ezMode User Guide

I. INPUT FILES

Input files accept a set of keywords in the format **Keyword=** *value* with a space appearing between the '=' and the value. A description of each possible keyword and it's acceptable values appears below. Some non-sensible mixtures of values will be ignored and replaced with sensible values, other mixtures, however, may cause the program to fail or produce false results. Care should be exercised when preparing input files!

PES_type= *number* [Required]

PES_type takes an integer which defines the type of potential energy surface, i.e., the multiplicities of the atoms involved. A list of the corresponding number and multiplicites is provided at the end of this guide.

PES_dir= *path* [Required]

PES_dir gives the location of the coefficient files for the indicated potential energy surface.

DMS= *yes/no* [Optional]

DMS specifices whether a dipole moment surface also exists and also controls whether or not transition dipole moments will be calculated. Set to *yes* only when the surface exists and the transition dipole moments are desired, otherwise it defaults to *no*.

direct_dipole= *yes/no* [Optional]

direct_dipole controls whether the full dipole matrices are stored or whether the elements are calculated on the fly. Setting to *no* will significantly increase the memory requirements but will run faster, while *yes* is the preferred setting for large jobs.

N= *number* [Required]

N defines the number of atoms.

geometry= *newline* [Required]

atom mass x y z

geomtery defines the reference geometry used in the calculations. The geometry will be rotated to it's principal axes, but will otherwise be used as is. No optimization is performed. The format for input is **geometry=** on the first line followed by N lines of consisting of *atom label*, *mass* in atomic units, x y z Cartesian coordinates in Angstroms of space delimited values.

V_nmode= *number (1-4)* [Required]

V_nmode defines the level of N-mode representation used for the potential. Currently levels of 1 to 4 are supported.

mu_nmode= *number (0-3)* [Required]

mu_nmode defines the level of N-mode representation used for the inertia tensor and coriolis coupling. Set to 0 this totally disables coriolis coupling, at 1 in includes only the so-called "Watson" term, and at 2 and higher it includes the appropriate vibrational angular momentum terms.

grid= *number* [Required]

grid defines the number of grid points used in the primitive 1-d representation along each normal mode.

wh_modes= *number list* [Required]

wh_modes takes as input a space delimited list corresponding to the modes to couple in the VCI calculation. The modes are labelled in order of increasing energy (1, 2, ..., 3N-6). The 6 lowest frequency modes, generally corresponding to translation/rotation, are labelled in increaseing order as (-6, ..., -1). To run a calculation of a system with $N = 5$ starting from a first-order saddle point, the appropriate input would be **wh_modes=** *-6 2 3 4 5 6 7 8 9*.

max_ex= *number list* [Required]

max_ex takes a space delimited list corresponding to the maximum number of quanta possible for each mode.

tot_ex= *number* [Required]

tot_ex specifies the total number of quanta that can be distributed among the available moes.

num_state= *number* [Required]

num_state specifices the number of vibrational states to calculate.

coeff_threshold= *number* [Required]

coeff_threshold defines the number of contributing basis functions that will be printed out for each vibrational level. Starting from the leading coefficients, they will be printed in decreasing order until $\sum_i c_i^2 \geq coeff\_threshold$.

calc_DM_num= *amount list* [Optional]

calc_DM_num specifies for which states to calculate 1-mode density matrices and to print

expectation values and transition dipole moments for. The input is a single number specifying the number of states for which these properties are calculated followed by a space delimited list of the desired states. Optionall the keyword *all* may be specified to print properties for all calculated states.

calc_2DM_num= *amount list* [Optional]

calc_2DM_num specifies the states for which to calculate 2-mode density matrices. The input format is the same as for **calc_DM_num**.

print_density_matrix= *yes/no* [Optional]

print_density_matrix specifices whether printing of density matrices, in a format which is suitable for GnuPlot, should be performed.

TABLE I: PES_type values.

multiplicity	number
x2	0
x1y1	1
x3	2
x2y1	3
x1y1z1	4
x4	5
x3y1	6
x2y2	7
x2y1z1	8
x5	9
x4y1	10
x3y2	11
x3y1z1	12
x2y2z1	13
x6	14
x5y1	15
x4y2	16
x4y1z1	17
x3y3	18
x3y2z1	19
x2y2z2	20
x7	21
x6y1	22
x5y2	23
x5y1z1	24
x4y3	25

TABLE II: PES_type values (cont.).

multiplicity	number
x4y2z1	26
x3y3z1	27
x3y2z2	28
x8	29
x7y1	30
x6y2	31
x6y1z1	32
x5y3	33
x5y2z1	34
x4y4	35
x4y3z1	36
x4y2z2	37
x3y3z2	38
x9	39
x8y1	40
x7y2	41
x7y1z1	42
x6y3	43
x6y2z1	44
x5y4	45
x5y3z1	46
x5y2z2	47
x4y4z1	48
x4y3z2	49
x3y3z3	50