Manchester SAFT code

Version 2.7

July 2018

School of Chemical Engineering and Analytical Science, The Mill, Sackville St. Manchester

A.J. Masters ([andrew.masters@manchester.ac.uk](mailto:andrew.masters@manchester.ac.uk))

S.J. Halstead ([simon.halstead@gmail.com](mailto:simon.halstead@gmail.com))

**Key reference publications:**

T. Lafitte, A. Apostolakou, C. Avendano, A. Galindo, C.S. Adjiman, E.A. Muller, G. Jackson “Accurate statistical associating fluid theory for chain molecules formed from Mie segments”, J. Chem. Phys. 139 (2013), 154504

V. Papaioannou, T. Lafitte, C. Avendano , C.S. Adjiman, G. Jackson, E.A. Muller, A. Galindo “Group contribution methodology based on the statistical associating fluid theory for heteronuclearmoleculas formed from Mie segments”, J. Chem. Phys. 140 (2014), 054107

S. Dufal, T. Lafitte, A.J. Haslam, A. Galindo, G.N.I. Clark, C. Vega, G. Jackson “The A in SAFT: developing the contribution of association to the Helmholtz free energy within a Wertheim TPT1 treatment of generic Mie fluids”, Mol. Phys. 113 (2015), 948-984

D.K. Eriksen, G. Lazarou, A. Galindo, G. Jackson, C.S. Adjiman, A.J. Haslam “Development of intermolecular potential models for electrolyte solutions using an electrolyte SAFT-VR Mie equation of state”, Mol. Phys. 114 (2016), 2724-2749

**Compilation**

This code has been written to follow FORTRAN standards.

MANSAFT can be compile using the included Makefile:

# make

This will create two executable files:

1. mansaft.exe

This is the executable for running standard property calculations with known parameters.

1. optimiser.exe

This is the executable for using existing data sets to parameterise new Mie beads.

Note:

To compile on CSF you will need to load the module:

# module load compilers/gcc/4.9.0

**Input**

The input format is as follows:

Title – first line is user chosen title which will also appear in output

**BLANK LINE**

nstypes – an integer for the number of different types of Mie bead

Next nstypes lines contain definitions of the different beads, 1 line per bead type, organised as follows:

Bead name σ(Å) ε(K) λʳ λᵃ Sf k\* mass(amu) q σBorn(Å) Nassoc

Next Nassoc lines specify number of each type of association, one integer per line

**BLANK LINE**

N unlike εij values not calculated from mixing rules

Next N lines specify correct εij values as:

i j ε(K)

**BLANK LINE**

N unlike λʳ ij values not calculated from mixing rules

Next N lines specify correct λʳ ij values as:

i j λʳ

**BLANK LINE**

N associations definitions (all others possible interactions are set to zero)

Next N lines specify association interactions as:

bead1 association\_type1 bead2 association\_type2 EHB(K) KHB(Å3)

* here association\_type refers to the order given following the bead definition

**BLANK LINE**

N components (molecular types)

Next N lines define the components of the system as:

Component\_name xi mass(amu) dv(m3 mol⁻1) dt (K) N\_unique\_beads

* here dv and dt are only required for the solvent of an electrolyte solution. They may be set to 0.0 for solutes and no-electrolyte systems.

Following N\_unique\_beads lines show the composition of the component as:

bead\_index (integer) N\_beads

**BLANK LINE**

Calculation\_keyword N\_calculations

**Optimisation input**

To run a parameter optimisation, you will need an additional file listing the parameters to be optimised and the min / max values allowed. The format is:

n\_param – number of parameters to be optimised

Next n\_param lines are:

param\_key bead\_indexes association\_site\_indexes min\_value max\_value

Parameter key:

|  |  |  |  |
| --- | --- | --- | --- |
| Index | Parameter | Bead indexes | Association site indexes |
| 1 |  ii | i | - |
| 2 |  ii | i | - |
| 3 | r ii | i | - |
| 4 | a ii | i | - |
| 5 | Sf | i | - |
| 6 | Nseg | i | - |
| 7 |  ij | i, j | - |
| 8 | r ij | i, j | - |
| 9 | EHB | i, j | k, l |
| 10 | KHB | i, j | k, l |

**Calculation keywords**

|  |  |  |
| --- | --- | --- |
| Keyword | Explanation | Input |
| P | Pressure calculation | T(K), V(m3 mol⁻1) |
| PX | Pressure calculation for varying compositions | T(K), V(m3 mol⁻1), N xi values |
| V | Volume calculation | T(K), P(Pa) |
| VX | Volume calculation for varying compositions | T(K), P(Pa), All xi values |
| PHA1 | Pure VL equilibrium | T(K) |
| VLE | Binary VL equilibrium | T(K), P(Pa) |
| ACT | Electrolyte activity coefficient | T(L), P(Pa)  Cation\_integer, cation\_stoichiometry  Anion\_integer, anion\_stoichiometry  All xi values – N lines for varying compositions |
| CP | Cp heat capacity | Ideal gas Cp parameters A B C D E  N lines:  All\_xi\_values, T(K), P(Pa), Phase(l, v, u - unknown) |
| THERM | Calculate various thermodynamics papers | Ideal gas Cp parameters A B C D E  N lines:  All\_xi\_values, T(K), P(Pa), Phase(l, v, u - unknown) |
| MU | Excess chemical potential | All\_xi\_values, V(m3 mol⁻1), T(K) |
|  |  |  |
| OPT | Bead optimiser | Check input\_mod input\_opt\_mod for current details |
| EOPT | Electrolyte bead optimiser | Check input\_mod input\_opt\_mod for current details |
| ILOPT | IL bead optimiser | Check input\_mod input\_opt\_mod for current details |

**Running**

./mansaft.exe input.in

./optimiser.exe input.in opt\_param.in

Output will be printed to screen

**Program overview**

Main

* Main driver program
  + Calls input reader
  + Performs requested calculations

Types\_mod

* Defines globally used user-defined types:
  + SEGMENT
    - Contains data about Mie segments / beads
  + COMPONENT
    - Contains data about components
  + SYSTEM
    - Contains data about the system and calculation type

Global\_mod

* Contains all constants and variables used globally in the program

Ideal\_mod

* Calculates ideal term, and N / V derivatives

Mono\_mod

* Calculates mono term, and N / V derivatives
  + This can be divided in to hard sphere and dispersion contributions

Chain\_mod

* Calculates chain term, and N / V derivatives

Assoc\_mod

* Calculates association term, and N / V derivatives

Ion\_mod

* Calculates ion and Born term, and N / V derivatives
* Calculates dielectric

Diffs\_mod

* This module setups many of the common differentials required for the N / V derivatives (i.e. chemical potential and pressure)

GL\_mod

* Module for Gaussian Legendre abscissas and weights
* N\_GL points defined in GLOBAL

Setup\_mod

* Uses system composition data, temperature, mole fractions and system molar volume to calculate a variety of common terms used throughout SAFT

Press\_mod

* Calculates pressure analytically

Input\_mod

* Reads input file and sets up Types\_mod arrays

Input\_opt\_mod

* Reads additional data specifying which bead / parameters to optimise

Pure\_phase\_mod

* VLE of a single component system

Mu\_mod

* Calculates chemical potential

Act\_ion\_mod

* Calculates the activity coefficient of an ion in solution
* \*\*Somewhat of a work in progress\*\*

Phase\_simplex\_mod

* Calculates VLE of a binary system via a simplex optimisation
* At present this is quite a simplistic method!

Volume\_mod

* Calculates volume
* Vol\_dens\_g() is the standard routine, or Vol\_dens\_g2() if approximate volume range is known

Therm\_mod

* Calculates a range of thermodynamic properties
* \*\*Somewhat of a work in progress\*\*

Optimiser

* Main control program for parameter optimisation

Simplex\_mod

* Simplex program for optimiser

Error\_mod

* Calculates the error for the simplex optimiser
* \*\*This will likely need editing depending on the bead needing optimisation and the reference data!\*\*