

Release notes for RESCU-2.0.5

1. Improved real space solver. Notably, non-collinear spin solvers have been ported to GPUs and the efficiency of the CFSI algorithm for GPUs has been improved by a factor 2-6 compared with rescu-2.0.4 (cuBLAS required). An improved restarting scheme makes CFSI well-suited for non-self-consistent calculations. LOBPCG is restarted using a “sliding window” scheme which improves its robustness. Hamiltonian-wavefunction products are optimized using an improved planning algorithm, accounting for geometry, hardware and threading.
2. Improved numerical atomic orbital solver. Improved sparse matrix communication routines improve efficiency generally. Orbital interpolation has been optimized. The memory requirements associated with the calculation of the density matrix, the density and the forces are lowered such that larger systems can be simulated.
3. Extend DFT+U to the non-collinear spin treatment with the numerical atomic orbital framework.
4. New option `info.calculationType`; the “potential” calculation allows recalculating various potentials from an input density.
5. New option `domain.boundary`; open boundary conditions are available for wire geometries (i.e. 1D periodic + 2D open).
6. New option `dos.ldosERange` allows calculating the local density of states over a certain energy range.
7. New option `kpoint.sampling` accepts “Marzari-Vanderbilt”. This smearing scheme prevents negative occupancies unlike Methfessel-Paxton.
8. Fix a bug affecting relaxation calculations. Simply put, the relaxation procedure will not continue unless self-consistency is duly achieved.
9. Fix a bug affecting the model core charge forces. Atomic forces, equilibrium structures and phonon band structures should be generally improved for systems with pseudopotentials including a model core charge. **This bug may influence previous results.**

10. Fix a bug affecting the LDA+U energy and potential. **This bug may influence previous results.**

11. Fix a bug affecting the occupancies calculated using the Methfessel-Paxton scheme. **This bug may influence previous results.**

Note: Unless otherwise specified, bug fixes do not influence previous results.