ISORROPIA v2.1
REFERENCE MANUAL

TABLE OF CONTENTS

ISORROPIA v2.1 REFERENCE MANUAL

Preface

For the most recent copies of the ISORROPIA computer code and manual, for reporting errors, for subscribing to the ISORROPIA mail list, or for any other question please contact one of the following persons:

Name: Dr. Christos Fountoukis

Address: Institute of Chemical Engineering & High Temperature Chemical Processes (ICE-HT), Foundation for Research & Technology, Hellas (FORTH)
            Stadiou St., Platani, P.O. Box 1414, 26504, Patras, GREECE

Telephone: +30 (2610) 965251

e-mail: cfountoukis@iceht.forth.gr

Name: Dr. Athanasios Nenes

Address: Schools of Earth and Atmospheric Sciences and Chemical & Biomolecular Engineering, Georgia Institute of Technology
            311 Ferst Drive NW, Atlanta, GA, 30332, USA

Telephone: +1 (404) 894-9225

e-mail: athanasios.nenes@ gatech.edu

Name: Dr. Spyros Pandis

Address: Department of Chemical Engineering and Engineering and Public Policy, Carnegie Mellon University, Pittsburgh, PA, 15213, USA

Telephone: +1 (412) 268-3531

e-mail: spyros@andrew.cmu.edu

Name: Dr. Christodoulos Pilinis

Address: Environmental Science Department, University of the Aegean
            Karadoni 17, Mytilene, 81100, GREECE

Telephone: +30 (2510) 36233

e-mail: xpil@aegean.gr

Acknowledgments

1. INTRODUCTION

This manual documents the usage, structure and programming interface of the ISORROPIA inorganic aerosol thermodynamic equilibrium model. This manual can serve either as a user's guide or a programmer's reference guide. All the information provided pertains to version 2.1 of ISORROPIA (dated May 26, 2009). In subsequent sections, the program structure and the file system of ISORROPIA are described in detail. Then, the program interface subroutines and global variables allocated in the COMMON blocks are presented. Finally, a few examples displaying the usage of ISORROPIA are given. One example demonstrates its usage as an independent program, while another displays how to call ISORROPIA from within another program. Finally, another example outlines the basic principles for embedding ISORROPIA into larger air quality (or global) models.

2. ISORROPIA OVERVIEW

Introduction

ISORROPIA was originally developed at the Division of Marine and Atmospheric Chemistry of the Rosenstiel School of Marine and Atmospheric Science, University of Miami. The extensions to include crustal species (Ca, K, Mg) were developed at the Georgia Institute of Technology, and are included in ISORROPIA-II (hereon called "ISORROPIA 2.1"). The objectives were to develop an aerosol thermodynamics model that balances accuracy with computational efficiency. The complete theory of ISORROPIA, together with a detailed description of the equations solved, the activity coefficient calculation methods and the computational algorithms used is presented elsewhere (Nenes *et al.*, 1998a,b; Fountoukis and Nenes, 2007). The performance and advantages of ISORROPIA over the usage of other thermodynamic equilibrium codes is assessed by Nenes *et al.* (1998b), Ansari and Pandis (1999a) and Fountoukis and Nenes (2007).

ISORROPIA can solve two classes of problems:

- *Forward problems*, in which known quantities are temperature, relative humidity and the total (i.e. gas+aerosol) concentrations of NH_3, H_2SO_4, Na, HCl and HNO_3. The concentrations of these precursors refer to the total amount in the gas and aerosol phases.

- *Reverse problems*, in which known quantities are temperature, relative humidity and the aerosol phase concentrations of NH_3, H_2SO_4, Na, HCl and HNO_3. These concentrations refer to the total amount of these precursors in the aerosol phase. For example, under certain

conditions, one could find NH_4^+ present in both the aqueous and solid phases. The NH_3 would then be the sum of ammonium found in both phases of the aerosol.

For both problem classes, ISORROPIA predicts which species can exist in the gas and aerosol phase and calculates their concentrations at chemical equilibrium. The user can specify the aerosol to be either in a thermodynamically stable state, where salts precipitate if saturation is exceeded, or in a metastable state, where salts do not precipitate under supersaturated conditions. The aerosol in the first case can be solid, liquid or both, while the second case is always an aqueous solution.

The source code of ISORROPIA is written primarily in FORTRAN 77. The only extensions to the ANSI standard implemented were the usage of `INCLUDE` statements, trailing comments, and variable names with length larger than 6 characters (but not larger than 9). These extensions are supported by the majority of FORTRAN 77 compilers, and are also included in the Fortran 90 standard, so they should not pose any portability problems. The model has extensively been tested on many platforms (DEC Alpha, HP and IBM RISC workstations under UNIX, Intel-based PC's under MS-DOS and Windows 95) and compilers (both FORTRAN 77 and Fortran 90). The results during these tests were found to be independent of the system used.

Filing system

The filing system of ISORROPIA is presented in Figure 1. Four files, `ISOCOM.FOR`, `ISOFWD.FOR`, `ISOREV.FOR` and `ISRPIA.INC` contain the ISORROPIA source code. The `.FOR` files consist the program segments (subroutines, functions, block data), while the `.INC` file contains the definition of the `COMMON` blocks. Of these files, the three `.FOR` are those that

have to be compiled. The `.INC` file is automatically read by the compiler during compilation (via `INCLUDE` statements) and should not be separately compiled.

Apart from the model source code, a driver is provided that allows the interactive execution of ISORROPIA. This driver is contained in files `MAIN.FOR`, `MAIN.INC`, and has been designed to be as versatile as possible. Input can be provided either directly from the console (interactive mode), or from an input file (batch mode). The results can be saved in two formats: the first being a text report, and the second being a format that can easily be imported into a spreadsheet. Apart from these facilities, the user can generate customized output through an easily programmable interface. Finally, there are two files, `ISRPIA.CNF` and `DEMO.INP`, used by the driver. The first (`ISRPIA.CNF`) is a configuration file which is read by the driver during execution, and specifies all the solution parameters that can be changed by the user. The second (`DEMO.INP`) is a sample input file for batch mode runs.

SOURCE CODE **INPUT**

INTERACTIVE DRIVER

| MAIN.INC (COMMON block) | → | MAIN.FOR (Interactive/batch driver) |

INPUT DATA (Console or File)

ISRPIA.CNF (Model configuration file)

Programming Interface

OUTPUT

ISORROPIA MODEL

ISRPIA.INC (COMMON block)

ISOCOM.FOR (General purpose routines)

ISOFWD.FOR (Forward problem routines)

ISOREV.FOR (Reverse problem routines)

RESULTS.TXT (Results in report format)

RESULTS.DAT (Results in spreadsheet format)

User specified output (Optional)

Figure 1: The filing system of ISORROPIA

The interactive driver of ISORROPIA

Introduction

This section describes the interactive driver provided with this package. The driver (files `MAIN.FOR` and `MAIN.INC`) should be compiled and linked together with ISORROPIA, in order to create a stand-alone executable. The driver has been written in FORTRAN 77, using the same programming conventions and extensions of ISORROPIA, in order to ensure the same level of portability and compatibility.

Figure 1 illustrates how the driver interacts with ISORROPIA. Input can be provided either directly from the console (interactive mode), or from an input file (batch mode). The driver can save the results either as a comprehensive text report, or in a format for importing into spreadsheet programs.

The configuration file ISRPIA.CNF

During execution, the driver reads the configuration file `ISRPIA.CNF`, which controls the type of input/output generated, and also specifies solution control parameters. `ISRPIA.CNF` is a standard ASCII text file. An example of a configuration file is shown in List 1. If `ISRPIA.CNF` is not found in the current directory, the driver assumes default values, which are those presented in List 1.

The first two parameters determine the type of output generated by the driver. The first parameter, when set to `.T.`, leads to the generation of a comprehensive report file. When the same parameter is set to `.F.`, the report is written to the console. The second parameter specifies

whether a spreadsheet file is generated. The remaining parameters control the solution process, and are further explained in. Section 3.2.

List 1: A sample `ISRPIA.CNF` configuration file.

```
1.  *** [ INPUT/OUTPUT PARAMETERS ] ***
2.  CREATE REPORT IN FILE? (.T.=yes, .F.=no, on screen)
3.  .T.
4.  CREATE SPREADSHEET FILE? (.T.=yes, .F.=no)
5.  .T.
6.
7.  *** [ SOLUTION CONTROL ] ***
8.  Convergence criterion (DEFAULT:1.D-6)
9.  1.D-10
10. Max iterations for bisection method (DEFAULT:100)
11. 100
12. Sweeps for activity coef. calculation (DEFAULT:4)
13. 10
14. Accuracy in activity coef. calculation (DEFAULT:5e-2)
15. 1e-2
16. Subdivisions for root tracking (DEFAULT:5)
17. 5
18. Method for binary activity coefs (0-online, 1-tables) (DEFAULT:1)
19. 0
20. Force mass conservation to machine precision? (0-no, 1-yes) (DEFAULT:0)
21. 0
```

The input file name requested during execution specifies whether the driver will input from a file or the console. When no name is specified (i.e., a carriage return is given), the interactive mode of input is assumed, and data is read from the console. Any other value is assumed to be a filename, and the batch mode of input is activated. A basic difference between input modes is that the interactive mode reads data for only one run, while the batch mode has no limit in the number of runs it can process. The batch mode input file name should be compatible with the conventions of the operating system. For example, an input file named `INPUT.13.DAT` is accepted under UNIX, but not under MS-DOS. If the file specified does not exist, then the input mode is automatically switched to interactive, and the program requests input from the console. The format of batch mode input files is explained in section 2.3.3.

When interactive mode is activated, the report is saved in file "`SCREEN.TXT`" and the spreadsheet file is assumed to be "`SCREEN.DAT`". When batch mode is activated, the name

given for the input file is used to generate the name of the output files. The characters before the first "`.`" (if there is one) is used as the "*basename*". The driver then appends "`.txt`" to this basename for the report and "`.dat`" for the spreadsheet file. The whole input file name is used as "*basename*" if there is no "`.`" present. For example, if the input file is "`RESULTS.INP`", then the report will be saved in file "`RESULTS.TXT`" and the spreadsheet file will be saved in "`RESULTS.DAT`". Also, if the input name is "`RESULTS`", (without any "`.`" in its name), the same output files are generated. The name provided should also be compatible with the operating system conventions. Finally, if the output files already exist, then the results from the new runs are appended to them.

Batch mode input files

This section describes the format for creating batch mode input files. An example of such a file is given in List 2 and specifies an aerosol that contains 0.0 µmoles Na, 0.379 µmoles H_2SO_4, 0.758 µmoles NH_3, 0.0 µmoles HNO_3, 0.0 µmoles HCl, 0.01 µmoles Ca, 0.01 µmoles K and 0.0 µmoles Mg per m^3 of air. Ambient conditions are set to 90% relative humidity and 298K temperature. Furthermore, precipitation of salts is allowed.

Batch mode input files consist of two sections: an 8 line *header*, which specifies the concentration units given and the type of problem solved, and the *data section*, which could be unlimited in length. The data section provides input for specific runs, one line per run. Each line should contain 8 numbers, separated by any valid FORTRAN delimiter (space, comma or tab). As can be seen in List 2, these numbers are respectively, the concentration of sodium, sulfuric acid, ammonia, nitrate, hydrochloric acid, calcium, potassium, magnesium the ambient relative humidity (0-1 scale) and temperature (Kelvin). The concentration units, and the problem type

solved are those specified in the header. Output is controlled by the parameters specified in

`ISRPIA.CNF`.

List 2: Sample batch mode input file.

```
1.  Input units (0=umol/m3, 1=ug/m3)
2.  0
3.
4.  Problem type (0=forward, 1=reverse); Phase state (0=solid+liquid, 1=metastable)
5.  0, 0
6.
7.  NH4-SO4 system case
8.  Na   SO4     NH3     NO3    Cl   Ca    K     Mg   RH      TEMP
9.  0    0.379   0.758   0      0    0.01  0.01  0.0  0.90    298.0
```

Spreadsheet output format

The spreadsheet data file is a standard ASCII file, that uses double quotes (i.e., ") as text

delimiters and regular spaces as numerical delimiters. The first line is a header which provides a

label for each data column. Whenever ISORROPIA completes a run, one line of data is written

with all the information regarding the input and equilibrium concentrations. Table 1 contains a

detailed description of each column. All concentrations are expressed in $\mu g\ m^{-3}$ air.

Table 1: Spreadsheet file format

Column #	Quantity (concentrations in $\mu g\ m^{-3}$ air)
1	Total sodium (gas+aerosol) expressed as equivalent Na
2	Total sulfate (gas+aerosol) expressed as equivalent H_2SO_4
3	Total ammonium (gas+aerosol) expressed as equivalent NH_3
4	Total nitrate (gas+aerosol) expressed as equivalent HNO_3
5	Total chloride (gas+aerosol) expressed as equivalent HCl
6	Ambient relative humidity, expressed on a (0,1) scale
7	Ambient temperature, K
8	Gas phase NH_3
9	Gas phase HCl
10	Gas phase HNO_3

11	Solid phase NaCl
12	Solid phase $NaNO_3$
13	Solid phase Na_2SO_4
14	Solid phase $NaHSO_4$
15	Solid phase NH_4Cl
16	Solid phase NH_4NO_3
17	Solid phase $(NH_4)_2SO_4$
18	Solid phase NH_4HSO_4
19	Solid phase $(NH_4)_3H(SO_4)_2$
20	Solid phase $CaSO_4$
21	Solid phase $Ca(NO_3)_2$
22	Solid phase $CaCl_2$
23	Solid phase K_2SO_4
24	Solid phase $KHSO_4$
25	Solid phase KNO_3
26	Solid phase KCl
27	Solid phase $MgSO_4$
28	Solid phase $Mg(NO_3)_2$
29	Solid phase $MgCl_2$
30	Aqueous phase H^+
31	Aqueous phase Na^+
32	Aqueous phase NH_4^+
33	Aqueous phase Cl^-
34	Aqueous phase SO_4^{-2}
35	Aqueous phase HSO_4^-
36	Aqueous phase NO_3^-
37	Aqueous phase Ca^{2+}
38	Aqueous phase K^+
39	Aqueous phase Mg^{2+}
40	Total aerosol ammonium (solid + aqueous) expressed as equivalent NH_3
41	Total aerosol chloride (solid + aqueous) expressed as equivalent HCl
42	Total aerosol nitrate (solid + aqueous) expressed as equivalent HNO_3
43	Aerosol water content
44	Aerosol aqueous phase mass (water+dissolved salts)
45	Aerosol solid phase mass
46	ISORROPIA subcase

Creating customized output

Customized data output can be generated by using SUBROUTINE USROUT (List 3), which is found in file MAIN.FOR. The driver calls USROUT in three phases of a session. The formal INTEGER argument ICODE informs the programmer which phase USROUT is called from. First (ICODE=1), USROUT is called right before the driver performs any runs, so the modeler can OPEN any files, generate any initial output, etc. Then (ICODE=2) it is called right after a run, so any run-specific output can be generated. Finally, (ICODE=3) USROUT is called once before the driver terminates, and allows for any final output. By default, USROUT does not generate anything, but provides a basic BLOCK IF structure on which the modeler "builds" upon. An example of user output generation is given in section 7.3.

List 3: The default USROUT subroutine

```
1.  C=========================================================================
2.  C
3.  C *** ISORROPIA CODE
4.  C *** SUBROUTINE USROUT
5.  C *** THIS SUBROUTINE IS USED TO GENERATE CUSTOMIZED OUTPUT
6.  C
7.  C *** COPYRIGHT 1996-2006, UNIVERSITY OF MIAMI, CARNEGIE MELLON UNIVERSITY,
8.  C *** GEORGIA INSTITUTE OF TECHNOLOGY
9.  C *** WRITTEN BY ATHANASIOS NENES
10. C *** http://nenes.eas.gatech.edu/ISORROPIA
11. C
12. C=========================================================================
13. C
14.       SUBROUTINE USROUT (ICODE)
15.       INCLUDE 'isrpia.inc'
16.       INCLUDE 'main.inc'
17. C
18. C  INITIAL OUTPUT
19.       IF (ICODE.EQ.1) THEN
20. C
21. C  OUTPUT AFTER EACH RUN
22.       ELSE IF (ICODE.EQ.2) THEN
23. C
24. C  FINAL OUTPUT
25.       ELSE IF (ICODE.EQ.3) THEN
26.       ENDIF
27. C
28.       RETURN
29.       END
```

3. ISORROPIA PROGRAMMING REFERENCE

Equilibrium calculations are performed by calling `SUBROUTINE ISOROPIA`. This is mainly the routine in which a programmer will `CALL` (and most of the time, the only one he has to deal with). In addition, ISORROPIA v2.1 offers a set of utility subroutines which allows the modeler to configure, obtain information and access the error system of ISORROPIA. The routines are versatile in the sense that they can be called at any time, regardless if a run is going to be performed or not. Running ISORROPIA is simplified, since the `CALL` command has relatively few arguments. In addition, if ISORROPIA is embedded into a larger model, configuration and error access can be done from host model's routines. For example, configuration can be done once, in the initialization module of an airshed model, or the error system can be accessed from an airshed's error-handling routine. The interface routines and their calling syntax follows.

SUBROUTINE ISOROPIA

This is the routine that runs the ISORROPIA (v 2.1) model.

Calling Syntax

```
CALL ISOROPIA (WI, RHI, TEMPI, CNTRL, WT, GAS, AERLIQ, AERSLD,
                    SCASI, OTHER)
```

Input Variables

- **WI**, DOUBLE PRECISION array of length [8]. Concentration of precursors, expressed in moles m^{-3} air. Depending on the type of problem solved, `WI` contains either *gas+aerosol* or *aerosol only* concentrations.

Each array element contains:

`WI(1)`	sodium
`WI(2)`	sulfate
`WI(3)`	ammonium
`WI(4)`	nitrate
`WI(5)`	chloride
`WI(6)`	calcium
`WI(7)`	potassium
`WI(8)`	magnesium

- **RHI**, DOUBLE PRECISION variable. Ambient relative humidity expressed on a (0,1) scale.

- **TEMPI**, DOUBLE PRECISION variable. Ambient temperature expressed in Kelvins.

- **CNTRL**, DOUBLE PRECISION array of length [2]. Characterizes the problem solved.

`CNTRL(1)`	Defines the type of problem solved.
	0 - Forward problem is solved. In this case, array WI contains GAS and AEROSOL concentrations together.
	1 - Reverse problem is solved. In this case, array WI contains AEROSOL concentrations only.
`CNTRL(2)`	Defines the aerosol state
	0 - The aerosol can have both solid+liquid phases (deliquescent).
	1 - The aerosol is in only liquid state (metastable aerosol)

Output Variables

- **WT**, DOUBLE PRECISION array of length [8], total concentrations (gas+aerosol) of species, expressed in moles m^{-3} air. If the forward problem is solved (CNTRL(1)=0), array WT is identical to input array WI. Each array element contains:

`WT(1)`	total sodium
`WT(2)`	total sulfate
`WT(3)`	total ammonium
`WT(4)`	total nitrate

    `WT(5)`   total chloride
    `WT(6)`   total calcium
    `WT(7)`   total potassium
    `WT(8)`   total magnesium

- **GAS**, DOUBLE PRECISION array of length [03]. Gaseous species concentrations, expressed in moles m^{-3} air. Each array element contains:

  `GAS(1)` NH_3
  `GAS(2)` HNO_3
  `GAS(3)` HCl

- **AERLIQ**, DOUBLE PRECISION array of length [12]. Liquid aerosol species concentrations, expressed in moles m^{-3} air. Each array element contains:

  `AERLIQ(01)` $H^+_{(aq)}$
  `AERLIQ(02)` $Na^+_{(aq)}$
  `AERLIQ(03)` $NH_4^+{}_{(aq)}$
  `AERLIQ(04)` $Cl^-_{(aq)}$
  `AERLIQ(05)` $SO_4^{-2}{}_{(aq)}$
  `AERLIQ(06)` $HSO_4^-{}_{(aq)}$
  `AERLIQ(07)` $NO_3^-{}_{(aq)}$
  `AERLIQ(08)` H_2O
  `AERLIQ(09)` $NH_{3(aq)}$ (undissociated)
  `AERLIQ(10)` $HCl_{(aq)}$ (undissociated)
  `AERLIQ(11)` $HNO_{3(aq)}$ (undissociated)
  `AERLIQ(12)` $OH^-_{(aq)}$
  `AERLIQ(13)` $Ca^{2+}_{(aq)}$
  `AERLIQ(14)` $K^+_{(aq)}$
  `AERLIQ(15)` $Mg^{2+}_{(aq)}$

- **AERSLD**, DOUBLE PRECISION array of length [09]. Solid aerosol species concentrations, expressed in moles m^{-3} air. Each array element contains:

  `AERSLD(01)` $NaNO_{3(s)}$
  `AERSLD(02)` $NH_4NO_{3(s)}$

`AERSLD(03)`	$NaCl_{(s)}$
`AERSLD(04)`	$NH_4Cl_{(s)}$
`AERSLD(05)`	$Na_2SO_{4(s)}$
`AERSLD(06)`	$(NH_4)_2SO_{4(s)}$
`AERSLD(07)`	$NaHSO_{4(s)}$
`AERSLD(08)`	$NH_4HSO_{4(s)}$
`AERSLD(09)`	$(NH_4)_4H(SO_4)_{2(s)}$
`AERSLD(10)`	$CaSO_{4(s)}$
`AERSLD(11)`	$Ca(NO_3)_{2(s)}$
`AERSLD(12)`	$CaCl_{2(s)}$
`AERSLD(13)`	$K_2SO_{4(s)}$
`AERSLD(14)`	$KHSO_{4(s)}$
`AERSLD(15)`	$KNO_{3(s)}$
`AERSLD(16)`	$KCl_{(s)}$
`AERSLD(17)`	$MgSO_{4(s)}$
`AERSLD(18)`	$Mg(NO_3)_{2(s)}$
`AERSLD(19)`	$MgCl_{2(s)}$

- **SCASI**, CHARACTER*15 variable. The subcase which the input corresponds to.

- **OTHER**, DOUBLE PRECISION array of length [9]. Information regarding the solution.

`OTHER(1)` 0 - Aerosol is WET. This means that an aqueous phase exists, but does not exclude the existence of a solid phase.

1 - Aerosol is DRY.

`OTHER(2)` Total sulfate molar ratio, $\dfrac{\left[NH_3^{gas+aerosol}\right]+\left[Na^{gas+aerosol}\right]}{\left[H_2SO_4^{gas+aerosol}\right]}$

`OTHER(3)` Aerosol sulfate molar ratio, $\dfrac{\left[NH_3^{aerosol}\right]+\left[Na^{aerosol}\right]}{\left[H_2SO_4^{aerosol}\right]}$

`OTHER(4)` Total sodium molar ratio, $\dfrac{\left[Na^{gas+aerosol}\right]}{\left[H_2SO_4^{gas+aerosol}\right]}$

`OTHER(5)` Ionic strength of the aqueous aerosol (if it exists).

`OTHER(6)` Total number of calls to the activity coefficient calculation subroutine.

OTHER(7)   Total sulfate molar ratio with crustal species,

$$\frac{\left[NH_3^{\,gas+aerosol}\right]+\left[Ca^{\,gas+aerosol}\right]+\left[K^{\,gas+aerosol}\right]+\left[Mg^{\,gas+aerosol}\right]+\left[Na^{\,gas+aerosol}\right]}{\left[H_2SO_4^{\,gas+aerosol}\right]}$$

OTHER(8)   Crustals and sodium molar ratio,

$$\frac{\left[Ca^{\,gas+aerosol}\right]+\left[K^{\,gas+aerosol}\right]+\left[Mg^{\,gas+aerosol}\right]+\left[Na^{\,gas+aerosol}\right]}{\left[H_2SO_4^{\,gas+aerosol}\right]}$$

OTHER(9)   Crustal species molar ratio, $\dfrac{\left[Ca^{\,gas+aerosol}\right]+\left[K^{\,gas+aerosol}\right]+\left[Mg^{\,gas+aerosol}\right]}{\left[H_2SO_4^{\,gas+aerosol}\right]}$

SUBROUTINE SETPARM

This routine configures the solution parameters for subsequent runs of ISORROPIA.

Calling Syntax

```
CALL SETPARM (WFTYPI, IACALCI, EPSI, MAXITI, NSWEEPI, EPSACTI,
                   NDIVI, NADJI)
```

Input Variables

- **WFTYPI**, INTEGER variable. Defines the type of weighting algorithm for the solution in Mutual Deliquescence Regions (MDR's):

  0   MDR's are assumed dry. This is equivalent to the approach used by SEQUILIB.

  1   The solution is assumed "half" dry and "half" wet throughout the MDR.

  2   The solution is a relative-humidity weighted mean of the dry and wet solutions (as defined in Nenes *et al.*, 1998a)

  The default value of WFTYPI is 2.

- **IACALCI**, INTEGER variable. Method of activity coefficient calculation:

  0   Calculate coefficients during runtime.

  1   Use precalculated tables.

  The default value of IACALCI is 1.

- **EPSI,** DOUBLE PRECISION variable. Defines the convergence criterion for all iterative processes in ISORROPIA, except those for activity coefficient calculations (`EPSACTI` controls that). The default value of `EPSI` is 10^{-6}

- **MAXIT,** INTEGER variable. Defines the maximum number of iterations for all iterative processes in ISORROPIA, except those for activity coefficient calculations (`NSWEEPI` controls that). The default value of `MAXIT` is 100.

- **NSWEEPI,** INTEGER variable. Defines the maximum number of iterations for activity coefficient calculations. The default value of `NSWEEPI` is 4.

- **EPSACTI,** DOUBLE PRECISION variable. Defines the convergence criterion for activity coefficient calculations. The default value of `EPSACTI` is 5×10^{-2}.

- **NDIV,** INTEGER variable. Defines the number of subdivisions for root tracking, when the bisection method is used for numerically solving equilibrium equations. Normally this parameter should not have to change, but is included for completeness. The default value of `NDIV` is 5.

- **NADJI,** INTEGER variable. Defines whether ISORROPIA will force conservation of mass up to the precision of the machine.

   0   Normal calculation mode.
   1   After calculating the thermodynamic solution, "adjust" concentrations of species so that total conservation of mass is satisfied to machine precision (i.e., 10^{-14}).

   The default value of `NADJI` is 0.

NOTE: If negative values are given for a parameter, it is ignored and the current value is used instead.

*Output Variables***:** None

SUBROUTINE GETPARM

This subroutine obtains the current values of solution parameters.

Calling Syntax

```
CALL GETPARM (WFTYPI, IACALCI, EPSI, MAXITI, NSWEEPI, EPSACTI,
                     NDIVI, NADJI)
```

*Input Variables***:** None

Output Variables:

- **WFTYPI**, INTEGER variable. Defines the type of weighting algorithm for the solution in Mutual Deliquescence Regions (MDR's):

  0   MDR's are assumed dry. This is equivalent to the approach used by SEQUILIB.

  1   The solution is assumed "half" dry and "half" wet throughout the MDR.

  2   The solution is a relative-humidity weighted mean of the dry and wet solutions (as defined in Nenes *et al.*, 1998a)

- **IACALCI,** INTEGER variable. Method of activity coefficient calculation:

  0   Calculate coefficients during runtime.

  1   Use precalculated tables.

- **EPSI,** DOUBLE PRECISION variable. Defines the convergence criterion for all iterative processes in ISORROPIA, except those for activity coefficient calculations (EPSACTI controls that).

- **MAXIT,** INTEGER variable. Defines the maximum number of iterations for all iterative processes in ISORROPIA, except those for activity coefficient calculations (NSWEEPI controls that).

- **NSWEEPI,** INTEGER variable. Defines the maximum number of iterations for activity coefficient calculations.

- **EPSACTI,** DOUBLE PRECISION variable. Defines the convergence criterion for activity coefficient calculations.

- **NDIV,** INTEGER variable. Defines the number of subdivisions for root tracking, when the bisection method is used for numerically solving equilibrium equations. Normally this parameter should not have to change, but is included for completeness.

- **NADJI,** INTEGER variable. Defines whether ISORROPIA will force conservation of mass up to the precision of the machine.

0	Normal calculation mode.
1	After calculating the thermodynamic solution, "adjust" concentrations of species so that total conservation of mass is satisfied to machine precision (i.e., 10^{-14}).

SUBROUTINE ISERRINF

This is the routine that accesses the error system and obtains a copy of the error stack. To print error messages, use SUBROUTINE ERRSTAT.

Calling Syntax

```
CALL ISERRINF (ERRSTKI, ERRMSGI, NOFERI, STKOFL)
```

*Input Variables***:** None

Output Variables:

- **ERRSTKI**, INTEGER array of length [25]. This is the error stack, with the numerical codes of errors that may have occurred in a run, in reverse order of appearance. The size of the stack varies, but the user can obtain the exact length of it from SUBROUTINE ISORINF (parameter NERR).

- **ERRMSGI,** CHARACTER*40 array of length [25]. Contains information related to the corresponding element of array ERRSTKI. This information is used when SUBROUTINE ERRSTAT is used to printout error messages. The size of this array varies, but the user can obtain the exact length of it from SUBROUTINE ISORINF (parameter NERR).

- **NOFERI**, INTEGER variable. The number of errors (warning and fatal) that occurred in the most recent run. If no errors were detected, then NOFERI=0.

- **STKOFL**, LOGICAL variable. Stack overflow flag. When STKOFL=.TRUE., then more errors appeared than the stack could hold, and were not saved. Otherwise, STKOFL=.FALSE.

NOTE: If a fatal error has occurred, it is only one (fatal errors interrupt calculation) and it is the last error found in the error stack, i.e. the error code is found in ERRSTKI(NOFERI).

SUBROUTINE ERRSTAT

This routine is used for printing error messages.

Calling Syntax

```
CALL ERRSTAT (IO,IERR,ERRINF)
```

Input Variables

- **IO**, INTEGER variable. This is the unit number where the error message is going to be written to. If output is intended to be saved in a file, unit "IO" has to previously be connected via a FORTRAN OPEN statement.

- **IERR,** INTEGER variable. This is the error code of the error message printed. It is obtained from the error stack (look at paragraph 3.4 for more details).

- **ERRINF**, CHARACTER variable (of any length). The information related to the error code `IERR`. This information is obtained from the error stack (look at paragraph 3.4 for more details).

Output Variables: None

NOTE: An example of using `SUBROUTINE ERRSTAT` can be found in paragraph 8.1.

SUBROUTINE ISORINF

This routine obtains information about ISORROPIA (ex., version number and storage requirements).

Calling Syntax

```
 SUBROUTINE ISORINF (VERSI, NCMP, NION, NAQGAS, NSOL, NERR, TIN,
                             GRT)
```

*Input Variables***:** None

Output Variables

- **VERSI,** CHARACTER*14 variable. Version/date information of ISORROPIA code.

- **NCMP**, INTEGER variable. The number of components in input array `WI`.

- **NION**, INTEGER variable. The maximum number of ionic species in the aqueous phase.

- **NAQGAS**, INTEGER variable. The maximum number of undissociated species in the aqueous phase.

- **NSOL**, INTEGER variable. The maximum number of solids in the aerosol phase.

- **NERR**, INTEGER variable. The size of the error stack (maximum number of errors that can be traced in a run).

- **TIN**, DOUBLE PRECISION variable. The value used for a very small number.

- **GRT**, DOUBLE PRECISION variable. The value used for a very large number.

4. ISORROPIA STRUCTURE

This section describes the internal structure and calling sequence of the subroutines included in ISORROPIA. The main calling tree is shown in Figure 2, while the calling sequence for each "submodel" is shown in Figures 3 to 8. A more detailed analysis of each subcase is given in Chapter 5. The thermodynamic equations, together with mass conservation and electroneutrality are solved by a "CALC" subroutine that corresponds to the input conditions and problem type (forward or reverse).
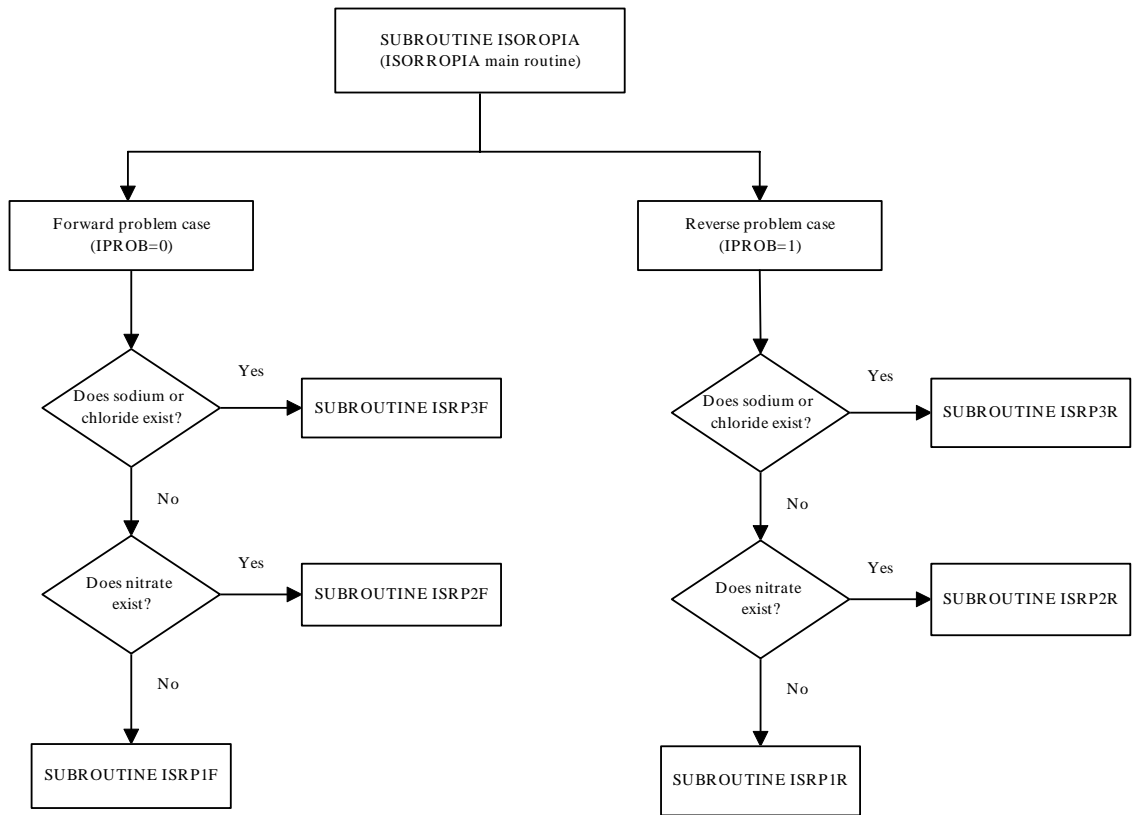

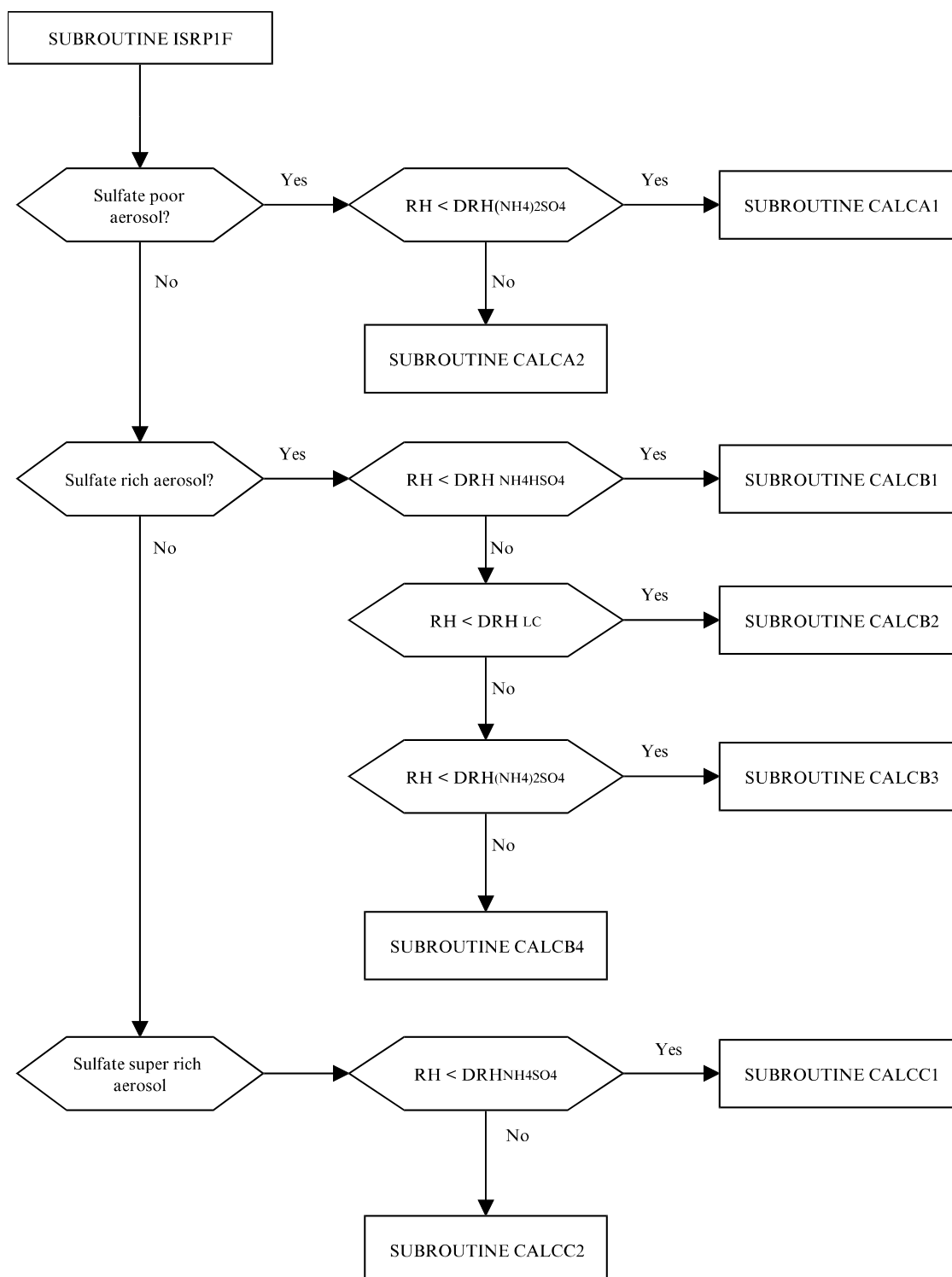Figure 2: The main calltree of ISORROPIA, with the six basic "submodels".

Figure 3: The calltree for solving the forward problem of an ammonium-sulfate aerosol.

SUBROUTINE ISRP2F

Sulfate poor aerosol?

Yes → RH < DRH NH4NO3

Yes → SUBROUTINE CALCD1

No

RH < DRH (NH4)2SO4

Yes → SUBROUTINE CALCD2

No

SUBROUTINE CALCD3

No

Sulfate rich aerosol?

Yes → RH < DRH NH4HSO4

Yes → SUBROUTINE CALCB1

No

RH < DRH LC

Yes → SUBROUTINE CALCB2

No

RH < DRH(NH4)2SO4

Yes → SUBROUTINE CALCB3

No

SUBROUTINE CALCB4

No

Sulfate super rich aerosol

RH < DRHNH4SO4

Yes → SUBROUTINE CALCC1
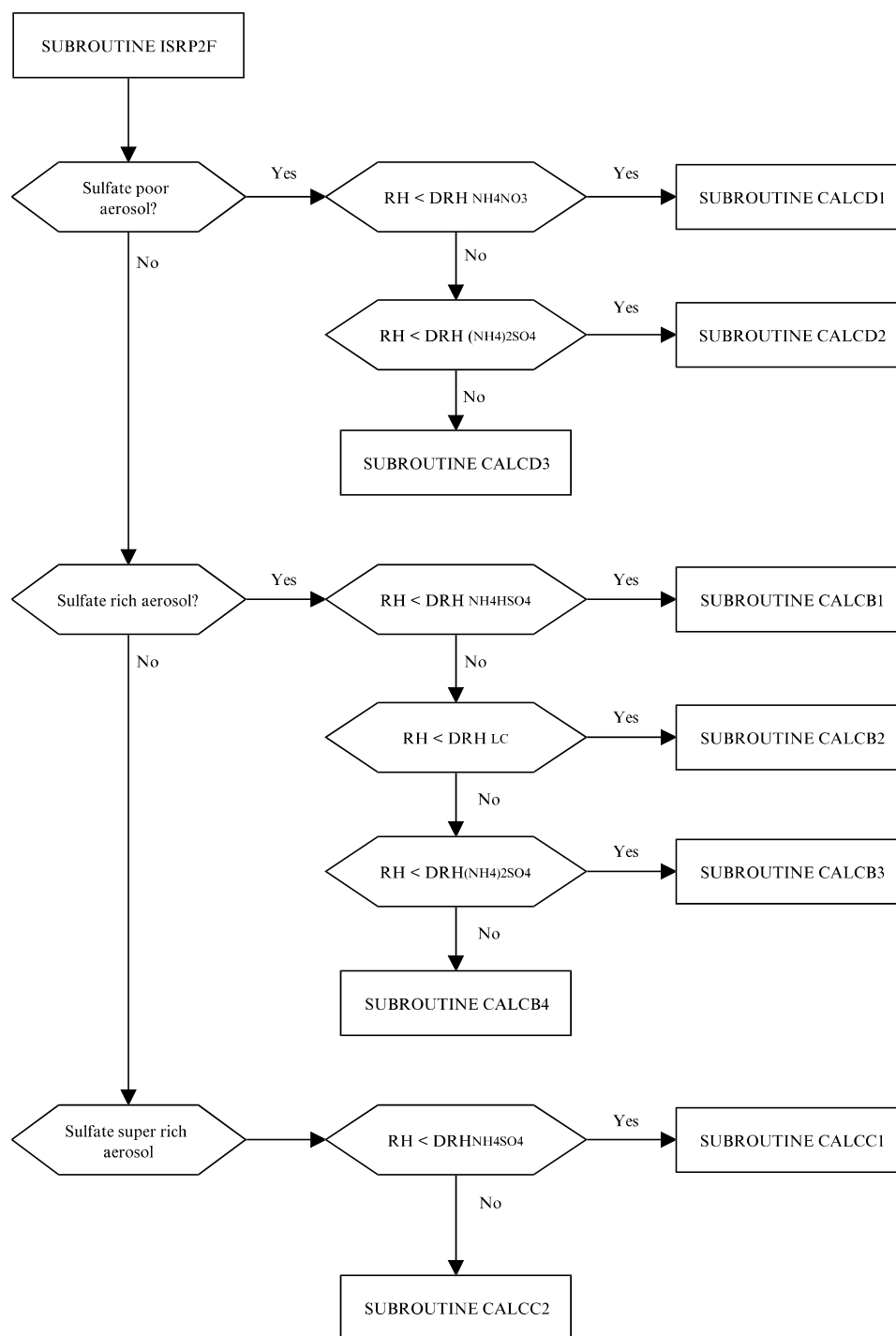
No

SUBROUTINE CALCC2

Figure 4:   The calltree for solving the forward problem of an ammonium-sulfate-nitrate aerosol.
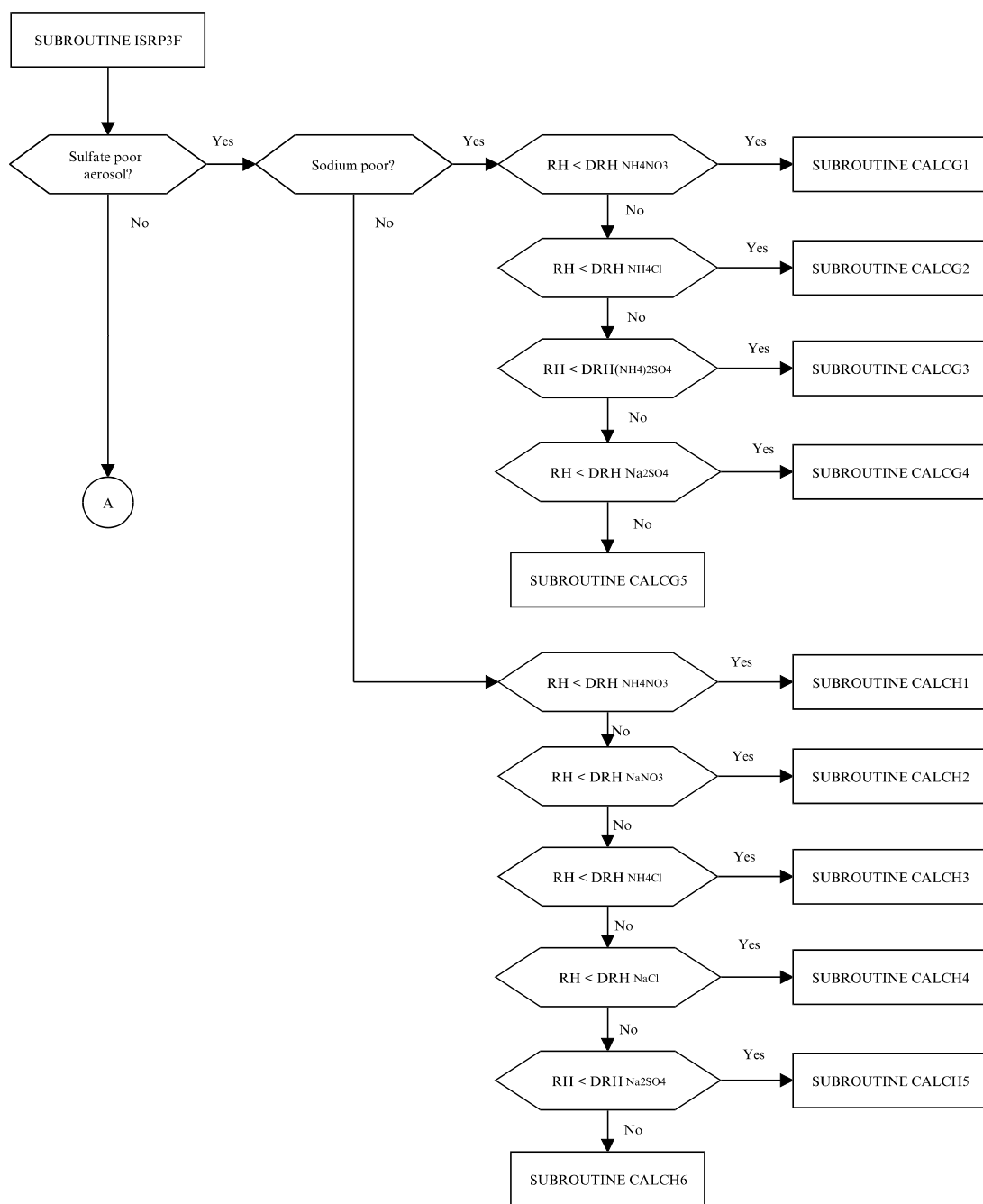
Figure 5   The calltree for solving the forward problem of a sodium-ammonium-sulfate-nitrate-chloride aerosol.
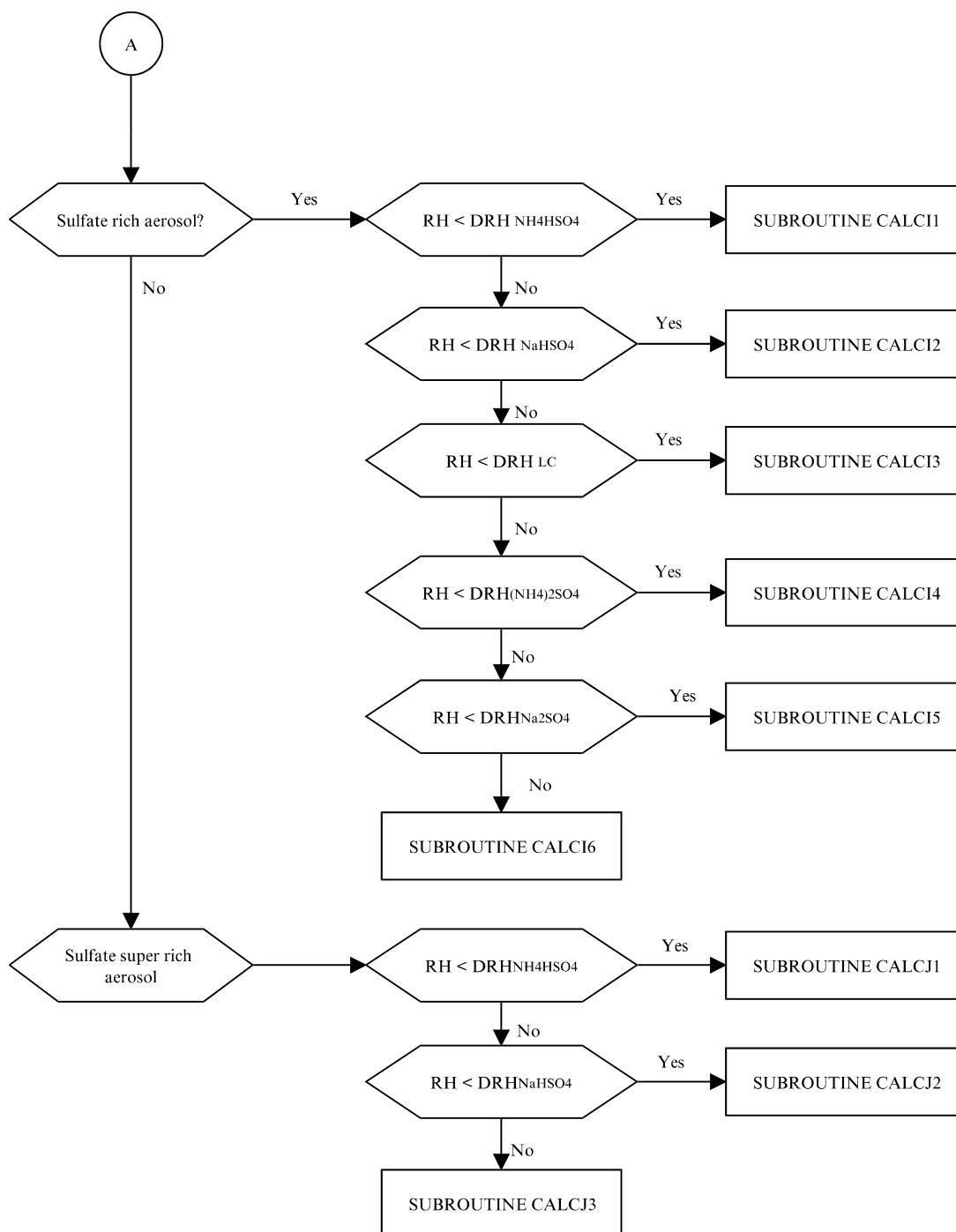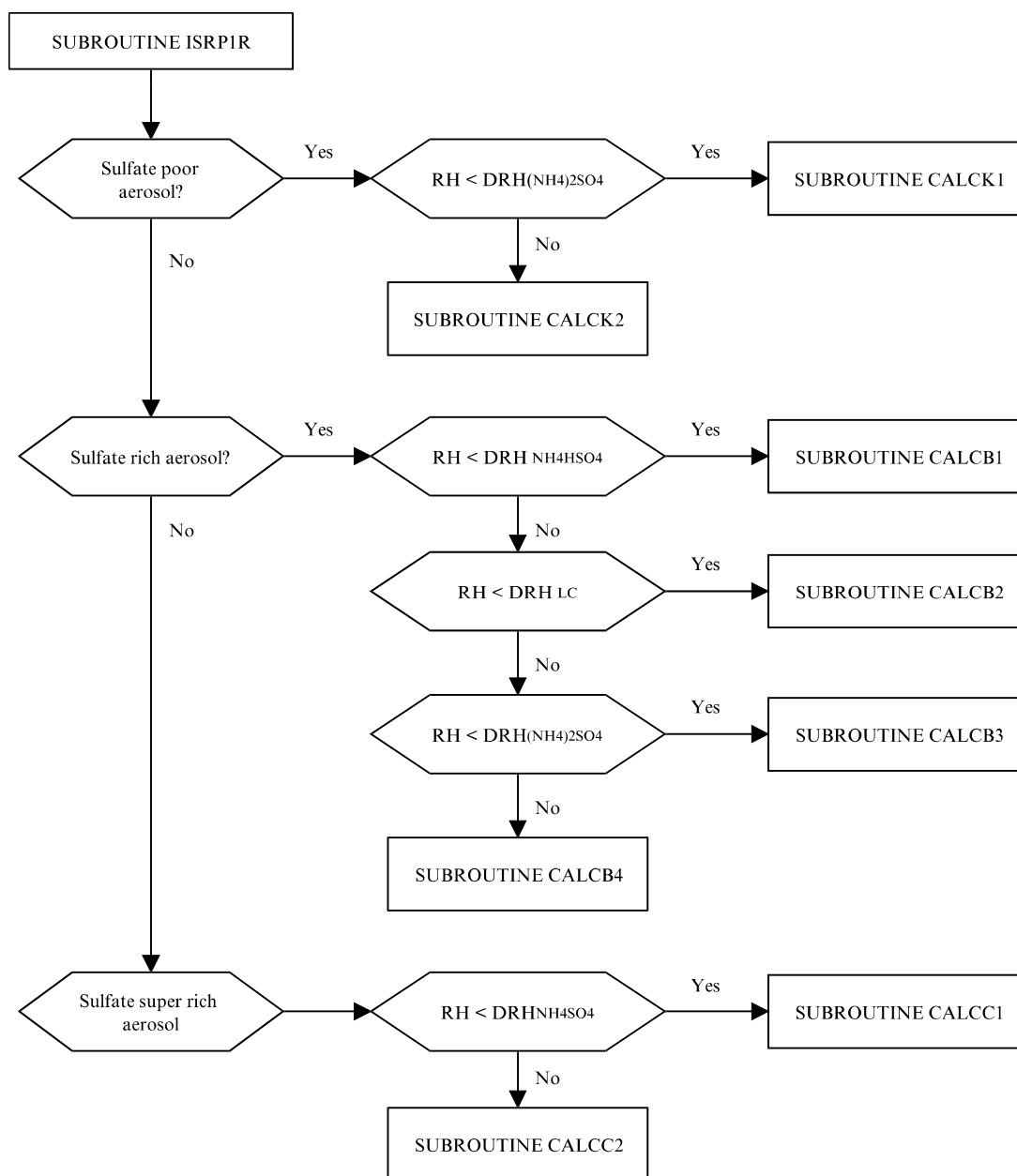
Figure 5   (Continued)

Figure 6: The calltree for solving the reverse problem of an ammonium-sulfate aerosol.

SUBROUTINE ISRP2R

Sulfate poor aerosol? —Yes→ RH < DRH NH4NO3 —Yes→ SUBROUTINE CALCN1

No (RH < DRH NH4NO3) ↓ → RH < DRH (NH4)2SO4 —Yes→ SUBROUTINE CALCN2

No ↓ SUBROUTINE CALCN3

No (Sulfate poor aerosol?) ↓

Sulfate rich aerosol? —Yes→ RH < DRH NH4HSO4 —Yes→ SUBROUTINE CALCB1

No ↓ RH < DRH LC —Yes→ SUBROUTINE CALCB2

No ↓ RH < DRH(NH4)2SO4 —Yes→ SUBROUTINE CALCB3

No ↓ SUBROUTINE CALCB4

No (Sulfate rich aerosol?) ↓

Sulfate super rich aerosol —→ RH < DRHNH4HSO4 —Yes→ SUBROUTINE CALCC1
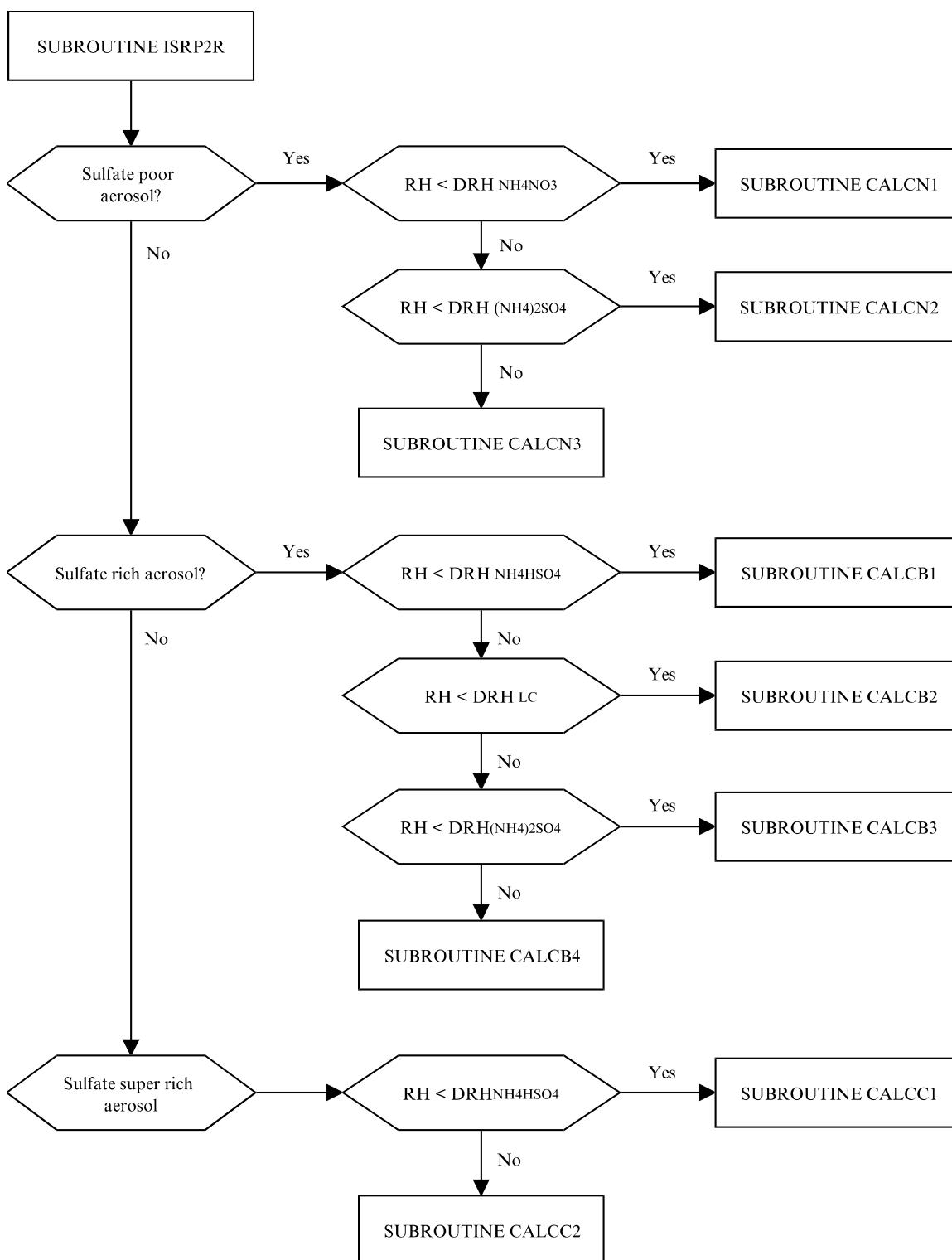
No ↓ SUBROUTINE CALCC2

Figure 7:   The calltree for solving the reverse problem of an ammonium-sulfate-nitrate aerosol.

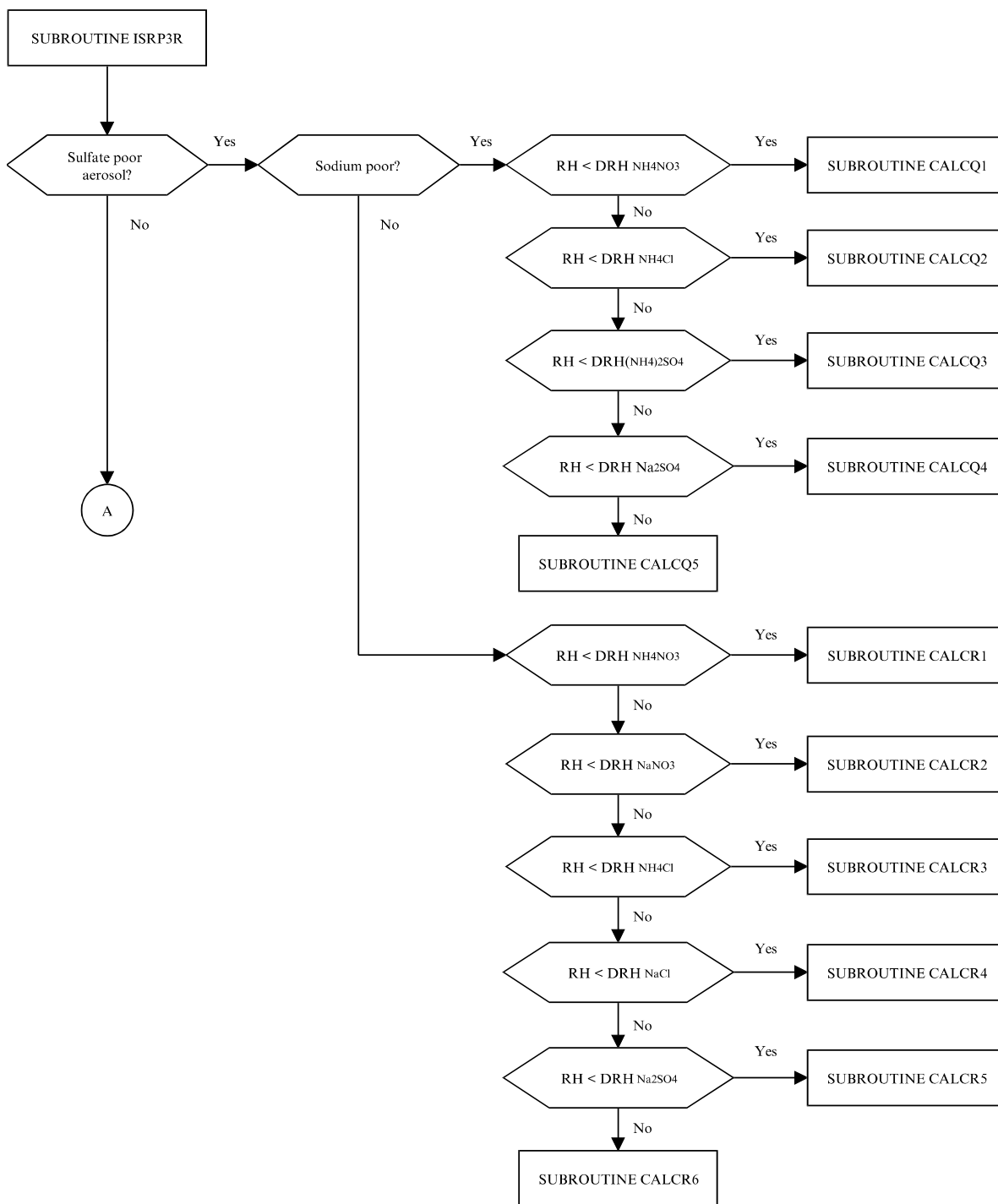Figure 8:   The calltree for solving the reverse problem of a sodium-ammonium-sulfate-nitrate-chloride aerosol.

A

Sulfate rich aerosol? —Yes→ RH < DRH NH4HSO4 —Yes→ SUBROUTINE CALCI1

No ↓

RH < DRH NaHSO4 —Yes→ SUBROUTINE CALCI2

No ↓

RH < DRH LC —Yes→ SUBROUTINE CALCI3

No ↓

RH < DRH(NH4)2SO4 —Yes→ SUBROUTINE CALCI4

No ↓

RH < DRHNa2SO4 —Yes→ SUBROUTINE CALCI5

No ↓

SUBROUTINE CALCI6

(No from Sulfate rich aerosol?) ↓

Sulfate super rich aerosol → RH < DRHNH4HSO4 —Yes→ SUBROUTINE CALCJ1

No ↓

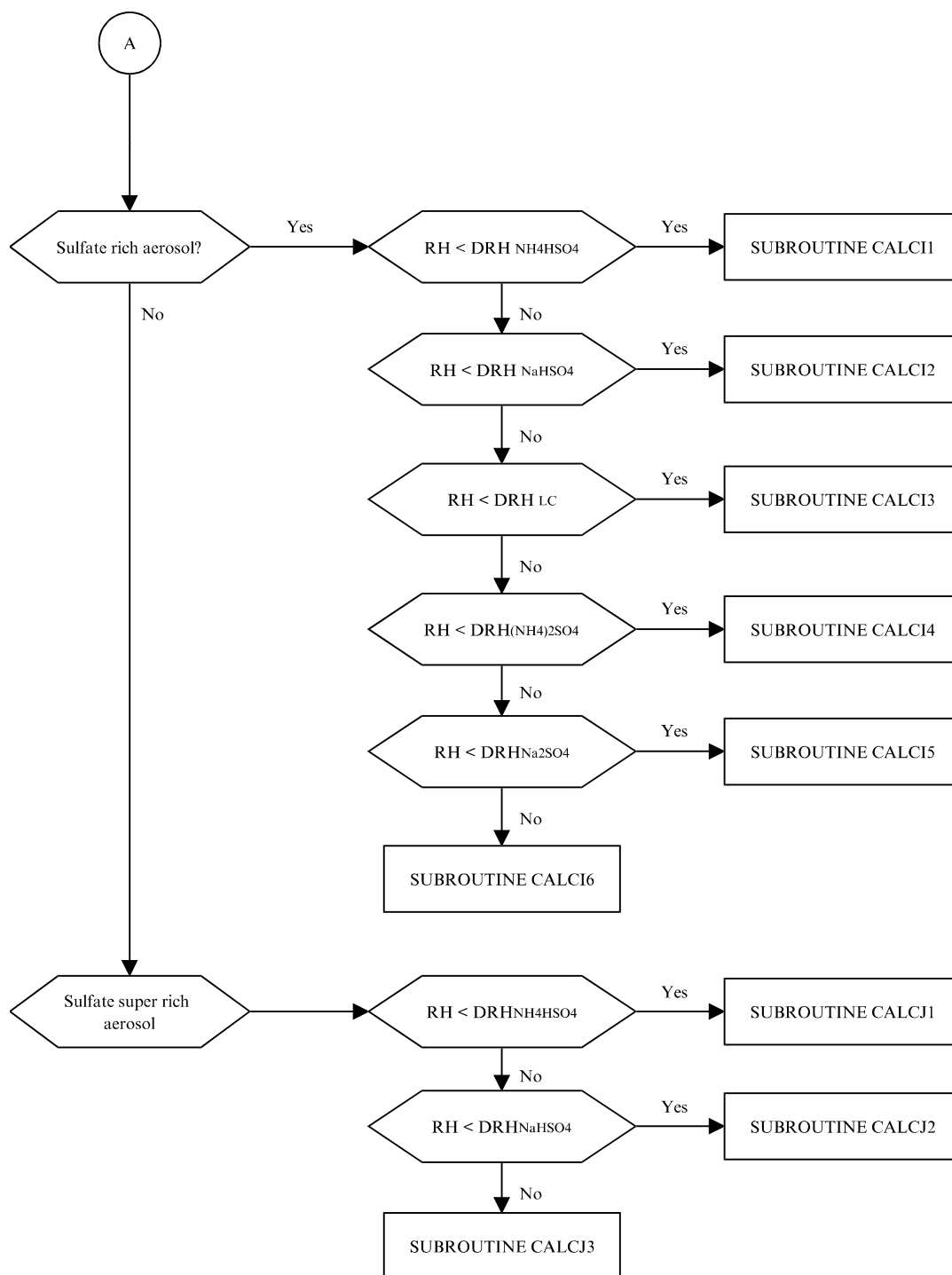RH < DRHNaHSO4 —Yes→ SUBROUTINE CALCJ2

No ↓

SUBROUTINE CALCJ3

Figure 8: (Continued)

5. SUBROUTINE REFERENCE

This section briefly describes most of the subroutines used in ISORROPIA. In the following routines, the term "objective function" refers to a function which is numerically solved for its root. The reader can refer to Nenes *et.al* (1998a) or Fountoukis and Nenes (2007) for a more detailed description of any terms.

SUBROUTINE   ISOROPIA, SETPARM, GETPARM, ISERRINF, ERRSTAT, ISORINF

These are the interface routines of ISORROPIA (v.1.1 and above). A detailed description of them can be found in Chapter 3.

SUBROUTINE ISOPLUS

This is the interface routine of ISORROPIA v.1.0, and is included for backward compatibility with models that use the calling conventions of this older version. This routine is described in detail in appendix B.

BLOCK DATA BLKISO

This subroutine provides initial (default) values to program parameters via data statements.

SUBROUTINE INIT1

This subroutine initializes the global variables for aerosol systems composed only of ammonium and sulfate.

SUBROUTINE INIT2

This subroutine initializes the global variables for aerosol systems composed only of ammonium, nitrate and sulfate.

SUBROUTINE INIT3

This subroutine initializes the global variables for aerosol systems composed of ammonium, sodium, chloride, nitrate and sulfate.

SUBROUTINE INIT4

This subroutine initializes the global variables for aerosol systems composed of ammonium, sodium, chloride, nitrate, sulfate, calcium, potassium and magnesium.

SUBROUTINE ISRP1F

This routine handles the forward problem for ammonium-sulfate aerosol systems. The possible species are determined from the sulfate ratio and ambient relative humidity. Cases solved by subroutines `CALCA1`, `CALCA2`, `CALCB1` through `CALCB4`, `CALCC1` and `CALCC2` are called here. The calling sequence is shown in Figure 3.

SUBROUTINE ISRP2F

This routine handles the forward problem for ammonium-nitrate-sulfate aerosol systems. The possible species are determined from the sulfate ratio and ambient relative humidity. Cases solved by subroutines `CALCD1` through `CALCD3`, `CALCB1` through `CALCB4`, `CALCC1` and `CALCC2` are called here. The calling sequence is shown in Figure 4.

SUBROUTINE ISRP3F

This routine handles the forward problem for sodium-ammonium-nitrate-sulfate-chloride aerosol systems. The possible species are determined from the sulfate ratio, the sodium ratio and from the ambient relative humidity. Cases solved by subroutines `CALCG1` through `CALCG5`, `CALCH1` through `CALCH6`, `CALCI1` through `CALCI6`, `CALCJ1` through `CALCJ3` are called here. The calling sequence is shown in Figure 5.

SUBROUTINE ISRP4F

This routine handles the forward problem for sodium-ammonium-nitrate-sulfate-chloride-calcium-potassium-magnesium aerosol systems. The possible species are determined from the total sulfate ratio, the crustals+sodium ratio, the crustals ratio and from the ambient relative humidity. Cases solved by subroutines `CALCO1` through `CALCO7`, `CALCM1` through `CALCM8`, `CALCP1` through `CALCP13`, `CALCL1` through `CALCL9`, `CALCK1` through `CALCK4` are called here.

SUBROUTINE ISRP1R

This routine handles the reverse problem for ammonium-sulfate aerosol systems. The possible species are determined from the sulfate ratio and ambient relative humidity. Cases solved by subroutines `CALCS1`, `CALCS2`, `CALCB1` through `CALCB4`, `CALCC1` and `CALCC2` are called here. Figure 6 displays the calling sequence.

SUBROUTINE ISRP2R

This subroutine is the driver routine for the reverse problem of an ammonium-nitrate-sulfate aerosol system. The possible species are determined from the sulfate ratio and ambient relative humidity. Cases solved by subroutines `CALCN1` through `CALCN3`, `CALCB1` through `CALCB4`, `CALCC1` and `CALCC2` are called here. The calling sequence is shown in Figure 7.

SUBROUTINE ISRP3R

This routine handles the reverse problem for sodium-ammonium-nitrate-sulfate-chloride aerosol systems. The possible species are determined form the sulfate ratio, the sodium ratio and from the ambient relative humidity. Cases solved by subroutines `CALCQ1` through `CALCQ5`, `CALCR1` through `CALCR6`, `CALCI1` through `CALCI6`, `CALCJ1` through `CALCJ3` are called here. The calling sequence is shown in Figure 8.

SUBROUTINE ISRP4R

This routine handles the reverse problem for sodium-ammonium-nitrate-sulfate-chloride-calcium-potassium-magnesium aerosol systems. The possible species are determined form the total sulfate ratio, the crustals+sodium ratio, the crustals ratio and from the ambient relative humidity. Cases solved by subroutines `CALCV1` through `CALCV7`, `CALCU1` through `CALCU8`, `CALCW1` through `CALCW13`, `CALCL1` through `CALCL9`, `CALCK1` through `CALCK4` are called here.

SUBROUTINE CALCACT

This subroutine calculates the activity coefficients in the aqueous phase using Bromley's method. Binary (single component) activity coefficients are calculated by the Kusik-Meissner relation,

either during runtime (`SUBROUTINE KMFUL`) or obtained from precalculated tables (`SUBROUTINE KMTAB`).

SUBROUTINE ADJUST

This subroutine adjusts to satisfy mass balance up to machine precision. It calculates the excess of each aerosol precursor, and if it exists, it adjusts sequentially aerosol phase species which contain the excess precursor.

SUBROUTINE CALCA1

This subroutine solves the forward problem of a sulfate poor, ammonium-sulfate aerosol system, when the relative humidity is below the DRH of $(NH_4)_2SO_4$. The following species are possible:

- Solids: $(NH_4)_2SO_4$

- Aqueous phase: Not present

Solid $(NH_4)_2SO_4$ is calculated from the amount of sulfates available. Excess ammonia is partitioned between gas and aqueous phase according to the $NH_{3(g)} \longleftrightarrow NH_{4(aq)}^+ + OH_{(aq)}^-$ equilibrium.

SUBROUTINE CALCA2

This subroutine solves the forward problem of a sulfate poor, ammonium-sulfate aerosol system, when the relative humidity is above the DRH of $(NH_4)_2SO_4$. The following species are possible:

- Solids: None

- Aqueous phase: Present

FUNCTION FUNCA2 calculates the concentration of ions from the $NH_{3(g)} \longleftrightarrow NH_{4(aq)}^{+} + OH_{(aq)}^{-}$ equilibrium, based on an estimation of H^{+}. This process is iterated with respect to the concentration of H^{+}, until electroneutrality is satisfied.

SUBROUTINE CALCB1

This subroutine solves the forward problem of a sulfate rich, ammonium-sulfate aerosol system, when the relative humidity is below the DRH of NH_4HSO_4. The following species are possible:

- Solids: $(NH_4)_3H(SO_4)_2$, $(NH_4)_2SO_4$, NH_4HSO_4

- Aqueous phase: Present when RH ≥ MDRH, not present when RH < MDRH

SUBROUTINE CALCB1A

This subroutine solves the forward problem of a sulfate rich, ammonium-sulfate aerosol system, when the relative humidity is below the MDRH regions of SUBROUTINE CALCB1. The following species are possible:

- Solids: $(NH_4)_3H(SO_4)_2$, NH_4HSO_4 or $(NH_4)_3H(SO_4)_2$, $(NH_4)_2SO_4$

- Aqueous phase: Not present

A material balance is performed which calculates the amount of $(NH_4)_3H(SO_4)_2$, $(NH_4)_2SO_4$ and NH_4HSO_4. Only two of the three salts can exist simultaneously, one of which is always $(NH_4)_3H(SO_4)_2$.

SUBROUTINE CALCB1B

This subroutine solves the forward problem of a sulfate rich, ammonium-sulfate aerosol system, when the relative humidity is in the MDRH regions of `SUBROUTINE CALCB1`. The following species are possible:

- Solids: $(NH_4)_3H(SO_4)_2$, NH_4HSO_4 or $(NH_4)_3H(SO_4)_2$, $(NH_4)_2SO_4$

- Aqueous phase: Present

The solution is assumed to be the weighted average of two states; one being the dry aerosol calculated by `SUBROUTINE CALCB1A`, and the other, a liquid phase calculated by `SUBROUTINE CALCB2`.

SUBROUTINE CALCB2

This subroutine solves the forward problem of a sulfate rich, ammonium-sulfate aerosol system, when the relative humidity is between the DRH of NH_4HSO_4 and $(NH_4)_3H(SO_4)_2$. The following species are possible:

- Solids: $(NH_4)_3H(SO_4)_2$, $(NH_4)_2SO_4$

- Aqueous phase: Present if in an MDRH, or if NH_4HSO_4 is possible. Otherwise, no aqueous phase is present.

There are two subcases possible, depending on the type of salts that can exist.

- When both $(NH_4)_3H(SO_4)_2$ and $(NH_4)_2SO_4$ are possible, `SUBROUTINE CALCB2A` is called.

- When only $(NH_4)_3H(SO_4)_2$ is possible, `SUBROUTINE CALCB2B` is called.

SUBROUTINE CALCB2A1

This subroutine solves the forward problem of a sulfate rich, ammonium-sulfate aerosol system, when the relative humidity is between the DRH of NH_4HSO_4 and $(NH_4)_3H(SO_4)_2$. The following species are possible:

- Solids: $(NH_4)_3H(SO_4)_2$, $(NH_4)_2SO_4$

- Aqueous phase: Present when RH $\geq$ MDRH, not present when RH < MDRH

SUBROUTINE CALCB2A2

This subroutine solves the forward problem of a sulfate rich, ammonium-sulfate aerosol system, when the relative humidity is above the DRH of NH_4HSO_4 and in the MDRH regions of `SUBROUTINE CALCB2A`. The following species are possible:

- Solids: $(NH_4)_3H(SO_4)_2$, $(NH_4)_2SO_4$

- Aqueous phase: Present

The solution is assumed to be the weighted average of two states; one being the dry aerosol calculated by `SUBROUTINE CALCB2A1`, and the other, a liquid phase calculated by `SUBROUTINE CALCB3`.

SUBROUTINE CALCB2B

This subroutine solves the forward problem of a sulfate rich, ammonium-sulfate aerosol system, when the relative humidity is between the DRH of NH_4HSO_4 and $(NH_4)_3H(SO_4)_2$. The following species are possible:

- Solids: $(NH_4)_3H(SO_4)_2$

- Aqueous phase: Present

The system is solved iteratively, using the amount of dissolved $(NH_4)_3H(SO_4)_2$ as the estimation variable. For each estimation, `FUNCTION FUNCB2A` calculates the ionic equilibria in the aqueous phase. The solubility product of $(NH_4)_3H(SO_4)_2$ is used as the objective function.

SUBROUTINE CALCB3

This subroutine solves the forward problem of a sulfate rich, ammonium-sulfate aerosol system, when the relative humidity lies between the DRH of $(NH_4)_3H(SO_4)_2$ and $(NH_4)_2SO_4$. The following species are possible:

- Solids: $(NH_4)_2SO_4$

- Aqueous phase: Present

There are two possible cases, depending on if solids can form. When solid $(NH_4)_2SO_4$ is possible (i.e. when the sulfate ratio is greater than 1.5), `SUBROUTINE CALCB3A` is called, otherwise, `SUBROUTINE CALCB3B` is called.

SUBROUTINE CALCB3A

This subroutine solves the forward problem of a sulfate rich, ammonium-sulfate aerosol system, when the relative humidity lies between the DRH of $(NH_4)_3H(SO_4)_2$ and $(NH_4)_2SO_4$. The following species are possible:

- Solids: $(NH_4)_2SO_4$

- Aqueous phase: Present

The system is solved iteratively, using the amount of dissolved $(NH_4)_2SO_4$ dissolved as the estimation variable. For each estimation, `FUNCTION FUNCB3A` calculates the ionic equilibria in the aqueous phase. The solubility product of $(NH_4)_2SO_4$ is used as the objective function.

SUBROUTINE CALCB3B

This subroutine solves the forward problem of a sulfate rich, ammonium-sulfate aerosol system, when the relative humidity lies between the DRH of $(NH_4)_3H(SO_4)_2$ and $(NH_4)_2SO_4$. The following species are possible:

- Solids: None

- Aqueous phase: Present

The speciation is controlled mainly by the $HSO_{4(aq)}^- \longleftrightarrow H_{(aq)}^+ + SO_{4(aq)}^{2-}$ equilibrium.

SUBROUTINE CALCB4

This subroutine solves the forward problem of a sulfate rich, ammonium-sulfate aerosol system, when the relative humidity is above the DRH of $(NH_4)_2SO_4$. The following species are possible:

- Solids: None

- Aqueous phase: Present

`FUNCTION FUNCB4` calculates the concentration of ions from the $NH_{3(g)} \longleftrightarrow NH_{4(aq)}^+ + OH_{(aq)}^-$ equilibrium, based on an estimation of H^+. This process is iterated with respect to the concentration of H^+, until electroneutrality is satisfied.

SUBROUTINE CALCC1

This subroutine solves the forward problem of a sulfate super-rich, ammonium-sulfate aerosol system, when the relative humidity is below the DRH of NH_4HSO_4. The following species are possible:

- Solids: NH_4HSO_4

- Aqueous phase: Present

The system is solved iteratively, using the amount of dissolved NH_4HSO_4 as the estimation variable. For each estimation, `FUNCTION FUNCC1A` calculates the ionic equilibria in the aqueous phase. The solubility product of NH_4HSO_4 is used as the objective function.

SUBROUTINE CALCC2

This subroutine solves the forward problem of a sulfate super-rich, ammonium-sulfate aerosol system, when the relative humidity is above the DRH of NH_4HSO_4. The following species are possible:

- Solids: None

- Aqueous phase: Present

The speciation is controlled mainly by the $HSO_{4(aq)}^{-} \longleftrightarrow H_{(aq)}^{+} + SO_{4(aq)}^{2-}$ equilibrium.

SUBROUTINE CALCD1

This subroutine solves the forward problem of a sulfate poor, ammonium-nitrate-sulfate aerosol system, when the relative humidity is below the DRH of NH_4NO_3. The following species are possible:

- Solids: $(NH_4)_2SO_4$, NH_4NO_3

- Aqueous phase: Present when RH $\geq$ MDRH, not present when RH $<$ MDRH

The case of a dry aerosol (RH $<$ MDRH) is solved by `SUBROUTINE CALCD1A`. The composition in the mutual deliquescence region is solved by `SUBROUTINE CALCMDRH`.

SUBROUTINE CALCD1A

This subroutine solves the forward problem of a sulfate poor, ammonium-nitrate-sulfate aerosol system, when the relative humidity is below the MDRH point of `SUBROUTINE CALCD1`. The following species are possible:

- Solids: $(NH_4)_2SO_4$, NH_4NO_3

- Aqueous phase: None

The amount of solid $(NH_4)_2SO_4$ is calculated from the sulfates. Solid NH_4NO_3 is calculated from the $NH_4NO_{3(s)} \longleftrightarrow NH_{3(g)} + HNO_{3(g)}$ equilibrium.

SUBROUTINE CALCD2

This subroutine solves the forward problem of a sulfate poor, ammonium-nitrate-sulfate aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and $(NH_4)_2SO_4$. The following species are possible:

- Solids: $(NH_4)_2SO_4$

- Aqueous phase: Present

The system is solved iteratively, using the amount of gaseous NH_3 as the estimation variable. Based on this estimation, `FUNCTION FUNCD2` calculates the ionic equilibria in the aqueous

phase and the amount of dissolved $(NH_4)_2SO_4$. The objective function used is the departure of $NH_4^+{}_{(aq)}$, $H^+{}_{(aq)}$ and $NH_{3(g)}$ concentrations from their equilibrium values, according to the reaction

$$NH_{3(g)} + H_2O_{(aq)} \longleftrightarrow NH_{4(aq)}^+ + OH_{(aq)}^-.$$

SUBROUTINE CALCD3

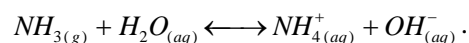This subroutine solves the forward problem of a sulfate poor, ammonium-nitrate-sulfate aerosol system, when the relative humidity is above the DRH of $(NH_4)_2SO_4$. The following species are possible:

- Solids: None

- Aqueous phase: Present

The system is solved iteratively, using the amount of gaseous NH_3 as the estimation variable. Based on this estimation, `FUNCTION FUNCD3` calculates the ionic equilibria in the aqueous phase. The objective function used is the departure of $NH_4^+{}_{(aq)}$, $H^+{}_{(aq)}$ and $NH_{3(g)}$ from the equilibrium state according to the reaction $NH_{3(g)} + H_2O_{(aq)} \longleftrightarrow NH_{4(aq)}^+ + OH_{(aq)}^-$.

SUBROUTINE CALCG1

This subroutine solves the forward problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of NH_4NO_3. The following species are possible:

- Solids: $(NH_4)_2SO_4$, NH_4NO_3, NH_4Cl, Na_2SO_4

- Aqueous phase: Present when RH $\geq$ MDRH, not present when RH $<$ MDRH

The case of a dry aerosol (RH < MDRH) is solved by `SUBROUTINE CALCG1A`. The composition in the mutual deliquescence region is solved by `SUBROUTINE CALCMDRH`.

SUBROUTINE CALCG1A

This subroutine solves the forward problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the MDRH region of `SUBROUTINE CALCG1`. The following species are possible:

- Solids: $(NH_4)_2SO_4$, NH_4NO_3, NH_4Cl, Na_2SO_4

- Aqueous phase: Not present

All of the sodium reacts with the sulfates to form Na_2SO_4. The remaining sulfates react with ammonia to form $(NH_4)_2SO_4$. The excess NH_3 then reacts with HNO_3 and HCl to form NH_4NO_3, NH_4Cl.

SUBROUTINE CALCG2

This subroutine solves the forward problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and NH_4Cl. The following species are possible:

- Solids: $(NH_4)_2SO_4$, NH_4Cl, Na_2SO_4

- Aqueous phase: Present when there is NH_4NO_3 (that deliquesces) or when in a MDRH.

In the presence of NH_4NO_3, the problem is solved by `SUBROUTINE CALCG2A`. When in a MDRH, `SUBROUTINE CALCMDRH` is called. Otherwise, the aerosol is dry, and `SUBROUTINE CALCG1A` is used to calculate the composition.

SUBROUTINE CALCG2A

This subroutine solves the forward problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and NH_4Cl. An aqueous phase is also presumed to exist . The following species are possible:

- Solids: $(NH_4)_2SO_4$, NH_4Cl, Na_2SO_4

- Aqueous phase: Present.

The system is solved iteratively, with respect to the amount of gas phase HCl. For each estimation, `FUNCTION FUNCG2A` calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated salts and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $NH_4^-{}_{(aq)}$, $HCl_{(g)}$ and $NH_{3(g)}$ from the equilibrium state, according to the reaction $NH_{3(g)} + HCl_{(g)} \longleftrightarrow NH_{4(aq)}^+ + Cl_{(aq)}^-$.

SUBROUTINE CALCG3

This subroutine solves the forward problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4Cl and $(NH_4)_2SO_4$. The following species are possible:

- Solids: $(NH_4)_2SO_4$, Na_2SO_4

- Aqueous phase: Present when there is some NH_4NO_3 or NH_4Cl (which deliquesces), or when in a MDRH.

In the presence of NH_4NO_3 or NH_4Cl, the problem is solved by `SUBROUTINE CALCG3A`. When in a MDRH, `SUBROUTINE CALCMDRH` is called. Otherwise, the aerosol is dry, and `SUBROUTINE CALCG1A` is used to calculate the composition.

SUBROUTINE CALCG3A

This subroutine solves the forward problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4Cl and $(NH_4)_2SO_4$. An additional presumption made is that nitrate and/or chloride exists in the aerosol phase. The following species are possible:

- Solids: $(NH_4)_2SO_4$, Na_2SO_4

- Aqueous phase: Present.

The system is solved iteratively, with respect to the amount of gas phase HCl. For each estimation, `FUNCTION FUNCG3A` calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated salts and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $NH_4^-{}_{(aq)}$, $HCl_{(g)}$ and $NH_{3(g)}$ from the equilibrium state, according to the reaction $NH_{3(g)} + HCl_{(g)} \longleftrightarrow NH_{4(aq)}^+ + Cl_{(aq)}^-$.

SUBROUTINE CALCG4

This subroutine solves the forward problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $(NH_4)_2SO_4$ and Na_2SO_4. The following species are possible:

- Solids: Na_2SO_4

- Aqueous phase: Present

The system is solved iteratively, with respect to the amount of gas phase HCl. For each estimation, `FUNCTION FUNCG4` calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated Na_2SO_4 and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $NH_4^-{}_{(aq)}$, $HCl_{(g)}$ and $NH_{3(g)}$ from equilibrium, according to the reaction

$$NH_{3(g)} + HCl_{(g)} \longleftrightarrow NH_{4(aq)}^+ + Cl_{(aq)}^-.$$

SUBROUTINE CALCG5

This subroutine solves the forward problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is larger than the DRH of Na_2SO_4. The following species are possible:

- Solids: None

- Aqueous phase: Present

The iterative solution algorithm of `SUBROUTINE CALCG4` is used, with the difference that Na_2SO_4 does not precipitate.
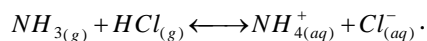
SUBROUTINE CALCH1

This subroutine solves the forward problem of a sulfate poor, sodium rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of NH_4NO_3. The following species are possible:

- Solids: Na_2SO_4, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present when RH $\geq$ MDRH, not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by `SUBROUTINE CALCH1A`. The composition in the mutual deliquescence region is solved by `SUBROUTINE CALCMDRH`.

SUBROUTINE CALCH1A

This subroutine solves the forward problem of a sulfate poor, sodium rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the MDRH regions of `SUBROUTINE CALCH1`. The following species are possible:

- Solids: Na_2SO_4, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Not present

The sodium reacts with all of the sulfates to produce Na_2SO_4. The remaining sodium first reacts to form $NaNO_3$, then if possible, $NaCl$. HCl and HNO_3 then react with NH_3 to form NH_4NO_3 and NH_4Cl.

SUBROUTINE CALCH2

This subroutine solves the forward problem of a sulfate poor, sodium rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and $NaNO_3$. The following species are possible:

- Solids: Na_2SO_4, $NaNO_3$, $NaCl$, NH_4Cl

- Aqueous phase: Present when NH_4NO_3 exists or when RH $\geq$ MDRH, not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by `SUBROUTINE CALCH1A`. The composition in the mutual deliquescence region is solved by `SUBROUTINE CALCMDRH`.

SUBROUTINE CALCH2A

This subroutine solves the forward problem of a sulfate poor, sodium rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and $NaNO_3$. An aqueous phase is also presumed to exist. The following species are possible:

- Solids: Na_2SO_4, $NaNO_3$, $NaCl$, NH_4Cl

- Aqueous phase: Present.

The system is solved iteratively, with respect to the amount of gas phase HCl. For each estimation, `FUNCTION FUNCH2A` calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated salts and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $NH_4^-{}_{(aq)}$, $HCl_{(g)}$ and $NH_{3(g)}$ from equilibrium, according to the reaction

$$NH_{3(g)} + HCl_{(g)} \longleftrightarrow NH^+_{4(aq)} + Cl^-_{(aq)}.$$
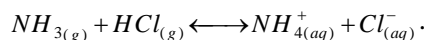
SUBROUTINE CALCH3

This subroutine solves the forward problem of a sulfate poor, sodium rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $NaNO_3$ and NaCl. An aqueous phase is presumed to exist. The following species are possible:

- Solids: Na_2SO_4, $NaCl$, NH_4Cl

- Aqueous phase: Present.

The iterative solution algorithm of `SUBROUTINE CALCH2` is used, with the difference that $NaNO_3$ does not precipitate.

SUBROUTINE CALCH4

This subroutine solves the forward problem of a sulfate poor, sodium rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NaCl and NH_4Cl. The following species are possible:

- Solids: Na_2SO_4, NH_4Cl

- Aqueous phase: Present.

The iterative solution algorithm of `SUBROUTINE CALCH2` is used, with the difference that $NaNO_3$ and NaCl do not precipitate.

SUBROUTINE CALCH5

This subroutine solves the forward problem of a sulfate poor, sodium rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4Cl and Na_2SO_4. The following species are possible:

- Solids: Na_2SO_4

- Aqueous phase: Present.

The iterative solution algorithm of `SUBROUTINE CALCH2` is used, with the difference that $NaNO_3$, NaCl and NH_4Cl do not precipitate.

SUBROUTINE CALCH6

This subroutine solves the forward problem of a sulfate poor, sodium rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is above the DRH of Na_2SO_4. The following species are possible:

- Solids: None

- Aqueous phase: Present.

The iterative solution algorithm of `SUBROUTINE CALCH2` is used, with the difference that no salts precipitate.

SUBROUTINE CALCI1

This subroutine solves the forward problem of a sulfate rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of NH_4HSO_4. The following species are possible:

- Solids: Na_2SO_4, $(NH_4)_2SO_4$, $NaHSO_4$, NH_4HSO_4, $(NH_4)_3H(SO_4)_2$

- Aqueous phase: Present when RH ≥ MDRH, not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is handled by `SUBROUTINE CALCI1A`, while the mutual deliquescence region is handled by `SUBROUTINE CALCMDRH`.

SUBROUTINE CALCI1A

This subroutine solves the forward problem of a sulfate rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of NH_4HSO_4. The following species are possible:

- Solids: Na_2SO_4, $(NH_4)_2SO_4$, $NaHSO_4$, NH_4HSO_4, $(NH_4)_3H(SO_4)_2$

- Aqueous phase: None

All of the sodium reacts with sulfate to produce Na_2SO_4 and $NaHSO_4$. The remaining sulfate reacts with ammonia to form $(NH_4)_2SO_4$, $(NH_4)_3H(SO_4)_2$ and/or NH_4HSO_4. HCl and HNO_3 remain in the gas phase.

SUBROUTINE CALCI2

This subroutine solves the forward problem of a sulfate rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4HSO_4 and $NaHSO_4$. The following species are possible:

- Solids: Na_2SO_4, $(NH_4)_2SO_4$, $NaHSO_4$, $(NH_4)_3H(SO_4)_2$

- Aqueous phase: Present when NH_4HSO_4 exists (which deliquesces) or when RH $\geq$ MDRH. Water not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by `SUBROUTINE CALCI1A`. The mutual deliquescence region is handled by `SUBROUTINE CALCMDRH`. If NH_4HSO_4 exists, then it is solved by `SUBROUTINE CALCI2A`.

SUBROUTINE CALCI2A

This subroutine solves the forward problem of a sulfate rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4HSO_4 and $NaHSO_4$. An aqueous phase is presumed to exist. The following species are possible:

- Solids: Na_2SO_4, $(NH_4)_2SO_4$, $NaHSO_4$, $(NH_4)_3H(SO_4)_2$

- Aqueous phase: Present

The system is solved iteratively, using the amount of dissolved $(NH_4)_3H(SO_4)_2$ as the estimation variable. For each estimation, FUNCTION FUNCI2A calculates the ionic equilibria in the aqueous phase. The solubility product of $(NH_4)_3H(SO_4)_2$ is used as the objective function.

SUBROUTINE CALCI3

This subroutine solves the forward problem of a sulfate rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $NaHSO_4$ and $(NH_4)_3H(SO_4)_2$. The following species are possible:

- Solids: Na_2SO_4, $(NH_4)_2SO_4$, $(NH_4)_3H(SO_4)_2$

- Aqueous phase: Present when NH_4HSO_4 or $NaHSO_4$ exists (dissolved) or when RH $\geq$ MDRH. Water not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by SUBROUTINE CALCI1A. The composition in the mutual deliquescence region is solved by SUBROUTINE CALCMDRH. If NH_4HSO_4 or $NaHSO_4$ exists, then the case is solved by SUBROUTINE CALCI3A.

SUBROUTINE CALCI3A

This subroutine solves the forward problem of a sulfate rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $NaHSO_4$ and $(NH_4)_3H(SO_4)_2$. Bisulfate salts are dissolved in an aerosol phase which is presumed to exist. The following species are possible:

- Solids: Na_2SO_4, $(NH_4)_2SO_4$, $(NH_4)_3H(SO_4)_2$

- Aqueous phase: Present

The iterative solution algorithm of `SUBROUTINE CALCI2A` is used, with the difference that

$NaHSO_4$ does not precipitate.

SUBROUTINE CALCI4

This subroutine solves the forward problem of a sulfate rich, sodium-ammonium-nitrate-sulfate-

chloride aerosol system, when the relative humidity is between the DRH of $(NH_4)_3H(SO_4)_2$ and

$(NH_4)_2SO_4$. The following species are possible:

- Solids: Na_2SO_4, $(NH_4)_2SO_4$

- Aqueous phase: Present

The system is solved iteratively, using the amount of dissolved Na_2SO_4 as an estimation variable.

For each estimation, `FUNCTION FUNCI4A` calculates the ionic equilibria in the aqueous phase.

The solubility product of Na_2SO_4 is used as the objective function.

SUBROUTINE CALCI5

This subroutine solves the forward problem of a sulfate rich, sodium-ammonium-nitrate-sulfate-

chloride aerosol system, when the relative humidity is between the DRH of $(NH_4)_2SO_4$ and

Na_2SO_4. The following species are possible:

- Solids: Na_2SO_4

- Aqueous phase: Present

The system is solved iteratively, using the same algorithm as in `SUBROUTINE CALCI4`. The

only difference is that $(NH_4)_2SO_4$ does not precipitate.

SUBROUTINE CALCI6

This subroutine solves the forward problem of a sulfate rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is above the DRH of Na_2SO_4. The following species are possible:

- Solids: None

- Aqueous phase: Present

The concentrations of species are mainly determined by the $HSO_{4(aq)}^- \longleftrightarrow H_{(aq)}^+ + SO_{4(aq)}^{2-}$ equilibrium.

SUBROUTINE CALCJ1

This subroutine solves the forward problem of a sulfate super-rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of NH_4HSO_4. The following species are possible:

- Solids: NH_4HSO_4, $NaHSO_4$

- Aqueous phase: Present

The system is solved iteratively, using the amount of dissolved $NaHSO_4$ as an estimation variable. For each estimation, `FUNCTION FUNCJ1` calculates the dissolved NH_4HSO_4 and the ionic equilibria in the aqueous phase. The solubility product of $NaHSO_4$ is used as the objective function.

SUBROUTINE CALCJ2

This subroutine solves the forward problem of a sulfate super-rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4HSO_4 and $NaHSO_4$. The following species are possible:

- Solids: $NaHSO_4$

- Aqueous phase: Present

The system is solved iteratively, using the amount of dissolved $NaHSO_4$ as an estimation variable. For each estimation, `FUNCTION FUNCJ2` calculates the ionic equilibria in the aqueous phase. The solubility product of $NaHSO_4$ is used as the objective function.

SUBROUTINE CALCJ3

This subroutine solves the forward problem of a sulfate super-rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is above the DRH of $NaHSO_4$. The following species are possible:

- Solids: None

- Aqueous phase: Present

Speciation is mainly controlled by the $HSO_{4\,(aq)}^{-} \longleftrightarrow H_{(aq)}^{+} + SO_{4\,(aq)}^{2-}$ equilibrium.

SUBROUTINE CALCO1

This subroutine solves the forward problem of a sulfate poor, crustals&sodium poor, calcim-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of NH_4NO_3. The following species are possible:

- Solids: $CaSO_4$, NH_4NO_3, $NH4Cl$, $(NH_4)_2SO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present when RH $\geq$ MDRH, not present when RH $<$ MDRH

The case of a dry aerosol (RH $<$ MDRH) is solved by SUBROUTINE CALCO1A. The composition in the mutual deliquescence region is solved by SUBROUTINE CALCMDRH2.

SUBROUTINE CALCO1A

This subroutine solves the forward problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the MDRH region of SUBROUTINE CALCO1. The following species are possible:

- Solids: $CaSO_4$, NH_4NO_3, NH_4Cl, $(NH_4)_2SO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Not present

All of the calcium, potassium, magnesium and sodium reacts with the sulfates to form $CaSO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4. The remaining sulfates react with ammonia to form $(NH_4)_2SO_4$. The excess NH_3 then reacts with HNO_3 and HCl to form NH_4NO_3, NH_4Cl.

SUBROUTINE CALCO2

This subroutine solves the forward problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and NH_4Cl. The following species are possible:

- Solids: $CaSO_4$, NH_4Cl, $(NH_4)_2SO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present when there is NH_4NO_3 (that deliquesces) or when in a MDRH.

In the presence of NH_4NO_3, the problem is solved by `SUBROUTINE CALCO2A`. When in a MDRH, `SUBROUTINE CALCMDRH2` is called. Otherwise, the aerosol is dry, and `SUBROUTINE CALCO1A` is used to calculate the composition.

SUBROUTINE CALCO2A

This subroutine solves the forward problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and NH_4Cl. An aqueous phase is also presumed to exist. The following species are possible:

- Solids: $CaSO_4$, NH_4Cl, $(NH_4)_2SO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present.

The system is solved iteratively, with respect to the amount of gas phase HCl. For each estimation, `FUNCTION FUNCO2A` calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated salts and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $NH_{4\ (aq)}^-$, $HCl_{(g)}$ and $NH_{3(g)}$ from the equilibrium state, according to the reaction $NH_{3(g)} + HCl_{(g)} \longleftrightarrow NH_{4(aq)}^+ + Cl_{(aq)}^-$.

SUBROUTINE CALCO3

This subroutine solves the forward problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4Cl and $(NH_4)_2SO_4$. The following species are possible:

- Solids: $CaSO_4$, $(NH_4)_2SO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present when there is some NH_4NO_3 or NH_4Cl (which deliquesces), or when in a MDRH.

In the presence of NH_4NO_3 or NH_4Cl, the problem is solved by `SUBROUTINE CALCO3A`. When in a MDRH, `SUBROUTINE CALCMDRH2` is called. Otherwise, the aerosol is dry, and `SUBROUTINE CALCO1A` is used to calculate the composition.

SUBROUTINE CALCO3A

This subroutine solves the forward problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4Cl and $(NH_4)_2SO_4$. An additional presumption made is that nitrate and/or chloride exists in the aerosol phase. The following species are possible:

- Solids: $CaSO_4$, $(NH_4)_2SO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present.

The system is solved iteratively, with respect to the amount of gas phase HCl. For each estimation, `FUNCTION FUNCO3A` calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated salts and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $NH_4^-{}_{(aq)}$, $HCl_{(g)}$ and $NH_{3(g)}$ from the equilibrium state, according to the reaction $NH_{3(g)} + HCl_{(g)} \longleftrightarrow NH_{4(aq)}^+ + Cl_{(aq)}^-$.

SUBROUTINE CALCO4

This subroutine solves the forward problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $(NH_4)_2SO_4$ and $MgSO_4$. The following species are possible:

- Solids: $CaSO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present

The system is solved iteratively, with respect to the amount of gas phase HCl. For each estimation, `FUNCTION FUNCO4` calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated $MgSO_4$, Na_2SO_4, K_2SO_4 and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $HCl_{(g)}$ from equilibrium, according to the reaction $HCl_g \longleftrightarrow H^+_{(aq)} + Cl^+_{(aq)}$.

SUBROUTINE CALCO5

This subroutine solves the forward problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $MgSO_4$ and Na_2SO_4. The following species are possible:

- Solids: $CaSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present

The iterative solution algorithm of `SUBROUTINE CALCO4` is used, with the difference that $MgSO_4$ does not precipitate.

SUBROUTINE CALCO6

This subroutine solves the forward problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of Na_2SO_4 and K_2SO_4. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4

- Aqueous phase: Present

The iterative solution algorithm of `SUBROUTINE CALCO5` is used, with the difference that Na_2SO_4 does not precipitate.

SUBROUTINE CALCO7

This subroutine solves the forward problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is above the DRH of K_2SO_4. The following species are possible:

- Solids: $CaSO_4$

- Aqueous phase: Present

The iterative solution algorithm of `SUBROUTINE CALCO6` is used, with the difference that K_2SO_4 does not precipitate.

SUBROUTINE CALCM1

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of NH_4NO_3. The following species are possible:

- Solids: $CaSO_4$, NH_4NO_3, NH_4Cl, $MgSO_4$, Na_2SO_4, K_2SO_4, $NACl$, $NaNO_3$

- Aqueous phase: Present when RH $\geq$ MDRH, not present when RH $<$ MDRH

The case of a dry aerosol (RH $<$ MDRH) is solved by `SUBROUTINE CALCM1A`. The composition in the mutual deliquescence region is solved by `SUBROUTINE CALCMDRH2`.

SUBROUTINE CALCM1A

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the MDRH regions of `SUBROUTINE CALCM1`. The following species are possible:

- Solids: $CaSO_4$, NH_4NO_3, NH_4Cl, $MgSO_4$, Na_2SO_4, K_2SO_4, NACl, $NaNO_3$
- Aqueous phase: Not present

All of the calcium, potassium and magnesium reacts with the sulfates to form $CaSO_4$, $MgSO_4$, K_2SO_4. The remaining sulfates react with sodium to form Na_2SO_4. The remaining sodium first reacts to form $NaNO_3$, then if possible, NaCl. HCl and HNO_3 then react with NH_3 to form NH_4NO_3 and NH_4Cl.

SUBROUTINE CALCM2

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and $NaNO_3$. The following species are possible:

- Solids: $CaSO_4$, NH_4Cl, $MgSO_4$, Na_2SO_4, K_2SO_4, NACl, $NaNO_3$

- Aqueous phase: Present when NH_4NO_3 exists or when RH $\geq$ MDRH, not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by `SUBROUTINE  CALCM1A`. The composition in the mutual deliquescence region is solved by `SUBROUTINE CALCMDRH2`.

SUBROUTINE CALCM2A

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and $NaNO_3$. An aqueous phase is also presumed to exist. The following species are possible:

- Solids: $CaSO_4$, NH_4Cl, $MgSO_4$, Na_2SO_4, K_2SO_4, NACl, $NaNO_3$

- Aqueous phase: Present.

The system is solved iteratively, with respect to the amount of gas phase HCl. For each estimation, `FUNCTION FUNCM2A` calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated salts and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $HCl_{(g)}$ from equilibrium, according to the reaction $HCl_g \longleftrightarrow H^+_{(aq)} + Cl^+_{(aq)}$.

SUBROUTINE CALCM3

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $NaNO_3$ and NaCl. An aqueous phase is presumed to exist. The following species are possible:

- Solids: $CaSO_4$, NH_4Cl, $MgSO_4$, Na_2SO_4, K_2SO_4, $NACl$

- Aqueous phase: Present.

The iterative solution algorithm of `SUBROUTINE CALCM2` is used, with the difference that $NaNO_3$ does not precipitate.

SUBROUTINE CALCM4

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NaCl and NH_4Cl. The following species are possible:

- Solids: $CaSO_4$, NH_4Cl, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present.

The iterative solution algorithm of `SUBROUTINE CALCM3` is used, with the difference that NaCl does not precipitate.

SUBROUTINE CALCM5

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4Cl and $MgSO_4$. The following species are possible:

- Solids: $CaSO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCM4 is used, with the difference that NH$_4$Cl does not precipitate.

SUBROUTINE CALCM6

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of MgSO$_4$ and Na$_2$SO$_4$. The following species are possible:

- Solids: CaSO$_4$, Na$_2$SO$_4$, K$_2$SO$_4$

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCM5 is used, with the difference that MgSO$_4$ does not precipitate.

SUBROUTINE CALCM7

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of Na$_2$SO$_4$ and K$_2$SO$_4$. The following species are possible:

- Solids: CaSO$_4$, K$_2$SO$_4$

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCM6 is used, with the difference that Na$_2$SO$_4$ does not precipitate.

SUBROUTINE CALCM8

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is above the DRH of K_2SO_4. The following species are possible:

- Solids: $CaSO_4$

- Aqueous phase: Present.

The iterative solution algorithm of `SUBROUTINE CALCM7` is used, with the difference that K_2SO_4 does not precipitate.

SUBROUTINE CALCP1

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of $CaCl_2$. The following species are possible:

- Solids: $CaSO_4$, $Ca(NO_3)_2$, $CaCl_2$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $MgCl_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present when RH $\geq$ MDRH, not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by `SUBROUTINE CALCP1A`. The composition in the mutual deliquescence region is solved by `SUBROUTINE CALCMDRH2`.

SUBROUTINE CALCP1A

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when

the relative humidity is below the MDRH regions of SUBROUTINE CALCP1. The following species are possible:

- Solids: $CaSO_4$, $Ca(NO_3)_2$, $CaCl_2$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $MgCl_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Not present

Calcium, potassium and magnesium react with the sulfates to form $CaSO_4$, $MgSO_4$, K_2SO_4. The remaining calcium, potassium and magnesium first reacts to form $Ca(NO_3)_2$, $Mg(NO_3)_2$, KNO_3 then if possible, $CaCl_2$, $MgCl_2$, KCl. Na and NH_3 then reacts to form $NaNO_3$, $NaCl$, and NH_4NO_3 and NH_4Cl.

SUBROUTINE CALCP2

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $CaCl_2$ and $MgCl_2$. The following species are possible:

- Solids: $CaSO_4$, $Ca(NO_3)_2$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $MgCl_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present when $CaCl_2$ exists or when RH ≥ MDRH, not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by SUBROUTINE CALCP1A. The composition in the mutual deliquescence region is solved by SUBROUTINE CALCMDRH2.

SUBROUTINE CALCP2A

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $CaCl_2$ and $MgCl_2$. An aqueous phase is also presumed to exist. The following species are possible:

- Solids: $CaSO_4$, $Ca(NO_3)_2$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $MgCl_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present.

The system is solved iteratively, with respect to the amount of gas phase HCl. For each estimation, `FUNCTION FUNCP2A` calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated salts and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $HCl_{(g)}$ from equilibrium, according to the reaction $HCl_g \longleftrightarrow H^+_{(aq)} + Cl^+_{(aq)}$.

SUBROUTINE CALCP3

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $MgCl_2$ and $Ca(NO_3)_2$. The following species are possible:

- Solids: $CaSO_4$, $Ca(NO_3)_2$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present when nitrates or chlorides exist or when RH $\geq$ MDRH, not present when RH < MDRH.

The case of a dry aerosol (RH < MDRH) is solved by SUBROUTINE CALCP1A. The composition in the mutual deliquescence region is solved by SUBROUTINE CALCMDRH2.

SUBROUTINE CALCP3A

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $MgCl_2$ and $Ca(NO_3)_2$. An aqueous phase is presumed to exist. The following species are possible:

- Solids: $CaSO_4$, $Ca(NO_3)_2$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCP2A is used, with the difference that $MgCl_2$ does not precipitate.

SUBROUTINE CALCP4

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $Ca(NO_3)_2$ and $Mg(NO_3)_2$. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present when nitrates exist or when RH $\geq$ MDRH, not present when RH < MDRH.

The case of a dry aerosol (RH < MDRH) is solved by SUBROUTINE CALCP1A. The composition in the mutual deliquescence region is solved by SUBROUTINE CALCMDRH2.

SUBROUTINE CALCP4A

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $Ca(NO_3)_2$ and $Mg(NO_3)_2$. An aqueous phase is presumed to exist. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCP3A is used, with the difference that $Ca(NO_3)_2$ does not precipitate.

SUBROUTINE CALCP5

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $Mg(NO_3)_2$ and NH_4NO_3. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present when nitrates exist or when RH $\geq$ MDRH, not present when RH < MDRH.

The case of a dry aerosol (RH < MDRH) is solved by `SUBROUTINE CALCP1A`. The composition in the mutual deliquescence region is solved by `SUBROUTINE CALCMDRH`.

SUBROUTINE CALCP5A

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $Mg(NO_3)_2$ and NH_4NO_3. An aqueous phase is presumed to exist. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl
- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCP4A is used, with the difference that $Mg(NO_3)_2$ does not precipitate.

SUBROUTINE CALCP6

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and $NaNO_3$. An aqueous phase is presumed to exist. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $NaNO_3$, $NaCl$, NH_4Cl
- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCP5A is used, with the difference that NH_4NO_3 does not precipitate.

SUBROUTINE CALCP7

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $NaNO_3$ and NaCl. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$, NaCl, NH_4Cl

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCP6 is used, with the difference that $NaNO_3$ does not precipitate.

SUBROUTINE CALCP8

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NaCl and NH_4Cl. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$, NH_4Cl

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCP7 is used, with the difference that NaCl does not precipitate.

SUBROUTINE CALCP9

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4Cl and KCl. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCP8 is used, with the difference that NH_4Cl does not precipitate.

SUBROUTINE CALCP10

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of KCl and $MgSO_4$. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, $MgSO_4$

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCP9 is used, with the difference that KCl does not precipitate.

SUBROUTINE CALCP11

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $MgSO_4$ and KNO_3. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCP10 is used, with the difference that $MgSO_4$ does not precipitate.

SUBROUTINE CALCP12

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of KNO_3 and K_2SO_4. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCP11 is used, with the difference that KNO_3 does not precipitate.

SUBROUTINE CALCP13

This subroutine solves the forward problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is above the DRH of K_2SO_4. The following species are possible:

- Solids: $CaSO_4$

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCP12 is used, with the difference that K_2SO_4 does not precipitate.

SUBROUTINE CALCL1

This subroutine solves the forward problem of a sulfate rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of NH_4HSO_4. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, $MgSO_4$, $KHSO_4$, Na_2SO_4, $(NH_4)_2SO_4$, $NaHSO_4$, NH_4HSO_4, $(NH_4)_3H(SO_4)_2$

- Aqueous phase: Present when $RH \geq MDRH$, not present when $RH < MDRH$

The case of a dry aerosol ($RH < MDRH$) is handled by SUBROUTINE CALCL1A, while the mutual deliquescence region is handled by SUBROUTINE CALCMDRH2.

SUBROUTINE CALCL1A

This subroutine solves the forward problem of a sulfate rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of NH_4HSO_4. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, $MgSO_4$, $KHSO_4$, Na_2SO_4, $(NH_4)_2SO_4$, $NaHSO_4$, NH_4HSO_4, $(NH_4)_3H(SO_4)_2$

- Aqueous phase: None

All of the calcium, magnesium, potassium and sodium reacts with sulfate to produce $CaSO_4$, K_2SO_4, $MgSO_4$, $KHSO_4$, Na_2SO_4 and $NaHSO_4$. The remaining sulfate reacts with ammonia to form $(NH_4)_2SO_4$, $(NH_4)_3H(SO_4)_2$ and/or NH_4HSO_4. HCl and HNO_3 remain in the gas phase.

SUBROUTINE CALCL2

This subroutine solves the forward problem of a sulfate rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4HSO_4 and $NaHSO_4$. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, $MgSO_4$, $KHSO_4$, Na_2SO_4, $(NH_4)_2SO_4$, $NaHSO_4$, $(NH_4)_3H(SO_4)_2$

- Aqueous phase: Present when NH_4HSO_4 exists (which deliquesces) or when RH $\geq$ MDRH. Water not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by SUBROUTINE CALCL1A. The mutual deliquescence region is handled by SUBROUTINE CALCMDRH2. If NH_4HSO_4 exists, then it is solved by SUBROUTINES CALCL2A and CALCL2B.

SUBROUTINE CALCL2A

This subroutine solves the forward problem of a sulfate rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4HSO_4 and $NaHSO_4$. An aqueous phase is presumed to exist. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, $MgSO_4$, $KHSO_4$, Na_2SO_4, $(NH_4)_2SO_4$, $NaHSO_4$, $(NH_4)_3H(SO_4)_2$

- Aqueous phase: Present

The system is solved iteratively, using the amount of dissolved $(NH_4)_3H(SO_4)_2$ as the estimation variable. For each estimation, FUNCTION FUNCL2A calculates the ionic equilibria in the aqueous phase. The solubility product of $(NH_4)_3H(SO_4)_2$ is used as the objective function.

SUBROUTINE CALCL2B

This subroutine solves the forward problem of a sulfate rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4HSO_4 and $NaHSO_4$. An aqueous phase is presumed to exist. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, $MgSO_4$, $KHSO_4$, Na_2SO_4, $(NH_4)_2SO_4$, $NaHSO_4$, $(NH_4)_3H(SO_4)_2$

- Aqueous phase: Present

The system is solved iteratively, using the amount of dissolved Na_2SO_4 as the estimation variable. For each estimation, `FUNCTION FUNCL2B` calculates the ionic equilibria in the aqueous phase. The solubility product of Na_2SO_4 is used as the objective function.

SUBROUTINE CALCL3

This subroutine solves the forward problem of a sulfate rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $NaHSO_4$ and $(NH_4)_3H(SO_4)_2$. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, $MgSO_4$, $KHSO_4$, Na_2SO_4, $(NH_4)_2SO_4$, $(NH_4)_3H(SO_4)_2$

- Aqueous phase: Present when NH_4HSO_4 or $NaHSO_4$ exists (dissolved) or when RH $\geq$ MDRH. Water not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by `SUBROUTINE CALCL1A`. The composition in the mutual deliquescence region is solved by `SUBROUTINE CALCMDRH2`. If NH_4HSO_4 or $NaHSO_4$ exists, then the case is solved by `SUBROUTINES CALCL3A and CALCL3B`.

SUBROUTINE CALCL3A

This subroutine solves the forward problem of a sulfate rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $NaHSO_4$ and $(NH_4)_3H(SO_4)_2$. Bisulfate salts are dissolved in an aerosol phase which is presumed to exist. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, $MgSO_4$, $KHSO_4$, Na_2SO_4, $(NH_4)_2SO_4$, $(NH_4)_3H(SO_4)_2$

- Aqueous phase: Present

The iterative solution algorithm of `SUBROUTINE CALCL2A` is used, with the difference that $NaHSO_4$ does not precipitate.

SUBROUTINE CALCL3B

This subroutine solves the forward problem of a sulfate rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $NaHSO_4$ and $(NH_4)_3H(SO_4)_2$. Bisulfate salts are dissolved in an aerosol phase which is presumed to exist. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, $MgSO_4$, $KHSO_4$, Na_2SO_4, $(NH_4)_2SO_4$, $(NH_4)_3H(SO_4)_2$

- Aqueous phase: Present

The iterative solution algorithm of `SUBROUTINE CALCL2B` is used, with the difference that $NaHSO_4$ does not precipitate.

SUBROUTINE CALCL4

This subroutine solves the forward problem of a sulfate rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $(NH_4)_3H(SO_4)_2$ and $(NH_4)_2SO_4$. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, $MgSO_4$, $KHSO_4$, Na_2SO_4, $(NH_4)_2SO_4$

- Aqueous phase: Present

The system is solved iteratively, using the amount of dissolved Na_2SO_4 as an estimation variable. For each estimation, `FUNCTION FUNCL4` calculates the ionic equilibria in the aqueous phase. The solubility product of Na_2SO_4 is used as the objective function.

SUBROUTINE CALCL5

This subroutine solves the forward problem of a sulfate rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $(NH_4)_2SO_4$ and $KHSO_4$. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, $MgSO_4$, $KHSO_4$, Na_2SO_4

- Aqueous phase: Present

The system is solved iteratively, using the same algorithm as in `SUBROUTINE CALCL4` with the difference that $(NH_4)_2SO_4$ does not precipitate.

SUBROUTINE CALCL6

This subroutine solves the forward problem of a sulfate rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $KHSO_4$ and $MgSO_4$. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, $MgSO_4$, Na_2SO_4

- Aqueous phase: Present

The system is solved iteratively, using the same algorithm as in `SUBROUTINE CALCL5` with the difference that $KHSO_4$ does not precipitate.

SUBROUTINE CALCL7

This subroutine solves the forward problem of a sulfate rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $MgSO_4$ and Na_2SO_4. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, Na_2SO_4

- Aqueous phase: Present

The system is solved iteratively, using the same algorithm as in `SUBROUTINE CALCL6` with the difference that $MgSO_4$ does not precipitate.

SUBROUTINE CALCL8

This subroutine solves the forward problem of a sulfate rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of Na_2SO_4 and K_2SO_4. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4

- Aqueous phase: Present

The system is solved iteratively, using the amount of dissolved K_2SO_4 as the estimation variable. For each estimation, `FUNCTION FUNCL8` calculates the ionic equilibria in the aqueous phase. The solubility product of K_2SO_4 is used as the objective function.

SUBROUTINE CALCL9

This subroutine solves the forward problem of a sulfate rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is above the DRH of K_2SO_4. The following species are possible:

- Solids: $CaSO_4$

- Aqueous phase: Present

The concentrations of species are mainly determined by the $HSO_{4(aq)}^{-} \longleftrightarrow H_{(aq)}^{+} + SO_{4(aq)}^{2-}$ equilibrium.

SUBROUTINE CALCK1

This subroutine solves the forward problem of a sulfate super-rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of NH_4HSO_4. The following species are possible:

- Solids: NH_4HSO_4, $NaHSO_4$, $KHSO_4$, $CaSO_4$

- Aqueous phase: Present

The system is solved iteratively, using the amount of dissolved $KHSO_4$ as an estimation variable. For each estimation, `FUNCTION FUNCK1` calculates the dissolved $KHSO_4$ and the ionic

equilibria in the aqueous phase. The solubility product of $KHSO_4$ is used as the objective function.

SUBROUTINE CALCK2

This subroutine solves the forward problem of a sulfate super-rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4HSO_4 and $NaHSO_4$. The following species are possible:

- Solids: $NaHSO_4$, $KHSO_4$,$CaSO_4$

- Aqueous phase: Present

The system is solved iteratively, using the amount of dissolved $KHSO_4$ as an estimation variable. For each estimation, `FUNCTION FUNCK2` calculates the dissolved $KHSO_4$ and the ionic equilibria in the aqueous phase. The solubility product of $KHSO_4$ is used as the objective function.

SUBROUTINE CALCK3

This subroutine solves the forward problem of a sulfate super-rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $NaHSO_4$ and $KHSO_4$. The following species are possible:

- Solids: $KHSO_4$,$CaSO_4$

- Aqueous phase: Present

The system is solved iteratively, using the amount of dissolved $KHSO_4$ as an estimation variable. For each estimation, `FUNCTION FUNCK3` calculates the dissolved $KHSO_4$ and the ionic

equilibria in the aqueous phase. The solubility product of KHSO$_4$ is used as the objective function.

SUBROUTINE CALCK4

This subroutine solves the forward problem of a sulfate super-rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is above the DRH of KHSO$_4$. The following species are possible:

- Solids: CaSO$_4$

- Aqueous phase: Present

Speciation is mainly controlled by the $HSO_{4\,(aq)}^{-} \longleftrightarrow H_{(aq)}^{+} + SO_{4\,(aq)}^{2-}$ equilibrium.

SUBROUTINE CALCS1

This subroutine solves the reverse problem of a sulfate poor, ammonium-sulfate aerosol system, when the relative humidity is below the DRH of (NH$_4$)$_2$SO$_4$. The following species are possible:

- Solids: (NH$_4$)$_2$SO$_4$

- Aqueous phase: Not present

Solid (NH$_4$)$_2$SO$_4$ is calculated from the amount of sulfates available. Gas phase ammonia in this case is undefined (it is a free parameter).

SUBROUTINE CALCS2

This subroutine solves the reverse problem of a sulfate poor, ammonium-sulfate aerosol system, when the relative humidity is above the DRH of (NH$_4$)$_2$SO$_4$. The following species are possible:

- Solids: None

- Aqueous phase: Present

The concentrations in this case are mainly determined by the equilibria: $NH_{3(g)} \xleftrightarrow{K_{21}} NH_{3(aq)}$

and $NH_{3(aq)} + H_2O_{(aq)} \xleftrightarrow{K_{22}} NH_{4(aq)}^+ + OH_{(aq)}^-$.

SUBROUTINE CALCN1

This subroutine solves the reverse problem of a sulfate poor, ammonium-nitrate-sulfate aerosol system, when the relative humidity is below the DRH of NH_4NO_3. The following species are possible:

- Solids: $(NH_4)_2SO_4$, NH_4NO_3

- Aqueous phase: Present when RH ≥ MDRH, not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by `SUBROUTINE CALCN1A`. The composition in the mutual deliquescence region is solved by `SUBROUTINE CALCMDRP`.

SUBROUTINE CALCN1A

This subroutine solves the forward problem of a sulfate poor, ammonium-nitrate-sulfate aerosol system, when the relative humidity is below the DRH of NH_4NO_3. The following species are possible:

- Solids: $(NH_4)_2SO_4$, NH_4NO_3

- Aqueous phase: None

$(NH_4)_2SO_4$ is calculated from the sulfates, while NH_4NO_3 is calculated from the nitrates. Species in the gas phase are determined from the $NH_4NO_{3(s)} \xleftrightarrow{} NH_{3(g)} + HNO_{3(g)}$ equilibrium.

SUBROUTINE CALCN2

This subroutine solves the reverse problem of a sulfate poor, ammonium-nitrate-sulfate aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and $(NH_4)_2SO_4$. The following species are possible:

- Solids: $(NH_4)_2SO_4$

- Aqueous phase: Present

The system is solved iteratively, using the amount of dissolved $(NH_4)_2SO_4$ as the estimation variable. For each estimation, `FUNCTION FUNCN2` calculates the gas phase and ionic species. The solubility product of $(NH_4)_2SO_4$ is used as the objective function.

SUBROUTINE CALCN3

This subroutine solves the reverse problem of a sulfate poor, ammonium-nitrate-sulfate aerosol system, when the relative humidity is above the DRH of $(NH_4)_2SO_4$. The following species are possible:

- Solids: None

- Aqueous phase: Present

The concentrations in this case are determined from the equilibria: $NH_{3(g)} \longleftrightarrow NH_{3(aq)}$
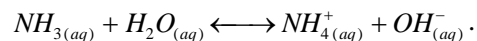
$NH_{3(aq)} + H_2O_{(aq)} \longleftrightarrow NH_{4(aq)}^+ + OH_{(aq)}^-$.

SUBROUTINE CALCQ1

This subroutine solves the reverse problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of NH_4NO_3. The following species are possible:

- Solids: $(NH_4)_2SO_4$, NH_4NO_3, NH_4Cl, Na_2SO_4

- Aqueous phase: Present when RH $\geq$ MDRH, not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by SUBROUTINE CALCQ1A. The composition in the mutual deliquescence region is solved by SUBROUTINE CALCMDRP.

SUBROUTINE CALCQ1A

This subroutine solves the reverse problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the MDRH region of SUBROUTINE CALCQ1. The following species are possible:

- Solids: $(NH_4)_2SO_4$, NH_4NO_3, NH_4Cl, Na_2SO_4

- Aqueous phase: Not present

All of the sodium reacts with sulfates to form Na_2SO_4. Ammonia the reacts with the remaining sulfate, forming $(NH_4)_2SO_4$. The excess NH_3 then reacts with HNO_3 and HCl to form NH_4NO_3, NH_4Cl, until equilibrium is achieved.

SUBROUTINE CALCQ2

This subroutine solves the reverse problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and NH_4Cl. The following species are possible:

- Solids: $(NH_4)_2SO_4$, NH_4Cl, Na_2SO_4

- Aqueous phase: Present when there is NH_4NO_3 (that deliquesces), or when in a MDRH.

If an aqueous phase is present, SUBROUTINE CALCQ2A is called. When in a MDRH, SUBROUTINE CALCMDRP is called. Otherwise, the aerosol is dry, and SUBROUTINE CALCQ1A is used to calculate the composition.

SUBROUTINE CALCQ2A

This subroutine solves the reverse problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and NH_4Cl. An aqueous phase is presumed to exist. The following species are possible:

- Solids: $(NH_4)_2SO_4$, NH_4Cl, Na_2SO_4

- Aqueous phase: Present.

The system is solved iteratively, using gas phase HCl as the estimation variable. For each estimation, FUNCTION FUNCQ2A calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated salts and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $NH_4^-{}_{(aq)}$, $HCl_{(g)}$ and $NH_{3(g)}$ from equilibrium, according to the reaction

$$NH_{3(g)} + HCl_{(g)} \longleftrightarrow NH_{4(aq)}^+ + Cl_{(aq)}^- .$$
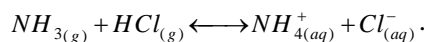
SUBROUTINE CALCQ3

This subroutine solves the forward problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4Cl and $(NH_4)_2SO_4$. The following species are possible:

- Solids: $(NH_4)_2SO_4$, Na_2SO_4

- Aqueous phase: Present when there is some NH_4NO_3 or NH_4Cl (which deliquesces), or when in a MDRH.

If an aqueous phase exists, the problem is solved by SUBROUTINE CALCQ3A. When in a MDRH, SUBROUTINE CALCMDRP is called. Otherwise, the aerosol is dry, and SUBROUTINE CALCQ1A is used to calculate the composition.

SUBROUTINE CALCQ3A

This subroutine solves the reverse problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4Cl and $(NH_4)_2SO_4$. An aqueous phase is presumed to exist. The following species are possible:

- Solids: $(NH_4)_2SO_4$, Na_2SO_4

- Aqueous phase: Present.

The system is solved iteratively, using the amount of gas phase HCl as the estimation variable. For each estimation, FUNCTION FUNCQ3A calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated salts and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $NH_4^-_{(aq)}$, $HCl_{(g)}$ and $NH_{3(g)}$ from equilibrium, according to the reaction $NH_{3(g)} + HCl_{(g)} \longleftrightarrow NH_{4(aq)}^+ + Cl_{(aq)}^-$.

SUBROUTINE CALCQ4

This subroutine solves the reverse problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $(NH_4)_2SO_4$ and Na_2SO_4. The following species are possible:

- Solids: Na_2SO_4

- Aqueous phase: Present

The system is solved iteratively, using the amount of gas phase HCl as the estimation variable. For each estimation, FUNCTION FUNCQ4 calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated Na_2SO_4 and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $NH_4^-{}_{(aq)}$, $HCl_{(g)}$ and $NH_{3(g)}$ from equilibrium, according to the reaction $NH_{3(g)} + HCl_{(g)} \longleftrightarrow NH^+_{4(aq)} + Cl^-_{(aq)}$.

SUBROUTINE CALCQ5

This subroutine solves the reverse problem of a sulfate poor, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is above the DRH of Na_2SO_4. The following species are possible:

- Solids: None

- Aqueous phase: Present

The solution algorithm of SUBROUTINE CALCQ4 is used, with the difference that Na_2SO_4 does not precipitate.

SUBROUTINE CALCR1

This subroutine solves the reverse problem of a sulfate poor, sodium rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of NH_4NO_3. The following species are possible:

- Solids: Na_2SO_4, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present when RH ≥ MDRH, not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by SUBROUTINE CALCR1A. The composition in the mutual deliquescence region is calculated by SUBROUTINE CALCMDRP.

SUBROUTINE CALCR1A

This subroutine solves the reverse problem of a sulfate rich, sodium poor, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the MDRH region of SUBROUTINE CALCR1. The following species are possible:

- Solids: Na_2SO_4, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Not present

All of the sodium reacts with the sulfates to produce Na_2SO_4. The excess sodium reacts to form first $NaNO_3$, then $NaCl$. The remaining HCl and HNO_3 then react with NH_3 to form NH_4NO_3 and NH_4Cl respectively.

SUBROUTINE CALCR2

This subroutine solves the reverse problem of a sulfate poor, sodium rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and $NaNO_3$. The following species are possible:

- Solids: Na_2SO_4, $NaNO_3$, $NaCl$, NH_4Cl

- Aqueous phase: Present when NH_4NO_3 exists or when RH $\geq$ MDRH, not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by SUBROUTINE CALCR1A. The composition in the mutual deliquescence region is solved by SUBROUTINE CALCMDRP.

SUBROUTINE CALCR2A

This subroutine solves the reverse problem of a sulfate poor, sodium rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and $NaNO_3$. An aqueous phase is presumed to exist. The following species are possible:

- Solids: Na_2SO_4, $NaNO_3$, $NaCl$, NH_4Cl

- Aqueous phase: Present.

The system is solved iteratively, using the amount of gas phase HCl as the estimation variable. For each estimation, `FUNCTION FUNCR2A` calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated salts and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $NH_4^-_{(aq)}$, $HCl_{(g)}$ and $NH_{3(g)}$ from equilibrium, according to the reaction $NH_{3(g)} + HCl_{(g)} \longleftrightarrow NH^+_{4(aq)} + Cl^-_{(aq)}$.

SUBROUTINE CALCR3

This subroutine solves the reverse problem of a sulfate poor, sodium rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $NaNO_3$ and NaCl. An aqueous phase is presumed to exist. The following species are possible:

- Solids: Na_2SO_4, NaCl, NH_4Cl
- Aqueous phase: Present.

The solution algorithm of `SUBROUTINE CALCR2` is used, with the difference that $NaNO_3$ does not precipitate.

SUBROUTINE CALCR4

This subroutine solves the reverse problem of a sulfate poor, sodium rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NaCl and NH_4Cl. An aqueous phase is presumed to exist. The following species are possible:

- Solids: Na_2SO_4, NH_4Cl
- Aqueous phase: Present.

The solution algorithm of SUBROUTINE CALCR2 is used, with the difference that $NaNO_3$ and NaCl do not precipitate.

SUBROUTINE CALCR5

This subroutine solves the forward problem of a sulfate poor, sodium rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4Cl and Na_2SO_4. The following species are possible:

- Solids: Na_2SO_4

- Aqueous phase: Present.

The solution algorithm of SUBROUTINE CALCR2 is used, with the difference that $NaNO_3$, NaCl and NH_4Cl do not precipitate.

SUBROUTINE CALR6

This subroutine solves the forward problem of a sulfate poor, sodium rich, sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is above the DRH of Na_2SO_4. The following species are possible:

- Solids: None

- Aqueous phase: Present.

The solution algorithm of SUBROUTINE CALCR2 is used, with the difference that salts do not precipitate.

SUBROUTINE CALCV1

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium poor, calcim-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of NH_4NO_3. The following species are possible:

- Solids: $CaSO_4$, NH_4NO_3, NH4Cl, $(NH_4)_2SO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present when RH ≥ MDRH, not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by `SUBROUTINE CALCV1A`. The composition in the mutual deliquescence region is solved by `SUBROUTINE CALCMDRPII`.

SUBROUTINE CALCV1A

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the MDRH region of SUBROUTINE CALCV1. The following species are possible:

- Solids: $CaSO_4$, NH_4NO_3, NH_4Cl, $(NH_4)_2SO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Not present

All of the calcium, potassium, magnesium and sodium reacts with the sulfates to form $CaSO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4. The remaining sulfates react with ammonia to form $(NH_4)_2SO_4$. The excess NH_3 then reacts with HNO_3 and HCl to form NH_4NO_3, NH_4Cl.

SUBROUTINE CALCV2

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and NH_4Cl. The following species are possible:

- Solids: $CaSO_4$, NH_4Cl, $(NH_4)_2SO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present when there is NH_4NO_3 (that deliquesces) or when in a MDRH.

In the presence of NH_4NO_3, the problem is solved by `SUBROUTINE CALCV2A`. When in a MDRH, `SUBROUTINE CALCMDRPII` is called. Otherwise, the aerosol is dry, and `SUBROUTINE CALCV1A` is used to calculate the composition.

SUBROUTINE CALCV2A

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and NH_4Cl. An aqueous phase is also presumed to exist. The following species are possible:

- Solids: $CaSO_4$, NH_4Cl, $(NH_4)_2SO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present.

The system is solved iteratively, with respect to the amount of gas phase HCl. For each estimation, `FUNCTION FUNCV2A` calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated salts and gas phase NH_3, HNO_3. The objective function used is

the departure of $Cl^-_{(aq)}$, $NH_4^-{}_{(aq)}$, $HCl_{(g)}$ and $NH_{3(g)}$ from the equilibrium state, according to the

reaction $NH_{3(g)} + HCl_{(g)} \longleftrightarrow NH^+_{4(aq)} + Cl^-_{(aq)}$.

SUBROUTINE CALCV3

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium poor, calcium-

potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the

relative humidity is between the DRH of NH_4Cl and $(NH_4)_2SO_4$. The following species are

possible:

- Solids: $CaSO_4$, $(NH_4)_2SO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present when there is some NH_4NO_3 or NH_4Cl (which deliquesces), or when

  in a MDRH.

In the presence of NH_4NO_3 or NH_4Cl, the problem is solved by SUBROUTINE CALCV3A. When

in a MDRH, SUBROUTINE CALCMDRPII is called. Otherwise, the aerosol is dry, and

SUBROUTINE CALCV1A is used to calculate the composition.

SUBROUTINE CALCV3A

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium poor, calcium-

potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the

relative humidity is between the DRH of NH_4Cl and $(NH_4)_2SO_4$. An additional presumption made

is that nitrate and/or chloride exists in the aerosol phase. The following species are possible:

- Solids: $CaSO_4$, $(NH_4)_2SO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present.

The system is solved iteratively, with respect to the amount of gas phase HCl. For each estimation, FUNCTION FUNCV3A calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated salts and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $NH_4^-_{(aq)}$, $HCl_{(g)}$ and $NH_{3(g)}$ from the equilibrium state, according to the reaction $NH_{3(g)} + HCl_{(g)} \longleftrightarrow NH^+_{4(aq)} + Cl^-_{(aq)}$.

SUBROUTINE CALCV4

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $(NH_4)_2SO_4$ and $MgSO_4$. The following species are possible:

- Solids: $CaSO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present

The system is solved iteratively, with respect to the amount of gas phase HCl. For each estimation, FUNCTION FUNCV4 calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated $MgSO_4$, Na_2SO_4, K_2SO_4 and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $HCl_{(g)}$ from equilibrium, according to the reaction $HCl_g \longleftrightarrow H^+_{(aq)} + Cl^+_{(aq)}$.

SUBROUTINE CALCV5

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $MgSO_4$ and Na_2SO_4. The following species are possible:

- Solids: $CaSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present

The iterative solution algorithm of `SUBROUTINE CALCV4` is used, with the difference that $MgSO_4$ does not precipitate.

SUBROUTINE CALCV6

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of Na_2SO_4 and K_2SO_4. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4

- Aqueous phase: Present

The iterative solution algorithm of `SUBROUTINE CALCV5` is used, with the difference that Na_2SO_4 does not precipitate.

SUBROUTINE CALCV7

This subroutine solves the forward problem of a sulfate poor, crustals&sodium poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is above the DRH of K_2SO_4. The following species are possible:

- Solids: $CaSO_4$

- Aqueous phase: Present

The iterative solution algorithm of `SUBROUTINE CALCV6` is used, with the difference that K_2SO_4 does not precipitate.

SUBROUTINE CALCU1

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of NH_4NO_3. The following species are possible:

- Solids: $CaSO_4$, NH_4NO_3, NH_4Cl, $MgSO_4$, Na_2SO_4, K_2SO_4, $NACl$, $NaNO_3$

- Aqueous phase: Present when RH ≥ MDRH, not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by `SUBROUTINE CALCU1A`. The composition in the mutual deliquescence region is solved by `SUBROUTINE CALCMDRPII`.

SUBROUTINE CALCU1A

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when

the relative humidity is below the MDRH regions of SUBROUTINE CALCU1. The following species are possible:

- Solids: $CaSO_4$, NH_4NO_3, NH_4Cl, $MgSO_4$, Na_2SO_4, K_2SO_4, $NACl$, $NaNO_3$

- Aqueous phase: Not present

All of the calcium, potassium and magnesium reacts with the sulfates to form $CaSO_4$, $MgSO_4$, K_2SO_4. The remaining sulfates react with sodium to form Na_2SO_4. The remaining sodium first reacts to form $NaNO_3$, then if possible, NaCl. HCl and HNO_3 then react with NH_3 to form NH_4NO_3 and NH_4Cl.

SUBROUTINE CALCU2

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and $NaNO_3$. The following species are possible:

- Solids: $CaSO_4$, NH_4Cl, $MgSO_4$, Na_2SO_4, K_2SO_4, $NACl$, $NaNO_3$

- Aqueous phase: Present when NH_4NO_3 exists or when RH $\geq$ MDRH, not present when RH $<$ MDRH

The case of a dry aerosol (RH < MDRH) is solved by SUBROUTINE CALCU1A. The composition in the mutual deliquescence region is solved by SUBROUTINE CALCMDRPII.

SUBROUTINE CALCU2A

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and $NaNO_3$. An aqueous phase is also presumed to exist. The following species are possible:

- Solids: $CaSO_4$, NH_4Cl, $MgSO_4$, Na_2SO_4, K_2SO_4, $NaCl$, $NaNO_3$

- Aqueous phase: Present.

The system is solved iteratively, with respect to the amount of gas phase HCl. For each estimation, `FUNCTION FUNCU2A` calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated salts and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $HCl_{(g)}$ from equilibrium, according to the reaction $HCl_g \longleftrightarrow H^+_{(aq)} + Cl^+_{(aq)}$.

SUBROUTINE CALCU3

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $NaNO_3$ and $NaCl$. An aqueous phase is presumed to exist. The following species are possible:

- Solids: $CaSO_4$, NH_4Cl, $MgSO_4$, Na_2SO_4, K_2SO_4, $NaCl$

- Aqueous phase: Present.

The iterative solution algorithm of `SUBROUTINE CALCU2` is used, with the difference that $NaNO_3$ does not precipitate.

SUBROUTINE CALCU4

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NaCl and NH_4Cl. The following species are possible:

- Solids: $CaSO_4$, NH_4Cl, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCU3 is used, with the difference that NaCl does not precipitate.

SUBROUTINE CALCU5

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4Cl and $MgSO_4$. The following species are possible:

- Solids: $CaSO_4$, $MgSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCU4 is used, with the difference that NH_4Cl does not precipitate.

SUBROUTINE CALCU6

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $MgSO_4$ and Na_2SO_4. The following species are possible:

- Solids: $CaSO_4$, Na_2SO_4, K_2SO_4

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCU5 is used, with the difference that $MgSO_4$ does not precipitate.

SUBROUTINE CALCU7

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of Na_2SO_4 and K_2SO_4. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCU6 is used, with the difference that Na_2SO_4 does not precipitate.

SUBROUTINE CALCU8

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is above the DRH of K_2SO_4. The following species are possible:

- Solids: $CaSO_4$

- Aqueous phase: Present.

The iterative solution algorithm of `SUBROUTINE CALCU7` is used, with the difference that K_2SO_4 does not precipitate.

SUBROUTINE CALCW1

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is below the DRH of $CaCl_2$. The following species are possible:

- Solids: $CaSO_4$, $Ca(NO_3)_2$, $CaCl_2$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $MgCl_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present when RH ≥ MDRH, not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by `SUBROUTINE CALCW1A`. The composition in the mutual deliquescence region is solved by `SUBROUTINE CALCMDRPII`.

SUBROUTINE CALCW1A

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when

the relative humidity is below the MDRH regions of `SUBROUTINE CALCP1`. The following species are possible:

- Solids: $CaSO_4$, $Ca(NO_3)_2$, $CaCl_2$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $MgCl_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Not present

Calcium, potassium and magnesium react with the sulfates to form $CaSO_4$, $MgSO_4$, K_2SO_4. The remaining calcium, potassium and magnesium first reacts to form $Ca(NO_3)_2$, $Mg(NO_3)_2$, KNO_3 then if possible, $CaCl_2$, $MgCl_2$, KCl. Na and NH_3 then reacts to form $NaNO_3$, $NaCl$, and NH_4NO_3 and NH_4Cl.

SUBROUTINE CALCW2

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $CaCl_2$ and $MgCl_2$. The following species are possible:

- Solids: $CaSO_4$, $Ca(NO_3)_2$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $MgCl_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present when $CaCl_2$ exists or when RH $\geq$ MDRH, not present when RH < MDRH

The case of a dry aerosol (RH < MDRH) is solved by `SUBROUTINE CALCW1A`. The composition in the mutual deliquescence region is solved by `SUBROUTINE CALCMDRPII`.

SUBROUTINE CALCW2A

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $CaCl_2$ and $MgCl_2$. An aqueous phase is also presumed to exist. The following species are possible:

- Solids: $CaSO_4$, $Ca(NO_3)_2$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $MgCl_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present.

The system is solved iteratively, with respect to the amount of gas phase HCl. For each estimation, `FUNCTION FUNCW2A` calculates the ionic equilibria in the aqueous phase, together with the amount of precipitated salts and gas phase NH_3, HNO_3. The objective function used is the departure of $Cl^-_{(aq)}$, $HCl_{(g)}$ from equilibrium, according to the reaction $HCl_g \longleftrightarrow H^+_{(aq)} + Cl^+_{(aq)}$.

SUBROUTINE CALCW3

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $MgCl_2$ and $Ca(NO_3)_2$. The following species are possible:

- Solids: $CaSO_4$, $Ca(NO_3)_2$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present when nitrates or chlorides exist or when RH ≥ MDRH, not present when RH < MDRH.

The case of a dry aerosol (RH < MDRH) is solved by SUBROUTINE CALCW1A. The composition in the mutual deliquescence region is solved by SUBROUTINE CALCMDRPII.

SUBROUTINE CALCW3A

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $MgCl_2$ and $Ca(NO_3)_2$. An aqueous phase is presumed to exist. The following species are possible:

- Solids: $CaSO_4$, $Ca(NO_3)_2$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCW2A is used, with the difference that $MgCl_2$ does not precipitate.

SUBROUTINE CALCW4

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $Ca(NO_3)_2$ and $Mg(NO_3)_2$. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl

- Aqueous phase: Present when nitrates exist or when RH ≥ MDRH, not present when RH < MDRH.

The case of a dry aerosol (RH < MDRH) is solved by `SUBROUTINE CALCW1A`. The composition in the mutual deliquescence region is solved by `SUBROUTINE CALCMDRPII`.

SUBROUTINE CALCW4A

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals poor, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $Ca(NO_3)_2$ and $Mg(NO_3)_2$. An aqueous phase is presumed to exist. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $Mg(NO_3)_2$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl
- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCW3A is used, with the difference that $Ca(NO_3)_2$ does not precipitate.

SUBROUTINE CALCW5

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $Mg(NO_3)_2$ and NH_4NO_3. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl
- Aqueous phase: Present when nitrates exist or when RH $\geq$ MDRH, not present when RH < MDRH.

The case of a dry aerosol (RH < MDRH) is solved by SUBROUTINE CALCW1A. The composition in the mutual deliquescence region is solved by SUBROUTINE CALCMDRPII.

SUBROUTINE CALCW5A

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $Mg(NO_3)_2$ and NH_4NO_3. An aqueous phase is presumed to exist. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $NaNO_3$, $NaCl$, NH_4NO_3, NH_4Cl
- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCW4A is used, with the difference that $Mg(NO_3)_2$ does not precipitate.

SUBROUTINE CALCW6

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4NO_3 and $NaNO_3$. An aqueous phase is presumed to exist. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$, $NaNO_3$, $NaCl$, NH_4Cl
- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCW5A is used, with the difference that NH_4NO_3 does not precipitate.

SUBROUTINE CALCW7

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $NaNO_3$ and NaCl. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$, NaCl, NH_4Cl

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCW6 is used, with the difference that $NaNO_3$ does not precipitate.

SUBROUTINE CALCW8

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NaCl and NH_4Cl. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$, NH_4Cl

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCW7 is used, with the difference that NaCl does not precipitate.

SUBROUTINE CALCW9

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of NH_4Cl and KCl. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, KCl, $MgSO_4$

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCW8 is used, with the difference that NH_4Cl does not precipitate.

SUBROUTINE CALCW10

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of KCl and $MgSO_4$. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3, $MgSO_4$

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCW9 is used, with the difference that KCl does not precipitate.

SUBROUTINE CALCW11

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of $MgSO_4$ and KNO_3. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4, KNO_3

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCW10 is used, with the difference that $MgSO_4$ does not precipitate.

SUBROUTINE CALCW12

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is between the DRH of KNO_3 and K_2SO_4. The following species are possible:

- Solids: $CaSO_4$, K_2SO_4

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCW11 is used, with the difference that KNO_3 does not precipitate.

SUBROUTINE CALCW13

This subroutine solves the reverse problem of a sulfate poor, crustals&sodium rich, crustals rich, calcium-potassium-magnesium-sodium-ammonium-nitrate-sulfate-chloride aerosol system, when the relative humidity is above the DRH of K_2SO_4. The following species are possible:

- Solids: $CaSO_4$

- Aqueous phase: Present.

The iterative solution algorithm of SUBROUTINE CALCW12 is used, with the difference that K_2SO_4 does not precipitate.

SUBROUTINE CALCMDRH, CALCMDRP

These subroutines are called when the solution is in a mutual deliquescence relative humidity region. The solution is assumed to be the weighted mean of two states; the "dry" solution (SUBROUTINE DRYCASE) and the "saturated liquid" solution (SUBROUTINE LIQCASE). SUBROUTINE CALCMDRH is used for forward, and CALCMDRP for reverse problems.

SUBROUTINE CALCMR

This subroutine calculates the aerosol water content from the ZSR correlation.

SUBROUTINE PUSHERR

This subroutine saves an error message in the error stack.

SUBROUTINE ERRSTAT

This subroutine reports error messages to unit "IO".

SUBROUTINE KMFUL

Calculates binary activity coefficients by Kusik-Meissner method. Performs real-time calculations.

SUBROUTINE KMTAB

Obtains Kusik-Meissner binary activity coefficients from precalculated tables. The computations have been performed for a wide range of ionic activities and temperatures. `SUBROUTINE KMTAB` does not access the activity coefficient tables itself but rather calls the appropriate `SUBROUTINE KMXXX` (XXX being a temperature) to obtain the data.

SUBROUTINE KM198, KM223, KM248, KM273, KM298, KM323

Obtains Kusik-Meissner binary activity coefficients from precalculated tables, for a given temperature (e.g., `KM198` is for 198K and `KM223` is for 223K). The computations have been performed for a wide range of ionic activities, and the results are stored in lookup tables. These tables are initialized by `BLOCK DATA KMCF198, KMCF223, KMCF248, KMCF273, KMCF298, KMCF323`.

6.   Description of Global Variables

This paragraph presents the most important global variables of ISORROPIA. These variables are allocated in the COMMON blocks defined in include file ISRPIA.INC. They are presented according to the sequence in which the blocks are allotted.

PARAMETERs

NCOMP   Number of input concentrations (=8)

NIONS   Number of ionic species in aqueous phase (=10)

NGASAQ   Number of undissociated gases in aqueous phase (=3)

NSLDS   Number of solids in solid phase (=19)

NPAIR   Number of salts dissolved in aqueous phase (=23)

NERRMAX   Size of error stack (=25)

Input Data

W   Total concentrations (gas+aerosol) of precursors, moles m^{-3} air

        W(1):   sodium

        W(2):   sulfate

        W(3):   ammonium

        W(4):   nitrate

        W(5):   chloride

        W(6):   calcium

        W(7):   potassium

        W(8):   magnesium

WAER   Concentrations of precursors in aerosol phase, moles m^{-3} air.
        The structure is identical to array W.

TEMP   Ambient temperature, K

RH   Ambient relative humidity (0-1 scale)

IPROB   Type of problem solved

        0: Forward ; total concentrations in array W are known

        1: Reverse ; aerosol concentrations in array WAER are known

METSTBL   Physical state of aerosol

        0: Soluble salts precipitate when supersaturation occurs.

        1: Soluble salts do not precipitate when supersaturation occurs-

aerosol is in metastable state

Deliquescence Relative Humidities

DRH2SO4   Pure H_2SO_4

DRNH42S4   Pure $(NH_4)_2SO_4$

DRNAHSO4   Pure $NaHSO_4$

DRNACL   Pure $NaCl$

DRNANO3   Pure $NaNO_3$

DRNA2SO4   Pure Na_2SO_4

DRNH4HS4   Pure NH_4HSO_4

DRLC   Pure$(NH_4)_3H(SO_4)_2$

DRNH4NO3   Pure NH_4NO_3

DRNH4CL   Pure NH_4Cl

DRCANO32   Pure $Ca(NO_3)_2$

DRCACL2   Pure $CaCl_2$

DRK2SO4   Pure K_2SO_4

DRKHSO4   Pure $KHSO_4$

DRKNO3   Pure KNO_3

DRKCL   Pure KCl

DRMGSO4   Pure $MgSO_4$

DRMGNO32   Pure $Mg(NO_3)_2$

DRMGCL2   Pure $MgCl_2$

DRMLCAB $(NH_4)_3H(SO_4)_2$ - NH_4HSO_4 mixture

DRMLCAS $(NH_4)_3H(SO_4)_2$ -$(NH_4)_2SO_4$ mixture

DRMASAN $(NH_4)_2SO_4$ - NH_4NO_3 mixture

DRMG1 $(NH_4)_2SO_4$ - NH_4NO_3 - Na_2SO_4 - NH_4Cl mixture

DRMG2 $(NH_4)_2SO_4$ - Na_2SO_4 - NH_4Cl mixture

DRMG3 $(NH_4)_2SO_4$ - Na_2SO_4 mixture

DRMH1 Na_2SO_4 - $NaNO_3$ - $NaCl$ - NH_4NO_3 - NH_4Cl mixture

DRMH2 Na_2SO_4 - $NaNO_3$ - $NaCl$ - NH_4Cl mixture

`DRMI1` $(NH_4)_3H(SO_4)_2$ - NH_4HSO_4 - $NaHSO_4$ - Na_2SO_4 - $(NH_4)_2SO_4$ mixture

`DRMI2` $(NH_4)_3H(SO_4)_2$ -$NaHSO_4$ - Na_2SO_4 - $(NH_4)_2SO_4$ mixture

`DRMI3` $(NH_4)_3H(SO_4)_2$ - Na_2SO_4 - $(NH_4)_2SO_4$ mixture

`DRMQ1` Na_2SO_4 - $(NH_4)_2SO_4$ - NH_4NO_3 mixture

`DRMR1` Na_2SO_4 -$NaNO_3$ - $NaCl$ mixture

`DRMR2` Na_2SO_4 - $NaCl$ mixture

`DRMR3` Na_2SO_4 - $NaNO_3$ mixture

`DRMR4` Na_2SO_4 - $NaCl$ - NH_4Cl mixture

`DRMR5` Na_2SO_4 - NH_4Cl mixture

`DRMR6` Na_2SO_4 - $NaNO_3$ - NH_4Cl mixture

`DRMR7` Na_2SO_4 - $NaNO_3$ - $NaCl$ - NH_4Cl mixture

`DRMR8` Na_2SO_4 - $NaCl$ - NH_4Cl mixture

`DRMR9` Na_2SO_4 - NH_4NO_3 mixture

`DRMR10` Na_2SO_4 - $NaNO_3$ - NH_4NO_3 mixture

`DRMR11` Na_2SO_4 - NH_4NO_3 - $NaCl$ - NH_4Cl mixture

`DRMR12` Na_2SO_4 - NH_4NO_3 - NH_4Cl mixture

`DRMR13` Na_2SO_4 - NH_4NO_3 - $NaNO_3$ - NH_4Cl mixture

`DRMO1` $(NH_4)_2SO_4$ - NH_4NO_3 - NH_4Cl - Na_2SO_4 - K_2SO_4 - $MgSO_4$

`DRMO2` $(NH_4)_2SO4$, NH_4Cl, NA_2SO_4, K_2SO_4, $MGSO_4$

`DRMO3` $(NH_4)_2SO_4$, NA_2SO4, K_2SO4, $MGSO_4$

`DRML1` K_2SO_4 - $MGSO_4$ - $KHSO_4$ - NH_4HSO_4 - $NAHSO_4$ - $(NH_4)_2SO_4$ - NA_2SO_4 - LC

`DRML2` K_2SO_4 - $MGSO_4$ - $KHSO_4$ - $NAHSO_4$ - $(NH_4)_2SO_4$ - NA_2SO_4 - LC

`DRML3` K_2SO_4 - $MGSO_4$ - $KHSO_4$ - $(NH_4)_2SO_4$ - NA_2SO_4 - LC

`DRMM1` K_2SO_4 - NA_2SO_4 - $MGSO_4$ - NH_4CL - $NACL$ - $NANO_3$ - NH_4NO_3

`DRMM2` K_2SO_4 - NA_2SO_4 - $MGSO_4$ - NH_4CL - $NACL$ - $NANO_3$

`DRMP1` $CA(NO_3)_2$ - $CACL_2$ - K_2SO_4 - KNO_3 - KCL - $MGSO_4$ - $MG(NO_3)_2$ - $MGCL_2$ - $NANO_3$ - $NACL$ - NH_4NO_3 - NH_4CL

`DRMP2` $CA(NO_3)_2$ - K_2SO_4 - KNO_3 - KCL - $MGSO_4$ - $MG(NO_3)_2$ - $MGCL_2$ - $NANO_3$ - $NACL$ - NH_4NO3 - NH_4CL

`DRMP3` $CA(NO_3)_2$ - K_2SO_4 - KNO_3 - KCL - $MGSO_4$ - $MG(NO_3)_2$ - $NANO_3$ - $NACL$ - NH_4NO_3 - NH_4CL

`DRMP4`	K_2SO_4 - KNO_3 - KCL - $MGSO_4$ - $MG(NO_3)_2$ - $NANO_3$ - NACL - NH_4NO_3 - NH_4CL
`DRMP5`	K_2SO_4 - KNO_3 - KCL - $MGSO_4$ - $NANO_3$ - NACL - NH_4NO_3 - NH_4CL
`DRMV1`	$(NH_4)_2SO_4$ - NH_4NO_3 - NA_2SO_4 - K_2SO_4 - $MGSO_4$
`WFTYP`	Weighting factor type for Mutual Deliquescence Regions.
	0: Dry aerosol (no MDRH); 1:weighting factor always 0.5; 2: relative-humidity averaged (default)

Liquid Aerosol Phase

`MOLAL`	Concentration of ionic species in aqueous phase - moles m^{-3} air.
	`MOLAL(1):` H^+
	`MOLAL(2):` Na^+
	`MOLAL(3):` NH_4^+
	`MOLAL(4):` Cl^-
	`MOLAL(5):` SO_4^{2-}
	`MOLAL(6):` HSO_4^-
	`MOLAL(7):` NO_3^-
	`MOLAL(8):` Ca^{2+}
	`MOLAL(9):` K^+
	`MOLAL(10):` Mg^{2+}
`GASAQ`	Concentration of undissociated gases in aqueous phase - moles m^{-3} air.
	`GASAQ(1):` NH_3
	`GASAQ(2):` HCl
	`GASAQ(3):` HNO_3
`WATER`	Aerosol water content - kg m^{-3} air
`DRYF`	Contains information about the physical state of the system.
	`.TRUE.:` There is no aqueous phase present
	`.FALSE.:` There is an aqueous phase present
`GAMA`	Activity coefficients for salts dissolved in the aqueous phase.
	`GAMA(01):` NaCl
	`GAMA(02):` Na_2SO_4
	`GAMA(03):` $NaNO_3$
	`GAMA(04):` $(NH_4)_2SO_4$
	`GAMA(05):` NH_4NO_3
	`GAMA(06):` NH_4Cl

GAMA(07): $2H-SO_4$

GAMA(08): $H-HSO_4$

GAMA(09): NH_4HSO_4

GAMA(10): HNO_3

GAMA(11): HCl

GAMA(12): $NaHSO_4$

GAMA(13): $(NH_4)_3H(SO_4)_2$

GAMA(14): $CaSO_4$

GAMA(15): $Ca(NO_3)_2$

GAMA(16): $CaCl_2$

GAMA(17): K_2SO_4

GAMA(18): $KHSO_4$

GAMA(19): KNO_3

GAMA(20): KCl

GAMA(21): $MgSO_4$

GAMA(22): $Mg(NO_3)_2$

GAMA(23): $MgCl_2$

IONIC	Ionic activity of aqueous phase.
EPSACT	Convergence criterion in activity coefficient calculations.
NADJ	Forces the solution obtained to satisfy total mass balance to machine precision. 0: No adjustment is performed 1: Adjust solution
IACALC	Selects how binary activity coefficients are calculated. 0: Full calculations are performed 1: Pre-calculated tables are used
FRST, CALAIN, CALAOU	Logical flags that control the activity coefficient calculations.

Solid Aerosol Phase

CNH42S4	Concentration of solid $(NH_4)_2SO_4$ - moles m^{-3} air.
CNH4HS4	Concentration of solid NH_4HSO_4 - moles m^{-3} air.
CNACL	Concentration of solid $NaCl$ - moles m^{-3} air.
CNA2SO4	Concentration of solid Na_2SO_4 - moles m^{-3} air.
CNANO3	Concentration of solid $NaNO_3$ - moles m^{-3} air.

`CNH4NO3`	Concentration of solid NH_4NO_3 - moles m^{-3} air.
`CNH4CL`	Concentration of solid NH_4Cl - moles m^{-3} air.
`CNAHSO4`	Concentration of solid $NaHSO_4$ - moles m^{-3} air.
`CLC`	Concentration of solid $(NH_4)_3H(SO_4)_2$ - moles m^{-3} air.
`CKCL`	Concentration of solid KCl - moles m^{-3} air.
`CK2SO4`	Concentration of solid K_2SO_4 - moles m^{-3} air.
`CKHSO4`	Concentration of solid $KHSO_4$ - moles m^{-3} air.
`CKNO3`	Concentration of solid KNO_3 - moles m^{-3} air.
`CCACL2`	Concentration of solid $CaCl_2$ - moles m^{-3} air.
`CCASO4`	Concentration of solid $CaSO_4$ - moles m^{-3} air.
`CCANO32`	Concentration of solid $Ca(NO_3)_2$ - moles m^{-3} air.
`CMGCL2`	Concentration of solid $MgCl_2$ - moles m^{-3} air.
`CMGNO32`	Concentration of solid $Mg(NO_3)_2$ - moles m^{-3} air.
`CMGSO4`	Concentration of solid $MgSO_4$ - moles m^{-3} air.

Gas Phase

`GNH3`	Concentration of gas phase NH_3 - moles m^{-3} air.
`GHCl`	Concentration of gas phase HCl - moles m^{-3} air.
`GHNO3`	Concentration of gas phase HNO_3 - moles m^{-3} air.

Equilibrium Constants

`XK1`	$HSO_{4(aq)}^- \xleftrightarrow{K_1} H_{(aq)}^+ + SO_{4(aq)}^{2-}$
`XK2`	$NH_{3(g)} \xleftrightarrow{K_2} NH_{4(aq)}^+ + OH_{(aq)}^-$
`XK21`	$NH_{3(g)} \xleftrightarrow{K_{21}} NH_{3(aq)}$
`XK22`	$NH_{3(aq)} + H_2O_{(aq)} \xleftrightarrow{K_{22}} NH_{4(aq)}^+ + OH_{(aq)}^-$
`XK3`	$HCl_{(g)} \xleftrightarrow{K_3} H_{(aq)}^+ + Cl_{(aq)}^-$
`XK31`	$HCl_{(g)} \xleftrightarrow{K_{31}} HCl_{(aq)}$
`XK32`	$HCl_{(aq)} \xleftrightarrow{K_{32}} H_{(aq)}^+ + Cl_{(aq)}^-$
`XK4`	$HNO_{3(g)} \xleftrightarrow{K_4} H_{(aq)}^+ + NO_{3(aq)}^-$
`XK41`	$HNO_{3(g)} \xleftrightarrow{K_{41}} HNO_{3(aq)}$
`XK42`	$HNO_{3(aq)} \xleftrightarrow{K_{42}} H_{(aq)}^+ + NO_{3(aq)}^-$

XK5 $\qquad Na_2SO_{4(s)} \xleftrightarrow{K_5} 2Na^+_{(aq)} + SO^{2-}_{4(aq)}$

XK6 $\qquad NH_4Cl_{(s)} \xleftrightarrow{K_6} NH_{3(g)} + HCl_{(g)}$

XK7 $\qquad (NH_4)_2 SO_{4(s)} \xleftrightarrow{K_7} 2NH^+_{4(aq)} + SO^{2-}_{4(aq)}$

XK8 $\qquad NaCl_{(s)} \xleftrightarrow{K_8} Na^+_{(aq)} + Cl^-_{(aq)}$

XK9 $\qquad NaNO_{3(s)} \xleftrightarrow{K_9} Na^+_{(aq)} + NO^-_{3(aq)}$

XK10 $\qquad NH_4NO_{3(s)} \xleftrightarrow{K_{10}} NH_{3(g)} + HNO_{3(g)}$

XK11 $\qquad NaHSO_{4(s)} \xleftrightarrow{K_{11}} Na^+_{(aq)} + HSO^-_{4(aq)}$

XK12 $\qquad NH_4HSO_{4(s)} \xleftrightarrow{K_{12}} NH^+_{4(aq)} + HSO^-_{4(aq)}$

XK13 $\qquad (NH_4)_3 H(SO_4)_{2(s)} \xleftrightarrow{K_{13}} 3NH^+_{4(aq)} + HSO^-_{4(aq)} + SO^{2-}_{4(aq)}$

XK14 $\qquad NH_4Cl_{(s)} \xleftrightarrow{K_{14}} NH^+_{4(aq)} + Cl^-_{(aq)}$

XKW $\qquad H_2O_{(aq)} \xleftrightarrow{K_w} H^+_{(aq)} + OH^-_{(aq)}$

XK15 $\qquad Ca(NO_3)_{2(s)} \leftrightarrow Ca^{2+}_{(aq)} + 2NO^-_{3(aq)}$

XK16 $\qquad CaCl_{2(s)} \leftrightarrow Ca^{2+}_{(aq)} + 2Cl^-_{(aq)}$

XK17 $\qquad K_2SO_{4(s)} \leftrightarrow 2K^+_{(aq)} + SO^{2-}_{4(aq)}$

XK18 $\qquad KHSO_{4(s)} \leftrightarrow K^+_{(aq)} + HSO^-_{4(aq)}$

XK19 $\qquad KNO_{3(s)} \leftrightarrow K^+_{(aq)} + NO^-_{3(aq)}$

XK20 $\qquad KCl_{(s)} \leftrightarrow K^+_{(aq)} + Cl^-_{(aq)}$

XK23 $\qquad MgSO_{4(s)} \leftrightarrow Mg^{2+}_{(aq)} + SO^{2-}_{4(aq)}$

XK24 $\qquad Mg(NO_3)_{2(s)} \leftrightarrow Mg^{2+}_{(aq)} + 2NO^-_{3(aq)}$

XK25 $\qquad MgCl_{2(s)} \leftrightarrow Mg^{2+}_{(aq)} + 2Cl^-_{(aq)}$

XK26 $\qquad CaSO_4.2H_2O_{(s)} \leftrightarrow Ca^{2+}_{(aq)} + SO^{2-}_{4\ (aq)} + 2H_2O$

Molecular Weights

WMW $\qquad$ Molecular weights of substances in array W or WAER.

$\qquad$ WMW(1): $\quad$ 23 (sodium)

$\qquad$ WMW(2): $\quad$ 98 (sulfate)

$\qquad$ WMW(3): $\quad$ 17 (ammonia)

$\qquad$ WMW(4): $\quad$ 63 (nitrate)

$\qquad$ WMW(5): $\quad$ 36.5 (chloride)

$\qquad$ WMW(6): $\quad$ 40.1 (calcium)

$\qquad$ WMW(7): $\quad$ 39.1 (potassium)

`WMW(8):`   24.3 (magnesium)

`SMW`   Molecular weights of substances in array `MOLALR`.

     `SMW(01):`   NaCl

     `SMW(02):` Na_2SO_4

     `SMW(03):` $NaNO_3$

     `SMW(04):` $(NH_4)_2SO_4$

     `SMW(05):` NH_4NO_3

     `SMW(06):` NH_4Cl

     `SMW(07):` $2H-SO_4$

     `SMW(08):` $H-HSO_4$

     `SMW(09):` NH_4HSO_4

     `SMW(10):` HNO_3

     `SMW(11):`   HCl

     `SMW(12):` $NaHSO_4$

     `SMW(13):` $(NH_4)_3H(SO_4)_2$

     `SMW(14):` $CaSO_4$

     `SMW(15):` $Ca(NO_3)_2$

     `SMW(16):` $CaCl_2$

     `SMW(17):` K_2SO_4

     `SMW(18):` $KHSO_4$

     `SMW(19):` KNO_3

     `SMW(20):`   KCl

     `SMW(21):` $MgSO_4$

     `SMW(22):` $Mg(NO_3)_2$

     `SMW(23):` $MgCl_2$

`IMW`   Molecular weights of ions in array `MOLAL`.

     `IMW(01):` H^+

     `IMW(02):` Na^+

     `IMW(03):` NH_4^+

     `IMW(04):` Cl^-

     `IMW(05):` SO_4^{-2}

     `IMW(06):` HSO_4^-

     `IMW(07):` NO_3^-

     `IMW(08):` Ca^{2+}

     `IMW(09):` K^+

     `IMW(10):` Mg^{2+}

General Variables

VERSION	Version information of ISORROPIA
GREAT	A very large number
TINY, TINY2	Very small numbers

7. Examples of using ISORROPIA

Interactive demo

This example demonstrates the execution of ISORROPIA when compiled and linked with the interactive driver. For an interactive run to take place, an empty line ("carriage return") is given when the data input file name is requested. The configuration file used is shown in List 4. The run session, as it would take place, is shown in List 5. The driver first requests the input file name. When a carriage return is given, an interactive session is assumed. Concentration units, type of problem solved, precursor concentrations and ambient conditions are then requested. Results obtained from the session of List 5 are shown in List 6.

List 4: The configuration file `ISRPIA.CNF` used.

```
1.  *** [ INPUT/OUTPUT PARAMETERS ] ***
2.  CREATE REPORT IN FILE? (.T.=yes, .F.=no, on screen)
3.  .T.
4.  CREATE SPREADSHEET FILE? (.T.=yes, .F.=no)
5.  .T.
6.
7.  *** [ SOLUTION CONTROL ] ***
8.  Convergence criterion (DEFAULT:1.D-6)
9.  1.D-10
10. Max iterations for bisection method (DEFAULT:100)
11. 100
12. Sweeps for activity coef. calculation (DEFAULT:4)
13. 10
14. Accuracy in activity coef. calculation (DEFAULT:5e-2)
15. 1e-2
16. Subdivisions for root tracking (DEFAULT:5)
17. 5
18. Method for binary activity coefs (0-online, 1-tables) (DEFAULT:1)
19. 0
20. Force mass conservation to machine precision? (0-no, 1-yes) (DEFAULT:0)
21. 0
```

List 5: A sample interactive session with ISORROPIA.

```
1.  -----------------------------------------------
2.  ISORROPIA v2.1 (05/26/09)
3.  Copyright 1996-2009
4.  University of Miami, Carnegie Mellon University,
5.  Gerogia Institute of Technology
6.  Written by Athanasios Nenes (nenes@eas.gatech.edu)
7.  and Christos Fountoukis
8.  (Christos.Fountoukis@chbe.gatech.edu)
9.  -----------------------------------------------
10.
```

```
11. Parameters read from file [isrpia.cnf]
12. File name with runs [Enter=screen input]:
13.
14. Input units (0=umol/m3 air, 1=ug/m3 air) [1.]: 1
15. Problem type? (0=forward, 1=reverse) [0.]: 0
16. Aerosol state? (0=solid+liquid possible, 1=metastable) [0.]: 0
17. Natot  (umol/m3 air)? [1.000]: 1.0
18. SO4tot (umol/m3 air)? [1.000]: 6.0
19. NH3tot (umol/m3 air)? [1.000]: 1.0
20. NO3tot (umol/m3 air)? [1.000]: 2.0
21. Cltot  (umol/m3 air)? [1.000]: 1.5
22. Catot  (umol/m3 air)? [1.000]: 0.5
23. Ktot   (umol/m3 air)? [1.000]: 0.5
24. Mgtot  (umol/m3 air)? [1.000]: 0.1
25. RH (0-1)? [0.900]: 0.9
26. Temperature (K)? [ 298.0]: 298.0
27.
28. Results saved in file [SCREEN.txt]
29. Data    saved in file [SCREEN.dat]
```

List 6: Results of ISORROPIA run.

```
1.  *** [ INPUT ] ***
2.
3.   TOTAL:            (ug/m3 air)    (umol/m3 air)
4.   [Na  ]            1.000E+00       4.348E-02
5.   [H2SO4]           6.000E+00       6.122E-02
6.   [NH3 ]            1.000E+00       5.882E-02
7.   [HNO3 ]           2.000E+00       3.175E-02
8.   [HCL ]            1.500E+00       4.110E-02
9.   [Ca  ]            5.000E-01       1.247E-02
10.  [K   ]            5.000E-01       1.279E-02
11.  [Mg  ]            1.000E-01       4.115E-03
12.
13.  TEMPERATURE (K) [298.00]
14.  REL. HUMIDITY   [ 90.00] %
15.
16.  *** [ SOLUTION ] ***
17.
18.  GENERAL INFO:
19.  PROBLEM TYPE   [FOREWARD]
20.  AEROSOL STATE  [SOLID + LIQUID POSSIBLE ]
21.  ACTIVITY COEFS [KUSSIK-MEISSNER; TABULATED COEFFICIENTS]
22.  MASS CONSERV.  [NORMAL SOLUTION PROCEDURE]
23.  REGIME CASE         [O5]
24.  SO4 RATIO          [ 0.000E+00]
25.  Na  RATIO          [ 0.000E+00]
26.  CRUSTAL RATIO      [ 4.797E-01]
27.  CRUSTAL & Na RATIO [ 1.190E+00]
28.  SO4 RATIO W/CRUSTALS [ 2.151E+00]
29.  SO4 POOR RATIO     [ 2.000E+00]
30.
31.  COMPOSITION:
32.  GAS+AEROSOL:     (ug/m3 air)    (umol/m3 air)
33.  [Na  ]            1.000E+00       4.348E-02
34.  [H2SO4]           6.000E+00       6.122E-02
35.  [NH3 ]            1.000E+00       5.882E-02
36.  [HNO3 ]           2.000E+00       3.175E-02
37.  [HCL ]            1.500E+00       4.110E-02
38.  [Ca  ]            5.000E-01       1.247E-02
39.  [K   ]            5.000E-01       1.279E-02
40.  [Mg  ]            1.000E-01       4.115E-03
41.
42.  GAS:             (ug/m3 air)    (umol/m3 air)    (% mass)    (% mole)
43.  [NH3 ]            3.109E-01       1.829E-02        9.209      22.061
```

```
44. [HNO3 ]              1.680E+00       2.667E-02    49.774      32.175
45. [HCL  ]              1.385E+00       3.793E-02    41.017      45.764
46.
47. AEROSOL MASS:     (ug/m3 air)    (% of total)
48. [TOTAL ]             2.727E+01
49. [LIQUID]             2.557E+01          93.776
50. [SOLID ]             1.697E+00           6.224
51.
52. SOLID AEROSOL:    (ug/m3 air)    (umol/m3 air)  (% mass)    (% mole)
53. [NaNO3     ]         0.000E+00       0.000E+00     0.000       0.000
54. [Na2SO4    ]         1.420E-12       1.000E-14     0.000       0.000
55. [NaHSO4    ]         0.000E+00       0.000E+00     0.000       0.000
56. [NaCL      ]         0.000E+00       0.000E+00     0.000       0.000
57. [NH4CL     ]         0.000E+00       0.000E+00     0.000       0.000
58. [NH4NO3    ]         0.000E+00       0.000E+00     0.000       0.000
59. [(NH4)2SO4]          0.000E+00       0.000E+00     0.000       0.000
60. [NH4HSO4  ]          0.000E+00       0.000E+00     0.000       0.000
61. [LC        ]         0.000E+00       0.000E+00     0.000       0.000
62. [CaSO4     ]         1.697E+00       1.247E-02   100.000     100.000
63. [Ca(NO3)2  ]         0.000E+00       0.000E+00     0.000       0.000
64. [CaCl2     ]         0.000E+00       0.000E+00     0.000       0.000
65. [K2SO4     ]         1.742E-12       1.000E-14     0.000       0.000
66. [KHSO4     ]         0.000E+00       0.000E+00     0.000       0.000
67. [KNO3      ]         0.000E+00       0.000E+00     0.000       0.000
68. [KCl       ]         0.000E+00       0.000E+00     0.000       0.000
69. [MgSO4     ]         0.000E+00       0.000E+00     0.000       0.000
70. [MgNO32    ]         0.000E+00       0.000E+00     0.000       0.000
71. [MgCl2     ]         0.000E+00       0.000E+00     0.000       0.000
72.
73. LIQUID AEROSOL:   (ug/m3 air)    (umol/m3 air)  (% mass)    (% mole)
74. [WATER ]             1.813E+01       1.007E+00    70.910      86.433
75. [H+    ]             1.894E-04       1.894E-04     0.001       0.016
76. [Na+   ]             1.000E+00       4.348E-02     3.911       3.731
77. [NH4+  ]             7.297E-01       4.054E-02     2.854       3.479
78. [Cl-   ]             1.123E-01       3.162E-03     0.439       0.271
79. [NO3-  ]             3.147E-01       5.076E-03     1.231       0.436
80. [SO4-- ]             4.630E+00       4.823E-02    18.108       4.139
81. [HSO4- ]             5.107E-02       5.265E-04     0.200       0.045
82. [Ca    ]             0.000E+00       0.000E+00     0.000       0.000
83. [K     ]             5.000E-01       1.279E-02     1.956       1.097
84. [Mg    ]             1.000E-01       4.115E-03     0.391       0.353
85. [NH3aq ]             0.000E+00       0.000E+00     0.000       0.000
86. [HClaq ]             0.000E+00       0.000E+00     0.000       0.000
87. [HNO3aq]             0.000E+00       0.000E+00     0.000       0.000
88. [pH    ]             1.981E+00
89. [IONIC STRENGTH]     8.718E+00
90.
91. *** [ MASS BALANCE (% ERROR) ] ***
92.
93.  Na  TOTAL  [ 4.600E-11]
94.  SO4 TOTAL  [ 3.268E-11]
95.  NH4 TOTAL  [ 0.000E+00]
96.  NO3 TOTAL  [ 0.000E+00]
97.  CL  TOTAL  [ 0.000E+00]
98.  CA  TOTAL  [ 0.000E+00]
99.  K   TOTAL  [ 1.564E-10]
100. MG  TOTAL  [ 0.000E+00]
101.
102. *** [ CHARGE BALANCE (umole/m3) ] ***
103.
104. TOTAL POSITIVE [ 1.052E-01]
105. TOTAL NEGATIVE [ 1.052E-01]
106.
107. *** [ ERROR MESSAGES ] ***
108. NO ERRORS DETECTED
109.
110. =================================================
111.
```

Batch mode demo

List 7 displays a sample input file, for running ISORROPIA in batch mode. The user specifies the name of the input file during runtime.

List 7: Sample input file for running ISORROPIA in batch mode.

```
1.  Input units (0=umol/m3, 1=ug/m3)
2.  0
3.
4.  Problem type (0=forward, 1=reverse); Phase state (0=solid+liquid, 1=metastable)
5.  0, 0
6.
7.  NH4-SO4 system case
8.  Na  SO4         NH3         NO3   Cl   Ca    K     Mg   RH    TEMP
9.  0   0.3787878  0.75757575  0.0   0.0  0.01  0.01  0.0  0.5   280.0
```

Generating customized output

This section displays an example of generating customized output via SUBROUTINE USROUT. List 9 shows how SUBROUTINE USROUT would look like, if the amount of dissolved undissociated ammonia (% w/w) and the partial pressure of gas phase ammonia (in mm Hg) are calculated. The first section (ICODE=1) opens the output file ('diss.dat') and writes a header. The second section (ICODE=2) calculates dissolved and gas phase ammonia (in appropriate units) and writes it to 'diss.dat'. The third section (ICODE=3), closes 'diss.dat' and displays a message on screen.

List 9: Example of SUBROUTINE USROUT .

```
1.  C========================================================================
2.  C
3.  C *** SUBROUTINE USROUT
4.  C *** THIS SUBROUTINE IS USED TO GENERATE CUSTOMIZED OUTPUT
5.  C
6.  C========================================================================
7.  C
8.        SUBROUTINE USROUT (ICODE)
9.        INCLUDE 'isrpia.inc'
10.       INCLUDE 'main.inc'
11. C
12. C  INITIAL OUTPUT
13. C
14.       IF (ICODE.EQ.1) THEN
15.       OPEN (99, FILE='diss.dat', STATUS='UNKNOWN', RECL=2048)
16.       WRITE(99,*) '"NH3(%ww)" "PNH3(mmHg)"'
17. C
18. C  OUTPUT AFTER EACH RUN
19. C
20.       ELSE IF (ICODE.EQ.2) THEN
21.       GG = 2.0*(MOLAL(6)+MOLAL(5)) + MOLAL(7) + MOLAL(4) + GASAQ(2) +
22.     &     GASAQ(3) - MOLAL(2) - MOLAL(3) - GASAQ(1)
23.       FRNH3 = 0.0
24.       IF (GG.LT.TINY) FRNH3 = GASAQ(1)
25.       WRITE (99,100) FRNH3*17.0/WATER/10.0, GNH3*TEMP*R*760.0
26. 100   FORMAT (1x, 2(1x, 1PE10.3))
27. C
28. C  FINAL OUTPUT
29. C
30.       ELSE IF (ICODE.EQ.3) THEN
31.       CLOSE (99, STATUS='KEEP')
32.       WRITE(*,*) 'Data   saved in file [diss.dat]'
33.       ENDIF
34.       RETURN
35.       END
```

8. EXAMPLES OF PROGRAMMING ISORROPIA

Using the full interface

List 8 displays an example of using the full programming interface of ISORROPIA. First, SUBROUTINE SETPARM is used to specify the convergence criterion (10^{-6}) and the maximum iterations for all iterative procedures (10). The remaining parameters are assigned negative values, so remain unchanged. ISORROPIA is then called (input values are assumed to be assigned elsewhere in the code), and subsequently the error system is accessed. If any errors occurred during the run, all of them are reported (to unit IO, which is assumed to be connected to a file via an OPEN statement). The program terminates if a fatal error has occurred.

NOTE: If a fatal error has occurred, it is only one (fatals interrupt calculation) and it is the last error found in the error stack.

List 8: Example of using the full interface of ISORROPIA.

```
1.  ( . . . )
2.
3.  C
4.  C   SETUP SOLUTION PARAMETERS
5.        CALL SETPARM (-1d0, -1, 1d-6, 10, -1, -1d0, -1, -1) ! Only EPSI, NSWEEP are set
6.  C
7.  C   CALL ISORROPIA
8.        CNTRL(1) = 0              ! 0=FORWARD PROBLEM, 1=REVERSE PROBLEM
9.        CNTRL(2) = 1              ! 0=SOLID+LIQUID AEROSOL, 1=METASTABLE
10.       CALL ISOROPIA (WI, RHI, TEMPI,  CNTRL,
11.     &                W,  GAS, AERLIQ, AERSLD, SCASE, OTHER)
12. C
13. C   ACCESS ERROR SYSTEM
14.       CALL ISERRINF (ERRSTKI, ERRMSGI, NOFER, STKOFL) ! Obtain error stack
15.       WRITE (IO,*) '*** [ ERROR MESSAGES ] ***'       ! Write error messages
16.       IF (NOFER.EQ.0) THEN                            ! No errors
17.         CALL ERRSTAT (IO, 0, ' ')
18.       ELSE
19.         DO 60 I=1,NOFER                               ! Write error message for
20.            WRITE (IO,*) ' '                           ! each error found in the
21.            CALL ERRSTAT (IO, ERRSTKI(I), ERRMSGI(I)) ! error stack
22. 60      CONTINUE
23.       ENDIF
24.       IF(ERRSTKI(NOFER).GT.1000) STOP 'Fatal error'   ! Fatal error, ABORT.
25.
26. ( . . . )
```

Incorporation into an air quality (or global) model

     This section shows how ISORROPIA can be incorporated into an air quality (or global) model. In the example provided (List 10), we assume that the host model has a subroutine especially for interfacing with a thermodynamic model; this interface is `SUBROUTINE ISRPINTR`, and interacts with ISORROPIA in three stages. First, the total concentration of species are obtained from the host model (sodium, sulfate, nitrate, ammonium and chloride) and saved in array `WI`, after converting to mole m^{-3} units. Second, ISORROPIA is called using `WI` and the ambient RH, T as input. Finally, results are passed back to the main program, expressed in appropriate units (here, concentrations in the host model are expressed in μg m^{-3} for aerosol and partial pressure for gas species). This routine does not setup ISORROPIA solution parameters (through `SUBROUTINE SETPARM`), so the default parameters are used. Finally, the error system is not accessed (through `SUBROUTINE ISERRINF` and `SUBROUTINE ERRSTAT`). Both of these tasks, if needed, could be done somewhere else in the host model.

List 10:.A sample interface between ISORROPIA and an air quality model.

```
1.  C=====================================================================
2.  C
3.  C *** SUBROUTINE ISRPINTR
4.  C *** THIS SUBROUTINE IS A SAMPLE INTERFACE BETWEEN AN AIRSHED MODEL
5.  C *** AND ISORROPIA.
6.  C
7.  C=====================================================================
8.  C
9.        SUBROUTINE ISRPINTR (q, c, ps)
10. C
11.       INCLUDE 'airshed.inc'      ! airshed model declarations
12.       DOUBLE PRECISION WI(8), GAS(3), AERLIQ(15), AERSLD(19), CNTRL(2),
13.      &                 WT(8), OTHER(9), RHI,  TEMPI
14.       CHARACTER SCASE*15
15. C
16. C *** CONVERT INPUT CONCENTRATIONS TO moles/m3 *********************
17. C
18.       WI(1) = q(Na )/emw(Kna )*1.D-6
19.       WI(2) = q(SO4)/emw(KSO4)*1.D-6
20.       WI(3) = q(NH4)/emw(KNH4)*1.D-6
21.       WI(4) = q(NO3)/emw(KNO3)*1.D-6
22.       WI(5) = q(CL )/emw(KCL )*1.D-6
23.       WI(6) = q(Ca )/emw(KCa )*1.D-6
24.       WI(7) = q(K  )/emw(KK  )*1.D-6
25.       WI(8) = q(Mg )/emw(KMg )*1.D-6
26. C
```

```
27. C *** CALL ISORROPIA ************************************************
28. C
29.       RHI     = rh            ! Relative humidity provided by airshed model
30.       TEMPI   = temp          ! Temperature (K) provided by airshed model
31.       CNTRL(1) = 0            ! 0=FORWARD PROBLEM, 1=REVERSE PROBLEM
32.       CNTRL(2) = 0            ! 0=SOLID+LIQUID AEROSOL, 1=METASTABLE
33. C
34.       CALL ISOROPIA (WI, RHI, TEMPI,  CNTRL,
35.      &               WT, GAS, AERLIQ, AERSLD, SCASE, OTHER)
36. C
37. C *** CONVERT OUTPUT CONCENTRATIONS TO ug/m3 ************************
38. C
39. C AEROSOL SPECIES
40. C
41.       q(H2O)   = AERLIQ(8)*18.D6              ! H2O
42.       C(H2O)   = q(H2O)                       ! H2O
43.       C(HP)    = AERLIQ(01)          *1.D6    ! H+(aq)
44.       C(NAP)   = AERLIQ(02)*intmw(03)*1.D6    ! Na+(aq)
45.       C(NH4P)  = AERLIQ(03)*intmw(04)*1.D6    ! NH4+(aq)
46.       C(CLM)   = AERLIQ(04)*intmw(05)*1.D6    ! Cl-(aq)
47.       C(SO4M)  = AERLIQ(05)*intmw(06)*1.D6    ! SO4--(aq)
48.       C(HSO4M) = AERLIQ(06)*intmw(07)*1.D6    ! HSO4-(aq)
49.       C(NO3M)  = AERLIQ(07)*intmw(08)*1.D6    ! NO3-(aq)
50.       C(CaM)   = AERLIQ(13)*intmw(09)*1.D6    ! Ca++(aq)
51.       C(KM)    = AERLIQ(14)*intmw(10)*1.D6    ! K+(aq)
52.       C(MgM)   = AERLIQ(15)*intmw(11)*1.D6    ! Mg++(aq)
53.       C(SC)    = AERSLD(03)*intmw(12)*1.D6    ! NaCl(s)
54.       C(SS)    = AERSLD(05)*intmw(13)*1.D6    ! Na2SO4(s)
55.       C(SN)    = AERSLD(01)*intmw(14)*1.D6    ! NaNO3(s)
56.       C(AS)    = AERSLD(06)*intmw(15)*1.D6    ! (NH4)2SO4(s)
57.       C(AN)    = AERSLD(02)*intmw(16)*1.D6    ! NH4NO3(s)
58.       C(AC)    = AERSLD(04)*intmw(17)*1.D6    ! NH4Cl(s)
59.       C(SA)    = 0.0D0                        ! H2SO4(aq)
60.       C(AB)    = AERSLD(08)*intmw(18)*1.D6    ! NH4HSO4(s)
61.       C(SB)    = AERSLD(07)*intmw(19)*1.D6    ! NaHSO4(s)
62.       C(LC)    = AERSLD(09)*intmw(20)*1.D6    ! (NH4)4H(SO4)2(s)
63.       C(CS)    = AERSLD(10)*intmw(21)*1.D6    ! CaSO4(s)
64.       C(CN)    = AERSLD(11)*intmw(22)*1.D6    ! Ca(NO3)2(s)
65.       C(CCL)   = AERSLD(12)*intmw(23)*1.D6    ! CaCl2(s)
66.       C(PS)    = AERSLD(13)*intmw(24)*1.D6    ! K2SO4(s)
67.       C(PB)    = AERSLD(14)*intmw(25)*1.D6    ! KHSO4(s)
68.       C(PN)    = AERSLD(15)*intmw(26)*1.D6    ! KNO3(s)
69.       C(PCl)   = AERSLD(16)*intmw(27)*1.D6    ! KCl(s)
70.       C(MS)    = AERSLD(17)*intmw(28)*1.D6    ! MgSO4(s)
71.       C(MN)    = AERSLD(18)*intmw(29)*1.D6    ! Mg(NO3)2(s)
72.       C(MCl)   = AERSLD(19)*intmw(30)*1.D6    ! MgCl2(s)
73.
74. C
75. C GASEOUS SPECIES
76. C
77.       ps(k,INH3  ) = GNH3*rgas*temp          ! NH3(g)    (Pa)
78.       ps(k,IHNO3 ) = GHNO3*rgas*temp         ! HNO3(g)   (Pa)
79.       ps(k,IH2SO4) = 0.0d0                    ! H2SO4(g)  (Pa)
80.       ps(k,IHCL  ) = GHCL*rgas*temp          ! HCL(g)    (Pa)
81. C
82.       RETURN
83. C
84. C *** END OF SUBROUTINE ISRPINTR **********************************
85. C
86.       END
```

9. References

Nenes A, Pandis SN, Pilinis C (1998a). ISORROPIA: A new thermodynamic equilibrium model for multiphase multicomponent inorganic aerosols, *Aquat.Geoch.*, **4**, 123-152

Nenes A., Pilinis C., and Pandis S.N. (1998b) Continued Development and Testing of a New Thermodynamic Aerosol Module for Urban and Regional Air Quality Models, *Atmos. Env.*, **33**, 1553-1560

Ansari AS, Pandis SN (1999a) An analysis of four models predicting the partitioning of semivolatile inorganic aerosol components, *Aeros.Sci.Tech.*, **31**, 129-153

Ansari AS, Pandis SN (1999b) Prediction of multicomponent inorganic atmospheric aerosol behavior, *Atmos.Env.*, **33**, 745-757

Fountoukis, C., and Nenes, A.: ISORROPIA II: A computationally efficient thermodynamic equilibrium model for K^+-Ca^{2+}-Mg^{2+}-NH_4^+-Na^+-SO_4^{2-}-NO_3^--Cl^--H_2O aerosols, Atmos. Chem. Phys. Discuss., 7, 1893-1939, 2007.

APPENDIX A. Changes between versions of ISORROPIA

Changes between versions 1.7 and 2.1

- Gas/liquid/solid partitioning has been extended to include crustal elements which resulted in 10 more salts in the solid phase and 3 more ions in the aqueous phase.

- The activity coefficient calculation algorithm has been optimized to increase computational speed and avoid numerical errors.

Changes between versions 1.5 and 1.7

- The variable `LN10` in `SUBROUTINE CALCACT` is declared `REAL`.

- A bug in calculation of aqueous ammonium when the sulfate ratio approached 2 in `FUNCTION FUNCA2` was corrected.

- A typo in `SUBROUTINE CALCI2A` was corrected.

- A correction was introduced in `FUNCTION FUNCI3B` and `FUNCI4A` to avoid the rare occurance of negative sulfate concentrations.

- A bug in the calculation of vapor-phase nitric acid concentration for reverse ammonium-sulfate-nitrate-water problems (sulfate super-rich aerosol only) was corrected.

- `SUBROUTINE ADJUST` is added to provide the user with the option to "force" ISORROPIA to conserve mass up to machine precision.

- The variable `NADJ` is added to the `COMMON` block to control whether `SUBROUTINE` `ADJUST` is to be called after every run or not.

- Inaccuracy in the polynomial fit used to produce the water activity database would sometimes give too high water content for relative humidity near saturation (RH>99.4%). This problem has been corrected.

- `SUBROUTINE CALCMR: CASE F` and `CASE C, MOLALR(9)` is used instead of `MOLALR(4)` for computation of liquid water.

- `CASE J` and `SUBROUTINE FUNCG4A`, computation of solid Na_2SO_4 has been corrected..

- `SUBROUTINE KMFUL`, unnecessary computations of the temperature dependence of binary activity coefficients are now avoided.

- The computational efficiency of some "forward" problem routines is improved.

- `SUBROUTINE FUNCI2A, GAMA(12)` is used instead of `GAMA(9)` for the "A3" constant calculation.

- `BLOCK DATA KMCFxxxA, KMCFxxxB, and KMCFxxxC` (where xxx are one of 198, 223, 248, 273, 298, 323) are now merged into one `BLOCK DATA` subprogram, to avoid compatibility issues with some `FORTRAN` compilers. The tabulated Kusik-Meissner binary activity coefficients have been recomputed by calling `KMFUL` for the midpoint of each ionic strength interval. The code that computes ipos (position in Kusik-Meissner binary activity coefficient lookup tables, as defined in the above block data sections) has been fixed in subroutines `KM198, KM223, KM248, KM273, KM298, and KM323`.

- `SUBROUTINE CACLACT`, upper limit in ionic strength is allowed to be 100 (as opposed to 20 in the older versions). The upper limit is implemented to avoid numerical exception errors.

- In `main.for`, a bug in the `CALL SETPARM` was fixed ("-1d0" in the argument list was changed to the appropriate "-1").

- The water activity database was updated, using the output from the AIM model (http://www.hpc1.uea.ac.uk/~e770/aim.html)

- The DRH of NH4NO3 is strongly dependent on temperature. Under low temperature conditions (T<270K), this changes the order (starting from low to high RH) with which salts deliquesce. For these cases the DRH of NH4NO3 is assumed to not "cross over" the DRH of the other salts present in the solution, especially since thermodynamic data for supercooled NH4NO3 solutions are not known. The same is assumed for NH4Cl and NaNO3 which exhibit similar behavior with NH4NO3.

- Temperature corrections for the water activities have been turned off.

- In `SUBROUTINE INIT1` and `INIT2` unnecessary computations of m_{oi} for salts not present in the solution are now avoided.

Changes between versions 1.4 and 1.5

- When the aerosol water content was very low, `SUBROUTINE CALCD2, CALCD3` and `CALCHS4` would occasionally generate numerical exception errors. These problems have been fixed.

- `SUBROUTINE CALCH1, CALCH2, CALCH3, CALCH4, CALCH5, CALCH6, CALCG3, CALCG4` and `CALCG5` have been modified so that NH_4^+ and H^+ concentrations are more accurately determined.

- `SUBROUTINE PUSHEND` (interactive driver routine) would crash on some platforms whenever an end-of-file (EOF) was encountered. This problem is now fixed.

- Typographical errors in the manual were corrected.

Changes between versions 1.3 and 1.4

- Temperature dependency of the activity coefficients is included. This has been done for both pre-calculated tables and online calculation of activity coefficients.

- The temperature dependence of the activity coefficients is used to correct the water uptake data for temperatures other than 298.15K.

- The MDRH points for all the systems considered in ISORROPIA are updated and complete. For this purpose, the GFEMN model of Ansari and Pandis (1999b) was used to calculate the necessary data.

- The computational efficiency of the activity coefficient routines and of some "forward" problem routines is improved.

- The interactive driver has changed. The input file name is now specified at runtime, and not specified in the configuration file `ISRPIA.CNF` (as done in previous versions). Furthermore, the name of the input file is used to generate the names of the output files.

- ISORROPIA v1.1 through 1.3 was referred to occasionally as ISORROPIA-plus (The "plus" was added initially to distinguish the code from ISORROPIA version 1.0). To avoid confusion, the "plus" suffix is ommited from all versions of ISORROPIA (starting from 1.4).

Changes between versions 1.1 and 1.3

- An error in the code that decreased the amount of aerosol nitrate and chloride (when the sulfate ratio is exactly 2.0) is corrected.

- The water activity data is updated with the most recent datasets available. As a consequence, the water uptake calculated by previous versions of the code is slightly different.

- Errors in the filesystem of the interactive driver are corrected.

- In "reverse problems", the MDRH weighting algorithm was occasionally not weighing the dry solution correctly. This problem is now corrected.

Changes between versions 1.0 and 1.1

- The aerosol can either be in a thermodynamically stable or a metastable state. In the first case, salts precipitate when supersaturation occurs, while in the second, the aerosol is always an aqueous solution (with no precipitate).

- A new solution algorithm was developed for ammonia-sulfate systems that accelerates the solution process.

- A completely new programming interface was developed. Apart from running the model itself, there are now a set of utility subroutines with which one could configure the model, obtain information and access the error system.

- A new error system was developed. There is now an error stack that stores all the errors that appear during a run. The stack is checked for overflow, and there is a routine that prints error messages to a user-defined unit. This error system could be accessed any time after a run has been completed, at any point of a program.

- Backward compatibility with previous versions of ISORROPIA is ensured. Call interfaces from older versions are supported, thus allowing the modeler to use newer versions of the code, without any additional programming effort.

APPENDIX B. Older interfaces of ISORROPIA

ISORROPIA supports calling interfaces of older versions for the sake of backward compatibility. This allows a modeler to use newer versions of the code, without any additional programming effort. The interfaces of previous versions are described below.

Interface for versions 1.1 to 1.7

These versions use the same caller routine as 2.1 (SUBROUTINE ISOROPIA). Because of this, host models setup with these previous versions of ISORROPIA need to change the size of the I/O arrays to accommodate the additional species handled (see section 3.1).

Interface for version 1.0

This version of the model is accessed through SUBROUTINE ISOPLUS (found in file ISOCOM.FOR). Metastable aerosol calculations (first introduced in version 1.1) cannot be performed using this interface.

Calling Syntax

```
CALL ISOPLUS (WI, RHI, TEMPI, IPROBI, GAS, AERLIQ, AERSLD, DRYI)
```

Input Variables

- **WI**, DOUBLE PRECISION array of length [8]. Precursor concentrations, expressed in moles m^{-3} air. Depending on the type of problem solved, WI contains either *gas+aerosol* or *aerosol only* concentrations. Each array element contains:

`WI(1)`	1.	sodium
`WI(2)`	2.	sulfate
`WI(3)`	3.	ammonium
`WI(4)`	4.	nitrate
`WI(5)`	5.	chloride
`WI(6)`	6.	calcium
`WI(7)`	7.	potassium
`WI(8)`	8.	magnesium

- **RHI**, DOUBLE PRECISION variable. Ambient relative humidity expressed on a (0,1) scale.

- **TEMPI**, DOUBLE PRECISION variable. Ambient temperature expressed in Kelvins.

- **IPROB**, INTEGER variable. The type of problem solved:

`IPROB = 0`	Forward problem is solved. In this case, array `WI` contains GAS and AEROSOL concentrations together.
`IPROB = 1`	Reverse problem is solved. In this case, array `WI` contains AEROSOL concentrations only.

Output Variables

- **GAS**, DOUBLE PRECISION array of length [03]. Gaseous species concentrations, expressed in moles m^{-3} air. Each array element contains:

`GAS(1)`	NH_3
`GAS(2)`	HNO_3
`GAS(3)`	HCl

- **AERLIQ**, DOUBLE PRECISION array of length [15]. Liquid aerosol species concentrations, expressed in moles m^{-3} air. Each array element contains:

`AERLIQ(01)`	$H^+_{(aq)}$
`AERLIQ(02)`	$Na^+_{(aq)}$
`AERLIQ(03)`	$NH_4^+{}_{(aq)}$
`AERLIQ(04)`	$Cl^-_{(aq)}$
`AERLIQ(05)`	$SO_4^{-2}{}_{(aq)}$
`AERLIQ(06)`	$HSO_4^-{}_{(aq)}$
`AERLIQ(07)`	$NO_3^-{}_{(aq)}$
`AERLIQ(08)`	H_2O
`AERLIQ(09)`	$NH_{3(aq)}$ (undissociated)
`AERLIQ(10)`	$HCl_{(aq)}$ (undissociated)
`AERLIQ(11)`	$HNO_{3(aq)}$ (undissociated)
`AERLIQ(12)`	OH-(aq)
`AERLIQ(13)`	$Ca^{2+}_{(aq)}$
`AERLIQ(14)`	$K^+_{(aq)}$
`AERLIQ(15)`	$Mg^{2+}_{(aq)}$

- **AERSLD**, DOUBLE PRECISION array of length [19]. Solid aerosol species concentrations, expressed in moles m^{-3} air. Each array element contains:

`AERSLD(01)`	$NaNO_{3(s)}$
`AERSLD(02)`	$NH_4NO_{3(s)}$
`AERSLD(03)`	$NaCl_{(s)}$
`AERSLD(04)`	$NH_4Cl_{(s)}$
`AERSLD(05)`	$Na_2SO_{4(s)}$
`AERSLD(06)`	$(NH_4)_2SO_{4(s)}$
`AERSLD(07)`	$NaHSO_{4(s)}$
`AERSLD(08)`	$NH_4HSO_{4(s)}$
`AERSLD(09)`	$(NH_4)_4H(SO_4)_{2(s)}$
`AERSLD(10)`	$CaSO_{4(s)}$
`AERSLD(11)`	$Ca(NO_3)_{2(s)}$
`AERSLD(12)`	$CaCl_{2(s)}$
`AERSLD(13)`	$K_2SO_{4(s)}$
`AERSLD(14)`	$KHSO_{4(s)}$
`AERSLD(15)`	$KNO_{3(s)}$
`AERSLD(16)`	$KCl_{(s)}$
`AERSLD(17)`	$MgSO_{4(s)}$
`AERSLD(18)`	$Mg(NO_3)_{2(s)}$
`AERSLD(19)`	$MgCl_{2(s)}$

- **DRY**, LOGICAL variable. Contains information about the physical state of the system.

`.TRUE.`	There is no aqueous phase present
`.FALSE.`	There is an aqueous phase present.

APPENDIX C. Compiling ISORROPIA

Compiling with interactive driver

Compiling and linking ISORROPIA (files `ISOFWD.FOR`, `ISOREV.FOR`, `ISOCOM.FOR`) depends on the platform and compiler system used. A typical UNIX f77 compilation command would look like:

```
f77 main.for isocom.for isofwd.for isorev.for -o isorropia
```

The result of this command would be the generation of an executable, "`isorropia`". If the UNIX system does not support FORTRAN source files with a `.FOR` extension (e.g.., IBM AIX xlf compilers), the user must then rename all the `.FOR` files to corresponding `.F` files (i.e., `main.for` should be renamed to `main.f`, etc.).

For Intel-based PC's under MS-DOS and using the Lahey Fortran 77 compiler (version 5.x), the command sequence should be:

```
f77l3 main.for
f77l3 isocom.for
f77l3 isofwd.for
f77l3 isorev.for
386link main.obj isocom.obj isofwd.obj isorev.obj -EXE isoropia.exe
```

the first four commands compile the `.FOR` files and generate the compiled object files. The fifth command links the object files together and generates the executable '`ISOROPIA.EXE`'. Other compilers usually require a similar compile/link procedure.

Compiling with an air quality (or global) model

In this case, the host model has to be compiled and linked only with the ISORROPIA model files (`ISOFWD.FOR`, `ISOREV.FOR`, `ISOCOM.FOR`). A procedure similar to the previous is followed, but for each of the host model and ISORROPIA files.