Val v V
right sy

CHAPTER 9

Module 4

Verification and Validation of Models

9.1 Introduction

- Model verification and validation (V&V) are essential parts of the model development process if models to be accepted and used to support decision making.
- In general, verification focuses on the internal consistency of a model, while validation is concerned with the correspondence between the model and the reality. The term validation is applied to those processes which seek to determine whether or not a simulation is correct with respect to the "real" system. More prosaically, validation is concerned with the question "Are we building the right system?"
- Verification, on the other hand, seeks to answer the question "Are we building the system right?" Verification checks that the implementation of the simulation model (program) corresponds to the model. Validation checks that the model corresponds to reality. Calibration checks that the data generated by the simulation matches real (observed) data.

Validation

- The process of comparing the model's output with the behavior of the phenomenon. In other words : comparing model execution to reality (physical or otherwise) .
- Validation 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.

Verification

- The process of comparing the computer code with the model to ensure that the code is a correct implementation of the model.

- Verification 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 ?

Calibration

- The process of parameter estimation for a model. Calibration is a tweaking/tuning of existing parameters and usually does not involve the introduction of new ones, changing the model structure.
- In the context of optimization, calibration is an optimization procedure involved in system identification or during experimental design.

9.2 Model Building

Building a model of a system is the most important task involved in its simulation. There are various steps for the same :

i. To build the model, modeller first observes the real system. List the various component of real system; also list the interaction between these components. In order to get details of system, how it operates modeller generally discusses with people familiar with system.

ii. In second step modeller create the conceptual model, based on component assumptions, structural assumptions, input parameter assumption and data assumptions.

iii. In third step is implementation. Simulation software is used to build operational model.

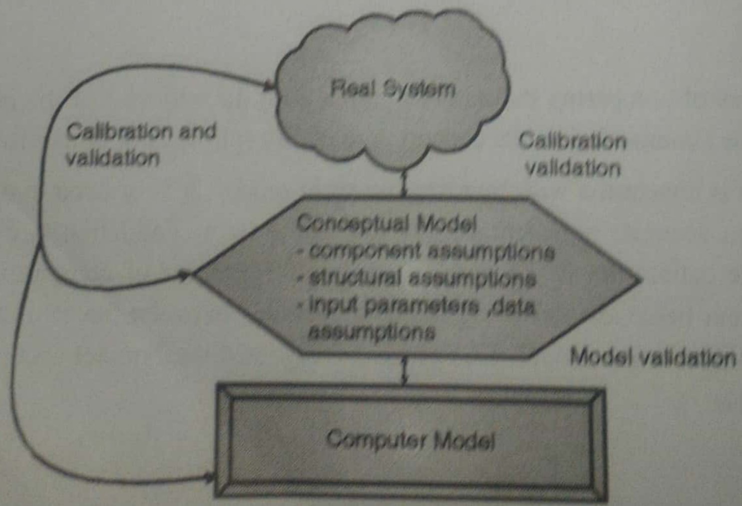

Fig. 9.2.1 : model building

9.3 Verification of Models

Q. Explain verification process. MU - May 16

Q. Explain in detail verification of simulation model. MU - Dec. 06, May 07, Dec. 07, May 08, Dec. 08, May 09, MU - Dec. 10, Dec. 11, May 12, Dec. 13, May 14, May 15

- Verification is defined as the process of correctly building a system model.
- Verification is done to ensure that :
  - The model is programmed correctly.
  - The algorithms have been implemented properly.
  - The model does not contain errors, oversights, or bugs.
- Verification ensures that the specification is complete and that mistakes have not been made in implementing the model.
- Verification does not ensure the model :
  - Solves an important problem.
  - Meets a specified set of model requirements.
  - Correctly reflects the workings of a real world process.
- The purpose of verification is to make sure that the conceptual model is being reflected accurately in the operational model.
- It proceeds by the comparison of the conceptual model to the computer representation that implements the conception.
- It asks questions regarding the correct implementation of the model, correct representation of the input parameters and logical structure, etc.
- Verification is determining whether the simulation computer program performs as intended (debugging the computer program).
- Verification checks for translation of conceptual simulation model (flowcharts and assumptions) onto a correctly working program.

9.3.1 Verification Process

There are various approaches that can be useful in the process of verification. They are as :

1. Getting the operational model checked by someone other than the developer, preferably some expert.

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. Have the implemented model display a wide variety of outputs for a range of inputs.
4. Print the input parameters at the end of the simulation to be sure that the values have not changed.
5. If the operational model is animated, verify that what is seen in the animation imitates the actual system.
6. Graphical interfaces are also recommended for accomplishing verification. GUI simplifies task of model understanding.
7. The Interactive Run Controller or Debugger is used for finding and correcting any kind of errors. It can be done in the following ways :
   i. The simulation is monitored as it progresses.
   ii. Attention can be focused on a particular entity line, code or procedure.
   iii. Values of selected model components can be observed.
   iv. The simulation can be temporarily suspended or paused so as to reassign values or redirect entities.
8. Make the model as self – documenting as possible. Brief comments, descriptions of major sections plus definitions of all variables and parameters should be written. Documentation is important for clarifying the logic of the model as well as verifying its completeness.
9. Another common sense approach is to check the model for reasonableness. For this various sets of statistics are used namely current contents and total count. Current contents refer to the number of items in each component of the system at a given time. Total count refers to the total number of items that have entered each component of the system at a given time.
10. *Another sophisticated is the use of trace. A trace is a computer printout, which gives the values of every variable in a computer program, every time that one of these variables changes in value.*
   i. *It can be very labor intensive if the programming language does not support statistic collection.*
   ii. *Labor can be reduced by a centralized tracing mechanism.*
   iii. *In object-oriented simulation framework, trace support can be integrated into class hierarchy. New classes need only to add little for the trace support.*

11. Compute certain long-run measures of performance. e.g. compute the long –run server utilization and compare to simulation results.

In short, steps of verification are :

Step 1 : Get the computer representation checked by independent and knowledgeable person.

Step 2 : Make the detail flow diagram.

Step 3 : Examine the output by giving different – different input to system.

Step 4 : Prepare detail documentation.

Step 5 : With the help of two software IRC (Interrupt Run Controller) and debugger, check the mistake.

9.4 Validation of Models

MU - May 16

Q. How would you validate simulation models ?
MU - May 05, Dec. 05, May 07, Dec. 07, Dec. 08, May 09

Q. Explain in detail validation of simulation model.
MU - May 12, Dec. 13, May 14, Dec. 14

- The standard method to validate model is to construct a model of the existing system. Then, change the model appropriately in order to analyze each alternative. The model of the existing system can be validated by comparing its results against actual data obtained from the system under investigation.
- Goal of validation is to ensure that the simulation model is good enough so that it can be used to make decisions about the systems that we ideally would like to work with.
- Ease or difficulty of the validation process depends on the complexity of the system being modeled and on whether a version of a system currently exists.
- A simulation model of a complex system can only be an approximation to the actual system, regardless of how much effort is put into development. There is no such thing as absolutely valid model!
- A simulation model is always developed for a particular purpose.
- A logbook of the simulation model's assumptions should be updated on a regular basis and eventually should form integral part of the final report.
- A simulation model should be validated relative to those measures of performance that will actually be used for decision making. For example, if through-put is important parameter, the ability of the simulation model to predict that accurately should be tested.

The following checks can be carried out in order to validate a simulation model.

1. **Check the pseudo-random number generators :** Are the pseudo-random numbers uniformly distributed in (0,1) and do they satisfy statistical criteria of independence? Usually, one takes for granted that a random number generator is valid.
2. **Check the stochastic variate generators :** Similar statistical tests can be carried out for each stochastic variate generator built into a simulation model.
3. **Check the logic of the simulation program :** This is a rather difficult task. One way of going about it is to print out the status variables, the future event list, and other relevant data structures each time an event takes place in the simulation. Then, one has to check by hand whether the data structures are updated appropriately. This is a rather tedious task. However, using this method one can discover possible logical errors and also get a good feel about the simulation model.
4. **Relationship validity :** Quite frequently the structure of a system under study is not fully reflected down to its very detail in a simulation model. Therefore, it is important that the management has the opportunity to check whether the model's assumptions are credible.
5. **Output validity :** This is one of the most powerful validity checks. If actual data are available regarding the system under study, then these data can be compared with the output obtained from the simulation model. Obviously, if they do not compare well, the simulation model is not valid.

9.4.1 Validation Process

Q.	Explain validation process.	MU - May 08
Q.	Iterative process of calibrating a model.	MU - May 11

- Validation is process by which model user gain confidence that output analysis is making valid inferences about the real system under study.
- Goal of validation is to ensure that the simulation model is good enough so that it can be used to make decisions about the systems that we ideally would like to work with.
- Validations used to make sure model meet intended requirements. Model should address right problem and provide accurate information about modelled system
- Validation is achieved by regular calibration. Calibration is iterative process of comparing the model to the real system, making required adjustments comparing again and so on. i.e. Calibration is adjusting models and parameters to fit real system behaviour.

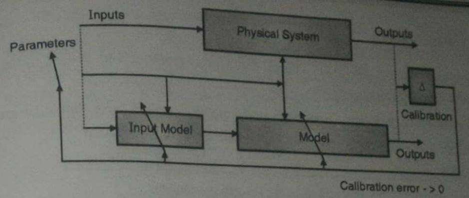

Fig. 9.4.1 : Calibration process

- The comparison of model is done by either subjective or objective tests.
- The subjective involves people who are knowledgeable about one or more aspects of the system, making judgment about the model and its output.
- Objective tests require on the systems behavior plus the corresponding data produced by the model. Then any one statistical test is carried out to compare some aspect of the system data with the model data.
- Calibration should always be done for more than one data set. Hence, after the calibration has been done using the original data set, a final validation is conducted, using the second data set.
- In case of any discrepancy, calibration is performed again to make the required modification. Calibration stops when error is less that some threshold.
- If a specified level of accuracy cannot be achieved in a model, then the model should be abandoned.

9.4.2 Naylor and Finger Approach

Q.	Explain 3 step approaches of Naylor and Finger in validation process.	MU - Dec. 04, May 06
Q.	Explain Naylor and Finger validation approach.	MU - May 11, May 12, Dec. 13

Naylor and Finger validation method is three-step process.

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.

9.4.2.1 Face Validity

- Simulation molder constructs a model that appears reasonable on its face to the user and to other knowledgeable people.
- To ensure a high degree of realism in the model, potential users should be involved in model construction from its conceptualization to its implementation.
- Potential user involvement increases models validity and credibility.
- Model's face validity is checked using sensitivity analysis.
- Model is checked for an expected kind of behavior when the input variables are changed.
- Consider bank as an example, if we increase arrival rate of customers then it is expected that number of people in queue and delay for each customer will increase.
- For large-scale simulation models, we have to consider many input variables. To check models face validity we have to carry out possibly many sensitity tests. Sometimes not possible to perform all of these tests, select the most critical ones.
- Thus to check Face validity we have to ask two question :
  - Is the model reasonable on its face(without deep inspection or analysis) ?
  - Does model behavior change in expected ways with modification of parameters ?
- Face validity : example.

  We know that the server utilization for an M/M/c queue is

$$\rho = \frac{\lambda}{c\mu}$$

  What happens if we assume in our model that the server utilization is

$$\rho = \frac{\mu}{c\lambda}$$

- Is this a valid model ? Let's do a face validity test. Is the model reasonable on its face ?
- Verify units : utilization is a dimensionless quantity and λ, μ are rates, and c is dimensionless hence this passes the first test.
- Does model behavior change in expected ways with modification of parameters ?
- What happens to the server utilization as the arrival rate λ, or service rate μ increases, or as the number of servers, c, changes ?
  - Does Utilization, Increases with increasing service rate ?

  - Does Utilization, Decreases with increasing arrival rates ?
  - Does Utilization, Decreases with the increase in the number of servers?
- As utilization does not increase with service rate and also utilization does not decrease with increasing arrival rates hence this model is invalid on its face.

9.4.2.2 Validation of Model Assumptions

- General classes of model assumptions are :
  - Structural assumptions : How the system operates.
  - Data assumptions : Reliability of data and its statistical analysis.
- A structural assumption involves questions of how the system operates and usually involves simplification and abstraction of reality. They are verified by actual observation.
- Consider customer queueing and service facility in a bank. Structural assumptions are
  - Customer waiting in one line versus many lines.
  - Customers are served according FCFS versus priority.
- Data assumptions are based on reliability of data and correct statistical analysis of the data.
- Data reliability is verified by consultation. The reliability can be verified by conducting objective statistical test s for data homogeneity, when two data sets are being combined.
- Example of data assumption : Interarrival time of customers, service times for commercial accounts.
- Data regarding interarrival times during slack period and during rush hours in restaurants is also example of data assumption. You should verify reliability of data with managers.
- The analysis of input data from a random sample consists of three steps :
  - Identifying the appropriate probability distribution.
  - Estimating the parameters of the hypothesized distribution.
  - Validating the assumed statistical model by a goodness-of fit test, such as the chi-square or Kolmogorov-Smirnov test, and by graphical methods.

  Validation of model assumption is summarized in following Table 9.4.1.

Table 9.4.1

Type of model assumptions	Based on
Structural assumptions • System operation • Simplifying assumptions • Abstractions of reality	Observation of the system
Data assumptions • Homogeneity • Independence • Distribution	Collection of reliable data Correct statistical analysis

9.4.2.3 Validating Input-output Transformations

MU - May 16

Q. Describe briefly method of validating input model. MU - Dec. 07, Dec. 08, May 09

Q. How would you validate input-output transformation of a model? MU - May 13

- The ultimate test of the model is its ability to predict the future behavior of the real system when the input data matches the real inputs.
- The structure of the model should be accurate enough to make good predictions for the range of input data sets of interest.
- We can see the outputs of the system as being a functional transform of the inputs, based on parameter settings. i.e. the model accepts values of inputs parameters and transforms them into suitable outputs measures of performances.
- Instead of validating the model input-output transformations by predicting the future we may use past historical data which have been reserved for validation purposes only.
- A model is developed with primary interest in a specific set of system responses to be measured under some range of input condition.
- The Model used main responses of interest as criteria for validating model.

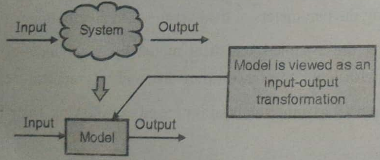

Fig. 9.4.2

- If the model is later used for some other purposes different from the original one it should be revalidated in terms of new responses under new condition.
- Complete input/ output validation is not possible when system is in planning stages. As system is in planning stage, no operating data can be collected.
- Partial input –output validation can be done, when subsystems are in under planning stages.
- Changes can also be made in the operational model. Either major or minor e.g. minor changes of numerical parameters, major changes in the logical structure of the subsystem.
- Input-output validation can be by following methods :
  - Using historical input data
  - Using a Turing test

(a) Validation using Historical Input Data

- This method is an alternative to generating input data that uses the actual historical record.
- The simulation model is driven with historical record and then the model is compared to system data.
- Using historical input data method modeller will try to duplicate important events that are occurred in real system. If modeller have created accurate model, then various parameter like server utilization can be predicted close to real system.
- To conduct validation test using this method, all the input data and all the system responses data should be collected during same time period, otherwise comparison of model responses to system responses could be misleading.
- The responses depend on the inputs and the structure of the system.
- The implementation of this method is generally difficult for large system.
- The reason for this is the need for the simultaneous data collection of all the inputs variables and all responses variable of primary interest.

(b) Validation Using a Turing Test

- This method is used when no statistical test is readily applicable.
- This method basically utilizes person's knowledge about the system to compare model output to system output.

- Consider an observer knowing nothing about the real system or model. If the observer cannot distinguish the output for the real and the simulated system with any consistency then validity of the simulation is not rejected.
- If the observer consistently identifies simulation and real system then improve model, based on the observer's critical observations.

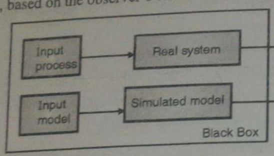

Fig. 9.4.3

- Simulation output data are used to produce fake reports in exactly the same format of the system output.
- For example present 10 system performance reports to a manager of the system. Five of them are form the real system and the rest are fake reports based on simulation output.
- If the person identifies a substantial number of the fake reports, interview the person to get information for model improvement.
- If person cannot distinguish between fake and real reports with consistency, conclude that the test gives no evidence of model inadequacy.

Review Questions

Q. 1 What do you understand by model verification ?

Q. 2 Explain verification process.

Q. 3 Explain verification and validation process.

Q. 4 How would you validate simulation models ?

Q. 5 Explain 3 step approach of Naylor and Finger in validation process.

Q. 6 How would you validate input-output transformation of a model ?

Q. 7 Describe briefly method of validating input model.

Q. 8 What do you understand by model verification and validation ? How would you validate input-output transformation of a model ?

9.5 University Questions and Answer

May 2010

Q. 1 What do you understand by calibration and validation of models ? How can one increase the face validity of a model and validate the model assumptions ? **(Sections 9.1 and 9.4)** **(10 Marks)**

Dec. 2010

Q. 2 Explain in detail verification of a simulation model. **(Section 9.3)** **(10 Marks)**

May 2011

Q. 3 Explain Naylor and Finger validation approach. **(Section 9.4.2)** **(10 Marks)**

Q. 4 Iterative process of calibrating a model. **(Section 9.4.1)** **(10 Marks)**

Dec. 2011

Q. 5 Explain the verification process. **(Section 9.3)** **(10 Marks)**

May 2012

Q. 6 Explain Naylor and Finger validation approach. **(Section 9.4.2)** **(10 Marks)**

Q. 7 Write short notes on Verification and Validation Process. **(Sections 9.3 and 9.4)** **(5 Marks)**

May 2013

Q. 8 What do you understand by model verification and validation ? How would you validate input-output transformation of a model ? **(Sections 9.3, 9.4 and 9.4.2.3)** **(10 Marks)**

Dec. 2013

Q. 9 Explain Naylor and Finger validation approach. **(Section 9.4.2)** **(10 Marks)**

May 2014

Q. 10 What do you understand by model verification and validation? Describe Briefly the various methods of validating input model.**(Sections 9.3 and 9.4)** **(10 Marks)**

Dec. 2014

Q. 11 Explain in detail validation of simulation model. **(Section 9.4)** **(10 Marks)**