INTRODUCTION

How do we know that the model we have used, is an accurate representation of the system being simulated? This is an important question and must be answered satisfactorily before a simulation study can be made use of. The area of experimentation and results analysis for simulation models is well developed and a range of rules and guiding methods can be found in the literature, e.g. are available. Many of the techniques developed are here to ensure that dangerous mistakes are not made when analyzing and interpreting the results. In fact , without establishing the validity of the model , if we accept the(erroneous) simulation results the consequences may be disastrous. Put simply the power of modern simulation software to generate large quantities of data can leave the user with the false sense of security that the results generated are credible and truly representative of the system under study. Like all modelling techniques care needs to be exercised.

What is a valid model? Since no simulation model will duplicate the given system in every detail it is not an appropriate question to ask if a simulator is a 'true' model of a real system. We should only ask if the model is a 'reasonable' approximation of the real system. The acceptable levels of reasonableness and approximation will vary from system to system and simulation to simulation. There is no universally acceptable criteria for accepting a simulation model as a valid representation. There are only guidelines that aid in establishing confidence in the model.

There are a number of phases as shown in Figure 6.1, for checking a simulation model prior to experimental analysis:

- **Verification** - the accuracy of transforming a problem formulation into a model (specification) deals with building the **model right**

- **Validation** - model behaves with satisfactory accuracy consistent with the study objectives deals with building the **right model**

The above are essential checks performed prior to analysis of the model and are used to establish what is known as model credibility. Verification and validation is shown in Figure 2, as cause - effect and input – output terms.

A simulation run typically starts in the empty and idle state. The run is therefore characterized by a "run-in" phase followed by a "steady state" phase, see Figure 6.1. The run-in phase is generally ignored and is only used for investigating the effects of transient conditions such as starting up a new factory or performing radical changes within an existing facility.

Typically the steady state phase is of greater interest. At this stage checks must be made to ensure no long term trends exist, such as continual build up of stock in the factory, that suggest the model (hence the real system) will be unstable and unworkable.

Generally what is known as multiple replications is performed. This is where the model is run several times. Each time the random number generators are set to provide different sequences of random numbers, e.g. in case of manufacturing process the breakdown patterns of machines are different and the points at which material is scrapped is different. This allows confidence that the results being compiled represent the average and the range of conditions that are likely and therefore play down 'freak' or 'unusual' behaviour.

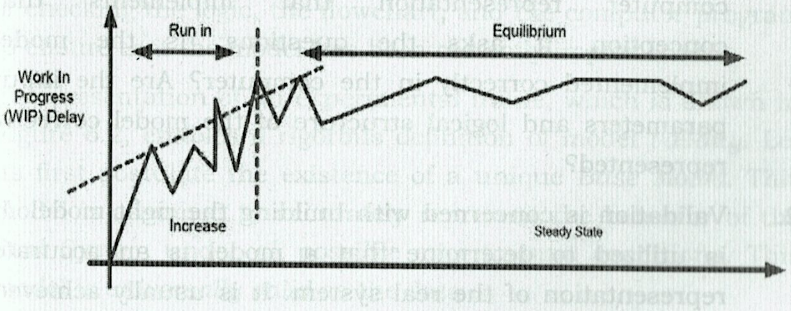


Figure 6.1: The two key phases of a simulation run

Scanned by VueScan - get a free trial at www.hamrick.com

One of the most important and difficult tasks facing a model developer is the verification and validation of the simulation model. The engineers and analysts who use the model outputs to aid in making design recommendations and the managers who make decisions based on these recommendations-justifiably look upon a model with some degree of skepticism about its validity. It is the job of the model developer to work closely with the end users throughout the period (development and validation to reduce this skepticism and to increase the credibility.

The goal of the validation process is twofold:

1. To produce a model that represents true system behavior closely enough for the model to be used as a substitute for the actual system for the purpose of experimenting with system.
2. To increase an acceptable, level the credibility of the model ,so that the model will be used by managers and other decision makers.

Validation should not be seen as an isolated set of procedures that follows model development, but rather an integral part of model development. The verification and validation process consists of the following components:

1. Verification is concerned with building the model right. It is utilized in 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?
2. Validation is concerned with building the right model. It is utilized to determine that a model is an accurate representation of the real system. It is usually achieved through the calibration of the model.

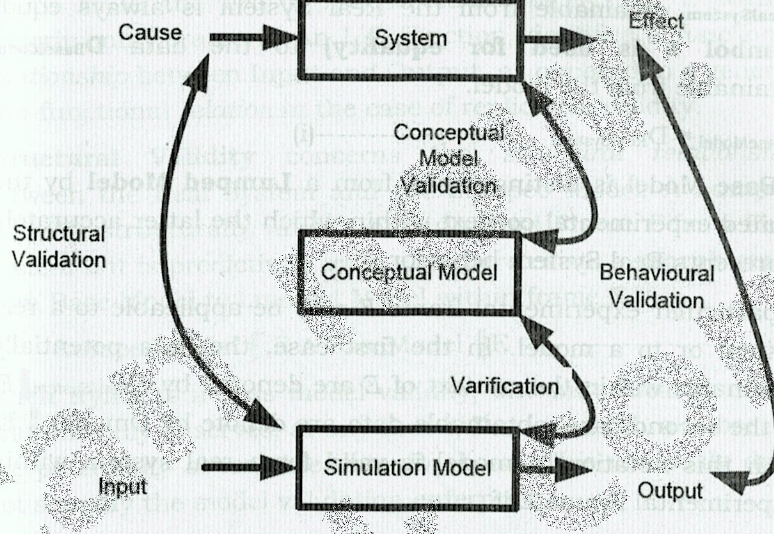


Figure 6.2: Verification and Validation Activities

VERIFICATION AND VALIDATION

The validation efforts can be grouped into two parts

1. validation of the abstract model itself
2. validation of its implementation

The first part consists of examining all assumptions, which transform the real world system into the conceptual model. A great deal of judgment and an intimate knowledge of the real system are involved in this step. The validation of the abstract model is often highly subjective. Testing the validity of an implementation is a more objective and easier task. It consists of checking the logic, the flowchart, and the computer program to ensure that the model has been correctly implemented.

The presentation of an experimental frame, which is shown in Figure 6.2, enables a rigorous definition of model *validity*. Let us first postulate the existence of a unique *Base Model*. This model is assumed to accurately represent the behavior of the Real System under *all* possible experimental conditions. This model is *universally valid* as the data

$D_{RealSystem}$ obtainable from the Real System is always equal **(symbol ≡ is used for equality)** to the data $D_{BaseModel}$ obtainable from the model.

$$D_{BaseModel} \equiv D_{RealSystem} \quad \text{-------------------(i)}$$

A Base Model is distinguished from a **Lumped Model** by the limited experimental context within which the latter accurately represents Real System behavior.

A particular experimental frame *E* may be applicable to a real system or to a model. In the first case, the data potentially obtainable within the context of *E* are denoted by $D_{RealSystem} \| E$. In the second case, obtainable data are denote by Dmodel ‖ E. With this notation, a model is valid for a real system within Experimental Frame *E* if

$$D_{LumpedModel} \| E \equiv D_{RealSystem} \| E \quad \text{-------------------(ii)}$$

The data equality ≡ *m*ust be interpreted as equal to a certain degree of accuracy. It shows how the concept of validity is not absolute, but is related to the experimental *context* within which Model and Real System *behavior* are compared and to the *accuracy metric* used.

One typically distinguishes between the following types of model validity.

Replicative Validity concerns the ability of the Lumped Model to *replicate* the input/output data of the Real System. With the definition of a Base Model, a **Lumped Model** is respectively valid in Experimental Frame *E* for a Real System if

$$D_{LumpedModel} \| E \equiv D_{BaseModel} \| E \quad \text{-------------------(iii)}$$

Predictive Validity concerns the ability to identify the *state* a model should be set into to allow *prediction* of the response of the Real System to *any* (not only the ones used to identify the model) input segment. A Lumped Model is predicatively valid in Experimental Frame *E* for a Real System if it is replicatively valid and

$$F_{LumpedModel} \| E \subseteq F_{BaseModel} \| E \quad \text{-------------------(iv)}$$

where *Fs* is the set of I/O functions of system *S* within Experimental Frame *E*. An I/O function identifies a *functional relationship* between Input and Output, as opposed to a general non-functional *relation* in the case of replicative validity.

Structural Validity concerns the *structural relationship* between the Real System and the Lumped Model. A Lumped Model is structurally valid in Experimental Frame *E* for a Real System if it is predictively valid and there exists a **morphism** Δ from Base Model to Lumped Model within frame *E*.

$$LumpedModel \| E \,\Delta\, BaseModel \| E \quad \text{-------------------(v)}$$

When trying to assess model validity, one must bear in mind that one only observes, at any time *t*, $D_{RealSystem}(t)$, a subset of the potentially observable data $D_{RealSystem}$. This obviously does not simplify the model validation enterprise.

Whereas assessing model validity is intrinsically impossible, the *verification* of a *model implementation* can be done rigorously. A *simulator* implements a lumped model and is thus a source of obtainable data $D_{Simulator}$. If it is possible to prove (often by design) a structural relationship (morphism) between Lumped model and Simulator, the following will hold unconditionally

$$D_{Simulator} \equiv D_{LumpedModel} \quad \text{-------------------(vi)}$$

Before we go deeper into predictive validity, the relationship between different *refinements* of both Experimental Frames and models is elaborated. In Figure 6.3, the *derived from* relationship for Experimental Frames and the *homomorphism*.

Relationship for Models is depicted. If we think of an Experimental Frame as a formal representation of the context within which the model is a valid representation of the dynamics of the system, a more restricted Experimental Frame means a more specific behaviour. It is obvious that such a restricted Experimental Frame will match far more models than a more general Experimental Frame. Few models are elaborate enough to be valid in a very general input/parameter/ performance range. Hence, the large number of applies to (*i.e.*, match) lines emanating from a restricted Experimental Frame.

Scanned by VueScan - get a free trial at www.hamrick.com

The homomorphism between models means that, when modifying/transforming a model (*e.g.*, adding some non-linear term to a model), the simulation results (*i.e.*, the behaviour) within the same experimental frame must remain the same.

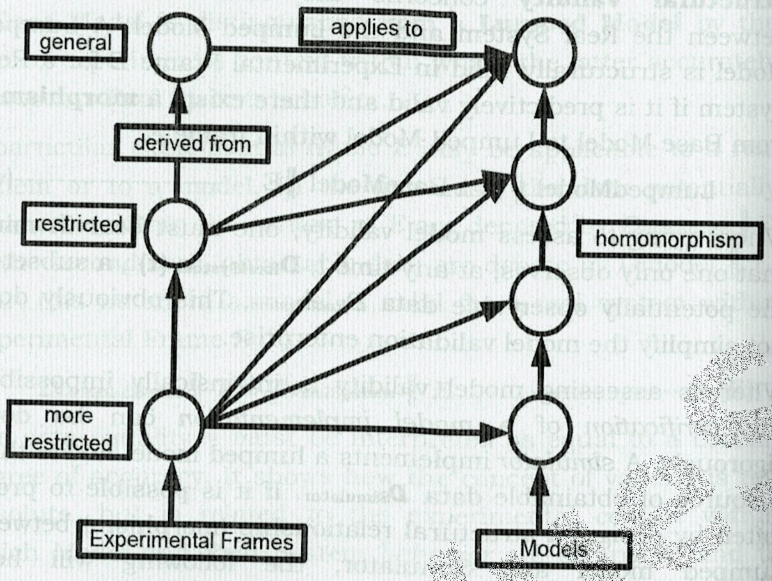


Figure 6.3: Experimental Frame – Model Relationship

Though it is meaningful to keep the above in mind during model development and use, the highly non-linear nature of many continuous models makes it very difficult to automate the management of information depicted in Figure 6.3. Non-linear behaviour makes it almost impossible, based on a model or experimental frame symbolic representation, to make a statement about the area in state-space, which will be covered (*i.e.*, behaviour). A pragmatic approach is to

1. Let an expert indicate what the different relations are. This is based on some insight into the nonlinear dynamics. Such expert knowledge can be built from a large number of conducted experiments.
2. Constantly with each experiment validate the expert information.

A crucial question is whether a model has predictive validity, is it capable not only of reproducing data which was used to choose the model and parameters but also of predicting new behavior? The predictive validity of a model is usually substantiated by comparing new experimental data sets to those produced by simulation, an activity known as model validation. Due to its special importance in the communication between model builders and users, model validation has received considerable attention in the past few decades. The comparison of the experimental and simulation data are accomplished either subjectively, such as through graphical comparison, Turing test, or statistically, such as through analysis of the mean and variance of the residual signal employing the standard F statistics, multivariate analysis of variance regression analysis, spectral analysis, autoregressive analysis, autocorrelation function testing, error analysis, and some non-parametric methods.

The above-mentioned methods are designed to determine, through comparison of measured and simulated data, the validity of a model. As one might intuitively expect, different modelling errors usually cause the behavior of the model to deviate in different ways from that of the real system or, in other words, different modelling errors correspond to different pattern in the error signal, the difference between experimental data and simulated data. These patterns if extractable, can obviously be used to identify the modelling errors.

Model Building, Verification, and Validation

The first step in model building consists of observing the real system and the interactions among its various components and collecting data on its behavior. Operators, technicians, repair and maintenance personnel, engineers, supervisors, and managers under certain aspects of the system which may be unfamiliar to others. As model development proceeds, new questions may arise, and the model developers will return, to this step of learning true system structure and behavior.

Scanned by VueScan - get a free trial at www.hamrick.com

The second step in model building is the construction of a conceptual model - a collection of assumptions on the components and the structure of the system, plus hypotheses on the values of model input parameters, illustrated by the following figure 6.4

The third step is the translation of the operational model into a computer recognizable form- the computerized model.

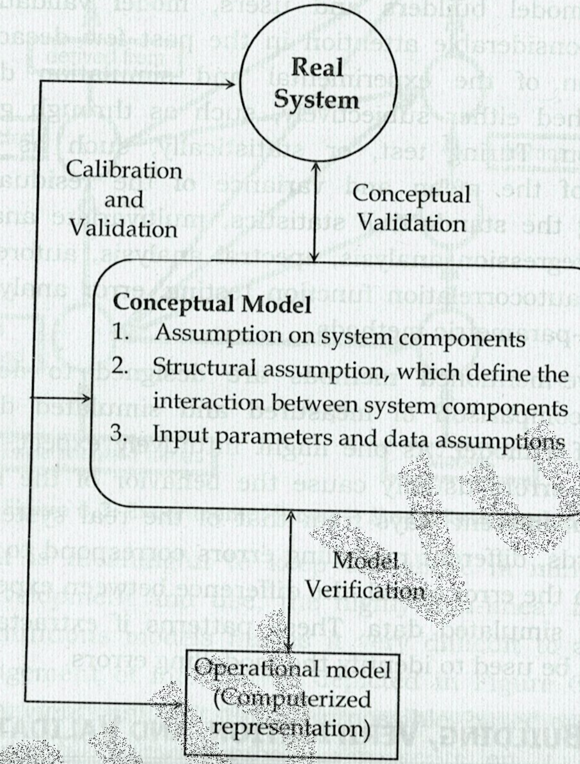


Figure 6.4: Model Building, Verification and Validation

VERIFICATION OF SIMULATION MODELS

The purpose of model verification is to assure that the conceptual model is reflected accurately in the computerized representation. The conceptual model quite often involves some degree of abstraction about system operations, or some amount of simplification of actual operations. Many common-sense 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 a 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.
8. Graphical interfaces are recommended for accomplishing verification & validation .

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. The following figure 6.5 shows the relationship of the model calibration to the overall validation process. The comparison of the model to reality is carried out by variety of test. Test are subjective and objective.

Scanned by VueScan - get a free trial at www.hamrick.com

Subjective test usually involve people, who are knowledgeable about one or more aspects of the system, making judgments about the model and its output. Objective tests always require data on the system's behavior plus the corresponding data produced by the model

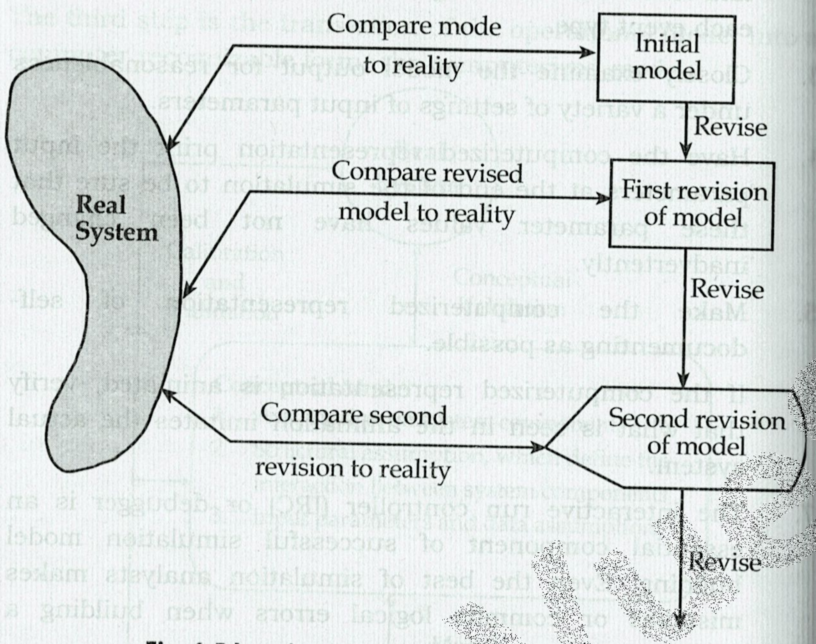


Fig. 6.5 Iterative process of calibration a model

A possible criticism of the calibration phase, were it to stop at point, i.e., the model has been validated only for the one data set used; that is, the model has been "fit" to one data set. Validation is not an either/or proposition—no model is ever totally representative of the system under study. In addition, each revision of the model, as in the Figure above involves some cost, time, and effort.

Three-Step Approach for Validation of Simulation Models

As an aid in the validation process, Naylor and Finger [1967] formulated a three step approach which has been widely followed:-

1. Build a model that has high face validity.
2. Validate model assumptions.

3. Compare the model input-output transformations to corresponding input-output transformations for the real system.

1. 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 users of a model should be involved in model construction from its conceptualization to its implementation *to* ensure that a high degree of realism is built into the model through reasonable assumptions regarding system structure, and reliable data.

Another advantage of user involvement is the increase in the models perceived validity or credibility without which manager will not be willing to trust simulation results as the basis for decision making.

Sensitivity analysis can also be used to check model's face validity. The model user is asked if the model behaves in the expected way when one or more input variables is changed. Based on experience and observations on the real system the modulus r and model builder would probably have some notion at least of the direction of change in model output when an input variable is increased or decreased. The model builder must attempt to choose the most critical input variables for testing if it is too expensive or time consuming to vary all input variables. If real system data are available for at least two settings of the input parameters, objective scientific sensitivity tests can be conducted via appropriate statistical techniques.

2. Validation of Model Assumptions

Model assumptions fall into two general classes: structural assumptions and data assumptions. Structural assumptions involve questions of how the system operates and usually involve simplification and abstractions of reality.

Scanned by VueScan - get a free trial at www.hamrick.com

For example, consider the customer queuing and service facility in a bank. Customers may form one line, or there may be an individual line for each teller. If there are many lines, customers may be served strictly on a first-come, first-served basis, or some customers may change lines if one is moving faster. The number of tellers may be fixed or variable. These structural assumptions should be verified by actual observation during appropriate time periods together with discussions with managers and tellers regarding bank policies and actual implementation of these policies.

Data assumptions should be based on the collection of reliable data and correct statistical analysis of the data. For example: Data were collected on:

- Inter arrival times of customers during several 2-hour periods of peak loading ("rush-hour" traffic)
- Inter arrival times during a slack period
- Service times for commercial accounts
- Service times for personal accounts.

The procedure for analyzing input data consist of three steps:-

- Identifying the appropriate probability distribution.
- Estimating the parameters of the hypothesized distribution.
- Validating the assumed statistical model by goodness – of – fit test such as the chi-square test, KS test and by graphical methods.

3. **Validating Input-Output Transformation:**

In this phase of validation process the model is viewed as input – output transformation. That is, the model accepts the values of input parameters and transforms these inputs into output measure of performance. It is this correspondence that is being validated. Instead of validating the model input-output transformation by predicting the future ,the modeler may use past historical

data which has been served for validation purposes that is, if one set has been used to develop calibrate the model, its recommended that a separate data test be used as final validation test. Thus accurate " prediction of the past" may replace prediction of the future for purpose of validating the future.

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

If the change to the computerized representation of the system is minor such as in items one or two these change can be carefully verified d output from new model can be accepted with considerable confidence. Partial validation of substantial model changes in item three and four may be possible.

Scanned by VueScan - get a free trial at www.hamrick.com

Input-Output Validation: Using Historical Input Data

When using artificially generated data as input data the modeler expects the model produce event patterns that are compatible with, but not identical to, the event patterns that occurred in the real system during the period of data collection.

Thus, in the bank model, artificial input data $\{X_n, X_{2n}, n = 1, 2, .\}$ for inter arrival and service times were generated and replicates of the output data Y_2 were compared to what was observed in the real system

An alternative to generating input data is to use the actual historical record, $\{A_n, S_n, n = 1, 2, ...\}$, to drive simulation model and then to compare model output to system data.

To implement this technique for the bank model, the data A_i, $A_2, ..., S_1\ S_2$ would have to be entered into the model into arrays, or stored on a file to be read as the need arose.

To conduct a validation test using historical input data, it is important that all input data $(A_n, S_n, ...)$ and all the system response data, such as average delay(Z_2), be collected during the same time period.

Otherwise, comparison of model responses to system responses, such as the comparison of average delay in the model (Y_2) to that in the system (Z_2), could be misleading

Responses (Y_2 and $Z2$) depend on the inputs (A_n and S_n) as well as on the structure of the system, or model.

Implementation of this technique could be difficult for a large system because of the need for simultaneous data collection of all input variables and those response variables of primary interest.

Input-Output Validation: Using a Turing Test

In addition to statistical tests, or when no statistical test is readily applicable persons knowledgeable about system behavior can be used to compare model output to system output. For example, suppose that five reports of system performance over five different days are prepared, and simulation output are used to produce five "fake" reports. The 10 reports should all be in exactly in the same format and should contain information of the type that manager and engineer have previously s n on the system. The ten reports are randomly shuffled and given to the engineers , who is asked to decide which report are fake and which are real. If engineer identifies substantial number of 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 called as TURING TEST.

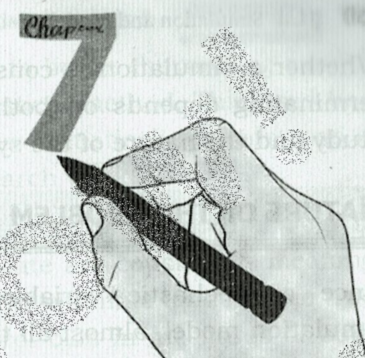
Discussion Exercise

1. Two similar terms used in the steps of a simulation study are "verification" and "validation." One of them refers to the debugging of the simulation code itself. Which one refers to the process of insuring that the model is a correct representation of the system?
2. What do you understand the term "Model Validation and Verification"? Explain.
3. What do you understand by the term Face Validity of a Conceptual Model?
4. What is the difference between Validation and Verification?
5. Why Validation is so important in Modelling and Simulation?
6. Give some advantages and disadvantage of Validation in Simulation.
7. How model can be build verification and validate? Explain with diagram.

Scanned by VueScan - get a free trial at www.hamrick.com

8. What are the techniques for verification of simulation model?
9. Describe in detail the three step approach for model validation?
10. What is model reasonable ness & explain how current contents and total count can verify it?
11. Briefly explain the validation of input-output transformations of the model and the various techniques used?
11. What do you mean by V & V? Explain the process of calibration and validation and the approaches used for validation with suitable example.

□□□



ANALYSIS OF SIMULATION OUTPUT

CHAPTER OUTLINE

After studying this chapter, the students will be able to understand the

- Confidence Intervals and Hypothesis Testing
- Estimation Methods
- Simulation run statistics
- Replication of runs
- Elimination of initial bias

Buy VueScan now! www.hamrick.com

Scanned by VueScan - get a free trial at www.hamrick.com