LA-UR-17-22089

Approved for public release; distribution is unlimited.

Title:	Notes on modeling and simulation
Author(s):	Redondo, Antonio
Intended for:	Pedagogical notes for new students and postdocs
Issued:	2017-03-10

Notes on modeling and simulation

Antonio Redondo

Theory, Simulation and Computation Directorate,
Los Alamos National Laboratory,
Los Alamos, New Mexico, 87545

(Dated: March 10, 2017)

Abstract

These notes present a high-level overview of how modeling and simulation are carried out by practitioners. The discussion is of a general nature; no specific techniques are examined but the activities associated with all modeling and simulation approaches are briefly addressed. There is also a discussion of validation and verification and, at the end, a section on why modeling and simulation are useful.

I. INTRODUCTION

Practitioners of modeling and simulation are often asked questions like: can you explain to me how modeling and simulation works?, how do you verify that what you are modeling corresponds to reality?, if I change my experimental setup in a given way, can you use your model under those conditions? These notes are meant to try to answer some of those questions in a manner that is, hopefully, understandable to people with technical backgrounds that are not necessarily versed in the details of how we carry out modeling and simulation in practice.

The basic premise of modeling and simulation is that, given a system, one can develop a theoretical construct, which we call a *model*, that aims to describe the nature and some of the behavior of the system. In addition, it is also assumed that one can quantify the model using physical and mathematical principles, as well as algorithms or procedures, that help us simulate the operation of the model. In general, all models are approximations to the real system but, if the model is reasonably faithful, the simulation gives us information and insight that allows us to understand how different parts of the system may function. Most commonly, it helps us understand results of experiments or observations and to form a partial picture of the nature and behavior of the system.

Although modeling and simulation are used most extensively in engineering and the natural sciences, they are regularly applied to many other fields, such as social science and economics. Because of the variety of potential systems to which modeling and simulation can be applied there exist a myriad of approaches and techniques that theoreticians and modelers have developed over the years. To keep the discussion manageable, in these notes we have chosen to describe only the general activities that are common to all modeling and simulation approaches.

II. WHAT IS A MODEL?

There is a well-know joke about a physicist who is addressing an audience of biologists and explaining how he is modeling cows. He starts by saying, "Let us assume that a cow is a sphere ..." Most biologists will take this as a grossly oversimplified model of a cow. However, if what one is studying is the spatial distribution of cows in a field from photographs taken from an airplane at high altitude, this may be a legitimate model of a cow. This is because the object of study may be how the cows move in the field and what rough area they cover as they roam around. However, if what one is interested in is the details of the motion of the legs as the cow moves around, this spherical model of a cow misses the point completely.

This example tells us that one must be careful in choosing what our model should be. To explore this point in some more depth let us discuss what we mean by a 'model'.

In our context,[1] a *model* is a conceptual representation of an object, an entity, a phenomenon or a system that is used to develop understanding and insight about the actual object that it represents. The best models also help us predict future behavior of the object in question. All models are, at best, approximations to the systems they aim to describe. On a fundamental level, the development of scientific theories always starts with the development of models, however, we do not have a general or unified method to derive models so we must resort to a variety of approaches.

The models in which we are interested consist of at least two main parts: (i) the *physical model* of the system and (ii) the *mathematical or logical implementation* of the properties of the physical model. For example, a physical model of a piston driven by steam may consist of a hollow metallic cylinder open on one end, a cylindrical block of metal or piston head that can slide inside the containment vessel and the steam in the space between the piston head and the closed end of the hollow cylinder. The physical model usually will include details of the geometry and nature of the constitutive materials. The mathematical implementation may include quantitative expressions for the physical laws that describe the behavior of gases at a given temperature and pressure, the mathematical description of the friction between the piston head and the inner surface of the hollow cylinder and Newton's laws for the description of the motion of the piston head. The level of detail that one includes in either the physical model or the mathematical implementation depends on the questions one wants to address with the model.

The physical model gives us a 'picture' of the system in question; the mathematical implementation may require the extension of theoretical constructs developed in other areas to the particular problem at hand; it also allows us to go to the next step, that is, the simulation step.

III. SIMULATION

Once we have determined the physical model and its mathematical or logical implementation we are ready to carry out simulations. A *simulation* is usually carried out in a computer, for which we must write code that implements the mathematical expressions that describe the properties of the model. Thus, computer simulations aim to reproduce the behavior of the model in a quantitative manner. The computer programs that carry out the simulations involve large-scale computation, possible only in the largest computers, or small calculations that can be run on laptop or desktop computers.

All simulations require *input data* specifying the initial conditions. These often include details about the geometry of the physical model, composition of its components, initial state of the different parts of the system (e.g., initial positions and velocities of all particles), environmental physical conditions, such as temperature and pressure, and all other quantities that are needed to specify the initial characteristics of the model. The sources of the input data may be obtained from experiments, sensors or other devices that are associated with the model, from geometrical constraints that limit the extension of space in which the physical model is located, or from the results of previous simulations.

While the computer program that implements the simulation runs, many intermediate quantities are calculated and stored. Some of these quantities are only necessary for intermediate steps and are eventually discarded after they have been used. Other quantities form part of the *output* and are stored or displayed for further use in the interpretation of the results of the simulation.

The components of a simulation are shown schematically in Fig. 1.

In a practical vein, computer simulation may be regarded[2] as the exercise of solving numerically an initial-value or boundary-value problem. At time $t = 0$ the initial state of the system is specified in some finite region of space—*the computational box*—on the surface of which prescribed boundary conditions hold. The simulation consist of following the evolution of the configuration, either temporally or to reach a steady or a prescribed state.

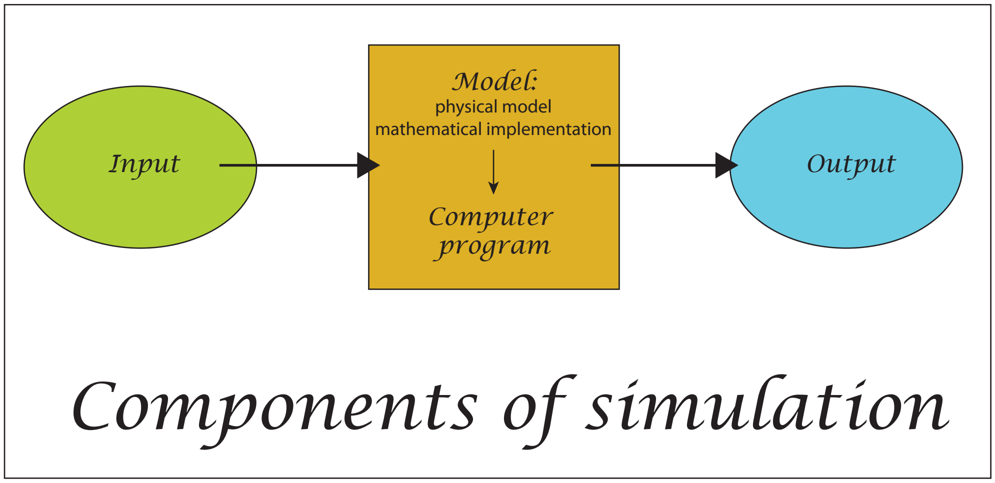

FIG. 1: Components of simulation.

IV. CODE CONSTRUCTION

The construction of the code that implements a model, in its most complete form, consists of the following activities:[3]

- Problem definition
- Requirements analysis
- Implementation planning
- High level design or architecture
- Detailed design
- Construction or implementation
- Integration
- Unit testing
- System testing
- Corrective maintenance
- Functional enhancements

This is a complicated formal process that is only used for codes that involve large numbers of users, such as the development of operating systems, like Windows and Mac OSX. In practical research situations, most of these steps are collapsed into a smaller number of activities. In most cases, all steps above 'detailed design' are part of the development of the model. Most computational scientists collapse the other activities into what is commonly referred to as *programming* and *validation and verification*. Programming, also known as 'coding and debugging', involves tasks such as designing routines and modules and finding and fixing errors.

In many research projects one can use software that is already available, either commercially or in the open domain. This is the case, for example, with engineering codes that carry out finite element analysis. There are also computer packages for carrying out molecular dynamics simulations that are in the open domain and are free of charge. These codes have limitations determined by the level of flexibility that their authors have incorporated in the releases available to the public. For example, although there are a number of commercially available codes to carry out computational fluid dynamics, most of them do not have recent developments that have appeared in the research literature. In fact, such developments are only included in these codes after a significant length of time, when the commercial outfits that release them are confident that the expense of incorporating them into their code will be justified by a return on their investment.

In a large number of research projects where modeling and simulation is necessary, there are often special conditions and requirements that make the use of commercial or open domain codes impractical. In these situations, we have to develop new codes or modify old ones. These new codes can occasionally be used in conjunction with some of the commercial codes, but this usually requires that the new code be adapted to interface smoothly.

V. VALIDATION AND VERIFICATION

Validation and verification are not the same thing but they are often confused. *Validation* is the process by which we ascertain that the simulation code delivers its intended purpose, that is, that it calculates the properties of the model in a faithful manner. *Verification* ensures that the implementation of the model is mathematically and algorithmically correct. In the context of code development, validation and verification are often referred to as 'software quality control'.

When we validate a simulation code we endeavor to make sure that the appropriate physical science principles, those that are present in the model, are correctly incorporated in the mathematical implementation. Suppose that the model includes the solution of Newton's second law for the simultaneous motion of a set of atoms or molecules. Then, the validation of the code needs to ascertain first that all the differential equations corresponding to Newton's second law for the system are correctly included and, second, that the quantities they will deliver are enough to calculate the properties that we expect to observe in the model.

To validate a code one must review and test each routine individually against known results, either from experiment or from analytical calculations. For example, in the case of the solution of Newton's equations of motion, the routines that carry out the integration of the differential equations can be tested against the solution of the gravitational motion of two or three bodies, by calculating the pressure-temperature relationship in a gas, or by many other problems whose solutions are well known.

Verification means that we must ensure that the mathematical operations and algorithms are carried out correctly. This includes checking that the coded formulas are free of mistakes. Often, a small mistake, for example the transposition of two numbers in the value of a physical parameter (e.g., Boltzmann's constant in the expression for the rate of a reaction), may lead to a small change in the value of a calculated quantity (e.g., the concentration of the products of the reaction as a function of time). This may not be a fundamental flaw if we just follow the process for a short time because the errors may be in the last decimal place of the quantity we are calculating. However, this could be very significant if the calculation involves long times in which the small errors may accumulate many times.

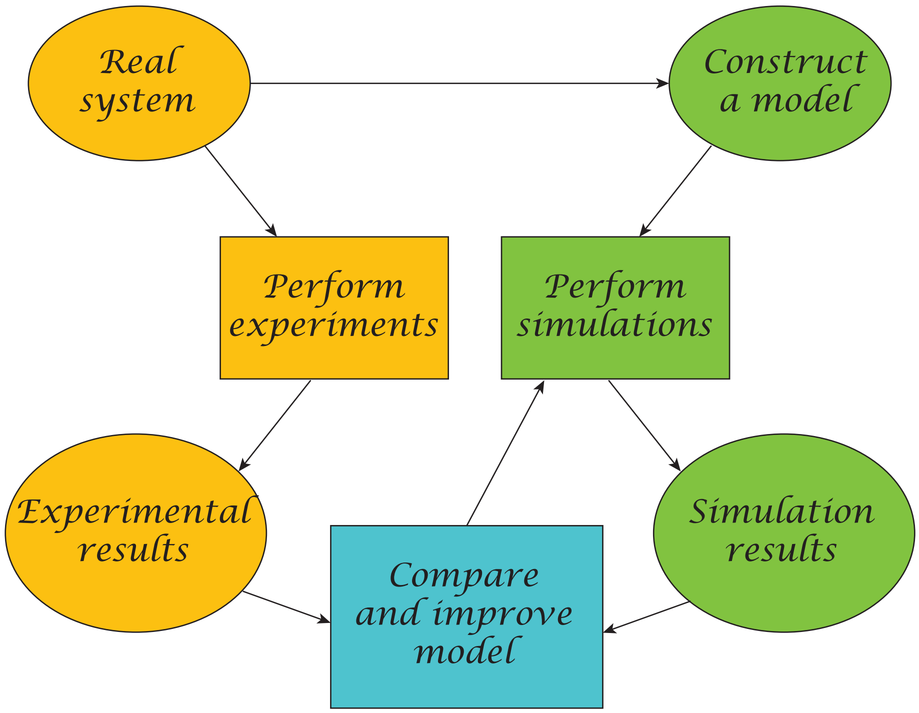

FIG. 2: Interaction between modeling and simulation and experiment.

Verification can also be carried out one routine at a time. In a manner similar to that of validation, one must compare the numerical or algorithmic solution of a known problem with experiment, with the exact answer or with results of other codes that have been validated and verified over a long period of time. Fortunately, many of the numerical procedures we use for the integration of differential equations and other operations have been thoroughly vetted and have been tested over and over. Many of these routines are in the open literature[4] and can be directly implemented in any new codes we may write to carry out simulations. Of course, one must be careful that in copying the routines to the new simulation codes one does not introduce typing mistakes that may lead to the problems outlined above.

VI. SYSTEM TESTING

The validation and verification described above is usually done one or a few routines at a time. This, however, does not guarantee that the complete code will function correctly. To do this, one must do *system testing*, which is the testing of the whole, assembled code, to make sure that it produces correct results. This is normally done by comparison to experiment. In this case, the simulation code is run under conditions that reproduce the condition of a predetermined experiment. The results of the simulation are compared to the experimental results. Often, particularly at the beginning of the modeling and simulation process, the simulation results show discrepancies with the experimental results. Assuming that the experiments have been done correctly and that they have the same conditions included in the simulation, the discrepancies are used to improve the model. A new set of simulations is then done to compare with the experiments and the process is repeated until the model is deemed to be verified and validated to a sufficient level of accuracy and faithfulness.

The interaction between modeling and simulation and experiment is displayed in Fig. 2.

VII. WHY MODELING AND SIMULATION?

In the late 1980's a US funding agency started a modeling and simulation program called 'Materials by Design'. A number of materials modelers, the author of these notes included, thought that this was a poor choice for a name of a program that had significant visibility in the materials science and condensed matter communities. The reason for this apprehension was that many people that did not have a background in materials modeling got the impression that the goal of the program was to produce a code that, given a list of desired properties for a new material, would allow us to sit down in front of a computer and design the material from scratch. Unfortunately, as many of us knew, at that time we were far from achieving this goal. This produced a lot of discontent, particularly among experimentalists that were not familiar with modeling, which made it more difficult to justify the use of modeling and simulation in advancing the science of materials.

Today, about 30 years later, although we have made much progress, we are still not capable of designing a material from scratch in a computer. Nevertheless, modeling and simulation are now completely accepted as absolutely essential for the advancement of the science and technology of materials. So, what is it that makes modeling and simulation important enough to be considered on a par with experimentation?

There are at least five fundamental roles for modeling and simulation in modern science and technology:

- To attain a fundamental understanding about the problem at hand,
- To help design and analyze experiments,
- To reduce the parameter space that needs to be considered in experiments,
- As an additional scientific tool to solve problems,
- To predict the future behavior of systems.

Modeling and simulation can provide a fundamental understanding of the systems we are studying. This is because new concepts are often produced during the process of developing new models. Such new concepts often lead to the proposal of new experiments. Of course, the advancements benefit most from a strong interaction between the experimentalists and the modelers.

In addition to producing new ideas for future experiments, modeling and simulation are often crucial in the interpretation of experimental data. This is particularly poignant when several experimental techniques are applied to the same system. The development of a successful model that can explain the interplay between the results of the different experiments leads to a deeper understanding of the system and its behavior.

Modeling and simulation are also very useful when a series of experimental parameters must be explored to achieve optimal results. This is important when, for example, one is trying to design a new catalyst to accelerate a particular chemical reaction. The traditional approach used to be to ask a scientist with 30 years of experience making catalysts—probably because they knew better than the early career people—to try many different compositions in as many different trials for the reaction. While there is no doubt that this approach has had many successes, it is very expensive and time consuming, particularly when the number of parameters in the composition space to be explored is large. In this case, modeling and simulation can be enormously productive tools to reduce the parameter space that must be explored, significantly reducing the cost of the experiments.

Modeling and simulation are also important scientific tools in their own right. One of the reasons for this is that modeling and simulation can explore phenomena, as well as spatial and temporal regions, that are inaccessible to experimental techniques. This may mean, for instance, the analysis of atomic and molecular scale effects in a solid that cannot be probed with current experimental techniques or phenomena that occur in times that are observable experimentally, like the determination of the intermediate species in a sequence of chemical reactions.

Finally, most mature models are predictive. This means that they can be used to study the effect of external conditions that have not been tried experimentally. In chemical problems, mature models can tell us the effect produced by the introduction of new molecules, such as inhibitors or catalysts. Predictive models can also be used to produce new ideas for production and manufacture of products.

For all these reasons, modeling and simulation have become an integral part of the modern toolset of scientists, engineers and technologists.

---

[1] The word 'model' is used in a different sense in biology and medicine. There, a model is usually an animal whose response to disease or therapeutic measures is thought to be similar or representative of the response that humans would have to the same process. For example, laboratory mice and rats are often used as models to test vaccines and medication.

[2] R W Hockney and J W Eastwood, *Computer Simulation Using Particles* (Adam Hilger, Bristol, 1988).

[3] S McConnell, *Code Complete. A Practical Handbook of Software Construction* (Microsoft Press, Redmond, Washington, 1993).

[4] See, for example, W H Press, S A Teukolsky, W T Vetterling and B P Flannery, *Numerical Recipes. The Art of Scientific Computing* (Cambridge University Press, Cambridge, 2007), or D E Knuth, *The Art of Computer Programming, Vols. 1-3* (Addison-Wesley, Reading, Massachusetts, 1997, 1998, 1998).