Chapter 10
Verification and Validation of Simulation Models

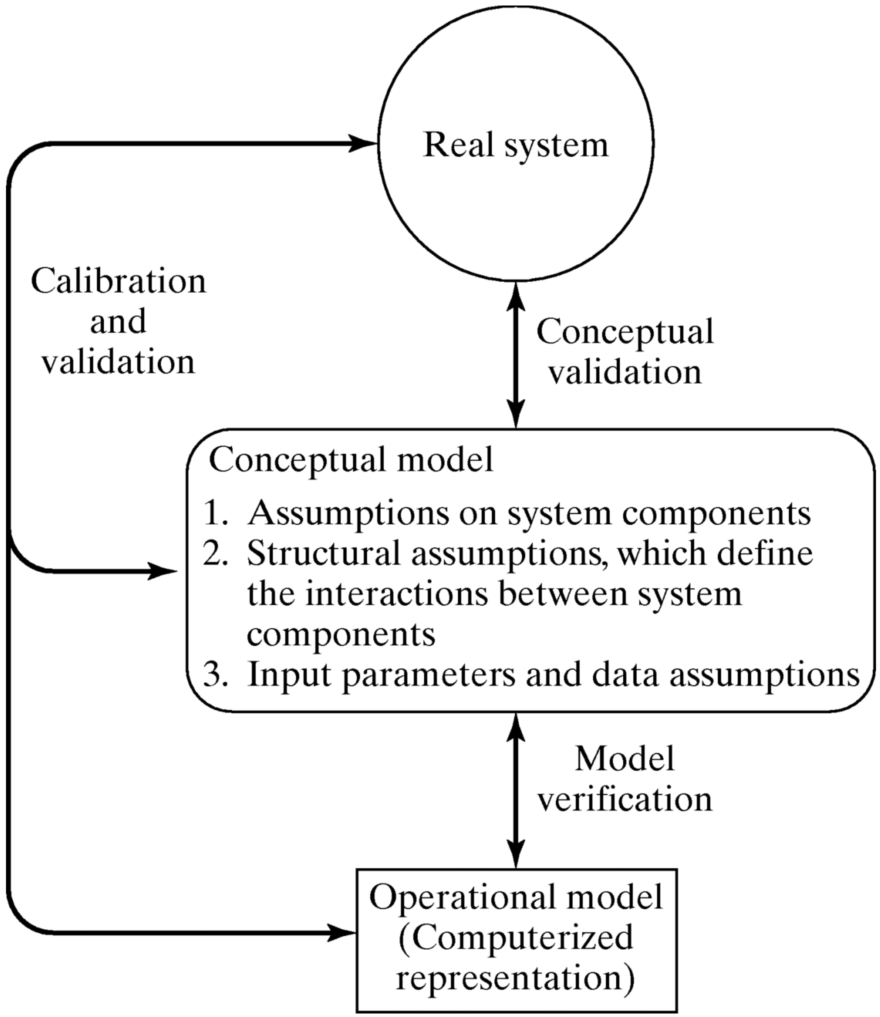
Banks, Carson, Nelson & Nicol

Discrete-Event System Simulation

Purpose & Overview

- The goal of the validation process is:
  - To produce a model that represents true behavior closely enough for decision-making purposes
  - To increase the model's credibility to an acceptable level
- Validation is an integral part of model development
  - Verification – building the model correctly (correctly implemented with good input and structure)
  - Validation – building the correct model (an accurate representation of the real system)
- Most methods are informal subjective comparisons while a few are formal statistical procedures

Modeling-Building, Verification & Validation



Verification

- Purpose: ensure the conceptual model is reflected accurately in the computerized representation.
- Many common-sense suggestions, for example:
  - Have someone else check the model.
  - Make a flow diagram that includes each logically possible action a system can take when an event occurs.
  - Closely examine the model output for reasonableness under a variety of input parameter settings. (Often overlooked!)
  - Print the input parameters at the end of the simulation, make sure they have not been changed inadvertently.

Examination of Model Output for Reasonableness [Verification]

- Example: A model of a complex network of queues consisting many service centers.
  - Response time is the primary interest, however, it is important to collect and print out **many statistics in addition to response time**.
    - Two statistics that give a quick indication of model reasonableness are **current contents** and **total counts**, for example:
      - If the current content grows in a more or less linear fashion as the simulation run time increases, it is likely that a queue is unstable
      - If the total count for some subsystem is zero, indicates no items entered that subsystem, a highly suspect occurrence
      - If the total and current count are equal to one, can indicate that an entity has captured a resource but never freed that resource.
    - Compute certain long-run measures of performance, e.g. compute the long-run server utilization and compare to simulation results

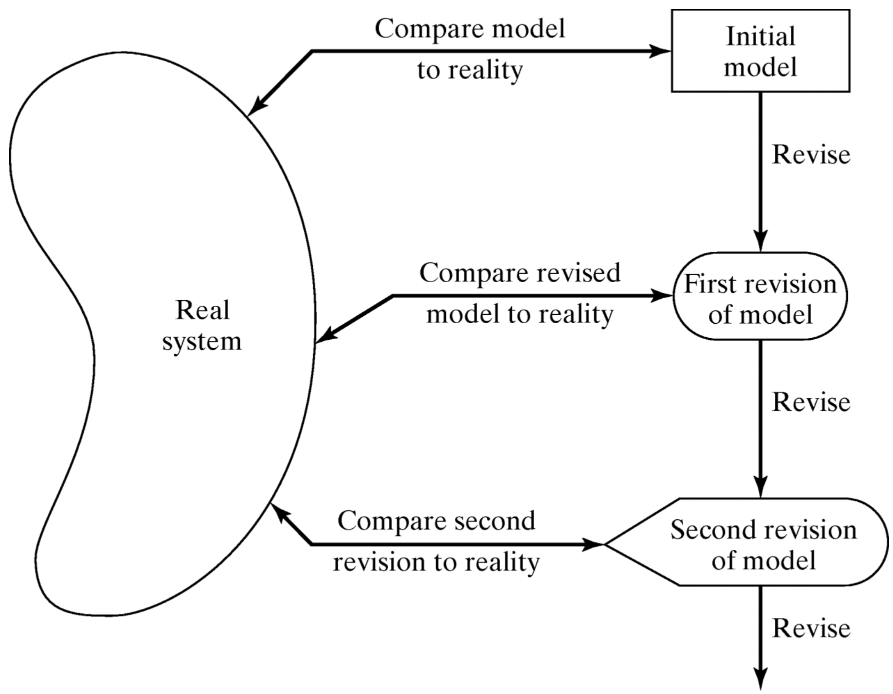
Other Important Tools

[Verification]

- Documentation
  - A means of clarifying the logic of a model and verifying its completeness
- Use of a trace
  - A detailed printout of the state of the simulation model over time.

Calibration and Validation

- Validation: the overall process of comparing the model and its behavior to the real system.
- Calibration: the iterative process of comparing the model to the real system and making adjustments.



Calibration and Validation

- No model is ever a perfect representation of the system
  - The modeler must weigh the possible, but not guaranteed, increase in model accuracy versus the cost of increased validation effort.
- Three-step approach:
  - Build a model that has high face validity.
  - Validate model assumptions.
  - Compare the model input-output transformations with the real system's data.

High Face Validity

- Ensure a high degree of realism: Potential users should be involved in model construction (from its conceptualization to its implementation).
- Sensitivity analysis can also be used to check a model's face validity.
  - Example: In most queueing systems, if the arrival rate of customers were to increase, it would be expected that server utilization, queue length and delays would tend to increase.

Validate Model Assumptions

[Calibration & Validation]

- General classes of model assumptions:
  - Structural assumptions: how the system operates.
  - Data assumptions: reliability of data and its statistical analysis.
- Bank example: customer queueing and service facility in a bank.
  - Structural assumptions, e.g., customer waiting in one line versus many lines, served FCFS versus priority.
  - Data assumptions, e.g., interarrival time of customers, service times for commercial accounts.
    - Verify data reliability with bank managers.
    - Test correlation and goodness of fit for data (see Chapter 9 for more details).

Validate Model Assumptions

[Calibration & Validation]

- General classes of model assumptions:
  - Structural assumptions: how the system operates.
  - Data assumptions: reliability of data and its statistical analysis.
- Bank example: customer queueing and service facility in a bank.
  - Structural assumptions, e.g., customer waiting in one line versus many lines, served FCFS verses priority.
  - Data assumptions, e.g., interarrival time of customers, service times for commercial accounts.
    - Verify data reliability with bank managers.
    - Test correlation and goodness of fit for data (see Chapter 9 for more details).

Validate Input-Output Transformation

[Calibration & Validation]

- Goal: Validate the model's ability to predict future behavior
  - The only objective test of the model.
  - The structure of the model should be accurate enough to make good predictions for the range of input data sets of interest.
- One possible approach: use historical data that have been reserved for validation purposes only.
- Criteria: use the main responses of interest.

Bank Example [Validate I-O Transformation]

- Example: One drive-in window serviced by one teller, only one or two transactions are allowed.
  - Data collection: 90 customers during 11 am to 1 pm.
    - Observed service times $\{S_i, i = 1,2, \ldots, 90\}$.
    - Observed interarrival times $\{A_i, i = 1,2, \ldots, 90\}$.
  - Data analysis let to the conclusion that:
    - Interarrival times: exponentially distributed with rate $\lambda = 45$
    - Service times: $N(1.1, 0.2^2)$

The Black Box

[Bank Example: Validate I-O Transformation]

- A model was developed in close consultation with bank management and employees
- Model assumptions were validated
- Resulting model is now viewed as a "black box":

Input Variables

Uncontrolled variables, X

Possion arrivals
$\lambda = 45/hr$: $X_{11}, X_{12}, \ldots$
Services times,
$N(D_2, 0.22)$: $X_{21}, X_{22}, \ldots$

Controlled Decision variables, D

$D_1 = 1$ (one teller)
$D_2 = 1.1$ min
(mean service time)
$D_3 = 1$ (one line)

Model "black box"
$\boldsymbol{f(X,D) = Y}$

Model Output Variables, *Y*

Primary interest:
Y_1 = teller's utilization
Y_2 = average delay
Y_3 = maximum line length

Secondary interest:
Y_4 = observed arrival rate
Y_5 = average service time
Y_6 = sample std. dev. of service times
Y_7 = average length of time

Comparison with Real System Data

[Bank Example: Validate I-O Transformation]

- Real system data are necessary for validation.
  - System responses should have been collected during the same time period (from *11*am to *1*pm on the same Friday.)
- Compare the average delay from the model Y_2 with the actual delay Z_2:
  - Average delay observed, $Z_2 = 4.3$ minutes, consider this to be the true mean value $\mu_0 = 4.3$.
  - When the model is run with generated random variates X_{1n} and X_{2n}, Y_2 should be close to Z_2.
  - Six statistically independent replications of the model, each of *2*-hour duration, are run.

Hypothesis Testing

[Bank Example: Validate I-O Transformation]

- Compare the average delay from the model Y_2 with the actual delay Z_2 (continued):
  - Null hypothesis testing: evaluate whether the simulation and the real system are *the same* (w.r.t. output measures):

$$H_0: E(Y_2) = 4.3 \text{ minutes}$$
$$H_1: E(Y_2) \neq 4.3 \text{ minutes}$$

    - If H_0 is not rejected, then, there is no reason to consider the model invalid
    - If H_0 is rejected, the current version of the model is rejected, and the modeler needs to improve the model

Hypothesis Testing

[Bank Example: Validate I-O Transformation]

- Conduct the *t* test:
  - Chose level of significance (α = *0.5*) and sample size (*n* = *6*), see result in Table 10.2.
  - Compute the same mean and sample standard deviation over the n replications:

$$\bar{Y}_2 = \frac{1}{n}\sum_{i=1}^{n} Y_{2i} = 2.51 \text{ minutes} \qquad S = \frac{\sum_{i=1}^{n}(Y_{2i} - \bar{Y}_2)^2}{n-1} = 0.81 \text{ minutes}$$

  - Compute test statistics:

$$|t_0| = \left|\frac{\bar{Y}_2 - \mu_0}{S/\sqrt{n}}\right| = \left|\frac{2.51 - 4.3}{0.82/\sqrt{6}}\right| = 5.24 \quad > \quad t_{critical} = 2.571 \quad \text{(for a 2 - sided test)}$$

  - Hence, reject H_0. Conclude that the model is inadequate.
  - Check: the assumptions justifying a *t* test, that the observations (Y_{2i}) are normally and independently distributed.

Hypothesis Testing

[Bank Example: Validate I-O Transformation]

- Similarly, compare the model output with the observed output for other measures:
  $Y_4 \leftrightarrow Z_4$, $Y_5 \leftrightarrow Z_5$, and $Y_6 \leftrightarrow Z_6$

Type II Error

[Validate I-O Transformation]

- For validation, the power of the test is:
  - Probability[detecting an invalid model] = $1 - \beta$
  - β = P(Type II error) = P(failing to reject $H_0 | H_1$ is true)
  - Consider failure to reject H_0 as a strong conclusion, the modeler would want β to be small.
  - Value of β depends on:
    - Sample size, n
    - The true difference, δ, between $E(Y)$ and μ: $\delta = \dfrac{|E(Y) - \mu|}{\sigma}$

- In general, the best approach to control β error is:
  - Specify the critical difference, δ.
  - Choose a sample size, n, by making use of the operating characteristics curve (OC curve).

Type I and II Error

[Validate I-O Transformation]

- Type I error (α):
  - Error of rejecting a valid model.
  - Controlled by specifying a small level of significance α.
- Type II error (β):
  - Error of accepting a model as valid when it is invalid.
  - Controlled by specifying critical difference and find the n.
- For a fixed sample size *n*, increasing α will decrease β.

Confidence Interval Testing

[Validate I-O Transformation]

- Confidence interval testing: evaluate whether the simulation and the real system are close enough.
- If Y is the simulation output, and $\mu = E(Y)$, the confidence interval (C.I.) for μ is:

$$\bar{Y} \pm t_{\alpha/2,n-1} S / \sqrt{n}$$

- Validating the model:
  - Suppose the C.I. does not contain μ_0:
    - If the best-case error is > ε, model needs to be refined.
    - If the worst-case error is ≤ ε, accept the model.
    - If best-case error is ≤ ε, additional replications are necessary.
  - Suppose the C.I. contains μ_0:
    - If either the best-case or worst-case error is > ε, additional replications are necessary.
    - If the worst-case error is ≤ ε, accept the model.

Confidence Interval Testing

[Validate I-O Transformation]

- Bank example: $\mu_0 = 4.3$, and “close enough” is $\varepsilon = 1$ minute of expected customer delay.
  - A *95%* confidence interval, based on the *6* replications is *[1.65, 3.37]* because:

$$\bar{Y} \pm t_{0.025,5} S / \sqrt{n}$$

$$4.3 \pm 2.51(0.82 / \sqrt{6})$$

  - Falls outside the confidence interval, the best case $|3.37 - 4.3| = 0.93 < 1$, but the worst case $|1.65 - 4.3| = 2.65 > 1$, additional replications are needed to reach a decision.

Using Historical Output Data

[Validate I-O Transformation]

- An alternative to generating input data:
  - Use the actual historical record.
  - Drive the simulation model with the historical record and then compare model output to system data.
  - In the bank example, use the recorded interarrival and service times for the customers $\{A_n, S_n, n = 1,2,\ldots\}$.
- Procedure and validation process: similar to the approach used for system generated input data.

Using a Turing Test

[Validate I-O Transformation]

- Use in addition to statistical test, or when no statistical test is readily applicable.
- Utilize persons' knowledge about the system.
- For example:
  - Present 10 system performance reports to a manager of the system. Five of them are from the real system and the rest are "fake" reports based on simulation output data.
  - If the person identifies a substantial number of the fake reports, interview the person to get information for model improvement.
  - If the person cannot distinguish between fake and real reports with consistency, conclude that the test gives no evidence of model inadequacy.

Summary

- Model validation is essential:
  - Model verification
  - Calibration and validation
  - Conceptual validation
- Best to compare system data to model data, and make comparison using a wide variety of techniques.
- Some techniques that we covered (in increasing cost-to-value ratios):
  - Insure high face validity by consulting knowledgeable persons.
  - Conduct simple statistical tests on assumed distributional forms.
  - Conduct a Turing test.
  - Compare model output to system output by statistical tests.