

Computer Simulation

Module 1: Intro + Course Tour

Dave Goldsman, Ph.D.

Professor

Stewart School of Industrial and Systems Engineering

Whirlwind Tour – History

Lesson Overview

Last Time: Talked about modeling and introduced simulation.

This Time: Let's talk history. Talk history. (History repeating itself.)

Simulation has been around a long time, but has really come into its own with the rise of the machines.



History

1777 – Buffon's Needle Problem – a new spin on things



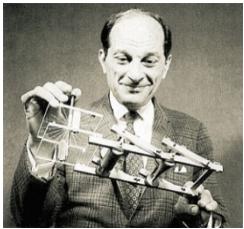
History

Early 1900's – Beer and Student's t distribution



More History

- 1946 – Ulam, Metropolis, von Neumann, and the H-Bomb



- 1960's – Industrial Applications
 - Manufacturing
 - Queueing Models



Recent History

- Development of Simulation Languages
 - Easy-to-use modeling tools
 - Graphics



← Harry Markowitz

- Rigorous Theoretical Work
 - Computational algorithms
 - Probabilistic and statistical methods



Origins: Manufacturing

Simulation is the technique of choice

- Calculates movement of parts and interaction of system components.
- Evaluates part flow thru the system.
- Examines conflicting demand for resources.
- Studies contemplated changes before their introduction.
- Prevents design blunders.



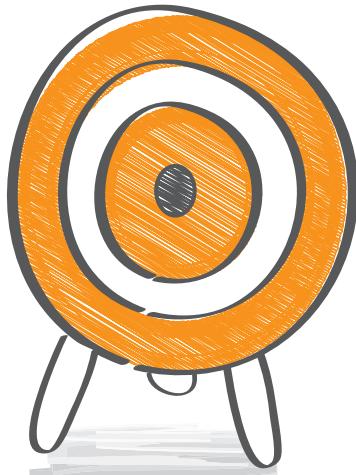
Typical Questions

- What will be the throughput?
- How can we change it?
- Where are the bottlenecks?
- Which is the best design?
- What is the reliability of the system?
- What is the impact of breakdowns?



Summary

- You can take your seat belts off...
the history lesson is over!
- Next Time: What kinds of stuff is
simulation used for?



Computer Simulation

Module 1: Intro + Course Tour

Dave Goldsman, Ph.D.

Professor

Stewart School of Industrial and Systems Engineering

What Can We Do For You?



Lesson Overview

Last Time: Discussed the history of simulation.

This Time: What kinds of stuff is simulation used for?

I'll give away the punch line:
Simulation can be used for an amazing array of different applications.



What I Used to Think Simulation Was

$$\begin{aligned}
& \sum_{i=n c_{k-1}+1}^{n c_k} \sum_{j=n c_k+1}^{n c_{k+1}} h_i h_j R_{j-i} \\
&= \left\{ \sum_{i=n c_{k-1}+1}^{n(c_k-\epsilon_k)} \sum_{j=n c_k+1}^{n c_{k+1}} + \sum_{i=n(c_k-\epsilon_k)+1}^{n c_k} \sum_{j=n c_k+1}^{n(c_k+\epsilon_k)} + \sum_{i=n(c_k-\epsilon_k)+1}^{n c_k} \sum_{j=n(c_k+\epsilon_k)+1}^{n c_{k+1}} \right\} h_i h_j R_{j-i} \\
&= \sum_{i=n c_k-n \epsilon_k+1}^{n c_k} \sum_{j=n c_k+1}^{n c_k+n \epsilon_k} h_i h_j R_{j-i} + O(n^4 \delta^{n \epsilon_k})
\end{aligned}$$

By complete enumeration as done in Appendix A.2, we get

$$\sum_{i=n c_k-n \epsilon_k+1}^{n c_k} \sum_{j=n c_k+1}^{n c_k+n \epsilon_k} h_i h_j R_{j-i} = \sum_{i=1}^{n \epsilon_k} \sum_{j=1}^i h_{n c_k-j+1} h_{n c_k-j+i+1} R_i + O(n^4 \delta^{n \epsilon_k})$$

Therefore,

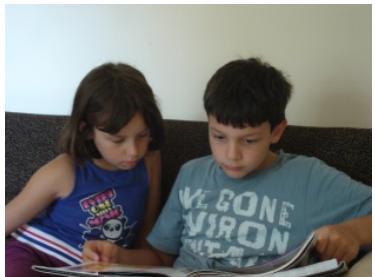
$$E[\mathcal{A}_{C_m}^2(f; n)] = E[\mathcal{A}^2(f; n)] - \frac{4}{n^3} \sum_{k=1}^m \sum_{i=1}^{n \epsilon_k} \sum_{j=1}^i h_{n c_k-j+1} h_{n c_k-j+i+1} R_i + O(n \delta^{n \epsilon}) \quad (9)$$

Actual Applications

- Manufacturing
 - Automobile Production Facility
 - Carpet Production Facility
- Queueing Problems
 - Call Center Analysis
 - Fast Food Drive-Thru
 - Fast Food Drive-Thru Call Center
 - Airport Security Line

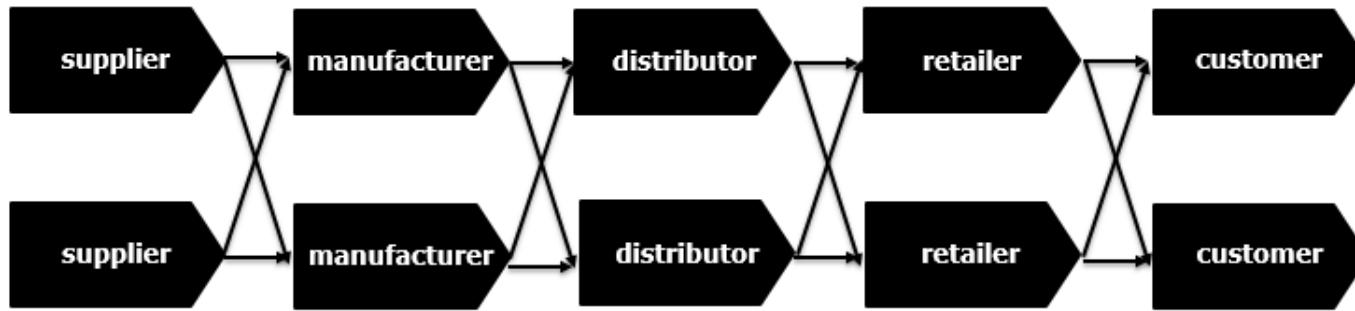
Some Queueing Systems

- Ice Cream Shoppe
- Combination meat processing center / fast-food restaurant / amusement park



← Denis (8) and Mina (6)

Generic Supply Chain



- Think of supply chain as a network where any node can potentially connect to any other node.
- In order to be lazy, let's have things flow from left to right. Looks a bit like a flowchart, eh?

Supply Chains (cont'd)

- 1985: Professor and CEO of SC company: “Why would anyone want to simulate a SC?”
- Now: Many SC tools with simulation capability.
 - Use simulation to determine how much value-added a forecasting application provides.
 - Use simulation to analyze how SC randomness or model errors affect a proposed solution. Is it robust? What's the best solution?



More Applications

- Inventory and Supply Chain Analysis
- Financial Analysis
 - Portfolio Analysis
 - Options Pricing
- Traffic Simulation
- Airspace Simulation
- Service Sector
- **Health Systems**

Health Systems

- Patient Flow in a Hospital
- Hospital Room Allocation
- Optimization of Doctor / Nurse Scheduling
- Procurement of Supplies
- **Disease Surveillance**
- Propagation of Disease Spread
- Humanitarian Logistics

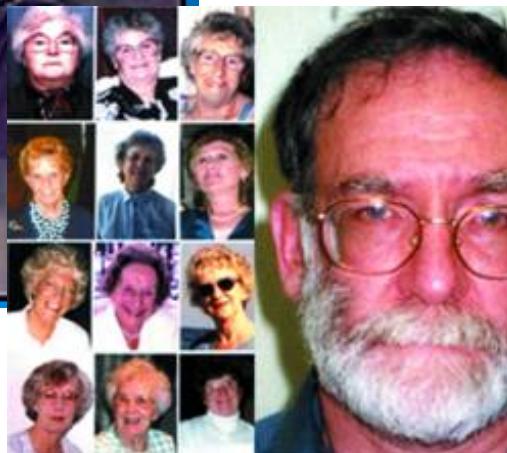
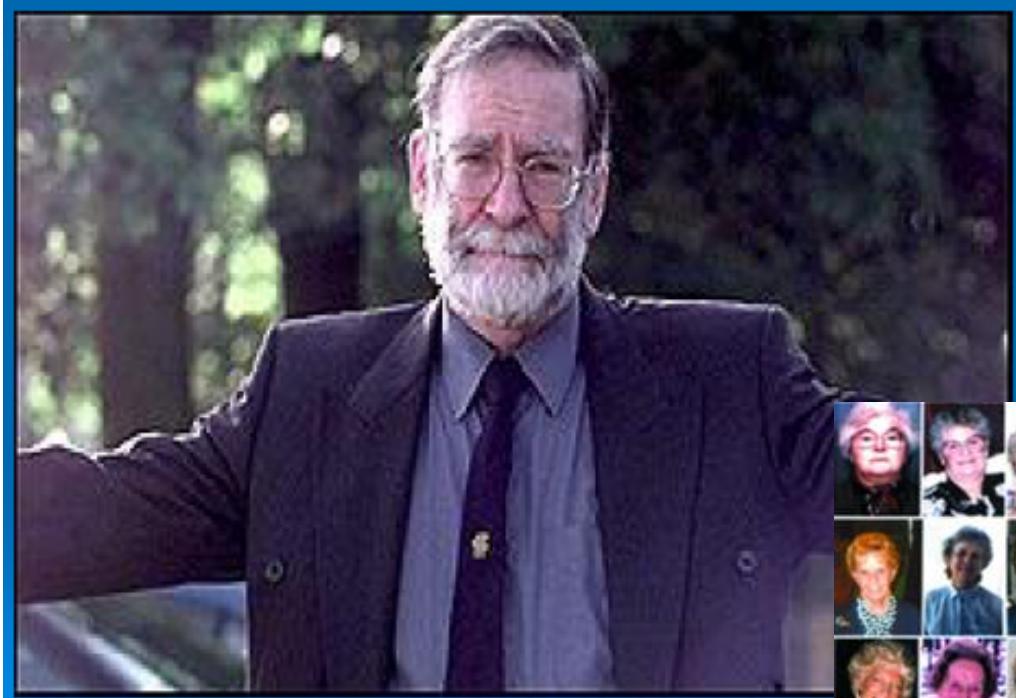


Surveillance Application

- Use simulation to monitor certain time series.
- Predict issues as or before they happen.
- Is a disease in the process of becoming an outbreak?
- When is something out of the ordinary occurring?
- Take advantage of **HUGE** data sets.



Who's Mr. Handsome?



Dr. Harold Shipman

- Killed his patients using heroin overdoses (#2 UK serial killer).
- Caught after carelessly revising a patient's will, leaving all her assets to himself (don't use your own typewriter next time, dummy).
- Doctored his records to show his patients had needed morphine. But the software recorded the dates of these modifications.
- Hung himself in prison, never confessed.



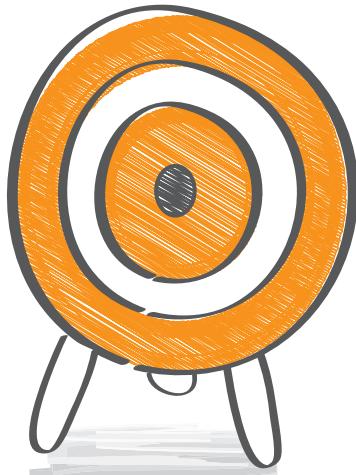
Simulation Can Help!

- Surveillance uses sequential hypothesis tests, where the null hyp H_0 is “no disease” (“no murder”).
- The test statistics have very difficult distributions, even under H_0 .
- Use simulation to approximate the probability distributions of the statistics under the null hyp.
- If sampling casts doubt on this distribution, then reject.



Summary

- Looked at a wide variety of practical simulation applications.
- Next Time: We'll give some very easy examples of the actual use of simulation in simple settings.



Computer Simulation

Module 1: Intro + Course Tour

Dave Goldsman, Ph.D.

Professor

Stewart School of Industrial and Systems Engineering

Some Baby Examples

Lesson Overview

Last Time: Gave high-level examples of the kinds of stuff simulation is used for.

This Time: Some very easy examples of the use of simulation in simple settings.

The idea here is to start us off on our simulation journey via some really simple problems.



Happy Birthday

How many people do you need in a room in order to have a 50% chance that at least two will have the same birthday?

- 9
- 23
- 42
- 183



Birthday Paradox

1. Enter Seed:

2. Click Start Button:

START

New Birthday Date:

Total # Birthdays:

!!! MATCH !!!

BACK

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1		happy				happy			happy			
2												
3		happy			happy							
4										happy		
5											happy	
6								happy				
7		happy										
8							happy					
9												
10							happy					
11								happy				
12												
13					happy							
14												
15		happy					happy					
16						happy						
17				happy								
18												
19							happy					
20												
21												
22							happy					
23		happy							happy			
24				happy								
25												
26									happy			
27												
28												
29		XXXX						happy				
30	happy	XXXX										
31	XXXX		XXXX		XXXX				XXXX		XXXX	

Let's Make Some Pi

- Use simulation to estimate π .
- Idea:
 - Area of a unit square is 1.
 - Area of an inscribed circle is $\pi/4$.
 - Probability that a dart thrown at the square will hit the circle is $\pi/4$.
 - Throw lots of darts. Proportion that will land in circle should approach $\pi/4$.

Bonus Quiz: What is the volume of a pizza with radius z and height a ?



Monte Carlo Simulation

Number of Points ?

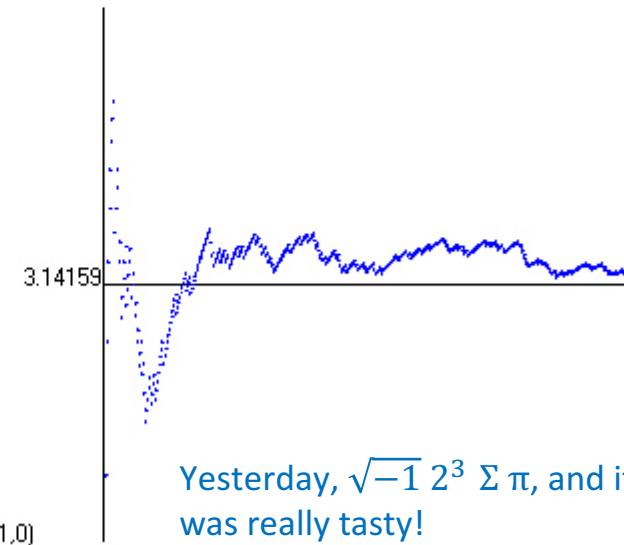
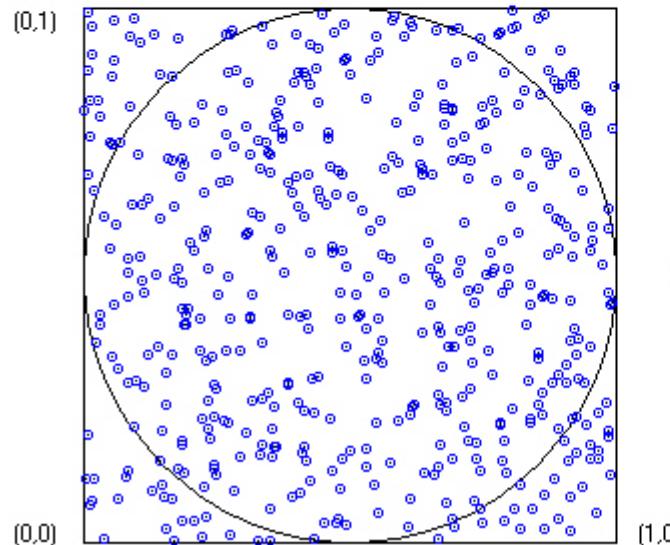
Points Plotted:

Random Seed ?

Real Value of Pi:

Animation Delay (1-100) ?

Estimator for Pi:

START**BACK**

Yesterday, $\sqrt{-1} 2^3 \Sigma \pi$, and it was really tasty!

Fun With Calculus

- Use simulation to integrate
 $f(x) = \sin(\pi x)$ over $[0,1]$.
- Idea:
 - Sample n rectangles.
 - Each is centered randomly on $[0,1]$ and has width $1/n$ and height $f(x)$.
 - Add up areas.
 - Make n really, really big.
 - Sum of areas approaches the integral of $f(x)$.

Monte Carlo Integration

1. Number of Points:

2. Enter Seed:

3. Animation Delay:

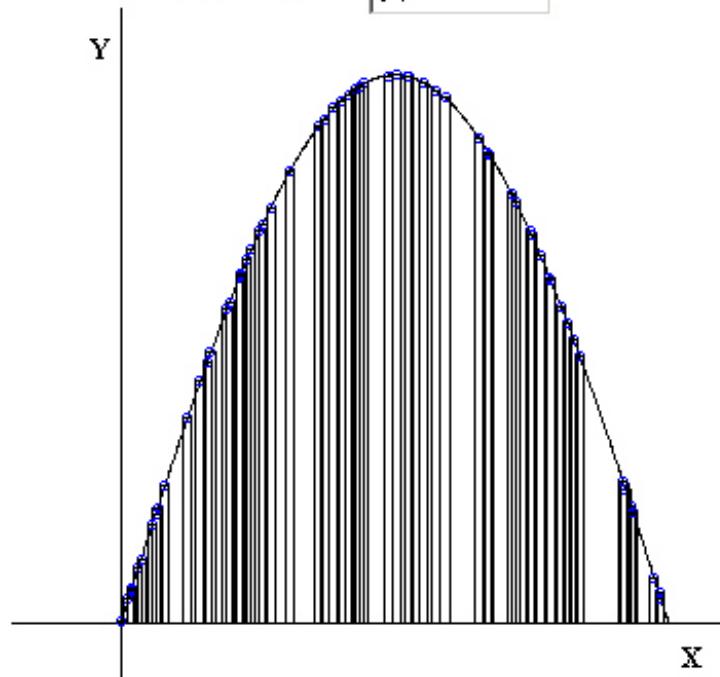
**ESTIMATE
INTEGRAL**

Actual Result:

Estimator:

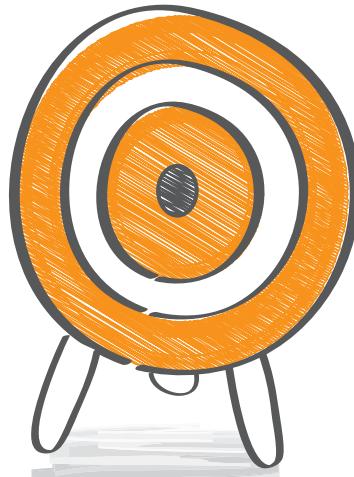
BACK

Points Plotted:



Summary

- Looked at some easy examples of the use of simulation applications in simple settings.
- Next Time: More more more! How do you like it? How do you like it?



Computer Architecture

Module 1: Introduction + Whirlwind Tour

Dave Goldsman, Ph.D.

Professor

Stewart School of Industrial and Systems Engineering

More Baby Examples

Lesson Overview

Last Time: Some baby examples.

This Time: More baby examples.

Evil Random Numbers

- See what happens when you use a bad random number generator.
- Idea:
 - Simulate heights vs weights.
 - Should be a 2-D bell curve (normal distribution) with most observations in the middle and some on the outside.
 - Do observations “look” random?



Box Muller Method

1. Number of Points:
(Max 5000 points)

Points Plotted:

2. Enter Seed:
(Any Positive Integer)

3. Animation Delay:
(Value between 1 and 100)
(Default is 1)

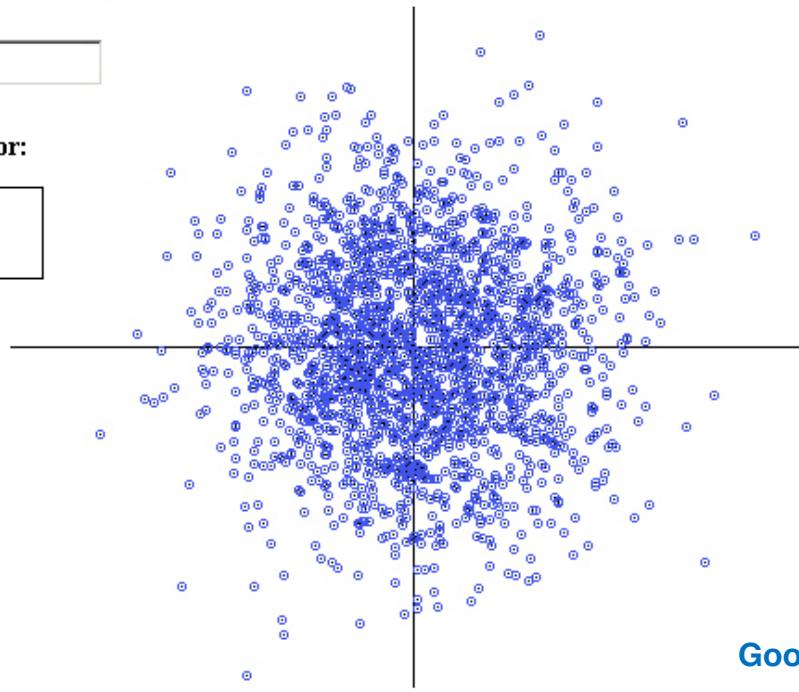
4. Select Type of Generator:

Random Number Generator
 Good Bad

5. Click Start:

START

BACK



Good Dawg!

Box Muller Method

1. Number of Points:
(Max 5000 points)

Points Plotted:

2. Enter Seed:
(Any Positive Integer)

3. Animation Delay:
(Value between 1 and 100)
(Default is 1)

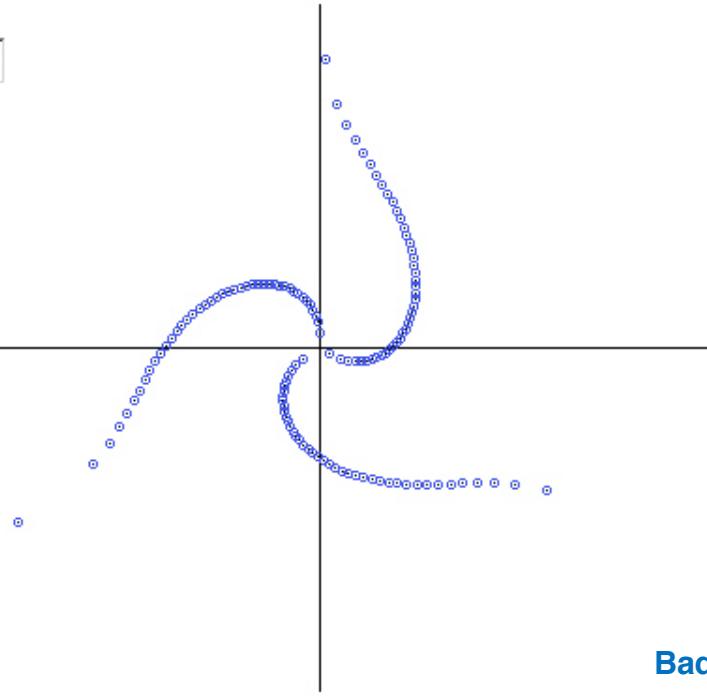
4. Select Type of Generator:

Random Number Generator
 Good Bad

5. Click Start:

START

BACK



Bad Dawg!

Queues 'R Us

- Single-server queue at McWendy's.
- Customers show up, wait in line, get served first-in-first-out.
- What happens as arrival rate approaches service rate?
 - Nothing much?
 - Line gets pretty long?
 - Hamburgers start to taste better?



Queues ‘R Us (cont’d)

- Can analyze queues via simulation.
- Can analyze via numerical or exact methods.
- Fun fact: Notice anything interesting about the word “queueing”? How about “queueoid”?

MM1 Queue Simulation

Interarrival Mean?

4

STOP

Service Mean?

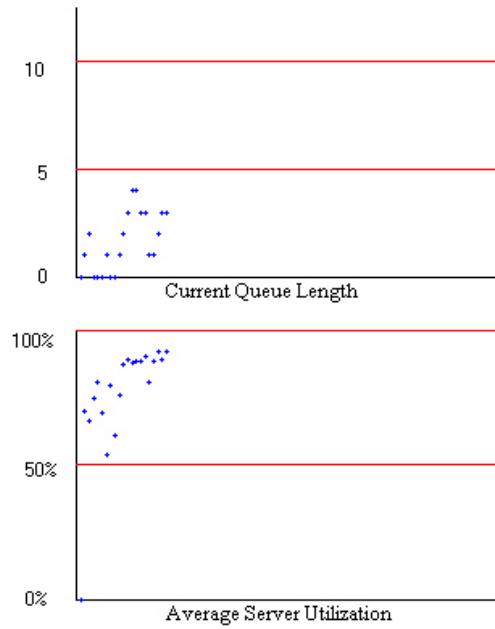
3

NEXT CUSTOMER

i	Arrive	Start	Service	Leave	Wait
1	0	0	7	7	0
2	2	7	3	10	5
3	5	10	5	15	5
4	18	18	2	20	0
5	20	20	1	21	0
6	24	24	2	26	0
7	25	26	11	37	1
8	37	37	2	39	0
9	39	39	15	54	0
10	43	54	9	63	11
11	47	63	2	65	16
12	50	65	1	66	15
13	51	66	2	68	15
14	61	68	2	70	7
15	65	70	2	72	5
16	66	72	1	73	6
17	72	73	10	83	1
18	74	83	4	87	9
19	76	87	1	88	11
20	77	88	4	92	11
21	83	92	1	93	9

Interarrival time

6



Output Analysis

DISPLAY DATA**BACK**

SIMTIME:

093

THROUGHPUT:

021

Average # of Customers in the System:

2.30

Average Time in the System:

10.19

Server Utilization:

93 %

Average Length of the Queue:

1.37

Average Waiting Time in the Queue:

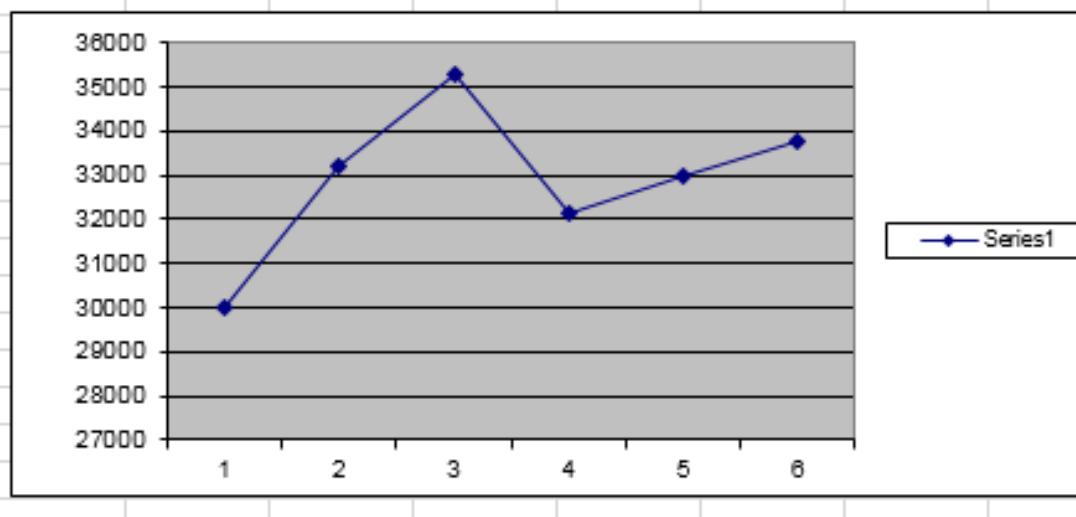
6.05

Stock Market Follies

- Simulate a portfolio of various stocks.
- Stock prices change randomly from year to year, with various volatilities.
- Can consider different mixes for portfolio.
- Simple spreadsheet application.

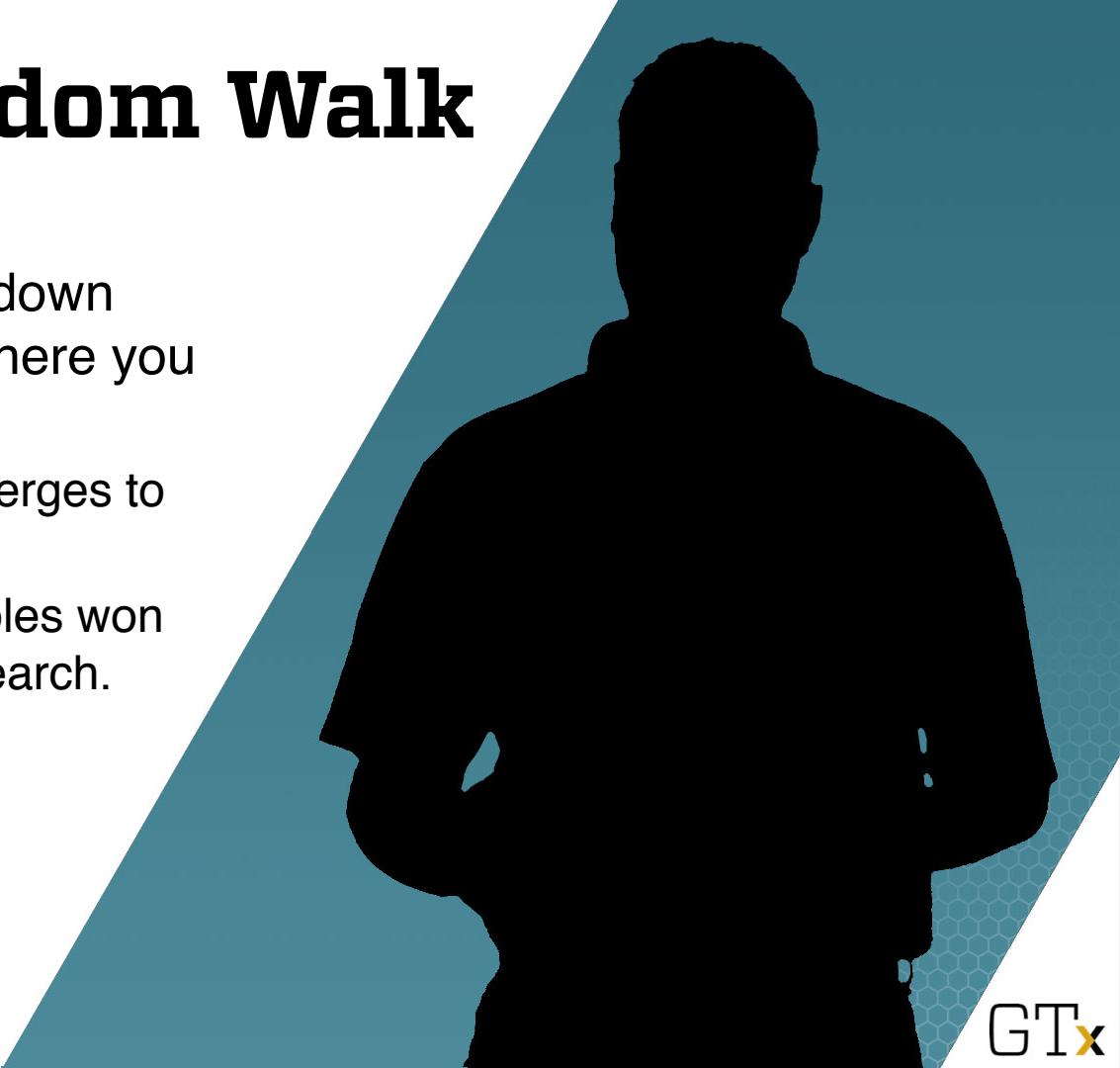
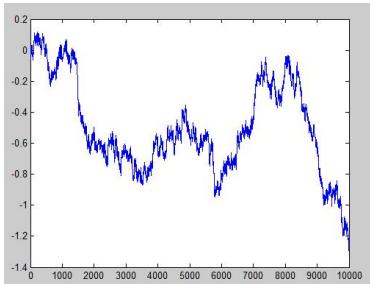


	mean	sd	year0	year1	year2	year3	year4	year5
general year performance (so stocks are correlated)				1.27	1.05	0.77	0.90	0.87
energy	0.05	0.30	5000	2701	3889	3889	3355	4262
pharmaceuticals	0.06	0.20	5000	7730	6913	6522	8764	8136
entertainment	0.04	0.10	5000	6813	8078	6633	5900	5683
insurance	0.07	0.05	5000	6977	7820	6234	6616	6213
banking	0.06	0.30	5000	5238	5935	6803	5837	6240
computer technology	0.18	0.50	5000	3751	2654	2030	2491	3226
Totals				30000	33210.8	35290.2	32110.5	32963.9
								33759.2



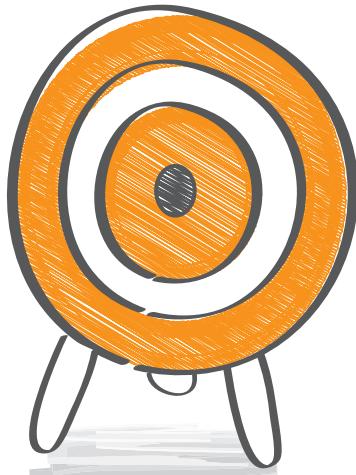
Taking a Random Walk

- Take a normal step up or down every time unit and plot where you are as time progresses.
 - This “random walk” converges to [Brownian motion](#).
 - Einstein and Black+Scholes won Nobel Prizes for this research.



Summary

- We ran simulations on more easy examples, all of which involved randomness.
- Next Time: Let's generate that randomness on a computer!



Computer Architecture

Module 1: Introduction + Whirlwind Tour

Dave Goldsman, Ph.D.

Professor

Stewart School of Industrial and Systems Engineering

Generating Randomness

Lesson Overview

Last Time: Presented some really simple baby examples of simulations, all of which used some sort of randomness.

This Time: So how can we generate that randomness?

The big reveal: The algorithms that generate “randomness” aren’t random at all! They just appear to be to you and me.



Randomness

- Need random variables (RV's) to run the simulation, e.g., interarrival times, service times, etc.
- Generate $\text{Unif}(0,1)$ **pseudo-random** numbers (PRN's)
 - Use a **deterministic** algorithm
 - Not really random, but seem to be
- Generate other RV's
 - Start with $\text{Unif}(0,1)$'s
 - Apply transformations to get any other type of random variable



Unif(0,1) PRN's

- Deterministic algorithm
- Example: [Linear Congruential Generator](#)
 - Choose an integer “seed,” $X(0)$
 - Set $X(i) = a X(i-1) \text{ mod}(m)$, where a and m are carefully chosen constants, and mod is the modulus function
 - Set the i th PRN as $U(i) = X(i)/m$

Unif(0,1) PRN's

Pretend Example:

- Start with $X(0) = 4$ (for no reason)
- Set $X(i) = 5 X(i-1) \text{ mod}(7$
- Then $X(1) = 20 \text{ mod } 7 = 6$
- $X(2) = 2, X(3) = 3, X(4) = 1, X(5) = 5$, etc.
- So $U(1) = X(1)/m = 6/7$
- $U(2) = 2/7, U(3) = 3/7$, etc.
- Numbers not so random.

Unif(0,1) PRN's

Real Example

- $X(i) = 16807 X(i-1) \text{ mod}(2^{31} - 1)$
- $U(i) = X(i) / m$
 - This generator is used in a number of simulation languages
 - Has nice properties, including long “cycle times”
 - Better generators are out there

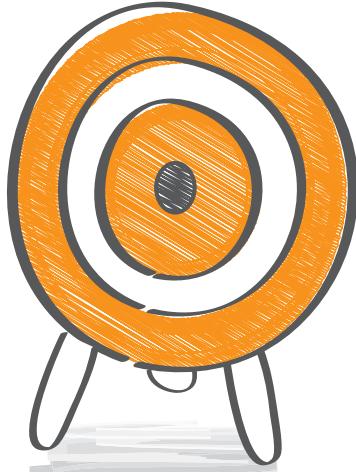
Generating Other RV's

- Start with $U(i) \sim \text{Unif}(0,1)$
- Apply some appropriate transformation
- Example: $-(1/\lambda) \ln(U(i)) \sim \text{Exp}(\lambda)$
 - Inverse transform method – can use this for various important distributions
 - Many other more-sophisticated methods available, e.g., Box-Muller method for normals



Summary

- Showed how to generate *appears* to be randomness on a computer!
- Next Time: Random input means random output. Trouble with a Capital T. What to do?



Computer Simulation

Module 1: Introduction + Whirlwind Tour

Dave Goldsman, Ph.D.

Professor

Stewart School of Industrial and Systems Engineering

Simulation Output Analysis



Lesson Overview

Last Time: How can we generate that randomness so as to run the simulation?

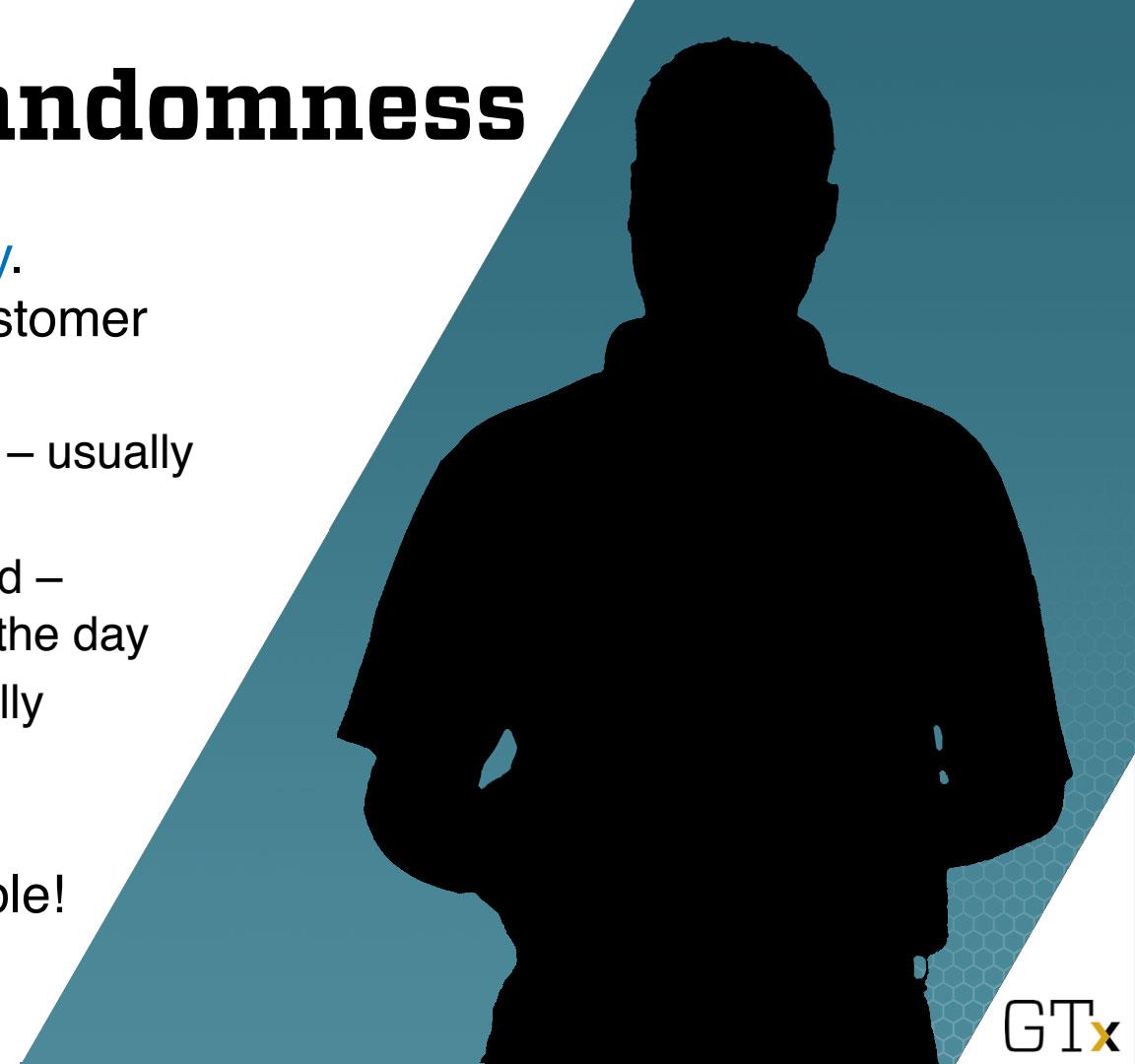
This Time: Random input means random output, which requires careful analysis. What can we do about that?

Bottom Line: Everything that taught you in your Baby Stats class is a **big fat lie** – simulation output is never independent and normal. So we'll need new methods.



Analyzing Randomness

- Simulation output is **nasty**.
Consider consecutive customer waiting times.
 - Not normally distributed – usually skewed
 - Not identically distributed – patterns change during the day
 - Not independent – usually correlated
- Can't analyze via “usual” statistics methods! Trouble!



Analyzing Randomness

Two general cases to consider

- Terminating Simulations
 - Interested in short-term behavior
 - Example: Avg customer waiting time in a bank over the course of a day
 - Example: Avg # of infected victims during a pandemic
- Steady-State Simulations
 - Interested in long-term behavior
 - Example: Long-running assembly line



Terminating Simulations

Usually analyzed via **Independent Replications**

- Make independent runs (replications) of the simulation, each under identical conditions
- Sample means from each replication are assumed to be approximately i.i.d. normal
- Use classical statistics techniques on the i.i.d. sample means (not on the original observations)



Steady-State Simulations

- First deal with initialization (start-up) bias.
 - Usually “warm up” simulation before collecting data
 - Failure to do so can ruin statistical analysis
- Many methods for dealing with steady-state data
 - Batch Means
 - Overlapping Batch Means / Spectral Analysis
 - Standardized Time Series
 - Regeneration



Steady-State Simulations

Method of Batch Means

- Make one long run (vs. many shorter reps)
- Warm up simulation before collecting data
- Chop remaining observations into contiguous batches
- Sample means from each batch are approximately i.i.d. normal
- Use classical statistics on the i.i.d. batch means



Summary

- Talked about analysis of troublesome simulation output.
- This completes Module 1, which went over introductory material to get us salivating for more!
- Coming up: Module 2 will present lots of calc/prob/stats review material from your distant past, albeit it with a little bit of simulation thrown in.

