

Density Functional Theory

General information

Density functional theory in Firefly is implemented for restricted, unrestricted, and restricted-open wavefunctions. For each wavefunction, energies and analytical gradients are available. Analytical Hessians are currently not available. Instead, Hessians can be calculated in a semi-numerical fashion. Double hybrid functionals are an exception – for these functionals, only energies are currently available. Various functionals can be used, a list of these is provided in the next section. It should be noted that the DFT code in Firefly is completely different from that of GAMESS (US) with respect to both implementation and input specification.

A DFT calculation can be requested by specifying a functional using the **DFTTYP** keyword in **\$CONTRL**. The value of SCFTYP determines which type of wavefunction is used. For example, a RO-BLYP calculation can be requested using

```
SCFTYP=ROHF DFTTYP=BLYP
```

Additional DFT-related input is provided with the optional **\$DFT** group. The majority of the keywords in this group can be used to change the accuracy of various aspects of the DFT calculation. The default values are fine for most cases, so providing this input usually isn't necessary.

Information on the convergence accelerators and conventional vs direct runs can be found in the chapter on Hartree-Fock.

A few things should be mentioned about performance. First of all, the current DFT implementation in Firefly does not support multithreading. It is therefore recommended to run calculations in parallel mode. Secondly, the molecular symmetry specified in **\$DATA** is used only partially during calculation of the DFT contributions to the Fock matrix. To be more precise, the so-called octant symmetry is not used at present.

Resolution of identity / Coulomb fitting techniques, which serve to speed up calculations that use pure DFT functionals, are currently not implemented. As a result, one might expect that other DFT programs that exploit these techniques (for example, TURBOMOLE) will outperform Firefly on pure DFT calculations. This is especially true with large molecules. For such molecules, the time for the pure DFT part of calculations depends approximately linearly on the number of atoms in molecule. The reason for this is that, while for small molecular systems the pure DFT part of the calculations is usually the time-limiting step, for large systems the cost of the standard Coulomb (and possible exchange for hybrid functionals) contributions to the Fock matrix due to two-electron integrals becomes the dominant part of the calculations. The implementation of Coulomb fitting techniques should overcome this shortcoming.

It should also be noted that the RODFT energy/gradient code is routed through the generic UDFT code, therefore, the RODFT performance is identical to that of UDFT. This situation might be changed in the future by adding separate RODFT routines which should provide a speedup of approximately 15 to 20 % for small and medium size molecules.

Double hybrid functionals, which use an MP2-like perturbation term in the correlation part of the functional, can make use of any of the MP2 methods in Firefly. Just as with MP2 calculations, **\$MP2**

METHOD=1 is the preferred method except for small jobs. Please see the chapter on MP2 for more information.

An example input file for a BLYP geometry optimization and subsequent vibrational analysis for a water molecule:

```
$CONTRL SCFTYP=RHF DFTTYP=BLYP RUNTYP=OPTIMIZE $END
$SYSTEM TIMLIM=3000 MEMORY=3000000 $END
$BASIS GBASIS=N31 NGAUSS=6 NDFUNC=1 $END
$STATPT HSEND=.T. NPRT=-2 $END
$FORCE NVIB=2 VIBSIZ=0.005 $END
$DATA
H2O
CNV 2

O          8.0    0.0000000000    0.0000000000    0.7205815395
H          1.0    0.0000000000    0.7565140024    0.1397092302
$END
```

Finally, some technical details should be mentioned. The DFT implementation in Firefly is grid-based. Angular integration is based on Lebedev quadratures, while radial integration is based on Mura-Knowles quadratures. The atomic partitioning function used is the modification of Stratmann, Scuseria and Frisch (Chem. Phys. Lett. (1996) 257, 213).

Available functionals

Before providing the full list of functionals available in Firefly, it is important to mention that the specification of a functional in the input is unambiguous except for a few specific cases that will be discussed below.

Firstly, there are two variations of the VWN functional available as components of the B3LYP and O3LYP functionals, namely VWN formula 1 RPA and VWN formula 5. The reason for the inclusion of both variants is to improve compatibility with other QC programs, which can use either of the two formulae. Formula 1 RPA type B3LYP is used by for example NWChem and Gaussian, while formula 5 type B3LYP is the default with GAMESS (US). The use of VWN formula 1 RPA in B3LYP and O3LYP can be specified with **DFTTYP=B3LYP1** and **DFTTYP=O3LYP1**, respectively. Similarly, the use of VWN formula 5 can be specified with **DFTTYP=B3LYP5** and **DFTTYP=O3LYP5**. Note that some QC programs refer to VWN formula 1 RPA as “VWN functional III” and that these are the same functionals.

For B3LYP, it is also possible to specify only **DFTTYP=B3LYP**. The choice of VWN formula will in this case depend on the value of **B3LYP** in the **\$DFT** group. Possible values are **NWCHEM** (formula 1 RPA, the default) and **GAMESS** (formula 5). **DFTTYP=O3LYP** will always be interpreted as **O3LYP5**.

For the O3LYP functionals, the weight of the non-local exchange can be adjusted with the **O3LYP** keyword in **\$DFT**. The reason for including this option is that recent versions of Gaussian use a weight different from the one used by most other QC programs. Setting **O3LYP** to **GAUSSIAN** changes the weight so that the resulting functional will be identical to one used in Gaussian 03 Rev D.01 and above. Specifying **O3LYP=DEFAULT** (which is the default value) enables compatibility with the implementation used in other QC programs. Note that the reason for the existence of two different implementations is that there are some ambiguities in O3LYP-related papers. Due to these ambiguities, it is not possible to say which implementation is “correct”. Instead, these two implementations should be considered as two different functionals.

There is also a keyword pertaining to the Perdew-Zunger 1981 LDA correlation that should be mentioned. In the paper defining this functional, its parameters were given with only four digits. As there are two branches in the fit, two parameters of the fit were selected in such a manner that E_{xc} and V_{xc} are globally continuous functions, even at the branch point itself. However, the four-digit precision is not enough for this purpose as it causes small deviations from exact continuity that in turn result in errors such as non-precise gradients. Tight geometry optimizations in particular are not possible with this functional. By default, the parameters of the fit are used with increased precision (for P81LDA as well as for all functionals using P81 local correlation), hereby achieving smooth E_{xc} and V_{xc} . However, this corrected fit does result in a slightly different functional. If desired, the functional can be reverted to its slightly discontinuous form by specifying **FIXP81=.F.** in the **\$DFT** group.

With respect to double hybrid functionals: most of these functionals by default do not freeze any cores. This behavior is forced by Firefly, but can be overridden by specifying a negative value for **NCORE**. This keyword belongs to the **\$MP2** group (not the **\$DFT** group), and input should look like this:

```
$MP2 NCORE=-5 $END
```

Here, the minus sign tells Firefly to override the default behavior and the number is the amount of cores to be frozen.

Note that there are two functionals which by default *do* use frozen cores, namely D-LYP and D-PBEP86 (as these functionals were designed to be used together with the frozen core approximation). This behavior is not forced though so, if desired, it can be changed using the **NCORE** keyword without having to use the minus sign.

The list of available functionals is as follows:

Exchange functionals (no correlation)

DFTTYP value	Functional	Refs.
<i>LDA exchange</i>		
SLATER	Slater exchange	1
LSDA	A synonym of SLATER	
<i>GGA exchange</i>		
B88	Becke 1988 exchange	2
XPW91	Perdew-Wang 1991 exchange	3
XMPW91	Modified Perdew-Wang 1991 exchange. Modification by Adamo and Barone	11
GILL96	Gill 1996 exchange	4
XPBE96	Perdew-Burke-Ernzerhof 1996 exchange	5
OPTX	Handy-Cohen 2001 OPTX exchange	6
XSOGGA	The exchange part of the SOGGA functional by Zhao and Truhlar	20

XSOGGA11	The exchange part of the SOGGA-11 functional by Peverati, Zhao, and Truhlar	21
<i>Hybrid exchange</i>		
HFS	Slater (0.5) + Hartree-Fock (0.5) exchange	1
XSOGGAX	The exchange part of the SOGGA-11X hybrid functional by Peverati and Truhlar	31

Pure correlation functionals – these use 100 % exact (i.e. Hartree-Fock) exchange

DFTTYP value	Functional	Refs.
<i>LDA correlation</i>		
VWN1RPA	VWN formula 1 RPA local correlation. This functional is referred to as “VWN formula 3” by some programs (such as Gaussian, Q-Chem).	7
VWN5	VWN formula 5 local correlation	8
P81LDA	Perdew-Zunger 1981 local correlation	36
PW91LDA	Perdew-Wang 1991 local correlation	9
<i>GGA correlation</i>		
P86	Perdew 1986 nonlocal (1.0) + Perdew-Zunger 1981 local (1.0) correlation	37,36
LYP	Lee-Yang-Parr 1988 correlation	10
CPBE96	Perdew-Burke-Ernzerhof 1996 nonlocal (1.0) + Perdew-Wang 1991 local (1.0) correlation	5,9
CPW91	Perdew 1991 nonlocal (1.0) + Perdew-Wang 1991 local (1.0) correlation	12,9
CSOGGA11	The correlation part of the SOGGA-11 functional by Peverati, Zhao, and Truhlar	21
CSOGGAX	The correlation part of the SOGGA-11X functional by Peverati and Truhlar	31

Exchange-correlation functionals

DFTTYP value	Functional	Refs.
SVWN1RPA	Slater exchange, VWN formula 1 RPA local correlation	1,7
BVWN1RPA ¹	Becke 1988 exchange, VWN formula 1 RPA local correlation	2,7
SVWN5	Slater exchange, VWN formula 5 local correlation	1,8
BVWN5 ¹	Becke 1988 exchange, VWN formula 5 local correlation	2,8
SP81LDA	Slater exchange, Perdew-Zunger 1981 local correlation	1,36
BP81LDA	Becke 1988 exchange, Perdew-Zunger 1981 local correlation	2,36
BP86	Becke 1988 exchange, Perdew 1986 nonlocal (1.0) + Perdew-Zunger 1981 local (1.0) correlation	2,37,36
PBEP86	Perdew-Burke-Ernzerhof 1996 exchange, Perdew 1986 nonlocal (1.0) + Perdew-Zunger 1981 local (1.0) correlation	5,37,36

SLYP	Slater exchange, Lee-Yang-Parr 1988 correlation	1,10
BLYP	Becke 1988 exchange, Lee-Yang-Parr 1988 correlation	2,10
GLYP	Gill 1996 exchange, Lee-Yang-Parr 1988 correlation	4,10
MPW91LYP	Modified Perdew-Wang 91 exchange, Lee-Yang-Parr 1988 correlation	11,10
OLYP	OPTX exchange, Lee-Yang-Parr 1988 correlation	6,10
XLYP	Extended exchange functional (a combination of Slater local (1.0), Becke 88 nonlocal (0.722), and Perdew-Wang 91 nonlocal (0.347) exchange), Lee-Yang-Parr 1988 correlation	13
SPW91LDA	Slater exchange, Perdew-Wang 1991 local correlation	1,9
BPW91LDA ¹	Becke 1988 exchange, Perdew-Wang 1991 local correlation	2,9
BPW91	Becke 1988 exchange, Perdew 1991 nonlocal (1.0) + Perdew-Wang 1991 local (1.0) correlation	2,12,9
PW91	Perdew-Wang 1991 exchange, Perdew 1991 nonlocal (1.0) + Perdew-Wang 1991 local (1.0) correlation	3,12,9
MPW91	Modified Perdew-Wang 91 exchange, Perdew 1991 nonlocal (1.0) + Perdew-Wang 1991 local (1.0) correlation	11,12,9
PBEPW91L ¹	Perdew-Burke-Ernzerhof 1996 exchange + Perdew-Wang 1991 local correlation	5,9
PBEPW91	Perdew-Burke-Ernzerhof 1996 exchange, Perdew 1991 nonlocal (1.0) + Perdew-Wang 1991 local (1.0) correlation	5,12,9
PBE96	Perdew-Burke-Ernzerhof 1996 exchange, Perdew-Burke-Ernzerhof 1996 nonlocal (1.0) + Perdew-Wang 1991 local (1.0) correlation	5,9
REVPBE	Revised Perdew-Burke-Ernzerhof 1996 exchange, Perdew-Burke-Ernzerhof 1996 nonlocal (1.0) + Perdew-Wang 1991 local (1.0) correlation	43,5,9
MPW91PBE	Modified Perdew-Wang 91 exchange, Perdew-Burke-Ernzerhof 1996 nonlocal (1.0) + Perdew-Wang 1991 local (1.0) correlation	11,5,9
HCTH93	HCTH/93 exchange-correlation functional	14
HCTH120	HCTH/120 exchange-correlation functional	15
HCTH147	HCTH/147 exchange-correlation functional	15
HCTH407	HCTH/407 exchange-correlation functional	16
HCTH407P	HCTH/407+ exchange-correlation functional	17
HCTHP14	HCTH potential-fitted 1/4 functional	18
B97GGA1	Becke 97, reparametrized as pure GGA by Cohen and Handy (2000)	19
B97D	Becke 97-D, another pure GGA reparameterization of B97, by Grimme (2006). To be used with DFT-D. Selection of this functional automatically enables the use of the DFT-D2 scheme (unless the user chooses a different scheme or explicitly disables the use of DFT-D)	30
SOGGA	SOGGA exchange-correlation functional by Zhao and Truhlar	20
SOGGA11	SOGGA-11 exchange-correlation functional by Peverati, Zhao, and Truhlar	21

¹ For these functionals the HFX parameter (see below) can be used to include exact exchange.

Hybrid functionals

DFTYP value	Functional	Refs.
B1P86	Becke 1988 (0.75) + Hartree-Fock (0.25) exchange, Perdew 1986 nonlocal (1.0) + Perdew-Zunger 1981 local (1.0) correlation	2,37,36
B1LYP	Becke 1988 (0.75) + Hartree-Fock (0.25) exchange, Lee-Yang-Parr 1988 correlation	2,10
BHLYP	Becke 1988 (0.5) + Hartree-Fock (0.5) exchange, Lee-Yang-Parr 1988 correlation	24,10
B1PW91	Becke 1988 (0.75) + Hartree-Fock (0.25) exchange, Perdew 1991 nonlocal (1.0) + Perdew-Wang 1991 local (1.0) correlation	2,12,9
B3P86	Becke 1988 nonlocal (0.72) + Slater local (0.80) + Hartree-Fock (0.20) exchange, Perdew 1986 nonlocal (1.0) + Perdew-Zunger 1981 LDA local (1.0) correlation	22,37,36
B3LYP1	Becke 1988 nonlocal (0.72) + Slater local (0.80) + Hartree-Fock (0.20) exchange, Lee-Yang-Parr 1988 (0.81) + VWN formula 1 RPA local (0.19) correlation. This is B3LYP as implemented in NWCHEM and GAUSSIAN	22,10
B3LYP5	Same as B3LYP1, but with VWN formula 5 instead of VWN formula 1 RPA local correlation. This is B3LYP as implemented in GAMESS (US)	
B3LYP	Either B3LYP1 or B3LYP5, depending on the value of the B3LYP keyword in the \$DFT group (default is B3LYP1)	
B3PW91	Becke 1988 nonlocal (0.72) + Slater local (0.80) + Hartree-Fock (0.20) exchange, Perdew 1991 nonlocal (0.81) + Perdew-Wang 1991 LDA local (1.0) correlation	22,12,9
O3LYP1	OPTX nonlocal (0.8133) + Slater local (0.9262) + Hartree-Fock (0.1161) exchange, Lee-Yang-Parr 1988 (0.81) + VWN formula 1 RPA local (0.19) correlation	23
O3LYP5	Same as O3LYP1, but with VWN formula 5 instead of VWN formula 1 RPA local correlation	
O3LYP	A synonym of O3LYP5	
X3LYP	Extended hybrid exchange functional (a combination of Slater local (0.782), Becke 88 nonlocal (0.542), and Perdew-Wang 91 nonlocal (0.167) exchange + Hartree-Fock (0.218)), Lee-Yang-Parr (0.871) + VWN formula 1 RPA local (0.129) correlation	13
PBE1P86	Perdew-Burke-Ernzerhof 1996 (0.75) + Hartree-Fock (0.25) exchange, Perdew 1986 nonlocal (1.0) + Perdew-Zunger 1981 local (1.0) correlation	5,37,36
PBE1PW91	Perdew-Burke-Ernzerhof 1996 (0.75) + Hartree-Fock (0.25) exchange, Perdew 1991 nonlocal (1.0) + Perdew-Wang 1991 LDA local (1.0) correlation	25,12,9
PBE0	Perdew-Burke-Ernzerhof 1996 (0.75) + Hartree-Fock (0.25) exchange, Perdew-Burke-Ernzerhof 1996 nonlocal (1.0) + Perdew-Wang 1991	25

	LDA local (1.0) correlation. This functional is also known as PBE1PBE	
REVPBE0	Revised Perdew-Burke-Ernzerhof 1996 (0.75) + Hartree-Fock (0.25) exchange, Perdew-Burke-Ernzerhof 1996 nonlocal (1.0) + Perdew-Wang 1991 LDA local (1.0) correlation	43,25
MPW1LYP	Modified Perdew-Wang 91 (0.75) + Hartree-Fock (0.25) exchange, Lee-Yang-Parr 1988 correlation	11,10
MPW1PW91	Modified Perdew-Wang 91 (0.75) + Hartree-Fock (0.25) exchange, Perdew 1991 nonlocal (1.0) + Perdew-Wang 1991 LDA local (1.0) correlation	11
MPW1K	Modified Perdew-Wang 91 (0.572) + Hartree-Fock (0.428) exchange, Perdew 1991 nonlocal (1.0) + Perdew-Wang 1991 LDA local (1.0) correlation. Functional by Truhlar and coworkers	41
MPW1PBE	Modified Perdew-Wang 91 (0.75) + Hartree-Fock (0.25) exchange, Perdew-Burke-Ernzerhof 1996 nonlocal (1.0) + Perdew-Wang 1991 local (1.0) correlation	11,5,9
MPW3PBE	Modified Perdew-Wang 91 local and nonlocal (0.72) + Slater local (0.08) + Hartree-Fock (0.20) exchange Perdew-Burke-Ernzerhof 1996 nonlocal (0.81) + Perdew-Wang 1991 local (1.0) correlation	11,1,5,9
B970	Becke 97 hybrid exchange-correlation functional	26
B980	Becke 98 hybrid exchange-correlation functional	27
B971	Becke 97-1, a 1998 reparametrized version of B97 by Handy, Tozer, and coworkers	14
B972	Becke 97-2, a 2001 reparametrized version of B97 by Wilson, Bradley, and Tozer	28
B973	Becke 97-3, a 2005 reparametrized version of B97 by Keal and Tozer	29
SOGGAX	SOGGA-11X hybrid exchange-correlation functional by Peverati and Truhlar	31

Double hybrid functionals

DFTTYP value	Functional	Refs.
<i>Self-consistent functionals</i>		
B2PLYP	Becke 1988 (0.47) + Hartree-Fock (0.53) exchange, Lee-Yang-Parr 1988 (0.73) + MP2 (0.27) correlation	32
B2GPPLYP	Becke 1988 (0.35) + Hartree-Fock (0.65) exchange, Lee-Yang-Parr 1988 (0.64) + MP2 (0.36) correlation	33
MPW2PLYP	Modified Perdew-Wang 91 (0.45) + Hartree-Fock (0.55) exchange, Lee-Yang-Parr 1988 (0.75) + MP2 (0.25) correlation	42
DSD-BLYP	Initial version of DSD-BLYP(full), which uses the following terms and parameters: Becke 1988 (0.31) + Hartree-Fock (0.69) exchange, Lee-Yang-Parr 1988 (0.54) + opposite spin MP2 (0.46) + same spin MP2 (0.37) correlation + DFT-D version 2 with an alpha value of 60.0. Selection of this functional automatically sets DFTD= . T .	38

D-BLYP	Newer version DSD-BLYP(fc), which uses the following terms and parameters: Becke 1988 (0.29) + Hartree-Fock (0.71) exchange, Lee-Yang-Parr 1988 (0.55) + opposite spin MP2 (0.46) + same spin MP2 (0.43) correlation + DFT-D version 2 with an alpha value of 20.0. Selection of this functional automatically sets DFTD=. T .	39
D-PBEP86	DSD-PBEP86(fc)-D2 functional: Perdew-Burke-Ernzerhof 1996 (0.32) + Hartree-Fock (0.68) exchange, Perdew 1986 nonlocal (0.45) + Perdew-Zunger 1981 LDA local (0.45) + opposite spin MP2 (0.51) + same spin MP2 (0.23) correlation + DFT-D version 2. Selection of this functional automatically sets DFTD=.T. Functional and empirical dispersion parameters recommended for the use of D-PBEP86 with DFT-D version 3BJ are not implemented but can be given manually.	39
DSD-PBE	DSD-PBEPBE(fc)-D3BJ functional: Perdew-Burke-Ernzerhof 1996 (0.32) + Hartree-Fock (0.68) exchange, Perdew-Burke-Ernzerhof 1996 nonlocal (0.49) + Perdew-Wang 1991 local (0.49) correlation + opposite spin MP2 (0.55) + same spin MP2 (0.13) correlation + DFT-D version 3BJ	44
PBE0-2	Perdew-Burke-Ernzerhof 1996 (~0.206) + Hartree-Fock (~0.794) exchange, Perdew-Burke-Ernzerhof 1996 nonlocal (0.5) + Perdew-Wang 1991 local (0.5) correlation + MP2 (0.5) correlation	45
<i>Non-self-consistent functionals</i>		
XYG3	Becke 1988 nonlocal (0.2107) + Slater local (0.1967) + Hartree-Fock (0.8033) exchange, Lee-Yang-Parr 1988 (0.6789) + MP2 (0.3211) correlation. Each term in this functional (incl. the MP2-like term) is evaluated using B3LYP1 orbitals and density.	34
XYGJ-OS	Slater local (0.2269) + Hartree-Fock (0.7731) exchange, Lee-Yang-Parr 1988 (0.2754) + VWN formula 1 RPA local (0.2309) + MP2 (0.4364) correlation. Each term in this functional (incl. the MP2-like term) is evaluated using B3LYP1 orbitals and density. From the MP2-like term, only the opposite spin part is used.	40
XDH-PBE0	Perdew-Burke-Ernzerhof 1996 (0.1665) + Hartree-Fock (0.8335) exchange, Perdew-Burke-Ernzerhof 1996 nonlocal (0.5292) + Perdew-Wang 1991 LDA local (0.5292) + MP2 (0.5428) correlation. Each term in this functional (incl. the MP2-like term) is evaluated using PBE0 orbitals and density. From the MP2-like term, only the opposite spin part is used.	35

References

1. J.C. Slater; Phys. Rev., 81, 385-390 (1951)
2. A.D. Becke; Phys. Rev. A, 38, 3098-3100 (1988)
3. J.P. Perdew, J.A. Chevary, S.H. Vosko, K.A. Jackson, M.R. Pederson, D.J. Singh, C.Fiolhais; Phys. Rev. B, 46, 6671-6687 (1992)
4. P.M.W. Gill; Mol. Phys., 89, 433-445 (1996)
5. J.P. Perdew, K. Burke, M. Ernzerhof, Phys. Rev. Lett., 77, 3865-3868 (1996)
6. N.C. Handy, A.J. Cohen; Mol. Phys., 99, 403-412 (2001)
7. Functional III in S.H. Vosko, L. Wilk, M. Nusair; Can. J. Phys., 58, 1200-1211 (1980)

8. Functional V in S.H. Vosko, L. Wilk, M. Nusair; *Can. J. Phys.*, 58, 1200-1211 (1980)
9. J.P. Perdew, Y. Wang; *Phys. Rev. B*, 45, 13244-13249 (1992)
10. C. Lee, W. Yang, R.G. Parr; *Phys. Rev. B*, 37, 785-789 (1988)
11. C. Adamo, V. Barone, *J. Chem. Phys.*, 108, 664-675 (1998)
12. J.P. Perdew, J.A. Chevary, S.H. Vosko, K.A. Jackson, M.R. Pederson, D.J. Singh, C.Fiolhais; *Phys. Rev. B*, 46, 6671-6687 (1992)
13. X. Xu, W. A. Goddard III, *P. Natl. Acad. Sci. USA*, 101, 2673-2677 (2004)
14. F.A. Hamprecht, A.J. Cohen, D.J. Tozer, N.C. Handy *J. Chem. Phys.*, 109, 6264-6271 (1998)
15. A.D. Boese, N.L. Doltsinis, N.C. Handy, M. Sprik. *J. Chem. Phys.*, 112, 1670-1678 (2000)
16. A.D. Boese, N.C. Handy, *J. Chem. Phys.*, 114, 5497-5503 (2001)
17. A.D. Boese, A. Chandra, J.M.L. Martin, D. Marx, *J. Chem. Phys.*, 119, 5965-5980 (2003)
18. G. Menconi, P.J. Wilson, D.J. Tozer, *J. Chem. Phys.*, 114, 3958-3967 (2001)
19. A.J. Cohen and N.C. Handy, *Chem. Phys. Lett.*, 316, 160-166 (2000)
20. Y. Zhao, D.G. Truhlar, *J. Chem. Phys.*, 128, 184109 (2008)
21. R. Peverati, Y. Zhao, D.G. Truhlar, *J. Phys. Chem. Lett.*, 2, 1991-1997 (2011)
22. A.D. Becke, *J. Chem. Phys.*, 98, 5648-5652 (1993)
23. W.M. Hoe, A.J. Cohen, N.C. Handy, *Chem. Phys. Lett.*, 341, 319-328 (2001)
24. A.D. Becke *J. Chem. Phys.*, 98, 1372-1377 (1993)
25. C. Adamo, V. Barone, *J. Chem. Phys.*, 110, 6158-6169 (1999)
26. A.D. Becke, *J. Chem. Phys.*, 107, 8554-8560 (1997)
27. H.L. Schmider, A.D. Becke, *J. Chem. Phys.*, 108, 9624-9631 (1998)
28. P.J. Wilson, T.J. Bradley, D.J. Tozer, *J. Chem. Phys.*, 115, 9233-9242 (2001)
29. T.W. Keal, D.J. Tozer, *J. Chem. Phys.*, 123, 121103 (2005)
30. S. Grimme, *J. Comp. Chem.*, 27, 1787-1799 (2006)
31. R. Peverati, D.G. Truhlar, *J. Chem. Phys.*, 135, 191102 (2011)
32. S. Grimme, *J. Chem. Phys.*, 124, 034108 (2006)
33. A. Karton, A. Tarnopolsky, J.-F. Lamère, G.C. Schatz, J.M.L. Martin, *J. Phys. Chem. A*, 112, 12868-12886 (2008)
34. Y. Zhang, X. Xu, W. A. Goddard III, *P. Natl. Acad. Sci. USA*, 106, 4963-4968 (2009)
35. I.Y. Zhang, N.Q. Su, E.A.G. Brémond, C. Adamo, X. Xu, *J. Chem. Phys.* 136, 174103 (2012)
36. J.P. Perdew, A. Zunger, *Phys. Rev. B*, 23, 5048-5079 (1981)
37. J.P. Perdew, *Phys. Rev. B*, 33, 8822-8824 (1986)
38. S. Kozuch, D. Gruzman, J.M.L. Martin, *J. Phys. Chem. C*, 114, 20801-20808 (2010)
39. S. Kozuch, J.M.L. Martin, *Phys. Chem. Chem. Phys.*, 13, 20104-20107 (2011)
40. I.Y. Zhang, X. Xu, Y. Jung, W.A. Goddard III, *Proc. Natl. Acad. Sci. U.S.A.*, 108, 19896-19900 (2011)
41. B.J. Lynch, P.L. Fast, M. Harris and D.G. Truhlar, *J. Phys. Chem. A*, 104, 4811-4815 (2000)
42. T. Schwabe, S. Grimme, *Phys. Chem. Chem. Phys.*, 8, 4398-4401 (2006)
43. Y. Zhang, W. Yang, *Phys. Rev. Lett.* 80, 890 (1998)
44. S. Kozuch, J. M. L. Martin, *J. Comp. Chem.*, 34, 2327-2344 (2013)
45. J.-D. Chai, S.-P. Mao, *Chem. Phys. Lett.*, 538, 121-25 (2012)

Making modifications to functionals

Firefly has four important keywords, all belonging to the \$DFT group, which can be used to modify some of the (double) hybrid functionals implemented. These are:

- **HFX**, which provides control over the amount of exact exchange;
- **CPT2**, which provides control over the amount of MP2-like correlation;
- **SCS**, which controls scaling of spin components of the MP2-like perturbation;

- **PARENT**, which selects the parent functional used in non-self-consistent double hybrid functionals (such as XYG3, for which the parent functional is B3LYP1).

Changing the amount of exact exchange used in a functional can, for example, be useful in TDDFT calculations to distinguish between Rydberg and valence states. The amount of exact exchange can be controlled through the **HFX** option. Formally, **HFX** should be given as an array (for reasons that will be discussed later), *e.g.*:

```
HFX(1)=0.75
```

However, it is allowed to specify **HFX** as a single variable:

```
HFX=0.75
```

The **HFX** option is available for most hybrid functional and all double hybrid functionals. In addition, it can be used for certain pure functionals in order to include a portion of exact exchange (these are BVWN1RPA, BVWN5, BPW91LDA, and PBEPW91L). However, for the B97/B98 type functionals as well as for the SOGGA-X functional this option has not been implemented. This is because these functionals were parameterized for a certain amount of exact exchange and changing this amount does not make much sense. Note that there is no limit to **HFX** – it is possible to set its value larger than 1.0 for experimentation purposes.

For (double) hybrid functionals that combine exact exchange with only one type of DFT exchange, such as BHHLYP, PBE0, and MPW1PW91, the way the **HFX** keyword works is quite straightforward. Setting **HFX(1)=0** for example removes all exact exchange from the functional, while **HFX(1)=1** makes the exchange part of the functional fully exact. To illustrate, setting **HFX(1)=0** for PBE1PW91 makes the functional identical to PBEPW91, while **HFX(1)=1** makes the functional identical to PW91. Naturally, any value between 0 and 1 is possible. **HFX(1)=0.25** corresponds to the default amount of exact exchange for the PBE1PW91.

For the B3LYP, O3LYP, MPW3PBE, and XYG3 functionals, the interpolation with **HFX** is a bit more complex and is best explained by an example. The exchange part of B3LYP consists of the following parameters (default values in parentheses):

```
B88X – the amount of Becke 88 local and nonlocal (0.72)
LDAX – the amount of excess Slater local (0.08)
      (because Becke 88 non-local + Slater local = Becke 88 GGA)
HFX – the amount of exact exchange (0.2)
```

When the value of **HFX** is changed, B88X and LDAX change as follows:

```
B88X_new = B88X * (1 - HFX) / (B88X + LDAX)
LDAX_new = LDAX * (1 - HFX) / (B88X + LDAX)
```

For X3LYP, the interpolation scheme is different as this functional combines 4 types of exchange. Its parameters in the exchange part are as follows:

```
B88X – the amount of Becke 88 local and nonlocal (0.542)
PW91X – the amount of Perdew-Wang 91 local and nonlocal (0.167)
LDAX – the amount of excess Slater local (0.073)
HFX – the amount of exact exchange (0.218)
```

When the value of **HFX** is changed, the other parameters change as follows:

```
B88X_new = B88X * (1 - HFX) / (1 - HFX_default)
PW91X_new = PW91X * (1 - HFX) / (1 - HFX_default)
LDAX_new = LDAX * (1 - HFX) / (1 - HFX_default)
```

where **HFX_default** is the default value of 0.218.

For non-self-consistent double hybrid functionals, **HFX** can be used to control the amount of exact exchange for the double hybrid functional as well as its parent function. This can be done by giving an array of two variables. As an example, for xDH-PBE0, the default values can be specified as:

```
HFX(1)=0.8335,0.25
```

where the first value specifies the amount of exact exchange in the double hybrid functional and the second value specified the amount of exact exchange in the parent functional.

The **CPT2** keyword works in the same way as the **HFX** keyword. As noted earlier, it can be used to control the amount of MP2-like correlation in double hybrid functionals.

Finally, the **SCS** keyword can be used to scale the spin components of the MP2-like part by certain factors. It functions the same way as the **SCS** keyword in the **\$MP2** group. For more information, please see the MP2 section.

Of course, the **HFX**, **CPT2**, and **SCS** can be used to define a few functionals currently not present in Firefly, provided that these are of the same form as a functional already implemented. For example, the B2K-PLYP functional can be specified as follows:

```
$CONTRL DFTTYP=B2PLYP $END
$DFT HFX(1)=0.72 CPT2=0.42 $END
```

Here, **DFTTYP** is set to **B2PLYP** as this functional is of the same form as B2K-PLYP – only the fractions of exact exchange and MP2-like correlation are different. It would also have been possible to specify **DFTTYP=B2GPPLYP** as this functional is of the same form.

The DSD-PBEP86 functional implemented in Firefly was designed to be used with DFT-D version 2. However, reference 39 also contains parameters optimized for use with version 3 with BJ dampening. As a second example, this functional can be specified as follows:

```
$CONTRL DFTTYP=D-PBEP86 DFTD=.T. $END
$DFT HFX(1)=0.70 CPT2=0.57
SCS(1)=0.9298245614035088,0.4385964912280702 $END
$DFTD VERSN=4 S6=0.418 RS6=0.0 S18=0.25 RS18=5.65 ALP=14.0 $END
```

More information on DFT-D can be found in the next section.

Lastly, the **PARENT** keyword should be discussed. This keyword is only used for non-self-consistent double hybrid functionals such as XYG3 and selects the parent functional. This parent functional is used for generating the orbitals and density used for the evaluation of each term in the double hybrid functional. Any GGA exchange, GGA correlation, exchange-correlation, and hybrid functional may be specified. LDA functionals and double hybrid functionals cannot be used.

Empirical dispersion correction (DFT-D)

Newly available in Firefly 8.0.0 is the empirical dispersion correction to DFT proposed by Grimme and co-workers, a method commonly abbreviated as DFT-D. Three versions of the correction are available:

- DFT-D version 2, which was proposed in 2006 as part of the B97-D functional and which is an update of the original DFT-D version.
- DFT-D version 3, proposed in 2010.
- DFT-D version 3 with Becke-Johnson damping (from here on referred to as 'version 4'), proposed in 2011.

The corresponding references are:

- Version 2: S. Grimme, J. Comput. Chem., 27 (2006), 1787-1799
- Version 3: S. Grimme, J. Antony, S. Ehrlich, and H. Krieg, J. Chem. Phys. 132 (2010), 154104
- The BJ-damping used in version 4: S. Grimme, S. Ehrlich, and L. Goerigk, J. Comput. Chem. 32 (2011), 1456-1465

It is important to point out that the available range of versions (together with their default parameters) is defined by the DFT-D extension file in use (either `dftd.dll` or `dftd.ex`). Consequently, the empirical dispersion correction functionality is only available if Firefly is capable to find and load this extension file.

The empirical dispersion correction is designed to improve the long-range behavior of DFT methods. However, its use is not limited to DFT only – it can actually be used with any computational method present in Firefly. Both energies and analytical gradients are available with DFT-D, provided of course that analytical gradients are available for the method the correction is used with. Analytical second derivatives are not available, instead, second derivatives should be obtained numerically.

The use of the correction can be enabled by specifying `DFTD=.T.` in the `$CONTRL` group. Control over the correction is provided by keywords in the `$DFTD` group. Here, the `VERSN` keyword can be used to specify the DFT-D version to be used (default is `VERSN=3`). Thus, if one would like to use DFT-D version 4, one should specify:

```
$CONTRL DFTD=.T. $END
$DFTD  VERSN=4 $END
```

Parameters for many functionals (as well as for Hartree-Fock) are stored internally in Firefly. Their availability is as follows:

Version 2	Version 3	Version 3 with TZ=.T. (*)	Version 4
BP86	Hartree-Fock	BP86	Hartree-Fock
BLYP	Slater (LSDA)	BLYP	BP86
B3LYP1	BP86	B3LYP1	BLYP
B3LYP5	BLYP	B3LYP5	B3LYP1
PBE96	B3LYP1	PBE96	B3LYP5
PBE0	B3LYP5	PBE0	BHHLYP

B97-D B2PLYP B2GP-PLYP DSD-BLYP D-BLYP D-PBEP86	BHHLYP B3PW91 PBE96 PBE0 HCTH/120 OLYP B97-D B2PLYP B2GP-PLYP	MPW91LYP B97-D B2PLYP	B3PW91 PBE96 PBE0 MPW91LYP HCTH/120 OLYP B97-D B2PLYP B2GP-PLYP
--	---	-----------------------------	---

(*) The **TZ** keyword selects the use of a special set of DFT-D version 3 parameters optimized for Ahlrichs' TZVPP basis set. Its default value is **.FALSE.**

The parameters of the correction model can be changed with the **S6**, **RS6**, **S18**, **RS18**, and **ALP** keywords. These should be used if one uses a functional/method for which parameters are not stored internally, or if one would like to use custom parameters. Their functions are:

- S6** The s_6 global scaling factor, the main scaling factor in DFT-D version 2. For DFT-D version 3 and 4 it is of lesser importance and is usually set to 1.0 (except with double hybrid functionals).
- RS6** For DFT-D version 2, this parameter is used in calculating the dampening factor – its value is 1.1 (for all functionals). For DFT-D version 3, this parameter corresponds to the $s_{r,6}$ scaling factor, which is the main scaling factor in this version. For DFT-D version 4, it corresponds to the α_1 free fit parameter.
- S18** The s_8 scaling factor used in DFT-D version 3 and 4. For DFT-D version 2, this parameter has no function.
- RS18** For DFT-D version 3, this parameter is used in calculating the dampening factor, its value being 1.0 except with Slater exchange (where its value is 0.697). For DFT-D version 4, this parameter corresponds to the α_2 free fit parameter. For DFT-D version 2, this parameter has no function.
- ALP** The global scaling parameter of the damping function (which dampens the dispersion correction at short ranges). Its value is usually 20 for DFT-D version 2, and 14 for DFT-D version 3 and 4. For DFT-D version 4, this parameter is only used when **ABC=.T.** (see below).

The above parameter naming scheme is identical to the one used in the DFT-D3 program by Grimme. However, a few of these parameters have names which do not correspond to the parameters they represent, or have different meanings depending on the DFT-D version used. As we assume this can be confusing for some, aliases have been created for these parameters. Available aliases are:

Keyword	Aliases
RS6	SR6, ALPHA1
S18	S8

It is important to note that, when specifying a custom set of parameters, all five of the above parameters should be given a value, even those which are zero or not used by the DFT-D version used!

In addition to the above parameters, there are a few other parameters and keywords that should be mentioned. First of all, it is possible to control the coordination number dependent dispersion (used in DFT-D version 3 and 4) through the **K1**, **K2**, and **K3** keywords, which correspond to the k_1 , k_2 , and k_3 parameter, respectively. Their default values are **K1=16**, **K2=4/3**, and **K3=-4**. Typically, there is no need to change them.

Furthermore, it is possible to enable the three-body non-additive contribution to the dispersion correction. This is done by specifying **ABC=.T.** and pertains to DFT-D versions 3 and 4. By default, the three-body contribution is disabled as Grimme and co-workers found that inclusion of three-body terms had only a marginal impact overall while leading to poorer results in a few specific cases.

It should finally be noted that some parameters, such as C_6 and R_0 , cannot be controlled. The values of these parameters are used as specified in the references mentioned at the start of this section.

DFT with the nonlocal term of the VV10 functional (DFT-NL)

In addition to the DFT-D scheme discussed above, the performance of density functionals for long-range interactions can also be improved by use of the DFT-NL scheme (available as of Firefly 8.1.1). In this scheme, the nonlocal part of the VV10 van der Waals-density functional is used as an additional term for the functional specified with the **DFTTYP** keyword. The total exchange-correlation energy thus becomes:

$$E_{xc-NL} = E_x + E_c + E_{c-NL}$$

As implemented in Firefly, the NL term will be evaluated separately after the main DFT calculation has reached self-consistency. Because of this non-self-consistent implementation, analytical gradients (as well as time-dependent calculations) are not available. The reason for this approach is that a self-consistent implementation would be very expensive for large molecules. It should in addition also be noted that the complete VV10 functional is not available as rPW86 exchange is currently not implemented.

DFT-NL is requested by specifying:

```
$CONTRL DFTNL=.T. $END
```

Keywords of the **\$DFTNL** group can be used to control the NL part of the calculation. Noteworthy keywords are **B** and **C**, which correspond to the values of the b and C parameters of the VV10 functional, respectively. Hereby, the default value of **B** is **4.8** though optimized values (obtained from Grimme's group) are used for the following functionals:

```
BLYP, B3LYP1, B3LYP5, PBE96, PBE0, B3PW91, B2PLYP, B3P86, mPW1PW, revPBE, revPBE0
```

Finally, it should be pointed out that keywords of the **\$DFT** group that affect the accuracy of the calculation (such as **NRAD** and **LMAX**) do not affect the NL part. Instead, duplicates of these keywords in the **\$DFTNL** group can be used instead.

Relevant references are:

- O. A. Vydrov, T. Van Voorhis, J. Chem. Phys. (2010) 133, 244103
- W. Hujo, S. Grimme, J. Chem. Theory Comput. (2011) 7, 3866-3871