[8] P. Seleson, Q. Du, and M. L. Parks, *On the consistency between nearest-neighbor peridynamic discretizations and discretized classical elasticity models*, Comput. Methods Appl. Mech. Engrg., 311 (2016), pp. 698–722.
[9] S. A. Silling, *Reformulation of elasticity theory for discontinuities and long-range forces*, J. Mech. Phys. Solids, 48 (2000), pp. 175–209.
[10] X. Tian, *Nonlocal Models with a Finite Range of Nonlocal Interactions*, Ph.D. thesis, Columbia University, 2017.

Christian Vollmann
Trier University

Dynamical System Models in the Life Sciences and Their Underlying Scientific Issues. *By Y. M. Wan.* World Scientific, Hackensack, NJ, 2018. $128.00. xx+379 pp., hardcover. ISBN 978-981-3143-33-3.

This book is based on a premise: there exists a "third way" to teach mathematical modeling for life sciences applications that adds one complementary direction to the classic "case studies" and "method-based" approaches. This third way focuses on the scientific issues. As someone who does share this point of view and has tried to use this approach in both research and teaching, I found this quite promising, and it is in this frame of mind that I read the book, with the (hereby avowed) aim to decide whether I can use some of the material in an undergraduate course on modeling.

The book consists of four main parts, together with some appendices. Part 1 concerns the growth of a (single) population, Part 2 deals with interacting populations, Part 3 introduces issues involving optimization, and Part 4 presents constraints and control. Each part comprises several chapters dealing with a variety of topics. My understanding is that there will be two other volumes devoted to PDEs and stochasticity, so the focus here is on approaches involving ordinary differential equations. Exercises are present throughout the book, and some sample assignments are provided in one of the appendices. The preface also explains how the material in the book is used at UCI.

Mathematical biology textbooks abound, so why should you buy this one? What can you expect? Let me start by pointing out some aspects I do find annoying from the aspect of personal taste. A pet peeve of mine is the use of nonexplicit state variable names, and this book sometimes takes that direction: the first variable is x, the second y, etc. Another pet peeve of mine is "nontheoremized" discourse à la physicist. There are theorems and proofs in this book, but their presence seems almost random. Also, the focus on ODEs belies the objective to "let the science do the directing"; if the latter were really true, then there would be all sorts of models, not just ODEs and (in later volumes) PDEs and stochastic systems. Another comment is that for a book that emphasizes scientific issues, there are surprisingly few direct references to the underlying science. The mathematical biology references are completely appropriate, but if science is doing the talking, it should not be through the mouths of us mathematical biologists. Finally, I tend to favor a more integrated presentation of numerics.

These points are by no means deal breakers and probably only lay bare my biases. They also emphasize how difficult an endeavor a project like Wan's can be, and they are sometimes even strengths: a more integrated approach to numerics, for instance, requires the choice of one, perhaps two programming languages, thereby making life awkward for those using another solution.

Now, how does the book hold up to its objective to focus on science? Quite well, I would say. The range of topics covered is quite extensive. The usual population models are present (logistic growth, fish harvesting, predator-prey, etc.), but some unusual problems typically found in more specialized books or textbooks are also discussed, such as DNA mutation, HIV and drug dynamics, or the fastest time to cancer. The treatment of these subjects varies from very short presentations a couple of pages long to entire chapters. The same is true of the mathematical techniques presented: defective matrices take seven pages, optimization over a planning period makes up a whole chapter. Speaking of optimization, I particularly enjoyed the fact that both this and control are techniques covered in the book, together making up two entire parts. As with some of the biological problems under consideration, this eclecticism is refreshing.

Altogether, this is an interesting book that is difficult to pigeonhole. I think I might take it, or at least parts of it, for a test ride the next time I teach a modeling course.

JULIEN ARINO
University of Manitoba

Numerical Python: Scientific Computing and Data Science Applications with Numpy, SciPy and Matplotlib. *By Robert Johansson.* Springer, New York, 2019. $44.99. xxiv+700 pp., softcover. ISBN 978-1-4842-4245-2.

Johansson's book ambitiously aims to familiarize the reader with the current numerical and scientific computing ecosystem within the Python programming language. The book's topics include vectors, matrices, visualization, computer algebra/symbolic computing, integration, differential equations, data storage, regression, Monte Carlo methods, and more. The book ultimately succeeds in its goal, and readers will feel satisfied having learned a diverse set of Python tools.

About the Author. Robert Johansson has made several contributions related to Python scientific computing. He was one of the original developers of QuTip, a popular quantum toolbox in Python [1]. As of writing this review, Johansson remains the individual with the most contributions to the QuTiP source code.[1] Additionally, Johansson has a popular lecture series on scientific Python,[2] which I imagine was an inspiration for this book.

Intended Audience. This book is intended for readers who want to learn about scientific computing within the Python programming language. I believe it will be most useful for individuals with prior experience in numerical methods and programming, as these individuals will be able to quickly apply concepts from the book to practical problems. Readers with little experience with a numerical method will find the author's explanation adequate to understand the practical importance of the example.

This book is not for readers unfamiliar with computer programming. The author even states, "readers without experience in Python programming will probably find it useful to read this book together with a book that focuses on the Python programming language itself." The book fails to detail much of the syntax and semantics of the Python language that other books might have included. The trade-off here is that this book includes a more diverse set of numerical methods (and Python libraries) than other books. I'd recommend [2] as a suitable text to accompany this book for readers familiar with numerical methods and programming, but not the Python language.

Layout and Structure of the Book. The book includes 19 chapters and a small appendix section. Each chapter follows a similar format that starts with a general introduction, followed by a short list of the Python modules required for that chapter. The bulk of the material follows an IPython/Jupyter Notebook learn-by-example style, which makes it simple to follow every line of code in the most complicated examples. The chapters end with a single paragraph summary, followed by a single paragraph on recommended additional reading. The first chapter and appendix cover basic details on the installation and execution of Python code. As presented, this is the bare minimum to get up and running with Python. The remaining chapters each cover a specific numerical topic.

[1] https://github.com/qutip/qutip/graphs/contributors
[2] https://github.com/jrjohansson/scientific-python-lectures