

Input Manual for ACES II
Release 2.0

Quantum Theory Project
Departments of Chemistry and Physics
University of Florida
Gainesville, FL 32611

November 17, 1997

I. Introduction to The ACES II Program System

ACES II is a series of programs for performing *ab initio* calculations. The package has a high degree of flexibility, and supports many kinds of calculations at a number of levels of theory. The major strengths of the program are in performing calculations using “many-body” methods to treat electron correlation. These approaches, broadly categorized as many-body perturbation theory (MBPT) and the coupled-cluster (CC) approximation, offer a reliable treatment of correlation and have the attractive property of size-extensivity, meaning that energies scale properly with the size of the system. As a result of this property, MBPT and CC methods are ideally suited for the study of chemical reactions.

Two important features of the ACES II program system are its effective use of molecular symmetry, particularly in MBPT and CC calculations, and the sophisticated gradient methods which are included in the program. The implementation of symmetry currently is limited to D_{2h} and its subgroups, and the expected speedup due to symmetry utilization will usually be approximately equal to the square of the order of the point group used in the calculation for all steps except integral and integral derivative generation, and integral sorts, where the speedup can be no greater than the order of the group.

Gradient techniques are implemented for SCF and the following correlated levels of theory : MBPT(2), MBPT(3), MBPT(4), CCD, QCISD, CCSD, QCISD(T), and CCSD(T) for both restricted and unrestricted Hartree-Fock (RHF and UHF, respectively) reference functions. In addition, for the MBPT(2), MBPT(3), CCSD, and CCSD(T) methods, gradients are available for restricted open-shell Hartree-Fock (ROHF) reference functions. They are also available for certain CCSD calculations based on quasi-restricted Hartree-Fock (QRHF) reference functions, namely those for high-spin doublet cases and, more recently, two-determinant CCSD (TD-CCSD) calculations for open-shell singlet states, where the open-shell orbitals have different symmetries.

Efficient algorithms for geometry optimization and transition state location have also been included, and may be used at all levels of theory for which analytical gradients are available. Analytic second derivatives have been implemented for SCF using RHF, UHF and ROHF reference functions. In addition, analytically evaluated NMR chemical shift tensors are available at the SCF and MBPT(2) levels using gauge-including atomic orbitals (GIAOs) to ensure exact gauge-invariance. Other features include the direct calculation of electronic excitation energies using the Tamm-Dancoff (or configuration interaction singles) model, the random-phase approximation and the equation of motion coupled-cluster approach. Transition moments between ground and excited states can be calculated for all of the methods, as well as selected excited state properties.

The programs collectively known as ACES II began development in early 1990, and the first version of the code was written by J.F. Stanton, J. Gauss, J.D. Watts, W.J. Lauderdale and R.J. Bartlett. Program development is continuing, and the capabilities of the package are continually increasing. At present, there are approximately 20 modules, each of which performs a well-defined function and communicates with the rest of ACES II via a small number of files. All of the modules have been written explicitly for Cray supercomputers,

but perform efficiently on any vector computer or RISC architecture workstation. For the most time-consuming computational steps, essentially all of the most expensive steps have been coded as matrix multiplications using appropriate BLAS library routines. Hence, the CPU time for ACES II jobs is heavily dependent on the quality of the existing BLAS library.

Since ACES II is the product of academic research group, and not a software company, we are unable to guarantee that all results obtained with it are correct. Although we have made great progress in removing serious errors from the codes, problems may still occur and may be reported by electronic mail to aces2@qtp.ufl.edu. Any suggestions for improving the input or output, “wish-lists” for features, or other comments may also be communicated by electronic mail to aces2@qtp.ufl.edu.

For readers who are familiar with the first release of ACES II, we now list the principal new features in the second release :

- Equation of motion CCSD methods for excited states, including excitation energies, transition moment and excited state property evaluation. *Author : J.F. Stanton.*
- Gauge including atomic orbital SCF and MBPT(2) NMR chemical shift calculations. *Author : J. Gauss.*
- Complete coupled-cluster singles, doubles, and triples (CCSDT) energy calculations. *Author : J.D. Watts*
- Noniterative fifth-order triple and quadruple excitation methods (CCSD+TQ*(CCSD), CCSD(TQ), and QCISD(TQ)). *Author : J.D. Watts.*
- Analytical gradients for ROHF-CCSD(T). *Authors : J.D. Watts and J. Gauss.*
- Equation of motion CCSD method for electron attachment energies. *Author : M. Nooijen.*
- Hybrid Hartree-Fock Density Functional Theory methods. *Author : N. Oliphant.*
- Time-dependent Hartree-Fock calculations. *Authors : H. Sekino and W.J. Lauderdale.*
- Effective core potentials. *Authors : C. Huber and J. Gauss.*

Some features which are now available in the RJB group code and will be available in a future release are :

- TD-CCSD energies *Authors : P.G. Szalay and A. Balkova.*
- TD-CCSD analytical derivatives *Author : P.G. Szalay.*
- Dropped core and/or virtual orbitals in analytical derivative calculations for RHF, UHF, and ROHF references *Author : K.-K. Baeck.*

- Equation-of-motion CCSD calculation of dynamic polarizabilities (including partitioned scheme) *Authors : J.F. Stanton, S.A. Perera, and M. Nooijen.*
- Equation-of-motion CCSD calculation of NMR spin-spin coupling constants (including partitioned scheme) *Authors : S.A. Perera and M. Nooijen.*
- Partitioned equation-of-motion CCSD calculations of excitation energies *Authors : S.R. Gwaltney and M. Nooijen.*

II. Using this Manual

In principle, all aspects of preparing an ACES II input deck and running a job are covered here, as well as some tips on the construction of Z-matrices and judicious use of certain keywords. All new users of the program are encouraged to *browse* through the entire input manual. Following this, all sections which are of interest should be read in detail. Mandatory reading for new users is the section entitled “Common mistakes and sources of confusion”, which begins on page 50. Any errors detected in the manual should be reported to aces2@qtp.ufl.edu.

A useful supplement to this manual is a set of six or so laboratory exercises which have been used in ACES II workshops. Users interested in receiving this material may send email to aces2@qtp.ufl.edu.

III. Basic Aspects of Input Files for ACES II

A. Overview

Running ACES II calculations is straightforward. In most calculations two files are needed. First, the user must construct an input file, which must be copied in to the user's workspace as ZMAT. Second, the basis set library file GENBAS must also be copied in to the workspace. The program is invoked simply by the command "xaces2 ", and the user must direct output to an output file. For example, a simple script for execution on one of the QTP IBM RS6000 workstations is as follows :

```
#!/bin/ksh
. ./kshrc
mkdir      /scr/whirl_2/tmp/tmp.$$
cd         /scr/whirl_2/tmp/tmp.$$
cp        ~watts/GENBAS                GENBAS
cp        /camp/whirl_1/qtp/rjb/jw/jobs/c2h5.zmat  ZMAT
xaces2 > /camp/whirl_1/qtp/rjb/jw/jobs/c2h5.out
rm *
cd $HOME
rmdir     /scr/whirl_2/tmp/tmp.$$
exit
```

The first two lines reflect our local kornshell environment, and users will have to tailor this to their local environment. In particular, the second line sets the appropriate path to the ACES II executables. The next two lines are also site specific and are to create and change to a temporary directory on a scratch disk. The next three lines are the essentials of running an ACES II job. The basis set file and the input file are copied to the working directory, and the calculation is executed with the xaces2 command. Output is directed to a file on a permanent disk partition. In order for the calculation to run just by invoking xaces2, the path to the ACES II executables must be appropriately included in the user's .kshrc (or .cshrc or other environment setting files). There are a number of variations on this basic theme. For example, the user may on occasion need to copy additional files to the workspace (for example, FCMINT, GUESS, OLDMOS) prior to execution. Similarly, the user may wish to copy out certain files created by ACES II to permanent storage (for example, FCMINT, JOBARC, JAINDX, NEWMOS). The final four lines clear the workspace and remove the temporary directory, thus freeing scratch space for the next job or for other users.

The next section describes the ZMAT file in depth. This file contains the information about the molecule and calculation. This information includes the geometry of the molecule (specified by either a Z-matrix or Cartesian coordinates), the choice of basis set, the type of calculation to run, and several other parameters which are used by the program. The input is keyword-driven, and usually only a few keywords must be specified since intelligent defaults exist for all keywords. Several basis sets are available for immediate use. These include

the standard Dunning basis sets, most of those used in the GAUSSIAN program system, and the new correlation-consistent basis sets of Dunning and coworkers, the polarized basis sets of Sadlej, and the atomic natural orbital basis sets of Widmark, Malmqvist, and Roos. In addition, the user may create his or her own basis set for use by the program. Several different calculation types can be performed at different levels. For example, single point, geometry optimization, property evaluation, and vibrational frequency calculations may be performed. Also many levels of calculation such as SCF, MBPT, CCSD and CI methods are currently available.

A description of the ZMAT file is included in the next section. A second file, the GENBAS file, will be described later. The GENBAS file contains complete specifications of the “standard” basis sets currently available to the user. However, the format of the GENBAS file is very simple, and it is a straightforward matter for the user to add additional basis sets to the library. During the course of the calculation, several additional files are created but on the whole it is not necessary for the user to be familiar with their content. Some exceptions will be discussed in later sections of this manual.

A description of the other parameters that can be adjusted via the ZMAT file is included in the ACES2 namelist section. These parameters can be used to set the type and level of calculation as well as many other options ranging from the SCF occupation to the convergence criterion for geometry optimizations. Although a large number of options are available, only a small subset of them will be needed in most calculations.

A set of test cases, both the ZMAT input files and the corresponding output files, is provided which, along with testing the various modules, provides a sample ZMAT for many different calculation types.

B. The ZMAT File

The ZMAT file contains up to four sections. The first contains the title and either the “Z-matrix” or Cartesian coordinates for the system to be studied. The second, omitted if Cartesian coordinates are used, is the specification of values for the internal coordinates used in the Z-matrix. The third section comprises the ACES2 namelist, which specifies the program options such as type of calculation, basis set, etc. The final section is used for nonstandard basis set specification and is omitted if “standard” basis sets are to be used. The individual sections are discussed separately below. We also discuss the conventions used by ACES II for the orientations of the coordinate axes in the different point groups. This information is needed to specify the irreducible representations of orbitals correctly. Finally, some sample ZMAT files are presented and described. It should be stressed that the following is a *complete* description of the various input options and as such may seem unnecessarily complicated to the general user. However, the vast majority of the variables in the namelist need never be specified since they have adequate defaults, and in practice the user will only need to specify three or four items, or even less, except in special cases. This will further be illustrated by the five sample ZMAT files presented in section B..10. A general remark pertinent to all sections of the ZMAT file is that all characters should be *upper case*. *Another very important point is that the ZMAT file must terminate with a blank line.*

B..1 Z-Matrix

Provided no nonstandard file declarations are used (see Sec. B..8), the first line of the ZMAT file is reserved for the job title, which is communicated to the rest of ACES II and printed out in a number of places during the execution of the program. Following this is the internal coordinate specification for the molecule, using a Z-matrix. Each line of the Z-matrix specifies the position of an individual atom relative to three previously specified atoms. The specification includes a length, a bond angle, and a dihedral angle. The number associated with each atom is governed by its position in the Z-matrix. The essentials of Z-matrix construction may be illustrated by considering a Z-matrix for a system of four atoms ABCD, not linked in any particular order.

```
Arbitrary ABCD molecule
A
B 1 RAB
C 1 RAC 2 CAB
D 3 RCD 2 DCB 1 TAU
```

The first line in the Z-matrix contains just the atomic symbol of one of the atoms, say A. The second line specifies the position of a second atom, say B, relative to the first atom (A). Suppose that B is a distance RAB from A. The second line then contains the atomic symbol B, followed by the number 1 (A is atom number 1), and a parameter label, RAB, (“B 1 RAB”). For the specification of the third atom, a distance and an angle are needed.

We may use the distance between atoms A and C and the angle CAB or we may use the distance between atoms B and C and the angle CBA. In the first case the third line would have the form “C 1 RAC 2 CAB”, while in the second case it would have the form “C 2 RBC 1 CBA”. Finally, there is a line specifying the position of D relative to the other atoms. This line must contain a distance, a bond angle, and a dihedral angle, and might have the form “D 3 RCD 2 DCB 1 TAU”, TAU being the angle between the BCD and ABC planes. For a system with more than four atoms, the fifth and subsequent lines follow the same pattern as the fourth line of the example given above, i.e. they also contain a length, angle, and dihedral angle and numbers of three previously specified centers. It should be emphasized that this is a somewhat simplified description which would work for a tetraatomic molecule such as hydrogen peroxide, but would not be satisfactory for acetylene. For the latter “dummy” atoms are necessary. These and several other tips for forming Z-matrices are discussed below. Several examples of Z-matrices are given in this manual to assist users who are unfamiliar with the concept.

A more formal description of a line in the Z-matrix input is as follows. Each line may have as many as seven entries. We consider the Ith line. The contents are the Ith element of the ZSYM array, the 3*Ith, (3*I-1)th, and (3*I-2)th elements of the NCON and PARNAM arrays. The ZSYM array is of length N, where N is the number of lines in the Z-matrix (this includes those for any dummy and ghost atoms), and contains the chemical symbols of all the atoms in the Z-matrix. The NCON and PARNAM arrays are of length 3*N. NCON contains the numbers of atoms relative to which each atom is specified. The PARNAM array contains the names of the lengths, angles, and dihedral angles contained in the Z-matrix. Positions 3*I-2 are for lengths, 3*I-1 for angles, and 3*I are for dihedral angles (I=1,2,...,N). It should be clear that elements 1,2,3,5,6, and 9 of NCON and PARNAM are not defined. The Ith line (I=1,2,...,N) then has the general form:

ZSYM The atomic symbol of the atom (dummy atom is ‘X’).

NCON(3*I-2) The number of the atomic center to which the atom is “bonded”. (Only for I>1).

PARNAM(3*I-2) A character variable name corresponding to the bond distance between atom NCON(3*I-2) and atom I.

NCON(3*I-1) The third member of the triangle formed by atoms I and NCON(3*I-2). (Only for I>2).

PARNAM(3*I-1) A character variable corresponding to the associated bond angle.

NCON(3*I) The fourth member of the dihedral angle formed by atoms I, NCON(3*I-2) and NCON(3*I-1). (Only for I>3).

PARNAM(3*I) A character variable name corresponding to a dihedral angle determined as follows: In the plane perpendicular to the NCON(3*I-2)↔NCON(3*I-1) axis, the angle is that needed to rotate the projection of the [I←NCON(3*I-2)] vector into

the projection of the $[\text{NCON}(3^*I) \leftarrow \text{NCON}(3^*I-1)]$ vector. Clockwise is taken to be positive. Values restricted from -180 to 180 degrees.

All variable names (PARNAM array) are *limited to three characters (or perhaps it is five now)*. An asterisk (*) immediately after the variable name specifies that that particular internal coordinate is to be optimized. *One and only one space must separate the different fields on each Z-matrix line.*

B..2 Z-Matrix Parameter Input

After a blank line following the Z-matrix, the values of all *unique* internal coordinates (those which have different names) are specified as follows,

$$\text{PNM}=\text{Value}$$

where PNM is one of the unique members of the PARNAM array (see above), and Value is the value assigned to that coordinate. The parameters may be specified in any order. Angles must be entered in degrees, distances must be in Ångströms. Bond angles of 0 and 180 degrees are not allowed since these lead to a singularity in the transformation between Cartesian and internal coordinates. This of course does not mean that ACES II is unable to handle linear molecules such as carbon dioxide. However, for molecules such as CO₂, dummy atoms must be used in the ZMAT file to avoid problematic bond angles.

To facilitate construction of Z-matrices for highly symmetric molecules, certain variable names have been reserved for specific values. To use these parameters, which are listed below, the user still must specify a value in the parameter input section, but it need not be correct since it will be converted to the exact value internally.

TDA Specifies the tetrahedral angle, $\text{Cos}^{-1}\left(\frac{-1}{3}\right)$ ($= 109.471 \dots$).

IHA Specifies the angle $\text{Cos}^{-1}\left(\frac{1}{5^{1/2}}\right)$ ($=63.4349 \dots$), which is useful for icosahedral molecules.

As an example, a Z-matrix and internal coordinate specification is given below for the formaldehyde molecule, in which only the (equivalent) C-H distances are to be optimized.

Sample formaldehyde Z-matrix input

```
O
C 1 R
H 2 R2* 1 A
H 2 R2* 1 A 3 T
```

```
R=1.22
R2=1.03
T=180.
A=122.
```

B.3 The Z-matrix Analyzer

A unique feature of ACES II is the Z-matrix analyzer, which is capable of detecting subtle and obvious deficiencies in the definition of internal coordinates. This is particularly important for geometry optimizations, in which the construction of the Z-matrix and the choice of parameters to be optimized are of vital importance. The analyzer inspects the internal coordinates, and carries out a number of checks. These include determination of whether coordinates given the same name are actually equivalent, if coordinates having different names are equivalent, whether a non-zero gradient is possible with respect to modes which are not being optimized, etc. In addition, it determines the number of degrees of freedom within the totally symmetric subspaces of nuclear configurations, and compares this value with the number of independent coordinates which are being optimized. If they are not equal, a warning message is printed out.

For most Z-matrices with poorly defined internal coordinates, the analyzer prints out a number of warning messages, but does not halt the ACES II execution sequence. However, for Z-matrices which are particularly bad, it will terminate your job. All users are encouraged to carefully inspect the output of the analyzer, and also to check that the full molecular point group (printed out below the output from the analyzer) is the one they expect. If warning messages are printed out, or the symmetry of the molecule specified by the Z-matrix is not what was intended, reconstruction is necessary.

A rule of thumb for Z-matrix construction is that each internal coordinate included in the Z-matrix must be accompanied by all others which are equivalent to it by the symmetry of the molecule. For example, in water, it is best to specify the molecule by the two O-H distances and the H-O-H bond angle, rather than by the O-H distance, the H-H distance, and the H-H-O angle. Although most users would definitely use the first Z-matrix, the idea of using just chemical bonds as internuclear distances can be dangerous. For example, in a regular hexagonal ring, a little reflection will show that one cannot include *all six* intervertex distances in the Z-matrix. If only five are specified, such as in the Z-matrix below (the 1-6 distance is missing), the internal coordinate gradient cannot have the full symmetry of the molecule, and the first step of a geometry optimization will break the molecular symmetry. This results in extremely slow convergence and significantly increased CPU time due to the reduced molecular symmetry.

```
A poor Z-matrix for hexagonal C6
```

```
C
```

```
C 1 R
```

```
C 2 R 1 A
```

```
C 3 R 2 A 1 T
```

```
C 4 R 3 A 2 T
```

```
C 5 R 4 A 3 T
```

```
R=1.33
```

```
A=120.
```

T=0.

In these situations, it is always best to use “dummy atoms”, which is simply a way to use a point in space as a reference for the coordinates. A dummy atom is denoted by the symbol “X”, and one or more is needed in essentially all Z-matrices for molecules with high symmetry. As an example of their use, a “good” Z-matrix for the C₆ ring is shown below. Don’t be afraid to use dummy atoms!

A better Z-matrix for hexagonal C₆

```
X
X 1 RX
C 2 R 1 A
C 2 R 1 A 3 T
C 2 R 1 A 4 T
C 2 R 1 A 5 T
C 2 R 1 A 6 T
C 2 R 1 A 7 T
```

RX=1.0

R=1.33

A=90.

T=60.

B.4 Cartesian coordinate geometry specification

In addition to Z-matrix geometry specification, a facility exists for geometry specification by Cartesian coordinates. This is of limited value since only single point calculations can be performed with this form of input. The format is straightforward. The first line is a title line, as for Z-matrix input. Following this there is a line for each atom, containing the atomic symbol and the values of the Cartesian coordinates in free format. The coordinates may be given in either atomic or Ångström units. For example, the formaldehyde geometry specified in the Z-matrix on page 5 could be specified with Cartesian coordinates (in Ångströms) as follows.

Formaldehyde using cartesian coordinates

```
O 0.0      0.0      1.22
C 0.0      0.0      0.0
H 0.0      0.873489539  0.545816842
H 0.0     -0.873489539 -0.545816842
```

The ACES2 command line must contain COORDINATE=CARTESIAN for this to work. Otherwise the program will try to process the above coordinates as though they are a Z-matrix. If the cartesian coordinates are specified in atomic units, this can be accomplished by using the keyword UNITS=BOHR in the ACES2 command line which will be described later.

B..5 Ghost atoms

Ghost atoms, which are specified by the symbol GH, have zero nuclear charge. However, while dummy atoms are only used to simplify the Z-matrix input, ghost atoms serve as a center for basis functions. This feature is particularly useful for calculations performed to determine the basis set superposition error (BSSE), and has several other applications, such as describing “lone pair” electrons of a molecule by functions which are not centered at any of the molecular nuclei. Symmetry can be used in such calculations but is in all cases restricted to the symmetry of the “supermolecule” comprised of the real and ghost atoms. The additional basis functions do not necessarily form a complete set of symmetry adapted functions within the point group of the considered molecule. This is different from the use of dummy atoms, which do not affect the symmetry of the calculation.

Note that currently only single point energy calculations are possible with ghost atoms. Also, the basis set has to be supplied in all cases explicitly via the BASIS=SPECIAL feature which is explained in another section.

B.6 The ACES2 Namelist Job Control Parameters

The job control parameters in the ACES2 namelist define the currently available options. These options are set through the use of keywords. In some cases the value for a keyword can be specified by an integer or by a character string. In our opinion the latter is preferable as it makes the input file more readable.

All possible keywords in the ACES2 namelist are discussed below. As there are a lot of keywords (100 or so), we have grouped them according to the type of calculation they affect. Keywords of **General Relevance** :

The first group of keywords are of general relevance, and include such fundamental parameters as the memory and type of coordinates. Others in this group are rather obscure. They are listed alphabetically. The most commonly used ones are CALCLEVEL, COORDINATES, MEMORY_SIZE, PRINT, and UNITS.

CACHE_RECS The number of records held in the i/o cache used by the post-SCF programs. The maximum number of records which can be held is 100. (Default : 10)

CALCLEVEL Defines the level of calculation to be performed. =0 SCF; =1 MBPT[2]; =2 MBPT[3]; =3 SDQ-MBPT[4]; =4 MBPT[4]; =5 LCCD; =6 LCCSD; =7 UCCSD[4]; =8 CCD; =9 UCC[4]; =10 CCSD; =11 CCSD + T[CCSD]; =12 CCSD + TQ*; =13 CCSDT-1; =14 CCSDT-1b; =15 CCSDT-2; =16 CCSDT-3; =17 CCSDT-4; =18 CCSDT; =19 LCCSDT (NCI); =20 CCD+ST[CCD] (NCI); =21 QCISD[T]; =22 CCSD[T]; =23 QCISD; =24 CID; =25 CISD; =26 QCISD[TQ]; =27 CCSD[TQ]; =28 CCSD+TQ (NCI); =29 CCSDT+Q* (NCI); =30 CCSDT+Q (NCI); =41 HFDFT. Calculation levels with NCI after them are not currently implemented but will be available in future versions of the program. For CALC=HFDFT note that there are several functionals available, which are specified by the FUNCTIONAL keyword. (Default : SCF).

COORDINATES The keyword INTERNAL(=0) means that the geometry is supplied in the usual Z-matrix format, while CARTESIAN(=1) means that the geometry is given in cartesian coordinates. Note that geometry optimizations are only possible with internal coordinates. (Default : INTERNAL)

FILE_RECSIZ This specifies the physical length (in integer words) of the records used in the word-addressable direct access files used by ACES II. This value should always be chosen as a multiple of 512 bytes, as your local system manager certainly understands. (Default : 2048).

HBAR This keyword determines which action is taken by the linear response program. ON (= 1) the full effective Hamiltonian is calculated and written to disk; OFF (= 0) the "lambda" linear response equations are solved. (Default : 0)

INCORE This keyword can be used to significantly reduce disk i/o, and should be implemented very soon. The following options are available: OFF (= 0), no special

algorithms are used (the default case); ALL (=1) all quantities except the $\langle ab||cd \rangle$ molecular integral lists are held in core; PARTIAL (= 2), the T2 and T1 vectors are held in core throughout the calculation; (=4) all quantities except the $\langle ab||cd \rangle$ and $\langle ab||ci \rangle$ integrals are held in core; (=5) $\langle ij||kl \rangle$ and $\langle ij||ka \rangle$ and two-index quantities are held in core; (=6) *all direct access files (MOINTS, GAMLAM, etc.) are held in core. At present, these options have been implemented only in the energy code (xvcc) and the excitation energy code (xvee).* (Default : 0)

INTGRL_TOL Tolerance for storage of two-electron integrals. If this has value N , integrals having absolute value greater than 10^{-N} will be stored on disk. (Default : 14).

JODA_PRINT Controls amount of debug printing performed by xjoda. The higher the number, the more information is printed. Values of 25 or higher generally do not produce anything of interest to the general user. Do not set JODA_PRINT to 999 as this will cause the core vector to be dumped to disk. (Default: 0; Value must be specified as an integer).

MEMORY_SIZE Specifies the total core memory used, in units of *integer* words. (Default : 6 500 000).

PRINT Controls the amount of printing in the energy and energy derivative calculation programs. Using a value of 1 will produce a modest amount of additional output over the default value of 0, which includes some useful information such as SCF eigenvectors. (Default : 0; Value must be specified as an integer).

SAVE_FILES Switch which tells ACES II whether to delete large files (AO integrals and MOINTS file for now) when they are no longer needed. OFF (= 0) They will not be saved, ON (= 1) they will be saved. (Default : 0). *This must currently be set to ON in NMR chemical shift calculations.*

SOLVENT This keyword allows a simple simulation of solvent effects. If SOLVENT is set to an integer N a dielectric constant of N is used to determine the orbitals. A cavity size may also be specified by creating a file "radius" which is read by the SCF code. This contains the cavity radius in Å. If the file is not present, the program uses a value calculated from $0.5 \text{ Å} + \text{half the longest internuclear distance}$. (Default : 0 — i.e. gas phase).

SUBGROUP This keyword allows the user to specify a specific Abelian subgroup to be used in a calculation. Acceptable arguments are DEFAULT (=0); C1 (= 1); C2 (= 2); CS (= 3); CI (= 4); C2V (= 5); C2H (= 6); D2 (= 7) and D2H (= 8). Use of C1 is of course equivalent to setting SYMMETRY=OFF in the input. The DEFAULT option (which is the default) uses the highest order Abelian subgroup (Default : DEFAULT).

SUBGRPAXIS This is a somewhat complicated keyword to use. Allowed values are the integers 1, 2 and 3, which specify the x, y and z axes, respectively. The meaning

of the keyword is best described by example : Suppose one is running a calculation on water, and wishes to run it in the C_s point group with the “special” plane being the one which bisects the H–O–H bond angle. Now, what SUBGRPAXIS does is to specify which Cartesian direction in the C_{2v} frame becomes the special direction in the C_s frame. ACES II will orient water in the yz plane, so one wants the y axis in the C_{2v} frame to be the z axis in the C_s frame. Hence, for this case, one would specify SUBGRPAXIS=2. Use of this keyword may be facilitated by studying section D1 of this chapter, entitled “Molecular Orientation”. However, when the true Abelian subgroup is either C_{2v} or D_{2h} , the ACES II orientation is not well defined, and it may be necessary to run the XJODA executable directly two times. If SUBGROUP=0 in the first pass, then the reference orientation for the true Abelian subgroup can be determined and the appropriate value of SUBGRPAXIS selected.

SYMMETRY Specifies what subgroup of the full point group is to be used in the energy and/or gradient calculation (The “computational” point group). OFF (=1) forces a no symmetry run (in C_1), ON (=0) runs the calculation in the largest self-adjoint subgroup (D_{2h} and its subgroups), and FULL (=2) uses the full point group. Currently, ACES II does not support groups with degenerate representations, so the FULL option has no value unless Joda is being used to make input decks for another program package. Symmetry is turned off in finite difference vibrational frequency calculations which use the FINDIF_OLD algorithm (see keyword VIBRATION). However, the algorithm which determines the number of gradients to be evaluated and performs the resulting construction of the force constant and dipole derivative matrices uses the *full* point group symmetry. (Default: ON unless VIBRATION=FINDIF_OLD, which forces SYMMETRY=OFF). *NOTE: In the vast majority of cases, the general user should not need to use this keyword. Exceptions include error detection (for example, to see if an incorrect result or a program crash occurs with and without symmetry), and some calculations in the presence of finite fields.*

UNITS Specifies the units used for molecular geometries using the Cartesian coordinate format (see entry for COORDINATES). ANGSTROM (= 0) uses Ångström units, BOHR (= 1) specifies atomic units. (Default : ANGSTROM)

Keywords relevant to **Basis Set Specification** :

BASIS Specifies the name of the basis set for all atoms in the system under study. Accepted values are: STO-3G, DZ, 3-21G, 4-31G, 6-31G, 6-311G, DZP, TZP, TZ2P, 6-31G*, 6-31G**, 6-311G*, 6-311G**, PVDZ, PVTZ, PVQZ, PV5Z, PBS and WMR. For hydrogen atoms, the Pople “*” and “**” basis sets default to the corresponding basis with no polarization functions and *p* functions, respectively. In order for this to work, the appropriate basis sets must be in the GENBAS file for all atoms. If you want to use different basis sets on different atoms, to use ghost atoms, or to use a basis which is not in the above list (there are several such families in the standard GENBAS file), the section entitled “Non-Standard Basis Set Specification” should be consulted. Non-standard basis set specification involves setting the BASIS keyword to SPECIAL (equivalently, it may be set to 0 or omitted). A complete list of the basis sets currently in the standard GENBAS file is given in section A of Part IV of this manual. (Default: SPECIAL)

CONTRACTION The contraction scheme used by the integral and integral derivative programs. SEGMENTED (= 0) uses a segmented contraction scheme; GENERAL (= 1) uses a general contraction scheme. *NOTE: Even for truly segmented basis sets, both programs run significantly faster in GENERAL mode, and this should be used in practice.* (Default: GENERAL).

ECP This specifies whether effective core potentials (some kind of pseudopotentials) are used (ON = 1) or not (OFF = 0). ECP=ON requires BASIS=SPECIAL and specification of the ECP (see sections E.1 and B.7 for further explanation). (Default : OFF)

GENBAS_1 This keyword applies only to Hydrogen and Helium atoms and specifies the number of contracted Gaussian functions per shell. There is usually no need to use this keyword, but it can be useful for using a subset of the functions in a particular entry in the GENBAS file, particularly for generally contracted basis sets. For example, if entry H:BASIS in the GENBAS file contains 7 contracted *s* functions, 4 *p* functions and a single *d* function, then setting GENBAS_1=730 would eliminate the last *p* function and the *d* function. The default for this keyword is to use the unaltered GENBAS entry.

GENBAS_2 This keyword performs the same function as GENBAS_1 above, but applies only to second-row atoms.

GENBAS_3 This keyword performs the same function as GENBAS_1 and GENBAS_2, but applies only to third-row atoms.

SPHERICAL This keyword specifies whether spherical harmonic (5d, 7f, 9g, etc.) or Cartesian (6d, 10f, 15g, etc.) basis functions are to be used. ON (= 1) uses spherical harmonics, OFF (= 0) uses Cartesians. The latter must be used for correlated gradient calculations at present. (Default : OFF).

Keywords relevant to **Self-Consistent Field Calculations** :

CHARGE The molecular charge. (Default : 0).

DAMPSCF Specifies the static damping factor used in the SCF iterations. =N The static damping factor is N x 0.01. This keyword is only operative when DAMP_TYP=STATIC (Default : 20).

DAMP_TOL Specifies the cutoff for Davidson damping (see DAMP_TYP). If this is N, the cutoff is N x 0.1. Default : 1.

DAMP_TYP Specifies what type of damping is used during the SCF iterations. The choices are NONE (no damping); DAVIDSON (use Davidson's empirical dynamical damping scheme); STATIC (use a fixed damping parameter, the value of which is specified by DAMPSCF). Note that RPP convergence extrapolation is not turned on until after the damp factor has gone below DAMP_TOL and the energy change is below 0.05 a.u. DAMP_TYP=DAVIDSON is recommended. Default : NONE.

FOCK This option is used to control the algorithm used for construction of the Fock matrix in SCF calculations. PK (= 0) uses the PK-supermatrix approach while AO (= 1) constructs the matrix directly from the basis function integrals. In general, PK is faster, but results in *considerable* use of disk space when out-of-core algorithms are required. (Default : PK).

GUESS Where the initial SCF eigenvectors are read from. MOREAD means to read from the disk (the "JOBARC" file) and CORE means to use a core Hamiltonian initial guess. If MOREAD is chosen but no disk file is present, the core Hamiltonian is used. (Default : MOREAD)

LINDEP_TOL The tolerance for basis set linear dependence. =N The basis set is considered linearly dependent and eigenvectors of the overlap matrix are neglected if the associated eigenvalues are less than 10^{-N} . (Default: 5).

LOCK_ORBOCC This keyword is used by the SCF program to determine if the orbital occupancy (by symmetry block) is allowed to change in the course of the calculation. ON (= 1) locks the occupation to that set by the keyword OCCUPATION [or the initial guess if OCCUPATION is omitted]; OFF (= 0) permits the occupation to change. (Default : 1 if the occupation is specified with the OCCUPATION keyword and in second and later steps of optimizations ; 0 if OCCUPATION is omitted.)

LSHF_A1 Sets the doubles-singles level shifting parameter α . If this is set to N, α is N x 0.01. This keyword is currently only meaningful in ROHF calculations. Default : 0.

LSHF_B1 Sets the singles-virtuals level shifting parameter β . If this is set to N, α is N x 0.01. This keyword is currently only meaningful in ROHF calculations. Default : 0.

MULTIPLICITY The spin multiplicity ($2S+1$). (Default : 1)

NEWVRT Specifies if the usual virtual orbitals are to be replaced by a set determined from an $N - 1$ potential. This is an orthogonal transformation within the virtual space and so as long as appropriate f_{ij} and f_{ab} are included, the energies of standard single reference methods are unchanged. However, for TD-CC methods this keyword mixes one of the occupied orbitals with the virtual space, and so the results are changed. It is anticipated that TD-CCSD calculations will be improved by the use of this keyword, but this has not yet been demonstrated. NEWVRT is useful for interpreting results of EOM-EE calculations, as excitations tend to be more easily identified as involving one occupied and one virtual orbital. =OFF (do not rotate the virtual space); =ON (rotate the virtual space). (Default : OFF).

OCCUPATION Specifies the orbital occupancy of the reference function in terms of the occupation numbers of the orbitals and their irreducible representations. The occupancy is specified by either NIRREP or 2*NIRREP integers specifying the number of occupied orbitals of each symmetry type, where NIRREP is the number of irreducible representations in the computational point group. If there are no orbitals of a particular symmetry type a zero must be entered. If the reference function is for an open-shell system, two strings of NIRREP occupation numbers separated by a slash (‘/’) are input for the α and β sets of orbitals. An example of the use of the OCCUPATION keyword for the water molecule would be OCCUPATION=3-1-1-0. For the 2A_1 water cation, an open-shell system, the keyword would be specified by OCCUPATION=3-1-1-0/2-1-1-0. It should be noted that the VMOL integral program orders the irreducible representations in a strange way which most users do not perceive to be a logical order. Hence, it is usually advisable initially to run just a single point integral and SCF calculation in order to determine the number and ordering of the irreducible representations. The occupation keyword may be omitted, in which case an initial orbital occupancy is determined by diagonalization of the core Hamiltonian. In many cases, SCF calculations run with the core Hamiltonian guess will converge to the lowest energy SCF solution, but this should not be blindly assumed. (Default : The occupation is given by the core Hamiltonian initial guess).

REFERENCE The type of SCF calculation to be performed. RHF (= 0) restricted Hartree-Fock reference; UHF (= 1) unrestricted Hartree-Fock reference=1; ROHF (= 2) restricted open-shell Hartree-Fock calculation. For the RJB group the following option for REFERENCE is also available : TWODET (=3) two-determinant reference for open-shell singlet CC calculations. *Note : REFERENCE=TWODET is not supported in Release 2.* Note also that when REFERENCE=TWODET the orbitals are obtained for a closed-shell state, and so are *not* optimum open-shell singlet (or low spin triplet) SCF orbitals. The orbital occupancy for subsequent CCSD calculations is specified by the various QRHF keywords. (Default : 0).

RPP Specifies whether or not the reduced partitioning procedure is to be used to accelerate

convergence of the SCF equations. =0 Do not use RPP, =1 use RPP. (Default : 1).
Note : This keyword name has been changed to SCF_EXTRAPO.

RPP_LATEST Sets the latest iteration for initiation of the RPP convergence acceleration procedure in SCF calculations. RPP is switched on when the error falls below a certain threshold, but in difficult cases where the iterations are oscillatory it is necessary to force it on when the error is still large. In such a case, the RPP will begin on the iteration number specified by this parameter. (Default : 15). *Note : This keyword has been changed to SCF_EXPSTAR.*

RPP_ORDER Specifies the number of density matrices to be used in the RPP convergence acceleration procedure. =N N density matrices will be used. (Default : 6). *Note : This keyword has been changed to SCF_EXPORDE.*

SCF_CONV The convergence criterion for the SCF equations. Equations are considered converged when the maximum change in density matrix elements is less than 10^{-N} . (Default : 7).

SCF_MAXCYC Specifies the maximum number of SCF iterations. (Default : 150).

SYM_CHECK This specifies the action taken when the density matrix does not transform as the totally symmetric irreducible representation of the full molecular point group. NORMAL (=0) terminates the run if the molecule is nonlinear, while OVERRIDE (=1) allows the job to continue but prints a warning message. Nonsymmetric density matrices result from calculations on electronically degenerate states or from broken-symmetry SCF solutions. Often, such calculations do not give meaningful results and inexperienced users are encouraged to use SYM_CHECK=NORMAL in their calculations on nonlinear molecules. For Π , Δ , Φ , etc. states of linear molecules, however, meaningful calculations can still be performed even though the density matrix is not symmetric. (Default : OVERRIDE)

Keywords relevant to **Density Functional Theory Calculations** :

- CALC must be set to HFDFIT

FUNCTIONAL This keyword only has meaning when CALC=HFDFIT and a calculation type other than SINGLE_POINT has been requested. Possible values are : 0 XALPHA; 1 BECKE; 2 LYP; 3 XLYP; 4 BLYP (the default). The ACES II HFDFIT program always calculates all of the above energies, but only one functional can be used in, for example, (numerical) geometry optimizations. This keyword specifies which one is to be used.

Keywords relevant to **Hartree-Fock stability analysis** :

HFSTABILITY This is used to control checks of the stability of RHF and UHF wavefunctions, as well as a cursory search for a lower SCF solution. There are three possible options for this keyword. OFF (=0) does nothing, while ON (=1) performs a stability check and returns the number of negative eigenvalues in the orbital rotation Hessian. A third option, FOLLOW (=2) performs the stability check and then proceeds to rotate the SCF orbitals in the direction of a particular negative eigenvalue of the orbital rotation Hessian (see the explanation of keyword ROT_EVEC), after which the SCF is rerun. A more detailed discussion of HF stability testing may be found in the section entitled Hartree-Fock Wavefunction Stability Analysis, below. (Default : OFF)

ROT_EVEC This keyword specifies which eigenvector of the orbital rotation Hessian is to be used to rotate the original SCF orbitals. By default, it will use that associated with the lowest eigenvalue of the totally symmetric part of the block-factored Hessian, as this choice often leads to the lowest energy SCF solution. For RHF stability checks, only those instabilities which correspond to RHF solutions will be considered. It is important to understand that following non-symmetric eigenvectors lowers the symmetry of the wavefunction and that following RHF→UHF stabilities leads to a UHF solution. To converge the SCF roots associated with such instabilities, one must run the calculation in reduced symmetry and as a closed-shell UHF case, respectively. ROT_EVEC=n directs the program to follow the vector associated with the nth lowest eigenvalue having the proper symmetry (totally symmetric) and spin (RHF→RHF or UHF→UHF) properties. (Default : 0 [use the lowest eigenvalue])

Keywords relevant to the execution of the **Integral Transformation** :

DROPMO Specifies which molecular orbitals will be dropped from the post-SCF calculation. The orbitals are numbered in ascending order from the most stable (negative energy) to the most unstable (largest positive energy). Individual orbitals must be separated with a dash, while $x>y$ means orbitals x through y inclusive. For example, the string $1>10-55-58>64$, would result in orbitals 1,2,3,4,5,6,7,8,9,10,55,58,59,60,61,62,63 and 64 being dropped. For UHF calculations, the appropriate orbitals are deleted for both spin cases. No dropped MOs are currently allowed for gradient or property calculations. (Default : No dropped MOs)

VTRAN This keyword defines what type of integral transformation is to be performed in the program VTRAN. FULL/PARTIAL (=0) allows the transformation program to choose the appropriate type of transformation, while FULL (=1) requires a full integral transformation and PARTIAL (=2) means an MBPT(2) specific transformation where the (ab | cd) integrals are not formed. (Default : FULL/PARTIAL)

XFORM_TOL The tolerance for storing transformed integrals. Integrals less than 10^{-N} are neglected and not stored on disk. (Default: 11).

Keywords relevant to **Integral Processing** :

ABCDFULL This is a debug aid and is concerned with the storage of four virtual index integrals and effective Hamiltonian elements. It is only of relevance in RHF closed-shell calculations. It should never be necessary for users to set this keyword. Possible values are : UNKNOWN (=0) (the program will choose appropriate value); ON (=1) (full storage); OFF (=2) (reduced storage). Default : UNKNOWN.

ABCDTYPE Specifies the way that the $\langle ab||cd \rangle$ molecular orbital integrals are handled in post-MBPT(2) calculations. STANDARD (= 0) uses a technique which results in a minimum amount of CPU time but a maximum usage of disk space (particularly during execution of program XINTPRC); MULTIPASS (= 1) reduces the amount of disk storage required during XINTPRC, but at the cost of additional CPU time (note that this option requires HF2_FILE=2); AOBASIS (= 2) uses an AO-based algorithm to evaluate all terms involving these integrals. Again, use of this option results in considerably larger CPU times (particularly on vector supercomputers), but reduces the amount of required disk storage significantly. The use of ABCDTYPE=AOBASIS, however, is strongly recommended for single-point CC calculations on work station computers. Implementation within CC gradient calculations is planned, but currently not available.(Default: STANDARD).

HF2_FILE Specifies whether the HF2 (or HF2AA, HF2BB, and HF2AB) files are deleted after the first stage of integral processing. The default is to delete these files. However, when ABCDTYPE=MULTIPASS, these files must not be deleted at this time, and to stop deletion one must put HF2_FILE=2.

Keywords relevant to **Coupled-Cluster and Many-Body Perturbation Theory Calculations** :

- Note : the keywords NEWVRT and REFERENCE in the SCF section should also be consulted.

ABCDTYPE Specifies the way that the $\langle ab||cd \rangle$ molecular orbital integrals are handled in post-MBPT(2) calculations. STANDARD (= 0) uses a technique which results in a minimum amount of CPU time but a maximum usage of disk space (particularly during execution of program XINTPRC); MULTIPASS (= 1) reduces the amount of disk storage required during XINTPRC, but at the cost of additional CPU time (note that this option requires HF2_FILE=2); AOBASIS (= 2) uses an AO-based algorithm to evaluate all terms involving these integrals. Again, use of this option results in considerably larger CPU times (particularly on vector supercomputers), but reduces the amount of required disk storage significantly. The use of ABCDTYPE=AOBASIS, however, is strongly recommended for single-point CC calculations on work station computers. Implementation within CC gradient calculations is planned, but currently not available.(Default: STANDARD).

AO_LADDERS Can be used to control the algorithm used by ACES2 when terms involving $\langle ab||cd \rangle$ molecular orbital integrals are calculated in the atomic orbital basis (see keyword ABCDTYPE above). MULTIPASS (= 0) uses an approach where the AO integral file is read a number of times in order to ensure maximal vectorization and is usually the optimal strategy on supercomputers; SINGLEPASS (= 1) determines the contributions with only a single pass through the AO integrals, but at the cost of significantly reduced vectorization. In general, however, SINGLEPASS is *definitely preferable on workstations with RISC architectures*. (Default : MULTIPASS on all 64-bit machines (e.g., CRAY-YMP) ; SINGLEPASS on all 32-bit machines (e.g., IBM-RS6000, HP-735, SGI-Indigo)).

BRUCK_CONV This option specifies the convergence criterion in Brueckner-based coupled-cluster or configuration interaction calculations. The calculation is considered converged when the large single excitation amplitude falls below 10^{-N} , where N is the value associated with the keyword. (Default : 4)

BRUECKNER Specifies whether Brueckner orbitals are to be determined for the specified CC method. OFF (= 0) Brueckner orbitals are not to be determined, ON (= 1) They are to be determined. (Default : 0).

CC_CONV The convergence criterion for the CC equations. Equations are considered converged when the maximum change in amplitudes is less than 10^{-N} . (Default : N=7).

CC_MAXCYC The maximum number of CC iterations. (Default : 50).

NON-HF This flag tells the correlation energy code if the reference function satisfies the Hartree-Fock equations. Usually there is no need to set this parameter, since standard non-HF reference functions (QRHF and ROHF) set this flag internally. However, if you are inputting a set of orbitals to the correlation energy code directly, it may be necessary to use NON-HF. ON (= 1) signifies that a non-HF reference function is used; OFF (= 0) is used for HF reference functions. (Default : 0).

NTOP_TAMP Specifies how many t amplitudes will be printed for each spin case and excitation level. =N The largest N amplitudes for each spin case and excitation level will be printed. (Default : 15).

ORBITALS This keyword is used in non-HF calculations to specify semicanonical orbitals are used. “Semicanonical” orbitals are obtained by diagonalizing the occupied-occupied and virtual-virtual blocks of the spin-orbital Fock matrix and can be advantageously exploited in certain post-SCF calculations (particularly for ROHF-MBPT and non-iterative triple excitation corrections). There is no specific default value for this parameter, and considerable logic is used internally to choose the orbital type in post-SCF non-HF calculations if the keyword is not included. It is strongly recommended that this keyword not be used by anyone who is not thoroughly familiar with non-HF CC/MBPT methods, since the logic used to set the default value is sound. STANDARD (= 0) uses the orbitals obtained in the reference function calculation without modification; SEMICANONICAL (= 1) forces a transformation to semicanonical orbitals. (Default : See above).

ORDER_RLE Maximum size of the R matrix used in the reduced linear equation method for solving the CC equations. The maximum value allowed is 19. (Default : 5). *Note : This keyword has been changed to CC_EXPORDER.*

QRHF_GENERAL The presence of this keyword specifies that a QRHF-based coupled cluster calculation is to be performed. In this method, an SCF calculation is first performed on a closed-shell state specified by either the OCCUPATION keyword or the CHARGE and MULTIPLICITY keywords. Using the closed-shell state orbitals, a coupled-cluster calculation is then performed on an open-shell state generated from this closed-shell state by removing, adding, or exciting electrons. Any number of electrons may be added to the α spin orbitals, and any number may be removed from the β orbitals. The values associated with this keyword must be separated by slashes “/” and take on either negative (for removal of β electrons) or positive (addition of α electrons) integer values. The absolute values of these parameters specify the symmetry block(s) involved in the addition or removal of electrons. The numerical ordering of the symmetry blocks is consistent with that used in specifying the orbital occupations. By default, the electrons are added to the lowest unoccupied molecular orbital in the symmetry block and removed from the highest occupied molecular orbital. Different orbitals may be specified with the QRHF_ORBITAL keyword (see below). **NOTE:** In Release 2 of ACES

II gradients and property calculations are currently available *only* for cases involving addition or removal of electrons. Mixed cases involving both processes are not available in Release 2. In the RJB group code gradients and properties are available for open-shell singlet CCSD wavefunctions for the case that the open-shell orbitals have different symmetries.

QRHF_ORBITAL By default, in QRHF calculations, electrons are removed from the highest occupied orbital in a symmetry block (symmetry block HOMO), while electrons are added to the lowest unoccupied orbital within a symmetry block (symmetry block LUMO). The purpose of the QRHF_ORBITAL keyword is to allow additional flexibility in choosing which orbitals will have their occupation numbers altered. The value of this keyword gives the offset with respect to the default orbital for the orbital which will be depopulated (or populated) in QRHF-CC calculations. For calculations involving more than one removal or addition of electrons, values are separated by commas and correspond to the QRHF_GENERAL input on a one-to-one basis. For example, specifying QRHF_GENERAL=2/-4,QRHF_ORBITAL=3/2 means that an electron will be added to the third lowest virtual in symmetry block 2 and another will be removed from the second highest occupied orbital in symmetry block 4. Examples given later in this manual further illustrate the QRHF input options and may help to clarify any confusion resulting from this documentation. (Default : 1)

QRHF_SPIN This keyword specifies the spin of the electrons modified by the QRHF_GENERAL and QRHF_ORBITAL keywords, where a value of 1 means α spin, while 2 corresponds to a β electron. By default, electrons that are removed are assigned to β spin, while added electrons are α . Note that this option allows one to construct low-spin determinants, which generally are unsuitable for single-reference coupled-cluster calculations. An important exception is the open-shell singlet coupled-cluster method (see keyword REFERENCE=TWODET below).

RLE Specifies the type of convergence acceleration used to solve the CC equations. STANDARD (= 0), the RLE method of Bartlett and Purvis is used with periodic extrapolation of the solution vector; DIIS (= 1) uses the DIIS approach of Pulay; NOJACOBI (= 2) uses the RLE method with continuous extrapolation; OFF (= 3) no convergence acceleration method is used. In general, STANDARD provides the best performance, but DIIS and particularly NOJACOBI may offer advantages in cases where the reduced subspace becomes singular too rapidly. NOJACOBI requires some additional disk storage which may be disadvantageous for very large calculations. Use of OFF is generally a bad idea for CC calculations, but may be preferable to the other choices for configuration interaction calculations. (Default : 0) *Note : This keyword has been changed to CC_EXTRAPOL and the default has been changed to DIIS.*

TAMP_SUM Specifies how often the largest t amplitudes are to be printed. =0 Amplitudes are printed at the beginning and end of the run, =1 Amplitudes are printed every iteration, =2 Amplitudes are printed every other iteration, etc. (Default : 5).

Keywords relevant to **Geometry Optimizations, Transition State Searches and Other Energy Derivative Calculations** :

CONVERGENCE Specifies convergence criterion for geometry optimization. Job terminates when RMS gradient is below 10^{-N} Hartree/bohr, where N is the value specified by CONVERGENCE. (Default: 4; Value must be specified as an integer).

CPHF_CONVER Specifies the convergence criterion for the iterative solution of the CPHF and Z-vector equations. The solutions are considered to be converged when the error falls below 10^{-N} . (Default : 12)

CPHF_MAXCYC The maximum number of cycles allowed for the solution of the CPHF and Z-vector equations. (Default : 64)

CURVILINEAR Specifies whether or not Hessian matrix is transformed (nonlinearly) to curvilinear internal coordinates. A value of 0 (or OFF) turns the transformation off if the analytic force constants are not available, while it is always performed if CURVILINEAR=1 (or ON). Values higher than 1 (or NO) unconditionally turn the transformation off.(Default: ON if analytic Hessian is available, OFF otherwise).

DERIV_LEV Specifies whether or not derivatives of the energy are to be calculated and if so whether first or second. = 0 Derivatives not calculated, =1 First derivatives to be calculated, =2 Second derivatives to be calculated. This need not be set in geometry optimization or vibrational frequency calculations since it is automatically set if the appropriate options in the ACES2 namelist are set. (Default : 0). *NOTE: It is probably dangerous to use this keyword.*

EIGENVECTOR Specifies which eigenvector of the totally symmetric part of the block-factored Hessian is to be followed *uphill* in a transition state search. Eigenvectors are indexed by their eigenvalues – the lowest eigenvalue is 1, the next lowest is 2, etc. The default is 1, which should *always* be used if you are not looking for a specific transition state which you know corresponds to motion along a different mode. Of course, the value of EIGENVECTOR has no meaning if METHOD is not set to TS. (Default: 1; Value must be specified as an integer).

EVAL_HESS Tells ACES II to recompute the Hessian every N cycles, where N is the supplied argument. For correlated calculations, the Hessian is evaluated *only at the SCF level*. (Default: 0 (no recomputation); Value must be specified as an integer).

MAX_STEP Specifies largest step (in millibohr) which is allowed in geometry optimizations. (Default: 300).

METHOD Specifies the geometry optimization strategy. Four values are permitted: 0 (or NR) — Straightforward Newton-Raphson search for minimum; 1 (or RFA) — Rational Function Approximation search for minimum (this method can be used to find

minima when the initial structure is in a region where the number of negative Hessian eigenvalues is nonzero); 2 (or TS) Cerjan-Miller eigenvector following search for a transition state (can be started in a region where the Hessian index (the number of negative Hessian eigenvalues) is not equal to one); 3 (or MANR) — Morse-adjusted Newton-Raphson search for minimum (very efficient minimization scheme, particularly if the Hessian is available); 4 is currently unavailable; 5 (or SINGLE_POINT) is a single point calculation. (Default: SINGLE_POINT); 6 — Do a geometry optimization but compute the gradients from energies only.

NEGEVAL Tells the program what to do if negative eigenvalues are encountered in the totally symmetric Hessian during an NR or MANR search. If NEGEVAL=0 (or ABORT), then the job will terminate with an error message; if NEGEVAL=1 (or SWITCH) the program will just switch the eigenvalue to its absolute value and keep plugging away (this is strongly discouraged); and if NEGEVAL=2 (or →RFA), METHOD is switched to RFA internally and the optimization is continued. (Default: ABORT).

OPT_MAXCYC This specifies the maximum allowed number of geometry optimization cycles. (Default : 50)

PERT_ORB The keyword STANDARD means that the gradient formulation assumes that the perturbed orbitals are not those in which the Fock matrix is diagonal. CANONICAL means that the perturbed orbitals are assumed to be canonical. This keyword must be set to CANONICAL in gradient calculations with methods which include triple excitations (MBPT[4], CCSD+T[CCSD], CCSD[T], and QCISD[T]). (Default : STANDARD)

RELAX_DENS Specifies whether the relaxed density matrix is computed for correlated wave functions. OFF (= 0) The relaxed density will not be computed, ON (= 1) it will be computed. This keyword is set by the program and should *not* be set by the user; if the relaxed density is desired, then PROPS=1 should be set. (Default : 0).

SCALE_ON Controls whether step scaling is based on the absolute step length (1-norm) (=0 or MAG(S)) or the largest individual step in the internal coordinate space (=1 or MAX(S)). (Default: MAG(S)).

TRANS_INV This keyword specifies whether or not translational invariance is exploited in derivative calculations. USE(=0) specifies that translational invariance is exploited, while IGNORE (=1) turns it off. (Default : USE)

Keywords relevant to **Vibrational Frequency Calculations** :

FD_CALTYPE In finite difference calculations using the FINDIF option, this keyword controls the algorithm used to compute the harmonic force constants. GRADONLY (= 0) evaluates the force constants and dipole moment derivatives by numerical differentiation of analytic gradients; ENERONLY (= 1) evaluate the force constants by second differences of energies (dipole moment derivatives are not evaluated); while MIXED (= 2) evaluates 1×1 blocks of the symmetry-blocked force constant matrix by second differences of energies and all other elements by first differences of gradients. The GRADONLY and MIXED approaches may, of course, only be used when using computational methods for which analytic gradients are available.

FD_PROJECT In finite difference calculations using the FINDIF option, this keyword controls whether or not rotational degrees of freedom are projected out of symmetry-adapted coordinates. ON (= 0) uses rotationally projected coordinates, while OFF (= 1) retains the rotational degrees of freedom. At a stationary point on the potential surface, both options will give equivalent harmonic force fields, but OFF should be used at non-stationary points. (Default : ON)

FD_IRREPS This keyword may be used to request that only vibrational frequencies of certain symmetry types be evaluated in a VIBRATION=FINDIF calculation. The numbers of the irreducible representations for which vibrational analysis is to be performed are separated by slashes. For example, FD_IRREP=1/3/4 means compute the frequencies of modes transforming as the first, third, and fourth irreducible representations. If a symmetry is specified for which there are no vibrational modes, the program will terminate. The labels of the irreducible representations for this keyword are not usually the same as those used in the rest of the calculation. Moreover, for some point groups, for example those of linear molecules, the two sets of labels refer to different subgroups. There is as yet no straightforward way to determine what they will be without starting a calculation. If one runs the xjoda and then the xsymcor executables the relevant irreducible representations will be listed. If all vibrational frequencies are desired, this keyword need not be included. (Default : compute vibrational frequencies for all irreducible representations)

FD_STEPSIZE Specifies step length (in 10^{-4} amu^{1/2} bohr) used in generating the force constant matrix by finite difference of Cartesian gradients. (Default: 50 (0.005 amu^{1/2} bohr)).

FD_USEGROUP In finite difference calculations using the FINDIF option, this keyword specifies the point group to be used in generating the symmetry-adapted vibrational coordinates. FULL (= 0) specifies the full molecular point group, COMP (= 1) specifies the Abelian subgroup used in the electronic structure calculation. (Default : FULL)

POINTS Specifies either single (=1, or SINGLE) or double (=2, DOUBLE) sided numerical differentiation in the finite difference evaluation of the Hessian. Two-sided numerical

differentiation is considerably more accurate than the single-sided method and its use is *strongly recommended for production work*. (Default: DOUBLE).

VIBRATION For vibrational frequency calculations. A value of 1 (or EXACT) means perform normal mode analysis on an analytic force constant matrix and computes rotationally projected frequencies and infrared intensities. A value of 2 (or FINDIF) signals ACES II to compute the force constant matrix by finite difference of analytically computed gradients or energies using symmetry-adapted mass-weighted Cartesian coordinates (see entries for keywords FD_CALCTYPE, FD_PROJECT and FD_USEGROUP). A value of 3 (or FINDIF_OLD) uses an alternative algorithm based on energy gradients which minimizes the number of gradients which must be evaluated. However, these gradients will be evaluated for nonsymmetric structures, as opposed to the FINDIF method which evaluates more gradients but in general does not require them to run in C_1 symmetry. A choice between FINDIF and FINDIF_OLD requires a great deal of knowhow and experience, and the FINDIF option is recommended for all but the most knowledgeable of ACES II gurus. (Default: NO)

Keywords relevant to **Property Calculations** :

- Review also keywords for geometry optimizations. For excited state properties, see also keywords for excited state calculations. Consult also sections of the manual on NMR and TDHF calculations.

EOMPROP Specifies method of calculating EOM-CCSD second-order properties (polarizabilities and spin-spin coupling constants). =CILIKE (use CI-like formula, which is not rigorously size-intensive); =LINEAR (remove unlinked diagrams from CI-like formula); =QUADRATIC (use quadratic formula); =COMBO (compute all approximations). (Default : CILIKE). This keyword only has meaning when PROP=VCCEH, PROP=J_SO, PROP=J_FC, PROP=J_SD, or PROP=JSC_ALL.

PROPS Specifies whether one-electron properties are to be calculated at the end of the run. OFF (=0) Do not compute properties; FIRST_ORDER(=1) compute first-order properties (dipole moment, quadrupole moment, electrical field gradients, spin densities, etc.; SECOND_ORDER(=2) compute second-order properties (static polarizabilities; limited to CALC=SCF and (C_1 symmetry only) CALC=MBPT[2] — for other methods finite field calculations can be used); NMR(=3) compute NMR chemical shifts; J_SO(=8) calculate spin-orbit contribution to spin-spin coupling constant J; J_FC(=9) calculate Fermi contact contribution to spin-spin coupling constant J; J_SD(=10) calculate spin-dipole contribution to spin-spin coupling constant J; CCSDEH(=11) calculate EOM-CCSD dynamic polarizabilities; JSC_ALL(=13) calculate all of J. Note : NMR calculations currently require Cartesian basis functions (*i.e.* SPHERICAL=ON cannot be used). (Default : 0).

TDHF Specifies whether a time-dependent Hartree-Fock calculation of nonlinear optical properties is to be performed. TDHF=0/TDHF=OFF means a TDHF calculation is not to be performed (the default); TDHF=1/TDHF=ON means that a TDHF calculation is to be performed. This keyword can only be used for CALC=SCF and for closed-shell systems, and no orbitals may be dropped. The nonlinear properties which are to be calculated are specified by a namelist which is put at the end of the ZMAT file. A description of this namelist and examples are provided in the section on time-dependent Hartree-Fock calculations.

TREAT_PERT This keyword is used for certain types of correlated second derivative calculations [presently only GIAO NMR shift calculations] and directs ACES II to either treat all perturbations at once or treat them sequentially. The latter approach results in less demand for physical disk space, but at the cost of increased cpu time. Available options are SIMULTANEOUS (=0); and SEQUENTIAL (=1). (Default : SIMULTANEOUS).

XFIELD Specifies the X-component of an external electric field. The value must be specified as an integer and the field used by the program will be the value of the keyword $\times 10^{-6}$. This allows field strengths $|\mathcal{E}| > 10^{-6}$ to be used. (Default : 0)

YFIELD Specifies the Y-component of an external electric field. See above. (Default : 0)

ZFIELD Specifies the Z-component of an external electric field. See above. (Default : 0)

Keywords relevant to **Equation-of-motion Excitation Energy Calculations** :

EOM_MAXCYC This parameter controls the number of iterations in EOM excited state calculations. If it has the value N and $NROOT$ roots have been requested for a given symmetry, the program will allow up to $N*NROOT$ iterations to find all the requested roots for that symmetry. (Default : 50)

ESTATE_MAXC The maximum number of iterative diagonalization steps for each root in excited state calculations. (Default : 30) [*Note* : this option is not currently used by the EOM excited state code; its role has been taken by EOM_MAXCYC]

ESTATE_PROP This keyword applies only to EOM-CC calculations and specifies whether any excited state one-electron properties are to be calculated. Proper use of this keyword might require some fairly advanced knowledge of quantum chemistry and the available options are discussed here. The options are : OFF (=0) [no properties or transition moments are calculated]; EXPECTATION (=1) [transition moments and dipole strengths are calculated along with selected one-electron properties which are evaluated as expectation values]; UNRELAXED (=2) [selected one-electron properties are calculated in an approximation that neglects relaxation of molecular orbitals]; RESPONSE (=3) [selected one-electron properties are calculated as analytic first derivatives of the energy]. Except for EOMCC calculations on two-electron systems (which are exact), properties obtained by the three approaches will not be equivalent. The default value for this keyword is slightly complicated. For TDA calculations, the default is EXPECTATION since the evaluation of transition moments involves only a negligible amount of additional computation relative to the evaluation of the excitation energies. For EOMCC, the default is OFF since evaluation of any transition moments or properties requires approximately twice the computational time. ESTATE_PROP=RESPONSE is not available for EOMCC calculations. Transition moments and dipole strengths are evaluated by default for all values of ESTATE_PROP other than OFF.

ESTATE_SYM This specifies the number of excited states which are to be determined in each irreducible representation of the computational subgroup. The program attempts to find all of the lowest roots, but this is not guaranteed because the eigenvalue problem is not solved by direct matrix diagonalization, but rather by an iterative (modified Davidson) algorithm. For excited state gradient calculations (TDA only), only one root can be specified, so only one non-zero entry in the string is allowed, and that must be set to one. The format used for this keyword is identical to that used in the OCCUPATION keyword. For example, for a computational subgroup having four symmetry species, the string ESTATE_SYM=3/1/0/2 specifies that 6 total roots should be searched for, three in the first block, one in the second block, and two in the fourth block. (Default : All zeros).

ESTATE_TOL The tolerance used in converging EOM-CC excited state calculations. By default, the iterative diagonalization continues until the RMS residual falls below 10^{-5} , and this value can be changed to 10^{-N} by specifying ESTATE_TOL=N. (Default : 5)

EXCITE This keyword specifies the type of excitation energy calculation which is to be performed. Available options are NONE (=0); TDA (=1) [the Tamm-Dancoff, or configuration interaction singles (CIS) approach]; and EOMEE (=3) [the equation of motion coupled-cluster approach]. At present, the EOMCC method is limited to CCSD reference states. (Default : NONE).

Keywords relevant to **Equation-of-motion Electron Affinity Calculations** :

EA_CALC Specifies whether an equation-of-motion coupled-cluster electron affinity calculation is to be performed or not. =NONE means that an EOM-CC EA calculation will not be performed; =EA_EOMCC means that an EOM-CC EA calculation will be performed. (Default : NONE)

EA_SYM Specifies the symmetry and spin of the added electron in EOM-CC EA calculations. In RHF calculations a string of NIRREP integers, separated by hyphens ('-'), where NIRREP is the order of the computational point group, is given. For example : EA_SYM=3-2-0-2 means that the program is to calculate the energies of 3 electron attached states in which the additional electron occupies an orbital of symmetry 1, 2 electron attached states in which the additional electron occupies an orbital of symmetry 2, and so on. In UHF or ROHF calculations 2*NIRREP integers are specified. The first set denotes α spin and the second β spin, and the two sets are separated by a slash ('/'). For example : EA_SYM=3-2-0-2/0-1-0-4. If EA_CALC has been set to EA_EOMCC and EA_SYM is not specified, the program will try to find the lowest energy electron attached state. For further information about EOM-CC EA calculations, see the section of this manual entitled "Equation-of-motion coupled-cluster electron attachment calculations".

The job control parameters are entered following a mandatory blank line at the end of the parameter input section (see above). Delimiters separating individual keywords can be commas (","), semicolons (";"), ampersands ("&") or carriage returns.

B.7 Non-standard Basis Set Specification

If the BASIS keyword is set to SPECIAL, or is not included in the keyword list, the basis set specification will be read directly from the ZMAT file after the job control parameters. One blank line must separate the last line of the job control parameters from the beginning of the basis set input section. Each entry must be placed on an individual line, and the ordering of atoms must follow the Z-matrix ordering *exactly*. The names will then be checked against the GENBAS file in the current directory and the basis, if found, will be extracted and placed into the input file appropriate for the integral program. If the basis is not found, ACES II exits with an error condition. ACES II does not check to make sure that the atom to which the basis set belongs corresponds with the atomic designation in the corresponding row of the Z-matrix. No entries are made for dummy atoms. The format of the basis set names in GENBAS is: XX:BASNAM, where XX is the atomic symbol of the atom (in capital letters), and BASNAM is the name of the basis, usually assigned by the whim of the individual who entered it. For inexperienced users, it is probably best to search the GENBAS file for “XX:”, where again XX is the atomic symbol since this will allow you to see all of the available basis sets for that atom. A description of the format of the GENBAS file and its contents are given in Section IV D of this manual.

B.8 Non-standard File Handling and Restart Calculations

Facilities are available in ACES II which allow user control over file locations and permit certain types of restart calculations to be carried out in a straightforward way. By default, all files used by ACES II [IIII, IIJJ, IJIJ, IJKL, JOBARC, JAINDX, MOINTS, GAMLAM, MOABCD, DERGAM, SECDEF, OPTARC, *etc.*] are kept in the working area [the directory from which the **xaces ii** program is invoked]. In the usual case, this is a scratch directory of some sort and these files are often deleted immediately after the job terminates. However, the location of these files can be respecified in a simple way through lines entered *at the very top* of the ZMAT file. The location of any of file (except the input file ZMAT of course) may be respecified as follows :

```
% FILE=/u/usr/safe/FILE
```

where FILE is the name of the particular file. For example, if one wants to calculate isotopic shifts after a vibrational frequency calculation has been performed [see the section which describes how to run these calculations], it is useful to keep the JOBARC and JAINDX files in a safe place. For example, if one wanted to do such a calculation for water, the following ZMAT file might be used :

```
% JOBARC=/u/usr/safe/JOBARC
% JAINDX=/u/usr/safe/JAINDX
```

```
Water DZP vibrational frequency calculation
H
```

```
O 1 R
H 2 R 1 A
```

```
R=0.957
A=104.51
```

```
*ACES2(CALC=SCF,VIB=EXACT,BASIS=DZP)
```

After this job executes, the files JOBARC and JAINDX are located in the directory /u/usr/safe. Then one can simply move to this directory, create the appropriate ISOMASS files (see section on isotopic shift calculations) and run the ACES II executable **XJODA** directly.

The re specification of file locations also can be used to facilitate restarts of geometry optimization and finite difference vibrational frequency calculations. For optimizations, the following files must be saved : JOBARC, JAINDX and OPTARC. For finite difference harmonic frequency calculations (*using* VIB=FINdif *only!*), one also needs to save JOBARC and JAINDX. If the job exits because a cpu time limit has been exceeded, then re submission will automatically initiate the restart process (no additional keywords are necessary). If however, the maximum number of optimization cycles has been exceeded, then the OPT_MAXCYC keyword needs to be modified. If for example, one wants to run an optimization three cycles at a time, the following ZMAT file might be used for the first stage:

```
% JOBARC=/u/usr/safe/JOBARC
% JAINDX=/u/usr/safe/JAINDX
% OPTARC=/u/usr/safe/OPTARC
```

```
Water DZP geometry optimization
```

```
H
O 1 R*
H 2 R* 1 A*
```

```
R=0.957
A=104.51
```

```
*ACES2(CALC=SCF,VIB=EXACT,BASIS=DZP,OPT_MAXCYC=3)
```

After this first phase completes, then OPT_MAXCYC can be reset to 6 and the job resubmitted. The program system only keeps track of the total number of steps completed, and not the incremental values.

While on supercomputers, it is usually recommended to save the restart files in the user's own disk area, the situation is quite different for work station computers. Here, the available

scratch space for running a job is often put on several independent disks (in most cases two or three) and it is necessary to redirect several of the large files in order to run the job. This can be done in the same way as described above using the % cards in the ZMAT file.

B.9 Molecular Orientation

The orientation of the molecule in the Cartesian basis is related to its point group. Two orientations are used extensively in the ACES II program system: the “standard” or “computational” orientation (which is a standard orientation for the computational point group) and a “canonical” orientation (which is the standard orientation for the full point group). Note that in some cases, the two orientations are identical. All calculations are performed in the computational orientation, so orbital symmetries should be specified according to this Cartesian axis system. For the most part, the canonical orientation is used internally for tasks such as determining irreducible representations or other properties related to the full point group. The standard orientation for each point group follows

C_n Groups Rotation axis along z .

D_n Groups Rotation axes coincident with Cartesian axes, with the highest order axis along z .

C_s Group Plane of symmetry is xy .

S_n Groups S_n axis along z .

C_{nv} Groups C_n axis along z , xz is σ_v .

C_{nh} Groups C_n axis along z , xy is σ_h .

D_{nh} Groups C_n axis along z , one C_2 axis along x .

D_{nd} Groups S_{2n} axis along z , one C_2 along x .

T Group C_3 axis along (q, q, q) .

T_d Group S_4 axis along z .

T_h Group C_3 axis along (q, q, q) , symmetry planes are xy , xz and yz .

O Group C_4 axes along x , y and z .

O_h Group C_4 axes along x , y and z .

I Group C_5 axis along z , one C_3 lies in the xz plane.

I_h Group C_5 axis along z , xz is a symmetry plane.

For groups with ambiguities (D_2 , D_{2h} , C_{2v}), there is no standard orientation at present, and you may want to run ACES II once to see which orientation is used before assigning orbital symmetries. For example, water belongs to the C_{2v} point group and the symmetry plane containing the hydrogen atoms may be assigned to either the xz or yz planes, leading to an ambiguity between the b_1 and b_2 irreducible representations. Eventually, some criterion may be established which defines a standard orientation for these groups thereby alleviating this annoyance.

B..10 Summary and Examples

To summarize, the ZMAT file contains the following information:

Preamble Nonstandard file declarations (if used).

Line 1 Title for calculation.

Lines 2→i Z-matrix or Cartesian coordinates.

Line i+1 Blank.

Lines i+2→j Parameter input (if Z-matrix above).

Line j+1 Blank.

Lines j+2→k ACES2 Namelist.

Line k+1 Blank.

Lines k+2→l Basis set input (if BASIS=SPECIAL).

Line l+1 Blank.

Lines l+2→m ECP input (if ECP=ON).

Example 1:

```
CH2O --> H2 + CO transition state search.  DZP basis.
```

```
0
```

```
C 1 R*
```

```
H 2 R2* 1 A*
```

```
H 3 RHH* 2 A* 1 T
```

```
R=1.254
```

```
R2=1.08
```

```
RHH=1.01
```

```
A=133.5
```

```
T=0.
```

```
*ACES2(METHOD=TS,MAX=150,JODA_PRINT=15,CONVERGENCE=6,BASIS=DZP  
CALCLEVEL=MBPT[2],OCCUPATION=7-1)
```

As implied by the job title, this ZMAT file specifies an initial geometry and basis set for a transition state search. In the Z-matrix file, the parameters designated R, R2, RHH, and A will be optimized, while parameter T will be fixed at 0 degrees. It is rather easy to see that this constrains the structure to be planar. The ACES2 namelist parameters tell ACES II that 1) this is a transition state search, using the Cerjan-Miller eigenvector following method; 2) the maximum allowed step length is 150 millibohr (this corresponds to the absolute step length (1-norm) since the default value of SCALE_ON has not been overridden); 3) that the joda_print level is set to 15; 4) that the optimization will continue until the RMS internal coordinate gradient falls below 10^{-6} hartree/bohr; 5) the DZP basis sets is to be used; 6) the calculation type is MBPT[2]; and 7) the occupation, appropriate for the C_s point group, is 7-1. It should be noted that the CALCLEVEL keyword uses brackets '[']' rather than parentheses '()'. The keyword "TS" automatically turns on the necessary gradient options.

Example 2.

```
N4 finite difference frequency calculation
```

```
N
```

```
X 1 R
```

```
N 2 R 1 TDA
```

```
N 2 R 1 TDA 3 T
```

```
N 2 R 1 TDA 4 T
```

```
R=0.945
```

```
TDA=110.
```

```
T=120.
```

```
*ACES2(VIB=FINDIF,BASIS=TZ2P,CALCLEVEL=1)
```

This ZMAT file directs ACES II to perform a harmonic frequency calculation for N_4 , and to compute the force constants by numerical differentiation of analytic gradients. The TZ2P basis set is selected. Note that the TDA parameter is used in the Z-matrix. Although the specified value in the parameter input section is not the exact tetrahedral angle, the program will convert TDA to the correct value (109.4712206... degrees) internally. As for example 1, the ACES2 namelist specifies an MBPT(2) calculation, but in a different manner. As the FINDIF option is being used, no symmetry must be specified and so the orbital symmetry specification is omitted. This is because the determination of the force constants requires several gradient calculations at geometries with different symmetries. The FINDIF option automatically turns on the requisite gradient and property options. Note that no asterisks may appear in the Z-matrix in a frequency calculation — this is a common error since frequency calculations are usually preceded by geometry optimizations, which require the asterisks.

Example 3.

Beryllium borohydride, D3d structure, geometry optimization.

X

BE 1 R1

B 2 R* 1 A

B 2 R* 1 A 3 T

X 2 RX* 1 A 3 T

X 2 RX* 1 A 4 T

H 5 RHX* 2 A 1 T

H 5 RHX* 2 A 1 T60

H 5 RHX* 2 A 1 TM6

H 6 RHX* 2 A 1 T12

H 6 RHX* 2 A 1 TM2

H 6 RHX* 2 A 1 T0

X 3 R1 2 A 1 T0

X 4 R1 2 A 1 T0

H 3 RHT* 13 A 2 T

H 4 RHT* 14 A 2 T

R1=1.207

R=1.05183

RX=0.08546

RHX=1.48313

RHT=0.569

A=90.

T=180.

T60=60.

T12=120.

T0=0.

TM2=-120.

TM6=-60.

*ACES2(METHOD=MANR,EVAL_HESS=3,MAX=750,SYM=OFF,CALC=1)

BE:6-31G*

B:DZP

B:DZP

H:DZ

H:DZ

H:DZ

H:DZ

H:DZ

H:DZ
H:DZ
H:DZ

This ZMAT file specifies a geometry optimization for the D_{3d} isomer of beryllium borohydride, BeB_2H_8 , using the MANR algorithm. The Hessian will be reevaluated every three cycles, and the maximum step length is set to 750 milliboehr. The `SYM=OFF` statement tells ACES II to turn off symmetry in the calculation. The “non-standard” basis set input format is used, which is automatically selected since the `BASIS` keyword defaults to its default value, `SPECIAL`. An `MBPT(2)` calculation is specified. Since the symmetry has been turned off, the `OCCUPATION` variable is omitted. Again, the geometry optimization request automatically turns on the necessary gradient options.

Example 4.

```
TEST.011 ROHF-HFDFT NUMERICAL VIBRATION CALCULATION FOR NH2
```

```
N
```

```
H 1 R
```

```
H 1 R 2 A
```

```
R=1.031
```

```
A=102.9
```

```
*ACES2(CALC=HFDFT,REF=ROHF,MULTI=2,BASIS=TZ2P,VIB=FINDIF  
FD_CALTYPE=ENERONLY)
```

This ZMAT file specifies a finite difference frequency calculation for NH_2 using energy points only with the HF-DFT method. The default Becke-Lee-Yang-Parr (B-LYP) functional value for the energy is used. Another functional could be selected as, for example, `FUNCT=LYP` (which would use the LYP correlation functional and the Hartree-Fock exchange energy). The `FD_CALTYPE=ENERONLY` option should be used only for those cases for which analytical gradients are not available (ACES II chooses whether to use gradients or energy points in a finite difference calculation in a way which minimizes the number of points which must be determined).

Example 5.

NO3 calculation

N

O 1 R

O 1 R 2 A

O 1 R 2 A 3 T

R=1.236

A=120.

T=180.

*ACES2(BASIS=TZ2P,CALC=CCSD,QRHF_G=-2,QRHF_O=2,CHARGE=-1,PROP=1)

This ZMAT file specifies a QRHF/CCSD property calculation for the NO_3 radical, using a triple-zeta plus double polarization (TZ2P) basis set. Note that the charge corresponds to the M species (the NO_3 anion) rather than NO_3 itself. The OCCUPATION keyword is left out, therefore the orbital occupations will be estimated by the SCF program. Keep in mind that the SCF calculation *is performed for NO_3^- and not NO_3* , so that occupancy specification, charge and multiplicity *must always correspond to the closed-shell species*. The zeroth-order state for the QRHF calculation will be that obtained by removing an electron from the second highest lying molecular orbital in symmetry block 2. With luck, the CCSD will converge to the second lowest doublet state of the this symmetry type. After the CCSD calculation has completed, the first-order properties will be calculated analytically.

Example 6.

EOM-CCSD excitation energies and transition moments for water

H

O 1 R

H 2 R 1 A

R=0.957

A=104.5

*ACES2(BASIS=TZ2P,CALC=CCSD,EXCITE=EOMEE,ESTATE_SYM=1/1/1/1
ESTATE_PROP=EXPECTATION)

This ZMAT file specifies an equation of motion coupled-cluster excitation energy calculation for the water molecule, using the TZ2P basis set. The program will attempt to find the lowest root in each of the four symmetry species of the C_{2v} point group, and oscillator strengths will be evaluated for all excited states.

Example 7.

GIAO-MBPT(2) NMR chemical shift calculations for the benzonium cation

C	0.000000	0.000000	-1.393615
C	0.000000	0.000000	1.413386
C	0.000000	-1.250334	-0.632255
C	0.000000	1.250334	-0.632255
C	0.000000	-1.237545	0.742530
C	0.000000	1.237545	0.742530
H	0.853412	0.000000	-2.102274
H	-0.853412	0.000000	-2.102274
H	0.000000	-2.189398	-1.180688
H	0.000000	2.189398	-1.180688
H	0.000000	-2.161755	1.310996
H	0.000000	2.161755	1.310996
H	0.000000	0.000000	2.502007

```
*ACES2(CALC=MBPT[2],BASIS=SPECIAL,PROP=NMR,CHARGE=1
COORD=CARTESIAN,TREAT_PERTURBATION=SEQUENTIAL,MEMORY=24000000)
```

```
C: dzp
C: dzp
C: dzp
C: dzp
C: dzp
C: dzp
C: dzp
H: dz
H: dz
H: dz
H: dz
H: dz
H: dz
H: dz
```

This example features a calculation of the NMR chemical shifts at the MBPT(2) level using gauge-including atomic orbitals (GIAOs). NMR shifts are requested by the keyword PROP=NMR. The input of the coordinates in this example is via cartesian coordinates (in Ångströms) and the basis set is specified via the non-standard input which allows the use of different basis sets for different atoms (in this case, dzp for C and dz for H). To reduce disk space requirements, the keyword TREAT_PERTURBATION=SEQUENTIAL is specified

which forces the program to treat each magnetic field perturbation separately, thus storing only GIAO integrals of one particular type at one time. The memory requirement in this example is set to 24000000.

C. Common Mistakes and Sources of Confusion

In an ideal world a computer program will trap every input error and tell the user exactly what is wrong and how to fix it. Some might even say that the program should figure out what the user intended and proceed accordingly ! ACES II, traps a number of input errors with explanation. Some errors, however, which will cause the input processor to fail but without a clear explanation. There are other errors that will not become apparent until well into, or even after, execution. The following are some tips which we hope will help users to detect errors. We are, of course, continually trying to improve error detection. At the same time, users should be aware of the need to be careful in their preparation of input decks, and should not expect software to be a substitute for elementary checking. In addition, *small* test calculations are recommended when one is trying a type of calculation for the first time. We welcome suggestions for additions to this list.

- There is no blank line at the end of the ZMAT file.
- The copy of the ZMAT file in the workspace has been changed while the job is running. It is dangerous and in fact should never be necessary to do this if one works in a sensible manner.
- The BASIS=SPECIAL option has been used but the order of the basis sets does not correspond with the order of atoms in the Z-matrix. The code does not check this. For example, one is allowed to put a C basis set on a CL atom, if one so desires. It is quite possible to make this type of error and for the calculation not to abort. It is *usually* clear from the results that something strange has occurred. Note that this “flexibility” is convenient for using ghost atoms, since we often want to put various basis sets on ghost atoms.
- All atomic symbols must be in upper case. For example, “Cl” has to be entered as “CL”.
- Problems with *’s in the Z-matrix. They were absent in a geometry optimization, but were needed. They were present in a vibrational frequency calculation but were not needed. *Most likely both of these scenarios are trapped now.*
- Vibrational frequency and geometry optimization keywords are both included in the *ACES2 namelist (*e.g.* both VIBRATION and METHOD are present) and the program is highly confused.
- There is more than one space separating entries in the Z-matrix.
- Lower case characters have been used instead of upper case ones.
- The hierarchy of the OCCUPATION, CHARGE, and MULTIPLICITY keywords and what happens in open-shell SCF calculations can lead to confusion. Several helpful pointers : (a) If OCCUPATION has been specified, the CHARGE and MULTIPLICITY

keywords are ignored. It is, however, good practice to make these consistent with OCCUPATION; (b) If the REFERENCE keyword is absent from an open-shell calculation or is erroneously set to RHF, unpredictable things happen. ACES II does not have a default for type of open-shell SCF, so basically if the system is an open-shell, a value for REFERENCE needs to be specified; (c) the occupation specified in the GUESS file takes precedence over that specified in the ZMAT file. Again, it is sensible to make them the same to avoid confusion; (d) There have been cases where one is running an open-shell calculation but the OCCUPATION keyword only specifies an α occupation. What can happen then is that the program thinks that the system has no β electrons, though this was not intended.

- There is a comma at the end of a line of the ACES2 namelist.
- One of the ACES2 namelist lines extends beyond the 80th character.
- There is no title line, and the first line of the Z-matrix has been assumed to be the title.
- There were files from a previous ACES II calculation in the workspace. In general, one should clear the workspace of all previous ACES II files prior to copying in the new files.

IV. Some Special Topics and Utilities

A. The GENBAS File

A..1 The format of the GENBAS file

The GENBAS file contains a description of each of the basis sets that are available to the user. Users may easily add additional basis sets to their GENBAS files so a description of the format is given here.

Line #	Fortran format	Description
1	A80	blank line
2	A80	name of the basis set
3	A80	comment line
4	A80	blank line
5	I3	the number of shells in the basis set (NS)
6	NS(I5)	angular momentum for each shell (L)
7	NS(I5)	number of contracted basis functions for each shell (NC)
8	NS(I5)	number of exponents for that shell (NE)
9	A80	blank line
10	NE(F14.7)	exponents for the first shell
11	A80	blank line
12	NC(F10.7,1X)	contraction coefficients for the first shell
13	A80	blank line

It is important that the shells are grouped by angular momentum, with the s shell(s) first, followed by p shell(s), etc. Otherwise, the input file written for the VMOL integral program will be incorrect. Lines 10 through 13 repeat NS number of times. An example of the boron PVTZ basis set entry in the GENBAS file is included below.

B:PVTZ

JFS DUNNING CORRELATION CONSISTENT BASIS FROM FTP

```
4
  0   1   2   3
  4   3   2   1
 10   5   2   1

5473.0000000   820.9000000   186.8000000   52.8300000   17.0800000
```

5.9990000	2.2080000	0.5879000	0.2415000	0.0861000
0.0005550	-0.0001120	0.0000000	0.0000000	
0.0042910	-0.0008680	0.0000000	0.0000000	
0.0219490	-0.0044840	0.0000000	0.0000000	
0.0844410	-0.0176830	0.0000000	0.0000000	
0.2385570	-0.0536390	0.0000000	0.0000000	
0.4350720	-0.1190050	0.0000000	0.0000000	
0.3419550	-0.1658240	0.0000000	0.0000000	
0.0368560	0.1201070	1.0000000	0.0000000	
-0.0095450	0.5959810	0.0000000	0.0000000	
0.0023680	0.4110210	0.0000000	1.0000000	
12.0500000	2.6130000	0.7475000	0.2385000	0.0769800
0.0131180	0.0000000	0.0000000		
0.0798960	0.0000000	0.0000000		
0.2772750	0.0000000	0.0000000		
0.5042700	1.0000000	0.0000000		
0.3536800	0.0000000	1.0000000		
0.6610000	0.1990000			
1.0000000	0.0000000			
0.0000000	1.0000000			
0.4900000				
1.0000000				

A..2 The contents of the GENBAS file

The following is a listing of the basis sets currently included in the standard GENBAS file. Included are the atoms for which the sets are available and miscellaneous other comments.

STO-3G This is the well known minimal basis set developed by Pople and coworkers in the early 1970s. entries are in the GENBAS file for all atoms from H to Cl. STO-3G basis sets are available in the literature for the third and fourth row main group elements as well as some transition metal elements. The STO-3G basis is now largely obsolete, except for some calculations on large molecules. Its use is not recommended for other than testing and rough preliminary investigations.

3-21G This is a small “split valence” basis set developed by Pople and coworkers. It uses a minimal basis (or single zeta) description for the core orbitals and a double-zeta description for the valence orbitals. Entries are in the GENBAS file for H-Cl. 3-21G sets are available in the literature for heavier elements. This and other sets of this type (such as 4-31G and 6-31G) are often termed “double zeta valence”. This is not strictly accurate since the s and p exponents are constrained to be equal, which they are not in a true double zeta set, although the two sets give results of similar quality.

4-31G This is similar to the 3-21G set, but with more primitive gaussian functions. Entries are available for H-Cl.

6-31G This is yet another split valence set, employing still more primitive gaussians. Entries are available for H-Cl.

6-311G This is a split valence set with a triple-zeta description of the valence orbitals and a minimal basis set description of the core orbitals. Entries are available for H-Ne. It has been claimed in the literature that this basis set is not really of triple-zeta valence quality.

6-31G** 6-31G supplemented with d polarization functions (p functions for H and He). It should be used with the SPHERICAL=OFF option (i.e. $6d$ functions) since this is how the set was defined and developed. Entries are available for H-Cl. The 6-31G* basis set excludes p functions from H and He, while retaining d functions on other atoms. Polarization exponents are as follows: H, He (1.1); Li (0.2); Be (0.4); B (0.6); C-Ne (0.8); Na, Mg (0.175); Al (0.325); Si (0.45); P (0.55); S (0.65); Cl (0.75).

6-311G** 6-311G supplemented with polarization functions. This should be used with the SPHERICAL=ON option (i.e. $5d$ functions). Entries are available for H-Ne. Polarization exponents are: H, He (0.75); Li (0.2); Be (0.401); C (0.626); N (0.913); O (1.292); F (1.75); Ne (2.304). This basis set was developed for correlated calculations.

DZ This is the well known Dunning double-zeta contraction of Huzinaga’s $9s5p$ primitive gaussian basis set for first row atoms. Entries are available for H, B-F.

- DZP This is the DZ set augmented with the polarization functions recommended by Redmon, Purvis, and Bartlett, which were determined from correlated calculations. Entries are available for H, B-F. The polarization exponents are: H (0.7); B (0.386); C (0.654); N (0.902); O (1.211); F (1.580).
- D95 This is the same as DZ for H, B-F but also includes $6s4p$ contractions by Dunning and Hay of Huzinaga's $11s7p$ primitive set for Al, Si, P, S, and Cl. The published data for Cl are erroneous, having at least two typographical errors (see W.Craven et al, Chem. Phys. Lett. **116**, 119 (1985)). It is not clear if these are the only errors.
- D95* This the D95 set augmented with a set of polarization exponents of uncertain origin. Entries are available for H, B-F, and Al-Cl. The polarization exponents for the first row elements appear to be those recommended by Dunning and Hay, while the source of the second row d exponents is uncertain. The exponents are: H (1.0); B (0.7); C (0.75); N (0.8); O (0.85); F (0.9); Al (0.25); Si (0.3247); P (0.37); S (0.532); Cl (0.6).
- TZ2P For first row elements this set comprises Dunning's $5s3p$ contraction of Huzinaga's $10s6p$ primitive set augmented with two optimized d (p for H) functions in a (2,1) contraction of three primitives. Entries are available for H, B-F. The F basis is not optimized. There is also an entry for Cl under this name. This is based on a McLean and Chandler sp set.
- PVDZ Dunning's polarized valence double-zeta correlation consistent basis set. In terms of contracted functions this set is $3s2p1d$ for second row atoms, with one fewer shell of each angular momentum for H and He, and one extra shell plus an f function for Na-Cl. These and the PVTZ and PVQZ sets are hybrids between segmented and generally contracted sets. All of these sets appear to be best used with the SPHERICAL=ON option. Indeed, whether SPHERICAL is ON or OFF has a significant effect on the results. Entries are available for H, He, B-Ne, and Al-Ar. We do not recommend the PVDZ set. For little extra cost one may use the DZP basis set and obtain significantly improved results.
- PVTZ Dunning's polarized valence triple-zeta correlation consistent basis ($4s3p2d1f$ for second row atoms). Entries are available for H, He, B-Ne, Al-Ar.
- PVQZ Dunning's polarized quadruple-zeta valence correlation consistent basis set ($5s4p3d2f1g$ for second row atoms). Entries are available for H,He, B-Ne, and Al-Ar.
- PV5Z Dunning's polarized pentuple-zeta valence correlation consistent basis set ($6s5p4d3f2g1h$ for second-row atoms). Entries are available for H, B-F, Al-Ar.
- WMR Generally contracted basis functions developed by Widmark, Malmqvist and Roos for the study of molecular and atomic properties. These are rather large basis sets [$6s4p3d$ for H, $7s4p3d$ for He, $7s6p4d3f$ for second row atoms], and can be reduced to normal size through use of the GENBAS_X keywords. A limited amount of experience with

these basis sets suggests that valence double-zeta and valence triple-zeta contractions [$3s2p1d$ and $4s3p2d1f$ for first row atoms, $2s1p$ for H and He] work reasonably well. For a given level of contraction, these basis sets appear to provide superior molecular structures and properties to the correlation consistent sets of Dunning. Entries are available for H-Ne.

PBS These are double-zeta plus diffuse basis sets developed by Sadlej for the calculation of electrical properties. They seem to do a good job of predicting dipole moments and polarizabilities and are also useful in excited state calculations, where the first member of a Rydberg series is usually recovered.

TZP This is a triple-zeta valence plus polarization basis set. Entries are currently available for H,B-Ne,Na, and Mg. For H it comprises Dunning's $3s$ contraction of Huzinaga's $5s$ primitive set augmented with the p exponent of Redmon, Purvis, and Bartlett (0.7). For B-F it comprises Dunning's $5s3p$ contraction of Huzinaga's $10s6p$ primitive set augmented with the polarization exponents of Redmon, Purvis, and Bartlett (see entry DZP above). For Ne the basis set is the Dunning $5s3p$ set augmented with a polarization exponent of 1.9 (an estimate based on the values of Redmon et al). For Na and Mg the basis set is $6s5p1d$, with the sp part coming from McLean and Chandler and d exponents of 0.1 and 0.2 (reasonable estimates).

$5s4p1d$ As TZP but the sp part Dunning's $5s4p$ contraction of Huzinaga's $10s6p$ primitive. Entries are available for B-Ne.

VDZP This is a valence double-zeta plus polarization basis set for all first row atoms except He, taken from Dunning and Hay. It has the advantage that sets for Li and Be exist, in contrast to the DZP set. However, it has not been well tested. The polarization exponents are those of Redmon et al for H,B-F while estimates of 0.2, 0.3, and 1.9 are used for Li, Be, and F.

svp,dzp,tzp,tzplarge,qz2p,... These are the new basis sets from Schäfer, Horn and Ahlrichs which have been optimized for atoms and supplemented by suitable polarization functions. Note that these basis sets are denoted in the GENBAS file by lower case letters. They are recommended, in particular, for chemical shift calculations and should in the long run replace the old and not completely optimized Dunning-Huzinaga basis sets (denoted by DZP and TZ2P) in the GENBAS file. Basis sets are, in principle, available for all atoms. If a needed basis set is not included in the GENBAS file, it can be obtained via FTP.

B. The use of effective core potentials

B.1 Specification in the ZMAT file

The use of effective core potentials is specified in the ZMAT file by the keyword ECP=ON. The keyword BASIS must be set to BASIS=SPECIAL. The ecp nicknames are listed below the last basis set (separated by a blank line) using the same format as the non-standard basis set specification, XX:ECPNAM. As for the GENBAS file, the ECPDATA file may be searched for XX, where XX is the atomic symbol, in order to see what ECP's are available for that atom. Atoms included in the ZMAT file without an ecp parameter set are marked by the nickname NONE (see example below). The following ZMAT demonstrates the specifications:

```
CRF6 SINGLE POINT ENERGY CALCULATION USING AN ECP FOR CR
CR
```

```
F 1 RMC
F 1 RMC 2 W1
F 1 RMC 3 W1 2 T1
F 1 RMC 4 W1 3 T2
F 1 RMC 5 W1 4 T1
F 1 RMC 6 W1 5 T2
```

```
RMC=1.676125
W1=90.
T1=90.
T2=270.
```

```
*ACES2(SYMMETRY=ON,BASIS=SPECIAL,ECP=ON,CALCLEVEL=SCF
OCC=10-6-6-2-6-2-2-0,SPHERICAL=ON)
```

```
CR:ECP-10 DZP
F:DZP
F:DZP
F:DZP
F:DZP
F:DZP
F:DZP
```

```
CR:ECP-10-MDF
F:NONE
F:NONE
F:NONE
F:NONE
F:NONE
```

F: NONE

B..2 The format of the ECPDATA file

The parameters of the effective core potentials are specified in the file ECPDATA in the same way as the basis set are gathered in GENBAS. As an example, the entry for copper is given below.

```
*
CU:ECP-10-SK
# ECP BY STEVENS/KRAUSS FOR CU - 10 CORE ELECTRONS - LMAX = 2
*
      NCORE = 10      LMAX = 2
d
-10.00000000      1  511.9951763
-72.55482820      2   93.2801074
-12.74502310      2   23.2206669
s-d
   3.00000000      0  173.1180854
  23.83518250      1  185.2419886
 473.89304880      2   73.1517847
 157.63458230      2   14.6884157
p-d
   5.00000000      0  100.7191369
   6.49909360      1  130.8345665
 351.46053950      2   53.8683720
   85.50160360      2   14.0989469
*
CU:ECP-18-SK
# ECP BY STEVENS/KRAUSS FOR CU - 18 CORE ELECTRONS - LMAX = 3
*
      NCORE = 18      LMAX = 3
f
-18.00000000      1  359.2137111
-119.92593970      2   67.5347369
-29.55328670      2   14.7222923
-10.28924330      2    3.9975558
  -78363630      2    1.1889410
s-f
   3.00000000      0   19.6202650
 20.15792750      1    5.1604389
```

```

    34.50019060    2    1.2306099
   -18.98120030    2    1.0850105
p-f
    5.00000000    0    31.9385762
   20.60853280    1    14.9202125
   56.00168880    2    15.6835232
   57.21701070    2    4.9311614
    7.71778780    2    1.0622167
d-f
    .25986160    2    5.1159991
   -.46216800    2    .7396784
*
```

The first line contains a single star. The name of the data group starts with the element symbol followed by the ECP nickname. It is useful to give a comment introduced by the #-symbol in the line below in order to indicate the origin of the ECP. The actual ECP data is given in between two lines with a *-symbol. The first line specifies the number of core electrons described by the ECP (NCORE) and the maximum angular momentum number of the projector operators (LMAX) in integers i.e. s=0, p=1, d=2 These are followed by the description of the effective core potential which consists of the angular momentum numbers and by the analytical representation of the operator. The latter includes the coefficient c_m , the exponent N_m of r and the exponent α_m of the gaussian:

$$U_l(r) = \sum_m c_m e^{\alpha_m r^2} r^{N_m}.$$

For a detailed description see, for example, L.R. Kahn, P. Baybutt and D.G. Truhlar, *J. Chem. Phys.* **65**, 3826 (1976).

C. Harmonic Vibrational Frequency Calculations with Nonstandard Atomic Masses

ACES II provides a straightforward way to calculate changes in harmonic vibrational frequencies and infrared intensities due to isotopic substitutions. In order to do this, users need to create a file called ISOMASS which contains the desired atomic masses. The file is read free-format; the only restriction is that the atomic masses must appear in the same order as the declaration of the atoms in the ZMAT file, *with dummy centers excluded*. As an example, suppose one wanted to calculate the ^{16}O – ^{18}O isotopic shift for the vibrational frequencies of water with the following ZMAT file

```
Water frequency calculation
X
O 1 R
H 2 R1 1 A
H 2 R1 1 A 3 T

R=1.0
R1=0.95
A=130.
T=180.

*ACES2(CALC=SCF,BASIS=DZP,VIB=EXACT)
```

In order to do this, one would use the following ISOMASS file :

```
18.0
1.00797
1.00797
```

where it should be noted that the position of the dummy center has been ignored.

Once an appropriate ISOMASS file has been constructed, the new set of frequencies and intensities can be obtained by *direct* invocation of the ACES II executable **XJODA** (*i.e.* not the **XACES2** executable). In addition, it is necessary to have the JOBARC and JAINDX direct access files from the frequency calculation in the same directory. The simplest way to do this sort of calculation is through use of nonstandard file declarations, as described in detail in the Section III C of this manual.

D. Use of Force Constant Matrices in Geometry Optimizations

All of the geometry optimization algorithms incorporated into ACES II are based on the Newton-Raphson method, in which step directions and sizes are related to the first and second derivatives of the molecular potential energy. However, in almost all practical calculations the exact second derivative matrix is not evaluated but rather approximated. As the calculation progresses, well-established numerical methods are used to estimate the elements of the Hessian matrix based on all previous optimization steps. After a large number of steps have been taken, one may safely assume that the totally symmetric Hessian used to form the step is a reasonable approximation to the correct Hessian. However, in the early stages of the optimization there is not a sufficient amount of available information to accurately estimate the Hessian and problems may ensue. By default, ACES II geometry optimizations begin with a very crude estimate of the Hessian in which all force constants for bonded interactions (as specified by the Z-matrix connectivity) are set to 1 hartree/bohr², all bending force constants (corresponding to bond angles in the ZMAT file) are set to 0.25 hartree/bohr² and all torsional force constants are set to 0.10 hartree/bohr². While this initialization and the subsequent numerical updates work satisfactorily for most small molecules, there can be occasional problems. In these cases, one might wish to use an alternative initial force constant matrix, particularly one obtained by ACES II at the same or another level of theory. There are a number of ways in which one might do this. First, the EVAL_HESS keyword can be used. If nonzero, the value associated with this keyword directs ACES II to calculate the SCF Hessian matrix prior to the first optimization step and then every N steps thereafter, where N is the value of EVAL_HESS. By setting N to a sufficiently large value (larger than OPT_MAXCYC), then the Hessian will never be recalculated and the optimization will begin with the SCF Hessian.

However, the strategy based on EVAL_HESS is not sufficient for all purposes. For example, one might wish to use a Hessian which is evaluated at the correlated level. This is not possible with EVAL_HESS, since it will direct ACES II to calculate only the SCF Hessian, regardless of the calculation type specified in the ZMAT file. Alternatively, one might adopt the economical strategy of using a Hessian which is evaluated at a low level of theory, such as SCF with the STO-3G basis set. This is the recommended approach for transition state searches and all optimizations for which the default Hessian is inadequate. In any event, it is quite straightforward to use another set of force constants to begin an optimization. First, one simply runs a harmonic frequency calculation at the desired level of theory and saves the FCMINT file. This formatted file contains the full internal coordinate force constant matrix. When the XJODA executable performs a geometry optimization, it checks the active working area for the presence of an FCMINT file. If the file is available, XJODA uses the contents of the file to initialize the Hessian matrix. Hence, one only needs to place the FCMINT file in the same directory as the ZMAT and GENBAS files and the force constants will be read automatically.

While the geometry (and even the point group symmetry) specified by the ZMAT file in the harmonic frequency calculation need not be the same as that used in the first step of the geometry optimization, *the Z-matrix connectivity must be identical*. If one attempts

to use an entirely different Z-matrix, then the definitions of internal coordinates are no longer the same and chaos may ensue. While this point may seem to be unimportant, this situation occurs relatively frequently. Suppose you assume that the equilibrium geometry for a transition state is planar and locate a stationary point with METHOD=TS. However, when the harmonic frequencies are calculated, two modes are found to have imaginary frequencies: an a' mode (in-plane) and an a'' mode, with the a' mode corresponding (approximately) to the reaction coordinate of interest. As a result, the true transition state geometry does not contain a plane of symmetry and another search must be performed in a reduced symmetry. To this end, it would certainly be useful to use the FCMINT file obtained in the frequency calculation to start the search, but one must be careful that the ZMAT file used in the reduced symmetry has *exactly* the same connectivity scheme as that used in the search for the planar structure! As a general rule of thumb, you should always save FCMINT files when you do frequency calculations. Keep a library of these files, as they may be of some use in the future.

E. Initial Guessing and SCF Convergence Options.

It is often convenient to be able to use a set of MOs from one calculation as an initial guess for another (with the same basis set). There are some facilities for doing this with the ACES II program system. The first way is to reuse the JOBARC and JAINDX files and the RESET keyword. For example, suppose we wish to run an ROHF calculation but the default ROHF procedure does not converge or converges to the wrong electronic state. We might try to start the ROHF calculation from a converged set of UHF orbitals. We first run a UHF SCF calculation and do not delete any of the files created by ACES II. We then edit the ZMAT file, replacing REF=UHF by REF=ROHF and including the string RESET=ON. Next we run the **xjoda** executable. This resets the option values. We then run the **xvscf** module. This will perform an ROHF SCF calculation, starting with the UHF vectors. This procedure is somewhat limited. In particular, one must keep JOBARC and JAINDX files, and there is no way of doing such things as swapping orbitals to get the right electronic state. A more flexible procedure is to use the GUESS file, as will now be described.

The usage of the GUESS file in general terms will be illustrated by considering how we might use a set of UHF orbitals as the initial guess for an ROHF calculation, but without having JOBARC and JAINDX files from the UHF calculation. First, we run a UHF calculation. The SCF writes the UHF orbitals to a small formatted file called NEWMOS. These are expressed in terms of the symmetry adapted basis functions, and are printed by spin and symmetry block (*i.e.* as in the output file). To run an ROHF calculation starting from these UHF orbitals, we first rename or copy NEWMOS to another file. This file must be copied to the work area as OLD MOS when we run the ROHF calculation. We then create a GUESS file, giving the occupation (this overrides the occupation keyword in ZMAT) and instructions to read an initial set of orbitals from OLD MOS (see below for GUESS file options). This file must be copied to the work area as GUESS when we run the ROHF calculation. We then change REF=UHF to REF=ROHF in the ZMAT file. We then are ready to run the ROHF calculation. We copy in GENBAS, ZMAT, OLD MOS, and GUESS and then run the program. Simple scripts for the calculations could be as follows :

First, the UHF job :

```
#
cd /tmp/osu937/jdw
rm *
cp /d/osu937/jdw/craps/tests/h2odzp.zmat      ZMAT
cp /d/osu937/jdw/basis/GENBAS                GENBAS
xaces2 > /d/osu937/jdw/craps/tests/h2o.out_uhf
cp NEWMOS /d/osu937/jdw/craps/tests/h2ouhf.oldmos
```

Then the ROHF job :

```
#
cd /tmp/osu937/jdw
```

```

rm *
cp /d/osu937/jdw/craps/tests/h2odzp.zmat      ZMAT
cp /d/osu937/jdw/basis/GENBAS                GENBAS
cp /d/osu937/jdw/craps/tests/h2odzp.gss      GUESS
cp /d/osu937/jdw/craps/tests/h2ouhf.oldmos    OLDMOS
xaces2 > /d/osu937/jdw/craps/tests/h2o.out_roh
cp NEWMOS /d/osu937/jdw/craps/tests/h2orohf.oldmos

```

Evidently, the OLD MOS file will be much smaller than JOBARC, and it is a formatted file. It is therefore easier to store sets of MOs which may be of use in a future calculation, to move sets of MOs from one computer system to another, and to use MOs from other program systems. Regarding the latter point, one must of course have the appropriate ordering of basis functions and coefficients in terms of symmetry adapted basis functions.

We now consider the format of the GUESS file. The following is an example of the GUESS file for a state of the water cation.

```

H2O. TEST.
 3 1 1 0      alpha occ
 3 0 1 0      beta occ
 0 0 0 0      alpha pairs to be swapped in each irrep
 0 0 0 0
 0 0 0 0
 0 0 0 0
 0 0 0 0      beta pairs to be swapped in each irrep
 0 0 0 0
 0 0 0 0
 0 0 0 0
 0 0 0 0      alpha locking within each irrep
 0 0 0 0      beta locking within each irrep
 0 0 0 0      alpha printing of initial guess
 0 0 0 0      beta printing of initial guess
 0 0          stopping parameters
 1 0          reading and writing mos
 1            nonzero if a uhf wfn is to be got from rhf
 0            nonzero if GUESS is always to be read

```

The file is a formatted file. The words on the right are not part of the input and are unnecessary, but have been included as a reminder of the options. The input is as follows. NIRREP is the order of the computational point group (4 in the present case).

Line 1 : the title, up to 80 characters

Line 2 : alpha occupation, format (NIRREP I3)

Line 3 : beta occupation, format (NIRREP I3) (omitted if RHF closed-shell)

Line 4 : pairs of orbitals to be swapped in alpha symmetry block 1 (The format is 4I3 regardless of value of NIRREP). Two numbers are needed to specify each pair. Therefore no more than two interchanges may be made for a given symmetry block and spin

Line 5 : as line 4, but the numbers refer to the second symmetry block

Line 6 : as line 4, but the numbers refer to the third symmetry block

Line 7 : as line 4, but the numbers refer to the fourth symmetry block

Lines 8-11 : as Lines 4-7 but for beta spin (omitted if RHF)

Line 12 : (NIRREP I3), specify if any alpha symmetry blocks are to be "locked" (i.e. orbital occupation proceeds in direction of "minimum change", by monitoring $C(OLD)T * S * C$). Put a zero if no locking and a positive integer if you wish a particular block to be locked.

Line 13 : as for Line 12, but for beta spin (omitted if closed-shell RHF)

Line 14 : (NIRREP I3), determine if alpha initial guess is to be printed. Put a 0 if not, put a positive integer if you wish a particular symmetry block to be printed.

Line 15 : as previous line, but for beta spin (omitted if closed-shell RHF)

Line 16 : (2I3) stopping parameters. Set first to a positive integer if you wish the SCF to stop after computing

the initial guess.

Line 17 : (2I3) parameters READMO and WRITMO. If READMO is set to a positive integer then initial guess MOS are to be read from a file called OLD MOS. If WRITMO is set to a positive integer then the final converged MOS are to be written to a file called NEW MOS. the two files have the same format (and are formatted rather than binary). Hence, NEW MOS from one job may be used as OLD MOS in a subsequent job. For no reading or writing, set both parameters to 0. WRITMO is now redundant, NEW MOS is always written.

Line 18 : (I3) a parameter which specifies if a UHF guess is to be generated from one or two sets of MOS. This is only meaningful if READMO has been set to positive integer. this option lets one, for example, start a a UHF calculation with an RHF closed-shell set of orbitals. set to a positive integer for reading just one set of orbitals. set to 0 for reading both sets from OLD MOS.

Line 19 : (I3) a parameter which specifies if initial guess instructions are to be read from GUESS every time an SCF calculation is performed during a job, or just the first time. Set to 0 for just the first time. otherwise set to positive integer. If it is set to 0, GUESS is deleted after it has been read. One would usually want to do this in a geometry optimization, for example, the second and subsequent points reading the MOs on JOBARC from the previous point.

It should be noted that one must specify all options in the GUESS file — *i.e.* there are no defaults.

We turn now to the subject of SCF convergence options. Within ACES II a number of options are available. By default the first few iterations proceed by repeated diagonalization of appropriate Fock matrices. Once either a certain number of iterations (see keyword RPP_LATEST) have been performed or an initial convergence criterion has been met, the DIIS convergence extrapolation procedure of Pulay begins. If convergence difficulties are

experienced with the default scheme, one has a number of options. In RHF closed-shell and UHF open-shell calculations, difficult cases will often converge with the use of a dynamical damping algorithm due to E.R. Davidson. This is achieved through the option DAMP_TYP=DAVIDSON. Damping serves to prevent excessively large oscillations in the early iterations. Once the SCF convergence appears to be sufficiently smooth (the damp factor is smaller than DAMP_TOL and the energy difference is sufficiently small), the program reverts to repeated diagonalization and DIIS extrapolation. For ROHF calculations, one can also use the damping algorithm, but in addition the level-shifting technique (M.F. Guest and V.R. Saunders, *Mol. Phys.* **28**, 819 (1974)) is particularly useful. In this scheme one adds a positive number α to all diagonal elements of the singly occupied orbital block of the Fock matrix and a positive number $(\alpha + \beta)$ to the diagonal elements of the virtual orbital block of the MO basis Fock matrix. α and β are set by the LSHF_A1 and LSHF_B1. If one wishes to use a value 0.2 a.u. for the level-shifters, LSHF_A1 and LSHF_B1 should be set to 20. This is a reasonable value for most systems. Larger values of level shifters are sometimes necessary for transition metal systems, especially when the singly occupied orbitals lie below some of the double occupied orbitals. An example is provided by the following FeCl_4^- input deck (excitation energies of this system were recently studied by N. Oliphant and R.J. Bartlett, *J. Am. Chem. Soc.* **116**, 4091 (1994)) :

```
FeCl4 sextet
```

```
CL 0.0000 -3.4826 2.2358
FE 0.0000 0.0000 0.0000
CL 0.0000 3.4826 2.2358
CL 3.4826 0.0000 -2.2358
CL -3.4826 0.0000 -2.2358
```

```
*ACES2(REF=ROHF,PRINT=1,CALC=SCF,MULT=6,CHARGE=-1
      BASIS=SPECIAL,UNITS=BOHR,COORDINATES=CARTESIAN
      RLE=NOJACOBI,SPHERICAL=ON,ECP=ON
      DAMP_TYP=DAVIDSON,DAMP_TOL=5,LSHF_A1=50
      LSHF_B1=50
      OCCUPATION=7-6-6-6/5-5-5-5)
```

```
CL:10-MWB3S3P
FE:ECP-10-MDF-F
CL:10-MWB3S3P
CL:10-MWB3S3P
CL:10-MWB3S3P
```

```
CL:ECP-10-MWB
FE:ECP-10-MDF
CL:ECP-10-MWB
CL:ECP-10-MWB
```

CL:ECP-10-MWB

Another tip for ROHF convergence, which we strongly recommend, is to start from orbitals from a closely related closed-shell system.

F. Equation-of-motion coupled-cluster electron attachment calculations.

EOM-CC electron attachment calculations yield energy differences between an N electron reference state and one or more electronic states of the (N+1) electron system obtained by adding an electron. The keywords such as REFERENCE, CHARGE, MULTIPLICITY, and OCCUPATION define the electronic state of the N electron system. If EA_CALC is set to EA_EOMCC, then energies of (N+1) electron states are calculated. The states are specified by the EA_SYM keyword as a string of NIRREP (REFERENCE=RHF) or 2*NIRREP (REFERENCE=UHF or ROHF) integers, where NIRREP is the number of irreducible representations in the computational point group. The string of numbers specifies the numbers of (N+1) electron states of a given symmetry and the spin of the additional electron in each (N+1) electron state. For closed-shell systems only the alpha-roots have to be specified, e.g. EA_SYM=3-2-0-2. For open-shell systems one can either attach an electron of alpha spin or one of beta spin, leading to different states of the (N+1)-electron system. The different spin blocks are separated by a slash ('/'), e.g. EA_SYM=3-2-0-2/0-1-0-4. This keyword does not have to be specified in an EA-EOMCC calculation. If EA_SYM is not specified but EA_CALC=EA_EOMCC, the program tries to find the ground state of the (N+1)-electron system internally.

Basically we can recommend three types of applications of the EA-EOMCC program.

a) The calculation of electron affinities. Only EA_CALC=EA_EOMCC needs to be specified. If it is known what the symmetry of the ground state is for the (N+1)-electron system, specify also EA_SYM.

The following input yields the electron affinity of the sodium atom.

```
NA atom
NA

*ACES2(REFERENCE=UHF,CALC=CCSD,BASIS=DZP
MULTIPLCTY=2,SPHERICAL=0N,EA_CALC=EA_EOMCC)
```

In addition the keyword EA_SYM=0-0-0-0-0-0-0/1-0-0-0-0-0-0 might have been specified such that only the closed-shell $3s^2$ state of the sodium anion is calculated, and no other possibilities are considered for the symmetry of the anion ground state.

b) The calculation of excitation spectra for systems with an odd number of electrons. Take as a reference a closed-shell configuration of the system with one less electron. Specify EA_SYM to obtain a number of roots of desired symmetry. The excitation spectrum is then calculated.

The following example specifies the input for calculation of the excitation spectrum of MgF. The two core-orbitals are excluded from the correlation treatment (Both CCSD and EA_EOMCC). 5 roots are found in symmetry block 1 (Σ and Δ symmetries), 3 in block 2 (Π symmetry) and one in block 4.

```
MgF Excitation Spectrum
MG
F 1 R
```

```
R = 1.752
```

```
*ACES2(REFERENCE=RHF,CALC=CCSD,BASIS=TZ2P,CHARGE=1
MULTIPLCTY=1,SPHERICAL=ON,DROPMO=1-2,EA_CALC=EA_EOMCC
EA_SYM=5-3-0-1)
```

c) The calculation of high spin triplet states for systems with a closed-shell ground state. Take as a CC reference the high spin doublet ground state of the positive ion, and add an extra alpha electron by specifying EA_SYM, e.g. EA_SYM=4-3-0-2/0-0-0-0. This yields high spin triplet excited states of the neutral. In addition the closed-shell ground state can be obtained by adding a beta electron to the proper symmetry-block, e.g. EA_SYM=4-3-0-2/1-0-0-0. This has the advantage that the proper excitation energies of the system are tabulated by the program. We note that singlet and low spin triplet excited state energies can also be obtained by adding a beta electron. However such calculations do not yield satisfactory results, due to spin contamination of the resulting EA_EOMCC states.

The following input yields triplet excited states for the beryllium atom. The SCF calculation is on the closed-shell neutral system, while the QRHF option is used to create the positive Be ion.

```
BE Atom Excitation spectrum, QRHF Reference
BE
```

```
*ACES2(REFERENCE=RHF,CALC=CCSD,BASIS=SPECIAL
SPHERICAL=ON,QRHF_G=-1,EA_CALC=EA_EOMCC
EA_SYM=8-4-0-4-0-0-0-0/0-0-0-0-0-0-0-0)
```

```
BE:WMR
```

WARNING: In all EOMCC calculations it is highly recommended that the reference state transforms according to a one-dimensional representation of the true molecular point group. Otherwise the likely outcome is inaccurate results, which in addition are very hard to interpret due to a breaking of the symmetry in such a calculation.

G. Calculation of NMR Chemical Shifts

NMR spectroscopy is a very important analytical tool for the identification and characterization of molecules. However, since there is no simple correlation between the measured chemical shifts and structural parameters, the interpretation of experimental NMR spectra is not trivial and can be in many cases quite involved. The ability to calculate NMR chemical shifts *ab initio* is therefore a very important advancement in quantum chemistry. The calculation of chemical shifts can provide in many cases the necessary information for the correct interpretation of experimental NMR spectra. Therefore, methods for the computation of NMR chemical shifts at SCF and correlated levels (currently limited to MBPT(2)) have been added in the last years to the functionalities of the ACES II program system.

However, before discussing the details of the calculation of NMR chemical shifts, a few general remarks are required. The main problem in all calculations of magnetic properties (i.e., NMR chemical shifts and magnetizabilities) using finite basis sets (as they are usually employed in quantum chemical calculations) is the gauge-invariance problem. This simply means that the results of such a calculation depend on the chosen gauge-origin and are not invariant with respect to gauge transformations as required by exact theory. A trivial solution to the gauge-invariance problem would be the use of very large basis sets in order to minimize the gauge error, but this approach, due to large computational costs, is limited to small molecules. More satisfying solutions are offered by approaches which introduce local gauge-origins to define the vector potential such as the IGLO method of Kutzelnigg and Schindler, the LORG method of Hansen and Bouman and the GIAO-approach of Ditchfield. The latter originates in London's work on molecular diamagnetism in the thirties and was first used by Hameka during the sixties to calculate magnetizabilities and chemical shieldings. ACES II incorporates the GIAO-SCF and GIAO-MBPT(2) method for calculating chemical shifts, since in our opinion, the GIAO approach is the most elegant solution to the gauge-invariance problem and in contrast to the IGLO method, is easily extended to correlated approaches (as for example in the GIAO-MBPT(2) method).

The calculation of NMR chemical shifts are invoked via the keyword PROP=NMR together with the appropriate specification of the quantum chemical method (CALC=SCF gives then GIAO-SCF, CALC=MBPT[2] gives GIAO-MBPT(2)). In principle, no other option is required to run calculations of NMR chemical shifts. However, to ensure the success of GIAO-MBPT(2) calculations and in particular large-scale calculations with the GIAO-MBPT(2) method, the computational requirements of such calculations should be kept in mind. While CPU requirements are of less interest (a GIAO-MBPT(2) calculation is in terms of the CPU usually less expensive than the corresponding MBPT(2) geometry optimization), memory and disk space requirements are of special concern. The memory requirements are for the **xnmr** module approximately $2 * n^2 N^2 / h^2$ eight byte words with n denoting the number of occupied orbitals, N denoting the number of virtual orbitals, and h specifying the order of the molecular point group. However, the current memory bottleneck is the integral sorting in the module **xintprc** which requires roughly $2 * n^2 N^2 / h$ eight byte words and which therefore is more demanding. The disk space requirement of a GIAO-MBPT(2) calculation is for most parts determined by the fact that the current version depends on the

storage of the GIAO integrals. To summarize the necessary resources, a GIAO-MBPT(2) calculations requires the storage of the AO two-electron integrals (files IIII, IIJJ, IJIJ, and IJKL depending on symmetry and with a total size of approximately $1.5 * N_{basis}^4 / (8h)$ eight byte words (note that the CRAY version requires $2 * N_{basis}^4 / (8h)$ words), the storage of the MO integrals (file MOINTS, the largest portion is here given by the integrals $\langle ab || ci \rangle$ with a total size nN^3/h eight byte words), the storage of the GIAO integrals (files IIIIX, IIJJX, IJIJX, IJKLX in case of a symmetric perturbation B_x or IIIJX, IJIKX, IJKLX for a non-symmetric perturbation B_x). The files for the perturbations B_y and B_z are named using the same convention with Y or Z instead of X. The total size of these files is for each perturbation approximately given by $1.5 * N_{basis}^4 / (4h)$ (CRAY version : $2 * N_{basis}^4 / (4h)$), and the storage of the derivative MO integrals and amplitudes (files DERGAM and DERINT with the largest portion given by the perturbed $\langle ab || ci \rangle$ integrals (size nN^3/h eight byte words)). To keep the disk space requirements at a minimum, it is *strongly recommended* to use the keyword TREAT_PERTURBATION=SEQUENTIAL, as this forces the program to treat each magnetic field component separately and thus requires only the storage of one type of GIAO integrals at one time. TREAT_PERTURBATION=SIMULTANEOUS (which is currently the default) requires the simultaneous storage of all GIAO-integrals and is therefore much more demanding in terms of disk space, though more efficient with respect to CPU timings.

The current limits for GIAO-MBPT(2) calculations depend on the available hardware resources. However, as a rough guide the limits for a IBM RS6000/350 work station with 80 Mbyte memory and 3.5 Gbyte scratch space are given. Within this environment, calculations with about 250 basis functions in D_{2h} symmetry, about 200 basis functions in C_{2v} , C_{2h} , and D_2 symmetry, about 170 basis functions in C_2 , C_i , and C_s -symmetry and about 130 basis functions in cases without symmetry are feasible. From these estimates it is seen that the limits strongly depend on the molecular symmetry. Thus, users are urged to use symmetry whenever possible. It should be also noted that the computational requirements depend somewhat on the ratio n/N and that the costs increases with the number of occupied orbitals.

As the final result the corresponding chemical shielding tensors of all nuclei in the molecule are obtained in a GIAO-SCF and/or GIAO-MBPT(2) calculation. The output produced by the module **xjoda** gives the absolute isotropic shielding (one third of the trace of the shielding tensor) as well as the anisotropic shielding which is usually of less interest. In order to compare with experimental results, the absolute shielding σ must be converted to the relative shifts δ . This is easily accomplished via

$$\delta = \sigma_{ref} - \sigma \quad (1)$$

with σ_{ref} as the absolute shielding of the chosen reference compounds. For ^{13}C , tetramethylsilane (TMS) has been chosen as standard and the corresponding σ values are 198.944 (dzp/dz), 193.103 (tzp/dz), 193.419 (tzp/dzp) at GIAO-SCF level and 205.720 (dzp/dz), 198.890 (tzp/dz), and 197.191 (tzp/dzp) at GIAO-MBPT(2) level. The MBPT(2)/6-31G* optimized geometry has been used in all the GIAO calculations for TMS.

With respect to basis sets, the following recommendations can be made. In case of ^{11}B and ^{13}C - nuclei which have been extensively studied - the dzp/dz basis set (dzp for C

and dz for H) is in most cases sufficient for relative shifts, though the use of the tzp/dz basis (tzp for C and dz for H) is recommended. Larger basis sets (i.e., tzp, tzp2, or even qz2p) are in most cases not needed for the accurate prediction of chemical shifts. On the other side, ^{15}N , ^{17}O , and ^{19}F NMR chemical shift calculations require larger basis sets of at least triple-zeta plus polarization quality. Though qualitatively good results are obtained in many cases with the tzp basis, even larger basis sets such as tz2p or qz2p are recommended for more reliable calculations. For other nuclei, not very much can be said at the moment and the user is strongly urged to check carefully the basis set dependence to ensure reliable theoretical results. However, limited experience suggests that for second-row elements quite large basis sets are needed for accurate calculations.

Note : NMR chemical shift calculations can currently only be performed with Cartesian basis functions, i.e. SPHERICAL=ON may not be specified.

H. Hartree-Fock Wavefunction Stability Analysis

The Hartree-Fock procedure, at convergence, guarantees the resulting wavefunction is a stationary point in the space of orbital rotations (mixing of occupied and virtual orbitals). Though in the majority of cases this stationary point is also a minimum (all orbital rotations increase the energy), it may not always be so. In some cases, the second derivative of the energy with respect to one or more orbital rotations may be zero or negative, indicating rotations which will leave the energy unchanged or lower it. The ACES II program system has the ability to test RHF and UHF wavefunctions for some of the most common instabilities, controlled by the HFSTABILITY and ROT_EVEC keywords.

Using HFSTABILITY=ON will perform a stability analysis after the two-electron integral transformation and processing. (Use of HFSTABILITY=ON is compatible with continuing on to correlated calculations in the same job.) Stability analysis is accomplished by forming the orbital rotation hessian and diagonalizing it. The eigenvalues of this matrix, and their associated eigenvectors indicate the number and type of instabilities present.

Each negative eigenvalue indicates an instability, and their magnitude indicates the severity. Analysis of the eigenvector corresponding to each instability reveals its nature. The direct product of the symmetry irreps of the orbitals involved in the rotation determines the symmetry of the instability. Only for instabilities whose direct product is the totally symmetric irrep (irrep 1) will the wavefunction maintain the symmetry of the molecular framework. Any other result means the instability leads, at least initially, to a symmetry broken wavefunction. In some cases, symmetry breaking instabilities arise from the presence of lower energy electronic states (different occupations), and the rotation specified by the corresponding eigenvector will correspond to changing the occupation to relieve the instability. For an RHF wavefunction, the UHF orbital rotation hessian is constructed, which allows detection of instabilities in which the wavefunction would prefer to be UHF by comparison of the α and β spin eigenvectors.

Zero eigenvalues will occur for degenerate electronic states and merely indicate the equivalence of occupations within the computational point group. Small numerical inaccuracies frequently result in nominally zero eigenvalues having small non-zero values. The sign of such eigenvalues will determine whether or not they are reported as instabilities. Thus you may not see the number of zero (small) eigenvalues you expect.

Other instabilities, such as the wavefunction becoming complex are not tested since complex wavefunctions are not presently supported in ACES II.

In the program output, stability analysis is headed by the label “@RHFSTAB-I” or “@UHFSTAB-I”. The number of instabilities in each irrep is given along with their eigenvalue and classification. For example,

```
@RHFSTAB-I, Performing stability analysis of RHF wavefunction.
Orbital rotation parameters will be evaluated for each symmetry block.
There are 0 instabilities within irrep 1.
There are 1 instabilities within irrep 2.
Eigenvalue = -0.1411211526:
```

```

Instability classification : RHF -> UHF with broken symmetry
There are 1 instabilities within irrep 3.
Eigenvalue = -0.1411211526:
Instability classification : RHF -> UHF with broken symmetry
There are 0 instabilities within irrep 4.
There are 0 instabilities within irrep 5.
There are 2 instabilities within irrep 6.
Eigenvalue = -0.3586792381:
Instability classification : RHF -> UHF with broken symmetry
Eigenvalue = -0.2166464610:
Instability classification : RHF -> RHF with broken symmetry
There are 2 instabilities within irrep 7.
Eigenvalue = -0.3586792381:
Instability classification : RHF -> UHF with broken symmetry
Eigenvalue = -0.2166464610:
Instability classification : RHF -> RHF with broken symmetry
There are 0 instabilities within irrep 8.

```

It is sometimes desirable to obtain solutions corresponding to following instabilities to lower energy stationary points in the orbital rotation space. The keywords HFSTABILITY=FOLLOW and ROT_EVEC are provided to assist with this.

When HFSTABILITY=FOLLOW is set, the stability analysis is performed, as for HFSTABILITY=ON, then the orbital rotation corresponding to the chosen instability is applied to the SCF eigenvectors and the SCF calculation is repeated with these rotated vectors as a starting guess. This is not strictly eigenvector following, nor direct minimization SCF, but in practice, the procedure is quite effective. By default, the lowest eigenvalue of the totally symmetric irrep is followed. Others can be followed by explicitly specifying them with the ROT_EVEC parameter.

Because instabilities in other than the totally symmetric irrep reduce the symmetry of the wavefunction, only those in irrep 1 can be followed. Following other instabilities requires performing the calculation in reduced symmetry. Note that in C_1 symmetry, *all* instabilities will be in the totally symmetric irrep. Likewise, instabilities which take RHF wavefunctions into UHF ones must be followed using a UHF calculation.

Following instabilities often leads to solutions which are lower in energy but lack the symmetry of the molecular framework, are heavily spin contaminated (in the UHF case) or are otherwise non-physical. Instabilities in Hartree-Fock wavefunctions are a very complex problem and must be treated with great care in order to obtain any meaningful information.

Experience has shown that, in general, molecules with instabilities will have more than one and that solutions obtained by following an instability may have further instabilities. It is impossible to predict exactly what will be uncovered by a stability analysis, and the current drivers for ACES II are configured to handle only the simplest cases. Examining a “tree” of instabilities or other specialized procedures will require the user to create an appropriate driver of their own, which is usually specialized to the system under study.

A surprising number of molecules will exhibit instabilities under various conditions. If you encounter one, don't panic. Take a deep breath, go for a walk, get a cup of coffee. Not all instabilities indicate pathological cases or intractable problems.

I. Time-dependent Hartree-Fock Calculations.

In order to provide properties such as frequency dependent polarizabilities, ACES II provides time-dependent Hartree-Fock (TDHF) calculations. The TDHF code can solve the general-order TDHF problem for closed-shell RHF wavefunctions. The TDHF keyword must be set to ON, and additional parameters controlling the TDHF calculation are included in a namelist which is located at the end of the ZMAT file. The variables in the namelist are as follows :

Printing options :

- IOPDA controls density matrix printing (0 means no printing (default) and 1 means print the matrix)
- IOPEV controls MO coefficient printing (0 means no printing (default) and 1 means print the matrix)
- IOPU controls U matrix printing (0 means no printing (default) and 1 means print the matrix)
- IOPFE controls Fock matrix printing (0 means no printing (default) and 1 means print the matrix)
- IOPPR controls property integral printing (0 means no printing (default) and 1 means print the matrix)

For subsequent options, 0 means do not do this type of calculation, otherwise do it. *The default is to do the calculation, i.e.* if a parameter is not specified, the program will perform that calculation.

For the polarizability α :

- IDALPH specifies if α is to be calculated

For the first hyperpolarizability β :

- IOR specifies if optical rectification calculation is to be performed
- IEOPPE specifies if electro-optical Pockels effects calculation is to be performed
- ISHG specifies if second harmonic generation calculation is to be performed

For the second hyperpolarizability γ :

- IOKE specifies if an optical Kerr effect calculation is to be performed
- IDCOR specifies if a DC electric field induced OR calculation is to be performed
- IIDRI specifies if an intensity dependent refractive index calculation is to be performed

- IDCSHG specifies if DC electric field SHG calculation is to be performed
- ITHG specifies if third harmonic generation calculation is to be performed

The method of solution of the TDHF equations is controlled by the parameter NITER (default value is 20). If NITER is 0 an iterative method of solution is used; if it is 1 a non-iterative method of solution is used; if it is greater than 1 the reduced linear equation method will be used.

When NITER=1 more than one frequency can be solved for. For the other methods of solution, only one frequency can be considered in a single calculation. The number of nonzero frequencies is specified by NFREQ. The NFREQ frequencies are listed one per line following the INPUTP namelist.

Static calculations are also performed along with the dynamic calculations. To obtain only static results, all parameters should be set to 0.

The following is an example of a TDHF calculation on N₂ :

```
N2. TDHF. TEST. Sadlej basis set.
```

```
N
```

```
N 1 R
```

```
R=2.07434
```

```
*ACES2(UNITS=BOHR,BASIS=SPECIAL,TDHF=ON)
```

```
N:PBS
```

```
N:PBS
```

```
Hyperpolarizabilities
```

```
$INPUTP IOPU=0,IOPFE=0,IOPEV=0,IOPPR=0,NITER=1,NFREQ=4,IDCSHG=1,IOKE=1,  
IDCOR=1,IIDRI=1,ITHG=1,IWRPA=1 $END
```

```
0.072
```

```
0.0886
```

```
0.0934
```

```
0.0995
```

J. Organization of the ACES II program system

The ACES II program system is set up as a library of various modules which exchange information via several files. For running a job, it is not necessary to know the exact function of these modules, since the **XACES2** module usually takes care that the correct modules are called in the appropriate sequence. However, to give the user some familiarity with the program modules currently included in the ACES II program system, a short description of their tasks will be given in the following:

XACES2 determines the order in which the other executables must be invoked and drives the overall calculation.

XJODA reads the geometry of the molecule and the program options specified by the user. Furthermore, it is the driver of the geometry optimization algorithm implemented in ACES II and performs the update of the geometrical parameters based on analytical gradients and Hessians.

XFINDIF is the driver for a finite difference evaluation of harmonic frequencies and infrared intensities based on analytically evaluated gradients and dipole moments. It is used if VIBRATION=FINDEF_OLD has been specified.

XSVMCOR is the driver for the new finite difference evaluation of harmonic frequencies and infrared intensities based either on single point energy calculations or analytically evaluated gradients and dipole moments. XSVMCOR makes use of symmetry-adapted coordinates and is therefore ideally suited for frequency calculations at correlated levels. It is used if VIBRATION=FINDEF has been specified.

XVMOL evaluates the one- and two-electron integrals over Gaussian basis functions. XVMOL has been written by J. Amlöf and P.R. Taylor and has been interfaced to ACES II. The latest version of XVMOL includes in addition an option for effective core potentials.

XVMOL2JA acts as an interface between XVMOL and the ACES II JOBARC files. Using a knowledge of the internals of XVMOL and information from XJODA, it produces a number of vectors and matrices which relate the “computational” ordering of atoms and basis functions to the order given in the ZMAT file. This data is used by many later ACES II modules during symmetry analysis and other functions.

XVSCF solves the SCF equations for RHF, UHF, and high-spin ROHF wave functions.

XHFDFT calculates the energies for various hybrid HF-DFT methods.

XVTRAN transforms the two-electron integrals from the AO into the MO basis. Both, partial and full transformations are available.

XINTPRC sorts the transformed two-electron integrals and forms their antisymmetrized combinations, the so-called double-bar integrals.

XVCC solves the CC equations and evaluates the correlation energy for the CCD, QCISD, CCSD, CCSDT-n (n=1-4), CCSDT, QCISD(T), CCSD(T), UCC(4), and several other methods. In the case of finite-order perturbation theory (MBPT(2), MBPT(3), SDQ-MBPT(4),

and MBPT(4)) **XVCC** calculates the corresponding contributions to the correlation energy.

XVCC5T calculates the fifth-order triple excitation energy contributions for methods such as CCSD(TQ).

XVCC5Q calculates the fifth-order quadruple excitation energy contributions for methods such as CCSD(TQ).

XVCCEH calculates EOM-CCSD polarizability and NMR spin-spin coupling constants.

XLAMBDA solves the Λ equations to determine the response of the CC amplitudes to a given perturbation. This module is only called in analytical gradient calculations. For finite-order perturbation theory, several higher-order pieces of the wave function as needed for a gradient calculation are computed by this module.

XDENS calculates the one and two-particle density matrices in the MO basis for MBPT and CC methods.

XANTI sorts the two-particle density matrices prior to the back transformation and de-antisymmetrizes them.

XBCKTRN transforms the two-particle density matrices from the MO basis back to the AO basis.

XVDINT is a heavily modified version of the integral derivative program ABACUS written by T. Helgaker, P. Jørgensen, H. Aa. Jensen, and P.R. Taylor, suitable for CC/MBPT gradient calculations. In addition to integral derivatives with respect to geometrical perturbations it calculates one- and two-electron integrals required for chemical shift calculations within the GIAO scheme.

XCPHF solves the coupled-perturbed HF (CPHF) equations either for geometrical displacements, electric or magnetic field components as perturbation.

XVPROP evaluates one-electron integrals needed for the calculation of various first-order properties such as dipole moment, quadrupole moment, electrical field gradients, or spin densities. It originates from POLYATOM and was interfaced to the VMOL integral program by P.R. Taylor.

XDTRAN transforms the integral derivatives from the AO to the MO basis, calculates the derivatives of the excitation amplitudes, and evaluates various contributions to second derivatives at correlated levels.

XSQUINT calculates various contributions which involve two-electron integrals and CPHF coefficients only to second derivatives at correlated levels.

XVEE calculates excitation energies, transition moments and excited state density matrices for TDA EOM-CC methods.

XVEA calculates electron attachment energies by the EOM-CCSD method.

XNMR calculates the paramagnetic contribution to NMR chemical shifts at correlated levels.

XTDHF performs time-dependent Hartree-Fock calculations.

V. Bibliography

Of the many methods currently implemented in ACES II some are well established, while others are new and descriptions have not yet been published. It is the purpose of this section to list pertinent literature references which provide more information about the techniques and their implementations and should be cited when results from ACES II calculations are published. Some references to basis sets included in the program are also included. Guide to Correlated Methods.

- R.J. Bartlett and J.F. Stanton, “Applications of Post-Hartree-Fock Methods : A Tutorial”, in *Reviews of Computational Chemistry* **5**, 65-169, edited by K.B. Lipkowitz and D.B. Boyd (VCH Publishers, New York, 1994).

A. MBPT for molecules

Review article

- R.J.Bartlett, Ann. Rev. Phys. Chem. **32**, 359 (1981).

a) Closed-shell RHF-MBPT for molecules

- R.J.Bartlett and D.M.Silver, Phys. Rev. **A10**, 1927 (1974); J. Chem. Phys. **62**,3258 (1975); *ibid.* **64**, 4578 (1976).
- R.J.Bartlett and I.Shavitt, Chem. Phys. Lett. **50**, 190 (1977).

b) Open-shell UHF-MBPT for molecules

- R.J.Bartlett and G.D.Purvis III, Int. J. Quantum Chem. **14**, 561 (1978).

c) Open-shell ROHF-MBPT for molecules

- W.J.Lauderdale, J.F.Stanton, J.Gauss, J.D.Watts, and R.J.Bartlett, Chem. Phys. Lett. **187**, 21 (1991); J. Chem. Phys. **97**, 6606 (1992).

B. Coupled-cluster (CC) theory

Review article

- R.J.Bartlett, J. Phys. Chem. **93**, 1697 (1989).

a) CCD method

- R.J.Bartlett and G.D.Purvis III, Int. J. Quantum Chem. **14**, 561 (1978).

b) CCSD method

- G.D.Purvis III and R.J.Bartlett, *J. Chem. Phys.* **76**, 1910 (1982).

c) CCSDT-1 method

- Y.S.Lee, S.A.Kucharski, and R.J.Bartlett, *J. Chem. Phys.* **81**, 5906 (1984).

d) CCSD+T(CCSD) method

- M.Urban, J.Noga, S.J.Cole, and R.J.Bartlett, *J. Chem. Phys.* **83**, 4041 (1985).

e) CCSD(T) method (adds one (HF case) or two (non-HF case) small terms to CCSD+T(CCSD))

- K. Raghavachari, G.W. Trucks, J.A. Pople, and M. Head-Gordon, *Chem. Phys. Lett.* **157**, 479 (1989).
- R.J.Bartlett, J.D.Watts, S.A.Kucharski, and J.Noga, *Chem. Phys. Lett.* **165**, 513 (1990).
- J. Gauss, W.J. Lauderdale, J.F. Stanton, J.D. Watts, and R.J. Bartlett, *Chem. Phys. Lett.* **182**, 207 (1991).
- J.D. Watts, J. Gauss, and R.J. Bartlett, *J. Chem. Phys.* **98**, 8718 (1993).

f) CCSDT-2 and CCSDT-3 methods

- J.Noga, R.J.Bartlett, and M.Urban, *Chem. Phys. Lett.* **134**, 146 (1987).

g) CCSDT methods

- J. Noga and R.J. Bartlett, *J. Chem. Phys.* **86**, 7041 (1987).
- G.E. Scuseria and H.F. Schaefer, *Chem. Phys. Lett.* **152**, 382 (1988).
- J.D. Watts and R.J. Bartlett, *J. Chem. Phys.* **93**, 6104 (1990).
- J.D. Watts and R.J. Bartlett, *Int. J. Quant. Chem. Symp.* **27**, 51 (1993).

h) ROHF and QRHF CC methods

- M.Rittby and R.J.Bartlett, *J. Phys. Chem.* **92**, 3033 (1988).

i) Brueckner CC methods

- R.A. Chiles and C.E. Dykstra, *J. Chem. Phys.* **74**, 4544 (1981).
- J.F.Stanton, J.Gauss, and R.J.Bartlett, *J. Chem. Phys.* **97**, 5554 (1992).

j) QCISD and QCISD(T) methods

- J.A. Pople, M. Head-Gordon, and K. Raghavachari *J. Chem. Phys.* **87**, 5968 (1987).

k) UCC methods

- J.D. Watts, G.W. Trucks, and R.J. Bartlett, *Chem. Phys. Lett.* **157**, 359 (1989).

l) CCSD+TQ*(CCSD), CCSD(TQ), and QCISD(TQ) methods

- R.J. Bartlett, J.D. Watts, S.A. Kucharski, and J. Noga, *Chem. Phys. Lett.* **165**, 513 (1990).
- K. Raghavachari, J.A. Pople, E.S. Replogle, and M. Head-Gordon, *J. Phys. Chem.* **94**, 5579 (1990).

m) Two-determinant CCSD method

- A. Balkova and R.J. Bartlett, *Chem. Phys. Lett.* **193**, 364 (1992).

C. Analytical gradients for MBPT/CC methods

Review article

- R.J. Bartlett, J.F. Stanton, and J.D. Watts, in *Advances in Molecular Vibrations and Collision Dynamics*, Vol. 1, ed. J. Bowman, JAI Press, p. 139 (1991).
- J. Gauss and D. Cremer, *Adv. Quant. Chem.* **23**, 205 (1992).

a) General MBPT/CC gradient theory

- E.A. Salter, G.W. Trucks, and R.J. Bartlett, *J. Chem. Phys.* **90**, 1752 (1989).

b) Implementation for closed- and open-shell CCSD

- J. Gauss, J.F. Stanton, and R.J. Bartlett, *J. Chem. Phys.* **95**, 2623 (1991).

c) QRHF-CCSD gradients

- J. Gauss, J.F. Stanton, and R.J. Bartlett, *J. Chem. Phys.* **95**, 2639 (1991).

d) ROHF-CCSD gradients

- J. Gauss, W.J. Lauderdale, J.F. Stanton, J.D. Watts, and R.J. Bartlett, *Chem. Phys. Lett.* **182**, 207 (1991).

e) QCISD gradients

- J. Gauss and D. Cremer, *Chem. Phys. Lett.* **150**, 280 (1988).
- J. Gauss, J.F. Stanton, and R.J. Bartlett, *J. Chem. Phys.* **95**, 2623 (1991).

f) MBPT(4) gradients for closed- and open-shells

- J. Gauss and D. Cremer, *Chem. Phys. Lett.* **138**, 131 (1987); **153**, 303 (1988).
- G.W. Trucks, J.D. Watts, E.A. Salter, and R.J. Bartlett, *Chem. Phys. Lett.* **153**, 490 (1988).
- J.D. Watts, G.W. Trucks, and R.J. Bartlett, *Chem. Phys. Lett.* **164**, 502 (1989).
- J.D. Watts, J. Gauss, and R.J. Bartlett, *Chem. Phys. Lett.* **200**, 1 (1992).

g) CCSD+T(CCSD), CCSD(T) gradients for closed- and open-shells

- J.D. Watts, J. Gauss, and R.J. Bartlett, *Chem. Phys. Lett.* **200**, 1 (1992).
- J.D. Watts, J. Gauss, and R.J. Bartlett, *J. Chem. Phys.* **98**, 8718 (1993).

h) QCISD(T) gradients

- J. Gauss and D. Cremer, *Chem. Phys. Lett.* **163**, 549 (1990).
- J.D. Watts, J. Gauss, and R.J. Bartlett, *Chem. Phys. Lett.* **200**, 1 (1992).

i) UCC(4) gradients for closed- and open-shells

- J.D. Watts, G.W. Trucks, and R.J. Bartlett, *Chem. Phys. Lett.* **157**, 359 (1989).
- J.D. Watts, G.W. Trucks, and R.J. Bartlett, *Chem. Phys. Lett.* **164**, 502 (1989).

j) Two-determinant CCSD (TD-CCSD) analytical gradients for open-shell singlet states

- P.G. Szalay and R.J. Bartlett, *J. Chem. Phys.* **101**, 4936 (1994).

D. Analytical second derivatives for MBPT/CC methods

a) MBPT(2) second derivatives for closed shells

- N.C. Handy, R.D. Amos, J.F. Gaw, J.E. Rice, E.D. Simandiras, T.J. Lee, R.J. Harrison, W.D. Laidig, G.B. Fitzgerald and R.J. Bartlett, in *Geometrical Derivatives of Energy Surfaces and Molecular Properties*, edited by P. Jørgensen and J. Simons (Reidel, Dordrecht, 1986).
- N.C. Handy, R.D. Amos, J.F. Gaw, J.E. Rice, E.D. Simandiras, *Chem. Phys. Letters* **120**, 151 (1985).
- R.J. Harrison, G.B. Fitzgerald, W.D. Laidig, R.J. Bartlett, *Chem. Phys. Lett.* **124**, 291 (1986).

b) MBPT(2) second derivatives for open shells (UHF and ROHF)

- J.F. Stanton, J. Gauss, and R.J. Bartlett, *Chem. Phys. Lett.* **195**, 194 (1992).
- J. Gauss, J.F. Stanton, and R.J. Bartlett, *J. Chem. Phys.* **97**, 7825 (1992).

E. NMR chemical shift calculations

a) GIAO-SCF NMR chemical shift calculations

- R. Ditchfield, *Mol. Phys.* **27**, 789 (1974).
- K. Wolinski, J.F. Hinton, and P. Pulay, *J. Am. Chem. Soc.* **112**, 8251 (1990).
- M. Häser, R. Ahlrichs, H.P. Baron, P. Weis, and H. Horn, *Theoret. Chim. Acta*, **83**, 455 (1992).

b) GIAO-MBPT(2) NMR chemical shift calculations

- J. Gauss, *Chem. Phys. Lett.* **191**, 614 (1992); *J. Chem. Phys.* **99**, 3629 (1993).

F. Methods for calculating excitation energies

a) Tamm-Dancoff (CI singles) approximation

b) Random-Phase approximation (RPA)

c) Equation-of-Motion Coupled Cluster (EOM-CC) methods

- J.F. Stanton and R.J. Bartlett, *J. Chem. Phys.* **98**, 7029 (1993)

G. Methods for calculating electron attachment energies.

The electron affinity equation-of-motion coupled-cluster method :

- M. Nooijen and R.J. Bartlett, *J. Chem. Phys.* **102**, 3629 (1995).

H. Time-dependent Hartree-Fock Methods.

- H. Sekino and R.J. Bartlett, *J. Chem. Phys.* **85**, 976 (1986).
- H. Sekino and R.J. Bartlett, *Int. J. Quantum Chem.* **43**, 119 (1992).

I. The HFDFP Method.

- P. M. W. Gill, B. G. Johnson, and J. A. Pople, *Int. J. Quantum Chem. Symp.* **26**, 319 (1992).
- G. E. Scuseria, *J. Chem. Phys.* **97**, 7528 (1992).
- N. Oliphant and R. J. Bartlett, *J. Chem. Phys.* **100**, 6550 (1994).

J. Basis sets

STO-3G

- W.J.Hehre, R.F.Stewart, and J.A.Pople, *J. Chem. Phys.* **51**, 2657 (1969); first-row elements.
- W.J.Hehre, R.Ditchfield, R.F.Stewart, and J.A.Pople, *J. Chem. Phys.* **52**, 2769 (1970); second-row elements and improved scale factors for Li and Be.
- W.J.Pietro, B.A.Levi, W.J.Hehre, and R.F.Stewart, *Inorg. Chem.* **19**, 2225 (1980); third-row elements.
- W.J.Pietro, E.S.Blurock, R.F.Hout,Jr, W.J.Hehre, D.J.DeFrees, and R.F.Stewart, *Inorg. Chem.* **20**, 3650 (1981); fourth-row elements.
- W.J.Pietro and W.J.Hehre, *J. Comput. Chem.* **4**, 241 (1983); first- and second-row transition metal elements.

3-21G

- J.S.Binkley, J.A.Pople, and W.J.Hehre, *J. Am. Chem. Soc.* **102**, 939 (1980); first-row elements.
- M.S.Gordon, J.S.Binkley, J.A.Pople, W.J.Pietro, and W.J.Hehre, *J. Am. Chem. Soc.* **104**, 2797 (1982); second-row elements.

4-31G

- R.Ditchfield, W.J.Hehre, and J.A.Pople, J. Chem. Phys. **54**, 724 (1971); H,C,O,N, and F.
- W.J.Hehre and J.A.Pople, J. Chem. Phys. **56**, 4233 (1972); B.
- W.J.Hehre and W.A.Lathan, J. Chem. Phys. **56**, 5255 (1972); P,S, and Cl.

6-31G

- W.J.Hehre, R.Ditchfield, and J.A.Pople, J. Chem. Phys. **56**, 2257 (1972); C,N,O, and F.
- J.D.Dill and J.A.Pople, J. Chem. Phys. **62**, 2921 (1975); Li,Be,B.
- M.M.Francl, W.J.Pietro, W.J.Hehre, J.S.Binkley, M.S.Gordon, D.J.DeFrees, and J.A.Pople, J. Chem. Phys. **77**, 3654 (1982); second-row elements.

6-31G* (and 6-31G**)

- P.C.Hariharan and J.A.Pople, Theor. Chim. Acta **28**, 213 (1973); first-row elements.
- M.M.Francl, W.J.Pietro, W.J.Hehre, J.S.Binkley, M.S.Gordon, D.J.DeFrees, and J.A.Pople, J. Chem. Phys. **77**, 3654 (1982); second-row elements.

6-311G**

- R.Krishnan, J.S.Binkley, R.Seeger, and J.A.Pople, J. Chem. Phys. **72**, 650 (1980); first-row elements.

Dunning basis sets

- T.H.Dunning, Jr, J. Chem. Phys. **53**, 2823 (1970); double-zeta and triple-zeta valence contractions of $9s5p$ primitive set for H,B-F, including $4s2p$ and $4s3p$.
- T.H.Dunning, Jr, J. Chem. Phys. **55**, 716 (1971); triple-zeta valence and more flexible contractions of $10s6p$ primitive set for H,B-F, including $5s3p$ and $5s4p$. Also $4s$ contraction for Li, $5s$ contraction for Be.
- T.H.Dunning, Jr and P.J.Hay, in *Methods of Electronic Structure Theory*, edited by H.F.Schaefer III, Plenum, New York, 1977; $6s4p$ contractions of $11s7p$ for Al-Cl (Cl basis contains typographical errors, see Craven et al, Chem. Phys. Lett. **116**, 119 (1985)); VDZP basis set for first-row elements; diffuse and Rydberg functions; recommended polarization exponents from SCF calculations.

- T.H.Dunning, Jr, J. Chem. Phys. **90**, 1007 (1989) (PVDZ-PVQZ); R.A. Kendall, T.H. Dunning, Jr, and R.J. Harrison, J. Chem. Phys. **96**, 6796 (1992) (diffuse functions for PVDZ to PVQZ); D.E. Woon and T.H. Dunning, Jr, J. Chem. Phys. **98**, 1358 (1993) (PV5Z and diffuse functions for PV5Z); methodology of the “correlation consistent sets, but does not include the actual sets : these must be obtained directly from Dunning via FTP.

Atomic natural orbital basis sets of Roos and coworkers (WMR)

- P.O. Widmark, P.A. Malmqvist, and B.O. Roos, *Theor. Chim. Acta* **77**, 291 (1990).
- P.O. Widmark, B.J. Persson, and B.O. Roos, *Theor. Chim. Acta* **79**, 419 (1991).

Polarized basis sets of Sadlej and coworkers

- A.J. Sadlej, *Collec. Czech. Chem. Commun.* **53**, 1995 (1988).
- A.J. Sadlej and M. Urban, *J. Mol. Struct. (THEOCHEM)* **234**, 147 (1991).
- A.J. Sadlej, *Theor. Chim. Acta* **79**, 123 (1991).
- A.J. Sadlej, *Theor. Chim. Acta* **81**, 45 (1992).
- A.J. Sadlej, *Theor. Chim. Acta* **81**, 339 (1992).

Basis sets optimized by Ahlrichs and coworkers

- A. Schäfer, H. Horn, and R. Ahlrichs J. Chem. Phys. **97**, 2571 (1992). These basis sets have been optimized for atoms at the SCF level using analytic gradient techniques and have been supplemented with a suitable choice of polarization functions. Input in the ACES II program is in this case always via BASIS=SPECIAL. The actual basis sets can be obtained via FTP from the authors.

Some other extended basis sets for second-row atoms (*sp* parts)

- A.D.McLean and G.S.Chandler, J. Chem. Phys. **72**, 5639 (1980); up to $6s5p$ sets for second-row atoms contracted from up to $12s9p$ primitive sets.

Polarization exponents from correlated calculations

- L.T.Redmon, G.D.Purvis III, and R.J.Bartlett, J. Am. Chem. Soc. **101**, 2856 (1979); recommended polarization exponents for use with DZ basis sets for H,B,C,N, and O.

VI. Programmer's Manual

A. Structure and Philosophy of Basic Modules

B. Files Used by the Programs

ACES II makes use of a small number of unformatted files which contain various pieces of information relating to the calculation. For the most part, these files are direct access files which have an implicit logical record structure. This means that the programs which read from and/or write to these files treat the information in chunks (logical records) which do not correspond directly to the physical structure of the file on the disk drive. Translation between the logical and physical structure of the files is handled by specialized I/O routines (GETLST, PUTLST, GETREC and PUTREC). The following section lists all of the permanent files used by ACES II.

JOBARC : This file contains a considerable amount of information about the calculation, such as basis function and symmetry information, SCF eigenvectors, density matrices *etc.* All records resident on this file are associated with a character string which is used to reference them (see the description of the PUTREC and GETREC subroutines in the library routines handbook). Thus far, the programmers of ACES II have followed a philosophy which limits the size of records written to JOBARC to be of relatively small size (no larger than the square of the number of basis functions), with the result that this file is rarely larger than a few megabytes in size. JOBARC is a direct access file with an imposed logical record structure, with information regarding the translation held in the JOBARC common block and stored in the JASMRY file. Entries are read by the GETREC subroutine and written by the PUTREC subroutine. Subroutine JASMRY dumps a list of the contents of the file.

JASMRY : See JOBARC, above. This file contains a single record which is read and written by the DUMPJA subroutine.

OPTARC : This file contains information about the progress of optimizations and is a historical carry-over from the old days. Eventually, this file will probably be phased out with its information being moved over to JOBARC. This file has a sequential structure with relatively large records, and the number of records always equals the number of optimization cycles plus one. OPTARC records are written by subroutine ARCHIVE in joda.f and are read by subroutine RETRIEVE in the same program.

JODADONE : This file contains a single empty sequential record and is written by XJODA when an optimization has terminated. Its presence is used by XACES2 to test for the convergence of an optimization. It is written by subroutine EFOL of joda.f.

FCMINT : This file contains the full square internal coordinate force constant matrix and can be used to initialize the Hessian in geometry optimizations.

MOINTS, GAMLAM, MOINTS, SECDER, DERINT : These five files are actually treated as a single file by the program system, and contain the integral and amplitude lists used by the post-SCF program modules. As such, these files can become quite large and will always take up most of the disk space in actual calculations. The only difference between these files is the lists which are stored on them. MOINTS contains list numbers 1-100; GAMLAM

contains lists 101–200; MOABCD contains lists 201–300; SECDER contains lists 301–400; and DERINT contains lists 401–500. These files are composed of direct access records with an imposed logical record structure, and all I/O is supervised by the GETLST and PUTLST subroutines and their dependent routines FETCH and DUMP.

C. Job archive (JOBARC) file contents

This JOBARC file serves as a repository of many diverse data about the calculation being run. Records are read and written by nearly every module in ACES II. This section is intended as a catalog of the records which *may* be found on the JOBARC file, depending on the job type, etc. With each record is given a brief description, the size of the record, and which module first write it. Cross references are often given to related records.

Terms of reference

The “computational basis” or “computational order” is the order of centers and basis functions used by the integral package, and thus the rest of the calculation.

The “ZMAT basis” or “ZMAT order” refers to centers arranged in the order of the user’s ZMAT file. Basis functions are rearranged to correspond to the ZMAT center ordering and may be further rearranged within a center to facilitate symmetry checking operations.

For transformation matrices, we try to indicate which bases label the rows and columns of the matrix in order to avoid confusion in their application. For non-unitary transformations, we also specify which side the transformation acts on.

SOs are symmetry-adapted orbitals, the basis in which the bulk of ACES II operates. AOs are atomic orbitals which are used mainly for symmetry analysis within the codes.

The sizes of records are described symbolically using the following definitions:

NAO	Number of atomic orbitals
NMO	Number of molecular orbitals
NAtoms	Number of centers specified in the ZMAT
NIrreps	Number of irreps in computational point group
NOrbits	Number of “orbits” (symmetry-unique centers)
NSO	Numer of symmetry-adapted orbitals

The actual variables employed in the code may be different.

C.1 ANGMOMBF

l -quantum number of each AO basis function in the computational basis. $s=0, p=1, \dots$. See also ANMOMBF0.

Data type: integer. Dimension: NAO. Written by: xvmol2ja.

C..2 ANMOMBF0

l -quantum number of each AO basis function in the ZMAT order. $s=0, p=1, \dots$. See also ANGMOMBF.

Data type: integer. Dimension: NAO. Written by: xvmol2ja.

C..3 AO2SO

The transformation matrix connecting the symmetry-adapted orbitals (columns of AO2SO) with the atomic orbitals (rows of AO2SO). This matrix multiplied on the *left* of an SO basis object will transform it to the AO basis (note that the record is misnamed in this respect). This matrix is in general *not* unitary and frequently not even square. The inverse transformation is given in the AO2SOINV record.

Data type: floating point. Dimension: NAO \times NSO. Written by: xvmol2ja.

C..4 AO2SOINV

The transformation matrix connecting the atomic orbitals (columns of AO2SOINV) with the symmetry-adapted orbitals (rows of AO2SOINV). This matrix multiplied on the *left* of an AO basis object will transform it to the SO basis. This matrix is in general *not* unitary and frequently not even square. The inverse transformation is given in the AO2SOINV record.

Data type: floating point. Dimension: NSO \times NAO. Written by: xvmol2ja.

C..5 ATOMCHRG

Atomic number of all atoms specified in Z-matrix. Dummy atoms have a value of zero.

Data type: floating point. Dimension: NATOMS. Written by: xjoda.

C..6 ATOMMASS

Atomic masses of all atoms specified in Z-matrix. Dummy atoms have zero mass.

Data type: floating point. Dimension: NATOMS. Written by: xjoda.

C..7 BRUKTEST

A flag which tells the xaces ii drive program if a Brueckner orbital-based calculation has converged to the tolerance specified by the BRUCK_CONV keyword. A value of 1 indicates convergence, a value of 0 forces another iteration.

Data type: integer. Dimension: 1. Written by: xvcc.

C..8 CENTERBF

Atomic center to which each AO basis function belongs. Both centers and basis functions are in the computational order. See also CENTERBF0.

Data type: integer. Dimension: NAO. Written by: xvmol2ja.

C..9 CMP2ZMAT

The transformation matrix connecting the computational ordered SO basis (columns of CMP2ZMAT) to the ZMAT ordered AO basis (rows of CMP2ZMAT). This matrix operates on the left. This matrix is in general *not* unitary and frequently not even square. The inverse transformation is given in the ZMAT2CMP record.

Data type: floating point. Dimension: NAO×NSO. Written by: xvmol2ja.

C..10 CENTERBF0

Atomic center to which each AO basis function belongs. Both centers and basis functions are in the ZMAT order. See also CENTERBF.

Data type: integer. Dimension: NAO. Written by: xvmol2ja.

C..11 COORD

Cartesian coordinates for all atoms in Z-matrix, according to the computational orientation.

Data type: floating point. Dimension: NATOMS*3. Written by: xjoda

C..12 FULLAOSO

The transformation matrix connecting the atomic orbitals (columns of FULLAOSO) with the symmetry-adapted orbitals (rows of FULLAOSO). This matrix multiplied on the *left* of an AO basis object will transform it to the SO basis. This matrix is in general *not* unitary. This matrix differs from AO2SOINV in that it is always a full square – it includes in the SO basis, functions such as spherical harmonic contaminants, which are deleted in the computational basis. The inverse transformation is given in the FULLSOAO record.

Data type: floating point. Dimension: NAO×NAO. Written by: xvmol2ja.

C..13 FULLSOAO

The transformation matrix connecting the symmetry-adapted orbitals (columns of FULLSOAO) with the atomic orbitals (rows of FULLSOAO). This matrix multiplied on the *left* of an SO basis object will transform it to the AO basis. This matrix is in general *not* unitary. This matrix differs from AO2SO in that it is always a full square – it includes in the SO basis, functions such as spherical harmonic contaminants, which are deleted in the computational basis. The inverse transformation is given in the FULLAOSO record.

Data type: floating point. Dimension: NAO×NAO. Written by: xvmol2ja.

C..14 GRADIENT

This record contains the Cartesian gradient. The entries are ordered $\partial E/\partial x$, $\partial E/\partial y$, $\partial E/\partial z$, atom-by-atom. The atoms follow the computational order.

Data type: integer. Dimension: 3*NATOMS. Written by: xvdint.

C..15 HESSIANM

This record contains the Hessian matrix. The ordering is F_{1x1x} , F_{1x1y} , F_{1x1z} , F_{1x2x} \cdots F_{nx1x} , F_{nx1y} \cdots . Atoms follow the computational order. The record is written by xvdint and xcphf.

Data type: integer. Dimension: 9*NATOMS*NATOMS. Written by: see above.

C..16 IFLAGS

The job control parameters which can be specified by the ZMAT file. This record is initially written by xjoda, but may be modified by subsequent programs in the execution sequence.

Data type: integer. Dimension: 100. Written by: see above.

C..17 LINEAR

A value of 1 indicates that the molecule is linear, 0 otherwise.

Data type: integer. Dimension: 1. Written by: xjoda.

C..18 MAP2ZMAT

Maps atomic centers from the computational ordering to the ZMAT ordering. If i is a computational center, MAP2ZMAT(i) is the number of the same center in the ZMAT file.

Data type: integer. Dimension: NAtoms. Written by: xvmol2ja.

C..19 MODROPA

The alpha molecular spin-orbitals which will be dropped prior to the electron correlation calculation. The orbitals are indexed by eigenvalue, with the smallest eigenvalue (most strongly bound) being 1 and the largest eigenvalue equal to the number of basis functions.

Data type: integer. Dimension: NUMDROPA (see below). Written by: xjoda.

C..20 MODROPB

The beta molecular spin-orbitals which will be dropped prior to the electron correlation calculation. While this record is present on JOBARC, it is never read, as the program system does not allow different values of MODROPA and MODROPB.

Data type: integer. Dimension: NUMDROPB. Written by: xjoda.

C..21 NAOBASFN

Number of atomic orbitals in the basis. Because the SO basis may have functions deleted from it (i.e. spherical harmonic contaminants), the AO basis may be larger than the SO basis. See also NBASTOT.

Data type: integer. Dimension: 1. Written by: xvmol2ja.

C..22 NAOBFORB

Number of atomic orbitals in each “orbit” (symmetry-unique center). This is the AO basis analog of NBASATOM.

Data type: integer. Dimension: NOrbits. Written by: xvmol2ja.

C..23 NBASATOM

Number of symmetry-adapted orbitals in each “orbit” (symmetry-unique center). This is the SO basis analog of NAOBFORB.

Data type: integer. Dimension: NOrbits. Written by: xvmol2ja.

C..24 NATOMS

The number of atoms in the Z-matrix (includes dummy atoms).

Data type: integer. Dimension: 1. Written by: xjoda.

C..25 NREALATM

The number of non-dummy centers in the Z-matrix.

Data type: integer. Dimension: 1. Written by: xjoda.

C..26 NUCREP

The nuclear repulsion energy in atomic units.

Data type: floating point. Dimension: 1. Written by: xjoda.

C..27 NBASTOT

Number of symmetry-adapted orbitals in the basis. Because the SO basis may have functions deleted from it (i.e. spherical harmonic contaminants), the AO basis may be larger than the SO basis. See also NAOBASFN.

Data type: integer. Dimension: 1. Written by: xvmol2ja.

C..28 NMPROTON

The number of protons in the molecule.

Data type: integer. Dimension: 1. Written by: xjoda.

C..29 NOCCORB

The number of occupied molecular orbitals used in the post-SCF calculation according to spin.

Data type: integer. Dimension: 2. Written by: xintprc.

C..30 NUCREP

Nuclear repulsion energy in Hartrees (atomic units).

Data type: floating point. Dimension: 1. Written by: xvmol2ja.

C..31 NUMBASIR

Number of SOs in each irrep.

Data type: integer. Dimension: NIrreps. Written by: xvmol2ja.

C..32 NUMDROPA

Number of alpha molecular spin-orbitals which will be dropped prior to the electron correlation calculation.

Data type: integer. Dimension: 1. Written by: xjoda.

C..33 NUMDROPB

Number of beta molecular spin-orbitals which will be dropped prior to the electron correlation calculation. While this record is present on JOBARC, it is never read, as the program system does not allow different values of NUMDROPA and NUMDROPB.

Data type: integer. Dimension: 1. Written by: xjoda.

C..34 NVRTORB

The number of virtual molecular orbitals used in the post-SCF calculation according to spin.

Data type: integer. Dimension: 2. Written by: xintprc.

C..35 OCCUPYA

Orbital occupancy vector for alpha spin-orbitals. Originally written by xjoda as specified by ZMAT input.

Data type: integer. Dimension: NIrreps. Written by: xjoda.

C..36 OCCUPYB

Orbital occupancy vector for beta spin-orbitals. Originally written by xjoda as specified by ZMAT input.

Data type: integer. Dimension: NIrreps. Written by: xjoda.

C..37 ORIENTMT

Matrix which relates computational and canonical orientations.

Data type: floating point. Dimension: 9. Written by: xjoda.

C..38 QRHFIRR

The symmetries of the orbitals which are involved in the QRHF orbital occupation alteration. A minus sign indicates that a β orbital of the associated symmetry will be depopulated, while a positive sign indicates that an α orbital will be populated.

Data type: floating point. Dimension: QRHFTOT. Written by: xjoda.

C..39 QRHFLOC

The offset for orbitals which are involved in the QRHF orbital occupation alteration. For depopulation calculations, a value of 1 indicates the occupied orbital with the largest eigenvalue in the symmetry block; larger values give offsets with respect to the Fermi level. For electron addition, 1 refers to the lowest unoccupied orbital within the symmetry block, and larger values give offsets as above.

Data type: floating point. Dimension: QRHFTOT. Written by: xjoda.

C..40 QRHFTOT

The number of orbitals which will undergo a change in occupation number in the QRHF procedure.

Data type: floating point. Dimension: 1. Written by: xjoda.

C..41 S2SCF

The value of $\langle S^2 \rangle$ for the reference wavefunction.

Data type: floating point. Dimension: 1. Written by: xvscf.

C..42 SCFENEG

The SCF energy.

Data type: floating point. Dimension: 1. Written by: xvscf.

C..43 SCFEVALA

The diagonal elements of the α Fock matrix, expressed in the molecular orbital basis. Ordered according to the post-SCF orbital numbering scheme.

Data type: floating point. Dimension: NMO. Written by: xftran*.

C..44 SCFEVALB

The diagonal elements of the β Fock matrix, expressed in the molecular orbital basis. Ordered according to the post-SCF orbital numbering scheme.

Data type: floating point. Dimension: NMO. Written by: xftran*.

C..45 SCFEVCA0

The α molecular orbitals, ordered according to the SCF orbital numbering scheme.

Data type: floating point. Dimension: NAO*NAO. Written by: xvscf.

C..46 SCFEVCB0

The β molecular orbitals, ordered according to the SCF orbital numbering scheme.

Data type: floating point. Dimension: NAO*NAO. Written by: xvscf.

C..47 SCFEVECA

The α molecular orbitals, ordered according to the post-SCF orbital numbering scheme, as used in post-SCF calculations.

Data type: floating point. Dimension: NMO*NAO. Written by: xftran*.

C..48 SCFEVECB

The β molecular orbitals, ordered according to the post-SCF orbital numbering scheme, as used in post-SCF calculations.

Data type: floating point. Dimension: NMO*NAO. Written by: xftran*.

C..49 SCFKICK

A flag which tells the xaces ii driver program if an SCF orbital instability has been found which can be followed within the current symmetry and spin restrictions. A value of 1 forces reexecution of the SCF code with a rotated eigenvector, a value of 0 terminates the job. Used only in jobs which invoke HFSTABILITY=FOLLOW.

Data type: integer. Dimension: 1. Written by: xintprc.

C..50 TOTENERG

The total energy. This record is written by many programs, beginning with xvscf and keeps a running tab on the total energy.

Data type: floating point. Dimension: 1. Written by: see above.

C..51 UHFRHF

This flag gives the type of SCF calculation which will be performed. 0 indicates RHF, 1 indicates UHF and 2 indicates ROHF. UHFRHF is originally written by xjoda, but subsequent programs may modify it.

Data type: floating point. Dimension: 1. Written by: see above.

C..52 ZMAT2CMP

The transformation matrix connecting the ZMAT ordered AO basis (columns of ZMAT2CMP) to the computationally ordered SO basis (rows of ZMAT2CMP). This matrix operates on the left. This matrix is in general *not* unitary and frequently not even square. The inverse transformation is given in the CMP2ZMAT record.

Data type: floating point. Dimension: NSO×NAO. Written by: xvmol2ja.

D. Integral Lists

The following section describes the contents of the MOINTS, GAMLAM, MOABCD, SECDER and DERINT files, which are used to store two-electron integrals and various other quantities.

PART I. Lists used in standard calculation types (single point energy, first and second derivative calculations)

J	I	Quantity	Storage Mode
ALL ; FOR			
EACH			

1-NIRREP	5	<AB IJ>	A,I ; B,J (NOT ANTISYMMETRIZED)
1-NIRREP	6	<ab ij>	a,i ; b,j (NOT ANTISYMMETRIZED)
1-NIRREP	11	<IJ KL>	I<J ; K<L ***
1-NIRREP	12	<ij kl>	i<j ; k<l ***
1-NIRREP	13	<Ij Kl>	I,j ; K,l

1-NIRREP	7	$\langle IJ KA \rangle$	$I < J ; K, A$	
1-NIRREP	8	$\langle ij ka \rangle$	$i < j ; k, a$	***
1-NIRREP	9	$\langle Ij Ak \rangle$	$I, j ; A, k$	***
1-NIRREP	10	$\langle Ij Ka \rangle$	$I, j ; K, a$	
1-NIRREP	14	$\langle AB IJ \rangle$	$A < B ; I < J$	
1-NIRREP	15	$\langle ab ij \rangle$	$a < b ; i < j$	***
1-NIRREP	16	$\langle Ab Ij \rangle$	$A, b ; I, j$	
1-NIRREP	17	$\langle Ab Ij \rangle$	$b, j ; A, I$	***
1-NIRREP	18	$\langle Ab Ij \rangle$	$A, I ; b, j$	
1-NIRREP	19	$\langle AB IJ \rangle$	$A, I ; B, J$	
1-NIRREP	20	$\langle ab ij \rangle$	$a, i ; b, j$	***
1-NIRREP	21	$\langle Ab Ij \rangle$	$A, j ; b, I$	
1-NIRREP	22	$\langle Ab Ij \rangle$	$b, I ; A, j$	***
1-NIRREP	23	$\langle IA JB \rangle$	$B, I ; A, J$	
1-NIRREP	24	$\langle ia jb \rangle$	$b, i ; a, j$	***
1-NIRREP	25	$\langle iA jB \rangle$	$B, i ; A, j$	
1-NIRREP	26	$\langle aI bJ \rangle$	$b, I ; a, J$	***
1-NIRREP	27	$\langle AB CI \rangle$	$A < B ; C, I$	
1-NIRREP	28	$\langle ab ci \rangle$	$a < b ; c, i$	***
1-NIRREP	29	$\langle Ab Ic \rangle$	$A, b ; I, c$	***
1-NIRREP	30	$\langle Ab Ci \rangle$	$A, b ; C, i$	
1-NIRREP	34	$T2(IJ, AB)$	$A, J ; B, I$	
1-NIRREP	35	$T2(ij, ab)$	$a, j ; b, i$	***
1-NIRREP	36	$T2(Ij, Ab)$	$b, j ; A, I$	***
1-NIRREP	37	$T2(Ij, Ab)$	$A, I ; b, j$	
1-NIRREP	38	$T2(Ij, Ab)$	$b, I ; A, j$	***
1-NIRREP	39	$T2(Ij, Ab)$	$A, j ; b, I$	
1-NIRREP	40	$T2(IJ, AB)$ (increment)	$A, J ; B, I$ (rings)	***
1-NIRREP	41	$T2(ij, ab)$ (increment)	$a, j ; b, i$ (rings)	***
1-NIRREP	42	$T2(Ij, Ab)$ (increment)	$A, I ; b, j$ (rings)	
1-NIRREP	43	$T2(Ij, Ab)$ (increment)	$A, j ; b, I$ (rings)	
1-NIRREP	44	$T2(IJ, AB)$	$A < B ; I < J$	
1-NIRREP	45	$T2(ij, ab)$	$a < b ; i < j$	***
1-NIRREP	46	$T2(Ij, Ab)$	$A, b ; I, j$	
1-NIRREP	48	$D(IJ, AB)$	$A < B ; I < J$	

1-NIRREP	49	D(ij,ab)	a<b ; i<j	***
1-NIRREP	50	D(Ij,Ab)	A,b ; I,j	
1-NIRREP	51	W(MN,IJ) intermed.	M<N ; I<J	***
1-NIRREP	52	W(mn,ij) intermed.	m<n ; i<j	***
1-NIRREP	53	W(Mn,Ij) intermed.	M,n ; I,j	
1-NIRREP	54	W(MB,EJ) intermed.	E,M ; B,J	
1-NIRREP	55	W(mb,ej) intermed.	e,m ; b,j	***
1-NIRREP	56	W(Mb,Ej) intermed.	E,M ; b,j	
1-NIRREP	57	W(mB,eJ) intermed.	e,m ; B,J	***
1-NIRREP	58	W(mB,Ej) intermed.	E,m ; B,j	
1-NIRREP	59	W(Mb,eJ) intermed.	e,M ; b,J	***
1-NIRREP	61	T2(IJ,AB) increment	A<B ; I<J	***
		L2(IJ,AB) increment	A<B ; I<J in Lambda iterations	***
		T2(IJ,AB) [3]	A<B ; I<J iterative	ROHF
1-NIRREP	62	T2(ij,ab) increment	a<b ; i<j	***
		L2(ij,ab) increment	a<b ; i<j in Lambda iterations	***
		T2(ij,ab) [3]	a<b ; i<j iterative	ROHF
1-NIRREP	63	T2(Ij,Ab) increment	A,b ; I,j	
		L2(Ij,Ab) increment	A,b ; I,j in Lambda iterations	
		T2(Ij,Ab) [3]	A,b ; I,j iterative	ROHF
1-NIRREP	64	Reciprocal D(IJ,AB)	A<B ; I<J	
9	64	Reciprocal D(I,A)	A,I	
1-NIRREP	65	Reciprocal D(ij,ab)	a<b ; i<j	***
9	65	Reciprocal D(i,a)	a,i	***
1-NIRREP	66	Reciprocal D(Ij,Ab)	A,b ; I,j	
1-3	70-89	Reserved for RLE Jacobi iterates		
1	90	T1AA	A,I	
2	90	T1BB	a,i	***
3	90	T1AA AND L1AA INCREM	A,I	
4	90	T1BB AND L1BB INCREM	a,i	***
5	90	AB MO OVERLAP MATRIX	I,j (o-o)	***
6	90	AB MO OVERLAP MATRIX	A,j (v-o)	***
7	90	AB MO OVERLAP MATRIX	I,b (o-v)	***
8	90	AB MO OVERLAP MATRIX	A,b (v-v)	***
9	90	T1AA[2]; iterative	A,I	ROHF
10	90	T1BB[2]; iterative	a,i	ROHF

1	91	F(MI) intermed.	M,I	
2	91	F(mi) intermed.	m,i	***
3	91	f(M,I) (fock matrix)	M,I	(only non-HF methods)
4	91	f(m,i) (fock matrix)	m,i	(only non-HF methods) ***
1	92	F(EA) intermed.	E,A	
2	92	F(ea) intermed.	e,a	***
3	92	f(E,A) (fock matrix)	E,A	(only non-HF methods)
4	92	f(e,a) (fock matrix)	e,a	(only non-HF methods) ***
1	93	F(AI) intermed.	A,I	
2	93	F(ai) intermed.	a,i	***
3	93	f(A,I) (fock matrix)	A,I	(only non-HF methods)
4	93	f(a,i) (fock matrix)	a,i	(only non-HF methods) ***
1-NIRREP	94	T2(IJ,AB)[2] iterative	A<B ; I<J	ROHF
1-NIRREP	95	T2(ij,ab)[2] iterative	a<b ; i<j	ROHF
1-NIRREP	96	T2(Ij,Ab)[2] iterative	A,b ; I,j	ROHF
1-NIRREP	98	T2(Ij,Ab) cumulative	A,b ; I,j (MBPT S^2)	ROHF
1-NIRREP	107	G(IJ,KA)	I<J ; K,A	***
		W(IJ,KA)	I<J ; K,A in VLAMCC	***
1-NIRREP	108	G(ij,ka)	i<j ; k,a	***
		W(ij,ka)	i<j ; k,a in VLAMCC	***
1-NIRREP	109	G(Ij,Ak)	I,j ; A,k	***
		W(Ij,Ak)	I,j ; A,k in VLAMCC	***
1-NIRREP	110	G(Ij,Ka)	I,j ; K,a	
		W(Ij,Ka)	I,j ; K,a in VLAMCC	
1-NIRREP	111	G(IJ,KL)	I<J ; K<L	***
1-NIRREP	112	G(ij,kl)	i<j ; k<l	***
1-NIRREP	113	G(Ij,Kl)	I,j ; K,l	
1-NIRREP	114	G(IJ,AB)	I<J ; A<B	
1-NIRREP	115	G(ij,ab)	i<j ; a<b	***
1-NIRREP	116	G(Ij,Ab)	I,j ; A,b	
1-NIRREP	117	G(iA,bJ)	b,j ; A,I	***
		resorted in VDENS to	b,I ; A,j	***
1-NIRREP	118	G(Ia,Bj)	B,J ; a,i	
		resorted in VDENS to	B,i ; a,J	
1-NIRREP	123	G(IA,JB)	B,I ; A,J	
		resorted in VDENS to	B,J ; A,I	

1-NIRREP 124	G(ia,jb)	b,i ; a,j	***
resorted in VDENS	to	b,j ; a,i	***
1-NIRREP 125	G(iA,jB)	B,i ; A,j	
resorted in VDENS	to	B,j ; A,i	
1-NIRREP 126	G(Ia,Jb)	b,I ; a,J	***
resorted in VDENS	to	b,J ; a,I	***
1-NIRREP 127	G(AB,CI)	A<B ; C,I	***
	W(AB,CI)	A<B ; C,I in VLAMCC	***
1-NIRREP 128	G(ab,ci)	a<b ; c,i	***
	W(ab,ci)	a<b ; c,i in VLAMCC	***
1-NIRREP 129	G(Ab,Ic)	A,b ; I,c	***
	W(Ab,Ic)	A,b ; I,c in VLAMCC	***
1-NIRREP 130	G(Ab,Ci)	A,b ; C,i	
	W(Ab,Ci)	A,b ; C,i in VLAMCC	
1-NIRREP 131	G(AB,CD)	A<B ; C<D	***
1-NIRREP 132	G(ab,cd)	a<b ; c<d	***
1-NIRREP 133	G(Ab,Cd)	A,b ; C,d	
1-NIRREP 134	L2(IJ,AB)	A,J ; B,I	
1-NIRREP 135	L2(ij,ab)	a,j ; b,i	***
1-NIRREP 136	L2(Ij,Ab)	b,j ; A,I	***
1-NIRREP 137	L2(Ij,Ab)	A,I ; b,j	
1-NIRREP 138	L2(Ij,Ab)	b,I ; A,j	***
1-NIRREP 139	L2(Ij,Ab)	A,j ; b,I	
1-NIRREP 144	L2(IJ,AB)	A<B ; I<J	
1-NIRREP 145	L2(ij,ab)	a<b ; i<j	***
1-NIRREP 146	L2(Ij,Ab)	A,b ; I,j	

Derivatives of one-electron integrals in AO basis:

1-NIRREP 150	S(mu,nu)^chi	ipert ; mu >= nu	
1-NIRREP 151	f(mu,nu)^(chi)	(alpha)ipert ; mu >= nu	
1-NIRREP 152	f(mu,nu)^(chi)	(beta) ipert ; mu >= nu	***
1-3 153	dipole(mu,nu)	ixyz ; mu >= nu	
1-NIRREP 154	A^(chi)*DREL	(alpha)ipert ; mu >= nu	
1-NIRREP 155	A^(chi)*DREL	(beta) ipert ; mu >= nu	***

Relaxed density matrix (correlation correction only):

1	160	D(I,J)	occ.-occ. block of relaxed density matrix (alpha)	
2	160	D(i,j)	occ.-occ. block of relaxed density matrix (beta)	***
3	160	D(A,B)	virt.-virt. block of relaxed density matrix (alpha)	
4	160	D(a,b)	virt.-virt. block of relaxed density matrix (beta)	***
5	160	D(A,I)	occ.-virt. block of relaxed density matrix (alpha)	
6	160	D(a,i)	occ.-virt. block of relaxed density matrix (beta)	***

Intermediate matrix I(p,q) (correlation correction only):

1	161	I(I,J)	occ.-occ. block of intermediate matrix (alpha)	
2	161	I(i,j)	occ.-occ. block of intermediate matrix (beta)	***
3	161	I(A,B)	virt.-virt. block of intermediate matrix (alpha)	
4	161	I(a,b)	virt.-virt. block of intermediate matrix (beta)	***
5	161	I(A,I)	virt.-occ. block of intermediate matrix (alpha)	
6	161	I(a,i)	virt.-occ. block of intermediate matrix (beta)	***

Gamme-intermediates created in the triple code for CCSD(T), ...

1-NIRREP	162	G^trip(IJ,KA)	I<J ; K,A	***
1-NIRREP	163	G^trip(ij,ka)	i<j ; k,a	***
1-NIRREP	164	G^trip(Ij,Ak)	I,j ; A,k	***
1-NIRREP	165	G^trip(Ij,Ka)	I,j ; K,a	
1-NIRREP	166	G^trip(AB,CI)	A<B ; C,I	***
1-NIRREP	167	G^trip(ab,ci)	a<b ; c,i	***
1-NIRREP	168	G^trip(Ab,Ic)	A,b ; I,c	***
1-NIRREP	169	G^trip(Ab,Ci)	A,b ; C,i	

Derivatives of Overlap and Fock matrices and CPHF coefficients in MO basis:

1-NIRREP 170		$S(I,J)^{\chi}$		
1-NIRREP 171		$S(i,j)^{\chi}$		***
1-NIRREP 172		$S(A,B)^{\chi}$		
1-NIRREP 173		$S(a,b)^{\chi}$		***
1-NIRREP 174		$S(A,I)^{\chi}$		
1-NIRREP 175		$S(a,i)^{\chi}$		***
1-NIRREP 176		$F(I,J)^{\chi}$		
1-NIRREP 177		$F(i,j)^{\chi}$		***
1-NIRREP 178		$F(A,B)^{\chi}$		
1-NIRREP 179		$F(a,b)^{\chi}$		***
1-NIRREP 180		$F(A,I)^{\chi}$		
1-NIRREP 181		$F(a,i)^{\chi}$		***
1-NIRREP 182		$U(A,I)^{\chi}$		
1-NIRREP 183		$U(a,i)^{\chi}$		***
1-NIRREP 184		additional lists used in CPHF (RHF and UHF)		
1-NIRREP 185		additional lists used in CPHF (UHF only)		***
1-NIRREP 186		$A^{\chi} * D(I,J)$		
1-NIRREP 187		$A^{\chi} * D(i,j)$		***
1-NIRREP 188		$A^{\chi} * D(A,I)$		
1-NIRREP 189		$A^{\chi} * D(a,i)$		***
1	190	L1AA	A,I	
2	190	L1BB	a,i	***
1	191	G(MI) intermed.	M,I	
2	191	G(mi) intermed.	m,i	***
1	192	G(A,E) intermed.	E,A	
2	192	G(a,e) intermed.	e,a	***
1-NIRREP 231		$\langle AB CD \rangle$	$A < B ; C < D$	***
		$W(AB,CD)$	$A < B ; C < D$ DURING VLAMCC	***
1-NIRREP 232		$\langle ab cd \rangle$	$a < b ; c < d$	***
		$W(ab,cd)$	$a < b ; c < d$ DURING VLAMCC	***
1-NIRREP 233		$\langle Ab Cd \rangle$	$A, b ; C, d$	
		$\langle Ab Cd \rangle$	$A < = b ; C, d$ single point RHF	
		$W(Ab, Cd)$	$A, b ; C, d$ DURING VLAMCC	

Some additional lists are used in MBPT(2) second derivative calculations.
These are:

1-NIRREP 314		$d \langle IJ AB \rangle / dx$	$A < B ; I < J, IPERT$	***
--------------	--	-----------------------------------	------------------------	-----

```

1-NIRREP 315      d<ij||ab>/dx      a<b ; i<j,IPERT      ***
1-NIRREP 316      d<Ij||Ab>/dx      A,b ; I,j,IPERT

```

In analytic gradient calculations, the direct access files are reinitialized and restructured by the modules XANTI and XBCKTRN, which process and transform the two-particle reduced density matrix. These programs use the MOINTS and GAMLAM files in the following way

a) Lists used by ANTI

left index	right index	quantity	symmetry type	storage	
1	1-NSYM	G(PQ,rs)	AAAA	P<Q ; r<s	
2	1-NSYM	G(PQ,rs)	AABB	P<Q ; r<s	
3	1-NSYM	G(PQ,rs)	ABAB	P,Q ; r,s	
4	1-NSYM	G(PQ,rs)	ABCD	P,Q ; r,s	
1	51-50+NSYM	G(pq,RS)	AAAA	p<q ; R<S	***
2	51-50+NSYM	G(pq,RS)	AABB	p<q ; R<S	***
3	51-50+NSYM	G(pq,RS)	ABAB	p,q ; R,S	***
4	51-50+NSYM	G(pq,RS)	ABCD	p,q ; R,S	***
6	1-NSYM	G(PQ,RS)	AAAA	P<Q ; R<S	***
7	1-NSYM	G(PQ,RS)	AABB	P<Q ; R<S	***
8	1-NSYM	G(PQ,RS)	ABAB	P,Q ; R,S	***
9	1-NSYM	G(PQ,RS)	ABCD	P,Q ; R,S	***
6	51-50+NSYM	G(pq,rs)	AAAA	p<q ; r<s	***
7	51-50+NSYM	G(pq,rs)	AABB	p<q ; r<s	***
8	51-50+NSYM	G(pq,rs)	ABAB	p,q ; r,s	***
9	51-50+NSYM	G(pq,rs)	ABCD	p,q ; r,s	***

b) Lists used in BCKTRN and ABACUS

1	1-NSYM	G(mu,nu,sigma,rho)	AAAA	mu < nu ; sigma < rho
2	1-NSYM	G(mu,nu,sigma,rho)	AABB	mu < nu ; sigma < rho
3	1-NSYM	G(mu,nu,sigma,rho)	ABAB	mu,nu ; sigma,rho
4	1-NSYM	G(mu,nu,sigma,rho)	ABCD	mu,nu ; sigma,rho

In the above, the number of possible sublists defined by the right index of the list depends upon the symmetry type. Obviously, there are NIRREP different sublists for symmetry type 1 : 1,1,1,1;; NIRREP,NIRREP,NIRREP,NIRREP. For the second symmetry type, there are $NIRREP*(NIRREP-1)/2$ different sublists, namely those for each possible pair of irreps with $A < B$ (A faster running than B). We have thus 1,1,2,2; 1,1,3,3; 2,2,3,3; NIRREP-1,NIRREP-1,NIRREP, NIRREP. The symmetry type ABAB involves also $NIRREP*(NIRREP-1)/2$ sublists. They are however given in such a way that the main loop is over IRREP(AB), and the second, faster loop over B with A faster than B and IRREP(A) < IRREP(B). The fourth symmetry type, ABCD, is by far the most complicated. The main loop runs first over IRREP(CD)=IRREP(AB) with a subloop over IRREP(C) and IRREP(D) in such a way that C is faster than D and IRREP(C) < IRREP(D). For a given CD (note again IRREP(AB) = IRREP(CD)), the fastest loops runs over all possible combinations of IRREP(A) and IRREP(B) with A faster than B, IRREP(A) < IRREP(B) and also (IRREP(A), IRREP(B)) > IRREP(C),IRREP(D). The loop structure is set up for example in a very clear way in the routine GAMSRT which might be consulted for further questions regarding the set up of the loops and dealing with the AO-two-particle density matrix.

PART II. Extra lists used in excitation energy calculations

J	I	Quantity	Storage Mode	
			ALL ; FOR	
			EACH	
1-NIRREP	434	C(IJ,AB)	A,J ; B,I	[INCREMENT] ***
1-NIRREP	435	C(ij,ab)	a,j ; b,i	[INCREMENT] ***
1-NIRREP	436	C(Ij,Ab)	b,j ; A,I	[INCREMENT] ***
1-NIRREP	437	C(Ij,Ab)	A,I ; b,j	[INCREMENT]
1-NIRREP	438	C(Ij,Ab)	b,I ; A,j	[INCREMENT] ***
1-NIRREP	439	C(Ij,Ab)	A,j ; b,I	[INCREMENT]
1-NIRREP	444	C(AB,IJ)	A<B ; I<J	***
1-NIRREP	445	C(ab,ij)	a<b ; i<j	***
1-NIRREP	446	C(Ab,Ij)	A,b ; I,j	
1-NIRREP	448	D(IJ,AB)	A<B ; I<J	

1-NIRREP	449	D(ij,ab)	a<b ; i<j		***
1-NIRREP	450	D(Ij,Ab)	A,b ; I,j		
1-NIRREP	461	C(AB,IJ)	A<B ; I<J	[INCREMENT]	***
1-NIRREP	462	C(ab,ij)	a<b ; i<j	[INCREMENT]	***
1-NIRREP	463	C(Ab,Ij)	A,b ; I,j	[INCREMENT]	
1-NRECS	470	C vectors (one/record)			
1-NRECS	471	HC vectors (one/record)			
1	472	Converged right C vector (on one record)			
2	472	Converged left C vector (on one record)			
1	490	C1(AI)	AI ;		
2	490	C1(ai)	ai ;		
3	490	C1(AI)	AI ;	[INCREMENT]	
4	490	C1(ai)	ai ;	[INCREMENT]	***
1	491	X(IJ)	IJ ;		
2	491	X(ij)	ij ;		***
1	492	X(AB)	AB ;		
2	492	X(ab)	ab ;		***
1	493	X(AI)	AB ;		
2	493	X(ai)	ab ;		***
1-NIRREP	94	TDA eigenvectors (one/record)			
1-NIRREP	95	TDA eigenvalues (one/record)			

E. Common Blocks

The following is a listing of all common blocks which are automatically initialized in the CRAPSI procedure, and which programs make use of these subroutines. One should not assume that any of these common blocks have been initialized unless the program uses them. Initialization of these common blocks depends on logic based on the third argument passed to CRAPSI, as well as the presence or absence of specific files on disk.

```
COMMON // ICORE(1)
```

This is the blank common block which is used for purposes of memory allocation, and serves to provide an address for the first element of the ICORE array. The value of ICORE(1) is never used. Used in all programs except aces ii.f, joda.f, vmol.f and vprops.f.

```
COMMON /CACHEINF/ CACHNUM,CACHNMP1,CACHDIR(100),CACHPOS(100),
&                CACHFILE(100),CACHMOD(100),OLDEST
```

This common block contains information about the contents of the software I/O cache.
 CACHNUM : The total number of records held in the cache; CACHNMP1 : CACHNUM+1;
 CACHDIR : The record and file held in each of the 100 positions [these values are bitpacked

as follows : Record number (Bits 1-IBITWD*4-3); File number (remaining bits)]; CACHPOS : The location in ICORE (*relative to ICORE(0) !!!* where the record is held; CACHFILE : The actual unit number of the file associated with the record (*i.e. not in general an integer between 1 and 5*); CACHMOD : A flag indicating whether the record has been modified during its stay in core memory, with a value of 1 indicating modification, 0 otherwise; OLDEST : The location of the oldest record in the cache, which will be the first one flushed to disk when a new record must be read from the physical disk. All variables are integers. Used in post-SCF programs only. (Ed. note : The author of this section must admit that he has learned a great deal in writing this entry, even though he also authored all of the code for the I/O cache!)

```
COMMON /FLAGS/ IFLAGS(100)
```

The IFLAGS array contains the specified and/or default values of the ACES2 keywords, and is set in subroutine GTFLGS of joda.f. All values are integers. Used in all program modules.

```
COMMON /INFO/ NOCCO(2),NVRTO(2)
```

The information contained in this common block is perhaps obvious. The NOCCO and NVRTO arrays give the number of occupied and virtual orbitals by spin, irrespective of the symmetry. All variables are again integers. symmetry. Used in post-SCF programs only.

```
COMMON /IOPOS/ ICRSIZ,ICHCSZ,IOFF(2),LENREC
```

This common block contains information about memory usage and details of the software cache maintained by the ACES II I/O subsystem. ICRSIZ : The total working core area (in integer words) after subsidiary arrays have been allocated (*this value is always less than that specified by the MEMORY_SIZE keyword*); ICHCSZ : The size of a cache record in integer words; IOFF(2) : Not used; LENREC : The length of a physical disk record. The values of ICHCSZ and LENREC are always the same, and both variables persist for purely historical reasons. Used in all programs except aces ii.f, joda.f, vmol.f and vprops.f.

```
COMMON /ISTART/ IO
```

This is the number of integer words *relative to ICORE(0)* where the program working area begins. Thus, the program workspace begins at ICORE(IO), which is the address which must be passed to dependent subroutines. In general, this is necessary only in the main program unit. Used in all programs except aces ii.f, joda.f, vmol.f and vprops.f.

```
COMMON /JOBARC/ MARKER(LENGTH),LOC(LENGTH),SIZE(LENGTH),NRECS,
&                IRECWD,IRECLN
```

This common block contains information about the JOBARC and JAINDX files. MARKER : The eight character string associated with the logical record; LOC : The absolute word address of the first element of the record on the JOBARC file; SIZE : The size of the logical record in integer words; NRECS : The total number of physical records in the JOBARC file; IRECWD : The total number of integer words per physical record; IRECLN : Apparently not used. LENGTH is declared in parameter statements in all routines which are related to JOBARC/JAINDX I/O (DUMPJA, GETREC, JASMRY, PUTREC and ZEROJA). All variables are integers, as usual. Used in all programs.

```
COMMON /LISTS/ MOIO(10,500),MOIOWD(10,500),MOIOSZ(10,500),
&                MOIODS(10,500),MOIOFL(10,500)
```

This common block contains all information needed to retrieve a list from disk. The five variables are indexed by the list number (right index) and the “sublist” value, which is usually (but not always!) the symmetry species (left index). The following information is stored : MOIO : The direct access physical record on which the list begins; MOIOWD : The total number of words which make up the list; MOIOSZ : The distribution size of the list [length of the left-hand two indices, which is equivalent to IRPDPD(IRREP,ISYTYP(1,LIST)), where IRREP is the left-hand index of MOIOSZ]; MOIODS : The number of distributions of the list [length of the right-hand two indices, which is equivalent to IRPDPD(IRREP,ISYTYP(2,LIST))]; MOIOFL : A value between 1 to 5 which refers to the specific file where the list resides [1=MOINTS (lists 1-100); 2=GAMLAM (lists 101-200); 3=MOABCD (lists 201- 300); 4=DERGAM (lists 301-400); 5=SECDER (lists 401-500)]. All values are integers. Used in post-SCF programs only.

```
COMMON /MACHSP/ IINTLN,IFLTLN,IINTFP,IALONE,IBITWD
```

This common block contains machine-specific information which is related to the machine word size. The meaning of the variables is as follows : IINTLN : Length of an integer in bytes; IFLTLN : Length of a floating point number in bytes; IINTFP : The ratio IFLTLN/IINTLN ; IALONE : The integer value with all bits set to "1" (*i.e.* 255 for IINTLN=4 and 65535 for IINTLN=8); IBITWD : 8*IINTLN/4. The most commonly used variable in this common block is undoubtedly IINTFP, as it is extremely useful in the coding of memory allocation. All variables are integers. Used in all programs.

```
COMMON /MACHSP2/ MASK1,MASK2,ISHFSZ
```

This common block contains information required for bit packing and bit unpacking of the CACHDIR variable used in common block /CACHINF/. MASK1 : 2**ISHFSZ-1, MASK2 : 7; ISHFSZ = 4*IBITWD-3. Used in post-SCF programs only.

```
COMMON /SYM/ POP(8,2),VRT(8,2),NT(2),NFMI(2),NFEA(2)
```

This common block contains information about the distribution of molecular orbitals among the various symmetry species. POP : The number of occupied orbitals by symmetry (left index) and spin (right index); VRT : The number of virtual orbitals by symmetry (left index) and spin (right index); NT : The lengths of both $\alpha\alpha$ and $\beta\beta$ virtual-occupied vectors *which are overall totally symmetric*; NFMI : The lengths of both $\alpha\alpha$ and $\beta\beta$ occupied-occupied vectors *which are overall totally symmetric*; NFEA : The lengths of both $\alpha\alpha$ and $\beta\beta$ virtual-virtual vectors *which are overall totally symmetric*. All variables are integers. Used in post-SCF programs only.

```
COMMON /SYMINF/ NSTART,NIRREP,IRREPS(255,2),DIRPRD(8,8)
```

This common block contains some rudimentary information which pertains to molecular symmetry. The meaning of the variables is as follows : NSTART : Not used; NIRREP : The number of symmetry species in the computational point group; IRREPS(255,2) : Not used; DIRPRD(8,8) : The direct product table for the symmetry species of the computational point group. All variables are integers. Used in post-SCF programs only.

```
COMMON /SYMPOP/ IRPDPD(8,22),ISYTYP(2,500),ID(18)
```

This common block contains information which is extremely useful in the coding of contractions. The meaning of the variables is slightly complicated, but worth learning if any serious coding is being planned. The IRPDPD array contains information about the total population of all possible two index arrays, by symmetry block. The right-hand index of IRPDPD refers to the specific type of two-index quantity (such as ab , $a\bar{i}$, $\bar{i}\bar{j}$ etc.), while the left-hand index refers to the symmetry block. The 22 different binary combinations are listed in the subsection entitled “Symmetry Types” below. However, these need not be consulted that frequently, since the ISYTYP(2,500) array contains these values for all lists on disk. The left-hand index of ISYTYP refers to the “side” of the list (left-hand or right-hand set of two indices, as stored) and the right-hand index refers to the list number. For example, ISYTYP(1,44) tells you the symmetry type (an integer between 1 and 22) for the left-hand two indices of list 44 (which happens to be $a \leq b$). Hence, to determine the number of $a \leq b$ combinations belonging to symmetry species 3 (this means that $\Gamma_a \otimes \Gamma_b = 3$), one need only inspect the value of IRPDPD(3,ISYTYP(1,44)). Use of these variables is perhaps most easily learned by studying pieces of the ACES II source code, particularly programs such as vcc.f, lambda.f and dens.f. The variable ID is not used. All variables are integers. Used in post-SCF programs only.

This concludes the list of common blocks which are initialized by the CRAPSI subroutine and are therefore used in most programs. Of course, a number of other common blocks are used, but these are necessarily specific to the program in which they are used and knowledge of their content is usually not necessary for program interfacing or development.

F. Symmetry Information

Some useful information regarding the molecular symmetry is stored on the JOBARC file and is used by various programs in the ACES II program system. In order to understand old code or to write new code which uses these data structures, a careful reading of this section is strongly recommended.

For each molecule, two point groups are considered. The first is the actual point group to which the molecule belongs, and is called the “full” point group (FPG). The second point group is an Abelian subgroup of the FPG, and is termed the “computational” point group (CPG), since it is the group in which the electronic structure calculation is actually performed. The FPG and the CPG are both determined by the *JODA* program in subroutine SYMMETRY, and this information is written to the output file. For each group, a large quantity of information regarding the effects of the operations on the atoms and the operations themselves are also determined by *JODA* in subroutine SYMDRV and its dependents.

F..1 Point Groups

The FPG and CPG are stored on JOBARC in records FULLPTGP and COMPPTGP, respectively. The order of each group is stored in records FULLORDR and COMPORDR, respectively. It is usually necessary to know the order of the point group before reading other JOBARC records dealing with symmetry.

F..2 Symmetry Operations

Subroutine CHRTAB in *JODA* forms 3×3 Cartesian representations for *all* symmetry operations in the FPG and the CPG and writes them to the JOBARC file records FULLSYOP and COMPSYOP, respectively. In subsequent programs, these operations are indexed by number and are ordered by class as follows:

C_1 group

Operation 1 is the identity.

C_s group

Operation 1 is the identity, operation 2 is σ_{xy} .

C_i group

Operation 1 is the identity, operation 2 is i .

C_n groups

Operation j corresponds to C_n^j .

D_n groups (n even)

Operation Number	Operation Type
1 thru $n - 2$	$C_n^{i **}$
$n - 1$	$C_2 (=C_n^{\frac{n}{2}})$
n thru $n - 1 + \frac{n}{2}$	C_2'
$n + \frac{n}{2}$ thru $2n - 1$	C_2''
$2n$	E

S_{2n} groups

Operation Number	Operation Type
------------------	----------------

1 thru n	$S_{2n}^{2i-1 *}$
$n + 1$ thru n	$C_n^{i *}$

D_n groups (n odd)

Operation Number	Operation Type
------------------	----------------

1 thru $\frac{n}{2} - 1$	$C_n^{i **}$
$\frac{n}{2}$ thru $2n - 1$	C_2
$2n$	E

C_{nh} groups ¹.

Operation Number	Operation Type
------------------	----------------

1 thru n	S_n
$n + 1$ thru $2n$	$C_n^{i *}$

C_{nv} groups (n even)

Operation Number	Operation Type
------------------	----------------

1 thru $n - 2$	$C_n^{i *}$
$n - 1$	C_2
n thru $n - 1 + \frac{n}{2}$	σ_v
$n + \frac{n}{2}$ thru $2n - 1$	σ_d
$2n$	E

C_{nv} groups (n odd)

¹The S_n operations are ordered as follows: $S_n, S_n^{n+2}, S_n^3, S_n^{n+4} \dots S_n^{2n-1}$ (n odd); $S_n, S_q, S_n^3, S_q^{q+2}, S_n^5, S_q^3 \dots S_n^{n-1}, E$ (n even, where $q = \frac{n}{2}$)

Operation Number	Operation Type
---------------------	-------------------

1 thru $\frac{n}{2} - 1$	C_n^{i*}
$\frac{n}{2}$ thru $2n - 1$	σ_v
$2n$	E

D_{nh} groups (n even) ²

Operation Number	Operation Type
---------------------	-------------------

1 thru $n - 2$	C_n^{i**}
$n - 1$	C_2
n thru $2n - 3$	S_n
$2n - 2$	i
$2n - 1$ thru $2n + \frac{n}{2} - 2$	C_2'
$2n + \frac{n}{2} - 1$ thru $3n - 2$	C_2''
$3n - 1$ thru $3n + \frac{n}{2} - 2$	σ_v
$3n + \frac{n}{2} - 1$ thru $4n - 2$	σ_d
$4n - 1$	σ_h
$4n$	E

D_{nh} groups (n odd) ³.

Operation Number	Operation Type
---------------------	-------------------

1 thru $n - 1$	C_n^{i**} pair of operations.
n thru $2n - 2$	S_n
$2n - 1$ thru $3n - 2$	C_2
$3n - 1$ thru $4n - 2$	σ_v
$4n - 1$	σ_h
$4n$	E

D_{nd} groups (n even)

² S_n operations ordered as in C_{nh} , n even.

³ S_n operations ordered as in C_{nh} , n odd

Operation Number	Operation Type
------------------	----------------

1 thru $n - 2$	$C_n^{i **}$
$n - 1$	$C_2 (= C_n^{\frac{n}{2}})$
n thru $2n - 2$	$S_{2n}^{2i-1 **}$
$2n - 1$	i
$2n$ thru $3n - 1$	C_2'
$3n$ thru $4n - 1$	σ_d
$4n$	E

D_{nd} groups (n odd)

Operation Number	Operation Type
------------------	----------------

1 thru $n - 1$	$C_n^{i **}$
n thru $2n - 2$	$S_{2n}^{2i-1 **}$
$2n - 1$	i
$2n$ thru $3n - 1$	C_2
$3n$ thru $4n - 1$	σ_d
$4n$	E

T group

Operation Number	Operation Type
------------------	----------------

1 thru 4	C_3
5 thru 8	C_3^2
9 thru 11	C_2
12	E

T_d group

Operation Number	Operation Type
------------------	----------------

1 thru 6	S_4
7 thru 9	C_2
10 thru 17	C_3
18 thru 23	σ_d
24	E

O_h group

Operation Number	Operation Type
------------------	----------------

1 thru 3	C_4
4 thru 6	S_4
7 thru 9	σ_h
10 thru 12	C_4^2
13 thru 15	C_4
16 thru 18	S_4
19 thru 24	C_2
25 thru 30	σ_d
31 thru 38	C_3
39 thru 46	S_6
47	i
48	E

O group

Operation Number	Operation Type
------------------	----------------

1 thru 3	C_4
4 thru 6	C_2
7 thru 9	C_4
10 thru 15	C_2'
16 thru 23	C_3
24	E

T_h group

Operation Number	Operation Type
------------------	----------------

1 thru 3	σ_d
4 thru 6	C_2
7 thru 10	C_3
11 thru 14	C_3^2
15 thru 18	S_6
19 thru 22	S_6^2
23	i
24	E

I_h group

Operation Number	Operation Type
1 thru 6	C_5
7 thru 12	C_5^2
13 thru 18	S_{10}
19 thru 24	S_{10}^3
25 thru 30	C_5
31 thru 36	C_5^2
37 thru 42	S_{10}
43 thru 48	S_{10}^3
49 thru 68	C_3
69 thru 88	S_6
89 thru 103	C_2
104 thru 118	σ_v
119	i
120	E

I group

Operation Number	Operation Type
1 thru 6	C_5
7 thru 12	C_5^2
13 thru 18	C_5
19 thru 24	C_5^2
25 thru 44	C_3
45 thru 59	C_2
60	E

F..3 Permutation Vectors

For each symmetry operation in the CPG and FPG, a vector is constructed which associates each atom in the Z-matrix with its image under the transformation. Atoms are numbered according to their absolute position in the Z-matrix. For example, if operation i maps atom 4 into atom 17, the value of the fourth element of the permutation vector is 17. These vectors are stored in JOBARC records FULLPERM and COMPPERM, respectively. The set of permutation vectors is stored on disk as a single vector of length $N_{atoms}h$, where h is the order of the point group. Each permutation vector occurs sequentially in the composite record, and follows the ordering of the symmetry operations given in the preceding subsection.

F..4 Orbit-Specific Information

A group of symmetry equivalent atoms is called an *orbit*. Analysis of symmetry in terms of the orbits of a molecule is very useful and is used extensively in ACES II. Information on JOBARC dealing with the orbits is summarized in the following. Records FULLNORB and COMPNORB contain the value of the number of orbits in the FPG and CPG, respectively. Records FULLSTGP and COMPSTGP contain the site group (the subgroup under which the atoms in the orbit are mapped into themselves by all operations) of each orbit. Records FULLPOPV and COMPPOPV contain the number of atoms in each orbit. Finally, the records FULLMEMB and COMPMEMB order the Z-matrix centers by orbit, so that the FULLMEMB and FULLPOPV (or COMPMEMB and COMPPOPV) records can be used together to list the individual atoms in each orbit.

F..5 Symmetry Types

The post-SCF program modules of ACES II assign an integer value to specific binary combinations of orbital types (occupied or virtual; α or β) which is then used to determine symmetry information (see, for example, the description of the SYMPOP common block). A complete listing of the assignments for these values is given below

“Symmetry Number”	Distribution Type
1	$a < b$
2	$\bar{a} < \bar{b}$
3	$i < j$
4	$\bar{i} < \bar{j}$
5	$a \leq b$
6	$\bar{a} \leq \bar{b}$
7	$i \leq j$
8	$\bar{i} \leq \bar{j}$
9	ai
10	$\bar{a}\bar{i}$
11	$a\bar{i}$
12	$\bar{a}i$
13	$a\bar{b}$
14	$i\bar{j}$
15	$a\bar{b}$
16	ia
17	$\bar{i}\bar{a}$
18	$i\bar{a}$
19	ab
20	$\bar{a}\bar{b}$
21	ij
22	$\bar{i}\bar{j}$

Note that entries 13 and 15 are identical, which is simply a result of carelessness in the initial coding. Although not all possible combinations are given, these are the only combinations which are used in the distributions which make up the integral lists.