Session 7:
Introduction to Process Simulation

Richard Welke
Director, CEPRIN
Professor, CIS
Robinson College of Business
Georgia State University
Atlanta, GA
From process description to analysis

- Process models provide
  - Shared agreement about what a process does
  - “Stand-back and critique” of current process practice
  - Static analysis of costs (processing, resource)

- However ...
  - Effectiveness of running processes are affected by “stochastic” behavior of:
    - Arrival rate of customers (the rate and variability of new transaction arrivals into the process (Start Event)
    - Service rates (the speed and variability of service times associated with each task in the process)
    - The number of “performers” available at any given time to perform the tasks
    - The relative frequency of different paths taken (and thus the business rules used to govern these (gateway) “splits”
The theory behind all this: “Queuing”

We all experience “queues” (waiting for resources to service our needs/requests)

- Common examples:
  - Telephone queues (“You’re number XX to be serviced; holding time estimated to be YY minutes)
  - Line queues (bank withdrawals, ticket-taking at events, boarding an airplane, toll booth queues)
  - Online server queues (slow responses, “try again later,” 404 errors, lagged responses to “Click Next,” etc.)

Holding the process constant one can (often dramatically) affect queuing by:

- Altering the rate/variability of new “customer” arrivals
- Reducing variability in service times
- Assigning more capacity (performers) to servicing tasks
- Changing the “queue discipline”
  - How people queue and form lines
Basics of Queuing (waiting time) theory

Arrivals \( \lambda \) → Waiting \( W_q, L_q \) → Service time \( \mu, c \)

Basic characteristics:
- \( \lambda \) (mean arrival rate) = average # of arrivals/time unit (inter-arrival rate)
- \( \mu \) (mean service rate) = average number of jobs handled by one server per time unit
- \( c \) = number of servers (capacity)
- \( W_q \) = average time in queue
- \( L_q \) = average number in queue (i.e. length of queue)

Secondary phenomena: “Aborts” and “Reneges”

Questions – what happens to queue \( (W_q, L_q) \) if:
- **Arrival rates and service times constant**
  - Arrival rates at 2/min \( (\lambda) \) and service times \( (\mu) \) at 2/min?
- **Arrival rates “stochastic”**
  - Arrival rates at 2/min +/- 0.25 min; ditto service rates?
Solution approaches to queuing (OR)

“Closed form” (mathematical formula) solutions

- Can **calculate** queue length and waiting times
  - Make specific assumptions about
    - Arrival rates and distributions
    - Queue behavior (e.g. FIFO, reneges, etc.)
    - Servers and service time rates and distributions
    - Configurations of multi-stage queues

- For example: **M/M/1 Queue assumptions** ...
  - Arrivals follow a Poisson distribution (mean: $\lambda$)
  - Service rates follow a negative exponential distrib. (mean: $\mu$)
  - Only 1 server for the queue and it has FIFO queue discipline

\[
\rho = \frac{\text{Capacity Demand}}{\text{Available Capacity}} = \frac{\lambda}{\mu}
\]

\[
L = \rho/(1-\rho)
\]
\[
W = L/\lambda = 1/(\mu - \lambda)
\]
\[
L_q = \rho^2/(1-\rho) = L - \rho
\]
\[
W_q = L_q/\lambda = \lambda / (\mu(\mu - \lambda))
\]

(CIS4120 Fa13 Session 7: Simulation Intro)
Example application of “closed form”

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>One doctor (c=1)</th>
<th>Two Doctors (c=2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \rho )</td>
<td>( \frac{2}{3} )</td>
<td>( \frac{1}{3} )</td>
</tr>
<tr>
<td>( L_q )</td>
<td>( \frac{4}{3} ) patients</td>
<td>( \frac{1}{12} ) patients</td>
</tr>
<tr>
<td>( L )</td>
<td>( 2 ) patients</td>
<td>( \frac{3}{4} ) patients</td>
</tr>
<tr>
<td>( W_q )</td>
<td>( \frac{2}{3} ) h = 40 minutes</td>
<td>( \frac{1}{24} ) h = 2.5 minutes</td>
</tr>
<tr>
<td>( W )</td>
<td>( 1 ) h</td>
<td>( \frac{3}{8} ) h = 22.5 minutes</td>
</tr>
</tbody>
</table>

Question 1: How many doctor’s should be assigned to this process to service the patients?

\( \lambda = 2 \) patients per hour

\( \mu = 3 \) patients per hour

Question 2: What assumptions are made here about means, etc.?
Closed form vs. simulation (pros/cons)

Queuing theory limitations:
- Generally not applicable when system includes multi-stage queues
- Requires case-by-case mathematical analysis
- Assumes “steady-state” (valid only for “long-term” analysis)
- Limited range of statistical distributions for arrivals and services

Process simulation more versatile
- Run a (large) number of process instances, gather data (cost, duration, resource usage) and calculate statistics from resulting output

However …
- Takes computational time
- Must attend to various issues related to statistical accuracy
  - Random number generation
  - “Burn-in” (pre-loading) of scenarios
Another consideration

“Validity”

- Analytic/Numerical validity
  - Closed form solutions published have scientific validity
  - They require “trust” of those deciding in the basis of the formulae used and person presenting them

- Face validity
  - The process model “looks like” what’s actually occurring
  - A simulation dynamically shows “build ups” of queues and related data
  - The results appear based on decider’s experience/knowledge of the process; can more easily “play” what-if dynamically

Reality

- Face validity often trumps analytical or numeric validity (when it comes to deciding alternatives)
Simulation steps with Process Modeler

Steps in evaluating a process with simulation

1. Model the process (e.g. BPMN)
2. Enhance the process model with simulation info
   - simulation model
   - Based on assumptions or better based on data (logs)
3. Run the simulation
4. Analyze the simulation outputs
   1. Process duration and cost stats and histograms
   2. Waiting times (per activity)
   3. Resource utilization (per resource)
5. Repeat for alternative scenarios
BizAgi simulation “levels”

- **Process validation**
  - Checks to see if the process is “simulation ready”
  - Assumes equal likelihood splits on gateways unless you change these; infinite resources on service tasks
  - Reports errors if problems detected in process diagram

- **Time analysis**
  - Wants the arrival (start event) and service (task) distribution and timing values
  - Runs simulation assuming “infinite” performer resources

- **Resource analysis**
  - Assign performers resources to tasks (number available)
  - Simulation now limited to who and how many are available to do work

- **Calendar analysis**
  - Add-in when (what days, times) resources are available over a day, week or month
Below is a typical small bank (called SoCom) process model for “handling” customers:
Let’s first do “Time Analysis”

- **Arrival rate**
  - We’ll initially assume a fixed arrival rate of $\lambda = 3$ (per minute)
  - That means the inter-arrival rate is $1/\lambda$ or $1/3$ minutes per arrival (average) of customers at the bank or one every 20 seconds
  - As 90% will be normal banking customers and there are two queues, they’ll be one customer every $20/0.9 \times 2 = 44.4$ secs arriving in each “normal” queue

- **Service times**
  - We’ll initially assume a servicing time of $\mu = 2/3$ minute (44 seconds) per transaction for normal task servicing
  - $(20/0.10 = 3$ minute 20 sec’s (200 seconds) per business transaction servicing

We’ll run 3000 arrivals through the system to see what we experience with these values

In theory, since the average arrival rates and service rates are “matched” we shouldn’t see any service delays

Let’s see what the simulation produces …
Specifying the proceeding to BizAgi (1)

- First, open “Simulation View”
- Next, click on the start event
  - You’ll see a “gear” symbol – click on it
  - Then you’ll see a dialog box:
    - Fill-in as shown for type of distribution (we’ll initially assume non-random) and inter-arrival rate (one every 1/3 of a minute)
- Then click on the gateway for customer type
  - We’ll assume a 10% business, 90% normal split which we show as follows:

  ![Customer](image)

  ![Probability](image)

  Hint: Move the slider shown in the red box to achieve the 10-90 split.
Specifying the proceeding to BizAgi (2)

- Similarly, we’ll create the “which queue” split on a 50-50 basis:

![Probability table](image)

- Then we’ll specify the service times for each of the tasks:
  - Service commercial (one server; $\mu = 3$ mins 20 secs)
  - Service regular (one server/each queue; $\mu = 44$ secs)
  - Note: These service times exactly match the arrival rates to the various queues!
Now let’s run the simulation with these

Note: The only variability we’ve specified so far is in the gateway split behavior; everything else is deterministic.
### Summary results (with infinite servers)

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Instances completed</th>
<th>Instances started</th>
<th>Min. time</th>
<th>Max. time</th>
<th>Avg. time</th>
<th>Total time</th>
</tr>
</thead>
<tbody>
<tr>
<td>SoCom Bank</td>
<td>Process</td>
<td>3,000</td>
<td>3,000</td>
<td>44s</td>
<td>3m 20s</td>
<td>57s</td>
<td>2d 1m 12s</td>
</tr>
<tr>
<td>Bank customer arrives for service</td>
<td>Start event</td>
<td>3,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Customer type?</td>
<td>Gateway</td>
<td>3,000</td>
<td>3,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Which queue?</td>
<td>Gateway</td>
<td>2,738</td>
<td>2,738</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Complete business xact</td>
<td>End event</td>
<td>262</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Complete regular xact</td>
<td>End event</td>
<td>2,738</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service business banking</td>
<td>Task</td>
<td>262</td>
<td>262</td>
<td>3m 20s</td>
<td>3m 20s</td>
<td>3m 20s</td>
<td>14h 33m 20s</td>
</tr>
<tr>
<td>Service regular banking-1</td>
<td>Task</td>
<td>1,378</td>
<td>1,378</td>
<td>44s</td>
<td>44s</td>
<td>44s</td>
<td>16h 50m 32s</td>
</tr>
<tr>
<td>Service regular banking-2</td>
<td>Task</td>
<td>1,360</td>
<td>1,360</td>
<td>44s</td>
<td>44s</td>
<td>44s</td>
<td>16h 37m 20s</td>
</tr>
</tbody>
</table>
Additional specs you have to add

Start with prior process model, add the following...

- Arrival rate distribution of process instances
  - Mean, statistical distribution type
- Task (service) processing times & distribution(s)
  - Per activity or per activity-resource pair
- Resource assignment
  - Mapping from activities to resource classes
  - Which available resources are able to perform tasks
    - People, equipment (or both)
    - When (calendar by day/time)
- Costs
  - Per activity and/or per activity-resource pair
- Conditional branching probabilities (XOR/OR splits)
  - Alternative “business rules” governing paths taken
Arrival rate distributions (Poisson)

- Poisson arrival process:
  - Common arrival assumption in many queuing and simulation models
  - Times between arrivals are independent, identically distributed & exponential
    - \( P(\text{arrival} < t) = 1 - e^{-\lambda t} \)
    - Arrival rate is negative exponential distributed

- Fact that a certain event has not happened tells us nothing about how long it will take to happen
  - e.g., \( P(X > 1 \mid X \geq y) = P(X > 1) \)
How a simulator deals with RV’s

- **Concept of a “random variable”**
  - The variable “Y” in an equation such as \( X = Y + Z \) takes on a fixed (deterministic) value
  - If Y is a random variable, it “samples” from a (cumulative) distribution of possible values with different likelihoods of these values
  - So, for example, a (cumulative) normal distribution sampling would look like:

Random number generation zero and one from simulator to obtain the next simulation value (e.g., 0.60 shown; resulting in a time of the mean + 0.31 times mean value.
Service rate distributions

- Generally, lots of choices
  - Exponentially distributed
  - Normally distributed
    - Can be full or “truncated” (limits on upper/lower bounds)
    - Also an approximation for exponential if large sample size
  - Uniformly distributed (min, max)
  - Triangular, Erlang, etc. also used
- Best way to decide is to sample (observe and record)
- Or chose the generally-accepted, typical behavior
  - Poisson arrival rates (exponential inter-arrival rates)
  - Exponential/normal distributed service times
Return to the bank simulation

- We’ll now choose a Poisson arrival rate for customers to the bank
- We’ll elect an exponential service rate for servicing the various customer types
- We’ll keep the “average” service times the same and the split between business and “regular” the same
- We’re interested in how “randomness” affects the overall customer experience (total xact time)
  - Exponential distribution inter-arrival rates
  - Exponential distribution service times
  - Mean values the same as before
From deterministic to random (but “infinite” performers)

- **Arrival rates**
  - Set distribution as “negative exponential”
  - Inter-arrival rate of 20 secs

![Control panel showing negative exponential distribution](image)

- **Service times (example)**
  - Means as before ($\mu = 44$ secs normal, $\mu = 200$ secs business)

![Service time distribution](image)
Re-running simulation with distributions

- But with (still) infinite number of servers
- Consider this “best case”

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Instances completed</th>
<th>Instances started</th>
<th>Min. time</th>
<th>Max. time</th>
<th>Avg. time</th>
</tr>
</thead>
<tbody>
<tr>
<td>SoCom Bank</td>
<td>Process</td>
<td>3,000</td>
<td>3,000</td>
<td>0</td>
<td>24m 48s</td>
<td>1m 1s</td>
</tr>
<tr>
<td>Bank customer arrives for service</td>
<td>Start event</td>
<td>3,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Customer type?</td>
<td>Gateway</td>
<td>3,000</td>
<td>3,000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Which queue?</td>
<td>Gateway</td>
<td>2,693</td>
<td>2,693</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Complete business xact</td>
<td>End event</td>
<td>307</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Complete regular xact</td>
<td>End event</td>
<td>2,693</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service business banking</td>
<td>Task</td>
<td>307</td>
<td>307</td>
<td>0 s</td>
<td>24m 48s</td>
<td>3m 31s</td>
</tr>
<tr>
<td>Service regular banking-1</td>
<td>Task</td>
<td>1,398</td>
<td>1,398</td>
<td>0 s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service irregular banking-2</td>
<td>Task</td>
<td>1,295</td>
<td>1,295</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
However …

- We need to know how long clients wait in queues.
- To find this out we go to the next level:
  - Constrained resources (one person servicing each of the queues)
- First specify the resources:
Resources

- We’ll add two resources:
  - Commercial teller (one of these available for work)
  - Non-commercial teller (two of these available)

- We could also add costing information for each resource to obtain total running costs
- Next we’ll assign these resources to the three tasks
Re-running the simulation

With constrained resources

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Instances completed</th>
<th>Instances started</th>
<th>Min. time</th>
<th>Max. time</th>
<th>Avg. time</th>
<th>Total time</th>
<th>Min. time waiting resource</th>
<th>Max. time waiting resource</th>
<th>Avg. time waiting for resource</th>
<th>Standard deviation waiting resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>SoCom Bank</td>
<td>Process</td>
<td>5,000</td>
<td>5,000</td>
<td>0 s</td>
<td>1h 49m 38s</td>
<td>16m 9s</td>
<td>2d 3h 26m 29s</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bank customer arrives for service</td>
<td>Start event</td>
<td>3,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Customer type?</td>
<td>Gateway</td>
<td>3,000</td>
<td>3,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Which queue?</td>
<td>Gateway</td>
<td>2,683</td>
<td>2,683</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Complete business xact</td>
<td>End event</td>
<td>307</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Complete regular xact</td>
<td>End event</td>
<td>2,683</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service banking</td>
<td>Task</td>
<td>507</td>
<td>507</td>
<td>8 s</td>
<td>1h 49m 38s</td>
<td>58m 36s</td>
<td>12d 17h 1m 27s</td>
<td>0</td>
<td>1h 47m 52s</td>
<td>56m 5s</td>
<td>28m 14s</td>
</tr>
<tr>
<td>Service regular banking-1</td>
<td>Task</td>
<td>1,398</td>
<td>1,398</td>
<td>0 s</td>
<td>26m 17s</td>
<td>11m 14s</td>
<td>10d 22h 18m 11s</td>
<td>0</td>
<td>25m 14s</td>
<td>10m 31s</td>
<td>7m 4s</td>
</tr>
<tr>
<td>Service regular banking-2</td>
<td>Task</td>
<td>1,205</td>
<td>1,205</td>
<td>0 s</td>
<td>27m 55s</td>
<td>11m 8s</td>
<td>10d 23h 3s</td>
<td>0</td>
<td>25m 7s</td>
<td>10m 24s</td>
<td>7m 4s</td>
</tr>
</tbody>
</table>
What happens if the bank’s successful?

- The arrival rate average increases!
  - Let’s assume it increases by 10%
  - Inter-arrival rate goes from 20 secs to 18 secs.
  - No change in service staffing or processing time

### Control
- Arrival interval [secs]
  - Poisson Distribution
  - Mean: 18.00

### Table
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Instances completed</th>
<th>Instances started</th>
<th>Min. time</th>
<th>Max. time</th>
<th>Avg. time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service business banking</td>
<td>Task</td>
<td>307</td>
<td>307</td>
<td>8s</td>
<td>3h 19m 35s</td>
<td>1h 41m 43s</td>
</tr>
<tr>
<td>Service regular banking-1</td>
<td>Task</td>
<td>1,398</td>
<td>1,398</td>
<td>0s</td>
<td>1h 49m 45s</td>
<td>50m 52s</td>
</tr>
<tr>
<td>Service irregular banking-2</td>
<td>Task</td>
<td>1,295</td>
<td>1,295</td>
<td>2s</td>
<td>1h 50m 27s</td>
<td>50m 39s</td>
</tr>
</tbody>
</table>
Considering alternatives

What could we do to improve the customer wait times?
Without increasing costs?
1. 
2. 

With increased costs? What’s the justification?
1. 
2. 
3.
Altering the “queue” formation

- Go from multiple personal banking queues to a single line with multiple servers
  - Costs the same as we’re not changing type/number of performers

Each serving queue requires one of these roles (performers) to do its work
Simulation results

- Using original inter-arrival rate of 1 per 20 secs.
- Prior results (with three queues)

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Instances completed</th>
<th>Instances started</th>
<th>Min. time</th>
<th>Max. time</th>
<th>Avg. time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service business banking</td>
<td>Task</td>
<td>307</td>
<td>307</td>
<td>8s</td>
<td>1h 49m 38s</td>
<td>59m 36s</td>
</tr>
<tr>
<td>Service regular banking-1</td>
<td>Task</td>
<td>1,398</td>
<td>1,398</td>
<td>0s</td>
<td>26m 19s</td>
<td>11m 16s</td>
</tr>
<tr>
<td>Service regular banking-2</td>
<td>Task</td>
<td>1,295</td>
<td>1,295</td>
<td>0s</td>
<td>27m 55s</td>
<td>11m 8s</td>
</tr>
</tbody>
</table>

- Re-running the simulation (with two queues)

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Instances completed</th>
<th>Instances started</th>
<th>Min. time</th>
<th>Max. time</th>
<th>Avg. time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service business banking</td>
<td>Task</td>
<td>292</td>
<td>292</td>
<td>8s</td>
<td>1h 20m 52s</td>
<td>44m 22s</td>
</tr>
<tr>
<td>Service personal banking</td>
<td>Task</td>
<td>2,708</td>
<td>2,708</td>
<td>0s</td>
<td>43m 5s</td>
<td>20m 56s</td>
</tr>
</tbody>
</table>