1

Principles of Mathematical Modeling

We begin this introduction to mathematical modeling and simulation with an explanation of basic concepts and ideas, which includes definitions of terms such as *system, model, simulation, mathematical model,* reflections on the objectives of mathematical modeling and simulation, on characteristics of ''good'' mathematical models, and a classification of mathematical models. You may skip this chapter at first reading if you are just interested in a hands-on application of specific methods explained in the later chapters of the book, such as regression or neural network methods (Chapter 2) or differential equations (DEs) (in Chapters 3 and 4). Any professional in this field, however, should of course know about the principles of mathematical modeling and simulation. It was emphasized in the preface that everybody uses mathematical models – ''even those of us who are not aware of doing so''. You will agree that it is a good idea to have an idea of what one is doing. . .

Our starting point is the complexity of the problems treated in science and engineering. As will be explained in Section 1.1, the difficulty of problems treated in science and engineering typically originates from the complexity of the systems under consideration, and models provide an adequate tool to break up this complexity and make a problem tractable. After giving general definitions of the terms *system, model,* and *simulation* in Section 1.2, we move on toward mathematical models in Section 1.3, where it is explained that mathematics is *the* natural modeling language in science and engineering. Mathematical models themselves are defined in Section 1.4, followed by a number of example applications and definitions in Sections 1.5 and 1.6. This includes the important distinction between phenomenological and mechanistic models,which has been used as the main organization principle of this book (see Section 1.6.1 and Chapters 2–4). The chapter ends with a classification of mathematical models and Golomb's famous ''Don'ts of mathematical modeling'' in Sections 1.7 and 1.8.

1.1
A Complex World Needs Models

Generally speaking, engineers and scientists try to understand, develop, or optimize ''systems''. Here, ''system'' refers to the object of interest, which can be a part of

Mathematical Modeling and Simulation: Introduction for Scientists and Engineers. Kai Velten
Copyright © 2009 WILEY-VCH Verlag GmbH & Co. KGaA, Weinheim
ISBN: 978-3-527-40758-8

nature (such as a plant cell, an atom, a galaxy etc.) or an artificial technological system (see Definition 1.2.3 below). Principally, everybody deals with systems in his or her everyday life in a way similar to the approach of engineers or scientists. For example, consider the problem of a table which is unstable due to an uneven floor. This is a technical system and everybody knows what must be done to solve the problem: we just have to put suitable pieces of cardboard under the table legs. Each of us solves an abundant number of problems relating to simple technological systems of this kind during our lifetime. Beyond this, there is a great number of really difficult technical problems that can only be solved by engineers. Characteristic of these more demanding problems is a high complexity of the technical system. We would simply need no engineers if we did not have to deal with complex technical systems such as computer processors, engines, and so on. Similarly, we would not need scientists if processes such as the photosynthesis of plants could be understood as simply as an unstable table. The reason why we have scientists and engineers, virtually their right to exist, is the complexity of nature and the complexity of technological systems.

Note 1.1.1 (The complexity challenge) It is the genuine task of scientists and engineers to deal with complex systems, and to be effective in their work, they most notably need specific methods to deal with complexity.

The general strategy used by engineers or scientists to break up the complexity of their systems is the same strategy that we all use in our everyday life when we are dealing with complex systems: simplification. The idea is just this: if something is complex, make it simpler. Consider an everyday life problem related to a complex system: A car that refuses to start. In this situation, everyone knows that a look at the battery and fuel levels will solve the problem in most cases. Everyone will do this automatically, but to understand the problem solving strategy behind this, let us think of an alternative scenario. Assume someone is in this situation for the first time. Assume that ‘‘someone’’ was told how to drive a car, that he has used the car for some time, and now he is for the first time in a situation in which the car does not start. Of course, we also assume that there is no help for miles around! Then, looking under the hood for the first time, our ‘‘someone’’ will realize that the car, which seems simple as long as it works well, is quite a complex system. He will spend a lot of time until he will eventually solve the problem, even if we admit that our ‘‘someone’’ is an engineer. The reason why each of us will solve this problem much faster than this ‘‘someone’’ is of course the simple fact that this situation is not new to us. We have experienced this situation before, and from our previous experience we know what is to be done. Conceptually, one can say that we have a simplified picture of the car in our mind similar to Figure 1.1. In the moment when we realize that our car does not start, we do not think of the car as the complex system that it really is, that is, we do not think of this conglomerate of valves, pistons, and all the kind of stuff that can be found under the hood; rather, we have this simplified picture of the car in our mind. We know that this simplified

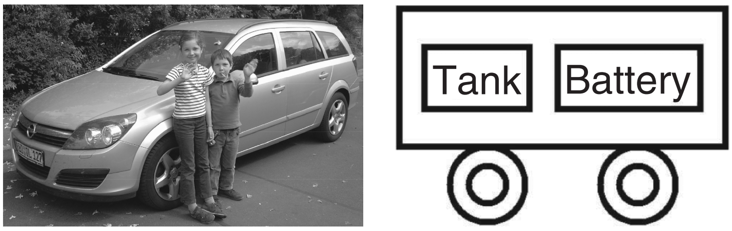


Fig. 1.1 Car as a real system and as a model.

picture is appropriate in this given situation, and it guides us to look at the battery and fuel levels and then to solve the problem within a short time.

This is exactly the strategy used by engineers or scientists when they deal with complex systems. When an engineer, for example, wants to reduce the fuel consumption of an engine, then he will not consider that engine in its entire complexity. Rather, he will use simplified descriptions of that engine, focusing on the machine parts that affect fuel consumption. Similarly, a scientist who wants to understand the process of photosynthesis will use simplified descriptions of a plant focusing on very specific processes within a single plant cell. Anyone who wants to understand complex systems or solve problems related to complex systems needs to apply appropriate simplified descriptions of the system under consideration. This means that anyone who is concerned with complex systems needs models, since simplified descriptions of a system are models of that system by definition.

Note 1.1.2 (Role of models) To break up the complexity of a system under consideration, engineers and scientists use simplified descriptions of that system (i.e. models).

1.2 Systems, Models, Simulations

In 1965, Minsky gave the following general definition of a model [1, 2]:

Definition 1.2.1 (Model) To an observer B, an object A* is a *model* of an object A to the extent that B can use A* to answer questions that interest him about A.

Note 1.2.1 (Formal definitions) Note that Definition 1.2.1 is a *formal definition* in the sense that it operates with terms such as *object* or *observer* that are not defined in a strict axiomatic sense similar to the terms used in the definitions of standard mathematical theory. The same remark applies to several other definitions in this book, including the definition of the term *mathematical model* in Section 1.4. Definitions of this kind are justified for practical reasons, since they allow us to talk about the formally defined terms in a concise way. An example is Definition 2.5.2 in Section 2.5.5, a concise formal definition of the term *overfitting*, which uses several of the previous formal definitions.

The application of Definition 1.2.1 to the car example is obvious – we just have to identify B with the car driver, A with the car itself, and A* with the simplified tank/battery description of the car in Figure 1.1.

1.2.1 Teleological Nature of Modeling and Simulation

An important aspect of the above definition is the fact that it includes the purpose of a model, namely, that the model helps us to answer questions and to solve problems. This is important because particularly beginners in the field of modeling tend to believe that a good model is one that mimics the part of reality that it pertains to as closely as possible. But as was explained in the previous section, modeling and simulation aims at simplification, rather than at a useless production of complex copies of a complex reality, and hence, the contrary is true:

Note 1.2.2 (The best model) The best model is the simplest model that still serves its purpose, that is, which is still complex enough to help us understand a system and to solve problems. Seen in terms of a simple model, the complexity of a complex system will no longer obstruct our view, and we will virtually be able to look through the complexity of the system at the heart of things.

The entire procedure of modeling and simulation is governed by its purpose of problem solving – otherwise it would be a mere l'art pour l'art. As [3] puts it, ''modeling and simulation is always goal-driven, that is, we should know the purpose of our potential model before we sit down to create it''. It is hence natural to define fundamental concepts such as the term *model* with a special emphasis on the purpose-oriented or *teleological nature of modeling and simulation*. (Note that teleology is a philosophical discipline dealing with aims and purposes, and the term *teleology* itself originates from the Greek word *telos*, which means end or purpose [4].) Similar teleological definitions of other fundamental terms, such as *system*, *simulation*, and *mathematical model* are given below.

1.2.2 Modeling and Simulation Scheme

Conceptually, the investigation of complex systems using models can be divided into the following steps:

Note 1.2.3 (Modeling and simulation scheme)

Definitions

- Definition of a problem that is to be solved or of a question that is to be answered
- Definition of a system, that is, a part of reality that pertains to this problem or question

Systems Analysis

- Identification of parts of the system that are relevant for the problem or question

Modeling

- Development of a model of the system based on the results of the systems analysis step

Simulation

- Application of the model to the problem or question
- Derivation of a strategy to solve the problem or answer the question

Validation

- Does the strategy derived in the simulation step solve the problem or answer the question for the real system?

The application of this scheme to the examples discussed above is obvious: in the *car example*, the problem is that the car does not start and the car itself is the system. This is the ''definitions'' step of the above scheme. The ''systems analysis'' step identifies the battery and fuels levels as the relevant parts of the system as explained above. Then, in the ''modeling'' step of the scheme, a model consisting of a battery and a tank such as in Figure 1.1 is developed. The application of this model to the given problem in the ''simulation'' step of the scheme then leads to the strategy ''check battery and fuel level''. This strategy can then be applied to the real car in the ''validation'' step. If it works, that is, if the car really starts after refilling its battery or tank, we say that the model is valid or validated. If not, we probably need a mechanic who will then look at other parts of the car, that is, who will apply more complex models of the car until the problem is solved.

In a real modeling and simulation project, the *systems analysis step* of the above scheme can be a very time-consuming step. It will usually involve a thorough evaluation of the literature. In many cases, the literature evaluation will show that similar investigations have been performed in the past, and one should of course try to profit from the experiences made by others that are described in the literature. Beyond this, the system analysis step usually involves a lot of discussions and meetings that bring together people from different disciplines who can answer your questions regarding the system. These discussion will usually show that new data are needed for a better understanding of the system and for the validation of the models in the validation step of the above scheme. Hence, the definition of an experimental program is also another typical part of the systems analysis step.

The *modeling step* will also involve the identification of appropriate software that can solve the equations of the mathematical model. In many cases, it will be possible to use standard software such as the software tools discussed in the next chapters. Beyond this, it may be necessary to write your own software in cases where the mathematical model involves nonstandard equations. An example of this case is the modeling of the press section of paper machines, which involves highly convection-dominated diffusion equations that cannot be treated by standard software with sufficient precision, and which hence need specifically tailored numerical software [5].

In the *validation step*, the model results will be compared with experimental data. These data may come from the literature, or from experiments that have been specifically designed to validate the model. Usually, a model is required to fit the data not only quantitatively, but also qualitatively in the sense that it reproduces the general shape of the data as closely as possible. See Section 3.2.3.4 for an example of a qualitative misfit between a model and data. But, of course, even a model that perfectly fits the data quantitatively and qualitatively may fail the validation step of the above scheme if it cannot be used to solve the problem that is to be solved, which is the most important criterion for a successful validation.

The modeling and simulation scheme (Note 1.2.3) focuses on the essential steps of modeling and simulation, giving a rather simplified picture of what really happens in a concrete modeling and simulation project. For different fields of application, you may find a number of more sophisticated descriptions of the modeling and simulation process in books such as [6–9]. An important thing that you should note is that a real modeling and simulation project will very rarely go straight through the steps of the above scheme; rather, there will be a lot of interaction between the individual steps of the scheme. For example, if the validation step fails, this will bring you back to one of the earlier steps in a *loop-like structure*: you may then improve your model formulation, reanalyze the system, or even redefine your problem formulation (if your original problem formulation turns out to be unrealistic).

Note 1.2.4 (Start with simple models!) To find the best model in the sense of Note 1.2.2, start with the simplest possible model and then generate a sequence of increasingly complex model formulations until the last model in the sequence passes the validation step.

1.2.3 Simulation

So far we have given a definition of the term *model* only. The above modeling and simulation schemes involve other terms, such as *system* and *simulation*, which we may view as being implicitly defined by their role in the above scheme. Can this be made more precise? In the literature, you will find a number of different definitions, for example of the term *simulation*. These differences can be explained by different interests of the authors. For example, in a book with a focus on the so-called *discrete event simulation* which emphasizes the development of a system over time, simulation is defined as ''the imitation of the operation of a real-world process or system over time'' [6]. In general terms, simulation can be defined as follows:

Definition 1.2.2 (Simulation) *Simulation* is the application of a model with the objective to derive strategies that help solve a problem or answer a question pertaining to a system.

Note that the term *simulation* originates from the Latin word ''simulare'', which means ''to pretend'': in a simulation, the model pretends to be the real system. A similar definition has been given by Fritzson [7] who defined simulation as ''an experiment performed on a model''. Beyond this, the above definition is a *teleological* (purpose-oriented) definition similar to Definition 1.2.1 above, that is, this definition again emphasizes the fact that simulation is always used to achieve some goal. Although Fritzson's definition is more general, the above definition reflects the real use of simulation in science and engineering more closely.

1.2.4 System

Regarding the term *system*, you will again find a number of different definitions in the literature, and again some of the differences between these definitions can be explained by the different interests of their authors. For example, [10] defines a system to be ''a collection of entities, for example, people or machines, that act and interact together toward the accomplishment of some logical end''. According to [11], a system is ''a collection of objects and relations between objects''. In the context of mathematical models, we believe it makes sense to think of a ''system'' in very general terms. Any kind of object can serve as a system here if we have a question relating to that object and if this question can be answered using mathematics. Our view of systems is similar to a definition that has been given by [12] (see also the discussion of this definition in [3]): '' A system is whatever is distinguished as a system.'' [3] gave another definition of a ''system'' very close to our view of systems here: ''A system is a potential source of data''. This definition emphasizes the fact that a system can be of scientific interest only if there is some communication between the system and the outside world, as it will be discussed below in Section 1.3.1. A definition that includes the teleological principle discussed above has been given by Fritzson [7] as follows:

Definition 1.2.3 (System) A *system* is an object or a collection of objects whose properties we want to study.

1.2.5 Conceptual and Physical Models

The model used in the car example is something that exists in our minds only. We can write it down on a paper in a few sentences and/or sketches, but it does not have any physical reality. Models of this kind are called *conceptual models* [11]. Conceptual models are used by each of us to solve everyday problems such as the car that refuses to start. As K.R. Popper puts it, ‘‘all life is problem solving’’, and conceptual models provide us with an important tool to solve our everyday problems [13]. They are also applied by engineers or scientists to simple problems or questions similar to the car example. If their problem or question is complex enough, however, they rely on experiments, and this leads us to other types of models. To see this, let us use the modeling and simulation scheme (Note 1.2.3) to describe a possible procedure followed by an engineer who wants to reduce the fuel consumption of an engine: In this case, the problem is the reduction of fuel consumption and the system is the engine. Assume that the systems analysis leads the engineer to the conclusion that the fuel injection pump needs to be optimized. Typically, the engineer will then create some experimental setting where he can study the details of the fuel injection process.

Such an experimental setting is then a model in the sense that it will typically be a very simplified version of that engine, that is, it will typically involve only a few parts of the engine that are closely connected with the fuel injection process. In contrast to a conceptual model, however, it is not only an idea in our mind but also a real part of the physical world, and this is why models of this kind are called *physical models* [11]. The engineer will then use the physical model of the fuel injection process to derive strategies – for example, a new construction of the fuel injection pump – to reduce the engine’s fuel consumption, which is the simulation step of the above modeling and simulation scheme. Afterwards, in the validation step of the scheme, the potential of these new constructions to reduce fuel consumption will be tested in the engine itself, that is, in the real system. Physical models are applied by scientists in a similar way. For example, let us think of a scientist who wants to understand the photosynthesis process in plants. Similar to an engineer, the scientist will set up a simplified experimental setting – which might be some container with a plant cell culture – in which he can easily observe and measure the important variables, such as CO_2, water, light, and so on. For the same reasons as above, anything like this is a physical model. As before, any conclusion drawn from such a physical model corresponds to the simulation step of the above scheme, and the conclusions need to be validated by data obtained from the real system, that is, data obtained from real plants in this case.

1.3
Mathematics as a Natural Modeling Language

1.3.1
Input–Output Systems

Any system that is investigated in science or engineering must be observable in the sense that it produces some kind of output that can be measured (a system that would not satisfy this minimum requirement would have to be treated by theologians rather than by scientists or engineers). Note that this observability condition can also be satisfied by systems where nothing can be measured directly, such as black holes, which produce measurable gravitational effects in their surroundings. Most systems investigated in engineering or science do also accept some kind of input data, which can then be studied in relation to the output of the system (Figure 1.2a). For example, a scientist who wants to understand photosynthesis will probably construct experiments where the carbohydrate production of a plant is measured at various levels of light, CO_2, water supply, and so on. In this case, the plant cell is the system; the light, CO_2, and water levels are the input quantities; and the measured carbohydrate production is the output quantity. Or, an engineer who wants to optimize a fuel injection pump will probably change the construction of that pump in various ways and then measure the fuel consumption resulting from these modified constructions. In this case, the fuel injection pump is the system, the construction parameters changed by the engineer are the input parameters and the resulting fuel consumption is the output quantity.

Note 1.3.1 (Input–output systems) Scientists or engineers investigate ''input–output systems'', which transform given input parameters into output parameters.

Note that there are of course situations where scientists are looking at the system itself and not at its input–output relations, for example when a botanist just wants

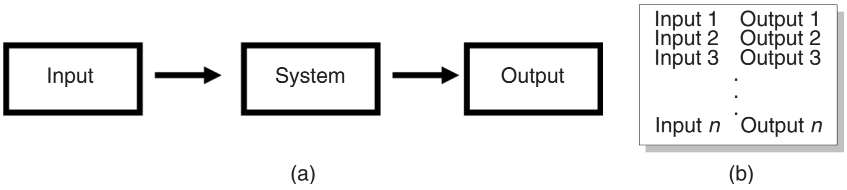


Fig. 1.2 (a) Communication of a system with the outside world. (b) General form of an experimental data set.

to describe and classify the anatomy of a newly discovered plant. Typically, however, such purely descriptive studies raise questions about the way in which the system works, and this is when input–output relations come into play. Engineers, on the other hand, are always concerned with input–output relations since they are concerned with technology. The Encyclopedia Britannica defines technology as ''the application of scientific knowledge to the practical aims of human life''. These ''practical aims'' will usually be expressible in terms of a system output, and the tuning of system input toward optimized system output is precisely what engineers typically do, and what is in fact the genuine task of engineering.

1.3.2 General Form of Experimental Data

The experimental procedure described above is used very generally in engineering and in the (empirical) sciences to understand, develop, or optimize systems. It is useful to think of it as a means to explore *black boxes*. At the beginning of an experimental study, the system under investigation is similar to such a ''black box'' in the sense that there is some uncertainty about the processes that happen inside the system when the input is transformed into the output. In an extreme case, the experimenter may know only that ''something'' happens inside the system which transforms input into output, that is, the system may be really a black box. Typically, however, the experimenter will have some hypotheses about the internal processes, which he wants to prove or disprove in the course of his study. That is, experimenters typically are concerned with systems as gray boxes which are located somewhere between black and white boxes (more details in Section 1.5).

Depending on the hypothesis that the experimenter wants to investigate, he confronts the system with appropriate input quantities, hoping that the outputs produced by the system will help prove or disprove his hypothesis. This is similar to a question-and-answer game: the experimenter poses questions to the system, which is the input, and the system answers to these questions in terms of measurable output quantities. The result is a data set of the general form shown in Figure 1.2b. In rare cases, particularly if one is concerned with very simple systems, the internal processes of the system may already be evident from the data set itself. Typically, however, this experimental question-and-answer game is similar to the questioning of an oracle: we know there is some information about the system in the data set, but it depends on the application of appropriate ideas and methods if one wants to uncover the information content of the data and, so to speak, shed some light into the black box.

1.3.3 Distinguished Role of Numerical Data

Now what is an appropriate method for the analysis of experimental datasets? To answer this question, it is important to note that in most cases experimental data are numbers and can be quantified. The input and output data of Figure 1.2b will typically consist of columns of numbers. Hence, it is natural to think of a system in mathematical terms. In fact, a system can be naturally seen as a mathematical function, which maps given input quantities x into output quantities $y = f(x)$ (Figure 1.2a). This means that if one wants to understand the internal mechanics of a system ''black box'', that is, if one wants to understand the processes inside the real system that transform input into output, a natural thing to do is to translate all these processes into mathematical operations. If this is done, one arrives at a simplified representation of the real system in mathematical terms. Now remember that a simplified description of a real system (along with a problem we want to solve) is a model by definition (Definition 1.2.1). The representation of a real system in mathematical terms is thus a mathematical model of that system.

Note 1.3.2 (Naturalness of mathematical models) Input–output systems usually generate numerical (or quantifiable) data that can be described naturally in mathematical terms.

This simple idea, that is, the mapping of the internal mechanics of real systems into mathematical operations, has proved to be extremely fruitful to the understanding, optimization, or development of systems in science and engineering. The tremendous success of this idea can only be explained by the naturalness of this approach – mathematical modeling is simply the best and most natural thing one can do if one is concerned with scientific or engineering problems. Looking back at Figure 1.2a, it is evident that mathematical structures emanate from the very heart of science and engineering. Anyone concerned with systems and their input–output relations is also concerned with mathematical problems – regardless of whether he likes it or not and regardless of whether he treats the system appropriately using mathematical models or not. The success of his work, however, depends very much on the appropriate use of mathematical models.

1.4 Definition of Mathematical Models

To understand mathematical models, let us start with a general definition. Many different definitions of mathematical models can be found in the literature. The differences between these definitions can usually be explained by the different scientific interests of their authors. For example, Bellomo and Preziosi [14] define a mathematical model to be a set of equations which can be used to compute the time-space evolution of a physical system. Although this definition suffices for the problems treated by Bellomo and Preziosi, it is obvious that it excludes a great number of mathematical models. For example, many economical or sociological problems cannot be treated in a time-space framework or based on equations only. Thus, a more general definition of mathematical models is needed if one wants to cover all kinds of mathematical models used in science and engineering. Let us start with the following attempt of a definition:

A mathematical model is a set of mathematical statements $M = \{\Sigma_1, \Sigma_2, \ldots, \Sigma_n\}$.

Certainly, this definition covers all kinds of mathematical models used in science and engineering as required. But there is a problem with this definition. For example, a simple mathematical statement such as $f(x) = e^x$ would be a mathematical model in the sense of this definition. In the sense of Minsky's definition of a model (Definition 1.2.1), however, such a statement is not a model as long as it lacks any connection with some system and with a question we have relating to that system. The above attempt of a definition is incomplete since it pertains to the word ''mathematical'' of ''mathematical model'' only, without any reference to purposes or goals. Following the philosophy of the teleological definitions of the terms *model*, *simulation*, and *system* in Section 1.2, let us define instead:

Definition 1.4.1 (Mathematical Model) A mathematical model is a triplet (S, Q, M) where S is a system, Q is a question relating to S, and M is a set of mathematical statements $M = \{\Sigma_1, \Sigma_2, \ldots, \Sigma_n\}$ which can be used to answer Q.

Note that this is again a formal definition in the sense of Note 1.2.1 in Section 1.2. Again, it is justified by the mere fact that it helps us to understand the nature of mathematical models, and that it allows us to talk about mathematical models in a concise way. A similar definition was given by Bender [15]: ''A mathematical model is an abstract, simplified, mathematical construct related to a part of reality and created for a particular purpose.'' Note that Definition 1.4.1 is not restricted to physical systems. It covers psychological models as well that may deal with essentially metaphysical quantities, such as thoughts, intentions, feelings, and so on. Even mathematics itself is covered by the above definition. Suppose, for example, that S is the set of natural numbers and our question Q relating to S is whether there are infinitely many prime numbers or not. Then, a set (S, Q, M) is a mathematical model in the sense of Definition 1.4.1 if M contains the statement ''There are infinitely many prime numbers'' along with other statements which prove this statement. In this sense, the entire mathematical theory can be viewed as a collection of mathematical models.

The notation (S, Q, M) in Definition 1.4.1 emphasizes the chronological order in which the constituents of a mathematical model usually appear. Typically, a system is given first, then there is a question regarding that system, and only then a mathematical model is developed. Each of the constituents of the triplet (S, Q, M) is an indispensable part of the whole. Regarding M, this is obvious, but S and Q are important as well. Without S, we would not be able to formulate a question Q; without a question Q, there would be virtually ''nothing to do'' for the mathematical model; and without S and Q, the remaining M would be no more than ''l'art pour l'art''. The formula $f(x) = e^x$, for example, is such a purely mathematical ''l'art pour l'art'' statement as long as we do not connect it with a system and a question. It becomes a mathematical model only when we define a system S and a question Q relating to it. For example, viewed as an expression of the exponential growth period of plants (Section 3.10.4), $f(x) = e^x$ is a mathematical model which can be used to answer questions regarding plant growth. One can say it is a genuine property of mathematical models to be more than ''l'art pour l'art'', and this is exactly the intention behind the notation (S, Q, M) in Definition 2.3.1. Note that the definition of mathematical models by Bellomo and Preziosi [14] discussed above appears as a special case of Definition 1.4.1 if we restrict S to physical systems, M to equations, and only allow questions Q which refer to the space-time evolution of S.

Note 1.4.1 (More than ''l'art pour l'art'') The system and the question relating to the system are indispensable parts of a mathematical model. It is a genuine property of mathematical models to be more than mathematical ''l'art pour l'art''.

Let us look at another famous example that shows the importance of Q. Suppose we want to predict the behavior of some mechanical system S. Then the appropriate mathematical model depends on the problem we want to solve, that is, on the question Q. If Q is asking for the behavior of S at moderate velocities, classical (Newtonian) mechanics can be used, that is, $M = \{$*equations of Newtonian mechanics*$\}$. If, on the other hand, Q is asking for the behavior of S at velocities close to the speed of light, then we have to set $M = \{$*equations of relativistic mechanics*$\}$ instead.

1.5 Examples and Some More Definitions

Generally speaking, one can say we are concerned with mathematical models in the sense of Definition 1.4.1 whenever we perform computations in our everyday life, or whenever we apply the mathematics we have learned in schools and universities. Since everybody computes in his everyday life, everybody uses mathematical models, and this is why it was valid to say that ''everyone models and simulates'' in the preface of this book. Let us look at a few examples of mathematical models now, which will lead us to the definitions of some further important concepts.

Note 1.5.1 (Everyone models and simulates) Mathematical models in the sense of Definition 1.4.1 appear whenever we perform computations in our everyday life.

Suppose we want to know the *mean age of some group of people*. Then, we apply a mathematical model (S, Q, M) where S is that group of people, Q asks for their mean age, and M is the mean value formula $\overline{x} = \left(\sum_{i=1}^{n} x_i\right)/n$. Or, suppose we want to know the mass X of some substance in the cylindrical tank of Figure 1.3, given

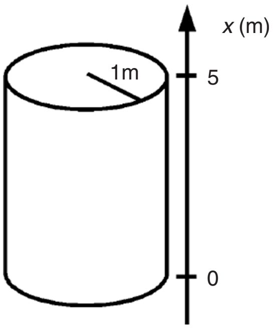


Fig. 1.3 Tank problem.

a constant concentration c of the substance in that tank. Then, a multiplication of the tank volume with c gives the mass X of the substance, that is,

$$X = 5\pi c \tag{1.1}$$

This means we apply a model (S, Q, M) where S is the tank, Q asks for the mass of the substance, and M is Equation 1.1. An example involving more than simple algebraic operations is obtained if we assume that the *concentration c in the tank* of Figure 1.3 depends on the height coordinate, x. In that case, Equation 1.1 turns into

$$X = \pi \cdot \int_0^5 c(x)\, dx \tag{1.2}$$

This involves an integral, that is, we have entered the realms of calculus now.

Note 1.5.2 (Notational convention) Variables such as X and c in Equation 1.1, which are used without further specification are always assumed to be real numbers, and functions such as $c(x)$ in Equation 1.2 are always assumed to be real functions with suitable ranges and domains of definition (such as $c : [0, 5] \to \mathbb{R}_+$ in the above example) unless otherwise stated.

In many mathematical models (S, Q, M) involving calculus, the question Q asks for the optimization of some quantity. Suppose for example we want to *minimize the material consumption* of a cylindrical tin having a volume of 1 l. In this case,

$$M = \{\pi r^2 h = 1, A = 2\pi r^2 + 2\pi rh \to \min\} \tag{1.3}$$

can be used to solve the problem. Denoting by r and h the radius and height of the tin, the first statement in Equation 1.3 expresses the fact that the tin volume is 1 l. The second statement requires the surface area of the tin to be minimal, which is equivalent to a minimization of the metal used to build the tin. The mathematical problem 1.3 can be solved if one inserts the first equation of (1.3) into the second equation of (1.3), which leads to

$$A(r) = 2\pi r^2 + \frac{2}{r} \to \min \tag{1.4}$$

This can then be treated using standard calculus ($A'(r) = 0$ etc.), and the optimal tin geometry obtained in this way is

$$r = \sqrt[3]{\frac{1}{2\pi}} \approx 0.54\ dm \tag{1.5}$$

$$h = \sqrt[3]{\frac{4}{\pi}} \approx 1.08\ dm \tag{1.6}$$

1.5.1
State Variables and System Parameters

Several general observations can be made referring to the examples in the last section. As discussed in Section 1.1 above, the main benefit of the modeling procedure lies in the fact that the complexity of the original system is reduced. This can be nicely seen in the last example. Of course, each of us knows that a cylindrical tin can be described very easily based on its radius r and its height h. This means everyone of us automatically applies the correct mathematical model, and hence, – similar to the car problem discussed in Section 1.1 – everybody automatically believes that the system in the tin problem is a simple thing. But if we do not apply this model to the tin, it becomes a complex system. Imagine a *Martian* or some other extraterrestrial being who never saw a cylinder before. Suppose we would say to this Martian: ''Look, here you have some sheets of metal and a sample tin filled with water. Make a tin of the same shape which can hold that amount of water, and use as little metal as possible.'' Then this Martian will – at least initially – see the original complexity of the problem. If he is smart, which we assume, he will note that infinitely many possible tin geometries are involved here. He will realize that an infinite set of (x, y)-coordinates would be required to describe the sample tin based on its set of coordinates. He will realize that infinitely many measurements, or, equivalently, algebraic operations would be required to obtain the material consumption based on the surface area of the sample tin (assuming that he did not learn about transcendental numbers such as π in his Martian school . . .).

From this original (''Martian'') point of view we thus see that the system S of the tin example is quite complex, in fact an infinite-dimensional system. And we see the power of the mathematical modeling procedure which reduces those infinite dimensions to only two, since the mathematical solution of the above problem involves only two parameters: r and h (or, equivalently, r and A). Originally, the system ''tin'' in the above example is an infinite-dimensional thing not only with respect to its set of coordinates or the other aspects mentioned above, but also with respect to many other aspects which have been neglected in the mathematical model since they are unimportant for the solution of the problem, for example the thickness of the metal sheets, or its material, color, hardness, roughness and so on. All the information which was contained in the original system $S =$ ''tin'' is reduced to a description of the system as a mere $S_\mathrm{r} = \{r, h\}$ in terms of the mathematical model. Here, we have used the notation S_r to indicate that S_r is not the original system which we denote S, but rather the description of S in terms of the mathematical model, which we call the ''reduced system''. The index ''r'' indicates that the information content of the original system S is reduced as we go from S to S_r.

Note 1.5.3 (A main benefit) The reduction of the information content of complex systems in terms of *reduced systems* (Definition 1.5.2) is one of the main benefits of mathematical models.

A formal definition of the reduced system S_r can be given in two steps as follows:

Definition 1.5.1 (State variables) Let (S, Q, M) be a mathematical model. Mathematical quantities $s_1, s_2, \ldots, s_n$ which describe the state of the system S in terms of M and which are required to answer Q are called the *state variables* of (S, Q, M).

Definition 1.5.2 (Reduced system and system parameters) Let $s_1, s_2, \ldots, s_n$ be the state variables of a mathematical model (S, Q, M). Let $p_1, p_2, \ldots, p_m$ be mathematical quantities (numbers, variables, functions) which describe properties of the system S in terms of M, and which are needed to compute the state variables. Then $S_\mathrm{r} = \{p_1, p_2, \ldots, p_m\}$ is the *reduced system* and $p_1, p_2, \ldots, p_m$ are the *system parameters* of (S, Q, M).

This means that the state variables describe the system properties we are really interested in, while the system parameters describe system properties needed to obtain the state variables mathematically. Although we finally need the state variables to answer Q, the information needed to answer Q is already in the system parameters, that is, in the reduced system S_r. Using S_r, this information is expressed in terms of the state variables by means of mathematical operations, and this is then the final basis to answer Q. For example, in the *tank problem* above we were interested in the mass of the substance; hence, in this example we have one state variable, that is, $n = 1$ and $s_1 = X$. To obtain s_1, we used the concentration c; hence, we have one system parameter in that example, that is, $m = 1$ and $p_1 = c$. The reduced system in this case is $S_\mathrm{r} = \{c\}$. By definition, the reduced system contains all information about the system which we need to get the state variable, that is, to answer Q. In the *tin example*, we needed the surface area of the tin to answer Q, that is, in that case we had again one state variable $s_1 = A$. On the other hand, two system parameters $p_1 = r$ and $p_2 = h$ were needed to obtain s_1, that is, in this case the reduced system is $S_\mathrm{r} = \{r, h\}$.

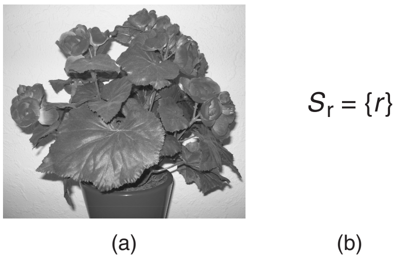


Fig. 1.4 (a) Potted plant. (b) The same potted plant written as a reduced system.

Let us look at another example. In Section 3.10.4 below, a *plant growth model* will be discussed which is intended to predict the time evolution of the overall biomass of a plant. To achieve this, none of the complex details of the system ‘‘plant’’ will be considered except for its growth rate. This means the complex system $S =$ ‘‘plant’’ is reduced to a single parameter in this model: the growth rate r of the plant. In the above notation, this means we have $S_r = \{r\}$ (Figure 1.4). It is not necessary to be a botanist to understand how dramatic this information reduction really is: everything except for the growth rate is neglected, including all kinds of macroscopic and microscopic substructures of the plant, its roots, its stem, its leaves as well as its cell structure, all the details of the processes that happen inside the cells, and so on. From the point of view of such a brutally simplified model, it makes no difference whether it is really concerned with the complex system ‘‘plant’’, or with some shapeless green pulp of biomass that might be obtained after sending the plant through a shredder, or even with entirely other systems, such as a bacteria culture or a balloon that is being inflated.

All that counts from the point of view of this model is that a growth rate can be assigned to the system under consideration. Naturally, botanists do not really like this brutal kind of models, which virtually send there beloved ones through a shredder. Anyone who presents such a model on a botanist’s conference should be prepared to hear a number of questions beginning with ‘‘Why does your model disregard . . . ’’. At this point we already know how to answer this kind of question: we know that according to Definition 1.4.1, a mathematical model is a triplet (S, Q, M) consisting of a system S, a question Q, and a set of mathematical statements M, and that the details of the system S that are represented in M depend on the question Q that is to be answered by the model. In this case, Q was asking for the time development of the plant biomass, and this can be sufficiently answered based on a model that represents the system $S =$ ‘‘plant’’ as $S_r = \{r\}$. Generally one can say that the reduced system of a well-formulated mathematical model will consist of no more than exactly those properties of the original system that are important to answer the question Q that is being investigated.

Note 1.5.4 (Importance of experiments) Typically, the properties (parameters) of the reduced system are those which need experimental characterization. In this way, the modeling procedure guides the experiments, and instead of making the experimenter superfluous (a frequent misunderstanding), it helps to avoid superfluous experiments.

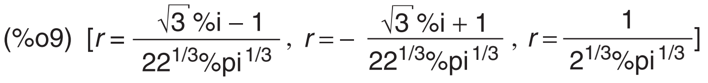
1.5.2
Using Computer Algebra Software

Let us make a few more observations relating to the "1 l tin" example above. The mathematical problem behind this example can be easily solved using software. For example, using the *computer algebra software Maxima*, the problem can be solved as follows:

```
1: A(r):=2 *%pi *r ˆ 2 +2/r;
2: define(A1(r),diff(A(r),r));
3: define(A2(r),diff(A1(r),r));
4: solve(A1(r) =0);                                   (1.7)
5: r:rhs(solve(A1(r)=0)[3]);
6: r,numer;
7: A2(r)>0,pred;
```

These are the essential commands in the *Maxima* program `Tin.mac` which you find in the book software. See Appendix A for a description of the book software and Appendix C for a description of how you can run `Tin.mac` within *Maxima*. In 1.7, the numbers `1:`, `2:`, and so on are not a part of the code, but just line numbers that we will use for referencing. Line 1 of the code defines the function $A(r)$ from Equation 1.4, which describes the material consumption that is to be minimized (note that `%pi` is the *Maxima* notation of π). As you know from calculus, you can minimize $A(r)$ by solving $A'(r) = 0$ [16, 17]. The solutions of this equations are the critical points, which can be relative maxima or minima depending on the sign of the second derivative A'' of A. Lines 2 and 3 of the above code define the first and second derivatives of $A(r)$ as the *Maxima* functions `A1(r)` and `A2(r)`. Line 4 solves $A'(r) = 0$ using *Maxima*'s `solve` command, which gives the result shown in Figure 1.5 if you are using *wxMaxima* (see Appendix C for details on *wxMaxima*).

As the figure shows, $A'(r) = 0$ gives three critical points. The first two critical points involve the imaginary number i (which is designated as "`%i`" within *Maxima*), so these are complex numbers which can be excluded here [17]. The third solution in Figure 1.5 is the solution that really solves the tin problem (compare Equation 1.5 above). Line 5 of Equation 1.7 stores this solution in the variable r, using "`[3]`" to address the third element in the list shown in Figure 1.5. Since



(%o9) $[r=\dfrac{\sqrt{3}\,\%i-1}{2 2^{1/3}\%pi^{1/3}},\ r=-\dfrac{\sqrt{3}\,\%i+1}{2 2^{1/3}\%pi^{1/3}},\ r=\dfrac{1}{2^{1/3}\%pi^{1/3}}]$

Fig. 1.5 Result of line 4 of Equation 1.7 in *wxMaxima*.

this element is an equation, `rhs` is then used to pick the right-hand side of this equation. *Maxima*'s `numer` command can be used as in line 6 of Equation 1.7 if you want to have the solution in a decimal numerical format. Finally, *Maxima*'s `pred` command can be used as in line 7 of Equation 1.7 to verify that the value of the second derivative is positive at the critical point that was stored in `r` (a necessary condition for that critical point to be a minimum [17]). In *Maxima*, line 7 gives "true", which means that the second derivative is indeed positive as required.

1.5.3
The Problem Solving Scheme

In this example – and similarly in many other cases – one can clearly distinguish between the formulation of a mathematical model on the one hand and the solution of the resulting mathematical problem on the other hand, which can be done with appropriate software. A number of examples will show this below. This means that it is not necessary to be a professional mathematician if one wants to work with mathematical models. Of course, it is useful to have mathematical expertise. Mathematical expertise is particularly important if one wants to solve more advanced problems, or if one wants to make sure that the results obtained with mathematical software are really solutions of the original problem and no numerical artifacts. As we will see below, the latter point is of particular importance in the solution of partial differential equations (PDEs). However, people with insufficient mathematical expertise may of course just ask a mathematician. Typically, mathematical modeling projects will have an interdisciplinary character. The important point that we should note here is the fact that the formulation of mathematical models can also be done by nonmathematicians. Above all, the people formulating the models should be experts regarding the system under consideration. This book is intended to provide particularly nonmathematicians with enough knowledge about the mathematical aspects of modeling such that they can deal at least with simple mathematical models on their own.

Note 1.5.5 (Role of software) Typically, the formulation of a mathematical model is clearly separated from the solution of the mathematical problems implied by the model. The latter ("the hard work") can be done by software in many cases. People working with mathematical models hence do not need to be professional mathematicians.

The tin example shows another important advantage of mathematical modeling. After the tin problem was formulated mathematically (Equation 1.4), the powerful and well-established mathematical methods of calculus became applicable. Using the appropriate software (see 1.7), the problem could then be solved with little effort. Without the mathematical model for this problem, on the other hand, an

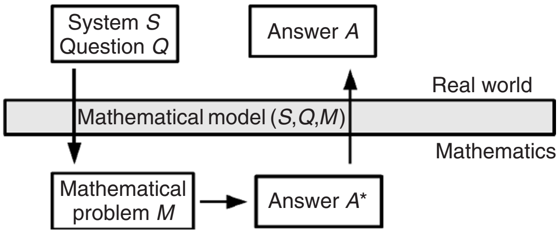


Fig. 1.6 Problem solving scheme.

experimental solution of this problem would have taken much more time. In a similar way, many other problems in science and engineering can be solved effectively using mathematics. From the point of view of science and engineering, mathematics can be seen as a big resource of powerful methods and instruments that can be used to solve problems, and it is the role of mathematical models to make these methods and instruments applicable to originally nonmathematical problems. Figure 1.6 visualizes this process. The starting point is a real-world system S together with a question Q relating to S. A mathematical model (S, Q, M) then opens up the way into the ''mathematical universe'', where the problem can be solved using powerful mathematical methods. This leads to a problem solution in mathematical terms (A^*), which is then translated into an answer A to the original question Q in the last step.

Note 1.5.6 (Mathematical models as door opener) Translating originally nonmathematical problems into the language of mathematics, mathematical models virtually serve as a door opener toward the ''mathematical universe'' where powerful mathematical methods become applicable to originally nonmathematical problems.

As the figure shows, the mathematical model virtually controls the ''problem solving traffic'' between the real and mathematical worlds, and hence, its natural position is located exactly at the borderline between these worlds. The role of mathematics in Figure 1.6 can be described like a subway train: since it would be a too long and hard way to go from the system S and question Q to the desired answer A in the real world, smart problem solvers go into the ''mathematical underground'', where powerful mathematical methods provide fast trains toward the problem solution.

1.5.4
Strategies to Set up Simple Models

In many cases, a simple three-step procedure can be used to set up a mathematical model. Consider the following

Problem 1:
Which volumes of fluids A and B should be mixed to obtain 150 l of a fluid C that contains 70 $\mathrm{g\,l^{-1}}$ of a substance, if A and B contain 50 $\mathrm{g\,l^{-1}}$ and 80 $\mathrm{g\,l^{-1}}$, respectively?

For this simple problem, many of us will immediately write down the correct equations:

$$x + y = 150 \tag{1.8}$$

$$50x + 80y = 70 \cdot 150 \tag{1.9}$$

where x [l] and y [l] are the unknown volumes of the fluids A and B. For more complex problems, however, it is good to have a systematic procedure to set up the equations. A well-proven procedure that works for a great number of problems can be described as follows:

Note 1.5.7 (Three steps to setup a model)

- *Step 1:* Determine the number of unknowns, that is, the number of quantities that must be determined in the problem. In many problem formulations, you just have to read the last sentence where the question is asked.
- *Step 2:* Give *precise* definitions of the unknowns, including units. It is a practical experience that this should not be lumped with step 1.
- *Step 3:* Reading the problem formulation sentence by sentence, translate this information into mathematical statements which involve the unknowns defined in step 2.

Let us apply this to *Problem 1* above. In *step 1* and *step 2*, we would ascertain that *Problem 1* asks for two unknowns which can be defined as

- x: volume of fluid A in the mixture [l]
- y: volume of fluid B in the mixture [l]

These steps are important because they tell us about the unknowns that can be used in the equations. As long as the unknowns are unknown to us, it will be hard to write down meaningful equations in step 3. Indeed, it is a frequent beginner's mistake in mathematical modeling to write down equations which involve unknowns that are not sufficiently well defined. People often just pick up symbols that appear in the problem formulation – such as *A, B, C* in *problem 1* above – and then write down equations like

$$50A + 80B = 70 \tag{1.10}$$

This equation is indeed *almost* correct, but it is hard to check its correctness as long as we lack any precise definitions of the unknowns. The intrinsic problem with equations such as Equation 1.10 lies in the fact that *A*, *B*, *C* are already defined in the problem formulation. There, they refer to the names of the fluids, although they are (implicitly) used to express the volumes of the fluids in Equation 1.10. Thus, let us now write down the same equation using the unknowns x and y defined above:

$$50x + 80y = 70 \tag{1.11}$$

Now the definitions of x and y can be used to check this equation. What we see here is that on the left-hand side of Equation 1.11, the unit is (grams), which results from the multiplication of $50\,\mathrm{gl}^{-1}$ with x [l]. On the right-hand side of Equation 1.11, however, the unit is grams per liter. So we have different units on the different sides of the equation, which proves that this is a wrong equation. At the same time, a comparison of the units may help us to get an idea of what must be done to obtain a correct equation. In this case, it is obvious that a multiplication of the right-hand side of Equation 1.11 with some quantity expressed in liter would solve the unit problem. The only quantity of this kind in the problem formulation is the 150 l volume which is required as the volume of the mixture, and multiplying the 70 in Equation 1.11 with 150 indeed solves the problem in this case.

Note 1.5.8 (Check the units!) Always check that the units on both sides of your equations are the same. Try to ''repair'' any differences that you may find using appropriate data of your problem.

A major problem in *step 3* is to identify those statements in the problem formulation which correspond to mathematical statements, such as equations, inequalities, and so on. The following note can be taken as a general guideline for this:

Note 1.5.9 (Where are the equations?) The statements of the problem formulation that can be translated into mathematical statements, such as equations, inequalities, and so on, are characterized by the fact that they impose restrictions on the values of the unknowns.

Let us analyze some of the statements in *Problem 1* above in the light of this strategy:

- *Statement 1*: 150 l of fluid C are required.
- *Statement 2*: Fluid A contains 50 gl^{-1} of the substance.
- *Statement 3*: Fluid B contains 80 gl^{-1} of the substance.
- *Statement 4*: Fluid C contains 70 gl^{-1} of the substance.

Obviously, *statement 1* is a restriction on the values of x and y, which translates immediately into the equation:

$$x + y = 150 \tag{1.12}$$

Statement 2 and *statement 3*, on the other hand, impose no restriction on the unknowns. Arbitrary values of x and y are compatible with the fact that fluids A and B contain 50 gl^{-1} and 80 gl^{-1} of the substance, respectively. *Statement 4*, however, does impose a restriction on x and y. For example, given a value of x, a concentration of 70 gl^{-1} in fluid C can be realized only for one particular value of y. Mathematically, *statement 4* can be expressed by Equation 1.9 above. You may be able to write down this equation immediately. If you have problems to do this, you may follow a heuristic (i.e. not 100% mathematical) procedure, where you try to start as close to the statement in the problem formulation as possible. In this case, we could begin with expressing *statement 4* as

$$\left\{\text{Concentration of substance in fluid C}\right\} = 70 \tag{1.13}$$

Then, you would use the definition of a concentration as follows:

$$\frac{\left\{\text{Mass of substance in fluid C}\right\}}{\left\{\text{Volume of the mixture}\right\}} = 70 \tag{1.14}$$

The next step would be to ascertain two things:

- The mass of the substance in fluid C comes from fluids A and B.
- The volume of the mixture is 150 l.

This leads to

$$\frac{\left\{\text{Mass of substance in fluid A}\right\} + \left\{\text{Mass of substance in fluid B}\right\}}{150} = 70 \tag{1.15}$$

The masses of the substance in A and B can be easily derived using the concentrations given in *Problem 1* above:

$$\frac{50x + 80y}{150} = 70 \tag{1.16}$$

This is Equation 1.9 again. The heuristic procedure that we have used here to derive this equation is particularly useful if you are concerned with more complex problems where it is difficult to write down an equation like Equation 1.9 just based on intuition (and where it is dangerous to do this since your intuition can be misleading). Hence, we generally recommend the following:

Note 1.5.10 (Heuristic procedure to set up mathematical statements) If you want to translate a statement in a problem formulation into a mathematical statement, such as an equation or inequality, begin by mimicking the statement in the problem formulation as closely as possible. Your initial formulation may involve nonmathematical statements similar to Equation 1.13 above. Try then to replace all nonmathematical statements by expressions involving the unknowns.

Note that what we have described here corresponds to the *systems analysis* and *modeling* steps of the modeling and simulation scheme in Note 1.2.3. Equations 1.8 and 1.9 can be easily solved (by hand and . . .) on the computer using *Maxima*'s `solve` command as it was described in Section 1.5.2 above. In this case, the *Maxima* commands

```
1: solve([
2: x+y =150
3:  ,50*x+80*y=70*150
4:  ]);
```

(1.17)

yield the following result:

```
[[x =50, y =100]]
```

(1.18)

You find the above code in the file `Mix.mac` in the book software (see Appendix A). As you see, the result is written in a nested *list structure* (lists are written in the form "`[a,b,c,...]`" in *Maxima*): the inner list `[x = 50,y = 100]` gives the values of the unknowns of the solution computed by *Maxima*, while the outer list brackets are necessary to treat situations where the solution is nonunique (see the example in Section 1.5.2 above).

Note that lines 1–4 of Equation 1.17 together form a single `solve` command that is distributed over several lines here to achieve a better readability of the system of equations. Note also that the comma at the beginning of line 3 could also have been written at the end of line 2, which may seem more natural at a first glance. The reason for this notation is that in this way it is easier to generate a larger system of equations, by using copies of line 3 with a "paste and copy" mechanism for example. If you do that and have the commas at the end of each line, your last equation generated in this way will end with a comma which should not be there – so we recommend this kind of notation as a "foolproof" method, which makes your life with *Maxima* and other computer algebra software easier.

1.5.4.1 **Mixture Problem**

Since *Problem 1* in the last section was rather easy to solve and the various recommendations made there may thus seem unnecessary at least with respect to this particular problem, let us now see how a more complex problem is solved using these ideas:

Problem 2:
Suppose the fluids A, B, C, D contain the substances S_1, S_2, S_3 according to the following table (concentrations in grams per liter):

	A	B	C	D
S_1	2.5	8.2	6.4	12.7
S_2	3.2	15.1	13.2	0.4
S_3	1.1	0.9	2.2	3.1

What is the concentration of S_3 in a mixture of these fluids that contains 75% (percent by volume) of fluids A and B and which contains 4 gl^{-1} and 5 gl^{-1} of the substances S_1 and S_2, respectively?

Referring to *step 1* and *step 2* of the three-step procedure described in Note 1.5.7, it is obvious that we have only one unknown here which can be defined as follows:

- x: concentration of S_3 in the mixture (grams per liter)

Now *step 3* requires us to write down mathematical statements involving x. According to Note 1.5.9, we need to look for statements in the above problem formulation that impose a restriction on the unknown x. Three statements of this kind can be identified:

- *Statement 1*: 75% of the mixture consists of A and B.
- *Statement 2*: The mixture contains 4 gl^{-1} of S_1.
- *Statement 3*: The mixture contains 5 gl^{-1} of S_2.

Each of these statements excludes a great number of possible mixtures and thus imposes a restriction on x. Beginning with *statement 1*, it is obvious that this statement can not be formulated in terms of x. We are here in a situation where a number of auxiliary variables is needed to translate the problem formulation into mathematics.

Note 1.5.11 (Auxiliary variables) In some cases, the translation of a problem into mathematics may require the introduction of *auxiliary variables*. These variables are ‘‘auxiliary’’ in the sense that they help us to determine the unknowns. Usually, the problem formulation will provide enough information such that the auxiliary variables *and* the unknowns can be determined (i.e. the auxiliary variables will just increase the size of the system of equations).

In this case, we obviously need the following auxiliary variables:

- x_A: percent (by volume) of fluid A in the mixture
- x_B: percent (by volume) of fluid B in the mixture
- x_C: percent (by volume) of fluid C in the mixture
- x_D: percent (by volume) of fluid D in the mixture

Now the above *statement 1* can be easily expressed as

$$x_A + x_B = 0.75 \tag{1.19}$$

Similar to above, *statement 2* and *statement 3* can be formulated as

$$\{\text{Concentration of S1 in the mixture}\} = 4 \tag{1.20}$$

and

$$\{\text{Concentration of S2 in the mixture}\} = 5 \tag{1.21}$$

Based on the information provided in the above table (and again following a similar procedure as in the previous section), these equations translate to

$$2.5x_A + 8.2x_B + 6.4x_C + 12.7x_D = 4 \tag{1.22}$$

and

$$3.2x_A + 15.1x_B + 13.2x_C + 0.4x_D = 5 \tag{1.23}$$

Since x is the concentration of S_3 in the mixture, a similar argumentation shows

$$1.1x_A + 0.9x_B + 2.2x_C + 3.1x_D = x \tag{1.24}$$

So far we have the four equations 1.19, 1.22, 1.23, and 1.24 for the five unknowns x, x_A, x_B, x_C, and x_D, that is, we need one more equation. In this case, the missing equation is given implicitly by the definition of x_A, x_B, x_C, and x_D. These variables express percent values, and hence, we have

$$x_A + x_B + x_C + x_D = 1 \tag{1.25}$$

Altogether, we have now obtained the following system of linear equations:

$$\begin{align}
x_A + x_B &= 0.75 \tag{1.26}\\
2.5x_A + 8.2x_B + 6.4x_C + 12.7x_D &= 4 \tag{1.27}\\
3.2x_A + 15.1x_B + 13.2x_C + 0.4x_D &= 5 \tag{1.28}\\
1.1x_A + 0.9x_B + 2.2x_C + 3.1x_D &= x \tag{1.29}\\
x_A + x_B + x_C + x_D &= 1 \tag{1.30}
\end{align}$$

Again, this system of equations can be solved similar to above using *Maxima*. In the *Maxima* program `Mix1.mac` in the book software (see Appendix A), the problem is solved using the following code

```
1: out:solve([
2: xA+xB=0.75
3: ,2.5*xA+8.2*xB+6.4*xC+12.7*xD=4
4: ,3.2*xA+15.1*xB+13.2*xC+0.4*xD=5
5: ,1.1*xA+0.9*xB+2.2*xC+3.1*xD=x
6: ,xA+xB+xC+xD=1
7: ]);
8: out,numer;
```

(1.31)

which yields the following results in *Maxima*:

```
          141437       1365        1783       77          14485
(%o6) [[x = ------, xD =----, xC = -----, xB =----, xA =------]]
          98620        19724       9862       4931        19724
(%o7) [[x = 1.434161427702292, xD = 0.06920502940580003, xC = 0.1807949705942,
                         xB = 0.01561549381464206, xA = 0.7343845061853579]]
```

As can be seen, the equation system 1.26–1.30 corresponds to lines 2–6 of the above code and these lines of code are embedded into *Maxima*'s `solve` command similar to the code in 1.17 that was discussed in the previous section. The only new thing is that the result of the solve command is stored in a variable named `out` in line 1 of Equation 1.31. This variable `out` is then used in line 8 of the code to produce a decimal result using *Maxima*'s `numer` command. This is why the *Maxima* output above comprises of two parts: The output labeled as "`(%o6)`" is the immediate output of the `solve` command, and as you can see above the solution is expressed in terms of fractions there. Although this is the most precise way to express the solution, one may prefer decimal numbers in practice. To achieve this, the `numer` command in line 8 of code (1.31) produces the second part of the above output which is labeled as "`(%o7)`". So we can finally say that the solution of *problem* 2 above is $x \approx 1.43$ gl^{-1}, which is the approximate concentration of S_3 in the mixture.

1.5.4.2 **Tank Labeling Problem**

When fluids are stored in horizontal, cylindrical tanks similar to the one shown in Figure 1.7b, one typically wants to have labels on the front side of the tank as shown in Figure 1.7a. In practice, this problem is often solved "experimentally", that is, by filling the tank with well-defined fluid volumes, and then setting the labels at the position of the fluid surface that can be seen from outside. This procedure may of course be inapplicable in situations where the fluid surface cannot be seen from outside. More important, however, is the cost argument: this

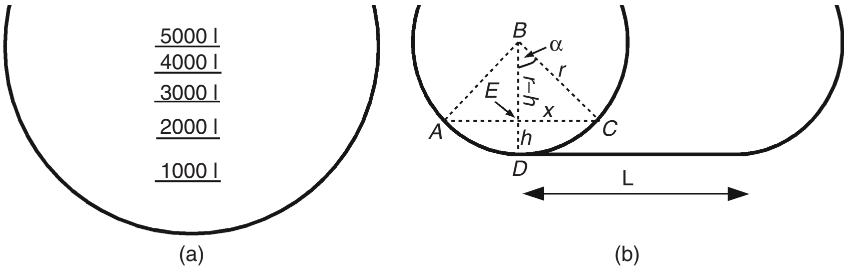


Fig. 1.7 (a) Tank front side with volume labels. (b) Unknowns and auxiliary variables of the tank labeling problem.

experimental procedure is expensive in terms of time (working time of the people who are performing the experiment) and material (e.g. the water that is wasted during the experiment). It is much cheaper here to apply the mathematical model that will be developed below. Unfortunately, the situation in this example – where the problem *could* be solved cheap and efficiently using mathematical models and open-source software, but where expensive experimental procedures or, in some cases, expensive commercial software solutions are used – is still rather the rule than the exception in many fields.

Let us start with the development of an appropriate mathematical model. Let h (decimeters) be the height of a label at the front side of the tank as indicated in Figure 1.7b, and let $V(h)$ (cubic decimeters) be the filling volume of the tank that corresponds to h. If we want to determine the label height for some filling volume V_f, then the following equation must be solved for h:

$$V(h) = V_f \tag{1.32}$$

Referring to Figure 1.7b, $V(h)$ can be expressed as

$$V(h) = ACD \cdot L \tag{1.33}$$

where ACD (square decimeters) corresponds to the surface at the front side of the tank that is enclosed by the line segments AC, CD and DA. ACD can be expressed as

$$ACD = ABCD - ABC \tag{1.34}$$

where the circular segment $ABCD$ is

$$ABCD = \frac{2\alpha}{2\pi}\pi r^2 = \alpha r^2 \tag{1.35}$$

In the last equation, α is expressed in radians (which makes sense here since the problem is solved based on *Maxima* below, which uses radians in its trigonometric functions). The surface of the triangle ABC is

$$ABC = x(r-h) \tag{1.36}$$

where

$$x = \sqrt{r^2 - (r-h)^2} \tag{1.37}$$

due to the theorem of Pythagoras. Using the last five equations and

$$\alpha = \cos^{-1}\left(\frac{r-h}{r}\right) \tag{1.38}$$

in Equation 1.32 yields

$$L \cdot \cos^{-1}\left(\frac{r-h}{r}\right) r^2 - L\sqrt{r^2-(r-h)^2}(r-h) = V_f \tag{1.39}$$

Unlike the equations treated in the last sections, this is now a *transcendental equation* that cannot be solved in closed form using *Maxima*'s `solve` command as before. To solve Equation 1.39, numerical methods such as the bisection method or Newton's method must be applied [18]. In *Maxima*, the `find_root` command can be applied as follows:

```
1: for i:1 thru 4 do
2: (
3: out:find_root(
4: L*acos((r-h)/r)*r^2-L*sqrt(r^2-(r-h)^2)*(r-h)=i*1000
5: ,h,0,r
6: ),
7: print("Label for V=",i*1000,"l:",out,"dm")
8: );
```

(1.40)

This is the essential part of `Label.mac`, a *Maxima* code which is a part of the book software (see Appendix A), and which solves the tank labeling problem assuming a 10 000 l tank of length $L = 2$ m based on Equation 1.39. Equation 1.39 appears in line 4 of the code, with its right-hand side replaced by `i*1000` which successively generates 1000, 2000, 3000, and 4000 as the right-hand side of the equation due to the `for` command that is applied in line 1, so the problem is solved for 1000, 2000, 3000, and 4000 l of filling volume in a single run of the code (note that the 5000, 6000, and so on labels can be easily derived from this if required). What the `for`...`thru`...`do` command in line 1 precisely does is this: it first sets $i = 1$ and then executes the entire code between the brackets in lines 2 and 8, which solves the problem for $V_f = 1000$ l; then, it sets $i = 2$ and executes the entire code between the brackets in lines 2 and 8 again, which solves the problem for $V_f = 2000$, and so on until the same has been done for $i = 4$ (the upper limit given by "`thru`" in line 1).

Note that the arguments of the `find_root` command are in lines 4 and 5, between the brackets in lines 3 and 6. Its first argument is the equation that is to be solved (line 4), which is then followed by three more arguments in line 5: the variable to be solved for (`h` in this case), and upper and lower limits for the interval in which the numerical algorithm is expected to look for a solution of the equation (0 and r in this case). Usually, reasonable values for these limits can be derived from the application – in this case, it is obvious that $h > 0$, and it is likewise obvious that we will have $h = r$ for 5000 l filling volume since a 10 000–l tank is assumed, which means that we will have $h < r$ for filling volumes below 5000 l. The `print` command prints the result to the computer screen. Note how text, numbers and variables (such as the variable `out` that contains the result of the `find_root` command, see line 3) can be mixed in this command. Since the `print` is a part of the `for`...`thru`...`do` environment, it is invoked four times and produces the following result:

```
Label for V= 1000 l: 3.948086422946864 dm
Label for V= 2000 l: 6.410499677168014 dm
Label for V= 3000 l: 8.582542383270068 dm
Label for V= 4000 l: 10.62571600771833 dm
```

1.5.5 Linear Programming

All mathematical models considered so far were formulated in terms of equations only. Remember that according to Definition 1.4.1, a mathematical model may involve any kind of mathematical statements. For example, it may involve inequalities. One of the simplest class of problems involving inequalities are linear programming problems that are frequently used e.g. in operations research. Consider the following problem taken from the linear programming article of *Wikipedia.org:*

Linear programming example

Suppose a farmer has a piece of farm land, say A square kilometers large, to be planted with either wheat or barley or some combination of the two. Furthermore, suppose the farmer has a limited permissible amount F of fertilizer and P of insecticide which can be used, each of which is required in different amounts per unit area for wheat (F_1, P_1) and barley (F_2, P_2). Let S_1 be the selling price of wheat, and S_2 the price of barley. How many square kilometers should be planted with wheat versus barley to maximize the revenue?

Denoting the area planted with wheat and barley with x_1 and x_2 respectively, the problem can be formulated as follows:

$$x_1, x_2 \geq 0 \tag{1.41}$$

$$x_1 + x_2 \leq A \tag{1.42}$$

$$F_1x_1 + F_2x_2 \leq F \tag{1.43}$$

$$P_1x_1 + P_2x_2 \leq P \tag{1.44}$$

$$S_1x_1 + S_2x_2 \to \max \tag{1.45}$$

Here, Equation 1.41 expresses the fact that the farmer cannot plant a negative area, Equation 1.42 the fact that no more than the given A square kilometers of farm land can be used, Equations 1.43 and 1.44 express the fertilizer and insecticide limits, respectively, and Equation 1.45 is the required revenue maximization. Taking Equations 1.41–1.45 as M, the system S as the farm land and the question Q, ‘‘How many square kilometers should be planted with wheat versus barley to maximize the revenue?’’, a mathematical model (S, Q, M) is obtained. For any set of parameter values for A, F, P, . . . , the problem can again be easily solved using *Maxima*. This is done in the *Maxima* program `Farm.mac` which you find in the book software (see Appendix A). Let us look at the essential commands of this code:

```
1: load(simplex);
2: U:[x1>=0
3: ,x2>=0
4: ,x1+x2<=A
5: ,F1*x1+F2*x2 <=F
6: ,P1*x1+P2*x2<=P];
7: Z:S1*x1+S2*x2;
8: maximize_lp(Z,U);
```

(1.46)

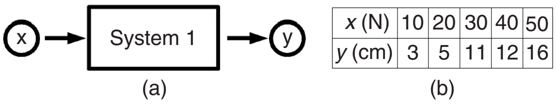
Line 1 of this code loads a package required by *Maxima* to solve linear programming problems. Lines 2–6 define the inequalities, corresponding to Equations 1.41–1.44 above. Note that lines 2–6 together make up a single command that stores the list of inequalities in the variable `U`. Line 7 defines the function `Z` that is to be maximized, and the problem is then solved in line 8 using *Maxima*’s `maximize_lp` command. Based on the parameter settings in `Farm.mac`, *Maxima* produces the following result:

```
[100, [x2 = 50, x1 = 0]]
```

This means that a maximum revenue of 100 is obtained if the farmer plants barley only (50 square kilometers).

1.5.6 Modeling a Black Box System

In Section 1.3 it was mentioned that the systems investigated by scientists or engineers typically are ‘‘input–output systems’’, which means they transform the given input parameters into output parameters. Note that the previous examples were indeed referring to such ‘‘input–output systems’’. In the tin example, the radius and height of the tin are input parameters and the surface area of the tin is



x (N)	10	20	30	40	50
y (cm)	3	5	11	12	16

Fig. 1.8 (a) System 1 with input x (N) and output y (cm). (b) System 1 data (file `spring.csv` in the book software).

the output parameter. In the plant growth example, the growth rate of the plant and its initial biomass is the input and the resulting time–biomass curve is the output (details in Chapter 3). In the tank example, the geometrical data of the tank and the concentration distribution are input parameters while the mass of the substance is the output. In the linear programming examples, the areas planted with wheat or barley are the input quantities and the resulting revenue is the output. Similarly, all systems in the examples that will follow can be interpreted as input–output systems.

The exploration of an example input–output system in some more detail will now lead us to further important concepts and definitions. Assume a ''system 1'' as in Figure 1.8 which produces an output length y (centimeters) for every given input force x [N]. Furthermore, assume that we do not know about the processes inside the system that transform x into y, that is, let this system be a ''black box'' to us as described above. Consider the following problem:

Q: Find an input x that generates an output $y = 20\,\text{cm}$.

This defines the question Q of the mathematical model (S, Q, M) that we are going to define. S is the ''system 1'' in Figure 1.8a, and we are now looking for an appropriate set of mathematical statements M that can help us to answer Q.

All that the investigator of system 1 can do is to produce some data using the system, hoping that these data will reveal something about the processes occurring inside the ''black box''. Assume that the data in the file `spring.csv` (which you find in the `PhenMod/LinReg` directory of the book software, see Appendix A) have been obtained from this system, see Figure 1.8b. To see what happens, the investigator will probably produce a plot of the data as in Figure 1.9a. Note that the plots in Figure 1.9 were generated using the scatter plot option of *OpenOffice.org Calc* (see Appendix A on how you can obtain this software). Figure 1.9a suggests that there is an approximately linear dependence between the x- and y-data. Mathematically, this means that the function $y = f(x)$ behind the data is a straight line:

$$f(x) = ax + b \tag{1.47}$$

Now the investigator can apply a statistical method called *linear regression* (which will be explained in detail in Section 2.2) to determine the coefficients a and b of this equation from the data, which leads to the ''regression line''

$$f(x) = 0.33x - 0.5 \tag{1.48}$$

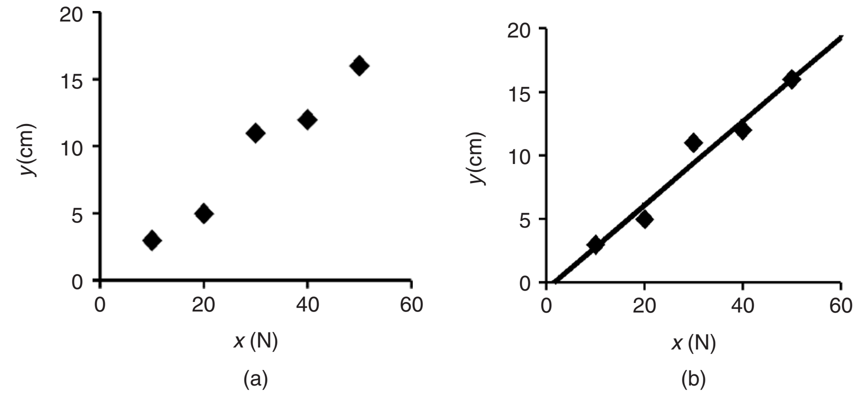


Fig. 1.9 (a) Plot of the data in `spring.csv`. (b) System 1 data with regression line. Both plots generated using *Calc*, see Section 2.1.1.1.

Figure 1.9b shows that there is a good coincidence or, in statistical terminology, a good "fit" between this regression line and the data. Equation 1.48 can now be used as the M of a mathematical model of system 1. The question Q stated above ("Which system input x generates a desired output $y = 20$ cm? ") can then be easily answered by setting $y = f(x) = 20$ in Equation 1.48, that is,

$$20 = 0.33x - 0.5 \tag{1.49}$$

which gives $x \approx 62.1$ N. Of course, this is just an *approximate result* for several reasons. First of all, Figure 1.9 shows that there are some deviations between the regression line and the data. These deviations may be due to measurement errors, but they may also reflect some really existing effects. If the deviations are due to measurement errors, then the precise location of the regression line and hence, the prediction of x for $y = 20$ cm is affected by these errors. If, on the other hand, the deviations reflect some really existing effects, then Equation 1.48 is no more than an approximate model of the processes that transform x into y in system 1, and hence, the prediction of x for $y = 20$ cm will be only approximate. Beyond this, predictions based on data such as the data in Figure 1.8b are always approximate for principal reasons. The y-range of these data ends at 16 cm, and system 1 may behave entirely different for y-values beyond 16 cm which we would not be able to see in such a data set. Therefore, the experimental validation of predictions derived from mathematical models is always an indispensable part of the modeling procedure (see Section 1.2). See also Chapter 2 for a deeper discussion of the quality of predictions obtained from black box models.

The example shows the *importance of statistical methods* in mathematical modeling. First of all, statistics itself is a collection of mathematical models that can be used to describe data or to draw inferences from data [19]. Beyond this, statistical methods provide a necessary link between nonstatistical mathematical models and the real world. In mathematical modeling, one is always concerned with experimental data, not only to validate model predictions, but also to develop hypotheses about the system, which help to set up appropriate equations. In the example, the data led us to the hypothesis that there is a linear relation between x and y. We have used a plot of the data (Figure 1.9) and the regression method to find the coefficients in Equation 1.48. These are methods of descriptive statistics, which can be used to summarize or describe data. Beyond this, inferential statistics provides methods that allow conclusions to be drawn from data in a way that accounts for randomness and uncertainty. Some important methods of descriptive and inductive statistics will be introduced below (Section 2.1).

Note 1.5.12 Statistical methods provide the link between mathematical models and the real world.

The reader might say that the estimate of x above could also have been obtained without any reference to models or computations, by a simple tuning of the input using the real, physical system 1. We agree that there is no reason why models should be used in situations where this can be done with little effort. In fact, we do not want to propose any kind of a fundamentalist ''mathematical modeling and simulation'' paradigm here. A *pragmatic approach* should be used, that is, any problem in science and engineering should be treated using appropriate methods, may this be mathematical models or a tuning of input parameters using the real system. It is just a fact that in many cases the latter cannot be done in a simple way. The generation of data such as in Figure 1.8 may be expensive, and thus, an experimental tuning of x toward the desired y may be inapplicable. Or, the investigator may be facing a very complex interaction of several input and output parameters, which is rather the rule than the exception as explained in Section 1.1. In such cases, the representation of a system in mathematical terms can be the only efficient way to solve the problem.

1.6
Even More Definitions

1.6.1
Phenomenological and Mechanistic Models

The mathematical model used above to describe system 1 is called a *phenomenological model* since it was constructed based on experimental data only, treating the system as a black box, that is, without using any information about the internal processes occurring inside system 1 when x is transformed into y. On the other hand, models that are constructed using information about the system S are called *mechanistic models*, since such models are virtually based on a look into the internal mechanics of S. Let us define this as follows [11]:

Definition 1.6.1 (Phenomenological and mechanistic models) A mathematical model (S, Q, M) is called

- *phenomenological*, if it was constructed based on experimental data only, using no a priori information about S,
- *mechanistic*, if some of the statements in M are based on a priori information about S.

Phenomenological models are also called *empirical models, statistical models, data-driven models* or *black box models* for obvious reasons. Mechanistic models for which all necessary information about S are available are also called *white box models*. Most mechanistic models are located somewhere between the extreme black and white box cases, that is, they are based on some information about S while some other important information is unavailable. Such models are sometimes called *gray box models* or *semi-empirical models* [20].

To better understand the differences between phenomenological and mechanistic models, let us now construct an alternative mechanistic model for system 1 (Figure 1.8). Above, we have treated system 1 as a black box, that is, we have used no information about the way in which system 1 transforms some given input x into the output y (Figure 1.8). Let us now assume that the internal mechanics of system 1 looks as shown in Figure 1.10, that is, assume that system 1 is a mechanical spring, x is a force acting on that spring, and y is the resulting elongation. This is now an *a priori information* about system 1 in the sense of Definition 1.6.1 above, and it can be used to construct a mechanistic mathematical model based on elementary physical knowledge. As is well known, mechanical springs can be described by Hooke's law, which in this case reads

$$x = k \cdot y \tag{1.50}$$

where k is the spring constant (newtons per centimeter), a measure of the elasticity of the spring. The parameter k is either known (e.g. from the manufacturer of the spring), or estimated based on data such as those in Figure 1.8. Now the following mechanistic mathematical model (S, Q, M) is obtained:

- S: System 1
- Q: Which system input x generates a desired output of $y = 20\,\text{cm}$?
- M: Equation 1.50

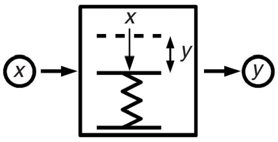


Fig. 1.10 Internal mechanics of system 1.

Based on this model, question Q can be answered as before by setting $y = 20\,\text{cm}$ in the model equation (1.50), which leads to

$$x = k \cdot 20 \tag{1.51}$$

that is, we can answer the question Q depending on the value of the spring constant, k. For example, assuming a value of $k \approx 3.11\,\text{N}\,\text{cm}^{-1}$ for the spring constant, we would get the same estimate $x \approx 62.1$ N as above. The mechanistic model of system 1 has several important advantages compared to the phenomenological model, and these advantages are *characteristic advantages of the mechanistic approach*. First of all, mechanistic models generally allow better predictions of system behavior. The phenomenological model equation (1.48) was derived from the data in Figure 1.8. These data involve forces x between 10 and 50 N. As mentioned below in our discussion of regression methods, this means that one can expect Equation 1.48 to be valid only close to this range of data between 10 and 50 N. The mechanistic model equation (1.50), on the other hand, is based on the well-established physical theory of a spring. Hence, we have good reason to expect its validity even outside the range of our own experimental testing.

Mechanistic models do also allow *better predictions* of modified systems. Assume for example that system 1 in Figure 1.10 is replaced by a system 2 that consists of two springs. Furthermore, assume that each of these system 2 springs has the same spring constant k as the system 1 spring. Then, in the phenomenological approach, the model developed for system 1 would be of no use, since we would not know about the similarity of these two systems (remember that the phenomenological approach assumes that no details are known about the internal mechanics of the system under consideration). This means that a new phenomenological model would have to be developed for system 2. A new data set similar to Figure 1.8 would be required, appropriate experiments would have to be performed, and afterwards, a new regression line similar to Figure 1.9 would have to be derived from the data. In the mechanistic approach, on the other hand, Hooke's law would immediately tell us that in the case of two springs the appropriate modification of Equation 1.50 is

$$x = 2k \cdot y \tag{1.52}$$

Another advantage of mechanistic models is the fact that they usually involve *physically interpretable parameters*, that is, parameters which represent real properties of the system. To wit: the numerical coefficients of the phenomenological model equation 1.47 are just numbers which cannot be related to the system. The parameter k of the mechanistic model equation 1.50, on the other hand, can be related to system properties, and this is of particular importance when we want to optimize system performance. For example, if we want smaller forces x to be required for a given elongation y, then in the phenomenological approach we would have to test a number of systems 2, 3, 4, . . . , until we would eventually arrive at some system with the desired properties. That is, we would have to apply a trial-and-error method. The mechanistic model, on the other hand, tells us exactly what we have to do: we have to replace the system 1 spring with a spring having a smaller spring constant k, and this will reduce the force x required for a given elongation y. In this simple example, it may be hard to imagine that someone would really use the phenomenological approach instead of Hooke's law. But the example captures an essential difference between phenomenological and mechanistic models, and it tells us that we should use mechanistic models if possible.

So, if mechanistic models could be set up easily in every imaginable situation, we would not have to talk about phenomenological models here. However, in many situations, it is not possible or feasible to use mechanistic models. As an essential prerequisite, *mechanistic models need a priori knowledge of the system*. If nothing is known about the system, then we are in the ''black box'' situation and have to apply phenomenological models. Suppose, for example, we want to understand why some roses wilt earlier than others (this example will be explained in more detail in Section 2.3). Suppose we assume that this is related to the concentrations of certain carbohydrates that can be measured. Then we cannot set up a mechanistic model as long as we do not know all the relevant processes that connect those carbohydrate concentrations with the observed freshness of the rose. Unless these processes are known, all we can do is to produce some data (carbohydrate concentration versus some appropriate measure of rose freshness) and analyze these data using phenomenological models.

This kind of situation where little is known about the system under investigation is rather the rule than the exception, particularly at early stages of a scientific investigation, or at the early stages of a product development in engineering. We may also be in a situation where we principally know enough details about the system under investigation, but where the system is so complex that it would take too much time and resources to setup a mechanistic model. An example is the optimization of the wear resistance of composite materials: Suppose that a composite material is made of the materials $M_1, M_2, \ldots, M_n$, and we want to know how the relative proportions of these materials should be chosen in order to maximize the composite materials resistance to wear. Then, the wear resistance of the composite material can depend in an extremely complex way on its composition. The author has investigated a situation of this kind where mechanistic modeling attempts failed due to the complexity of the overall system, and where a black box-type phenomenological neural network approach (see Section 2.5) was used instead [21]. An important *advantage of phenomenological models* is that they can be used in black box situations of this kind, and that they typically require much less time and resources. Pragmatic considerations should decide which type of model is used in practice. A mechanistic model will certainly be a bad choice if we need three weeks to make it work, and if it does not give substantially better answers to our question Q compared to a phenomenological model which can be set up within a day.

Note 1.6.1 (Phenomenological vs. mechanistic) *Phenomenological models* are universally applicable, easy to set up, but limited in scope. *Mechanistic models* typically involve physically interpretable parameters, allow deeper insights into system performance and better predictions, but they require a priori information on the system and often need more time and resources.

1.6.2 Stationary and Instationary models

It was already mentioned above that the question Q is an important factor that determines the appropriate mathematical model (S, Q, M). As an example, we have considered the alternative treatment of mechanical problems with the equations of classical or relativistic mechanics depending on the question Q that is investigated. In the system 1 example, we have used Q: "Which system input x generates a desired output of $y = 20\,\text{cm}$? ". Let us now modify this Q in order to find other important classes of mathematical models. Consider the following question:

Q: If a constant force x acts on the spring beginning with $t = 0$, what is the resulting elongation $y(t)$ of the spring at times $t > 0$?

This question cannot be answered based on the models developed above. The phenomenological model (Equation 1.48) as well as the mechanistic model (Equation 1.50) both refer to the so-called stationary state of system 1. This means that the elongation y expressed by these equations represents the time-independent (= stationary) state of the spring which is achieved after the spring has been elongated into the state of equilibrium where the force x exactly matches the force of the spring. On the other hand, the above question asks for the instationary (i.e. time-dependent) development of the elongation $y(t)$, beginning with time $t = 0$ when the force x is applied to the spring. To compute this $y(t)$, an instationary mathematical model (S, Q, M) is needed where the mathematical statements in M involve the time t. Models of this kind can be defined based on ordinary differential equations (details in Chapter 3). To make this important distinction between stationary and instationary models precise, let us define

Definition 1.6.2 (Stationary/instationary models) A mathematical model (S, Q, M) is called

- *instationary*, if at least one of its system parameters or state variables depends on time and
- *stationary* otherwise.

1.6.3 Distributed and Lumped models

Suppose now that the spring in system 1 broke into pieces under normal operational conditions, and that it is now attempted to construct a more robust spring. In such a situation, it is natural to ask the following question:

Q: Which part of the spring should be reinforced?

Naturally, those parts of the spring which bear the highest mechanical stresses should be reinforced. To identify these regions, we need to know the distribution of stresses inside the spring under load. Let $\sigma(x, y, z)$ denote the mechanical stress distribution inside the spring depending on the spatial coordinates x, y, *and* z. Then we need a mathematical model with $\sigma(x, y, z)$ as a state variable. Such a mathematical model can be formulated based on PDEs as will be explained in Chapter 4. The important difference between this model and the previous models of system 1 lies in the fact that in this case the state variable depends on the spatial coordinates. To predict the equilibrium elongation of the spring using Equations 1.47 or 1.50, it was sufficient to describe the spring based on the spring constant k only. These equations, however, cannot be used to derive any spatially distributed information regarding the spring. In this kind of models, all spatial information is lumped together into the parameter k. In the case above, this was justified by the fact that the equilibrium position of a spring can be predicted with sufficient precision using k. On the other hand, if one is asking for the internal stress distribution in the spring, a spatially distributed description of the stresses inside the spring is needed. This motivates the following:

Definition 1.6.3 (Distributed/lumped models) A mathematical model (S, Q, M) is called

- *distributed*, if at least one of its system parameters or state variables depends on a space variable,
- *lumped* otherwise.

1.7 Classification of Mathematical Models

Based on the examples in the last section, the reader can now distinguish between some basic classes of mathematical models. We will now widen our perspective toward a look at the entire ''space of mathematical models'', that is, this section will give you an idea of various types of mathematical models that are used in practice.

Note 1.7.1 The practical use of a classification of mathematical models lies in the fact that you understand ''where you are'' in the space of mathematical models, and which types of models might be applicable to your problem beyond the models that you have already used.

1.7.1 From Black to White Box Models

The "space of mathematical models" evolves naturally from Definition 1.4.1, where we have defined a mathematical model to be a triple (S, Q, M) consisting of a system S, a question Q, and a set of mathematical statements M. Based on this definition, it is natural to classify mathematical models in an *SQM space*. Figure 1.11a shows one possible approach to visualize this SQM space of mathematical models, based on a classification of mathematical models between black and white box models. Psychological and social systems constitute the "black box" end of the spectrum. Only very vague phenomenological models can be developed for these systems due to their complexity and due to the fact that too many subprocesses are involved which are not sufficiently understood. On the other hand, mechanical systems, electrical circuits etc. are at the white box end of the spectrum since they can be very well understood in terms of mechanistic models (a famous example is Newton's model of planetary motion).

Note that the three dimensions of a mathematical model (S, Q, M) can be seen in the figure: the systems (S) are classified on top of the bar, immediately below the bar there is a list of objectives that mathematical models in each of the segments may have (which is Q), and at the bottom end there are corresponding mathematical structures (M) ranging from algebraic equations (AEs) to differential equations (DEs). Equation 1.47 (Section 1.5.6) is an example of a mathematical model in the form of an AE. As suggested by Figure 1.11, black box regression models of this kind are widely used for the modeling for example, of psychological, social, or economic systems (see Chapter 2 for more on regression models). On the other hand, the wine fermentation model discussed in Section 3.10.2 exemplifies the modeling of a biological/chemical system using ODEs (see Chapters 3 and 4 for more examples of DE models).

The "Q"-criteria in Figure 1.11a illustrate that mathematical models can be used to solve increasingly challenging problems as the model gradually turns from a

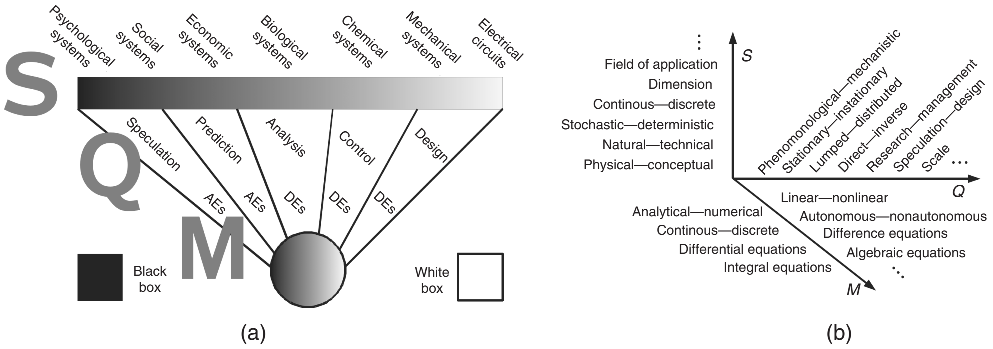


Fig. 1.11 (a) Classification of mathematical models between black and white box models (adapted from [3]). (b) Classification of mathematical models in the SQM space.

black box to a white box model. At the black box end of the spectrum, models can be used to make more or less reliable predictions based on data. For example, you may think here of attempts that have been made to predict share prices using the neural network methods described in Chapter 2 [22]. The model of a biological predator–prey system discussed in Section 3.10.1 is already ''white enough'' such that it can be used for an analysis of the dynamical system behavior in terms of phase plot diagrams such as Figure 3.17. Beyond this, models of chemical systems can be so precise that they can be used for a control of a process such as the wine fermentation process discussed in Section 3.10.2.

At the white box end of the spectrum, mathematical models can be applied to design, test, and optimize systems and processes on the computer before they are actually physically realized. This is used e.g. in *virtual engineering*, which includes techniques such as interactive design using CFD (see [23] and Section 4.10.3) or *virtual prototyping* [7, 24, 25]. As an example, you may think of the computation of the temperature distribution within a three-dimensional device using finite-element software, as it will be discussed in Section 4.9 below. Based on the method described there, *what-if studies* can be performed, that is, it can be investigated what happens with the temperature distribution if you change certain characteristics of the device virtually on the computer, and this can then be used to optimize the construction of the device so as to achieve certain desired characteristics of the temperature distribution.

1.7.2 SQM Space Classification: S Axis

Since mathematical models are characterized by their respective individual *S*, *Q* and *M* ''values'', one can also think of each model as being located somewhere in the ''SQM space'' of Figure 1.11b. On each of the *S*-, *Q*- and *M*-axes of the figure, mathematical models are classified with respect to a number of criteria which were compiled based on various classification attempts in the literature [3, 11, 20, 26–30]. Let us explain these criteria, beginning with the S axis of Figure 1.11b:

Physical – conceptual. Physical systems are part of the real world, for example, a fish or a car. Conceptual systems are made up of thoughts and ideas, for example, a set of mathematical axioms. This book focuses entirely on physical systems.

Natural – technical. Naturally, a natural system is a part of nature, such as a fish or a flower, while a technical system is a car, a machine, and so on. An example of a natural system is the predator–prey system treated in Section 3.10.1, the stormer viscometer treated in Section 2.4 exemplifies a technical system.

Stochastic – deterministic. Stochastic systems involve random effects, such as rolling dice, share prices and so on. Deterministic systems involve no or very little random effects, for example, mechanical systems, such as the planetary system, a pendulum, and so on. In a deterministic system, a particular state A of the system is always followed by one and the same state B, while A may be followed by B, C or other states in an unpredictable way if the system is stochastic [31]. Below, stochastic models will be considered mainly in Chapter 2 and deterministic models mainly in Chapters 3 and 4.

Continuous – discrete. Continuous systems involve quantities that change continuously with time, such as sugar and ethanol concentrations in a wine fermenter (Section 3.10.2). Discrete systems, on the other hand, involve quantities that change at discrete times only, such as the number of individuals in animal populations (Section 3.10.1). Note that on the M axis of Figure 1.11, continuous systems can be represented by discrete mathematical statements and vice versa (e.g. a continuous mathematical formulation is used in Section 3.10.1 to describe the discrete predator–prey system).

Dimension. Depending on their spatial symmetries, physical systems can be described using 1, 2, or 3 space variables. As will be discussed in Section 4.3.3, the number of space variables used to describe a physical system is called its *dimension* (frequently denoted 1D, 2D, or 3D). Examples: a 1D temperature distribution is computed in Section 4.6 and a 3D temperature distribution in Section 4.9.

Field of application. We can distinguish between chemical systems, physical systems, biological systems, and so on. Systems from these and more fields of application will be considered below.

1.7.3
SQM Space Classification: Q Axis

On the *Q*- axis of Figure 1.11b, we have the following categories:

Phenomenological – mechanistic. This has been discussed in detail in Section 1.6. Phenomenological models are treated in Chapter 2 and mechanistic models in Chapters 3 and 4.

Stationary – instationary. Again, this has been discussed in Section 1.6. As discussed there, it depends on the question which we are asking (i.e. on the "*Q*" of a mathematical model (*S*, *Q*, *M*)) whether a stationary (time-independent) or instationary (time-dependent) model is appropriate. See also *Problem* 1 (instationary) and *Problem* 2 (stationary) in Section 4.1.3.

Lumped – distributed. Again, see Section 1.6. As was discussed there, it depends on the question which we are asking (i.e. on the "*Q*" of a mathematical model (*S*, *Q*, *M*)) whether a lumped (space-independent) or distributed (space-dependent) model is appropriate. The wine fermentation model (Section 3.10.2) is an example of a lumped model since it does not use spatial coordinates. On the other hand, the computation of a 3D temperature distribution in Section 4.9 is based on a distributed model.

Direct – inverse. Consider an input–output system as in Figure 1.2a. If *Q* assumes given input and system parameters and asks for the output, the model solves a so-called *direct problem* [3]. Most of the models below refer to direct problems. If, on the other hand, Q asks for the input or for parameters of S, the model solves a so-called inverse problem [32]. If Q asks for parameters of S, the resulting problem is also called a *parameter identification problem*. Examples are the regression and neural network models discussed in Chapter 2, and the fitting of ODEs to data discussed in Section 3.9. If Q asks for input parameters, the resulting problem is also called a *control problem*, since in this case the problem is to *control* the input in a way that generates some desired output ([33] and 4.11.3).

Research – management. Research models are used if Q aims at the understanding of S; management models, on the other hand, are used if the focus is on the solution of practical problems related to S. As pointed out in [20], research models tend to be more complex and less manageable from a practical point of view. Depending on Q, the same mathematical equations can be a part of a research or of a management model. For example, the predator–prey model described in Section 3.10.1 is a research model if the investigator just wants to understand the oscillations of the predator and prey populations, and it is a management model if is used to control the predator and prey populations (but as discussed in Section 3.10.1, this model is so simple that it cannot be seriously used as a management model).

Speculation – design. See the above discussion of Figure 1.11a.

Scale. Depending on Q, the model will describe the system on an appropriate scale. For example, depending on Q it can be appropriate to virtually follow a fluid particle on its way through the complex channels of a porous medium, or just to compute the pressure drop across a porous medium based on its permeability. Obviously, these cases correspond to a description of a porous medium on two scales (microscopic/macroscopic). Details of this example will follow in Section 4.10.2.

1.7.4 SQM Space Classification: M Axis

Finally, let us look at the categories on the M-axis of Figure 1.11b:

Linear – nonlinear. In linear models, the unknowns (or their derivatives) are combined using linear mathematical operations only, such as addition/subtraction or multiplication with parameters. Nonlinear models, on the other hand, may involve the multiplication of unknowns, the application of transcendental functions, and so on. Nonlinear models typically have more (and more interesting) solutions but are harder to solve. Examples are linear or nonlinear regression models (Sections 2.2 and 2.4, respectively) and linear or nonlinear ODEs (Section 3.5).

Analytical – numerical. In analytic models, the system behavior can be expressed in terms of mathematical formulas involving the system parameters. Based on these models, qualitative effects of parameters and the entire system behavior can be studied theoretically, without using concrete values for the parameters. Numerical models, on the other hand, can be used to obtain the system behavior for specific parameter values. See Section 3.6 for a general discussion of analytical models (which are also called *closed form models*) versus numerical models.

Autonomous – nonautonomous. This is a mathematical classification of instationary models (see above). If an equation does not depend explicitly on time, it is called *autonomous*, otherwise nonautonomous; see the examples in Section 3.5.

Continuous – discrete. In continuous models, the independent variables may assume arbitrary (typically real) values within some interval. For example, many of the ODE models discussed in Chapter 3 use time (within some time interval) as the independent variable. In discrete models, on the other hand, the independent variables may assume some discrete values only. An example is the *discrete event simulation* technique discussed in Section 2.7.2, or the Nicholson–Bailey host–parasite interaction model discussed in Section 4.11.1, where the time variable just counts the number of breeding seasons instead of expressing the (continuous) physical time.

Difference equations. In difference equations, the quantity of interest is obtained as a sequence of discrete values. Usually, this is expressed in terms of recurrence relations in which each term of the sequence depends on previous terms. Difference equations are frequently used to describe discrete systems. See the examples in Section 4.11.1.

Differential equations. Differential equations are equations involving derivatives of an unknown function. They are a main tool to set up continuous mechanistic models, see the examples in Chapters 3 and 4.

Integral equations. Integral equations are equations involving an integral of an unknown function.

Algebraic equations. AEs are equations involving the usual algebraic operations such as addition, subtraction, division, and so on. Examples are Equations (1.1) or (1.4) in Section 1.5, or the regression equations discussed in Chapter 2.

Note that some of the above categorizations of mathematical models overlap. For example, both phenomenological and mechanistic models can be lumped or distributed, stationary or instationary, and so on. Thus, it may have confused the reader if a single chapter would have been devoted to each of these categorizations. Instead, it was decided to select the categorization between phenomenological models (Chapter 2) and mechanistic models (Chapters 3 and 4) as the main perspective and as a principle to organize the book. The other categorizations are treated within this perspective, that is, they will be referred to in the context of appropriate examples. Note that referring to Figure 1.11b we can say that the categorization of mathematical models between phenomenological and mechanistic models divides the *SQM* space of mathematical models into two different ‘‘half-spaces’’ along the *Q*-axis. We will repeatedly come back to the above classification of mathematical models in the course of this book, using it like a compass (or, in more up-to-date terminology: like a GPS system) so that the reader will always know about his actual position in the overall space of mathematical models.

1.8
Everything Looks Like a Nail?

To some extent, the modeling and simulation scheme discussed above is just an idealistic theory of how mathematical modeling *should* work, and this must of course be distinguished from the way in which people are dealing with mathematical models in practice. Being aware of this fact, Golomb [34] compiled the following:

Note 1.8.1 (Don'ts of Mathematical Modeling)
1. Don't believe that the model is the reality.
2. Don't extrapolate beyond the region of fit.
3. Don't distort reality to fit the model.
4. Don't retain a discredited model.
5. Don't fall in love with your model.

Don't No. 1 reminds us of the limitations of our models, that is, we should always be aware of the simplifying assumptions made in a model when discussing its implications for the real system. You may know the *cave allegory* of the Greek philosopher Plato, which provides a nice picture of the relationship between a model and the reality, Figure 1.12 [35]. In this allegory, prisoners are chained deep inside a cave in a way that restricts their view to one particular wall of the cave. Behind the prisoners, there is a big fire and some people who are using the light of that fire to project three-dimensional objects such as puppets, animals, and plants onto the cave wall. Plato assumes that the prisoners are chained in the cave since their childhood and thus have never seen anything else apart from the shadows on that cave wall. Thus, they believe that these shadows are the reality, although the

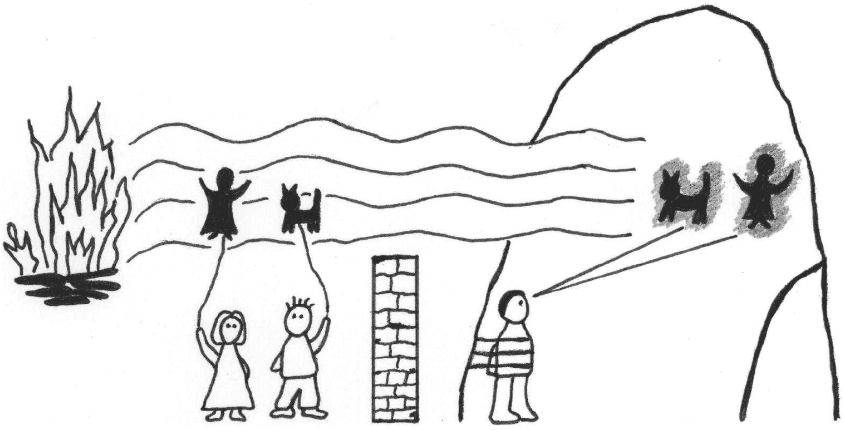


Fig. 1.12 Plato's cave allegory: Don't believe that the model is the reality! (Figure: B. Blüm, idea: *http://commons.wirimedia.org*.)

shadows are of course no more than simplified, two-dimensional models of the real, three-dimensional objects behind them. Very similarly, we must be aware of the fact that we are always ''chained'' in some way as long as we think about reality in terms of a scientific model, which restricts our view on the real system more or less depending on its inherent assumptions.

Don't No. 2 says that models should be used for prediction only in those regions of the parameter space where they are sufficiently supported by experimental data (see Section 2.2.2 and Note 2.2.3 for more details), while *Don'ts Nos 3–5* basically require us to abandon models that fail to pass the validation step of the modeling and simulation scheme (Note 1.2.3). In [11], the message of *Don't Nos 3–5* is expressed as follows:

When you have a hammer, you look for a nail.
When you have a *good* hammer, everything looks like a nail.

You understand the message: People always tend to solve problems similar to the way in which they successfully solved problems in the past. Yesterday, our problem might have been to drive a nail into a piece of wood, and we might have solved this problem adequately using a hammer. Today, however, we may have to drive a *screw* into a piece of wood, and it is of course not quite such a good idea to use the hammer again. Similarly, mathematical models are like tools that help us to solve problems, and we will always tend to reuse the models that helped us to solve our yesterday's problems. This is like a law of nature in mathematical modeling, similar to Newton's law of inertia; let us call it the ''*law of inertia of mathematical modeling*''. Forces need to be applied to physical bodies to change their state of motion, and in a similar way forces need to be applied in a mathematical modeler's mind before he will eventually agree to replace established models by more adequate approaches. Even great scientists such as A. Einstein were affected by this kind of inertia. Einstein did not like the idea that the physical universe is probabilistic rather than deterministic (a consequence of the ''Copenhagen interpretation'' of quantum mechanics), and he expressed this aversion in his famous quote ''God does not play dice with the universe'' [36]. But do not take this as an excuse for any violation of Golomb's *Don't's*. It just shows that *everybody*, including yourself, should use models with care.