***ANT 2023* abstract.**

ANT ("Adiabatic and Nonadiabatic Trajectories") is a Fortran 90 trajectory program designed primarily for classical, quasiclassical, and semiclassical calculations of the dynamics of atoms, molecules, and clusters in the gas phase.

ANT is especially recommended for calculations that employ analytic potential energy surfaces. It also has direct dynamics capability although it is not the most convenient code for direct dynamics.

- *ANT* can be used for dynamics governed either by a single potential energy surface (electronically adiabatic processes) or by two or more coupled potential energy surfaces (electronically nonadiabatic processes).
- For an electronically adiabatic process, the user must supply a potential energy surface and its gradients. The dynamics routines obtain this either from a user-supplied analytic potential energy surface subroutine or by direct dynamics. In direct dynamics, the is obtained by performing new electronic structure calculations whenever the dynamics algorithm needs potential information.

  For an electronically nonadiabatic process the user must supply two or more surfaces and their gradients. For most of the nonadiabatic dynamics algorithms (all those except the ones lapelled curvature-driven), couplings between electronic states are also required. When using an analytic potential surface this can be provided in the diabatic or adiabatic representation. When using direct dynamics, either representation may be used (although most electronic structure packages will only provide information in adiabatic representations). To use the electronically adiabatic representation for electronically nonadiabatic dynamics, the user may either provide the adiabatic surfaces and nonadiabatic couplings by direct dynamics, or the program may calculate them from diabatic surfaces and diabatic couplings, which are usually analytic. One can also use analytic fits to the surfaces and couplings to carry out calculations entirely in the diabatic representation.

  For direct dynamics, the code has well tested interfaces to the *Gaussian 09*, *Gaussian 16*, *MOPAC-mn, and Molpro* electronic structure codes. The interfaces are included, but these electronic structure programs must be obtained separately. Users may also provide their own interfaces to other electronic structure programs.
- For electronically adiabatic processes, the code can perform either classical or quasiclassical trajectories.

  For electronically nonadiabatic processes, the following semiclassical trajectory options ae available:
  - surface hopping by with Tully's fewest-switches (TFS) algorithm,
  - surface hopping by the fewest switches with time uncertainty (FSTU) algorithm,
  - FSTU with stochastic decoherence (FSTU/SD),
  - mean-field methods (also called self-consistent-potential methods):

    - the semiclassical Ehrenfest (SE) method,
    - coherent switching with decay of mixing (CSDM),
    - curvature-driven coherent switching with decay of mixing (κCSDM)
    - self-consistent decay-of-mixing algorithm (SCDM)
- *ANT* can calculate
    - bimolecular reactive collisions
    - inelastic collisions
    - unimolecular processes

  with various initial conditions. It can calculate cross sections and rate constants.

  ANT can be run at fixed energy with various initial conditions or for thermal ensembles. Another option is that one may begin trajectories at a dividing surface passing through a saddle point as used for unified dynamical model calculations.

  An especially well-developed feature of the program is initial state preparation for quasiclassical atom–diatom and diatom–diatom collisions.

  A limited set of final-state analysis options is available. Or one can let program write initial and final coordinates and momenta and selected other information to a file for external (post-trajectory) analysis.
- Quantum mechanical enhancements:
    - The army ants tunneling algorithm is implemented as an option for both electronically adiabatic and electronically nonadiabatic trajectories on any unimolecular process. For electronically adiabatic processes, it may be used with classical or quasiclassical trajectories. For electronically nonadiabatic processes, it is only implemented for mean-field methods. The tunneling path can be any valence internal coordinate or any combination of two stretch coordinates.
    - Three methods (TRAPZ, mTRAPZ, and mTRAPZ* methods) are available to ensure zero-point energy maintenance in classical trajectory simulations, if desired.
- Additional capabilities:
    - The program can handle periodic boundary conditions (cubic or cuboid only) if a periodic potential is given.
    - The program can also optimize geometry by following a steepest-descents trajectory in Cartesian coordinates.