A Practical Guide for XMVB Calculations

Peifeng Su

Xiamen University

Outline

- General introduction
- How to write input files?
- How to get information from output files?

General introduction

An Ab Initio Non-orthogonal Valence Bond Program

Version 2.0

Lingchun Song, Zhenhua Chen, Fuming Ying, Jinshuai Song, Xun Chen, Peifeng Su, Yirong Mo, Qianer Zhang, Wei Wu*

State Key Laboratory of Physical Chemistry of Solid Surfaces,

Fujian Provincial Key Laboratory of Theoretical and Computational Chemistry,

and Department of Chemistry,

Xiamen University, Xiamen, Fujian 361005, CHINA

Distributions of XMVB 2.0

Module distribution → A module embedded in GAMESS-US

Stand-alone distribution → A stand-alone program

Current capabilities of XMVB 2.0

VBSCF

Post-VBSCF: VBCI, BOVB, VBPT2 etc

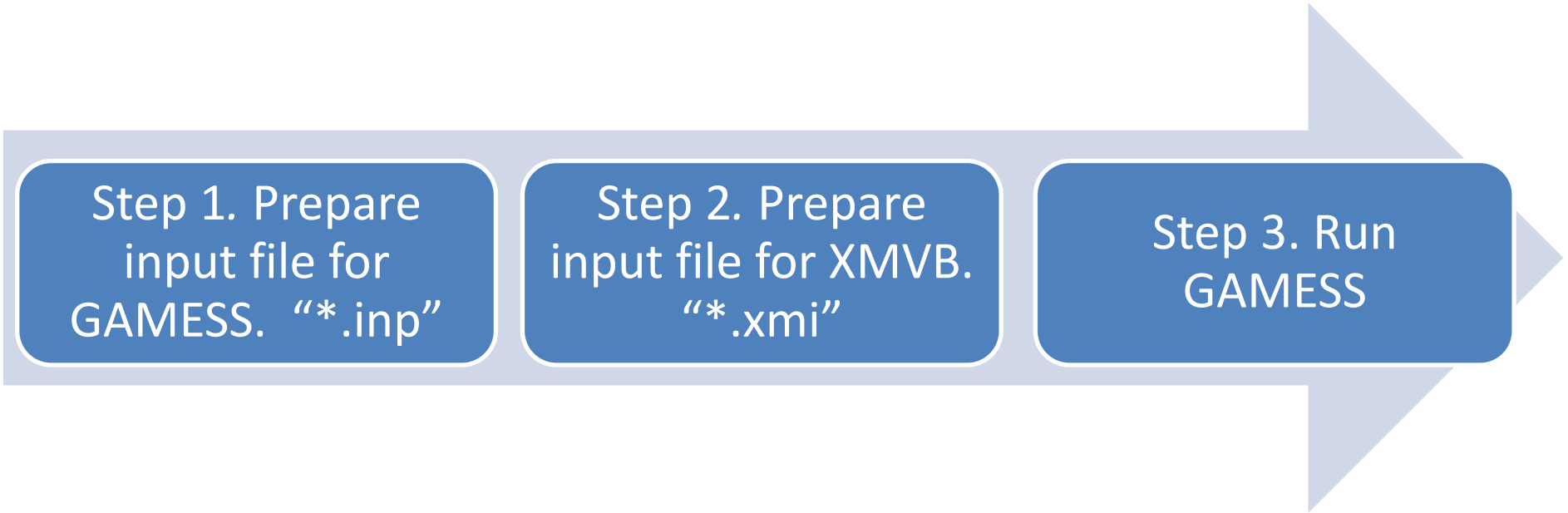
Solvation VB methods: VBPCM, VBEFP etc.

Energy (Total, individual), Weights of structure

Resonance energy

Properties: bond order, atomic charges, dipole etc

How to Run An XMVB Job?

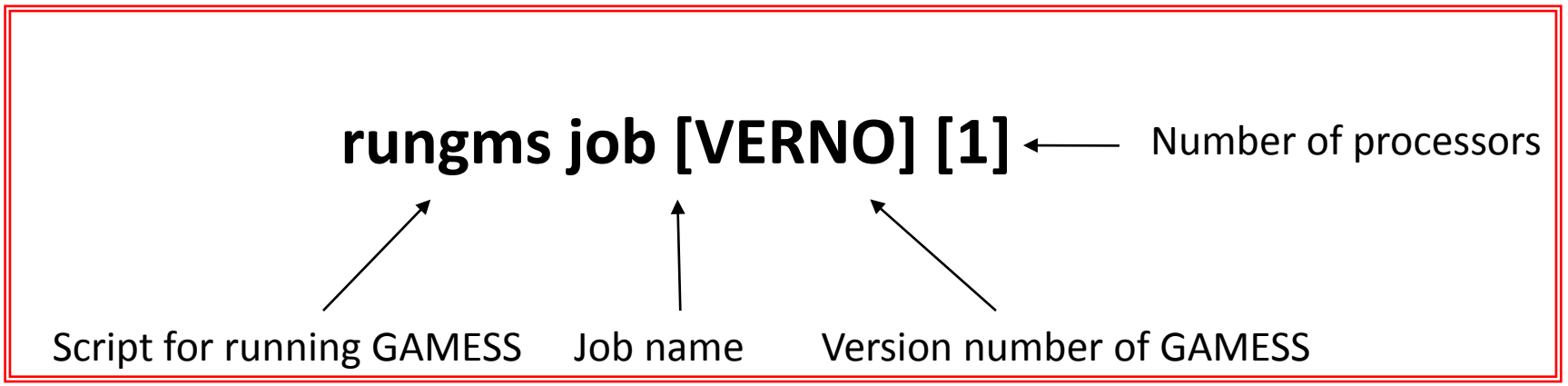


For a “XXX.xmi” file, you must prepare corresponding “XXX.inp” file
The job name “XXX” for the two input files should be the same.

GAMESS

XMVB module

job.inp job.xmi



job.out (or job.log), job.dat

job .xmo, job.xdat, job.orb, job.den

Useful documents

GAMESS manual

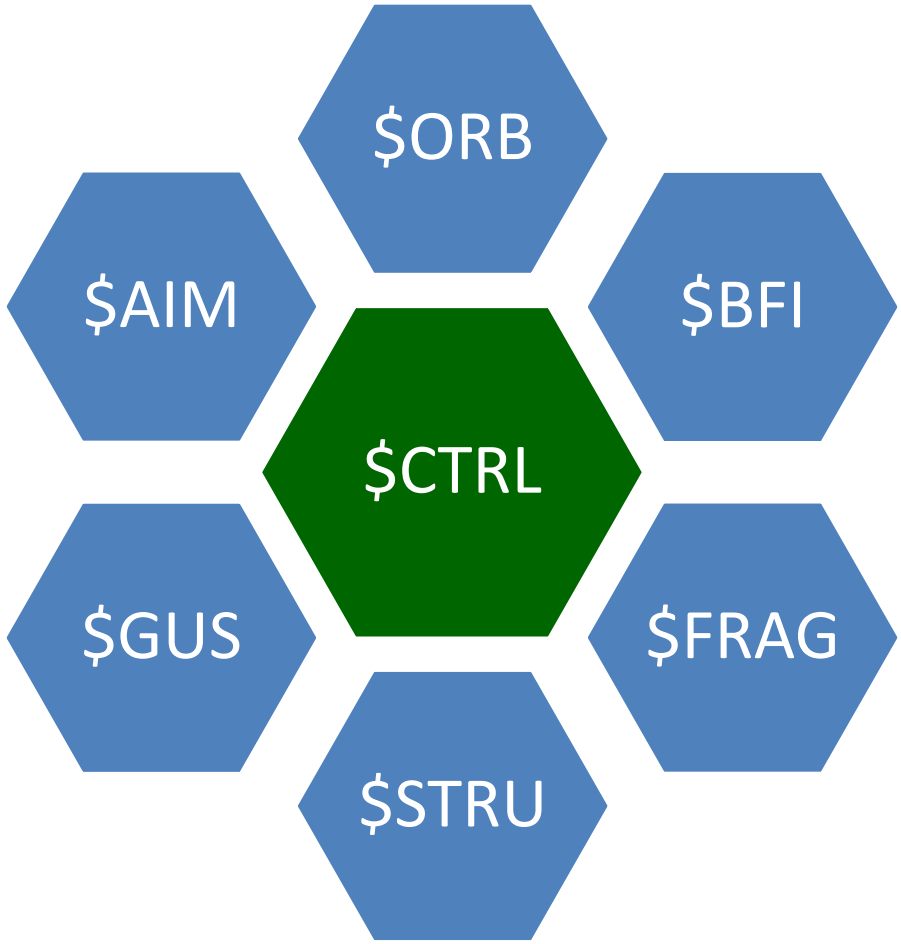
http://www.msg.ameslab.gov/gamess/documentation.html

XMVB manual

http://ftcc.xmu.edu.cn/xmvb/downloads.html

How to write input files?

Structure of Typical XMVB Input File



Optional

Necessary

Example 1. H_2 molecule

h2.inp

```
$contrl runtyp=energy scftyp=rhf $end
$contrl vbtyp=xmvb $end
$basis gbasis=n31 ngauss=6 NPFUNC=1 $end
$data
H2 molecule
C1
H   1.0   0.000000  0.000000  0.000000
H   1.0   0.000000  0.000000  0.742000
$end
```

→ rhf for mult=1
rohf or uhf for mult >1

→ Basis set: 6-31G(d,p)

The detailed information of GAMESS input file is available in GAMESS manual.

```
$ctrl
nstr=3  iscf=5 nao=2 nae=2 boys
orbtyp=hao frgtyp=sao
$end
$stru
1 2
1 1
2 2
$end
$frag
1 1
spz 1
spz 2
$end
$orb
1 1
1
2
$end
```

iscf : orbital optimization algorithm
nao: number of active orbitals
nae: number of active electrons

iscf	Gradient	Performance	Required keywords
1	Numerical	Stable, slow	
2	Analytical	Fast	
3	Numerical	Stable, slow	
4	Numerical	Stable, very slow	
5	Analytical	Very Fast	nao, nae

```
$ctrl
nstr=3  iscf=5 nao=2 nae=2 boys
orbtyp=hao frgtyp=sao
$end
$stru
1 2
1 1
2 2
$end
$frag
1 1
spz 1
spz 2
$end
$orb
1 1
1
2
$end
```

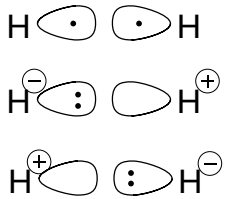
Boys: performing boys localization, which is required for VBCI calculations.

```
$ctrl
nstr=3  iscf=5 nao=2 nae=2 boys
orbtyp=hao frgtyp=sao
$end
$stru
1 2
1 1
2 2
$end
$frag
1 1
spz 1
spz 2
$end
$orb
1 1
1
2
$end
```

nstr =3, the number of structures
$str : describing VB structure or determinant.
In $str, a typical structure is written as:

a a b b c c … m n

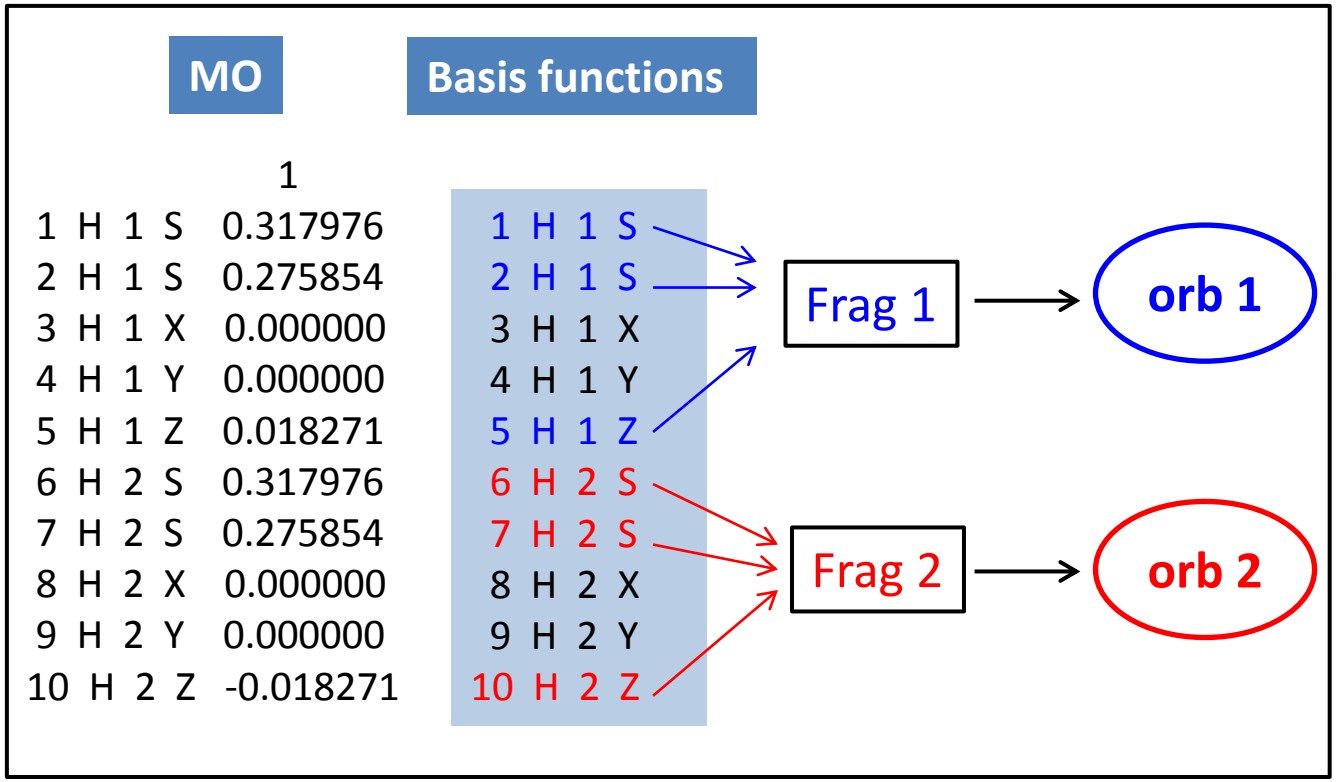
a b c : double occupied orbitals (inactive)
m n : single occupied orbitals (active)



1 2
1 1
2 2

```
$ctrl
nstr=3 iscf=5 nao=2 nae=2 boys
orbtyp=hao frgtyp=sao
$end
$stru
1 2
1 1
2 2
$end
$frag
1 1
spz 1
spz 2
$end
$orb
1 1
1
2
$end
```

frgtyp=sao, fragments are defined with **s**ymmetrized **a**tomic **o**rbitals.



```
$ctrl
nstr=3 iscf=5 nao=2 nae=2 boys
orbtyp=hao frgtyp=sao
$end
$stru
1 2
1 1
2 2
$end
$frag
1 1
spz 1
spz 2
$end
$orb
1 1
1
2
$end
```

$frag: description for fragments
Line 1: the number of atoms in each fragment.
Line 2 : the basis functions of s and pz in atom 1
Line 3 : the basis functions of s and pz in atom 2.

$orb: description for VB orbitals
Line 1: the number of fragments in each orbital.
Line 2: orbital 1 is described by fragment 1
Line 3: orbital 2 is described by fragment 2.

Example 2. HF molecule

hf.inp

```
$contrl runtyp=energy scftyp=rhf $end
$contrl vbtyp=xmvb $end
$basis gbasis=n31 ngauss=6 $end
$data
HF molecule 6-31G basis set
C1
H   1.0   0.000000  0.000000  0.000000
F   9.0   0.000000  0.000000  0.917000
$end
```

HF molecule, 3 structures

hf.xmi

```
$ctrl
str=full nae=2 nao=2 iprint=3 orbtyp=hao
frgtyp=atom iscf=5 guess=mo boys
$end
$orb
1 1 1 1 1 1
2
2
2
2
2
1
$end
$gus
1 1
2 2
3 4
4 5
5 3
6 3
$end
```

Automatically generate structures based on the active orbitals and electrons.

str = full => generate all VB structures

cov => only covalence

Ion => only ionic structures

nao = m, the final m orbitals are selected as active orbitals.

nae = n, the number of active VB electrons

Iprint=3, full print-out message.

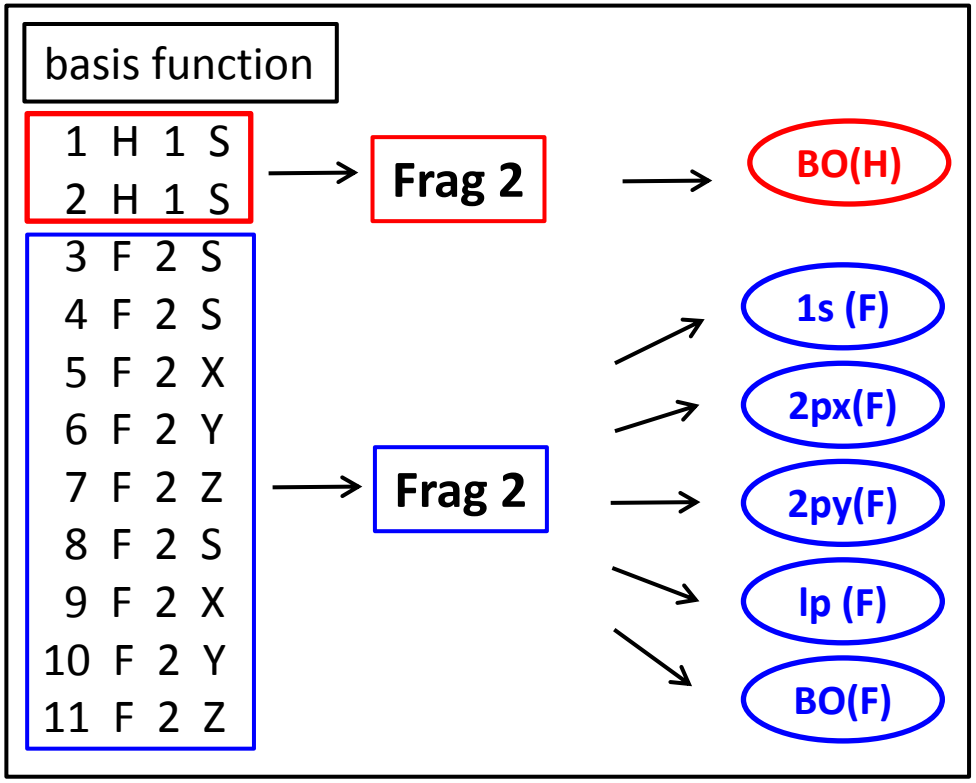
Fragtyp=atom and without $frag means that the orbitals will be determined by the atom directly.

Using str, $str can be absent.

HF molecule, 3 structures hf.xmi

```
$ctrl
str=full nae=2 nao=2 iprint=3 orbtyp=hao
frgtyp=atom iscf=5 guess=mo boys
$end
$orb
1 1 1 1 1 1
2
2
2
2
2
1
$end
$gus
1 1
2 2
3 4
4 5
5 3
6 3
$end
```

frgtyp=atom
The fragments of system are defined with atoms



HF molecule, 3 structures hf.xmi

```
$ctrl
str=full nae=2 nao=2 iprint=3 orbtyp=hao
frgtyp=atom iscf=5 guess=mo boys
$end
$orb
1 1 1 1 1 1
2
2
2
2
2
1
$end
$gus
1 1
2 2
3 4
4 5
5 3
6 3
$end
```

$orb is required when orbtyp=hao
The first line shows that there are six orbitals.
While the first 5 orbitals are expanded by the basis functions of F atom. (frag 2)
The last orbitals are expanded by the basis functions of H atom. (frag 1)

HF molecule, 3 structures hf.xmi

```
$ctrl
str=full nae=2 nao=2 iprint=3 orbtyp=hao
frgtyp=atom iscf=5 guess=mo boys
$end
$orb
1 1 1 1 1 1
2
2
2
2
2
1
$end
$gus
1 1
2 2
3 4
4 5
5 3
6 3
$end
```

guess=mo : Initial guess of VB orbitals will be obtained directly from molecular orbitals

VB orb		MO
1	1s orbital of F atom	1
2	Lone pair of F atom	2
3	px pair of F atom	4
4	py pair of F atom	5
5	H-F bonding orbital on F atom	3
6	H-F bonding orbital on H atom	3

				1	2	3	4	5
				-26.2757	-1.5897	-0.7391	-0.631	-0.631
				A	A	A	A	A
1	H	1	S	0.000301	0.132423	-0.293056	0.000000	0.000000
2	H	1	S	0.001771	-0.013969	-0.107756	0.000000	0.000000
3	F	2	S	0.995513	-0.230516	-0.061755	0.000000	0.000000
4	F	2	S	0.021368	0.497220	0.137028	0.000000	0.000000
5	F	2	X	0.000000	0.000000	0.000000	0.654168	0.091145
6	F	2	Y	0.000000	0.000000	0.000000	-0.091145	0.654168
7	F	2	Z	-0.001488	-0.084307	0.548328	0.000000	0.000000
8	F	2	S	-0.005826	0.529428	0.244193	0.000000	0.000000
9	F	2	X	0.000000	0.000000	0.000000	0.485512	0.067646
10	F	2	Y	0.000000	0.000000	0.000000	-0.067646	0.485512
11	F	2	Z	0.001400	-0.047087	0.349640	0.000000	0.000000

Example 3. L-BOVB calculation for HF

Using VBSCF orbitals as guess orbitals

cp hf.orb hf-bovb.gus

Preparing GAMESS and XMVB input files for BOVB

hf-bovb.inp

hf-bovb.xmi

Running XMVB

hf-bovb.xmi

```
HF molecule, 3 structures
$ctrl
str=full nae=2 nao=2 bovb
iprint=3 orbtyp=hao frgtyp=atom iscf=1
guess=read
$end
$orb
1 1 1 1 1 1
2
2
2
2
2
1
$end
```

iscf = 4, 5 can not be used for BOVB calculations.

Example 4. VBCI calculation for HF

a VBSCF calculation with boyslocalization using the orbitals as guess orbitals

cp hf.orb hf-vbci.gus

Preparing GAMESS and XMVB input files for VBCI

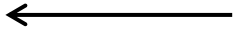
hf-vbci.xmi
hf-vbci.inp

Running XMVB.

hf-vbci.xmi

hf.xmi

F2 molecule
$ctrl
nstr=3 boys
$end
…
…
…



HF molecule
$ctrl
str=full nae=2 nao=2 vbcisd
iprint=3 orbtyp=hao frgtyp=atom guess=read
$end
$orb
1 1 1 1 1 1
2
2
2
2
2
1
$end

Example 5. VBPCM calculation for HF

Using VBSCF orbitals as guess orbitals

cp hf.orb hf-pcm.gus

Preparing GAMESS and XMVB input files for VBCI

hf-pcm.inp

hf-pcm.xmi

Running XMVB

hf-pcm.inp

```
$contrl runtyp=energy scftyp=rhf $end
$contrl vbtyp=xmvb $end
$pcm solvnt=water $end
$basis gbasis=n31 ngauss=6 $end
$data
HF molecule 6-31G basis set
C1
H   1.0   0.000000  0.000000  0.000000
F   9.0   0.000000  0.000000  0.916800
$end
```

hf-pcm.xmi

```
HF molecule, 3 structures
$ctrl
str=full nae=2 nao=2 vbpcm
iprint=3 orbtyp=hao frgtyp=atom
iscf=5 guess=read
$end
$orb
1 1 1 1 1 1
2
2
2
2
2
1
$end
```

How to get information from output files?

Information of XMO file

- Input message → Input file, Integrals, Guess etc
- SCF iterative procedure (final energy) →
  Iteration ** E = **.** DE = **
  Gxnorm = ** Max Comp. = **
  icovg= 1
  Total Energy: **
- S_{ij} and H_{ij} Matrices
- Information of VB wavefunction →
  Coefficients of structures and determinants
  Weights of structures
  Optimized orbitals
- Properties and analysis → Bond order, atomic charge, dipole moments

Optional

Default

The VB calculation message in gamess output file is not correct

hf.xmo

```
ITER   20 E =    -100.002122
ICOVG =                  1
Boys localizing
...
Total Energy:    -100.00212152
```

SCF procedures (final energy)

```
******  MATRIX  OF  OVERLAP  ******

          1          2          3
 1    1.000000   0.630143   0.630143
 2    0.630143   1.000000   0.247723
 3    0.630143   0.247723   1.000000

******  MATRIX  OF  HAMILTONIAN ******

          1          2          3
 1   -2.370630  -1.835188  -1.509357
 2   -1.835188  -2.248158  -0.822557
 3   -1.509357  -0.822557  -1.314986
```

S_{ij} and H_{ij} matrices

```
******  WEIGHTS OF STRUCTURES ******

   1    0.56496  ******  1:4   5  6
   2    0.42059  ******  1:4   5  5
   3    0.01445  ******  1:4   6  6
```

Weights of VB structures

	1	2	3	4	5	6
1	0.000000	0.000000	0.000000	0.000000	0.000000	-0.835424
2	0.000000	0.000000	0.000000	0.000000	0.000000	-0.227552
3	1.012748	-0.084340	0.084340	-0.084340	0.049531	0.000000
4	-0.025976	0.300588	-0.300588	0.300588	-0.165593	0.000000
5	0.000000	-0.020440	-0.477373	-0.456933	0.000000	0.000000
6	0.000000	0.539422	0.252010	-0.287413	0.000000	0.000000
7	-0.010873	0.069765	-0.069765	0.069765	0.701583	0.000000
8	-0.054032	0.316476	-0.316475	0.316476	-0.042957	0.000000
9	0.000000	-0.015134	-0.353442	-0.338309	0.000000	0.000000
10	0.000000	0.399383	0.186585	-0.212797	0.000000	0.000000
11	-0.006949	0.056493	-0.056493	0.056493	0.419452	0.000000

Optimized orbitals

****** POPULATION AND CHARGE ******

ATOM	MULL.POP.	CHARGE	LOW.POP.	CHARGE
1 H	0.593861	0.406139	0.675425	0.324575
2 F	9.406139	-0.406139	9.324575	-0.324575

XMVB atomic population analysis

****** BOND ORDER ******

ATOM 1	ATOM 2	DIST	BOND ORDER
1 H	2 F	0.917	0.794

The adaptability of orbital optimization algorithms

	ISCF				
	1	2	3	4	5
VBSCF	√	√	√	√	√
BOVB	√	√	√		
VBSCF/PCM	√	√	√		√
VBSCF/EFP	√	√	√		√
BOVB/PCM	√	√	√		