Steps in a simulation study

1. **Set objectives for simulation**
2. **Define the problem**
3. **Define overall approach**
4. **Sketch out model**
5. **Iterate, as needed**
6. **Create simulation**
7. **Collect data**
8. **Design experiment**
9. **Exercise simulation**
10. **Analyze results**
11. **Complete?**
12. **Implement system**

- **verified?**
- **Valid?**
- **document**
Verification and Validation of Simulation Models

• Model
  – Building the model
  – Verification of the model - is the model built correctly?
  – Validation of the model - was the correct model built?
    • Face validity
    • Model assumptions
    • Input-output transformation
    • Is the model a reliable and credible representation of the real system?
      – Reliability -> user issue
      – Credibility -> manager issue

• Simulation
  – Verification of the simulation model - is the simulation a correct implementation of the model intended?
Model Building, Verification, and Validation

1. Observe and understand the real system’s operation and behavior
Model Building, Verification, and Validation

1. Observe and understand the real system’s operation and behavior
2. Create a conceptual model

Real system

Conceptual model
- Component assumptions
- Structural assumptions
- Input parameters, data assumptions
Model Building, Verification, and Validation

1. Observe and understand the real system’s operation and behavior
2. Create a conceptual model

- Component assumptions
- Structural assumptions
- Input parameters, data assumptions
Model Building, Verification, and Validation

1. Observe and understand the real system’s operation and behavior
2. Create a conceptual model
3. Create a computerized model

Real system

Calibration and validation

Conceptual validation

Conceptual model
- Component assumptions
- Structural assumptions
- Input parameters, data assumptions

Operational Model (Computerized representation)
Model Building, Verification, and Validation

1. Observe and understand the real system’s operation and behavior
2. Create a conceptual model
3. Create a computerized model

Real system

Calibration and validation

Conceptual model
- Component assumptions
- Structural assumptions
- Input parameters, data assumptions

Operational Model (Computerized representation)

Conceptual validation
Model validation
Verification of Simulation Models

• Is the computer simulation an accurate representation of the conceptual model?

• Common sense approach to verification:
  – **VERIFY INITIAL CONDITIONS!**
  – Use graphical representation whenever possible - you can scan a 2 dimensional (graphical) view faster than a 1 dimensional (textual) view
    • Flow charts
    • Graphs of output behavior
    • Graphs of internal operation of model
    • Animations
  – Where possible, create internal model partitions that correspond to partitions in the real system - i.e., a top-down hierarchical view of model
  – **Test the simulation model for reasonableness under extreme value conditions.**
    – Test the simulation model for reasonableness under variation in parameters. Does the response follow the direction of the change?
    – Examine inputs and outputs of subsystems - too few or too many events may indicate errors elsewhere
  – Whenever possible, compare selected test conditions to analytical results
Calibration and Validation of Model

- Calibration and validation processes are distinct processes, normally performed simultaneously and interactively.
Calibration and Validation of Model

- Validation - does the behavior of the simulated system match that of the real system?

- Validation: $\approx$?
Calibration and Validation of Model

- Calibration - adjust or alter models and parameters to best fit real system behavior
- Consider the calibration process as a feedback control system, adjusting model to minimize the error (difference between model and reality)
- Calibration/Validation stops when error is less than some threshold, often cost-based
Face Validity

• Is the model reasonable on its face, i.e., without deep inspection or analysis
• Does behavior of model change in expected ways with modification of parameters?
Face Validity

• Is the model reasonable on its face, i.e., without deep inspection or analysis
• Does behavior of model change in expected ways with modification of parameters?

• Example:
  – For an $M/M/c$ queue, we’ll hypothesize that the server utilization is:
    \[ \rho = \frac{\mu}{c\lambda} \]
Face Validity

• Is the model reasonable on its face, i.e., without deep inspection or analysis
• Does behavior of model change in expected ways with modification of parameters?

• Example:
  – For an $M/M/c$ queue, we’ll hypothesize that the server utilization is:

\[
\rho = \frac{\mu}{c \lambda}
\]

  • Utilization is a dimensionless quantity - since $\lambda$ and $\mu$ are rates, and $c$ is a number, this passes the first test.
Face Validity

• Is the model reasonable on its face, i.e., without deep inspection or analysis
• Does behavior of model change in expected ways with modification of parameters?

• Example:
  – For an $M/M/c$ queue, we’ll hypothesize that the server utilization is:
    \[ \rho = \frac{\mu}{c \lambda} \]
    • Utilization is a dimensionless quantity - since $\lambda$ and $\mu$ are rates, and $c$ is a number, this passes the first test.
    • What happens to server utilization as the arrival rate, $\lambda$, or service rate, $\mu$, increase, or as we change the number of servers, $c$?
Face Validity

• Is the model reasonable on its face, i.e., without deep inspection or analysis
• Does behavior of model change in expected ways with modification of parameters?

• Example:
  – For an $M/M/c$ queue, we’ll hypothesize that the server utilization is:

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

• Utilization is a dimensionless quantity - since $\lambda$ and $\mu$ are rates, and $c$ is a number, this passes the first test.
• What happens to server utilization as the arrival rate, $\lambda$, or service rate, $\mu$, increase, or as we change the number of servers, $c$?
  • $\rho$ increases with increasing $\mu$, and decreases with increasing $\mu$, which is obviously incorrect. The change with changing $c$ is appropriate.
  • This model is invalid on its face.
Validation of Model Assumptions

Model assumptions

Structural assumptions

- System operation
- Simplifying assumptions
- Abstractions of reality

Based on:
- Observation

Data assumptions

- Homogeneity
- Independence
- Distribution

Based on:
- Collection of reliable data
- Correct statistical analysis
Validating Input-Output Transformations

- Consider real system and simulation to be functional transformations of inputs, based on parameter settings

- To what extent does simulation:
  - agree with real system
  - accurately predict future behavior of real system
  - accurately predict behavior of real system under different input conditions or parameters?

- This is the only objective measure of the model as a whole

- Usable for comparison only if there is something to compare to!
The Turing Test

- If expert observer cannot distinguish output from real and simulated system with any consistency, validity of simulation is not refuted by this test
- If observer consistently identifies simulation, their critical observations can be used to improve simulation
Summary

• The quality of decisions made, based on a simulation and its underlying model, are only as good as the validity of the simulation and the model.

• Possible validation techniques:
  – Develop models with high face validity, use existing knowledge
  – Conduct statistical tests on input
  – Conduct a Turing test
  – Compare model to real system output with statistical tests
  – Build new system based on simulation, collect data and use to validate model \( (a.k.a. \text{ a cart-horse ordering problem}) \)
  – Do little or no validation \( (Is\ this\ your\ lucky\ day?) \)
Homework

• Within the context of your class project describe:
  – Examples of input processes or system operation that you will be modeling
  – How you will be validating the model
  – How you will be verifying the model
  – How you will be verifying the overall simulation project.

  – Collaboration with the other project team members to discuss the project model and simulation code is expected, but the description of the processes used for validation and verification must be individual

• Separate submission (graded as part of class project - one per group):
  – Provide a flow chart of the tasks you have completed and must still complete for your project (e.g., data collection, analysis, etc.). Indicate completion status by color coding task blocks.
  – Provide a flow chart of your simulation software
  – Each chart should be ~1 page.