Manual

MULTILEVELRATE - version 9.4

Jingzhi Pu, José C. Corchado, Benjamin J. Lynch, Patton L. Fast, Benjamin A. Ellingson, and Donald G. Truhlar

Department of Chemistry and Supercomputer Institute, University of Minnesota, Minneapolis, MN 55455-0431

Program Version: 9.4/P9.4.1-M4.2

Program Version Date: July 25, 2006

Manual Version Date: July 21 2021

Abstract

MULTILEVELRATE is a set of FORTRAN subroutines and Unix scripts for interfacing the POLYRATE and MULTILEVEL computer programs. The interfaced programs can be used either to carry out direct dynamics calculations of gas-phase chemical reaction rates of polyatomic species (and also atoms and diatoms as special cases) using the electronic structure methods available in MULTILEVEL to calculate the potential energy surface and POLYRATE for the dynamics, or (ii) to carry out normal mode analysis using the multilevel methods of MULTILEVEL. The interface is based on the POLYRATE hooks protocol. The dynamical methods used are variational or conventional transition state theory and multidimensional semiclassical approximations for tunneling and nonclassical reflection. Rate constants may be calculated by any of the methods available in the POLYRATE-version 9.3 program for canonical or microcanonical ensembles or for specific vibrational states of selected modes with the translational, rotational, and other vibrational modes treated thermally. Bimolecular and unimolecular reactions are included. Both single-level and dual-level dynamics calculations may be carried out. In single-level mode, optimized geometries, potential energies, gradients, and Hessians can be calculated by any of the integrated molecular orbital (IMO) methods and linear combination (LC) methods provided by the MULTILEVEL package. In dual-level mode, the lower-level data is calculated by MULTILEVEL, and the higher-level data is read in from an external file.

Description

MULTILEVELRATE-version 9.4 provides an interface between two other programs:

POLYRATE-version 9.4.1

José C. Corchado, Yao-Yuan Chuang, Patton L. Fast, Wei-Ping Hu, Yi-Ping Liu, Gillian C. Lynch, Kiet A. Nguyen, Charles F. Jackels, Antonio Fernandez Ramos, Benjamin A. Ellingson, Benjamin J. Lynch, Vasilios S. Melissas, Jordi Villà, Ivan Rossi, Elena L. Coitino, Jingzhi Pu, Titus V. Albu
Department of Chemistry and Supercomputer Institute
University of Minnesota, Minneapolis, Minnesota

Rozeanne Steckler
San Diego Supercomputer Center, La Jolla, California

Bruce C. Garrett
Environmental Molecular Sciences Laboratory
Pacific Northwest Laboratory, Richland, Washington

Alan D. Isaacson
Department of Chemistry, Miami University, Oxford, Ohio

and Donald G. Truhlar
Department of Chemistry and Supercomputer Institute
University of Minnesota, Minneapolis, Minnesota

MULTILEVEL-version 4.2

Jocelyn M. Rodgers, Benjamin J. Lynch, Patton L. Fast,
Yan Zhao, Jingzhi Pu, Yao-Yuan Chuang, Benjamin A. Ellingson
and Donald G. Truhlar
Department of Chemistry and Supercomputer Institute
University of Minnesota, Minneapolis, Minnesota

User agreement

MULTILEVELRATE - version 9.4 is licensed under the Apache License, Version 2.0.
The manual of *MULTILEVELRATE* - version 9.4 is licensed under CC-BY-4.0.

Publications of results obtained with the *MULTILEVELRATE* - version 9.4 software should cite the program and/or the article describing the program.

No guarantee is made that this software is bug-free or suitable for specific applications, and no liability is accepted for any limitations in the mathematical methods and algorithms used within. No consulting or maintenance services are guaranteed or implied.

The use of the *MULTILEVELRATE* - version 9.4 implies acceptance of the terms of the licenses.

Table of Contents

Chapter One

1

1. Introduction

MULTILEVELRATE is a program for the analysis of reactants, products, and transition states of chemical reactions and for direct dynamics calculation of variational transition state theory (VTST) rate constants and multi-dimensional semiclassical tunneling probabilities using the dual-level or multi-level electronic structure methods in MULTILEVEL package to represent the potential energy of interaction and its first and second derivatives whenever they are needed. This version 9.4/P9.4.1-M4.2 interfaces POLYRATE-version 9.4.1, which is a program for dynamical rate calculations, with the electronic structure program MULTILEVEL-version 4.2.

MULTILEVEL is a program for calculating optimized geometries, single point energies , single point gradients, and/or single-point Hessians using dual-level and multi-level methods, in which GAUSSIAN03 (or 98 or 94) is used as an external program for the components calculation.

MULTILEVELRATE requires that the user has both POLYRATE-version 9.4.1 and MULTILEVEL-version 4.2 on the same computer working properly before attempting to run MULTILEVELRATE.

No part of POLYRATE-version 9.4.1 or MULTILEVEL-version 4.2 is contained in MULTILEVELRATE or needs to be modified for running this program.

This MULTILEVELRATE manual covers only material that is not covered in the POLYRATE and MULTILEVEL manuals. The user of MULTILEVELRATE should read those manuals first.

The interface between POLYRATE and MULTILEVEL is made by a series of FORTRAN subroutines included in the files `hooks_mr.f` and `hooks_mdep.f`, which are specialized versions of the routines in the `hooks.f` file in POLYRATE-version 9.3 and by a C shell script `shuttle_mr` for running MULTILEVEL. This C shell script `shuttle_mr` takes the name of a MULTILEVEL input file and a MULTILEVEL output file as its first and second argument respectively.

After call to MULTILEVEL, the MULTILEVELRATE program generates formatted summary file that contains necessary information, such as geometries, energies, gradients, and Hessians. This formatted summary file will be read by MULTILEVELRATE, which passes all the requested information to the dynamics subroutines of POLYRATE.

1.1. Reference

Publications based on MULTILEVELRATE should give the following references:

> J. Pu, J. C. Corchado, B. J. Lynch, P. L. Fast, B. A. Ellingson, and D. G. Truhlar, MULTILEVELRATE-version 9.4, University of Minnesota, Minneapolis, 2004 based on (i) J. C. Corchado, Y.-Y. Chuang, P. L. Fast, W.-P. Hu, Y.-P. Liu, G. C. Lynch, K. A. Nguyen, C. F. Jackels, A. Fernandez Ramos, B. A. Ellingson, B. J. Lynch, V. S. Melissas, J. Villà, I. Rossi, E. L. Coitiño, J. Pu, T. V. Albu, R. Steckler, B. C. Garrett, A. D. Isaacson, and D. G. Truhlar, POLYRATE- version 9.4.1, University of Minnesota, Minneapolis, 2006 and (ii) J. M. Rodgers, B. J. Lynch, P. L. Fast, Y. Zhao, J. Pu, Y. -Y. Chuang, B. A. Ellingson, and D. G. Truhlar, MULTILEVEL-version 4.2, University of Minnesota, Minneapolis, 2006.

1.2. Versions

The version number of MULTILEVELRATE has three parts, e.g., MULTILEVELRATE-version *x*/P*y*-M*z*, where *y* denotes the version of POLYRATE, *z* denotes the version of MULTILEVEL, and *x* denotes the version of MULTILEVELRATE. If *y* or *z* changes, or if anything except the manual changes in the interface package (which is

MULTILEVELRATE proper), then *x* will always changes. Thus *x* is a unique identifier, and one can say version *x* instead of version *x*/P*y*-M*z* except when one wants to emphasize the versions of the underlying codes.

Chapter Two

2

2. Installation

2.1. Obtaining the codes

The user needs to obtain three items of code:

POLYRATE:

The POLYRATE code is available from the University of Minnesota. For further information see:

http://comp.chem.umn.edu/polyrate

MULTILEVEL:

The MULTILEVEL code is also available from the University of Minnesota. For further information see:

http://comp.chem.umn.edu/multilevel

MULTILEVELRATE:

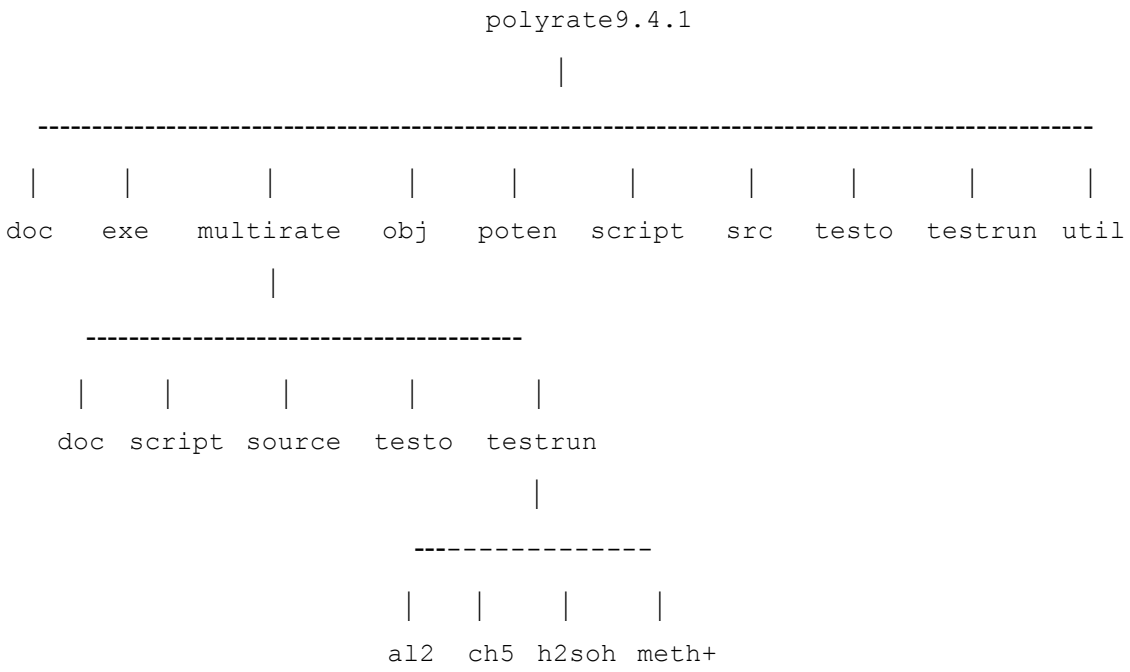
The MULTILEVELRATE code is also available from the University of Minnesota. For further information see:

http://comp.chem.umn.edu/multilevelrate

The prospective user of MULTILEVELRATE must obtain and install POLYRATE and MULTILEVEL before proceeding with MULTILEVELRATE.

2.2. Installation

Before installing MULTILEVELRATE, the user should first successfully install POLYRATE and MULTILEVEL. MULTILEVELRATE is distributed in tar format. The tar file needs to be untarred in the PO LYRATE-version 9.4.1 directory. The command is `tar -xvf multilevelrate9.4.tar`. This command will create a directory called `multirate`. The C shell `shuttle_mr` and the directories `script`, `source`, `doc`, `testo`, and `testrun` are located in this MULTILEVELRATE parent directory. If instructions are followed correctly, the directory structure should be the following:



The files will be distributed among the directories as follow:

`multirate`: `configure`, and five subdirectories

`doc`: empty

`script`: `mrcompile`, `shuttle_mr`

`source`: `hooks_mr.f`, `hooks_mr.wks`, `head_mr.f`, `multirate.inc`

`testo`: `al2sac.fu6`, `al2mcg3.fu6`, `al2-mcg3.fu6`, `al2+mcg3.fu6`, `ch5hf.fu6`, `ch5sac.fu6`, `ch5mcomp2_gp.fu6`, `ch5mcomp2_srp.fu6`, `h2soh_sac.fu6`, `meth+mqcisd.fu6`

`testrun`: `al2/`, `ch5/`, `h2soh/`, `meth+/`,

2.3. Compiling

The configure script will make use of Makefile in the polyrate9.4.1/src directory to compile MULTILEVELRATE. The configure script will create a section in that Makefile and then call the script mrcompile, which will compile the source code. The multirate.exe executable will be placed in the polyrate9.4.1/exe directory.

Chapter Three

3

3. Subroutines and C Shell Scripts

Below is a brief description of MULTILEVELRATE subprograms and scripts utilized in MULTILEVELRATE. The final name on the first line is the name of the file in which the routine is found. Detailed caller and callee information can be found as comments in the source code.

EHOOK Subroutine `hooks_mr.f`

Calculates the energy using MULTILEVEL for a given set of coordinates.

ENATIN Subroutine `hooks_mdep.f`

Calculates a MULTILEVEL energy for atomic systems.

ESCENER Subroutine `hooks_mdep.f`

Carries out a MULTILEVEL single-point energy calculation.

ESCFIRST Subroutine `hooks_mdep.f`

Carries out a MULTILEVEL energy and gradient calculation.

ESCSEC Subroutine `hooks_mdep.f`

Carries out a MULTILEVEL Hessian calculation.

GEOIN Subroutine `hooks_mdep.f`

Carries out a MULTILEVEL geometry optimization.

GHOOK Subroutine `hooks_mr.f`

Calculates the energy and its first derivatives for a given geometry.

HEADR Subroutine `head_mr.f`

Writes the program header in the file `fu6`.

HHOOK Subroutine `hooks_mr.f`

Calculates the Hessian matrix for a given geometry using either MULTILEVEL or one of the methods in POLYRATE.

MINPENE Subroutine `hooks_mr.f`

Writes a standard MULTILEVEL input file for single-point energy calculation.

MINPFRS Subroutine `hooks_mr.f`

Writes a standard MULTILEVEL input file for a first derivative calculation.

MINPGEO Subroutine `hooks_mr.f`

Writes the new geometry to the MULTILEVEL input file.

MINPSEC Subroutine `hooks_mr.f`

Writes a standard MULTILEVEL input file used to perform a second derivative calculation.

MOPTOFF Subroutine `hooks_mr.f`

Removes MULTILEVEL optimization options to generate a MULTILEVEL input file for a single-point calculation.

MOUTSEC Subroutine `hooks_mr.f`

Extracts electronic structure information from the MULTILEVEL format summary file `ml.sum`.

MREST Subroutine `hooks_mr.f`

Reads and writes the information in the restart files for a MULTILEVELRATE restart calculation.

MTASK Subroutine `hooks_mr.f`

Rewrites charge, multiplicity, number of atoms, and calculation task options of a MULTILEVEL input file.

OHOOK Subroutine `hooks_mr.f`

Optionally accepts an initial guess to the Cartesian coordinates of the full system and carries out a geometry optimization using either MULTILEVEL or a method in POLYRATE.

PREP Subroutine `hooks_mr.f`

Prepares the calculation and, depending on the selected options, carries out certain geometry optimizations and Hessian calculations.

PREPJ Subroutine `hooks_mr.f`

Initializes variables dependent on each stationary point.

RCHMUL Subroutine `hooks_mr.f`

Reads the charge and multiplicity from the MULTILEVEL formatted summary file `ml.sum`.

RCODMR Subroutine `hooks_mr.f`

Reads MULTILEVELRATE general option in file `fu70`.

REDINF Subroutine `hooks_mr.f`

Reads the information about reactants, products and saddle point calculated in

PREP subroutine.

RENERMR Subroutine hooks_mr.f

Reads the input options in file fu70 for MULTILEVEL energy calculation.

RFIRSTMR Subroutine hooks_mr.f

Reads the input option for MULTILEVEL gradient calculation.

RGENMR Subroutine hooks_mr.f

Reads the general option in file fu70.

RINPMR Subroutine hooks_mr.f

Reads MULTILEVELRATE options from file fu70.

RLINMR Subroutine hooks_mr.f

Reads a line and parses out the comments and extra spaces.

RSECMR Subroutine hooks_mr.f

Reads the input option for a MULTILEVEL Hessian calculation.

RSTATMR Subroutine hooks_mr.f

Reads input options for stationary points.

SETUP Subroutine hooks_mr.f

Initializes some variables and calls the subroutines for reading in input file fu70.

SET_VERSION Perl script set_version

Interactively sets the POLYRATE and MULTILEVEL versions.

SHUTTLE_MR C Shell Script shuttle.mr

Runs a MULTILEVEL calculation and creates the summary file.

STORE Subroutine `hooks_mr.f`

Stores information that may be useful for future calculations to avoid duplicated MULTILEVEL calls.

SECIN Subroutine `hooks_mr.f`

Carries out the MULTILEVEL Hessian calculation for stationary points.

Chapter Four

4

4. File usage

File	Usage
esp.fu70	General option for MULTILEVELRATE
esp.fu71	MULTILEVEL input file for geometry optimization, energy, and gradient calculation for reactant 1.
esp.fu72	MULTILEVEL input file for geometry optimization, energy, and gradient calculation for reactant 2.
esp.fu73	MULTILEVEL input file for geometry optimization, energy, and gradient calculation for product 1.
esp.fu74	MULTILEVEL input file for geometry optimization, energy, and gradient calculation for product 2.
esp.fu75	MULTILEVEL input file for geometry optimization, energy, and gradient calculation for saddle point.
esp.fu77	MULTILEVEL input file for geometry optimization, energy, and gradient calculation for the well on the product side.
esp.fu78	MULTILEVEL input file for geometry optimization, energy, and gradient calculation for the well on the reactant side.
esp.fu81	MULTILEVEL temporary input file, generated by MULTILEVELRATE.
esp.fu82	MULTILEVEL temporary output file, generated by MULTILEVELRATE.
esp.fu83	Restart file to be read during a restart calculation.
esp.fu84	Restart file to be written during a restart calculation.

`esp.fu86`	A temporary MULTILEVEL input file template for all the generalized transition state points along the reaction path.
`ml.sum`	A formatted MULTILEVEL output summary generated by MULTILEVEL. It uses the `fu85` unit.
`.dat(fu5)`	Input file for POLYRATE option
`.fu6(fu6)`	MULTILEVELRATE output file

If the execution script .mr (see Chapter 7) is used for running MULTILEVELRATE, files generated by MULTILEVEL including MULTILEVEL output files, GAUSSIAN input/output files, GAUSSIAN checkpoint files, and non-standard basis sets files, will be all transferred to a new directory named `MRTEMPFILES` in the working directory at the end of each calculation. One can check for those MULTILEVEL files if a task crashes.

Chapter Five

5

5. Input Description

MULTILEVELRATE is written in a way that puts no limitation on MULTILEVEL and POLYRATE options. In other words, all valid options for MULTILEVEL and POLYRATE are usable in the MULTILEVELRATE input file. The MULTILEVELRATE program simply acts as an interface without interfering with the dynamics and electronic structure parts.

The following files are MULTILEVELRATE input files:

	Fortran Unit	Extension	Description
			MULTILEVELRATE
1)	`fu70`	`.70`	general options

Comment:

The input option format follows the POLYRATE input convention. The section containing general options is "*MRGENERAL". Currently, only one keyword called "MRRESTART" is available in this section. The switch "MRRESTART" is used to specify that a restart calculation if restart information exists or to write a restart file if a restart file does not exist. The remaining input sections are regarding to stationary points: *MRREACT1, *MRREACT2, *MRPROD1, *MRPROD2, *MRSTART, *MRWELLR, and *MRWELLP, corresponding to reactants, products, the saddle point, the reactant well, and the product well. One can specify the charge of each species under these sections by variable keywords "CHARGE *n*", where *n* is the

charge of the species (see the revision history of MULTILEVELRATE-v9.3 for more detailed information).

2) `fu71-fu78` `.71-.78` MULTILEVEL input

Comment:

The usage of these files has been discussed in Chapter 4. They are MULTILEVEL input files for stationary points, the saddle point, and the reactant or the product well. Please check the MULTILEVEL manual for details of options, lists, and switches keywords.

The order of the atoms should be consistent with the order used in the POLYRATE input file `.dat(fu5)` specified. "GRADIENT" switch is supposed to be turned on for all the stationary points except for an atomic species in those MULTILEVEL input files. The switch keyword "PRSUM" **MUST** be turned on to specify a summary file named "ml.sum" is printed by MULTILEVEL. For atomic species, the "MULTIOPT" section should not be present due to the meaninglessness of geometry optimization on an atom.

3) `fu5` `.dat` POLYRATE input

Comment:

The format and options follow the POLYRATE input file `fu5`. Check the POLYRATE manual for further details.

Chapter Six

6

6. Testing the code

MULTILEVELRATE-version 9.4 was tested on the following computers and operating systems:

Platform	Operating System	Compilers
IBM Regatta, Power4 procs	AIX 5.2	XL Fortran version 8.1.1
		g77 from gcc version 3.4.1
IBM Netfinity, Dual Pentium III procs	Redhat Enterprise Linux 3	Portland group fortran compiler version 5.2
		g77 from gcc version 3.3.3
SGI Altix, Itanium 2 procs	SuSE Linux	Intel ifort version 9.0
		g77 from gcc version 3.2.3

Chapter Seven

7

7. Test run

To run test runs, well commented scripts ending with `.mr` are provided in this package. For example, type `ch5hf.mr` to run the first CH_5 test run. The output file inherits the root name of the test run, for example, in this case the output file is `ch5hf.fu6`.

7.1. CH_5 system test run

Reaction: $CH_3 + H_2 \rightarrow CH_4 + H$

Test run `ch5hf`

Dynamics:	TST and CVT reaction rates, with ZCT and SCT tunneling.
Electronic Structure:	HF/STO-3G

Test run `ch5mcomp2_gp`

GP (general parameters)

Dynamics:	TST and CVT reaction rates, with ZCT and SCT tunneling. ESD/RODS algorithm is used to follow the reaction path.
Multilevel Method:	MCCM-CO-MP2/MG3;6-31G(d), using the version v2m parameters.

Test run `ch5mcomp2_srp`

SRP (specific reaction parameters)

Dynamics:	TST and CVT reaction rates, with ZCT and SCT tunneling. ESD/RODS algorithm is used to follow the reaction path.
Multilevel Method:	MCCM-CO-MP2/MG3;6-31G(d) using the reaction specific parameters.

Test run `ch5sac`

Dynamics:	TST and CVT reaction rates, with ZCT and SCT tunneling. ESD/RODS algorithm is used to follow the reaction path.
Multilevel Method:	SAC with the version v2m parameters

7.2. H_2SOH system test run

Reaction: $H_2S + OH \rightarrow SH + H_2O$

Test run `h2soh_sac`

Dynamics:	TST reaction rates
Multilevel Method:	SAC with v3s parameters

Test runs 7.3 and 7.4 show how to use MULTILEVELRATE to carry out multilevel frequency calculations for stationary points, in which the POLYRATE normal mode analyzer takes Hessians computed by MULTILEVEL as input.

7.3. Al_2 frequencies

Test run `al2sac`	Al_2 frequency, SAC with v3s parameters
Test run `al2mcg3`	Al_2 frequency, MCG3 with v3s parameters
Test run `al2-mcg3`	Al_2^- frequency, MCG3 with v3s parameters
Test run `al2+mcg3`	Al_2^+ frequency, MCG3 with v3s parameters

7.4. CH_3^+ frequencies

Test run `meth+mcqcisd`	CH_3^+ frequencies, MC-QCISD with v3s parameters

Chapter Eight

8

8. Bibliography

1. "Transition State Structure, Barrier Height, and Vibrational Frequencies for the Reaction Cl + $CH_4 \rightarrow$ CH_3 + HCl," T. N. Truong, D. G. Truhlar, K. K. Baldridge, M. S. Gordon, and R. Steckler, Journal of Chemical Physics **90**, 7137-7142 (1989).

2. "*Ab Initio* Transition State Theory Calculations of the Reaction Rate for OH + CH_4 $\rightarrow$ H_2O + CH_3," T. N. Truong and D. G. Truhlar, Journal of Chemical Physics **93**, 1761-1769 (1990). Erratum: **97**, 8820 (1992).

3. "Direct Dynamics for Free Radical Kinetics in Solution: Solvent Effect on the Rate Constant for the Reaction of Methanol with Atomic Hydrogen," Y.-Y. Chuang, M. L. Radhakrishnan, P. L. Fast, C. J. Cramer, and D. G. Truhlar, Journal of Physical Chemistry A **103**, 4893-4909 (1999).

4. "Parameterized Direct Dynamics Study of Rate Constants of H with CH_4 from 250 to 2400 K," J. Pu and D. G. Truhlar, Journal of Chemical Physics **116**, 1468-1478 (2002).

5. "Tests of Potential Energy Surfaces for H + CH_4 $\leftrightarrow$ CH_3 + H_2: Deuterium and Muonium Kinetic Isotope effects for the Forward and Reverse Reaction," J. Pu and D. G. Truhlar, Journal of Chemical Physics **117**, 10675-10687 (2002).

Chapter Nine

9

9. Revision History

MULTILEVELRATE - version 8.7/P8.6-M2.3 (May 2001)

- The first version

MULTILEVELRATE - version 9.3/P9.3-M3.1 (January 2004)

- The scripts (*.mr) for running MULTILEVELRATE jobs are modified. The script automatically make a scratch directory so that all intermediate files created by MULTILEVELRATE will be put into this scratch directory which is separate from the input files. At the end of the calculations, the MULTILEVELRATE output files are copied from the scratch directory to the directory containing the input files. The MULTILEVEL input and output files are moved to a directory called MRTEMPFILES for any possible debug. Then the temporary scratch directory containing unnecessary intermediate files is deleted. It makes the end user much easier to manage the major input and output files by running MULTILEVELRATE jobs through this modified script.

- Input sections *MRREACT1, *MRREACT2, *MRPROD1, *MRPROD2, *MRSTART, *MRWELLR, and *MRWELLP are added to the fu70 file in order to specify charges for each corresponding species by the keyword CHARGE. For a MULTILEVELRATE calculation in which the stationary points are optimized through hooks, above information are ignored, regardless whether the species is neutral or not. When one defines the initial geometry as an optimized structure in POLYRATE (STATUS=2) for a charged species, this additional input of charges in fu70 is required. This corrects a bug for charged species in the previous version.

- A bug has been fixed in subroutine RSTATMR (hooks_mr.f), where a comma was missed in a print format statement. For some compilers, this bug could generate an error.

- The input keywords "SGLV" and "MTLV" (in section *MRGENERAL of fu70) are removed. By utilizing the summary file (ml.sum) printed out by MULTILEVEL, the special treatment required for a single level calculation does not exist any longer.

- Test runs for H_2SOH, Al_2, and CH_3^+ are added. The latter two cases are to show how to compute multilevel frequencies by using MULTILEVELRATE. The original CH_5 test runs have been reorganized.

MULTILEVELRATE - version 9.4/P9.4-M4.2 (July 2006)

- Installation is now accomplished by using the Makefile in the polyrate/src directory. The numerous installation scripts for various platforms have been replaced by the single configure script.

- All system calls now include "./", which makes the code portable to environments that do not include the current working directory in the path.

- The scripts have been updated to the more generic naming system using .poly_path and .multi_path, which allows for easier updating.

- The testruns have been updated to use the g03 rather than g98 as the default.