AlaSim International 2014
6-7, May 2014 • Huntsville, Alabama • USA
Organized By AMSC
Alabama Modeling and Simulation Council

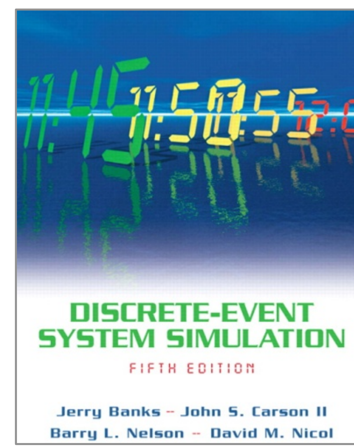
Defense-related Applications of Discrete Event Simulation

Mikel D. Petty, Ph.D.
University of Alabama in Huntsville

THE UNIVERSITY OF ALABAMA IN HUNTSVILLE

Outline

- Introduction and basic concepts
- Event-driven time advance
- Probability distributions
- Input modeling
- Random variate generation
- Example defense DES applications
  - UAV dispatching and loitering policies
  - Aircraft maintenance and availability
  - Additional examples
- Summary



Primary source
[Banks, 2010]

Introduction and basic concepts

Motivation and learning objectives

- Motivation
  - DES widely used in industrial, manufacturing, computing, and communications applications
  - Powerful, easy to use, and well understood
  - Less frequently used for defense applications
- Learning objectives
  - Basic concepts of DES
  - Suitable applications (general and defense) of DES
  - Introduction to key DES topics: event logic, probability distributions, data modeling, …
  - Exposure to example DES applications

There's more than one way to model a system.

Definitions

- Model: representation of something else
- Simulation: executing a model over time
- Simuland: system or phenomenon modeled

$$R = 2.59 \times \sqrt[4]{\sigma \times \left(\frac{\log^{-1}\left(\frac{ERP_t}{10}\right) \log^{-1}\left(\frac{G_r}{10}\right) \log^{-1}\left(\frac{MDS_r}{10}\right)}{\log^{-1}\left(\frac{FEL_r}{10}\right) F_t^2} \right)}$$

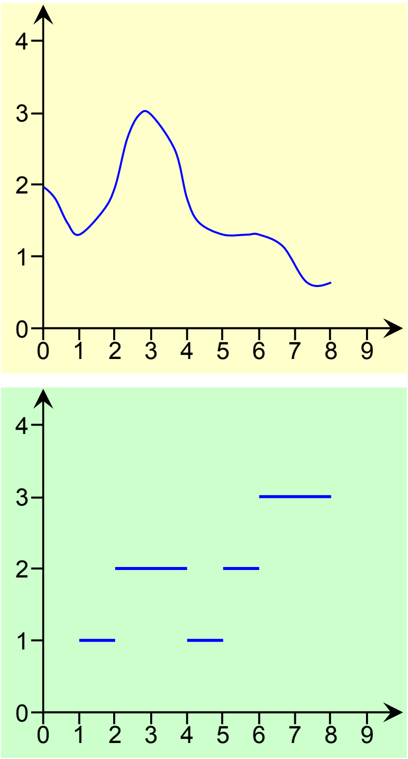
Model

Simulation

Both

What is discrete event simulation?

- DES is not:
  - Time-stepped
  - Continuous (or pseudo-continuous)
  - Physics-based
- DES is:
  - Event-driven
  - Discrete
  - Probability-based

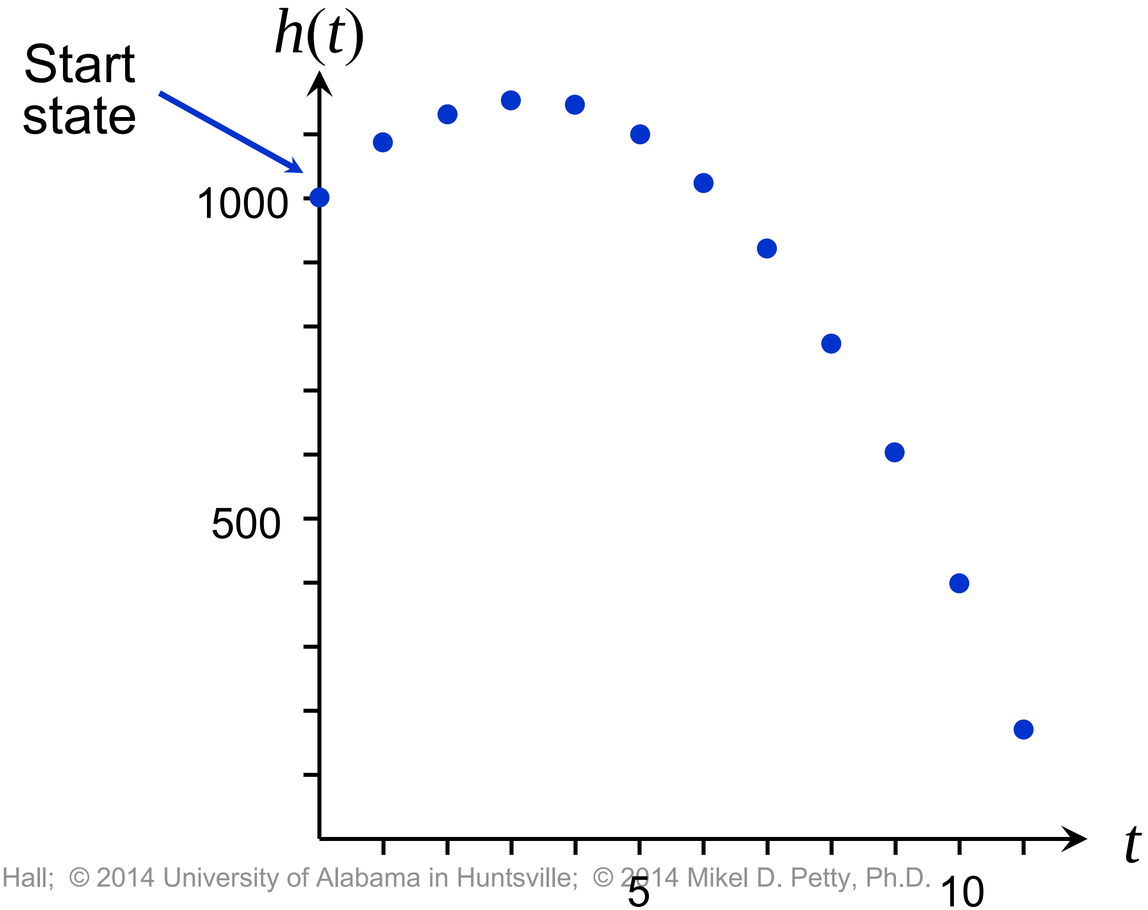


Non-DES simulation: Height under gravity

Time-stepped, continuous, physics-based

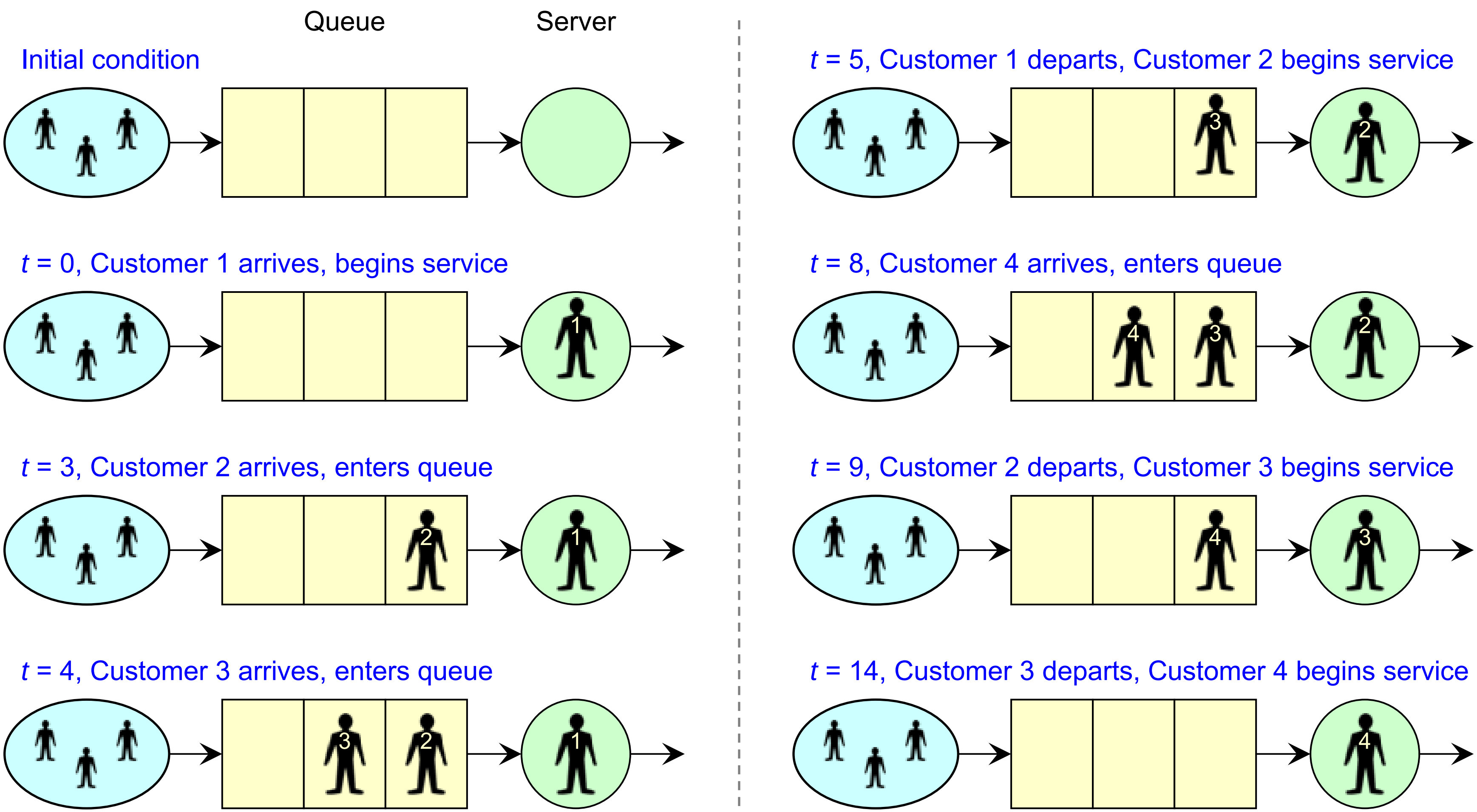
Model: $h(t) = -16t^2 + vt + s$ Data: $v = 100$, $s = 1000$

t	$h(t)$
0	1000
1	1084
2	1136
3	1156
4	1144
5	1100
6	1024
7	916
8	776
9	604
10	400
11	164



DES simulation: Customers in line (1 of 2)

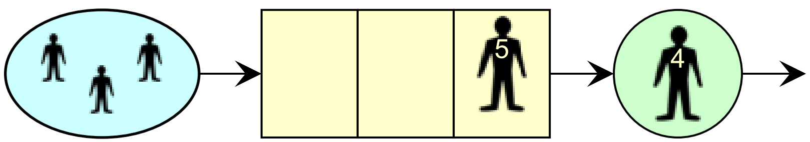
Event-driven, discrete, probability-based



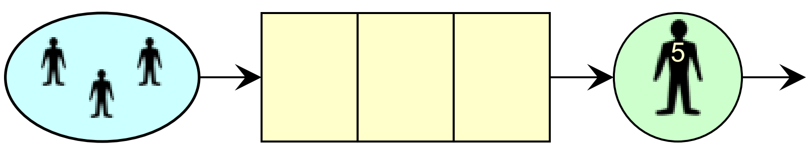
DES simulation: Customers in line (2 of 2)

Event-driven, discrete, probability-based

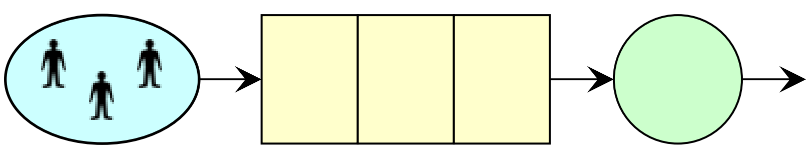
t = 16, Customer 5 arrives, enters queue



t = 17, Customer 4 departs, Customer 5 begins service



t = 22, Customer 5 departs, simulation ends



- How long did the queue get?
- What was the average queue length?
- What long did a customer wait for service, on average?
- How long did it take to service a customer, on average?

Analyzing DES simulation results

Maximum queue length = 2
Mean queue length = 0.636

Mean waiting time = 2.8
Mean service time = 4.4

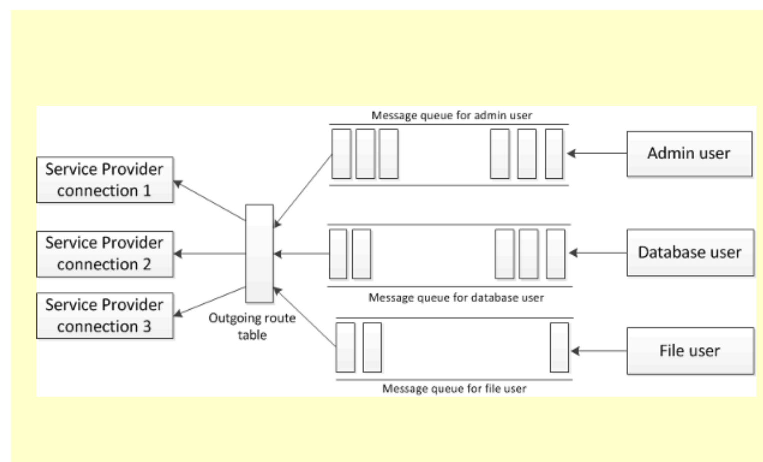
Time	Event	Queue length after event	Queue length * Time
0	1 arrives	0	0
3	2 arrives	1	1
4	3 arrives	2	2
5	1 departs	1	3
8	4 arrives	2	2
9	2 departs	1	5
14	3 departs	0	0
16	5 arrives	1	1
17	4 departs	0	0
22	5 departs	0	0
		Sum	14
		Mean	0.636

Customer	Arrive	Begin service	End service	Wait time	Service time
1	0	0	5	0	5
2	3	5	9	2	4
3	4	9	14	5	5
4	8	14	17	6	3
5	16	17	22	1	5
			Sum	14	22
			Mean	2.8	4.4

Basic concepts of DES

- Models built from abstract building blocks
  - Customers: entities requiring service or processing
  - Servers: entities providing service to customers
  - Queues: sets of customers waiting to be served
  - Events: changes in model (simuland) state
- Probability distributions model phenomena
  - e.g., time between customer arrivals
  - e.g., time required to serve customer
- Event-driven time advance
  - Model's state changes only at events
  - Time advances to time of next event, without modeling intervening time steps

Customers, queues, servers, and events



Scope of DES

- DES can model any simuland representable as a queuing system
- Queueing system
  - Characterized by waiting lines, or queues
  - State changes discretely at events

Simuland	Customers	Attributes	Servers	Events	Activities
Bank	Customers	Account balance	Teller ATM	Arrival Departure	Deposit Withdrawal
Subway	Riders	Origin Destination	Subway car	Arrival at station Arrival at destination	Travel
Assembly line	Assemblies	Speed Breakdown rate	Welding robot Installation worker	Breakdown	Weld Stamp
Comm network	Messages	Length Destination	Router Switch	Arrival at destination	Transmit
Field hospital	Wounded	Wound type Blood pressure	Surgeon Operating room	Arrival at hospital Begin treatment	Triage Treat

Questions to be answered about DES

- What is the logic for arrivals and departures?
- How are interarrival and service times determined during a simulation?
- How are probability distributions used to model physical phenomena and processes?
- How are the probability distributions developed?
- Is DES useful for defense-related applications?

Event-driven time advance algorithm

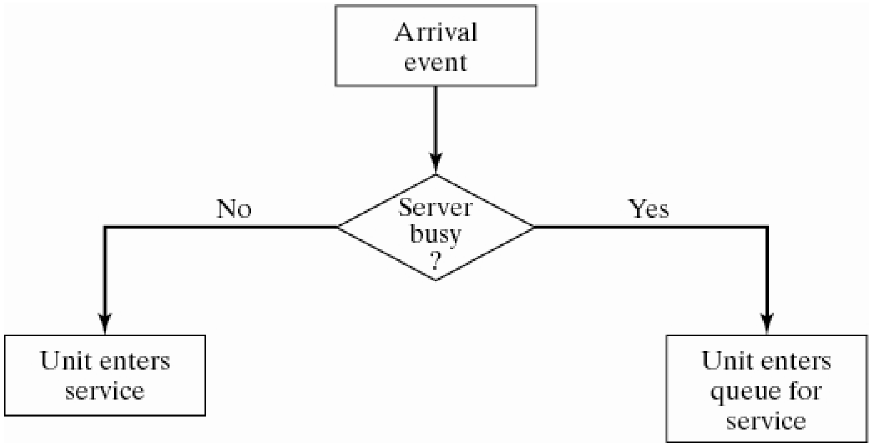
Future Event List

- Purpose
  - Organize advance of simulation time
  - Guarantee events occur in sequence
- FEL contents
  - Events scheduled at future times
  - Ordered chronologically, by scheduled event time
  - e.g., scheduled event times $t < t_1 \leq t_2 \leq t_3 \leq \ldots \leq t_n$
- Scheduling future events
  - Executing current event may schedule future event(s)
  - Future events added to FEL

Event-driven time advance algorithm

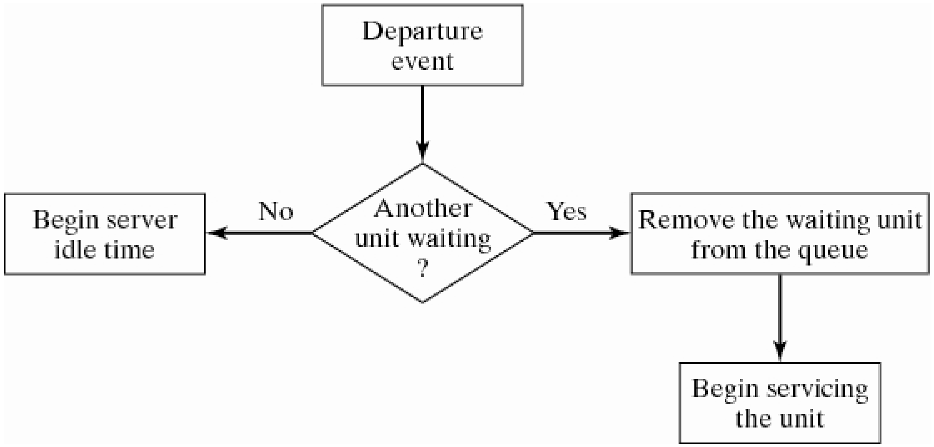
- Model status
  - Current $\mathrm{CLOCK} = t_0$
  - Imminent event (e_1, t_1) scheduled for t_1
- Algorithm
  - After processing for time t_0 complete …
  - Remove imminent event (e_1, t_1) from FEL
  - Advance (set) CLOCK to t_1
  - Process event e_1 per rules for event type:
    create new system state;
    possibly schedule future events by placing events on FEL
  - Repeat

Event logic: Arrival



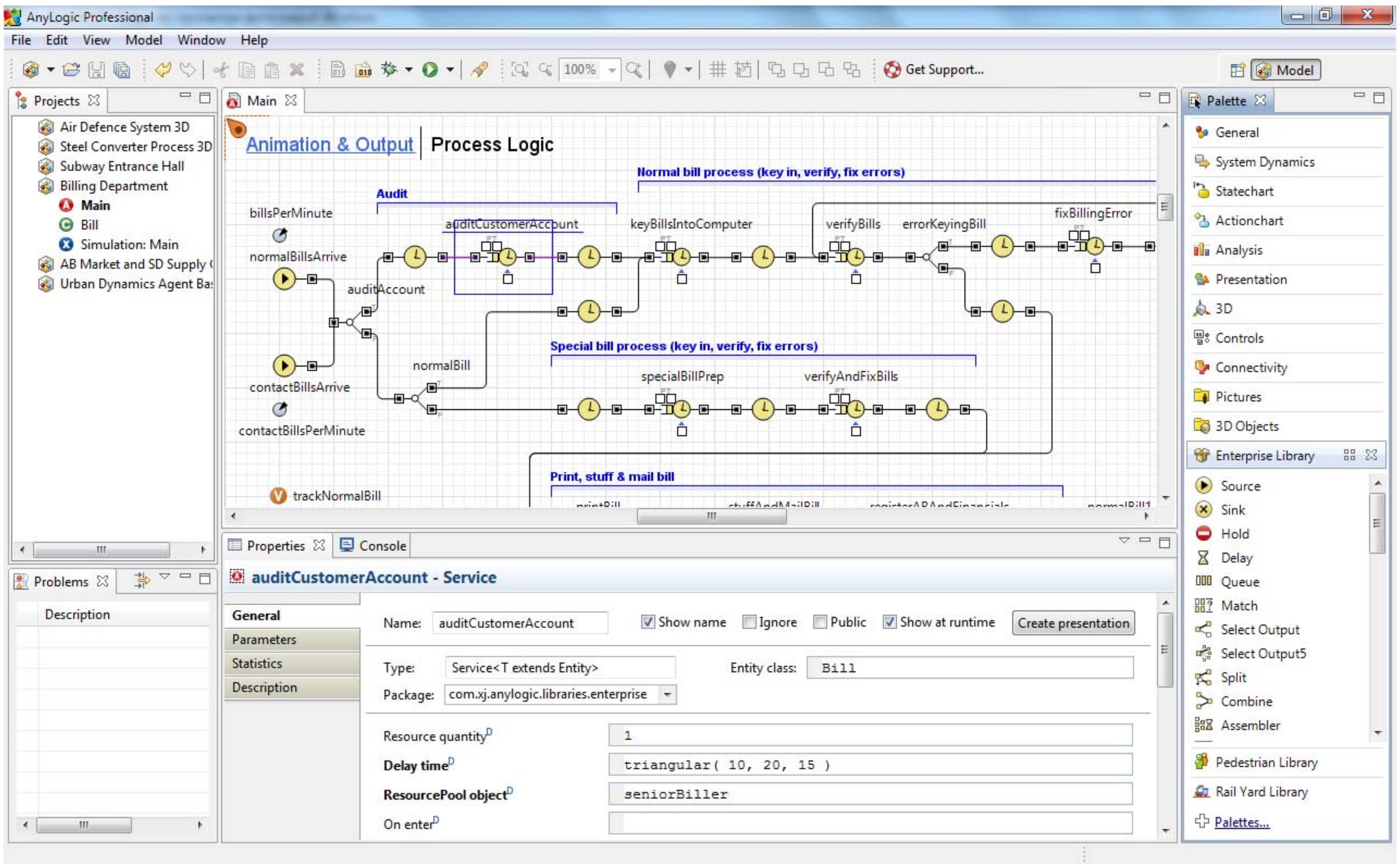
- Arriving customer may begin service immediately or enter queue
- Number of customers in system increases by 1
- Next arrival scheduled as part of processing current arrival

Event logic: Departure



- Customer departs when service complete
- Server becomes idle or begins service of next waiting customer
- Number of customers in system decreases by 1
- Next departure scheduled as part of processing current departure

Modeling multistep processes



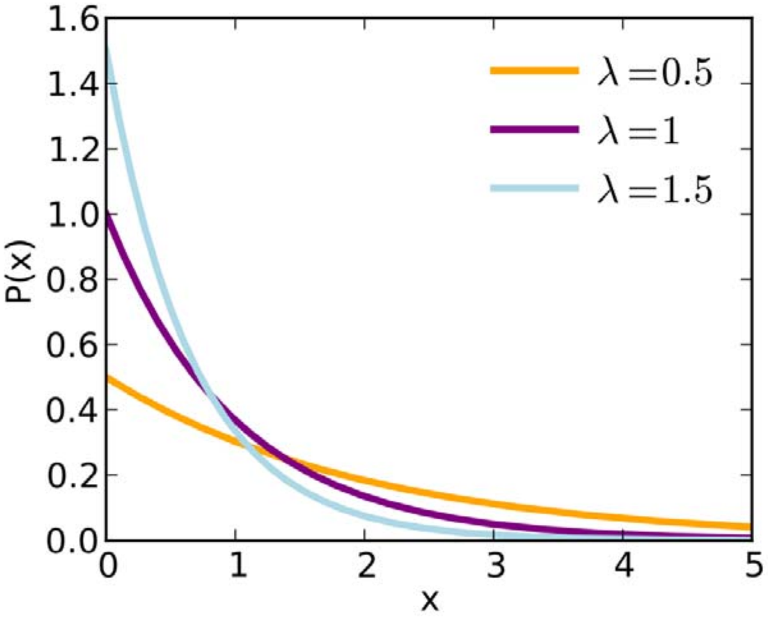
Probability distributions

Randomness and random variates

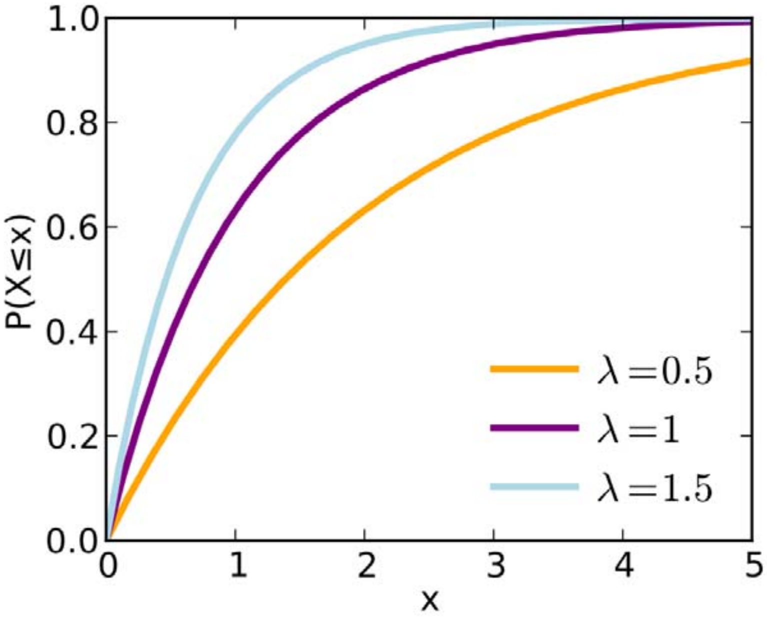
- Randomness in discrete event simulation
  - Randomness used extensively in DES
  - DES randomness imitates uncertainty in real life
  - Represents system aspects not otherwise modeled, individually unpredictable but follow a pattern
  - e.g., system events (interarrival times)
  - e.g., system activities (service times)
  - e.g., system inputs (inventory demand)
- Random variates
  - Random values for quantities of interest
  - Generated per probability distributions that model phenomenon or process

Exponential distribution

Probability density function



Cumulative distribution function



Exponential distribution

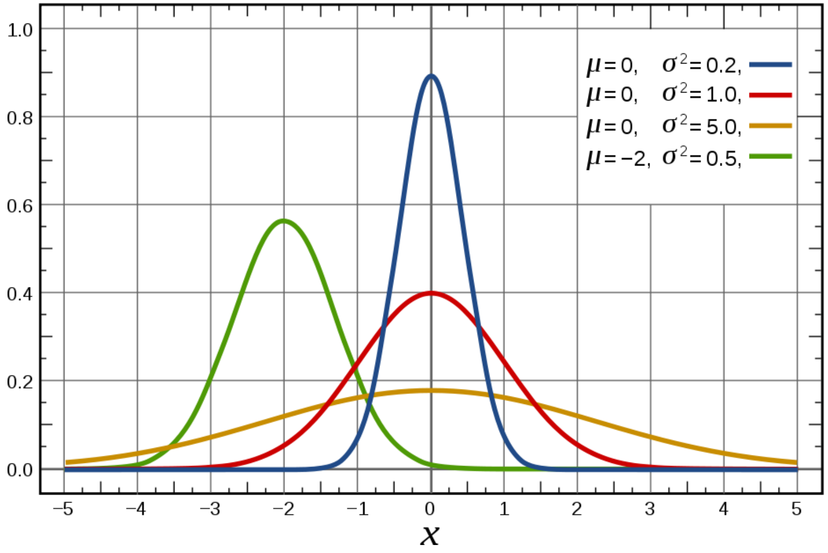
Larger values increasingly less probable.
Random variable X exponentially distributed, parameter λ.

$$\text{pdf} \quad f(x) = \begin{cases} \lambda e^{-\lambda x} & x \geq 0 \\ 0 & otherwise \end{cases}$$

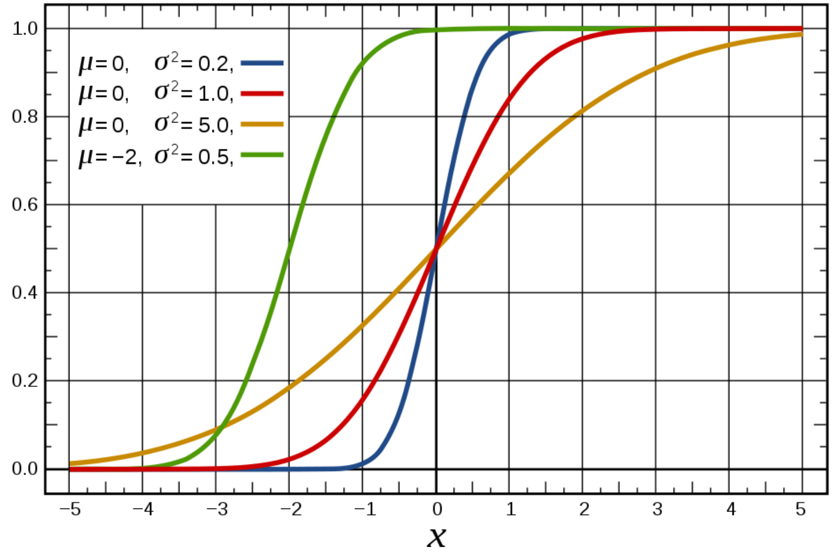
$$\text{cdf} \quad F(x) = \begin{cases} 0 & x < 0 \\ \int_0^x \lambda e^{-\lambda t} dt = 1 - e^{-\lambda x} & x \geq 0 \end{cases}$$

Normal distribution

Probability density function



Cumulative distribution function



Normal distribution

Values clustered around mean with variations.
Random variable X, mean $-\infty < \mu < +\infty$, variance σ^2.

$$\text{pdf} \quad f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2 \right] \quad -\infty < x < +\infty$$

$$\text{cdf} \quad F(x) = \int_{-\infty}^{x} \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{t-\mu}{\sigma}\right)^2 \right] dt$$

Distributions: General queueing systems

Interarrival time	
Exponential	Random, independent arrivals; mode 0
Gamma	Similar to exponential; parametric mode
Weibull	Similar to exponential; parametric mode; large values more likely
Service times	
Normal	Clustered around mean with variations; e.g., machining operation with material differences
Truncated normal	Normal but with minimum and/or maximum values
Exponential	Random, independent service durations; mode 0
Gamma	Similar to exponential; parametric mode

Distributions: Inventory and supply-chain

Demand	
Poisson	Simple, well known, extensively tabulated; large values less likely, given mean
Negative binomial	Large values more likely, given mean
Geometric	Special case of negative binomial
Time between demands	
Poisson	Simple, well known, extensively tabulated; large values less likely, given mean
Exponential	Random, independent time intervals; mode 0
Lead time	
Gamma	Similar to exponential; parametric mode

Distributions: Reliability and maintainability

Time to failure	
Exponential	Random, independent failures; mode 0
Gamma	Similar to exponential; parametric mode; useful for modeling standby redundancy (multiple components, each fails exponentially)
Weibull	Similar to exponential; parametric mode; useful for modeling failure due to most serious defect in multiple components
Normal	Clustered around mean with variations; e.g., failure due to wear
Lognormal	Specific component types

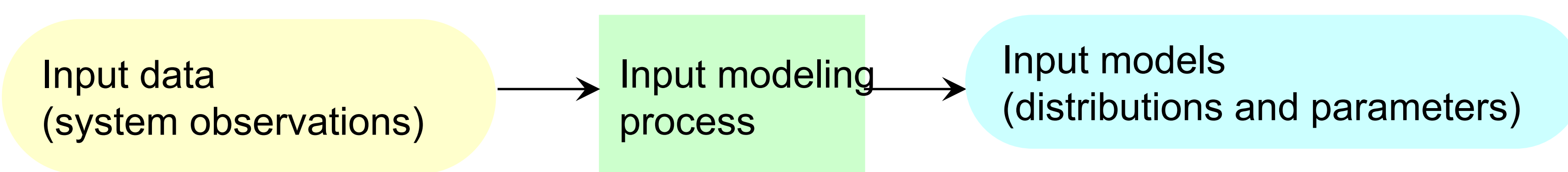
Distributions: All system types

Limited data available	
Triangular	Min, max, mode parameters estimated by SMEs
Uniform	Min, max parameters estimated by SMEs
Beta	Flexible distribution, highly parameterizable
All system types	
Empirical	Based on observation data, not theory; useful if data available, simuland not understood
Constant	Modeled phenomenon has consistent behavior; useful as means to simplify model

Input modeling

Input modeling

- Basic concept
  - Find suitable distribution and parameters ("model") to represent system component or phenomenon
  - AKA "input data modeling", "data modeling"
- Examples
  - Queueing system: interarrival times, service times
  - Supply-chain system: demand, lead time
  - Reliability analysis: time to failure



Input modeling procedure

1. Collect data from real-world system of interest
   - Record events of interest, e.g., queue arrival times
   - Manual or automatic
   - Can be difficult and/or time consuming
   - If data not available, expert opinion can be surrogate
2. Identify a probability distribution
   - Develop histogram of data, visually match distribution
   - Software tools available


Manual data collection

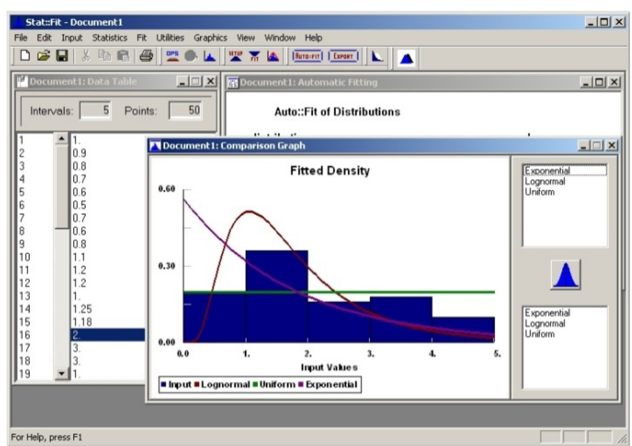

Automatic data collection

3 Choose parameters for the distribution

- Distributions defined by parameters, e.g., $N(\mu, \sigma^2)$
- Choose parameter values that best fit data
- Software tools available

4 Evaluate the selected distribution

- Perform goodness-of-fit tests to evaluate
- e.g., chi-square or Kolmogorov-Smirnov
- If fit not satisfactory, repeat from step 2



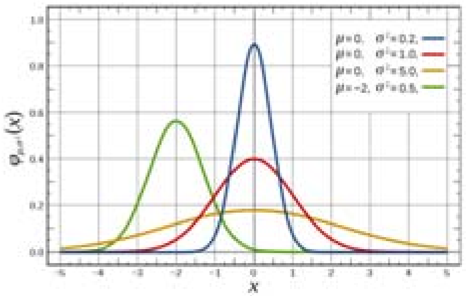
Stat::Fit screen shot

Data collection

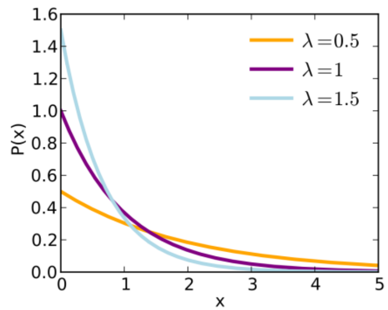
- Process
  - Collect data for system component or phenomenon
  - e.g., queue arrival times, machine service times
  - Manual; e.g., observers with watches, clipboards
  - Automatic; e.g., machine records starts and stops
- Comments
  - In class, often given as part of the exercise
  - In reality, can be difficult and/or time consuming
  - One of the most important parts of the project

Identifying the distribution

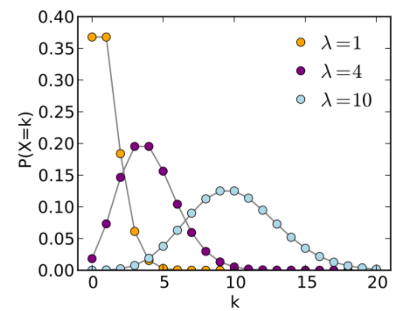
- Description
  - Given data, identify "family of distributions", e.g., normal, exponential, …
  - Later: determine "specific distribution", i.e., specific parameters of selected distribution
- Methods
  - Visual inspection of histogram
  - Consideration of "physical basis" of distribution
  - Construction of quantile-quantile plots



Normal



Exponential



Poisson

Example: Visual inspection of histogram

- Vehicles arriving at NW corner of intersection 7:00-7:05
- Counted for 100 days (5 workdays, 20 weeks)

Arrivals	Frequency
0	12
1	10
2	19
3	17
4	10
5	8
6	7
7	5
8	5
9	3
10	3
11	1

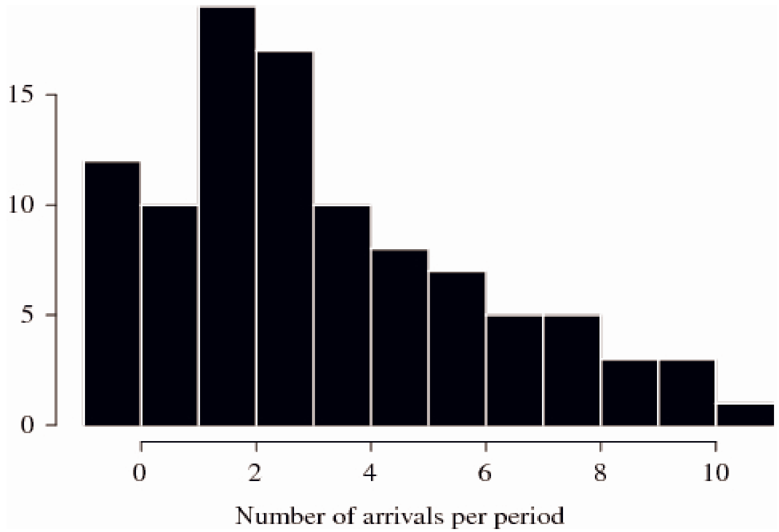


Table 9.1

Figure 9.4

Parameter estimation

- Process
  - Probability distributions have parameters that determine shape, scale, location
  - e.g., mean μ and standard deviation σ for normal
  - Once distribution selected (Step 2 of input modeling), parameter values must be estimated (Step 3)
- Comments
  - Formulas exist for estimated parameters for most simulation-related distributions
  - Formulas often use sample mean, sample variance
  - "Sample" is data collected
  - Sample mean, sample variance calculations vary

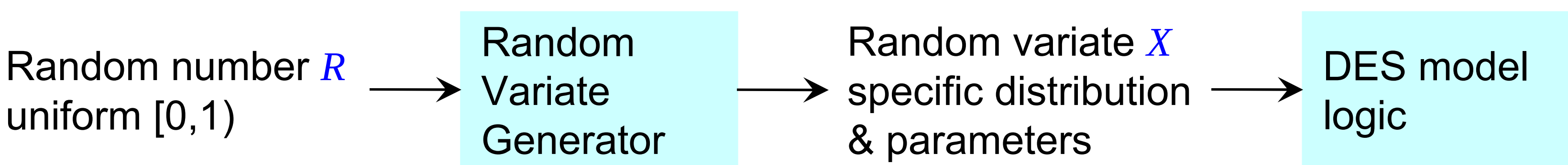
Goodness-of-fit tests

- Process
  - Once distribution selected (Step 2) and parameter values estimated (Step 3), suitability of input model evaluated (Step 4)
  - Evaluation done using hypothesis test
- Comments
  - Commonly used goodness-of-fit tests: chi-square, Kolmogorov-Smirnov
  - Can give false positive (small samples) and false negative (large samples)

Random variate generation

Random variate generation

- Basic concept
  - Input: Random number, uniformly distributed [0, 1)
  - Process: Convert input to output
  - Output: Random variate, specific distribution & parameters
- Method details depend on desired distribution
- Generation routines sometimes available



Inverse transform

- Description
  - Set cdf equal to R (random number)
  - Solve cdf for X (random variate)
- Comments
  - Applicable to continuous distributions: exponential, uniform, Weibull, triangular, empirical
  - Applicable to discrete distributions
  - Computationally and conceptually straightforward

Inverse transform general procedure

- Preparation
  1 Identify cdf: $F(x)$
  2 Set cdf $F(X) = R$ on range of X
  3 Solve equation $F(X) = R$ for X in terms of R; written as $X = F^{-1}(R)$
- Run-time
  4 Generate random variates $X_1, X_2, \ldots$ from random numbers $R_1, R_2, \ldots$ as $X_i = F^{-1}(R_i)$

Exponential distribution recap

Random variable X exponentially distributed, parameter λ.

$$\text{pdf} \quad f(x) = \begin{cases} \lambda e^{-\lambda x} & x \geq 0 \\ 0 & otherwise \end{cases}$$

$$\text{cdf} \quad F(x) = \begin{cases} 0 & x < 0 \\ \int_0^x \lambda e^{-\lambda t} dt = 1 - e^{-\lambda x} & x \geq 0 \end{cases}$$

λ = mean arrivals per time unit, i.e., “rate”
$1/\lambda$ = mean time between arrivals, i.e., “mean”

Exponential distribution inverse transform

1 Identify cdf: $F(x) = 1 - e^{-\lambda x}, x \geq 0$

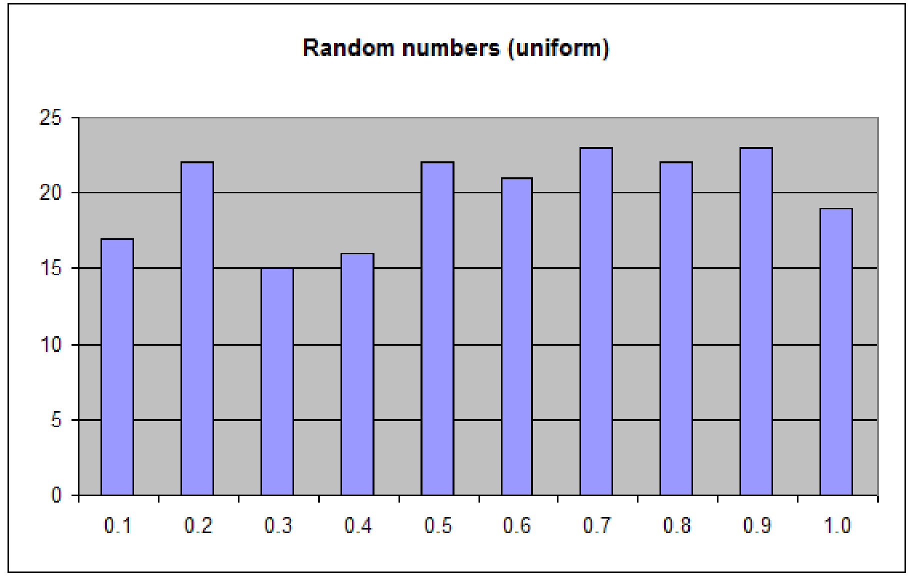
2 Set cdf $F(X) = R$ on range of X: $1 - e^{-\lambda X} = R$

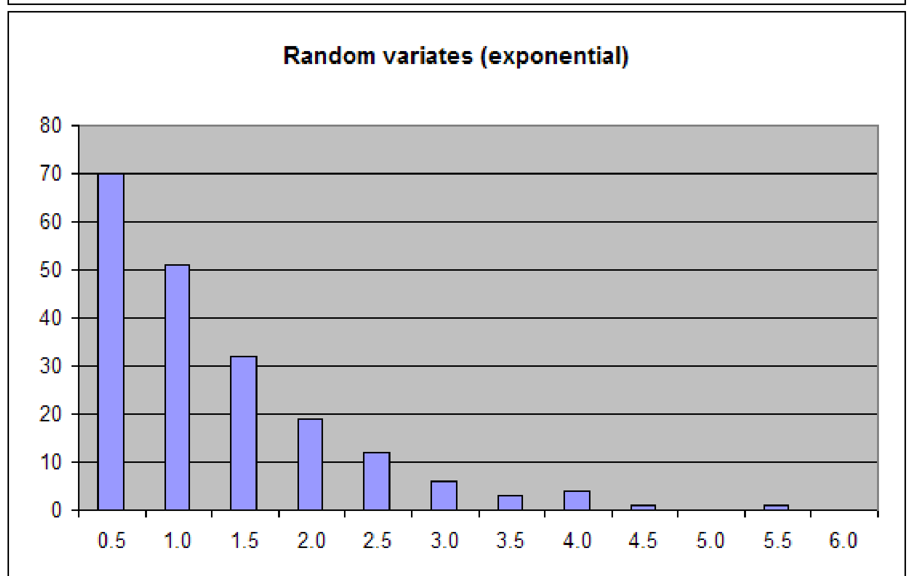
3 Solve equation $F(X) = R$ for X in terms of R:

$$1 - e^{-\lambda X} = R$$
$$e^{-\lambda X} = 1 - R$$
$$-\lambda X = \ln(1 - R)$$
$$X = -(1/\lambda) \ln(1 - R)$$

4 Generate random variates $X_1, X_2, \ldots$ from random numbers $R_1, R_2, \ldots$ as $X_i = -(1/\lambda) \ln(1 - R_i) = -(1/\lambda) \ln(R_i)$

i	R_i	X_i
1	0.7164	1.2601
2	0.4907	0.6747
3	0.8466	1.8745
4	0.9559	3.1202
5	0.6742	1.1216
6	0.8085	1.6527
7	0.1850	0.2046
8	0.3520	0.4338
9	0.0467	0.0478
10	0.4943	0.6818
11	0.0960	0.1009
12	0.6632	1.0884
13	0.5854	0.8804
14	0.2194	0.2477
15	0.5743	0.8540
16	0.2710	0.3161
17	0.0910	0.0954
18	0.2259	0.2561
19	0.9437	2.8769
20	0.0018	0.0018





Direct transformation: normal

Inverse transform not suitable, no inverse cdf.
Generate standard normal $N(0, 1)$ first,
then normal $N(\mu, \sigma^2)$ from that.

Standard normal

$$\text{pdf} \quad \phi(z) = \frac{1}{\sqrt{2\pi}} e^{-z^2/2} \qquad -\infty < z < +\infty$$

$$\text{cdf} \quad \Phi(x) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} e^{-t^2/2} dt$$

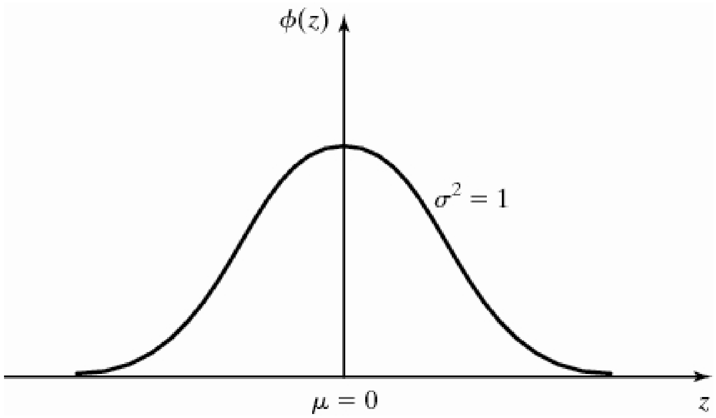


Figure 5.13

Standard normal variates Z_1, Z_2 as point in polar coords

$Z_1 = B \cos \theta$

$Z_2 = B \sin \theta$

Known that $B^2 = Z_1{}^2 + Z_2{}^2$
has chi-square distribution with 2 d.f.,
equivalent to exponential mean 2,
thus radius B can be generated as
$B = (-2 \ln R)^{1/2}$

Angle θ uniformly distributed $[0, 2\pi]$
B and θ independent

Thus Z_1 and Z_2 can be generated as

$Z_1 = (-2 \ln R_1)^{1/2} \cos (2\pi R_2)$

$Z_2 = (-2 \ln R_1)^{1/2} \sin (2\pi R_2)$

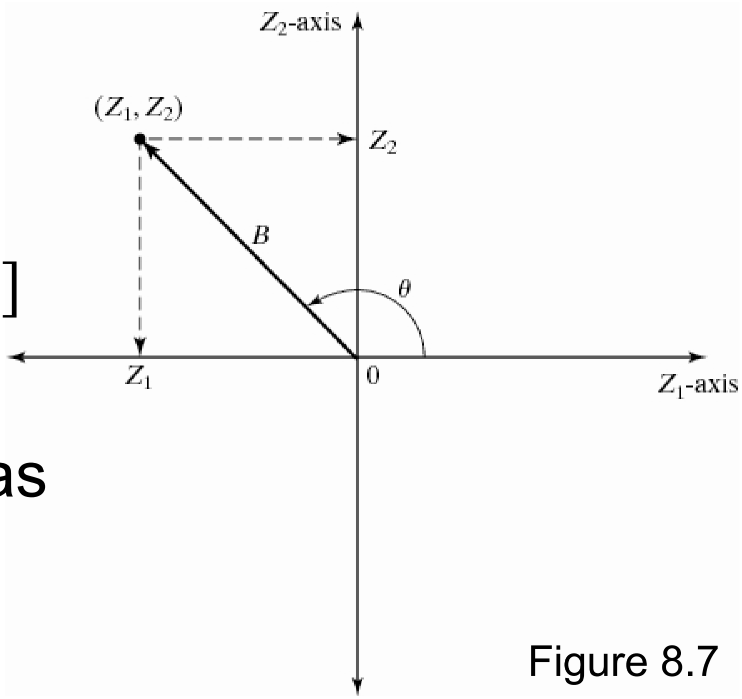
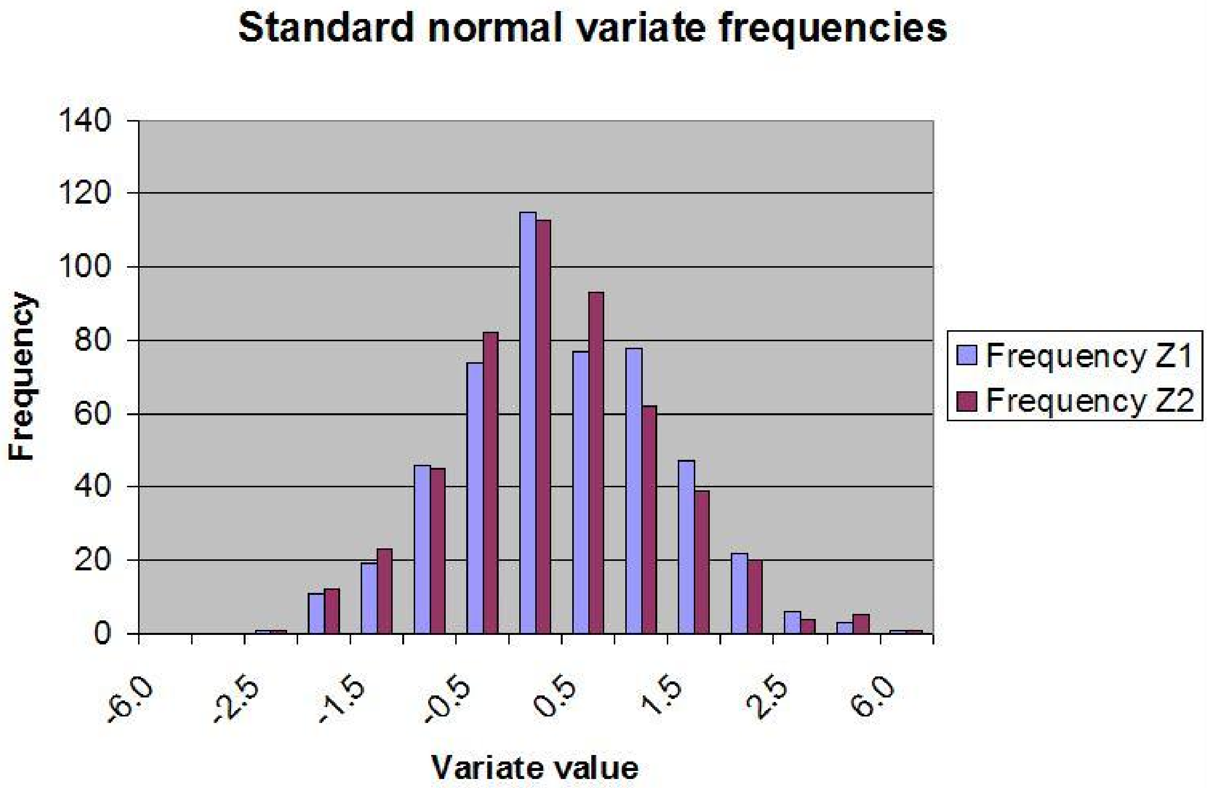


Figure 8.7

$$Z_1 = (-2 \ln R_1)^{1/2} \cos(2\pi R_2)$$
$$Z_2 = (-2 \ln R_1)^{1/2} \sin(2\pi R_2)$$

R_1	R_2	Z_1	Z_2
0.3857	0.9973	1.3801	0.0506
0.3767	0.5163	-1.3901	-0.7323
0.9814	0.8417	0.1056	0.3615
0.5443	0.9866	1.0991	0.0958
0.3646	0.5407	-1.3742	-0.7879
0.6553	0.8665	0.6145	-0.3528
0.4306	0.2442	0.0471	0.4899
0.5589	0.8844	0.8065	-0.4648
0.9413	0.3768	-0.2487	-1.3971
0.6150	0.0638	0.9078	-1.2846
0.0709	0.7878	0.5418	-0.1795
0.2052	0.8846	1.3322	0.4306
0.8755	0.3768	-0.3686	-1.0266
0.4161	0.9182	1.1532	0.3391
0.2337	0.0261	1.6821	-2.4579
0.7043	0.4734	-0.8256	1.0876
0.4120	0.2187	0.2605	1.7399
0.5392	0.9159	0.9598	-0.1046
0.2734	0.2227	0.2748	1.7122
0.1169	0.4743	-2.0448	-0.3395



To generate normal variates X_1, X_2 with mean μ variance σ^2

$X_i = \mu + \sigma Z_i$

For example, mean $\mu = 10$ variance $\sigma^2 = 4$

$X_1 = 10 + 2(1.3801) = 12.7602$

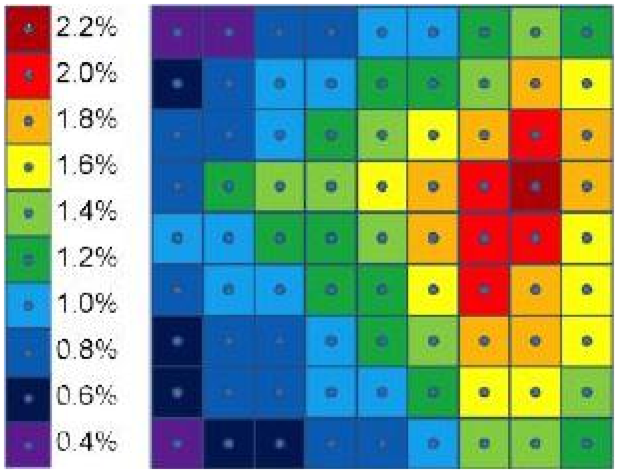
$X_2 = 10 + 2(0.0506) = 10.1012$

Example defense DES applications:
UAV dispatching and loitering policies

[Bednowitz, 2012]

Simuland

- Hostile targets detected intermittently at random locations in engagement area
- Group of UAVs available to engage targets
- When target appears, UAV selected to engage target
- After target destroyed, UAV loiters at selected location



Simulation study

- Question: Which UAV dispatching and loitering policies are most effective at engaging targets?
- Input variables: engagement area size,
  target arrival rate, target priority distribution,
  time required to engage target
- Output variables: weighted reward for engaging target
- Experimental design: 6 dispatching policies ·
  5 loitering policies · 4 input variables · 3 values for each
  = 360 combinations · 20 runs each = 7200 runs

Dispatching policies

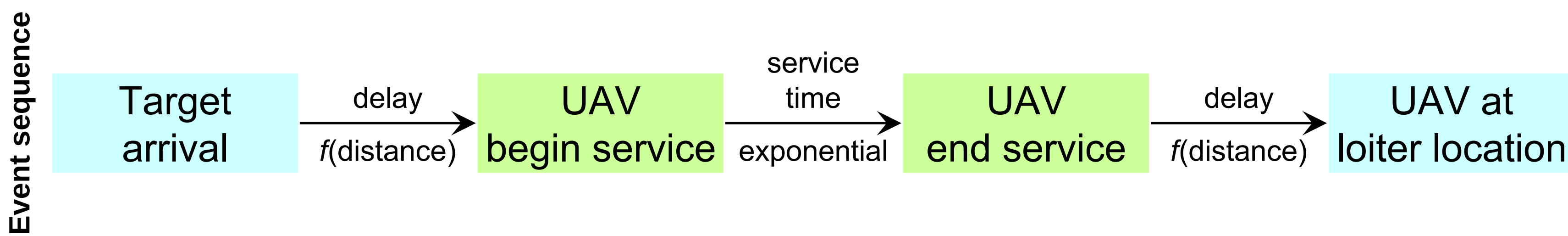
Policy	Target initiated	UAV initiated
DP1	First available	First come first served
DP2	Closest available	First come first served
DP3	Closest to be available	First come first served
DP4	First available	Shortest travel time or distance
DP5	Closest available	Shortest travel time or distance
DP6	Closest to be available	Shortest travel time or distance

Loitering policies

Policy	
LP1	Last location
DP2	Single location
DP3	p-Median
DP4	p-Median considering busy
DP5	Dynamic p-Median

Model components and implementation

- Customers: targets, exponential interarrival times
- Servers: UAVs, exponential service times, number 3
- Events
  - Target arrival
  - UAV begin service
  - UAV end service
  - UAV at loiter location
- Implementation: C++, custom code

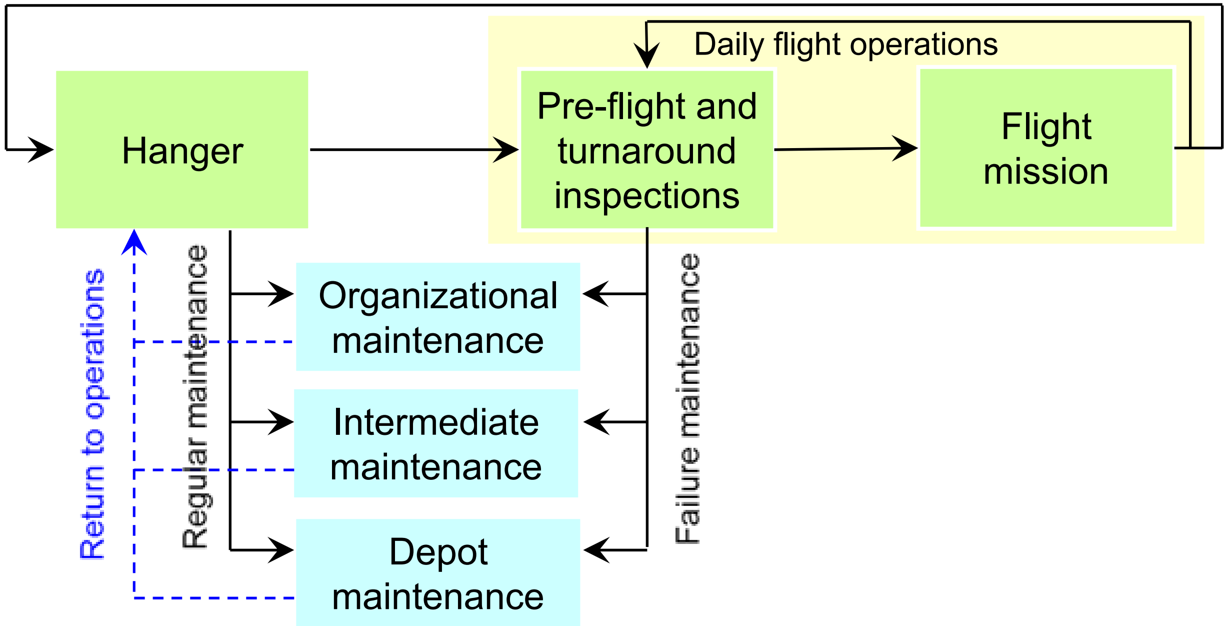


Example defense DES applications:
Aircraft maintenance and availability

[Raivio, 2001]

Simuland

- Military aircraft (BAE Hawk 51) maintenance operations
- Regular maintenance occurs at scheduled intervals
- Failure maintenance occurs after random failures
- Three levels of maintenance: Organizational (easiest), Intermediate, and Depot (hardest)



Simulation study

- Question: How can the aircraft flight and maintenance processes be optimized?
- Input variables: available maintenance manpower, maintenance duration
- Output variables: daily aircraft availability
- Experimental design: 20 manpower percentages · 5 maintenance duration percentages = 100 combinations · 30? runs each = 3000 runs

Maintenance manpower (percentage of nominal)	50%, 55%, 60%, …, 120%, 125%, 130%
Maintenance duration (percentage of nominal)	85%, 95%, 100%, 105%, 115%

Model components and implementation

- Customers: aircraft requiring maintenance
- Servers: maintenance personnel at each level
- Events
  - Aircraft needs regular maintenance
  - Aircraft needs failure maintenance
  - Begin aircraft maintenance
  - End aircraft maintenance
- Implementation: Arena, DES modeling package

Maintenance type	Occurs	Maintenance level	Maintenance time
Regular	Scheduled intervals (accumulated flight hours)	Organization	Normal
		Intermediate	Weibull
		Depot	Normal
Failure	Exponential interarrival (accumulated flight hours)	Organization	Gamma
		Intermediate	Gamma

Example defense DES applications:
Additional examples

Additional examples

- Combat casualty transport and treatment [Anderson, 2010]
- TML+, specialized DES environment

- Anti-torpedo defense countermeasures [Seo, 2011]
- DEVS, specialized DES language

- Anti-missile defense command and control [Kim, 2011]
- DEVS, specialized DES language

- F-15E availability during operational test [Pohl, 1991]
- SLAM, specialized Fortran-based DES language

Summary

Tutorial summary

- DES models queueing systems
  - Customers, servers, queues, and events
  - Many simulands of interest in this class
- DES consists of well-understood subtopics
  - Time advance and event logic
  - Probability distributions
  - Input modeling
  - Random variate generation
- DES packages available to simplify development
- DES useful for defense applications

References

[Anderson, 2010] C. Anderson, P. Konoske, J. Davis, and R. Mitchell, “Determining How eFunctional Characteristics of a Dedicated Casualty Evacuation Aircraft Affect Patient Movement and Outcomes”, *Journal of Defense Modeling and Simulation*, Vol. 7, No. 3, July 2010, pp. 167-177.

[Banks, 2010] J. Banks, J. S. Carson, B. L. Nelson, and D. M. Nicol, *Discrete-Event System Simulation, Fifth Edition*, Prentice Hall, Upper Saddle River NJ, 2010.

[Bednowitz, 2012] N. Bednowitz, R. Batta, and R. Nagi, “Dispatching and loitering policies for unmanned aerial vehicles under dynamically arriving multiple priority targets”, *Journal of Simulation*, Vol. 8, Iss. 1, February 2014, pp. 9–24, doi:10.1057/jos.2011.22.

[Brase, 2009] C. H. Brase and C. P. Brase, *Understandable Statistics: Concepts and Methods*, Houghton Mifflin, Boston MA, 2009.

[Kim, 2011] J. H. Kim, C. B. Choi, and T. G. Kim, “Battle Experiments of Naval Air Defense with Discrete Event System-based Mission-level Modeling and Simulation”, *Journal of Defense Modeling and Simulation*, Vol. 8, No. 3, July 2011, pp. 173-187.

[Pohl, 1991] L. M. Pohl, “Evaluation of F-15E availability during operational test”, *Proceedings of the 1991 Winter Simulation Conference*, Phoenix AZ, December 8-11 1991, pp. 549-554.

[Raivio, 2001] T. Raivio, E. Kuumola, V. A. Mattila, K. Virtanen, and R. P. Hämäläinen, “A Simulation model for Military Aircraft Maintenance and Availability”, *Proceedings of the 15th European Simulation Multiconference,* Prague, Czech Republic, June 6-9 2001, pp. 190-194.

[Seo, 2011] K. Seo, H. S. Song, S. J. Kwon, and T. G. Kim, “Measurement of Effectiveness for an Anti-torpedo Combat System Using a Discrete Event Systems Specification-based Underwater Warfare Simulator”, *Journal of Defense Modeling and Simulation*, Vol. 8, No. 3, July 2011, pp. 157-171.

End notes

- More information
  - Mikel D. Petty, Ph.D.
  - University of Alabama in Huntsville
  - Center for Modeling, Simulation, and Analysis
  - 256-824-4368, pettym@uah.edu
- Questions?