

DIRAC04 使用说明

DIRAC 04.0 发布版 (2004 年 9 月)

Program

- for

Atomic

- and

Molecular

Direct

Iterative

Relativistic

All-electron

Calculations



Zork 译

qcband@gmail.com

2006 年 2 月

2019 年 4 月 8 日修订

目录

前言	1
致谢	2
第一章 DIRAC 04 安装指南	3
1.1 硬件和软件支持	3
1.2 安装程序	3
1.3 维护	4
1.3.1 新版本	4
1.3.2 报告错误	4
1.3.3 用户支持	4
第二章 用户指南	5
2.1 运行程序	5
2.2 pam shell 脚本	5
2.3 pamq shell 脚本	6
2.4 一个简单的任务实例	6
2.5 编写自己的脚本文件	7
2.5.1 环境变量一览	8
2.6 测试的例子	9
第三章 DIRAC 菜单文件	10
3.1 **DIRAC——任务详细说明	11
3.1.1 OPTIMIZE——几何优化指令	11
3.2 **GENERAL——基本输入	12
3.2.1 示例输入	13
3.3 **INTEGRALS——积分指令	13
3.3.1 *READIN——读取基组文件	14
3.3.2 *ONEINT——单电子积分	14
3.3.3 *TWOINT——双电子积分	15
3.3.4 示例输入	15
3.4 **HAMILTONIAN——定义哈密顿量	16
3.4.1 *DFT——DFT 指令	18
3.4.2 示例输入	19
3.5 **WAVE FUNCTIONS——得到波函	19
3.5.1 *DHFCAL——Hartree-Fock/Kohn-Sham 计算	20
3.5.2 示例输入	26
3.5.3 *MP2CAL——MP2 计算	27
3.5.4 *RESOLVE——求解开壳层态	28
3.5.5 *MVOCAL——修改的虚轨道	29
3.6 **ANALYZE——分析波函	29
3.6.1 *PRIVEC——矢量打印	29
3.6.2 *MULPOP——Mulliken 布居分析	30
3.6.3 *PROJECTION——投影分析	31
3.6.4 *DENSIT——产生用于显示的密度	31
3.6.5 *RHO1——沿着成键打印密度	34

3.6.6 示例输入.....	34
3.7 **PROPERTIES——特性模块	35
3.7.1 *EXPECTATION VALUE——求解期待值	37
3.7.2 *LINEAR RESPONSE——线性响应	37
3.7.3 *NMR——控制 NMR 参数的求解.....	40
3.7.4 *MOLGRD——控制分子梯度的求解.....	40
3.7.5 *QUADRATIC RESPONSE —二次响应函数.....	40
3.7.6 输入示例.....	42
3.8 **MOLTRA——积分转换模块	42
3.8.1 *PRPTRA——特性积分变换.....	44
3.8.2 示例输入.....	44
3.9 RELCCSD——耦合簇模块.....	44
3.9.1 RELCCSD——基本输入和控制输入.....	45
3.9.2 CCENER——能量模块的输入	46
3.9.3 CCSORT——分类模块的输入.....	46
3.9.4 CCFOPR——一阶特性的输入.....	46
3.9.5 CCFSPC——Fock 空间模块的输入	47
3.9.6 RELCCSD 的示例输入.....	48
3.10 DIRRCI——直接 CI 模块	48
3.10.1 RASORB——指定 CI 的类型和活性空间	48
3.10.2 CIROOT——选择进行收敛的电子态	49
3.10.3 DIRECT——收敛控制	49
3.10.4 OPTIM——算法的精细调整	49
3.10.5 LEADDET——分析 CI 波函	50
3.10.6 DIRRCI 的示例输入	50
3.11 GOSCI——COSCI 模块.....	50
3.11.1 GOSCI——定义 CI 空间.....	50
3.11.2 POPANA——波函分析	51
3.11.3 GOSCI 的示例输入.....	51
3.12 LUCITA——直接 GAS CI 模块	51
3.12.1 一般输入.....	52
3.12.2 特殊输入.....	53
3.12.3 LUCITA 的示例输入	53
3.13 特殊功能.....	54
3.13.1 轨道串.....	54
第四章 DIRAC 基文件	56
4.1 标题与对称性.....	56
4.2 原子坐标.....	57
4.3 大分量基组.....	58
4.3.1 来自基组库的基组.....	58
4.3.2 直接输入基组.....	58
4.3.3 MOLFDIR 型基组.....	60
4.3.4 平缓 (even-tempered) 基组 (几何级数)	61
4.3.5 均匀 (well-tempered) 基组.....	62

4.3.6 同类基组.....	63
4.3.7 双同类基组.....	64
4.4 小分量基组.....	65
4.4.1 利用动能平衡.....	65
4.4.2 直接输入基组.....	66
4.4.3 MOLFDIR 型基组.....	66
第五章 输出文件.....	67
5.1 重新开始计算.....	67
第六章 定义单电子算符.....	68
6.1 定义算符的语法.....	68
6.2 单电子算符列表.....	70
附录 A. 最近的修正.....	73
附录 B. 示例文件.....	74
参考文献.....	78

前言

这是相对论从头电子结构程序 DIRAC 04 的使用文档。

源代码和程序的使用、复制、传递或存储受使用许可的保护（见 <http://dirac.chem.sdu.dk>）。源代码的任何一部分以及编译好的模块禁止向使用许可所有者以外的研究团体发布。这意味着离开持有使用许可研究团体的人员（如博士后），不能带走 DIRAC 04 的任何一部分，包括修改过的文件，除非他或她已经获得了自己的使用许可。

关于 DIRAC 程序，版权，以及如何获得使用许可的问题，见 <http://dirac.chem.sdu.dk>，或者发邮件给：dirac-admin@dirac.chem.sdu.dk。

在公开的出版物中使用本程序的结果，需要引用：

“DIRAC 04, a relativistic ab initio electronic structure program, Release DIRAC04.0 (2004), written by H. J. Aa. Jensen, T. Saue, and L. Visscher with contributions from V. Bakken, E. Eliav, T. Enevoldsen, T. Fleig, O. Fossgaard, T. Helgaker, J. Laerdahl, C. V. Larsen, P. Norman, J. Olsen, M. Pernpointner, J. K. Pedersen, K. Ruud, P. Salek, J. N. P. van Stralen, J. Thyssen, O. Visser, and T. Winther.”

本程序是实验性的代码，正在不断开发中。这意味着不能保证结果完全正确，或是适于特殊的用途。

致谢

我们感谢由 K. G. Dyll, K. Faegri Jr.和 A. S. P. Gomes 所进行的基组开发和测试。我们还要感谢 S. Dubillard, U. Ekström, I. Infante, J. Henriksson 和 C. R. Jacob 所做的端口和 BETA 版测试。

第一章 DIRAC 04 安装指南

1.1 硬件和软件支持

DIRAC 可运行于多种不同的 UNIX 操作系统中。DIRAC04 发布版已在 Linux, SGI, IBM AIX, 苹果 OsX, Cray, HP, Sun, Dec 等上进行测试。

程序的大部分用 Fortran 77 编写, 虽然主程序和一些工具程序用 C 语言书写。依赖于计算机 (编译器) 的功能由 C 预处理程序控制。所有的浮点运算以双精度 (64 位) 执行。

DIRAC04 也可以运行于其它的 UNIX 环境中, 我们鼓励用户反馈成功的移植信息。

DIRAC04 尚未移植到任何 WINDOWS 操作系统中。

1.2 安装程序

程序可以安装在各种不同的计算机中。程序是一系列.F 主文件, 它们由 `cpp` C 预处理程序进行处理, 来控制与计算机有关的特性。安装由 UNIX 的 `make` 命令生成器控制。

程序的安装如下:

- 程序系统作为用 `gzip` 压缩的 `tar` 文件 `DIRAC04.tar.gz` 提供。用以下命令解压缩程序:

```
tar -zxvf DIRAC04.tar.gz
```

在某些机器上, 或者

```
gunzip DIRAC04.tar.gz
```

```
tar xvf DIRAC04.tar
```

源代码将创建目录 `DIRAC04`。
- `cd DIRAC04`
- 查看 `README` 文件的内容。
- 现在运行 `configure` 脚本

```
./configure
```

 - o 这个脚本将自动猜测您的计算机/体系结构。
 - o `configure` 将用依赖于系统的信息创建 `Makefile.config`, 用于 `make` 和开始 DIRAC 计算的 `pam shell` 脚本。
 - o `configure` 的帮助用:

```
configure -help
```

注意 `configure` 可以调用标记, 例如选择编译器或用于数学库的路径。
- 检查 `Makefile.config`, 如果需要的话进行编辑。
- 如果想使用队列系统: 检查 `pamq`, 如果需要的话进行编辑。
- 可选: 运行 `make depend`:

```
make depend
```

这将寻找每个文件的依赖关系, 即, 如果改变了一个头文件, 那么所有使用这个头文件

的 Fortran 和 C 文件都将被重新编译。对于程序员，或者您要改变公共区中的某些维数，这是非常有用的。

- 运行 make 进行编译/安装：
make

1.3 维护

1.3.1 新版本

如果发现并修改了 BUG，并且加入了新功能，将发布新版本。这些升级将在邮件列表中公布，可以在 DIRAC 主页下载。

DIRAC 正在进行不断的开发，一些新的功能可能不会在您访问到的发布版中出现。如果您对在 DIRAC 作者群中开发的未发布功能感兴趣，可以直接和作者联系。如果作者决定移交某些新的代码，这将受到未在许可协议中规定的限制。

1.3.2 报告错误

鼓励用户报告 BUG。既可以联系出错模块的作者，也可发信给 dirac-admin@dirac.chem.sdu.dk。

1.3.3 用户支持

虽然可以利用邮件列表(dirac@dirac.chem.sdu.dk)，但是还没有建立用户支持。所有的许可文件持有者都被添加到邮件列表中。另外订阅：发信给 dirac-request@dirac.chem.sdu.dk，使用标题“Subscribe”。作者将收到来自邮件列表的信件，但是我们不能保证任何作者都会回复邮件列表中的问题。相反，我们鼓励用户之间相互帮助。请检查从前信息的存档，位于 <http://dirac.chem.sdu.dk/>。

第二章 用户指南

2.1 运行程序

运行 DIRAC 需要两个输入文件：

- **基文件：**定义基组，原子核坐标和对称性。
- **菜单文件：**定义计算。

在计算的开始，先处理基文件，接下来各个模块按照菜单文件给出的信息被激活。因此可以运行一系列基于同一个基文件的计算。对于这种情况推荐保存 MO 因子文件 DFCOEF，它可以重新开始计算。例如，可以先运行一个 DC-HF 计算，给出定义收敛波函的一组 MO 因子。布居数分析可以在另外的计算中进行，接下来的计算除了基文件以外，只需要因子文件。

2.2 pam shell 脚本

脚本 **pam** 用于自动计算。调用 **pam** 不需要任何标记。给出标记的输出如下（本文档中作了修改）：

用法：pam [标记] mol[.mol] menu[.inp]

标记：

-mpi n	用 n 个节点运行 MPI；默认为串行
-machinefile 文件名	指定文件，它包含在 MPI 运行中所用机器的列表
-mpiarg 字符串	传递给 MPI 运行的其它参量
-mw mem	为主节点（如果未指定-nw mem，还包括从节点）设置内存（单位：MW）
-nw mem	为从节点设置内存（单位：MW）
-basdir 目录	把基组目录预加到搜索列表中（可以重复）
-wrkdir 目录	工作目录；默认为/scr/jth/. \$\$
-clean	在计算前清除工作目录
-keepwrk	在计算后保存工作目录（默认：清除）
-copy "文件1 文件2.."	从任务目录复制指定的文件到工作区
-get "文件1 文件2.."	从工作区获得指定的文件到任务目录
-incmo	从任务目录复制 DFCOEF 到工作区（当重新开始计算时）
-outcmo	从工作区保存 DFCOEF 到任务目录（用于重新开始计算）
-infck	从任务目录复制 DFFCK2 到工作区
-outfck	从工作区保存 DFFCK2 到任务目录
-dirac dirac_cmd	用 dirac_cmd 作为可执行程序，代替默认的 dirac.x
-run name	输出文件的后缀；默认为 out
-nice	用 nice/npri 执行 DIRAC
-notify	当完成后，通知使用者进行的任务

-user	<i>e-mail</i>	当完成后，使用指定的 e-mail 地址通知使用者
-debug		在调试器 gdb 中运行 DIRAC
-ssrun	<i>exp_type</i>	使用 <i>ssrun exp_type</i> (SGI 模仿器) 运行 DIRAC (如果可执行程序不在输入的路径中，那么必须包含路径)
-help		比默认更多的关于 pam 的帮助

使用 **pam** 脚本，一个输入文件为 **Au2.mol** 和 **ccsd.inp** 的计算任务可以用下面的 shell 脚本开始：

```
pam Au2 ccsd
```

重新开始的例子是下面这个对碘苯的 DFT 计算，其中保存 MO 因子用于分析，使用的输入文件为 **C6H5I.mol**，**B3LYP.inp**，和 **analysis.inp**：

```
pam -outcmo C6H5I B3LYP
```

```
pam -incmo C6H5I analysis
```

2.3 pamq shell 脚本

人们常想把 **pam** 脚本提交到批处理队列系统中，而不是像在 **pam** 中那样交互运行 DIRAC。要在这样的环境中简化运行 DIRAC，可以使用 **pamq** 脚本，它将产生适当的输入文件，并考虑任务的提交。这个脚本使用与 **pam** 相同的脚本，此外还有一些与最大总时间和指定用户选定序列有关的额外变量。调用 **pamq** 不需要任何标记。给出标记的输出如下(但采用了本地默认值)：

用法: `pamq [pam 标记] [pamq 标记] mol[.mol] menu[.inp]`

<i>pam</i> 标记:	调用没有任何选项的 pam ，得到 pam 的标记列表
<i>pamq</i> 标记:	
-q 队列	提交任务给指定的队列；默认为 default
-qsys queuesys	使用哪个队列系统；默认为 pbs。 已知的队列系统有: pbs, bsub, LoadLeveler, XGrid
-t time	任务的时间限制；默认为 15m。 可以指定为小时 (如在 20h 中) 或分钟 (如在 30m 中)
-mpi n	用 n 个节点运行 MPI；默认为 8
-mw mem	为主节点 (如果未指定 -nw mem, 还包括从节点) 设置内存 (单位: MW)；默认为 100
-nw mem	为从节点设置内存 (单位: MW)；默认为 100

运行

```
pamq -mpi 6 -t 2h C2H4Cl2_ec1_c2v lda
```

将提交任务脚本到默认的队列，时间限制为 2 小时，并请求 6 个处理器。例如，输入文件可以是下一部分说明的文件 (虽然这个例子实际上并不需要 6 个节点)。

2.4 一个简单的任务实例

这一部分讲述如何指定重叠构象下二氯乙烷的 DFT 计算。在这个例子中，DIRAC 基文件 **C2H4Cl2_ec1_c2v.mol** (以 # 开始的行是注释) 为：

```

DIRAC
1,2-dichloro-ethane in eclipsed_1 conformation (Cl eclipsed), angles sp3
Full symmetry : C2v
# Number of different atoms, symmetry specification
C   3   2X Y   A
# Atom 1: charge and number of distinct atoms of this type
      6.   1
# Name of atoms and coordinates
C  -0.7000 0.0000 0.0000
# Basis set (taken from the library)
LARGE BASIS cc-pVDZ
      17.   1
Cl  -1.3000 0.0000 1.6971
LARGE BASIS cc-pVDZ
      1.   1
H   -1.0667 0.8981 -0.5185
LARGE BASIS cc-pVDZ
FINISH

```

(这个基文件还用于 test 目录下的测试 29。)基文件的精确格式在第四章给出。菜单文件 lda.inp 中的 DIRAC 输入说明为:

```

**DIRAC
.TITLE
LDA energy calculation
.WAVE F
**HAMILTONIAN
# Use Simple Coulombic Correction to model effect of (SS|SS) integrals
# This is highly recommended for the calculation of spectroscopic
# constants and valence properties
.LVCORR
# Use Density Functional Theory
.DFT
LDA
**WAVE FUNCTIONS
.DHF
*END OF INPUT

```

2.5 编写自己的脚本文件

有些用户更喜欢用自己编写的脚本文件运行 DIRAC04。运行可执行程序 dirac.x 的细节直接在下面给出。运行程序的基本 UNIX 命令为:

```
dirac.x > {输出文件}
```

其中菜单文件必须是 DIRAC.INP, 基文件必须是 MOLECULE.INP。

各个程序模块有不同的内存要求。通过设置环境变量 DIRWRK 可以修改运行期间可用的

内存，它定义了计算中需要的 8 位字节的内存量。例如：

```
setenv DIRWRK 30000000
```

表示分配 30 MW 内存¹。对于并行任务，节点的内存可以用类似的变量 DIRNOD 设置。注意，在主节点给定这些环境变量就足够了；它们将会被传递到从节点上。并行版本既可以进行并行任务，也可以进行串行任务。如果变量 DIRPAR 直接设置为 0，那么 `dirac.x` 串行执行，否则并行计算。默认的内存存在 `configure` 脚本中指定。

2.5.1 环境变量一览

DIRWRK	主节点以 8 字节为单位的内存大小，如果没有定义 DIRNOD，也是从节点的内存大小。例如， <pre>setenv DIRWRK 30000000</pre> 分配 30 MW 内存，近似相当于 228 MB。
DIRNOD	从节点以 8 字节为单位的内存大小。大多数情况下，由于主节点需要的内存多于从节点，因此一般给从节点分配比主节点少一些的内存。对于集群类型的计算机，如 IBM SP 或 Beowulf，这是很有用的，你可以在具有大内存的主节点上运行主要计算，在一般节点上运行从属计算。它还可以用于共享内存计算机，如 SGI Origin，用于减少分配的总内存。在 CC 运行中，人们不想选择与 DIRWRK 不同的 DIRNOD，因为并行化假定节点全部相同，不再使用主-从节点的算法。代码仍旧执行，但由于加载平衡的问题，执行效率不是最佳。
DIRPAR	如果 DIRPAR 设置为 0，即： <pre>setenv DIRPAR</pre> 并行版本的 DIRAC 可以在串行模式下运行（不需要调用任何 MPI 例程）。如果 DIRPAR 没有设置或设为非零的值，将在并行模式下运行。
BASDIR	DIRAC 寻找基组的路径。BASDIR 可以包含多个用冒号分开的目录。对于用户 <code>jth</code> （当 DIRAC04 安装在 <code>/home/jth/prog/DIRAC04</code> ），在 <code>pam shell</code> 脚本提供的默认路径为： <pre>setenv BASDIR ".:/home/jth/prog/DIRAC04/basis:/home/jth/prog/DIRAC04/basis_dalton"</pre>

用户 `jth` 用 DIRAC 运行任务实例的 `shell` 脚本在 2.4 节有说明，可能的形式为：

```
#!/bin/tcsh

setenv SCR /scr/jth

cp lda.inp $SCR/DIRAC.INP
cp C2H4C12_ec1_c2v.mol $SCR/MOLECULE.INP

setenv DIRWRK 10000000

cd $SCR
```

¹ 1MW=8,000,000 字节=7.63 MB，1MB=(1024)² 字节。

```
$HOME/DIRAC04/dirac.x > exciting_output
```

2.6 测试的例子

DIRAC 提供了大量测试的例子。这些测试位于 DIRAC04/ 目录下的 ./test 中。在 DIRAC04/test/ 目录里有许多子目录，每个子目录都包含一个测试任务和一个描述测试的 README 文件。这些测试还可以作为自己设置计算的起点。

所有的测试可以用 testlast 脚本自动运行，它对安装的正确性提供检测。可以很简单地在主目录输入

```
testlast
```

进行调用。接下来它将交互运行，并在 test/logs 目录中创建所有运行的日志文件。检查文件 test/logs/lastrun 看是否发现了错误。这一运行在所有正确配置的计算机上应该在 5 分钟内完成。更高级的测试可以为 testlast 指定选项来完成。输入

```
testlast -help
```

查看所有选项。

第三章 DIRAC 菜单文件

DIRAC 菜单文件定义了计算，一般结构为：

```
**DIRAC
  <必需：任务的说明；指定激活哪些模块，如 wave functions, properties, 或 analysis>
**GENERAL
  <可选：被研究体系的一般输入及其描述，例如光速>
**HAMILTONIAN
  <可选：指定电子哈密顿量，例如无自旋的哈密顿量>
**INTEGRALS
  <可选：HERMIT 积分生成程序的指令>
**WAVE FUNCTION
  <可选：如果模块在**DIRAC 中被激活，描述波函数>
**ANALYZE
  <可选：如果模块在**DIRAC 中被激活，描述波函分析>
**PROPERTIES
  <可选：如果模块在**DIRAC 中被激活，描述特性计算>
**MOLTRA
  <可选：如果激活该模块，改变描述 2 指数或 4 指数积分转换的默认值>
**END OF DIRAC
```

菜单文件分成多个部分，每一部分都以关键字形式 XXXXXX 开始。只有第一部分**DIRAC 是强制的。某些部分(如**HAMILTONIAN)如果出现就会读取；而其它的则由**DIRAC 部分的关键字调用。每一部分都有一般结构：

```
**XXXXX
.<关键字>
* <小节>
.<关键字>
* <小节>
.<关键字>
* <小节>
.<关键字>
# 一个注释
! 另一个注释
...
```

每个小节由**XXXXX 形式的关键字调用。对每个小节，可以指定一组关键字，以及可能带有的自变量。用户可以很灵活地用几组小节和关键字定义当前的计算。

注释行(!或#在第一列开头)将被忽略。每个关键字只使用前 6 个字符，但是为了增加可读性，可以写得更多。

所有部分、小节和关键字在下面的部分说明。关键字分为“基本选项”，也就是必需的或者用得最多的关键字，“高级选项”(可选的关键字)以及“程序员选项”(激活调试或测试的功能)。

几乎每个部分或小节都有一个打印级别的标记，用于为某一给定的模块指定有多少输

出。标准设置给出足够的输出，因此不需要改变，因为这可能产生庞大的输出。典型的打印级别 5 将打印非常详细的信息，如完整的 Fock 矩阵，等。

3.1 **DIRAC——任务详细说明

基本选项 DIRAC 默认只激活生成基组（例如小分量函数的动能平衡条件）和单电子模块。原子基函数的双电子积分将在以下任务模块需要时才计算。¹

. WAVE FUNCTION

激活波函模块。此命令激活**WAVE FUNCTION 部分的读取，其中必须指定所需的波函类型（默认：无）。

. ANALYZE

激活 Hartree-Fock 分析模块。此命令激活**ANALYZE 部分的读取。

. PROPERTIES

激活特性模块（将调用进行特性积分计算的积分模块）。此命令激活**PROPERTIES 部分的读取。

. OPTIMIZE

激活几何优化。此命令激活*OPTIMIZE 小节的读取。

. 4INDEX

激活把积分转换到分子轨道基。此命令激活**MOLTRA 部分的读取。

这些转换的积分目前仅能用于 RELCCSD, DIRRCI, 和 LUCITA 模块，如果在**WAVE FUNCTION 下请求此三个模块之一，那么自动激活该标记，除非在本输入模块中定义了.NO4INDEX。

默认包含轨道能量在-10 和+20 a. u. 之间的分子轨道，这可以在**MOLTRA 输入部分修改。

. NO4INDEX

如果在**WAVE FUNCTION 下请求 RELCCSD, DIRRCI, 和 LUCITA 中的任何一个，不自动激活到分子轨道基的积分转换。

高级选项

. TITLE

标题行。

参量：标题行（最多 50 个字符）

默认：DIRAC: No title specified !!!

. INPTEST

输入测试：不调用任务模块。

默认：不激活输入测试

这在提交任务之前，为了检查输入文件的配置是否正确而打开一组新的输入测试计算，这经常是很有用的。

. ONLY INTEGRALS

在哈密顿单电子积分和**INTEGRALS 定义的单电子积分后停止计算。积分将被写到硬盘。

3.1.1 OPTIMIZE——几何优化指令

这一部分控制几何优化。几何优化算法基于 DALTON 中的一种，因此指令与 DALTON 相同（参见：<http://www.kjemi.uio.no/software/dalton/dalton.html>），除了以下

¹ 如果想把双电子积分保存到硬盘用于其它目的，我们建议使用 DALTON 程序包，见 <http://www.kjemi.uio.no/software/dalton/dalton.html>。这在 DIRAC 中还不可以。

一点变化:

- 由于没有执行分子的 Hessian, 所以不能使用二阶算法。
- 一些新的关键字 (见下):

如果在任务的输入部分 .PROPERTIES (.ANALYZE) 与 .OPTIMIZE 一起指定, 那么在每次优化迭代和在收敛结构中调用特性(分析)模块。在每次迭代中, 计算**PROPERTIES (**ANALYZE) 中请求的特性(分析), 在收敛结构中, 计算**PRP F (**ANALYZE) 中给出的特性(分析)。

.NO SKIP 见*MOLGRD 部分关于 .TRICK 的说明。在几何优化中默认激活 “trick”, 因为在当前结构远离平衡时, 也就是梯度的模比如说为 1 时, 此时不需要计算 LS 和/或 SS 双电子梯度, 因为它们的模大约为 0.001。
该关键字强制在所有的结构迭代中总是求解 LS 和 SS 双电子梯度 (依赖于积分标记)。

.NUMGRA 在几何优化中强制使用数值梯度, 例如在 Hartree-Fock 计算中。

.TWOGRD 见*MOLGRD 部分的说明。

默认: TWOGRD(1) = TWOGRD(2) = TWOGRD(3) = 1

3.2 **GENERAL——基本输入

这一部分允许描述当前运算的全局特性。

基本选项

高级选项

.ACMOUT 把 C_i 对称性的因子转存到无格式文件 DFACMO 中。这可能会很有用, 例如, 当进行投影分析时: 分子在其最高对称性进行计算, 所含的每一原子类型在线性对称性下计算 (在基文件中不指定对称操作, 见第四章), 接下来在 C_i 对称性下执行分析。

.DIRECT Fock 矩阵的双电子积分直接求解。(其它模块, 如 CI, CCSD, 和 MCSCF 使用的所有双电子积分总是直接求解而不是从硬盘读取。)

参量: 整数 ILL, ISL, ISS

IXX = 1 (打开) / 0 (关闭) (XX = LL, SL 或 SS)

默认: ILL = ISL = ISS = 1

.DOJACO 使用 Jacobi 方法进行矩阵对角化。

默认使用 Householder 方法, 一般更有效。但是 Householder 方法可能会混合不同对称性的简并本征矢。Jacobi 方法目前仅限于实群。

默认: 进行 Householder 对角化。

.PCMOIN 从格式化文件 .DFPCMO 中读取分子轨道因子。这对于具有不同二进制结构的机器之间传递因子, 例如在 Silicon Graphics 和 IBM 之间, 是非常有用的。参见关键字 .PCMOUT。

.PCMOUT 把 DHF 分子轨道因子写入格式化文件 .DFPCMO 中。这对于具有不同二进制结构的机器之间传递因子, 例如在 Silicon Graphics 和 IBM 之间, 是非常有用的。参见关键字 .PCMOIN。

.LINDEP 大、小分量的线性依赖关系阈值; 与重叠矩阵本征值的最小允许值有关。

参量: 实型 STOL(1), STOL(2)

默认: 大分量: STOL(1) = 1.0D-6

小分量: STOL(2) = 1.0D-8

. SPHTRA

转换为标准正交基时的球谐转换, 将删除全部对称性的贡献。

参量: 整型 ISPHL, ISPHS = 1 (打开) / 0 (关闭)

默认: ISPHL = 1, ISPHS = 1

ISPHL 和 ISPHS 分别指定大、小分量的球变换。大分量的变换是从笛卡尔函数到球谐函数的标准转换。而小分量的变换是根据动能平衡关系的限制进行修正的。

程序员选项

. PRINT

一般打印级别。

参量: 整型 IPRGEN

默认: IPRGEN = 0

. CVALUE

设置光速值为原子单位。

参量: CVAL

默认: CVAL = 137.0359998

3.2.1 示例输入

以下是这一节的示例输入:

```
**GENERAL
# Use a reduced value for the speed of light to amplify
# the effect of relativity (for testing)
.CVALUE
50.0
```

3.3 **INTEGRALS——积分指令

这一部分给出了 HERMIT 积分代码的指令。

基本选项

. NUCMOD

核模型。

参量: 整型 INUC

INUC = 1: 点核

INUC = 2: 高斯电荷分布

默认: INUC = 2

要把 Lévy-Leblond 型的计算与其它代码, 如 DALTON 所作的非相对论计算结果进行比较, 点核模型非常有用。两种方法应当给出完全相同的能量。对于高斯电荷分布, 默认的指数与 Visscher 和 Dyll 在文献[1]中建议的值一致, 参见网页

<http://dirac.chem.sdu.dk/doc/FiniteNuclei/FiniteNuclei.shtml>。如果需要, 还可以在基文件中指定其它指数 (见 4.2 节)。

高级选项

- . SELECT** 在关于单原子中心（例如限制电场梯度）的单电子积分中，限制核的范围。
参量：整型 NPATOM——原子中心的数量
参量：如果 NPATOM 非零，(IPATOM(I), I=1, NPATOM)
默认：使用全部原子
- . DIPORG** 包括偶极矩在内，所有力矩积分的原点（对中性体系，偶极与原点无关）。
参量：实型 (DIPORG(I), I=1, 3)
默认：原点 (0, 0, 0)
- . MAGCOR** 打印对称化的核磁矩；这对应于在每个核的中心获得转动的对称组合而不是坐标。
在标记各种磁积分时，使用了编号方式。

程序员选项

- . PRINT** HERMIT 中的一般打印级别。
默认：1

3.3.1 *READIN——读取基组文件

这一小节允许在读取基组文件时改变默认值（见第四章）。

高级选项

- . UNCONTRACT** 把（基组库中）定义为收缩的基组进行非收缩。
默认：只有小分量非收缩。

程序员选项

- . PRINT** 读取基组文件时的打印级别。
默认：来自**INTEGRALS 的打印级别
- . MAXPRI** 在基组文件的给定区域，原函数的最大数量。
参量：整型 MAXPRI
默认：MAXPRI = 22

3.3.2 *ONEINT——单电子积分

这一小节给出产生单电子积分的指令。基于其它部分的输入，程序将会决定计算何种积分。

程序员选项

- . PRINT** 单电子积分程序中的打印级别。
默认：来自**INTEGRALS 的打印级别

3.3.3 *TWOINT——双电子积分

这一小节给出产生双电子积分的指令。它还给出构造 Fock 矩阵的指令，例如屏蔽。

高级选项

- . SOFOCK** 在对称匹配(SO)基中，直接构造 Fock 矩阵。
默认：在原子轨道(AO)基中使用框架矩阵方法进行直接 Fock 矩阵构造。这可能会给出更好的屏蔽，但需要更多的内存。
- . ICEDIF** 把库仑屏蔽和交换屏蔽的贡献区分开[2]。
参量：整型 ICDIFF(库仑)，IEDIFF(交换) 打开 (1) / 关闭 (0)
默认：ICDIFF = IEDIFF = 1
- . SCREEN** Fock 矩阵积分直接计算的屏蔽阈值[2]。
参量：实型 SCRFCK
默认：SCRFCK= 1.0D-12。注意屏蔽阈值会影响收敛。一般屏蔽阈值必须比想要得到的收敛电子梯度的模要小约三个数量级。
- . THRFAC** 对 SL- 和 SS- 积分调整积分阈值。对于一般的积分计算，只有在 MOLECULE. INP 文件中（见第四章）给出的阈值以上的积分才会写到硬盘上。SSLL 和 SSSS 积分的阈值由这里给出的 THRFAC 定义。
参量：实型 THRFAC(1), THRFAC(2)
默认：THRFAC(1) = THRFAC(2) = 1

程序员选项

- . PRINT** 在双电子积分程序中，为特殊的壳层四元指数计算设置打印级别。打印级别仅对给定的壳层四元指数改变。零对应于所有的壳层，因此
. PRINT
4 0 0 0 0
或者仅仅是
. PRINT
4
将对所有的壳层四元指数设置打印级别为 4。使用时应注意避免大量的输出!!! 在打印级别为 15 时，会打印各个积分。
参量：1-5 个整数(IPRINT, IPRNTA, IPRNTB, IPRNTC, IPRNTD)
默认：由**INTEGRALS 和 IPRNTx=0 获得打印级别
- . TIME** 给出积分计算的详细时间。
默认：无时间给出

3.3.4 示例输入

```
**INTEGRALS
#Use point nucleus model to compare with nonrelativistic programs
.NUCMOD
1
```

3.4 **HAMILTONIAN——定义哈密顿量

这一部分定义计算使用的电子哈密顿量。程序内部总是使用由不同的大、小分量基组展开的四分量算符工作。而在变换到正交基组中时，为了获得诸如无自旋修正的 Dirac 方程或 Lévy-Leblond 方程的矩阵展开[3]，可以把大、小分量函数和/或不同对称性函数组合到一起。默认的哈密顿量是未修正的 Dirac-Coulomb 哈密顿量。此外还可以引入附加的算符来对哈密顿量进行修正，例如描述外场。在分子点群对称性和时间反演对称性下，这一部分定义的任何附加算符必须是全对称的。后者还需要避免引入外部磁场。

基本选项

. LEVY-LEBLOND

使用 Lévy-Leblonds 哈密顿量[4]。需要在指定附加单电子算符前使用这一选项，因为它重新定义计算中使用的矩阵。

. SPINFREE

使用 Dyll 的无自旋哈密顿量[5]，以获得没有自旋-轨道耦合的结果。注意这一选项不要与时间反对称的（磁）算符一起用于响应计算，因为它排除了重要的贡献。

. LVCORR

通过小分量原子电荷的经典排斥，模拟原子间 SS 积分的贡献[6]。这是目前对完全 Dirac-Coulomb 哈密顿量最经济、最准确的近似，可用于光谱常数和价电子特性的计算；对于芯电子的特性，建议进行测试。

. DFT

执行 Kohn-Sham 计算，并指定一种 DFT 泛函。

预定义的泛函：LDA, BLYP, B3LYP

组合泛函：GGAKEY fun1=w1 fun2=w2 ...

泛函可以从一组预定义的交换和关联泛函的组合中选择，也可以通过指定一系列泛函及其权重来合成。

泛函列表：

1. 交换泛函

✧ LDA 交换[7]：SLATER

✧ Becke 1988 [8]：BECKE

✧ Becke [8]建议的 LDA 校正项：B88CORR

✧ Perdew-Wang 1986 [9]：PW86C

2. 关联泛函

✧ Vosko-Wilk-Nusair LDA 关联能 (VWN5) [10]：VWN

✧ Lee-Yang-Parr 泛函[11]：LYP

✧ Perdew 1986 [12]：P86C

高级选项

. OPERAT

哈密顿量中的附加单电子算符说明。这个算符在分子点群对称性和时间反演对称性中必须是全对称的。算符的场强由 COMFACTOR 定义。这个关键字可以重复使用从而加入多个算符。详细信息见 6.1 节。

. INTFLG

指定包含哪些双电子积分。所有其它模块使用这个 . INTFLG 作为默认值。

参量：整型 ILL, ISL, ISS

IXX = 1 (打开) / 0 (关闭) (XX = LL, SL 或 SS)

- 默认: ILL = ISL = ISS = 1
- .LVNEW** 修改.LVCORR 选项, 用来通过 Mulliken 分析而不是原来的表格查询方法获得原子小分量电荷 (表格查询的问题是, 如果指定的基组没有给出正确的小电子电荷, 由于芯区域的函数不够, 那么分子中的静电将是错误的)。
- .ZORA** 在 Hartree-Fock 程序中使用零级正则近似[13, 14, 3]。这一方法几乎没有有什么计算优势, 主要用于方法的比较。注意.SPINFREE 和.ZORA 的组合给出无自旋的形式, 这不同于通常的无自旋 ZORA 算法。在.ZORA 之后, 用自由格式定义两个整数。第一个表示密度是对二分量(0: ZORA)还是对四分量(1: ZORA4)度量进行归一化。第二个表示轨道能量不变(0: 标准 ZORA)还是可调(1: 换算的 ZORA)。
- .HFXFAC** Fock 矩阵构造中的交换权重。
参量: 实型 HFXFAC
默认: HFXFAC = 1.000
- 程序员选项**
- .PRINT** 打印级别。
参量: 整型 IPRHAM
默认: IPRHAM = 0
- .ONESYS** 忽略双粒子间的相互作用。
默认: ONESYS = .FALSE.
- .FREEPJ** 从分子轨道空间中投影掉所有的自由正电子的解。
默认: .FALSE.
- .URKBAL** 非限制性动能平衡。
默认: 限制性动能平衡。它强制从自由粒子正电子谱中删除所有的非物理解。这会导致电子与正电子解的比例为 1:1。提前进行投影对于线性依赖关系非常敏感, 因此更适用于加上球变换的大、小分量。
- .VEXTPJ** 从分子轨道空间中投影掉所有外场的正电子解。
默认: 不投影。
- .NOSMLV** 删除 SS 核吸引积分。这去掉单电子自旋-轨道相互作用和 Darwin 相互作用的贡献。
默认: 不删除 SS 核吸引积分
- .SMLV1C** 忽略多中心 SS-区的势, 即多中心小-小核吸引积分和多中心 SSSS 双电子积分将被忽略。
默认: 不删除 SS 核吸引积分
- .ONECAP** 进一步考虑 SMLV1C 模型。只有双电子积分对 LS 和 SS 区的单中心贡献, 以及核吸引积分的 SS-区的单中心贡献被明确计算。通过从基于 Mulliken 布居分析的小分量电荷计算经典排斥, 可以得到被该法所忽略的项的静电影响。因此当计算分子梯度时, 我们仅需要计算积分的 LL 区的导数。
默认: ONECAP = .FALSE.
- .ONECNV** 使用 ONECAP 所给出的单中心模型, 直到收敛到特定的阈值。在此之后, 使用整套双电子积分。
只要选择了收敛标准(EVCCNV, ERGCNV, FCKCNV), 就使用这一阈值。
参量: 实型 ONECNV

3.4.1 *DFT——DFT 指令

DFT 模块在参考文献[15, 16]中描述。它使用标准的非相对论泛函（见上），因为各种研究表明[17, 18, 19]，相对论关于交换-关联泛函的校正，对于光谱常数的影响可以忽略；但对于芯电子特性的影响还需要进一步研究。数值积分方案使用 Becke 把分子体积分割为原子体积的方法，其中积分在球坐标下进行。径向积分用 Lindh 等[21]建议的方案执行，角向积分由一套高精度的 Lebedev 格点进行处理。注意，Becke 分割方案一般使用 Slater-Bragg 半径，用于调整原子尺寸。对于较重的元素，由于有较多的可变氧化态，这可能会导致错误。如果用户调用 .ATSIZE 关键字，DIRAC 将尝试从可用的密度导出原子的相对尺寸，只要程序能找到因子。

在*DFT 小节，用户可以修改数值积分的参数。不要改变默认积分参数，除非知道你要做什么。

- .PRINT 打印级别。
 参量：整型 IPRDFT
 默认：IPRDFT = 2
- .RADINT 指定径向积分中的最大误差。
 参量：实型 RADERR
 默认：RADERR = 1.0D-13
- .ANGINT 指定 Lebedev 角向格点的精度。球谐角向积分精确到 L 值 NSCHEME。默认情况下，格点会被修整。
 参量：整型 NSCHEME
 默认：NSCHEME = 41
- .NOPRUN 关闭角向格点的修整。
- .ANGMIN 指定修整后 Lebedev 角向格点的最小精度。接近于核的位置，角向格点的精度会低于 NSCHEME，但不会小于 LEBMIN。注意，如果给出大于或等于 NSCHEME 的 LEBMIN 值，等价于关闭修整。
 参量：整型 LEBMIN
 默认：LEBMIN = 9
- .ATSIZE 对原子尺寸比率产生新的估计，用于 Becke 分割方案。一对原子 A 和 B 的相对尺寸从其对 A-B 连线上大分量密度的贡献来计算。
- .ERRELS 指定在积分电子数中可接受的误差。若出现更大的误差，程序将停止。在 SCF 的第一次迭代中，如果从尝试矢量开始，则会关闭该测试，因为如果是在其它结构产生的矢量，就不会是标准正交的。
 参量：实型 DFTELS
 默认：DFTELS = 1.0D-3

高级选项

- .INTCHK 通过数值计算重叠矩阵和分析误差来测试格点。此外还可以打印误差矩阵。
 参量：整型 INTCHK
 输出：无（INTCHK = 0），误差分析（INTCHK = 1），以及附加的打印误差矩阵（INTCHK = 2）。
 默认：INTCHK = 0
- .FROMVX 在 Kohn-Sham 计算中，交换-关联势一般不明确计算，而是在程序内部作为

交换-关联势的矩阵元计算，参见[15]，因为这会使计算效率更高。.FROMVX 会直接产生交换-关联势。

3.4.2 示例输入

在 Z 方向上为非相对论 Lévy-Leblond 哈密顿量添加偶极微扰，强度为 0.01 a.u.。

```
**HAMILTONIAN
. LEVY-LEBLOND
. OPERATOR
  ZDIPLN
  COMFACTOR
  0.01
```

3.5 **WAVE FUNCTIONS——得到波函

这一部分定义激活哪个（些）波函模块。默认不激活任何模块。要激活的那些模块必须也在**DIRAC 下的.WAVE F 中指定，否则不读取这些输入。

注意，如果请求一个以上的模块，以下出现的顺序定义了不同模块调用的顺序。

基本选项

- . DHF 激活 Hartree-Fock/Kohn-Sham 模块。DHF 模块的详细说明见**DHFCAL 小节。如果在**HAMILTONIAN 下指定了.DFT，那么将执行 Kohn-Sham 计算，否则执行 Hartree-Fock 计算。
- . RESOLVE 求解开壳层态：运行小的 CI 计算，获得组态平均开壳层 Dirac-Hartree-Fock 计算中出现的某个态能量。见下面的**RESOLVE。
- . MP2 激活二阶 Møller-Plesset 模块。MP2 模块的详细说明在*MP2CAL 小节给出。
- . MVO 计算修正的虚轨道。见下面的*MVOCAL。
- . RELCCSD 激活耦合簇模块（以及 MOLTRA 模块获得四指数转换积分）。RELCCSD 模块的输入说明在名称列表 RELCCSD 中给出（参见 3.9 章）。默认包含轨道能量在-10 至+20 a.u.之间的分子轨道，这可以在**MOLTRA 输入部分修改。
- . DIRRCI 激活 MOLFDIR CI 模块（以及 MOLTRA 模块获得四指数转换积分）。DIRRCI 模块的输入说明在名称列表 DIRRCI 和 GOSCI 中给出（参见 3.10 和 3.11 章）。默认包含轨道能量在-10 至+20 a.u.之间的分子轨道，这可以在**MOLTRA 输入部分修改。
- . LUCITA 激活 LUCITA CI 模块（以及 MOLTRA 模块获得四指数转换积分）。LUCITA 模块的输入说明在 3.12 章给出。默认包含轨道能量在-10 至+20 a.u.之间的分子轨道，这可以在**MOLTRA 输入部分修改。

3.5.1 *DHFCAL——Hartree-Fock/Kohn-Sham 计算

这一部分给出了闭壳层和组态平均 Hartree-Fock 计算的指令。Kohn-Sham 通过在 **HAMILTONIAN 下调用 .DFT 关键字执行，细节参见 3.4 章。注意组态平均 Kohn-Sham 计算还不能很好地定义。

1. 占据

. CLOSED SHELL 对每一个费米不可约表示，给出闭壳层电子的数量。

参量：整型 (NELEC(I), I=1, NFSYM)

默认：NELEC(1) = NELEC(2) = 0

. OPEN SHELL 开壳层的说明。定义开壳层数量。

参量：整型 NOPEN

对每一个开壳层 J，电子数量和活性旋量的数量按下面格式给出：NAELEC/ASC1, ASC2，其中 NAELEC 是活性电子数量，ASCx 是共表示 x 中的活性旋量数量。例如：

```
. OPEN SHELL
```

```
1  
5/0, 6
```

这给出了在共表示 2 中，6 个旋量（相当于 3 个 Kramers 对）内有 5 个电子的开壳层。因此分数占据是 5/6。

闭壳层的输入非常简单。对于没有转置对称性的对称群，只有一个费米不可约表示，因此仅需要指定电子的数量。例如，水分子(C_{2v} 点群)的最小波函数输入为：

```
**WAVEFUNCTION  
. DHF  
*DHFCAL  
. CLOSED SHELL  
10
```

对于有转置对称性的点群，需要指定电子在两个费米不可约共表示 (co-representation) 中的分布。例如对 Ne 原子，最小输入为：

```
**WAVEFUNCTION  
. DHF  
*DHFCAL  
. CLOSED SHELL  
4 6
```

DIRAC 中的开壳层模型建立在组态平均[24]的基础上。最简单的情况是两个旋量（一个 Kramers 对）中的一个电子。对于这个特例，组态平均计算给出了与通常的限制性开壳层 Hartree-Fock 方法相同的结果。对于其它情况，计算给出多个态的平均能量。两个旋量中有一个电子的例子是硼原子：

```
**WAVEFUNCTION  
. DHF  
*DHFCAL  
. CLOSED SHELL
```

```

4 0
.OPEN SHELL
1
1/0, 2

```

这样解释输入：我们有4个非活性电子（在1s和2s原子轨道中）和一个开壳层，其中一个活性电子位于共表示2的两个旋量中（ $2p_{1/2}$ 和 $2\tilde{p}_{1/2}$ ）。这个计算给出了硼 $P_{1/2}$ 的能量，还可以做这样的计算：

```

**WAVEFUNCTION
.DHF
*DHFAL
.CLOSED SHELL
4 0
.OPEN SHELL
1
1/0, 6

```

这表示把一个活性电子分布在2p轨道的6个分量上，因此给出六个组态的平均能量：

$$\frac{2 \cdot E_{P_{1/2}} + 4 \cdot E_{P_{3/2}}}{6} \quad (3.1)$$

还可以指定一个以上的开壳层。让我们考察Pt原子两种不同的开壳层输入（ $6s^1 5d^9$ ）：

(a)

```

.OPEN SHELL
2
9/10, 0
1/2, 0

```

(b)

```

.OPEN SHELL
1
10/12, 0

```

在第一个例子中，得到 $\binom{10}{9} \cdot \binom{2}{1} = 10 \cdot 2 = 20$ 个态的平均能量。所有的20

个态都是 $s^1 d^9$ 。在第二个输入中，我们把同样的10个电子分配到全部的12

个旋量中，这将给出总共 $\binom{12}{10} = 66$ 个态，也就是1个 $s^0 d^{10}$ ，20个 $s^1 d^9$ ，和

45个 $s^2 d^8$ 态的混合。

注意闭壳层和开壳层的顺序，假定如图3.1所示，也就是最低的分壳轨道是双占据的，之后的轨道由电子占据1号、2号等的开壳层，其它的顺序可以用关键字.REORDER MO'S和重叠部分(OVLSEL)得到（参见3.5.1节）。

如果要得到组态平均中出现的各个态的能量，指定.RESOLVE（参见 3.5 和

3.5.4 节)。要得到组态平均中出现的某几个态的能量，可以使用 GOSCIIP 模块(参见 3.11)，RIRRCI 模块(参见 3.10)，或 LUCITA 模块(参见 3.12)。在 SCF 循环中，程序允许改变占据。

.AUTOCC

默认：不激活。但是，如果 .CLOSED SHELL ELECTRONS 和 .OPEN SHELL ELECTRONS 均未定义，那么程序仍将自动初始化占据。

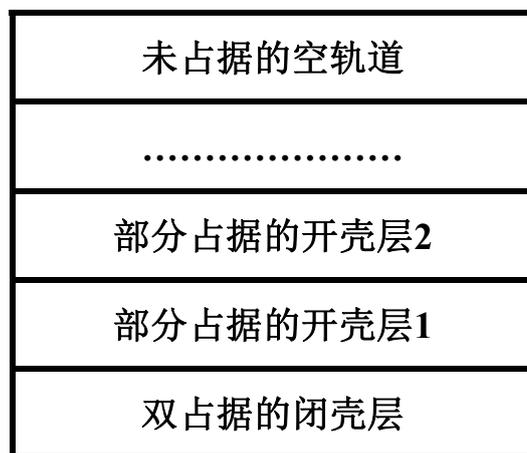


图3.1 闭壳层和开壳层的顺序

2. 测试函数

SCF 计算 (HF 或 DFT) 可以有三种不同的方法进行初始化：

- 使用前一计算得到的分子轨道因子。
- 使用由对角化单电子 Fock 矩阵得到的因子：裸核方法。
- 使用前一计算得到的双电子 Fock 矩阵。这可看作从收敛的 DHF 势开始。

如果出现了 DFCOEF 文件，那么默认从分子轨道因子开始。否则使用校正的裸核方法。在这三种方法中，线性依赖在第 0 次迭代中就删除了。

.NOBNCR

关闭裸核校正。这一校正默认开启，通过用核吸引型的积分估算双电子排斥来改善核的屏蔽。

$$\left\langle X_A \left| \sum_C \frac{-Z_C \cdot \sum_j a_j e^{(-a_j^C r_C^2)}}{r_C} \right| X_B \right\rangle, X=L, S \quad (3.2)$$

表达式中的因子 a_j 和指数 a_j^C 按照 Slater 的规则进行选择，获得用于初始猜测的近似原子电子密度。例如对于含一个重元素的体系，若没有该校正（也就是用裸核哈密顿量），初始猜测中所有的电子将在该重元素处截止。

默认：NOBNCR = .FALSE.

.TRIVEC

从矢量文件开始 SCF 迭代。

.TRIFCK

从前一次计算的双电子 Fock 矩阵（存储于文件 DFFCK2 中）开始 SCF 迭代。

3. 收敛标准

可以选择不同的收敛标准：

- DIIS 误差矢量 $\mathbf{e}=[\mathbf{F}, \mathbf{D}]$ （在 MO 基中）的模。这对应于电子梯度的模，是推荐的收敛标

准。当只对能量感兴趣时，通常 $EVCCNV = 1.0D-5$ 足够了。对于特性以及有关的方法，应当收敛到 $EVCCNV = 1.0D-9$ 。大的正电子本征值会丢失精度，导致收敛问题。还要记住，过于严格的屏蔽阈值将阻碍收敛，如果修改了 $EVCCNV$ 或其它两个收敛标准中的一个，那么必须修改 $*TWOINT$ 下的 **SCREEN**。

- 连续两次迭代的总能量差值。
- 连续两次迭代中，总 Fock 矩阵的最大绝对差值。

总能量的变化近似为误差矢量或 Fock 矩阵最大变化的平方。默认对电子梯度收敛，阈值为 $SCFCNV = 1.0D-6$ 。或者达到最大迭代数后停止。有时候，指定的收敛标准对于给定的基组和/或其它输入参数过于严格。遇到这种情况就需要决定是否进行后-HF 步骤（像关联能计算）。程序通过考虑第二收敛标准来决定，它给出所允许的收敛。这个值默认与第一收敛标准或所要的收敛标准相同，但可以设得更低，以保证当收敛略微高于阈值时不会意外中断。

- .MAXITR** SCF 迭代的最大数量。
参量：整型 MAXITR
默认：MAXITR = 50
- .EVCCNV** 误差矢量（电子梯度）的收敛。
参量：实型 SCFCNV (1) SCFCNV (2)
- .ERGCNV** 总能量的收敛阈值。
参量：实型 SCFCNV (1) SCFCNV (2)
- .FCKCNV** Fock 矩阵最大绝对变化的收敛。
参量：实型 SCFCNV (1) SCFCNV (2)

注意，第二个（允许的）阈值 $SCFCNV (2)$ 可以忽略。此时它等于第一个（所要的）阈值 $SCFCNV (1)$ 。

4. 收敛加速

保证 SCF 迭代数最小是强制的。这可以用收敛加速方案来解决。

- **阻尼**。最简单的方案是 Fock 矩阵的阻尼，这可以消除振荡。在第 $n+1$ 次迭代中，Fock 矩阵对角化为：

$$F' = (1 - c)F_{n+1} + cF_n; \quad c: \text{阻尼因子} \quad (3.3)$$

- **DIIS**（迭代子空间的直接反演[25, 26, 27]）。可以看作是包括 Fock 矩阵多次迭代的广义阻尼。阻尼因子可以通过求解包含 **B** 矩阵（由误差矢量（近似梯度）构成）的简单矩阵方程得到。**B** 矩阵的线性依赖列被删除。

在 DIRAC 中，DIIS 好于阻尼。

- .DIISTH** 基于误差矢量的最大元素，为开始 DIIS 改变默认阈值。
参量：实型 DIISTH——开始 DIIS 的收敛阈值
默认：一个非常大的数
- .MXDIIS** 在 DIIS 模块中，**B** 矩阵的最大维数。
参量：实型 MXDIIS——**B** 矩阵的最大维数
默认：MXDIIS = 10
- .NODIIS** 不进行 DIIS。
默认：DIISMO 对闭壳层的计算激活，DIISAO 对组态平均的计算激活。
- .DAMPFC** 改变默认的阻尼因子。
参量：实型 DAMPFC——阻尼因子

默认: DAMPFC = 0.25

.NODAMP 不执行 Fock 矩阵的阻尼。

默认: 激活阻尼, 但是 DIIS 优先。一旦 B 矩阵所有的列被线性依赖删除, 则激活阻尼。

.DIISMO 在正交基 (MO) 中使用上面的误差矢量 ($\mathbf{e}=[\mathbf{F},\mathbf{D}]$) 激活 DIIS。仅用于闭壳层计算。

.DIISAO 激活使用 AO 基的 Dalton 型 DIIS。可用于开壳层和闭壳层的计算。误差矢量为 $\mathbf{e}=\mathbf{f}\cdot\mathbf{f}^\dagger$, 其中的 \mathbf{f} 为:

$$\mathbf{f} = \mathbf{C}^\dagger \cdot \mathbf{S}_{AO} \cdot \left(\mathbf{D}_{AO}^C \cdot \mathbf{F}_{AO}^D + \sum_{O \in O} f_O \cdot \mathbf{D}_{AO}^O \cdot (\mathbf{F}_{AO}^D + (a_O - 1) \mathbf{Q}_{AO}^{V,O}) \right) \mathbf{C}$$

5. 态选择

收敛可以通过选择与前一迭代有最大重叠的矢量来改善。这种方法还可用于到一些激发态的收敛。

如果使用了动态重叠的选择, 前一迭代的矢量集将用做收敛标准。对于第一个迭代, 使用重新开始矢量或者裸核近似产生的矢量 (不推荐)。

如果给出了 NODYNSEL, 则必须使用重新开始矢量和裸核矢量之一, 也就是说, 重叠选择的矢量在每一次迭代中并不升级。请注意, 基于裸核近似矢量的重叠选择是不推荐的。

重叠选择结合 .REORDER MO'S 关键字非常有用。这将重新安排重新开始因子中的矢量。例如, 首先进行硼原子的开壳层计算, 这会给出 $\mathbf{P}_{1/2}$ 态。但是如果我们从 $\mathbf{P}_{1/2}$ 因子重新开始, 交换 $\mathbf{p}_{1/2}$ 和 $\mathbf{p}_{3/2}$ 轨道, 并使用重叠选择, 就可以收敛到 $\mathbf{P}_{3/2}$ 态。还有一个关键字用于重新排列收敛的 DHF 轨道。这对于重新排布轨道, 用于四指数变换和随后的关联能计算 (CCSD, CI 等) 的都非常有用 (参见关键字 .POST DHF REORDER MO'S)。

.OVLSEL 激活动态重叠选择。

默认: 无重叠选择

.NODYNSEL 没有重叠选择矢量的动态升级。

默认: 动态升级

.REORDER MO'S 交换初始分子轨道。从 DFCOEF 读取开始的轨道, 并重新排序。

参量: 字符型(A72)。

对每个费米共表示, 给出轨道新的次序。

例如:

```
.REORDER MO' s
```

```
1. . 8, 10, 9
```

.POST DHF REORDER MO'S

交换收敛的分子轨道。在退出 DHF 子程序之前, 从 DFCOEF 读取轨道并重新排序。

参量: 字符型(A72)。对每个费米不可约表示, 给出新的轨道次序。

例如:

```
.POST DHF REORDER MO' s
```

```
1. . 8, 10, 9
```

6. 迭代加速

通过减少每次迭代中处理的积分数量, 可以极大地缩短总运行时间:

- **积分的屏蔽:** 通过设定阈值, 排除阈值以下的积分[2]。LL-积分的阈值在基文件中设定, 但是这个阈值可以通过阈值因子 (在*INTEGRALS 中设定) 对 SL-和 SS-积分进行调整:
LL-积分的阈值: THRS
SL-积分的阈值: THRS*THRFA(1)
SS-积分的阈值: THRS*THRFA(2)
这些阈值可以在积分部分设定。
- **密度的屏蔽:** 直接模式中, 再通过屏蔽密度矩阵[2], 进一步减少计算量。如果使用了微分密度会更加有效。微分密度为:

$$\Delta D = D_{n+1} - \alpha \cdot D_n \quad (3.4)$$

α 的默认值为:

$$\alpha = \frac{D_{n+1} \cdot D_n}{D_n \cdot D_n} \quad (3.5)$$

对应 Gram-Schmidt 正交化。当 SCF 收敛, α 趋向于 1, 但是 α 也可以用 FIXDIF 直接设定为 1。

- **积分的忽略:** 在 DHF 迭代的高级阶段, 即使添加了 SL-和 SS-积分, 处理积分的数量也可以大大减少, 因为它由迭代数量或能量收敛决定, 而且后者比前者优先。

- .NODSCF 不用微分密度矩阵进行 SCF 迭代。
默认: 在直接 SCF 中使用微分密度矩阵。
- .FIXDIF 设定上面方程 (3.4) 中的 α 等于 1。
默认: FIXDIF = .FALSE.
- .CNVINT 在把 SL-和 SS-积分添加到 SCF 迭代之前, 设定收敛的阈值。
参量: 实型 CNVINT(1) (SL), CNVINT(2) (SS)
默认: 非常大的数
- .ITRINT 在把 SL-和 SS-积分添加到 SCF 迭代之前, 设定迭代的数量。
参量: 整型 ITRINT(1) (SL), ITRINT(2) (SS)
默认: ITRINT(1) = ITRINT(2) = 1
- .INTFLG 指定包含哪些双电子积分。
参量: 整型 ILL, ISL, ISS
IXX = 1 (打开) / 0 (关闭) (XX = LL, SL 或 SS)
默认: 来自**GENERAL 的 INTFLG

7. 输出控制

- .PRINT 一般打印级别。
参量: 整型 IPRSCF
默认: IPRSCF = 0
- .EIGPRI 控制电子和正电子解的打印。
参量: 整型 IPREIG(1) (电子), IPREIG(2) (正电子)
默认: 打印电子本征值
- .PHCOEF (DFCOEF 中) 因子的相位调整: 使给定轨道的最大元素为正实型。

8. 排除/冻结轨道

在电子结构的研究中, 可能想排除或冻结某些轨道。此选项对于收敛也很有用, 特别是

对于激发态。一个简单的例子是铯原子。基态 $^2P_{1/2}$ 的电子组态为 $[\text{Xe}]4f^{14}5d^{10}6s^26p^{1/2}$ 。第一激发态 $^2P_{3/2}$ 的电子组态 $[\text{Xe}]4f^{14}5d^{10}6s^26p^{1/2}$ 可以很容易地得到：首先计算基态，接下来，在随后的计算中排除 $6p^{1/2}$ 。在最后的计算中，激发态用重叠选择来放松。冻芯轨道的使用在测试 33. frozen 中有说明：当冻结氧的轨道 1s 和 2s 进行水分子的优化时，发现键角为 96.242 度，与 $s-p$ 杂化禁阻所预期的 90 度[28]相反。

轨道的排除通过把所选的轨道从矩阵到正交基的变换中投影出去来实现。所选的轨道即可以表示为整个分子的基，也可以在所选片段的基组中。对于后一种情况，同一组片段轨道可用于也使用原子片段的不同分子结构。有人可能还想进行几何优化，但只能用数值梯度。对于冻结轨道，所选的轨道首先从到正交基的变换中排除出去，但接下来再次引入到反变换步骤中。它们会出现在输出中，具有零轨道本征值。注意当冻结轨道时，也必须同时指定要排除的轨道。冻结轨道必须是排除轨道的子集。

片段使用 DIRAC 基文件中出现的，与对称性无关的原子列表定义：考虑全对称性 C_{2v} 的水分子。有两个片段：氧原子和余下的一半 H2。但是当不使用对称性时，会有三个片段：氧原子和两个氢原子。目前还没有对不同片段的片段正交归一化，因此实际上只能使用来自于一个片段的轨道。

- . PROJEC 通过把所选的轨道从矩阵到正交基的变换中投影出去来排除轨道。
 参量：整型 NPRJREF——片段数。接下来对每个片段读取名为 PRJFIL 的因子文件，之后是对每个费米不可约表示所选轨道的轨道串（见 3.13.1 节）。
 DO J = 1, NPRJREF
 READ(LUCMD, '(A6)') PRJFIL(J)
 READ(LUCMD, '(A72)') (VCPROJ(I, J), I=1, NFSYM)
 ENDDO
- . PRJTHR 当排除轨道时，所接受的最小模。
 参量：实型 PRJTHR——投影阈值
 默认：PRJTHR = 1.0D-10
- . OWNBAS 片段基中给出的排除轨道/冻结轨道。注意，片段的列表假定接在 DIRAC 基文件中对称性无关核的列表之后。
- . FROZEN 冻结轨道（必须只能与 PROJEC 关键字一起使用）。对于每个费米不可约表示，给出冻结轨道的轨道串（见 3.13.1 节）。
 参量：字符型 (A72) (VCFROZ(I), I=1, NFSYM)

3.5.2 示例输入

```
*DHFCAL
.PRINT
1
.NELECT
10 0
.MAXITR
30
.EVCCNV
1.0D-6
.MXDIIS
```

```

10
.CNVINT
1.0D-2 1.0D-4
.EIGPRI
1 0

```

3.5.3 *MP2CAL——MP2 计算

这一部分给出用于直接积分闭壳层 MP2 计算的指令。

基本选项

- .OCCUPIED** 活性占据电子解。对每个费米不可约表示给出活性轨道的轨道串（参见 3.13.1 节）。
 参量：字符型(A72) (MP2_INDSTR(1, I), I=1, NFSYM)
 默认：占据电子解
- .VIRTUAL** 活性虚轨道解。对每个费米不可约表示给出活性轨道的轨道串（参见 3.13.1 节）。
 参量：字符型(A72) (MP2_INDSTR(2, I), I=1, NFSYM)
 默认：所有未占据的虚轨道解

高级选项

- .INTFLG** 在直接 MP2 计算中指定包含哪些双电子积分。
 参量：整型 ILL, ISL, ISS
 IXX = 1 (打开) / 0 (关闭) (XX = LL, SL 或 SS)
 默认：来自于**GENERAL 的 INTFLG
- .SCREEN** 四指数变换中的屏蔽阈值。负数使屏蔽无效。
 参量：实型 SCRMP2
 默认：1.0D-14

程序员选项

- .PRINT** 打印级别。
 参量：整型 IPRMP2
 默认：IPRMP2 = 0
- .VIRTHR** 用来忽略高位虚轨道的阈值。忽略能量位于 DMP2_VIRTHR 以上的所有轨道。保留该选项用于向下兼容的目的，建议使用轨道串输入中更一般的能量选项。
 参量：实型 DMP2_VIRTHR
 默认：无阈值
- .IJTSK** 在每一分批计算中，I（或 J）轨道（占据轨道）的最大数量。通常分一批处理就足够了，但是通过减少批处理量的大小，才有可能进行更大的计算。批处理量减半，近似相当于减少到 1/4 的内存需求，并且增加 4 倍的 CPU 时间。
 参量：整数 IJTSK

默认：所有活性占据电子解
. SCLMEM 标量的未转换积分可以使用的内存。更大的批处理量将写在硬盘上。
 参量：整型 MAXSCL
 默认：由程序设定，通常足够

Au₂ 价电子+亚价电子计算的输入示例。19 个活性占据的 **Kramers** 对分为两组，计算分三批完成(对 I+J 是 1..10+1..10, 11..19+1..10, 和 11..19+11..19)。虚空间中所有能量在 100 a.u. 以上的轨道都被忽略。计算中没有包含 **SS** 积分。注意活性空间中的轨道未必连续。

```
. NELECT
80 78
.....
*MP2CAL
.OCCUP
24, 32.. 40
24, 32.. 39
.VIRTUAL
all
all
.VIRTHR
100.0
.INTFLG
1 1 0
.IJTSK
10
```

3.5.4 *RESOLVE——求解开壳层态

组态平均计算中的各个电子态用 GOSCIP 模块在开壳层旋量中通过完全 CI 来求解。注意，即使是小的 CI 计算，原则上也需要对初始四指数变换产生所有的 AO 积分。在四指数变换中调用积分屏蔽（由 **. SCREEN** 控制），可能的话至少忽略 **SS** 积分（使用 **. INTFLG**），可以极大地减少计算量。

. PRINT 打印级别。
. INTFLG 指定包含哪些双电子积分。
 参量：整型 ILL, ISL, ISS
 IXX = 1（打开）/0（关闭）（XX = LL, SL 或 SS）
 默认：来自于**GENERAL 的 INTFLG
. SCREEN 四指数变换中的屏蔽阈值。负数使屏蔽无效。
 参量：实型 SCRRES
 默认：1.0D-14
. SCHEME 积分变换算法。
 参量：整型 ISTRES
 默认：ISTRES = 4

3.5.5 *MVOCAL——修改的虚轨道

对于相关能计算，虚的正则 Hartree-Fock 轨道不是最好的选择：虚空间会被截断；由于考虑 N+1 电子的体系，最低的轨道趋向于弥散。可以使用虚轨道中的转动自由度，以获得更适合于相关能计算的轨道。DIRAC 中的默认选项是通过删除了某些电子的体系构造 Fock 算符，产生修正的虚轨道，这最早由 Bauschlicher 建议[29]。用.MVOVEC 指定的轨道串表示在构造 Hartree-Fock 势时删除哪些占据轨道。

注意，产生的修正虚轨道不再是正则的，也不能用于 MP2 计算；它们可以用在 DIRRCI 和 RELCCSD 模块中。

- .PRINT 打印级别。
 参量：整型 IPRMVO
 默认：IPRMVO = 0
- .VECMVO 当构造阳离子的 Fock 算符时，读取要从 Hartree-Fock 势删除的轨道串（参见 3.13.1 节）。
 READ(LUCMD, '(A72)') (VECMVO(I), I=1, NFSYM)
- .INTFLG 在构造修正的虚轨道时，指定包含哪些双电子积分。
 参量：整型 ILL, ISL, ISS
 IXX = 1 (打开) / 0 (关闭) (XX = LL, SL 或 SS)
 默认：来自于**GENERAL 的 INTFLG

3.6 **ANALYZE——分析波函

这一部分对最后的波函进行分析。

- 基本选项** 默认不激活分析模块。
- .DENSIT 把密度写到 Gaussian 立方格式的文件中。
- .MULPOP 进行 Mulliken 布居分析。
- .PRIVEC 打印矢量。
- .PROJEC 把矢量投影到基的另一矢量集中。
- .RH01 沿中心的连线转存密度。

3.6.1 *PRIVEC——矢量打印

可以打印从 DFCOEF 读入的矢量。

- 基本选项**
- .VECPRI 对每个费米不可约表示，给出需要打印轨道的轨道串（参见 3.13.1 节）。
 参量：字符型 (A72) (VECPRI(I), I=1, NFSYM)

默认：占据的电子解

高级选项

- .PRICMP** 单独控制大、小分量的打印。
参量：整型 IPRCMP(1)（大分量），IPRCMP(2)（小分量）
默认：只打印大分量
- .AOLAB** 打印 AO 基中的矢量。
默认：打印 SO 基中的矢量

程序员选项

- .PRINT** 打印级别。

3.6.2 *MULPOP——Mulliken 布居分析

在 AO 基中进行 Mulliken 布居分析。分析基于“标记”的概念。每个基函数由函数的类型和中心（以及玻色不可约表示，当在在默认的 SO 基中时）标记，标记在输出文件中给出。一套初始标记可以按照用户指定的群标记收集起来进行分组。

- .VECPop** 对每个费米不可约表示，给出需要分析轨道的轨道串（参见 3.13.1 节）。
参量：字符型 (A72) (VECPop(I), I=1, NFSYM)
默认：占据的电子解

高级选项

- .NETPOP** 除了总布居（默认）之外，还给出 Mulliken 净布居/重叠布居。
默认：不给出净布居/重叠布居
- .LABDEF** 定义用于 Mulliken 布居分析的标记。
参量：整型 NPOPLAB——定义标记的数量。接下来对每一费米不可约表示读取标记的名称和轨道串（关于标记，参见 3.13.1 节）。
DO I = 1, NPOPLAB
 READ(LUCMD, '(A12, A60)') POPLAB(I), LABPOP(I)
ENDDO
- .AOLAB** AO 基标记的基本定义。
默认：SO 基标记的基本定义
- .INDSML** 使用专用的小分量标记。
默认：收集属于给定中心和不可约表示的所有小分量函数，不考虑函数类型

程序员选项

- .PRINT** 打印级别。
参量：整型 IPRPOP
默认：IPRPOP = 0

3.6.3 *PROJECTION——投影分析

当前的解可以投影到另一套由基产生的因子中。例如，可以把分子解投影到原子解上，以评价原子的贡献。同时读取片断的因子，并考虑它们之间的重叠[30]。因此投影分析可以看作是基原子（片断）轨道的 Mulliken 分析；因此它对基组不敏感。

片断一般在完全分子基中计算。这可以通过把基组输入部分的分子中剩余原子的电荷调为零，并在单文件中修改菜占据数得到。但是也可以使用 OWNBAS 关键字，在自己的基（全部分子基的子集）中计算片断。这样做的优点是计算更快，对于原子片断保持原子对称性。为了避免在片断中使用对称性污染的原子中心，有时候可以把分子因子转存到 C_i 对称性中，使用的关键字为 .ACMOUT (3.2 节)，并且不用对称性进行分析。当使用关键字 .OWNBAS 时，接下来每个原子类型需要在自己的基中计算一次，以便做完整的分析。

基本选项

- .VECPRJ 对每个费米不可约表示，给出需要分析轨道的轨道串（参见 3.13.1 节）。
参量：字符型 (A72) (VECPRJ(I), I=1, NFSYM)
默认：占据的电子解
- .VECREF 先给出要投影到多少个片段中，接着对每一片断给出 MO 因子的文件名，并对每一费米不可约表示给出参考轨道的轨道串（参见 3.13.1 节）。
参量：整型 NREFS——片断数量
参量：字符型 (A72) (REFFIL(IFRAGMENT))
参量：字符型 (A72) (VECREF(I, IFRAGMENT), I=1, NFSYM)
默认：无。所有的行必须给出。
- .OWNBAS 在它们自己的基中计算片断。注意！这个关键字在使用时必须小心，例如在假定片断列表与对称无关中心的片断列表一致的情况下。

高级选项

- .PROTHR 设定要打印的投影因子绝对值的阈值。
参量：实型 PROTHR
默认：PROTHR = 1.0D-2
- .WGPOP 按照贡献的权重分割重叠密度。
默认：不激活

程序员选项

- .PRINT 打印级别。
参量：整型 IPRPRJ
默认：IPRPRJ = 0

3.6.4 *DENSIT ——产生用于显示的密度

密度可以转存到 Gaussian 立方格式的格式化文件中，用于以后的显示，如使用 MOLEKEL 软件包（见 <http://www.cscs.ch/molekel/>）。

Gaussian 立方格式存储了体数据以及原子位置。体数据由三个轴和一个原点定义。具体

格式为:

行 1-2 两个注释行

行 3 NATOM XO YO ZO

原子数目, 之后是体数据的原点。

行 4-6 接下来的三行给出了沿着每个轴(x, y, z)上的体元数, 后面是轴矢量。如果某个维度的体元数符号为正, 则使用的单位为埃, 负数使用原子单位。

NATOM 个行 对每个原子是原子序号和坐标的一行。

余下的文件包含体数据, 在这种情况下是密度。

大分量和小分量密度分别写到文件 rhoL.cube 和 rhoS.cube 中。获得它们可以在工作区使用命令

```
pam -incmo -get "rhoL.cube rhoS.cube" ...
```

还要注意, 为了产生密度, 必须有因子文件 DFCOEF。

.PRINT 打印级别。

参量: 整型 IPRRHO

默认: IPRRHO = 0

.DOCUBE 指定大小分量密度的产生。

参量: 整型 ILC, ISC 值: 0 (关闭), 1 (打开)

默认: ILC = ISC = 1, 意味着大小分量密度都被产生

.NCUBE 沿着立方边界的体元数。

参量: 整型 NCUBE(1), NCUBE(2), NCUBE(3)

默认: NCUBE(1-3) = 80

.CUBADJ 校正立方的体积。

参量: 实型 CUBADJ(1), CUBADJ(2)

默认: CUBADJ(1)=4.000, CUBADJ(2)=8.000

DIRAC 将会检查原子位置并放置密度立方, 使得包含的密度最多。立方的原点通过在 x, y, z 方向分别寻找最小原子位置来确定, 接下来加入到 CUBADJ(1)中。张开(平行管状)立方的矢量长度类似地用最大原子位置定义, 加入到 CUBADJ(2)中。

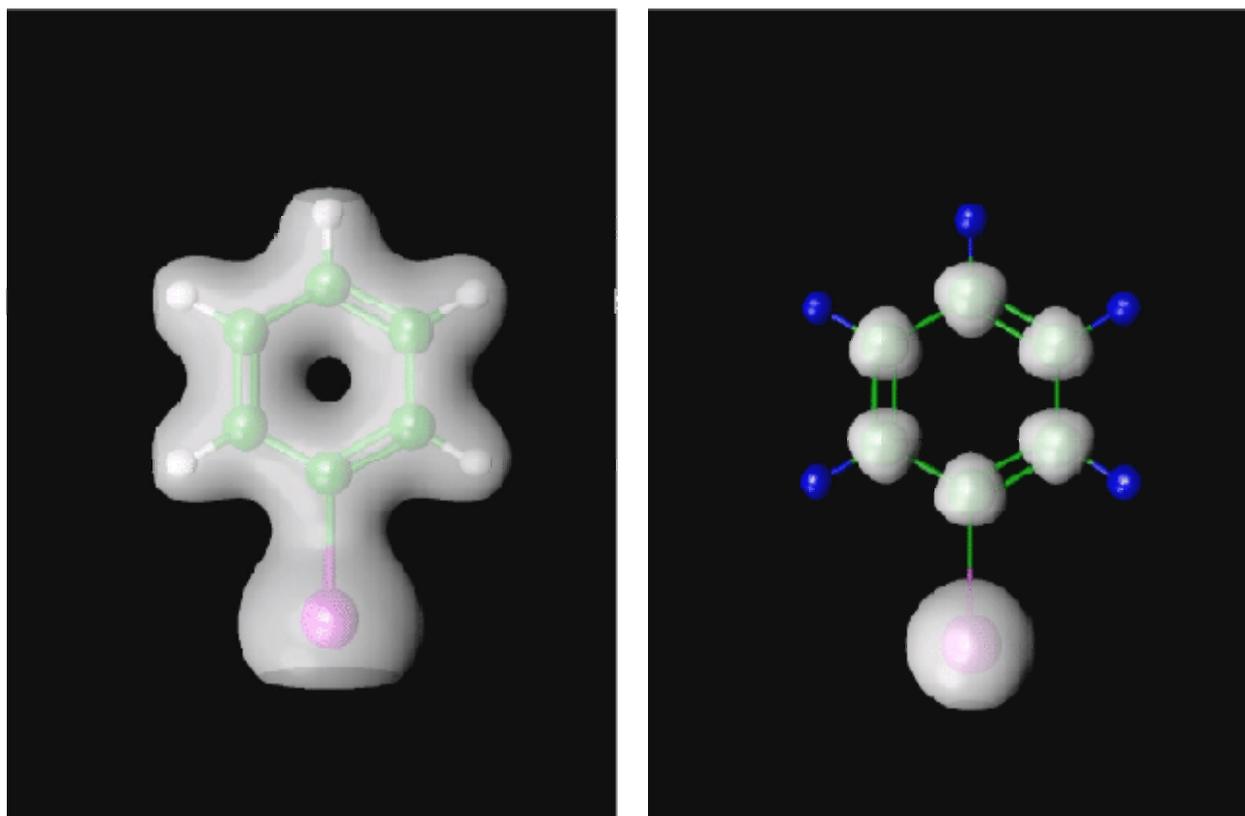


图 3.2: 碘苯的大分量和小分量密度。用 MOLEKEL 软件绘制, 使用的截断因子分别为 0.01 和 0.0001。



图 3.3: 还可以绘制改变占据的轨道密度, 通过减去由给定的因子文件产生的密度。图中显示了铈酰阳离子 HOMO 的大分量密度, 通过从常规占据产生的密度中, 减去去掉两个电子占据产生的密度。

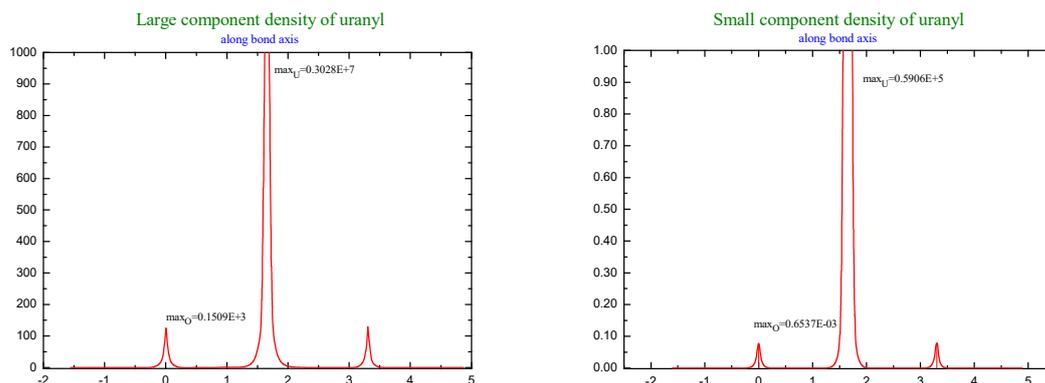
3.6.5 *RHO1——沿着成键打印密度

对每个唯一的原子中心对(A,B)产生格式化文件，包括四列数值：原点位于原子 A，第一列给出沿 A-B 中心连线的坐标，单位为埃。后面的列分别包含总密度，大分量密度和小分量密度(原子单位)。文件命名为' RH' ,NAMN(ICENTA),NAMN(ICENTB), IDEGB,其中 ICENTB 是对称无关中心 B 的名称, IDEGB 是相关中心的编号。只要有空格,就插入下划线。产生密度需要因子文件 DFCOEF。注意,原则上可以获得沿任何连线的密度,只要简单地插入虚原子。对于这里描述的铀酰的例子,可以使用命令

```
pam -incmo -get ' 'RHU_ _ _0_ _ _1 RHU_ _ _0_ _ _2' ' ...
```

.PRINT 打印级别。
参量: 整型 IPRH01
默认: IPRH01 = 0

.MESH 格点的步长。
参量: 实型 DSTEP
默认: DSTEP = 0.01D0



3.6.6 示例输入

一个打印费米不可约共表示 1 中所有轨道和 2 中某些轨道的例子,只打印大分量,使用非对称匹配的 AO 基:

```
**ANALYZE
.PRIVEC
*PRIVEC
.VECPRI    # what molecular orbitals to print
all
-4..2,4
.PRICMP    # print only large components
1 0
```

```
.AOLAB      # use A0-basis
```

一个 Mulliken 布居分析的例子，对每个原子壳层定义标记，并指定用于分析的轨道：

```
**ANALYZE
.MULPOP
*MULPOP
.LABDEF     # redefinition of labels
3
H(s)        1
H(p)        2..4
H(d)        5..10
.VECPop     # what molecular orbitals to analyze
1..10
1,2
```

使用两个片段进行投影分析的例子，片段分别位于因子文件 DFFRG1 和 DFFRG2 中，每一个在其自己的基中计算：

```
**ANALYZE
..PROJEC
*PROJECTION
.OWNBAS     # use the individual basis set of fragments
.VECPRJ     # what molecular orbitals to analyze
1,2
1..3
.VECREf     # definition of reference orbitals
2
DFFRG1
all
all
DFFRG2
1..4
1..2
```

3.7 **PROPERTIES——特性模块

这一部分讨论如何求解各种分子特性。如果所需的积分出现在 HERMIT 的积分菜单中，那么这种特性原则上能够被计算。可以计算的特性包括：

- 期待值（例如：偶极矩和电场梯度）
- 线性响应特性（例如：极化率和 NMR 参数）
- 二次响应函数（例如，超极化率）

为了方便，一些常用的特性可以在这一部分直接指定，这意味着用户通常不必知道它们是如何计算的。但是注意，响应函数默认是静态的，可以在有关的子部分（例如，*LINEAR

RESPONSE) 中包含频率。

未做预定义的特性必须在有关的输入部分详细指定 (参见 6.1 节)。

默认不计算特性。

1. 一般控制选项

- . PRINT 打印级别。
参量: 整型 IPRPRP
默认: IPRPRP = 0
- . ABUNDANCIES 对于涉及同位素的特性, 给出用于打印同位素的阈值级别 (单位: 丰度百分比)。
参量: 实型 ABUND
默认: 丰度大于 1.0%

2. 预定义的电特性

- . DIPOLE 求解电偶极矩 (期待值)。
- . QUADRUPOLE 求解电四极矩 (期待值)。
- . EFG 求解电场梯度 (期待值)。
原子中心可以用**INTEGRALS 下的. SELECT 进行限制。
- . NQCC 求解核四极耦合常数 (期待值)。
原子中心可以用**INTEGRALS 下的. SELECT 进行限制。
- . POLARIZABILITY 求解静态电偶极极化率 (线性响应函数)。
- . FIRST ORDER HYPERPOLARIZABILITY 求解静态电偶极一阶超极化率 (二次响应函数)。结果还给出静态电偶极极化率
- . VERDET 对于波长 694 nm, 静态磁场沿着光束传播方向的红宝石激光动态电场 (这种情况下, 二次响应函数的默认频率是, $\omega_B=0.0656$ 以及 $\omega_C=0$), 求解 Verdet 常数 (二次响应函数)。Verdet 计算不能与其它二次响应计算一起指定。
- . TWO-PHOTON 求解静态频率的双光子吸收交叉部分 (二次响应函数)。在每一个玻色对称性中输入所需的态数。不能与其它二次响应计算一起指定。
READ(LUCMD,*) (TPACNV(I), I=1, NBSYM)

3. 预定义的磁特性

- . NMR 求解核磁屏蔽和间接自旋-自旋耦合 (线性响应函数)。
原子中心可以用**INTEGRALS 下的. SELECT 进行限制。
- . SHIELDING 求解核磁屏蔽 (线性响应函数)。打印级别(IPRPRP) 2 给出张量, 以及较多的输出。打印级别(IPRPRP) 4 还给出对称坐标的粗略值。
原子中心可以用**INTEGRALS 下的. SELECT 进行限制。
- . SPIN-SPIN COUPLING 求解间接自旋-自旋耦合 (线性响应函数)。
原子中心可以用**INTEGRALS 下的. SELECT 进行限制。
- . DSO 作为期待值, 求解反磁性对间接自旋-自旋耦合的贡献。
原子中心可以用**INTEGRALS 下的. SELECT 进行限制。

. NSTDIAMAGNETIC

作为期待值，求解反磁性对核磁屏蔽的贡献。
原子中心可以用**INTEGRALS 下的. SELECT 进行限制。

4. 其它预定义的特性

. MOLGRD

求解分子梯度，即 $\frac{\partial E}{\partial X_A}$ ，其中的 X_A 是核坐标。这是单、双电子算符的期待值。通常不用这个关键字直接调用分子梯度，而是在几何优化模块的内部调用。

. PVC

对核自旋无关的违反奇偶的算符计算矩阵元素，例如计算对映异构体之间的能量差[31]。

3.7.1 *EXPECTATION VALUE——求解期待值

这一部分给出计算期待值的指令。

基本选项

. OPERAT

定义普通单电子算符。
详见 6.1 节。

高级选项

. ORBANA

按照各个轨道分析期待值。

程序员选项

. PRINT

打印级别。
参量：整型 IPREXP（默认是 0）

3.7.2 *LINEAR RESPONSE——线性响应

线性响应模块由 T. Saue 和 H. J. Aa. Jensen [32] 编写

随机相位近似（RPA）[32, 33]中线性响应函数的一般形式是：

$$\langle\langle A; B \rangle\rangle_{\omega} = -E_A^{[1]+} \left(E^{[2]} - \omega S^{[2]} \right)^{-1} E_B^{[1]} \quad (3.6)$$

其中 $E_A^{[1]}$ 和 $E_B^{[1]}$ 是对应于特性 A 和 B 的特性梯度， $E^{[2]}$ 是分子 Hessian， $S^{[2]}$ 是长度， ω 是频率。 $E^{[2]}$ 的维度一般不允许预解式 $\left(E^{[2]} - \omega S^{[2]} \right)^{-1}$ 的直接反演，所以一般借助于迭代技术。

我们写下：

$$\left(E^{[2]} - \omega S^{[2]} \right)^{-1} E_B^{[1]} = X \quad (3.7)$$

稍做整理，我们得到线性响应方程：

$$(E^{[2]} - \omega S^{[2]})X = E_B^{[1]} \quad (3.8)$$

RPA 方程通过把 X 展开到一组测试矢量来求解：

$$X = \sum_{i=1} b_i a_i; \quad Y = [b_1, b_2, \dots, b_n] \quad (3.9)$$

并求解约化方程

$$(\tilde{E}^{[2]} - \omega \tilde{S}^{[2]})a = \tilde{E}_B^{[1]} \quad (3.10)$$

其中，

$$\tilde{E}^{[2]} = Y^+ E^{[2]} Y; \quad \tilde{S}^{[2]} = Y^+ S^{[2]} Y; \quad \tilde{E}_B^{[1]} = Y^+ E_B^{[1]} \quad (3.11)$$

这一部分的关键字是：

1. 一般控制选项

.PRINT 打印级别。
 参量：整型 IPXLR（默认是 0）

2. 定义线性响应函数

.OPERAT 定义单电子算符（A 和 B）。
 详见 6.1 节。

.A OPER 定义单电子 A 算符。
 详见 6.1 节。

.B OPER 定义单电子 B 算符。
 详见 6.1 节。

.TRIAB 强制响应函数的三角化，也就是 $\langle\langle A; B \rangle\rangle_\omega = \langle\langle B; A \rangle\rangle_\omega$ ；只计算一个函数。
 默认：不激活

.B FREQ 定义算符 B 的频率。
 默认：只有静态情况。
 READ (LUCMD, *) NBFREQ
 READ (LUCMD, *) (BFREQ(J), J=1, NBFREQ)

.ALLCMB 构成所有可能的 $\langle\langle A; B \rangle\rangle_\omega$ ，即使出现虚数。
 默认：ALLCMB = .FALSE.

3. 控制变分参量

.SKIPEE 执行所有的电子-电子转动。
 默认：包含电子-电子转动

.SKIPEP 执行所有的电子-正电子转动。
 默认：包含电子-正电子转动

4. 控制约化方程

.MAXITR 迭代的最大数量。
 参量：整型 ITRXLR

- 默认: 50 次迭代
- . MAXRED** 在约化体系中, 矩阵的最大维数。
参量: 整型 MAXRM
默认: MAXRM = 200
- . THRESH** 约化体系收敛的阈值。
参量: 实型 THCLR
默认: THCLR = 1.0D-5

5. 控制积分的贡献

鼓励用户测试这些选项, 因为它们可能对运行时间有重要影响。

- . INTFLG** 指定包含哪些双电子积分。
参量: 整型 ILL,ISL,ISS
IXX = 1 (打开) / 0 (关闭) (XX = LL,SL 或 SS)
默认: 来自于**GENERAL 的 INTFLG
- . CNVINT** 在添加 SL-和 SS-积分之前, 设定收敛的阈值。
参量: 实型 CNVINT(1) (SL), CNVINT(2) (SS)
默认: 非常大的数
- . ITRINT** 在添加 SL-和 SS-积分之前, 设定迭代的数量。
参量: 整型 ITRINT(1) (SL), ITRINT(2) (SS)
默认: ITRINT(1) = ITRINT(2) = 1

6. 控制测试矢量

- . REAXVC** 从 XVCFIL 文件读取解的矢量。
参量: 字符串(A6) XVCFIL——文件名
默认: 不从解的矢量重新开始
- . XLRNRM** 归一化的测试矢量。使用归一化的测试矢量将减少屏蔽的效率。
默认: 使用非归一化的测试矢量

7. 高级/调试标记

- . ONLYSF** 在 sigma 矢量程序中只调用 FMOLI: 只产生一指数转换的 Fock 矩阵[32]。
- . ONLYSG** 在 sigma 矢量程序中只调用 GMOLI: 用一指数转换密度只产生双电子 Fock 矩阵[32]。
- . STERNH** 对于+/-转动, 也就是占据正能量轨道与虚的负能量轨道之间的转动, 设定 Hessian 轨道部分的对角元等于 $-2mc^2$ 。
默认: XSTERN = .FALSE.
- . COMPRE** 通过检查阈值以上梯度矢量相应的元素, 减少轨道变分参数的数目。这可以减少内存需求。
参量: 实型 THRCOM
默认: THRCOM = 0.0D0 (不压缩)
- . NOPREC** 不进行初始测试矢量的预处理。
默认: 进行初始测试矢量的预处理
- . RESFAC** 仅对变分参数类产生新的测试矢量, 它的残差模为最大模的倒数 1/RESFAC。
参量: 实型 RESXLR
默认: RESXLR = 1000.0D0

3.7.3 *NMR——控制 NMR 参数的求解

这一部分给出计算 NMR 参数的指令。

高级选项

- . GAUgeo** 读入用户定义的度规原点。注意为了一致，对度规原点和偶极原点都进行了改变。
默认: GAGORG = 0.0 0.0 0.0
READ (LUCMD, *) (GAGORG(ICOO), ICOR = 1, 3)

3.7.4 *MOLGRD——控制分子梯度的求解

这一部分给出用于控制单点求解分子梯度的指令，使用 **.MOLGRD** 激活。通常分子梯度模块从几何优化模块调入。

高级选项

- . DOTRCK** 如果激活 SSSL 和 SSSS 双电子积分的计算，它们对梯度的贡献如果估计很小的话就会被跳过。经验上基于 LL 双电子梯度的模，可以估计 LS 和 SS 双电子梯度的模。如果计算了所有对梯度不激活与激活的贡献，那么就算出了所要的贡献。
默认: 不激活
- . INTFLG** 指定包含哪些双电子积分。
参量: 整型 ILL,ISL,ISS
IXX = 1 (打开) / 0 (关闭) (XX = LL,SL 或 SS)
默认: 来自于**GENERAL 的 INTFLG

程序员选项

- . PRINT** 打印级别。小于 3 的打印级别只给出总梯度。3 以上的打印级别给出对梯度的各种贡献 (动能梯度, 核吸引梯度, 等)。5 以上的打印级别打印一些矩阵, 10 以上给出大量的输出!
参量: 整型 IPRGRD
默认: IPRGRD = IPRGEN
- . NUMGRA** 用数值差分计算分子梯度。
默认: 如果执行的话, 解析求解

3.7.5 *QUADRATIC RESPONSE —二次响应函数

这一部分给出计算二次响应函数[33]:

$$\langle\langle A; B, C \rangle\rangle_{\omega_1, \omega_2}$$

的指令。

基本选项

- .DIPLN 对 A, B, 和 C 指定偶极算符。详见 6.1 节。
- .A OPER 指定单电子 A 算符。详见 6.1 节。
- .B OPER 指定单电子 B 算符。详见 6.1 节。
- .C OPER 指定单电子 C 算符。详见 6.1 节。
- .B FREQ 指定算符 B 的频率。
默认: 只有静态的情况
READ (LUCMD, *) NBQFRF
READ (LUCMD, *) (BQFRF(J), J=1, NBQFRF)
- .C FREQ 指定算符 C 的频率。
默认: 只有静态的情况
READ (LUCMD, *) NCQFRF
READ (LUCMD, *) (CQFRF(J), J=1, NCQFRF)

高级选项

- .ALLCMB 求解所有非零二次响应函数, 因此忽略对全部置换对称性的分析。
默认: 仅求解唯一、非零的响应函数。
- .INTFLG 指定包含哪些双电子积分。
参量: 整型 ILL, ISL, ISS
IXX = 1 (打开) / 0 (关闭) (XX = LL, SL 或 SS)
默认: 来自**GENERAL 的 INTFLG
- .XQRNRM 对测试矢量归一化。使用归一化的测试矢量将会降低屏蔽的效率。
默认: 使用未归一化矢量
- .SKIPEE 排除所有的电子-电子转动。
默认: 包括电子-电子转动
- .SKIPEP 排除所有的电子-正电子转动。
默认: 包括电子-正电子转动
- .MAXITR 迭代的最大数量。
参量: 整型 ITRXQR
默认: 30 次迭代
- .MAXRED 约化体系中, 矩阵的最大维度。
参量: 整型 MAXQRM
默认 MAXQRM = 100
- .THRESH 约化体系中的收敛阈值。
参量: 实型 THCQR
默认: THCQR = 1.0D-5
- .ITRINT 设定加入 SL 和 SS 积分前的迭代数。
参量: 整型 ITRIQR(1) (SL), ITRIQR(2) (SS)
默认: ITRIQR(1) = ITRIQR(2) = 1
- .CNVINT 设定加入 SL 和 SS 积分前的收敛阈值。
参量: 实型 CNVXQR(1) (SL), CNVXQR(2) (SS)

默认：非常大的数

程序员选项

- . PRINT** 打印级别。
参量：整型 IPREXP（默认为 0）
- . NOPREC** 不进行初始测试矢量的预处理。
默认：进行测试矢量的预处理
- . RESFAC** 仅对残差模大于最大模的 1/RESFAC 的那些变分参数类产生新的测试矢量。
参量：实型 RESXQR
默认：RESXQR = 1000.000

3.7.6 输入示例

计算 NMR 参数：

```
**PROPERTIES  
. NMR
```

计算含频极化率，跳过占据正能轨道和负能虚轨道之间的转动：

```
**PROPERTIES  
. POLAR  
*LINEAR RESPONSE  
. B FREQ  
  3  
  0.0 0.1 0.2  
. SKIPEP
```

3.8 **MOLTRA——积分转换模块

这一模块把标量 AO 基的积分转换为分子旋量基的积分。算法不再详述，阅读[34]找到关于所用的双四元指数方法的信息，可能会有帮助。MOLTRA 模块通常被某个关联能计算模块调用，但也可以作为独立的模块，通过在输入的**DIRAC 部分指定.MOLTRA 来运行。输入改变默认的活性空间，并指定特殊的要求（例如.INTFLG 的修改）。由三种不同的变换：

- 芯 Fock 矩阵变换，给出有效单电子矩阵
- 双电子库仑积分的变换
- 对一般（单电子）算符积分的变换

输出文件分别写到 MDCINT, MRCONEE 和 MDPROP 中，并可以被 RELCCSD 和 DIRRCI 程序读入进行有关的计算。有些公共的输入需要在这一部分和在 DIRRCI 中直接指定（参见**DIRRCI 部分）。

基本选项

- . ACTIVE** 定义活性旋量集。
参量：对每一个费米不可约表示，给出一行轨道串（参见 3.13.1 节）

默认: energy -20.0 10.0 1.0

高级选项

- . INTFLG** 指定在二指数和四指数的转换中, 包含哪些双电子积分。如果需要在二指数和四指数的转换中分别使用不同的积分类型, 那么使用下面的关键字. **INTFL2** 和 **INTFL4**。
参量: 整型 ILL, ISL, ISS
IXX = 1 (打开) / 0 (关闭) (XX = LL, SL 或 SS)
默认: 来自于**GENERAL 的 INTFLG
- . INTFL2** 指定在二指数的转换中 (也就是在 Fock 芯矩阵的双电子部分), 包含哪些双电子积分。
参量: 整型 ILL, ISL, ISS
IXX = 1 (打开) / 0 (关闭) (XX = LL, SL 或 SS)
默认: 来自于**GENERAL 的 INTFLG
- . INTFL4** 指定在四指数的转换中, 包含哪些双电子积分。
参量: 整型 ILL, ISL, ISS
IXX = 1 (打开) / 0 (关闭) (XX = LL, SL 或 SS)
默认: 来自于**GENERAL 的 INTFLG
- . CORE** 定义冻芯旋量。
参量: 对每一个费米不可约表示, 给出一行轨道串
默认: 所有轨道都不激活
- . POSITRONS** 活性集合中允许正电子旋量。如果没有指定该关键字, 那么仅包含电子旋量, 任何用. **ACTIVE** 关键字指定的正电子旋量都会被忽略。
- . PRPTRA** 进行特性积分转换。
默认: 跳过特性积分转换
- . NO2IND** 跳过有效 Fock 矩阵的二指数变换。
默认: 不跳过二指数变换
- . NO4IND** 跳过四指数变换。
默认: 不跳过四指数变换
- . SCREEN** 四指数变换中的屏蔽阈值。
默认: SCREEN = 1.0D-14
(负值禁止屏蔽。)
- . THROUT** 把转换后的双电子积分写入文件的阈值。
默认: THROUT = 1.E-14
- . RCORBS** 在转换前识别轨道。
默认: RCORBS = .FALSE.

程序员选项

- . PRINT** 打印级别。
参量: 整型 IPRTRA
默认: IPRTRA = 0
- . 4INDEX** 在四指数转换模块中, 为指数 1 至 4 以及费米子不可约表示 1 和 2 指定活性轨道的范围。
默认: **.ACTIVE** 关键字设定的范围

- . 2INDEX 在二指数转换模块中, 为指数 1 和 2 以及费米子不可约表示 1 和 2 指定活性轨道的范围。
默认: .ACTIVE 关键字设定的范围
- . SCHEME 定义积分变换的算法。
默认: SCHEME = 4
- . MDCINT 把四指数变换积分写入文件。
默认: MDCINT = .TRUE.
- . SCATTER 把 MDCINT/4IND 分布到所有的节点上。
默认: SCATTER = .TRUE.
- . NOSCAT 不把 MDCINT/4IND 分布到所有的节点上。
默认: 分布到所有的节点上
- . PAR4BAS 在四指数变换中, 为每一节点指定批处理数量。只对并行发布版有效。
默认: PAR4BAS = -1

3.8.1 *PRPTRA——特性积分变换

这一部分定义需要被转换的特性算符。这些积分仅用于 RELCCSD 的测试部分, 因此这一小节一般可以忽略。

程序员选项

- . OPERAT 定义一般单电子算符。详见 6.1 节。
- . PRINT 打印级别。
参量: 整型 IPRTRP
默认: IPRTRP = 0

3.8.2 示例输入

```

**MOLTRA
# use all spinors with energies between -12.0 and 20.0 au as active space
# note that one may NOT place a comment line directly after .ACTIVE
.ACTIVE
energy -12.0 20.0 1.0

```

3.9 RELCCSD——耦合簇模块

相对论耦合簇模块 RELCCSD 由 Lucas Visscher 编写

这一部分列出相对论耦合簇计算的必要输入。程序目前可以在 MP2, CCSD 和 CCSD(T) 理论级别上进行能量计算[35, 36], 其中一阶特性只能用于 MP2 理论级别。开壳层的态可以通过 Fock 空间耦合簇方法[37]进行计算, 它为闭壳层的芯加入或减去电子。在涉及的所有部分中, 加入或减去的电子不能多于两个。输入应当以名称列表的形式给出, 进一步又分为

一般菜单部分和描述能量、一阶和二阶特性计算的部分。

注意，CCSOPR 所需的积分必需在**MOLTRA 部分指定。

3.9.1 RELCCSD——基本输入和控制输入

基本选项

- DOENER** 计算能量。
默认: **DOENER** = .TRUE.
- DOFOPR** 计算(松弛)密度矩阵和一阶特性。
默认: **DOFOPR** = .FALSE.
- DOFSPC** 运行 Fock 空间 CC 计算。
默认: **DOFSPC** = .FALSE.

高级选项

- NELEC** 在两个偶(g)不可约表示或两个奇(u)不可约表示中的电子数。这个变量决定了在波函数的指数展开中所用的参考行列式。
参量: 整型(**NELEC**(I), I=1, NFSYM*2)
默认: 这些不可约表示中的活性电子(由 MOLTRA 写入)
- NELEC_F1(2)** 在阿贝尔对称群的奇(偶)不可约表示中的电子数。
参量: 整型(**NELEC_F1**(I), I=1, 不可约表示的个数)
默认: MOLTRA 或 NELEC 设定的值
- IPRNT** 设定打印级别。级别 1 给出 CCSD 迭代的收敛信息, 可能会有用。
默认: 整型 **IPRNT** = 0
- TIMING** 写出时间信息。
默认: **TIMING** = .FALSE.

详细的 NELEC_F1 和 NELEC_F2 输入主要是用于程序使用 D_{oh} 对称群的原子计算。在这种情况下该选项允许精确描述参考行列式。后面我们给出了氮原子高自旋态的例子。

程序员选项

- DEBUG** 写出调试信息。
默认: **DEBUG** = .FALSE.
- DOSORT** 按照分子旋量对得到的积分进行分类。这是必需的步骤, 只有在特定的情况下(例如重新计算)才可以被跳过。
默认: **DOSORT** = .TRUE.
- NFROZ** 每个不可约表示中冻结电子的数量。令与冻结旋量有关的 T1 和 T2 的振幅为 0。仅用于测试的目的。
参量: 整型(**NFROZ**(I), I=1, NFSYM*2)
默认: 0

3.9.2 CCENER——能量模块的输入

基本选项

DOMP2	计算 MP2 能量。 默认: <code>DOMP2 = .TRUE.</code>
DOCCSD	计算 CCSD 能量。 默认: <code>DOCCSD = .TRUE.</code>
DOCCSDT	计算 CCSD(T)能量。实际上程序计算三种不同的非迭代三阶修正 (参见 [35])。 默认: <code>DOCCSDT = .TRUE.</code>

高级选项

NTOL	DIIS 误差矢量的收敛阈值 ($10^{-\text{NTOL}}$) 参量: 整型 NTOL 默认: <code>NTOL = 12</code>
MAXIT	求解 CCSD 方程的 DIIS 迭代最大数量 参量: 整型 MAXIT 默认: <code>MAXIT = 30</code>
MAXDIM	约化空间的最大维数。 参量: 整型 MAXDIM 默认: <code>MAXDIM = 8</code>

程序员选项

NOCCS	忽略单激发, 使用 CCD 的形式。仅用于测试目的。 默认: <code>NOCCS = .FALSE.</code>
NOCCD	忽略双激发, 使用 CCS 的形式。仅用于测试目的。 默认: <code>NOCCD = .FALSE.</code>

3.9.3 CCSORT——分类模块的输入

基本选项

USEOE	使用来自于 HF 计算的轨道能量, 而不是重构 Fock 矩阵的对角元。 默认: <code>USEOE = .TRUE.</code>
-------	---

3.9.4 CCFOPR——一阶特性的输入

目前只有 MP2 一阶特性可用。它们通过构成松弛密度矩阵来计算, 这 (部分地) 在 AO 基中完成, 以获得全部的松弛贡献。求解的特性在输入的期待值部分指定。

基本选项

DOMP2G 计算 MP2 梯度，也就是用松弛单电子密度矩阵计算期待值。注意这不是分子的梯度，而是在输入的期待值部分指定的一阶特性。
默认: **DOMP2G** = .FALSE.

高级选项

NTOL DIIS 误差矢量的收敛阈值($10^{-\text{NTOL}}$)。
参量: 整型 **NTOL**
默认: **NTOL** = 12

MAXIT 求解 Z 矢量方程的迭代最大数量。
参量: 整型 **MAXIT**
默认: **MAXIT** = 30

MAXDIM 约化空间的最大维数。
参量: 整型 **MAXDIM**
默认: **MAXDIM** = 8

3.9.5 CCFSPC——Fock 空间模块的输入

基本选项

DOEA 计算电子亲和能。
默认: **DOEA** = .FALSE.

DOIE 计算电离能。
默认: **DOIE** = .FALSE.

DOEA2 计算双电子亲和能 (加入两个电子)。
默认: **DOEA2** = .FALSE.

DOIE2 计算双电离能 (去除两个电子)。
默认: **DOIE2** = .FALSE.

NACTP 允许放入电子的活性旋量数目。这一变量确定电子亲和能 (EA) 计算的活性空间。
参量: 整型 (**NACTP**(I), I=1, NFSYM*2)
默认: 无默认值, 应当总是要对 **DOEA** 或 **DOEA2** 设定!

NACTH 允许产生空穴的活性旋量数目。这一变量确定了 IE 计算的活性空间。
参量: 整型 (**NACTH**(I), I=1, NFSYM*2)
默认: 无默认值, 应当总是要对 **DOIE** 或 **DOIE2** 设定!

高级选项

NTOL DIIS 误差矢量的收敛阈值($10^{-\text{NTOL}}$)。
参量: 整型 **NTOL**
默认: **NTOL** = 12

MAXIT 求解 CCSD 方程的 DIIS 迭代最大数量。
参量: 整型 **MAXIT**
默认: **MAXIT** = 30

MAXDIM 约化空间的最大维数。
参量: 整型 **MAXDIM**

默认: **MAXDIM** = 8

程序员选项

FSSECT 活性 Fock 空间部分 (按照 00,01,10,11,02,20 的顺序)。
默认: **FSSECT** = 0, 0, 0, 0, 0, 0

3.9.6 RELCCSD 的示例输入

示例输入 1: 计算 CCSD(T)能量和 MP2 一阶特性。完整的计算在 test 目录中作为输入例子 1.energy+dipole 给出。

```
&RELCCSD TIMING=T, DOENER=T, DOFOPR=T &END
&CCFOPR DOMP2G=T &END
```

示例输入 2: 计算氮原子的 CCSD 能量。我们把 $|\frac{1}{2}, \frac{1}{2}; \frac{3}{2}, \frac{1}{2}; \frac{3}{2}, \frac{3}{2}|$ (符号: $j_1, mj_1; j_2, mj_2$; 等) 作为参考行列式。

```
&RELCCSD NELEC_F1=1, 1, NELEC_F2=2, 0, 1, 0 DOENER=T &END
&CCSORT USEOE=F &END
```

3.10 DIRRCI——直接 CI 模块

相对论 RASCI 模块由 Lucas Visscher 编写

这一部分进行相对论组态相互作用计算。注意当名称列表 GOSCIP 也出现在输入中时, 将调用 GOSCIP 而不是 DIRRCI!

输入以名称列表的形式给出。

3.10.1 RASORB——指定 CI 的类型和活性空间

基本选项

NELEC (除了冻芯电子之外的) 电子数。
默认: **NELEC** = 0

NRAS1 RAS1 空间中, 每个阿贝尔不可约表示的旋量数量。
默认: **NRAS1** = NSYMRP * 0

NRAS2 RAS2 空间中, 每个阿贝尔不可约表示的旋量数量。
默认: **NRAS2** = NSYMRP * 0

MAXH1 RAS1 旋量的最大空穴数。
默认: **MAXH1** = 0

MAXE3 RAS3 旋量的最大电子数。
默认: **MAXE3** = 0

3.10.2 CIROOT——选择进行收敛的电子态

基本选项

- IREPNA** 所需电子态的阿贝尔对称群。¹
默认：列表中的第一个阿贝尔对称性。
- NROOTS** 进行优化的态的数量。
默认：**NROOTS** = 1

高级选项

- ISTART** 初始矢量的方法：
1. COSCI 初始矢量。
2. 有最低本征值的行列式。
3. 第一个（参考）行列式。
如果 COSCI 初始矢量不可用，默认是 3。
- NSEL (NROOTS)** 优化的态的序列。
默认：**NSEL** = 1, 2, 3, ...
- SELECT** 选择与初始波函数有最大重叠的波函数。
默认：**SELECT** = F

3.10.3 DIRECT——收敛控制

基本选项

- CONVERE** 能量的收敛阈值。
默认：**CONVERR** = 1.0D-9
- MAXITER** 直接 CI 迭代的最大数量。
默认：**MAXITER** = 10

高级选项

- CONVERR** 残留矢量的收敛阈值。
默认：**CONVERR** = 1.0D-10
- RESTART** 对出现在 MRCFINV 中的 CI 矢量重新开始。
默认：**RESTART** = F
- CPUMAX** 使用 CPU 秒的最大数量。
默认：**CPUMAX** = 604800

3.10.4 OPTIM——算法的精细调整

程序员选项

- IGENEX** 把耦合因子写到文件中(2)，或者在需要的时候计算(1)。

¹ 四个字符的字符串，包含在单引号或双引号之中。需要注意空格数量、位置和字符的大小写。见示例。

默认: IGENEX = 2

3.10.5 LEADDET——分析 CI 波函

高级选项

GETDET 得到主行列式的列表。

默认: GETDET = T

COMIN 仅在因子的平方大于 COMIN 的时候, 打印行列式的贡献。

默认: COMIN = 0.1

3.10.6 DIRRCI 的示例输入

示例输入

```
**MOLTRA
. ACTIVE
2..5
1..12
*END OF

&RASORB NELEC=7, NRAS1=1, 1, NRAS2=2*0, 3, 3, MAXH1=2, MAXE3=2 &END
&CIROOT IREPNA=' 1Eu', NROOTS=3 &END
&DIRECT MAXITER=15 &END
```

3.11 GOSCIP——COSCI 模块

一般开壳层 CI 程序由 Olivier Visser 编写

这一模块进行小的完全组态相互作用计算。在 DIRRCI 中如果输入出现名称列表 GOSCIP, 或者在进行开壳层 HF 计算时定义了 .RESOLVE, 则调用 GOSCIP。输入应当以名称列表的形式¹ 给出。

3.11.1 GOSCIP——定义 CI 空间

基本选项

NELEC (除了冻芯电子之外的) 电子数量。

默认: NELEC = 0

¹ 关于历史的注释: 这一程序最初为 MOLFDIR 程序[38]编写, 后来又添加到 DIRAC 中。

程序员选项

IPRNT 打印级别。
默认: IPRNT = 0

3.11.2 POPANA——波函数分析

高级选项

THRESH 仅打印因子高于阈值 THRESH 的行列式。
默认: 1.0D-3

DEGEN 当认为几个态简并时的阈值。
默认: 1.0D-10

SELPOP 仅选择相对能量低于 SELPOP 的态。
默认: 1.0D2

3.11.3 GOSCIIP 的示例输入

示例输入

```
&GOSCIIP NELEC=5, IPRNT=1 &END  
&POPANA THRESH=1.0D-4, DEGEN=1.0D-12, SELPOP=50.0 &END
```

3.12 LUCITA——直接 GAS CI 模块

无自旋相对论 GASCI 模块由 Jeppe Olsen 编写, Timo Fleig 改写到 DIRAC 中

LUCITA 是基于串的直接哈密顿量通用组态相互作用 (CI) 程序, 它基于 Jeppe Olsen (丹麦奥尔胡斯大学理论化学系) 写的 LUCIA 程序。它使用一般活性空间, 可以进行任意激发级别的高效 CI 计算, 例如 FCI, SDCI, RASCI 和 MRCI。该代码接口到无自旋 Dirac 方法并使用非相对论点群对称性的分子积分中[39]。

程序的核心功能是广义活性空间 (Generalized Active Space, GAS) 概念, 其中基本的总轨道空间进一步被分为任意数量的 (直到上限), 有任意占据数限制的子空间。这是细分轨道空间的最一般办法。程序使用的 DIRAC 轨道来自于无自旋相对论方法或非相对论 Lévy-Leblond 方法的闭壳层或开壳层计算 (见 3.4 节)。

设定的技术限制, CI 展开的行列式在普通 PC 上大约为 100 兆, 大型 PC 上大约为 400 兆, 充足内存的超级计算机上大约为 1-2 千兆。对于包含 100,000 个以上行列式的计算, 特别是双激发以上的高激发计算, 强烈建议在无自旋 DIRAC 应用中使用 LUCITA。

如果需要, 程序还可以从优化的 CI 波函数计算单、双粒子密度。密度矩阵可以和自然轨道占据以及相应的本征矢 (NO) 一起打印。

代码中的 LUCITA 与接口部分 DIRLUC 一起提供。DIRLUC 环境读取 DIRAC 输入, 通过 ASCII 码文件转化为 LUCIA 输入。可以用关键字指定的 CI 类型有: 完全 CI (FCI), 单双激发 CI (SDCI), 单双三四激发 CI (SDTQ), 限制性活性空间 CI (RASCI) 以及广义活性空间 CI (GASCI)。

所有类型的 CI 都可以看作是 GASCI 的特例，因此在程序内部，DIRLUC 把它们的信息都转化为 GAS 格式。

3.12.1 一般输入

强制的关键字

INIWFC	初始 HF 波函（闭壳层或开壳层）。 值：DHFSCF 或 OSHSCF 默认：无
CITYPE	CI 计算的类型。 值：FCI, SDCI, SDTQ, RASCI, GASCI 默认：无
MULTIP	态的自旋多重度。 默认：无

可选的关键字

TITLE	CI 计算的一行标题。 默认：Default title
NROOTS	要优化的态数。 默认：NROOTD = 1
SYMMET	态的对称性。 默认：ISSYMD = 1
SZCALC	计算的近似大小。 值：NOR 用于常规 CI, LAR 用于大 CI（100 兆以上的行列式），HUG 用于巨型 CI。 默认：SZCALD = NOR
INACTI	每个玻色对称性中的非活性轨道，用逗号隔开。 默认：全部轨道为活性
DENSI	计算密度矩阵的级别。 值：1 仅计算单粒子密度矩阵，2 计算单、双粒子密度矩阵 默认：IDENSD = 0
RSTRCI	可选，从 LUCVECT 文件中的 CI 矢量重新开始。 值：1 重新开始计算 默认：IRSTLT = 0
PRINTL	用于 DIRLUC 接口的本地打印标记。 值：0-10 等级标记 默认：IPRNLD = 0
PRINTG	LUCIA 的全局打印标记。 值：0-4 等级。使用要当心！设置为 1 用于打印密度矩阵（如果指定的话）以及自然轨道。 默认：IPRNGD = 0

3.12.2 特殊输入

GASCI

NACTEL

活性电子数。

默认：无

GASSHE

GAS 轨道的数目和说明。

使用的 GA 空间数

值：1-16 等级

默认：无

GA 空间说明。每个 GAS 一行，包括每个玻色对称性的轨道数，用逗号隔开

默认：无

GASSPC

一系列 CI 计算的数目和说明。

给定 GA 空间的 CI 计算数目（目前只允许 1!）

默认：无

CI 计算的说明。每个 GAS 一行，包括两个数：第一个给出在这个 GAS 之后最小的累积电子数，第二个对应于最大累积电子数，之间用空格隔开（定义了每个 GAS 的占据限制）。

默认：无

RASCI

NACTEL

活性电子数。

默认：无

INACTI

每个玻色对称性中的活性轨道，用逗号隔开。

默认：无

RAS1

RAS1 说明以及空穴的最大数量。

说明：每个玻色对称性中轨道数的一行，用逗号隔开

默认：无

RAS1 中最大空穴数

默认：无

RAS2

RAS2 说明。

说明：每个玻色对称性中轨道数的一行，用逗号隔开

默认：无

RAS3

RAS3 说明以及电子的最大数量。

说明：每个玻色对称性中轨道数的一行，用逗号隔开

默认：无

RAS3 中最大电子数

默认：无

3.12.3 LUCITA 的示例输入

SDCI 示例输入

```

*LUCITA
. INIWFC
  DHFSCF
. CITYPE
  SDCI
. MULTIP
  3
*END OF

```

GASCI 示例输入

```

*LUCITA
. TITLE
  Aluminum atom in C2h symmetry, 7s5p2d (L) basis, SDT/SD CI (1s inactive).
. INIWFC
  OSHSCF
. CITYPE
  GASCI
. MULTIP
  2
. NACTEL
  13
. GASSHE
  5
  1, 0, 0, 0   ! 1s
  2, 0, 0, 0   ! 2s3s
  0, 0, 2, 4   ! 2p3p
  5, 2, 1, 2   ! 4s5s 4p 3d
  5, 2, 2, 4   ! 6s7s 5p6p 4d
. GASSPC
  1
  2 2
  3 6
  11 13
  11 13
  13 13
*END OF

```

3.13 特殊功能

3.13.1 轨道串

可以用两种方法指定轨道的子集。指定关键字“energy”，之后是分别定义下限，上限

以及准简并阈值的三个实数，这通常就足够了。这样就选择了两个阈值间的所有轨道。如果上限或下限穿过一组能量非常接近的轨道，准简并阈值可以延伸能量范围。它会移动上限或下限，直到使轨道间的能隙足够大。例如在 AO-MO 积分变换程序中，指定活性空间用于关联计算的默认值用字符串给出为

```
energy -10.0 20.0 1.0
```

这将把-10.0 和+20.0 a.u.能量之间，能隙最小为 1.0 a.u.的所有轨道挑选出来。更详细的控制可以用轨道串。给出一个字符串，其中列出用逗号分开的不同轨道，或者给出轨道的范围。范围的界限由“..”分开。正能量（电子）和负能量（正电子）的轨道分别用正、负数指定。例如对轨道串：

```
-5..5,7
```

将选出五个较上的正电子轨道和五个较低的电子轨道，加上第七个电子轨道。如果需要包括正电子和电子的所有轨道，则可以写：

```
all
```

轨道串的概念为定义轨道提供了极大的灵活性，可作为编辑矢量文件的另一种工具。但缺点是，用户需要人工数出轨道数量，对于有大量需轨道的情况，这会变得很麻烦。

第四章 DIRAC 基文件

基文件定义当前的基组，分子结构，以及体系的对称性。在相对论计算中选择合适的基组很重要，因为程序提供的基组仅支持周期表中的一部分元素。对于轻元素（直到氩），建议使用标准的非相对论基组，并用动能平衡产生小分量基组。对于更重的元素，最好使用 K.G. Dyal 开发的关联一致基组（更完整的说明参见基组目录）。还可以使用基组库中 K. Faegri 的基组，但需要用户自行添加极化函数进行扩展。这两种基组在 DIRAC 中都使用非收缩形式。小分量基组通过动能平衡的预设条件自动生成，但（有经验的用户）也可以直接指定。只有进行理论研究并且了解小分量基组标准的用户，才可以尝试后一种办法，要知道程序中还有很多缺陷！

下面用一个例子来解释详细的输入。

4.1 标题与对称性

```
1:DIRAC
2:Water
3:aug-cc-pVDZ basis (note that a tight p has NOT been added)
4:C 2 2 X Y
```

- 1 KEYWRD (A6).
DIRAC 中未用过的关键字。可以是任何字符序列。
- 2-3 TITLE(1), TITLE(2) (A72/A72).
两个任意的标题行。
- 4 CRT, NONTYP, KCHARG, SYMTXT, ((KASYM(I, J), I=1, 3), J=1, 3), ID3, THRS
(A1, I4, I3, A2, 10A1, D10. 2).
CRT
必须设定为“C”，表示使用直角高斯函数。否则，DIRAC 将发出警告，并继续假定为直角高斯函数。
注意：到球谐基的变换（对小分量做了修改）默认是在转换到正交基时完成的。
- NONTYP**
指定原子类型的数量。本例中是 2。
- KCHARG**
分子的电荷。如果未直接指定 Hartree-Fock/Kohn-Sham 的初始占据，将由它来确定。如果没有赋值，假定是 0。
- SYMTXT**
指定对称群中生成元的数量。对于 C_{2v} 有两个生成元，在本例中是 σ_x 和 σ_y 。如果在计算中不使用对称性，需要直接设定为零“0”。如果 SYMTXT 留空，那么 DIRAC 将会分析结构，并寻找合适的对称群。对于线型分子，要想在某些模块中使用完全对称群 $D_{\infty h}$ 或 $C_{\infty v}$ ，必须这么做。
- KASYM(I, J)**

对称性限定为二元群，也就是 D_{2h} 及其子群，这意味着作用于主轴(x,y,z)上的对称操作最多只能反转其方向。因此可以用三个字符构成的字符串来标识每个群生成元，对应在该操作下发生反转的轴。表 4.1 中给出了 8 个二元群的例子。在本例中，X 和 Y 分别表示生成元 σ_x 和 σ_y 。

ID3

如果未留空，则读入的分子坐标单位为埃，而不是原子单位 Bohr。

THRS

忽略最后积分的阈值。若没有给出则用默认值 1.0D-15。阈值 1.0D-15 将使积分精确到 1.0D-13。在菜单文件中，还可以分别指定 SL-和 SS-积分的阈值(参见 3.3.3 节的 THRFAC 关键字)。

表 4.1: 指定二元群的示例 (“-” 表示空格字符)。

群	SYMTXT	KASYM	操作
D_{2h}	3	--Z--Y--X	$\sigma(xy), \sigma(xz), \sigma(yz)$
D_2	2	XY--YZ---	$C_2(z), C_2(x)$
C_{2v}	2	-Y-X-----	$\sigma(xz), \sigma(yz)$
C_{2h}	2	--ZXYZ---	$\sigma(xy), i$
C_2	1	XY-----	$C_2(z)$
C_s	1	--Z-----	$\sigma(xy)$
C_i	1	XYZ-----	i
C_1	0	-----	

4.2 原子坐标

```

1:DIRAC
2:Water
3:aug-cc-pVDZ basis (note that a tight p has NOT been added)
4:C 2 2 X Y
5: 8. 1
6:0 .0000000000 0.0000000000 -.2249058930

```

5 Q, NONT(I), QEXP (1X, F9. 0, I5, F20. 5)

Q

该原子类型的核电荷。对于氧原子是 8。

NONT

在该原子类型中，对称不等的原子个数。水中的氧是 1。

QEXP

核电荷分布的高斯指数（如果留空使用默认值）。

6 NAMN, (CORD(J), J=1, 3) (A4, *)

NAMN

原子名称。对相同类型的每个原子建议使用不同的名称，以便在输出中区分它们。

CORD

以 Bohr 为单位（原子单位）的 x-, y-和 z-原子坐标。如果想要输入的坐标单位为埃，那么卡 4 中的 ID3（见 4.1 节）必须非空。直角坐标以自由格式给出。但是原子名称只能占用最左边的四列，因此最开始的四个位置不能输入坐标。

4.3 大分量基组

在原子坐标之后定义大分量基组。有多种设定基组的方法。

4.3.1 来自基组库的基组

```
1:DIRAC
2:Water
3:aug-cc-pVDZ basis (note that this is a nonrelativistic basis)
4:C 2 2 X Y
5: 8. 1
6:0 .0000000000 0.0000000000 -.2249058930
7:LARGE BASIS aug-cc-pVDZ
```

7 BSET, BSKEYWORD, BASFIL (A5, 1X, A5, 1X, A)
BSET
 必须是 LARGE。
BSKEYWORD
 当调用基组库中的基组时，必须是 BASIS。
BASFIL
 基组名称。

4.3.2 直接输入基组

```
1:DIRAC
2:Water
3:aug-cc-pVDZ basis (note that a tight p has NOT been added)
4:C 2 2 X Y
5: 8. 1
6:0 .0000000000 0.0000000000 -.2249058930
7:LARGE INTGRL 3 1 1 1
8:# S-TYPE FUNCTIONS
9:f 10 4
10: 11720.0000000 0.00071000 -0.00016000 0.00000000 0.00000000
11: 1759.0000000 0.00547000 -0.00126300 0.00000000 0.00000000
```

12:	400.8000000	0.02783700	-0.00626700	0.00000000	0.00000000
13:	113.7000000	0.10480000	-0.02571600	0.00000000	0.00000000
14:	37.0300000	0.28306200	-0.07092400	0.00000000	0.00000000
15:	13.2700000	0.44871900	-0.16541100	0.00000000	0.00000000
16:	5.0250000	0.27095200	-0.11695500	0.00000000	0.00000000
17:	1.0130000	0.01545800	0.55736800	0.00000000	0.00000000
18:	0.3023000	-0.00258500	0.57275900	1.00000000	0.00000000
19:	0.0789600	0.00000000	0.00000000	0.00000000	1.00000000
20:# P-TYPE FUNCTIONS					
21:f	5	3			
22:	17.7000000	0.04301800	0.00000000	0.00000000	
23:	3.8540000	0.22891300	0.00000000	0.00000000	
24:	1.0460000	0.50872800	0.00000000	0.00000000	
25:	0.2753000	0.46053100	1.00000000	0.00000000	
26:	0.0685600	0.00000000	0.00000000	1.00000000	
27:# D-TYPE FUNCTIONS					
28:f	2	2			
29:	1.1850000	1.00000000	0.00000000	0.00000000	
30:	0.3320000	0.00000000	1.00000000	0.00000000	

7 BSET, BSKEYWORD, IQM, (JCO(K), K=1, IQM) (A5, 1X, A6, 12I5)

为了向下兼容，关键字 **BSKEYWORD** 可以不出现，如：

```
7:LARGE 3 1 1 1
```

这种情况下按照 BSET, IQM, (JCO(K), K=1, IQM) (A5, 12I5) 的格式读取。

BSET

必须是 LARGE。

BSKEYWORD

对直接输入的基组必须是 INTGRL 或 EXPLICIT。

IQM

最高角量子数 l 加 1，例如 s 是 1， p 是 2，等。本例中是 3，因为我们使用 *spd* 基组。

JCO

对每个 l 值设定数据区的个数。在一个数据区中，所需内存随着基函数的数量迅速增长（例如 4 个 g 函数实际上有 60 个基函数，因为每个 g 函数有 15 个直角分量）。因此通过把某个量子数的基函数分割为不同的数据区，可以大大减少内存需求。但是这会降低积分计算的执行效率。

8, 20, 27 由!, \\$, 或#开始的注释行（可选）。

9, 21, 28 FRMT, NUCIJ, NRCIJ, ISGEN (A1, I4, 2I5)

FRMT

描述本数据区内基组输入格式的一个字符。如果 **FRMT** 留空，那么使用默认的格式 (8F10.4)。在这种格式中，第一列是轨道指数，后七列是收缩因子。没有给出数值的列假定因子都是 0。若一个数据区内有 7 个以上的收缩函数，收缩因子可以续行，但第一列（给出轨道指数的位置）必须留空。

第一个位置中的 F 或 f 表示输入是自由格式。这需要写入所有的收缩因子，每一

行都要输入数值。不过，该选项与完全非收缩基组一起使用时特别容易，说明见下。需要注意，程序从一个 80 字符长的内部文件中读取自由格式的输入，因此一行不能超过 80 字符。

还可以使用 H 或 h 格式。这对应高精度格式 (4F20.8)，其中第一列仍为轨道指数保留，后三列指定收缩因子。没有给出数值的列假定因子都是 0。若一个数据区内有 3 个以上的收缩函数，收缩因子可以续行，但轨道指数列留空。

NUCIJ

本数据区中的原高斯函数的个数。

NRCIJ

本数据区中收缩高斯函数的个数。如果设为 0，则假定是非收缩基组，只需要给出轨道指数。

ISGEN

指定如何通过动能平衡产生小分量函数。

ISGEN = 1

向上生成小分量函数，例如 $p \rightarrow d$ 。

ISGEN = 2

向下生成小分量函数，例如 $p \rightarrow s$ 。

ISGEN = 0, ISGEN = 3

同时向上和向下生成小分量函数，例如 $p \rightarrow s, d$ 。

其它值

不产生小分量函数。

只有在用动能平衡产生小分量函数时，才会用到这一信息。如果没有给出数值，则假定 **ISGEN = 0**。用动能平衡产生小分量基组的程序将删除重复的函数。

10-19, 22-26, 29-30

读取的指数（第一列）和收缩因子。非收缩基组只读取指数。格式由上面的 **FRMT** 给出。

4.3.3 MOLFDIR 型基组

```
1:DIRAC
2:Water
3:using MOLFDIR basis for large component
4:C 2 2 X Y
5: 8. 1
6:0 .0000000000 0.0000000000 -.2249058930
7:LARGE MOLFBAS Oxygen-xyz. bas
```

7 BSET, BSKEYWORD, BASFIL (A5, 1X, A7, 1X, A)

BSET

必须是 LARGE。

BSKEYWORD

对于 MOLFDIR 型基组，必须是 MOLFBAS。

BASFIL

基组文件名。这个文件必须复制到草稿文件区，例如使用 pam 脚本：

```
pam ... -copy H.bas ...
```

4.3.4 平缓 (even-tempered) 基组 (几何级数)

指数由平缓数列产生：

$$\eta_{N-k+1} = \alpha\beta^{N-1}, \quad k=1,\dots,N \quad (4.1)$$

```
1:DIRAC
2:Water
3:with even tempered basis set
4:C 2 2 X Y
5: 8. 1
6:0 .0000000000 0.0000000000 -.2249058930
7:LARGE EVENTEMP 0.05 3.0 11 3 2 1 1
8:1..5
9:6..11
10:7..11
11:9..10
```

7 BSET,BSKEYWORD,ALPHA,BETA,N,IQM,(JCO(K),K=1,IQM) (A5,1X,A8,1X,*)

BSET

必须是 LARGE。

BSKEYWORD

必须是 EVENTEMP 或 GEOM 平缓基组。

ALPHA

平缓数列中的参数 α 。

BETA

平缓数列中的参数 β 。

N

平缓数列中的参数 N 。

IQM

最高角量子数 l 加 1，例如 s 是 1， p 是 2，等。本例中是 3，因为我们对大分量使用 *spd* 基组。

JCO

对每个 l 值设定数据区的个数。在一个数据区中，所需内存随着基函数的数量迅速增长（例如 4 个 g 函数实际上有 60 个基函数，因为每个 g 函数有 15 个直角分量）。因此通过把某个量子数的基函数分割为不同的数据区，可以大大减少内存需求。但是这会降低积分计算的执行效率。

8-11 对每个数据区，给出用于该数据区的指数范围。在本例中，第一个区域 (s 函数) 由第 1 到第 5 个指数构成，第二个区域 (s 函数) 由第 6 到第 11 个指数构成，第三个区域 (p 函数) 由第 7 到第 11 个指数构成，最后一个区域 (d 函数) 由第 9

到第 10 个指数构成。

4.3.5 均匀 (well-tempered) 基组

指数由均匀数列产生:

$$\eta_N = \alpha \tag{4.2}$$

$$\eta_{N-k+1} = \eta_{N-k+2} \beta \left[1 + \gamma \left(\frac{k}{N} \right)^\delta \right], \quad k=1, \dots, N \tag{4.3}$$

```

1:DIRAC
2:Water
3:with well-tempered basis set
4:C 2 2 X Y
5: 8. 1
6:0 .0000000000 0.0000000000 -.2249058930
7:LARGE WELLTEMP 0.05 2.5 2.0 6.0 11 3 2 1 1
8:1..5
9:6..11
10:7..11
11:9..10

```

7 BSET, BSKEYWORD, ALPHA, BETA, GAMMA, DELTA, N, IQM, (JCO (K), K=1, IQM)
(A5, 1X, A8, 1X, *)

BSET

必须是 LARGE。

BSKEYWORD

对于均匀数列必须是 WELLTEMP。

ALPHA

均匀数列中的参数 α 。

BETA

均匀数列中的参数 β 。

GAMMA

均匀数列中的参数 γ 。

DELTA

均匀数列中的参数 δ 。

N

均匀数列中的参数 N 。

IQM

最高角量子数 l 加 1, 例如 s 是 1, p 是 2, 等。本例中是 3, 因为我们使用 *spd* 基组。

JCO

对每个 l 值设定数据区的个数。在一个数据区中，所需内存随着基函数的数量迅速增长（例如 4 个 g 函数实际上有 60 个基函数，因为每个 g 函数有 15 个直角分量）。因此通过把某个量子数的基函数分割为不同的数据区，可以大大减少内存需求。但是这会降低积分计算的执行效率。

8-11 对每个数据区，给出用于该数据区的指数范围。在本例中，第一个区域 (s 函数) 由第 1 到第 5 个指数构成，第二个区域 (s 函数) 由第 6 到第 11 个指数构成，第三个区域 (p 函数) 由第 7 到第 11 个指数构成，最后一个区域 (d 函数) 由第 9 到第 10 个指数构成。

4.3.6 同类基组

在输入的基组中，对所有的函数使用同一指数集。除了指数不从公式产生而必须在文件中给出之外，这类类似于均匀基组和平缓基组。这些指数可以来自用 GRASP [40] 优化的基组。

```

1:DIRAC
2:Water
3:with family basis set
4:C 2 2 X Y
5: 8. 1
6:0 .0000000000 0.0000000000 -.2249058930
7:LARGE FAMILY 10 3 1 1 1
8: 6665.0000000
9: 1000.0000000
10: 228.0000000
11: 64.7100000
12: 21.0600000
13: 7.4950000
14: 2.7970000
15: 0.5215000
16: 0.1596000
17: 0.0469000
18: 1..10
19: 6..10
20: 8..9

```

7 BSET, BSKEYWORD, N, IQM, (JCO(K), K=1, IQM) (A5, 1X, A6, 12I5)

BSET

必须是 LARGE。

BSKEYWORD

对于同类基组，必须是 FAMILY。

N

读入的指数个数。

IQM

最高角量子数 l 加 1，例如 s 是 1， p 是 2，等。本例中是 3，因为我们使用 spd

基组。

JCO

对每个 l 值设定数据区的个数。在一个数据区中，所需内存随着基函数的数量迅速增长（例如 4 个 g 函数实际上有 60 个基函数，因为每个 g 函数有 15 个直角分量）。因此通过把某个量子数的基函数分割为不同的数据区，可以大大减少内存需求。但是这会降低积分计算的执行效率。

8-17 读取 N 行指数。指数以自由格式读取。

18-20 对每个数据区，给出用于该数据区的指数范围。在本例中，第一个区域（ s 函数）由第 1 到第 10 个指数构成，第二个区域（ p 函数）由第 6 到第 10 个指数构成，最后一个区域（ d 函数）由第 8 到第 9 个指数构成。

4.3.7 双同类基组

除了一套指数用于所有的 $l=0,2,4,\dots$ (s, d, g, \dots) 基函数，另一套指数用于所有的 $l=1,3,5,\dots$ (p, f, h, \dots) 基函数以外，类似于同类基组。

```
1:DIRAC
2:Water
3:with dual family basis set
4:C 2 2 X Y
5: 8. 1
6:0 .0000000000 0.0000000000 -.2249058930
7:LARGE DUALFAMILY 10 5 3 1 1 1
8:# s, d, g, ... exponents
9: 6665.0000000
10: 1000.0000000
11: 228.0000000
12: 64.7100000
13: 21.0600000
14: 7.4950000
15: 2.7970000
16: 0.5215000
17: 0.1596000
18: 0.0469000
19:# p, f, h, ... exponents
20: 9.4390000
21: 2.0020000
22: 0.5456000
23: 0.1517000
24: 0.0404100
25:# ranges ...
26: 1..10
27: 1..5
28: 8..9
```

- 7 BSET, BSKEYWORD, N1, N2, IQM, (JCO(K), K=1, IQM) (A5, 1X, A10, 14I5)
BSET
 必须是 LARGE。
BSKEYWORD
 对于双同类基组，必须是 DUALFAMILY。
N1
 第一套指数数量。
N2
 第二套指数数量。
IQM
 最高角量子数 l 加 1，例如 s 是 1， p 是 2，等。本例中是 3，因为我们使用 *spd* 基组。
JCO
 对每个 l 值设定数据区的个数。在一个数据区中，所需内存随着基函数的数量迅速增长（例如 4 个 g 函数实际上有 60 个基函数，因为每个 g 函数有 15 个直角分量）。因此通过把某个量子数的基函数分割为不同的数据区，可以大大减少内存需求。但是这会降低积分计算的执行效率。
- 9-18 为 s, d, g, \dots 基函数读取 N1 行指数。指数以自由格式读取。
 20-24 为 p, f, h, \dots 基函数读取 N2 行指数。指数以自由格式读取。
 26-28 对每个数据区，给出用于该数据区的指数范围。在本例中，第一个区域 (s 函数) 由第一套指数的第 1 到第 10 个指数构成，第二个区域 (p 函数) 由第二套指数的第 1 到第 5 个指数构成，最后一个区域 (d 函数) 由第一套指数的第 8 到第 9 个指数构成。

4.4 小分量基组

在大分量基组之后，可以定义小分量基组。如果没有定义，等价于指定 SMALL KINBAL (见后)。有三种指定小分量基组的方法：

4.4.1 利用动能平衡

```

1:DIRAC
2:Water
3:aug-cc-pVDZ basis (note that a tight p has NOT been added)
4:C 2 2 X Y
5: 8. 1
6:0 .0000000000 0.0000000000 -.2249058930
...
29: 1.1850000 1.00000000 0.00000000 0.00000000
30: 0.3320000 0.00000000 1.00000000 0.00000000

```

```
31:SMALL KINBAL
```

- 31 BSET, BSKEYWORD (A5, 1X, A6)
若使用动能平衡，必须是 SMALL 和 KINBAL。
小分量基组由动能平衡产生。小分量基组总是非收缩的，与大分量基组是否收缩无关。

4.4.2 直接输入基组

```
1:DIRAC
2:Water
3:aug-cc-pVDZ basis (note that a tight p has NOT been added)
4:C 2 2 X Y
5: 8. 1
6:0 .0000000000 0.0000000000 -.2249058930
...
29: 1.1850000 1.00000000 0.00000000 0.00000000
30: 0.3320000 0.00000000 1.00000000 0.00000000
31:SMALL INTGRL 3 1 1 1
32:...
33:...
```

- 31 BSET, BSKEYWORD, IQM, (JCO(K), K=1, IQM) (A5, 1X, A6, 12I5)
同上面 LARGE INTGRL ... 的说明。**BSKEYWORD** 可以是 INTGRL, EXPLICIT, 或什么都不写。

4.4.3 MOLFDIR 型基组

```
1:DIRAC
2:Water
3:with MOLFDIR basis set for both large and small components
4:C 2 2 X Y
5: 8. 1
6:0 .0000000000 0.0000000000 -.2249058930
7:LARGE MOLFBAS Oxygen-xyz. bas
8:SMALL MOLFBAS Oxygen-xyz. bas
```

- 31 BSET, BSKEYWORD, BASFIL (A5, 1X, A7, 1X, A)
同上面 LARGE MOLFBAS ... 的说明。注意：使用 MOLFDIR 型小分量基组对菜单输入文件设置有特殊的要求。见附录 B 中 9. contraction 的说明。

第五章 输出文件

格式化的输出文件通过标准输出连接到 DIRAC。通过在菜单文件中设定各类打印级别，用户可以修改输出。DIRAC 将另外产生一个包含 SCF 过程的格式化文件 DFCYCL。除了格式化文件以外，DIRAC 还产生大量的无格式文件。它们是：

- **控制文件**

DFDIIS DIIS 过程的信息。
DFEVEC 包含 DIIS 误差矢量的直接访问文件。

- **轨道因子**

DFCOEF 来自当前 SCF 迭代的 MO 因子。

- **单电子积分和矩阵**

DF1INT 单电子积分对单电子 Fock 矩阵的贡献。
DFOVLP 重叠矩阵。
DFTMAT MO-变换矩阵。
DFFCK1 单电子 Fock 矩阵（在 QO 基中）。

- **双电子积分和矩阵**

对于按照习惯模式的每一类积分($XX = LL, SL, SS$):

DFXXSA 分类单重积分（库仑和交换贡献）。
DFXXSB 分类单重积分（库仑和交换贡献）。
DFXXTA 分类三重积分（仅交换贡献）。
DFXXTB 分类三重积分（仅交换贡献）。
DFTWXX 来自 HERMIT 的未分类积分的草稿文件。
DFXXTB 用于分类过程的草稿文件。

此外还有：

DFFCK2 QO 基中的双电子 Fock 矩阵。

5.1 重新开始计算

DIRAC 具有某些重新开始计算的能力：

- 当计算势能曲面上一个**新的点**时，DIRAC 可以从因子（DFCOEF 文件）开始，从 AO 基中的双电子 Fock 矩阵（DFFCK2 文件）开始，或从单电子 Fock 矩阵的解（裸核近似）开始。如果出现无格式的矢量文件 DFCOEF，默认从因子开始，否则 DIRAC 使用裸核近似。重新从 Fock 矩阵开始计算，可以用关键字 TRIFCK 指定。
- 当在势能曲面上**同一点**重新开始计算时，DIRAC 需要格式化的文件 DFCYCL 来升级 SCF 过程的状态。完整的 SCF 总结在当前迭代结束后给出，所以前一个 SCF 迭代的输出文件一般并不需要。此外 DIRAC 还需要因子（文件 DFCOEF）。要重新开始 DIIS，DIRAC 需要以下文件：DFDIIS, DFCMOS, DFFOCK 和 DFEVEC。如果不需要 DIIS，且存在 DFFOCK，DIRAC 可以重新开始阻尼计算。

第六章 定义单电子算符

6.1 定义算符的语法

在四分量计算中，一般特性算符由基本形式的线性组合产生：

$$a_i M_i \Omega_j \quad (6.1)$$

其中， M_i 是以下 4×4 矩阵之一：

$$\begin{aligned} M_i = & I_4, \gamma_5, \\ & \alpha_x, \alpha_y, \alpha_z, \\ & \Sigma_x, \Sigma_y, \Sigma_z, \\ & \beta \Sigma_x, \beta \Sigma_y, \beta \Sigma_z, \\ & i\beta \alpha_x, i\beta \alpha_y, i\beta \alpha_z \end{aligned} \quad (6.2)$$

它由表示一般形式的关键字，通过指定标量算符 Ω_j 和因子 α_i 来说明：

	关键字	算符形式	因子
1	DIAGONAL	$a_1 I_4 \Omega$	1
2	XALPHA	$a_1 \alpha_x \Omega$	1
3	YALPHA	$a_1 \alpha_y \Omega$	1
4	ZALPHA	$a_1 \alpha_z \Omega$	1
5	XAVECTOR	$a_1 \alpha_y \Omega_z - a_2 \alpha_z \Omega_y$	2
6	YAVECTOR	$a_1 \alpha_z \Omega_x - a_2 \alpha_x \Omega_z$	2
7	ZAVECTOR	$a_1 \alpha_x \Omega_y - a_2 \alpha_y \Omega_x$	2
8	ALPHADOT	$a_1 \alpha_x \Omega_x + a_2 \alpha_y \Omega_y + a_3 \alpha_z \Omega_z$	3
9	GAMMA5	$a_1 \gamma_5 \Omega$	1
10	XSIGMA	$a_1 \Sigma_x \Omega$	1
11	YSIGMA	$a_1 \Sigma_y \Omega$	1
12	ZSIGMA	$a_1 \Sigma_z \Omega$	1

13	XBETASIG	$a_1\beta\Sigma_x\Omega$	1
14	YBETASIG	$a_1\beta\Sigma_y\Omega$	1
15	ZBETASIG	$a_1\beta\Sigma_z\Omega$	1
16	XiBETAAL	$a_i\beta\alpha_x\Omega$	1
17	YiBETAAL	$a_i\beta\alpha_y\Omega$	1
18	ZiBETAAL	$a_i\beta\alpha_z\Omega$	1

算符形式 5-7 构成 $(\alpha \times \Omega)$ 的分量, 算符形式 8 是 $(\alpha \cdot \Omega)$, 算符 9 是 $\gamma_5 = \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}$ 矩阵。

因子的定义为:

FACTORS 定义因子 $\{a_i\}$

COMFACTO 所有分量的共用因子

程序假定所有的算符都是厄密共轭的, 因此如果需要的话, 将插入一个虚相位 i (用于反对称的标量算符)。

算符用关键字. OPERATOR 和以下参量定义:

. OPERATOR

<操作名称>

<操作类型>

<每个分量的标号>

FACTORS

<每个分量的因子>

COMFACTOR

<所有分量的共用因子>

注意关键字. OPERATOR 之后接的参量必须由一个空格开始。除了算符的标号以外, 参量是可选的。如果没有给出其它参量, 程序假定算符为对角算符, 并认为算符的名称就是分量的标号。例如:

. OPERATOR

OVERLAP

作为例子, Dirac 哈密顿量的动能部分可以定义为:

. OPERATOR

'Kinetic energy'

ALPHAD

XDIPVEL

YDIPVEL

ZDIPVEL

COMFACTOR

-137.0359895

另一个例子是把偶极长度算符 \hat{z} 添加到哈密顿量 $H = H_0 + 0.01 \cdot \hat{z}$ 中的有限场计算:

```
. OPERATOR
ZDIPLN
COMFACTOR
0. 01
```

6.2 单电子算符列表

MOLFLD	核吸引积分 <ul style="list-style-type: none"> ● 对称的 ● 有一个分量: $\text{MOLFIELD } \Omega_1 = \sum_I \hat{V}_{en}$ <p>点原子核: 有限原子核:</p>
BETAMT	只有 SS 区域的重叠积分 <ul style="list-style-type: none"> ● 对称的 ● 有一个分量: <p>BETAMAT $\Omega_1=1$; 只有 SS 区域</p>
OVERLAP	重叠积分 <ul style="list-style-type: none"> ● 对称的 ● 有一个分量: <p>OVERLAP $\Omega_1=1$</p>
DIPLN	偶极长度积分 <ul style="list-style-type: none"> ● 对称的 ● 有三个分量: <p>XDIPLN $\Omega_1=x$ YDIPLN $\Omega_2=y$ ZDIPLN $\Omega_3=z$</p>
DIPVEL	偶极速率积分 <ul style="list-style-type: none"> ● 反对称的 ● 有三个分量: <p>XDIPVEL $\Omega_1 = \frac{\partial}{\partial x}$ YDIPVEL $\Omega_2 = \frac{\partial}{\partial y}$ ZDIPVEL $\Omega_3 = \frac{\partial}{\partial z}$</p>
QUADRUP	四极矩积分 <ul style="list-style-type: none"> ● 对称的积分 ● 有六个分量:

$$\text{XXQUADRU} \quad \Omega_1 = \frac{\partial^2}{\partial x \partial x}$$

$$\text{XYQUADRU} \quad \Omega_2 = \frac{\partial^2}{\partial x \partial y}$$

$$\text{XZQUADRU} \quad \Omega_3 = \frac{\partial^2}{\partial x \partial z}$$

$$\text{YYQUADRU} \quad \Omega_4 = \frac{\partial^2}{\partial y \partial y}$$

$$\text{YZQUADRU} \quad \Omega_5 = \frac{\partial^2}{\partial y \partial z}$$

$$\text{ZZQUADRU} \quad \Omega_6 = \frac{\partial^2}{\partial z \partial z}$$

SPNORB

空间自旋-轨道积分

- 反对称的积分
- 有三个分量:

$$\text{X1SPNORB} \quad \Omega_1 = \sum_A \frac{l_x}{r_{iA}^3}$$

$$\text{Y1SPNORB} \quad \Omega_2 = \sum_A \frac{l_y}{r_{iA}^3}$$

$$\text{Z1SPNORB} \quad \Omega_3 = \sum_A \frac{l_z}{r_{iA}^3}$$

SECMOM

二阶矩积分

- 对称的积分
- 有六个分量:

$$\text{XXSECMOM} \quad \Omega_1 = x^2$$

$$\text{XYSECMOM} \quad \Omega_2 = xy$$

$$\text{XZSECMOM} \quad \Omega_3 = xz$$

$$\text{YYSECMOM} \quad \Omega_4 = y^2$$

$$\text{YZSECMOM} \quad \Omega_5 = yz$$

$$\text{ZZSECMOM} \quad \Omega_6 = z^2$$

THETA
CARMOM

无迹的 θ 四极积分

笛卡尔坐标的力矩积分

- 对称的积分
- 有 $(l+1)(l+2)/2$ 个分量 ($l = i + j + k$):

$$\text{CM} i i j j k k \quad \Omega_l = x^i y^j z^k$$

SPHMOM	球坐标的力矩积分（实部）
	<ul style="list-style-type: none"> ● 对称的积分 ● 有 $2l+1$ 个分量 ($m = +0, -1, +1, \dots, +l$): $SM_{l \pm m} \quad \Omega_l = R_{l \pm m}$
SOLVENT	电子解的积分
FERMI C	单电子 Fermi 接触积分
PSO	顺磁性自旋-轨道积分
SPIN-DI	自旋偶极积分
DSO	反磁性自旋-轨道积分
SDFC	自旋偶极 + Fermi 接触积分
HDO	半导出重叠积分
S1MAG	重叠矩阵对磁性的二阶贡献
S2MAG	重叠矩阵对磁性的贡献
ANGLON	围绕原子核的角动量
ANGMOM	围绕原点的电子角动量
LONMOM	对角动量的 London 轨道贡献
MAGMOM	对磁矩的单电子贡献
KINENER	电子动能
DSUSNOL	没有 London 贡献的反磁性磁化系数
DSUSLAN	对反磁性磁化系数的角 London 轨道贡献
DSUSLH	对反磁性磁化系数的角 London 轨道贡献
DIASUS	对反磁性磁化系数的角 London 轨道贡献
NUCSNLO	没有 London 轨道贡献的核屏蔽积分
NUCSLO	对核屏蔽张量积分的 London 轨道贡献
NUCSHI	核屏蔽张量积分
NEFIELD	单个核的电场
ELFGRDC	单个核的笛卡尔坐标电场梯度
ELFGRDS	单个核的球坐标电场梯度
S1MAGL	磁场重叠矩阵的左矢微分
S1MAGR	磁场重叠矩阵的右矢微分
HDOBR	磁场 HDO 积分的右矢微分
NUCPOT	原子核的势能
HBDO	半 B 微分重叠矩阵
SQHDO	不进行反对称化的半导出重叠积分
DSUSCGO	具有共同尺度原点的反磁性磁化系数
NSTCGO	具有共同尺度原点的核屏蔽积分
EXPIKR	余弦和正弦积分
MASSVEL	质量速率积分
DARWIN	Darwin 型积分
CM1	电场的一阶磁场导数
CM2	电场的二阶磁场导数
SQHDOR	不进行反对称化的半导出重叠积分
SQOVLAP	不进行反对称化的重叠积分

附录 A. 最近的修正

1998 年 11 月 1 日

Dirac 3.1 发布版

2000 年 10 月 6 日

Dirac 3.2 发布版

2004 年 9 月 1 日

DIRAC 04.0 发布版

附录 B. 示例文件

DIRAC 04的全部例子在test目录下，包括输入和输出文件。

子目录	说明
1. energy+dipole	在不同理论级别上计算水分子的能量和偶极矩。注意示例计算中总是忽略SS积分。 这个小例子仅用于测试的目的：忽略SL积分并且不进行任何补偿，以及对虚空间做了过多的截断，都不是好的近似。
2. nmrspinpin	这是水分子NMR自旋-自旋耦合的计算。使用了无电子对方法，因此通过Sternheimer近似计算DSO项。 注意我们总是忽略SS积分。
3. geomopt. dhf	对不同的分子做几何优化。还对最终的结构计算了一些特性。
4. levy-leblond. nmr	用Levy-Leblond方法计算非相对论极限的特性。我们把具有转置对称性的H ₂ 作为例子。注意由于公式中的g=2，使得结果稍微偏离DALTON的计算结果。
5. cisd. energy	用CISD波函，计算氟 ² P态的自旋-轨道分量。
6. cosci. energy	用COSCI波函，计算氟 ² P态的自旋-轨道分量。
7. directmp2. energy	在DHF之后用MP2方法计算水分子的总能量。
8. spinfree. cosci. energy	用无自旋的方法计算氟的 ² P态。
9. contraction	用收缩基组计算水分子。 <ol style="list-style-type: none">1. 使用DIRAC格式的大、小分量基组。2. 使用从MOLFDIR基组文件读取的基组。 需要保证这些基组位于执行计算的目录中。注意收缩的MOLFDIR型基组只能用在URKBAL (**HAMILTONIAN)中，并定义.SPHTRA 1 0 (**GANERAL)。用其它选项会给出错误的结果。
10. efg	用小基组计算HF分子的EFG有限场。 运行零电场用来产生MO矢量，并给出DHF和MP2级别的EFG期待值。有限场方法用来计算CCSD(T)关联的贡献，它可以添加到解析的HF值上。因此我们使用松弛轨道，因为这会给出更严格的结果。 结果（电子贡献）： DHF解析计算（通过property模块作为期待值计算） 2.728 au DHF有限差分计算（通过有限场） 2.727 au

关联的贡献	
MP2解析计算 (通过RELCCSD中新的模块CCFOPR)	-0.118 au
MP2有限差分计算	-0.118 au
CCSD有限差分计算	-0.056 au
CCSD(T)有限差分计算	-0.057 au
总CCSD(T)值 (加入核 + 解析HF + 有限差分关联的贡献)	3.056 au

注意:

- ✧ MP2方法把关联的贡献高估了两倍。
- ✧ 基组太小, 给出的结果不可靠 (这只是一个测试)。

11. methods. cosci

用COSCI波函在不同理论级别计算由氧原子 p^4 组态产生的电子态。

1. 使用Levy-Leblond选项的非相对论结果。
2. 使用spinfree选项的无自旋相对论结果。
3. 使用第二步的轨道计算微扰SO的结果。
4. 使用优化轨道的完全相对论结果。

结果 (激发能单位: eV):

	非相对论	无自旋	微扰SO	完全相对论
3P2	0.0000	0.0000	0.0000	0.0000
3P1	0.0000	0.0000	0.0196	0.0189
3P0	0.0000	0.0000	0.0293	0.0282
1D	2.1928	2.1923	2.2022	2.2019
1S	5.4820	5.4807	5.4907	5.4904
3P(平均)	0.0000	0.0000	0.0098	0.0094
1D(相对于平均值)	2.1928	2.1923	2.1924	2.1924
1S(相对于平均值)	5.4820	5.4807	5.4809	5.4809

12. zora

用两个测试体系, 计算ZORA的各种变体: U^{91+} ($2p$ 是空的) 使用Dyall基组, Xe原子使用Faas的小基组。

测试检验了ZORA和标量ZORA选项, 以及所谓的完全密度 (或ZORA4) 和近似密度ZORA方案。标度只影响输出轨道的能量。

对于多电子体系, ZORA和ZORA4的区别很大, 完全密度方案给出更准确的结果, 但是耗费的时间相当于通常的DIRAC计算。近似密度方案是(LL+SL)计算级别, 因此略微便宜些。

13. rpa

使用线性响应模块, 计算水分子各种频率的偶极极化率。

14. basis. set. input

用不同格式的各种基组计算 CH_4 分子。

17. dft. energy

DFT能量计算。

注意这一功能不会自动测试, 需要人工检查输出, 以保证DIRAC04正确安装。

19. fockspace
用Fock CC空间方法计算镁离子(2+)的电离能和电子亲和能。算法的说明见L. Visscher, E. Eliav, and U. Kaldor, *Journal of Chemical Physics*, 115, 9720-9726 (2001).
22. lucita_short
用FCI波函, 计算He原子基态(1S)的能量。
24. lucita_large
用CISD波函, 计算HBr分子基态($^1\Sigma^+$)的能量。
25. qrpa
LiH分子二阶响应的计算。
29. benchmark_cc
不同构型的1,2-二氯乙烷计算(用不同的点群对称性)。
30. linear
用CC理论计算氮分子的能量。注意我们总是忽略SS积分。
这个测试用于检查自动探测对称性, 以及执行线型对称性。我们假定在您的标准路径中有相关一致基组。这些基组可以从DALTON程序的发布版中得到, 以后也会包含在DIRAC的发布版中。
31. symmetry_recognition
用B3LYP泛函, 计算非相对论DFT能量。注意我们总是忽略SS积分。
这个测试用于检查自动探测对称性, 以及使用非相对论基组。我们假定在您的标准路径中有相关一致基组。这些基组可以从DALTON程序的发布版中得到, 以后也会包含在DIRAC的发布版中。
32. pr_jana
这个测试计算执行甲烷分子的投影分析。
包含了标准的穆利肯布居分析用于比较。对各种基组比较两种方法的原子电荷是一个有趣的练习。投影布居分析相当稳定, 而穆利肯布居分析有强烈的基组依赖。
此外注意, 片断不是在整个分子基中计算, 而是在它们自己的基组中计算。这用.OWNBAS关键字得到。
投影分析通常基于对称性无关的中心, 这意味着在 C_{2v} 对称性中, 水分子的片断是氧原子和H2片断。为了在真实的原子片断上进行投影分析, 必须降低对称性。可以先在最高的对称性下计算, 接下来在**GENERAL下使用ACMOUT关键字, 把因子转存为C对称性。
还要注意, 使用弱电场会破坏简并性, 因此会得到具有明确意义的 m_j 值的原子轨道。
33. frozen
这个测试计算对水分子冻结1s和2s轨道, 执行几何优化。水分子的键角为104.5度, 对于更重的同系物该角度倾向于90度。这被认为是由于价层s和p轨道之间逐渐增加的能量分裂阻碍了轨道杂化。在这个测试中我们冻结来自于前一个氧原子计算中的2s(和1s)轨道。这些轨道在氧的基中计算, 因此在几何优化的整个迭代中使用它们。目前只能执行数值计算。
在几何优化的最后, 键角并不是90度(但却接近)。这是由于氢原子间的Pauli排斥, Kutzelnigg在其经典论文中有解释: *Angew. Chem. Int. Ed. Engl.* 23 (1984) 272.

34. rhonuc

在一氧化碳的实验键长执行B3LYP计算，并用期待值模块计算核上的电子密度。这意味着期待值可以用各种工具进行分析。

36. mp2_properties

对于各种特性的MP2一阶特性计算。计算的特性有：

- ◇ PVC: 宇称不守恒（手性）能量
- ◇ EFG: 电场梯度
- ◇ DIPOLE: 偶极矩
- ◇ EFN: 核上的电场
- ◇ NQCC: 核四极偶合常数

注意：结构参数与在Chem. Phys上关于MP2一阶特性的论文中所用的相同，二面角是45度。

参考文献

- [1] L. Visscher and K. G. Dyall. Dirac-fock atomic electronic structure calculations using different nuclear charge distributions. *At. Data Nucl. Data Tabl.*, 67:207, 1997.
- [2] T. Saue, K. Fægri, T. Helgaker, and O. Gropen. Principles of direct 4-component relativistic scf: Application to cesium auride. *Mol. Phys.*, 91:937, 1997.
- [3] L. Visscher and T. Saue. Approximate relativistic electronic structure methods based on the quaternion modified dirac equation. *J. Chem. Phys.*, 113:3996, 2000.
- [4] J. M. Lévy-Leblond. Nonrelativistic particles and wave equations. *Commun. Math. Phys.*, 6:286, 1967.
- [5] K. G. Dyall. An exact separation of the spin-free and spin-dependent terms of the dirac-coulomb-breit hamiltonian. *J. Chem. Phys.*, 100:2118, 1994.
- [6] L. Visscher. Approximate molecular dirac-coulomb calculations using a simple coulombic correction. *Theor. Chem. Acc.*, 96:68–70, 1997.
- [7] P. A. M. Dirac. Note on exchange phenomena in the thomas atom. *Proc. Roy. Soc. London*, 26:376, 1930.
- [8] A. D. Becke. Density-functional exchange-energy approximation with correct asymptotic behavior. *Phys. Rev. A*, 38:3098, 1988.
- [9] J. P. Perdew and Y. Wang. Accurate and simple density functional for the electronic exchange energy: Generalized gradient approximation. *Phys. Rev. B*, 33:8800, 1986.
- [10] S. J. Vosko, L. Wilk, and M. Nusair. Accurate spin-dependent electron liquid correlation energies for local spin density calculations: A critical analysis. *Can. J. Phys.*, 58:1200, 1980.
- [11] C. Lee, W. Yang, and R. G. Parr. Development of the colle-salvetti correlation-energy formula into a functional of the electron density. *Phys. Rev. B*, 37:758, 1988.
- [12] J. P. Perdew. Density-functional approximation for the correlation energy of the inhomogenous electron gas. *Phys. Rev. B*, 33:8822, 1986.
- [13] E. van Lenthe, E. J. Baerends, and J. G. Snijders. Relativistic total energies using regular approximations. *J. Chem. Phys.*, 101:9783, 1994.
- [14] E. van Lenthe, J. G. Snijders, and E. J. Baerends. The zero-order regular approximation for relativistic effects: The effect of spin-orbit coupling in closed shell molecules. *J. Chem. Phys.*, 105:6505, 1996.
- [15] T. Saue and T. Helgaker. Four-component relativistic kohn–sham theory. *J. Comput. Chem.*, 23:814, 2002.
- [16] O. Fossgaard, O. Gropen, M. Corral Valero, and T. Saue. On the performance of four-component relativistic density functional theory: Spectroscopic constants and dipole moments of the diatomics hx and xy (x,y=f, cl, br and i). *J. Chem. Phys.*, 118:10418, 2003.
- [17] M. Mayer, O. D. Häberlen, and N. Rösch. *Phys. Rev. A*, 54:4775, 1996.
- [18] S. Varga, E. Engel, W.-D. Sepp, and B. Fricke. *Phys. Rev. A*, 59:4288, 1999.
- [19] S. Varga, B. Fricke, H. Nakaamatsu, T. Mukoyama, J. Anton, D. Geschke, A. Heitmann, and E. Engel et T. Bastug. *J. Chem. Phys.*, 112:3499, 2000.
- [20] A. D. Becke. A multicenter numerical integration scheme for polyatomic molecules. *J. Chem. Phys.*, 88:2547, 1988.

- [21] R. Lindh, P. A. Malmqvist, and L. Galgiardi. Molecular integrals by numerical quadrature. i. radial integration. *Theor. Chem. Acc.*, 106:178, 2001.
- [22] T. Saue and H. J. Aa Jensen. Quaternion symmetry in relativistic molecular calculations: I. the dirac-fock method. *J. Chem. Phys.*, 111:6211, 1999.
- [23] T. Saue and H. J. Aa. Jensen. Quaternion symmetry of the dirac equation. In M. Defranceschi and C. Le Bris, editors, *Mathematical Methods for Ab Initio Quantum Chemistry*, volume 74 of *Lecture Notes in Chemistry*, page 227. Springer, Berlin, 2000.
- [24] J. Thyssen and H. J. Aa. Jensen. Average-of-configurations SCF manuscript, 1998.
- [25] P. Pulay. Convergence acceleration of iterative sequences. the case of scf iteration. *Chem. Phys. Lett.*, 73:393, 1980.
- [26] P. Pulay. Improved scf acceleration. *J. Comput. Chem.*, 3:556, 1982.
- [27] T. Hamilton and P. Pulay. Direct inversion in the iterative subspace(diis) optimization of open-shell, excited-state, and small multiconfiguration scf wave functions. *J. Chem. Phys.*, 84:5728, 1986.
- [28] W. Kutzelnigg. Chemical bonding in the higher main group elements. *Angew. Chem. Int. Ed. Engl.*, 23:272, 1984.
- [29] C. W. Bauschlicher. *J. Chem. Phys.*, 72:880, 1980.
- [30] K. Faegri and T. Saue. Diatomic molecules between very heavy elements of group 13 and group 17: A study of relativistic effects on bonding. *J. Chem. Phys.*, 115:2456, 2001.
- [31] J. K. Laerdahl and P. Schwerdtfeger. Fully relativistic ab initio calculations of the energies of chiral molecules including parity-violating weak interactions. *Phys. Rev. A*, 60:4439, 1999.
- [32] T. Saue and H. J. Aa. Jensen. Linear response at the 4-component relativistic level: Application to the frequency-dependent dipole polarizabilities of the coinage metal dimers. *J. Chem. Phys.*, 118:522, 2003.
- [33] T. Saue. Post dirac-hartree-fock methods - properties. In P. Schwerdtfeger, editor, *Relativistic Electronic Structure Theory - Part 1 : Fundamentals*. Elsevier, Amsterdam, 2002.
- [34] L. Visscher. The dirac equation in quantum chemistry : strategies to overcome the current computational problems. *J. Comput. Chem.*, 23:759, 2002.
- [35] L. Visscher, T. J. Lee, and K. G. Dyall. Formulation and implementation of a relativistic unrestricted coupled cluster method including noniterative connected triples. *J. Chem. Phys.*, 105:8769, 1996.
- [36] M. Pernpointner and L. Visscher. Parallelization of four-component calculations. ii. symmetry- driven parallelization of the 4-spinor ccsd algorithm. *J. Comput. Chem.*, 24:754, 2003.
- [37] L. Visscher, E. Eliav, and U. Kaldor. Formulation and implementation of the relativistic fock-space coupled cluster method for molecules. *J. Chem. Phys.*, 115:9720, 2001.
- [38] L. Visscher, O. Visser, H. Aerts, H. Merenga, and W. C Nieuwpoort. Relativistic quantum chemistry: the molmdir program package. *Comput. Phys. Comm.*, 81:120, 1994.
- [39] T. Fleig and L. Visscher. Large-Scale Electron Correlation Calculations in the Framework of the Spin-Free Dirac Formalism. The Au₂ Molecule Revisited. *Chem. Phys.*, 2004.
- [40] K. G. Dyall, I. P. Grant, C. T. Johnson, F. A. Parpia, and E. P. Plummer. Grasp: A general-purpose relativistic atomic structure program. *Comput. Phys. Comm.*, 55:425, 1989.