

BrianQC 1.2

User Manual

StreamNovation Ltd.

November 4, 2021

Contents

| | | |
|----------|--|-----------|
| 1 | Introduction | 2 |
| 2 | Prerequisites | 3 |
| 2.1 | Software requirements | 3 |
| 2.2 | System configuration requirements | 3 |
| 2.3 | Supported GPUs | 4 |
| 3 | Installing and uninstalling BrianQC | 4 |
| 3.1 | Installing BrianQC for Q-Chem | 4 |
| 3.2 | Installing BrianQC for PSI4 | 5 |
| 3.3 | Installing BrianQC as an SDK | 5 |
| 3.3.1 | Building the user-built components | 6 |
| 3.4 | Unattended install | 7 |
| 3.5 | Uninstalling | 8 |
| 4 | Configuring BrianQC | 8 |
| 4.1 | The configuration file | 8 |
| 4.1.1 | logLevels property | 9 |
| 4.1.2 | GPUNameFilter property | 9 |
| 4.1.3 | nodes property | 10 |
| 4.1.4 | Example configuration file | 11 |
| 4.2 | The license file | 12 |
| 4.3 | The kernel database files | 12 |
| 5 | Using BrianQC | 12 |
| 5.1 | Overview | 12 |
| 5.2 | General usage notes | 12 |
| 5.2.1 | Dual- and quad-GPU cards | 13 |
| 5.3 | Handling multiple GPUs | 13 |
| 5.4 | Supported calculations | 14 |
| 5.5 | Supported MM force fields | 15 |
| 5.6 | Additional utilities | 16 |

| | | |
|----------|---|-----------|
| 6 | Using BrianQC with Q-Chem | 16 |
| 6.1 | The input file | 16 |
| 6.2 | Setting the number of CPU threads | 16 |
| 6.3 | Using basis sets with g functions | 17 |
| 7 | Using BrianQC with PSI4 | 17 |
| 7.1 | Enabling BrianQC in PSI4 | 17 |
| 7.2 | Necessary and recommended option values | 17 |
| 8 | Using BrianQC as an SDK | 18 |
| 8.1 | SDK sample and example programs | 18 |
| 8.1.1 | Sample programs | 18 |
| 8.1.2 | Specifying the DFT grid for the sample programs | 20 |
| 8.1.3 | Specifying the DFT functional for the sample programs | 21 |
| 8.1.4 | Passing molecules to the sample programs | 21 |
| 8.1.5 | Example programs | 21 |
| 9 | Troubleshooting | 22 |
| 9.1 | “cuMemAlloc : CUDA error: CUDA_ERROR_OUT_OF_MEMORY” . | 22 |
| 9.2 | “cuMemAlloc : CUDA error: CUDA_ERROR_INVALID_VALUE” . . | 22 |
| 9.3 | “CUDA error: CUBLAS_STATUS_NOT_INITIALIZED” | 23 |
| 9.4 | “The computational device (...) is currently not supported” warning . . . | 23 |
| 9.5 | “Cannot find any suitable accelerator devices” error | 23 |
| 9.6 | Contact us | 23 |
| A | Supported DFT functionals | 24 |

1 Introduction

BrianQC is an advanced quantum chemistry software that speeds up every calculation that uses Coulomb or Exchange integrals over Gaussian basis functions or their first analytic derivative (including HF-SCF, DFT, SCF geometry optimization, DFT geometry optimization, Molecular Mechanics, etc.). BrianQC is optimized for simulating large molecules and has been tested for up to 40,000 Cartesian Gaussian basis functions with full support for s, p, d, f and g-type orbitals. BrianQC works on NVIDIA Kepler, Maxwell, Pascal, Volta, Turing and Ampere architectures and provides calculation results with double-precision accuracy.

There are two main methods to use BrianQC:

Through a host software BrianQC can potentially work with multiple quantum chemistry software suites and is currently integrated into Q-Chem and interfaced to PSI4 so users can easily try out the provided GPU-acceleration in their current workflows. This means that only minor modifications (e.g. a command-line argument or an environment variable) are necessary to calculate on GPU with BrianQC. Currently the following host software packages are supported:

1. Q-Chem 5.4.1 with BrianQC version 1.2.1

2. Q-Chem 5.4 with BrianQC version 1.2
3. PSI4 1.4 with BrianQC version 1.2
4. Q-Chem 5.3 with BrianQC version 1.1
5. Q-Chem 5.2.2 with BrianQC version 1.0
6. Q-Chem 5.2.0 and 5.2.1 with BrianQC version 0.9
7. Q-Chem 5.1.2 with BrianQC version 0.8
8. Q-Chem 5.1.1 with BrianQC version 0.7
9. Q-Chem 5.0 with BrianQC version 0.5

Some of these host programs might be at end-of-life; please inquire at support@brianqc.com for details. For using BrianQC with a specific software, please refer to later sections of the manual. If using a previous version of BrianQC, please refer to the corresponding manual. Please note that you might need a license not only for BrianQC, but for the specific software as well.

Directly as an SDK BrianQC is designed to work standalone as a Software Development Kit so it can be used to build various quantum chemistry simulation software on. This method is mainly aimed at developers wishing to create efficient simulation programs or extend their existing software with GPU-accelerated calculations. This manual is concerned mostly with the first method, i.e. running through a host software. For further instructions about using BrianQC as an SDK, please refer to the API documentation in the `.../<brianqc_install_dir>/docs/` folder.

2 Prerequisites

2.1 Software requirements

BrianQC has the following software dependencies:

- bash, sudo, chown, curl, wget, tar, gzip
- NVIDIA driver with CUDA support (libcuda.so.1), version 460.91 or higher

Under some Ubuntu versions, the default NVIDIA driver package (`nvidia-current-...`) contains an older CUDA version which BrianQC is not compatible with. Instead, download and install a compatible driver from the NVIDIA page. Attempting to use BrianQC with an old CUDA driver version might cause system instability.

2.2 System configuration requirements

On desktop systems, the same GPU should never be used for both display and BrianQC's calculations, as that might cause the display to become unresponsive. On newer Linux systems, the Xorg X server automatically initializes itself on all GPUs, and enables a kernel timeout to ensure that the display remains responsive. This interferes with BrianQC's long GPU calculations, and can be mitigated by properly configuring the X server.

The most typical use case is that the BIOS default GPU is used for rendering, and BrianQC uses only the other GPUs. In this case, the X server can be configured to ignore every GPU except the BIOS default. To do so, create a `noautogpu.conf` file in the `/etc/X11/xorg.conf.d/` directory with the following contents:

```
Section "ServerFlags"
    Option "AutoAddGPU" "off"
EndSection
```

Note that BrianQC will still need to be configured to ignore the display GPU; please see “Using BrianQC” for details.

2.3 Supported GPUs

The list of currently supported GPUs is displayed during installation (compute capability 6.0, 6.1, 7.0, 7.5, 8.0, 8.6). Furthermore the installer is capable to recognize the GPUs present in the system and detects the ones that are compatible with BrianQC. BrianQC requires at least 4 GB of GPU memory; however 8 GB or above are recommended for any moderately sized system.

BrianQC 1.2.1 is the last version with K80 support. Future BrianQC releases will not be compatible with Kepler architecture.

3 Installing and uninstalling BrianQC

3.1 Installing BrianQC for Q-Chem

The BrianQC installer is completely integrated within the Q-Chem installer. It requires working `curl` and `wget` command line tools. Within the Q-Chem installer choose the option matching your platform and labeled “with GPU support”. Proceed with the normal Q-Chem parts until the installation summary. The BrianQC installer will start automatically and query the system for compatible GPUs. Follow the instructions of the BrianQC installer. To perform the installation and a basic test of BrianQC, please follow the process below. If an error occurs, see the rest of this manual for troubleshooting.

Overview of the BrianQC installation process for Q-Chem:

1. Make sure that you have the supported NVIDIA driver installed (at least 460.91, or higher).
2. Make sure that no calculation is running on the GPUs in the machine.
3. Make sure that you have the Q-Chem installer.
4. Choose the Q-Chem binary option matching your platform, and having the label “with GPU support”.
5. The BrianQC installer runs automatically at the end of the Q-Chem installation. Follow the instructions below.

6. The BrianQC installer automatically detects that it is running from within the Q-Chem installer and sets up Q-Chem configuration to support BrianQC.
7. Run a calculation with GPU support by adding the `-gpu` command line argument like `qchem -nt 8 -gpu input.in` on an eight-core CPU.

The BrianQC installer is an interactive program that asks questions to configure the environment. BrianQC is installed under the Q-Chem installation directory into the `brianqc_qchem` folder. Overview of the BrianQC installation process:

1. Query the target system for compatible GPUs and ask which ones to download kernel databases for.
2. Download and install BrianQC and chosen GPU kernel databases.
3. Ask for user data, send it to our server and download a trial license if applicable. Please enter valid data here so we can generate a proper license for you.
4. Send an automated e-mail that confirms the installation.
5. Copy the trial license to the file `.../<brianqc_install_dir>/license.json`. Please do not change the filename.

3.2 Installing BrianQC for PSI4

To use BrianQC with PSI4, BrianQC must first be installed as an SDK, and the user-built components built; please see “Installing BrianQC as an SDK” and “Building the user-built components” for instructions.

After the BrianQC SDK has been successfully installed, PSI4 must be built from source with BrianQC enabled at compile time. For detailed instructions, please see the PSI4 manual section “Interface to the BrianQC GPU module by the BrianQC team”; the following is a short summary.

When CMake-ing PSI4, set the “ENABLE_BrianQC” CMake variable to 1 and set the “BrianQC_DIR” CMake variable to the path where BrianQC’s components have been built (usually `<brianqc_install_path>/build`), then build PSI4 normally.

3.3 Installing BrianQC as an SDK

For installing BrianQC as an SDK the ‘BRIANQC_SDK_INSTALL’ environment variable has to be set to 1. This type of install is mainly for developers. Starting the installer is slightly different depending on which installer type is used:

Default (online) installer Simply download and run the `1_2_brianqc_installer.bin` file and follow the instructions. Please note that the file must be given execute permissions before running it.

Offline installer Download and extract the `1_2_brianqc_offline_installer.tar.gz` archive and run the contained `brianqc_offline_installer.bin` file. Please note that the extracted contents of the archive must all reside in the same directory for the installer to work.

The BrianQC installer is an interactive program that asks questions to configure the environment. If you would like to carry out an unattended install, please see “Unattended install”.

Overview of the BrianQC installation process:

1. Ask the user for the installation directory.
2. Query the target system for compatible GPUs and ask which ones to download kernel databases for.
3. Download and install BrianQC and chosen GPU kernel databases.
4. Ask for user data, send it to our server and download a trial license if applicable. Please enter valid data here so we can generate a proper license for you.
5. Send an automated e-mail that confirms the installation.
6. Copy the trial license to the file `.../<brianqc_install_dir>/license.json`. Please do not change the filename.
7. Detect or ask for supported host software and their installation paths and configure them automatically.

3.3.1 Building the user-built components

The BrianQC SDK includes components that must be built locally. Detailed instruction can be found in the README file in the SDK install directory; the following is a short summary:

1. Create a build directory to keep the source tree clean.

```
cd <brianqc_install_path>
mkdir build
cd build
```

2. Configure project and generate makefiles with CMake. You will require Eigen and boost.

```
cmake ..
```

3. Build the examples and samples.

```
make
```

4. Test the installation by starting a small calculation. Make sure to set the ‘BRIANQC_SDK_INSTALL’ environment variable to <brianqc_install_path>!

```
export BRIANQC_INSTALL_PATH=<brianqc_install_path>
bin/sample_hf_and_dft \
--molecule ../share/qc_molecules/cis-decalin.raw \
--basis ../share/basis_sets/cc-pvdz
```

3.4 Unattended install

It is possible to install BrianQC in an automated way without any kind of manual user interaction. This is accomplished by supplying a JSON configuration file to the installer, which describes the options that the installer needs. An example JSON configuration file can be generated with the following command:

```
.../<installer_path> --generate-unattended-json
```

This will output a file called `brianqc_unattended_install_example.json` with the possible options. By default, the contents of the file are the following:

```
{
  "installDir": "~/brianqc",
  "createInstallDir": "Y",
  "forceInstall": "N",
  "doubleCheckForceInstall": "N",
  "fullName": "",
  "institute": "",
  "emailAddress": "",
  "defaultKernelDb": "",
  "haveQChem": "N",
  "QChemDir": "",
  "addEnvVarToBashrc": "Y"
}
```

The options and their meanings:

installDir The directory in which to install BrianQC.

createInstallDir Whether to create the installation directory if it does not exist. Possible choices are 'Y' (yes) or 'N' (no).

forceInstall Whether to force installation into the specified directory if it already contains files. Possible choices are 'Y' (yes) or 'N' (no).

doubleCheckForceInstall Forced installation must be double-confirmed. Possible choices are 'Y' (yes) or 'N' (no).

fullName The full name of the user corresponding to your BrianQC license.

institute The institute corresponding to your BrianQC license.

emailAddress The e-mail address corresponding to your BrianQC license.

defaultKernelDb Specifies which kernel DB to download during installation. If left as an empty string, it automatically detects and downloads the proper one based on the GPU hardware in the computer.

haveQChem Whether a supported and working Q-Chem version is installed on the computer.

QChemDir Directory in which Q-Chem is installed. Its config file will be automatically updated to support BrianQC. The **haveQChem** flag must be set to yes for it to take effect.

addEnvVarToBashrc Whether to add the `BRIANQC_INSTALL_PATH` environment variable to the `.bashrc` file of the user.

Please note that a valid name, institute and e-mail address are required for getting a proper license.

To perform an automated install, please create a JSON configuration file or extend the example and re-run the BrianQC installer as follows:

```
.../<installer_path> --unattended-json <unattended install JSON path>
```

3.5 Uninstalling

If you need to remove your installation of BrianQC, please remove the install directory. If the `BRIANQC_INSTALL_PATH` environment variable setting was written into a configuration file (e.g. `qcenv.sh` of Q-Chem or `.bashrc`), the corresponding lines can be removed.

4 Configuring BrianQC

During a normal installation, BrianQC will configure itself automatically, and without any further input from the user. However, if the hardware or software environment changes, it might be necessary to reconfigure or update the installation.

There are two methods to do this:

- Using the Q-Chem or the BrianQC installer to reinstall the software. Following the instructions of the installer, the configuration will be updated to match the current state. Additionally, the installer updates BrianQC to the latest patch version corresponding to the major and minor version of the installer. For example, the `v1.0` installer will update an installed `v1.0.0` version to `v1.0.1`, but **not** to `v1.1.0`.
- Editing the configuration file manually by following the instructions in the “The configuration file” section below. This enables a larger degree of freedom for setting up everything as desired, which might be useful in a complex configuration involving multiple nodes.

4.1 The configuration file

The software configures itself during the installation process, so if no problem occurs, there is no need for further configuration. This part contains information about the re-configuration of the software in case of an installation problem or changes in the hardware or software environment.

All configuration variables are stored in `.../<brianqc_install_dir>/config.json`. The config file of BrianQC is a standard JSON file storing all necessary and optional configuration parameters for the module. In the following there are the descriptions of the top-level JSON properties used by the BrianQC config.

brianRootDir JSON string containing the absolute path of the BrianQC installation. In the current version, it should always be `.../<brianqc_install_dir>/`.

logLevels JSON object controlling the verbosity of the BrianQC module. For details, see below.

kernelDBs JSON array of strings containing the absolute paths of the integrator kernel databases used by the software. Specifying different kernel databases is useful if the computer has various GPUs that should be used.

DFTDBs JSON array of strings containing the absolute paths of the Density Functional Theory kernel databases used by the software. Similarly to the integrator kernel databases, multiple can be specified for supporting various GPUs.

MMDBs JSON array of strings containing the absolute paths of the Molecular Mechanics kernel databases used by the software. Similarly to the integrator and DFT kernel databases, multiple can be specified for supporting various GPUs.

GPUNameFilter JSON string controlling which GPUs BrianQC tries to allocate. For details, see below.

nodes JSON object that can be used to selectively disable specific GPUs in specific nodes. For details, see below.

scratchDir JSON string specifying a path to the scratch directory. Currently only used for storing DIIS history on disk. Can be overridden by the `BRIANQC_SCRATCH_DIR` environment variable. Useful when simulating large systems and DIIS history is large.

4.1.1 logLevels property

logLevels is a JSON object with the properties `TRACE2`, `TRACE1`, `DEBUG2`, `DEBUG1`, `DEBUG`, `TIMING2`, `TIMING1`, `INFO`, `WARNING`, `DEVWARNING`. Each property can have the value `true` or `false`, disabling or enabling the respective type of log messages. Under normal circumstances, the default settings should suffice; the other log levels might be useful when sending a bug report.

4.1.2 GPUNameFilter property

GPUNameFilter is a JSON string. If specified, only the GPUs whose name contains this as a substring are used by BrianQC. This can be used to selectively disable certain GPUs in certain use cases:

- Example 1: One NVIDIA 1080 Ti GPU used for computation and one NVIDIA 970 GPU used for rendering:
 - In this case, BrianQC needs to be instructed to use only the 1080 Ti card so as not to interfere with rendering. This can be achieved, for example, by specifying “1080” as **GPUNameFilter**, which will exclude the 970 card even if otherwise there are kernels available for it.

- Example 2: One NVIDIA 1080 Ti GPU used for computation and another NVIDIA 1080 Ti GPU used for rendering:
 - In this case, the GPUs can only be distinguished based on their PCIe bus IDs. Running NVIDIA’s `nvidia-smi` tool gives the list of GPUs in the machine (complete with their bus IDs), and also specifies which card is allocated for the X windowing system. If, for example, the PCIe bus ID of GPU 0 is **00000000:01:00.0**, that of GPU 1 is **00000000:83:00.0**, and Xorg is running on GPU 1, then **GPUNameFilter** should be set to “01:00.0” to allow only GPU 0. Note that NVIDIA’s and BrianQC’s format is slightly different for the PCIe bus ID (the number of leading zeros might be different), so only the part after the first colon should be used.
- Example 3: Three NVIDIA 1080 Ti GPUs used for computation and another NVIDIA 1080 Ti GPU used for rendering:
 - In this case, the **GPUNameFilter** property cannot be used, as there is probably no string which is found in all of the computing GPUs’ full names or PCIe bus IDs but not in those of the rendering GPU. Instead, the `nvidia-smi` tool can be used to determine the PCIe bus IDs of the computing cards, and the **nodes** property should be used to specify the allowed GPUs.
- Example 4: Four NVIDIA 1080 Ti GPUs used for computation, and no rendering GPU (compute-only node with no monitor attached and no X server running):
 - If all available cards can be used by BrianQC, then **GPUNameFilter** doesn’t need to be specified, or can be set to an empty string.
- Example 5: One NVIDIA 1080 Ti GPU used for computation and one NVIDIA 630 GPU used for rendering:
 - Since BrianQC currently doesn’t support the GTX 630, **GPUNameFilter** doesn’t need to be specified (or can be set to an empty string), as BrianQC will ignore the rendering GPU either way.

4.1.3 nodes property

nodes is a JSON object, whose each property must be the MAC address of a node. The value corresponding to the property is itself a JSON object with the following properties:

GPUs JSON array of strings, each element must be the name of a GPU as reported by the `.../<brianqc_install_dir>/bin/list_compute_devices` utility.

maxAllocableGPUs Only the first this many GPUs are enabled from the list.

If a node has no corresponding entry (or the **nodes** property doesn’t exist) then all GPUs are enabled. If there is an entry for the node, then only the ones in the respective **GPUs** array are enabled. If there is an entry for the node, and it has a **maxAllocableGPUs** property, then only the first that many GPUs are enabled. Note that if a node has an entry, but the list of GPUs is empty (or no **GPUs** property is

present), then all GPUs will be disabled on that node. So, if you want to allow BrianQC to use any and all GPUs it finds on the computation nodes, then creating this section can be avoided entirely.

4.1.4 Example configuration file

```
{
  "brianRootDir": "/home/simulation/brianqc/",
  "logLevels": {
    "DEBUG": true
  },
  "kernelDBs": [
    "/home/simulation/brianqc/integrators/kernel.db"
  ],
  "DFTDBs": [
    "/home/simulation/brianqc/integrators/dft.db"
  ],
  "GPUNameFilter": "CUDA",
  "nodes": {
    "ab:cd:ef:00:11:23": {
      "maxAllocableGPUs": 1,
      "GPUs": [
        "CUDA GeForce GT 970#0000:01:00.0",
        "CUDA GeForce GTX 1080#0000:08:00.0",
        "CUDA GeForce GTX 1080#0000:09:00.0"
      ]
    },
    "ab:cd:ef:00:11:24": {
      "GPUs": [
        "CUDA GeForce GT 970#0000:01:00.0",
        "CUDA GeForce GTX 1080#0000:08:00.0",
        "CUDA GeForce GTX 1080#0000:09:00.0"
      ]
    },
    "ab:cd:ef:00:11:25": {
    }
  }
}
```

This configuration will instruct BrianQC to:

- enable DEBUG-level log messages during running;
- load the `kernel.db` integrator kernel database from the default location;
- load the `dft.db` DFT kernel database from the default location;
- enable a single GTX 970 GPU in the node "...:23";

- enable all listed GPUs in the node "...:24", but disable any other;
- disable all GPUs in the node "...:25";
- and enable all GPUs in any other nodes.

4.2 The license file

You receive the license file when you purchase BrianQC, and it should be placed in `.../<brianqc_install_dir>/license.json`. It contains some human-readable fields (for example, the expiration date), and it is cryptographically signed. Under normal circumstances, you should never modify it, as that might invalidate the signature, and render your installation unusable. Also, due to the way the signature is computed, transferring the license by copying and pasting might break its validity as well. If there is a problem with licensing, please contact us at support@brianqc.com, and we will help you resolve the issue.

4.3 The kernel database files

The kernel databases are located in `.../<brianqc_install_dir>/integrators/`, and their location is stored in the configuration file. Their default location is sufficient for most use cases, but see the note about reading kernel databases over NFS in "General usage notes". If the databases are relocated for any reason, their path must be updated in the configuration file; see "The configuration file" for details.

5 Using BrianQC

5.1 Overview

BrianQC is not a standalone executable; invocation always happens by running a host software as usual for an input file, and setting certain command-line arguments or environment variables to enable GPU computation. Note that only certain performance-intensive calculations will be performed on the GPU; the rest are done by the host software as usual.

To test the capabilities of BrianQC, some sample host programs are provided by the SDK installer; please refer to the "Installing BrianQC as an SDK" and "SDK sample and example programs" chapters.

5.2 General usage notes

- BrianQC reads a significant amount of data from the kernel databases in `.../<brianqc_install_dir>/integrators/`, which is inefficient to do over NFS, and may impact performance. See "The kernel database files" for details on how to move the databases to a different location.
- For symmetric molecules, results are only correct if c1 symmetry is enforced for the computation. See host software specific sections for details on input file settings.

- BrianQC's computation involves running some unusual kernels on the GPU, whose execution can potentially be very long. In desktop systems, due to the way the Linux kernel handles the display, executing kernels for a long time on the GPU which also handles the display might cause system instability. To avoid this, please make sure to forbid BrianQC to use the GPU handling the display. See the chapters about the configuration file and the environment variables for two ways of doing this.
- Currently, we only support using CUDA to access the GPUs; OpenCL support is in an experimental phase, and is currently disabled.
- BrianQC should work out of the box on clusters even if the master node has no GPUs (assuming that at least one slave node does). However, the installer might give a warning if it could not detect any GPUs in the node where it was run. This is not an error, and can be safely ignored.
- A bug has been encountered within the NVIDIA compiler, and the integrator kernels known to be affected have been removed; however, there might be other kernels affected. The bug has been reported to NVIDIA; however its content and status is not public as per NVIDIA policy. The bug, if triggered, may cause severe precision errors. In case of encountering this bug, please send us the input and the output so that we are able to supplement our bug report to NVIDIA.

5.2.1 Dual- and quad-GPU cards

BrianQC handles dual- and quad-GPU cards in a special way. ¹ When setting the number of GPUs to use, dual-GPU cards are counted as two devices. However, for the sake of licensing, they are considered to be one card. So, for example, if you wish to compute on a machine with 2 full K80 cards, then you will need to license BrianQC for two cards, but parametrize it to use four GPUs.

5.3 Handling multiple GPUs

BrianQC by default will use as many GPUs as possible during an execution. As a result if no environment variables of the above are given BrianQC will use all devices in your system allowed by the license. There are three ways to control the number of GPUs used for calculations:

- for all calculations with the GPUName property in the configuration file
- with the CUDA_VISIBLE_DEVICES environment variable, for more info visit this [page](#).
- with the BRIANQC_GPU_COUNT environment variable, in this way you specify only the number of GPUs and not the specific devices (BRIANQC_GPU_COUNT=2 will allow BrianQC to use two systems GPU)

¹Currently, the only such card that is supported is the NVIDIA Tesla K80.

For limiting BrianQC to 2 GPUs on linux with Q-Chem:

```
$ BRIANQC_GPU_COUNT=2 qchem -gpu <path to your input>/input.in
```

or alternatively:

```
$ export BRIANQC_GPU_COUNT=2
$ qchem -gpu <path to your input>/input.in
```

Similarly on windows platform:

```
set BRIANQC_GPU_COUNT=2
qchem -gpu <path to your input>/input.in
```

For selecting specific devices use the `CUDA_VISIBLE_DEVICES` environment variables (this will specify to run on the first two GPUs in the system):

```
$ CUDA_VISIBLE_DEVICES=0,1 qchem -gpu <path to your input>/input.in
```

Similarlay these environment variables work the same with other host softwares or with SDK samples (e.g. `sample_hf_and_df`).

5.4 Supported calculations

BrianQC is currently capable of performing several computational steps, including:

- Computing one- and two-electron integrals and their first derivatives;
- Computing the Hartree-Fock Coulomb and exact exchange matrices;
- Computing long range and short range part of exact exchange matrices;
- Computing the DFT exchange-correlation matrix and energy for a variety of functionals (see “Supported DFT functionals”);
- Performing all steps of the DIIS convergence acceleration method;
- Computing all terms of the molecular gradient of the Hartree-Fock energy;
- Computing the Fock-like terms of the CPHF equations;
- Computing the Molecular Mechanics energy and its gradient with a variety of force fields (see “Supported MM force fields”);
- Efficiently computing several linear algebra-heavy steps, such as diagonalizing the Fock matrix and computing the Fock-density commutator.

This enables the host program to speed up any job involving at least one of the listed computations. BrianQC’s accuracy and speedup has been tested for several job types:

- Single point HF energy calculation
- Single point DFT energy calculation

- Single point MM energy calculation
- HF geometry optimization
- DFT geometry optimization
- HF vibration analysis
- DFT vibration analysis
- MM analytic gradient calculation

Note that certain parts of the above jobs are still performed by the host program on the CPU. Thus, performance will still depend on the capabilities of the CPU and the host program's settings, for example, the thread count. Also, any other method involving the listed computational steps might use the GPU, but this has not been tested, so you should only enable BrianQC for the explicitly listed jobs.

The performance properties of GPU calculations are different from CPU implementations. In practice, this means that there is a lower bound for molecule size and basis function count, below which there is no advantage in moving the computations to the graphics card. For typical CPU-GPU combinations, this bound is approximately 500 basis functions, so you should only use BrianQC for inputs larger than that. Detailed benchmark data (including breakeven system sizes for certain hardware configurations) can be found at the Benchmarks section of the BrianQC homepage.

BrianQC has been tested on workstations with the following memory configurations:

- 16GB RAM up to 8000 basis functions
- 32GB RAM up to 13000 basis functions
- 128GB RAM up to 20000 basis functions
- 256GB RAM up to 40000 basis functions

5.5 Supported MM force fields

BrianQC supports the following Molecular Mechanics force fields:

- **MMFF94** (Merck Molecular Force Field) - References:
 Thomas A. Halgren, J. Comput. Chem. 17, 490-519 (1996) (10.1002/(SICI)1096-987X(199604)17:5/6<490::AID-JCC1>3.0.CO;2-P)
 Thomas A. Halgren, J. Comput. Chem. 17, 520-552 (1996) (10.1002/(SICI)1096-987X(199604)17:5/6<520::AID-JCC2>3.0.CO;2-W)
 Thomas A. Halgren, J. Comput. Chem. 17, 553-586 (1996) (10.1002/(SICI)1096-987X(199604)17:5/6<553::AID-JCC3>3.0.CO;2-T)
 Thomas A. Halgren, J. Comput. Chem. 17, 587-615 (1996) (10.1002/(SICI)1096-987X(199604)17:5/6<587::AID-JCC4>3.0.CO;2-Q)
 Thomas A. Halgren, J. Comput. Chem. 17, 616-641 (1996) (10.1002/(SICI)1096-987X(199604)17:5/6<616::AID-JCC5>3.0.CO;2-X)

- **UFF** (Universal Force Field) - References:
A. K. Rappe, C. J. Casewit, K. S. Colwell, W. A. Goddard III and W. M. Skiff, J. Am. Chem. Soc. 114 (25), 10024-10035 (1992) (10.1021/ja00051a040)

5.6 Additional utilities

BrianQC includes some additional utilities. Under normal circumstances, you will rarely need to use any of these.

list_compute_devices Lists all devices in the node that can be accessed through CUDA, usually including the CPU itself. Useful to get the full identifier of a GPU when creating a node descriptor in the configuration.

check_compute_devices Besides listing the computation devices, it also performs a basic test to see if they work properly.

6 Using BrianQC with Q-Chem

To use BrianQC, an appropriate version of Q-Chem (5.4 for BrianQC 1.2) is necessary, along with a Q-Chem license for all nodes which will be used for computation.

6.1 The input file

When BrianQC is enabled, Q-Chem should work with any regular input file as usual. However, there are certain fields in the input file which might impact the accuracy of the result.

- Due to the differing numerical characteristics of Q-Chem and BrianQC, precision might be suboptimal, unless Q-Chem is instructed to compute the one-electron integrals with high accuracy. This can be done by setting the **S2THRESH** keyword in the input file to at least 12 or preferably to 16. (Since computing the one-electron integrals takes very little time, this should not impact performance.)
- BrianQC currently only supports calculations with the c1 molecular symmetry group. If the input geometry has any symmetries, then Q-Chem should be forced to compute without exploiting that symmetry. This is done by setting the **SYM_IGNORE** keyword to True and the **SYMMETRY** keyword to False. If c1 symmetry is not enforced, the results might be incorrect.

6.2 Setting the number of CPU threads

Both Q-Chem and BrianQC use BLAS, LAPACK and similar linear algebra libraries. Q-Chem sets the number of threads used by these based on its **-nt** command line switch, which, if not given, defaults to 1. BrianQC cannot override the thread count set by the host program, which might impact the performance of calculations not handled by BrianQC. So, to ensure maximum performance from BrianQC, Q-Chem should be run with the number of threads set appropriately.

6.3 Using basis sets with g functions

In case of high precision basis sets with g functions like def2-qzvp or cc-pvqz the following parameters should be set in the **\$rem** section of the input file:

- Set **S2THRESH** to at least 12 or preferably to 16, as mentioned in “The input file”. This has minor 1-2% slowing effect on the speed of the calculation.
- Set **THRESH** to at least 9 since with default value (8) Q-Chem will not converge.

7 Using BrianQC with PSI4

To use BrianQC with PSI4, BrianQC must be installed in SDK mode and PSI4 compiled with BrianQC support. See Installing BrianQC for PSI4.

7.1 Enabling BrianQC in PSI4

There are two ways to enable BrianQC in PSI4 for a specific calculation:

- setting the **brianqc_enable** option in the PSI4 input file to True;
- setting the “BRIANQC_ENABLE” environment variable to 1.

The operation of BrianQC is transparent from a user’s point of view; every internal computation is either performed by PSI4 (if unsupported by BrianQC), or taken over by BrianQC, yielding the same result to within the required precision. In particular, BrianQC will respect all of PSI4’s usual precision parameters, like **ints_tolerance** and **e_convergence**.

BrianQC can speed up a number of internal computations, including Fock and gradient computation. Thus, BrianQC will speed up any calculation involving those terms, such as:

- HF and DFT single point energies
- HF and DFT geometry optimizations
- HF and DFT frequency analysis

Note that not every term of every calculation can be handled by BrianQC, thus, the actual speedup depends on the specifics of the calculation.

7.2 Necessary and recommended option values

To ensure that a calculation works with BrianQC, the following options need to be set when enabling BrianQC:

- BrianQC currently only handles the C1 molecular symmetry point group. Thus, if the molecule has any symmetries (which PSI4 would detect by default), the input geometry must contain the line “symmetry c1” to force PSI4 to disregard the symmetry.

To achieve peak performance, the following settings are recommended when enabling BrianQC:

- By default, PSI4 uses density-fitted preiterations for SCF, which BrianQC doesn't handle yet. To ensure that no non-BrianQC-accelerated iterations are performed, disable the preiterations by setting `df_scf_guess` to False.
- By default, PSI4 uses a disk-based Fock building, but BrianQC currently only accelerates direct Fock builds. To ensure that Fock building is accelerated by BrianQC, set `scf_type` to "direct".

8 Using BrianQC as an SDK

For in-depth capabilities, available functionality, and advice for writing a host program using the BrianQC API, please consult the API documentation.

8.1 SDK sample and example programs

The BrianQC SDK comes with several larger, more complex, parametrizable demonstration programs ("samples"), and several smaller, simpler, parameter-less demonstration programs ("examples").

8.1.1 Sample programs

The samples provided with the SDK have two dependencies:

- Boost (tested with 1.62)
- Eigen (tested with 3.1.2)

We recommend using the packaged version of Boost and Eigen of your selected distribution. If this is not possible then please be aware that Eigen is a header-only linear algebra library. For the manual building of Boost please consult the Boost home page <https://www.boost.org>. Project files in CMake are available for the provided samples. The source codes are located in the `.../<brianqc_install_dir>/samples` directory.

We provide the following samples:

sample_hf_and_dft Capable of calculating Hartree-Fock level or Density Functional Theory level energy. It has the following program options:

- `--molecule` Input molecule geometry, the path to an .xyz or .raw file. See below for a description of the .raw format.
- `--basis` NWChem formatted basis set full path.
- `--basis-type` Overrides the default cartesian/spherical setting for the basis; must be either 'spherical' or 'cartesian'.
- `--unrestricted` If true, use UHF, otherwise use RHF. Note that currently the sample program does not support ROHF.

--initial-guess Initial guess to use; must be either 'core', 'diagonal', or 'atomic'
--overlap-threshold Overlap eigenvalue threshold for basis orthonormalization.
 Default is 6 meaning 10^{-6} is the actual threshold value.
--integral-threshold Integral computation threshold. Default is 8 meaning
 10^{-8} is the actual value.
--differential-density Whether to use differential density. **Please note that
 differential density is a trade-off between stability and performance!**
--differential-density-reset If not zero, then reset differential density (and
 DIIS, if enabled) after this many iterations.
--diis Whether to enable DIIS.
--diis-algorithm DIIS algorithm to use; must be either 'basic' or 'stabilized'.
--diis-start-iter DIIS start iteration.
--diis-history-size DIIS history size.
--diis-commutator-method DIIS commutator method to use; must be either 'ba-
 sic', 'projected', 'corrected', or 'cholesky'.
--diis-reset-if-failed If true, reset DIIS history if DIIS extrapolation is failed.
--scf-convergence-condition SCF convergence condition; must be either 'energy-
 and-density-difference' or 'commutator-norm'.
--scf-convergence-threshold SCF convergence threshold. Default value is 5
 meaning 10^{-5} is the actual threshold value.
--scf-max-iter SCF max iterations. Default is 50.
--dft-functional DFT functional name; 'HF' disables DFT. Default is 'HF' See
 "Supported DFT functionals" for a list of supported DFT functionals.
--dft-grid Name of the DFT grid, like 'SG-0', default is 'EML(50,110)' See below
 for supported grids.

sample_hf_geom_opt Capable of calculating Hartree-Fock level geometry optimization.
 It has the following program options additionally to the options of **sample_hf_and_**
dft:

--opt-coords Geometry optimization coordinate system; must be either 'carte-
 sian' or 'z-matrix', default is 'z-matrix'.
--opt-method Geometry optimization method; must be either 'gradient', 'conjugate-
 gradient', 'newton' or 'eigenvector-following'; default is 'eigenvector-following'.
--opt-max-iter Geometry optimization max iterations, default is 50.
--opt-max-step Geometry optimization max step size, default is 0.3.
--opt-gradient-threshold Geometry optimization gradient threshold, default
 is 3×10^{-4} .
--opt-displacement-threshold Geometry optimization displacement threshold,
 default is 1.2×10^{-6} .

--opt-energy-threshold Geometry optimization energy threshold, default is 10^{-6} .

The geometry optimization is considered converged if from the 3 available conditions (gradient, displacement or energy) the gradient is below the convergence threshold and either the displacement or the energy difference is below the convergence threshold. Please note that Density Functional Theory level geometry optimization is not yet supported in the current version.

sample_mm Capable of calculating Molecular Mechanics level energy and spatial Cartesian gradient of the energy.

--molecule Input molecule geometry, the path to an .sdf file.

--forcefield Force field to use; must be either 'mmff94' or 'uff'.

sample_oniom Capable of calculating ONIOM level energy. It has the following program options additionally to the options of both **sample_hf_and_dft** and **sample_mm**:

--model-atom-indices Zero-based indices of the atoms of the ONIOM model system.

8.1.2 Specifying the DFT grid for the sample programs

The values accepted by the **--dft-grid** argument, and their respective meanings are the following:

SG-x Standard grids SG-0, SG-1, SG-2 and SG-3. Note that standard grids are not defined for all atoms.

EML(x,y) Euler-Maclaurin radial grid with x points, Lebedev angular grid with y points.

MKL(x,y) Mura-Knowles radial grid with x points, Lebedev angular grid with y points.

MEL(x,y) MultiExp radial grid with x points, Lebedev angular grid with y points.

DEL(x,y) Double Exponential radial grid with x points, Lebedev angular grid with y points.

TAL(x,y) Treutler-Ahlrichs radial grid with x points, Lebedev angular grid with y points.

Please note that while all radial grids are defined for all possible sizes, the Lebedev angular grid is only defined for certain sizes. The list of valid values can be found in the literature, or in the entry for `brianCOMGenerateDFTAngularAtomicGrid` in the API documentation.

8.1.3 Specifying the DFT functional for the sample programs

The `--dft-functional` argument of the sample programs accepts either the name of a single DFT functional, or a linear combination of functionals. To find the name of each supported functional used internally by BrianQC, please check the macro names “DFT functional IDs” chapter of the API documentation. The names accepted by the sample program are the same as the macro names, minus the `BRIAN_FUNCTIONAL_` prefix. Functional names are case-insensitive. So, for example, the B3LYP exchange-correlation functional can be specified by passing `hgga_b3lyp_xc` as the `--dft-functional` argument. The special functional name “HF” is available to specify the Hartree-Fock exact exchange, which also disables the DFT computation.

To pass a linear combination of functionals, concatenate the functional names with plus (+) signs, and optionally follow each functional with a constant multiplier, separated by an asterisk (*). For example, to specify the sum of B3LYP, 0.5 times PBE, and 0.1 times PW, the appropriate string is “`hgga_b3lyp_xc+gga_pbe_x*0.5+lda_pw_c*0.1`”, where the multiplier of the B3LYP term has been omitted, as it is 1. Note that the parameter string must not contain any whitespaces.

8.1.4 Passing molecules to the sample programs

The HF and DFT samples accept two molecule file formats: the standard `.xyz` and the custom `.raw`. A `.raw` file is essentially the `$molecule` section of a Q-Chem input file; first line is the charge of the molecule, followed by the spin multiplicity. The remaining lines are atoms in the same format as in an `.xyz` file. Note that currently the charge and spin multiplicity can only be set with a `.raw` input file, and they are forced to the defaults when passing a `.xyz` as input.

8.1.5 Example programs

We also provide small examples that show how to implement low-level functionality with BrianQC, for example handling basis sets, building a Fock matrix, initializing a DFT grid etc. The source codes for these examples are located in the `.../<brianqc_install_dir>/examples` directory. Most example programs print a simple message if they complete successfully; some compare the results against a reference, and report the difference.

Demonstrating basic API usage

example_multiple_init Shows initializing, releasing and reinitializing BrianQC.

example_basis Shows setting molecule and basis data.

Demonstrating HF single-point computations

example_1e_build Shows computing one-electron integrals.

example_fock_build Shows computing the Coulomb and exact exchange matrices.

example_diagonalization Shows diagonalizing the Fock matrix. Prints the difference between the original Fock matrix, and the one reconstructed from the diagonalization results.

Demonstrating DFT usage

example_dft_simple_functional Shows setting a DFT functional.

example_dft_composite_functional Shows setting a linear combination of DFT functionals.

example_dft_grid Shows setting the DFT grid.

Demonstrating HF gradient computations

example_1e_gradient Shows computing the one-electron contributions to the energy gradient.

example_repulsion_gradient Shows computing the two-electron contribution to the energy gradient.

9 Troubleshooting

9.1 “cuMemAlloc : CUDA error: CUDA_ERROR_OUT_OF_MEMORY”

On GPUs with memory 6GB or lower, BrianQC might crash with the error message in the section title. In these cases try to set the following environment variable:

```
export BRIANQC_FORCE_THREADS_PER_GPU=2
```

Restart your calculation. You can check the effect taking place by viewing BrianQC logs at startup.

It should have a line:

```
tid: 0x0 [2019.04.25-16:26:58.936] [INFO] Allocating 2 threads for GPU: XXXX
```

Instead of:

```
tid: 0x0 [2019.04.25-16:26:58.936] [INFO] Allocating 3 threads for GPU: XXXX
```

9.2 “cuMemAlloc : CUDA error: CUDA_ERROR_INVALID_VALUE”

This issue might be present when the computer is using disk swap due to high memory usage and the NVIDIA driver is trying to allocate pinned host memory. In this case try to disable pinned memory usage by setting the following environment variable:

```
export BRIANQC_DISABLE_PINNED_MEMORY_USAGE=1
```

Restart your calculation. You can check the effect taking place by viewing BrianQC logs at startup.

It should have a line:

```
tid: 0x0 [2020.01.06-12:52:04.052] [WARNING] Pinned memory usage is disabled! This might result in significant performance loss!
```

Please note that, as described by the warning, this might cause significant performance loss!

9.3 “CUDA error: CUBLAS_STATUS_NOT_INITIALIZED”

This issue is caused by running BrianQC with an old NVIDIA driver. This is detected during execution and the following message appears:

Please install the newest version of the NVIDIA display driver from <https://www.nvidia.com/Download/index.aspx> If that fails to solve the problem, please contact us at support@brianqc.com

We advise to install the driver in DKMS mode without the OpenGL files in case the display is not handled by the GPUs used for BrianQC. For example:

```
sudo ./NVIDIA-Linux-x86_64-470.63.01.run --dkms --no-opengl-files
```

9.4 “The computational device (...) is currently not supported” warning

This error message means that BrianQC has found a GPU in your system that it was allowed to use, but no compatible kernels are available. Since by default, the BrianQC installer only downloads kernel databases for a single GPU type, you might get this warning if there are any other types of GPUs in the system. If you don’t need to use the other cards, then this warning can safely be ignored (or eliminated by restricting BrianQC to the appropriate GPUs). However, if you need to use BrianQC in a GPU-heterogenous system, please contact us, and we will help you receive the appropriate kernel databases and configure BrianQC to use them.

You might also get this warning - along with the error “Couldn’t allocate any threads on any of the nodes” - if none of the kernel databases you have are compatible with any of your GPUs. If that is the case, please contact us to receive the appropriate database. Alternatively, you can rerun the BrianQC installer to select the appropriate database for download.

9.5 “Cannot find any suitable accelerator devices” error

During installation or reconfiguration, you might get the “Error: I cannot find any suitable accelerator devices on the machine, check CUDA or OpenCL configuration” message. This usually means that there is a problem with the CUDA configuration of your system. Please try or installing or reinstalling CUDA (at least version 11.2 is needed). Make sure to reinstall BrianQC afterwards by running the installer again.

9.6 Contact us

If your issue is not listed here or you have any other questions or general remarks, please contact us at support@brianqc.com or by filling out the “Ask and report” form on our website <https://brianqc.com>. We will answer as soon as possible.

A Supported DFT functionals

In cases where BrianQC can compute the exchange contribution separately it is denoted by (**X**) at the end of the functional name. In cases where BrianQC can compute the correlation contribution separately it is denoted by (**C**) at the end of the functional name. In cases where BrianQC can compute the combined exchange-correlation contribution jointly it is denoted by (**XC**) at the end of the functional name.

- **EXACT_EXCHANGE**
Exact Hartree-Fock exchange.
- **EXACT_EXCHANGE_LONG_RANGE**
Long-range component of the Hartree-Fock exact exchange.
- **EXACT_EXCHANGE_SHORT_RANGE**
Short-range component of the Hartree-Fock exact exchange.
- **GGA_AIRY_X**
Constantin et al based on the Airy gas
References:
L. A. Constantin, A. Ruzsinszky, and J. P. Perdew, Phys. Rev. B 80, 035125 (2009) (10.1103/PhysRevB.80.035125)
- **GGA_AK13_X**
Armiento & Kuemmel 2013
References:
R. Armiento and S. Kümmel, Phys. Rev. Lett. 111, 036402 (2013) (10.1103/PhysRevLett.111.036402)
- **GGA_AM05_C**
Armiento & Mattsson 05
References:
R. Armiento and A. E. Mattsson, Phys. Rev. B 72, 085108 (2005) (10.1103/PhysRevB.72.085108)
A. E. Mattsson, R. Armiento, J. Paier, G. Kresse, J. M. Wills, and T. R. Mattsson, J. Chem. Phys. 128, 084714 (2008) (10.1063/1.2835596)
- **GGA_AM05_X**
Armiento & Mattsson 05
References:
R. Armiento and A. E. Mattsson, Phys. Rev. B 72, 085108 (2005) (10.1103/PhysRevB.72.085108)
A. E. Mattsson, R. Armiento, J. Paier, G. Kresse, J. M. Wills, and T. R. Mattsson, J. Chem. Phys. 128, 084714 (2008) (10.1063/1.2835596)
- **GGA_APBE_C**
mu fixed from the semiclassical neutral atom
References:
L. A. Constantin, E. Fabiano, S. Laricchia, and F. Della Sala, Phys. Rev. Lett. 106, 186406 (2011) (10.1103/PhysRevLett.106.186406)
- **GGA_APBE_X**
mu fixed from the semiclassical neutral atom
References:
L. A. Constantin, E. Fabiano, S. Laricchia, and F. Della Sala, Phys. Rev. Lett. 106, 186406 (2011) (10.1103/PhysRevLett.106.186406)

- **GGA_B86_MGC_X**
Becke 86 with modified gradient correction
References:
A. D. Becke, J. Chem. Phys. 84, 4524 (1986) (10.1063/1.450025)
A. D. Becke, J. Chem. Phys. 85, 7184 (1986) (10.1063/1.451353)
- **GGA_B86_R_X**
Revised Becke 86 with modified gradient correction
References:
I. Hamada, Phys. Rev. B 89, 121103 (2014) (10.1103/PhysRevB.89.121103)
A. D. Becke, J. Chem. Phys. 84, 4524 (1986) (10.1063/1.450025)
A. D. Becke, J. Chem. Phys. 85, 7184 (1986) (10.1063/1.451353)
- **GGA_B86_X**
Becke 86
References:
A. D. Becke, J. Chem. Phys. 84, 4524 (1986) (10.1063/1.450025)
- **GGA_B88_X**
Becke 88
References:
A. D. Becke, Phys. Rev. A 38, 3098 (1988) (10.1103/PhysRevA.38.3098)
- **GGA_B88M_X**
Becke 88 reoptimized to be used with tau1
References:
E. Proynov, H. Chermette, and D. R. Salahub, J. Chem. Phys. 113, 10013 (2000) (10.1063/1.1321309)
- **GGA_B97_D3_XC**
Becke 97-D
References:
S. Grimme, J. Comput. Chem. 27, 1787 (2006) (10.1002/jcc.20495)
- **GGA_B97_D_XC**
Becke 97-D
References:
S. Grimme, J. Comput. Chem. 27, 1787 (2006) (10.1002/jcc.20495)
- **GGA_B97_GGA1_XC**
Becke 97 GGA-1
References:
A. J. Cohen and N. C. Handy, Chem. Phys. Lett. 316, 160 (2000) (10.1016/S0009-2614(99)01273-7)
- **GGA_BAYESIAN_X**
Bayesian best fit for the enhancement factor
References:
J. J. Mortensen, K. Kaasbjerg, S. L. Frederiksen, J. K. Nørskov, J. P. Sethna, and K. W. Jacobsen, Phys. Rev. Lett. 95, 216401 (2005) (10.1103/PhysRevLett.95.216401)
- **GGA_BCGP_C**
Burke, Cancio, Gould, and Pittalis
References:
K. Burke, A. Cancio, T. Gould, and S. Pittalis, ArXiv e-prints (2014), arXiv:1409.4834 [cond-mat.mtrl-sci].

- **GGA_BCGP_X**
Burke, Cancio, Gould, and Pittalis
References:
K. Burke, A. Cancio, T. Gould, and S. Pittalis, ArXiv e-prints (2014), arXiv:1409.4834 [cond-mat.mtrl-sci].
- **GGA_BEEFVDW_X**
BEEF-vdW exchange
References:
J. Wellendorff, K. T. Lundgaard, A. M{gelh{j}}, V. Petzold, D. D. Landis, J. K. N{rskov}, T. Bligaard, and K. W. Jacobsen, }Phys. Rev. B 85, 235149 (2012) (10.1103/PhysRevB.85.235149)
- **GGA_BEEFVDW_XC**
BEEF-vdW exchange-correlation
References:
J. Wellendorff, K. T. Lundgaard, A. M{gelh{j}}, V. Petzold, D. D. Landis, J. K. N{rskov}, T. Bligaard, and K. W. Jacobsen, }Phys. Rev. B 85, 235149 (2012) (10.1103/PhysRevB.85.235149)
- **GGA_BLYP_XC**
B88 exchange and LYP correlation
Functional components: GGA_B88_X + GGA_LYP_C
References:
A. D. Becke, Phys. Rev. A 38, 3098 (1988) (10.1103/PhysRevA.38.3098)
C. Lee, W. Yang, and R. G. Parr, Phys. Rev. B 37, 785 (1988) (10.1103/PhysRevB.37.785)
B. Miehlich, A. Savin, H. Stoll, and H. Preuss, Chem. Phys. Lett. 157, 200 (1989) (10.1016/0009-2614(89)87234-3)
- **GGA_BMK_C**
Boese-Martin for kinetics
References:
A. D. Boese and J. M. L. Martin, J. Chem. Phys. 121, 3405 (2004) (10.1063/1.1774975)
- **GGA_BPCCAC_X**
BPCCAC (GRAC for the energy)
References:
E. Brémond, D. Pilard, I. Ciofini, H. Chermette, C. Adamo, and P. Cortona, Theor. Chem. Acc. 131, 1184 (2012) (10.1007/s00214-012-1184-0)
- **GGA_C09X_X**
C09x to be used with the VdW of Rutgers-Chalmers
References:
V. R. Cooper, Phys. Rev. B 81, 161104 (2010) (10.1103/PhysRevB.81.161104)
- **GGA_CAP_X**
Correct Asymptotic Potential
References:
J. Carmona-Espíndola, J. L. Gázquez, A. Vela, and S. B. Trickey, J. Chem. Phys. 142, 054105 (2015), 10.1063/1.4906606 (10.1063/1.4906606)
- **GGA_CHACHIYO_X**
Chachiyo exchange
References:
T. {Chachiyo and H. {Chachiyo}, }ArXiv e-prints (2017), arXiv:1706.01343 [cond-mat.mtrl-sci].
- **GGA_CS1_C**
A dynamical correlation functional
References:

- N. C. Handy and A. J. Cohen, *J. Chem. Phys.* 116, 5411 (2002) (10.1063/1.1457432)
E. I. Proynov and A. J. Thakkar, *Int. J. Quantum Chem.* 106, 436 (2006) (10.1002/qua.20758)
- **GGA_DK87_R1_X**
dePristo & Kress 87 version R1
References:
A. E. DePristo and J. D. Kress, *J. Chem. Phys.* 86, 1425 (1987) (10.1063/1.452230)
 - **GGA_DK87_R2_X**
dePristo & Kress 87 version R2
References:
A. E. DePristo and J. D. Kress, *J. Chem. Phys.* 86, 1425 (1987) (10.1063/1.452230)
 - **GGA_EB88_X**
Non-empirical (excogitated) B88 functional of Becke and Elliott
References:
P. Elliott and K. Burke, *Can. J. Chem.* 87, 1485 (2009) (10.1139/V09-095)
 - **GGA_EDF1_XC**
EDF1
References:
R. D. Adamson, P. M. W. Gill, and J. A. Pople, *Chem. Phys. Lett.* 284, 6 (1998) (10.1016/S0009-2614(97)01282-7)
 - **GGA_EV93_X**
Engel and Vosko
References:
E. Engel and S. H. Vosko, *Phys. Rev. B* 47, 13164 (1993) (10.1103/PhysRevB.47.13164)
 - **GGA_FT97_A_X**
Filatov & Thiel 97 (version A)
References:
M. Filatov and W. Thiel, *Mol. Phys.* 91, 847 (1997) (10.1080/002689797170950)
 - **GGA_FT97_B_X**
Filatov & Thiel 97 (version B)
References:
M. Filatov and W. Thiel, *Mol. Phys.* 91, 847 (1997) (10.1080/002689797170950)
 - **GGA_FT97_C**
Filatov & Thiel correlation
References:
M. Filatov and W. Thiel, *Int. J. Quantum Chem.* 62, 603 (1997) (10.1002/(SICI)1097-461X(1997)62:6<603::AID-QUA4>3.0.CO;2-#)
M. Filatov and W. Thiel, *Mol. Phys.* 91, 847 (1997) (10.1080/002689797170950)
 - **GGA_G96_X**
Gill 96
References:
P. M. W. Gill, *Mol. Phys.* 89, 433 (1996) (10.1080/002689796173813)
 - **GGA_GAM_X**
Minnesota GAM exchange functional
References:
H. S. Yu, W. Zhang, P. Verma, X. He, and D. G. Truhlar, *Phys. Chem. Chem. Phys.* 17, 12146 (2015) (10.1039/C5CP01425E)

- **GGA_GAPC_C**
GapC
References:
E. Fabiano, P. E. Trevisanutto, A. Terentjevs, and L. A. Constantin, *J. Chem. Theory Comput.* 10, 2016 (2014), pMID: 26580528 (10.1021/ct500073b)
- **GGA_GAPLOC_C**
Gaploc
References:
E. Fabiano, P. E. Trevisanutto, A. Terentjevs, and L. A. Constantin, *J. Chem. Theory Comput.* 10, 2016 (2014), pMID: 26580528 (10.1021/ct500073b)
- **GGA_GG99_X**
Gilbert and Gill 1999
References:
A. T. Gilbert and P. M. Gill, *Chem. Phys. Lett.* 312, 511 (1999) (10.1016/S0009-2614(99)00836-2)
- **GGA_HCTH_120_XC**
HCTH/120
References:
A. D. Boese, N. L. Doltsinis, N. C. Handy, and M. Sprik, *J. Chem. Phys.* 112, 1670 (2000) (10.1063/1.480732)
- **GGA_HCTH_147_XC**
HCTH/147
References:
A. D. Boese, N. L. Doltsinis, N. C. Handy, and M. Sprik, *J. Chem. Phys.* 112, 1670 (2000) (10.1063/1.480732)
- **GGA_HCTH_407_XC**
HCTH/407
References:
A. D. Boese and N. C. Handy, *J. Chem. Phys.* 114, 5497 (2001) (10.1063/1.1347371)
- **GGA_HCTH_407P_XC**
HCTH/407+
References:
A. D. Boese, A. Chandra, J. M. L. Martin, and D. Marx, *J. Chem. Phys.* 119, 5965 (2003) (10.1063/1.1599338)
- **GGA_HCTH_93_XC**
HCTH/93
References:
F. A. Hamprecht, A. J. Cohen, D. J. Tozer, and N. C. Handy, *J. Chem. Phys.* 109, 6264 (1998) (10.1063/1.477267)
- **GGA_HCTH_A_C**
HCTH-A
References:
F. A. Hamprecht, A. J. Cohen, D. J. Tozer, and N. C. Handy, *J. Chem. Phys.* 109, 6264 (1998) (10.1063/1.477267)
- **GGA_HCTH_A_X**
HCTH-A
References:
F. A. Hamprecht, A. J. Cohen, D. J. Tozer, and N. C. Handy, *J. Chem. Phys.* 109, 6264 (1998) (10.1063/1.477267)

- **GGA_HCTH_P14_XC**
HCTH $p=1/4$
References:
G. Menconi, P. J. Wilson, and D. J. Tozer, *J. Chem. Phys.* 114, 3958 (2001) (10.1063/1.1342776)
- **GGA_HCTH_P76_XC**
HCTH $p=7/6$
References:
G. Menconi, P. J. Wilson, and D. J. Tozer, *J. Chem. Phys.* 114, 3958 (2001) (10.1063/1.1342776)
- **GGA_HERMAN_X**
Herman Xalphabeta GGA
References:
F. Herman, J. P. V. Dyke, and I. B. Ortenburger, *Phys. Rev. Lett.* 22, 807 (1969) (10.1103/PhysRevLett.22.807)
F. Herman, I. B. Ortenburger, and J. P. V. Dyke, *Int. J. Quantum Chem.* 4, 827 (1969) (10.1002/qua.560040746)
- **GGA_HJS_B88_V2_X**
HJS screened exchange B88 corrected version
References:
E. Weintraub, T. M. Henderson, and G. E. Scuseria, *J. Chem. Theory Comput.* 5, 754 (2009) (10.1021/ct800530u)
- **GGA_HJS_B88_X**
HJS screened exchange B88 version
References:
T. M. Henderson, B. G. Janesko, and G. E. Scuseria, *J. Chem. Phys.* 128, 194105 (2008) (10.1063/1.2921797)
- **GGA_HJS_B97X_X**
HJS screened exchange B97x version
References:
T. M. Henderson, B. G. Janesko, and G. E. Scuseria, *J. Chem. Phys.* 128, 194105 (2008) (10.1063/1.2921797)
- **GGA_HJS_PBE_SOL_X**
HJS screened exchange PBE_SOL version
References:
T. M. Henderson, B. G. Janesko, and G. E. Scuseria, *J. Chem. Phys.* 128, 194105 (2008) (10.1063/1.2921797)
- **GGA_HJS_PBE_X**
HJS screened exchange PBE version
References:
T. M. Henderson, B. G. Janesko, and G. E. Scuseria, *J. Chem. Phys.* 128, 194105 (2008) (10.1063/1.2921797)
- **GGA_HLE16_XC**
high local exchange 2016
References:
P. Verma and D. G. Truhlar, *J. Phys. Chem. Lett.* 8, 380 (2017), pMID: 28033712 (10.1021/acs.jpcclett.6b02757)
- **GGA_HTBS_X**
Haas, Tran, Blaha, and Schwarz
References:

- P. Haas, F. Tran, P. Blaha, and K. Schwarz, Phys. Rev. B 83, 205117 (2011) (10.1103/PhysRevB.83.205117)
- **GGA_HYB_TAU_HCTH_C**
correlation part of hyb-tau-hcth
References:
A. D. Boese and N. C. Handy, J. Chem. Phys. 116, 9559 (2002) (10.1063/1.1476309)
 - **GGA_ITYH_X**
Short-range recipe for exchange GGA functionals
References:
H. Iikura, T. Tsuneda, T. Yanai, and K. Hirao, J. Chem. Phys. 115, 3540 (2001) (10.1063/1.1383587)
 - **GGA_KGG99_X**
Gilbert and Gill 1999 (mixed)
References:
A. T. Gilbert and P. M. Gill, Chem. Phys. Lett. 312, 511 (1999) (10.1016/S0009-2614(99)00836-2)
 - **GGA_KT1_X**
Exchange part of Keal and Tozer version 1
References:
T. W. Keal and D. J. Tozer, J. Chem. Phys. 119, 3015 (2003) (10.1063/1.1590634)
 - **GGA_KT1_XC**
Keal and Tozer, version 1
References:
T. W. Keal and D. J. Tozer, J. Chem. Phys. 119, 3015 (2003) (10.1063/1.1590634)
 - **GGA_KT2_XC**
Keal and Tozer, version 2
References:
T. W. Keal and D. J. Tozer, J. Chem. Phys. 119, 3015 (2003) (10.1063/1.1590634)
 - **GGA_LAG_X**
Local Airy Gas
References:
L. Vitos, B. Johansson, J. Kollár, and H. L. Skriver, Phys. Rev. B 62, 10046 (2000) (10.1103/PhysRevB.62.10046)
 - **GGA_LAMBDA_CH_N_X**
lambda_CH(N) version of PBE
References:
M. M. Odashima, K. Capelle, and S. B. Trickey, J. Chem. Theory Comput. 5, 798 (2009) (10.1021/ct8005634)
 - **GGA_LAMBDA_LO_N_X**
lambda_LO(N) version of PBE
References:
M. M. Odashima, K. Capelle, and S. B. Trickey, J. Chem. Theory Comput. 5, 798 (2009) (10.1021/ct8005634)
 - **GGA_LAMBDA_OC2_N_X**
lambda_OC2(N) version of PBE
References:
M. M. Odashima, K. Capelle, and S. B. Trickey, J. Chem. Theory Comput. 5, 798 (2009) (10.1021/ct8005634)

- **GGA_LB_X**
van Leeuwen & Baerends
References:
R. van Leeuwen and E. J. Baerends, Phys. Rev. A 49, 2421 (1994) (10.1103/PhysRevA.49.2421)
- **GGA_LBM_X**
van Leeuwen & Baerends modified
References:
P. R. T. Schipper, O. V. Gritsenko, S. J. A. van Gisbergen, and E. J. Baerends, J. Chem. Phys. 112, 1344 (2000) (10.1063/1.480688)
- **GGA_LG93_X**
Lacks & Gordon 93
References:
D. J. Lacks and R. G. Gordon, Phys. Rev. A 47, 4681 (1993) (10.1103/PhysRevA.47.4681)
- **GGA_LM_C**
Langreth & Mehl
References:
D. C. Langreth and M. J. Mehl, Phys. Rev. Lett. 47, 446 (1981) (10.1103/PhysRevLett.47.446)
C. D. Hu and D. C. Langreth, Phys. Scr. 32, 391 (1985) (10.1088/0031-8949/32/4/024)
- **GGA_LV_RPW86_X**
Berland and Hyldgaard
References:
K. Berland and P. Hyldgaard, Phys. Rev. B 89, 035412 (2014) (10.1103/PhysRevB.89.035412)
- **GGA_LYP_C**
Lee, Yang & Parr
References:
C. Lee, W. Yang, and R. G. Parr, Phys. Rev. B 37, 785 (1988) (10.1103/PhysRevB.37.785)
B. Miehlich, A. Savin, H. Stoll, and H. Preuss, Chem. Phys. Lett. 157, 200 (1989) (10.1016/0009-2614(89)87234-3)
- **GGA_MB88_X**
Modified Becke 88 for proton transfer
References:
V. Tognetti and C. Adamo, J. Phys. Chem. A 113, 14415 (2009) (10.1021/jp903672e)
- **GGA_MOHLYP2_XC**
Functional for barrier heights
References:
J. Zheng, Y. Zhao, and D. G. Truhlar, J. Chem. Theory Comput. 5, 808 (2009) (10.1021/ct800568m)
- **GGA_MOHLYP_XC**
Functional for organometallic chemistry
References:
N. E. Schultz, Y. Zhao, and D. G. Truhlar, J. Phys. Chem. A 109, 11127 (2005) (10.1021/jp0539223)
- **GGA_MPBE_X**
Adamo & Barone modification to PBE
References:
C. Adamo and V. Barone, J. Chem. Phys. 116, 5933 (2002) (10.1063/1.1458927)

- **GGA_MPW91_X**
mPW91 of Adamo & Barone
References:
C. Adamo and V. Barone, *J. Chem. Phys.* 108, 664 (1998) (10.1063/1.475428)
- **GGA_MPWLYP1W_XC**
mPWLYP1w
References:
E. E. Dahlke and D. G. Truhlar, *J. Phys. Chem. B* 109, 15677 (2005) (10.1021/jp052436c)
- **GGA_N12_C**
Minnesota N12 functional
References:
R. Peverati and D. G. Truhlar, *J. Chem. Theory Comput.* 8, 2310 (2012) (10.1021/ct3002656)
- **GGA_N12_SX_C**
Minnesota N12-SX functional
References:
R. Peverati and D. G. Truhlar, *Phys. Chem. Chem. Phys.* 14, 16187 (2012) (10.1039/C2CP42576A)
- **GGA_N12_X**
Minnesota N12 exchange functional
References:
R. Peverati and D. G. Truhlar, *J. Chem. Theory Comput.* 8, 2310 (2012) (10.1021/ct3002656)
- **GGA_OBLYP_D_XC**
oBLYP-D functional of Goerigk and Grimme
References:
L. Goerigk and S. Grimme, *J. Chem. Theory Comput.* 6, 107 (2010) (10.1021/ct900489g)
- **GGA_OL2_X**
Exchange form based on Ou-Yang and Levy v.2
References:
P. Fuentealba and O. Reyes, *Chem. Phys. Lett.* 232, 31 (1995) (10.1016/0009-2614(94)01321-L)
H. Ou-Yang and M. Levy, *Int. J. Quantum Chem.* 40, 379 (1991) (10.1002/qua.560400309)
- **GGA_OP_B88_C**
one-parameter progressive functional (B88 version)
References:
T. Tsuneda, T. Suzumura, and K. Hirao, *J. Chem. Phys.* 110, 10664 (1999) (10.1063/1.479012)
- **GGA_OP_G96_C**
one-parameter progressive functional (G96 version)
References:
T. Tsuneda, T. Suzumura, and K. Hirao, *J. Chem. Phys.* 110, 10664 (1999) (10.1063/1.479012)
T. Tsuneda, T. Suzumura, and K. Hirao, *J. Chem. Phys.* 111, 5656 (1999) (10.1063/1.479954)
- **GGA_OP_PBE_C**
one-parameter progressive functional (PBE version)
References:
T. Tsuneda, T. Suzumura, and K. Hirao, *J. Chem. Phys.* 110, 10664 (1999) (10.1063/1.479012)
T. Tsuneda, T. Suzumura, and K. Hirao, *J. Chem. Phys.* 111, 5656 (1999) (10.1063/1.479954)
- **GGA_OP_PW91_C**
one-parameter progressive functional (PW91 version)
References:

- T. Tsuneda, T. Suzumura, and K. Hirao, J. Chem. Phys. 110, 10664 (1999) (10.1063/1.479012)
T. Tsuneda, T. Suzumura, and K. Hirao, J. Chem. Phys. 111, 5656 (1999) (10.1063/1.479954)
- **GGA_OP_XALPHA_C**
one-parameter progressive functional (Xalpha version)
References:
T. Tsuneda, T. Suzumura, and K. Hirao, J. Chem. Phys. 110, 10664 (1999) (10.1063/1.479012)
T. Tsuneda, T. Suzumura, and K. Hirao, J. Chem. Phys. 111, 5656 (1999) (10.1063/1.479954)
 - **GGA_OPBE_D_XC**
oPBE-D functional of Goerigk and Grimme
References:
L. Goerigk and S. Grimme, J. Chem. Theory Comput. 6, 107 (2010) (10.1021/ct900489g)
 - **GGA_OPTB88_VDW_X**
opt-Becke 88 for vdW
References:
J. Klime{s}, D. R. Bowler, and A. Michaelides, J. Phys.: Condens. Matter 22, 022201 (2010) (10.1088/0953-8984/22/2/022201)
 - **GGA_OPTC_C**
Optimized correlation functional of Cohen and Handy
References:
A. J. Cohen and N. C. Handy, Mol. Phys. 99, 607 (2001) (10.1080/00268970010023435)
 - **GGA_OPTPBE_VDW_X**
Reparametrized PBE for vdW
References:
J. Klime{s}, D. R. Bowler, and A. Michaelides, J. Phys.: Condens. Matter 22, 022201 (2010) (10.1088/0953-8984/22/2/022201)
 - **GGA_OPTX_X**
Handy & Cohen OPTX 01
References:
N. C. Handy and A. J. Cohen, Mol. Phys. 99, 403 (2001) (10.1080/00268970010018431)
 - **GGA_OPWLYP_D_XC**
oPWLYP-D functional of Goerigk and Grimme
References:
L. Goerigk and S. Grimme, J. Chem. Theory Comput. 6, 107 (2010) (10.1021/ct900489g)
 - **GGA_P86_C**
Perdew 86
References:
J. P. Perdew, Phys. Rev. B 33, 8822 (1986) (10.1103/PhysRevB.33.8822)
 - **GGA_PBE1W_XC**
PBE1W
References:
E. E. Dahlke and D. G. Truhlar, J. Phys. Chem. B 109, 15677 (2005) (10.1021/jp052436c)
 - **GGA_PBE_C**
Perdew, Burke & Ernzerhof
References:
J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. 77, 3865 (1996) (10.1103/PhysRevLett.77.3865)
J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. 78, 1396 (1997) (10.1103/PhysRevLett.78.1396)

- **GGA_PBE_JRGX_C**
Reparametrized PBE by Pedroza, Silva & Capelle
References:
L. S. Pedroza, A. J. R. da Silva, and K. Capelle, Phys. Rev. B 79, 201106 (2009) (10.1103/PhysRevB.79.201106)
- **GGA_PBE_JSJR_X**
Reparametrized PBE by Pedroza, Silva & Capelle
References:
L. S. Pedroza, A. J. R. da Silva, and K. Capelle, Phys. Rev. B 79, 201106 (2009) (10.1103/PhysRevB.79.201106)
- **GGA_PBE_MOL_C**
Reparametrized PBE by del Campo, Gazquez, Trickey & Vela
References:
J. M. del Campo, J. L. Gázquez, S. B. Trickey, and A. Vela, J. Chem. Phys. 136, 104108 (2012) (10.1063/1.3691197)
- **GGA_PBE_MOL_X**
Reparametrized PBE by del Campo, Gazquez, Trickey & Vela
References:
J. M. del Campo, J. L. Gázquez, S. B. Trickey, and A. Vela, J. Chem. Phys. 136, 104108 (2012) (10.1063/1.3691197)
- **GGA_PBE_R_X**
Revised PBE from Zhang & Yang
References:
Y. Zhang and W. Yang, Phys. Rev. Lett. 80, 890 (1998) (10.1103/PhysRevLett.80.890)
- **GGA_PBE_SOL_C**
Perdew, Burke & Ernzerhof SOL
References:
J. P. Perdew, A. Ruzsinszky, G. I. Csonka, O. A. Vydrov, G. E. Scuseria, L. A. Constantin, X. Zhou, and K. Burke, Phys. Rev. Lett. 100, 136406 (2008) (10.1103/PhysRevLett.100.136406)
- **GGA_PBE_SOL_X**
Perdew, Burke & Ernzerhof SOL
References:
J. P. Perdew, A. Ruzsinszky, G. I. Csonka, O. A. Vydrov, G. E. Scuseria, L. A. Constantin, X. Zhou, and K. Burke, Phys. Rev. Lett. 100, 136406 (2008) (10.1103/PhysRevLett.100.136406)
- **GGA_PBE_TCA_X**
PBE revised by Tognetti et al
References:
V. Tognetti, P. Cortona, and C. Adamo, Chem. Phys. Lett. 460, 536 (2008) (10.1016/j.cplett.2008.06.032)
- **GGA_PBE_X**
Perdew, Burke & Ernzerhof
References:
J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. 77, 3865 (1996) (10.1103/PhysRevLett.77.3865)
J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. 78, 1396 (1997) (10.1103/PhysRevLett.78.1396)
- **GGA_PBE_XC**

Functional components: GGA_PBE_X + GGA_PBE_C

References:

J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. 77, 3865 (1996) (10.1103/PhysRevLett.77.3865)

J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. 78, 1396 (1997) (10.1103/PhysRevLett.78.1396)

J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. 77, 3865 (1996) (10.1103/PhysRevLett.77.3865)

J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. 78, 1396 (1997) (10.1103/PhysRevLett.78.1396)

- **GGA_PBEA_X**

Madsen 07

References:

G. K. H. Madsen, Phys. Rev. B 75, 195108 (2007) (10.1103/PhysRevB.75.195108)

- **GGA_PBEFE_C**

PBE for formation energies

References:

R. Sarmiento-Pérez, S. Botti, and M. A. L. Marques, J. Chem. Theory Comput. 11, 3844 (2015) (10.1021/acs.jctc.5b00529)

- **GGA_PBEFE_X**

PBE for formation energies

References:

R. Sarmiento-Pérez, S. Botti, and M. A. L. Marques, J. Chem. Theory Comput. 11, 3844 (2015) (10.1021/acs.jctc.5b00529)

- **GGA_PBEINT_C**

PBE for hybrid interfaces

References:

E. Fabiano, L. A. Constantin, and F. Della Sala, Phys. Rev. B 82, 113104 (2010) (10.1103/PhysRevB.82.113104)

- **GGA_PBEINT_X**

PBE for hybrid interfaces

References:

E. Fabiano, L. A. Constantin, and F. Della Sala, Phys. Rev. B 82, 113104 (2010) (10.1103/PhysRevB.82.113104)

- **GGA_PBEK1_VDW_X**

Reparametrized PBE for vdW

References:

J. Klimeš, D. R. Bowler, and A. Michaelides, J. Phys.: Condens. Matter 22, 022201 (2010) (10.1088/0953-8984/22/2/022201)

- **GGA_PBELOC_C**

Semilocal dynamical correlation

References:

L. A. Constantin, E. Fabiano, and F. Della Sala, Phys. Rev. B 86, 035130 (2012) (10.1103/PhysRevB.86.035130)

- **GGA_PBELYP1W_XC**

PBELYP1W

References:

E. E. Dahlke and D. G. Truhlar, J. Phys. Chem. B 109, 15677 (2005) (10.1021/jp052436c)

- **GGA_PBEPOW_X**
PBE power
References:
Éric Brémond, J. Chem. Phys. 145, 244102 (2016) (10.1063/1.4972815)
- **GGA_PBETRANS_X**
Gradient-regulated connection-based correction for the PBE exchange
References:
Éric Brémond, I. Ciofini, and C. Adamo, Mol. Phys. 114, 1059 (2016) (10.1080/00268976.2015.1132788)
- **GGA_PW86_X**
Perdew & Wang 86
References:
J. P. Perdew and W. Yue, Phys. Rev. B 33, 8800 (1986) (10.1103/PhysRevB.33.8800)
- **GGA_PW91_C**
Perdew & Wang 91
References:
J. P. Perdew, in Proceedings of the 75. WE-Heraeus-Seminar and 21st Annual International Symposium on Electronic Structure of Solids, edited by P. Ziesche and H. Eschrig (Akademie Verlag, Berlin, 1991) p. 11
J. P. Perdew, J. A. Chevary, S. H. Vosko, K. A. Jackson, M. R. Pederson, D. J. Singh, and C. Fiolhais, Phys. Rev. B 46, 6671 (1992) (10.1103/PhysRevB.46.6671)
J. P. Perdew, J. A. Chevary, S. H. Vosko, K. A. Jackson, M. R. Pederson, D. J. Singh, and C. Fiolhais, Phys. Rev. B 48, 4978 (1993) (10.1103/PhysRevB.48.4978.2)
- **GGA_PW91_X**
Perdew & Wang 91
References:
J. P. Perdew, in Proceedings of the 75. WE-Heraeus-Seminar and 21st Annual International Symposium on Electronic Structure of Solids, edited by P. Ziesche and H. Eschrig (Akademie Verlag, Berlin, 1991) p. 11
J. P. Perdew, J. A. Chevary, S. H. Vosko, K. A. Jackson, M. R. Pederson, D. J. Singh, and C. Fiolhais, Phys. Rev. B 46, 6671 (1992) (10.1103/PhysRevB.46.6671)
J. P. Perdew, J. A. Chevary, S. H. Vosko, K. A. Jackson, M. R. Pederson, D. J. Singh, and C. Fiolhais, Phys. Rev. B 48, 4978 (1993) (10.1103/PhysRevB.48.4978.2)
- **GGA_Q2D_C**
Chiodo et al
References:
L. Chiodo, L. A. Constantin, E. Fabiano, and F. Della Sala, Phys. Rev. Lett. 108, 126402 (2012) (10.1103/PhysRevLett.108.126402)
- **GGA_Q2D_X**
Chiodo et al
References:
L. Chiodo, L. A. Constantin, E. Fabiano, and F. Della Sala, Phys. Rev. Lett. 108, 126402 (2012) (10.1103/PhysRevLett.108.126402)
- **GGA_REGTPSS_C**
regularized TPSS correlation
References:
J. P. Perdew, A. Ruzsinszky, G. I. Csonka, L. A. Constantin, and J. Sun, Phys. Rev. Lett. 103, 026403 (2009) (10.1103/PhysRevLett.103.026403)

- **GGA_REVTCA_C**
 Tognetti, Cortona, Adamo (revised)
 References:
 V. Tognetti, P. Cortona, and C. Adamo, Chem. Phys. Lett. 460, 536 (2008) (10.1016/j.cplett.2008.06.032)
- **GGA_RGE2_C**
 Regularized PBE
 References:
 A. Ruzsinszky, G. I. Csonka, and G. E. Scuseria, J. Chem. Theory Comput. 5, 763 (2009) (10.1021/ct8005369)
- **GGA_RGE2_X**
 Regularized PBE
 References:
 A. Ruzsinszky, G. I. Csonka, and G. E. Scuseria, J. Chem. Theory Comput. 5, 763 (2009) (10.1021/ct8005369)
- **GGA_RPBE_X**
 Hammer, Hansen, and Norskov
 References:
 B. Hammer, L. B. Hansen, and J. K. Nørskov, Phys. Rev. B 59, 7413 (1999) (10.1103/PhysRevB.59.7413)
- **GGA_RPW86_X**
 Refitted Perdew & Wang 86
 References:
 E. D. Murray, K. Lee, and D. C. Langreth, J. Chem. Theory Comput. 5, 2754 (2009) (10.1021/ct900365q)
- **GGA_SCAN_E0_C**
 GGA component of SCAN
 References:
 J. Sun, A. Ruzsinszky, and J. P. Perdew, Phys. Rev. Lett. 115, 036402 (2015) (10.1103/PhysRevLett.115.036402)
- **GGA_SFAT_X**
 Short-range recipe for exchange GGA functionals - Yukawa
 References:
 A. Savin and H.-J. Flad, Int. J. Quantum Chem. 56, 327 (1995) (10.1002/qua.560560417)
 Y. Akinaga and S. Ten-no, Chem. Phys. Lett. 462, 348 (2008) (10.1016/j.cplett.2008.07.103)
- **GGA_SG4_C**
 Semiclassical GGA at fourth order
 References:
 L. A. Constantin, A. Terentjevs, F. Della Sala, P. Cortona, and E. Fabiano, Phys. Rev. B 93, 045126 (2016) (10.1103/PhysRevB.93.045126)
- **GGA_SG4_X**
 Semiclassical GGA at fourth order
 References:
 L. A. Constantin, A. Terentjevs, F. Della Sala, P. Cortona, and E. Fabiano, Phys. Rev. B 93, 045126 (2016) (10.1103/PhysRevB.93.045126)
- **GGA_SOGGA11_C**
 Second-order generalized gradient approximation 2011
 References:

- R. Peverati, Y. Zhao, and D. G. Truhlar, *J. Phys. Chem. Lett.* **2**, 1991 (2011) (10.1021/jz200616w)
- **GGA_SOGGA11_X**
Second-order generalized gradient approximation 2011
References:
R. Peverati, Y. Zhao, and D. G. Truhlar, *J. Phys. Chem. Lett.* **2**, 1991 (2011) (10.1021/jz200616w)
 - **GGA_SOGGA11_X_C**
To be used with HYB_GGA_X_SOGGA11_X
References:
R. Peverati and D. G. Truhlar, *J. Chem. Phys.* **135**, 191102 (2011) (10.1063/1.3663871)
 - **GGA_SOGGA_X**
Second-order generalized gradient approximation
References:
Y. Zhao and D. G. Truhlar, *J. Chem. Phys.* **128**, 184109 (2008) (10.1063/1.2912068)
 - **GGA_SPBE_C**
PBE correlation to be used with the SSB exchange
References:
M. Swart, M. Solá, and F. M. Bickelhaupt, *J. Chem. Phys.* **131**, 094103 (2009) (10.1063/1.3213193)
 - **GGA_SSB_D_X**
Swart, Sola and Bickelhaupt dispersion
References:
M. Swart, M. Solá, and F. M. Bickelhaupt, *J. Chem. Phys.* **131**, 094103 (2009) (10.1063/1.3213193)
 - **GGA_SSB_SW_X**
Swart, Sola and Bickelhaupt correction to PBE
References:
M. Swart, M. Solá, and F. M. Bickelhaupt, *J. Comput. Methods Sci. Eng.* **9**, 69 (2009) (10.3233/JCM-2009-0230)
 - **GGA_SSB_X**
Swart, Sola and Bickelhaupt
References:
M. Swart, M. Solá, and F. M. Bickelhaupt, *J. Chem. Phys.* **131**, 094103 (2009) (10.1063/1.3213193)
 - **GGA_TAU_HCTH_C**
correlation part of tau-hcth
References:
A. D. Boese and N. C. Handy, *J. Chem. Phys.* **116**, 9559 (2002) (10.1063/1.1476309)
 - **GGA_TCA_C**
Tognetti, Cortona, Adamo
References:
V. Tognetti, P. Cortona, and C. Adamo, *J. Chem. Phys.* **128**, 034101 (2008) (10.1063/1.2816137)
 - **GGA_TH1_XC**
Tozer and Handy v. 1
References:
D. J. Tozer and N. C. Handy, *J. Chem. Phys.* **108**, 2545 (1998) (10.1063/1.475638)

- **GGA_TH2_XC**
Tozer and Handy v. 2
References:
D. J. Tozer and N. C. Handy, J. Phys. Chem. A 102, 3162 (1998) (10.1021/jp980259s)
- **GGA_TH3_XC**
Tozer and Handy v. 3
References:
N. C. Handy and D. J. Tozer, Mol. Phys. 94, 707 (1998) (10.1080/002689798167863)
- **GGA_TH4_XC**
Tozer and Handy v. 4
References:
N. C. Handy and D. J. Tozer, Mol. Phys. 94, 707 (1998) (10.1080/002689798167863)
- **GGA_TH_FC_XC**
Tozer and Handy v. FC
References:
D. J. Tozer, N. C. Handy, and W. H. Green, Chem. Phys. Lett. 273, 183 (1997) (10.1016/S0009-2614(97)00586-1)
- **GGA_TH_FCFO_XC**
Tozer and Handy v. FCFO
References:
D. J. Tozer, N. C. Handy, and W. H. Green, Chem. Phys. Lett. 273, 183 (1997) (10.1016/S0009-2614(97)00586-1)
- **GGA_TH_FCO_XC**
Tozer and Handy v. FCO
References:
D. J. Tozer, N. C. Handy, and W. H. Green, Chem. Phys. Lett. 273, 183 (1997) (10.1016/S0009-2614(97)00586-1)
- **GGA_TH_FL_XC**
Tozer and Handy v. FL
References:
D. J. Tozer, N. C. Handy, and W. H. Green, Chem. Phys. Lett. 273, 183 (1997) (10.1016/S0009-2614(97)00586-1)
- **GGA_TM_LYP_C**
Takkar and McCarthy reparametrization
References:
A. J. Thakkar and S. P. McCarthy, J. Chem. Phys. 131, 134109 (2009) (10.1063/1.3243845)
- **GGA_TM_PBE_C**
Thakkar and McCarthy reparametrization
References:
A. J. Thakkar and S. P. McCarthy, J. Chem. Phys. 131, 134109 (2009) (10.1063/1.3243845)
- **GGA_VMT84_GE_X**
VMT{8,4} with constraint satisfaction with $\mu = \mu_{\text{GE}}$
References:
A. Vela, J. C. Pacheco-Kato, J. L. Gázquez, J. M. del Campo, and S. B. Trickey, J. Chem. Phys. 136, 144115 (2012) (10.1063/1.3701132)
- **GGA_VMT84_PBE_X**
VMT{8,4} with constraint satisfaction with $\mu = \mu_{\text{PBE}}$

References:

A. Vela, J. C. Pacheco-Kato, J. L. Gázquez, J. M. del Campo, and S. B. Trickey, J. Chem. Phys. 136, 144115 (2012) (10.1063/1.3701132)

- **GGA_VMT_GE_X**

Vela, Medel, and Trickey with $\mu = \mu_{\text{GE}}$

References:

A. Vela, V. Medel, and S. B. Trickey, J. Chem. Phys. 130, 244103 (2009) (10.1063/1.3152713)

- **GGA_VMT_PBE_X**

Vela, Medel, and Trickey with $\mu = \mu_{\text{PBE}}$

References:

A. Vela, V. Medel, and S. B. Trickey, J. Chem. Phys. 130, 244103 (2009) (10.1063/1.3152713)

- **GGA_VV10_XC**

Vydrov and Van Voorhis

References:

O. A. Vydrov and T. Van Voorhis, J. Chem. Phys. 133, 244103 (2010) (10.1063/1.3521275)

- **GGA_W94_C**

Wilson 94 (Eq. 25)

References:

L. C. Wilson, Chemical Physics 181, 337 (1994) (10.1016/0301-0104(93)E0444-Z)

- **GGA_WC_X**

Wu & Cohen

References:

Z. Wu and R. E. Cohen, Phys. Rev. B 73, 235116 (2006) (10.1103/PhysRevB.73.235116)

- **GGA_WI0_C**

Wilson & Ivanov initial version

References:

L. C. Wilson and S. Ivanov, Int. J. Quantum Chem. 69, 523 (1998) (10.1002/(SICI)1097-461X(1998)69:4<523::AID-QUA9>3.0.CO;2-X)

- **GGA_WI_C**

Wilson & Ivanov

References:

L. C. Wilson and S. Ivanov, Int. J. Quantum Chem. 69, 523 (1998) (10.1002/(SICI)1097-461X(1998)69:4<523::AID-QUA9>3.0.CO;2-X)

- **GGA_WL_C**

Wilson & Levy

References:

L. C. Wilson and M. Levy, Phys. Rev. B 41, 12930 (1990) (10.1103/PhysRevB.41.12930)

- **GGA_WPBEH_X**

short-range part of the PBE (default $w=0$ gives PBEh)

References:

J. Heyd, G. E. Scuseria, and M. Ernzerhof, J. Chem. Phys. 118, 8207 (2003) (10.1063/1.1564060)

J. Heyd, G. E. Scuseria, and M. Ernzerhof, J. Chem. Phys. 124, 219906 (2006) (10.1063/1.2204597)

M. Ernzerhof and J. P. Perdew, J. Chem. Phys. 109, 3313 (1998) (10.1063/1.476928)

J. Heyd and G. E. Scuseria, J. Chem. Phys. 120, 7274 (2004) (10.1063/1.1668634)

T. M. Henderson, A. F. Izmaylov, G. Scalmani, and G. E. Scuseria, J. Chem. Phys. 131, 044108 (2009) (10.1063/1.3185673)

- **GGA_XLYP_XC**
XLYP
References:
X. Xu and W. A. Goddard, Proc. Natl. Acad. Sci. U. S. A. 101, 2673 (2004) (10.1073/pnas.0308730100)

- **GGA_XPBE_C**
Extended PBE by Xu & Goddard III
References:
X. Xu and W. A. Goddard, J. Chem. Phys. 121, 4068 (2004) (10.1063/1.1771632)

- **GGA_XPBE_X**
Extended PBE by Xu & Goddard III
References:
X. Xu and W. A. Goddard, J. Chem. Phys. 121, 4068 (2004) (10.1063/1.1771632)

- **GGA_ZPBEINT_C**
spin-dependent gradient correction to PBEint
References:
L. A. Constantin, E. Fabiano, and F. Della Sala, Phys. Rev. B 84, 233103 (2011) (10.1103/PhysRevB.84.233103)

- **GGA_ZPBESOL_C**
spin-dependent gradient correction to PBEsol
References:
L. A. Constantin, E. Fabiano, and F. Della Sala, Phys. Rev. B 84, 233103 (2011) (10.1103/PhysRevB.84.233103)

- **GGA_ZVPBEINT_C**
another spin-dependent correction to PBEint
References:
L. A. Constantin, E. Fabiano, and F. D. Sala, J. Chem. Phys. 137, 194105 (2012) (10.1063/1.4766324)

- **GGA_ZVPBESOL_C**
another spin-dependent correction to PBEsol
References:
L. A. Constantin, E. Fabiano, and F. D. Sala, J. Chem. Phys. 137, 194105 (2012) (10.1063/1.4766324)

- **HGGA_B1LYP_XC**
B1LYP
Functional components: 0.25*HF_Exchange
References:
C. Adamo and V. Barone, Chem. Phys. Lett. 274, 242 (1997) (10.1016/S0009-2614(97)00651-9)

- **HGGA_B1PW91_XC**
B1PW91
Functional components: 0.25*HF_Exchange
References:
C. Adamo and V. Barone, Chem. Phys. Lett. 274, 242 (1997) (10.1016/S0009-2614(97)00651-9)

- **HGGA_B1WC_XC**
B1WC
Functional components: 0.16*HF_Exchange
References:
D. I. Bilc, R. Orlando, R. Shaltaf, G.-M. Rignanese, J. Íñiguez, and P. Ghosez, Phys. Rev. B 77, 165107 (2008) (10.1103/PhysRevB.77.165107)

- **HGGA_B3LYP5_XC**
 B3LYP with VWN functional 5 instead of RPA
 Functional components: 0.2*HF_Exchange
 References:
 P. J. Stephens, F. J. Devlin, C. F. Chabalowski, and M. J. Frisch, J. Phys. Chem. 98, 11623 (1994) (10.1021/j100096a001)

- **HGGA_B3LYP_LXC_XC**
 B3LYP
 Functional components: 0.2*HF_Exchange
 References:
 P. J. Stephens, F. J. Devlin, C. F. Chabalowski, and M. J. Frisch, J. Phys. Chem. 98, 11623 (1994) (10.1021/j100096a001)

- **HGGA_B3LYP_XC**
 B3LYP
 Functional components: 0.2*HF_Exchange + 0.08*LDA_SLATER_X + 0.19*LDA_VWN1RPA_C + 0.72*GGA_B88_X + 0.81*GGA_LYP_C
 References:
 P. J. Stephens, F. J. Devlin, C. F. Chabalowski, and M. J. Frisch, J. Phys. Chem. 98, 11623 (1994) (10.1021/j100096a001)

- **HGGA_B3LYPS_XC**
 B3LYP*
 Functional components: 0.15*HF_Exchange
 References:
 M. Reiher, O. Salomon, and B. A. Hess, Theor. Chem. Acc. 107, 48 (2001) (10.1007/s00214-001-0300-3)

- **HGGA_B3P86_XC**
 B3P86
 Functional components: 0.2*HF_Exchange
 References:
 Defined through Gaussian implementation.

- **HGGA_B3PW91_XC**
 The original (ACM, B3PW91) hybrid of Becke
 Functional components: 0.2*HF_Exchange
 References:
 A. D. Becke, J. Chem. Phys. 98, 5648 (1993) (10.1063/1.464913)

- **HGGA_B5050LYP_XC**
 B5050LYP
 Functional components: 0.5*HF_Exchange
 References:
 Y. Shao, M. Head-Gordon, and A. I. Krylov, J. Chem. Phys. 118, 4807 (2003) (10.1063/1.1545679)

- **HGGA_B97_1_XC**
 Becke 97-1
 Functional components: 0.21*HF_Exchange
 References:
 F. A. Hamprecht, A. J. Cohen, D. J. Tozer, and N. C. Handy, J. Chem. Phys. 109, 6264 (1998) (10.1063/1.477267)

- **HGGA_B97_1P_XC**
 version of B97 by Cohen and Handy

- Functional components: 0.15*HF_Exchange
References:
A. J. Cohen and N. C. Handy, Chem. Phys. Lett. 316, 160 (2000) (10.1016/S0009-2614(99)01273-7)
- **HGGA_B97_2_XC**
Becke 97-2
Functional components: 0.21*HF_Exchange
References:
P. J. Wilson, T. J. Bradley, and D. J. Tozer, J. Chem. Phys. 115, 9233 (2001) (10.1063/1.1412605)
 - **HGGA_B97_3_XC**
Becke 97-3
Functional components: 0.269288*HF_Exchange
References:
T. W. Keal and D. J. Tozer, J. Chem. Phys. 123, 121103 (2005) (10.1063/1.2061227)
 - **HGGA_B97_K_XC**
Boese-Martin for Kinetics
Functional components: 0.42*HF_Exchange
References:
A. D. Boese and J. M. L. Martin, J. Chem. Phys. 121, 3405 (2004) (10.1063/1.1774975)
 - **HGGA_B97_XC**
Becke 97
Functional components: 0.1943*HF_Exchange
References:
A. D. Becke, J. Chem. Phys. 107, 8554 (1997) (10.1063/1.475007)
 - **HGGA_BHANDH_XC**
BHandH
Functional components: 0.5*HF_Exchange
References:
A. D. Becke, J. Chem. Phys. 98, 1372 (1993) (10.1063/1.464304)
Defined through Gaussian implementation.
 - **HGGA_BHANDHLYP_XC**
BHandHLYP
Functional components: 0.5*HF_Exchange
References:
A. D. Becke, J. Chem. Phys. 98, 1372 (1993) (10.1063/1.464304)
Defined through Gaussian implementation.
 - **HGGA_CAM_B3LYP_XC**
CAM version of B3LYP
Functional components: 0.19*HF_Exchange_ShortRange + 0.65*HF_Exchange_LongRange
References:
T. Yanai, D. P. Tew, and N. C. Handy, Chem. Phys. Lett. 393, 51 (2004) (10.1016/j.cplett.2004.06.011)
 - **HGGA_CAM_QTP_01_XC**
CAM-B3LYP retuned using ionization potentials of water
Functional components: 0.23*HF_Exchange_ShortRange + HF_Exchange_LongRange
References:
Y. Jin and R. J. Bartlett, J. Chem. Phys. 145, 034107 (2016), <http://dx.doi.org/10.1063/1.4955497>
(10.1063/1.4955497)

- **HGGA_CAMY_B3LYP_XC**
 CAMY version of B3LYP
 Functional components: $0.19*HF_Exchange_ShortRange + 0.65*HF_Exchange_LongRange$
 References:
 M. Seth and T. Ziegler, *J. Chem. Theory Comput.* 8, 901 (2012) (10.1021/ct300006h)
- **HGGA_CAMY_BLYP_XC**
 CAMY version of BLYP
 Functional components: $0.2*HF_Exchange_ShortRange + HF_Exchange_LongRange$
 References:
 Y. Akinaga and S. Ten-no, *Chem. Phys. Lett.* 462, 348 (2008) (10.1016/j.cplett.2008.07.103)
- **HGGA_CAP0_XC**
 Correct Asymptotic Potential hybrid
 Functional components: $0.25*HF_Exchange$
 References:
 J. Carmona-Espíndola, J. L. Gázquez, A. Vela, and S. B. Trickey, *Theor. Chem. Acc.* 135, 120 (2016) (10.1007/s00214-016-1864-2)
- **HGGA_EDF2_XC**
 EDF2
 Functional components: $0.1695*HF_Exchange$
 References:
 C. Y. Lin, M. W. George, and P. M. W. Gill, *Australian Journal of Chemistry* 57, 365 (2004) (10.1071/CH03263)
- **HGGA_HJS_B88_XC**
 HJS hybrid screened exchange B88 version
 Functional components: $0.25*HF_Exchange_ShortRange + 0*HF_Exchange_LongRange$
 References:
 T. M. Henderson, B. G. Janesko, and G. E. Scuseria, *J. Chem. Phys.* 128, 194105 (2008) (10.1063/1.2921797)
- **HGGA_HJS_B97X_XC**
 HJS hybrid screened exchange B97x version
 Functional components: $0.25*HF_Exchange_ShortRange + 0*HF_Exchange_LongRange$
 References:
 T. M. Henderson, B. G. Janesko, and G. E. Scuseria, *J. Chem. Phys.* 128, 194105 (2008) (10.1063/1.2921797)
- **HGGA_HJS_PBE_SOL_XC**
 HJS hybrid screened exchange PBE_SOL version
 Functional components: $0.25*HF_Exchange_ShortRange + 0*HF_Exchange_LongRange$
 References:
 T. M. Henderson, B. G. Janesko, and G. E. Scuseria, *J. Chem. Phys.* 128, 194105 (2008) (10.1063/1.2921797)
- **HGGA_HJS_PBE_XC**
 HJS hybrid screened exchange PBE version
 Functional components: $0.25*HF_Exchange_ShortRange + 0*HF_Exchange_LongRange$
 References:
 T. M. Henderson, B. G. Janesko, and G. E. Scuseria, *J. Chem. Phys.* 128, 194105 (2008) (10.1063/1.2921797)
- **HGGA_HSE03_XC**
 HSE03
 Functional components: $0.25*HF_Exchange_ShortRange + 0*HF_Exchange_LongRange$

References:

J. Heyd, G. E. Scuseria, and M. Ernzerhof, J. Chem. Phys. 118, 8207 (2003) (10.1063/1.1564060)

J. Heyd, G. E. Scuseria, and M. Ernzerhof, J. Chem. Phys. 124, 219906 (2006) (10.1063/1.2204597)

- **HGGA_HSE06_XC**

HSE06

Functional components: $0.25*HF_Exchange_ShortRange + 0*HF_Exchange_LongRange$

References:

J. Heyd, G. E. Scuseria, and M. Ernzerhof, J. Chem. Phys. 118, 8207 (2003) (10.1063/1.1564060)

J. Heyd, G. E. Scuseria, and M. Ernzerhof, J. Chem. Phys. 124, 219906 (2006) (10.1063/1.2204597)

A. V. Krukau, O. A. Vydrov, A. F. Izmaylov, and G. E. Scuseria, J. Chem. Phys. 125, 224106 (2006) (10.1063/1.2404663)

- **HGGA_HSE12_XC**

HSE12

Functional components: $0.313*HF_Exchange_ShortRange + 0*HF_Exchange_LongRange$

References:

J. E. Moussa, P. A. Schultz, and J. R. Chelikowsky, J. Chem. Phys. 136, 204117 (2012)

(10.1063/1.4722993)

- **HGGA_HSE12S_XC**

HSE12 (short-range version)

Functional components: $0.425*HF_Exchange_ShortRange + 0*HF_Exchange_LongRange$

References:

J. E. Moussa, P. A. Schultz, and J. R. Chelikowsky, J. Chem. Phys. 136, 204117 (2012)

(10.1063/1.4722993)

- **HGGA_HSE_SOL_XC**

HSEsol

Functional components: $0.25*HF_Exchange_ShortRange + 0*HF_Exchange_LongRange$

References:

L. Schimka, J. Harl, and G. Kresse, J. Chem. Phys. 134, 024116 (2011) (10.1063/1.3524336)

- **HGGA_KMLYP_XC**

Kang-Musgrave hybrid

Functional components: $0.557*HF_Exchange$

References:

J. K. Kang and C. B. Musgrave, J. Chem. Phys. 115, 11040 (2001), <http://dx.doi.org/10.1063/1.1415079>

(10.1063/1.1415079)

- **HGGA_LC_VV10_XC**

Vydrov and Van Voorhis

Functional components: $0*HF_Exchange_ShortRange + HF_Exchange_LongRange$

References:

O. A. Vydrov and T. Van Voorhis, J. Chem. Phys. 133, 244103 (2010) (10.1063/1.3521275)

- **HGGA_LC_WPBE_XC**

Long-range corrected PBE (LC-wPBE) by Vydrov and Scuseria

Functional components: $0*HF_Exchange_ShortRange + HF_Exchange_LongRange$

References:

O. A. Vydrov and G. E. Scuseria, J. Chem. Phys. 125, 234109 (2006), 10.1063/1.2409292

(10.1063/1.2409292)

- **HGGA_LCY_BLYP_XC**

LCY version of BLYP

Functional components: $0*HF_Exchange_ShortRange + HF_Exchange_LongRange$

References:

Y. Akinaga and S. Ten-no, Chem. Phys. Lett. 462, 348 (2008) (10.1016/j.cplett.2008.07.103)
M. Seth, T. Ziegler, M. Steinmetz, and S. Grimme, J. Chem. Theory Comput. 9, 2286 (2013)
(10.1021/ct301112m)

- **HGGA_LCY_PBE_XC**

LCY version of PBE

Functional components: $0*HF_Exchange_ShortRange + HF_Exchange_LongRange$

References:

M. Seth and T. Ziegler, J. Chem. Theory Comput. 8, 901 (2012) (10.1021/ct300006h)
M. Seth, T. Ziegler, M. Steinmetz, and S. Grimme, J. Chem. Theory Comput. 9, 2286 (2013)
(10.1021/ct301112m)

- **HGGA_LRC_WPBE_XC**

Long-range corrected PBE (LRC-wPBE) by Rohrdanz, Martins and Herbert

Functional components: $0*HF_Exchange_ShortRange + HF_Exchange_LongRange$

References:

M. A. Rohrdanz, K. M. Martins, and J. M. Herbert, J. Chem. Phys. 130, 054112 (2009)
(10.1063/1.3073302)

- **HGGA_LRC_WPBEh_XC**

Long-range corrected short-range hybrid PBE (LRC-wPBEh) by Rohrdanz, Martins and Herbert

Functional components: $0.2*HF_Exchange_ShortRange + HF_Exchange_LongRange$

References:

M. A. Rohrdanz, K. M. Martins, and J. M. Herbert, J. Chem. Phys. 130, 054112 (2009)
(10.1063/1.3073302)

- **HGGA_MB3LYP_RC04_XC**

B3LYP with RC04 LDA

Functional components: $0.2*HF_Exchange$

References:

V. Tognetti, P. Cortona, and C. Adamo, Chem. Phys. Lett. 439, 381 (2007) (10.1016/j.cplett.2007.03.081)

- **HGGA_MPW1K_XC**

mPW1K

Functional components: $0.428*HF_Exchange$

References:

B. J. Lynch, P. L. Fast, M. Harris, and D. G. Truhlar, J. Phys. Chem. A 104, 4811 (2000)
(10.1021/jp000497z)

- **HGGA_MPW1LYP_XC**

mPW1LYP

Functional components: $0.25*HF_Exchange$

References:

C. Adamo and V. Barone, J. Chem. Phys. 108, 664 (1998) (10.1063/1.475428)

- **HGGA_MPW1PBE_XC**

mPW1PBE

Functional components: $0.25*HF_Exchange$

References:

C. Adamo and V. Barone, J. Chem. Phys. 108, 664 (1998) (10.1063/1.475428)

- **HGGA_MPW1PW_XC**

mPW1PW

Functional components: $0.25*HF_Exchange$

References:

C. Adamo and V. Barone, J. Chem. Phys. 108, 664 (1998) (10.1063/1.475428)

- **HGGA_MPW3LYP_XC**
 MPW3LYP
 Functional components: 0.218*HF_Exchange
 References:
 Y. Zhao and D. G. Truhlar, J. Phys. Chem. A 108, 6908 (2004) (10.1021/jp048147q)
- **HGGA_MPW3PW_XC**
 MPW3PW of Adamo & Barone
 Functional components: 0.2*HF_Exchange
 References:
 C. Adamo and V. Barone, J. Chem. Phys. 108, 664 (1998) (10.1063/1.475428)
- **HGGA_MPWLYP1M_XC**
 MPW with 1 par. for metals/LYP
 Functional components: 0.05*HF_Exchange
 References:
 N. E. Schultz, Y. Zhao, and D. G. Truhlar, J. Phys. Chem. A 109, 11127 (2005) (10.1021/jp0539223)
- **HGGA_N12_SX_X**
 Minnesota N12-SX exchange functional
 Functional components: 0.25*HF_Exchange_ShortRange + 0*HF_Exchange_LongRange
 References:
 R. Peverati and D. G. Truhlar, Phys. Chem. Chem. Phys. 14, 16187 (2012) (10.1039/C2CP42576A)
- **HGGA_O3LYP_XC**
 O3LYP
 Functional components: 0.1161*HF_Exchange
 References:
 A. J. Cohen and N. C. Handy, Mol. Phys. 99, 607 (2001) (10.1080/00268970010023435)
- **HGGA_PBE0_13_XC**
 PBE0-1/3
 Functional components: 0.333333*HF_Exchange
 References:
 P. Cortona, J. Chem. Phys. 136, 086101 (2012) (10.1063/1.3690462)
- **HGGA_PBE50_XC**
 PBE50
 Functional components: 0.5*HF_Exchange
 References:
 Y. A. Bernard, Y. Shao, and A. I. Krylov, J. Chem. Phys. 136, 204103 (2012) (10.1063/1.4714499)
- **HGGA_PBE_MOL0_XC**
 PBE_{mol0}
 Functional components: 0.25*HF_Exchange
 References:
 J. M. del Campo, J. L. Gázquez, S. B. Trickey, and A. Vela, J. Chem. Phys. 136, 104108 (2012) (10.1063/1.3691197)
- **HGGA_PBE_MOLB0_XC**
 PBE_{molbeta0}
 Functional components: 0.25*HF_Exchange
 References:
 J. M. del Campo, J. L. Gázquez, S. B. Trickey, and A. Vela, J. Chem. Phys. 136, 104108 (2012) (10.1063/1.3691197)

- **HGGA_PBE_SOLO_XC**
PBEsol0
Functional components: 0.25*HF_Exchange
References:
J. M. del Campo, J. L. Gázquez, S. B. Trickey, and A. Vela, J. Chem. Phys. 136, 104108 (2012) (10.1063/1.3691197)

- **HGGA_PBEB0_XC**
PBEbeta0
Functional components: 0.25*HF_Exchange
References:
J. M. del Campo, J. L. Gázquez, S. B. Trickey, and A. Vela, J. Chem. Phys. 136, 104108 (2012) (10.1063/1.3691197)

- **HGGA_PBEH_XC**
PBEH (PBE0)
Functional components: 0.25*HF_Exchange
References:
C. Adamo and V. Barone, J. Chem. Phys. 110, 6158 (1999) (10.1063/1.478522)
M. Ernzerhof and G. E. Scuseria, J. Chem. Phys. 110, 5029 (1999) (10.1063/1.478401)

- **HGGA_REVB3LYP_XC**
Revised B3LYP
Functional components: 0.2*HF_Exchange
References:
L. Lu, H. Hu, H. Hou, and B. Wang, Comput. Theor. Chem. 1015, 64 (2013) (10.1016/j.comptc.2013.04.009)

- **HGGA_SB98_1A_XC**
SB98 (1a)
Functional components: 0.229015*HF_Exchange
References:
H. L. Schmider and A. D. Becke, J. Chem. Phys. 108, 9624 (1998) (10.1063/1.476438)

- **HGGA_SB98_1B_XC**
SB98 (1b)
Functional components: 0.199352*HF_Exchange
References:
H. L. Schmider and A. D. Becke, J. Chem. Phys. 108, 9624 (1998) (10.1063/1.476438)

- **HGGA_SB98_1C_XC**
SB98 (1c)
Functional components: 0.192416*HF_Exchange
References:
H. L. Schmider and A. D. Becke, J. Chem. Phys. 108, 9624 (1998) (10.1063/1.476438)

- **HGGA_SB98_2A_XC**
SB98 (2a)
Functional components: 0.232055*HF_Exchange
References:
H. L. Schmider and A. D. Becke, J. Chem. Phys. 108, 9624 (1998) (10.1063/1.476438)

- **HGGA_SB98_2B_XC**
SB98 (2b)
Functional components: 0.237978*HF_Exchange
References:
H. L. Schmider and A. D. Becke, J. Chem. Phys. 108, 9624 (1998) (10.1063/1.476438)

- **HGGA_SB98_2C_XC**
 SB98 (2c)
 Functional components: 0.219847*HF_Exchange
 References:
 H. L. Schmider and A. D. Becke, J. Chem. Phys. 108, 9624 (1998) (10.1063/1.476438)
- **HGGA_SOGGA11_X_X**
 Hybrid based on SOGGA11 form
 Functional components: 0.4015*HF_Exchange
 References:
 R. Peverati and D. G. Truhlar, J. Chem. Phys. 135, 191102 (2011) (10.1063/1.3663871)
- **HGGA_TUNED_CAM_B3LYP_XC**
 CAM version of B3LYP, tuned for excitations and properties
 Functional components: 0.0799*HF_Exchange_ShortRange + HF_Exchange_LongRange
 References:
 K. Okuno, Y. Shigeta, R. Kishi, H. Miyasaka, and M. Nakano, J. Photochem. Photobiol., A 235, 29 (2012) (10.1016/j.jphotochem.2012.03.003)
- **HGGA_WB97_XC**
 wB97 range-separated functional
 Functional components: 0*HF_Exchange_ShortRange + HF_Exchange_LongRange
 References:
 J.-D. Chai and M. Head-Gordon, J. Chem. Phys. 128, 084106 (2008) (10.1063/1.2834918)
- **HGGA_WB97X_D_XC**
 wB97D range-separated functional
 Functional components: 0.222036*HF_Exchange_ShortRange + HF_Exchange_LongRange
 References:
 J.-D. Chai and M. Head-Gordon, Phys. Chem. Chem. Phys. 10, 6615 (2008) (10.1039/B810189B)
- **HGGA_WB97X_V_XC**
 wB97X-V range-separated functional
 Functional components: 0.167*HF_Exchange_ShortRange + HF_Exchange_LongRange
 References:
 N. Mardirossian and M. Head-Gordon, Phys. Chem. Chem. Phys. 16, 9904 (2014) (10.1039/C3CP54374A)
- **HGGA_WB97X_XC**
 wB97X range-separated functional
 Functional components: 0.157706*HF_Exchange_ShortRange + HF_Exchange_LongRange
 References:
 J.-D. Chai and M. Head-Gordon, J. Chem. Phys. 128, 084106 (2008) (10.1063/1.2834918)
- **HGGA_X3LYP_XC**
 X3LYP
 Functional components: 0.218*HF_Exchange
 References:
 X. Xu and W. A. Goddard, Proc. Natl. Acad. Sci. U. S. A. 101, 2673 (2004) (10.1073/pnas.0308730100)
- **HMGGA_B86B95_XC**
 Mixture of B86 with BC95
 Functional components: 0.28*HF_Exchange
 References:
 A. D. Becke, J. Chem. Phys. 104, 1040 (1996) (10.1063/1.470829)

- **HMGGA_B88B95_XC**
 Mixture of B88 with BC95 (B1B95)
 Functional components: 0.28*HF_Exchange
 References:
 A. D. Becke, J. Chem. Phys. 104, 1040 (1996) (10.1063/1.470829)
- **HMGGA_BB1K_XC**
 Mixture of B88 with BC95 from Zhao and Truhlar
 Functional components: 0.42*HF_Exchange
 References:
 Y. Zhao, B. J. Lynch, and D. G. Truhlar, J. Phys. Chem. A 108, 2715 (2004) (10.1021/jp049908s)
- **HMGGA_BMK_X**
 Boese-Martin for kinetics
 Functional components: 0.42*HF_Exchange
 References:
 A. D. Boese and J. M. L. Martin, J. Chem. Phys. 121, 3405 (2004) (10.1063/1.1774975)
- **HMGGA_DLDF_X**
 Dispersionless Density Functional
 Functional components: 0.614413*HF_Exchange
 References:
 K. Pernal, R. Podeszwa, K. Patkowski, and K. Szalewicz, Phys. Rev. Lett. 103, 263201 (2009) (10.1103/PhysRevLett.103.263201)
- **HMGGA_M05_2X_X**
 Minnesota M05-2X hybrid exchange functional
 Functional components: 0.56*HF_Exchange
 References:
 Y. Zhao, N. E. Schultz, and D. G. Truhlar, J. Chem. Theory Comput. 2, 364 (2006) (10.1021/ct0502763)
- **HMGGA_M05_X**
 Minnesota M05 hybrid exchange functional
 Functional components: 0.28*HF_Exchange
 References:
 Y. Zhao, N. E. Schultz, and D. G. Truhlar, J. Chem. Phys. 123, 161103 (2005) (10.1063/1.2126975)
- **HMGGA_M06_2X_X**
 Minnesota M06-2X hybrid exchange functional
 Functional components: 0.54*HF_Exchange
 References:
 Y. Zhao and D. G. Truhlar, Theor. Chem. Acc. 120, 215 (2008) (10.1007/s00214-007-0310-x)
- **HMGGA_M06_HF_X**
 Minnesota M06-HF hybrid exchange functional
 Functional components: HF_Exchange
 References:
 Y. Zhao and D. G. Truhlar, J. Phys. Chem. A 110, 13126 (2006) (10.1021/jp066479k)
- **HMGGA_M06_X**
 Minnesota M06 hybrid exchange functional
 Functional components: 0.27*HF_Exchange
 References:
 Y. Zhao and D. G. Truhlar, Theor. Chem. Acc. 120, 215 (2008) (10.1007/s00214-007-0310-x)

- **HMGGA_M08_HX_X**
Minnesota M08-HX hybrid exchange functional
Functional components: $0.5223 \cdot \text{HF_Exchange}$
References:
Y. Zhao and D. G. Truhlar, *J. Chem. Theory Comput.* 4, 1849 (2008) (10.1021/ct800246v)
- **HMGGA_M08_SO_X**
Minnesota M08-SO hybrid exchange functional
Functional components: $0.5679 \cdot \text{HF_Exchange}$
References:
Y. Zhao and D. G. Truhlar, *J. Chem. Theory Comput.* 4, 1849 (2008) (10.1021/ct800246v)
- **HMGGA_M11_X**
Minnesota M11 hybrid exchange functional
Functional components: $0.428 \cdot \text{HF_Exchange_ShortRange} + \text{HF_Exchange_LongRange}$
References:
R. Peverati and D. G. Truhlar, *J. Phys. Chem. Lett.* 2, 2810 (2011) (10.1021/jz201170d)
- **HMGGA_MN12_SX_X**
Minnesota MN12-SX hybrid exchange functional
Functional components: $0.25 \cdot \text{HF_Exchange_ShortRange} + 0 \cdot \text{HF_Exchange_LongRange}$
References:
R. Peverati and D. G. Truhlar, *Phys. Chem. Chem. Phys.* 14, 16187 (2012) (10.1039/C2CP42576A)
- **HMGGA_MN15_X**
Minnesota MN15 hybrid exchange functional
Functional components: $0.44 \cdot \text{HF_Exchange}$
References:
H. S. Yu, X. He, S. L. Li, and D. G. Truhlar, *Chem. Sci.* 7, 5032 (2016) (10.1039/C6SC00705H)
- **HMGGA_MPW1B95_XC**
Mixture of mPW91 with BC95 from Zhao and Truhlar
Functional components: $0.31 \cdot \text{HF_Exchange}$
References:
Y. Zhao and D. G. Truhlar, *J. Phys. Chem. A* 108, 6908 (2004) (10.1021/jp048147q)
- **HMGGA_MPWB1K_XC**
Mixture of mPW91 with BC95 for kinetics
Functional components: $0.44 \cdot \text{HF_Exchange}$
References:
Y. Zhao and D. G. Truhlar, *J. Phys. Chem. A* 108, 6908 (2004) (10.1021/jp048147q)
- **HMGGA_MS2H_X**
MS2 hybrid exchange of Sun, et al
Functional components: $0.09 \cdot \text{HF_Exchange}$
References:
J. Sun, R. Haunschild, B. Xiao, I. W. Bulik, G. E. Scuseria, and J. P. Perdew, *J. Chem. Phys.* 138, 044113 (2013) (10.1063/1.4789414)
- **HMGGA_MVSH_X**
MVSh hybrid exchange functional
Functional components: $0.25 \cdot \text{HF_Exchange}$
References:
J. Sun, J. P. Perdew, and A. Ruzsinszky, *Proc. Natl. Acad. Sci. U. S. A.* 112, 685 (2015) (10.1073/pnas.1423145112)

- **HMGGA_PW6B95_XC**
 Mixture of PW91 with BC95 from Zhao and Truhlar
 Functional components: 0.28*HF_Exchange
 References:
 Y. Zhao and D. G. Truhlar, J. Phys. Chem. A 109, 5656 (2005) (10.1021/jp050536c)
- **HMGGA_PW86B95_XC**
 Mixture of PW86 with BC95
 Functional components: 0.29*HF_Exchange
 References:
 A. D. Becke, J. Chem. Phys. 104, 1040 (1996) (10.1063/1.470829)
- **HMGGA_PWB6K_XC**
 Mixture of PW91 with BC95 from Zhao and Truhlar for kinetics
 Functional components: 0.46*HF_Exchange
 References:
 Y. Zhao and D. G. Truhlar, J. Phys. Chem. A 109, 5656 (2005) (10.1021/jp050536c)
- **HMGGA_REVSCAN0_X**
 revised SCAN hybrid exchange (SCAN0)
 Functional components: 0.25*HF_Exchange
 References:
 P. D. Mezei, G. I. Csonka, and M. Kállay, J. Chem. Theory Comput. 0, null (0) (10.1021/acs.jctc.8b00072)
- **HMGGA_REVTPSSH_XC**
 revTPSSH
 Functional components: 0.1*HF_Exchange
 References:
 G. I. Csonka, J. P. Perdew, and A. Ruzsinszky, J. Chem. Theory Comput. 6, 3688 (2010) (10.1021/ct100488v)
- **HMGGA_SCAN0_X**
 SCAN hybrid exchange (SCAN0)
 Functional components: 0.25*HF_Exchange
 References:
 K. Hui and J.-D. Chai, J. Chem. Phys. 144, 044114 (2016), 10.1063/1.4940734 (10.1063/1.4940734)
- **HMGGA_TAU_HCTH_X**
 Hybrid version of tau-HCTH
 Functional components: 0.15*HF_Exchange
 References:
 A. D. Boese and N. C. Handy, J. Chem. Phys. 116, 9559 (2002) (10.1063/1.1476309)
- **HMGGA_TPSSH_XC**
 TPSSH
 Functional components: 0.1*HF_Exchange
 References:
 V. N. Staroverov, G. E. Scuseria, J. Tao, and J. P. Perdew, J. Chem. Phys. 119, 12129 (2003) (10.1063/1.1626543)
- **HMGGA_WB97M_V_XC**
 wB97M-V exchange-correlation functional
 Functional components: 0.15*HF_Exchange_ShortRange + HF_Exchange_LongRange
 References:
 N. Mardirossian and M. Head-Gordon, J. Chem. Phys. 144, 214110 (2016) (10.1063/1.4952647)

- **HMGGA_X1B95_XC**
Mixture of X with BC95
Functional components: 0.3*HF_Exchange
References:
Y. Zhao and D. G. Truhlar, J. Phys. Chem. A 108, 6908 (2004) (10.1021/jp048147q)
- **HMGGA_XB1K_XC**
Mixture of X with BC95 for kinetics
Functional components: 0.43*HF_Exchange
References:
Y. Zhao and D. G. Truhlar, J. Phys. Chem. A 108, 6908 (2004) (10.1021/jp048147q)
- **LDA_BR78_C**
Bruel & Rothstein 78
References:
G. B. Jr. and S. M. Rothstein, J. Chem. Phys. 69, 1177 (1978) (10.1063/1.436705)
- **LDA_CHACHIYO_C**
Chachiyo simple 2 parameter correlation
References:
T. Chachiyo, J. Chem. Phys. 145, 021101 (2016) (10.1063/1.4958669)
- **LDA_ERF_X**
Attenuated exchange LDA (erf)
References:
J. Toulouse, A. Savin, and H.-J. Flad, Int. J. Quantum Chem. 100, 1047 (2004) (10.1002/qua.20259)
Y. Tawada, T. Tsuneda, S. Yanagisawa, T. Yanai, and K. Hirao, J. Chem. Phys. 120, 8425 (2004) (10.1063/1.1688752)
- **LDA_GDSMFB_XC**
Groth, Dornheim, Sjostrom, Malone, Foulkes, Bonitz
References:
S. {Groth}, T. {Dornheim}, T. {Sjostrom}, F. D. {Malone}, W. M. C. {Foulkes}, and M. {Bonitz},
{ArXiv e-prints (2017), arXiv:1703.08074 [physics.plasm-ph].
- **LDA_GK72_C**
Gordon and Kim 1972
References:
R. G. Gordon and Y. S. Kim, J. Chem. Phys. 56, 3122 (1972), <https://doi.org/10.1063/1.1677649>
(10.1063/1.1677649)
- **LDA_GL_C**
Gunnarson & Lundqvist
References:
O. Gunnarsson and B. I. Lundqvist, Phys. Rev. B 13, 4274 (1976) (10.1103/PhysRevB.13.4274)
- **LDA_GOMBAS_C**
Gombas
References:
P. Gombas, Pseudopotentiale (Springer-Verlag, Wien, New York, 1967)
- **LDA_HL_C**
Hedin & Lundqvist
References:
L. Hedin and B. I. Lundqvist, J. Phys. C: Solid State Phys. 4, 2064 (1971) (10.1088/0022-3719/4/14/022)

- **LDA_KARASIEV_C**
Karasiev reparameterization of Chachiyo
References:
V. V. Karasiev, J. Chem. Phys. 145, 157101 (2016), <https://doi.org/10.1063/1.4964758>
(10.1063/1.4964758)
- **LDA_KSDT_XC**
Karasiev, Sjostrom, Dufty & Trickey
References:
V. V. Karasiev, T. Sjostrom, J. Dufty, and S. B. Trickey, Phys. Rev. Lett. 112, 076403 (2014)
(10.1103/PhysRevLett.112.076403)
- **LDA_LP96_C**
Liu-Parr correlation
References:
S. Liu and R. G. Parr, Phys. Rev. A 53, 2211 (1996) (10.1103/PhysRevA.53.2211)
S. Liu and R. Parr, Journal of Molecular Structure:THEOCHEM 501–502, 29 (2000)} (10.1016/S0166-1280(99)00410-8)
- **LDA_LP_A_XC**
Lee-Parr reparametrization A
References:
C. Lee and R. G. Parr, Phys. Rev. A 42, 193 (1990) (10.1103/PhysRevA.42.193)
- **LDA_LP_B_XC**
Lee-Parr reparametrization B
References:
C. Lee and R. G. Parr, Phys. Rev. A 42, 193 (1990) (10.1103/PhysRevA.42.193)
- **LDA_MCWEENY_C**
McWeeny 76
References:
R. McWeeny, in The New World of Quantum Chemistry, edited by {editor {B. Pullman} and R. Parr} (Reidel, Boston, 1976) pp. 3–31
G. B. Jr. and S. M. Rothstein, J. Chem. Phys. 69, 1177 (1978) (10.1063/1.436705)
- **LDA_ML1_C**
Modified LSD (version 1) of Proynov and Salahub
References:
E. I. Proynov and D. R. Salahub, Phys. Rev. B 49, 7874 (1994) (10.1103/PhysRevB.49.7874)
- **LDA_ML2_C**
Modified LSD (version 2) of Proynov and Salahub
References:
E. I. Proynov and D. R. Salahub, Phys. Rev. B 49, 7874 (1994) (10.1103/PhysRevB.49.7874)
- **LDA_OB_PW_C**
Ortiz & Ballone (PW parametrization)
References:
G. Ortiz and P. Ballone, Phys. Rev. B 50, 1391 (1994) (10.1103/PhysRevB.50.1391)
G. Ortiz and P. Ballone, Phys. Rev. B 56, 9970 (1997) (10.1103/PhysRevB.56.9970)
J. P. Perdew and Y. Wang, Phys. Rev. B 45, 13244 (1992), added extra digits to some constants as in the PBE routine (<http://dft.rutgers.edu/pubs/PBE.asc>) (10.1103/PhysRevB.45.13244)
- **LDA_OB_PZ_C**
Ortiz & Ballone (PZ parametrization)
References:

- G. Ortiz and P. Ballone, Phys. Rev. B 50, 1391 (1994) (10.1103/PhysRevB.50.1391)
 G. Ortiz and P. Ballone, Phys. Rev. B 56, 9970 (1997) (10.1103/PhysRevB.56.9970)
- **LDA_OW_C**
 Optimized Wigner
 References:
 P. A. Stewart and P. M. W. Gill, J. Chem. Soc. {, Faraday Trans. 91, 4337 (1995)} (10.1039/FT9959104337)

 - **LDA_OW_LYP_C**
 Wigner with corresponding LYP parameters
 References:
 P. A. Stewart and P. M. W. Gill, J. Chem. Soc. {, Faraday Trans. 91, 4337 (1995)} (10.1039/FT9959104337)

 - **LDA_PK09_C**
 Proynov and Kong 2009
 References:
 E. Proynov and J. Kong, Phys. Rev. A 79, 014103 (2009) (10.1103/PhysRevA.79.014103)
 E. Proynov and J. Kong, Phys. Rev. A 95, 059904 (2017) (10.1103/PhysRevA.95.059904)

 - **LDA_PW_C**
 Perdew & Wang
 References:
 J. P. Perdew and Y. Wang, Phys. Rev. B 45, 13244 (1992) (10.1103/PhysRevB.45.13244)

 - **LDA_PW_MOD_C**
 Perdew & Wang (modified)
 References:
 J. P. Perdew and Y. Wang, Phys. Rev. B 45, 13244 (1992), added extra digits to some constants as in the PBE routine (<http://dft.rutgers.edu/pubs/PBE.asc>) (10.1103/PhysRevB.45.13244)

 - **LDA_PW_RPA_C**
 Perdew & Wang (fit to the RPA energy)
 References:
 J. P. Perdew and Y. Wang, Phys. Rev. B 45, 13244 (1992) (10.1103/PhysRevB.45.13244)

 - **LDA_PZ_C**
 Perdew & Zunger
 References:
 J. P. Perdew and A. Zunger, Phys. Rev. B 23, 5048 (1981) (10.1103/PhysRevB.23.5048)

 - **LDA_PZ_MOD_C**
 Perdew & Zunger (Modified)
 References:
 J. P. Perdew and A. Zunger, Phys. Rev. B 23, 5048 (1981), modified to improve the matching between the low- and high-rs parts (10.1103/PhysRevB.23.5048)

 - **LDA_RAE_X**
 Rae self-energy corrected exchange
 References:
 A. Rae, Chem. Phys. Lett. 18, 574 (1973) (10.1016/0009-2614(73)80469-5)

 - **LDA_RC04_C**
 Ragot-Cortona
 References:
 S. Ragot and P. Cortona, J. Chem. Phys. 121, 7671 (2004) (10.1063/1.1792153)

- **LDA_REL_X**
Slater exchange with relativistic corrections
References:
A. K. Rajagopal, J. Phys. C: Solid State Phys. 11, L943 (1978) (10.1088/0022-3719/11/24/002)
A. H. MacDonald and S. H. Vosko, J. Phys. C: Solid State Phys. 12, 2977 (1979) (10.1088/0022-3719/12/15/007)
E. Engel, S. Keller, A. F. Bonetti, H. Müller, and R. M. Dreizler, Phys. Rev. A 52, 2750 (1995) (10.1103/PhysRevA.52.2750)

- **LDA_RPA_C**
Random Phase Approximation (RPA)
References:
M. Gell-Mann and K. A. Brueckner, Phys. Rev. 106, 364 (1957) (10.1103/PhysRev.106.364)

- **LDA_SLATER_LXC_X**
Slater exchange
References:
P. A. M. Dirac, Math. Proc. Cambridge Philos. Soc. 26, 376 (1930) (10.1017/S0305004100016108)
F. Bloch, Z. Phys. 57, 545 (1929) (10.1007/BF01340281)

- **LDA_SLATER_X**
Slater exchange
References:
P. A. M. Dirac, Math. Proc. Cambridge Philos. Soc. 26, 376 (1930) (10.1017/S0305004100016108)
F. Bloch, Z. Phys. 57, 545 (1929) (10.1007/BF01340281)

- **LDA_TETER93_XC**
Teter 93
References:
S. Goedecker, M. Teter, and J. Hutter, Phys. Rev. B 54, 1703 (1996) (10.1103/PhysRevB.54.1703)

- **LDA_VBH_C**
von Barth & Hedin
References:
U. von Barth and L. Hedin, J. Phys. C: Solid State Phys. 5, 1629 (1972) (10.1088/0022-3719/5/13/012)

- **LDA_VWN1_C**
Vosko, Wilk & Nusair (VWN1)
References:
S. H. Vosko, L. Wilk, and M. Nusair, Can. J. Phys. 58, 1200 (1980) (10.1139/p80-159)

- **LDA_VWN1RPA_C**
Vosko, Wilk & Nusair (VWN5_RPA)
References:
S. H. Vosko, L. Wilk, and M. Nusair, Can. J. Phys. 58, 1200 (1980) (10.1139/p80-159)

- **LDA_VWN5_C**
Vosko, Wilk & Nusair (VWN5)
References:
S. H. Vosko, L. Wilk, and M. Nusair, Can. J. Phys. 58, 1200 (1980) (10.1139/p80-159)

- **LDA_VWN5RPA_C**
Vosko, Wilk & Nusair (VWN5_RPA)
References:
S. H. Vosko, L. Wilk, and M. Nusair, Can. J. Phys. 58, 1200 (1980) (10.1139/p80-159)

- **LDA_VWN_1_C**
Vosko, Wilk & Nusair (VWN1)
References:
S. H. Vosko, L. Wilk, and M. Nusair, *Can. J. Phys.* 58, 1200 (1980) (10.1139/p80-159)
- **LDA_VWN_2_C**
Vosko, Wilk & Nusair (VWN2)
References:
S. H. Vosko, L. Wilk, and M. Nusair, *Can. J. Phys.* 58, 1200 (1980) (10.1139/p80-159)
- **LDA_VWN_3_C**
Vosko, Wilk & Nusair (VWN3)
References:
S. H. Vosko, L. Wilk, and M. Nusair, *Can. J. Phys.* 58, 1200 (1980) (10.1139/p80-159)
- **LDA_VWN_4_C**
Vosko, Wilk & Nusair (VWN4)
References:
S. H. Vosko, L. Wilk, and M. Nusair, *Can. J. Phys.* 58, 1200 (1980) (10.1139/p80-159)
- **LDA_VWN_C**
Vosko, Wilk & Nusair (VWN5)
References:
S. H. Vosko, L. Wilk, and M. Nusair, *Can. J. Phys.* 58, 1200 (1980) (10.1139/p80-159)
- **LDA_VWN_RPA_C**
Vosko, Wilk & Nusair (VWN5_RPA)
References:
S. H. Vosko, L. Wilk, and M. Nusair, *Can. J. Phys.* 58, 1200 (1980) (10.1139/p80-159)
- **LDA_WIGNER_C**
Wigner
References:
E. Wigner, *Trans. Faraday Soc.* 34, 678 (1938) (10.1039/TF9383400678)
P. A. Stewart and P. M. W. Gill, *J. Chem. Soc. {, Faraday Trans. 91, 4337 (1995)}* (10.1039/FT9959104337)
- **LDA_XALPHA_C**
Slater's Xalpha
References:
J. C. Slater, *Phys. Rev.* 81, 385 (1951) (10.1103/PhysRev.81.385)
- **LDA_ZLP_XC**
Zhao, Levy & Parr, Eq. (20)
References:
Q. Zhao, M. Levy, and R. G. Parr, *Phys. Rev. A* 47, 918 (1993) (10.1103/PhysRevA.47.918)
- **MGGA_B88_C**
Meta-GGA correlation by Becke
References:
A. D. Becke, *J. Chem. Phys.* 88, 1053 (1988) (10.1063/1.454274)
- **MGGA_B97M_V_XC**
B97M-V exchange-correlation functional
References:
N. Mardirossian and M. Head-Gordon, *J. Chem. Phys.* 142, 074111 (2015) (10.1063/1.4907719)

- **MGGA_BC95_C**
Becke correlation 95
References:
A. D. Becke, *J. Chem. Phys.* 104, 1040 (1996) (10.1063/1.470829)
- **MGGA_BLOC_X**
functional with balanced localization
References:
L. A. Constantin, E. Fabiano, and F. Della Sala, *J. Chem. Theory Comput.* 9, 2256 (2013) (10.1021/ct400148r)
- **MGGA_DLDF_C**
Dispersionless Density Functional
References:
K. Pernal, R. Podeszwa, K. Patkowski, and K. Szalewicz, *Phys. Rev. Lett.* 103, 263201 (2009) (10.1103/PhysRevLett.103.263201)
- **MGGA_GVT4_X**
GVT4 (X part of VSXC)
References:
T. V. Voorhis and G. E. Scuseria, *J. Chem. Phys.* 109, 400 (1998) (10.1063/1.476577)
- **MGGA_GX_X**
GX functional of Loos
References:
P.-F. Loos, *J. Chem. Phys.* 146, 114108 (2017) (10.1063/1.4978409)
- **MGGA_HLE17_XC**
high local exchange 2017
References:
P. Verma and D. G. Truhlar, *J. Phys. Chem C* 121, 7144 (2017) (10.1021/acs.jpcc.7b01066)
- **MGGA_LTA_X**
Local tau approximation
References:
M. Ernzerhof and G. E. Scuseria, *J. Chem. Phys.* 111, 911 (1999) (10.1063/1.479374)
- **MGGA_M05_2X_C**
Minnesota M05-2X correlation functional
References:
Y. Zhao, N. E. Schultz, and D. G. Truhlar, *J. Chem. Theory Comput.* 2, 364 (2006) (10.1021/ct0502763)
- **MGGA_M05_C**
Minnesota M05 correlation functional
References:
Y. Zhao, N. E. Schultz, and D. G. Truhlar, *J. Chem. Phys.* 123, 161103 (2005) (10.1063/1.2126975)
- **MGGA_M06_2X_C**
Minnesota M06-2X correlation functional
References:
Y. Zhao and D. G. Truhlar, *Theor. Chem. Acc.* 120, 215 (2008) (10.1007/s00214-007-0310-x)
- **MGGA_M06_C**
Minnesota M06 correlation functional
References:
Y. Zhao and D. G. Truhlar, *Theor. Chem. Acc.* 120, 215 (2008) (10.1007/s00214-007-0310-x)

- **MGGA_M06_HF_C**
Minnesota M06-HF correlation functional
References:
Y. Zhao and D. G. Truhlar, *J. Phys. Chem. A* 110, 13126 (2006) (10.1021/jp066479k)
- **MGGA_M06_L_C**
Minnesota M06-L correlation functional
References:
Y. Zhao and D. G. Truhlar, *J. Chem. Phys.* 125, 194101 (2006) (10.1063/1.2370993)
Y. Zhao and D. G. Truhlar, *Theor. Chem. Acc.* 120, 215 (2008) (10.1007/s00214-007-0310-x)
- **MGGA_M06_L_X**
Minnesota M06-L exchange functional
References:
Y. Zhao and D. G. Truhlar, *J. Chem. Phys.* 125, 194101 (2006) (10.1063/1.2370993)
Y. Zhao and D. G. Truhlar, *Theor. Chem. Acc.* 120, 215 (2008) (10.1007/s00214-007-0310-x)
- **MGGA_M08_HX_C**
Minnesota M08 correlation functional
References:
Y. Zhao and D. G. Truhlar, *J. Chem. Theory Comput.* 4, 1849 (2008) (10.1021/ct800246v)
- **MGGA_M08_SO_C**
Minnesota M08-SO correlation functional
References:
Y. Zhao and D. G. Truhlar, *J. Chem. Theory Comput.* 4, 1849 (2008) (10.1021/ct800246v)
- **MGGA_M11_C**
Minnesota M11 correlation functional
References:
R. Peverati and D. G. Truhlar, *J. Phys. Chem. Lett.* 2, 2810 (2011) (10.1021/jz201170d)
- **MGGA_M11_L_C**
Minnesota M11-L correlation functional
References:
R. Peverati and D. G. Truhlar, *J. Phys. Chem. Lett.* 3, 117 (2012) (10.1021/jz201525m)
- **MGGA_M11_L_X**
Minnesota M11-L exchange functional
References:
R. Peverati and D. G. Truhlar, *J. Phys. Chem. Lett.* 3, 117 (2012) (10.1021/jz201525m)
- **MGGA_MBEEF_X**
mBEEF exchange
References:
J. Wellendorff, K. T. Lundgaard, K. W. Jacobsen, and T. Bligaard, *J. Chem. Phys.* 140, 144107 (2014) (10.1063/1.4870397)
- **MGGA_MBEEFVDW_X**
mBEEF-vdW exchange
References:
K. T. Lundgaard, J. Wellendorff, J. Voss, K. W. Jacobsen, and T. Bligaard, *Phys. Rev. B* 93, 235162 (2016) (10.1103/PhysRevB.93.235162)
- **MGGA_MN12_L_C**
Minnesota MN12-L correlation functional
References:

- R. Peverati and D. G. Truhlar, *Phys. Chem. Chem. Phys.* 14, 13171 (2012) (10.1039/C2CP42025B)
- **MGGA_MN12_L_X**
Minnesota MN12-L exchange functional
References:
R. Peverati and D. G. Truhlar, *Phys. Chem. Chem. Phys.* 14, 13171 (2012) (10.1039/C2CP42025B)
 - **MGGA_MN12_SX_C**
Minnesota MN12-SX correlation functional
References:
R. Peverati and D. G. Truhlar, *Phys. Chem. Chem. Phys.* 14, 16187 (2012) (10.1039/C2CP42576A)
 - **MGGA_MN15_C**
Minnesota MN15 correlation functional
References:
H. S. Yu, X. He, S. L. Li, and D. G. Truhlar, *Chem. Sci.* 7, 5032 (2016) (10.1039/C6SC00705H)
 - **MGGA_MN15_L_C**
Minnesota MN15-L correlation functional
References:
H. S. Yu, X. He, and D. G. Truhlar, *J. Chem. Theory Comput.* 12, 1280 (2016) (10.1021/acs.jctc.5b01082)
 - **MGGA_MN15_L_X**
Minnesota MN15-L exchange functional
References:
H. S. Yu, X. He, and D. G. Truhlar, *J. Chem. Theory Comput.* 12, 1280 (2016) (10.1021/acs.jctc.5b01082)
 - **MGGA_MODTPSS_X**
Modified Tao, Perdew, Staroverov & Scuseria
References:
J. P. Perdew, A. Ruzsinszky, J. Tao, G. I. Csonka, and G. E. Scuseria, *Phys. Rev. A* 76, 042506 (2007) (10.1103/PhysRevA.76.042506)
 - **MGGA_MS0_X**
MS exchange of Sun, Xiao, and Ruzsinszky
References:
J. Sun, B. Xiao, and A. Ruzsinszky, *J. Chem. Phys.* 137, 051101 (2012) (10.1063/1.4742312)
 - **MGGA_MS1_X**
MS1 exchange of Sun, et al
References:
J. Sun, R. Haunschild, B. Xiao, I. W. Bulik, G. E. Scuseria, and J. P. Perdew, *J. Chem. Phys.* 138, 044113 (2013) (10.1063/1.4789414)
 - **MGGA_MS2_X**
MS2 exchange of Sun, et al
References:
J. Sun, R. Haunschild, B. Xiao, I. W. Bulik, G. E. Scuseria, and J. P. Perdew, *J. Chem. Phys.* 138, 044113 (2013) (10.1063/1.4789414)
 - **MGGA_MVS_X**
MVS exchange of Sun, Perdew, and Ruzsinszky

References:

J. Sun, J. P. Perdew, and A. Ruzsinszky, Proc. Natl. Acad. Sci. U. S. A. 112, 685 (2015) (10.1073/pnas.1423145112)

- **MGGA_OTPSS_D_XC**

oTPSS-D functional of Goerigk and Grimme

References:

L. Goerigk and S. Grimme, J. Chem. Theory Comput. 6, 107 (2010) (10.1021/ct900489g)

- **MGGA_PBE_GX_X**

PBE-GX functional of Loos

References:

P.-F. Loos, J. Chem. Phys. 146, 114108 (2017) (10.1063/1.4978409)

- **MGGA_PKZB_C**

Perdew, Kurth, Zupan, and Blaha

References:

J. P. Perdew, S. Kurth, A. Zupan, and P. Blaha, Phys. Rev. Lett. 82, 2544 (1999) (10.1103/PhysRevLett.82.2544)

- **MGGA_PKZB_X**

Perdew, Kurth, Zupan, and Blaha

References:

J. P. Perdew, S. Kurth, A. Zupan, and P. Blaha, Phys. Rev. Lett. 82, 2544 (1999) (10.1103/PhysRevLett.82.2544)

- **MGGA_REVM06_L_C**

Minnesota revM06-L correlation functional

References:

Y. Wang, X. Jin, H. S. Yu, D. G. Truhlar, and X. He, Proceedings of the National Academy of Sciences 114, 8487 (2017) (10.1073/pnas.1705670114)

- **MGGA_REVM06_L_X**

Minnesota revM06-L exchange functional

References:

Y. Wang, X. Jin, H. S. Yu, D. G. Truhlar, and X. He, Proceedings of the National Academy of Sciences 114, 8487 (2017) (10.1073/pnas.1705670114)

- **MGGA_REVSCAN_C**

revised SCAN

References:

P. D. Mezei, G. I. Csonka, and M. Kállay, J. Chem. Theory Comput. 0, null (0) (10.1021/acs.jctc.8b00072)

- **MGGA_REVSCAN_VV10_C**

REVSCAN + VV10 correlation

References:

P. D. Mezei, G. I. Csonka, and M. Kállay, J. Chem. Theory Comput. 0, null (0) (10.1021/acs.jctc.8b00072)

- **MGGA_REVSCAN_X**

revised SCAN

References:

P. D. Mezei, G. I. Csonka, and M. Kállay, J. Chem. Theory Comput. 0, null (0) (10.1021/acs.jctc.8b00072)

- **MGGA_REVTPSS_C**
revised TPSS correlation
References:
J. P. Perdew, A. Ruzsinszky, G. I. Csonka, L. A. Constantin, and J. Sun, Phys. Rev. Lett. 103, 026403 (2009) (10.1103/PhysRevLett.103.026403)
J. P. Perdew, A. Ruzsinszky, G. I. Csonka, L. A. Constantin, and J. Sun, Phys. Rev. Lett. 106, 179902 (2011) (10.1103/PhysRevLett.106.179902)
- **MGGA_REVTPSS_X**
revised Tao, Perdew, Staroverov & Scuseria
References:
J. P. Perdew, A. Ruzsinszky, G. I. Csonka, L. A. Constantin, and J. Sun, Phys. Rev. Lett. 103, 026403 (2009) (10.1103/PhysRevLett.103.026403)
J. P. Perdew, A. Ruzsinszky, G. I. Csonka, L. A. Constantin, and J. Sun, Phys. Rev. Lett. 106, 179902 (2011) (10.1103/PhysRevLett.106.179902)
- **MGGA_SA_TPSS_X**
TPSS with correct surface asymptotics
References:
L. A. Constantin, E. Fabiano, J. M. Pitarke, and F. Della Sala, Phys. Rev. B 93, 115127 (2016) (10.1103/PhysRevB.93.115127)
- **MGGA_SCAN_C**
SCAN correlation of Sun, Ruzsinszky, and Perdew
References:
J. Sun, A. Ruzsinszky, and J. P. Perdew, Phys. Rev. Lett. 115, 036402 (2015) (10.1103/PhysRevLett.115.036402)
- **MGGA_SCAN_RVV10_C**
SCAN + rVV10 correlation
References:
H. Peng, Z.-H. Yang, J. P. Perdew, and J. Sun, Phys. Rev. X 6, 041005 (2016) (10.1103/PhysRevX.6.041005)
- **MGGA_SCAN_VV10_C**
SCAN + VV10 correlation
References:
J. G. Brandenburg, J. E. Bates, J. Sun, and J. P. Perdew, Phys. Rev. B 94, 115144 (2016) (10.1103/PhysRevB.94.115144)
- **MGGA_SCAN_X**
SCAN exchange of Sun, Ruzsinszky, and Perdew
References:
J. Sun, A. Ruzsinszky, and J. P. Perdew, Phys. Rev. Lett. 115, 036402 (2015) (10.1103/PhysRevLett.115.036402)
- **MGGA_TAU_HCTH_X**
tau-HCTH from Boese and Handy
References:
A. D. Boese and N. C. Handy, J. Chem. Phys. 116, 9559 (2002) (10.1063/1.1476309)
- **MGGA_TM_X**
Tao and Mo 2016
References:
J. Tao and Y. Mo, Phys. Rev. Lett. 117, 073001 (2016) (10.1103/PhysRevLett.117.073001)

- **MGGA_TPSS_C**
 Tao, Perdew, Staroverov & Scuseria
 References:
 J. Tao, J. P. Perdew, V. N. Staroverov, and G. E. Scuseria, Phys. Rev. Lett. 91, 146401 (2003) (10.1103/PhysRevLett.91.146401)
 J. P. Perdew, J. Tao, V. N. Staroverov, and G. E. Scuseria, J. Chem. Phys. 120, 6898 (2004) (10.1063/1.1665298)
- **MGGA_TPSS_LXC_C**
 Tao, Perdew, Staroverov & Scuseria
 References:
 J. Tao, J. P. Perdew, V. N. Staroverov, and G. E. Scuseria, Phys. Rev. Lett. 91, 146401 (2003) (10.1103/PhysRevLett.91.146401)
 J. P. Perdew, J. Tao, V. N. Staroverov, and G. E. Scuseria, J. Chem. Phys. 120, 6898 (2004) (10.1063/1.1665298)
- **MGGA_TPSS_LXC_X**
 Tao, Perdew, Staroverov & Scuseria
 References:
 J. Tao, J. P. Perdew, V. N. Staroverov, and G. E. Scuseria, Phys. Rev. Lett. 91, 146401 (2003) (10.1103/PhysRevLett.91.146401)
 J. P. Perdew, J. Tao, V. N. Staroverov, and G. E. Scuseria, J. Chem. Phys. 120, 6898 (2004) (10.1063/1.1665298)
- **MGGA_TPSS_X**
 Tao, Perdew, Staroverov & Scuseria
 References:
 J. Tao, J. P. Perdew, V. N. Staroverov, and G. E. Scuseria, Phys. Rev. Lett. 91, 146401 (2003) (10.1103/PhysRevLett.91.146401)
 J. P. Perdew, J. Tao, V. N. Staroverov, and G. E. Scuseria, J. Chem. Phys. 120, 6898 (2004) (10.1063/1.1665298)
- **MGGA_TPSS_XC**

 Functional components: MGGA_TPSS_C + MGGA_TPSS_X
 References:
 J. Tao, J. P. Perdew, V. N. Staroverov, and G. E. Scuseria, Phys. Rev. Lett. 91, 146401 (2003) (10.1103/PhysRevLett.91.146401)
 J. P. Perdew, J. Tao, V. N. Staroverov, and G. E. Scuseria, J. Chem. Phys. 120, 6898 (2004) (10.1063/1.1665298)
 J. Tao, J. P. Perdew, V. N. Staroverov, and G. E. Scuseria, Phys. Rev. Lett. 91, 146401 (2003) (10.1103/PhysRevLett.91.146401)
 J. P. Perdew, J. Tao, V. N. Staroverov, and G. E. Scuseria, J. Chem. Phys. 120, 6898 (2004) (10.1063/1.1665298)
- **MGGA_TPSSLOC_C**
 Semilocal dynamical correlation
 References:
 L. A. Constantin, E. Fabiano, and F. Della Sala, Phys. Rev. B 86, 035130 (2012) (10.1103/PhysRevB.86.035130)
- **MGGA_TPSSLYP1W_XC**
 TPSSLYP1W
 References:
 E. E. Dahlke and D. G. Truhlar, J. Phys. Chem. B 109, 15677 (2005) (10.1021/jp052436c)

- **MGGA_VSXC_C**
VSXC (correlation part)
References:
T. V. Voorhis and G. E. Scuseria, J. Chem. Phys. 109, 400 (1998) (10.1063/1.476577)
- **MGGA_VT84_X**
meta-GGA version of VT{8,4} GGA
References:
J. M. del Campo, J. L. Gázquez, S. Trickey, and A. Vela, Chem. Phys. Lett. 543, 179 (2012) (10.1016/j.cplett.2012.06.025)