Optimization Theory and Methods

2025 Autumn

同济经管 TONGJI SEM

魏可佶 kejiwei@tongji.edu.cn

https://kejiwei.github.io/

CAMEA 中国高质量MBA教育认证 AACSB ACCREDITED EQUIS ACCREDITED

- 16.1 Types of Simulation Models
- 16.2 Discrete event simulation approaches
- 16.3 Example of event-driven simulation
- 16.4 Statistical analysis of results of a simulation
- 16.5 Repair Shop Case Study

16.1 Types of Simulation Models

↳ Hierarchical structure of simulation models

- Simulation
  - Static
  - Dynamic
    - Continuous Event
    - Discrete Event
      - Time-Driven
      - Event-Driven

↳ Hierarchical structure of simulation models(cont.)

- The two major categories of simulation models are **static** and **dynamic**. In a static simulation, time evolution is not important, while in dynamic simulation it is.
- We will focus on dynamic simulation in this course.
- Within dynamic simulations, the two main categories are **discrete event** simulation and **continuous event** simulation.
- In continuous event simulation, the relevant random variables change continuously and the typical way of modeling these involves solving differential equations.
- In discrete event simulation, the events of relevance happen at discrete points in time and the relevant random variables stay constant between discrete events.
- We will focus on discrete event simulation in this class.

↳ Time-driven vs. Event-driven

- Within discrete event simulation, there are two general approaches: time-driven simulation (also called fixed-time-increment simulation) and event-driven simulation (also called event paced simulation).
- In **time-driven simulation**, the time is advanced by regular intervals, irrespective of whether any event happens during that regular time increment or not. This is useful in certain settings, e.g. large scale simulations with several small events occurring frequently.
- In **event-driven simulation**, clock is advanced only to certain instants in time, when important events occur. This approach is computationally efficient when the focus is on important but less frequent events and the models are often more detailed. In this class, we will focus on <u>event-driven</u> simulation.
- A large majority of useful event-driven simulations can be thought of as generalized queuing systems or queuing networks.

- In an event-driven simulation, the following entities are tracked:

1. **Simulation clock**: Indicates the amount of time that passed since the simulation started.
2. **State of the system**: Set of variables whose values completely define the present status of a system being simulated.
3. **Events**: Instants of time when the state of the system changes. The most important component of an event-driven simulation. System state is updated and information is recorded after each event.
4. **State transition**: Set of changes in the state of the system when an event occurs.
5. **Simulation end**: When the simulation clock exceeds a pre-specified time or a pre-specified number of certain types of events, the simulation is stopped.

↳ The core entities of the M|M|1 queuing model

- **Simulation clock:** It is convenient to start with an empty system. First customer arrives when the simulation starts. First service starts with the entry of the first customer.
- **State of the system:** Described by a single variable $N(t)$= Number of customers in the queueing system at time t.
- **Events:** There are two main types of events, (1) arrival of a customer, and (2) service completion for the customer currently in service.
- **State transition:** $N(t) =$

$$\begin{cases} N(preceding\ event) + 1 & if\ the\ event\ is\ an\ arrival \\ N(preceding\ event) - 1 & if\ the\ event\ is\ a\ service\ completion \end{cases}$$

- **Simulation end:** When simulation clock exceeds a pre-specified limit or when number of customers exceeds a pre-specified limit.

- $\boldsymbol{A_n}$: Arrival time of customer n.
- $\boldsymbol{D_n}$: Departure time (service completion time) of customer n.
- $\boldsymbol{H_n}$: Inter-arrival time of customer n (i.e., time between arrivals of $(n-1)^{\text{th}}$ and n^{th} customers).
- $\boldsymbol{S_n}$: Service time of customer n.
- $\boldsymbol{A_1} = 0$, $\boldsymbol{A_n} = A_{n-1} + H_n$ for $n = 2,3,\ldots$
- Service start time for customer n = $\max(A_n, D_{n-1})$.
- $\boldsymbol{D_1} = S_1$, $\boldsymbol{D_n} = S_n + \max(A_n, D_{n-1})$ for $n = 2,3,\ldots$
- $\boldsymbol{H_n}$ **and** $\boldsymbol{S_n}$ are two exponentially distributed random variables.
- In MATLAB, these exponentially distributed random variables can be generated as $-\log(rand())/\lambda$ and $-\log(rand())/\mu$ respectively.
- Alternatively, we could also use $poissrnd(\lambda)$ and $poissrnd(\mu)$ respectively.

- So the implementation will be as follows.

FOR $n = 1{:}\, n_{max}$

- $H_n = -\log(rand())/\lambda.$
- $S_n = -\log(rand())/\mu.$
- $A_n = \begin{cases} A_{n-1} + H_n & if\ n > 1 \\ 0 & if\ n = 1 \end{cases}.$
- $D_n = \begin{cases} S_n + \max(A_n, D_{n-1}) & if\ n > 1 \\ S_1 & if\ n = 1 \end{cases}.$
- (optionally) IF $D_n > T_{max}$, THEN STOP.

END

- In case of the M|M|1 example, the relevant system statistics are, W, L, W_q and L_q. They can be estimated using the following formulas. Assume that M is the total number of customers simulated.

- $W = \frac{\sum_{n=1}^{M}(D_n - A_n)}{M}$.

- $L = \frac{\sum_{n=1}^{M}(D_n - A_n)}{D_M}$ (Why?).

- $W_q = \frac{\sum_{n=1}^{M}(D_n - A_n - S_n)}{M}$.

- $L_q = \frac{\sum_{n=1}^{M}(D_n - A_n - S_n)}{D_M}$.

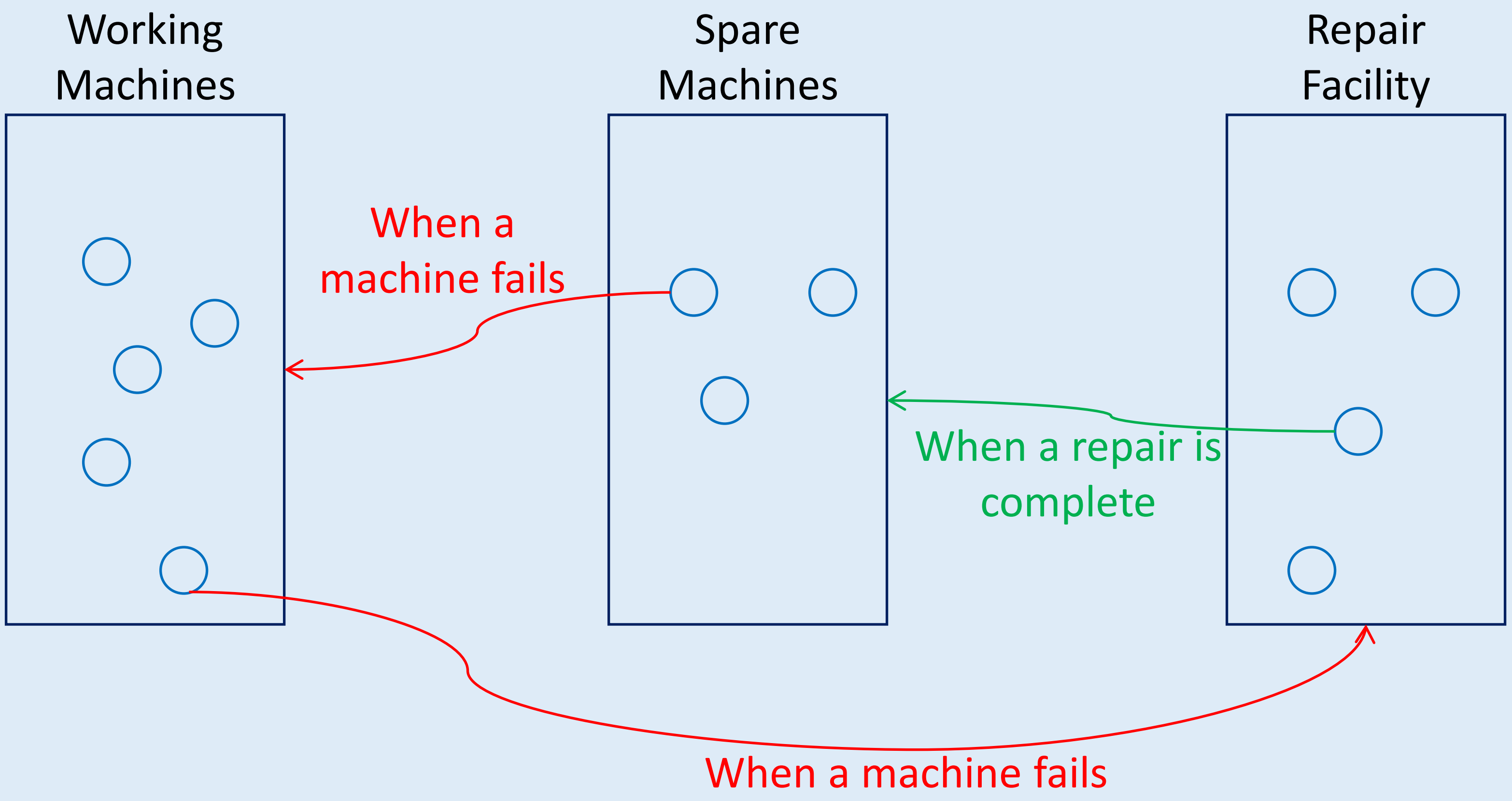
- Statistical analysis of results of a simulation can be difficult.
- Simulation results can be thought of as analogous to collecting a physical sample. So we could conduct hypothesis testing and confidence interval estimation.
- If we are interested in estimating the system characteristics under a steady state:
  - Check for the impact of initial conditions.
  - Ensure that the initial transient period is ignored, by discarding initial set of observations corresponding to a period called the simulation warm-up period.

↳Statistical Issues (cont.)

- Selection of starting condition can sometimes be tricky.
- Successive samples are often known to be correlated (e.g. wait times of successive customers in the M|M|1 example).
  - Think of the tradeoff between many runs versus a single run partitioned into multiple data points.

- At a factory, M_0 or more machines need to be working at any point of time, to keep the factory in operation. There are **$M_0 + S_0$ total machines**, M_0 operational and S_0 spare. Machines fail according to a known random process. When a machine fails, it is sent to a repair shop. The time to fix the machine is also a random variable with known distribution. Starting from a state where all $M_0 + S_0$ machines are operational, what is the expected time until the factory has to halt operation for the first time?
- Let us assume that the machines are repaired, one at a time, in the order in which they enter repair facility. Let the repair time of each machine be uniformly distributed between 0 and R hours and let the time till failure of each machine in operation be exponential with λ per hour.

↳ Machine maintenance dynamic diagram



↳ The core entities of the simulation model

- **Simulation clock:** Start with the system having no failures yet.
- **State of the system:** Described by a single variable $N(t)$ that indicates the number of machines down (i.e., in the repair facility) at any time.
- **Events:** An event happens when either a previously working machine fails, or when a repair is complete.
- **State transition:** $N(t) =$

$$\begin{cases} N(preceding\ event) + 1 & if\ the\ event\ is\ a\ failure \\ N(preceding\ event) - 1 & if\ the\ event\ is\ a\ repair\ completion \end{cases}$$

- **Simulation end:** When simulation clock exceeds a pre-specified limit or when the state of the system equals $S_0 + 1$.

↳ Model definition and event scheduling rules

- $\boldsymbol{A_n}$ = Arrival time of the n^{th} arrival into the repair facility.
- $\boldsymbol{D_n}$ = Departure time of the n^{th} departure from the repair facility.
- $\boldsymbol{F_n}$ = Time between $(n-1)^{\text{th}}$ and n^{th} failure (n^{th} inter-failure time). Note that F_1 is defined as the time from simulation start until first failure.
- $\boldsymbol{S_n}$= Repair time for n^{th} failure.
- $\boldsymbol{A_1} = F_1$, $\boldsymbol{A_n} = A_{n-1} + F_n$ for $n = 2,3,\ldots$. After computing A_n, check the state of the system $N(A_n)$. If $N(A_n) > S_0$, then stop simulation.
- Start time of the first repair $= A_1$.

- Start time of the n^{th} repair $= \max(A_n, D_{n-1})$ for $n = 2,3, \ldots$.
- $\boldsymbol{D_1} = S_1 + A_1$, $\boldsymbol{D_n} = S_n + \max(A_n, D_{n-1})$ for $n = 2,3, \ldots$
- $\boldsymbol{F_n}$ is an exponentially distributed random variable with parameter λM_0. Note that this simplifies the simulation significantly (Why?).
- $\boldsymbol{S_n} \sim U[0, R]$.

↳ Simulation model implementation

- So the implementation will be as follows.

FOR $n = 1: n_{max}$

- $F_n = -\log(rand())/(\lambda M_0)$
- $S_n = rand() * R$
- $A_n = \begin{cases} A_{n-1} + F_n & if\ n > 1 \\ F_1 & if\ n = 1 \end{cases}.$
- $D_n = \begin{cases} S_n + \max(A_n, D_{n-1}) & if\ n > 1 \\ S_1 + A_1 & if\ n = 1 \end{cases}$
- IF $n > S_0$ and $A_n < D_{n-S_0}$ THEN EXIT (Why?)
- IF $D_n > T_{max}$ THEN EXIT

END

Objective :

Key Concepts :