What's New in Gaussian 16

New Modeling Capabilities

- [REV C] NBO version 7 is supported. There are new options to the Population keyword: Pop=NPA7, Pop=NBO7, Pop=NBO7Read and Pop=NBO7Delete request Natural Population Analysis, full Natural Bond Orbital Analysis, full NBO with NBO input read from the input stream and NBO analysis of the effects of deletion of some interactions (respectively), using NBO7 via the external interface. In addition, Pop=NEDA is used to perform Natural Energy Decomposition Analysis. The analysis uses the same input information about fragments as counterpoise calculations. Deletions and optimizations with deletions now work with either NBO6 or NBO7.
- [REV C] The RESP (restrained electrostatic potential) constraint can be included in computing potential-derived charges. For example, Pop=(MK,Resp=N) applies a weight of N x 10^{-6} Hartrees to the squared charges. Other electrostatic potential-derived charge schemes also accept this option (e.g., CHelp, HLY). N defaults to 2.
- [REV C] Pop=SaveHirshfeld and Pop=SaveCM5 cause the specified charges to be saved as the MM charges to be used in a subsequent calculation.
- [REV B] Static Raman intensities can be computed for excited states at the CIS and TD levels of theory. TD Freq=Raman computes the polarizability by numerical differentiation with respect to an electric field, so the cost of Freq=Raman for these methods is 7x that of the frequencies without Raman intensities.
- TD-DFT analytic second derivatives for predicting vibrational frequencies/IR and Raman spectra and performing transition state optimizations and IRC calculations for excited states.
- EOMCC analytic gradients for performing geometry optimizations.
- Anharmonic vibrational analysis for VCD and ROA spectra: see Freq=Anharmonic.
- Vibronic spectra and intensities: see Freq=FCHT and related options.
- Resonance Raman spectra: see Freq=ReadFCHT.
- New DFT functionals: M08HX, MN15, MN15L, PW6B95, PW6B95D3.
- New double-hybrid methods: DSDPBEP86, PBE0DH and PBEQIDH.
- PM7 semi-empirical method.
- Ciofini excited state charge transfer diagnostic: see Pop=DCT.
- The EOMCC solvation interaction models of Caricato: see SCRF=PTED.
- Generalized internal coordinates, a facility which allows arbitrary redundant internal coordinates to be defined and used for optimization constraints and other purposes. See Geom=GIC and GIC Info.

Performance Enhancements

- NVIDIA K40, K80, P100 (Pascal) and V100 (Volta) GPUs are supported under Linux for Hartree-Fock and DFT calculations. V100 support is new with [REV C], and P100 support was new with [REV B]. Both revisions also provide performance improvements for all GPU types. See Using GPUs for details on GPU support and usage.
- Parallel performance on larger numbers of processors has been improved. See the Parallel Performance tab for information about how to get optimal performance on multiple CPUs and clusters.
- [REV B] Dynamic allocation of tasks among Linda workers is now the default, improving parallel efficiency.

- Gaussian 16 uses an optimized memory algorithm to avoid I/O during CCSD iterations.
- There are several enhancements to the GEDIIS optimization algorithm.
- CASSCF improvements for active spaces ≥ (10,10) increase performance and make active spaces of up to 16 orbitals feasible (depending on the molecular system).
- Significant speedup of the core correlation energies for W1 compound model.
- Gaussian 16 incorporates algorithmic improvements for significant speedup of the diagonal, second-order self-energy approximation (D2) component of composite electron propagator (CEP) methods as described in [DiazTinoco16]. See EPT.

Usage Enhancements

- [REV C] The ROA invariants for each vibrational mode are now only printed by G16 or by freqchk if normal mode derivatives were requested, rather than by default.
- [REV C] Utilities can now take the -m command-line argument to specify the amount of memory available to the utility. For example:

  formchk -m=1gb myfile

  The -m option must precede any file name or other arguments.

- [REV C] The %SSH Link 0 command and its equivalents can be used to name a command to run to start Linda workers, rather than either rsh or ssh.
- [REV C] Some defaults when Geom=AllCheck is specified can now be overridden:
  - Field=NoChk can be used to suppress reading external field coefficients from the checkpoint file.
  - Geom=GenConnectivity forces the connectivity to be recomputed rather than using the information in the checkpoint file.
  - Geom=UseStandardOrientation uses the coordinates in the standard orientation from the checkpoint file as the input orientation for the new job.
- [REV C] Some defaults during geometry optimizations to a minimum can now be overridden:
  - Opt=NGoUp=N allows the energy to increase N times before doing only linear searches. The default is 1 (only linear searches are performed after the second time in row that the energy increases); N=-1 forces only linear searches whenever the energy rises.
  - When near a saddle point, Opt=NGoDown=N causes the program to mix at most N eigenvectors of the Hessian with negative eigenvalues to form a step away from the saddle point. The default is 3; N=-1 turns this feature off, and the algorithm takes only the regular RFO step.
  - Opt=MaxEStep=N says to take a step of length N/1000 (Bohr or radians) when moving away from a saddle point. The default is N=600 (0.6) for regular optimizations and N=100 (0.1) for ONIOM Opt=Quadmac calculations.
- [REV C] Information on multidimensional relaxed scans is now stored on the formatted checkpoint file with details about the axes, rather than flattened, so these can be displayed in GaussView and other programs.
- [REV C] The program now stores and checks a version number in checkpoint files. This avoids obscure failure modes when an obsolete checkpoint is named. The c8616 utility can be used to update checkpoint files, and there is a -fixver option to unfchk to mark a checkpoint file it creates as current even if there was no version in the input formatted checkpoint file.
- [REV B] The ChkChk utility now reports the job status (whether the job completed normally, failed, is in progress, etc.)
- [REV B] The optional parameters in the input line for an atom can now specify the radius to use when finite (non-point) nuclei are used. The radius is specified as a floating point value in atomic units using the RadNuclear=val item. For example:

  C(RadNucl=0.001) 0.0 0.0 3.0

- Tools for interfacing Gaussian with other programs, both in compiled languages such as Fortran and C and with interpreted languages such as Python and Perl. Refer to [Interfacing to Gaussian 16](#) for details.
    - [REV C] supports raw binary files using either 4- or 8-byte integers. The former is the default except on NEC systems. Support for this feature includes new options to the Output keyword and the formchk utility, new Link 0 commands and new command line options and environment variables.
    - [REV C] adds information about ONIOM layers and optimization and trajectory results to the matrix element file. It also adds new options to the Output keyword for including AO two-electron integrals, derivatives of the overlap, core Hamiltonian and other matrices and/or the AO 2-electron integral derivatives.
    - [REV B] added many additional quantities to the matrix element file, including atomic populations, one-electron and property operator matrices and the non-adiabatic coupling vector. The new items are the labeled sections QUADRUPOLE INTEGRALS, OCTOPOLE INTEGRALS, HEXADECAPOLE INTEGRALS, [MULLIKEN,ESP,AIM,NPA,MBS] CHARGES, DIP VEL INTEGRALS, R X DEL INTEGRALS, OVERLAP DERIVATIVES, CORE HAMILTONIAN DERIVATIVES, F(X), DENSITY DERIVATIVES, FOCK DERIVATIVES, ALPHA UX, BETA UX, ALPHA MO DERIVATIVES, BETA MO DERIVATIVES, [Alpha,Beta] [SCF,MP2,MP3,MP4,CI Rho(1),CI,CC] DENSITY and TRANS MO COEFFICIENTS and the scalars 63-64.
- [REV C] Enhancements to facilitate scripting:
    - The AllAtoms and ActiveAtoms to the External keyword are used to provide information on all atoms or only those in the model system (high layer) when using an external program/script with ONIOM.
    - The file $g16root/g16/bsd/inp2mat is a script which takes a Gaussian input file and generates a matrix element file with the information implied by the input file (coordinates, basis set, etc.) without running the full calculation. This is used by the Python interface in GauOpen to import this information into a matrix element file object, but can also be used in other scripts to avoid any need to parse Gaussian input files.
    - The testrt utility now prints the integer size used by G16 so that scripts can check what size of integers will be used by default in matrix element files.
- Parameters specified in Link 0 (%) input lines and/or in a Default.Route file can now also be specified via either command-line arguments or environment variables. [REV B] introduces command-line options to specify input and/or data using a checkpoint or matrix element file (the equivalent of the %OldChk or %OldMatrix Link 0 commands for input). See the [Equivalencies](#) tab for details.
- You can now compute the force constants at every nth step of a geometry optimization: see [Opt=Recalc](#).
- [REV B] DFTB parameters are now read in Link 301 before the basis set is constructed, so that the presence or absence of d functions for an element can be taken from the parameter file