CHAPTER 1

Basic Concepts

It is often said that computers are revolutionizing science and engineering. By using computers we are able to construct complex engineering designs such as space shuttles. We are able to compute the properties of the universe as it was fractions of a second after the big bang. Our ambitions are ever-increasing. We want to create even more complex designs such as better spaceships, cars, medicines, computerized cellular phone systems, and the like. We want to understand deeper aspects of nature. These are just a few examples of computer-supported modeling and simulation. More powerful tools and concepts are needed to help us handle this increasing complexity, which is precisely what this book is about.

This text presents an object-oriented component-based approach to computer-supported mathematical modeling and simulation through the powerful Modelica language and its associated technology. Modelica can be viewed as an almost universal approach to high-level computational modeling and simulation, by being able to represent a range of application areas and providing general notation as well as powerful abstractions and efficient implementations. The introductory part of this book, consisting of the first two chapters, gives a quick overview of the two main topics of this text:

- Modeling and simulation
- The Modelica language

Introduction to Modeling and Simulation of Technical and Physical Systems with Modelica,
First Edition. By Peter Fritzson

The two subjects are presented together since they belong together. Throughout the text Modelica is used as a vehicle for explaining different aspects of modeling and simulation. Conversely, a number of concepts in the Modelica language are presented by modeling and simulation examples. The present chapter introduces basic concepts such as *system, model*, and *simulation*. Chapter 2 gives a quick tour of the Modelica language as well as a number of examples, interspersed with presentations of topics such as object-oriented mathematical modeling. Chapter 3 gives an introduction to the Modelica class concept, whereas Chapter 4 introduces modeling methodology for continuous, discrete, and hybrid systems. Chapter 5 gives a short overview of the Modelica Standard Library and some currently available Modelica model libraries for a range of application domains. Finally, in two of the appendices, examples are presented of textual modeling using the OpenModelica electronic book OMNotebook tool, as well as very simple graphical modeling.

1.1 SYSTEMS AND EXPERIMENTS

What is a system? We have already mentioned some systems such as the universe, a space shuttle, and the like. A system can be almost anything. A system can contain subsystems that are themselves systems. A possible definition of system might be:

- A system is an object or collection of objects whose properties we want to study.

Our wish to study selected properties of objects is central in this definition. The "study" aspect is fine despite the fact that it is subjective. The selection and definition of what constitutes a system is somewhat arbitrary and must be guided by what the system is to be used for.

What reasons can there be to study a system? There are many answers to this question but we can discern two major motivations:

- Study a system to understand it in order to build it. This is the engineering point of view.
- Satisfy human curiosity, for example, to understand more about nature—the natural science viewpoint.

1.1.1 Natural and Artificial Systems

A system according to our previous definition can occur naturally, for example, the universe, it can be artificial such as a space shuttle, or a mix of both. For example, the house in Figure 1.1 with solar-heated warm tap water is an artificial system, that is, manufactured by humans. If we also include the sun and clouds in the system, it becomes a combination of natural and artificial components.

Even if a system occurs naturally, its definition is always highly selective. This is made very apparent in the following quote from Ross Ashby (1956, p. 39):

> *At this point, we must be clear about how a* system *is to be defined. Our first impulse is to point at the* pendulum *and to say "the system is that thing there." This method, however, has a fundamental disadvantage: every material object contains no less than an infinity of variables, and therefore, of possible systems. The real pendulum, for instance, has not only length and position; it has also mass, temperature, electric conductivity, crystalline structure, chemical impurities, some radioactivity, velocity, reflecting power, tensile strength, a surface film of moisture, bacterial contamination, an optical absorption, elasticity, shape, specific gravity, and so on and on. Any suggestion that we should study all the facts is unrealistic, and actually the attempt is never made.*

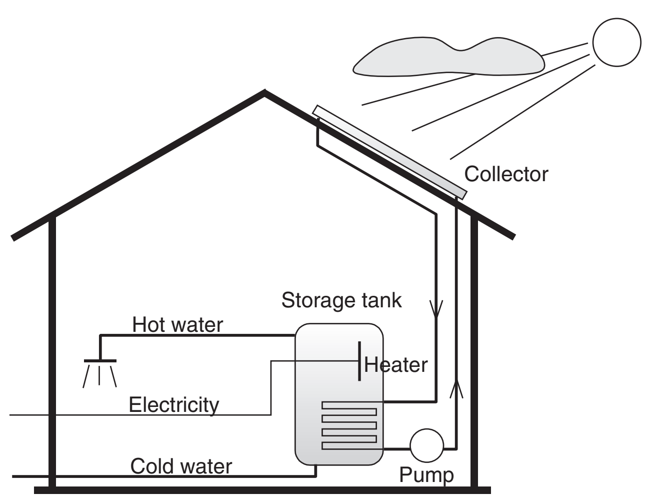

Figure 1.1 A system: a house with solar-heated warm tap water, together with clouds and sunshine.

What is necessary is that we should pick out and study the facts that are relevant to some main interest that is already given.

Even if the system is completely artificial, such as the cellular phone system depicted in Figure 1.2, we must be highly selective in its definition, depending on what aspects we want to study for the moment.

An important property of systems is that they should be *observable*. Some systems, but not large natural systems like the universe, are also *controllable* in the sense that we can influence their behavior through inputs, that is:

- The *inputs* of a system are variables of the environment that influence the behavior of the system. These inputs may or may not be controllable by us.
- The *outputs* of a system are variables that are determined by the system and may influence the surrounding environment.

In many systems the same variables act as *both inputs and outputs*. We talk about *acausal* behavior if the relationships or influences between variables do not have a causal direction, which is the case for relationships described by equations. For example, in a mechanical system the forces from the environment influence the displacement of an object, but on the other hand the displacement of the object influences the forces between the object and environment. What is input and what is output in this case is primarily a choice by the observer, guided by what is interesting to study, rather than a property of the system itself.

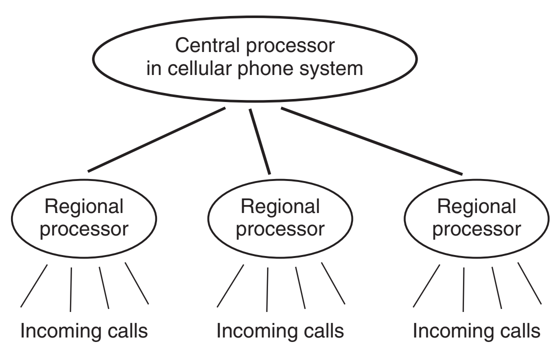

Figure 1.2 Cellular phone system containing a central processor and regional processors to handle incoming calls.

1.1.2 Experiments

Observability is essential in order to study a system according to our definition of system. We must at least be able to observe some outputs of a system. We can learn even more if it is possible to exercise a system by controlling its inputs. This process is called *experimentation*, that is:

- An *experiment* is the process of extracting information from a system by exercising its inputs.

To perform an experiment on a system, it must be both controllable and observable. We apply a set of external conditions to the accessible inputs and observe the reaction of the system by measuring the accessible outputs.

One of the disadvantages of the experimental method is that for a large number of systems many inputs are not accessible and controllable. These systems are under the influence of inaccessible inputs, sometimes called *disturbance inputs*. Likewise, it is often the case that many really useful possible outputs are not accessible for measurements; these are sometimes called *internal states* of the system. There are also a number of practical problems associated with performing an experiment, for example:

- The experiment might be too *expensive*: Investigating ship durability by building ships and letting them collide is a very expensive method of gaining information.
- The experiment might be too *dangerous*: Training nuclear plant operators in handling dangerous situations by letting the nuclear reactor enter hazardous states is not advisable.
- The *system* needed for the experiment might *not yet exist*. This is typical of systems to be designed or manufactured.

The shortcomings of the experimental method led us to the model concept. If we make a model of a system, this model can be investigated and may answer many questions regarding the real system if the model is realistic enough.

1.2 THE MODEL CONCEPT

Given the previous definitions of system and experiment, we can now attempt to define the notion of model:

- A *model* of a system is anything an "experiment" can be applied to in order to answer questions about that *system*.

This implies that a model can be used to answer questions about a system *without* doing experiments on the *real* system. Instead we perform simplified "experiments" on the model, which in turn can be regarded as a kind of simplified system that reflects properties of the real system. In the simplest case a model can just be a piece of information that is used to answer questions about the system.

Given this definition, any model also qualifies as a system. Models, just like systems, are hierarchical in nature. We can cut out a piece of a model, which becomes a new model that is valid for a subset of the experiments for which the original model is valid. A model is always related to the system it models and the experiments to which it can be subjected. A statement such as "a model of a system is invalid" is meaningless without mentioning the associated system and the experiment. A model of a system might be valid for one experiment on the model and invalid for another. The term model *validation*, see Section 1.5.3, always refers to an experiment or a class of experiment to be performed.

We talk about different kinds of models depending on how the model is represented:

- *Mental* model—a statement like "a person is reliable" helps us answer questions about that person's behavior in various situations.
- *Verbal* model—this kind of model is expressed in words. For example, the sentence "More accidents will occur if the speed limit is increased" is an example of a verbal model. Expert systems is a technology for formalizing verbal models.
- *Physical* model—this is a physical object that mimics some properties of a real system, to help us answer questions about that system. For example, during design of artifacts such as

buildings, airplanes, and so forth, it is common to construct small physical models with the same shape and appearance as the real objects to be studied, for example, with respect to their aerodynamic properties and aesthetics.

- *Mathematical* model—a description of a system where the relationships between variables of the system are expressed in mathematical form. Variables can be measurable quantities such as size, length, weight, temperature, unemployment level, information flow, bit rate, and so forth. Most laws of nature are mathematical models in this sense. For example, Ohm's law describes the relationship between current and voltage for a resistor; Newton's laws describe relationships between velocity, acceleration, mass, force, and the like.

The kinds of models that we primarily deal with in this book are mathematical models represented in various ways, for example, as equations, functions, computer programs, and the like. Artifacts represented by mathematical models in a computer are often called *virtual prototypes*. The process of constructing and investigating such models is virtual prototyping. Sometimes the term *physical modeling* is used also for the process of building mathematical models of physical systems in the computer if the structuring and synthesis process is the same as when building real physical models.

1.3 SIMULATION

In the previous section we mentioned the possibility of performing "experiments" on models instead of on the real systems corresponding to the models. This is actually one of the main uses of models, and is denoted by the term *simulation*, from the Latin *simulare*, which means to pretend. We define a simulation as follows:

- A *simulation* is an experiment performed on a model.

Analogous to our previous definition of *model*, this definition of simulation does not require the model to be represented in mathematical or computer program form. However, in the rest of this text we will concentrate on *mathematical models*, primarily those that have a computer-representable form. The following are a few examples of such experiments or simulations:

- A simulation of an industrial process such as steel or pulp manufacturing, to learn about the behavior under different operating conditions in order to improve the process.
- A simulation of vehicle behavior, for example, of a car or an airplane, for the purpose of providing realistic operator training.
- A simulation of a simplified model of a packet-switched computer network, to learn about its behavior under different loads in order to improve performance.

It is important to realize that the *experiment description* and *model description* parts of a simulation are conceptually separate entities. On the other hand, these two aspects of a simulation belong together even if they are separate. For example, a model is valid only for a certain class of experiments. It can be useful to define an *experimental frame* associated with the model, which defines the conditions that need to be fulfilled by valid experiments.

If the mathematical model is represented in executable form in a computer, simulations can be performed by *numerical experiments*, or in nonnumeric cases by *computed experiments*. This is a simple and safe way of performing experiments, with the added advantage that essentially all variables of the model are observable and controllable. However, the value of the simulation results is completely dependent on how well the model represents the real system regarding the questions to be answered by the simulation.

Except for experimentation, simulation is the only technique that is generally applicable for analysis of the behavior of arbitrary systems. Analytical techniques are better than simulation, but usually apply only under a set of simplifying assumptions, which often cannot be justified. On the other hand, it is not uncommon to combine analytical techniques with simulations, that is, simulation is used not alone but in an interplay with analytical or semianalytical techniques.

1.3.1 Reasons for Simulation

There are a number of good reasons to perform simulations instead of performing experiments on real systems:

- Experiments are too *expensive*, too *dangerous*, or the system to be investigated does *not yet exist*. These are the main difficulties of experimentation with real systems, previously mentioned in Section 1.1.2.
- The *time scale* of the dynamics of the system is not compatible with that of the experimenter. For example, it takes millions of years to observe small changes in the development of the universe, whereas similar changes can be quickly observed in a computer simulation of the universe.
- Variables may be *inaccessible*. In a simulation all variables can be studied and controlled, even those that are inaccessible in the real system.
- Easy *manipulation* of models. Using simulation, it is easy to manipulate the parameters of a system model, even outside the feasible range of a particular physical system. For example, the mass of a body in a computer-based simulation model can be increased from 40 to 500 kg at a keystroke, whereas this change might be hard to realize in the physical system.
- Suppression of *disturbances*. In a simulation of a model it is possible to suppress disturbances that might be unavoidable in measurements of the real system. This can allow us to isolate particular effects and thereby gain a better understanding of those effects.
- Suppression of *second-order effects*. Often, simulations are performed since they allow suppression of second-order effects such as small nonlinearities or other details of certain system components, which can help us to better understand the primary effects.

1.3.2 Dangers of Simulation

The ease of use of simulation is also its most serious drawback: It is quite easy for the user to forget the limitations and conditions under which a simulation is valid and therefore draw the wrong conclusions from the simulation. To reduce these dangers, one should always try to compare at least some results of simulating a model against experimental results from the real system. It also helps to be aware of the following three common sources of problems when using simulation:

- Falling in love with a model—the Pygmalion[1] effect. It is easy to become too enthusiastic about a model and forget all about the experimental frame, that is, that the model is not the real world but only represents the real system under certain conditions. One example is the introduction of foxes on the Australian continent to solve the rabbit problem, on the model assumption that foxes hunt rabbits, which is true in many other parts of the world. Unfortunately, the foxes found the indigenous fauna much easier to hunt and largely ignored the rabbits.
- Forcing reality into the constraints of a model—the Procrustes[2] effect. One example is the shaping of our societies after currently fashionable economic theories having a simplified view of reality, and ignoring many other important aspects of human behavior, society, and nature.
- Forgetting the model's level of accuracy. All models have simplifying assumptions, and we have to be aware of those in order to correctly interpret the results.

For these reasons, while analytical techniques are generally more restrictive since they have a much smaller domain of applicability, such techniques are more powerful when they apply. A simulation result is valid only for a particular set of input data. Many simulations are needed to gain an approximate understanding of a system. Therefore, if analytical techniques are applicable, they should be used instead of a simulation or as a complement.

1.4 BUILDING MODELS

Given the usefulness of simulation in order to study the behavior of systems, how do we go about building models of those systems? This

[1]Pygmalion is the mythical king of Cyprus who also was a sculptor. The king fell in love with one of his works, a sculpture of a young woman, and asked the gods to make her alive.

[2]Procrustes is a robber known from Greek mythology. He is known for the bed where he tortured travelers who fell into his hands: If the victim was too short, he stretched arms and legs until the person fit the length of the bed; if the victim was too tall, he cut off the head and part of the legs.

is the subject of most of this book and of the Modelica language, which has been created to simplify model construction as well as reuse of existing models.

There are in principle two main sources of general system-related knowledge needed for building mathematical models of systems:

- The collected *general experience* in relevant domains of science and technology, found in the literature and available from experts in these areas. This includes the *laws of nature*, for example, including Newton's laws for mechanical systems, Kirchhoff's laws for electrical systems, approximate relationships for nontechnical systems based on economic or sociological theories, and so on.
- The *system* itself, that is, observations of and experiments on the system we want to model.

In addition to the above system knowledge, there is also specialized knowledge about mechanisms for handling and using facts in model construction for specific applications and domains, as well as generic mechanisms for handling facts and models, that is:

- *Application expertise*—mastering the application area and techniques for using all facts relative to a specific modeling application.
- *Software and knowledge engineering*—generic knowledge about defining, handling, using, and representing models and software, for example, object orientation, component system techniques, expert system technology, and so on.

What is then an appropriate analysis and synthesis *process* to be used in applying these information sources for constructing system models? Generally, we first try to identify the main components of a system and the kinds of interaction between these components. Each component is broken down into subcomponents until each part fits the description of an existing model from some model library, or we can use appropriate laws of nature or other relationships to describe the behavior of that component. Then we state the component interfaces and make a mathematical formulation of the interactions between the components of the model.

Certain components might have unknown or partially known model parameters and coefficients. These can often be found by fitting experimental measurement data from the real system to the mathematical model using *system identification*, which in simple cases reduces to basic techniques like curve fitting and regression analysis. However, advanced versions of system identification may even determine the form of the mathematical model selected from a set of basic model structures.

1.5 ANALYZING MODELS

Simulation is one of the most common techniques for using models to answer questions about systems. However, there also exist other methods of analyzing models such as sensitivity analysis and model-based diagnosis or analytical mathematical techniques in the restricted cases where solutions can be found in a closed analytical form.

1.5.1 Sensitivity Analysis

Sensitivity analysis deals with the question how *sensitive* the behavior of the model is to *changes* of model parameters. This is a very common question in design and analysis of systems. For example, even in well-specified application domains such as electrical systems, resistor values in a circuit are typically known only by an accuracy of 5 to 10%. If there is a large sensitivity in the results of simulations to small variations in model parameters, we should be very suspicious about the validity the model. In such cases small random variations in the model parameters can lead to large random variations in the behavior.

On the other hand, if the simulated behavior is not very sensitive to small variations in the model parameters, there is a good chance that the model fairly accurately reflects the behavior of the real system. Such robustness in behavior is a desirable property when designing new products, since they otherwise may become expensive to manufacture since certain tolerances must be kept very small. However, there are also a number of examples of real systems which are very sensitive to variations of specific model parameters. In those cases that sensitivity should be reflected in models of those systems.

1.5.2 Model-Based Diagnosis

Model-based *diagnosis* is a technique somewhat related to sensitivity analysis. We want to find the causes of certain behavior of a system by analyzing a model of that system. In many cases we want to find the causes of problematic and erroneous behavior. For example, consider a car, which is a complex system consisting of many interacting parts such as a motor, an ignition system, a transmission system, suspension, wheels, and the like. Under a set of well-defined operating conditions each of these parts can be considered to exhibit a correct behavior if certain quantities are within specified value intervals. A measured or computed value outside such an interval might indicate an error in that component or in another part influencing that component. This kind of analysis is called model-based diagnosis.

1.5.3 Model Verification and Validation

We have previously remarked about the dangers of simulation, for example, when a model is not valid for a system regarding the intended simulation. How can we verify that the model is a good and reliable model, that is, is it valid for its intended use? This can be very hard, and sometimes we can hope only to get a partial answer to this question. However, the following techniques are useful to at least partially verify the validity of a model:

- Critically review the assumptions and approximations behind the model, including available information about the domain of validity regarding these assumptions.
- Compare simplified variants of the model to analytical solutions for special cases.
- Compare to experimental results for cases when this is possible.
- Perform sensitivity analysis of the model. If the simulation results are relatively insensitive to small variations of model parameters, we have stronger reasons to believe in the validity of the model.
- Perform internal consistency checking of the model, for example, checking that dimensions or units are compatible across equations. For example, in Newton's equation $F = ma$,

the unit (N) on the left-hand side is consistent with (kg m s^{-2}) on the right-hand side.

In the last case it is possible for tools to automatically verify that dimensions are consistent if unit attributes are available for the quantities of the model. This functionality, however, is not yet available for most current modeling tools.

1.6 KINDS OF MATHEMATICAL MODELS

Different kinds of mathematical models can be characterized by different properties reflecting the behavior of the systems that are modeled. One important aspect is whether the model incorporates *dynamic* time-dependent properties or is *static*. Another dividing line is between models that evolve *continuously* over time and those that change at *discrete* points in time. A third dividing line is between *quantitative* and *qualitative* models.

Certain models describe *physical distribution* of quantities, for example, mass, whereas other models are *lumped* in the sense that the physically distributed quantity is approximated by being lumped together and represented by a single variable, for example, a point mass.

Some phenomena in nature are conveniently described by stochastic processes and probability distributions, e.g. noisy radio transmissions or atomic-level quantum physics. Such models might be labeled *stochastic* or *probability*-based models where the behavior can be represented only in a statistic sense, whereas *deterministic* models allow the behavior to be represented without uncertainty. However, even stochastic models can be simulated in a “deterministic” way using a computer since the random number sequences often used to represent stochastic variables can be regenerated given the same seed values.

The same phenomenon can often be modeled as being either stochastic or deterministic depending on the level of detail at which it is studied. Certain aspects at one level are abstracted or averaged away at the next higher level. For example, consider the modeling of gases at different levels of detail starting at the quantum mechanical elementary particle level, where the positions of particles are described by probability distributions:

- Elementary particles (orbitals)—stochastic models
- Atoms (ideal gas model)—deterministic models
- Atom groups (statistical mechanics)—stochastic models
- Gas volumes (pressure and temperature)—deterministic models
- Real gases (turbulence)—stochastic models
- Ideal mixer (concentrations)—deterministic models

It is interesting to note the kinds of model changes between stochastic or deterministic models that occur depending on what aspects we want to study. Detailed stochastic models can be averaged as deterministic models when approximated at the next upper macroscopic level in the hierarchy. On the other hand, stochastic behavior such as turbulence can be introduced at macroscopic levels as the result of chaotic phenomena caused by interacting deterministic parts.

1.6.1 Kinds of Equations

Mathematical models usually contain equations. There are basically four main kinds of equations, where we give one example of each.

Differential equations contain time derivatives such as dx/dt, usually denoted $\dot{x}$, for example,

$$\dot{x} = a \cdot x + 3 \tag{1.1}$$

Algebraic equations do not include any differentiated variables:

$$x^2 + y^2 = L^2 \tag{1.2}$$

Partial differential equations also contain derivatives with respect to other variables than time:

$$\frac{\partial a}{\partial t} = \frac{\partial^2 a}{\partial z^2} \tag{1.3}$$

Difference equations express relations between variables, for example, at different points in time:

$$x(t+1) = 3x(t) + 2 \tag{1.4}$$

1.6.2 Dynamic Versus Static Models

All systems, both natural and man-made, are dynamic in the sense that they exist in the real world, which evolves in time. Mathematical models of such systems would be naturally viewed as *dynamic* in the sense that they evolve over time and therefore incorporate time. However, it is often useful to make the approximation of ignoring time dependence in a system. Such a system model is called *static*. Thus we can define the concepts of dynamic and static models as follows:

- A *dynamic* model includes *time* in the model. The word *dynamic* is derived from the Greek word *dynamis* meaning force and power, with dynamics being the (time-dependent) interplay between forces. Time can be included explicitly as a variable in a mathematical formula or be present indirectly, for example, through the time derivative of a variable or as events occurring at certain points in time.
- A *static* model can be defined *without* involving *time*, where the word *static* is derived from the Greek word *statikos*, meaning something that creates equilibrium. Static models are often used to describe systems in steady-state or equilibrium situations, where the output does not change if the input is the same. However, static models can display a rather dynamic behavior when fed with dynamic input signals.

It is usually the case that the behavior of a dynamic model is dependent on its *previous* simulation history. For example, the presence of a time derivative in a mathematical model means that this derivative needs to be integrated to solve for the corresponding variable when the model is simulated, that is, the *integration* operation takes the previous time history into account. This is the case, for example, for models of capacitors where the voltage over the capacitor is proportional to the accumulated charge in the capacitor, that is, integration/accumulation of the current through the capacitor. By differentiating that relation the time derivative of the capacitor voltage becomes proportional to the current through the capacitor. We can study the capacitor voltage increasing over time at a rate proportional to the current in Figure 1.3.

Another way for a model to be dependent on its previous history is to let preceding events influence the current state, for example, as in a

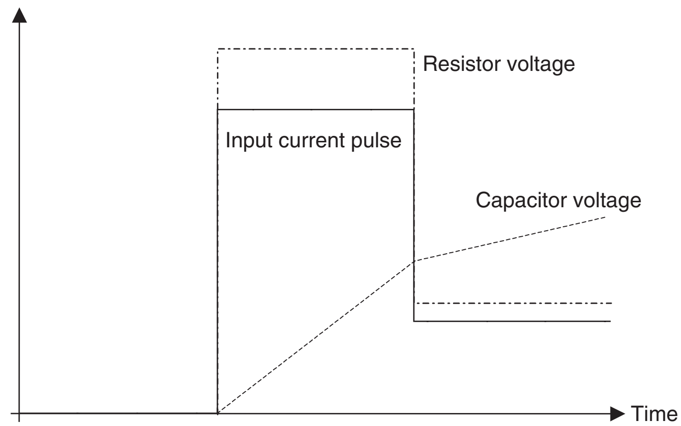

Figure 1.3 Resistor is a static system where the voltage is directly proportional to the current, independent of time, whereas a capacitor is a dynamic system where voltage is dependent on the previous time history.

model of an ecological system where the number of prey animals in the system will be influenced by events such as the birth of predators. On the other hand, a dynamic model such as a sinusoidal signal generator can be modeled by a formula directly including time and not involving the previous time history.

A resistor is an example of a static model that can be formulated without including time. The resistor voltage is directly proportional to the current through the resistor, for example, as depicted in Figure 1.3, with no dependence on time or on the previous history.

1.6.3 Continuous-Time Versus Discrete-Time Dynamic Models

There are two main classes of dynamic models: continuous-time and discrete-time models. The class of continuous-time models can be characterized as follows:

- *Continuous-time* models evolve their variable values continuously over time.

A variable from a continuous-time model A is depicted in Figure 1.4. The mathematical formulation of continuous-time models includes

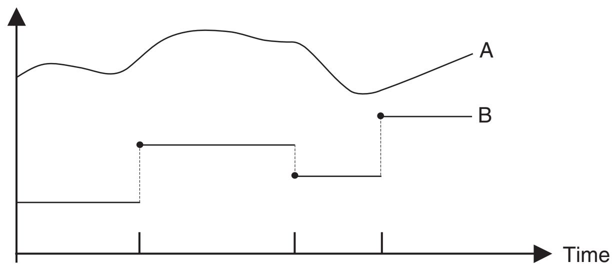

Figure 1.4 Discrete-time system B changes values only at certain points in time, whereas continuous-time systems like A evolve values continuously.

differential equations with time derivatives of some model variables. Many laws of nature, for example, as expressed in physics, are formulated as differential equations.

The second class of mathematical models is discrete-time models, for example, as B in Figure 1.4, where variables change value only at certain points in time:

- *Discrete-time* models may change their variable values only at discrete points in time.

Discrete-time models are often represented by sets of difference equations or as computer programs mapping the state of the model at one point in time to the state at the next point in time.

Discrete-time models occur frequently in engineering systems, especially computer-controlled systems. A common special case is sampled systems, where a continuous-time system is measured at regular time intervals and is approximated by a discrete-time model. Such sampled models usually interact with other discrete-time systems like computers. Discrete-time models may also occur naturally, for example, an insect population which breeds during a short period once a year; that is, the discretization period in that case is one year.

1.6.4 Quantitative Versus Qualitative Models

All of the different kinds of mathematical models previously discussed in this section are of a quantitative nature—variable values can be represented numerically according to a quantitatively measurable scale.

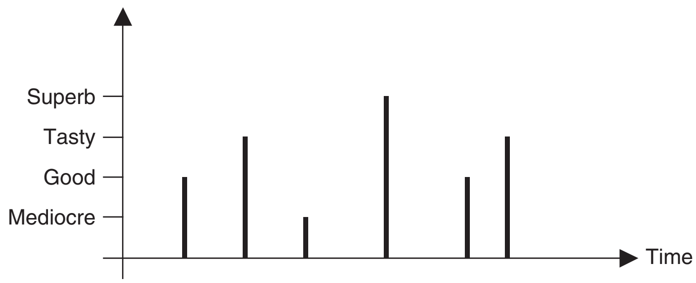

Figure 1.5 Quality of food in a restaurant according to inspections at irregular points in time.

Other models, so-called *qualitative* models, lack that kind of precision. The best we can hope for is a rough classification into a finite set of values, for example, as in the food quality model depicted in Figure 1.5. Qualitative models are by nature discrete-time models, and the dependent variables are also discretized. However, even if the discrete values are represented by numbers in the computer (e.g., mediocre—1, good—2, tasty—3, superb—4), we have to be aware of the fact that the values of variables in certain qualitative models are not necessarily according to a linear measurable scale, that is, tasty might not be three times better than mediocre.

1.7 USING MODELING AND SIMULATION IN PRODUCT DESIGN

What role does modeling and simulation have in industrial product design and development? In fact, our previous discussion has already briefly touched this issue. Building mathematical models in the computer, so-called *virtual prototypes*, and simulating those models, is a way to quickly determine and optimize product properties without building costly physical prototypes. Such an approach can often drastically reduce development time and time to market, while increasing the quality of the designed product.

The so-called product *design V*, depicted in Figure 1.6, includes all the standard phases of product development:

- Requirements analysis and specification
- System design

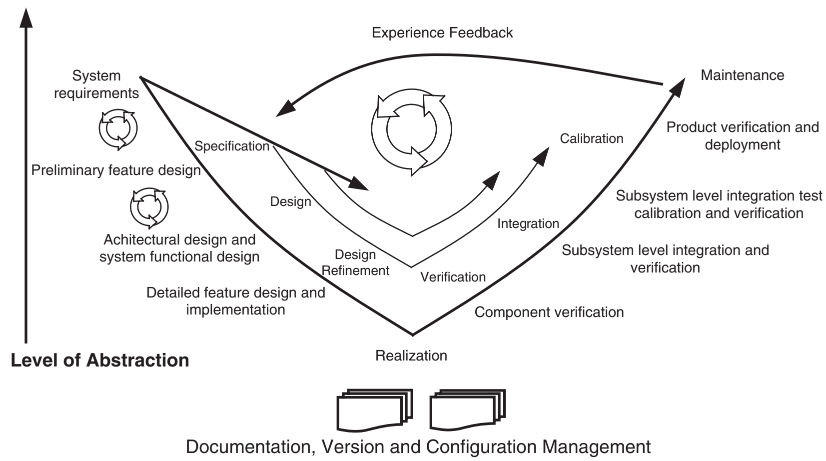

Figure 1.6 Product design V.

- Design refinement
- Realization and implementation
- Subsystem verification and validation
- Integration
- System calibration and model validation
- Product deployment

How does modeling and simulation fit into this design process?

In the first phase, *requirements analysis*, functional and nonfunctional requirements are specified. In this phase important design parameters are identified and requirements on their values are specified. For example, when designing a car, there might be requirements on acceleration, fuel consumption, maximum emissions, and the like. Those system parameters will also become parameters in our model of the designed product.

In the *system design phase* we specify the architecture of the system, that is, the main components in the system and their interactions. If we have a simulation model component library at hand, we can use these library components in the design phase or otherwise create new components that fit the designed product. This design process iteratively increases the level of detail in the design. A modeling tool that supports hierarchical system modeling and decomposition can help in handling system complexity.

The *implementation phase* will realize the product as a physical system and/or as a virtual prototype model in the computer. Here a virtual prototype can be realized before the physical prototype is built, usually for a small fraction of the cost.

In the *subsystem verification and validation phase*, the behavior of the subsystems of the product is verified. The subsystem virtual prototypes can be simulated in the computer and the models corrected if there are problems.

In the *integration phase* the subsystems are connected. Regarding a computer-based system model, the models of the subsystems are connected together in an appropriate way. The whole system can then be simulated, and certain design problems corrected based on the simulation results.

The system and model *calibration and validation phase* validates the model against measurements from appropriate physical prototypes. Design parameters are calibrated, and the design is often *optimized* to a certain extent according to what is specified in the original requirements.

During the last phase, *product deployment*, which usually only applies to the physical version of the product, the product is deployed and sent to the customer for feedback. In certain cases this can also be applied to virtual prototypes, which can be delivered and put in a computer that is interacting with the rest of the customer physical system in real time, that is, hardware-in-the-loop simulation.

In most cases, experience feedback can be used to tune both models and physical products. All phases of the design process continuously interact with the model and design database, as depicted at the bottom of Figure 1.6.

1.8 EXAMPLES OF SYSTEM MODELS

In this section we briefly present examples of mathematical models from three different application areas, in order to illustrate the power of the Modelica mathematical modeling and simulation technology to be described in the rest of this book:

- A thermodynamic system—part of an industrial GTX100 gas turbine model
- A three-dimensional (3D) mechanical system with a hierarchical decomposition—an industry robot
- A biochemical application—part of the citrate cycle (TCA cycle), see Figure 1.11

A connection diagram of the power cutoff mechanism of the GTX100 gas turbine is depicted in Figure 1.8, whereas the gas turbine itself is shown in Figure 1.7.

The connection diagram in Figure 1.8 might not appear as a mathematical model, but behind each icon in the diagram is a model component containing the equations that describe the behavior of the respective component.

In Figure 1.9 we show a few plots from simulations of the gas turbine, which illustrates how a model can be used to investigate the properties of a given system.

The second example, the industry robot, illustrates the power of hierarchical model decomposition. The 3D robot, shown to the right of Figure 1.10, is represented by a two-dimensional (2D) connection diagram (in the middle). Each part in the connection diagram can be a mechanical component such as a motor or joint, a control system for the robot, and so forth. Components may consist of other components that can in turn be decomposed. At the bottom of the hierarchy wehave model classes containing the actual equations.

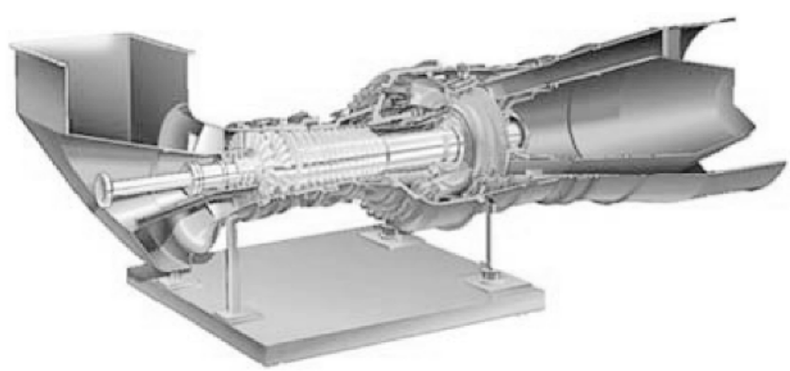

Figure 1.7 Schematic picture of the gas turbine GTX100. (Courtesy Siemens Industrial Turbomachinery AB, Finspång, Sweden.)

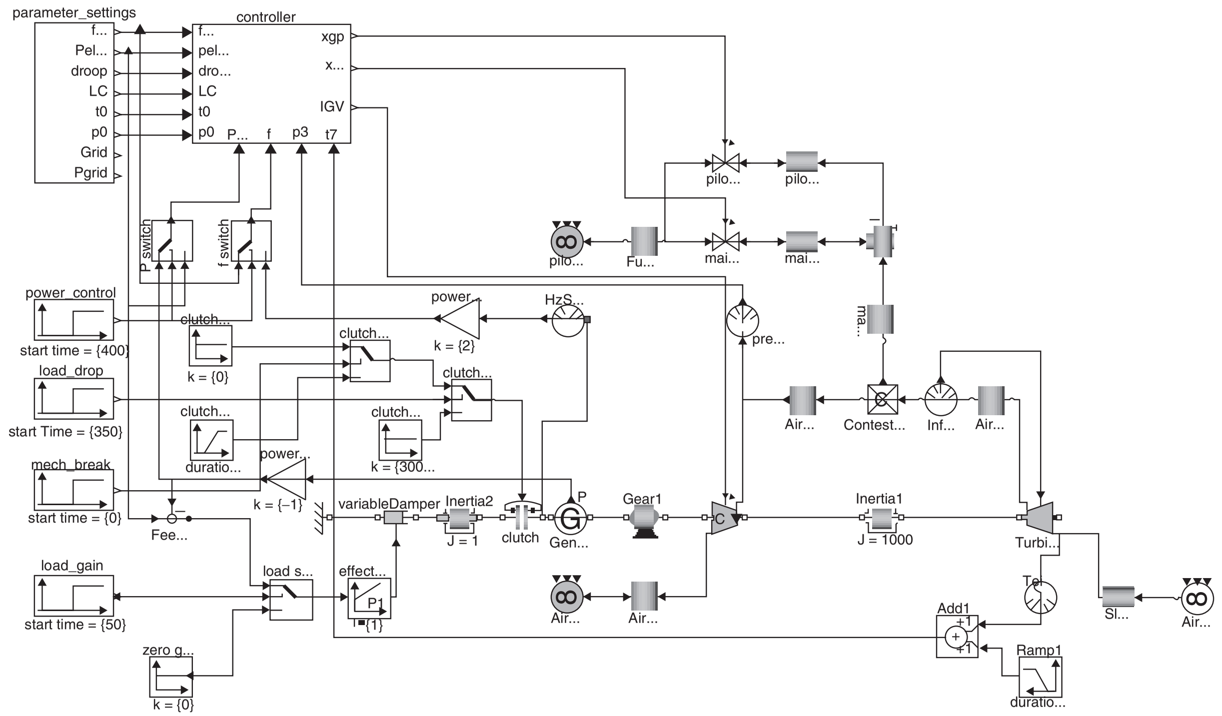

Figure 1.8 Detail of power cutoff mechanism in 40 MW GTX100 gas turbine model. (Courtesy Siemens Industrial Turbomachinery AB, Finspång, Sweden.)

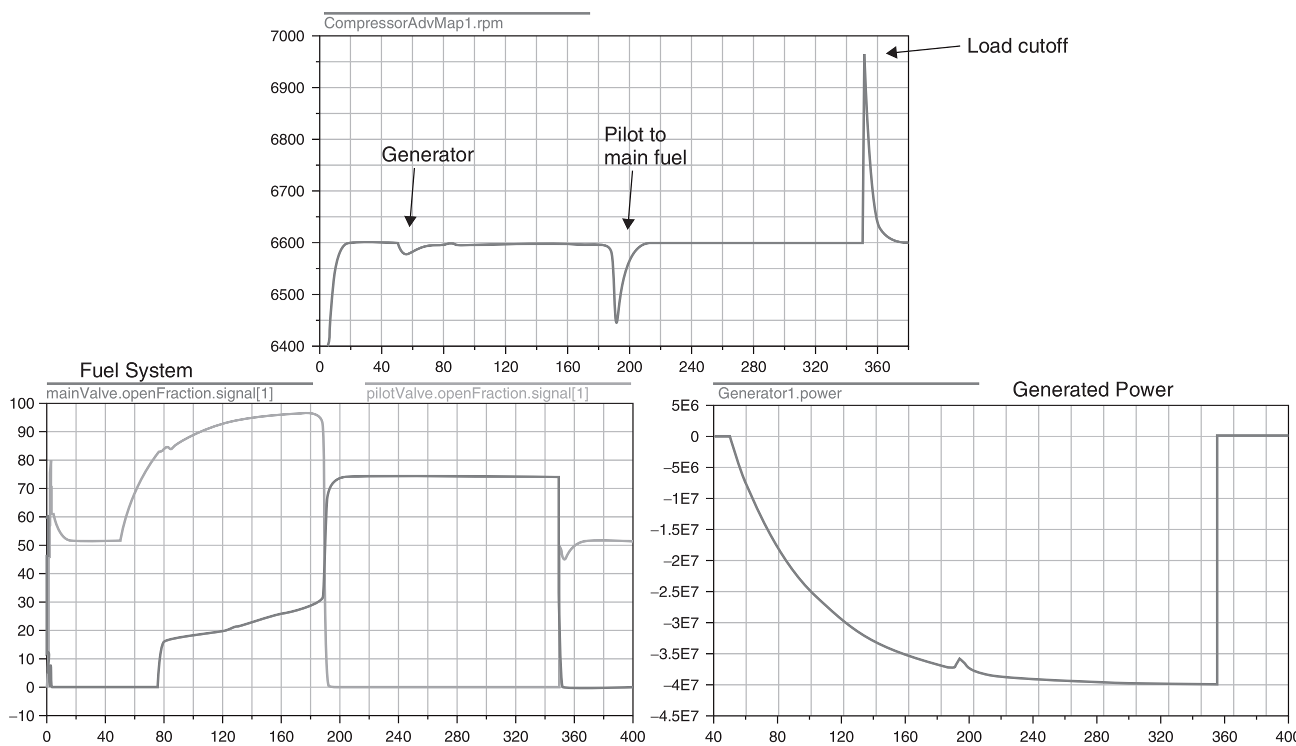

Figure 1.9 Simulation of GTX100 gas turbine power system cutoff mechanism. (Courtesy Alstom Industrial Turbines AB, Finspång, Sweden.)

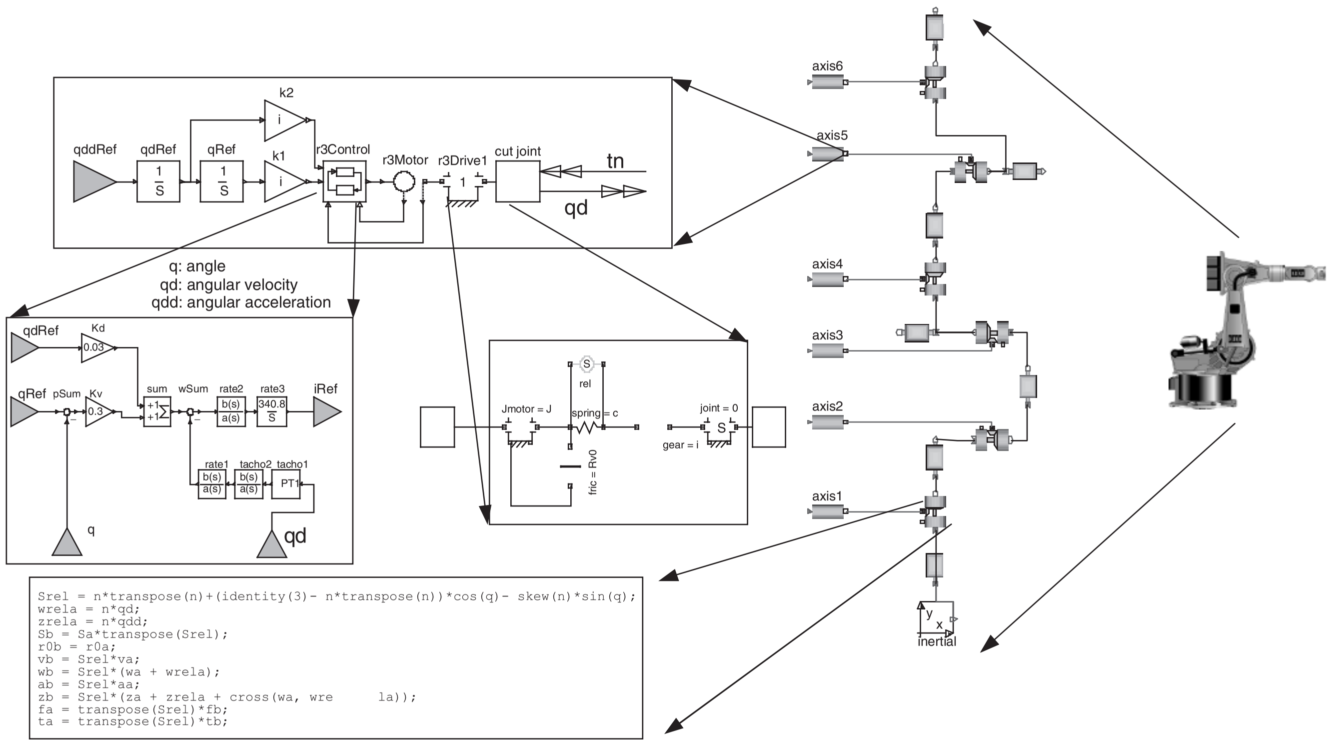

Figure 1.10 Hierarchical model of an industrial robot. (Courtesy Martin Otter.)

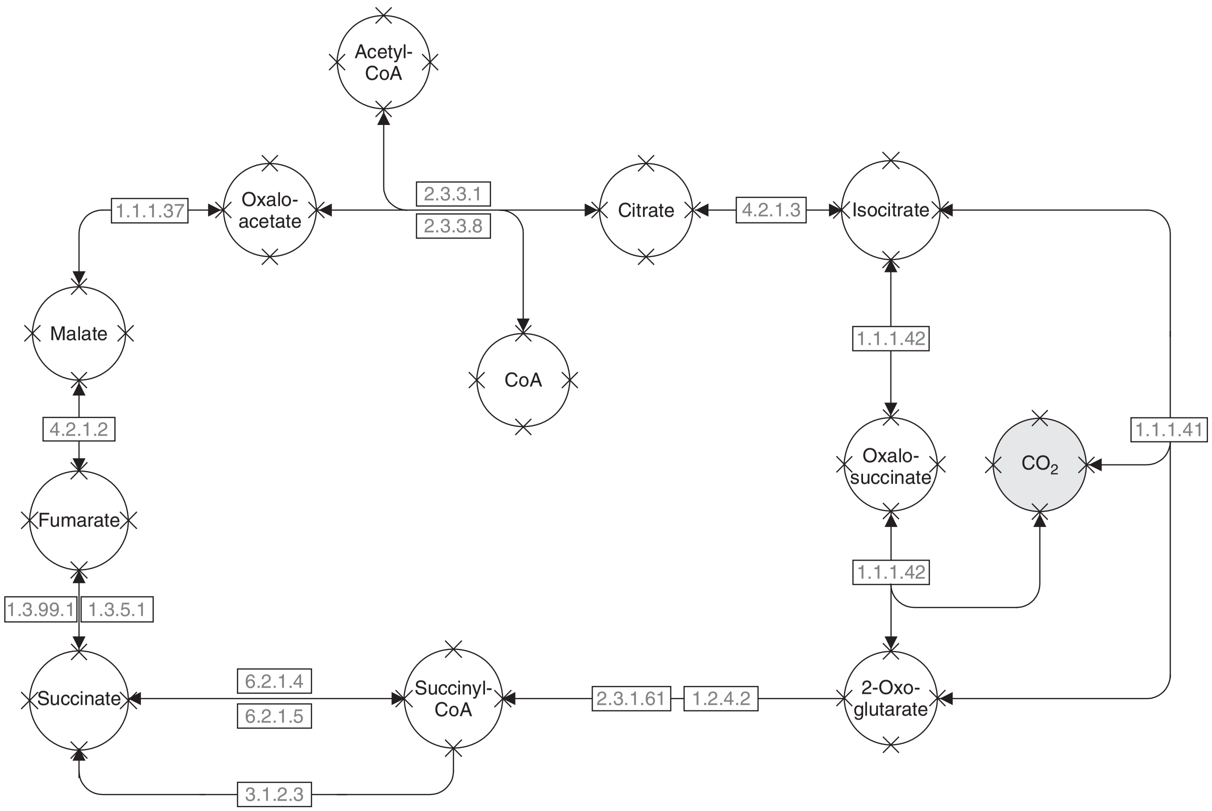

Figure 1.11 Biochemical pathway model of part of the citrate cycle (TCA cycle).

The third example is from an entirely different domain—biochemical pathways describing the reactions between reactants, in this particular case describing part of the citrate cycle (TCA cycle) as depicted in Figure 1.11.

1.9 SUMMARY

We have briefly presented important concepts such as system, model, experiment, and simulation. Systems can be represented by models, which can be subject to experiments, that is, simulation. Certain models can be represented by mathematics, so-called mathematical models. This book is about object-oriented component-based technology for building and simulating such mathematical models. There are different classes of mathematical models, for example, static versus dynamic models, continuous-time versus discrete-time models, and so forth, depending on the properties of the modeled system, the available information about the system, and the approximations made in the model.

1.10 LITERATURE

Any book on modeling and simulation needs to define fundamental concepts such as system, model, and experiment. The definitions in this chapter are generally available in modeling and simulation literature, including Ljung and Glad (1994) and Cellier (1991). The example of different levels of details in mathematical models of gases presented in Section 1.6 is mentioned in Hyötyniemi (2002). The product design-V process mentioned in Section 1.7 is described in Stevens et al. (1998) and Shumate and Keller (1992). The citrate cycle biochemical pathway part in Figure 1.11 is modeled after the description in Allaby (1998).