



# The Dawning of a New Era in Applied Mathematics

Weinan E

*Note: The opinions expressed here are not necessarily those of Notices.*

## 1. The Keplerian and Newtonian Paradigms

Ever since the time of Newton, there have been two different paradigms for doing scientific research: the Keplerian paradigm and the Newtonian paradigm. In the Keplerian paradigm, or the data-driven approach, one extracts scientific discoveries through the analysis of data. The classical example is Kepler's laws of planetary motion. Bioinformatics provides a compelling illustration of the success of the Keplerian paradigm in modern times. In the Newtonian paradigm, or the first-principle-based approach, the objective is to discover the fundamental principles that govern the world around us or the things we are interested in. The best example is theoretical physics through the work of Newton, Maxwell, Boltzmann, Einstein, Heisenberg, and Schrödinger. It is still a major playground for some of the best minds today.

The data-driven approach has become a very powerful tool with the advance of statistical methods and machine learning. It is very effective for finding the facts, but less effective for helping us to find the reasons behind the facts.

The first-principle-based approach aims at understanding at the most fundamental level. Physics, in particular, is driven by the pursuit of such first principles. A turning point was in 1929 with the establishment of quantum mechanics: as was declared by Dirac [2], with quantum

mechanics, we already have in our hands the necessary first principles for much of engineering and the natural sciences except physics at exceptional scales.

However, as was also pointed out by Dirac, the mathematical problem that describes the laws of quantum mechanics is exceptionally complicated. One of the difficulties is that it is a many-body problem: with the addition of one electron, the dimensionality of the problem goes up by 3. This is the dilemma we often face in the first-principle-based approach: it is fundamental but not very practical. Consequently in practice we often have to abandon the rigorous and elegant theories and resort to ad hoc and nonsystematic approximations. The price we pay is not just the lack of rigor and elegance, but also the reliability and transferability of the results.

Applied math has developed along a similar line. Since the first principles of physics are formulated in terms of partial differential equations (PDEs), the analysis and numerical algorithms for PDEs has occupied a central role in applied math, particularly during the period from the 1950s to the 1980s. The objective is three-fold: solving practical problems, understanding the mathematics behind them, providing physical insight into these practical problems. A very compelling success story is fluid mechanics. Not only has fluid mechanics been a major driving force for the study of PDEs, the fact that fluid mechanics research has largely become a computational discipline is also testimony to the success of the numerical algorithms that have been developed. The study of these PDEs and the numerical algorithms have been a central theme in applied math for many years and it is still an active area today.

---

Weinan E is a professor of mathematics at Princeton University. His email address is [weinan@math.princeton.edu](mailto:weinan@math.princeton.edu).

Communicated by Notices Associate Editor Reza Malek-Madani.

For permission to reprint this article, please contact:  
[reprint-permission@ams.org](mailto:reprint-permission@ams.org).

DOI: <https://doi.org/10.1090/noti2259>

When I was a graduate student at UCLA, we were proudly told that we belonged to the camp of “Courant-style applied math.” This term was coined to make a distinction with the “British-style applied math.” Both focused on fluid mechanics. The British-style championed physical insight and asymptotics. The leaders, Taylor, Batchelor, C. C. Lin, Lighthill, et al., were not only great applied mathematicians but also leading theoretical fluid dynamists. It is also known that they generally did not hold numerics and rigorous analysis in very high regard. The Courant-style championed numerics and theorems (“theorem provers”). Its philosophy was that as long as the underlying PDEs and numerical algorithms are solid, much can be learned through computing. After all, the physical processes are very complicated; one cannot go very far without computing. Some of their leaders, such as von Neumann, Courant, Friedrichs, and Lax, are not only great applied mathematicians, but also great pure mathematicians. The fact that the feud between these two schools was considered the main animosity in applied math speaks for the dominance of fluid mechanics during those times.

The card-carrying data-driven research community has been statistics. For whatever reason, until quite recently, statistics has developed pretty much independently of applied math, and in fact, independently of mathematics. It was very rare that a math department or applied math program contained statistics. It is only in recent years that there has been a call for change.

This does not mean that the applied math community has not been interested in the data-driven approach. To the contrary, since the late 1980s, with research work on wavelets and compressed sensing, signal and image processing has taken a center stage in applied math. In fact, this applied math version of a data-driven approach has been among the most productive areas of applied math in the last thirty years.

Neither does it mean that fluid mechanics was the only successful area for applied mathematicians interested in PDEs. In fact, some would argue that solid mechanics was equally successful: after all it was from solid mechanics that the finite element method, one of the most important success stories in applied mathematics, was originated. Another success story is numerical linear algebra: one just has to look at how popular Matlab is to appreciate its widespread impact. This list goes on.

## 2. Crisis for the “Courant-style Applied Math”

Unfortunately for my generation of “Courant-style” applied mathematicians, the dominance and success in fluid mechanics presented more of a challenge than an opportunity. The ground work for PDEs and fluid mechanics was already laid down by the generations before us. We were left to either address the remaining problems, such as turbulence, or conquer new territory. Both have proven to be

difficult, not to mention reproducing the kind of success that applied math had before in fluid mechanics.

Indeed after fluid mechanics, the Courant-style applied math has spread to many other scientific and engineering disciplines, such as material science, chemistry, biology, neuroscience, geoscience, and finance, with a lot of success. But generally speaking, the degree of success in these areas has not matched what we saw in fluid mechanics. Our contributions are welcomed, but they tend to be incremental rather than transformative. As a result, to deal with the central issues that they face, scientists or practitioners often have to resort to ad hoc approximations which are both unreliable and unpleasant. This situation is found in quantum mechanics, molecular dynamics, coarse-grained molecular dynamics, studies of chemical reactions, complex fluids models, plasticity models, protein structure and dynamics, turbulence modeling, control problems, dynamic programming, etc.

The central difficulty for most, if not all, of these problems is that they are intrinsically high-dimensional problems, and we are haunted by the curse of dimensionality.

For most of the problems listed above, the high dimensionality is the result of the multiscale nature of the problem, and one glimpse of hope was brought by the idea of multiscale, multiphysics modeling. By lumping together the inessential degrees of freedom at small scales, one should be able to directly use the more reliable microscale models to come up with much more efficient algorithms for the macroscale process we are interested in. Although very promising, so far the success of multiscale modeling has been limited due to the following:

1. The microscale models themselves are often not that reliable. For example when studying crack propagation, we often use molecular dynamics as the microscale model. But the accuracy of these models for processes that involve bond breaking is often questionable.
2. Even though multiscale modeling can drastically reduce the size of the microscale simulation required, it is still beyond our current capability.

## 3. Machine Learning Comes to the Rescue

The heart of the matter for the difficulties described above is our limited ability to handle functions of many variables, and this is exactly where machine learning can make a difference. By providing the ability to approximate functions of many variables, things that were considered impossible before become highly likely now.

Before machine learning, the one area in which functions of many variables were handled quite routinely was numerical integration. In statistical physics, we almost take for granted our ability to compute integrals of functions of millions of variables and forget to notice how remarkable this actually is. This is made possible by the

Monte Carlo algorithm and the variance reduction techniques that have been developed throughout the years. For one thing, in contrast with the grid-based algorithms such as Simpson's rule, the rate of convergence of the Monte Carlo algorithm is independent of the dimension.

Approximating functions in high dimensions is a much more difficult task, and the success of machine learning did not come easy [6]. Although neural networks models were discovered a long time ago, it was only recently that their full potential in approximating functions of many variables was recognized. However, during a short period of time, we have already seen a number of remarkable achievements made on several long-standing problems with the help of machine learning, with many more that promise to come in the near future.

The integration of machine learning into applied math will change both disciplines in a fundamental way. In the following, we will discuss some specific examples to illustrate the impact this will have on scientific computing, modeling, machine learning, and pure mathematics.

#### 4. Control Theory and PDEs in High Dimension

One of the first successful applications of machine learning in scientific computing is a deep-learning-based algorithm for high-dimensional control problems [5]. To begin with, it is interesting to note that the term "curse of dimensionality" was first coined by Richard Bellman in the context of dynamic programming [1]. Indeed, the dimensionality of the Bellman equation is the same as that of the state space of the control problem: if we are interested in controlling a PDE, the Bellman equation is infinite dimensional. This has severely limited the application of "first-principle-based" control theory and many practical problems have to be solved using ad hoc ansatz, just like what is done for quantum many-body problems.

Within the framework of closed loop control, the optimal policy function is a function of the state. If one parametrizes this policy function by a neural network, then there is a very nice similarity between stochastic control and deep learning: the cost function of the control problem is the loss function; the dynamical system for the control problem plays the role of the deep residual network; the noise in the dynamical system plays the role of training data that allows us to use the stochastic gradient descent algorithm for training. With this deep-learning-based algorithm, one can handle quite routinely stochastic control problems in hundreds and even higher dimensions [5]. It has also been extended to deterministic control problems [7] and general nonlinear parabolic PDEs.

These algorithms have opened the door to dealing with real-world control problems and high-dimensional PDEs. This is an exciting new possibility that should impact (and to some extent has already impacted) economics, finance, operational research, and a host of other disciplines.

#### 5. Machine Learning Assisted Modeling

In physics we are used to first-principle-based models. These models are not only widely applicable but also simple and elegant. Schrödinger's equation is a good example. Unfortunately as was pointed out earlier, solving practical problems using these models can be an exceedingly difficult task. For this reason, seeking simplified models has been a constant theme in physics, and science in general. However, as we have experienced with turbulence models, it is often very hard to come up with such simplified models if we do not resort to ad hoc approximations.

Machine learning is poised to provide a major boost to our ability to develop such physics-based models. This can happen and is already happening in three different ways. First, it provides the missing tool that can help to turn the dreams of multiscale modeling into reality. Secondly, it provides a framework for developing models directly from data. Thirdly, it may provide a very powerful tool for integrating physical models with observations, along the lines of data assimilation.

However, fitting data is one thing, constructing interpretable and truly reliable physical models is a quite different matter. Let us first discuss the issue of interpretability. It is well known that machine learning models carry a reputation of being "black boxes," and this has created a psychological barrier for using machine learning to help develop physical models. To overcome this barrier, note that interpretability should also be understood in relative terms. Take Euler's equation for gas dynamics as an example. The equations themselves are clearly interpretable since they represent nothing but the conservation of mass, momentum, and energy. But it is less critical whether the details of the equation of state can be interpreted. Indeed for complex gases, the equation of state might be in the form of a code that comes from interpolating experimental data using splines. Whether the coefficients of these splines are interpretable is not our real concern. The same principle should apply for machine-learning-based models. While the fundamental starting point of these models should be interpretable, just like the conservation laws in gas dynamics, the detailed form of the functions that enter into these models do not all have to be interpretable. These functions often represent some constitutive relations just like the equation of state for gas dynamics.

Turning now to the issue of reliability. Ideally we want our machine-learning-based model to be as reliable as general physical models such as the Navier-Stokes equation, for all practical purposes. To make this happen, two things are crucial. The first is that the machine-learning-based model has to satisfy all physical constraints such as the ones coming from symmetries and conservation laws. The second is that the data we use to train the model has to be

rich enough to adequately represent all the physical situations encountered in practice. Since labeling the data is almost always very expensive, selecting a good dataset that is both small and representative is a very important component for developing such models. We will say more about this in the next section.

These ideas have already been successfully applied to a number of problems, including molecular dynamics and rarefied gas dynamics [4]. In the case of molecular dynamics, machine learning, combined with high performance computing, has made it possible to simulate systems with hundreds of millions of atoms with *ab initio* accuracy, an improvement of five orders of magnitude (see [4] for reference).

These new developments are already quite exciting. But the impact of machine learning assisted modeling will be felt the most in areas such as biology and economics where first-principle-based modeling is difficult. Some exciting progress is already underway in these areas.

## 6. A New Frontier in Machine Learning

The integration of machine learning with applied math also leads to some new opportunities in machine learning. Here we discuss a couple.

**6.1. Concurrent machine learning.** In most traditional machine learning settings, the training data is either generated beforehand or passively observed. This is usually not the case when machine learning is applied to solving problems in scientific computing or computational science. In these situations, the training data is often generated on the fly. To make an analogy with multiscale modeling, in which one distinguishes sequential multiscale modeling with concurrent multiscale modeling according to whether the multiscale models are generated beforehand or on the fly, we call this style of machine learning *concurrent machine learning*. As was observed earlier, generating a minimal but representative dataset is a key issue in concurrent machine learning.

To this end, one needs an efficient procedure for exploring the state space and a criterion for deciding whether a new state encountered should be labeled or not. One example is the EELT algorithm suggested in [4].

**6.2. “Well-posed” formulation of machine learning.** Besides being surprisingly powerful, neural-network-based machine learning is also quite fragile—its performance depends sensitively on the hyperparameters in the model and the training algorithm. In many situations, parameter tuning is still quite an art even though with accumulated experience, this situation is steadily improving.

Part of the reason is that in machine learning, models and algorithms are constructed before the formulation of the problem is carefully thought out. Just imagine what would happen if we try to model physical processes

without constructing the PDE models beforehand. Indeed having a PDE model to begin with and making sure that PDE model is well-posed is one of the most important lessons that we learned in the Courant-style applied math.

This motivates the question: can we formulate “well-posed” models of machine learning? The expectation is that if we begin with nice continuous formulations and then discretize to obtain practical models and algorithms, the performance would be more robust with respect to the choice of the hyperparameters. Some initial attempts have been made in [3] along this line. Interestingly, a byproduct of the work in [3] is that neural network models are quite natural and inevitable, since the simplest continuous models and discretizations always lead to nothing but neural network models in one form or another. Despite this, this way of approaching machine learning does generate new models and algorithms. More importantly, it encourages us to look for first principles and allows us to think outside the box of neural network models.

A close analogy is found in image processing, say denoising. The standard way of denoising is to directly apply carefully designed filters to the image and see what happens. This approach has been very effective, particularly with advanced wavelet-based filters. Another approach is to write down a mathematical model for denoising, typically in the form of a continuous variational problem, then discretize and solve the discretized model with optimization algorithms. The well-known Mumford-Shah and Rudin-Osher-Fatemi models are examples of such mathematical models. One may question the validity of these mathematical models, but having a well-defined mathematical model to begin with clearly has its advantage. For one thing, it has helped to turn image processing into interesting PDE problems. It has also motivated people to think about the fundamental principles behind image processing, even though not much progress has been made along this line.

The hope is that this new mathematical understanding and formulation will not only help to foster the current success of machine learning but also extend its success to a broad spectrum of other disciplines. After all, machine learning is about function approximation, a very basic issue in mathematics. Having new ways to represent and approximate functions that are particularly effective in high dimension should surely have a significant and broad impact.

## 7. High-Dimensional Analysis

It is not just the application areas that will experience the impact, mathematics itself will also feel the impact, particularly analysis.

Machine learning has brought out a host of new analysis problems in high dimensions, from approximating functions to approximating probability distributions,



dynamical systems, and solving PDEs and Bellman-like equations. Studying these problems will inevitably give rise to a new subject in mathematics: high-dimensional analysis.

In this direction, the one area that has already received serious attention in mathematics is high-dimensional integration. The analysis of Monte Carlo methods, particularly Markov chain Monte Carlo, has been an active area in probability theory and mathematical physics for quite some time.

Integration is the most elementary problem in analysis. One can ask many more advanced questions about functions, probability distributions, dynamical systems, calculus of variations, and PDEs. For example, an important question is to characterize the complexity of these objects. At an abstract level, complexity should be defined by the difficulty with which the given object is approximated by simple elementary objects. For example, for functions, the elementary objects can be polynomials, piecewise polynomials, or neural networks. For probability distributions, the elementary objects can be mixtures of Gaussians.

Take for example the complexity of functions. Classically, this is done through smoothness, namely, how many times the function can be differentiated. Many hierarchies of function spaces have been defined along this line, such as the  $C^k$  spaces, Sobolev spaces, and Besov spaces. In low dimensions, this makes perfect sense. Indeed one can show that functions in these spaces are characterized by the rate of convergence when they are approximated by some classes of elementary functions such as piecewise polynomials or trigonometric functions.

Results of this type suffer from the curse of dimensionality. Indeed it has become increasingly clear that smoothness-based concepts are not the right way for measuring the complexity of functions in high dimensions. Rather one should measure the complexity of high-dimensional functions by whether they can be efficiently approximated by a particular neural network-like model. In this way, one obtains the reproducing kernel Hilbert space (RKHS), Barron space, multilayer spaces, and flow-induced space, each of which is naturally associated with a particular class of machine learning models.

What about PDEs in high dimensions? A natural question is whether we can develop a regularity theory for certain classes of PDEs in the function spaces mentioned above. If we can, this means that one should be able to efficiently approximate the solutions of these PDEs using the corresponding machine learning model. This problem is particularly important for the Hamilton-Jacobi-Bellman equation.

## 8. Applied Math as a Mature Scientific Discipline

Can applied math become a unified subject with a small number of major components, like pure math? Can we have a reasonably unified curriculum to educate applied mathematicians? These questions have long been difficult to address. Looking back, it is clear that the situation just wasn't ripe. On one hand, applied math is truly very diverse, touching upon almost every discipline in science and engineering. Seeking unity and a unified curriculum is undoubtedly a difficult task. On the other hand, the fact that major components such as machine learning were missing from the core of applied math means that it just wasn't ready. Just imagine what pure math would be like without algebra.

The situation has changed. With machine learning coming into the picture, all major components of applied math are now in place. This means that applied math is finally ready to become a mature scientific discipline. Yes new directions will continue to emerge, but there are reasons to believe that the fundamentals will more or less stay the same. These fundamentals are: (first-principle-based) modeling, learning, and algorithms.

**8.1. The main components of applied math.** Algebra, analysis, geometry, and topology constitute the main components of pure math. For physics, they are classical mechanics, statistical mechanics, electromagnetism, and quantum mechanics. What are the main components of applied math? The following is a proposal. It is not meant to be the final word on the subject, but a starting point for further discussion.

Applied math has three major components.

1. First-principle-based modeling, which includes the (physical) models themselves and the analytical tools for these models. To put it simply, the former is about physics and the latter is about differential equations.

The principles behind the physical models are the fundamental laws and principles of physics: the physical setup (e.g., classical vs. quantum, inertia dominant vs. overdamped), the variational principles, the conservation laws, etc.

These first principles are formulated in terms of variational problems or differential equations. Therefore we need analytical tools for dealing with these mathematical problems. Asymptotic methods can quickly capture the essence of the problem and give us much needed insight. Rigorous theorems can help to put things on a solid footing, in addition to shedding light on the problem.

2. Data-driven methods. By far the most important part of data-driven methods is machine learning. But also included are statistics and data (e.g., image) processing.

3. Algorithms. Here we have in mind both algorithms for first-principle-based applications and data-driven

applications. Luckily, there is a lot in common for the algorithms in both areas. One example is optimization algorithms. Not only have they played a pivotal role behind the success of machine learning, many first-principle-based models are formulated as variational problems for which optimization algorithms are needed.

**8.2. Curriculum and education.** Most, if not all, leading universities have rather mature pure math undergraduate and graduate programs. Very few have mature applied math programs. Worse than that, in some cases, applied math courses are taught as a set of tricks, not a unified subject. One example is the “Mathematical Modeling” course. Although this is meant to be a basic introductory course for applied math, it is often taught as a set of examples, without a coherent big picture. The lack of mature applied math undergraduate programs is the one single most important obstacle for applied math, because it hinders our ability to attract young talents.

With the major components of applied math being clear, we can now design a unified curriculum for applied math. Naturally this curriculum centers on the three major components discussed above. We briefly discuss each.

Modeling has two parts: The physical principles for the models, and the mathematical tools for analyzing these models. The former is like the fundamentals of physics, taught to mathematicians. The latter is applied analysis, including ODEs and PDEs, calculus of variations, analysis of probability distributions, asymptotics, and stochastic analysis. Each can be covered by a year-long course.

Learning really means data analysis. It consists of machine learning, data processing, and statistics. There are already mature courses for data processing and statistics suitable for applied mathematicians. The situation for machine learning is different. It is routinely taught in a style suited for computer scientists. We need a way to teach this to mathematicians. At this point, the mathematical perspective of machine learning is not yet a mature subject, but this situation is improving very quickly. We believe that a reasonable mathematical introduction to a machine learning course will be developed soon and it can be taught as a one-semester course.

Algorithms has two parts: Algorithms for continuous objects and algorithms for discrete objects. The former is covered by the numerical analysis course offered in math departments. The latter is covered by the algorithms/discrete math course, typically taught in computer science departments. With machine learning, these two parts are coming together so it is important to teach them in a more unified fashion.

Developing all these courses will take a huge amount of effort, but we should and can make it happen.

## 9. Applied Math as the Foundation for Interdisciplinary Research

With such a program in place, applied math will become the foundation for interdisciplinary research. After all, modeling, learning, and algorithms are the fundamental components of all theoretical interdisciplinary research. The applied math program described above will help to systematize the training of students as well as the organization of interdisciplinary research programs. Should this become reality, it will be a turning point in the history of interdisciplinary research.

All this will take time. For one thing, we need to start from the basics, the training of young students. But before we have a chance to train them, we have to be able to attract them to applied math. The one thing that I have been very impressed by, being a faculty member at Princeton for over twenty years, is how number theory has been able to attract talent. I now believe, to my own surprise, that applied math has the potential to do the same. Applied math shares all the major features that are particularly appealing to young students: the simplicity and elegance of the problems (e.g., machine learning) and the challenge that these problems pose (e.g., turbulence), with the added bonus that it is one of the main avenues towards the most exciting new developments in science and technology. We are beginning to see this change at the graduate level.

In the history of science, there were two periods of time that made the most impact for applied mathematics. The first was the time of Newton, during which it was established that mathematics should be the language of science. The second was the time of von Neumann, during which it was proposed that numerical algorithms should be the main bridge between mathematics and science. Now the third time is at the horizon, a time when all the major components of applied math are in place, to form the foundation of not only interdisciplinary scientific research but also exciting technological innovation. This is truly an exciting time. Let us all work together to make this a reality!

**ACKNOWLEDGMENTS.** I am very grateful to Professors Engquist and Osher for their suggestions on an earlier draft, as well as the inspirations they provided over the years which greatly influenced my career and my view of applied mathematics. I also want to thank Professor Reza Malek-Madani for his effort to run this opinion piece.

## References

- [1] Richard Bellman, *Dynamic programming*, Princeton University Press, Princeton, N. J., 1957. MR0090477
- [2] Paul A. Dirac, *Quantum mechanics of many-electron systems*, Proc. Roy. Soc. London Ser. A **123** (1929), no. 792.

- [3] Weinan E, Chao Ma, and Lei Wu, *Machine learning from a continuous viewpoint, I*, Sci. China Math. **63** (2020), no. 11, 2233–2266, DOI 10.1007/s11425-020-1773-8. MR4170870
- [4] Weinan E, Jiequn Han, and Linfeng Zhang, *Machine learning assisted modeling*, to appear in Physics Today.
- [5] Jiequn Han and Weinan E, *Deep learning approximation of stochastic control problems*, NIPS Workshop on Deep Reinforcement Learning, 2016. <https://arxiv.org/pdf/1611.07422.pdf>
- [6] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, *Deep learning*, Nature **521** (2015), 436–444.
- [7] Tenavi Nakamura-Zimmerer, Qi Gong, and Wei Kang, *Adaptive Deep learning for high dimensional Hamilton-Jacobi-Bellman equations*, 2019. <https://arxiv.org/pdf/1907.05317.pdf>



Weinan E

**Credits**

Photo of Weinan E is courtesy of Yiming Fu.

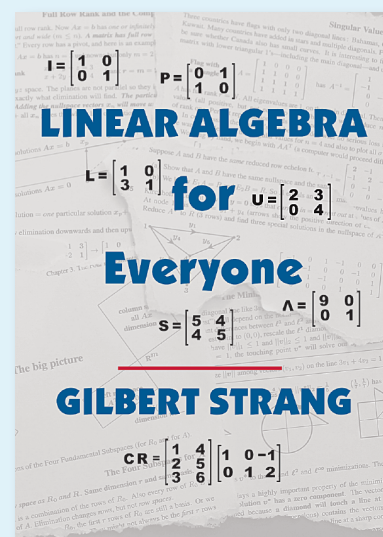
## Linear Algebra for Everyone

Gilbert Strang, *Massachusetts Institute of Technology, Cambridge, MA*

This book by Gilbert Strang, author of the widely-used *Introduction to Linear Algebra*, presents a “new start” in teaching and learning linear algebra, a subject that is increasingly important in all quantitative disciplines. Working with small matrices of integers, students can see dependent columns and the rank of  $A$  and its column space. This is a big step toward the four fundamental subspaces — a widespread success in teaching linear algebra. The new start makes linear algebra more approachable to students with a wide range of backgrounds. The text features many illustrative examples and exercises, and video lectures to accompany the book are available via MIT OpenCourseWare. The author’s site contains additional sample material and an instructor’s manual.

A publication of Wellesley-Cambridge Press. Distributed within the Americas by the American Mathematical Society.

**The Gilbert Strang Series**, Volume 4; 2020; 356 pages; Hardcover; ISBN: 978-1-7331466-3-0; List US\$85; AMS members US\$68; Order code STRANG/4



To see other books published by Gilbert Strang visit

[bookstore.ams.org/STRANG](https://bookstore.ams.org/STRANG)