEVEL=2, OSPT=RMP.

RMP's use of different orbitals for different spins adds to the CPU time required for integral transformations, of course. just like UMP2. RMP is invariant under all of the orbital transformations for which the ROHF itself is invariant. Unlike UMP2, the second order RMP energy does not suffer from spin contamination, since the reference ROHF wave-function has no spin contamination. The RMP wavefunction, however, is spin contaminated at 1st and higher order, and therefore the 3rd and higher order RMP energies are spin contaminated. Other workers have extended the RMP theory to gradients and Hessians at second order, and to fourth order in the energy,

3. W.J.Lauderdale, J.F.Stanton, J.Gauss, J.D.Watts,
R.J.Bartlett J.Chem.Phys. 97, 6606-6620(1992)

4. J.Gauss, J.F.Stanton, R.J.Bartlett
J.Chem.Phys. 97, 7825-7828(1992)

5. D.J.Tozer, J.S.Andrews, R.D.Amos, N.C.Handy
Chem.Phys.Lett. 199, 229-236(1992)

6. D.J.Tozer, N.C.Handy, R.D.Amos, J.A.Pople, R.H.Nobes,
Y.Xie, H.F.Schaefer Mol.Phys. 79, 777-793(1993)

We deliberately omit references to the ROMP precursor of the RMP formalism. RMP gradients are not available.

The Z-averaged perturbation theory (ZAPT) formalism for ROHF perturbation theory is the preferred implementation of open shell spin-restricted perturbation theory (OSPT=ZAPT in \$MP2). The ZAPT theory has only a single set of orbitals in the MO transformation, and therefore runs in a time similar to the RHF perturbation code. The second order energy is free of spin-contamination, but some spin-contamination enters into the first order wavefunction (and hence properties). This should be much less contamination than for OSPT=RMP. For these reasons, OSPT=ZAPT is the default open shell method.

References for ZAPT are

7. T.J.Lee, D.Jayatilaka Chem.Phys.Lett. 201, 1-10(1993)

8. T.J.Lee, A.P.Rendell, K.G.Dyall, D.Jayatilaka

J.Chem.Phys. 100, 7400-7409(1994)

The formulae for the seven terms in the ZAPT energy are clearly summarized in the paper

9. I.M.B.Nielsen, E.T.Seidl

J.Comput.Chem. 16, 1301-1313(1995)

The ZAPT gradient equations are found in

10. G.D.Fletcher, M.S.Gordon, R.L.Bell

Theoret.Chem.Acc. 107, 57-70(2002)

11. C.M.Aikens, G.D.Fletcher, M.W.Schmidt, M.S.Gordon

J.Chem.Phys. 124, 014107/1-14(2006)

We would like to thank Tim Lee for his gracious assistance in the implementation of the ZAPT energy.

There are a number of other open shell theories, with names such as HC, OPT1, OPT2, and IOPT. The literature for these is

12. I.Hubac, P.Carsky Phys.Rev.A 22, 2392-2399(1980)

13. C.Murray, E.R.Davidson

Chem.Phys.Lett. 187,451-454(1991)

14. C.Murray, E.R.Davidson

Int.J.Quantum Chem. 43, 755-768(1992)

15. P.M.Kozłowski, E.R.Davidson

Chem.Phys.Lett. 226, 440-446(1994)

16. C.W.Murray, N.C.Handy

J.Chem.Phys. 97, 6509-6516(1992)

17. T.D.Crawford, H.F.Schaefer, T.J.Lee

J.Chem.Phys. 105, 1060-1069(1996)

The latter two of these give comparisons of the various high spin methods, and the numerical results in ref. 17 are the basis for the conventional wisdom that restricted open shell theory is better convergent with order of the perturbation level than unrestricted theory. Paper 8 has some numerical comparisons of spin-restricted theories as well. We are aware of one paper on low-spin coupled open shell SCF perturbation theory

18. J.S.Andrews, C.W.Murray, N.C.Handy

Chem.Phys.Lett. 201, 458-464(1993)

but this is not implemented in GAMESS. See the MCSCF reference perturbation code for this case.

GVB based MP2

This is not implemented in GAMESS. Note that the MCSCF perturbation program discussed below should be able to develop the perturbation corrections to open shell singlets, by using a \$DRT input such as

NMCC=N/2-1 NDOC=0 NAOS=1 NBOS=1 NVAL=0

which generates a single CSF if the two open shells have different symmetry, or for a one pair GVB function

NMCC=N/2-1 NDOC=1 NVAL=1

which generates a 3 CSF function entirely equivalent to the two configuration TCSCF, a.k.a GVB-PP(1). For the record, note that if we attempt a triplet state with the MCSCF program,

NMCC=N/2-1 NDOC=0 NALP=2 NVAL=0

we get a result equivalent to the OPT1 open shell method described above, not the RMP or ZAPT result. It is possible to generate the orbitals with a simpler SCF computation than the MCSCF \$DRT examples just given, and read them into the MCSCF based MP2 program described below, by RDVECS=.TRUE..

MCSCF reference perturbation theory

Just as for the open shell case, there are several ways to define a multireference perturbation theory. The most noteworthy are the CASPT2 method of Roos' group, the MRMP2 method of Hirao, the closely related MCQDPT2 method of Nakano, and the MROPTn methods of Davidson. Although the total energies of each method are different, energy differences should be rather similar. In particular, the MRMP/MCQDPT method implemented in GAMESS gives results for the singlet-triplet splitting of methylene in close agreement to CASPT2, MRMP2(Fav), and MROPT1, and differs by 2 Kcal/mole from MRMP2(Fhs), and the MROPT2 to MROPT4 methods.

The MCQDPT method implemented in GAMESS is a multistate perturbation theory due to Nakano. If applied to 1 state, it is the same as the MRMP model of Hirao. When applied to more than one state, it is of the philosophy "perturb first, diagonalize second". This means that perturbations are made to both the diagonal and off-diagonal elements to give an effective Hamiltonian, whose dimension equals the number of states being treated. The effective Hamiltonian is diagonalized to give the second order state energies. Diagonalization after inclusion of the off-diagonal perturbation ensures that avoided crossings of states of the same symmetry are treated correctly. Such an avoided crossing is found in the LiF molecule, as shown in the first of the two papers on the MCQDPT method:

H.Nakano, J.Chem.Phys. 99, 7983-7992(1993)

H.Nakano, Chem.Phys.Lett. 207, 372-378(1993)

The closely related single state "diagonalize, then perturb" MRMP model is discussed by

K.Hirao, Chem.Phys.Lett. 190, 374-380(1992)
K.Hirao, Chem.Phys.Lett. 196, 397-403(1992)
K.Hirao, Int.J.Quant.Chem. S26, 517-526(1992)
K.Hirao, Chem.Phys.Lett. 201, 59-66(1993)

Computation of reference weights and energy contributions is illustrated by

H.Nakano, K.Nakayama, K.Hirao, M.Dupuis
J.Chem.Phys. 106, 4912-4917(1997)

T.Hashimoto, H.Nakano, K.Hirao
J.Mol.Struct.(THEOCHEM) 451, 25-33(1998)

Single state MCQDPT computations are very similar to MRMP computations. A beginning set of references to the other multireference methods used includes:

P.M.Kozlowski, E.R.Davidson

J.Chem.Phys. 100, 3672-3682(1994)

K.G.Dyall J.Chem.Phys. 102, 4909-4918(1995)

B.O.Roos, K.Andersson, M.K.Fulscher, P.-A.Malmqvist,
L.Serrano-Andres, K.Pierloot, M.Merchan

Adv.Chem.Phys. 93, 219-331(1996).

and a review article is available comparing these methods, E.R.Davidson, A.A.Jarzecki in "Recent Advances in Multi-reference Methods" K.Hirao, Ed. World Scientific, 1999, pp 31-63.

The CSF (GUGA-based) MRMP/MCQDPT code was written by Haruyuki Nakano, and was interfaced to GAMESS by him in the summer of 1996. This program makes extensive use of disk files during its specialized transformations and the perturbation steps. Its efficiency is improved if you can add extra physical memory to reduce the number of file reads. In practice we have used this program up to about 12 active orbitals, and with very large disks, to about 500 AOs. In 2005, Joe Ivanic programmed a determinant based MRMP/MCQDPT program. This uses the normal integral transformation routines already present in GAMESS, and direct CI technology to avoid disk I/O. The determinant program is able to handle larger active spaces than the CSF program, and has already been used for cases with 16 electrons in 16 orbitals, and basis sets up to 500 AOs.

When proper care is taken with numerical cutoffs, such as CI vector convergence and the generator cutoff in the CSF code, both programs produce identical results. Both are enabled for parallel execution. The more mature CSF program has several interesting options not found in the determinant program: perturbative treatment of spin-orbit coupling, energy denominators which are a band-aid for the horribly named "intruder states", and the ability to find the weight of the MCSCF reference in the 1st order

wavefunction. Neither program produces a density matrix for property evaluation, nor are analytic gradients programmed.

Second order perturbation corrections are also available for ORMAS-type references, due to a program by Luke Roskop. This is accessed by MRPT=DETMPT, defining the reference by CISTEP=ORMAS in \$MCSCF, with a \$ORMAS group. In its present form, this program is limited to cases with equal numbers of alpha and beta spins (i.e. singlets, triplets, ... with SZ=0, but not doublets, quartets...). See L.Roskop, M.S.Gordon J.Chem.Phys. 135, 044101/1-11(2012)

Finally, note the existence of the MRMP=GMCPT program by Haruyuki Nakano, for other non-CAS references.

- - - - -

We end with an input example to illustrate open shell reference and multi-reference perturbation computations on the ground state of NH₂ radical:

```
! 2nd order perturbation test on NH2, following
! T.J.Lee, A.P.Rendell, K.G.Dyall, D.Jayatilaka
! J.Chem.Phys. 100, 7400-7409(1994), Table III.
! State is 2-B-1, 69 AOs, either 1 or 49 CSFs.
!
! For 1 CSF reference,
!   E(ROHF) = -55.5836109825
!   E(ZAPT) = -55.7763947115
!   [E(ZAPT) = -55.7763947289 at lit's ZAPT geom]
!   E(RMP) = -55.7772299958
!   E(OPT1) = -55.7830422945
!   [E(OPT1) = -55.7830437413 at lit's OPT1 geom]
!
! For 49 CSF full valence MCSCF reference,
!   CSFs: E(MRMP2) = -55.7857440268
!   dets: E(MRMP2) = -55.7857440267
!
! $contrl scftyp=mcscf mplevl=2 runtyp=energy mult=2 $end
! $system mwords=1 memddi=1 $end
! $guess guess=moread norb=69 $end
! $mcscf fullnr=.true. $end
!
! Next set of lines carry out a MRMP computation,
! after a preliminary MCSCF orbital optimization.
!
```



```

!      using determinants
$det      stsym=B1 ncore=1 nact=6 nels=7 $end

!      using CSFs, for the very same calculation.
--- $mcscf  cistep=guga $end
--- $drt    group=c2v stsym=B1 fors=.t.
---        nmcc=1 ndoc=3 nalp=1 nval=2 $end
--- $mrmp   mrpt=mcqdpt $end
--- $mcqdpt stsym=B1 nmofzc=1 nmodoc=0 nmoact=6 $end

!      Next lines carry out a single reference OPT1.
--- $det    stsym=B1 ncore=4 nact=1 nels=1 $end
--- $mrmp   mrpt=mcqdpt rdvecs=.true. $end
--- $mcqdpt nmofzc=1 nmodoc=3 nmoact=1 stsym=B1 $end

!      Next lines are single reference RMP and/or ZAPT
--- $contrl scftyp=rohlf $end
--- $mp2     ospt=rmp $end

```

\$data

2-B-1 state...TZ2Pf basis, RMP geom. of Lee, et al.

Cnv 2

Nitrogen 7.0

S 6

1	13520.0	0.000760
2	1999.0	0.006076
3	440.0	0.032847
4	120.9	0.132396
5	38.47	0.393261
6	13.46	0.546339

S 2

1	13.46	0.252036
2	4.993	0.779385

S 1 ; 1 1.569 1.0

S 1 ; 1 0.5800 1.0

S 1 ; 1 0.1923 1.0

P 3

1	35.91	0.040319
2	8.480	0.243602
3	2.706	0.805968

P 1 ; 1 0.9921 1.0

P 1 ; 1 0.3727 1.0

P 1 ; 1 0.1346 1.0

D 1 ; 1 1.654 1.0

D 1 ; 1 0.469 1.0

F 1 ; 1 1.093 1.0

Hydrogen 1.0 0.0 0.7993787 0.6359684

```
S 3      ! note that this is unscaled
  1 33.64  0.025374
  2  5.058 0.189684
  3  1.147 0.852933
S 1 ; 1 0.3211 1.0
S 1 ; 1 0.1013 1.0
P 1 ; 1 1.407 1.0
P 1 ; 1 0.388 1.0
D 1 ; 1 1.057 1.0

$end

OPT1 geom: H 1.0  0.0 0.7998834 0.6369401
RMP  geom: H 1.0  0.0 0.7993787 0.6359684
ZAPT geom: H 1.0  0.0 0.7994114 0.6357666

E(ROHF)= -55.5836109825, E(NUC)= 7.5835449477, 9 ITERS
$VEC
...omitted...
$END
```

Coupled-Cluster Theory

The single-reference coupled-cluster (CC) theory, employing the exponential wave function ansatz

$$|\Psi_0\rangle = \exp(T) |\Phi\rangle = \exp(T_1+T_2+\dots) |\Phi\rangle,$$

where T_1 , T_2 , etc. are the singly excited (1-particle-1-hole), doubly excited (2-particle-2-hole), etc. components of the cluster operator T and $|\Phi\rangle$ is the single-determinantal reference state (e.g., the Hartree-Fock determinant), is widely recognized as one of the most accurate methods for describing ground electronic states of atoms and molecules. CC approaches provide the best compromise between relatively low computer costs and high accuracy. They are particularly effective in accounting for the dynamical correlation effects. For example, the CCSD(T) approach, which is a $N_o^{**2} * N_u^{**4}$ (or N^{**6}) procedure in the iterative CCSD steps and a $N_o^{**3} * N_u^{**4}$ (or N^{**7}) procedure in the non-iterative steps related to the calculation of triples (T3) energy corrections, is capable of providing results of the CISDTQ or better quality (CISDTQ is an iterative $N_o^{**4} * N_u^{**6}$ or N^{**10} procedure) when closed-shell molecules are examined. Here and elsewhere in this section, N_o and N_u are the numbers of correlated occupied and unoccupied orbitals. Symbol N designates a measure of the system size in the following sense: $N=2$ means a simultaneous increase of the number of correlated electrons and basis functions by a factor of two. Unlike single- and multi-reference CI methods and some variants of multi-reference perturbation theory, all standard CC methods, such as CCSD or CCSD(T), provide a size extensive description of molecular systems, i.e. no loss of accuracy occurs due to the mere increase of the system size when CC calculations are performed.

Thanks to numerous advances in both the formal aspects of CC theory and the development of efficient computer codes, the single-reference CC approaches, such as CCSD and CCSD(T), are nowadays routinely used in calculations for non-degenerate closed- and open-shell electronic ground states of atomic and molecular systems with up to 50 or so correlated electrons and up to 200-300 or so basis functions. The application of the local correlation formalism within the context of CC theory enables one to extend the applicability of the CCSD(T) and similar CC approaches to systems with approximately 100 light atoms

(hundreds of correlated electrons and > 1000 basis functions). Generalizations of CC theory to open-shell, quasi-degenerate, and excited states are possible, via the multi-reference, renormalized, extended, equation-of-motion, and response CC formalisms, and some of these extensions (for example, the equation-of-motion CC methods for excited states) have become as popular as the multi-reference CI, multi-reference perturbation theory, or CASSCF methods. We should also add that CC theory is a fundamental many-body formalism, whose applicability ranges from electronic structure of atoms and molecules and nuclear physics to extended systems, phase transitions, condensed matter theory, theories of homogeneous electron gas, and relativistic quantum field theory, to mention a few examples. Examples of applications of quantum chemical CC methods in ab initio calculations for atomic nuclei using modern nucleon-nucleon interactions by Piecuch and co-workers are listed in the reference section below.

A number of review articles have been written over the years and it is difficult to cite all of them here. We recommend that users of GAMESS planning to use CC/EOMCC methods read one or more reviews listed below:

"Coupled-cluster theory"

J. Paldus, in S. Wilson and G.H.F. Diercksen (Eds.), *Methods in Computational Molecular Physics*, NATO Advanced Study Institute, Series B: Physics, Vol. 293, Plenum, New York, 1992, pp. 99-194.

"Applications of post-Hartree-Fock methods: a tutorial."

R.J. Bartlett and J.F. Stanton, in K.B. Lipkowitz and D.B. Boyd (Eds.), *Reviews in Computational Chemistry*, Vol. 5, VCH Publishers, New York, 1994, pp. 65-169.

"Coupled-Cluster Theory: Overview of Recent Developments"

R.J. Bartlett, in D.R. Yarkony (Ed.), *Modern Electronic Structure Theory*, Part I, World Scientific, Singapore, 1995, pp. 1047-1131.

"Achieving chemical accuracy with coupled-cluster theory"

T.J. Lee and G.E. Scuseria, in S.R. Langhoff (Ed.), *Quantum Mechanical Electronic Structure Calculations with Chemical Accuracy*, Kluwer, Dordrecht, The Netherlands, 1995, pp. 47-108.

"Coupled-cluster Theory"

J. Gauss, in *Encyclopedia of Computational Chemistry*, P.v.R. Schleyer, N.L. Allinger, T. Clark, J. Gasteiger, P.A. Kollman, H.F. Schaefer III, P.R. Schreiner (Eds.) Wiley, Chichester, U.K., 1998, Vol. 1, pp. 615-636.

"A Critical Assessment of Coupled Cluster Method in Quantum Chemistry"

- J. Paldus and X. Li, *Adv. Chem. Phys.* 110, 1-175 (1999),
"EOMXCC: A New Coupled-Cluster Method for Electronically
Excited States"
P. Piecuch and R.J. Bartlett, *Adv. Quantum Chem.* 34,
295-380 (1999).
- "An Introduction to Coupled Cluster Theory for
Computational Chemists"
T.D.Crawford, H.F.Schaefer in K.B. Lipkowitz and D.B.Boyd
(Eds.), *Reviews in Computational Chemistry*, Vol. 14, VCH
Publishers, New York, 2000, pp. 33-136.
- "In Search of the Relationship between Multiple Solutions
Characterizing Coupled-Cluster Theories"
P. Piecuch and K. Kowalski, in J. Leszczynski (Ed.),
Computational Chemistry: Reviews of Current Trends,
Vol. 5, World Scientific, Singapore, 2000), pp. 1-104.
- "Recent Advances in Electronic Structure Theory: Method of
Moments of Coupled-Cluster Equations and Renormalized
Coupled-Cluster Approaches"
P. Piecuch, K. Kowalski, I.S.O. Pimienta, M.J. McGuire,
Int. Rev. Phys. Chem. 21, 527-655 (2002).
- "New Alternatives for Electronic Structure Calculations:
Renormalized, Extended, and Generalized Coupled-Cluster
Theories"
P. Piecuch, I.S.O. Pimienta, P.-F. Fan, and K. Kowalski,
in J. Maruani, R. Lefebvre, and E. Brandas (Eds.),
Progress in Theoretical Chemistry and Physics, Vol. 12,
Advanced Topics in Theoretical Chemical Physics,
Kluwer, Dordrecht, 2003, pp. 119-206.
- "Coupled Cluster Methods"
J. Paldus, in *Handbook of Molecular Physics and Quantum
Chemistry*, edited by S. Wilson (Wiley, Chichester, 2003),
Vol. 2, pp. 272-313.
- "Method of Moments of Coupled-Cluster Equations: A New
Formalism for Designing Accurate Electronic Structure
Methods for Ground and Excited States"
P. Piecuch, K. Kowalski, I.S.O. Pimienta, P.-D. Fan, M.
Lodriguito, M.J. McGuire, S.A. Kucharski, T. Kus, and M.
Musiał,
Theor. Chem. Acc. 112, 349-393 (2004).
- "Noniterative Coupled-Cluster Methods for Excited
Electronic States"
P. Piecuch, M. Wloch, M. Lodriguito, and J.R. Gour,
in *Progress in Theoretical Chemistry and Physics*, Vol.
15, *Recent Advances in the Theory of Chemical and
Physical Systems*," edited by S. Wilson, J.-P. Julien, J.
Maruani, E. Brandas, and G. Delgado-Barrio (Springer,
Berlin, 2006), pp. XXX-XXXX, in press.
- "Bridging Quantum Chemistry and Nuclear Structure Theory:

Coupled-Cluster Calculations for Closed- and Open-Shell Nuclei"

P. Piecuch, M. Wloch, J.R. Gour, D.J. Dean, M. Hjorth-Jensen, and T. Papenbrock, in V. Zelevinsky (Ed.), *Nuclei and Mesoscopic Physics: Workshop on Nuclei and Mesoscopic Physics WNMP 2004*, AIP Conference Proceedings, Vol. 777, AIP Press, 2005, pp. 28-45.

These reviews point to the other review articles and many original papers. The list of original papers relevant to CC/EOMCC methods implemented in GAMESS is provided below.

available computations (ground states)

The CC programs incorporated in GAMESS enable user to perform conventional LCCD, CCD, CCSD, CCSD[T] (also known as CCSD+T(CCSD)), CCSD(T), and CCSD(TQ) calculations, renormalized (R) and completely renormalized (CR) CCSD[T], CCSD(T), and CCSD(TQ) calculations, and calculations using the rigorously size extensive completely renormalized CR-CC(2,3) (or CR-CCSD(T)L) approach for closed-shell RHF references. Performance of the ground-state CC methods has been discussed in a number of places (cf. the review articles mentioned above and references listed at the end of the "Coupled-Cluster Theory" section). Methods such as, for example, CCSD(T), CR-CC(2,3), and CCSD(TQ) provide excellent results for molecules in or near the equilibrium geometries. Almost all CC methods are excellent in describing dynamical correlation, while being relatively inexpensive and easy to use. One must remember, however, that the conventional single-reference CC methods, such as CCSD(T), should not be applied to bond breaking, diradicals, and other quasi-degenerate states, particularly (but not only) when the RHF determinant is used as a reference. In some of the most frequent cases of electronic quasi-degeneracies, including single-bond breaking and diradicals, the CR-CCSD(T), CR-CCSD(TQ), and CR-CC(2,3)=CR-CCSD(T)L methods can be used instead. The recently proposed CR-CC(2,3) approach seems particularly promising in this regard, although the CR-CCSD(T) and CR-CCSD(TQ) approaches are very useful as well. The CR-CC(2,3) method has costs similar to those characterizing the CCSD(T) approach, while providing the results of the very high, full CCSDT, quality for diradicals and single-bond breaking where CCSD(T) fails. At the same time, the accuracy of CR-CC(2,3) calculations is comparable to or, sometimes, even better than that obtained with the conventional CCSD(T) approach for closed-shell molecules near the equilibrium

geometries. Just like CCSD(T), the CR-CC(2,3) approximation is rigorously size extensive, while working much better than CCSD(T) when non-dynamical correlation effects become large. CR-CC(2,3) (CCTYP=CR-CCL) is among the most attractive ground-state CC options in GAMESS, providing GAMESS users with the highly accurate energies in the closed-shell, single-bond breaking, and diradical regions of molecular potential energy surfaces, and a number of one-electron properties calculated at the CCSD level at a price of single, relatively inexpensive calculation of the CCSD(T) type.

One of the interesting features of GAMESS that can be particularly useful in high accuracy calculations for closed-shell systems is the presence of the (TQ) corrections to CCSD energies among various ground-state CC options. This includes the factorized CCSD(TQ),b method suggested by Kowalski and Piecuch, which describes triples effects at the CCSD(T) level, using noniterative steps that scale as N^7 with the system size, while providing information about the dominant effects due to quadruply excited clusters. The CCSD(TQ),b method is closely related to its CCSD(TQf) predecessor proposed by Kucharski and Bartlett. In fact, if desired, one can extract the CCSD(TQf) energy from the information printed in the GAMESS output when CCTYP=CCSD(TQ) or CR-CC(Q) as follows:

$$\text{CCSD} + [\text{R1-CCSD(TQ),A} - \text{CCSD}] * [\text{CCSD(TQ),A} / \text{DENOMINATOR}]$$

(the R1-CCSD(TQ),A method in the GAMESS output represents one of the renormalized CCSD(TQ) approaches, termed R-CCSD(TQ)-1,a, which are discussed below). The differences between the CCSD(TQ),b and CCSD(TQf) methods are minimal and the accuracies and costs of both approaches are virtually identical. In particular, both methods use relatively inexpensive noniterative steps that scale as N^6 or N^7 with the system size to determine the quadruples corrections.

The unique features of the ground-state CC code in GAMESS are the renormalized (R) and completely renormalized (CR) CCSD[T], CCSD(T), and CCSD(TQ) methods [see K. Kowalski and P. Piecuch, J. Chem. Phys. 113, 18-35 (2000), *idem.*, *ibid.* 113, 5644-5652 (2000), and P. Piecuch and K. Kowalski, in J. Leszczynski (Ed.), Computational Chemistry: Reviews of Current Trends, Vol. 5, World Scientific, Singapore, 2000, pp. 1-104], and the most recent (Fall 2005), rigorously size extensive formulation of CR-CCSD(T), termed CR-CC(2,3) or CR-CCSD(T)L [see P. Piecuch and M.

Wloch, J. Chem. Phys. 123, 224105-1 - 224105-10 (2005) and P. Piecuch, M. Wloch, J.R. Gour, and A. Kinal, Chem. Phys. Lett. 418, 467-474 (2006)]. All of these approaches are based on the more general formalism of the method of moments of coupled-cluster equations (MMCC; biorthogonal MMCC in the case of CR-CC(2,3)), developed by the Piecuch group at Michigan State University. They remove or considerably reduce the pervasive failing of the conventional CCSD[T], CCSD(T), and CCSD(TQ) approximations at larger internuclear separations and for diradical systems, while preserving the ease of use and the relatively low cost of the single-reference methods of the CCSD(T) or CCSD(TQ) type. In analogy to the CCSD[T], CCSD(T), and CCSD(TQ) methods, the R-CCSD[T], R-CCSD(T), R-CCSD(TQ)-n,x (n=1,2;x=a,b), CR-CCSD[T], CR-CCSD(T), CR-CC(2,3), and CR-CCSD(TQ),x (x=a,b) approaches are based on an idea of improving the CCSD results by adding a posteriori noniterative corrections to CCSD energies. These corrections employ the generalized moments of CCSD equations (projections of the Schroedinger equation for the CCSD wave function on the triply (T) or triply and quadruply (TQ) excited determinants) and are designed by extracting the leading terms that define the theoretical difference between the CCSD and full CI energies. The CR-CCSD[T], CR-CCSD(T), and CR-CC(2,3) approaches are capable of eliminating the unphysical humps on the potential energy surfaces involving single bond breaking produced by the conventional CCSD[T] and CCSD(T) methods. They also significantly improve the poor description of diradical species (for example, diradical transition states and intermediates) by the CCSD[T] and CCSD(T) methods. What is important in practical applications, the CR-CCSD(T) and CR-CC(2,3) approaches are capable of providing a good balance between the dynamical and nondynamical correlation effects when the diradical and closed-shell structures have to be examined together. The rigorously size extensive CR-CC(2,3) method is particularly effective in this regard, although the older and somewhat less expensive CR-CCSD(T) approach is very useful as well. The R-CCSD[T] and R-CCSD(T) approaches may improve the CCSD[T] and CCSD(T) results at intermediate internuclear separations, but they usually fail at larger distances. The CR-CCSD[T], CR-CCSD(T), and CR-CC(2,3) methods are better in this regard, since they often provide a very good description of single bond breaking at all internuclear separations. This includes various cases of unimolecular dissociations and exchange and bond insertion chemical reactions, in which single bonds break and form. We DO NOT recommend applying the CR-CCSD[T], CR-CCSD(T), and CR-CC(2,3) approaches to multiple

bond breaking, although some types of multiple bond stretching can be described by these methods very well if the relevant stretches of chemical bonds are not too large. In general, however, multiple bond dissociations require using the higher-order methods, such as the completely renormalized CCSD(TQ) and CCSDT(Q) approaches (the CR-CCSD(TQ) methods are available in GAMESS), the so-called MMCC(2,6) method, and the more recent generalized and quadratic MMCC methods, if the single-reference approach is preferred, or the multi-reference CC methods of the state-universal and state-specific type (some of the most promising approaches in these categories, including active-space and state-universal CC methods, will be included in GAMESS in the future). In particular, the CR-CCSD(TQ) approaches available in GAMESS are reasonably accurate in situations involving double bond dissociations and a simultaneous stretching or breaking of two single bonds. They may work reasonably well even when the triple bond stretching or breaking is examined, but the results for more complicated cases of bond breaking are not as good as those that one can obtain with the best multi-reference approaches. A detailed description of the R-CCSD[T], R-CCSD(T), CR-CCSD[T], CR-CCSD(T), CR-CC(2,3), R-CCSD(TQ), and CR-CCSD(TQ) approaches and other MMCC methods can be found in several papers by Piecuch and coworkers listed at the very end of the "Coupled-Cluster Theory" section.

Unlike the newest CR-CC(2,3) approximation, the somewhat older R-CCSD[T], R-CCSD(T), CR-CCSD[T], CR-CCSD(T), R-CCSD(TQ), and CR-CCSD(TQ) methods are not strictly size extensive, i.e. there are unlinked terms in the MBPT (many-body perturbation theory) expansions of the renormalized and completely renormalized [T], (T), and (TQ) corrections to CCSD energies. This has little or no effect on bond breaking (on the contrary, the CR-CCSD[T], CR-CCSD(T), and CR-CCSD(TQ) potential surfaces are MUCH better than potential energy surfaces obtained in the standard and size extensive CCSD[T], CCSD(T), and CCSD(TQ) calculations), but lack of strict size extensivity may have an effect on the results of calculations for larger and extended systems. A lot depends on the values of T2 amplitudes and the chemical problem of interest. If the T2 amplitudes are small, then the overlap denominator expressions which define the renormalized [T], (T), and (TQ) corrections of the R-CCSD[T], R-CCSD(T), CR-CCSD[T], CR-CCSD(T), R-CCSD(TQ), and CR-CCSD(TQ) methods are close to 1, in which case there is no major problem. If the T2 amplitudes are large, then these denominators may become significantly greater than 1. This behavior of the R-CCSD[T], R-CCSD(T), CR-CCSD[T], CR-

CCSD(T), R-CCSD(TQ), and CR-CCSD(TQ) denominator expressions is extremely useful for improving the results for bond breaking, since the denominators defining the renormalized [T], (T), and (TQ) corrections damp the unphysical values of the standard [T], (T), and (TQ) corrections at larger internuclear separations or when the wave function gains a significant multi-reference character. The same applies to diradical species, where the standard [T], (T), and (TQ) corrections produce unphysical results and need damping that the renormalized methods provide. However, for larger many-electron systems (with 50 correlated electrons or more), the denominators defining the renormalized [T], (T), and (TQ) corrections may "overdamp" the [T], (T), and (TQ) energy corrections. On the other hand, the renormalized [T], (T), and (TQ) energy corrections are constructed using the cluster amplitudes resulting from the size extensive CCSD calculations. Moreover, it is often the case that the number of correlated electrons used in CC calculations for larger molecules (and only these electrons are used in constructing the renormalized [T], (T), and (TQ) corrections to CCSD energies) is much smaller than the total number of electrons. Thus, the consequences of the lack of strict size extensivity of the R-CCSD[T], R-CCSD(T), CR-CCSD[T], CR-CCSD(T), R-CCSD(TQ), and CR-CCSD(TQ) methods do not have to be serious for larger systems, particularly when one examines, for example, the relative energies of stationary points along the reaction pathways relative to the relevant reactants (see comments below). A number of interesting chemical problems involving smaller and medium size polyatomic diradical systems, including, for example, the Cope rearrangement of 1,5-hexadiene, the cycloaddition of cyclopentyne to ethylene, the isomerizations of bicyclopentene and tricyclopentane into cyclopentadiene, the thermal stereomutations of cyclopropane, and the relative energetics of dicopper systems relevant to molecular oxygen activation by copper metalloenzymes, where the standard CCSD(T) approach and, in some cases, the low-order multi-reference perturbation theory methods encounter serious difficulties, have been successfully examined with the CR-CCSD(T) approach, demonstrating that problems of size extensivity in CR-CCSD(T) calculations are of no major significance in molecules of these sizes. But one may have to be more careful when chemical systems have more than 50 correlated electrons. Extensive numerical tests indicate that lack of strict size extensivity has little (fraction of a millihartree or so) effect on the results of the CR-CCSD[T], CR-CCSD(T), and CR-CCSD(TQ) calculations for

smaller systems. For larger systems, such as the glycine dimer described by the 6-31G basis set, the departure from rigorous size extensivity, as measured by forming the difference of the sum of the energies of isolated glycine molecules from the energy of the dimer consisting of glycine molecules at very large (200 bohr) distance, is ca. 3 millihartree (2 kcal/mol). The violation of strict size extensivity by the CR-CCSD(T) methods has been estimated at approximately 0.5 % of the total correlation energy (changes in the correlation energy if the relative energies along reaction pathways are examined), which is often a small price to pay considering the significant improvements that the renormalized CC methods offer for potential energy surfaces and diradicals and the ease with which the CR-CC calculations can be performed. **IMPORTANT PRACTICAL ADVICE:** In studies of reaction pathways with the CR-CCSD(T) approach, where reactants and products are connected by one or more transition states and intermediates and where there are two or more reactants, we **STRONGLY RECOMMEND** that the user of CR-CCSD(T) proceeds in a manner similar to multi-reference CI calculations. Thus, we advise to calculate the energies of transition states, intermediates, and products relative to reactants, using the total CR-CCSD(T) energy of a noninteracting complex formed by reactants (reactants separated by a large distance, say, 200 Angs.) as the reference energy of reactants rather than the sum of the CR-CCSD(T) energies of isolated reactants. This reduces the possible size extensivity errors in the CR-CCSD(T) calculations for larger systems to a minimum, since all species along a reaction pathway (including reactants, transition states, intermediates, and products) are treated then in the same, well balanced, manner. Similar remarks apply to the CR-CCSD(TQ) (and all R-CC) calculations. None of the above has to be done when the CR-CC(2,3) approach is employed, since CR-CC(2,3) is size extensive and the CR-CC(2,3) energy of A+B equals the sum of CR-CC(2,3) energies of A and B.

The rigorously size extensive modifications of the CR-CC methods have recently (2005) been developed, using the idea of locally renormalized methods, such as LR-CCSD(T), which lead to size extensive results when localized orbitals are employed, and, in an alternative formulation, the idea of exploiting the left CC states combined with the so-called biorthogonal MMCC theory. The latter development seems particularly attractive. The resulting CR-CC(2,3) method, also called CR-CCSD(T)L, which combines the best features of CCSD(T) and CR-CCSD(T) and which we already mentioned above, satisfies the following criteria: (i) is

at least as accurate as (sometimes more accurate than) CCSD(T) for nondegenerate ground states, (ii) provides highly accurate results for single-bond breaking and diradicals with the noniterative $N_o^{**3} * N_u^{**4}$ steps similar to those of CCSD(T) and CR-CCSD(T), (iii) is more accurate than the CR-CCSD(T), LR-CCSD(T), and other non-iterative triples CC approaches, such as CCSD(2)T, which all aim at eliminating the failures of CCSD(T) in the diradical/bond breaking regions, and (iv) is rigorously size extensive without localizing orbitals. The criterion (ii) of a highly accurate description at the triples level of CC theory is defined here by the accuracy provided by the full CCSDT approach, which is almost exact in studies of diradicals and single-bond breaking, but also limited to very small systems with up to 2-3 light atoms due to very expensive iterative $N_o^{**3} * N_u^{**5}$ steps that it uses. As demonstrated, for example, in recent studies of the relative energetics of the Cu₂O₂ systems with up to six ammonia ligands and thermal stereomutations of cyclopropane involving the trimethylene diradical as a transition state, CR-CC(2,3) has a wide range of applicability that includes larger polyatomic systems with up to 10-20 light and a few transition metal atoms. At the same time, CR-CC(2,3) provides a size extensive, highly accurate, and well balanced description of dynamical and nondynamical correlation effects in studies of single bond breaking and diradicals, particularly when the molecular systems involving a varying degree of diradical character along the relevant reaction pathways are examined.

For all these reasons, the CR-CC(2,3) approach has been recently included in GAMESS. The CR-CC(2,3) method (invoked by typing CCTYP=CR-CCL in the input) seems to represent the most accurate non-iterative triples CC approximation formulated to date. Since the construction of the triples corrections to CCSD energies in CR-CC(2,3) calculations requires the determination of the left CCSD eigenstates, the CCPRP variable from \$CCINP is automatically set at .TRUE. when variable CCTYP in \$CONTRL is set at CR-CCL. As a result, by running the CR-CC(2,3) calculations, the user of GAMESS obtains a great deal of useful information in addition to excellent energetics (excellent as long as multiple bonds are not broken). This information includes the first-order reduced density matrices (printed in the PUNCH file), natural occupation numbers, and a variety of one-electron properties (e.g., electrostatic multipole moments) calculated at the CCSD level of theory. The ground-state CR-EOMCCSD(T) energies (cf. the next

subsection), corresponding to CCTYP=CR-EOM calculations with NSTATE(1)=0,0,0,0,0,0,0,0, are printed as well.

The CR-CC(2,3) approach has several variants, labeled with an additional letter, A-D (D means a full treatment of the perturbative denominators that are used to define triple excitation components, based on the diagonal matrix elements of the triples-triples block of the CCSD similarity transformed Hamiltonian; A means the crudest treatment of these denominators through bare orbital energies). Of all printed CR-CC(2,3) energies, the CR-CC(2,3),D value, which corresponds to the most complete variant of CR-CC(2,3), is the most accurate one and we STRONGLY RECOMMEND to use it in high accuracy calculations of molecular energetics. Because of the way the CR-CC(2,3),D approach is presently implemented in GAMESS, it is safer, for now, to use the simplified CR-CC(2,3),A or CR-CC(2,3),B models in numerical derivative calculations if there are orbital degeneracies (the aforementioned CCSD(2)T approach is equivalent to the CR-CC(2,3),A approximation). Because of some small simplifications in the present computer implementation of the CR-CC(2,3),D method, the CR-CC(2,3),D energies may slightly depend on the choice of molecular coordinate system if there are orbital degeneracies. Although changes in the most accurate CR-CC(2,3),D energies for systems with orbital degeneracies due to changes of the coordinate system are minimal (0.1 millihartree or less), it is safer to calculate numerical CR-CC(2,3) derivatives for systems with orbital degeneracies using the CR-CC(2,3),A or CR-CC(2,3),B approximations. For this reason, the CR-CC(2,3),A energy is automatically passed to the numerical derivative calculations with GAMESS if they are requested by the user, with the most complete CR-CC(2,3),D approach providing the most accurate energetics. We should emphasize, however, that the above technical issues are only limited to systems with orbital degeneracies. When there are no orbital degeneracies (which is the case when the highest molecular symmetry group is an Abelian group), the present implementation of the CR-CC(2,3),D approach in GAMESS leads to perfectly invariant energies. The issue of a slight (0.1 millihartree or less) dependence of the CR-CC(2,3),D (also CR-CC(2,3),C) energies on the choice of molecular coordinate system when orbital degeneracies are present is only temporary and will be eliminated in the future releases of GAMESS via a suitable modification of the CR-CC(2,3) code.

Since CR-CC methods can find use in applications involving bond breaking and reaction pathways, one has to make sure that the underlying solution of the CCSD equations, on which the completely renormalized [T], (T), (2,3), and (TQ) corrections are based, represents the same physical solution as those defining other regions of a given molecular potential energy surface. This remark is quite important, since, for example, diradical regions of potential energy surface are characterized by larger cluster amplitudes and one has to make sure that the properly converged values of these amplitudes are obtained. GAMESS is equipped with a good algorithm for converging CCSD equations and a restart option discussed in a later part of this document that facilitate converging larger cluster amplitudes in difficult cases.

The user is encouraged to examine various interesting elements of the CC input and output. In addition to CC energies, GAMESS prints the largest T1 and T2 cluster amplitudes obtained in the CCSD calculations, the T1 diagnostic, norms of T1 and T2 vectors, and the R-CCSD[T], R-CCSD(T), and R-CCSD(TQ) denominators that define the renormalized and completely renormalized triples and quadruples corrections. For example, bond breaking and diradical cases are characterized by larger cluster amplitudes (particularly, T2) and a significant increase in the values of the R-CCSD[T], R-CCSD(T), and CR-CCSD(TQ) denominators, which damp unphysical triples and quadruples corrections of the standard CCSD[T], CCSD(T), and CCSD(TQ) approximations, compared to closed-shell regions of potential energy surface. As already mentioned, the CR-CC(2,3) calculations provide user with one-particle reduced density matrices, natural occupation numbers, and a number of one-electron properties, calculated at the CCSD level, in addition to the highly accurate CR-CC(2,3) and some other CR-CC energies.

available computations (excited states)

The equation of motion coupled cluster (EOMCC) method and the closely related response CC and symmetry-adapted cluster configuration interaction (SAC-CI) approaches provide very useful extensions of the ground-state CC theory to excited states. In the EOMCC theory, the excited states $|\Psi_i^K\rangle$ are obtained by applying the excitation operator

$$R = R_0 + R_1 + R_2 + \dots,$$

where R_0 , R_1 , R_2 , etc. are the reference, singly excited (1-particle-1-hole), doubly excited (2-particle-2-hole), etc. components of R , to the CC ground state $|\Psi_0\rangle$. Thus, the EOMCC expression for the excited state $|\Psi_K\rangle$ is

$$\begin{aligned} |\Psi_K\rangle &= R |\Psi_0\rangle = R \exp(T) |\Phi\rangle \\ &= (R_0 + R_1 + R_2 + \dots) \exp(T_1 + T_2 + \dots) |\Phi\rangle . \end{aligned}$$

In practice, the standard EOMCC calculations are performed by diagonalizing the CC similarity transformed Hamiltonian $\bar{H} = \exp(-T) H \exp(T)$ in the space of excited determinants included in the cluster operator T and the excitation operator R . For example, the basic EOMCCSD calculations defined by the truncation schemes $T=T_1+T_2$ and $R=R_0+R_1+R_2$ are performed by diagonalizing $\exp(-T_1-T_2) H \exp(T_1+T_2)$ in the space of singly and doubly excited determinants defining the CCSD ($T=T_1+T_2$) approximation. The direct result of such diagonalization are the vertical excitation energies $\omega_K = E_K - E_0$ (E_K and E_0 are the excited- and ground- state energies, respectively).

The EOMCC methods have several advantages. The most expensive steps of the basic EOMCCSD calculations scale only as $N_o^2 * N_u^4$ and yet the accuracy of the EOMCCSD results for excited states dominated by one-electron transitions (single excitations or singles or 1-particle-1-hole excitations) is very good. The errors in the EOMCCSD calculations for such states are often on the order of 0.1-0.3 eV, which is acceptable in many applications. The EOMCCSD approximation and other standard EOMCC methods have an ease of application that is not matched by the multi-reference techniques, since formally the EOMCC theory is a single-reference formalism. Thus, the EOMCC methods are particularly well suited for calculations where active orbital spaces required in CASSCF-related calculations become very large or difficult to identify. Given sufficient computational resources, the EOMCCSD calculations for systems involving up to 10-20 light or a few heavy atoms are nowadays (meaning year 2004 and on) routine. The EOMCCSD method works reasonably well for excited states dominated by singles, but it fails to describe states dominated by two-electron transitions (doubles) and potential energy surfaces along bond breaking coordinates. These failures can be remedied by the CR-EOMCCSD(T) approximations described below.

The EOMCC programs incorporated in GAMESS enable user to perform standard EOMCCSD calculations employing the RHF

reference determinant. They also enable to improve the EOMCCSD results by adding the state-selective noniterative corrections due to triples to the ground and excited-state CCSD/EOMCCSD energies via the completely renormalized EOMCCSD(T) (CR-EOMCCSD(T)) approaches developed by the Piecuch group. The CR-EOMCCSD(T) approaches represent extensions of the ground-state CR-CCSD(T) method to excited states. In particular, in analogy to the CR-CCSD(T) approximation, the excited-state CR-EOMCCSD(T) approaches are based on the formalism of the method of moments of coupled-cluster equations (MMCC). Moreover, the CR-EOMCCSD(T) methods preserve the relatively low computer costs and ease of use of the ground-state CCSD(T) calculations. The most expensive noniterative steps of the CR-EOMCCSD(T) approach scale as $N_o^3 * N_u^4$. The CR-EOMCCSD(T) option (CCTYP=CR-EOM) is a unique feature of GAMESS. At this time, the applicability of the EOMCCSD and CR-EOMCCSD(T) codes in GAMESS is limited to singlet states.

The main advantage of the MMCC-based CR-EOMCCSD(T) approximations, in addition to their "black-box" character and relatively low computer costs, is their high (0.1 eV or so) accuracy in the calculations of excited states dominated by double excitations and excited-state potential energy surfaces along bond breaking coordinates, for which the standard EOMCCSD method fails (producing errors on the order of 1 eV or even bigger). In this regard, the CR-EOMCCSD(T) methods are quite similar to the CR-CCSD(T) approach, which is capable of describing ground-state potential energy surfaces involving single bond breaking. As a matter of fact, when limited to the ground-state problem, the CR-EOMCCSD(T) approximations become essentially identical to the CR-CCSD(T) method. There are, however, small differences and the CR-EOMCCSD(T) energies of the ground state are slightly different than the CR-CCSD(T) energies discussed in the earlier section. This is due to the fact that the original CR-CCSD(T) approximation has been designed for the ground states only, whereas the CR-EOMCCSD(T) approaches apply to ground and excited states and this required small modifications in the ground-state energy equations.

A few different variants of the CR-EOMCCSD(T) method, termed the CR-EOMCCSD(T),IX, CR-EOMCCSD(T),IIX, and CR-EOMCCSD(T),III approaches (X=A,B,C,D) have been proposed and included in GAMESS. Types I, II, and III refer to three different ways of defining the approximate wave functions $|\Psi_i\rangle$ that are used to construct the CR-EOMCCSD(T) triples corrections to EOMCCSD energies in the

underlying MMCC formalism. Types I and II use perturbative expressions for $|\Psi_i^K\rangle$ in terms of cluster components T1 and T2 and excitation components R0, R1, and R2. Type III uses additional CISD (CI singles and doubles) calculations in designing the wave functions $|\Psi_i^K\rangle$ that enter the CR-EOMCCSD(T) triples corrections. Thus, user should be aware of the fact that CR-EOMCCSD(T),III calculations involve the single-reference CISD calculations, in addition to the CCSD, EOMCCSD, and (T) steps common to all CR-EOMCCSD(T) methods. This increases the CPU timings of the CR-EOMCCSD(T),III calculations, when compared to CR-EOMCCSD(T),IX and CR-EOMCCSD(T),IIX (X=A-D) approaches. Additional letters A-D that label the CR-EOMCCSD(T),I and CR-EOMCCSD(T),II approximations refer to different ways of treating perturbative denominators in evaluating the (T) triples corrections (D means full treatment of these denominators, based on the diagonal matrix elements of the triples-triples block of the CCSD similarity transformed Hamiltonian, A means the crudest treatment through bare orbital energies). The user interested in further details is referred to a 2004 paper by Kowalski and Piecuch (J. Chem. Phys. 120, 1715-1738 (2004)).

Our experience to date indicates that the CR-EOMCCSD(T),ID and CR-EOMCCSD(T),III methods are the most accurate ones when it comes to the calculations of excited states dominated by double excitations and excited-state potential energy surfaces along bond breaking coordinates, at least for moderate bond stretches. The CR-EOMCCSD(T),ID and CR-EOMCCSD(T),III methods are particularly good when examining the total energies of excited states (for example, as functions of nuclear geometries). If the user is only interested in vertical excitation energies rather than total energies, the good balance between ground and excited states, particularly when excited states are dominated by doubles, can be achieved by considering mixed approximations, such as CR-EOMCCSD(T),ID/IB. The ID/IB acronym means that the excitation energy is obtained by subtracting the CR-EOMCCSD(T),IB ground-state energy from the CR-EOMCCSD(T),ID energy of excited state. Other mixed approaches (IID/IB, etc.) are obtained in a similar way. The ID/IB results are particularly good when the excited states have significant doubly excited character. The fact that the CR-EOMCCSD(T),ID results for excited states are usually better than the CR-EOMCCSD(T),IA,IB,IC results is related to a better treatment of perturbative denominators in evaluating the (T) triples corrections in the CR-EOMCCSD(T),ID approximation.

In addition to the total CR-EOMCCSD(T),IX, CR-EOMCCSD(T),IIX (X=A-D), and CR-EOMCCSD(T),III energies and vertical excitation energies based on the idea of mixing different approximations for excited and ground states (the ID/IA, IID/IA, ID/IB, and IID/IB excitation energies), GAMESS prints the so-called DELTA-CR-EOMCCSD(T) values (the del(IA), del(IB), del(IC), del(ID), del(IIA), del(IIB), del(IIC), del(IID), and del(III) energies). These are the vertical excitation energies obtained by directly correcting the EOMCCSD excitation energies rather than the total CCSD/EOMCCSD energies by triples corrections. For example, del(ID) refers to the vertical excitation energy obtained by subtracting the CCSD ground-state energy from the excited-state CR-EOMCCSD(T),ID energy. The DELTA-CR-EOMCCSD(T) values may be somewhat worse than the pure CR-EOMCCSD(T) (e.g., CR-EOMCCSD(T),ID) or CR-EOMCCSD(T),III) or mixed CR-EOMCCSD(T) (e.g., CR-EOMCCSD(T),ID/IB)) values of vertical excitation energies for states dominated by doubles, but they may provide a reasonable balance between ground and excited states and somewhat bigger improvements for vertical excitation energies corresponding to states dominated by singles. The DELTA-CR-EOMCCSD(T) methods provide a reasonably good balance between improvements in the results for excited states dominated by singles and improvements in the results for excited states dominated by doubles, but one should treat this remark with caution.

In addition to the above CR-EOMCCSD(T) results, GAMESS also prints the so-called (T)/R excitation energies. These are the analogs of the EOMCCSD(T~) excitation energies proposed by Watts and Bartlett, obtained by using the right eigenvectors of the CCSD similarity transformed and right-hand moments of EOMCCSD equations rather than the left eigenstates of EOMCCSD and left-hand analogs of the EOMCCSD moments (see K. Kowalski and P. Piecuch, J. Chem. Phys. 120, 1715-1738 (2004) for details). Just like the EOMCCSD(T~) method of Watts and Bartlett, the (T)/R approach is based on the idea of directly correcting the EOMCCSD vertical excitation energies by triples. In analogy to the EOMCCSD(T~) method, the (T)/R corrections improve the EOMCCSD results for states dominated by singles, but they may fail to produce reasonable results for states dominated by doubles and for excited-state potential energy surfaces along bond breaking coordinates. The CR-EOMCCSD(T) methods are considerably more robust in this regard.

In performing the CR-EOMCCSD(T) calculations, user should realize that the EOMCCSD method can provide a wrong

state ordering if low-lying doubly excited states are mixed up with singly excited states in the electronic spectrum. This may require calculating a larger number of EOMCCSD states before correcting them for triples. An example of this situation has been described in K. Kowalski and P. Piecuch, *J. Chem. Phys.* 120, 1715-1738 (2004). The EOMCCSD method provides an incorrect ordering of the singlet A1 states of ozone, so that one must use the third excited EOMCCSD state of the singlet A1 (1A1) symmetry (the fourth 1A1 state total, using the CCSD/EOMCCSD energy ordering of ground and excited states) to calculate the noniterative CR-EOMCCSD(T) triples correction that describes the first excited singlet A1 (the second 1A1) state. Without calculating several states of each symmetry at the EOMCCSD level prior to CR-EOMCCSD(T) calculations, one would risk losing information about some important low-lying doubly excited states. Because of the inherent limitations of the EOMCCSD approximation, complicated doubly excited states resulting from the EOMCCSD calculations may be shifted to high energies, mixing with the singly excited states that are accurately described by the EOMCCSD method. After correcting the EOMCCSD energies for the effect of triples, these doubly excited states may become low-lying states. This is exactly what we observe in the case of ozone and other cases of severe quasi-degeneracies.

The issues of size extensivity in the EOMCCSD and CR-EOMCCSD(T) calculations are highly complex and much beyond the scope of this writing. Briefly, none of the EOMCC methods are rigorously size extensive and yet all EOMCC methods are very useful in great many applications. The EOMCCSD approach is size intensive for excited states dominated by singles and the EOMCCSD energies correctly separate when the one-electron charge-transfer excitations are considered. Thus, the EOMCCSD approach correctly describes the dissociation of a singly excited system $(AB)^*$ into the $A^* + B$, $A + B^*$, $(A^+) + (B^-)$, and $(A^-) + (B^+)$ fragments (* designates a one-electron excitation). We must remember, however, that the above separability properties of the EOMCCSD energies are no longer true if the reference determinant $|\Phi\rangle$ does not separate correctly (for example, the RHF determinant does not correctly separate if the $AB \rightarrow A+B$ fragmentation involves the dissociation of the closed-shell system AB into open-shell fragments A and B). As in the case of the ground-state CR-CCSD(T) approach, the CR-EOMCCSD(T) methods slightly violate the rigorous size extensivity/intensivity (at the level of 1-2 millihartree for systems with up to 30-50 correlated electrons), but at

the same time the CR-EOMCCSD(T) approaches significantly improve a poor description of excited states with significant double excitation components by the EOMCCSD method. As a result, lack of strict size extensivity of the CR-EOMCCSD(T) theories is of relatively minor significance in applications for systems with up to at least 50 correlated electrons [see M. Wloch, J.R. Gour, K. Kowalski, and P. Piecuch, J. Chem. Phys. 122, 214107-1 - 214107-15 (2005) for a thorough discussion of the complicated extensivity issues in EOMCCSD and CR-EOMCCSD(T) calculations].

The user is encouraged to examine various interesting elements of the EOMCC input and output. In addition to EOMCC energies, GAMESS prints the largest R1 and R2 excitation amplitudes and the so-called reduced excitation level (REL) diagnostic, which provides information about the character of a given excited state (REL close to 1 means singly excited, REL close to 2 means doubly excited). GAMESS also prints the R0 value (the coefficient at the reference in the EOMCCSD wave function). If a molecule has symmetry and R0 equals 0, user immediately learns the excited state has a different symmetry than the ground state. GAMESS provides full information about irreps of the calculated excited states.

density matrices and properties

One of the major advantages of EOMCC methods, including EOMCCSD, is a relatively straightforward access to reduced density matrices and molecular properties that these methods offer. This is done by considering the left eigenstates of the similarity transformed Hamiltonian $\bar{H} = \exp(-T) H \exp(T)$ mentioned in the earlier sections. The similarity transformed Hamiltonian \bar{H} is not hermitian, so that, in addition to the right eigenstates $R|\Phi\rangle$, which define the "ket" CC or EOMCC wave functions discussed in the previous section, we can also define the left eigenstates of \bar{H} , $\langle\Phi|L$, which determine the "bra" CC or EOMCC wave functions,

$$\begin{aligned}\langle\psi_K| &= \langle\Phi| L \exp(-T) \\ &= \langle\Phi| (L_0 + L_1 + L_2 + \dots) \exp(-T_1 - T_2 - \dots),\end{aligned}$$

where $L_0=1$ for the ground state and 0 for excited states and where L_1 , L_2 , etc. are the one-body, two-body, etc. deexcitation operators, respectively. In the ground-state case, we often write

$$\begin{aligned}\langle \Psi_0 | &= \langle \Phi | (1 + \Lambda) \exp(-T) \\ &= \langle \Phi | (1 + \Lambda_1 + \Lambda_2 + \dots) \exp(-T_1 - T_2 - \dots),\end{aligned}$$

where Λ_1 , Λ_2 , etc. are the one-body, two-body, etc. components of the deexcitation Λ operator of the analytic gradient CC theory. The left eigenstates of \bar{H} , $\langle \Phi | L$, and the right eigenstates $R | \Phi \rangle$ form a biorthonormal set. We can use these eigenstates to calculate expectation values and transition matrix elements of quantum-mechanical operators (observables), involving the CC and EOMCC ground and excited states, as follows:

$$\langle \Psi_K | W | \Psi_M \rangle = \langle \Phi | L(K) \bar{W} R(M) | \Phi \rangle,$$

where $\bar{W} = \exp(-T) W \exp(T)$ is a similarity transformed form of the observable W we are interested in and where we added labels K and M to operators L and R to indicate the CC/EOMCC electronic states they are associated with. The operator W could be, for example, a dipole or quadrupole moment. It could also be a product of creation and annihilation operators, which we could use to calculate the reduced density matrices. For example, if the operator $W = (a_p^\dagger) a_q$, where a_p^\dagger and a_q are the creation and annihilation operators associated with the spin-orbitals p and q , respectively, we can calculate the CC or EOMCC one-body reduced density matrix in the electronic state K , $\Gamma(qp, K)$, as

$$\begin{aligned}\Gamma(qp, K) \\ = \langle \Phi | L(K) \{ \exp(-T) [(a_p^\dagger) a_q] \exp(T) \} R(K) | \Phi \rangle.\end{aligned}$$

For the corresponding transition density matrix involving two different states K and M , say ground and excited states or some other combination, we can write

$$\begin{aligned}\Gamma(qp, KM) \\ = \langle \Phi | L(K) \{ \exp(-T) [(a_p^\dagger) a_q] \exp(T) \} R(M) | \Phi \rangle.\end{aligned}$$

By having access to reduced density matrices, we can calculate various properties analytically. For example, by calculating the one-body reduced density matrices of ground and excited states and the corresponding transition density matrices, we can determine all one-electron properties and the corresponding transition matrix elements involving one-electron properties using a single mathematical expression:

$$\langle \Psi_K | W | \Psi_M \rangle = \sum_{pq} \langle p | w | q \rangle \Gamma(qp, KM),$$

where $\langle p|w|q \rangle$ are matrix elements of the one-body property operator W in a basis set of molecular spin-orbitals used in the calculations. The calculation of reduced density matrices provides the most convenient way of calculating CC and EOMCC properties of ground and excited states. In addition, by having reduced density matrices, one can calculate CC and EOMCC electron densities,

$$\rho_K(x) = \sum_{pq} \Gamma(qp, K) (\phi_q(x))^* \phi_p(x),$$

where $\phi_p(x)$ and $\phi_q(x)$ are molecular spin-orbitals and x represents the electronic (spatial and spin) coordinates. By diagonalizing $\Gamma(qp, K)$, one can determine the natural occupation numbers and natural orbitals for the CC or EOMCC state $|\Psi_K\rangle$.

The above strategy of handling molecular properties analytically by determining one-body reduced density matrices was implemented in the CC/EOMCC programs incorporated in GAMESS. At this time, the calculations of reduced density matrices and selected properties are possible at the CCSD (ground states) and EOMCCSD (ground and excited states) levels of theory ($T=T_1+T_2$, $R=R_1+R_2$, $L=L_0+L_1+L_2$). Currently, in the main output the program prints the CCSD and EOMCCSD electric multipole (dipole, quadrupole, etc.) moments and several other one-electron properties that one can extract from the CCSD/EOMCCSD density matrices, the EOMCCSD transition dipole moments and the corresponding dipole and oscillator strengths, and the natural occupation numbers characterizing the CCSD/EOMCCSD wave functions. In addition, the complete CCSD/EOMCCSD one-body reduced density matrices and transition density matrices in the RHF molecular orbital basis and the CCSD and EOMCCSD natural orbital occupation numbers are printed in the PUNCH output file. The eigenvalues of the density matrix (natural occupation numbers) are ordered such that the corresponding eigenvectors (CCSD or EOMCCSD natural orbitals) have the largest overlaps with the consecutive ground-state RHF MOs. Thus, the first eigenvalue of the density matrix corresponds to the CCSD or EOMCCSD natural orbital that has the largest overlap with the RHF MO 1, the second with RHF MO 2, etc. This ordering is particularly useful for analyzing excited states, since in this way one can easily recognize orbital excitations that define a given excited state.

One has to keep in mind that the reduced density matrices originating from CC and EOMCC calculations are not symmetric. Thus, if we, for example, want to calculate the

dipole strength between states K and M for the x component of the dipole μ_x , $|\langle \Psi_K | \mu_x | \Psi_M \rangle|^2$, we must write

$$|\langle \Psi_K | \mu_x | \Psi_M \rangle|^2 = \langle \Psi_K | \mu_x | \Psi_M \rangle \langle \Psi_M | \mu_x | \Psi_K \rangle,$$

where each matrix element in the above expression is evaluated using the expression for $\langle \Psi_K | W | \Psi_M \rangle$ shown above. A similar remark applies to the corresponding component of the oscillator strength,

$$(2/3) * |E_K - E_M| * |\langle \Psi_K | \mu_x | \Psi_M \rangle|^2,$$

which we have to write as

$$(2/3) * |E_K - E_M| * \langle \Psi_K | \mu_x | \Psi_M \rangle \langle \Psi_M | \mu_x | \Psi_K \rangle.$$

In other words, both matrix elements $\langle \Psi_K | \mu_x | \Psi_M \rangle$ and $\langle \Psi_M | \mu_x | \Psi_K \rangle$ have to be evaluated, since they are not identical. This is reflected in the GAMESS output, where the user can see quantities such as the left and right transition dipole moments.

From the above description, it follows that in order to calculate reduced density matrices and properties using CC and EOMCC methods, one has to determine the left as well as the right eigenstates of the similarity transformed Hamiltonian \bar{H} . For the ground state, this is done by solving the linear system of equations for the deexcitation operator Λ (in the CCSD case, the one- and two-body components Λ_1 and Λ_2). For excited states, we can proceed in several different ways. We can solve the linear system of equations for the amplitudes defining the EOMCC deexcitation operator L , after determining the corresponding EOMCC excitation operator R and excitation energy ω (recommended option, default in GAMESS), or we can solve for the L and R amplitudes simultaneously in the process of diagonalizing the similarity transformed Hamiltonian. These different ways of solving the EOMCC problem are discussed in section "Eigensolvers for excited-state calculations."

As already mentioned, the left eigenstates of the similarity transformed Hamiltonian of the CCSD approach are also used to construct the triples corrections to CCSD energies defining the rigorously size extensive completely renormalized CR-CC(2,3) approximation. This is why the user gets an immediate access to electrostatic multipole moments and other one-electron properties calculated at the CCSD level, when running the CR-CC(2,3) calculations.

excited state example

```

!   excited states of methylidyne cation...CH+
!   Basis set and geometry come from a FCI study by
!   J.Olsen, A.M.Sanchez de Meras, H.J.Aa.Jensen,
!   P.Jorgensen Chem. Phys. Lett. 154, 380-386(1989).
!
!   EOMCC methods give:
!   STATE          EOMCCSD ID/IA IID/IA ID/IB  IID/IB      FCI
!   B1 (1Pi)       3.261  3.226  3.226  3.225  3.224      3.230
!   A1 (1Delta)    7.888  6.988  6.963  6.987  6.962      6.964
!   A1 (1Sigma+)   9.109  8.656  8.638  8.654  8.637      8.549
!   A1 (1Sigma+)  13.580 13.525 13.526 13.524 13.525     13.525
!   B1 (1Pi)      14.454 14.229 14.221 14.228 14.219     14.127
!   A1 (1Sigma+)  17.316 17.232 17.220 17.231 17.219     17.217
!   A2 (1Delta)   17.689 16.820 16.790 16.819 16.789     16.833
!   Note the improvements in the EOMCCSD results by the
!   CR-EOMCCSD(T) approaches (e.g., ID/IB) for the Sigma+
!   state at 8.549 eV and both Delta states.
!
!   The ground state CCSD dipole is z=-0.645, and the
!   right/left transition moment to the first pi state
!   is x=0.297 and 0.320, with oscillator strength 0.0076
!
$contrl scftyp=rhf cctyp=cr-eom runtyp=energy
        icharg=1 units=bohr $end
$system mwords=5 $end
$ccinp  ncore=0 $end
$eominp nstate(1)=4,2,2,0 minit=1 noact=3 nuact=7
        ccprpe=.true. $end
$data
CH+ at R=2.13713...basis set from CPL 154, 380 (1989)
Cnv    2

Carbon  6.0    0.0 0.0 0.16558134
S      6
      1    4231.610      0.002029
      2     634.882      0.015535
      3     146.097      0.075411
      4      42.4974      0.257121
      5      14.1892      0.596555
      6       1.9666      0.242517
S      1 ; 1    5.1477    1.0
S      1 ; 1    0.4962    1.0
S      1 ; 1    0.1533    1.0
S      1 ; 1    0.0150    1.0
P      4
      1     18.1557      0.018534
      2      3.9864      0.115442

```



```

      3      1.1429      0.386206
      4      0.3594      0.640089
P      1 ; 1      0.1146      1.0
P      1 ; 1      0.011      1.0
D      1 ; 1      0.75      1.0

Hydrogen      1.0      0.0      0.0      -1.97154866
S      3
      1      1.924060D+01      3.282800D-02
      2      2.899200D+00      2.312080D-01
      3      6.534000D-01      8.172380D-01
S      1 ; 1      1.776D-01      1.0
S      1 ; 1      2.5D-02      1.0
P      1 ; 1      1.0      1.0
$end

```

resource requirements

User can perform LCCD, CCD, and CCSD calculations, that is without calculating the [T], (T), (2,3), and (TQ) corrections, or calculate the entire set of the standard and renormalized [T], (T), (2,3), and (TQ) ground-state corrections, in addition to the CCSD energies. User can also perform the EOMCCSD calculations of excited states and stop at EOMCCSD or continue to obtain some or all CR-EOMCCSD(T) triples corrections (cf. the values of input variable CCTYP in \$CONTRL and \$EOMINP group). Finally, user can perform the calculations of ground-state properties at the CCSD level or calculate ground- and excited-state properties. It is also possible to combine some of the above calculations. For example, one can calculate the CCSD and EOMCCSD properties and obtain triples corrections to the calculated CCSD and EOMCCSD energies from a single input (see the example above). The CR-CC(2,3) calculation produces the MBPT(2) and CCSD energies, and CCSD one-electron properties and density matrices, in addition to the CR-CC(2,3) and some other CR-CC triples corrections to the CCSD energies, again all from a single input (CCTYP=CR-CCL). The most expensive steps in CC/EOMCC calculations scale as follows:

LCCD, CCD, CCSD, EOMCCSD

No**2 times Nu**4 (iterative)

CCSD[T], CCSD(T), R-CCSD[T], R-CCSD(T), CR-CCSD[T],
CR-CCSD(T), CR-CC(2,3) (#1), CR-EOMCCSD(T) (#2)

No**3 times Nu**4 (non-iterative)
plus No**2 times Nu**4 (iterative)

CCSD(TQ), R-CCSD(TQ), CR-CCSD(TQ)

No**2 times Nu**5 or
 Nu**6 (#3) (non-iterative) plus
 No**3 times Nu**4 (non-iterative)
 plus No**2 times Nu**4 (iterative)

(#1) In addition to the usual No**2 times Nu**4 iterative CCSD steps and No**3 times Nu**4 non-iterative steps needed to determine the (2,3) triples correction, the CR-CC(2,3) calculations require extra No**2 times Nu**4 iterative steps needed to obtain the left CCSD state, which enters the CR-CC(2,3) triples correction formula.

(#2) In addition to the No**2 times Nu**4 iterative CCSD and EOMCCSD steps and No**3 times Nu**4 non-iterative (T) steps that are common to all CR-EOMCCSD(T) models, the CR-EOMCCSD(T),III method requires the iterative No**2 times Nu**4 steps of CISD. The CR-EOMCCSD(T),IX and CR-EOMCCSD(T),IIX (X=A-D) methods do not require these additional CISD calculations.

(#3) To reduce the cost, the program will automatically choose between the No**2 times Nu**5 and Nu**6 algorithms in the (Q) part, depending on the ratio of Nu to No.

The cost of calculating the standard CCSD[T] and CCSD(T) energies and the cost of calculating the R-CCSD[T] and R-CCSD(T) energies are essentially the same. The cost of calculating the triples corrections of the CR-CCSD[T] and CR-CCSD(T) approaches is essentially twice the cost of calculating the standard CCSD[T] and CCSD(T) corrections. Similar relationships hold between the costs of the CCSD(TQ), R-CCSD(TQ), and CR-CCSD(TQ) calculations. The cost of calculating the triples corrections of the CR-CC(2,3),X (X=A-D) approaches is also twice the cost of calculating the CCSD[T] and CCSD(T) triples corrections, but additional No**2 times Nu**4 iterative steps are required to generate the left CCSD state after converging the CCSD equations in order to calculate the final CR-CC(2,3) energies. Although the noniterative triples corrections may be seen to grow as the seventh power of the system size, they often require less time than the sixth power iterations of the CCSD step, while providing a great increase in accuracy. Similar remarks apply to the CR-EOMCCSD(T) calculations: The cost of the CR-EOMCCSD(T) calculation for a single electronic state, in its noniterative triples part, is twice the cost of computing

the standard (T) corrections of CCSD(T). The total CPU time of the CR-EOMCCSD(T) calculations scales linearly with the number of calculated states. In spite of the formal N^6 scaling, the calculations of the CCSD/EOMCCSD properties per single electronic state are considerably less expensive than the CCSD calculations for two reasons. First of all, the process of obtaining the left eigenstates of the similarity transformed Hamiltonian \bar{H} can reuse the intermediates (matrix elements of \bar{H}) which are obtained in the prior CCSD calculations. Second, converging left eigenstates of \bar{H} is usually much quicker than converging the CCSD equations when one obtains the left eigenstates of \bar{H} by solving the linear system of equations for the L deexcitation amplitudes after determining the R excitation amplitudes and excitation energies. This means that computing properties at the CCSD/EOMCCSD level is not very expensive once the CCSD and EOMCCSD right eigenvectors are obtained. Similar remarks apply to the CR-CC(2,3) calculations, which require the left CCSD eigenstates in addition to the CCSD T1 and T2 amplitudes: The determination of the left CCSD states that are needed to determine the non-iterative triples corrections of the CR-CC(2,3) approach makes the entire CCSD part of the CR-CC(2,3) calculation only somewhat more expensive than the regular CCSD iterations needed to obtain T1 and T2 clusters. The CCSD(TQ), R-CCSD(TQ), and CR-CCSD(TQ) calculations are more expensive than the CCSD(T) calculations, in spite of the fact that all of these methods use non-iterative N^7 steps. This is related to the fact that the N^2 times N^5 steps of the (TQ) methods are more expensive than the N^3 times N^4 steps of the (T) approaches. On the other hand, the CCSD(TQ), R-CCSD(TQ), and CR-CCSD(TQ) methods are much less expensive than the iterative ways of obtaining the information about quadruply excited clusters. This is a result of an efficient use of diagram factorization in coding the CCSD(TQ), R-CCSD(TQ), and CR-CCSD(TQ) methods, which leads to a reduction of the N^9 -type steps in the original (Q) expressions to N^7 steps.

Rough estimates of the memory required are:

CCSD $4 N^2 \text{ times } N^2 + N \text{ times } N^3$

CCSD[T], CCSD(T), R-CCSD[T], R-CCSD(T)
 $4 N^2 \text{ times } N^2 + N \text{ times } N^3$

CR-CCSD[T], CR-CCSD(T)
 $N^2 \text{ times } N^2 + 2 * N \text{ times } N^3$ (faster algorithm)

4 No**2 times Nu**2 + No times Nu**3 (slower, less memory)

CR-CC(2,3)

The most expensive routine requires $3 * \text{No} * \text{Nu}^{**3} + 3 * \text{Nu}^{**3} + 5 * \text{No}^{**2} * \text{Nu}^{**2}$ words

CCSD(TQ),b, R-CCSD(TQ)-n,x (n=1,2;x=a,b), CR-CCSD(TQ),x (x=a,b)

$2 * \text{No}$ times Nu**3 + No**2 times Nu**2 + Nu**3, preceded and followed by steps that require memories, such as, for example, $3 * \text{Nu}^{**3} + 5 * \text{No}^{**2} * \text{Nu}^{**2}$

EOMCCSD No times Nu**3 + 4 No**2 times Nu**2 (MEOM=0,1)
if MEOM=2, add to this
(4 times number of roots + 2) times No**2 times Nu**2

CR-EOMCCSD(T),IX, $2 * \text{No}$ times Nu**3 + 3 No**2 times Nu**2
CR-EOMCCSD(T),IIX(X=A-D) [MTRIP=1 in \$EOMINP]

CR-EOMCCSD(T) $3 * \text{No}$ times Nu**3 + 5 No**2 times Nu**2
all variants (faster algorithm) [MTRIP=2 in \$EOMINP]

CR-EOMCCSD(T),III $2 * \text{No}$ times Nu**3 + 5 No**2 times Nu**2
[MTRIP=3 in \$EOMINP]

CR-EOMCCSD(T) $2 * \text{No}$ times Nu**3 + 5 No**2 times Nu**2
all variants (slower algorithm) [MTRIP=4 in \$EOMINP]

The program automatically selects the algorithm for the CR-CCSD[T] and CR-CCSD(T) calculations, depending on the amount of available memory. A similar remark applies to the EOMCCSD calculations, where some additional reductions of memory requirements are possible if memory is low. The above estimates are rough.

The time required for calculating the CR-CCSD[T] and CR-CCSD(T) triples corrections is only twice the time used to calculate the standard CCSD[T] and CCSD(T) corrections. Thus, by just doubling the CPU time for the noniterative triples corrections and by selecting CCTYP=CR-CC, we gain access to all six noniterative triples corrections (the CCSD[T], CCSD(T), R-CCSD[T], R-CCSD(T), CR-CCSD[T], and CR-CCSD(T) energies) plus, of course, to the MBPT(2) and CCSD energies. At the same time, the CR-CCSD[T] and CR-CCSD(T) results for stretched nuclear geometries and diradicals are better than the results of the conventional CCSD[T] and CCSD(T) calculations. In some cases, choosing CCTYP=R-CC might be reasonable, too. The choice CCTYP=R-CC gives five different energies (CCSD, CCSD[T], CCSD(T), R-CCSD[T], and

R-CCSD(T)) for the price of three (CCSD, CCSD[T], and CCSD(T)) as there is no extra time needed for the R-theories compared to the standard ones. If we ignore the iterative CCSD steps and additional iterative steps needed to determine the left CCSD state, the time required for calculating the size extensive CR-CC(2,3) triples corrections is also only twice the time of calculating the CCSD[T] and CCSD(T) corrections. There is an additional bonus though: The CR-CC(2,3) calculations automatically produce a variety of CCSD one-electron properties at no extra cost. Similar remarks apply to quadruples and excited state calculations, although in the latter case a lot depends on user's expectations. If user is only interested in excited states dominated by singles and if accuracies on the order of 0.1-0.3 eV (sometimes better, sometimes worse) are acceptable, EOMCCSD is a good choice. However, it may be worth improving the EOMCCSD results by performing the CR-EOMCCSD(T) calculations, which often lower the errors in calculated excited states to 0.1 eV or less without making the calculations a lot more expensive (the CR-EOMCCSD(T) corrections are noniterative, so that the CPU time needed to calculate them may be comparable to the time spent in all EOMCCSD iterations). If there is a risk of encountering low-lying states having significant doubly excited contributions or multi-reference character, choosing CR-EOMCCSD(T) is a necessity, since errors obtained in EOMCCSD calculations for states dominated by doubles can easily be on the order of 1 eV. The CCSD(T) approach is often fine for closed-shell molecules, but there are cases, such as the vibrational frequencies of ozone and properties of other multiply bonded systems, where inclusion of quadruples is necessary. The CR-CCSD(T) approach is very useful in cases involving single bond breaking and diradicals, but CR-CC(2,3) and CR-CCSD(TQ) should be better. In addition, the CR-CC(2,3) method provides rigorously size extensive results. In cases of multiple bond dissociations, CR-CCSD(TQ) is a better alternative. The program is organized such that choosing a CR-CCSD(TQ) option (CCTYP=CR-CC(Q)) produces all energies obtained with CCTYP=CR-CCSD(T) and all CCSD(TQ), R-CCSD(TQ), and CR-CCSD(TQ) energies. By selecting CCTYP=CCSD(TQ), the user can obtain the CCSD(TQ) and R-CCSD(TQ) energies, in addition to the CCSD, CCSD[T], CCSD(T), R-CCSD[T], and R-CCSD(T) energies.

We encourage the user to read papers, such as
P. Piecuch, S.A. Kucharski, K. Kowalski, M. Musiał
Comput. Phys. Comm., 149, 71-96(2002);
K. Kowalski and P. Piecuch,

J. Chem. Phys., 120, 1715-1738 (2004);
M. Wloch, J.R. Gour, K. Kowalski, and P. Piecuch, J.
Chem. Phys. 122, 214107 (2005);
K. Kowalski, P. Piecuch, M. Wloch, S.A. Kucharski, M.
Musial, and M.W. Schmidt, in preparation,
where time and memory requirements for various types of CC
and EOMCC calculations are described in considerable
detail.

restarts in ground-state calculations

The CC code incorporated in GAMESS is quite good in converging the CCSD equations with the default guess for cluster amplitudes. The code is designed to converge in relatively few iterations for significantly stretched nuclear geometries, where it is not unusual to obtain large cluster amplitudes whose absolute values are close to 1. This is accomplished by combining the standard Jacobi algorithm with the DIIS extrapolation method of Pulay. The maximum number of amplitude vectors used in the DIIS extrapolation procedure is defined by the input variable MXDIIS. The default for MXDIIS is as follows:

MXDIIS = 5, for $5 < \text{No} * \text{Nu}$,
MXDIIS = 3, for $2 < \text{No} * \text{Nu} < 6$,
MXDIIS = 0, for $\text{No} * \text{Nu} < 3$.

Thus, in the vast majority of cases, the default value of MXDIIS is 5. However, for very small problems, when the DIIS expansion subspace leads to singular systems of linear equations, it is necessary to reduce the value of MXDIIS to 2-4 (we chose 3) or switch off DIIS altogether (which is the case when MXDIIS = 0).

It may, of course, happen that the solver for the CCSD equations does not converge, in spite of increasing the maximum number of iterations (input variable MAXCC; the default value is 30) and in spite of changing the default value of MXDIIS. In order to facilitate the calculations in all such cases, we included the restart option in the CC codes incorporated in GAMESS. Thus, user can restart a CCSD (or (L)CCD) calculation from the restart file created by an earlier CC calculation. In order to use the restart option, user must save the disk file CCREST (unit 70) from the previous CC run (cf. the GAMESS script run.gms) and make sure that this file is copied to scratch directory where the restarted calculation is carried out. A restart is invoked by entering a nonzero value for IREST, which should be the number of the last iteration completed, and must be

some value greater than or equal 3. Examples of using the restart option include the following situations:

- o The CCSD program did not converge in MAXCC iterations, but there is a chance to converge it if the value of MAXCC is increased. User restarts the calculation with the increased value of MAXCC.
- o User ran a CCSD calculation, obtaining the converged CCSD energy, but later decided to run CR-CCSD(T) or CR-CC(2,3) calculation. Instead of running the entire CCSD --> CR-CCSD(T) or CCSD --> CR-CC(2,3) task again, user restarts the calculation after changing the value of input variable CCTYP to CR-CC (the CR-CCSD(T) case) or CR-CCL (the CR-CC(2,3) case) and entering IREST to reuse the previous CCSD amplitudes, proceeding at once to the non-iterative triples corrections (left CCSD calculations and triples corrections in the CR-CC(2,3) case).
- o The CCSD program diverged for some geometry with a significantly stretched bond. User performs an extra calculation for a different nuclear geometry, for which it is easier to converge the CCSD equations, and restarts the calculation from the restart file generated by an extra calculation. This technique of restarting the CC calculations from the cluster amplitudes obtained for a neighboring nuclear geometry is particularly useful for scanning PESs and for calculating energy derivatives by numerical differentiation.

There also are situations where restart of the ground-state CCSD calculations is useful for excited-state and property calculations:

- o User ran a CCSD, CCSD(T), or CR-CCSD(T) calculation, obtaining the converged CC energies for the ground state, but later decided to run an excited-state EOMCCSD or CR-EOMCCSD(T) calculations. Instead of running the entire CCSD --> EOMCCSD or CCSD --> CR-EOMCCSD(T) task, user restarts the calculation after changing the value of input variable CCTYP to EOM-CCSD or CR-EOM, selecting excited-state options in \$EOMINP, and entering IREST greater or equal to 3 to reuse the previously converged CCSD amplitudes, proceeding at once to the excited-state (EOMCCSD or CR-EOMCCSD(T)) calculations.
- o User ran an EOMCCSD excited-state calculation, obtaining the converged CCSD amplitudes, but later discovered (by analyzing R1 and R2 amplitudes and REL values)

that some states are dominated by doubles, so that the EOMCCSD results need to be improved by the CR-EOMCCSD(T) triples corrections. Instead of running the entire CCSD --> CR-EOMCCSD(T) task, user restarts the calculation after changing the value of input variable CCTYP from EOM-CCSD to CR-EOM, and entering IREST greater or equal to 3 to reuse the previously converged CCSD amplitudes, proceeding at once to the EOMCCSD and CR-EOMCCSD(T) calculations.

o User ran a CR-CCSD(T) calculation, obtaining the converged ground-state energies, but later decided to run CCSD and EOMCCSD properties. Instead of running the CCSD --> EOMCCSD task again, user restarts the calculation after changing the value of input variable CCTYP to EOM-CCSD, adding CCPRPE=.TRUE. and the desired values of NSTATE in \$EOMINP, and entering IREST to reuse the previously converged CCSD amplitudes, proceeding at once to CCSD and EOMCCSD properties.

initial guesses in excited-state calculations

The EOMCCSD calculation is an iterative procedure which needs initial guesses for the excited states of interest. The popular initial guess for the EOMCCSD calculations is obtained by performing the CIS calculations (diagonalizing the Hamiltonian in a space of singles only). This is acceptable for states dominated by singles, but user may encounter severe convergence difficulties or even miss some states entirely if the calculated states have significant doubly excited character. One possible philosophy is not to worry about it and use the CIS initial guess only, since EOMCCSD fails to describe states with large doubly excited components. This is not the philosophy of the EOMCC programs in GAMESS. GAMESS is equipped with the CR-EOMCCSD(T) triples corrections to EOMCCSD energies, which are capable of reducing the large errors in the EOMCCSD results for states dominated by two-electron transitions, on the order of 1 eV, to 0.1 eV or even less. Thus, the ability to capture states with significant doubly excited contributions is an important element of the EOMCC GAMESS codes.

Excited states with significant contributions from double excitations can easily be found by using the EOMCCSD (little d) initial guesses provided by GAMESS. In the EOMCCSD calculations (and analogous CISd calculations used to initiate the CISD calculations for the CR-EOMCCSD(T), III

method), the initial guesses for the calculated excited states are defined using all single excitations (letter S in EOMCCSD and CISd) and a small subset of double excitations (the little d in EOMCCSD and CISd) defined by active orbitals or orbital range specified by the user. The inclusion of a small set of active double excitations in addition to all singles in the initial guess greatly facilitates finding excited states characterized by relatively large doubly excited amplitudes. GAMESS input offers a choice between the CIS and EOMCCSD/CISd initial guesses. The use of EOMCCSD/CISd initial guesses is highly recommended. This is accomplished by setting the input variable MINIT at 1 and by selecting the orbital range (active orbitals to define "little doubles" d) through the numbers of active occupied and active unoccupied orbitals (variables NOACT and NUACT, respectively) or an array of active orbitals called MOACT.

eigensolvers for excited-state calculations

The basic eigensolver for the EOMCCSD calculations is the Hirao and Nakatsuji's generalization of the Davidson diagonalization algorithm to non-Hermitian problems (the similarity transformed Hamiltonian \bar{H} is non-Hermitian). GAMESS offers the following three choices of EOMCCSD eigensolvers for the right eigenvalue problem (R amplitudes and energies only):

- o the true multi-root eigensolver based on the Hirao and Nakatsuji's algorithm, in which all states are calculated at once using a united iterative space (variable MEOM=2).
- o the single-root eigensolver, in which one calculates one state at a time, but the iterative subspace corresponding to all calculated roots remains united (variable MEOM=0).
- o the single-root eigensolver, in which one calculates one state at a time and each calculated root has a separate iterative subspace (variable MEOM=1).

The latter option (MEOM=1) leads to the fastest algorithm, but there is a risk (often worth taking) that some states will be converged more than once. The true multi-root eigensolver (MEOM=2) is probably the safest, but it is also the most expensive solver and there are some risks associated with using it too. When MEOM=2, there is a risk that one root, which is difficult to converge, may cause the entire multi-root procedure fail in spite of the fact that all other roots participating in the calculation

converged. The EOMCCSD program in GAMESS is prepared to handle this problem by saving individual roots that converged during multi-root iterations in case the entire procedure fails because of one or more roots which are difficult to converge. In this way, at least some roots are saved for the subsequent CR-EOMCCSD(T) calculations. The middle option (MEOM=0) seems to offer the best compromise. MEOM=0 is a single-root eigensolver, so there are no risks associated with losing some states during multi-root calculations. At the same time, the use of the united iterative subspace for all calculated roots helps to eliminate the problem of MEOM=1 of obtaining the same root more than once. The single-root eigensolver with a united iterative subspace (MEOM=0) is recommended (and used as a default), although other ways of converging the right EOMCCSD equations (MEOM=1,2) are very useful too.

As pointed out earlier, in order to calculate reduced density matrices and properties using CCSD and EOMCCSD methods, one has to determine the left as well as the right eigenstates of the non-Hermitian similarity transformed Hamiltonian \bar{H} . For the ground state, this is done by solving the linear system of equations for the deexcitation operator Λ (in the CCSD case, the one- and two-body components Λ_1 and Λ_2). For the amplitudes defining the L_1 and L_2 components of the excited-state operator L , one can proceed in several different ways and these different ways are reflected in the EOMCCSD algorithm incorporated in GAMESS. One can, for example, solve the linear system of equations for the amplitudes defining the EOMCCSD deexcitation operator $L=L_1+L_2$, after determining the corresponding excitation operator $R=R_1+R_2$ and excitation energy ω . This is a highly recommended option, which is also a default in GAMESS. This option is executed with any choice of MEOM=0,1,2 and when the user selects CPRPE=.TRUE. In case of unlikely difficulties with obtaining the L_1 and L_2 components, one can solve for the EOMCCSD values of the L_1, L_2 and R_1, R_2 amplitudes and excitation energies simultaneously in the process of diagonalizing the similarity transformed Hamiltonian \bar{H} completely in a single sequence of iterations. This approach is reflected by the following two additional choices of the input variable MEOM:

- o MEOM=3, one root at a time, separate iterative space for each computed root, left and right eigenvectors of the similarity transformed Hamiltonian and energies (like MEOM=1, but both left and right eigenvectors are iterated).
- o MEOM=4, one root at a time, united iterative spaces

for all calculated roots, left and right eigenvectors of the similarity transformed Hamiltonian and energies (like MEOM=0, but both left and right eigenvectors are iterated). In both cases, the user has to select CCPRPE=.TRUE. in order for these two choices of MEOM to work.

references and citations required in publications

Any publication describing the results of CC calculations obtained using GAMESS should give reference to the relevant papers. Depending on the specific CCTYP value, these are:

CCTYP = LCCD, CCD, CCSD, CCSD(T)

P. Piecuch, S.A. Kucharski, K. Kowalski, and M. Musiał
Comput. Phys. Commun. 149, 71-96 (2002).

CCTYP = R-CC, CR-CC, CCSD(TQ), CR-CC(Q)

P. Piecuch, S.A. Kucharski, K. Kowalski, and M. Musiał
Comput. Phys. Commun. 149, 71-96 (2002);

K. Kowalski and P. Piecuch
J. Chem. Phys. 113, 18-35 (2000);

K. Kowalski and P. Piecuch
J. Chem. Phys. 113, 5644-5652 (2000).

CCTYP = CR-CCL

P. Piecuch, S.A. Kucharski, K. Kowalski, and M. Musiał
Comput. Phys. Commun. 149, 71-96 (2002);

P. Piecuch and M. Włoch
J. Chem. Phys. 123, 224105/1-10 (2005).

CCTYP = EOM-CCSD, CR-EOM

P. Piecuch, S.A. Kucharski, K. Kowalski, and M. Musiał
Comput. Phys. Commun. 149, 71-96 (2002);

K. Kowalski and P. Piecuch,
J. Chem. Phys. 120, 1715-1738 (2004);

M. Włoch, J.R. Gour, K. Kowalski, and P. Piecuch,
J. Chem. Phys. 122, 214107-1 - 214107-15 (2005).

CCTYP = CR-EOML

P. Piecuch, J. R. Gour, and M. Włoch
Int. J. Quantum Chem. 109, 3268-3304(2009)

and the first two papers cited for CR-EOM just above

CCTYP = IP-EOM2, EA-EOM2

J. R. Gour, P. Piecuch, M. Włoch
J. Chem. Phys. 123, 134113/1-14(2005)

J. R. Gour, P. Piecuch

J. Chem. Phys. 125, 234107/1-17(2006)

In addition, the explicit use of CCPRP=.TRUE. in \$CCINP and/or the use of CCPRPE=.TRUE. in \$EOMINP should reference

M. Wloch, J.R. Gour, K. Kowalski, and P. Piecuch,
J. Chem. Phys. 122, 214107/1-15 (2005).

The rest of this section is a list of references to the original formulation of various areas in Coupled-Cluster Theory relevant to methods available in GAMESS:

Electronic structure:

- J. Cizek, J. Chem. Phys. 45, 4256 (1966).
- J. Cizek, Adv. Chem. Phys. 14, 35 (1969).
- J. Cizek, J. Paldus, Int.J.Quantum Chem. 5, 359 (1971).

Nuclear theory (examples):

- F. Coester, Nucl. Phys. 7, 421 (1958).
- F. Coester, H. Kuemmel, Nucl. Phys. 17, 477 (1960).
- K. Kowalski, D.J. Dean, M. Hjorth-Jensen, T. Papenbrock, P. Piecuch, Phys. Rev. Lett. 92, 132501 (2004).
- D.J. Dean, J.R. Gour, G. Hagen, M. Hjorth-Jensen, K. Kowalski, T. Papenbrock, P. Piecuch, M. Wloch, Nucl. Phys. A. 752, 299 (2005).
- M. Wloch, D.J. Dean, J.R. Gour, P. Piecuch, M. Hjorth-Jensen, T. Papenbrock, K. Kowalski, Eur. Phys. J. A 25 (Suppl. 1), 485 (2005).
- M. Wloch, J.R. Gour, P. Piecuch, D.J. Dean, M. Hjorth-Jensen, T. Papenbrock, J. Phys. G: Nucl. Phys. 31, S1291 (2005).
- M. Wloch, D.J. Dean, J.R. Gour, M. Hjorth-Jensen, K. Kowalski, T. Papenbrock, P. Piecuch, Phys. Rev. Lett. 94, 212501 (2005).
- P. Piecuch, M. Wloch, J.R. Gour, D.J. Dean, M. Hjorth-Jensen, T. Papenbrock, in V. Zelevinsky (Ed.), Nuclei and Mesoscopic Physics, AIP Conference Proceedings, Vol. 777 (AIP Press, 2005), p. 28.
- D.J. Dean, M. Hjorth-Jensen, K. Kowalski, T. Papenbrock, M. Wloch, and P. Piecuch, in Key Topics in Nuclear Structure, Proceedings of the 8th International Spring Seminar on Nuclear Physics, edited by A. Covello (World Scientific, Singapore, 2005), p. 147.

Coupled-Cluster Method with Doubles (CCD) -

- J. Cizek, J. Chem. Phys. 45, 4256 (1966).
- J. Cizek, Adv. Chem. Phys. 14, 35 (1969).
- J. Cizek, J. Paldus, Int.J.Quantum Chem. 5, 359 (1971).

- J.A. Pople, R. Krishnan, H.B. Schlegel, J.S. Binkley,
Int. J. Quantum Chem. Symp. 14, 545 (1978).
R.J. Bartlett and G.D. Purvis,
Int. J. Quantum Chem. Symp. 14, 561 (1978).
J. Paldus, J. Chem. Phys. 67, 303 (1977)
[orthogonally spin-adapted formulation].

Linearized Coupled-Cluster Method with Doubles (LCCD)

- J. Cizek, J. Chem. Phys. 45, 4256 (1966).
J. Cizek, Adv. Chem. Phys. 14, 35 (1969).
R.J. Bartlett, I. Shavitt, Chem. Phys. Lett. 50, 190 (1977)
57, 157 (1978) [Erratum].
R. Ahlrichs, Comp. Phys. Commun. 17, 31 (1979).

Coupled-Cluster Method with Singles and Doubles (CCSD) -

- G.D. Purvis III, R.J. Bartlett, J. Chem. Phys. 76, 1910 (1982)
[spin-orbital formulation].
P. Piecuch, J. Paldus, Int. J. Quantum Chem. 36, 429 (1989).
[orthogonally spin-adapted formulation].
G.E. Scuseria, A.C. Scheiner, T.J. Lee, J.E. Rice,
H.F. Schaefer III, J. Chem. Phys. 86, 2881 (1987)
[non-orthogonally spin-adapted formulation].
G.E. Scuseria, C.L. Janssen, H.F. Schaefer III
J. Chem. Phys. 89, 7382 (1988)
[non-orthogonally spin-adapted formulation].
T.J. Lee and J.E. Rice, Chem. Phys. Lett. 150, 406 (1988)
[non-orthogonally spin-adapted formulation].

Coupled-Cluster Method with Singles and Doubles and
Noniterative Triples, CCSD[T] = CCSD+T(CCSD) -

- M. Urban, J. Noga, S. J. Cole, and R. J. Bartlett,
J. Chem. Phys. 83, 4041 (1985).
P. Piecuch and J. Paldus, Theor. Chim. Acta 78, 65 (1990)
[orthogonally spin-adapted formulation].
P. Piecuch, S. Zarrabian, J. Paldus, and J. Cizek,
Phys. Rev. B 42, 3351-3379 (1990)
[orthogonally spin-adapted formulation].
P. Piecuch, R. Tobola, and J. Paldus,
Int. J. Quantum Chem. 55, 133-146 (1995)
[orthogonally spin-adapted formulation].

Coupled-Cluster Method with Singles and Doubles and
Noniterative Triples, CCSD(T) -

- K. Raghavachari, G. W. Trucks, J. A. Pople, M. Head-Gordon,
Chem. Phys. Lett. 157, 479 (1989).

Equation of Motion Coupled-Cluster Method, Response

- CC/Time Dependent CC Approaches, SAC-CI (Original Ideas), -
H. Monkhorst, Int. J. Quantum Chem. Symp. 11, 421 (1977).

- K. Emrich, Nucl. Phys. A 351, 379 (1981).
H. Sekino and R.J. Bartlett,
Int. J. Quantum Chem. Symp. 18, 255 (1984).
E. Daalgard and H. Monkhorst, Phys. Rev. A 28, 1217 (1983).
M. Takahashi and J. Paldus, J. Chem. Phys. 85, 1486 (1986).
H. Koch and P. Jorgensen, J. Chem. Phys. 93, 3333 (1990).
H. Nakatsuji, K. Hirao, Chem. Phys. Lett. 47, 569 (1977).
H. Nakatsuji, K. Hirao, J. Chem. Phys. 68, 2053, 4279 (1978).

Equation of Motion Coupled-Cluster Method with Singles and Doubles, EOMCCSD -

- J. Geertsen, M. Rittby, and R.J. Bartlett,
Chem. Phys. Lett. 164, 57 (1989).
J.F. Stanton and R.J. Bartlett,
J. Chem. Phys. 98, 7029 (1993).

Method of Moments of Coupled-Cluster Equations and Renormalized and Completely Renormalized Coupled-Cluster Methods (Overviews) -

- P. Piecuch, K. Kowalski, I.S.O. Pimienta, S.A. Kucharski,
in M.R. Hoffmann, K.G. Dyall (Eds.), Low-Lying
Potential Energy Surfaces, ACS Symposium Series, Vol.
828, Am. Chem. Society, Washington, D.C., 2002, p. 31
[ground and excited states].
P. Piecuch, K. Kowalski, I.S.O. Pimienta, M.J. McGuire,
Int. Rev. Phys. Chem. 21, 527 (2002)
[ground and excited states].
P. Piecuch, I.S.O. Pimienta, P.-F. Fan, K. Kowalski,
in J. Maruani, R. Lefebvre, E. Brandas (Eds.),
Progress in Theoretical Chemistry and Physics,
Vol. 12, Advanced Topics in Theoretical Chemical
Physics, Kluwer, Dordrecht, 2003, p. 119
[ground states].
P. Piecuch, K. Kowalski, I.S.O. Pimienta, P.-D. Fan,
M. Lodriguito, M.J. McGuire, S.A. Kucharski, T. Kus,
M. Musiał, Theor. Chem. Acc. 112, 349 (2004)
[ground and excited states].
P. Piecuch, M. Wloch, M. Lodriguito, and J.R. Gour, in S.
Wilson, J.-P. Julien, J. Maruani, E. Brandas, and
G. Delgado-Barrio (Eds.), Progress in Theoretical
Chemistry and Physics, Vol. 15, Recent Advances in the
Theory of Chemical and Physical Systems, Springer,
Berlin, 2006, p. XX, in press
[excited states].
P. Piecuch, I.S.O. Pimienta, P.-D. Fan, and K. Kowalski,
in A.K. Wilson (Ed.), Recent Progress in Electron
Correlation Methodology, ACS Symposium Series, Vol.
XXX, Am. Chem. Society, Washington, D.C., 2006, p. XX
[in press; ground states].

P.-D. Fan and P. Piecuch, Adv. Quantum Chem., in press (2006).

Renormalized and Completely Renormalized Coupled-Cluster Methods, Method of Moments of Coupled-Cluster Equations (Initial Original Papers, Ground States) -

P. Piecuch, K. Kowalski,
in J. Leszczynski (Ed.), Computational Chemistry:
Reviews of Current Trends, Vol. 5, World Scientific,
Singapore, 2000, p. 1.

K. Kowalski, P. Piecuch, J. Chem. Phys. 113, 18 (2000).

K. Kowalski, P. Piecuch, J. Chem. Phys. 113, 5644 (2000).

Biorthogonal Method of Moments of Coupled-Cluster Equations and Size Extensive Completely Renormalized Coupled-Cluster Singles, Doubles, and Non-iterative Triples Approach (CR-CC(2,3)=CR-CCSD(T)L; Initial Original Papers) -

P. Piecuch and M. Wloch, J. Chem. Phys. 123, 224105(2005).

P. Piecuch, M. Wloch, J.R. Gour, and A. Kinal, Chem. Phys. Lett. 418, 467-474 (2006).

Renormalized and Completely Renormalized Coupled-Cluster Methods, Method of Moments of Coupled-Cluster Equations (Other Original Papers, Higher-Order Methods, Ground-State Benchmarks) -

K. Kowalski, P. Piecuch, Chem. Phys. Lett. 344, 165 (2001).

P. Piecuch, S.A. Kucharski, K. Kowalski,
Chem. Phys. Lett. 344, 176 (2001).

P. Piecuch, S.A. Kucharski, V. Spirko, K. Kowalski,
J.Chem.Phys. 115, 5796 (2001).

P. Piecuch, K. Kowalski, and I.S.O. Pimienta,
Int. J. Mol. Sci. 3, 475 (2002).

M.J. McGuire, K. Kowalski, P. Piecuch,
J. Chem. Phys. 117, 3617 (2002).

P. Piecuch, S.A. Kucharski, K. Kowalski, M. Musial,
Comput. Phys. Comm., 149, 71 (2002).

I.S.O. Pimienta, K. Kowalski, and P. Piecuch,
J. Chem. Phys. 119, 2951 (2003).

S. Hirata, P.-D. Fan, A.A. Auer, M. Nooijen, P. Piecuch,
J. Chem. Phys. 121, 12197 (2004).

K. Kowalski and P. Piecuch, J. Chem. Phys. 122, 074107 (2005).

P.-D. Fan, K. Kowalski, and P. Piecuch, Mol. Phys. 103, 2191 (2005).

Completely Renormalized Coupled-Cluster Methods, Examples of Large-Scale Applications to Ground-State Properties -

I. Ozkan, A. Kinal, M. Balci,
J.Phys.Chem. A 108, 507 (2004).

- R.L. DeKock, M.J. McGuire, P. Piecuch, W.D. Allen,
H.F. Schaefer III, K. Kowalski, S.A. Kucharski, M. Musiał,
A.R. Bonner, S.A. Spronk, D.B. Lawson, S.L. Laursen,
J. Phys. Chem. A 108, 2893 (2004).
M.J. McGuire, P. Piecuch, K. Kowalski, S.A. Kucharski,
M. Musiał, J. Phys. Chem. A 108, 8878 (2004).
M.J. McGuire, P. Piecuch
J. Am. Chem. Soc. 127, 2608 (2005).
A. Kinal, P. Piecuch, J. Phys. Chem. A 110, 367 (2006).
C.J. Cramer, M. Wloch, P. Piecuch, C. Puzzarini, and L.
Gagliardi, J. Phys. Chem. A 110, 1991 (2006).

Completely Renormalized Equation of Motion Coupled-Cluster
Methods, Method of Moments of Coupled-Cluster Equations
for Ground and Excited States (Original Papers) -

- K. Kowalski P. Piecuch, J. Chem. Phys. 115, 2966 (2001).
K. Kowalski P. Piecuch, J. Chem. Phys. 116, 7411 (2002).
K. Kowalski P. Piecuch, J. Chem. Phys. 120, 1715 (2004).
M. Wloch, J.R. Gour, K. Kowalski, and P. Piecuch, J. Chem.
Phys. 122, 214107 (2005).

Also, multi-reference and other externally corrected MMCC
methods including ground and excited states,

- K. Kowalski and P. Piecuch,
J. Molec. Struct.: THEOCHEM 547, 191 (2001).
K. Kowalski and P. Piecuch, Mol. Phys. 102, 2425 (2004).
M.D. Lodriguito, K. Kowalski, M. Wloch, and P. Piecuch
J. Mol. Struct: THEOCHEM, in press (2006).

Completely Renormalized Equation of Motion Coupled-Cluster
Methods, Method of Moments of Coupled-Cluster Equations
for Ground and Excited States (Selected Benchmarks and
Applications)

- C.D. Sherrill, P. Piecuch, J.Chem.Phys. 122, 124104 (2005)
R.K. Chaudhuri, K.F. Freed, G. Hose, P. Piecuch,
K. Kowalski, M. Wloch, S. Chattopadhyay, D. Mukherjee,
Z. Rolik, A. Szabados, G. Toth, and P.R. Surjan,
J. Chem. Phys. 122, 134105-1 (2005).
K. Kowalski, S. Hirata, M. Wloch, P. Piecuch, and T.L.
Windus, J. Chem. Phys. 123, 074319 (2005).
S. Nangia, D.G. Truhlar, M.J. McGuire, and P. Piecuch
J. Phys. Chem. A 109, 11643 (2005).
P. Piecuch, S. Hirata, K. Kowalski, P.-D. Fan, and T.L.
Windus, Int. J. Quantum Chem. 106, 79 (2006).
M. Wloch, M.D. Lodriguito, P. Piecuch, and J.R. Gour
Mol. Phys., in press (2006).
S. Coussan, Y. Ferro, A. Trivella, P. Roubin, R. Wieczorek,
C. Manca, P. Piecuch, K. Kowalski, M. Wloch, S.A.
Kucharski, and M. Musiał, J. Phys. Chem. A, in press
(2006).

Completely Renormalized Coupled-Cluster and Equation of Motion Coupled-Cluster Methods, GAMESS Implementations - P. Piecuch, S.A. Kucharski, K. Kowalski, M. Musial, Comput. Phys. Comm., 149, 71 (2002).
K. Kowalski and P. Piecuch J. Chem. Phys. 120, 1715 (2004).
M. Wloch, J.R. Gour, K. Kowalski, and P. Piecuch, J. Chem. Phys. 122, 214107 (2005).
P. Piecuch and M. Wloch, J. Chem. Phys. 123, 224105 (2005).
K. Kowalski, P. Piecuch, M. Wloch, S.A. Kucharski, M. Musial, and M.W. Schmidt, in preparation.

T1 diagnostic:

T.J.Lee, P.R.Taylor Int.J.Quantum Chem., S23, 199-207(1989).

It is often assumed that $T1 > 0.02$ indicates that CCSD may not be correct for a system which is not very single reference in nature. (T) corrections tolerate greater singles amplitudes. However, T1 diagnostic is in many cases misleading, since one can easily have small (or even vanishing) T1 cluster amplitudes due to symmetry and a significant configurational quasi-degeneracy and multi-reference character. In general, in typical multi-reference situations, such as bond stretching and diradicals, one observes a significant increase of T2 cluster amplitudes. The larger values of T2 amplitudes are a clear signature of a multi-reference character of the wave function. The CR-CCSD(T), CR-CCSD(TQ), and CR-CC(2,3) methods tolerate significant increases of T2 amplitudes in cases of single-bond breaking and diradicals. CCSD(T) and CCSD(TQ) approaches cannot do this, when the spin-adapted RHF references are employed.

Written by Piotr Piecuch, Michigan State University
(updated March 18, 2006)

Density Functional Theory

There are actually two DFT programs in GAMESS, one using the typical grid quadrature for integration of functionals, and one using resolution of the identity to avoid the need for grids. The default METHOD=GRID program is discussed below, following a short description of METHOD=GRIDFREE. The final section is references to various functionals, and other topics of interest.

DFTTYP keywords

Let's begin with a translation table to NWchem's input:

GAMESS	NWchem's XC keyword
Slater	slater
Gill	gill96
SVWN	slater vwn_5 (SVWN1RPA=slater vwn_1_rpa, etc)
Becke	becke88
BVWN	becke88 vwn_5
BLYP	becke88 lyp
B97	becke97
B97-1,B97-2,B97-3	becke97-1, becke-2, becke-3
HCTH93,HCTH120,HCTH147,HCTH407	hcth,hcth120,hcth147,hcth407
B98	becke98
B3LYP	HFexch 0.20 slater 0.80 \ becke88 nonlocal 0.72 \ lyp 0.81 vwn_5 0.19
B3LYPV1R	b3lyp or, if you like to type: HFexch 0.20 slater 0.80 \ becke88 nonlocal 0.72 \ lyp 0.81 vwn_1_rpa 0.19
B3P86	HFexch 0.20 slater 0.80 \ becke88 nonlocal 0.72 \ vwn_5 0.19 perdew81 0.81 perdew86 0.81
X3LYP	HFexch 0.218 slater 0.782 \ becke88 nonlocal 0.542 \ xperdew91 nonlocal 0.167 \ lyp 0.871 vwn_1_rpa 0.129
PW91	xperdew91 perdew91
B3PW91	HFexch 0.20 slater 0.80 \ becke88 nonlocal 0.72 \ perdew91 0.81 pw91lda 1.00

PBE	xpbe96 cpbe96
PBE0	pbe0
revPBE	revpbe cpbe96
VS98	vs98
M06	m06 (and similarly for M05-2X, etc.)
PKZB	xpkzb99 cpkzb99
TPSS	xtpss03 cptss03
TPSSh	xctpssh

Note that B3LYP in GAMESS is based in part on the VWN5 electron gas correlation functional. Since there are five formulae with two possible parameterizations mentioned in the VWN paper about local correlation, other programs may use other choices, and therefore generate different B3LYP energies. For example, NWChem's manual says it uses the "VWN 1 functional with RPA parameters as opposed to the prescribed Monte Carlo parameters" as its default. Should you wish to use this parameterization of the VWN1 formula in a B3LYP hybrid, simply choose "DFTTYP=B3LYPV1R".

grid-free DFT

The grid-free code is a research tool into the use of the resolution of the identity to simplify evaluation of integrals over functionals, rather than quadrature grids. This trades the use of finite grids and their associated errors for the use of a finite basis set used to expand the identity, with an associated truncation error. The present choice of auxiliary basis sets was obtained by tests on small 2nd row molecules like NH₃ and N₂, and hence the built in bases for the 3rd row are not as well developed. Auxiliary bases for the remaining elements do not exist at the present time.

The grid-free Becke/6-31G(d) energy at a C1 AM1 geometry for ethanol is -154.084592, while the result from a run using the "army grade grid" is -154.105052. So, the error using the AUX3 RI basis is about 5 milliHartree per 2nd row atom (the H's must account for some of the error too). The energy values are probably OK, the differences noted should by and large cancel when comparing different geometries.

The grid-free gradient code contains some numerical inaccuracies, possibly due to the manner in which the RI is implemented for the gradient. Computed gradients consequently may not be very reliable. For example, a Becke/6-31G(d) geometry optimization of water started from the EXAM08 geometry behaves as:

```

FINAL E= -76.0439853638, RMS GRADIENT = .0200293
FINAL E= -76.0413274662, RMS GRADIENT = .0231574
FINAL E= -76.0455283912, RMS GRADIENT = .0045887
FINAL E= -76.0457360477, RMS GRADIENT = .0009356
FINAL E= -76.0457239113, RMS GRADIENT = .0001222
FINAL E= -76.0457216186, RMS GRADIENT = .0000577
FINAL E= -76.0457202264, RMS GRADIENT = .0000018
FINAL E= -76.0457202253, RMS GRADIENT = .0000001

```

Examination shows that the point on the PES where the gradient is zero is not where the energy is lowest, in fact the 4th geometry is the lowest encountered.

The behavior for Becke/6-31G(d) ethanol is as follows:

```

FINAL E= -154.0845920132, RMS GRADIENT = .0135540
FINAL E= -154.0933138447, RMS GRADIENT = .0052778
FINAL E= -154.0885472996, RMS GRADIENT = .0009306
FINAL E= -154.0886268185, RMS GRADIENT = .0002043
FINAL E= -154.0886352947, RMS GRADIENT = .0000795
FINAL E= -154.0885599794, RMS GRADIENT = .0000342
FINAL E= -154.0885514829, RMS GRADIENT = .0000679
FINAL E= -154.0884955093, RMS GRADIENT = .0000205
FINAL E= -154.0886438244, RMS GRADIENT = .0000330
FINAL E= -154.0886596883, RMS GRADIENT = .0000325
FINAL E= -154.0886094081, RMS GRADIENT = .0000120
FINAL E= -154.0886054003, RMS GRADIENT = .0000109
FINAL E= -154.0885939751, RMS GRADIENT = .0000152
FINAL E= -154.0886711482, RMS GRADIENT = .0000439
FINAL E= -154.0886972557, RMS GRADIENT = .0000230

```

with similar fluctuations through a total of 50 steps without locating a zero gradient. Note that the second energy above is substantially below all later points, so geometry optimizations with the grid-free DFT gradient code are at this time unsatisfactory.

DFT with grids

METHOD=GRID (the default for DFT) produces good energy and gradient quantities. Its energy errors should usually be less than 10 microHartree/atom, using the default grid.

The default grid was changed on April 11, 2008 to use Lebedev angular grids. This changes all results obtained prior to that date using the original polar coordinate angular grid. The old grids can still be used,

```
$dft nrad=96 nthe=12 nphi=24 $end
```

```
$tdft nrad=24 nthe=8 nphi=16 $end
```

in case you need to reproduce numbers from older versions. Since April 2008, the default is

```
$dft nrad=96 nleb=302 $end
```

```
$tdft nrad=48 nleb=110 $end
```

The default for the more accurate meta-GGA functionals was changed in February 2012 to a more accurate grid

```
$dft nrad=99 nleb=590 $end
```

```
$tdft nrad=48 nleb=110 $end
```

but all GGA and LDA functionals remain 96/302 and 48/110. The 96/302 grid settings produce root mean square gradient vectors accurate to about 0.00010, which matches the default value for OPTTOL in \$STATPT. The "standard grid-one" contains many fewer points,

```
$dft sgl=.true. $end
```

```
$tdft sgl=.true. $end
```

so SG1 will produce nuclear gradients accurate only to about 5 times OPTTOL, namely 0.00050 or so. SG1 is a very fast grid, and will provide substantial speedups if SG1 is used for the early steps of geometry optimizations. Rather high quality results, meaning an OPTTOL near 0.00001 can be used, may be obtained by

```
$dft nrad=96 nleb=590 $end
```

Very accurate (converged) results come from using the "army grade" grid,

```
$dft nrad=96 nleb=1202 $end
```

Turn to the next page to see numerical results.

A numerical demonstration of grid accuracies can be obtained from ethanol, DFTTYP=BECKE:

	energy	RMS grad.	CPU
sgl=.true.	-154.105070	0.010837	11
nrad=96 nthe=12 nphi=24	-154.104863	0.010724	56
nrad=96 nleb=302	-154.105042	0.010704	58
nrad=96 nleb=590	-154.105051	0.0107349	108
nrad=96 nleb=1202	-154.105052	0.0107353	214

Note that the energies are a function of the grid size, just as they are a function of the basis used, so you must only compare runs that use the same grid size (and of course the same basis set). The default grid (and the 590 point grid) will give nuclear gradients which are accurate enough to lead to satisfactory geometry optimizations. This means that DFT frequencies obtained by numerical differentiation should also be OK. RUNTYP=ENERGY, GRADIENT, HESSIAN, and their chemical combinations for OPTIMIZE, SADPOINT, IRC, DRC, VSCF, RAMAN, and FFIELD should all work.

The grid DFT program uses symmetry during the numerical quadrature in two ways. First, the integration runs only

over grid points placed around the symmetry unique atoms. Your run should be done in the full non-Abelian group, so that grid points as well as the usual integrals and the SCF steps can exploit full molecular symmetry. Symmetry is turned off during any TD-DFT stages, since excited states often have different symmetry than the ground state, but will be used in the ground state DFT.

Secondly, for polar coordinate angular grids only, "octant symmetry" is implemented using an appropriate Abelian subgroup of the full group. The grid evaluation automatically uses an appropriate subgroup to reduce the number of grid points for atoms that lie on symmetry axes or planes. For example, in Cs, atoms lying in the xy plane will be integrated only over the upper hemisphere of their grid points. Octant symmetry is not used for any of these:

- a) if a non-standard axis orientation is input in \$DATA
- b) if the angular grid size (NTHE,NTHE0,NPHI,NPIO) is not a multiple of the octant symmetry factors, such as NTHE=15 in C2v. The permissible values depend on the group, but NTHE a multiple of 2 and NPHI a multiple of 4 is generally safe.

Time Dependent Density Functional Theory (TD-DFT)

Two review articles are available,

"Single-Reference ab Initio Methods for the Calculation of Excited States of Large Molecules"

A.Dreuw, M.Head-Gordon
Chem.Rev. 105, 4009-4037(2005)

"Excited states from time-dependent density functional theory"

P.Elliott, F.Furche, K.Burke
Rev.Comp.Chem. 26, 91-166(2009)

The following article is very informative:

S.Hirata, M.Head-Gordon
Chem.Phys.Lett. 314, 291-299(1999)

It also explains the Tamm/Dancoff approximation which connects TD-DFT to CIS.

TD-DFT requires higher functional derivatives of the exchange correlation energy with respect to the density: 2nd derivatives to do TD-DFT excitation energies, and 3rd derivatives to do TD-DFT nuclear gradients. Consequently,

some of the functionals permit only excitation energies. To use metaGGAs in TD-DFT, the above functional derivatives involve a non-trivial differentiation of the kinetic energy τ 's density dependence. The latter is the subject of

F.Zahariev, S.S.Leang, M.S.Gordon
J.Chem.Phys. 138, 244108/1-11(2013)

The TD-DFT nuclear gradient implementation in GAMESS is
M.Chiba, T.Tsuneda, K.Hirao
J.Chem.Phys. 124, 144106/1-11(2006)
and the long-range correction (useful in Rydberg and/or charge transfer states is

Y.Tawada, T.Tsuneda, S.Yanagisawa, Y.Yanai, K.Hirao
J.Chem.Phys. 120, 8425-8433(2004)

See also

K.A.Nguyen, P.N.Day, R.Pachter
Int.J.Quantum Chem. 110, 2247-2255(2010)

The "lambda diagnostic" is described by

M.J.G.Peach, P.Benfield, T.Helgaker, D.J.Tozer
J.Chem.Phys. 128, 044118/1-8(2008)

This is a criterion for separating valence states from charge transfer and Rydberg states.

Note that it is possible to do TD-HF excitation energies, by requesting TDDFT=EXCITE, but leaving DFTTYP=NONE.

* * * * *

Two-photon absorption (TPA) cross-sections are computed by first evaluating the excitation energy to some desired number of excited states. The oscillator strengths for each state give some idea of the intensity of one-photon absorption (OPA cross-section). Then, the TPA cross-sections to these same excited states are evaluated. Note that Franck-Condon factors are not included in either OPA or TPA oscillator strengths.

Beta hyperpolarizabilities (see BETA in \$TDDFT) are computed by the TDDFT code, but the computation of excited states is skipped.

TPA and BETA calculations

- a) should not use the Tamm/Dancoff approximation,
- b) may include solvent effects by EFP,
- c) make sense only at a single geometry: use the ordinary TD-DFT program to optimize excited state geometries, if needed.

TPA and BETA calculations are described in
 F.Zahariev, M.S.Gordon
 J.Chem.Phys. 140, 18A523/1-10(2014)

* * * * *

Solvation effects on the excited state energies can be modeled by PCM or EFP or both, with nuclear gradients.

PCM + TD-DFT gradient:

Y.Wang, H.Li J.Chem.Phys. 133, 034108/1-11(2010)

EFP1 + TD-DFT energy:

S.Yoo, F.Zahariev, S.Sok, M.S.Gordon

J.Chem.Phys. 129, 144112/1-8(2008)

EFP1 + TD-DFT gradient:

N.Minezawa, N.De Silva, F.Zahariev, M.S.Gordon

J.Chem.Phys. 134, 05411(2011)

POL5P + TD-DFT gradient (similar polarizable solvent):

D.Si, H.Li J.Chem.Phys. 133, 144112(2010)

* * * * *

In some cases, a more balanced description of the states might be obtained if the orbitals are optimized for a reference with unpaired electrons. This is possible with spin-flip methods, see TDDFT=SPNFLP. For example, in C₂H₄, one might optimize the orbitals for the triplet state $(\pi)1(\pi^*)1$, but be interested in the energies of the three singlets and one triplet states $N=(\pi)2$, $T=(\pi)1(\pi^*)1$, $V=(\pi)1(\pi^*)1$, and $Z=(\pi^*)2$. Using the T state as the reference optimizes the shape of both π and π^* , since both are occupied. Flipping one of the two unpaired alpha spins in the T reference will access all four valence states (recall that a triplet state with $M_s=0$ is perfectly OK, namely $ab+ba$). For more information, see

Y.Shao, M.Head-Gordon, A.I.Krylov

J.Chem.Phys. 118,4807(2003)

F.Wang, T.Ziegler

J.Chem.Phys. 121, 12191(2004)

J.Chem.Phys. 122, 074109(2005)

O.Vahtras, Z.Rinkevicius

J.Chem.Phys. 126, 114101(2007)

Z.Rinkevicius, H.Agren

Chem.Phys.Lett. 491, 132(2010)

Z.Rinkevicius, O.Vahtras, H.Agren

J.Chem.Phys. 113, 114101(2010)

M.Huix-Rotllant, B.Natarajan, A.Ipatov, C.M.Wawire,

T.Deutsch, M.E.Casida

Phys.Chem.Chem.Phys. 12, 12811(2010)

The penalty constrained optimization procedure was used to find ethylene's conical intersections by

N.Minezawa, M.S.Gordon

J.Phys.Chem.A 113, 12749(2009)

references for DFT

An excellent overview of DFT can be found in Chapter 6 of Frank Jensen's book. Two other monographs are

"Density Functional Theory of Atoms and Molecules"

R.G.Parr, W.Yang Oxford Scientific, 1989

"A Chemist's Guide to Density Functional Theory"

W.Koch, M.C.Holthausen Wiley-VCH, 2001

If you would like to understand the "theory" of Density Functional Theory, see Kieron Burke's online book "The ABC of DFT", at <http://dft.uci.edu/dftbook.html>

A delightful and thought provoking paper on the relationship of DFT to conventional wavefunction theory:

"Obituary: Density Functional Theory (1927-1993)"

P.M.W.Gill Aust.J.Chem. 54, 661-662(2001)

You may also enjoy

"Fourteen easy lessons in Density Functional Theory",

John Perdew and Adrienn Ruzsinszky

Int. J. Quantum Chem. 110, 2801-2807(2010)

"Perspective on density functional theory"

Kieron Burke J.Chem.Phys. 136, 150901/1-9(2012)

"Perspective: fifty years of density-functional theory in chemical physics"

A.D.Becke J.Chem.Phys. 140, 18A301/1-18(2014)

A paper comparing DFT's approach to correlation to traditional quantum chemistry methods:

E.J.Baerends, O.V.Gritsenko

J.Phys.Chem.A 101, 5383-5403(1997)

Some philosophy about designing functionals at each rung of DFT's "Jacob's ladder":

J.P.Perdew, A.Ruzsinszky, J.Tao, V.N.Staroverov,

G.E.Scuseria, G.I.Csonka

J.Chem.Phys. 123, 062201/1-9(2005)

On hybridization:

J.P.Perdew, M.Ernzerhof, K.Burke

J.Chem.Phys. 105, 9982-9985(1996)

G.I.Csonka, J.P.Perdew, A.Ruzsinszky

J.Chem.Theory Comput. 6, 3688-3703(2010)

Some reading on the grid-free approach to density functional theory is:

- Y.C.Zheng, J.Almlof
Chem.Phys.Lett. 214, 397-401(1996)
Y.C.Zheng, J.Almlof
J.Mol.Struct.(Theochem) 288, 277(1996)
K.Glaesemann, M.S.Gordon
J.Chem.Phys. 108, 9959-9969(1998)
K.Glaesemann, M.S.Gordon
J.Chem.Phys. 110, 6580-6582(1999)
K.Glaesemann, M.S.Gordon
J.Chem.Phys. 112, 10738-10745(2000)

References about gridding:

- A.D.Becke
J.Chem.Phys. 88, 2547-2553(1988)
C.W.Murray, N.C.Handy, G.L.Laming
Mol.Phys. 78, 997-1014(1993)
P.M.W.Gill, B.G.Johnson, J.A.Pople
Chem.Phys.Lett. 209, 506-512(1993)
A.A.Jarecki, E.R.Davidson
Chem.Phys.Lett. 300, 44-52(1999)
R.Lindh, P.-A.Malmqvist, L.Gagliardi
Theoret.Chem.Acc. 106, 178-187(2001)
S.-H.Chien, P.M.W.Gill
J.Comput.Chem. 27, 730-739(2006)
J.Grafenstein, D.Izotov, D.Cremer
J.Chem.Phys. 127, 164113/1-7(2007)
Gill's 1993 paper is the reference for SG1=.TRUE.
Handy's 1993 paper is a reference for polar coordinates.
Lebedev grids may be referenced as
V.I.Lebedev, D.N.Laikov Doklady Math. 59, 477-481(1999)
GAMESS uses Christoph van Wuelen's FORTRAN translation of
these grids, originally coded in C by Laikov (www.ccl.net).

--- exchange functionals

Slater exchange:

- J.C.Slater Phys.Rev. 81, 385-390(1951)
XALPHA is Slater with $\alpha=0.70$

BECKE (often called B88) exchange:

- A.D.Becke Phys.Rev. A38, 3098-3100(1988)

GILL (often called G96) exchange:

- P.M.W.Gill Mol.Phys. 89, 433-445(1996)

OPTX exchange:

N.C.Handy, A.J.Cohen Mol.Phys. 99, 403-412(2001)

Depristo/Kress exchange:

A.E.DePristo, J.E.Kress J.Chem.Phys. 86, 1425-1428(1987)

--- correlation functionals

VWN local correlation:

S.H.Vosko, L.Wilk, M.Nusair

Can.J.Phys. 58, 1200-1211(1980)

This paper has five formulae in it, and since the 5th is a good quality fit, it states "since formula 5 is easiest to implement in LSDA calculations, we recommend its use".

PZ81 correlation:

J.P.Perdew, A.Zunger Phys.Rev.B 23, 5048-5079(1981)

P86 GGA correlation:

J.P.Perdew Phys.Rev.B 33, 8822(1986)

PW local correlation (used in PW91):

J.P.Perdew, Y.Wang Phys.Rev.B 45, 13244-13249(1992)

LYP correlation:

C.Lee, W.Yang, R.G.Parr Phys.Rev. B37, 785-789(1988)

For practical purposes this is always used in a transformed way, involving the square of the density gradient:

B.Miehlich, A.Savin, H.Stoll, H.Preuss

Chem.Phys.Lett. 157, 200-206(1989)

OP (One-parameter Progressive) correlation:

T.Tsuneda, K.Hirao Chem.Phys.Lett. 268, 510-520(1997)

T.Tsuneda, T.Suzumura, K.Hirao

J.Chem.Phys. 110, 10664-10678(1999)

--- exchange/correlation functionals

PW91 exchange/correlation:

J.P.Perdew, J.A.Chevray, S.H.Vosko, K.A.Jackson,

M.R.Pederson, D.J.Singh, C.Fiolhais

Phys.Rev. B46, 6671-6687(1992)

EDF1 - empirical density functional #1, a tweaked BLYP developed for use with 6-31+G(d) basis sets,

R.D.Adamson, P.M.W.Gill, J.A.Pople

Chem.Phys.Lett. 284, 6-11(1998)

MOHLYP - metal optimized OPTX exchange,
half LYP correlation

N.E.Schultz, Y.Zhao, D.G.Truhlar
J.Phys.Chem.A 109, 11127-11143(2005)
See also comp.chem.umn.edu/info/MOHLYP_reference.pdf for
information about the related functional MOHLYP2.

PBE exchange/correlation functional:
J.P.Perdew, K.Burke, M.Ernzerhof
Phys.Rev.Lett. 77, 3865-8(1996); Err. 78,1396(1997)

revPBE (revised PBE exchange, but see RPBE below):
Y.Zhang, W.Yang Phys.Rev.Lett. 80, 890(1998)

RPBE (a different revision of PBE exchange):
B.Hammer, L.B.Hansen, J.K.Norskov
Phys.Rev.B 59, 7413-7421(1999)
This revision retains the same increase in accuracy for
atomization energies that revPBE affords, while rigorously
preserving the correct Lieb-Oxford limit, unlike revPBE.

PBEsol (modified PBE parameters, for solid properties):
J.P.Perdew, A.Ruzsinszky, G.I.Csonka, O.A.Vydrov,
G.E.Scuseria, L.A.Constantin, Z.Zhou, K.Burke
Phys.Rev.Lett. 100, 136406/1-7(2008)

The next two occur in the grid-free program only,

various WIGNER exchange/correlation functionals:
Q.Zhao, R.G.Parr Phys.Rev. A46, 5320-5323(1992)

CAMA/CAMB exchange/correlation functionals:
G.J.Laming, V.Termath, N.C.Handy
J.Chem.Phys. 99, 8765-8773(1993)

--- dispersion corrections:

dispersionless Density Functional Theory (d1DF)
K.Pernal, R.Podeszwa, K.Patkowski, K.Szalewicz
Phys.Rev.Lett. 103, 263201/1-4(2009)

This approach recognizes that density functionals may be
optimized to reproduce interaction energies from which the
dispersion energy has been subtracted. d1DF adjusts the
M05-2X parameterization to accomplish this for a training
set. A -D correction specific to d1DF was developed (see
the paper's supplementary material) to address the now
cleanly separated dispersion energy. The d1df-D correction
term is available in the form of a Python script at the

Szalewicz web site. Usage of d1DF by itself is not sensible.

Local Response Dispersion (LRD)

T.Sato, H.Nakai J.Chem.Phys. 131, 224104/1-12(2009)

T.Sato, H.Nakai J.Chem.Phys. 132, 194101/1-9(2010)

This computes dispersion energies using C6/C8 parameters evaluated from the final electron density of the molecule's DFT calculation.

empirical dispersion correction (DC):

This is developed in three successive versions by Grimme

1: S.Grimme J.Comput.Chem. 25, 1463-1473(2004)

2: S.Grimme J.Comput.Chem. 27, 1787-1799(2006)

3: S.Grimme, J.Antony, S.Ehrlich, H.Krieg

J.Chem.Phys. 132, 154104/1-19(2010)

which are applied to different functionals with different parameterizations of the correction. Setting DC=.TRUE. thus converts functionals such as BLYP/B3LYP/PBE/BP86/TPSS to BLYP-D, B3LYP-D, and so forth. See the papers for more details.

A functional where the input keyword contains already the -D, namely B97-D, consists of a revamping of the B97 functional to remove its hybridization with HF exchange and reparameterization, as well as adding the dispersion correction:

S.Grimme J.Comput.Chem. 27, 1787-1799(2006)

A somewhat different form for the dispersion correction is used in the wB97-D functional. Selection of DFTTYP=B97-D or wB97-D does not require setting DC on.

--- hybrids with HF exchange

B3PW91 hybrid:

A.D.Becke J.Chem.Phys. 98, 5648-5642(1993)

B3LYP hybrid:

A.D.Becke J.Chem.Phys. 98, 5648-5642(1993)

P.J.Stephens, F.J.Devlin, C.F.Chablowski, M.J.Frisch

J.Phys.Chem. 98, 11623-11627(1994)

R.H.Hertwig, W.Koch Chem.Phys.Lett. 268, 345-351(1997)

The first paper is actually on B3PW91 hybridization, and optimizes the mixing of five functionals with PW91 as the correlation GGA. The second paper then proposed use of LYP in place of PW91, without reoptimizing the mixing ratios of the hybrid. The final paper discusses the controversy

surrounding which VWN functional is used in the hybrid. GAMESS uses VWN5 in its B3LYP hybrid, but see also B3LYPV1R to use the RPA parameterized VWN1 formula.

B97 hybrid:

A.D.Becke J.Chem.Phys. 107, 8554-8560(1997)

B97-1 hybrid, a reparameterization of B97:

F.A.Hamprecht, A.J.Cohen, D.J.Tozer, N.C.Handy
J.Chem.Phys. 109, 6264-6271(1998)

B97-2 hybrid, a reparameterization of B97:

P.J.Wilson, T.J.Bradley, D.J.Tozer
J.Chem.Phys. 115, 9233-9242(2001)

B97-3 hybrid, a reparameterization of B97:

T.W.Keal, D.J.Tozer
J.Chem.Phys. 123, 121103-1/4(2005)

B97-K and BMK hybrids, K=kinetics:

A.D.Boese, J.M.L.Martin
J.Chem.Phys. 121, 3405-3416(2004)

HCTH93, HCTH120, HCTH147, and HCTH407 use training sets with the indicated number of atoms and molecules used to adjust the B97 functional:

HCTH93 is defined in the B97-1 paper.

HCTH120 and HCTH147:

A.D.Boese, N.L.Doltsinis, N.C.Handy, M.Sprick
J.Chem.Phys. 112, 1670-1678(2000)

HCTH407:

A.D.Boese, N.C.Handy
J.Chem.Phys. 114, 5497-5503(2001)

B98, Becke's reparameterization of B97:

A.D. Becke J.Chem.Phys. 108, 9624-9631(1998)

...bringing to an end "the B97 family".

X3LYP hybrid:

X.Xu, Q.Zhang, R.P.Muller, W.A.Goddard
J.Chem.Phys. 122, 014105/1-14(2005)

PBE0 hybrid:

C.Adamo, V.Barone J.Chem.Phys. 110, 6158-6170(1999)

in the grid free program only,

HALF exchange:

This is programmed as 50% HF plus 50% B88 exchange.

BHHLYP exchange/correlation:

This is 50% HF plus 50% B88, with LYP correlation.

Note: neither is the HALF-AND-HALF exchange/correlation:

A.D.Becke J.Chem.Phys. 98, 1372-1377(1993)

which he defined as 50% HF + 50% SVWN.

--- meta-GGA functionals

These are pure DFT meta-GGAs, unless the description explicitly says it is a hybrid!

PKZB (a prototype of the TPSS family):

J.P.Perdew, S.Kurth, A.Zupan, P.Blaha
Phys.Rev.Lett. 82, 2544-2547(1999)

tHCTH and tHCTHhyb=15% HF exchange:

A.D.Boese, N.C.Handy
J.Chem.Phys. 116, 9559-9569(2002)

TPSS:

J.P.Perdew, J.Tao, V.N.Staroverov, G.E.Scuseria
Phys.Rev.Lett. 91, 146401/1-4(2003)
J.P.Perdew, J.Tao, V.N.Staroverov, G.E.Scuseria
J.Chem.Phys. 120, 6898-6911(2004)

TPSSm, a modified TPSS improving atomization energies:

J.P.Perdew, A.Ruzsinszky, J.Tao, G.I.Csonka, G.E.Scuseria
Phys.Rev.A 76, 042506/1-6(2007)

TPSSh, a 10% hybrid using TPSS:

V.N.Staroverov, G.E.Scuseria, J.Tao, J.P.Perdew
J.Chem.Phys. 119, 12129-12137(2003),
erratum is J.Chem.Phys. 121, 11507(2004)

revTPSS, "workhorse functional for CMP and QC"

J.P.Perdew, A.Ruzsinsky, G.I.Csonka, L.A.Constantin, J.Sun
Phys.Rev.Lett. 103, 026403/1-4(2009)

VS98 (whose form is the prototype of the M06 family):

T.V.Voorhis, G.E.Scuseria J.Chem.Phys. 109, 400-410(1998)

U.Minnesota xc family:

M05: Y.Zhao, N.E.Schultz, D.G.Truhlar
J.Chem.Phys. 123, 161103/1-4(2005)
M05-2X: Y.Zhao, D.G.Truhlar

J.Comput.Chem.Theory Comput. 2, 1009-1018(2006)
M06: Y.Zhao, D.G.Truhlar
Theoret.Chem.Acc. 120, 215-241(2008)
M06-2X: ibid
M06-HF: Y.Zhao, D.G.Truhlar
J.Phys.Chem.A 110, 13126-13130(2006)
M06-L: Y.Zhao, D.G.Truhlar
J.Chem.Phys. 125, 194101/1-18(2006)
SOGGA: Y.Zhao, D.G.Truhlar
J.Chem.Phys. 128, 184109/1-8(2008)
M08-HX and M08-S0: Y.Zhao, D.G.Truhlar
J.Chem.Theory Comput. 4, 1849-1868(2008)
SOGGA11: R.Peverati, Y.Zhao, D.G.Truhlar
J.Phys.Chem.Lett. 2, 1991-1997(2011)
SOGGA11-X: R.Peverati, D.G.Truhlar
J.Chem.Phys. 135, 191102(2011)
M11: R.Peverati, D.G.Truhlar
J.Phys.Chem.Lett. 2, 2810-2817(2011)
M11-L: R.Peverati, D.G.Truhlar
J.Phys.Chem.Lett. 3, 117-124(2012)
For reviews, please see the paper for M06, and also
Y.Zhao, D.G.Truhlar Acc.Chem.Res. 41, 157-167(2008)
These contain recommendations for choosing the one most
appropriate to your problem.

---- long-range corrected functionals:

LC-BLYP, LC-BOP, LC-BVWN:

Y.Tawada, T.Tsuneda, S.Yanagisawa, Y.Yanai, K.Hirao
J.Chem.Phys. 120, 8425-8433(2004)

CAM-B3LYP:

T.Yanai, D.P.Tew, N.C.Handy
Chem.Phys.Lett. 393, 51-57(2004)

wB97, wB97X, wB97X-D:

J.-D. Chai, M.Head-Gordon
J.Chem.Phys. 128, 084106/1-15(2004)
J.-D. Chai, M.Head-Gordon
Phys.Chem.Chem.Phys. 10, 6615-6620(2008)

A review on the topic of long range corrections, which are
also called 'range separated hybrids', is

D.Jacquemin, E.A.Perpete, G.E.Scuseria, I.Ciofini,
C.Adamo J.Chem.Theory Comput. 4, 123-135(2008)

---- "double-hybrid" ----

The B2PLYP family is a mixture of B88 and HF exchange, and a mixture of LYP and MP2 correlation:

B2-PLYP: S.Grimme J.Chem.Phys. 124, 034108/1-15(2006)

B2G-PLYP: A.Karton, A.Tarnopolsky, J.F.Lamere,
G.C.Schatz, J.M.L.Martin
J.Phys.Chem. A 112, 12868(2008)

B2K-PLYP, B2T-PLYP: A.Tarnopolsky, A.Karton,
R.Sertchook, D.Vuzman, J.M.L.Martin
J.Phys.Chem. A 112, 3(2008)

Double hybrids which are also "long range corrected" (and whose parameters depend on the basis set):

wB97X-2, wB97X-2L: J.-D. Chai, M.Head-Gordon
J.Chem.Phys. 131, 174105/1-13(2009)

* * * * *

Some of the functionals now present in GAMESS were made using code from the "density functional repository",

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

We thank Huub van Dam for his assistance with this, and particularly for providing the VWN1RPA functional. The Minnesota functionals are based on subroutines provided by the Truhlar group at the University of Minnesota. Some functionals, and particularly their high derivatives needed by TDDFT, were created by MAXIMA's algebraic manipulation, along the lines described by

P.Salek, A.Hesselmann
J.Comput.Chem. 28, 2569-2575(2007)

* * * * *

The paper of Johnson, Gill, and Pople listed below has a useful summary of formulae, and details about a gradient implementation. A paper on 1st and 2nd derivatives of DFT with respect to nuclear coordinates and applied fields is

A.Komornicki, G.Fitzgerald
J.Chem.Phys. 98, 1398-1421(1993)

and see also

P.Deglmann, F.Furche, R.Ahlrichs
Chem.Phys.Lett. 362, 511-518(2002).

A few of the many papers assessing the accuracy of DFT:

B.Miehlich, A.Savin, H.Stoll, H.Preuss
Chem.Phys.Lett. 157, 200-206(1989)

B.G.Johnson, P.M.W.Gill, J.A.Pople
J.Chem.Phys. 98, 5612-5626(1993)

N.Oliphant, R.J.Bartlett

J.Chem.Phys. 100, 6550-6561(1994)

L.A.Curtiss, K.Raghavachari, P.C.Redfern, J.A.Pople

J.Chem.Phys. 106, 1063-1079(1997)

E.R.Davidson Int.J.Quantum Chem. 69, 241-245(1998)

B.J.Lynch, D.G.Truhlar

J.Phys.Chem.A 105, 2936-2941(2001)

R.A.Pascal J.Phys.Chem.A 105, 9040-9048(2001)

A.D.Boese, J.M.L.Martin, N.C.Handy

J.Chem.Phys. 119, 3005-3014(2003)

Y.Zhao, D.G.Truhlar,

J.Phys.Chem.A 109, 5656-5667(2005)

K.E.Riley, B.T.Op't Holt, K.M.Merz

J.Chem.Theory Comput. 3, 407-433(2007)

S.F.Sousa, P.A.Fernandes, M.J.Ramos

J.Phys.Chem.A 111, 10439-10452(2007)

Boese et al. include basis set comparisons, as well as functional comparisons. The final paper is a review of reviews, and encourages you to think past B3LYP, which after all dates from 1993! Of course there are assessments in many of the functional papers as well.

On the accuracy of DFT for large molecule thermochemistry:

L.A.Curtiss, K.Ragavachari, P.C.Redfern, J.A.Pople

J.Chem.Phys. 112, 7374-7383(2000)

P.C.Redfern, P.Zapol, L.A.Curtiss, K.Ragavachari

J.Phys.Chem.A 104, 5850-5854(2000)

On the accuracy of TD-DFT excitation energies:

S.S.Leang, F.Zahariev, M.S.Gordon

J.Chem.Phys. 136, 104101/1-12(2012)

Spin contamination in DFT:

1. It is empirically observed that the $\langle S^2 \rangle$ values for unrestricted DFT are smaller than for unrestricted HF.
2. GAMESS computes the $\langle S^2 \rangle$ quantity in an approximate way, namely it pretend that the Kohn-Shan orbitals can be used to form a determinant (WRONG, WRONG, WRONG, there is no wavefunction in DFT!!!) and then uses the same formula that UHF uses to evaluate that determinant's spin expectation value. See

G.J.Laming, N.C.Handy, R.D.Amos

Mol.Phys. 80, 1121-1134(1993)

J.Baker, A.Scheiner, J.Andzelm

Chem.Phys.Lett. 216, 380-388(1993)

C.Adamo, V.Barone, A.Fortunelli

J.Chem.Phys. 98, 8648-8652(1994)

J.A.Pople, P.M.W.Gill, N.C.Handy
Int.J.Quantum Chem. 56, 303-305(1995)
J.Wang, A.D.Becke, V.H.Smith
J.Chem.Phys. 102, 3477-3480(1995)
J.M.Wittbrodt, H.B.Schlegel
J.Chem.Phys. 105, 6574-6577(1996)
J.Grafenstein, D.Cremer
Mol.Phys. 99, 981-989(2001)
and commentary in Koch & Holthausen, pp 52-54.

Orbital energies:

The discussion on page 49-50 of Koch and Holthausen shows that although the highest occupied orbital's eigenvalue should be the ionization potential for exact Kohn-Sham calculations, the functionals we actually have greatly underestimate IP values. The 5th reference below shows how inclusion of HF exchange helps this, and provides a linear correction formula for IPs. The first two papers below connect the HOMO eigenvalue to the IP, and the third shows that while the band gap is underestimated by existing functionals, the gap's center is correctly predicted. However, the 5th paper shows that DFT is actually pretty hopeless at predicting these gaps. The 4th paper uses SCF densities to generate exchange-correlation potentials that actually give fairly good IP values:

J.F.Janak Phys.Rev.B 18, 7165-7168(1978)
M.Levy, J.P.Perdew, V.Sahni
Phys.Rev.A 30, 2745-2748(1984)
J.P.Perdew, M.Levy Phys.Rev.Lett. 51, 1884-1887(1983)
A.Nagy, M.Levy Chem.Phys.Lett. 296, 313-315(1998)
G.Zhang, C.B.Musgrave J.Phys.Chem.A 111, 1554-1561(2007)

Summary of excited state methods

This is not a "how to" section, as the actual calculations will be carried out by means described elsewhere in this chapter. Instead, a summary of methods that can treat excited states is given.

The simplest possibility is SCFTYP. For example, a closed shell molecule's first triplet state can always be treated by SCFTYP=ROHF MULT=3. Assuming there is some symmetry present, the GVB program may be able to do excited singlets variationally, provided they are of a different space symmetry than the ground state. The MCSCF program gives a general entree into excited states, since upper roots of a Hamiltonian are always variational: see for example NSTATE and WSTATE and IROOT in \$DET. Of course, 2nd order perturbation theory can include correlation energy into these SCF level calculations. Note in particular the usefulness of quasi-degenerate multireference perturbation theory when electronic states have similar energies.

CI calculations also give a simple entree into excited states. There are a variety of programs, selected by CITYP in \$CONTRL. Note in particular CITYP=CIS, programmed for closed shell ground states, with gradient capability for singlet excited states, and for the calculation of triplet state energies. The other CI programs can generate very flexible wavefunctions for the evaluation of the excitation energy, and property values. Note that the GUGA program will do nuclear gradients provided the reference is RHF.

The TD-DFT method treats singly excited states, including correlation effects, and is a popular alternative to CIS. The program allows for excitation energies from a UHF reference, but is much more powerful for RHF references: nuclear gradients and/or properties may be computed. Use of a "long range corrected" or "range separated" functional (the two terms are synonymous) is often thought to be important when treating charge transfer or Rydberg states: see the LC=.TRUE. flag or CAMB3LYP. Spin-flip TDDFT allows the users to select as the reference state something more appropriate to the orbital optimization stage. See \$TDDFT for details.

Equation of Motion (EOM) coupled cluster can give accurate estimates of excitation energies. There are no gradients, and properties exist only for the EOM-CCSD level, but

triples corrections to the energy are available. See \$EOMINP for more details.

Most of the runs will predict oscillator strengths, or Einstein coefficients, or similar data regarding the electronic transition moments. Full prediction of UV-vis spectra is not possible without Franck-Condon information.

Excited states frequently come close together, and crossings between them are of great interest.

See RUNTYP=TRANSITION for spin-orbit coupling, which is responsible for InterSystem Crossing (ISC) between states of different spin multiplicity. See RUNTYP=NACME for the computation of the non-adiabatic coupling matrix elements that cause Internal Conversion (IC) between states of the same spin multiplicity. Alternatively, diabatic potential surfaces may be generated at the MCSCF or MCQDPT levels: see DIABAT in the \$MCSCF group.

It is possible to search for the lowest energy on the crossing seam between two surfaces. In case those surfaces have different spins, or different space symmetries (or both), see RUNTYP=MEX. When the surfaces have the same symmetry, see RUNTYP=CONINT for location of conical intersections.

Solvent effects (EFP and/or PCM) can easily be incorporated when using SCFTYP to generate the states, and nuclear gradients are available. It is now possible to assess solvent effects on TD-DFT excitation energies from closed shell references, using either EFP or PCM.

Excited states often possess Rydberg character, so diffuse functions in the basis set are likely to be important.

Geometry Searches and Internal Coordinates

Stationary points are places on the potential energy surface with a zero gradient vector (first derivative of the energy with respect to nuclear coordinates). These include minima (whether relative or global), better known to chemists as reactants, products, and intermediates; as well as transition states (also known as saddle points).

The two types of stationary points have a precise mathematical definition, depending on the curvature of the potential energy surface at these points. If all of the eigenvalues of the hessian matrix (second derivative of the energy with respect to nuclear coordinates) are positive, the stationary point is a minimum. If there is one, and only one, negative curvature, the stationary point is a transition state. Points with more than one negative curvature do exist, but are not important in chemistry. Because vibrational frequencies are basically the square roots of the curvatures, a minimum has all real frequencies, and a saddle point has one imaginary vibrational "frequency".

GAMESS locates minima by geometry optimization, as RUNTYP=OPTIMIZE, and transition states by saddle point searches, as RUNTYP=SADPOINT. In many ways these are similar, and in fact nearly identical FORTRAN code is used for both. The term "geometry search" is used here to describe features which are common to both procedures. The input to control both RUNTYPs is found in the \$STATPT group.

As will be noted in the symmetry section below, an OPTIMIZE run does not always find a minimum, and a SADPOINT run may not find a transition state, even though the gradient is brought to zero. You can prove you have located a minimum or saddle point only by examining the local curvatures of the potential energy surface. This can be done by following the geometry search with a RUNTYP=HESSIAN job, which should be a matter of routine.

quasi-Newton Searches

Geometry searches are most effectively done by what is called a quasi-Newton-Raphson procedure. These methods assume a quadratic potential surface, and require the

exact gradient vector and an approximation to the hessian. It is the approximate nature of the hessian that makes the method "quasi". The rate of convergence of the geometry search depends on how quadratic the real surface is, and the quality of the hessian. The latter is something you have control over, and is discussed in the next section.

GAMESS contains different implementations of quasi-Newton procedures for finding stationary points, namely METHOD=NR, RFO, QA, and the seldom used SCHLEGEL. They differ primarily in how the step size and direction are controlled, and how the Hessian is updated. The CONOPT method is a way of forcing a geometry away from a minimum towards a TS. It is not a quasi-Newton method, and is described at the very end of this section.

The NR method employs a straight Newton-Raphson step. There is no step size control, the algorithm will simply try to locate the nearest stationary point, which may be a minimum, a TS, or any higher order saddle point. NR is not intended for general use, but is used by GAMESS in connection with some of the other methods after they have homed in on a stationary point, and by Gradient Extremal runs where location of higher order saddle points is common. NR requires a very good estimate of the geometry in order to converge on the desired stationary point.

The RFO and QA methods are two different versions of the so-called augmented Hessian techniques. They both employ Hessian shift parameter(s) in order to control the step length and direction.

In the RFO method, the shift parameter is determined by approximating the PES with a Rational Function, instead of a second order Taylor expansion. For a RUNTYP=SADPOINT, the TS direction is treated separately, giving two shift parameters. This is known as a Partitioned Rational Function Optimization (P-RFO). The shift parameter(s) ensure that the augmented Hessian has the correct eigenvalue structure, all positive for a minimum search, and one negative eigenvalue for a TS search. The (P)-RFO step can have any length, but if it exceeds DXMAX, the step is simply scaled down.

In the QA (Quadratic Approximation) method, the shift parameter is determined by the requirement that the step size should equal DXMAX. There is only one shift parameter for both minima and TS searches. Again the augmented Hessian will have the correct structure. There

is another way of describing the same algorithm, namely as a minimization on the "image" potential. The latter is known as TRIM (Trust Radius Image Minimization). The working equation is identical in these two methods.

When the RFO steplength is close to DXMAX, there is little difference between the RFO and QA methods. However, the RFO step may in some cases exceed DXMAX significantly, and a simple scaling of the step will usually not produce the best direction. The QA step is the best step on the hypersphere with radius DXMAX. For this reason QA is the default algorithm.

Near a stationary point the straight NR algorithm is the most efficient. The RFO and QA may be viewed as methods for guiding the search in the "correct" direction when starting far from the stationary point. Once the stationary point is approached, the RFO and QA methods switch to NR, automatically, when the NR steplength drops below 0.10 or DXMAX, whichever is the smallest.

The QA method works so well that we use it exclusively, and so the SCHLEGEL method will probably be omitted from some future version of GAMESS.

You should read the papers mentioned below in order to understand how these methods are designed to work. The first 3 papers describe the RFO and TRIM/QA algorithms. A good but slightly dated summary of search procedures is given by Bell and Crighton, and see also the review by Schlegel. Most of the FORTRAN code for geometry searches, and some of the discussion in this section was written by Frank Jensen of the University of Aarhus, whose paper compares many of the algorithms implemented in GAMESS:

1. J.Baker J.Comput.Chem. 7, 385-395(1986)
2. T.Helgaker Chem.Phys.Lett. 182, 503-510(1991)
3. P.Culot, G.Dive, V.H.Nguyen, J.M.Ghuysen
Theoret.Chim.Acta 82, 189-205(1992)
4. H.B.Schlegel J.Comput.Chem. 3, 214-218(1982)
5. S.Bell, J.S.Crighton
J.Chem.Phys. 80, 2464-2475(1984).
6. H.B.Schlegel Advances in Chemical Physics (Ab Initio
Methods in Quantum Chemistry, Part I), volume 67,
K.P.Lawley, Ed. Wiley, New York, 1987, pp 249-286.
7. F.Jensen J.Chem.Phys. 102, 6706-6718(1995).

the nuclear Hessian

Although quasi-Newton methods require only an approximation to the true hessian, the quality of this matrix has a great affect on convergence of the geometry search.

There is a procedure contained within GAMESS for guessing a positive definite hessian matrix, HESS=GUESS. If you are using Cartesian coordinates, the guess hessian is based on pairwise atom stretches. The guess is more sophisticated when internal coordinates are defined, as empirical rules will be used to estimate stretching and bending force constants. Other angular force constants are set to 1/4. The guess often works well for minima, but cannot possibly find transition states (because it is positive definite). Therefore, GUESS may not be selected for SADPOINT runs.

Two options for providing a more accurate hessian are HESS=READ and CALC. For the latter, the true hessian is obtained by direct calculation at the initial geometry, and then the geometry search begins, all in one run. The READ option allows you to feed in the hessian in a \$HESS group, as obtained by a RUNTYP=HESSIAN job. The second procedure is actually preferable, as you get a chance to see the frequencies. Then, if the local curvatures look good, you can commit to the geometry search. Be sure to include a \$GRAD group (if the exact gradient is available) in the HESS=READ job so that GAMESS can take its first step immediately.

Note also that you can compute the hessian at a lower basis set and/or wavefunction level, and read it into a higher level geometry search. In fact, the \$HESS group could be obtained at the semiempirical level. This trick works because the hessian is $3N \times 3N$ for N atoms, no matter what atomic basis is used. The gradient from the lower level is of course worthless, as the geometry search must work with the exact gradient of the wavefunction and basis set in current use. Discard the \$GRAD group from the lower level calculation!

You often get what you pay for. HESS=GUESS is free, but may lead to significantly more steps in the geometry search. The other two options are more expensive at the beginning, but may pay back by rapid convergence to the stationary point.

The hessian update frequently improves the hessian for a few steps (especially for HESS=GUESS), but then breaks down. The symptoms are a nice lowering of the energy or the RMS gradient for maybe 10 steps, followed by crazy steps. You can help by putting the best coordinates into \$DATA, and resubmitting, to make a fresh determination of the hessian.

The default hessian update for OPTIMIZE runs is BFGS, which is likely to remain positive definite. The POWELL update is the default for SADPOINT runs, since the hessian can develop a negative curvature as the search progresses. The POWELL update is also used by the METHOD=NR and CONOPT since the Hessian may have any number of negative eigenvalues in these cases. The MSP update is a mixture of Murtagh-Sargent and Powell, suggested by Josep Bofill, (J.Comput.Chem., 15, 1-11, 1994). It sometimes works slightly better than Powell, so you may want to try it.

coordinate choices

Optimization in cartesian coordinates has a reputation of converging slowly. This is largely due to the fact that translations and rotations are usually left in the problem. Numerical problems caused by the small eigenvalues associated with these degrees of freedom are the source of this poor convergence. The methods in GAMESS project the hessian matrix to eliminate these degrees of freedom, which should not cause a problem. Nonetheless, Cartesian coordinates are in general the most slowly convergent coordinate system.

The use of internal coordinates (see NZVAR in \$CONTRL as well as \$ZMAT) also eliminates the six rotational and translational degrees of freedom. Also, when internal coordinates are used, the GUESS hessian is able to use empirical information about bond stretches and bends. On the other hand, there are many possible choices for the internal coordinates, and some of these may lead to much poorer convergence of the geometry search than others. Particularly poorly chosen coordinates may not even correspond to a quadratic surface, thereby ending all hope that a quasi-Newton method will converge.

Internal coordinates are frequently strongly coupled. Because of this, Jerry Boatz has called them "infernial coordinates"! A very common example to illustrate this might be a bond length in a ring, and the angle on the

opposite side of the ring. Clearly, changing one changes the other simultaneously. A more mathematical definition of "coupled" is to say that there is a large off-diagonal element in the hessian. In this case convergence may be unsatisfactory, especially with a diagonal GUESS hessian, where a "good" set of internals is one with a diagonally dominant hessian. Of course, if you provide an accurately computed hessian, it will have large off-diagonal values where those are truly present. Even so, convergence may be poor if the coordinates are coupled through large 3rd or higher derivatives. The best coordinates are therefore those which are the most "quadratic".

One very popular set of internal coordinates is the usual "model builder" Z-matrix input, where for N atoms, one uses N-1 bond lengths, N-2 bond angles, and N-3 bond torsions. The popularity of this choice is based on its ease of use in specifying the initial molecular geometry. Typically, however, it is the worst possible choice of internal coordinates, and in the case of rings, is not even as good as Cartesian coordinates.

However, GAMESS does not require this particular mix of the common types. GAMESS' only requirement is that you use a total of $3N-6$ coordinates, chosen from these 3 basic types, or several more exotic possibilities. (Of course, we mean $3N-5$ throughout for linear molecules). These additional types of internal coordinates include linear bends for 3 collinear atoms, out of plane bends, and so on. There is no reason at all why you should place yourself in a straightjacket of N-1 bonds, N-2 angles, and N-3 torsions. If the molecule has symmetry, be sure to use internals which are symmetrically related.

For example, the most effective choice of coordinates for the atoms in a four membered ring is to define all four sides, any one of the internal angles, and a dihedral defining the ring pucker. For a six membered ring, the best coordinates seem to be 6 sides, 3 angles, and 3 torsions. The angles should be every other internal angle, so that the molecule can "breathe" freely. The torsions should be arranged so that the central bond of each is placed on alternating bonds of the ring, as if they were pi bonds in Kekule benzene. For a five membered ring, we suggest all 5 sides, 2 internal angles, again alternating every other one, and 2 dihedrals to fill in. The internal angles of necessity skip two atoms where the ring closes. Larger rings should generalize on the idea

of using all sides but only alternating angles. If there are fused rings, start with angles on the fused bond, and alternate angles as you go around from this position.

Rings and more especially fused rings can be tricky. For these systems, especially, we suggest the Cadillac of internal coordinates, the "natural internal coordinates" of Peter Pulay. For a description of these, see

P.Pulay, G.Fogarosi, F.Pang, J.E.Boggs,
J.Am.Chem.Soc. 101, 2550-2560 (1979).
G.Fogarasi, X.Zhou, P.W.Taylor, P.Pulay
J.Am.Chem.Soc. 114, 8191-8201 (1992).

These are linear combinations of local coordinates, except in the case of rings. The examples given in these two papers are very thorough.

An illustration of these types of coordinates is given in the example job EXAM25.INP, distributed with GAMESS. This is a nonsense molecule, designed to show many kinds of functional groups. It is defined using standard bond distances with a classical Z-matrix input, and the angles in the ring are adjusted so that the starting value of the unclosed OO bond is also a standard value.

Using Cartesian coordinates is easiest, but takes a very large number of steps to converge. This however, is better than using the classical Z-matrix internals given in \$DATA, which is accomplished by setting NZVAR to the correct 3N-6 value. The geometry search changes the OO bond length to a very short value on the 1st step, and the SCF fails to converge. (Note that if you have used dummy atoms in the \$DATA input, you cannot simply enter NZVAR to optimize in internal coordinates, instead you must give a \$ZMAT which involves only real atoms).

The third choice of internal coordinates is the best set which can be made from the simple coordinates. It follows the advice given above for five membered rings, and because it includes the OO bond, has no trouble with crashing this bond. It takes 20 steps to converge, so the trouble of generating this \$ZMAT is certainly worth it compared to the use of Cartesians.

Natural internal coordinates are defined in the final group of input. The coordinates are set up first for the ring, including two linear combinations of all angles and all torsions withing the ring. After this the methyl is

hooked to the ring as if it were a NH group, using the usual terminal methyl hydrogen definitions. The H is hooked to this same ring carbon as if it were a methine. The NH and the CH₂ within the ring follow Pulay's rules exactly. The amount of input is much greater than a normal Z-matrix. For example, 46 internal coordinates are given, which are then placed in $3N-6=33$ linear combinations. Note that natural internals tend to be rich in bends, and short on torsions.

The energy results for the three coordinate systems which converge are as follows:

NSERCH	Cart	good Z-mat	nat. int.
0	-48.6594935049	-48.6594935049	-48.6594935049
1	-48.6800538676	-48.6806631261	-48.6838361406
2	-48.6822702585	-48.6510215698	-48.6874045449
3	-48.6858299354	-48.6882945647	-48.6932811528
4	-48.6881499412	-48.6849667085	-48.6946836332
5	-48.6890226067	-48.6911899936	-48.6959800274
6	-48.6898261650	-48.6878047907	-48.6973821465
7	-48.6901936624	-48.6930608185	-48.6987652146
8	-48.6905304889	-48.6940607117	-48.6996366016
9	-48.6908626791	-48.6949137185	-48.7006656309
10	-48.6914279465	-48.6963767038	-48.7017273728
11	-48.6921521142	-48.6986608776	-48.7021504975
12	-48.6931136707	-48.7007305310	-48.7022405019
13	-48.6940437619	-48.7016095285	-48.7022548935
14	-48.6949546487	-48.7021531692	-48.7022569328
15	-48.6961698826	-48.7022080183	-48.7022570260
16	-48.6973813002	-48.7022454522	-48.7022570662
17	-48.6984850655	-48.7022492840	
18	-48.6991553826	-48.7022503853	
19	-48.6996239136	-48.7022507037	
20	-48.7002269303	-48.7022508393	
21	-48.7005379631		
22	-48.7008387759		
	...		
50	-48.7022519950		

from which you can see that the natural internals are actually the best set. The \$ZMAT exhibits upward burps in the energy at step 2, 4, and 6, so that for the same number of steps, these coordinates are always at a higher energy than the natural internals.

The initial hessian generated for these three columns contains 0, 33, and 46 force constants. This assists the natural internals, but is not the major reason for

its superior performance. The computed hessian at the final geometry of this molecule, when transformed into the natural internal coordinates is almost diagonal. This almost complete uncoupling of coordinates is what makes the natural internals perform so well. The conclusion is of course that not all coordinate systems are equal, and natural internals are the best. As another example, we have run the ATCHCP molecule, which is a popular geometry optimization test, due to its two fused rings:

H.B.Schlegel, Int.J.Quantum Chem., Symp. 26, 253-264(1992)
 T.H.Fischer and J.Almlof, J.Phys.Chem. 96, 9768-9774(1992)
 J.Baker, J.Comput.Chem. 14, 1085-1100(1993)

Here we have compared the same coordinate types, using a guess hessian, or a computed hessian. The latter set of runs is a test of the coordinates only, as the initial hessian information is identical. The results show clearly the superiority of the natural internals, which like the previous example, give an energy decrease on every step:

	HESS=GUESS	HESS=READ
Cartesians	65	41 steps
good Z-matrix	32	23
natural internals	24	13

A final example is phosphinoazasilatrane, with three rings fused on a common SiN bond, in which 112 steps in Cartesian space became 32 steps in natural internals. The moral is:

"A little brain time can save a lot of CPU time."

In late 1998, a new kind of internal coordinate method was included into GAMESS. This is the delocalized internal

coordinate (DLC) of

J.Baker, A. Kessi, B.Delley
 J.Chem.Phys. 105, 192-212(1996)

although as is the usual case, the implementation is not exactly the same. Bonds are kept as independent coordinates, while angles are placed in linear combination by the DLC process. There are some interesting options for applying constraints, and other options to assist the automatic DLC generation code by either adding or deleting coordinates. It is simple to use DLCs in their most basic form:

```
$contrl nzvar=xx $end
```

```
$zmat dlc=.true. auto=.true. $end
```

Our initial experience is that the quality of DLCs is

not as good as explicitly constructed natural internals, which benefit from human chemical knowledge, but are almost always better than carefully crafted \$ZMATs using only the primitive internal coordinates (although we have seen a few exceptions). Once we have more numerical experience with the use of DLC's, we will come back and revise the above discussion of coordinate choices. In the meantime, they are quite simple to choose, so give them a go.

the role of symmetry

At the end of a succesful geometry search, you will have a set of coordinates where the gradient of the energy is zero. However your newly discovered stationary point is not necessarily a minimum or saddle point!

This apparent mystery is due to the fact that the gradient vector transforms under the totally symmetric representation of the molecular point group. As a direct consequence, a geometry search is point group conserving. (For a proof of these statements, see J.W.McIver and A.Komornicki, Chem.Phys.Lett., 10,303-306(1971)). In simpler terms, the molecule will remain in whatever point group you select in \$DATA, even if the true minimum is in some lower point group. Since a geometry search only explores totally symmetric degrees of freedom, the only way to learn about the curvatures for all degrees of freedom is RUNTYP=HESSIAN.

As an example, consider disilene, the silicon analog of ethene. It is natural to assume that this molecule is planar like ethene, and an OPTIMIZE run in D2h symmetry will readily locate a stationary point. However, as a calculation of the hessian will readily show, this structure is a transition state (one imaginary frequency), and the molecule is really trans-bent (C2h). A careful worker will always characterize a stationary point as either a minimum, a transition state, or some higher order stationary point (which is not of great interest!) by performing a RUNTYP=HESSIAN.

The point group conserving properties of a geometry search can be annoying, as in the preceeding example, or advantageous. For example, assume you wish to locate the transition state for rotation about the double bond in ethene. A little thought will soon reveal that ethene is D2h, the 90 degrees twisted structure is D2d, and structures in between are D2. Since the saddle point is

actually higher symmetry than the rest of the rotational surface, you can locate it by RUNTYP=OPTIMIZE within D2d symmetry. You can readily find this stationary point with the diagonal guess hessian! In fact, if you attempt to do a RUNTYP=SADPOINT within D2d symmetry, there will be no totally symmetric modes with negative curvatures, and it is unlikely that the geometry search will be very well behaved.

Although a geometry search cannot lower the symmetry, the gain of symmetry is quite possible. For example, if you initiate a water molecule optimization with a trial structure which has unequal bond lengths, the geometry search will come to a structure that is indeed C_{2v} (to within OPTTOL, anyway). However, GAMESS leaves it up to you to realize that a gain of symmetry has occurred.

In general, Mother Nature usually chooses more symmetrical structures over less symmetrical structures. Therefore you are probably better served to assume the higher symmetry, perform the geometry search, and then check the stationary point's curvatures. The alternative is to start with artificially lower symmetry and see if your system regains higher symmetry. The problem with this approach is that you don't necessarily know which subgroup is appropriate, and you lose the great speedups GAMESS can obtain from proper use of symmetry. It is good to note here that "lower symmetry" does not mean simply changing the name of the point group and entering more atoms in \$DATA, instead the nuclear coordinates themselves must actually be of lower symmetry.

practical matters

Geometry searches do not bring the gradient exactly to zero. Instead they stop when the largest component of the gradient is below the value of OPTTOL, which defaults to a reasonable 0.0001. Analytic Hessians usually have residual frequencies below 10 cm⁻¹ with this degree of optimization. The sloppiest value you probably ever want to try is 0.0005.

If a geometry search runs out of time, or exceeds NSTEP, it can be restarted. For RUNTYP=OPTIMIZE, restart with the coordinates having the lowest total energy (do a string search on "FINAL"). For RUNTYP=SADPOINT, restart with the coordinates having the smallest gradient

(do a string search on "RMS", which means root mean square).

These are not necessarily at the last geometry!

The "restart" should actually be a normal run, that is you should not try to use the restart options in \$CONTRL (which may not work anyway). A geometry search can be restarted by extracting the desired coordinates for \$DATA from the printout, and by extracting the corresponding \$GRAD group from the PUNCH file. If the \$GRAD group is supplied, the program is able to save the time it would ordinarily take to compute the wavefunction and gradient at the initial point, which can be a substantial savings. There is no input to trigger reading of a \$GRAD group: if found, it is read and used. Be careful that your \$GRAD group actually corresponds to the coordinates in \$DATA, as GAMESS has no check for this.

Sometimes when you are fairly close to the minimum, an OPTIMIZE run will take a first step which raises the energy, with subsequent steps improving the energy and perhaps finding the minimum. The erratic first step is caused by the GUESS hessian. It may help to limit the size of this wrong first step, by reducing its radius, DXMAX. Conversely, if you are far from the minimum, sometimes you can decrease the number of steps by increasing DXMAX.

When using internals, the program uses an iterative process to convert the internal coordinate change into Cartesian space. In some cases, a small change in the internals will produce a large change in Cartesians, and thus produce a warning message on the output. If these warnings appear only in the beginning, there is probably no problem, but if they persist you can probably devise a better set of coordinates. You may in fact have one of the two problems described in the next paragraph. In some cases (hopefully very few) the iterations to find the Cartesian displacement may not converge, producing a second kind of warning message. The fix for this may very well be a new set of internal coordinates as well, or adjustment of ITBMAT in \$STATPT.

There are two examples of poorly behaved internal coordinates which can give serious problems. The first of these is three angles around a central atom, when this atom becomes planar (sum of the angles nears 360). The other is a dihedral where three of the atoms are nearly linear, causing the dihedral to flip between 0 and 180. Avoid these two situations if you want your geometry

search to be convergent.

Sometimes it is handy to constrain the geometry search by freezing one or more coordinates, via the IFREEZ array. For example, constrained optimizations may be useful while trying to determine what area of a potential energy surface contains a saddle point. If you try to freeze coordinates with an automatically generated \$ZMAT, you need to know that the order of the coordinates defined in \$DATA is

```
y
y  x r1
y  x r2  x a3
y  x r4  x a5  x w6
y  x r7  x a8  x w9
```

and so on, where y and x are whatever atoms and molecular connectivity you happen to be using.

saddle points

Finding minima is relatively easy. There are large tables of bond lengths and angles, so guessing starting geometries is pretty straightforward. Very nasty cases may require computation of an exact hessian, but the location of most minima is straightforward.

In contrast, finding saddle points is a black art. The diagonal guess hessian will never work, so you must provide a computed one. The hessian should be computed at your best guess as to what the transition state (T.S.) should be. It is safer to do this in two steps as outlined above, rather than HESS=CALC. This lets you verify you have guessed a structure with one and only one negative curvature. Guessing a good trial structure is the hardest part of a RUNTYP=SADPOINT!

This point is worth iterating. Even with sophisticated step size control such as is offered by the QA/TRIM or RFO methods, it is in general very difficult to move correctly from a region with incorrect curvatures towards a saddle point. Even procedures such as CONOPT or RUNTYP=GRADEXTR will not replace your own chemical intuition about where saddle points may be located.

The RUNTYP=HESSIAN's normal coordinate analysis is rigorously valid only at stationary points on the surface. This means the frequencies from the hessian at your trial

geometry are untrustworthy, in particular the six "zero" frequencies corresponding to translational and rotational (T&R) degrees of freedom will usually be $300-500\text{ cm}^{-1}$, and possibly imaginary. The Sayvetz conditions on the printout will help you distinguish the T&R "contaminants" from the real vibrational modes. If you have defined a \$ZMAT, the PURIFY option within \$STATPT will help zap out these T&R contaminants).

If the hessian at your assumed geometry does not have one and only one imaginary frequency (taking into account that the "zero" frequencies can sometimes be $300i!$), then it will probably be difficult to find the saddle point. Instead you need to compute a hessian at a better guess for the initial geometry, or read about mode following below.

If you need to restart your run, do so with the coordinates which have the smallest RMS gradient. Note that the energy does not necessarily have to decrease in a SADPOINT run, in contrast to an OPTIMIZE run. It is often necessary to do several restarts, involving recomputation of the hessian, before actually locating the saddle point.

Assuming you do find the T.S., it is always a good idea to recompute the hessian at this structure. As described in the discussion of symmetry, only totally symmetric vibrational modes are probed in a geometry search. Thus it is fairly common to find that at your "T.S." there is a second imaginary frequency, which corresponds to a non-totally symmetric vibration. This means you haven't found the correct T.S., and are back to the drawing board. The proper procedure is to lower the point group symmetry by distorting along the symmetry breaking "extra" imaginary mode, by a reasonable amount. Don't be overly timid in the amount of distortion, or the next run will come back to the invalid structure.

The real trick here is to find a good guess for the transition structure. The closer you are, the better. It is often difficult to guess these structures. One way around this is to compute a linear least motion (LLM) path. This connects the reactant structure to the product structure by linearly varying each coordinate. If you generate about ten structures intermediate to reactants and products, and compute the energy at each point, you will in general find that the energy first goes up, and then down. The maximum energy structure is a "good" guess for the true T.S. structure. Actually, the success of

this method depends on how curved the reaction path is.

A particularly good paper on the symmetry which a saddle point (and reaction path) can possess is by
P.Pechukas, J.Chem.Phys. 64, 1516-1521(1976)

mode following

In certain circumstances, METHOD=RFO and QA can walk from a region of all positive curvatures (i.e. near a minimum) to a transition state. The criteria for whether this will work is that the mode being followed should be only weakly coupled to other close-lying Hessian modes. Especially, the coupling to lower modes should be almost zero. In practise this means that the mode being followed should be the lowest of a given symmetry, or spatially far away from lower modes (for example, rotation of methyl groups at different ends of the molecule). It is certainly possible to follow also modes which do not obey these criteria, but the resulting walk (and possibly TS location) will be extremely sensitive to small details such as the stepsize.

This sensitivity also explain why TS searches often fail, even when starting in a region where the Hessian has the required one negative eigenvalue. If the TS mode is strongly coupled to other modes, the direction of the mode is incorrect, and the maximization of the energy along that direction is not really what you want (but what you get).

Mode following is really not a substitute for the ability to intuit regions of the PES with a single local negative curvature. When you start near a minimum, it matters a great deal which side of the minima you start from, as the direction of the search depends on the sign of the gradient. We strongly urge that you read before trying to use IFOLLOW, namely the papers by Frank Jensen and Jon Baker mentioned above, and see also Figure 3 of C.J.Tsai, K.D.Jordan, J.Phys.Chem. 97, 11227-11237 (1993) which is quite illuminating on the sensitivity of mode following to the initial geometry point.

Note that GAMESS retains all degrees of freedom in its hessian, and thus there is no reason to suppose the lowest mode is totally symmetric. Remember to lower the symmetry in the input deck if you want to follow non-symmetric modes. You can get a printout of the modes in internal

coordinate space by a EXETYP=CHECK run, which will help you decide on the value of IFOLLOW.

* * *

CONOPT is a different sort of saddle point search procedure. Here a certain "CONstrained OPTimization" may be considered as another mode following method. The idea is to start from a minimum, and then perform a series of optimizations on hyperspheres of increasingly larger radii. The initial step is taken along one of the Hessian modes, chosen by IFOLLOW, and the geometry is optimized subject to the constraint that the distance to the minimum is constant. The convergence criteria for the gradient norm perpendicular to the constraint is taken as $10 \times \text{OPTTOL}$, and the corresponding steplength as $100 \times \text{OPTTOL}$.

After such a hypersphere optimization has converged, a step is taken along the line connecting the two previous optimized points to get an estimate of the next hypersphere geometry. The stepsize is DXMAX, and the radius of hyperspheres is thus increased by an amount close (but not equal) to DXMAX. Once the pure NR step size falls below $\text{DXMAX}/2$ or 0.10 (whichever is the largest) the algorithm switches to a straight NR iterate to (hopefully) converge on the stationary point.

The current implementation always conducts the search in cartesian coordinates, but internal coordinates may be printed by the usual specification of NZVAR and ZMAT. At present there is no restart option programmed.

CONOPT is based on the following papers, but the actual implementation is the modified equations presented in Frank Jensen's paper mentioned above.

Y. Abashkin, N. Russo,
J.Chem.Phys. 100, 4477-4483(1994).
Y. Abashkin, N. Russo, M. Toscano,
Int.J.Quant.Chem. 52, 695-704(1994).

There is little experience on how this method works in practice, experiment with it at your own risk!

Intrinsic Reaction Coordinate Methods

The Intrinsic Reaction Coordinate (IRC) is defined as the minimum energy path connecting the reactants to products via the transition state. In practice, the IRC is found by first locating the transition state for the reaction. The IRC is then found in halves, going forward and backwards from the saddle point, down the steepest descent path in mass weighted Cartesian coordinates. This is accomplished by numerical integration of the IRC equations, by a variety of methods to be described below.

The IRC is becoming an important part of polyatomic dynamics research, as it is hoped that only knowledge of the PES in the vicinity of the IRC is needed for prediction of reaction rates, at least at threshold energies. The IRC has a number of uses for electronic structure purposes as well. These include the proof that a certain transition structure does indeed connect a particular set of reactants and products, as the structure and imaginary frequency normal mode at the saddle point do not always unambiguously identify the reactants and products. The study of the electronic and geometric structure along the IRC is also of interest. For example, one can obtain localized orbitals along the path to determine when bonds break or form.

The accuracy to which the IRC is determined is dictated by the use one intends for it. Dynamical calculations require a very accurate determination of the path, as derivative information (second derivatives of the PES at various IRC points, and path curvature) is required later. Thus, a sophisticated integration method (such as GS2), and small step sizes (STRIDE=0.05, 0.01, or even smaller) may be needed. In addition to this, care should be taken to locate the transition state carefully (perhaps decreasing OPTTOL by a factor of 10, to OPTTOL=1D-5), and in the initiation of the IRC run. The latter might require a hessian matrix obtained by double differencing, certainly the hessian should be PROJCT'd or PURIFY'd. Note also that EVIB must be chosen carefully, as described below.

On the other hand, identification of reactants and products allows for much larger step sizes, and cruder integration methods. In this type of IRC one might want to be careful in leaving the saddle point (perhaps STRIDE should be reduced to 0.10 or 0.05 for the first few steps away from the transition state), but once a few points have

been taken, larger step sizes can be employed. In general, the defaults in the \$IRC group are set up for this latter, cruder quality IRC. The STRIDE value for the GS2 method can usually be safely larger than for other methods, no matter what your interest in accuracy is.

The next few paragraphs describe the various integrators, but note that GS2 is superior to the others.

The simplest method of determining an IRC is linear gradient following, PACE=LINEAR. This method is also known as Euler's method. If you are employing PACE=LINEAR, you can select "stabilization" of the reaction path by the Ishida, Morokuma, Komornicki method. This type of corrector has no apparent mathematical basis, but works rather well since the bisector usually intersects the reaction path at right angles (for small step sizes). The ELBOW variable allows for a method intermediate to LINEAR and stabilized LINEAR, in that the stabilization will be skipped if the gradients at the original IRC point, and at the result of a linear prediction step form an angle greater than ELBOW. Set ELBOW=180 to always perform the stabilization.

A closely related method is PACE=QUAD, which fits a quadratic polynomial to the gradient at the current and immediately previous IRC point to predict the next point. This pace has the same computational requirement as LINEAR, and is slightly more accurate due to the reuse of the old gradient. However, stabilization is not possible for this pace, thus a stabilized LINEAR path is usually more accurate than QUAD.

Two rather more sophisticated methods for integrating the IRC equations are the fourth order Adams-Moulton predictor-corrector (PACE=AMPC4) and fourth order Runge-Kutta (PACE=RK4). AMPC4 takes a step towards the next IRC point (prediction), and based on the gradient found at this point (in the near vicinity of the next IRC point) obtains a modified step to the desired IRC point (correction). AMPC4 uses variable step sizes, based on the input STRIDE. RK4 takes several steps part way toward the next IRC point, and uses the gradient at these points to predict the next IRC point. RK4 is one of the most accurate integration method implemented in GAMESS, and is also the most time consuming.

The Gonzalez-Schlegel 2nd order method (PACE=GS2) finds the next IRC point by a constrained optimization on the

surface of a hypersphere, centered at a point $1/2$ STRIDE along the gradient vector leading from the previous IRC point. By construction, the reaction path between two successive IRC points is a circle tangent to the two gradient vectors. The algorithm is much more robust for large steps than the other methods, so it has been chosen as the default method. Thus, the default for STRIDE is too large for the other methods. The number of energy and gradients need to find the next point varies with the difficulty of the constrained optimization, but is normally not very many points. Taking more than 2-3 steps in this constrained optimization is indicative of reaction path curvature, and thus it may help to reduce the step size. Use a small GCUT (same value as OPTTOL) when trying to integrate an IRC very accurately, to be sure the hypersphere optimizations are well converged. Be sure to provide the updated hessian from the previous run when restarting PACE=GS2.

The number of wavefunction evaluations, and energy gradients needed to jump from one point on the IRC to the next point are summarized in the following table:

PACE	# energies	# gradients
----	-----	-----
LINEAR	1	1
stabilized		
LINEAR	3	2
QUAD	1	1 (+ reuse of historical gradient)
AMPC4	2	2 (see note)
RK4	4	4
GS2	2-4	2-4 (equal numbers)

Note that the AMPC4 method sometimes does more than one correction step, with each such correction adding one more energy and gradient to the calculation. You get what you pay for in IRC calculations: the more energies and gradients which are used, the more accurate the path found.

A description of these methods, as well as some others that were found to be not as good is given by Kim Baldridge and Lisa Pederson, *Pi Mu Epsilon J.*, 9, 513-521 (1993).

* * *

All methods are initiated by jumping from the saddle point, parallel to the normal mode (CMODE) which has an imaginary frequency. The jump taken is designed to lower

the energy by an amount EVIB. The actual distance taken is thus a function of the imaginary frequency, as a smaller FREQ will produce a larger initial jump. You can simply provide a \$HESS group instead of CMODE and FREQ, which involves less typing. To find out the actual step taken for a given EVIB, use EXETYP=CHECK. The direction of the jump (towards reactants or products) is governed by FORWRD. Note that if you have decided to use small step sizes, you must employ a smaller EVIB to ensure a small first step. The GS2 method begins by following the normal mode by one half of STRIDE, and then performing a hypersphere minimization about that point, so EVIB is irrelevant to this PACE.

The only method which proves that a properly converged IRC has been obtained is to regenerate the IRC with a smaller step size, and check that the IRC is unchanged. Again, note that the care with which an IRC must be obtained is highly dependent on what use it is intended for.

Some key IRC references are:

- K.Ishida, K.Morokuma, A.Komornicki
J.Chem.Phys. 66, 2153-2156 (1977)
K.Müller
Angew.Chem., Int.Ed.Engl. 19, 1-13 (1980)
M.W.Schmidt, M.S.Gordon, M.Dupuis
J.Am.Chem.Soc. 107, 2585-2589 (1985)
B.C.Garrett, M.J.Redmon, R.Steckler, D.G.Truhlar,
K.K.Baldrige, D.Bartol, M.W.Schmidt, M.S.Gordon
J.Phys.Chem. 92, 1476-1488(1988)
K.K.Baldrige, M.S.Gordon, R.Steckler, D.G.Truhlar
J.Phys.Chem. 93, 5107-5119(1989)
C.Gonzalez, H.B.Schlegel
J.Chem.Phys. 90, 2154-2161(1989)

The IRC discussion closes with some practical tips:

The \$IRC group has a confusing array of variables, but fortunately very little thought need be given to most of them. An IRC run is restarted by moving the coordinates of the next predicted IRC point into \$DATA, and inserting the new \$IRC group into your input file. You must select the desired value for NPOINT. Thus, only the first job which initiates the IRC requires much thought about \$IRC.

The symmetry specified in the \$DATA deck should be the symmetry of the reaction path. If a saddle point happens to have higher symmetry, use only the lower symmetry in the

\$DATA deck when initiating the IRC. The reaction path will have a lower symmetry than the saddle point whenever the normal mode with imaginary frequency is not totally symmetric. Be careful that the order and orientation of the atoms corresponds to that used in the run which generated the hessian matrix.

If you wish to follow an IRC for a different isotope, use the \$MASS group. If you wish to follow the IRC in regular Cartesian coordinates, just enter unit masses for each atom. Note that CMODE and FREQ are a function of the atomic masses, so either regenerate FREQ and CMODE, or more simply, provide the correct \$HESS group.

Gradient Extremals

This section of the manual, as well as the source code to trace gradient extremals was written by Frank Jensen of the University of Aarhus.

A Gradient Extremal (GE) curve consists of points where the gradient norm on a constant energy surface is stationary. This is equivalent to the condition that the gradient is an eigenvector of the Hessian. Such GE curves radiate along all normal modes from a stationary point, and the GE leaving along the lowest normal mode from a minimum is the gentlest ascent curve. This is not the same as the IRC curve connecting a minimum and a TS, but may in some cases be close.

GEs may be divided into three groups: those leading to dissociation, those leading to atoms colliding, and those which connect stationary points. The latter class allows a determination of many (all?) stationary points on a PES by tracing out all the GEs. Following GEs is thus a semi-systematic way of mapping out stationary points. The disadvantages are:

- i) There are many (but finitely many!) GEs for a large molecule.
- ii) Following GEs is computationally expensive.
- iii) There is no control over what type of stationary point (if any) a GE will lead to.

Normally one is only interested in minima and TSs, but many higher order saddle points will also be found. Furthermore, it appears that it is necessary to follow GEs radiating also from TSs and second (and possibly also higher) order saddle point to find all the TSs.

A rather complete map of the extremals for the H₂CO potential surface is available in a paper which explains the points just raised in greater detail:

K.Bondensgaard, F.Jensen,
J.Chem.Phys. 104, 8025-8031(1996).

An earlier paper gives some of the properties of GEs:

D.K.Hoffman, R.S.Nord, K.Ruedenberg,
Theor. Chim. Acta 69, 265-279(1986).

There are two GE algorithms in GAMESS, one due to Sun and Ruedenberg (METHOD=SR), which has been extended to include the capability of locating bifurcation points and

turning points, and another due to Jorgensen, Jensen, and Helgaker (METHOD=JJH):

J. Sun, K. Ruedenberg, J.Chem.Phys. 98, 9707-9714(1993)

P. Jorgensen, H. J. Aa. Jensen, T. Helgaker
Theor. Chim. Acta 73, 55 (1988).

The Sun and Ruedenberg method consist of a predictor step taken along the tangent to the GE curve, followed by one or more corrector steps to bring the geometry back to the GE. Construction of the GE tangent and the corrector step requires elements of the third derivative of the energy, which is obtained by a numerical differentiation of two Hessians. This puts some limitations on which systems the GE algorithm can be used for. First, the numerical differentiation of the Hessian to produce third derivatives means that the Hessian should be calculated by analytical methods, thus only those types of wavefunctions where this is possible can be used. Second, each predictor/corrector step requires at least two Hessians, but often more. Maybe 20-50 such steps are necessary for tracing a GE from one stationary point to the next. A systematic study of all the GE radiating from a stationary point increases the work by a factor of $\sim 2 \cdot (3N-6)$. One should thus be prepared to invest at least hundreds, and more likely thousands, of Hessian calculations. In other words, small systems, small basis sets, and simple wavefunctions.

The Jorgensen, Jensen, and Helgaker method consists of taking a step in the direction of the chosen Hessian eigenvector, and then a pure NR step in the perpendicular modes. This requires (only) one Hessian calculation for each step. It is not suitable for following GEs where the GE tangent forms a large angle with the gradient, and it is incapable of locating GE bifurcations.

Although experience is limited at present, the JJH method does not appear to be suitable for following GEs in general (at least not in the current implementation). Experiment with it at your own risk!

The flow of the SR algorithm is as follows: A predictor geometry is produced, either by jumping away from a stationary point, or from a step in the tangent direction from the previous point on the GE. At the predictor geometry, we need the gradient, the Hessian, and the third derivative in the gradient direction. Depending on HSDFDB, this can be done in two ways. If .TRUE. the gradient is calculated, and two Hessians are calculated at

SNUMH distance to each side in the gradient direction. The Hessian at the geometry is formed as the average of the two displaced Hessians. This corresponds to a double-sided differentiation, and is the numerical most stable method for getting the partial third derivative matrix. If HSDFDB = .FALSE., the gradient and Hessian are calculated at the current geometry, and one additional Hessian is calculated at SNUMH distance in the gradient direction. This corresponds to a single-sided differentiation. In both cases, two full Hessian calculations are necessary, but HSDFDB = .TRUE. require one additional wavefunction and gradient calculation. This is usually a fairly small price compared to two Hessians, and the numerically better double-sided differentiation has therefore been made the default.

Once the gradient, Hessian, and third derivative is available, the corrector step and the new GE tangent are constructed. If the corrector step is below a threshold, a new predictor step is taken along the tangent vector. If the corrector step is larger than the threshold, the correction step is taken, and a new micro iteration is performed. DELCOR thus determines how closely the GE will be followed, and DPRED determine how closely the GE path will be sampled.

The construction of the GE tangent and corrector step involve solution of a set of linear equations, which in matrix notation can be written as $Ax=B$. The A-matrix is also the second derivative of the gradient norm on the constant energy surface.

After each corrector step, various things are printed to monitor the behavior: The projection of the gradient along the Hessian eigenvalues (the gradient is parallel to an eigenvector on the GE), the projection of the GE tangent along the Hessian eigenvectors, and the overlap of the Hessian eigenvectors with the mode being followed from the previous (optimized) geometry. The sign of these overlaps are not significant, they just refer to an arbitrary phase of the Hessian eigenvectors.

After the micro iterations has converged, the Hessian eigenvector curvatures are also displayed, this is an indication of the coupling between the normal modes. The number of negative eigenvalues in the A-matrix is denoted the GE index. If it changes, one of the eigenvalues must have passed through zero. Such points may either be GE bifurcations (where two GEs cross) or may just be "turning

points", normally when the GE switches from going uphill in energy to downhill, or vice versa. The distinction is made based on the B-element corresponding to the A-matrix eigenvalue = 0. If the B-element = 0, it is a bifurcation, otherwise it is a turning point.

If the GE index changes, a linear interpolation is performed between the last two points to locate the point where the A-matrix is singular, and the corresponding B-element is determined. The linear interpolation points will in general be off the GE, and thus the evaluation of whether the B-element is 0 is not always easy. The program additionally evaluates the two limiting vectors which are solutions to the linear sets of equations, these are also used for testing whether the singular point is a bifurcation point or turning point.

Very close to a GE bifurcation, the corrector step become numerically unstable, but this is rarely a problem in practice. It is a priori expected that GE bifurcation will occur only in symmetric systems, and the crossing GE will break the symmetry. Equivalently, a crossing GE may be encountered when a symmetry element is formed, however such crossings are much harder to detect since the GE index does not change, as one of the A-matrix eigenvalues merely touches zero. The program prints a message if the absolute value of an A-matrix eigenvalue reaches a minimum near zero, as such points may indicate the passage of a bifurcation where a higher symmetry GE crosses. Run a movie of the geometries to see if a more symmetric structure is passed during the run.

An estimate of the possible crossing GE direction is made at all points where the A-matrix is singular, and two perturbed geometries in the + and - direction are written out. These may be used as predictor geometries for following a crossing GE. If the singular geometry is a turning point, the + and - geometries are just predictor geometries on the GE being followed.

In any case, a new predictor step can be taken to trace a different GE from the newly discovered singular point, using the direction determined by interpolation from the two end point tangents (the GE tangent cannot be uniquely determined at a bifurcation point). It is not possible to determine what the sign of IFOLLOW should be when starting off along a crossing GE at a bifurcation, one will have to try a step to see if it returns to the bifurcation point or not.

In order to determine whether the GE index change it is necessary to keep track of the order of the A-matrix eigenvalues. The overlap between successive eigenvectors are shown as "Alpha mode overlaps".

Things to watch out for:

1) The numerical differentiation to get third derivatives requires more accuracy than usual. The SCF convergence should be at least 100 times smaller than SNUMH, and preferably better. With the default SNUMH of $10^{*(-4)}$ the SCF convergence should be at least $10^{*(-6)}$. Since the last few SCF cycles are inexpensive, it is a good idea to tighten the SCF convergence as much as possible, to maybe $10^{*(-8)}$ or better. You may also want to increase the integral accuracy by reducing the cutoffs (ITOL and ICUT) and possibly also try more accurate integrals (INTTYP=HONDO). The CUTOFF in \$TRNSFM may also be reduced to produce more accurate Hessians. Don't attempt to use a value for SNUMH below $10^{*(-6)}$, as you simply can't get enough accuracy. Since experience is limited at present, it is recommended that some tests runs are made to learn the sensitivity of these factors for your system.

2) GEs can be followed in both directions, uphill or downhill. When starting from a stationary point, the direction is implicitly given as away from the stationary point. When starting from a non-stationary point, the "+" and "-" directions (as chosen by the sign of IFOLLOW) refers to the gradient direction. The "+" direction is along the gradient (energy increases) and "-" is opposite to the gradient (energy decreases).

3) A switch from one GE to another may be seen when two GE come close together. This is especially troublesome near bifurcation points where two GEs actually cross. In such cases a switch to a GE with -higher- symmetry may occur without any indication that this has happened, except possibly that a very large GE curvature suddenly shows up. Avoid running the calculation with less symmetry than the system actually has, as this increases the likelihood that such switches occurring. Fix: alter DPRED to avoid having the predictor step close to the crossing GE.

4) "Off track" error message: The Hessian eigenvector which is parallel to the gradient is not the same as the one with the largest overlap to the previous Hessian mode. This usually indicate that a GE switch

has occurred (note that a switch may occur without this error message), or a wrong value for IFOLLOW when starting from a non-stationary point. Fix: check IFOLLOW, if it is correct then reduce DPRED, and possibly also DELCOR.

5) Low overlaps of A-matrix eigenvectors. Small overlaps may give wrong assignment, and wrong conclusions about GE index change. Fix: reduce DPRED.

6) The interpolation for locating a point where one of the A-matrix eigenvalues is zero fail to converge. Fix: reduce DPRED (and possibly also DELCOR) to get a shorter (and better) interpolation line.

7) The GE index changes by more than 1. A GE switch may have occurred, or more than one GE index change is located between the last and current point. Fix: reduce DPRED to sample the GE path more closely.

8) If SNRMAX is too large the algorithm may try to locate stationary points which are not actually on the GE being followed. Since GEs often pass quite near a stationary point, SNRMAX should only be increased above the default 0.10 after some consideration.

Continuum Solvation Methods

In a very thorough 1994 review of continuum solvation models, Tomasi and Persico divide the possible approaches to the treatment of solvent effects into four categories:

- a) virial equations of state, correlation functions
- b) Monte Carlo or molecular dynamics simulations
- c) continuum treatments
- d) molecular treatments

The Effective Fragment Potential method, documented in the following section of this chapter, falls into the latter category, as each EFP solvent molecule is modeled as a distinct object (discrete solvation). This section describes the four continuum models which are implemented in the standard version of GAMESS, and a fifth model which can be interfaced.

Continuum models typically form a cavity of some sort containing the solute molecule, while the solvent outside the cavity is thought of as a continuous medium and is categorized by a limited amount of physical data, such as the dielectric constant. The electric field of the charged particles comprising the solute interact with this background medium, producing a polarization in it, which in turn feeds back upon the solute's wavefunction.

Self Consistent Reaction Field (SCRF)

A simple continuum model is the Onsager cavity model, often called the Self-Consistent Reaction Field, or SCRF model. This represents the charge distribution of the solute in terms of a multipole expansion. SCRF usually uses an idealized cavity (spherical or ellipsoidal) to allow an analytic solution to the interaction energy between the solute multipole and the multipole which this induces in the continuum. This method is implemented in GAMESS in the simplest possible fashion:

- i) a spherical cavity is used
- ii) the molecular electrostatic potential of the solute is represented as a dipole only, except a monopole is also included for an ionic solute.

The input for this implementation of the Kirkwood-Onsager model is provided in \$SCRF.

Some references on the SCRF method are

1. J.G.Kirkwood J.Chem.Phys. 2, 351 (1934)

2. L.Onsager J.Am.Chem.Soc. 58, 1486 (1936)
3. O.Tapia, O.Goscinski Mol.Phys. 29, 1653 (1975)
4. M.M.Karelson, A.R.Katritzky, M.C.Zerner
Int.J.Quantum Chem., Symp. 20, 521-527 (1986)
5. K.V.Mikkelsen, H.Agren, H.J.Aa.Jensen, T.Helgaker
J.Chem.Phys. 89, 3086-3095 (1988)
6. M.W.Wong, M.J.Frisch, K.B.Wiberg
J.Am.Chem.Soc. 113, 4776-4782 (1991)
7. M.Szafran, M.M.Karelson, A.R.Katritzky, J.Koput,
M.C.Zerner J.Comput.Chem. 14, 371-377 (1993)
8. M.Karelson, T.Tamm, M.C.Zerner
J.Phys.Chem. 97, 11901-11907 (1993)

The method is very sensitive to the choice of the solute RADIUS, but not very sensitive to the particular DIELEC of polar solvents. The plots in reference 7 illustrate these points very nicely. The SCRF implementation in GAMESS is Zerner's Method A, described in the same reference. The total solute energy includes the Born term, if the solute is an ion. Another limitation is that a solute's electrostatic potential is not likely to be fit well as a dipole moment only, for example see Table VI of reference 5 which illustrates the importance of higher multipoles. Finally, the restriction to a spherical cavity may not be very representative of the solute's true shape. However, in the special case of a roundish molecule, and a large dipole which is geometry sensitive, the SCRF model may include sufficient physics to be meaningful:

M.W.Schmidt, T.L.Windus, M.S.Gordon
J.Am.Chem.Soc. 117, 7480-7486(1995).

Most cases should choose PCM (next section) over SCRF!!!

Polarizable Continuum Model (PCM)

A much more sophisticated continuum method, named the Polarizable Continuum Model, is also available. The PCM method places a solute in a cavity formed by a union of spheres centered on each atom. PCM includes a more exact treatment of the electrostatic interaction of the solute with the surrounding medium, on the cavity's surface. The computational procedure divides this surface into many small tesserae, each having a different "apparent surface charge", reflecting the solute's and other tesserae's electric field at each. These surface charges are the PCM model's "solvation effect" and make contributions to the energy and to the gradient of the solute.

Typically the cavity is defined as a union of atomic spheres, which should be roughly 1.2 times the atomic van der Waals radii. A technical difficulty caused by the penetration of the solute's charge density outside this cavity is dealt with by a renormalization. The solvent is characterized by its dielectric constant, surface tension, size, density, and so on. Procedures are provided not only for the computation of the electrostatic interaction of the solute with the apparent surface charges, but also for the cavitation energy, and for the dispersion and repulsion contributions to the solvation free energy.

Methodology for solving the Poisson equation to obtain the "apparent surface charges" has progressed from D-PCM to IEF-PCM to C-PCM over time, with the latter preferred. Iterative solvers require far less computer resources than direct solvers. Advancements have also been made in schemes to divide the surface cavity into tiny tesserae. As of fall 2008, the FIXPVA tessellation, which has smooth switching functions for tesserae near sphere boundaries, together with iterative C-PCM, gives very satisfactory geometry optimizations for molecules of 100 atoms. The FIXPVA tessellation was extended to work for cavitation (ICAV), dispersion (IDP), and repulsion (IREP) options in fall 2009, and dispersion/repulsion (IDISP) in spring 2010. Other procedures remain, and make the input seem complex, but their use is discouraged. Thus

```
$PCM SOLVNT=WATER $END
```

chooses iterative C-PCM (IEF=-10) and FIXPVA tessellation (METHOD=4 in \$TESCAV) to do basic electrostatics in an accurate fashion.

The main input group is \$PCM, with \$PCMCAV providing auxiliary cavity information. If any of the optional energy computations are requested in \$PCM, the additional input groups \$IEFPCM, \$NEWCAV, \$DISBS, or \$DISREP may be required.

It is useful to summarize the various cavities used by PCM, since as many as three cavities may be used:

- the basic cavity for electrostatics,
- cavity for cavitation energy, if ICAV=1,
- cavity for dispersion/repulsion, if IDISP=1.

The first and second share the same radii (see RADII in \$PCMCAV), which are scaled by ALPHA=1.2 for electrostatics, but are used unscaled for cavitation. The dispersion cavity is defined in \$DISREP with a separate set of atomic radii, and even solvent molecule radii! Only the electrostatics cavity can use any of the GEPOL-GB, GEPOL-AS

or recommended FIXPVA tessellation, while the other two use only the original GEPOL-GB.

Radii are an important part of the PCM parameterization. Their values can have a significant impact on the quality of the results. This is particularly true if the solute is charged, and thus has a large electrostatic interaction with the continuum. John Emsley's book "The Elements" is a useful source of van der Waals and other radii.

PCM is at heart a means of treating the electrostatic interactions between the solute's wavefunction and a dielectric model for the bulk solvent. The former is represented as an electron density from whatever quantum mechanical treatment is used for the solute, and the latter is a set of surface charges on the finite elements of the cavity (tessellation). This leaves out other important contributions to the solvation energy! These include the energy needed to make a hole in the solvent (cavitation energy), dispersion or repulsive interactions between the solute and solvent, and in the SMD model (see below) solvent structure changes such as would occur in the first solvation shell. Some empirical formulae for such "CDS" corrections are provided as keywords ICAV, IDISP, IREP/IDP, which may not work with all wavefunctions, and may not be compatible with gradients.

The SMD model gives an alternative set of such "CDS" corrections, which are compatible with nuclear gradients: see SMD=.TRUE. in \$PCM. A more detailed description of SMD is given in the paper cited below. The SMD solvent parameters is described (as of 2010) in

<http://comp.chem.umn.edu/solvation/mnsddb.pdf>

This gives numerical parameters, all built into GAMESS, as the various SOLX values, for SOLVNT=

ACETACID	CLPROPAN	PHOPH	EGME	E2PENTEN
ACETONE	OCLTOLUE	DPROAMIN	MEACETAT	PENTACET
ACETNTRL	M-CRESOL	DODECAN	MEBNZATE	PENTAMIN
ACETPHEN	O-CRESOL	MEG	MEBUTATE	PFB
ANILINE	CYCHEXAN	ETSH	MEFORMAT	BENZALCL
ANISOLE	CYCHEXON	ETHANOL	MIBK	PROPANAL
BENZALDH	CYCPENTN	ETOAC	MEPROPYL	PROPACID
BENZENE	CYCPNTOL	ETOME	ISOBUTOL	PROPANOL
BENZNTRL	CYCPNTON	EB	TERBUTOL	PROPNOL2
BENZYLCL	DECLNCIS	PHENETOL	NMEANILN	PROPNTOL
BRISOBUT	DECLNTRA	C6H5F	MECYCHEX	PROPENOL
BRBENZEN	DECLNMIX	FOCTANE	NMFMIXTR	PROPACET
BRETHANE	DECANE	FORMAMID	ISOHEXAN	PROPAMIN
BROMFORM	DECANOL	FORMACID	MEPYRID2	PYRIDINE

BROCTANE	EDB12	HEPTANE	MEPYRID3	C2CL4
BRPENTAN	DIBRMETN	HEPTANOL	MEPYRID4	THF
BRPROPA2	BUTYLETH	HEPTNON2	C6H5NO2	SULFOLAN
BRPROPAN	ODICLBNZ	HEPTNON4	C2H5NO2	TETRALIN
BUTANAL	EDC12	HEXADECN	CH3NO2	THIOPHEN
BUTACID	C12DCE	HEXANE	NTRPROP1	PHSH
BUTANOL	T12DCE	HEXNACID	NTRPROP2	TOLUENE
BUTANOL2	DCM	HEXANOL	ONTRTOLU	TBP
BUTANONE	ETHER	HEXANON2	NONANE	TCA111
BUTANTRL	ET2S	HEXENE	NONANOL	TCA112
BUTILE	DIETAMIN	HEXYNE	NONANONE	TCE
NBA	MI	C6H5I	OCTANE	ET3N
NBUTBENZ	DIPE	IOBUTANE	OCTANOL	TFE222
SBUTBENZ	DMDS	C2H5I	OCTANON2	TMBEN124
TBUTBENZ	DMSO	IOHEXDEC	PENTDECN	ISOCTANE
CS2	DMA	CH3I	PENTANAL	UNDECANE
CARBNTET	CISDMCHX	IOPENTAN	NPENTANE	M-XYLENE
CLBENZEN	DMF	IOPROPAN	PENTACID	O-XYLENE
SECBUTCL	DMEPEN24	CUMENE	PENTANOL	P-XYLENE
CHCL3	DMEPYR24	P-CYMENE	PENTNON2	XYLENEMX
CLHEXANE	DMEPYR26	MESITYLN	PENTNON3	
CLPENTAN	DIOXANE	METHANOL	PENTENE	

and provides a translation table to full chemical names, if you can't guess from the input choices given above. The translations can also be found in the source code. Two important things to note about SMD are:

a) the atomic radii are changed, so although the algorithms for electrostatics are those of standard PCM, the numerical results for the electrostatics do change.

b) SMD's parameterization was developed for IEF-PCM using GEPOLE tessellation with a fine grid: IEF=-3 and MTHALL=2, NTSALL=240. However it is considered acceptable to use SMD's parameters, unchanged, with C-PCM and with the FIXPVA tessellation, at default coarseness. Hence, input such as

```
$pcm solvnt=dmsol smd=.true. $end
```

is enough to carry out a SMD-style C-PCM treatment in DMSO.

c) The CDS correction involves cavitation, dispersion, and as a collective "solvent structure contribution" estimates for partial hydrogen bonding, repulsion, and deviation of the dielectric constant from its bulk value.

d) See also SMVLE in the more sophisticated SS(V)PE continuum model's description.

Solvation of course affects the non-linear optical properties of molecules. The PCM implementation extends RUNTYP=TDHF to include solvent effects. Both static and frequency dependent hyperpolarizabilities can be found. Besides the standard PCM electrostatic contribution, the

IREP and IDP keywords can be used to determine the effects of repulsion and dispersion on the polarizabilities.

The implementation of the PCM model in GAMESS has received considerable attention from Hui Li and Jan Jensen at the University of Iowa, Iowa State University, and University of Nebraska. This includes new iterative techniques to solving the surface charge problem, new tessellations that provide for numerically stable nuclear gradients, the implementation of C-PCM equations, the extension of PCM to all SCFTYPs and TDDFT, development of an interface with the EFP model (quo vadis), and heterogenous dielectric. Dmitri Fedorov at AIST has interfaced PCM to the FMO method (quo vadis), and reduced storage requirements.

Due to its sophistication, users of the PCM model are strongly encouraged to read the primary literature:

Of particular relevance to PCM in GAMESS:

- 1) "Continuum solvation of large molecules described by QM/MM: a semi-iterative implementation of the PCM/EFP interface"
H.Li, C.S.Pomelli, J.H.Jensen
Theoret.Chim.Acta 109, 71-84(2003)
- 2) "Improving the efficiency and convergence of geometry optimization with the polarizable continuum model: new energy gradients and molecular surface tessellation"
H.Li, J.H.Jensen J.Comput.Chem. 25, 1449-1462(2004)
- 3) "The polarizable continuum model interfaced with the Fragment Molecular Orbital method"
D.G.Fedorov, K.Kitaura, H.Li, J.H.Jensen, M.S.Gordon
J.Comput.Chem. 27, 976-985(2006)
- 4) "Energy gradients in combined Fragment Molecular Orbital and Polarizable Continuum Model (FMO/PCM)"
H.Li, D.G.Fedorov, T.Nagata, K.Kitaura, J.H.Jensen, M.S.Gordon J.Comput.Chem. 31, 778-790(2010)
- 5) "Continuous and smooth potential energy surface for conductor-like screening solvation model using fixed points with variable area"
P.Su, H.Li J.Chem.Phys. 130, 074109/1-13(2009)
- 6) "Heterogenous conductorlike solvation model"
D.Si, H.Li J.Chem.Phys. 131, 044123/1-8(2009)
- 7) "Quantum mechanical/molecular mechanical/continuum style solvation model: linear response theory, variational treatment, and nuclear gradients"
H.Li J.Chem.Phys. 131, 184103/1-8(2009)

8) "Smooth potential energy surface for cavitation, dispersion, and repulsion free energies in polarizable continuum model"

Y.Wang, H.Li J.Chem.Phys. 131, 206101/1-2(2009)

9) "Excited state geometry of photoactive yellow protein chromophore: a combined conductorlike polarizable continuum model and time-dependent density functional study"

Y.Wang, H.Li J.Chem.Phys. 133, 034108/1-11(2010)

Paper number 7 is about the treatment of QM systems with the solvation models EFP and/or C-PCM.

SMD and its CDS cavitation/dispersion/solvent structure corrections are described in
"Universal solution model based on solute electron density and on a continuum model of the solvent defined by the bulk dielectric constant and atomic surface tensions"

A.V.Marenich, C.J.Cramer, D.G.Truhlar
J.Phys.Chem.B 113, 6378-6396(2009)

General papers on PCM:

10) S.Miertus, E.Scrocco, J.Tomasi
Chem.Phys. 55, 117-129(1981)

11) J.Tomasi, M.Persico Chem.Rev. 94, 2027-2094(1994)

12) R.Cammi, J.Tomasi J.Comput.Chem. 16, 1449-1458(1995)

13) J.Tomasi, B.Mennucci, R.Cammi
Chem.Rev. 105, 2999-3093(2005)

The GEPOL-GB method for cavity construction:

14) J.L.Pascual-Ahuir, E.Silla, J.Tomasi, R.Bonaccorsi
J.Comput.Chem. 8, 778-787(1987)

Charge renormalization (see also ref. 12):

15) B.Mennucci, J.Tomasi J.Chem.Phys. 106, 5151-5158(1997)

Derivatives with respect to nuclear coordinates:
(energy gradient and hessian) See also paper 2 and 3.

16) R.Cammi, J.Tomasi J.Chem.Phys. 100, 7495-7502(1994)

17) R.Cammi, J.Tomasi J.Chem.Phys. 101, 3888-3897(1995)

18) M.Cossi, B.Mennucci, R.Cammi
J.Comput.Chem. 17, 57-73(1996)

Derivatives with respect to applied electric fields:
(polarizabilities and hyperpolarizabilities)

19) R.Cammi, J.Tomasi
Int.J.Quantum Chem. Symp. 29, 465-474(1995)

20) R.Cammi, M.Cossi, J.Tomasi
J.Chem.Phys. 104, 4611-4620(1996)

21) R.Cammi, M.Cossi, B.Mennucci, J.Tomasi

- J.Chem.Phys. 105, 10556-10564(1996)
22) B. Mennucci, C. Amovilli, J. Tomasi
Chem.Phys.Lett. 286, 221-225(1998)

Cavitation energy:

- 23) R.A.Pierotti Chem.Rev. 76, 717-726(1976)
24) J.Langlet, P.Claverie, J.Caillet, A.Pullman
J.Phys.Chem. 92, 1617-1631(1988)

Dispersion and repulsion energies:

- 25) F.Floris, J.Tomasi J.Comput.Chem. 10, 616-627(1989)
26) C.Amovilli, B.Mennucci
J.Phys.Chem.B 101, 1051-1057(1997)

Integral Equation Formalism PCM. The first of these deals with anisotropies, the last 2 with nuclear gradients.

- 27) E.Cances, B.Mennucci, J.Tomasi
J.Chem.Phys. 107, 3032-3041(1997)
28) B.Mennucci, E.Cances, J.Tomasi
J.Phys.Chem.B 101, 10506-17(1997)
29) B.Mennucci, R.Cammi, J.Tomasi
J.Chem.Phys. 109, 2798-2807(1998)
30) J.Tomasi, B.Mennucci, E.Cances
J.Mol.Struct.(THEOCHEM) 464, 211-226(1999)
31) E.Cances, B.Mennucci J.Chem.Phys. 109, 249-259(1998)
32) E.Cances, B.Mennucci, J.Tomasi
J.Chem.Phys. 109, 260-266(1998)

Conductor PCM (C-PCM):

- 33) V.Barone, M.Cossi J.Phys.Chem.A 102, 1995-2001(1998)
34) M.Cossi, N.Regà, G.Scalmani, V.Barone
J.Comput.Chem. 24, 669-681(2003)

C-PCM with TD-DFT:

- 35) M.Cossi, V.Barone J.Chem.Phys. 115, 4708-4717(2001)
See also paper #8 above for the coding in GAMESS.

At the present time, the PCM model in GAMESS has the following limitations:

- a) Any SCFTYP may be used (RHF to MCSCF). MP2 or DFT may be used with any of the RHF, UHF, and ROHF gradient programs. Closed shell TD-DFT excited state gradients may also be used.
CI and Coupled Cluster programs are not available.
- b) semi-empirical methods may not be used.
- c) the only other solvent method that may be used at used with PCM is the EFP model.
- d) point group symmetry is switched off internally

during PCM.

- e) The PCM model runs in parallel for IEF=3, -3, 10, or -10 and for all 5 wavefunctions (energy or gradient), but not for RUNTYP=TDHF jobs.
- f) D-PCM stores electric field integrals at normals to the surface elements on disk.
IEF-PCM and C-PCM using the explicit solver (+3 and +10) store electric potential integrals at normals to the surface on disk.
This is true even for direct AO integral runs, and the file sizes may be considerable (basis set size squared times the number of tesserae).
IEF-PCM and C-PCM with the iterative solvers do not store the potential integrals, when IDIRECT=1 in the \$PCMITR group (this is the default)
- g) nuclear derivatives are limited to gradients, although theory for hessians is given in paper 17.

* * *

The only PCM method prior to Oct. 2000 was D-PCM, which can be recovered by selecting IEF=0 and ICOMP=2 in \$PCM. The default PCM method between Oct. 2000 and May 2004 was IEF-PCM, recoverable by IEF=-3 (but 3 for non-gradient runs) and ICOMP=0. As of May 2004, the default PCM method was changed to C-PCM (IEF=-10, ICOMP=0). The extension of PCM to all SCFTYPs as of May 2004 involved a correction to the MCSCF PCM operator, so that it would reproduce RHF results when run on one determinant, meaning that it is impossible to reproduce prior MCSCF PCM calculations.

The cavity definition was GEPOL-GB (MTHALL=1 in \$TESCAV) prior to May 2004, GEPOL-AS (MTHALL=2) from then until September 2008, and FIXPVA (MTHALL=4) to the present time. The option for generation of 'extra spheres' (RET in \$PCM) was changed from 0.2 to 100.0, to suppress these, in June 2003.

* * *

In general, use of PCM electrostatics is very simple, as may be seen from exam31.inp supplied with the program.

The calculation shown next illustrates the use of some of the older PCM options. Since methane is non-polar, its internal energy change and the direct PCM electrostatic interaction is smaller than the cavitation, repulsion, and dispersion corrections. Note that the use of ICAV, IREP, and IDP are currently incompatible with gradients, so a

reasonable calculation sequence might be to perform the geometry optimization with PCM electrostatics turned on, then perform an additional calculation to include the other solvent effects, adding extra functions to improve the dispersion correction.

```
! calculation of CH4 (metano), in PCM water.
!
! This input reproduces the data in Table 2, line 6, of
! C.Amovilli, B.Mennucci J.Phys.Chem.B 101, 1051-7(1997)
! To do this, we must use many original PCM options.
!
! The gas phase FINAL energy is -40.2075980292
! The FINAL energy in PCM water is -40.2048210283
!
!                                     (lit.)
! FREE ENERGY IN SOLVENT          = -25234.89 KCAL/MOL
! INTERNAL ENERGY IN SOLVENT      = -25230.64 KCAL/MOL
! DELTA INTERNAL ENERGY           =      .01 KCAL/MOL ( 0.0)
! ELECTROSTATIC INTERACTION        =     -.22 KCAL/MOL (-0.2)
! PIEROTTI CAVITATION ENERGY      =      5.98 KCAL/MOL ( 6.0)
! DISPERSION FREE ENERGY          =     -6.00 KCAL/MOL (-6.0)
! REPULSION FREE ENERGY            =      1.98 KCAL/MOL ( 2.0)
! TOTAL INTERACTION                 =      1.73 KCAL/MOL ( 1.8)
! TOTAL FREE ENERGY IN SOLVENT= -25228.91 KCAL/MOL
!
$contrl scftyp=rhf runtyp=energy $end
$guess guess=huckel $end
$system mwords=2 $end
! the "W1 basis" input here exactly matches HONDO's DZP
$DATA
CH4...gas phase geometry...in PCM water
Td

Carbon      6.
  DZV
  D 1 ; 1 0.75 1.0

Hydrogen    1.  0.6258579976  0.6258579976  0.6258579976
  DZV 0 1.20 1.15 ! inner and outer scale factors
  P 1 ; 1 1.00 1.0

$END
! The reference cited used a value for H2O's solvent
! radius that differs from the built in value (RSOLV).
! The IEF, ICOMP, MTHALL, and RET keywords are set to
! duplicate the original code's published results,
! namely D-PCM and GEPOLE-GB. This run doesn't put in
! any "extra spheres" but we try that option (RET)
! like it originally would have.
```

```

$PCM      SOLVNT=WATER RSOLV=1.35 RET=0.2
          IEF=0 ICOMP=2 IDISP=0 IREP=1 IDP=1 ICAV=1 $end
$TESCAV MTHALL=1 $END
$NEWCAV IPTYPE=2 ITSNUM=540 $END
!      dispersion "W2 basis" uses exponents which are
!      1/3 of smallest exponent in "W1 basis" of $DATA.
$DISBS  NADD=11 NKTYP(1)=0,1,2, 0,1, 0,1, 0,1, 0,1
        XYZE(1)=0.0,0.0,0.0, 0.0511
            0.0,0.0,0.0, 0.0382
            0.0,0.0,0.0, 0.25
        1.1817023, 1.1817023, 1.1817023, 0.05435467
        1.1817023, 1.1817023, 1.1817023, 0.33333333
        -1.1817023, 1.1817023, -1.1817023, 0.05435467
        -1.1817023, 1.1817023, -1.1817023, 0.33333333
        1.1817023, -1.1817023, -1.1817023, 0.05435467
        1.1817023, -1.1817023, -1.1817023, 0.33333333
        -1.1817023, -1.1817023, 1.1817023, 0.05435467
        -1.1817023, -1.1817023, 1.1817023, 0.33333333 $end

```

SVPE and SS(V)PE.

The Surface Volume Polarization for Electrostatics (SVPE), and an approximation to SVPE called the Surface and Simulation of Volume Polarization for Electrostatics (SS(V)PE) are continuum solvation models. Compared to other continuum models, SVPE and SS(V)PE pay careful attention to the problems of escaped charge, the shape of the surface cavity, and to integration of the Poisson equation for surface charges.

The original references for what is now called the SVPE (surface and volume polarization for electrostatics) method are the theory paper:

"Charge penetration in Dielectric Models of Solvation"
D.M.Chipman, J.Chem.Phys. 106, 10194-10206 (1997)
and two implementation papers:

"Volume Polarization in Reaction Field Theory"
C.-G.Zhan, J.Bentley, D.M.Chipman
J.Chem.Phys. 108, 177-192 (1998)

"New Formulation and Implementation for Volume Polarization in Dielectric Continuum Theory"
D.M.Chipman, J.Chem.Phys. 124, 224111-1/10 (2006)
which should be cited in any publications that utilize the SVPE code.

There are two options to include with SS(V)PE or SVPE additional models that describe short-range solute-solvent

interactions to achieve a more complete description of solvation energies. Both have their keywords merged into the \$SVP input group for convenience. One option to include short-range interactions is CMIRS1.0 (Composite Method for Implicit Representation of Solvent, Version 1.0) that combines the SS(V)PE dielectric continuum model with the DEFESR (Dispersion, Exchange, and Field-Extremum Short-Range) model. A complete account of CMIRS1.0 with application to hydration energies is given in:

"Hydration Energy from a Composite Method for Implicit Representation of Solvent"

A.Pomogaeva, D.M.Chipman

J.Chem.Theory Comput. 10, 211-219(2014)

which should be referenced in any publication that uses the model. References to earlier works that develop the individual components of the DEFESR parts of the full CMIRS recipe are:

"Modeling short-range contributions to hydration energies with minimal parameterization"

A.Pomogaeva, D.W.Thompson, and D.M.Chipman

Chem.Phys.Lett. 511, 161-165(2011).

"Field-Extremum Model for Short-Range Contributions to Hydration Free Energy"

A.Pomogaeva and D.M.Chipman

J.Chem.Theory Comput. 7, 3952-3960(2011).

"New Implicit Solvation Models for Dispersion and Exchange Energies"

A.Pomogaeva and D.M.Chipman

J.Phys.Chem.A, 117, 5812-5820 (2013).

The other option to include short-range interactions is the SMVLE (solvation model with volume and local electrostatics) as described in

"Free energies of solvation with surface, volume, and local electrostatic effects and atomic surface tensions to represent the first solvation shell"

J.Liu, C.P.Kelly, A.C.Goren, A.V.Marenich,

C.J.Cramer, D.G.Truhlar, C.-G. Zhan

J.Chem.Theory Comput. 6, 1109-1117(2010).

Further information on the performance of SVPE and of SS(V)PE can be found in:

"Comparison of Solvent Reaction Field Representations"

D.M.Chipman, Theor.Chem.Acc. 107, 80-89 (2002).

Details of the SS(V)PE convergence behavior and programming strategy are in:

"Implementation of Solvent Reaction Fields for Electronic Structure" D.M.Chipman, M.Dupuis,

Theor.Chem.Acc. 107, 90-102 (2002).

The SMVLE option (solvation model with volume and local electrostatics) is described in

"Free energies of solvation with surface, volume, and local electrostatic effects and atomic surface tensions to represent the first solvation shell" J.Liu, C.P.Kelly, A.C.Goren, A.V.Marenich, C.J.Cramer, D.G.Truhlar, C.-G. Zhan J.Chem.Theory Comput. 6, 1109-1117(2010).

The SVPE and SS(V)PE models are like PCM and COSMO in that they treat solvent as a continuum dielectric residing outside a molecular-shaped cavity, determining the apparent charges that represent the polarized dielectric by solving Poisson's equation. The main difference between SVPE and SS(V)PE is in treatment of volume polarization effects that arise because of the tail of the electronic wave function that penetrates outside the cavity, sometimes referred to as the "escaped charge." SVPE treats volume polarization effects explicitly by including apparent charges in the volume outside the cavity as well as on the cavity surface. With a sufficient number of grid points, SVPE can then provide an exact treatment of charge penetration effects. SS(V)PE, like PCM and COSMO, is an approximate treatment that only uses apparent charges located on the cavity surface. The SS(V)PE equation is particularly designed to simulate as well as possible the influence of the missing volume charges. For more information on the similarities and differences of the SVPE and SS(V)PE models with other continuum methods, see the paper "Comparison of Solvent Reaction Field Representations" cited just above.

In addition, the cavity construction and Poisson solver used in this implementation of SVPE and SS(V)PE also receive careful numerical treatment. For example, the cavity may be chosen to be an isodensity contour surface, and the Lebedev grids for the Poisson solver can be chosen very densely. The Lebedev grids used for surface integration are taken from the Fortran translation by C. van Wuelen of the original C programs developed by D. Laikov. They were obtained from the CCL web site www.ccl.net/cca/software/SOURCES/FORTRAN/Lebedev-Laikov-Grids. A recent leading reference is V. I. Lebedev and D. N. Laikov, Dokl. Math. 59, 477-481 (1999). All these grids have octahedral symmetry and so are naturally adapted for any solute having an Abelian point group. The larger and/or the less spherical the solute may be, the more surface points are needed to get satisfactory precision in the results. Further experience will be required to develop detailed recommendations for this parameter. Values as small as 110 are usually sufficient for simple diatomics

and triatomics. The default value of 1202 has been found adequate to obtain the energy to within 0.1 kcal/mol for solutes the size of monosubstituted benzenes. The SVPE method uses additional layers of points outside the cavity. Typically just two layers are sufficient to converge the direct volume polarization contribution to better than 0.1 kcal/mol.

The SVPE and SS(V)PE codes both report the amount of solute charge penetrating outside the cavity as calculated by Gauss' Law. The SVPE code additionally reports the same quantity as alternatively calculated from the explicit volume charges, and any substantial discrepancy between these two determinations indicates that more volume polarization layers should have been included for better precision. The energy contribution from the outermost volume polarization layer is also reported. If it is significant then again more layers should have been included. However, these tests are only diagnostic. Passing them does not guarantee that enough layers are included.

The SVPE and SS(V)PE models treat the electrostatic interaction between a quantum solute and a classical dielectric continuum solvent. No treatment is yet implemented for cavitation, dispersion, or any of a variety of other specific solvation effects. Note that corrections for these latter effects that might be reported by other programs are generally not transferable. The reason is that they are usually parameterized to improve the ultimate agreement with experiment. In addition to providing corrections for the physical effects advertised, they therefore also inherently include contributions that help to make up for any deficiencies in the electrostatic description. Consequently, they are appropriate only for use with the particular electrostatic model in which they were originally developed.

Analytic nuclear gradients are not yet available for the SVPE or SS(V)PE energy, but numerical differentiation will permit optimization of small solute molecules. Wavefunctions may be any of the SCF type: RHF, UHF, ROHF, GVB, and MCSCF, or the DFT analogs of some of these. In the MCSCF implementation, no initial wavefunction is available so the solvation code does not kick in until the second iteration.

We close with a SVPE example. The gas phase energy, obtained with no \$SVP group, is -207.988975, and the run

just below gives the SVPE energy -208.006282. The free energy of solvation, -10.860 kcal/mole, is the difference of these, and is quoted at the right side of the 3rd line from the bottom of Table 2 in the paper cited. The "REACTION FIELD FREE ENERGY" for SVPE is -12.905 kcal/mole, which is only part of the solvation free energy. There is also a contribution due to the SCRF procedure polarizing the wave function from its gas phase value, causing the solute internal energy in dielectric to differ from that in gas. Evaluating this latter contribution is what requires the separate gas phase calculation. Changing the number of layers (NVLPL) to zero produces the SS(V)PE approximation to SVPE, $E = -208.006208$.

```

!               SVPE solvation test...acetamide
!   reproduce data in Table 2 of the paper on SVPE,
!   D.M.Chipman  J.Chem.Phys. 124, 224111/1-10(2006)
!
$contrl scftyp=rhf  runtyp=energy  $end
$system mwords=4 $end
$basis  gbasis=n31  ngauss=6  ndfunc=1  npfunc=1  $end
$guess  guess=moread  norb=16  $end
$scf     nconv=8  $end
$svp     nvlpl=3  rhoiso=0.001  dielst=78.304  nptleb=1202  $end
$data
CH3CONH2  cgz  geometry  RHF/6-31G(d,p)
C1
C           6.0           1.361261      -0.309588      -0.000262
C           6.0          -0.079357       0.152773      -0.005665
H           1.0           1.602076      -0.751515       0.962042
H           1.0           1.537200      -1.056768      -0.767127
H           1.0           2.002415       0.542830      -0.168045
O           8.0          -0.387955       1.310027       0.002284
N           7.0          -1.002151      -0.840834      -0.011928
H           1.0          -1.961646      -0.589397       0.038911
H           1.0          -0.752774      -1.798630       0.035006
$end
gas phase vectors, E(RHF)=      -207.9889751769
$VEC
1 1 1.18951670E-06 1.74015997E-05
...snipped...
$END

```

The example just above can be changed to a CMIRS run, by changing NVLPL to zero and also requesting calculation of the DEFESR contributions. The result should be a CMIRS1.0 total free energy of -208.013517. The input should (1) in \$CONTRL, specify DFTTYP=HFX, which requests a Hartree-Fock calculation, but sets up DFT grids,

(2) in \$DFT, request a very accurate grid
\$dft method=grid nrad=96 nleb=1202 nrad0=96 nleb0=1202 \$end
(3) in \$SVP, invoke the keyword IDEF=1.

Adding the keyword EGAS=-207.9889751769 to any of these examples allows reporting of the free energy of solvation (-10.860 kcal/mol for SVPE, -10.814 for SS(V)PE, -15.400 for CMIRS1.0).

Conductor-like screening model (COSMO)

The COSMO (conductor-like screening model) represents a different approach for carrying out polarized continuum calculations. COSMO was originally developed by Andreas Klamt, with extensions to ab initio computation in GAMESS by Kim Baldridge.

In the COSMO method, the surrounding medium is modeled as a conductor rather than as a dielectric in order to establish the initial boundary conditions. The assumption that the surrounding medium is well modeled as a conductor simplifies the electrostatic computations and corrections may be made a posteriori for dielectric behavior.

The original model of Klamt was introduced using a molecular shaped cavity, which had open parts along the crevices of intersecting atomic spheres. While having considerable technical advantages, this approximation causes artifacts in the context of the more generalized theory, so the current method for cavity construction includes a closure of the cavity to eliminate crevices or pockets.

Two methodologies are implemented for treatment of the outlying charge errors (OCE). The default is the well-established double cavity procedure using a second, larger cavity around the first one, and calculates OCE through the difference between the potential on the inner and the outer cavity. The second involves the calculation of distributed multipoles up to hexadecapoles to represent the entire charge distribution of the molecule within the cavity.

The COSMO model accounts only for the electrostatic interactions between solvent and solute. Klamt has proposed a novel statistical scheme to compute the full solvation free energy for neutral solutes, COSMO-RS, which is formulated for GAMESS by Peverati, Potier and Baldridge,

and is available as external plugin to the COSMOtherm program by COSMOlogic GmbH&Co.

The iterative inclusion of non-electrostatic effects is also possible right after a COSMO-RS calculation. The DCOSMO-RS approach was implemented in GAMESS by Peverati, Potier, and Baldridge, and more information is available on Baldridge website at:

<http://ocikbws.uzh.ch/gamess/>

The simplicity of the COSMO model allows computation of gradients, allowing optimization within the context of the solvent. The method is programmed for RHF and UHF, all corresponding kinds of DFT (including DFT-D), and the corresponding MP2, energy and gradient.

Some references on the COSMO model are:

- A.Klamt, G.Schuurman
J.Chem.Soc.Perkin Trans 2, 799-805(1993)
- A.Klamt J.Phys.Chem. 99, 2224-2235(1995)
- K.Baldridge, A.Klamt
J.Chem.Phys. 106, 6622-6633 (1997)
- V.Jonas, K.Baldridge
J.Chem.Phys. 113, 7511-7518 (2000)
- L.Gregerson, K.Baldridge
Helv.Chim.Acta 86, 4112-4132 (2003)
- R.Peverati, Y.Potier, K.Baldridge
TO BE PUBLISHED SOON

Additional references on the COSMO-RS model, with explanation of the methodology and program can be found:

- A.Klamt, F.Eckert, W.Arlt
Annu.Rev.Chem.Biomol.Eng. 1, (2010)

The Effective Fragment Potential Method

The basic idea behind the effective fragment potential (EFP) method is to replace the chemically inert part of a system by EFPs, while performing a regular ab initio calculation on the chemically active part. Here "inert" means that no covalent bond breaking process occurs. This "spectator region" consists of one or more "fragments", which interact with the ab initio "active region" through non-bonded interactions, and so of course these EFP interactions affect the ab initio wavefunction. The EFP particles can be closed shell or open shell (high spin ROHF) based potentials. The "active region" can use nearly every kind of wavefunction available in GAMESS.

A simple example of an active region might be a solute molecule, with a surrounding spectator region of solvent molecules represented by fragments. Each discrete solvent molecule is represented by a single fragment potential, in marked contrast to continuum models for solvation.

The quantum mechanical part of the system is entered in the \$DATA group, along with an appropriate basis. The EFPs defining the fragments are input by means of a \$EFRAG group, and one or more \$FRAGNAME groups describing each fragment's EFP. These groups define non-bonded interactions between the ab initio system and the fragments, and also between the fragments. The former interactions enter via one-electron operators in the ab initio Hamiltonian, while the latter interactions are treated by analytic functions. The only electrons explicitly treated (with basis functions used to expand occupied orbitals) are those in the active region, so there are no new two electron terms. Thus the use of EFPs leads to significant time savings, compared to full ab initio calculations on the same system.

There are two types of EFP available in GAMESS, EFP1 and EFP2. EFP1, the original method, employs a fitted repulsive potential. EFP1 is primarily used to model water molecules to study aqueous solvation effects, at the RHF/DZP or DFT/DZP (specifically, B3LYP) levels, see references 1-3 and 26, respectively. EFP2 is a more general method that is applicable to any species, including water, and its repulsive potential is obtained from first principles. EFP2 has been extended to include other effects as well, such as charge transfer and dispersion.

EFP2 forms the basis of the covalent EFP method described below for modeling enzymes, see reference 14.

Parallelization of the EFP1 and EFP2 models is described in reference 32.

MD simulations with EFP are described in reference 31.

The ab initio/EFP1, or pure EFP system can be wrapped in a Polarizable Continuum Model, see references 23, 43, and 50.

terms in an EFP

The non-bonded interactions currently implemented are:

- 1) Coulomb interaction. The charge distribution of the fragments is represented by an arbitrary number of charges, dipoles, quadrupoles, and octopoles, which interact with the ab initio hamiltonian as well as with multipoles on other fragments (see reference 2 and 18). It is possible to use a screening term that accounts for the charge penetration (reference 17 and 42). This screening term is automatically included for EFP1. Typically the multipole expansion points are located on atomic nuclei and at bond midpoints.
- 2) Dipole polarizability. An arbitrary number of dipole polarizability tensors can be used to calculate the induced dipole on a fragment due to the electric field of the ab initio system as well as all the other fragments. These induced dipoles interact with the ab initio system as well as the other EFPs, in turn changing their electric fields. All induced dipoles are therefore iterated to self-consistency. Typically the polarizability tensors are located at the centroid of charge of each localized orbital of a fragment. See reference 41.
- 3) Repulsive potential. Two different forms are used in EFP1: one for ab initio-EFP repulsion and one for EFP-EFP repulsion. The form of the potentials is empirical, and consists of distributed Gaussian or exponential functions, respectively. The primary contribution to the repulsion is the quantum mechanical exchange repulsion, but the fitting technique used to develop this term also includes the effects of charge transfer. Typically these fitted potentials are located on each atomic nucleus within the fragment (see reference 3). In EFP2, polarization energies

can also be augmented by screening terms, analogous to the electrostatic screening, to prevent "polarization collapse" (MS in preparation)

For EFP2, the third term is divided into separate analytic formulae for different physical interactions:

- a) exchange repulsion
- b) dispersion
- c) charge transfer

A summary of EFP2, and its contrast to EFP1 can be found in reference 18 and 44. The repulsive potential for EFP2 is based on an overlap expansion using localized molecular orbitals, as described in references 5, 6, and 9.

Dispersion energy is described in reference 34, and charge transfer in reference 39 (which supercedes reference 22's formulae).

EFP2 potentials have no fitted parameters, and can be automatically generated during a RUNTYP=MAKEFP job, as described below.

constructing an EFP1

RUNTYP=MOROKUMA assists in the decomposition of inter-molecular interaction energies into electrostatic, polarization, charge transfer, and exchange repulsion contributions. This is very useful in developing EFPs since potential problems can be attributed to a particular term by comparison to these energy components for a particular system.

A molecular multipole expansion can be obtained using \$ELMOM. A distributed multipole expansion can be obtained by either a Mulliken-like partitioning of the density (using \$STONE) or by using localized molecular orbitals (\$LOCAL: DIPDCM and QADDCM). The dipole polarizability tensor can be obtained during a Hessian run (\$CPHF), and a distributed LMO polarizability expression is also available (\$LOCAL: POLDCM).

In EFP1, the repulsive potential is derived by fitting the difference between ab initio computed intermolecular interaction energies, and the form used for Coulomb and polarizability interactions. This difference is obtained at a large number of different interaction geometries, and is then fitted. Thus, the repulsive term is implicitly a function of the choices made in representing the Coulomb

and polarizability terms. Note that GAMESS currently does not provide a way to obtain these EFP1 repulsive potential.

Since a user cannot generate all of the EFP1 terms necessary to define a new \$FRAGNAME group using GAMESS, in practice the usage of EFP1 is limited to the internally stored H2ORHF or H2ODFT potentials mentioned below.

constructing an EFP2

As noted above, the repulsive potential for EFP2 is derived from a localized orbital overlap expansion. It is generally recommended that one use at least a double zeta plus diffuse plus polarization basis set, e.g. 6-31++G(d,p) to generate the EFP2 repulsive potential. However, it has been observed that 6-31G(d) works reasonably well due to a fortuitous cancellation of errors. The EFP2 potential for any molecule can be generated as follows:

- (a) Choose a basis set and geometry for the molecule of interest. The geometry is ordinarily optimized at your choice of Hartree-Fock/MP2/CCSD(T), with your chosen basis set, but this is not a requirement. It is good to recall, however, that EFP internal geometries are fixed, so it is important to give some thought to the chosen geometry.
- (b) Perform a RUNTYP=MAKEFP run for the chosen molecule using the chosen geometry in \$DATA and the chosen basis set in \$BASIS. This will generate the entire EFP2 potential in the run's .efp file. The only user-defined variable that must be filled in is changing the FRAGNAME's group name, to \$C2H5OH or \$DMSO, etc. This step can use RHF or ROHF to describe the electronic structure of the system.
- (c) Transfer the entire fragment potential for the molecule to any input file in which this fragment is to be used. Since the internal geometry of an EFP is fixed, one need only specify the first three atoms of any fragment in order to position them in \$EFRAG. Coordinates of any other atoms in the rigid fragment will be automatically determined by the program.

If the EFP contains less than three atoms, you can still generate a fragment potential. After a normal MAKEFP run, add dummy atoms (e.g. in the X and/or Y directions) with zero nuclear charges, and add corresponding dummy bond midpoints too. Carefully insert zero entries in the multipole sections, and in the electrostatic screening

sections, for each such dummy point, but don't add data to any other kind of EFP term such as polarizability. This trick gives the necessary 3 points for use in \$EFRAG groups to specify "rotational" positions of fragments.

current limitations

1. For EFP1, the energy and energy gradient are programmed, which permits RUNTYP=ENERGY, GRADIENT, and numerical HESSIAN. The necessary programming to use the EFP gradients to move on the potential surface are programmed for RUNTYP=OPTIMIZE, SADPOINT, IRC, and VSCF, but the other gradient based potential surface explorations such as DRC are not yet available. Finally, RUNTYP=PROP is also permissible.

For EFP2, the gradient terms for ab initio-EFP interactions have not yet been coded, so geometry optimizations are only sensible for a COORD=FRAGONLY run; that is, a run in which only EFP2 fragments are present.

2. The ab initio part of the system must be treated with RHF, ROHF, UHF, the open shell SCF wavefunctions permitted by the GVB code, or MCSCF. DFT analogs of RHF, ROHF, and UHF may also be used. Correlated methods such as MP2 and CI should not be used.

3. EFPs can move relative to the ab initio system and relative to each other, but the internal structure of an EFP is frozen.

4. The boundary between the ab initio system and EFP1's must not be placed across a chemical bond. However, see the discussion below regarding covalent bonds.

5. Calculations must be done in C1 symmetry at present.

6. Reorientation of the fragments and ab initio system is not well coordinated. If you are giving Cartesian coordinates for the fragments (COORD=CART in \$EFRAG), be sure to use \$CONTRL's COORD=UNIQUE option so that the ab initio molecule is not reoriented.

7. If you need IR intensities, you have to use NVIB=2. The potential surface is usually very soft for EFP motions, and double differenced Hessians should usually be obtained.

practical hints for using EFPs

At the present time, we have only two internally stored EFP potentials suitable for general use. These model water, using the fragment name H2ORHF or H2ODFT. The H2ORHF numerical parameters are improved values over the values which were presented and used in reference 2, and they also include the improved EFP-EFP repulsive term defined in reference 3. The H2ORHF water EFP was derived from RHF/DH(d,p) computations on the water dimer system. When you use it, therefore, the ab initio part of your system should be treated at the SCF level, using a basis set of the same quality (ideally DH(d,p), but probably other DZP sets such as 6-31G(d,p) will give good results as well). Use of better basis sets than DZP with this water EFP has not been tested. Similarly, H2ODFT was developed using B3LYP/DZP water wavefunctions, so this should be used (rather than H2ORHF) if you are using DFT to treat the solute. Since H2ODFT water parameters are obtained from a correlated calculation, they can also be used when the solute is treated by MP2.

As noted, effective fragments have frozen internal geometries, and therefore only translate and rotate with respect to the ab initio region. An EFP's frozen coordinates are positioned to the desired location(s) in \$EFRAG as follows:

- a) the corresponding points are found in \$FRAGNAME.
- b) Point -1- in \$EFRAG and its FRAGNAME equivalent are made to coincide.
- c) The vector connecting -1- and -2- is aligned with the corresponding vector connecting FRAGNAME points.
- d) The plane defined by -1-, -2-, and -3- is made to coincide with the corresponding FRAGNAME plane.

Therefore the 3 points in \$EFRAG define only the relative position of the EFP, and not its internal structure. So, if the "internal structure" given by points in \$EFRAG differs from the true values in \$FRAGNAME, then the order in which the points are given in \$EFRAG can affect the positioning of the fragment. It may be easier to input water EFPs if you use the Z-matrix style to define them, because then you can ensure you use the actual frozen geometry in your \$EFRAG. Note that the H2ORHF EFP uses the frozen geometry $r(\text{OH})=0.9438636$, $a(\text{HOH})=106.70327$, and the names of its 3 fragment points are Z01, ZH2, ZH3.

* * *

Building a large cluster of EFP particles by hand can be tedious. The RUNTYP=GLOBOP program described below has an option for constructing dense clusters. The method tries to place particles near the origin, but not colliding with other EFP particles already placed there, so that the clusters grow outwards from the center. Here are some ideas:

- a) place 100 water molecules, all with the same coords in \$EFRAG. This will build up a droplet of water with particles close together, but not on top of each other, with various orientations.
- b) place 16 waters (same coords, all first) followed by 16 methanols (also sharing their same coords, after all waters). A 50-50 mixture of 32 molecules will be created, if you choose the default of picking the particles randomly from the initial list of 32.
- c) to solvate a solute, create a dummy \$FRAGNAME group for the solute that is to be modeled as ab initio, so the GLOBOP run thinks the solute is an EFP. Its potential can consist of just monopoles with zero charges. Place the solute first in \$EFRAG, with as many solute particles as you like after it. Of course the latter can all have the same coordinates. Pick the option to preserve the order of particles, so the solute is kept at the center of the cluster. After the check run gives you coordinates, move the solute from \$EFRAG to \$DATA.

Example, allowing the random cluster to have 20 geometry optimization steps:

```
$contrl runtyp=globop coord=fragonly $end
$globop rndini=.true. riord=rand mcmmin=.true.
      mctyp=4 nblock=0 $end
$statpt nstep=20 $end
$efrag
coord=cart
FRAGNAME=WATER
O1   -2.8091763203009   -2.1942725073400   -0.2722207394107
H2   -2.3676165499399   -1.6856118830379   -0.9334073942601
H3   -2.1441965467625   -2.5006167998896    0.3234583094693
...repeat this 15 more times...
FRAGNAME=MeOH
O1    4.9515153249      .4286994611      1.3368662306
H2    5.3392575544      .1717424606      3.0555957053
C3    6.2191743799      2.5592349960      .4064662379
H4    5.7024200977      2.7548960076     -1.5604873643
H5    5.6658856694      4.2696553371      1.4008542042
H6    8.2588049857      2.3458272252      .5282762681
...repeat 15 more times...
$end
```



```
$water
...give a full EFP2 potential for water...
$end
$meoh
...give a full EFP2 potential for methanol...
$end
```

Note that the random cluster generation now proceeds into a full Monte Carlo simulation.

* * *

The translations and rotations of EFPs with respect to the ab initio system and one another are automatically quite soft degrees of freedom. After all, the EFP model is meant to handle weak interactions! Therefore the satisfactory location of structures on these flat surfaces will require use of a tight convergence on the gradient: OPTTOL=0.00001 in the \$STATPT group.

The effect of a bulk continuum surrounding the solute plus EFP waters can be obtained by using the PCM model, see reference 23 and 43. To do this, simply add a \$PCM group to your input, in addition to the \$EFRAG. The simultaneous use of EFP and PCM allows for gradients, so geometry optimization can be performed.

global optimization

If there are a large number of particles to move (EFP and/or FMO and/or atom groups), it is difficult to locate the lowest energy structures by hand. Typically these are numerous, and one would like to have a number of them, not just the very lowest energy. The RUNTYP of GLOBOP contains a Monte Carlo procedure to generate a random set of starting structures to look for those with the lowest energy at a single temperature. If desired, a simulated annealing protocol to cool the temperature may be used. These two procedures may be combined with a local minimum search, at some or all of the randomly generated structures. The local minimum search is controlled by the usual geometry optimizer, namely \$STATPT input, and thus permits the optimization of any ab initio atoms.

The Monte Carlo procedure by default uses a Metropolis algorithm to move just one of the fragments. The method of Parks to move all fragments simultaneously is no longer implemented.

The present program was used to optimize the structure of water clusters. Let us consider the case of the twelve water cluster, for which the following ten structures were published by Day, Pachter, Gordon, and Merrill:

- | | | | |
|--------------|-----------|----------------|-----------|
| 1. (D2d)2 | -0.170209 | 6. (D2d)(C2) | -0.167796 |
| 2. (D2d)(S4) | -0.169933 | 7. S6 | -0.167761 |
| 3. (S4)2 | -0.169724 | 8. cage b | -0.167307 |
| 4. D3 | -0.168289 | 9. cage a | -0.167284 |
| 5. (C1c)(Cs) | -0.167930 | 10. (C1c)(C1c) | -0.167261 |

A test input using Metropolis style Monte Carlo to examine 300 geometries at each temperature value, using simulated annealing cooling from 200 to 50 degrees, and with local minimization every 10 structures was run ten times. Each run sampled about 7000 geometries. One simulation found structure 2, while two of the runs found structure 3. The other seven runs located structures with energy values in the range -0.163 to -0.164. In all cases the runs began with the same initial geometry, but produced different results due to the random number generation used in the Monte Carlo. Clearly one must try a lot of simulations to be confident about having found most of the low energy structures. In particular, it is good to try more than one initial structure, unlike what was done in this test.

Ab initio atoms can be addressed using FMO, either in multiple fragments, or perhaps a single large fragment. Alternatively, ab initio atoms can be put into groups and used directly in globop, which for small systems has a lower overhead than FMO.

Monte Carlo references:

- N.Metropolis, A.Rosenbluth, A.Teller
J.Chem.Phys. 21, 1087(1953).
- G.T.Parks Nucl.Technol. 89, 233(1990).

Monte Carlo with local minimization:

- Z.Li, H.A.Scheraga
Proc.Nat.Acad.Sci. USA 84, 6611(1987).

Simulated annealing reference:

- S.Kirkpatrick, C.D.Gelatt, M.P.Vecchi
Science 220, 671(1983).

The present program is described in reference 15. It is patterned on the work of

- D.J.Wales, M.P.Hodges Chem.Phys.Lett. 286, 65-72 (1998).

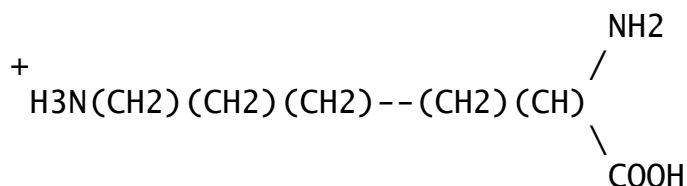
QM/MM across covalent bonds

Recent work by Visvaldas Kairys and Jan Jensen has made it possible to extend the EFP methodology beyond the simple solute/solvent case described above. When there is a covalent bond between the portion of the system to be modeled by quantum mechanics, and the portion which is to be treated by EFP multipole and polarizability terms, an additional layer is needed in the model. The covalent linkage is not so simple as the interactions between closed shell solute and solvent molecules. The "buffer zone" between the quantum mechanics and the EFP consists of frozen nuclei, and frozen localized orbitals, so that the quantum mechanical region sees a orbital representation of the closest particles, and multipoles etc. beyond that. Since the orbitals in the buffer zone are frozen, it need extend only over a few atoms in order to keep the orbitals in the fully optimized quantum region within that region.

The general outline of this kind of computation is as follows:

- a) a full quantum mechanics computation on a system containing the quantum region, the buffer region, and a few atoms into the EFP region, to obtain the frozen localized orbitals in the buffer zone. This is called the "truncation run".
- b) a full quantum mechanics computation on a system with all quantum region atoms removed, and with the frozen localized orbitals in the buffer zone. The necessary multipole and polarizability data to construct the EFP that will describes the EFP region will be extracted from the wavefunction. This is called the "MAKEFP run". It is possible to use several such runs if the total EFP region is quite large.
- c) The intended QM/MM run(s), after combining the information from these first two types of runs.

As an example, consider a protonated lysine residue which one might want to consider quantum mechanically in a protein whose larger parts are to be treated with an EFP. The protonated lysine is



The bonds which you see drawn show how the molecule is partitioned between the quantum mechanical side chain, a

CH₂CH group in the buffer zone, and eventually two different EFPs may be substituted in the area of the NH₂ and COOH groups to form the protein backbone.

The "truncation run" will be on the entire system as you see it, with the 13 atoms in the side chain first in \$DATA, the 5 atoms in the buffer zone next in \$DATA, and the simplified EFP region at the end. This run will compute the full quantum wavefunction by RUNTYP=ENERGY, followed by the calculation of localized orbitals, and then truncation of the localized orbitals that are found in the buffer zone so that they contain no contribution from AOs outside the buffer zone. The key input groups for this run are

```
$contrl
```

```
$struncn doproj=.true. plain=.true. natab=13 natbf=5 $end
```

This will generate a total of 6 localized molecular orbitals in the buffer zone (one CC, three CH, two 1s inner shells), expanded in terms of atomic orbitals located only on those atoms.

The truncation run prepares template input files for the next run, including adjustments of nuclear charges at boundaries, etc.

The "MAKEFP" run drops all 13 atoms in the quantum region, and uses the frozen orbitals just prepared to obtain a wavefunction for the EFP region. The carbon atom in the buffer zone that is connected to the now absent QM region will have its nuclear charge changed from 6 to 5 to account for a missing electron. The key input for this RUNTYP=MAKEFP job is the six orbitals in \$VEC, plus the groups

```
$guess guess=huckel insorb=6 $end
```

```
$mofrz frz=.true. ifrz(1)=1,2,3,4,5,6 $end
```

```
$stone
```

```
QMMMbuf
```

```
$end
```

which will cause the wavefunction optimization for the remaining atoms to optimize orbitals only in the NH₂ and COOH pieces. After this wavefunction is found, the run extracts the EFP information needed for the QM/MM third run(s). This means running the Stone analysis for distributed multipoles, and obtaining a polarizability tensor for each localized orbital in the EFP region.

The QM/MM run might be RUNTYP=OPTIMIZE, etc. depending on what you want to do with the quantum atoms, and its \$DATA group will contain both the 13 fully optimized atoms,

and the 5 buffer atoms, and a basis set will exist on both sets of atoms. The carbon atom in the buffer zone that borders the EFP region will have its nuclear charge set to 4 since now two bonding electrons to the EFP region are lost. \$VEC input will provide the six frozen orbitals in the buffer zone. The EFP atoms are defined in a fragment potential group.

The QM/MM run could use RHF or ROHF wavefunctions, to geometry optimize the locations of the quantum atoms (but not of course the frozen buffer zone or the EFP piece). It could remove the proton to compute the proton affinity at that terminal nitrogen, hunt for transition states, and so on. Presently the gradient for GVB and MCSCF is not quite right, so their use is discouraged.

Input to control the QM/MM preparation is \$TRUNCN and \$MOFRZ groups. There are a number of other parameters in various groups, namely QMMMBUF in \$STONE, MOIDON and POLNUM in \$LOCAL, NBUFFMO in \$EFRAG, and INSORB in \$GUESS that are relevant to this kind of computation. For RUNTYP=MAKEFP, the biggest choices are LOCAL=RUEDENBRG vs. BOYS, and POLNUM in \$LOCAL, otherwise this is pretty much a standard RUNTYP=ENERGY input file.

Source code distributions of GAMESS contain a directory named ~/gamess/tools/efp, which has various tools for EFP manipulation in it, described in file readme.1st. A full input file for the protonated lysine molecule is included, with instructions about how to proceed to the next steps. Tips on more specialized input possibilities are appended to the file readme.1st.

Simpler potentials

Since the EFP model's electrostatics is a set of distributed multipoles (monopole to octopole) and distributed polarizabilities (dipole), it is possible to generate some water potentials found in the literature by setting many EFP terms to zero. It is also necessary to provide a Lennard-Jones 6-12 repulsive potential, and then make a choice to follow the EFP1 type formula for QM/EFP repulsion. Accordingly, EFP1 type calculations can be made with the following water potentials,

FRAGNAME=SPC, SPCE, TIP5P, TIP5PE, or POL5P
The Wikipedia page
http://en.wikipedia.org/wiki/Water_model

defines the first four of these, which are not polarizable potentials. The same web site references the primary literature, so that is not repeated here. POL5P is a polarizable potential, with parameters given by

D.Si and H.Li J.Chem.Phys. 133, 144112/1-8(2010)

references

The first paper is more descriptive, while the second presents a very detailed derivation of the EFP1 method. Reference 18 is an overview article on EFP2. Reference 44 is the most recent review.

The model development papers are: 1, 2, 3, 5, 6, 9, 14, 17, 18, 22, 23, 26, 31, 32, 34, 39, 41, 42, 43, 44, 46, 50, 51, 55, 57, 58.

1. "Effective fragment method for modeling intermolecular hydrogen bonding effects on quantum mechanical calculations"
J.H.Jensen, P.N.Day, M.S.Gordon, H.Basch, D.Cohen, D.R.Garmer, M.Krauss, W.J.Stevens in "Modeling the Hydrogen Bond" (D.A. Smith, ed.) ACS Symposium Series 569, 1994, pp 139-151.
2. "An effective fragment method for modeling solvent effects in quantum mechanical calculations".
P.N.Day, J.H.Jensen, M.S.Gordon, S.P.Webb, W.J.Stevens, M.Krauss, D.Garmer, H.Basch, D.Cohen J.Chem.Phys. 105, 1968-1986(1996).
3. "The effective fragment model for solvation: internal rotation in formamide"
W.Chen, M.S.Gordon, J.Chem.Phys., 105, 11081-90(1996)
4. "Transphosphorylation catalyzed by ribonuclease A: Computational study using ab initio EFPs"
B.D.Wladkowski, M. Krauss, W.J.Stevens J.Am.Chem.Soc. 117, 10537-10545(1995)
5. "Modeling intermolecular exchange integrals between nonorthogonal orbitals"
J.H.Jensen J.Chem.Phys. 104, 7795-7796(1996)
6. "An approximate formula for the intermolecular Pauli repulsion between closed shell molecules"
J.H.Jensen, M.S.Gordon Mol.Phys. 89, 1313-1325(1996)
7. "A study of aqueous glutamic acid using the effective fragment potential model"
P.N.Day, R.Pachter J.Chem.Phys. 107, 2990-9(1997)
8. "Solvation and the excited states of formamide"
M.Krauss, S.P.Webb J.Chem.Phys. 107, 5771-5(1997)
9. "An approximate formula for the intermolecular Pauli repulsion between closed shell molecules. Application

- to the effective fragment potential method"
J.H.Jensen, M.S.Gordon
J.Chem.Phys. 108, 4772-4782(1998)
10. "Study of small water clusters using the effective fragment potential method"
G.N.Merrill, M.S.Gordon J.Phys.Chem.A 102, 2650-7(1998)
 11. "Solvation of the Menshutkin Reaction: A Rigorous test of the Effective Fragment Model"
S.P.Webb, M.S.Gordon J.Phys.Chem.A 103, 1265-73(1999)
 12. "Evaluation of the charge penetration energy between nonorthogonal molecular orbitals using the Spherical Gaussian Overlap approximation"
V.Kairys, J.H.Jensen
Chem.Phys.Lett. 315, 140-144(1999)
 13. "Solvation of Sodium Chloride: EFP study of $\text{NaCl}(\text{H}_2\text{O})_n$ "
C.P.Petersen, M.S.Gordon
J.Phys.Chem.A 103, 4162-6(1999)
 14. "QM/MM boundaries across covalent bonds: frozen LMO based approach for the Effective Fragment Potential method"
V.Kairys, J.H.Jensen J.Phys.Chem.A 104, 6656-65(2000)
 15. "A study of water clusters using the effective fragment potential and Monte Carlo simulated annealing"
P.N.Day, R.Pachter, M.S.Gordon, G.N.Merrill
J.Chem.Phys. 112, 2063-73(2000)
 16. "A combined discrete/continuum solvation model: Application to glycine" P.Bandyopadhyay, M.S.Gordon
J.Chem.Phys. 113, 1104-9(2000)
 17. "Evaluation of charge penetration between distributed multipolar expansions"
M.A.Freitag, M.S.Gordon, J.H.Jensen, W.J.Stevens
J.Chem.Phys. 112, 7300-7306(2000)
 18. "The Effective Fragment Potential Method: a QM-based MM approach to modeling environmental effects in chemistry"
M.S.Gordon, M.A.Freitag, P.Bandyopadhyay, J.H.Jensen, V.Kairys, W.J.Stevens J.Phys.Chem.A 105, 293-307(2001)
 19. "Accurate Intraprotein Electrostatics derived from first principles: EFP study of proton affinities of lysine 55 and tyrosine 20 in Turkey Ovomucoid"
R.M.Minikis, V.Kairys, J.H.Jensen
J.Phys.Chem.A 105, 3829-3837(2001)
 20. "Active site structure & mechanism of Human Glyoxalase"
U.Richter, M.Krauss J.Am.Chem.Soc. 123, 6973-6982(2001)
 21. "Solvent effect on the global and atomic DFT-based reactivity descriptors using the EFP model. Solvation of ammonia." R.Balawender, B.Safi, P.Geerlings
J.Phys.Chem.A 105, 6703-6710(2001)
 22. "Intermolecular exchange-induction and charge transfer:

- Derivation of approximate formulas using nonorthogonal localized molecular orbitals." J.H.Jensen J.Chem.Phys. 114, 8775-8783(2001)
23. "An integrated effective fragment-polarizable continuum approach to solvation: Theory & application to glycine" P.Bandyopadhyay, M.S.Gordon, B.Mennucci, J.Tomasi J.Chem.Phys. 116, 5023-5032(2002)
24. "The prediction of protein pKa's using QM/MM: the pKa of Lysine 55 in turkey ovomucoid third domain" H.Li, A.W.Hains, J.E.Everts, A.D.Robertson, J.H.Jensen J.Phys.Chem.B 106, 3486-3494(2002)
25. "Computational studies of aliphatic amine basicity" D.C.Caskey, R.Damrauer, D.McGoff J.Org.Chem. 67, 5098-5105(2002)
26. "Density Functional Theory based Effective Fragment Potential" I.Adamovic, M.A.Freitag, M.S.Gordon J.Chem.Phys. 118, 6725-6732(2003)
27. "Intraprotein electrostatics derived from first principles: Divide-and-conquer approaches for QM/MM calculations" P.A.Molina, H.Li, J.H.Jensen J.Comput.Chem. 24, 1971-1979(2003)
28. "Formation of alkali metal/alkaline earth cation water clusters, $M(H_2O)_1-6$, $M=Li^+$, K^+ , Mg^{+2} , Ca^{+2} : an effective fragment potential case study" G.N.Merrill, S.P.Webb, D.B.Bivin J.Phys.Chem.A 107, 386-396(2003)
29. "Anion-water clusters $A-(H_2O)_1-6$, $A=OH$, F , SH , Cl , and Br . An effective fragment potential test case" G.N.Merrill, S.P.Webb J.Phys.Chem.A 107, 7852-7860(2003)
30. "The application of the Effective Fragment Potential to molecular anion solvation: a study of ten oxyanion-water clusters, $A-(H_2O)_1-4$ " G.N.Merrill, S.P.Webb J.Phys.Chem.A 108, 833-839(2004)
31. "The effective fragment potential: small clusters and radial distribution functions" H.M.Netzloff, M.S.Gordon J.Chem.Phys. 121, 2711-4(2004)
32. "Fast fragments: the development of a parallel effective fragment potential method" H.M.Netzloff, M.S.Gordon J.Comput.Chem. 25, 1926-36(2004)
33. "Theoretical investigations of acetylcholine (ACh) and acetylthiocholine (ATCh) using ab initio and effective fragment potential methods" J.Song, M.S.Gordon, C.A.Deakyne, W.Zheng J.Phys.Chem.A 108, 11419-11432(2004)
34. "Dynamic polarizability, dispersion coefficient C_6 , and dispersion energy in the effective fragment potential method"

- I.Adamovic, M.S.Gordon Mol.Phys. 103, 379-387(2005)
35. "Solvent effects on the SN2 reaction: Application of the density functional theory-based effective fragment potential method"
I.Adamovic, M.S.Gordon J.Phys.Chem.A 109, 1629-36(2005)
36. "Theoretical study of the solvation of fluorine and chlorine anions by water"
D.D.Kemp, M.S.Gordon J.Phys.Chem.A 109, 7688-99(2005)
37. "Modeling styrene-styrene interactions"
I.Adamovic, H.Li, M.H.Lamm, M.S.Gordon J.Phys.Chem.A 110, 519-525(2006)
38. "Methanol-water mixtures: a microsolvation study using the Effective Fragment Potential method"
I.Adamovic, M.S.Gordon J.Phys.Chem.A 110, 10267-10273(2006)
39. "Charge transfer interaction in the effective fragment potential method" H.Li, M.S.Gordon, J.H.Jensen J.Chem.Phys. 124, 214108/1-16(2006)
40. "Incremental solvation of nonionized and zwitterionic glycine"
C.M.Aikens, M.S.Gordon J.Am.Chem.Soc. 128, 12835-12850(2006)
41. "Gradients of the polarization energy in the Effective Fragment Potential method"
H.Li, H.M.Netzloff, M.S.Gordon J.Chem.Phys. 125, 194103/1-9(2006)
42. "Electrostatic energy in the Effective Fragment Potential method: Theory and application to benzene dimer"
L.V.Slipchenko, M.S.Gordon J.Comput.Chem. 28, 276-291(2007)
43. "Polarization energy gradients in combined Quantum Mechanics, Effective Fragment Potential, and Polarizable Continuum Model Calculations"
H.Li, M.S.Gordon J.Chem.Phys. 126, 124112/1-10(2007)
44. "The Effective Fragment Potential: a general method for predicting intermolecular interactions"
M.S.Gordon, L.V.Slipchenko, H.Li, J.H.Jensen Annual Reports in Computational Chemistry, Volume 3, pp 177-193 (2007).
45. "An Interpretation of the Enhancement of the Water Dipole Moment Due to the Presence of Other Water Molecules"
D.D.Kemp, M.S.Gordon J.Phys.Chem.A 112, 4885-4894(2008)
46. "Solvent effects on optical properties of molecules: a combined time-dependent density functional/effective fragment potential approach"
S.Yoo, F.Zahariev, S.Sok, M.S.Gordon

- J.Chem.Phys. 129, 144112/1-8(2008)
47. "Modeling pi-pi interactions with the effective fragment potential method: The benzene dimer and substituents"
T.Smith, L.V.Slipchenko, M.S.Gordon
J.Phys.Chem.A 112, 5286-5294(2008)
48. "Water-benzene interactions: An effective fragment potential and correlated quantum chemistry study"
L.V.Slipchenko, M.S.Gordon
J.Phys.Chem.A 113, 2092-2102(2009)
49. "Ab initio QM/MM excited-state molecular dynamics study of Coumarin 151 in water solution"
D.Kina, P.Arora, A.Nakayama, T.Noro, M.S.Gordon, T.Taketsugu Int.J.Quantum Chem. 109, 2308-2318(2009)
50. "Damping functions in the effective fragment potential method"
L.V.Slipchenko, M.S.Gordon
Mol.Phys. 197, 999-1016 (2009)
51. "A combined effective fragment potential-fragment molecular orbital method. 1. the energy expression"
T.Nagata, D.G.Fedorov, K.Kitaura, M.S.Gordon
J.Chem.Phys. 131, 024101/1-12(2009)
52. "Alanine: then there was water"
J.M.Mullin, M.S.Gordon
J.Phys.Chem.B 113, 8657-8669(2009)
53. "Water and Alanine: from puddles(32) to ponds(49)"
J.M.Mullin, M.S.Gordon
J.Phys.Chem.B 113, 14413-14420(2009)
54. "Structure of large nitrate-water clusters at ambient temperatures: simulations with effective fragment potentials and force fields with implications for atmospheric chemistry"
Y.Miller, J.L.Thoman, D.D.Kemp, B.J.Finlayson-Pitts, M.S.Gordon, D.J.Tobias, R.B.Gerber
J.Phys.Chem.A 113, 12805-12814(2009)
55. "Quantum mechanical/molecular mechanical/continuum style solvation model: linear response theory, variational treatment, and nuclear gradients"
H.Li J.Chem.Phys. 131, 184103/1-8(2009)
56. "Aqueous solvation of bihalide anions"
D.D.Kemp, M.S.Gordon
J.Phys.Chem.A 114, 1298-1303(2010)
57. "Exchange repulsion between effective fragment potentials and ab initio molecules"
D.D.Kemp, J.M.Rintelman, M.S.Gordon, J.H.Jensen
Theoret.Chem.Acc. 125, 481-491(2010).
58. Modeling Solvent Effects on Electronic Excited States
A.DeFusco, N.Minezawa, L.V.Slipchenko, F.Zahariev, M.S.Gordon J.Phys.Chem.Lett. 2, 2184-2192(2011)

The Fragment Molecular Orbital method

coded by D.G. Fedorov, M.Chiba, T. Nagata and K. Kitaura at
Research Institute for Computational Sciences (RICS)
National Institute of Advanced Industrial Science
and Technology (AIST)
AIST Tsukuba Central 2, Umezono 1-1-1,
Tsukuba, 305-8568, Japan.

with code contributions by:

N. Asada (Kyoto U.), C. H. Choi (Kyungpook U.),
C. Steinmann (U. Copenhagen).

The method was proposed by Professor Kitaura and coworkers in 1999, based on the Energy Decomposition Analysis (EDA, sometimes called the Morokuma-Kitaura energy decomposition). The FMO method is completely independent of and bears no relation to:

1. Frontier molecular orbitals (FMO),
2. Fragment molecular orbitals (FMO).

The latter name is often used for the process of construction of full molecular orbitals by combining MO diagrams for parts of a molecule, ala Roald Hoffmann. The effective fragment molecular orbital method (EFMO) is closely related to but also bears significant difference to FMO, and discussed below.

The FMO program was interfaced with GAMESS and follows general GAMESS guidelines for code distribution and usage. The users of the FMO program are requested to cite the FMO3-RHF paper as the basic FMO reference,
D.G. Fedorov, K. Kitaura,
J. Chem. Phys. 120, 6832-6840(2004)
and other papers as appropriate (see below).

The basic idea of the method is to acknowledge the fact the exchange and self-consistency are local in most molecules (and clusters and molecular crystals), which permits treating remote parts with Coulomb operators only, ignoring the exchange. This idea further evolves into doing molecular calculations, piecewise, with Coulomb fields due to the remaining parts. In practice one divides the molecule into fragments and performs n-mer calculations of these in the Coulomb field of other fragments (n=1,2,3). There are no empirical parameters, and the only departure from ab initio rigor is the subjective fragmentation. It has been observed that if performed physically reasonably, the fragmentation scheme alters the results very little.

What changes the accuracy the most is the fragment size, which also determines the computational efficiency of the method.

The first question is how to get started. The easiest way to prepare an FMO input file for GAMESS is to use the free GUI software Facio, developed by M. Suenaga at Kyushu University. It can do molecular modeling, automatic fragmentation of peptides, nucleotides and saccharides and create GAMESS/FMO input files:

<http://www1.bbiq.jp/zzzfelis/Facio.html>

A web based interface to FMO is maintained by Y. Alexeev (ANational Lab):

<http://www.fmo-portal.info>

Alternatively, if you prefer a command line interface, and your molecule is a protein found in the PDB

<http://www.rcsb.org/pdb>

you can simply use the fragmentation program "fmoutil" that is provided with GAMESS in tools/fmo, or the FMO home page

<http://staff.aist.go.jp/d.g.fedorov/fmo/main.html>

If you have a cluster of identical molecules, you can perform fragmentation with just one keyword (\$FMO NACUT=).

Computationally, it is always better to partition in a geometrical way (close parts together), so that the distance-based approximations are more efficient. The accuracy depends mainly upon the locality of the density distribution, and the appropriateness of partitioning it into fragments. There is no simple connexion between the geometrical proximity of fragmentation and accuracy.

Supposing you know how to fragment, you should choose a basis set and fragment size. We recommend 2 amino acid residues or 2-4 water molecules per fragment for final energetics (or, even better, three-body with 1 molecule or residue per fragment). For geometry optimizations one may be able to use 1 res/mol per fragment, especially if gradient convergence to about 0.001 is desired. Note that although it was claimed that FMO gradient is analytic (Chem. Phys. Lett., 336 (2001), 163.) it is not so. Neither theory nor program for fully analytic gradient has been developed, to the best of our knowledge up to this day (December 21, 2006). The gradient implementation is nearly analytic, meaning three small terms are missing, one which can now be included using MODGRD=8+2. The magnitude of these small terms depends upon the fragment size (larger fragments have smaller errors). It has been our experience that in proteins with 1 residue per fragment one gets 1e-

3...1e-4 error in the gradient, and with 2 residues per fragment it is about 1e-4...1e-5. If you experience energy rising during geometry optimizations, you can consider two countermeasures:

1. increase approximation thresholds, e.g. RESPPC from 2.0->2.5, RESDIM from 2.0 -> 2.5.
2. increase fragment size (e.g. by merging very small fragments with their neighbors).

Finally a word of caution: optimizing systems with charged fragments in the absence of solvent is frequently not a good idea: oppositely charged fragments will most likely seek each other, unless there is some conformational barrier.

One thing you should clearly understand about gradients is that if you compare FMO gradients with full molecule ab initio gradients, there are two sources of errors:

- a) error of the analytic FMO gradient compared to ab initio.
- b) error of a "nearly analytic" FMO gradient compared to the analytic FMO gradient.

Since the analytic FMO gradient is not available, these two are not separable at the moment. If FMO gradients were fully analytic, geometry optimization and dynamics would have run perfectly, irrespective of error a).

For basis sets you should use general guidelines and your experience developed for ab initio methods. There is a file provided (HMOs.txt) that contains hybrid molecular orbitals (HMO) used to divide the MO space along fragmentation points at covalent bonds. If your basis set is not there you need to construct your own set of HMOs. See the example file makeLMO.inp for this purpose.

Next you choose a wave function type. At present one can use RHF, DFT, MP2, CC, and MCSCF (all except MCSCF support the 3-body expansion). Geometry optimization can be performed with all of these methods, except CC.

Note that presence of \$FMO turns FMO on.

Surfaces and solids

Until 2008, for treating covalently connected fragments, FMO had fully relaxed electron density of the detached bonds. This method is now known as FMO/HOP (HOP=hybrid orbital projection operator). It allows for a full polarization of the system and is thus well suited to very

polar systems, such as proteins with charged residues. In 2008, an alternative fragmentation was suggested, based on adaptive frozen orbitals (AFO), FMO/AFO. In it, the electron density for each detached bond is first computed in the automatically generated small model system (with the bond intact), and in the FMO fragment calculations this electron density is frozen. It was found that FMO/AFO works quite well for surfaces and solids, where there is a dense network of bonds to be detached in order to define fragments (and the detached bonds interact quite strongly). In addition, by restricting the polarization, FMO/AFO was found to give a more balanced properties for large basis sets (triple-zeta with polarization or larger), or in comparing different isomers. However, for proteins with charged residues the original FMO/HOP scheme has a better accuracy (except large basis sets). At this point, FMO/AFO was applied to zeolites only, and some more experience is needed to give more practical advice to applications. FMO/AFO is turned on by a nonzero `rafo(1)` parameter (`rafo` array provides the thresholds to build model systems).

FMO variants

In 2007, Dahlke et al. introduced the Electrostatically Embedded Many-Body Expansion method (see E. E. Dahlke and D. G. Truhlar, *J. Chem. Theory Comput.* 4, 1-6 (2008) for more recent work). This method is essentially FMO with the RESPPC approximation (point charges for the electrostatic field) applied to all fragments, with the further provision that these charges may be defined at will (whereas RESPPC uses Mulliken charges), and they are kept frozen (not optimized, as in FMO). Next, Kamiya et al. suggested a fast electron correlation method (M. Kamiya, S. Hirata, M. Valiev, *J. Chem. Phys.* 128, 074103 (2008)), where again FMO with the RESPPC approximation to all fragments is applied with the further provision that the charges are derived from the electrostatic potential (so called ESP charges), and BSSE correction is added. The Dahlke's method was generalized in GAMESS with the introduction of an arbitrary hybrid approach, in which some fragments may have fixed and some variationally optimized charges. This implementation was employed in FMO-TDDFT calculations of solid state quinacridone (see Ref. 16 below) by using DFT/PBC frozen charges. The present energy only implementation is mostly intended for such cases as that (i.e., TDDFT), and some more work is needed to finish it for general calculations. To turn this on, set `RESPPC=-1` and define `NOPFRG` for frozen

charge fragments to 64, set frozen charges in ATCHRG. Another FMO-like method is EFMO, see its own subsection below. EFMO itself is related to several methods (PMISP: P. Soederhjelm, U. Ryde, J. Phys. Chem. A 2009, 113, 617–627; another is G. J. O. Beran, J. Chem. Phys. 2009, 130, 164115).

Effective fragment molecular orbital method (EFMO)

EFMO has been formulated by combining the physical models in EFP and FMO, namely, in EFMO, fragments are computed without the ESP (of FMO), and the polarization is estimated using EFP models of fragment polarizabilities, which are computed on the fly, so this can be thought of as automatically generated potentials in EFP. Consequently, close dimers are computed quantum-mechanically (without ESP) and far dimers are computed using the electrostatic multipole models of EFP. At present, only vacuum closed-shell RHF and DFT are supported, for energy and gradient; and only molecular clusters can be computed (no systems with detached bonds). From the user point of view, EFMO functionality is very intensively borrowed from FMO, and the calculation setup is almost identical. Most additional physical models such as PCM are not supported in EFMO. EFMO should not be confused with FMO/EFP. The latter uses FMO for some fragments and EFP for others. EFMO uses the same model (EFMO), which is neither FMO nor EFP. For approximations, EFMO at present has only RESDIM.

EFMO references are:

1. Effective Fragment Molecular Orbital Method: A Merger of the Effective Fragment Potential and Fragment Molecular Orbital Methods.
C. Steinmann, D. G. Fedorov, J. H. Jensen
J. Phys. Chem. A 114, 8705–8712 (2010).
2. The Effective Fragment Molecular Orbital Method for Fragments Connected by Covalent Bonds.
C. Steinmann, D. G. Fedorov, J. H. Jensen
PLoS One, 7, e41117(2012).
3. Mapping enzymatic catalysis using the effective fragment molecular orbital method: towards all ab initio biochemistry.
C. Steinmann, D. G. Fedorov, J. H. Jensen
PLoS One 8, e60602 (2013).

Guidelines for approximations with FMO3

Three sets are suggested, for various accuracies:

low: resppc=2.5 resdim=2.5 ritrim(1)=0.001,-1,1.25

medium: resppc=2.5 resdim=3.25 ritrim(1)=1.25,-1,2.0

high: resppc=2.5 resdim=4.0 ritrim(1)=2,2,2

For correlated runs, add one more value to ritrim, equal to the third element (i.e., 1.25 or 2.0). Note that gradient runs do not support nonzero RESDIM and thus use RESDIM=0 if gradient is to be computed. The "low" level of accuracy for FMO3 has an error versus full ab initio similar to FMO2, except for extended basis sets (6-311G** etc) where it is substantially better than FMO2. Thus the low level is only recommended for those large basis sets, and if a better level cannot be afforded. The medium level is recommended for production FMO3 runs; the high level is mostly for accuracy evaluation in FMO development. The cost is roughly: 3(low), 6(medium), 12(high). This means that FMO3 with the medium level takes roughly six times longer than FMO2.

Some of the default tolerances were changed as of January 2009, when FMO 3.2 was included in GAMESS. In general, stricter parameters are now enforced when using FMO3, which of course is intended to produce more accurate results. If you wish to reproduce earlier results with the new code, use the input to revert to the earlier values:

former -> FMO2 or FMO3 (as of 1/2009)

RESPPC: 2.0 2.0 2.50

RESDIM: 2.0 2.0 3.25

RCORSD: 2.0 2.0 3.25

RITRIM: 2.0,2.0,2.0,2.0 -> 1.25,-1.0,2.0,2.0 (FMO3 only)

MODESP: 1 0 1

MODGRD: 0 10 0

and two other settings which are not strictly speaking FMO keywords may change FMO results:

MTHALL: 2 -> 4 (FMO/PCM only, see \$TESCAV)

DFT grid: spherical -> Lebedev (FMO-DFT only, see \$DFT)

Note that FMO2 energies printed during a FMO3 run will differ from those in a FMO2 run, due to the different tolerances used.

How to perform FMO-MCSCF calculations

Assuming that you are reasonably acquainted with ab initio MCSCF, only FMO-specific points are highlighted. The active space (the number of orbitals/electrons) is specified for the MCSCF fragment. The number of core/virtual orbitals for MCSCF dimers will be automatically computed. The most important issue is the initial orbitals for the MCSCF

monomer. Just as for ab initio MCSCF, you should exercise chemical knowledge and provide appropriate orbitals. There are two basic ways to input MCSCF initial orbitals:

- A) through the FMO monomer density binary file
- B) by providing a text \$VEC group.

The former way is briefly described in INPUT.DOC (see orbital conversion). The latter way is really identical to ab initio MCSCF, except the orbitals should be prepared for the fragment (so in many cases you would have to get them from an FMO calculation). Once you have the orbitals, put them into \$VEC1, and use the IJVEC option in \$FMOPRP (e.g., if your MCSCF fragment is number 5, you would use \$VEC1 and ijvec(1)=5,0). For two-layer MCSCF the following conditions apply. Usually one cannot simply use F40 restart, because its contents will be overwritten with RHF orbitals and this will mess up your carefully chosen MCSCF orbitals. Therefore, two ways exist. One is to modify A) above by reordering the orbitals with something like

```
$guess guess=skip norder=1 iorder(28)=29,30,31,32,28 $end
```

Then the lower RHF layer will converge RHF orbitals that you reorder with iorder in the same run (add 512 to nguess in \$FMO). This requires you know how to reorder before running the job so it is not always convenient. Probably

the best way to run two-layer MCSCF is verbatim B) above, so just provide MCSCF monomer orbitals in \$VEC1. Finally, it may happen that some MCSCF dimer will not converge.

Beside the usual MCSCF tricks to gain convergence as the last resort you may be able to prepare good initial dimer orbitals, put them into \$VEC2 (\$VEC3 etc) and read them with ijvec. SOSCF is the preferred converger in FMO, and the other one (FULLNR) has not been modified to eradicate the artefacts of convergence (due to detached bonds). In the bad cases you can try running one or two monomer SCF iterations with FULLNR, stop the job and use its orbitals in F40 to do a restart with SOSCF. We also found useful to set CASDII=0.005 and nofo=10 in some cases running FOCAS longer to get better orbitals for SOSCF.

How to perform multilayer runs

For some fragments you may like to specify a different level of electron correlation and/or basis set. In a typical case, you would use high level for the reaction center and a lower level for the remaining part of the system. The set up for multilayer runs is very similar to the unilayer case. You only have to specify to what layer each fragment belongs and for each layer define DFTTYP, MPLEVEL, SCFTYP as well as a basis set. If detached bonds

are present, appropriate HMOs should be defined. See the paragraph above for multilayer MCSCF. Currently geometry optimizations of multilayer runs require adding 128 to NGUESS, if basis sets in layers differ from each other.

How to mix basis sets in FMO

You can mix basis sets in both uni and multilayer cases. The difference between a 2-layer run with one basis set per layer and a 1-layer run with 2-basis sets is significant: in the former case the lower level densities are converged with all fragments computed at the lower level. In the latter case, the fragments are converged simultaneously, each with its own basis set. In addition, dimer corrections between layers will be computed differently: with the lower basis set in the former case and with mixed basis set in the latter. The latter approach may result in unphysical polarization, so mixing basis sets is mainly intended to add diffuse functions to anionic (e.g., carboxyl) groups, not as a substitute for two-layer runs.

How to perform FMO/PCM calculations

Solvent effects can be taken into account with PCM. PCM in FMO is very similar to regular PCM. There is one basic difference: in FMO/PCM the total electron density that determines the electrostatic interaction is computed using the FMO density expansion up to n-body terms. The cavity is constructed surrounding the whole molecule, and the whole cavity is used in each individual m-mer calculation. There are several levels of accuracy (determined by the "n" above), and the recommended level is FMO/PCM[1(2)], specified by:

```
$pcm ief=-10 icomp=2 icav=1 idisp=1 ifmo=2 $end
$fmoprp npcmit=2 $end
$tescav ntsall=240 $end
$pcmcav radii=suahf $end
```

Many PCM options can be used as in the regular PCM. The following restrictions apply:

IEF may be only -3 or -10, IDP must be 0.

No FMO/PCM gradients are available. Multilayer FMO runs are supported. Restarts are only limited to IREST=2, and in this case PCM charges (the ASCs) are not recycled. However, the initial guess for the charges is fairly

reasonable, so IREST=2 may be useful although reading the ASCs may be implemented in future.

Note for advanced users. IFMO < NBODY runs are permitted. They are denoted by FMO_m/PCM[n], where m=NBODY and n=IFMO. In FMO_m/PCM[n], the ASCs are computed with n-body level. The difference between FMO2/PCM[1] and FMO2/PCM[1(2)] is that in the former the ASCs are computed at the 1-body level, whereas for the former at the 2-body level, but without self-consistency (which would be FMO2/PCM[2]). Probably, FMO3/PCM[2] should be regarded as the most accurate and still affordable (with a few thousand nodes) method. However, FMO3/PCM[1(2)] (specified with NBODY=3, IFMO=2 and NPCMIT=2) is much cheaper and slightly less accurate than FMO3/PCM[2]. FMO3/PCM[3] is the most accurate and expensive level of all.

How to perform FMO/EFP calculations

Solvent effects can also be taken into account with the Effective Fragment Potential model. The presence of both \$FMO and \$EFRAG groups selects FMO/EFP calculations. See the \$EFRAG group and the \$FMO group for details.

In the FMO/EFP method, the Quantum Mechanical part of the calculation in the usual EFP method is replaced by the FMO method, which may save time for large molecules such as proteins.

In the present version, only FMO_n/EFP1 (water solvent only) is available for RHF, DFT and MP2. One can use the MC global optimization technique for FMO/EFP by RUNTYP=GLOBOP. Of course, the group DDI (GDDI) parallelization technique for the FMO method can be used.

Geometry optimization or saddle point search for FMO

The standard optimizers in GAMESS are now well parallelized, and thus recommended to be used with FMO up to the limit hardwired in GAMESS (2000 atoms). In practice, if more than about 1000 atoms are present, numeric Hessian updates often result in the improper curvature and optimization stops. One can either do a restart, or use RUNTYP=OPTFMO (which does not diagonalize the Hessian).

RUNTYP=OPTIMIZE applies to Cartesian coordinates or DLC. RUNTYP=OPTFMO works only with Cartesian coordinates. If

your system has more than 2000 atoms you can consider RUNTYP=OPTFMO, which can now use Hessian updates and provides reasonable way to optimize although it is not as good as the standard means in RUNTYP=OPTIMIZE.

A transition state search for FMO can be performed with RUNTYP=SADPOINT using either Cartesian coordinates or DLC.

FMO hessian calculations

Analytic FMO Hessian with RUNTYP=HESSIAN may be computed for RHF in gas phase (no PCM, EFP etc), without approximations, e.g., resppc and resdim should be 0.

Molecular dynamics with FMO

MD can be run for any FMO method, which has the gradient implemented. However, in many cases the approximations in the gradient for a particular method may lead to large discrepancies in MD. The following methods have a fully analytic gradient (which has to be turned on with \$fmopr modgrd=42): FMO-RHF, FMO-MP2, FMO-RHF/EFP; the following condition should be satisfied: no ESP approximations, RESPPC=0. FMO/FD is not implemented for MD.

Pair interaction energy decomposition analysis (PIEDA)

PIEDA can be performed for the PL0 and PL states. The PL0 state is the electronic state in which fragments are polarised by the environment in its free (noninteracting) state. The simplest example is that in a water cluster, each molecule is computed in the electrostatic field exerted by the electron densities of free water molecules. The PL state is the FMO converged monomer state, that is, the state in which fragments are polarised by the self-consistently converged environment. Namely, following the FMO prescription, fragments are recomputed in the external field, until the latter converges. Using the PL0 state requires a series of separate runs; and it also relies on a "free state" which can be defined in many ways for molecules with detached covalent bonds.

What should be done to do the PL0 state analysis?

1. run FMO0.

This computes the free state for each fragment, and those electron densities are stored on file 30 (to be renamed file 40 and reused in step 3).

2. compute BDA energies (if detached bonds are present), using sample files in tools/fmo/pieda. This corrects for artifacts of bond detaching, and involves running a model system like H3C-CH3, to amend for C-C bond detaching.

3. Using results of (1) and (2), one can do the PL0 analysis. In addition to pasting the data from the two punch files in steps 1,2 and the density file in step 1 should be provided.

What should be done to do the PL state analysis? The PL state itself does not need either the free state or PL0 results. However, if the PL0 results are available, coupling terms can be computed, and in this case IPIEDA is set to 2; otherwise to 1.

So the easiest and frequently sufficient way to run PIEDA is to set IPIEDA=1 and do not provide any data from preceding calculations. The result of a PIEDA calculation is a set of pair interaction energies (interfragment interaction energies), decomposed into electrostatic, exchange-repulsion, charge transfer and dispersion contributions.

Finally, PIEDA (especially for the PL state) can be thought of as FMO-EDA, EDA being the Kitaura-Morokuma decomposition (RUNTYP=MOROKUMA). In fact, PIEDA (for the PL state) in the case of just two fragments of standalone molecules is entirely equivalent to EDA, which can be easily verified, by running the full PIEDA analysis (ipieda=2). Note that PIEDA can be run as direct SCF, whereas EDA cannot be, and for large fragments PIEDA code can be used to perform EDA. Also, EDA in GAMESS has no explicit dispersion.

In 2012, PIEDA/PCM was developed describing the solvent screening. RO-PIEDA based on RO-(HF, MP2 or CC) may be used for radicals. Grimme's dispersion models may be used in PIEDA.

Excited states

At present, one can use CI, MCSCF, or TDDFT to compute excited states in FMO. MCSCF is discussed separately above, so here only TDDFT and CI are explained. They are enabled by setting the IEXCIT option (EXCIT(1) defines the excited state's fragment ID).

Two levels are implemented for TDDFT (FM01-TDDFT and FM02-TDDFT). In the former, only monomer TDDFT calculations are performed, whereas the latter adds pair corrections from TDDFT dimers. PCM may be used for solvent effects with TDDFT (PCM[1] is usually sufficient).

CI can only be done for CIS at the monomer level (nbody=1), FM01-CIS. The set-up for CI is similar to that for TDDFT.

Selective FMO

Sometimes, one is interested only in some pair interactions, for example, between ligand and protein, or the opposite, only pair interactions within ligand. This saves a lot of CPU time by omitting all other pair calculations, but does not give the total properties. To use this feature, define MOLFRG and MODMOL. RUNTYP=ENERGY only is implemented.

Frozen domain

To accelerate geometry optimisations, one can specify that the electronic state of the first layer in a 2-layer FMO can be computed at the initial geometry and consequently be frozen. One can define the polarizable buffer (equal to layer 2) and frozen domain (layer 1). Fragments in the polarizable buffer which contain the atoms active in geometry optimisation form the active domain. The fragments in the active domain should have a nonzero separation from the frozen domain. In FMO/FD all dimers in the polarizable buffer are computed; in FMO/FDD only those dimers which have at least one monomer in the active domain are computed. FMO/FD and FNI/FDD are only implemented for RUNTYP=OPTIMIZE. MODFD and IACTAT in \$FMO specify FMO/FD(D) atop of the usual multilayer FMO setup with some atoms frozen in geometry optimization by the standard means (i.e., IACTAT in \$STATPT). Note that in FMO/FD(D) the Hessian as used in RUNTYP=OPTIMIZE is formed only for the atoms in the second layer, so this upper layer should not have more than the GAMESS limit (currently, 2000 atoms).

IMOMM with FMO

IMOMM (namely, SIMOMM) calculations can be performed with

the "MO" in IMOMM treated using FMO, i.e., this is like QM/MM but without electronic embedding of QM by MM. This calculation uses Tinker, a plug-in source code, available from the GAMESS web site. You should compile and link in the Tinker plug-in by changing a single line in comp/compall/lked,

```
set TINKER=false  
into
```

```
set TINKER=true
```

In addition, you should change MAXATM=10 (the maximum number of atoms in the whole system, as used by Tinker) in several GAMESS source files into MAXATM=12000 (this number is used inside Tinker). If you need a larger number, change it within Tinker as well. After changing this, recompile and link GAMESS.

The input file style is in general like that of SIMOMM (q.v.). Different from regular FMO, the atomic coordinates are given in \$TINXYZ, not in \$FMOXYZ. The fragmentation in FMO applies to QM atoms only, selected by IQMATM, and numbered consequently in FMO, so that INDAT in \$FMO applies to the atoms renumbered from 1 (defined in IQMATM). Other than \$FMOXYZ being superceded by \$TINXYZ, the rest of FMO options is like in normal FMO. IMOMM based on FMO is usually referred to as FMO/MM for short, although "FMO-based SIMOMM" is probably easier to understand. The somewhat tautological FMO-IMOMM has also been used by some. Covalent boundaries between FMO and MM are supported (via link atoms). FMO/MM can be used to run geometry optimizations, which is really what it is designed for.

Analyzing and visualizing the results

Annotated outputs provided in tools/fmo have matching mathematical formulae added onto the outputs, for easier reading.

Facio (<http://www1.bbiq.jp/zzzfelis/Facio.html>) can plot various FMO properties such as interaction energies, using interactive GUI viewers.

To plot orbitals for an n-mer, set NPUNCH=2 in \$SCF and PLTORB=.T. There are several ways to produce cube files with electron densities. They are described in detail in tools/fmo/fmocube/README. To plot pair interaction maps, use tools/fmo/fmogradres to generate CSV files from GAMESS output, which can be easily read into Gnuplot or Excel.

FMO portal offers tools for visualising FMO results:

<http://www.fmo-portal.info/>

Parallelization of FMO runs with GDDI

The FMO method has been developed within a 2-level hierarchical parallelization scheme, group DDI (GDDI), allowing massively parallel calculations. Different groups of processors handle the various monomer, dimer, and maybe trimer computations. The processor groups should be sized so that GAMESS' innate scaling is fairly good, and the fragments should be mapped onto the processor groups in an intelligent fashion.

This is a very important and seemingly difficult issue. It is very common to be able to speed up parallel runs at least several times just by using GDDI better. First of all, do not use plain DDI and always employ GDDI when running FMO calculations. Next, learn that you can and should divide nodes into groups to achieve better performance. The very basic rule of thumb is to try to have several times fewer groups than jobs. Since the number of monomers and dimers is different, group division should reflect that fact. Ergo, find a small parallel computer with 8-32 nodes and experiment changing just two numbers: `ngrfmo(1)=N1,N2` and see how performance changes for your particular system.

Limitations of the FMO method in GAMESS

1. dimensions: in general none, except that geometry optimizations use the standard GAMESS engine which means you are limited to 2000 atoms. This can be changed by changing the source and recompiling GAMESS (see elsewhere).
2. Almost none of the "SCF extensions" in GAMESS can be used with FMO. This means, in particular, no ECPs, no CHARMM, no SIMOMM, etc. Anything that is not a plain basis set, with atoms given only by those legally entered in `$FMOXYZ`, will not work. The only "SCF extensions" supported are the PCM and EFP solvent models, and MCP type pseudopotentials. Not every illegal combination is trapped, caveat emptor!
3. RUNTYP is limited to ENERGY, GRADIENT, OPTIMIZE, OPTFMO, FMO0, MD, GLOBOP, SADPOINT, FMOHESS and RAMAN only:
Do not even try other ones!
4. For the three-body FMO expansion (`$FMO NBODY=3`). RESDIM may be used with RUNTYP=ENERGY. Three-body FMO-MCSCF or FMO-TDDFT is not implemented.

5. No semiempirical methods may be used, but DFTB was interfaced with FMO..

What will work the same way as ab initio:

The various SCF convergers, all DFT functionals, in-core integrals, direct SCF.

Restarts with the FMO method

RUNTYPE=ENERGY can be restarted from anywhere before trimers. To restart monomer SCF, copy file F40 with monomer densities to the grandmaster node. To restart dimers, provide file F40 and monomer energies (\$FMOENM). Optionally, some dimer energies can be supplied (\$FMOEND) to skip computation of corresponding dimers.

RUNTYPE=GRADIENT can be easily restarted from monomer SCF (which really means it is a restart of RUNTYPE=ENERGY, since gradient is computed at the end of this step). Provide file F40. There is another restart option (1024 in \$FMOPRP irest=), supporting full gradient restart, requiring, however, that you set this option in the original run (whose results you use to restart). To use this option, you would also need to keep (or save and restore) F38 files on each node (they are different).

RUNTYPE=OPTIMIZE can be restarted from anywhere within the first RUNTYPE=GRADIENT run (q.v.). In addition, by replacing FMOXYZ group, one can restart at a different geometry.

RUNTYPE=OPTFMO can be restarted by providing a new set of coordinates in \$FMOXYZ and, optionally, by transferring \$OPTRST from the punch into the input file.

Note on accuracy

The FMO method is aimed at computation of large molecules. This means that the total energy is large, for example, a 6646 atom molecule has a total energy of -165,676 Hartrees. If one uses the standard accuracy of roughly $1e-9$ (that should be taken relatively), one gets an error as much as 0.001 hartree, in a single calculation. FMO involves many ab initio single point calculations of fragments and their n-mers, thus it can be expected that numeric accuracy is 1-2 orders lower than that given by $1e-9$. Therefore, it is

compulsory that accuracy should be raised, which is done by default.

The following default parameters are reset by FMO:

ICUT/\$CONTRL (9->12), ITOL/\$CONTRL(20->24),
CONV/\$SCF(1e-5 -> 1e-7),
CUTOFF/\$MP2 (1e-9->1e-12), CUTTRF/\$TRANS(1e-9->1e-10).
CVGTOL/\$DET,\$GUGDIA (1e-5 -> 1e-6)

This to some extent slows down the calculation (perhaps on the order of 10-15%). It is suggested that you maintain this accuracy for all final energetics. However, you may be able to drop the accuracy a bit for the initial part of geometry optimization if you are willing to do manual work of adjusting accuracy in the input. It is recommended to keep high accuracy at the flat surfaces (the final part of optimizations) though. For DFT the numeric grid's accuracy may be increased in accordance with the molecule size, e.g. extending the default grid of 96*12*24 to 96*20*40.

However, some tests indicate that energy differences are quite insensitive to this increase.

FMO References

I. Basic FMO papers

A book chapter contains an introduction to FMO basics:

Theoretical development of the fragment molecular orbital (FMO) method, D. G. Fedorov, K. Kitaura, in "Modern methods for theoretical physical chemistry of biopolymers", E. B. Starikov, J. P. Lewis, S. Tanaka, Eds., pp 3-38, Elsevier, Amsterdam, 2006.

There is now a full FMO book (11 chapters), which contains an introduction to FMO aimed at general application chemists, and a wealth of practical advice on doing FMO calculations:

The Fragment Molecular Orbital Method: Practical Applications to Large Molecular System,
D. G. Fedorov, K. Kitaura, Eds.,
CRC Press, Boca Raton, FL, 2009.

FMO reviews:

D. G. Fedorov, K. Kitaura (Feature Article)
J. Phys. Chem. A 111, 6904-6914 (2007).
D. G. Fedorov, T. Nagata, K. Kitaura (Perspective)
Phys. Chem. Chem. Phys., 14, 7562-7577 (2012)

A review of FMO in the context of other fragment-based methods is

M. S. Gordon, D. G. Fedorov, S. R. Pruitt,
L. V. Slipchenko Chem. Rev. 112, 632-672 (2012).

A very concise and detailed mathematical formulation of FMO including various extensions and property calculations is published as

Mathematical formulation of the fragment molecular orbital method.

T. Nagata, D. G. Fedorov, K. Kitaura.

In "Linear-Scaling Techniques in Computational Chemistry and Physics". R. Zalesny, M. G. Papadopoulos, P. G. Mezey, J. Leszczynski, Eds., pp. 17-64, Springer, New York, 2011.

1. Fragment molecular orbital method: an approximate computational method for large molecules"

K. Kitaura, E. Ikeo, T. Asada, T. Nakano, M. Uebayasi
Chem. Phys. Lett., 313, 701(1999).

2. Fragment molecular orbital method: application to polypeptides

T. Nakano, T. Kaminuma, T. Sato, Y. Akiyama,
M. Uebayasi, K. Kitaura Chem.Phys.Lett. 318, 614(2000).

3. Fragment molecular orbital method: analytical energy gradients

K. Kitaura, S.-I. Sugiki, T. Nakano, Y. Komeiji,
M. Uebayasi, Chem. Phys. Lett., 336, 163(2001).

4. Fragment molecular orbital method: use of approximate electrostatic potential

T. Nakano, T. Kaminuma, T. Sato, K. Fukuzawa,
Y. Akiyama, M. Uebayasi, K. Kitaura
Chem. Phys. Lett., 351, 475(2002).

5. The extension of the fragment molecular orbital method with the many-particle Green's function,

K. Yasuda, D. Yamaki, J. Chem. Phys. 125, 154101(2006).

6. The role of the exchange in the embedding electrostatic potential for the fragment molecular orbital method.

D. G. Fedorov, K. Kitaura
J. Chem. Phys. 131, 171106(2009).

7. Analytic second derivatives of the energy in the fragment molecular orbital method.

H. Nakata, T. Nagata, D. G. Fedorov, S. Yokojima, K.
Kitaura, S. Nakamura, J. Chem. Phys. 138 (2013) 164103.

II. FMO in GAMESS

1. A new hierarchical parallelization scheme: generalized distributed data interface (GDDI), and an application to the fragment molecular orbital method (FMO).
D. G. Fedorov, R. M. Olson, K. Kitaura, M. S. Gordon, S. Koseki J. Comput. Chem. 25, 872-880(2004).
2. The importance of three-body terms in the fragment molecular orbital method.
D. G. Fedorov and K. Kitaura
J. Chem. Phys. 120, 6832-6840(2004).
3. On the accuracy of the 3-body fragment molecular orbital method (FMO) applied to density functional theory
D. G. Fedorov and K. Kitaura
Chem. Phys. Lett. 389, 129-134(2004).
4. Second order Moeller-Plesset perturbation theory based upon the fragment molecular orbital method.
D. G. Fedorov and K. Kitaura
J. Chem. Phys. 121, 2483-2490(2004).
5. Multiconfiguration self-consistent-field theory based upon the fragment molecular orbital method.
D. G. Fedorov and K. Kitaura
J. Chem. Phys. 122, 054108/1-10(2005).
6. Multilayer Formulation of the Fragment Molecular Orbital Method (FMO).
D. G. Fedorov, T. Ishida, K. Kitaura
J. Phys. Chem. A. 109, 2638-2646(2005).
7. Coupled-cluster theory based upon the Fragment Molecular Orbital method.
D. G. Fedorov, K. Kitaura
J. Chem. Phys. 123, 134103/1-11 (2005)
8. The polarizable continuum model (PCM) interfaced with the fragment molecular orbital method (FMO).
D. G. Fedorov, K. Kitaura, H. Li, J. H. Jensen, M. S. Gordon, J. Comput. Chem., 27, 976-985(2006)
9. The three-body fragment molecular orbital method for accurate calculations of large systems,
D. G. Fedorov, K. Kitaura
Chem. Phys. Lett. 433, 182-187(2006).
10. Pair interaction energy decomposition analysis,
D. G. Fedorov, K. Kitaura
J. Comp. Chem. 28, 222-237(2007).
11. On the accuracy of the three-body fragment molecular orbital method (FMO) applied to Moeller-Plesset perturbation theory,
D. G. Fedorov, K. Ishimura, T. Ishida, K. Kitaura, P. Pulay, S. Nagase
J. Comput. Chem., 28, 1476-1484 (2007).
12. The Fragment Molecular Orbital method for geometry optimizations of polypeptides and proteins,
D.G.Fedorov, T. Ishida, M. Uebayasi, K. Kitaura

- J.Phys.Chem.A, 111, 2722-2732(2007).
13. Time-dependent density functional theory with the multilayer fragment molecular orbital method
M. Chiba, D. G. Fedorov, K. Kitaura
Chem. Phys. Lett. 444, 346-350 (2007).
14. Time-dependent density functional theory based upon the fragment molecular orbital method
M. Chiba, D. G. Fedorov, K. Kitaura
J. Chem. Phys. 127, 104108(2007).
15. Polarizable continuum model with the fragment molecular orbital-based time-dependent density functional theory.
M. Chiba, D. G. Fedorov, K. Kitaura
J. Comput. Chem. 29, 2667-2676 (2008).
16. Theoretical Analysis of the Intermolecular Interaction Effects on the Excitation Energy of Organic Pigments: Solid State Quinacridone.
H. Fukunaga, D.G.Fedorov, M. Chiba, K. Nii, K. Kitaura
J. Phys. Chem. A 112, 10887-10894 (2008).
17. Covalent Bond Fragmentation Suitable To Describe Solids in the Fragment Molecular Orbital Method.
D. G. Fedorov, J. H. Jensen, R. C. Deka, K. Kitaura
J. Phys. Chem. A 112, 11808-11816 (2008).
18. Excited state geometry optimizations by time-dependent density functional theory based on the fragment molecular orbital method.
M. Chiba, D. G. Fedorov, T. Nagata, K. Kitaura
Chem. Phys. Lett. 474, 227-232 (2009).
19. Derivatives of the approximated electrostatic potentials in the fragment molecular orbital method.
T. Nagata, D. G. Fedorov, K. Kitaura,
Chem. Phys. Lett. 475, 124-131 (2009).
20. A combined effective fragment potential - fragment molecular orbital method. I. The energy expression and initial applications.
T. Nagata, D. G. Fedorov, K. Kitaura, M. S. Gordon,
J. Chem. Phys. 131, 024101 (2009).
21. Analytic gradient for the adaptive frozen orbital bond detachment in the fragment molecular orbital method.
D. G. Fedorov, P. V. Avramov, J.H. Jensen, K. Kitaura,
Chem. Phys. Lett. 477, 169-175 (2009).
22. Fragment molecular orbital study of the electronic excitations in the photosynthetic reaction center of *Blastochloris viridis*.
T. Ikegami, T. Ishida, D. G. Fedorov, K. Kitaura,
Y. Inadomi, H. Umeda, M. Yokokawa, S. Sekiguchi,
J. Comp. Chem. 31, 447-454 (2010).
23. Open-Shell Formulation of the Fragment Molecular Orbital Method.
S. R. Pruitt, D. G. Fedorov, K. Kitaura, M. S. Gordon

- J. Chem. Theor. Comp. 6, 1-5 (2010)
24. Energy gradients in combined fragment molecular orbital and polarizable continuum model (FMO/PCM) calculation.
H. Li, D. G. Fedorov, T. Nagata, K. Kitaura,
J. H. Jensen, M. S. Gordon
J. Comput. Chem. 31, 778-790 (2010).
25. Nuclear-Electronic Orbital Method within the Fragment Molecular Orbital Approach.
B. Auer, M. V. Pak, S. Hammes-Schiffer,
J. Phys. Chem. C 114, 5582-5588 (2010).
26. Importance of the hybrid orbital operator derivative term for the energy gradient in the fragment molecular orbital method.
T. Nagata, D. G. Fedorov, K. Kitaura,
Chem. Phys. Lett. 492, 302-308 (2010).
27. Systematic Study of the Embedding Potential Description in the Fragment Molecular Orbital Method.
D. G. Fedorov, L. V. Slipchenko, K. Kitaura,
J. Phys. Chem. A 114, 8742-8753 (2010).
28. A combined effective fragment potential - fragment molecular orbital method. II. Analytic gradient and application to the geometry optimization of solvated tetraglycine and chignolin.
T. Nagata, D. G. Fedorov, T. Sawada, K. Kitaura,
M. S. Gordon, J. Chem. Phys. 134, 034110 (2011).
29. Geometry optimization of the active site of a large system with the fragment molecular orbital method.
D. G. Fedorov, Y. Alexeev, K. Kitaura,
J. Phys. Chem. Lett. 2, 282-288 (2011).
30. Fully analytic energy gradient in the fragment molecular orbital method.
T. Nagata, K. Brorsen, D. G. Fedorov, K. Kitaura,
M. S. Gordon, J. Chem. Phys. 134, 124115(2011).
31. Analytic energy gradient for second-order Moeller-Plesset perturbation theory based on the fragment molecular orbital method.
T. Nagata, D. G. Fedorov, K. Ishimura, K. Kitaura,
J. Chem. Phys. 135, 044110 (2011).
32. Large-Scale MP2 Calculations on the Blue Gene Architecture Using the Fragment Molecular Orbital Method.
G. D. Fletcher, D. G. Fedorov, S.R.Pruitt, T.L.Windus,
M. S. Gordon, J. Chem. Theory Comput. 8, 75-79(2012).
33. Energy decomposition analysis in solution based on the fragment molecular orbital method.
D.G.Fedorov, K.Kitaura
J.Phys.Chem. A 116, 704-719(2012).
34. Analytic gradient and molecular dynamics simulations using the fragment molecular orbital method combined with effective potentials.

- T. Nagata, D. G. Fedorov, K. Kitaura
Theor. Chem. Acc. 131, 1136 (2012).
35. Geometry Optimizations of Open-Shell Systems with the Fragment Molecular Orbital Method.
S. R. Pruitt, D. G. Fedorov, M. S. Gordon,
J. Phys. Chem. A, 116, 4965-4974 (2012).
36. Analytic gradient for second order Moeller-Plesset perturbation theory with the polarizable continuum model based on the fragment molecular orbital method.
T. Nagata, D. G. Fedorov, H. Li, K. Kitaura,
J. Chem. Phys., 136, 204112 (2012).
37. Reducing scaling of the fragment molecular orbital method using the multipole method.
C. H. Choi, D. G. Fedorov
Chem. Phys. Lett. 543, 159-165(2012).
38. Unrestricted Hartree-Fock based on the fragment molecular orbital method: energy and its analytic gradient.
H. Nakata, D. G. Fedorov, T. Nagata, S. Yokojima, K. Ogata, K. Kitaura, S. Nakamura
J. Chem. Phys. 137, 044110 (2012).
39. Analytic gradient for the embedding potential with approximations in the fragment molecular orbital method.
T. Nagata, D. G. Fedorov, K. Kitaura
Chem. Phys. Lett. 544, 87-93 (2012).
40. Analysis of solute-solvent interactions in the fragment molecular orbital method interfaced with the effective fragment potentials: theory and application to solvated griffithsin-carbohydrate complex.
T. Nagata, D. G. Fedorov, T. Sowada, K. Kitaura
J. Phys. Chem. A, 116, 9088-9099 (2012).
41. Open-shell pair interaction energy decomposition analysis (PIEDA): Formulation and application to the hydrogen abstraction in tripeptides.
M.C.Green, D.G.Fedorov, K.Kitaura, J.S.Francisco,
L.V.Slipchenko, J.Chem.Phys. 138 (2013) 074111.

Other FMO references including applications can be found at:

<http://staff.aist.go.jp/d.g.fedorov/fmo/main.html>

EFMO references are given in its own subsection.

The Cluster-in-Molecules method

sequential and parallel execution

If the user is not interested in parallel CIM calculations, MTDCIM must be set at 0, which is a default value, and no additional steps have to be taken. If the user is interested in a parallel execution, MTDCIM must initially be set at 1 to prepare for individual subsystem GAMESS runs, and then, after the individual subsystem runs are completed, reset to 2 to complete the CIM calculation. When MTDCIM is initially set at 1, multiple input files \$JOB.Sys-N.inp for individual subsystem calculations with GAMESS, which can be run independent of one another, and the \$JOB.cim file, which contains the information about all subsystems needed to complete the CIM calculation, are automatically generated, and the program stops awaiting further execution. Each subsystem N has then to be run as an independent GAMESS calculation using the \$JOB.Sys-N.inp input file. This produces the \$JOB.Sys-N.cim files which contain the information about the correlation energy contributions due to the occupied LMOs central in subsystems. All \$JOB.Sys-N.cim files resulting from the individual subsystem calculations with GAMESS and the \$JOB.cim file are used to assemble the final results of a CIM calculation for the entire system. In order to accomplish this and complete the CIM run, one has to reset MTDCIM in the main \$JOB.inp input file to 2 and run GAMESS again.

restarts

If any of the subsystem GAMESS calculations using the \$JOB.Sys-N.inp input files fails, the user can always rerun it (editing the corresponding \$JOB.Sys-N.inp file(s), if need be), and then use MTDCIM=2 to complete the desired CIM calculation for the entire system. This applies to sequential and parallel CIM calculations. In the latter case, this is a natural consequence of the way the parallel execution is structured (see note 1). In the former case (MTDCIM=0), if the entire calculation is completed, the \$JOB.Sys-N.* subsystem files are deleted, but if one of the subsystem calculations fails, the program aborts, leaving all \$JOB.Sys-N.inp input files, the \$JOB.Sys-N.cim output files from the completed subsystem calculations, and the

`$JOB.cim` file on the disk. One can rerun the subsystem GAMESS calculation that failed, editing the corresponding `$JOB.Sys-N.inp` file if need be, and the remaining subsystem calculations, and then, once all subsystem GAMESS runs are completed, finish the calculation by using `MTDCIM=2` in the main `$JOB.inp` input. This has an advantage over the more automated method of restarting the sequential CIM calculations described below in that the Hartree-Fock and orbital localization calculations for the entire system do not have to be repeated.

If the sequential (`MTDCIM=0`) run does not complete due to the failure of one of the subsystem GAMESS calculations, one can also follow a simpler, more automated restart strategy. At the time of failure, all `$JOB.Sys-N.inp` input files, the `$JOB.Sys-N.cim` output files from the completed subsystem calculations, and the `$JOB.cim` file are saved on the disk. After inspecting the main output file, the user can simply delete the `$JOB.dat` file and the `$JOB.Sys-N.cim` file resulting from the failed subsystem calculation, edit the corresponding `$JOB.Sys-N.inp` file, if necessary, and rerun the GAMESS calculation with `MTDCIM=0`. The Hartree-Fock and orbital localization calculations for the entire system will be performed again, but the user will avoid the need for running individual subsystem calculations one-by-one, as described above.

the cimshell script

The Python script "cimshell" that automatically produces typical `$JOB.sh` files for parallel OpenMP and MPI subsystem calculations using the `$JOB.Sys-N.inp` files can be found in `$GMS_PATH/tools/cim/`, where `$GMS_PATH` is the GAMESS main directory. In order to run the subsystem calculations with OpenMP or MPI, and with the help of the "cimshell" script, the following steps should be performed:

1. Run GAMESS CIM calculation using `MTDCIM=1` to produce the `$JOB.Sys-N.inp` and `$JOB.cim` files.
2. After all subsystem `$JOB.Sys-N.inp` input files are generated, use "cimshell" to automatically generate the OpenMP or MPI script `$JOB.sh` for parallel execution. By default, the "cimshell" program must be run in the directory where the `$JOB.inp` and all `$JOB.Sys-N.inp` files reside. For example, "cimshell --np 4 `$JOB`" generates the `$JOB.sh` script for an OpenMP parallel calculation on 4 cores, whereas "cimshell --np 8 --para mpi --submit pbs --MPI_EXEC mpiexec `$JOB`" generates the `$JOB.sh` script for an

- MPI parallel calculation on 8 processors using the PBS queue system for submitting the job. In these two examples, we are assuming that the "cimshell" has been copied to the directory where all \$JOB.Sys-N.inp files reside; this can be altered by redefining the \$PATH variable (adding \$GMS_PATH/tools/cim/ to it). Use "cimshell -h" for more information about the "cimshell" options.
3. Run or submit the \$JOB.sh script to have subsystem calculations performed in parallel. In order to do this, the user must compile the ompjob.for (the OpenMP case) or mpijob.for (the MPI case) programs that reside in \$GMS_PATH/tools/cim/. The corresponding executables used by \$JOB.sh are called ompjob and mpijob, respectively. The example of the Makefile that can be used to install ompjob and mpijob can be found in \$GMS_PATH/tools/cim/.
 4. After all subsystem calculations are completed, run the final GAMESS CIM calculation using the original input file \$JOB.inp in which with MTDCIM=2. GAMESS will automatically find the relevant \$JOB.Sys-N.cim and \$JOB.cim files to complete the CIM calculation for the entire system and print the final CIM energies in the main output.

CIM references

THE FOLLOWING PAPERS SHOULD BE CITED WHEN USING CLUSTER-IN-MOLECULE OPTIONS:

DUAL-ENVIRONMENT CIM (CIMTYP=DECIM)

W. LI, P. PIECUCH, J.R. GOUR, AND S. LI,
J. CHEM. PHYS. 131, 114109-1 - 114109-30 (2009).
SEE, ALSO, S. LI, J. SHEN, W. LI, AND Y. JIANG,
J. CHEM. PHYS. 125, 074109-1 - 074109-10 (2006)

SINGLE-ENVIRONMENT CIM (CIMTYP=SECIM,GSECIM)

W. LI, P. PIECUCH, J.R. GOUR, AND S. LI,
J. CHEM. PHYS. 131, 114109-1 - 114109-30 (2009);
W. LI AND P. PIECUCH, J. PHYS. CHEM. A 114, 8644-8657
(2010).

IN ADDITION, THE USE OF MULTI-LEVEL CIM SHOULD REFERENCE
W. LI AND P. PIECUCH, J. PHYS. CHEM. A 114, 6721-6727
(2010).

MOPAC Calculations within GAMESS

Parts of MOPAC 6.0 have been included in GAMESS giving access to four semiempirical wavefunctions: MNDO, AM1, PM3, and RM1. RM1 is the most recent parameterization, replacing AM1 data for H, C-F, P-Cl, Br, and I. See G. Bruno Rocha, R. Oliveira Freire, A. Mayall Simas, and J.J.P. Stewart, J.Comput.Chem. 27, 1101-1111(2006).

These wavefunctions are quantum mechanical in nature but neglect most two electron integrals, a deficiency that is (hopefully) compensated for by introduction of empirical parameters. The quantum mechanical nature of semiempirical theory makes it quite compatible with the ab initio methodology in GAMESS. As a result, very little of MOPAC 6.0 actually is incorporated into GAMESS. The part that did survive is the code that evaluates

- 1) the one- and two-electron integrals,
- 2) the two-electron part of the Fock matrix,
- 3) the cartesian energy derivatives, and
- 4) the ZDO atomic charges and molecular dipole.

Everything else is actually GAMESS: coordinate input (including point group symmetry), the SCF convergence procedures, the matrix diagonalizer, the geometry searcher, the numerical hessian driver, and so on. Most of the output will look like an ab initio output.

It is extremely simple to perform these calculations. All you need to do is specify GBASIS=MNDO, AM1, PM3, or RM1 in the \$BASIS group. Note that this not only selects a particular "Hamiltonian" (parameter set), it also picks a Slater Type Orbital (STO) basis.

MOPAC parameters exist for the following elements. The printout when you run will give you specific references for each kind of atom. The quote on alkali's below means that these elements are treated as "sparkles", rather than as atoms with genuine basis functions.

For MNDO:

H						
Li	*		B	C	N	O F
Na'	*		Al	Si	P	S Cl
K'	*	...	Zn	*	Ge	* * Br
Rb'	*	...	*	*	Sn	* * I
*	*	...	Hg	*	Pb	*

For AM1:

H							
*	*		B	C	N	O	F
Na	Mg		Al	Si	P	S	Cl
K	Ca	...	Zn	*	Ge	*	Br
Rb'	*	...	*	Sn	*	*	I
*	*	...	Hg	*	*	*	

For PM3:

H							
Li	Be		*	C	N	O	F
Na	Mg		Al	Si	P	S	Cl
K	Ca	...	Zn	Ga	Ge	As	Se
Rb'	*	...	Cd	In	Sn	Sb	Te
*	*	...	Hg	Tl	Pb	Bi	

For RM1:

H				
	C	N	O	F
		P	S	Cl
			Br	
			I	

RM1 uses AM1 parameters for any element not listed above.

* * * * *

MOPAC will not work with every option in GAMESS: the semiempirical wavefunctions must be RHF, UHF, and ROHF in any combination with run types ENERGY, GRADIENT, OPTIMIZE, SADPOINT, HESSIAN, and IRC. Note that nuclear hessian runs use numerical finite differencing of analytic gradients. MOPAC's CI and half electron methods are not supported.

Because the majority of the implementation is GAMESS rather than MOPAC6, you will notice a few improvements. Dynamic memory allocation is used, so GAMESS uses far less memory for a given size of molecule. The starting orbitals for SCF calculations are generated by a Huckel initial guess routine. Spin restricted (high spin) ROHF can be performed. Converged SCF orbitals will be labeled by their symmetry type. Numerical hessians will make use of point group symmetry, so that only the symmetry unique atoms need to be displaced. Infrared intensities will be calculated at the end of hessian runs. We have not at present used the block diagonalizer during intermediate SCF iterations, so that the run time for a single geometry point in GAMESS is usually longer than in MOPAC. However, the geometry optimizer in GAMESS can frequently optimize the structure in fewer steps than the procedure in MOPAC. Orbitals and hessians are punched out for convenient reuse in subsequent calculations. Your molecular orbitals can be drawn with the PLTORB graphics program, which has been taught about s and p STO basis sets.

However, because of the STO basis set used in semi-empirical runs, the various property calculations coded for Gaussian (GTO) basis sets are unavailable. This means

\$ELMOM, \$ELPOT, etc. properties are unavailable. Note that MOPAC6 did not include d STO integrals, so it is quite impossible to run transition metals.

The PCM solvation model implemented in GAMESS can be used with MOPAC runs by GAMESS.

To reduce CPU time, by default only the EXTRAP convergence accelerator is used by the SCF procedures. For difficult cases, the DIIS, RSTRCT, and/or SHIFT options will work, but may add significantly to the run time. With the Huckel guess procedure from GAMESS, most calculations will converge acceptably without these special options.

The MOPAC implementation is able to run in parallel.

Semiempirical calculations are very fast. One of the motives for the MOPAC implementation within GAMESS is to take advantage of this speed. Semiempirical models can rapidly provide reasonable starting geometries for ab initio optimizations. Semiempirical hessian matrices are obtained at virtually no computational cost, and may help dramatically with an ab initio geometry optimization. Simply use HESS=READ in \$STATPT to use a MOPAC \$HESS group in an ab initio run.

It is important to exercise caution as semiempirical methods can be dead wrong! The reasons for this are bad parameters (in certain chemical situations), and the underlying minimal basis set. A good question to ask before using MOPAC is "how well is my system modeled by an ab initio minimal basis set, such as STO-3G"? If the answer is "not very well", there is a good chance that a semiempirical description is equally poor.

Molecular Properties and Conversion Factors

These two papers are of general interest:

A.D.Buckingham, J.Chem.Phys. 30, 1580-1585(1959).

D.Neumann, J.W.Moskowitz J.Chem.Phys. 49, 2056-2070(1968).

The first deals with multipoles, and the second with other properties such as electrostatic potentials.

All units are derived from the atomic units for distance and the monopole electric charge, as given below.

distance	1 au = 5.291771E-09 cm
monopole	1 au = 4.803242E-10 esu 1 esu = sqrt(g-cm**3)/sec
dipole	1 au = 2.541766E-18 esu-cm 1 Debye = 1.0E-18 esu-cm
quadrupole	1 au = 1.345044E-26 esu-cm**2 1 Buckingham = 1.0E-26 esu-cm**2
octopole	1 au = 7.117668E-35 esu-cm**3
electric potential	1 au = 9.076814E-02 esu/cm
electric field	1 au = 1.715270E+07 esu/cm**2 1 esu/cm**2 = 1 dyne/esu
electric field gradient	1 au = 3.241390E+15 esu/cm**3

The atomic unit for the total electron density is electron/bohr**3, but 1/bohr**3 for an orbital density.

The atomic unit for spin density is excess alpha spins per unit volume, $h/4\pi \cdot \text{bohr}^3$. Only the expectation value is computed, with no constants premultiplying it.

IR intensities are printed in Debye**2/amu-Angstrom**2. These can be converted into intensities as defined by Wilson, Decius, and Cross's equation 7.9.25, in km/mole, by multiplying by 42.255. If you prefer 1/atm-cm**2, use a conversion factor of 171.65 instead. A good reference for deciphering these units is A.Komornicki, R.L.Jaffe J.Chem.Phys. 1979, 71, 2150-2155. A reference showing how IR intensities change with basis improvements at the HF

level is Y.Yamaguchi, M.Frisch, J.Gaw, H.F.Schaefer, J.S.Binkley, J.Chem.Phys. 1986, 84, 2262-2278.

Raman activities in $\text{Å}^4/\text{amu}$ multiply by $6.0220\text{E}-09$ for units of cm^4/g . One of the many sources explaining how activity relates to intensity is D.Michalska, R.Wysokinski Chem.Phys.Lett. 403, 211-217(2005)

Polarizabilities

Static polarizabilities are named alpha, beta, and gamma; these are called the polarizability, hyperpolarizability, and second hyperpolarizability. They are the 2nd, 3rd, and 4th derivatives of the energy with respect to uniform applied electric fields, with the 1st derivative being the dipole moment.

It is worth mentioning that a uniform (static) electric field can be applied using \$EFIELD, if you wish to develop custom usages, but \$EFIELD input must not be given for any kind of run discussed below.

A general approach to computing static polarizabilities is numerical differentiation, namely RUNTYP=FFIELD, which should work for any energy method provided by GAMESS. A sequence of computations with fields applied in the x, y, and/or z directions will generate the three alpha, beta, and gamma tensors. See \$FFCALC for details. Analytic computation of all three tensors is available for closed shells only, see RUNTYP=TDHF and \$TDHF input, or TDDFT=HPOL and \$TDDFT input. If you need to know just the static alpha polarizability, see POLAR in \$CPHF during any analytic hessian job.

A break down of the static alpha polarizability in terms of contributions from individual localized orbitals can be obtained by setting POLDCM=.TRUE. in \$LOCAL. Calculation will be by analytic means, unless POLNUM in that group is selected. This option is available only for SCFTYP=RHF. The keyword LOCHYP in \$FFCALC gives a similar analysis for all three static polarizabilities, determined by numerical differentiation.

Polarizabilities in a static electric field differ from those in an oscillating field, such as a laser produces. These are called frequency dependent alpha, beta, or gamma,

and in the limit of entering a zero frequency, become the static quantities discussed just above.

For RHF cases, various frequency dependent alpha, beta, and gamma polarizabilities can be generated, depending on the experiment. A particularly easy one to understand is 'second harmonic generation', governed by a beta tensor describing the absorption of two photons with the emission of one photon at doubled frequency. See RUNTYP=TDHF, and papers listed under \$TDHF, for many other non-linear optical experiments. A program for the computation of the frequency dependent beta hyperpolarizability at the DFT level is also available, for closed shell molecules: see TDDFT=HPOL and keywords in \$TDDFT input.

Nuclear derivatives of the dipole moment and the various polarizabilities are also of interest. For example, knowledge of the derivative of the dipole with respect to nuclear coordinates yields the IR intensity. Similarly, the nuclear derivative of the static alpha polarizability gives Raman activities: see RUNTYP=RAMAN. Analytically computed 1st or 2nd nuclear derivatives of static or frequency dependent polarizabilities are available for SCFTYP=RHF, see RUNTYP=TDHFX and \$TDHFX, giving rise to experimental observations such as resonance Raman and hyper-Raman.

Finally, instead of considering polarizabilities to be a function of real frequencies, they can be considered to be dependent on the imaginary frequency. The imaginary frequency dependent alpha polarizability can be computed analytically for SCFTYP=RHF only, using POLDYN=.TRUE. in \$LOCAL. Integration of this quantity over the imaginary frequency domain can be used to extract C6 dispersion constants.

Polarizabilities are tensor quantities. There are a number of different ways to define them, and various formulae to extract "scalar" and "vector" quantities from the tensors. A good reference for learning how to compare the output of a theoretical program to experiment is

A.Willetts, J.E.Rice, D.M.Burland, D.P.Shelton
J.Chem.Phys. 97, 7590-7599(1992)

Localized Molecular Orbitals

Three different orbital localization methods are implemented in GAMESS. The energy and dipole based methods normally produce similar results, but see M.W.Schmidt, S.Yabushita, M.S.Gordon in J.Chem.Phys., 1984, 88, 382-389 for an interesting exception. You can find references to the three methods at the beginning of this chapter.

The method due to Edmiston and Ruedenberg works by maximizing the sum of the orbitals' two electron self repulsion integrals. Most people who think about the different localization criteria end up concluding that this one seems superior. The method requires the two electron integrals, transformed into the molecular orbital basis. Because only the integrals involving the orbitals to be localized are needed, the integral transformation is actually not very time consuming.

The Boys method maximizes the sum of the distances between the orbital centroids, that is the difference in the orbital dipole moments.

The population method due to Pipek and Mezey maximizes a certain sum of gross atomic Mulliken populations. This procedure will not mix sigma and pi bonds, so you will not get localized banana bonds. Hence it is rather easy to find cases where this method give different results than the Ruedenberg or Boys approach.

GAMESS will localize orbitals for any kind of RHF, UHF, ROHF, or MCSCF wavefunctions. The localizations will automatically restrict any rotation that would cause the energy of the wavefunction to be changed (the total wavefunction is left invariant). As discussed below, localizations for GVB or CI functions are not permitted.

The default is to freeze core orbitals. The localized valence orbitals are scarcely changed if the core orbitals are included, and it is usually convenient to leave them out. Therefore, the default localizations are: RHF functions localize all doubly occupied valence orbitals. UHF functions localize all valence alpha, and then all valence beta orbitals. ROHF functions localize all valence doubly occupied orbitals, and all singly occupied orbitals, but do not mix these two orbital spaces. MCSCF functions

localize all valence MCC type orbitals, and localize all active orbitals, but do not mix these two orbital spaces. To recover the invariant MCSCF function, you must be using a `FORS=.TRUE.` wavefunction, and you must set `GROUP=C1` in `$DRT`, since the localized orbitals possess no symmetry.

In general, GVB functions are invariant only to localizations of the NCO doubly occupied orbitals. Any pairs must be written in natural form, so pair orbitals cannot be localized. The open shells may be degenerate, so in general these should not be mixed. If for some reason you feel you must localize the doubly occupied space, do a `RUNTYP=PROP` job. Feed in the GVB orbitals, but tell the program it is `SCFTYP=RHF`, and enter a negative `ICHARG` so that GAMESS thinks all orbitals occupied in the GVB are occupied in this fictitious RHF. Use `NINA` or `NOUTA` to localize the desired doubly occupied orbitals. Orbital localization is not permitted for CI, because we cannot imagine why you would want to do that anyway.

Boys localization of the core orbitals in molecules having elements from the third or higher row almost never succeeds. Boys localization including the core for second row atoms will often work, since there is only one inner shell on these. The Ruedenberg method should work for any element, although including core orbitals in the integral transformation is more expensive.

The easiest way to do localization is in the run which generates the wavefunction, by selecting `LOCAL=xxx` in the `$CONTRL` group. However, localization may be conveniently done at any time after determination of the wavefunction, by executing a `RUNTYP=PROP` job. This will require only `$CONTRL`, `$BASIS/$DATA`, `$GUESS` (pick `MOREAD`), the converged `$VEC`, possibly `$SCF` or `$DRT` to define your wavefunction, and optionally some `$LOCAL` input.

There is an option to restrict all rotations that would mix orbitals of different symmetries. `SYMLOC=.TRUE.` yields only partially localized orbitals, but these still possess symmetry. They are therefore very useful as starting orbitals for MCSCF or GVB-PP calculations. Because they still have symmetry, these partially localized orbitals run as efficiently as the canonical orbitals. Because it is much easier for a user to pick out the bonds which are to be correlated, a significant number of iterations can be saved, and convergence to false solutions is less likely.

* * *

The most important reason for localizing orbitals is to analyze the wavefunction. A simple example is to look at shapes of the orbitals with the MacMolPlt program. Or, you might read the localized orbitals in during a RUNTYP=PROP job to examine their Mulliken populations.

Localized orbitals are a particularly interesting way to analyze MCSCF computations. The localized orbitals may be oriented on each atom (see option ORIENT in \$LOCAL) to direct the orbitals on each atom towards their neighbors for maximal bonding, and then print a bond order analysis. The orientation procedure is newly programmed by J.Ivanic and K.Ruedenberg, to deal with the situation of more than one localized orbital occurring on any given atom. Some examples of this type of analysis are

D.F.Feller, M.W.Schmidt, K.Ruedenberg

J.Am.Chem.Soc. 104, 960-967 (1982)

T.R.Cundari, M.S.Gordon

J.Am.Chem.Soc. 113, 5231-5243 (1991)

N.Matsunaga, T.R.Cundari, M.W.Schmidt, M.S.Gordon

Theoret.Chim.Acta 83, 57-68 (1992).

In addition, the energy of your molecule can be partitioned over the localized orbitals so that you may be able to understand the origin of barriers, etc. This analysis can be made for the SCF energy, and also the MP2 correction to the SCF energy, which requires two separate runs. An explanation of the method, and application to hydrogen bonding may be found in J.H.Jensen, M.S.Gordon, J.Phys.Chem. 99, 8091-8107(1995).

Analysis of the SCF energy is based on the localized charge distribution (LCD) model: W.England and M.S.Gordon, J.Am.Chem.Soc. 93, 4649-4657 (1971). This is implemented for RHF and ROHF wavefunctions, and it requires use of the Ruedenberg localization method, since it needs the two electron integrals to correctly compute energy sums. All orbitals must be included in the localization, even the cores, so that the total energy is reproduced.

The LCD requires both electronic and nuclear charges to be partitioned. The orbital localization automatically accomplishes the former, but division of the nuclear charge may require some assistance from you. The program attempts to correctly partition the nuclear charge, if you select the MOIDON option, according to the following: a Mulliken type analysis of the localized orbitals is made. This determines if an orbital is a core, lone pair, or

bonding MO. Two protons are assigned to the nucleus to which any core or lone pair belongs. One proton is assigned to each of the two nuclei in a bond. When all localized orbitals have been assigned in this manner, the total number of protons which have been assigned to each nucleus should equal the true nuclear charge.

Many interesting systems (three center bonds, back-bonding, aromatic delocalization, and all charged species) may require you to assist the automatic assignment of nuclear charge. First, note that MOIDON reorders the localized orbitals into a consistent order: first comes any core and lone pair orbitals on the 1st atom, then any bonds from atom 1 to atoms 2, 3, ..., then any core and lone pairs on atom 2, then any bonds from atom 2 to 3, 4, ..., and so on. Let us consider a simple case where MOIDON fails, the ion NH_4^+ . Assuming the nitrogen is the 1st atom, MOIDON generates

```

NNUCMO=1,2,2,2,2
MOIJ=1,1,1,1,1
      2,3,4,5
ZIJ=2.0,1.0,1.0,1.0,1.0,
      1.0,1.0,1.0,1.0

```

The columns (which are LMOs) are allowed to span up to 5 rows (the nuclei), in situations with multicenter bonds. MOIJ shows the Mulliken analysis thinks there are four NH bonds following the nitrogen core. ZIJ shows that since each such bond assigns one proton to nitrogen, the total charge of N is +6. This is incorrect of course, as indeed will always happen to some nucleus in a charged molecule. In order for the energy analysis to correctly reproduce the total energy, we must ensure that the charge of nitrogen is +7. The least arbitrary way to do this is to increase the nitrogen charge assigned to each NH bond by 1/4. Since in our case NNUCMO and MOIJ and much of ZIJ are correct, we need only override a small part of them with \$LOCAL input:

```

IJMO(1)=1,2, 1,3, 1,4, 1,5
ZIJ(1)=1.25, 1.25, 1.25, 1.25

```

which changes the charge of the first atom of orbitals 2 through 5 to 5/4, changing ZIJ to

```

ZIJ=2.0,1.25,1.25,1.25,1.25,
      1.0, 1.0, 1.0, 1.0

```

The purpose of the IJMO sparse matrix pointer is to let you give only the changed parts of ZIJ and/or MOIJ.

Another way to resolve the problem with NH_4^+ is to change one of the 4 equivalent bond pairs into a "proton". A "proton" orbital AH treats the LMO as if it were a

lone pair on A, and so assigns +2 to nucleus A. Use of a "proton" also generates an imaginary orbital, with zero electron occupancy. For example, if we make atom 2 in NH₄⁺ a "proton", by

```
IPROT(1)=2
NNUCMO(2)=1
IJMO(1)=1,2,2,2  MOIJ(1)=1,0  ZIJ(1)=2.0,0.0
the automatic decomposition of the nuclear charges will be
NNUCMO=1,1,2,2,2,1
MOIJ=1,1,1,1,1,2
      3,4,5
ZIJ=2.0,2.0,1.0,1.0,1.0,1.0
      1.0,1.0,1.0
```

The 6th column is just a proton, and the decomposition will not give any electronic energy associated with this "orbital", since it is vacant. Note that the two ways we have dissected the nuclear charges for NH₄⁺ will both yield the correct total energy, but will give very different individual orbital components. Most people will feel that the first assignment is the least arbitrary, since it treats all four NH bonds equivalently.

However you assign the nuclear charges, you must ensure that the sum of all nuclear charges is correct. This is most easily verified by checking that the energy sum equals the total SCF energy of your system.

As another example, H₃PO is studied in EXAM26.INP. Here the MOIDON analysis decides the three equivalent orbitals on oxygen are 0 lone pairs, assigning +2 to the oxygen nucleus for each orbital. This gives Z(O)=9, and Z(P)=14. The least arbitrary way to reduce Z(O) and increase Z(P) is to recognize that there is some backbonding in these "lone pairs" to P, and instead assign the nuclear charge of these three orbitals by 1/3 to P, 5/3 to O.

Because you may have to make several runs, looking carefully at the localized orbital output before the correct nuclear assignments are made, there is an option to skip directly to the decomposition when the orbital localization has already been done. Use

```
$CONTRL RUNTYP=PROP
$GUESS  GUESS=MOREAD  NORB=
$VEC containing the localized orbitals!
$TWOEI
```

The latter group contains the necessary Coulomb and exchange integrals, which are punched by the first localization, and permits the decomposition to begin

immediately.

SCF level dipoles can also be analyzed using the DIPDCM flag in \$LOCAL. The theory of the dipole analysis is given in the third paper of the LCD sequence. The following list includes application of the LCD analysis to many problems of chemical interest:

W.England, M.S.Gordon J.Am.Chem.Soc. 93, 4649-4657 (1971)
W.England, M.S.Gordon J.Am.Chem.Soc. 94, 4818-4823 (1972)
M.S.Gordon, W.England J.Am.Chem.Soc. 94, 5168-5178 (1972)
M.S.Gordon, W.England Chem.Phys.Lett. 15, 59-64 (1972)
M.S.Gordon, W.England J.Am.Chem.Soc. 95, 1753-1760 (1973)
M.S.Gordon J.Mol.Struct. 23, 399 (1974)
W.England, M.S.Gordon, K.Ruedenberg,
Theoret.Chim.Acta 37, 177-216 (1975)
J.H.Jensen, M.S.Gordon, J.Phys.Chem. 99, 8091-8107(1995)
J.H.Jensen, M.S.Gordon, J.Am.Chem.Soc. 117, 8159-8170(1995)
M.S.Gordon, J.H.Jensen, Acc.Chem.Res. 29, 536-543(1996)

* * *

It is also possible to analyze the MP2 correlation correction in terms of localized orbitals, for the RHF case. The method is that of G.Peterssen and M.L.Al-Laham, J.Chem.Phys., 94, 6081-6090 (1991). Any type of localized orbital may be used, and because the MP2 calculation typically omits cores, the \$LOCAL group will normally include only valence orbitals in the localization. As mentioned already, the analysis of the MP2 correction must be done in a separate run from the SCF analysis, which must include cores in order to sum up to the total SCF energy.

* * *

Typically, the results are most easily interpreted by looking at "the bigger picture" than at "the details". Plots of kinetic and potential energy, normally as a function of some coordinate such as distance along an IRC, are the most revealing. Once you determine, for example, that the most significant contribution to the total energy is the kinetic energy, you may wish to look further into the minutia, such as the kinetic energies of individual localized orbitals, or groups of LMOs corresponding to an entire functional group.

Transition Moments and Spin-Orbit Coupling

A review of various ways of computing spin-orbit coupling:
 D.G.Fedorov, S.Koseki, M.W.Schmidt, M.S.Gordon,
 Int.Rev.Phys.Chem. 22, 551-592(2003)

GAMESS can compute transition moments and oscillator strengths for the radiative transitions between states written in terms of CI wavefunctions (GUGA only). The transition moments are computed using both the "length form" and the "velocity form". In a.u., where $\hbar=m=1$, we start from

$$[A,q] = -i \, dA/dp$$

For $A=H$, $dH/dp=p$, and $p = -i \, d/dq$,

$$[H,q] = -i \, p = -d/dq.$$

For non-degenerate states,

$$\langle a|[H,q]|b \rangle = \langle a|-d/dq|b \rangle$$

$$(E_a-E_b)\langle a|q|b \rangle = - \langle a|d/dq|b \rangle$$

This relates the dipole to the velocity form,

$$\langle a|q|b \rangle = -1/(E_a-E_b) \langle a|d/dq|b \rangle$$

but the CI states will give different numbers for each side, since the states aren't exact eigenfunctions.

Transition moment computation is OPERAT=DM in \$TRANST. For transition moments, the CI is necessarily performed on states of the same multiplicity.

All other operators are various spin-orbit coupling options. There are two kinds of calculations possible, which we will call SO-CI and SO-MCQDPT. Note that there is a hyphen in "spin-orbit CI" to avoid confusion with "second order CI" in the sense of the SOCI keyword in \$DRT input. For SO-CI, the initial states may be any CI wavefunction that the GUGA package can generate. For SO-MCQDPT the initial states for spin-orbit coupling are of CAS type, and the operator mixing them corresponds to MCQDPT generalised for spin-dependent operators (with certain approximations).

GAMESS can compute the "microscopic Breit-Pauli spin-orbit operator", which includes both a one and two electron operator. Additional information is given in a subsection below

states

For transition moments, the states are generated by CI calculations using the GUGA package. These states are

the final states, and the results are just the transition moments between these states. The states are defined by \$DRTx input groups.

For SO-CI, the energy of the CI states forms the diagonal of a spin-orbit Hamiltonian, as in the state basis the spin-free Hamiltonian is of course diagonal. Addition of the Pauli-Breit operator does not change the diagonal, but does add off-diagonal H-so elements. For SO-MCQDPT, the spin-free MCQDPT matrix elements are expanded into matrices corresponding to all Ms values for a pair of multiplicities. These matrices are block-diagonal before the addition of spin-orbit coupling terms, coupling Ms values. The diagonalization of this spin-orbit Hamiltonian gives new energy levels, and spin-mixed final states. Optionally, the transition dipoles between the final states can be computed. The input requirements are \$DRTx or \$MCQDx groups which define the original pure spin states.

We will call the initial states CAS-CI, since most of the time they will be MCSCF states. There may be cases such as the Na example below where SCF orbitals are used, or other cases where a FOCI or SOCI wavefunction might be used for the initial states. Please keep in mind that the term does not imply the states must be MCSCF states, just that they commonly are.

In the above, x may vary from 1 to 64. The reason for allowing such a large range is to permit the use of Abelian point group symmetry during the generation of the initial states. The best explanation will be an example, but the number of these input groups depends on both the number of orbital sets input, and how much symmetry is present. The next two subsections discuss these points.

orbitals

The orbitals for transition moments or for SO-CI can be one common set of orbitals used by all CI states. If one set of orbitals is used, the transition moment or spin-orbit coupling can be found for any type of GUGA CI wavefunction. Alternatively, two sets of orbitals (obtained by separate MCSCF orbital optimizations) can be used. Two or more separate CIs will be carried out. The two MO sets must share a common set of frozen core orbitals, and the CI must be of the complete active space type. These restrictions are needed to leave the CI wavefunctions invariant under the necessary rotation to corresponding

orbitals. The non-orthogonal procedure implemented is a GUGA driven equivalent to the method of Lengsfeld, et al. Note that the FOCI and SOCI methods described by these workers are not available in GAMESS.

If you would like to use separate orbitals during the CI, they may be generated with the FCORE option in \$MCSCF. Typically you would optimize the ground state completely, then use these MCSCF orbitals in an optimization of the excited state, under the constraint of FCORE=.TRUE.

For SO-MCQDPT calculations, only one set of orbitals may be input to describe all CAS-CI states. Typically that orbital set will be obtained by state-averaged MCSCF, see WSTATE in \$DET/\$DRT, and also in the \$MCQDx input. Note that although the RUNTYP=TRANSITN driver is tied to the GUGA CI package, there is no reason the orbitals cannot be obtained using the determinant CI package. In fact, for the case of spin-orbit coupling, you might want to utilize the ability to state average over several spins, see PURES in \$DET.

If there is no molecular symmetry present, transition moment calculations will provide \$DRT1 if there is one set of orbitals, otherwise \$DRT1 defines the CI based on \$VEC1 and \$DRT2 the CI based on \$VEC2. Also for the case of no symmetry, a spin-orbit job should enter one \$DRTx or \$MCQDx for every spin multiplicity, and all states of the same multiplicity have to be generated from \$VEC1 or \$VEC2, according to IVEX input.

symmetry

The CAS-CI states are most efficiently generated using symmetry, since states of different symmetry have zero Hamiltonian matrix elements. It is probably more efficient to do four CI calculations in the group C2v on A1, A2, B1, and B2 symmetry, than one CI with a combined Hamiltonian in C1 symmetry (unless the active space is very small), and similar remarks apply to the SO-MCQDPT case. In order to avoid repeatedly saying \$DRTx or \$MCQDx, the following few paragraphs say \$DRTx only.

Again supposing the group is C2v, and you are interested in singlet-triplet coupling. After some preliminary CI calculations, you might know that the lowest 8 states are two 1-a1, 1-b1, two 1-b2, one 3-a1, and two 3-

b2 states. In this case your input would consist of five \$DRTx, of which you can give the three singlets in any order but these must precede the two triplet input groups to follow the rule of increasing multiplicity. Clearly it is not possible to write a formula for how many \$DRTx there will be, this depends not only on the point group, but also the chemistry of the situation.

If you are using two sets of orbitals, the generation of the corresponding orbitals for the two sets will permute the active orbitals in an unpredictable way. Use STSYM to define the desired state symmetry, rather than relying on the orbital order. It is easy and safer to be explicit about the spatial orbital symmetry.

The users are encouraged to specify full symmetry in their \$DATA input even though they may choose to set the symmetry in \$DRTx to C1. The CI states will be labelled in the group given in \$DATA. The use of non-Abelian symmetry is limited by the absence of non-Abelian CI or MCQDPT. In this case the users can choose between setting full non-Abelian symmetry in \$DATA and C1 in \$DRT or else an Abelian subgroup in both \$DATA and \$DRT. The latter choice appears to be most efficient at present.

An example of SO-MCQDPT illustrating how the carbon atom of Kh symmetry (full rotation-reflection group) can be entered in D2h, Kh's highest Abelian group. The run time is considerably longer in C1 symmetry.

As another example, consider an organic molecule with a singly excited state, where that state might be coupled to low or high spin, and where these two states might be close enough to have a strong spin-orbit coupling. If it happens that the S1 and S0 states possess different symmetry, a very reasonable calculation would be to treat the S1 and T1 state with the same \$VEC2 orbitals, leaving the ground state described by \$VEC1. After doing an MCSCF on the S0 ground state for \$VEC1, you could do a state-averaged MCSCF for \$VEC2 optimized for T1 and S1 simultaneously, using PURES. The spin orbit job would obtain its initial states from three GUGA CI computations, S0 from \$VEC1 and \$DRT1, S1 from \$VEC2 and \$DRT2, and T1 from \$VEC2 and \$DRT3. Your \$TRANST would be NUMCI=3, IROOTS(1)=1,1,1, IVEX(1)=1,2,2. Note that the second IROOTS value is 1 because S1 was presumed to have a different symmetry than S0, so STSYM in \$DRT1 and \$DRT2 will differ. The calculation just outlined cannot be done if S0 and S1 have the same spatial symmetry, as IROOTS(1)=1,2,1 to obtain S1 during the second CI will

bring in an additional S0 state (one expressed in terms of the \$VEC2, at slightly higher energy). This problem is the origin of the statement several paragraphs above that a system with no symmetry will have one \$DRTx for every spin multiplicity included.

For transition moments, which do not diagonalize a matrix containing these duplicated states, it is OK to proceed, provided you ignore all transition moments between the same states obtained in the two different CIs.

spin orbit coupling

Spin-orbit coupling calculations are always performed in a quasi-degenerate perturbative manner. Typically the states close in energy are included into the spin-orbit coupling matrix. "Close" has a easily understandable meaning, since in the limit of small coupling the quasi-degenerate treatment is reduced to a second order perturbative treatment, that is, the affect of a state upon the state of primary interest is given by the square of the spin-orbit coupling matrix element divided by the difference of the adiabatic energies. This is useful to keep in mind when deciding how many CI states to include in the matrix. The states that are included are treated in a fashion that is equivalent to infinite order perturbation theory (exact) whereas the states that are not included make no contribution.

Spin-orbit runs can be done for even or odd numbers of electrons (any spin), for more than two different spin multiplicities at once, for general active spaces. At times, when the spatial wavefunction is degenerate, a spin-orbit run might involve only one spin multiplicity, e.g. a triplet-pi state in a linear molecule. The most common case is two different spins, with non-zero spin orbit coupling possible only for $\Delta S=1$: singlets spin-orbit couple with triplets, but not with quintets. Use of three spins, such as $S=0,1,2$, will generate couplings between singlets and triplets, and between triplets and quintets, which together engender an indirect singlet/quintet mixing.

As noted above, the treatment of spin-orbit involves first obtaining a handful of spin-pure states, whose energies form the diagonal of a model Hamiltonian. The spin-orbit operator introduces off-diagonal couplings, and the resulting small Hamiltonian is diagonalized. This

generates spin-mixed states in the model space. Since the model states are not fully relaxed (internally contracted), this is essentially a perturbative treatment: certainly the spin-orbit effects have no influence on orbital optimizations or potential energy surfaces when treated in this manner, at the very last stage.

The Breit-Pauli spin-orbit operator contains a one electron term arising from Pauli's reduction of the Dirac hydrogenic equation to a single-component form, and a two electron term due to Breit. Computation of the full Breit-Pauli operator is OPERAT=HSO2 (or HSOFF). A close approximation to the latter is HSO2P (P=partial), which neglects all active-active two electron terms, which usually do not contribute very much to the total coupling, while saving substantial computer time.

HSO1 completely omits the two electron terms, so is much faster than any of the two electron operators, but represents a potentially much greater loss of accuracy. HSO1's error can be remedied to some extent by regarding the nuclear charge in the one electron term as an adjustable parameter. In addition, these effective charges are often used to compensate for missing nodes in valence orbitals of ECP runs, in which case the ZEFF are typically very far from the two nuclear charges. ZEFTYP selects some built in values obtained by S.Koseki et al, but if you have some favorite parameters, they can be read in as the ZEFF input array. Effective charges may be used for any OPERAT, but are most often used with HSO1.

Theoretical considerations indicate that the Breit-Pauli operator is not variationally stable, if it were to be used during the SCF iterations determining orbitals. However, a first order Douglas-Kroll type correction to the one electron part of the operator reduces its size by means of certain kinematic factors, removing this problem. This can produce substantially better results, even if the operator is being treated perturbatively. This correction (at first order) is automatically applied to the one electron part of the Breit-Pauli operator by any run that selects RELWFN=DK (at any order) in the scalar relativity treatment during the variational steps prior to the spin-orbit perturbation. See paper 32 below.

Because the diagonalization of the model spin-orbit Hamiltonian leads to spin-mixed eigenvectors, approximate wavefunctions including spin-orbit coupling are generated. It is now possible to generate the density matrix for the

spin-mixed states, so that property values for spin-mixed states can be found: see keyword ISTNO. The natural orbitals of these spin-orbit density matrices turn out to be good approximations to the two spinors of the large components of full Dirac four component runs. The total density of these spinors can be obtained for interpretation purposes. Recognition that the spin-orbit coupling is a rotational operator ($L \cdot S$, where $L = R \text{ cross } P$) when it acts on an orbital can lead to chemical interpretability of the spin orbit results. See papers 40 and 41 below.

It is also possible to obtain the dipole transition moments between the final spin-mixed wavefunctions, which of course do not any longer have a rigorous S quantum no. When the run is SO-MCQDPT, the transition moment are first computed only between CAS states, and then combined with the spin-mixed SO-MCQDPT coefficients.

technical matters:

The only practical limitation on the computation of the Breit term is that HS02FF is limited to 10 active orbitals on 32 bit machines, and to about 26 active orbitals on 64 bit machines. The spin-orbit matrix elements vanish for $|\Delta S| > 1$, but it is possible to include three or more spins in the computation. Since singlets interact with triplets, and triplets interact with pentuplets, inclusion of $S=0,1,2$ simultaneously lets you pick up the indirect interaction between singlets and pentuplets that the intermediate triplets afford.

The choice between HS02 and HS02FF is very often in favor of the former. HS02 computes the matrix elements in CSF basis and then contracts them with CI coefficients, whereas HS02FF uses a generalized density in AO basis computed for each pair of states, thus HS02 is much more efficient in case of multiple states given in IROOTS. HS02FF takes less memory for integral storage, thus it can be superior in case of small active spaces and large basis sets, in part because it does not store 2e SOC integrals on disk and secondly, it does not redundantly treat the same pair of determinants if they appear in different CSFs. The numerical results with HS02 and HS02FF should be identical within machine and algorithmic accuracy.

Various symmetries are used to avoid computing zero spin-orbit matrix elements. NOSYM in \$TRANST allows some control over this: NOSYM=1 gives up point group symmetry completely, while 2 turns off additional symmetries such

as spin selection rules. HS01,2,2P compute all matrix elements in a group (i.e. between two \$DRTx groups with fixed Ms(ket)-Ms(bra)) if at least one of them does not vanish by symmetry, and HS02P actually avoids computation for each pair of states if forbidden by symmetry. Setting NOSYM=2 will cause HS02FF to consider the elements between two singlets, which are always calculated for HS01,2,2P when transition dipoles are requested as well.

SYMTOL has a dramatic effect on the run speed. This cutoff is applied to CSF coefficients, their products, and these products times CSF orbital overlaps. The value permits a tradeoff of accuracy for run time, and since the error in the spin-orbit properties approaches SYMTOL mainly for SOCI functions, it may be useful to increase SYMTOL to save time for CAS or FOCI functions. Some experimenting will tell you what you can get away with. SYMTOL is also used during CI state symmetry assignment, for NOIRR=-1 in \$DRT.

In case if you do not provide enough storage for the form factors sorting then some extra disk space will be used; the extra disk space can be eliminated if you set SAVDSK=.TRUE. (the amount of savings depends on the active space and memory provided, in some cases it can decrease the disk space up to one order of magnitude). The form factors are in binary format, and so can be transferred between computers only if they have compatible binary files. There is a built-in check for consistency of a restart file DAFL30 with the current run parameters.

input nitty-gritty

The transition moment and spin-orbit coupling driver is a rather restricted path through the GUGA CI part of GAMESS. Note that \$GUESS is not read, instead the MOs will be MOREAD in a \$VEC1 and perhaps a \$VEC2 group. It is not possible to reorder MOs. For S0-CI,

- 1) Give SCFTYP=NONE CITYP=GUGA MPLEVL=0.
- 2) \$CIINP is not read. The CI is hardwired to consist of CI DRT generation, integral transformation/sorting, Hamiltonian generation, and diagonalization. This means \$DRT1 (and maybe \$DRT2,...), \$TRANS, \$CISORT, \$GUGEM, and \$GUGDIA input is read, and acted upon.
- 3) The density matrices are not generated, and so no

properties (other than the transition moment or the spin-orbit coupling) are computed.

- 4) There is no restart capability provided, except for saving some form-factor information.
- 5) \$DRT1, \$DRT2, \$DRT3, ... must go from lowest to highest multiplicity.
- 6) IROOTS will determine the number of CI states in each CI for which the properties are calculated. Use NSTATE to specify the number of CI states for the CI Hamiltonian diagonalization. Sometimes the CI convergence is assisted by requesting more roots to be found in the diagonalization than you want to include in the property calculation.

For SO-MCQDPT, the steps are

- 1) Give SCFTYP=NONE CITYP=NONE MPLEVL=2.
- 2) the number of roots in each MCQDPT is controlled by \$TRANST's IROOTS, and each such calculation is defined by \$MCQD1, \$MCQD2, ... input. These must go from lowest multiplicity to highest.

references

The review already mentioned:

"Spin-orbit coupling in molecules: chemistry beyond the adiabatic approximation".

D.G.Fedorov, S.Koseki, M.W.Schmidt, M.S.Gordon,
Int.Rev.Phys.Chem. 22, 551-592(2003)

Reference for separate active orbital optimization:

1. B.H.Lengsfeld, III, J.A.Jafri, D.H.Phillips,
C.W.Bauschlicher, Jr. J.Chem.Phys. 74,6849-6856(1981)

References for transition moments:

- 2a. H.C.Longuet-Higgins
Proc.Roy.Soc.(London) A235, 537-543(1956)
- 2b. F.Weinhold, J.Chem.Phys. 54,1874-1881(1970)
3. C.W.Bauschlicher, S.R.Langhoff
Theoret.Chim.Acta 79:93-103(1991)
4. "Intermediate Quantum Mechanics, 3rd Ed." Hans A.
Bethe, Roman Jackiw Benjamin/Cummings Publishing,
Menlo Park, CA (1986), chapters 10 and 11.
5. S.Koseki, M.S.Gordon

J.Mol.Spectrosc. 123, 392-404(1987)

References for HS01 spin-orbit coupling, and Zeff values:

6. S.Koseki, M.W.Schmidt, M.S.Gordon
J.Phys.Chem. 96, 10768-10772 (1992)
7. S.Koseki, M.S.Gordon, M.W.Schmidt, N.Matsunaga
J.Phys.Chem. 99, 12764-12772 (1995)
8. N.Matsunaga, S.Koseki, M.S.Gordon
J.Chem.Phys. 104, 7988-7996 (1996)
9. S.Koseki, M.W.Schmidt, M.S.Gordon
J.Phys.Chem.A 102, 10430-10435 (1998)
10. S.Koseki, D.G.Fedorov, M.W.Schmidt, M.S.Gordon
J.Phys.Chem.A 105, 8262-8268 (2001)
11. S.Koseki, T.Matsushita, M.S.Gordon
J.Phys.Chem.A 110, 2560-2570(2006)

References for full Breit-Pauli spin-orbit coupling:

20. T.R.Furlani, H.F.King
J.Chem.Phys. 82, 5577-5583 (1985)
21. H.F.King, T.R.Furlani
J.Comput.Chem. 9, 771-778 (1988)
22. D.G.Fedorov, M.S.Gordon
J.Chem.Phys. 112, 5611-5623 (2000)

Paper 22 contains information on the HS02P partial two electron operator method.

Symmetry in spin-orbit coupling:

23. D.G.Fedorov, M.S.Gordon
ACS Symposium Series 828, pp 1-22(2002)

Reference for SO-MCQDPT:

25. D.G.Fedorov, J.P.Finley Phys.Rev.A 64, 042502 (2001)

Reference for Spin-Orbit with Model Core Potentials:

30. D.G.Fedorov, M.Klobukowski
Chem.Phys.Lett. 360, 223-228(2002)
31. T.Zeng, D.G.Fedorov, M.Klobukowski
J.Chem.Phys. 131, 124109/1-17(2009)
32. T.Zeng, D.G.Fedorov, M.Klobukowski
J.Chem.Phys. 132, 074102/1-15(2010)

The last two of these also discuss the 1st order Douglas-Kroll transformation of the 1e- part of the spin orbit operator.

Reference for properties, interpretations, and Spin-Orbit Natural Spinors:

40. T. Zeng, D. G. Fedorov, M. W. Schmidt, M. Klobukowski
J. Chem. Phys. 134, 214107/1-9(2011)
41. T. Zeng, D. G. Fedorov, M. W. Schmidt, M. Klobukowski

- J. Chem. Theor. Comput. 7, 2864-2875(2011)
 with an application being
 42. T.Zeng, D.G.Fedorov, M.W.Schmidt, M.Klobukowski
 J.Chem.Theo.Comput. 8, 3061-3071(2012).

Recent applications (see also 32,40,41):

50. S.P.Webb, M.S.Gordon
 J.Chem.Phys. 109, 919-927(1998)
 51. D.G.Fedorov, M.Evans, Y.Song, M.S.Gordon, C.Y.Ng
 J.Chem.Phys. 111, 6413-6421 (1999)
 52. D.G.Fedorov, M.S.Gordon, Y.Song, C.Y.Ng
 J.Chem.Phys. 115, 7393-7400 (2001)
 53. B.J.Duke J.Comput.Chem. 22, 1552-1556 (2001)
 54. C.M.Aikens, M.S.Gordon
 J.Phys.Chem.A 107, 104-114(2003)
 55. D.G.Fedorov, S.Koseki, M.W.Schmidt, M.S.Gordon
 K.Hirao and Y.Ishikawa (eds.) Recent Advances in
 Relativistic Molecular Theory, Vol. 5,
 (World Scientific, Singapore), 2004, pp 107-136.

* * *

Special thanks to Bob Cave and Dave Feller for their assistance in performing check spin-orbit coupling runs with the MELDF programs. Special thanks to Tom Furlani for contributing his 2e- spin-orbit code and answering many questions about its interface. Special thanks to Haruyuki Nakano for explaining the spin functions used in the MCQDPT package.

examples

We end with 2 examples. Note that you must know what you are doing with term symbols, J quantum numbers, point group symmetry, and so on in order to make skillful use of this part of the program. Seeing your final degeneracies turn out like a text book says it should is beautiful!

! Compute the splitting of the famous sodium D line.
 ! Joseph von Fraunhofer (Denkschriften der Koeniglichen
 ! Akademie der Wissenschf. zu Muenchen, 5, 193(1814-1815))
 ! observed the sun through good prisms, finding 700 lines,
 ! and named the brightest ones A, B, C... just in order.
 ! He was able to resolve the D line into two lines, which
 ! occur at 5895.940 and 5889.973 Angstroms. It would take
 ! a century to understand the D line is Na's 3s <-> 3p
 ! transition, and that spin-orbit coupling is what splits

! the D line into two. Charlotte Moore's Atomic Energy
 ! Levels, volume 1, gives the experimental 2-P interval
 ! as 17.1963, since the three relevant levels are at
 ! $2-S-1/2 = 0.0$, $2-P-1/2 = 16,956.183$, $2-P-3/2 = 16,973.379$.

1. generate ground state 2-S orbitals by conventional ROHF. the energy of the ground state is -161.8413919816

```
--- $contrl scftyp=rohf mult=2 $end
--- $system kdiag=3 memory=300000 $end
--- $guess guess=huckel $end
```

2. generate excited state 2-P orbitals, using a state-averaged SCF wavefunction to ensure radial degeneracy of the 3p shell is preserved. The open shell SCF energy is -161.7682895801. The computation is both spin and space restricted open shell SCF on the 2-P Russell-Saunders term. Starting orbitals are reordered orbitals from step 1.

```
--- $contrl scftyp=gvb mult=2 $end
--- $system kdiag=3 memory=300000 $end
--- $guess guess=moread norb=13
--- norder=1 iorder(6)=7,8,9,6 $end
--- $scf nco=5 nseto=1 no(1)=3 rstrct=.t. couple=.true.
--- f(1)= 1.0 0.166666666666667
--- alpha(1)= 2.0 0.333333333333333 0.0
--- beta(1)= -1.0 -0.166666666666667 0.0 $end
```

3. compute spin-orbit coupling in the 2-P term. The use of C1 symmetry in \$DRT1 ensures that all three spatial CSFs are kept in the CI function. In the preliminary CI, the spin function is just the alpha spin doublet, and all three roots should be degenerate, and furthermore equal to the GVB energy at step 2. The spin-orbit coupling code uses both doublet spin functions with each of the three spatial wavefunctions, so the spin-orbit Hamiltonian is a 6x6 matrix. The two lowest roots of the full 6x6 spin-orbit Hamiltonian are the doubly degenerate 2-P-1/2 level, while the other four roots are the degenerate 2-P-3/2 level.

```
$contrl scftyp=none cityp=guga runtyp=transitn mult=2 $end
$system memory=2000000 $end
$basis gbasis=n31 ngauss=6 $end
$gugdia nstate=3 $end
$transt operat=hsol numvec=1 numci=1 nfzc=5 nocc=8
iroots=3 zeff=10.04 $end
$drt1 group=c1 fors=.true. nfzc=5 nalp=1 nval=2 $end
```

```
$data
```

```
Na atom...2-P excited state...6-31G basis
```

```
Dnh 2
```

Na 11.0

\$end

--- GVB ORBITALS --- GENERATED AT 7:46:08 CST 30-MAY-1996

Na atom...2-P excited state

E(GVB)= -161.7682895801, E(NUC)= .0000000000, 5
ITERS

\$VEC1

1 1 9.97912679E-01 8.83038094E-03 0.00000000E+00...

... orbitals from step 2 go here ...

13 3-1.10674398E+00 0.00000000E+00 0.00000000E+00

\$END

As an example of both SO-MCQDPT, and the use of as much symmetry as possible, consider carbon. The CAS-CI uses an active space of 2s,2p,3s,3p orbitals, and the spin-orbit job includes all terms from the lowest configuration, 2s2,2p2. These terms are 3-P, 1-D, and 1-S. If you look at table 58 in Herzberg's book on electronic spectra, you will be able to see how the Kh spatial irreps P, D, S are partitioned into the D2h irreps input below.

! C SO-MRMP on all levels in the s**2,p**2 configuration.

!

levels	CAS	and	MCQDPT	
1	.0000		.0000	cm-1 3-P-0
2-4	12.6879-12.8469		13.2721-13.2722	3-P-1
5-9	37.8469-37.8470		39.5638-39.5639	3-P-2
10-14	12169.1275		10251.7910	1-D-2
15	19264.4221		21111.5130	1-S-0

!

! The active space consists of (2s,2p,3s,3p) with 4 e-.

! D2h symmetry speeds up the calculation considerably,

! on the same computer D2h = 78 and C1 = 424 seconds.

\$contrl scftyp=none cityp=none mplevl=2

runtyp=transitn \$end

\$system memory=5000000 \$end

!

! below is input to run in C1 subgroup

!

--- \$transt operat=hso2 numvec=-2 numci=2 nfzc=1 nocc=9

--- iroots(1)=6,3 parmp=3

--- ivex(1)=1,1 \$end

--- \$mrmp mrpt=mcqdpt rdvecs=.t. \$end

--- \$MCQD1 nosym=1 nstate=6 mult=1 iforb=3

--- nmofzc=1 nmodoc=0 nmoact=8

--- wstate(1)=1,1,1,1,1,1 thrcon=1e-8 thrngen=1e-10 \$END

--- \$MCQD2 nosym=1 nstate=3 mult=3 iforb=3

```

---          nmofzc=1 nmodoc=0 nmoact=8
---          wstate(1)=1,1,1 thrcon=1e-8 thrngen=1e-10 $END
!
!          below is input to run in D2h subgroup
!
$transt operat=hso2 numvec=-7 numci=7 nfzc=1 nocc=9
          iroots(1)=3,1,1,1, 1,1,1  parmp=3
          ivex(1)=1,1,1,1,1,1,1 $end
$mrmp mrpt=mcqdpt rdvecs=.t. $end
$MCQD1  nosym=-1 mult=1 iforb=3
          nmofzc=1 nmodoc=0 nmoact=8
          stsym=Ag wstate(1)=1,1,1 thrcon=1e-8 thrngen=1e-10 $END
$MCQD2  nosym=-1 mult=1 iforb=3
          nmofzc=1 nmodoc=0 nmoact=8
          stsym=B1g wstate(1)=1 thrcon=1e-8 thrngen=1e-10 $END
$MCQD3  nosym=-1 mult=1 iforb=3
          nmofzc=1 nmodoc=0 nmoact=8
          stsym=B2g wstate(1)=1 thrcon=1e-8 thrngen=1e-10 $END
$MCQD4  nosym=-1 mult=1 iforb=3
          nmofzc=1 nmodoc=0 nmoact=8
          stsym=B3g wstate(1)=1 thrcon=1e-8 thrngen=1e-10 $END
$MCQD5  nosym=-1 mult=3 iforb=3
          nmofzc=1 nmodoc=0 nmoact=8
          stsym=B1g wstate(1)=1 thrcon=1e-8 thrngen=1e-10 $END
$MCQD6  nosym=-1 mult=3 iforb=3
          nmofzc=1 nmodoc=0 nmoact=8
          stsym=B2g wstate(1)=1 thrcon=1e-8 thrngen=1e-10 $END
$MCQD7  nosym=-1 mult=3 iforb=3
          nmofzc=1 nmodoc=0 nmoact=8
          stsym=B3g wstate(1)=1 thrcon=1e-8 thrngen=1e-10 $END
!
!          input to prepare the 3-P ground state orbitals
!          great care is taken to create symmetry equivalent p's
!
--- $contrl scftyp=mcscf cityp=none mplevl=0
---          runtyp=energy mult=3 $end
--- $guess  guess=moread norb=55 purify=.t. $end
--- $mcscf  cistep=guga fullnr=.t. $end
--- $drt    group=c1 fors=.true.
---          nmcc=1 ndoc=1 nalp=2 nval=5 $end
--- $gugdia nstate=9 maxdia=1000 $end
--- $gugdm2 wstate(1)=1,1,1 $end
!
$data
C...aug-cc-pvtz (10s,5p,2d,1f) -> [4s,3p,2d,1f]
(1s,1p,1d,1f)
Dnh 2

```

S	8		
1		8236.000000	0.5310000000E-03
2		1235.000000	0.4108000000E-02
3		280.800000	0.2108700000E-01
4		79.27000000	0.8185300000E-01
5		25.59000000	0.2348170000
6		8.997000000	0.4344010000
7		3.319000000	0.3461290000
8		0.3643000000	-0.8983000000E-02
S	8		
1		8236.000000	-0.1130000000E-03
2		1235.000000	-0.8780000000E-03
3		280.800000	-0.4540000000E-02
4		79.27000000	-0.1813300000E-01
5		25.59000000	-0.5576000000E-01
6		8.997000000	-0.1268950000
7		3.319000000	-0.1703520000
8		0.3643000000	0.5986840000
S	1		
1		0.9059000000	1.0000000000
S	1		
1		0.1285000000	1.0000000000
P	3		
1		18.71000000	0.1403100000E-01
2		4.133000000	0.8686600000E-01
3		1.200000000	0.2902160000
P	1		
1		0.3827000000	1.0000000000
P	1		
1		0.1209000000	1.0000000000
D	1		
1		1.097000000	1.0000000000
D	1		
1		0.3180000000	1.0000000000
F	1		
1		0.7610000000	1.0000000000
S	1		
1		0.4402000000E-01	1.0000000000
P	1		
1		0.3569000000E-01	1.0000000000
D	1		
1		0.1000000000	1.0000000000
F	1		
1		0.2680000000	1.0000000000

\$end

--- OPTIMIZED MCSCF MO-S --- GENERATED 22-AUG-2000
 E(MCSCF)= -37.7282408589, 11 ITERS
 \$VEC1

1 1 9.75511467E-01 ...snipped...
\$END