Software Requirements Specification for Mass-Mass Stoichiometry Problem

Deemah Alomair

October 7, 2019

Contents

Revision History

Date	Version	Notes
Date 1	1.0	First virsion of document

1 Reference Material

This section records information for easy reference.

1.1 Table of Units

Throughout this document SI (Système International d'Unités) is employed as the unit system. In addition to the basic units, several derived units are used as described below. For each unit, the symbol is given followed by a description of the unit and the SI name.

symbol	**unit**	**SI**
mol	amount of substance	mole
g	mass	gram
g/mol	molecular weight	gram/mole
mol/mol	mole ratio	mole/mole

Table 1: Table of Units.

1.2 Table of Symbols

The table that follows summarizes the symbols used in this document along with their units. The choice of symbols was made to be consistent with the stoichiometry literature and with existing documentation for stoichiometry mass-mass program. The symbols are listed in alphabetical order.

symbol	**unit**	**description**
m	g	mass.
mol	mol	mole amount of substance.
mw	g/mol	molecular weight.
$R1$	–	reactant with known mass.
$R2$	–	reactant with unknown mass.

1.3 Abbreviations and Acronyms

symbol	description
A	Assumption
DD	Data Definition
GD	General Definition
GS	Goal Statement
IM	Instance Model
LC	Likely Change
PS	Physical System Description
R	Requirement
SRS	Software Requirements Specification
SMMP	Stoichiometry Mass-Mass Program
T	Theoretical Model

Table 2: Abbreviations and Acronyms

2 Introduction

2.1 Purpose of Document

The purpose of this document is to describe the requirements for Stoichiometry Mass-Mass Program. A software product that will produce a mass for unknwon reactant in chemical reaction. The goal statements and theoretical models used in the SMMP code are provided, with an emphasis on explicitly identifying assumptions and unambiguous definitions. This document is intended to be used as a reference to provide ad hoc access to all information necessary to understand and verify the model. The SRS is abstract because the contents say what problem is being solved, but do not say how to solve it.
This document will be used as a starting point for subsequent development phases, including writing the design specification and the software verification and validation plan. The design document will show how the requirements are to be realized, including decisions on the numerical algorithms and programming environment. The verification and validation plan will show the steps that will be used to increase confidence in the software documentation and the implementation. Although the SRS fits in a series of documents that follow the so-called waterfall model, the actual development process is not constrained in any way. Even when the waterfall model is not followed, as Parnas and Clements point out [1], the most logical way to present the documentation is still to "fake" a rational design process.

2.2 Scope of Requirements

The scope of SMMP is limited to get the mass of unknown reactant in chemical reaction.

2.3 Characteristics of Intended Reader

Reviewers of this documentation should have a background on Chemistry. Especially on chemical reactions and stoichiometry.

2.4 Organization of Document

The organization of this document follows the template for an SRS for scientific computing software proposed by [dParnas1972] and [parnasClements1984]. The presentation follows the standard pattern of presenting goals, theories, definitions, and assumptions. For readers that would like a more bottom up approach, they can start reading the instance models in Section: Instance Models and trace back to find any additional information they require.
The goal statements (Section: Goal Statements) are refined to the theoretical models and the theoretical models (Section: Theoretical Models) to the instance models (Section: Instance Models).

3 General System Description

This section provides general information about the system. It identifies the interfaces between the system and its environment, describes the user characteristics and lists the system constraints.

3.1 System Context

Figure 1 shows the system context. A circle represents an external entity outside the software, the user in this case. A rectangle represents the software system itself (SMMP). Arrows are used to show the data flow between the system and its environment. SMMP is mostly self-contained. The only external interaction is through the user interface. The responsibilities of the user and the system are as follows:

Figure 1: System Context

- User Responsibilities:

  - Ensure the inputs are correct.
  - Ensure that all chemical reaction's elements entered are belong to chemical substances group.
  - Take care that consistent units are used for input variables.

- SMMP Responsibilities:
  - Detect data type mismatch, such as a string of characters instead of a floating point number.
  - Determine if the inputs satisfy the required physical and software constraints.
  - Do the necessary calculations to get the final result.
  - Display the result to the end user.

3.2 User Characteristics

The end user of SMMP should have some basic knowledge on chemical stoichiometry.

3.3 System Constraints

There are no system constraints.

4 Specific System Description

This section first presents the problem description, which gives a high-level view of the problem to be solved. This is followed by the solution characteristics specification, which presents the assumptions, theories, definitions and finally the instance models.

4.1 Problem Description

SMMP is intended to solve Mass-Mass Stoichiometry problem in any chemical reaction.

4.1.1 Terminology and Definitions

This subsection provides a list of terms that are used in the subsequent sections and their meaning, with the purpose of reducing ambiguity and making it easier to correctly understand the requirements:

- Stoichiometry: Is the relationship between reactants and products in chemical reactions.
- Chemical reaction: Is a process that leads to the chemical transformation of one set of chemical substances to another.

- Reactant: A substance that takes part in and undergoes change during a reaction.
- Product: Is the substance formed from chemical reactions.
- Coefficients: Is the number of molecules (or atoms) involved in the reaction.
- Mole ration: Is the ration between the amounts in modles of any two compounds involved in a chemical reaction.
- Molecular weight (Molar mass) : A measure of the sum of the atomic weight values of the atoms in a molecule.
- Molecule: A chemical reactant or product consist of chemical element with more than one atom.
- Compound: Molecule made up of two or more elements.

4.1.2 Physical System Description

The physical system of SMMP, as shown in Figure 2, includes the following elements:

PS1: Reactants mass.

PS2: Reactants mole ratio.

PS3: Reactants mole amount.

PS4: Reactants Molecular weight (molar mass).

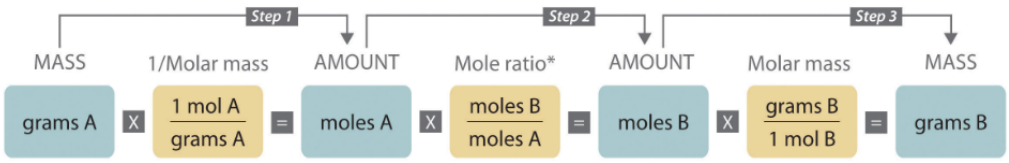

Figure 2: Physical representation of SMMP.

4.1.3 Goal Statements

Given unbalanced chemical reaction with sets of reactants and products , the goal statement are:

GS1: Balance the chemical reaction.

GS2: Get the mass of one of unknown reactant.

4.2 Solution Characteristics Specification

The instance models that govern SMMP are presented in Subsection 4.2.6. The information to understand the meaning of the instance models and their derivation is also presented, so that the instance models can be verified.

4.2.1 Assumptions

This section simplifies the original problem and helps in developing the theoretical model by filling in the missing information for the physical system. The numbers given in the square brackets refer to the type definition[TD], theoretical model [T], general definition [GD], data definition [DD], instance model [IM], or likely change [LC], in which the respective assumption is used.

A1: Chemical reaction with two chemical reactants.

A2: Chemical reactant could be single element, molecule or Compound with two elements.

A3: Chemical product could be single element, molecule or compound with two elements.

A4: Known mass of one chemical reactant.

A5: Known mass entered in g unit.

A6: Known molecular weight for both chemical reactants.

A7: Known molecular weight entered in g/mole unit.

4.2.2 Type Definition

This section focuses on the general variable's types that SMMP is based on.

TD1: ElementT = $\forall$ e $\in$ set of chemical element.

TD2: MoleculeT = tuple of (number : $\mathbb{N}$, element : ElementT).

TD3: CompoundT = tuple of (element1, element2 : MoleculeT).

TD4: ChemicalEqT = tuple of (coefficient1, coefficient2 : $\mathbb{N}$, Compound1 , Compound2 : CompoundT).

4.2.3 Theoretical Models

This section focuses on the general equations and laws that SMMP is based on.

Number	T1
Label	**Law of Conservation of Mass**
Equation	$\sum$ mass for R = $\sum$ mass for P.
Description	The above equation states that there is no change in total mass occurs in a chemical reaction. The total number of products mass is equal to the total number of reactants mass.
Source	[2]
Ref. By	DD5 , IM1

4.2.4 General Definitions

N.A

4.2.5 Data Definitions

This section collects and defines all the data needed to build the instance models. The dimension of each quantity is also given.

Number	DD1
Label	**Count number of atoms in molecule.**
Symbol	-
SI Units	-
Equation	molecule_atoms_count (m) = m.number.
Description	molecule_atoms_count (m) is a function that takes the molecule and return number of atoms in a molecule used in chemical reaction. Where m is a molecule. m.number is molecule's atoms number .
Sources	[3]
Ref. By	IM1

Number	DD2
Label	**Count number of atoms in compound**
Symbol	-
SI Units	-
Equation	compound_atoms_count(c) = molecule_atoms_count (m1) + molecule_atoms_count (m2).
Description	compound_atoms_count(c) is a function that takes molecule1 and molecule2, count the number of atoms in both then return the total number of atoms in compound. Where c is the compound. m1 is the first molecule in the compound. m2 is the second molecule in the compound.
Sources	[3]
Ref. By	IM1

Number	DD3
Label	**Count number of atoms in one side of chemical reaction.**
Symbol	-
SI Units	-
Equation	chemicalEq_atoms_count(eq) = coefficient1 × compound_atoms_count(c1) + coefficient2 × compound_atoms_count(c2).
Description	chemicalEq_atoms_count(eq) is a function that takes one side of a chemical reaction and count the number of atoms in that side. Where eq is the one side of chemical reaction. compound is reactant or product used in the chemical reaction. coefficient is the number preceding the compound.
Sources	[4].
Ref. By	IM1

Number	DD4
Label	**molecular weight**
Symbol	mw
SI Units	g/mol
Equation	mw = $\sum$ atomic weights × molecule atoms
Description	*mw* is the molecular weight of a compound. It is computed by multiplying the atomic weight for the molecule given in periodic table with the amount atoms of molecule then get the total for all molecule atoms forming single compound. Where atomic weights is the weight for single atom of molecule taken from periodic table. molecule atoms is how many atoms this molecule use to form a compound.
Sources	[5]
Ref. By	IM2 , IM4

Number	DD5
Label	**Mole ratio**
Symbol	–
SI Units	mol/mol
Equation	$\frac{coefficient of R2}{coefficient of R1}$
Description	coefficient of R2 is the coefficient for unknown reactant. coefficient of R1 is the coefficient for known reactant. The above equation gives the amount of mole ratio between two reactants by dividing the coefficient amount of unknown mass reactant with coefficient number of known mass reactant.
Sources	[6].
Ref. By	IM3.

4.2.6 Instance Models

This section transforms the problem defined in Section 4.1 into one which is expressed in mathematical terms. It uses concrete symbols defined in Section 4.2.5 to replace the abstract symbols in the models identified in Sections 4.2.3 and 4.2.4.

The goals 1 and 2 are solved by 4.2.6.

Number	IM1
Label	**Balanced chemical reaction**
Input	Unbalanced chemical reaction .
Output	Balanced Chemical reaction = $\sum$ chemicalEq_atoms_count (eq1) = $\sum$ chemicalEq_atoms_count (eq2).
Description	Balance a chemical reaction is done by adjusting the coefficient amount used in unbalanced chemical reaction. Where the total atoms for all reactants used in the equation is equal to total atoms for all products. This is done by: 1- Count the atoms of each side of the chemical reaction. 2- Use coefficients; place them in front of the compounds as needed.
Sources	[4]
Ref. By	DD5

Number	IM2
Label	**Converting mass to mole**
Input	m , mw
Output	mole = $\frac{m}{mw}$
Description	m is the substance mass. mw is the substance molecular weight. The above equation gives the amount of mole for particular substance by dividing the mass amount with molecular weight.
Sources	[7]
Ref. By	IM3

Number	IM3
Label	**Converting mole ratio to mole.**
Input	mole amount of R1 , mole ratio.
Output	mole amount of R2 = $\frac{molofR1}{moleratio}$
Description	mol of R1 is the mole amount for known reactant. mole ratio is the mole ratio between R1 and R2. The above equation gives the amount of mole for R2 substance by dividing the mole amount of R1 with the mole ratio.
Sources	[8]
Ref. By	IM4

Number	IM4
Label	**Converting mole to mass**
Input	mole amount , mw
Output	mass = $mol \times mw$
Description	mol is the mole amount for the substance. mw is the substance molecular weight. The above equation gives the mass for substance by multiplying the mole amount with the molecular weight .
Sources	[8]
Ref. By	GS1

4.2.7 Input Data Constraints

Table 3 shows the data constraints on the input output variables. The column for physical constraints gives the physical limitations on the range of values that can be taken by the variable. The column for software constraints restricts the range of inputs to reasonable values. The software constraints will be helpful in the design stage for picking suitable algorithms. The constraints are conservative, to give the user of the model the flexibility to experiment with unusual situations. The column of typical values is intended to provide a feel for a common scenario. The uncertainty column provides an estimate of the confidence with which the physical quantities can be measured. This information would be part of the

input if one were performing an uncertainty quantification exercise.

Table 3: Input Variables

Var	Physical Constraints	Software Constraints	Typical Value	Uncertainty
mw	mw > 0	-	-	-
m	m > 0	-	-	-

4.2.8 Properties of a Correct Solution

4 shows the data constraints on the output variables. The column for physical constraints gives the physical limitations on the range of values that can be taken by the variable.

Table 4: Output Variables

Var	Physical Constraints
m	m > 0
atoms	$\sum$ atoms for R $=$ $\sum$ atoms for P.

5 Requirements

This section provides the functional requirements, the business tasks that the software is expected to complete, and the nonfunctional requirements, the qualities that the software is expected to exhibit.

5.1 Functional Requirements

R1: The system will use the following inputs:
1- Unbalanced chemical reaction.
2- Mass of R1.
3- Molecular weight for R1 , R2.

R2: The system will produce the mass amount for R2.

R3: The system needs to calculate the followings:
1- number of atoms used in each reactant and product.
2- Coefficients needed for balance chemical reaction.
3- Mole amount for R1.
4- Mole ratio between R1 and R2.
5- Mole amount for R2.

R4: The system will verify the followings:
1- Output is correct.
2- Inputs satisfy the required physical constraints.
3- Mass Conservation Low applied in the chemical reaction.
4- No integer overflow.
5-Outputs satisfy the required output constraints.

R5: The system output must satisfy the following:

1- Mass is not negative number.

5.2 Nonfunctional Requirements

This section provides the non-functional requirements, the qualities that the software is expected to exhibit.

NF1: SMMP provides a correct Solution.

NF2: The ability to test and verify the system.

NF3: The system is understandable for novel users without any ambiguities.

NF4: The system could be used for different applications and modified easily.

6 Likely Changes

LC1: The system may calculate the molecular weight of the reactants.

LC2: The system may ask the user to enter balance chemical reaction.

7 Unlikely Changes

ULC2: The units used in the system will not likely to be change.

8 Traceability Matrices and Graphs

The purpose of the traceability matrices is to provide easy references on what has to be additionally modified if a certain component is changed. Every time a component is changed, the items in the column of that component that are marked with an "X" may have to be modified as well. Table 5 shows the dependencies of theoretical models, general definitions, data definitions, instance models, and likely changes on the assumptions. Table 6 shows the dependencies of theoretical models, general definitions, data definitions, and instance models with each other. Table 7 shows the dependencies of instance models, requirements, and data constraints on each other.

	A1	A2	A3	A4	A5	A6	A7
T1	X	X	X				
DD1							
DD2		X	X				
DD3	X	X	X				
DD4		X	X			X	
DD5	X	X	X				
IM1	X	X	X				
IM2				X	X	X	X
IM3	X X	X					
IM4						X	X
R1	X	X	X	X	X	X	X
R2	X	X	X	X	X	X	X
R3	X	X	X	X	X	X	X
R4	X	X	X	X		X	
R5				X			
NF1	X	X	X	X	X	X	X
NF2							
NF3							
NF4							
LC1						X	
LC2		X	X				
ULC1				X	X	X	X

Table 5: Traceability Matrix Showing the Connections Between Assumptions and Other Items

	T1	DD1	DD2	DD3	DD4	DD5	IM1	IM2	IM3	IM4
T1						X	X		X	X
DD1	X		X	X	X	X			X	X
DD2	X			X	X	X			X	X
DD3	X					X	X		X	X
DD4								X		X
DD5									X	X
IM1	X					X			X	X
IM2									X	X
IM3										X
IM4										

Table 6: Traceability Matrix Showing the Connections Between Items of Different Sections

	IM1	IM2	IM3	IM4	4.2.7	R1	R2	R3	R4	R5
IM1			X	X	X	X	X	X	X	
IM2			X	X	X	X	X	X		
IM3				X			X	X		
IM4					X		X	X		X
R1	X	X	X	X	X		X	X	X	
R2					X	X		X	X	X
R3			X	X	X X	X			X	X
R4							X	X		X
R5							X		X	

Table 7: Traceability Matrix Showing the Connections Between Requirements and Instance Models

9 Values of Auxiliary Constants

NA

References

[1] David L. Parnas and Paul C Clements. rational design process: How and why to fake it. In *IEEE Transactions on Software Engineering 12.2*, pages 251–257., Berlin, Heidelberg, February 1986. Springer Berlin Heidelberg.

[2] https://courses.lumenlearning.com/introchem/chapter/the-law-of-conservation-of-mass/.

[3] https://www.thoughtco.com/what-is-a-molecule-definition-examples-608506.

[4] https://study.com/academy/lesson/balanced-chemical-equation-definition-examples.html.

[5] https://www.thoughtco.com/definition-of-molecular-weight-605369.

[6] https://www.thoughtco.com/definition-of-mole-ratio-and-examples-605365.

[7] https://www.britannica.com/science/mole-chemistry.

[8] https://www.khanacademy.org/science/chemistry/chemical-reactions-stoichiome/stoichiometry-ideal/a/stoichiometry.