

# QuantumATK

## Release Features

### Version O-2018.06

Last amended: June 2018

## QuantumATK O-2018.06



QuantumATK is a leading industry-proven platform for atomic-scale modeling of materials, nanostructures, and nanoelectronic devices. It includes quantum mechanical methods such as density functional theory (DFT) with either LCAO or plane-wave basis sets and semi-empirical models, simulation engine for atomic-scale simulations using classical potentials, module for nanoscale device and transport simulations using non-equilibrium Green's function (NEGF) methodology. QuantumATK combines the power of a Python scripting engine with the ease-of-use provided by an intuitive graphical user interface, NanoLab. All simulation engines share a common infrastructure for analysis, ion dynamics and parallel performance techniques.

► <b>New Features in QuantumATK O-2018.06</b>	<b>2</b>
• QuantumATK PlaneWave DFT Code	2
• New Pseudopotentials	3
• New Semiempirical Models	3
• New ForceField Potentials	4
• Performance Improvements	4
• New Study Object Concept	5
• Device Configuration Updates	6
• Electron Mobility Module Updates	7
• Calculation of Spin Parameters	7
• Photocurrent Module	8
• Special Quasi-random Structures (SQS) Algorithm for Generating Random Alloys	8
• Major NanoLab Builder Updates	8
• General NanoLab Updates	9
• Major NanoLab Builder Updates	11
• Miscellaneous	11
► <b>Service Pack QuantumATK O-2018.06-SP1-1</b>	<b>12</b>

### QuantumATK PlaneWave DFT Code

With the O-2018.06 release the Plane-Wave code is ready for production runs. Being fully integrated into the QuantumATK NanoLab environment, it is probably the most flexible and user friendly plane-wave code available. This makes it possible to shift seamlessly from LCAO to Plane-Wave basis sets, thus allowing for easily adjusting and testing tradeoffs between speed and accuracy. Check out the [tutorial on the QuantumATK PlaneWave calculator](#).

- ▶ **Modern plane-wave DFT code, implemented from scratch based on state-of-the-art methods**
- ▶ **Comparable performance to QuantumEspresso in all respects**
- ▶ **Noncollinear spin & spin-orbit calculations**
- ▶ **Norm-conserving pseudopotentials (SG15 & PseudoDojo potentials for LDA and GGA provided)**
- ▶ **HSE06 hybrid functional**
  - Uses the novel Adaptively Compressed Exchange (ACE) operator, which provides much better performance than traditional algorithms (cf. <https://arxiv.org/abs/1601.07159> & <https://pubs.acs.org/doi/10.1021/acs.jctc.6b00092>)
- ▶ **Eigensolvers**
  - Default: Generalized Davidson method – stable and robust method
  - Also includes (as the first code) the Projected Preconditioned Conjugate Gradient (PPCG) algorithm for computing many extreme eigenpairs of a Hermitian matrix (cf. <https://arxiv.org/abs/1407.7506> & <https://doi.org/10.1016/j.jcp.2015.02.030>); provides better scaling for large scale parallel calculations (single k-point, many processes per k-point)
- ▶ **Multilevel parallelism**
  - Over images in NEB and similarly for other complex tasks
  - Over k-points
  - Over plane-wave states (using multiple processes per k-point)
  - Also for band structure, DOS etc.
  - Automatic algorithm to determine the default (optimal) number of processes per k-point
- ▶ **Almost all the same functionality as in LCAO**
  - Band structure, effective mass, DOS, total energy, LocalBandStructure, etc.
  - Phonon calculations
  - Force/stress optimization, ab initio MD, NEB

#### QuantumATK Advantages Over other Plane-Wave Codes:

- ▶ **User-friendliness**
  - Integrated in the QuantumATK NanoLab GUI + natively coupled to Python for batch processing and advanced logic
  - Self-consistent calculation, geometry optimization, band structure, calculation of dynamical matrix, etc. can be done in one workflow (single Python script)
  - No need to use the same number of processes on save and read of data in parallel
- ▶ **Fast HSE implementation use the ACE method**
  - Not available in VASP
  - HSE band structure (only available in QE by including all the k-points in the self-consistent loop)
  - Automatically turns on restarting from wave functions for highest efficiency (can be turned off to save memory)
- ▶ **Performance benefits via advanced restart/initialization capabilities**
  - Use LCAO DFT calculation as starting density (beneficial for large and/or non-symmetric systems)
  - Restart with increased cutoff; beneficial in convergence studies
  - Noncollinear calculations can be initialized from a noncollinear or collinear (spin polarized or not) valence density, which is rotated into its noncollinear form (using the same direction theta/phi for all atoms); beneficial for e.g. magnetic anisotropy calculations
  - Non-selfconsistent calculation: density > effective potential > Hamiltonian > eigenvalues/wavefunctions

- ▶ First code to implement PPCG
- ▶ Support for non-periodic boundary conditions in Poisson solver for slabs, metallic regions, electric fields, ...
- ▶ Default cut-offs for the pseudopotentials, benchmarked to give good values in  $\Delta$ -tests (cf. <https://molmod.ugent.be/deltacodesdft>)
- ▶ Available as calculator for Sentaurus Materials Workbench from Synopsys

## New Pseudopotentials

Modern norm conserving pseudo potentials provide accuracy essentially similar to all electron calculations. With the inclusion of the Pseudo-Dojo set, QuantumATK provides the current state of the art within norm-conserving pseudo potentials with carefully tested default numerical settings.

- ▶ Added SG15 for LDA
- ▶ New default PPs, taken from [pseudo-dojo.org](http://pseudo-dojo.org)
  - LDA and GGA
  - Good delta-test values
  - Significantly softer than SG15 (about 25% reduction in cut-off), important for plane-wave
- ▶ Low basis sets for Si/Ge (PseudoDojo)

## New Semiempirical Models

Our ambition is to provide the most complete semi-empirical package for material systems. In this release we have added the complete DFTB.org parameter set, the famous Naval Research tight-binding library, and tight-binding models for describing the electronic structure of large strained systems.

- ▶ All parameters from <http://dftb.org> are now included and easily available via the Script Generator
- ▶ Unified SemiEmpiricalCalculator in a python back-engine
  - Replaces SlaterKosterCalculator and Huckel Calculator
  - Simplifies implementation of new tight-binding models
- ▶ Added tight-binding model for strained systems
  - T. B. Boykin *et al.*, Phys. Rev. B **81**, 125202 (2010)
  - Efficient method for calculating both structural (phonons, equation of state, molecular dynamics, etc.) and electronic properties of materials
  - Non-orthogonal environment-dependent Slater-Koster-like tight-binding model, which gives good transferability from bulk to surfaces, wires, amorphous systems, etc.
  - Unlike classical empirical potentials, the model explicitly incorporates the real electronic structure and bonding of the material, obtained by an interpolation from a database of first-principles results
  - Significantly lower memory and runtime compared to DFT – for instance MD simulations with 10,000 atoms have been demonstrated efficiently
  - Able to handle monoatomic, binary, and ternary compounds (metals, insulators, and semiconductors)
  - Support in QuantumATK 2018.06 is so far limited to electronic properties (band structure, DOS) for a validated subset of materials, but work is in progress to support the full capabilities of the model for more materials
  - Uses the same electron charge density model as in other QuantumATK semi-empirical methods to provide self-consistency
  - Supported in the Script Generator

## New ForceField Potentials

The most important aspect of a force field package is the available potential parameters. The QuantumATK force field is a very flexible package with one of the most complete potential parameters database for materials systems. It is still often difficult to find good potentials for complex materials combinations; in the new release we overcome this problem by providing a materials editor that makes it easy to connect a number of force fields into a combined materials description.

The database in QuantumATK contains more than 300 empirical classical potentials

<https://docs.quantumatk.com/manual/ATKForceField.html>

The ForceField simulation engine is based on the TremoloX Calculator which is optimized for large scale simulations and provides state-of-the-art scalable fast methods. TremoloX is developed by the **Fraunhofer Institute of Algorithms and Scientific Calculations (SCAI)**.

## BYOP (Bring Your Own Potential)

- ▶ **Python interface for adding your own or literature potential of any of the supported types**
- ▶ **Support for custom combinations of potentials**
  - E.g. use a Stillinger-Weber potential with a Lennard-Jones term to account for van der Waals interaction
  - Several such potentials from literature are already provided: Pedone, Guillot-Sator, Marian-Gastreich, Feuston-Garofalini, Matsui, Leinenweber, Madden, and more

## Performance Improvements

QuantumATK strives to be the most computationally efficient atomic-scale modelling suite. In each release we aim at a factor 2 speed improvement. O.2018.06 is no exception and enjoys a large number of both memory and performance improvements, both for serial and parallel runs.

### Memory

- ▶ **Reduction of the memory requirements for many different calculation types**
  - Improvements in both baseline and peak memory, as well as parallel memory scaling
  - Clever caching strategies (up to 30-50% memory reduction in gated devices, 5-20% for bulk)
  - Memory distribution of Hamiltonian (both non-selfconsistent and selfconsistent parts)
    - 10% memory reduction for 1000 atom systems
    - Gives better memory parallel scaling (speedup) for large systems
  - Memory-distributed FFT initialization of real-space Hamiltonian (removes significant memory peaks)
- ▶ **Better GGA implementation**
  - Memory usage reduced (previously, it was sometimes the peak memory, especially for systems with a lot of vacuum)
- ▶ **Memory profiler (internal, helps us troubleshoot support cases, may be exposed to users later on)**

### Speed

- ▶ **Significant performance gain for LCAO-MD and geometry optimizations**
  - Parallel distribution of the exchange-correlation calculation among processes
  - Improved algorithms and parallelization strategy for energy, force, stress
  - Result: 5x speedup in EFS, 3.3x in MD (200 atoms, 64 processes, SZP basis set, <1 min/MD step, 1.5 min for spin-polarized); speedup larger for larger basis sets
- ▶ **Better GGA implementation**
  - Performance improvements (up to 3x) for both FFT (periodic in all directions) and stencil method (used for all other boundary conditions)
  - FFT version parallelized

- ▶ **Improved parallelization**
  - Automatic choice of number of processes per k-point for optimal parallel performance
  - Improved parallel scaling (to higher process count) by parallelizing serial bottle necks (neutral atom potential, constant terms, LDA and GGA-FFT exchange-correlation terms, and difference density)
- ▶ **Improved performance of specific algorithms**
  - Parallel Green's function
  - Self-energies stored on disk in parallel – now same performance as SaveInMemory
  - Mobility module, through code redesign
- ▶ **Built-in profiling for device algorithms (sparse vs dense, performance/memory) - helps you choose the best parameters**
  - Example: Quickly determine if SparseGreensFunction is the best choice for a given problem (typically it is, but not always) without actually running the calculation

## Threading

- ▶ **QuantumATK calculations can make use of both MPI parallelization and threading (hybrid parallelization)**
- ▶ **Version O-2018.06 improves on earlier versions by implementing automatic threading intelligence: the number of threads is now automatically controlled to make the best use of the available resources, taking into account the number of MPI processes used**
  - The log file lists the number of threads used by each MPI process
  - In general, the number of processes multiplied by the number of threads per process should equal the total number of available physical processors on the machine for maximum efficiency
- ▶ **If you want to set the number of threads per process to a non-default value, this can be done by exporting the environment variable MKL\_NUM\_THREADS. Threading can be effectively disabled by setting this to 1**
  - Note that OMP\_NUM\_THREADS can also be used instead of MKL\_NUM\_THREADS, although if both are set then MKL\_NUM\_THREADS takes precedence
- ▶ **Footnote: The number of threads per process set by MKL\_NUM\_THREADS will automatically be capped to the total number of physical processors**
  - This check can be disabled by setting MKL\_DYNAMIC to FALSE, although doing so is not recommended
  - OMP\_DYNAMIC is not referenced at all by QuantumATK

## New Study Object Concept

Many atomic-scale modelling tasks require a large number of independent calculations that are combined into a result. The new feature called “study object” takes care of all the bookkeeping in such tasks and allows for easy restarting, storing, and analyzing all data. In O.2018.6, we introduce study objects for charge point defect analysis and different Device configuration tasks (see Device Configuration Updates part).

- ▶ **Charged Point Defect**
- ▶ **IV Characteristics**
- ▶ **Optimize Device Configuration**
- ▶ **Dynamical Matrix**
- ▶ **Hamiltonian Derivatives**
- ▶ **A framework for studying point defects in bulk materials: vacancies, substitutionals, interstitials**
- ▶ **Neutral and charged defects**
- ▶ **Calculate relaxed defect structures, formation energies and thermodynamic transition levels**
- ▶ **Can use with the new PlaneWave calculator**
- ▶ **Can specify different calculators for the relaxation and the calculation of formation energies**
- ▶ **Ghost atoms enabled for vacancy sites, also during relaxation, for significantly better accuracy**
- ▶ **FNV correction scheme for charged defects with automatic Gaussian model charge fitting**
- ▶ **Check convergence of defect properties with cell size and extrapolate to infinite size**

- ▶ Automatic adjustment of k-point sampling for different cell sizes to ensure best possible consistency between sizes
- ▶ Analyzer & Scripter
- ▶ Tutorial: formation energies and transition levels of charged defects

## Device Configuration Updates

Besides the substantial performance improvements described above, there are 2 new study objects: a) automating the steps in a device relaxation, and b) studies of the I-V and gate bias scans. A new technical feature, minimal electrode, automates the setup and repetition of the electrode, so the user no longer needs to adjust the electrode length according to the range of the Hamiltonian but can choose the smallest electrode repeat unit.

### Minimal Electrode Concept

- ▶ Reduced electrode – automatically repeated for computing self-energies
- ▶ Note that the electrode extensions are now visualized
- ▶ Saves time in the electrode calculation which is  $O(N^3)$
- ▶ In 2019 we plan to do the same for A/B repetitions where possible
- ▶ Combined framework for running multiple source-drain/gate voltage calculations and collecting and analyzing the results
- ▶ Set up from Script Generator
- ▶ Multilevel parallelism
- ▶ “Smart restart”
- ▶ Custom settings for individual voltages
- ▶ Analysis
  - Regions highlighted in 3D
  - State persistence (via data in HDF5)
  - Custom drain-source voltages, plot current as function of gate source; current as function of drain-source for one or many gate voltages
  - Show on/off ratio, subthreshold slope, transconductance, DIBL, source-drain saturation voltage
- ▶ Tutorial: [Electrical characteristics of devices using the IVCharacteristics study object](#)

### Optimize Device Configuration Study Object for Relaxation of Devices

- ▶ This study object fully automates the Bulk Rigid Relaxation (BRR) method, making it very simple to optimize the geometry of a device
- ▶ Can relax also complex device systems
- ▶ It includes automatic identification of central region atoms that should be allowed to relax
- ▶ It consists of 4 steps:
  - 1. The region of full relaxation is determined
  - 2+3. Bulk Rigid Relaxation is executed
  - 4. The device configuration is re-assembled
- ▶ The study object allows you to change certain settings for device relaxation:
  - You may select which part of the central region to fully relax, i.e., you may specify the center and width of the relaxation region
  - The relaxation region may extend to the entire device, except of course the electrodes and their extensions
  - If needed, the left and right surfaces of the central region may be hydrogen-passivated during steps 2 and 3
  - The usual force convergence criterion for geometry optimization may also be tuned
- ▶ Tutorial: [Relaxation of devices using the OptimizeDeviceConfiguration study object](#)

## Electron Mobility Module Updates

The electron mobility is limited by electron phonon scattering. Including such effects is very complex and most implementations are time consuming and notoriously difficult to use. In QuantumATK, we strive to make this type of calculations as easily accessible as doing a band structure calculation. With the new improvements to the Mobility module we are getting closer to this goal.

- ▶ **Automated workflows for dynamical matrix (D) and Hamiltonian derivatives (dH/dR)**
  - New advanced checkpointing: if the calculation is interrupted, just submit it again and it will be restarted from where it left off, keeping all previous progress
  - Updated: Linear scaling parallelization by eliminating the zero-displacement calculation
- ▶ **Enable Wigner-Seitz approximation for both D and dH/dR**
- ▶ **Tetrahedron integration method**
  - Converges faster in q-points compared to using Gaussian broadening
- ▶ **T. Markussen *et al.*, Phys. Rev. B 95, 245210 (2017).**
- ▶ **Experiment: Canali *et al.* Phys. Rev. B 12, 2265 (1975).**
- ▶ **Tutorial: [Phonon-limited mobility in graphene using the Boltzmann transport equation](#)**
- ▶ **See also: <http://www.ioffe.ru/SVA/NSM/Semicond/Si/electric.html>**

## Calculation of Spin Parameters

DFT can provide highly accurate materials parameters of spin system. This release of QuantumATK provides the latest perturbation theory methods for calculating spin parameters as easy to use analysis modules.

### Heisenberg Exchange Analysis Module

- ▶ A novel method of computing exchange coupling constants for the Heisenberg model, using the Green's-function method
- ▶ Works within the framework of density functional theory (DFT+U) combined with the LCAO basis set approach
- ▶ The Heisenberg spin-lattice model is an empirical approach to study various magnetic properties at finite temperatures, e.g., to understand phase diagrams, phase transitions, and magnetization dynamics of the magnetic system
- ▶ The main difference from the traditional approach, which is based on total energy calculations for multiple magnetic configurations, is that all the exchange coupling constants can be computed from a single magnetic configuration calculation
- ▶ Easy to set up the HeisenbergExchange analysis such that only couplings between certain atoms are considered

### Spin Life Time

- ▶ Spin life time is an important parameter for spintronics devices
- ▶ At technologically relevant temperatures (>100 K) the spin life time will be limited by electron-phonon interactions, mediated by spin-orbit coupling (Elliot-Yafet mechanism<sup>[1]</sup>)
- ▶ QuantumATK 2018.06 can calculate the phonon-limited spin life time from an ElectronPhononCoupling object (if computed with noncollinear spin and spin-orbit coupling)
- ▶ **Note: Only matrix elements between spin-up and spin-down states are included**
- ▶ **Fast tetrahedron integration in 3D (Gaussian recommended in 1D)**

[1] O. D. Restrepo and W. Windl. Phys. Rev. Lett. 109, 166604 (2012)

## Photocurrent Module

Light can drive an electrical current and this can be calculated with the photocurrent module. Finite temperature effects are included through the Special Temperature Displacement (STD) method. For a case study, check our latest publication: <https://arxiv.org/pdf/1801.03683.pdf>

- ▶ **New analysis module for calculating the photocurrent and photon-mediated transmission in a device using first-order perturbation theory within the 1st Born approximation<sup>[1,2,3]</sup>**
- ▶ **Also gives the total current based on illumination by the AM1.5 standard solar spectrum**
- ▶ **Tutorial: Photocurrent in a silicon p-n junction**

[1] Phys. Rev. B **85**, 155441 (2012)

[2] Journal of Applied Physics **91**, 6273 (2002)

[3] Phys. Rev. B **90**, 195428 (2014)

## Special Quasi-random Structures (SQS) Algorithm for Generating Random Alloys

Random alloys are becoming important for many technological applications. The SQS method generates a single configuration that represents the structure of a Random alloy and is both an accurate and practical approach. With the new built-in SQS generator, it is very easy and fast to generate SQS structures within NanoLab.

- ▶ **Built-in SQS algorithm to generate random alloys**
  - Use a genetic algorithm (unlike other codes that perform an open-ended Monte Carlo simulation, which can be very slow)
  - For now supports 2-component systems like SiGe or InGaAs
  - Any type of geometry, also nanowires etc.
- ▶ **New configuration type: AlloyConfiguration**
  - Fractional occupancy on sites, primarily to be used in the SQS generator

## Major NanoLab Builder Updates

Our popular builder, often quoted as the best materials science builder, comes with a number of important feature updates. A groundbreaking new feature is a log window which will generate a python script that replicates the building steps you perform, thus allowing for automating structure building once you have built a template structure.

### Builder Feature Improvements

- ▶ **Better favorite Builder tools organization**
  - Active item is remembered when you close NanoLab or just the Builder
  - Filter (search) by stash title or configuration type
  - New list view mode (old icon grid still available)
  - Autosave - each time you change active stash item or minimize the Builder
- ▶ **Crystal symmetry**
  - Tolerance for angles and positions can now be tuned in the plugin
  - Warning for large systems (can lock up the backend for very long time)
- ▶ **Spatial regions can now be added/cloned/moved more easily**
- ▶ **Vacuum size can now be set in Surface From Bulk**
- ▶ **Cleaver - choose where the selected atom is placed in the cleave direction (default and earlier only choice is c=1, might not always be desirable)**
- ▶ **Camera/rotation**
  - New default rotation model (center of mass of visible atoms) for better rotation of oddly shaped systems in 3D
  - Added possibility to set rotation center manually
  - More options to align the camera precisely



## New Builder Features

- ▶ Swap indices of two atoms (tool button) – useful for setting up NEB paths with vacancies
- ▶ Configuration information button
- ▶ Make your favorite [Builder tools](#) appear at the top in a separate category
  - Also added a search filter for tools
- ▶ Press “spacebar” in Builder – quick menu for operations appears
- ▶ Drag and drop individual NEB images to the stash
- ▶ The interpolation algorithms (LI-LinearInterpolation, HLC-HalgrenLipscomb, and IDDP-ImageDependentPairPotential) can now be accessed in Python scripts for easier automation of NEB path generation
- ▶ Widget for choosing a specific image when a trajectory (MD or optimization) is imported into the Builder
- ▶ Warnings shown in the Builder if atoms are too close, if the system is too closely packed, spatial regions overlapping atoms, or atoms outside the cell, etc.
- ▶ Builder canvas activates/deactivates when you switch to other tools
  - Means atom selection will not be removed when you click the 3D window to activate the Builder again

## Python Console

- ▶ [Provides direct Python access to interact with the configurations in the Builder](#)
- ▶ Maps (some) operations in the Builder to Python commands
- ▶ [Create pre-defined scripts \(“snippets”\) to automate repeated tasks](#)
- ▶ API still under development, more features will be added!

## Move Tool

- ▶ [Widget moved into the right-hand panel](#)
  - Extended with new functionality to pick and align the rotation vector etc.
- ▶ Also added a “simple move mode”
  - When atoms are selected, they can immediately be moved by the mouse
  - Shift (Ctrl) restrict movement left/right (up/down)
- ▶ In both modes, atoms can also now be moved using the arrow keys
- ▶ [Snap to electrode extension points, passivation sites, etc.](#)

## General NanoLab Updates

Check the long feature list below and see what an amazing job our expanded Nanolab team has performed for this release.

- ▶ Project Dialog - added filter (title and path), last opened date, custom sorting
- ▶ LabFloor load time improved through better logic of which files are checked/unchecked
- ▶ Trajectories: easy access to the “last image” via separate icon on LabFloor
- ▶ [New plugin to compare two configurations](#)
- ▶ Improvements to vibrational analyzer
  - Arrows default, auto-update, only show available modes, length of arrow scaling, preserve bonds
- ▶ FatBS/PDOSAnalyzer
  - Linewidth retained on zoom
  - PDOS: reorder rows to change plot order
  - Same default color ordering in both tools for easier comparison
- ▶ LocalBandstructure
  - Added option to show all bands for better navigation in the band structure
  - Sorting of bands
    - Avoiding spurious renumbering of eigenstates due to band crossings
    - Essential for fitting effective masses in a wider k-point range
    - Also available for EffectiveMass

- ▶ **Viewer: change units that grids are plotted in (isosurface, cutplane, density plot)**
- ▶ **New constraints:**
  - Fix fractional coordinates (pure cell optimization)
  - Cartesian: fixed X, Y, Z, XY, YZ, YZ
- ▶ **Neumann boundary condition with finite electric field value for devices and surfaces**
- ▶ **Fixed magnetic moment (up minus down spin; separate up/down Fermi levels)**
  - Keyword fixed\_spin\_moment to the LCAOcalculator
- ▶ **Transmission spectrum, automatically cover bias range**

## 1D and 2D Plotting

- ▶ **New way to visualize configurations: simplified 2D matplotlib plot**
  - Will later allow to overlay grid data (axis projection) on the configurations
- ▶ **Many minor improvements provide a better overall 2D data plotting experience in NanoLab**
- ▶ **1DProjector**
  - Axis and ProjectionPoint can now be specified in Cartesian coordinates
  - [Macroscopic average plotting tool](#)

## Database Support

- ▶ **Internal database**
  - Metadata separated from data ("mountpoint")
  - Guide for setting up a database is being published

## MD Analyzer

- ▶ **Overlay multiple plots**
- ▶ **Time-series data for e.g. coordination numbers, Q-numbers**
- ▶ **Link to documentation**
- ▶ **Interactive line fitting (e.g. for MSD to extract diffusivity)**
- ▶ **Radial distribution (and other quantities) can now be calculated for single unit cell but with an arbitrarily large cutoff**
- ▶ **Mean square displacement: anisotropy parameter**
  - Also along selected axis (x, y, z)
- ▶ **Change font size of labels, customize linestyle, thickness, colors, etc**
- ▶ **Added Q-number analysis & vibrational DOS from MD trajectory, calculated from velocities**
- ▶ **Vibrational DOS from MD trajectory, calculated from velocities**

## Job Manager

- ▶ **Added support for SGE (Sun Grid Engine)**
- ▶ **PBS machine interface keeps all settings of MPI processes vs. threads vs. available cores in sync and warns about over-subscription**
- ▶ **Add custom commands to be executed before the job environment is set up**
- ▶ **Queueing of jobs on local machine**
  - Local jobs can be set up to run consecutively instead of concurrently (to avoid license and/or CPU overload)
- ▶ **"Local queue"**
  - All remote plugins can be set up to operate on localhost:22
  - This skips the actual network connection and instead submits the job to the queue locally (i.e. the queue is supposed to operate on the current machine)
  - No copying of files - the job is run in the project directory

## Script Generator

- ▶ Set verbosity
- ▶ Clearer logic around how objects are saved to results files
- ▶ Duplicate (clone) script blocks
- ▶ InitialState widget – many new ways to set initial spins
  - Use collinear as shortcut to set up noncollinear
  - Select atoms interactively with mouse or by element, or by tags
  - Define initial spin via expressions
  - Visualization of initial spins as arrows

## Major NanoLab Builder Updates

- ▶ FatBandstructure object for analysis of the spd- and site projected character of the bandstructure is now supported for VASP output (PROCAR)
- ▶ Support for import of QE trajectories
- ▶ Added scripiter for ORCA (molecules)

## Miscellaneous

- ▶ Switch to Synopsys Common License (SCL)
- ▶ Upgrade to PyQt5 (all private plugins will need to be upgraded)
- ▶ Added possibility to combine EAM with other potentials using tags
- ▶ Log file improvements
  - Improved report in case of non-convergence
  - Restart info in DFT+MD written to log file
  - Verbosity levels (introduced in 2017, can now easily be set via Script Generator)
  - Reports unit cell size (molecular calculations)
- ▶ Help>About contains additional information helpful in troubleshooting
  - Install path, username
  - "Copy to clipboard" adds specific environment variables (HOME, LANG, license setup, Python environment)
- ▶ FHI-aims has been removed and is no longer supported or offered
- ▶ Removed CP2K non-selfconsistent Hotbit DFTB sets (generally do not give good results)
- ▶ NetCDF write support removed (replaced by HDF5)

Sep 12 2018

The numbers in parenthesis are our internal issue tracking numbers, listed here just for our own reference.

- ▶ FHI LDA (DoubleZetaPolarized) or SG15 LDA (Medium) basis sets were chosen by default in HybridGGA calculations instead of new default settings, PseudoDojo GGA (Medium) or SG15 GGA (Medium) when using spin-orbit, unless explicitly set in the input file. (26916)
- ▶ Problem with accuracy when computing the DynamicalMatrix, caused by a new parallelization strategy. Solved by using the undisplaced super-configuration as initial state for all displacements (as was the case in the 2017 release), but still retaining the new algorithm which allows the calculation to scale to 6N parallel processes ( $N$ =number of atoms) instead of only 3N in the 2017 release. (27451)
- ▶ Two particle types were switched in TersoffBrennerTriplePotential potentials. (27301)
- ▶ The doping level was calculated incorrectly in a DeviceConfiguration when the doped region was referenced to the left electrode, because internally the compensation charge was always computed based on the volume of the right electrode. (27493)
- ▶ Spin-polarized DOS spectra calculated with QuantumEspresso could not be plotted. (26837)
- ▶ Transmission analyzer incorrectly plotted transmission vs.  $k_A, k_B$  in cases where the k-point list was not ordered. (27492)
- ▶ SpecialThermalDisplacement gave an error if used with devices that have minimal electrodes, because it did not properly respect the repeated electrode extension in the central region. (27227)
- ▶ The resulting configuration saved after running EvolutionarySQS could not be read back from HDF5 files due to missing serialization support. (27488)
- ▶ Graphics glitches in the 3D engine could make the configuration appear empty in certain situations. (25329, 26168, 26208, 25301)
- ▶ Some Materials Project searches resulted only in error messages, due to newly introduced fields in the database which could be empty, which tripped up our parser. (27124)
- ▶ QuantumATK could crash when running in parallel if a user gave a specific filename for the checkpoint file, due to an MPI race condition. (27481)
- ▶ Result files generated with absolute paths for checkpoint files could not be opened in NanoLab. (27478)
- ▶ Selection of automatic number of threads in the Job Manager (PBS and Slurm plugins for remote execution) produced an unnecessary warning. (26505)
- ▶ Files containing unicode (in case special characters were used in usernames or folders) were not correctly copied back from the server. (26505)



### Synopsys QuantumATK Team

Fruebjergvej 3  
DK-2100 Copenhagen  
DENMARK  
Tel: +45 333 32 300  
Email: [quantumatk@synopsys.com](mailto:quantumatk@synopsys.com)

**SYNOPSYS**  
Silicon to Software™

©2018 Synopsys, Inc. All rights reserved. Synopsys is a trademark of Synopsys, Inc. in the United States and other countries. A list of Synopsys trademarks is available at <https://www.synopsys.com/copyright.html>. All other names mentioned herein are trademarks or registered trademarks of their respective owners.  
06/28/18.snSheet\_qatk2018features.indd