Communications Network Simulation

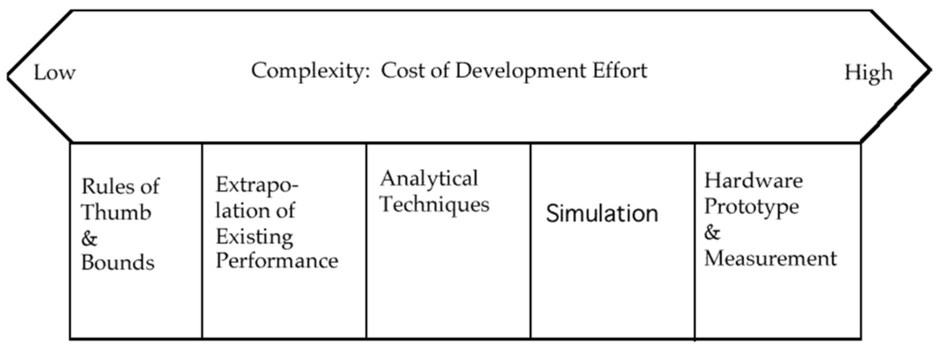
Victor S. Frost
Dan F. Servey Distinguished Professor
Electrical Engineering and Computer Science
University of Kansas
2291 Irving Hill Dr.
Lawrence, Kansas 66045
Phone: (785) 864-1028
e-mail: vsfrost@ku.edu
http://www.ittc.ku.edu/

Communications Network Simulation: Overview

- Introduce Discrete Event Simulation and Approaches
- Building Extendsim models
- Discuss Verification and Validation of Communication Network Simulation Models
- Deriving statistically significant results from simulation models
- Discussion of Statistical Considerations in the Analysis of Results from Communication Network Simulation Models
- Communication Network Simulation Modeling Tools and Examples

Communications Network Simulation

Evaluation Techniques



Communications Network Simulation: Attributes

- Simulation can be used to model general communications network in minute detail.
- Simulation models can be expensive to construct.
- Simulation models can be expensive to run.
- Statistical analysis of the results generated by a simulation can be difficult.
- It can be difficult to gain general insights into system behavior based on simulation results.

Communications Network Simulation: When to Use Simulation

- For studying transient behavior of networks.
- For systems with adaptive routing.
- For systems with adaptive flow control.
- For systems with blocking (finite buffers).

Communications Network Simulation: When to Use Simulation

- For systems with general message interarrival statistics.
- Validation of analytic models and approximations.
- For experimentation without disturbing an operational system.

Common Mistakes in Simulation

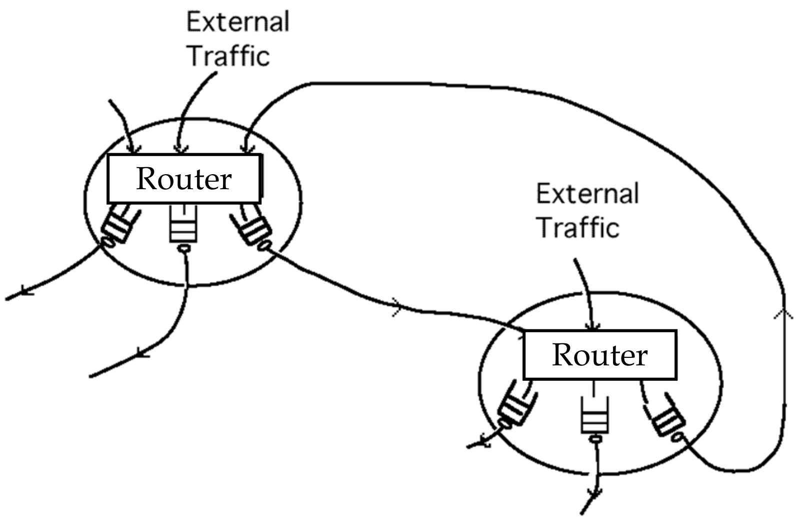
- Inappropriate Level of Detail:
  - More detail ⇒ More time ⇒ More Bugs ⇒ More CPU
  - More parameters ≠ More accurate
- Unverified Models: Bugs
- Invalid Models: Model vs. reality
- Improperly Handled Initial Conditions
- Too Short Simulations: Need confidence intervals
- Poor Random Number Generators: Safer to use a well-known generator
- Improper Selection of pseudo random number seeds

Common Mistakes in Simulation

- Inadequate Estimate of Development Time & Effort
- Unclear Goal: Inadequate framing of question
- Project Team has Incomplete Mix of Essential Skills: Team Lacks-
  - Project Leadership
  - Modeling and Programming
  - Knowledge of the Target System
  - Statistical Analysis
- Inadequate Level of User Participation
- Lack of Planning for Success

Communications Network Simulation:

Modeling Elements for Networks

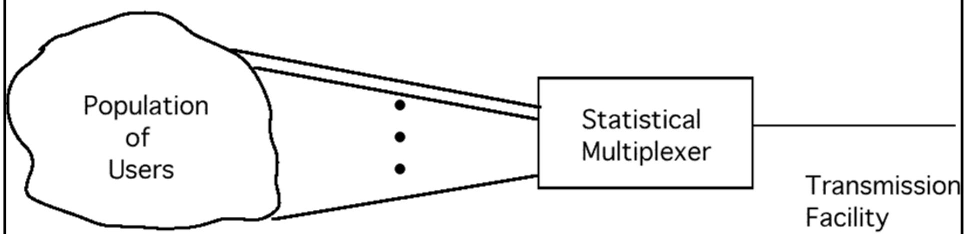


Communications Network Simulation:

Definition of Communication Network Simulation

Communication network simulation involves generating ***pseudo-random sequences*** of message lengths an interarrival times (or other input processes, e.g. time varying link quality) the using these sequences to ***exercise an algorithmic description of the network operation.***

Communications Network Simulation:
Example of a Single-Channel Statistical Multiplexer



Traffic & Input Processes

- Message Length - L_k's
  - E.g., Exponential pdf
- Message Interarrival Times - A_k's
  - E.g., Exponential pdf
- Other
  - Time of day variations
  - BER vs. Time
  - Link Reliability
  - Node Reliability

Sample Realization of an Input Process

Message number	1	2	3	4	5	6	7	8	9	10	11	12
Interarrival time between i+1 and i message (seconds)	2	1	3	1	1	4	2	5	1	4	2	--
Length of i^{th} message (seconds)	1	3	6	2	1	1	4	2	5	1	1	3

Communications Network Simulation:

- Approaches to Discrete Event Simulation
  - Time Step Approach
  - Event-Scheduling Approach
- Network Modeling Approaches
  - Extended Queueing Networks
  - Finite State Machine
  - Petri Nets
  - Data Flow Block Diagram

Communications Network Simulation:
Time Step Approach to Network Simulation

- Fixed-Increment Time Advance
- Update System States at End of Each Fixed Time Interval

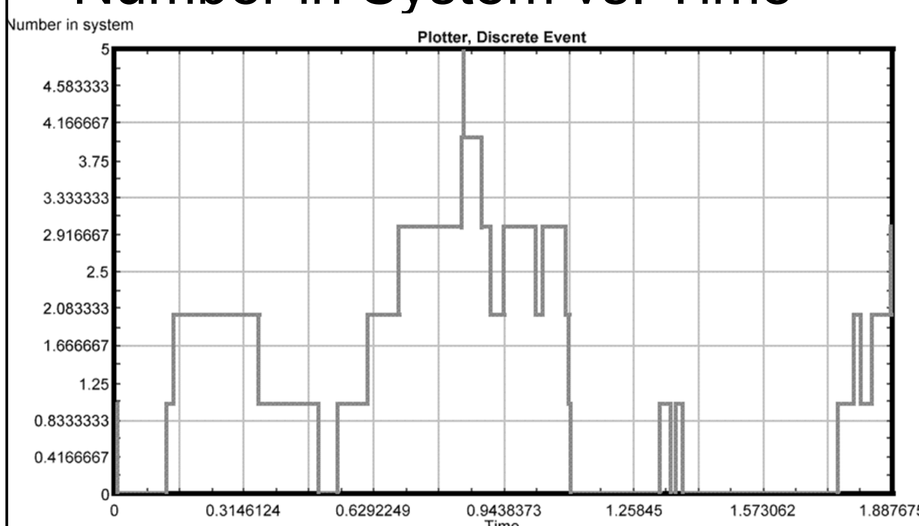
Communications Network Simulation:
Sample Realization of an Input Process

Message number	1	2	3	4	5	6	7	8	9	10	11	12
Interarrival time between i+1 and i message (seconds)	2	1	3	1	1	4	2	5	1	4	2	--
Length of i^{th} message (seconds)	1	3	6	2	1	1	4	2	5	1	1	3

Example-Time History of Time Step Analysis of Statistical Multiplexer

Simulated Time	Message Arrival	Start of Transmission	End of Transmission	Number in Buffer	Number in System	Time in Buffer	Time in System
0	(1)	(1)		0	1	0, (1)	
1			(1)	0	0		1, (1)
2	(2)	(2)		0	1	0, (2)	
3	(3)			1	2		
4				1	2		
5		(3)	(2)	0	1	2, (3)	3, (2)
6	(4)			1	2		
7	(5)			2	3		
8	(6)			3	4		
9				3	4		
10				3	4		
11		(4)	(3)	2	3	5, (4)	8, (3)
12	(7)			3	4		
13		(5)	(4)	2	3	6, (5)	7, (4)
14	(8)	(6)	(5)	2	3	6, (6)	7, (5)
15		(7)	(6)	1	2	3, (7)	7, (6)
16				1	2		
17				1	2		
18				1	2		
19		(8)	(7)	1	2	5, (8)	7, (7)
20				2	3		
21		(9)	(8)	1	2	2, (9)	7, (8)
22				1	2		
23				1	2		
24	(11)			2	3		
25				2	3		
26	(12)	(10)	(9)	2	3	6, (10)	7, (9)

Number in System vs. Time



Performance Metrics Derived from Time History

$$\text{Average Delay} = \frac{\sum \text{Time in System for i}^{\text{th}} \text{ Message}}{\text{Total Number of Messages Processed}}$$

$$\text{Average number in System} = \frac{\sum N(t)}{\text{Total time}}$$

Communications Network Simulation:

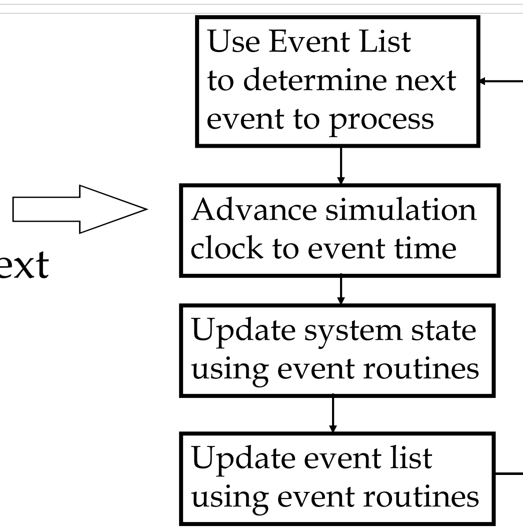
- Event Scheduling Approach
  - Variable Time Advance
  - Update System State Only When Events Occur, e.g. Arrivals or Departures

Communications Network Simulation: Event Calendar

- Events are Instantaneous Occurrences Which Change the State of the System
- An Event is Described by
  - The time the event is to occur
  - The action to take place at the event time
- The Event Calendar is a Time Ordered List of Events

Simplified Flow Control for the Event Scheduling Approach

An Executive (or Mainline) Controls the Selection of Next Event



Entities, Attributes, Activities, and Files

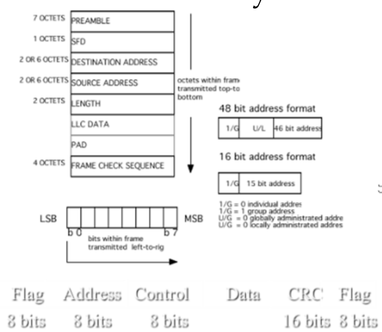
- Entities (items) are the objects upon which action is performed. In network simulation entities are messages.
- Attributes are characteristics which describe entities, e.g. message length or message type.

Entities, Attributes, Activities, and Files

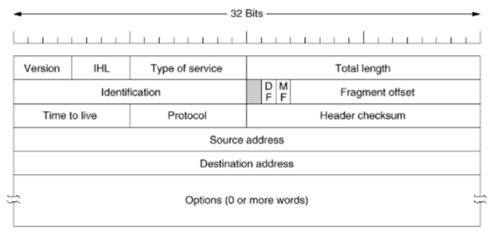
- Activities are the operations that change the state of the network, e.g. increment number of messages waiting in a buffer.
- Files are groupings of entities which share a common attribute, all messages waiting in buffer.

Network Packet Data Structure (Attributes)

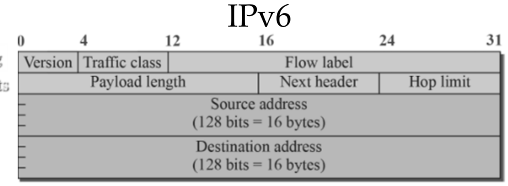
Data Link Layer



IPv4

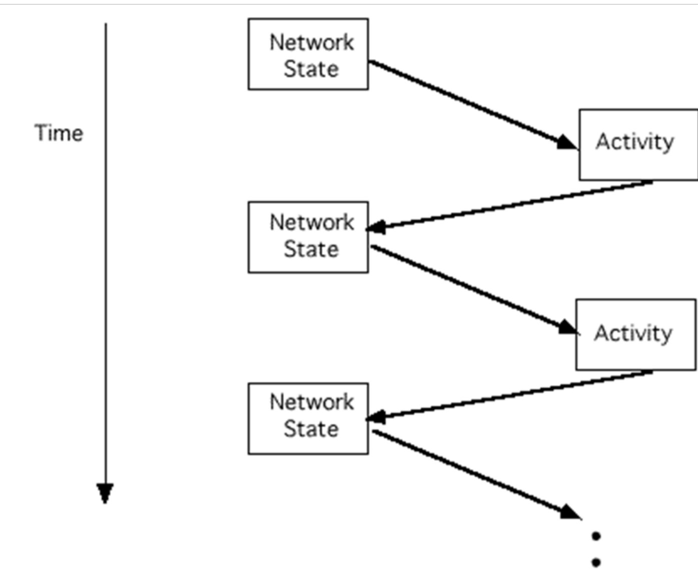


IPv6

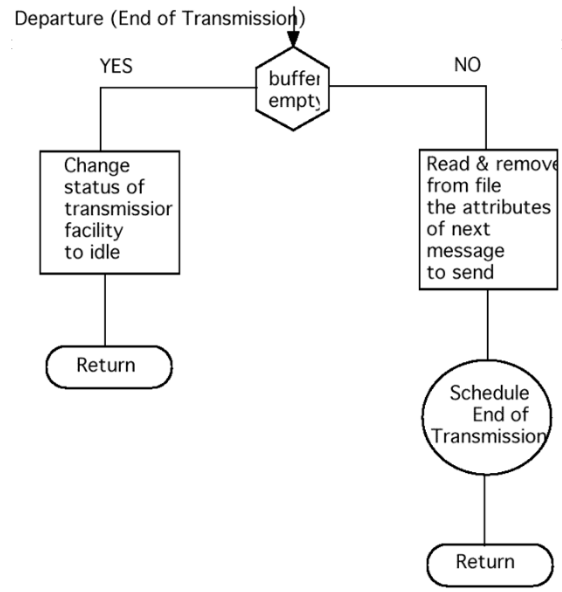


b. Base header

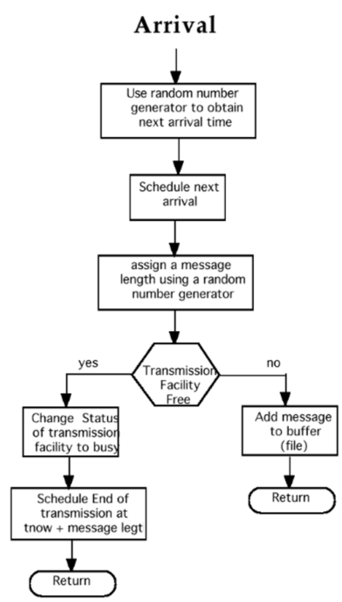
Dynamics of a Discrete Event Simulation



Example - Discrete Event Simulation of a Statistical Multiplexer



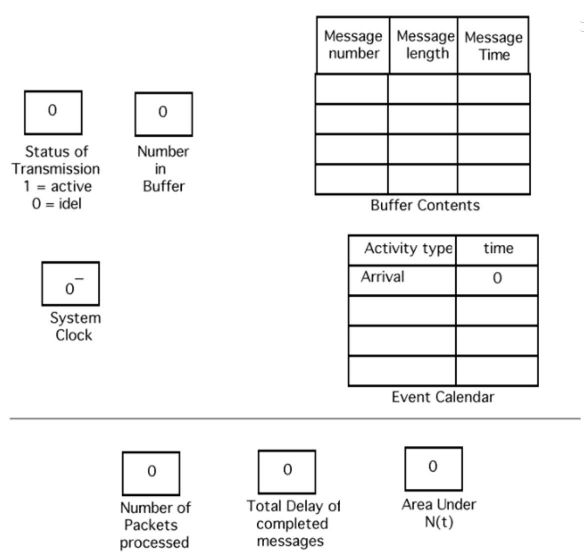
Example - Discrete Event Simulation of a Statistical Multiplexer



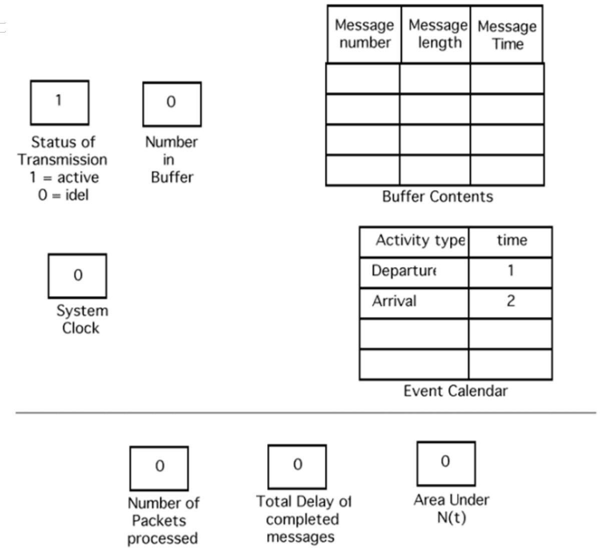
Example - Discrete Event Simulation of a Statistical Multiplexer

Message number	1	2	3	4	5	6	7	8	9	10	11	12
Interarrival time between i+1 and i message (seconds)	2	1	3	1	1	4	2	5	1	4	2	--
Length of i^{th} message (seconds)	1	3	6	2	1	1	4	2	5	1	1	3

Example - Discrete Event Simulation of a Statistical Multiplexer



Example - Discrete Event Simulation of a Statistical Multiplexer



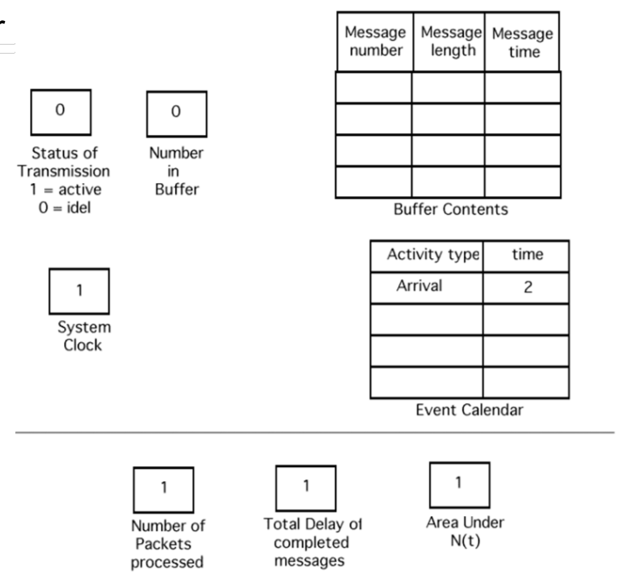
Buffer Contents

Message number	Message length	Message Time

Event Calendar

Activity type	time
Departur	1
Arrival	2

Example - Discrete Event Simulation of a Statistical Multiplexer



Buffer Contents

Message number	Message length	Message time

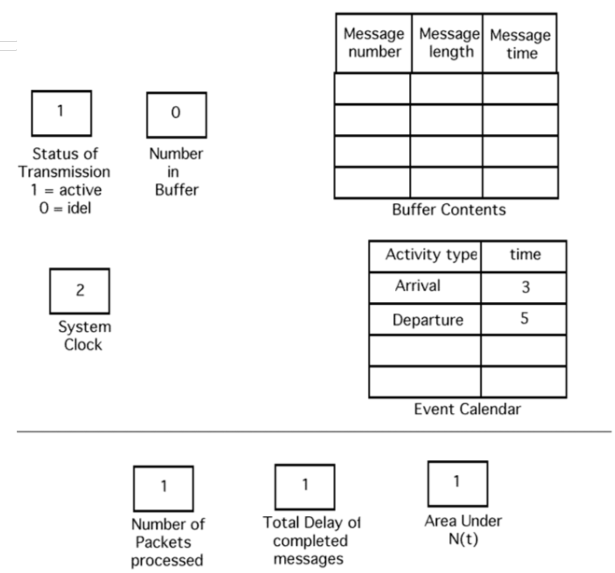
Event Calendar

Activity type	time
Arrival	2

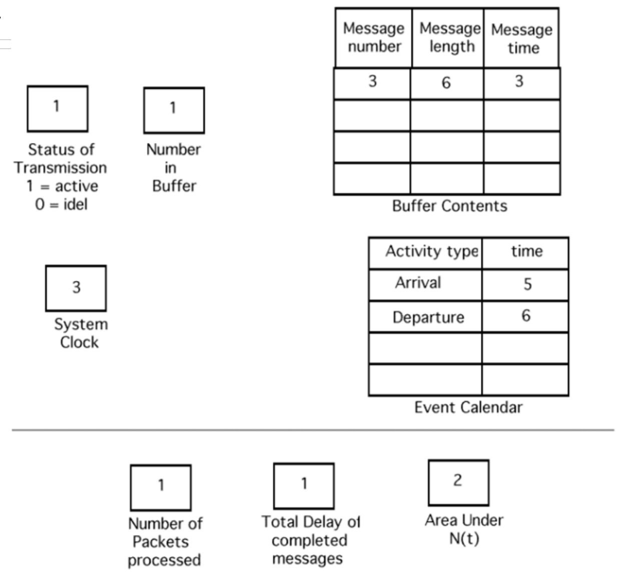
Example - Discrete Event Simulation of a Statistical Multiplexer

Message number	1	2	3	4	5	6	7	8	9	10	11	12
Interarrival time between i+1 and i message (seconds)	2	1	3	1	1	4	2	5	1	4	2	--
Length of i^{th} message (seconds)	1	3	6	2	1	1	4	2	5	1	1	3

Example - Discrete Event Simulation of a Statistical Multiplexer



Example - Discrete Event Simulation of a Statistical Multiplexer



Example - Discrete Event Simulation of a Statistical Multiplexer

Message number	1	2	3	4	5	6	7	8	9	10	11	12
Interarrival time between i+1 and i message (seconds)	2	1	3	1	1	4	2	5	1	4	2	--
Length of i^{th} message (seconds)	1	3	6	2	1	1	4	2	5	1	1	3

Example - Discrete Event Simulation of a Statistical Multiplexer

1 — Status of Transmission (1 = active, 0 = idel)

0 — Number in Buffer

5 — System Clock

Message number	Message length	Message time

Buffer Contents

Activity type	time
Arrival	6
Departure	11

Event Calendar

2 — Number of Packets processed

4 — Total Delay of completed messages

6 — Area Under N(t)

Example - Discrete Event Simulation of a Statistical Multiplexer

1 — Status of Transmission (1 = active, 0 = idel)

1 — Number in Buffer

6 — System Clock

Message number	Message length	Message time
4	2	6

Buffer Contents

Activity type	time
Arrival	7
Departure	11

Event Calendar

2 — Number of Packets processed

4 — Total Delay of completed messages

7 — Area Under N(t)

Example - Discrete Event Simulation of a Statistical Multiplexer

Message number	1	2	3	4	5	6	7	8	9	10	11	12
Interarrival time between i+1 and i message (seconds)	2	1	3	1	1	4	2	5	1	4	2	--
Length of i^{th} message (seconds)	1	3	6	2	1	1	4	2	5	1	1	3

Example - Discrete Event Simulation of a Statistical Multiplexer

1
Status of Transmission
1 = active
0 = idel

2
Number in Buffer

7
System Clock

Message number	Message length	Message time
4	2	6
5	1	7

Buffer Contents

Activity type	time
Arrival	8
Departure	11

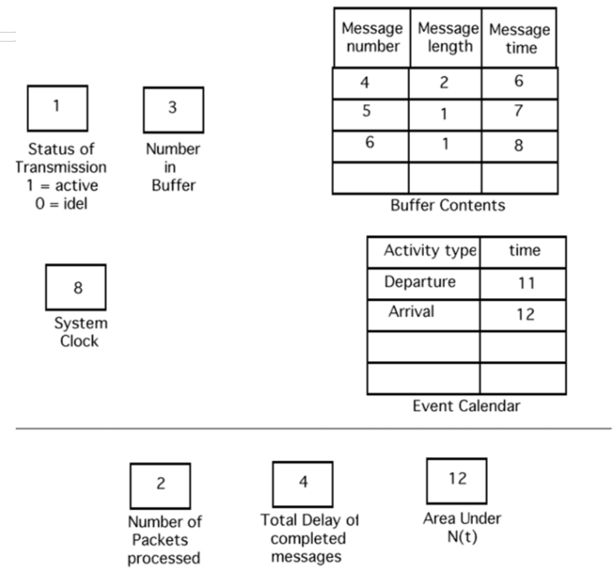
Event Calendar

2
Number of Packets processed

4
Total Delay of completed messages

9
Area Under N(t)

Example - Discrete Event Simulation of a Statistical Multiplexer



Relative Merits of Time Step and Event Scheduling

Approach	**Advantages**	**Disadvantages**
Time Step	Efficient for system with very frequently occurring events Efficient for regularly spaced events	Must process at each time step Error induced by fixed finite time increment Must establish rules to order events that occur in same time increment
Event Scheduling	Only process at event times No time increment to select Flexible	Significant programming effort required

Introduction to Extend

- Allows graphical description of networks
- Data flow block diagrams
- Hierarchical structure to control complexity

Example:
M/M/1 Simulation in Extend

- Execute Extend 10
- Add traffic source
- Add queue
- Add server
- Add data structure
- Add performance measurements
- Make Hierarchical block
- Sensitize parameter and show multiple run.
- Ehance model after discussion on quality of measurements

Verification and Validation of Communication Network Simulation Models

- Verification is determining whether the simulation model performs as intended
- Validation is determining whether the simulation model is a "accurate" representation of the communication network under study

Verification Methods

- Modular development and verification
- Structured walk-through
- Event trace

Verification Methods

- Model simplification and comparison to analytic results
- Graphical display of network status as the model progresses

Some Comments on Validation

- Simulation models are always approximations
- A simulation model developed for one application may not be valid for others
- Model development and validation should be done simultaneously

Verify M/M/1 Extend Simulation Model

Some Comments on Validation

- Specific modeling assumptions should be tested
- Sensitivity analysis should be performed
- Attempt to establish that the model results resemble the expected performance of the actual system; expert analysis

Some Comments on Validation

"For models which attempt to describe *new* systems where *no historical data* are available and which are of such a nature that there are *no input conditions* that correspond to known analytical results, the problem of model validation is, indeed, extremely difficult. About all one can do in this situation is to recheck the logic of the design, run the model over a range of different inputs to determine whether the outputs are within the realm of plausibility, and, if one is so inclined, **pray a lot**."

(From D. Gross and C.M. Harris, "Fundamentals of Queueing Theory," John Wiley & Sons, Inc., New York, 1985)

Statistical Considerations: Analysis of Results from Communication Network Simulation

- Quality of Performance Estimates
  - Variance of estimated performance measures
  - Confidence Intervals (CI)
  - Relative Error (RE)
- Starting Rules
  - Overcoming initial transients
  - Ignore initial transient
- Stopping Rules
  - Set fixed simulation end time
  - Run simulation until specified CI (or RE) met

Quality of the Performance Estimates

- Performance estimates should be unbiased
- Performance estimates should have "acceptable" error
  - Variance (standard deviation)
  - Confidence interval
  - Relative Error

Quality of the Performance Estimate

- The desired confidence interval width (relative error) determines the length of the simulation run
- Observations tend to be correlated: cannot directly apply standard statistical approaches based on iid observations

Introduction to Estimation

- Common pdf is a Gaussian with two parameters;
  - Mean (expected value) μ
  - Standard deviation σ

$$N(x;\mu,\sigma)=\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- Generalize the notation for any pdf with M parameters. $f_X(x;\theta_1,\theta_2,..\theta_M)$
- Example: Exponential pdf

$$f_X(x;\theta)=\frac{1}{\theta}e^{-\frac{x}{\theta}} \quad x>0$$
$$f_X(x;\theta)=0 \quad x<0$$

Introduction to Estimation

- Goal:
  - Given N samples (observations) of the random variable X with $f_X(x;\theta)$, $x_1, x_2, x_3, x_4, \ldots x_N$
    find –estimate- θ.
- Example: X is normal with known variance (σ=1) and unknown expected value E[X]

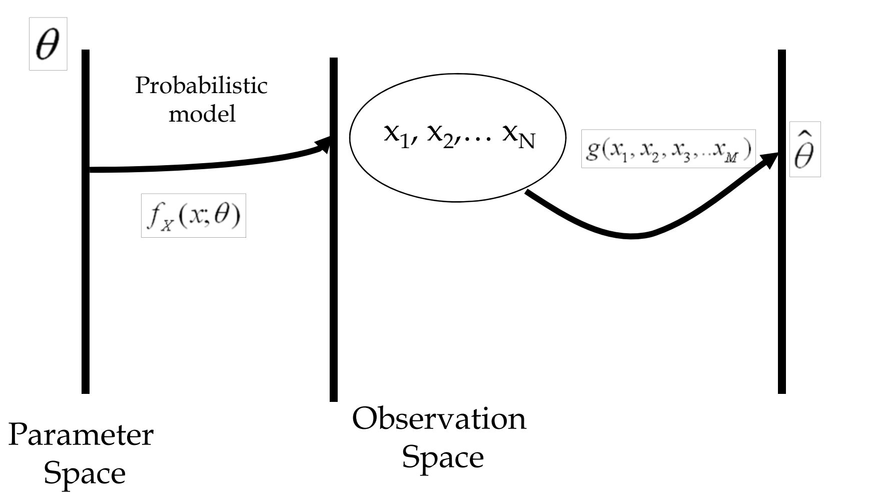
$$N(x;\theta)=\frac{1}{\sqrt{2\pi}}e^{-\frac{(x-\theta)^2}{2}}$$

- Here θ=E[X] and is unknown

Introduction to Estimation

- Goal: find a function g(…) that maps N observations $x_1, x_2, x_3, x_4, \ldots x_N$ into a "good" estimate for θ, i.e., find
$$\hat{\theta} = g(x_1, x_2, x_3, ..x_M)$$
- Before the measurement, each observation is a random variable X_i
- Typically the RV's X_i are assumed to be i.i.d.
- Now $\hat{\theta}$ is a RV.
- The properties of $\hat{\theta}$ determine the "goodness" of the estimate
$$\hat{\theta} = g(X_1, X_2, X_3, ..X_M)$$

Introduction to Estimation



Introduction to Estimation

- Criteria for good estimators g(…)
  - Unbiased $\hat{\theta}$ is an unbiased estimate for θ if

$$E[\hat{\theta}] = \theta$$

  - The sample mean is an unbiased estimator for the expected value

$$\hat{\theta} = \frac{1}{N}\sum_{i=1}^{N} X_i$$

$$E[\hat{\theta}] = E[\frac{1}{N}\sum_{i=1}^{N} X_i] = \frac{1}{N}\sum_{i=1}^{N} E[X_i] = E[X]$$

Introduction to Estimation

- Example: Binomial RV
  - Probability of exactly k successes in n trials, where probability success on one trial = p

$$\text{Prob}(k; n, p) = \binom{n}{k} p^k (1-p)^{n-k}$$

  - Observing x successes in n trials results in the estimator for p $\hat{p} = \frac{x}{n}$

Introduction to Estimation

- For $g(x_1, x_2, x_3, ..x_M)$ to be a "good" estimator the variation about the desired parameter should be small, i.e.,
  $E[|\widehat{\theta} - \theta|]$ or $E[(\widehat{\theta} - \theta)^2]$ or θ_{Max} - θ_{Min} should be small.
- The most common measure of variation is the
  Mean Square Error

$$MSE = E[(\widehat{\theta} - \theta)^2] = \sigma_{\hat{\theta}}^2 - (\theta - E[\hat{\theta}])^2$$

$$\text{if } \hat{\theta} \text{ is unbiased then } MSE = \sigma_{\hat{\theta}}^2$$

- The variance of the estimator is the MSE

Introduction to Estimation

- The variance of the sample mean is

$$\hat{\theta} = \frac{1}{N}\sum_{i=1}^{N} X_i$$

$$E[\hat{\theta}] = E[\frac{1}{N}\sum_{i=1}^{N} X_i] = \frac{1}{N}\sum_{i=1}^{N} E[X_i] = E[X]$$

$$\sigma_{\hat{\theta}}^2 = \frac{1}{N}\sigma_X^2$$

$$\text{Note } \lim_{N\to\infty} \sigma_{\hat{\theta}}^2 = 0$$

Introduction to Estimation

- Estimator for the variance

$$f_X(x;\theta_1,\theta_2) \text{ where } \theta_1 = E[X] \text{ and } \theta_2 = \sigma_X^2$$

$$\widehat{\theta}_1 = \overline{X} = \frac{1}{N}\sum_{k=1}^{N} X_k$$

$$\widehat{\theta}_2 = \frac{1}{N-1}\sum_{k=1}^{N}(X_k - \overline{X})^2 = \text{Sample variance} = s^2$$

$$E[s^2] = \sigma_X^2 \text{ so } s^2 \text{ is an unbiased estimator for the variance}$$

Confidence Interval

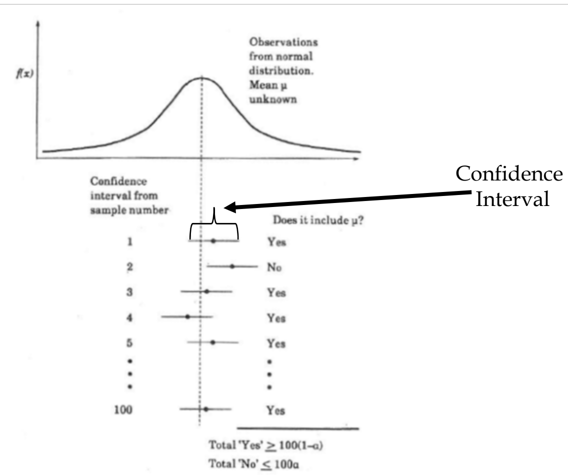


FIGURE 13.1 Meaning of a confidence interval.

From: "The Art of Computer Systems Performance Analysis" Raj Jain, Wiley, 1991

Confidence Interval: Example

- m=25 samples of a RV X are collected $x_1, x_2, x_3, \ldots x_m$
- What is the probability that the true mean is in interval $\overline{X} - 3.92 < \mu < \overline{X} + 3.92$
- Assumptions:
  - 25 is large enough such at the sample mean is Gaussian.
  - The variance of X is ***known*** and $\sigma^2 = 100$

Confidence Interval: Example

- Next find the standard deviation of the sample mean

$$E[\overline{X}] = \mu \text{ and } \mathrm{Var}[\overline{X}] = \frac{\sigma^2}{m} = \frac{\sigma^2}{25}$$

$$\sigma_{\overline{X}} = \frac{\sigma}{\sqrt{m}} = 2$$

$$P(\overline{X} - 3.92 < \mu < \overline{X} + 3.92) = P(-3.92 < \mu - \overline{X} < 3.92)$$

$$E[\mu - \overline{X}] = 0 \text{ and } Var[\mu - \overline{X}] = Var[\overline{X}] = 4$$

so

$$\frac{\mu - \overline{X}}{2} \sim N(0,1) \text{ and}$$

$$P(\overline{X} - 3.92 < \mu < \overline{X} + 3.92) = P(-3.92 < \mu - \overline{X} < 3.92) = P(\frac{-3.92}{2} < \frac{\mu - \overline{X}}{2} < \frac{3.92}{2})$$

$$= P(-1.96 < Z < 1.96) \text{ where } Z \sim N(0,1)$$

Confidence Interval: Example

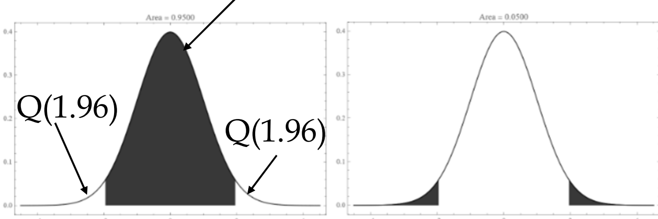
$P(-1.96 < Z < 1.96)$ where $Z \sim N(0,1)$

$$P(-1.96 < Z < 1.96) = \frac{1}{\sqrt{2\pi}} \int_{-1.96}^{1.96} e^{\frac{-z^2}{2}} dz$$

Define $Q(x) = \frac{1}{\sqrt{2\pi}} \int_{x}^{\infty} e^{\frac{-z^2}{2}} dz$

$P(-1.96 < Z < 1.96) = 1 - 2Q(1.96) = 0.95$

$P(-1.96 < Z < 1.96) = 1 - 2Q(1.96) = 0.95$



https://demonstrations.wolfram.com/StandardNormalDistributionAreas/

Confidence Interval: Example

- If the sample mean = 55 then there is a 95% probability that the true mean is between ~51 and 59.
- If the sample mean = 60 then there is a 95% probability that the true mean is between ~56 and 64.

Confidence Interval

$$P(1.96 < \frac{\mu - \overline{X}}{\frac{\sigma}{\sqrt{m}}} < 1.96) = 1 - \alpha = 0.95$$

$$\alpha = 0.05$$

$100(1\text{-}\alpha)$=% confidence

Confidence Interval

- Unknown Variance
- Use the sample variance.
- If a "large" number of samples

  (e.g., m>25) are collected then Gaussian approximations are valid and process is the same.

Confidence Interval

- Small sample size.
- Find the CI using the t-distribution

$$\frac{\mu - \overline{X}}{\frac{\sigma}{\sqrt{m}}} \sim t \text{ distribution with m-1 degrees of freedom}$$

$$\text{Find } P(-1.96 < T < 1.96) \text{ with } T \sim t \text{ distribution with m-1 degrees of freedom}$$

Confidence Interval

- Finding the limits on the CI in general, i.e. the 1-α CI
- Going back to the example: Note Q(1.96)=0.025 or

  z_T=1.96 and Q(z_T) = 0.025

$$P(\overline{X} - 3.92 < \mu < \overline{X} + 3.92)$$
$$3.92 = z_T\sqrt{\frac{s^2}{m}}$$

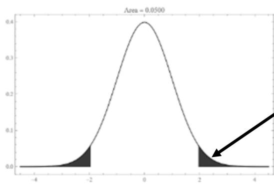
In general

$$P(\overline{X} - z_\alpha\sqrt{\frac{s^2}{m}} < \mu < \overline{X} + z_\alpha\sqrt{\frac{s^2}{m}})$$
$$Q(z_\alpha) = \frac{\alpha}{2}$$

Confidence Interval

$$P(\overline{X} - z_\alpha \sqrt{\frac{s^2}{m}} < \mu < \overline{X} + z_\alpha \sqrt{\frac{s^2}{m}}) \qquad Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{\frac{-z^2}{2}} dz$$

$$Q(z_\alpha) = \frac{\alpha}{2}$$



CI %	α	$\alpha/2$	z_α
99	0.01	0.005	2.57
97.5	0.025	0.0125	2.24
95	0.05	0.025	1.96
90	0.1	0.05	1.64

Confidence Interval

- Example: Find 95% CI for the loss probability given 30 replications of the M/M/1/S with S= 10 & ρ=0.7.
- Data m=30 [show extend model]

0.066524	1.052449	1.233736	0.197161	0.023002	0.564409	1.943335	0.648561	0.761048	0.130256	1.079867	0.742315	0.724162	1.542741	0.206498
1.38573	4.341059	0.478882	0.207289	0.638751	1.066117	0.916407	1.04683	0.837411	0.859991	1.572843	0.067779	0.134131	0.274906	0.169894

- Sample mean = 0.830
- Sample standard deviation = 0.837
- Assuming Gaussian
  - CI 0.830 +/-0.299 0.531<μ<1.13
- Assuming student-t
  - CI 0.830 +/-0.31255   0.5179<μ<1.14303

Confidence Interval

- Finding the 90% CI for m=25 and s^2=100 (assume 25 large enough)
- Going back to the example: Note Q(1.645)=0.05 or

  z_T=1.645

$$z_\alpha \sqrt{\frac{s^2}{m}} = 1.64\sqrt{\frac{100}{25}} = 3.29$$

$$P(\overline{X} - 3.29 < \mu < \overline{X} + 3.29)$$

See: https://demonstrations.wolfram.com/CriticalValueZForZScoresForConfidenceLevels/

Confidence Interval

- As the probability of the true mean being in the CI increases the width of the CI decreases.

$$P(\overline{X} - z_\alpha \sqrt{\frac{s^2}{m}} < \mu < \overline{X} + z_\alpha \sqrt{\frac{s^2}{m}})$$

$$Q(z_\alpha) = \frac{\alpha}{2}$$

See: https://demonstrations.wolfram.com/CriticalValueZForZScoresForConfidenceLevels/

Relative Error

- A quality measure related to the CI is the relative error (RE)

$$RE(\%) = \frac{100 z_\alpha \sqrt{\frac{s^2}{m}}}{\left|\overline{X}\right|}$$

Using Gaussian assumption, i.e., m large

- Show Extend M/M/1 examples

Relative Error

- Example: Find the relative error for the loss probability given 30 replications of the M/M/1/S ρ=0.7.
- Data m=30 [show extend model]

0.066524	1.052449	1.233736	0.197161	0.023002	0.564409	1.943335	0.648561	0.761048	0.130256	1.079867	0.742315	0.724162	1.542741	0.206498
1.38573	4.341059	0.478882	0.207289	0.638751	1.066117	0.916407	1.04683	0.837411	0.859991	1.572843	0.067779	0.134131	0.274906	0.169894

- Sample mean = 0.830
- Sample standard deviation = 0.837
- Assuming Gaussian
  - RE= 0.36 (36%)

Statistical Considerations: Common Techniques for Dealing with the Lack of Independence

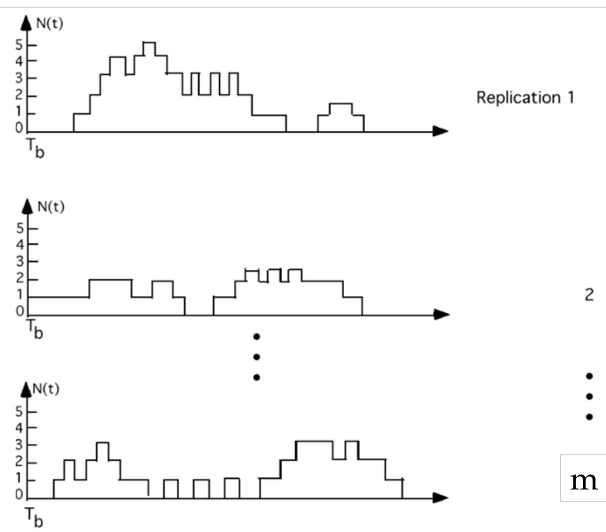
- Replication
- Batching
- Regenerative

* To overcome the initial transient the data from a start-up period is deleted (more later) Collect data for $t>T_b$ or for discrete observations for $n>n_0$

Replication

m Replications

Assume: Means from Replications are i.i.d.



Process: Replication

Conduct m replications of size $n+n_0$ each

1. Compute a mean for each replication:

$$\bar{x}_i = \frac{1}{n}\sum_{j=n_0+1}^{n_0+n} x_{ij} \quad i = 1, 2, \ldots, m$$

2. Compute an overall mean for all replications:

$$\bar{\bar{x}} = \frac{1}{m}\sum_{i=1}^{m} \bar{x}_i$$

3. Calculate the variance of replicate means:

$$\text{Var}(\bar{x}) = \frac{1}{m-1}\sum_{i=1}^{m} (\bar{x}_i - \bar{\bar{x}})^2$$

4. Confidence interval for the mean response is:

$$\left[\bar{\bar{x}} \mp z_\alpha \sqrt{\text{Var}(\bar{x})/m}\right]$$

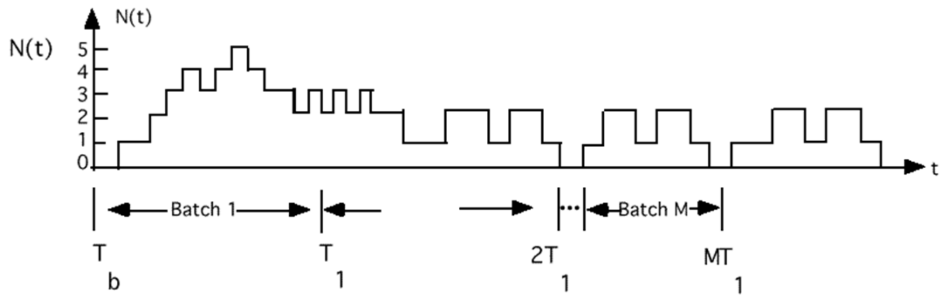
$$RE(\%) = \frac{100 z_\alpha \sqrt{\frac{Var(\bar{x})}{m}}}{\left|\bar{\bar{x}}\right|}$$

Modified rom: Raj Jain "The Art of Computer Systems Performance Analysis"
https://www.cse.wustl.edu/~jain/books/perf_sli.htm

Replication

- Assume results for each replication are independent
- Apply standard statistical techniques
- Inefficient because of M startup periods

Batching: Match Mean

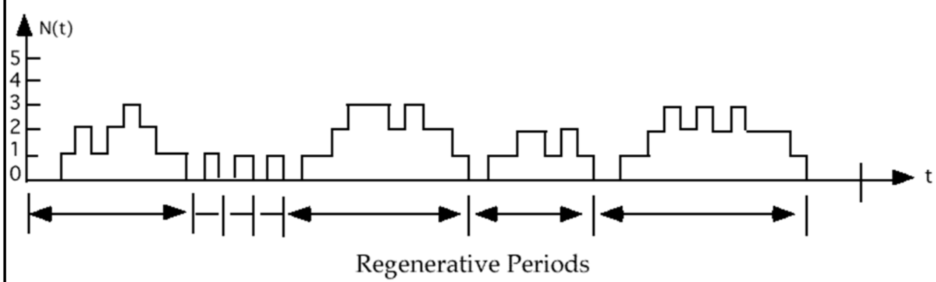


Run long simulation
Divide into equal duration segments
Treat each Batch as i.i.d.

Statistical Considerations: Batching

- Assume results from each batch are independent
- Apply standard statistical techniques
- Batches can be correlated unless “dead” periods between batches are employed
- Treatment of “dead” periods similar to dealing with initial transients

Statistical Considerations: Regenerative



Statistical Considerations: Regenerative

- Assume results from each regenerative period are independent
- Apply standard statistical techniques
- Regenerative period can become very long in some cases
- Note the regenerative technique overcomes the initial transient problem

Initial Transient Removal

- An initial transient period is present which can bias the performance measurements.
- Achieving Steady State
  - Run the simulation >> longer than the transient period. Then assume the transient has insignificant impact.
  - Use a run-in period:
    - Determine t_p such that the long-run distribution adequately describes the system for $t > t_p$
    - The system state at time t_p is a random variable, here we assume the probability of the system state at time t_p is sufficiently close to the steady state distribution.
    - Delete collected data before time t_p.
  - Use a "typical" starting condition (state) to initialize the model

Initial Transient Removal

1. Get a mean trajectory by averaging across replications

$$\bar{x}_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij} \quad j = 1, 2, \ldots, n$$

2. Get the overall mean:

$$\bar{\bar{x}} = \frac{1}{n}\sum_{j=1}^{n} \bar{x}_j$$

Set l=1 and proceed to the next step.

3. Delete the first *l* observations and get an overall mean from the remaining *n-l* values:

$$\bar{\bar{x}}_l = \frac{1}{n-l}\sum_{j=l+1}^{n} \bar{x}_j$$

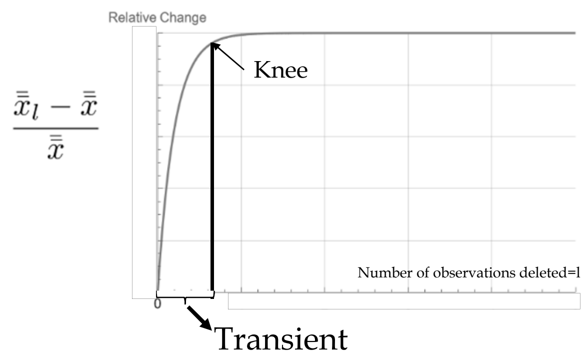
4. Compute the relative change:

$$\text{Relative change} = \frac{\bar{\bar{x}}_l - \bar{\bar{x}}}{\bar{\bar{x}}}$$

5. Repeat steps 3 and 4 by varying *l* from 1 to *n-1*.
6. Plot the overall mean and the relative change
7. *l* at knee = length of the transient interval.

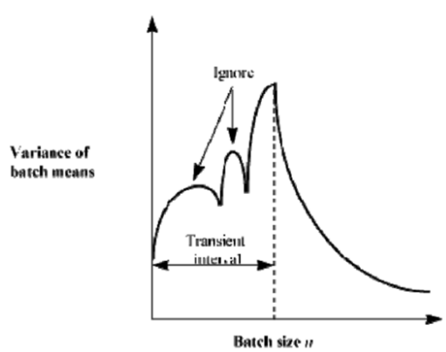
Modified from: Raj Jain "The Art of Computer Systems Performance Analysis"
https://www.cse.wustl.edu/~jain/books/perf_sli.htm

Initial Transient Removal



Batching: Transient Removal

- Finding batch size and transient time ?



Modified from: Raj Jain "The Art of Computer Systems Performance Analysis"
https://www.cse.wustl.edu/~jain/books/perf_sli.htm

Stopping Rules

- Replication
- Assume initial transient has been removed
- Stop simulation when RE<set threshold

$$RE = \frac{z_{\alpha}\sqrt{\frac{Var(\bar{x})}{m}}}{\left|\bar{\bar{x}}\right|} \leq \beta$$

Stopping Rules: Process

- Start simulation with m_1 replication.
- Check condition if satisfied then stop
- Else execute another replication and check again.
- Stop simulation when RE<β

$$RE = \frac{z_{\alpha}\sqrt{\frac{Var(\bar{x})}{m}}}{\left|\bar{\bar{x}}\right|} \leq \beta$$

Stopping Rules: Process

- Show Extend Example

Evolution of Computer-Aided Analysis and Design Tools for Networks

- "Zeroth" generation – general purpose languages
  - Fortran
  - C/C++
  - Pascal
  - Basic
- "First" generation – general purpose queueing system simulations
  - GPSS
  - SLAM
  - SIMSCRIPT

Evolution of Computer-Aided Analysis and Design Tools for Networks

- "Second" generation — application specific: computer systems and wide-area communication networks
  - RESQ
  - PAWS
- "Third" generation — integration of second generation languages with a graphics-oriented analysis and modeling environment
  - Extendsim (www.imaginethatinc.com)
  - GENESIS (from University of Massachusetts)
  - ns3
  - SES/Workbench (SES/Workbench is a trademark of Scientific and Engineering Software, Inc., Austin, TX)
  - OPNET (Mil 3, Inc., Washington, DC)

Relative Merits of General Purpose Languages

Advantages	**Disadvantages**
Wide Availability	Longer programming and debugging time
Few restrictions imposed on the model	Difficult verification
User may have prior knowledge of the language	Unless object-oriented, limited ability to reuse models
Generally more computationally efficient	Model enhancement and evolution are difficult

Relative Merits of Special Purpose Languages

Advantages	Disadvantages
Provide built-in simulation services to reduce programming effort	Must adhere to a particular “world view” of the language
Provide error-checking techniques superior to those provided in general purpose languages	Availability and support
Provide a brief, direct vehicle for expressing the concepts arising in a simulation study	Cost
Provide ability to construct user subroutines required as a part of any simulation routine	Increased computer running time
Contain set of subroutines for common random numbers	Training required to learn the language and modeling paradigm
Facilitate collection and display of data produced	
Facilitate model reuse	

Relative Merits of Computer-Aided Analysis and Design Environments

Advantages	Disadvantages
Provide a complete integrated performance analysis environment	Tailored to a specific modeling paradigm
Graphically based	May be tied to a specific hardware platform
Typically integrate language, database, prior knowledge, and statistical analysis packages	Increased execution time
Support management of models and input/output data	Cost
Facilitate model reuse and group model development	

Criteria for Selecting a Computer-Aided Analysis and Design Tool

- Availability
- Cost
- Usage
- Documentation
- Ease of Learning
- Computation Efficiency
- Flexibility
- Portability
- User Interface
- Extendibility
- Memory Requirements

Simulation Case Study:
Simulation of ATM WAN's

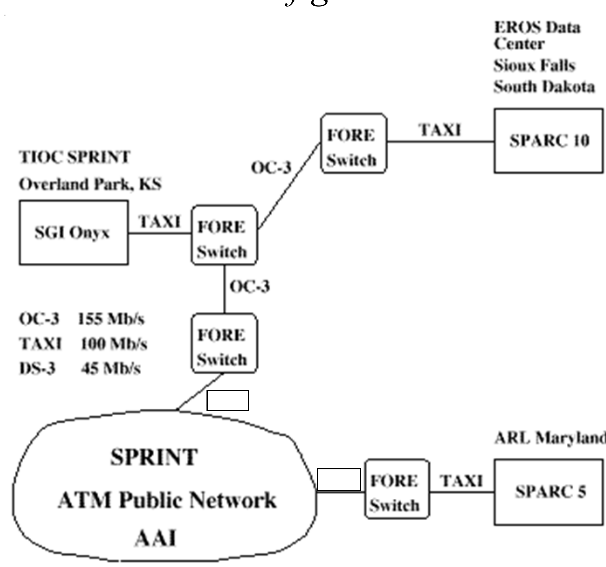
- Determine the level of model fidelity required to accurately predict ATM WAN performance
- Determine the feasibility of measurement based validation of ATM WAN simulation models
- Identify factors influencing ATM WAN performance

Simulation Case Study:

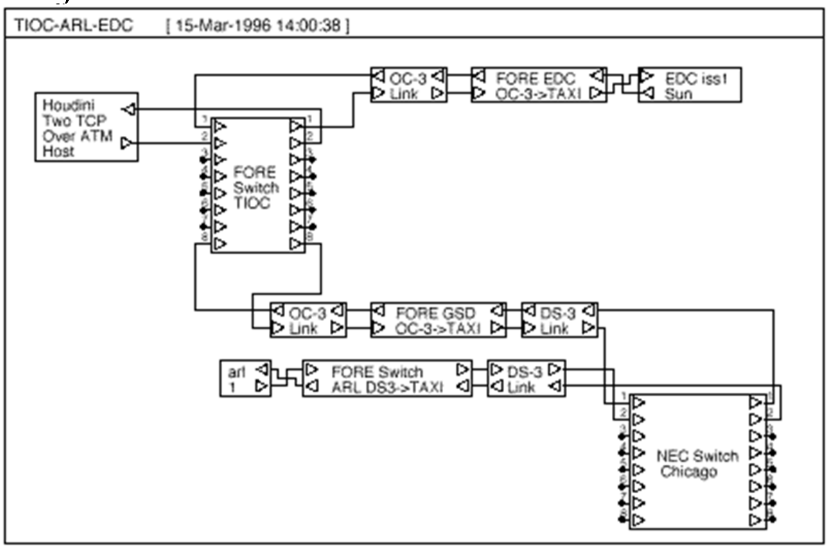
Simulation of ATM WAN's

System Parameter	Value
TCP MTU size	9180 bytes
TCP processing and OS overhead time	
- DEC 3000 AXP	200-300 μs
- SGI	550 μs
- SPARC 10	550 μs
- SPARC 5	700μs
TCP user send buffer size	64 kBytes
Slow-timer period	0.5 s
Fast-timer period	0.2 s
Minimum RTO	1.0 s
AAL5 SAR processing time	0.2 μs
AAL5 cell payload size	48 Bytes
Switch processing time	4 μs
Switch output buffer size per VC	256 cells
OC-3c link speed	155 Mb/s
TAXI link speed	100 Mb/s
DS-3 link speed	45 Mb/s

Simulation Case Study: *Simulation of ATM WAN's: Network Configuration*



Simulation Case Study: *Simulation of ATM WAN's: Simulation Model*



Comparison of Experimental and Simulation Performance Predictions

Connection	Experimental Results	Simulation Results
Baseline results: Point-to-point connections		
TIOC to ARL	4.2 Mb/s	7.18 Mb/s
TIOC to EDC	64.2 Mb/s	65.98 Mb/s
Simultaneous traffic streams: Single source, two destinations		
TIOC to ARL	4.45 Mb/s	4.60 Mb/s
TIOC to EDC	64.36 Mb/s	61.37 Mb/s
Simultaneous traffic streams: Two sources, single destination		
ARL to TIOC	2.15 Mb/s	4.87 Mb/s
EDC to TIOC	52.42 Mb/s	65.01 Mb/s
Simultaneous full duplex traffic streams		
TIOC to ARL	4.34 Mb/s	5.16 Mb/s
ARL to TIOC	4.3 Mb/s	5.16 Mb/s
TIOC to EDC	22.18 Mb/s	41.80 Mb/s
EDC to TIOC	31.18 Mb/s	41.30 Mb/s

Guidelines to Simulation, Modeling and Analysis of Communications Networks

- Know the customer
- Know the network
- Know the important performance metrics

Guidelines to Simulation, Modeling and Analysis of Communications Networks

- Know how to establish a credible model
- Expect the model to evolve
- Know how to apply good software management techniques

Conclusions

- Simulation can be an important tool for communication network design and analysis
- Care and thought must go into construction of communication network models
- Care and thought must go into interpretation of model output