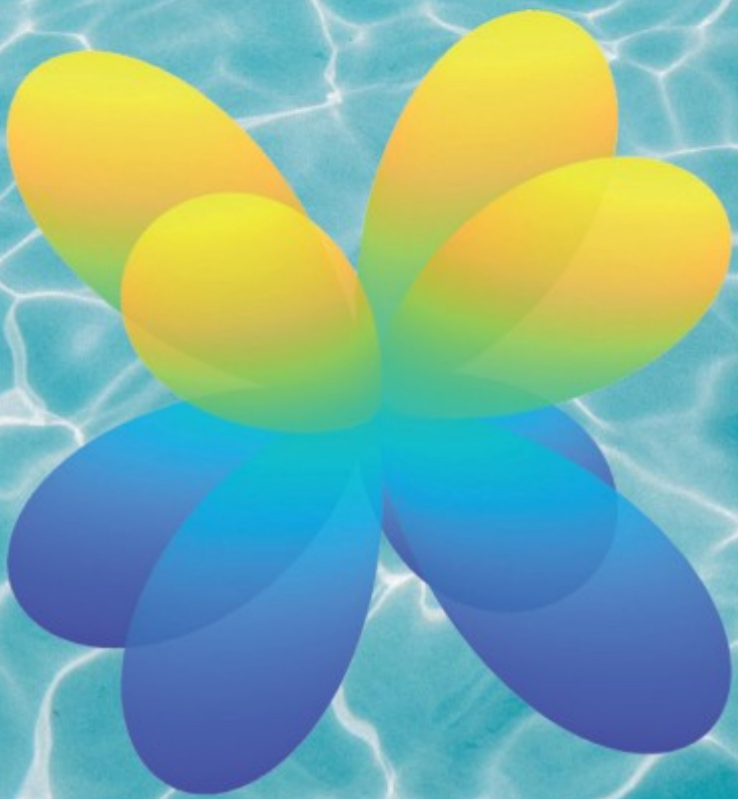
FUNDAMENTAL CHEMISTRY WITH MATLAB

Daniele Mazza | Enrico Canuto



ELSEVIER

Fundamental Chemistry with MATLAB

A guide to using MATLAB scripts, models, and algorithms to explore the fundamentals and applications of key topics in chemistry

Key Features

- Provides practical examples of using the MATLAB platform to explore contemporary problems in chemistry.
- Outlines the use of MATLAB Simulink to produce block diagrams for dynamic systems, such as in chemical reaction kinetics
- Heavily illustrated with supportive block-diagrams and both 2D and 3D MATLAB plots throughout

MATLAB offers huge capabilities for data processing, graphical presentation and modelling, making it an important tool for chemists in navigating their increasingly data-driven field. Fundamental Chemistry with MATLAB highlights how this functionality can be used to explore the fundamentals and applications of key topics in chemistry.

After an introduction to MATLAB, the book goes on to provide examples of its application in both fundamental and developing areas of chemistry, from atomic orbitals, chemical kinetics and gaseous reactions, to clean coal combustion and ocean equilibria amongst others. Complimentary scripts and datasets are provided to support experimentation and learning, with scripts thoroughly outlined enabling the reader to modify them to fit their particular needs.

Drawing on the experience of its expert authors, *Fundamental Chemistry with MATLAB* is a practical guide for all those in chemistry who are interested in harnessing the scripts, models, and algorithms of the MATLAB platform to further enhance their use, understanding and visualization of data.

About the Authors

Daniele Mazza was Professor of Chemistry and Materials Science at the Politecnico di Torino. His main research fields were the electrical and crystallochemical properties of ionic conductors and ceramic superconductors.

He is author or co-author of more than 80 publications in international journals about solid state chemistry, crystal structure determination, X-ray diffraction and modelization of ionic movement in crystalline lattices. In addition, he is author of three books in chemistry. His current research interests focus on environmental and climatic issues, in particular ocean chemistry and CO_2 equilibria in seawater.

Enrico Canuto has taught Automatic Control for more than 40 years at Politecnico di Torino, Turin, Italy. He developed and applied the Embedded Model Control methodology for the design and implementation of digital control systems. He contributed to data reduction of the European astrometric mission Hipparcos, concluding with the publication of the Hipparcos star Catalogue of 120,000 stars. He was active in the design of spacecraft control systems, contributing to the European GOCE mission, to other forthcoming missions, and to instruments for space qualification like the Nanobalance thrust-stand. In the last ten years he has collaborated with the Center for Gravity Experiments, Huazhong University of Science and Technology, Wuhan, and the Tianqin Centre, Sun-Yat-Sen University, Zhuhai, China, in the field of scientific space missions. He has authored one book and published over 150 papers.

Introduction

Matlab language and environment

Most developers today use the so called ‘ third-generation languages’ such as C, C++, Python, and Java. A third-generation language, a general purpose language in nature, gives the developer the kind of precise control needed to write exceptionally fast applications that can perform a wide array of tasks. Fourth-generation languages, like Matlab (short for Matrix Laboratory), on the contrary, are designed with a specific purpose in mind, which in the case of Matlab, is *scientific and technical computing*. In this sense, it has been designed for empowering the user to work with heterogeneous collections of data, rather than individual variables, making it easier for the user to focus on the task, instead on the language.

As programming languages progress through generations, they become more intelligible and closer to human abstraction and language. Matlab took advantage since the early developments of the abstraction levels inherent in the mathematical language and operations, thus offering the user a mathematical environment suitable to any hardware and operating system platform. Subsequent developments added graphical environments like that of Simulink devoted to dynamic systems and their automatic control (to be employed in Chapters 3 and 4), Graphical user interfaces (GUI), document editing, and a rich set of toolboxes covering a wide spectrum of scientific and technical computing and real-time applications in measurements and control.

The chemistry computations proposed in the book will just employ the Matlab core, without any reference to Matlab toolboxes. To this end, we will exploits functions of the following areas addressed by the core: Algebra, Linear algebra (many equations dealing with many unknowns), Calculus, Differential equations, Optimization, Linear regression, Statistics, Curve fitting, Graphing.

Code listing in this book is specifically designed to work on Matlab platform, however with a little efforts most of the simple scripts can be exported to Octave, a very similar but free platform. Indeed if you are looking for an open-source environment close to Matlab in terms of compatibility and computational ability, then Octave is the best alternative. It runs on any operating system without any modifications. Scilab is another open-source option for numerical computing which runs across all the major platforms. Like Octave it is very similar to Matlab in its implementation, although exact compatibility is not a goal of the project developers. It has the advantage of possessing a graphical interface similar to Simulink, named Xcos .

The Matlab scripts

Any computational exercise will be accompanied by a Matlab script, name like InitChem.m, where <.m> is the extension of Matlab scripts, collecting data from external files, performing computations and providing results either in a tabular form in the Command Window or graphical plots in appropriate windows. No specific introduction to Matlab is

provided, except to Simulink in the Appendix D (Chapter 14), but the book scripts and the key Matlab functions will be explained in some detail. The web version of the book will show the default colors which distinguish four kinds of sentences and variables in the script:

1) Black is reserved to statements to be executed.

2) Green is reserved to comments which are introduced by the % character.

3) Bold green is reserved to comments introduced by the %% characters. They split the script into sections which can run separately or step by step. Such comments have been exploited to split the script into sections with different functionalities, for instance, initialization, user data statements, data collection from external excel sheets, main computation, result printout, 2D and 3D graphical plots.

Fairly all of the scripts designed to produce the book graphical plots, are reported in the book itself. The complete library of scripts (main scripts and functions) of extension <.m>, excel files of extension <.xlsx> and Simulink block diagrams (models) of extension <.slx> can be found in the companion website [8] . Care has been taken to ensure the script execution without warnings and errors, at least as they are reported in the book and in the editor's website. Of course, any bug can be reported to the authors via their e-mail addresses.

Simulink models of extension <.slx> could not be included in the book (see the Appendix D, Chapter 14), but the image of several block-diagrams has been added to Chapters 3 and 4.

Matlab scripts and Simulink models have been developed with an academic license of Matlab/Simulink. No Matlab Toolbox has been employed.

The book topics

Fundamental or general chemistry is a huge field, therefore the ten chapters of this book will focus on topics which are only partial sections of this field. The curious reader will be able to spot similarities in adjacent fields of chemistry, not covered yet (but may be a future treatise), and to adapt codes to this aim. Clues to this are be given below, in the summary of chapter's topics. Numerical data elaboration for advanced techniques in chemistry (e.g. X-ray diffraction, NMR analysis, molecular dynamics or lattice energetics) are not covered in this context; the relevant programs are to be found in specific web sites or textbooks, being mainly addressed to expertises in the respective sectors.

Book chapters can be arranged into four groups, plus six appendixes of help to specific topics and chapters.

Group one: atomic orbitals

Plotting atomic orbitals with Matlab (Chapter 1). The mathematical description of the ondulatory behavior of electrons in atoms is shortly reviewed. On this basis, electron wave functions are described and plotted in a simplified manner using Matlab surf and fsurf functions. Some elements of wave functions theory and the Schroedinger equation is

recalled in the Appendix F (Chapter 16). Molecular orbitals are not covered yet but could be in a next edition.

Group two: stoichiometry and kinetics

Balancing chemical reactions with Matlab (Chapter 2). The algebra underlying to the balancing of chemical reactions is highlighted. In such a way, a generic and straightforward solution is devised for simple acid-base or complex redox reactions. This could help solve troublesome issues, being the correct mass balance of utmost relevance in general and educational chemistry. The foundations of the algebraic algorithm employed are reviewed in the Appendix A (Chapter 11). Only some examples of reactions are given, while the reader will exploit other as well among the many possible.

Chemical kinetics aided by Matlab/Simulink (Chapter 3). The study of the rate of a chemical reaction both in textbooks and in advanced applications requires to deal with differential equations. Their solution, even for simple applications, is here easily enabled by Matlab ODE (Ordinary Differential Equations) solver and by Matlab/Simulink block-diagrams. The treatment agrees with the methods of system and control engineering. The foundations are recalled in the Appendix B (Chapter 12). The two case-studies, NO oxidation and ozone decomposition, can be extended to many other known reactions

More complex kinetics aided by Matlab/Simulink (Chapter 4). As an example for those interested in more complex kinetics, like the famous oscillating reactions, here a complete description is to be found, with stunning graphics and plotting, enabled by Matlab. As already mentioned, this topic may be of interest of researchers in system theory and control engineering.

Group three: gases and vapors

Gaseous reactions and equilibria aided by Matlab (Chapter 5). In dealing with thermodynamics of even simple equilibrium reactions in a varying temperature/pressure environment requires a precise approach. Matlab functions enable to read excel files with the parameters needed to calculate entropy, enthalpy and free energy and therefrom the equilibrium parameters for some reactions. By exploiting the indicated thermodynamic database other reactions will be tackled, among the many of interest in environmental issues.

Physical properties of gases and vapors aided by Matlab (*Chapter 6)*. Electric interaction between molecules are responsible for the phase transformation of gases to liquids. A number of equations can be used for describing the behavior of real gases, in a wide range of temperatures and pressures, which are graphically displayed by Matlab scripts. A gas of choice can be used, as long as its van der Waals parameters are known.

Group four: aqueous solutions

Exploring acid-base equilibria in water with Matlab (Chapter 7), In general chemistry this is a relevant topic. Many Matlab functions, among them fzero, enables in a few program lines to solve even complex equilibria of this kind. Examples range from monoprotic acids

to titratrion of polyprotic acids. Though a few working examples are given, the reader will be able to insert other acid/base equilibria in solution, even admixtures of acids and bases.

Colligative properties of solutions aided by Matlab (Chapter 8). These properties only depend on the solute molar concentration and temperature, being independent of the nature of the solute particles. As an example, aqueous solution of NaCl is chosen and discussed, but the conclusion can be extended to all other solutions of this type, including other binary, ternary or complex salts.

Exploring seawater chemical equilibria with Matlab (Chapter 9). Seawater is not a simple reservoir of different dissolved salts, like sodium chloride, but is capable of reacting with different substances like the atmosphere's carbon dioxide, by modifying its concentration and buffering the anthropogenic increase. This is one of to-day most debated topics. The only quantitative way to tackle the issue is by proper solution of chemical equilibrium equations. All this is discussed, explained and calculated by the simple use of Matlab scripts. As with other scripts of this book, the reader can vary the relevant parameters, and discover what happens. The variety of simulations is really amazing, but only some of them are discussed

Prevalence diagrams for some common elements aided by Matlab (Chapter 10), Any element in aqueous solution is capable of transforming into different chemical forms, according to pH and E_H, the redox potential of the same solution. In this way, we obtain pH/E_H diagrams, known as prevalence (or Pourbaix) diagrams (M. Pourbaix, 1904-1998). Matlab graphical capabilities enable an easy representation of these diagrams for some common elements, like Fe, S and Mn. The methods can be applied as well to other elements, by providing the pertinent electrochemical potentials.

Appendices

Appendix A (Chapter 11): Introduction to linear algebra

Appendix B (Chapter 12): Introduction to dynamic systems

Appendix C (Chapter 13): Introduction to linear regression

Appendix D (Chapter 14): Introduction to Matlab Simulink

Appendix E (Chapter 15): Table of sea water coefficients

Appendix F (Chapter 16): Introduction to the Schroedinger equation

Authorship and acknowledgments

The book was conceived by the first author, as an offspring of his several decade experience as a teacher of fundamental chemistry at Politecnico di Torino, Turin, Italy, and recently as author of two books on current hot topics in environment chemistry, The second author was firstly involved to improve the stoichiometric balance of Chapter 2, taking advantage of the nullspace algorithm. Despite his scholar-level chemistry knowledge, he became passionate of the first author's conception, topics and algorithms, leading him to a full immersion in the

draft revision, enrichment and completion. He brought two-year experience in the conception and writing of the eight hundred pages of the book as well as a lomg-time experience in the field of dynamic system simulation and control aided by Matlab/Simulink environment.

A first draft of the book was written with the Word text editor of Microsoft Office, but soon the authors realized that their collaboration could greatly benefit from a free text editor like Writer of the LibreOffice suite. The only Microsoft Office residuals are the excel tables, which accompany, as data sources, the abundant suite of Matlab scripts presented by the book.

Last but not least, the work of both authors could not have been matured without the patience and support of their respective wives, Maria Teresa and Maria Angela.