**NAME**

**surfgen**, **surfgen.global** – fit quasi-diabatic Hamiltonian from ab initio data.

**SYNOPSIS**

**surfgen**

**DESCRIPTION**

**surfgen** applies the quasi-diabatic Hamiltonian(Hd) approach to construct coupled potential energy surfaces from ab initio data. Evaluation libraries are also provided to allow spectroscopy and dynamics simulation programs to access fit potentials, either in diabatic or adiabatic representation.

Here, *quasi-diabatic* means the residual coupling, the coupling between diabatic states, is minimized in a least squares sense in the fitting procedure. The diabatic representation used by Hd is therefore the most diabatic representation in a leaset squares sense.

Developed by Yarkony group, Johns Hopkins University 2010-2013.

**FEATURES**

*Fitting Ab Initio Data*

The program is capable of fitting energy, energy gradient and derivative coupling (AKA nonadiabatic coupling or vibronic coupling) data obtained from ab initio calculations. A weighed least squares procedure is used to generate the fit to simultaneously optimize the reproduction of adiabatic energies and energy gradients and the diabaticity of the quasi-diabaticrepresentation, defined by the residual coupling between diabatic states. Lagrange multipliers are used to enable the exact reproduction of arbitrary selected set of data, such as energy and gradients at critical points on the potential or energy difference and derivative couplings at points of conical intersections.

*Nonadiabatic couplings and Seams of Conical Intersections*

The Hd approach is capable of extremely accurate description of nonadiabatic interactions. It has been used to successfully describe large portions of the seam of conical intersections, spanning completely different geometrically structures. This is enabled by the application of intersection adapted coordinates.

*Flexibility* The program allows the user to define the blocks of the matrix Hd with a large set of customizable basis functions, enabling the flexibility to describe complex features on the surface to a high level of accuracy.

*Global Symmetry Treatment*

Projection operator method is used to allow the program to use an arbitrary subgroup of the Complete Nuclear Permutation Inversion group to construct symmetry adapted basis for the fitting procedure. With the help of such feature, the program can correctly treat the symmetry in problems that involve large amplitude motions, as well as vibrational problems.

*Efficiency* With the fully analytical form of Hd, the evaluation time for a single point is usually within 50ms. Future update is planned to allow vectorized evaluation of large number of data points. The fitting program achieve high efficiency through the extensive use of optimized and threaded LAPACK and BLAS libraries. Other functionalities such as automatic local internal coordinate construction, automatic null space removal and GDIIS extrapolations provide tools to enhance performance.

**ENVIRONMENT**

`stacksize` In order to allow smooth execution, please remove the limit for stacksize. use

**limit stacksize unlimited**

in csh or tcsh, or

**ulimit -s unlimited**

in sh or bash

**FILES**

`surfgen.in` General input for the program, which specifies job specifications such as job type, fitting basis and expansion, molecule composition, state symmetry etc. This file also includes options for each type of jobs such as optimization and evaluation. For a detailed list of parameters, see `surfgen.in`(1)

`surfgen.out` Standard output file that contains the statistics of the data points, Hd expansion and fitting equations, followed by brief summary (1 line per iteration) of the fitting procedure, and analysis of the final fit.

`stdout` Detailed information are printed to the standard output, including details about coordinates, symmetry, expansion, local coordinate construction, null space removal, iterative procedure etc. The amount of output can be controlled with parameter `printlvl` in `surfgen.in`.

`hd.data` Default name for the storage file of the quasi-diabatic Hamiltonian(Hd). The name of the file can be specified in `surfgen.in`. Currently only expansion coefficients is stored in `hd.data`. The expansion is instead defined in `surfgen.in`. For a detailed description of the format of Hd storage files, see `hd.data`(1)

`points.in` Point specification file for surface fitting. Specifies special fitting options for individual data points or a set of individual data points. Currently the options include

1. Exclusion of energies, gradients or derivative couplings for a specific state or block
2. Enforcing exact reproduction of energies, energy difference, gradients or derivative couplings of a specific state or block through the use of Lagrange multipliers.
3. Changing the weight in for a set of data points in the fitting equations. The weights can be an increase or decrease from the default weight of 1.0, or be set to zero.

For details of the format of file `points.in`, see `points.in`(1)

This file is only required for surface fitting jobs.

`coords.in` Coordinate definition file. The file defines a set of internal coordinates that will be used as basis for the fitting program. The coordinates are defined by specifying the *types* and *scaling modes* of coordinate, as well as the atoms involved in the coordinate and scaling parameters for certain modes. The maximum order allowed for each coordinate and *additional order constraints* are also defined in this file. For a detailed description of input format and list of coordinate types and scaling modes provided by the program, see `coords.in`(1)

Note that the coordinate defined here serves as global basis and this may be, and in general will be, redundant at a specificy geometry. Change in this file will change the expansion and therefore will require `hd.data` file to be regenerated.

`irrep.in` Representation definition file. This file defines the representation *matrices* (not just characters) for each irreducible representation of the *permutation group* used by the fitting procedure. Note that inversion symmetry is specified separately in `surfgen.in`. The user has to generate all irreducible representation matrices. This feature may be added in the future.

The user is responsible for the validation of the representations, otherwise the result may be unreliable. For a list of symmetry operations, call **surfgen** with *jobtype=-1*. For details about the format of this file, see `irrep.in`(1)

`connect.in` Connectivity defintion file. This file defines the bonds that will not be broken throughout the scope of Hd. The name of this file can be changed in *surfgen.in*.

A set of feasible permutations will be selected based on the perservation of the connectivity matrix defined by this file. As a result, this file simultanesouly defines the fragments that will remain intact and a subgroup of the full CNPI group that will be used to construct the surface.

This feature is used to enable mixed local and global type of expansions to significantly cut down the number of coordinates, making large problems tractable. Note that change in this file will change the symmetry group and therefore will require regeneration of *irrep.in* and *hd.data* files. For a detailed description of the format of this file, see `connect.in`(1)

*Ab initio Data* Ab initio data input files for surface fitting program. All data points are arranged in to data groups. Each group of points has a number of input files. In each input file, one specific type of ab initio data for all data points are appended together. For gradients and derivative couplings, components must be given in Cartesian coordinates.
All the search paths are given in SearchPaths input option in `surfgen.in`. The program will search each of the search paths for a set of input files. **Note that you will have to include '.' in the search paths list to let the program search it.** The default SearchPath has only current directory. The naming pattern for each input file can be changed through changing the corresponding options in `surfgen.in`
For gradients and couplings, these naming patterns use character '$' as a wildcard character which stand for state index. For example, gradient pattern *grd$* would result in a file name which is characters grd appended by the state index.
Use *COLUMBUS* input/output format for all data files. A list of default input file names include

`note` A note to remind yourself what these data are. The first line of it will also be written to standard output when reading displacement data. Change the naming pattern by setting noteptn option.

`geom.all` Cartesian geometry file for all data points. Change the naming pattern by setting option *gmfptn*.
*COLUMBUS* format is used, where each line contains atom name (character*3), nuclear charge (f), cartesian coordinates (3f) and atomic weight (f). Only the coordinate information is used in the fitting process. Data for different geometries are appended together.

`energy.all` Energy information file. All energies are given in hartree. If the file does not contain energies of all the states, use the following syntax in the first line:
STATES 2 4
In this example, each line of the energy file contains 4 energy data, from states 2, 3 and 4. When this is omitted, the program will assume all *nstates* energies are present each line. The energies are given in floating point or scientific format. The number of states is specified in *surfgen.in*. The number and ordering of data points must be consistent with *geom.all*. Data for all points and states must be present. If an energy data is absent, put in any number (for example, .0) then specify it to be ignored in *points.in*.

`cartgrd.drt1.state$.all`
: Adiabatic energy gradient data for one particular electronic state. Each line contains three floating point or scientific numbers specifying the component of the energy gradient vector on one atom. The data for all atoms and geometries are appended with the same order as in *`geom.all`*. Data for all atoms must be present. Empty lines are ignored. To change the naming pattern, use option *grdfptn*.

`cartgrd.nad.drt1.state$.drt2.state$.all`
: Derivative coupling data of all data points between two specific states. The format is identical with energy gradient input files. One file has to be given for each pair of states. The "_total" files contain the actually derivative coupling data, which approach infinity at conical intersections, while the data in ".nad" files are obtained by multiplying the derivative couplings by energy difference, i.e. the matrix element of the derivative of the Hamiltonian operator, which does not experience singularity. These files are consistent with *`COLUMBUS`* output . To change the naming pattern, use cpfptn option in `surfgen.in`

## SEE ALSO

`connect.in`(1), `coords.in`(1), `irrep.in`(1), `hd.data`(1), `points.in`(1), `surfgen.in`(1),

## REFERENCES


**Quadratic local Hd**

Michael Schuurman and David Yarkony, "On the vibronic coupling approximation: A generally applicable approach for determining fully quadratic quasidiabatic coupled electronic state Hamiltonians", *J. Chem. Phys.*, 127, 094104, 2007.
http://dx.doi.org/10.1063/1.2756540

**Generalized algorithm for global problems**

Xiaolei Zhu and David Yarkony, "Toward eliminating the electronic structure bottleneck in nonadiabatic dynamics on the fly: An algorithm to fit nonlocal, quasidiabatic, coupled electronic state Hamiltonians based on ab initio electronic structure data", *J. Chem. Phys.*, 132, 104101, 2010.
http://dx.doi.org/10.1063/1.3324982

**Example: NH3 photodissociation branching ratio**

Jianyi Ma, Hua Guo, Xiaolei Zhu, and David Yarkony, "First principles determination of the NH2/ND2(,) branching ratios for photodissociation of NH3/ND3 via full-dimensional quantum dynamics based on a new quasi-diabatic representation of coupled ab initio potential energy surfaces", *J. Chem. Phys.*, 137, 22A541, 2012.
http://dx.doi.org/10.1063/1.4753425


## BUGS

Please send bug reports to Xiaolei Zhu ⟨virtualzx@gmail.com⟩