

MQCT User Guide

by Alexander Semenov,^{1,‡} Bikramaditya Mandal[‡] and Dmitri Babikov²

*Chemistry Department, Wehr Chemistry Building, Marquette University,
Milwaukee, Wisconsin 53201-1881, USA*

Appendix A. Input Parameters for MQCT calculations

Three tables below give a comprehensive list of all required and optional input parameters for three blocks of the program input file: blocks \$BASIS, \$SYSTEM, and \$POTENTIAL. Default values are indicated, where applicable. A keyword has to be specified only if the value different from the default is desired. Datatype “real” corresponds to double precision. The values “YES” and “NO” correspond to logical datatype.

Special note for Table A3: For Gauss-Legendre integration of matrix elements user must indicate the number of quadrature points for each degree of freedom, but this depends on the system type. For all molecule + atom systems (SYS_TYPE=1 to 4) only one integer value should be assigned to each of GRD_VIB, GRD_ANG1, GRD_ANG2, and GRD_ANG3. But for all molecule + molecule systems (SYS_TYPE=5 to 0) two integer numbers should be given sequentially, separated by coma. Note, however, that depending on the system type, some of these numbers are dummy (not used) and can be arbitrary. Details are given in Appendix C below.

¹ Present Address: Department of Chemistry, University of Pennsylvania, Philadelphia, Pennsylvania, 19104.

² Author to whom all correspondence should be addressed; electronic mail: dmitri.babikov@mu.edu

[‡] These authors made equal contributions.

Table A1: Description of keywords for the block \$BASIS

Keyword	Type, Range, Units	Description	Relevant SYS_TYPE
SYS_TYPE	integer, 0 to 9	1 -- rigid diatom + atom 2 -- vibrating diatom + atom 3 -- symmetric top + atom 4 -- asymmetric top + atom 5 -- diatom + diatom (both rigid) 6 -- vibrating diatom + vibrating diatom 7 -- symmetric top + diatom (rigid) 8 -- asymmetric top + diatom (rigid) 9 -- asymmetric top + symmetric top 0 -- asymmetric top + asymmetric top	
NMB_CHNLS	integer	Number of channels	all
CHNLS_LIST	integers	Quantum numbers to specify channels	all
INIT_CHNL	integers	Quantum numbers to specify the initial channel	all
EXCLUDE_STATES	default is "NO"	User can exclude specified states from the basis set, e.g., the weakly coupled states or the states with certain j and m .	all
BE, DE	real, positive, cm^{-1}	Rotational constants B_e, D_e (exp. format for D_e)	1,2,7,8
A, B, C	real, positive, cm^{-1}	Rotational constants A, B, C (for x, y and z)	3,4,7,8
BE1, DE1	real, positive, cm^{-1}	Rotational constants B_e, D_e for molecule #1	5,6
BE2, DE2	real, positive, cm^{-1}	Rotational constants B_e, D_e for molecule #2	5,6,7,8
A1, B1, C1	real, positive, cm^{-1}	Rotational constants A, B, C for molecule #1 (for x, y and z)	9,0
A2, B2, C2	real, positive, cm^{-1}	Rotational constants A, B, C for molecule #2 (for x, y and z)	9,0
WE, XE	real, positive, cm^{-1}	Vibrational constants ω_e, x_e	2
WE1, XE1	real, positive, cm^{-1}	Vibrational constants ω_e, x_e for molecule #1	6
WE2, XE2	real, positive, cm^{-1}	Vibrational constants ω_e, x_e for molecule #2	6
JMIN, JMAX	integer	The range of rotational number j included in the basis set; optional, used to avoid listing all levels individually	1,2
VMIN, VMAX	integer	The range of vibrational number v included in basis set; optional, used to avoid listing all levels individually	2

JMIN1, JMAX1, JMIN2, JMAX2	integer	The range of rotational numbers included in basis sets for molecules #1, 2; option, used to avoid listing the levels	5,6
VMIN1, VMAX1, VMIN2, VMAX2	integer	The range of vibrational numbers included in basis sets for molecules #1, 2; option, used to avoid listing the levels	6
NCHL1, NCHL2	integer	Number of lower energy channels included for molecules #1, 2; optional, used to avoid listing the levels individually	5,6,7,8,9,0
EMAX	real, cm ⁻¹	Channel energy cut-off; only the states below it are included in calculations; option, to avoid listing all levels	all
EMAX1, EMAX2	real, cm ⁻¹	Channel energy cut-offs for molecules #1 and #2; only the states below this energy are included in calculations	5,6,7,8,9,0
SYMMETRY	default is "NO"	If "YES", only the states coupled to the initial state are retained in the basis (ortho vs para states).	3,4,7,8,9,0
ATOMIC_MASSES	real, positive, amu	Masses of atoms in the diatomic; to determine COM	2,6
MORSE_DEPTH	real, positive, cm ⁻¹	Depth parameter of Morse oscillator (dissociation energy)	2,6
MORSE_WIDTH	real, positive, Bohr	Width parameter of Morse oscillator	2,6
MORSE_POSITN	real, positive, Bohr	Equilibrium Distance of Morse oscillator	2,6
RMIN_VIBGRID	real, positive, Bohr	Minimum diatomic bond length	2,6
RMAX_VIBGRID	real, positive, Bohr	Maximum diatomic bond length	2,6
WGHT_POSPAR	real, positive, ≤1.d0	Weight of positive total parity wave functions in the case of identical particles collision. Default is 1.d0	5,6,0
CHNL_ENERGS	real, cm ⁻¹	User can list energies of states (e.g., computed externally)	all
LEVELS_FILE	default is "NO"	If "YES", user provides energies, wave functions and assignments of externally-computed states, in a file	all
CS_APPROX	default is "NO"	If "YES", the coupled-states approximation will be used	all
IDENTICAL	default is "NO"	If "YES", collision partners are treated as indistinguishable	5,6,0
PRINT_STATES	default is "NO"	If "YES", prints out states and structure of coupling matrix	all

Table A2: Description of keywords for the block \$SYSTEM

Keyword	Type, Range, Units	Description
LABEL	text	Name of your job
MASS_RED	real, positive, amu	Reduced mass of two scattering partners
RMIN, RMAX	real, positive, Bohr	Minimum and maximum values of distance between partners
B_IMPCT	real, positive, Bohr	Maximum value of collision impact parameter
JTOTL, JTOTU	integer	Lower and upper limits of total angular momentum J
DL	integer	Step size for orbital angular momentum ℓ , default is 1
NMB_ENERGS	integer	Number of collision energy values U to propagate
U_ENERGY	real, positive, cm^{-1}	List of collision energies U to propagate
UMIN, UMAX	real, positive, cm^{-1}	Minimum and maximum collision energies U
DU	real, positive, cm^{-1}	Step size for setting collision energies U
TIME_STEP	real, positive, au	Propagation time step for RK4, or maximum time step allowed for ODEINT
MIN_TMSTP	real, positive, au	Minimum time step allowed in ODEINT
SINGLE_STEP	default is "NO"	Propagates each trajectory in one step using ODEINT
TIME_LIM	real, positive, au	Time limit for propagation
EPS_ODEINT	real, positive, $<1.d0$	Relative error for step-size control in ODEINT (exp. format)
EPS_MONCAR	real, positive, %	Desirable error in Monte Carlo sampling of initial conditions
PROPAGATOR	text	RK4 is default, ODEINT is optional
NMB_LOOPS	integer, default is 1	Number of full loops (360 deg.) to propagate for orbiting trajectories
NMB_OSCIL	integer, default is 1	Number of outer turning points to propagate for oscillating trajectories
NO_RESONANCE	default is "NO"	If "YES", orbiting trajectories are removed from analysis
DIFF_CROSS	default is "NO"	If "YES", differential cross section is computed (elastic only)
ANG_RES	integer, default is 1000	Number of points for angular resolution of the differential cross section
MONTE_CARLO	default is "NO"	If "YES", initial conditions are sampled randomly
NMB_TRAJ	integer, default is 100	Number of trajectories to sample using Monte Carlo (total number of trajectories)
CHECK_POINT	integer	Wall clock time (minutes after the start) to start writing a checkpoint file
RESTART	default is "NO"	If "YES", program will start from a check point file
PRN_TRJCT	integer	Indicates the value of ℓ for which all the trajectory data will be printed
PRN_JM	two integers	Indicates the desired values of j and m for the option above
MPI_PERTRAJ	integer, default is 1	Number of CPUs (MPI tasks) to use for calculations of each trajectory

Table A3: Description of keywords for the block \$POTENTIAL

Keyword	Type, Range, Units	Description	Relevant SYS_TYPE
READ_MTRX	default is “NO”	If “YES”, read the potential coupling matrix from file	all
SAVE_MTRX	default is “NO”	If “YES”, write the potential coupling matrix to file	all
UNFORMAT	default is “YES”	Saves matrix in binary form; set “NO” to save it as text	all
PROG_RUN	default is “YES”	Propagates trajectories; set “NO” to compute matrix only	all
E_UNITS	text	Energy units of PES: “A.U.”, “CM-1”, “KCAL” or “KLVN”	all
R_UNITS	text	Distance units for supplied PES: “A.U.” or “ANGS”	all
GRD_R	integer	Number of points for R -grid	all
GRD_VIB	integer / two integers	Number of points for vibrational grid / grids	2 / 6
GRD_ANG1	integer / two integers	Number of points for α -grid / grids	1,2,3,4,5/6,7,8,9,0
GRD_ANG2	integer / two integers	Number of points for β -grid / grids	1,2,3,4,5/6,7,8,9,0
GRD_ANG3	integer / two integers	Number of points for γ -grid / grids	3,4 / 5,6,7,8,9,0
VGRID_FILE	default is “NO”	If “YES”, PES values at grid points are stored/read to the file	6,7,8,9,0
EXPANSION	default is “NO”	If “YES”, the PES is represented by expansion over basis	except 2, 6
NMB_TERMS	integer	Number of PES expansion terms	except 2, 6
TERMS	sets of integers	List of the expansion terms (labeled appropriately)	except 2, 6
TERMS_ONFLY	default is “NO”	If “YES”, computes PES expansion on-the-fly for each R	except 2, 6
TERMS_FILE	default is “NO”	If “YES”, reads externally-computed PES expansion terms	except 2, 6
CALC_EXPANSION	default is “NO”	If “YES”, the expansion coefficients are computed.	except 2, 6
IR_BGN, IR_FIN	integer	The range of R -grid points used, defaults are 1 to GRD_R	all
RGRID_EQDS	default is “YES”	If “NO”, non-equidistant R -grid is generated by the code	all
RGRID_FILE	default is “NO”	If “YES”, user-defined R -grid is read from file	all
L_MAX	integer	Maximum value of index l in the expansion of the PES	all
M_MAX	integer	Maximum value of index μ in the expansion of the PES	3,4
L1_MIN, L1_MAX	integer	Min. and max. values of index l_1 in the expansion of the PES	5,7,8,9,0
L2_MIN, L2_MAX	integer	Min. and max. values of index l_1 in the expansion of the PES	5,7,8,9,0
M1_MAX	integer	Maximum value of index μ_1 in the expansion of the PES	5,7,8
M2_MAX	integer	Maximum value of index μ_2 in the expansion of the PES	9,0
MIJ_CUTOFF	real, default is 1.d-12	Neglects matrix elements that are below this value	all
MIJ_SHIFT	default is “YES”	Shifts each matrix elements by its value at the last point of R -grid	all
PRINT_DIAGONAL	default is “NO”	If “YES”, prints diagonal elements of transition matrix	all

Appendix B. Description of user-supplied data files and the corresponding subroutines

For several special cases (discussed further below), some of the input data must be generated externally by users and supplied in separate files. The following table gives brief description of these data files (extension `*.DAT`). Examples can be found in the directory `ROUTINES` of the code. In order to create properly formatted files users can employ our subroutines supplied with the code, all located in the file `user_input.f`. They are listed in the table below. Note that these are not ready-to-use utility programs to generate these data, but merely the examples of data formats required by the MQCT code.

Table B1: Description of the input data files for MQCT calculations

Data File Name	Generating Subroutine	Brief Description
<code>USER_DEFINED_RGRID.DAT</code>	<code>DEFINE_RGRID</code>	Contains user defined (<i>e.g.</i> , non-equidistant) grid for the molecule-molecule distance R . Can be useful for deep short-range interaction potential energy wells.
<code>USER_DEFINED_BASIS.DAT</code>	<code>DEFINE_BASIS</code>	Contains channel labels (quantum numbers), energies, and wave functions of the externally computed ro-vibrational states. For <code>SYS_TYPE=2</code> and <code>6</code> the vibrational wave functions should be pre-computed on a grid. For system types <code>4</code> , <code>8</code> , <code>9</code> and <code>0</code> the rotational states (<i>e.g.</i> , of Kyrö hamiltonian) should be represented by expansion over the basis set of symmetric-top eigenstates.
<code>PES_EXPAN_TERMS.DAT</code>	<code>EXPAND_PES</code>	Contains R -dependence of the expansion coefficients for analytic representation of the PES using the basis sets of spherical harmonics. Works for all values of <code>SYS_TYPE</code> , except <code>2</code> and <code>6</code> where the vibrational motion is involved.
<code>STATES_TO_EXCLUDE.DAT</code>	<code>EXCLUDE_STATES</code>	Optional. Undesired states can be excluded by listing in this file the state numbers (as they appear in the file <code>STATES.out</code>), and setting the keyword <code>EXCLUDE_STATES=YES</code> .

Appendix C. Options for PES representation and computation of the transition matrix

Within MQCT code there are four ways of computing the potential coupling matrix. Differences are in how the PES is represented and how the data are handled. Description of the format for four options can be found in the file `user_suppl_pot.f` in the directory `PES_USER`, and the file `pes_sys_type.f` in the main code directory `MQCT_v1.01`

Option 1: keyword `EXPANSION=NO`, which is the default. In this case user should provide the potential subroutine `USER_DEFINED_PES` that generates the value of potential energy as a function of the molecule-molecule distance R and the internal coordinates (Euler angles in the body-fixed reference frame, and bond lengths). MQCT code will use this subroutine to compute elements of the state-to-state transition matrix directly, by numerical integration over the internal molecular degrees of freedom. Multi-dimensional Gauss-Legendre quadrature is employed with the number of points indicated by the corresponding keyword in the input file (`GRD_VIB`, `GRD_ANG1`, `GRD_ANG2`, and `GRD_ANG3`). Such calculations are done for every grid point of the molecule-molecule distance R and the data are stored in the memory. For calculations of the collision process, when the values of matrix elements and their derivatives (for classical equations of motion) are needed at certain values of R along trajectory, one-dimensional cubic spline of each matrix element is computed.

Option 2: keyword `EXPANSION=YES`. In this case user should provide subroutine `USER_DEFINED_TERMS` that generates coefficients of expansion of the PES over basis set of spherical harmonics (different for different system types, see Appendix D). Description of the format can be found in the file `user_suppl_pot.f`. Using these coefficients MQCT code will calculate elements of the state-to-state transition matrix analytically at every grid point of the molecule-molecule distance R , store them in the memory and finally spline (during the dynamics calculations, just like in the Option 1). This option is specifically created for users of MOLSCAT, since an identical subroutine is used there (called `VSTAR`) and can be employed here without modifications. The other benefit of this approach is that transitions forbidden by symmetry, and the corresponding states, can be excluded *a priori* to ease calculations and slightly improve accuracy. Finally, Option 2 can be used to confirm convergence of Option 1 (where forbidden transitions should show up negligible probabilities), and vice versa.

Option 3: keywords `EXPANSION=YES`, `TERMS_FILE=YES`. This case is methodologically equivalent to the Option 2 above, except that here user is required to supply the data file `PES_EXPAN_TERMS.DAT` that contains the externally-computed expansion coefficients at every grid point of the molecule-molecule distance R (instead of the subroutine to compute them). This is convenient when these data are already available, say from literature. Format of the data file is described in the subroutine `EXPAND_PES`, see Appendix B. MQCT code will read these data as input, calculate elements of the state-to-state transition matrix analytically, store

them in the memory and then spline for trajectory calculations as needed, like in the Options 1 and 2. Of course, the user-supplied subroutine suitable for Option 2 can also be used to pre-compute the expansion coefficients and create the data file for Option 3. Or, user can pre-run the MQCT code with the optional keyword `CALC_EXPANSION=YES` to generate the file `PES_EXPAN_TERMS.DAT`. In this case projection integrals are computed using multi-dimensional Gauss-Legendre quadrature with the number of points indicated by `GRD_VIB`, `GRD_ANG1`, `GRD_ANG2`, and `GRD_ANG3`.

Option 4: keywords `EXPANSION=YES`, `TERMS_ONFLY=YES`. In this case no grid over the molecule-molecule distance R is employed, no data are stored in the memory, and no splining is involved. The user should supply the subroutine `USER_DEFINED_COEFFS` that will generate *both* the potential expansion coefficients and their R -derivatives at any value of R . The required format is described in the file `user_suppl_pot.f` in the directory `PES_USER`. Using this subroutine MQCT code will compute analytically the matrix elements and their R -derivatives on-the-fly, as requested by propagator along the trajectory. This approach is only practical if generation of the expansion coefficients and their derivatives is computationally inexpensive, for example, if their R -dependencies are described by simple analytic model (e.g., for simple molecular system). Other options, within the `USER_DEFINED_COEFFS` subroutine, would be to spline the expansion coefficients, or to re-compute the expansion at every point by projection, but those, again, would be practical for the simplest systems with smallest basis sets. Moreover, in the current version of the code Option 4 is implemented only for calculations with one processor per trajectory (*i.e.*, without the second level of parallelization). It can be used for debugging, for machines with small number of processors and small memory, and for model systems.

Appendix D. Expansion of the PES over the basis set of analytic functions

In the current version of MQCT we use PES expansions identical to those implemented in the MOLSCAT package for these system types:

`SYS_TYPE=1`: for diatomic + atom the PES is expanded over the basis set of Legendre polynomials (Arthurs, Dalgarno, 1960). To define the expansion terms by the keyword `TERMS`, user will specify only one integer number for each term, the value of l , which is a rank of Legendre polynomial. The values for different terms are separated by coma. The order of indicated terms defines the order in which they will be handled (read from `PES_EXPAN_TERMS.DAT`, computed by the subroutine `USER_DEFINED_TERMS`, summed into the matrix element, *etc.*). Note, that the total number of terms employed must be specified by the keyword `NUMB_TERMS`, before they are listed one by one. If user does not wish to list all terms, the code can automatically assign them based on the keyword `L_MAX`.

SYS_TYPE=3 and 4: for any top + atom the PES is expanded over the basis set of spherical harmonics (Green, 1976). User should specify two integer numbers for each term, l and its projection m (in this order, separated by coma), that define spherical harmonic functions. If user does not wish to list all the terms, the code can automatically assign them using the optional keywords `L_MAX` and `M_MAX`. Note that in the literature the symbols λ and μ are often used, instead of l and m .

SYS_TYPE=5: for diatom + diatom the PES is expanded over the basis set of generalized spherical harmonics (Green, 1974). For each term listed the user should specify three numbers: l_1 , l_2 , and l (in this order, separated by coma). Alternatively, the optional keywords can be employed to generate the terms automatically up to `L1_MAX`, `L2_MAX` and `L_MAX`.

However, for the following system types the PES expansions of MQCT are different from those used in MOLSCAT, since MOLSCAT uses the molecule-fixed reference frame (Phillips, Green, 1994), while MQCT exploits the body-fixed reference frame (Semenov, Babikov, 2016). Namely:

SYS_TYPE=7 and 8: for any top + diatom MQCT code uses expansion over normalized functions which are products of Wigner D -functions and spherical harmonics. Users should indicate four integers for each expansion term: l_1 , m_1 , l_2 and l (in this order, separated by coma). The option of generating these terms automatically is also available, through `L1_MAX`, `M1_MAX`, `L2_MAX` and `L_MAX`. The same expansion was used in the HYBRIDON package (Alexander, 2015) and also in some earlier calculations but with different symbols: p_1 , q_1 , p_2 and p (Green, 1994). Subroutine `MFTOBF_CONV` (in the file `user_suppl_pot.f`) is available for conversion of the PES from the more standard molecule-fixed reference frame to the body-fixed reference used by MQCT.

SYS_TYPE=9 and 0: top + top collisions have never been studied before, although similar expansion was proposed in the past (Szalewicz, 1996). In the MQCT code the PES is expanded over the basis set of normalized functions which are products of Wigner D -functions for each molecule. The expansion terms are labeled by l_1 , m_1 , l_2 , m_2 and l . Note that value of m_2 can be negative integer and the code will read it without an error message. As in all previous cases, automatic generation of the terms is enabled by `L1_MAX`, `M1_MAX`, `L2_MAX`, `M2_MAX` and `L_MAX`. Note that in the literature symbol η was used, instead of m .

Also note that, if desired, the keywords `L1_MIN` and `L2_MIN` can be used for several system types, as indicated in the Table A3, to define the minimum values of labels for the expansion basis in the case of automatic generation of terms.

Appendix E. Computation of coupling matrix and/or potential expansion over the basis set

Both the direct calculation of transition matrix (default) and the expansion of PES over basis set of analytic functions (optional keyword `CALC_EXPANSION=YES`) deal with numerical integration over the internal degrees of freedom using multi-dimensional numerical quadrature. For both cases the number of quadrature points should be indicated by user in the input file using keywords `GRD_VIB`, `GRD_ANG1`, `GRD_ANG2`, and `GRD_ANG3`. Integration over vibrational coordinate uses weights indicated for each point in the file `USER_DEFINED_BASIS.DAT`, or uses Gauss-Legendre quadrature for automatically-generated Morse oscillator states. Integration over Euler angles always uses Gauss-Legendre quadrature. Some of these are essential internal degrees of freedom, others are dummy variables, depending on the system, as described below:

`SYS_TYPE=1`: for diatomic + atom integration is carried out along β -angle only, with the number of points `GRD_ANG2`. The values of `GRD_ANG1` and `GRD_ANG3` are dummy. Vibrational degree of freedom is dummy for all `SYS_TYPE` but 2 and 6.

`SYS_TYPE=2`: for vibrating diatomic + atom, besides β -angle described above, the integral includes `GRD_VIB` points for vibration. The values of `GRD_ANG1` and `GRD_ANG3` are dummy, just as in `SYS_TYPE=1`.

`SYS_TYPE=3 and 4`: for any top + atom the number of points for angles β and γ is indicated by `GRD_ANG2` and `GRD_ANG3`, respectively. The value of `GRD_ANG1` is dummy

`SYS_TYPE=5`: for diatom + diatom the number of points along two β -angles is given by two entries of the keyword `GRD_ANG2`, while the number of points for γ -angle is indicated by the *first* entry of the keyword `GRD_ANG3`. The second entry of the keyword `GRD_ANG3`, and both entries of the keyword `GRD_ANG1` are dummy.

`SYS_TYPE=6`: for vibrating diatom + diatom the number of points along each bond length is indicated by two entries of the keyword `GRD_VIB`, in addition to the angular coordinates of `SYS_TYPE=5`.

`SYS_TYPE=7 and 8`: for any top + diatom the number of points along two β -angles is given by two entries of the keyword `GRD_ANG2`, the number of points along α is given by the second entry of `GRD_ANG1` (the first entry is dummy), while the number of points along γ is given by the first entry of the keyword `GRD_ANG3` (the second entry is dummy).

`SYS_TYPE=9 and 0`: for top + top systems the number of points along α is given by the second entry of `GRD_ANG1` (only the first entry is dummy), the number of points along two β -angles is given by two entries of the keyword `GRD_ANG2`, the number of points along two γ -angles is given by two entries of the keyword `GRD_ANG3`.

Several practical aspects of multi-dimensional integration for computing transition matrix directly, and/or for expanding the PES over the basis set, should be discussed. First of all, as mentioned in the main text of the paper, it is recommended to compute and store the transition matrix (into the file `MTRX.DAT`, or `MTRX_UF.DAT`) in a separate program run, independently from the main run of collision dynamics. This is optional for the case when the matrix is calculated by direct integration (Option 1 of Appendix C), but is mandatory in the case when the PES expansion is computed (e.g., prior to using Option 3 of Appendix C). Moreover, in the first run, initiated by the keyword `CALC_EXPANSION` to generate the PES expansion coefficients, the number of processors must be equal to the number of requested expansion terms, since each processor will be responsible for computing one expansion coefficient. The code will compute the expansion coefficients and will stop, without proceeding to calculations of matrix elements or collision dynamics. Then, user should replace `CALC_EXPANSION` in the input file by `EXPANSION=YES` and run the code again with the number of processors appropriate for calculations of matrix elements (keywords `SAVE_MTRX=YES`, `PROG_RUN=NO`). Finally, the code should be run third time for the actual trajectory calculations (keywords `READ_MTRX=YES`, `PROG_RUN=YES`), with an appropriate value of `MPI_PERTRAJ` set up. In this procedure, one should be careful about the units of distance and energy. The file of expansion terms, printed by the code, `PES_EXPAN_TERMS.DAT`, will always contain the distance in Bohr and energy in wavenumber, irrespectively of the units of the potential energy surface routine used for the calculations of the expansion. If the expansion is used further (to compute the state-to-state transition matrix and/or the collision dynamics), the units should be set as Bohr and wavenumber in the `$POTENTIAL` block of the input file, to comply with the expansion file, rather than with the original PES routine that is not anymore used.

If the angular grid (for Option 1) is very large, or the number of expansion terms (for Option 3) is large, and in particular when the number of *R*-grid points is large, it may be convenient to split calculations of the matrix or the PES expansion into several runs. To do that, user can specify the range of *R*-grid to cover in one run, using keywords `IR_BGN` and `IR_FIN`. Results of successive runs are combined automatically into a single file for the matrix, or for the expansion terms and coefficients.

When computing the state-to-state transition matrix by direct integration (Option 1), or computing the PES expansion by projection (e.g., Option 3), it may be advantageous, in terms of CPU time, to keep in the memory the values of potential at the grid points, rather than calling the PES subroutine each time when the value for new point is needed. However, a very large grid (for larger molecules and complicated PES) may not fit as a whole into the memory of one CPU. For this case the option `VGRID_FILE=YES` is recommended. The code first generates the PES at the points of the grid and saves these data to the unformatted file `VGRID_UF.DAT`. The number of processors should be at least equal to the number of points of *R*-grid, or larger. Then the code loads this information into the memory of processors to compute matrix elements (or the expansion coefficients) but does it by slices, sequentially for each value of *R* on the grid, since calculation at each value of *R* is independent. Different processors will be responsible for computing different elements of the matrix, or different expansion terms. Note that if the expansion is being computed the code will save the data file `VGRID_UF.DAT` and will normally stop (except a rare special case when the number of *R*-grid points is equal to the number of the PES expansion terms). It should be run again with the number of CPUs equal to the number of the expansion terms. The code will read the data file `VGRID_UF.DAT` and proceed to calculations of the expansion coefficients, one term per processor. Calculations of matrix elements and the collision dynamics should be done in the following independent runs, as explained above.

In order to take the full advantage of symmetry (when the symmetry is not obvious) the code automatically neglects transitions described by matrix elements with absolute values smaller than `MIJ_CUTOFF=1.d-12`. If needed, an alternative value of the cut-off criterion can be

specified. This keyword can also be used to make calculations more efficient by disregarding transitions between the weakly coupled states. For this, a suitable value of `MIJ_CUTOFF` should be determined by the convergence studies. Another relevant keyword is `MIJ_SHIFT`. By default, each matrix element is automatically shifted by its value at the last point of R -grid, to ensure that no transitions occur in the asymptotic region. This shift can be disabled by indicating `MIJ_SHIFT=NO`.

Appendix F. User-supplied PES subroutine

Formally, the potential energy surface subroutine `USER_DEFINED_PES` operates with the same coordinates for all system types, but, as it follows from Appendix E (above), for certain system types, some of these coordinates are dummy variables. Namely, the input for the PES subroutine requires, besides the molecule-molecule distance R , one vibrational coordinate and three Euler angles for each collision partner (see the file `user_suppl_pot.f` in the directory `PES_USER`, or the file `pes_sys_type.f` in the code directory `MQCT_v1.01`). However, the vibrational coordinate is a dummy variable for all values of `SYS_TYPE`, except 2 and 6 where the vibrational motion of the diatomic is explicitly described. In the future, our plan is to add one vibrational degree of freedom (such as bending motion in triatomic molecules) for other system types, but this is not yet implemented in the present release of the code. Concerning the angular coordinates, some of them are dummy as described in Appendix E. For completeness, we summarized these properties in the Table F1 below, where dash denotes a dummy variable. We want to emphasize one more time that our reference frame, called the body-fixed reference frame, tied to the molecule-molecule vector, is different from coordinates used in some other codes (such as `MOLSCAT`) where the reference frame is tied to one of the molecules (see Appendix E for details).

Table F1: Degrees of freedom in the user-supplied PES subroutine

<code>SYS_TYPE</code>	R	r_1	r_2	α_1	β_1	γ_1	α_2	β_2	γ_2
1	distance	--	--	--	polar	--	--	--	--
2	distance	vibration	--	--	polar	--	--	--	--
3	distance	--	--	--	polar	azimuthal	--	--	--
4	distance	--	--	--	polar	azimuthal	--	--	--
5	distance	--	--	--	polar	azimuthal	--	polar	--
6	distance	vibration	vibration	--	polar	azimuthal	--	polar	--
7	distance	--	--	--	Euler	Euler	azimuthal	polar	--
8	distance	--	--	--	Euler	Euler	azimuthal	polar	--
9	distance	--	--	--	Euler	Euler	Euler	Euler	Euler
0	distance	--	--	--	Euler	Euler	Euler	Euler	Euler