ELECTRONIC SUPPLEMENTARY INFORMATION

OCT. 8, 2018

Extrapolation of High-Order Correlation Energies: The WMS Model

Yan Zhao,[a] Lixue Xia,[a] Xiaobin Liao,[a] Qiu He,[a] Maria X. Zhao,[b] and Donald G. Truhlar[c]

[a]*State Key Laboratory of Silicate Materials for Architectures,International School of Materials Science and Engineering, Wuhan University of Technology, Wuhan 430070, People's Republic of China.*

[b]*Lynbrook High School, 1280 Johnson Avenue, San Jose, California 95129*

[c]*Department of Chemistry and Supercomputing Institute, University of Minnesota, 207 Pleasant Street S.E., Minneapolis, MN 55455-0431*

TABLE OF CONTENTS

Table S1: Calculated classical barrier heights (kcal/mol) for the DBH24-W4 database (The ZPE contributions are excluded.)

Reactions		Best Est.	WMS
Hydrogen Atom Transfers			
$OH + CH_4 \rightarrow CH_3 + H_2O$	$\Delta E_f^{\neq}$	6.35	6.25
	$\Delta E_r^{\neq}$	19.26	19.28
$H + OH \rightarrow O + H_2$	$\Delta E_f^{\neq}$	10.77	10.88
	$\Delta E_r^{\neq}$	13.17	13.24
$H + H_2S \rightarrow H_2 + HS$	$\Delta E_f^{\neq}$	3.69	3.74
	$\Delta E_r^{\neq}$	17.75	17.84
Heavy-Atom Transfers			
$H\cdot + N_2O \rightarrow OH\cdot + N_2$	$\Delta E_f^{\neq}$	17.13	17.26
	$\Delta E_r^{\neq}$	82.47	82.15
$H\cdot + ClH \rightarrow HCl + H\cdot$	$\Delta E_f^{\neq}$	17.47	17.46
	$\Delta E_r^{\neq}$	17.47	17.46
$CH_3 + FCl \rightarrow CH_3F + Cl$	$\Delta E_f^{\neq}$	6.75	6.56
	$\Delta E_r^{\neq}$	60.00	59.71
S_N2 Reactions			
$Cl^-\cdots CH_3Cl \rightarrow ClCH_3\cdots Cl^-$	$\Delta E_f^{\neq}$	13.41	13.37
	$\Delta E_r^{\neq}$	13.41	13.37
$F^-\cdots CH_3Cl \rightarrow FCH_3\cdots Cl^-$	$\Delta E_f^{\neq}$	3.44	3.48
	$\Delta E_r^{\neq}$	29.42	29.20
$OH^- + CH3F \rightarrow HOCH_3 + F^-$	$\Delta E_f^{\neq}$	-2.44	-2.67
	$\Delta E_r^{\neq}$	17.66	17.46
Unimolecular and Association Reactions			
$H\cdot + N_2 \rightarrow HN_2\cdot$	$\Delta E_f^{\neq}$	14.36	14.29
	$\Delta E_r^{\neq}$	10.61	10.71
$H\cdot + C_2H_4 \rightarrow CH_3CH_2\cdot$	$\Delta E_f^{\neq}$	1.83	1.94
	$\Delta E_r^{\neq}$	41.84	41.85
$HCN \rightarrow HNC$	$\Delta E_f^{\neq}$	48.07	48.18
	$\Delta E_r^{\neq}$	32.82	32.93

Note: ‡ denotes the conventional transition state structure (saddle point); f denotes forward, and r dentotes reverse.

Table S2. RMSE of the scalar relativistic contribution in WMS (kcal/mol) [a]

Basis sets	α	MR17	SR183
		RMSE	RMSE
jul-cc-pV{DT}Z-DK	2.0	0.05	0.07

[a] The reference scalar relativistic contribution were taken from the W4-17 paper,[39] and they are obtained from the difference between nonrelativistic CCSD(T)/aug'-cc-pV(Q+d)Z and relativistic CCSD(T)/aug'-cc-pV(Q+d)Z-DK calculations. RMSE = root mean square error = root mean square deviation (RMSD)

Table S3. RMSE of the composite methods for the W4-17 database (kcal/mol)[a]

Method	Ref. [b]	MR17	SR183	ARMSE [c]
		RMSE	RMSE	
W3X-L	This work	0.69	0.44	0.56
WMS	This work	0.74	0.39	0.57
W3X	This work	0.97	0.69	0.83
W2-F12	[39]	1.39	0.55	0.97
ccCA-PS3	[39]	1.20	0.92	1.06
W2X	[39]	1.65	0.63	1.14
W1-F12	[39]	1.76	0.72	1.24
G4	[39]	1.71	0.95	1.33
G3SXMP3	This work	1.59	1.43	1.51
ROG4(MP2)-6X	[39]	1.60	1.65	1.63
MCG3-MPW	This work	1.90	1.42	1.66
MCG3/3	This work	2.55	1.51	2.03
ROCBS-QB3	[39]	2.55	2.04	2.30
MCQCISD-MPW	This work	2.89	2.16	2.53
G3	[39]	4.26	1.88	3.07
G2	This work	3.68	2.56	3.12
DL-jun[e]	This work	2.25	3.79	3.02
BMC-CCSD	This work	4.63	1.81	3.22

[a] The reference TAEs were taken from the W4-17 paper,[39] and they are obtained by using the W4 protocol and its variants. MSE=mean signed error = mean signed deviation (MSD); MSE=mean signed error = mean absolute deviation (MAD); RMSE=root mean square error = root mean square deviation (RMSD). Note that the average number of bonds per molecule is 2.24 for MR17 and 4.14 for the

[b] The reference for the performance data; see section 2 of thearticle proper for referecnes for the methods.

[c] ARMSE is the average of the RMSEs of the two databases.

[d] Computational time for thiophene normalized by the HF/aug-VDZ calculation. N. A. denotes "not available).

[e] DL-jun = E(MP2-F12/jun-V(T+d)Z) + [E(CCSD(T)-F12a) – E(MP2-F12)]/jun-V(D+d)Z

Table S4: Calculated Born-Oppenheimer atomization energy (kcal/mol) for the W4-17 database (The ZPE contributions are excluded)

Molecule	Molecular Formula	W4-17 Reference	WMS
MR17			
boron nitride	BN ($^1\Sigma^+$)	105.0	106.4
ozone	O_3	146.5	146.0
dioxygen difluoride	F_2O_2	151.0	151.9
dioxygen chloride	OOCl•	124.9	123.9
dioxygen fluoride	OOF•	133.7	133.1
chlorine pentafluoride	ClF_5	183.3	183.5
diboron	B_2	67.4	66.5
difluorine monoxide	FOF	92.7	93.2
chlorine trioxide	OClOO	166.7	166.3
chlorine dioxide	O_2Cl•	126.0	126.1
dicarbon	C_2	146.7	147.0
oxygen monofluoride	FO•	52.7	52.6
phosphorus trifluoride	ClF_3	126.3	126.8
dioxygen dichloride	ClOOCl	146.7	147.8
tetrasulfur	S3	231.4	230.1
dichlorine monoxide	ClOCl	99.3	99.7
trisulfur	S_3	166.1	165.6
SR183			
difluoride	F_2	38.3	38.6
chlorine monoxide	ClO•	64.6	64.4
dinitrogen tetraoxide	N_2O_4	469.8	470.8
nitrosyl chloride	ClNO	190.7	190.0
nitrogen dioxide	O_2N•	227.0	226.8
chlorine monofluoride	ClF	61.4	61.4
diphosphorus	P_2	117.3	116.2
chloric acid	$HOClO_2$	270.4	270.6
disulfur oxide	SSO	206.7	206.1
perchloric acid	$HClO_4$	331.0	329.9
hydrotrioxy radical	trans-HO_3•	232.3	232.1
nitrosyl fluoride	FNO	214.6	214.5

dioxygen	O_2	120.2	120.0
dichlorine	Cl_2	57.9	57.7
tetraphosphorus	P_4	289.9	289.1
chlorous acid	HOClO	202.4	202.5
disulfur	S_2	102.8	102.3
hydrotrioxy radical	cis-HO_3•	232.1	232.1
nitrous oxide	N≡N-O	270.2	270.0
sulfur monoxide	SO	125.4	125.5
nitric oxide	ON•	152.5	152.3
sulfur dioxide	SO_2	258.7	258.6
boron nitride	BN ($^3\Pi$)	105.7	104.8
carbon dichloride	Cl_2C:	175.3	174.9
sulfur trioxide	SO_3	343.8	343.5
cyano radical	N≡C•	181.1	180.6
carbon sulfide	CS	171.4	171.1
hydrogen azide	HN_3	331.2	330.9
carbon disulfide	CS_2	278.9	278.5
nitrous acid	trans-HON=O	311.7	311.8
nitrous acid	cis-HON=O	311.3	311.4
hypofluorous acid	HOF	157.8	158.1
carbon tetrachloride	CCl_4	311.3	311.7
hexachloroethane	C_2Cl_6	555.0	556.6
nitrosyl hydride	HN=O	205.4	205.4
formonitrile oxide	HCNO	364.0	364.0
hydroperoxy radical	HOO•	174.8	175.0
tetrachloroethylene	C_2Cl_4	467.1	468.3
cyanogen chloride	ClC≡N	284.0	283.6
phosgene	CCl_2O	344.5	344.5
silicon monoxide	SiO	192.1	191.7
carbon oxide sulfide	OCS	334.3	334.1
sulfur hexafluoride	SF_6	479.4	480.2
dinitrogen monohydride	HN≡N•	224.6	224.4
dinitrogen	N_2	228.3	227.7
cyanogen	N≡C–C≡N	501.3	500.3
dichloroacetylene	C_2Cl_2	347.9	348.0
hypochlorous acid	HOCl	164.8	164.9

isofulminic acid	HONC:	349.3	349.2
1,2,5-oxadiazole	$H_2C_2N_2O$	704.9	705.4
hydrogen disulfide radical	HSS•	163.5	163.2
fluoromethylidyne	FC•	132.2	132.0
carbonyl chloride fluoride	ClCOF	381.2	381.3
cis-dichlorodifluoroethene	cis-$C_2F_2Cl_2$	525.3	526.0
dichlorodifluoromethane	CF_2Cl_2	388.9	389.3
trans-dichlorodifluoroethene	trans-$C_2F_2Cl_2$	525.7	526.4
dioxirane	H_2CO_2	409.1	409.3
difluoroacetylene	FC≡CF	384.4	384.4
carbon dioxide	CO_2	389.1	388.7
carbon difluoride	F_2C:	257.6	257.5
diazene	cis-H_2N_2	290.8	290.8
chloroform	CCl_3H	342.2	342.4
isocyanic acid	HNCO	433.8	433.5
trans-diazene	trans-N_2H_2	296.2	296.1
hydrogen peroxide	H_2O_2	268.3	268.6
cyanic acid	HOCN	409.2	408.9
carbonyl fluoride	F_2C=O	418.9	419.2
perfluoroethene	C_2F_4	587.3	588.0
oxirene	H_2C_2O	455.2	455.1
carbon monoxide	CO	259.3	258.7
formyl radical	H(O=)C•	278.7	278.5
chloroethyne	C_2ClH	377.0	376.9
hydrogen isocyanide	HNC:	297.8	297.5
ethynyl radical	HC≡C•	265.7	265.2
hydrogen cyanide	HCN	313.1	312.6
dihydronitroxide radical	H_2NO	274.3	274.5
formic_anhydride	$H_2C_2O_3$	755.3	755.2
1,3-dioxetan-2-one	$H_2C_2O_3$	751.3	751.5
hexafluoroethane	C_2F_6	788.1	789.4
fluoroamine	NH_2F	255.8	255.9
fluoroacetylene	HC≡CF	397.4	397.2
tetrafluoromethane	CF_4	476.3	477.1
chloramine	H_2NCl	246.8	246.8

formylfluoride	HC(=O)F	402.5	402.6
phosphorus trifluoride	PF_3	363.5	363.3
glyoxal	H(O=)C–C(=O)H	633.8	633.5
iminometyl radical	HN=CH•	335.8	335.6
phosphorus pentafluoride	PF_5	556.1	556.7
formic acid	HC(=O)OH	500.8	500.8
silicon fluoride	FSi•	141.9	141.4
cyanomethyl	H_2CCN•	512.1	511.4
ketene	$H_2C=C=O$	532.6	532.2
aluminum monochloride	AlCl	121.3	120.6
thiophene	H_4C_4S	963.0	963.5
hydroxylamine	NH_2OH	359.0	359.3
hydroxymethylene	cis-H(HO)C:	317.0	316.8
fluoroform	CHF_3	457.6	458.1
aluminium chloride	$AlCl_3$	308.6	306.8
formamide	CH_3NO	567.2	567.1
furan	H_4C_4O	993.7	993.8
hydroxymethylene	trans-H(HO)C:	321.8	321.6
cyclobutadiene	C_4H_4	819.8	819.4
1,3-dithiotane	$H_4C_2S_2$	682.0	682.7
dichloromethane	CCl_2H_2	369.4	369.4
boron monofluoride	BF	182.0	181.4
beta-lactim	H_5C_3NO	992.4	992.6
methanimine radical	$H_2C=N$•	343.4	343.2
tetrahedrane	$C4H_4$	792.4	792.5
pyrrole	H_5C_4N	1071.5	1071.7
formaldehyde	$H_2C=O$	374.0	373.8
acetylene	C_2H_2	405.1	404.7
1,3-dioxetane	$H_4C_2O_2$	748.5	748.6
chlorofluoromethane	CH_2ClF	401.0	401.0
acetic acid	$CH_3C(=O)OH$	802.6	802.5
aluminum monofluoride	AlF	162.9	162.6
benzene	C_6H_6	1367.3	1367.6
methanimine	$H_2C=NH$	439.0	438.9

hydrazine	N_2H_4	437.7	437.8
vinyl chloride	C_2ClH_3	542.8	542.7
vinylidene	$H_2C{=}C{:}$	359.6	359.1
borole	H_5BC_4	1056.6	1056.1
aluminium fluoride	AlF_3	428.3	428.2
difluoromethane	CF_2H_2	436.6	436.7
oxirane	C_2H_4O	650.5	650.5
cyclopentadiene	H_6C_5	1180.6	1180.7
cyclopropene	C_3H_4	681.2	681.0
silole	H_6C_4Si	1098.8	1099.0
fluoroethylene	C_2H_3F	572.8	572.7
trifluoroborane	BF_3	469.1	468.9
tetrafluorosilane	SiF_4	573.9	573.8
allene	C_3H_4	703.4	703.3
acetaldehyde	$H_3C{-}C(=O)H$	676.9	676.7
propyne	C_3H_4	704.8	704.5
trans-butadiene	C_4H_6	1012.1	1012.1
cyclobutene	C_4H_6	1000.5	1000.5
oxetane	H_6C_3O	946.9	946.9
hydroxyl radical	$HO\bullet$	107.1	107.1
vinyl radical	$H_2C{=}HC\bullet$	445.6	445.4
hydrogen fluoride	HF	141.1	141.2
water	H_2O	232.5	232.6
difluoroborane	HBF_2	409.7	409.5
allyl radical	$C_3H_5\bullet$	766.1	765.7
aminomethyl radical	$H_2N{-}CH_2\bullet$	481.7	481.7
imidogen radical	NH	83.0	82.9
cyclopropyl radical	$C_3H_5\bullet$	736.9	736.7
amino radical	$H_2N\bullet$	182.4	182.3
hydrogen chloride	HCl	106.4	106.4
methanol	CH_3OH	512.7	512.8
chloromethane	$CClH_3$	394.7	394.7
ethene	C_2H_4	563.6	563.4

chloroethane	C_2ClH_5	691.3	691.4
methylamine radical	$H_3C–NH•$	474.2	474.1
ethanol	C_2H_5OH	810.2	810.2
ammonia	NH_3	297.8	297.7
propene	C_3H_6	860.8	860.6
cyclobutane	C_4H_8	1149.3	1149.4
cyclopropane	C_3H_6	853.1	853.0
fluoromethane	CH_3F	422.1	422.1
methylamine	CH3NH2	581.7	581.8
fluoroethane	C2H5F	720.4	720.3
mercapto radical	HS•	87.5	87.4
hydrogen sulfide	H2S	183.0	182.9
n-pentane	C_5H_{12}	1595.9	1596.1
methylphosphine	CH_3PH_2	538.6	538.5
n-butane	C_4H_{10}	1301.5	1301.5
singlet methylene	H_2C: (^{1}A1)	181.3	181.1
methylidyne	HC•	84.1	84.1
propane	C_3H_8	1007.1	1007.0
ethane	C_2H_6	712.5	712.4
phosphine	PH_3	241.8	241.8
fluorosilane	SiH_3F	381.0	381.1
diborane	B_2H_6	606.8	606.4
methane	CH_4	420.2	420.2
methyl radical	$H_3C•$	307.6	307.5
disilane	Si_2H_6	533.7	534.0
silicon hydride	HSi•	73.6	73.5
triplet methylene	H_2C: (^{3}B1)	190.5	190.4
boron monohydride	BH	84.9	84.9
borane	BH_3	281.2	280.9
silane	SiH4	323.9	323.9
aluminum hydride	AlH	73.3	73.4
aluminum trihydride	AlH_3	212.5	212.7
dihydrogen	H2	109.5	109.6

Table S5: Example input file for a WMS calculation: the H_2O molecule

```
***,h2o
memory,1900,m
Set,Charge=0,Spin=0;
geomtyp=xyz
geometry={
3
h2o
O  0.000000  0.000000  0.117790
H  0.000000  0.755453  -0.471161
H  0.000000  -0.755453  -0.471161
}
basis,aug-cc-pV(D+d)Z, H=VDZ
HF
{ccsd(t)-f12,gem_beta=0.9 ,df_basis=vdz/mp2fit,df_basis_exch=vdz/jkfit,ri_basis=vdz/jkfit}
SHOW,ENERG*,EMP2*,EF12*

basis,aug-cc-pV(T+d)Z, H=VTZ
HF
{ccsd(t)-f12,gem_beta=1.0,df_basis=vtz/mp2fit,df_basis_exch=vtz/jkfit,ri_basis=vtz/jkfit}
SHOW,ENERG*,EMP2*,EF12*

basis, cc-pwCVDZ
HF;
ccsd(t)
SHOW,ENERG*,EMP2*
HF;
{ccsd(t); Core,c1,c2,c3,c4,c5,c6,c7,c8;}
SHOW,ENERG*,EMP2*

basis, cc-pwCVTZ
HF;
RMP2;
SHOW,ENERG*,EMP2*
HF;
{RMP2; Core,c1,c2,c3,c4,c5,c6,c7,c8;}
SHOW,ENERG*,EMP2*

basis, AVDZ-DK, h=VDZ-DK
dkroll=1
HF
RMP2;
SHOW,ENERG*,EMP2*

basis, AVDZ, h=VDZ
dkroll=0
HF
RMP2;
SHOW,ENERG*,EMP2*

basis, AVTZ-DK, h=VTZ-DK
dkroll=1
HF
RMP2;
SHOW,ENERG*,EMP2*
```

```
basis, AVTZ, h=VTZ
dkroll=0
HF
RMP2;
SHOW,ENERG*,EMP2*

---
```

Table S6: Example input file for a WMS calculation: the CH3 radical

```
***,ch3
memory,1900,m
Set,Charge=0,Spin=1;
geomtyp=xyz
geometry={
4
ch3
C  0.000000  0.000000  0.000000
H  0.000000  1.077700  0.000000
H  0.933316  -0.538850  0.000000
H  -0.933316  -0.538850  0.000000
}
basis,aug-cc-pV(D+d)Z, H=VDZ
HF
{uccsd(t)-f12,gem_beta=0.9,df_basis=vdz/mp2fit,df_basis_exch=vdz/jkfit,ri_basis=vdz/jkfit}
SHOW,ENERG*,EMP2*,EF12*

basis,aug-cc-pV(T+d)Z, H=VTZ
HF
{uccsd(t)-f12,gem_beta=1.0,df_basis=vtz/mp2fit,df_basis_exch=vtz/jkfit,ri_basis=vtz/jkfit}
SHOW,ENERG*,EMP2*,EF12*

basis, cc-pwCVDZ
HF;
uccsd(t)
SHOW,ENERG*,EMP2*
HF;
{uccsd(t); Core,c1,c2,c3,c4,c5,c6,c7,c8;}
SHOW,ENERG*,EMP2*

basis, cc-pwCVTZ
HF;
RMP2;
SHOW,ENERG*,EMP2*
HF;
{RMP2; Core,c1,c2,c3,c4,c5,c6,c7,c8;}
SHOW,ENERG*,EMP2*

basis, AVDZ-DK, h=VDZ-DK
dkroll=1
HF
RMP2;
SHOW,ENERG*,EMP2*

basis, AVDZ, h=VDZ
dkroll=0
HF
RMP2;
SHOW,ENERG*,EMP2*

basis, AVTZ-DK, h=VTZ-DK
dkroll=1
HF
```

```
RMP2;
SHOW,ENERG*,EMP2*

basis, AVTZ, h=VTZ
dkroll=0
HF
RMP2;
SHOW,ENERG*,EMP2*

---
```

Table S7. perl scripts for performing WMS calculations

We have employed perl scripts for the WMS calculations. mWMS.pl is a script for preparing the *Molpro* input files for the WMS methods. It takes a geometry input file in xyz format, and it generates a *Molpro* input file with an extension .com.

The example xyz input for H_2O is as follows:

h2o.xyz

```
3
0  1
O  0.000000  0.000000  0.117790
H  0.000000  0.755453  -0.471161
H  0.000000  -0.755453  -0.471161
```

The first line in h2o.xyz is the number of atoms in the molecule, and the second line gives the charge and spin multiplicity of the molecule. The Cartesian coordinates of the ayoms begin on the third line.

If one issues the following command:

```
perl mWMS.pl h2o.xyz
```

a file named h2o_WMS.com that has the same content as in Table S10 will be generated by the mWMS.pl script. The user can invoke the *Molpro* program to do the WMS calculations, for example

```
molpro –n 4 h2o_WMS.com
```

After the *Molpro* calculation is done, the *Molpro* program will generate the h2o_WMS.out output file. One can extract the WMS energies by using the rWMS.pl script:

```
perl rWMS.pl h2o_WMS.out
```

The mWMS.pl and rWMS.pl are given in the following section, and they are included in the *WMSPac* software. The *WMSPac* software contains more perl scripts to generate the WMS input files and the W2X/W3X/W3X-L input files. For example, one can take a *Gaussian* output file or *Molpro* output file as the geometry input to generate the WMS input files. The *WMPac* software is available from the following website:

```
http://truhlar.chem.umn.edu/content/software
```

mWMS.pl

```
#!/usr/bin/perl
#
# Usage:
#      perl mWMS.pl h2o.xyz
#

$mem='memory,800,m';

$workdir=`pwd`;
chomp $workdir;
$basis1 = 'basis, aug-cc-pV(D+d)Z, H=VDZ';
```

```
$basis2 = 'basis, aug-cc-pV(T+d)Z, H=VTZ';
$basis3 = 'basis,cc-pwCVDZ';
$basis4 = 'basis,cc-pwCVTZ';
$WM = 'WMS';

for $i (0 .. @ARGV-1){
  $name = $ARGV[$i];
  @out=split(/\./, $name);
  open(FILE2,">$out[0]_$WM.com") || die "can't open $!";
  print FILE2 "***,$out[0]\n";
  print FILE2 "$mem\n";

  open(FILE4, "<$name") || die "can't open $!";
  @lines = (<FILE4>);
  close FILE4;
  print "$WM \n";

  @cm = split(/\s+/, $lines[1]);
  @temp = split(/\s+/, $lines[0]);
  $noa = $temp[0];
  $spin = $cm[1]-1 ;
  print FILE2 "Set,Charge=$cm[0],Spin=$spin;\n";
  print FILE2 "geomtyp=xyz\n";
  print FILE2 'geometry={';
  print FILE2 "\n$noa \n";
  print FILE2 "$out[0]\n";
  for $i (2 .. $noa+1){
     print FILE2 "$lines[$i]";
   }
  print FILE2 "}\n";
  print FILE2 "$basis1 \n";
  if ($spin == 0) {
    print FILE2 "HF \n{ccsd(t)-
f12,gem_beta=0.9 ,df_basis=vdz/mp2fit,df_basis_exch=vdz/jkfit,ri_basis=vdz/jkfit}\n";
  } else {
    print FILE2 "HF \n{uccsd(t)-
f12,gem_beta=0.9,df_basis=vdz/mp2fit,df_basis_exch=vdz/jkfit,ri_basis=vdz/jkfit}\n";
  }

  print FILE2 "SHOW,ENERG*,EMP2*,EF12* \n\n";
  print FILE2 "$basis2 \n";
  if ($spin == 0) {
    print FILE2 "HF \n{ccsd(t)-
f12,gem_beta=1.0,df_basis=vtz/mp2fit,df_basis_exch=vtz/jkfit,ri_basis=vtz/jkfit}\n";
  } else {
    print FILE2 "HF \n{uccsd(t)-
f12,gem_beta=1.0,df_basis=vtz/mp2fit,df_basis_exch=vtz/jkfit,ri_basis=vtz/jkfit}\n";
  }

  print FILE2 "SHOW,ENERG*,EMP2*,EF12* \n\n";

  print FILE2 "$basis3 \n";

  if ($spin == 0) {
    print FILE2 "HF; \nccsd(t)\nSHOW,ENERG*,EMP2* \nHF;\n{ccsd(t);
Core,c1,c2,c3,c4,c5,c6,c7,c8;}\nSHOW,ENERG*,EMP2*\n\n";
```

```
  } else {
    print FILE2 "HF; \nuccsd(t)\nSHOW,ENERG*,EMP2* \nHF;\n{uccsd(t);
Core,c1,c2,c3,c4,c5,c6,c7,c8;}\nSHOW,ENERG*,EMP2*\n\n";
  }

  print FILE2 "$basis4 \n";
  print FILE2 "HF; \nRMP2;\nSHOW,ENERG*,EMP2* \nHF;\n{RMP2;
Core,c1,c2,c3,c4,c5,c6,c7,c8;}\nSHOW,ENERG*,EMP2* \n\n";

  print FILE2 "basis, AVDZ-DK, h=VDZ-DK \ndkroll=1 \nHF \nRMP2; \nSHOW,ENERG*,EMP2*  \n\n";
  print FILE2 "basis, AVDZ, h=VDZ \ndkroll=0 \nHF \nRMP2;\nSHOW,ENERG*,EMP2*   \n\n";

  print FILE2 "basis, AVTZ-DK, h=VTZ-DK \ndkroll=1 \nHF \nRMP2; \nSHOW,ENERG*,EMP2*  \n\n";
  print FILE2 "basis, AVTZ, h=VTZ \ndkroll=0 \nHF \nRMP2;\nSHOW,ENERG*,EMP2*   \n\n";

  print FILE2 "---";
  close FILE2;

 }
```

rWMS.pl

```
#!/usr/bin/perl
#
# Usage:
#      perl rWMS.pl ch3.out
#
@E_num = (2, 2, 2, 2, 3, 3, 2);
@E_name = ('_ENERGR', '_EF12_RHFRELAX', 'EMP2            =', 'EF12            =', 'CCSD-
F12b correlation energy', 'Triples (T) contribution', 'ENERGC         =') ;

$workdir=`pwd`;
chomp $workdir;
# parameters define WMS
 $HF_a = 2.178;
 $CARBS_a = 2.309;
 $MP2c_a = 1.018;
 $F12_a =1.126;
 $deta_CCSD_MP2_F12_a = 1.569;
 $T_a = 2.175;
 $MP2CV_a = 3.55;
 $CCCV_a = 1.0;
 $TCV_a = 3.8;
 $Rel_a = 2.00;

# grab the raw energies
@HF = graben($E_name[0], $E_num[0]);
@CARBS = graben($E_name[1], $E_num[1]);
@MP2 = graben($E_name[2], $E_num[2]);
@F12 = graben($E_name[3], $E_num[3]);
@CCSD_F12b = graben($E_name[4], $E_num[4]);
@T = graben($E_name[5], $E_num[5]);
@CCSD_CV = graben($E_name[6], $E_num[6]);
```

```
$name = $ARGV[0];
@out=split(/\./, $name);

# core valence correlation
$MP2_val_DZ = $MP2[2] - $HF[2];
$MP2_core_DZ = $MP2[3] - $HF[3];
$dMP2_CV_DZ = $MP2_core_DZ - $MP2_val_DZ;

$MP2_val_TZ = $MP2[4] - $HF[4];
$MP2_core_TZ = $MP2[5] - $HF[5];
$dMP2_CV_TZ = $MP2_core_TZ - $MP2_val_TZ;

$CCSD_val_DZ = $CCSD_CV[0] - $MP2_val_DZ ;
$CCSD_core_DZ = $CCSD_CV[1] - $MP2_core_DZ ;
$dCCSD_CV_DZ = $CCSD_core_DZ - $CCSD_val_DZ ;


$T_val_DZ = $T[4];
$T_core_DZ = $T[5];
$dT_CV_DZ = $T_core_DZ - $T_val_DZ ;

$MP2CV_c = 3**$MP2CV_a/(3**$MP2CV_a - 2**$MP2CV_a);
$MP2CV_CBS = $dMP2_CV_DZ + $MP2CV_c*($dMP2_CV_TZ - $dMP2_CV_DZ);
$CV = $MP2CV_CBS + $CCCV_a*$dCCSD_CV_DZ + $TCV_a*$dT_CV_DZ;

# Relativistic
$HF_DK_DZ = $HF[6];
$HF_noRel_DZ= $HF[7];
$dHF_Rel_DZ = $HF_DK_DZ - $HF_noRel_DZ;

$MP2_DK_DZ = $MP2[6] - $HF[6];
$MP2_noRel_DZ= $MP2[7] - $HF[7];
$dMP2_Rel_DZ = $MP2_DK_DZ - $MP2_noRel_DZ;

$HF_DK_TZ = $HF[8];
$HF_noRel_TZ= $HF[9];
$dHF_Rel_TZ = $HF_DK_TZ - $HF_noRel_TZ;

$MP2_DK_TZ = $MP2[8] - $HF[8];
$MP2_noRel_TZ = $MP2[9] - $HF[9];
$dMP2_Rel_TZ = $MP2_DK_TZ - $MP2_noRel_TZ;

$Rel_DZ = $MP2[6] -$MP2[7] ;
$Rel_TZ = $MP2[8] -$MP2[9] ;
$Rel_c = 3**$Rel_a/(3**$Rel_a - 2**$Rel_a);
$Rel_CBS = $Rel_DZ + $Rel_c*($Rel_TZ - $Rel_DZ);

# valence correlation
$HF_DZ = $HF[0] - $CARBS[0];
$HF_TZ = $HF[1] - $CARBS[1];
$HF_CBS = $HF_DZ + $HF_a*($HF_TZ - $HF_DZ);


$CARBS_DZ =  $CARBS[0];
$CARBS_TZ =  $CARBS[1];
$CARBS_CBS = $CARBS_DZ + $CARBS_a*($CARBS_TZ - $CARBS_DZ);
```

```
$MP2c_DZ = $MP2[0] - $HF[0];
$MP2c_TZ = $MP2[1] - $HF[1];
$MP2c_CBS = $MP2c_DZ + $MP2c_a*($MP2c_TZ - $MP2c_DZ);

$F12_DZ = $F12[0];
$F12_TZ = $F12[1];
$F12_CBS = $F12_DZ + $F12_a*($F12_TZ - $F12_DZ);

$deta_CCSD_MP2_F12_DZ = $CCSD_F12b[0] - $F12_DZ - $MP2c_DZ ;
$deta_CCSD_MP2_F12_TZ = $CCSD_F12b[1] - $F12_TZ - $MP2c_TZ ;
$deta_CCSD_MP2_F12_CBS = $deta_CCSD_MP2_F12_DZ +
$deta_CCSD_MP2_F12_a*($deta_CCSD_MP2_F12_
TZ - $deta_CCSD_MP2_F12_DZ);

$T_DZ = $T[0];
$T_TZ = $T[2];
$T_CBS = $T_DZ + $T_a*($T_TZ - $T_DZ);

$HF_Sum_CBS = $HF_CBS + $CARBS_CBS;
$MP2_CBS = $MP2c_CBS + $F12_CBS ;
$CCSD_CBS = $MP2c_CBS + $F12_CBS + $deta_CCSD_MP2_F12_CBS;


$EWM = $HF_Sum_CBS + $CCSD_CBS + $T_CBS + $HOVC + $CV + $Rel_CBS;
print "$HF_Sum_CBS       $CCSD_CBS   $T_CBS  $CV   $Rel_CBS  $HOVC  $EWM\n";
print "$EWM \n";


sub graben {
  my (@en, $i, $j, $k, $ename, $enum, @temp, $tl);
  $ename = $_[0];
  $enum = $_[1];
  system("grep \"$ename\" $ARGV[0] > $workdir/tempen.out");
  open(FILE1, "<$workdir/tempen.out") || die "can't open $!";
  @lines = (<FILE1>);
  close FILE1;
  for $j (0 .. @lines-1) {
    $tl = $lines[$j];
    @temp=($tl =~ /\S+/g);
    $en[$j] = $temp[$enum];
  }
  system("rm -rf  $workdir/tempen.out");
return @en;
}
```