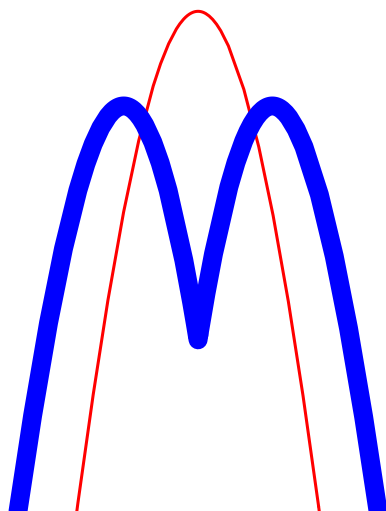


# MOLPRO 2009.1 用户手册



Zork 译

[zorkzou@yahoo.com.cn](mailto:zorkzou@yahoo.com.cn)

修改日期: 2009 年 11 月 5 日



# MOLPRO User's Manual

## Version 2009.1

**H.-J. Werner and P. J. Knowles**

2009



# 目 录

目录	i
I. MOLPRO简介	xxv
II. 互联网上的MOLPRO	xxxix
III. 引用	xxxiii
第一章 如何阅读手册	1
第二章 运行MOLPRO	3
2.1 选项	3
2.2 在并行计算机上运行MOLPRO	6
2.2.1 定义并行的执行	6
第三章 定义MOLPRO输入语言	9
3.1 输入格式	9
3.2 命令	10
3.3 指令	11
3.4 全局指令	12
3.5 选项	12
3.6 数据	13
3.7 表达式	13
3.8 内部函数	14
3.9 变量	15
3.9.1 设定变量	15

3.9.2	字符串变量	15
3.10	流程	16
3.10.1	流程的定义	16
3.10.2	流程的调用	17
<b>第四章</b>	<b>一般程序结构</b>	<b>19</b>
4.1	输入结构	19
4.2	文件	20
4.3	记录	20
4.4	续算	21
4.5	数据集的操作	21
4.6	内存分配	22
4.7	通过输入做循环	22
4.8	对称性	22
4.9	定义波函	24
4.10	定义轨道子空间	24
4.11	选择轨道和密度矩阵(ORBITAL, DENSITY)	25
4.12	控制程序的关键词总结	27
4.13	MOLPRO帮助	31
<b>第五章</b>	<b>示范的例子</b>	<b>33</b>
5.1	使用molpro命令	33
5.2	简单的SCF计算	34
5.3	几何优化	34
5.4	CCSD(T)	34
5.5	CASSCF和MRCI	35
5.6	制表	35
5.7	流程	36
5.7.1	SA-CASSCF和MRCI	37
5.7.2	MP2几何优化	39
5.7.3	DFT频率计算	40
5.8	D0循环	41

<b>第六章 程序控制</b>	<b>45</b>
6.1 开始一个任务 (***)	45
6.2 结束一个任务 (---)	45
6.3 续算一个任务 (RESTART)	45
6.4 包含辅助输入文件 (INCLUDE)	46
6.5 分配动态内存 (MEMORY)	46
6.6 DO循环 (DO/ENDDO)	46
6.6.1 do循环的例子	47
6.7 分支 (IF/ELSEIF/ENDIF)	47
6.7.1 IF声明	47
6.7.2 GOTO命令	48
6.7.3 标记 (LABEL)	48
6.8 流程 (PROC/ENDPROC)	49
6.9 文本卡 (TEXT)	51
6.10 检查程序状态 (STATUS)	51
6.11 全局阈值 (GTHRESH)	52
6.12 全局打印选项 (GPRINT/NOGPRINT)	53
6.13 单电子算符和期待值 (GEXPEC)	54
6.13.1 计算期待值的例子	55
6.13.2 计算相对论校正的例子	56
<b>第七章 文件操作</b>	<b>59</b>
7.1 FILE	59
7.2 DELETE	60
7.3 ERASE	60
7.4 DATA	60
7.5 分配punch文件 (PUNCH)	61
7.6 MOLPRO系统参数 (GPARAM)	61
<b>第八章 变量</b>	<b>63</b>
8.1 设置变量	63
8.2 加索引的变量	64
8.3 字符串变量	64
8.4 系统变量	65

8.5	使用字符串变量的宏定义	66
8.6	加索引的变量（矢量）	68
8.7	矢量操作	69
8.8	特殊变量	70
8.8.1	程序设置的变量	70
8.8.2	程序认识的变量	73
8.9	显示变量	76
8.9.1	SHOW命令	76
8.10	清除变量	77
8.11	从外部文件读取变量	77
<b>第九章</b>	<b>制表与画图</b>	<b>79</b>
9.1	制表	79
9.2	画图	81
9.3	双原子分子势能曲线分析	82
<b>第十章</b>	<b>结构定义与积分</b>	<b>83</b>
10.1	结构说明	83
10.1.1	Z矩阵输入	84
10.1.2	XYZ输入	85
10.2	对称性说明	86
10.3	写Gaussian, Xmol或Molden的输入（PUT）	87
10.3.1	用Molden显示结果	87
10.4	结构文件	88
10.5	点电荷的格子	89
10.6	重新定义和打印原子质量	89
10.7	虚原子中心	90
10.7.1	平衡计算	90
10.7.2	示例：OH-Ar的相互作用能	90
<b>第十一章</b>	<b>基组输入</b>	<b>93</b>
11.1	概述：基组和基组库	93
11.2	笛卡尔和球谐基函数	93
11.3	基组库	94
11.4	默认基组	95



11.5 不同原子的默认基组 . . . . .	96
11.6 原始集的定义 . . . . .	99
11.7 收缩集的定义 . . . . .	101
11.8 例子 . . . . .	101
<b>第十二章 有效芯势 . . . . .</b>	<b>103</b>
12.1 来自ECP库的输入 . . . . .	103
12.2 直接输入ECP . . . . .	104
12.3 直接输入ECP的例子 . . . . .	105
12.4 从基组库输入ECP的例子 . . . . .	105
<b>第十三章 芯极化势 . . . . .</b>	<b>107</b>
13.1 输入选项 . . . . .	107
13.2 ECP/CPP的例子 . . . . .	108
<b>第十四章 积分 . . . . .</b>	<b>109</b>
14.1 分类的积分 . . . . .	109
14.2 直接积分计算 (GDIRECT) . . . . .	110
14.2.1 直接积分计算的例子 . . . . .	119
<b>第十五章 密度拟合 . . . . .</b>	<b>121</b>
15.1 密度拟合的选项 . . . . .	122
15.1.1 选择拟合基组的选项 . . . . .	122
15.1.2 屏蔽阈值 . . . . .	123
15.1.3 局域拟合参数 . . . . .	123
15.1.4 拟合域的参数 . . . . .	124
15.1.5 各种控制选项 . . . . .	126
<b>第十六章 SCF程序 . . . . .</b>	<b>127</b>
16.1 选项 . . . . .	128
16.1.1 控制HF收敛的选项 . . . . .	128
16.1.2 对角化方法的选项 . . . . .	129
16.1.3 收敛加速方法 (DIIS) 的选项 . . . . .	129
16.1.4 积分直接计算的选项 . . . . .	129
16.1.5 UHF计算的特殊选项 . . . . .	130
16.1.6 局域密度拟合计算选项 . . . . .	130

16.1.7 极化率的选项 . . . . .	130
16.1.8 打印选项 . . . . .	131
16.2 定义波函 . . . . .	131
16.2.1 定义每个对称性的占据轨道数 . . . . .	131
16.2.2 定义闭壳层轨道 . . . . .	131
16.2.3 指定开壳层轨道 . . . . .	132
16.3 保存最终轨道 . . . . .	132
16.4 初始轨道 . . . . .	132
16.4.1 初始轨道猜测 . . . . .	133
16.4.2 由前一轨道开始 . . . . .	134
16.4.3 由前一密度矩阵开始 . . . . .	135
16.5 转动轨道对 . . . . .	135
16.6 使用附加的点群对称性 . . . . .	135
16.7 期待值 . . . . .	136
16.8 极化率 . . . . .	136
16.9 各种指令 . . . . .	136
16.9.1 能级移动 . . . . .	136
16.9.2 迭代的最大数量 . . . . .	136
16.9.3 收敛阈值 . . . . .	136
16.9.4 能量的合理性检查 . . . . .	137
16.9.5 打印选项 . . . . .	137
16.9.6 内插 . . . . .	137
16.9.7 轨道的重新正交归一化 . . . . .	137
16.9.8 直接SCF . . . . .	137
<b>第十七章 密度泛函程序</b>	<b>139</b>
17.1 选项 . . . . .	140
17.2 指令 . . . . .	140
17.2.1 密度来源 (DENSITY, ODDENSITY) . . . . .	140
17.2.2 阈值 (DFTTHRESH) . . . . .	140
17.2.3 精确交换计算 (EXCHANGE) . . . . .	141
17.2.4 区域杂化方法 (RANGEHYBRID) . . . . .	141
17.2.5 交换-关联势 (POTENTIAL) . . . . .	141
17.2.6 网格分块因子 (DFTBLOCK) . . . . .	142

17.2.7 转存积分值 (DFTDUMP)	142
17.3 数值积分网格控制 (GRID)	142
17.3.1 目标积分精度 (GRIDTHRESH)	143
17.3.2 径向积分网格 (RADIAL)	143
17.3.3 角向积分网格 (ANGULAR)	144
17.3.4 积分网格的原子分区 (VORONOI)	145
17.3.5 网格的缓存 (GRIDSAVE, NOGRIDSAVE)	145
17.3.6 网格对称性 (GRIDSYM, NOGRIDSYM)	145
17.3.7 网格打印 (GRIDPRINT)	145
17.4 密度泛函	145
17.4.1 泛函的别名	147
17.4.2 执行新的泛函	148
17.5 经验阻尼色散校正	150
17.6 例子	151
<b>第十八章 轨道局域化</b>	<b>153</b>
18.1 定义输入轨道 (ORBITAL)	153
18.2 保存局域化轨道 (SAVE)	154
18.3 选择局域化方法 (METHOD)	154
18.4 轨道的离域化 (DELOCAL)	154
18.5 局域化AO (LOCAO)	154
18.6 选择轨道空间	154
18.6.1 定义占据空间 (OCC)	154
18.6.2 定义芯轨道 (CORE)	154
18.6.3 定义轨道组 (GROUP, OFFDIAG)	155
18.6.4 组之间的局域化 (OFFDIAG)	155
18.7 局域化轨道的排序	155
18.7.1 不重新排序 (NOORDER)	155
18.7.2 使用域进行排列 (SORT)	156
18.7.3 定义参考轨道 (REFORB)	156
18.7.4 选择Fock矩阵 (FOCK)	156
18.7.5 选择密度矩阵 (DENSITY)	157
18.8 局域化阈值 (THRESH)	157
18.9 PM局域化选项 (PIPEK)	157
18.10 打印选项 (PRINT)	157

<b>第十九章 MCSCF程序MULTI</b>	<b>159</b>
19.1 输入的结构	160
19.2 定义轨道子空间	161
19.2.1 占据轨道	161
19.2.2 冻芯轨道	161
19.2.3 闭壳层轨道	161
19.2.4 冻结轨道	161
19.3 定义被优化的态	162
19.3.1 定义态的对称性	162
19.3.2 定义当前对称性中态的数量	162
19.3.3 定义态平均计算中的权重	162
19.3.4 动态权重	163
19.4 定义组态空间	163
19.4.1 占据限制	163
19.4.2 选择组态	163
19.4.3 指定轨道组态	164
19.4.4 选择主要组态集	165
19.4.5 对于线型分子，投影到特定的 $\Lambda$ 态	165
19.5 恢复、保存轨道和CI矢量	165
19.5.1 定义初始猜测	165
19.5.2 转动初始轨道对	166
19.5.3 保存最终的轨道	166
19.5.4 对梯度计算保存CI矢量和信息	166
19.5.5 自然轨道	167
19.5.6 赝正则轨道	168
19.5.7 局域化轨道	168
19.5.8 透热轨道	169
19.6 选择优化方法	171
19.6.1 选择CI方法	171
19.6.2 选择轨道优化方法	171
19.6.3 禁用优化	172
19.6.4 禁止额外对称性机制	172
19.7 计算期待值	172
19.7.1 单电子算符的矩阵元	172

19.7.2 双电子算符的矩阵元 . . . . .	172
19.7.3 保存密度矩阵 . . . . .	173
19.8 各类选项 . . . . .	173
19.8.1 打印选项 . . . . .	173
19.8.2 收敛阈值 . . . . .	174
19.8.3 最大迭代数量 . . . . .	174
19.8.4 测试选项 . . . . .	175
19.8.5 特殊的优化参数 . . . . .	175
19.8.6 为CASVB保存波函信息 . . . . .	176
19.8.7 保存转换积分 . . . . .	176
19.9 耦合微扰MCSCF . . . . .	176
19.9.1 SA-MCSCF的梯度 . . . . .	176
19.9.2 SA-MCSCF的差分梯度 . . . . .	177
19.9.3 SA-MCSCF的非绝热耦合矩阵元 . . . . .	177
19.9.4 MCSCF的hessian . . . . .	177
19.10 优化价键波函 . . . . .	178
19.11 提示与诀窍 . . . . .	178
19.12 例子 . . . . .	178
<b>第二十章 CI程序</b>	<b>181</b>
20.1 简介 . . . . .	182
20.2 定义波函 . . . . .	182
20.2.1 占据轨道 . . . . .	182
20.2.2 冻芯轨道 . . . . .	182
20.2.3 闭壳层轨道 . . . . .	182
20.2.4 定义轨道 . . . . .	183
20.2.5 定义态的对称性 . . . . .	183
20.2.6 附加的参考对称性 . . . . .	183
20.2.7 选择组态 . . . . .	184
20.2.8 占据限制 . . . . .	185
20.2.9 直接指定参考组态 . . . . .	185
20.2.10 定义态的数量 . . . . .	185
20.2.11 定义参考态的编号 . . . . .	186
20.2.12 定义轨道对的关联 . . . . .	186

20.2.13 激发类型的限制	187
20.3 选项	187
20.3.1 耦合电子对近似	187
20.3.2 耦合对泛函 (ACPF, AQCC)	187
20.3.3 投影激发态的计算	188
20.3.4 跃迁矩阵元选项	188
20.3.5 收敛阈值	188
20.3.6 能级移动	188
20.3.7 最大数量的迭代	189
20.3.8 限制展开矢的数量	189
20.3.9 选择主组态集	189
20.3.10 正则化外部轨道	189
20.3.11 保存波函	189
20.3.12 初始波函	190
20.3.13 单电子特性	190
20.3.14 计算跃迁矩	190
20.3.15 保存密度矩阵	190
20.3.16 自然轨道	191
20.3.17 杂项	191
20.3.18 各种参数	192
20.4 各种阈值	193
20.5 打印选项	193
20.6 例子	196
<b>第二十一章 多参考Rayleigh-Schrödinger微扰理论</b>	<b>199</b>
21.1 简介	200
21.2 激发态计算	201
21.3 多态CASPT2	201
21.3.1 执行SS-SR-CASPT2计算	202
21.3.2 执行MS-MR-CASPT2计算	204
21.4 零阶哈密顿量中的修正Fock算符	205
21.5 能级移动	205
21.6 积分直接计算	206
21.7 CASPT2梯度	206
21.8 MRCI与MRPT2耦合: CIPT2方法	209
21.9 CASPT2和CASPT3的更多选项	210

<b>第二十二章 NEVPT2计算</b>	<b>213</b>
22.1 总的考虑	213
22.2 输入说明	214
<b>第二十三章 Møller-Plesset微扰理论</b>	<b>217</b>
23.1 MP2的期待值	217
23.2 MP2的极化率和二阶特性	217
23.3 梯度, 期待值和极化率的CPHF	218
23.4 密度拟合MP2 (DF-MP2, RI-MP2)	218
23.5 自旋分量换算MP2 (SCS-MP2)	219
<b>第二十四章 闭壳层CCSD程序</b>	<b>221</b>
24.1 耦合簇, CCSD	222
24.2 二次组态相互作用, QCI	222
24.3 Brueckner耦合簇计算, BCCD	222
24.3.1 BRUECKNER指令	223
24.4 单-双组态相互作用, CISD	223
24.5 DIIS指令	223
24.6 示例	224
24.6.1 H <sub>2</sub> O的单参考关联处理	224
24.6.2 N <sub>2</sub> F <sub>2</sub> 的单参考关联处理	224
24.7 保存密度矩阵	225
24.8 自然轨道	225
24.9 双基组计算	226
<b>第二十五章 使用运动方程CCSD (EOM-CCSD) 的激发态</b>	<b>227</b>
25.1 EOM的选项	227
25.2 EOMPAR卡的选项	228
25.3 EOMPRINT卡的选项	229
25.4 例子	230
25.4.1 氟化氢最低激发态的势能面	230
25.4.2 氟化氢的EOM-CCSD跃迁矩	231
25.4.3 计算最类似于给定CIS态的EOM-CCSD态	232
25.5 CIS的激发态	232
25.6 CCSD的一阶和二阶特性	233

<b>第二十六章 开壳层耦合簇理论</b>	<b>235</b>
26.1 CN的开壳层耦合簇计算	236
<b>第二十七章 M. Kallay的MRCC程序 (MRCC)</b>	<b>237</b>
27.1 安装MRCC	237
27.2 运行MRCC	237
<b>第二十八章 SMILES (Slater基组)</b>	<b>245</b>
28.1 内部基组	246
28.2 外部基组	246
<b>第二十九章 局域关联处理</b>	<b>249</b>
29.1 介绍	249
29.2 开始	251
29.3 选项的总结	252
29.4 指令总结	255
29.5 一般选项	255
29.6 用于选择域的选项	259
29.6.1 标准域	259
29.6.2 扩展域	261
29.6.3 人工定义轨道域 (DOMAIN)	261
29.7 选择轨道对分类的选项	262
29.8 指令	263
29.8.1 LOCAL指令	263
29.8.2 MULTP指令	263
29.8.3 保存波函 (SAVE)	264
29.8.4 续算 (START)	264
29.8.5 电子的关联子集 (REGION)	264
29.8.6 域的合并 (MERGEDOM)	265
29.8.7 分子簇计算中的能量分割 (ENEPART)	265
29.9 正确使用	266
29.9.1 基组	266
29.9.2 对称性和方位	267
29.9.3 局域化	267
29.9.4 轨道域	268



29.9.5 冻结域 . . . . .	269
29.9.6 轨道对分类 . . . . .	269
29.9.7 梯度和频率计算 . . . . .	270
29.9.8 分子间相互作用 . . . . .	271
29.10 密度拟合LMP2 (DF-LMP2) 和耦合簇 (DF-LCCSD(T0)) . . . . .	271
<b>第三十章 显关联方法</b>	<b>273</b>
30.1 参考函数 . . . . .	275
30.2 波函的解 . . . . .	275
30.2.1 一般解 . . . . .	275
30.2.2 对角的解 (D) . . . . .	275
30.2.3 固定振幅的解 (FIX) . . . . .	276
30.3 RI近似 . . . . .	276
30.4 基组 . . . . .	276
30.5 对称性 . . . . .	277
30.6 选项 . . . . .	277
30.7 选择近似的解和级别 . . . . .	279
30.8 CABS单激发校正 . . . . .	282
30.9 CCSD(T)-F12 . . . . .	282
30.10 使用局域近似的DF-LMP2-F12计算 . . . . .	283
30.11 F12程序设定的变量 . . . . .	285
<b>第三十一章 完全CI程序</b>	<b>287</b>
31.1 定义轨道 . . . . .	287
31.2 占据轨道 . . . . .	287
31.3 冻芯轨道 . . . . .	288
31.4 定义态的对称性 . . . . .	288
31.5 打印选项 . . . . .	288
31.6 连接到其它的程序 . . . . .	288
31.7 例子 . . . . .	289
<b>第三十二章 对称匹配分子间微扰理论</b>	<b>291</b>
32.1 简介 . . . . .	291
32.2 第一个例子 . . . . .	292
32.3 DFT-SAPT . . . . .	294

32.4 高阶项 . . . . .	295
32.5 密度拟合 . . . . .	295
32.6 示例 . . . . .	296
32.6.1 使用 $\delta(HF)$ 校正的HF-SAPT计算 $H_2O$ 二聚物 . . . . .	296
32.6.2 使用 $\delta(HF)$ 校正的DFT-SAPT计算 $NeAr$ 二聚物 . . . . .	297
32.6.3 使用 $\delta(HF)$ 校正的DF-DFT-SAPT计算 $NeAr$ 二聚物 . . . . .	298
32.7 选项 . . . . .	300
<b>第三十三章 特性和期待值</b>	<b>303</b>
33.1 特性程序 . . . . .	303
33.1.1 调用特性程序 (PROPERTY) . . . . .	303
33.1.2 期待值 (DENSITY) . . . . .	303
33.1.3 轨道分析 (ORBITAL) . . . . .	303
33.1.4 指定单电子算符 . . . . .	304
33.1.5 打印选项 . . . . .	304
33.1.6 例子 . . . . .	304
33.2 分布多极分析 . . . . .	307
33.2.1 调用DMA程序 (DMA) . . . . .	307
33.2.2 指定密度矩阵 (DENSITY) . . . . .	307
33.2.3 线型分子 (LINEAR, GENERAL) . . . . .	307
33.2.4 多极的最大等级 (LIMIT) . . . . .	308
33.2.5 忽略核的贡献 (NONUCLEAR) . . . . .	308
33.2.6 指定多极格点的位置 (ADD, DELETE) . . . . .	308
33.2.7 定义多极格点位置的半径 (RADIUS) . . . . .	308
33.2.8 注释与参考文献 . . . . .	308
33.2.9 例子 . . . . .	309
33.3 Mulliken布居分析 . . . . .	309
33.3.1 调用布居分析程序 (POP) . . . . .	309
33.3.2 定义密度矩阵 (DENSITY) . . . . .	309
33.3.3 基函数的布居 (INDIVIDUAL) . . . . .	309
33.3.4 例子 . . . . .	309
33.4 自然键轨道分析 . . . . .	310
33.4.1 调用自然键轨道分析程序 (NBO) . . . . .	310
33.4.2 保存NLMO轨道 (SAVE) . . . . .	310

33.5 有限场计算 . . . . .	310
33.5.1 偶极场 (DIP) . . . . .	310
33.5.2 四极场 (QUAD) . . . . .	311
33.5.3 一般场 (FIELD) . . . . .	311
33.5.4 例子 . . . . .	311
33.6 相对论校正 . . . . .	313
33.6.1 例子 . . . . .	313
33.7 CUBE—转存密度或轨道值 . . . . .	314
33.7.1 STEP—设定点的间距 . . . . .	314
33.7.2 DENSITY—密度的来源 . . . . .	314
33.7.3 ORBITAL—轨道的来源 . . . . .	314
33.7.4 AXIS—网格轴的方向 . . . . .	315
33.7.5 BRAGG—网格的空间范围 . . . . .	315
33.7.6 ORIGIN—网格的中心 . . . . .	315
33.7.7 TITLE—用户定义的标题 . . . . .	315
33.7.8 DESCRIPTION—用户定义的描述 . . . . .	315
33.7.9 立方文件的格式 . . . . .	315
33.8 GOPENMOL—计算用于gOpenMol显示的网格 . . . . .	316
<b>第三十四章 相对论校正</b>	<b>319</b>
34.1 使用Douglas-Kroll-Hess哈密顿量 . . . . .	319
34.2 计算相对论校正的例子 . . . . .	320
<b>第三十五章 透热轨道</b>	<b>321</b>
<b>第三十六章 非绝热耦合矩阵元</b>	<b>325</b>
36.1 DDR流程 . . . . .	325
<b>第三十七章 准透热化</b>	<b>329</b>
<b>第三十八章 VB程序CASVB</b>	<b>337</b>
38.1 输入的结构 . . . . .	338
38.2 定义CASSCF波函 . . . . .	338
38.2.1 VBDUMP指令 . . . . .	338
38.3 其它的波函指令 . . . . .	339
38.4 定义价键波函 . . . . .	339

38.4.1 指定轨道组态 . . . . .	339
38.4.2 选择自旋基 . . . . .	339
38.5 重新得到CASSCF CI矢量和VB波函 . . . . .	340
38.6 保存VB波函 . . . . .	340
38.7 指定猜测 . . . . .	341
38.7.1 轨道初始猜测 . . . . .	341
38.7.2 用于结构因子的猜测 . . . . .	341
38.7.3 读取轨道或结构因子 . . . . .	341
38.8 轨道改序 . . . . .	341
38.9 优化控制 . . . . .	342
38.9.1 优化标准 . . . . .	342
38.9.2 迭代数量 . . . . .	342
38.9.3 CASSCF的投影结构因子 . . . . .	342
38.9.4 鞍点优化 . . . . .	342
38.9.5 定义多步优化 . . . . .	342
38.9.6 多步优化 . . . . .	343
38.10 点群对称性和束缚 . . . . .	343
38.10.1 对称操作 . . . . .	343
38.10.2 IRREPS关键词 . . . . .	343
38.10.3 COEFFS关键词 . . . . .	344
38.10.4 TRANS关键词 . . . . .	344
38.10.5 轨道之间的对称性关系 . . . . .	344
38.10.6 SYMPROJ关键词 . . . . .	344
38.10.7 在优化中冻结轨道 . . . . .	344
38.10.8 在优化中冻结结构因子 . . . . .	345
38.10.9 从优化中删除结构 . . . . .	345
38.10.10 正交束缚 . . . . .	345
38.11 波函分析 . . . . .	345
38.11.1 自旋关联分析 . . . . .	345
38.11.2 打印价键结构的权重 . . . . .	346
38.11.3 在VB基中打印CASSCF波函的权重 . . . . .	346
38.12 控制输出的总量 . . . . .	347
38.13 更多工具 . . . . .	347
38.14 服务模式 . . . . .	347
38.15 例子 . . . . .	348

<b>第三十九章 自旋轨道耦合</b>	<b>351</b>
39.1 简介	351
39.2 计算SO积分	351
39.3 计算各个SO矩阵元	352
39.4 在自旋轨道积分和矩阵元的计算中使用近似	353
39.5 整个SO矩阵的计算和对角化	353
39.6 修改非微扰能量	354
39.6.1 自旋轨道计算的打印选项	354
39.6.2 自旋轨道计算的选项	354
39.7 例子	355
39.7.1 对S原子用BP算符计算SO	355
39.7.2 对I原子用ECP计算SO	356
<b>第四十章 能量梯度</b>	<b>361</b>
40.1 解析能量梯度	361
40.1.1 加入梯度 (ADD)	362
40.1.2 缩放梯度 (SCALE)	362
40.1.3 为SCF梯度定义轨道 (ORBITAL)	362
40.1.4 MCSCF梯度 (MCSCF)	362
40.1.5 用SEWARD计算态平均MCSCF梯度	362
40.1.6 用CADPAC计算态平均MCSCF梯度	363
40.1.7 非绝热耦合矩阵元 (NACM)	364
40.1.8 SA-MCSCF的差分梯度 (DEMC)	364
40.1.9 例子	364
40.2 数值梯度	364
40.2.1 坐标的选择 (COORD)	366
40.2.2 变量的数值导数	367
40.2.3 数值梯度的步长	367
40.2.4 活性和非活性坐标	367
40.3 把梯度保存到变量	367
<b>第四十一章 几何优化 (OPTG)</b>	<b>369</b>
41.1 选项	369
41.1.1 选择进行优化的波函和能量选项	370

41.1.2	优化方法的选项 . . . . .	370
41.1.3	修改收敛判据的选项 . . . . .	371
41.1.4	指定优化空间的选项 . . . . .	372
41.1.5	指定优化坐标的选项 . . . . .	372
41.1.6	数值梯度的选项 . . . . .	372
41.1.7	计算Hessian的选项 . . . . .	373
41.1.8	各种选项 . . . . .	374
41.2	OPTG的指令 . . . . .	374
41.2.1	选择优化方法 (METHOD) . . . . .	375
41.2.2	优化坐标 (COORD) . . . . .	376
41.2.3	位移坐标 (DISPLACE) . . . . .	377
41.2.4	定义活性结构参数 (ACTIVE) . . . . .	377
41.2.5	定义非活性结构参数 (INACTIVE) . . . . .	377
41.2.6	Hessian近似 (HESSIAN) . . . . .	377
41.2.7	数值Hessian (NUMHES) . . . . .	379
41.2.8	Hessian元素 (HESSELEM) . . . . .	380
41.2.9	Hessian升级 (UPDATE) . . . . .	380
41.2.10	数值梯度 (NUMERICAL) . . . . .	381
41.2.11	过渡态 (鞍点) 优化 (ROOT) . . . . .	382
41.2.12	设定最大的步长 (STEP) . . . . .	382
41.2.13	重新定义可信比率 (TRUST) . . . . .	382
41.2.14	设定截断参数 (CUT) . . . . .	383
41.2.15	线搜索 (LINESEARCH) . . . . .	383
41.2.16	反应路径跟踪选项 (OPTION) . . . . .	383
41.2.17	优化能量变量 (VARIABLE) . . . . .	384
41.2.18	打印选项 (PRINT) . . . . .	384
41.2.19	圆锥交点优化 (CONICAL) . . . . .	385
41.3	使用SLAPAF程序用于几何优化 . . . . .	387
41.3.1	定义束缚 . . . . .	388
41.3.2	定义内坐标 . . . . .	389
41.3.3	SLAPAF的附加选项 . . . . .	389
41.4	例子 . . . . .	390
41.4.1	使用Z矩阵的简单HF优化 . . . . .	391
41.4.2	使用自然内坐标的优化 (BMAT) . . . . .	392

41.4.3 使用流程的MP2优化 . . . . .	392
41.4.4 使用结构DIIS的优化 . . . . .	393
41.4.5 HCN — HNC异构体的过渡态 . . . . .	394
41.4.6 HCN — HNC异构体的反应路径 . . . . .	396
41.4.7 平衡校正能量的优化 . . . . .	398
<b>第四十二章 振动频率 (FREQUENCIES)</b>	<b>405</b>
42.1 选项 . . . . .	405
42.2 打印选项 (PRINT) . . . . .	407
42.3 保存Hessian及其它信息 (SAVE) . . . . .	407
42.4 续算Hessian或频率 (START) . . . . .	408
42.5 用于数值Hessian计算的坐标 (COORD) . . . . .	408
42.6 用于数值Hessian计算的步长 (STEP) . . . . .	408
42.7 使用能量变量的数值Hessian (VARIABLE) . . . . .	408
42.8 热力学性质 (THERMO) . . . . .	408
42.9 示例 . . . . .	409
<b>第四十三章 函数最小化</b>	<b>413</b>
43.1 例子 . . . . .	414
43.1.1 结构优化 . . . . .	414
43.1.2 基函数优化 . . . . .	414
<b>第四十四章 基组外推</b>	<b>417</b>
44.1 选项 . . . . .	418
44.2 外推函数 . . . . .	419
44.3 用外推能量做结构优化 . . . . .	421
44.4 用外推能量算谐振频率 . . . . .	423
<b>第四十五章 势能曲面 (SURF)</b>	<b>425</b>
45.1 选项 . . . . .	426
45.2 多级别计算 . . . . .	429
45.3 续算功能 . . . . .	432
45.4 简正模式的线性组合 . . . . .	433
45.5 个别坐标的换算 . . . . .	434
45.6 建议 . . . . .	434
45.7 标准问题 . . . . .	434

<b>第四十六章 VSCF程序 (VSCF)</b>	<b>437</b>
46.1 选项 . . . . .	437
46.2 标准问题 . . . . .	438
<b>第四十七章 VCI程序 (VCI)</b>	<b>439</b>
47.1 选项 . . . . .	439
47.2 建议 . . . . .	441
47.3 例子 . . . . .	441
<b>第四十八章 COSMO模型</b>	<b>443</b>
48.1 基本理论 . . . . .	444
<b>第四十九章 QM/MM接口</b>	<b>447</b>
49.1 Chemshell . . . . .	447
<b>第五十章 轨道合并</b>	<b>449</b>
50.1 定义输入的轨道 (ORBITAL) . . . . .	449
50.2 把轨道移动到输出集 (MOVE) . . . . .	449
50.3 把轨道添加到输出集 (ADD) . . . . .	450
50.4 定义额外的对称性 (EXTRA) . . . . .	450
50.5 定义输出集的偏移量 (OFFSET) . . . . .	451
50.6 投影轨道 (PROJECT) . . . . .	451
50.7 对称正交归一化 (ORTH) . . . . .	451
50.8 Schmidt正交归一化 (SCHMIDT) . . . . .	451
50.9 转动轨道 (ROTATE) . . . . .	451
50.10 初始化新的输出集 (INIT) . . . . .	452
50.11 保存合并的轨道 . . . . .	452
50.12 打印选项 (PRINT) . . . . .	452
50.13 例子 . . . . .	452
50.13.1 H <sub>2</sub> F . . . . .	452
50.13.2 NO . . . . .	454
<b>第五十一章 矩阵操作</b>	<b>457</b>
51.1 调用矩阵工具 (MATROP) . . . . .	458
51.2 加载矩阵 (LOAD) . . . . .	458
51.2.1 加载轨道 . . . . .	458



51.2.2 加载密度矩阵 . . . . .	459
51.2.3 加载AO重叠矩阵S . . . . .	459
51.2.4 加载 $S^{-1/2}$ . . . . .	459
51.2.5 加载单电子哈密顿量 . . . . .	459
51.2.6 加载动能或势能算符 . . . . .	459
51.2.7 加载单电子特性算符 . . . . .	459
51.2.8 从无格式记录加载矩阵 . . . . .	460
51.3 保存矩阵 (SAVE) . . . . .	460
51.4 矩阵相加 (ADD) . . . . .	460
51.5 矩阵或两个矩阵乘积的迹 (TRACE) . . . . .	461
51.6 设置变量 (SET) . . . . .	461
51.7 矩阵相乘 (MULT) . . . . .	461
51.8 变换操作 (TRAN) . . . . .	461
51.9 把密度矩阵转换到MO基 (DMO) . . . . .	461
51.10 矩阵对角化 (DIAG) . . . . .	462
51.11 产生自然轨道 (NATORB) . . . . .	462
51.12 构成两个矢量的外积 (OPRD) . . . . .	462
51.13 组合矩阵列 (ADDVEC) . . . . .	462
51.14 构成闭壳层密度矩阵 (DENS) . . . . .	462
51.15 计算Fock矩阵 (FOCK) . . . . .	462
51.16 计算库仑算符 (COUL) . . . . .	463
51.17 计算交换算符 (EXCH) . . . . .	463
51.18 打印矩阵 (PRINT) . . . . .	463
51.19 打印矩阵的对角元 (PRID) . . . . .	463
51.20 打印轨道 (PRIO) . . . . .	463
51.21 把矩阵元赋值给变量 (ELEM) . . . . .	463
51.22 从输入文件读取矩阵 (READ) . . . . .	463
51.23 把矩阵写入ASCII文件 (WRITE) . . . . .	464
51.24 示例 . . . . .	464
51.25 练习: SCF程序 . . . . .	466

<b>附录 A</b>	<b>MOLPRO的安装</b>	<b>471</b>
A.1	获得发布程序 . . . . .	471
A.2	安装预编译的二进制文件 . . . . .	471
A.3	从源文件安装 . . . . .	472
A.3.1	概述 . . . . .	472
A.3.2	预先的准备 . . . . .	472
A.3.3	配置 . . . . .	474
A.3.4	编译与链接 . . . . .	478
A.3.5	修改MOLPRO的默认环境变量 . . . . .	479
A.3.6	调节 . . . . .	479
A.3.7	测试 . . . . .	479
A.3.8	安装产品级的程序 . . . . .	479
A.3.9	获取和使用补丁 . . . . .	480
A.3.10	安装文档 . . . . .	482
A.3.11	openSuSE 11.0的简易安装 . . . . .	482
A.3.12	在Cygwin系统安装 . . . . .	483
A.3.13	在MacOS下的并行安装 . . . . .	484
<b>附录 B</b>	<b>最近的修改</b>	<b>487</b>
B.1	MOLPRO2009.1的新功能 . . . . .	487
B.1.1	显关联计算 . . . . .	487
B.1.2	Hartree-Fock程序的改善 . . . . .	488
B.1.3	结构输入的变化 . . . . .	488
B.1.4	MPI-2并行执行 . . . . .	488
B.2	MOLPRO2008.1的新功能 . . . . .	488
B.3	MOLPRO2006.1的新功能 . . . . .	489
B.4	MOLPRO2002.6的新功能 . . . . .	490
B.5	MOLPRO2002的新功能 . . . . .	491
B.6	MOLPRO2000的新功能 . . . . .	492
B.7	MOLPRO98的新功能 . . . . .	493
<b>附录 C</b>	<b>密度泛函说明</b>	<b>495</b>
C.1	B86: $X\alpha\beta\gamma$ . . . . .	495
C.2	B86MGC: 加上改进梯度校正的 $X\alpha\beta\gamma$ . . . . .	495

C.3 B86R: 重新优化的 $X_{\alpha\beta\gamma}$ . . . . .	496
C.4 B88: 1988年Becke的交换泛函 . . . . .	497
C.5 B88C: 1988年Becke的关联泛函 . . . . .	497
C.6 B95: 1995年Becke的关联泛函 . . . . .	498
C.7 B97DF: B97的密度泛函部分 . . . . .	500
C.8 B97RDF: Hamprecht等人重新参数化B97的密度泛函部分 . . . . .	501
C.9 BR: Becke-Roussel交换泛函 . . . . .	503
C.10 BRUEG: Becke-Roussel交换泛函—均匀电子气极限 . . . . .	503
C.11 BW: Becke-Wigner交换-关联泛函 . . . . .	503
C.12 CS1: Colle-Salvetti关联泛函 . . . . .	504
C.13 CS2: Colle-Salvetti关联泛函 . . . . .	504
C.14 DIRAC: Slater-Dirac交换能量 . . . . .	505
C.15 ECERF: 短程LDA关联泛函 . . . . .	505
C.16 EXERF: 短程LDA交换泛函 . . . . .	507
C.17 G96: Gill的1996梯度校正交换泛函 . . . . .	507
C.18 HCTH120: Handy最小二乘拟合泛函 . . . . .	508
C.19 HCTH147: Handy最小二乘拟合泛函 . . . . .	509
C.20 HCTH93: Handy最小二乘拟合泛函 . . . . .	511
C.21 LTA: 局域 $\tau$ 近似 . . . . .	512
C.22 LYP: Lee, Yang和Parr的关联泛函 . . . . .	513
C.23 MK00: 用于准确虚轨道能量的交换泛函 . . . . .	514
C.24 MK00B: 用于准确虚轨道能量的交换泛函 . . . . .	514
C.25 P86: . . . . .	514
C.26 PBEC: PBE关联泛函 . . . . .	516
C.27 PBEX: PBE交换泛函 . . . . .	518
C.28 PBEXREV: 改进的PBE交换泛函 . . . . .	518
C.29 PW86: . . . . .	519
C.30 PW91C: Perdew-Wang 1991 GGA关联泛函 . . . . .	520
C.31 PW91X: Perdew-Wang 1991 GGA交换泛函 . . . . .	522
C.32 PW92C: Perdew-Wang 1992 GGA关联泛函 . . . . .	522
C.33 STEST: 电子数测试 . . . . .	523
C.34 TH1: Tozer和Handy 1998 . . . . .	523
C.35 TH2: . . . . .	524
C.36 TH3: . . . . .	525

C.37 TH4:	526
C.38 THGFC:	527
C.39 THGFCF0:	528
C.40 THGFC0:	529
C.41 THGFL:	530
C.42 VSXC:	530
C.43 VWN3: Vosko-Wilk-Nusair (1980) III局域关联能	532
C.44 VWN5: Vosko-Wilk-Nusair (1980) V局域关联能	533
索引	535

## I. MOLPRO简介

MOLPRO是一套用于分子电子结构计算的完整的从头计算程序体系，由H.-J. Werner和P. J. Knowles负责设计维护，同时也包含了许多其它作者的贡献。不同于其它的常用量子化学软件包，MOLPRO的重点是高精度计算，通过多组态参考的CI，耦合簇和有关的方法，可以广泛处理电子关联问题。用最近开发的直接积分局域电子关联方法可以极大地减少随分子尺寸增加的计算量，它用准确的从头计算所能处理的分子体系，比大多数其它程序更大。

程序的核心由多组态SCF，多参考CI，和耦合簇程序构成，以及一整套支持功能。软件包包括：

- ◇ 用于一般收缩对称匹配高斯基函数 (*spdfghi*) 的积分生成程序。有两个功能相同的程序：首选的代码是SEWARD (R. Lindh)，在大多数计算机上是最好的；做为替代也可以使用ARGOS (R. M. Pitzer)，在某些情况下，它对小内存的标量计算机做了优化。还有两个不同的梯度积分代码，即CADPAC (R. Amos) 和ALASKA (R. Lindh)。只有后者允许使用一般收缩的对称匹配高斯基函数。
- ◇ 有效芯势 (H. Stoll贡献)。
- ◇ 多种单电子特性。
- ◇ 一些双电子特性，如 $L_x^2$ ， $L_y^2$ ， $L_z^2$ ， $L_x L_y$ ，.....等。
- ◇ 闭壳层和开壳层（自旋限制与非限制）自恰场。
- ◇ Kohn-Sham框架内的密度泛函理论，使用各种梯度修正的交换和关联势。
- ◇ 多组态自恰场。这是二次收敛MCSCF流程，说明在J. Chem. Phys. **82** (1985) 5053。程序可以对几个态平均的加权能量做优化，既可以处理完全的一般组态展开式，也可以处理在Chem. Phys. Letters **115** (1985) 259中描述的长CASSCF展开式。
- ◇ 多参考CI。除了能执行一般的单参考函数的方法 (MP2, SDCI, CEPA) 以外，这一模块还能执行内收缩多参考CI方法，说明见J. Chem. Phys. **89** (1988)

- 5803和Chem. Phys. Lett. **145** (1988) 514。还可以使用非变分的变体（例如MR-ACPF），说明见Theor. Chim. Acta **78** (1990) 175。电子激发态的计算见Theor. Chim. Acta, **84** 95 (1992)。
- ◇ 多参考二阶、三阶微扰理论(MR-PT2, MR-PT3)，说明在Mol. Phys. **89**, 645 (1996)和J. Chem. Phys. **112**, 5546 (2000)。
  - ◇ 闭壳层体系的Møller-Plesset微扰理论 (MPPT)，耦合簇 (CCSD)，二次组态相互作用 (QCISD) 和Brueckner耦合簇 (BCCD)，说明见Chem. Phys. Lett. **190** (1992) 1。还可以计算三阶激发的微扰校正 (Chem. Phys. Letters **227** (1994) 321)。
  - ◇ 开壳层耦合簇理论，说明见J. Chem. Phys. **99** (1993) 5219 和Chem. Phys. Letters **227** (1994) 321。
  - ◇ 完全组态相互作用。它是在Comp. Phys. Commun. **54** (1989) 75中所描述的基于行列式的基准程序。
  - ◇ SCF, DFT, 态平均MCSCF/CASSCF, MRPT2/CASPT2, MP2和QCISD(T)方法的解析能量梯度。
  - ◇ MCSCF解析的非绝热耦合矩阵元。
  - ◇ CASSCF波函以及能量优化价键波函的价键分析，说明在Int. J. Quant. Chem. **65**, 439 (1997)。
  - ◇ MCSCF, MRCI, 和EOM-CCSD波函的单电子跃迁特性，还可以计算不同轨道的波函之间的CASSCF和MRCI跃迁特性。
  - ◇ 自旋-轨道耦合，说明在Mol. Phys., **98**, 1823 (2000)。
  - ◇ MCSCF波函的一些双电子跃迁特性（例如 $L_x^2$ ，等）。
  - ◇ 布居分析。
  - ◇ 轨道局域化。
  - ◇ 分布式多极分析 (A. J. Stone)。
  - ◇ 自动几何优化，说明在J. Comp. Chem. **18**, (1997), 1473。
  - ◇ 自动计算振动频率，强度以及热力学特性。
  - ◇ 跟踪反应路径，说明在Theor. Chem. Acc. **100**, (1998), 21。
  - ◇ 各种工具程序，可以进行更多的一般优化，循环和分枝（例如自动产生完整的势能面），一般的文件管理等操作。
  - ◇ 结构输出有xyz, MOLDEN 和GAUSSIAN 格式；分子轨道和频率输出为MOLDEN 格式。
  - ◇ Hartree-Fock, DFT, 和电子对关联方法 (MP, CCSD, MRCI, 等) 的直接积分

- 执行, 说明在Mol. Phys., **96**, (1999), 719。目前还不能执行微扰三阶激发方法。
- ◇ 局域的二阶Møller-Plesset微扰理论 (LMP2) 和局域的耦合簇方法, 说明在J. Chem. Phys. **104**, 6286 (1996), Chem. Phys. Lett. **290**, 143 (1998), J. Chem. Phys. **111**, 5691 (1999), J. Chem. Phys. **113**, 9443 (2000), J. Chem. Phys. **113**, 9986 (2000), Chem. Phys. Letters **318**, 370 (2000), J. Chem. Phys. **114**, 661 (2001), Phys. Chem. Chem. Phys. **4**, 3941 (2002).
  - ◇ 局域密度拟合方法, 说明在J. Chem. Phys. **118**, 8149 (2003), Phys. Chem. Chem. Phys. **5**, 3349 (2003), Mol. Phys. **102**, 2311 (2004).
  - ◇ LMP2和DF-LMP2的解析能量梯度, 说明在J. Chem. Phys. **108**, (1998), 5185, J. Chem. Phys. **121**, 737 (2004)。
  - ◇ 显关联方法, 说明在J. Chem. Phys. **119**, 4607 (2003), J. Chem. Phys. **121**, 4479 (2004), J. Chem. Phys. **124**, 054114 (2006), J. Chem. Phys. **124**, 094103 (2006).
  - ◇ 分布式内存计算机的并行执行, 说明见J. Comp. Chem. **19**, (1998), 1215。在通常的积分求解计算中, SCF, DFT, MRCI, MP2, LMP2, CCSD(T)能量以及SCF, DFT梯度目前是并行化的; 直接积分和密度拟合的SCF, DFT, LMP2和LCCSD(T)也是并行的。

程序主要由标准的Fortran-90编写。依赖计算机的部分用预处理程序管理, 可以比较容易地在不同的计算机上转换。程序的每个发布版在IBM RS/6000, Hewlett-Packard, Silicon Graphics, Compaq, 以及Linux系统都移植和测试过。这些计算机硬件和操作系统的较完整说明, 可以在这里找到<http://www.molpro.net/supported>。程序还可以运行在Cray, Sun, Convex, Fujitsu, 和NEC SX4平台上, 以及前面所列计算机的旧体系和/或操作系统上, 但是对这些系统没有可靠的测试, 在移植时可能还需要手工调节代码。程序提供了相当庞大的公用轨道基组库, 可以根据需要扩充。还有详尽的用户手册, 其中包括安装说明。手册的格式有PostScript, PDF, 以及可以在互联网服务器上安装的HTML<sup>1</sup>。

2006.1版本中的新方法和增强包括:

- ◇ 更一致的输入语言和输入预检查。
- ◇ 更灵活的基组输入, 允许操作多个基组。
- ◇ 更高效的新密度泛函执行, 额外的密度泛函。
- ◇ 低标度的局域耦合簇方法, 以及微扰处理三激发 (LCCSD(T)以及各种变体, 如LQCISD(T))。

---

<sup>1</sup>中文译本只提供PDF格式。——译注。

- ◇ 更高效的密度拟合 (DF) 程序, 用于Hartree-Fock (DF-HF), 密度泛函Kohn-Sham理论 (DF-KS), 二阶Møller-Plesset微扰理论 (DF-MP2), 以及所有的局域方法 (DF-LMP2, DF-LMP4, DF-LQCISD(T), 和DF-LCCSD(T))。
- ◇ QCISD(T)解析梯度。
- ◇ MRPT2 (CASPT2) 和多态CASPT2的解析梯度, 使用态平均MCSCF参考函数。
- ◇ DF-HF, DF-KS, DF-LMP2, 和DF-SCS-LMP2解析梯度。
- ◇ 使用密度拟合的显关联方法DF-MP2-R12/2A', DF-MP2-F12/2A', 以及局域变体DF-LMP2-R12/2\*A(loc)和DF-LMP2-F12/2\*A(loc)。
- ◇ 多态MRPT2, MS-CASPT2。
- ◇ 多参考微扰理论和组态相互作用的组合 (CIPT2)。
- ◇ DFT-SAPT。
- ◇ 不同轨道的CASSCF和MRCI波函之间的跃迁矩和跃迁哈密顿量。
- ◇ 到任意阶的Douglas-Kroll-Hess哈密顿量。
- ◇ 新的自旋-轨道积分程序, 用于一般收缩基组。
- ◇ 改善了几何优化和数值Hessian的计算流程, 包括束缚优化。
- ◇ 对于QM/MM计算, 包括格子梯度计算, 改善了处理大格子点电荷的工具。
- ◇ 到M. Kallay的MIRCC程序的接口, 可以进行到任意激发级别的耦合簇计算。
- ◇ 数值梯度和Hessian的自动完全并行 (*embarrassingly parallel*) 计算 (mppx版)。
- ◇ 另外的并行代码, 例如, DF-HF, DF-KS, DF-LCCSD(T) (含三激发在内, 只实现了部分并行)。

2008.1版本中的新方法和增强包括:

- ◇ 使用密度拟合的显关联MP2-F12方法: 使用多种不同解的闭壳层和开壳层DF-MP2-F12, DF-LMP2-F12, 以及DF-RMP2-F12, 说明见H.-J. Werner, T. B. Adler, and F. R. Manby, J. Chem. Phys. **126**, 164102 (2007)和G. Knizia and H.-J. Werner, J. Chem. Phys. **128**, 154103 (2008)。
- ◇ (闭壳层和开壳层的) 显关联CCSD-F12和CCSD(T)方法, 说明见T. B. Adler, G. Knizia, and H.-J. Werner, J. Chem. Phys. **127**, 221106 (2007)。
- ◇ 自然局域轨道, NPA和NBO分析以及改善的方法, 用于局域关联计算的域选择, 说明见R. A. Mata and H.-J. Werner, Mol. Phys. **105**, 2753 (2007); 此外参见R. A. Mata and H.-J. Werner, J. Chem. Phys. **125**, 184110 (2006)。
- ◇ 局域关联计算中的关联区, 说明见R. A. Mata, H.-J. Werner and M. Schütz, J. Chem. Phys. **128**, 144106 (2008)。



- ◇ 用VCI方法自动计算非谐振动频率和零点能，说明见T. Hrenar, H.-J. Werner, and G. Rauhut, J. Chem. Phys. **126**, 134108 (2007)及其中的参考文献。
- ◇ 动态的态加权，说明见M. P. Deskevich and D. J. Nesbitt, and H.-J. Werner, J. Chem. Phys. **120**, 7281 (2004)。
- ◇ DFT和耦合簇的耦合方法，说明见E. Goll, T. Leininger, F. R. Manby, A. Mitrushchenkov, H.-J. Werner, and H. Stoll, Phys. Chem. Chem. Phys. **10**, 3353 (2008)及其中的参考文献。
- ◇ 局域响应方法（LCC2）用于计算大分子的激发能和跃迁特性，说明见D. Kats, T. Korona, M. Schütz, J. Chem. Phys. **125**, 104106 (2006)和D. Kats, T. Korona, M. Schütz, J. Chem. Phys. **127**, 064107 (2007)。
- ◇ 自动的基组外推。
- ◇ 增强了到其它程序的连接，包括图形显示输出和三维结构。
- ◇ MPI2并行执行。
- ◇ 支持Mac OS X。

目前正在开发中的未来增强功能有：

- ◇ 开壳层局域耦合簇方法。
- ◇ CCSD(T)和CCSD(T)-F12的解析能量梯度。
- ◇ DFT的解析二阶导。

这些功能将包含在以后阶段的基本版中。以上所列只是提供消息，不代表它们中的任何一个会在确定的时间发布。



## II. 互联网上的MOLPRO

关于MOLPRO以及程序升级在内的最新消息，可以在以下网址找到  
<http://www.molpro.net/>



所有使用本程序结果的出版物必须包含以下引用：

MOLPRO, version 2009.1, a package of *ab initio* programs, H.-J. Werner, P. J. Knowles, R. Lindh, F. R. Manby, M. Schütz, P. Celani, T. Korona, A. Mitrushenkov, G. Rauhut, T. B. Adler, R. D. Amos, A. Bernhardsson, A. Berning, D. L. Cooper, M. J. O. Deegan, A. J. Dobbyn, F. Eckert, E. Goll, C. Hampel, G. Hetzer, T. Hrenar, G. Knizia, C. Köppl, Y. Liu, A. W. Lloyd, R. A. Mata, A. J. May, S. J. McNicholas, W. Meyer, M. E. Mura, A. Nicklass, P. Palmieri, K. Pflüger, R. Pitzer, M. Reiher, U. Schumann, H. Stoll, A. J. Stone, R. Tarroni, T. Thorsteinsson, M. Wang, and A. Wolf, see <http://www.molpro.net>.

有些杂志要求使用较短的作者列表。这种情况下，换用以下引用：

MOLPRO, version 2009.1, a package of *ab initio* programs, H.-J. Werner, P. J. Knowles, R. Lindh, F. R. Manby, M. Schütz, and others, see <http://www.molpro.net>.

根据使用程序的不同，还应当引用下面的参考文献：

#### 积分求解（SEWARD）

R. Lindh, U. Ryu, and B. Liu, J. Chem. Phys. **95**, 5889 (1991).

#### 积分直接执行

M. Schütz, R. Lindh, and H.-J. Werner, Mol. Phys. **96**, 719 (1999).

#### MCSCF/CASSCF

H.-J. Werner and P. J. Knowles, J. Chem. Phys. **82**, 5053 (1985);

P. J. Knowles and H.-J. Werner, Chem. Phys. Lett. **115**, 259 (1985).

此外，参见：

H.-J. Werner and W. Meyer, J. Chem. Phys. **73**, 2342 (1980);

H.-J. Werner and W. Meyer, J. Chem. Phys. **74**, 5794 (1981);

H.-J. Werner, Adv. Chem. Phys. **LXIX**, 1 (1987).

### 内收缩MRCI

H.-J. Werner and P.J. Knowles, J. Chem. Phys. **89**, 5803 (1988);

P.J. Knowles and H.-J. Werner, Chem. Phys. Lett. **145**, 514 (1988).

此外, 参见:

H.-J. Werner and E.A. Reinsch, J. Chem. Phys. **76**, 3144 (1982);

H.-J. Werner, Adv. Chem. Phys. **LXIX**, 1 (1987).

### 用内收缩MRCI计算激发态

P. J. Knowles and H.-J. Werner, Theor. Chim. Acta **84**, 95 (1992).

### 内收缩MR-ACPF, QDVPT, 等

H.-J. Werner and P. J. Knowles, Theor. Chim Acta **78**, 175 (1990).

非收缩MR-ACPF, QDVPT, MR-ACQQ的原始文献:

R. J. Gdanitz and R. Ahlrichs, Chem. Phys. Lett. **143**, 413 (1988);

R. J. Cave and E. R. Davidson, J. Chem. Phys. **89**, 6798 (1988);

P. G. Szalay and R. J. Bartlett, Chem. Phys. Lett. **214**, 481 (1993).

### 多参考微扰理论 (CASPT2/CASPT3)

H.-J. Werner, Mol. Phys. **89**, 645 (1996);

P. Celani and H.-J. Werner, J. Chem. Phys. **112**, 5546 (2000).

多参考组态相互作用与多参考微扰理论的组合: P. Celani, H. Stoll, and H.-J. Werner, Mol. Phys. **102**, 2369 (2004).

### 解析能量梯度和几何优化

梯度积分求解 (ALASKA): R. Lindh, Theor. Chim. Acta **85**, 423 (1993);

MCSCF梯度: T. Busch, A. Degli Esposti, and H.-J. Werner, J. Chem. Phys. **94**, 6708 (1991);

MP2和LMP2梯度: A. El Azhary, G. Rauhut, P. Pulay, and H.-J. Werner, J. Chem. Phys. **108**, 5185 (1998);

DF-LMP2梯度: M. Schütz, H.-J. Werner, R. Lindh and F. R. Manby, J. Chem. Phys. **121**, 737 (2004).

QCISD和LQCISD梯度: G. Rauhut and H.-J. Werner, Phys. Chem. Chem. Phys. **3**, 4853 (2001);

CASPT2梯度: P. Celani and H.-J. Werner, *J. Chem. Phys.* **119**, 5044 (2003).

几何优化: F. Eckert, P. Pulay and H.-J. Werner, *J. Comp. Chemistry* **18**, 1473 (1997);

反应路径跟踪: F. Eckert and H.-J. Werner, *Theor. Chem. Acc.* **100**, 21, (1998).

#### 谐振频率

G. Rauhut, A. El Azhary, F. Eckert, U. Schumann, and H.-J. Werner, *Spectrochimica Acta* **55**, 651 (1999).

T. Hrenar, G. Rauhut, H.-J. Werner, *J. Phys. Chem. A* **110**, 2060 (2006).

#### 非谐振频率 (SURF, VSCF, VCI)

G. Rauhut, *J. Chem. Phys.* **121**, 9313 (2004);

T. Hrenar, H.-J. Werner, G. Rauhut, *J. Chem. Phys.* **126**, 134108 (2007);

G. Rauhut, T. Hrenar, *Chem. Phys.* **346**, 160 (2008);

G. Rauhut, *J. Chem. Phys.* **127**, 184109 (2007).

#### Møller-Plesset微扰理论 (MP2, MP3, MP4)

直到四阶 (MP4(SDTQ)) 的闭壳层Møller-Plesset微扰理论是耦合簇代码的一部分。参见CCSD。

#### 开壳层Møller-Plesset微扰理论 (RMP2)

R. D. Amos, J. S. Andrews, N. C. Handy, and P. J. Knowles, *Chem. Phys. Lett.* **185**, 256 (1991).

#### 耦合簇处理 (QCI, CCSD, BCCD)

C. Hampel, K. Peterson, and H.-J. Werner, *Chem. Phys. Lett.* **190**, 1 (1992), 及其引用的文献。计算微扰三阶校正的程序来自M. J. O. Deegan and P. J. Knowles, *Chem. Phys. Lett.* **227**, 321 (1994).

#### 运动方程单双耦合簇 (EOM-CCSD)

T. Korona and H.-J. Werner, *J. Chem. Phys.* **118**, 3006 (2003).

#### 开壳层耦合簇 (RCCSD, UCCSD)

P. J. Knowles, C. Hampel and H.-J. Werner, *J. Chem. Phys.* **99**, 5219 (1993); 勘误: *J. Chem. Phys.* **112**, 3106 (2000).

#### 局域MP2 (LMP2)

G. Hetzer, P. Pulay, and H.-J. Werner, Chem. Phys. Lett. **290**, 143 (1998);  
M. Schütz, G. Hetzer, and H.-J. Werner, J. Chem. Phys. **111**, 5691 (1999);  
G. Hetzer, M. Schütz, H. Stoll, and H.-J. Werner, J. Chem. Phys. **113**, 9443 (2000).  
此外, 参见能量梯度和密度拟合的文献。

### 局域耦合簇方法 (LCCSD, LQCISD, LMP4)

C. Hampel and H.-J. Werner, J. Chem. Phys. **104** 6286 (1996);  
M. Schütz and H.-J. Werner, J. Chem. Phys. **114**, 661 (2001);  
M. Schütz, Phys. Chem. Chem. Phys. **4**, 3941 (2002).  
此外, 参见能量梯度和密度拟合的文献。

### 局域三激发

M. Schütz and H.-J. Werner, Chem. Phys. Lett. **318**, 370 (2000);  
M. Schütz, J. Chem. Phys. **113**, 9986 (2000);  
M. Schütz, J. Chem. Phys. **116**, 8772 (2002).

### 密度拟合方法

DFT泊松拟合: F. R. Manby, P. J. Knowles, and A. W. Lloyd, J. Chem. Phys. **115**, 9144 (2001).

DF-MP2, DF-LMP2: H.-J. Werner, F. R. Manby, and P. J. Knowles, J. Chem. Phys. **118**, 8149 (2003).

DF-LCCSD: M. Schütz and F. R. Manby, Phys. Chem. Chem. Phys. **5**, 3349 (2003).

DF-HF: R. Polly, H.-J. Werner, F. R. Manby, and Peter J. Knowles, Mol. Phys. **102**, 2311 (2004).

DF-LMP2梯度: M. Schütz, H.-J. Werner, R. Lindh and F. R. Manby, J. Chem. Phys. **121**, 737 (2004).

DF-LCCSD(T): H.-J. Werner and M. Schütz, in preparation.

### 使用密度拟合的显关联方法

DF-MP2-R12: F. R. Manby, J. Chem. Phys. **119**, 4807 (2003).

DF-MP2-F12/2A: A. J. May and F. R. Manby, J. Chem. Phys. **121**, 4479 (2004).

DF-MP2-F12: H.-J. Werner, T. B. Adler, and F. R. Manby, J. Chem. Phys. **126**, 164102 (2007).

DF-RMP2-F12: G. Knizia and H.-J. Werner, J. Chem. Phys. **128**, 154103 (2008).



DF-LMP2-F12: H.-J. Werner and F. R. Manby, J. Chem. Phys. **124**, 054114 (2006); F. R. Manby H.-J. Werner, T. B. Adler, and A. J. May, J. Chem. Phys. **124**, 094103 (2006); H.-J. Werner, T. B. Adler, and F. R. Manby, J. Chem. Phys. **126**, 164102 (2007); T. B. Adler, F. R. Manby and H.-J. Werner, J. Chem. Phys. **130**, 054106 (2009).

CCSD(T)-F12: T. B. Adler, G. Knizia, and H.-J. Werner, J. Chem. Phys. **127**, 221106 (2007).

UCCSD(T)-F12: G. Knizia, T. B. Adler, and H.-J. Werner, J. Chem. Phys. **130**, 054104 (2009).

### 范围分离的杂化方法

T. Leininger, H. Stoll, H.-J. Werner, and A. Savin, Chem. Phys. Lett. **275**, 151 (1997).

J. G. Ángyán, I.C. Gerber, A. Savin, and J. Toulouse, Phys. Rev. A **72**, 012510 (2005).

E. Goll, H.-J. Werner, and H. Stoll, Phys. Chem. Chem. Phys. **7**, 3917 (2005).

### 完全CI (FCI)

P. J. Knowles and N. C. Handy, Chem. Phys. Letters **111**, 315 (1984);

P. J. Knowles and N. C. Handy, Comp. Phys. Commun. **54**, 75 (1989).

### 分布式多极分析 (DMA)

A. J. Stone, Chem. Phys. Letters **83**, 233 (1981).

### 价键

D. L. Cooper, T. Thorsteinsson, and J. Gerratt, Int. J. Quant. Chem. **65**, 439 (1997);

D. L. Cooper, T. Thorsteinsson, and J. Gerratt, Adv. Quant. Chem. **32**, 51-67 (1998).

此外, 参见: “An overview of the CASVB approach to modern valence bond calculations”, T. Thorsteinsson and D. L. Cooper, in *Quantum Systems in Chemistry and Physics. Volume 1: Basic problems and models systems*, eds. A. Hernández-Laguna, J. Maruani, R. McWeeny, and S. Wilson (Kluwer, Dordrecht, 2000); pp 303-26.

### 使用Douglas-Kroll哈密顿量的相对论校正

M. Reiher, A. Wolf, J. Chem. Phys. **121**, 2037-2047 (2004);

M. Reiher, A. Wolf, J. Chem. Phys. **121**, 10945-10956 (2004);

A. Wolf, M. Reiher, B. A. Hess, J. Chem. Phys. **117**, 9215-9226 (2002).

### 自旋-轨道耦合

A. Berning, M. Schweizer, H.-J. Werner, P. J. Knowles, and P. Palmieri, Mol. Phys., **98**, 1823 (2000).

### 透热化流程

H.-J. Werner and W. Meyer, J. Chem. Phys. **74**, 5802 (1981);

H.-J. Werner, B. Follmeg, and M. H. Alexander, J. Chem. Phys. **89**, 3139 (1988);

D. Simah, B. Hartke, and H.-J. Werner, J. Chem. Phys. **111**, 4523 (1999).

### DF-DFT-SAPT

A. Heßelmann, G. Jansen and M. Schütz, J. Chem. Phys. **122**, 014103 (2005).

## 第一章

## 如何阅读手册

手册的结构如下：

下一章给出MOLPRO一般结构的概述，为了了解对称性，记录与文件，以及轨道空间等定义的习惯，新用户阅读这一章是必要的。在后面的章节中详细讲述了各个程序模块的输入，并假定你已经熟悉了这些概念。附录详细讲解了程序的运行，以及安装过程。

在整个手册中，打印体（**Typewriter Font**）的字是MOLPRO认识的关键词。在输入文件中，它们如原样显示，但可以是大写或小写。必须由用户提供的数值或者选项是斜体（*italic*）的。在有些情况下，一个输入记录可能有几种不同的形式。这表示为[选项]（*[options]*），可用的选项在随后的各个章节分别说明。



## 第二章

## 运行MOLPRO

在UNIX系统中，MOLPRO通过unix命令 `molpro` 访问。语法为：

```
molpro [options] [datafile]
```

执行MOLPRO由用户准备的数据控制；如果命令行没有给出 *datafile*，那么从标准输入读取数据，程序的结果放到标准输出。否则从 *datafile* 获得数据，并把输出写到一个名称由 *datafile* 截掉尾部扩展名并加上 `.out` 产生的文件中。如果输出文件已经存在，那么把旧文件添加到名称相同但后缀为 `.out.1` 的文件中，并删除原文件。这提供了保存旧输出文件防止覆盖的机制。注意，上面的功能可以用 `-o` 或 `-s` 选项修改。

### 2.1 选项

大多数的选项是不需要的，因为一般情况下设置了合适的系统默认值。下面给出了详细的选项，可以位于命令行中，环境变量 `MOLPRO_OPTIONS` 中，或文件 `./molpro.rc`，`$HOME/.molpro.rc`，和 `library` 文件目录下的 `tuning.rc` 中，按照优先级递减的顺序排列。

```
-d dir1:dir2:...
```

其中的 *dir1:dir2:...* 是一列用于创建临时文件的目录。每个目录对于程序使用者都应当是可写的，并且目录的定义可以用脚本形式包含嵌入的环境变量，例如 `$TMPDIR` 或 `/tmp/$USER`；它们在运行时将会被展开。如果有多个草稿文件系统，那么方便的做法是定义目录列表，在每个文件系统中都有一个目录。MOLPRO的有些部分需要额外的I/O需求，因此在优化草稿目录的设置和规定时仔细一些是非常重要的。

注意，在创建bin/molpro时使用了环境变量\$TMPDIR， \$TMPDIR2， \$TMPDIR3， ...， 用于构建-d选项的草稿目录列表。因此在编译时，这些环境变量应当填入每个可用的草稿文件系统的目录名（参见A.3.3）。

-o | --output *outfile*

定义不同的输出文件。

-x | --executable *executable*

定义备选的MOLPRO可执行文件。

-d | --directory *directory1:directory2...*

定义程序放置临时文件的一组目录。最优定义的详细讨论见安装说明。

--backup *nfile*

允许保存以前的输出文件，直到最大数量*nfile*。如果忽略了*nfile*，默认为无穷大。通过为输出文件名后面追加\_和序号来构造备份文件名，产生常规的和xml格式的输出文件，以及log文件。

-a | --append-backup

以前的输出文件通过追加而串连起来，而不是分别保存。

--directory-backup

备份文件存储在一个名为*datafile.d*的单独子目录中，其中又有子目录1， 2， 等。--directory-backup和--append-backup是互相排斥的，打开其中一个将会强制关闭另一个。

--backup-directory *dir*

同--directory-backup， 但使用*dir*而不是默认子目录作为备份文件的存放位置。

-s | --nobackup

禁止保存已存在输出文件的机制。

-v | --verbose

导致回显调试信息的流程；用--noverbose选择安静操作（默认）。

-k *key*

其中*key*是许可文件。通常不需要这个选项，因为在安装MOLPRO 时，许可文件是全站安装的。

**-m** 指定分配给程序的工作内存，单位是八字节的字。内存也可以用1000个字的单位给出，通过在数值的后面添加字母**k**，或者用字母**m**表示单位1000000，字母**g**表示 $10^9$ 。K, B, G分别代表 $2^{10}$ ,  $2^{20}$ ,  $2^{30}$ 。

**-I | --main-file-repository *directory***

指定任何积分文件（文件1）永久拷贝保存的目录。这可以是绝对路径或者相对于当前目录的相对路径（例如，“.”定义当前的目录）。-I目录和-d工作目录通常相同，避免复制大积分文件，因为在完成任务后，文件将复制到-I后面所给的目录中。在有些主机下，任务终止后会自动擦除草稿目录，这种情况下可以指定不同的-I目录，例如\$HOME/int（环境变量将在运行时展开）。但是考虑到积分文件的大尺寸，这在使用时需要小心。注意，在一个处理器以上的并行运行中，积分文件即不会复制，也不会重新开始。

**-W | --wavefunction-file-repository *directory***

类似于--main-file-repository，但是它和波函文件（2, 3, 和4）的目录有关。它决定了永久波函（转存）文件的目标目录，用于存储像轨道和CI因子一类的信息。这些文件对于续算任务是必不可少的。如在上面积分文件所解释的，在任务完成后，永久波函文件将被复制到*directory*。*directory*默认为\$HOME/wfu。

**-X | --xml-output**

指定输出文件为格式规范的XML文件，适于自动后期处理。重要的数据如输入，结构，以及结果都被加上标记，通常的大块描述性输出都作为XML注释折叠起来。用--no-xml-output关闭这一功能，强制产生纯文本输出。

**-L | --library *directory***

指定能找到基组库文件（LIBMOL\*）的目录。

**-1 | --file-1-directory *directory:directory:...***

指定运行时间文件1所在的目录，仅对这个文件覆盖--directory。也可以类似地使用-2, -3, -4, -5, -6, -7, -8, 和-9。通常不用给出这些选项，因为程序将尝试用-d中给出的目录，以达到I/O的最佳分配。

**-t | --omp-num-threads *n***

如果环境变量OPENMP\_NUM\_THREADS设为 $n$ 的话，指定OpenMP线程的数量。

还有其它一些用于变量调节和系统参数的选项，但是通常它们和普通用户无关。

这些选项通常都不需要指定，因为默认设置已经很好。有时和安装有关的选项可以在系统配置文件 `molpro.rc` 中找到，它和MOLPRO库文件的目录相同。

## 2.2 在并行计算机上运行MOLPRO

在Global Arrays并行工具箱或MPI-2库的控制下，MOLPRO能运行于分布式内存的多处理器系统上，包括工作站集群。部分代码通过OpenMP协议还能利用共享内存并行，但是并行功能有限，这一功能目前不推荐使用。应当注意其余部分代码还没有并行化，或者只做了部分并行化，因此用复制工作方式运行。另外，有些并行化的部分依赖于快速节点间的通信，在普通的网络上效率会很低。因此需要小心测试，以避免浪费多用户环境的资源。

MOLPRO可以用三种不同的方式编译：

1. 仅串行执行。在这种情况下，运行时不能做并行。
2. “MPP”：程序的多个拷贝对一个任务同时执行。例如，可以并行运行一个CCSD(T) 计算，通过把工作在几个处理器之间分配，实现较少的占用时间。
3. “MPPX”：一定数量的程序拷贝用串行方式运行同一计算中独立的各个任务。例子是有限差分的梯度和频率计算：对于初始波函的计算，计算复制到所有的进程上，但是此后，每个进程串行计算不同的位移结构。目前，这只能对数值梯度和Hessian执行。

使用三种模式中的哪一个在编译时就定下了，这会在任务的输出中给出。以下是选择处理器数量和位置的选项，它们对于MPP和MPPX是相同的。

### 2.2.1 定义并行的执行

`molpro`命令的以下附加选项可用于指定和控制并行执行。

`-n | --tasks tasks/tasks_per_node:smp_threads`

tasks定义设置进程的数量，默认是1。 `tasks_per_node`设置每个节点运行的合适的GA（或MPI-2）进程数量。默认设置和安装有关。在某些



环境下（例如，在Loadleveler下运行IBM；PBS批处理任务），`-n`给定的值为环境变量允许的最大值的上限，在这种情况下把`-n`的值设置为非常大的数是非常有用的，于是进程数量的控制由批处理任务来定义。`smp_threads`与使用OpenMP共享内存并行有关，用于指定打开的最大数量OpenMP线程，默认是1。这三部分的任何一个都可以略去，而它们的合适组合可以进行GA（或MPI-2）并行，OpenMP并行，或者混合并行。

`-N | --task-specification` `user1:node1:tasks1,user2:node2:tasks2...`

`node1`, `node2`, 等指定工作节点的主机名。对于大多数并行系统，`node1`默认为本地主机名，`node2`和其它没有默认设置。对于Cray T3E和IBM SP系统，以及在PBS批处理体系下运行的系统，如果没有指定`-N`，那么节点会按照标准方式从系统获得。`tasks1`, `tasks2`等可用于控制每个节点上的任务数量，它们比用`-n / tasks_per_node`更灵活。如果忽略，每个都设定为等于`-n / tasks_per_node`。`user1`, `user2`等给出了创建进程的用户名。这些参数中的大多数可以忽略，因为默认值通常已足够。

`-G | --global-memory` *memory*

程序的有些部分利用Global Arrays用于临时数据结构的保存和通信。该选项用来为这一功能设置在所有处理器上分配内存的总量。

注意：对于用MPI-2库创建的MOLPRO，当指定 $n$ 个处理器时，那么 $n-1$ 个处理器用于计算，一个处理器用于作为共享计数服务器（对于 $n=1$ ，一个处理器用于计算，不需要共享计数服务器）。即便如此，当用大量处理器运行时，这种性能也是很有竞争力的。



## 第三章

# 定义MOLPRO输入语言

### 3.1 输入格式

MOLPRO的执行由输入文件控制。通常每个输入记录由一个关键词开始，之后可以是数据或者其它关键词。MOLPRO的输入包含命令，指令，选项和数据。命令和指令按照出现的顺序先后执行。此外，可以在输入的任何地方定义流程，流程中可以包含任意数量的命令和指令。它们仅有当调用时才执行（这可能在输入文件中的定义之前或之后）。

输入文件可按照自由格式书写。可遵照下面的约定：

- |            |                        |
|------------|------------------------|
| , (逗号)     | 移到下一个记录停止点，即，用来划定输入范围。 |
| ;(分号)      | 结束记录，即，开始新的记录。         |
| !(感叹号)     | 忽略该输入行后面的内容（对于注释很有用）。  |
| --- (三个横线) | 结束文件（忽略以后的输入行）。        |

输入可以是大写或者小写。输入处理程序将把全部字符转换为大写。所有的整数后面都会加上“.”（程序只读取浮点数）。

实际上，在一行内可以输入几个逻辑输入记录，并用分号隔开，即，一个给定的输入行可以包含多个有效命令（用分号隔开），如果喜欢的话也可以只有一个。这些用分号隔开的基本命令单位（记录）在这个手册中也经常称为卡（*card*）。

这些一般规则的例外有：

- |         |           |
|---------|-----------|
| ***     | 总是第一个数据行  |
| INCLUDE | 包含其它的输入文件 |

FILE	指定已命名的文件
TEXT	打印文本
TITLE	定义运行的标题或表格的标题
CON	指定轨道组态
---	输入的最后一行

这些命令总是占据一整行。使用INCLUDE可以打开辅助输入文件。如果遇到了INCLUDE命令，则打开新的输入文件并读取到结束。然后在第一个输入文件中继续INCLUDE卡之后的输入。INCLUDE也可以嵌套。

一条MOLPRO输入记录（卡）由多个输入域构成。输入域的宽度最多到256个字符，可以包含表达式或者字符串。域可以由逗号或空格分隔。为了避免意想不到的结果，我们推荐一般使用逗号。

每一行可以从一个标签开始。标签与该行的主体部分可以用一个冒号(:)隔开。冒号是标签的一部分。标签的长度不能超过六个字符（包括冒号），而且标签必须唯一。标签对于GOTO命令很有用。例如：

```
GOTO,START:
...
START: CCSD(T)
```

这里的START:就是标签，CCSD(T)是命令。

输入包含空格的字符串可以加上引号。例如，'This is a string'会被当作一个字符串，但This is a string则是四个位置连续的字符串。引号内的字符串不会转化成大写。

输入行的续行可以在行末加上\用于连接。续行数量是任意的，包含的字符总长度最高可以到1024（早期的IBM系统只能到500个字符）。

如果所用的Unix系统支持长文件名，那么文件名的长度可以到31个字符。但是早期的CRAY系统例外，它的二进制MOLPRO文件名只允许8个字符。

## 3.2 命令

一个命令调用一个特定的程序。它之后可以接该程序的局域输入，并且用花括号括起来<sup>1</sup>。

<sup>1</sup>根据lib/variable.registry文件中所设的参数STRICTCHECK，程序可以接受未用花括号的命令之后的指令。程序检查输入中不明确的地方。若命令或流程与已知的名称相同，并且指令没有位于命令区（也就是说，没有使用花括号），那么就认为指令的定义不明确。STRICTCHECK=0：若不明确的指令之后是对当前命令有效的明确指令，那

一般格式为

```
命令,选项
```

或者

```
{命令,选项  
指令  
数据  
}
```

命令的例子有HF, MP2, CCSD(T), MCSCF, MRCI。指令的例子有OCC, CLOSED, WF, PRINT。指令可以是任何顺序, 除非另外说明。某些指令后可以接数据。关于选项, 指令和数据的格式, 分别参见第3.3, 3.5, 和3.6小节。

在下面, 这样的一系列输入称为命令区。特殊的命令区有结构区和基组区。

在命令行中所给的选项, 可以包含与当前程序有关的任何选项。例如, 在DF-LMP2-R12中, 可以有密度拟合选项, 局域选项, 显关联选项, 和/或阈值选项。另一种方法是在不同的指令中指定选项, 例如

```
DFIT,选项  
LOCAL,选项  
EXPLICIT,选项  
THRESH,选项
```

在这种情况下, 只有属于相应指令的选项才有效。因此, 如果EXPLICIT的选项指定给了DFIT指令, 就会导致错误。错误会在输入预检查的阶段检测到。

如前所述, 如果需要一个以上的输入行, 通常要使用花括号。对于单行命令, 如果下一命令或流程含有的指令与当前命令中的有效指令在名称上相同, 也需要使用花括号。

**注意:** DIRECT及有关的选项已经不再允许用命令行指定。

### 3.3 指令

指令用来为程序指定输入数据和特殊选项。它们由一个关键词开始, 后面接数据和/或选项。一般格式为:

```
DIRECTIVE, data,options
```

么输入检查程序可以接受它。STRICTCHECK=1: 输入检查程序不接受任何花括号外的指令。默认是STRICTCHECK=0, 它与以前版本的MOLPRO有最大的兼容性。

数据和选项的格式在随后的小节中介绍。必须总是在所有选项前给出数据。指令的例子有：OCC, CORE, CLOSED, WF, LOCAL, DFIT, ...

### 3.4 全局指令

某些指令可以在输入的任何位置给出，也就是说，既可以在命令区之外，也可以在命令区之内。若是在命令区内给出，指定的选项只对当前程序有效。但是，如果它们在命令区之外给出，就会作用全局，用于输入位置之后执行的所有程序。局域选项优先于全局选项。

以下指令既可以是局域的，也可以是全局的：

波函的定义：OCC, CORE, CLOSED, FROZEN, WF

阈值和选项：LOCAL, DFIT, DIRECT, EXPLICIT, THRESH, PRINT, GRID

如果这些选项在命令区之外给出，可以指定CONTEXT

DIRECTIVE, *data*, CONTEXT=*context*

例如，

```
OCC,3,1,1,CONTEXT=HF
OCC,4,1,2,CONTEXT=MCSCF
```

*context*可以是任何有效的命令名称（或其有效的同义词），但在程序内部会转化为以下之一：HF（Hartree-Fock和DFT），MC（MCSCF和CASSCF），CC（CCSD程序中执行的单参考关联方法），CI（MRCI程序中执行的多参考关联方法）。接下来指令将会用于这四种情况之一。可以指定几个*context*，用冒号隔开，例如：

```
CONTEXT=HF:CCSD
```

如果只给出了一个*context*（没有冒号），还可以用快捷方式指定*context*选项，即，为命令名称添加*context*后缀，例如：

```
OCC_HF,3,1,1
OCC_MCSCF,4,1,2
```

如果没有给出*context*，默认为HF。单参考方法（例如MP2，CCSD）的默认占据同HF所用的占据，多参考方法（例如RS2，MRCI）的默认占据同MCSCF所用的占据。

### 3.5 选项

选项的一般形式为NAME[=*value*]，其中的*value*可以是数值，表达式，或者字符串。多个选项之间可以用逗号或空格隔开。NAME 必须由[A-Z]的字符开头。如果

在COMMAND行中或在命令区的指令中给出选项，它们只对当前的程序有效（参见第3.3节）。如果在流程中给出选项，那么选项仅在流程以及当前流程所调用的流程中有效：只要流程一结束，就会恢复以前的选项。

选项也可以是一个关键词，像SYM或NOSYM。在这种情况下，根据关键词是否由NO开始，选项会打开或关闭。另一种方法是，例如逻辑选项，可以用NAME=ON或NAME=OFF进行设置或反设置。例如，SYM=OFF 等价于NOSYM。此外，YES和NO分别等同于ON和OFF。

### 3.6 数据

数据用一系列的数，表达式，或字符串定义，中间用逗号或空格分开。通常，数据的顺序是非常重要的。未定义的位置将被解释为零。字符串和变量必须由[A-Z]的字符开头。如果直接接在数值或变量前的+或--位于空格之后，将会被解释为符号而不是双值操作符。如果在这两个符号前后都没有空格，或者只在后面有空格，那么将会被解释为双值操作符。例如，

```
3 - 4 4 产生 [-1,4]
3-4 4    产生 [-1,4]
3 -4 4   产生 [3,-4,4]
3,-4 4   产生 [3,-4,4]
3, -4, 4 产生 [3,-4,4]
```

表达式（包括数值）也可以包含变量。

使用数值的例子有：结构输入和基组输入，LATTICE, OCC, CLOSED, CORE, 和WF指令。

在有些情况下，某些命令或指令需要几行数据。这时数据必须直接接在相应的命令或指令之后，并且必须用方括号括起来：

COMMAND,options

[data]

通常，数据的输入格式是MOLPRO风格的，也就是说，数值用逗号分开，并且可以使用变量和表达式。

如果通过使用外部文件来包含数据，那么数据的输入格式为自由格式：不需要逗号，但是不能使用变量和表达式。

### 3.7 表达式

在任何一个输入域中，数据都可以用表达式的形式输入。数值和变量是表达式的

特殊情况。表达式需要用Fortran风格书写，可以包含任何数量的嵌套括号。也可以用标准的内部函数（见下一节）。MOLPRO认识算术表达式和逻辑表达式。算术表达式的结果是（双精度）实数。所有的整数在程序内部也都转化为实数。逻辑表达式的结果要么是.TRUE.，要么是.FALSE.。 .TRUE.在程序内部存储为一(1.0)， .FALSE.存储为零(0.0)。表达式也可以包含任何数目的变量。

以下是可以进行的标准运算：

$expr + expr$	加
$expr - expr$	减
$expr * expr$	乘
$expr / expr$	除
$expr .OR. expr$	逻辑或
$expr .AND. expr$	逻辑和
$expr .XOR. expr$	异或
$.NOT. expr$	逻辑非
$expr .GT. expr$	大于
$expr .EQ. expr$	等于
$expr .LT. expr$	小于
$expr .GE. expr$	大于等于
$expr .LE. expr$	小于等于
$expr .NE. expr$	不等于
$expr **expr$	乘幂
$expr \wedge expr$	乘幂
$(expr)$	括号（无影响）
$-expr$	改变符号
$+expr$	保持符号（无影响）

### 3.8 内部函数

表达式可以包含以下的内部函数：

$ABS(expr)$	绝对值
$MAX(expr, expr, \dots)$	任意多个数值或表达式的最大值
$MIN(expr, expr, \dots)$	任意多个数值或表达式的最小值
$EXP(expr)$	指数



LOG( <i>expr</i> )	自然对数
LOG10( <i>expr</i> )	常用对数
SQRT( <i>expr</i> )	平方根
NINT( <i>expr</i> )	最近的整数
INT( <i>expr</i> )	截掉小数成为整数
SIN( <i>expr</i> )	正弦
COS( <i>expr</i> )	余弦
TAN( <i>expr</i> )	正切
ASIN( <i>expr</i> )	反正弦
ACOS( <i>expr</i> )	反余弦
ATAN( <i>expr</i> )	反正切
COSH( <i>expr</i> )	双曲余弦
SINH( <i>expr</i> )	双曲正弦
TANH( <i>expr</i> )	双曲正切
MOD( <i>expr1</i> , <i>expr2</i> )	求余: $expr1 - INT(expr1 / expr2) * expr2$

注意：在所有三角函数中，使用的或产生的角度单位都是度。

## 3.9 变量

### 3.9.1 设定变量

数据和结果可以存储在MOLPRO变量中。变量可以是字符串型，浮点数值型，或逻辑型，并且可以在输入的任何位置使用。

设定变量的语法为

`VARNAME1=expression [unit], VARNAME2=expression [unit]`

其中的unit可选。若变量未定义，假定是零。

变量可用于对不同的实际参数（例如几何结构，或者基函数的指数）运行同一个输入，以及用来存储和操作结果。数组是括号中有索引号的变量，例如，`var(1)`。数组`var`中的元素数量为`#var`。数组的长度可以用`CLEAR`指令重新设置为零，或者简单地修改`#var`。变量和变量数组可以用`SHOW`命令在输出的任何位置显示，整个变量表可以用`TABLE`命令产生。关于变量的更多细节，参见第八章。

### 3.9.2 字符串变量

使用字符串时要特别小心。为了避免不希望的结果，要么在设置字符串变量时加

上\$ 前缀，要么把字符串放在引号中。可能的形式有：

```
$name=string
name='string'
name=字符串变量
$name=字符串变量
```

例子：

```
string1='This is a string'  !定义一个字符串变量。引号中的文本不会转化为大写。
string2=string1             !把字符串变量string1赋值给新变量。
$string3=string1            !与上个例子相同。
$string4=mystring           !定义一个字符串变量。由于“mystring”未在引号中给出，
                             !因此将会转化成大写。
string5=mystring            !string5不是字符串变量，因为没有$。
```

上面的例子将会产生

```
SETTING STRING1      =   This is a string
SETTING STRING2      =   This is a string
SETTING STRING3      =   This is a string
SETTING STRING4      =   MYSTRING
VARIABLE MYSTRING UNDEFINED, ASSUMING 0
SETTING STRING5      =           0.00000000
```

关于字符串和字符串变量的更多信息，参见第8.3节。

## 3.10 流程

### 3.10.1 流程的定义

流程是一系列的命令和/或选项。可以在输入文件的任何位置定义流程，形式为

```
[PROC ]流程名={
命令区
指令
}
```

或者

```
PROC 流程名
命令区
指令
ENDPROC
```

为了避免不希望的结果，流程名必须不同于所有已知的命令名称。流程中千万不要包含geometry命令区。

注意，流程并不是在输入中遇到时就执行，而是只有当调用才会执行。流程的定义不能嵌套在别的流程中。流程可以包含流程的调用，直到10层的嵌套。

### 3.10.2 流程的调用

流程可以在输入的任何地方调用。除了用流程名替换命令名之外，与命令的语法相同（参见第3.2节）。

PROCEDURE
-----------

在流程的调用中不允许有选项。但是，可以用指令在流程中设定特殊的选项，这些选项会对流程中的指令之后的所有程序有效。在流程的执行结束后，恢复以前的全局选项。选项处理的等级如下：

全局选项

流程的选项

命令行选项

命令区中对指令所给的选项

实际使用的选项是最近一次设置的。因此，命令行或命令区中指定的选项优先于流程的选项，流程的选项优先于全局选项。



## 第四章

## 一般程序结构

这一章给出MOLPRO最重要功能的概述。新用户需要对这一部分讲述的方法和约定有基本的了解，特别是文件和记录的含义，以及对称性的使用。这一章主要集中在基本方面；每个命令详细的信息将在以后的章节中给出。关于命令和参数的信息也可以通过MOLPRO的帮助功能获得（参见第4.13节）。

### 4.1 输入结构

典型的MOLPRO输入具有如下结构：

```
***,title           ! 标题（可选）
memory,4,m          ! 内存指定（可选）
file,1,name.int      ! 命名永久的积分文件（可选）
file,2,name.wfu      ! 命名永久的波函文件（可选）

gprint,options       ! 全局打印选项（可选）
gthresh,options      ! 全局阈值（可选）
gdirect[,options]    ! 全局直接积分选项（可选）
gexpec,opnames       ! 单电子算符的全局定义（可选）

basis=basisname      ! 基的定义。如果不出现，使用cc-pVDZ
geometry={...}       ! 结构定义。

var1=value,var2=value,... ! 设定结构和/或波函定义中的变量

{command,options     ! 程序或者流程名称
  directive,data,option ! 命令的指令（可选）
  ...
}
```

---

! 结束输入 (可选)

如果给出了memory卡，那么它应当是（在可选的标题卡之后的）第一个卡。如果给出了任何file卡，它们应当紧跟其后。basis, geometry, gprint, gdirect, gthresh, gexpec, 以及变量定义的顺序是任意的。可以在一个程序后调用其它几个程序。在程序的调用之间还可以重新定义基组和/或分子结构；程序将会自动确认是否需要重新计算积分。

## 4.2 文件

MOLPRO使用三个连续的文本文件，分别为input文件，output文件，以及punch文件。punch文件是输出文件的简易形式，它包含了最重要的数据和结果，例如几何结构，能量，偶极矩，等。punch文件可以用独立的程序READPUN来处理，它用关键词选择特定的结果，并根据用户提供的格式产生排列好的表格。另外，可以利用最多达9个二进制的MOLPRO文件，程序通过它们的编号（1到9）可以简单地识别它们。它们默认是临时文件，通常由程序动态分配，但是用FILE命令可以把它们连接到永久文件中。每个文件都是直接存取，字可寻址的（word addressable；1个字通常是64位），并且在任意长度的记录中组织起来。每个记录的名称、地址和长度都保存在文件开始的目录中。

文件1是主文件，存有基组，几何结构，以及单电子和双电子积分。文件2默认是转存文件，用于存储波函数的信息，即：轨道，CI因子，以及密度矩阵。文件3是辅助文件，和文件2一起用于续算。通常文件1和2（以及3）被声明为用于续算的永久文件（参见FILE）。把波函数信息存储在文件2中是非常有用的，因为对于每个新的结构，积分文件都会被改写，而一个计算中的轨道和CI因子可以用作下一个相近几何结构计算的初始猜测。文件4至8用作草稿文件空间，例如用来分类积分，存储转换的积分和CI矢量。这些文件通常无需存为永久文件。

## 4.3 记录

记录的名称是正整数，通常的引用格式为记录.文件（*record.file*），例如，2100.2表示文件2上名为2100的记录。注意这些名称是非常任意的，它们的数值与记录在文件中的顺序无关。 $\leq 2000$ 的记录名专门为标准量保留（例如积分、特性等），在输入文件中千万不要使用它们，除非你知道你要做什么。需要记住的一些重要的默认记录有：

2100            RHF转存记录（闭壳层和开壳层）

2200	UHF转存记录
2140	MCSCF转存记录
4100	CPHF续算信息
5000	MCSCF梯度信息
5100	CP-MCSCF梯度信息
5200	MP2梯度信息
5300	Hessian续算信息
5400	频率续算信息
6300	域的续算信息

如果一个输入包含同一类型的几个波函计算，例如不同活性空间的几个MCSCF计算，则对同一类型的每个计算，记录编号加1。因此输入中第一个SCF计算的结果存储在转存记录2100.2中，第二个SCF计算存于2101.2记录，第一个MCSCF计算存于2140.2，第二个MCSCF存于2141.2，等等。注意，这些数字和输入中出现的顺序有关，而不是按照计算实际执行的顺序。如果用DO循环重复输入或部分输入，只要输入中的命令顺序保持不变，都会使每个计算从前一循环的相应轨道开始。例如，如果在第二轮循环中用IF/ENDIF结构跳过第一个SCF，那么第二个SCF仍旧使用2101.2记录。因此在大多数情况下，程序的默认设置是合适的，用户无需指定记录。

在续算后，如果SCF和MCSCF命令的数量和顺序保持不变，这一逻辑关系仍旧是正确的。因此，如果想在续算后跳过输入的某些部分，推荐使用IF/ENDIF结构或者GOTO命令，而不需要删除或者注释掉某些命令。如果由于某些原因导致无法进行，可以使用START和ORBITAL指令直接指定要用的记录。

只要可能，我们一般推荐使用程序的默认设置，因为这样的输入错误最少。

在完成每个程序步骤之后，MOLPRO打印每个文件中记录的摘要。

#### 4.4 续算

永久文件的信息能够在以后的计算中自动获取。这可以用RESTART指令控制。

#### 4.5 数据集的操作

可以用DATA命令截断文件，重命名或者复制记录。可以用MATROP 进行几种标准的矩阵操作，例如打印记录，矩阵的线性组合或者相乘，或者构成两个矩阵乘积的迹。

## 4.6 内存分配

根据用户需要，MOLPRO可以用MEMORY卡动态分配内存。因此不需要保存有不同内存大小的不同版本程序。如果忽略了MEMORY命令，程序将使用默认的内存大小，这依赖于所用的硬件以及程序的安装方法。注意，在Unix机器上，可以在 `molpro` 命令行中用标记 `-m` 设定默认的内存。

## 4.7 通过输入做循环

使用DO循环，可以对输入中的某些部分执行循环，这与在FORTRAN程序中非常类似。DO循环可以任意多层嵌套。这很方便使用，例如可以自动计算整个势能曲面。

## 4.8 对称性

MOLPRO只能使用阿贝尔点群对称性。对于有简并对称性的分子，必须使用阿贝尔子群，——例如对线型分子使用 $C_{2v}$ 或 $D_{2h}$ 。所用的对称群在积分输入中通过对称元素 $x$ ,  $y$ , 和 $z$ 的组合来定义，它们指定哪些坐标轴在相应的生成对称性操作下改变符号。适当的话（对于 $C_2$ 和 $C_{2h}$ 是必须的），选择 $z$ 轴作为唯一的轴通常是最好的。这种情况下的可能性示于表4.1。

表 4.1: 点群的对称生成元

生成元	点群
(空白卡)	$C_1$ (也即不使用点群对称性)
X (或Y或Z)	$C_s$
XY	$C_2$
XYZ	$C_i$
X,Y	$C_{2v}$
XY,Z	$C_{2h}$
XZ,YZ	$D_2$
X,Y,Z	$D_{2h}$

通常情况下，MOLPRO自动确定对称性，相应地转动和平移分子。但是，有时候为了固定分子的方位或者使用较低的对称性，必须直接指定分子的对称性。



表 4.2:  $D_{2h}$ 的不可约表示编号

$D_{2h}$		
No.	名称	函数
1	$A_g$	$s$
2	$B_{3u}$	$x$
3	$B_{2u}$	$y$
4	$B_{1g}$	$xy$
5	$B_{1u}$	$z$
6	$B_{2g}$	$xz$
7	$B_{3g}$	$yz$
8	$A_u$	$xyz$

表 4.3: 四维点群的不可约表示编号

$C_{2v}$			$C_{2h}$		$D_2$	
No.	名称	函数	名称	函数	名称	函数
1	$A_1$	$s, z$	$A_g$	$s, xy$	$A$	$s$
2	$B_2$	$x, xz$	$A_u$	$z$	$B3$	$x, yz$
3	$B_2$	$y, yz$	$B_u$	$x, y$	$B2$	$y, xz$
4	$A_2$	$xy$	$B_g$	$xz, yz$	$B1$	$xy$

表 4.4: 二维点群的不可约表示编号

$C_s$			$C_2$		$C_i$	
No.	名称	函数	名称	函数	名称	函数
1	$A'$	$s, x, y, xy$	$A$	$s, z, xy$	$A_g$	$s, xy, xz, yz$
2	$A''$	$z, xz, yz$	$B$	$x, y, xz, yz$	$A_u$	$x, y, z$

对于每个点群的不可约表示，编号为1至8。它们的顺序非常重要，在表4.2 – 4.4 给出。表中还给出了 $x$ ， $y$ ，和 $z$ 乘积的转换特性。 $s$ 表示各向同性函数，例如 $s$ 轨道，另外，它还可以给出了这些群的 $x^2$ ， $y^2$ ，和 $z^2$ 的转换特性。轨道或基函数的引用格式一般是序号.不可约表示编号，也就是说，3.2表示所用点群对称性的第二个不可约表示的第三个轨道。

## 4.9 定义波函

在需要这一信息的所有的程序模块中， $N$ 电子波函的总对称性在WF（wavefunction的缩写）卡定义，方式为：

WF, *nelec*, *irrep*, *spin*

或者用

WF, [NELEC=*nelec*], [SYMMETRY=*irrep*], [spin= *spin*], [CHARGE=*charge*]

其中，*nelec*是电子总数，*irrep*是不可约表示的序号，*spin*等于 $2 \times S$ ，其中 $S$ 是总的自旋量子数。除了*nelec*以外，也可以用*charge*，它定义分子的总电荷。例如，对于10个电子 $C_{2v}$ 对称性的计算，WF,10,3,0表示 $^1B_2$ 态，WF,10,1,2表示 $^3A_1$ 态。电荷也可以通过设定CHARGE变量来定义：

SET, CHARGE=*charge*

这一电荷将用于输入之后的所有能量计算。注意SET是必须的，因为CHARGE是系统变量（参见第8.4节）。

原则上，虽然每个程序单元都需要WF命令，但实际上很少需要给出。程序会记住WF卡的信息，因此典型情况下，在SCF计算中指定WF信息之后，后面的MCSCF或CI计算可以不指定WF；这也适用于续算。此外，*nelec*默认为核电荷的总和，*irrep*默认是1，*spin*默认是0或1；因此在很多情况下，根本不需要指定WF卡。

如果在命令输入区外给出WF指令，那么它作为全局指令处理，即，给定的值用于所有的随后计算。在全局WF指令中设定NELEC、SPIN、或SYMMETRY变量具有相同的作用。如果全局WF指令在变量的定义后给出，那么变量的值用WF指令的值替换。反之亦然，如果变量的定义在全局WF指令之后，那么新的变量值将在后面使用。注意，命令区的WF输入卡优先于全局WF指令或输入的变量。

## 4.10 定义轨道子空间

在SCF，MCSCF，和CI程序中，可能需要指定每个对称性中有多少个占据轨道

(或者在CI中有多少内部轨道), 以及这些轨道中哪些是芯壳层或闭壳层 (也就是在所有的CSF中都是双占据的)。这一信息由OCC, CORE, 和CLOSED卡提供, 方式为:

OCC, $m_1, m_2, \dots, m_8$ ; CORE, $co_1, co_2, \dots, co_8$ ; CLOSED, $cl_1, cl_2, \dots, cl_8$ ;

FROZEN, $fr_1, fr_2, \dots, fr_8$ ;

其中 $m_i$ 是不可约表示 $i$ 的占据轨道数 (包括芯轨道/冻结轨道和闭壳层轨道),  $co_i$ 是芯轨道数,  $cl_i$ 是闭壳层轨道数 (包括芯轨道)。通常,  $m_i \geq cl_i$ ,  $cl_i \geq co_i$ 。程序假定这些序号是从每个不可约表示最低的轨道开始。FROZEN只能用于MCSCF程序, 表示不做优化的冻芯轨道 (注意, 在旧版本MOLPRO程序中, 冻芯轨道表示为CORE)。

注意, 在SCF, MCSCF, 以及CI或CCSD程序中, OCC和CLOSED卡有略微不同的含义。在SCF和MCSCF中, 占据轨道是任何出现在CSF中的轨道。而在电子关联方法 (CI, MPn, CCSD等) 中, OCC表示在所有参考CSF中占据的轨道。在MCSCF中, FROZEN轨道在所有CSF中都是双占据, 并且冻结的 (也就是不做优化), 而CLOSED表示所有的双占据轨道 (冻结的加上做优化的)。在CI和CCSD程序中, 芯轨道是不做相关能处理的轨道, 闭壳层轨道是在所有的参考CSF中保持双占据的轨道。

OCC, CORE, 和CLOSED命令一般在每个有关的程序模块中都是必须的; 但是程序会记住最近一次使用的值, 因此如果不改变前面的轨道空间值的话, 这些命令就可以忽略。注意, 这一信息也保存在续算中。还要注意, 和使用WF信息一样, 程序对这些轨道空间也假定了合适的默认值。完整的细节参见相应的程序说明。

轨道空间也可以在命令区外定义, 作为全局指令处理, 即, 它用于随后所有的程序。对某些波函类型特定的空间, 可以通过为CONTEXT选项指定程序名来定义, 例如,

```
OCC,4,2,1,CONTEXT=MULTI
```

程序名也可以附加到指令名后, 并加上一个下划线。例如,

```
OCC_MULT,4,2,1
```

它等价于前一种形式。

命令区的局域输入优先于全局输入。

#### 4.11 选择轨道和密度矩阵(ORBITAL, DENSITY)

如在第4.3节概述的, 每个SCF或MCSCF计算的信息被存储在一个转存记录中。转存记录包含轨道, 密度矩阵, 轨道能量, 占据数, Fock矩阵以及其它信息, 如波函对称性等。此后的计算可以用ORBITAL和DENSITY指令, 分别从特定的记录访问轨道和密

度矩阵。这些输入卡在所有的程序中具有相同的结构。ORBITAL和DENSITY指令的一般格式如下。

```
ORBITAL[, [RECORD=record] [, [TYPE=orbtype] [, STATE=state] [, SYM[METRY]=symmetry]
[, SPIN=spin] [, MS2=ms2] [, [N] ELEC=nelec] [, SET=iset] [, OVL] [, NOCHECK]
[, IGNORE[_ERROR]]
```

```
DENSITY[, [RECORD=record] [, [TYPE=dentype] [, STATE=state] [, SYM[METRY]=symmetry]
[, SPIN=spin] [, MS2=ms2] [, [N] ELEC=nelec] [, SET=iset]
```

如果几个不同轨道集存储于同一个记录的话，其中（可选的）说明可用于选择特定的轨道。每个说明的含义如下：

*orbtype*            轨道类型。可以是以下之一

CANONICAL	正则或赝正则轨道；
NATURAL	自然轨道；
LOCAL	局域轨道；
LOCAL (PM)	局域Pipek-Mezey轨道；
LOCAL (BOYS)	局域Boys轨道；
PROJECTED	用于局域计算的投影虚轨道。

如果没有给出进一步的信息，则使用给定类型的最近计算的轨道。如果没有指定轨道类型，程序将会试着自动寻找最合适的轨道。例如在MRCI计算中，如果可能的话将优先使用NATURAL轨道；MRPT (CASPT2) 计算将首先寻找CANONICAL 轨道，局域计算将首先寻找LOCAL 轨道。因此在大多数情况下，不需要指定轨道类型。

*state*            以*istate.isym*的形式指定特定的态。例如，2.1表示对称性1中的第二个态。这可以用于在态平均CASSCF计算中对不同的态计算了密度矩阵或自然轨道的情况。如果没有给出，则使用态平均轨道。*isym*的定义是可选的；也可以用SYMMETRY 关键词定义。

*dentype*            密度类型。可以是以下之一

CHARGE	电荷密度；
SPIN	UHF自旋密度；
TRANSITION	跃迁密度矩阵。

默认是CHARGE。

*symmetry*        指定特定的态对称性。态的对称性也可以用STATE指定（见上）。

<i>spin</i>	自旋量子数，也即，0是单重态，1/2是双重态，1是三重态，等等。也可以用MS2定义。
<i>ms2</i>	2 <i>Ms</i> ，也即，0是单重态，1是双重态，2是三重态，等等。也可以用SPIN定义。
<i>nelec</i>	电子数量。
<i>iset</i>	设定轨道数量。轨道集按照被存储的顺序排序。

如果指定了OVL，将尽可能利用与前一轨道重叠的最大化获得初始轨道。默认只能用于前一轨道的基组维度不同于当前的情况。如果指定了OVL，即使是基组维度相同也使用这个流程，如果改变了收缩方案，这可能会有用。

如果指定了NOCHECK，就会跳过一些寻找正确轨道的一致性检查，并且忽略“ORBITALS CORRESPOND TO DIFFERENT GEOMETRY”一类的错误信息。

如果指定了IGNORE\_ERROR，可以强制MPn或三激发计算使用正则轨道之外的轨道。注意，这可能会导致无意义的结果。

如果给出了以上的任何选项，那么它们必须被严格遵守，也就是说，如果要求不能满足，程序就会终止。

例如：

```
ORBITAL,2100.2      ! 使用SCF轨道
ORBITAL,2140.2      ! 使用（态平均的）MCSCF轨道
ORBITAL,2140.2,CANONICAL ! 使用正则MCSCF轨道
ORBITAL,2140.2,NATURAL,STATE=2.1 ! 对于对称性1的第二个态，使用自然MCSCF轨道
```

## 4.12 控制程序的关键词总结

这是目前用到的用于控制程序的所有关键词的总结。每个模块都能识别更深层的关键词，这在各个关键词的章节中都有说明。关于使用下列命令的详细信息，参看以后的章节。

程序控制：

***	表示开始一个新的计算
MEMORY	分配动态内存
PUNCH	打开一个punch文件
FILE	把设备连接到永久文件
RESTART	恢复文件信息

INCLUDE	包含其它输入文件
BASIS	可用于定义默认基组
GEOMETRY	可用于指定结构
ZMAT	可用于定义Z-矩阵
PARALLEL	可用于控制并行
STATUS	检查程序步骤的状态
PRINT, GPRINT	控制全局打印级别
THRESH, GTHRESH	控制全局阈值
DIRECT, GDIRECT	直接计算积分和用于设定直接选项的标记
EXPEC, GEXPEC	控制期待值的计算
TEXT	打印文本
EXIT	停止运行
DO	控制do循环
ENDDO	结束do循环
IF	控制条件行为
ELSEIF	控制条件行为
ENDIF	结束IF区
GOTO	用于跳过部分输入，以及用于对输入做循环
LABEL	无作用
DATA	管理数据集
DELETE, ERASE	删除数据集
MATROP	执行矩阵操作
GRID	定义网格
CUBE	把数据转存到格点中
CARTESIAN	使用笛卡尔基函数
SPHERICAL	使用球谐基函数
USER	调用用户提供的子程序
---	输入的最后一行
变量:	
SET	设定变量（已作废）
SETI	把变量或数值设定为其倒数（已作废）
SETA	设定变量数组（已作废）
CLEAR	清除变量

CLEARALL	清除所有变量
GETVAR	从文件恢复变量
SHOW	显示变量的值
TABLE	打印表格
波函优化:	
INT	调用机器默认的积分程序。这是可选的，不必给出。
LSINT	调用自旋-轨道积分程序
SORT	调用双电子分类程序。该程序自动调用，不需要给出
CPP	计算芯极化势的积分
HF, RHF, HF-SCF, 或RHF-SCF	调用自旋限制Hartree-Fock程序（开壳层或闭壳层）
UHF或UHF-SCF	调用自旋非限制Hartree-Fock程序
DFT	调用密度泛函程序
KS, RKS	调用Kohn-Sham自旋限制密度泛函程序
UKS	调用Kohn-Sham自旋非限制密度泛函程序
MULTI, MCSCF, 或CASSCF	调用MCSCF/CASSCF程序
CASVB	调用CASVB价键程序
CI, MRCI, 或CI-PRO	调用内收缩MRCI程序
CIPT2	调用内收缩CIPT2程序
ACPF, AQCC	调用内收缩MR-ACPF程序
CEPA	调用单参考CEPA程序（闭壳层或者开壳层）
RS2, RS3	调用内收缩多参考微扰理论
RS2C	更快的内收缩多参考微扰理论程序
MP2	调用闭壳层MP2程序
MP3	调用闭壳层MP3程序
MP4	调用闭壳层MP4程序
CISD	调用闭壳层CISD程序
CCSD	调用闭壳层耦合簇程序
BCCD	调用闭壳层Brueckner CCD程序
QCI, QCSID	调用闭壳层二次组态相互作用程序
UCCSD	调用自旋非限制开壳层耦合簇程序
RCCSD	调用自旋限制开壳层耦合簇程序

FCI或FULLCI	调用基于行列式的完全CI程序
局域关联方法:	
LMP2	调用闭壳层局域MP2程序
LMP3	调用闭壳层局域MP3程序
LMP4	调用闭壳层局域MP4程序
LCISD	调用闭壳层局域CISD程序
LCCSD	调用闭壳层局域耦合簇程序
显关联方法:	
DF-MP2-R12	使用密度拟合的MP2-R12程序
DF-MP2-F12	使用密度拟合的MP2-F12程序
DF-LMP2-R12	使用密度拟合的局域MP2-R12程序
DF-LMP2-F12	使用密度拟合的局域MP2-F12程序
轨道操作:	
LOCALI	调用轨道局域化程序
MERGE	调用轨道操作程序
特性与波函分析:	
POP	调用布居分析程序
DMA	调用分布式多极分析程序
PROPERTY	调用特性程序
DIP	为 $h$ 添加偶极场
QUAD	为 $h$ 添加四极场
LATTICE	读取或关闭点电荷格子
梯度和几何优化:	
FORCES	调用梯度程序
OPTG	进行自动的几何优化
MIN	对某些参数进行能量最小化
PUT	把结构打印或者写到一个文件中
HESSIAN	计算Hessian
FREQUENCY	计算振动频率
MASS	定义原子质量
DDR	求解近似的非绝热耦合矩阵元

单参考耦合簇方法的命令名称QCISD, CCSD, LQCISD, LCCSD可以加上后缀(T), 进行三激发的微扰校正计算 (例如CCSD(T))。



HF, KS, MP2以及所有的局域关联方法可以加上前缀DF-, 调用密度拟合。

### 4.13 MOLPRO帮助

help命令可用于获得该命令的简短说明, 输入参数, 以及变量。语法是:

HELP,*set*,*name*,*[keys]*

其中*set*可以是COMMAND, VARIABLE, 或输入中的名称 (例如THRESH, PRINT, LOCAL, EOM, CFIT), *name*是参数的名称。如果*name*为空, 则显示所有设置的参数。还可以指定可选的*keys*请求特殊的信息 (例如short\_description, long\_description, default\_value, type, program)。如果*keys*没有给出, 那么假定是short\_description。

目前, help仅对有限数目的参数和命令有效。但是在不久的将来, 数据库还会扩充。



## 第五章

## 示范的例子

为了使新用户知道计算是多么的容易，这一部分介绍了一些非常简单的计算。

### 5.1 使用molpro命令

1. 对氢分子执行简单的SCF计算。输入直接敲入，输出被送到终端：

```
molpro <<!  
basis=vdz;  
geometry={angstrom;h1;h2,h1,.74}  
hf  
!
```

2. 相同的计算，但是数据来自文件h2.com。输出被送到h2.out。计算完成后，h2.pun文件返回到当前的目录中，h2.wf文件直接放到（默认的）\$HOME/wfu目录中：

```
molpro h2.com
```

h2.com的内容：

```
***,H2  
file,2,h2.wf,new;  
punch,h2.pun;  
basis=vdz;  
geometry={angstrom;h1;h2,h1,.74}  
hf
```

<http://www.molpro.net/info/current/examples/h2.com>

3. 同上，但是h2.wf文件发送到/tmp/wfu目录：

```
molpro -W /tmp/wfu h2.com
```

## 5.2 简单的SCF计算

第一个例子对H<sub>2</sub>O进行SCF计算，使用全部可能的默认值：

```
***,h2o                ! 标题
r=1.85,theta=104        ! 设定结构参数
geometry={0;            ! z矩阵结构输入
    H1,0,r;
    H2,0,r,H1,theta}
hf                      ! 闭壳层scf
```

[http://www.molpro.net/info/current/examples/h2o\\_scf.com](http://www.molpro.net/info/current/examples/h2o_scf.com)

在上面的例子中，使用了默认基组（VDZ）。我们可以用BASIS 指令修改默认的基组。

```
***,h2o cc-pVTZ basis   ! 标题
r=1.85,theta=104        ! 设定结构参数
geometry={0;            ! z矩阵结构输入
    H1,0,r;
    H2,0,r,H1,theta}
basis=VTZ               ! 使用VTZ基组
hf                      ! 闭壳层scf
```

[http://www.molpro.net/info/current/examples/h2o\\_scf.vtz.com](http://www.molpro.net/info/current/examples/h2o_scf.vtz.com)

## 5.3 几何优化

现在，我们还可以做几何优化，只需要简单地加入OPTG卡。

```
!examples/h2o_scf_opt_631g.com $Revision: 2002.10 $
***,h2o                ! 标题
r=1.85,theta=104        ! 设定结构参数
geometry={0;            ! z矩阵结构输入
    H1,0,r;
    H2,0,r,H1,theta}
basis=6-31g**          ! 使用Pople基组
hf                      ! 闭壳层scf
optg                   ! 进行scf几何优化
```

[http://www.molpro.net/info/current/examples/h2o\\_scf\\_opt\\_631g.com](http://www.molpro.net/info/current/examples/h2o_scf_opt_631g.com)

## 5.4 CCSD(T)

下面的任务用较大的基组（VTZ）进行CCSD(T)计算（对氧包含了 $f$ 函数，对氢包含了 $d$ 函数）。

```
***,h2o                ! 标题
```

```
r=1.85,theta=104      ! 设定结构参数
geometry={0;          ! z矩阵结构输入
    H1,0,r;
    H2,0,r,H1,theta}
basis=VTZ              ! 使用VTZ基组
hf                     ! 闭壳层scf
ccsd(t)                ! 进行ccsd(t)计算
```

[http://www.molpro.net/info/current/examples/h2o\\_ccsdt\\_vtz.com](http://www.molpro.net/info/current/examples/h2o_ccsdt_vtz.com)

## 5.5 CASSCF和MRCI

或许你想进行CASSCF计算以及随后的MRCI计算用于比较。在下面的CASSCF 和MRCI参考函数中，使用了全部价活性空间。

```
***,h2o                ! 标题
r=1.85,theta=104        ! 设定结构参数
geometry={o;            ! z矩阵结构输入
    h1,0,r;
    h2,0,r,H1,theta}
basis=vtz               ! 使用VTZ基组
hf                      ! 闭壳层scf
ccsd(t)                 ! 进行ccsd(t)计算
casscf                  ! 进行casscf计算
mrci                    ! 进行mrci计算
```

[http://www.molpro.net/info/current/examples/h2o\\_mrci\\_vtz.com](http://www.molpro.net/info/current/examples/h2o_mrci_vtz.com)

## 5.6 制表

现在你可能想把全部结果的概要打印到表格中。要这么做，必须先把计算的能量存储到变量中：

```
***,h2o                ! 标题
r=1.85,theta=104        ! 设定结构参数
geometry={o;            ! z矩阵结构输入
    h1,0,r;
    h2,0,r,H1,theta}
basis=vtz               ! 使用VTZ基组
hf                      ! 闭壳层scf
e(1)=energy              ! 把scf能量保存到变量e(1)中
method(1)=program        ! 把字符串“HF”保存到变量method(1)中

ccsd(t)                 ! 进行ccsd(t)计算
e(2)=energy              ! 把ccsd(t)能量保存到变量e(2)中
```

```

method(2)=program      ! 把字符串“CCSD(T)”保存到变量method(2)中

casscf                 ! 进行casscf计算
e(3)=energy            ! 把casscf能量保存到变量e(3)中
method(3)=program      ! 把字符串“CASSCF”保存到变量method(3)中
mrci                   ! 进行mrci计算
e(4)=energy            ! 把mrci能量保存到变量e(4)中
method(4)=program      ! 把字符串“MRCI”保存到变量method(4)中

table,method,e         ! 打印结果的表格
title,Results for H2O, basis=$basis ! 表格的标题

```

[http://www.molpro.net/info/current/examples/h2o\\_table.com](http://www.molpro.net/info/current/examples/h2o_table.com)

任务产生下面的表格:

Results for H2O, basis=VTZ

METHOD	E
HF	-76.05480122
CCSD(T)	-76.33149220
CASSCF	-76.11006259
MRCI	-76.31960943

## 5.7 流程

可以通过定义如下的流程SAVE.E来简化任务:

```

! $Revision: 2006.0 $
***,h2o                ! 标题

proc save_e            ! 定义流程save_e
if(#i.eq.0) i=0        ! 初始化变量i, 若它不存在的话
i=i+1                  ! 增加i
e(i)=energy            ! 把scf能量保存到变量e(i)中
method(i)=program      ! 把当前的方法保存到变量method(i)中
endproc                ! 结束流程

r=1.85,theta=104       ! 设定结构参数
geometry={o;           ! z矩阵结构输入
          h1,0,r;
          h2,0,r,H1,theta}
basis=vtz              ! 使用VTZ基组
hf                     ! 闭壳层scf

```

```

save_e                ! 调用流程, 保存结果

ccsd(t)               ! 进行ccsd(t)计算
save_e                ! 调用流程, 保存结果

casscf                ! 进行casscf计算
save_e                ! 调用流程, 保存结果

mrci                  ! 进行mrci计算
save_e                ! 调用流程, 保存结果

table,method,e        ! 打印结果的表格
title,Results for H2O, basis=$basis ! 表格的标题

```

<http://www.molpro.net/info/current/examples/h2o-proce.com>

该任务产生与前面相同的表格。

### 5.7.1 SA-CASSCF和MRCI

下面对OH基的基态和第一激发态进行态平均CASSCF及随后的MRCI计算。计算使用了全部价活性空间。

```

! $Revision: 2006.0 $
***,OH
include procedures
geometry={O;H,0,1.83}
set,symmetry=[2,3,1],spin=1 ! 2Pix, 2Piy, 和2Sigma+态
runmrci                    ! SA-CASSCF和MRCI

```

[http://www.molpro.net/info/current/examples/oh\\_runmrci1.com](http://www.molpro.net/info/current/examples/oh_runmrci1.com)

在输出结果的最后, 打印以下表格:

RESULTS						
METHOD	STATE	S	ENERGY	DIPX	DIPY	DIPZ
CASSCF	1.2	0.5	-75.41331789	0.0	0.0	0.67158730
CASSCF	1.3	0.5	-75.41331789	0.0	0.0	0.67158730
CASSCF	1.1	0.5	-75.24125256	0.0	0.0	0.69975340
MRCI	1.2	0.5	-75.55518444	0.0	0.0	0.66457191
MRCI+D	1.2	0.5	-75.56014871	0.0	0.0	0.66457191
MRCI+P	1.2	0.5	-75.55853208	0.0	0.0	0.66457191
MRCI	1.3	0.5	-75.55518444	0.0	0.0	0.66457191
MRCI+D	1.3	0.5	-75.56014871	0.0	0.0	0.66457191
MRCI+P	1.3	0.5	-75.55853208	0.0	0.0	0.66457191
MRCI	1.1	0.5	-75.39442202	0.0	0.0	0.70484623

MRCI+D	1.1	0.5	-75.40040680	0.0	0.0	0.70484623
MRCI+P	1.1	0.5	-75.39846312	0.0	0.0	0.70484623

这个计算对 $^2\Pi_x$ 和 $^2\Pi_y$ 都执行了MRCI。程序还不足以聪明到发现这两个态简并。不过，可以修改输入除掉这一不足。

```
! $Revision: 2006.0 $
***,OH
include procedures
geometry={0;H,0,1.83}
set,symmetry=[2,3,1],spin=1 ! 2Pix, 2Piy, 和2Sigma+态 (指定spin=1是可选的)
runcas ! 对所有三个态运行SA-CASSCF
set,symmetry=[2,1] ! 去掉Piy
runmrci ! 对2Pix和2Sigma+运行MRCI
```

[http://www.molpro.net/info/current/examples/oh\\_runmrci2.com](http://www.molpro.net/info/current/examples/oh_runmrci2.com)

这将产生

RESULTS						
METHOD	STATE	S	ENERGY	DIPX	DIPY	DIPZ
CASSCF	1.2	0.5	-75.41331789	0.0	0.0	0.67158730
CASSCF	1.3	0.5	-75.41331789	0.0	0.0	0.67158730
CASSCF	1.1	0.5	-75.24125256	0.0	0.0	0.69975340
MRCI	1.2	0.5	-75.55518444	0.0	0.0	0.66457191
MRCI+D	1.2	0.5	-75.56014871	0.0	0.0	0.66457191
MRCI+P	1.2	0.5	-75.55853208	0.0	0.0	0.66457191
MRCI	1.1	0.5	-75.39442202	0.0	0.0	0.70484623
MRCI+D	1.1	0.5	-75.40040680	0.0	0.0	0.70484623
MRCI+P	1.1	0.5	-75.39846312	0.0	0.0	0.70484623

你可能想扩大活性空间，把 $2\pi$ 轨道包含进来。这可以通过设定OCC变量实现。

```
! $Revision: 2006.0 $
***,OH
include procedures
geometry={0;H,0,1.83} ! 结构定义
set,symmetry=[2,3,1],spin=1 ! 2Pix, 2Piy, 和2Sigma+态
occ=[4,2,2] ! 4 sigma和2 pi占据
runcas ! 对所有三个态运行SA-CASSCF
set,symmetry=[2,1] ! 去掉Piy
runmrci ! 对2Pix和2Sigma+运行MRCI
```

[http://www.molpro.net/info/current/examples/oh\\_runmrci3.com](http://www.molpro.net/info/current/examples/oh_runmrci3.com)



要准确计算电子跃迁矩，1 $\delta$ 轨道的贡献也很重要。该轨道对称性为1 ( $\delta_{x^2-y^2}$ ) 和4 ( $\delta_{xy}$ )。为了使5 $a_1$ 轨道变为 $\delta$ ，我们必须在SCF计算中使用SYM指令。由于不能把这个指令插入到流程中，我们必须直接写SCF的输入。

```
! $Revision: 2006.0 $
***,OH
include procedures
basis=avqz                ! 使用aug-cc-pVQZ基组
geometry={0;H,0,1.83}     ! 结构定义
set,symmetry=[2,3,1],spin=1 ! 2Pix, 2Piy, 和2Sigma+态
occ=[5,2,2,1]            ! 4 sigma, 2 pi, 和1 delta占据
hf                        ! 进行scf计算
sym,1, 1,1,1,1,2         ! 对称性1的第5个轨道具有额外的对称性(= delta)
runcas                   ! 对所有三个态运行SA-CASSCF
set,symmetry=[2,1]        ! 去掉Piy
runmrci                  ! 对2Pix和2Sigma+运行MRCI
```

[http://www.molpro.net/info/current/examples/oh\\_runmrci4.com](http://www.molpro.net/info/current/examples/oh_runmrci4.com)

注意：这个计算相当昂贵！

### 5.7.2 MP2几何优化

下面的输入对水执行MP2几何优化。

```
! $Revision: 2006.0 $
***,H2O                    ! 标题
include procedures
basis=vtz                  ! 使用cc-pVTZ基组
geometry={0;                ! 水的Z矩阵
          H1,0,R;
          H2,0,R,H1,THETA}
R=0.96 Ang                 ! 初始键长
Theta=104                  ! 初始键角
optmp2                    ! 进行MP2几何优化
show,R,Theta               ! 显示优化的结构参数
```

[http://www.molpro.net/info/current/examples/h2o\\_optmp2.com](http://www.molpro.net/info/current/examples/h2o_optmp2.com)

在输出的最后，打印以下的结果总结。

```
RESULTS FOR BASIS=VTZ

METHOD  STATE    S      ENERGY    DIPX  DIPY    DIPZ
MP2(D)   1.1    0.0   -76.31865774    0.0   0.0    0.76152434

R / ANG              =          0.95906396
THETA / DEGREE       =          103.51638614
```

下一个计算在MP2级别做结构优化，随后在优化的结构上做MP4和CCSD(T) 计算。

```
!examples/h2o_optmp2.com $Revision: 2006.0 $
include procedures
geometry={0;                ! 水的Z矩阵
          H1,0,R;
          H2,0,R,H1,THETA}
basis=vtz                    ! 使用VTZ基组
R=0.96 Ang                   ! 初始键长
Theta=104                    ! 初始键角

hf
mp2
optg    ! 优化MP2能量
mp4     ! 在MP2的结构上做MP4单点计算
ccsd(t) ! 在MP2的结构上做CCSF(T)单点计算
```

[http://www.molpro.net/info/current/examples/h2o\\_optmp2\\_runccsd.com](http://www.molpro.net/info/current/examples/h2o_optmp2_runccsd.com)

在输出文件的最后，打印以下的结果总结。

```
RESULTS FOR BASIS=VTZ

METHOD      STATE      S      ENERGY
MP2(D)      1.1      0.0     -76.31865774
MP2         1.1      0.0     -76.31865774
MP3         1.1      0.0     -76.32273584
MP4(SDQ)    1.1      0.0     -76.32484084
MP4(SDTQ)   1.1      0.0     -76.33305159
MP2         1.1      0.0     -76.31865774
CCSD        1.1      0.0     -76.32454712
CCSD(T)     1.1      0.0     -76.33221602
CCSD[T]     1.1      0.0     -76.33240959
```

这里反复出现了MP2能量，因为在MP2，MP4，和CCSD(T)中都进行MP2计算。

### 5.7.3 DFT频率计算

以下的输入进行水的DFT/B3LYP结构优化和频率计算：

```
! $Revision: 2006.1 $
include procedures
geometry={0;                ! 水的Z矩阵
          H1,0,R;
```

```

      H2,O,R,H1,THETA}
R=0.96 Ang           ! 初始键长
Theta=104            ! 初始键角
basis=6-31g**        ! Pople 基组
$functional=b3lyp     ! 定义泛函 (可选, 因为b3lyp是默认的)
freqdft              ! 运行频率计算

```

[http://www.molpro.net/info/current/examples/h2o\\_freqdft.com](http://www.molpro.net/info/current/examples/h2o_freqdft.com)

结果为:

```

RESULTS FOR BASIS=6-31G**

METHOD      STATE      S      ENERGY      DIPX      DIPY      DIPZ
FREQ[KS/B3LYP]  1.1    0.0    -76.38101813    0.0    0.0    0.80039692
ZPE          1.1    0.0      0.02697203    0.0    0.0    0.80039692
E+ZPE        1.1    0.0    -76.35404610    0.0    0.0    0.80039692
HTOTAL       1.1    0.0    -76.35023712    0.0    0.0    0.80039692
GTOTAL       1.1    0.0    -76.37236272    0.0    0.0    0.80039692

```

频率, 强度, 以及更多的信息, 可以在输出文件里找到。

## 5.8 DO循环

现在你可能认为一个几何结构还不够。为什么不计算整个势能面呢? DO循环会使这个计算很容易。这里是一个例子, 计算了H<sub>2</sub>O的整个势能面。

```

! $Revision: 2009.1 $
***,H2O potential
symmetry,x           ! 使用cs对称性
geometry={
    o;                ! z矩阵
    h1,o,r1(i);
    h2,o,r2(i),h1,theta(i) }
basis=vdz            ! 定义基组
angles=[100,104,110] ! 角度列表
distances=[1.6,1.7,1.8,1.9,2.0] ! 键长列表
i=0                  ! 初始化计数器
do ith=1,#angles     ! 对所有角度H1-O-H2循环
do ir1=1,#distances  ! 对所有键长O-H1循环
do ir2=1,ir1        ! 对O-H2键长循环 (r1.ge.r2)
i=i+1                ! 增加计数器
r1(i)=distances(ir1) ! 对该结构保存r1
r2(i)=distances(ir2) ! 对该结构保存r2
theta(i)=angles(ith) ! 对该结构保存theta

```

```

hf;                                ! 进行SCF计算
escf(i)=energy                      ! 对该结构保存scf能量
ccsd(t);                           ! 进行CCSD(T)计算
eccsd(i)=energy                    ! 对该结构保存CCSD能量
eccsd(t)=energy                    ! 对该结构保存CCSD(T)能量
enddo                               ! 结束do循环ith
enddo                               ! 结束do循环ir1
enddo                               ! 结束do循环ir2
{table,r1,r2,theta,escf,eccsd,eccsd(t)} ! 产生结果的表格
head, r1,r2,theta,scf,ccsd,ccsd(t)   ! 修改表格的列标题
save,h2o.tab                       ! 把表格保存到文件h2o.tab中
title,Results for H2O, basis $basis ! 表格的标题
sort,3,1,2}                        ! 表格排序

```

[http://www.molpro.net/info/current/examples/h2o\\_pes\\_ccsd\(t\).com](http://www.molpro.net/info/current/examples/h2o_pes_ccsd(t).com)

这将产生以下的表格。

Results for H2O, basis VDZ

R1	R2	THETA	SCF	CCSD	CCSD(T)
1.6	1.6	100.0	-75.99757338	-76.20140563	-76.20403920
1.7	1.6	100.0	-76.00908379	-76.21474489	-76.21747582
1.7	1.7	100.0	-76.02060127	-76.22812261	-76.23095473
...					
2.0	1.9	110.0	-76.01128923	-76.22745359	-76.23081968
2.0	2.0	110.0	-76.00369171	-76.22185092	-76.22537212

还可以用DO循环对不同的方法重复输入。

```

! $Revision: 2006.0 $
***,h2o benchmark
$method=[hf,fci,ci,cepa(0),cepa(1),cepa(2),cepa(3),mp2,mp3,mp4,\
        qci,ccsd,bccd,qci(t),ccsd(t),bccd(t),casscf,mrci,acpf]
basis=dz                                ! 双zeta基组
geometry={o;h1,o,r;h2,o,r,h1,theta}   ! Z矩阵结构
r=1 ang, theta=104                     ! 结构参数
do i=1,#method                          ! 对所有请求的方法做循环
$method(i);                            ! 调用程序
e(i)=energy                             ! 保存该方法的能量
enddo
escf=e(1)                               ! scf能量
efci=e(2)                              ! fci能量
table,method,e,e-escf,e-efci           ! 打印结果的表格
!表格的标题:
title,Results for H2O, basis $basis, R=$r Ang, Theta=$theta degree

```

[http://www.molpro.net/info/current/examples/h2o\\_manymethods.com](http://www.molpro.net/info/current/examples/h2o_manymethods.com)

计算产生以下表格。

Results for H2O, basis DZ, R=1 Ang, Theta=104 degree			
METHOD	E	E-ESCF	E-EFCI
HF	-75.99897339	.00000000	.13712077
FCI	-76.13609416	-.13712077	.00000000
CI	-76.12844693	-.12947355	.00764722
CEPA(0)	-76.13490643	-.13593304	.00118773
CEPA(1)	-76.13304720	-.13407381	.00304696
CEPA(2)	-76.13431548	-.13534209	.00177868
CEPA(3)	-76.13179688	-.13282349	.00429728
MP2	-76.12767140	-.12869801	.00842276
MP3	-76.12839400	-.12942062	.00770015
MP4	-76.13487266	-.13589927	.00122149
QCI	-76.13461684	-.13564345	.00147732
CCSD	-76.13431854	-.13534515	.00177561
BCCD	-76.13410586	-.13513247	.00198830
QCI(T)	-76.13555640	-.13658301	.00053776
CCSD(T)	-76.13546225	-.13648886	.00063191
BCCD(T)	-76.13546100	-.13648762	.00063315
CASSCF	-76.05876129	-.05978790	.07733286
MRCI	-76.13311835	-.13414496	.00297580
ACPF	-76.13463018	-.13565679	.00146398

你甚至还可以做更奇特的事情，例如，使用存储为字符串变量的宏。示范的例子参见第8.5节的oh\_macros.com。



### 6.1 开始一个任务 (\*\*\*)

每个输入的第一个卡应当是：

\*\*\*, *text*

其中, *text*是任意的。如果文件1续算, *text*必须总是相同。这个卡的作用是重新设置所有程序的计数器等。如果忽略\*\*\*卡, *text*假定用默认值, 全是空格。

### 6.2 结束一个任务 (---)

输入文件的结束信号为文件结束, 或者用---卡。---卡之后的所有输入都被忽略。

在某些位置插入EXIT卡也可以停止任务。它也可以位于DO循环或IF块中。在这种情况下如果换用---卡, 将会导致错误, 因为程序无法找到ENDDO或ENDIF卡。

### 6.3 续算一个任务 (RESTART)

与MOLPRO92或者更早的版本相反, 目前版本的MOLPRO默认将尝试从所有的永久文件获得所有的信息。如果不需要续算, 则在FILE指令中使用NEW选项。下面讲到的RESTART指令仍可像在MOLPRO92中那样使用, 但通常不需要。

RESTART,  $r_1, r_2, r_3, r_4, \dots$ ;

其中 $r_i$ 指定续算哪个文件。在使用FILE卡之前, 这些文件必须已经分配。 $r_i$ 可以有两种格式:

a)  $0 < r_i < 10$ : 续算文件 $r_i$ 并恢复所有的信息。

b)  $r_i = \text{name.nr}$ : 续算文件 $nr$ , 但是在记录 $\text{name}$ 之前的截去。

如果所有的 $r_i = 0$ , 那么所有的永久文件都续算。但是, 如果至少有一个 $r_i$ 不等于零的话, 那么只有指定的文件续算。

例子:

RESTART;	用FILE卡分配的所有永久文件都续算 (默认)
RESTART,1;	只续算文件1
RESTART,2;	只续算文件2
RESTART,1,2,3;	续算文件1-3
RESTART,2000.1;	续算文件1, 并把记录2000之前截去

## 6.4 包含辅助输入文件 (INCLUDE)

INCLUDE,*file*[,ECHO];

把指定文件 $\text{file}$ 的内容插入到输入串中。对于大多数的执行, 给出的文件名直接在Fortran的OPEN语句中使用。如果指定了ECHO选项, 那么插入的文件按照正常方式返回给输出, 但是默认不打印它的内容。插入的文件自身也可以包含INCLUDE 命令, 嵌套最多可达10层。

## 6.5 分配动态内存 (MEMORY)

MEMORY, $n$ , $scale$ ;

设定动态内存的上限为 $n$ 个浮点字。如果 $scale$ 给定为K, 那么 $n$ 要乘以1,000; 如果 $scale$ 给定为M, 那么 $n$ 要乘以1,000,000。

注意: MEMORY卡必须在所有的FILE卡之前。

例子:

<u>MEMORY,90000</u>	<u>分配90,000个字的内存</u>
<u>MEMORY,500,K</u>	<u>分配500,000个字的内存</u>
<u>MEMORY,2,M</u>	<u>分配2,000,000个字的内存</u>

## 6.6 DO循环 (DO/ENDDO)

DO循环可以用DO和ENDDO命令构建。DO命令的一般格式与Fortran类似:

DO 循环变量=初值, 终值 [[,增量] [[,]单位]

其中初值, 终值, 和增量可以是表达式或变量。默认的增量是1。与Fortran相反, 这些



变量可以在循环中修改（使用应小心!）。例如：

```
DR=0.2
DO R=1.0,6.0,DR,ANG
IF (R.EQ.2) DR=0.5
IF (R.EQ.3) DR=1.0
....
ENDDO
```

将会对R的以下值执行循环：1.0，1.2，1.4，1.6，1.8，2.0，2.5，3.0，4.0，5.0，6.0 Å。下面的例子可以得到同样的效果：

```
RVEC=[1.0,1.2,1.4,1.6,1.8,2.0,2.5,3.0,4.0,5.0,6.0] ANG
DO I=1,#RVEC
R=RVEC(I)
....
ENDDO
```

可以嵌套直到20个DO循环。每个DO必须用自己的ENDDO结束。

如果DO变量已知，可以跳入DO循环。这对续算可能会有用，因为它允许继续中断的计算而不需要改变输入（所有的变量在续算中都被恢复）。

### 6.6.1 do循环的例子

（略）<sup>1</sup>

## 6.7 分支（IF/ELSEIF/ENDIF）

可以像在FORTRAN中那样，构造IF块和IF/ELSEIF块。

### 6.7.1 IF声明

IF块与在Fortran中的形式相同：

```
IF (逻辑表达式) THEN
声明
ENDIF
```

如果只需要一个声明，可以使用单行形式，除非声明是一个流程名称：

```
IF (逻辑表达式) 声明
```

<sup>1</sup>这部分提供的两个例子和前面重复。参见第5.8节。——译注。

ELSE和ELSE IF的使用与在Fortran中完全相同。IF声明可以任意嵌套。允许跳入IF或ELSE IF块。在这种情况下不执行测试；当遇到一个ELSE，那么控制延续到ENDIF之后。

逻辑表达式可以包含代数表达式或字符串的逻辑比较。例如：

```
IF(STATUS.LT.0) THEN
TEXT,An error occurred, calculation stopped
STOP
ENDIF
```

```
IF($method.eq.'HF') then
...
ENDIF
```

在前面的例子中，美元号和引号是可选的：

```
IF(METHOD.EQ.HF) then
...
ENDIF
```

### 6.7.2 GOTO命令

GOTO命令可用于跳过部分输入。它的一般形式为：

**GOTO**, *command*, [*n*], [*nrep*]

程序控制跳过第|*n*|次出现的*command*（默认： $n = 1$ ）。*command*必须是输入域第一个位置的关键词。如果*n*为正值，那么从当前位置向后搜索。如果*n*为负值，那么搜索从输入的顶端开始。GOTO命令最多执行*nrep*次。如果 $n < 0$ ，则*nrep*默认为1，否则默认为无穷大。我们建议在构造循环时不要使用GOTO 命令。

还可以跳到某个标记，使用：

**GOTO**, *label*

由于标记必须是唯一的，因此搜索总是从输入的顶端开始。*label*标记结束时必须用一个冒号。

### 6.7.3 标记 (LABEL)

**LABEL**

这是一个虚命令，有时与GOTO结合会很有用。

## 6.8 流程（PROC/ENDPROC）

流程的定义可以位于输入的顶端，或者在如下的INCLUDE文件中：

PROC 名称

声明

ENDPROC

还可以用这样的形式：

PROC 名称 [=] {声明}

对于后一种情况，需要左边的花括号（{）与PROC出现在同一行，但是声明可以包含几行。如果在随后的输入中发现名称是输入行第一个域中的命令，那么名称将会用声明替换。例子：

```
PROC SCF
IF (#SPIN.EQ.0.OR.MOD(SPIN,2).NE.MOD(NELEC,2)) SET,SPIN=MOD(NELEC,2)
IF (SPIN.EQ.0) THEN
    HF
ELSE
    RHF
ENDIF
ENDPROC
```

这也可以写作

```
PROC SCF={
IF (#SPIN.EQ.0.OR.MOD(SPIN,2).NE.MOD(NELEC,2)) SET,SPIN=MOD(NELEC,2)
IF (SPIN.EQ.0) THEN; HF; ELSE; RHF; ENDIF}
```

流程可以一直嵌套10层。在下面的例子中，SCF就是流程：

```
PROC CC
SCF
IF (SPIN.EQ.0) THEN
    CCSD
ELSE
    RCCSD
ENDPROC
```

注意：流程名称只有在输入行的第一个域中发现时才会被替换。因此它们绝不能用在单行的IF声明中；而要使用IF / ENDIF结构代替。

如果ECHO定义为流程的第一个声明，那么将打印当前和更低级别流程中被替代的命令。如果在主输入文件中定义了ECHO，那么将打印所有随后的流程。

某些重要的输入数据可以用变量传递给程序。例如占据方式，对称性，电子数，以及多重度，都可以用这种方式指定（细节参见第8.8节）。这使得可以更一般地使用流程。例如，假定在输入顶端定义了以下的流程：

```
PROC MRCI
IF (INTDONE.EQ.0) INT
IF (SCFDONE.EQ.0) THEN
SCF
ENDIF
MULTI
CI
ENDPROC
```

这个流程可以用于计算H<sub>2</sub>O的垂直电离势，如下：

```
R=1 ANG          ! 设定键长
THETA=104 DEGREE ! 设定键角

BASIS=VTZ        ! 定义基组

GEOMETRY          ! 结构输入区
0                ! Z矩阵
H1,O,R
H2,O,R,H1,THETA
ENDG             ! 结束结构输入
HF
MRCI             ! 用默认设置计算水的mrci能量
EH20=ENERGY      ! 把mrci能量保存到EH20变量中

SET,NELEC=9      ! 把电子数设为9
SET,SYMMETRY=2   ! 把波函对称性设为2
HF
MRCI             ! 计算H2O+ (2B2态) 的mrci能量

IPCI=(ENERGY-EH20)*TOEV ! 计算MRCI电离势，单位eV
```

注意：目前所有的变量都是全局的，也就是说，所有的流程通常都能识别变量，在流程中定义的所有变量随后也会在流程之外被识别。原因在于，把流程包含到内部输入层是在任务的一开始而不是在执行时；出于同样的原因，不可能用变量替代流程的名称，例如，不能使用这样的结构：

```
method=scf
$method      ! 不能这么用!
```

## 6.9 文本卡 (TEXT)

TEXT,xxxxxx

会在输出中打印xxxxxx。如果文本包含由美元号(\$)开头的变量，它们将会用实际值替换，例如：

```
r=2.1
text,Results for R=\$r
```

将会打印

```
Results for R=2.1
```

## 6.10 检查程序状态 (STATUS)

STATUS,[ALL|LAST|一些命令],[IGNORE|STOP|CRASH],[CLEAR]

这个命令检查并打印指定程序步骤的状态。一些命令可以是在当前任务中执行过的用于波函计算的命令列表。如果没有指定一些命令或指定了LAST，将会检查最后一步的状态。若给定了ALL，则检查所有的程序步骤。

如果给定了CRASH或STOP，在状态不正常时候，程序将会崩溃或者停止（默认是STOP）。如果给定了IGNORE，那么将忽略所有错误的状态。如果指定了CLEAR，那么已检查程序步骤的所有状态信息都将被擦除，于是程序在随后的状态检查中不会崩溃。

例子：

STATUS,HF,CRASH;	将检查最后HF-SCF步骤的状态，如果非正常（也就是不收敛）则崩溃。CRASH对于避免执行程序链中的下一个程序很有用。
STATUS,MULTI,CI,STOP;	将会检查最早MULTI和CI步骤的状态，如果未收敛则停止。
STATUS,RHF,CLEAR;	将为最后一次RHF清除状态标记。即使在RHF不收敛时也不会有操作。

注意，状态变量在续算中不会恢复。

程序默认将自动进行以下检查：

- 1) 如果轨道优化不收敛，并且产生的轨道用于随后的相关能计算，将会发生错误。这种出错退出可以用ORBITAL指令的IGNORE\_ERROR选项避免。
- 2) 如果CCSD|QCI|BCC|LMPn计算不收敛，依赖于此结果的随后程序步骤（如三激发，CPHF，EOM）将不会进行，并且提示出错。这可以在命令行中用NOCHECK选项避免。
- 3) 在结构优化或频率计算中，不收敛将会导致立即出错退出。

## 6.11 全局阈值 (GTHRESH)

在各个程序以外，可以用GTHRESH命令设置许多全局阈值（第一个字母G是可选的，但是应当使用，以避免与程序特定的THRESH卡相混淆）。语法为：

GTHRESH, *key1*=值1, *key2*=值2, ...

*key*可以是以下之一：

ZERO	数字零（默认1.d-12）
ONEINT	单电子积分的阈值（默认1.d-12，目前不再使用）
TWOINT	忽略双电子积分的阈值（默认1.d-12）
PREFAC	测试TWOINT前因子的阈值（默认1.d-14）
LOCALI	轨道局域化的阈值（默认1.d-8）
EORDER	局域化后，轨道重新排序的阈值（默认1.d-4）
<u>ENERGY</u>	<u>能量的收敛阈值（默认1.d-6）</u>
GRADIENT	MCSCF中轨道梯度的收敛阈值（默认1.d-2）
STEP	在MCSCF轨道优化中，步长的收敛阈值（默认1.d-3）
ORBITAL	在SCF程序中，用于轨道优化的收敛阈值（默认1.d-5）
CIVEC	MCSCFCI因子和CI参考矢量的收敛阈值（默认1.d-5）
COEFF	在CI和CCSD中，因子的收敛阈值（默认1.d-4）
PRINTCI	打印CI因子的阈值（默认0.05）
PUNCHCI	输出CI因子的阈值（默认99，不输出）
SYMTOL	寻找对称等价原子的阈值（默认1.d-6）
GRADTOL	梯度的对称性阈值（默认1.d-6）

THROVL            重叠矩阵所允许的最小本征值的阈值（默认1.d-8）

THRORTH        检测正交归一化的阈值（默认1.d-8）

## 6.12 全局打印选项（GPRINT/NOGPRINT）

在各个程序以外，可以用GPRINT命令设置全局打印选项（第一个字母G 是可选的，但是应当使用，以避免与程序特定的PRINT卡相混淆）。语法为：

GPRINT,*key1*[=值1],*key2*[=值2],...

NOGPRINT,*key1*,*key2*,...

通常，值可以忽略，但是大于0的值可用于调试的目的，在有些情况下会给出更多的信息。除了DISTANCE，ANGLES（默认=0），和VARIABLE之外，全部选项默认都不打印。NOGPRINT, *key*等价于PRINT, *key* = -1。*key*可以是以下之一：

BASIS            打印基的信息

DISTANCE        打印键长（默认）

ANGLES          打印键角信息（默认）。如果大于0，还打印二面角

ORBITAL         打印SCF和MCSCF的轨道

CIVECTOR        打印MCSCF的CI矢量

PAIRS            打印CI和CCSD中的电子对列表

CS               打印CI和CCSD中的单激发信息

CP               打印CI和CCSD中的对激发信息

REF              打印CI中的参考CSF及其因子

PSPACE          打印p-空间组态

MICRO           打印MCSCF和CI中的微迭代

CPU              打印详细的CPU信息

IO                打印详细的I/O信息

VARIABLE        每当变量设置或改变时，就打印变量（默认）

### 6.13 单电子算符和期待值 (GEXPEC)

进行期待值计算的算符在全局GEXPEC指令中用关键字定义。第一个字母G是可选的，但是应当使用，以避免与程序特定的EXPEC卡相混淆。EXPEC卡与GEXPEC卡有相同的形式。对于GEXPEC卡定义的所有算符，在以后的所有程序中都会计算期待值（如果可能）。

对于很多算符，可以使用类算符名称，例如，DM用于偶极矩，表示计算所有的三个分量DMX，DMY，和DMZ。或者，也可以单独计算每个分量。

一般格式如下：

```
[G]EXPEC,opname[,][icen,[x,y,z]],...
```

其中，

*opname*            算符名称（字符串），可以是类算符或者算符分量。

*icen*            Z-矩阵行号或者Z-矩阵符号，用于确定原点（不要指定*x, y, z*）。  
如果*icen*=0或者为空，则必须在*x, y, z*指定原点。

可以连用几个GEXPEC卡，也可以在一个卡中定义几个算符。

例子：

```
GEXPEC,QM          计算四极矩，原点位于(0,0,0)
GEXPEC,QM1         计算四极矩，原点位于中心1
GEXPEC,QM,O1       计算四极矩，原点位于原子O1
GEXPEC,QM,,1,2,3   计算四极矩，原点位于(1,2,3)
```

下面的表6.1总结了所有可用的算符。

对于对称算符（奇偶性=1）而言，期待值只能非零。其它的算符可用于计算跃迁量（自旋-轨道算符需要特殊的处理）。默认计算偶极矩。

表 6.1: 单电子算符及其分量

类名称	奇偶性	分量	说明
OV	1		重叠
EKIN	1		动能
POT	1		势能
DELTA	1		Delta函数
DEL4	1		$\Delta^4$



DARW	1		单电子Darwin项，即，具有适当因子并对所有原子求和的DELTA
MASSV	1		质量-速率项，即，具有适当因子的DELTA
REL	1		总的Cowan-Griffin相对论修正，即，DARW + MASSV
DM	1	DMX, DMY, DMZ	偶极矩
SM	1	XX, YY, ZZ, XY, XZ, YZ	二次力矩
TM	1	XXX, XXY, XXZ, XYY, XYZ, XZZ, YYY, YYZ, YZZ, ZZZ	三次力矩
MLTP $n$	1	所有唯一的 $n$ 阶笛卡尔乘积 $n$	多极力矩
QM	1	QMXX, QMYX, QMZZ, QMXY, QMXZ, QMYZ, QMRR=XX + YY + ZZ, QMXX=(3 XX - RR)/2, QMXY=3XY/2, 等	四极矩和 $R^2$
EF	1	EFX, EFY, EFZ	电场
FG	1	FGXX, FGYY, FGZZ, FGXY, FGXZ, FGYZ	电场梯度
DMS	1	DMSXX, DMSYX, DMSZX, DMSXY, DMSYY, DMSZY, DMSXZ, DMSYZ, DMSZZ	反磁屏蔽张量
LOP	-1	LX, LY, LZ	角量子算符 $\hat{L}_x$ , $\hat{L}_y$ , 和 $\hat{L}_z$
LOP2	1	LXLX, LYLY, LZLZ, LXLX, LXLZ, LYLY	角量子算符乘积的单电子部分。计算对称性组合 $\frac{1}{2}(\hat{L}_x\hat{L}_y + \hat{L}_y\hat{L}_x)$ 等
VELO	-1	D/DX, D/DY, D/DZ	速率
LS	-1	LSX, LSY, LSZ	自旋-轨道算符
ECPLS	-1	ECPLSX, ECPLSY, ECPLSZ	ECP自旋-轨道算符

### 6.13.1 计算期待值的例子

下面的任务计算H<sub>2</sub>O的偶极矩和四极矩。

```
! $Revision: 2006.0 $
***,h2o properties
geometry={o;h1,o,r;h2,o,r,h1,theta} ! Z矩阵结构输入
r=1 ang ! 键长
theta=104 ! 键角
gexpec,dm,sm,qm ! 计算偶极矩和四极矩
$methods=[hf,multi,ci] ! 对hf, casscf, 和mrci做计算
do i=1,#methods ! 对所有方法循环
$methods(i) ! 进行能量计算
e(i)=energy
```

```
dip(i)=dmz          ! 把偶极矩保存到变量dip中
quadxx(i)=qmxx      ! 保存四极矩
quady(i)=qmyy
quadzz(i)=qmzz
smxx(i)=xx          ! 保存二次力矩
smyy(i)=yy
smzz(i)=zz
enddo
table,methods,dip,smxx,smyy,smzz    ! 打印一次和二次力矩的表格
table,methods,e,quadxx,quady,quadzz ! 打印四极矩的表格
```

[http://www.molpro.net/info/current/examples/h2o\\_gexpec2.com](http://www.molpro.net/info/current/examples/h2o_gexpec2.com)

这个任务产生以下表格:

METHODS	DIP	SMXX	SMYY	SMZZ
HF	0.82747571	-5.30079792	-3.01408114	-4.20611391
MULTI	0.76285513	-5.29145148	-3.11711397	-4.25941000
CI	0.76868508	-5.32191822	-3.15540500	-4.28542917

METHODS	E	QUADXX	QUADYY	QUADZZ
HF	-76.02145798	-1.69070039	1.73937477	-0.04867438
MULTI	-76.07843443	-1.60318949	1.65831677	-0.05512728
CI	-76.23369821	-1.60150114	1.64826869	-0.04676756

### 6.13.2 计算相对论校正的例子

```
! $Revision: 2006.3 $
***,ar2
geometry={ar1;ar2,ar1,r} ! 结构定义
r=2.5 ang                ! 键长
{hf;                     ! 非相对论scf计算
expec,rel,darwin,massv} ! 用Cowan-Griffin算符计算相对论校正
e_nrel=energy            ! 把非相对论能量保存到enrel变量中
show,massv,darwin,erel   ! 显示各部分的贡献及其总和

dkroll=1                 ! 使用douglas-kroll单电子积分
hf;                      ! 相对论scf计算
e_dk=energy              ! 把相对论scf能量保存到e_dk变量中
show,massv,darwin,erel   ! 显示质量-速率和darwin的贡献及其总和
show,e_dk-e_nrel         ! 显示Douglas-Kroll的相对论贡献
```

[http://www.molpro.net/info/current/examples/ar2\\_rel.com](http://www.molpro.net/info/current/examples/ar2_rel.com)

这个任务在最后显示以下变量:

MASSV / AU	=	-14.84964285
------------	---	--------------

DARWIN / AU	=	11.25455679
EREL / AU	=	-3.59508606
E_DK-E_NREL / AU	=	-3.05862746



## 第七章

## 文件操作

### 7.1 FILE

FILE指令用于打开永久文件，永久文件可用于以后的续算。在MOLPRO94及其以后版本中的语法为：

FILE,*file*,*name*,*[status]*

其中*file*是逻辑MOLPRO文件序号(1-9)。*name*是文件名（将被转换成小写字母）。*status*可以是以下之一：

UNKNOWN	打开永久文件。如果存在，自动续算。这是默认的。
OLD	作用同UNKNOWN。如果该文件不存在，不会发生错误。
NEW	打开永久文件。如果已经存在，则清除，不续算。
ERASE	作用同NEW。
SCRATCH	打开临时文件。如果已经存在，则清除，不续算。任务完成后，文件不再存在。
DELETE	作用同SCRATCH。

注意，现在对所有的永久文件，默认设置是RESTART。所有的临时文件在需要时通常自动分配。I/O缓存分配到动态内存的顶端，缓存的大小会减少可用的内存。因此在第一个FILE卡之前必须出现MEMORY卡！

例子：

FILE,1,H2O.INT      分配永久文件1的名称H2O.INT。将恢复该文件中以前保存的信息。

FILE,2,H2O.WFU,NEW      分配永久文件2的名称H2O.WFU。将清除该文件中以前保存的信息。

注意在UNIX计算机上，文件名将被转化成小写。

## 7.2 DELETE

DELETE,*file1*, *file2*, ...

删除指定的文件。同FILE卡的指定，*file*表示逻辑MOLPRO文件序号。

## 7.3 ERASE

ERASE,*file1*, *file2*, ...

清除指定的文件。同FILE卡的指定，*file*表示逻辑MOLPRO文件序号。

## 7.4 DATA

DATA命令用于修改MOLPRO二进制文件。

UNIT      NPL的同义字（不再使用）。

RENAME,*rec1*,*rec2*

用于把*rec1*重新命名为*rec2*。*rec1*和*rec2*必须以*name.ifil*的形式给出，其中*ifil*是MOLPRO的二进制文件序号（NAME的同义字）。

TRUNCATE, *nen* 用于截断文件*nen-1*之后的记录（NEN的同义字）。

TRUNCATE, *rec* 用于截断*rec*前的记录。*rec*必须以*name.ifil*的形式给出，其中*ifil*是MOLPRO二进制文件的序号。

COUNT      NRE的同义字（目前不再使用）。

COPY,*rec1*,*rec2* 把记录*rec1*复制到*rec2*中。*rec1*和*rec2*必须以 *nam1.ifil1*, *nam2.ifil2*的形式给出。如果*nam2=0*，那么*nam2=nam1*。如果*nam1=0*，那么所有的记录从文件*ifil1*复制到文件*ifil2*。

## 7.5 分配punch文件 (PUNCH)

PUNCH,*filename*, [REWIND]

打开名为*filename*的punch文件。如果文件已经存在，则对其添加内容，除非指定了REWIND或NEW选项；在这种情况下，punch文件中所有以前的信息都被覆盖。对文件名的解释与机器有关，参见FILE。punch文件包含了所有的重要结果（结构，能量，偶极矩，跃迁矩，等）。它可以用独立的程序READPUN读取，产生用户指定格式的表格。

例子：

```
PUNCH,H2O.PUN
```

分配punch文件H2O.PUN。

注意在UNIX计算机中，文件名将转化成小写。

## 7.6 MOLPRO系统参数 (GPARAM)

GPARAM卡允许用户改变MOLPRO系统参数。只能由专家使用！

GPARAM,*option=value*,...

以下的选项可以按照任何顺序给出。

NOBUFF	如果出现，禁止系统缓冲。
LSEG	磁盘扇区长度。
INTREL	在每个实型字中整型字的数量（千万不要修改!）。
IBANK	内存库的数量。默认值2总是合适的。
IVECT	0=标量计算机，1=矢量计算机。
MINVEC	调用给mxmb的最小矢量长度。
LTRACK	缓冲区的页面大小（必须乘以 <i>lseg</i> ）。
LENBUF	积分缓冲区的长度（文件1）。
NTR	积分记录的长度（必须乘以3· <i>ltrack</i> ）。
LTR	CI中假定的磁盘扇区长度（默认值1是合理的）。
NCACHE	以字节为单位的机器高速缓存大小。
IASYN	如果非零，在CONVEX计算机上使用异步I/O。
MXMBLK	mxma的行/列块大小。

---

MXMBLN	mxma的连接块大小。
NCPUS	用于多任务处理的CPU最大数量。
MINBR1	每个处理器每秒浮点运算的最小数量。
MXDMP	用作完全功能转存文件的最高文件序号 ( $1 \leq \text{MXDMP} \leq 3$ )。

MXDMP选项仅用于专家！这将防止基函数和几何结构的信息写到比给定序号 *value* 高的转存文件中，有时对于平衡修正的几何优化会很有用。注意，有些功能在给出该选项时没有列出，如果输入有一点不正确，将会导致错误！



数据可以存储在变量中。根据表达式定义的类型，变量可以是字符串型，实型或逻辑型。任何不能作为表达式或变量识别的字符序列，全都作为字符串处理。在这一部分，我们只讨论实型和逻辑型变量。字符串型变量的详细讨论参见8.3节。可以在输入的任何地方使用变量，但是只能在特定程序的输入区之外设置变量。例如，如果HF的输入区使用了变量，那么它必须在HF{...}输入区之前设置。

MOLPRO自动把各种结果和数据存储到系统变量中（参见8.8.1节），用于进一步的处理。MOLPRO2002增加了一个新的功能，大多数系统变量是写保护的，用户不能重写。在计算任务开始前会自动检查输入，一旦在输入任务中设置了系统变量，任务将会终止，并打印出错信息。只有某些例外（参见8.4节），系统变量可以用SET命令修改（但不能简单地用NAME=值这样的语法）。注意，由于改变了SET命令的用法和语法，因此不再保留与MOLPRO92输入语法的兼容性。

## 8.1 设置变量

定义变量可以用：

*variable1=value1, variable2=value2, ...*

变量的定义通过输入卡第一个位置的等号来识别。例如，

THRESH,ENERGY=1.d-8,GRADIENT=1.d-5

没有定义变量；这里的ENERGY和GRADIENT是THRESH指令的选项。

变量可以有不同的类型：

数值	值是数值或表达式。值的一般形式为 表达式 [,] [单位]
----	----------------------------------

单位是可选的字符串，可用于为值加上单位。例如ANG[STROM]，DEGREE，HARTREE。表达式中未定义的变量假定是零（并同时定义为零）。

逻辑符 其值可以是.TRUE.，或.FALSE.（也可以用.T.和.F.）或逻辑表达式。在程序内部，.TRUE.存储为1，.FALSE.存储为0。

字符串 其值既可以是引号内的字符串，也可以是字符串变量。更多细节参见8.3节。

## 8.2 加索引的变量

变量可进行索引，但只能用一维索引。加索引的变量既可以单独定义，如

```
R(1)=1.0 ANG
R(2)=1.2 ANG
R(3)=1.3 ANG
```

也可以用方括号扩起来作为矢量值定义：

```
R=[1.0,1.1,1.2] ANG
```

还可以对范围的一部分进行定义，如，

```
R(1)=1.0 ANG
R(2:3)=[1.1,1.2] ANG
```

会得到与前面两个一样的结果。

每种变量元的类型依赖于所赋值的类型，也可以在一个变量中混合几种类型。如：

```
geometry={he}
hf
result=[program,energy,status.gt.0]
```

产生：

```
RESULT(1)      =    HF-SCF
RESULT(2)      =    -2.85516048  AU
RESULT(3)      =    TRUE
```

在这个例子中，变量PROGRAM，ENERGY，和STATUS是系统变量，由程序设置（参见8.4节）。

## 8.3 字符串变量

如在8.1节解释的，可以像设置其它变量那样设置字符串变量，形式为

变量 = '字符串'

变量 = 字符串变量

字符串必须包含在引号中。否则假定字符串是变量名，若此变量名未定义，假定是零。

另一种办法是，如果变量名前有一个美元号（\$），那么它所有的值假定是字符串。这可以是字符串变量，加引号的字符串，或者未加引号的字符串。注意，未加引号的字符串会转化为大写。还要注意的，如果字符串包含空格，那么引号是必需的。

例子：

```
$str=[a,b+4,'This is an example for strings']
```

产生

```
STR(1)      =      A
STR(2)      =      B+4
STR(3)      =      This is an example for strings
```

作为一般规则，只有美元号（\$）开头的字符串变量才可以用值替换（也有例外：在变量定义部分，在SHOW卡中，以及在IF卡的逻辑表达式中，美元号是可选的）。这是为了防止与变量名相同的命令被解释为变量。变量也可以出现在TEXT或TITLE卡，以及字符串中，但是在这些情况下，前面必须有\$。例如：

```
$METHOD=MCSCF
R=1.5
TEXT,$method results for R=$R Bohr
```

将会打印

```
MCSCF results for R=1.5 Bohr
```

用下面的方法，字符串变量可以和字符串或者其它的字符串变量连接。假定变量PROGRAM的值为MRCI。设定

```
METHOD='$PROGRAM+Q'
```

将会把METHOD设为MRCI+Q。另一种方法，如果还有值为Q的变量VERSION，可以写为

```
METHOD='$PROGRAM+$VERSION'
```

同样会把METHOD的值设为MRCI+Q。注意，在这些情况下必须使用引号。

不能对字符串的子串进行操作。

## 8.4 系统变量

如上面所提到的，大多数系统变量不能由用户写。但是有一些例外，可以用SET命令重新定义系统变量：

SET,变量 = 表达式 [,] [单位]

这可用于以下的变量：

CHARGE	分子的总电荷。
NELEC	电子数目。
SPIN	自旋量子数，定义为 $2 \cdot M\_S$ （整数）。
SCFSPIN	同SPIN，但仅用于HF。
MCSPIN	同SPIN，但仅用于MCSCF。
CISPIN	同SPIN，但仅用于MRCI。
STATE	需要优化的态。
MCSTATE	同STATE，但仅用于MCSCF。
CISTATE	同STATE，但仅用于MRCI。
SYMMETRY	态的对称性。
SCFSYM[METRY]	同SYMMETRY，但仅用于HF。
MCSYM[METRY]	同SYMMETRY，但仅用于MCSCF。
CISYM[METRY]	同SYMMETRY，但仅用于MRCI。
ZSYMEL	对称元。
LQUANT	线型分子的 $\Lambda$ 量子数。
OPTCONV	几何优化的收敛标准。
PROGRAM	最后的程序名。
CPUSTEP	最后程序步骤的CPU时间。
SYSSTEP	最后程序步骤的系统时间。
WALLSTEP	最后程序步骤的占用时间。
FOCKDONE	指示闭壳层Fock算符是否可用。
MAXBASIS	存储在转存文件的基组最大数量。如果达到了最大值，将用新的基组覆盖最后一个基组，因此前一个基组的所有信息（包括转存记录，等）将会丢失。默认为基组最大的可能数量（30），不能超过这个值。

## 8.5 使用字符串变量的宏定义

存储为代数表达式形式的字符串变量，如果前面有两个美元号（\$\$），会作为数值求解。例如：

```
string='a+b'
a=3
b=4
text,This is string $string which evaluates to $$string
```

## 将会打印

```
** This is string a+b which evaluates to 7
```

这可用于定义简单的宏，用在此后输入的任何位置。例如，

```
ECORR='ENERGY-ESCF' ! 定义宏
HF                ! 进行SCF计算
ESCF=ENERGY       ! 存储SCF能量到变量ESCF中
MULTI             ! 进行CASSCF计算
DEMC=$$ECORR      ! 存储CASSCF相关能到变量DEMC中
MRCI              ! 进行MRCI计算
DECI=$$ECORR      ! 存储MRCI相关能到变量DECI中
```

下面是一个高级使用宏和字符串变量的例子：

```
! $Revision: 2008.2 $
***,test for parser
text,This fancy input demonstrates how string variables and macros can be used
text
basis=vdz                ! 定义基组
geometry={0;H,0,r}       ! 定义结构（z-矩阵）

text,methods
$method=[rhf,2[casscf,2[mrci]]]
text,active spaces
$spaces=['3,1,1'],'3['4,2,2'],'3['5,2,2']']
text,symmetries
$symset=['1',2['1,2,3'],'1','2']]
text,weight factors for state averaged casscf
$weights=['1','1,1,1',2[' '],['1,0.5,0.5'],'2[' ']]
text,scf occupation
set,scfocc=[3,2[1]]
text,bond distance
r=1.85

hf
do i=1,#method           ! 对方法循环
mcooc=$$spaces(i)       ! 设定该运行的活性空间
set,symmetry=$$symset(i) ! 设定该运行的对称性
set,weight=$$weights(i) ! 设定该运行的权重
$method(i)              ! 现在运行该方法
e(i)='$energy'           ! 把能量保存到字符串中
dipol(i)='$dmz'          ! 把偶极矩保存到字符串中
enddo
```

```
table,method,spaces,symset,weights,e,dipol
title,Results for OH, r=$r, basis=$basis
head,method,spaces,symmetries,weights,energies,'dipole moments'
exit
```

[http://www.molpro.net/info/current/examples/oh\\_macros.com](http://www.molpro.net/info/current/examples/oh_macros.com)

## 8.6 加索引的变量（矢量）

变量可以加索引，但是只支持一维数组（矢量）。索引本身可以是变量。例如，  
METHOD(I)=PROGRAM

E(I)=ENERGY

是有效的变量定义，如果已经定义了I，PROGRAM，和ENERGY的话。索引可以嵌套任意的层。

数组的不同元素可以是不同的类型（可以是实型或逻辑型）。但是一个数组只能分配一个单位。字符串变量没有相应的值，不能与其它的变量类型混合。因此，一个给定的变量名只能要么用作字符串变量要么用作实型（逻辑型）变量。

用方括号可以很容易地定义矢量（数组）：

R=[1.0,1.2,1.3] ANG

它用三个元素定义了一个数组，它们可通过索引访问；例如R(2)的值为1.2 ANG。在左方括号之前可以给出重复次数：5[0]等价于[0,0,0,0,0]。方括号甚至可以嵌套：例如2[1,2,2[2.1,3.1]]等价于[1,2,2.1,3.1,2.1,3.1,1,2,2.1,3.1,2.1,3.1]。

数组可以从给定的位置添加，只需要输入要添加的元素；例如，

R(4)=[1.4,1.5] ANG

或者

R(4:)= [1.4,1.5] ANG

将会把上面的数组扩充到长度5。前面定义的值可以被覆盖。例如

R(2)=[1.25,1.35,1.45]

将会把上面的矢量修改为(1.0, 1.25, 1.35, 1.45, 1.5)。

如果等号左侧没有给出索引，那么已存在的同名变量将会用新的值替换，所有的旧值会消失。例如，

THETA=[100,110,120,130]

设置四个值。

THETA(1)=104 用新值替换THETA(1)；THETA(2:4)不变。

```
THETA=[140,150]
```

替换THETA的旧值；删除THETA(3:4)。

方括号也可以用于定义字符串数组，例如，

```
METHOD=[INT,HF,CASSCF,MRCI]
```

它们可以通过以下方式使用：

```
DO I=1,4
  $METHOD(I)
ENDDO
```

上面的输入等价于

```
INT
HF
CASSCF
MRCI
```

一个数组的当前长度可以在变量名前加#获得。例如在上面的例子中，#R 和#METHOD 的值分别为5和4。如果变量没有定义，将会返回零而不会产生错误。这可以用于测试变量是否存在，例如：

```
IF(#SPIN.EQ.0.AND.#NELEC.EQ.1) SET,SPIN=MOD(NELEC,2)
```

它定义了变量SPIN是否未知，以及NELEC是否是标量（一维）变量。

## 8.7 矢量操作

可以进行以下简单的矢量操作：

- 把一个矢量复制或添加到另一个矢量。例如，S=R把矢量R复制到矢量S。S(3)=R把R复制到S(3)，S(4)，…。S(#S+1)=R把矢量R添加到矢量S。还可以访问矢量连续的元素：S=R(2:4)把R的元素2到4复制到S(1)，S(2)，S(3)。注意，R(2:)表示元素R(2)到R(#R)，而R(2)表示R的一个元素。
- 矢量-标量操作：R=R\*2对R的每个元素乘2。数字2也可以换成标量（一维）变量或表达式，例如，R=R\*ANG把R所有的元素从Ångström转换到Bohr，Z=R\*COS(THETA)用元素Z(i) = R(i)\*COS(THETA)创建矢量Z。除了“\*”以外，还可以用所有其它的代数运算符。
- 矢量-矢量操作：如果A和B是同样长度的矢量，那么A × B也是同样长度的矢量。这里×代表任何代数运算符，对每一对相应的元素进行操作。例如，A + B把矢

量A和B相加， $A * B$ 把它们的元素相乘。注意对于后者，不是标量乘积。如果想用代数运算符连接两个不同长度的矢量，就会发生错误。

- 内部函数：假定 $THETA=[100,110,120,-130]$ 是角度的矢量（单位：度）。在这种情况下 $X=2*\cos(THETA)$ 也是矢量，包含 $THETA$ 每个元素的余弦乘以2，即， $X(i) = 2*\cos(THETA(i))$ 。 $MAX(THETA)$ 或 $MIN(THETA)$ 分别返回数组 $THETA$ 的最大值和最小值。矢量操作也可以嵌套，例如， $MAX(ABS(THETA))$ 返回数组 $ABS(THETA)$ 的最大值。

目前，矢量操作还不支持字符串变量。

## 8.8 特殊变量

### 8.8.1 程序设置的变量

程序预定义了许多变量。下面的变量可以用于在原子单位和其它单位之间进行转换：

```
EV=1.d0/27.2113961d0 HARTREE
KELVIN=1.d0/3.157733d5 HARTREE
KJOULE=1.d0/2625.500d0 HARTREE
KCAL=1.d0/627.5096d0 HARTREE
CM=1.d0/219474.63067d0 HARTREE
CM-1=1.d0/219474.63067d0 HARTREE
HZ=1.d0/6.5796838999d15 HARTREE
HERTZ=1.d0/6.5796838999d15 HARTREE
ANG=1.d0/0.529177249d0 BOHR
ANGSTROM=1.d0/0.529177249d0 BOHR
```

```
TOEV=27.2113961d0 EV
TOK=3.157733d5 K
TOKELVIN=3.157733d5 K
TOCM=219474.63067d0 CM-1
TOHERTZ=6.5796838999d15 HZ
TOHZ=6.5796838999d15 HZ
TOKJ=2625.500d0 KJ/MOL
```



```
TOKJOULE=2625.500d0 KJ/MOL
TOKCAL=627.5096d0 KCAL/MOL
TOA=0.529177249d0 ANGSTROM
TOANG=0.529177249d0 ANGSTROM
TODEBYE=2.54158d0 DEBYE
```

程序运行时设置的更多变量有：

INTYP	定义使用的积分程序。可以是INTS (Seward) 或INTP (Argos)。
INTDONE	如果对当前的结构完成了积分，值为 <code>.true.</code> 。
CARTESIAN	如果使用了笛卡尔基函数，设置为1。
SCFDONE	如果对当前的结构完成了SCF计算，值为 <code>.true.</code> 。
NUMVAR	当前定义的变量数目。
STATUS	最后一步的状态 (1=无错误，1=有错误或者不收敛)。
CHARGE	分子的总电荷。
NELEC	最后一个波函的电子总数。
SPIN	最后一个波函的自旋多重度减一。
ORBITAL	最后优化轨道的记录 (虽然设置，但是程序从不使用)。
LASTORB	最后优化轨道的类型 (RHF, UHF, UHFNAT, 或MCSCF)。
LASTSYM	最后优化轨道的波函对称性。
LASTSPIN	最后优化轨道的波函的 $2 * M_S$ 。
LASTNELEC	最后优化轨道的波函的电子数。
ENERGR(istate)	MRCI和CCSD计算中， <i>istate</i> 态的参考能量。
ENERGY(istate)	对于输入中指定的方法 (例如，HF, MULTI, CCSD(T), 或CCSD[T])， <i>istate</i> 态最后计算的总能量。
ENERGD(istate)	包含Davidson修正的 <i>istate</i> 态总能量 (仅在CI中设置)。
ENERGP(istate)	包含Pople修正的 <i>istate</i> 态总能量 (仅在CI中设置)。
ENERGT(1)	包含微扰三激发(T)修正的总能量 (仅在CCSD(T), QCI(T)中设置)。
ENERGT(2)	包含微扰三激发[T]修正的总能量 (仅在CCSD(T), QCI(T)中设置)。
ENERGT(3)	包含微扰三激发-t修正的总能量 (仅在CCSD(T), QCI(T)中设置)。

EMP2	在MP <sub>n</sub> , CCSD, BCCD, 或QCISD计算中保存MP2能量, 在MRPT2 (CASPT2) 计算中保存RS2能量。
EMP3	在MP3和MP4计算中保存MP3能量, 在MRPT3 (CASPT3) 计算中保存RS3能量。
EMP4	在MP4计算中保存MP4(SDQ)能量。MP4(SDTQ)能量保存在变量ENERGY中。
METHODC	字符串变量, 保存用于ENERGC计算的方法名称, 例如, CCSD, BCCD, QCI。
METHODT(1)	字符串变量, 保存用于ENERGT(1)计算的方法名称, 例如, CCSD(T), BCCD(T), QCI(T)。
METHODT(2)	字符串变量, 保存用于ENERGT(2)计算的方法名称, 例如, CCSD[T], BCCD[T], QCI[T]。
METHODT(3)	字符串变量, 保存用于ENERGT(3)计算的方法名称, 例如, CCSD-T, BCCD-T, QCI-T。
ENERGC	除去微扰三激发修正以外的总能量 (仅能在包含了微扰三激发修正的QCI或CCSD中设置)。
DFTFUN	在DFT或KS中密度泛函的总值。
DFTFUNS(ifun)	在DFT或KS中密度泛函第ifun个分量的值。
DFTNAME(ifun)	在DFT或KS中密度泛函第ifun个分量的名称。
DFTFAC(ifun)	在DFT或KS中乘以密度泛函第ifun个分量的因子。
DFTEXFAC	在KS中精确交换的因子。
PROP(istate)	计算态istate的特性。参见下面各种特性PROP的名称。
PROGRAM	在输入中指定的最后程序调用 (例如, HF, CCSD(T), 等)。
ITERATIONS	使用的迭代数量。如果不收敛, 或者达到了最大迭代数量, 则设为负值。
CPUSTEP	最后程序调用的用户CPU时间, 单位: 秒。
SYSSTEP	最后程序调用的系统CPU时间, 单位: 秒。
WALLSTEP	最后程序调用的占用时间, 单位: 秒。

特性的变量名与在EXPEC输入卡使用的相同。

OV	重叠
EKIN	动能
POT	势能

DELTA	Delta函数
DEL4	$\nabla^4$
DARWIN	相对论校正的Darwin项
MASSV	相对论校正的质量-速率项
EREL	总相对论校正
DMX, DMY, DMZ	偶极矩
XX, YY, ZZ, XY, XZ, YZ	二次力矩
XXX, XXY, XXZ, XYY, XYZ, XZZ, YYY, YYZ, YZZ, ZZZ	三次力矩
QMXX, QMYX, QMZZ, QMXY, QMXZ, QMYZ	四极矩
EFX, EFY, EFZ	电场
FGXX, FGYY, FGZZ, FGXY, FGXZ, FGYZ	电场梯度
D/DX, D/DY, D/DZ	速率
LSX, LSY, LSZ	单电子自旋-轨道
LL	总角动量的平方 $L^2$
LX, LY, LZ	电子角动量
LXLX, LYLY, LZLZ, LXLZ, LYLZ	双电子角动量

默认只定义和计算偶极矩。如果其它特性被EXPEC卡请求，那么它们的值仅存储在变量中。如果计算了一个以上的态（例如态平均的MCSCF），那么返回相应的数组PROP(istate)。如果对一个以上的中心计算特性，那么名称中会加入中心序号，例如EFX1, EFX2, 等。

如果计算跃迁特性，它们的值存储在前缀为TR的相应变量中，例如，跃迁偶极矩是TRDMX, TRDMY, TRDMZ。如果计算两个以上的态，那么指数为 $(i-1) * (i-2)/2 + j$ ，其中 $i > j \geq 1$ 是态的序号。在态平均计算中，对所有的态对称性连续计数。

例如，在下面的态平均MCSCF中：

```
MULTI;WF,14,1,0;STATE,3;WF,14,2,0;STATE,2;WF,3,0
```

态的计数为：

$i$	1	2	3	4	5	6
对称性	1	1	1	2	2	3
对称性中的根	1	2	3	1	2	1

### 8.8.2 程序认识的变量

以下所列的所有变量都会被程序检查，但是不会设置（除了NELEC和SPIN以外）。如果它们没有被用户定义，程序将使用它们的内部默认值。

SCF程序认识的变量有：

CHARGE	分子的总电荷（给出可以替代NELEC）。
NELEC	电子总数。
SPIN	自旋多重度减一。
SCFSYM[METRY]	波函对称性。
SYMMETRY	同SCFSYMM；只能用于SCFSYMM未出现的情况。
SCFOC[C]	SCF每个对称性的占据轨道数。
SCFCL[OSED]	SCF每个对称性的闭壳层轨道数。
SCFORB	SCF中保存轨道的记录。
SCFSTART	SCF中使用的初始轨道记录。

MCSCF程序认识的变量有：

CHARGE	分子的总电荷（给出可以替代NELEC）。
NELEC	电子总数。
MCSYM[METRY]	波函对称性。对于态平均计算可以是数组。
SYMMETRY	同MCSYMM；只能用于MCSYMM未出现的情况。
MCSPIN	自旋多重度减一。对于态平均计算可以是数组，但是不同的自旋多重度只能用于行列式CASSCF中。如果只定义了一个值，则用于所有的电子态。
SPIN	同MCSPIN；只能用于MCSPIN未出现的情况。
MCSTATE	在MCSCF中每个对称性的态数。
STATE	同MCSTATE；只能用于MCSTATE未出现的情况。
WEIGHT	由SYMMETRY和STATE定义的所有态的权重因子。
LQUANT	对于线型分子，由SYMMETRY和STATE定义的每个态的 $L_z^2$ 本征值。
MCSELECT	用来选择组态的记录和选择阈值。
SELECT	同MCSELECT；只能用于MCSELECT未出现的情况。
MCRESTRCT	用于定义占据限制。
RESTRICT	同MCRESTRCT；只能用于MCRESTRCT未出现的情况。
CONFIG	如果设置为.true.或者1，触发CSF的使用。
MCOC[C]	每个对称性的占据轨道数。
OCC	同MCOCC；只能用于MCOCC未出现的情况。
MCCL[OSED]	对每个对称性优化的闭壳层轨道数。

CLOSED	同MCCLOSED; 只能用于MCCLOSED未出现的情况。
MCFROZEN	每个对称性的冻芯轨道数。
FROZEN	同MCFROZEN; 只能用于MCFROZEN未出现的情况。
MCSTART	初始轨道的记录。
COREORB	冻芯轨道的记录。
MCORB	保存优化轨道的记录。
MCSAVE	保存CI波函的记录 (类似于MCSCF的SAVE卡)。

CI/CCSD程序认识的变量有:

CHARGE	分子的总电荷 (给出可以替代NELEC)。
NELEC	电子总数。
SPIN	自旋多重度减一。
CISYM[METRY]	波函对称性。如果是数组, 只使用SYMMETRY(1)。
SYMMETRY	同CISYMM; 只能用于CISYMM未出现的情况。
CISTATE	CI中的态数。
STATE	同CISTATE; 只能用于CISTATE未出现的情况。
CISELECT	用于选择组态的记录。
SELECT	同CISELECT; 只能用于CISELECT未出现的情况。
CIRESTRICT	定义占据限制。
RESTRICT	同RESTRICT; 只能用于RESTRICT未出现的情况。
CIOCC[C]	每个对称性的占据轨道数。
OCC	同CIOCC; 只能用于CIOCC未出现的情况。
CICL[OSED]	每个对称性的闭壳层轨道数。
CLOSED	同CICLOSED; 只能用于CICLOSED未出现的情况。
CICO[RE]	每个对称性的芯轨道数。
CORE	同CICORE; 只能用于CICORE未出现的情况。
CIORB	CI中使用的轨道记录。
CISAVE	用于保存CI波函的记录 (类似于CI的SAVE卡)。
CISTART	用于从前面CI波函续算的记录 (类似于CI的START卡)。

DFT/KS程序认识的变量有:

DF(ifun)或DFTNAME(ifun)

密度泛函第ifun个分量的名称。

DFTFAC(ifun)      乘以密度泛函第ifun个分量的因子。  
 DFTEXFAC          在KS中乘以精确交换的因子。

对态平均MCSCF使用这些变量的例子（注意系统变量只能用SET命令修改，参见8.4节）：

SET,NELEC=9          定义电子数。  
 SET,SPIN=1          定义双重态波函。  
 SET,SYMMETRY=[1,2,3] 定义态平均计算的波函对称性。  
 SET,STATE=[2,1,1]   定义每个对称性中进行平均的态数。  
 WEIGHT=[2,2,1,1]   定义以上四个态的权重。  
 OCC=[5,2,2]        每个对称性的占据轨道数。  
 CLOSED=2           对称性1中闭壳层轨道数。  
 MCOORB=3100.2      优化轨道的记录。  
 MULTI               用以上的参数进行mcscf计算。

注意：设定变量NELEC, SPIN, 或SYMMETRY, 与在全局WF指令中给定选项具有相同的作用。如果在变量定义以后给出全局WF指令，那么变量值用WF指令中给定的值替换。反之亦然，如果全局WF指令之后是变量定义，那么将在后面使用新的变量值。注意命令区中的WF输入卡优先于全局WF指令或输入变量。

## 8.9 显示变量

变量或者表达式的结果可以用SHOW和TABLE在输出中显示。

### 8.9.1 SHOW命令

SHOW命令的一般形式如下：

SHOW[*ncol*,*format*],*expression*

其中的*expression*可以是表达式或者变量，*ncol*是在每行打印的多少个值（默认6），*format*是格式（默认6F15.8）。这可用于打印矩阵形式的矢量。*ncol*和*format*的说明是可选的。假设E是矢量：

SHOW,E              用默认设置打印E。  
 SHOW[n],E          每行打印E的n个元素；（如果n>6，则需要一行以上，但是无论在任何情况下，新的一行都从n个元素之后开始）。  
 SHOW[n,10f10.4],E 以给定的格式打印E，在n个元素以后强制换行。

注意格式的总长度不能超过100个字符（左边总需要30个字符的空白）。

也可以用通配符格式，可以更容易地显示多个变量：

`SHOW, qm*, dm*`

显示名称由QM和DM开始的全部变量。注意在\*之后不能出现文字，也就是说，通配符格式不如UNIX命令中的通配符通用。

关于其它可以把结果列表的命令，参见TABLE命令。

## 8.10 清除变量

删除变量可以用

`CLEAR, name1, name2, ...`

可以像在SHOW中那样使用通配符，例如，

`CLEAR, ENERG*`

将清除名称以ENERG开始的所有变量。清除所有变量可以用

`CLEARALL`

矢量的长度可以简单地用重新定义的长度说明来截断：`#R=2`把数组R的长度截断到长度2。更高的元素不再可用（但是可以重新定义）。设定`#R=0`等价于命令`CLEAR, R`。

## 8.11 从外部文件读取变量

变量可以从外部文件读取，使用

`READVAR, 文件名`

这样的文件可以保存，例如用几何优化程序，在以后想恢复到某个优化的结构时，可以重新使用。文件名的输入格式与通常的输入文件相同。





## 第九章

## 制表与画图

### 9.1 制表

变量可以用表格的形式打印，命令为

`TABLE, var1, var2, ...`

每个变量的值打印在一列，因此所有使用的变量必须对同一个范围进行定义，相应的元素应当全部属于该范围。例如，如果一个计算中对每个几何结构存储了`R(i)`，`THETA(i)`，`ECI(i)`，打印这些数据可以简单地使用

```
TABLE, R, THETA, ECI
```

默认的行数等于第一个变量的元素数量。但是这可以用`RANGE`子命令改变。

表格的前十列可以包含字符串变量。例如，

```
hf; etot(1)=energy; method(1)=program; cpu(1)=cpustep
ccsd; etot(2)=energy; method(2)=program; cpu(2)=cpustep
qci; etot(3)=energy; method(3)=program; cpu(3)=cpustep
table, method, etot, cpu
```

将在表格的第一列，第二列和第三行分别打印SCF，CCSD和QCI的结果。其它关于使用字符串变量和表格的事项，参见例子[h2o.tab.com](http://h2o.tab.com)和[oh.macros.com](http://oh.macros.com)。

表格的格式可以用以下命令修改，它们可以直接在TABLE卡之后（以任何顺序）给出：

`HEADING, head1, head2, ...` 指定每一列的标题。默认的标题使用变量名。

FORMAT, <i>format</i>	用FORTRAN风格指定每一行的格式。 <i>format</i> 必须加引号。通常，程序可以自动确定合适的格式，这依赖于要打印的数据的类型和大小。
FTYP, <i>typ1, typ2, typ3, ...</i>	修改格式的简化形式。对每一列给出类型（A, F或D）。（通常使用自动判断的默认设置。）
DIGITS, <i>dig1, dig2, dig3, ...</i>	给出每一列在小数点后要打印的小数位数（通常使用自动判断的默认设置）。
TYPE	指定表格的数据格式。默认为TEXT，给出普通的文本格式。其它可选的有CSV（用逗号隔开，适用于电子数据表格），LATEX（L <sup>A</sup> T <sub>E</sub> X的table环境），MATHEMATICA（MATHEMATICA代码，把表格赋值给数组），MATLAB（MATLAB代码，把表格赋值给数组），MAPLE（MAPLE代码，把表格赋值给数组），HTML（HTML的TABLE结构），以及XML（包含树结构表格的XML文档。实际的结构是XHTML）。
SAVE, <i>file, status</i>	指定写入表格的文件。如果 <i>status</i> 是NEW，文件倒回重写，否则在后面添加。如果 <i>file</i> 具有扩展名txt, csv, tex, m, mpl, html, xml之一，并且没有指定TYPE命令，那么类型将被设置为与扩展名相符的类型。如果忽略了 <i>file</i> ，则自动产生文件名，形式为 <input/> .tablen.ext: <i>input</i> 是输入文件的基本名称（若从标准输入运行，则是molpro）； <i>n</i> 是序列号，每产生一次表格就增加1； <i>ext</i> 是文件格式相应的扩展名，例如txt, html, 等。
TITLE, <i>title</i>	指定一行标题（可以连用多个TITLE卡）。注意，如果SAVE命令在TITLE卡之前给出，那么标题只能显示在SAVE文件中。
SORT, <i>col1, col2, ...</i>	对给定的列，依照值增加的顺序对行分类。列按照指定的顺序存储。
PRINT, <i>key1, key2, ...</i>	指定打印选项（TABLE, HEADING, TITLE, WARNING, FORMAT, SORT）。默认是前三个打印，后三个不打印。
NOPRINT, <i>key1, key2, ...</i>	禁止打印给定的关键字。
NOPUNCH	不把数据写入到punch文件中（默认写入数据）。
RANGE, <i>start, end</i>	对要打印的变量，指定起始和结束的编号。

STATISTICS

还打印数据列的线性回归和二次拟合。

## 9.2 画图

[PLOT[,*col1,col2,...*][,*options*]

用表格作为数据，构造绘图程序的输入。PLOT是TABLE的子命令，因此必须接在TABLE之后，或前一节列出的TABLE的任何有效子命令之后。在一个TABLE中可以包含一个以上的PLOT命令，每一个请求产生一个新的绘图。但是，PLOT必须出现在所有其它TABLE子命令之后。

*col1,col2,...*是要绘图的表格列名称。它们必须是TABLE命令中所给名称的精确子集。第一列作为横轴，余下的值将作为纵轴，绘制相对于横轴的图。如果没有指定列，那么绘制整个表格；若指定了单列，则它用作横轴，表格其余的列将作为纵轴绘图。*options* 可以从下面选择：

CMD=*unix\_plot\_command*

*unix\_plot\_command*由用于启动绘图程序所需的系统命令构成，之后可以接任何需要的选项。这些内容通常应当全部放入引号中，并保持小写字母形式。默认是'xmgrace'。目前只支持使用数值型数据的xmgrace, grace, 和gracebat程序。

FILE=*plotfile*

绘图程序默认的输入文件保存在.tablen.plotm.agr, 其中*m*是自动产生的序列号。绘图文件名称可以用FILE选项修改。

INTERACTIVE

默认图形不显示在屏幕上，而是所有的图形文件保存在给定的文件里。通过给定INTERACTIVE 选项，可以交互式地启动绘图程序。

TYPE=*type*

若指定了TYPE，那么*type*应设定为pdf, svgf, pngf, jpegf, 或eps中的一个。结果是对绘图输入文件执行gracebat程序，产生所需格式的图形输出文件。这一功能依赖于gracebat程序是否可用，以及是否支持所需的输出格式（例如，目前仅在Mac OS X 下支持pdf）。

NOSPLINE

禁止数据点的三次样条插值。

NSPLINE,*number*

插值的点数（默认20）。

以下的附加指令可以在PLOT指令之前给出：

`COLOR,icolor1,icolor2,...` 彩色图用于列1, 2, ...; 零表示使用默认值 (黑, 蓝, 红, 绿, 依次循环)。

`SYMBOL,isymb1, isymb2,...` 用于列1, 2, ...的符号类型; -1表示没有符号; 零表示使用默认值。

### 9.3 双原子分子势能曲线分析

对于双原子分子, 在表格包含一个或多个势能函数, 其中第一列是Bohr或Ångstrom单位的键长的情况下, 可以计算光谱常数, 使用命令

`DIATOMIC[,DEGREE=n][,MASS=m][,PRINT=p]`

数据用 $n$ 次多项式拟合 (默认是点数减一, 即内插), 光谱常数用约化质量 $m$  (原子量单位) 计算。注意, 通过使用子命令**RANGE**, 可以限定使用哪些键长。

### 10.1 结构说明

分子结构可以用标准Z矩阵形式，或XYZ形式。结构说明用以下格式给出：

[SYMMETRY, *options* ]

[ORIENT, *options* ]

[ANGSTROM ]

geometry={

原子说明

}

GEOMETRY必须出现在其它修改分子结构的命令之后。下面是SYMMETRY的选项：

对称生成元的任何有效组合，说明见10.2节。

NOSYM          禁止使用对称性。

以下是ORIENT的选项：

CHARGE          定位分子，使得电荷中心是原点，轴是四极矩的本征矢。

MASS            定位分子，使得质心是原点，轴是惯性张量的本征矢（对Z矩阵输入是默认的）。也可以用CENTRE=MASS | CHARGE指定对称中心。

NOORIENT       禁止分子重新定位（对XYZ输入是默认的）。

SIGNX= $\pm 1$     强制第一个非零x坐标为正或负。类似地，可以用SIGNY和SIGNZ设置y、z坐标。这对固定遍及几种不同计算和结构的分子方位非常有用。也可

以设定系统变量ZSIGNX, ZSIGNY, ZSIGNZ为正值或负值来获得同样的效果。

**PLANEXZ** 对于 $C_{2v}$ 和 $D_{2h}$ 点群, 强制主平面是 $xz$ , 而不是默认的 $yz$ 。结构构建程序尝试交换坐标轴, 使尽可能多的原子位于主平面中, 因此对于平面分子的特殊情况, 这意味着所有原子都会位于主平面。默认设置使用J. Chem. Phys. 55, 1997 (1955)中的建议5a, 以及建议5b的第一部分。也可以指定PLANEYZ和PLANEXY, 但是注意, 后者目前对 $C_{2v}$ 点群会发生错误。

**ANGSTROM:** 用数或变量指定的键长, 如果没有单位, 那么单位是Å而不是默认的Bohr。

### 10.1.1 Z矩阵输入

原子说明行的一般格式为:

`[group[,]] atom, p1, r, p2, α, p3, β, J`

或者用

`[group[,]] atom, p1, x, y, z`

其中,

- |                      |  |
|----------------------|--|
| <i>group</i>         | 原子组序号 (可选)。用于对同一类的不同原子使用不同基组。接下来, 基组用组序号而不是原子符号引用。   |
| <i>atom</i>          | 位于位置 $p_0$ 的新原子化学符号。后面也可以选择加一个整数 (没有空格) 作为序号, 例如, C1, H2, 等。没有电荷和基函数的虚原子中心用Q或X表示, 后面还可以加一个数字, 例如Q1; 注意, <u>Z矩阵的第一个原子决不能用x</u> , 因为这可能与对称说明混淆 (可以用Q替代)。 |
| <i>p<sub>1</sub></i> | 当前原子所连接的原子。这既可以是数值 $n$ ( $n$ 表示Z矩阵的第 $n$ 行), 也可以是一个包含字母数字的字符串, 就像在前面卡的 <i>atom</i> 区定义的那样, 例如, C1, H2, 等。后一种形式只能用于原子被唯一标号的情况。                          |
| <i>r</i>             | 从 $p_1$ 到新原子的距离。这个值的单位是bohr, 除非在对称性说明部分之前或之后直接指定了ANG。  |
| <i>p<sub>2</sub></i> | 定义键角 $\alpha(p_0, p_1, p_2)$ 所需的第二个原子。使用与 $p_1$ 说明相同的规则。   |
| <i>α</i>             | 原子核间夹角 $\alpha(p_0, p_1, p_2)$ 。这个角度用单位度指定, 而且必须在 $0 < \alpha < 180^\circ$ 的范围内。   |

$p_3$	定义二面角 $\beta(p_0, p_1, p_2, p_3)$ 所需的第三个原子。只适用于 $J = 0$ ，见下。
$\beta$	以度为单位的二面角 $\beta(p_0, p_1, p_2, p_3)$ 。这个角定义为 $(p_0, p_1, p_2)$ 和 $(p_1, p_2, p_3)$ 平面之间的角度 ( $-180^\circ \leq \beta \leq 180^\circ$ )。只适用于 $J = 0$ ，见下。
$J$	如果指定并且非零，则用两个键角定义新的位置，而不是用一个键角和一个二面角。如果 $J = \pm 1$ ，则 $\beta$ 是角度 $\beta(p_0, p_1, p_3)$ 。如果 $J = 1$ ，则三个矢量的乘积 $(\mathbf{p}_1 - \mathbf{p}_0) \cdot [(\mathbf{p}_1 - \mathbf{p}_2) \times (\mathbf{p}_1 - \mathbf{p}_3)]$ 为正，若 $J = -1$ 则为负。
$x, y, z$	新原子的笛卡尔坐标。如果 $p_1 \leq 0$ ，则假定为这种格式；如果 $p_1 < 0$ ，则在几何优化时冻结坐标。

包括对称变换所涉及的原子在内，所有原子都应当在Z矩阵中指定。注意，对于第一个原子，不需要给出坐标，第二个原子只需要给出 $p_1, r$ ，而第三个原子的 $p_3, \beta, J$ 可以省略。通过平移和重新定位分子使得原点位于分子质心并且轴对应惯性张量的本征矢方向（参见上面的CHARGE选项），程序可以自动获得六个缺少的坐标。

一旦进行了重新定位，程序就会寻找对称性（ $D_{2h}$ 及其子群），除非定义了NOSYM选项。通过使用选项X,Y,Z,XY,XZ,YZ,XYZ合适的组合，还可以要求使用低对称性。它们指定对称性操作，这些符号定义了哪些坐标轴在操作下会改变符号。通过考虑给定对称元的所有组合来构造点群。如果用这种方法直接定义对称性，程序将会测试定义的点群是否可用，必要的时候还会交换坐标轴。这提供了确保使用同一点群的机制，例如，在产生的整个势能面上所有的点，可以安全地重复利用临近结构的分子轨道作为初始猜测，等等。

注意，对于密度拟合方法，不使用对称性。这种情况下自动执行NOSYM选项。

还要注意，默认分子自动定位仅发生在用内坐标（Z矩阵）定义结构时。对于XYZ输入，方位不变，除非在结构区指定了MASS选项。

### 10.1.2 XYZ输入

作为Z矩阵的替代，可以用以Ångstrom为单位的简单笛卡尔坐标。这一程序通过在几何结构说明给定之前，设置MOLPRO变量GEOMTYP的值为XYZ进行触发，但通常不需要<sup>1</sup>，因为结构说明的第一行若只有一个整数，将会看作是XYZ格式，也就是一行由化学符号加上三个笛卡尔坐标构成的情况。接下来的结构说明区域应当包含Minnesota Computer Centre, Inc. XYZ格式的笛卡尔坐标。可以使用变量名以及固定的数值。

<sup>1</sup>对于2009以下的版本，GEOMTYP=XYZ仍是必须的。——译注。

XYZ格式的文件包含两个标题行，第一个包含原子数，第二个是标题。其余的每一行都定义一个原子的坐标，最开始的位置是化学符号，接下来是 $x$ ,  $y$ ,  $z$ 坐标。化学符号后面也可以加序号；它被认为是原子组序号，用于对同一类型的不同原子使用不同的基组。此后，这个基组将对这个序号的组定义而不是对原子符号定义。作为进一步的扩展，前两个标题行可以省略<sup>2</sup>。

注意对于XYZ输入，默认不对分子重定位。但是可以强制重定位，通过在结构输入区设定MASS或CHARGE选项。

```
! $Revision: 2006.3 $
geomtyp=xyz
geometry={
3          ! 原子数
This is an example of geometry input for water with an XYZ file
O ,0.0000000000,0.0000000000,-0.1302052882
H ,1.4891244004,0.0000000000, 1.0332262019
H,-1.4891244004,0.0000000000, 1.0332262019
}
hf
```

[http://www.molpro.net/info/current/examples/h2o\\_xyzinput.com](http://www.molpro.net/info/current/examples/h2o_xyzinput.com)

XYZ格式在MSCI的XMOLE软件包发布的文档中有说明。注意，MOLPRO可以用PUT命令写XYZ文件（参见10.3节）。

## 10.2 对称性说明

如果使用了标准的Z矩阵输入，MOLPRO默认自动确定对称性。但是，有时需要使用较低的对称性或者不同于默认设置的方位，这可以通过直接指定使用的对称元实现，说明如下。

在第一个积分输入卡中（接在INT卡之后，或者是分子结构区的第一个卡），可以给出对称生成元，它们唯一地定义点群。点群的维度是2\*\*（给定的域的个数）。每个域由X, Y或Z中的一个或几个构成（中间没有空格），用于指定哪些坐标轴在相应的生成对称操作下会改变符号。在可能的情况下，选择z轴作为唯一轴通常是最佳选择（对 $C_2$ 和 $C_{2h}$ 是必需的）。在这种情况下，可能的域有：

（不指定卡）	$C_1$ （也即，不使用点群对称性）
Z	$C_s$
XY	$C_2$
XYZ	$C_i$

<sup>2</sup>对于2009以下的版本，两个标题行不能省略。——译注。



X,Y	$C_{2v}$
XY,Z	$C_{2h}$
XZ,YZ	$D_2$
X,Y,Z	$D_{2h}$

注意，只能使用阿贝尔点群对称性，因此对于有简并对称性的分子，必须使用阿贝尔子群——例如，线型分子用 $C_{2v}$ 或 $D_{2h}$ 。

关于对称群和不可约表示顺序的更多细节，参见4.8节。关于自动产生对称平面的更多信息，参见10.1.1节。

注意，只有当用内坐标（Z矩阵）定义结构时，默认才会自动发生分子重定位。对于XYZ输入的情况，方位不变，除非在结构区指定MASS选项。

### 10.3 写Gaussian, Xmol或Molden的输入（PUT）

PUT命令可用在输入文件的任何位置，用于把当前结构打印或写入文件中。语法为：

PUT,*style,file,status,info*

- 如果*style*为GAUSSIAN，将写一个完整的GAUSSIAN输入文件；在这种情况下，*info*将用于（执行路径）第一个数据行，默认为“# SP”。
- 如果*style*为XYZ，将写一个XYZ文件（参见10.1.2节）。
- 如果*style*是CRD，坐标将用CHARMm CRD格式书写。
- 如果*style*为MOLDEN，将创建用于MOLDEN可视化程序的接口文件；更详细的信息和例子在下面给出。
- 如果忽略*style*，将写入当前结构的Z矩阵，可能的话还写入梯度。

*file*定义写入数据的文件名；如果为空，数据将写入到输出流中。如果忽略*status*或设置为NEW，那么文件中所有的旧内容都被清除；否则对文件追加。

#### 10.3.1 用Molden显示结果

如果计算包含了结构，分子轨道，和简正模式的信息，可以用PUT,MOLDEN转存为MOLDEN格式。

到GOPENMOL程序的接口提供了另一种可视化方法，说明参见33.8节。

在以下的输入中，产生绘制水分子轨道图和显示简正振动模式的所有必要信息：

```
! $Revision: 2006.0 $
***,H2O
geometry={angstrom;o;h,o,roh;h,o,roh,h,theta};
roh=1.0
theta=104.0
rhf;
optg;
{frequencies;
print,low,img;}
put,molden,h2o.molden;
```

[http://www.molpro.net/info/current/examples/h2o-put\\_molden.com](http://www.molpro.net/info/current/examples/h2o-put_molden.com)

下面的例子通过为MOLDEN引入自然轨道，产生差密度。注意，虽然MOLDEN有绘制差密度图的内部功能，但这里仍使用了更一般的方法，顺便还可以避免STO-3G，3-21G，4-31G，和6-31G基组的限制。

```
! $Revision: 2009.1 $
gprint,orbitals
symmetry,y,planexz
geometry={0;H1,0,r;h2,0,r,h1,alpha}
r=1.8
alpha=104
int;
{hf;wf,10,1;orbital,2100.2}
{multi;wf,10,1;orbital,2140.2}

{matrop
load,dscf,density,2100.2      ! 加载scf密度
load,dmcsf,density,2140.2    ! 加载mcscf密度
add,ddiff,dmcsf,-1,dscf     ! 计算dmcsf-dscf
natorb,neworb1,dscf
natorb,neworb2,dmcsf
natorb,neworbs,ddiff
save,neworbs,2110.2
save,ddiff,2110.2}

put,molden,h2o_ddens.molden;orb,2110.2
```

[http://www.molpro.net/info/current/examples/h2o-diffden\\_molden.com](http://www.molpro.net/info/current/examples/h2o-diffden_molden.com)

## 10.4 结构文件

使用格式

GEOMETRY=*file*

从*file*而不是从输入行读取结构说明。这一文件必须包含对称区的全部信息，也即，对称性说明（可选），Z矩阵或者xyz输入。

## 10.5 点电荷的格子

LATTICE,[INFILE=*input\_file*,] [OUTFILE=*output\_file*,] [VARGRAD,] [NUCONLY,] [REMOVE]

包含点电荷格子，例如用于QM/MM计算（见第四十九章）。应当给出外部文件（*input\_file*）作为输入，格式为：

*Comment line*

*number of point charges N*

*x1, y1, z1, q1, flag1*

⋮

*xN, yN, zN, qN, flagN*

*x*, *y*和*z*的内容表示点电荷坐标（单位Å），*q*为电荷，*flag=1*表示对该格子点计算梯度（0表示无梯度）。

*outfile*指定用于写入格点梯度的文件名。如果为空，将写到输出流中。

VARGRAD                   （逻辑型）把格点梯度存储到变量VARGRAD中。

NUCONLY                   （逻辑型）禁止求解关于格点的梯度，这与格点文件中的*flag*无关。

REMOVE                   （逻辑型）删除格点。

格点梯度不支持对称性。

## 10.6 重新定义和打印原子质量

可以打印当前所有原子的质量，使用

MASS, PRINT

可以重新定义原子的质量，使用

MASS, [*type*,] [*symbol=mass*, ...]

可选关键词*type*的值可以是AVER[AGE]，使用平均同位素质量，也可以是ISO[TOPE]，使用最丰同位素质量。这只对转动常数和振动频率产生影响。和大多数量子化学软件相同，默认的*type*为AVERAGE。如果给出了INIT，则删除所有以前的质量定义，重新设置默认值。

每个质量可以用下面的输入改变，其中 $symbol$ 是原子的化学符号， $mass$ 是相应的质量。一个MASS卡可以给出多项输入，也可以连用几个MASS卡。程序使用最后一次给出的质量。

注意，如果把分子质心作为原点的话，若是对于对称性所涉及原子指定不同的同位素质量，会降低体系的对称性。这一影响可以通过把电荷中心作为原点来避免，也就是在GEOMETRY输入的第一项指定CHARGE:

GEOMETRY={CHARGE; ...}

## 10.7 虚原子中心

DUMMY, $atom1,atom2,\dots$

设定原子1, 2等的核电荷为零，例如，用于进行基组平衡计算。 $atom1, atom2,\dots$ 可以是Z矩阵行号或标记名称。注意，通过用MOLPRO的变量DUMMYATOMS，程序可以记录虚原子的当前设置用于续算。如果用不加选项的DUMMY卡，可以把虚原子的电荷重新设置为初始值。如果（1）遇到INT命令，或者（2）遇到新的结构输入，那么虚原子中心也重新设置为初始电荷。

如果通过定义虚原子降低体系对称性，程序将不能自动识别对称性。因此对于给定的虚原子，要么所有的对称性等价原子也必须是虚原子，要么根据需要人工降低对称性。如果对称性与虚原子中心的定义不一致，将会导致错误。

### 10.7.1 平衡计算

用 $dummy$ 卡，很容易进行平衡校正。首先计算整个体系的能量，接下来用 $dummy$ 卡计算子体系的能量。

### 10.7.2 示例：OH-Ar的相互作用能

```
! $Revision: 2006.0 $
***,OH(2Sig+)-Ar linear
memory,2,m
geometry={q1;           ! 虚原子中心位于质心
o,q1,ro;h,q1,rh,o,180;  ! OH的结构
ar,q1,rar,o,theta,h,0}  ! Ar的结构
roh=1.8                 ! OH键长
rar=7.5                 ! Ar到质心的距离
theta=0                 ! OH-Ar角度
ro=roh*16/17            ! O到质心的距离
rh=roh*1/17             ! H到质心的距离
basis=avdz              ! 基组
```

```

text,calculation for complex
{rhf;occ,8,3,3;wf,27,1,1}      ! RHF计算整个体系
rccsd(t)                        ! CCSD(T)计算整个体系
e_ohar=energy                    ! 把能量保存到变量e_ohar中

text,cp calculation for OH
dummy,ar                        ! 把Ar作为虚原子中心
{rhf;occ,3,1,1;wf,9,1,1}      ! RHF计算OH
rccsd(t)                        ! CCSD(T)计算OH
e_oh=energy                      ! 把能量保存到变量e_oh中

text,cp calculation for Ar
dummy,o,h                       ! 把OH作为虚原子
hf                              ! scf计算Ar
ccsd(t)                         ! CCSD(T)计算Ar
e_ar=energy                      ! 把能量保存到变量e_ar中

text,separate calculation for OH
geometry={O;H,O,roh}           ! 单个OH的结构
{rhf;occ,3,1,1;wf,9,1,1}      ! RHF计算OH
rccsd(t)                        ! CCSD(T)计算OH
e_oh_inf=energy                 ! 把能量保存到变量e_oh_inf中

text,separate calculation for Ar
geometry={AR}                   ! 单个Ar的结构
hf                              ! scf计算Ar
ccsd(t)                         ! CCSD(T)计算Ar
e_ar_inf=energy                 ! 把能量保存到变量e_ar_inf中

de=(e_ohar-e_oh_inf-e_ar_inf)*tocrm ! 计算未校正的相互作用能
de_cp=(e_ohar-e_oh-e_ar)*tocrm      ! 计算平衡校正的相互作用能
bsse_oh=(e_oh-e_oh_inf)*tocrm      ! OH的BSSE
bsse_ar=(e_ar-e_ar_inf)*tocrm      ! Ar的BSSE
bsse_tot=bsse_oh+bsse_ar           ! 总的BSSE

```

[http://www.molpro.net/info/current/examples/ohar\\_bsse.com](http://www.molpro.net/info/current/examples/ohar_bsse.com)

关于进行平衡校正的结构优化计算，参见41.4.7节。



## 第十一章

## 基组输入

### 11.1 概述：基组和基组库

MOLPRO使用的基函数不仅用于描述轨道，也提供用于密度拟合的辅助基（参见第十五章），以及在显关联方法中，通过近似恒等解提供简化积分（参见第三十章）。为了方便，程序内部保存大量的基组。其中第一种基组的名称总是ORBITAL，是描述轨道的主要基组。其它的基组根据需要定义（说明见后），或者在需要时由程序自动产生。在后一种情况中，密度拟合以及其它模块尝试猜测适于轨道基的合理库存拟合基组。当使用其它非标准轨道基组时，建议检查拟合基组的选择。

基组既可以来自程序的PHP脚本基组库，也可以另外定义，或者二者结合。还可以用CARTESIAN或SPHERICAL命令选择基函数的类型。

### 11.2 笛卡尔和球谐基函数

MOLPRO默认使用球谐基函数(5d, 7f, 等)，即使是对Pople基组，如6-31G\*\*, 也是如此。这一性能可能不同于其它的程序，不过可以用CARTESIAN命令使用笛卡尔函数：

CARTESIAN

如果遇到这个命令，MOLPRO的逻辑变量CARTESIAN设置为真(1.0)，随后所有的计算都使用笛卡尔基函数。这在续算时可以继续保持。可以通过以下命令换成球谐基函数：

SPHERICAL

### 11.3 基组库

基组库由一组无格式文本文件以及相应的索引构成，组成了通用的基组（原高斯函数和相应的收缩）和有效芯势数据库。这些文件可以在源文件的目录树中找到，分别为lib/\*.libmol和lib/libmol.index，但通常更方便的方法是使用提供的工具之一查询数据库。

这些基组很多都直接来自美国太平洋西北国家实验室（Pacific Northwest National Laboratory）的基组数据库，但是还有其它的，特别是Stuttgart 有效芯势和基组。

libmol程序提供到数据库的简单命令行接口。它需要定义环境变量LIBMOL，用于指向lib/目录，默认指向编译时源文件树的位置，因此通常不需要定义。命令行的语法是：

```
libmol [-p print] [-e element] [-k key] [-t type] [-f format]
```

其中的参数是：

<i>print</i>	输出级别：0表示列出匹配关键词，1表示还打印这些项。
<i>element</i>	定义化学元素。如果忽略，寻找所有元素。
<i>key</i>	定义记录的关键词。如果忽略，寻找所有关键词。
<i>type</i>	定义项的类型，即 <i>s, p, ...</i> 。如果忽略，寻找所有类型。
<i>format</i>	给定输出格式，需要是text（默认），molpro（MOLPRO 输入格式），table（列表）或html（html表格）之一。

浏览基组库更方便的方法是通过基于网络的界面。PHP脚本basis.php 具有基于图形和视窗的界面，用于进行寻找，如下图。它可以本地安装，但也可以从主页上的基组库 获得：<http://www.molpro.net/info/current/basis.php>

H																	He	
Li	Be											B	C	N	O	F	Ne	
Na	Mg											Al	Si	P	S	Cl	Ar	
K	Ca	Sc	Ti	V	Cr	Mn	Fe	Co	Ni	Cu	Zn	Ga	Ge	As	Se	Br	Kr	
Rb	Sr	Y	Zr	Nb	Mo	Tc	Ru	Rh	Pd	Ag	Cd	In	Sn	Sb	Te	I	Xe	
Cs	Ba	La	Hf	Ta	W	Re	Os	Ir	Pt	Au	Hg	Tl	Pb	Bi	Po	At	Rn	
Fr	Ra	Ac	Rf	Ha														
		La																
			La	Ce	Pr	Nd	Pm	Sm	Eu	Gd	Tb	Dy	Ho	Er	Tm	Yb	Lu	
		Ac	Th	Pa	U	Np	Pu	Am	Cm	Bk	Cf	Es	Fm	Md	No	Lr		



## 11.4 默认基组

如果对所有对称唯一的原子组都没有定义基组，那么程序假定使用全局默认值。目前的默认值是VDZ，但也可以用

`BASIS,basis`

或

`BASIS=basis`

替换。*basis*在文件lib/defbas中查找，从基组库调用相应的完全收缩基组，有些情况还有ECP。这个映射关系包含以下通用基组。

- 所有的Dunning关联一致基，使用基组的标准名称（例如aug-cc-pVDZ）或缩写（例如AVDZ）调用。
- 对于第一行<sup>1</sup>原子，有旧一些的Dunning/Hay分块收缩双zeta基（DZ和DZP）。
- Roos的ANO基组（R00S）。
- Stuttgart ECP和相应基组（例如ECP10MWB）。
- Hay ECP和相应基组（ECP1和ECP2）。
- 一些Karlsruhe基组（SV, TZV, 以及用于某些元素的SVP, TZVP, TZVPP, TZVPPP）。
- 用于Ga-Kr的Binning/Curtiss基（BINNING-SV, BINNING-SVP, BINNING-VTZ和BINNING-VTZP）。
- 大多数Pople基组使用其标准名称（例如6-31G\*, 6-311++G(D,P), 等）。在这种情况下需要特别注意，不能使用下面描述的用于限定基组的括弧修饰机制，必须使用被指定的整个标准基组。

例子：

`BASIS=VTZ`

对所有的原子产生价电子三zeta基组。因此，输入文件

```
! $Revision: 2006.3 $
***,h2o cc-pVTZ basis      ! 标题
r=1.85,theta=104           ! 设定结构参数
```

<sup>1</sup> 氢和氦比较特殊，它们不在理论化学家的元素周期表中。因此，第一行元素指的是3-10号元素（第二周期）。相应地，第二行元素对应第三周期，第三行元素对应第四周期，等。——译注。

```

geometry={0;          ! z-矩阵结构输入
      H1,0,r;
      H2,0,r,H1,theta}
basis=VTZ              ! 使用VTZ基
hf                     ! 闭壳层scf

```

[http://www.molpro.net/info/current/examples/h2o\\_scf.vtz.com](http://www.molpro.net/info/current/examples/h2o_scf.vtz.com)

完全等同于

```

! $Revision: 2006.3 $
***,h2o cc-pVTZ basis      ! 标题
r=1.85,theta=104          ! 设定结构参数
geometry={0;              ! z-矩阵结构输入
      H1,0,r;
      H2,0,r,H1,theta}
basis={
  spdf,o,vtz;c;
  spd,h,vtz;c}
hf;

```

[http://www.molpro.net/info/current/examples/h2o\\_scf.vtz\\_explicit.com](http://www.molpro.net/info/current/examples/h2o_scf.vtz_explicit.com)

默认基组可以在输入文件能量计算前的任何位置定义，并且只能使用一个BASIS卡。默认基组应用于所有类型的原子，但是对特定的原子可以用不同的基组代替，这在后面解释。可以使用一些关于最大角动量函数的限制，或是收缩函数的数量限制，如下。

减少基组的最大角动量可以用如下语法：

**BASIS,VQZ(D)**

这会忽略正常情况下出现在VQZ基组中的*f*和*g*函数。

**BASIS,VQZ(D/P)**

将会另外指定氢的最大角动量是1，也就是忽略氢的*d*轨道。

对于一般收缩基组，可以用扩充的语法另外给出每个角动量收缩函数的数量。例如，

**BASIS,R00S(3s2p1d/2s)**

将会从Roos的ANO数据集产生6-31G\*尺寸的基组。

## 11.5 不同原子的默认基组

在BASIS输入区中可以对不同原子给定特定的基组，一般形式如下：

```

BASIS

```

```

SET=type           ! type可以是ORBITAL, DENSITY或用于密度拟合基组的
                   ! 其它名称; 可选; 默认=ORBITAL
DEFAULT=name       ! 设定默认的基组为name
atom1=name1       ! 对atom1使用基组name1
atom2=name2       ! 对atom2使用基组name2
原始基组指定      ! 附加的基函数
SET=type          ! 在下面的行指定其它类型的基
...
END

```

可以在基组区给出任何数量的基组。

默认的原子说明可以放入同一行，之间用逗号隔开：

```
DEFAULT=name,atom1=name1,atom2=name2
```

这里基组 $name1$ ， $name2$ 分别覆盖了特定原子 $atom1$ ， $atom2$ 的默认基组 $name$ 。例如，

```
DEFAULT=VTZ,O=AVTZ,H=VDZ
```

使用VTZ作为默认的基组，但是对氧和氢分别是AVTZ和VDZ。

原子特定基组的命名习惯与默认基组部分的说明完全相同。关键词DEFAULT可以缩写为DEF。在基组区定义的任何DEFAULT基组将覆盖以前在基组区之外定义的基组。

SET,DEFAULT,atom=name的定义是可选的。如果没有给出DEFAULT，将使用前面最后一个BASIS卡定义的默认基组。

如果之后没有加入更多的原始基组定义，也可以使用单行的形式：

```
BASIS,DEFAULT=VTZ,O=AVTZ,H=VDZ
```

或

```
BASIS=VTZ,O=AVTZ,H=VDZ
```

二者都相当于

```

BASIS
DEFAULT=VTZ
O=AVTZ
H=VDZ
END

```

注意，任何新的BASIS卡都将覆盖前面所有的基组输入，但默认基组除外（除非也给出了这一部分）。

可选的附加原始基组定义（见下一节）将添加到给出的特定原子基组中，即，原子的给定基组和原始基组联合用于该原子。

例子:

```
BASIS
DEFAULT=VTZ      ! 默认使用cc-pVTZ基组
H=VDZ            ! 对氢原子使用cc-pVDZ基组
END
```

这也可以写作:

```
BASIS={DEF=VTZ,H=VDZ}
```

```
BASIS
DEFAULT=VTZ      ! 默认使用cc-pVTZ基组
H=VDZ            ! 对氢原子使用cc-pVDZ基组
D,H,VTZ          ! 对于氢原子, 把VTZ的d函数加到VDZ基组上
END
```

```
BASIS
SPD,O,VTZ        ! 对氧原子使用VTZ基组的非收缩s,p,d函数
S,H,H07           ! 对氢原子使用Huzinaga 7s
C,1.4            ! 收缩前四个s-函数
P,H,1.0,0.3      ! 对氢原子添加两个p-函数
END
```

可以先后连续出现几个BASIS卡/区。对给定的原子总是使用最后指定的说明和类型。用BASIS命令给定的默认值, 可以用积分输入部分的说明覆盖。如果对一个原子给定了特定的基函数类型, 那么还需要同时定义所有的其它类型。例如在上面的例子中, O不包含*f*-函数, 即使在全局默认值中包含了*f*-函数也是如此。同样, 对氢定义*s*函数就会关闭氢的默认基组, 因此必须定义*p*函数。还可以使用原子组序号代替原子符号。

相同的输入形式也可以作为积分程序的直接输入。与MOLPRO92相反, 现在基函数定义的第二个位置除了可以用原子组序号外, 还可以用原子符号:

```
{\tt SPD,O,VTZ}    ! 对所有的氧原子使用{\tt VTZ}基组
{\tt SPD,1,VTZ}    ! 对原子组1使用{\tt VTZ}基组
```

除了BASIS ... END块, 还可以用BASIS [=] { ... }结构。

如果对一个唯一的原子组没有定义基函数, 程序假定是默认值。更详细的内容, 包括使用默认值的定义, 见下面BASIS子命令的说明。

## 11.6 原始集的定义

定义一组基函数需要指定包含一组原高斯函数的数据卡，后面可以选择接一个或多个卡，用于定义包含在最终基组内的原函数的特殊收缩（关于指定收缩，参见11.7节）。当读取了全部收缩定义后（由下一个数据卡是不是定义收缩来确定），没有包含在任何收缩集中的其余原函数，将会被添加到非收缩基组中。

有四种不同的输入形式，解释见下面的从a)到d)。在指定了选项的情况下（例如SCALE, NPRIM），它们可以按照任何顺序给出，但是在每个选项后，不能用不带选项关键词的值。

在所有的情况下，*type*定义角动量对称性（S, P, D, F, G, H, 或I）。*type*可以包含几种类型，例如SPD或DF。这通常只能用于来自默认基组库的收缩或非收缩。

所有原子的基组用结构输入中的标签名称*atom*加载。如果*atom*是整数，则代表z-矩阵的行。

### a) 库存基组

*type, atom, key, scale2, nprim;*

或

*type, atom, key, [SCALE=scale|SCALE2=scale2], [NPRIM=nprim|DELETE=ndel];*

从基组库加载名为*key*的基。

如果出现了*scale*或*scale2*，那么所有的指数都分别用*scale*或*scale\*\*2*换算。如果指定了*nprim*，那么前*nprim*个指数仅取自基组库。如果*nprim*为负值，或者给出了*ndel*，那么将删除来自基组库的最后的|*nprim*|（或*ndel*）个基函数。伴随基组库基组的，是一组默认的收缩因子，可以用随后的收缩卡访问。*type*可以包含几种类型，例如SPD或DF。这通常只能用于默认的收缩，即，这样的卡后面只能接“C”，不能有其它关于收缩的说明。

### b) 基组直接输入

*type, atom, exp1, exp2, ... expn; expn+1, ...;*

指数的一般说明：可以使用随后的卡继续（分号隔开），如上所示（第一个卡可以包含到19个指数，后面的每个卡是20个指数）。

指数（以及后面描述的其它数值参数，例如函数和收缩因子的数量）可以按照一般的输入表达式给出，允许包含变量。但是需要注意，典型情况下只求解一次表达式，与分析整个基组是同时的。这一般发生在第一次读取基组的时候，可能在第一次SCF计算之前。如果改变了基组所依赖的变量，程序将不会注意，因此新的基组将不能用于后面的计算。然而，如果在输入中出现了新的基组区，那么程序将把用旧基组计算的积分标记为用过的量；后面的任务步骤将使用新的基组。

## c) 平缓 (even tempered) 基组

*type, atom, EVEN, nprim, ratio, centre, dratio*

或者

*type, atom, EVEN, NPRIM=nprim, [RATIO=ratio], [CENTRE=centre], [DRATIO=dratio]*

产生一般平缓函数集。函数的数量 $n$ 用 $nprim$ 指定, 几何中心 $c$ 用 $centre$ , 连续指数的平均比率 $r$ 用 $ratio$ , 这个比率的变化 $d$ 用 $dratio$ 。如果没有给出 $centre$ , 那么同一类型的前一个基就会用弥散函数扩充。在这种情况下如果没有给出 $ratio$ , 那么从前一个基的最后两个函数的指数决定 $r$ 。如果这不可用, 采用默认的 $r = 2.5$ 。 $d = 1$  (默认) 定义一个真正的平缓集, 否则连续指数的变化率呈线性; 指数用公式另外给出:

$$\log e_i = \log c + ((n+1)/2 - i) \log r + \frac{1}{2}((n+1)/2 - i)^2 \log d \quad i = 1, 2, \dots, n$$

例1

SP, 1, VTZ; C; SP, 1, EVEN, 1;

对原子1产生一般收缩 $s$ 和 $p$ 的三zeta基组, 并用一个弥散函数扩充。

例2

SPD, 1, VTZ, DELETE=1; C;

SP, 1, EVEN, NPRIM=2, RATIO=2.5;

对原子1产生一般收缩 $s$ 和 $p$ 的三zeta基组。包含了Dunning对能量优化的两个 $d$ -函数。删除了最后的 $s$ 和 $p$ 函数, 并用比率2.5的两个平缓函数替换。

## d) 三项调节的基组:

*type, atom, EVEN3, nprim,  $\alpha$ ,  $\beta$ ,  $\gamma$*

产生 $nprim$ 个函数的三参数集, 指数的公式为

$$e_i = \alpha; \quad e_i = e_{i-1} \beta \left( 1 + \frac{\gamma i^2}{(nprim + 1)^2} \right)$$

## e) 正则平缓 (regular even tempered) 基组

*type, atom, EVENR, nprim, aa, ap, bb, bp*

按照M W Schmidt and K Ruedenberg, J. Chem. Phys. 71 (1970) 3951中描述的“正则”规定, 产生 $nprim$ 个函数的平缓集。如果参数 $aa$ ,  $ap$ ,  $bb$ ,  $bp$ 中的任何一个被忽略或者为零, 则数值从上面文献的表三获得。

## f) 限定级数的平缓基组

*type, atom, EVENP, nprim,  $\alpha$ ,  $\beta$ ,  $\gamma$*

用 $nprim$ 个函数和用 $\alpha$ 给定的最大指数产生平缓基组。第一个和第二个指数的级数（比率）由 $\beta$ 调整，倒数第二个和最后一个指数的级数用参数 $\gamma$ 调整。级数之间用线性内插。级数因子的具体值由下式给定：

$$p(\beta) = \frac{\text{exponent}^i}{\text{exponent}^{i+1}} = \frac{5}{\pi}(\arctan(\beta - 2.5) + \frac{\pi}{2}) + \sqrt{2}$$

因此，对于 $\beta \ll 0 : p \rightarrow \sqrt{2}$ 和 $\beta \gg 0 : p \rightarrow 5 + \sqrt{2}$ ，上式限定了两个值之间的级数因子，并可以对非限定基组进行优化。对于 $\beta \approx 0$ ，级数的因子近似为2。

$type, atom, EVENP2, nprim, \alpha, \beta, \gamma, \delta$

通过第三个参数（这里是 $\gamma$ ）定义上面的级数，对限定级数的平缓基组进行推广。比率因子通过在 $p(\beta) \rightarrow p(\gamma)$ 和 $p(\gamma) \rightarrow p(\delta)$ 之间插值确定。

## 11.7 收缩集的定义

a)  $C, first.last, c1, c2, \dots, cn; cn+1, \dots;$

收缩函数的一般说明。 $first.last$ 定义进行收缩的原函数的范围。顺序对应于前一个输入卡指定的原函数。 $c1, c2, \dots$ 是是 $last - first + 1$ 个收缩因子。如所示，可以在后面的卡中继续。

b)  $C;$

使用来自基组库的默认收缩。这用于收缩原函数的数量和不同收缩集的数量。

c)  $nC, first.last;$

来自基组库的 $n$ 个收缩函数。 $first.last$ 定义进行收缩的原函数的范围。如果忽略了 $n$ ，并指定了 $first.last$ ，那么 $n = 1$ 。如果忽略了 $first.last$ ，则使用基组库的默认值。如果 $n$ 和 $first.last$ 都被忽略，则使用二者的默认值。

d)  $nC, first.last, record.file, orb.sym;$

$n$ 个收缩函数来自molpro的 $record.file$ 文件中对称性为 $sym$ 的轨道 $orb, orb + 1, \dots, orb + n - 1$ 。给定轨道中的第一个非零因子对应第一个有关基函数。 $first.last$ 定义进行收缩的原函数的范围。如果忽略了 $first.last$ ，则使用来自指定轨道的所有因子。

例子

$2C, 1.12, 2100.2, 1.1$

使用来自轨道1.1和2.1的前12个因子，产生两个收缩。轨道从记录2100.2读取。

## 11.8 例子

这是对水使用默认基组的例子：

```
! $Revision: 2006.3 $
***, H2O
```

```
basis=VQZ(f/p)
R=0.95 ANG,THETA=104 DEGREE
geometry={O;H1,O,R;H2,O,R,H1,THETA}
hf                ! 进行闭壳层SCF计算
```

[http://www.molpro.net/info/current/examples/h2o\\_vqz\\_fp.com](http://www.molpro.net/info/current/examples/h2o_vqz_fp.com)

这等价于直接输入形式:

```
! $Revision: 2006.3 $
***,H2O
R=0.95 ANG,THETA=104 DEGREE
geometry={O;H1,O,R;H2,O,R,H1,THETA}
basis={spdf,o,vqz;c;sp,h,vqz,c;}
hf                ! 进行闭壳层SCF计算
```

[http://www.molpro.net/info/current/examples/h2o\\_vqz\\_fp\\_explicit.com](http://www.molpro.net/info/current/examples/h2o_vqz_fp_explicit.com)



## 第十二章

## 有效芯势

赝势（有效芯势，ECP）可以在BASIS区的开始定义。输入卡的一般形式是：  
`ECP,atom,[ECP specification]`  
它用化学符号或原子组序号对一个原子定义赝势。ECP说明部分可以包含一个关键字，用于引用基组库中存储的赝势，也可以（连续在几个输入卡中）直接定义，见下。

### 12.1 来自ECP库的输入

基组库目前包含了赝势和相应的价电子基组，来自

a) Los Alamos小组（P. J. Hay and W. R. Wadt, J. Chem. Phys. **82**, 270 (1985), 以及此后的两篇文献），

b) Stuttgart/Köln小组（例如A. Nicklass, M. Dolg, H. Stoll and H. Preuß, J. Chem. Phys. **102**, 8942 (1995)；更多的细节和文献参见  
<http://www.theochem.uni-stuttgart.de/pseudopotentials/>。

赝势a)校正到一个合适的原子参考态的轨道能量和密度，而赝势b)用多个原子态的总价电子能量产生。

第一种情况a)的基组库关键词是ECP1和ECP2；当对给定的原子有一个以上的赝势时，ECP2用于表示小芯ECP。（例如对Cu，ECP1表示类Ar 18电子的芯，而ECP2模拟类Ne 10电子的芯，把3s和3p电子放入到价层）。对于包含电子关联的精确计算，推荐把主量子数等于价轨道的所有芯轨道放入到价轨道中。

在第二种情况b)中，基组库关键词具有ECP $n$ XY的形式： $n$ 是用赝势代替的芯电子数量， $X$ 表示用于产生赝势的参考体系（ $X = S$ ，单价电子离子； $X = M$ ，中性原子）， $Y$ 表示参考数据的理论级别（ $Y=HF$ ：Hartree-Fock； $Y=WB$ ：准相对论； $Y=DF$ ：相

对论)。对于单个或两个价电子的原子而言,  $X = S$ ,  $Y=DF$  是很好的选择, 否则推荐  $X = M$ ,  $Y=WB$  (或  $Y=DF$ )。 (对于较轻的原子, 或者为了讨论相对论的影响, 相应的  $Y=HF$  赝势也可能会有用。) 此外还可以使用自旋轨道 (SO) 势和芯极化势 (CPP), 用于 b) 类 ECP, 但目前不包含在基组库中, 因此需要在这里另外输入 (参看下面)。

在 a) 和 b) 两种情况中, 赝势和相应的基组使用相同的关键词, 但对 a) 需要使用前缀 MBS-...。

## 12.2 直接输入 ECP

对每种赝势, 必须提供以下信息:

- 一个卡, 形式为

ECP, atom,  $n_{core}$ ,  $l_{max}$ ,  $l'_{max}$ ;

其中  $n_{core}$  是用赝势  $V_{ps}$  代替的芯电子数量,  $l_{max}$  是  $V_{ps}$  标量相对论部分的半局域项数量,  $l'_{max}$  是相应的 SO 部分的项个数:

$$V_{ps} = -\frac{Z - n_{core}}{r} + V_{l_{max}} + \sum_{l=0}^{l_{max}-1} (V_l - V_{l_{max}}) \mathcal{P}_l + \sum_{l=1}^{l'_{max}} \Delta V_l \mathcal{P}_l \vec{l} \cdot \vec{s} \mathcal{P}_l;$$

半局域项 (角动量投影为  $\mathcal{P}_l$ ) 由  $l = l_{max}$  的局域项补充。

- 定义  $V_{l_{max}}$  的一些卡, 其中第一个给出

$$V_{l_{max}} = \sum_{j=1}^{n_{l_{max}}} c_j r^{m_j-2} e^{-\gamma_j r^2}$$

中的展开式长度  $n_{l_{max}}$ , 其余的  $n_{l_{max}}$  个卡给出参数, 形式为

$$m_1, \gamma_1, c_1; m_2, \gamma_2, c_2; \dots$$

- 定义标量相对论半局域项的一些卡, 顺序是  $l = 0, 1, \dots, l_{max} - 1$ 。对每一项, 必须给出一个在

$$V_l - V_{l_{max}} = \sum_{j=1}^{n_l} c_j^l r^{m_j^l-2} e^{-\gamma_j^l r^2}$$

中展开长度是  $n_l$  的卡, 后面立即接  $n_l$  个卡, 这些卡相应的参数形式为  $m_1^l, \gamma_1^l, c_1^l; m_2^l, \gamma_2^l, c_2^l; \dots$

- 类似地, 是一些径向势  $\Delta V_l$  因子的卡, 定义  $V_{ps}$  的 SO 部分。

## 12.3 直接输入ECP的例子

```
! $Revision: 2006.3 $
***,CU
! Cu原子SCF d10s1 -> d9s2激发能，使用相对论Ne-芯的赝势
! 和Stuttgart/Koeln小组的基组
gprint,basis,orbitals
geometry={cu}
basis
ECP,1,10,3; ! ECP输入
1; ! 无局域势
2,1.,0.;
2; ! S势
2,30.22,355.770158;2,13.19,70.865357;
2; ! P势
2,33.13,233.891976;2,13.22,53.947299;
2; ! D势
2,38.42,-31.272165;2,13.26,-2.741104;
! (8s7p6d)/[6s5p3d] 基组
s,1,27.69632,13.50535,8.815355,2.380805,.952616,.112662,.040486,.01;
c,1.3,.231132,-.656811,-.545875;
p,1,93.504327,16.285464,5.994236,2.536875,.897934,.131729,.030878;
c,1.2,.022829,-1.009513;C,3.4,.24645,.792024;
d,1,41.225006,12.34325,4.20192,1.379825,.383453,.1;
c,1.4,.044694,.212106,.453423,.533465;
end
rhf;
e1=energy
{rhf;occ,4,1,1,1,1,1,closed,4,1,1,1,1,1;wf,19,7,1;}
e2=energy
de=(e2-e1)*toev ! Delta E = -0.075 eV
```

[http://www.molpro.net/info/current/examples/cu\\_ecp\\_explicit.com](http://www.molpro.net/info/current/examples/cu_ecp_explicit.com)

## 12.4 从基组库输入ECP的例子

```
! $Revision: 2006.3 $
***,AuH
! AuH分子在r(exp)的CCSD(T)结合能；使用Stuttgart/Koeln
! 小组标量相对论19价电子的赝势
gprint,basis,orbitals;
geometry={au}
basis={
ecp,au,ECP60MWB; ! ECP输入
spd,au,ECP60MWB;c,1.2; ! 基组
```

```
f,au,1.41,0.47,0.15;  
g,au,1.2,0.4;  
spd,h,avtz;c;  
}  
rhf;  
{rccsd(t);core,1,1,1,,1;}  
e1=energy  
geometry={h}  
rhf  
e2=energy;  
rAuH=1.524 ang          ! 分子计算  
geometry={au,h,au,rAuH}  
hf;  
{ccsd(t);core,2,1,1;}  
e3=energy  
de=(e3-e2-e1)*toev      ! 结合能 = 3.11 eV
```

<http://www.molpro.net/info/current/examples/auh-ecp-lib.com>

### 13.1 输入选项

芯极化矩阵元的计算需要使用CPP卡，它可以在MOLPRO输入的任何位置调用，只要此前已经计算了积分。CPP卡具有以下格式：

- CPP,INIT,*ncentres*;
- CPP,ADD[,*factor*];
- CPP,SET[,*fcpp*];

CPP,INIT,<*ncentres*>;

需要继续读取abs(<*ncentres*>)个卡，按照下面的格式：

<*atomtype*>,<*ntype*>,< $\alpha_d$ >,< $\alpha_q$ >,< $\beta_d$ >,<*cutoff*>;

其中，

<*atomtype*>对应程序积分部分原子中心的标记；

<*ntype*>固定截断函数的形式（选择1是Stoll/Fuentealba，2是Mueller/Meyer）；

< $\alpha_d$ >是静态偶极极化率；

< $\alpha_q$ >是静态四极极化率；

< $\beta_d$ >是偶极极化率的一阶非绝热修正；

<*cutoff*>是截断函数的指数参数。

当<*ncentres*>低于零的时候，只计算积分并保存在1490.1记录中。否则将修改 $h_0$ 矩阵（记录1200.1和1210.1）和双电子积分（记录1300.1）。

CPP,ADD[,< factor >];

使用这个变体，前面计算的极化矩阵的矩阵元可以用可变因子< factor >（默认< factor > = 1）添加到 $h_0$ 矩阵和双电子积分中。特别是CPP,ADD,-1; 可以用来重新获得没有极化贡献的积分。

CPP,SET[,< fcpp >];

一般不需要，但可用于通知MOLPRO在续算后，用什么因子使极化积分在此时有效。

目前CPP积分限定为直到角动量4，即 $g$ 函数的基函数。

## 13.2 ECP/CPP的例子

```
! $Revision: 2006.0 $
***,Na2
! Na2分子的势能曲线，使用1个价电子的ECP + CPP
gprint,basis,orbitals;
rvec=[2.9,3.0,3.1,3.2,3.3] ang
do i=1,#rvec
  rNa2=rvec(i)
  geometry={na;na,na,rNa2}
  basis={
    ecp,na,ecp10sdf;      ! ecp输入
    s,na,even,8,3,.5;    ! basis输入
    p,na,even,6,3,.2;
    d,na,.12,.03;
  }
  cpp,init,1;            ! CPP输入
  na,1,.9947,,.62;
  hf;
  ehf(i)=energy
  {cisd;core;}
  eci(i)=energy
enddo
table,rvec,ehf,eci
---
```

[http://www.molpro.net/info/current/examples/na2\\_ecp-cpp.com](http://www.molpro.net/info/current/examples/na2_ecp-cpp.com)

在开始任何能量计算之前，必须在GEOMETRY区和BASIS区分别定义结构和基组。双电子积分默认求解一次，并存于磁盘上，这一功能可以用输入命令`gdirect`禁止（参见14.2节），强制在需要时随时求解积分。MOLPRO检查单电子和双电子积分对当前基组和结构是否可用，必要的话自动计算。如果只改变了核电荷，程序也可以自动识别，如同在基组平衡计算的情况一样。在这种情况下，不需要重新计算双电子积分。

### 14.1 分类的积分

如果积分存储在磁盘上，在求解之后立即把它们分类到完全对称组合矩阵中，因此后面使用它们的程序模块可以尽可能地提高效率。如上面所讨论的，通常不需要直接调用积分和分类程序，但有时可能需要附加的选项。它们用`INT`命令指定，应当出现在结构和基组的定义之后，求解能量的任何命令之前。

`INT, [[NO] SORT,] [SPRI=value]`

`SORT, [SPRI=value]`

`INT, NOSORT`; `SORT`可用于把积分求解和分类步骤直接分开做，例如用于收集各自的计算时间信息。在`SPRI`选项中，把`value`设定为大于1的数，会打印所有的双电子积分。

积分分类的详细选项可以用`AOINT`参数集指定，输入形式为

`AOINT, key1=value1, key2=value2, ...`

下面总结了可用的关键词，以及它们的含义和默认值。

<code>c_final</code>	整数，指定用于最终分类积分的压缩算法。可用的值有0（不压缩），1（用1, 2, 4或8字节的值压缩），2（用2, 4或8字节的值压缩），4（用4或8字节的值压缩），和8。默认：0。
----------------------	--

c_sort1	整数，指定用于分类过程中间文件的压缩算法。默认：0。
c_seward	整数，在积分程序书写和分类程序读入中，指定符号标记格式与压缩格式。默认：0。
compress	全部压缩；c_final, c_seward, 和c_sort1在程序内部要求不能小于这个参数。默认：1。
thresh	实数，给出压缩的截断阈值。默认：0.0，表示使用积分求解阈值（GTHRESH, TWOINT）。
io	字符串，定义如何写分类积分。可用的值有molpro（文件1的标准Molpro记录）和eaf（专有访问文件）。eaf只能用于程序对使用MPP进行配置的情况，目前molpro只能用于串行执行。如果在续算任务中需要积分，则必须用molpro。在并行机上为了实现最高的效率，应当使用eaf，因为这种情况下积分分布于各自本地处理器的文件中。

为了向下兼容，还定义了两个有用的命令：COMPRESS等价于AOINT,COMPRESS=1，而 UNCOMPRESS等价于AOINT,COMPRESS=0。

## 14.2 直接积分计算（GDIRECT）

参考文献：

一般的直接方法：M. Schütz, R. Lindh, and H.-J. Werner, Mol. Phys. 96, 719 (1999).  
线性标度LMP2：M. Schütz, G. Hetzer, and H.-J. Werner J. Chem. Phys. 111, 5691 (1999).

除了完全CI（FCI）和微扰三激发（T）以外，MOLPRO中的所有方法都可以用直接积分方法执行，也就是说，这些方法受AO基中的双电子积分驱动，只要需要就重新计算，以避免磁盘存储这些数据的瓶颈。对小分子，这需要更多的CPU时间，但是当使用大基组时，会减少磁盘空间需求。然而，由于使用了高效的预屏蔽技术，计算量相对于分子尺寸的标度在直接积分模式中要低于常规模式，因此大分子的直接积分计算可能并不比常规计算昂贵。平衡点强烈依赖于分子的尺寸，硬件，以及基组。根据可用的磁盘空间，一个对称性中有150-200个以上基函数的计算通常应当使用直接积分模式。

直接积分计算用DIRECT或GDIRECT指令调用。如果其中一个卡在特定程序的输入区之外给出，那么它的作用是全局的，也就是说，后面所有的计算都用直接积分模式执行。另一方面，如果DIRECT卡是特定程序输入区（例如HF, CCSD）的一部分，那么



它只影响该程序。GDIRECT指令不能被单独的程序识别，它的作用总是全局的。通常，一个任务中的所有计算都将用直接积分模式执行，因此在第一个能量计算之前需要一个DIRECT或GDIRECT卡。但是，为了修改特定程序的选项或阈值，还可以给出更多的DIRECT或GDIRECT指令。

在MOLPRO中执行的直接积分计算包含三个不同的流程：1) Fock矩阵求解 (DFOCK)，2) 积分转换 (DTRAF)，以及，3) 外部交换算符 (DKEXT)。这三个程序都有特定的选项和阈值，但是也可以用一般参数指定最重要的阈值，它们将用作所有的三个程序的默认值。

程序通常自动使用合适的默认值，在大多数情况下，DIRECT指令无需指定参数。但是，为了保证足够的精度，默认的阈值是非常严格的，在大体系的计算中，较大的值对于减少CPU时间是很有用的。

DIRECT指令的格式为：

DIRECT, key1=value1, key2=value2...

下面的表格总结了可用的关键词及其含义。默认值在随后的表格中给出。在不同情况下，有各种层次的默认值。例如，如果THREST\_D2EXT没有给出，则使用下面中的一个：[THR\_D2EXT, THREST\_DTRAF, THR\_DTRAF, THREST, *default*]。括号中的列表从左到右依次检查，使用在输入中发现的第一个值。*default*是与能量阈值和基组有关的默认值（如果重叠矩阵包含非常小的本征值，则减小阈值）。

一般选项（用于所有程序）：

THREST	积分预屏蔽阈值。如果最大的估算积分值（基于Cauchy-Schwarz不等式）与对壳层区域有贡献的最大密度矩阵元的乘积小于这个值，则跳过积分壳层区块的计算。在DTRAF和DKEXT中，有效密度矩阵分别从MO因子和振幅构建。
THRINT	积分预屏蔽阈值。它应用于精确（也就是计算的）积分值与密度矩阵的乘积。这个阈值仅能用于DTRAF和DKEXT。如果最大积分与对壳层区域有贡献的最大有效密度矩阵元的乘积小于这个阈值，则跳过这个积分的壳层区块。如果设为负值，计算的积分不会被忽略。
THRPROD	积分与MO因子（DTRAF）或振幅（DKEXT）的乘积的预屏蔽阈值。如果壳层区块的最大积分与最大因子的乘积小于这个值，则忽略MO因子或振幅的壳层区块。如果此值设置为负，则不执行乘积屏蔽。

THRMAX	在迭代方法（SCF，CI，CCSD）中，预屏蔽阈值THREST对DFOCK和DKEXT的初始值。如果非零，它也用作MP3和MP4(SDQ)计算中的DKEXT值。一旦达到某个精度（参见VARRED），或者进行了MAXRED次迭代之后，阈值将减小到THREST。在CI和CCSD计算中，初始阈值THRINT_DKEXT和THRPROD_DKEXT也受这个值的影响。说明参见THRMAX_DKEXT。如果THRMAX=0，从一开始就对所有方法使用最终阈值。
SCREEN	使用或禁止使用预屏蔽。 SCREEN ≥ 0: 使用完全屏蔽。 SCREEN < 0: 不使用THRPROD。在直接SCF中无密度屏蔽。 SCREEN < -1: 不使用THRINT。 SCREEN < -2: 不使用THREST。
MAXRED	在CI和CCSD计算中，阈值减小到其最终值后的最大迭代数。如果MAXRED=0，最终的阈值从一开始就用于CI和CCSD（同THRMAX=0，但MAXRED不影响DSCF。后者使用固定值10）。
VARRED	如果振幅变化的平方和小于这个值，阈值减小到其最终值。
SWAP	在SEWARD中，使用或禁止使用标签交换。仅用于测试的目的。

#### 直接SCF（DFOCK）的特殊选项:

THREST_DSCF	直接SCF中的最终预屏蔽阈值。如果给出，它将替代THREST的值。
THRMAX_DSCF	直接SCF中的初始预屏蔽阈值。它用于最初的7到10个迭代。一旦达到某个精度，阈值就减小为THREST_DSCF。
SWAP_DFOCK	在Fock矩阵的计算中，使用或禁止使用标签交换（仅用于测试的目的）。

#### 直接积分转换（DTRAF）的一般选项:

PAGE_DTRAF	选择转换的方法。 PAGE_DTRAF=0: 使用最小内存算法，需要四个积分求解。 PAGE_DTRAF=1: 使用分页算法，有最少的CPU时间（对DMP2/LMP2是一个次积分求解，其它情况是两个）。
SCREEN_DTRAF	如果给出，将会对DTRAF替代SCREEN的值。

MAXSHLQ1_DTRAF	在第一个四分之一的转换步骤中，合并壳层的最大尺寸（0：不使用）。
MINSHLQ1_DTRAF	如果壳层的尺寸小于这个值，则只进行壳层合并（0：不使用）。
MAXSHLQ2_DTRAF	在第二个四分之一的转换步骤中，合并壳层的最大尺寸（0：不使用）。
MINSHLQ2_DTRAF	如果壳层的尺寸小于这个值，则只进行壳层合并（0：不使用）。
MAXCEN_DTRAF	在合并的壳层中，中心的最大数量（0：无限制）。
PRINT_DTRAF	打印DTRAF的参数。

#### 所有直接积分变换的一般阈值：

THR_DTRAF	DTRAF的一般阈值。如果给出将用于以下所有阈值的默认值。
THREST_DTRAF	DTRAF的AO预屏蔽阈值。 默认：[THR_DTRAF, THREST, <i>default</i> ]
THRINT_DTRAF	DTRAF的积分阈值。 默认：[THR_DTRAF, THRINT, <i>default</i> ]
THRPROD_DTRAF	DTRAF的乘积阈值。 默认：[THR_DTRAF, THRPROD, <i>default</i> ]

#### 直接积分转换的特殊阈值：

THR_D2EXT	产生双外部积分的一般阈值。如果给出，则用作下面所有D2EXT阈值的默认值。
THREST_D2EXT	产生双外部积分的预屏蔽阈值。 默认：[THR_D2EXT, THREST_DTRAF, THR_DTRAF, THREST, <i>default</i> ]
THRINT_D2EXT	产生双外部积分的积分阈值。 默认：[THR_D2EXT, THRINT_DTRAF, THR_DTRAF, THRINT, <i>default</i> ]
THRPROD_D2EXT	产生双外部积分的乘积阈值。 默认：[THR_D2EXT, THRPROD_DTRAF, THR_DTRAF, THRPROD, <i>default</i> ]
THR_D3EXT	产生三外部积分的一般阈值。如果给出，则用作下面所有D3EXT阈值的默认值。
THREST_D3EXT	产生三外部积分的预屏蔽阈值。 默认：[THR_D3EXT, THREST_DTRAF, THR_DTRAF, THREST, <i>default</i> ]

THRINT_D3EXT	产生三外部积分的积分阈值。 默认: [THR_D3EXT, THRINT_DTRAF, THR_DTRAF, THRINT, <i>default</i> ]
THRPROD_D3EXT	产生三外部积分的乘积阈值。 默认: [THR_D3EXT, THRPROD_DTRAF, THR_DTRAF, THRPROD, <i>default</i> ]
THR_D4EXT	产生四外部积分的一般阈值。如果给出, 则用作下面所有D4EXT阈值的默认值。
THREST_D4EXT	产生四外部积分的预屏蔽阈值。 默认: [THR_D4EXT, THREST_DTRAF, THR_DTRAF, THREST, <i>default</i> ]
THRINT_D4EXT	产生四外部积分的积分阈值。 默认: [THR_D4EXT, THRINT_DTRAF, THR_DTRAF, THRINT, <i>default</i> ]
THRPROD_D4EXT	产生四外部积分的乘积阈值。 默认: [THR_D4EXT, THRPROD_DTRAF, THR_DTRAF, THRPROD, <i>default</i> ]
THR_DCCSD	在每一CCSD迭代中所需的广义转换的一般阈值。如果给出, 则用作下面THREST_DCCSD, THRINT_DCCSD和 THRPROD_DCCSD的一般阈值。
THREST_DCCSD	DCCSD转换的预屏蔽阈值。 默认: [THR_DCCSD, THREST_DTRAF, THR_DTRAF, THREST, <i>default</i> ]
THRINT_DCCSD	DCCSD转换的积分阈值。 默认: [THR_DCCSD, THRINT_DTRAF, THR_DTRAF, THRINT, <i>default</i> ]
THRPROD_DCCSD	DCCSD转换的乘积阈值。 默认: [THR_DCCSD, THRPROD_DTRAF, THR_DTRAF, THRPROD, <i>default</i> ]
THRMAX_DCCSD	在CCSD计算中, THREST_DCCSD的初始值。一旦达到某个精度 (参见VARRED), 或者进行了MAXRED次迭代之后, 阈值将减小到THREST_DCCSD。初始阈值 THRINT_DCCSD和THRPROD_DCCSD通过其输入值 (或默认值) 乘以THRMAX_DCCSD/THREST_DCCSD获得, 但限制是初始值不能小于最终值。

### 直接MP2 (DMP2) 的特殊选项:

DMP2	选择直接MP2的转换方法: DMP2=-1: 根据可用的内存自动选择。 DMP2=0: 对DMP2使用完全直接方法 (最少两次积分求解, 可能求解多次, 无磁盘空间)。
------	--

	DMP2=1: 对DMP2使用半直接方法（一至四次积分求解，依赖于PAGE_DTRAF）。
	DMP2=2: 在DMP2中用DKEXT计算交换算符（一次积分求解）。这只能用在有多个远距离轨道对的局域DMP2计算中。
THR_DMP2	在DMP2中产生双外部积分的一般阈值。如果给出，则用作以下所有DMP2阈值的默认值。
THREST_DMP2	产生双外部积分的预屏蔽阈值。 默认: [THR_DMP2, THREST_DTRAF, THR_DTRAF, THREST, <i>default</i> ]
THRINT_DMP2	产生双外部积分的积分阈值。 默认: [THR_DMP2, THRINT_DTRAF, THR_DTRAF, THRINT, <i>default</i> ]
THRPROD_DMP2	产生双外部积分的乘积阈值。 默认: [THR_DMP2, THRPROD_DTRAF, THR_DTRAF, THRPROD, <i>default</i> ]

#### 直接局域MP2（LMP2）的特殊选项:

DTRAF	选择直接LMP2的转换方法: DTRAF $\geq 0$ : 首先在AO基中产生双外部积分（交换算符），随后在第二步转换到投影的局域基中。因此磁盘存储需求量与分子的尺寸呈三次方关系。 DTRAF = -1: 在投影基中直接产生双外部积分（交换算符）。因此磁盘存储需求量与分子的尺寸呈线性关系。对于非常大的分子，它（加上PAGE_DTRAF = 0）是推荐的算法（参看第二十九章的线性标度LMP2）。 DTRAF = -2: 在投影基中直接产生交换算符的另外一种算法。这个算法通常比用DTRAF = -1选择的算法在计算上更昂贵。注意，DTRAF = -1和DTRAF = -2都不能在LMP2梯度中使用。
THR_LMP2	在线性标度LMP2中，产生双外部积分的一般阈值。如果给出，则用作下面所有LMP2阈值的默认值。
THREST_LMP2	产生双外部积分的预屏蔽阈值。 默认: [THR_LMP2, THREST_DTRAF, THR_DTRAF, THREST, <i>default</i> ]
THRQ1_LMP2	用于第一个四分之一转换的阈值。 默认: [THR_LMP2, THRPROD_DTRAF, THR_DTRAF, THRPROD, <i>default</i> ]
THRQ2_LMP2	用于第二个和随后的四分之一转换的阈值。 默认: [THR_LMP2, THRINT_DTRAF, THR_DTRAF, THRINT, <i>default</i> ]

THRAO\_ATTEN      衰减积分( $\mu\mu|\nu\nu$ )预屏蔽的特殊阈值。  
默认: THREST\_LMP2

#### 外部交换算符的直接积分计算 (DKEXT) 的选项:

DKEXT      选择DKEXT的驱动。  
DKEXT=-1: 使用分页算法 (内存最少)。如果对芯内算法需要一个以上的积分传递, 这是自动使用的。  
DKEXT=0: 使用芯内算法, 无三次积分。  
DKEXT=1: 使用芯内算法和三次积分。  
DKEXT=2: 如果三次积分中至少两个积分不同, 则使用芯内算法和三次积分。  
DKEXT=3: 如果三次积分中所有积分都不同, 则使用芯内算法和三次积分。

SCREEN\_DKEXT      如果给出, 则对DKEXT替代SCREEN的值。

MAXSIZE\_DKEXT      DKEXT中合并壳层的最大尺寸 (0: 不使用)。

MINSIZE\_DKEXT      只有尺寸小于这个值的壳层, 才进行合并 (0: 不使用)。

MAXCEN\_DKEXT      在被合并的壳层中, 中心的最大数量 (0: 无限制)。

SCREEN\_DKEXT      在DKEXT中使用或禁止使用屏蔽。

PRINT\_DKEXT      打印DKEXT的参数。

SWAP\_DKEXT      在DKEXT中使用或禁止使用标签交换 (仅用于测试目的)。

MXMBLK\_DKEXT      在DKEXT中最大的矩阵块大小 (仅用于DKEXT $\geq$ 1)。

#### 外部交换算符的直接积分计算 (DKEXT) 的阈值:

THR\_DKEXT      DKEXT的一般阈值。如果给出, 则用作下面所有DKEXT阈值的默认值。

THREST\_DKEXT      DKEXT的预屏蔽阈值。  
默认: [THR\_DKEXT, THREST, *default*]

THRINT\_DKEXT      DKEXT的积分阈值。  
默认: [THR\_DKEXT, THRINT, *default*]

THRPROD\_DKEXT      DKEXT的乘积阈值。  
默认: [THR\_DKEXT, THRPROD, *default*]

THRMAX\_DKEXT 在CI和CCSD计算中，THREST\_DKEXT的初始值。如果非零，它还用于MP3 和MP4(SDQ)计算中的DKEXT。一旦达到某个精度(参见VARRED)，或者进行了MAXRED次迭代之后，阈值将减小到THREST\_DKEXT。初始阈值THRINT\_DKEXT和THRPROD\_DKEXT通过其输入值(或默认值)乘以THRMAX\_DKEXT/THREST\_DKEXT 获得，但初始值不能小于最终值。

由于历史原因，很多选项还有别名。下面的表格总结了所有选项和阈值的默认值，并给出可能的别名。

表 14.1: direct选项的默认值和别名

参数	别名	默认值
SCREEN		1
MAXRED		7
VARRED		1.d-7
SWAP		1
SWAP_DFOCK		SWAP
DMP2	DTRAF	-1
PAGE_DTRAF	PAGE	1
SCREEN_DTRAF		SCREEN
MAXSHLQ1_DTRAF	NSHLQ1	32
MINSHLQ1_DTRAF		0
MAXSHLQ2_DTRAF	NSHLQ2	16
MINSHLQ2_DTRAF		0
MAXCEN_DTRAF		0
PRINT_DTRAF		-1
SWAP_DTRAF		SWAP
DKEXT	DRVKEXT	3
SCREEN_DKEXT		SCREEN
MAXSIZE_DKEXT		0
MINSIZE_DKEXT		5
MAXCEN_DKEXT		1
PRINT_DKEXT		-1

SWAP\_DKEXT

SWAP

MXMBLK\_DKEXT

依赖于硬件 (molpro命令的-B参数)

表 14.2: direct计算的默认阈值和别名

参数	别名	默认值
THREST	THRAO	$\min(\Delta E \cdot 1.d - 2, 1.d - 9)^{a,b}$
THRINT	THRSO	$\min(\Delta E \cdot 1.d - 2, 1.d - 9)^{a,b}$
THRPROD	THRP	$\min(\Delta E \cdot 1.d - 3, 1.d - 10)^{a,b}$
THRMAX		1.d-8 <sup>b</sup>
THREST_DSCF	THRDSCF	$\leq 1.d-10$ (依赖于精度和基组)
THRMAX_DSCF	THRDSCF_MAX	THRMAX
THR_DTRAF	THRDTRAF	
THREST_DTRAF	THRAO_DTRAF	[THR_DTRAF, THREST]
THRINT_DTRAF	THRAO_DTRAF	[THR_DTRAF, THRINT]
THRPROD_DTRAF	THRP_DTRAF	[THR_DTRAF, THRPROD]
THR_D2EXT	THR2EXT	THR_DTRAF
THREST_D2EXT	THRAO_D2EXT	[THR_D2EXT, THREST_DTRAF]
THRINT_D2EXT	THRSO_D2EXT	[THR_D2EXT, THRINT_DTRAF]
THRPROD_D2EXT	THRP_D2EXT	[THR_D2EXT, THRPROD_DTRAF]
THR_D3EXT	THR3EXT	THR_DTRAF
THREST_D3EXT	THRAO_D3EXT	[THR_D3EXT, THREST_DTRAF]
THRINT_D3EXT	THRSO_D3EXT	[THR_D3EXT, THRINT_DTRAF]
THRPROD_D3EXT	THRP_D3EXT	[THR_D3EXT, THRPROD_DTRAF]
THR_D4EXT	THR4EXT	THR_DTRAF
THREST_D4EXT	THRAO_D4EXT	[THR_D4EXT, THREST_DTRAF]
THRINT_D4EXT	THRSO_D4EXT	[THR_D4EXT, THRINT_DTRAF]
THRPROD_D4EXT	THRP_D4EXT	[THR_D4EXT, THRPROD_DTRAF]
THR_DCCSD	THRCSD	THR_DTRAF
THREST_DCCSD	THRAO_DCCSD	[THR_DCCSD, THREST_DTRAF]
THRINT_DCCSD	THRSO_DCCSD	[THR_DCCSD, THRINT_DTRAF]
THRPROD_DCCSD	THRP_DCCSD	[THR_DCCSD, THRPROD_DTRAF]
THRMAX_DCCSD	THRMAX_DTRAF	THRMAX



THR_DMP2	THRDMP2	THR_DTRAF
THREST_DMP2	THRAO_DMP2	[THR_DMP2, THREST_DTRAF, <i>default</i> <sup>c)</sup> ]
THRINT_DMP2	THRSO_DMP2	[THR_DMP2, THRINT_DTRAF, <i>default</i> <sup>c)</sup> ]
THRPROD_DMP2	THRP_DMP2	[THR_DMP2, THRPROD_DTRAF, <i>default</i> <sup>c)</sup> ]
THR_LMP2	THRLMP2	THR_DTRAF
THREST_LMP2	THRAO_LMP2	[THR_LMP2, THREST_DTRAF, <i>default</i> <sup>c)</sup> ]
THRQ1_LMP2	THRQ1	[THR_LMP2, THRPROD_DTRAF, <i>default</i> <sup>c)</sup> ]
THRQ2_LMP2	THRQ2	[THR_LMP2, THRINT_DTRAF, <i>default</i> <sup>c)</sup> ]
THRAO_ATTEN	THRATTEN	THREST_LMP2
THR_DKEXT	THRKEXT	
THREST_DKEXT	THRAO_DKEXT	[THR_DKEXT, THREST]
THRINT_DKEXT	THRSO_DKEXT	[THR_DKEXT, THRINT]
THRPROD_DKEXT	THRP_DKEXT	[THR_DKEXT, THRPROD]
THRMAX_DKEXT		THRMAX

<sup>a)</sup>  $\Delta E$  是所需的能量精度（默认 1.d-6）。

<sup>b)</sup> 如果重叠矩阵有较小的本征值，则减少阈值。

<sup>c)</sup> DMP2 和 LMP2 的默认阈值是  $0.1 \cdot \Delta E$ 。

### 14.2.1 直接积分计算的例子

```
! $Revision: 2006.0 $
memory,2,m
$method=[hf,mp2,ccsd,qci,bccd,multi,mrci,acpf,rs3] ! 一些方法
basis=vdz ! 基组
geometry={o;h1,o,r;h2,o,r,h1,theta} ! 几何结构
gdirect ! direct选项
r=1 ang,theta=104 ! 键长和键角
do i=1,#method ! 对方法做循环
$method(i) ! 运行方法(i)
e(i)=energy ! 把结果保存到变量中
dip(i)=dmz
enddo
table,method,e,dip ! 把结果打印成表格
```

[http://www.molpro.net/info/current/examples/h2o\\_direct.com](http://www.molpro.net/info/current/examples/h2o_direct.com)

任务产生以下的表格：

METHOD	E	DIP
HF	-76.02145798	0.82747348

MP2	-76.22620591	0.00000000
CCSD	-76.23580191	0.00000000
QCI	-76.23596211	0.00000000
BCCD	-76.23565813	0.00000000
MULTI	-76.07843443	0.76283026
MRCI	-76.23369819	0.76875001
ACPF	-76.23820180	0.76872802
RS3	-76.23549448	0.75869972

## 第十五章

## 密度拟合

在自旋限制的Hartree-Fock (HF)，密度泛函理论 (KS)，二阶Møller-Plesset微扰理论 (MP2和RMP2)，显关联MP2 (MP2-F12)，以及所有级别的闭壳层局域关联方法 (LCC2, LMP2-LMP4, LQCISD(T), LCCSD(T)) 中，密度拟合可用于近似计算积分。调用密度拟合需要为命令的名称加上前缀DF-，例如DF-HF，DF-KS，DF-MP2，等等。梯度可以用于DF-HF，DF-KS，和DF-LMP2。密度拟合程序不能使用对称性。因此，如果输入中发现了DF-，则自动关闭对称性。

默认自动选择拟合基组，它们对应于当前的轨道基组，并且适于所用的计算方法。例如，如果轨道基组是VTZ，那么DF-HF 或DF-KS默认的拟合基组是VTZ/JKFIT，而DF-MP2的默认拟合基组是VTZ/MP2FIT。可以用DF\_BASIS选项选择基组库中的其它拟合基组，例如：

```
BASIS=VTZ           ! 使用VTZ轨道基组
DF-HF,DF_BASIS=VQZ   ! 使用VQZ/JKFIT拟合基组
DF-MP2,DF_BASIS=VQZ  ! 使用VQZ/MP2FIT拟合基组
```

程序接下来自动选择适用于当前方法的基组。基组类型也可以加到基组名称后，以后代替默认值。例如，

```
DF-HF,DF_BASIS=VQZ/JKFIT  !使用VQZ/JKFIT拟合基组
```

选择轨道基组可以输入ORBITAL（但这通常不推荐!）。收缩/非收缩可以在基组名后强制加上(CONTRACT)或(UNCONTRACT)，例如，

```
DF_BASIS=AVQZ(UNCONTRACT)/ORBITAL
```

如果在括号中给出了其它选项，那么可以用逗号分开，例如，

```
DF_BASIS=AVQZ(f/d,UNCONTRACT)/ORBITAL
```

其它可以用的形式还有

```
DF_BASIS=AVQZ(f/d)(UNCONTRACT)/ORBITAL
```

或

```
DF_BASIS=AVQZ(f/d)/ORBITAL(UNCONTRACT)
```

注意，(CONTRACT)/(UNCONTRACT)选项不能用于基组区（见下）以前定义的基组名。

也可以在前面的基组区定义拟合基组（参见第十一章），以后可以引用数据集的名称。例如：

```
DF-HF, DF_BASIS=MYJKBASIS  
DF-MP2, DF_BASIS=MYMP2BASIS
```

其中的MYJKBASIS和MYMP2BASIS就是在基组区定义的数据集。在这种情况下，用户有责任保证基组适用于所选的方法。

在第15.1节有更多选项的完整描述，它们可以加在命令行上。这种情况下它们仅对当前的命令有效。选项也可以在另外的DFIT指令中指定。如果它们在一个命令区的内部给出，那么选项只能用于当前程序；这完全等价于在命令行指定选项的情况。但是，如果在命令区外部给出了DFIT（或GDFIT）指令，那么指定的选项是全局的，用于整个运行中所有后面的密度拟合计算。

全局DFIT指令指定的选项也遗传给流程。但是，如果在一个流程中给出了DFIT，那么相应的选项只能用于同一个流程以及其中调用的流程。流程中止后，恢复上一级的选项。

## 15.1 密度拟合的选项

这一部分描述的选项具有灵敏的默认值，通常不需要给出。以下讲到的许多选项具有别名，可以用HELP,CFIT,ALIASES察看。

### 15.1.1 选择拟合基组的选项

BASIS	用于拟合的基组（默认：对应于轨道基的数集）
BASIS_COUL	库仑拟合基组（默认：BASIS）
BASIS_EXCH	交换拟合基组（默认：BASIS）
BASIS_MP2	DF-MP2的拟合基组（默认：BASIS）
BASIS_CCSD	DF-LCCSD的拟合基组（默认：BASIS）

### 15.1.2 屏蔽阈值

THRAO	在AO基中忽略收缩3-指数积分的阈值（默认：1.d-8）。
THRMO	忽略半转换3-指数积分的阈值（默认：1.d-8）。
THRSW	Schwarz屏蔽阈值（默认：1.d-5）。
THROV	在AO基中忽略2-指数积分的阈值（默认：1.d-10）。
THRPROD	第一个半转换的乘积屏蔽阈值（默认：1.d-8）。

用于特定程序的类似的阈值可以用上面的关键字加上后缀，说明如下：

_SCF	DF-HF/DF-KS中的库仑拟合和交换拟合
_COUL	DF-HF/DF-KS中的库仑拟合
_EXCH	DF-HF/DF-KS中的交换拟合
_CPHF	CPHF中的库仑拟合和交换拟合
_SCFGRD	DF-HF/DF-KS梯度中的库仑拟合和交换拟合

它们的默认值与一般阈值的默认值相同。

更多的阈值：

THR2HLF	在交换拟合中，第二个半转换的阈值（默认：THRAO_SCF）
THRASM_SCF	交换矩阵局部集合的阈值（默认：THRAO_SCF）
THRAO_FOCK	DF-KS中的库仑拟合阈值（默认：MIN(THRAO_SCF*1.d-2, 1.d-12)）

### 15.1.3 局域拟合参数

局域拟合的说明见：

H.-J. Werner, F. R. Manby, and P. J. Knowles, J. Chem. Phys. **118**, 8149 (2003),  
 Polly, H.-J. Werner, F. R. Manby, and Peter J. Knowles, Mol. Phys. **102**, 2311 (2004),  
 和

M. Schütz, H.-J. Werner, R. Lindh and F. R. Manby, J. Chem. Phys. **121**, 737 (2004). 可以通过设定LOCFIT=1激活。默认禁止局域拟合，因为在某些情况下可能会导致无法接受的错误。例如，局域拟合千万不要用于基组平衡计算，因为在虚原子上缺少拟合函数，这会导致错误的结果。

使用以下的选项，局域拟合可以对某些程序加以限制：

LOCFIT	如果为正，在所有的程序中只要可能就使用局域拟合（默认：0）。
LOCFIT_SCF	如果为正，在SCF中使用局域拟合（默认：LOCFIT）。
LOCFIT_MP2	如果为正，在DF-LMP2中使用局域拟合。1：使用轨道域；2：使用轨道对域（默认：LOCFIT）。
LOCFIT_F12	如果为正，在DF-LMP2-F12中使用局域拟合（默认：LOCFIT）。
LOCFIT_CCSD	如果为正，在DF-LCCSD中使用局域拟合（默认：LOCFIT）。
LOCFIT_2EXT	如果为正，在LCCSD 2外部转换中使用局域拟合（默认：LOCFIT_CCSD）。
LOCFIT_3EXT	如果为正，在LCCSD 3外部转换中使用局域拟合（默认：LOCFIT_CCSD）。
LOCFIT_4EXT	如果为正，在LCCSD 4外部转换中使用局域拟合（默认：LOCFIT_CCSD）。
LOCFIT_CPHF	如果为正，在CPHF中使用局域拟合（默认：LOCFIT）。
LOCFIT_SCFGRD	如果为正，在梯度计算中使用局域拟合（默认：LOCFIT）。
LOCORB	如果为正，在DF-HF中使用局域拟合（默认：1）。
LOCTRA	如果为正，在第一个半转换中使用局域拟合（默认：LOCFIT）。
DSCREEN	如果为正，在LMP2中使用密度屏蔽（默认：0）。
KSCREEN	如果为正，在LMP2中使用拟合基Schwarz屏蔽（默认：依赖于LOCTRA）。

#### 15.1.4 拟合域的参数

下面的选项可用于修改局域拟合使用的域。这些参数只有当LOCFIT=1才起作用。局域拟合的域由两步确定：首先确定主要轨道域。在LMP2和LCCSD程序中，主要轨道域与激发域使用的相同，通过Boughton-Pulay程序确定，说明见第二十九章。分别根据LMP2和LCCSD的值FITDOM\_MP2或FITDOM\_CCSD，要么直接使用轨道域，要么产生

联合轨道对域。在DF-HF中，主要轨道域包含了Löwdin电荷大于或等于THRCHG\_SCF的所有原子基函数。在第二步，主要拟合域用距离标准（RDOMAUX，单位Bohr）或键连接标准（IDOMAUX）进行扩展。IDOMAUX=1表示只要原子到主要域的键长至少有一个满足要求，就包含原子的函数。默认使用距离标准。但是，如果IDOMAUX.ge.0，则忽略距离标准，改用连接性标准。

THRCHG_SCF	在局域交换拟合中，用来选择主要轨道域的参数（默认：0.1）。Löwdin 电荷大于此值的所有原子都包含在内。主要域根据RDOMAUX_SCF或IDOMAUX_SCF 进行扩展。
FITDOM_MP2	在LMP2转换中，用来选择主要拟合域的参数（默认：3）。 1：使用轨道域； 2：使用强轨道对的联合轨道域； 3：使用强、弱轨道对的联合轨道域（默认：3）。 主要域按照RDOMAUX_MP2或IDOMAUX_MP2进行扩展。
FITDOM_CCSD	类似于FITDOM_MP2，但用于LCCSD 2-外部转换。
RDOMAUX_SCF	在SCF中用于拟合域扩展的距离标准（默认：5.0）。
IDOMAUX_SCF	在SCF中用于拟合域扩展的连接性标准（默认：0）。
RDOMAUX_CORE	在SCF中用于芯轨道拟合域扩展的距离标准（默认：RDOMAUX_SCF）。
IDOMAUX_CORE	在SCF中用于芯轨道拟合域扩展的连接性标准（默认：IDOMAUX_SCF）。
RDOMSCF_START	在初始SCF迭代中，用于拟合域扩展的距离标准（默认：3.0）。
IDOMSCF_START	在初始SCF迭代中，用于拟合域扩展的连接性标准（默认：1）。
RDOMSCF_FINAL	在最终SCF迭代中，用于拟合域扩展的距离标准（默认：RDOMAUX_SCF）。
IDOMSCF_FINAL	在最终SCF迭代中，用于拟合域扩展的连接性标准（默认：IDOMAUX_SCF）。
RDOMAUX_MP2	在LMP2中用于拟合域扩展的距离标准。默认值依赖于FITDOM_MP2。
IDOMAUX_MP2	在LMP2中用于拟合域扩展的连接性标准。默认值依赖于FITDOM_MP2。

RDOMAUX_CCSD	在LCCSD中用于拟合域扩展的距离标准。默认值依赖于FITDOM_CCSD。
IDOMAUX_CCSD	在LCCSD中用于拟合域扩展的连接性标准。默认值依赖于FITDOM_CCSD。
RDOMAUX_CPHF	在CPHF中用于拟合域扩展的距离标准（默认：3.0）。
RDOMAUX_SCFGRD	在梯度中用于拟合域扩展的距离标准（默认：5.0）。
SCSGRD	把DF-LMP2解析梯度切换到Grimmes SCS的换算MP2 能量泛函（默认：0）。

#### 15.1.5 各种控制选项

有大量的参数。其中绝大部分通常不需要修改，因此这里只对其中一部分进行说明。完整的列表可以用HELP,CFIT察看。



## 第十六章

## SCF程序

Hartree-Fock自洽场程序可以用以下命令之一调用：

HF或RHF	调用自旋限制Hartree-Fock程序
UHF或UHF-SCF, <i>options</i>	调用自旋非限制Hartree-Fock程序

与旧版本MOLPRO相反，HF和RHF指令具有相同的功能，都可用于闭壳层或开壳层计算。其它的别名是HF-SCF或RHF-SCF。

通常，不需要进一步的输入。默认电子数等于核电荷数，波函假定是全对称的（第一个不可约表示），自旋多重度对偶数个电子是1（单重态），否则是2（双重态）。使用构造原理（Aufbau principle）确定每个对称性的占据数。这对于闭壳层的情况通常很好，但是有时会得到错误的占据数，或者波函在两个不同的轨道空间交替变换。对于这种情况必须使用OCC指令，强制收敛到所需的态。程序默认的属性可以用命令行选项或用指令修改。

对于开壳层情况，我们推荐使用WF, OCC, CLOSED, 或OPEN 卡，用来唯一地定义波函。其它常用的命令还有START和ORBITAL（或SAVE），分别用于修改初始轨道和优化轨道的默认记录。SHIFT选项或指令允许在RHF程序中修改能级移动，EXPEC可用于计算单电子算符的期待值（参见第6.13节）。

密度拟合可用于闭壳层和开壳层自旋限制的HF，调用时加上前缀DF-（DF-HF或DF-RHF，参见第十五章）。对于UHF，只能做库仑拟合（CF-UHF）。对于大分子，密度拟合能极大地加速计算。对使用高角动量函数的大基组，可以看到极大的节省时间。细节见R. Polly, H.-J. Werner, F. R. Manby, and Peter J. Knowles, *Fast Hartree-Fock theory using local density fitting approximations*, Mol. Phys. **102**, 2311 (2004)。结果来

自DF-HF或DF-KS计算的所有出版物应当引用这一文献。

## 16.1 选项

在这一部分讲述HF|RHF|UHF的选项。关于影响Kohn-Sham计算的更多选项，参见第十七章。为了与以前的MOLPRO版本兼容，选项也可以在随后的指令中给出，在后面的章节说明。

### 16.1.1 控制HF收敛的选项

- ACCU[RACY]=*accu*      密度矩阵的收敛阈值（密度矩阵元变化平方和）。若  $accu > 1$ ，使用阈值 $10^{-accu}$ 。默认值依赖于全局 ENERGY 阈值。
- ENERGY=*thrden*      能量的收敛阈值。默认值依赖于全局ENERGY阈值。
- START=*record*      保留初始轨道的记录。
- SAVE|ORBITAL=*record*      轨道的转存记录。
- MAXIT=*maxit*      最大的迭代数（默认60）。
- SHIFTA|SHIFTC=*shifta*      RHF闭壳层轨道和 $\alpha$ 自旋UHF轨道的能级移动，默认分别为-0.3和0。
- SHIFTB|SHIFTO=*shiftb*      RHF开壳层轨道和 $\beta$ 自旋UHF轨道的能级移动，默认为0。
- NITORD|NITORDER=*nitord*      在开壳层计算中，每轮迭代后轨道都重新排序，使得与前一轮迭代的轨道有最大的重叠。这只在*nitord*轮迭代后才起作用。默认值依赖于初始猜测的质量。
- NITSH|NITSHIFT=*nitsh*      如果迭代计数小于*nitsh*，那么能级移动设为0。默认值依赖于初始猜测的质量。
- NITCL|NITCLOSED=*nitcl*      如果迭代计数小于*nitcl*，那么仅使用Fock矩阵的闭壳层部分（默认*nitcl*= 0）。
- NITOC= *nitocc*      在刚开始的前*nitocc*次迭代固定占据方式。默认值依赖于初始猜测的质量。

NITORT|NITORTH=*nitort*

每进行*nitort*次迭代，就对轨道进行一次重新正交归一化。默认*nitort*= 10

POTFAC=*potfac*

在第一次迭代中的势能换算因子（默认1.0）。

注意在续算的情况下，迭代计数从3开始。

### 16.1.2 对角化方法的选项

当使用非常大的基组进行计算时，对角化的时间将占据总CPU时间的很大比例。可以用文献R. Polly, H.-J. Werner, F. R. Manby, and Peter J. Knowles, Mol. Phys. **102**, 2311 (2004)中的轨道转动方法减少时间。

MINROT=*minrot*

如果*minrot*≥ 0，则执行轨道转动方法。完全Fock矩阵的直接对角化在前*minrot*次迭代和最后一次迭代中执行。如果*minrot*=0，则使用默认设置，它依赖于初始猜测。

NEXPR=*nexpr*

在西轨道变换矩阵的指数展开中使用的项的数量（默认4）。

DEROT=*derot*

轨道转动方法中使用的能隙。对HOMO轨道能量在±*derot* hartree以内的轨道构造Fock矩阵并对角化（默认1.0）。

JACOBI=*jacobi*

如果非零，使用Jacobi对角化。

### 16.1.3 收敛加速方法（DIIS）的选项

更多细节参见IPOL指令。

IPTYP=*iptyp*

内插类型（默认DIIS，参见IPOL指令）。

IPNIT|DIIS\_START=*ipnit*

DIIS内插的第一个迭代。

IPSTEP|DIIS\_STEP=*ipstep*

DIIS内插的迭代增量。

MAXDIS|MAXDIIS=*maxdis*

用于DIIS内插的Fock矩阵最大数量（默认10）。

### 16.1.4 积分直接计算的选项

DIRECT

（逻辑型）如果给出，进行积分直接HF计算。

THRMIN | THRDSCF\_MIN=*value*

DSCF的最终积分屏蔽阈值。

THRMAX | THRDSCF\_MAX=*value*

DSCF的初始积分屏蔽阈值。

THRINT | THRDSCF=*value*

同THRDSCF\_MIN。

PRESCREEN=*value* 如果非零，是用密度屏蔽（默认）。

DISKSIZE=*value* 在半直接计算中，以“位”为单位的磁盘最大尺寸（目前禁用）。

BUFSIZE=*value* 在半直接计算中，缓存的最大尺寸（目前禁用）。

THRDISK=*value* 把积分写到磁盘的阈值（目前禁用）。

PRINT\_DFOCK=*value* 直接Fock矩阵计算的打印选项。

#### 16.1.5 UHF计算的特殊选项

NATORB=*record* 把自然电荷轨道保存到给定的记录中。

UNOMIN=*unomin* UNO-CAS的最小占据数（默认0.02）。

UNOMAX=*unomax* UNO-CAS的最大占据数（默认1.98）。

#### 16.1.6 局域密度拟合计算选项

关于密度拟合的更多选项，参见第十五章。以下的选项影响局域密度拟合，方法的说明见H.-J. Werner, F. R. Manby, and P. J. Knowles, J. Chem. Phys. **118**, 8149 (2003)，以及R. Polly, H.-J. Werner, F. R. Manby, and Peter J. Knowles, Mol. Phys. **102**, 2311 (2004)。注意，局域拟合影响精度。

LOCFIT=*locfit* 如果非零，对交换使用局域拟合。如果> 1，还对库仑使用局域拟合（不推荐）。

RDOM=*locfit* 在局域拟合中，拟合域选择的半径（默认5 bohr）。

RDOMC=*locfit* 在局域拟合中，芯轨道拟合域选择的半径（默认RDOM）。

DOMSEL=*domesel* 在局域拟合中，选择轨道域的标准（默认0.1）。

#### 16.1.7 极化率的选项

POLARI=*value*      如果非零。计算解析偶极极化率。此外参见POLARI指令 (16.8节)，它可以指定各种单电子算符（默认使用偶极算符）。

THRCPHF=*thresh*      如果计算极化率，CPHF的阈值（默认1.d-6）。

### 16.1.8 打印选项

PRINT|ORBPRINT=*value*      打印的虚轨道数量。如果*value*=0，打印占据轨道。

DEBUG=*value*      调试选项。

## 16.2 定义波函

波函的电子数和总对称性在WF卡定义：

WF,*elec,sym,spin*

其中，

<i>elec</i>	电子数
<i>sym</i>	<u>不可约表示序号</u>
<i>spin</i>	定义自旋对称性， $spin = 2 * S$ （单重态=0，双重态=1，三重态=2，等）

注意这些值中凡是没有指定的，将采用自动判断的默认值（参见4.8节）。

### 16.2.1 定义每个对称性的占据轨道数

OCC, $n_1, n_2, \dots, n_8$

对于高对称性的情况，为了避免收敛问题，只要事先知道占据方式就要把这个卡包含在输入中。 $n_i$ 是不可约表示*i*中的占据轨道数。轨道的总数必须等于 $(elec+spin)/2$ （参见WF卡）。

### 16.2.2 定义闭壳层轨道

CLOSED, $n_1, n_2, \dots, n_8$

这一选项卡可用于开壳层计算，指定每个对称性中闭壳层轨道的数量。这可以在不用OPEN卡的情况下，强制特定的态。

### 16.2.3 指定开壳层轨道

OPEN,*orb*<sub>1</sub>.*sym*<sub>1</sub>,*orb*<sub>2</sub>.*sym*<sub>2</sub>,...,*orb*<sub>*n*</sub>.*sym*<sub>*n*</sub>

这一选项卡可用于指定单占据轨道。单占据轨道的数量必须等于 $spin$ ，其对称性直积必须等于 $sym$ （参见WF卡）。如果没有出现OPEN卡，则自动选择开壳层轨道。算法将试着寻找基态，但是在可用的轨道中如果有几种开壳层电子分布，可能会得到错误的电子态。这可以通过使用CLOSED卡避免。如果 $orb_i.sym$ 为负，这个轨道将用负自旋占据（仅允许在UHF中）。

## 16.3 保存最终轨道

ORBITAL,*record.file*

SAVE,*record.file*

把优化的轨道及相应的密度矩阵，Fock矩阵和轨道能量，保存到*record.file*中。SAVE是ORBITAL的别名。如果这个卡没有出现，*record*默认为：

RHF	2100	
UHF	2200	（保存 $\alpha$ 和 $\beta$ 自旋轨道，以及有关的量）

在同一输入中，对于随后的每一次同类型的计算，这些数值加1。注意这用于输入文件中的顺序，与执行的顺序无关（参见4.3节）。

*file*的默认值是2。

## 16.4 初始轨道

START指令可用于指定SCF迭代计算的初始轨道。程序既可以产生初始轨道猜测，也可以从以前优化的轨道开始。还可以使用以前的密度矩阵构造第一个Fock算符。

如果不指定START卡，程序会按照以下方式试着寻找合适的初始轨道：

- 首先：      试着从ORBITAL或SAVE卡指定的*record*或它们相应的默认值（参见ORBITAL）中读取轨道。所有的文件都被搜索。
- 第二：      试着从以前的SCF或MCSCF计算寻找轨道。所有的文件都被搜索。
- 第三：      如果没有发现轨道，则初始轨道用近似原子密度或 $h$ 的本征矢产生（见下）。

由于这些默认值通常都是合适的，因此在大多数情况下不需要START卡。

### 16.4.1 初始轨道猜测

可以用以下命令请求初始轨道猜测：

START,[TYPE=]*option*

其中的*option*关键词可以是：

H0                      用*h*的本征矢作为初始猜测。

ATDEN                  使用从原子占据数构造的对角密度矩阵的自然轨道。

对于最小基组或前几个收缩函数对应原子轨道1*s*, 2*s*, 2*p*...的一般收缩基组，例如Dunning的cc-pVnZ集，STO-3G或6-31G集，原子密度猜测非常好。对于这些基组，默认使用ATDEN。如果使用了每个壳层有几个收缩的分块基组，不要指定ATDEN，默认将使用H0。由于*h*的本征矢通常是非常差的初始猜测，因此推荐使用像STO-3G这样的小基组产生初始轨道（参见后面的16.4.2节）。

例子：

```
! $Revision: 2006.3 $
r=1.85,theta=104      ! 设定结构参数
geometry={0;          ! z矩阵结构输入
    H1,0,r;
    H2,0,r,H1,theta}
basis=STO-3G          ! 第一个基组
hf                    ! scf用STO-3G基组
basis=6-311G          ! 第二个基组
hf                    ! scf用6-311G基组
```

[http://www.molpro.net/info/current/examples/h2o\\_sto3gstart1.com](http://www.molpro.net/info/current/examples/h2o_sto3gstart1.com)

第二个计算使用STO-3G计算优化的轨道作为初始猜测。这用默认执行，不需要START卡。直接使用START和SAVE卡的用法在下一部分的例子中演示。

下面的输入完全等价于前一个的例子：

```
! $Revision: 2006.3 $
r=1.85,theta=104      ! 设定结构参数
geometry={0;          ! z矩阵结构输入
    H1,0,r;
    H2,0,r,H1,theta}
basis=STO-3G          ! 第一个基组
hf                    ! scf用STO-3G基组
start,atdens          ! 使用原子密度猜测
save,2100.2           ! 把轨道保存到记录2100.2中
basis=6-311G          ! 第二个基组
hf                    ! scf用6-311G基组
start,2100.2          ! 从前一个STO-3G计算的轨道开始
```

```
save,2101.2          ! 保存优化的轨道到记录2101.2中
```

[http://www.molpro.net/info/current/examples/h2o\\_sto3gstart2.com](http://www.molpro.net/info/current/examples/h2o_sto3gstart2.com)

### 16.4.2 由前一轨道开始

`START,[RECORD=]record.file,[specifications]`

从文件`file`的`record`记录中，读取以前优化的轨道。还可以指定特定的轨道集，说明见4.11节。

指定的转存记录可以和前一计算使用的不同的结构、基组以及对称性。如果没有以前计算的轨道可用，并且ATDENS选项不能使用（见上），那么使用由不同基组得到的初始轨道是非常有用的。

下面的例子显示如何改变SCF计算之间的对称性。当然这个例子几乎是没什么用的，但有些时候，首先获得较高对称性的解会比较容易，接下来转换到较低对称性中作进一步计算。

```
! $Revision: 2008.2 $
r1=1.85,r2=1.85,theta=104      ! 设定结构参数
geometry={0;                    ! z矩阵结构输入
      H1,0,r1;
      H2,0,r2,H1,theta}
basis=vdz
hf                               ! scf使用c2v对称性
orbital,2100.2                  ! 保存到2100.2记录中

set,zsymel=x

hf
start,2100.2                     ! 从c2v对称性的前一个轨道开始
orbital,2101.2                   ! 保存新的轨道

geometry={0;                    ! 必须重新指定结构，使得H1和H2
      H1,0,r1;                  ! 重新标记为对称性相关的原子
      H2,0,r2,H1,theta}
set,zsymel=[x,y]
hf
start,2101.2                     ! 从cs对称性的轨道开始
orbital,2102.2                   ! 保存新的轨道
```

[http://www.molpro.net/info/current/examples/h2o\\_c2v\\_cs\\_start.com](http://www.molpro.net/info/current/examples/h2o_c2v_cs_start.com)

但是注意，这需要分子方位不发生改变时才能正常工作。有时候用`noorient` 选项可能会有帮助。



还要注意，单独一个转存记录不能对不同的基组维度保存轨道。因此在第二个计算中使用save=2100.2将会产生错误。

如果有在临近结构用SCF计算的轨道，它们可以用作初始猜测。

### 16.4.3 由前一密度矩阵开始

START,DENSITY=record.file,[specifications]

从给定的转存记录读入密度矩阵，用于构造第一个Fock矩阵。指定特定密度矩阵的说明参见4.11节。通常不推荐使用DENSITY选项。

## 16.5 转动轨道对

ROTATE,orb<sub>1</sub>.sym,orb<sub>2</sub>.sym,angle

执行对称性sym中初始轨道orb<sub>1</sub>和orb<sub>2</sub>的2 × 2转动，角度为angle。若angle= 0则交换轨道。其它可能的轨道操作参见MERGE。在UHF中，默认只转动β自旋的轨道。转动初始的α自旋轨道可以用

ROTATEA,orb<sub>1</sub>.sym,orb<sub>2</sub>.sym,angle

在这种情况下ROTATEB是ROTATE的同义词。

## 16.6 使用附加的点群对称性

由于MOLPRO只能操作阿贝尔点群，因此有很多对称性不能直接使用。例如，如果线型分子用C<sub>2v</sub>而不是C<sub>∞v</sub>处理，那么δ<sub>(x<sup>2</sup>-y<sup>2</sup>)</sub>轨道就会出现在对称性1（即A<sub>1</sub>）中。对于其它的情况，线型结构可能会作为C<sub>s</sub>对称性计算的特例出现，接下来π轨道的一个分量会在对称性1（即A'）中出现。通过单电子哈密顿量h和重叠矩阵S的分块，程序可以探测被隐藏的“额外”对称性。在每个不可约表示中，“额外”对称性的序号被分配到每个基函数中。这些序号在积分输出的结尾打印。通常，额外的对称性按照基函数的l量子数增加的顺序排列。通过SYM命令，这一信息可用于确定和调整分子轨道的额外对称性。

SYM,irrep,sym(1),sym(2),,sym(n)

sym(i)是不可约表示irrep中，前n个轨道的额外对称性。例如，如果需要线型分子的轨道1.1到3.1是σ，4.1和5.1是δ，那么SYM卡将会读入（计算中用X,Y作为对称生成元）：

SYM,1,1,1,1,2,2

如果必要，程序将重新排列每个不可约表示的轨道，以强制这一占据。可以指定占据轨道和虚轨道的对称性。默认不去除对称性污染。但是如果irrep设置为负值，则去除对称性污染。注意，如果出现简并轨道，这会阻碍收敛。

## 16.7 期待值

EXPEC,*oper*<sub>1</sub>, *oper*<sub>2</sub>, ..., *oper*<sub>n</sub>

计算单电子算符*oper*<sub>1</sub>, *oper*<sub>2</sub>, ..., *oper*<sub>n</sub>的期待值。可用的算符参见6.13节。默认计算偶极矩。如果对其它算符的期待值感兴趣, 通常推荐使用GEXPEC指令。详见6.13节。

## 16.8 极化率

POLARIZABILITY[,*oper*<sub>1</sub>, *oper*<sub>2</sub>, ..., *oper*<sub>n</sub>]

对给定算符*oper*<sub>1</sub>, *oper*<sub>2</sub>, ..., *oper*<sub>n</sub>计算极化率。可用的算符参见6.13节。如果没有指定算符, 默认计算偶极极化率。

目前, 这只能用于没有direct选项的闭壳层计算。

计算的极化率存储在变量POLXX, POLXY, POLXZ, POLYY, POLYZ, 和POLZZ中。

## 16.9 各种指令

这一部分讲述的所有命令都是可选的。通常已经使用了合适的默认值。

### 16.9.1 能级移动

SHIFT,*shifta*,*shiftb*

对 $\alpha$ 和 $\beta$ 自旋轨道分别使用*shifta*和*shiftb* Hartree的能级移动。这可以改善收敛, 但对求解没有影响。典型的值是*shifta*= -0.2到-0.3。闭壳层和开壳层计算的默认值分别是*shifta*= 0和*shifta*= -0.3, *shiftb*= 0。

### 16.9.2 迭代的最大数量

MAXIT,*maxit*

设置迭代的最大数量为*maxit*。默认为*maxit*= 30。

### 16.9.3 收敛阈值

ACCU,*accu*

收敛阈值设置为10\*\*(-*accu*)。这用于密度矩阵元变化量的平方和。默认为*accu*= 10。

### 16.9.4 能量的合理性检查

NOENEST

这会禁止能量的合理性检查，即使是能量值不合理时。否则，默认自动检查能量。

### 16.9.5 打印选项

ORBPRINT,*print*,*test*

这决定了计算结束后打印虚轨道的数量。默认 $print=0$ ，即只打印占据轨道。 $print=-1$ 完全禁止打印轨道。 $test=1$ 在每次迭代后加上打印轨道的作用。

### 16.9.6 内插

IPOL,*iptyp*,*ipnit*,*ipstep*,*maxdis*

这个命令控制DIIS内插。*iptyp*可以是：

DIIS	迭代子空间的直接反演。这是默认的，通常能产生最快的收敛。
DM	废弃。在MOLPRO98中无影响。
HFM	废弃。在MOLPRO98中无影响。
NONE	无内插。

*ipnit*是开始内插的迭代序号。*ipstep*是内插之间的迭代增量。*maxdis*是DIIS矩阵的最大维数（默认是10）。

### 16.9.7 轨道的重新正交归一化

ORTH,*nitort*

在每*nitort*次迭代后，轨道都进行重新正交归一化。默认是*nitort*=10。

### 16.9.8 直接SCF

DIRECT,*options*

如果出现这个卡，计算将以直接模式进行。选项参见14.2节。通常，推荐使用全局GDIRECT命令以进行直接模式。详见14.2节。



## 第十七章

## 密度泛函程序

密度泛函理论计算可以用下面的命令执行：

DFT	计算以前计算的密度的泛函。
RKS或RKS-SCF	调用自旋限制Kohn-Sham程序。KS和KS-SCF是RKS 的同义词。
UKS或UKS-SCF	调用自旋非限制Kohn-Sham程序。

以上每一个命令都可以定义所用泛函的关键词名称，以及更多的选项：

*command, key1, key2, key3, ..., options*

如果没有定义泛函关键词的名称，默认为LDA（见下）。此命令之后可以出现定义密度泛函模块（参见17.2节）或者Hartree-Fock程序（参见16.1节）选项的指令。

在泛函求解完成时，或者Kohn-Sham自洽计算中，各个泛函的值存储在MOLPRO的矢量变量DFTFUNS中；总能量存储于DFTFUN，对应的各个泛函的名称存储于DFTNAME。

能量梯度可用于Kohn-Sham自洽计算。

密度拟合可用于闭壳层和开壳层自旋限制的KS，调用时加上前缀DF-（DF-KS或DF-RKS，参见第十五章）。对于UKS，只能做库仑拟合（CF-UKS）。对于大分子，密度拟合能极大地加速计算。对使用高角动量函数的大基组，可以看到极大的省时。细节见R. Polly, H.-J. Werner, F. R. Manby, and Peter J. Knowles, *Fast Hartree-Fock theory using local density fitting approximations*, Mol. Phys. **102**, 2311 (2004)。结果来自DF-HF或DF-KS计算的所有出版物应当引用这一文献。

积分网格的定义通常使用了灵敏的默认设置。精度可以用17.1 节的选项或17.2节的指令进行控制。更多的控制由GRID 命令提供，说明见17.3节。

## 17.1 选项

可以在KS或UKS的命令行指定以下选项：

GRID= <i>target</i>	指定（每个原子上）网格的目标精度。默认为1.d-6，除非是用全局的THRESH，GRID选项修改。
COARSE( <i>logical</i> )	如果为真，用更粗糙的格点执行最初的几个迭代。默认为 <i>false</i> 。
GRIDMAX= <i>gridmax</i>	在最初的几个迭代中，网格精度为 $\min(\text{gridmax}, \text{target} * \text{coarse-fac})$ （只有设置了COARSE时）。
COARSEFAC= <i>coarsefac</i>	初始网格精度的因子（见上）。默认为1000。
DFTFAC=[ <i>fac1,fac2,...</i> ]	每个泛函的因子。给定值的数目必须与泛函的数目一致。
EXFAC= <i>factor</i>	加到泛函上的精确交换的因子。默认值依赖于泛函。
TOLORB= <i>value</i>	轨道屏蔽的阈值（目前的默认值为1.d-15）。
MATRIX= <i>matrix</i>	选择积分程序的选项。 <i>matrix=0</i> ：使用旧的（较慢的）积分程序； <i>matrix=1</i> ：使用新的矩阵驱动的积分程序（默认）。

此外，还可以给出HF的所有选项（参见16.1节）。

## 17.2 指令

以下的选项可用于控制DFT模块的运行。对于Kohn-Sham的情况，选项可以在第十六章讲述的SCF程序指令之前或之后，以任何顺序出现。

### 17.2.1 密度来源（DENSITY, ODENSITY）

DENSITY,*orb.filec,...*

ODENSITY,*orbo.fileo,...*

对于非自恰的DFT计算，定义密度矩阵的来源。总密度从*orb.filec*文件读取，还有更多选项以标准方法定义密度集，说明见4.11节。ODENSITY可以用来定义自旋密度。默认是最后一次SCF或MCSCF程序写入的密度。

### 17.2.2 阈值（DFTTHRESH）

DFTTHRESH,*key1=value1,key2=value2...*

设置各种截断阈值。*key*可以是以下之一：

TOTAL	密度泛函（对每个原子）的总目标精度。默认是全局阈值GRID，或者GRID选项所指定的值。为了正确使用这个阈值，其它的阈值应当保持为默认零值。
ORBITAL	轨道截断阈值。
DENSITY	密度截断阈值。
FOCK	Fock矩阵截断阈值。

### 17.2.3 精确交换计算 (EXCHANGE)

EXCHANGE, *factor*

对于Kohn-Sham计算，依照Hartree-Fock形式计算交换能，并把用*factor*换算的贡献加入到Fock矩阵和能量泛函中。否则默认值为*factor*=0，也就是说，泛函中假定包含了交换，只需另外计算库仑相互作用。

DFTFACTOR, *fac1*, *fac2*, ...

对每一个指定的泛函提供因子，据此组合泛函。所有的因子默认都是1。

### 17.2.4 区域杂化方法 (RANGEHYBRID)

对于短程 (sr-) DFT和长程 (lr-) 从头方法的耦合，首先必须在sr电子间相互作用 $\sum_{i<j} \text{erf}(\mu r_{ij})/r_{ij}$ 中指定耦合参数 $\mu$ ；这需要设置变量（如mu=0.5）。第二步，通过调用积分程序计算长程ERI（例如，int;erf,mu;）。

接下来用附加的子命令rangehybrid调用RKS程序，执行sr-DFT/lr-HF计算。对于sr-LDA，可用的短程泛函有exerf和ecerf。对于sr-PBE则有exerfpbe和ecerfpbe。和往常一样，需要在rks命令之后指定泛函（例如，rks,exerf,ecerf;）。对底层的短程LDA关联泛函的描述，见S. Pazziani, S. Moroni, P. Gori-Giorgi, G. B. Bachelet, Phys. Rev. B **73**, 155111 (2006)。

最后，在调用所选的后-HF计算中完成sr-DFT/lr-后-HF计算，需要加入两个子命令：在srxcdft之后是所需要的短程泛函（例如srxcdft,exerf,ecerf;），dftden之后是记录编号，表示从这里获取sr泛函的密度。计算的执行可用于ci, mp2, ccscd, ccscd(t)，以及相应的局域MP2和CC方法，可以加上或不加密度拟合。

### 17.2.5 交换-关联势 (POTENTIAL)

POTENTIAL, *rec.fil*

对于独立的DFT计算任务，计算交换-关联势的赝矩阵元，通常定义为所有指定泛函之和相对于原子轨道密度矩阵元的微分。这个矩阵写入*fil*文件的*rec*记录中。

### 17.2.6 网格分块因子 (DFTBLOCK)

DFTBLOCK, *nblock*

在DFT积分程序中，重新指定作为一个区块同时处理的空间积分点的数量（默认128）。增加*nblock*会提高执行效率，例如在矢量计算机中，但会导致内存使用量的增加。

### 17.2.7 转存积分值 (DFTDUMP)

DFTDUMP, *file, status*

把网格点的积分值写到文件*file*中。*file*的第一行包含泛函分量的数量；接下来对每个泛函给出一行泛函输入的关键词。之后的行以Fortran格式(I2,3F15.10,F23.15)给出泛函的代号，笛卡尔坐标，积分值以及积分权重。

## 17.3 数值积分网格控制 (GRID)

密度泛函通过三维空间网格的数值求积分方法求解。虽然灵敏的默认值通常已经足够了，但是定义网格的参数也可以用GRID顶级命令定义，这一命令必须出现在使用网格的DFT或KS命令之前。GRID及其子命令也可以在KS程序内作为指令出现。

GRID, *orb, file, status*

积分网格存储到*orb, file*记录中（默认：1800.2）。磁盘上的信息由两部分组成：定义网格必需的参数，以及计算的网格点和权重的缓存。后者只要有参数和几何结构的改变，都标记为“污染”；如果缓存被污染，当需要使用网格的时候，它将被重新计算，否则使用缓存的值。

如果*status*是OLD，将从前一个计算恢复网格；实际上，旧的网格提供了参数的模版，它可以用下面讲到的参数命令调整。如果*status*是NEW，则网格总是用默认的参数构造。如果*status*是UNKNOWN（默认），当记录*orb, file*不存在时构造新网格，否则使用旧网格。

GRID命令之后可以是一些修改参数的子命令。目前使用的默认参数等价于以下的输入命令。

```
GRIDTHRESH,1e-5,0,0
RADIAL,LOG,3,1.0,20,25,25,30
ANGULAR,LEBEDEV,0.0,0.0
LMIN,3,5,5,7
LMAX,53,53,53,53
VORONOI,10
GRIDSAVE
GRIDSYM
```



### 17.3.1 目标积分精度 (GRIDTHRESH)

GRIDTHRESH, *acc*, *accr*, *acca*

指定积分的目标精度。通过对Slater-Dirac泛函求解给定精度的积分，产生合适的径向和角向网格。*acc*是总的目标精度，通常需要使用；径向和角向网格目标精度从这个算法产生。但是它们可以通过分别指定*accr* 和*acca*来调整。

### 17.3.2 径向积分网格 (RADIAL)

RADIAL, *method*, *m<sub>r</sub>*, *scale*, *n<sub>0</sub>*, *n<sub>1</sub>*, *n<sub>2</sub>*, *n<sub>3</sub>*

定义径向求积分方案的细节。通过指定*method* = EM, BECKE, AHLRICHS或LOG，可以使用四种不同的径向求积分方案，最后一个默认的。

EM是Euler-Maclaurin方案，定义参见C. W. Murray, N. C. Handy and G. J. Laming, Mol. Phys. 78 (1993) 997。*m<sub>r</sub>*的默认值是2，在上面文献的方程(6)中定义为

$$r = \alpha \frac{x^{m_r}}{(1-x)^{m_r}} \quad (17.1)$$

同时，*scale*（默认是1）乘以原子的Bragg-Slater半径给出换算参数 $\alpha$ 。

LOG方案的说明参见M. E. Mura and P. J. Knowles, J. Chem. Phys. 104 (1996) 9848。它基于变换：

$$r = -\alpha \log_e(1 - x^{m_r}), \quad (17.2)$$

并且 $0 \leq x \leq 1$ ，在 $x$ 空间中进行简单Gauss积分求解。对分子体系推荐的*m<sub>r</sub>*值是3，产生Log3网格；对原子*m<sub>r</sub>*=4更合适。 $\alpha$ 采用Mura和Knowles推荐 $\alpha$ 值的*scale*倍，*scale*默认是1。

BECKE的定义参见A. D. Becke, J. Chem. Phys. 88 (1988) 2547。它基于变换

$$r = \alpha \frac{(1+x)}{(1-x)}, \quad (17.3)$$

并且使用 $-1 \leq x \leq +1$ 中的点，对 $x$ 空间的积分求解使用第二类标准Gauss-Chebyshev积分方案。Becke全部采用Bragg-Slater半径的一半作为换算参数，但氢除外，它的换算参

数是整个Bragg-Slater半径。设定 $scale$ 为1之外的值可以使用不同的 $\alpha$ 。对于这个径向方案，不需要 $m_r$ 。

AHLRICHS径向方案的定义参见O. Treutler and R. Ahlrichs, J. Chem. Phys. 102 (1995) 346。它基于M4映射的变换：

$$r = \frac{\alpha}{\log_e 2} (1+x)^{0.6} \log_e \left( \frac{2}{1-x} \right), \quad (17.4)$$

对 $x$ 空间的积分使用第二类标准Gauss-Chebyshev求积分方案。对于这个径向方案，不需要 $m_r$ 。

对于氢/氦，第一行，第二行，和其它元素， $n_0, n_1, n_2, n_3$ 分别是积分 $n_r$ 的级别（参见Murray等的方程(3)）。

当在THR命令中给出 $accr$ ，则定义目标精度，径向点的数量依照模型来选择，而不是直接使用 $n_i$ 。 $n_i$ 越严格，越要使用 $accr$ ，除非其中一个为零，这种情况下会被忽略。

### 17.3.3 角向积分网格 (ANGULAR)

ANGULAR, *method*, *acca*, *crowd*

LMIN,  $l_0^{\min}, l_1^{\min}, l_2^{\min}, l_3^{\min}$

LMAX,  $l_0^{\max}, l_1^{\max}, l_2^{\max}, l_3^{\max}$

指定角向积分方案的细节。*method*的默认选择是LEBEDEV（也即A. D. Becke, J. Chem. Phys. 88 (1988) 2547），它提供了八面体对称性的角向网格。*method*的另一个选择是LEGENDRE，它对 $\theta$ 使用Gauss-Legendre积分，对 $\phi$ 给出简单积分，定义参见C. W. Murray, N. C. Handy and G. J. Lamming, Mol. Phys. 78 (1993) 997。

每一种类型的网格指定了一类单元，各个单元由量子数 $l$ 描述；直到 $l$ 级的球谐函数被精确积分。 $l_{\min_i}$ 和 $l_{\max_i}, i = 0, 1, 2, 3$ 分别对氢/氦，第一行元素，第二行元素，以及其它元素指定了 $l$ 所允许的范围。对于Lebedev 网格，如果 $l$ 不是MOLPRO所用数集中的值(3, 5, 7, 9, 11, 13, 15, 17, 19, 23, 29, 41, 47, 53)，则增加 $l$ 直到给出下一个可用的最大角向网格。一般而言，不同的径向点具有不同的 $l$ ，如果后面的定义中没有适合的 $l$ ，都将采用 $l_i^{\max}$ 。

*crowd*是控制近核区减小积分级别的参数，否则这里的点距离太近而没有必要；较大的*crowd*值表示减小的级别较小从而导致较大的网格。如果该参数使用非常大的值，或者习惯上设置为0，将关闭这个功能。

*acca*是目标能量精度。它用于尽可能地把给定径向点的 $l$ 减小到 $l_i^{\max}$ 以下，但不低于 $l_i^{\max}$ 。在Slater-Dirac交换泛函的核心部分，近似原子密度之和的角向积分误差将用于这个计算。如果*acca*为零，则使用全局阈值，如果未设置全局阈值则忽略。

### 17.3.4 积分网格的原子分区 (VORONOI)

VORONOI,  $m_\mu$

控制Becke-Voronoi空间分割。使用的算法参见C. W. Murray, N. C. Handy and G. J. Laming, Mol. Phys. 78 (1993) 997, 其中的方程(24)定义了 $m_\mu$ 。默认值是10。

### 17.3.5 网格的缓存 (GRIDSAVE, NOGRIDSAVE)

NOGRIDSAVE 禁止网格的磁盘缓存, 也就是说, 只要需要网格, 总是重新计算。

GRIDSAVE 只要可能就使用网格的缓存。

### 17.3.6 网格对称性 (GRIDSYM, NOGRIDSYM)

NOGRIDSYM 在产生网格积分时关闭对称性。而

GRIDSYM 则强制使用点群对称性。

### 17.3.7 网格打印 (GRIDPRINT)

GRIDPRINT,  $key=value, \dots$

控制网格的打印, 默认不执行。目前,  $key$ 唯一的值是GRID,  $value$ 应当定义为整数。GRID=0导致求解积分点的总数并打印报告; GRID=1还显示每个原子上格点的数量; GRID=2打印全部的格点和权重。

## 17.4 密度泛函

以下的 $\rho_\alpha$ 和 $\rho_\beta$ 分别是 $\alpha$ 和 $\beta$ 自旋密度; 总自旋密度是 $\rho$ 。

密度的梯度记为:

$$\sigma_{\alpha\alpha} = \nabla\rho_\alpha \cdot \nabla\rho_\alpha, \sigma_{\beta\beta} = \nabla\rho_\beta \cdot \nabla\rho_\beta, \quad (17.5)$$

$$\sigma_{\alpha\beta} = \sigma_{\beta\alpha} = \nabla\rho_\alpha \cdot \nabla\rho_\beta, \sigma = \sigma_{\alpha\alpha} + \sigma_{\beta\beta} + 2\sigma_{\alpha\beta}.$$

$$\chi_\alpha = \frac{\sqrt{\sigma_{\alpha\alpha}}}{\rho_\alpha^{4/3}}, \chi_\beta = \frac{\sqrt{\sigma_{\beta\beta}}}{\rho_\beta^{4/3}}. \quad (17.6)$$

$$v_\alpha = \nabla^2\rho_\alpha, v_\beta = \nabla^2\rho_\beta, v = v_\alpha + v_\beta. \quad (17.7)$$

另外可以引入动能密度, 用于对一组(Kohn-Sham)轨道产生密度:

$$\tau_\alpha = \sum_i^\alpha |\nabla\phi_i|^2, \tau_\beta = \sum_i^\beta |\nabla\phi_i|^2, \tau = \tau_\alpha + \tau_\beta. \quad (17.8)$$

所有可用泛函的一般形式为：

$$F[\rho_s, \rho_{\bar{s}}, \sigma_{ss}, \sigma_{\bar{s}\bar{s}}, \sigma_{s\bar{s}}, \tau_s, \tau_{\bar{s}}, u_s, u_{\bar{s}}] = \int d^3\mathbf{r} K(\rho_s, \rho_{\bar{s}}, \sigma_{ss}, \sigma_{\bar{s}\bar{s}}, \sigma_{s\bar{s}}, \tau_s, \tau_{\bar{s}}, u_s, u_{\bar{s}}) \quad (17.9)$$

其中 $\bar{s}$ 是 $s$ 的共轭自旋。

以下是MOLPRO支持的泛函关键词列表。此外，下一部分还有一组关键词别名，其中列出了主要泛函的各种组合。

B86:  $X\alpha\beta\gamma$

B86MGC: 加上改进梯度校正的 $X\alpha\beta\gamma$

B86R: 重新优化的 $X\alpha\beta\gamma$

B88: Becke 1988交换泛函

B88C: Becke 1988关联泛函

B95: Becke 1995关联泛函

B97DF: B97的密度泛函部分

B97RDF: Hamprecht等人重新参数化B97的密度泛函部分

BR: Becke-Roussel交换泛函

BRUEG: Becke-Roussel交换泛函—均匀电子气极限

BW: Becke-Wigner交换-关联泛函

CS1: Colle-Salvetti关联泛函

CS2: Colle-Salvetti关联泛函

DIRAC: Slater-Dirac交换能

ECERF: 短程LDA关联泛函

G96: Gill的1996梯度校正交换泛函

HCTH120: Handy最小二乘拟合泛函

HCTH147: Handy最小二乘拟合泛函

HCTH93: Handy最小二乘拟合泛函

LTA: 局域 $\tau$ 近似

LYP: Lee, Yang和Parr的关联泛函

MK00: 用于准确虚轨道能量的交换泛函

MK00B: 用于准确虚轨道能量的交换泛函

P86:

PBEC: PBE关联泛函

PBEX: PBE交换泛函

PBEXREV: 改进的PBE交换泛函

PW86:

PW91C: Perdew-Wang 1991 GGA关联泛函

PW91X: Perdew-Wang 1991 GGA交换泛函

PW92C: Perdew-Wang 1992 GGA关联泛函

STEST: 电子数测试

TH1: Tozer和Handy 1998

TH2:

TH3:

TH4:

THGFC:

THGFCF0:

THGFCO:

THGFL:

VSXC:

VWN3: Vosko-Wilk-Nusair (1980) III局域关联能

VWN5: Vosko-Wilk-Nusair (1980) V局域关联能

### 17.4.1 泛函的别名

还根据习惯的别名定义了附加的泛函关键词。以下是对照表。

别名	泛函	Ref
B	B88	[4]
B-LYP	B88 + LYP	
B-P	B88 + P86	
B-VWN	B88 + VWN5	
B3LYP	0.2d0 EXACT + 0.72d0 B88 + 0.08d0 DIRAC + 0.81d0 LYP + 0.19d0 VWN5	[4]
B3LYP3	0.2d0 EXACT + 0.72d0 B88 + 0.08d0 DIRAC + 0.81d0 LYP + 0.19d0 VWN3	
B3LYP5	0.2d0 EXACT + 0.72d0 B88 + 0.08d0 DIRAC + 0.81d0 LYP + 0.19d0 VWN5	
B88X	B88	
B97	0.1943d0 EXACT + B97DF	[4]
B97R	0.21d0 EXACT + B97RDF	
BECKE	B88	

---

BH-LYP	0.5d0 EXACT + 0.5d0 B88 + LYP	
CS	CS1	
D	DIRAC	
HFB	B88	[4]
HFS	DIRAC	
LDA	DIRAC + VWN5	
LSDAC	PW92C	[17]
LSDC	PW92C	[17]
LYP88	LYP	
PBE	PBEX + PBEC	[14]
PBE0	0.25d0 EXACT + 0.75d0 PBEX + PW91C	[1]
PBEREV	PBEXREV + PBEC	
PW91	PW91X + PW91C	[15]
S	DIRAC	
S-VWN	DIRAC + VWN5	
SLATER	DIRAC	
VS99	VSXC	
VWN	VWN5	
VWN80	VWN5	

---

### 17.4.2 执行新的泛函

自动代码生成程序 (ACG) [20] 用于在MOLPRO中执行新的密度泛函。这样的计算需要MAPLE数学程序和XSLT分析器，在CONFIG 中用变量XSLT定义。

输入文件的格式为XML文件，其中包含新泛函的所有信息。所有的密度泛函XML文件位于lib/df目录，下一次在MOLPRO主目录执行make命令时自动激活。

文件的格式由表达式构成，表达式必须由空行分开。表达式包括参量和值，语法为：

XML文档的根元素是content。在下一层有两个允许的元素，functional和alias。

functional元素有一个id标志，在MOLPRO中用作泛函的关键词。允许的元素在表17.1中定义。

ref和alias可以请求多次。最终的元素是maple，它也可以请求多次。典型的MAPLE 表达式如：

表 17.1: 定义泛函允许的元素

title	出现在泛函文档头中的文字
tex	泛函的文字说明
ref@id	包含在doc/references.xml中参考文献的别名
alias@id	泛函的关键词别名

A:=1.2:

写作

```
<maple lhs="A">1.2</maple>
```

要输入这样的MAPLE流程,

```
add_together:=proc(a,b) a+b end:
```

应当写作

```
<maple lhs="add_together" proc="a,b">a+b</maple>
```

Perdew-Wang 1991 GGA交换泛函的例子如下:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<content>
  <functional id="PW91X">
    <title>Perdew-Wang 1991 GGA Exchange Functional</title>
    <ref>PW91APPS</ref>
    <maple lhs="g">1/2*E(2*rho(s))</maple>
    <maple lhs="G">1/2*E(2*rho(s))</maple>
    <maple lhs="E" proc="n"> -3/(4*Pi)*(3*Pi^2)^(1/3)*n^(4/3)*F(S)</maple>
    <maple lhs="S">chi(s)/(2*(6*Pi^2)^(1/3))</maple>
    <maple lhs="F" proc="S">
      (1+0.19645*S*arcsinh(7.7956*S) + (0.2743-0.1508*exp(-100*S^2))*S^2)/
      (1+0.19645*S*arcsinh(7.7956*S)+0.004*S^4)
    </maple>
  </functional>
</content>
```

杂化泛函通过紧接在content层之后给定alias元素来定义。还定义了key, functional, 和ref元素。例如, B3LYP泛函定义为:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<content>
  <alias>
    <key id="B3LYP"/>
  </alias>
</content>
```

```

<key id="B3LYP5"/>
<functional id="EXACT" factor="0.2d0"/>
<functional id="B88" factor="0.72d0"/>
<functional id="DIRAC" factor="0.08d0"/>
<functional id="LYP" factor="0.81d0"/>
<functional id="VWN5" factor="0.19d0"/>
</alias>
</content>

```

## 17.5 经验阻尼色散校正

可以在Kohn-Sham计算中加入经验阻尼色散校正。在长程关联效应成为主导的时候，这会变得特别重要。色散校正使用范德瓦尔斯半径和来自文献[1,2]的原子色散因子，它的泛函形式为：

$$E_{\text{disp}} = -s_6 \sum_{i,j < i}^{N_{\text{atoms}}} f_{\text{damp}}(R_{ij}) \frac{C_6^{ij}}{R_{ij}^6} \quad (17.10)$$

在上面的方程中， $N_{\text{atoms}}$ 是原子总数， $R_{ij}$ 是原子*i*和*j*之间的距离， $s_6$ 是依赖于所选泛函的全局换算参数（见下）， $C_6^{ij}$ 的值由原子色散因子 $C_6^i$ 和 $C_6^j$ 按照以下方式进行计算：

$$C_6^{ij} = 2 \frac{C_6^i C_6^j}{C_6^i + C_6^j} \quad (\text{Ref.}[1]) \quad (17.11)$$

$$C_6^{ij} = \sqrt{C_6^i C_6^j} \quad (\text{Ref.}[2]) \quad (17.12)$$

函数 $f_{\text{damp}}$ 对短程原子间距的色散校正加上阻尼，定义为：

$$f_{\text{damp}}(R_{ij}) = \frac{1}{1 + \exp[-\alpha(R_{ij}/(R_{\text{vdW}}^i + R_{\text{vdW}}^j) - 1)]} \quad (17.13)$$

其中 $R_{\text{vdW}}^i$ 是原子*i*的范德瓦尔斯半径， $\alpha$ 是参数，通常设定为23（文献[1]）或20（文献[2]）。

目前，有以下泛函可以与经验色散校正一起使用：



表 17.2: 对密度泛函优化的换算参数 $s_6$  (见方程17.10)

泛函	文献[1]	文献[2]
BLYP	1.40	1.20
PBE	0.70	0.75
BP86	1.30	1.05
B3LYP	—	1.05

色散校正可以在MOLPRO输入中用`dispcorr`进行计算, 用以下模板可以添加到DFT能量上:

```
ks,<func>
eks=energy
dispcorr
eks_plus_disp=eks+edisp
```

即, 色散校正 (方程17.10) 存于变量`edisp`中。注意, 通过读取MOLPRO的内部变量`DFTNAME`, `dispcorr`可以知道以前使用的泛函类型。

`dispcorr`程序有以下选项:

- MODE**            调整使用的参数: **MODE=1**使用来自文献[1]的参数, **MODE=2**使用来自文献[2]的参数 (默认**MODE=1**)。
- SCALE**           全局换算参数 $s_6$  (关于优化值, 参见方程17.10和表17.2)。
- ALPHA**           阻尼函数参数 (见方程17.13)。较小的值会导致对中等间距有较大的校正。

参考文献:

1. S. Grimme, J. Comp. Chem. 25, 1463 (2004).
2. S. Grimme, J. Comp. Chem. 27, 1787 (2006).

## 17.6 例子

以下是使用非自洽和自洽DFT的例子。

```
! $Revision: 2006.3 $
geometry={c;n,c,r}
r=1.1 angstrom
```

```
df=[b,lyp]
rhf;method(1)=program
dft;edf(1)=dftfun
uhf;method(2)=program
dft;edf(2)=dftfun
uks;method(3)=program,edf(3)=dftfun
dft;method(4)=program,edf(4)=dftfun
table,dftname,dftfuns
table,method,edf
```

<http://www.molpro.net/info/current/examples/cndft.com>

局域化轨道依据Boys, Pipek-Mezey, 或者NLMO标准进行计算。局域化在每个对称类中分别进行。如果需要完全局域化, 就不要使用对称性。所有的子命令都可以缩写为三个字符。

局域化程序用LOCALI命令调用

```
LOCALI [,method],[LOCMETHOD=locmethod],[REFORB=record]
```

*method*关键词可以是BOYS, PIPEK, 或NATURAL。默认用Boys标准对最后一次能量计算的价轨道进行局域化。只有那些使能量不变的轨道子集才会进行变换。这些默认值可以用一些可选的命令修改, 说明如下。

LOCMETHOD选项仅用于Pipek-Mezey局域化。*locmethod*可以采用以下值:

- |                    |  |
|--------------------|--|
| <i>locmethod=1</i> | 标准的迭代局域化方法。  |
| <i>locmethod=2</i> | 使用二阶局域化方法。将会去掉冗余转动。  |
| <i>locmethod=3</i> | 首先用标准方法做第一次迭代, 接下来调用 <i>locmethod=2</i> 。对于存在冗余转动的情况, 如苯, 该选项是推荐的。 |

REFORB选项与REFORB指令具有相同的作用, 后面进一步说明。

## 18.1 定义输入轨道 (ORBITAL)

```
ORBITAL,record.file,specifications
```

进行局域化的轨道从转存记录*record.file*中读取。对某个态特定的轨道集可以用*specifications*选择, 解释参见4.11节。默认为最后计算的轨道。

## 18.2 保存局域化轨道 (SAVE)

SAVE, *record.file*

定义保存局域化轨道的转存记录。如果转存记录已存在，则往上添加局域化轨道。默认是输入的记录（参见ORBITAL）。

## 18.3 选择局域化方法 (METHOD)

METHOD, *method*

局域化方法 $method$ 可以是BOYS, PIPEK, 或NATURAL。这也可以作为LOCALI卡的参数指定（见上）。

## 18.4 轨道的离域化 (DELOCAL)

DELOCAL

如果出现这个卡，则对轨道离域化。

## 18.5 局域化AO (LOCAO)

LOCAO

如果出现这个卡，则对每个MO有贡献的AO数量减到最少。这对于旋转简并轨道（例如原子的px, py, pz）是很有用的，可以产生纯轨道（在这种情况下是px, py, pz）。

这意味着使用Pipek-Mezey局域化。

## 18.6 选择轨道空间

默认只对价轨道进行局域化，以保证随后电子关联处理的不变性。这一性质可以用OCC和CORE指令修改。

### 18.6.1 定义占据空间 (OCC)

OCC,  $o_1, o_2 \dots$

在每个对称性 $i$ 中，定义进行局域化的最高占据轨道 $o_i$ 。

### 18.6.2 定义芯轨道 (CORE)

CORE,  $c_1, c_2 \dots$

每个对称性中，前 $c_i$ 个轨道作为芯轨道处理，不进行局域化。因此在对称性 $i$ 中，只对轨道 $c_i + 1$ 到 $o_i$ 进行局域化。

### 18.6.3 定义轨道组 (GROUP, OFFDIAG)

GROUP, *orb1, orb2, orb3, ...*

这个卡定义进行局域化的轨道组，如下：

GROUP, 1.1, 2.1, 3.1	对称性1中的一组轨道1-3
GROUP, 1.1, -3.1	和前一个例子等价
GROUP, 3.1, 5.1, -8.1	这一组包含对称性1中的轨道3,5,6,7,8

不同组的轨道分别独立进行局域化。不包含在任何组中的轨道保持不变。

### 18.6.4 组之间的局域化 (OFFDIAG)

OFFDIAG

如果出现这个卡，则对组间而不是对组内进行局域化。

## 18.7 局域化轨道的排序

ORDER, *type*

如果 *type*=CHARGE，则轨道依照其电荷中心排序（默认）。

如果 *type*=FOCK，则轨道依照Fock算符(PIPEK)中递增的对角元，或者递增的Coulson加和轨道能量(BOYS)进行排序。这需要前一能量计算的Fock算符。对于Hartree-Fock轨道的局域化，这个算符存储于转存记录中，由程序自动寻找。对于MCSCF轨道的局域化，有效Fock算符从MCSCF密度矩阵计算（参见DENSITY选项）。也可以在FOCK卡中指定前一个SCF计算的转存记录，接下来从这个记录读取Fock算符。对于简并轨道，还会尝试按照电荷中心的坐标进一步排序（首先按照最大的Z-坐标，接下来按照先x后y的顺序）。

这个卡不能用于NLMO局域化。

### 18.7.1 不重新排序 (NOORDER)

NOORDER

如果出现这个卡，则局域化轨道不再重新排序。如果局域化轨道用作初始猜测，这是很有用的，意味着轨道顺序保持不变。

### 18.7.2 使用域进行排列 (SORT)

`SORT,[THRCHCHG=charge][THREIG=eps],GROUP=igrp],[REVERT],centrelist`

这个指令仅对Pipek-Mezey局域化有效。轨道按照域和给定的中心列表进行排序。中心对域的贡献通过Löwdin电荷确定。只有电荷大于THRCHCHG（默认0.4）的中心才会包含在这些域中。轨道根据以下标准重新排序：

- 1) 域的首要中心具有最大的电荷，次级中心具有次大的电荷。轨道在每个局域化的组中分别排序。首先对所有的轨道排序，以使首要中心符合*centrelist*给定的顺序。具有首要中心但未在*centrelist*给出的轨道，将放在最后。
- 2) 在给定首要中心的每组轨道内，最先包含只具有单中心（孤对）的轨道。对其余的轨道进行排序，使得次级原子符合*centrelist*的顺序。具有次级中心但未在*centrelist*给出的轨道，将放在最后。
- 3) 如果给出了REVERT，则在每个局域化组中颠倒顺序。
- 4) 如果给出了GROUP，只有给定组中的轨道才会重新排序。*igrp*对闭壳层和非活性轨道是2，对单参考方法的开壳层是1，对CASSCF计算的活性轨道是3。
- 5) 如果给出了THREIG，只有能量大于给定值的轨道才会重新排序。*eps* 必须是负值。其余的轨道放在最后（如果给出了REVERT则是最先）。

注意，芯轨道既不做局域化，也不重新排序。

### 18.7.3 定义参考轨道 (REFORB)

`REFORB,record.file,specifications`

重新排序局域化轨道，使之与从*record.file*读入的参考轨道重叠最大。这对不同几何结构的局域关联处理是很有用的，可以使局域化常数的顺序保持不变。对某个态特定的轨道集，可以用*specifications*选择，解释参见4.11节。

### 18.7.4 选择Fock矩阵 (FOCK)

`FOCK,record.file`

指定包含Fock算符的记录，用于轨道排序。注意，只有SCF转存记录包含Fock算符。默认是来自产生轨道输入的能量计算的Fock算符。

### 18.7.5 选择密度矩阵 (DENSITY)

DENSITY, *record.file, specifications*

指定包含密度矩阵的记录，用于构建Fock算符进行轨道排序。这可用于没有Fock算符的时候，并且只对MCSCF局域化起作用。默认使用（态平均）MCSCF密度。对某个态特定的密度矩阵可以用*specifications*选择，说明参见4.11节。

## 18.8 局域化阈值 (THRESH)

THRESH, *thresh, eorder*

*thresh*是局域化的阈值（默认1.d-12）。如果*eorder*非零（默认1.d-4），那么能量差小于*eorder*的轨道被认为是简并，并按照其电荷中心的位置重新排序（参见18.7节）。

## 18.9 PM局域化选项 (PIPEK)

对Pipek-Mezey局域化有一些特殊选项（都是可选的）：

PIPEK, METHOD=*method*, DELETE=*ndel*, MAXDL=*maxdl*, THRESH=*thresh*, ORDER=*iorder*,  
STEP=*step*

METHOD	<i>method</i> =1: 使用2x2转动方法（默认）； <i>method</i> =2: 使用Newton-Raphson方法； <i>method</i> =3: 最初的迭代使用使用2x2转动方法，最后一次收敛用NR方法。
DELETE	在PM局域化中，对每个原子的每个角动量删除最后 <i>ndel</i> 个基函数。对于弥散的（扩充的）基组，这可能会有用，可以得到合适的局域化。
MAXDL	如果 <i>ndel</i> >0，只删除到角动量 <i>maxdl</i> 的函数。
ORDER	如果 <i>iorder</i> =1，根据Fork矩阵对角元增加的顺序排列最终的轨道； 如果 <i>iorder</i> =2，根据电荷中心排列最终的轨道（默认）。
THRESH	局域化阈值（同THRESH指令）。
STEP	NR方法的最大步长（默认0.1d0）。

## 18.10 打印选项 (PRINT)

PRINT, [ORBITAL=*pri*] [, CHARGE] [, CENTRES] [, TEST] [, TRAN];

如果给定了ORB[ITAL]，则打印局域化轨道。

如果给定了CHA[RGE]或CEN[TRES]，则打印局域化轨道的电荷中心。

如果给定了TRAN，则打印转换矩阵（仅Boys）。

如果给定了TEST，则打印中间信息。



## 第十九章

## MCSCF程序MULTI

*MULTI*是一般MCSCF/CASSCF程序，编写者是  
P. J. Knowles和H.-J. Werner (1984)

参考文献：

- H.-J. Werner and P. J. Knowles, J. Chem. Phys. 82, 5053 (1985).  
P. J. Knowles and H.-J. Werner, Chem. Phys. Lett. 115, 259 (1985).  
使用本程序的所有出版结果必须引用以上文献。此外参见：  
H.-J. Werner and W. Meyer, J. Chem. Phys. 73, 2342 (1980).  
H.-J. Werner and W. Meyer, J. Chem. Phys. 74, 5794 (1981).  
H.-J. Werner, Adv. Chem. Phys. LXIX, 1 (1987).

本程序可以执行CASSCF以及一般MCSCF计算。对于CASSCF计算，可以使用Slater行列式或者作为N电子基的CSF。在大多数情况下，使用Slater行列式更有效率。一般MCSCF计算必须使用CSF作为基。

程序使用了相当复杂的优化方法。这个算法对轨道和CI因子的变化是二阶的，因此是二阶收敛的。由于在与轨道无关的参数中包含了重要的高阶项，因此经常能得到三阶收敛。对于简单的情形，一般两三轮迭代就能收敛。但是在某些应用中仍有可能发生收敛问题，通常意味着活性空间选择不合适。例如，如果两个弱占据轨道对能量同等重要，但只有一个包含在了活性轨道集中，程序可能会在两者之间交替变化。对这种情况，减少或增加活性轨道空间就能解决这个问题。对于其它收敛困难的情况，如果有相同对称性的两个电子态近简并或完全简并，也会产生收敛问题，因为程序会在两个态之间交替变化。这可能发生在接近避免交叉或者接近离解极限的情况中。这种问题可以通过对特定的几个态优化平均能量来避免。通过选择组态的子集作为主空

间（即P空间，PSPACE），还可以强制收敛到特定的态。在这个空间中，构造哈密顿量并直接对角化；其余组态的因子用P空间波函作为零级近似迭代优化。对于线型分子，还可以使用LQUANT选项，它可以强制收敛到给定 $\Lambda$ 量子数的电子态，即， $\Sigma$ ， $\Pi$ ， $\Delta$ 等态。

## 19.1 输入的结构

所有MULTI已知的子命令可以缩写为四个字符。输入命令分为几个逻辑类；在每一类中，命令可以按任何顺序出现，但是各类命令必须按照正确的顺序。

1. 用命令MULTI或MCSCF调用程序。
2. 定义轨道空间分区的卡：OCC,FROZEN,CLOSED
3. 一般选项（大多数未特别提到的命令都在这里）
4. WF卡，定义态的对称性
5. 与态对称性有关的选项：WEIGHT,STATE,LQUANT
6. 态对称性的组态定义：SELECT,CON,RESTRICT
7. 定义态对称性的主要组态：PSPACE
8. 更多的一般选项。

其中，4)到8)部分可以重复数次；这个方法指定对于不同对称性的几个态的能量做平均。

在MULTI命令行中可以指定多个选项：

MULTI,*options*

可选的选项有：

MAXIT	最大迭代数（默认10）
ENERGY	能量收敛阈值
GRADIENT	梯度收敛阈值
STEP	步长收敛阈值
FAILSAFE	（逻辑型）使用更稳定的收敛选项

还有许多其它的选项和阈值也可以在命令行给出，说明见第19.8.5节。

## 19.2 定义轨道子空间

### 19.2.1 占据轨道

OCC, $n_1, n_2, \dots, n_8$ ;

$n_i$ 指定在不可约表示编号*i*中的占据轨道数（包括FROZEN和CLOSED）。如果OCC卡没有给定，默认的信息来自最近的MCSCF计算，如果没有，使用对应最小的价轨道集的轨道，即，使用全部价轨道空间。

### 19.2.2 冻芯轨道

FROZEN, $n_1, n_2, \dots, record.file$ ;

$n_i$ 指定在不可约表示编号*i*中的冻芯轨道数。这些轨道在全部组态中都是双占据的，并且不会被优化。注意在早期版本的MOLPRO中，这个指令为CORE，现在被重新命名，以避免与MRCI和CCSD程序的CORE轨道混淆。

*record.file*是冻芯轨道的记录名称；如果没有提供，来自START卡的*orb*。*record.file*可以在最后一个非零 $n_i$ 之后的任何位置指定。如果轨道初始猜测来自邻近的几何结构，它必须给出，接下来要定义在当前结构计算的SCF轨道。如果使用当前波函数进行随后的梯度计算，那么*record.file*是强制的，必须指定当前结构的闭壳层SCF轨道。注意，*record*必须大于2000。

如果没有给出FROZEN卡，芯轨道数来自最近一次MCSCF计算，如果没有则不做轨道冻结。如果FROZEN卡给定为FROZEN,*record.file*，那么使用对应原子内壳层的轨道，即Li-Ne是1s，Na-Ar是1s2s2p，等。没有任何说明的FROZEN卡把冻芯轨道数重新设置为0。

### 19.2.3 闭壳层轨道

CLOSED, $n_1, n_2, \dots, n_8$

$n_i$ 指定在不可约表示编号*i*中的闭壳层轨道数，包含所有的FROZEN轨道。这些轨道不构成任何活性空间，即：它们在所有的CSF中都是双占据的；但是与芯轨道相反（参见FROZEN），这些轨道是完全优化的。

如果忽略CLOSED卡，那么默认数据来自最近的MCSCF计算，如果没有则使用原子内壳层，说明参见上面的FROZEN。

### 19.2.4 冻结轨道

FREEZE,*orb.sym*;

指定的轨道将不被优化，仍保持和初始猜测相同。*orb.sym*应当是一个活性或闭壳层轨道。如果*orb.sym*是冻芯轨道，这个卡不起作用。

### 19.3 定义被优化的态

每个需要优化的态的对称性用一个WF卡定义，之后可以接STATE, WEIGHT, RESTRICT, SELECT, CON, 和/或PSPACE卡。属于由WF卡定义的特定态对称性的全部卡必须连在一起，接在WF卡之后。但是各个卡之间的顺序是任意的。

#### 19.3.1 定义态的对称性

在WF卡中指定电子数和波函的总对称性：

WF,elec,sym,spin

其中：

elec            是电子数

sym            是不可约表示的序号

spin            定义自旋对称性， $spin=2S$ （单重态=0，双重态=1，三重态=2，等）

注意，如果未定义某个值或者全部值，使用自动判断的默认值（参见4.8节）。

输入指令STATE, WEIGHT, LQUANT, SELECT, PUNCSF总是指前一个WF卡定义的态对称性。如果在WF卡之前给出这些指令，那么当前态的对称性假定来自前一个计算，或者来自变量[MC]SYMMETRY(1)和[MC]SPIN(1)（如果定义了的话）。如果给了这些卡中的任何一个，或给了WF卡，则不使用变量STATE, WEIGHT, LQUANT, 和SELECT，并且态对称性的数量默认是1，而不考虑变量[MC]SYMMETRY中定义了多少对称性。

#### 19.3.2 定义当前对称性中态的数量

STATE,nstate;

nstate是当前对称性态的数量。默认所有的态都用权重1来优化（参见WEIGHT卡）。

#### 19.3.3 定义态平均计算中的权重

WEIGHT,w(1),w(2),...,w(nstate);

$w(i)$ 是当前对称性第*i*个态的权重。所有的权重默认都是1.0。参见STATE卡。如果对特定的态对称性只想优化第二个态，可以指定

STATE,2;WEIGHT,0,1;

但是注意，这可能会导致根的跳跃问题。

### 19.3.4 动态权重

动态权重的描述见J. Chem. Phys. 120, 7281 (2004)。可以用  
DYNW, *dynfac*  
激活。接下来用下式计算每个态的权重

$$w = 1 / \cosh (\text{dynfac} * \Delta E)^2 \quad (19.1)$$

其中 $\Delta E$ 是所考虑的态与基态之间的能量差（单位Hartree）。权重在优化过程中自动校正。

## 19.4 定义组态空间

程序默认在活性空间中产生完全组态集（CAS）。通过使用RESTRICT 选项，可以对整个空间限制为特定的占据方式。组态也可以用SELECT 选项，从前一个计算的波函选择，或者在CON卡直接指定。注意这个程序只允许选择或指定轨道组态。对于每个轨道组态，总是包含所有的自旋耦合。可能用到的RESTRICT，SELECT和CON卡必须直接接在定义相应态对称性的WF卡之后。

### 19.4.1 占据限制

RESTRICT, *nmin*, *nmax*, *orb*<sub>1</sub>, *orb*<sub>2</sub>, ... *orb*<sub>*n*</sub>;

这个卡可用于限制占据方式。只有在指定轨道 $orb_1, orb_2, \dots, orb_n$ 中包含电子数在 $nmin$ 和 $nmax$ 之间的组态，才能包含在波函中。如果 $nmin$ 和 $nmax$ 为负，那么指定轨道中有 $abs(nmin)$ 和 $abs(nmax)$ 个电子的组态将被删除。例如，这可以用于忽略单激发组态。轨道按照 $number.sym$ 的形式定义，其中 $number$ 是不可约表示 $sym$ 中的轨道序号。可以连续用几个RESTRICT卡。只有在第一个WF卡之前定义了CONFIG卡，RESTRICT才起作用。

在第一个WF卡之前给出的RESTRICT卡是全局的，也就是说，用于所有的态对称性。如果给出了一个全局的RESTRICT卡，则不使用[MC]RESTRICT变量。

特定态的RESTRICT卡也可以在WF卡之后给出。它们在全局轨道限制之上，进一步加以限制。

如果特定态的和全局的RESTRICT卡均未发现，则使用变量[MC]RESTRICT 的值。

### 19.4.2 选择组态

SELECT, *ref1*, *ref2*, *refthr*, *refstat*, *mxshrf*;

这个卡用来定义组态集而不是使用默认CAS。这一选项自动触发CONFIG选项，用于选择CSF而不是行列式。组态可以用卡CON定义，它必须立即出现在SELECT卡之后。如果 $ref1$ 是已经存在的MOLPRO记录名称，组态也可以从这个记录读入，并按照给定的阈值进行选择。

$ref1=rec1.file$	( $rec1 > 2000$ )从指定的卡读入组态。如果 $ref1$ 没有指定，程序假定组态从后面的CON卡读入（参见CON）。
$ref2=rec2.file$	( $rec2 > 2000$ )从指定的卡读入附加组态。如果 $rec2$ 为负，则读取 $rec1$ 和 $abs(rec2)$ 之间的所有的记录。这种方法发现的所有组态将合并在一起。
$refthr$	从磁盘（记录 $rec1-rec2$ ）读取组态的筛选阈值。应用于每个轨道组态所有CSF的模。
$refstat$	指定从哪个态矢量选择组态。只用于态平均计算保存组态的情况。如果没有指定 $refstat$ ，组态从所有的态中选择。
$mxshrf$	在选择的或生成的组态中，开壳层的最大数量。

### 19.4.3 指定轨道组态

CON, $n_1, n_2, n_3, n_4, \dots$

指定包含在当前对称性中的轨道组态。第一个CON卡之前必须有SELECT卡。 $n_1, n_2$ 等是活性轨道的占据数（0，1，或2）。例如对于

OCC,5,2,2;CLOSED,2,1,1;

$n_1$ 是轨道3.1 (number.sym)的占据数， $n_2$ 是轨道4.1的占据数， $n_3, n_4, n_5$ 是轨道5.1, 2.2, 2.3的占据数。可以连续用任意数量的CON卡。

BH分子的例子：

```
OCC,4,1,1;      ! 四个sigma, 一个pi轨道被占据
FROZEN,1;       ! 第一个sigma是双占据的, 且冻结
WF,6,1;         ! 6个电子, 单重Sigma+态
SELECT          ! 触发组态输入
CON,2,2         ! 2sigma**2, 3sigma**2
CON,2,1,1       ! 2sigma**2, 3sigma, 4sigma
CON,2,0,2       ! 2sigma**2, 4sigma**2
CON,2,0,0,2     ! 2sigma**2, 1pi_x**2
CON,2,0,0,0,2   ! 2sigma**2, 1pi_y**2
```

### 19.4.4 选择主要组态集

PSPACE,thresh

在主要组态空间中构造哈密顿量并直接对角化，这可以用PSPACE卡选择。残余组态（Q-空间）的因子用P-空间波函在零级近似下迭代优化。

如果thresh非零则是阈值，用于自动选择能量小于 $e_{min} + thresh$ 的所有组态作为P-空间组态，其中 $e_{min}$ 是所有组态的最低能量。P-空间组态可以进一步用CON卡定义，它必须直接接在PSPACE卡之后。它们与用阈值选择的组态合并到一起。通过指定非常小的阈值可以避免自动选择。程序已经设置了非常好的默认阈值（0.4），因此输入中通常不需要pspace卡。此外，如果MCSCF中的组态数小于20，那么所有的组态都进入P-空间，除非输入中已经设置了PSPACE卡。

第一个WF卡（或者STATE, WEIGHT, SELECT, PUNCSF，如果WF没有给出的话）之前的PSPACE卡定义的P-空间阈值是全局的，也就是说，对所有的态对称性都有效。特定态的阈值说明可以由位于相应WF卡之后的PSPACE卡来定义。对于后一种情况，PSPACE卡之后可以接CON卡，用来定义特定态的P-空间组态。

### 19.4.5 对于线型分子，投影到特定的 $\Lambda$ 态

由于MOLPRO只能使用阿贝尔点群（例如，线型分子用 $C_{2v}$ 而不是 $C_{\infty v}$ ），例如 $\Delta_{x^2-y^2}$ 和 $\Sigma^+$ 都出现在不可约表示1中。有时不可能预测程序将收敛于哪个态。在这种情况下，可以使用LQUANT选项定义所需要的态。

LQUANT,lam(1),lam(2),...,lam(nstate);

其中的 $lam(i)$ 是第 $i$ 个态的 $\Lambda$ 量子数，例如， $\Sigma$ 态是0， $\Pi$ 态是1， $\Delta$ 态是2，等等。将对 $\Lambda^2$ 构造矩阵，并在P空间组态基中进行对角化。使用本征矢把P空间哈密顿量转换到对称匹配基中，接下来程序选择具有正确对称性的本征矢。态将按照LQUANT卡指定的对称性排序；在每个对称性中，态按照能量增加的顺序排序。

## 19.5 恢复、保存轨道和CI矢量

MULTI通常需要初始轨道猜测。这一部分我们将说明如何定义这些轨道，以及如何保存优化的轨道。在CASSCF的计算中，可以选择把最终的轨道转换为自然轨道（一阶密度矩阵是对角化的），赝正则轨道（有效Fock算符是对角化的），或者局域化轨道。

### 19.5.1 定义初始猜测

START,record,[options];

*record*: 包含初始轨道的转存记录。通常*record*具有*irec.ifil*的形式，其中*irec* 是记录的序号（例如，2140），*ifil*是文件序号（通常是2）。*options*可用于选择特定类型的轨道；细节参见4.11节。

如果没有指定这个卡，程序将按照以下方式寻找合适的初始轨道：

- 首先：      试着从ORBITAL卡指定的记录（或者相应的默认值，参见ORBITAL）读取轨道。搜索所有的文件。
- 其次：      试着从最近的MCSCF计算寻找轨道。搜索所有的文件。
- 第三：      试着从最近的SCF计算寻找轨道。搜索所有的文件。

如果没有发现轨道，则产生初始轨道猜测。

使用临近结构的MCSCF轨道作为初始猜测经常会很有用（如果发现了这样的轨道，将是自动的，参见上面的默认设置）。但是注意，冻芯轨道应当总是来自当前几何结构的SCF或MCSCF计算，而且必须在FROZEN卡单独指定。否则程序可能会停止，并出现错误信息：“non-orthogonal core orbitals”。如果在前一个MCSCF计算的FROZEN卡中指定了芯轨道，程序可以记住芯轨道来自哪里。

### 19.5.2 转动初始轨道对

ROTATE,*orb1.sym,orb2.sym,angle*

将*sym*对称性中的初始轨道*orb1*和*orb2*做*angle*度的 $2 \times 2$ 转动。*angle*=0表示交换轨道。ROTATE只在START卡之后才有意义。其它可能的轨道操作，参见MERGE。

### 19.5.3 保存最终的轨道

ORBITAL,*record.file*

把轨道转存到记录*record.file*中。默认的*record*是2140，*file*=2。在同一个任务中，默认的记录序号对随后的每个MCSCF计算都加一（见4.11节）。因此，如果在一个任务中，对几个几何结构做几种不同的MCSCF计算，那么每一个MCSCF能正常地从合适的轨道开始，即使没有出现ORBITAL或START卡也是如此。

如果定义了NATORB，CANORB或LOCORB卡，则可以忽略ORBITAL卡，因为*orb*也可以在这些卡中定义（在这些情况下，*orb*的默认值与上面应用的例子相同）。

### 19.5.4 对梯度计算保存CI矢量和信息

旧的形式（已作废）：

SAVE,*cidump,refsav,grdsav*;



新的形式:

SAVE,[CI=*cidump*,] [REF=*refsav*,] [GRD=*grdsav*];

这个指令必须位于任何WF或STATE卡之前。选项可以是任何顺序。

- cidump*: 用于保存CI矢量的记录名称。默认的矢量仅写入草稿文件中。如果出现了NATORB, CANORB 或LOCORB卡, 应当在这些卡中定义*cidump*。由于目前很少使用保存的CI矢量, 因此这一选项很少使用。
- refsav*: 保存轨道组态及其权重的记录名称, 用于后面使用SELECT指令的MULTI或CI计算。在态平均计算中, 如果优化一个以上的态对称性的波函, 那么每个态对称性的权重分别保存到各自的记录*refsav*+(*istsym*-1)\*100 中, 其中*istsym*是WF卡在输入文件中的序号。如果出现了多个NATORB, CANORB或LOCORB卡, 那么记录的序号对随后的每个轨道集增加1000。注意, 这个选项意味着使用CSF, 即使没有定义CONFIG卡(参见19.6.1节)也是如此。
- grdsav*: 记录名称, 保存后面梯度计算所需的信息。如果输入包含FORCE或OPTG卡, 它会自动保存到记录5000.1中, 因此通常情况下不需要GRD选项。

### 19.5.5 自然轨道

NATORB,[*record*,] [*options*]

要求计算最终的自然轨道, 并写到记录*record*中。默认的*record*是2140.2, 或者在ORBITAL卡指定(如果出现的话)。默认不打印轨道, 哈密顿量不对新轨道做对角化。可以指定以下的*options* (以任意顺序):

- CI 以计算的自然轨道为基, 把哈密顿量对角化, 并打印组态及其有关的因子。这与GPRINT,CIVECTOR 指令具有相同的作用(参见6.12节)。默认只打印因子大于0.05的组态。阈值可以用THRESH (参见19.8.2节)或GTHRESH(参见6.11节)选项修改。
- STATE=*state* 对指定的态计算自然轨道。*state*的形式为*istate.isym*, 例如3.2是对称性2的第三个态。与早期版本相反, *isym*表示不可约表示的序号, 而不是态对称性的序号。因此它不依赖于WF 卡中给定的顺序。指定的态必

须已经是优化的。如果STATE没有给定，并且对两个或更多的态做了平均，那么使用态平均密度矩阵计算自然轨道（默认）。

**SPIN=*ms2*** 对给定自旋的态计算自然轨道。*ms2*等于 $2 * S$ ，也就是说，0表示单重态，1表示双重态，等等。这可以和STATE一起使用，用于对不同自旋的态平均的情况选择特定态。如果没有指定STATE，则使用给定自旋的所有态的态平均密度。

**SAVE=*record*** 把CI矢量保存到指定的记录中。

**ORBITAL=*record***

把轨道保存到指定的记录中（把*record*指定为第一个变量具有相同的作用，见上）。

**PRINT=*nvirt*** 打印每个对称性的*nvirt*个虚轨道。默认不打印轨道，除非定义了ORBPRINT选项（参见19.8.1节），或者此前给定了全局的GPRINT, ORBITALS指令（参见6.12节）。此卡的PRINT选项仅用于当前的轨道。

可以连续使用几个NATORB, CANORB和LOCORB卡（对不同的态）。与早期版本MOLPRO不同，不同的轨道集可以存储在一个转存记录中（但仍可以用几个不同的记录）。关于转存记录的信息以及如何指定后面计算需要的轨道集，参见4.11节。

### 19.5.6 赝正则轨道

CANORB,[*record*,] [*options*]

或

CANONICAL,[*record*,] [*options*]

要求正则化最终的轨道，并把它们写到记录*record*中。所有的选项与在NATORB中的作用相同。

### 19.5.7 局域化轨道

LOCORB,[*record*,] [*options*]

或

LOCAL,[*record*,] [*options*]

要求局域化最终的轨道，并把它们写到记录*record*中。所有的选项与在NATORB中的作用相同。

注意：LOCAL用MULTI解释，而LOCALI是另外一个命令，它调用局域化程序，并不能被MULTI程序识别。为了避免混淆，推荐使用LOCORB而不是LOCAL作为MULTI的子命令。

### 19.5.8 透热轨道

为了构造透热态，需要确定在绝热波函中透热态的混合。原则上，可以通过非绝热耦合矩阵元的积分来获得混合。使用近似方法通常会更容易，其中的混合可以通过检查MCSCF或CI波函的CI因子来确定。但是这种方法仅能用于轨道混合可忽略的情况。对于CASSCF波函，这可以通过当前结构与参考结构活性轨道重叠的最大化来实现，这里假定波函是透热的（例如，由于对称性的原因）。使用MCSCF程序中新的命令DIAB进行轨道重叠的最大化。只对活性轨道进行变换。

这个流程的工作方式如下：首先确定参考结构的轨道。接下来在移动后的结构上进行计算，获得与参考结构活性轨道有最大重叠的“透热的”活性轨道，这可以通过在输入中添加DIAB指令来实现：

旧的形式（Molpro96，已作废）：

DIAB,orbref, orbsav, orb1,orb2,pri

新的形式：

DIAB,orbref[,TYPE=orbtype][,STATE=state][,SPIN=spin][,MS2=ms2][,SAVE=orbsav]  
[,ORB1=orb1, ORB2=orb2][,PRINT=pri][,METHOD=method]

这里的orbref是保存有参考结构轨道的记录，orbsav是存储新轨道的记录。如果没有给出orbsav（推荐！），那么新的轨道存储在默认的转存记录（2140.2）或ORBITAL指令定义的记录中（参见19.5.3节）。与早期版本的MOLPRO相反，orbref和orbsav可以相同。指定TYPE，STATE，和SPIN可用于选择参考轨道的特定轨道集，详细说明见4.11节。orb1和orb2是做重叠最大化的一对轨道。这些轨道定义的形式为number.sym，例如3.1表示对称性1的第三个轨道。如果orb1和orb2没有给出，那么对所有活性轨道的重叠做最大化。pri是打印参数，如果设置为1，那么对每次Jacobi迭代打印每个轨道的转换角。method指定透热化的方法。method=1（默认）：使用Jacobi转动；method=2：使用分块对角化。两种方法产生非常接近的结果。method=2只能用于CASSCF波函。method=-1和method=-2：与正值相同，但是使用当前结构的AO重叠矩阵。这将使MO因子的变化最小，而不是对与临近轨道的重叠做最大化。

通过使用以上描述的默认值，以下的输入对大多数情况已经足够用了：

DIAB,orbref

使用Molpro98不需要给出GEOM和DISPL卡。会自动计算位移和重叠矩阵（结构和相应的轨道存储于转存记录中）。

透热轨道的性质是：在非绝热耦合矩阵元中，轨道贡献与重叠贡献的和近似是零，因此只能通过CI因子的改变来产生绝热混合。这允许用简单的方式（说明参见J. Chem. Phys. **89**, 3139 (1988)）或者更高级的方式（说明参见Pacher, Cederbaum, and

Köppel in J. Chem. Phys. **89**, 7367 (1988)), 直接从CI因子确定混合的角度。最近, MOLPRO又增加了自动流程 (参见J. Chem. Phys. **102**, 0000, (1999))。这在99.1版以后可以使用, 说明见第三十七章。

下面我们介绍H<sub>2</sub>S的例子, 它的两个最低激发态在C<sub>2v</sub>群分别是B<sub>1</sub>和A<sub>2</sub>对称性, 在C<sub>s</sub>群都是A''对称性。我们首先在C<sub>2v</sub>对称性执行参考计算, 接下来在C<sub>s</sub>对称性的位移结构确定透热轨道。随后的每个计算使用前面的轨道作为参考。也可以在所有其它计算中使用C<sub>2v</sub>的轨道作为参考。对这种情况必须去掉设置reforb=2141.2的倒数第二个输入卡。

```
! $Revision: 2009.1 $
***,H2S diabatic A" states

basis=VDZ                                ! 使用cc-pVDZ基组
symmetry,x,planeyz                       ! 使用Cs对称性并固定分子方位
orient,noorient                          ! 禁止自动重定位
geometry={s;h1,s,r1;h2,s,r2,h1,theta}   ! Z矩阵结构输入

gprint,orbitals,civector                 ! 全局打印选项

text,reference calculation for C2V
theta=92.12,r1=2.3,r2=2.3                ! 参考结构

{hf;occ,7,2;wf,18,1}                     ! 基态的scf计算

{multi;occ,9,2;closed,4,1;               ! 定义活性和非活性空间
wf,18,2;state,2;                          ! 两个A"态 (C2v中的1B1和1A2)
orbital,2140.2}                           ! 把轨道保存到2140.2
reforb=2140.2

text,calculations at displaced geometries

rd=[2.4,2.5,2.6]                          ! 定义键长范围

do i=1,#rd                                ! 对位移结构作循环

r2=rd(i)                                  ! 把r2设定为当前结构

{multi;occ,9,2;closed,4,1;               ! 波函定义与在参考结构相同
wf,18,2;state,2;                          ! 把新的轨道保存到记录
orbital,2141.2                             ! 用存到记录reforb中的参考轨道计算透热轨道
diab,reforb}

reforb=2141.2                             ! 把变量reforb设定为新轨道
```

```
enddo
```

[http://www.molpro.net/info/current/examples/h2s\\_diab.com](http://www.molpro.net/info/current/examples/h2s_diab.com)

关于自动生成透热能量，参见第三十七章。

## 19.6 选择优化方法

MULTI默认使用Werner, Meyer和Knowles开发的非线性优化方法。也可以使用其它的方法，例如Newton-Raphson流程或者扩充的Hessian流程，这可以用ITERATIONS指令进行选择（对于态平均计算，只能使用非线性优化方法）。对于CASSCF计算，CI问题用一组Slater行列式求解，除非给定了CONFIG卡。有些流程可以用DONT指令禁用。

### 19.6.1 选择CI方法

CONFIG,*key*;

*key*可以是DET或CSF，默认是CSF。如果没有给定CONFIG 或SELECT卡，默认使用行列式（即CASSCF）。

### 19.6.2 选择轨道优化方法

ITERATIONS指令可以用于修改默认的优化方法。它由依次的几个卡构成，包含在一对花括号里。

```
{ITERATIONS;
DO,method1,iter1[,TO,iter2];
DONT,method2,iter3[,TO,iter4];
...
}
```

其中*method*可以是以下之一：

DIAGCI	从指定的迭代开始，对哈密顿量做对角化。默认是迭代1。
INTERNAL	从指定的迭代开始，优化内部轨道。默认是第二次及之后的迭代。
WERNER	对指定的迭代，使用Werner-Meyer-Knowles非线性优化方法。这对所有的迭代是默认的。
AUGMENT	对指定的迭代，使用步长限制的扩充Hessian方法。
NEWTON	对指定的迭代，使用Newton-Raphson方法。
UNCOUPLE	在指定的迭代中，不同时优化轨道和CI因子。该选项将对这些迭代设定DIAGCI。

NULL 无轨道优化。

### 19.6.3 禁用优化

除了以上的ITERATIONS指令，有些流程还可以用DONT指令简单地禁止。

DONT, *code*

其中的*code*可以是：

ORBITAL	进行最初的CI，但不优化轨道。
WAVEFUNC	不优化轨道和CI因子，也就是说，如果提供了轨道和CI因子（参见START卡），那么仅做波函分析。
WVFN	WAVEFUNC的别名。
ANAL	不做波函分析。

### 19.6.4 禁止额外对称性机制

NOEXTRA

这个卡禁止寻找额外的对称性。默认如果出现了额外的对称性，每个轨道都分配给额外的对称性，并且不执行不同额外对称性轨道之间的转动。

## 19.7 计算期待值

程序默认计算偶极期待值和跃迁矩。更多的期待值或跃迁特性可以用TRAN，TRAN2和EXPEC，EXPEC2指令计算。

### 19.7.1 单电子算符的矩阵元

EXPEC, *oper*<sub>1</sub>, *oper*<sub>2</sub>, ..., *oper*<sub>*n*</sub>

TRAN, *oper*<sub>1</sub>, *oper*<sub>2</sub>, ..., *oper*<sub>*n*</sub>

对给定的单电子算符计算期待值和跃迁矩阵元。使用EXPEC只计算期待值。*oper*<sub>*i*</sub>是算符的代码。可以使用的算符及其相应的关键词在6.13节给出。

### 19.7.2 双电子算符的矩阵元

EXPEC2, *oper*<sub>1</sub>, *oper*<sub>2</sub>, ..., *oper*<sub>*n*</sub>

TRAN2, *oper*<sub>1</sub>, *oper*<sub>2</sub>, ..., *oper*<sub>*n*</sub>

对双电子算符计算跃迁矩阵元。目前只能用于角动量算符。用EXPEC2只能计算对角矩阵元。例如

TRAN2,LXX            计算 $L_x^2$ 的矩阵元。  
 TRAN2,LYY            计算 $L_y^2$ 的矩阵元。  
 TRAN2,LXZ            计算 $\frac{1}{2}(L_x L_z + L_z L_x)$ 的矩阵元。  
 TRAN2,LXX,LYY,LZZ 计算 $L_x^2$ ,  $L_y^2$ 和 $L_z^2$ 的矩阵元。同时也打印总和 $L^2$ 的矩阵元。

### 19.7.3 保存密度矩阵

DM,[*spindens*]

如果给出了DM指令,那么AO基中的一阶密度矩阵会写到ORBITAL卡指定的转存记录中(默认是2140.2)。如果没有给定ORBITAL卡,但是在NATORB, CANORB或LOCORB卡指定了记录,那么密度保存到输入中出现的第一个记录中。在态平均计算中,态平均密度以及各个态的密度都被保存。关于后面的程序如何重新获得并使用这些密度的信息,参见4.11节。

如果*spindens*是大于零的数,还保存自旋密度矩阵。注意,在一个转存记录中,最多可以保存50个密度矩阵。

如果没有给出DM指令,那么在单个态的计算中保存一阶密度矩阵,而在态平均计算中仅保存态平均密度。

## 19.8 各类选项

这一节讲述的所有命令都是可选的。通常已经设置了合适的默认值。注意,也可以用全局命令GPRINT打印轨道和CI矢量,或者在NATORB或CANORB选项中定义。

### 19.8.1 打印选项

ORBPRINT[,*nvirt*]

要求打印每个对称性中的占据轨道和*nvirt*个虚轨道(默认*nvirt*=0)。程序默认不打印轨道,除非出现了ORBPRINT指令或全局的GPRINT,ORBITALS命令(参见6.12节)。特定的轨道集可以在NATORB, CANORB或LOCORB卡(参见19.5.5节),用PRINT选项打印。如果需要在计算的最后打印附加信息,对*key1*, *key2*, ...使用

PRINT,*key1*,*key2*,...;

打开打印开关。如果打印每次迭代的信息,使用

IPRINT,*key1*,*key2*,...;

可用的打印关键词有:

MICRO            打印“微迭代”的细节—对于寻找不收敛的原因很有用。  
 REF              打印组态集的概要(仅CSF)。

REF1	打印组态集的列表（仅CSF）。
COR	打印CSF计算所用的中间空间的概要。
COR1	打印中间组态集的列表（仅CSF）。
PSPACE	打印构成“主要”空间的组态列表。
ORBITALS	打印轨道（参见ORBPRINT）。
NATORB	打印自然轨道（参见ORBPRINT）。
VIRTUALS	打印虚轨道（参见ORBPRINT）。
CIVECTOR	打印CI矢量（最好使用CANORB或NATORB）
INTEGRAL	打印转换积分（仅用于测试!）。
DENSITY	打印密度矩阵。
HESSIAN	打印hessian。
DIAGONAL	打印hessian的对角元。
GRADIENT	打印梯度。
LAGRANGI	打印拉格朗日量。
STEP	打印升级的矢量。
ADDRESS	打印寻址信息（仅用于测试!）。
DEBUG	打印调试信息。
CI2	打印例程ci2的调试信息（警告：可能相当长!）。
IO	打印I/O例程的调试信息。

### 19.8.2 收敛阈值

修改收敛阈值可以用

ACCURACY,[GRADIENT=*conv*] [,STEP=*sconv*] [,ENERGY=*econv*]

其中：

<i>conv</i>	轨道梯度的阈值（默认 $10^{-2}$ ）。
<i>econv</i>	总能量变化的阈值（默认 $10^{-6}$ ）。
<i>sconv</i>	步长的阈值（默认 $10^{-3}$ ）。

默认值可以用全局命令GTHRESH修改（参见6.11节）。上面的默认值通常是合适的。

### 19.8.3 最大迭代数量

MAXITER,*maxit*;



*maxit*是最大的迭代数量（默认6）。在默认的迭代数量内如果计算不收敛，在增加迭代数量之前应当仔细考虑不收敛的原因。大多数情况下是选择的活性轨道或优化的态不合适（参见MULTI 的介绍）。

#### 19.8.4 测试选项

TEST,*i1, i2, i3, ...*;

激活标号为*i1, i2, ...*的测试选项。除非弄清楚要做什么，否则不要使用。

#### 19.8.5 特殊的优化参数

以下参数也可以作为MULTI命令行的选项给出。

STEP,*radius, trust1, tfac1, trust2, tfac2*;

用于扩充Hessian方法的特殊参数。仅用于专家！

GOPER,*igop*;

在微迭代中使用G算符技术（默认）。如果*igop*大于0，则不使用G算符。

COPT,*ciacc, copvar, maxci, cishft, icimax, icimx1, icimx2, icstrt, icstep*;

用于直接CI方法的特殊参数。仅用于专家！

<i>ciacc</i>	CI对角化的梯度阈值。
<i>copvar</i>	CI优化的初始阈值。
<i>maxci</i>	每次微迭代中CI优化的最大次数。
<i>cishft</i>	q空间的分子移动。
<i>icimax</i>	在第一个微迭代中CI优化的最大次数。
<i>icimx1</i>	在第二微迭代及后来的迭代中，CI优化的最大次数。
<i>icimx2</i>	在内吸收步骤中，CI优化的最大次数。
<i>icstrt</i>	进行CI优化的第一次微迭代。
<i>icstep</i>	CI优化之间的微迭代增量。

INTOPT,*maxito, maxitc, maxrep, nitrep, iuprod*;

用于内优化方案的特殊参数。仅用于专家！

NONLINEAR,*itmaxr, ipri, drmax, drdamp, gfak1, gfak2, gfak3, irdamp, ntexp*

用于非线性优化方案的特殊参数。仅用于专家！

旧的形式（已作废）：

THRESH,*thrpri, thrpun, varmin, varmax, thrdiv, thrdoub*

新的形式：

THRESH [,THR PRI=*thrpri*] [,THR PUN=*thrpun*] [,VAR MIN=*varmin*]  
 [,VAR MAX=*varmax*] [,THR DIV=*thrdiv*] [,THR DOUB=*thrdoub*]

*thrpri*                    打印CI因子的阈值（默认0.04）。  
*thrpun*                   把CI因子写入punch文件的阈值。默认不写入punch文件中。  
*varmin, varmax*          非线性优化方案的阈值。仅用于专家！  
*thrdoub*                用于探测近于双占据轨道的阈值，用于包含到赝正则集中（默认0，也即，该功能禁用）。

DIIS, *disvar*, *augvar*, *maxdis*, *maxaug*, *idsci*, *igwgt*, *igvec*, *idstrt*, *idstep*;  
 用于DIIS收敛加速的特殊参数。仅用于专家！

### 19.8.6 为CASVB保存波函数信息

VBDUMP[,*vbdump*];

对于价键程序CASVB的用户，随后可能用到的所有波函数信息都保存到*vbdump*记录中。默认不写入这一信息。如果定义这个关键词时没有定义*vbdump*的值，那么使用记录4299.2。这个关键词不必出现在变分CASVB计算之前。

### 19.8.7 保存转换积分

TRNINT,*trnint*;

在以活性CASSCF MO为基的积分中，*trnint*定义记录名称。例如可用于CASVB计算（参见38.5节）。*trnint*的默认值是1900.1。

## 19.9 耦合微扰MCSCF

对于用态平均轨道计算梯度，非绝热耦合，差分梯度或者极化率的计算，需要耦合微扰MCSCF。我们请大家注意，目前的程序只是初步的，效率还不是很高。

### 19.9.1 SA-MCSCF的梯度

要计算态平均梯度，使用

CPMCSCF, GRAD, *state*, [SPIN=*spin*], [MS2=*ms2*], [ACCU=*thresh*], [RECORD=*record*]

其中*state*指定需要计算梯度的态（例如，2.1表示对称性1的第二个态）。*spin*指定态的自旋：这是相应WF卡中相应的值的一半（例如，0=单重态，0.5=双重态，1=三重态）。也可以使用MS2，这里的 $ms2 = 2 * spin$ ，也就是说，与在WF卡的定义相同。只有在对不

同自旋的态进行平均时，SPIN或MS2的定义才是必需的。*record*指定存储梯度信息的记录（默认是5101.1）。*thresh*是求解CP-MCSCF的精度阈值，默认值是1.d-7。

梯度通过随后调用FORCES或OPTG来计算。

注意：如果由于某些原因，需要从有限能量差分计算数值梯度，在态平均计算中需要给出以下命令，而不是CPMCSCF输入：

SAVE,GRAD=-1

否则程序停止并出现错误信息。

### 19.9.2 SA-MCSCF的差分梯度

如果计算差分梯度，使用

CPMCSCF,DGRAD,*state1*,*state2*,[ACCU=*thresh*],[RECORD=*record*]

其中*state1*和*state2*指定被考虑的两个态（例如，2.1,3.1表示对称性1的第二和第三个态）。将计算能量差分梯度。两个态必须有同样的对称性。*record*指定存储梯度信息的记录（默认是5101.1）。*thresh*是求解CP-MCSCF的精度阈值，默认值是1.d-7。

梯度通过随后调用FORCES或OPTG来计算。

### 19.9.3 SA-MCSCF的非绝热耦合矩阵元

计算解析的非绝热耦合矩阵元，使用

CPMCSCF,NACM,*state1*,*state2*,[ACCU=*thresh*],[RECORD=*record*]

其中*state1*和*state2*指定被考虑的两个态（例如，2.1,3.1表示对称性1的第二和第三个态）。两个态必须有同样的对称性。*record*指定存储梯度信息的记录（默认是5101.1）。这将在随后的梯度计算中读入。*thresh*是求解CP-MCSCF的精度阈值，默认值是1.d-7。

NADC和NADK是NADC的别名，SAVE是RECORD的别名。

每个原子的矩阵元通过后面FORCES的调用来计算。

注意：这个程序还没有做大量的测试，使用时应当心。

### 19.9.4 MCSCF的hessian

可以解析计算MCSCF/CASSCF的hessian（只能用于没有对称性的情况），使用  
CPMCSCF,HESS,[ACCU=*value*]

其中的ACCU选项指定CPMCSCF计算的收敛阈值（默认1.d-4）。Hessian存储于5300.2记录，可用于随后的频率计算。

例子：

```
{multi;cpmcscf,hess,accu=1.d-5}
frequencies
```

注意，在计算hessian时，使用了NOEXTRA选项。

## 19.10 优化价键波函

VB={...}

使用这个关键词，用CASVB优化CI因子。VB关键词之后可以接第三十八章的任何指令。默认基于能量优化VB参数，主要CASVB迭代的输出级别减小到-1。

## 19.11 提示与诀窍

MCSCF不是像SCF那样的一个“黑箱”流程！对于简单的情况，例如一个没有CLOSED轨道的简单CASSCF计算，这个程序将在两到三步迭代内收敛。对于更复杂的情况，可能会遇到很多问题。遇到这类情况，可以这样考虑：

- 如果可能，总是从邻近结构的轨道开始（这是默认的）。
- 如果没有CLOSED轨道，也就是说，在全部组态中除了完全优化的轨道就是双占据轨道，那么收敛的算法更稳定。因此合理的近似是在初始计算中用FROZEN代替CLOSED（冻结全部双占据轨道）。
- 如果还不成功，可以对一些迭代关闭CI因子之间的耦合和轨道转动，例如：

```
{ITERATIONS;DO,UNCOUPLE,1,TO,2;}
```

以及/或者禁止同时优化内部轨道和CI，例如：

```
{ITERATIONS;DONT,INTERNAL,1,TO,2;}
```

如果想得到程序从何处开始发散的线索，可以在输入中包含  
IPRINT,MICRO;

还要考虑本章开始时的一般性注释。所用算法的细节，参见J. Chem. Phys. 82, 5053 (1985); Chem. Phys. Letters 115, 259 (1985); Advan. Chem. Phys. 59, 1 (1987)。

## 19.12 例子

CASSCF计算H<sub>2</sub>O，C<sub>2v</sub>对称性的最简单输入是：

```
! $Revision: 2006.3 $  
geometry={o;h1,o,r;h2,o,r,h1,theta} ! Z矩阵结构输入  
r=1 ang ! 键长
```

```

theta=104          ! 键角
hf                 ! 进行scf计算
multi              ! 进行全部价电子的casscf

```

[http://www.molpro.net/info/current/examples/h2o\\_casscf.com](http://www.molpro.net/info/current/examples/h2o_casscf.com)

它还可以展开，例如以下的输入卡：

```

OCC,4,1,2;        ! 指定占据空间
CLOSED,2           ! 指定闭壳层的（非活性的）轨道
FROZEN,1;          ! 指定冻芯轨道
WF,10,1;           ! 定义波函数对称性
START,2100.2;      ! 从文件2的记录2100读取猜测轨道
ORBITAL,2140.2;    ! 把最终轨道保存到文件2的记录2140
NATORB,PRINT,CI    ! 打印自然轨道，并对自然轨道做哈密顿量的对角化。
                   ! 还打印最大的CI因子。

```

CN态平均计算的例子， $X$ 和 $B^2\Sigma^+$ 态，以及 $A^2\Pi_x$ ， $^2\Pi_y$ 态的平均。执行全部价电子的CASSCF计算：

```

! $Revision: 2006.0 $
***,cn
r=2.2              ! 定义键长
geometry={c;n,c,r}
{rhf;occ,5,1,1;wf,13,1,1;          ! sigma态的RHF计算
orbital,2100.2}                  ! 把轨道保存到2100.2记录中（默认）

{multi;occ,6,2,2;closed,2;         ! 定义活性和非活性轨道
start,2100.2;                      ! 从上面的RHF轨道开始
save,ref=4000.2                    ! 把CI的组态权重保存到4000.2记录中
wf,13,1,1;state,2;wf,13,2,1;wf,13,3,1; ! 定义四个态
natorb,ci,print;                  ! 打印自然轨道和相应的CI因子
tran,lz                           ! 计算LZ的矩阵元
expec2,lzz}                       ! 计算LZZ的期待值

```

[http://www.molpro.net/info/current/examples/cn\\_sa\\_casscf.com](http://www.molpro.net/info/current/examples/cn_sa_casscf.com)

对 $N_2$ 进行RASSCF（限制活性空间）计算的例子，包括SCF行列式加上全部到价轨道的双激发。单激发排除在外。使用 $D_{2h}$ 对称性，CSF方法：

```

! $Revision: 2006.0 $
***,N2
geometry={N1;N2,N1,r}            ! 结构输入
r=2.2                             ! 键长
{hf;occ,3,1,1,,2;wf,14,1;save,2100.2} ! scf计算

{multi;occ,3,1,1,,3,1,1;         ! 定义占据轨道

```

```
freeze,1,,,1,2100.2;      ! 定义scf冻芯轨道
config;                    ! 使用CSF方法
wf,14,1;                   ! 定义态的对称性
restrict,0,2,3.5,1.6,1.7;  ! 限制为单、双激发
restrict,-1,-1,3.5,1.6,1.7; ! 排除单激发
print,ref1                 ! 打印组态
natorb,ci,print}           ! 打印自然轨道和CI因子
```

[http://www.molpro.net/info/current/examples/n2\\_rasscf.com](http://www.molpro.net/info/current/examples/n2_rasscf.com)

## 多组态参考内收缩组态相互作用程序

## 参考文献:

H.-J. Werner and P.J. Knowles, J. Chem. Phys. 89, 5803 (1988).

P.J. Knowles and H.-J. Werner, Chem. Phys. Lett. 145, 514 (1988).

使用本程序的所有出版结果必须引用以上文献。此外参见:

H.-J. Werner and E.A. Reinsch, J. Chem. Phys. 76, 3144 (1982).

H.-J. Werner, Adv. Chem. Phys. 59, 1 (1987).

使用CI或CI-PR0命令调用程序。CISD命令调用快速闭壳层CISD程序。QCI 命令调用闭壳层二次CI程序。CCSD命令调用闭壳层耦合簇程序。

在命令行可以指定以下选项:

NOCHECK	若不收敛的话, 不停止。
DIRECT	进行积分直接计算。
NOSING	不包含单激发外部组态。
NOPAIR	不包含双激发外部组态 (对单参考方法无效)。
MAXIT= <i>value</i>	最大迭代次数。
MAXITI= <i>value</i>	最大微迭代次数 (用于内组态)。
SHIFTI= <i>value</i>	用于内组态升级的分母移动。
SHIFTS= <i>value</i>	用于单激发外组态升级的分母移动。
SHIFTP= <i>value</i>	用于双激发外组态升级的分母移动。
THRDEN= <i>value</i>	能量的收敛阈值。
THRVAR= <i>value</i>	CI矢量的收敛阈值。它用于CI因子变化的平方和。

## 20.1 简介

内收缩MRCI程序用CI命令调用。这包含作为特例的单参考CI, CEPA, ACPF, MR-ACPF和MR-AQCC。对于闭壳层的参考函数, 还有更快的特殊代码, 这可以用CISD, QCI 或CCSD命令调用。对这三种情况也允许计算Brueckner轨道 (此时QCI和CCSD相同)。

如果没有进一步的输入卡, 波函数的定义 (芯、闭壳层和活性轨道空间, 对称性) 对应于最近一次SCF或MCSCF计算使用的定义。默认产生CASSCF参考空间。其它选项可以用OCC, CORE, CLOSED, WF, SELECT, CON和RESTRICT卡。轨道来自相应的SCF或MCSCF计算, 除非给定了ORBITAL指令。波函数可以用SAVE指令保存, 用START续算。EXPEC指令允许计算单电子算符的期待值, TRAN指令可用于计算单电子特性的跃迁矩阵元。用NATORB指令可以打印并保存自然轨道。

对于激发态的计算, 参见STATE, REFSTATE和PROJECT。

## 20.2 定义波函数

### 20.2.1 占据轨道

OCC, $n_1, n_2, \dots, n_8$ ;

$n_i$ 指定在不可约表示 $i$ 中的占据轨道数 (包括CORE和CLOSED)。如果没有给定, 默认的信息来自最近的SCF, MCSCF或CI计算。

### 20.2.2 冻芯轨道

CORE, $n_1, n_2, \dots, n_8$ ;

$n_i$ 是不可约表示 $i$ 中的冻芯轨道数。这些轨道在全部组态中都是双占据的, 即: 不进行关联处理。如果没有给出CORE卡, 程序使用和最后一次CI计算相同的芯轨道; 如果没有做CI计算, 那么原子的内壳层作为芯处理。为了避免这一功能, 并对所有电子计算关联能, 可以定义:

CORE

### 20.2.3 闭壳层轨道

CLOSED, $n_1, n_2, \dots, n_8$

$n_i$ 指定在不可约表示 $i$ 中的闭壳层轨道数, 包含所有芯轨道。这些轨道不构成任何活性空间, 即: 它们在所有的参考CSF中都是双占据的; 但是与芯轨道相反 (见CORE), 这些轨道通过单、双激发进行关联。如果没有给出这个卡, 这些信息默认来自最近



的SCF, MCSCF或CI计算。对于闭壳层参考函数的计算 (closed=occ), 参见CISD, QCI和CCSD。

#### 20.2.4 定义轨道

ORBIT, *name.file*, [*specifications*];

*name.file*定义的记录用于从中读取轨道。如果*name.file*包含一个以上的轨道集, 可以选择给出各种*specifications*以选择特定轨道。细节详见4.11节。注意, IGNORE\_ERROR选项可用于强制用非正则轨道进行MP<sub>n</sub>或三激发计算。

默认是来自最后一次SCF, MCSCF或CI计算的轨道集。

#### 20.2.5 定义态的对称性

电子数和波函总对称性可以用WF卡定义:

WF, *elec*, *sym*, *spin*

其中:

<i>elec</i>	电子数。
<i>sym</i>	不可约表示编号。
<i>spin</i>	定义自旋对称性, $spin = 2S$ (单重态=0, 双重态=1, 三重态=2, 等等)。

WF卡必须位于定义轨道空间的全部卡 (OCC, CORE, CLOSED) 之后。

REF卡可用于进一步定义产生组态空间的参考对称性, 参见REF。

#### 20.2.6 附加的参考对称性

REF, *sym*;

这个卡必须接在WF指令之后, 定义用于产生非收缩内部和单激发外部组态空间的附加参考对称性。如果使用了不同点群对称性, 这对于获得相同的组态空间有时候会很有用。例如对 $C_s$ 对称性的计算,  $\Pi$ 态的两个分量 $A'$ 和 $A''$ 可能会不完全简并。这个问题可以用以下的例子避免:

对于双重 $A'$ 态:

```
WF,15,1,1;      ! 定义波函对称性(1)
REF,2;          ! 定义附加的参考对称性(2)
```

对于双重 $A''$ 态:

```
WF,15,2,1;      ! 定义波函对称性(2)
```

```
REF,1;           ! 定义附加的参考对称性(1)
```

对于线型结构，用 $C_{2v}$ 对称性可以获得同样的结果，而且计算更便宜，

```
WF,15,2,1;      ! 定义波函对称性(2)
REF,1;          ! 定义附加的参考对称性(1)
REF,3;          ! 定义附加的参考对称性(3)
```

或者：

```
WF,15,3,1;      ! 定义波函对称性(3)
REF,1;          ! 定义附加的参考对称性(1)
REF,2;          ! 定义附加的参考对称性(2)
```

每个REF卡之后可以是按照给定顺序的RESTRICT，SELECT，和CON卡。

### 20.2.7 选择组态

```
SELECT,ref1,ref2,refthr,refstat,mxshrf;
```

这个卡用来定义参考组态集，而不是默认CAS。组态可以用CON卡定义，它必须出现在SELECT卡之后。如果 $ref1$ 是已经存在的MOLPRO记录名称，组态也可以从这个记录读入，并且可以按照给定的阈值进行筛选。

SELECT卡通常应当直接位于WF或REF卡之后，或者RESTRICT卡之后（如果出现的话）。这些卡的一般顺序是：

```
WF （或REF）
RESTRICT （可选）
SELECT （可选）
CON （可选）
```

$ref1=rec1.file$  （ $rec1>2000$ ）从指定的卡读入组态。关于在MCSCF计算中如何保存组态，参见19.5.4节。如果 $ref1$ 没有指定，程序假定组态从后面的CON卡读入（参见CON）。

$ref2=rec2.file$  （ $rec2>2000$ ）从指定的卡读入附加组态。如果 $rec2$ 为负，则读取 $rec1$ 和 $abs(rec2)$ 之间的所有的记录。用这种方法发现的所有组态进行合并。

$refthr$  选择从磁盘（记录 $rec1-rec2$ ）读取组态的阈值。应用于每个轨道组态所有CSF的模。

- refstat* 指定从哪个态矢量选择组态。只用于态平均计算中保存组态的情况。如果*refstat*是零或没有指定，组态从所有的态中选择。如果*refstat*大于零，则使用指定的参考态。如果*refstat*小于零，那么使用所有合适的参考态。最后，如果*refstat*是*istat1.istat2*的形式，则使用从*istat1*到*istat2*的态。
- mxshrf* 在选择的或产生的组态中，开壳层的最大数量。

### 20.2.8 占据限制

RESTRICT,*nmin*,*nmax*,*orb*<sub>1</sub>,*orb*<sub>2</sub>,...*orb*<sub>*n*</sub>;

这个卡用于限制参考组态的占据方式。只有在指定轨道*orb*<sub>1</sub>, *orb*<sub>2</sub>, ..., *orb*<sub>*n*</sub>中包含*nmin*和*nmax*个电子之间的组态，才能包含在参考函数中。如果*nmin*和*nmax*为负，那么在指定轨道中恰好有abs(*nmin*)和abs(*nmax*)个电子的组态将被删除。例如，这可以用于忽略单激发组态。轨道按照*number.sym*的形式定义，其中*number*是不可约表示*sym*中的轨道序号。可以连续用几个RESTRICT卡。

RESTRICT卡必须接在所用的WF或REF卡之后。这些卡的一般顺序是：

WF (或REF)  
 RESTRICT (可选)  
 SELECT (可选)  
 CON (可选)

如果RESTRICT卡先于WF卡，则它应用于所有的参考对称性中。注意RESTRICT也影响SELECT和/或CON卡产生的空间。

### 20.2.9 直接指定参考组态

CON,*n*<sub>1</sub>,*n*<sub>2</sub>,*n*<sub>3</sub>,*n*<sub>4</sub>,...

指定包含在参考函数中的轨道组态。*n*<sub>1</sub>, *n*<sub>2</sub>等是活性轨道的占据数(0, 1, 或2)。可以连续用任意数量的CON卡，但是它们必须直接接在SELECT卡之后。

### 20.2.10 定义态的数量

STATE,*nstate*,*nroot*(1),*nroot*(2),...,*nroot*(*nstate*);

*nstate*是同时处理的态的数量；*nroot*(*i*)是计算的根的编号。它们用于初始内部CI中态的顺序。如果没有定义，则*nroot*(*i*)=*i*。注意，可以跳过电子态，即

STATE,1,2;     ! 计算第二个态

```
STATE,2,1,3; ! 计算第一和第三个态
```

所有指定的态必须用内组态空间合理描述。对每个态可以用不同的收敛阈值（参见ACCU卡）。还可以对包含在列表 $nroot(i)$ 中一些较低的根不进行收敛（参见REFSTATE卡）。例子参见REFSTATE卡。

### 20.2.11 定义参考态的编号

```
REFSTATE,nstatr,nrootr(1),nrootr(2),...,nrootr(nstatr);
```

*nstatr*是产生收缩电子对的参考态数量。它可以大于或小于 $nstate$ 。如果这个卡没有出现，那么 $nstatr=nstate$ 且 $nrootr(i)=nroot(i)$ 。在STATE卡定义的根（或者默认包含的根，如果在STATE卡没有定义 $nroot(i)$ 的话）如果没有定义参考态，将不会进行收敛，因为结果总是坏的。但是在 $nroot(i)$ 列表中包含这些态通常会很有用，因为它可以帮助避免根的震荡问题。例如：

```
state,2;
```

将会用两个参考态计算两个态。

```
state,2;refstate,1,2;
```

将会用一个参考态优化第二个态。为了避免根的震荡，对于基态将会产生一个外部展开矢量。第一个态打印的结果很差，不能使用（除非是完整的电子对空间，这可能发生在很小的计算中）。

```
state,1,2;refstate,1,2;
```

类似于第二个例子，但是对基态没有产生外部展开矢量。如果没有根的震荡（然而这是经常会发生的），这会给出与前面的态2完全相同的能量。

```
state,2;accu,1,1,1;
```

将会用两个参考态计算第二个态。基态不进行收敛计算（对态1只有一次迭代）。这会给出与第一个例子中态2完全相同的能量。

### 20.2.12 定义轨道对的关联

```
PAIR,iorb1.isy1,iorb2.isy2,np;
```

请求对给定的轨道对进行关联能计算。

$np=1$	单重电子对
$np=-1$	三重电子对
$np=0$	单重和三重电子对（如果可能）

默认对活性和闭壳层轨道所有的电子对进行关联计算。另外参见PAIRS卡。

PAIRS,iorb1.isy,iorb2.isy,np;

对iorb1.isy1到iorb2.isy2轨道的所有电子对进行关联计算。芯轨道排除在外。要么iorb2大于iorb1，要么isy2大于isy1。如果iorb1.isy1=iorb2.isy2，那么PAIRS卡与PAIR卡的作用相同。PAIR和PAIRS卡可以结合使用。

如果PAIR和PAIRS卡都没有定义，则对所有的价轨道进行关联计算。产生的电子对列表不仅用于限制双激发外部组态，还限制所有的内部和半内部组态。

### 20.2.13 激发类型的限制

NOPAIR;

不包含双激发的外部组态。

NOSINGLE;

不包含单激发的外部组态。

NOEXC;

仅对参考组态执行CI。

## 20.3 选项

### 20.3.1 耦合电子对近似

CEPA(ncepa);

( $0 \leq ncepa \leq 3$ )。不做哈密顿量对角化，执行CEPA计算，CEPA类型ncepa。目前仅能用于单组态参考函数。

### 20.3.2 耦合对泛函 (ACPF, AQCC)

ACPF,options

AQCC,options

其中的options可以是：

GACPFI=gacpfi

GACPFE=gacpfe

不做哈密顿量对角化，执行ACPF计算或者AQCC计算。利用GACPFI和GACPFE选项，内部和外部的归一化因子 $gacpfi$ 和 $gacpfe$ 可以分别从默认值 $1, 2/nelec$ 和 $1, 1-(nelec-2)(nelec-3)/nelec(nelec-1)$ 重新设置。

ACPF及有关的方法目前对激发态的计算不够稳定。尽管有时候可以计算，但我们目前不推荐，也不对做这类方法的激发态计算提供支持。

### 20.3.3 投影激发态的计算

PROJECT,record,nprojc;

开始或继续投影激发态的计算，并把信息存储于 $record$ 。如果 $nprojc > 0$ ，那么 $nprojc$ 个以前计算的CI矢量将用于产生投影算符。如果 $nprojc = -1$ ，这个计算强制对第一个态，即，基态不做投影。如果 $nprojc = 0$ ，并且 $record$ 不存在，那么作用同 $nprojc = -1$ ；否则 $nprojc$ 从 $record$ 的转存信息中获得。因此在计算的开始，最好使用 $project,record,-1$ ；而对于此后的激发态计算，使用 $project,record$ 。在计算的结束，保存波函，并升级转存记录 $record$ 中的信息。PROJECT卡还设置 $tranh$ 选项，因此默认计算跃迁哈密顿矩阵。

例如，要连续计算三个态，使用：

```
ci;...;project,3000.3,-1;
ci;...;project,3000.3;
ci;...;project,3000.3;
```

### 20.3.4 跃迁矩阵元选项

TRANH,option;

如果 $option > -1$ ，在TRANS或PROJECT计算中强制计算跃迁哈密顿矩阵元。如果 $option < 1$ ，强制计算单电子跃迁特性。

### 20.3.5 收敛阈值

ACCU,istate,energy,coeff;

$istate$ 态的收敛阈值。能量和CI因子的实际阈值分别是 $10^{**}(-energy)$ 和 $10^{**}(-coeff)$ 。如果这个卡没有出现，那么所有态的阈值使用默认值，或者在THRESH卡定义的值。

### 20.3.6 能级移动

SHIFT,shiftp,shifts,shifti;

分别是电子对，单激发和内部组态的分母移动。

### 20.3.7 最大数量的迭代

MAXITER,*maxit*,*maxiti*;

<i>maxit</i>	最大数量的宏迭代
<i>maxiti</i>	最大数量的微迭代（内部CI）

### 20.3.8 限制展开矢的数量

MAXDAV,*maxdav*,*maxvi*;

<i>maxdav</i>	在宏迭代中，外部展开矢的最大数量
<i>maxvi</i>	在内部CI中，内部展开矢的最大数量

### 20.3.9 选择主组态集

PSPACE,*select*,*nspac*;

<i>select</i>	选择p-空间组态的能量标准。如果为负，执行对p-空间H的测试。
<i>nspac</i>	p-空间组态的最小数量。如果指定，或者发现了与最终p-空间组态简并的组态，则添加更多的组态。最小数量的 <i>nspac</i> 自动由态的指定确定。

### 20.3.10 正则化外部轨道

FOCK,*n*<sub>1</sub>,*n*<sub>2</sub>,...;

外部轨道通过指定占据数*n<sub>i</sub>*的Fock算符本征函数获得。占据数必须提供给所有的价轨道。

### 20.3.11 保存波函

SAVE,*savecp*,*saveco*,*idelcg*;

或者

SAVE [,CIVEC=*savecp*] [,CONFIG=*saveco*] [,DENSITY=*dumpprec*] [,NATORB=*dumpprec*] [,FILES]

<i>savecp</i>	用于保存波函的记录名称。如果是负值，波函在每次迭代后保存，否则只在任务结束后保存。对于耦合簇方法的计算（包括CCSD，QCISD，BCCD），波函无论何时都在每次迭代中保存（目前只能用于闭壳层计算的情况）。
---------------	---

<i>saveco</i>	用于对所有态保存内部组态及其最大权重的记录名称，用于后面的参考输入（参见SELECT卡）。如果记录已经存在，记录名称依次加一，直到能够创建新记录。
<i>idelcg</i>	如果非零或定义了FILES，那么在计算结束后不擦除icfil和igfil（保存有CI和残差矢量）。
<i>dumprec</i>	保存密度矩阵和自然轨道的转存记录。只能给定一个转存记录。无论何时都保存密度矩阵和自然轨道。参见DM或NATORB卡。

### 20.3.12 初始波函

START,*readc1*,*irest*;

<i>readc1</i>	储存波函的记录名称对于续算。对于耦合簇方法的计算（包括CCSD, QCISD, BCCD），振幅从记录 <i>readc1</i> 读取，并用于重新开始计算（目前只能用于闭壳层计算的情况）。
<i>irest</i>	如果非零，读取CI因子并用于续算；否则只读入波函的定义。

### 20.3.13 单电子特性

EXPEC,*oper*<sub>1</sub>,*oper*<sub>2</sub>,*oper*<sub>3</sub>,...;

在波函行列式之后，对单电子算符*oper*<sub>*i*</sub>计算期待值。关于可用的算符及其关键字，参见6.13节。在多个态或投影的计算中，还计算跃迁矩阵元。

### 20.3.14 计算跃迁矩

TRANS,*readc1*,*readc2*,[BIORTH],[*oper*<sub>1</sub>,*oper*<sub>2</sub>,*oper*<sub>3</sub>,...];

不做能量计算，只计算保存在记录*readc1*和*readc2*中波函之间的跃迁矩阵元。关于可用的算符及其相应的关键字列表，参见6.13节。如果没有定义算符名称，计算偶极跃迁矩。

如果给出了BIORTH选项，那么两个波函可以使用不同的轨道。不过，活性轨道数和非活性轨道数必须一致。注意，BIORTH不能用于自旋-轨道矩阵元的计算。在某些情况下可能无法进行双正交归一化，这时会打印错误信息。

### 20.3.15 保存密度矩阵

DM,*record.ifil*,[*idip*];



所有计算的态的一阶密度矩阵存储于文件`ifil`的`record`记录中。如果`idip`非零, 则偶极矩从第`idip`次迭代开始打印。参见NATORB。对于跃迁矩的计算, 如果所涉及的两个态对称性相同, 还存储跃迁密度。

### 20.3.16 自然轨道

NATORB,[RECORD=`record.ifil`],[PRINT=`nprint`],[CORE[=`natcor`]];

计算自然轨道。在任何给定的对称性中, 打印的外部轨道数是`nprint` (默认2)。`nprint=-1`禁止打印。如果`record`非零, 所有态的自然轨道和密度矩阵保存在`ifil`文件的转存记录`record`中。如果`record.ifil`在DM卡定义 (见上), 则使用这个记录。如果在DM和NATORB卡定义了不同的记录, 将会导致错误。记录也可以在SAVE卡给出。如果定义了CORE, 则不打印芯轨道。

注意: 转存记录不必和SAVE卡的`savecp`或`saveco`, 或者PROJECT给定的记录相同。

### 20.3.17 杂项

OPTION,`code1=value,code2=value,...`

可用于定义程序的参数和选项。如果没有定义代码和它的值, 显示默认值。等号可以忽略。允许以下的代码 (每个卡最多7个):

NSTATE:	参见state卡
NSTATI:	在内部CI计算的态数量
NSTATR:	参见refstat卡
NCEPA:	参见CEPA卡
NOKOP:	如果非零, 跳过积分转换
ITRDM:	如果大于等于0, 计算跃迁矩
ITRANS:	如果非零, 执行完全积分转换 (还不能执行)
IDIP:	打印来自第 <code>value</code> 次迭代的偶极矩
REFOPT:	如果非零, 优化参考因子; 否则从内部CI提取参考因子
IAVDEN:	如果非零, 对自旋耦合进行HII和HSS分母的平均
IDELCG:	如果不等于零, 那么结束时删除icfil,igfil文件
IREST:	如果非零, 续算
NATORB:	如果非零, 计算并打印自然轨道。每个对称性打印的外部轨道数量是 $\min(\text{natorb}, 2)$
WFNAT:	如果非零, 把自然轨道保存到该记录中
IPUNRF:	如果非零, 对参考组态的因子进行存档

NPUPD:	如果非零, 升级非正交基的电子对, 否则是正交基
MAXIT:	参见maxiter卡
MAXITI:	参见maxiter卡
MAXDAV:	参见maxdav卡
MAXVI:	参见maxdav卡
NOSING:	参见nosing卡
NOPAIR:	参见nopair卡
MXSHRF:	参见select卡
IKCPS=0:	在CIKEXT中, 只计算K(CP); 这个选项用于并且只能用于没有单激发的时候
IKCPS=1:	只计算K(CP')。意味着使用修改的耦合因子
IKCPS=2:	计算K(CP)和K(CP')。默认是IKCPS=2, 但对单参考组态默认是IKCPS=1。
IOPTGM:	密度矩阵例程的选项
IOPTGM=0:	密度矩阵例程中, 所有的量对每个中间对称性重新计算 (CPU最大, 内存最小)
IOPTGM=1:	预先计算数值, 并存储到硬盘 (I/O最大, 内存最小)
IOPTGM=2:	预先计算数值, 并保存到内存 (内存最大, CPU最小)
IOPTOR:	如果非零, 对每个电子对计算中间轨道。有些情况下可以改善收敛, 特别是当使用了局域轨道时。

### 20.3.18 各种参数

PARAM, *code1=value, code2=value...*

重新定义系统参数。如果没有指定代码, 显示默认值。允许以下的代码:

LSEG:	磁盘扇区长度
INTREL:	每REAL*8字的整数数量
IVECT=0:	标量计算机
IVECT=1:	矢量计算机
MINVEC:	如果矢量长度大于这个值, 在耦合因子例程中调用MXMB
IBANK:	矢量计算机的内存单元数量。如果IBANK>1, 作为IBANK倍数的矢量跨度只要遇到就会避开
LTRACK:	每个磁道或区块以REAL*8字为单位的数量 (用于文件定位)

LTR:	确定矩阵如何存储到磁盘上。如果LTR=LSEG, 所有的矩阵从扇形边界开始(这将优化I/O), 但(在磁盘和内存中)未使用的空间位于矩阵之间。对于LTR=1, 所有的矩阵密集存储。如果需要很多页面调度, 这会增加I/O, 但是如果所有矩阵能放在内存中, 将会降低I/O。
NCPUS:	用于多任务处理的CPU最大数量

## 20.4 各种阈值

THRESH, *code1=value, code2=value...*

如果 *value*=0, 相应的阈值设置为0, 否则是  $10^{**}(-value)$ 。等号可以忽略。如果没有定义代码, 那么打印默认值。可以使用以下代码(每个卡最多7个):

ZERO:	数值0
THRDLP:	如果重叠矩阵的本征值小于这个阈值, 删除电子对
PNORM:	如果模小于这个阈值, 删除电子对(对闭壳层的情况, 所有的电子对都是归一化的)
PRINTCI:	打印大于这个值的CI因子
INTEG:	忽略小于这个值的双电子积分
ENERGY:	能量收敛阈值; 参见ACCU卡
COEFF:	因子收敛阈值; 参见ACCU卡
SPARSE:	忽略小于这个值的因子变化
EQUAL:	如果内部矢量和对角元的差小于这个值, 认为它们相等。对于对称性的追踪非常有用

## 20.5 打印选项

PRINT, *code1=value, code2=value...*

打印选项。一般而言, 这个值确定打印多少中间信息。*value*=-1表示不打印(对所有的代码都是默认的)。在有些情况下, 指定以下列出的较高的值将产生比说明更多的输出。等号和零可以忽略。所有的代码可以缩写为前三个字符。可以使用以下代码(每个卡最多7个):

ORBITALS:	打印轨道
JOP=0:	打印算符列表
JOP=1:	打印MO基中的库仑算符

JOP=2:	打印AO和MO基中的库仑算符
KOP:	类似于JOP, 用于内部交换算符
KCP=0:	对CIKEXT打印分页信息
KCP=1:	打印MO基中的外部交换算符
KCP=2:	打印AO和MO基中的算符
DM=0:	对CIDIMA打印分页信息
DM=1:	打印MO基中的密度矩阵
DM=2:	打印AO和MO基中的密度矩阵
FPP=0:	对电子对打印能量分母
FPP=1:	另外还打印正交基中的对角耦合因子
FPP=2:	打印算符FPP
CP=0:	打印每次迭代中电子对的升级信息
CP=1:	打印电子对矩阵的升级 (MO基)
CP=2:	此外还打印电子对矩阵 (MO基)
CP=3:	打印AO基中的CP (在CIKEXT中)
CI=0:	对内部CI打印收敛信息
CI=1:	打印内部CI因子和外部展开因子
CS:	同CP, 用于单激发
CPS=0:	对CICPS打印分页信息
CPS=1:	打印MO基中的矩阵CPS
GPP=0:	对CIGPQ打印分页信息
GPP=1:	在CIGPQ退出时打印矩阵GP
GPS=0:	对CIGPS打印分页信息
GPS=1:	在CIGPS退出时打印矢量GS
GSP=1:	在CIGPS退出时打印矩阵GP
GPI=0:	对CIGPI打印分页信息
GPI=1:	打印正交基中的总GP
GPI=2:	打印矩阵GP和TP
GIP=0:	对CIGIP打印分页信息
GIP=1:	在CIGIP退出时打印GI
GSS=0:	对CIGSS打印分页信息
GSS=1:	在CIGSS退出时打印矢量GS
GSI=0:	对CIGSI打印分页信息

GSI=1:	在CIGSI退出时打印GS
GIS=0:	对CIGIS打印分页信息
GIS=1:	在CIGIS退出时打印GI
GII:	打印内部CI的中间信息
DPQ:	打印耦合因子 $\alpha(P, Q)$
EPQ:	打印耦合因子 $\beta(P, Q)$
HPQ:	打印耦合因子 $\gamma(P, Q)$
DPI:	对电子对—内部相互作用, 打印耦合因子
DSS:	还不能使用
DSI:	还不能使用
LOG:	在每一迭代的最后, 把概要写入日志文件中。如果LOG=0, 在任务结束后删除。
CC=0:	打印耦合因子的地址列表
CC=1:	打印耦合因子
DEN=1:	打印内部一阶密度
DEN=2:	打印内部二阶密度
DEN=3:	打印内部三阶密度
DEN=4:	打印一阶、二阶、三阶密度
GAM=1:	打印一阶跃迁密度
GAM=2:	打印二阶跃迁密度
GAM=3:	打印一阶和二阶跃迁密度
PAIRS=0:	打印非冗余电子对列表
PAIRS=1:	打印所有电子对列表
CORE=0:	打印内部组态 ( $N$ , $N-1$ 和 $N-2$ 电子) 的概要
CORE=1:	打印内部组态 ( $N$ , $N-1$ 和 $N-2$ )
REF=0:	打印参考组态的概要
REF=1:	打印参考组态及其因子
PSPACE:	打印p-空间组态
HII:	打印内部的对角元
HSS:	打印单激发的对角元
SPQ:	在电子对正交化的例程中, 各种级别的中间信息
TEST=0:	打印每个子程序调用的信息
TEST=1:	另外还打印LESW和SREIBW中关于I/O的信息

TEST=2:            还打印FREAD, FWRITE中关于I/O的信息  
CPU:                打印CPU和I/O时间的分析  
ALL:                打印给定级别的所有信息（使用时当心！）

## 20.6 例子

```
! $Revision: 2006.0 $
***,Single reference CISD and CEPA-1 for water
r=0.957,angstrom
theta=104.6,degree;
geometry={0;                            ! z矩阵结构输入
         H1,0,r;
         H2,0,r,H1,theta}
{hf;wf,10,1;}                          ! 总SCF能量        -76.02680642
{ci;occ,3,1,1;core,1;wf,10,1;}        ! 总CI(SD)能量   -76.22994348
{cepa(1);occ,3,1,1;core,1;wf,10,1;}   ! 总CEPA(1)能量 -76.23799334
```

[http://www.molpro.net/info/current/examples/h2o\\_cep1.com](http://www.molpro.net/info/current/examples/h2o_cep1.com)

```
! $Revision: 2006.0 $
***,Valence multireference CI for X and A states of H2O
gthresh,energy=1.d-8
r=0.957,angstrom,theta=104.6,degree;
geometry={0;                            ! z矩阵结构输入
         H1,0,r;
         H2,0,r,H1,theta}
{hf;wf,10,1;}                          ! 总SCF能量        -76.02680642
{multi;occ,4,1,2;closed,2;freeze,1;wf,9,2,1;wf,9,1,1;tran,ly}
                                      ! MCSCF能量        -75.66755631
                                      ! MCSCF能量        -75.56605896
{ci;occ,4,1,2;closed,2;core,1;wf,9,2,1;save,7300.1}
                                      ! 总MRCI能量       -75.79831209
{ci;occ,4,1,2;closed,2;core,1;wf,9,1,1;save,7100.1}
                                      ! 总MRCI能量       -75.71309879
{ci;trans,7300.1,7100.1,ly}
                                      ! 跃迁矩<1.3|X|1.1> = -0.14659810 a.u.
                                      ! 跃迁矩<1.3|LY|1.1> = 0.96200488i a.u.
```

[http://www.molpro.net/info/current/examples/h2op\\_mrci\\_trans.com](http://www.molpro.net/info/current/examples/h2op_mrci_trans.com)

```
! $Revision: 2006.3 $
***,BH singlet Sigma and Delta states
r=2.1
geometry={b;h,b,r}
{hf;occ,3;wf,6,1;}
```

```
{multi;  
occ,3,1,1;frozen,1;wf,6,1;state,3;lquant,0,2,0;wf,6,4;lquant,2;  
tran,lz;  
expec2,lzlz;}  
! Sigma态 -- 能量 -25.20509620 -24.94085861  
{ci;occ,3,1,1;core,1;wf,6,1;state,2,1,3;}  
! Delta态 -- 能量 -24.98625171  
{ci;occ,3,1,1;core,1;wf,6,1;state,1,2;}  
! Delta态 -- xy分量  
{ci;occ,3,1,1;core,1;wf,6,4;}
```

[http://www.molpro.net/info/current/examples/bh\\_mrci-sigma\\_delta.com](http://www.molpro.net/info/current/examples/bh_mrci-sigma_delta.com)





## 第二十一章

# 多参考RAYLEIGH-SCHRÖDINGER微扰理论

参考文献:

原始的RS2/RS3:

H.-J. Werner, Mol. Phys. 89, 645-661 (1996)

新的内收缩RS2C:

P. Celani and H.-J. Werner, J. Chem. Phys. 112, 5546 (2000)

使用本程序的所有出版结果必须引用以上文献。

命令:

RS2,*options*

RS2C,*options*

RS3,*options*

用于执行二阶或三阶微扰计算。RS3总是包含RS2作为第一步计算。对于闭壳层单参考的情况，这等价于MP2或MP3（但是使用了不同的程序）。RS2C调用新的更高效的二阶程序（见下），如果不计算三阶项的话，一般使用这个（注意，不能使用RS3C）。

*Options*可以是:

Gn	使用修改的零阶哈密顿量，参见21.4节。
SHIFT= <i>value</i>	能级移动，参见21.5节。
MIX= <i>nstates</i>	调用 <i>nstates</i> 个态的多态（MS-CASPT2）处理。更多细节参见21.3节。
ROOT= <i>ioptrout</i>	在几何优化中优化的根编号。这和MS-CASPT2中的 <i>nstates</i> 有关。更多细节参见21.7节。

SAVEH=record	在MS-CASPT2计算中，保存有效哈密顿量的记录。如果没有给出，则使用默认的记录（推荐）。
INIT	（逻辑型）用单个态的参考函数初始化MS-CASPT2，参见21.3节。
IGNORE	（逻辑型）不使用CP-CASPT2的近似梯度计算标记；细节参见21.7节。

此外，还可以用MRCI的全部有效选项（参见第二十章）。

## 21.1 简介

多参考微扰计算作为MRCI程序的特例执行。对于RS2（CASPT2，RASPT2），只需要计算基于单电子算符的矩阵元，因此计算量远远小于相应的MRCI。对于RS3（CASPT3），必须计算一阶波函数的能量期待值，计算量几乎与一次MRCI迭代相同。RS2和RS3程序使用与MRCI相同的组态空间，即，只对双激发的外组态进行内收缩。

还包含新版本的程序，其中单激发外部和内部组态空间的子空间也都是内收缩的（参见上面给出的参考文献）。这个程序用RS2C关键词调用，比RS2效率更高，特别是对具有多个闭壳层（非活性）轨道的大分子。对于二阶多参考微扰理论（CASPT2，RASPT2）的一般应用，推荐使用这个程序。注意由于收缩的方案不同，它给出与RS2略微不同的结果。还要注意RS2和RS2C都不同于Roos等的CASPT2 [J. Chem. Phys. **96**, 1218 (1992)]，因为某些组态子空间没有收缩。不过，在一般情况下相差很小。需要提到的最后一点是CASPT2/RASPT2密度矩阵的计算（以及分子的特性），目前只能使用RS2命令而不能使用RS2C。

多参考微扰理论的结果可能对零阶哈密顿量的选择非常敏感。这种依赖性在二阶中比在三阶中更显著。可以使用几种选项，它们在以下章节讲述。在组态态函数的基中，还可能会发生 $(\hat{H}^{(0)} - E^{(0)})$ 变得（几乎）奇异。这就是众所周知的“入侵态问题”，将导致收敛问题，或者波函数变坏。这种问题通常可以用包含更多的轨道到参考波函数中来避免，不过这也会导致CPU时间增加。使用修改的Fock算符（见下）或者Roos和Andersson建议的能级移动[Chem. Phys. Lett. **245**, 215 (1995)]，可能也有帮助。目前，只能使用“实数”能级移动。

如果没有进一步的输入卡，波函数的定义（芯轨道，闭壳层及活性轨道空间和对称性）对应最近的SCF或MCSCF计算所用的定义。默认产生CASSCF参考空间。其它的选择可以使用OCC，CORE，CLOSED，WF，SELECT，CON和RESTRICT卡，说明见CI程序。轨道来自相应的SCF或MCSCF计算，除非给出了ORBITAL指令。

对于CASPT2计算，如果使用了（赝）正则轨道，零阶哈密顿量将会成为分块对角化的形式。这会加速收敛。因此在前面的MULTI计算中，推荐用CANONICAL 指令保存轨道（注意，默认是NATORB）。

MRCI计算的大多数选项（例如，STATE，REFSTATE等）也可用于RS2(C)和RS3，不在这里重复。专门用于CASPT2/3的特殊选项说明见下。

## 21.2 激发态计算

计算激发态有两种可能的方法：

1. 分别计算每个态。使用卡：

STATE,1,*root*

其中*root*是所需要的根（即，2对应第一激发态）。在这种情况下，用于零阶哈密顿量的Fock算符用给定态的密度进行计算。

2. 或者，同时计算两个或更多的态，使用

STATE, *n* [,*root1*, *root2*, ..., *rootn*]

其中的*n*是计算的态的个数。默认计算最低的*n*个根。默认设置也可以通过指定所需要的根*rooti*来修改，如上所示。应当注意，这并不对应于多态CASPT2，多态CASPT2的说明见21.3节。

对于同时计算几个态的情况，零阶哈密顿量中使用的Fock算符从态平均密度矩阵计算得到，所有态的零阶哈密顿量从同一Fock算符构造。默认对所有的态使用相同的权重。这一默认设置可以用WEIGHT指令修改：

WEIGHT,*w1*, *w2*, ..., *wn*.

如果给出了REFSTATE卡（见20.2.11节），将对所有的参考态产生态平均Fock算符，相应态的权重用WEIGHT卡定义。

## 21.3 多态CASPT2

执行多态CASPT2（MS-CASPT2）的描述见Finley等，CPL **288**, 299 (1998)。目前，这只能用于RS2程序（也就是说，不能用于RS2C）。执行MS-CASPT2有两种不同的模式：

(1) 进行混合的每个态单独计算，最后对所有的态进行混合。以后，这种计算称为SS-SR-CASPT2（单个态单参考的CASPT2）。对于每个CASPT2计算，有一个收缩的参考态，它是所关心的特定态。这是最便宜的方法，但是这种情况不能计算梯度。用户需要保证没有态被计算了两次。

(2) 所有的 $nstates$ 个态一起处理，有 $nstates$ 个收缩的参考态。这种计算更昂贵，但是由于不同的参考态可以在CASPT2中混合，会给出更平衡的描述。这种计算需要 $nstates$ 等于在STATE指令指定的态数。这种情况称为“MS-MR-CASPT”（多态多参考CASPT2），并且可以计算解析能量梯度，参见21.7节。

### 21.3.1 执行SS-SR-CASPT2计算

如果要把 $nstates$ 个CASPT2波函混合到一起，那么需要运行 $nstates$ 个单态单参考的CASPT2计算。

第一个计算必须用：

```
{RS2,MIX= $nstates$ , INIT,  $options$ 
STATE,1,1;}
```

随后的计算

```
{RS2,MIX= $nstates$ ,  $options$ 
STATE,1, $istate$ ;}}
```

用于 $istate = 2, \dots, nstates$ 。可以给定更多的 $options$ ，例如能级移动。

在每个计算的最后，保存CASPT2波函；在最后一个CASPT2计算之后，自动组合并打印Bloch 哈密顿量和相应的重叠矩阵。在对称化之后，按照Brandow IJQC 15, 207 (1979)的方法进行哈密顿量的对角化，以及简单的半加和（平均）。在每一步打印MS-CASPT2能量和混合因子。

变量MSENERGY(i) ( $i=1, \dots, nstates$ ) 设定为半加和对角化获得的多态能量。如果使用了能级移动，那么MSENERGY(i)包含Bloch哈密顿量的半加和对角化获得的多态能量，其中的对角元（CASPT2能量）已经用能级移动校正。

例子：SS-SR-CASPT2计算LiF

```
! $Revision: 2006.1 $
r=[3,4,5,6,7,8,9,10] ang

i=1
geometry={Li
          F,1,r(i)}

basis=vtz,F=avtz

hf                      ! Hartree-Fock

do i=1,#r                ! 对键长循环
{multi
closed,3,0,0,0
```

```

occ, 5,2,2,0
state,2          ! 两个态的SA-CASSCF
canonical,ci}

{rs2,MIX=2,INIT
state,1,1}      ! 参考态1的单个态CASPT2

e1_caspt2(i)=energy    ! 基态的非混合caspt2能量

{rs2,MIX=2
state,1,2}      ! 参考态2的单个态CASPT2

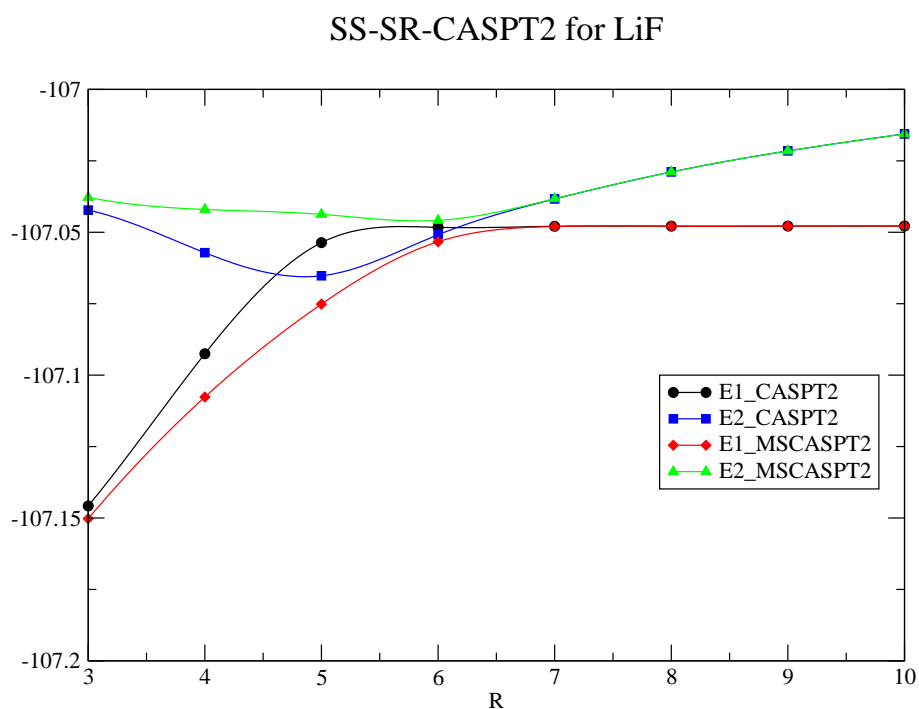
e2_caspt2(i)=energy    ! 激发态的非混合caspt2能量
e1_mscaspt2(i)=msenergy(1) ! 基态的ms-caspt2能量
e2_mscaspt2(i)=msenergy(2) ! 激发态的ms-caspt2能量
enddo

{table,r,e1_caspt2,e2_caspt2,e1_mscaspt2,e2_mscaspt2
title,SS-SR-CASPT2 for LiF
plot,file='lif_sr_mscaspt2.plot'
}

```

[http://www.molpro.net/info/current/examples/lif\\_sr\\_mscaspt2.com](http://www.molpro.net/info/current/examples/lif_sr_mscaspt2.com)

这将产生以下的图



### 21.3.2 执行MS-MR-CASPT2计算

对于多态多参考CASPT2计算，只需要一个运行：

```
{RS2,MIX=nstates, options
STATE,nstates}
```

例子：MS-MR-CASPT2计算LiF

```
! $Revision: 2006.1 $
r=[3,4,5,6,7,8,9,10] ang

i=1
geometry={Li
          F,1,r(i)}

basis=vtz,F=avtz

hf                      ! Hartree-Fock

do i=1,#r                ! 对键长循环
{multi
closed,3,0,0,0
occ,    5,2,2,0
state,2                  ! 两个态的SA-CASSCF
canonical,ci}

{rs2,MIX=2
state,2}                 ! 两个参考态的双态CASPT2

e1_caspt2(i)=energy(1)    ! 基态的非混合caspt2能量
e2_caspt2(i)=energy(2)    ! 激发态的非混合caspt2能量

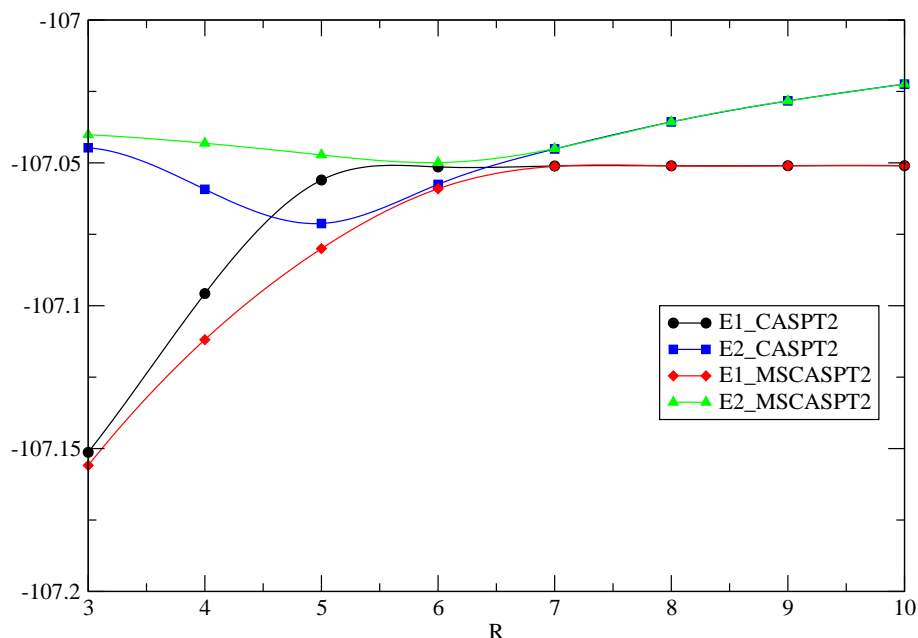
e1_mscaspt2(i)=msenergy(1) ! 基态的ms-caspt2能量
e2_mscaspt2(i)=msenergy(2) ! 激发态的ms-caspt2能量
enddo

{table,r,e1_caspt2,e2_caspt2,e1_mscaspt2,e2_mscaspt2
title,MS-MR-CASPT2 for LiF
plot,file='lif_mr_mscaspt2.plot'
}
```

[http://www.molpro.net/info/current/examples/lif\\_mr\\_mscaspt2.com](http://www.molpro.net/info/current/examples/lif_mr_mscaspt2.com)

这将产生以下的图

## MR-MR-CASPT2 for LiF



可以清楚地看到，这会给出比上一节SS-SR-CASPT2计算更平滑的势。此外，由于F的电子亲和的改善，避免交叉被移到了更长的键长。

## 21.4 零阶哈密顿量中的修正Fock算符

可以使用Andersson [Theor. Chim. Acta **91**, 31 (1995)]提议的 $g_1$ ,  $g_2$ 和 $g_3$ 算符，以及更进一步的 $g_4$ 算符。当分子离解到高自旋开壳层（RHF）原子的时候， $g_4$ 可以使CASPT2计算具有大小一致性。

算符的指数 $n$ 可以在RS2, RS2C或RS3卡定义：

RS2,*option*

RS2C,*option*

RS3,*option*

其中的*option*可以是G1, G2, G3, 或G4。该选项之后还可以接其它的有效选项。

## 21.5 能级移动

能级移动常用于避免激发态计算的入侵态问题。MOLPRO允许使用Roos和Andersson [Chem. Phys. Lett. **245**, 215 (1995)]描述的移动。可以对RS2或RS2C卡定义移动：

RS2 [,G*n*] [,SHIFT=*shift*]

RS2C [,G*n*] [,SHIFT=*shift*]

典型的移动是0.1 – 0.3。在小数点之后只考虑两位数字。移动会影响结果，打印的能量以及ENERGY 变量包含了Roos和Andersson建议的能级移动的能量修正。收敛后还打印未修正的能量用于比较。

## 21.6 积分直接计算

使用非常大基组的RS2, RS2C和RS3计算可以用积分直接模式执行。如果全局的DIRECT 或GDIRECT卡出现在输入较前的位置，计算将是直接的。或者（主要用于测试），作为RS $n$ [C] 卡的选项，定义DIRECT:

```
RS2 [,Gn] [,SHIFT=shift] [,DIRECT]
RS2C [,Gn] [,SHIFT=shift] [,DIRECT]
```

## 21.7 CASPT2梯度

P. Celani and H.-J. Werner, J. Chem. Phys. **119**, 5044 (2003)

如果之后接FORCE或OPTG命令（参见四十和四十一章），那么自动计算CASPT2解析能量梯度。解析梯度目前只能用于RS2计算（不能用于RS2C），且只能用于标准的 $\hat{H}^{(0)}$ （不能用于G1, G2, 等）。梯度可以对单个态进行计算，也可以对多态MS-MR-CASPT2进行计算（参见21.3节）。

在单态计算中，对在CASPT2/RASPT2中计算的态自动计算梯度（即，用STATE,1,2 计算所考虑对称性的第二个态，参见21.2节）。程序也能用态平均MCSCF（或CASSCF）轨道，不需要CPMCSCF指令。被考虑的态需要包含在前面的态平均MCSCF/CASSCF计算中。通过用NOEXC指令，RS2梯度程序也可以用来计算态平均MCSCF/CASSCF梯度。

在多态MS-MR-CASPT2计算中，需要计算梯度的态必须用ROOT 选项指定（默认ROOT=1），即，

```
RS2,MIX=nstates, ROOT=ioptroot
```

其中  $1 \leq ioptroot \leq nstates$ 。

还可以使用能级移动。默认计算能级移动校正后的能量的精确梯度。对于非零的移动，这要求解CASPT2的Z 矢量方程，使得计算量几乎增加一倍。在单个态的计算中，可以忽略能级移动对梯度的影响，因此不用求解Z矢量方程。这种变体在以上的论文中有说明，对于大多数计算足够精确。它用IGNORE选项调用，例如，

```
RS2,SHIFT=0.2,IGNORE
OPTG
```



所有执行CASPT2梯度的出版物应当引用以上论文。MS-CASPT2梯度方法的引用为P. Celani and H.-J. Werner, *to be published*。

例子：

H<sub>2</sub>O的CASPT2结构优化：

```
!$Revision: 2006.1 $
***
memory,8,m
gthresh,energy=1.d-10
!
basis=vdz
R=2.0
R0=R
Theta=100
geometry={0
          H1,0,R;
          H2,0,R,H1,THETA}

hf;accu,12

{multi;closed,2}

rs2,shift=0.3,ignoreshift      ! 在梯度计算中忽略移动，即，不进行cp-caspt2计算
optg,gradient=1.d-5
e_opt(1)=energy
r_opt(1)=r
theta_opt(1)=theta
method(1)='rs2,analytical,ignore'

rs2,shift=0.3                  ! 包含移动的精确梯度
optg,gradient=1.d-5
e_opt(2)=energy
r_opt(2)=r
theta_opt(2)=theta
method(2)='rs2,analytical,exact'

rs2,shift=0.3                  ! 包含移动的数值梯度
optg,gradient=1.d-5,numerical,fourpoint ! 使用四个点的数值梯度
e_opt(3)=energy
r_opt(3)=r
theta_opt(3)=theta
method(3)='rs2,numerical'

rs2c,shift=0.3                 ! 包含移动的rs2c数值梯度
```

```

optg,gradient=1.d-5,fourpoint  ! 使用四个点的数值梯度
e_opt(4)=energy
r_opt(4)=r
theta_opt(4)=theta
method(4)='rs2c,numerical'

table,method,r_opt,theta_opt,e_opt
digits,,4,4,8

```

[http://www.molpro.net/info/current/examples/h2o\\_caspt2\\_opt.com](http://www.molpro.net/info/current/examples/h2o_caspt2_opt.com)

这将产生表格:

METHOD	R_OPT	THETA_OPT	E_OPT
rs2,analytical,ignore	1.8250	102.1069	-76.22789382
rs2,analytical,exact	1.8261	102.1168	-76.22789441
rs2,numerical	1.8261	102.1168	-76.22789441
rs2c,numerical	1.8260	102.1187	-76.22787681

对H<sub>2</sub>O的第二个激发态<sup>3</sup>B<sub>2</sub>, 用MS-CASPT2做结构优化:

```

!$Revision: 2006.1 $
***
memory,8,m
gthresh,energy=1.d-12
!
basis=vdz
R=2.0
R0=R
Theta=100
step=0.001
geometry={0
          H1,0,R;
          H2,0,R,H1,THETA}

hf;accu,12

multi      ! 各种三重态的态平均casscf
closed,2
wf,10,1,2
state,3
wf,10,2,2
state,2
wf,10,3,2
state,3
canonical,2140.2

```

```

rs2,mix=3,root=2,shift=0.2    ! 优化第二个3B2态
wf,10,3,2                    ! 3B2波函的对称性
state,3                       ! 包含3个态
optg,gradient=1.d-5           ! 使用解析梯度进行几何优化

e_opt(1)=msenergy(2)          ! 优化的ms-caspt2能量
r_opt(1)=r                     ! 优化的键长
theta_opt(1)=theta            ! 优化的键角
method(1)='rs2,analytical'

rs2,mix=3,shift=0.2
wf,10,3,2                    ! 3B2波函对称性
state,3                       ! 包含3个态
optg,variable=msenergy(2),gradient=1.d-5,fourpoint
                                ! 使用数值梯度进行几何优化

e_opt(2)=msenergy(2)          ! 优化的ms-caspt2能量
r_opt(2)=r                     ! 优化的键长
theta_opt(2)=theta            ! 优化的键角
method(2)='rs2,numerical'

table,method,r_opt,theta_opt,e_opt
digits,,4,4,8

```

[http://www.molpro.net/info/current/examples/h2o\\_mscaspt2\\_opt.com](http://www.molpro.net/info/current/examples/h2o_mscaspt2_opt.com)

这将产生表格:

METHOD	R_OPT	THETA_OPT	E_OPT
rs2,analytical	2.4259	96.7213	-75.81630628
rs2,numerical	2.4259	96.7213	-75.81630628

## 21.8 MRCI与MRPT2耦合: CIPT2方法

P. Celani, H. Stoll, and H.-J. Werner, Mol. Phys. **102**, 2369 (2004).

对于有强入侵态问题的特别困难的情况, 或者二阶微扰理论未能获得可靠的结果的情况, 还开发了MRCI与CASPT2 耦合的新方法。这个变体用CIPT2指令调用:

CIPT2

对于这种情况, 所有从活性轨道的激发用MRCI处理, 其余涉及非活性(闭壳层)轨道的激发用二阶微扰理论处理。两种方法通过合适的能量泛函的最小化来耦合。当然, 这种方法比MRPT2昂贵得多。耗时相当于没有非活性轨道关联的MRCI。

## 21.9 CASPT2和CASPT3的更多选项

其它选项可以用OPTION命令设置。这些选项主要用于测试目的，所以使用时要小心。需要注意，在RS2C程序中只有IFDIA选项可以修改；所有其它选项只能用于RS2/RS3。

OPTION,code1=value,code2=value,...

CASPT2/3程序有关的选项如下：

- |                   |   |
|-------------------|---|
| IPROCS=0          | （默认。）RS2计算使用非收缩的单激发。  |
| IPROCS=1          | 在更新时，把非相互作用的单激发投影掉。这是一个近似的流程，使用时要当心。  |
| IPROCS=2          | RS2中的单激发是完全内收缩的。在因子更新的过程中，这可以通过投影算符获得，可能效率很差。   |
| IPROCS=3          | 在闭壳层中，只有一个或两个空穴的单激发在RS2中用投影算符进行内收缩。   |
| IPROCI=0          | （默认。）RS2计算使用非收缩的内空间。  |
| IPROCI=1          | 在非活性空间内，有两个空穴的内空间在RS2中用投影算符进行内收缩。   |
| IPROCS=3,IPROCI=1 | 这个选项组合使用投影算符，用RS2重现RS2C的结果。这需要大量的内存和磁盘空间，只能用于小分子。   |
| IFDIA=0           | （默认。）包含有效Fock矩阵的所有非对角元。   |
| IFDIA=1           | 忽略Fock矩阵的内-外区域。这会消除单激发-电子对的耦合。  |
| IFDIA=2           | 忽略Fock矩阵的所有非对角元。这对应Andersson等人的CASPT2D方法。注意：在这种情况下，对活性轨道转动的结果不再是不变量。                                      |
| IHINT=0           | （默认。）在零阶哈密顿量中，对所有的相互作用只使用单电子积分。   |
| IHINT=1           | 在零阶哈密顿量中，对所有的内部-内部和单激发-单激发相互作用，使用全内部双电子积分。  |
| IHINT=2           | 在零阶哈密顿量中，对内部-内部，单激发-单激发和电子对-电子对相互作用，使用全内部双电子积分。使用IHINT=2和IDFIA=1对应于Dyall的CAS/A方法，其中使用了没有闭壳层（非活性轨道）的CASSCF |

参考。注意，这比标准CASPT2计算需要更多的CPU时间。此外，CAS/A方法的收敛通常很慢（在SHIFT卡指定的分母移动有助于这种情况的收敛）。一般来说，我们不推荐IHINT使用非零值。

NOREF=1      （默认。）忽略参考组态和单激发之间的相互作用。

NOREF=0      包含参考组态和单激发之间的相互作用。这会放松参考因子，但也可能导致入侵态问题。

IMP3=2      CASPT2之后，用所有态全部的内组态和一阶波函作为基，执行变分CI。这种情况的二阶能量对应于变分能量，三阶能量近似为Davidson校正能量。这对近简并情况的激发态计算非常有用。



参考文献:

- C. Angeli, R. Cimiraglia, S. Evangelisti, T. Leininger and J. P. Malrieu, J. Chem. Phys., 114, 10252, (2001)
- C. Angeli, R. Cimiraglia and J. P. Malrieu, J. Chem. Phys., 117, 9138, (2002)
- C. Angeli, M. Pastore and R. Cimiraglia, Theor. Chem. Acc., 117, 743 (2007)

使用本程序的所有出版结果必须引用以上文献。

### 22.1 总的考虑

NEVPT2是二阶多参考微扰理论的一种形式，它可用于CAS-SCF波函或者更一般的CAS-CI波函。术语NEVPT是“*n*电子价态微扰理论 (*n-electron valence state perturbation theory*)”的缩写。在我们向读者提到相关文献时（见上），让我们特别回顾一下NEVPT2的最重要功能，包括：波函的一阶校正是对一组正确选择的多参考函数进行的展开，其中正确考虑了发生在活性电子之间的双电子相互作用。NEVPT2的特点有：

- 严格可分（大小一致性）：非相互作用体系的总能量等于孤立体系的能量和。
- 没有入侵态：关于外部空间函数的零阶能量与所研究态的零阶能量能够很好地分开，因此避免了微扰求和的不收敛。

- 波函的一阶校正是自旋算符 $S^2$ 和 $S_z$ 的本征函数。
- 电子激发态用与基态相同精度的级别进行处理。
- 在活性轨道的酉变换下，NEVPT2能量不变。此外，对芯轨道和虚轨道选择正则轨道（这是默认选择），保证了结果与理论扩充后（在芯轨道和虚轨道中分别转动应是完全不变的）的一致性。
- 对于HF波函，NEVPT2与MP2一致。

MOLPRO中的NEVPT2执行两种变体，即强收缩（SC）变体和部分收缩（PC）变体。两种变体在微扰求和中使用的微扰函数个数有所不同。PC-NEVPT2使用更多的函数空间，一般比SC-NEVPT2更准确。SC-NEVPT2 和PC-NEVPT2的结果通常总是非常接近。

## 22.2 输入说明

NEVPT2必须接在CAS-SCF或CAS-CI计算之后。执行二阶微扰理论计算，需要指定命令

NEVPT2,*options*

NEVPT2是MRCI程序的一部分，使用后者的选项。特别有关的有MRCI程序的CORE, CLOSED, OCC, WF, 和STATE选项。目前，只有一个针对NEVPT2的选项需要由用户提供：

THRNEVPT2      去除CAS波函中小因子的阈值（默认= 0.0）。

NEVPT2目前的执行是对特定态的，也即微扰理论只能用于一个态。在不远的将来，MOLPRO中还会执行多态（或准简并）版的NEVPT2。

这里提供了一个例子，其中计算了甲醛基态和第一激发态 $^1A_2$  ( $n \rightarrow \pi^*$ ) 的能量。

```
! $Revision: 2008.3 $
***,
memory,20,m
file,1,h2co.int
file,2,h2co.wf
gthresh,energy=1.d-9
gthresh,orbital=1.d-8
gthresh,civec=1.d-8

geomtyp=zmata
```



```
geometry
O,,      0.000000000,    0.000000000,    0.0196594609
C,,      0.000000000,    0.000000000,    2.3248507925
H1,,     0.000000000,    1.7597110083,    3.3972521023
H2,,     0.000000000,   -1.7597110083,    3.3972521023
end

basis=6-31G*

{hf
wf,16,1,0}

{multi
closed,4,0,1,0
occ,6,2,4,0
wf,16,1,0
state,1
natorb,2140.2,state=1.1
}

{nevpt2,thrden=1.0d-10,thrvar=1.0d-10
core,2,0,0,0
closed,4,0,1,0
occ,6,2,4,0
orbit,2140.2,state=1.1
wf,16,1,0
state,1,1
}

{multi
closed,4,0,1,0
occ,6,2,4,0
wf,16,4,0
state,1
start,2140.2
natorb,2141.2,state=1.4
}

{nevpt2,thrden=1.0d-10,thrvar=1.0d-10
core,2,0,0,0
closed,4,0,1,0
occ,6,2,4,0
orbit,2141.2,state=1.4
wf,16,4,0
state,1,1
```

```
}
```

[http://www.molpro.net/info/current/examples/form\\_nevpt2.com](http://www.molpro.net/info/current/examples/form_nevpt2.com)

## 第二十三章

# MØLLER-PLESSET微扰理论

直到完全四阶[MP4(SDTQ)]的闭壳层Møller-Plesset微扰理论是耦合簇程序的一部分。

MP2, MP3, MP4命令执行到指定级别的MP计算（高阶包含低阶）。

MP4;NOTRIPL;执行MP4(SDQ)计算。

如果MPn卡直接在相应的HF-SCF计算之后，通常不需要更多的输入。否则，需要指定占据和轨道，就像在CI程序中一样。计算的能量存储于变量中，说明参见8.8节。

可以对闭壳层方法执行双基组计算，参见24.9节。

### 23.1 MP2的期待值

单电子特性可以作为MP2的解析能量导数进行计算。这个计算比简单的MP2计算要昂贵得多，因此只有在MP2卡之后接EXPEC卡时才执行（GEXPEC指令在这种情况下不起作用）。EXPEC卡的语法在6.13节有解释。例子参见24.6节。保存密度矩阵使用

DM,[*record.ifil*];

另外参见24.7和24.8节。

要改变CPHF计算的精度及其它参数，参见23.3节。

### 23.2 MP2的极化率和二阶特性

MP2的解析静态偶极极化率以及其它的二阶特性可以用POLARI计算。默认使用偶极算符，但是可以在POLARI指令中指定其它的单电子算符。目前还不能使用冻芯轨道，即，必须给出CORE,0。偶极极化率存储于变量POLXX, POLXY, POLXZ, POLYY, POLYZ, POLZZ。

例子:

```
{ mp2
  core,0
  polari,dm,qm}
```

将对偶极和四极算符计算完全张量。

### 23.3 梯度，期待值和极化率的CPHF

梯度和响应特性计算所需的CPHF精度和其它参数，可以用CPHF指令修改：

CPHF, [THRMIN=*thrmin*] , [THRMAX=*thrmax*] , [MAXIT=*maxit*] , [SHIFT=*shift*] , [SAVE=*record*] ,  
[START=*record*] , [DIIS=*idiis*] , [DISM=*idism*]

THRMIN	CPHF收敛阈值（默认1.d-6）。别名为THRESH和ACCU。
THRMAX	在结构优化中，CPHF的初始收敛阈值。一旦结构收敛到某个精度（依赖于OPTCONV），那么阈值就逐步减小到THRMIN。 默认为THRMAX=min(1.d-6,THRMIN*100)，大于1.d-6的值会被忽略。
MAXIT	最大数量的迭代（默认50）。
SHIFT	CPHF的能级移动（默认0.1）。
DIIS	使用DIIS的第一次宏迭代（默认1）。
DISM	使用DIIS的第一次微迭代（默认1；在微迭代中，芯的贡献被冻结）。
SAVE	保存哪个CPHF解的记录用于续算。在结构优化和频率计算中自动保存解。
START	从哪个记录读取初始猜测。在结构优化和频率计算中自动读取初始猜测。

### 23.4 密度拟合MP2（DF-MP2，RI-MP2）

DF-MP2,*options*

调用密度拟合MP2程序。目前执行计算还不能使用对称性。RI-MP2是命令DF-MP2的别名。

可以指定以下的选项：

BASIS_MP2= <i>basis</i>	拟合基组。 <i>basis</i> 既可以是BASIS区定义的基组，也可以是默认的拟合基组（仅能用于关联一致基组）。如果使用了关联一致轨道基组，那么默认产生相应的MP2拟合基组。对于所有的其它情况，必须定义拟合基组。
THRAO= <i>value</i>	在AO基中三指数积分的屏蔽阈值。
THRMO= <i>value</i>	在MO基中三指数积分的屏蔽阈值。
THROV= <i>value</i>	拟合基二指数积分的屏蔽阈值。
THRPROD= <i>value</i>	第一次半转换的屏蔽乘积阈值。
SPARSE= <i>value</i>	如果非零，在第二个半转换中使用稀疏算法（默认）。

关于密度拟合的更多一般说明，见第十五章。

目前，期待值和梯度还不能用于DF-MP2，但可以用于局域变体DF-LMP2。

### 23.5 自旋分量换算MP2 (SCS-MP2)

自旋分量换算MP2能量由Grimme建议[J. Chem. Phys. **118**, 9095 (2003)]，用默认的换算因子（反平行自旋1.2，平行自旋1/3）自动打印。这些因子可以分别用选项SCSFACS和SCSFACT 修改，即，

MP2, SCSFACS=*facs*, SCSFACT=*fact*

SCS-MP2总能量存储在变量EMP2\_SCS中。通过设定选项SCSGRD=1，可以计算SCS-MP2的梯度。这个选项只能用于密度拟合MP2，即，使用

DF-MP2,[DF\_BASIS=*fitbasis*],SCSGRD=1,[ SCSFACS=*facs*], [SCSFACT=*fact*]

之后接上FORCES或OPTG。对于后一种情况，用SCS-MP2能量优化结构。



## 第二十四章

## 闭壳层CCSD程序

参考文献:

C. Hampel, K. Peterson, and H.-J. Werner, Chem. Phys. Lett. 190, 1 (1992)

使用本程序的所有出版结果必须引用以上文献。

CCSD程序用CISD, CCSD, BCCD, 或QCI指令调用。作为特例, 可以用NOSINGL指令实现CID或CCD。代码还允许计算Brueckner轨道 (这种情况下QCI与CCSD相同)。如果CCSD卡接在相应的HF-SCF 之后, 通常不需要进一步的输入。可选的选项有ORBITAL, OCC, CLOSED, CORE, SAVE, START, 和PRINT, 与在二十章MRCI程序中的说明相同。这个代码的特殊输入指令是BRUECKNER和DIIS, 将在后面说明。

可以在命令行指定以下选项:

NOCHECK	忽略收敛检查。
DIRECT	进行积分直接计算。
NOSING	不包含单激发外部组态。
MAXIT= <i>value</i>	最大迭代数。
SHIFTS= <i>value</i>	升级单激发的分母移动。
SHIFTP= <i>value</i>	升级双激发的分母移动。
THRDEN= <i>value</i>	能量收敛阈值。
THRVAR= <i>value</i>	CC振幅收敛阈值。这用于振幅变化的平方和。

可以修改收敛阈值, 用:

THRESH,ENERGY=*thrden*,COEFF=*thrvar*

如果能量变化小于*thrden* (默认1.d-6), 且振幅变化平方和小于*thrvar* (默认1.d-

10), 则认为达到收敛。THRESH卡必须接在方法的命令(例如CCSD)之后, 并覆盖相应的全局选项(参见6.11节的GTHRESH)。

计算的能量存储在变量中, 在8.8节有说明。除了能量, 还打印 $T_1$ 特征信息(T. J. Lee and P. R. Taylor, Int. J. Quant. Chem. S23 (1989) 199)和 $D_1$ 特征信息(C. L. Janssen and I. M. B. Nielsen, Chem. Phys. Lett. 290 (1998), 423和T. J. Lee. Chem. Phys. Lett. 372 (2003), 362), 分别存储于变量T1DIAG和D1DIAG中, 用于后面的分析。

## 24.1 耦合簇, CCSD

CCSD命令执行闭壳层耦合簇计算。使用CCSD(T)命令, 还计算有关的三激发微扰贡献。

如果CCSD不收敛, 且计算微扰三激发的话, 就会出错退出。这可以用NOCHECK选项避免:

CCSD(T), NOCHECK

这种情况下, 即使CCSD不收敛也计算(T)校正。注意: NOCHECK对结构优化和频率计算无效。

关于三阶校正的更多信息, 见下面的RCCSD。

## 24.2 二次组态相互作用, QCI

QCI或QCISD指令执行二次组态相互作用QCISD。使用QCI(T)或QCISD(T)命令, 还用微扰理论计算有关的三激发的贡献。如果QCI卡接在相应的HF-SCF之后, 通常不需要进一步的输入。否则需要像在CI程序中那样定义占据和轨道。要修改DIIS指令, 见24.5节。对于QCISD 和QCISD(T)的一阶特性, 见24.7节。

要避免不收敛造成的出错退出, 参见CCSD(T)。

## 24.3 Brueckner耦合簇计算, BCCD

BCCD,[SAVE=*record*],[PRINT],[TYPE=,*type*]

BCCD执行Brueckner耦合簇计算, 并计算Brueckner轨道。使用这些轨道, 单激发振幅在收敛时消失。使用BCCD(T)命令, 还用微扰理论计算有关的三激发的贡献。如果BCCD卡接在相应的HF-SCF之后, 通常不需要进一步的输入。否则需要像CI程序中那样定义占据和轨道。BRUECKNER参数可以用BRUECKNER 指令修改。

Brueckner轨道和近似的密度矩阵可以用SAVE选项保存在MOLPRO转存记录中。如果给出PRINT选项, 还打印轨道。TYPE可以用来定义计算的近似密度类型:



TYPE=REF	仅计算和存储参考行列式的密度（默认）。这对应于 Chem. Phys. Lett. 315, 248 (1999)中的BOX（Brueckner轨道期待值）方法。
TYPE=TOT	计算和存储包含电子对振幅（线性项）贡献的密度。通常这不会有什么改进。
TYPE=ALL	计算和存储两种密度。

注意：期待值照常存储于变量中。对于两种密度都计算的情况，变量包含两个值，第一个对应REF，第二个对应TOT（例如DMZ(1)和DMZ(2)）。如果给定了TYPE=REF或TYPE=TOT，则只存储相应的值。

对于避免不收敛造成的出错退出，参见CCSD(T)。

### 24.3.1 BRUECKNER指令

BRUECKNER, *orbbrk*, *ibrstr*, *ibrueck*, *brsfak*;

这一指令允许修改Brueckner计算的选项。通常不需要定义这些选项，只需BCCD命令就可用于执行Brueckner CCD计算。

<i>orbbrk</i>	如果非零，Brueckner轨道保存于这个记录中。
<i>ibrstr</i>	第一个修改轨道的迭代（默认=3）。
<i>ibrueck</i>	轨道升级的迭代增量（默认=1）。
<i>brsfak</i>	轨道升级中单激发的换算因子（默认=1）。

## 24.4 单-双组态相互作用，CISD

执行闭壳层组态相互作用CISD。这与CI程序获得的结果相同，但是这个代码略快。通常不需要进一步的输入。要指定DIIS指令，见24.5节。

## 24.5 DIIS指令

DIIS, *itedis*, *incdis*, *maxdis*, *itydis*;

这一指令可以修改CCSD，QCISD或BCCD计算的DIIS参数。

<i>itedis</i>	执行DIIS外推的第一次迭代（默认=2）。
<i>incdis</i>	DIIS迭代之间的增量（默认=1）。
<i>maxdis</i>	使用展开矢量的最大数量（默认=6）。

*itydis* DIIS外推类型。*itydis*=1 (默认): 残差最小。*itydis*=2:  $\Delta T$ 最小。

此外, 还可以用THRESH指令修改阈值 *THRESH*。DIIS外推仅用于变化小于 *THRDIS* 的时候。

## 24.6 示例

### 24.6.1 H<sub>2</sub>O的单参考关联处理

```
! $Revision: 2006.0 $
***,h2o test
memory,1,m                ! 分配1 MW动态内存
geometry={o;h1,o,r;h2,o,r,h1,theta} ! Z矩阵结构输入
basis=vtz                  ! cc-pVTZ基组
r=1 ang                    ! 键长
theta=104                  ! 键角
hf                          ! 进行scf计算

text,examples for single-reference correlation treatments

ci                          ! 用MRCI代码计算CISD
cepa(1)                    ! 用MRCI代码计算cepa-1
mp2                         ! 二阶Moeller-Plesset
mp3                         ! 二阶和三阶MP
mp4                         ! 二阶、二阶和四阶MP4(SDTQ)
mp4;notripl                ! MP4(SDQ)
cisd                        ! 用特殊的闭壳层代码计算CISD
ccsd(t)                    ! 耦合簇CCSD(T)
qci(t)                     ! 二次组态相互作用QCISD(T)
bccd(t)                    ! Brueckner CCD(T)计算
---
```

[http://www.molpro.net/info/current/examples/h2o\\_ccsd.com](http://www.molpro.net/info/current/examples/h2o_ccsd.com)

### 24.6.2 N<sub>2</sub>F<sub>2</sub>的单参考关联处理

```
! $Revision: 2006.0 $
***,N2F2 CIS GEOMETRY (C2h)
rnn=1.223,ang              ! 定义N-N距离
rnf=1.398,ang              ! 定义N-F距离
alpha=114.5;               ! 定义FNN键角
geometry={N1
              N2,N1,rnn
```

```

      F1,N1,rnf,N2,alpha
      F2,N2,rnf,N1,alpha,F1,180}
basis=vtz                      ! cc-pVTZ 基组
$method=[hf,cisd,ccsd(t),qcisd(t),bccd(t)] ! 所有使用的方法
do i=1,#method                  ! 对方法循环
$method(i)                     ! 对给定的方法进行计算
e(i)=energy                    ! 把能量保存到变量e
enddo                          ! 结束方法的循环
table,method,e                 ! 打印结果的表格
title,Results for n2f2, basis=$basis ! 表格标题

```

[http://www.molpro.net/info/current/examples/n2f2\\_ccsd.com](http://www.molpro.net/info/current/examples/n2f2_ccsd.com)

计算产生以下的表格:

Results for n2f2, basis=VTZ

METHOD	E	E-ESCF
CISD	-308.4634948	-0.78283137
BCCD(T)	-308.6251173	-0.94445391
CCSD(T)	-308.6257931	-0.94512967
QCISD(T)	-308.6274755	-0.94681207

## 24.7 保存密度矩阵

DM,[*record.ifil*];

计算有效一阶密度矩阵，并存储于文件*ifil*的*record*记录中。当前可用于闭壳层的MP2，QCISD和QCISD(T)。参见NATORB。注意，这比单点能计算更昂贵，因为需要求解响应方程。注意：没有轨道松弛贡献的CCSD一阶特性计算可以用：

ccsd

expec,dm

包含轨道松弛的计算可以用

ccsd

core,0

expec,relax,dm

目前这只能用于不加芯的情况，即，要求CORE,0。

## 24.8 自然轨道

NATORB,[RECORD=]*record.ifil*,[PRINT=*nprint*],[CORE[=*natcor*]];

计算自然轨道。当前只能用于闭壳层的MP2和QCISD。在任何给定对称性中，要打印的外部轨道数是`nprint`（默认是2）。`nprint=-1`禁止打印。自然轨道和密度矩阵保存于文件`ifil`的`record`转存记录中。如果在DM卡定义了`record.ifil`（见上），则使用该记录。如果在DM和NATORB卡中定义了不同的记录，将会导致错误。记录也可以在SAVE卡中给出。注意，非变分方法如MP2或QCISD，它们的有效密度矩阵不是严格的密度矩阵。例如，在芯和价轨道之间具有非零的矩阵元，因此芯轨道会受自然轨道变换的影响。另外，芯轨道的占据数可以大于2.0。如果给定了CORE（`natcor=1`），可以通过把它们从自然轨道变换中排除来冻结芯轨道。

## 24.9 双基组计算

双基组计算可用于闭壳层的MP2和CCSD代码（常规的和局域的，可能的话还可以用密度拟合）。通常这意味着用比关联计算小的基组完成了Hartree-Fock计算。在MOLPRO中有两种方法：推荐的是用相同基组执行HF和MP2或CCSD计算，仅仅在HF计算中忽略角动量较高的函数。这意味着产生的HF轨道可直接用于关联计算。另一种方法是在HF和关联计算中使用完全不同的基组；这种情况下，关联计算中的轨道通过对HF轨道的最小二乘拟合来确定。这种方法效率较低（特别是在完全直接计算中），精度较差。无论哪种方法，在MP2/CCSD程序中都计算新的Fock矩阵，并在占据轨道和虚轨道子空间分块对角化。为了减少HF基组误差，对MP2使用了微扰单激发校正。

典型的输入如下：

```
basis=vtz(d/p)  ! 三zeta基组，对重原子没有f函数，对氢原子没有d函数
hf              ! 用小基组计算Hartree-Fock
basis=vtz       ! ccscd使用完整的cc-pVTZ基组
ccsd(t),dual=1  ! ccscd计算
```

选项`dual=1`是必须的，否则程序会停止，并显示出错信息为：参考轨道的基组不一致。这一预防措施是为了避免不希望的结果。

类似地，这也可以用于其它的闭壳层单参考方法，如MP2，QCISD，MP2-F12，CCSD-F12，以及局域变体LMP2，LQCISD，LCCSD，既可以是常规模式也可以是直接模式。此外，还可以做DF-LMP2，DF-LCCSD，DF-LMP2-F12双基组计算。

## 第二十五章

# 使用运动方程CCSD (EOM-CCSD) 的激发态

单重态的激发能可以用运动方程 (EOM) 方法计算。EOM-CCSD方法对激发能可以给出与线性响应CCSD (LR-CCSD) 理论同样的结果。只能对单电子激发态得到准确结果。计算的态在EOM输入卡中指定，它是CCSD的子命令。可以用以下的输入形式：

EOM, *state1*, *state2*, *state3*, ...

计算给定的态。每个态用*number.sym*的形式给出，例如5.3表示对称性3的第5个态。注意态1.1对应于CCSD 波函的基态，如果给出会被忽略。

EOM, -*n1.sym1*, -*n2, sym2*, ...

计算对称性*sym1*的前*n1*个态，对称性*sym2*的前*n2*个态，等。

EOM, *n1.sym1*, -*n2, sym1*, ...

计算对称性*sym1*的第*n1*到第*n2*之间的态。

不同形式之间可以组合，例如，

EOM, -3.1, 2.2, 2.3, -5.3

计算对称性1的第1-3个态，对称性2的第2个激发态，对称性3的第2到第5个激发态。注意态1.1是基态的CCSD波函。

在CCSD不收敛的情况下，如果还指定后面的EOM计算，默认出错退出。出错退出可以用CCSD命令中的NOCHECK选项避免（另外参见CCSD(T)）。

### 25.1 EOM的选项

计算激发能通常不需要进一步的输入。

EOM-CCSD的振幅可以用SAVE=*record.ifil*保存。在每一次迭代空间更新后以及在计算的最后，将保存矢量。如果指定了START=*record.ifil*，计算可以从保存的矢量

续算。在旧的计算和续算中，要计算的矢量集可以不同。不过，如果同时指定了（SAVE和START）两个卡，并且保存记录和续算记录相同，那么矢量集也要相同，否则会有噪声。相同的SAVE和START记录对于势能面计算非常有用，见25.4.1节。

默认只计算激发能，因为特性的计算几乎要花能量计算两倍的时间。单电子特性和跃迁矩（期待值型，定义见J.F. Stanton and R.J. Bartlett, J. Chem. Phys., **98** 7029 (1993)）可以在卡EOM卡中加入TRANS=1进行计算。CCSD基态作为特殊情况进行处理。如果在EXPEC卡指定了RELAX 选项，那么还对基态计算松弛单电子密度矩阵。（目前，松弛CCSD密度矩阵只能用于全电子计算。）默认计算偶极矩。其它需要计算的特性可以用EXPEC卡指定。如果出现了DM卡，还保存激发态密度。例子见25.4.2节。如果计算了特性，那么把它们保存到MOLPRO变量中，例如，偶极矩的 $x$  分量保存在DMX，它的纯电子部分保存在DMXE，跃迁矩保存在TRDMX（左、右跃迁矩分别保存）。如果TRANS=2，还计算激发态之间的跃迁距。如果指定了DENSARE=record.ifil，那么把激发态密度（如果TRANS=2，还有跃迁密度）保存到record.ifil，否则保存到DM 卡给定的记录中。

当需要计算特性时，首先计算左EOM-CCSD波函。可以用它们作为右EOM-CCSD波函的初始猜测。这一选项用STARTLE控制（默认0）。如果STARTLE=1，仅用左矢作为右矢的起始点；如果STARTLE=2，从左矢获得的起始矢量再对左矢做二次正交化；如果STARTLE=3，还用最终的右矢对左矢做二次正交化。最后一种选择对简并态非常重要。

可以让程序收敛到一个矢量，类似于一个指定的单激发矢量。这一选项可以用FOLLOW= $n$ 卡打开（通常应设定为 $n=2$ ）。FOLLOW卡应当总是与EXFILE=record.ifil卡一起使用，其中record.ifil包含了前一计算的单激发矢量，参见25.4.3节。

## 25.2 EOMPAR卡的选项

通常不需要更多输入。但是，有些默认值可以用EOMPAR指令修改：

EOMPAR, key1=value1, key2=value2,...

其中，可以使用以下这些关键词key:

MAXDAV= $nv$	在Davidson流程中，每个态的展开矢量的最大值（默认20）。
INISINGL= $ns$	包含在初始哈密顿量中的单激发组态数量（默认20；组态按照能量排序）。有时为了获得双激发为主的态，INISINGL需要设置为0。
INIDOUBL= $nd$	包含在初始哈密顿量中的双激发组态数量（默认10）。
INIMAX= $nmax$	包含在初始哈密顿量中的激发组态最大数量。默认 $nmax = ns + nd$ 。

MAXITER= <i>itmax</i>	EOM-CCSD迭代的最大数量 (默认50)。
MAXEXTRA= <i>maxex</i>	包含在初始哈密顿量中的额外组态最大数量 (默认0)。对近简并的情况, 为了避免收敛慢, 最好包含一些额外的组态。
EOMLOCAL= <i>eoml</i>	如果设为0, 进行非局域计算 (默认)。EOMLOCAL=1打开局域模快 (实验性的! )。
INIMAX= <i>ini</i>	包含在初始哈密顿量中的CSF数量, 仅用于INISINGL和INIDOUBL都为零的情况。

所有的关键词都可以缩写为至少前四个字符。

### 25.3 EOMPRINT卡的选项

下面的打印选项主要用于测试目的, 以及寻找收敛问题。

EOMPRINT, key1=*value1*, key2=*value2*,...

其中, key可以使用以下的关键词:

DAVIDSON= <i>ipr</i>	Davidson流程的信息: <i>ipr=1</i> 打印每个“小对角化”的结果 <i>ipr=2</i> 还打印复本征值的警告信息 <i>ipr=3</i> 还打印测试空间的哈密顿量和重叠矩阵
DIAGONAL= <i>ipr</i>	组态的信息: <i>ipr=1</i> 打印转换后哈密顿量的最低近似对角元 <i>ipr=2</i> 打印重要组态的轨道标记 <i>ipr=3</i> 打印所有的近似对角元 <i>ipr=4</i> 还打印以上选项的详细形式
PSPACE= <i>ipr</i>	打印初始近似哈密顿量的信息: <i>ipr=2</i> 打印用于寻找一阶近似的近似哈密顿量
HEFF= <i>ipr</i>	打印有效哈密顿量的信息: <i>ipr=2</i> 在每次迭代中打印有效哈密顿量的列和重叠矩阵
RESIDUUM= <i>ipr</i>	打印残差矢量的信息: <i>ipr=-1</i> 在迭代中不打印 <i>ipr=0</i> (默认) 打印每个迭代的能量值+ 残差 (平方) 模 <i>ipr=1</i> 还打印复本征值的警告, 以及由于残差矢量模太小致使没有新矢量添加到测试空间的警告 <i>ipr=2</i> 还打印剩余了多少矢量

LOCEOM= <i>ipr</i>	如果出现FOLLOW卡, <i>ipr=1</i> 在每次迭代打印样本矢量和测试矢量的重叠。增加 <i>ipr</i> 会打开越来越多的打印, 主要与局域EOM-CCSD方法有关。
POPUL= <i>ipr</i>	如果 <i>ipr=1</i> , 进行EOM-CCSD波函单激发部分的布居分析。默认使用Löwdin方法。进行Mulliken分析可以在EOM卡加上MULLPRINT=1。注意, 更正确(但也更昂贵)的方法是计算和分析EOM-CCSD 密度矩阵, 见25.1节。
INTERMED= <i>ipr</i>	打印依赖于基态CCSD振幅的中间量: <i>ipr=0</i> 不打印(默认) <i>ipr=1</i> 打印最新创建的中间量 <i>ipr=2</i> 还打印更多的中间量有关信息

## 25.4 例子

### 25.4.1 氟化氢最低激发态的势能面

这个例子演示如何从前一计算续算, 计算几个激发态的势能面。

```
! $Revision: 2006.3 $
***, PES for several lowest states of hydrogen fluoride
memory,2,m
basis=avdz                                ! 定义基组
geometry={h;f,h,r}                        ! z矩阵
r=0.8 Ang                                 ! 从这个间距开始
do n=1,100                                 ! 对这个间距循环
  rr(n)=r                                  ! 把间距存为表格
  hf                                       ! 进行SCF计算
  ccsd                                    ! 进行CCSD计算, 续算
  start,4000.2,save,4000.2                ! 保存最终的T振幅
  eom,-2.1,-1.2,-1.4,start=6000.2,save=6000.2 ! 进行EOM-CCSD计算, 续算
                                           ! 保存最终激发态的振幅

  ebase(n)=energy(1)                      ! 保存该结构的基态能量
  e2(n)=energy(2)-energy(1)               ! 保存该结构的激发态能量
  e3(n)=energy(3)-energy(1)
  e4(n)=energy(4)-energy(1)
  r=r+0.01                                ! 增加间距
enddo                                     ! 结束循环
table,rr,ebase,e2,e3,e4                   ! 制作结果的表格
digits,2,8,5,5,5,5,5,5,5,5,5,5,5,5,5,5 ! 修改小数位数
head,R(Ang),EGRST,E_EXC(2.1),E_EXC(1.2),E_EXC(1.4) ! 修改表头
```



```
! title of table
title,EOM-CCSD excitation energies for hydrogen fluoride (in hartree), basis $basis
save,hf_eom_ccsd.tab                ! 把表格存为文件
```

[http://www.molpro.net/info/current/examples/hf\\_eom\\_pes.com](http://www.molpro.net/info/current/examples/hf_eom_pes.com)

这个计算产生以下表格:

EOM-CCSD excitation energies for hydrogen fluoride (in hartree), basis AVDZ

R(ANG)	EGRST	E_EXC(2.1)	E_EXC(1.2)	E_EXC(1.4)
0.80	-100.23687380	0.56664	0.41204	0.56934
0.81	-100.24094256	0.56543	0.40952	0.56812
0.82	-100.24451598	0.56422	0.40695	0.56690

等等。

## 25.4.2 氟化氢的EOM-CCSD跃迁矩

这个例子演示如何计算并存储CCSD和EOM-CCSD密度矩阵, 计算偶极和四极矩 (计算基态到激发态之间的跃迁矩), 以及如何使用EOM-CCSD激发态密度用于Mulliken布居分析。

```
! $Revision: 2006.0 $
***, Properties and transition moments for several lowest states of hydrogen fluoride
memory,2,m
basis=avdz                                ! 定义基组
geometry={h;f,h,r}                        ! z矩阵
r=0.92 Ang                               ! 定义间距

hf                                         ! 进行SCF计算
{ccsd                                    ! 进行CCSD计算
dm,5600.2                                ! 在这里存储密度矩阵
expec,qm                                ! 请求四极矩计算
eom,-3.1,-2.2,-2.3,-2.4,trans=1}        ! 计算EOM-CCSD及特性

pop;density,5600.2,state=2.4             ! 对2.4态进行布居分析
```

[http://www.molpro.net/info/current/examples/hf\\_eom\\_prop.com](http://www.molpro.net/info/current/examples/hf_eom_prop.com)

这个计算产生以下表格:

```
-----
Final Results for EOM-CCSD
(moments in a.u.)
-----
```

State	Exc. Energy (eV)	X	Y	Z
-------	------------------	---	---	---

2.1	14.436		
Right transition moment	0.00000000	0.00000000	0.65349466
Left transition moment	0.00000000	0.00000000	0.68871635
Dipole strength	0.45007246		
Oscillator strength	0.15917669		
Dipole moment	0.00000000	0.00000000	0.88758090

等等。

### 25.4.3 计算最类似于给定CIS态的EOM-CCSD态

这个例子显示如何强制EOM-CCSD程序收敛到一个态，并且这个态最类似于给定的CIS态。

```
! $Revision: 2006.3 $
***, EOM-CCSD, vector following procedure
memory,2,m
basis=avdz                                ! 定义基组
geometry={h;f,h,r}                        ! z矩阵
r=0.92 Ang                                ! 定义间距
hf;save,2100.2                             ! 进行SCF计算，保存轨道
cis,-4.4,exfile=6000.2                     ! 进行CIS计算，保存振幅
ccsd;save,4000.2                           ! 进行CCSD计算，保存振幅
eom,-4.4,checkovlp=1,exfile=6000.2         ! 进行EOM-CCSD计算，检查与exfile记录存储的
                                           ! CIS矢量的单激发重叠
eompar,inisingl=200,inidoubl=0             ! 作为第一个近似，取近似哈密顿量的200个单激发CSF
ccsd;start,4000.2                           ! 进行CCSD计算，续算
eom,2.4,follow=2,exfile=6000.2,checkovlp=1 ! 对最接近2.4的CIS态进行EOM-CCSD计算，检查
                                           ! 存储在exfile的CIS矢量的单激发重叠
eompar,inisingl=200,inidoubl=0
eomprint,loce=1                             ! 在每次迭代，打印样本矢量和EOM矢量的重叠
```

[http://www.molpro.net/info/current/examples/hf\\_eom\\_conv.com](http://www.molpro.net/info/current/examples/hf_eom_conv.com)

在这个例子中，CIS态2.4对应于EOM-CCSD态1.4!

## 25.5 CIS的激发态

还可以用单激发组态相互作用（CIS）方法计算激发能。默认计算单重激发态。通过在EOM卡设置triplet=1，可以获得三重激发态。不要指望这个方法给出精确结果，但它可以用于相当大的分子。指定要计算的态的方法类似于EOM。设定trans=1将打开单电子特性的计算。默认计算偶极矩。需要其它特性可以用EXPEC卡指定。还可以计算偶极跃迁矩。

```
hf
```

```
cis,-3.1,1.2,trans=1
```

## 25.6 CCSD的一阶和二阶特性

一阶和含频二阶特性来自基于单电子算符期待值的表达式，它们可以用CPROP指令从闭壳层CCSD方法获得。方法的说明见以下论文：

- [1] B. Jeziorski and R. Moszynski, *Int. J. Quantum Chem.*, **48**, 161 (1993);
- [2] T. Korona and B. Jeziorski, *J. Chem. Phys.*, **125**, 184109 (2006);
- [3] R. Moszynski, P. S. Zuchowski and B. Jeziorski, *Coll. Czech. Chem. Commun.*, **70**, 1109 (2005);
- [4] T. Korona, M. Przybytek and B. Jeziorski, *Mol. Phys.*, **104**, 2303 (2006).

注意，用耦合簇波函数从期待值表达式获得的特性不等于从梯度或线性响应方法得到的特性，虽然两种方法得到的结果非常相似。

对于一阶特性，单电子算符应当用EXPEC卡指定，二阶特性用POLARI卡。保存密度可以通过指定DM卡。

对于一阶特性，总是应当给出XDEN=1选项。其它的选项指定单电子密度类型，既可以是直接得自期待值表达式的密度，参见第二篇论文的方程(8)，也可以是严格校正到 $\mathcal{O}(W^3)$  Møller-Plesser (MP) 级别的修正公式，在第一、二篇论文中表示为 $\bar{X}(3)_{\text{resp}}$ 。对于第一种情况，选项PROP\_ORDER=n可用于指定所谓S算符（参见第二篇论文的方程(9)）的单双激发部分的近似级别； $n = \pm 2, \pm 3, \pm 4$ ：对于正n，使用直到n的所有对S的近似；对于负n，只用从|n|级别获得的S计算密度。S算符的另一个选项是HIGHW=n，其中 $n = 0, 1$ ；如果 $n=0$ ，那么高级别MP的 $S_1$ 和 $S_2$ 算符的某些部分会被忽略。下面给出了用标准方法使用这种密度的例子：

```
CPROP,XDEN=1,PROP_ORDER=-4,HIGHW=1
```

在第二种情况中，应当指定选项X3RESP=1和CPHF,1卡（也可以用EXPEC卡），

```
CPROP,XDEN=1,X3RESP=1;CPHF,1
```

对于二阶特性，总是要给出以下选项：

```
CPROP,ROPAGATOR=1,EOMPROP=1
```

第四篇论文推荐的CCSD(3)模型需要另外指定PROP\_ORDER=3和HIGHW=0选项。动力学特性的频率（原子单位）应当在变量OMEGA\_RE（实部）和OMEGA\_IM（虚部）中给出。如果这些数组有一个未给出，则用零填写。二阶特性的其它选项包括，

OMEGAG                      （默认0.3。）有两个线性方程求解程序，OMEGAG是最小频率，  
对其使用第二个求解程序（用于大的频率）。

DISPCOEF= $n$	如果 $n > 0$ ，用POLARI卡给定的算符对范德瓦耳斯因子计算色散积分，用 $n$ 作为数值积分的频率数。这种情况下忽略在OMEGA_RE和OMEGA_IM中给定的频率值。如果在一个脚本中依次计算两个分子，那么还计算混合的色散积分。各向同性 $C_6$ 因子存储于变量DISPC6中。所有计算色散积分所必需的信息都写到ascii文件 $name.dispinfo$ 中，其中的 $name$ 是MOLPRO脚本名称。
THRPROPAG	如果给出，这个阈值作为用线性方程求解程序求解一阶微扰CCSD振幅的收敛标准。
STARTT1= $n$	用迭代线性方程求解程序求解一阶微扰CCSD振幅的各种初始选项，其中最有用的是 $n = 0$ （零初始）和 $n = 7$ （从负的右矢开始，并用某些依赖于Fock矩阵对角项和给定频率的能量因子重新换算）。

## 第二十六章

## 开壳层耦合簇理论

可以在MOLPRO中使用自旋非限制 (RHF-UCCSD) 和部分自旋限制 (RHF-RCCSD) 开壳层耦合簇理论, 描述见J. Chem. Phys. **99** (1993) 5219 (以及勘误, J. Chem. Phys., **112** (2000) 3106)。在两种情况中, 均使用了高自旋RHF参考波函。在MOLPRO中, 没有基于UHF轨道的耦合簇方法 (MOLPRO中唯一使用UHF轨道的关联方法是UMP2)。在后面的描述中使用缩写RCCSD和UCCSD, 但是为了与基于自旋非限制轨道的方法相区别, 通常提到的理论应当是RHF-RCCSD和RHF-UCCSD。在输入命令中, 程序既接受全称, 也接受缩写。

在RCCSD理论中, 对振幅引入了某些限制, 使得波函的线性部分变成了自旋本征函数 (但在UCCSD 方法中不是这样, 即使是使用了RHF参考波函)。目前RCCSD的执行仅仅是初步的, 相比于UCCSD 并不节省CPU时间。不过, 以上文献中提到的改进算法目前正在编写, 在不久的将来就能使用了。

除了使用RCCSD或UCCSD作为关键词之外, 与闭壳层CCSD的输入完全相同。默认情况下, 开壳层轨道和在RHF参考波函中使用的相同, 但是可以用OCC, CLOSED, 和WF卡进行修改。

微扰三阶修正的计算如下:

RCCSD(T), UCCSD(T)	三阶修正的计算定义为: J. D. Watts, J. Gauss and R. J. Bartlett, J. Chem. Phys. <b>98</b> 8718 (1993)
RCCSD[T], UCCSD[T]	三阶修正的计算不考虑单激发的贡献 (有时也称为CCSD+T(CCSD))
RCCSD-T, UCCSD-T	三阶修正的计算定义为: M. J. O. Deegan and P. J. Knowles, Chem. Phys. Letters 227 (1994) 321

实际上，总是计算和打印全部三个修正。以下的变量用于存储结果（这里的CCSD表示UCCSD或RCCSD）：

ENERGY	输入中指定方法的总能量。
ENERGC	没有三阶修正的CCSD总能量。
ENERGT(1)	CCSD(T)总能量。
ENERGT(2)	CCSD [T] 总能量。
ENERGT(3)	CCSD-T总能量。

应当注意到，在开壳层情况下，三激发能量略微依赖于芯轨道的处理。在MOLPRO中，赝正则的alpha和beta 自旋轨道（Chem. Phys. Letters **186** (1991) 130）是在价轨道空间，通过分块对角化相应的Fock矩阵产生的，没有用到冻芯轨道。有些其它程序在正则化和变换过程中包含了冻芯轨道。由于芯-价混合，导致能量略微不同。不能认为这两种方法中的哪个更好或者更合理——这仅仅是一种定义。不过，MOLPRO中的方法效率更高，因为随后涉及的积分变换仅包含价轨道，没有芯轨道。

## 26.1 CN的开壳层耦合簇计算

```
! $Revision: 2006.3 $  
geometry={c;n,c,r}  
r=2.2  
{rhf;  
occ,5,1,1;  
closed,4,1,1;  
wf,13,1,1}  
rccsd-t
```

[http://www.molpro.net/info/current/examples/cn\\_rccsd-t.com](http://www.molpro.net/info/current/examples/cn_rccsd-t.com)

## 第二十七章

### M. KALLAY的MRCC程序 (MRCC)

Molpro有个接口，可以使用M. Kallay和J. Gauss的MRCC程序。MRCC程序的许可和源代码必须从Mihaly Kallay获得<http://www.mrcc.hu>。目前，只支持RHF参考函数<sup>1</sup>的单参考方法，微扰方法和CCn方法只能用于闭壳层。此外，MOLPRO只支持串行执行，也就是说，不能使用mpp版本。

#### 27.1 安装MRCC

`mrcc.tar.gz`文件将由M. Kallay提供。这个文件需要复制到MOLPRO主目录的MRCC目录下。在这个目录中有一个GNUmakefile，输入`make`就会编译MRCC程序，并把MRCC可执行程序链接到MOLPRO的bin目录中，自动由MOLPRO调用。轨道和其它输入信息经由外部文件传递，这些文件对用户是透明的。一旦安装了程序，请在testjobs目录中运行`make mrcctest`。

#### 27.2 运行MRCC

调用MRCC程序的命令为

`MRCC, options`

`directives`

可用的选项总结在表27.1中。更详细地说明请参阅M. Kallay的MRCC手册（mrcc目录中的manual文件）。

在MOLPRO中，使用的方法可以用字符串给出（选项`METHOD=string`）。可用的方法和相应的MRCC输入参数（参见MRCC手册）在表27.2列出。

---

<sup>1</sup>即RHF和ROHF。——译著。

表 27.1: MRCC的选项

选项	别名	默认值 <sup>a</sup>	含义
METHOD	CALC	CC(n)	计算方法。参见表27.2。
EXCITATION	LEVEL	-1	簇算符中的激发级别。
RESTART_CC	RESTART	0	续算选项。如果是1，用前面的振幅续算。
DIRECTORY	DIR	, ,	MRCC运行的子目录（对续算任务必须）。
EOM_NSING	NSING	-1	闭壳层情况下单重激发态的数目。
EOM_NTRIP	NTRIP	0	闭壳层情况下三重激发态的数目。
EOM_NSTATES	NDOUB	-1	开壳层情况下态的数目。
SYMM	SYMMETRY	-1	激发态的对称性。
DENSITY	IDENS	0	密度计算的参数。
HF		1	1用于正则Hartree-Fock轨道，否则为0。
SPATIAL		1	0用于自旋限制轨道，1用于自旋非限制轨道。
NACTO		0	活性占据轨道数目。
NACTV		0	活性虚轨道数目。
SACC		0	自旋匹配耦合簇。
DBOC		0	对角BO校正。
MEMORY		-1	内存。
TOL	ENERGY	-1.0	能量收敛阈值。
FREQ		0.0	动态极化率的频率。
FILE		fort	MRCC fortran文件的名称。
CONVER	ICONV	0	参见mrcc手册。
CS		1	参见mrcc手册。
DIAG		0	参见mrcc手册。
MAXEX		0	参见mrcc手册。

<sup>a</sup> -1表示默认值来自MOLPRO

除了CORE, OCC, ORBITAL, MAXIT指令与在MOLPRO CCSD程序中的指令功能相同以外，通常不需要指令。此外，态的数量可在STATE指令中给出，与EOM\_NSTATES选项有相同的含义。



表 27.2: MRCC程序中使用的的方法

关键词	MRCC参数		注释
	METHOD	LEVEL	
CI(n)组态相互作用方法			
CISD	0	2	
CISDT	0	3	
CISDTQ	0	4	
CI(N)	0	N	用LEVEL指定激发级别N
CC(N)耦合簇方法			
CCSD	1	2	
CCSDT	1	3	
CCSDTQ	1	4	
CC(N)	1	N	用LEVEL指定激发级别N
CC(N-1)[N]耦合簇方法			
CCSD[T]	2	3	
CCSDT[Q]	2	4	
CC(N-1)[N]	2	N	用LEVEL指定激发级别N
CC(N-1)(N) 耦合簇方法。也计算[n]校正			
CCSD(T)	3	3	
CCSDT(Q)	3	4	
CC(N-1)(N)	3	N	用LEVEL指定激发级别N
CC(n-1)(n)-L方法（也计算(n)和[n]校正）			
CCSD(T)-L	4	3	
CCSDT(Q)-L	4	4	
CC(N-1)(N)-L	4	N	用LEVEL指定激发级别N
CC(n)-1a方法			
CCSDT-1A	5	3	
CCSDTQ-1A	5	4	
CC(N)-1A	5	N	用LEVEL指定激发级别N
CC(n)-1b方法			

CCSDT-1B	6	3	
CCSDTQ-1B	6	4	
CC(N)-1B	6	N	用LEVEL指定激发级别N
CCn方法（仅能用于基态）			
CC3	7	3	
CC4	7	4	
CCN	7	N	用LEVEL指定激发级别N
CC(n)-3方法			
CCSDT-3	8	3	
CCSDTQ-3	8	4	
CC(N)-3	8	N	用LEVEL指定激发级别N

例子：H<sub>2</sub>O的闭壳层基态计算：

```
! $Revision: 2006.3 $
***,mrcc calculations for h2o
memory,8,m
gthresh,energy=1.d-8

geometry={
o;h1,o,r;h2,o,r,h1,theta}
theta=104
r=1 ang
basis=vdz

hf
mrcc,method=cc3;                ! CC3计算
method(1)=program
e(1)=energy                     ! 最终能量返回到变量energy中

ccsd(t)                         ! 用Molpro计算CCSD(T)
method(2)='CCSD(T) (MOLPRO)'
e(2)=energy

mrcc,method=ccsd(t)             ! 用MRCC计算CCSD(T)
method(3)='CCSD(T) (MRCC)'
e(3)=energy

mrcc,method=ccsdt,dir=mrccdir   ! CCSDT计算，在mrccdir目录中运行
method(4)=program
```

```

e(4)=energy

mrcc,method=ccsdt(q),restart=1,dir=mrccdir ! CCSDT(Q)计算
                                         ! 用前一个振幅续算

method(5)=program
e(5)=energy

mrcc,method=CC(n),excitation=4,restart=1,dir=mrccdir ! CCSDTQ计算
method(6)=program
e(6)=energy

table,method,e

```

[http://www.molpro.net/info/current/examples/h2o\\_mrcc.com](http://www.molpro.net/info/current/examples/h2o_mrcc.com)

这将产生

METHOD	E
CC3	-76.23912734
CCSD(T) (MOLPRO)	-76.23905150
CCSD(T) (MRCC)	-76.23905150
CCSDT	-76.23922746
CCSDT(Q)	-76.23976632
CCSDTQ	-76.23973043

$H_2O$ 的激发能:

```

! $Revision: 2006.3 $
***,h2o excitation energies
memory,8,m
gthresh,energy=1.d-8
geometry={
o;h1,o,r;h2,o,r,h1,theta}
theta=104
r=1 ang
basis=vdz
hf

ii=0
s=2                ! 每个对称性中的态数
do sym=1,4         ! 对不可约表示循环
ccsd,eom,-(s+0.1*sym);$p=molpro;save_energy
mrcc,method=ccsd, symm=sym,nstates=2;$p=mrcc;save_energy
mrcc,method=ccsdt, symm=sym,nstates=2;$p=mrcc;save_energy
s=1
enddo

```

```
{table,method,prog,states,e,exc
  sort,3}

save_energy={      ! 把结果保存到变量的流程
!nogprint,variable
e1=energy(1)
do i=1,#energy
ii=ii+1
e(ii)=energy(i)
method(ii)=program
prog(ii)=p
states(ii)=i+0.1*sym
exc(ii)=(e(ii)-e1)*toev
end do
}
```

[http://www.molpro.net/info/current/examples/h2o\\_mrcc\\_eom.com](http://www.molpro.net/info/current/examples/h2o_mrcc_eom.com)

这将产生

METHOD	PROG	STATES	E	EXC
CCSD	MOLPRO	1.1	-76.23580212	0.000
CCSD	MRCC	1.1	-76.23580212	0.000
CCSDT	MRCC	1.1	-76.23922746	0.000
CCSD	MOLPRO	1.2	-76.23580212	0.000
CCSD	MRCC	1.2	-76.23580212	0.000
CCSDT	MRCC	1.2	-76.23922746	0.000
CCSD	MOLPRO	1.3	-76.23580212	0.000
CCSD	MRCC	1.3	-76.23580212	0.000
CCSDT	MRCC	1.3	-76.23922746	0.000
CCSD	MOLPRO	1.4	-76.23580212	0.000
CCSD	MRCC	1.4	-76.23580212	0.000
CCSDT	MRCC	1.4	-76.23922746	0.000
CCSD	MOLPRO	2.1	-75.85033256	10.489
CCSD	MRCC	2.1	-75.85033257	10.489
CCSDT	MRCC	2.1	-75.85316687	10.505
CCSD	MOLPRO	2.2	-75.95093334	7.752
CCSD	MRCC	2.2	-75.95093335	7.752
CCSDT	MRCC	2.2	-75.95299013	7.789
CCSD	MOLPRO	2.3	-75.77630664	12.504
CCSD	MRCC	2.3	-75.77630665	12.504
CCSDT	MRCC	2.3	-75.77972816	12.504
CCSD	MOLPRO	2.4	-75.87776149	9.743
CCSD	MRCC	2.4	-75.87776150	9.743
CCSDT	MRCC	2.4	-75.88051189	9.761

$O_2$ 的开壳层基态计算:

```
! $Revision: 2006.3 $
***,O2 tests
memory,8,m
gthresh,energy=1.d-8

geometry={o1;o2,o1,r1}
r1=2.2
set,state=1,symmetry=4,spin=2 ! 三重sigma-态
basis=vdz

rhf
uccsd(t)
method(1)='UCCSD(T) MOLPRO'
e(1)=energy

rccsd(t)
method(2)='RCCSD(T) MOLPRO'
e(2)=energy

mrcc,method=ccsdt,dir=mrccdir
method(3)='CCSDT MRCC'
e(3)=energy

mrcc,method=ccsdtq,restart=1,dir=mrccdir
method(4)='CCSDT MRCC'
e(4)=energy

table,method,e
```

[http://www.molpro.net/info/current/examples/o2\\_mrcc.com](http://www.molpro.net/info/current/examples/o2_mrcc.com)

这将产生

METHOD	E
UCCSD(T) MOLPRO	-149.9815472
RCCSD(T) MOLPRO	-149.9812566
CCSDT MRCC	-149.9816705
CCSDT MRCC	-149.9832255



## 第二十八章

### SMILES (SLATER基组)

SMILES程序包用于使用Slater函数的分子积分，由J. Fernandez Rico, R. Lopez, G. Ramirez, I. Ema, D. Zorrilla, 和K.Ishida编写。它综合了几种求解不同类型积分的技术，综述参见

J. Fernandez Rico, R. Lopez, G. Ramirez, I. Ema, J. Comput. Chem., **25**, 1987-1994 (2004)

和

J. Fernandez Rico, I. Ema, R. Lopez, G. Ramirez and K. Ishida, *SMILES: A package for molecular calculations with STO* in Recent Advances in Computational Chemistry. Molecular Integrals over Slater Orbitals, Telhat Ozdogan and Maria Belen Ruiz eds., ISBN ...

这个代码不包含在二进制版本的MOLPRO中，在从源代码创建MOLPRO程序时，默认也不包含这个代码。应当使用`-slater`配置选项，使这个代码能被编译。

SMILES模块用`intyp='SLATER'`卡调用。程序包产生的辅助文件名可以用`SLFILES=`文件名提供。默认名称是`'slscratch'`

$(AB|AC)$ 和 $(AB|CD)$ 类型的三中心双电子积分用高斯函数展开式(STO-nG)计算。展开式的默认长度是9(STO-9G)。但是对于后-HF计算，用STO-9G展开式获得的积分无法提供足够的精度，特别是当使用了高质量基组。在这些情况下，展开式的长度可通过设置变量`NGSSTO=`高斯函数个数 来修改。尽管允许的最大值是`NGSSTO=30`，但程序包实际使用的展开式长度依赖于 $(n,l)$ 量子数。如果请求未包含的展开式，程序将采用目前尽可能最大的展开式。

程序的限制：在当前版本中，最多允许511个基函数，并且不能使用收缩函数。

## 28.1 内部基组

可以使用以下内部基组。

表 28.1: SMILES中的内部基组

基组	原子			
	H-He	Li-Be	B-Ne	Na-Ar
VB1	[3,1]	[5,1]	[5,3,1]	—
CVB1	[3,1]	[6,2]	[6,4,1]	—
FVB1	[3,2,1]	[5,3,1,1]	[5,4,2,1]	—
ZVB1	[3,1]	[4,3,1]	[4,3,1]	[6,5,1]
VB2	[4,2,1]	[6,2,1]	[6,4,2,1]	—
CVB2	[4,2,1]	[7,3,2]	[7,5,3,1]	—
ZVB2	[4,2,1]	[5,4,2,1]	[5,4,2,1]	[7,6,2,1]
VB3	[5,3,2,1]	[7,3,2,1]	[7,5,3,2,1]	—
CVB3	[5,3,2,1]	[8,4,3,2]	[8,6,4,3,1]	—
ZVB3	[5,3,2,1]	[6,5,3,2,1]	[6,5,3,2,1]	[8,7,3,2,1]

## 28.2 外部基组

外部基组可以通过文件提供，它必须位于工作目录中。每个记录包含以下数据（自由格式）：

```
I N L EXP NG
```

其中，

I	原子类型索引号（整数）
N	主量子数（整数）
L	角量子数（整数）
EXP	指数（双精度）
NG	用于 $(AB AC)$ 和 $(AB CB)$ 积分的高斯函数个数（整数）



原子类型索引号用于建立基函数和中心（原子）之间的对应关系。具有同样I值的所有记录用来定义所有相应类型原子的基组。例如，对于CO<sub>2</sub>分子以下结构数据的计算：

```
Geometry={
  3

C1, 0.000000000000E+00, 0.000000000000E+00, 0.000000000000E+00
O2, 0.000000000000E+00, 0.000000000000E+00, -0.117963799946E+01
O3, 0.000000000000E+00, 0.000000000000E+00, 0.117963799946E+01
}
```

可以通过外部文件提供Clementi和Roetti的单 $\zeta$ 基组，形如：

1	1	0	5.67263	12
1	2	0	1.60833	12
1	2	1	1.56788	12
2	1	0	7.65781	12
2	2	0	2.24588	12
2	2	1	2.22662	12

前三个记录定义第一类原子（本例是碳）的基组，接下来的三个定义第二类原子（氧）的基组。

即使是对双原子分子，NG也是必须的，虽然此时不使用高斯指数，NG的值在这种情况下是无关的量。



## 第二十九章

## 局域关联处理

### 29.1 介绍

MOLPRO的局域关联程序目前可以进行闭壳层LMP2, LMP3, LMP4(SDTQ), LCISD, LQCISD(T), 和LCCSD(T)计算。对于大分子, 如果忽略相距很远的轨道对, 并且使用积分直接算法的话, 那么所有方法随分子尺寸的标度是线性的。

通过对积分的计算使用密度拟合(DF)近似, 可以获得更高的效率。密度拟合可以用于直到LCCSD(T)的所有局域方法, 以及用于LMP2的解析梯度。目前只有迭代三激发方法, 如LCCSDT-1b, 还不能使用密度拟合。

DF引入的误差可以忽略, 因此强烈推荐使用DF方法。在DF-LMP2中用LOCFIT选项可以获得线性标度(参见本节的文献[1]); 在DF-LCCSD(T)中, 最重要的部分也是线性标度的, 但有些转换步骤的标度是二次的。

能量梯度可用于LMP2, DF-LMP2, DF-SCS-LMP2, 和LQCISD(后者只能用LOCAL=1, 也就是说, 局域计算用正则程序模拟, 并且仅保存轨道对个数做了约化的结果)。

局域显关联方法(DF-LMP2-R12和DF-LMP2-F12)的说明在第三十章。

在使用这些方法前, 强烈建议阅读文献, 以便对基本概念和近似有所了解。最近的评论[1]和参考文献[2]适合作为入门的介绍。

参考文献:

评论:

[1] H.-J. Werner and K. Pflüger, *On the selection of domains and orbital pairs in local correlation treatments*, Ann. Rev. Comp. Chem., in press. (预印本位于 <http://www.theochem.uni-stuttgart.de/~werner/reprints.html>)

一般的局域耦合簇:

- [2] C. Hampel and H.-J. Werner, *Local Treatment of electron correlation in coupled cluster (CCSD) theory*, J. Chem. Phys. **104**, 6286 (1996).
- [3] M. Schütz and H.-J. Werner, *Local perturbative triples correction (T) with linear cost scaling*, Chem. Phys. Letters **318**, 370 (2000).
- [4] M. Schütz, *Low-order scaling local electron correlation methods. III. Linear scaling local perturbative triples correction (T)*, J. Chem. Phys. **113**, 9986 (2000).
- [5] M. Schütz and H.-J. Werner, *Low-order scaling local electron correlation methods. IV. Linear scaling local coupled-cluster (LCCSD)*, J. Chem. Phys. **114**, 661 (2001).
- [6] M. Schütz, *Low-order scaling local electron correlation methods. V. Connected Triples beyond (T): Linear scaling local CCSDT-1b*, J. Chem. Phys. **116**, 8772 (2002).
- [7] M. Schütz, *A new, fast, semi-direct implementation of Linear Scaling Local Coupled Cluster Theory*, Phys. Chem. Chem. Phys. **4**, 3941 (2002).

#### 长距离轨道对的多极处理:

- [8] G. Hetzer, P. Pulay, H.-J. Werner, *Multipole approximation of distant pair energies in local MP2 calculations*, Chem. Phys. Lett. **290**, 143 (1998).

#### 线性标度的局域MP2:

- [9] M. Schütz, G. Hetzer and H.-J. Werner, *Low-order scaling local electron correlation methods. I. Linear scaling local MP2*, J. Chem. Phys. **111**, 5691 (1999).
- [10] G. Hetzer, M. Schütz, H. Stoll, and H.-J. Werner, *Low-order scaling local electron correlation methods II: Splitting the Coulomb operator in linear scaling local MP2*, J. Chem. Phys. **113**, 9443 (2000).

#### 密度拟合局域方法:

- [11] H.-J. Werner, F. R. Manby, and P. J. Knowles, *Fast linear scaling second-order Møller-Plesset perturbation theory (MP2) using local and density fitting approximations*, J. Chem. Phys. **118**, 8149 (2003).
- [12] M. Schütz and F.R. Manby, *Linear scaling local coupled cluster theory with density fitting. Part I: 4- external integrals*, Phys. Chem. Chem. Phys. **5**, 3349 (2003).
- [13] Polly, H.-J. Werner, F. R. Manby, and Peter J. Knowles, *Fast Hartree-Fock theory using local density fitting approximations*, Mol. Phys. **102**, 2311 (2004).
- [14] H.-J. Werner and M. Schütz, *Low-order scaling coupled cluster methods (LCCSD(T)) with local density fitting approximations*, in preparation.

#### LMP2梯度和几何优化:

- [15] A. El Azhary, G. Rauhut, P. Pulay and H.-J. Werner, *Analytical energy gradients for*

- local second-order Møller-Plesset perturbation theory*, J. Chem. Phys. **108**, 5185 (1998).
- [16] G. Rauhut and H.-J. Werner, *Analytical Energy Gradients for Local Coupled-Cluster Methods*, Phys. Chem. Chem. Phys. **3**, 4853 (2001).
- [17] M. Schütz, H.-J. Werner, R. Lindh and F.R. Manby, *Analytical energy gradients for local second-order Møller-Plesset perturbation theory using density fitting approximations*, J. Chem. Phys. **121**, 737 (2004).

#### LMP2振动频率:

- [18] G. Rauhut, A. El Azhary, F. Eckert, U. Schumann and H.-J. Werner, *Impact of Local Approximations on MP2 Vibrational Frequencies*, Spectrochimica Acta **55**, 651 (1999).
- [19] G. Rauhut and H.-J. Werner *The vibrational spectra of furoxan and dichlorofuroxan: a comparative theoretical study using density functional theory and Local Electron Correlation Methods*, Phys. Chem. Chem. Phys. **5**, 2001 (2003).
- [20] T. Hrenar, G. Rauhut and H.-J. Werner, *Impact of local and density fitting approximations on harmonic vibrational frequencies*, J. Phys. Chem. A., **110**, 2060 (2006).

#### 分子间相互作用以及BSSE问题:

- [21] M. Schütz, G. Rauhut and H.-J. Werner, *Local Treatment of Electron Correlation in Molecular Clusters: Structures and Stabilities of  $(H_2O)_n$ ,  $n = 2 - 4$* , J. Phys. Chem. **102**, 5997 (1998). 此外参见[2]以及其中的参考文献。
- [22] N. Runeberg, M. Schütz and H.-J. Werner, *The aurophilic attraction as interpreted by local correlation methods*, J. Chem. Phys. **110**, 7210 (1999).
- [23] L. Magnko, M. Schweizer, G. Rauhut, M. Schütz, H. Stoll, and H.-J. Werner, *A Comparison of the metallophilic attraction in  $(X-M-PH_3)_2$  ( $M=Cu, Ag, Au$ ;  $X=H, Cl$ )*, Phys. Chem. Chem. Phys. **4**, 1006 (2002).

## 29.2 开始

通过在命令名前面加上一个L打开局域关联处理, 也就是使用LMP2, LMP3, LMP4, LQCISD, LCCSD, 或LCISD命令。

LQCISD和LCCSD命令还可以在后面加上微扰三激发处理的定义(例如, LCCSD(T0)):

- (T)            使用默认三激发方法。目前这就是(T0)。
- (T0)          非迭代的局域三激发。这是最快的三激发选项。它的精度通常足够, 在大多数情况下推荐使用。

- (T1) T0加上三激发振幅的一次微扰更新。如果T0的精度不够（这种情况很少见!），这可用于改善精度。计算耗时至少是T0的两倍。与T0相反，三激发振幅必须存在硬盘上，对于大分子计算这可能是瓶颈。另外，内存需求也比T0大得多。
- (T1C) 同T1，但使用了缓存算法，可以避免把所有的三激发振幅同时存储到硬盘上（同(T1)或(TF)的情况）。因此，T1C需要比T1较少的磁盘空间，但CPU时间更长。用于缓存算法的磁盘空间越多（在local卡使用T1DISK选项，见下），使用的CPU时间越少。
- (TF) 完全迭代三激发计算。这种方法使用完全域，没有弱轨道近似，会给出与正则(T)计算同样的结果。典型情况下，需要3-5次迭代，因此计算量要比(T1)多2-3倍。磁盘和内存需求与T1相同。同时还计算并打印T0能量。TFULL和FULL是TF的别名。
- (TA) 同TF，但还计算T1能量。由于第一次迭代不同于T1，因此三激发迭代的收敛略微不同于TF和TA（大多数情况下TF更快）。TALL和ALL是TA的别名。

调用密度拟合可以用命令名加上前缀DF-，例如，DF-LMP2，DF-LCCSD(T0)，等。在密度拟合计算中，需要附加的辅助基组。选择这种基组的细节及密度拟合的选项在第29.10节和第十五章描述。

局域耦合LMP2或耦合簇计算的一般输入为：

LMP2, <i>options</i>	局域MP2计算
LCCSD, <i>options</i>	局域CCSD计算
LCCSD(T0), <i>options</i>	局域CCSD(T0)计算

在随后的LOCAL和MULTP指令中，还可以在命令行给出同样的选项。除了使用MULTP指令以外，也可以在命令行使用MULTP选项。

下面，我们首先给出所有选项和指令的总结。它们在以后的章节有更详细的说明。对于新用户，在开始计算前建议阅读本章最后的29.9节。

### 29.3 选项的总结

在命令行可以指定很多选项。对所有选项设置了合适的默认值，因此这些选项通常只有对特殊目的才需要修改。为了方便以及由于历史原因，各个选项还有别名，它

们通常对应于程序中使用的变量名。表29.1总结了这些选项，别名，和默认值。下面详细讲述这些参数。

表 29.1: `local (multp)`选项及其默认值的总结

参数	别名	默认值	含义
一般参数:			
LOCAL		4	确定使用哪个程序
MULTP		0	打开远距离轨道对的多极近似
SAVEDOM	SAVE	0	指定用于保存域信息的记录
RESTDOM	START	0	指定用于读入域信息的记录
LOCORB		0	激活或关闭轨道局域化
LOC_METHOD		0	指定使用哪种局域化方法
CANONICAL		0	允许使用正则虚轨道（用于测试）
PMDEL	CPLDEL	0	在PM局域化中，去除弥散函数的贡献
SAVORB	SAVLOC	0	指定用于保存局域轨道的记录
DOMONLY		0	如果设置为1，只产生域 如果设置为2，只产生轨道域
定义域的参数:			
THRBP	DOMSEL	0.98	轨道域的Boughton-Pulay选择标准
NPASEL		0	在NPA选择标准中，对轨道域使用的电荷
CHGMIN		0.01	确定域中允许的最小原子电荷
CHGMINH		0.03	同CHGMIN，但用于氢原子（默认0.03）
CHGMAX		0.40	如果原子电荷大于这个值，那么这个原子总是包含在域中
MAXANG	MAXL	99	BP域选择的角动量限制
MAXBP		0	在BP流程中，确定原子如何分类
MULLIKEN	LOCMUL	0	确定原子电荷的方法
MERGEDOM		0	合并重叠域
DELCOR	IDLCOR	2	删除直到某个壳层的投影芯AO
DELBAS	IBASO	0	确定如何去除冗余
域展开的距离标准:			
REXT		0	用于全部轨道对域的标准
REXTS		0	用于强轨道对域的标准
REXTC		0	用于强轨道对和封闭轨道对域的标准
REXTW		0	用于强轨道对、封闭轨道对和弱轨道对域的标准
域展开的连结性标准			
IEXT		0	用于全部轨道对域的标准
IEXTS		0	用于强轨道对域的标准
IEXTC		0	用于强轨道对和封闭轨道对域的标准
IEXTW		0	用于强轨道对、封闭轨道对和弱轨道对域的标准

选择轨道对分类的参数:

USE_DIST		1	确定是否使用了距离的连结性标准
RCLOSE	CLOSEP	1	用于选择封闭轨道对的距离标准
RWEAK	WEAKP	3	用于选择弱轨道对的距离标准
RDIST	DISTP	8	用于选择远距离轨道对的距离标准
RVDIST	VERYD	15	用于选择非常远距离轨道对的距离标准
ICLOSE		1	用于选择封闭轨道对的连结性标准
IWEAK		2	用于选择弱轨道对的连结性标准
IDIST		5	用于选择远距离轨道对的连结性标准
IVDIST		8	用于选择非常远距离轨道对的连结性标准
CHGMIN_PAIRS	CHGMINP	0.20	确定用于轨道对分类的最小原子电荷
KEEPCL		0	确定在LCCSD中是否包含了封闭轨道对

对交换算符进行多极处理的参数:

DSTMLT	3	远距离轨道对的多极展开级别
--------	---	---------------

用于能量分区的参数:

IEPART	0	如果非零, 进行能量分区
EPART	3.0	用于确定各个单体的截断因子

使用DELBAS=1检查冗余的参数 (不推荐):

TYPECHECK	TYPECHK	1	激活基函数类型的限制
DELSHL	IDLSHL	1	确定是否删除整个壳层
DELEIG	IDLEIG	1	确定如何选择冗余函数
DELCMIN	CDELMIN	0.1	和DELEIG=1一起使用的参数

在LCCSD中, 用于选择算符域的参数:

OPDOM	IOPDOM	5	如何对LCCSD确定算符域
RMAXJ		8	J-算符列表的距离标准
RMAXK		8	K-算符列表的距离标准
RMAXL		15	L-算符列表的距离标准
RMAX3X		5	3-ext积分列表的距离标准
RDOMJ		0	J-算符域的距离标准
RDOMK		8	K-算符域的距离标准
IMAXJ		5	J-算符列表的连接性标准
IMAXK		5	K-算符列表的连接性标准
IMAXL		8	L-算符列表的连接性标准
IMAX3X		3	3-ext积分列表的连接性标准
IDOMJ		0	J-算符域的连接性标准
IDOMK		5	K-算符域的连接性标准

各种选项:

SKIPDIST	SKIPD	3	确定在什么阶段除去弱轨道对和远距离轨道对
----------	-------	---	----------------------



ASYDOM	JITERM	0	用于使用反对称域的参数
LOCSING	LOCSNG	0	确定用于单激发的虚空间
PIPEKAO	LOCAO	0	激活AO局域化标准
NONORM		2	确定投影函数是否归一化
LMP2ALGO	MP2ALGO	3	如果非零，在LMP2迭代中使用较低标度的方法
OLDDEF		0	允许恢复到旧的默认值
T1DISK		10	用于T1缓存算法的最大磁盘空间（单位GB）
阈值:			
THRBP		0.98	Boughton-Pulay方法的阈值
THRPIP		1.d-12	Pipek-Mezey局域化阈值
THRRORB		1.d-6	用小模排除投影轨道的阈值
THRLOC		1.d-6	除去冗余投影轨道的阈值
THRCOR		1.d-1	除去投影芯轨道的阈值
THRMP2		1.d-8	在LMP2迭代中，忽略小Fock矩阵元的阈值

## 29.4 指令总结

与在正则程序中的标准指令相同，例如，OCC，CLOSED，CORE，WF，ORBITAL等对局域方法也是有效的。此外，还有一些仅能用于局域计算的指令：

LOCAL	调用局域方法，并允许指定选项，和命令行相同。
MULTP	同LOCAL，但对远距离轨道对使用多极近似。
DOMAIN	手动定义域（不推荐）。
MERGEDOM	允许合并域。
REGION	允许选择分子区域进行某种理论级别的处理。
ENEPART	轨道对能量的分析。
SAVE	保存域和LCCSD振幅。
START	用前一计算的域和LCCSD振幅续算。

## 29.5 一般选项

LOCAL=*local* 确定使用哪种方法：

LOCAL=0: 常规的（非局域）计算。

LOCAL=1: 用正则MO模拟局域方法。局域基组只用在中间阶段，用于在每次迭代升级振幅（仅用于测试目的）。

LOCAL=2: 用局域基组进行计算，但是不使用局域分块（也即，使用整

个矩阵)。这是最昂贵的方法, 仅用于测试目的。

LOCAL=3: 完全局域计算 (已作废)。

LOCAL=4: 完全局域计算 (默认)。对于有很多弱轨道对的大分子, 这是最快的方法, 并且内存需求最少。

LOCORB=*option*

如果给出了这个选项且 *option* > 0, 那么轨道用Pipek-Mezey技术局域化。如果没有给出这个选项, 或者 *option*=0 (默认), 则轨道局域化, 除非在轨道记录中发现了局域化轨道 (参看ORBITAL指令和LOCALI命令)。对于后一种情况, 使用最近一次的局域化轨道。设定 *option*=-1将关闭局域化。如果 *option* > 1, 则打印局域化轨道。注意: 只能用LOCALI命令执行Boys局域化。如果在轨道记录中发现了Boys轨道, 且没有LOCORB选项, 或者 *option* ≤ 0, 那么程序将使用Boys轨道。

LOC\_METHOD=*record*

这一选项允许在Pipek-Mezey (*method*=PM) 或自然轨道 (*method*=NBO) 局域化之间作出选择。如果请求Pipek-Mezey轨道, 将使用默认的Boughton-Pulay域选择。当*method*=NBO时, 域选择将基于NPA电荷, 且NPASEL=0.03 (默认)。在两种情况下, 直接给出域选择参数 (参见DOMSEL和NPASEL选项) 可以使用其它的域标准。如果在轨道记录中发现了局域化轨道, 但类型不一致, 那么将按照*method*重新进行轨道局域化, 并用于计算中。

SAVORB=*record*

允许把局域化的和投影的轨道保存到 *record*=*name.ifil* 中, 用于以后使用 (例如, 绘图)。两个轨道集存储在同一个转存记录中, 并且在以后的阶段可以重新获得, 分别使用

ORBITAL,*record*,[TYPE=]LOCAL 或

ORBITAL,*record*,[TYPE=]PROJECTED

DOMONLY=*value* 如果 *value* > 0, 只产生域, 但不计算能量。这可用于检查和保存域, 用于以后使用。

DSTMLT=*level* 确定远距离轨道对多极展开式的展开级别 (例如, 1表示偶极近似, 2表示四极近似, 等)。对MULTP 默认是3。

INTERACT 在一个计算中自动确定各个分子, 并设置合适的轨道对列表用于相互作用能的计算。更多细节参见29.9.8节。

### 能量分割的参数:

IEPART=*value* 使用/禁用能量分割。

*iepart*=0: 禁止能量分割。

*iepart*=1: 允许能量分割。

*iepart*=2: 允许能量分割。此外，还打印所有轨道对能量及其成分的列表。

EPART=*cutoff* 用于确定簇（也就是中心组）中各个单体的截断参数。应当略大于最大的分子内部键长（单位a.u.）。

### 各类选项:

SKIPDIST=*skipdist*

测试参数。它的值只影响效率但不影响结果。

*skipdist*=-1: 在MP2之后，弱轨道对和远距离轨道对设置为0，但是不会从轨道对列表中删除，并且在任何循环中不会被跳过。

*skipdist*=0: 不从轨道对列表中删除轨道对，但是在适当情况下，在循环中跳过弱轨道对和远距离轨道对。

*skipdist*=1: 从一开始就忽略非常远距离的轨道对。远距离轨道对在MP2之后除去。

*skipdist*=2: 同*skipdist*=1，但是在MP2之后还除去弱轨道对。

*skipdist*=3: 同*skipdist*=2，但是对于远距离轨道对使用多极近似的LMP2，并从算符列表中除去远距离轨道对。这是默认的。

ASYDOM=*jiterm*

实验性的测试参数。对于远距离轨道对，允许使用反对称域。反对称域是对远距离轨道对多极方法的近似补充，它禁止处理那些没有使用多极展开计算积分的组态。这会使计算省时，并改善数值稳定性。

*jiterm*=0: 禁止反对称域。

*jiterm*=-1: 允许反对称域（默认）。

*jiterm*=-2: 允许改变反对称域的形式：交换算符在一开始将会投影到反对称域中，而不是简单地挑选。

LOCSING=*locsing*

如果*locsing*不等于0，则单激发使用全空间，也即，它们不会做局域处理。这只能用于LOCAL=1。

**MAXANG=*lmax*** 这一实验性选项的目的是在域的选择中降低基组对于Boughton-Pulay (BP)方法的敏感度。当计算近似轨道与精确轨道的重叠时，只包含角动量到 $lmax-1$ 的基函数。例如，**MAXANG=2**表示忽略所有 $d$ ,  $f$ , 和更高角动量函数的贡献。为获得合理的域，通常必须减小**THRBP**的值（大约到0.97）。使用这个选项要小心！

**PIPEKAO=*option***

如果 $option \geq 0$ ，轨道通过对给定原子的给定类型基函数因子做最大化，进行局域化。通常这只对原子中唯一定义的简并轨道有用。例如，当用这个选项对像 $(Ar)_2$ 这样的大键长的二聚物进行轨道局域化时，将获得干净的 $s$ ,  $p_x$ ,  $p_y$ , 和 $p_z$ 原子轨道。对于分子计算不推荐使用这个选项！

**NONORM=*value*** 确定投影函数是否归一化（不推荐）。

*value*=-1: 在冗余检查前对投影轨道进行归一化。

*value*=0: 在冗余检查后对投影轨道进行归一化（默认）。

*value*=1: 在冗余检查中对投影轨道进行归一化，之后不进行归一化。

*value*=2: 从不进行投影轨道归一化（在梯度计算中默认）。

**LMP2ALGO=*value***

如果非零，在LMP2迭代中使用低级别的标度方法。可用的值有1, 2, 3。如果使用了大基组，3通常最快。

**OLDDEF=*value*** 用于与旧版本兼容：如果非零，恢复旧的默认值。以上设置的选项可能会被改写。

**阈值：**

**THRPIP=*thresh***

用于Pipek-Mezey局域化的阈值。如果所有的 $2 \times 2$ 转角都小于 $thresh$ ，那么假定局域化收敛。默认是 $1.d-12$ 。还可以用**GTHRESH**, **LOCALI=*thresh***做全局修改。

**THRORB=*thresh***

在投影出占据空间后，从轨道对的域中除去模小于  $thresh$  的函数的阈值。默认是 $throrb=1.d-6$ 。

**THRLOC=*thresh***

用于从轨道对的域中除去冗余基函数的阈值。对于 $\tilde{S}^{ij} < thresh$ 的每个本征值，删除一个函数。默认是1.d-6。用于删除函数的方法依赖于参数IDLEIG 和IBASO。

THRMP2=*thresh*

用于忽略LMP2迭代中小Fock矩阵耦合的阈值（默认1.d-8）。指定较大的阈值会加速迭代，但是可能会导致能量上的小误差。在最初的几个迭代中，会自动选择较大的阈值。在迭代过程中会逐渐减小到指定的最终值。

THRCOR=*thresh*

用于删除投影芯轨道的阈值。如果函数的模小于*thresh*，则只删除函数（默认0.1）。

这些阈值也可以用THRESH指令指定。

## 29.6 用于选择域的选项

以下部分描述影响域的最重要选项。

### 29.6.1 标准域

总是首先确定标准域。它们用于定义强轨道对、封闭轨道对、弱轨道对和远距离轨道对。更精确的结果可以用扩展域获得，在29.6.2节描述。

THRBP=*value* 用Boughton和Pulay (BP) 的方法，选择对轨道域有贡献的原子的阈值。需要在域中包含尽可能多的满足BP标准的原子。被考虑的原子的顺序依赖于参数MAXBP，见下。默认是THRBP=0.98。THRBP=1.0在每个轨道域中包含所有原子，也就是将导致全域。如果没有忽略轨道对，这会产生正则MP2能量。

标准对基组略有依赖性。关于这个阈值的推荐值，见29.9.4节。

CHGMIN=*value*

在一个域中，对一个原子（除H以外）确定允许的Mulliken（或Löwdin）电荷最小值，也即，不包含电荷（绝对值）较小的原子，即使是在THRBP标准不满足的情况下也是如此（默认0.01）。

CHGMINH=*value*

同CHGMIN，但用于H原子（默认0.03）。

CHGMAX=*value*

如果原子电荷大于这个值，则包含该原子，而不依赖于任何其它顺序。

MAXBP=*maxbp*

如果 $maxbp=1$ ，则原子按照其对Boughton-Pulay重叠的贡献排列。如果 $maxbp=0$ （默认），则原子按照原子电荷排列。对于这两种情况，总是包含电荷大于CHGMAX的原子，电荷相同的原子作为组加入。

MULLIKEN=*option*

确定原子电荷的方法。MULLIKEN=0（默认）：使用 $\mathbf{S}^{-\frac{1}{2}}\mathbf{C}$ 对角元的平方（Löwdin 电荷）；MULLIKEN=1：使用Mulliken总电荷。第一个选项对基组依赖性较小，对弥散基组更可靠。

MERGEDOM=*number*

如果 $number$ 大于0，将对共同含有 $number$ 个或更多原子的所有轨道域进行合并（ $number=1$ 作为 $number=2$ 处理，默认是0）。这对于共轭体系或芳香体系的几何优化特别有用，例如，苯。对于后一种情况，MERGEDOM=1会产生完全的 $\pi$ 域，也就是说，所有三个 $\pi$ 轨道的域由所有碳的基函数构成。注意：在打印以上的轨道域之后，产生合并的域，关于合并域的信息另外打印。几何优化的更多讨论参见29.9.7节。

还有一些选项通常不需要修改：

DELBAS=*ibas0*

这个参数确定在轨道对的域中排除冗余函数的方法。

$ibas0=0$ ：除去归一化本征矢 $\tilde{\mathbf{S}}^{ij}$ 的空间（默认），它对应于小本征值。其它的值不推荐，这里不再详述。

DELCOR=*nshell*

激活排除芯轨道函数的功能。

如果 $nshell=1$ ，只有1s函数从投影空间除去。

如果 $nshell=2$ （默认），对第一行原子除去1s函数，对第二行原子除去1s，2s，和2p函数。对H和He原子不除去函数。如果使用了有效芯

势，对相应的原子不排除芯轨道。如果投影函数的模低于THRCOR（默认0.1），那么也删除这个函数。

### 29.6.2 扩展域

域的扩展有两种模式：可以使用距离标准REXT，REXTS，REXTC，或REXTW。单位是Bohr，表示在标准轨道域[*ij*]中，任何原子与其它原子的最小距离。如果发现一个原子位于给定的距离内，那么该原子的所有PAO会被加入到域[*ij*]中。还可以用连接性标准IEXT，IEXTS，IEXTC，或IEXTW。表示在标准域[*ij*]内，所含的任何原子与另一个原子之间的成键数量。距离标准的优点是，这种方法还选择与当前域没有键连接，但位于给定半径内的原子。另一方面，连接性标准不依赖于不同的键长，例如对于第一和第二行原子。两种方法只能用一个，也就是说，二者是互相排斥的。

REXT= <i>value</i>	用于扩展所有轨道对域的距离标准。
REXTS= <i>value</i>	用于扩展强轨道对域的距离标准。
REXTC= <i>value</i>	用于扩展强轨道对和封闭轨道对域的距离标准。
REXTW= <i>value</i>	用于扩展强轨道对、封闭轨道对和弱轨道对域的距离标准。
IEXT= <i>value</i>	用于扩展所有轨道对域的连接性标准。
IEXTS= <i>value</i>	用于扩展强轨道对域的连接性标准。
IEXTC= <i>value</i>	用于扩展强轨道对和封闭轨道对域的连接性标准。
IEXTW= <i>value</i>	用于扩展强轨道对、封闭轨道对和弱轨道对域的连接性标准。

默认不对域扩展，即，以上所列所有参数的默认值是零。注意，轨道对的分类由标准域的基础确定，因此域的扩展对轨道对列表没有影响。

还要注意，计算量随着域的大小呈四次方增加，因此对域的扩展将显著增加计算量。这并不影响渐进极限的线性标度性质。

### 29.6.3 人工定义轨道域（DOMAIN）

用DOMAIN指令可以人工定义轨道域：

```
DOMAIN,orbital,atom1, atom2 ...
```

其中，*orbital*的形式为iorb.isym，例如，3.1是对称性1中的第三个轨道，*atom<sub>i</sub>*是Z-矩阵结构输入中给出过的原子标记，或者Z-矩阵的行号。以给定原子为中心的所有基函数包含在域中。例如，

```
DOMAIN,3.1,C1,C2
```

对碳原子C1和C2之间的双中心键定义了一个域。DOMAIN 指令必须在所有OCC，CLOSED，

或CORE指令后给出。注意，局域轨道的顺序依赖于局域流程，甚至可能会随着结构改变，因此使用人工DOMAIN输入时要特别当心！没有直接用DOMAIN指令定义的所有轨道的域，像往常一样由程序自动确定。

## 29.7 选择轨道对分类的选项

定义轨道对的分类有两种模式：可以使用距离标准RCLOSE, RWEAK, RDIST, RVDIST。单位是Bohr，表示对于给定的轨道对 $(ij)$ ，标准轨道域 $[i]$ 和标准轨道域 $[j]$ 中任何原子间的最小距离 $R^{(ij)}$ 。还可以用连接性标准ICLOSE, IWEAK, IDIST, IVDIST。表示标准域 $[i]$ 和标准域 $[j]$ 中包含的任何原子之间的最少成键数。使用连接性标准的优点是不依赖于键长，而距离标准（默认）的优点是在非键情况下同样有效。两种方法只能用一个，也就是说，二者是互相排斥的。默认使用距离标准。要对轨道对选择使用连接性标准，需要设置选项USE\_DIST=0。

USE\_DIST      （默认1）如果非零，使用距离标准，否则使用连接性标准。

CHGMIN\_PAIRS   如果原子Löwdin电荷大于CHGMIN\_PAIRS（默认值0.2），在轨道对分类中只考虑主域中的原子。引入这个标准是为了减小轨道对的选择对局域化尾数的依赖性。

RCLOSE      （默认1）强轨道对定义为 $0 \leq R^{(ij)} < \text{RCLOSE}$ 。封闭轨道对定义为 $\text{RCLOSE} \leq R^{(ij)} < \text{RWEAK}$ 。

RWEAK      （默认3）弱轨道对定义为 $\text{RWEAK} \leq R^{(ij)} < \text{RDIST}$ 。

RDIST      （默认8）远距离轨道对定义为 $\text{RDIST} \leq R^{(ij)} < \text{RVDIST}$ 。

RVDIST      （默认15）非常远距离的轨道对 $R^{(ij)} \geq \text{RVDIST}$ 被忽略。

ICLOSE      （默认1）强轨道对用至少ICLOSE个键分开。封闭轨道对用至少ICLOSE但少于IWEAK个键分开。

IWEAK      （默认2）弱轨道对用至少IWEAK但少于IDIST个键分开。

IDIST      （默认5）远距离轨道对用至少IDIST但少于IVDIST个键分开。

IVDIST      （默认8）非常远距离轨道对（被忽略）用至少IVDIST个键分开。

KEEPCL      （默认0）

如果KEEPCL=1，封闭轨道对的LMP2振幅包含在强轨道对LCCSD残差的计算中。

如果KEEPCL=2，所有封闭轨道对全部包含在LCCSD中（这不影响三激



发列表)。该选项目前的执行效率很低，因此会导致CPU时间急剧增加。

把距离标准设为零意味着直到相应分类的所有轨道对都作为强轨道对处理。例如， $R_{CLOSE}=0$ 表示强轨道对和封闭轨道对全部包含在LCCSD中（这种情况下， $KEEPCL=1$ 没有影响）。但是注意，设定 $R_{CLOSE}=0$ 会增加三激发列表的长度。

## 29.8 指令

### 29.8.1 LOCAL指令

LOCAL指令可用于指定局域计算的选项。如果在局域计算的命令区域出现这个指令，那么选项仅用于当前计算，这完全等价于在命令行指定选项。LOCAL指令也可在命令区域之外给出，这种情况下，选项还用于同一输入中随后的局域关联计算。

例如：

```
DF-LMP2,THRBP=0.985
```

等价于

```
{DF-LMP2
LOCAL,THRBP=0.985}
```

在以下的例子中，LOCAL指令是全局的，作用于随后的所有局域计算，即，两个计算都使用 $THRBP=0.985$ 。

```
LOCAL,THRBP=0.985
DF-LMP2      ! 局域MP2计算
OPTG         ! 用DF-LMP2能量进行几何优化
DF-LCCSD(T)  ! 优化结构的局域耦合簇计算
```

### 29.8.2 MULTP指令

MULTP指令打开远距离轨道对的多极近似，在本章前面的文献[8]有说明。还可以给出更多的选项，说明见上面的LOCAL指令。

LOCAL,MULTP,*options*

等价于

MULTP,*options*

多极近似的级别可以用DSTMLT选项选择（1表示偶极近似，2表示四极近似，等；默认3）。

对于非常大的分子，多极近似可以降低LMP2计算的计算量，但会导致一些额外的误差，参见本章前面的文献[8]。在耦合簇计算中，通常不推荐使用这种方法，也不要用于分子间作用力的计算。它也不能用于几何优化和梯度计算。

### 29.8.3 保存波函 (SAVE)

可以保存波函用于后面的续算，使用

`SAVE,record`

其中`record`具有通常的形式，例如4000.2表示文件2中的记录4000。如果给出这个指令，还会保存域的信息以及振幅（对于MP<sub>n</sub>则不保存振幅）。如果只保存域的信息，必须在LOCAL指令中使用SAVE选项（参见29.4节）。

### 29.8.4 续算 (START)

局域CCSD或QCISD计算可以续算，使用

`START,record`

给出的记录必须已经在前面的局域计算中用SAVE指令保存（否则忽略这个指令）。如果给出了START指令，那么续算使用前一个计算的域信息以及振幅。例如，这可以使用前面局域QCISD计算保存的振幅，开始一个局域CCSD计算（但是，不能使用非局域CCSD或QCISD计算保存的记录）。如果想只使用域信息而不使用振幅进行续算，必须在命令行或LOCAL指令中使用SAVE选项（参见29.4节）。

### 29.8.5 电子的关联子集 (REGION)

在大分子中，只对活性组附近的电子做关联计算已经足够，分子的其余部分只需在SCF级别进行处理。这一方法还可以推广，对体系的不同部分使用不同的关联级别。REGION指令允许指定原子的子集：

`REGION,METHOD=method,[DEFAULT=default_method],  
[TYPE=INCLUSIVE|EXCLUSIVE],atom1, atom2 ...`

位于这些原子的轨道将在`method`指定的级别进行处理。余下的轨道将根据`default`进行处理。如果用户没有给出后一选项，将设定为HF。

可以用两种方法选择轨道。如果`type`设为INCLUSIVE，在域中心列表中含有任何一个原子的轨道将包含在内。如果`type`设为EXCLUSIVE，程序将只添加那些域被给定原子占有的轨道。任何局域关联处理都可以作为`method`给出，而`default_method`限定为只能使用MP2和HF。在一个计算中最多只能包含两个REGION指令，根据对区域指定的关联级别（`method`）进行排序。最高级别的区域应当在最后给出。

建议检查程序打印的区域轨道列表和轨道域。使用区域可能会显著降低计算时间，另外，如果活性组的活性原子选择得合适，仍会给出足够精确的结果，例如，计算键长和键角。

### 29.8.6 域的合并 (MERGEDOM)

在局域计算中，由于依赖于结构的域发生变化，虚空间的限制可能会导致反应路径计算的不连续。这可以用MERGEDOM指令避免

MERGEDOM,[NEIGHBOUR=*value*],[CENTERS=[*atom1, atom2...*]], [RECORD=*...*],CHECK

这个指令提供扩充域，它可以保存（使用SAVE选项或指令，见29.8.3节），用于以后的反应路径或单点计算（轨道域的描述发生不平衡的情况）。*neighbour*选项的使用与局域选项MERGEDOM相同，其中*value*指定共存中心的数量。如果使用了*centres*选项，应当给出原子列表（用方括号扩起）。只有位于这些原子上的所有轨道域才会被合并，产生的合并域将用于所有这些轨道。

还可以给出前一局域计算保存的*record*编号。记录中包含的域列表将与当前列表进行匹配，并对轨道域进行扩充（合并）以包含两套轨道域。因此这个域的定义应当适于两个点（以及两点之间）的计算。这个流程可以重复，以包含更多的结构。用这种方法，可以定义适于整个结构范围（如反应路径）的域。如果这些域用于所有点的计算，可以获得严格平滑的势能面。

### 29.8.7 分子簇计算中的能量分割 (ENEPART)

在局域关联的处理中，占据轨道和虚轨道的局域特性还提供了诱人的功能，可以把分子簇的分子间相互作用能分解为各个不同激发类的贡献。这可以把相互作用能按照关联贡献划分为分子内的、弥散的、以及离子的成分（参看M. Schütz, G. Rauhut and H.J. Werner, J. Phys. Chem. **102**, 5197 (1998)）。能量分割算法可以用ENEPART指令激活：

ENEPART,[*epart*],[*iepart*]

或者在命令行给出选项的参数。

*epart*参数确定（分子内）键长的截断距离（单位a.u.，默认是3 a.u.），用于自动确定团簇内各个单体的亚单位。参数*iepart*如果设为大于零的值（默认是1），则进行能量分割。此外，如果*iepart*设为2，则打印所有分子间轨道对的能量及其分量的列表。

能量分割算法产生的输出部分外观类似于下面的例子：

```
energy partitioning enabled !
centre groups formed for cutoff [au] = 3.00
1   :O1  H11 H12
```

```

2      :02  H21 H22
energy partitioning relative to centre groups:
intramolecular correlation:      -.43752663
exchange dispersion              :      .00000037
dispersion energy                :      -.00022425
ionic contributions              :      -.00007637

```

中心基团对应于  $epart=3$  确定的各个单体。在当前的例子中，发现了两个水单体。关联能分割为上面的四个分量。交换色散、色散以及离子的分量直接反映了复合物中有关分子间的成分，而分子内对相互作用能的关联贡献必须通过超分子计算得到，也就是从输出所给的复合物中的分子内关联成分，减去（两个）相应单体的关联能。

也可以用下面的形式：

```
ENEPART,RMAX=[r1,r2,r3,...]
```

程序将打印在给定距离的范围内（单位Bohr，用方括号扩起，例如，`enepart,rmax=[0,3,5,7,9,11]`），所有轨道对的能量贡献。此外还打印第二个列表，其中的贡献是轨道对域之间成键数目的函数。

## 29.9 正确使用

MOLPRO的局域关联方法使用局域化分子轨道（LMO）。推荐Pipek-Mezey局域化，但也可以用Boys局域化。虚轨道空间用非正交的投影原子轨道（PAO）张开。这个基的局域特性使得可以引入两个不同的近似：第一，激发限制到域（*domains*）中，这是PAO的子空间，非常接近于电子被激发的轨道。第二，轨道对按照其重要性（根据距离或连结性标准）进行分类，只有强轨道对才在最高级别（如CCSD）进行处理。余下的弱轨道对和远距离轨道对在LMP2级别处理，忽略非常远距离的轨道对。这些近似导致计算量随分子尺寸变化的函数呈线性标度。

自然地，这些近似会引入一些误差，因此用户必须比用标准黑箱方法更加小心。另一方面，低级别的标度使得在高理论级别处理的体系比迄今为止所能计算的更大。

这一部分总结了一些局域关联计算需要记住的重点。

### 29.9.1 基组

由于数值的原因，除去投影的芯轨道是非常有用的，因为这些轨道可能有非常小的模。投影芯轨道的模如果小于0.1（这个值可以用DELCO和THRCOR选项改变），默认是除去的。对于局域计算，我们推荐使用一般收缩的基组，例如Dunning及其合作者的关联一致cc-pVnZ基组。对于这些基组，芯基函数是唯一定义的，如果使用了DELCO和THRCOR的默认值，它们将总是会被除去。

还推荐把关联一致基组用于所有密度拟合计算，因为对每个基有优化好的拟合基组。

### 29.9.2 对称性和方位

**1. 关闭对称性!** 否则，你不会得到合适的局域化轨道（局域轨道将趋向于对称等价而不是对称匹配）。只有当所有的原子是对称唯一时，才可以使用对称性。这可以局域处理 $C_s$ 对称性的平面分子。但是注意，多极程序和密度拟合程序根本不支持对称性，因此对于DF计算，或者用MULTP选项的计算，总是要使用 $C_1$ 对称性。

要关闭对称性，只需要在结构输入的第一行指定NOSYM，例如

```
geometry={
  nosym
  01
  H1,01,roh
  H2,01,roh,h1,hoh
}
```

也可以在结构区域之前加上

SET,ZSYMEL=NOSYM

**2. 使用NOORIENT!** 我们推荐在结构输入中使用NOORIENT选项，当结构改变时避免分子不需要的转动。对于几何优化，以及相互作用能计算中的域的续算，这特别重要（参见29.9.8节）。

### 29.9.3 局域化

默认使用Pipek-Mezey局域化，在局域关联计算的开始自动执行。因此

```
df-hf      ! 使用密度拟合的Hartree-Fock
df-lmp2    ! 使用Pipek-Mezey LMO的LMP2
```

等价于

```
df-hf      ! 使用密度拟合的Hartree-Fock
locali,pipek ! 使用Pipek-Mezey标准的轨道局域化
df-lmp2    ! 使用Pipek-Mezey LMO的LMP2
```

也可以用Boys局域化，但是在这种情况下，局域化必须提前完成，例如，

```
df-hf      ! 使用密度拟合的Hartree-Fock
locali,boys ! 使用Boys标准的轨道局域化
df-lmp2    ! 使用Boys LMO的LMP2
```

差的局域化有时是体系的内部问题，特别是对强共轭体系，或者使用弥散基组时。这是由重叠弥散函数的局域化尾数所导致。在计算含有短键长的体系时，如芳香分子，这个问题会经常出现。避免这个问题可以用指令

PIPEK,DELETE= $n$

其中 $n = 1$ 或 $2$ 。这意味着在局域化中忽略每个角动量的 $n$ 个最弥散基函数的贡献。当使用了弥散基组时，这通常会产生更好的局域化轨道。对于aug-cc-pVTZ，发现 $n = 2$ 较好；而对aug-cc-pVDZ， $n = 1$ 较好。

在极少数情况下，还可能会发生局域化流程不收敛。这种情况可以选用二阶Newton-Raphson局域化方案，使用的指令为

PIPEK,METHOD=2,[DELETE= $n$ ]

还可以使用（推荐）：

PIPEK,METHOD=3,[DELETE= $n$ ]

它首先执行一些标准的Pipek-Mezey迭代，接下来调用二阶局域化方案。它的收敛通常非常快。

#### 29.9.4 轨道域

用Boughton and Pulay, J. Comput. Chem., **14**, 736 (1993)和J. Chem. Phys. **104**, 6286 (1996)的流程自动确定轨道域。为了更高的精度，还可以对域进行扩展，在这种情况下可以系统地逼近正则结果（参见本章前面的文献[1]和29.6.2节）。细节在29.6节讲述。

对于饱和分子，域的选择在大多数情况下并不敏感。然而，特别是对离域体系，推荐总是检查轨道域，它在每个局域计算的开始打印。做这种检查可以用DOMONLY=1选项，在域产生后就停止计算。轨道域由原子子集的全部基函数构成。选择这些原子使得域按照预定的精度（可用选项THRBP修改）张开相应的局域化轨道。对于水，典型的域输出就像这样：

##### Orbital domains

Orb.	Atom	Charge	Crit.
2.1	1 O1	1.17	0.00
	3 H2	0.84	1.00
3.1	1 O1	2.02	1.00
4.1	1 O1	1.96	1.00
5.1	1 O1	1.17	0.00
	2 H1	0.84	1.00

这就告诉你轨道2.1和5.1的域由氧原子和一个氢原子的基函数组成，而轨道3.1和4.1的域只由氧原子的基函数组成。后者对应于氧的孤对电子，前者对应于两个OH键，因此这正是我们期待的。对于AO的每个域产生相应的投影原子轨道（PAO），它张开虚空间的子空间，在其中产生激发。影响域选择的选项在29.6节讲述。不合适的域可能由较差的局域化轨道（参见29.9.3节）或忘记加NOSYM指令导致。这甚至会负面地影响计算执行和内存需求，还可能导致意外的结果。

选择标准THRBP默认为0.98。对于小基组，像cc-pVDZ，这通常很好。而对于大一些的基组，像cc-pVTZ，我们推荐使用略大一些的值0.985，以保证有足够的原子包含在每个域中。对于cc-pVQZ，推荐THRBP=0.990。遇到拿不准的情况，可以把得到的域与较小基组（如cc-pVDZ）的域做比较。

对于近平衡点的性质，如，平衡构型和谐振频率，域选择的影响很弱。影响较大的是能量差，如反应能量和能垒高度。对于电子结构有较大变化的情况，如双键个数改变，建议在执行昂贵的LCCSD(T)计算前先比较DF-LMP2和DF-MP2的结果。更平衡的结果和平滑的势可以用MERGEDOM指令获得，参见29.8.6节。

### 29.9.5 冻结域

为了获得平滑的势能面，必须冻结域。域的信息可以用SAVE选项存储，用START选项恢复。另外也可以用SAVE和START卡，参见29.8.3节。对于后一种情况，还保存/续算CCSD振幅。冻结域对于分子间相互作用的计算特别重要，见29.9.8节。可以用MERGEDOM指令产生适于大结构范围的域，如反应路径，见29.8.6节。在几何优化和频率计算中，自动冻结域，见29.9.7节。但是注意，这一自动流程只能用于在优化中调用单个局域化的计算。对于补偿校正的优化，复合物和每个单体的域必须在不同的记录中，用SAVE和START指令分别冻结。

### 29.9.6 轨道对分类

强，封闭，弱，和远距离轨道对的选择使用距离标准或连接性标准，细节在29.7节描述。强轨道对用CCSD处理，其它轨道对用LMP2。在三激发计算中，包含所有轨道的三激发( $ijk$ )，其中( $ij$ )，( $ik$ )，和( $jk$ )是封闭轨道对。此外，这些轨道对中必须有一个强轨道对。三激发能量依赖于强轨道对和封闭轨道对的振幅。封闭轨道对的振幅来自LMP2计算。因此，对封闭轨道对和弱轨道对增加距离标准或连接性标准，将会得到更准确的三激发能量。对于结构和谐振频率一类的近平衡结构特性，默认值通常合适，在计算能量差时，特别是对能垒高度，有时会需要更大的距离标准。对于拿不准的情

况，应当首先增加RWEAK直到收敛，接下来可以修改RCLOSE。这种测试可以使用较小的基组，如cc-pVDZ，获得的优化值接下来用于大基组的最终计算。

### 29.9.7 梯度和频率计算

可以对局域MP2使用解析能量梯度[15-17]进行结构优化[15-17]和数值频率[18-20]计算。对于弱束缚体系，LMP2结构优化特别引人注目，因为事实上得到的是无BSSE的结构（参见29.9.8节和本章前面的文献[21-23]）。由于效率的原因，对所有结构优化强烈建议使用DF-LMP2 梯度[17]。在DF-LMP2命令或DFIT指令中设定SCSGRD=1会激活关于Grimmes SCS换算的MP2能量泛函的梯度（另外参见DFIT部分）。解析能量梯度还不能用于远距离轨道对的多极方法，因此MULTP不能用于结构优化和频率计算。

在结构优化中，最开始的优化步骤允许改变域。当步长小于某个阈值后（默认0.01），自动冻结域。在数值Hessian或频率计算中也冻结域。因此不需要包含SAVE和START选项。

在优化高对称的芳香体系时，如苯，要特别当心。在 $D_{6h}$ 对称性中， $\pi$ 轨道的局域化不是唯一的，也就是说，局域化轨道可以绕 $C_6$ 轴转动而无需改变局域化标准。如果对称性略微变形，这种冗余就会消失，将导致局域化轨道的突然改变。如果现在用SAVE和START选项使域保持固定，就会导致较大的能量误差。另一方面，如果域不保持固定，它们的尺寸和性质就会在优化过程中改变，仍会导致错误的能量差和优化的偏差。

避免这一问题的最佳方法是使用MERGEDOM=1选项（参见29.6节）。如果给出这个选项， $\pi$ 轨道的域将会由所有六个碳原子的基函数组成，能量关于三个 $\pi$ 轨道的酉变换是不变的。注意，如果芳香体系的对称性被取代基降低，这个问题就不会发生。

冗余轨道的转动也会导致Pipek-Mezey局域化的收敛困难。它的解决可以用选项  
PIPEK,METHOD=2  
或  
PIPEK,METHOD=3

用METHOD=2，将计算局域化标准关于轨道转动的二阶导数，并进行对角化，对应于零本征值的转动将被排除。这一方法是二次收敛的。用METHOD=3，前几个迭代用标准的Pipek-Mezey方法执行，接下来调用二阶方法。这是最灵活最准确的局域化方法。

最后，我们指出，LMP2梯度对SCF收敛的精度非常灵敏（MP2也是如此）。如果需要非常准确的结构，或者从梯度计算数值频率，那么默认的SCF精度可能不够。在这些情况下，我们建议在HF命令后加上指令ACCU,14（甚至可以是ACCU,16）。SCF精度不够的标志是在结构优化即将结束时，有小的正能量变化。



### 29.9.8 分子间相互作用

对于分子间弱相互作用的计算，局域方法特别有用，因为基组重叠误差被大大降低[1,13,14]，通常不需要补偿校正（假设在该方法下Hartree-Fock的BSSE很小）。但是，必须当心域的定义是否合适，在最高计算级别必须包含所有的分子间轨道对。定义合适的域和轨道对列表的方便方法是用INTERACT=1选项。如果给出该选项，程序自动识别各个分子，所有的分子间轨道对自动处理为强轨道对，并包含在LCCSD中。同时，还对LCCSD(T)计算产生合适的三激发列表。这需要所有的轨道域位于各自的分子上。但是注意，包含分子间轨道对会极大地增加强轨道对和三激发的数量，因此高级别计算可能会变得很昂贵。

要计算弱相互作用体系的相互作用势，应当在非常大的距离下确定子体系的域，并在LOCAL或MULTIP指令中用SAVE=record选项，或者用SAVE命令保存（参见29.8.3节）。如果不需要渐进能量，用DOMONLY=1选项作初始计算就足够了。这些域可以用于随后的其它分子间距的计算，使用START=record选项或START指令（参见29.8.4节）。只有用这种方法，基组重叠误差才最小，通常可以忽略（当然这不会影响SCF的BSSE，因此基组应当足够大使得SCF的BSSE可以忽略）。

通常，弥散基函数对于获得精确的分子间相互作用很重要。当使用弥散基函数时，Pipek-Mezey局域化有可能无法很好地产生局域化轨道。在大多数情况下，避免这一问题可以用指令

```
PIPEK,DELETE=n
```

说明见29.9.3节。

最后的警告是关于局域密度拟合（参见29.10节和十五章）：千万不要在补偿计算中使用局域拟合，因为在虚原子上不应当出现拟合函数，这会导致极大的错误。

关于这些方面的例子和讨论，参见本章前面的文献[21-23]。

## 29.10 密度拟合LMP2（DF-LMP2）和耦合簇（DF-LCCSD(T0））

密度拟合LMP2和LCCSD计算可以用命令名加上前缀DF-来执行。输入如下：

```
DF-LMP2,[options]
```

```
DF-LCCSD(T),[options]
```

密度拟合的选项可以与LOCAL的任何选项混合。密度拟合的选项也可以在DFIT指令给出（见十五章）。

在局域方法中，最重要的密度拟合选项有

```
BASIS_MP2=string
```

LMP2和LCCSD使用的拟合基组，用于直到2-外部轨道的积分。如果使用了关联一致基组（如cc-pVTZ），默认使用MP2的拟合基组（cc-pVTZ/MP2FIT）。否则拟合基组必须在前面的基组区定义（参见十一章）。

**BASIS\_CCSD**=*string*

在LCCSD中对3-、4-外部轨道积分使用的拟合基组。默认为**BASIS\_MP2**，通常已经足够。但是，在LCCSD中4-外部轨道积分的精确方法需要比LMP2更大的拟合基组。因此，为了使拟合误差最小，建议使用下一个更大的拟合基，例如，对VTZ轨道基组使用**BASIS\_CCSD=VQZ**。

**LOCFIT**=*value*

如果**LOCFIT**=1，使用局域拟合。这对获得DF-LMP2的线性标度是必要的（参见本章前面的文献[11-14]）。局域拟合引入的误差通常很小，但有些例外。例如，**LOCFIT**=1不要用于补偿计算，参见29.9.8 节。注意对于小分子，**LOCFIT**=1可能比**LOCFIT**=0更昂贵。

关于密度拟合的更多细节和选项，参见第十五章。

显关联计算可以显著改善MP2和CCSD关联能的基组收敛性。实现这种计算可以用以下形式的命令：

*command, options*

其中的**command**可以是以下之一：

- MP2-F12** 闭壳层正则MP2-F12。默认使用固定振幅解（**FIX**，见下），但也可以用其它解。用密度拟合计算F12校正，并把它加到不用密度拟合得到的MP2关联能上。
- DF-MP2-F12** 同MP2-F12，但使用DF-MP2关联能。它不如MP2-F12昂贵。
- DF-LMP2-F12** 使用局域轨道的闭壳层DF-MP2-F12。可以使用在J. Chem. Phys. 126, 164102 (2007) 中描述的方法和解（参见30.2和30.7节）。
- DF-RMP2-F12** 使用3C解的自旋限制性开壳层DF-RMP2-F12。可以使用在J. Chem. Phys. 128, 154103 (2008) 描述的方法（参见30.2和30.7节）。
- CCSD-F12** 闭壳层CCSD-F12近似，说明见J. Chem. Phys. 127, 221106 (2007)。默认使用固定振幅解，并计算CCSD-F12A和CCSD-F12B能量。还可以为命令加上**A**或**B**后缀，这时只计算相应的能量。更多细节参见30.9节。
- CCSD(T)-F12** 同CCSD-F12，但加入了微扰三激发。
- UCCSD-F12** 开壳层非限制UCCSD-F12方法，说明见G. Knizia, T. B. Adler, and H.-J. Werner, 130, 054104 (2009)。使用限制性开壳层Hartree-Fock（RHF）轨道。还可以为命令加上**A**或**B**后缀，这时只计算相应的能量。更多细节参见30.9节。

UCCSD(T)-F12 同UCCSD-F12, 但加入了微扰三激发。

RCCSD(T)-F12 同UCCSD(T)-F12, 但使用了部分自旋匹配方案。

使用这些方法的出版物应当引用以下文献:

F. R. Manby, J. Chem. Phys. **119**, 4607 (2003)

(线性R12方法中的密度拟合近似)

A. J. May and F R Manby, J. Chem. Phys. **121**, 4479 (2004)

(用于冻结杂函数展开式)

H.-J. Werner and F R Manby, J. Chem. Phys. **124**, 054114 (2006);

F. R. Manby, H.-J. Werner, T. B. Adler and A. J. May, J. Chem. Phys. **124**, 094103 (2006);

H.-J. Werner, J. Chem. Phys. **129**, 101103 (2008);

T. B. Adler, F. R. Manby and H.-J. Werner, J. Chem. Phys. **130**, 054106 (2009);

(用于DF-LMP2-F12)

H.-J. Werner, T. B. Adler, and F. R. Manby, J. Chem. Phys. **126**, 164102 (2007)

(用于所有其它的闭壳层MP2-F12方法)

G. Knizia and H.-J. Werner, J. Chem. Phys. **128**, 154103 (2008)

(用于所有开壳层F12计算)

T. B. Adler and G. Knizia and H.-J. Werner, J. Chem. Phys. **127**, 221106 (2007)

(用于CCSD(T)-F12)

G. Knizia, T. B. Adler, and H.-J. Werner, J. Chem. Phys. **130**, 054104 (2009)

(用于CCSD(T)-F12和UCCSD(T)-F12计算)

T. B. Adler and H.-J. Werner, J. Chem. Phys. **130**, 241101 (2009)

(用于LCCSD-F12)

K.A. Peterson and H.-J. Werner, J. Chem. Phys. **128**, 084102 (2008)

(用于VnZ-F12基组)

K. E. Yousaf and K. A. Peterson, J. Chem. Phys. **129**, 184108 (2009)

(用于VnZ-F12/OPTRI基组)

K. E. Yousaf and K. A. Peterson, Chem. Phys. Lett., **476**, 303 (2009)

(用于AVnZ/OPTRI基组)

下面我们简要总结了可用的解和近似方法。关于其他作者有关工作的更多细节和参考文献, 参见: H.-J. Werner, T. B. Adler, and F. R. Manby, *General orbital invariant MP2-F12 theory*, J. Chem. Phys. **126**, 164102 (2007) (在以后简写做I)。

### 30.1 参考函数

MP2-F12, CCSD-F12, 和UCCSD-F12必须使用通常的（非密度拟合的）自旋限制Hartree-Fock 参考函数（HF或RHF）。DF-HF不能用于这些方法。这一限定是为了保证Fock矩阵是对角的，并与这些方法所用的积分一致。对于DF-MP2-F12, DF-LMP2-F12, 和DF-RMP2-F12, HF或DF-HF 参考函数都可以使用。

目前只能使用有限偶极场，不支持其它的微扰。可以用ECP，但这目前仍是经验性的，未做充分的基准性测试。Douglas-Kroll-Hess哈密顿量不能与F12方法一起使用。

### 30.2 波函的解

所谓的“解”决定了显关联波函的定义。这是为了与近似哈密顿矩阵元中的各种近似区分开。我们一般使用解3（参见I），其中的投影算符为

$$\hat{Q}_{12} = (1 - \hat{o}_1)(1 - \hat{o}_2)(1 - \hat{v}_1\hat{v}_2) \quad (30.1)$$

其中的 $\hat{o}_i$ 是电子 $i$ 到占据空间的单电子投影算符， $\hat{v}_i$ 用来投影到虚轨道空间。在局域显关联波函中使用域近似的情况下，用算符 $\hat{d}^{ij}$ 替换算符 $\hat{v}$ ，它会仅仅投影到轨道对 $ij$ 的域上。

在MOLPRO中，可以使用以下的波函的解：

#### 30.2.1 一般解

通常的外部轨道对函数用下面形式的项进行扩充：

$$|u_{ijp}^{F12}\rangle = \sum_{p=\pm 1} \sum_{kl} T_{kl}^{ijp} \hat{Q}_{12} \hat{F}_{12} |kl\rangle \quad (30.2)$$

这个解具有轨道不变性（也就是说，用正则和局域轨道得到的结果相同），但经常会遇到基组的重叠误差。此外，在零阶哈密顿量可能会发生奇异性，特别是对较大的体系。因此，通常不推荐这种解。

#### 30.2.2 对角的解（D）

在方程30.2中，对 $kl$ 的求和限制为到 $ij$ 。这种解不具有轨道不变性，只有当使用局域轨道时才有大小一致性。但是，它没有基组的重叠误差，因此通常比用一般解更准确。

### 30.2.3 固定振幅的解 (FIX)

使用对角解，显关联组态的振幅由波函的尖端点条件 (cusp condition) 决定，即，

$$T_{ij}^{ij,1} = 1/2 \quad (30.3)$$

$$T_{ij}^{ij,-1} = 1/4 \quad (30.4)$$

解具有轨道不变性，大小一致性，而且没有GBSSE。

## 30.3 RI近似

对于一阶哈密顿量矩阵元，存在各种近似，如A, B, C, HY1, HY2 (参见文献I)。它们的区别在于RI近似的方法不同。在完全RI基组的极限，B和C近似是等价的，对给定的波函解产生精确的结果。我们通常推荐近似C，它比近似B更简单更高效。通常，AO和RI基组联合用于对恒等解做近似 (CABS方法)。在混合方法 (HY1, HY2, HX) 中，对有些不太重要的项仅使用AO基。HY1或HY2可以和推荐的近似C一起使用；HY2更准确，HY1更高效。在大多数情况下，近似3C(HY1)能得到在精度和效率之间的最佳折中。在近似A中，忽略所有涉及交换算符的项。在我们的低级别标度LMP2-F12/3\*A(loc)方法中，这一近似和局域近似一起使用，可用于大分子 (参见30.10节)。

如果假定了扩展Brillouin条件 (EBC, 参见文献I)，那么显关联振幅方程和常规振幅方程是解耦合的，可以独立求解。这些近似用星号表示，例如3\*C。

## 30.4 基组

在MOLPRO中，F12积分只能用密度拟合 (DF) 近似计算。多电子积分用恒等解 (RI) 的展开近似。因此，F12计算需要三种不同的基组：轨道基 (AO)，DF基，和RI基。

我们推荐的AO基是缀加的关联一致基组 (表示为AVnZ) 或者特别优化的关联一致F12基组 (表示为VnZ-F12, 参见K.A. Peterson and H.-J. Werner, J. Chem. Phys. **128**, 084102 (2008))。通常三ζ 基组 (AVTZ或VTZ-F12) 产生接近于基组极限的非常好的结果。弥散基函数对于HF和MP2-F12能量非常重要，因此不推荐标准的VTZ基组。如果使用了AVnZ或VnZ-F12轨道基组，会自动选择合适的密度拟合 (DF) 基和恒等解 (RI) 基组。AVnZ/MP2FIT和VnZ/JKFIT默认分别用于DF和RI。其它的基组可以用DF\_BASIS, DF\_BASIS\_EXCH, 和RI\_BASIS选项进行选择 (参见30.6节)。关于密度拟合的更多细节，参见第十五章。

### 30.5 对称性

在DF-LMP2-F12计算中不能使用对称性。不过，在MP2-F12，DF-MP2-F12，DF-RMP2-F12，CCSD(T)-F12，和UCCSD(T)-F12计算中，可以像往常一样使用阿贝尔对称性；在这些情况下，最前面的DF-MP2-F12 计算的执行自动不用对称性，而对随后CCSD-F12或UCCSD-F12计算必须的积分，将转换到对称匹配基中。这对用户是完全自动和透明的。但是注意，前缀DF-自动关闭对称性，如果想在DF-MP2-F12或DF-MP2-F12之前的HF计算中使用对称性，那么要么在geometry区指定对称元（或AUTO），要么设定ZSYME1（设定zsymel=auto可以运行得很好）。

### 30.6 选项

有很多可用的选项，但是很少使用。通常，当使用了诸如aug-cc-PVTZ或VTZ-F12的标准轨道基组时，将使用合适的默认值，不需要更多选项。简单的输入仅仅写做

```
basis=vtz-f12
```

```
hf
```

```
ccsd(t)-f12
```

只有当必要时，以及只有当用户充分了解他们所做的一切时，我们才推荐指定选项！

正则和局域版的选项：

```
DF_BASIS=basis
```

选择密度拟合的基组（细节参见第十五章）。*basis*既可以是基组区定义的基组名，也可以是默认的MP2拟合基（例如，DF\_BASIS=VTZ产生VTZ/MP2FIT基）。默认使用对应于轨道基的MP2FIT基。

```
DF_BASIS_EXCH=basis
```

选择密度拟合基用于计算交换和Fock算符。默认使用对应于轨道基组的JKFIT基组。

```
RI_BASIS=basis
```

为恒等解（RI）选择基组。这可以是默认基组，或者设定为在基组区定义的名称。对于R12方法，这应当选为大的非收缩AO基（至少是AVQZ）。对于F12方法，Hartree-Fock JKFIT基组对RI的表现很好，尽管为其它的目的作了优化。基组名称可以在斜线之后加上基组类型，例如

RI\_BASIS=AVQZ/ORBITAL

将使用非收缩的AVQZ轨道基组，

RI\_BASIS=VQZ/JKFIT

将使用JKFIT基组。对于R12方法，默认的类型为ORBITAL和UNCONTRACT，而对F12方法是JKFIT。可以为基组名加上(CONTRACT)或(UNCONTRACT)后缀强制使用收缩或非收缩基组。例如，

RI\_BASIS=AVQZ(UNCONTRACT)/ORBITAL

如果在括号中还给出了其它选项，那么用逗号分开，例如，

RI\_BASIS=AVQZ(f/d,UNCONTRACT)/ORBITAL

另外的形式是

RI\_BASIS=AVQZ(f/d)(UNCONTRACT)/ORBITAL

或

RI\_BASIS=AVQZ(f/d)/ORBITAL(UNCONTRACT)

注意，(CONTRACT)/(UNCONTRACT)选项不能用于在前面基组区定义的基组名。

CONTEXT=*context*

可用于改变RI基的默认类型，例如，CONTEXT=OPTRI将使用OPTRI基组，它对应于VnZ-F12或AVnZ基组。

ANSATZ=*ansatz*

选择显关联解*ansatz*的方法。可用选择和更多细节参见30.7节。

GEM_BASIS	孪函数展开的基组名；忽略原子标记。它可以是OPTFULL（孪函数展开的完全非线性拟合），EVEN（平缓拟合），或者是前面BASIS区定义的基组名。默认是OPTFULL。
GEM_TYPE	冻结孪函数类型：LINEAR或SLATER，默认是SLATER。
GEM_NUMBER	Gaussian孪函数数量（默认6）。
GEM_CENTRE	如果GEM_BASIS=EVEN，平缓孪函数指数的中心（默认1.0）。
GEM_RATIO	如果GEM_BASIS=EVEN，平缓孪函数指数的比率（默认3.0）。
GEM_BETA	Slater型冻结孪函数的指数，或者在其它冻结孪函数模型中加权函数的参数（默认1.0 $a_0^{-1}$ ）。
GEM_OMEGA	加权函数的指数（默认-1，表示从GEM_BETA得到的值）。



GEM_MOM	在omega拟合中，r的指数（默认0）。
GEM_M	在加权函数中，r的指数（默认0）。
GEM_MAXIT	在乘函数优化中的最大迭代数（默认200）。
GEM_PRINT	乘函数优化的打印参数（默认0）。
GEM_DEBUG	乘函数优化的调试选项（默认0）。
GEM_ACC	用于乘函数线搜索的收敛阈值（默认0.001）。
GEM_FAC	在乘函数优化中，指数的换算因子（默认1.0）。
GEM_METHOD	乘函数优化方法[扩充Hessian（AH）或Newton-Raphson（NR）]，默认是（AH）。
GEM_TRUST	AH乘函数优化的可信比率（默认0.4）。
GEM_SHIFT	AH乘函数优化中的Hessian移动（默认0）。
GEM_NUMERICAL	乘函数优化中的数值积分标记（默认0）。
GEM_PLOT	乘函数绘图文件（默认为空）。
PRINT= <i>ipri</i>	选择输出级别： <i>ipri</i> =0：标准输出 <i>ipri</i> =1：标准输出加关于积分求解的更多详细信息 <i>ipri</i> =2：调试输出
THRBINV	低于这个阈值的非物理本征值，将从近似B矩阵投影掉。
THRF12	积分屏蔽贡献的阈值。

### 30.7 选择近似的解和级别

可以用ANSATZ选项选择解，或者从命令行选项选择解。

3A	解3A。
3*A	使用EBC近似的解3A。
3B	解3B。
3*B	使用EBC近似的解3B。
3C	解3C。
3*C	使用EBC近似的解3C。

可以在括号中为解加上更详细的选项，例如

ANSATZ=3B(D)

这些选项可以是以下之一：

D	使用对角解。
FIX	使用固定振幅的对角解（具有轨道不变性）。
FIXC	使用固定振幅和正则轨道的对角解。
DX	使用对角解，并假定X矩阵是对角的（仅用于3A解）。
GBC	使用GBC近似（仅用于3B解，在3A解中是默认的）。
EBC	使用EBC近似（同*）。
HX1	使用HX1近似（仅用于3B）。
HY1	使用HY1近似（仅用于3B和3C）。
HY2	使用HY2近似（仅用于3B和3C）。
HY	默认的混合近似。在3B中是HX1和HY2近似，在3C中是HY2。
NOZ	忽略Z项（仅用于3B和3C）。
NOX	忽略X项（仅用于3A和3B）。

可以给出用逗号隔开的多个选项。例如，

ANSATZ=3C(FIX,HY1)

使用固定因子和混合（HY1）近似的对角解3C(D)。

此外，还可以在命令行给出以下选项：

DIAG=1	使用对角解。
DIAGX=1	使用对角解，并假定X矩阵是对角的。
GBC=1	使用GBC近似（仅用于3B解，在3A解中是默认的）。
EBC=1	使用EBC近似（同*）。
HYPRIID= <i>n</i>	使用HY <sub>n</sub> 近似。
HYPRIIDX=1	使用HX1近似。
NOZ=1	忽略Z项（仅用于3B和3C）。
NOX=1	忽略X项（仅用于3A和3B）。
FIX=1	使用固定因子近似的对角解（具有轨道不变性）。
FIX=2	使用固定因子近似的对角解。仅求解一阶能量表达式，不求解Hylleraas泛函。非常快，但精度和可靠性较差。
FIXCAN=1	使用固定因子近似和正则轨道的对角解。用非迭代方法求解能量。这等价于FIX=1,CANONICAL=1，是最高效的方法。

FIXCAN=-1	同FIXCAN=1, 但迭代求解方程 (仅用于测试的目的)。
CABS=1	使用CABS (默认)。
ORTHO_CABS=1	从正交MO和ABS基构造CABS基, 而不是从AO和RI基。
THRABS= <i>thrabs</i>	在辅助的ABS中, S最小本征值的阈值 (只能与ORTHO_CABS=1一起使用; 默认THRCABS)。
THRCABS= <i>thrcabs</i>	在CABS中, S最小本征值的阈值 (默认1.d-8)。
THRCABS_REL= <i>thrcabs_rel</i>	相对CABS阈值 (默认1.d-9)。实际的阈值为 $\max(\text{thrcabs}, \text{eigmax} * \text{thrcabs\_rel})$ , 其中 $\text{eigmax}$ 是重叠矩阵最大的本征值。
PRINT= <i>level</i>	打印参数。PRINT=1给出计算中所有积分和迭代的信息。
DEBUG= <i>level</i>	用于获得更多的调试信息。
SOLVE=0	使用最有效的特定对完全迭代方法 (默认)。
SOLVE=1	简单地使用完全迭代方法。
SOLVE=2	使用特定对的迭代方法 (更昂贵)。
SOLVE=3	使用特定对的非迭代方法 (最昂贵, 只能用于正则轨道)。
CANONICAL=1	使用正则轨道和完全域。
DOMSEL=1	使用完全域和局域轨道 (除非给出了CANONICAL=1)。
SCALE_TRIP=1	对三激发能量进行换算, 解释见30.9节。
SINGLES	如果设为1, 还包含CABS单激发校正 (默认=1)。
CORE_SINGLES	如果设为1, 还包含芯轨道的CABS单激发校正 (默认=0)。
EXTGEN	对于开壳层体系: 如果是1 (推荐的默认值), 那么对于 $T_{mn}^{ij}$ 中的 $mn$ 包含所有的占据价轨道对, 并且与自旋无关 (说明见J. Chem. Phys. 130, 054104 (2009), II.E节)。如果是0, 那么仅使用 $mn$ 轨道对, 其中 $i$ 和 $m$ 以及 $j$ 和 $n$ 的自旋是相同的。

例如,

```
ANSATZ=3C,fix=1,hybrid=1,canonical=1
```

意味着使用对角解3C、固定因子和混合近似的正则3C计算。选项fix=1和canonical=1的组合意味着非迭代计算能量, 是推荐方法。以上输入等价于下面的所有输入:

```
ANSATZ=3C(FIXC, HY1)
```

```
ANSATZ=3C(D, FIXC, HY1)
```

```
ANSATZ=3C(D, HY1, FIX), canonical=1
```

注意, HF收敛阈值需要非常严格, 以获得准确结果 (在HF中使用ACCU, 14)。

还有更多的选项仅用于专家，这里不再讲述。完整的列表参见`explicit.registry`。

### 30.8 CABS单激发校正

在所有的MP2-F12和CCSD-F12计算中（闭壳层和开壳层，LMP2-F12/3\*A(loc)除外，它用另外的程序执行），微扰CABS单激发校正默认包含在参考能量中，说明见J. Chem. Phys. **127**, 221106 (2007)和J. Chem. Phys. **128**, 154103 (2008)。校正后的参考能量存储在变量ENERGR中，因此总的关联能为ENERGY-ENERGR。对于F12程序其它变量的设定，参见30.11节。

可以用选项SINGLES=0关闭单激发校正，例如：

MP2-F12,SINGLES=0

默认不包含芯轨道对单激发能的贡献，但是可以用CORE.SINGLES选项打开，例如，

MP2-F12,CORE.SINGLES=1

但是，我们不推荐使用芯的单激发，因为它们对CABS基的结构依赖性很灵敏，不会对相对能量有显著改善。

### 30.9 CCSD(T)-F12

CCSD-F12和UCCSD-F12程序首先进行DF-MP2-F12/3C(FIX)（闭壳层）或DF-RMP2-F12/3C(FIX)（开壳层）计算，接下来执行没有密度拟合的CCSD-F12（或UCCSD-F12）计算。CCSD-F12A和CCSD-F12B能量默认都被计算。可以为-F12后缀加上A或B来请求特定的方法。此外，除了3C(FIX)解，还可以用不同的解（如3C）。在这种情况下，显关联项的振幅在MP2-F12计算中确定，并在CCSD-F12中保持固定。

应当注意，这些方法包含了近似，不会产生精确的CCSD-F12能量。初步的经验显示，CCSD-F12A方法略微高估了相关能，而CCSD-F12B低估了相关能。对于AVDZ或AVTZ基组，CCSD-F12A通常给出非常好的结果，但对更大的基组可能会高估基组极限，从下面收敛到极限。因此收敛可能不是单调的，不要做关联能外推。CCSD-F12B通常从下面单调收敛到极限。因此，对于AVDZ和AVTZ基组，我们目前推荐CCSD-F12A，对更大的基组（很少用到）推荐CCSD-F12B。

微扰三激发校正可以用CCSD(T)-F12或UCCSD(T)-F12调用。对三激发没有直接的F12校正。因此三激发的基组误差不受F12的影响（三激发小的能量变化是由于双激发振幅受F12项的影响）。在很多情况下，改善三激发能量的简单实用办法是换算三激发能量的贡献，

$$\Delta E_{(T^*)} = \Delta E_{(T)} * E_{corr}^{MP2-F12} / E_{corr}^{MP2} \quad (30.5)$$

这可以通过设定SCALE\_TRIP=1选项自动完成，即  
CCSD(T)-F12,SCALE\_TRIP=1

### 30.10 使用局域近似的DF-LMP2-F12计算

可以对3\*A(loc)解使用命令DF-LMP2-F12，来调用DF-MP2-F12方法的局域变体。这些计算用和非局域计算不同的程序执行，目前不包含微扰CABS单激发校正（因为在这个程序中不使用CABS）。(loc)选项意味着使用域近似执行LMP2计算，默认使用局域投影，描述见H.-J. Werner, J. Chem. Phys. (submitted, 2008)。这会在低计算量的情况下，产生非常类似于使用完全域的DF-LMP2-F12/3\*A(D,NOX) 方法的能量，可用于非常大的分子的计算。

局域投影算符也可以用于其它的解，例如通过指定ANSATZ=3C(loc)，但是在这种情况下，仅模拟局域投影，但不节省时间。是否使用局域投影算符，依赖于选项USEPAO，见下。

用于这些局域变体的特殊选项（局域RI仅用于3\*A解），有：

PAIRS	指定哪些对用R12或F12处理（STRONG CLOSE WEAK ALL；包含直到给定级别的对）。默认为ALL。注意，如果在LMP2忽略了非常远距离的对，那么即使使用了ALL，在这个计算中也会忽略非常远距离的对。
USEVRT	如果为0，使用 $1 - pp + po + op - p'o - op'$ 形式的投影算符，并把局域RI近似应用到 $p$ 和 $p'$ 上。若设定为1，那么使用 $1 + oo - vv - p'o - op'$ 形式的投影算符；局域RI近似仅应用于RI的贡献 $p'$ 。
USEPAO	如果USEPAO=1，使用特定对的局域投影算符而不是 $vv$ 。如果解包含“(LOC)”或者在LMP2中对于域做了近似的话（即，直接指定DOMSEL<1），那么这是默认的。否则，默认为USEPAO=0。USEPAO=1自动意味着USEVRT=1，即，局域RI近似只影响RI的贡献 $p'$ 。此外，如果指定了USEPAO=1并且未给出DOMSEL，那么在LMP2中假定默认的域。如果USEPAO=0，将使用完全域（DOMSEL=1）。
FULLAO	如果USEVRT=0和FULLAO=1，那么局域RI近似只用于RI的贡献 $p'$ 。这会给出和USEVRT=1 相同的结果（仅用于USEPAO=0）。

DEBUG	调试打印的参数。
LOCFIT_F12	如果设为1, 使用局域拟合。默认无局域拟合 (LOCFIT_F12=0)。
LOCFIT_R12	LOCFIT_F12的别名。在R12计算中, 不推荐局域拟合。
FITDOM	如何确定本底拟合域 (仅用于LOCFIT_F12=1): 0: 拟合域基于联合算符域; 1: 拟合域基于轨道域 (默认); 2: 拟合域基于使用强对的联合对域; 3: 拟合域基于使用强、封闭和弱对的联合对域。注意: 在DF-LMP2程序中, 这是唯一使用的选项。因此如果用默认值, DF-LMP2和DF-LMP2-F12程序可能会给出略微不同的结果。
RDOMAUX	在局域拟合中, 用于密度拟合域扩展的距离标准。默认依赖于FITDOM。
IDOMAUX	在局域拟合中, 用于密度拟合域扩展的连接性标准。
RAODOM	用于RI域扩展的距离标准。零表示完全RI基 (默认)。如果USEPA0=1或USEVRT=1或FULLA0=1, 那么推荐5 Bohr的值。在其它的情况下, 局域RI域必须非常大 (RAODOM>12), 不推荐使用局域RI近似。
IAODOM	用于RI域扩展的连接性标准。零表示完全RI基 (默认)。如果使用了局域投影算符 (USEPA0=1), 那么大于等于2的值可以给出足够精确的结果。
THRA0	在AO或RI基中的库仑积分屏蔽阈值。
THR_F12	F12积分的屏蔽阈值。
THRM0	用于半转换积分的屏蔽阈值。
THRPROD	在第一个半转换中的乘积屏蔽阈值。
NOMP2	如果设定为1, 则只执行F12计算, 跳过LMP2。这在使用完全域时有时会有用, 因为迭代LMP2 将会导致大量的额外计算量, 需要大量内存。接下来分开计算DF-MP2会更高效, 总能量计算为DF-MP2能量和F12能量之和。

密度拟合的更多选项在第十五章有说明, 选择解的更多选项见30.7。

使用局域近似进行计算的典型输入为:

! 局域密度拟合参数:  
DFIT, LOCFIT\_F12=1, FITDOM\_MP2=1, IDOMAUX\_MP2=3, DSCREEN=1

```
! 使用局域RI的LMP2-F12(loc):
{DF-LMP2-F12,ANSATZ=3*A(LOC),DOMSEL=0.985,RAODOM=5,PAIRS=WEAK}
```

这会用Boughton-Pulay域执行局域MP2，完备标准为0.985。在F12部分，不包含远距离轨道对（PAIRS=WEAK），并使用局域投影算符（USEPA0=1，默认）。使用局域密度拟合和局域RI近似。

相应的非局域计算为（仍旧使用局域轨道和对角解）：

```
{DF-LMP2-F12,ANSATZ=3*A(LOC),DOMSEL=1.0,USEVRT=1,NOMP2=1}
ecorr_F12=ef12

{DF-MP2}
ecorr_MP2=energy-energr          ! mp2关联能
ecorr_MP2_F12=ecorr_MP2+ecorr_F12 ! 总关联能
```

注意：使用局域DF和RI域仍在实验阶段，使用时要当心！

### 30.11 F12程序设定的变量

以下的变量由F12程序设定：

ENERGR	参考能量。它包含微扰CABS单激发校正，如果计算了的话。
ENERGY	指定方法的总能量（含F12和单激发校正）。ENERGY(1)和ENERGY(2)分别保存F12A和F12B 的值（如果都计算了的话）。
ENERGC	在ccsd-f12或uccsd-f12计算中，总CCSD-F12能量。ENERGC(1)和ENERGC(2)分别保存F12A 和F12B的值（如果都计算了的话）。ENERGY和ENERGC的区别是三激发能量的贡献。
ENERGT	三激发能量的贡献。这是个矢量。相应的方法存储于METHODT（字符串）。
EMP2	总MP2能量（不包含F12校正，但包含单激发校正）。
EMP2_SCS	总SCS-MP2能量（不包含F12校正，但包含单激发校正）。
EMP2_SING	单重MP2关联能（不包含F12校正）。
EMP2_TRIP	三重MP2关联能（不包含F12校正）。
EMP2_STRONG	LMP2关联能的强对贡献（只要可用）。
EMP2_CLOSE	LMP2关联能的封闭对贡献（只要可用）。
EMP2_WEAK	LMP2关联能的弱对贡献（只要可用）。
EMP2_DIST	LMP2-F12关联能的远距离对的F12贡献（只要可用）。

EF12	对指定解的MP2-F12关联能的F12贡献。
EF12S	对指定解使用EBC近似的MP2-F12关联能的F12贡献。
EF12D	对指定解使用EBC近似和对角（DX）近似的MP2-F12关联能的F12贡献。
EF12_SING	对指定解的MP2-F12关联能的单重F12贡献。
EF12_TRIP	对指定解的MP2-F12关联能的三重F12贡献。
EF12_STRONG	LMP2-F12关联能的强对F12的贡献（只要可用）。
EF12_CLOSE	LMP2-F12关联能的封闭对F12的贡献（只要可用）。
EF12_WEAK	LMP2-F12关联能的弱对F12的贡献（只要可用）。
EF12_DIST	LMP2-F12关联能的远距离对F12的贡献（只要可用）。
EF12_SCS	指定解的MP2-F12关联能的F12贡献。
EF12_SINGLES	总CABS单激发的贡献（闭壳层情况下等于EF12_RHFRELAX）。
EF12_RHFRELAX	在参考能量中CABS单激发的贡献（仅使用无自旋的贡献）。
ANSATZ	需要的解（字符串变量）。

有对应于EF12\_\*的变量，同样也有用于EF12S\_\*和EF12D\_\*的变量。

对于有疑问的情况，可以在测试计算中试一试

SHOW,ENERG\*,EMP2\*,EF12\*

这将显示所有存在的有关变量。注意，系统变量添加下划线前缀，在内部存储，可以用SHOW 命令显示。加上或去掉下划线也可以访问变量（但是，如果用户定义了同名的变量，那么在访问系统变量时下划线是必须的，它不是用户定义的变量）。



## 第三十一章

## 完全CI程序

这个模块是行列式完全CI程序，说明参见

P.J. Knowles and N.C. Handy, Chem. Phys. Letters 111 (1984) 315,

P.J. Knowles and N.C. Handy, Comp. Phys. Commun. 54 (1989) 75.

使用本程序结果的出版物应当引用这些文献。

正常情况下使用本程序寻找完全CI哈密顿矩阵的最低本征矢；也可以进行更复杂的使用，但这里不再说明。程序通过使用DUMP选项，可以连接到CPC程序库提供的免费标准版本程序。

本程序用FCI命令调用。

### 31.1 定义轨道

ORBIT,*name.file*;

*name.file*指定从哪个轨道读取记录。默认为最后一次SCF，MCSCF或CI 计算的轨道集。

### 31.2 占据轨道

OCC, $n_1, n_2, \dots, n_8$ ;

$n_i$ 定义在第*i*个不可约表示中的占据轨道数（含CORE）。如果没有给出，默认为完全基组。

### 31.3 冻芯轨道

CORE, $n_1, n_2, \dots, n_8$ ;

$n_i$ 是在第 $i$ 个不可约表示中的冻芯轨道数。这些轨道在所有的组态中都是双占据的，也就是说，不进行关联计算。如果CORE卡没有给出，程序使用与最后一次CI计算相同的芯轨道；如果没有CI计算，那么使用原子的内壳层作为芯。要避免这一特性并对所有的电子进行关联计算，可以指定：

CORE

### 31.4 定义态的对称性

电子的个数和波函的总对称性在WF卡定义：

WF,*elec,sym,spin*

其中，

*elec*                电子数。

*sym*                不可约表示的序号。

*spin*                定义自旋对称性， $spin = 2S$ （单重态=0，双重态=1，三重态=2，等）。

### 31.5 打印选项

PRINT,*code,value*;

打印选项。这个值通常确定打印多少中间信息。*value*=-1表示不打印（对所有代码都是默认的）。如果忽略了*value*，则作为零看待，通常是合适的。指定更高的值将产生更多的输出。允许指定以下*code*：

ORBITAL            打印分子轨道。

INTEGRAL          打印积分。

TIMING             打印额外的时间信息。

DIAGONAL          打印哈密顿量的对角元。

HAMILTONIAN       打印更多的中间信息。

### 31.6 连接到其它的程序

DUMP;

导致绕过FCI对角化，把输入信息和转换积分写到格式化的FCIDUMP文件中。格式的说明见Comp. Phys. Commun. 54 (1989) 75。

## 31.7 例子

```
! $Revision: 2006.0 $  
***,FCI calculation for water in dz basis set  
memory,4,m  
basis=dz  
geometry={  
o  
h,,0.0, 1.1,1.4  
h,,0.0,-1.1,1.4}  
hf  
{fci  
wf,10,1;core,1}      ! 选项可忽略  
                      ! FCI能量 = -76.08088039  
---;
```

[http://www.molpro.net/info/current/examples/h2o\\_fci.com](http://www.molpro.net/info/current/examples/h2o_fci.com)



### 32.1 简介

SAPT (对称匹配分子间微扰理论, symmetry-adapted intermolecular perturbation theory) 程序计算闭壳层分子间的总相互作用能, 它是一阶和二阶相互作用项之和, 也就是静电项 $E_{\text{pol}}^{(1)}$ , 诱导项 $E_{\text{ind}}^{(2)}$ , 扩散项 $E_{\text{disp}}^{(2)}$ , 以及它们各自的交换部分 ( $E_{\text{exch}}^{(1)}$ ,  $E_{\text{exch-ind}}^{(2)}$ , 和 $E_{\text{exch-disp}}^{(2)}$ )。后几项由分子相互靠近时的各单体间的电子交换产生, 有时也称为Pauli排斥。由于上面所有的项可以通过单体的密度矩阵以及静态和动态的密度-密度响应函数获得, 因此在原则上不需要计算二聚物波函 (参见32.4节)。因此SAPT不会有超分子方法中出现的零阶基组重叠误差。

参考文献:

一般的对称匹配微扰理论和多体SAPT:

- [1] B. Jeziorski, R. Moszynski and K. Szalewicz, Chem. Rev. **94**, 1887. (1994).

DFT-SAPT:

- [2] G. Jansen and A. Heßelmann, J. Phys. Chem. A **105**, 646 (2001).  
[3] A. Heßelmann and G. Jansen, Chem. Phys. Lett. **357**, 464 (2002).  
[4] A. Heßelmann and G. Jansen, Chem. Phys. Lett. **362**, 319 (2002).  
[5] A. Heßelmann and G. Jansen, Chem. Phys. Lett. **367**, 778 (2003).  
[6] A. Heßelmann and G. Jansen, Phys. Chem. Chem. Phys. **5**, 5010 (2003).

密度拟合DFT-SAPT (DF-DFT-SAPT):

- [7] A. Heßelmann, G. Jansen and M. Schütz, J. Chem. Phys. **122**, 014103 (2005).

(关于和DFT-SAPT有关的方法, 称作SAPT(DFT), 另外参见:  
K. Szalewicz, K. Patkowski and B. Jeziorski, Struct. Bond **116**, 43 (2005)  
及其中的文献。)

## 32.2 第一个例子

SAPT典型的输入有如下形式:

```
r=5.6
geometry={nosym; he1; he2,he1,r}
basis=avqz

! 波函记录
ca=2101.2
cb=2102.2

! 单体A
dummy,he2
{hf; save,$ca}
sapt;monomerA

! 单体B
dummy,he1
{hf; start,atdens; save,$cb}
sapt;monomerB

! 相互作用的贡献
sapt;intermol,ca=$ca,cb=$cb
```

这里的 `sapt;monomerA/B` 存储关于两个单体的信息, 它们用于随后的SAPT计算, 用 `sapt;intermol` 调用。各个相互作用能量项存储在不同的变量中 (单位millihartree), 可以保存在数组中用来产生势能面。例如, 输入

```
geometry={nosym; he1; he2,he1,r}
basis=avtz

! 波函记录
ca=2101.2
cb=2102.2

! 距离
dist=[4.5,5.0,5.5,5.6,6.0,6.5,7.0]
```

```

do i=1,#dist
  r=dist(i)

  ! 单体A
  dummy,he2
  {hf; save,$ca}
  sapt;monomerA

  ! 单体B
  dummy,he1
  {hf; start,atdens; save,$cb}
  sapt;monomerB

  ! 相互作用的贡献
  sapt;intermol,ca=$ca,cb=$cb

  elst(i)=E1pol;  exch(i)=E1ex
  ind(i)=E2ind;   exind(i)=E2exind
  disp(i)=E2disp; exdisp(i)=E2exdisp
  etot(i)=E12tot

  data,truncate,$ca
enddo

{table,dist,elst,exch,ind,exind,disp,exdisp,etot
ftyp,d,d,d,d,d,d,d,d,d
plot}

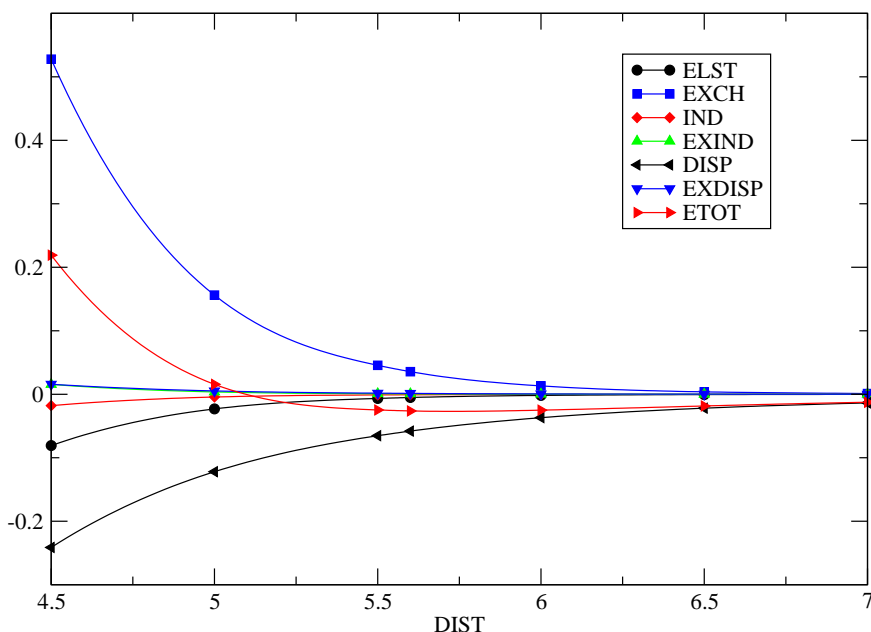
```

将会产生下面的图。

目前SAPT对单体仅接受单行列式波函，也就是说，来自Hartree-Fock或Kohn-Sham DFT（见下一节）计算的波函。这意味着如果对单体使用了Hartree-Fock波函，那么可以获得以下的量（上角标0表示没有包含单体间相互作用）[1]：

$$E_{\text{SAPT}} = E_{\text{pol}}^{(10)} + E_{\text{exch}}^{(10)} + E_{\text{ind,resp.}}^{(20)} + E_{\text{exch-ind,resp.}}^{(20)} + E_{\text{disp}}^{(20)} + E_{\text{exch-disp}}^{(20)}$$

SAPT计算中不能使用点群对称性。



### 32.3 DFT-SAPT

对各个SAPT项考虑分子内的关联影响至关重要，因为Hartree-Fock理论通常产生较差的一阶和二阶静电特性。但这可以（在双微扰理论解内）用多体微扰理论[1]获得。更有效的方法是使用静态和含时DFT理论。这种SAPT变体称为DFT-SAPT [2-6]，与Hartree-Fock-SAPT相反，它有一个诱人的功能是，极化项 ( $E_{\text{pol}}^{(1)}$ ,  $E_{\text{ind}}^{(2)}$ ,  $E_{\text{disp}}^{(2)}$ ) 可以做得非常准确，也就是说，如果已知单体的精确交换关联势 (xc) 和精确的 (含频) xc响应内核，那么就可以算得非常准确。另一方面，这对交换项是不可能的，因为对于多电子体系的精确密度矩阵，Kohn-Sham最多只能给出一个不错的近似。发现恰恰就是这种情况[6]，因此DFT-SAPT只是可能产生精度比高阶超分子多体微扰或耦合簇理论更高的相互作用能。但是为了获得这样的精度，在流DFT泛函中校正XC势错误的渐近行为就非常重要[3-5]。例如，这可以用：

```
{ks,lda; asymp,<shift>}
```

它对两个单体计算激活Grüning等的梯度调节渐进校正方法 (J. Chem. Phys. **114**, 652 (2001))。用户必须对总电势提供`shift`参数 ( $\Delta_{xc}$ )，它近似为单体的 (负) 电离势 (IP) 和标准Kohn-Sham 计算得到的HOMO能量 ( $\varepsilon_{HOMO}$ ) 之差：

$$\Delta_{xc} = \varepsilon_{HOMO} - (-IP)$$

这个方法考虑了精确XC势的导数不连续，它在近似方法中不会出现。注意，这对每个体系只需要计算一次 (使用的例子参见32.6.2节)。

关于DFT单体计算的更多技术参数，推荐使用比标准Kohn-Sham计算更低的收敛



阈值和更大的积分格点进行计算。

## 32.4 高阶项

发现如果一个或两个单体是极性的，那么三阶和更高阶项就变得非常重要。由于目前在SAPT中没有高于二阶的项，我们可以从超分子Hartree-Fock 计算对这些项做非关联的估计（例如，参见[7]）。这可以通过修改以下模板来完成：

```
! 二聚物
hf
edm=energy

! 单体A
dummy,<monomer2>
{hf; save,$ca}
ema=energy
sapt;monomerA

! 单体B
dummy,<monomer1>
{hf; start,atdens; save,$cb}
emb=energy
sapt;monomerB

! 相互作用的贡献
sapt,sapt_level=2;intermol,ca=$ca,cb=$cb

esup=(edm-ema-emb)*1000. mH
dHF=esup-e1pol-e1ex-e2ind-e2exind
```

它把结果 $\delta(\text{HF})$ 项存到dHF中。

## 32.5 密度拟合

为了研究延长单体间的相互作用，在SAPT中可以使用密度拟合对积分做近似[7]。对这种计算可以使用输入：

```
{sapt;intermol,ca=$ca,cb=$cb,fitlevel=3
dfit,basis_coul=jkfit,basis_exch=jkfit,basis_mp2=mp2fit,cfit_scf=3}
```

在基组部分需要定义jkfit和mp2fit拟合基组（参见第十五章）。

对于SAPT\_LEVEL=3和SAPT\_FITLEVEL=3的情况，目前只能执行ALDA 交换关联内核。这意味着相应的SAPT计算与用杂化DFT计算的单体轨道/轨道能量不兼容。因此

在DF-DFT-SAPT计算中，建议对扩散/交换扩散能量项使用非杂化泛函。另一种可能办法是局域化交换关联势，例如通过OEP方法（另外参见32.6.3节的例子）。

## 32.6 示例

### 32.6.1 使用 $\delta(HF)$ 校正的HF-SAPT计算 $H_2O$ 二聚物

```
! examples/h2odimer_sapt_hf.com $Revision: 2009.1 $
gthresh,energy=1.d-8,orbital=1.d-8,grid=1.d-8
symmetry,nosym
orient,noorient
GEOMETRY={
1,O1,,0.00000000,0.00000000,0.00000000
2,H1,,0.00000000,0.00000000,1.83606000
3,H2,,1.77604000,0.00000000,-0.4656040
4,O2,, -0.6605540,0.00000000,5.54064000
5,H3,, -1.6582100,-1.4536300,6.05324000
6,H4,, -1.6582100,1.45363000,6.05324000
}
basis=avdz

! sapt文件
ca=2101.2
cb=2102.2

! 二聚物
hf
edm=energy

! 单体A
dummy,o2,h3,h4
{hf; save,$ca}
ema=energy
sapt;monomerA

! 单体B
dummy,o1,h1,h2
{hf; start,atdens; save,$cb}
emb=energy
sapt;monomerB

! 相互作用的贡献
sapt,SAPT_LEVEL=3;intermol,ca=$ca,cb=$cb,icpks=1
```

```
! HF超分子相互作用能和delta(HF)贡献
eint_hf=(edm-ema-emb)*1000 mH
delta_hf=eint_hf-e1pol-e1ex-e2ind-e2exind

! 把E2disp + E2exch-disp加入到HF相互作用能
eint_sapt=eint_hf+e2disp+e2exdisp
```

[http://www.molpro.net/info/current/examples/h2odimer\\_sapt\\_hf.com](http://www.molpro.net/info/current/examples/h2odimer_sapt_hf.com)

### 32.6.2 使用 $\delta(HF)$ 校正的DFT-SAPT计算 $NeAr$ 二聚物

```
! examples/near_sapt_acdft.com $Revision: 2009.1 $
gthresh,energy=1.d-8,orbital=1.d-8,grid=1.d-8
symmetry,nosym
orient,noorient
geometry={
Ne,,0.0,0.0,0.0
Ar,,0.0,0.0,6.5}
basis=avtz

!=====delta(HF)对高阶相互作用项的贡献=====
! sapt文件
ca=2101.2
cb=2102.2

! 二聚物
hf
edm=energy

! 单体A
dummy,ar
{hf; save,$ca}
ema=energy
sapt;monomerA

! 单体B
dummy,ne
{hf; start,atdens; save,$cb}
emb=energy
sapt;monomerB

! 相互作用的贡献
sapt,SAPT_LEVEL=2;intermol,ca=$ca,cb=$cb,icpks=1
```

```

! 通过减去一阶和二阶能量，计算高阶项
eint_hf=(edm-ema-emb)*1000 mH
delta_hf=eint_hf-e1pol-e1ex-e2ind-e2exind

!=====二阶的DFT-SAPT分子间微扰理论=====
! sapt文件
ca=2103.2
cb=2104.2

! 对xc势渐进校正的移动
eps_homo_pbe0_ar=-0.440936      ! HOMO(Ar)/PBE0泛函
eps_homo_pbe0_ne=-0.589207      ! HOMO(Ne)/PBE0
ip_ar=0.5792                    ! 电离势的实验值
ip_ne=0.7925                    ! 电离势的实验值
shift_ar=ip_ar+eps_homo_pbe0_ar ! 总xc势的移动(Ar)
shift_ne=ip_ne+eps_homo_pbe0_ne ! 总xc势的移动(Ne)

! 单体A
dummy,ar
{ks,pbe0; asymp,shift_ne; save,$ca}
sapt;monomerA

! 单体B
dummy,ne
{ks,pbe0; start,atdens; asymp,shift_ar; save,$cb}
sapt;monomerB

! 相互作用的贡献
sapt;intermol,ca=$ca,cb=$cb,icpks=0

! 加上高阶近似，获得总相互作用能
eint_dftsapt=e12tot+delta_hf

```

[http://www.molpro.net/info/current/examples/near\\_sapt\\_acdft.com](http://www.molpro.net/info/current/examples/near_sapt_acdft.com)

### 32.6.3 使用 $\delta(HF)$ 校正的DF-DFT-SAPT计算 $NeAr$ 二聚物

```

! examples/near_df_sapt_acdft.com $Revision: 2009.1 $
gdirect; gthresh,energy=1.d-8,orbital=1.d-8,grid=1.d-8
symmetry,nosym
orient,noorient
geometry={
Ne,,0.0,0.0,0.0

```

```
Ar,,0.0,0.0,6.5}
basis={
set,orbital; default,avtz      ! 用于轨道
set,jkfit;   default,avtz/jkfit ! 用于JK积分
set,mp2fit;  default,avtz/mp2fit ! 用于E2disp/E2exch-disp
set,df1hf;   default,avtz/jkfit ! 用于LHF
}

!=====delta(HF)对高阶相互作用项的贡献=====
ca=2101.2; cb=2102.2 ! sapt文件

! 二聚物
{df-hf,basis=jkfit,locorb=0}
edm=energy

! 单体A
dummy,ar
{df-hf,basis=jkfit,locorb=0; save,$ca}
ema=energy; sapt;monomerA

! 单体B
dummy,ne
{df-hf,basis=jkfit,locorb=0; save,$cb}
emb=energy; sapt;monomerB

! 相互作用的贡献
{sapt,SAPT_LEVEL=2;intermol,ca=$ca,cb=$cb,icpks=1,filelevel=3
dfit,basis_coul=jkfit,basis_exch=jkfit,cfit_scf=3}

! 通过减去一阶和二阶能量, 计算高阶项
eint_hf=(edm-ema-emb)*1000 mH
delta_hf=eint_hf-e1pol-e1ex-e2ind-e2exind

!=====二阶的DFT-SAPT分子间微扰理论=====
ca=2103.2; cb=2104.2 ! sapt文件

! 对xc势渐进校正的移动
eps_homo_pbe0_ar=-0.440936      ! HOMO(Ar)/PBE0泛函
eps_homo_pbe0_ne=-0.589207      ! HOMO(Ne)/PBE0
ip_ar=0.5792                    ! 电离势的实验值
ip_ne=0.7925                    ! 电离势的实验值
shift_ar=ip_ar+eps_homo_pbe0_ar ! 总xc势的移动(Ar)
shift_ne=ip_ne+eps_homo_pbe0_ne ! 总xc势的移动(Ne)
```

```

! 单体A, 执行LPBEOAC计算
dummy,ar
{df-ks,pbex,pw91c,lhf; dftfac,0.75,1.0,0.25; asymp,shift_ne; save,$ca}
sapt;monomerA

! 单体B, 执行LPBEOAC计算
dummy,ne
{df-ks,pbex,pw91c,lhf; dftfac,0.75,1.0,0.25; start,atdens; asymp,shift_ar; save,$cb}
sapt;monomerB

! 相互作用的贡献
{sapt,SAPT_LEVEL=3;intermol,ca=$ca,cb=$cb,icpks=0,fitlevel=3,nlexfac=0.0
dft,basis_coul=jkfit,basis_exch=jkfit,cfit_scf=3}

! 加上高阶近似, 获得总相互作用能
eint_dftsapt=e12tot+delta_hf

```

[http://www.molpro.net/info/current/examples/near\\_df\\_sapt\\_acdft.com](http://www.molpro.net/info/current/examples/near_df_sapt_acdft.com)

## 32.7 选项

SAPT_LEVEL	设定为1使用一阶项 ( $E_{\text{pol}}^{(1)}$ 和 $E_{\text{exch}}^{(1)}$ ), 设定为2使用二阶 (交换) 诱导项 ( $E_{\text{ind}}^{(2)}$ 和 $E_{\text{exch-ind}}^{(2)}$ ), 设定为3使用所有一阶和二阶项 (还包含 $E_{\text{disp}}^{(2)}$ 和 $E_{\text{exch-disp}}^{(2)}$ )。 (默认3)
SAPT_FITLEVEL	在SAPT中, 密度拟合近似的级别, 值可以是0到3 (默认0)。
SAPT_ICPKS	对于耦合微扰Kohn-Sham方程, 在迭代 (=1) 和非迭代 (=0) 解之间切换 (默认0)。
SAPT_CPKSTHR	在耦合微扰Kohn-Sham程序中, 用于密度矩阵收敛的阈值 (默认1.d-6)。
SAPT_CPKMAXIT	在耦合微扰Kohn-Sham程序中, 迭代的最大数目 (默认50)。
SAPT_FROZENA	在单体A的响应计算中, 冻结的电子数 (默认0)。

如果SAPT\_FITLEVEL>0, 以下参数会很重要:

SAPT_NFRQ_DISP	用于Casimir-Polder积分的频率数 (默认12)。
SAPT_NORM_DISP	密度拟合的标准, 可以是COULOMB或NATURAL (默认COULOMB)。
SAPT_DISP_N4	可以用 $N^4$ 标度加速扩散能量的计算 (默认1)。
THR_XCKERN	xc内核矩阵元的密度阈值 (默认1.d-8)。

FIT_XCKERN	设置xc内核的两侧（默认0）。
SAPT_DISK	如果为0，把所有二聚物的振幅写入文件； 如果为1，把三指数响应传播子写入文件； 如果为2，把三指数响应传播子压缩到文件。 后两个变体可以节省磁盘空间，但需要更多的CPU时间来计算 $E_{\text{exch-disp}}^{(2)}$ （默认0）。
COMPRESS_THR	如果SAPT_DISK=2，这个值确定压缩因子（默认1d-12）。
UNCOUPLED	如果SAPT_DISK>0，还计算非耦合的（交换）扩散能（默认为否）。
THRAO	AO 3指数积分的阈值（默认1.d-12）。
THRMO	MO 3指数积分的阈值（默认1.d-8）。
THROV	AO 2指数积分的阈值（默认1.d-10）。
THRPROD	第一个半转换的乘积阈值（默认1.d-8）。
THRSW	Schwarz屏蔽的阈值（默认1.d-5）。
C6	对两个单体计算扩散因子（注意，整个二聚物的基组用于每个单体，单体距离太近会扰动结果）。
XCKERN_NBLOCK	对于 $(\text{aux} \text{f}_{\text{xc}} \text{occ}\times\text{virt})$ 积分，作为一个区块同时处理的格点数（默认128）。
CFAC	在ALDA xc内核中，VWN关联的因子（默认1.d0）。

在密度拟合计算中，用于2、3指数积分的最后几个阈值不要设得太高，因为这会导致相互作用项的较低精度。此外，SAPT还接受以下子命令：

MONOMERA	存储前一个单体A计算的信息（如电子数，等）。
MONOMERB	见上。
INTERMOL	开始SAPT计算。

INTERMOL可以有以下的子命令：

CA	单体A波函的记录编号（总是必须）。
CB	单体B波函的记录编号（总是必须）。
SAPTLEVEL	见上。
FITLEVEL	见上。
ICPKS	见上。
FROZA	见上。

---

FROZB	见上。
NLEXFAC	在杂化DFT-SAPT计算中，非局域精确交换的总量。
CPKSTHR	在耦合微扰Kohn-Sham程序中，密度矩阵的收敛阈值。
CPKSMAXIT	在耦合微扰Kohn-Sham程序中，最大的迭代数。



### 33.1 特性程序

特性程序允许求解单电子算符和期待值。当使用了全局GEXPEC指令（参见6.13节）或者在SCF，MCSCF，和CI程序中使用了EXPEC或TRAN命令时，通常会自动计算这些算符。在少数情况下，只有当用户对特性的轨道分析感兴趣时，才需要直接使用特性程序。

#### 33.1.1 调用特性程序（PROPERTY）

PROPERTY

调用特性程序。

#### 33.1.2 期待值（DENSITY）

DENSITY [*record.file*] [*specifications*]

如果出现了这个卡，将从记录*record.file*读取密度矩阵，并计算特性期待值。如果忽略了*record.file*的定义，则使用最后的转存记录。特定态的密度矩阵可以用*specifications* 选择，解释参见4.11节。注意，密度矩阵存储在和轨道相同的记录中。

#### 33.1.3 轨道分析（ORBITAL）

ORBITAL [*record.file*] [*specifications*]

如果出现了这个卡，则从记录*record.file*读取轨道，并打印期待值的轨道分析（还必须提供密度矩阵!）。如果忽略了*record.file*，则使用最后的转存记录。这只对对角的

密度矩阵（SCF或者自然轨道）有意义。特定态的自然轨道可以用`specifications`选择，解释参见4.11节。

### 33.1.4 指定单电子算符

需要的算符用代码字定义。也可以定义进行算符计算的几何的或核的中心。

对于每个算符，需要以下格式的输入卡：

`code,centre,x,y,z,,factor`

`code`指定特性。可用的算符在6.13节给出。

其它的参数具有以下的含义：

<code>centre</code>	定义将要进行特性计算的中心，可以是Z矩阵行号或原子符号；如果 <code>centre</code> ≠0，则不需要读入坐标。
<code>x,y,z</code>	点的笛卡尔坐标（只用于 <code>centre</code> =0）。
<code>factor</code>	算符乘上这个因子。默认值是 <code>factor</code> =1，但REL除外。在这种情况下，使用合适的因子用于相对论校正，除非给定了 <code>factor</code> 。因子前的两个逗号是必须的，以保证和Molpro96兼容。

### 33.1.5 打印选项

PRINT,`print`

这个卡控制输出，主要用于调试。

<code>print=0</code>	无测试输出（默认）。
<code>print&gt;0</code>	打印算符。

### 33.1.6 例子

下面的例子计算水的偶极和四极矩，并打印轨道分析。默认的原点位于质心，这也作为四极矩的原点。

```
! $Revision: 2006.0 $
***,h2o properties
geometry={o;h1,o,r;h2,o,r,h1,theta} ! Z矩阵结构输入
r=1 ang ! 键长
theta=104 ! 键角
hf ! 进行scf计算
property ! 调用特性程序
orbital ! 读取scf轨道
density ! 读取scf密度矩阵
dm ! 计算偶极矩并打印轨道贡献
```

```

qm                                ! 计算四极矩并打印轨道贡献
{multi,state,2;dm                ! 进行全部价电子的CASSCF计算
natorb,state=1.1                 ! 对1.1态计算自然轨道
natorb,state=2.1}               ! 对2.1态计算自然轨道

{property                         ! 调用特性程序
orbital,state=1.1                ! 读取1.1态的casscf自然轨道
density,state=1.1               ! 读取1.1态的casscf密度矩阵
dm                               ! 计算偶极矩并打印轨道贡献
qm}                              ! 计算四极矩并打印轨道贡献

{property                         ! 调用特性程序
orbital,state=2.1                ! 读取2.1态的casscf自然轨道
density,state=2.1               ! 读取2.1态的casscf密度矩阵
dm                               ! 计算偶极矩并打印轨道贡献
qm}                              ! 计算四极矩并打印轨道贡献

```

<http://www.molpro.net/info/current/examples/h2o-property.com>

偶极和四极矩也可以在SCF和MCSCF程序中直接计算，但是在这种情况下不打印轨道贡献：

```

! $Revision: 2006.0 $
***,h2o properties
geometry={o;h1,o,r;h2,o,r,h1,theta} ! Z矩阵结构输入
r=1 ang                               ! 键长
theta=104                             ! 键角
gexpec,dm,qm                          ! 请求全局的偶极矩和四极矩计算
hf                                    ! 进行scf计算
{multi,state,2                        ! 进行全部价电子的CASSCF计算
natorb,state=1.1                     ! 对1.1态计算自然轨道
natorb,state=2.1}                   ! 对2.1态计算自然轨道

```

<http://www.molpro.net/info/current/examples/h2o-gexpec1.com>

其它的例子：

```

! $Revision: 2006.0 $
***,h2o properties
geometry={o;h1,o,r;h2,o,r,h1,theta} ! Z矩阵结构输入
r=1 ang                               ! 键长
theta=104                             ! 键角
gexpec,dm,sm,qm                      ! 计算偶极矩和四极矩
$methods=[hf,multi,ci]              ! 指定hf, casscf, mrci方法
do i=1,#methods                      ! 对方法做循环
$methods(i)                          ! 进行能量计算
e(i)=energy

```

```
dip(i)=dmz          ! 把偶极矩保存到变量dip
quadxx(i)=qmxx      ! 保存四极矩
quady(i)=qmyy
quadzz(i)=qmzz
smxx(i)=xx          ! 保存二次矩
smyy(i)=yy
smzz(i)=zz
enddo
table,methods,dip,smxx,smyy,smzz    ! 打印一次、二次矩的表格
table,methods,e,quadxx,quady,quadzz ! 打印四极矩的表格
```

<http://www.molpro.net/info/current/examples/h2o-gexpec2.com>

```
! $Revision: 2006.0 $
***,H2O properties
memory,3,m          ! 分配1 MW动态内存
r=1.85              ! 定义初始键长
theta=104           ! 定义初始键角(=2 theta)
geometry={o;h1,o,r;h2,o,r,h1,theta} ! 结构输入
Basis={
s,1,v;c,1.6;        ! van Duijneveld的13s基函数, 前6个收缩
p,1,v;c,1.4;        ! van Duijneveld的8p基函数, 前4个收缩
d,1,3d,f,1,2f       ! Dunning优化的3d, 2f
s,2,v;c,1.4;p,2,3p} ! H的8s3p基组

gexpec,qm,ef1,ef2,fg,,0,0,0.5      ! (0,0,0.5)的场梯度

hf                                ! 闭壳层SCF

pop                                ! 使用SCF密度的Mulliken布居分析

{dma                                ! 使用SCF密度的分布多极分析
limit,,3};                          ! 限定为级别3

multi                              ! 全部价电子的CASSCF计算

{pop;                                ! 使用mcscf密度的Mulliken布居分析
individual}                          ! 给出各个基函数的占据

{dma;                                ! 使用mcscf密度的分布多极分析
limit,,3}                            ! 限定为级别3

{ci                                ! casscf参考的MRCI
natorb,2350.2
dm,2350.2}                          ! 把MRCI密度矩阵保存到文件2的记录2350
```

```

{pop;density,2350.2;                                ! 使用MRCI密度的Mulliken布居分析
individual}                                           ! 给出各个基函数的占据

{property;
density,2350.2;                                       ! 用特性程序计算期待值
orbital,2350.2;
qm;ef,1;ef,2;fg,,0.5}                               ! 算符同前
---
```

<http://www.molpro.net/info/current/examples/h2o-prop.com>

## 33.2 分布多极分析

任何密度矩阵都可以用分布多极分析方法进行分析，说明参见Stone, Chem. Phys. Letters (1981), 83, 233。对重叠中心计算每一对原函数重叠产生的多极，并移动到离得最近的一些多极格点位置上。默认格点包含在积分输入中指定的所有原子。但是可以通过删除和/或添加格点位置，或是限制从给定格点位置移动的多极的等级，对多极格点位置的列表进行修改。原子电荷存储于MOLPRO变量ATCHARGE中。ATCHARGE的第*i*个元对应于Z-矩阵输入的第*i*行。

选项可以以任何顺序出现，但DENSITY除外，它如果出现的话需要首先给出。

当前的版本还不允许使用一般收缩的AO基组。

### 33.2.1 调用DMA程序 (DMA)

DMA;

这个命令初始化DMA程序。

### 33.2.2 指定密度矩阵 (DENSITY)

DENSITY,record.file[,specifications]

进行分析的密度矩阵就是在文件*file*中找到的记录*record*。如果忽略，那么*record.file*默认为当前的轨道记录。如果指定，那么DENSITY必须在输入中首先出现。特定态的密度矩阵可以用*specifications*选择，解释参见4.11节。

### 33.2.3 线型分子 (LINEAR, GENERAL)

GENERAL;

(默认) 调用标准程序，它用于任何结构。

LINEAR

调用更快的程序，它可以用于当所有原子平行于Z-轴排列，并且只需要计算多极的 $m = 0$ 分量的情况。

### 33.2.4 多极的最大等级 (LIMIT)

LIMIT, *name*, *lmax*;

*lmax*是程序计算的多极的最大等级。默认值（以及最大值）对GENERAL程序是10，对于LINEAR程序是20。如果指定了*name*，则该限制只应用于多极格点*name*。

### 33.2.5 忽略核的贡献 (NONUCLEAR)

NONUCLEAR

不求解核对特性的贡献。

### 33.2.6 指定多极格点的位置 (ADD, DELETE)

ADD, *name*, *x*, *y*, *z*, *lmax*, *radius*;

在(*x*, *y*, *z*)添加用*name*指定的新的格点位置。如果指定了*lmax*这个值的话，限定多极的等级，否则使用LIMIT指令指定的*lmax*值。不考虑对称性，在对称等价点的集合中，每个格点位置必须直接指定。也可以指定格点位置的*radius*（默认1.0）。

DELETE, *name*

删除用*name*给定的作为多极格点的所有原子。注意，来自积分程序名为1, 2, 3, ... 的原始原子会在积分的输出中打印。DELETE, ALL删除所有的原子，并且只给出关于原点的多极。

### 33.2.7 定义多极格点位置的半径 (RADIUS)

RADIUS, *name*, *r*;

把半径*r*分配给*name*给出的所有的格点位置。程序把位于重叠中心*P*的多极移动到格点位置*S*上，以使 $|P - S|/r(S)$ 的值最小。如果没有给出RADIUS指令，则所有的格点位置给定的半径是1。

### 33.2.8 注释与参考文献

分析产生的多极用球哈密顿量的定义给出。多极的笛卡尔和球谐定义之间相互转化的公式，参见*Explicit formulae for the electrostatic energy, forces and torques between a pair of molecules of arbitrary symmetry*, S. L. Price, A. J. Stone, and M. Alderton, *Molec. Phys.*, 52, 987 (1984).

使用DMA分析的例子，参见Price and Stone, *Chem. Phys. Lett.*, 98, 419 (1983); Buckingham and Fowler, *J. Chem. Phys.*, 79, 6426 (1983).

### 33.2.9 例子

以下的输入计算水的SCF多极矩。

```
! $Revision: 2006.0 $
***,h2o distributed multipole analysis
geometry={o;h1,o,r;h2,o,r,h1,theta} ! Z矩阵结构输入
r=1 ang ! 键长
theta=104 ! 键角
basis=6-311g**
hf ! 进行scf计算
{dma;limit,,4} ! 总多极矩的结果
```

[http://www.molpro.net/info/current/examples/h2o\\_dma.com](http://www.molpro.net/info/current/examples/h2o_dma.com)

## 33.3 Mulliken布居分析

### 33.3.1 调用布居分析程序 (POP)

POP;

调用Mulliken分析程序，用于把任何密度矩阵分解为每个原子s,p,d,f...基函数的贡献。密度矩阵来自最后一个转存记录，除非使用DENSITY卡覆盖。子命令可以缩写为前四个字符。原子电荷存储在MOLPRO变量ATCHARGE中。ATCHARGE第*i*个元素对应于Z矩阵输入的第*i*行。

### 33.3.2 定义密度矩阵 (DENSITY)

DENSITY,record.file [,specifications]

从文件file的记录record获取用于分析的密度矩阵。特定态的密度矩阵可以用specifications选择，解释参见4.11节。注意，密度矩阵存储在与轨道相同的记录中。

### 33.3.3 基函数的布居 (INDIVIDUAL)

INDIVIDUAL;

如果指定，则打印每个基函数的Mulliken布居。

### 33.3.4 例子

```
! $Revision: 2006.3 $
***,h2o population analysis
geometry={o;h1,o,r;h2,o,r,h1,theta} ! Z矩阵结构输入
r=1 ang ! 键长
theta=104 ! 键角
basis=6-311g**
```

hf	! 进行scf计算
pop;	! 用mcscf密度的Mulliken布居分析
individual	! 给出各个基函数的占据

[http://www.molpro.net/info/current/examples/h2o\\_pop.com](http://www.molpro.net/info/current/examples/h2o_pop.com)

## 33.4 自然键轨道分析

### 33.4.1 调用自然键轨道分析程序 (NBO)

NBO,[WITH\_CORE=*core\_option*],[LEVEL=*level*],[KEEP\_WBI=*wbi\_option*];

可以用NBO卡调用Weinhold及其合作者的自然键轨道分析 (J. Chem. Phys. 83 (1985) 735, J. Chem. Phys. 83 (1985) 1736, 和J. Chem. Phys. 78 (1983) 4066)。它从密度或轨道记录读取, 并执行必要的转换, 成为自然原子轨道 (NAO), 自然键轨道 (NBO), 和自然局域原子轨道 (NLMO)。后者还可以保存到记录中, 用于以后的局域关联处理 (参见第二十九章)。默认使用完全轨道空间, 但是如果定义了*core\_option*=0, 那么芯轨道可以放到流程外。

用LEVEL关键词可以选择截断转换的轨道序列 (例如, 只计算NAO轨道)。如果LEVEL=1, 则只执行NAO转换。对于LEVEL=2, 执行NBO转换, =3为NLMO (默认)。

有时由于双中心键搜索顺序的错误, NBO流程可能不收敛。第一次运行基于Wiberg键级, 但是在随后的运行中, 算法调整到原子顺序。这可以用KEEP\_WBI选项避免。如果*wbi\_option*=1, 那么在所有的迭代中使用Wiberg键级。

### 33.4.2 保存NLMO轨道 (SAVE)

SAVE,*record.file*;

NLMO轨道和NPA电荷保存在指定的记录*record*中。

## 33.5 有限场计算

偶极矩, 四极矩等, 以及相应的极化率, 可以用有限差分方法通过能量的导数得到。这用DIP, QUAD, 或FIELD命令可以很容易地完成。如果添加的微扰不是全对称的 (也就是对称性1), 则会产生错误。注意, 在执行关联计算之前, 轨道必须重新计算。

### 33.5.1 偶极场 (DIP)

DIP,*xfield,yfield,zfield*;

DIP+,*xfield,yfield,zfield*;



为单电子哈密顿量和芯能量添加有限偶极场。场的强度用 $xfield, yfield, zfield$ 给出。用DIP+ 会对存在的场进行添加，否则删除以前所有的场。

### 33.5.2 四极场 (QUAD)

QUAD, $xfield, yfield, zfield, xyfield, xzfield, yzfield$ ;

QUAD+, $xfield, yfield, zfield, xyfield, xzfield, yzfield$ ;

完全同DIP命令，但是添加四级场。

### 33.5.3 一般场 (FIELD)

FIELD, $oper1, fac1, oper2, fac2, \dots$ ;

FIELD+, $oper1, fac1, oper2, fac2, \dots$ ;

为单电子哈密顿量添加单电子算符 $oper1, oper2, \dots$ ，相应的因子为 $fac1, fac2, \dots$ 。可以使用的算符在6.13节给出。如果添加的微扰不是全对称的（也就是对称性1），则会产生错误。

FIELD+会对存在的场进行添加，否则删除以前所有的场。

注意，FIELD目前不会修改芯极化势（CPP）。如果出现了CPP，那么只能使用DIP和QUAD。

### 33.5.4 例子

第一个例子显示为单电子哈密顿量添加微扰的各种可能性。

```
! $Revision: 2006.0 $
***,H2O finite fields
memory,4,m
R      =      0.96488518 ANG
THETA =  101.90140469
geometry={H1
          0,H1,R;
          H2,0,R,H1,THETA}
{hf;wf,10,1}      ! 没有外场的scf

f=0.05

dip,,,f            ! 为h0添加偶极(z)场
hf                ! 用修改的h0进行scf计算

field,dmz,f        ! 为H0添加偶极(z)场
                  ! 与前一个例子相同的结果
hf                ! 用修改的h0进行scf计算
```

```

quad,,,f           ! 为h0添加四极(qmzz)场
hf                 ! 用修改的h0进行scf计算

field,qmzz,f       ! 为h0添加四极(qmzz)场
                  ! 与前一个例子相同的结果
hf                 ! 用修改的h0进行scf计算

field,zz,f,xx,-0.5*f,yy,-0.5*f
                  ! 添加一般场; 结果同上面的四极场
hf                 ! 用修改的h0进行scf计算

field,zz,f         ! 与前面field命令的结果相同
field+,xx,-0.5*f
field+,yy,-0.5*f
hf                 ! 用修改的h0进行scf计算

field              ! 删除场
hf                 ! 没有外场的scf

```

<http://www.molpro.net/info/current/examples/field.com>

第二个例子显示如何用有限场计算偶极矩和极化率。

```

! $Revision: 2006.0 $
***,H2O finite field calculations

r=1.85,theta=104           ! 设定结构参数
geometry={0;                ! z矩阵输入
  H1,0,r;
  H2,0,r,H1,theta}
basis=avtz                  ! 定义默认基组
field=[0,0.005,-0.005]     ! 定义有限场强度
$method=[hf,mp4,ccsd(t),casscf,mrci]

k=0
do i=1,#field               ! 对场做循环
  dip,,,field(i)           ! 为H添加有限场
  do m=1,#method            ! 对方法做循环
    k=k+1
    $method(m)              ! 计算能量
    e(k)=energy             ! 保存能量
  enddo
enddo

k=0

```

```

n=#method
do m=1,#method
  k=k+1
  energ(m)=e(k)
  dipmz(m)=(e(k+n)-e(k+2*n))/(field(2)-field(3))    ! 作为能量一阶导的偶极矩
  dpolz(m)=(e(k+n)+e(k+2*n)-2*e(k))/((field(2)-field(1))*(field(3)-field(1)))
                                              ! 作为二阶导的极化率
enddo

table,method,energ,dipmz,dpolz
title,results for H2O, r=$R, theta=$theta, basis=$basis
---
```

[http://www.molpro.net/info/current/examples/h2o\\_field.com](http://www.molpro.net/info/current/examples/h2o_field.com)

## 33.6 相对论校正

相对论校正可以用Cowan-Griffin方法，通过计算质量-速率和单电子Darwin积分的期待值进行计算；它们应当使用关键词REL调用特性积分程序产生。期待值可以在SCF，MCSCF，和CI 程序内按照通常的方法使用EXPECT命令计算，并且使用REL关键词。此后，质量-速率与Darwin项及其总和可以分别通过MOLPRO变量MASSV，DARW，和EREL引用。

### 33.6.1 例子

```

! $Revision: 2006.3 $
***,ar2
geometry={ar1;ar2,ar1,r}    ! 结构定义
r=2.5 ang                  ! 键距离
{hf;                      ! 非相对论scf计算
expec,rel,darwin,masv}      ! 用Cowan-Griffin算符计算相对论校正
e_nrel=energy              ! 把非相对论能量保存到变量enrel中
show,masv,darwin,erel      ! 显示各部分的贡献及其总和

dkroll=1                  ! 用douglas-kroll单电子积分
hf;                      ! 相对论scf计算
e_dk=energy                ! 把相对论scf能量保存到变量e_dk中
show,masv,darwin,erel      ! 显示质量-速率和darwin贡献，及其总和
show,e_dk-e_nrel          ! 显示Douglas-Kroll的相对论校正
```

[http://www.molpro.net/info/current/examples/ar2\\_rel.com](http://www.molpro.net/info/current/examples/ar2_rel.com)

### 33.7 CUBE—转存密度或轨道值

CUBE,*filename*,*iflag*,*n<sub>1</sub>*,*n<sub>2</sub>*,*n<sub>3</sub>*

调用模块，把空间平行六面体网格的各种特性值转存到一个外部文件中。目的是用外部程序画轨道图，密度图，以及其它的量。文件的格式和其它程序产生的相同。

<i>filename</i>	是要写的文件的unix路径名称，它的定义是必需的。
<i>iflag</i>	如果 <i>iflag</i> 为负（默认），则写格式化的文件，否则使用非格式化的FORTRAN I/O。
<i>n<sub>1</sub></i> , <i>n<sub>2</sub></i> , <i>n<sub>3</sub></i>	指定三个方向中网格点的数量。如果没有指定，选择自动判断的默认值。

默认，最后一次计算的密度对网格求解，并写到*filename*。这一功能可以用以下子命令中的一个或者几个修改。

#### 33.7.1 STEP—设定点的间距

STEP,[*stepx*],[*stepy*],[*stepz*]

*stepx*,*stepy*,*stepz*指定在每个三维轴向上的点间距。*stepx*,*stepy*,*stepz* 的默认值由格点数、原子的Bragg半径以及一些有关的参数确定。

#### 33.7.2 DENSITY—密度的来源

DENSITY,[*density-source*]

GRADIENT,[*density-source*]

LAPLACIAN,[*density-source*]

计算密度及其可选的梯度和拉普拉斯算符。<*density-source*>可以是包含所需密度的记录编号，还可以包含更多的限定，例如用通常方式的集合编号。默认使用最后一次计算的密度。

#### 33.7.3 ORBITAL—轨道的来源

ORBITAL,[*orbital-list*],[RECORD=*orbital-source*]

<*orbital-list*>是一个或多个形式为*number.symmetry*的轨道编号列表，也可以是从HOMO, LUMO, OCC（全部占据轨道），ALL中选择的关键词。如果没有指定，默认为HOMO。<*orbital-source*>可以是包含所需密度的记录编号，还可以包含更多的限定，例如用通常方式的集合编号。默认使用最后一次计算的轨道。

注意，CUBE文件格式禁止同时进行轨道和密度的转存，但是这可以用GOPENMOL格式实现（参见33.8节）。

### 33.7.4 AXIS—网格轴的方向

AXIS,*x,y,z*

*x,y,z*指定用于定义网格三个轴的非归一化方向的余弦。最多可以给出三个AXIS命令，但都不是必须的。轴不需要是正交的。默认的第一个轴是笛卡尔*x*轴，第二个与第一个以及笛卡尔*z*轴正交，第三个与前两个正交。

### 33.7.5 BRAGG—网格的空间范围

基于坐标轴的方向，构造包含整个分子的平行六面体网格（在通常的正交轴情况下，是立方体）。原子假定是球形的，大小与Bragg半径成比例，修改默认的比例常数可以用

BRAGG,*scale*

在构造平行六面体网格之后，网格用相等的间隔展开，并用CUBE命令指定数量的点覆盖原子。

### 33.7.6 ORIGIN—网格的中心

ORIGIN,*x,y,z*

*x,y,z*指定网格的中心。通常不需要使用这个选项，因为默认设置对大多数的用途已经足够了。

### 33.7.7 TITLE—用户定义的标题

TITLE,*title*

在cube文件中设定用户定义的标题。

### 33.7.8 DESCRIPTION—用户定义的描述

DESCRIPTION,*description*

在cube文件中设定用户定义的描述。

### 33.7.9 立方文件的格式

格式化的立方文件包含以下的记录：

- (A) 任务标题。
- (A) 文件内容的简要描述。
- (I5,3F12.6) 原子数，网格原点的坐标（单位bohr）。
- (I5,3F12.6) 网格点 $n_1$ 的数量，第一个网格维度的步长矢量。

- (I5,3F12.6) 网格点 $n_2$ 的数量，第二个网格维度的步长矢量。
- (I5,3F12.6) 网格点 $n_3$ 的数量，第三个网格维度的步长矢量。
- (I5,4F12.6) 原子序号，电荷及坐标；每个原子需要一个这种记录。
- (6E13.5)  $n_1 \times n_2$ 个长度为 $n_3$ 的记录，包含每个网格点的密度或轨道值。对于有许多个轨道 $m$ 的情况，记录的长度为 $m \times n_3$ ，每个网格点的数据分组放在一起。对于密度梯度的情况，先有一个长度为 $n_3$ 包含密度的记录，接下来是一个长度为 $3n_3$ 包含梯度的记录，它有连续的三个笛卡尔分量。对于拉普拉斯量，还有一个长度为 $n_3$ 的记录。

### 33.8 GOPENMOL—计算用于gOpenMol显示的网格

GOPENMOL,*filename,iflag,n<sub>1</sub>,n<sub>2</sub>,n<sub>3</sub>*

除了产生的文件格式可直接用于gOpenMol显示程序以外，语法和子选项完全与CUBE相同。以下应当注意。

- 只使用*filename*中的（最后一个“.”前的）基本名，并加上不同的扩展名创建几个不同的文件：

- .crd** 总是产生包含坐标的CHARMm CRD格式的文件，可用于gOpenMol的调用：  
`rungOpenMol -i filename.crd`
- \_density.plt** 如果给出了DENSITY，则产生文件*filename\_density.plt*，并包含gOpenMol内部格式的密度网格。
- \_orbital\_number.symmetry.plt** 如果给出了ORBITAL，那么对每个指定的轨道*number.symmetry*产生文件*filename\_orbital\_number.symmetry.plt*，并包含gOpenMol内部格式的轨道网格。

- 默认不产生任何轨道或密度，因此只转存原子坐标。
- 默认使用未格式化的二进制文件，这通常不需要改变。
- 不要使用ORIGIN和AXIS命令。

- 如果在输入中给出了

#### INTERACT

并且计算了所有网格，则程序尝试执行Unix命令`rungOpenMol`开始运行`gOpenMol`。如果`rungOpenMol`不在`$PATH`中，那么什么也不做。否则将开始`gOpenMol`，并显示分子。产生的任何`.plt`文件可以在Plot/Contour菜单选项下加入并显示。默认的Unix命令名称`rungOpenMol`可以通过指定INTERACT指令的第一个参数来修改。默认不开始`gOpenMol`，这等价于给出命令BATCH。





MOLPRO考虑标量相对论效应应有三种方法：

- 使用Douglas-Kroll相对论单电子积分；
- 用Cowan-Griffin算符计算微扰校正（参见第6.13节）；
- 使用相对论有效芯势（参见第十二章）。

### 34.1 使用Douglas-Kroll-Hess哈密顿量

对全电子计算，最好的方法是用Douglas-Kroll-Hess (DKH) 哈密顿量，在MOLPRO中可以使用直到任意阶。它只需要在输入文件第一个能量计算之前的某个位置设置

DKROLL=1

就可激活。如果没有指定进一步的输入，则使用标准的二阶Douglas-Kroll-Hess 哈密顿量 (DKH2)。

但是从2006.1发布版开始，MOLPRO也提供（原则上）到任意阶去耦合的DKH (DKH $n$ )。需要在输入文件的DKROLL=1声明之后指定：

DKHO= $n$ , ( $n = 2, \dots, 12$ ),

DKHP= $m$ , ( $m = 1, \dots, 5$ )

用于需要的DKH级别 (DKHO) 和选择酉变换的参数化。MOLPRO支持的参数化有：

DKHP=1:	最优参数化 (OPT, 默认)
DKHP=2:	指数参数化 (EXP)
DKHP=3:	最小二乘参数化 (SQR)

DKHP=4: McWeeny参数化 (MCW)  
DKHP=5: Cayley参数化 (CAY)

### 例子:

```
DKROLL=1 ! 激活Douglas-Kroll-Hess单电子积分
DKHO=8   ! DKH级别 = 8
DKHP=4   ! 对酉变换选择McWeeny参数化
```

(注意: 对于 $DKHO \geq 13$ , src/common/dkhparameters.inc文件中某些参数的值必须相应地增加。只对明确知道自己在做什么的专家用户推荐! 大多数情况下,  $DKHO = 10$ 已经足够。)

直到四阶的DKH哈密顿量 ( $DKHO = 4$ ) 与选择的参数化无关。高阶DKH哈密顿量略微依赖于所选的酉变换的参数化, 它用于Dirac哈密顿量的去耦合。

关于无穷阶DKH哈密顿量的细节, 参见:

M. Reiher, A. Wolf, JCP **121**, 2037-2047 (2004),  
M. Reiher, A. Wolf, JCP **121**, 10945-10956 (2004).

关于酉变换的不同参数化的细节, 参见:

A. Wolf, M. Reiher, B. A. Hess, JCP **117**, 9215-9226 (2002).

## 34.2 计算相对论校正的例子

```
! $Revision: 2006.3 $
***,ar2
geometry={ar1;ar2,ar1,r} ! 结构定义
r=2.5 ang                ! 键长
{hf;                     ! 非相对论scf计算
expec,rel,darwin,massv} ! 用Cowan-Griffin算符计算相对论校正
e_nrel=energy             ! 把非相对论能量保存到e_nrel变量中
show,massv,darwin,erel   ! 显示各部分校正的贡献及其总和

dkroll=1                 ! 使用douglas-kroll单电子积分
hf;                      ! 相对论scf计算
e_dk=energy              ! 把相对论scf能量保存到e_dk变量中.
show,massv,darwin,erel   ! 显示质量-速率和darwin贡献及其总和
show,e_dk-e_nrel        ! 显示Douglas-Kroll相对论校正
```

[http://www.molpro.net/info/current/examples/ar2\\_rel.com](http://www.molpro.net/info/current/examples/ar2_rel.com)

## 第三十五章

### 透热轨道

为了构造透热态，需要确定绝热波函中透热态的混合。原则上，混合可以通过非绝热耦合矩阵元的积分获得。使用近似方法通常会更容易，其中的混合可以通过检查MCSCF或CI波函的CI因子确定。这种方法只能用于轨道混合可忽略的情况。对于CASSCF波函，这可以通过当前结构与参考结构的活性轨道重叠最大化来实现，其中的波函假定是透热的（例如，由于对称性的原因）。通过使用MCSCF程序的新命令DIAB来实现轨道重叠最大化。

工作的流程如下：首先确定参考几何结构的轨道。接下来在移动的几何结构上进行计算，获得与参考结构活性轨道有最大重叠的“透热”活性轨道，这可以通过在输入中添加DIAB指令来实现：

旧的形式（Molpro96，已作废）：

```
DIAB,orbref, orbsav, orb1,orb2,pri
```

新的形式：

```
DIAB,orbref[,TYPE=orbtype][,STATE=state][,SPIN=spin][,MS2=ms2][,SAVE=orbsav]  
[,ORB1=orb1, ORB2=orb2][,PRINT=pri]
```

这里的`orbref`是保存有参考结构轨道的记录，`orbsav`是存储新轨道的记录。如果`orbsav`没有给出（推荐!），那么新轨道存储在默认的转存记录（2140.2）或ORBITAL指令定义的记录中（参见19.5.3节）。与早期版本的MOLPRO相反，`orbref`和`orbsav`可能相同。TYPE, STATE, SPIN 的说明可以用于选择参考轨道的特定轨道集，这在4.11节有详细说明。`orb1`和`orb2`是重叠最大的一对轨道。这些轨道定义的形式为`number.sym`，例如3.1表示对称性1的第三个轨道。如果`orb1`和`orb2`没有给出，那么所有活性轨道的重叠是最大化的。`pri`是打印参数，如果设置为1，那么在每次Jacobi迭代中都打印每个轨道

的转换角。

在大多数情况下，以上描述的默认值对以下的输入已经足够了：

DIAB,orbref

用Molpro98不需要给出GEOM和DISPL卡。位移结构和重叠矩阵会自动计算（结构和相应的轨道存储于转存记录中）。

透热轨道的性质是：在非绝热耦合矩阵元中，轨道以及重叠贡献的总和近似是零，因此绝热混合的发生只能通过改变CI因子来实现。这允许用简单的方式（说明参见J. Chem. Phys. **89**, 3139 (1988)）或者较高级的方式（说明参见Pacher, Cederbaum, and Köppel, J. Chem. Phys. **89**, 7367 (1988)），直接从CI因子确定混合的角度。

下面我们介绍H<sub>2</sub>S的例子，它的两个最低激发态在C<sub>2v</sub>群分别是B<sub>1</sub>和A<sub>2</sub>对称性，在C<sub>s</sub>群是A''对称性。我们首先在C<sub>2v</sub>对称性执行参考计算，接下来在C<sub>s</sub>对称性的位移结构确定透热轨道。随后的每个计算使用前面的轨道作为参考。也可以在所有其它计算中使用C<sub>2v</sub>的轨道作为参考。在这种情况下，我们必须去掉倒数第二个输入卡reforb=2141.2。

```
! $Revision: 2009.1 $
***,H2S diabatic A" states

basis=VDZ                                ! 用cc-pVDZ基组
symmetry,x,planeyz                       ! 用Cs对称性并且固定分子方向
orient,noorient                           ! 禁止自动调整方向
geometry={s;h1,s,r1;h2,s,r2,h1,theta}    ! Z矩阵结构输入

gprint,orbitals,civector                  ! 全局打印选项

text,reference calculation for C2V
theta=92.12,r1=2.3,r2=2.3                 ! 参考结构

{hf;occ,7,2;wf,18,1}                      ! 基态的scf计算

{multi;occ,9,2;closed,4,1;                ! 定义活性和非活性空间
wf,18,2;state,2;                           ! 两个A"态（C2v的1B1和1A2态）
orbital,2140.2}                             ! 把轨道保存到2140.2
reforb=2140.2

text,calculations at displaced geometries

rd=[2.4,2.5,2.6]                          ! 定义键距离的范围

do i=1,#rd                                 ! 对位移结构循环
```

```
r2=rd(i)                ! 设定r2为当前距离

{multi;occ,9,2;closed,4,1;      ! 与参考结构的波函定义相同
 wf,18,2;state,2;
 orbital,2141.2              ! 把新轨道保存到记录
 diab,reforb}                ! 用参考轨道计算透热轨道
                              ! 存储到记录reforb
reforb=2141.2                ! 把变量reforb设定为新轨道
enddo
```

[http://www.molpro.net/info/current/examples/h2s\\_diab.com](http://www.molpro.net/info/current/examples/h2s_diab.com)

关于计算混合角度的更高级的例子，见示例部分的介绍。



## 第三十六章

# 非绝热耦合矩阵元

非绝热耦合矩阵元可以用DDR程序，通过对MCSCF或CI波函数计算有限差分获得。对于态平均MCSCF波函数，也可以做解析计算（参见19.9.2节）。

注意，目前的数值流程相对于Molpro96已经做了很大简化。不再需要GEOM和DISPL输入卡，而且三个必需的计算可以是任何顺序。

### 36.1 DDR流程

为了用有限差分的方法计算耦合矩阵元，必须计算和存储两个（一阶算法）或三个（二阶算法）少许位移结构的波函数。计算的顺序是任意的。

典型的过程如下：

1. 计算参考结构的波函数。两个态的波函数必须用CI程序的SAVE命令存储。如果对MCSCF波函数计算矩阵元，必须用CI程序加上NOEXC选项重新计算波函数。跃迁密度矩阵用CI程序的DM指令存储。
2. 计算（正方向）位移结构的波函数，并把CI波函数存储到第二个记录中。
3. 如果使用了二阶（三个点）的方法，对（负方向）位移结构重复第二步。
4. 计算参考结构和位移结构两个态之间的跃迁密度矩阵。这用CI程序的TRANS指令完成。
5. 最后，用DDR程序组合矩阵元。如果使用一阶的两点方法，只需要一个输入行：  
DDR, dr, orb1, orb2, trdm2

其中  $dr$  是结构增量，用作有限差分方法的分母， $orb1$  是保存参考结构轨道的记录， $orb2$  是保存位移结构轨道的记录， $trdm2$  保存的跃迁密度矩阵记录分别在  $R$  和  $R+DR$  从 CI 矢量计算。

如果使用了中心差分（三个点），输入为：

```
DDR,2*dr
```

```
ORBITAL,orb1,orb2,orb3
```

```
DENSITY,trdm1,trdm2,trdm3
```

其中  $dr$ ,  $orb1$ ,  $orb2$  同上， $orb3$  是保存负位移结构轨道的记录。 $trdm1$ ,  $trdm2$ ,  $trdm3$  分别是保存跃迁密度  $\gamma(R|R)$ ,  $\gamma(R|R+DR)$  和  $\gamma(R|R-DR)$  的记录。

如果同时计算两个以上的态，则每两个态之间的跃迁密度矩阵将保存到同一个记录中。在这种情况下，以及当仅有两个态但它们的对称性不是 1 时，需要用 **STATE** 指令指定对哪些态计算耦合：

```
STATE,state1, state2
```

其中的  $state_i$  形式为  $istate.isym$ （两个态的对称性必须相同，因此指定第一个态的对称性就足够了）。

一阶和二阶计算输入的例子在下面给出。计算对一组结构范围进行重复，在计算的最后，用 **TABLE** 命令打印结果。

在所给的计算中，通过位移结构与参考结构轨道重叠的最大化，用两个 CASSCF 计算产生“透热” CASSCF 轨道。这是可选的，（在数值精度内）并不影响最终的结果。但是轨道的相对贡献、重叠和 CI 对 NACME 的贡献都会改变。如果使用了位移小到几乎不改变结构作用的透热轨道，那么重叠与轨道贡献的和是最小化的，一个非常好的近似是 NACME 可以只从 CI-矢量获得。

```
! $Revision: 2006.0 $
***,lif non-adiabatic coupling
memory,1,m

basis,f=avdz,li=vdz           ! 定义基组
r=[10.0,10.5,11.0,11.5,12.0]  ! 定义键长
dr=0.01                       ! 定义增量
geometry={li;f,li,rlif}       ! 定义结构

rlif=3                        ! 首先在R=3计算
{hf;occ,4,1,1}                ! SCF
{multi;closed,3;              ! 3个非活性轨道的CASSCF
wf,12,1;state,2;              ! 两个1A1态
```



```

orbital,2140.2}           ! 把轨道转存到记录2140.2

do i=1,#r                 ! 对结构循环
  rlif=r(i)               ! 设定键长
  {multi;closed,3;        ! 3个非活性轨道的CASSCF
  wf,12,1;state,2;        ! 两个1A1态
  orbital,2140.2}          ! 用当前轨道覆盖以前的轨道

  {ci;state,2;noexc;       ! 2个态的无激发CI
  save,6000.2;             ! 把波函保存到记录6000.2
  dm,8000.2}              ! 把(跃迁)密度保存到记录8000.2

  rlif=r(i)+dr            ! 键长增加dr

  {multi;closed,3;        ! CASSCF同上
  wf,12,1;state,2;        ! 两个1A1态
  start,2140.2;           ! 从参考结构的轨道开始
  orbital,2141.2;          ! 把轨道保存到记录2141.2
  diab,2140.2}            ! 通过与参考结构轨道重叠的最大
                          ! 化,产生透热轨道

  {ci;state,2;noexc;save,6001.2} ! 2个态的CI,波函保存到记录6001.2

  {ci;trans,6000.2,6001.2; ! 计算重叠和跃迁密度<R|R+DR>
  dm,8100.2}              ! 把跃迁密度保存到记录8100.2

  rlif=r(i)-dr           ! 在r-dr重复

  {multi;closed,3;        ! CASSCF同上
  wf,12,1;state,2;        ! 两个1A1态
  start,2140.2;           ! 从参考结构的轨道开始
  orbital,2142.2;          ! 把轨道保存到记录2142.2
  diab,2140.2}            ! 通过与参考结构轨道重叠的最大
                          ! 化,产生透热轨道

  {ci;state,2;noexc;save,6002.2} ! 2个态的CI,波函保存到记录6002.2

  {ci;trans,6000.2,6002.2; ! 计算重叠和跃迁密度<R|R-DR>
  dm,8200.2}              ! 把跃迁密度保存到记录8200.2

  {ddr,dr,2140.2,2141.2,8100.2} ! 用两点公式(向前差分)计算NACME
  nacmelp(i)=nacme        ! 把结果存到变量nacmelp
  {ddr,-dr,2140.2,2142.2,8200.2} ! 用两点公式(向后差分)计算NACME
  nacmelm(i)=nacme        ! 把结果存到变量nacmelm

```

```

{ddr,2*dr                ! 用三点公式计算NACME
orbital,2140.2,2141.2,2142.2; ! R, R+DR, R-DR的轨道记录
density,8000.2,8100.2,8200.2} ! R, R+DR, R-DR的跃迁密度记录
nacme2(i)=nacme          ! 把结果存到变量nacme2

end do                    ! 结束差分键长的循环

nacmeav=(nacme1p+nacme1m)*0.5 ! 对向前、向后两个差分结果作平均
table,r,nacme1p,nacme1m,nacmeav,nacme2 ! 打印结果的表格
title,Non-adiabatic couplings for LiF ! 表格标题

```

[http://www.molpro.net/info/current/examples/lif\\_nacme.com](http://www.molpro.net/info/current/examples/lif_nacme.com)

计算产生下面的表格：

Non-adiabatic couplings for LiF

R	NACME1P	NACME1M	NACMEAV	NACME2
10.0	-0.22828936	-0.22328949	-0.22578942	-0.22578942
10.5	-0.51777034	-0.50728914	-0.51252974	-0.51252974
11.0	0.76672943	0.76125391	0.76399167	0.76399167
11.5	0.42565202	0.42750263	0.42657733	0.42657733
12.0	0.19199878	0.19246799	0.19223338	0.19223338

注意，由于一个波函的相变，符号发生了改变。为了跟踪符号，必须检查轨道和CI矢量。

## 第三十七章

## 准透热化

DDR流程也可以用来对MRCI波函产生准透热态和能量（CASSCF的情况可以在MRCI中使用NOEXC指令，作为特殊情况处理）。准透热态的性质是：它们相对于参考结构几乎没有什么变化；换句话说，通过对给定的态进行酉变换，当前结构与参考结构的态重叠是最大化的。参考结构的透热态和绝热态最好应当相同，例如由于对称性。例如，下面的例子给出了H<sub>2</sub>S的<sup>1</sup>B<sub>1</sub>和<sup>1</sup>A<sub>2</sub>态，C<sub>2v</sub>结构用作参考结构，这个结构的态由于对称性不同而不会混合。在位移结构，分子对称性下降到C<sub>s</sub>。现在两个态由于都属于<sup>1</sup>A''不可约表示而发生强烈的混合。关于以下流程的说明和使用，参见D. Simah, B. Hartke, and H.-J. Werner, J. Chem. Phys. **111**, 4523 (1999)。

透热化可以自动执行，需要两步：首先，对CASSCF计算的活性轨道进行转动，使与参考结构轨道的重叠最大。这使用19.5.8节描述的DIAB流程完成。第二步，用DDR流程寻找CI矢量之间的变换。

需要以下的输入：

DDR	调用DDR流程。
ORBITAL, <i>orb1</i> , <i>orb2</i>	<i>orb1</i> 和 <i>orb2</i> 分别是当前结构和参考结构的（透热）轨道。
DENSITY, <i>trdm1</i> , <i>trdm2</i>	<i>trdm1</i> 是当前结构计算的跃迁密度， <i>trdm2</i> 是用当前波函（左矢）和参考结构波函（右矢）计算的跃迁密度。
MIXING, <i>state1</i> , <i>state2</i> , ...	给定的态将包含在透热化中。
ENERGY, <i>e1</i> , <i>e2</i> , ...	态的绝热能量。如果出现这个输入卡，将计算并打印以透热态为基的哈密顿量。也可以用Molpro的EADIA变量把能量传递给DDR。

如果发现了ENERGY指令或EADIA变量，将在以下的Molpro变量中打印和存储结果：

包含一阶轨道校正的结果：

SMAT	包含态重叠矩阵（用左矢指数运行最快）的前 $nstate \times nstate$ 个矩阵元。
UMAT	包含转换矩阵的前 $nstate \times nstate$ 个矩阵元。
H DIA	包含透热哈密顿量态倒三角矩阵的前 $nstate \cdot (nstate + 1)/2$ 个矩阵元。
MIXANG	非绝热混合角（单位：度）。只用于两个态的情况。

仅从CI矢量（没有轨道校正）获得的相应的结果存储在变量[SMATCI]，UMATCI，HDIACI，和MIXANGCI中。

使用程序的最简单方法是给一些例子作为示范。在以下的输入中，首先在 $C_{2v}$ 参考结构计算波函，接下来在位移结构计算。

```
! $Revision: 2009.1 $
***,h2s Diabatization
memory,3,m

gprint,orbitals,civector

symmetry,x
orient,noorient          ! 透热化计算一定要使用noorient
geometry={
    s;
    h1,s,r1;
    h2,s,r2,h1,theta}

basis=avdz               ! 这个基组对实际应用太小

r1=2.5                   ! 参考结构
theta=[92]

r=[2.50,2.55,2.60]      ! 位移结构

reforb=2140.2            ! 参考结构的轨道转存记录
refci=6000.2             ! 参考结构的MRCI记录
savci=6100.2            ! 位移结构的MRCI记录

text,compute wavefunction at reference geometry (C2v)
r2=r1
```

```

{hf;occ,9,2;wf,18,2,4;
orbital,2100.2}

{multi;occ,9,2;closed,4,1;
wf,18,2;state,2;           ! 1B1和1A2态
natorb,reforb              ! 把参考轨道保存到reforb
noextra}                   ! 不使用额外对称性

{ci;occ,9,2;closed,4,1;    ! 参考结构的MRCI
wf,18,2,0;state,2;        ! 1B1和1A2态
orbital,reforb             ! 使用前-CASSCF的轨道
save,refci}               ! 保存MRCI波函

Text,Displaced geometries

do i=1,#r                   ! 对不同的r值循环
data,truncate,savci+1      ! 截断参考结构后的转存文件
r2=r(i)                    ! 设定当前r2

{multi;occ,9,2;closed,4,1;
wf,18,2,0;state,2;        ! 定义波函
start,reforb               ! 初始轨道
orbital,3140.2;           ! 轨道转存记录
diab,reforb                ! 产生关于参考结构的透热轨道
noextra}                   ! 不使用额外对称性

{ci;occ,9,2;closed,4,1;
wf,18,2,0;state,2;        ! 1B1和1A2态
orbital,diabatic           ! 使用透热轨道
save,savci}                ! 保存位移结构的MRCI

e1(i)=energy(1)            ! 保存绝热能量
e2(i)=energy(2)

{ci;trans,savci,savci     ! 在R2计算跃迁密度
dm,7000.2}                ! 把跃迁密度保存到该记录
{ci;trans,savci,refci;    ! 计算R2和R1的跃迁密度
dm,7100.2}                ! 把跃迁密度保存到该记录

{ddr
density,7000.2,7100.2     ! <R2| |R2>和<R2| |R1>的密度
orbital,3140.2,2140.2    ! <R2| |R2>和<R2| |R1>的轨道
energy,e1(i),e2(i)        ! 绝热能量

```

```

mixing,1.2,2.2}          ! 计算混合角和透热能量

mixci(i)=mixangci(1)      ! 仅从ci矢量获得混合角
h11ci(i)=hdiaci(1)        ! 仅从ci矢量获得透热能量
h21ci(i)=hdiaci(2)
h22ci(i)=hdiaci(3)

mixtot(i)=mixang(1)       ! 从总重叠获得混合角 (包含一阶校正)
h11(i)=hdiaci(1)          ! 从总重叠获得透热能量
h21(i)=hdiaci(2)
h22(i)=hdiaci(3)

{table,r,e1,e2,h11ci,h22ci,h21ci,mixci
title,Diabatic energies for H2S, obtained from CI-vectors
format,'(f10.2,5f14.8,f12.2)'
sort,1}

{table,r,e1,e2,h11,h22,h21,mixtot
title,Diabatic energies for H2S, obtained from CI-vectors and orbital correction
format,'(f10.2,5f14.8,f12.2)'
sort,1}

enddo                    ! 结束i循环

```

[http://www.molpro.net/info/current/examples/h2s\\_diab1.com](http://www.molpro.net/info/current/examples/h2s_diab1.com)

计算产生下面的结果:

Diabatic energies for H2S, obtained from CI-vectors

R	E1	E2	H11CI	H22CI	H21CI	MIXCI
2.50	-398.64296319	-398.63384782	-398.64296319	-398.63384782	0.00000000	0.00
2.55	-398.64572746	-398.63666636	-398.64509901	-398.63729481	-0.00230207	15.27
2.60	-398.64911752	-398.63771802	-398.64662578	-398.64020976	-0.00471125	27.87

Diabatic energies for H2S, obtained from CI-vectors and orbital correction

R	E1	E2	H11	H22	H21	MIXTOT
2.50	-398.64296319	-398.63384782	-398.64296319	-398.63384782	0.00000000	0.00
2.55	-398.64572746	-398.63666636	-398.64509941	-398.63729441	-0.00230139	15.26
2.60	-398.64911752	-398.63771802	-398.64662526	-398.64021027	-0.00471160	27.88

第一个表的结果仅从CI对态重叠矩阵的贡献获得, 而第二个表的结果还包含了对轨道的一阶修正。在这个例子中, 两个结果几乎是相同的, 这是因为使用了DIAB流程,

使活性轨道的变化最小。这是推荐的流程。如果简单地使用没有进行轨道透热化的自然轨道，上面的计算将会得到以下的结果：

Diabatic energies for H2S, obtained from CI-vectors						
R	E1	E2	H11CI	H22CI	H21CI	MIXCI
2.50	-398.64296319	-398.63384782	-398.64296319	-398.63384782	0.00000000	0.00
2.55	-398.64572742	-398.63666630	-398.64475612	-398.63763760	-0.00280315	19.11
2.60	-398.64911746	-398.63771803	-398.64521031	-398.64162518	-0.00541050	35.83
Diabatic energies for H2S, obtained from CI-vectors and orbital correction						
R	E1	E2	H11	H22	H21	MIXTOT
2.50	-398.64296319	-398.63384782	-398.64296319	-398.63384782	0.00000000	0.00
2.55	-398.64572742	-398.63666630	-398.64509146	-398.63730226	-0.00231474	15.36
2.60	-398.64911746	-398.63771803	-398.64648358	-398.64035190	-0.00480493	28.73

可以看到仅来自CI矢量的混合现在完全不同，已经没有什么意义，因为作为结构函数的轨道发生了极大的改变。但是第二个计算近似地考虑了这一改变，仍给出了与包含透热轨道的计算相当一致的结果。

最后的例子给出了更复杂的输入，其中还计算了非绝热耦合矩阵元。在两个态的模型中，NACME应当等于混合角的一阶导数。在这个例子中，NACME用三点DDR 方法（NACMECI）以及使用混合角（DCHI）的有限差分方法进行计算。

```
! $Revision: 2009.1 $
***,h2s Diabatization and NACME calculation
memory,3,m

gprint,orbitals,civector

symmetry,x
orient,noorient          ! 透热化计算一定要使用noorient
geometry={
    s;
    h1,s,r1;
    h2,s,r2,h1,theta}

basis=avdz               ! 这个基组对实际应用太小

r1=2.5                   ! 参考结构
theta=[92]

r=[2.55,2.60]           ! 位移结构
```

```
dr=[0,0.01,-0.01]          ! 用于有限差分NACME计算的小位移

reforb1=2140.2              ! 参考结构的轨道转存记录
refci=6000.2               ! 参考结构的MRCI记录
savci=6100.2               ! 位移结构的MRCI记录

text,compute wavefunction at reference geometry (C2v)
r2=r1

{hf;occ,9,2;wf,18,2,4;orbital,2100.2}

{multi;occ,9,2;closed,4,1;
wf,18,2;state,2;          ! 1B1和1A2态
natorb,reforb1            ! 把参考轨道保存到reforb1
noextra}                  ! 不使用额外对称性

{ci;occ,9,2;closed,4,1;   ! 参考结构的MRCI
wf,18,2,0;state,2;       ! 1B1和1A2态
orbital,reforb1          ! 使用前一CASSCF计算的轨道
save,refci}              ! 保存MRCI波函

Text,Displaced geometries

do i=1,#r                  ! 对不同的r值循环
data,truncate,savci+1     ! 截断参考结构后的转存文件
reforb=reforb1

do j=1,3                  ! 对NACME做小位移循环
r2=r(i)+dr(j)             ! 设定当前r2

{multi;occ,9,2;closed,4,1;
wf,18,2,0;state,2;       ! 定义波函
start,reforb              ! 初始轨道
orbital,3140.2+j;        ! 轨道转存记录
diab,reforb               ! 产生关于参考结构的透热轨道
noextra}                  ! 不使用额外对称性

reforb=3141.2             ! j=1使用的轨道,用作j=2,3的参考轨道

{ci;occ,9,2;closed,4,1;
wf,18,2,0;state,2;
orbital,diabatic          ! 使用透热轨道
save,savci+j}            ! 保存位移结构的MRCI
```



```

eadia=energy                ! 保存绝热能量, 用于ddr
if(j.eq.1) then
e1(i)=energy(1)             ! 保存绝热能量, 用于打印表格
e2(i)=energy(2)
end if

{ci;trans,savci+j,savci+j; ! 在R2+DR(j)计算跃迁密度
dm,7000.2+j}               ! 把跃迁密度保存到该记录
{ci;trans,savci+j,refci;    ! 计算R2+DR(j)和R1的跃迁密度
dm,7100.2+j}               ! 把跃迁密度保存到该记录
{ci;trans,savci+j,savci+1;  ! 计算R和R2+DR(j)的跃迁密度
dm,7200.2+j}               ! 把跃迁密度保存到该记录

{ddr
density,7000.2+j,7100.2+j ! <R2+DR||R2+DR>和<R2+DR||R1>的密度
orbital,3140.2+j,2140.2   ! <R2+DR||R2+DR>和<R2+DR||R1>的轨道
energy,eadia(1),eadia(2)  ! 绝热能量
mixing,1.2,2.2}           ! 计算混合角和透热能量

if(j.eq.1) then            ! 对R2存储透热能量 (DR(1)=0)
mixci(i)=mixangci(1)       ! 仅从ci矢量获得混合角
h11ci(i)=hdiaci(1)        ! 仅从ci矢量获得透热能量
h21ci(i)=hdiaci(2)        ! HDIA包含透热哈密顿量的倒三角矩阵元
h22ci(i)=hdiaci(3)
mixtot(i)=mixang(1)       ! 从总重叠获得混合角 (包含一阶校正)
h11(i)=hdia(1)            ! 从总重叠获得透热能量
h21(i)=hdia(2)
h22(i)=hdia(3)
end if

mix(j)=mixang(1)           ! 存储R2+DR(j)的混合角

enddo                      ! 结束j循环

dchi(i)=(mix(3)-mix(2))/(dr(2)-dr(3))*pi/180 ! 混合角的有限差分指令

{ddr
density,7201.2,7202.2,7203.2 ! 用三点公式计算NACME
orbital,3141.2,3142.2,3143.2
states,2.2,1.2}
nacmec(i)=nacme

{table,r,mixci,mixtot,dchi,nacmec
Title,Mixing angles and non-adiabatic coupling matrix elements for H2S

```

```

format,'(f10.2,4f14.4)'
sort,1
}

{table,r,e1,e2,h11ci,h22ci,h21ci
Title,Diabatic energies for H2S, obtained from CI-vectors
format,'(f10.2,5f16.8)'
sort,1}

{table,r,e1,e2,h11,h22,h21
title,Diabatic energies for H2S, obtained from CI-vectors and orbital correction
format,'(f10.2,5f16.8)'
sort,1}

enddo

```

[http://www.molpro.net/info/current/examples/h2s\\_diab2.com](http://www.molpro.net/info/current/examples/h2s_diab2.com)

计算产生以下的表格：

Mixing angles and non-adiabatic coupling matrix elements for H2S

R	MIXCI	MIXTOT	DCHI	NACMECI
2.55	15.2694	15.2644	-5.2226	-5.2365
2.60	27.8740	27.8772	-3.4702	-3.4794

Diabatic energies for H2S, obtained from CI-vectors

R	E1	E2	H11CI	H22CI	H21CI
2.55	-398.64572746	-398.63666636	-398.64509901	-398.63729481	-0.00230207
2.60	-398.64911752	-398.63771802	-398.64662578	-398.64020976	-0.00471125

Diabatic energies for H2S, obtained from CI-vectors and orbital correction

R	E1	E2	H11	H22	H21
2.55	-398.64572746	-398.63666636	-398.64509941	-398.63729441	-0.00230139
2.60	-398.64911752	-398.63771802	-398.64662526	-398.64021027	-0.00471160

如所预料的，从三个点的DDR计算（NACMECI）和从混合角差分计算（DCHI）获得的耦合矩阵元非常接近。

## 第三十八章

## VB程序CASVB

CASVB是一个用于价键计算的一般程序，编写者是  
T. Thorsteinsson和D. L. Cooper (1996–2005)

这个程序可以按照两种基本模式使用：

- a) 一般类型的非正交MCSCF或者现代价键波函的变分优化。
- b) 现代价键形式的CASSCF波函表象，使用了基于重叠（相对较便宜）或者基于能量的标准。

参考文献：

- T. Thorsteinsson, D. L. Cooper, J. Gerratt, P. B. Karadakov and M. Raimondi, *Theor. Chim. Acta* **93**, 343–66 (1996).
- D. L. Cooper, T. Thorsteinsson and J. Gerratt, *Int. J. Quant. Chem.* **65**, 439–51 (1997).
- D. L. Cooper, T. Thorsteinsson and J. Gerratt, *Adv. Quant. Chem.* **32**, 51–67 (1998).
- T. Thorsteinsson and D. L. Cooper, in *Quantum Systems in Chemistry and Physics. Volume 1: Basic problems and models systems*, eds. A. Hernández-Laguna, J. Maruani, R. McWeeny, and S. Wilson (Kluwer, Dordrecht, 2000); pp 303–26.

使用该程序的所有出版物必须引用以上相应的文献。完整的参考文献参见  
<http://www.liv.ac.uk/~dlc/CASVB.html>。

## 38.1 输入的结构

所有的CASVB子命令都可以缩写为四个字母。一般的输入结构总结如下：

- a) 程序可以用CASVB命令调用，产生CASSCF波函数表象。对于波函数的变分优化，通常在MULTI内用子命令VB调用（参见19.10节）。
- b) 定义CASSCF波函数（一般不需要）。
- c) 定义价键波函数。
- d) 重新获得和/或存储轨道和矢量。
- e) 人工输入初始猜测（可选）。
- g) 优化控制。
- f) 定义VB波函数的分子对称性以及可能的束缚。
- h) 波函数分析。
- i) 更多的一般选项。

其中的a)和b)项应当在输入中位于其它项之前；除此之外，所有的命令可以按照任何顺序输入。

## 38.2 定义CASSCF波函数

CASVB连接到MULTI的行列式部分（即，不能定义CONFIG,CSF;）。当MULTI程序在CASVB之前运行的时候，CI矢量必须用SAVE, NATORB, CANONICAL或LOCALI指令中的一个进行转存（参见19.5.4节）。推荐后三个。

### 38.2.1 VBDUMP指令

VBDUMP[,vbdump];

如果出现VBDUMP卡，它必须在CASVB输入的最开始。对于变分计算不需要。

注意，在多数情况下（例如，如果在MULTI之后立即运行CASVB，或者对于变分计算），不需要用vbdump对转存记录进行另外说明。

在这里可以用VBDUMP卡重新得到波函数的定义（参见19.8.6节）。（vbdump）默认的记录名为4299.2。如果没有出现VBDUMP卡，并且记录4299.2不存在，那么CASVB会基于最后一次MCSCF计算自动尝试产生波函数信息（不过在这种情况下，将不会重新得到STATE和WEIGHT信息）。

### 38.3 其它的波函指令

CASSCF波函的说明也可以人工定义，使用以下指令：

OCC	占据轨道。
CLOSED	闭壳层轨道。
FROZEN	冻芯轨道。
WF	波函卡。
STATE	该波函对称性的态数。
WEIGHT	态的权重。

这些卡的准确说明参见19.2和19.3节。这些命令也可以用来修改VBDUMP中定义的值。这些卡给出的信息应当对应于CASSCF计算保存的CI矢量。因此，所用的卡以及它们的顺序应当和在MULTI中使用的一致，除了WEIGHT卡可以不同。目前，VB波函必须对应于精确定义的电子数和总自旋。也可以出现其它的态，但是如果对不同 $elec$ 或 $spin$ 值的波函对称性定义了非零权重，将会产生错误情况。

### 38.4 定义价键波函

#### 38.4.1 指定轨道组态

芯轨道和活性轨道数 ( $mcore$ ,  $mact$ )、活性电子数 ( $Nact$ )，以及总自旋的值，与CASSCF波函的定义相同。空间的VB组态仅对活性轨道，用一个或者多个CON卡进行定义（注意，在CASVB中不能使用RESTRICT和SELECT关键词）：

CON, $n_1, n_2, n_3, n_4, \dots$ ;

组态可以用占据数定义，方法与在MULTI中完全相同（参见19.4.3节），因此 $n_i$ 是第 $i$ 个价键轨道的占据。也可以给出 $Nact$ 个活性轨道编号的列表（顺序任意），由程序确定使用哪个定义。因此两个卡CON,1,0,1,2;和CON,1,3,4,4;是等价的。

如果没有指定组态，假定是单共价组态 $\phi_1\phi_2\cdots\phi_{Nact}$ 。

#### 38.4.2 选择自旋基

SPINBASIS, $key$ ;

$key$ 可以从KOTANI（默认），RUMER，PROJECT，或LTRUMER中选择，指定自旋本征函数的基，用于定义价键结构。PROJECT表示用自旋投影算符产生的自旋函数，LTRUMER表示使用所谓“主要项”相位约定的Rumer函数。

### 38.5 重新得到CASSCF CI矢量和VB波函

可以用START指令直接指定合适的MOLPRO记录（另一种方法是使用***vbdump***机制，说明见38.2.1节）：

```
START,ci,vb,orb,trnint;
```

<i>ci</i>	用于CASSCF CI矢量的记录名。CI矢量必须预先用 <b>SAVE</b> , <b>NATORB</b> , <b>CANONICAL</b> , 或 <b>LOCALI</b> 指令转存（参见19.5.4节）。 <i>ci</i> 的默认值从最近的 <b><i>vbdump</i></b> 记录中确定。 注意：如果没有找到 <i>ci</i> 记录，则对VB波函只执行基于能量的优化。
<i>vb</i>	由前一CASVB计算保存的价键轨道和结构因子的记录名。如果以前的VB波函保存在AO基中，那么轨道将投影到当前的活性空间（注意：要使用这一功能，需要对分子轨道指定记录名（见下面的 <i>orb</i> ））。
<i>orb</i>	定义CASSCF波函的分子轨道记录名。如果要输出原子轨道基的价键轨道，这一信息是必须的。
<i>trnint</i>	用于转换CASSCF积分的记录名。这对基于能量的标准是必须的（也就是说，如果指定了 <b>CRIT,ENERGY</b> ），并且可以在 <b>MULTI</b> 中用 <b>TRNINT</b> 子命令保存（参见19.8.7节）。在这里和在 <b>MULTI</b> 中，默认的记录名都是1900.1。

### 38.6 保存VB波函

```
SAVE,vb,civb,vbao;
```

<i>vb</i>	VB波函，也即轨道和结构因子的记录名（默认是3200.2之后第一个可用的记录）。
<i>civb</i>	按照CASSCF MO定义的价键完全CI矢量的记录名（默认是3300.2）。对于进一步的特性、几何优化计算等，需要保存这个矢量。
<i>vbao</i>	在AO基中，价键波函的记录名。注意这个关键词需要提前在START指令中指定 <i>orb</i> 。在绘制轨道图，或者对不同活性空间CASSCF的解提供初始猜测时，这可能会有用。

对*vb*，*civb*，和*vbao*，通常建议使用文件2中的记录。

## 38.7 指定猜测

GUESS;*key-1*,...;*key-2*,...;...

GUESS关键词开始价键轨道和结构因子猜测的输入。*key-i*可以是ORB, STRUC, 或READ。这些关键词修改由程序提供的, 或者由START指令指定的猜测。因此可以修改前一个解的个别轨道来构建初始猜测。

### 38.7.1 轨道初始猜测

ORB,*i*, *c*<sub>1</sub>, *c*<sub>2</sub>,...*c*<sub>mact</sub>;

指定第*i*个价键轨道的初始猜测。猜测按照定义CASSCF波函的*mact*个活性MO指定。(注意, 这些MO的定义依赖于如何转存CI矢量——也即, 使用SAVE, NATORB, CANONICAL, 或LOCALI中的哪个指令(参见19.5.4节)。推荐使用后三个中的一个。)

### 38.7.2 用于结构因子的猜测

STRUC,*c*<sub>1</sub>, *c*<sub>2</sub>,...*c*<sub>NVB</sub>;

对*NVB*个结构因子指定初始猜测。如果没有提供此卡, 并且没有用START指定猜测, 那么假定为自旋耦合的最佳配对模式, 它对空间组态具有最少的双占据轨道数。注意, 结构的定义依赖于SPINBASIS的值。在所有的组态中首先出现双占据轨道, 自旋本征函数基于能量上升顺序的单占据轨道。

### 38.7.3 读取轨道或结构因子

READ关键词可以是以下形式之一:

READ,ORB,*iorb1*[*TO*,*iorb2*] [*AS*,*jorb1*[*TO*,*jorb2*]] [*FROM*,*record*];

READ,STRUC,*istruc1*[*TO*,*istruc2*] [*AS*,*jstruc1*[*TO*,*jstruc2*]] [*FROM*,*record*];

READ,ALL [*FROM*,*record*];

用这种办法, 轨道和/或结构因子的子集就可以从前面的计算中选出来。轨道或结构的重新编号可以用上面描述的AS结构完成。如果以前在AO基中保存了VB波函, 那么轨道将被投影到当前的活性空间中(注意: 要使用这一功能, 需要对分子轨道指定记录名(在START命令中的*orb*))。

*record*默认是在START关键词中指定的*vb*记录名(如果可用)。

## 38.8 轨道改序

ORBPERM,*i*<sub>1</sub>,...*i*<sub>mact</sub>;

价键波函中的轨道改序，并按照 $\phi'_j = \text{sign}(i_j)\phi_{\text{abs}(i_j)}$ 改变其相位。初始猜测可以用关键词GUESS进一步修改。同样，结构因子将按照给定的交换进行变换（注意：要使用这一功能，进行轨道交换的组态列表必须是封闭的）。

## 38.9 优化控制

### 38.9.1 优化标准

CRIT,*method*;

指定优化的标准。*method*可以是OVERLAP或ENERGY（默认是OVERLAP）。前者使与CASSCF波函的归一化重迭最大：

$$\max \left( \frac{\langle \Psi_{\text{CAS}} | \Psi_{\text{VB}} \rangle}{(\langle \Psi_{\text{VB}} | \Psi_{\text{VB}} \rangle)^{1/2}} \right)$$

后者简单地使能量最小：

$$\min \left( \frac{\langle \Psi_{\text{VB}} | \hat{H} | \Psi_{\text{VB}} \rangle}{\langle \Psi_{\text{VB}} | \Psi_{\text{VB}} \rangle} \right).$$

### 38.9.2 迭代数量

MAXITER, $N_{\text{iter}}$ ;

指定在二阶优化中的最大迭代数。默认是 $N_{\text{iter}}=50$ 。

### 38.9.3 CASSCF的投影结构因子

(NO)CASPROJ;

使用这个关键词，从转换后的CASSCF CI矢量选出结构因子，只留下轨道变量参数。关于这个关键词的更多细节，参看文献。该选项可以帮助收敛。

### 38.9.4 鞍点优化

SADDLE,*n*;

定义*n*阶鞍点的优化。参见参看T. Thorsteinsson and D. L. Cooper, Int. J. Quant. Chem. **70**, 637–50 (1998).

### 38.9.5 定义多步优化

在同一个CASVB程序组中，使用OPTIM关键词可以执行一个以上的优化：

OPTIM[;...;FINOPTIM];



子命令可以是这一部分定义的任何优化指令，以及38.10节中的任何对称性或限制条件的说明。作为OPTIM的变量，这些命令对这个优化步骤来说是特定的，其中，外部指定的命令将会用作后面所有OPTIM关键词的默认定义。

如果只需要一个优化步骤，则不需要指定OPTIM关键词。

当只有一个机器产生的猜测可用时，CASVB将尝试定义一系列的优化步骤，这种选择使得成功收敛的可能性最大，并且CPU花费最小。要禁止这一功能，可以简单地指定一个或几个OPTIM卡。

### 38.9.6 多步优化

对一个或几个优化步骤进行循环，可以指定：

ALTERN,*Niter*,...;FINALTER

使用这个说明，程序将重复包含的优化步骤，直到所有的优化实现收敛，或者迭代计数器*Niter*达到最大。

## 38.10 点群对称性和束缚

关于对称匹配价键波函数的有关问题也在考虑中，例如，参见：T. Thorsteinsson, D. L. Cooper, J. Gerratt and M. Raimondi, Theor. Chim. Acta **95**, 131 (1997).

### 38.10.1 对称操作

SYMELM,*label*,*sign*;

开始定义对称性操作，使用*label*（可以是任意的三个字符）。*sign*可以是+或-；分别用来指定总波函在该操作下是对称的还是反对称的。*sign*的值并不总是必须的，但是如果给出，将会把束缚条件加在结构因子上，以保证波函具有正确的总对称性（注意：要使用这一功能，在*label*包含的用于交换的轨道中，组态列表必须是封闭的）。

算符是按照其对活性MO的作用定义的，这些MO由关键词IRREPS，COEFFS，或TRANS 中的一个或几个指定（其它的关键词会终止该对称性算符的定义）。如果没有提供更多关键词，则对*label*假定为全同的。也可以用另一种格式：

SYMELM,*label*,*sign*;key-1,...;key-2,...;...

### 38.10.2 IRREPS关键词

IRREPS,*i*<sub>1</sub>, *i*<sub>2</sub>,...;

列表*i*<sub>1</sub>, *i*<sub>2</sub>,... 指定有哪些不可约表示（和在CASSCF波函定义的相同）在*label*操作下是反对称的。如果一个不可约表示没有另外定义，那么假定在该对称操作下是对称的。

### 38.10.3 COEFFS关键词

COEFFS, $i_1, i_2, \dots$ ;

列表 $i_1, i_2, \dots$ 指定哪个CASSCF MO在 $label$ 操作下是反对称的。如果一个MO没有另外定义，那么假定在该对称操作下是对称的。这一定义可能会有用，例如，当分子具有的对称性比CASSCF计算使用的高时。

### 38.10.4 TRANS关键词

TRANS, $n_{dim}, i_1, \dots, i_{n_{dim}}, c_{11}, c_{12}, \dots, c_{n_{dim}n_{dim}}$ ;

指定一个包含由 $c$ 因子定义的对MO  $i_1, \dots, i_{n_{dim}}$  进行的一般变换 $n_{dim} \times n_{dim}$ 。对于有二维或三维不可约表示的体系，或者用局域化轨道定义CASSCF波函，这一定义可能会有用。注意，指定的变换必须总是正交的。

### 38.10.5 轨道之间的对称性关系

一般地，对于有纯对称性的VB波函，轨道必须形成对称群的表示（未必是不可约的）。在SYMELM定义的对称操作下，轨道之间关系的可以定义为：

ORBREL, $i_1, i_2, label1, label2, \dots$ ;

轨道 $i_1$ 和轨道 $i_2$ 通过 $label$ （已在前面用SYMELM定义）描述的操作关系相联系。算符的操作从右到左。注意， $i_1$ 和 $i_2$ 可以相同。只需要定义最少的联系。应当给出所有的轨道；如果出现了多余的ORBREL说明，将会出错退出。

### 38.10.6 SYMPROJ关键词

作为组合束缚的另一种选择，还可以用投影算符来保证波函的正确对称性：

(NO)SYMPROJ[ $irrep_1, irrep_2, \dots$ ];

这个关键词的作用是对不必要的不可约表示设置因子为零。为此，就要使用为CASSCF波函定义的对称群（总是 $D_{2h}$ 的子群）。命令中的不可约表示列表定义波函的哪些分量应当保留。如果没有给出不可约表示，假定为当前波函的对称性。在态平均计算中，保留所有的不可约表示，其中的非零权重在波函定义中指定。SYMPROJ关键词也可以和束缚结合使用。

### 38.10.7 在优化中冻结轨道

FIXORB, $i_1, i_2, \dots$ ;

这个命令为初始猜测冻结在列表 $i_1, i_2, \dots$  中指定的轨道。也可以用特殊关键词ALL或NONE。这些轨道将从优化流程中去掉，但是对给定的ORBREL关键词仍进行归一化和对称匹配。

### 38.10.8 在优化中冻结结构因子

FIXSTRUC, $i_1, i_2, \dots$ ;

对结构 $i_1, i_2, \dots$ 冻结因子。也可以用特殊关键词ALL或NONE。这些结构将从优化流程中去掉，但是对于归一化或任何出现的关于对称性的关键词，这些结构仍受影响。

### 38.10.9 从优化中删除结构

DELSTRUC, $i_1, i_2, \dots$ , [ALL], [NONE];

从波函中删除指定的结构。也可以用特殊关键词ALL或NONE。由于对称性（用SYMELM和ORBREL定义），一个结构因子可能已经是零，这种情况下删除它不会有影响。

### 38.10.10 正交束缚

ORTHCON; $key-1, \dots; key-2, \dots; \dots$

ORTHCON关键词开始输入价键轨道对之间的正交束缚。子关键词 $key-i$ 可以是下面描述的ORTH, PAIRS, GROUP, STRONG, 或FULL 中之一。正交束缚应当慎用。注意由其它对称性操作（用ORBREL关键词）产生的轨道，正交束缚一般不能满足。

ORTH, $i_1, i_2, \dots$ ;

指定进行正交化的一组轨道。列表中每对轨道之间的所有重叠都设置为零。

PAIRS, $i_1, i_2, \dots$ ;

指定正交化轨道对的简单列表。轨道 $i_1$ 与 $i_2, i_3, i_4$ 等进行正交。

GROUP, $label, i_1, i_2, \dots$ ;

定义与ORTH或PAIRS关键词一起使用的轨道组。轨道组可以用 $label$ 引用，它可以是a到z开始的任何三个字符。定义不同轨道组的标签可以一起使用，或者可以和ORTH或PAIRS中的轨道序号结合使用。 $i_1, i_2, \dots$ 指定组中的轨道列表。因此组合GROUP,AZZ,1,2; GROUP,BZZ,3,4; ORTH,AZZ,BZZ; 将会正交化轨道对1-3, 1-4, 2-3和2-4。

STRONG;

这个关键词是强正交的缩写。唯一允许的非零重叠是在轨道对 $(2n-1, 2n)$ 之间。

FULL;

这个关键词是完全正交的缩写。它主要用于测试目的。

## 38.11 波函分析

### 38.11.1 自旋关联分析

(NO)SCORR;

用这个选项，求解自旋算符 $(\hat{s}_\mu + \hat{s}_\nu)^2$ 的期待值，对所有的轨道对 $\mu$ 和 $\nu$ 。默认为NOSCORR。流程的描述参见：G. Raos, J. Gerratt, D. L. Cooper and M. Raimondi, Chem. Phys. **186**, 233–250 (1994); *ibid*, 251–273 (1994); D. L. Cooper, R. Ponc, T. Thorsteinsson and G. Raos, Int. J. Quant. Chem. **57**, 501–518 (1996).

目前，这个分析只能用于自旋耦合波函。

### 38.11.2 打印价键结构的权重

关于CASVB权重计算的更多细节，参见T. Thorsteinsson and D. L. Cooper, J. Math. Chem. **23**, 105–26 (1998).

VBWEIGHTS, *key1*, *key2*, ...

计算并输出价键波函 $\Psi_{VB}$ 中结构的权重。*key*指定所用的非正交权重的定义，可以是以下之一：

CHIRGWIN	求解Chirgwin-Coulson权重（参见：B. H. Chirgwin and C. A. Coulson, Proc. Roy. Soc. Lond. <b>A201</b> , 196 (1950)）。
LOWDIN	执行结构的对称正交，并输出相应的权重。
INVERSE	输出“反转重叠布居”，参见G. A. Gallup and J. M. Norbeck, Chem. Phys. Lett. <b>21</b> , 495–500 (1973)。
ALL	以上全部。
NONE	禁止计算结构权重。

命令LOWDIN和INVERSE需要价键结构之间的重叠矩阵，因此需要一些计算上的耗时。

### 38.11.3 在VB基中打印CASSCF波函的权重

关于CASVB中权重计算的细节，参见T. Thorsteinsson and D. L. Cooper, J. Math. Chem. **23**, 105–26 (1998).

CIWEIGHTS, *key1*, *key2*, ... [,  $N_{\text{conf}}$ ];

打印转换到非正交VB结构基中的CASSCF波函权重。对于*key*选项，参见上面的VBWEIGHTS。注意：对于大活性空间求解反转重叠权重，需要大量的计算耗时。对总CASSCF波函和 $\Psi_{VB}$ 的正交余项给出权重。所需组态的数量 $N_{\text{conf}}$ 默认是10。如果 $N_{\text{conf}}=-1$ ，则包含所有的组态。

## 38.12 控制输出的总量

PRINT,  $i_1, i_2, \dots$ ;

每个数值根据以下约定，指定在执行的各个阶段需要的输出级别：

-1	除了危险的或严重的出错信息以外，无输出。
0	最小输出。
1	标准级别的输出。
2	额外输出。

可以控制的输出范围有：

$i_1$	打印输入的参数，波函定义，等。
$i_2$	打印和对称性束缚有关的信息。
$i_3$	一般收敛过程。
$i_4$	二阶优化流程的过程。
$i_5$	打印收敛的解和分析。
$i_6$	变分优化的过程。
$i_7$	使用文件2中的记录编号。

默认的输出级别是+1。如果  $i_5 \geq 2$ ，VB轨道将在AO基中打印（假定MO的定义可用）；这种输出对绘制轨道图会特别有用。

## 38.13 更多工具

还可以用各种类型的直积波函和/或严格局域化的轨道执行计算。细节请联系作者。这些工具将在以后的发布中说明。

## 38.14 服务模式

SERVICE;

这个关键词优先于CASVB以前定义的任何关键词。它提供了重新得到轨道因子和VB结构因子的简单工具。在运行CASVB过程中不要使用，因为它已经从MULTI内部调用了。

START, *record.file*;

因子来自 *record.file*。默认值是2100.2。

WRITE, *iwrite*;

对称性轨道基中的矢量写入通道*iabs*(*iwrite*)中。默认的操作是把这些矢量写到标准输出中。如果*iwrite*为负，那么矢量作为单独的记录，改为写到二进制文件中。

SPECIAL,*idim1*,*idim2*,*idim3*,*idim4*;

如果出现，这个关键词必须在最后。首先在跳过*idim4*个元素之后，程序尝试从*record.file*获得长度为*idim1\*idim2+idim3*的矢量。依照*iwrite*的设置写入矢量。(默认的*idim*值是零。)

### 38.15 例子

```
***, ch2                      ! A1单重态
geometry={angstrom
c
h1,c,1.117
h2,c,1.117,h1,102.4}
int
hf
{multi;occ,4,1,2;closed,1      ! 6电子6轨道的CASSCF
natorb,,ci,save=3500.2;vbdump}
{casvb                          ! 基于重叠的VB,
save,3200.2}                    ! 使用自旋耦合波函
{casvb                          ! 基于能量的VB计算
start,,3200.2;save,3220.2
crit,energy}
{multi;occ,4,1,2;closed,1      ! 完全变分VB计算
{vb;start,,3220.2;save,3240.2;print,,,,2}}
---
```

```
memory,4,m
***,n2s2 (model a)           ! N2S2的变分计算
geometry={x,y,z;
a1,n,-2.210137753,0,0;        ! 注意: 活性空间的其它选择给出
a2,n,+2.210137753,0,0;        ! 互相排斥的模型
a3,s,0,-2.210137753,0;
a4,s,0,+2.210137753,0}
basis=VTZ;
cartesian
{hf;wf,46,1}
{multi;occ,7,4,5,2,4,2,2,0;closed,7,4,5,2,1,0,1,0; natorb,,ci,save=3500.2}
{multi;occ,7,4,5,2,4,2,2,0;closed,7,4,5,2,1,0,1,0; vb}
---
```

```
***, lih          ! 完全变分VB计算, 和
r=2.8,bohr        ! 结构优化
basis={
s,1,921.300000,138.700000,31.940000,9.353000,3.158000,1.157000;
k,1.6,0.001367,0.010425,0.049859,0.160701,0.344604,0.425197;
s,1,0.444600,0.076660,0.028640;
p,1,1.488000,0.266700,0.072010,0.023700;
k,1.2,0.038770,0.236257;
s,2,13.36,2.013,0.4538,.1233;
k,1.2,0.032828,0.231204;}
geometry={li;h,li,r}
int;
{hf;wf,4,1}
{multi
occ,4,0,0,0
closed,0,0,0,0
natorb,,ci,save=3500.2}
{multi;maxiter,20
vb}
optg
---
```





### 39.1 简介

自旋轨道矩阵元和本征态可以用Breit-Pauli (BP) 算符或自旋轨道赝势 (ECP) 计算。计算执行态相互作用方法，这意味着自旋轨道本征态通过在本征函数 $\hat{H}_{el}$ 的基中对角化 $\hat{H}_{el} + \hat{H}_{SO}$ 获得。完全Breit-Pauli SO算符只能用于MCSCF波函。对于MRCI波函，完全BP算符用于计算内部组态（外部轨道没有电子）之间的矩阵元，而对外部组态的贡献使用有限场单电子Fock算符。这一近似导致的误差通常小于 $1 \text{ cm}^{-1}$ 。

程序既可以对一对给定的电子态计算各个自旋轨道矩阵元，也可以对给定的一组电子态自动设置和对角化整个矩阵。对于后一种情况，也计算单电子算符的矩阵元，并转换到自旋轨道本征态中（默认计算偶极矩阵元；其它算符可以在GEXPEC或EXPEC卡指定，见6.13节）。使用比能量计算小的基计算自旋轨道矩阵元一般足够，因此还可以用预先计算的值替换能量本征值，这通过设置MOLPRO变量HLSDIAG传递给自旋轨道程序。

### 39.2 计算SO积分

BP哈密顿量的单、双电子自旋轨道积分可以预先计算并存储到硬盘上，使用命令：

```
LSINT [,X] [,Y] [,Z] [,ONECENTER] [:TWOINT,twoint:] [:PREFAC,prefac:]
```

X, Y, 和Z定义计算的分量。如果没有给出，三个全部求解。预先计算积分的好处是它们可以用于后面的任何SO计算，但是这可能需要大量的硬盘空间（注意这是在能量计算中积分的六倍）。如果没有给出LSINT卡，那么只要需要就会计算积分。ONECENTER关键词激活单、双电子自旋轨道积分的单中心近似，对于大分子这会极大地

减少计算时间。TWOINT和PREFAC 可用于控制自旋轨道积分的精度。这些阈值类似于标准积分的TWOINT和PREFAC。PREFAC的默认值是TWOINT/100，TWOINT的默认值是 $10^{-7}$ 。当没有积分被预先计算时，这些阈值可以指定为HLSMAT或TRANLS卡的选项，见下。

自旋轨道ECP的输入在第十二章有说明。当然，在ECP-LS计算中，不需要LSINT卡。

### 39.3 计算各个SO矩阵元

各个自旋轨道矩阵元可以在MRCI程序中计算，使用：

TRANLS,record1.file, record2.file, bra2ms, ket2ms, lsop;

其中，

<i>record1.file</i>	存有左矢波函数的记录。
<i>record2.file</i>	存有右矢波函数的记录。这两个记录必须用MRCI程序的SAVE指令产生。
<i>bra2ms</i>	左矢波函数的 $2 \times M_S$ 值。
<i>ket2ms</i>	右矢波函数的 $2 \times M_S$ 值。
<i>lsop</i>	自旋轨道哈密顿量的笛卡尔分量。 在全电子计算中，这可以是LSX, LSY, 或LSZ，在ECP计算中可以是ECPLSX, ECPLSY, 或ECPLSZ。从MOLPRO 2008.1版开始，可以用更多的类型控制近似级别。说明见39.4节。

由于自旋轨道程序是MRCI程序的一部分，因此TRANLS卡必须在[MR]CI卡之后。对于用MCSCF波函数计算矩阵元的情况，还必须用MRCI程序，通过使用NOEXC指令避免包含MCSCF参考函数之外的更多激发，重新计算和保存CI矢量（见第二十章）。如果在MRCI步骤中，用STATE指令同时计算对称性相同的几个态，则对所有这些态计算矩阵元。注意，在TRANLS计算中，OCC和CLOSED卡对所有的态必须相同。

$M_S$ 值的选择定则对LSX和LSY算符是 $\Delta M_S = \pm 1$ ，对LSZ算符是 $\Delta M_S = 0$ 。注意必须指定 $2M_S$ ，因此用于输入值之差的选择定则是0或2。

在全电子SO计算中，计算的自旋轨道矩阵元的值（用原子单位）对x, y, z分量分别保存到MOLPRO变量TRLsx, TRLSy 和TRLsz中。对于ECP-LS计算，使用变量TRECPLSX, TRECPLSY和TRECPLSZ。注意，对于虚矩阵元（即SO 哈密顿量的x和z分量），矩阵元是虚数，因此存储的实数必须乘以*i*。如果计算了几个态的矩阵元，那么所有的值都用计算最快的左矢态存储在各自的变量数组中。

### 39.4 在自旋轨道积分和矩阵元的计算中使用近似

最近引入了更复杂的近似，以简化大分子的自旋轨道计算。它们通过指定自旋轨道算符类型 $lsop$ 来控制，如下（我们忽略了指定分量的后缀X, Y, Z）：

LS	标准的自旋轨道计算。
ALS	对单、双电子自旋轨道积分使用单中心近似。
FLS	对内部积分还使用有效Fock矩阵近似。
AFLS AMFI	单中心近似用于单、双电子自旋轨道积分，对内部积分使用有效Fock矩阵近似。
ECPLS	对于定义了芯势的所有原子使用有效芯势，忽略所有其它原子的贡献（见下）。

对于有效Fock矩阵用于所有贡献，并且没有预先计算自旋轨道积分并存储到硬盘（即，没有给出LSINT命令）的情况，在直接模式下求解Fock矩阵，并且不会有积分存到硬盘上。当这与单中心近似（AMFI）组合时，可以极大地减少计算时间和I/O时间，使得自旋轨道计算即使对大分子也相当快。

还修改了ECP型自旋轨道相互作用的处理，现在允许在一个计算中同时处理ECP和非ECP原子。因此在同时包含重原子和轻原子的分子中，重原子可以用ECP描述，而轻原子用全电子基组描述。如果算符类型是LS, ALS, FLS, AFLS, 那么对于在基组输入中定义了ECP自旋轨道算符的原子，将使用ECP算符，而对所有其它的原子使用完全BP算符（忽略耦合）。在这种情况下单中心近似和AMFI近似均可使用。另一方面，如果指定的算符类型为ECPLS，那么与以前版本的处理相同，即，只考虑ECP的贡献，忽略所有其它原子的贡献。

### 39.5 整个SO矩阵的计算和对角化

HLSMAT, *type*, *record1*, *record2*, *record3*, ...

使用记录*record1*, *record2*, *record3*, ...中所有的态，计算整个SO矩阵元并进行对角化。所有的记录必须用MRCI程序的SAVE指令产生。*type*既可以是用于Breit-Pauli计算的LS，也可以是用于ECP-LS计算的ECP。默认打印本征值以及基态与激发态之间的偶极跃迁矩阵元。

和使用TRANLS卡相同，HLSMAT卡仅能被MRCI程序识别，因此必须位于CI卡之后。此外，对于HLSMAT计算中使用的所有态，OCC和CLOSED卡必须相同。

## 39.6 修改非微扰能量

使用比能量计算小一些的基组或者在低一些的计算级别计算自旋轨道矩阵元，通常已经足够了。因此可以用预先计算的数值代替能量本征值，通过使用MOLPRO的变量HLSDIAG传递给自旋轨道程序。在HLSDIAG中，能量值的顺序必须和HLSMAT卡给定的记录中的态相同。在任何自旋轨道计算之前，变量HLSDIAG 要么是未定义的或清空的（因此使用原始能量），要么包含的能量数量与在随后自旋轨道计算中处理的态数必须完全相同（使用CLEAR,HLSDIAG可以清除变量以前的值）。用户应当保证HLSDIAG中的能量顺序必须正确！

### 39.6.1 自旋轨道计算的打印选项

PRINT, *option*<sub>1</sub>=*value*<sub>1</sub>, *option*<sub>2</sub>=*value*<sub>2</sub>, ...

其中的选项可以是：

HLS	HLS=-1, 只打印SO能量以及基态和激发态之间的跃迁矩阵元（默认）。
	HLS ≥ 0, 打印SO矩阵。
	HLS ≥ 1, 打印特性矩阵。
	HLS ≥ 2, 打印各个矩阵元（同OPTION,MATEL）。
	HLS ≥ 3, 打印调试信息。
VLS	VLS=-1, 不打印本征矢（默认）。
	VLS ≥ 0, 打印本征矢。

### 39.6.2 自旋轨道计算的选项

有些选项可以使用OPTION指令设置（以任何次序）：

OPTIONS [,WIGNER=*value*] [,HLSTRANS=*value*] [,MATEL=*value*]

其中，

WIGNER	这个选项确定在计算SO矩阵时是否使用Wigner-Eckart定理。WIGNER=1（默认）使用定理，WIGNER=0分别计算每个SO矩阵元。这个选项仅用于测试的目的。
HLSTRANS	这个选项确定SO矩阵计算是对非自旋对称匹配基组（HLSTRANS=0），自旋对称匹配基组（HLSTRANS=1，默认），还是对二者（HLSTRANS=2）都执行。目前对称匹配只能用于三重态，下面的符号系统用于确定对称匹配自旋函数： $ S, M_S\rangle_+ = \frac{1}{\sqrt{2}}( S, M_S\rangle +  S, -M_S\rangle)$ ， $ S, M_S\rangle_- =$

$\frac{1}{\sqrt{2}}(|S, M_S\rangle - |S, -M_S\rangle)$ 。如果只考虑单重和三重态，则自旋轨道矩阵按照双群对称性分块，每个块的本征值分别打印。对其它情况忽略HLSTRANS选项。

**MATEL** 如果用HLSMAT计算整个SO矩阵，那么通常不显示各个矩阵元。当给定了选项MATEL=1时，会打印各个矩阵元以及内部和外部组态类的贡献。

## 39.7 例子

### 39.7.1 对S原子用BP算符计算SO

```
! $Revision: 2006.0 $
***,SO calculation for the S-atom
geometry={s}
basis={spd,s,vtz}                                ! 使用非收缩基组

{rhf;occ,3,2,2,,2;wf,16,4,2}                      ! rhf计算3P态

{multi                                              ! casscf
wf,16,4,2;wf,16,6,2;wf,16,7,2;wf,16,1,0;state,3; ! 1D和1S态
wf,16,4,0;wf,16,6,0;wf,16,7,0}                  ! 3P态

{ci;wf,16,1,0;save,3010.1;state,3;noexc}          ! 用mrci保存casscf波函
{ci;wf,16,4,0;save,3040.1;noexc}
{ci;wf,16,6,0;save,3060.1;noexc}
{ci;wf,16,7,0;save,3070.1;noexc}
{ci;wf,16,4,2;save,3042.1;noexc}
{ci;wf,16,6,2;save,3062.1;noexc}
{ci;wf,16,7,2;save,3072.1;noexc}

{ci;wf,16,1,0;save,4010.1;state,3}                ! 1D, 1S态的mrci计算
ed=energy(1)                                       ! 把1D态的能量保存到变量ed
es=energy(3)                                       ! 把1S态的能量保存到变量es
{ci;wf,16,4,2;save,4042.1}                        ! 3P态的mrci计算
ep=energy                                          ! 把3P态的能量保存到变量ep
{ci;wf,16,6,2;save,4062.1}                        ! 3P态的mrci计算
{ci;wf,16,7,2;save,4072.1}                        ! 3P态的mrci计算
text,only triplet states, casscf

lsint                                              ! 计算so积分

text,3P states, casscf
{ci;hlsmat,ls,3042.1,3062.1,3072.1}              ! 仅计算三重态的casscf
```

```

text,3P states, mrci
{ci;hlsmat,ls,4042.1,4062.1,4072.1}          ! 仅计算三重态的mrci

text,3P, 1D, 1S states, casscf
{ci;hlsmat,ls,3010.1,3040.1,3060.1,3070.1,3042.1,3062.1,3072.1}      ! 全部态用casscf能量

text,only triplet states, use mrci energies and casscf SO-matrix elements
hlsdiag=[ed,ed,es,ed,ed,ed,ep,ep,ep]          ! 把hlsdiag设定为mrci能量
{ci;hlsmat,ls,3010.1,3040.1,3060.1,3070.1,3042.1,3062.1,3072.1}

```

[http://www.molpro.net/info/current/examples/s\\_so.com](http://www.molpro.net/info/current/examples/s_so.com)

### 39.7.2 对I原子用ECP计算SO

```

! $Revision: 2006.0 $
***,I
memory,5,M;
gprint,orbitals,civector,basis;
gthresh,energy=1.d-8,coeff=1.d-8;
geometry={I};

basis={
!
! 碘的包含SO耦合的ECP (Dirac-Fock)
!
ecp,I,46,4,3;
1; 2, 1.00000000, 0.00000000;          ! 局域项 = 0
2; 2, 3.50642001, 83.09814545; 2, 1.74736492, 5.06370919;      ! s项
4; 2, 2.99860773, 1/3* 81.88444526; 2, 3.01690894, 2/3* 83.41280402;  ! p项和权重
    2, 1.59415934, 1/3* 2.32392477; 2, 1.19802939, 2/3* 2.72079843;
4; 2, 1.03813792, 2/5* 6.40131754; 2, 1.01158599, 3/5* 6.21328827;  ! d项和权重
    2, 2.04193864, 2/5* 19.11604172; 2, 1.99631017, 3/5* 19.08465909;
4; 2, 2.64971585, -3/7* 24.79106489; 2, 2.75335574, -4/7* 24.98147319;  ! f项和权重
    2, 0.49970082, -3/7* 0.27936581; 2, 0.79638982, -4/7* 0.70184261;
4; 2, 2.99860773, -2/3* 81.88444526; 2, 3.01690894, 2/3* 83.41280402;  ! p项的ECP-SO
    2, 1.59415934, -2/3* 2.32392477; 2, 1.19802939, 2/3* 2.72079843;
4; 2, 1.03813792, -2/5* 6.40131754; 2, 1.01158599, 2/5* 6.21328827;  ! d项的ECP-SO
    2, 2.04193864, -2/5* 19.11604172; 2, 1.99631017, 2/5* 19.08465909;
4; 2, 2.64971585, 2/7* 24.79106489; 2, 2.75335574, -2/7* 24.98147319;  ! f项的ECP-SO
    2, 0.49970082, 2/7* 0.27936581; 2, 0.79638982, -2/7* 0.70184261;
!
! 碘的基组
!
s,I,0.2027624,0.4080619,0.8212297,1.6527350,3.3261500;
c,1.5,-0.4782372,-0.5811680,0.2617769,0.4444120,-0.1596560;

```

```

s,I,0.05,0.1007509;
p,I,0.2027624,0.4080619,0.8212297,1.6527350,3.3261500;
c,1.5,0.4251859,0.2995618,0.0303167,-0.2064228,0.0450858;
p,I,0.05,0.1007509,0.01; ! diffuse p-Funktion wegen evt. neg. Part.Ldg
d,I,0.2,0.4;
f,I,0.3;
}

{hf;occ,1,1,1,,1;wf,7,5,1}      ! 2Pz态的SCF
{multi;occ,1,1,1,,1;            ! 使用最小活性空间的casscf
wf,7,2,1;wf,7,3,1;wf,7,5,1}    ! 平均的2P态
{ci;wf,7,2,1;noexc;save,5000.2} ! 对2Px态保存casscf矢量
{ci;wf,7,3,1;noexc;save,5100.2} ! 对2Py态保存casscf矢量
{ci;wf,7,5,1;noexc;save,5200.2} ! 对2Pz态保存casscf矢量
{ci;wf,7,2,1;save,6000.2}       ! 2Px态的mrci
{ci;wf,7,3,1;save,6100.2}       ! 2Py态的mrci
{ci;wf,7,5,1;save,6200.2}       ! 2Pz态的mrci

{multi;occ,1,2,2,,2            ! 大活性空间的casscf
wf,7,2,1;wf,7,3,1;wf,7,5,1}    ! 平均的2P态
{ci;wf,7,2,1;noexc;save,5010.2}
{ci;wf,7,3,1;noexc;save,5110.2}
{ci;wf,7,5,1;noexc;save,5210.2}
{ci;wf,7,2,1;save,6010.2}
{ci;wf,7,3,1;save,6110.2}
{ci;wf,7,5,1;save,6210.2}

text,casscf, occ,1,1,1,,1
{ci;hlsmat,ecp,5000.2,5100.2,5200.2} ! 进行自旋轨道计算
text,casscf, occ,1,2,2,,2
{ci;hlsmat,ecp,5010.2,5110.2,5210.2}

text,mrci, occ,1,1,1,,1
{ci;hlsmat,ecp,6000.2,6100.2,6200.2}
text,mrci, occ,1,2,2,,2
{ci;hlsmat,ecp,6010.2,6110.2,6210.2}

```

[http://www.molpro.net/info/current/examples/i\\_ecp.com](http://www.molpro.net/info/current/examples/i_ecp.com)

```

! $Revision: 2006.0 $
***,I
memory,5,M;

gprint,orbitals,civector,basis;
gthresh,energy=1.d-8,coeff=1.d-8;

```

```

geometry={I};

basis={
!
! 碘的包含SO耦合和cpp的ECP (Dirac-Fock)
!
ecp,I,46,4,3;
1; 2, 1.00000000, 0.00000000; ! 局域项 = 0
2; 2, 3.50642001, 83.09814545; 2, 1.74736492, 5.06370919; ! s项
4; 2, 2.99860773, 1/3* 81.88444526; 2, 3.01690894, 2/3* 83.41280402; ! p项和权重
    2, 1.59415934, 1/3* 2.32392477; 2, 1.19802939, 2/3* 2.72079843;
4; 2, 1.03813792, 2/5* 6.40131754; 2, 1.01158599, 3/5* 6.21328827; ! d项和权重
    2, 2.04193864, 2/5* 19.11604172; 2, 1.99631017, 3/5* 19.08465909;
4; 2, 2.64971585, -3/7* 24.79106489; 2, 2.75335574, -4/7* 24.98147319; ! f项和权重
    2, 0.49970082, -3/7* 0.27936581; 2, 0.79638982, -4/7* 0.70184261;
4; 2, 2.99860773, -2/3* 81.88444526; 2, 3.01690894, 2/3* 83.41280402; ! p项的ECP-SO
    2, 1.59415934, -2/3* 2.32392477; 2, 1.19802939, 2/3* 2.72079843;
4; 2, 1.03813792, -2/5* 6.40131754; 2, 1.01158599, 2/5* 6.21328827; ! d项的ECP-SO
    2, 2.04193864, -2/5* 19.11604172; 2, 1.99631017, 2/5* 19.08465909;
4; 2, 2.64971585, 2/7* 24.79106489; 2, 2.75335574, -2/7* 24.98147319; ! f项的ECP-SO
    2, 0.49970082, 2/7* 0.27936581; 2, 0.79638982, -2/7* 0.70184261;
!
! 碘的基组
!
s,I,0.2027624,0.4080619,0.8212297,1.6527350,3.3261500;
c,1.5,-0.4782372,-0.5811680,0.2617769,0.4444120,-0.1596560;
s,I,0.05,0.1007509;
p,I,0.2027624,0.4080619,0.8212297,1.6527350,3.3261500;
c,1.5,0.4251859,0.2995618,0.0303167,-0.2064228,0.0450858;
p,I,0.05,0.1007509,0.01; ! diffuse p-Funktion wegen evt. neg. Part.Ldg
d,I,0.2,0.4;
f,I,0.3;
}

{cpp,init,1; ! 芯极化势
I,2,1.028,,1.23} ! 碘原子, 截断函数形式, 静态极化率,
! 1.23 = 截断函数的指数因子

{hf;occ,1,1,1,,1;wf,7,5,1} ! 2Pz态的SCF
{multi;occ,1,1,1,,1; ! 使用最小活性空间的casscf
wf,7,2,1;wf,7,3,1;wf,7,5,1} ! 平均的2P态
{ci;wf,7,2,1;noexc;save,5000.2} ! 对2Px态保存casscf矢量
{ci;wf,7,3,1;noexc;save,5100.2} ! 对2Py态保存casscf矢量
{ci;wf,7,5,1;noexc;save,5200.2} ! 对2Pz态保存casscf矢量

```



```
{ci;wf,7,2,1;save,6000.2}      ! 2Px态的mrci
{ci;wf,7,3,1;save,6100.2}      ! 2Py态的mrci
{ci;wf,7,5,1;save,6200.2}      ! 2Pz态的mrci

{multi;occ,1,2,2,,2            ! 大活性空间的casscf
wf,7,2,1;wf,7,3,1;wf,7,5,1}    ! 平均的2P态
{ci;wf,7,2,1;noexc;save,5010.2}
{ci;wf,7,3,1;noexc;save,5110.2}
{ci;wf,7,5,1;noexc;save,5210.2}
{ci;wf,7,2,1;save,6010.2}
{ci;wf,7,3,1;save,6110.2}
{ci;wf,7,5,1;save,6210.2}

text,casscf,  occ,1,1,1,,1
{ci;hlsmat,ecp,5000.2,5100.2,5200.2}  ! 进行自旋轨道计算
text,casscf,  occ,1,2,2,,2
{ci;hlsmat,ecp,5010.2,5110.2,5210.2}

text,mrci,  occ,1,1,1,,1
{ci;hlsmat,ecp,6000.2,6100.2,6200.2}
text,mrci,  occ,1,2,2,,2
{ci;hlsmat,ecp,6010.2,6110.2,6210.2}
```

[http://www.molpro.net/info/current/examples/i\\_ecp\\_cpp.com](http://www.molpro.net/info/current/examples/i_ecp_cpp.com)



### 40.1 解析能量梯度

MOLPRO使用两个不同的梯度程序：

Cadpac梯度程序，基于R. D. Amos的积分程序Cadpac。目前，这个程序可用于闭壳层SCF，高自旋RHF，以及（态平均的）MCSCF。在MCSCF情况下，波函必须要是完全优化的，要么冻芯轨道来自闭壳层SCF 计算（但这不能用于态平均的MCSCF）。注意，Cadpac不能用于一般收缩基函数。

Alaska梯度程序基于R. Lindh的积分程序Seward。它可以用一般收缩基函数对闭壳层SCF，开壳层RHF，UHF，RKS，UKS，MCSCF，MP2，LMP2，DF-LMP2，QCISD，QCISD(T)，以及RS2 (CASPT2)计算梯度。态平均MCSCF波函的梯度可以用RS2梯度程序进行计算，参见40.1.5节。关于CASPT2梯度的细节，参见21.7节。

只要可能，默认使用Alaska梯度。但是，还可以在调用梯度程序之前通过定义变量GRADTYP来强制使用特定的梯度程序：

```
GRADTYP=ALASKA
```

```
GRADTYP=CADPAC
```

梯度程序的调用使用FORCE命令：

```
FORCE
```

通常不需要FORCE命令，因为几何优化应当用OPTG流程执行。例外是平衡校正能量的优化，这需要几个力的计算（参见41.4.7节）。

如果没有给出进一步的数据卡，默认求解最后一次优化波函的梯度。在这种情况下，常规的梯度不需要进一步的输入（程序会记住存有波函信息的记录）。一个例外是在特殊情况下，在前一个MCSCF计算中进行了几个不同的CPMCSCF计算。在这种情况下

下必须使用SAMC指令选择所需的记录。如果最后一次的波函不能使用解析梯度，则计算数值梯度。关于数值能量梯度的更多细节，参见40.2 节。

#### 40.1.1 加入梯度 (ADD)

ADD,*factor*,[NOCHECK];

如果出现这个卡，当前的梯度和能量用给定的因子添加到前一个梯度和能量上。这对于平衡校正能量的优化很有用（参见41.4.7节）。除非在结构输入中指定了NOORIENT，否则程序默认将停止并给出出错信息。这一特性可以用NOCHECK 选项禁止。只有所有被添加的梯度是在完全相同的核坐标下求解得到的，才需要给出这一选项；否则由于体系的意外转动，会导致错误的结果。

#### 40.1.2 缩放梯度 (SCALE)

SCALE,*factor*;

如果出现这个卡，当前的梯度和能量根据给定的因子缩放。这对平衡校正能量的优化有时会很有用（参见41.4.7节）。

#### 40.1.3 为SCF梯度定义轨道 (ORBITAL)

ORBITAL,*record.file*;

在SCF情况下，*record.file*指定轨道的位置，它可用于构建密度矩阵，等。只有当需要进行梯度计算的SCF不是最近的能量计算时，才需要这个卡。

对于MCSCF波函不需要ORBITAL卡，因为轨道的位置存储于MCSCF的转存记录中。

#### 40.1.4 MCSCF梯度 (MCSCF)

MCSCF,*record.file*;

MCSCF梯度的触发代码。*record.file*指定MCSCF程序用SAVE,GRD=*recmc.filmc*卡转存的信息的位置。如果FORCE命令直接出现在相应的MCSCF输入之后，则不需要这个卡，因为程序会自动记住MCSCF信息存储的位置。如果使用OPTG也一样。

#### 40.1.5 用SEWARD计算态平均MCSCF梯度

通过使用Seward和RS2梯度程序，SA-MCSCF梯度可以用分块收缩或一般收缩基组计算。在RS2输入中必须使用NOEXC指令，但是在MULTI中不需要CPMCSCF卡。RS2梯度程序自动进行CP-MCSCF计算。

例子：对OH的 $^2\Pi$ 和 $^2\Sigma^+$ 态计算SA-CASSCF梯度。

```
! $Revision: 2006.3 $
geometry={o;h,o,r}
r=1.83
{multi;wf,9,2,1;wf,9,3,1;wf,9,1,1}      ! X(2PI)和A(2SIGMA)的态平均casscf
{rs2;noexc;wf,9,1,1}                    ! 计算A(2SIGMA)能量
forces                                    ! A(2SIGMA)态的能量梯度
{rs2;noexc;wf,9,2,1}                    ! 计算A(2PI)能量
forces                                    ! A(2PI)态的能量梯度
```

[http://www.molpro.net/info/current/examples/oh\\_samcforce.com](http://www.molpro.net/info/current/examples/oh_samcforce.com)

如果没有NOEXC指令，将会用态平均轨道求解RS2（CASPT2）梯度。

#### 40.1.6 用CADPAC计算态平均MCSCF梯度

当使用Cadpac时，对于态平均MCSCF的梯度计算，通常不需要进一步的输入。但是注意，在SA-MCSCF计算中需要CPMCSCF,GRAD,state指令（参见19.9节）。接下来对CPMCSCF卡指定的态自动计算梯度。这同样也适用于差分梯度（CPMCSCF,DGRAD,state1,state2）和非绝热耦合矩阵元（CPMCSCF,NACM,state1,state2）。在同一个MCSCF中可以依次进行几个耦合微扰MCSCF计算。在这种情况下FORCE默认使用最后一个解。在某些记录中（默认是5101.1, 5102.1, ...），来自CPMCSCF的某些信息传递给FORCE程序。如果在同一个MCSCF中执行了几个CPMCSCF计算，那么就会出现几个这样的记录，其中的特定记录可以在FORCE程序中用SAMC指令访问：

SAMC,record

SAMC的别名是CPMC。为了和早期版本兼容，还可以使用

NACM,record

用于非绝热耦合，或者

DEMC,record

用于差分梯度。

例子：

```
multi;
....
state,3
cpmcscf,nacm,1.1,2.1,save=5101.1      ! 对态1.1 - 2.1的耦合进行cpmcscf计算
cpmcscf,nacm,1.1,3.1,save=5102.1      ! 对态1.1 - 3.1的耦合进行cpmcscf计算
cpmcscf,nacm,2.1,3.1,save=5103.1      ! 对态2.1 - 3.1的耦合进行cpmcscf计算

force;samc,5101.1;                    ! 对态1.1 - 2.1计算NACME
```

```
force;samc,5102.1;           ! 对态1.1 - 3.1计算NACME
force;samc,5103.1;           ! 对态2.1 - 3.1计算NACME
```

另外参见测试任务lif\_nacme.test。

#### 40.1.7 非绝热耦合矩阵元 (NACM)

参见40.1.6节。

#### 40.1.8 SA-MCSCF的差分梯度 (DEMC)

参见40.1.6节。

#### 40.1.9 例子

```
! $Revision: 2006.3 $
***, Calculate Gradients for Water
alpha=104 degree           ! 设定结构参数
r=1 ang
geometry={0;                ! 定义z矩阵
          H1,o,r;
          H2,o,r,H1,alpha}
basis=vdz                   ! 基组
hf                           ! 进行scf
forces                       ! 计算SCF的梯度
mp2                           ! mp2计算
forces                       ! mp2梯度
multi                        ! casscf计算
forces                       ! casscf梯度
```

[http://www.molpro.net/info/current/examples/h2o\\_forces.com](http://www.molpro.net/info/current/examples/h2o_forces.com)

## 40.2 数值梯度

可以用有限差分方法计算梯度，使用  
FORCE,NUMERICAL,options

如果最后一步能量计算的解析梯度不可用，那么自动计算数值梯度。默认不需要进一步的输入，将会对最后一次能量计算梯度。在FORCE命令以及随后的指令（参见后面的部分）中，可以给出以下选项：

STARTCMD=*command command*和当前FORCE命令之间的输入用于指定对哪个能量计算梯度。对每个位移执行这一输入部分。

PROC= <i>procname</i>	指定在每个位移使用的流程。它必须定义一个完整的能量计算，并且不能包含梯度或Hessian计算。
VARIABLE= <i>varname</i>	计算变量 <i>varname</i> 的值的梯度。这意味着数值梯度。这个变量必须在相应的能量计算中设置。
COORD=ZMAT CART 3N	进行梯度计算的坐标。更多信息参见40.2.1节。
DISPLACE=ZMAT SYM UNIQUE CART	用于数值梯度计算的位移坐标。如果结构用依赖于变量的Z矩阵给出，默认为ZMAT，否则为SYM（对称位移坐标）。更多信息参见40.2.1节。
SYMMETRY=AUTO NOSYM	用于数值梯度中波函计算的对称性。该选项只和DISPLACE=UNIQUE CART有关。如果给定AUTO，那么对每个位移使用最大可能的对称性。这意味着能量独立于对称性。注意，在MRCI或CASPT2计算中，通常不是如此。这个选项也不能用于局域关联计算。
AUTO	（逻辑型）。同SYMMETRY=AUTO。
ZMAT	（逻辑型）。同COORD=ZMAT。
OPT3N	（逻辑型）。同COORD=3N。
RSTEP= <i>rstep</i>	在数值梯度计算中距离的步长（单位Bohr）。默认是0.01。
DSTEP= <i>dstep</i>	对称位移的步长（单位Bohr）。默认是0.01。
ASTEP= <i>astep</i>	在数值梯度计算中角度的步长（单位度）。默认是1。
CENTRAL	（逻辑型）。使用两点中心公式；对 <i>M</i> 个自由度，需要2 <i>M</i> 个能量计算。
FORWARD	（逻辑型）。使用向前梯度（仅需要 <i>M</i> + 1个能量计算，但不太准确）。
FOURPOINT	（逻辑型）。用4点公式计算准确的数值梯度；需要4 <i>M</i> 个能量计算。
NUMERICAL	（逻辑型）。强制使用数值梯度，即使是在有解析梯度的情况下。
VARSAV	（逻辑型）。把梯度保存到变量GRADX，GRADY，GRADZ。

例如，

hf

```
ccsd(t)
forces,numerical
```

程序将根据求解梯度的需要在多个结构自动重复HF和CCSD(T)。这等价于

```
hf
ccsd(t)
forces,numerical,startcmd=hf
```

或者使用流程

```
forces,numerical,proc=runccsdt
...
runccsdt={
hf
ccsd(t)}
```

#### 40.2.1 坐标的选择 (COORD)

数值梯度默认对Z矩阵依赖的所有变量进行计算。如果Z矩阵不依赖于变量，或者依赖于 $3N$ 个变量，那么梯度对所有的 $3N$ 个坐标作计算，并且使用对称位移坐标求解梯度。这会产生最少的计算耗时。

这些默认设置可以用COORD指令修改：

COORD, *coord\_type*, [*displacement\_type*]

其中 *coord\_type* 可以是以下之一：

ZMAT	对结构依赖的所有变量计算数值梯度（默认）。
3N或CART	对所有 $3N$ 个核坐标计算梯度。如果Z矩阵不依赖于变量，或者使用了xyz输入格式，那么这是默认的。如果使用了这个选项，并且原始结构用Z矩阵的形式给出，那么Z矩阵将会丢失。

*displacement\_type* 的指定是可选的，并且只影响 $3N$ 个坐标梯度的数值计算。还可以用

DISPLACE, *displacement\_type*

给出， *displacement\_type* 可以是以下之一：

SYM	使用对称位移。这会产生最少数量的位移，并且总是保持波函的对称性。这是默认的并且是唯一推荐的选项。
-----	--



**CART** 对所有的 $3N$ 个笛卡尔坐标产生位移。这通常是不推荐的，因为在分子具有对称性的情况下，它会产生比需要多得多的位移。此外，波函对称性也不能保持，计算必须用C1对称性执行。

**UNIQUE** 同CART，但是去除了对称性等价的位移。也是不推荐的。

### 40.2.2 变量的数值导数

计算变量值的数值导数可以用

**VARIABLE, *name***

默认计算当前能量的梯度。

### 40.2.3 数值梯度的步长

数值步长对距离或笛卡尔坐标默认是0.01 bohr，对角度默认是1度。修改这些默认值可以用

**RSTEP, *dr***

**ASTEP, *da***

其中 $dr$ 是距离（或笛卡尔坐标）的位移，单位bohr， $da$ 是角度的位移，单位是度。

RSTEP的值用于对称位移。修改各个变量的步长可以用

**VARSTEP, *varname=value,...***

其中 $value$ 对距离必须是原子单位，对角度必须是度。

### 40.2.4 活性和非活性坐标

数值梯度默认对z矩阵依赖的所有变量进行计算，如果没有定义变量或使用了XYZ输入形式，则对所有 $3N$ 个坐标进行计算。可以定义活性变量的子集，使用

**ACTIVE, *variables***

如果出现这个卡，那么所有未指定的变量都是非活性的。还可以用

**INACTIVE, *variables***

这种情况下，所有未给出的变量都是活性的。

## 40.3 把梯度保存到变量

如果给出了指令

**VARSAV**

那么把梯度保存到变量GRADX, GRADY, GRADZ中。GRADX( $n$ )是关于第 $n$ 个原子 $x$ 分量的指令。原子按照打印的顺序。这个顺序可以不同于z矩阵输入的顺序，因为中心重新作了排序，使得把同一类型的所有原子放在一起。



## 第四十一章

# 几何优化 (OPTG)

自动几何优化用OPTG命令调用。不再需要MOLPRO以前版本的OPT命令，目前已经不能使用。

OPTG[, *key1=value, key2=value, .....*]

OPTG命令可用于对所有类型的波函执行自动的几何优化。对于最小值搜索，仅用OPTG命令不加任何选项或指令通常已经足够，但是也可以用以下部分描述的各种选项。

可以选择各种优化方法，在41.2.1节有说明。MOLPRO可以做最小化（即，搜索平衡构型），过渡态优化（即，搜索势能面上的鞍点），以及反应路径跟踪。标准算法基于有理函数方法和结构DIIS方法。还可以用Sun和 Ruedenberg的二次最陡下降跟踪方法（参见J. Sun and K. Ruedenberg, *J. Chem. Phys.* **99**, 5257 (1993)）。这种方法对过渡态搜索通常会有优势。关于各种最小化算法的详细讨论，参见F. Eckert, P. Pulay and H.-J. Werner, *J. Comp. Chem* **18**, 1473 (1997)。反应路径跟踪的说明参见F. Eckert and H.-J. Werner, *Theor. Chem. Acc.* **100**, 21, (1998)。关于解析梯度方法的引用，请查阅参考文献部分。

如果解析梯度可用于优化能量，则使用之。否则从有限能量差分计算数值梯度。通常，使用最后一次计算的基态能量。但是可以用VARIABLE指令或选项进行例如Davidson 校正能量、激发态或平衡校正能量的优化。

### 41.1 选项

大多数参数可以在OPTG命令行中作为选项给出，如在本节所示。也可以使用指令，这在41.2节进行说明。

### 41.1.1 选择进行优化的波函和能量选项

默认优化最后计算的能量，并且自动执行与最后一次能量计算有关的所有命令。对于某些应用，例如优化平衡校正能量或Davidson校正能量，可以用以下选项调整默认属性。

**STARTCMD**=*command*

指定开始的命令。在每个几何优化步骤中，从 *command* 开始到当前OPTG的所有输入都被处理。这些输入不能包含数值梯度或Hessian计算。如果需要数值梯度，将用OPTG对最后的能量（或指定的变量）进行计算。假定这些命令在进入OPTG 程序之前已经执行了。

**PROC**=*procname*

指定在每个几何优化步骤中执行的流程。这必须定义一个完整的能量计算（轨道优化和关联能处理），并且不能包含数值梯度或Hessian计算（对于要优化的能量或变量自动计算数值梯度）。但是，流程可以包含解析梯度的计算，例如用于平衡校正优化，需要几个梯度计算的线性组合。

**VARIABLE**=*varname*

优化变量*varname*中的值。这意味着数值梯度。

### 41.1.2 优化方法的选项

**METHOD**=RF|AH|DIIS|QSD|QSTPATH|SRMIN|SRTRANS|STSTEEP

使用的优化方法。细节参见41.2.1节。

**ROOT**=1|2      最小化搜索（1，默认）还是过渡态搜索（2）。

**DIRECTION**=idir

确定反应路径跟踪计算中的步长和方向，参见41.2.16节。

**STEPMAX**=*value*

在一个优化步骤中的最大步长。关于说明的更多细节，参见41.2.12节。

**TRUST**=*value*      增广Hessian方法的可信比率（默认0.5）。

**AHMAX**=*value*      增广Hessian流程允许的最大步长。这与标度参数空间有关（默认0.5）。

**CUT**=*value*      正交归一化阈值，用于Hessian的共轭梯度升级（默认1.d-3）。

ROTATE (逻辑型) 如果是.true., 则转换笛卡尔坐标, 使转动最小 (默认=.true.)。

### 41.1.3 修改收敛判据的选项

标准的MOLPRO收敛判据要求梯度的最大分量小于 $3 \cdot 10^{-4}$  [a.u.]且能量的最大变化小于 $1 \cdot 10^{-6}$  [H], 或梯度的最大分量小于 $3 \cdot 10^{-4}$  [a.u.]且步长的最大分量小于 $3 \cdot 10^{-4}$  [a.u.]。

还可以使用GAUSSIAN程序包的收敛判据。它比MOLPRO的判据弱一些, 需要梯度的最大分量小于 $4.5 \cdot 10^{-4}$  [a.u.], 梯度的均方根 (RMS) 小于 $3 \cdot 10^{-4}$  [a.u.], 并且优化步长的最大分量小于0.0018 [a.u.], 优化步长的RMS小于0.0012 [a.u.]。

MAXIT=*maxit* 优化循环的最大数量。默认是50。

GRADIENT=*thrgrad*

优化梯度所需的精度。默认是 $3 \cdot 10^{-4}$ 。

ENERGY=*threnerg*

优化能量所需的精度。默认是 $1 \cdot 10^{-6}$ 。

STEP=*thrstep* 几何优化步长的收敛阈值。默认是 $3 \cdot 10^{-4}$ 。

BAKER (逻辑型)。使用Baker的收敛判据 (参见J. Baker, *J. Comp. Chem.* **14**, 1085 (1993))。

GAUSSIAN (逻辑型)。使用Gaussian收敛判据。

SRMS=*thrsrms* (对Gaussian收敛判据) 设置在优化步骤中RMS所需的精度。默认是0.0012。

GRMS=*thrgrms* (对Gaussian收敛判据) 设置梯度RMS所需的精度。默认是 $3 \cdot 10^{-4}$ 。

FREEZE=*thrfreez*

在局域计算中, 如果步长小于*thrfreez*, 则冻结DFT格点和域 (默认0.01)。

注意, 默认的收敛参数也可以用全局指令GTHRESH修改, 即,  
GTHRESH, OPTSTEP=*step*, OPTGRAD=*grad*, ENERGY=*energy*;

#### 41.1.4 指定优化空间的选项

如果结构用Z矩阵给出，那么默认为优化Z矩阵依赖的变量。对于xyz输入，总是优化3N个坐标，即使xyz输入只依赖很少几个变量也是如此。如果使用了笛卡尔Z矩阵输入，当在结构区使用了ORIENT, MASS, 或CHARGE选项要求自动定向时，那么只能强制全空间的优化。细节参见41.2.2节的`opt_space`。

SPACE=ZMAT|3N

指定用于优化的坐标。如果结构给定为Z矩阵，那么只能用Z矩阵优化。

OPT3N|3N (逻辑型)。同SPACE=3N。

ZMAT (逻辑型)。同SPACE=ZMAT。

#### 41.1.5 指定优化坐标的选项

这些选项指定优化进行的坐标。默认是使用局域简正坐标。细节参见41.2.2 节的`opt_coord`。

COORD=NORMAL|NONORMAL|BMAT

NORMAL (逻辑型)。同COORD=NORMAL。

NONORMAL (逻辑型)。同COORD=NONORMAL。

BMAT (逻辑型)。同COORD=BMAT。

#### 41.1.6 数值梯度的选项

可以对Z矩阵依赖的变量或笛卡尔坐标有关的变量计算数值梯度。对于后者，最有效的办法是使用对称位移坐标。它们不会改变分子的对称性，并且位移数最少。另一种方法是用对称唯一的笛卡尔位移或全部3N个笛卡尔位移计算梯度（主要用于测试目的）。在这些情况下，分子的对称性可能会被位移降低，通常不推荐使用这样的位移。

DISPLACE=ZMAT|SYMM|UNIQUE|CART

位移坐标用于数值梯度。如果用依赖于变量的Z矩阵给出结构，那么默认为ZMAT，否则用SYMM（对称位移坐标）。不推荐使用UNIQUE或CART。

SYMMETRY=AUTO|NOSYM

对称性用于数值梯度的波函计算。该选项只与DISPLACE=UNIQUE|CART有关。如果给定AUTO，那么在每个位移使用可能的最高对称性。这意味着能量与所用的对称性无关。注意对于MRCI或CASPT2计算通常不是如此。该选项也不能用于局域关联计算。

- AUTO (逻辑型)。同SYMMETRY=AUTO。
- NOSYM (逻辑型)。同SYMMETRY=NOSYM。
- RSTEP=*rstep* 在数值梯度计算中, 用于位移的步长 (单位Bohr)。默认是0.01。
- DSTEP=*dstep* 用于对称位移的步长 (单位Bohr)。默认是0.01。
- ASTEP=*astep* 在数值梯度计算中, 用于角度的步长 (单位度)。默认是1。
- FOURPOINT (逻辑型)。使用四点公式用于准确的数值梯度。
- NUMERICAL (逻辑型)。强制使用数值梯度, 即使是在解析梯度可用的情况下。

#### 41.1.7 计算Hessian的选项

默认使用近似Hessian (模型Hessian)。Hessian也可以在优化中计算, 或从前一个Hessian或频率计算读取。

NUMHESS=*hstep*

如果给出, 那么每*hstep*次迭代计算一次数值Hessian。如果*hstep*=0或没有给出, 则只计算一次初始Hessian。

HESSREC=*record*

从给定的记录读取初始Hessian。如果*record*没有给出或是零, 则使用最后计算的Hessian。

READHESS (逻辑型)。同HESSREC=0。

HESSPROC=*procname*

指定用于Hessian计算的流程。这个流程必须定义完整的能量计算 (轨道优化和关联能处理)。可以使用与优化能量不同的方法。例如, MP2 Hessian可用于CCSD(T)优化, 或者CASPT2 Hessian可用于MRCI优化。 Hessian使用的流程默认与优化能量使用的相同。

HESSVAR=*varname*

对变量*varname*计算Hessian。这意味着从能量数值计算Hessian。默认使用与能量和梯度计算相同的变量。

HESSCENT 对Hessian计算使用中心梯度差分 (只有梯度可用时才有效)。

HESSFORW 对Hessian计算使用向前梯度差分 (只有梯度可用时才有效)。这会在略微位移的结构上更有效地计算Hessian, 但是仅需要一半数量的位移。这是默认的。

UPDATE=BFGS|IBFGS|CGRD|PMS|POWELL|MS|NONE

使用的Hessian升级方法。细节参见41.2.9节。

MAXUPD=*maxupd*

最大数量的Hessian升级。在每次计算Hessian时，计数器重设为零。

NUMDIAG      如果为true，则用对角数值Hessian（如果有的话）替换模型Hessian的对角元。有时这能改善收敛，但是由于它会导致对称破缺，因此默认为no。

#### 41.1.8 各种选项

VARSAVE      把笛卡尔梯度保存到变量GRADX, GRADY, GRADZ中。

NONUC        不计算网格点的梯度。

DEBUG        设定调试打印选项。

PRINT=*iprint* 用于优化的打印选项。

SAVEXYZ=*file* 把优化的坐标保存到xyz文件。每一步都写一个文件。文件名为*file\_nn.xyz*，其中的*nn*是迭代编号。对于最终的结构，忽略*nn*。如果没有给出文件名，那么*file* 采用输入文件的基本名称，即，*test.inp*在第一次迭代将会创建*test\_1.xyz*，对收敛的结构是*test.xyz*。每一步的xyz信息默认写到日志文件。

SAVEACT=*file* 在每一步保存优化变量。文件名为*file.act*。如果没有给出*file*，将采用输入文件的基本名称。这个文件可以在以后用READVAR命令读取，或者复制到新的输入中。

SAVEGRD=*file* 在每一步写笛卡尔坐标和梯度。文件名为*file.grd*。如果没有给出*file*，将采用输入文件的基本名称，并加上.grd后缀。

APPEND        （逻辑型）。如果给出，会对存在的 SAVEACT 和 / 或 SAVEGRD文件进行追加。

## 41.2 OPTG的指令

指定选项的另一个方法是使用指令，在本节有说明。和OPTG命令行中的选项相比，有些情况下这可以指定更多的细节。特别是ACTIVE或INACTICE指令，可用于更详细地定义优化空间。



### 41.2.1 选择优化方法 (METHOD)

METHOD,*key*;

*key*定义优化的方法。

对于能量最小化，以下的选项对*key*有效：

RF	有理函数方法（默认）。
AH	增广Hessian方法。这类似于RF算法，但是使用更复杂的步长限制算法。
DIIS	Pulay的几何DIIS方法。作为附加选项，你可以加上用于GDIIS内插的结构数量（默认是5）和改变内插类型（也就是进行GDIIS内插的子空间）。 METHOD,DIIS, <i>number</i> , <i>type</i> <i>type</i> 可以是使用梯度的GRAD内插（默认）——它对于刚性分子表现很好；可以是使用准-牛顿步长的STEP内插，在处理非常松弛的分子时非常有利；也可以是使用能量的ENER内插，它介于以上两者之间。
QSD	Sun和Ruedenberg的二次最陡下降方法。
SRMIN	旧版本的QSD。

对于过渡态搜索（用ROOT选项调用，参见41.2.11节），*key*可以是：

RF	有理函数方法（默认）。
DIIS	Pulay的几何DIIS方法（见上）。
QSD	使用映像Hessian方法的二次最陡下降过渡态搜索（参见J. Sun and K. Ruedenberg, <i>J. Chem. Phys.</i> <b>101</b> , 2157 (1994)）。对于过渡态搜索推荐使用这个选项——特别是对于复杂的情况。当接近PES出问题的区域时，会检查优化的步长，并重新计算Hessian。因此， <b>相比于RF或DIIS方法，这个方法在达到收敛时要安全一些（而且经常更快）</b> 。重新计算Hessian的安全设置可以用输入卡METHOD,QSD,NOHESS关闭。
SRTRANS	旧版本的QSD。

对于反应路径跟踪，输入的*key*可以是：

QSDPATH	二次最陡下降反应路径跟踪。这个方法通过用二次近似跟踪势能曲面的精确最陡下降路线，确定反应路径（内反应坐标，IRC）。在第一
---------	---

个优化的步长计算数值Hessian矩阵，随后用Powell或BFGS方法升级。如果超过了给定的最陡下降路线弧长，则重新数值计算Hessian（参见41.2.16节的OPTION）。细节参见J. Sun and K. Ruedenberg, *J. Chem. Phys.* **99**, 5269 (1993)。还可以在每 $m$ 步之后用NUMHES, $m$ 命令重新计算Hessian（参见41.2.7节）。如果在每个优化步骤中重新计算Hessian矩阵（NUMHES,1），则使用与升级Hessian不同的非常精确的算法。通过使用PRINT,OPT卡，这个算法会在每个优化步骤打印一个反应路径点 $r$ ，它不同于计算能量和梯度的点，而是更接近于真实的反应路径（关于算法的更多细节，参见J. Sun and K. Ruedenberg, *J. Chem. Phys.* **99**, 5257 (1993)）。关于QSD反应路径跟踪的更多输入选项，参见41.2.16节的OPTION部分。

SRSTEEP 旧版本的QSDPATH。

#### 41.2.2 优化坐标 (COORD)

优化可以使用各种坐标类型和算法。这可以用附加的子命令控制，将在这一节和下一节说明。

COORD,[*opt\_space*],[*opt\_coord*],[NOROT]

这个选项选择进行几何优化的优化空间和坐标系。

*opt\_space*定义要优化的参数。如果结构输入用Z矩阵格式给出，那么Z矩阵依赖的所有变量默认全部优化。Z矩阵依赖的变量子集可以用ACTIVE或INACTIVE子指令选择。如果Z矩阵不依赖于变量，或者使用了xyz输入，则优化所有的 $3N$ 个笛卡尔坐标。

*opt\_space*可以是以下之一：

ZMAT 优化Z矩阵所依赖的所有变量（如果结构用Z矩阵给出，是默认的）。

3N 优化所有的 $3N$ 个笛卡尔坐标（如果Z矩阵不依赖于变量，或者使用了xyz输入，是默认的）。如果使用了3N，Z矩阵输入坐标会被破坏。

*opt\_coord*确定进行优化的坐标。默认使用局域简正坐标。也可以使用笛卡尔坐标或自然内坐标。

*opt\_coord*可以是以下之一：

NORMAL 在局域简正坐标中优化。如果用模型Hessian对Hessian做近似，这是默认的。

NONORM 不使用局域简正坐标。

BMAT[=*filename*]

使用Pulay的自然内坐标, 参见G. Fogarasi, X. Zhou, P. W. Taylor and P. Pulay *J. Am. Chem. Soc.* **114**, 8191 (1992); P. Pulay, G. Fogarasi, F. Pang, J. E. Boggs *J. Am. Chem. Soc.* **101**, 2550 (1979)。创建的坐标以及关于优化的附加信息也可以写到指定的文件中。这些坐标部分地类似于振动光谱学家使用的价坐标, 具有降低不同模式之间耦合的优点。这经常会加快收敛速度。强烈推荐使用这个选项, 特别是在带环的大型有机分子的能量最小化计算中。但是仍然要记住, 这些坐标是自动构造的, 可能存在着不能正确处理的不稳定键结构 (例如, 过渡态优化中的弱键分子)。在这种情况下, 如果BMAT优化的收敛很慢, 或者导致了对称性破坏的错误, 则应当尝试其它的优化方法和/或笛卡尔或Z矩阵坐标。

如果给出了[NOROT]选项, 那么不会对笛卡尔坐标做转动最小的变换。

#### 41.2.3 位移坐标 (DISPLACE)

DISPLACE,*displacement\_type*

细节参见40.2.1节。

#### 41.2.4 定义活性结构参数 (ACTIVE)

ACTIVE,*param*;

声明变量名称*param*在优化中激活。结构依赖的所有变量在一开始默认都是激活的; 包含ACTIVE卡使得所有的参数都是未激活的, 除非直接声明了是激活的 (参见INACTIVE)。

#### 41.2.5 定义非活性结构参数 (INACTIVE)

INACTIVE,*param*;

声明变量名称*param*在优化中不激活。如果在输入中出现了ACTIVE卡, 则忽略INACTIVE卡 (参见ACTIVE)。

#### 41.2.6 Hessian近似 (HESSIAN)

MOLPRO几何优化默认对Hessian使用力场近似 (“模型Hessian”, 参见R. Lindh, A. Bernhardsson, G. Karlström and P. Malmqvist *Chem. Phys. Lett.* **241**, 423 (1995)), 它会显著加速收敛。模型Hessian对直到第三行的元素实现了参数化。也可以使用

用Schlegel的模型Hessian，或者数值计算的Hessian（参见41.2.7节）。

HESSIAN,*options*

其中*options*可以是

MODEL            在优化中使用Lindh的模型Hessian（默认）。

MODEL=Schlegel

使用Schlegel模型Hessian。

MODEL=VDW       为Lindh的模型Hessian添加vdW项。

Schlegel        同MODEL=Schlegel。

VDW            同MODEL=VDW。

NOMODEL        不对Hessian使用模型Hessian近似。

NUMERICAL=*hstep*

在*hstep*次迭代后重新计算Hessian。这禁止使用模型Hessian。如果*hstep*=0，那么只在第一次迭代计算Hessian。默认的参数用于计算数值Hessian，除非用选项修改，参见41.2.7节对NUMHESS指令的说明。任何对NUMHESS指令有效的选项也可以接在HESSIAN指令中的NUMERICAL选项后。

READ | RECORD | HESSREC=*record*

从给定的记录*record*读取Hessian。如果*record*没有给出或为零，将读取最后计算的Hessian。关于数值Hessian的更多细节，参见41.2.7节。

UPDATE=*type*    用于Hessian升级的方法。细节参见41.2.9节。

MAXUPD=*maxupd*

最大数量的Hessian升级。在每次计算Hessian时，计数器重设为零。

如果禁止模型Hessian（NOMODEL），并且没有读取或计算Hessian，则初始Hessian假定是对角的，对所有的长度用值1 hartree\*bohr\*\*(-2)，对所有的角度用值1 hartree\*radian\*\*(-2)。Hessian的附加矩阵元可以用HESSELEM指令定义，参见41.2.8节。

在过渡态搜索中，Hessian矩阵默认在第一次迭代进行数值求解。如果指定了READ，也可以使用以前计算的Hessian。

### 41.2.7 数值Hessian (NUMHES)

NUMHES, *options*

或

NUMHES, *hstep, options*

如果出现这个指令, 用有限差分计算数值Hessian。如果解析梯度可用, 则可以使用向前梯度差分 (对每个坐标需要一次梯度计算) 或中心差分 (更精确, 对每个坐标需要两个次梯度计算)。对于过渡态优化, 使用向前差分通常已经足够。如果优化方法不支持解析梯度, 则能量求两次差分。在这种情况下只能使用中心差分。

可以给出以下选项:

HSTEP=*hstep* *hstep*=-1: 不计算数值Hessian (对能量最小化是默认的);

*hstep*=0: 仅在开始优化时计算一次数值Hessian (对过渡态搜索是默认的);

*hstep*=*n*: 在每*n*步优化后计算数值Hessian。这对有困难的过渡态优化会有帮助 (例如, 如果Hessian的本征值结构在优化时发生了变化)。

FORWARD 使用向前差分 (默认)。

CENTRAL 使用更准确的中心差分。

RSTEP=*rstep* 距离的步长 (单位Bohr)。默认是0.01。

ASTEP=*astep* 角度的步长 (单位度)。对小于和大于90度的角, 默认分别为0.5和1度。

DSTEP=*dstep* 对称位移的步长 (单位Bohr)。默认是0.01。

VARIABLE=*varname*

用给定的变量进行数值Hessian的计算。注意, 这不能使用梯度, 因此Hessian 的求解会非常昂贵。

PROCEDURE=*procname*

用于计算Hessian的流程。这个流程必须定义完整的能量计算 (轨道优化和关联能处理)。可以使用与优化能量不同的方法。例如, MP2 Hessian可用于CCSD(T)优化, 或者CASPT2 Hessian用于MRCI优化。Hessian使用的流程默认与优化能量使用的相同。

DISPLACE=*type*

*type*可以是以下之一:

SYMM	使用对称位移坐标（默认）。这是唯一推荐选项。
CART	使用 $3N$ 个笛卡尔位移（不推荐）。这比需要的能量计算多，并且不能保持分子的对称性。
UNIQUE	使用对称唯一笛卡尔位移（不推荐）。

注意，梯度和Hessian的位移类型必须相同。

**CALC=icalc**    *icalc*=0: 在每*hstep*个优化步骤之后，重新数值计算整个Hessian矩阵（默认）。  
*icalc*=1: 如果Hessian矩阵元在升级前后的变化大于*thresh*（参见41.2.9节的UPDATE），则重新数值计算所选的Hessian矩阵元。如果未指定*thresh*，则使用默认值*thresh* = 0.05（即，最大偏离是5%）。  
*icalc*=2: 如果Hessian矩阵在升级前后的RMS偏差大于*thresh*，则重新数值计算整个Hessian矩阵。如果未指定*thresh*，则使用默认值*thresh* = 0.5 a.u.。

**THRESH=thresh**

部分升级或动态升级Hessian的阈值，见上。

#### 41.2.8 Hessian元素 (HESSELEM)

**HESSELEM,value, active1,active2,...**

设定Hessian矩阵元*active1*和*active2*之间活性变量的初始值为*value*。如果忽略*active2*，则默认为*active1*（对角元）。可以根据需要给出多个HESSELEM指令。

#### 41.2.9 Hessian升级 (UPDATE)

**UPDATE,[TYPE=]type,MAX=maxupd**

这一指令用于选择升级类型，并限制用于Hessian升级的点数最大到*maxupd*。用于Hessian 升级流程的默认步数是5。如果在优化坐标中有对称束缚，那么默认数可能小于5。

在最小化中，*type*可以是：

BFGS	Hessian使用BFGS升级（默认）。
IBFGS	转置Hessian使用BFGS升级。
CGRD	使用共轭梯度升级（此外参见CUT,TRUST）。
NONE	不使用任何升级。

在过渡态优化中, *type*可以是:

PMS	Hessian的Powell/Murtagg-Sargent组合升级 (默认)。
POWELL	使用Powell的Hessian升级。
MS	使用Murtagg和Sargent的升级流程。
NONE	不使用任何升级。

#### 41.2.10 数值梯度 (NUMERICAL)

NUMERICAL, *options*, *active*<sub>1</sub>=*step*<sub>1</sub>, *active*<sub>2</sub>=*step*<sub>2</sub> ...;

使用这个指令, 用有限差分计算梯度。 *step*<sub>*i*</sub>是活性结构参数*active*<sub>*i*</sub>的增量。对于没有指定的活性结构参数, 使用默认值。键长默认的增量是0.01 Bohr, 小于或大于90度的键角默认增量分别是0.5或1度。这些默认值可以通过指定RSTEP或ASTEP修改。DSTEP是对称位移的长度, 用于对3N个坐标优化的情况。

对于每个活性变量, 每个优化步骤需要两个能量计算——因此数值优化会很昂贵! 对3N个坐标的优化, 对称位移坐标通常用于将能量计算的数量降到最低 (参见40.2.1节)。

对于特定能量的优化, 参见41.2.17节的VARIABLE。

可以给定以下选项:

RSTEP=*rstep* 距离的步长 (单位Bohr)。默认是0.01。

ASTEP=*astep* 角度的步长 (单位度)。对小于和大于90度的角, 默认分别为0.5和1度。

DSTEP=*dstep* 对称位移的步长 (单位Bohr)。默认是0.01。

CENTRAL 对梯度使用中心差分 (默认)。

FORWARD 使用向前差分 (对梯度计算不推荐)。

FOURPOINT 使用四点公式, 得到非常准确的数值梯度。

PROCEDURE=*procname*

用给定的流程计算数值梯度。这个流程必须定义完整的能量计算 (轨道优化和关联能处理)。

VARIABLE=*varname*

给出用于数值梯度计算的变量。

DISPLACE=*type*

位移类型。注意，梯度和Hessian的位移类型必须相同。*type*可以是以下之一：

SYMM	使用对称位移坐标（默认）。这是唯一推荐选项。
CART	使用 $3N$ 个笛卡尔位移（不推荐）。这比需要的能量计算多，并且不能保持分子的对称性。
UNIQUE	使用对称唯一笛卡尔位移（不推荐）。

#### 41.2.11 过渡态（鞍点）优化（ROOT）

ROOT,*root*

指定要跟踪的Hessian的本征矢。

*root*=1 指定最小化（默认）。

*root*=2 指定过渡态（鞍点）优化。

在目前的执行中，可以用有理函数方法（METHOD,RF），几何DIIS方法（METHOD,DIIS），以及Sun和Ruedenberg的二次最陡下降方法（METHOD,SRTRANS），进行鞍点搜索。

注意，获得收敛通常比最小化更加困难。因此特别需要较好的初始结构和对Hessian较好的近似。后者通过数值求解Hessian完成（参见41.2.7节），或者使用以前计算的Hessian（参见41.2.6节）。

#### 41.2.12 设定最大的步长（STEP）

STEP,*steplength*,*drmax*,*damax*,*drmax1*,*damax1*

*steplength*是换算参数空间的初始步长（默认0.3）。在AH方法中，这是动态调节的，可以有最大值*ahmax*（参见TRUST）。

*drmax*是距离的初始最大变化（单位Bohr，默认0.3）。在AH方法中，这是动态调节的，可以一直到最大值*drmax1*（默认0.5 Bohr）。

*damax*是角度的初始最大变化（单位度，默认是2）。在AH方法中，这是动态调节的，可以一直到最大值*damax1*（默认10度）。

#### 41.2.13 重新定义可信比率（TRUST）

TRUST,*ratio*,*ahmax*

*ratio*确定围绕当前最小点的半径，其内的点用于使用共轭梯度方法升级Hessian（默认0.5；另外参见UPDATE）。



*ahmax*是在增广Hessian流程中允许的最大步长。它与换算的参数空间有关（默认0.5）。初始的步长是*stepmx*（参见STEP卡）。

#### 41.2.14 设定截断参数 (CUT)

CUT,*threshold*

指定用于共轭梯度升级Hessian的正交归一化阈值（默认1.d-3；另外参见UPDATE）。

#### 41.2.15 线搜索 (LINESEARCH)

LINESEARCH,*iflag*,*thrlmin*,*thrlmax*

在当前结构和前一个结构之间，用四次多项式对稳定点（最低点或鞍点）结构进行内插。如果*iflag*=0或没有设定*iflag*，那么下一个优化步骤将来自使用对能量或梯度内插得到的内插结构。如果*iflag*=1，那么在新的优化步骤之前，能量和梯度将对内插结构重新计算。但是注意，重新计算能量和梯度对提高优化的收敛速度通常不会有什么影响。对于*iflag*=1，重新计算能量和梯度的最大和最小阈值分别是*thrlmin*和*thrlmax*。也就是说，重新计算仅发生在内插结构既不太接近实际结构*thrlmin*，又离实际结构*thrlmax*不太远的情况。在优化的换算参数空间中，默认值为*thrlmin*=0.001和*thrlmax*=0.05。

#### 41.2.16 反应路径跟踪选项 (OPTION)

OPTION,*key*=*param*;

其中*key*可以是

- |        |   |
|--------|---|
| IDIR   | 如果从过渡态（或者接近过渡态）开始，确定从哪里开始第一步。如果选择IDIR=0，第一步将朝着过渡态开始。这是默认的。如果在输入中给出IDIR=1，那么第一步优化将沿着“过渡矢量”，也就是本征值最小的Hessian本征矢，它总是指向最小值。如果使用更大的IDIR参数，第一步将会更大；如果使用负值，第一步将会朝着反方向。 |
| STPTOL | 如果使用了升级的Hessian矩阵，那么这个参数确定使用何种的升级。沿着求解的最陡下降路线上，如果后面两点间的步长小于 <i>stptol</i> （也就是如果接近了最小值），则使用BFGS 升级，否则用Powell升级。 <i>stptol</i> 的默认值是1.d - 6。                     |
| SLMAX  | 这个选项仅对旧版本的反应路径跟踪算法（即，METHOD,SRSTEEP）有效。在这个算法中， <i>slmax</i> 确定数值Hessian重新计算的频率。如果最后一步的总步长超过了 <i>slmax</i> ，将重新计算Hessian，否则将会升级。                                 |

*slmax*默认是优化步长 $steplength$ （参见41.2.12节的STEP）的最大步长的两倍。如果使用了METHOD,QSD, 则SLMAX选项无效, 应替换成NUMHES命令（见上）。

#### 41.2.17 优化能量变量 (VARIABLE)

VARIABLE,*name*;

定义变量 $name$ , 它保存用于有限差分方法优化的能量值。默认为最近一次程序设定的ENERGY(1)。可以使用的其它变量有

ENERGY( <i>i</i> )	保存态 <i>i</i> 最后的能量。
ENERGR( <i>i</i> )	保存态 <i>i</i> 最后的参考能量。
ENERGD( <i>i</i> )	保存态 <i>i</i> 最后的Davidson校正能量。
ENERGP( <i>i</i> )	保存态 <i>i</i> 最后的Pople校正能量。
ENERGC	保存CCSD(T) [QCI(T), BCCD(T)]计算中的CCSD (QCI, BCCD) 能量（优化单个态）。
ENERGT(1)	保存CCSD(T)计算中的CCSD(T)能量（单个态）。
ENERGT(2)	保存CCSD(T)计算中的CCSD[T]能量（单个态）。
ENERGT(3)	保存CCSD(T)计算中的CCSD-T能量（单个态）。

这些变量由CI和/或CCSD程序自动设置。用户应当使用正确的变量名；如果指定的变量名未被最后一个程序或由用户定义, 则会出错退出。

注意：使用VARIABLE选项触发NUMERICAL, 因此优化效率会非常低！

#### 41.2.18 打印选项 (PRINT)

PRINT,*code=level*,...;

打开打印选项。通常情况下, *level*可以忽略, 或者是0; *level* > 0的值仅产生对调试有用的输出。*code*可以是

HESSIAN	打印升级的Hessian矩阵。注意, 总是打印它的对角元。
HISTORY	打印前面计算的整套结构, 梯度和能量。
GRADIENT	打印扩展的梯度信息。
OPT	打印优化过程的详细信息（主要用于调试）。

一个PRINT命令可以指定几个打印选项。

### 41.2.19 圆锥交点优化 (CONICAL)

要优化两个同自旋电子态之间的圆锥交点，必须在SA-CPMCSCF级别求解三个矢量：

- 1) 非绝热导数耦合 (DC)。
- 2) 下态的梯度 (LSG)。
- 3) 上态的梯度 (USG)。

这可以在MULTI输入中通过添加三个不同的CPMCSCF指令完成：

CPMCSCF, NACM,  $S_i$ ,  $S_j$ , ACCU=1.0d-7, record=record1.file

CPMCSCF, GRAD,  $S_i$ , SPIN=Spin of state  $S_i$ , ACCU=1.0d-7, record=record2.file

CPMCSCF, GRAD,  $S_j$ , SPIN=Spin of state  $S_j$ , ACCU=1.0d-7, record=record3.file

其中 $S_i, S_j$ 是通常格式istate.istsym的电子态，record[n].file指定存储CPMCSCF解的文件名和文件编号。参数SPIN是在WF卡中的值的一半，用于定义电子态。

需要记住：

- i) 在CPMCSCF指令中总是指定三个不同的record.file。
- ii) 总是在最后对USG求解CPMCSCF。
- iii) 如果圆锥交点包含的态有不同的自旋（例如单/三重态交叉），就会跳过DC求解，因为耦合为零。

MULTI输入之后接三组FORCE命令（只有两个用于单/三重交点）。它们的形式为：

FORCE

SAMC, record[n].file

CONICAL, record4.file[,NODC]

其中record.file是包含CPMCSCF信息的记录之一，record4.file指向一个自由记录，用于CONICAL代码的内部存储。在所有的CONICAL指令中，record4.file必须相同。此外，在目前的程序中，只有当CONICAL指令的file=1时才能很好地执行。对于不同自旋的情况（例如S/T交叉），当不需要DC时，必须使用可选的关键词NODC。

实际的优化通过用OPTG, STARTCMD=MULTI来执行。下面的例子对LiH<sub>2</sub>优化圆锥交点（基态和激发态都是二重态）。

```
!examples/lih2_DOD1.com $Revision: 2009.1 $
***, LiH2
```

```

basis=sto-3g
print,orbitals,civector

symmetry,x          ! 只使用分子平面。两个态必须有相同对称性。
geometry={
    Li;
    h1,Li,r;
    h2,Li,r,h1,theta}

r=3.0
theta=35

{hf;wf,4,1,0}

{multi;occ,6,1;wf,5,1,1;state,2          ! 态平均casscf
CPMCSCF,NACM,1.1,2.1,accu=1.0d-7,record=5100.1 ! 非绝热耦合cpmcscf
CPMCSCF,GRAD,1.1,spin=0.5,accu=1.0d-7,record=5101.1 ! 态1梯度
CPMCSCF,GRAD,2.1,spin=0.5,accu=1.0d-7,record=5102.1} ! 态2梯度

{Force
SAMC,5100.1          ! 计算耦合矩阵元
CONICAL,6100.1}      ! 对圆锥交点的优化保存信息

{Force
SAMC,5101.1          ! 对态1计算梯度
CONICAL,6100.1}      ! 对圆锥交点的优化保存信息

{Force
SAMC,5102.1          ! 对态2计算梯度
CONICAL,6100.1}      ! 对圆锥交点的优化保存信息

optg,startcmd=multi  ! 寻找圆锥交点

```

[http://www.molpro.net/info/current/examples/lih2\\_D0D1.com](http://www.molpro.net/info/current/examples/lih2_D0D1.com)

第二个例子优化 $LiH_2(+)$ 的单—三态交点（基态是单重态，激发态是三重态）。

```

!examples/lih2+_S0T0.com $Revision: 2009.1 $
***, LiH2

basis=sto-3g

symmetry,nosym
geometry={
    Li;

```

```

      H1,Li,r;
      H2,Li,r,H1,theta}

r=3.7
theta=160

{hf;wf,4,1,0}

{multi;
  occ,7;
  wf,4,1,0;           ! 单态
  wf,4,1,2;           ! 三态
  CPMSCF,GRAD,1.1,spin=0,accu=1.0d-7,record=5101.1 ! 单态梯度的cpmcscf
  CPMSCF,GRAD,1.1,spin=1,accu=1.0d-7,record=5100.1 ! 三态梯度的cpmcscf
}

{Force
  SAMC,5101.1          ! 单态的态平均梯度
  CONICAL,6100.1,NODC} ! 保存信息用于OPTCONICAL

{Force
  SAMC,5100.1          ! 三态的态平均梯度
  CONICAL,6100.1,NODC} ! 保存信息用于OPTCONICAL

optg,startcmd=multi,gradient=1.d-6 ! 寻找单三态交点

```

[http://www.molpro.net/info/current/examples/lih2+\\_S0T0.com](http://www.molpro.net/info/current/examples/lih2+_S0T0.com)

### 41.3 使用SLAPAF程序用于几何优化

几何优化还可以使用Roland Lindh编写的SLAPAF程序。这可以通过为优化方法加上“SL”前缀来调用。支持以下方法：

SLRF	使用有理函数近似；
SLNR	使用Newton-Raphson方法；
SLC1	使用C1-DIIS方法；
SLC2	使用C2-DIIS方法；

当使用DIIS方法（SLC1或SLC2）时，DIIS参数的指定方法与在标准MOLPRO优化程序中的相同。

使用SLAPAF程序时，有以下一些差别：

- 1) 优化中不能使用Z矩阵坐标。
- 2) 相反，可以直接定义变化或固定的内坐标。
- 3) 在收敛的结构上可以用灵活的方法加入束缚。

#### 41.3.1 定义束缚

束缚和内坐标（见后）可以是键长、键角等的线性组合。后者在这里称作原始内坐标，可以在束缚的定义前或束缚的定义中指定。原始坐标的一般定义为：

PRIMITIVE,[NAME=] *symbolic name*, *explicit definition*;

或

PRIM,[NAME=] *symbolic name*, *explicit definition*;

这里的*symbolic name*是原始坐标中给定的名称（如果忽略，则会自动产生）。这个名称在后面参照原始坐标时会用到。

*explicit definition*的形式为：

*type*, *atoms*

*type*可以是以下之一：

BOND	由两个原子定义的键长。
ANGLE	由三个原子定义的键角（角1-2-3）。
DIHEDRAL	由四个原子定义的二面角（原子1,2,3和2,3,4构成平面之间的夹角）。
OUTOFPLANE	由四个原子定义的面外角（原子2,3,4构成平面和键1-4之间的夹角）。
DISSOC	由两组原子定义的离解坐标。
CARTESIAN	一个原子的笛卡尔坐标。

除了DISSOC和CARTESIAN类型之外，所有类型的原子定义为：

ATOMS=[*a1*,*a2*,*a3*,...]

其中所需的原子数随上面指定的*type*变化，原子名*a1*,*a2*,*a3*,... 既可以是Z矩阵输入的原子标记名，也可以是对应Z矩阵行号的整数。注意，这里需要用方括号，不是表示可选输入。

对于DISSOC，定义如下：

DISSOC, GROUP1=[*a1*,*a2*,...], GROUP2=[*b1*,*b2*,...];

相应的内坐标是两组原子的质心距离。

对于CARTESIAN, 定义为:

CARTESIAN, *I*, *atom*;

其中*I*可以是X,Y,Z或1,2,3之一, *atom*可以是Z矩阵原子名或者对应Z矩阵行的整数。

通过这种定义, 束缚可以定义为

CONSTRAINT,[VALUE=]*value*,[*unit*],[[FACTOR=]*fac*,*prim*,[[FACTOR=]*fac*],*prim*,...;

其中, *value*是加以束缚的值; *prim*要么是施加束缚前的原始定义名称, 要么直接定义; *fac*是在束缚中对应原始定义的因子。如果忽略*fac*, 则使用1。

如果*value*定义为Angstrom或Radian, 那么必须给定*unit*。

$C_s$ 对称性H<sub>2</sub>O的例子:

键角限定为100度:

```
constraint,100,deg,angle,atoms=[h1,o,h2];
```

它等价于

```
primitive,a1,angle,atoms=[h1,o,h2];
constraint,100,a1;
```

保持两个OH距离相等:

```
constraint,0,bond,atoms=[h1,o],-1.,bond,atoms=[h2,o];
```

它等价于

```
primitive,b1,bond,atoms=[h1,o];
primitive,b2,bond,atoms=[h2,o];
constraint,0,b1,-1.,b2;
```

### 41.3.2 定义内坐标

SLAPAF默认在力常数加权简正坐标中优化, 它们由程序自动确定。但是用户也可以定义自己的坐标。内坐标的定义和束缚类似, 都基于原始坐标。输入为:

INTERNAL,[[NAME=]*name*],[[FACTOR=]*fac*],*prim*,[[FACTOR=]*fac*],*prim*,...;

FIX, [[NAME=]*name*],[[FACTOR=]*fac*],*prim*,[[FACTOR=]*fac*],*prim*,...;

用INTERNAL指定的内坐标是可变的, 用FIX指定的内坐标固定在初始值。

内坐标的定义有一点非常重要, 内坐标总数必须等于分子自由度总数。否则会产生错误信息。只需要给出对称性无关的坐标。

### 41.3.3 SLAPAF的附加选项

有些选项可以传递到SLAPAF程序中。选项用SLOPT子指令指定:

```
{opt;method=slnr;{slopt;opt1;opt2,par1,par2;opt3;...}}
```

可用的选项有：

CART	使用近似Hessian的本征矢，用笛卡尔坐标表示，与内坐标定义相同。
NOMA	不对步长加以限制。
UORD	在升级Hessian之前，根据Schlegel的方法对梯度和位移矢量排序。默认不重新排序。
HWRs	使用力场加权内坐标（默认）。
RS-P	激活RS-P-RFO作为默认方法用于过渡态搜索；默认是RS-I-RFO。
NOHW	使用非加权的内坐标。
PRBM	打印B矩阵。
RTHR, <i>Thra, Thrb, Thrt</i>	选择冗余坐标的阈值，分别用于键、弯曲和扭转。默认为 0.2, 0.2, 0.2。
MODE, <i>index</i>	在过渡态计算中，用于模式跟踪的Hessian矢量索引。
FIND	在过渡态搜索中，对束缚的情况使用非束缚优化（参见MOLCAS手册）。
GMRM, <i>thr</i>	FIND的阈值，默认为0.2（参见MOLCAS手册）。
MEP-	执行最小能量路径（MEP）搜索。
NMEP, <i>npoints</i>	在MEP计算中，要寻找的MEP点数。

更多信息请参考MOLCAS使用手册。

## 41.4 例子

```
!examples/h2o_scfopt_631g.com $Revision: 2002.10 $
***,h2o                ! 标题
r=1.85,theta=104       ! 设定结构参数
geometry={0;           ! Z矩阵结构输入
    H1,0,r;
    H2,0,r,H1,theta}
basis=6-31g**          ! 使用Pople基组
hf                     ! 闭壳层scf
optg                   ! 进行scf结构优化
```



[http://www.molpro.net/info/current/examples/h2o\\_scfopt\\_631g.com](http://www.molpro.net/info/current/examples/h2o_scfopt_631g.com)

```
! $Revision: 2006.0 $
***,H2O                                ! 标题
include procedures
basis=vtz                              ! 使用cc-pVTZ基组
geometry={0;                            ! 水的Z矩阵
          H1,0,R;
          H2,0,R,H1,THETA}
R=0.96 Ang                             ! 初始键长
Theta=104                              ! 初始键角
optmp2                                 ! 进行MP2结构优化
show,R,Theta                           ! 显示优化的结构参数
```

[http://www.molpro.net/info/current/examples/h2o\\_optmp2.com](http://www.molpro.net/info/current/examples/h2o_optmp2.com)

```
!examples/h2o_optmp2.com $Revision: 2006.0 $
include procedures
geometry={0;                            ! 水的Z矩阵
          H1,0,R;
          H2,0,R,H1,THETA}
basis=vtz                              ! 使用VTZ基组
R=0.96 Ang                             ! 初始键长
Theta=104                              ! 初始键角

hf
mp2
optg    ! 优化MP2能量
mp4     ! 在MP2结构进行MP4单点计算
ccsd(t) ! 在MP2结构进行CCSF(T)单点计算
```

[http://www.molpro.net/info/current/examples/h2o\\_optmp2\\_runccsdt.com](http://www.molpro.net/info/current/examples/h2o_optmp2_runccsdt.com)

#### 41.4.1 使用Z矩阵的简单HF优化

```
!examples/allene_optscf.com $Revision: 2002.10 $
***, Allene geometry optimization using Z-Matrix
memory,1,m
basis=sto-3g

rcc=1.32 ang
rch=1.08 ang
acc=120 degree
Geometry={C1                                ! Z矩阵输入
          C2,c1,rcc
          Q1,c1,rcc,c2,45
```

```

      C3,c2,rcc,c1,180,q1,0
      h1,c1,rch,c2,acc,q1,0
      h2,c1,rch,c2,acc,h1,180
      h3,c3,rch,c2,acc,h1,90
      h4,c3,rch,c2,acc,h2,90}

hf
optg,saveact=allene.dat,savexyz=allene.xyz    ! 使用模型Hessian的默认优化
                                              ! 把优化变量保存到文件allene.dat
                                              ! 用xyz格式把优化的结构保存到文件allene.xyz

```

<http://www.molpro.net/info/current/examples/allene-optscf.com>

#### 41.4.2 使用自然内坐标的优化 (BMAT)

```

!examples/allene_opt_bmat.com $Revision: 2009.1 $
***, Allene geometry optimization using natural internal coordinates
memory,1,m
basis=sto-3g

rcc=1.32 ang
rch=1.08 ang
acc=120 degree
symmetry,nosym
Geometry={C1;                                ! Z矩阵输入
          C2,c1,rcc
          Q1,c1,rcc,c2,45
          C3,c2,rcc,c1,180,q1,0
          h1,c1,rch,c2,acc,q1,0
          h2,c1,rch,c2,acc,h1,180
          h3,c3,rch,c2,acc,h1,90
          h4,c3,rch,c2,acc,h2,90}

hf;
optg                ! 使用模型Hessian的默认优化
coord,bmat          ! 使用自然内坐标

optg,coord=bmat     ! 同上

```

[http://www.molpro.net/info/current/examples/allene-opt\\_bmat.com](http://www.molpro.net/info/current/examples/allene-opt_bmat.com)

#### 41.4.3 使用流程的MP2优化

```

!examples/allene_optmp2.com $Revision: 2002.10 $
***, Allene geometry optimization using Z-Matrix
memory,2,m
basis=vdz

```

```

rcc=1.32 ang
rch=1.08 ang
acc=120 degree
Geometry={C1                      ! Z矩阵输入
          C2,c1,rcc
          Q1,c1,rcc,c2,45
          C3,c2,rcc,c1,180,q1,0
          h1,c1,rch,c2,acc,q1,0
          h2,c1,rch,c2,acc,h1,180
          h3,c3,rch,c2,acc,h1,90
          h4,c3,rch,c2,acc,h2,90}

optg,procedure=runmp2      ! 使用流程optmp2

runmp2={hf;mp2}           ! 流程的定义

```

[http://www.molpro.net/info/current/examples/allene\\_optmp2.com](http://www.molpro.net/info/current/examples/allene_optmp2.com)

#### 41.4.4 使用结构DIIS的优化

```

!examples/cafeine_opt_diis.com $Revision: 2002.10 $
***, CAFFEINE cartesian coordinates (XYZ format)
memory,1,m
basis=sto-3g
geomtyp=xyz
geometry={
24
          CAFFEINE CARTESIAN COORDINATES
C          0.8423320060      -0.3654865620      0.0000000000
C          -0.2841017540     -1.1961236000      0.0000000000
N          2.0294818880     -1.1042264700      0.0000000000
N          0.0774743850     -2.5357317920      0.0000000000
N          -1.6472646000     -0.6177952290      0.0000000000
C          1.4531962870     -2.3678913120      0.0000000000
C          0.6373131870      1.1735112670      0.0000000000
C          -1.7812691930      0.7688916330      0.0000000000
N          -0.6771444680      1.6306355000      0.0000000000
O          1.6106752160      1.9349693060      0.0000000000
O          -2.9202890400      1.2510058880      0.0000000000
C          -0.9202462430      3.1094501020      0.0000000000
C          -2.8623938560     -1.4824503660      0.0000000000
C          3.4552156930     -0.6811094280      0.0000000000
H          2.0878150460     -3.2451913360      0.0000000000
H          -1.4989252090      3.4222116470     -0.8897886280

```

```

H      -1.4989252090      3.4222116470      0.8897886280
H      0.0071905670       3.7148499490      0.0000000000
H     -3.4903070930      -1.2888938190     -0.8907763360
H     -3.4903070930      -1.2888938190      0.8907763360
H     -2.6289534570      -2.5638654230      0.0000000000
H      4.1360211370      -1.5529079440      0.0000000000
H      3.6817059520      -0.0685850980      0.8931597470
H      3.6817059520      -0.0685850980     -0.8931597470
}

hf
optg,savexyz=caffeine.xyz  ! 把优化的结构保存到文件caffeine.xyz
coord,bmat                ! 在自然内坐标中优化
method,diis               ! 优化方法: 结构DIIS

optg,coord=bmat,method=diis,savexyz=caffeine.xyz  ! 同上

```

[http://www.molpro.net/info/current/examples/caffeine\\_opt\\_diis.com](http://www.molpro.net/info/current/examples/caffeine_opt_diis.com)

#### 41.4.5 HCN — HNC异构体的过渡态

第一个例子显示如何进行MP2过渡态优化。初始Hessian取自以前的HF频率计算。

```

!examples/hcn_mp2_ts.com $Revision: 2009.1 $
***, HCN <--> NHC Isomerization - Transition State Optimization and Frequencies

l1=1.18268242 ang
l2=1.40745082 ang
a1=55.05153416 degree

basis=3-21G

symmetry,nosym
geometry={
    C
    N,1,l1
    H,2,l2,1,a1}

hf                      ! HF-SCF

frequencies,analytical ! HF-SCF的振动频率 (解析Hessian)

mp2                     ! MP2

optg,root=2,method=rf,readhess ! 使用有理函数优化程序和HF Hessian的过渡态搜索

```

```
frequencies          ! MP2的振动频率 (数值Hessian)
---
```

[http://www.molpro.net/info/current/examples/hcn\\_mp2\\_ts.com](http://www.molpro.net/info/current/examples/hcn_mp2_ts.com)

第二个例子显示如何用MP2 Hessian进行CCSD(T)优化。注意, 当前CCSD(T)的梯度用有限差分方法进行数值计算, 对大分子这会需要很长时间。计算解析梯度的MP2 Hessian有限差分。

```
!examples/hcn_ccsd_ts.com $Revision: 2002.10 $
***, HCN <--> NHC Transition State Optimization and Frequencies

rcn=1.18 ang
rnh=1.40 ang
alpha=55 degree

basis=vtz

geometry={
    C
    N,1,rcn
    H,2,rnh,1,alpha}

hf
ccsd(t)
optg,root=2,hessproc=runmp2 ! 使用mp2 Hessian, 对ccsd(t)做过渡态优化
frequencies                ! CCSD(T)频率 (使用数值二阶导)

runmp2={hf;mp2}            ! 定义流程
---
```

[http://www.molpro.net/info/current/examples/hcn\\_ccsd\\_ts.com](http://www.molpro.net/info/current/examples/hcn_ccsd_ts.com)

最后一个例子显示如何用CASPT2 Hessian进行MRCI+Q (MRCI加上Davidson校正) 优化。同CCSD(T), 数值计算MRCI+Q梯度, 而CASPT2 Hessian通过使用解析CASPT2 梯度的有限差分获得。

```
!examples/hcn_mrci_ts.com $Revision: 2008.3 $
***, HCN <--> NHC Isomerization - Transition State Optimization and Frequencies
print,orbitals,civector
rcn=1.18 ang
rnh=1.40 ang
alpha=55 degree

basis=vtz
```

```

geometry={
    C
    N,1,rcn
    H,2,rnh,1,alpha}

closed,4                ! casscf非活性空间的全局设定

hf                        ! HF-SCF
multi                    ! CASSCF
mrci                     ! MRCI
optg,root=2,variable=energd,hessproc=runrs2    ! 用caspt2的Hessian优化mrci+q过渡态

runrs2={multi;rs2}      ! 对caspt2定义流程
---
```

[http://www.molpro.net/info/current/examples/hcn\\_mrci.ts.com](http://www.molpro.net/info/current/examples/hcn_mrci.ts.com)

#### 41.4.6 HCN — HNC异构体的反应路径

下面的输入首先优化过渡态，接下来执行两个方向的反应路径计算。并对结果绘图。

```

!examples/hcn_isomerization.com $Revision: 2009.1 $
***, HCN <---> NHC Isomerization Reaction Path
memory,1,m
basis=3-21G

rcn=1.18282 ang          ! 过渡态初始结构
rnh=1.40745 ang
alpha=55.05 degree

symmetry,x               ! Cs对称性
geometry={
    C
    N,1,rcn
    H,2,rnh,1,alpha}

int
rhf
optg,root=2,saveact=hcن.ts                ! 寻找过渡态
{optg,method=qsdpاب,dir=1, numhess=5,hesscentral,saveact=hcن.path}      ! 在正向寻找IRC

readvar,hcn.ts.act                ! 把结构重新设定为过渡态
{optg,method=qsdpاب,dir=-1,numhess=5,hesscentral,saveact=hcن.path,append} ! 在反向寻找IRC
```

```
readvar,hcn.path.act

alpha=alpha*pi/180    ! 角度转换为弧度

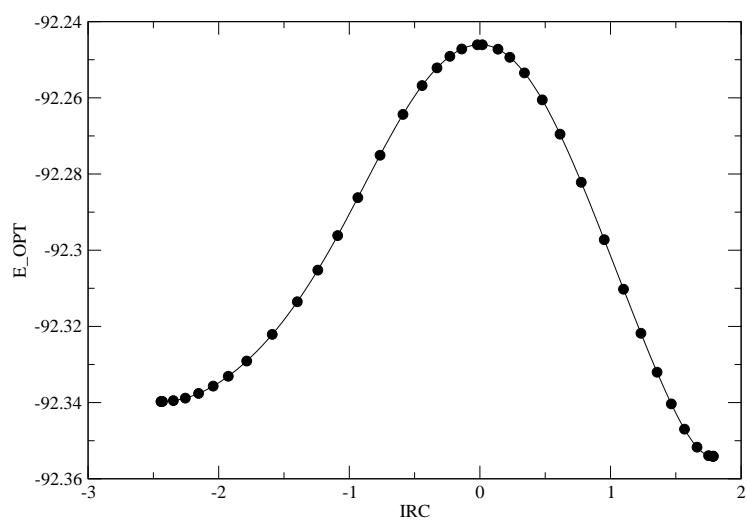
table,irc,rcn,rnh,alpha,e_opt    ! 对结果列表

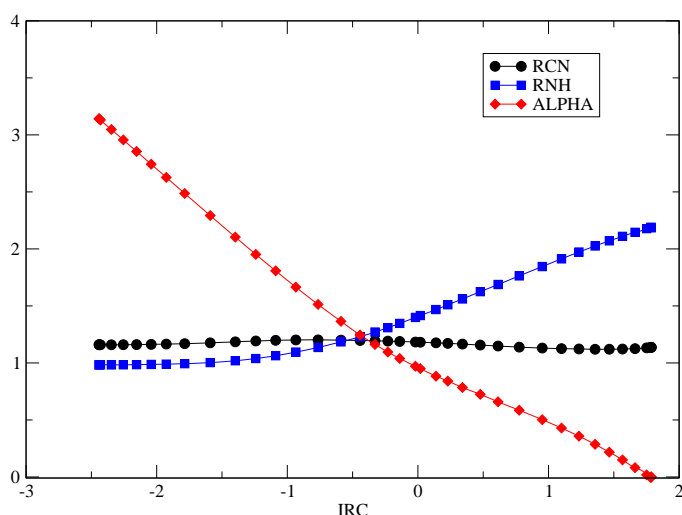
{table,irc,e_opt          ! 绘制作为irc函数的能量图
 plot,file='hcn_eopt.plot'}

{table,irc,rcn,rnh,alpha    ! 绘制作为irc函数的距离和角度
 plot,file='hcn_dist.plot'}
```

[http://www.molpro.net/info/current/examples/hcn\\_isomerization.com](http://www.molpro.net/info/current/examples/hcn_isomerization.com)

产生下面的图:





#### 41.4.7 平衡校正能量的优化

可以对整个体系执行平衡校正能量的几何优化，并且对每个单独的片段分别进行FORCE计算。使用ADD指令添加梯度和能量。为了避免由于无意间体系的转动导致错误，这需要在结构输入中指定NOORIENT。这一默认设置可以用NOCHECK选项禁止，参见上面的ADD。

平衡校正几何优化的工作方式如下面的例子所示。注意，总的平衡校正能量必须是优化的，而不能仅仅是相互作用能，因为相互作用能依赖于单体结构，并且有与总能量不同的最小值。但是如果单体的结构被冻结的话，就可以优化相互作用能。无论如何，在调用OPTG 之前，最后的计算必须是当前结构总体系的计算（在下面例子中，是二聚物的计算），否则优化程序会出现混乱。

```
!examples/hfdimer_cpcopt1.test $Revision: 2009.1 $
***,HF dimer MP2/CP optimization with relaxed monomers

basis=avtz
gthresh,energy=1.d-8

! 结构变量的初始值

RFF=      5.3
R1=       1.76
R2 =      1.75
THETA1 =  7.0
```



```

THETA2 = 111

symmetry,x
orient,noorient
geometry={
    f1
    f2 f1 rff
    h1 f1 r1 f2 theta1
    h2 f2 r2 f1 theta2 h1 180.}

label:

text, CALCULATION AT LARGE SEPARATION

rff_save=rff          ! 保存当前的rff距离
rff=1000              ! 在大间距的二聚物计算

text, HF1

dummy,f2,h2;          ! 第二个hf是虚原子
{hf;accu,16}          ! 第一个单体的scf
mp2;                  ! 第一个单体的mp2
ehf1inf=energy        ! 把mp2能量保存到变量
forces;               ! 计算第一个单体的mp2梯度

text, HF2

dummy,f1,h1;          ! 第一个hf是虚原子
{hf;accu,16}          ! 第二个单体的scf
mp2;                  ! 第二个单体的mp2
ehf2inf=energy        ! 把mp2能量保存到变量
forces;               ! 计算第二个单体的mp2梯度
add,1                 ! 加入到前一个梯度
einf=ehf1inf+ehf2inf  ! 非松弛单体的总能量

rff=rff_save          ! 把HF - HF距离重设为当前值

text, CP calculation for HF1 MONOMER

dummy,f2,h2;          ! 第二个hf是虚原子
{hf;accu,16}          ! 第一个单体的scf
mp2;                  ! 第一个单体的mp2
ehf1=energy           ! 把mp2能量保存到变量
forces;               ! 计算第一个单体的mp2梯度

```

```
add,-1          ! 从前一个梯度中减掉

text, CP calculation for HF2 MONOMER

dummy,f1,h1;      ! 第一个hf是虚原子
{hf;accu,16}      ! 第二个单体的scf
mp2;              ! 第二个单体的mp2
ehf2=energy       ! 把mp2能量保存到变量
forces;           ! 计算第二个单体的mp2梯度
add,-1           ! 从前一个梯度中减掉

text, DIMER CALCULATION
dummy            ! 重设虚原子
{hf;accu,16}     ! 二聚物的scf
mp2;             ! 二聚物的mp2
edimer=energy    ! 把mp2能量保存到变量
forces;          ! 计算二聚物的mp2梯度
add,1            ! 加到前一个梯度

optg,gradient=1.d-4,startcmd=label:  ! 寻找下一个能量

text, compute optimized monomer energy

rhf=r1
geometry={h1
          F1,H1,rhf}

{hf;accu,16}      ! 松弛单体的scf
mp2;              ! 松弛单体的mp2
ehf=energy        ! 把mp2能量保存到变量
optg              ! 优化HF结构

text,optimized geometry parameters
show,r1,r2,rhf,rff,theta1,theta2

text,computed interaction energies
decpc=(einf-ehf1-ehf2)*tocm  ! 平衡校正
de=(edimer-ehf1-ehf2)*tocm  ! 非松弛单体的CPC校正相互作用能
erelax=(2*ehf-einf)*tocm    ! 松弛能
derel=de-erelax             ! 松弛单体的CPC校正相互作用能
```

[http://www.molpro.net/info/current/examples/hfdimer\\_cpcopt1.com](http://www.molpro.net/info/current/examples/hfdimer_cpcopt1.com)

下一个例子显示如何用数值梯度进行同样的计算。在这种情况下，首先产生总平

衡校正能量，接下来优化。注意，ADD命令不能用于数值梯度。

```
!examples/hfdimer_cpcopt1_num.test $Revision: 2009.1 $
***,HF dimer MP2/CP optimization with relaxed monomers
```

```
basis=avtz
gthresh,energy=1.d-8
```

```
! 结构变量的初始值
```

```
RFF=      5.3
R1=      1.76
R2 =      1.75
THETA1 =  7.0
THETA2 = 111
```

```
symmetry,x
orient,noorient
```

```
geometry={
    f1
    f2 f1 rff
    h1 f1 r1 f2 theta1
    h2 f2 r2 f1 theta2 h1 180.}
```

```
label:
```

```
text, CALCULATION AT LARGE SEPARATION
```

```
rff_save=rff          ! 保存当前的rff距离
rff=1000              ! 在大间距的二聚物计算
```

```
text, HF1
```

```
dummy,f2,h2;          ! 第二个hf是虚原子
{hf;accu,16}          ! 第一个单体的scf
mp2;                  ! 第一个单体的mp2
ehf1inf=energy         ! 把mp2能量保存到变量
```

```
text, HF2
```

```
dummy,f1,h1;          ! 第一个hf是虚原子
{hf;accu,16}          ! 第二个单体的scf
mp2;                  ! 第二个单体的mp2
ehf2inf=energy         ! 把mp2能量保存到变量
einf=ehf1inf+ehf2inf   ! 非松弛单体的总能量
```

```
rff=rff_save          ! 把HF - HF距离重设为当前值

text, CP calculation for HF1 MONOMER

dummy,f2,h2;          ! 第二个hf是虚原子
{hf;accu,16}          ! 第一个单体的scf
mp2;                  ! 第一个单体的mp2
ehf1=energy           ! 把mp2能量保存到变量

text, CP calculation for HF2 MONOMER

dummy,f1,h1;          ! 第一个hf是虚原子
{hf;accu,16}          ! 第二个单体的scf
mp2;                  ! 第二个单体的mp2
ehf2=energy           ! 把mp2能量保存到变量

text, DIMER CALCULATION
dummy                ! 重设虚原子
{hf;accu,16}          ! 二聚物的scf
mp2;                  ! 二聚物的mp2
edimer=energy         ! 把mp2能量保存到变量
etot=edimer-ehf2-ehf1+ehf1inf+ehf2inf ! 总BSSE校正能量

optg,numerical,variable=etot,gradient=1.d-4,startcmd=label: ! 几何优化

text, compute optimized monomer energy

rhf=r1
geometry={h1
          F1,H1,rhf}

{hf;accu,16}          ! 松弛单体的scf
mp2;                  ! 松弛单体的mp2
ehf=energy            ! 把mp2能量保存到变量
optg                  ! 优化HF结构

text,optimized geometry parameters
show,r1,r2,rhf,rff,theta1,theta2

text,computed interaction energies
decpc=(einf-ehf1-ehf2)*tocm ! 平衡校正
de=(edimer-ehf1-ehf2)*tocm ! 非松弛单体的CPC校正相互作用能
```

```

erelax=(2*ehf-einf)*tocm      ! 松弛能
derel=de-erelax              ! 松弛单体的CPC校正相互作用能

```

[http://www.molpro.net/info/current/examples/hfdimer\\_cpcopt1\\_num.com](http://www.molpro.net/info/current/examples/hfdimer_cpcopt1_num.com)

在最后一个例子中，单体结构保持固定，优化相互作用能。

```

!examples/hfdimer_cpcopt2.test $Revision: 2009.1 $
***,HF dimer MP2/CP optimization without monomer relaxation

basis=avtz
gthresh,energy=1.d-8

! 结构变量的初始值

RFF=      5.3
THETA1 = 7
THETA2 = 111

symmetry,x
orient,noorient
geometry={
    f1
    f2 f1 rff
    h1 f1 1.74764059 f2 theta1
    h2 f2 1.74764059 f1 theta2 h1 180.} ! 使用孤立HF的固定HF距离

label:

text, CP calculation for HF1 MONOMER

dummy,f2,h2;          ! 第二个hf是虚原子
{hf;accu,16}          ! 第一个单体的scf
mp2;                  ! 第一个单体的mp2
ehf1=energy           ! 把mp2能量保存到变量
forces;               ! 对第一个单体计算mp2梯度
scale,-1              ! 梯度乘以-1

text, CP calculation for HF2 MONOMER

dummy,f1,h1;          ! 第一个hf是虚原子
{hf;accu,16}          ! 第二个单体的scf
mp2;                  ! 第二个单体的mp2
ehf2=energy           ! 把mp2能量保存到变量
forces;               ! 对第二个单体计算mp2梯度
add,-1                ! 从前一个梯度减去

```

```
text, DIMER CALCULATION
dummy                ! 重设虚原子
{hf;accu,16}         ! 二聚物的scf
mp2;                 ! 二聚物的mp2
edimer=energy        ! 把mp2能量保存到变量
forces;              ! 对二聚物计算mp2梯度
add,1                ! 加入到前一个梯度

optg,gradient=.d-5,startcmd=label:    ! 找到下一个能量

text,optimized geometry parameters
show,rhf,rff,theta1,theta2

text,computed interaction energies
de=(edimer-ehf1-ehf2)*tocrm          ! 使用固定单体的CPC校正相互作用能
```

[http://www.molpro.net/info/current/examples/hfdimer\\_cpcopt2.com](http://www.molpro.net/info/current/examples/hfdimer_cpcopt2.com)

## 第四十二章

# 振动频率 (FREQUENCIES)

`FREQUENCIES,options;`

计算谐振频率和简正模式。Hessian通过解析方法计算，或对 $3N$ 个直角坐标用有限差分方法进行数值计算（这种情况下Z矩阵坐标将会被破坏）。在数值差分中，如果能计算解析梯度，则进行一次差分来构建Hessian，否则对能量进行两次差分。对于波函数方法，如果有偶极矩可用，还会计算偶极导数和IR强度。注意，如果出现了使用基组的虚原子，则不能计算数值Hessian。为了得到合理的结果，在进行频率计算前需要进行几何优化。

### 42.1 选项

频率计算有以下的`options`可用：

- |                   |   |
|-------------------|---|
| <b>ANALYTICAL</b> | 使用能量的解析二阶导。 <u>目前解析二阶导只能用于闭壳层Hartree-Fock (HF) 和没有对称性的MCSCF波函。</u> 目前还不能解析计算IR强度。注意由于技术原因，在被FREQUENCIES使用之前，必须在MCSCF程序中计算MCSCF解析二阶导，例如用 <code>multi; cpmcscf,hess</code> （参见MULTI）。如果用 <code>multi; cpmcscf,hess</code> 计算了MCSCF解析二阶导，FREQUENCIES默认会使用它们。 |
| <b>CENTRAL</b>    | 使用中心差分计算高质量的力常数（默认）。  |
| <b>NUMERICAL</b>  | 使用中心差分，对能量求两次微分。  |
| <b>FORWARD</b>    | 使用前向差分计算低质量的力常数。  |

**SYMM=AUTO|NO** 在数值计算Hessian的过程中，分子的对称性可能会降低。如果给定**SYMM=AUTO**，程序在每一能量/梯度计算中将使用分子波函可能的最高对称性，因此这个选项会使计算量最小。如果使用**SYMM=NO**，则在频率计算中不使用对称性（默认）。对于单参考计算，例如HF，MP2，CCSD，RCCSD，**AUTO**选项可以安全使用，而且这也是推荐选项。但是**AUTO**选项不能用于多参考方法（MCSCF/MRCI/ACPF/AQCC/RS2）。在这些情况下即使给出也会被禁止。

**AUTO** 同**SYMM=AUTO**，见上。

**NOAUTO|NOSYM** 同**SYMM=NO**，见上。

**HESSREC|SAVE=record**

把Hessian保存到*record*记录中。默认把Hessian保存到5300.2记录。

**FREQREC=record**

把频率和简正模式保存到*record*记录中（默认是5400.2）。这一信息可用于例如VSCF/VCI程序。

**TASKREC=record**

把数值Hessian计算的任务信息保存到给定的记录中。这一信息对于数值Hessian的续算是必须的。默认把信息保存到记录5500.2。

**READ** 从默认的Hessian记录读取Hessian。

**READ|START,record**

使用以前保存在*record*的Hessian。如果对当前的方法和几何结构计算了Hessian，默认将使用它。

**RESTART** 使用默认的任务记录续算前一个数值频率/Hessian。

**RESTART=record**

使用任务记录*record*续算前一个数值频率/Hessian。

**LOW=value** 打印低振动频率的阈值，单位 $\text{cm}^{-1}$ 。如果给出这个选项，低于给定值的频率不会被打印。默认打印所有频率。

**STEP=value** 确定对能量或梯度做数值微分的步长。默认步长为0.01 a.u.。

**MAXTASK=value**

如果位移点的任务数超过此值，停止计算（以后还可以续算）。



- MAXCPU=*value* 如果超过给定的CPU时间 (秒), 停止计算 (以后还可以续算)。
- NEW 重新计算Hessian, 即使是已经计算了Hessian。
- PROJECT 把转动和平移投影到Hessian以外 (默认)。
- NOPROJECT 不把转动和平移投影到Hessian以外。
- PRINT=*value* 打印选项。如果*value*大于等于0, 将打印Hessian及其它信息 (默认-1)。
- DEBUG 打印调试信息。同PRINT=1。

为了与旧版本的MOLPRO兼容, 很多选项也可以通过指令来设置, 说明见以下小节。

## 42.2 打印选项 (PRINT)

PRINT,*options*

这个指令可用于控制输出。可以给出以下选项:

- HESSIAN 打印力常数矩阵 (Hessian), 也即, 能量的二阶导矩阵和质量加权Hessian矩阵。
- LOW 打印低振动频率 (也即, 属于转动和平移的5或6个频率) 及其简正模式 (这是默认的; LOW=-1 将禁止打印)。
- LOW=*value* 打印低振动频率的阈值, 单位 $\text{cm}^{-1}$  (默认150)。如果*value*>0, 将不会打印低于这个值的频率。

## 42.3 保存Hessian及其它信息 (SAVE)

SAVE,*options*

可以给出以下选项:

hessian=*record*

把Hessian保存到*record* (与HESSREC选项作用相同)。Hessian默认保存到记录5300.2。

FREQ=*record* 把频率和简正模式保存到*record* (与FREQREC选项作用相同)。频率默认保存到记录5400.2。

TASK=*record* 把用于续算Hessian的任务信息保存到*record* (与TASKREC选项作用相同)。任务信息默认保存到记录5500.2。

## 42.4 续算Hessian或频率 (START)

START,*options*

可以给出以下选项：

HESSIAN=*record*

从*record*记录读取Hessian（与READHESS选项作用相同）。

TASK=*record* 从*record*记录读取任务信息，并续算数值Hessian（与RESTART选项作用相同）。

## 42.5 用于数值Hessian计算的坐标 (COORD)

COORD,*type*

*type*可以是以下之一：

UNIQUE 在数值Hessian计算中，使用对称唯一位移（默认）。

3N 不使用对称唯一位移（不推荐）。

## 42.6 用于数值Hessian计算的步长 (STEP)

[STEP,*rstep*]

确定能量或梯度的数值微分步长。默认步长为*rstep*=0.01 a.u.。

## 42.7 使用能量变量的数值Hessian (VARIABLE)

VARIABLE,*name*;

定义保存能量值的变量*name*，用于有限差分方法计算Hessian。默认是最近一次程序设置的ENERGY(1)。其它可用的变量参见41.2.17节。注意，当出现使用基函数的虚原子时，不能计算数值Hessian。

## 42.8 热力学性质 (THERMO)

还可以计算分子的热力学性质。因为MOLPRO只能操作阿贝尔点群，因此需要在输入文件中给出分子的点群：

THERMO,SYM=*pointgroup*

其中*pointgroup*必须是熊夫利符号（例如，氨是C3v，线型分子分别为C\*v或D\*h）。如果没有给出点群卡，那么转动简并度将设置为1，最后会导致转动熵的偏差。如果没有给

出其它的输入卡，将对标准温度和压强 ( $T = 298.150$  [K],  $p = 1$  [atm]) 计算零点振动能和焓  $H(t) - H(0)$  [kJ/mol], 热容  $C_v$  [J/mol K] 和熵  $S$  [J/mol K]。

THERMO的子命令有：

PRINT,THERMO 把附加信息打印到输出中（例如原子质量，配分函数，以及卡为单位的热力学函数）。

SCALE,*factor* 在计算热力学性质时，为了考虑波函的系统误差，使用经过*factor*换算的振动频率（例如对SCF波函，合理的因子是*factor*=0.89）。

TEMP,*tmin*,*tmax*,*tstep*

计算不同温度下的热力学性质，从*tmin* [K]到*tmax* [K]，步长为*tstep* [K]。

PRESSURE,*p* 计算给定压强*p* [atm]下的热力学性质。

FREQUENCIES程序设定了包含原子单位的谐振零点能的变量*zpe*。如果使用THERMO选项，那么程序还设定变量*htotal*和*gtotal*，分别包含原子单位的体系的焓和自由焓。

## 42.9 示例

```
! $Revision: 2009.1 $
***,formaldehyde frequency calculation
memory,8,m

basis=vdz
gthresh,energy=1.d-8

geomtyp=xyz
symmetry,nosym
geometry={
  4
  FORMALDEHYDE
  C      0.0000000000      0.0000000000     -0.5265526741
  O      0.0000000000      0.0000000000      0.6555124750
  H      0.0000000000     -0.9325664988     -1.1133424527
  H      0.0000000000      0.9325664988     -1.1133424527
}

hf;accu,14
optg;coord,3n;
```

```
{frequencies,analytic
thermo,sym=c2v
print,thermo}

mp2
optg;coord,3n
{frequencies
thermo,sym=c2v
print,thermo}
```

[http://www.molpro.net/info/current/examples/form\\_freq.com](http://www.molpro.net/info/current/examples/form_freq.com)

```
! $Revision: 2009.1 $
***, Phosphorous-pentafluoride Vibrational Frequencies
memory,1,m
basis=3-21G

geomtyp=xyz      ! 使用xmol风格的笛卡尔坐标
symmetry,nosym   ! 不使用对称性
geometry={        ! 结构输入;
6
  PF5
  P      0.00000      0.00000      0.00000
  F      0.00000      1.11100     -1.12400
  F      0.00000     -1.52800     -0.40100
  F      0.00000      0.41700      1.52500
  F     -1.60400      0.00000      0.00000
  F      1.60400      0.00000      0.00000}

rhf
optg              ! 结构优化

frequencies       ! 计算振动频率
print,low         ! 打印低频的频率和模式
thermo,sym=d3h    ! 计算热力学性质
temp,200,400,50   ! 温度范围200 - 400 [K]
---
```

[http://www.molpro.net/info/current/examples/pf5\\_freq.com](http://www.molpro.net/info/current/examples/pf5_freq.com)

```
! $Revision: 2006.1 $
include procedures
geometry={0;      ! 水的Z矩阵
          H1,0,R;
          H2,0,R,H1,THETA}
```

```
R=0.96 Ang          ! 起始键长
Theta=104           ! 起始键角
basis=6-31g**       ! Pople基组
$functional=b3lyp    ! 定义泛函 (可选, 默认为b3lyp)
freqdft             ! 运行频率计算
```

[http://www.molpro.net/info/current/examples/h2o\\_freqdft.com](http://www.molpro.net/info/current/examples/h2o_freqdft.com)

```
! $Revision: 2006.0 $
***, Diboran Z-Matrix
memory,1,m
gthresh,energy=1.d-10,thrgrad=1.d-10
basis=sto-3g
r1=0.9023299 ang
r2=0.9722889 ang
r3=1.1540618 ang
a1=118.723 degree
```

```
geometry={
    Q1
    B1,Q1,r1
    H1,Q1,r2,B1,90
    B2,Q1,r1,H1,90,B1,180
    H2,Q1,r2,B1,90,H1,180
    H3,B1,r3,Q1,a1,H1,90
    H4,B1,r3,Q1,a1,H1,-90
    H5,B2,r3,Q1,a1,H1,90
    H6,B2,r3,Q1,a1,H1,-90}
```

```
hf
optg,gradient=1.d-4
```

```
{frequencies,coord=nosym
thermo,sym=d2h
print,thermo}
```

```
{frequencies,coord=sym,new
thermo,sym=d2h
print,thermo}
```

[http://www.molpro.net/info/current/examples/diboran\\_freq.com](http://www.molpro.net/info/current/examples/diboran_freq.com)



## 第四十三章

## 函数最小化

可以用以下命令执行一个或多个变量一般函数的最小化：

`MINIMIZE, func, x1[, x2, x3, ...]`

其中的`func`表示直到50个变量 $x_1, x_2, \dots$ 的函数。可以选择两种不同的优化方法，说明如下，它们使用或不使用数值导数信息。

可以用`METHOD`指令选择优化方法以及对`func`更精细的控制：

`METHOD, key [, key1=value, key2=value, ...]`

其中的`key`定义优化方法。`key`的有效选项有：

**BFGS**            Broyden-Fletcher-Goldfarb-Shanno共轭梯度方法，使用数值梯度（默认）。

**SIMPLEX**        Downhill单形方法，仅使用函数求解。

对于这些方法，选项`key1, key2, ...`为：

**VARSCALE=vscale**    在换算后的变量空间中优化。  
                         `vscale=0`：无换算（不推荐）  
                         `vscale=1`：在 $\ln(x)$ 的空间中优化  
                         `vscale=2`：在初始值换算的空间中，例如 $x_1/x_{1i}$ ，进行优化（默认）。

**THRESH=thresh**    优化的梯度（BFGS）或函数值（SIMPLEX）的精度。默认为 $1 \cdot 10^{-4}$ 。

**VSTEP=epsd**        数值梯度（BFGS）或初始SIMPLEX顶点的步长。

`PROC=procname` 指定在每一优化步骤执行的流程。这定义一个完整的函数求解（如果需要，还用这个流程求解数值梯度）。

`STARTCMD=command` 指定开始的命令。在每一个优化步骤，处理从`command`开始到当前MINIMIZE的所有输入。

各类指令：

`MAXIT=maxit` 优化循环的最大数量。默认对BFGS是30，SIMPLEX是 100。

## 43.1 例子

### 43.1.1 结构优化

```
!examples/min_optgeo.com $Revision: 2008.2 $
***, Simple geometry optimization

basis=vdz

geometry={
0
H 1 r
H 1 r 2 theta}

r=1.8
theta=104

hf
mp2

{minimize,energy,r,theta}
---
```

[http://www.molpro.net/info/current/examples/min\\_optgeo.com](http://www.molpro.net/info/current/examples/min_optgeo.com)

### 43.1.2 基函数优化

```
!examples/basisopt_simple.com $Revision: 2008.2 $
***, Optimization of 2 d functions

geometry={Ne}

dexp=[2.0,1.0]
```



```
basis={
sp,Ne,vdz;c;
d,Ne,dexp(1),dexp(2)
}

hf
mp2
eval=energy

minimize,eval,dexp(1),dexp(2)
---
```

[http://www.molpro.net/info/current/examples/basisopt\\_simple.com](http://www.molpro.net/info/current/examples/basisopt_simple.com)

```
!examples/basisopt_proc.com $Revision: 2008.2 $
***, Optimization of 2 d functions

geometry={Ne}

dexp=[2.0,1.0]

{minimize,eval,dexp(1),dexp(2)
method,bfgs,varscale=1,thresh=1e-5,proc=optd}

proc optd

basis={
sp,Ne,vdz;c;
d,Ne,dexp(1),dexp(2)
}

hf
mp2
eval=energy

endproc
```

[http://www.molpro.net/info/current/examples/basisopt\\_proc.com](http://www.molpro.net/info/current/examples/basisopt_proc.com)

```
!examples/basisopt_cv.com $Revision: 2008.2 $
***, MP2 optimization of core-valence cc-pCVDZ functions

geometry={Ne}
```

```
sexp=20.  
pexp=30.  
  
{minimize,ecv,sexp,pexp  
method,bfgs,varscale=1,thresh=1e-5,proc=myopt}  
  
proc myopt  
  
basis={  
spd,Ne,vdz;c;  
s,Ne,sexp  
p,Ne,pexp  
}  
  
hf  
{mp2;core,1}  
eval=energy  
  
{mp2;core,0}  
eall=energy  
  
ecv=eall-eval  
  
endproc
```

[http://www.molpro.net/info/current/examples/basisopt\\_cv.com](http://www.molpro.net/info/current/examples/basisopt_cv.com)

## 第四十四章

## 基组外推

可以用关联一致基组进行基组外推，使用命令  
`EXTRAPOLATE,BASIS=basislist,options`

其中的*basislist*是至少两种基组的基组列表，中间用冒号隔开，例如AVTZ:AVQZ:AV5Z。有些外推类型需要三种或更多的基组。其它的只需要两种。默认为使用 $n^{-3}$ 关联能外推，这种情况下，需要两种基组及其相应的能量。默认不对参考能量（HF）进行外推；用最大基组获得的值将用做CBS估计的参考能量。但是，也可以通过指定METHOD\_R选项，对参考能量做外推。

对MP2或CCSD(T)一类的标准模型进行外推的最简单方法，是使用诸如

```
***,H2O
memory,32,m
gthresh,energy=1.d-8

r = 0.9572 ang, theta = 104.52
geometry={0;
          H1,0,r;
          H2,0,r,H1,theta}

basis=avtz
hf
ccsd(t)
extrapolate,basis=avqz:av5z

table,basissets,energr,energy-energr,energy
head,basis,ehf,ecorr,etot
```

这将用AVTZ基组执行第一个计算，接下来用AVQZ和AV5Z基组计算估计的基组极限。在extrapolate命令之前的计算中获得的关联能（本例是CCSD(T)能量），将用于外推，而所需的一系列计算（这里是HF;CCSD(T)）将会自动执行。

计算的能量将返回到变量ENERGR（参考能量），ENERGY（总能量），和ENERGD（Davdison 校正能量，如果有的话）；相应的基组将返回到变量BASISSETS中。可以打印结果，例如上面的表格形式，或者用其它方式调用。以上的输入产生表格

BASIS	EHF	ECORR	ETOT
AVQZ	-76.06600082	-0.29758099	-76.36358181
AV5Z	-76.06732050	-0.30297495	-76.37029545
CBS	-76.06732050	-0.30863418	-76.37595468

外推的总能量还返回到变量ECBS中（ECBSD用于Davdison校正能量，如果有的话）。

若要对HF能量进行外推（用指数外推），需要三个能量。可以把输入做如下修改：  
extrapolate,basis=avtz:avqz:av5z,method\_r=ex1,npc=2  
method\_r确定参考能量的外推方法（本例中是单指数）；npc=2表示只有最后两个能量用于关联能外推（默认对所有给定的能量用最小二乘拟合）。这将产生：

BASIS	EREF	ECORR	ETOT
AVTZ	-76.06061330	-0.28167606	-76.34228936
AVQZ	-76.06600082	-0.29758099	-76.36358180
AV5Z	-76.06732050	-0.30297495	-76.37029545
CBS	-76.06774863	-0.30863419	-76.37638283

除了用上面的默认流程以外，还可以指定用于能量计算的流程，例如，

```
extrapolate,basis=avtz:avqz:av5z,proc=runccsd, method_r=ex1,npc=2}

procedure runccsd
hf
ccsd(t)
endproc
```

还可以用EREF，ECORR，ETOT等变量提供能量。它们必须是矢量，保存所有基组的值。

## 44.1 选项

可用的选项和外推方法有：

BASIS= <i>basissets</i>	指定一组关联一致基组，用冒号隔开。
PROC= <i>procname</i>	指定用于能量计算的流程。
STARTCMD= <i>command</i>	用于能量计算的起始命令：运行从STARTCMD到当前EXTRAPOLATE的一系列命令。在输入中，STARTCMD必须出现在EXTRAPOLATE之前。
METHOD= <i>key</i>	指定一个定义外推函数的关键词，见第44.2节。
METHOD_C= <i>key</i>	指定一个定义关联能外推函数的关键词，见第44.2节。
METHOD_R= <i>key</i>	指定一个定义参考能量外推函数的关键词，见第44.2节。
VARIABLE= <i>name</i>	指定变量名；此变量保存了进行外推的能量。
ETOT= <i>variable</i>	<i>variable</i> 提供总能量（需要在矢量中给出和基组数目一样多的能量）。如果只给出ETOT但没有给出EREF，将对总能量做外推。
EREF= <i>variable</i>	<i>variable</i> 提供进行外推的参考能量（需要在矢量中给出和基组数目一样多的能量）。
ECORR= <i>variable</i>	<i>variable</i> 提供进行外推的关联能（需要在矢量中给出和基组数目一样多的能量）。
ECORRD= <i>variable</i>	<i>variable</i> 提供进行外推的Davidson校正关联能（需要在矢量中给出和基组数目一样多的能量）。如果同时给出ECORR和ECORRD，那么二者都进行外推。
MINB= <i>number</i>	用于外推的第一个基组（默认1）。
MAXB= <i>number</i>	用于外推的最后一个基组（默认为基组数目）。
NPR= <i>number</i>	如果给出，那么最后一个NPR值用于参考能量的外推。NPR必须小于等于基组数目。
NPC= <i>number</i>	如果给出，那么最后一个NPC值用于参考能量的外推。NPC必须小于等于基组数目。
XR= <i>array</i>	提供指数的矢量，用于定义LX函数中参考能量的外推函数。
XC= <i>array</i>	提供指数的矢量，用于定义LX函数中关联能的外推函数。
PR= <i>array</i>	提供常数 $p$ ，用于定义参考能量的外推函数。
PC= <i>array</i>	提供常数 $p$ ，用于定义关联能的外推函数。

## 44.2 外推函数

选择外推函数可以给关键词加上METHOD，METHOD\_R，和/或METHOD\_C选项。默认函数为L3。在下面， $n$ 代表基组的基数（例如，2用于VDZ，3用于VTZ，等）。 $x$ 是任意数。 $p$ 是PR或PC选项给定的常数（默认 $p = 0$ ）。 $X$ 是XR或XC选项给定的数或矢量（仅用

于LX;  $n_x$ 是 $X$ 中的元素数量)。  $A$ ,  $B$ , 和 $A_i$ 是用最小二乘拟合方法优化的拟合因子。

$$\begin{array}{ll}
 \text{Lx} & E_n = E_{\text{CBS}} + A \cdot (n + p)^{-x} \\
 \text{LHx} & E_n = E_{\text{CBS}} + A \cdot (n + \frac{1}{2})^{-x} \\
 \text{LX} & E_n = E_{\text{CBS}} + \sum_{i=1}^{n_x} A_i \cdot (n + p)^{-x(i)} \\
 \text{EX1} & E_n = E_{\text{CBS}} + A \cdot \exp(-C \cdot n) \\
 \text{EX2} & E_n = E_{\text{CBS}} + A \cdot \exp(-(n - 1)) + B \cdot \exp(-(n - 1)^2)
 \end{array}$$

下面的例子显示了各种可能的的外推:

```
!examples/h2o_extrapolate_ccsd.com $Revision: 2008.2 $

***,h2o
memory,32,m

gthresh,energy=1.d-9
basis=avtz

r = 0.9572 ang, theta = 104.52
geometry={!nosym
          0;
          H1,0,r;
          H2,0,r,H1,theta}

hf
{ccsd(t)}
text,compute energies, extrapolate reference energy using EX1 and correlation energy using L3
extrapolate,basis=avtz:avqz:av5z,method_c=l3,method_r=ex1,npc=2

ehf=energr(1:3)
etot=energy(1:3)

text,extrapolate total energy using EX2
extrapolate,basis=avtz:avqz:av5z,etot=etot,method=ex2

text,extrapolate reference energy by EX1 and correlation energy by EX2
extrapolate,basis=avtz:avqz:av5z,etot=etot,method_c=ex2,eref=ehf,method_r=ex1

text,extrapolate reference energy by EX1 and correlation energy by LH3
extrapolate,basis=avtz:avqz:av5z,etot=etot,method_c=LH3,eref=ehf,method_r=ex1,npc=2

text,extrapolate reference energy by EX1 and correlation energy by LX
extrapolate,basis=avtz:avqz:av5z,etot=etot,method_c=LX,eref=ehf,method_r=ex1,xc=[3,4],pc=0.5
```

[http://www.molpro.net/info/current/examples/h2o\\_extrapolate\\_ccsd.com](http://www.molpro.net/info/current/examples/h2o_extrapolate_ccsd.com)

第二个例子显示MRCI能量的外推。在本例中，MRCI和MRCI+Q能量均进行外推。

```
!examples/h2o_extrapolate_mrci.com $Revision: 2008.2 $
***,h2o
memory,32,m

gthresh,energy=1.d-9
basis=avtz

r = 0.9572 ang, theta = 104.52
geometry={
    0;
    H1,0,r;
    H2,0,r,H1,theta}

hf
multi
mrci
text,Compute energies, extrapolate reference energy using EX1 and correlation energy using L3;
text,The Davidson corrected energy is also extrapolated
extrapolate,basis=avtz:avqz:av5z,method_c=l3,method_r=ex1,npc=2

emc=energr
ecorr_mrci=energy-emc
ecorr_mrciq=energd-emc

text,Extrapolate reference energy by EX1 and correlation energy by LH3
text,The Davidson corrected energy is also extrapolated
extrapolate,basis=avtz:avqz:av5z,ecorr=ecorr_mrci,ecorrd=ecorr_mrciq,method_c=LH3,eref=emc,method_r=ex1,npc=2
```

[http://www.molpro.net/info/current/examples/h2o\\_extrapolate\\_mrci.com](http://www.molpro.net/info/current/examples/h2o_extrapolate_mrci.com)

### 44.3 用外推能量做结构优化

可以用外推能量的数值梯度做结构优化。不能用解析梯度。有下面几种情况：

- 1) 如果OPTG直接接在EXTRAPOLATE命令之后，那么自动优化外推能量（在EXTRAPOLATE和OPTG之间仅允许设定变量）。

例子：

对最后一个命令做能量外推：

```
!examples/h2o_extrapol_opt1.com $Revision: 2008.3 $
```

```
geometry={o;h1,o,r;h2,o,r,h1,theta}  
theta=102  
r=0.96 ang  
basis=vtz  
  
hf  
ccsd(t)  
extrapolate,basis=vtz:vqz  
  
optg
```

[http://www.molpro.net/info/current/examples/h2o\\_extrapol\\_opt1.com](http://www.molpro.net/info/current/examples/h2o_extrapol_opt1.com)

对流程中计算的能量做外推:

```
!examples/h2o_extrapol_opt2.com $Revision: 2008.3 $  
  
geometry={o;h1,o,r;h2,o,r,h1,theta}  
theta=102  
r=0.96 ang  
  
proc ccsdt  
hf  
ccsd(t)  
endproc  
  
extrapolate,basis=vtz:vqz,proc=ccsdt  
  
optg
```

[http://www.molpro.net/info/current/examples/h2o\\_extrapol\\_opt2.com](http://www.molpro.net/info/current/examples/h2o_extrapol_opt2.com)

注意，如果EXTRAPOLATE从变量获取输入能量，则不能使用这种方法。

## 2) 用流程进行外推。

默认优化ECBS变量，但是可以在OPTG命令中用VARIABLE 选项指定其它变量（例如ECBSD）。

```
!examples/h2o_extrapol_opt3.com $Revision: 2008.3 $  
geometry={o;h1,o,r;h2,o,r,h1,theta}  
theta=102  
r=0.96 ang  
basis=vtz  
  
proc cbs34  
hf
```



```
ccsd(t)
extrapolate,basis=vtz:vqz
endproc

optg,variable=ecbs,proc=cbs34
```

[http://www.molpro.net/info/current/examples/h2o\\_extrapol\\_opt3.com](http://www.molpro.net/info/current/examples/h2o_extrapol_opt3.com)

```
!examples/h2o_extrapol_opt4.com $Revision: 2008.3 $
geometry={o;h1,o,r;h2,o,r,h1,theta}
theta=102
r=0.96 ang

proc cbs34
basis=vtz
hf
ccsd(t)
eref(1)=energr
ecc(1)=energy

basis=vqz
hf
ccsd(t)
eref(2)=energr
ecc(2)=energy
extrapolate,basis=vtz:vqz,eref=eref,etot=ecc
endproc

optg,variable=ecbs,proc=cbs34
```

[http://www.molpro.net/info/current/examples/h2o\\_extrapol\\_opt4.com](http://www.molpro.net/info/current/examples/h2o_extrapol_opt4.com)

## 44.4 用外推能量算谐振频率

这可以通过在流程中定义外推实现:

```
!examples/h2o_extrapol_freq.com $Revision: 2008.3 $
geometry={o;h1,o,r;h2,o,r,h1,theta}
theta=102
r=0.96 ang
basis=vtz

proc cbs34
hf
ccsd(t)
extrapolate,basis=vtz:vqz
```

```
endproc  
  
optg,variable=ecbs,proc=cbs34  
freq,variable=ecbs,proc=cbs34
```

[http://www.molpro.net/info/current/examples/h2o\\_extrapol\\_freq.com](http://www.molpro.net/info/current/examples/h2o_extrapol_freq.com)

## 第四十五章

## 势能曲面 (SURF)

`SURF, Start1D=label1, options`

`SURF`程序可以在平衡结构位置附近计算势能面，用于非谐振频率的计算（参见`VSCF`和`VCI`程序）。程序目前限制为一个最小点的情况。势函数用能量格点而不是泰勒展开式表示。在`SURF`程序中，势能面按照简正坐标展开。因此，首先需要进行谐振频率计算。接下来用分层方案，势函数表示为：

$$V(q_1, \dots, q_{3N-6}) = \sum_i V_i(q_i) + \sum_{i<j} V_{ij}(q_i, q_j) + \sum_{i<j<k} V_{ijk}(q_i, q_j, q_k) + \dots \quad (45.1)$$

其中，

$$V_i(q_i) = V_i^0(q_i) - V(0) \quad (45.2)$$

$$V_{ij}(q_i, q_j) = V_{ij}^0(q_i, q_j) - \sum_{r \in \{i,j\}} V_r(q_r) - V(0) \quad (45.3)$$

$$V_{ijk}(q_i, q_j, q_k) = V_{ijk}^0(q_i, q_j, q_k) - \sum_{\substack{r,s \in \{i,j,k\} \\ r>s}} V_{rs}(q_r, q_s) - \sum_{r \in \{i,j,k\}} V_r(q_r) - V(0) \quad (45.4)$$

$$V_{ijkl}(q_i, q_j, q_k, q_l) = \dots \quad (45.5)$$

其中的 $q_i$ 表示简正坐标。这个展开式需要在 $n$ 体贡献后终止，这由关键词`NDIM`控制。`SURF`程序从某总意义上说是完全并行的，不同格点的计算被发送到不同的处理器（`MPPX`方案）。`START1D`关键词是强制的，用于在输入中定义从哪里跳入电子结构计算的标记，这一部分用`SURF`命令终止。这种方法确定了势能面的质量。

`label1`

```
hf
ccsd
surf,start1D=label1
```

SURF程序基于迭代算法，即，自动把格点添加到势函数的格点表示中，直到满足收敛阈值。这就保证了势函数展开式中不同项之间的平衡描述，同时使描述势函数的从头计算数量最少。更多细节，参见：

G. Rauhut, *Efficient Calculation of Potential Energy Surfaces for the Generation of Vibrational Wave Functions*, J. Chem. Phys. **121**, 9313 (2004).

T. Hrenar, H.-J. Werner, G. Rauhut, *Accurate Calculation of Anharmonic Vibrational Frequencies of Medium Sized Molecules Using Local Coupled Cluster Methods*, J. Chem. Phys. **126**, 134108 (2007).

## 45.1 选项

可以使用以下选项：

**NDIM=值** 在 $n$ 体项后，关键词**NDIM**= $n$ 终止势能面的展开式。目前，最多可包含四体项，但默认设定为3。请注意，当使用**NDIM**=4作为SURF程序的关键词时，也必须把这一信息传递给VSCF和VCI程序。否则这些程序将忽略四体项。

**NGRID=值** 基于从头算能量点的粗糙格点，通过自动内插技术产生精细格点。关键词**NGRID**= $n$ 确定一维方向上等距离格点的数量。**NGRID**= $n$ 必须是偶数。当前默认为16。注意，格点的数量还控制 $n$ 维势能面的展开（参见关键词**SCALE**），因此影响多个内部阈值，这些都是对**NGRID**的默认值优化的。格点的数量还决定了VSCF程序中的基函数数量。目前最大的格点尺寸为20。

格点	14	16	18	20
曲面展开	4.30	4.69	5.05	5.39

**VAR1D=变量** SURF程序从MOLPRO内部变量读取电子结构计算的能量，例如，**NERGY**, **EMP2**, ...。内部变量用关键词**VAR1D**指定。在上面显示的例子中，**VAR1D=ENERGY**将读取CCSD能量，而**VAR1D=EMP2**读取MP2能量，它是CCSD计算的副产物。**VAR1D**关键词的默认值是内部变量**ENERGY**。

- DIPOLE=值** 对于MOLPRO中所有具有解析梯度的方法，可以计算偶极曲面。对于除了Hartree-Fock之外的所有方法，这需要在电子结构方法后使用关键词CPHF,1。偶极曲面的计算限定用多级别方案，其中所有的变量VAR1D, VAR2D, 和VAR3D限定为默认值，即ENERGY（参见VMULT 命令的部分）。目前，对称性只能用于1D, 2D, 和3D偶极曲面。对于4D的项，目前自动关闭对称性。计算偶极曲面在效率上是计算曲面时间的两倍。
- SYM=值** 电子结构计算的对称性可以用SYM=Auto关键词发现。通常这会显著地节省时间。默认关闭对称性识别，因为某些计算会导致问题（例如局域关联方法）。对于势能面有贡献的对称性，电子结构计算中的对称性不会出错（参见MPG关键词）。
- SCALE=值** 从Gauss-Hermite积分点确定势能面的展开。通过使用精细格点NGRID=16, 势能面会在每个方向展开成NGRID/2次的Gauss-Hermite点，即4.69次（见关键词NGRID）。在计算基频模式中，由于这些值非常大，因此引入了一个换算因子SCALE= $f$ 。默认使用换算因子0.75。增加曲面的尺寸，通常需要计算更多的从头计算点，因为曲面内插对小的曲面更稳定。
- FIT=值** 用于产生势能面的迭代算法基于不断增加内插点。一旦先后两次迭代步骤的内插变得稳定，就终止迭代。收敛阈值可以用关键词FIT= $f$ 修改。对于不同的1D, 2D, 3D, 和4D迭代，目前只有一个控制变量。四个阈值互不相同，但彼此依赖。因此，修改默认值（FIT=4.0d-2）将同时改变所有的阈值，使计算达到平衡。
- MIN1D=值** 粗糙格点的最小数量可以用关键词MIN1D, MIN2D, MIN3D, 和MIN4D控制。这四个关键词确定了每个1D, 2D, 3D, 和4D曲面的一维方向上，从头计算的最小数量。目前的默认值为MIN1D=4, MIN2D=4, MIN3D=2, MIN4D=2。
- MAX1D=值** 粗糙格点的最大数量可以用关键词MAX1D, MAX2D, MAX3D, 和MAX4D控制。这四个关键词确定了每个1D, 2D, 3D, 和4D曲面的一维方向上，从头计算的最大数量。目前的默认值为MAX1D=NGRID, MAX2D=NGRID, MAX3D=10, MAX4D=4。
- EXT12D=值** 势能面的外部区域由外推而不是内插方案确定。目前使用的默认外推为，1D和2D对势函数的贡献为15%，（Ext12D=0.85），3D和4D项的贡献为30%（Ext34D=0.7）。对于精确计算，把两个关键字设定为1.0可以关闭外推。改变这些关键词通常会增加执行的从头计算数量。

- SKIP3D=值**      由于3D和4D曲面数量增加得非常快，可以用SKIP3D和SKIP4D关键词忽略不太重要的3D和4D曲面。3D曲面预屏蔽的标准基于2D项，类似地，4D曲面使用3D项。忽略3D曲面自动导致忽略4D曲面，因为后者依赖于前者。默认打开预屏蔽，但是可以用SKIP3D=1.0和SKIP4D=1.0关闭。
- MPG=值**      简正模式的对称性可以用程序自动识别。目前只能用阿贝尔点群。程序将确定模式的对称性，即使是在电子结构计算中使用了NOSYM关键词。有些情况下，数值噪声可能会非常高，因此无法得到正确的对称符号。在这种情况下，该模式标记为Err。对于这种情况，应该在势能曲面和VCI的计算中关闭对称性，否则结果可能会出错。对称性可以用MPG=1关闭。
- INFO=值**      对于所有有关的程序参数（选项），INFO=1提供值的列表。

以下例子给出的输入在MP2/cc-pVTZ级别计算能量和偶极曲面，随后在VSCF和VCI级别确定非谐振频率。Hartree-Fock计算不会重新开始，日志（.log）文件定位到\$TMPDIR变量定义的草稿目录中。

```
memory,20,m
geomtyp=xyz
geometry={mass
  3
Water
O      0.0675762564      0.0000000000      -1.3259214590
H      -0.4362118830     -0.7612267436     -1.7014971211
H      -0.4362118830      0.7612267436     -1.7014971211
}

basis=vdz
logfile,scratch

hf
mp2
optg
{frequencies,symm=auto
 print,low=50}

label1
{hf
  start,atden}
{mp2
  cphf,1}
```

```
surf,start1D=label1,dipole=1,sym=auto
vscf,dipole=1,combi=1
vci,dipole=1,version=3,combi=1
```

## 45.2 多级别计算

VMULT, *options*

在势函数展开中，电子结构计算的级别可以对不同的 $i$ 体项进行修改。因此，有关键词START2D, START3D, VAR2D, 和VAR3D，它们完全类似于标准计算中的START1D和VAR1D（见上）。其中的数值总是表示展开项的级别。这种计算称为多级别计算。计算中没有对应四体项的关键词。四体项总是使用对三体项指定的变量。

**MULTI=值** 与命令VAR1D, VAR2D, 和VAR3D组合的关键词START1D, START2D, 和START3D可以计算多级别势能面。这在原则上意味着势函数的1D项需要在所有三种级别进行计算，2D项在两种级别计算。由于某些低级别结果是更复杂方法的副产物（例如HF能量是MP2计算的副产物，MP2能量是CCSD(T)计算的副产物），因此上一级计算可以用MULTI避免。

**MULTI=1:** 这是默认且最昂贵的选项。将在所有三种理论级别计算1D势函数。类似地，将在指定的两种级别计算2D势函数。例子有：

1D: CCSD(T)/cc-pVTZ

2D: MP4(SDQ)/cc-pVTZ

3D: MP2/cc-pVDZ

```
{SURF,Start1D=label1
  VMULT,Start2D=label2,Start3D=label3,Multi=1}
```

**MULTI=2:** 所有的信息用预先的计算产生，因此势函数的任何一部分都无需计算两次。例子：

1D: CCSD(T)/cc-pVTZ

2D: CCSD(T)/cc-pVTZ

3D: MP2/cc-pVTZ

```
{SURF,Start1D=label1
  VMULT,Start2D=label1,Start3D=label2
  VMULT,Var3D=EMP2,Multi=2}
```

1D: CCSD(T)/cc-pVTZ

2D: MP2/cc-pVTZ

3D: MP2/cc-pVTZ

```
{SURF,Start1D=label1  
VMULT,Start2D=label2,Start3D=label2  
VMULT,Var2D=EMP2,Var3D=EMP2,Multi=2}
```

MULTI=3: 2D势函数提供了3D部分的所有信息,,而在1D和2D之间没有联系。因此,1D的贡献需要计算两次(在1D和2D级别),而所有其它的项仅计算一次。例子:

1D: CCSD(T)/cc-pVTZ

2D: MP4(SDQ)/cc-pVTZ

3D: MP2/cc-pVTZ

```
{SURF,Start1D=label1  
VMULT,Start2D=label2,Start3D=label3  
VMULT,Var3D=EMP2,Multi=3}
```

1D: CCSD(T)/cc-pVTZ

2D: MP4(SDQ)/cc-pVTZ

3D: MP4(SDQ)/cc-pVTZ

```
{SURF,Start1D=label1  
VMULT,Start2D=label2,Start3D=label2,Multi=3}
```

MULTI=4: 1D计算提供2D势函数的所有信息,但不提供3D部分的信息。因此,1D和2D贡献都需要计算两次。例子:

1D: CCSD(T)/cc-pVTZ

2D: CCSD(T)/cc-pVTZ

3D: MP4(SDQ)/cc-pVTZ

```
{SURF,Start1D=label1  
VMULT,Start2D=label1,Start3D=label2,Multi=4}
```

1D: CCSD(T)/cc-pVTZ

2D: MP2/cc-pVTZ

3D: MP2/cc-pVDZ



```
{SURF,Start1D=label1
  VMULT,Start2D=label2,Start3D=label3
  VMULT,Var2D=EMP2,Multi=4}
```

在2D和4D计算中（即，NDIM=2,4），还可以用VMULT命令。在4D计算中，最后的级别必须总是等于3D的级别。在2D中，MULTI=1和MULTI=3的含义是一样的。类似地，在2D计算中，MULTI=2和MULTI=4的含义是一样的。

**START2D=标记** 为了在不同于1D项的电子结构理论级别计算2D和3D项，可以用START2D和START3D定义输入串中的标记。用START2D和START3D命令通常需要在输入中使用GOTO命令。

**VAR2D=变量** VAR2D和VAR3D关键词的定义与VAR1D选项类似。它们对给定的格点指定读取的内部变量（例如，ENERGY，EMP2，CCSD，...）。

下面的例子显示多级别计算：

1D: CCSD(T)/cc-pVTZ

2D: MP4(SDQ)/cc-pVTZ

3D: MP2/cc-pVTZ multi-level

由于MP2能量是CCSD(T)和MP4(SDQ)计算的副产物，因此只有1D格点计算两次（在CCSD(T)和MP4(SDQ)级别）。能够将从内部变量ENERGY获得1D和2D能量，而3D能量使用EMP2变量。

```
memory,50,m
geomtyp=xyz
geometry={ang;mass
6
Ethen
C      0.0000000000      0.0000000000     -0.6685890718
C      0.0000000000      0.0000000000      0.6685890718
H      0.0000000000     -0.9240027061     -1.2338497710
H      0.0000000000      0.9240027061     -1.2338497710
H      0.0000000000      0.9240027061      1.2338497710
H      0.0000000000     -0.9240027061      1.2338497710
}

basis=vtz
logfile,scratch

hf
```

```
ccsd(t)
optg
freq,symm=auto

label1
hf
ccsd(t)
goto,label4

label2
{hf
  start,atden}
{mp4
  notrip1}
goto,label4

label3
{hf
  start,atden}
mp2

label4
{surf,start1D=label1,sym=auto
  vmult,start2D=label2,start3D=label3,Var3D=EMP2,Multi=3}
vscf
vci
```

### 45.3 续算功能

DISK,*options*

由于SURF计算非常昂贵，因此强烈建议把势函数格点的描述转存到硬盘上。有两个不同的选项：

- 转存到二进制的.wfu文件中（SAVE和RESTART选项）
- 转存到外部的ASCII格式文件中（WHERE, DUMP, 和DISK选项）

从二进制文件续算只能用于SURF计算已经完成的情况。对于SURF计算失败的情况，续算只能用外部文件。

SAVE=值      一旦在MOLPRO输入中声明了永久文件2（file,2,filename.wfu），用SURF程序产生的势函数默认转存到硬盘。默认的记录是5600.2，但这

个值可以用SAVE命令修改。这可以用来在单独的任务中运行非谐频率计算，无需再重算势函数。

**RESTART=值** 一旦完成了整个势能面的计算，可以从二进制文件2续算，这个文件在MOLPRO输入中（参见SAVE命令）用RESTART选项指定，例如，Restart =5600.2。对于被中断的SURF 计算，用RESTART选项的续算会失败，该选项只能从完成的SURF计算续算。注意，进行续算总是需要指定SURF卡。不能简单地用VSCF卡进行续算。即使是5600.2记录中的势函数完全可用。

**WHERE=值** 与关键词DUMP和EXTERN结合使用，用来指定外部续算文件，关键词WHERE指定外部ASCII文件的路径。可以用两个选项，WHERE=home和WHERE=scr。用于SURF计算的外部文件可能很大，它们默认将存储于MOLPRO变量\$TMPDIR定义的草稿盘中

**DUMP=文件名** 势函数可以转存到外部的ASCII文件中，用于续算。它的名称必须提供为DUMP关键词的参数，例如，UMP='formaldehyde.pot'。ASCII文件提供了到其它程序的接口，并且可以控制存储和修改计算的势函数。如果可以，也转存偶极曲面。

**EXTERN=文件名** 原则上在截断计算的任何点上都能续算SURF。但是，由于只有完全收敛的差分势函数才会存储于ASCII文件中（DUMP），因此这是不可能的。因此，一旦一批曲面完成转存后，SURF计算就可以从转存文件续算。由于产生2D或3D曲面需要两到三次批处理，因此3D势函数的曲面计算有6-10个续算点。从ASCII转存文件的续算点可以用任何类型的VMULT计算。由于还从外部文件读入简正模式和谐振频率，因此对这种续算不需要重新计算谐振频率。

## 45.4 简正模式的线性组合

LINCOMB, *options*

对于势的展开，用LINCOMB指令可以计算简正坐标的线性组合。这用 $2 \times 2$ 的Jacobi转动实现。输入中最多可以提供 $3N-6/2$ 个转动。

NM1=*n*, NM2=*m* 表示进行转动的两个简正坐标。

ANGLE=值 转动角度，单位：度。

## 45.5 个别坐标的换算

SCALNM, *options*

SURF程序的SCALE选项可以用一个公共因子修改所有做差分的势的范围。与之相比，SCALNM 指令可以换算个别简正坐标。对于四次项而不是二次项为主的势，这是推荐的选择。最多可以提供3N-6个不同的换算因子。

MODE=*n*            表示进行换算的简正坐标。

SFAC=值            MODE模式的换算因子。默认为1.0。

## 45.6 建议

建议：

- 在结构定义部分使用MASS关键词，用来把分子转动到标准方位。在势能面的生成中，这对完全探测对称性是必须的。
- 把计算分为三部分：
  - (1) 进行非常准确的结构优化并把优化的结构放到标准方位，基于优化的结构开始第二个计算，
  - (2) 进行谐振频率计算，把它存储到续算文件（.wfu）中。做一个续算文件的备份。
  - (3) 从续算文件（.wfu）开始一个SURF，VSCF，和VCI计算，并把势能面转存到一个外部文件中。以后就可以从外部文件续算VSCF和VCI，不用重算曲面。
- 为了加速计算，使用1D: CCSD(T)/vtz，2D: MP4(SDQ)/vtz，和3D: MP2/vtz多级别方案，并与对称性和并行版本MOLPRO相结合。
- 降低简正坐标的质量。如果不需要十分高的精度，在MP2级别而不是CCSD(T)级别计算简正坐标（即，谐振频率计算）已经足够。这会节省大量的计算时间，而偏差通常不会大于1或2个波数。

## 45.7 标准问题

问题            SURF计算终止，出错信息为

```
?ERROR IN VIRTORB: INCORRECT NUMBER OF ORB...
```

```
ERROR EXIT
```

```
CURRENT STACK:        MAIN
```

- 解答** 当从平衡结构的参考计算续算Hartree-Fock时，程序在对称性转换上有些问题。需要重新开始独立的Hartree-Fock计算，并使用关键词`start,atden`。
- 问题** 在并行计算中 (MPPX)，VSCF计算的CPU时间完全不同于实际时间（时钟时间）。
- 解答** 有两个可能原因：(1) VSCF计算通常需要花大量的总计算时间在Hartree-Fock程序和双电子积分程序中。由于积分存储于磁盘上，同一机器上的两个处理器可能会在同一时刻往磁盘上写文件，因此计算时间依赖于磁盘控制器的某些延时。如果组合几块磁盘并使用几个控制器，效率会更高。这一问题可以通过把任务分布到几台计算机上，并把每台计算机上的处理器数限制为1来避免。(2) 积分程序对积分做了缓冲。并行任务可能需要大量的内存（乘上因子2 加上共享内存），因此积分缓冲的效率很低。可以尝试尽可能减少内存消耗。把耗内存的VCI 计算与SURF计算分开可能会更有利。



## 第四十六章

### VSCF程序 (VSCF)

VSCF, options

VSCF程序基于Watson哈密顿量:

$$\hat{H} = \frac{1}{2} \sum_{\alpha\beta} (\hat{J}_\alpha - \hat{\pi}_\alpha) \mu_{\alpha\beta} (\hat{J}_\beta - \hat{\pi}_\beta) - \frac{1}{8} \sum_{\alpha} \mu_{\alpha\alpha} - \frac{1}{2} \sum_i \frac{\partial^2}{\partial q_i^2} + V(q_1, \dots, q_{3N-6}) \quad (46.1)$$

其中的势能面 $V(q_1, \dots, q_{3N-6})$ 由SURF模块提供。默认关闭振动角动量 (Coriolis耦合) 项。对于求解一维Schrödinger方程, 默认使用Hamilton和Light的DVR流程。注意, 基函数 (分布高斯函数) 的数量由势函数的格点NGRID决定, 在不改变势能面的格点描述时, 不能增加。此外, 对于分布高斯函数 (DG), 基是固定的。由于VSCF计算非常快, 因此这些计算不能续算。细节参见:

G. Rauhut, T. Hrenar, *A Combined Variational and Perturbational Study on the Vibrational Spectrum of  $P_2F_4$* , Chem. Phys. **346**, 160 (2008).

### 46.1 选项

可以使用以下选项:

**PMP=value**      默认关闭振动角动量项 (Coriolis耦合), 即  $\frac{1}{2} \sum_{\alpha\beta} \hat{\pi}_\alpha \mu_{\alpha\beta} \hat{\pi}_\beta$ , 以及Watson校正项。PMP=1会添加Watson校正项 (见方程46.1), 以赝势类的贡献添加到势的精细格点上。PMP=2可以计算PMP算符的积分, 使用的近似是 $\mu$ 张量给定为平衡结构上转动惯量张量的反转。

- COMBI=***value* VSCF程序默认只计算分子的基模。但是，选择**COMBI=1**还可以计算第一振动泛频，以及两个模式的第一振动能级构成的 $n \times (n - 1)/2$ 个组合带。
- SOLVER=***value* 对于求解一维Schrödinger方程，可以使用两种不同的算法。默认为**SOLVER=1**，调用Hamilton和Light建议的离散变量表示（discrete variable representation, DVR）。也可以用Young和Peet的组合算法（**SOLVER=2**）。
- THERMO=***value* **THERMO=1**用于改善热力学特性的计算（与谐振频率组合的**THERMO**关键字相比较）。但是这里使用了近似的方法：在配分函数的方程中保留谐振近似，但这些函数所用的实际频率值是来自VSCF计算的非谐频率值。
- ROTJ=***value* 只能用近似方法计算振转能级（不能与**COMBI**关键字结合使用）。一旦求解了VSCF方程，将从每个振动能级的振动平均结构计算转动常数。这可以粗略估计和快速算出振转能级。**ROTJ=n**决定了 $J$ 的值。负数的 $J$ 将计算从 $J = 1$ 到指定 $J$ 值的所有振转能级。
- DIPOLE=***value* **DIPOLE=1**用于计算红外强度。红外强度的计算需要在SURF程序中计算偶极曲面。默认不计算强度。
- NDIM=***value* 在VSCF计算中，势函数的展开可以与SURF计算中的展开不同。但是，使用的值只能小于等于势能面计算所用的值。
- INFO=***value* **INFO=1**提供程序有关参数（选项）的所有值的列表。

## 46.2 标准问题

### 问题

VSCF结果非常不合理，并且对于一个或多个模式，在VSCF程序输出打印的STATUS 不是0.0。

### 解答

如果STATUS不是0.0，那么有些对应模式的本征值是负值，当然不应当是这样的。STATUS的第一个数表示在VSCF循环中，负本征值的最大个数，第二个数是最后一次迭代中的负本征值个数。当出现负本征值时，就不再升级有效势。这会强制收敛但不会追求准确性。唯一的解决办法是计算更准确的势函数，即，切换MULTI近似，或者使用更多的格点，等。



*VCI,options*

VCI计算考虑了振动关联的影响，它基于SURF程序产生的势能面和VSCF模型产生的基。所有的VCI计算对特定的态执行，即，对每个振动模式执行一次VCI计算。由于VCI计算需要大量的计算资源，因此这些计算可能会像产生势能面一样昂贵。目前可以用两个不同的VCI程序（组态选择程序和常规程序，见下）。关于不同版本的组态选择VCI程序及其中的组态选择方案，详细说明参见：

G. Rauhut, *Configuration Selection As a Route Towards Efficient Vibrational Configuration Interaction Calculations*, J. Chem. Phys. **127**, 184109 (2007).

**47.1 选项**

可以使用以下选项：

**VERSION=***value*

默认的组态选择VCI计算VERSION=1是MOLPRO中最快的VCI代码。组态选择基于振动基态，而最终的VCI矩阵是对特定态的。这一近似通过屏蔽减少了计算量。VERSION=2是特定态的组态选择VCI，它通常更可靠，但是比VERSION=1更昂贵。但是，组态选择仍旧会引入一些不准确性，对于非常准确的计算这可能是不希望的。对于特定态组态选择VCI，可以用VERSION=3选择做微小校正。这可能会使CPU时间增加到两倍。所有的组态选择VCI程序都实现了并行化，方式是把VCI矩阵的各列发送到不同的处理器中。通常的VCI计算可以选择用VERSION=4执行。这

一版本产生最准确的结果。但是在这种情况下，VCI矩阵可能会非常大，这种计算的VCI矩阵对角化通常会受CPU 需求和相应的内存需求的限制。

- CITYPE=***value* CITYPE定义同时激发的最大数量，即，单激发、双激发、三激发、...，因此确定了计算的类型，即，VCISD，VCISDT，...。默认是CITYPE=4 (VCISDTQ)，它是精度和计算速度之间非常不错的折中。目前的最大激发级别限定为CITYPE=6。
- LEVEX=***value* LEVEX确定一个模式中的最高激发级别，即， $0 \rightarrow 1$ ， $0 \rightarrow 2$ ， $0 \rightarrow 3$ ，...。默认是LEVEX=4，发现对大多数应用来说已经足够。
- CIMAX=***value* CIMAX是对应于CITYPE和LEVEX的最大激发级别。原则上三重组态(4, 4, 4)将对VCI空间有贡献。不过，CIMAX=7将限制为(4, 3, 0)，(3, 3, 1)，(3, 2, 2)，...。默认是CIMAX=8，对某些应用需要扩大。
- CITHR=***value* CITHR控制组态选择方案的阈值（参见VERSION=1-3）。默认是0.99995。
- BSIZE=***value* 在组态选择算法中，屏蔽区的尺寸用单双激发的数量来衡量，即，BSIZE=1.0。默认为BSIZE=1.5（细节参见上面给出的文献）。
- PMP=***value* 默认关闭振动角动量项（Coriolis耦合），即 $\frac{1}{2} \sum_{\alpha\beta} \hat{\pi}_{\alpha} \mu_{\alpha\beta} \hat{\pi}_{\beta}$ ，以及Watson校正项。PMP=1会添加Watson校正项（见方程46.1），以赝势类的贡献添加到势的精细格点上。PMP=2可以计算PMP算符的积分，使用的近似是 $\mu$ 张量给定为平衡结构上转动惯量张量的反转。PMP=2包含了仅在VCI矩阵中的对角贡献。这比计算全部矩阵元的贡献（对应于PMP=3）快得多，通常只会引入很小的偏差。
- COMBI=***value* VCI程序默认只计算分子的基模。但是，选择COMBI=1还可以计算第一振动泛频，以及两个模式的第一振动能级构成的 $n \times (n - 1)/2$ 个组合带。
- THERMO=***value* THERMO=1用于改善热力学特性的计算（对照与谐振频率相组合的THERMO关键字）。但是这里使用了近似的方法：在配分函数的方程中保留谐振近似，但这些函数所用的实际频率值是来自VCI 计算的非谐频率值。
- ROTJ=***value* 只能用近似方法计算振转能级（不能与COMBI选项结合使用）。一旦确定了VCI波函，将从每个振动能级的振动平均结构计算转动常数。这可以粗略估计和快速算出振转能级。ROTJ= $n$ 决定了 $J$ 的值。负数的 $J$ 将计算从 $J = 1$ 到指定 $J$ 值的所有振转能级。

DIPOLE=*value* DIPOLE=1用于计算红外强度。红外强度的计算需要在SURF程序中计算偶极曲面。默认不计算强度。

NDIM=*value* 在VCI计算中，势函数的展开可以与SURF计算中的展开不同。但是，使用的值只能小于等于SURF计算所用的值。

MPG=*value* 分子对称性可以在VCI计算中自动识别。MPG=1关闭对称性。

INFO=*value* INFO=1提供所有程序有关参数（选项）值的列表。

## 47.2 建议

建议把计算分为三部分：(1) 进行非常精确的结构优化，并把优化的结构放到标准方位（用MASS关键字），基于优化的结构开始第二个计算，进行(2) 谐振频率计算，把它存储到续算文件（.wfu）中。做一个续算文件的备份。(3) 从续算文件（.wfu）开始一个SURF，VSCF，和VCI计算，并把势能面转存到一个外部文件中（用DUMP关键字）。以后就可以从外部文件续算VSCF和VCI，不用重算曲面。

## 47.3 例子

在下面的输入例子中，(1) 优化水的结构，(2) 计算谐振频率，(3) 在平衡结构附近产生势能面，(4) 在VSCF级别计算核的波函和红外强度，最后(5) 用不同的组态选择方案执行三个不同的VCI计算。其中包含了振动角动量项。

```
memory,20,m
basis=vdz
geomtyp=xyz
geometry={mass
  3
Water
O      0.0675762564      0.0000000000      -1.3259214590
H      -0.4362118830     -0.7612267436     -1.7014971211
H      -0.4362118830      0.7612267436     -1.7014971211
}

hf
mp2
optg
{frequencies,symm=auto
print,low=50}

label1
{hf
```

```
start,atden}  
{mp2  
cphf,1}  
  
surf,start1D=label1,dipole=1,sym=auto  
vscf,pmp=2,dipole=1  
vci,pmp=3,dipole=1,version=1  
vci,pmp=3,dipole=1,version=2  
vci,pmp=3,dipole=1,version=3
```

## 第四十八章

## COSMO模型

类导体的屏蔽模型（COSMO）（A. Klamt and G. Schüürmann, J. Chem. Soc. Perkin Trans. II 799-805 (1993)）目前可用于HF（RHF,UHF）和DFT（RKS,UKS）能量和相应的梯度计算。

COSMO模型用COSMO卡调用：

`COSMO[,option1=value1, option2=value2,...]`

其中的`option`可以是

NPPA	基底层格点大小。值必须满足： $value = 10 \times 3^k \times 4^l + 2$ （默认= 1082；整型）。
NSPA	非氢原子的片段数。值必须满足： $values = 10 \times 3^k \times 4^l + 2$ （默认= 92；整型）。
CAVITY	分子面的交叉线是封闭的（1）还是开放的（0）（默认= 1；整型）。
EPSILON	电介质的介电常数（默认= -1.d0，表示 $\epsilon = \infty$ ；实型）。
DISEX	对A矩阵设定的距离标准。短程相互作用（片段中心间距 $< DISEX \times$ 平均原子直径）用基底层格点计算。长程相互作用通过片段中心计算（默认= 10.d0；浮点型）。
ROUTF	用于外部空穴构造的因子。外部空穴的半径定义为： $r_i^{out} = r_i + ROUTF \times RSOLV$ （默认= 0.85d0；浮点型）。
PHSRAN	坐标随机化中的相偏移（默认= 0.d0；浮点型）。
AMPAN	坐标随机化中的振幅因子（默认= 1.0d-5；浮点型）。

- RSOLV**            空穴结构的附加半径（默认= -1d0，使用优化的H半径；浮点型）。
- MAXNPS**        表面片段的最大数量（默认= -1，由程序估计；整型）。

只有在出现问题的情况下，才建议改变默认值。

程序默认使用优化的半径，如果存在的话，否则用 $1.17 \times \text{vdW}$ 半径。优化的半径为：  
H=1.30，C=2.00，N=1.83，O=1.72，F=1.72，S=2.16，Cl=2.05，Br=2.16，I=2.32 [Å]。用户个人的值可以在随后的cosmo卡中直接给出：

**RAD**, *symbol*, *radius*

其中*radius*必须用Å给出。

例子：

```
cosmo
rad,O,1.72
rad,H,1.3
```

输出文件：

COSMO输出文件会在每次SCF计算的收敛后写出。片段电荷和势用远离中心的电荷校正进行修正。对于总电荷和能量，会给出校正的和非校正的值。正常输出文件仅包含非校正的值。推荐使用输出文件中的校正的值。

优化：

推荐使用专门对梯度进行操作的优化程序。调用能量的线搜索技术很可能会失败，因为在结构步骤后重新组织片段，可能会发生能量不连续。由于同样的原因，也不推荐数值梯度。

## 48.1 基本理论

COSMO是连续溶剂化模型，其中，溶剂用介电常数为 $\epsilon$ 的连续电解质表示。溶质分子位于连续电解质内的空穴中。由溶质的电荷分布造成的连续电解质的响应，用在空穴表面产生的屏蔽电荷描述。这一电荷分布可以通过求解导体表面为零的静电势边界方程来计算。在把空穴表面离散化成足够小的片段之后，表面片段上的屏蔽电荷的矢量为

$$\mathbf{q}^* = -\mathbf{A}^{-1}\Phi$$

其中的 $\Phi$ 是溶质电荷在片段上分布产生的势的矢量， $\mathbf{A}$ 是片段上屏蔽电荷的相互作用

矩阵。这种溶液对电导体非常精确。对于有限电解质，真正的电解质屏蔽电荷通过用 $f(\epsilon)$ 换算导体的电荷密度，也可以近似得非常好：

$$\mathbf{q} = f(\epsilon)\mathbf{q}^*; \quad f(\epsilon) = (\epsilon - 1)/(\epsilon + 0.5)$$

在每个SCF步骤中，屏蔽电荷 $\mathbf{q}$ 必须从势 $\Phi$ 产生，接下来作为外部点电荷添加到哈密顿量中。体系的总能量为

$$E_{tot} = E_0 + E_{diel}; \quad E_{diel} = \frac{1}{2}\Phi\mathbf{q}$$

其中 $E_0$ 是裸体系的自能， $E_{diel}$ 是介电能量。

空穴结构：

首先，构造半径为 $R_i + rsolv$ 互不包含的球面，其中的 $R_i$ 是原子半径，定义为元素特定的半径； $rsolv$ 是代表溶剂分子表面典型最大曲率的半径。不要把 $rsolv$ 误解为净溶剂半径，也不要对不同溶剂进行修改。每个原子球用基底层格点表示，每个原子有`nppa`个点。与不同原子球相交的基本格点被忽略。在第二步，余下的基底层格点投影到用半径 $R_i$ 定义的表面。空穴结构的第三步，残余的基底层格点被收集到片段中，它是在数值求解中常数屏蔽电荷的区域。最后，原子间的交叉部分用附加片段进行填充。

现在可以设定A矩阵。对于近距离和中等距离的片段，矩阵元将从片段的基底层格点计算（用`disex`控制），或者对大距离的片段使用片段中心计算。

外部电荷校正：

空穴外的非零电荷密度导致的误差可以用外部电荷校正进行修正。这一修正使用所谓的外部曲面（定义为半径 $R_i + rsolv \times routf$ ）上的势来估计屏蔽电荷和能量的校正项（A. Klamt and V. Jonas, J. Chem. Phys., 105, 9972-9981(1996)）。在SCF计算收敛后，执行一次修正。所有的修正值可以在COSMO输出文件找到。





MOLPRO程序包可以与其它软件一起使用，执行量子力学/分子力学杂化计算(QM/MM)。通过使用点电荷，静电嵌入可用于能量和梯度运行。特别是，可以在一个外部文件中包含点电荷格子，可以计算关于电荷位置的梯度，说明见10.5节。关于QM核位置的梯度（并且包含MM电荷的影响），可以像平常一样用FORCE命令计算（见第四十章）。

虽然MOLPRO本身不提供任何到力场程序的接口，但是其它的商业和非商业软件提供了这种组合。下面是一些可以使用MOLPRO的QM/MM软件。

### 49.1 Chemshell

CHEMSHELL计算化学环境 (<http://www.chemshell.org>) 提供到很多著名力场程序 (CHARMM, GROMOS, GULP, ...) 的接口。程序支持数种结构优化算法；分子动力学对NVE, NVT, 和NPT系综驱动；Monte Carlo；以及许多其它的工具。

CHEMSHELL的使用手册可以在以下的网址找到：

<http://www.cse.scitech.ac.uk/ccg/software/chemshell/manual/>

其中有使用MOLPRO的说明。

关于CHEMSHELL环境，还发布了一个免费的图形用户界面 (GUI)。CCP1GUI帮助产生杂化计算的输入，可以显示分子结构，并包含分子编辑工具，参见 <http://www.cse.scitech.ac.uk/ccg/software/ccp1gui/>



轨道可以用MERGE工具进行操作。例如，通过合并和正交化不同的轨道集，或者执行各个轨道间的 $2 \times 2$ 转动，可以从原子轨道构建分子轨道。其它的轨道操作可以用LOCALI程序（参见十八章）或MATROP程序（参见五十一章）执行。

调用MERGE程序使用

MERGE [*namout.file*]

以下各节讲述的所有子命令均可以缩写为三个字符。*namout.file*定义输出数据集（参见SAVE命令）。如果忽略*namout.file*，并且没有出现SAVE卡，则不保存新的轨道。所有的输出轨道必须由ORBITAL和ADD，MOVE，EXTRA，或PROJECT指令产生后才可以保存。

## 50.1 定义输入的轨道（ORBITAL）

ORBITAL,*namin.file,specifications*

从转存记录读取输入轨道集。*specifications*可用于选择特定的轨道集，说明参见4.11节。这些轨道的子集可以用ADD，MOVE，或EXTRA 命令添加到输出集中。

## 50.2 把轨道移动到输出集（MOVE）

MOVE,*orb1.sym1,orb2.sym2,orb3.sym3,ioff,fac,istart,iend*

把输入集里从*orb1.sym1*到*orb2.sym2*的轨道移至输出集中对称性*sym3* 的第*orb3*个矢量<sup>1</sup>，这必须未定义。输出集前*orb3-1*个矢量将被跳过，不管其在前面是否定义。如果*sym2* > *sym1*，那么*sym3*将包括从*sym1* 到*sym2*，*sym3*的输入不起作用。

<sup>1</sup>原文为第一个矢量。——译注。

如果 $orb1.sym1$ 为负, 那么 $abs(orb1)$ 将作为最大的移动轨道数, 从轨道 $1.sym1$ 开始, 一直到 $orb2.sym2$ 。如果 $orb2.sym2$ 为负, 那么 $abs(orb2)$ 将作为最大的移动矢量数, 从 $orb1.isym1$ 开始, 一直到对称性 $sym2$ 的最后一个轨道。

输入集里那些已经移动到或添加到输出集的轨道一般都被跳过。如果 $orb1$ 和 $orb2$ 为零, 则把整个输入集移动到输出集里。在这种情况下输入和输出的大小必须相同。如果 $orb1$ 非零但 $orb2$ 为零, 那么 $orb2$ 将被设置为对称性 $sym2$ 的最后一个轨道。如果 $sym2=0$ , 则 $sym2$ 设置为 $sym1$ 。 $ioff$ 是在输出矢量中, 相对于OFFSET指令设置的全局偏移的偏移量。 $fac$ 对移动没有影响。输入矢量中从 $istart$ 到 $iend$ 的元素将被移动。如果 $istart=0$ 且 $iend=0$ , 则移动整个输入矢量。

通过看后面的例子, MOVE指令的用法更容易理解。此外参见ADD和EXTRA 命令。

### 50.3 把轨道添加到输出集 (ADD)

ADD, $orb1.sym1,orb2.sym2,orb3.sym3,ioff,fac,istart,iend$

把从 $orb1.sym1$ 到 $orb2.sym2$ 的轨道添加到从 $orb3.sym3$ 开始的输出矢量中。输入矢量由因子 $fac$ 进行换算。如果 $fac=0$ , 则 $fac$ 设置为1.0。其它的细节参见MOVE命令。但是注意, 类似于MOVE, 已经定义的输出矢量不能被跳过。

此外参见MOVE和EXTRA命令。

### 50.4 定义额外的对称性 (EXTRA)

EXTRA, $exsym,orb1.sym1,orb2.sym2,orb3.sym3,ioff,fac,istart,iend$

除了考虑具有额外对称性 $exsym$ 的输入矢量以外, 用法完全同MOVE。如果 $orb1.sym1$ 和 $orb2.sym2$ 是零, 那么所有的输入矢量都移动输出集里, 并按照额外对称性增加的顺序排列。

例子:

EXTRA,1,-4.1	在对称性1中, 移动具有额外对称性1的接下来四个轨道。以前移动的轨道被跳过。
EXTRA,2,1.1	在对称性1中, 移动具有额外对称性2的所有轨道。以前移动的轨道被跳过。
EXTRA	将移动(所有对称性的)所有轨道, 并按照额外对称性排序。
EXTRA,3,1.1,0.8	在所有对称性中, 移动具有额外对称性3的所有轨道。以前移动的轨道被跳过。

此外参见ADD和MOVE命令。

## 50.5 定义输出集的偏移量 (OFFSET)

OFFSET,*iof*<sub>1</sub>,*iof*<sub>2</sub>,...,*iof*<sub>8</sub>;

为对称性1到8设置输出矢量的偏移。在后面的MOVE或ADD命令中，输入矢量移动到输出矢量*iof*<sub>*i*</sub>+1的位置。各个ADD或MOVE命令的偏移可以用这些卡中的参数*ioff*修改。这个卡应当立即接在所用的轨道指令后面。这个卡一般只在输入和输出矢量具有不同大小时才需要。

如果输入轨道集的大小小于当前基的大小，那么自动按照以下方法确定偏移：每次读入一个轨道集，则前面的输入轨道大小被加入到偏移中。因此，只有当轨道集按照正确的顺序给出，并且当各个维度大小总计达到目前的总维度时，才能正常工作。如果不是这样，那么偏移应当在接到轨道指令之后的OFFSET卡中指定。

## 50.6 投影轨道 (PROJECT)

PROJECT,*namin.file*

这个命令将从记录*namin.file*读入矢量。这些矢量必须与当前计算的矢量有同样的大小。截止到当前，用ORBITAL, MOVE, 和ADD指令定义的所有轨道将从输入集投影掉。接下来，投影后的轨道进行对角化，并移动到未定义的输出矢量中。这总是产生完整的矢量集。

## 50.7 对称正交归一化 (ORTH)

ORTH,*n*<sub>1</sub>,*n*<sub>2</sub>,...,*n*<sub>8</sub>

对每个对称性*i*的前*n*<sub>*i*</sub>个矢量进行对称正交归一化。这些矢量必须在前面用ORBITAL和MOVE或ADD指令提供。

## 50.8 Schmidt正交归一化 (SCHMIDT)

SCHMIDT,*n*<sub>1</sub>,*n*<sub>2</sub>,...,*n*<sub>8</sub>

对每个对称性*i*的前*n*<sub>*i*</sub>个矢量进行Schmidt正交归一化。这些矢量必须在前面用ORBITAL 和MOVE或ADD指令提供。

## 50.9 转动轨道 (ROTATE)

ROTATE,*iorb1.sym,iorb2,angle*

将对对称性*sym*中的轨道*iorb1*和*iorb2*执行指定角度*angle*（单位：度）的2×2转动。*angle*=0表示交换轨道（等价于*angle*=90）。这些矢量必须在前面用ORBITAL 和MOVE或ADD指令提供。

## 50.10 初始化新的输出集 (INIT)

INIT,*namout.file*

将初始化一个新的输出集。输出集里所有以前的矢量会全部丢失，除非是已经用SAVE 指令保存！

## 50.11 保存合并的轨道

SAVE,*namout.file*

把当前的输出集保存到记录*namout.file*中。当前的输出集必须是完整的，在保存前将进行Schmidt正交归一化。如果没有给出SAVE指令，那么在所有的有效命令对MERGE卡指定的记录进行处理之后，再保存输出矢量。

## 50.12 打印选项 (PRINT)

PRINT,*iprint,ideb*

指定打印选项。

- iprint* = 0      不打印。
- iprint* ≥ 1      打印ORTH卡指定的正交归一化轨道。
- iprint* ≥ 2      还打印在这个正交归一化之前的轨道。
- iprint* ≥ 3      打印所有最终矢量。
- ideb* ≠ 0      打印各个阶段的重叠矩阵。

## 50.13 例子

### 50.13.1 H<sub>2</sub>F

这个例子合并H<sub>2</sub>和F的轨道。

```
! $Revision: 2009.1 $
***,example for merge
print,orbitals,basis
rh2=1.4
rhf=300.
basis=vdz
symmetry,x,y           ! 使用C2v对称性
geometry={F}
text,F
```

```
{rhf;wf,9,1,1;occ,3,1,1;orbital,2130.2} ! F原子的rhf

text,H2
symmetry,x,y ! 使用C2v对称性
geometry={
    H1,
    H2,H1,rh2}

{hf;orbital,2100.2} ! H2的scf
{multi;occ,2;orbital,2101.2} ! H2的mcscf

text,FH2
geometry={F; ! F+H2的线型结构
    H1,F,rhf
    H2,H1,rh2,F,180}

{merge
orbital,2130.2 ! F原子的rhf轨道
move,1.1,2.1,1.1 ! 移动轨道1.1, 2.1
move,3.1,0.4,4.1; ! 从4.1开始, 移动所有余下的轨道
orbital,2100.2 ! H2的hf轨道
move,1.1,0.4 ! 把这些轨道移到未用过的位置
save,2131.2} ! 保存合并的轨道

{rhf;occ,4,1,1;start,2131.2 ! F+H2的rhf
orbital,2132.2}

{merge
orbital,2130.2 ! F原子的rhf轨道
move,1.1,2.1,1.1 ! 移动轨道1.1, 2.1
move,3.1,3.1,4.1; ! 移动3.1到4.1的轨道
move,4.1,0.4,6.1 ! 从6.1开始, 移动所有余下的轨道
orbital,2101.2 ! H2的mcscf轨道
move,1.1,0.4 ! 把这些轨道移到未用过的位置
save,2141.2} ! 保存合并的轨道

{multi;occ,5,1,1;start,2141.2} ! 对F+H2使用价空间的casscf
```

### 50.13.2 NO

这个例子合并N和O的SCF轨道，以获得NO完整的价空间。在最简单的情况中，用各自的基组进行原子计算，但是使用与分子计算相同的对称性 ( $C_{2v}$ )。

```
! $Revision: 2009.1 $
***,NO merge
r=2.1

symmetry,x,y
geometry={n}          ! N原子, c2v对称性

{rhf;occ,3,1,1;        ! N原子的rhf
wf,7,4,3;              ! 4S态
orbital,2110.2}        ! 把轨道保存到文件2的记录2110中

symmetry,x,y
geometry={o}

{rhf;occ,3,1,1;        ! O原子的rhf
wf,8,4,2              ! 3P态
orbital,2120.2}        ! 把轨道保存到文件2的记录2120中

geometry={n;o,n,r}     ! NO分子, c2v对称性

{MERGE
ORBITAL,2110.2         ! 读取N原子的轨道
MOVE,1.1,1.1          ! 把1s轨道移动到输出矢量1.1
MOVE,2.1,2.1,3.1      ! 把2s轨道移动到输出矢量3.1
MOVE,3.1,3.1,5.1      ! 把2pz轨道移动到输出矢量5.1
MOVE,1.2,1.2          ! 把2px轨道移动到输出矢量1.2
MOVE,1.3,1.3          ! 把2py轨道移动到输出矢量1.3
MOVE,4.1,,7.1         ! 移动对称性1的虚轨道
MOVE,2.2,,3.2         ! 移动对称性2的虚轨道
MOVE,2.3,,3.3         ! 移动对称性3的虚轨道
MOVE,1.4              ! 移动对称性4的虚轨道
ORBITAL,2120.2        ! 读取O原子的轨道
MOVE,1.1,0.4          ! 移动所有O原子的轨道
ROT,3.1,4.1,45;       ! 转动2s轨道, 使之成为成键和反键的线性组合
ROT,5.1,6.1,-45;     ! 转动2pz轨道, 使之成为成键和反键的线性组合
PRINT,1              ! 设定打印选项
ORTH,6,2,2           ! 用对称性正交归一化价轨道, 并打印产生的轨道
save,2150.2}          ! 把合并的轨道保存到记录2150.2中

{multi;occ,6,2,2      ! 对NO执行全价电子的casscf
```



```

wf,15,2,1      ! 2Pix态
wf,15,3,1      ! 2Piy态
start,2150.2}  ! 从合并轨道开始

```

[http://www.molpro.net/info/current/examples/no\\_merge1.com](http://www.molpro.net/info/current/examples/no_merge1.com)

还可以在全部基组中使用dummy卡进行原子计算。这种情况下流程更加复杂，因为两个轨道空间的结合过于完备。计算如下：

- a) 对整个分子做SCF，轨道保存到2100.2
- b) O使用虚原子基，对N原子做SCF，轨道保存到2110.2
- c) N使用虚原子基，对O原子做SCF，轨道保存到2120.2
- d) 合并原子的SCF轨道。最后，通过从SCF轨道中投影掉合并轨道，获得NO的虚轨道。

```

! $Revision: 2006.0 $
***,NO merge
geometry={n;o,n,r}
r=2.1

{rhf;occ,5,2,1      ! NO的rhf
wf,15,2,1          ! 2Pi态
orbital,2100.2}    ! 把轨道保存到文件2的记录2100中

dummy,o           ! O为虚原子
{rhf;occ,3,1,1;     ! N原子的rhf
wf,7,4,3;          ! 4S态
orbital,2110.2}    ! 把轨道保存到文件2的记录2110中

dummy,n           ! N为虚原子
{rhf;occ,3,1,1;     ! O原子的rhf
wf,8,4,2           ! 3P态
orbital,2120.2}    ! 把轨道保存到文件2的记录2120中

{MERGE            ! 调用合并程序

ORBITAL,2110.2     ! 读取N原子的轨道
MOVE,1.1,1.1       ! 把输入矢量1.1移动到输出矢量1.1
MOVE,2.1,3.1,3.1   ! 把输入矢量2.1,3.1移动到输出矢量3.1和4.1
MOVE,1.2,1.2       ! 把输入矢量1.2移动到输出矢量1.2
MOVE,1.3,1.3       ! 把输入矢量1.3移动到输出矢量1.3
ORBITAL,2120.2     ! 读取O原子的轨道
MOVE,1.1,3.1       ! 把输入矢量1.1到3.1移动到输出矢量2.1, 5.1, 6.1
MOVE,1.2,1.2       ! 把输入矢量1.2移动到输出矢量2.2
MOVE,1.3,1.3       ! 把输入矢量1.3移动到输出矢量2.3

```

```
ROT,3.1,5.1,45;      ! 转动2s轨道，使之为成键和反键的线性组合
ROT,4.1,6.1,-45;     ! 转动2pz轨道，使之为成键和反键的线性组合
PRINT,1              ! 设定打印选项
ORTH,6,2,2           ! 用对称性正交归一化价轨道，并打印产生的轨道
PROJ,2100.2          ! 从SCF分子轨道中投影掉价轨道，并加入虚轨道集
SAVE,2150.2          ! 把合并的轨道保存到文件2的2150记录中
}

dummy                ! 删除虚原子
{multi;occ,6,2,2      ! 对NO执行全价空间的casscf
wf,15,2,1             ! 2Pi态
wf,15,3,1             ! 2Pi态
start,2150.2}         ! 从合并轨道开始
```

[http://www.molpro.net/info/current/examples/no\\_merge2.com](http://www.molpro.net/info/current/examples/no_merge2.com)

## 第五十一章

## 矩阵操作

**MATROP;**

**MATROP**可以对维度为单粒子基组的矩阵进行简单的矩阵操作。要进行这样的操作，首先用**LOAD**命令把所需要的矩阵加载到内存中。对每个矩阵，都分配一个内部的名称（一个用户定义的任意字符串），以后可以在其它的命令中引用。矩阵操作之后，产生的矩阵可以用**SAVE**指令保存到转存记录中。对于数值，例如矩阵的迹，或者每个矩阵元，可以保存在变量中。

*code*可以是以下之一：

<b>LOAD</b>	从文件加载矩阵。
<b>SAVE</b>	把矩阵保存到文件中。
<b>ADD</b>	矩阵相加。
<b>TRACE</b>	构成矩阵或两个矩阵乘积的迹。
<b>MULT</b>	两个矩阵相乘。
<b>TRAN</b>	变换矩阵。
<b>DMO</b>	把密度转换到MO基。
<b>NATORB</b>	计算自然轨道。
<b>DIAG</b>	矩阵对角化。
<b>OPRD</b>	构成两个矢量的外积。
<b>DENS</b>	构成闭壳层密度矩阵。
<b>FOCK</b>	计算闭壳层的Fock矩阵。
<b>COUL</b>	计算Coulomb算符。
<b>EXCH</b>	计算交换算符。

PRINT	打印矩阵。
PRID	打印矩阵的对角元。
PRI0	打印轨道。
ELEM	把矩阵元赋值给变量。
READ	从输入中读取方阵。
WRITE	把方阵写到文件。
SET	给一个变量赋值。
ADDVEC	把一个矩阵某列的倍数加到另一个矩阵的某列上。

注意，在UNIX计算机上，出现在以上命令中的文件名会转化为小写。  
具体解释参见以下部分。

## 51.1 调用矩阵工具 (MATROP)

这个程序用输入卡MATROP调用，不需要其它的任何说明。

### MATROP

以下命令可以按照任何顺序接在后面，最多可以操作50个矩阵。每一个命令行的第一项是命令关键词，之后是结果矩阵的名称。如果已经存在了结果矩阵`result`，它将会被覆盖，否则会创建一个新的矩阵。操作中需要的所有矩阵都必须事先加载或者定义，除非另外声明。

如果矩阵名称后面加上了反引号 (‘)，那么对矩阵进行转置操作。

## 51.2 加载矩阵 (LOAD)

所有在以后命令中用到的矩阵必须首先用LOAD命令加载到内存中。根据矩阵的类型，LOAD命令可以有略微不同的选项。对于所有形式的LOAD，`name`都是任意的字符串（最多可以有16个字符串），以后的命令可以用它来引用所加载的矩阵。

### 51.2.1 加载轨道

LOAD,`name`,ORB [`record`] [`specifications`]

从给定的转存记录中加载轨道因子矩阵。如果没有指定记录，则使用最后一个转存记录。特定的轨道集可以用可选的`specifications`选择，详细的说明参见4.11节。如果`name=ORB`，那么关键词ORB不需要给出。

### 51.2.2 加载密度矩阵

LOAD,*name*,DEN [,*record*] [,*specifications*]

从给定的转存记录中加载密度矩阵。如果没有指定记录，则使用最后一个转存记录。特定的轨道集可以用可选的*specifications*选择，详细的说明参见4.11节。如果*name*=DEN，那么关键词DEN不需要给出。

### 51.2.3 加载AO重叠矩阵S

LOAD,*name*,S

加载AO基中的重叠矩阵。如果*name*=S，那么关键词S不需要给出。

### 51.2.4 加载 $S^{-1/2}$

LOAD,*name*,SMH

加载 $S^{-1/2}$ ，其中S是AO基中的重叠矩阵。如果*name*=SMH，那么关键词SMH不需要给出。

### 51.2.5 加载单电子哈密顿量

LOAD,*name*,H0

LOAD,*name*,H01

加载AO基中的单电子哈密顿量。H01不同于H0之处在于加入了微扰，如果有的话（参见33.5.1节和33.5.2节）。如果*name*=H0（H01），不需要给出H0（H01）关键词。H0或H01的核能在内部存储。

### 51.2.6 加载动能或势能算符

LOAD,*name*,EKIN

LOAD,*name*,EPOT

加载AO基中单电子哈密顿量的各个部分。EPOT对所有的原子求和。核能与EPOT有关，在内部存储。如果*name*=EKIN（EPOT），不需要给出EKIN（EPOT）关键词。

### 51.2.7 加载单电子特性算符

LOAD,*name*,OPER,*opname*,[*isym*],*x*,*y*,*z*

加载单电子算符*opname*，其中*opname*是指定算符的关键词（必须给出一个分量）。有效的关键词参见6.13节。*isym*是算符的总对称性（默认是1），*x*,*y*,*z*是算符的原点。如果在算符记录中不能使用该算符，程序会自动计算。核值在程序内部关连到*name*，并存

储于变量OPNUC中（这个变量对每个加载的算符都会被覆盖，但是可以用SET命令复制到其它的变量中）。注意，偶极和四极算符的电子部分必须乘以-1。

### 51.2.8 从无格式记录加载矩阵

LOAD,*name*,TRIANG,*record*,[*isym*]

LOAD,*name*,SQUARE,*record*,[*isym*]

从无格式记录（不是转存记录或算符记录）加载三角阵或方阵。如果没有给定*isym*，那么假定是1。

## 51.3 保存矩阵（SAVE）

SAVE,*name*,*record*[,*type*]

目前，*type*可以是DENSITY, ORBITALS, FOCK, HO, ORBEN, OPER, TRIANG, SQUARE, 或VECTOR。如果*type*没有给出，但对LOAD或其它命令已知，则假定来自它们。轨道，密度矩阵，Fock矩阵，以及轨道能量保存在转存记录中（通常，所有的这些量都使用同一个转存记录）。如果*type*是HO，那么用当前的矩阵覆盖单电子哈密顿量，核能依照与*name*有关的值进行修改。核能还存储在变量ENUC中。其它的所有矩阵也可以用TRIANG和SQUARE选项从无格式记录分别存储为三角阵或方阵的形式（对于三角存储，矩阵在存储前进行对称化处理）。本征矢可以用VECTOR 选项保存到无格式记录中。在每个无格式记录中只能保存一个矩阵或矢量。

单电子算符可以存储在算符记录中，使用命令

SAVE,*name*,OPER, [PARITY=*np*], [NUC=*opnuc*], [CENTRE=*icen*],[COORD=[*x,y,z*]]

用户定义的算符*name*也可以用于后面的EXPEC或GEXPEC卡。*np* = 1, 0, -1 分别用于对称算符、平方算符、反对称算符（默认是1）。如果定义了CENTRE，那么假定算符在给定的中心上具有原点，其中*icen*表示Z矩阵输入的行号。也可以直接用COORD 定义坐标。默认假定是最后一次读入算符的坐标，否则为零。

如果产生了NATURAL轨道并保存到转存记录中，那么同时也自动存储占据数。这便于以后的使用，例如用于MOLDEN。

## 51.4 矩阵相加（ADD）

ADD,*result*[,*fac1*],*mat1*[,*fac2*],*mat2*,...

计算  $result = fac1 \cdot mat1 + fac2 \cdot mat2 + \dots$

字符串*result*, *mat1*, *mat2*指定矩阵的内部名称。*mat1*和*mat2* 必须存在，否则会产生错误。如果不存在*result*，则创建一个。

因子 $fac1$ 和 $fac2$ 是可选的（可以是变量）。如果没有给出，假定是1。

与各个矩阵相应的核值会相应地相加，结果关连到 $result$ 。

### 51.5 矩阵或两个矩阵乘积的迹（TRACE）

TRACE,  $variable$ ,  $mat1$ , [ $factor$ ]

计算 $variable = factor * tr(mat1)$ 。

TRACE,  $variable$ ,  $mat1$ ,  $mat2$ , [ $factor$ ]

计算 $variable = factor * trace(mat1 \cdot mat2)$ 。

迹运算的结果存储在MOLPRO的变量 $variable$ 中，用于以后的操作。

如果没有给出 $factor$ ，假定是1。

### 51.6 设置变量（SET）

SET,  $variable$ ,  $value$

把 $value$ 赋值给MOLPRO的变量 $variable$ ，其中的 $value$ 可以是包含任意个变量或数值的表达式。但是还不能用 $variable$ 的索引指数。

### 51.7 矩阵相乘（MULT）

MULT,  $result$ ,  $mat1$ ,  $mat2$ , [ $fac1$ ], [ $fac2$ ]

计算 $result = fac2 * result + fac1 * mat1 \cdot mat2$

字符串 $result$ ,  $mat1$ ,  $mat2$ 是矩阵的内部名称。如果没有给出 $fac1$ ，那么假定 $fac1=1$ 。如果没有给出 $fac2$ ，假定 $fac2=0$ 。如果 $mat1$ 或 $mat2$ 后面加上了反引号（```），那么相应的矩阵在操作前进行转置。如果 $result$ 加上了反引号，那么对结果的矩阵进行转置。

### 51.8 变换操作（TRAN）

TRAN,  $result$ ,  $Op$ ,  $C$

计算 $result = C(T) * Op * C$ 。字符串 $result$ ,  $C$ , 和 $Op$ 是矩阵的内部名称。如果 $C$ 或 $Op$ 后面加上了反引号（```），那么相应的矩阵在操作前进行转置。因此，

TRAN,  $result$ ,  $Op$ ,  $C'$

将计算 $result = C * Op * C(T)$ 。

### 51.9 把密度矩阵转换到MO基（DMO）

DMO,  $result$ ,  $D$ ,  $C$

计算 $result = C(T) * S * D * S * C$ 。字符串 $result$ ,  $C$ , 和 $D$ 是内部名称。

### 51.10 矩阵对角化 (DIAG)

DIAG, *eigvec*, *eigval*, *matrix* [, *iprint*]

对角化 *matrix*。内部存储本征矢和本征值，相应的名称分别为 *eigvec* 和 *eigval*（直到16个字符的任意字符串）。如果 *iprint.gt.0*，那么打印本征值。如果 *iprint.gt.1*，那么也打印本征矢。

### 51.11 产生自然轨道 (NATORB)

NATORB, *name*, *dens*, *thresh*

对密度矩阵 *dens* 计算自然轨道。打印占据数大于或等于 *thresh* 的轨道（默认1.d-4）。

### 51.12 构成两个矢量的外积 (OPRD)

OPRD, *result*, *matrix*, *orb1*, *orb2*, *factor*

从矩阵 *matrix* 获取列向量 *v1* 和 *v2*，并把它们的外积加入到 *result* 中。*v1* 和 *v2* 必须按照 *icol.isym* 的形式给出，例如，3.2 表示对称性2的第三个矢量。结果为：

$$result(a, b) = result(a, b) + factor * v1(a) * v2(b)$$

如果此前没有使用过 *result*，那么它在进行操作前是零。

### 51.13 组合矩阵列 (ADDVEC)

ADDVEC, *result*, *orbr*, *source*, *orbs*, *factor*

从 *source* 获取列矢量 *orbs*，并把它加到 *result* 的 *orbr* 列上。*v1* 和 *v2* 必须按照 *icol.isym* 的形式给出，例如，3.2 表示对称性2 的第三个矢量。

### 51.14 构成闭壳层密度矩阵 (DENS)

DENS, *density*, *orbitals*, *iocc<sub>1</sub>*, *iocc<sub>2</sub>* ...

从给定的轨道 *orbitals* 构成闭壳层密度矩阵 *density*。每个不可约表示 *i* 的占据轨道数必须在 *iocc<sub>i</sub>* 中给出。

### 51.15 计算Fock矩阵 (FOCK)

FOCK, *f*, *d*

用密度 *d* 计算闭壳层Fock矩阵。结果保存在 *f* 中。



### 51.16 计算库仑算符 (COUL)

COUL,  $J, d$

用密度  $d$  计算库仑算符  $J(d)$ 。

### 51.17 计算交换算符 (EXCH)

EXCH,  $K, d$

用密度  $d$  计算交换算符  $K(d)$ 。

### 51.18 打印矩阵 (PRINT)

PRINT, *name*, [*ncol*(1), *ncol*(2), ...]

打印矩阵 *name*。其中 *ncol*(*isym*) 是对行称性 *isym* 要打印的列数（如果未给出，那么打印所有的列）。打印轨道也可以使用 ORB。

### 51.19 打印矩阵的对角元 (PRID)

PRID, *name*

打印矩阵 *name* 的对角元。

### 51.20 打印轨道 (PRIO)

PRIO, *name*,  $n_1, n_2, n_3, \dots, n_8$

打印轨道 *name*。在对称性  $i$  中，打印前  $n_i$  个轨道。如果  $n_i = 0$ ，那么打印该对称性的所有轨道。

### 51.21 把矩阵元赋值给变量 (ELEM)

ELEM, *name*, *matrix*, *col*, *row*

把矩阵 *matrix* 的元素 (*col*, *row*) 赋值给变量 *name*。*col* 和 *row* 必须以 *number.isym* 的形式给出，其中 *number* 是对称性 *isym* 中的行数或列数。行和列对称性的乘积必须与矩阵的对称性一致。

### 51.22 从输入文件读取矩阵 (READ)

READ, *name*, [[TYPE=] *type*], [[SUBTYPE=] *subtype*], [[SYM=] *symmetry*], [FILE= *file*]  
{ *values* }

从输入或ASCII文件读取（对称性1的）方阵。*values*可以是自由格式，但它们的总数必须正确。由“#”，“\*”，或“'”开始的注释行将被跳过。如果在输入中给出数据，那么数据区必须要么用花括号括起来，要么首行是BEGIN\_DATA末行是END\_DATA。如果*filename*指定为选项，那么从该文件读取数据。在这种情况下，文件中的BEGIN\_DATA和END\_DATA行是可选的，不必接数据区。

为了与旧版本兼容，可以在输入文件里用INCLUDE命令包含数据（参见3.1节）。在这种情况下，被包含的文件中必须有BEGIN\_DATA和END\_DATA行（如果用MATROP,WRITE指令写文件，这是自动的）。

*type*是一个可以用来指定矩阵类型的字符串。如果合适，它可以是LOAD命令中所用的任何一个。此外，如果需要的话，也可以定义SUBTYPE。它用来描述例如轨道类型或密度矩阵（例如，对于自然轨道是TYPE=ORB和SUBTYPE=NATURAL）。只有当不是1时，才需要给定矩阵的对称性。

### 51.23 把矩阵写入ASCII文件（WRITE）

WRITE,*name*,[*filename* [*status*]]

把矩阵写到一个ASCII文件中。如果*filename*没有给出，那么矩阵写到输出文件，否则写到指定的文件（*filename*将转换成小写）。如果*filename*=PUNCH，那么矩阵将写到当前的punch文件中。

如果*status*=NEW, ERASE, 或REWIND, 将写到一个新的文件中，否则添加到已存在的文件中。

### 51.24 示例

下面的例子显示MATROP命令的各种使用。

```
! $Revision: 2006.0 $
***,h2o matrop examples
geometry={o;h1,o,r;h2,o,r,h1,theta} ! Z矩阵结构输入
r=1 ang ! 键长
theta=104 ! 键角
hf ! 进行scf计算
{multi
natorb
canonical}
{matrop
load,D_ao,DEN,2140.2 ! 载入mcscf密度矩阵
load,Cnat,ORB,2140.2,natural ! 载入mcscf自然轨道
load,Ccan,ORB,2140.2,canonical ! 载入mcscf正则轨道
```

```

load,Dscf,DEN,2100.2      ! 载入scf密度矩阵
load,S                    ! 载入重叠矩阵

prio,Cnat,4,1,2           ! 打印占据casscf轨道

elem,d11,Dscf,1.1,1.1    ! 打印元素D(1,1)
elem,d21,Dscf,2.1,1.1    ! 打印元素D(2,1)
elem,d12,Dscf,1.1,2.1    ! 打印元素D(1,2)

tran,S_mo,s,Cnat         ! 转换到MO基(同上)
print,S_mo               ! 打印结果---应当是单位矩阵

trace,Nao,S_mo           ! S_MO的迹=基函数个数
trace,Nel,D_ao,S        ! 构成迹(DS)=电子个数

mult,SC,S,Cnat           ! 构成SC=S*Cnat
tran,D_nat,D_ao,SC       ! 把密度转换到自然MO(也可以用dmo完成)

prid,D_nat              ! 打印对角元(占据数)

dmo,D_can,D_ao,Ccan      ! 把D_ao转换到正则MO基。同上面的简化
add,D_neg,-1,D_can       ! d_can乘以-1
diag,U,EIG,D_neg         ! 对角化密度D_can
mult,Cnat1,Ccan,U        ! 把正则轨道转换到自然轨道
prio,Cnat1,4,1,2        ! 打印新的自然轨道

natorb,Cnat2,D_ao        ! 自然轨道直接使用MCSCF密度D_ao
prio,Cnat2,4,1,2        ! 打印新的自然轨道(应当与上面的结果相同)

add,diffden,D_ao,-1,Dscf ! 构成mcscf-scf差分密度
natorb,C_diff,diffden    ! 对差分密度产生自然轨道

write,diffden,denfile     ! 把差分密度写入ASCII文件denfile
save,C_diff,2500.2       ! 对差分密度把自然轨道存储到转存记录2500.2
}

```

<http://www.molpro.net/info/current/examples/matrop.com>

第二个例子给H<sub>2</sub>O加入四极场。结果和使用QUAD指令完全相同。H<sub>2</sub>O用修改过的单电子矩阵覆盖，核能自动变为合适的值。之后的SCF计算使用修改的单电子算符。

注意，通常推荐用DIP，QUAD，或FIELD指令添加外场。

```

! $Revision: 2006.0 $
memory,2,m
R      =      0.96488518 ANG

```

```

THETA= 101.90140469
geometry={H1
          0,H1,R;
          H2,0,R,H1,THETA}
{hf;wf,10,1}

field=0.05          ! 定义场强

{matrop
load,h0,h0          ! 载入单电子哈密顿量
load,xx,oper,xx     ! 载入二次矩
load,yy,oper,yy
load,zz,oper,zz
add,h01,h0,field,zz,-0.5*field,xx,-0.5*field,yy ! 把二次矩加入到h0, 并存入h01
save,h01,1210.1,h0} ! 保存h0
hf                  ! 用修改的h0进行scf计算

{matrop
load,h0,h0          ! 载入h0
load,qmzz,oper,qmzz ! 载入四极矩qmzz
add,h01,h0,field,qmzz ! 把四极矩加入到h0 (同上面二次矩的结果)
save,h01,1210.1,h0} ! 保存h0
hf                  ! 用修改的h0进行scf计算

quad,,,field        ! 添加四极场到h0
hf                  ! 用修改的h0进行scf计算 (同上面用matrop的结果)

field,zz,field,xx,-0.5*field,yy,-0.5*field ! (添加一般场, 结果同上)
hf                  ! 用修改的h0进行scf计算 (同上面用matrop的结果)

field,zz,field       ! 与前面用单独的field命令相同
field+,xx,-0.5*field
field+,yy,-0.5*field
hf                  ! 用修改的h0进行scf计算 (同上面用matrop的结果)

```

<http://www.molpro.net/info/current/examples/matropfield.com>

## 51.25 练习: SCF程序

用MATROP写一个闭壳层SCF程序用于H<sub>2</sub>O计算!

提示:

首先通过寻找 $h0$ 的本征矢产生初始轨道猜测。把轨道保存到记录中。在第一次调用到MATROP之前, 基组和结构用通常的方法定义。

接下来使用MOLPRO的DO循环，并在每次迭代都调用MATROP。把当前的能量保存到变量中（注意，核能存储在变量ENUC中）。此外，在每一次迭代计算偶极矩。在迭代的最后用IF命令对能量变化执行收敛测试。这必须在MATROP之外且ENDDO 之前完成。此时也可以把迭代次数，能量，和偶极矩存储到数组中，在收敛后用TABLE 指令打印。对于以下的结构和基组，

```
geometry={o;h1,o,r;h2,o,r,h1,theta} ! Z-matrix结构输入
r=1 ang ! 键长
theta=104 ! 键角
basis=vdz ! 基组
thresh=1.d-8 ! 收敛阈值
```

结果如下：

SCF has converged in 24 iterations

ITER	E	DIP
1.0	-68.92227207	2.17407361
2.0	-71.31376891	-5.06209922
3.0	-73.73536433	2.10199751
4.0	-74.64753557	-1.79658706
5.0	-75.41652680	1.43669203
6.0	-75.77903293	0.17616098
7.0	-75.93094231	1.05644998
8.0	-75.98812258	0.63401784
9.0	-76.00939154	0.91637513
10.0	-76.01708679	0.76319435
11.0	-76.01988143	0.86107911
12.0	-76.02088864	0.80513445
13.0	-76.02125263	0.83990621
14.0	-76.02138387	0.81956198
15.0	-76.02143124	0.83202128
16.0	-76.02144833	0.82464809
17.0	-76.02145450	0.82912805
18.0	-76.02145672	0.82646089
19.0	-76.02145752	0.82807428
20.0	-76.02145781	0.82711046
21.0	-76.02145792	0.82769196
22.0	-76.02145796	0.82734386
23.0	-76.02145797	0.82755355
24.0	-76.02145797	0.82742787

虽然收敛并不快，但是能够运行！



- [1] C. Adamo and V. Barone. *J. Chem. Phys*, 110(13):6158, 1999.
- [2] A. D. Becke. *J. Chem. Phys*, 84(8):4524, 1986.
- [3] A. D. Becke. *J. Chem. Phys*, 85(12):7184, 1986.
- [4] A. D. Becke. *Phys. Rev. A*, 38(6):3098, 1988.
- [5] A. D. Becke. *J. Chem. Phys.*, 104(3):1040, 1995.
- [6] A. D. Becke. *J. Chem. Phys*, 107(20):8554, 1997.
- [7] A. D. Becke. *J. Chem. Phys.*, 88(2):1053, 1998.
- [8] A. D. Boese, N. L. Doltsinis, N. C. Handy, and M. Sprick. *J. Chem. Phys*, 112(4):1670, 2000.
- [9] M. Ernzerhof and G. Scuseria. *J. Chem. Phys.*, 111(3):911, 1999.
- [10] P. M. W. Gill. *Mol. Phys.*, 89:433, 1996.
- [11] F. A. Hamprecht, A. J. Cohen, D. J. Tozer, and N. C. Handy. *J. Chem. Phys*, 109(15):6264, 1998.
- [12] F. R. Manby and P. J. Knowles. *J. Chem. Phys.*, 112:7002, 2000.
- [13] J. P. Perdew. *Phys. Rev. B*, 33:8822, 1986.
- [14] J. P. Perdew, K. Burke, and M. Ernzerhof. *Phys. Rev. Lett*, 77(18):3865, 1996.

- 
- [15] J. P. Perdew, J. A. Chevary, S. H. Vosko, K. A. Jackson, M. R. Pederson, and C. Fiolhais. *Phys. Rev. B*, 46(11):6671, 1992.
  - [16] J. P. Perdew and Y. Wang. *Phys. Rev. B*, 33(12):8800, 1986.
  - [17] J. P. Perdew and Y. Wang. *Phys. Rev. B*, 45(23):13244, 1992.
  - [18] J. C. Slater. *Phys. Rev.*, 81:385, 1951.
  - [19] P. A. Stewart and P. M. W. Gill. *J. Chem. Faraday Trans.*, 91(24):4337, 1995.
  - [20] R. Strange, F. R. Manby, and P. J. Knowles. *Comput. Phys. Commun.*, 136(3):310, 2001.
  - [21] D. J. Tozer and N. C. Handy.
  - [22] D. J. Tozer and N. C. Handy. *J. Chem. Phys.*, 108(6):2545, 1998.
  - [23] D. J. Tozer and N. C. Handy. *Mol. Phys.*, 94(4):707, 1998.
  - [24] D. J. Tozer, N. C. Handy, and W. H. Green. *Chem. Phys. Lett.*, 273:183, 1997.
  - [25] T. Van Voorhis and G. E. Scuseria. *J. Chem. Phys.*, 109:400, 1998.
  - [26] S. H. Vosko, L. Wilk, and M. Nusair. *Can. J. Phys.*, 58:1200, 1980.
  - [27] Y. Zhang and W. Yang. *Phys. Rev. Lett*, 80:890, 1998.



## 附录 A

# MOLPRO的安装

### A.1 获得发布程序

MOLPRO通过使用互联网，以自助的形式发布给许可持有人。被授权的代码可以从<http://www.molpro.net/download>获得，并需要提供用户名和口令。网页既包含源代码也包含二进制文件，虽然并不是所有人都有资格获得源代码，而二进制文件并不能用于每种平台。

不管是从二进制文件还是从源代码创建，执行MOLPRO都需要有效的许可密钥。注意，密钥包含两部分，也就是一系列由逗号隔开的一对关键字=值，和一个口令字符串，它们用“&”隔开。在大多数情况下，当创建或安装软件时，许可密钥自动从网站下载。

### A.2 安装预编译的二进制文件

二进制程序作为自解压tar包的形式给出，在命令行运行它们就可以安装。有不同架构的二进制包。它们还支持并行执行。有一个一般的串行二进制包，它适用于所有的IA32体系架构。

tar包可以完全重定位。在交互运行脚本时可以修改位置，默认为：`/usr/local`。

如果在MOLPRO\_KEY环境变量中设定了许可密钥，或者脚本找到了隐藏在`$HOME/.molpro/token`来自上一次安装的许可密钥，那么这个许可密钥会被软件安装。如果脚本找不到密钥，或者需要从MOLPRO网站自动下载密钥，那么脚本将会提示这一部分安装失败。这时MOLPRO的所有文件都被安装，但是用户必须人工安装许可密钥，使用名为`.token`的库文件，例如：`/usr/local/lib/molpro-mpptype-arch/.token`

其它配置选项的说明见A.3.5节，它们可以在脚本文件/usr/local/bin/molpro中指定。

## A.3 从源文件安装

### A.3.1 概述

从源文件安装MOLPRO通常有四个不同阶段：

配置	运行shell脚本，用于指定配置选项，并创建驱动后面安装步骤的配置文件。
编译	编译和链接程序，创建其它各种工具和文件，包括默认的选项文件。产生的基本组件有： <ol style="list-style-type: none"><li>1. molpro的shell脚本，它驱动可执行主程序。在串行情况下，用户可以直接使用可执行主程序。</li><li>2. molpro.exe可执行程序，它是主程序。对于并行计算，通过使用合适的系统工具（例如mpirun, parallel, 等），用一个molpro shell脚本的请求开始运行多个molpro.exe的拷贝。</li><li>3. 机器中现成的基组，以及其它工具、库。</li></ol>
验证	运行一套自校验测试任务，用来验证编译的代码能正确运行。
安装	程序可以直接从编译时的源代码树运行，但是通常推荐运行安装流程，它把基本组件安装到标准的系统目录中。

### A.3.2 预先的准备

从源代码安装，以下是必需的或者强烈推荐的。

1. Fortran 90编译器。仅有Fortran77编译器还不够。对于大多数的系统，应当使用商家提供的最新的编译器。

在IA32 Linux上（例如Intel Pentium或AMD athlon），推荐的编译器是Intel编译器ifort version 10.1或更高，或者Portland编译器pgf90 version 7.2或更高。对于Opteron 和 EM64T 系统，推荐的编译器是Portland编译器version 7.2或更高，Pathscale 编译器pathf90 version 3.1或更高，或者Intel编译器version 10.1或更高。支持的编译器的完整列表可以在<http://www.molpro.net/supported>找到。

2. GNU *make*, 可以从<http://www.fsf.org>及其它镜像免费下载。必须使用GNU *make*; 不能使用大多数系统的标准*make*。为了避免使用错误的*make*, 并且禁止GNU *make*的过多输出, 设置别名会很有用, 例如, `alias make='gmake -s'`。
  3. GNU *curl*工具, 用于批处理模式的http传输, 它虽然不需要安装, 但是它对以后应用修改BUG的补丁是最基本的。
  4. 编译时需要大约10GB的磁盘空间 (强烈依赖于系统; 文件系统的区块尺寸越大, 二进制文件越大)。最后的安装程序典型需要100Mb。大型计算还会需要更大的磁盘空间。
  5. 一个或者多个大型草稿文件的系统, 每一个都包含用户可以写入的目录。部分程序对I/O 有要求, 它们需要对两个文件同时执行, 因此当其它解决方法如分散连接 (striping) 不能使用的时候, 在不同的物理磁盘上提供至少两个文件系统是很有帮助的。目录名应当存储于环境变量\$TMPDIR, \$TMPDIR2, \$TMPDIR3, ...中。这些变量应当在程序安装前设置 (最好在.profile或.cshrc中), 因为在有些阶段, 安装流程会检查这些变量 (参见A.3.5节)。
  6. 如果编译用于并行执行的程序, 需要Global Arrays工具箱或MPI-2库。要用Global Arrays工具箱编译MOLPRO, 我们推荐最新的稳定版本 (虽然较早的版本也能使用)。它可以在<http://www.emsl.pnl.gov/docs/global> 获得, 应当在编译MOLPRO之前安装。在Global Arrays文档中提到, 有三种方法创建GA:
    - (1) 使用MPI的GA
    - (2) 使用TCGMSG-MPI的GA
    - (3) 使用TCGMSG的GA
- MOLPRO可以使用任何一种接口。要用MPI-2库创建MOLPRO, 我们推荐使用内置的MPI-2库, 它的优点是对某些平台做了优化。如果系统没有内置的MPI-2库, 那么在编译MOLPRO之前, 应当安装新的MPI-2库 (例如MPICH2, 参见<http://www.mcs.anl.gov/research/projects/mpich2>)。已经测试过很多MPI-2库, 包括Intel MPI, Bull MPI, MPICH2, 以及Open MPI。也可以用其它未测试的库。
7. MOLPRO发布源程序, 由一个压缩的tar文件组成, 文件名形式为molpro.2009.1.tar.gz, 同时可能还会有一个或几个module文件, 文件名形式为molpro.module

.2009.1.tar.gz。module文件包含通常不发布的源代码，或其功能在安装代码时并不需要的源代码。例子是程序开发者的工具（*module=develop*）。这些文件可以用gunzip和tar解压缩。所有文件必须在同一目录下解压缩。一般方法是先解压基本文件，接下来在进一步安装前，解压所有module。

### A.3.3 配置

一旦发布程序被解压缩，就需要确定创建的根目录（通常是molpro2009.1）。在以下的说明中，所有的目录都是相对于这个根目录。一旦改变了根目录，就需要检查含有所用Fortran编译器的目录是否在你的PATH环境变量中。接下来运行命令：

```
./configure -batch
```

这将会创建CONFIG文件。这个文件包含与机器有关的参量，例如编译器选项。通常不需要修改CONFIG，但至少应当检查它，修改任何你认为必要的配置参数。对CONFIG所做的任何修改，将在下次调用 `./configure` 时丢失，因此这些修改尽量通过命令行提供。

configure流程也可以给出命令行选项，运行时如果不加-batch，会对一些参数加上提示：

1. 在某些机器上可以用32位或64位整数编译程序，在这种情况下，可以分别为configure给出命令行选项-i4或-i8，覆盖默认的设置。一般64位选项可以做更大的计算（文件大于2Gb，16个以上的活性轨道），但是如果底层的硬件不支持64位整数，计算可能会更慢。注意，如果使用了-i4，在大多数系统上也支持大文件（大于2Gb），尽管如此，MOLPRO记录文件的最大尺寸也只能到16 Gb，因为MOLPRO的内部地址使用了32位整数。如果使用了-i8，记录与文件尺寸实际上是没限制的。通常我们推荐由configure确定的默认值。

2. 对于创建并行执行，命令行必须给出-mpp或-mppx选项。两种并行模式的区别，参见手册的第二部分。在Global Arrays根目录的路径之后还要给出-mppbase选项，或者含有目录的MPI-2库。对于使用Global Arrays工具箱的情况，例如

```
./configure -mpp -mppbase /usr/local/ga-4-2
```

并且-mppbase目录应当包含include和lib/TARGET目录，例如对于Linux/x86\_64是TARGET=LINUX64。

如果用MPI库创建了Global Arrays，并且当计算机上存在多个MPI时，相应的MPI可执行程序应出现在PATH环境变量的最开始。

关于Global Arrays安装的问题要直接发送到Global Arrays小组，所有涉及MOLPRO的问题都假定完全功能的Global Arrays套件已安装好，并且通过了所有的内部测试。

使用MPI-2库的情况，例如

```
./configure -mpp -mppbase /usr/local/mpich2-install/include
```

并且-mppbase目录应当包含文件mpi.h。应保证内置或新装的MPI-2库完全支持MPI-2标准，并且工作正常。

3. 如果有系统库位于非常规的位置，可能需要在-L命令行的选项中作为参数直接指定。
4. `configure`询问是否希望使用系统BLAS子程序库。MOLPRO带有这个库的自优化Fortran版，可以安全使用。但是在大多数计算机上，换用系统优化版更有效。在命令行可以指定来自系统的BLAS的级别，例如-blas2。如果指定了2，那么系统库将会用于级别2和级别1的BLAS，而MOLPRO的内部例程将用于级别3（即，矩阵与矩阵相乘）。但是通常只需要选择0或3。根据是否发现了BLAS库，它们是默认的。

如果使用了64位整数(-i8)，会有特殊的情况，因为很多平台的系统BLAS库只支持32位整数自变量。在这种情况下（例如IBM，SGI，SUN），BLAS库的级别可以设定为0或4。BLAS=0总是工作，表示使用MOLPRO的Fortran BLAS例程。有些平台（IBM，SGI，SUN）用BLAS=4会运行得更好；在这种情况下使用来自系统库的一些32位BLAS程序（它们从外壳程序调用，把整数自变量从64位转为32位。注意，如果使用了2 GB以上的内存，这可能会产生问题）。

为了更好地执行，使用合适的BLAS库是很重要的；特别是快速执行矩阵相乘的dgemm，它对MOLPRO非常重要。因此只要可能，尽量使用系统优化的BLAS库。

在很多情况下，MOLPRO自动探测最合适的BLAS库。对于某些情况，特别是在非默认位置安装了BLAS库的情况，应当把configure重定位到合适的位置，使用命令：

```
./configure -blaspath /path/to/lib/dir
```

BLAS库的说明可以简化，只需要把被下载的相应的库放到blaslibs目录中，configure将会搜索这个目录下（接下来是较低优先级的一些可能的系统目录）与特定硬件有关的库。

对于Intel和AMD Linux系统，我们推荐以下的BLAS库：

mkl	Intel数学核心库（MKL）								
atlas	Automatically Tuned Linear Algebra Software（ATLAS）库。必须使用针对特定处理器的atlas库：								
	<table> <tr> <td>Pentium III</td><td>Linux_PIIISSE1</td></tr> <tr> <td>Pentium 4, Xeon</td><td>Linux_P4SSE2</td></tr> <tr> <td>AMD Athlon</td><td>Linux_ATHLON</td></tr> <tr> <td>AMD Opteron</td><td>Linux_HAMMER64SSE2_2（64位）</td></tr> </table>	Pentium III	Linux_PIIISSE1	Pentium 4, Xeon	Linux_P4SSE2	AMD Athlon	Linux_ATHLON	AMD Opteron	Linux_HAMMER64SSE2_2（64位）
Pentium III	Linux_PIIISSE1								
Pentium 4, Xeon	Linux_P4SSE2								
AMD Athlon	Linux_ATHLON								
AMD Opteron	Linux_HAMMER64SSE2_2（64位）								
	当使用atlas时，MOLPRO会在额外的lapack子程序中自动编译，它们默认不是与软件包在一起的，因此Atlas所带的liblapack.a就足够了。								
acml	对于Opteron系统，AMD Core Math Library（ACML）是最好的blas库。								

SGI Altix最好使用scsl库。HP平台可以使用mlib数学库。IBM Power 平台可以使用essl包。

5. `configure`提示目标目录（`INSTBIN`），用于MOLPRO可执行程序的最后安装。这个目录通常应当在所有访问MOLPRO用户的PATH中。它的配置依赖于安装是私有的还是公有的。
6. `configure`提示用于安装辅助文件的目标目录（`INSTLIB`），这些辅助文件在程序执行时需要。
7. `configure`提示文档的目标目录。这通常应当是安装在互联网服务器上的目录。这仅与文档是否也从该目录安装有关（见下）。

可以被`configure`识别的命令行选项的完整列表：

<code>-af90</code>	使用Absoft Pro Fortran编译器。
<code>-batch</code>	非交互运行脚本。
<code>-blas0</code>	不使用优化的BLAS库。
<code>-blas1</code>	使用BLAS库级别1。
<code>-blas2</code>	使用BLAS库级别2。
<code>-blas3</code>	使用BLAS库级别3。

-blas4	使用BLAS库级别4。
-blaspath	指定BLAS库的路径。
-cc	使用名为cc的C编译器。
-clearspeed	编译clearspeed代码。
-clearspeedbase	指定用于头文件目录和库目录的clearspeed根路径。
-f	指定MOLPRO的Fortran预处理程序标记。
-f90	使用f90 Fortran编译器。
-fcc	使用Fujitsu C编译器。
-force-link	强制链接主执行程序。
-fort	使用fort Fortran编译器。
-frt	使用frt Fortran编译器。
-g95	使用G95 Fortran编译器。
-gabase	使用mppbase, 仅用于向下兼容。
-gcc	使用GNU编译器套装的C编译器。
-gforker	使用mpich的设置, mpich用gforker选项编译。
-gfortran	使用gfortran Fortran编译器。
-hdf5	指定HDF5根目录。
-i386	使用i386机器的设置。
-i4	令默认整数变量为四位长度。
-i686	使用i686机器的设置。
-i8	令默认整数变量为八位长度。
-icc	使用Intel C编译器。
-ifort	使用Intel Fortran编译器。
-instroot	安装的根目录。
-lapack	使用LAPACK。
-lapackpath	指定LAPACK库的路径。
-letter	指定latex信纸的尺寸。
-Linux	使用Linux内核的设置。
-mpp	产生并行MOLPRO。
-mppbase	指定mpp用于头文件目录和库目录的根路径。
-mppx	产生试用的并行MOLPRO。
-nagfor	使用NAG Fortran编译器。
-natom	最大的原子数。

-nbasis	最大的基函数数量。
-noclearspeed	不编译clearspeed代码。
-nolapack	不使用LAPACK。
-nolargefiles	不使用大文件。
-noslater	不编译slater代码。
-nprim	原函数的最大数量。
-nrec	记录的最大数量。
-nstate	每个对称性的态最大数。
-nsymm	态对称性的最大数量。
-nvalence	价轨道的最大数量。
-openc	使用Open64 C编译器。
-openf90	使用Open64 Fortran编译器。
-pathcc	使用Pathscale C编译器。
-pathf90	使用Pathscale Fortran编译器。
-pgcc	使用Portland C编译器。
-pgf90	使用Portland Fortran编译器。
-rpm	用RPM设置创建。
-slater	编译slater代码。
-trial	使用测试的分析目标文件。
-x86_64	使用64位x86机器的设置。
-xlf	使用IBM Fortran编译器。

### A.3.4 编译与链接

在配置之后，安装的余下部分用GNU *make*命令完成。注意在很多系统中，不能用默认的*make*，必须使用GNU *make*（参见A.3.2节）。要创建包含所有辅助文件的运行程序，默认只需要在MOLPRO根目录中简单地输入命令

**make**

大多数GNU *make*标准选项都可以安全使用，特别是-j可以加速在并行计算机上的编译。在确保bin/ 目录包含在PATH环境变量以后，接下来运行命令molpro就可以调用程序了。



### A.3.5 修改MOLPRO的默认环境变量

MOLPRO默认的运行选项存储于bin/molpro脚本中。从二进制文件或从源文件安装程序后，需要的话可以查看并修改这个文件，使程序有更广的使用范围。关于这些选项，参见第二章。

### A.3.6 调节

MOLPRO可以对特殊的体系进行调整，通过在根目录下运行命令

```
make tuning
```

这个任务自动确定调整参数的数量，并添加到文件bin/molpro中。用这些参数，MOLPRO将根据问题的大小选择最佳的BLAS程序。这一任务应当在空系统中运行。根据处理器的速度，典型地需要十分钟，在进行下一步之前，需要等待运行完成。

### A.3.7 测试

这一阶段检查程序是否已经正确编译。*makefile*的目标*test*（即，命令**make test**）将用整套测试任务进行测试，虽然这要花很长时间，但在第一次安装后总是需要做。更快的测试只用程序检查主要例程，可以用**make quicktest**进行。对于并行安装，用一个以上的运行进程进行验证很有必要。这可以很方便地通过**make**命令行完成，例如，

```
make MOLPRO_OPTIONS=-n2 test
```

如果有任务失败，必须研究原因。在这样的环境下把目标平台与MOLPRO支持的平台列表作比较可能会有帮助，这位于<http://www.molpro.net/supported>。如果在解决问题的努力之后，问题仍未解决，MOLPRO的开发者将很高兴能收到错误报告。这在基于网络的报错机制<http://www.molpro.net/bugzilla>中，应尽量详细填写。**make test**产生形式为report.\*.sh.gz的文件，它包含了MOLPRO安装的细节，以及出错测试任务的输出文件。通常应当在错误报告中附上这一文件。请注意，错误报告的目的是帮助开发者改进代码，而不是提供安装和运行的建议。

### A.3.8 安装产品级的程序

虽然程序可以在原位置使用，但是更方便的做法是只把那些运行时需要的文件复制到合适的安装目录中，它们在配置时指定（参见A.3.3节），并存储于CONFIG文件中。要用这种方法安装程序，可以运行

```
make install
```

完整的源代码树接下来可以存档和删除。这一步的作用是在INSTLIB目录创建一个shell脚本。名称应当与体系架构、编译类型、整数位等有关。接下来产生与编译类型

有关的符号链接，最后，如果INSTLIB/molpro不是一个文件，那么将用符号链接创建新的脚本。在有些情况下，最好在INSTLIB/molpro中创建本地脚本，它将不会被覆盖。在通常情况下，这些链接的作用是提供molpro命令以及molpros（串行）和molprop（并行）命令之一，用于正常的使用，也可以选择长名称直接调用特定的版本。

对于通常的单版本安装，不需要上面的考虑，可以直接从INSTLIB目录使用molpro命令。

在安装过程中，来自\$HOME/.molpro/token的密钥将被复制到INSTLIB/.token，因此这个密钥可用于该安装版的所有用户。

当程序校验和/或安装了以后，make clean命令可以删除编译的日志记录。而make veryclean将删除所有的二进制文件和目标文件，只保留原始的发布文件；通常不推荐这么做，因为这意味着在以后使用升级和BUG修改时，整个程序必须重新编译。

### A.3.9 获取和使用补丁

通常，下载的发布程序是完全更新的，不需要以前的补丁。但是，以后的BUG修改和升级是需要的。加入了BUG修改和新功能的MOLPRO源代码升级机制需要通过自包含补丁文件，在使用的时候会替换或添加文件，并存储被替换的源代码，用于以后恢复到以前版本。可用的补丁参见<https://www.molpro.net/download/patch?version=2009.1>，同时，已经安装的补丁列表会在程序运行时打印出来。打补丁后，补丁文件自动更新所有需要重建的目标文件；例如，有关的目标文件会被删除。因此，当使用了所有的补丁文件之后，通常需要用make重新编译程序。

使用和删除补丁的顺序非常重要。有些补丁是其它补丁所必需的，而有些补丁是一个或多个“子”补丁的“父”补丁：“父”补丁和“子”补丁共同拥有一个或多个公用文件，只不过“父”补丁比“子”补丁旧。如果顺序违反了这一规则，个别的补丁脚本会拒绝安装或恢复。为了顺利解决这个问题，提供了patcher程序，用来管理一个或多个补丁的应用或删除。patcher程序可以对补丁应用或恢复的顺序进行排序，以避免冲突；如果需要，它还可以还原并再次应用补丁。

要运行patcher程序，应当运行命令：

```
make patch
```

这对大多数应用已经足够。如果尚未编译patcher，将会创建它，接下来它会访问网络服务器，应用可用的补丁，并返回使用的补丁级别。

如果需要把参数传递给patcher程序，可以在顶级目录使用：

```
./patcher [--apply | --revert | --list]
```

```
[--cache-directory] [--user] [--password]
[--url] [--local]
[--verbose] [--no-action] patch1 patch2 ....
```

依照选项，它可以用三种模式之一进行操作：

--apply, -a       （默认）应用（即，安装）补丁。  
--revert, -r       还原（即，删除）补丁。  
--list, -l         列出可用的和已安装的补丁。

安装或者删除的补丁列表可以在所有选项之后的命令行中给出，直接使用补丁名称列表，或在应用程序的情况下使用补丁文件列表。在应用程序的情况下，通常也可以通过选项请求本地缓存中的所有补丁，或者是所有可用的补丁。

来自中心网络服务器（默认<http://www.molpro.net>）的MOLPRO补丁，用此程序暂存在本地目录中（默认是`$HOME/.molpro/cache`）。典型情况下，访问网络服务器必须经过认证：第一次运行这个程序的时候，可以通过命令行选项定义用户名和口令，否则程序将会给出提示。接下来它们会被记录在缓存目录的`CONFIG`文件中。

如果出现问题，首先查看`patcher.log`文件，它包含了应用和还原每个补丁的输出信息。

可以给出以下选项。

--cache-directory, -c d	定位缓存目录。
--verbose, -v	增加打印的信息量。可以使用多个--verbose选项。
--noverbose	减少打印的信息量。
--url	网络服务器的URL。
--user, -u u	网络服务器的用户名。
--password, -p p	网络服务器的口令。
--noaction, -n	不应用补丁或者还原补丁。用于观察不打补丁的结果。
--local	不访问网络服务器，只使用本地文件。
--token, -k	下载许可密钥。
--ssl, -s	连接网络服务器时，使用SSL。
--nossl, -i	关闭SSL的使用。

例子：

`patcher`            使用所有可用但尚未安装的补丁。这个功能是使源代码树目录包含所有升级的标准用法。

`patcher -l`        列出已安装的和可用的补丁。

`patcher -r xx yy`   还原补丁xx和yy。

`patcher -n`        把所有未安装的补丁载入到缓存中，供以后使用。

`patcher --local`    应用缓存中的所有补丁，无需网络连接。

### A.3.10 安装文档

文档可以从<http://www.molpro.net/info/users>获得。它还包含在源代码中。PDF用户手册可以在molpro2009.1/doc/manual.pdf目录中找到，HTML版位于目录molpro2009.1/doc/manual（顶级文件是frames.html）。文档将被复制到最终的目标位置，这个位置用configure命令所产生的CONFIG文件指定。如果要安装文档和交互式基组工具，可以在doc目录中执行make install。手册中包括大量的示例输入文件，它们也可以在molpro2009.1/examples目录中找到。

### A.3.11 openSuSE 11.0的简易安装

在openSuSE 11.0 Linux上，仅用自由软件从源代码安装MOLPRO非常容易。这些说明假定openSuSE 11.0 是标准安装。对于所有发布版本，需要通过YaST安装以下的附加包：

`gcc-c++`        提供GNU C编译器。

`gcc-fortran`    提供GNU Fortran编译器。

`make`           提供GNU make。

`zlib-devel`    提供用于压缩文件的库。

还可以选择安装：

`blas`           提供BLAS库。

`lapack`        提供LAPACK库。

它们将用于代替编译MOLPRO中的等效例程。

#### A.3.11.1 串行MOLPRO

在MOLPRO解包后，编译MOLPRO仅需要简单地输入

```
./configure -batch -gcc -gfortran  
make
```

### A.3.11.2 并行MOLPRO使用带有MPI的Global Arrays

以下的附加包应当通过YaST安装:

`mpich` 提供MPI-1库。

`mpich-devel` 提供MPI-1库。

应当下载最终稳定版本的Global Arrays工具箱。对64位机器, 创建和测试Global Arrays工具箱使用:

```
make TARGET=LINUX64 FC=gfortran CC=gcc USE_MPI=y\  
    MPI_LIB=/opt/mpich/ch-p4/lib64\  
    MPI_INCLUDE=/opt/mpich/ch-p4/include LIBMPI=-lmpich  
/opt/mpich/ch-p4/bin/mpirun ./global/testing/test.x
```

或者在32位机上使用:

```
make TARGET=LINUX FC=gfortran CC=gcc USE_MPI=y\  
    MPI_LIB=/opt/mpich/ch-p4/lib\  
    MPI_INCLUDE=/opt/mpich/ch-p4/include LIBMPI=-lmpich  
/opt/mpich/ch-p4/bin/mpirun ./global/testing/test.x
```

接下来配置和编译MOLPRO, 使用:

```
./configure -batch -gcc -gfortran\  
            -mpp -mppbase /path/to/directory/ga-4-2  
make
```

### A.3.12 在Cygwin系统安装

在Windows机器上, 应当安装Cygwin。除了列出的默认包以外, 还应当安装表A.1列出的包。如果进行开发工作, 表A.2列出了可能有用的包。

表 A.1: Cygwin需要用户安装的包

包	包的分类	原因
gcc	Devel	用于编译C文件
make	Devel	需要GNU make
curl	Web	patcher需要curl

表 A.2: Cygwin用于开发用户的包

包	包的分类	原因
bison	Devel	bison
gdb	Devel	gdb
libxslt	Gnome	xsltproc
openssh	Net	ssh
vim	Editors	vi
wget	Web	wget, 对于patcher来说, 是curl的备选

除了以上的包外, 还应当安装Fortran编译器。目前对Cygwin测试过的唯一编译器是g95。需要从G95网站下载g95-x86-cygwin.tgz, 接下来在Cygwin基本目录中untar。下载稳定版本非常重要。我们不推荐下载自解压版本。

通过以上的步骤, 现在可以对MOLPRO运行configure, 它会把Cygwin处理成一般的Linux 系统。一旦完成了程序的编译, 就可以在Cygwin下按照通常的方式运行MOLPRO。

### A.3.13 在MacOS下的并行安装

这一部分包含了一些注释和例子, 用于创建并行MOLPRO所需的程序。这一部分还不完整, 在任何情况下用户应当直接参考出问题程序的文档, 并把问题直接报告他们。

#### A.3.13.1 对64位Mac OS X独立计算机创建MPICH 2

mpdallexit

```
CFLAGS=-m64 FC=ifort FFLAGS='-pc 64 -auto' F90=ifort \
F90FLAGS='-pc 64 -auto' \
./configure --prefix=/usr/local/mpich2-1.0.7_nemesis \
--with-pm=gforker --with-device=ch3:nemesis
```

```
make
sudo make install
```

### A.3.13.2 对64位Mac OS X独立计算机创建Global Arrays

```
mpdallexit
export MPI_LOC=/usr/local/mpich2-1.0.7_nemesis
export PATH=${MPI_LOC}/bin:$PATH
export MPI_LIB=${MPI_LOC}/lib
export MPI_INCLUDE=${MPI_LOC}/include
export USE_MPI=y
```

```
make USE_MPI=1 TARGET=MACX64 FC=ifort \
LIBMPI="-lpmpich -lmpich -lpmpich" \
USE_INTEGER8=y COPT="-m64"
```

```
export PATH=${MPI_LOC}/bin:$PATH
mpiexec -n 2 global/testing/test.x
```





## B.1 MOLPRO2009.1的新功能

### B.1.1 显关联计算

由于新的发现，在以下几点修改了F12程序的默认属性：

1. 对于开壳层体系，修改了默认的波函解。这会影响RMP2-F12和开壳层CCSD-F12计算。新的默认设置一般会改善开壳层的处理，并导致更一致的行为。恢复以前的属性可以用

`explicit,extgen=0`

（更多细节参见手册。）

2. 改变了构造互补辅助基组（CABS）的流程和阈值。这会影响所有的非局域F12计算。恢复以前的属性可以用

`explicit,ortho_cabs=0,thrcabs=1-7,thrcabs_rel=1e-8`

3. 在数值频率计算中，改善了对辅助基组的冻结。这会影响删除许多冗余函数的计算。

4. 要打印显关联方法的电子对能量，可以用

`print,pairs`

如果还关联内壳层轨道，那么还打印cc，cv，和vv对关联能的贡献。

### B.1.2 Hartree-Fock程序的改善

改善和扩充了Hartree-Fock中的原子密度猜测。现在，猜测基组可用于大多数的原子和所有的赝势。大多数的赝势已连接到合适的基组上，因此一般的设置足够，例如指定

```
basis=vtz-pp
```

将会选择关联一致三zeta基组和相应的（小芯）赝势。类似地，通常指定其它赝势的基组或基组组合已经足够。

如果在Hartree-Fock输入中没有给出波函对称性，并且无法从以前的计算中获知，那么HF程序将尝试从构造原理（Aufbau principle）自动获得对称性（以前的版本总是假定对称性1）。例如，

```
geometry=n;
```

```
{hf;wf,spin=3}
```

将会自动找到波函对称性是8。

### B.1.3 结构输入的变化

1. 分子结构选项的合理化。现在在结构区内指定对称性和方位选项（例如x;noorient;angstrom）是错误的，只能包含结构的定义（Z矩阵或XYZ）。指定选项必须用新的命令ORIENT或SYMMETRY，和/或已存在的命令，如ANGSTROM。不幸的是，这一变化将会使得许多输入与2008.1和更早版本的MOLPRO不兼容，但是可以纠正和澄清结构分析程序，例如，包含钇原子的结构，以前会与Y对称性选项冲突。
2. 简化了结构输入。程序现在会自动探测结构使用了Z矩阵，还是笛卡尔坐标，因此不再需要设定geomtyp变量。对于笛卡尔坐标，仍旧接受标准的XYZ格式，但是如果需要的话，前两行（原子数和注释）也可以忽略。

### B.1.4 MPI-2并行执行

对于并行执行，现在可以用Global Arrays工具箱或MPI-2库，从源文件创建程序。

## B.2 MOLPRO2008.1的新功能

MOLPRO 2008.1版本中的新功能包括：

1. 高效的闭壳层和开壳层MP2-F12和CCSD(T)-F12方法，可以显著改变基组的收敛性。说明见：J. Chem. Phys. **126**, 164102 (2007); 同上，**127**, 221106 (2007); 同上，**128**, 154103 (2008)。

2. 自然键级 (NBO) 和自然布居分析 (NPA), 说明见Mol. Phys. **105**, 2753 (2007)及其中的参考文献。
3. 局域分子轨道方法中的关联区, 说明见J. Chem. Phys. **128**, 144106 (2008)。
4. 用VCI方法自动计算非谐振动频率和零点能, 说明见J. Chem. Phys. **126**, 134108 (2007)及其中的参考文献。
5. DFT和耦合簇的耦合方法, 说明见Phys. Chem. Chem. Phys. **10**, 3353 (2008)及其中的参考文献。
6. 与其它程序的增强连接, 包括输出和三维结构的图形显示。
7. 支持最新的操作系统和编译器, 包括Mac OS X。

### B.3 MOLPRO2006.1的新功能

MOLPRO 2006.1版有许多新功能和增强, 最值得注意的是密度拟合方法, 显关联方法, 局域耦合簇方法, 以及几种新的梯度程序。如下:

1. 更一致的输入语言和输入预检查。
2. 更灵活的基组输入, 允许操作多个基组。
3. 新的更高效的密度泛函执行, 增加了密度泛函。
4. 低阶标度的局域耦合簇方法, 允许三激发微扰处理 (LCCSD(T)及其变体, 如LQCISD(T))。
5. 高效的密度拟合 (DF) 程序, 用于Hartree-Fock (DF-HF), 密度泛函Kohn-Sham理论 (DF-KS), 二阶Møller-Plesset微扰理论 (DF-MP2), 以及所有的局域方法 (DF-LMP2, DF-LMP4, DF-LQCISD(T), DF-LCCSD(T))。
6. QCISD(T)解析梯度。
7. MRPT2 (CASPT2) 和MS-CASPT2解析梯度, 使用态平均MCSCF参考函数。
8. DF-HF, DF-KS, DF-LMP2, 和DF-SCS-LMP2解析梯度。
9. 使用密度拟合的显关联方法: DF-MP2-R12/2A'和DF-MP2-F12/2A', 及其局域变体DF-LMP2-R12/2\*A(loc)和DF-LMP2-F12/2\*A(loc)。

10. 多参考微扰理论和组态相互作用的组合 (CIPT2)。
11. DFT-SAPT。
12. 用不同轨道的CASSCF和MRCI波函之间的跃迁矩和跃迁哈密顿量。
13. 新的自旋轨道积分程序，可以使用一般收缩基组。
14. 到任意级次的Douglas-Kroll-Hess标量相对论哈密顿量。
15. 改善了结构优化和数值Hessian计算（包括束缚优化）的流程。
16. 对QM/MM计算改善了程序，可以处理大格点的点电荷，包括格点梯度（参见第四十九章）。
17. 到M. Kallay的MRCC程序的接口，允许任意激发级别的耦合簇计算。
18. 数值梯度和Hessian的自动大规模并行计算 (mppx版)。
19. 加入并行代码，例如DF-HF，DF-KS，DF-LCCSD(T)（部分并行化，包括微扰三激发）。
20. 增加输出格式，用于制表 (XHTML, L<sup>A</sup>T<sub>E</sub>X, MAPLE, MATHEMATICA, MATLAB 和逗号分隔变量的格式)，输出轨道和基组 (XML)，以及可选的完美形式XML输出流，其中标出了重要结果。

## B.4 MOLPRO2002.6的新功能

相对于2002.1版，改变和增加了以下功能：

1. 对IA-64 (Itanium2) 支持IA-64 Linux系统 (HP和NEC) 和HP-UX 11.22。
2. 支持NEC-SX系统。
3. 支持IBM-power4系统。
4. 修改了MOLPRO系统变量的操作。改变了SET命令（参见第八章和8.4节）。
5. 分子的总电荷可以用CHARGE变量或在WF卡指定，参见4.9节。
6. 用对称位移坐标，改善了数值几何优化（参见第40.2节和四十一章）。
7. 利用对称性改善了数值频率计算 (AUTO选项，参见第四十二章)。

## B.5 MOLPRO2002的新功能

相对于2000.1版，有以下的主要改变和增强：

1. 直接和局域模块现在加入到基本版中。这意味着现在无需特别许可，就可以使用直接积分程序，说明见：

M. Schütz, R. Lindh, and H.-J. Werner, Mol. Phys. 96, 719 (1999),

线性标度局域MP2程序，说明见：

G. Hetzer, P. Pulay, and H.-J. Werner, Chem. Phys. Lett. 290, 143 (1998),

M. Schütz, G. Hetzer, and H.-J. Werner, J. Chem. Phys. **111**, 5691 (1999),

G. Hetzer, M. Schütz, H. Stoll, and H.-J. Werner, J. Chem. Phys. **113**, 9443 (2000),

以及LMP2梯度，说明见：

A. El Azhary, G. Rauhut, P. Pulay, and H.-J. Werner, J. Chem. Phys. **108**, 5185 (1998)

以后还可以使用线性标度LCCSD(T)方法：

M. Schütz and H.-J. Werner, J. Chem. Phys. **114**, 661 (2001),

M. Schütz and H.-J. Werner, Chem. Phys. Lett. **318**, 370 (2000),

M. Schütz, J. Chem. Phys. **113**, 9986 (2000)

2. 可以使用QCISD梯度，说明见：Phys. Chem. Chem. Phys. **3**, 4853 (2001)。
3. 对计算数值梯度和执行几何优化加入了更灵活的选项。
4. 加入了大量的密度泛函，以及对自动泛函执行程序的支持，说明见：Comp. Phys. Commun. **136** 310–318 (2001)
5. 可以计算任意级别的多极矩。
6. 更多的模块作了并行化，特别是CCSD(T)和直接LMP2代码。改善了并行执行流程。并行版本可作为可选的模块使用。
7. 扩充了基组库。
8. 基组输入中作了略微的改动：对于不同原子，不再允许连续使用几个单行的basis输入卡。现在每个新的BASIS 卡将覆盖前面的卡。因此，要么所有的说明必须在一个BASIS卡中给出，要么使用BASIS输入区。现在BASIS,NAME 完全等

价于BASIS=NAME，也就是说，在两种情况中都定义全局默认基组并设置BASIS变量。

9. 赝势能量计算可以执行到*i*函数，梯度计算到*h*函数。
10. 做了很多内部修改，使得MOLPRO更加模块化，并且更加稳定。增加了对最近的操作系统Compaq, HP, SGI, SUN 和Linux的支持。改善了补丁系统。

## B.6 MOLPRO2000的新功能

相对于98.1版，有以下的主要改变和增加：

1. 在文献J. Chem. Phys. 99, 5129 (1993)自旋限制开壳层耦合簇方程的推导中存在着基本错误，这也反映在MOLPRO的98.1及更早版本的RCCSD代码中。现在已经修改了这一错误，勘误发表于J. Chem. Phys. **112**, 3106 (2000)。万幸的是，这个错误造成的数值误差很小，并不表示已计算的特性有明显错误。
2. 笛卡尔坐标到内坐标的梯度变换有程序错误，有些情况下会导致几何优化的收敛变慢。这一错误已经修改。
3. 振动频率过去默认使用平均原子质量，而不是最常见的同位素，现在这已改为默认属性。
4. 加入MCSCF二阶导数（作者：Riccardo Tarroni）（初级版，还不能使用对称性）。修改了频率和几何优化程序，现在它们可以使用解析Hessian。
5. 新的内收缩多参考二阶微扰理论代码（作者：Paolo Celani），使用命令RS2C，说明见P. Celani and H.-J. Werner, J. Chem. Phys. **112**, 5546 (2000)。
6. EOM-CCSD激发态计算（作者：Tatiana Korona）。
7. 作为真实解析能量导数的QCISD偶极矩（作者：Guntram Rauhut）。
8. 线性标度（CPU和内存）的LMP2，说明见  
G. Hetzer, P. Pulay, and H.-J. Werner, Chem. Phys. Lett. 290, 143 (1998).  
M. Schütz, G. Hetzer, and H.-J. Werner, J. Chem. Phys. **111**, 5691 (1999).
9. 改善了基组和几何结构记录的操作。98.1和99.1转存文件可以续算，但是对于从旧文件开始有问题的情况，可以在file卡之后立即接RESTART,NOGEOM。同样，

对于非常大的体系，如果出现了不能验证的信息“ORBITALS CORRESPOND TO DIFFERENT GEOMETRY”，可以试试`ORBITAL,record,NOCHECK`。（这会在100个原子以上的情况，因为旧版本上限为100个原子。）

10. 重新组织和推广了基组输入。增加了基组库。
11. 平衡校正的几何优化。
12. 对MPP机器改善了的运行流程。可以并行使用直接SCF和SCF梯度。这些功能仅能用于MPP模块，尚未发布。
13. 修改了DFT格点，CCSD使用分页，没有芯轨道的有限场计算，以及自旋轨道耦合中的重大错误。
14. 很多其它的内部改变。

作为MOLPRO社团的附加服务，电子邮件列表已经设置为论坛，提供所有关于安装和使用MOLPRO的开放讨论。邮件列表主要用来发布如何有效使用MOLPRO的提示和技巧。这并不意味着直接向程序的作者提问。对于明显的程序错误，报告应发送到molpro-support@molpro.net；而发送到这里关于“如何做”的问题，仅仅会改发到邮件列表中。

如果要订阅邮件列表，请发邮件给molpro-user-request@molpro.net，并包含文本**subscribe**；要寻求帮助，用文本**help**发送电子邮件。

信息会发送到列表（molpro-user@molpro.net）中，但这只对订阅者才会进行。过去的记录可以在存档<http://www.molpro.net/molpro-user/archive> 中察看，无论是否订阅了列表。鼓励有经验的MOLPRO用户张贴问题的回复。请为这一资源发挥作用做出贡献。

## B.7 MOLPRO98的新功能

MOLPRO98具有MOLPRO96的全部功能，但是为了使代码更加模块化并且更容易使用和维护，做了一些结构上的变化。特别是不同记录的数量做了较大程度的减少。给定波函类型的信息，像轨道，密度矩阵，Fock矩阵，占据数，以及其它的信息，现在存储于单独的转存记录中。即使是不同的轨道类型，例如正则轨道、自然轨道或局域化轨道，也都存储于同样的记录中，用户可以在后面的步骤中用ORBITAL指令的关键字访问个别的数据集。对于SCF和MCSCF计算，新的工具允许使用不同基组和/或不同对称性计算的初始轨道。SCF计算的默认初始猜测已经做了很大改善，这对大分子

计算很有用。计算非绝热耦合或透热轨道的特殊流程在使用上已经做了极大的简化。我们希望这些变化会使程序更容易使用，并减少输入出错的可能。但是，为了更有效地使用新的工具，即使是有经验的MOLPRO用户也应当阅读手册的**记录**和**选择轨道和密度矩阵**部分。也许标准的MOLPRO96输入仍能工作，但在涉及轨道、密度矩阵或算符等特别记录的特殊情况下，可能需要作修改。

所有需要计算期待值和跃迁特性的单电子算符现在都存储于一个单独的记录中。计算所需期待值的算符可以用全局的**GEXPEC** 指令对给定计算中的所有程序进行全局选择，或者用**EXPEC**指令对特定的程序进行选择。当需要的时候，所有的算符自动计算，用户不需要再给出输入。细节参见手册的**单电子算符和期待值**部分。

由于转存和算符记录改变了结构，工具程序**MATROP**有了新的输入语法。**MATROP**的MOLPRO96输入不再可用。

除了这些组织上的变化以外，还添加了一些新的程序。现在可以对MP2和DFT波函求解解析能量梯度，用解析梯度的有限差分可以自动计算谐振频率，强度，以及热动力学特性。几何优化做了进一步的改进，加入了新的反应路径跟踪工具。

加入了到图形程序**MOLDEN**的接口，它可以显示分子结构，轨道，电子密度或者振动。

除了在MP4和CCSD(T)计算中的三激发微扰外，现在可以对各种类型的波函进行直接积分计算，其中AO基的双电子积分一旦需要总是重新计算，而不再存于磁盘中。这可以使用比以前大得多的基组。直接选项可以用**GDIRECT**指令进行全局选择，或者对特定的程序使用**DIRECT**指令。细节参见手册的**直接积分方法**部分。注意**DIRECT**模块是可选的，并不是MOLPRO基本发布程序的一部分。

进一步改进了局域电子关联方法。与使用了高效预屏蔽技术的直接积分模块相结合，它极大地减小了计算耗时与分子尺寸的标度，现在与MP2和更高的关联方法呈近似二次，甚至是线性的关系。这使得可以用关联方法计算比以前的尺寸更大的分子。但是，由于这些方法目前尚在研究中，仍在做进一步的开发，因此我们决定在目前的MOLPRO发布版中不包含它们。它们可以在以后的发布版中使用。



## 附录 C

## 密度泛函说明

### C.1 B86: $X_{\alpha\beta\gamma}$

无发散的半经验梯度校正交换能量泛函。文献中,  $\lambda = \gamma$ 。更多细节参见文献 [2]。

$$K = \sum_s -\frac{c (\rho_s)^{4/3} (1 + \beta (\chi_s)^2)}{1 + \lambda (\chi_s)^2}, \quad (\text{C.1})$$

其中,

$$c = 3/8 \sqrt[3]{34^{2/3} \sqrt[3]{\pi^{-1}}}, \quad (\text{C.2})$$

$$\beta = 0.0076, \quad (\text{C.3})$$

$$\lambda = 0.004. \quad (\text{C.4})$$

为了避免极限  $\rho_s \rightarrow 0$  的奇异性,

$$G = -\frac{c (\rho_s)^{4/3} (1 + \beta (\chi_s)^2)}{1 + \lambda (\chi_s)^2}. \quad (\text{C.5})$$

### C.2 B86MGC: 加上改进梯度校正的 $X_{\alpha\beta\gamma}$

对大密度梯度加上了改进梯度校正的B86。更多细节参见文献 [3]。

$$K = \sum_s -c (\rho_s)^{4/3} - \frac{\beta (\chi_s)^2 (\rho_s)^{4/3}}{(1 + \lambda (\chi_s)^2)^{4/5}}, \quad (\text{C.6})$$

其中,

$$c = 3/8 \sqrt[3]{34^{2/3} \sqrt[3]{\pi^{-1}}}, \quad (\text{C.7})$$

$$\beta = 0.00375, \quad (\text{C.8})$$

$$\lambda = 0.007. \quad (\text{C.9})$$

为了避免极限  $\rho_{\bar{s}} \rightarrow 0$  的奇异性,

$$G = -c (\rho_s)^{4/3} - \frac{\beta (\chi_s)^2 (\rho_s)^{4/3}}{(1 + \lambda (\chi_s)^2)^{4/5}}. \quad (\text{C.10})$$

### C.3 B86R: 重新优化的 $X_{\alpha\beta\gamma}$

重新优化  $\beta$  的 B86, 在 Becke 1997 年的论文第三部分使用。更多细节参见文献 [6]。

$$K = \sum_s -\frac{c (\rho_s)^{4/3} (1 + \beta (\chi_s)^2)}{1 + \lambda (\chi_s)^2}, \quad (\text{C.11})$$

其中,

$$c = 3/8 \sqrt[3]{34^{2/3} \sqrt[3]{\pi^{-1}}}, \quad (\text{C.12})$$

$$\beta = 0.00787, \quad (\text{C.13})$$

$$\lambda = 0.004. \quad (\text{C.14})$$

为了避免极限  $\rho_{\bar{s}} \rightarrow 0$  的奇异性,

$$G = -\frac{c (\rho_s)^{4/3} (1 + \beta (\chi_s)^2)}{1 + \lambda (\chi_s)^2}. \quad (\text{C.15})$$

## C.4 B88: 1988年Becke的交换泛函

更多细节参见文献 [4]。

$$K = \sum_s -(\rho_s)^{4/3} \left( c + \frac{\beta (\chi_s)^2}{1 + 6 \beta \chi_s \operatorname{arcsinh}(\chi_s)} \right), \quad (\text{C.16})$$

其中,

$$c = 3/8 \sqrt[3]{34^{2/3} \sqrt[3]{\pi^{-1}}}, \quad (\text{C.17})$$

$$\beta = 0.0042. \quad (\text{C.18})$$

为了避免极限  $\rho_s \rightarrow 0$  的奇异性,

$$G = -(\rho_s)^{4/3} \left( c + \frac{\beta (\chi_s)^2}{1 + 6 \beta \chi_s \operatorname{arcsinh}(\chi_s)} \right). \quad (\text{C.19})$$

## C.5 B88C: 1988年Becke的关联泛函

依赖于B86MGC交换泛函的关联泛函, 具有经验原子参数  $t$  和  $u$ 。与B88C结合使用的交换泛函应当替换这里的B88MGC。更多细节参见文献 [7]。

$$K = -0.8 \rho_\alpha \rho_\beta q^2 \left( 1 - \frac{\ln(1+q)}{q} \right) + \sum_s -0.01 \rho_s d z^4 \left( 1 - 2 \frac{\ln(1+1/2 z)}{z} \right), \quad (\text{C.20})$$

其中,

$$q = t(x + y), \quad (\text{C.21})$$

$$x = 0.5 \left( c\sqrt[3]{\rho_\alpha} + \frac{\beta (\chi_\alpha)^2 \sqrt[3]{\rho_\alpha}}{(1 + \lambda (\chi_\alpha)^2)^{4/5}} \right)^{-1}, \quad (\text{C.22})$$

$$y = 0.5 \left( c\sqrt[3]{\rho_\beta} + \frac{\beta (\chi_\beta)^2 \sqrt[3]{\rho_\beta}}{(1 + \lambda (\chi_\beta)^2)^{4/5}} \right)^{-1}, \quad (\text{C.23})$$

$$t = 0.63, \quad (\text{C.24})$$

$$z = 2ur, \quad (\text{C.25})$$

$$r = 0.5 \rho_s \left( c(\rho_s)^{4/3} + \frac{\beta (\chi_s)^2 (\rho_s)^{4/3}}{(1 + \lambda (\chi_s)^2)^{4/5}} \right)^{-1}, \quad (\text{C.26})$$

$$u = 0.96, \quad (\text{C.27})$$

$$d = \tau_s - 1/4 \frac{\sigma_{ss}}{\rho_s}, \quad (\text{C.28})$$

$$c = 3/8 \sqrt[3]{34^{2/3} \sqrt[3]{\pi^{-1}}}, \quad (\text{C.29})$$

$$\beta = 0.00375, \quad (\text{C.30})$$

$$\lambda = 0.007. \quad (\text{C.31})$$

为了避免极限 $\rho_s \rightarrow 0$ 的奇异性,

$$G = -0.01 \rho_s dz^4 \left( 1 - 2 \frac{\ln(1 + 1/2 z)}{z} \right). \quad (\text{C.32})$$

## C.6 B95: 1995年Becke的关联泛函

依赖于 $\tau$ 的动态关联泛函。更多细节参见文献 [5]。

$$K = \frac{E}{1 + l((\chi_\alpha)^2 + (\chi_\beta)^2)} + \sum_s \frac{F\epsilon(\rho_s, 0)}{H(1 + \nu(\chi_s)^2)^2}, \quad (\text{C.33})$$

其中,

$$E = \epsilon(\rho_\alpha, \rho_\beta) - \epsilon(\rho_\alpha, 0) - \epsilon(\rho_\beta, 0), \quad (\text{C.34})$$

$$l = 0.0031, \quad (\text{C.35})$$

$$F = \tau_s - 1/4 \frac{\sigma_{ss}}{\rho_s}, \quad (\text{C.36})$$

$$H = 3/5 6^{2/3} (\pi^2)^{2/3} (\rho_s)^{5/3}, \quad (\text{C.37})$$

$$\nu = 0.038, \quad (\text{C.38})$$

$$\begin{aligned} \epsilon(\alpha, \beta) = & (\alpha + \beta) \left( e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \right. \\ & - \frac{e(r(\alpha, \beta), T_3, U_3, V_3, W_3, X_3, Y_3, P_3) \omega(\zeta(\alpha, \beta)) (1 - (\zeta(\alpha, \beta))^4)}{c} \\ & \left. + \left( e(r(\alpha, \beta), T_2, U_2, V_2, W_2, X_2, Y_2, P_2) - e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \right) \right) \\ & \omega(\zeta(\alpha, \beta)) (\zeta(\alpha, \beta))^4, \end{aligned} \quad (\text{C.39})$$

$$r(\alpha, \beta) = 1/4 \sqrt[3]{34^{2/3}} \sqrt[3]{\frac{1}{\pi(\alpha+\beta)}}, \quad (\text{C.40})$$

$$\zeta(\alpha, \beta) = \frac{\alpha - \beta}{\alpha + \beta}, \quad (\text{C.41})$$

$$\omega(z) = \frac{(1+z)^{4/3} + (1-z)^{4/3} - 2}{2 \sqrt[3]{2} - 2}, \quad (\text{C.42})$$

$$e(r, t, u, v, w, x, y, p) = -2t(1 + ur) \ln \left( 1 + 1/2 \frac{1}{t(v\sqrt{r} + wr + xr^{3/2} + yr^{p+1})} \right), \quad (\text{C.43})$$

$$c = 1.709921, \quad (\text{C.44})$$

$$T = [0.031091, 0.015545, 0.016887], \quad (\text{C.45})$$

$$U = [0.21370, 0.20548, 0.11125], \quad (\text{C.46})$$

$$V = [7.5957, 14.1189, 10.357], \quad (\text{C.47})$$

$$W = [3.5876, 6.1977, 3.6231], \quad (\text{C.48})$$

$$X = [1.6382, 3.3662, 0.88026], \quad (\text{C.49})$$

$$Y = [0.49294, 0.62517, 0.49671], \quad (\text{C.50})$$

$$P = [1, 1, 1]. \quad (\text{C.51})$$

为了避免极限  $\rho_s \rightarrow 0$  的奇异性,

$$G = \frac{F\epsilon(\rho_s, 0)}{H(1 + \nu(\chi_s)^2)^2}. \quad (\text{C.52})$$

## C.7 B97DF: B97的密度泛函部分

这个泛函需要与0.1943倍的精确交换相混合。更多细节参见文献 [6]。

$$\begin{aligned}
 K = & (\epsilon(\rho_\alpha, \rho_\beta) - \epsilon(\rho_\alpha, 0) - \epsilon(\rho_\beta, 0)) (A_0 + A_1 \eta(d, \lambda_1) + A_2 (\eta(d, \lambda_1))^2) \\
 & + \sum_s \epsilon(\rho_s, 0) \left( B_0 + B_1 \eta((\chi_s)^2, \lambda_2) + B_2 (\eta((\chi_s)^2, \lambda_2))^2 \right) \\
 & - 3/8 \sqrt[3]{34}^{2/3} \sqrt[3]{\pi^{-1}} (\rho_s)^{4/3} \left( C_0 + C_1 \eta((\chi_s)^2, \lambda_3) + C_2 (\eta((\chi_s)^2, \lambda_3))^2 \right),
 \end{aligned} \tag{C.53}$$

其中,

$$d = 1/2 (\chi_\alpha)^2 + 1/2 (\chi_\beta)^2, \tag{C.54}$$

$$\eta(\theta, \mu) = \frac{\mu \theta}{1 + \mu \theta}, \tag{C.55}$$

$$\begin{aligned}
 \epsilon(\alpha, \beta) = & (\alpha + \beta) \left( e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \right. \\
 & - \frac{e(r(\alpha, \beta), T_3, U_3, V_3, W_3, X_3, Y_3, P_3) \omega(\zeta(\alpha, \beta)) (1 - (\zeta(\alpha, \beta))^4)}{c} \\
 & + (e(r(\alpha, \beta), T_2, U_2, V_2, W_2, X_2, Y_2, P_2) - e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1)) \\
 & \left. \omega(\zeta(\alpha, \beta)) (\zeta(\alpha, \beta))^4 \right),
 \end{aligned} \tag{C.56}$$

$$r(\alpha, \beta) = 1/4 \sqrt[3]{34}^{2/3} \sqrt[3]{\frac{1}{\pi(\alpha + \beta)}}, \tag{C.57}$$

$$\zeta(\alpha, \beta) = \frac{\alpha - \beta}{\alpha + \beta}, \tag{C.58}$$

$$\omega(z) = \frac{(1+z)^{4/3} + (1-z)^{4/3} - 2}{2 \sqrt[3]{2} - 2}, \tag{C.59}$$

$$\begin{aligned}
 e(r, t, u, v, w, x, y, p) = & -2t(1 + ur) \\
 & \ln \left( 1 + 1/2 \frac{1}{t(v\sqrt{r} + wr + xr^{3/2} + yr^{p+1})} \right),
 \end{aligned} \tag{C.60}$$

$$c = 1.709921, \tag{C.61}$$

$$T = [0.031091, 0.015545, 0.016887], \quad (\text{C.62})$$

$$U = [0.21370, 0.20548, 0.11125], \quad (\text{C.63})$$

$$V = [7.5957, 14.1189, 10.357], \quad (\text{C.64})$$

$$W = [3.5876, 6.1977, 3.6231], \quad (\text{C.65})$$

$$X = [1.6382, 3.3662, 0.88026], \quad (\text{C.66})$$

$$Y = [0.49294, 0.62517, 0.49671], \quad (\text{C.67})$$

$$P = [1, 1, 1], \quad (\text{C.68})$$

$$A = [0.9454, 0.7471, -4.5961], \quad (\text{C.69})$$

$$B = [0.1737, 2.3487, -2.4868], \quad (\text{C.70})$$

$$C = [0.8094, 0.5073, 0.7481], \quad (\text{C.71})$$

$$\lambda = [0.006, 0.2, 0.004]. \quad (\text{C.72})$$

为了避免极限  $\rho_{\bar{s}} \rightarrow 0$  的奇异性,

$$\begin{aligned} G = & \epsilon(\rho_s, 0) \left( B_0 + B_1 \eta((\chi_s)^2, \lambda_2) + B_2 (\eta((\chi_s)^2, \lambda_2))^2 \right) \\ & - 3/8 \sqrt[3]{34^{2/3}} \sqrt[3]{\pi^{-1}} (\rho_s)^{4/3} \left( C_0 + C_1 \eta((\chi_s)^2, \lambda_3) + C_2 (\eta((\chi_s)^2, \lambda_3))^2 \right). \end{aligned} \quad (\text{C.73})$$

### C.8 B97RDF: Hamprecht等人重新参数化B97的密度泛函部分

Hamprecht等人在自洽流程中重新参数化的B97泛函。这个泛函需要与0.21 倍的精确交换相混合。更多细节参见文献 [11]。

$$\begin{aligned} K = & (\epsilon(\rho_\alpha, \rho_\beta) - \epsilon(\rho_\alpha, 0) - \epsilon(\rho_\beta, 0)) (A_0 + A_1 \eta(d, \lambda_1) + A_2 (\eta(d, \lambda_1))^2) \\ & + \sum_s \epsilon(\rho_s, 0) \left( B_0 + B_1 \eta((\chi_s)^2, \lambda_2) + B_2 (\eta((\chi_s)^2, \lambda_2))^2 \right) \\ & - 3/8 \sqrt[3]{34^{2/3}} \sqrt[3]{\pi^{-1}} (\rho_s)^{4/3} \left( C_0 + C_1 \eta((\chi_s)^2, \lambda_3) + C_2 (\eta((\chi_s)^2, \lambda_3))^2 \right), \end{aligned} \quad (\text{C.74})$$

其中,

$$d = 1/2 (\chi_\alpha)^2 + 1/2 (\chi_\beta)^2, \quad (\text{C.75})$$

$$\eta(\theta, \mu) = \frac{\mu \theta}{1 + \mu \theta}, \quad (\text{C.76})$$

$$\epsilon(\alpha, \beta) = (\alpha + \beta) \left( e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \right. \\ \left. - \frac{e(r(\alpha, \beta), T_3, U_3, V_3, W_3, X_3, Y_3, P_3) \omega(\zeta(\alpha, \beta)) (1 - (\zeta(\alpha, \beta))^4)}{c} \right. \\ \left. + \left( e(r(\alpha, \beta), T_2, U_2, V_2, W_2, X_2, Y_2, P_2) - e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \right) \right) \quad (C.77)$$

$$\omega(\zeta(\alpha, \beta)) (\zeta(\alpha, \beta))^4 \Big), \\ r(\alpha, \beta) = 1/4 \sqrt[3]{34^{2/3}} \sqrt[3]{\frac{1}{\pi(\alpha + \beta)}}, \quad (C.78)$$

$$\zeta(\alpha, \beta) = \frac{\alpha - \beta}{\alpha + \beta}, \quad (C.79)$$

$$\omega(z) = \frac{(1+z)^{4/3} + (1-z)^{4/3} - 2}{2\sqrt[3]{2} - 2}, \quad (C.80)$$

$$e(r, t, u, v, w, x, y, p) = -2t(1 + ur) \\ \ln \left( 1 + 1/2 \frac{1}{t(v\sqrt{r} + wr + xr^{3/2} + yr^{p+1})} \right), \quad (C.81)$$

$$c = 1.709921, \quad (C.82)$$

$$T = [0.031091, 0.015545, 0.016887], \quad (C.83)$$

$$U = [0.21370, 0.20548, 0.11125], \quad (C.84)$$

$$V = [7.5957, 14.1189, 10.357], \quad (C.85)$$

$$W = [3.5876, 6.1977, 3.6231], \quad (C.86)$$

$$X = [1.6382, 3.3662, 0.88026], \quad (C.87)$$

$$Y = [0.49294, 0.62517, 0.49671], \quad (C.88)$$

$$P = [1, 1, 1], \quad (C.89)$$

$$A = [0.955689, 0.788552, -5.47869], \quad (C.90)$$

$$B = [0.0820011, 2.71681, -2.87103], \quad (C.91)$$

$$C = [0.789518, 0.573805, 0.660975], \quad (C.92)$$

$$\lambda = [0.006, 0.2, 0.004]. \quad (C.93)$$

为了避免极限  $\rho_s \rightarrow 0$  的奇异性,

$$G = \epsilon(\rho_s, 0) \left( B_0 + B_1 \eta((\chi_s)^2, \lambda_2) + B_2 (\eta((\chi_s)^2, \lambda_2))^2 \right) \\ - 3/8 \sqrt[3]{34^{2/3}} \sqrt[3]{\pi^{-1}} (\rho_s)^{4/3} \left( C_0 + C_1 \eta((\chi_s)^2, \lambda_3) + C_2 (\eta((\chi_s)^2, \lambda_3))^2 \right). \quad (C.94)$$



### C.9 BR: Becke-Roussel交换泛函

A. D. Becke and M. R. Roussel, Phys. Rev. A 39, 3761 (1989)

$$K = \frac{1}{2} \sum_s \rho_s U_s, \quad (\text{C.95})$$

其中,

$$U_s = -(1 - e^{-x} - x e^{-x}/2)/b, \quad (\text{C.96})$$

$$b = \frac{x^3 e^{-x}}{8\pi\rho_s} \quad (\text{C.97})$$

$x$ 由非线性方程定义

$$\frac{x e^{-2x/3}}{x - 2} = \frac{2\pi^{2/3} \rho_s^{5/3}}{3Q_s}, \quad (\text{C.98})$$

其中,

$$Q_s = (v_s - 2\gamma D_s)/6, \quad (\text{C.99})$$

$$D_s = \tau_s - \frac{\sigma_{ss}}{4\rho_s}, \quad (\text{C.100})$$

$$\gamma = 1. \quad (\text{C.101})$$

### C.10 BRUEG: Becke-Roussel交换泛函—均匀电子气极限

A. D. Becke and M. R. Roussel, Phys. Rev. A 39, 3761 (1989)

同BR, 但 $\gamma = 0.8$ 。

### C.11 BW: Becke-Wigner交换-关联泛函

组合了Becke的1998交换和Wigner的自旋极化关联泛函的杂化交换-关联泛函。更多细节参见文献 [19]。

$$K = -4 c \rho_{\alpha} \rho_{\beta} \rho^{-1} \left(1 + \frac{d}{\sqrt[3]{\rho}}\right)^{-1} + \sum_s \alpha (\rho_s)^{4/3} - \frac{\beta (\rho_s)^{4/3} (\chi_s)^2}{1 + 6 \beta \chi_s \operatorname{arcsinh}(\chi_s)}, \quad (\text{C.102})$$

其中,

$$\alpha = -3/8 \sqrt[3]{34}^{2/3} \sqrt[3]{\pi^{-1}}, \quad (\text{C.103})$$

$$\beta = 0.0042, \quad (\text{C.104})$$

$$c = 0.04918, \quad (\text{C.105})$$

$$d = 0.349. \quad (\text{C.106})$$

为了避免极限  $\rho_{\bar{s}} \rightarrow 0$  的奇异性,

$$G = \alpha (\rho_s)^{4/3} - \frac{\beta (\rho_s)^{4/3} (\chi_s)^2}{1 + 6 \beta \chi_s \operatorname{arcsinh}(\chi_s)}. \quad (\text{C.107})$$

## C.12 CS1: Colle-Salvetti关联泛函

R. Colle and O. Salvetti, *Theor. Chim. Acta* 37, 329 (1974); C. Lee, W. Yang and R. G. Parr, *Phys. Rev. B* 37, 785(1988)

CS1形式上等同于CS2, 但重新公式化, 其中包含 $v$ 的项从分步积分中去除。这使得泛函的求解更经济。在精确积分的极限, CS1 和CS2相同, 但在使用有限积分网格时会出现小的数值偏差。

## C.13 CS2: Colle-Salvetti关联泛函

R. Colle and O. Salvetti, *Theor. Chim. Acta* 37, 329 (1974); C. Lee, W. Yang and R. G. Parr, *Phys. Rev. B* 37, 785(1988)

CS2定义为

$$K = -a \left( \frac{\rho + 2b\rho^{-5/3} [\rho_{\alpha} t_{\alpha} + \rho_{\beta} t_{\beta} - \rho t_W] e^{-c\rho^{-1/3}}}{1 + d\rho^{-1/3}} \right) \quad (\text{C.108})$$

其中,

$$t_\alpha = \frac{\tau_\alpha}{2} - \frac{v_\alpha}{8} \quad (\text{C.109})$$

$$t_\beta = \frac{\tau_\beta}{2} - \frac{v_\beta}{8} \quad (\text{C.110})$$

$$t_W = \frac{1}{8} \frac{\sigma}{\rho} - \frac{1}{2} v \quad (\text{C.111})$$

其中的常数为  $a = 0.04918$ ,  $b = 0.132$ ,  $c = 0.2533$ ,  $d = 0.349$ 。

#### C.14 DIRAC: Slater-Dirac交换能量

自动产生的Slater-Dirac交换。更多细节参见文献 [18]。

$$K = \sum_s -c(\rho_s)^{4/3}, \quad (\text{C.112})$$

其中,

$$c = 3/8 \sqrt[3]{34^{2/3} \sqrt[3]{\pi^{-1}}}. \quad (\text{C.113})$$

#### C.15 ECERF: 短程LDA关联泛函

关联能的局域密度近似, 用于短程分子间相互作用  $\text{erf}(\mu r_{12})/r_{12}$ 。

S. Paziani, S. Moroni, P. Gori-Giorgi, and G. B. Bachelet, Phys. Rev. B 73, 155111 (2006).

$$\begin{aligned} \epsilon_c^{\text{SR}}(r_s, \zeta, \mu) = & \epsilon_c^{\text{PW92}}(r_s, \zeta) \\ & - \frac{[\phi_2(\zeta)]^3 Q\left(\frac{\mu\sqrt{r_s}}{\phi_2(\zeta)}\right) + a_1\mu^3 + a_2\mu^4 + a_3\mu^5 + a_4\mu^6 + a_5\mu^8}{(1 + b_0^2\mu^2)^4}, \end{aligned} \quad (\text{C.114})$$

其中,

$$Q(x) = \frac{2\ln(2) - 2}{\pi^2} \ln\left(\frac{1 + ax + bx^2 + cx^3}{1 + ax + dx^2}\right), \quad (\text{C.115})$$

以及  $a = 5.84605$ ,  $c = 3.91744$ ,  $d = 3.44851$ ,  $b = d - 3\pi\alpha/(4\ln(2) - 4)$ 。参数  $a_i(r_s, \zeta)$  为  $a_1 = 4b_0^6 C_3 + b_0^8 C_5$ ,

$$a_2 = 4b_0^6 C_2 + b_0^8 C_4 + 6b_0^4 \epsilon_c^{\text{PW92}},$$

$$a_3 = b_0^8 C_3,$$

$$a_4 = b_0^8 C_2 + 4b_0^6 \epsilon_c^{\text{PW92}},$$

$$a_5 = b_0^8 \epsilon_c^{\text{PW92}},$$

以及

$$C_2 = -\frac{3(1-\zeta^2)g_c(0, r_s, \zeta=0)}{8r_s^3},$$

$$C_3 = -(1-\zeta^2)\frac{g(0, r_s, \zeta=0)}{\sqrt{2\pi}r_s^3},$$

$$C_4 = -\frac{9c_4(r_s, \zeta)}{64r_s^3},$$

$$C_5 = -\frac{9c_5(r_s, \zeta)}{40\sqrt{2\pi}r_s^3},$$

$$\begin{aligned} c_4(r_s, \zeta) = & \left(\frac{1+\zeta}{2}\right)^2 g''\left(0, r_s \left(\frac{2}{1+\zeta}\right)^{1/3}, \zeta=1\right) + \left(\frac{1-\zeta}{2}\right)^2 \times g''\left(0, r_s \left(\frac{2}{1-\zeta}\right)^{1/3}, \zeta=1\right) \\ & + (1-\zeta^2)D_2(r_s) - \frac{\phi_8(\zeta)}{5\alpha^2 r_s^2}, \end{aligned}$$

$$\begin{aligned} c_4(r_s, \zeta) = & \left(\frac{1+\zeta}{2}\right)^2 g''\left(0, r_s \left(\frac{2}{1+\zeta}\right)^{1/3}, \zeta=1\right) + \left(\frac{1-\zeta}{2}\right)^2 \times g''\left(0, r_s \left(\frac{2}{1-\zeta}\right)^{1/3}, \zeta=1\right) \\ & + (1-\zeta^2)D_3(r_s), \end{aligned}$$

$$b_0(r_s) = 0.784949r_s,$$

$$g''(0, r_s, \zeta=1) = \frac{2^{5/3}}{5\alpha^2 r_s^2} \frac{1 - 0.02267r_s}{(1 + 0.4319r_s + 0.04r_s^2)},$$

$$D_2(r_s) = \frac{e^{-0.547r_s}}{r_s^2} (-0.388r_s + 0.676r_s^2),$$

$$D_3(r_s) = \frac{e^{-0.31r_s}}{r_s^3} (-4.95r_s + r_s^2).$$

最后,  $\epsilon_c^{\text{PW92}}(r_s, \zeta)$  是Perdew-Wang参数化的标准均匀电子气关联能[J.P. Perdew and Y. Wang, Phys. Rev. B 45, 13244 (1992)],

$$g(0, r_s, \zeta=0) = \frac{1}{2}(1 - Br_s + Cr_s^2 + Dr_s^3 + Er_s^4)e^{-dr_s}, \quad (\text{C.116})$$

是标准理想金属模型 (Jellium model) 的最上层电子对配分函数[P. Gori-Giorgi and

J.P. Perdew, Phys. Rev. B 64, 155102 (2001)], 其中  $B = -0.0207$ ,  $C = 0.08193$ ,  $D = -0.01277$ ,  $E = 0.001859$ ,  $d = 0.7524$ 。最上层电子对配分函数的关联部分是  $g_c(0, r_s, \zeta=0) = g(0, r_s, \zeta=0) - \frac{1}{2}$ 。

### C.16 EXERF: 短程LDA交换泛函

交换能的局域密度近似, 用于短程分子间相互作用  $\text{erf}(\mu r_{12})/r_{12}$ 。

A. Savin, in Recent Developments and Applications of Modern Density Functional Theory, edited by J.M. Seminario (Elsevier, Amsterdam, 1996).

$$\begin{aligned} \epsilon_x^{\text{SR}}(r_s, \zeta, \mu) = & \frac{3}{4\pi} \frac{\phi_4(\zeta)}{\alpha r_s} - \frac{1}{2} (1+\zeta)^{4/3} f_x(r_s, \mu(1+\zeta)^{-1/3}) \\ & + \frac{1}{2} (1-\zeta)^{4/3} f_x(r_s, \mu(1-\zeta)^{-1/3}), \end{aligned} \quad (\text{C.117})$$

其中

$$\phi_n(\zeta) = \frac{1}{2} [(1+\zeta)^{n/3} + (1-\zeta)^{n/3}], \quad (\text{C.118})$$

$$f_x(r_s, \mu) = -\frac{\mu}{\pi} \left[ (2y - 4y^3) e^{-1/4y^2} - 3y - 4y^4 + \sqrt{\pi} \text{erf}\left(\frac{1}{2y}\right) \right], \quad (\text{C.119})$$

$$y = \frac{\mu \alpha r_s}{2}, \quad (\text{C.120})$$

以及  $\alpha = (4/9\pi)^{1/3}$ 。

### C.17 G96: Gill的1996梯度校正交换泛函

更多细节参见文献 [10]。

$$K = \sum_s (\rho_s)^{4/3} \left( \alpha - \frac{1}{137} (\chi_s)^{3/2} \right), \quad (\text{C.121})$$

其中,

$$\alpha = -3/8 \sqrt[3]{34^{2/3} \sqrt[3]{\pi^{-1}}}. \quad (\text{C.122})$$

为了避免极限  $\rho_{\bar{s}} \rightarrow 0$  的奇异性,

$$G = (\rho_s)^{4/3} \left( \alpha - \frac{1}{137} (\chi_s)^{3/2} \right). \quad (\text{C.123})$$

### C.18 HCTH120: Handy最小二乘拟合泛函

更多细节参见文献 [8]。

$$\begin{aligned} K = & (\epsilon(\rho_\alpha, \rho_\beta) - \epsilon(\rho_\alpha, 0) - \epsilon(\rho_\beta, 0)) \\ & (A_0 + A_1 \eta(d, \lambda_1) + A_2 (\eta(d, \lambda_1))^2 + A_3 (\eta(d, \lambda_1))^3 + A_4 (\eta(d, \lambda_1))^4) + \sum_s \epsilon(\rho_s, 0) \\ & \left( B_0 + B_1 \eta((\chi_s)^2, \lambda_2) + B_2 (\eta((\chi_s)^2, \lambda_2))^2 + B_3 (\eta((\chi_s)^2, \lambda_2))^3 \right. \\ & \left. + B_4 (\eta((\chi_s)^2, \lambda_2))^4 \right) - 3/8 \sqrt[3]{34^{2/3} \sqrt[3]{\pi^{-1}}} (\rho_s)^{4/3} \\ & \left( C_0 + C_1 \eta((\chi_s)^2, \lambda_3) + C_2 (\eta((\chi_s)^2, \lambda_3))^2 + C_3 (\eta((\chi_s)^2, \lambda_3))^3 + C_4 (\eta((\chi_s)^2, \lambda_3))^4 \right), \end{aligned} \quad (\text{C.124})$$

其中,

$$d = 1/2 (\chi_\alpha)^2 + 1/2 (\chi_\beta)^2, \quad (\text{C.125})$$

$$\eta(\theta, \mu) = \frac{\mu \theta}{1 + \mu \theta}, \quad (\text{C.126})$$

$$\begin{aligned} \epsilon(\alpha, \beta) = & (\alpha + \beta) \left( e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \right. \\ & \left. - \frac{e(r(\alpha, \beta), T_3, U_3, V_3, W_3, X_3, Y_3, P_3) \omega(\zeta(\alpha, \beta)) (1 - (\zeta(\alpha, \beta))^4)}{c} \right. \\ & \left. + \left( e(r(\alpha, \beta), T_2, U_2, V_2, W_2, X_2, Y_2, P_2) - e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \right) \right) \\ & \omega(\zeta(\alpha, \beta)) (\zeta(\alpha, \beta))^4 \Big), \end{aligned} \quad (\text{C.127})$$

$$r(\alpha, \beta) = 1/4 \sqrt[3]{34^{2/3}} \sqrt[3]{\frac{1}{\pi(\alpha + \beta)}}, \quad (\text{C.128})$$

$$\zeta(\alpha, \beta) = \frac{\alpha - \beta}{\alpha + \beta}, \quad (\text{C.129})$$

$$\omega(z) = \frac{(1+z)^{4/3} + (1-z)^{4/3} - 2}{2\sqrt[3]{2} - 2}, \quad (\text{C.130})$$

$$e(r, t, u, v, w, x, y, p) = -2t(1 + ur) \ln \left( 1 + 1/2 \frac{1}{t(v\sqrt{r} + wr + xr^{3/2} + yr^{p+1})} \right), \quad (\text{C.131})$$

$$c = 1.709921, \quad (\text{C.132})$$

$$T = [0.031091, 0.015545, 0.016887], \quad (\text{C.133})$$

$$U = [0.21370, 0.20548, 0.11125], \quad (\text{C.134})$$

$$V = [7.5957, 14.1189, 10.357], \quad (\text{C.135})$$

$$W = [3.5876, 6.1977, 3.6231], \quad (\text{C.136})$$

$$X = [1.6382, 3.3662, 0.88026], \quad (\text{C.137})$$

$$Y = [0.49294, 0.62517, 0.49671], \quad (\text{C.138})$$

$$P = [1, 1, 1], \quad (\text{C.139})$$

$$A = [0.51473, 6.9298, -24.707, 23.110, -11.323], \quad (\text{C.140})$$

$$B = [0.48951, -0.2607, 0.4329, -1.9925, 2.4853], \quad (\text{C.141})$$

$$C = [1.09163, -0.7472, 5.0783, -4.1075, 1.1717], \quad (\text{C.142})$$

$$\lambda = [0.006, 0.2, 0.004]. \quad (\text{C.143})$$

## C.19 HCTH147: Handy最小二乘拟合泛函

更多细节参见文献 [8]。

$$\begin{aligned}
K = & (\epsilon(\rho_\alpha, \rho_\beta) - \epsilon(\rho_\alpha, 0) - \epsilon(\rho_\beta, 0)) \\
& \left( A_0 + A_1 \eta(d, \lambda_1) + A_2 (\eta(d, \lambda_1))^2 + A_3 (\eta(d, \lambda_1))^3 + A_4 (\eta(d, \lambda_1))^4 \right) + \sum_s \epsilon(\rho_s, 0) \\
& \left( B_0 + B_1 \eta((\chi_s)^2, \lambda_2) + B_2 (\eta((\chi_s)^2, \lambda_2))^2 + B_3 (\eta((\chi_s)^2, \lambda_2))^3 \right. \\
& \left. + B_4 (\eta((\chi_s)^2, \lambda_2))^4 \right) - 3/8 \sqrt[3]{34}^{2/3} \sqrt[3]{\pi^{-1}} (\rho_s)^{4/3} \\
& \left( C_0 + C_1 \eta((\chi_s)^2, \lambda_3) + C_2 (\eta((\chi_s)^2, \lambda_3))^2 + C_3 (\eta((\chi_s)^2, \lambda_3))^3 + C_4 (\eta((\chi_s)^2, \lambda_3))^4 \right),
\end{aligned} \tag{C.144}$$

其中,

$$d = 1/2 (\chi_\alpha)^2 + 1/2 (\chi_\beta)^2, \tag{C.145}$$

$$\eta(\theta, \mu) = \frac{\mu \theta}{1 + \mu \theta}, \tag{C.146}$$

$$\begin{aligned}
\epsilon(\alpha, \beta) = & (\alpha + \beta) \left( e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \right. \\
& - \frac{e(r(\alpha, \beta), T_3, U_3, V_3, W_3, X_3, Y_3, P_3) \omega(\zeta(\alpha, \beta)) (1 - (\zeta(\alpha, \beta))^4)}{c} \\
& \left. + \left( e(r(\alpha, \beta), T_2, U_2, V_2, W_2, X_2, Y_2, P_2) - e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \right) \right) \\
& \omega(\zeta(\alpha, \beta)) (\zeta(\alpha, \beta))^4 \Big),
\end{aligned} \tag{C.147}$$

$$r(\alpha, \beta) = 1/4 \sqrt[3]{34}^{2/3} \sqrt[3]{\frac{1}{\pi(\alpha + \beta)}}, \tag{C.148}$$

$$\zeta(\alpha, \beta) = \frac{\alpha - \beta}{\alpha + \beta}, \tag{C.149}$$

$$\omega(z) = \frac{(1+z)^{4/3} + (1-z)^{4/3} - 2}{2 \sqrt[3]{2} - 2}, \tag{C.150}$$

$$\begin{aligned}
e(r, t, u, v, w, x, y, p) = & -2t(1 + ur) \\
& \ln \left( 1 + 1/2 \frac{1}{t(v\sqrt{r} + wr + xr^{3/2} + yr^{p+1})} \right),
\end{aligned} \tag{C.151}$$



$$c = 1.709921, \quad (\text{C.152})$$

$$T = [0.031091, 0.015545, 0.016887], \quad (\text{C.153})$$

$$U = [0.21370, 0.20548, 0.11125], \quad (\text{C.154})$$

$$V = [7.5957, 14.1189, 10.357], \quad (\text{C.155})$$

$$W = [3.5876, 6.1977, 3.6231], \quad (\text{C.156})$$

$$X = [1.6382, 3.3662, 0.88026], \quad (\text{C.157})$$

$$Y = [0.49294, 0.62517, 0.49671], \quad (\text{C.158})$$

$$P = [1, 1, 1], \quad (\text{C.159})$$

$$A = [0.54235, 7.0146, -28.382, 35.033, -20.428], \quad (\text{C.160})$$

$$B = [0.56258, -0.0171, -1.3064, 1.0575, 0.8854], \quad (\text{C.161})$$

$$C = [1.09025, -0.7992, 5.5721, -5.8676, 3.0454], \quad (\text{C.162})$$

$$\lambda = [0.006, 0.2, 0.004]. \quad (\text{C.163})$$

## C.20 HCTH93: Handy最小二乘拟合泛函

更多细节参见文献 [11]。

$$\begin{aligned} K = & (\epsilon(\rho_\alpha, \rho_\beta) - \epsilon(\rho_\alpha, 0) - \epsilon(\rho_\beta, 0)) \\ & \left( A_0 + A_1 \eta(d, \lambda_1) + A_2 (\eta(d, \lambda_1))^2 + A_3 (\eta(d, \lambda_1))^3 + A_4 (\eta(d, \lambda_1))^4 \right) + \sum_s \epsilon(\rho_s, 0) \\ & \left( B_0 + B_1 \eta((\chi_s)^2, \lambda_2) + B_2 (\eta((\chi_s)^2, \lambda_2))^2 + B_3 (\eta((\chi_s)^2, \lambda_2))^3 \right. \\ & \left. + B_4 (\eta((\chi_s)^2, \lambda_2))^4 \right) - 3/8 \sqrt[3]{34}^{2/3} \sqrt[3]{\pi^{-1}} (\rho_s)^{4/3} \\ & \left( C_0 + C_1 \eta((\chi_s)^2, \lambda_3) + C_2 (\eta((\chi_s)^2, \lambda_3))^2 + C_3 (\eta((\chi_s)^2, \lambda_3))^3 + C_4 (\eta((\chi_s)^2, \lambda_3))^4 \right), \end{aligned} \quad (\text{C.164})$$

其中,

$$d = 1/2 (\chi_\alpha)^2 + 1/2 (\chi_\beta)^2, \quad (\text{C.165})$$

$$\eta(\theta, \mu) = \frac{\mu \theta}{1 + \mu \theta}, \quad (\text{C.166})$$

$$\begin{aligned} \epsilon(\alpha, \beta) = (\alpha + \beta) & \left( e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \right. \\ & - \frac{e(r(\alpha, \beta), T_3, U_3, V_3, W_3, X_3, Y_3, P_3) \omega(\zeta(\alpha, \beta)) (1 - (\zeta(\alpha, \beta))^4)}{c} \\ & \left. + \left( e(r(\alpha, \beta), T_2, U_2, V_2, W_2, X_2, Y_2, P_2) - e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \right) \right) \\ & \omega(\zeta(\alpha, \beta)) (\zeta(\alpha, \beta))^4, \end{aligned} \quad (\text{C.167})$$

$$r(\alpha, \beta) = 1/4 \sqrt[3]{34^{2/3}} \sqrt[3]{\frac{1}{\pi(\alpha + \beta)}}, \quad (\text{C.168})$$

$$\zeta(\alpha, \beta) = \frac{\alpha - \beta}{\alpha + \beta}, \quad (\text{C.169})$$

$$\omega(z) = \frac{(1+z)^{4/3} + (1-z)^{4/3} - 2}{2 \sqrt[3]{2} - 2}, \quad (\text{C.170})$$

$$\begin{aligned} e(r, t, u, v, w, x, y, p) = -2t(1 + ur) \\ \ln \left( 1 + 1/2 \frac{1}{t(v\sqrt{r} + wr + xr^{3/2} + yr^{p+1})} \right), \end{aligned} \quad (\text{C.171})$$

$$c = 1.709921, \quad (\text{C.172})$$

$$T = [0.031091, 0.015545, 0.016887], \quad (\text{C.173})$$

$$U = [0.21370, 0.20548, 0.11125], \quad (\text{C.174})$$

$$V = [7.5957, 14.1189, 10.357], \quad (\text{C.175})$$

$$W = [3.5876, 6.1977, 3.6231], \quad (\text{C.176})$$

$$X = [1.6382, 3.3662, 0.88026], \quad (\text{C.177})$$

$$Y = [0.49294, 0.62517, 0.49671], \quad (\text{C.178})$$

$$P = [1, 1, 1], \quad (\text{C.179})$$

$$A = [0.72997, 3.35287, -11.543, 8.08564, -4.47857], \quad (\text{C.180})$$

$$B = [0.222601, -0.0338622, -0.012517, -0.802496, 1.55396], \quad (\text{C.181})$$

$$C = [1.0932, -0.744056, 5.5992, -6.78549, 4.49357], \quad (\text{C.182})$$

$$\lambda = [0.006, 0.2, 0.004]. \quad (\text{C.183})$$

## C.21 LTA: 局域 $\tau$ 近似

密度为 $\tau$ 函数的LSDA交换泛函。更多细节参见文献 [9]。

$$K = \sum_s 1/2 E(2\tau_s), \quad (\text{C.184})$$

其中,

$$E(\alpha) = 1/9 c 5^{4/5} \sqrt[5]{9} \left( \frac{\alpha \sqrt[3]{3}}{(\pi^2)^{2/3}} \right)^{4/5}, \quad (\text{C.185})$$

$$c = -3/4 \sqrt[3]{3} \sqrt[3]{\pi^{-1}}. \quad (\text{C.186})$$

为了避免极限  $\rho_{\bar{s}} \rightarrow 0$  的奇异性,

$$G = 1/2 E(2\tau_s). \quad (\text{C.187})$$

## C.22 LYP: Lee, Yang和Parr的关联泛函

C. Lee, W. Yang and R. G. Parr, Phys. Rev. B 37, 785(1988); B. Miehlich, A. Savin, H. Stoll and H. Preuss, Chem. Phys. Letters 157, 200 (1989)

$$\begin{aligned} K &= 4 \frac{A \rho_\alpha \rho_\beta Z}{\rho} + AB\omega \sigma (\rho_\alpha \rho_\beta (47 - 7\delta) / 18 - 2\rho^2/3) \\ &+ \sum_s AB\omega \left( \rho_s \rho_{\bar{s}} \left( 8 \cdot 2^{2/3} e \rho_s^{8/3} - (5/2 - \delta/18) \sigma_{ss} - \frac{(\delta - 11) \rho_s \sigma_{ss}}{9\rho} \right) \right. \\ &\left. + (2\rho^2/3 - \rho_s^2) \sigma_{\bar{s}\bar{s}} \right), \end{aligned} \quad (\text{C.188})$$

其中,

$$\omega = e^{-\frac{c}{\rho^{1/3}}} Z \rho^{-11/3}, \quad (\text{C.189})$$

$$\delta = \frac{c + dZ}{\rho^{1/3}}, \quad (\text{C.190})$$

$$B = 0.04918, \quad (\text{C.191})$$

$$A = 0.132, \quad (\text{C.192})$$

$$c = 0.2533, \quad (\text{C.193})$$

$$d = 0.349, \quad (\text{C.194})$$

$$e = \frac{3}{10} (3\pi^2)^{2/3} \quad (\text{C.195})$$

$$Z = \left(1 + \frac{d}{\rho^{1/3}}\right)^{-1}. \quad (\text{C.196})$$

### C.23 MK00: 用于准确虚轨道能量的交换泛函

更多细节参见文献 [12]。

$$K = \sum_s -3 \frac{\pi (\rho_s)^3}{\tau_s - 1/4 v_s}. \quad (\text{C.197})$$

### C.24 MK00B: 用于准确虚轨道能量的交换泛函

MK00具有B88X形式的梯度校正，但有不同的经验参数。更多细节参见文献 [12]。

$$K = \sum_s -3 \frac{\pi (\rho_s)^3}{\tau_s - 1/4 v_s} - \frac{\beta (\rho_s)^{4/3} (\chi_s)^2}{1 + 6 \beta \chi_s \operatorname{arcsinh} (\chi_s)}, \quad (\text{C.198})$$

其中

$$\beta = 0.0016. \quad (\text{C.199})$$

为了避免极限  $\rho_s \rightarrow 0$  的奇异性，

$$G = -3 \frac{\pi (\rho_s)^3}{\tau_s - 1/4 v_s} - \frac{\beta (\rho_s)^{4/3} (\chi_s)^2}{1 + 6 \beta \chi_s \operatorname{arcsinh} (\chi_s)}. \quad (\text{C.200})$$

### C.25 P86:

VWN的梯度校正。更多细节参见文献 [13]。

$$K = \rho e + \frac{e^{-\Phi} C(r) \sigma}{d\rho^{4/3}}, \quad (\text{C.201})$$

其中，

$$r = 1/4 \sqrt[3]{34^{2/3}} \sqrt[3]{\frac{1}{\pi \rho}}, \quad (\text{C.202})$$

$$x = \sqrt{r}, \quad (\text{C.203})$$

$$\zeta = \frac{\rho_\alpha - \rho_\beta}{\rho}, \quad (\text{C.204})$$

$$e = \Lambda + \omega y (1 + h\zeta^4), \quad (\text{C.205})$$

$$y = \frac{9}{8} (1 + \zeta)^{4/3} + \frac{9}{8} (1 - \zeta)^{4/3} - 9/4, \quad (\text{C.206})$$

$$h = 4/9 \frac{\lambda - \Lambda}{(\sqrt[3]{2} - 1) \omega} - 1, \quad (\text{C.207})$$

$$\Lambda = q(k_1, l_1, m_1, n_1), \quad (\text{C.208})$$

$$\lambda = q(k_2, l_2, m_2, n_2), \quad (\text{C.209})$$

$$\omega = q(k_3, l_3, m_3, n_3), \quad (\text{C.210})$$

$$\begin{aligned} q(A, p, c, d) = & A \left( \ln \left( \frac{x^2}{X(x, c, d)} \right) + 2c \arctan \left( \frac{Q(c, d)}{2x + c} \right) (Q(c, d))^{-1} \right. \\ & - cp \left( \ln \left( \frac{(x - p)^2}{X(x, c, d)} \right) + 2(c + 2p) \arctan \left( \frac{Q(c, d)}{2x + c} \right) (Q(c, d))^{-1} \right. \\ & \left. \left. (X(p, c, d))^{-1} \right), \right. \end{aligned} \quad (\text{C.211})$$

$$Q(c, d) = \sqrt{4d - c^2}, \quad (\text{C.212})$$

$$X(i, c, d) = i^2 + ci + d, \quad (\text{C.213})$$

$$\Phi = 0.007390075 \frac{z\sqrt{\sigma}}{C(r)\rho^{7/6}}, \quad (\text{C.214})$$

$$d = \sqrt[3]{2} \sqrt{(1/2 + 1/2\zeta)^{5/3} + (1/2 - 1/2\zeta)^{5/3}}, \quad (\text{C.215})$$

$$C(r) = 0.001667 + \frac{0.002568 + \alpha r + \beta r^2}{1 + \xi r + \delta r^2 + 10000 \beta r^3}, \quad (\text{C.216})$$

$$z = 0.11, \quad (\text{C.217})$$

$$\alpha = 0.023266, \quad (\text{C.218})$$

$$\beta = 0.000007389, \quad (\text{C.219})$$

$$\xi = 8.723, \quad (\text{C.220})$$

$$\delta = 0.472, \quad (\text{C.221})$$

$$k = [0.0310907, 0.01554535, -1/6 \pi^{-2}], \quad (\text{C.222})$$

$$l = [-0.10498, -0.325, -0.0047584], \quad (\text{C.223})$$

$$m = [3.72744, 7.06042, 1.13107], \quad (\text{C.224})$$

$$n = [12.9352, 18.0578, 13.0045]. \quad (\text{C.225})$$

## C.26 PBEC: PBE关联泛函

更多细节参见文献 [14]。

$$K = \rho \left( \epsilon(\rho_\alpha, \rho_\beta) + H(d, \rho_\alpha, \rho_\beta) \right), \quad (\text{C.226})$$

其中,

$$d = 1/12 \frac{\sqrt{\sigma} 3^{5/6}}{u(\rho_\alpha, \rho_\beta) \sqrt[6]{\pi^{-1}} \rho^{7/6}}, \quad (\text{C.227})$$

$$u(\alpha, \beta) = 1/2 (1 + \zeta(\alpha, \beta))^{2/3} + 1/2 (1 - \zeta(\alpha, \beta))^{2/3}, \quad (\text{C.228})$$

$$H(d, \alpha, \beta) = 1/2 (u(\rho_\alpha, \rho_\beta))^3 \lambda^2$$

$$\ln \left( 1 + 2 \frac{\iota (d^2 + A(\alpha, \beta) d^4)}{\lambda (1 + A(\alpha, \beta) d^2 + (A(\alpha, \beta))^2 d^4)} \right) \iota^{-1}, \quad (\text{C.229})$$

$$A(\alpha, \beta) = 2 \iota \lambda^{-1} \left( e^{-2 \frac{\iota \epsilon(\alpha, \beta)}{(u(\rho_\alpha, \rho_\beta))^3 \lambda^2}} - 1 \right)^{-1}, \quad (\text{C.230})$$

$$\iota = 0.0716, \quad (\text{C.231})$$

$$\lambda = \nu \kappa, \quad (\text{C.232})$$

$$\nu = 16 \frac{\sqrt[3]{3} \sqrt[3]{\pi^2}}{\pi}, \quad (\text{C.233})$$

$$\kappa = 0.004235, \quad (\text{C.234})$$

$$Z = -0.001667, \quad (\text{C.235})$$

$$\phi(r) = \theta(r) - Z, \quad (\text{C.236})$$

$$\theta(r) = \frac{1}{1000} \frac{2.568 + \Xi r + \Phi r^2}{1 + \Lambda r + \Upsilon r^2 + 10 \Phi r^3}, \quad (\text{C.237})$$

$$\Xi = 23.266, \quad (\text{C.238})$$

$$\Phi = 0.007389, \quad (\text{C.239})$$

$$\Lambda = 8.723, \quad (\text{C.240})$$

$$\Upsilon = 0.472, \quad (\text{C.241})$$

$$\begin{aligned}
\epsilon(\alpha, \beta) &= e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \\
&- \frac{e(r(\alpha, \beta), T_3, U_3, V_3, W_3, X_3, Y_3, P_3) \omega(\zeta(\alpha, \beta)) (1 - (\zeta(\alpha, \beta))^4)}{c} \\
&+ (e(r(\alpha, \beta), T_2, U_2, V_2, W_2, X_2, Y_2, P_2) - e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1)) \\
&\omega(\zeta(\alpha, \beta)) (\zeta(\alpha, \beta))^4,
\end{aligned} \tag{C.242}$$

$$r(\alpha, \beta) = 1/4 \sqrt[3]{3} 4^{2/3} \sqrt[3]{\frac{1}{\pi(\alpha + \beta)}}, \tag{C.243}$$

$$\zeta(\alpha, \beta) = \frac{\alpha - \beta}{\alpha + \beta}, \tag{C.244}$$

$$\omega(z) = \frac{(1+z)^{4/3} + (1-z)^{4/3} - 2}{2\sqrt[3]{2} - 2}, \tag{C.245}$$

$$e(r, t, u, v, w, x, y, p) = -2t(1 + ur) \ln \left( 1 + 1/2 \frac{1}{t(v\sqrt{r} + wr + xr^{3/2} + yr^{p+1})} \right), \tag{C.246}$$

$$c = 1.709921, \tag{C.247}$$

$$C(d, \alpha, \beta) = K(Q, \alpha, \beta) + M(Q, \alpha, \beta), \tag{C.248}$$

$$M(d, \alpha, \beta) = 0.5 \nu (\phi(r(\alpha, \beta)) - \kappa - 3/7 Z) d^2 e^{-335.9789467 \frac{3^{2/3} d^2}{\sqrt[3]{\pi^5 \rho}}}, \tag{C.249}$$

$$K(d, \alpha, \beta) = 0.2500000000 \lambda^2$$

$$\ln \left( 1 + 2 \frac{\iota(d^2 + N(\alpha, \beta)d^4)}{\lambda(1 + N(\alpha, \beta)d^2 + (N(\alpha, \beta))^2 d^4)} \right) \iota^{-1}, \tag{C.250}$$

$$N(\alpha, \beta) = 2 \iota \lambda^{-1} \left( e^{-4 \frac{\iota \epsilon(\alpha, \beta)}{\lambda^2}} - 1 \right)^{-1}, \tag{C.251}$$

$$Q = 1/12 \frac{\sqrt{\sigma_{ss}} \sqrt[3]{2} 3^{5/6}}{\sqrt[6]{\pi^{-1}} \rho^{7/6}}, \tag{C.252}$$

$$T = [0.031091, 0.015545, 0.016887], \tag{C.253}$$

$$U = [0.21370, 0.20548, 0.11125], \tag{C.254}$$

$$V = [7.5957, 14.1189, 10.357], \tag{C.255}$$

$$W = [3.5876, 6.1977, 3.6231], \tag{C.256}$$

$$X = [1.6382, 3.3662, 0.88026], \tag{C.257}$$

$$Y = [0.49294, 0.62517, 0.49671], \tag{C.258}$$

$$P = [1, 1, 1]. \tag{C.259}$$

为了避免极限  $\rho_{\bar{s}} \rightarrow 0$  的奇异性,

$$G = \rho (\epsilon (\rho_s, 0) + C (Q, \rho_s, 0)) . \quad (\text{C.260})$$

### C.27 PBEX: PBE交换泛函

更多细节参见文献 [14]。

$$K = \sum_s 1/2 E (2 \rho_s) , \quad (\text{C.261})$$

其中,

$$E (n) = -3/4 \frac{\sqrt[3]{3} \sqrt[3]{\pi^2} n^{4/3} F (S)}{\pi} , \quad (\text{C.262})$$

$$S = 1/12 \frac{\chi_s 6^{2/3}}{\sqrt[3]{\pi^2}} , \quad (\text{C.263})$$

$$F (S) = 1 + R - R \left( 1 + \frac{\mu S^2}{R} \right)^{-1} , \quad (\text{C.264})$$

$$R = 0.804 , \quad (\text{C.265})$$

$$\mu = 1/3 \delta \pi^2 , \quad (\text{C.266})$$

$$\delta = 0.066725 . \quad (\text{C.267})$$

为了避免极限  $\rho_{\bar{s}} \rightarrow 0$  的奇异性,

$$G = 1/2 E (2 \rho_s) . \quad (\text{C.268})$$

### C.28 PBEXREV: 改进的PBE交换泛函

修改了原始PBEX泛函中常数R的值。更多细节参见文献 [27]。

$$K = \sum_s 1/2 E (2 \rho_s) , \quad (\text{C.269})$$

其中,



$$E(n) = -3/4 \frac{\sqrt[3]{3}\sqrt[3]{\pi^2}n^{4/3}F(S)}{\pi}, \quad (\text{C.270})$$

$$S = 1/12 \frac{\chi_s 6^{2/3}}{\sqrt[3]{\pi^2}}, \quad (\text{C.271})$$

$$F(S) = 1 + R - R \left(1 + \frac{\mu S^2}{R}\right)^{-1}, \quad (\text{C.272})$$

$$R = 1.245, \quad (\text{C.273})$$

$$\mu = 1/3 \delta \pi^2, \quad (\text{C.274})$$

$$\delta = 0.066725. \quad (\text{C.275})$$

为了避免极限  $\rho_{\bar{s}} \rightarrow 0$  的奇异性,

$$G = 1/2 E(2\rho_s). \quad (\text{C.276})$$

### C.29 PW86:

GGA交换泛函。更多细节参见文献 [16]。

$$K = \sum_s 1/2 E(2\rho_s), \quad (\text{C.277})$$

其中,

$$E(n) = -3/4 \sqrt[3]{3}\sqrt[3]{\pi^{-1}}n^{4/3}F(S), \quad (\text{C.278})$$

$$F(S) = (1 + 1.296 S^2 + 14 S^4 + 0.2 S^6)^{1/15}, \quad (\text{C.279})$$

$$S = 1/12 \frac{\chi_s 6^{2/3}}{\sqrt[3]{\pi^2}}. \quad (\text{C.280})$$

为了避免极限  $\rho_{\bar{s}} \rightarrow 0$  的奇异性,

$$G = 1/2 E(2\rho_s). \quad (\text{C.281})$$

### C.30 PW91C: Perdew-Wang 1991 GGA关联泛函

更多细节参见文献 [15]。

$$K = \rho (\epsilon (\rho_\alpha, \rho_\beta) + H (d, \rho_\alpha, \rho_\beta)), \quad (\text{C.282})$$

其中,

$$d = 1/12 \frac{\sqrt{\sigma} 3^{5/6}}{u (\rho_\alpha, \rho_\beta) \sqrt[6]{\pi^{-1}} \rho^{7/6}}, \quad (\text{C.283})$$

$$u (\alpha, \beta) = 1/2 (1 + \zeta (\alpha, \beta))^{2/3} + 1/2 (1 - \zeta (\alpha, \beta))^{2/3}, \quad (\text{C.284})$$

$$H (d, \alpha, \beta) = L (d, \alpha, \beta) + J (d, \alpha, \beta), \quad (\text{C.285})$$

$$L (d, \alpha, \beta) = 1/2 (u (\rho_\alpha, \rho_\beta))^3 \lambda^2$$

$$\ln \left( 1 + 2 \frac{\iota (d^2 + A (\alpha, \beta) d^4)}{\lambda (1 + A (\alpha, \beta) d^2 + (A (\alpha, \beta))^2 d^4)} \right) \iota^{-1}, \quad (\text{C.286})$$

$$J (d, \alpha, \beta) = \nu (\phi (r (\alpha, \beta)) - \kappa - 3/7 Z) (u (\rho_\alpha, \rho_\beta))^3$$

$$d^2 e^{-\frac{400}{3} \frac{(u (\rho_\alpha, \rho_\beta))^4 3^{2/3} d^2}{\sqrt[3]{\pi^5 \rho}}}, \quad (\text{C.287})$$

$$A (\alpha, \beta) = 2 \iota \lambda^{-1} \left( e^{-2 \frac{\iota \epsilon (\alpha, \beta)}{(u (\rho_\alpha, \rho_\beta))^3 \lambda^2}} - 1 \right)^{-1}, \quad (\text{C.288})$$

$$\iota = 0.09, \quad (\text{C.289})$$

$$\lambda = \nu \kappa, \quad (\text{C.290})$$

$$\nu = 16 \frac{\sqrt[3]{3} \sqrt[3]{\pi^2}}{\pi}, \quad (\text{C.291})$$

$$\kappa = 0.004235, \quad (\text{C.292})$$

$$Z = -0.001667, \quad (\text{C.293})$$

$$\phi (r) = \theta (r) - Z, \quad (\text{C.294})$$

$$\theta (r) = \frac{1}{1000} \frac{2.568 + \Xi r + \Phi r^2}{1 + \Lambda r + \Upsilon r^2 + 10 \Phi r^3}, \quad (\text{C.295})$$

$$\Xi = 23.266, \quad (\text{C.296})$$

$$\Phi = 0.007389, \quad (\text{C.297})$$

$$\Lambda = 8.723, \quad (\text{C.298})$$

$$\Upsilon = 0.472, \quad (\text{C.299})$$

$$\begin{aligned} \epsilon(\alpha, \beta) = & e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \\ & - \frac{e(r(\alpha, \beta), T_3, U_3, V_3, W_3, X_3, Y_3, P_3) \omega(\zeta(\alpha, \beta)) (1 - (\zeta(\alpha, \beta))^4)}{c} \\ & + \left( e(r(\alpha, \beta), T_2, U_2, V_2, W_2, X_2, Y_2, P_2) - e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \right) \end{aligned} \quad (\text{C.300})$$

$$\omega(\zeta(\alpha, \beta)) (\zeta(\alpha, \beta))^4, \quad r(\alpha, \beta) = 1/4 \sqrt[3]{34^{2/3}} \sqrt[3]{\frac{1}{\pi(\alpha + \beta)}}, \quad (\text{C.301})$$

$$\zeta(\alpha, \beta) = \frac{\alpha - \beta}{\alpha + \beta}, \quad (\text{C.302})$$

$$\omega(z) = \frac{(1+z)^{4/3} + (1-z)^{4/3} - 2}{2\sqrt[3]{2} - 2}, \quad (\text{C.303})$$

$$\begin{aligned} e(r, t, u, v, w, x, y, p) = & -2t(1+ur) \\ \ln \left( 1 + 1/2 \frac{1}{t(v\sqrt{r} + wr + xr^{3/2} + yr^{p+1})} \right), & \quad (\text{C.304}) \\ c = & 1.709921, \quad (\text{C.305}) \end{aligned}$$

$$C(d, \alpha, \beta) = K(Q, \alpha, \beta) + M(Q, \alpha, \beta), \quad (\text{C.306})$$

$$M(d, \alpha, \beta) = 0.5\nu(\phi(r(\alpha, \beta)) - \kappa - 3/7Z)d^2 e^{-335.9789467 \frac{3^{2/3}d^2}{\sqrt[3]{\pi^5\rho}}}, \quad (\text{C.307})$$

$$K(d, \alpha, \beta) = 0.2500000000 \lambda^2 \ln \left( 1 + 2 \frac{\iota(d^2 + N(\alpha, \beta)d^4)}{\lambda(1 + N(\alpha, \beta)d^2 + \iota(N(\alpha, \beta))^2 d^4)} \right) \iota^{-1}, \quad (\text{C.308})$$

$$N(\alpha, \beta) = 2\iota\lambda^{-1} \left( e^{-4\frac{\iota(N(\alpha, \beta))}{\lambda^2}} - 1 \right), \quad (\text{C.309})$$

$$Q = 1/12 \frac{\sqrt{\sigma_{ss}} \sqrt[3]{23^{5/6}}}{\sqrt[6]{\pi^{-1}} \rho^{7/6}}, \quad (\text{C.310})$$

$$T = [0.031091, 0.015545, 0.016887], \quad (\text{C.311})$$

$$U = [0.21370, 0.20548, 0.11125], \quad (\text{C.312})$$

$$V = [7.5957, 14.1189, 10.357], \quad (\text{C.313})$$

$$W = [3.5876, 6.1977, 3.6231], \quad (\text{C.314})$$

$$X = [1.6382, 3.3662, 0.88026], \quad (\text{C.315})$$

$$Y = [0.49294, 0.62517, 0.49671], \quad (\text{C.316})$$

$$P = [1, 1, 1]. \quad (\text{C.317})$$

为了避免极限  $\rho_s \rightarrow 0$  的奇异性,

$$G = \rho(\epsilon(\rho_s, 0) + C(Q, \rho_s, 0)). \quad (\text{C.318})$$

### C.31 PW91X: Perdew-Wang 1991 GGA交换泛函

更多细节参见文献 [15]。

$$K = \sum_s 1/2 E(2\rho_s), \quad (\text{C.319})$$

其中,

$$E(n) = -3/4 \frac{\sqrt[3]{3}\sqrt[3]{\pi^2}n^{4/3}F(S)}{\pi}, \quad (\text{C.320})$$

$$S = 1/12 \frac{\chi_s 6^{2/3}}{\sqrt[3]{\pi^2}}, \quad (\text{C.321})$$

$$F(S) = \frac{1 + 0.19645 S \operatorname{arcsinh}(7.7956 S) + \left(0.2743 - 0.1508 e^{-100 S^2}\right) S^2}{1 + 0.19645 S \operatorname{arcsinh}(7.7956 S) + 0.004 S^4}. \quad (\text{C.322})$$

为了避免极限  $\rho_s \rightarrow 0$  的奇异性,

$$G = 1/2 E(2\rho_s). \quad (\text{C.323})$$

### C.32 PW92C: Perdew-Wang 1992 GGA关联泛函

电子气关联能。更多细节参见文献 [17]。

$$K = \rho \epsilon(\rho_\alpha, \rho_\beta), \quad (\text{C.324})$$

其中,

$$\begin{aligned} \epsilon(\alpha, \beta) = & e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \\ & - \frac{e(r(\alpha, \beta), T_3, U_3, V_3, W_3, X_3, Y_3, P_3) \omega(\zeta(\alpha, \beta)) (1 - (\zeta(\alpha, \beta))^4)}{c} \\ & + \left( e(r(\alpha, \beta), T_2, U_2, V_2, W_2, X_2, Y_2, P_2) - e(r(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \right) \omega(\zeta(\alpha, \beta)) (\zeta(\alpha, \beta))^4, \end{aligned} \quad (\text{C.325})$$

$$r(\alpha, \beta) = 1/4 \sqrt[3]{34^{2/3}} \sqrt[3]{\frac{1}{\pi(\alpha + \beta)}}, \quad (\text{C.326})$$

$$\zeta(\alpha, \beta) = \frac{\alpha - \beta}{\alpha + \beta}, \quad (\text{C.327})$$

$$\omega(z) = \frac{(1+z)^{4/3} + (1-z)^{4/3} - 2}{2\sqrt[3]{2} - 2}, \quad (\text{C.328})$$

$$e(r, t, u, v, w, x, y, p) = -2t(1 + ur) \ln \left( 1 + 1/2 \frac{1}{t(v\sqrt{r} + wr + xr^{3/2} + yr^{p+1})} \right), \quad (\text{C.329})$$

$$c = 1.709921, \quad (\text{C.330})$$

$$T = [0.031091, 0.015545, 0.016887], \quad (\text{C.331})$$

$$U = [0.21370, 0.20548, 0.11125], \quad (\text{C.332})$$

$$V = [7.5957, 14.1189, 10.357], \quad (\text{C.333})$$

$$W = [3.5876, 6.1977, 3.6231], \quad (\text{C.334})$$

$$X = [1.6382, 3.3662, 0.88026], \quad (\text{C.335})$$

$$Y = [0.49294, 0.62517, 0.49671], \quad (\text{C.336})$$

$$P = [1, 1, 1]. \quad (\text{C.337})$$

### C.33 STEST: 电子数测试

$$K = \sum_s \rho_s. \quad (\text{C.338})$$

### C.34 TH1: Tozer和Handy 1998

依赖于密度和梯度的第一行元素的交换-关联泛函。更多细节参见文献 [22]。

$$K = \sum_{i=1}^n \omega_i R_i S_i X_i Y_i, \quad (\text{C.339})$$

其中,

$$n = 21, \quad (\text{C.340})$$

$$R_i = (\rho_\alpha)^{t_i} + (\rho_\beta)^{t_i}, \quad (\text{C.341})$$

$$S_i = \left( \frac{\rho_\alpha - \rho_\beta}{\rho} \right)^{2u_i}, \quad (\text{C.342})$$

$$X_i = 1/2 \frac{(\sqrt{\sigma_{\alpha\alpha}})^{v_i} + (\sqrt{\sigma_{\beta\beta}})^{v_i}}{\rho^{4/3 v_i}}, \quad (\text{C.343})$$

$$Y_i = \left( \frac{\sigma_{\alpha\alpha} + \sigma_{\beta\beta} - 2\sqrt{\sigma_{\alpha\alpha}\sigma_{\beta\beta}}}{\rho^{8/3}} \right)^{w_i}, \quad (\text{C.344})$$

$$t = [7/6, 4/3, 3/2, 5/3, 4/3, 3/2, 5/3, \frac{11}{6}, 3/2, 5/3, \frac{11}{6}, 2, 3/2, 5/3, \frac{11}{6}, 2, 7/6, 4/3, 3/2, 5/3, 1], \quad (\text{C.345})$$

$$u = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0], \quad (\text{C.346})$$

$$v = [0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0], \quad (\text{C.347})$$

$$w = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0], \quad (\text{C.348})$$

$$\begin{aligned} \omega = & [-0.728255, 0.331699, -1.02946, 0.235703, -0.0876221, 0.140854, \\ & 0.0336982, -0.0353615, 0.00497930, -0.0645900, 0.0461795, \\ & -0.00757191, -0.00242717, 0.0428140, -0.0744891, 0.0386577, -0.352519, \\ & 2.19805, -3.72927, 1.94441, 0.128877]. \end{aligned} \quad (\text{C.349})$$

为了避免极限  $\rho_s \rightarrow 0$  的奇异性,

$$G = \sum_{i=1}^n 1/2 \omega_i (\rho_s)^{t_i} (\sqrt{\sigma_{ss}})^{v_i} \left( \frac{\sigma_{ss}}{(\rho_s)^{8/3}} \right)^{w_i} \left( (\rho_s)^{4/3 v_i} \right)^{-1}. \quad (\text{C.350})$$

### C.35 TH2:

依赖于密度和梯度的第一行元素的交换-关联泛函。更多细节参见文献 [21]。

$$K = \sum_{i=1}^n \omega_i R_i S_i X_i Y_i, \quad (\text{C.351})$$

其中,

$$n = 19, \quad (\text{C.352})$$

$$R_i = (\rho_\alpha)^{t_i} + (\rho_\beta)^{t_i}, \quad (\text{C.353})$$

$$S_i = \left( \frac{\rho_\alpha - \rho_\beta}{\rho} \right)^{2u_i}, \quad (\text{C.354})$$

$$X_i = 1/2 \frac{(\sqrt{\sigma_{\alpha\alpha}})^{v_i} + (\sqrt{\sigma_{\beta\beta}})^{v_i}}{\rho^{4/3 v_i}}, \quad (\text{C.355})$$

$$Y_i = \left( \frac{\sigma_{\alpha\alpha} + \sigma_{\beta\beta} - 2\sqrt{\sigma_{\alpha\alpha}\sigma_{\beta\beta}}}{\rho^{8/3}} \right)^{w_i}, \quad (\text{C.356})$$

$$t = [\frac{13}{12}, 7/6, 4/3, 3/2, 5/3, \frac{17}{12}, 3/2, 5/3, \frac{11}{6}, 5/3, \frac{11}{6}, 2, 5/3, \frac{11}{6}, 2, 7/6, 4/3, 3/2, 5/3], \quad (\text{C.357})$$

$$u = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1], \quad (\text{C.358})$$

$$v = [0, 0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 0, 0, 0, 0, 0, 0, 0], \quad (\text{C.359})$$

$$w = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0], \quad (\text{C.360})$$

$$\begin{aligned} \omega = & [0.678831, -1.75821, 1.27676, -1.60789, 0.365610, -0.181327, \\ & 0.146973, 0.147141, -0.0716917, -0.0407167, 0.0214625, -0.000768156, \\ & 0.0310377, -0.0720326, 0.0446562, -0.266802, 1.50822, -1.94515, \\ & 0.679078]. \end{aligned} \quad (\text{C.361})$$

为了避免极限  $\rho_s \rightarrow 0$  的奇异性,

$$G = \sum_{i=1}^n 1/2 \omega_i (\rho_s)^{t_i} (\sqrt{\sigma_{ss}})^{v_i} \left( \frac{\sigma_{ss}}{(\rho_s)^{8/3}} \right)^{w_i} \left( (\rho_s)^{4/3 v_i} \right)^{-1}. \quad (\text{C.362})$$

### C.36 TH3:

依赖于密度和梯度的第一、二行元素的交换-关联泛函。更多细节参见文献 [23]。

$$K = \sum_{i=1}^n \omega_i R_i S_i X_i Y_i, \quad (\text{C.363})$$

其中,

$$n = 19, \quad (\text{C.364})$$

$$R_i = (\rho_\alpha)^{t_i} + (\rho_\beta)^{t_i}, \quad (\text{C.365})$$

$$S_i = \left( \frac{\rho_\alpha - \rho_\beta}{\rho} \right)^{2u_i}, \quad (\text{C.366})$$

$$X_i = 1/2 \frac{(\sqrt{\sigma_{\alpha\alpha}})^{v_i} + (\sqrt{\sigma_{\beta\beta}})^{v_i}}{\rho^{4/3 v_i}}, \quad (\text{C.367})$$

$$Y_i = \left( \frac{\sigma_{\alpha\alpha} + \sigma_{\beta\beta} - 2\sqrt{\sigma_{\alpha\alpha}\sigma_{\beta\beta}}}{\rho^{8/3}} \right)^{w_i}, \quad (\text{C.368})$$

$$t = [7/6, 4/3, 3/2, 5/3, \frac{17}{12}, 3/2, 5/3, \frac{11}{6}, 5/3, \frac{11}{6}, 2, 5/3, \frac{11}{6}, 2, 7/6, 4/3, 3/2, 5/3, \frac{13}{12}], \quad (\text{C.369})$$

$$u = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0], \quad (\text{C.370})$$

$$v = [0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0], \quad (\text{C.371})$$

$$w = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0], \quad (\text{C.372})$$

$$\begin{aligned} \omega = & [-0.142542, -0.783603, -0.188875, 0.0426830, -0.304953, 0.430407, \\ & -0.0997699, 0.00355789, -0.0344374, 0.0192108, -0.00230906, 0.0235189, \\ & -0.0331157, 0.0121316, 0.441190, -2.27167, 4.03051, -2.28074, \\ & 0.0360204]. \end{aligned} \quad (\text{C.373})$$

为了避免极限  $\rho_s \rightarrow 0$  的奇异性,

$$G = \sum_{i=1}^n 1/2 \omega_i (\rho_s)^{t_i} (\sqrt{\sigma_{ss}})^{v_i} \left( \frac{\sigma_{ss}}{(\rho_s)^{8/3}} \right)^{w_i} \left( (\rho_s)^{4/3 v_i} \right)^{-1}. \quad (\text{C.374})$$

### C.37 TH4:

依赖于密度和梯度的第一、二行元素的交换-关联泛函。更多细节参见文献 [23]。

$$K = \sum_{i=1}^n \omega_i R_i S_i X_i Y_i, \quad (\text{C.375})$$

其中,

$$n = 19, \quad (\text{C.376})$$



$$R_i = (\rho_\alpha)^{t_i} + (\rho_\beta)^{t_i}, \quad (\text{C.377})$$

$$S_i = \left( \frac{\rho_\alpha - \rho_\beta}{\rho} \right)^{2u_i}, \quad (\text{C.378})$$

$$X_i = 1/2 \frac{(\sqrt{\sigma_{\alpha\alpha}})^{v_i} + (\sqrt{\sigma_{\beta\beta}})^{v_i}}{\rho^{4/3 v_i}}, \quad (\text{C.379})$$

$$Y_i = \left( \frac{\sigma_{\alpha\alpha} + \sigma_{\beta\beta} - 2\sqrt{\sigma_{\alpha\alpha}}\sqrt{\sigma_{\beta\beta}}}{\rho^{8/3}} \right)^{w_i}, \quad (\text{C.380})$$

$$t = [7/6, 4/3, 3/2, 5/3, \frac{17}{12}, 3/2, 5/3, \frac{11}{6}, 5/3, \frac{11}{6}, 2, 5/3, \frac{11}{6}, 2, 7/6, 4/3, 3/2, 5/3, \frac{13}{12}], \quad (\text{C.381})$$

$$u = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0], \quad (\text{C.382})$$

$$v = [0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0], \quad (\text{C.383})$$

$$w = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0], \quad (\text{C.384})$$

$$\begin{aligned} \omega = [0.0677353, -1.06763, -0.0419018, 0.0226313, -0.222478, 0.283432, \\ -0.0165089, -0.0167204, -0.0332362, 0.0162254, -0.000984119, 0.0376713, \\ -0.0653419, 0.0222835, 0.375782, -1.90675, 3.22494, -1.68698, \\ -0.0235810]. \end{aligned} \quad (\text{C.385})$$

为了避免极限  $\rho_s \rightarrow 0$  的奇异性,

$$G = \sum_{i=1}^n 1/2 \omega_i (\rho_s)^{t_i} (\sqrt{\sigma_{ss}})^{v_i} \left( \frac{\sigma_{ss}}{(\rho_s)^{8/3}} \right)^{w_i} \left( (\rho_s)^{4/3 v_i} \right)^{-1}. \quad (\text{C.386})$$

### C.38 THGFC:

依赖于密度和梯度的第一行元素的交换-关联泛函, 用于闭壳层体系。通过加入  $DN$  改善总能量, 其中,  $N$  是电子数,  $D = 0.1863$ 。更多细节参见文献 [24]。

$$K = \sum_{i=1}^n \omega_i R_i X_i, \quad (\text{C.387})$$

其中,

$$n = 12, \quad (\text{C.388})$$

$$R_i = (\rho_\alpha)^{t_i} + (\rho_\beta)^{t_i}, \quad (\text{C.389})$$

$$X_i = 1/2 \frac{(\sqrt{\sigma_{\alpha\alpha}})^{v_i} + (\sqrt{\sigma_{\beta\beta}})^{v_i}}{\rho^{4/3 v_i}}, \quad (\text{C.390})$$

$$t = [7/6, 4/3, 3/2, 5/3, 4/3, 3/2, 5/3, \frac{11}{6}, 3/2, 5/3, \frac{11}{6}, 2], \quad (\text{C.391})$$

$$v = [0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 2], \quad (\text{C.392})$$

$$\omega = [-0.864448, 0.565130, -1.27306, 0.309681, -0.287658, 0.588767, \\ -0.252700, 0.0223563, 0.0140131, -0.0826608, 0.0556080, -0.00936227]. \quad (\text{C.393})$$

为了避免极限  $\rho_{\bar{s}} \rightarrow 0$  的奇异性,

$$G = \sum_{i=1}^n 1/2 \frac{\omega_i (\rho_s)^{t_i} (\sqrt{\sigma_{ss}})^{v_i}}{\rho^{4/3 v_i}}. \quad (\text{C.394})$$

### C.39 THGFCFO:

依赖于密度和梯度的第一行元素的交换-关联泛函。FCFO = FC + 开壳层拟合。更多细节参见文献 [24]。

$$K = \sum_{i=1}^n \omega_i R_i S_i X_i Y_i, \quad (\text{C.395})$$

其中,

$$n = 20, \quad (\text{C.396})$$

$$R_i = (\rho_\alpha)^{t_i} + (\rho_\beta)^{t_i}, \quad (\text{C.397})$$

$$S_i = \left( \frac{\rho_\alpha - \rho_\beta}{\rho} \right)^{2 u_i}, \quad (\text{C.398})$$

$$X_i = 1/2 \frac{(\sqrt{\sigma_{\alpha\alpha}})^{v_i} + (\sqrt{\sigma_{\beta\beta}})^{v_i}}{\rho^{4/3 v_i}}, \quad (\text{C.399})$$

$$Y_i = \left( \frac{\sigma_{\alpha\alpha} + \sigma_{\beta\beta} - 2 \sqrt{\sigma_{\alpha\alpha}} \sqrt{\sigma_{\beta\beta}}}{\rho^{8/3}} \right)^{w_i}, \quad (\text{C.400})$$

$$t = [7/6, 4/3, 3/2, 5/3, 4/3, 3/2, 5/3, \frac{11}{6}, 3/2, 5/3, \frac{11}{6}, 2, 3/2, 5/3, \\ \frac{11}{6}, 2, 7/6, 4/3, 3/2, 5/3], \quad (\text{C.401})$$

$$u = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1], \quad (\text{C.402})$$

$$v = [0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0], \quad (\text{C.403})$$

$$w = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0], \quad (\text{C.404})$$

$$\begin{aligned} \omega = & [-0.864448, 0.565130, -1.27306, 0.309681, -0.287658, 0.588767, \\ & -0.252700, 0.0223563, 0.0140131, -0.0826608, 0.0556080, -0.00936227, \\ & -0.00677146, 0.0515199, -0.0874213, 0.0423827, 0.431940, -0.691153, \\ & -0.637866, 1.07565]. \end{aligned} \quad (\text{C.405})$$

为了避免极限  $\rho_{\bar{s}} \rightarrow 0$  的奇异性,

$$G = \sum_{i=1}^n 1/2 \omega_i (\rho_s)^{t_i} (\sqrt{\sigma_{ss}})^{v_i} \left( \frac{\sigma_{ss}}{(\rho_s)^{8/3}} \right)^{w_i} \left( (\rho_s)^{4/3 v_i} \right)^{-1}. \quad (\text{C.406})$$

#### C.40 THGFC0:

依赖于密度和梯度的第一行元素的交换-关联泛函。更多细节参见文献 [24]。

$$K = \sum_{i=1}^n \omega_i R_i S_i X_i Y_i, \quad (\text{C.407})$$

其中,

$$n = 20, \quad (\text{C.408})$$

$$R_i = (\rho_\alpha)^{t_i} + (\rho_\beta)^{t_i}, \quad (\text{C.409})$$

$$S_i = \left( \frac{\rho_\alpha - \rho_\beta}{\rho} \right)^{2u_i}, \quad (\text{C.410})$$

$$X_i = 1/2 \frac{(\sqrt{\sigma_{\alpha\alpha}})^{v_i} + (\sqrt{\sigma_{\beta\beta}})^{v_i}}{\rho^{4/3 v_i}}, \quad (\text{C.411})$$

$$Y_i = \left( \frac{\sigma_{\alpha\alpha} + \sigma_{\beta\beta} - 2\sqrt{\sigma_{\alpha\alpha}\sigma_{\beta\beta}}}{\rho^{8/3}} \right)^{w_i}, \quad (\text{C.412})$$

$$\begin{aligned} t = & [7/6, 4/3, 3/2, 5/3, 4/3, 3/2, 5/3, \frac{11}{6}, 3/2, 5/3, \frac{11}{6}, 2, 3/2, 5/3, \\ & \frac{11}{6}, 2, 7/6, 4/3, 3/2, 5/3], \end{aligned} \quad (\text{C.413})$$

$$u = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1], \quad (\text{C.414})$$

$$v = [0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0], \quad (\text{C.415})$$

$$w = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0], \quad (\text{C.416})$$

$$\begin{aligned} \omega = & [-0.962998, 0.860233, -1.54092, 0.381602, -0.210208, 0.391496, \\ & -0.107660, -0.0105324, 0.00837384, -0.0617859, 0.0383072, -0.00526905, \\ & -0.00381514, 0.0321541, -0.0568280, 0.0288585, 0.368326, -0.328799, \\ & -1.22595, 1.36412]. \end{aligned} \quad (\text{C.417})$$

为了避免极限  $\rho_{\bar{s}} \rightarrow 0$  的奇异性,

$$G = \sum_{i=1}^n 1/2 \omega_i (\rho_s)^{t_i} (\sqrt{\sigma_{ss}})^{v_i} \left( \frac{\sigma_{ss}}{(\rho_s)^{8/3}} \right)^{w_i} \left( (\rho_s)^{4/3 v_i} \right)^{-1}. \quad (\text{C.418})$$

#### C.41 THGFL:

依赖于密度和梯度的第一行元素的交换-关联泛函, 用于闭壳层体系。更多细节参见文献 [24]。

$$K = \sum_{i=1}^n \omega_i R_i, \quad (\text{C.419})$$

其中,

$$n = 4, \quad (\text{C.420})$$

$$R_i = (\rho_{\alpha})^{t_i} + (\rho_{\beta})^{t_i}, \quad (\text{C.421})$$

$$t = [7/6, 4/3, 3/2, 5/3], \quad (\text{C.422})$$

$$\omega = [-1.06141, 0.898203, -1.34439, 0.302369]. \quad (\text{C.423})$$

#### C.42 VSXC:

更多细节参见文献 [25]。

$$\begin{aligned}
K = & F(x, z, p_3, q_3, r_3, t_3, u_3, v_3, \alpha_3) (\epsilon(\rho_\alpha, \rho_\beta) - \epsilon(\rho_\alpha, 0) - \epsilon(\rho_\beta, 0)) \\
& + \sum_s (\rho_s)^{4/3} F(\chi_s, zs, p_1, q_1, r_1, t_1, u_1, v_1, \alpha_1) \\
& + ds \epsilon(\rho_s, 0) F(\chi_s, zs, p_2, q_2, r_2, t_2, u_2, v_2, \alpha_2),
\end{aligned} \tag{C.424}$$

其中,

$$x = (\chi_\alpha)^2 + (\chi_\beta)^2, \tag{C.425}$$

$$zs = \frac{\tau_s}{(\rho_s)^{5/3}} - cf, \tag{C.426}$$

$$z = \frac{\tau_\alpha}{(\rho_\alpha)^{5/3}} + \frac{\tau_\beta}{(\rho_\beta)^{5/3}} - 2cf, \tag{C.427}$$

$$ds = 1 - \frac{(\chi_s)^2}{4zs + 4cf}, \tag{C.428}$$

$$\begin{aligned}
F(x, z, p, q, c, d, e, f, \alpha) = & \frac{p}{\lambda(x, z, \alpha)} + \frac{qx^2 + cz}{(\lambda(x, z, \alpha))^2} \\
& + \frac{dx^4 + ex^2z + fz^2}{(\lambda(x, z, \alpha))^3},
\end{aligned} \tag{C.429}$$

$$\lambda(x, z, \alpha) = 1 + \alpha(x^2 + z), \tag{C.430}$$

$$cf = 3/5 \, 3^{2/3} (\pi^2)^{2/3}, \tag{C.431}$$

$$\begin{aligned}
\epsilon(\alpha, \beta) = & (\alpha + \beta) \left( e(l(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \right. \\
& - \frac{e(l(\alpha, \beta), T_3, U_3, V_3, W_3, X_3, Y_3, P_3) \omega(\zeta(\alpha, \beta)) (1 - (\zeta(\alpha, \beta))^4)}{c} \\
& \left. + \left( e(l(\alpha, \beta), T_2, U_2, V_2, W_2, X_2, Y_2, P_2) - e(l(\alpha, \beta), T_1, U_1, V_1, W_1, X_1, Y_1, P_1) \right) \right) \\
& \omega(\zeta(\alpha, \beta)) (\zeta(\alpha, \beta))^4,
\end{aligned} \tag{C.432}$$

$$l(\alpha, \beta) = 1/4 \sqrt[3]{34^{2/3}} \sqrt[3]{\frac{1}{\pi(\alpha + \beta)}}, \tag{C.433}$$

$$\zeta(\alpha, \beta) = \frac{\alpha - \beta}{\alpha + \beta}, \tag{C.434}$$

$$\omega(z) = \frac{(1+z)^{4/3} + (1-z)^{4/3} - 2}{2\sqrt[3]{2} - 2}, \tag{C.435}$$

$$e(r, t, u, v, w, x, y, p) = -2t(1 + ur) \ln \left( 1 + 1/2 \frac{1}{t(v\sqrt{r} + wr + xr^{3/2} + yr^{p+1})} \right), \quad (\text{C.436})$$

$$c = 1.709921, \quad (\text{C.437})$$

$$p = [-0.98, 0.3271, 0.7035], \quad (\text{C.438})$$

$$q = [-0.003557, -0.03229, 0.007695], \quad (\text{C.439})$$

$$r = [0.00625, -0.02942, 0.05153], \quad (\text{C.440})$$

$$t = [-0.00002354, 0.002134, 0.00003394], \quad (\text{C.441})$$

$$u = [-0.0001283, -0.005452, -0.001269], \quad (\text{C.442})$$

$$v = [0.0003575, 0.01578, 0.001296], \quad (\text{C.443})$$

$$\alpha = [0.001867, 0.005151, 0.00305], \quad (\text{C.444})$$

$$T = [0.031091, 0.015545, 0.016887], \quad (\text{C.445})$$

$$U = [0.21370, 0.20548, 0.11125], \quad (\text{C.446})$$

$$V = [7.5957, 14.1189, 10.357], \quad (\text{C.447})$$

$$W = [3.5876, 6.1977, 3.6231], \quad (\text{C.448})$$

$$X = [1.6382, 3.3662, 0.88026], \quad (\text{C.449})$$

$$Y = [0.49294, 0.62517, 0.49671], \quad (\text{C.450})$$

$$P = [1, 1, 1]. \quad (\text{C.451})$$

为了避免极限  $\rho_{\bar{s}} \rightarrow 0$  的奇异性,

$$G = (\rho_s)^{4/3} F(\chi_s, zs, p_1, q_1, r_1, t_1, u_1, v_1, \alpha_1) + ds \epsilon(\rho_s, 0) F(\chi_s, zs, p_2, q_2, r_2, t_2, u_2, v_2, \alpha_2). \quad (\text{C.452})$$

### C.43 VWN3: Vosko-Wilk-Nusair (1980) III局域关联能

VWN 1980(III)泛函。在文献中,  $\Delta\epsilon_c(r_s, \zeta)_{III}$  的拟合参数位于方程4.4稍后的文字内。更多细节参见文献 [26]。

$$K = \rho e, \quad (\text{C.453})$$

其中,

$$x = 1/4 \sqrt[6]{34}^{5/6} \sqrt[6]{\frac{1}{\pi \rho}}, \quad (\text{C.454})$$

$$\zeta = \frac{\rho_\alpha - \rho_\beta}{\rho}, \quad (\text{C.455})$$

$$e = \Lambda + z(\lambda - \Lambda), \quad (\text{C.456})$$

$$y = \frac{9}{8} (1 + \zeta)^{4/3} + \frac{9}{8} (1 - \zeta)^{4/3} - 9/4, \quad (\text{C.457})$$

$$\Lambda = q(k_1, l_1, m_1, n_1), \quad (\text{C.458})$$

$$\lambda = q(k_2, l_2, m_2, n_2), \quad (\text{C.459})$$

$$(\text{C.460})$$

$$q(A, p, c, d) = A \left( \ln \left( \frac{x^2}{X(x, c, d)} \right) + 2c \arctan \left( \frac{Q(c, d)}{2x + c} \right) (Q(c, d))^{-1} \right. \\ \left. - cp \left( \ln \left( \frac{(x-p)^2}{X(x, c, d)} \right) + 2(c+2p) \arctan \left( \frac{Q(c, d)}{2x + c} \right) (Q(c, d))^{-1} \right) (X(p, c, d))^{-1} \right), \quad (\text{C.461})$$

$$Q(c, d) = \sqrt{4d - c^2}, \quad (\text{C.462})$$

$$X(i, c, d) = i^2 + ci + d, \quad (\text{C.463})$$

$$k = [0.0310907, 0.01554535, -1/6 \pi^{-2}], \quad (\text{C.464})$$

$$l = [-0.409286, -0.743294, -0.0047584], \quad (\text{C.465})$$

$$m = [13.0720, 20.1231, 1.06835], \quad (\text{C.466})$$

$$n = [42.7198, 101.578, 11.4813], \quad (\text{C.467})$$

$$z = 4 \frac{y}{9\sqrt[3]{2} - 9}. \quad (\text{C.468})$$

#### C.44 VWN5: Vosko-Wilk-Nusair (1980) V局域关联能

VWN 1980(V)泛函。在文献中,  $\Delta \varepsilon_c(r_s, \zeta)_V$  的拟合参数位于表7的标题。更多细节参见文献 [26]。

$$K = \rho e, \quad (\text{C.469})$$

其中,

$$x = 1/4 \sqrt[6]{34}^{5/6} \sqrt[6]{\frac{1}{\pi \rho}}, \quad (\text{C.470})$$

$$\zeta = \frac{\rho_\alpha - \rho_\beta}{\rho}, \quad (\text{C.471})$$

$$e = \Lambda + \alpha y (1 + h\zeta^4), \quad (\text{C.472})$$

$$y = \frac{9}{8} (1 + \zeta)^{4/3} + \frac{9}{8} (1 - \zeta)^{4/3} - 9/4, \quad (\text{C.473})$$

$$h = 4/9 \frac{\lambda - \Lambda}{(\sqrt[3]{2} - 1) \alpha} - 1, \quad (\text{C.474})$$

$$\Lambda = q(k_1, l_1, m_1, n_1), \quad (\text{C.475})$$

$$\lambda = q(k_2, l_2, m_2, n_2), \quad (\text{C.476})$$

$$\alpha = q(k_3, l_3, m_3, n_3), \quad (\text{C.477})$$

$$\begin{aligned} q(A, p, c, d) = & A \left( \ln \left( \frac{x^2}{X(x, c, d)} \right) + 2c \arctan \left( \frac{Q(c, d)}{2x + c} \right) (Q(c, d))^{-1} \right. \\ & \left. - cp \left( \ln \left( \frac{(x-p)^2}{X(x, c, d)} \right) + 2(c+2p) \arctan \left( \frac{Q(c, d)}{2x + c} \right) (Q(c, d))^{-1} \right) (X(p, c, d))^{-1} \right), \end{aligned} \quad (\text{C.478})$$

$$Q(c, d) = \sqrt{4d - c^2}, \quad (\text{C.479})$$

$$X(i, c, d) = i^2 + ci + d, \quad (\text{C.480})$$

$$k = [0.0310907, 0.01554535, -1/6 \pi^{-2}], \quad (\text{C.481})$$

$$l = [-0.10498, -0.325, -0.0047584], \quad (\text{C.482})$$

$$m = [3.72744, 7.06042, 1.13107], \quad (\text{C.483})$$

$$n = [12.9352, 18.0578, 13.0045]. \quad (\text{C.484})$$



## 索引

\*\*\*, 45  
---, 45  
SURF, 425  
VCI, 439  
VSCF, 437  
  
ACCURACY, 136, 174  
ACPF, 187  
ACTIVE, 377  
ADD, 308, 362, 398, 450  
ALTERN, 343  
ANGULAR, 144  
AOINT, 109  
AQCC, 187  
  
BASIS, 95, 96  
BCCD, 222  
BMAT, 376  
BRUECKNER, 223  
  
CANONICAL, 168  
CANORB, 168  
CASPROJ, 342  
CASPT2, 199  
    CASPT2梯度, 206  
    CIPT2, 209  
    G1, 205  
    G2, 205  
    G3, 205  
    G4, 205  
    MS-CASPT2, 201  
    能级移动, 205  
    入侵态, 205

CASPT3, 199  
CASSCF, 159, 337  
    CASVB, 337  
CCSD, 222  
CCSD, 181, 222  
CCSD(T), 222  
CEPA, 187  
CHARGE, 24  
CHECK, 222, 227  
CI, 181  
CI, 181  
CI-PRO, 181  
CIS, 232  
CIS, 232  
CISD, 223  
CISD, 181  
CIWEIGHTS, 346  
CLEAR, 77  
CLEARALL, 77  
CLOSED, 25, 131, 161, 182  
COEFFS, 344  
COMPRESS, 109  
CON, 164, 185, 339  
CONFIG, 171  
CONICAL, 385  
COORD, 376  
COPT, 175  
CORE, 25, 154, 182, 288  
COSMO, 443  
Cowan-Griffin, 313  
CPF, 187

CPHF, 218  
CPMCSCF, 176  
CPP, 107  
CPROP, 233  
CRD, 87  
CRIT, 342  
CUBE, 314  
CUT, 383  
  
Darwin, 313  
DATA, 21, 60  
*DDR*, 325  
DELETE, 60, 308  
DELOCAL, 154  
DELSTRUC, 345  
DEMC, 364  
DENSITY, 25, 140, 157, 303, 307, 309  
DF-LMP2, 271  
DF-MP2, 218  
DFT, 139  
DFTBLOCK, 142  
DFTDUMP, 142  
DFTFACTOR, 141  
DFTTHRESH, 140  
DGRAD, 177  
DIIS, 176, 223  
DIP, 310  
DIP+, 310  
DIRECT, 110, 137  
DISPCORR, 150  
DM, 173, 190, 217, 225  
DMA, 307  
DO, 46  
DONT, 172  
  
DO循环, 46  
DUMMY, 90  
DUMP, 288  
  
ECP  
    基组库, 103  
ECP, 103  
ELSEIF, 47  
ENDDO, 46  
ENDIF, 47  
ENDZ, 84  
EOM, 227  
EOM-CCSD, 227  
EOMPAR, 228  
EOMPRINT, 229  
ERASE, 60  
EXCHANGE, 141  
EXPEC, 54, 135, 172, 190  
EXPEC2, 172  
EXTRA, 450  
  
FCI, 287  
FIELD, 311  
FIELD+, 311  
FILE, 59  
FIXORB, 344  
FIXSTRUC, 345  
FOCK, 156, 189  
FORCE, 361  
FREEZE, 161  
FREQUENCIES, 405  
FROZEN, 25, 161  
FULL, 345  
  
Gaussian, 87

- GDIRECT, 110  
GENERAL, 307  
GEOMETRY, 83  
    XYZ输入, 85  
    Z矩阵, 84  
    结构文件, 88  
    写CRD文件, 87  
    写Gaussian输入, 87  
    写MOLDEN输入, 87  
    写XMol文件, 87  
GEXPEC, 54  
GOPENMOL, 316  
GOTO, 48  
GPARAM, 61  
GPRINT, 53  
GRADTYP, 361  
GRID, 142  
GRIDPRINT, 145  
GRIDSAVE, 145  
GRIDSYM, 145  
GRIDTHRESH, 143  
GROUP, 155, 345  
GTHRESH, 52  
GUESS, 341  
  
HESS, 177  
HESSELEM, 380  
HESSIAN, 177  
HESSIAN, 378  
Hessian, 378, 380  
    Hessian元素, 380  
    模型Hessian, 377, 378  
    数值Hessian, 379  
HF, 127  
    HF-SCF, 127  
  
IF, 47  
IF块, 47  
INACTIVE, 377  
INCLUDE, 46  
INDIVIDUAL, 309  
INIT, 452  
INTOPT, 175  
IPOL, 137  
IPRINT, 173  
IRC, 375, 383  
IRREPS, 343  
ITERATIONS, 171  
  
KS, 139  
KS-SCF, 139  
  
LABEL, 48  
LATTICE, 89  
LIBMOL, 103  
libmol, 94  
library, 103  
LIMIT, 308  
LINEAR, 307  
LINESEARCH, 383  
LOCAL, 168  
LOCALI, 153  
LOCAO, 154  
LOCORB, 168  
LQUANT, 165  
  
MASS, 89  
MATROP, 457  
MAXDAV, 189

- MAXITER, 136, 174, 189, 342  
MCSCF, 159  
MCSCF, 160, 362  
MCSCF hessian, 177  
MEMORY, 46  
MERGE, 449  
METHOD, 375  
*MOLDEN*, 87  
*molpro*, 3  
MOLPRO2000, 490  
MOLPRO2002, 489  
MOLPRO2002.6, 488  
MOLPRO2006.1, 487  
MOLPRO2008.1, 486  
MOLPRO2009.1, 485  
MOLPRO98, 491  
MOLPRO 基组, 94  
MOVE, 449  
MP2, 217  
MP2-F12, 273  
MP2-R12, 273  
MP3, 217  
MP4, 217  
MPP, 6  
MPPX, 6  
Mulliken 分析, 309  
*MULTI*, 159  
MULTI, 160  
  
NACM, 177, 364  
NACME, 177, 325  
NATORB, 167, 191, 225  
NBO, 310  
NELEC, 24  
  
NEVPT2, 213  
NOCASPROJ, 342  
NOCHECK, 222, 227  
NOENEST, 136  
NOEXC, 187  
NOEXTRA, 172  
NOGPRINT, 53  
NOGRIDSAVE, 145  
NOGRIDSYM, 145  
NONLINEAR, 175  
NONUCLEAR, 308  
NOORDER, 155  
NOPAIR, 187  
NOSINGLE, 187  
NOSYMPROJ, 344  
NUMERICAL, 364, 381  
NUMHES, 379  
  
OCC, 25, 131, 154, 161, 182, 287  
OFFDIAG, 155  
OFFSET, 451  
OPEN, 131  
OPTG, 369  
OPTIM, 342  
OPTION, 191, 210, 383  
ORB, 341  
ORBIT, 287  
ORBITAL, 25, 132, 153, 156, 166, 183,  
303, 362, 449  
ORBPERM, 341  
ORBPRINT, 137, 173  
ORBREL, 344  
ORTH, 137, 345, 451  
ORTHCON, 345

- PAIR, 186  
PAIRS, 187, 345  
PARAM, 192  
POLARIZABILITY, 136  
POP, 309  
POTENTIAL, 141  
PRINT, 157, 173, 193, 288, 304, 347, 384, 452, 463  
PROC, 49  
PROJECT, 188, 451  
PROPERTY, 303  
PSPACE, 165, 189  
PUNCH, 61  
PUT, 87  
  
QCI, 222  
QCI, 181  
QM/MM, 447  
QUAD, 311  
QUAD+, 311  
  
RADIAL, 143  
RADIUS, 308  
RANGEHYBRID, 141  
READ, 341  
*READPUN*, 20  
READVAR, 77  
REF, 183  
REFSTATE, 186  
REL, 313  
RELAX, 228  
RESTART, 21, 45  
RESTRICT, 163, 185  
RHF, 127  
RHF-SCF, 127  
RI-MP2, 218  
RKS, 139  
RKS-SCF, 139  
ROOT, 382  
ROTATE, 135, 166, 451  
RS2, 199  
RS2, 200  
RS2C, 200  
RS3, 199  
RS3, 200  
  
SADDLE, 342  
SAMC, 363  
SAPT, 291  
SAVE, 154, 166, 189, 340, 452  
SCALE, 362  
SCF, 127  
SCHMIDT, 451  
SCORR, 345  
SCS-MP2, 219  
SELECT, 163, 184  
SERVICE, 347  
SET, 63  
SHIFT, 136, 188, 205  
SHOW, 76  
Slater基组, 245  
SPECIAL, 348  
SPIN, 24  
SPINBASIS, 339  
START, 132, 165, 190, 340, 347  
STATE, 162, 185  
STATUS, 51  
STEP, 175, 382

- STRONG, 345  
STRUC, 341  
SYM, 135  
SYMELM, 343  
SYMPROJ, 344  
  
TABLE, 79  
TEST, 175  
THERMO, 408  
THRESH, 157, 176, 193, 221  
TRAN, 172  
TRAN2, 172  
TRANH, 188  
TRANS, 190, 228, 344  
TRNINT, 176  
TRUST, 382  
  
UHF, 127  
UHF-SCF, 127  
UKS, 139  
UKS-SCF, 139  
UNCOMPRESS, 109  
UPDATE, 380  
  
VARIABLE, 384, 408  
VB, 337  
VB, 178  
VBDUMP, 176, 338  
VBWEIGHTS, 346  
VCI程序, 439  
VORONOI, 145  
VSCF程序, 437  
  
WEIGHT, 162  
WF, 24, 131, 162, 183, 288  
  
WRITE, 347  
  
XYZ, 85, 87  
  
ZMAT, 84  
Z矩阵, 84  
  
安装, 469  
帮助, 31  
  
变量, 63  
    设定, 63  
    索引, 68  
    特殊, 70  
    系统, 65  
    字符串, 64  
  
并行, 6  
布居分析, 309, 310  
  
程序结构, 19  
  
定义波函, 24  
对称性, 86  
    WF卡, 24  
    附加对称性  
        MCSCF, 172  
        SCF, 135  
    积分程序, 22  
多参考微扰理论, 199, 213  
  
反应路径, 375, 383  
非绝热耦合, 177, 325, 329  
分布多极分析, 307  
分类的积分, 109  
  
关键词, 27  
轨道, 25

- 闭壳层轨道, 25
  - CI, 182
  - MCSCF, 161
- 冻结轨道, 25
  - MCSCF, 161
- 冻芯轨道, 25
  - CI, 182
  - FCI, 288
  - MCSCF, 161
- 内部轨道, 25
  - CI, 182
- 占据轨道, 25
  - CI, 182
  - FCI, 287
  - MCSCF, 161
- 轨道操作, 449
- 轨道局域化, 153
- 轨道空间, 24
- 函数最小化, 413
- 绘图, 87
- 基组, 93
  - 笛卡尔基函数, 93
  - 平缓基函数, 100
  - 球谐基函数, 93
  - 收缩, 101
  - 原始集, 99
- 积分, 109
- 极化率, 217
- 几何优化, 369
  - DIIS方法, 369, 375
  - 鞍点, 375, 382
  - 二次最陡下降方法, 369, 375, 383
- 过渡态, 375, 382
- 能量变量, 384
- 平衡校正, 398
- 收敛判据, 371
- 有理函数方法, 369, 375
- 圆锥交点, 385
- 自动, 369
- 记录, 20
- 加索引的变量, 68
- 结构, 83
- 局域关联, 249
- 局域化轨道空间, 154
- 矩阵操作, 457
- 流程, 49
- 密度泛函
  - B86, 493
  - B86MGC, 493
  - B86R, 494
  - B88, 495
  - B88C, 495
  - B95, 496
  - B97DF, 498
  - B97RDF, 499
  - BR, 501
  - BRUEG, 501
  - BW, 501
  - CS1, 502
  - CS2, 502
  - DIRAC, 503
  - ECERF, 503
  - EXERF, 505

- G96, 505
- HCTH120, 506
- HCTH147, 507
- HCTH93, 509
- LTA, 510
- MK00, 512
- MK00B, 512
- P86, 512
- PBEC, 514
- PBEX, 516
- PBEXREV, 516
- PW86, 517
- PW91C, 518
- PW91X, 520
- PW92C, 520
- STEST, 521
- TH1, 521
- TH2, 522
- TH3, 523
- TH4, 524
- THGFC, 525
- THGFCF0, 526
- THGFC0, 527
- THGFL, 528
- VSXC, 528
- VWN3, 530
- VWN5, 531
- 密度矩阵, 25
- 密度拟合, 121
- 内存分配, 22
- 内反应坐标, 375, 383
- 偶极场, 310
- 频率, 405
  - 能量变量, 408
- 期待值, 54
- 入侵态, 205
- 色散校正, 150
- 矢量操作, 69
- 示范的例子, 33
- 势能曲面, 425
- 输入结构, 19
- 数值梯度, 364
- 双基组, 226
- 四极场, 311
- 特殊变量, 70
- 特性, 303
  - CI, 190
  - MCSCF, 172
- 梯度, 361
- 提示, 1
- 同位素质量, 89
- 透热轨道, 321
- 外推, 417
- 完全CI, 287
- 文件, 20
- 系统变量, 65
- 显关联方法, 273
- 相对论效应
  - 标量相对论, 313, 319
  - 自旋轨道耦合, 351
- 虚中心 (Q,X), 84



循环, 22

引用, xi

有效芯势, 103

原子质量, 89

运行MOLPRO, 3

振动频率, 405

直接积分, 110

制表, 79

质量-速率, 313

准透热化, 329

自然键轨道分析, 310

字符串变量, 64

字符串变量的宏定义, 66

坐标, 376

    B矩阵, 376

    Z矩阵, 376

    笛卡尔, 376

    自然内坐标, 376

赝势, 103