Problem Statement and Goals
ChemCode

Samuel Crawford

March 4, 2023

Table 1: Revision History

Date	Author	Change
Jan. 18, 2023	Sam	Create document and fill in Problem, Stakeholders, and Goals sections
Jan. 19, 2023	Sam	Format for Drasil upload; fill in Inputs and Outputs, Environment, and Stretch Goals sections; update Stakeholders and Goals sections; and move Environment section
Jan. 20, 2023	Sam	Update Goals and Stretch Goals sections, fill in Problem Statement introduction, update Drasil reference
Jan. 22, 2023	Sam	Replace the notion of "fractional oxidation state" with "nonstoichiometric compound"
Jan. 31, 2023	Sam	Remove references to SciPy and comments about first-person usage and add stretch goals from SRS Likely Changes
Mar. 4, 2023	Sam	Replace method of solving with integer programming, clarify output, and make minor formatting improvements

1 Problem Statement

The following sections describe the problem statement of ChemCode by outlining the Problem, Inputs and Outputs, Environment, and Stakeholders.

1.1 Problem

Chemistry is a broad field that studies matter and its interactions [1], primarily through chemical reactions. During a chemical reaction, bonds between some

substances break and new ones are formed to create new substances; these reactions are often written as chemical equations [2]. Despite new chemicals being created, all atoms from the initial substances, or "reactants", must be present in the final substances, or "products" because of the Law of Conservation of Matter [2]. This means that for a chemical equation to be useful, it must be balanced by changing the coefficients of the substances involved in the reaction [2]. Additionally, since molecules only exist in whole numbers (since dividing a molecule changes its composition into new types of molecules), these coefficients must be whole numbers, and by convention should be as small as possible [2].

While these equations can be balanced by hand through the process of "balancing by inspection" [2], this can be time-consuming, prone to error, and inefficient, especially for more complicated chemical reactions. For each element present in the reaction, an equality can be written for the number of elements in each substance, with the reactants on one side and the products on the other, using the coefficients of each substance as the variables [3]. These equalities can then be used as constraints in an integer programming problem that, when solved and properly bounded, gives the smallest positive whole numbers that balance the chemical equation [2, 3]. This method can also identify reactions that are "infeasible" and balance reactions involving nonstoichiometric compounds [3], which are compounds "in which the numbers of atoms of the elements present cannot be expressed as a ratio of small whole numbers" [4].

1.2 Inputs and Outputs

Input:

- A representation of a chemical equation

Outputs:

- If the inputted chemical equation is feasible, a representation of it in its balanced form with the smallest whole number coefficients possible
- If the inputted chemical equation is infeasible, a message describing why

1.3 Environment

ChemCode will be developed using Drasil [5], "a framework for generating high-quality documentation and code for Scientific Computing Software" [6, p. iii] by encapsulating scientific knowledge as "chunks" to be reused among projects [6]. By building this project in Drasil, relevant concepts about chemistry and integer programming must first be added. Therefore, a byproduct of this project is that other programs that use chemistry and/or integer programming can be made using Drasil. The implementation in Drasil places some constraints on this project.

Since Drasil is built on the idea of reusability, external libraries will be used to solve the integer programming problems. This was previously done

with ordinary differential equation (ODE) solvers, since "creating a complete ODE solver in Drasil would take considerable time, and there are already many reliable external libraries ... tested by long use" [7, p. 24]; these rationales also apply to solvers of integer programming problems.

Additionally, Drasil can currently generate code in Python, C++, C#, Java, and Swift [7]. The scope of this project will be limited to generating code in Python since it is the language in which I have the most experience.

Since both ChemCode and Drasil are purely software systems, the only hardware involved is the user's computer used to run ChemCode.

1.4 Stakeholders

The main stakeholder of this project is Dr. Spencer Smith, the instructor for the CAS 741 Development of Scientific Computing Software course for which this project is being completed. Dr. Smith and Dr. Jacques Carette are in charge of the Drasil project that ChemCode seeks to extend, so the implementation and development process are of significance to them. Likewise, any future developers of Drasil, including myself, are potential stakeholders of this project, since they may use features added to Drasil, such as ideas about chemistry or integer programming. Jason Balaci, a fellow CAS 741 student and Drasil contributor, is of particular mention, since there may be some overlap between our projects so we may be collaborating throughout this project. I am also a stakeholder of ChemCode as the developer.

More generally, anyone in the field of chemistry in at least a high-school level may be a stakeholder of this project, as they may use this tool in their work.

2 Goals

The goals of this project are to develop a program that...

- can balance chemical equations (including ones with nonstoichiometric compounds).
- can determine if a given chemical reaction is "infeasible" (i.e., not able to be balanced).
- is generated by Drasil (along with relevant documentation).
- extends Drasil by introducing the concepts from chemistry necessary to balance equations, such as elements, compounds, and reactions.
- is written in Python (see Environment).
- uses appropriate external libraries to solve integer programming problems (see [7, Ch. 4]).

3 Stretch Goals

In descending order of priority, the stretch goals of this project are to...

1. add support for more complex chemical formulas, such as hydrates or those with polymers or isotopes.
2. add the ability to, given the amount of one substance in a reaction (in moles), calculate the amount of every other substance in the reaction (also in moles).[1]
3. add the ability to, given the amount of each reactant (in moles), determine the limiting reactant(s) in a reaction.[1]
4. add the ability to, given the amount of each reactant (in moles), determine the theoretical yield of each product and the amount of excess reactant(s).[1]
5. generate code for ChemCode in the other languages supported by Drasil. (While using external ODE solvers in Drasil, the developers "did not find a suitable library for Swift" [7, p. 24]; a similar problem may arise when using external solvers of integer programming problems, meaning that ChemCode may not be generated in all five languages supported by Drasil.)
6. add the ability to parse valid but incorrectly formatted chemical formulas inputted by the user and format them correctly when outputting them.
7. do the same as 2-4. but in terms of mass.[1]
8. add the ability to classify a chemical reaction as "combination (or synthesis), decomposition, combustion, single replacement, ... double replacement" [2, p. 301] or some combination of these.[1]
9. add support for phase labels.[1]
10. add support for precipitation reactions, including solubility and identifying when reactions will not take place.[1]

[1] These examples of problems related to chemical equations were taken from [2].

References

[1] E. Gordon and Furman University, *CHM101: Chemistry and Global Awareness*. Greenville, SC, USA: LibreTexts, Jan. 2023.

[2] L. Lund and Anoka-Ramsey Community College, *Introduction to Chemistry*. Cambridge and Coon Rapids, MN, USA: LibreTexts, Jan. 2023.

[3] I. Hamid, "Balancing Chemical Equations by Systems of Linear Equations," *Applied Mathematics*, vol. 10, pp. 521–526, July 2019.

[4] The Editors of Encyclopaedia Britannica, "nonstoichiometric compound," Oct. 2010.

[5] J. Carette, S. Smith, J. Balaci, T.-Y. Wu, S. Crawford, D. Chen, D. Szymczak, B. MacLachlan, D. Scime, and M. Niazi, "Drasil," Feb. 2021.

[6] B. MacLachlan, *A Design Language for Scientific Computing Software in Drasil*. MASc thesis, McMaster University, Hamilton, ON, Canada, June 2020.

[7] D. Chen, *Solving Higher-Order ODEs in Drasil*. MEng thesis, McMaster University, Hamilton, ON, Canada, Sept. 2022.