Selection of Techniques and Metrics

Raj Jain
Washington University in Saint Louis
Saint Louis, MO 63130
Jain@cse.wustl.edu

These slides are available on-line at:
http://www.cse.wustl.edu/~jain/cse567-15/
Overview

- Criteria for Selecting an Evaluation Technique
- Three Rules of Validation
- Selecting Performance Metrics
- Commonly Used Performance Metrics
- Utility Classification of Metrics
- Setting Performance Requirements
## Criteria for Selecting an Evaluation Technique

<table>
<thead>
<tr>
<th>Criterion</th>
<th>Analytical Modeling</th>
<th>Simulation Medium</th>
<th>Measurement Postprototype</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Stage</td>
<td>Any</td>
<td>Any</td>
<td></td>
</tr>
<tr>
<td>2. Time required</td>
<td>Small</td>
<td>Medium</td>
<td>Varies</td>
</tr>
<tr>
<td>3. Tools</td>
<td>Analysts</td>
<td>Computer languages</td>
<td>Instrumentation</td>
</tr>
<tr>
<td>4. Accuracy&lt;sup&gt;a&lt;/sup&gt;</td>
<td>Low</td>
<td>Moderate</td>
<td>Varies</td>
</tr>
<tr>
<td>5. Trade-off evaluation</td>
<td>Easy</td>
<td>Moderate</td>
<td>Difficult</td>
</tr>
<tr>
<td>6. Cost</td>
<td>Small</td>
<td>Medium</td>
<td>High</td>
</tr>
<tr>
<td>7. Saleability</td>
<td>Low</td>
<td>Medium</td>
<td>High</td>
</tr>
</tbody>
</table>

<sup>a</sup> In all cases, result may be misleading or wrong.
Three Rules of Validation

- Do not trust the results of a **simulation model** until they have been validated by analytical modeling or measurements.

- Do not trust the results of an **analytical model** until they have been validated by a simulation model or measurements.

- Do not trust the results of a **measurement** until they have been validated by simulation or analytical modeling.
Selecting Performance Metrics
Selecting Metrics

- **Include:**
  - Performance Time, Rate, Resource
  - Error rate, probability
  - Time to failure and duration

- **Consider including:**
  - Mean and variance
  - Individual and Global

- **Selection Criteria:**
  - Low-variability
  - Non-redundancy
  - Completeness
Case Study: Two Congestion Control Algorithms

- Service: Send packets from specified source to specified destination in order.

- Possible outcomes:
  - Some packets are delivered in order to the correct destination.
  - Some packets are delivered out-of-order to the destination.
  - Some packets are delivered more than once (duplicates).
  - Some packets are dropped on the way (lost packets).
Case Study (Cont)

- Performance: For packets delivered in order,
  - Time-rate-resource $\Rightarrow$
    - Response time to deliver the packets
    - Throughput: the number of packets per unit of time.
    - Processor time per packet on the source end system.
    - Processor time per packet on the destination end systems.
    - Processor time per packet on the intermediate systems.
  - Variability of the response time $\Rightarrow$ Retransmissions
    - Response time: the delay inside the network
Case Study (Cont)

- Out-of-order packets consume buffers
  ➔ Probability of out-of-order arrivals.
- Duplicate packets consume the network resources
  ➔ Probability of duplicate packets
- Lost packets require retransmission
  ➔ Probability of lost packets
- Too much loss cause disconnection
  ➔ Probability of disconnect
Case Study (Cont)

- Shared Resource ⇒ Fairness

\[ f(x_1, x_2, \ldots, x_n) = \frac{\left(\sum_{i=1}^{n} x_i\right)^2}{n \sum_{i=1}^{n} x_i^2} \]

- Fairness Index Properties:
  - Always lies between 0 and 1.
  - Equal throughput ⇒ Fairness = 1.
  - If \( k \) of \( n \) receive \( x \) and \( n-k \) users receive zero throughput: the fairness index is \( k/n \).
Case Study (Cont)

- Throughput and delay were found redundant ⇒ Use Power.

\[
\text{Power} = \frac{\text{Throughput}}{\text{Response Time}}
\]

- Variance in response time redundant with the probability of duplication and the probability of disconnection

- Total nine metrics.
Commonly Used Performance Metrics

- Response time and Reaction time

![Diagram showing the flow of users' request and system's response with response time indicated]
Response Time (Cont)

User starts request
User finishes request
System starts execution
System starts response
System completes response
User next request

----

Reaction time
Response time
(Definition 1)

Response time
(Definition 2)
Capacity

Throughput vs. Load

- Knee capacity
- Nominal capacity
- Usable capacity

Response time vs. Load

- Knee

Washington University in St. Louis
http://www.cse.wustl.edu/~jain/cse567-15/
©2015 Raj Jain
Common Performance Metrics (Cont)

- **Nominal Capacity**: Maximum achievable throughput under ideal workload conditions. E.g., bandwidth in bits per second. The response time at maximum throughput is too high.

- **Usable capacity**: Maximum throughput achievable without exceeding a pre-specified response-time limit.

- **Knee Capacity**: Knee = Low response time and High throughput
Common Performance Metrics (Cont)

- **Turnaround time** = the time between the submission of a batch job and the completion of its output.

- **Stretch Factor**: The ratio of the response time with multiprogramming to that without multiprogramming.

- **Throughput**: Rate (requests per unit of time) Examples:
  - Jobs per second
  - Requests per second
  - Millions of Instructions Per Second (MIPS)
  - Millions of Floating Point Operations Per Second (MFLOPS)
  - Packets Per Second (PPS)
  - Bits per second (bps)
  - Transactions Per Second (TPS)
Common Performance Metrics (Cont)

- **Efficiency**: Ratio usable capacity to nominal capacity. Or, the ratio of the performance of an $n$-processor system to that of a one-processor system is its efficiency.

- **Utilization**: The fraction of time the resource is busy servicing requests. Average fraction used for memory.
Common Performance Metrics (Cont)

- **Reliability:**
  - Probability of errors
  - Mean time between errors (error-free seconds).

- **Availability:**
  - Mean Time to Failure (MTTF)
  - Mean Time to Repair (MTTR)
  - $\frac{MTTF}{MTTF+MTTR}$
Utility Classification of Metrics

(a) Lower is better

(b) Higher is better

(c) Nominal is best
Setting Performance Requirements

- **Examples:**
  
  “The system should be both processing and memory efficient. It should not create excessive overhead”

  “There should be an extremely low probability that the network will duplicate a packet, deliver a packet to the wrong destination, or change the data in a packet.”

- **Problems:**
  
  Non-Specific
  Non-Measurable
  Non-Acceptable
  Non-Realizable
  Non-Thorough

  \[ \Rightarrow \text{SMART} \]
Case Study 3.2: Local Area Networks

- **Service**: Send frame to D

- **Outcomes**:
  - Frame is correctly delivered to D
  - Incorrectly delivered
  - Not delivered at all

- **Requirements**:

  - **Speed**
    - The access delay at any station should be less than one second.
    - Sustained throughput must be at least 80 Mbits/sec.

  - **Reliability**: Five different error modes.
    - Different amount of damage
    - Different level of acceptability.
Case Study (Cont)

- The probability of any bit being in error must be less than 1E-7.
- The probability of any frame being in error (with error indication set) must be less than 1%.
- The probability of a frame in error being delivered without error indication must be less than 1E-15.
- The probability of a frame being misdelivered due to an undetected error in the destination address must be less than 1E-18.
- The probability of a frame being delivered more than once (duplicate) must be less than 1E-5.
- The probability of losing a frame on the LAN (due to all sorts of errors) must be less than 1%.
Case Study (Cont)

- **Availability**: Two fault modes – Network reinitializations and permanent failures
  - The mean time to initialize the LAN must be less than 15 milliseconds.
  - The mean time between LAN initializations must be at least one minute.
  - The mean time to repair a LAN must be less than one hour. (LAN partitions may be operational during this period.)
  - The mean time between LAN partitioning must be at least one-half a week.
Summary of Part I

- **Systematic Approach**: Define the system, list its services, metrics, parameters, decide factors, evaluation technique, workload, experimental design, analyze the data, and present results.

- **Selecting Evaluation Technique**: The life-cycle stage is the key. Other considerations are: time available, tools available, accuracy required, trade-offs to be evaluated, cost, and saleability of results.
Summary (Cont)

- **Selecting Metrics:**
  - For each service list time, rate, and resource consumption
  - For each undesirable outcome, measure the frequency and duration of the outcome
  - Check for low-variability, non-redundancy, and completeness.

- **Performance requirements:** Should be SMART. Specific, measurable, acceptable, realizable, and thorough.
Homework 3: Exercise 3.1

What methodology would you choose:
   a. To select a personal computer for yourself?
   b. To select 1000 workstations for your company?
   c. To compare two spread sheet packages?
   d. To compare two data-flow architectures?

if the answer was required:
   i. Yesterday?
   ii. Next quarter?
   iii. Next year?

Prepare a table of 12 entries. Write 1 line explanation of each of 12 choices.

Common Mistake: Not specifying all 12 combinations.