UNIT 7. VERIFICATION AND VALIDATION

One of the most important and difficult tasks faced by a model developer is the verification and validation of the simulation model. Conceptually, the verification and validation process consists of the following components:

- **Verification**: It is concerned with building the ***model right***. It is utilized in the comparison of the conceptual model to the computer representation that implements that conception. It asks the questions: Is the model implemented correctly in the computer? Are the input parameters and logical structure of the model correctly represented?
- **Validation**: It is concerned with building the ***right model***. It is utilized to determine that a model is an accurate representation of the real system. Validation is usually achieved through the calibration of the model, an iterative process of comparing the model to actual system behavior and using the discrepancies between the two, and the insights gained, to improve the model. This process is repeated until model accuracy is judged to be acceptable.

7.1 MODEL BUILDING, VERIFICATION AND VALIDATION

There are three major steps in model building as shown in Figure 7.1.

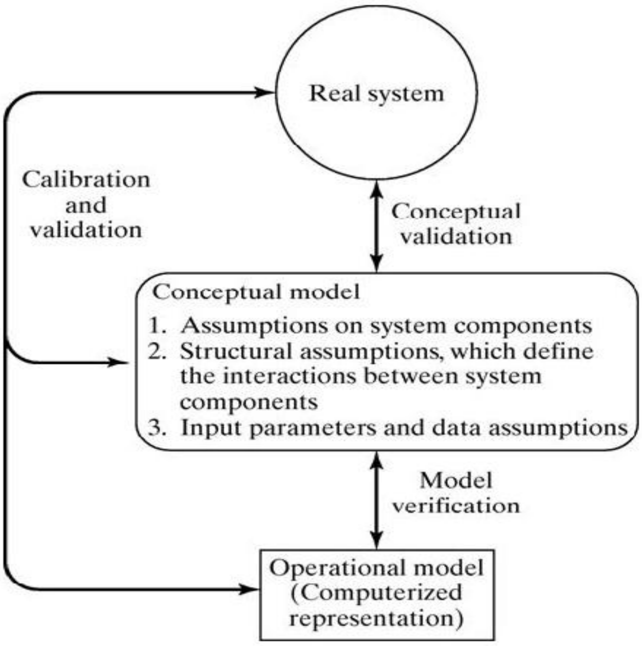

Figure 7.1 Model building, verification and validation

The steps are:

1. Observing the real system and the interactions among its various components and of collecting data on their behavior – As observation alone do not help in understanding the system behavior, the people who are familiar with the system have to be questioned to gain their special knowledge.

2. Construction of a conceptual model – a collection of assumptions about the components, the structure of the system and hypotheses about the values of the model parameters have to be made.
3. Implementation of an operational model – Using simulation software and by incorporating the assumptions of the conceptual model into the worldview.

7.2 VERIFICATION OF SIMULATION MODELS

The purpose of model verification is to assume that the conceptual model is reflected accurately in the operational model. Many suggestions can be given for use in the verification process –

1. Have the computerized representation checked by someone other than its developer.
2. Make a flow diagram which includes each logically possible action a system can take when an event occurs, and follow the model logic for each action for each event type.
3. Closely examine the model output for reasonableness under a variety of settings of input parameters.
4. Have the computerized representation print the input parameters at the end of the simulation to be sure that these parameter values have not been changed inadvertently.
5. Make the computerized representation of self-documenting as possible.
6. If the computerized representation is animated, verify that what is seen in the animation imitates the actual system.
7. The interactive run controller (IRC) or debugger is an essential component of successful simulation model building. Even the best of simulation analysts makes mistakes or commits logical errors when building a model. The IRC assists in finding and correcting those errors in the follow ways:
    a. The simulation can be monitored as it progresses.
    b. Attention can be focused on a particular line of logic or multiple lines of logic that constitute a procedure or a particular entity.
    c. Values of selected model components can be observed. When the simulation has paused, the current value or status of variables, attributes, queues, resources, counters, etc., can be observed.
    d. The simulation can be temporarily suspended, or paused, not only to view information but also to reassign values or redirect entities.
8. Graphical interfaces are recommended for accomplishing verification & validation.

7.3 CALIBRATION AND VALIDATION OF MODELS

Verification and validation although are conceptually distinct, usually are conducted simultaneously by the modeler. Validation is the ***overall process*** of comparing the model and its behavior to the real system and its behavior. Calibration is the ***iterative process*** of comparing the model to the real system, making adjustments to the model, comparing again and so on. Figure 7.2 shows the relationship of the model calibration to the overall validation process.

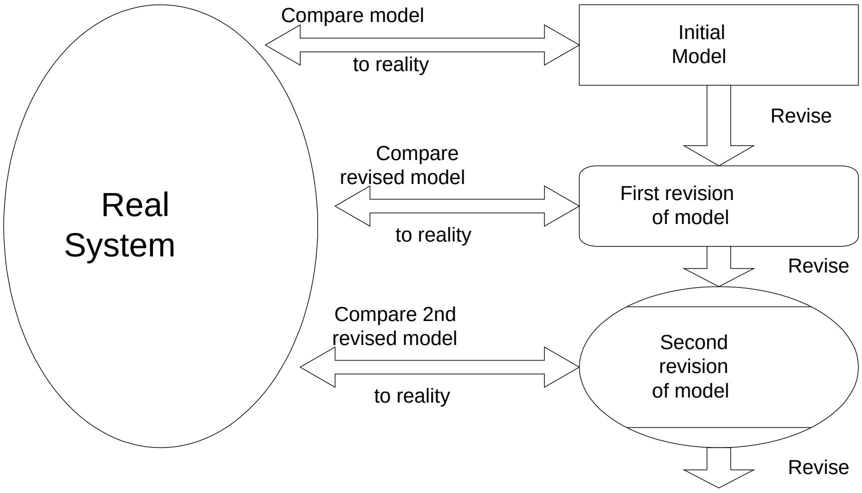

Figure 7.2 Iterative process of calibrating a model

The comparison of the model to reality is carried out by either subject or object tests. ***Subjective tests*** usually involve people who have knowledge about the system, making judgments about the model and its output. ***Objective tests*** require data on the behavior of the system and the corresponding data produced by the model. Then statistical tests are performed to compare few aspects of the system data set with the same aspect of the model data set.

As an aid in the validation process, **Naylor and Finger** formulated a three-step approach which has been widely followed:

- **Build a model that has high face validity**
- **Validate model assumptions**
- **Compare the model input-output transformations to corresponding input-output transformations for the real system**

These three steps are explained hereunder.

1. **Build a model that has high face validity:** The first goal of the simulation modeler is to construct a model that appears reasonable on its face to model users and others who are knowledgeable about the real system being simulated. The potential users of a model should be involved in model construction from its conceptualization to its implementation. So that, they can give useful input during the model development. Sensitivity analysis can also be used to check the face validity of the model. The model user is asked whether the model behaves in the expected way, when one or more variables are changed. From the experience and from the observations on the real system, the model user and mode builder would have an idea about effect of changes in the model.

2. **Validation of model assumptions:** Model assumptions may be any of –
   - **Structural assumptions:** involves questions of how the system operates and usually involves simplification and abstractions of reality. For example, in queuing system, we assume the serving as FCFS.

- **Data assumptions:** should be based on the collection of reliable data and correct statistical analysis of the data.

The use of goodness-of-fit tests is an important part of the validation of data assumptions.

3. **Validating input-output transformations:** The ultimate test of the model is the model's ability to predict the future behavior of the real system, when
   - the model input data match the real inputs
   - a policy used in the model is implemented at some point later in the system
   - the level of input variables are modified

   In this phase of validation process, the model is views as an input-output transformation. That is, the model accepts values of the input parameters and transforms these inputs into output measures of performance.

   A necessary condition for the validation of input-output transformation is that some version of the system under study exists, so that the system data satisfying at least one set of input conditions can be collected to compare with model predictions.

   If the system is in planning stage and no system operating data can be collected, complete input-output validation is not possible. Validation increases modeler's confidence that the model of existing system is accurate.

   Changes in the computerized representation of the system, ranging from relatively minor to relatively major include:
   - Minor changes of single numerical parameters such as speed of the machine, arrival rate of the customer etc.
   - Minor changes of the form of a statistical distribution such as distribution of service time or a time to failure of a machine.
   - Major changes in the logical structure of a subsystem such as change in queue discipline for waiting-line model, or a change in the scheduling rule for a job shop model.
   - Major changes involving a different design for the new system such as computerized inventory control system replacing a non computerized system.

7.3.1 Input-Output Validation: Using a Turing Test

Normally, statistical tests are used to compare model output with system output. Sometimes, statistical tests may not be applicable. Then, people have enough knowledge about system's behavior are asked to compare model output with system output.

For example, suppose that five reports of system performance over five different days are prepared, and simulation output data are used to produce five "fake" reports. The 10 reports should all be in exactly the same format and should contain relevant. The 10 reports are randomly shuffled and given to the engineer, who is asked to decide which reports are fake and which are real. If the engineer identifies a substantial number of the fake reports, the model builder questions the engineer and uses the information gained to improve the model. If the engineer cannot distinguish between fake and real reports with any consistency, the modeler will conclude that this test provides no evidence of model inadequacy. This type of validation test is commonly called a Turing test. It can be used as a valuable tool in detecting model inadequacies and, eventually, in increasing model credibility as the model is improved and refined.

Important Questions on this Unit:

1. What is verification of simulation models? List the suggestions given for verification of model.
2. Explain briefly three-step approach to validation by Nayler and Finger.
3. Define validation and calibration of a simulation model. Explain the iterative process of calibrating a model with a neat diagram.
4. Write a note on input-output transformation.
5. Briefly explain the use of Turing test on input-output validation.
6. Differentiate verification and validation of simulation models. Suggest techniques which help in verification.