Chapter 2

Fundamentals of Modeling

2.1 Models in Science and Engineering

Science is an endeavor to try to understand the world around us by discovering fundamental laws that describe how it works. Such laws include Newton's law of motion, the ideal gas law, Ohm's law in electrical circuits, the conservation law of energy, and so on, some of which you may have learned already.

A typical cycle of scientific effort by which scientists discover these fundamental laws may look something like this:

1. Observe nature.

2. Develop a hypothesis that could explain your observations.

3. From your hypothesis, make some predictions that are testable through an experiment.

4. Carry out the experiment to see if your predictions are actually true.

   - Yes $\rightarrow$ Your hypothesis is proven, congratulations. Uncork a champagne bottle and publish a paper.
   - No $\rightarrow$ Your hypothesis was wrong, unfortunately. Go back to the lab or the field, get more data, and develop another hypothesis.

Many people think this is how science works. But there is at least one thing that is not quite right in the list above. What is it? Can you figure it out?

As some of you may know already, the problem exists in the last part, i.e., when the experiment produced a result that matched your predictions. Let's do some logic to better understand what the problem really is. Assume that you observed a phenomenon P in nature and came up with a hypothesis H that can explain P. This means that a logical statement $H \rightarrow P$ is always true (because you chose H that way). To prove H, you also derived a prediction Q from H, i.e., another logical statement $H \rightarrow Q$ is always true, too. Then you conduct experiments to see if Q can be actually observed. What if Q is actually observed? Or, what if "$\mathrm{not}\ Q$" is observed instead?

If "$\mathrm{not}\ Q$" is observed, things are easy. Logically speaking, $(H \rightarrow Q)$ is equivalent to $(\mathrm{not}\ Q \rightarrow \mathrm{not}\ H)$ because they are *contrapositions* of each other, i.e., logically identical statements that can be converted from one to another by negating both the condition and the consequence and then flipping their order. This means that, if $\mathrm{not}\ Q$ is true, then it logically proves that $\mathrm{not}\ H$ is also true, i.e., your hypothesis is wrong. This argument is clear, and there is no problem with it (aside from the fact that you will probably have to redo your hypothesis building and testing).

The real problem occurs when your experiment gives you the desired result, Q. Logically speaking, *"$(H \rightarrow Q)$ and Q" doesn't tell you anything about whether H is true or not!* There are many ways your hypothesis could be wrong or insufficient even if the predicted outcome was obtained in the experiment. For example, maybe another alternative hypothesis R could be the right one ($R \rightarrow P$, $R \rightarrow Q$), or maybe H would need an additional condition K to predict P and Q (H and $K \rightarrow P$, H and $K \rightarrow Q$) but you were not aware of the existence of K.

Let me give you a concrete example. One morning, you looked outside and found that your lawn was wet (observation P). You hypothesized that it must have rained while you were asleep (hypothesis H), which perfectly explains your observation ($H \rightarrow P$). Then you predicted that, if it rained overnight, the driveway next door must also be wet (prediction Q that satisfies $H \rightarrow Q$). You went out to look and, indeed, it was also wet (if not, H would be clearly wrong). Now, think about whether this new observation really proves your hypothesis that it rained overnight. If you think critically, you should be able to come up with other scenarios in which both your lawn and the driveway next door could be wet without having a rainy night. Maybe the humidity in the air was unusually high, so the condensation in the early morning made the ground wet everywhere. Or maybe a fire hydrant by the street got hit by a car early that morning and it burst open, wetting the nearby area. There could be many other potential explanations for your observation.

In sum, obtaining supportive evidence from experiments doesn't prove your hypothesis in a logical sense. It only means that you have failed to disprove your hypothesis. However, many people still believe that science can prove things in an absolute way. *It* *can't.* There is no logical way for us to reach the ground truth of nature[1].

This means that all the "laws of nature," including those listed previously, are no more than well-tested hypotheses at best. Scientists have repeatedly failed to disprove them, so we give them more credibility than we do to other hypotheses. But there is absolutely no guarantee of their universal, permanent correctness. There is always room for other alternative theories to better explain nature.

In this sense, all science can do is just build *models* of nature. All of the laws of nature mentioned earlier are also models, not scientific facts, strictly speaking. This is something every single person working on scientific research should always keep in mind.

I have used the word "model" many times already in this book without giving it a definition. So here is an informal definition:

> A *model* is a simplified representation of a system. It can be conceptual, verbal, diagrammatic, physical, or formal (mathematical).

As a cognitive entity interacting with the external world, you are always creating a model of something in your mind. For example, at this very moment as you are reading this textbook, you are probably creating a model of what is written in this book. *Modeling* is a fundamental part of our daily cognition and decision making; it is not limited only to science.

With this understanding of models in mind, we can say that science is an endless effort to create models of nature, because, after all, modeling is the one and only rational approach to the unreachable reality. And similarly, engineering is an endless effort to control or influence nature to make something desirable happen, by creating and controlling its models. Therefore, modeling occupies the most essential part in any endeavor in science and engineering.

> *Exercise 2.1* In the "wet lawn" scenario discussed above, come up with a few more alternative hypotheses that could explain both the wet lawn and the wet driveway without assuming that it rained. Then think of ways to find out which hypothesis is most likely to be the real cause.

[1] This fact is deeply related to the impossibility of general system identification, including the identification of computational processes.

Exercise 2.2 Name a couple of scientific models that are extensively used in today's scientific/engineering fields. Then investigate the following:

- How were they developed?
- What made them more useful than earlier models?
- How could they possibly be wrong?

2.2 How to Create a Model

There are a number of approaches for scientific model building. My favorite way of classifying various kinds of modeling approaches is to put them into the following two major families:

Descriptive modeling In this approach, researchers try to specify the actual state of a system at a given time point (or at multiple time points) in a descriptive manner. Taking a picture, creating a miniature (this is literally a "model" in the usual sense of the word), and writing a biography of someone, all belong to this family of modeling effort. This can also be done using quantitative methods (e.g., equations, statistics, computational algorithms), such as regression analysis and pattern recognition. They all try to capture "what the system looks like."

Rule-based modeling In this approach, researchers try to come up with dynamical rules that can explain the observed behavior of a system. This allows researchers to make predictions of its possible (e.g., future) states. Dynamical equations, theories, and first principles, which describe how the system will change and evolve over time, all belong to this family of modeling effort. This is usually done using quantitative methods, but it can also be achieved at conceptual levels as well (e.g., Charles Darwin's evolutionary theory). They all try to capture "how the system will behave."

Both modeling approaches are equally important in science and engineering. For example, observation of planetary movement using telescopes in the early 17th century generated a lot of descriptive information about how they actually moved. This information was already a model of nature because it constituted a simplified representation of reality.

In the meantime, Newton derived the law of motion to make sense out of observational information, which was a rule-based modeling approach that allowed people to make predictions about how the planets would/could move in the future or in a hypothetical scenario. In other words, descriptive modeling is a process in which descriptions of a system are produced and accumulated, while rule-based modeling is a process in which underlying dynamical explanations are built for those descriptions. These two approaches take turns and form a single cycle of the scientific modeling effort.

In this textbook, we will focus on the latter, the rule-based modeling approach. This is because rule-based modeling plays a particularly important role in complex systems science. More specifically, developing a rule-based model at microscopic scales and studying its macroscopic behaviors through computer simulation and/or mathematical analysis is almost a necessity to understand emergence and self-organization of complex systems. We will discuss how to develop rule-based models and what the challenges are throughout this textbook.

A typical cycle of rule-based modeling effort goes through the following steps (which are similar to the cycle of scientific discoveries we discussed above):

1. Observe the system of your interest.
2. Reflect on the possible rules that might cause the system's characteristics that were seen in the observation.
3. Derive predictions from those rules and compare them with reality.
4. Repeat the above steps to modify the rules until you are satisfied with the model (or you run out of time or funding).

This seems okay, and it doesn't contain the logical problem of "proving a hypothesis" that we had before, because I loosened the termination criteria to be your satisfaction as a researcher. However, there is still one particular step that is fundamentally difficult. Which step do you think it is?

Of course, each of the four steps has its own unique challenges, but as an educator who has been teaching complex systems modeling for many years, I find that the second step (*"Reflect on possible rules that might cause the system's characteristics seen in the observation."*) is particularly challenging to modelers. This is because this step is so deeply interwoven with the modeler's knowledge, experience, and everyday cognitive processes. It is based on who you are, what you know, and how you see the world—it is, ultimately, a personal thinking process, which is very difficult to teach or to learn in a structured way.

Let me give you some examples to illustrate my point. The following figure shows an observation of a system over time. Can you create a mathematical model of this observation?

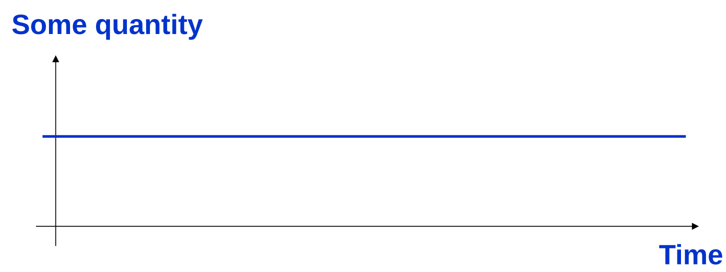

Figure 2.1: Observation example 1.

This one should be quite easy, because the observed data show that nothing changed over time. The description "no change" is already a valid model written in English, but if you prefer writing it in mathematical form, you may want to write it as

$$x(t) = C \tag{2.1}$$

or, if you use a differential equation,

$$\frac{dx}{dt} = 0. \tag{2.2}$$

Coming up with these models is a no brainer, because we have seen this kind of behavior many times in our daily lives.

Here is another example. Can you create a mathematical model of this observation?

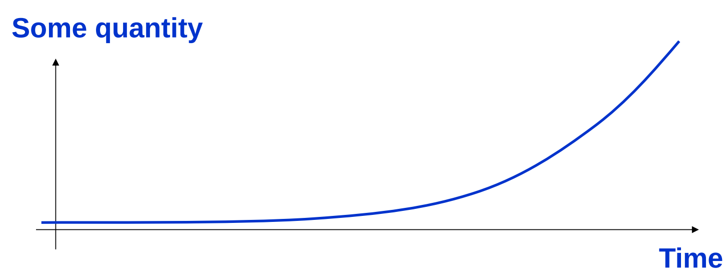

Figure 2.2: Observation example 2.

Now we see some changes. It seems the quantity monotonically increased over time. Then your brain must be searching your past memories for a pattern that looks like this curve you are looking at, and you may already have come up with a phrase like *"exponential growth,"* or more mathematically, equations like

$$x(t) = ae^{bt} \tag{2.3}$$

or

$$\frac{dx}{dt} = bx. \tag{2.4}$$

This may be easy or hard, depending on how much knowledge you have about such exponential growth models.

In the meantime, if you show the same curve to middle school students, they may proudly say that this must be the right half of a flattened parabola that they just learned about last week. And there is nothing fundamentally wrong with that idea. It could be a right half of a parabola, indeed. We never know for sure until we see what the entire curve looks like for $-\infty < t < \infty$.

Let's move on to a more difficult example. Create a mathematical model of the following observation.

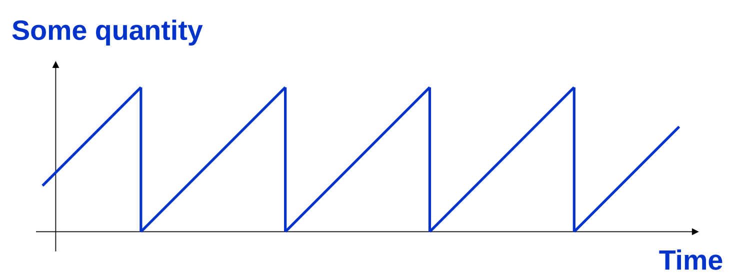

Figure 2.3: Observation example 3.

It is getting harder. Unless you have seen this kind of dynamic behavior before, you will have a hard time coming up with any concise explanation of it. An engineer might say, "This is a sawtooth wave," or a computer scientist might say, "This is time mod something." Or, an even more brilliant answer could come from an elementary school kid, saying, "This must be months in a calendar!" (which is equivalent to what the computer scientist said, by the way). In either case, people tend to map a new observation to something they already know in their mind when they create a model.

This last example is the toughest. Create a mathematical model of the following observation.

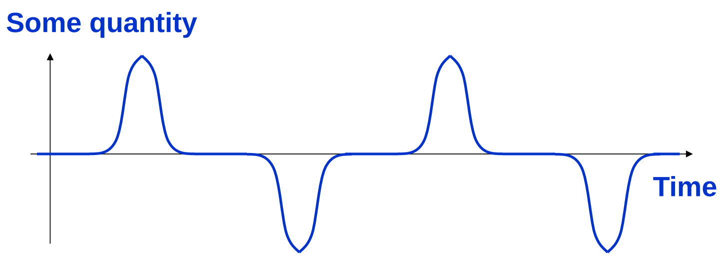

Figure 2.4: Observation example 4.

Did you come up with any ideas? I have seen only a few people who were able to make reasonable models of this observation throughout my career. The reason why this example is so hard to model is because we don't see this kind of behavior often in our lives. We are simply not experienced with it. We don't have a good mental template to use to capture the essence of this pattern[2].

I hope that these examples have made my point clear by now. Coming up with a model is inherently a personal process, which depends on your own knowledge, experience, and worldview. There is no single algorithm or procedure you can follow to develop a good model. The modeling process is a full-scale interaction between the external world and your whole, intellectual self. To become a good modeler, you will need to gain diverse knowledge and experience and develop rich worldviews. This is why I said it would be very difficult to be taught.

> *Exercise 2.3* Create a few different models for each of the examples shown above. Discuss how those models differ from each other, and what you should do to determine which model is more appropriate as an explanation of the observed behavior.

[2]For those who are curious—this kind of curve could be generated by raising a sine or cosine function of time to an odd number (e.g., $\sin^3(t)$, $\cos^5(t)$), but I am not sure if knowing this would ever help you in your future career.

2.3 Modeling Complex Systems

The challenge in developing a model becomes particularly tough when it comes to the modeling of complex systems, because their unique properties (networks, nonlinearity, emergence, self-organization, etc.) are not what we are familiar with. We usually think about things on a single scale in a step-by-step, linear chain of reasoning, in which causes and effects are clearly distinguished and discussed sequentially. But this approach is not suitable for understanding complex systems where a massive amount of components are interacting with each other interdependently to generate patterns over a broad range of scales. Therefore, the behavior of complex systems often appears to contradict our everyday experiences.

As illustrated in the examples above, it is extremely difficult for us to come up with a reasonable model when we are facing something unfamiliar. And it is even more difficult to come up with a reasonable set of microscopic rules that could explain the observed macroscopic properties of a system. Most of us are simply not experienced enough to make logical connections between things at multiple different scales.

How can we improve our abilities to model complex systems? The answer might be as simple as this: We need to become experienced and familiar with various dynamics of complex systems to become a good modeler of them. How can we become experienced? This is a tricky question, but thanks to the availability of the computers around us, *computational modeling* and *simulation* is becoming a reasonable, practical method for this purpose. You can construct your own model with full details of microscopic rules coded into your computer, and then let it actually show the macroscopic behavior arising from those rules. Such computational modeling and simulation is a very powerful tool that allows you to gain interactive, intuitive (simulated) experiences of various possible dynamics that help you make mental connections between micro- and macroscopic scales. I would say there are virtually no better tools available for studying the dynamics of complex systems in general.

There are a number of pre-built tools available for complex systems modeling and simulation, including NetLogo [13], Repast [14], MASON [15], Golly [16], and so on. You could also build your own model by using general-purpose computer programming languages, including C, C++, Java, Python, R, Mathematica, MATLAB, etc. In this textbook, we choose *Python* as our modeling tool, specifically Python 2.7, and use *PyCX* [17] to build interactive dynamic simulation models[3]. Python is free and widely used in scien-

[3]For those who are new to Python programming, see Python's online tutorial at `https://docs.python.org/2/tutorial/index.html`. Several pre-packaged Python distributions are available for free, such as Anaconda (available from `http://continuum.io/downloads`) and Enthought Canopy (available from

tific computing as well as in the information technology industries. More details of the rationale for this choice can be found in [17].

When you create a model of a complex system, you typically need to think about the following:

1. *What are the key questions you want to address?*

2. *To answer those key questions, at what scale should you describe the behaviors of the system's components?* These components will be the "microscopic" components of the system, and you will define dynamical rules for their behaviors.

3. *How is the system structured?* This includes what those microscopic components are, and how they will be interacting with each other.

4. *What are the possible states of the system?* This means describing what kind of dynamical states each component can take.

5. *How does the state of the system change over time?* This includes defining the dynamical rules by which the components' states will change over time via their mutual interaction, as well as defining how the interactions among the components will change over time.

Figuring out the "right" choices for these questions is by no means a trivial task. You will likely need to loop through these questions several times until your model successfully produces behaviors that mimic key aspects of the system you are trying to model. We will practice many examples of these steps throughout this textbook.

> *Exercise 2.4* Create a schematic model of some real-world system of your choice that is made of many interacting components. Which scale do you choose to describe the microscopic components? What are those components? What states can they take? How are those components connected? How do their states change over time? After answering all of these questions, make a mental prediction about what kind of macroscopic behaviors would arise if you ran a computational simulation of your model.

`https://enthought.com/products/canopy/`). A recommended environment is Anaconda's Python code editor named "Spyder."

2.4 What Are Good Models?

You can create various kinds of models for a system, but useful ones have several important properties. Here is a very brief summary of what a good model should look like:

A good model is simple, valid, and robust.

Simplicity of a model is really the key essence of what modeling is all about. The main reason why we want to build a model is that we want to have a shorter, simpler description of reality. As the famous principle of *Occam's razor* says, if you have two models with equal predictive power, you should choose the simpler one. This is not a theorem or any logically proven fact, but it is a commonly accepted practice in science. Parsimony is good because it is economical (e.g., we can store more models within the limited capacity of our brain if they are simpler) and also insightful (e.g., we may find useful patterns or applications in the models if they are simple). If you can eliminate any parameters, variables, or assumptions from your model without losing its characteristic behavior, you should.

Validity of a model is how closely the model's prediction agrees with the observed reality. This is of utmost importance from a practical viewpoint. If your model's prediction doesn't reasonably match the observation, the model is not representing reality and is probably useless. It is also very important to check the validity of not only the predictions of the model but also the assumptions it uses, i.e., whether each of the assumptions used in your model makes sense *at its face value*, in view of the existing knowledge as well as our common sense. Sometimes this *"face validity"* is more important in complex systems modeling, because there are many situations where we simply can't conduct a quantitative comparison between the model prediction and the observational data. Even if this is the case, you should at least check the face validity of your model assumptions based on your understanding about the system and/or the phenomena.

Note that there is often a trade-off between trying to achieve simplicity and validity of a model. If you increase the model complexity, you may be able to achieve a better fit to the observed data, but the model's simplicity is lost and you also have the risk of *overfitting*—that is, the model prediction may become adjusted too closely to a specific observation at the cost of generalizability to other cases. You need to strike the right balance between those two criteria.

Finally, *robustness* of a model is how insensitive the model's prediction is to minor variations of model assumptions and/or parameter settings. This is important because there are always errors when we create assumptions about, or measure parameter values from, the real world. If the prediction made by your model is sensitive to their minor variations, then the conclusion derived from it is probably not reliable. But if your model is robust, the conclusion will hold under minor variations of model assumptions and parameters, therefore it will more likely apply to reality, and we can put more trust in it.

> *Exercise 2.5* Humanity has created a number of models of the solar system in its history. Some of them are summarized below:
>
> - Ptolemy's geocentric model (which assumes that the Sun and other planets are revolving around the Earth)
> - Copernicus' heliocentric model (which assumes that the Earth and other planets are revolving around the Sun in concentric circular orbits)
> - Kepler's laws of planetary motion (which assumes that the Earth and other planets are revolving in elliptic orbits, at one of whose foci is the Sun, and that the area swept by a line connecting a planet and the Sun during a unit time period is always the same)
> - Newton's law of gravity (which assumes that a gravitational force between two objects is proportional to their masses and inversely proportional to their distance squared)
>
> Investigate these models, and compare them in terms of simplicity, validity and robustness.

2.5 A Historical Perspective

As the final section in this chapter, I would like to present some historical perspective of how people have been developing modeling methodologies over time, especially those for complex systems (Fig. 2.5). Humans have been creating descriptive models (diagrams, pictures, physical models, texts, etc.) and some conceptual rule-based models since ancient times. More quantitative modeling approaches arose as more advanced mathematical tools became available. In the descriptive modeling family, descriptive statistics is among such quantitative modeling approaches. In the rule-based modeling family, dynamical equations (e.g., differential equations, difference equations) began to be used to quantitatively formulate theories that had remained at conceptual levels before.

During the second half of the 20th century, computational tools became available to researchers, which opened up a whole new area of *computational modeling* approaches

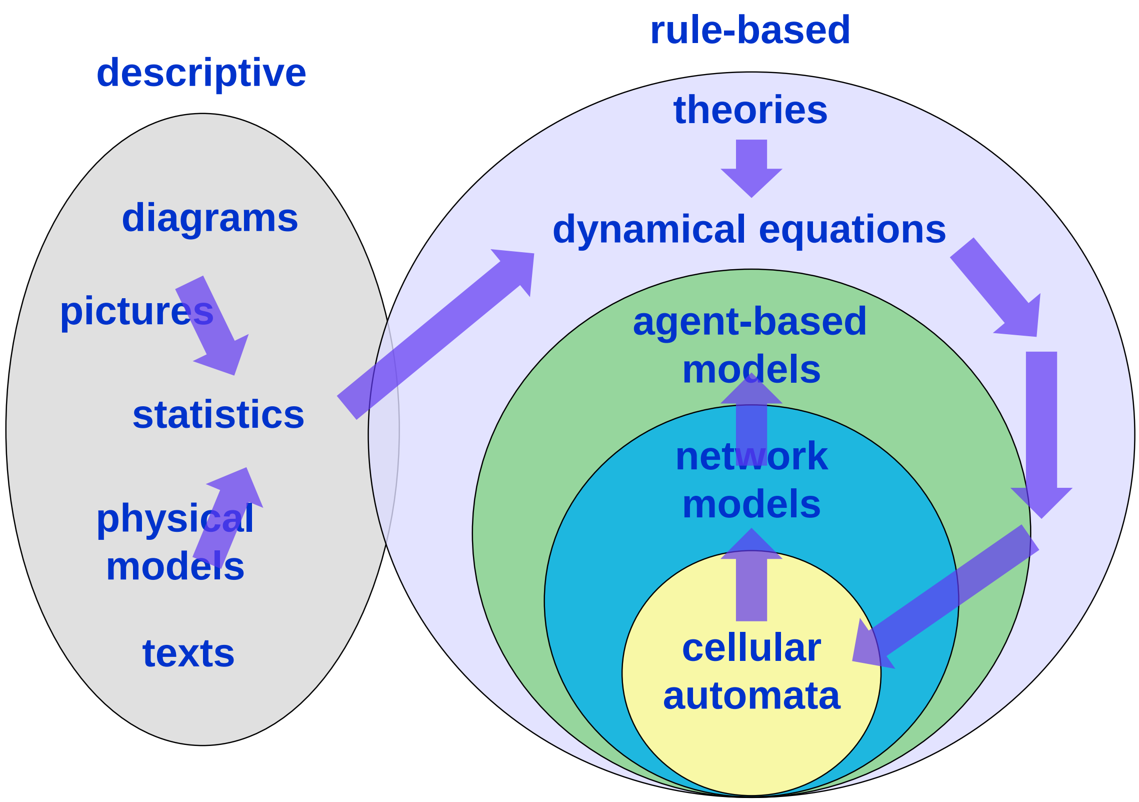

Figure 2.5: Schematic illustration of how modeling methodologies have developed historically.

for complex systems modeling. The first of this kind was *cellular automata*, a massive number of identical finite-state machines that are arranged in a regular grid structure and update their states dynamically according to their own and their neighbors' states. Cellular automata were developed by John von Neumann and Stanisław Ulam in the 1940s, initially as a theoretical medium to implement self-reproducing machines [11], but later they became a very popular modeling framework for simulating various interesting emergent behaviors and also for more serious scientific modeling of spatio-temporal dynamics [18]. Cellular automata are a special case of dynamical networks whose topologies are limited to regular grids and whose nodes are usually assumed to be homogeneous and identical.

Dynamical networks formed the next wave of complex systems modeling in the 1970s and 1980s. Their inspiration came from *artificial neural network* research by Warren McCulloch and Walter Pitts [19] as well as by John Hopfield [20, 21], and also from theoretical *gene regulatory network* research by Stuart Kauffman [22]. In this modeling framework, the topologies of systems are no longer constrained to regular grids, and the components and their connections can be heterogeneous with different rules and weights. Therefore, dynamical networks include cellular automata as a special case within them. Dynamical networks have recently merged with another thread of research on topological analysis that originated in graph theory, statistical physics, social sciences, and computational science, to form a new interdisciplinary field of *network science* [23, 24, 25].

Finally, further generalization was achieved by removing the requirement of explicit network topologies from the models, which is now called *agent-based modeling* (ABM). In ABM, the only requirement is that the system is made of multiple discrete "agents" that interact with each other (and possibly with the environment), whether they are structured into a network or not. Therefore ABM includes network models and cellular automata as its special cases. The use of ABM became gradually popular during the 1980s, 1990s, and 2000s. One of the primary driving forces for it was the application of complex systems modeling to ecological, social, economic, and political processes, in fields like game theory and microeconomics. The surge of *genetic algorithms* and other population-based search/optimization algorithms in computer science also took place at about the same time, which also had synergistic effects on the rise of ABM.

I must be clear that the historical overview presented above is my own personal view, and it hasn't been rigorously evaluated or validated by any science historians (therefore this may not be a valid model!). But I hope that this perspective is useful in putting various modeling frameworks into a unified, chronological picture. The following chapters of this textbook roughly follow the historical path of the models illustrated in this perspective.

Exercise 2.6 Do a quick online literature search to find a few scientific articles that develop or use mathematical/computational models. Read the articles to learn more about their models, and map them to the appropriate locations in Fig. 2.5.