Advanced Practical 2024/2025

Operations Research Case

Guanlian Xiao

Department of Operations Analytics
Vrije Universiteit Amsterdam

June 2025

Meeting times

- Lectures on project and final presentation
  - Tuesday 3, 10, 17 June 11:00-12:45 in HG-05A16
  - Wednesday 25 June 9:00-17:00 in HG-07A16

- Lecture on Writing: 5 June 11:00-12:45 in HG-12A00
- Lecture on Presentation: 19 June 11:00-12:45 in HG-12A00

Agenda

- Week 1:
  - introduce basics on simulation
  - introduce the OR project
- Week 2:
  - analyze an example on G/G/1 queue with raw data
- Week 3:
  - discuss your progress, problems and next step
- Week 4:
  - final report deadline: Tuesday 24 June 23:59 hr.
  - final presentation: Wednesday 25 June in HG-07A16

Assessment: 5 components

- Writing (grade by ALP): 10%
- Presentation skills (graded by ALP): 5%
- Report quality: 60%
- Argumentations in presentation: 25%

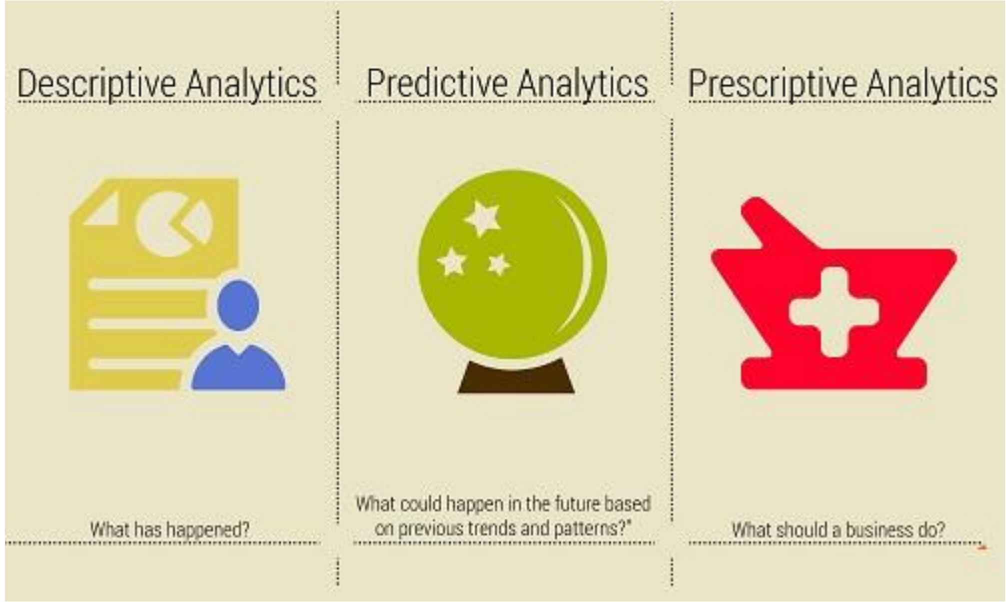
- Group Contribution Factor (GCF):0.75 and 1.05.
- Individual grade=grade of group report * GCF
  - For example: if your group assignment grade is 8 out of 10 and your Group Contribution Factor is 0.9, your final grade is calculated as, 8 * 0.9 = 7.2

- Role of AI.

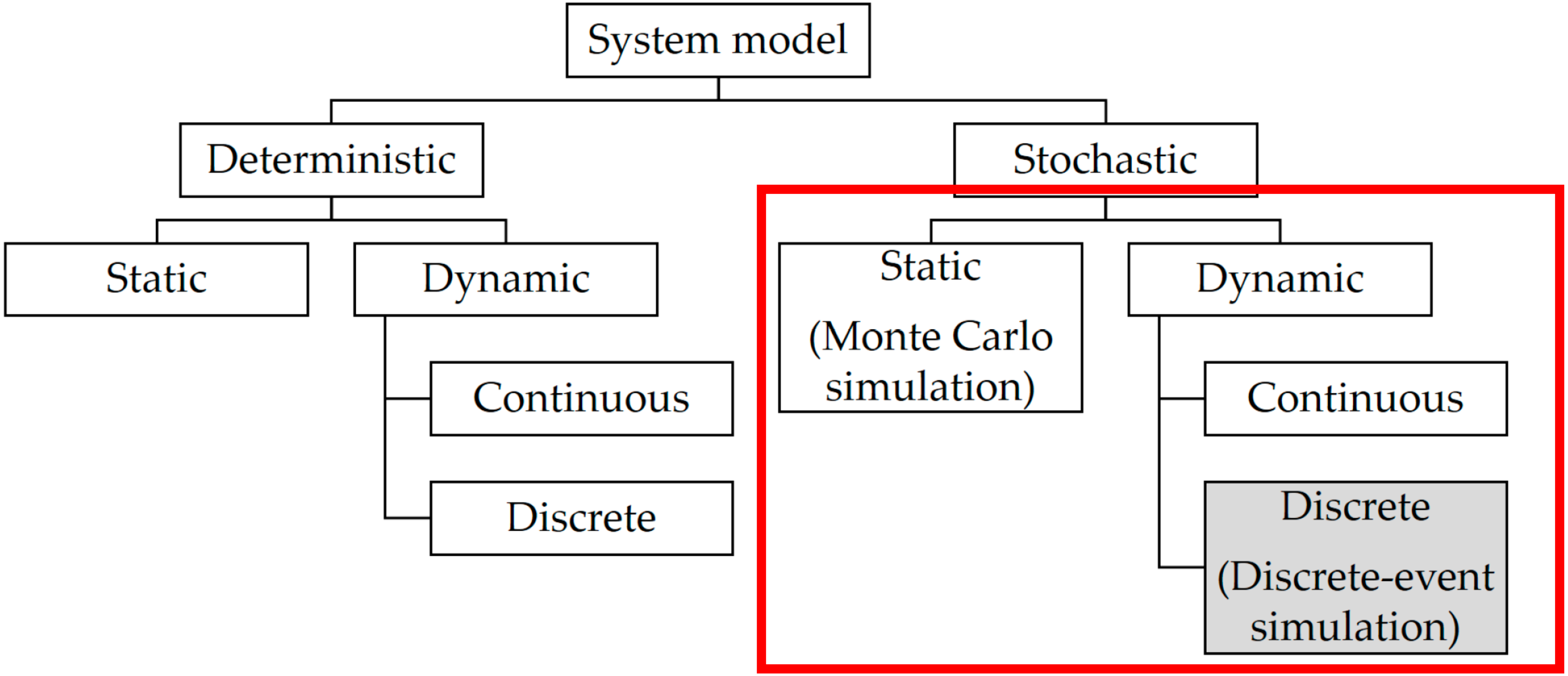
Data Analytics and Simulation

Categories of Data Analytics

- Descriptive analytics
- Predictive analytics
- Prescriptive analytics



Type of Simulation Models



Source: https://doi.org/10.3390/min7070116
https://softwaresim.com/blog/types-of-simulation-models-choosing-the-right-approach-for-your-simulation-project/

Monte Carlo Simulation

Monte Carlo Simulation

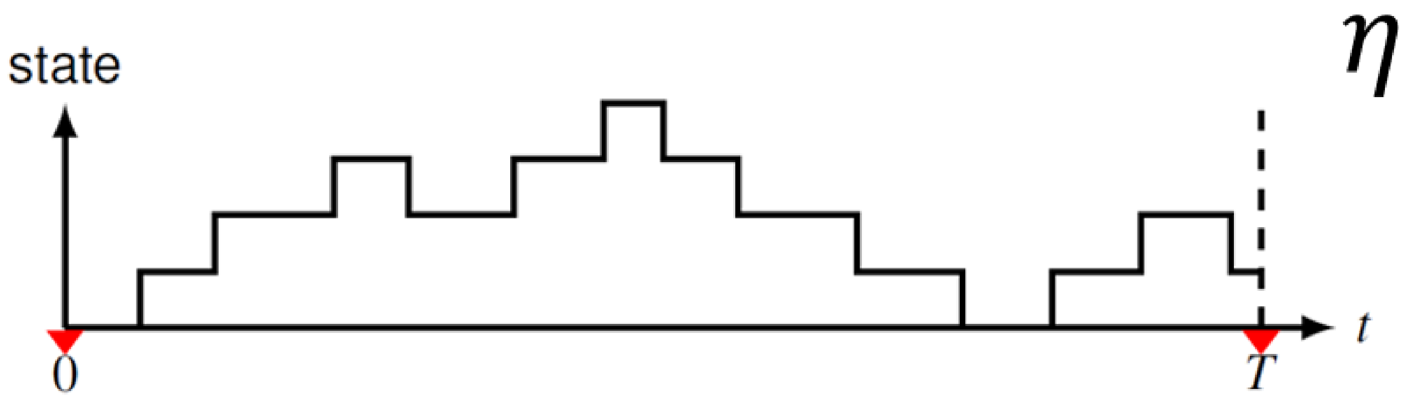
- For any unknown random variable η, how to evaluate any performance measure $E[f(\eta)]$?
- Monte Carlo Simulation!
  - Collect a large number of outcomes η_i to compute $f(\eta_i)$, then take the sample mean to approximate $E[f(\eta)]$:

$$E[f(\eta)] \approx \frac{\sum_{i=1}^{n} f(\eta_i)}{n}$$

Application of Monte Carlo Simulation for G/G/1 queue

- Considering a G/G/1 queue, how do you estimate the average waiting time of the first 20 customers arriving to a queue?
- Here η=$(\eta^1, \eta^2, \ldots, \eta^{20})$, η^i is the waiting time of the ith customer. $f(\eta) = \sum_{i=1}^{20} \eta^i / 20$ is the average waiting of the first 20 customers, which is a rv.

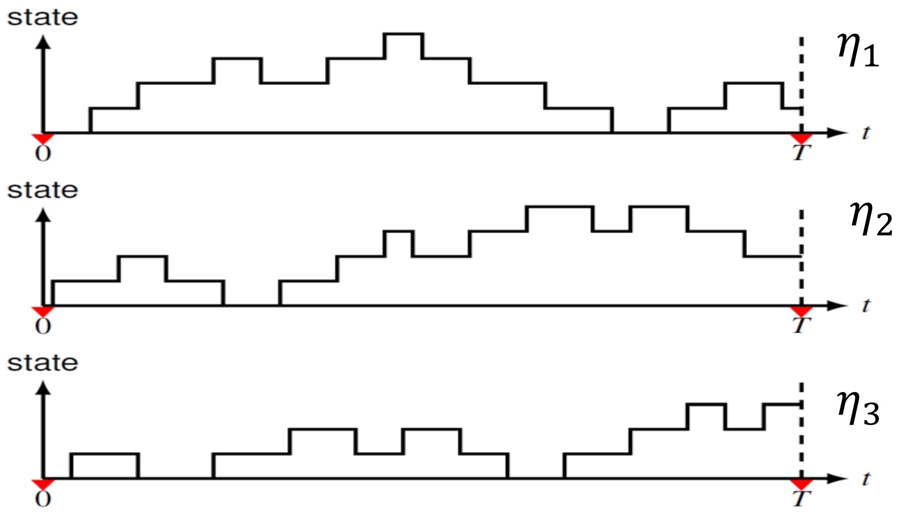
State: Number of customers in the system



How to estimate $E[f(\eta)]$??

Application of Monte Carlo Simulation for G/G/1 queue

- To estimate $E[f(\eta)]$, use Monte Carlo simulation:
  - generate a large number of sample paths $\eta_i = (\eta_i^1, \eta_i^2, \eta_i^3, \eta_i^4, \dots, \eta_i^{20})$;
  - take the sample mean $E[f(\eta)] \approx \frac{\sum_{i=1}^{n} f(\eta_i)}{n}$.



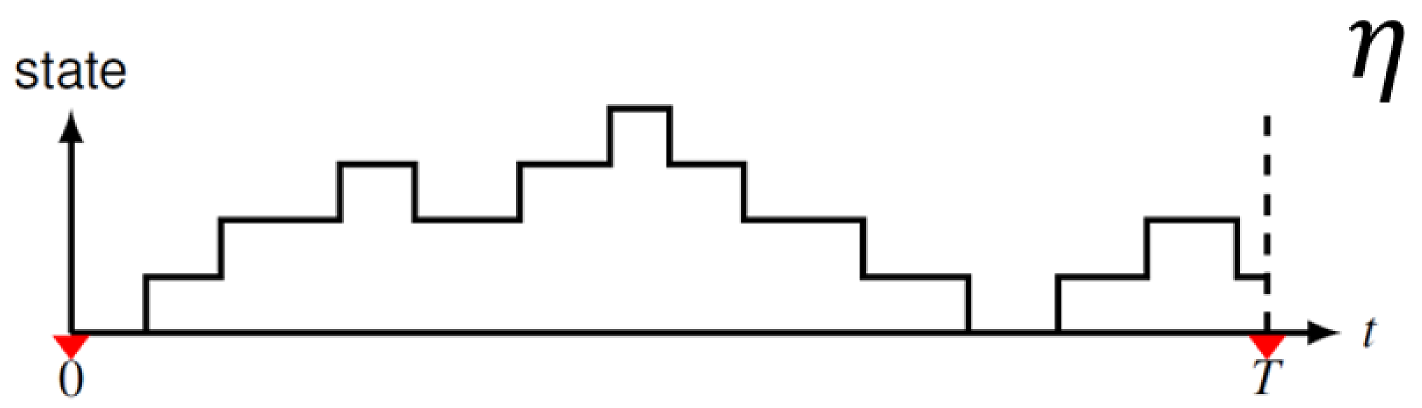
G/G/1 queue

How to collect the waiting time of each customer, i.e.,

$$\eta_i = (\eta_i^1, \eta_i^2, \eta_i^3, \eta_i^4, \ldots, \eta_i^{20})\ ?$$

Also called a sample path.

Discrete Event Simulation!



Discrete Event Simulation

A discrete-event simulation

models a system whose state may change only at discrete point in time.

Components of DES

- System state:
  - The collection of state variables necessary to described the system at a particular time
- Event
  - an instantaneous occurrence in time that may alter the state of the system
- Simulation clock (time variable):
  - A variable giving the current value of simulated time
- Event list (calendar):
  - A list containing the next time when each type of event will occur
- Statistical counters (counter variables):
  - Variables used for storing statistical information about system information, such as average request processing time, server load, or average queue length.

Discrete Event Simulation Model of G/G/1

- The DES is modeled using a 6-tuple

$$G = (S, E, L, \phi, s_0, X)$$

where

- S is the set of all states;
- E is the set of all events;
- $L : S \to 2^E$ is the active event function, forming the event list;
- $\phi: S \times E$ is the transition function;
- s_0 is the initial state;
- X =$\{X(\mathrm{e})$: e $\in E$, n = 1, 2,... $\}$ are the event life times;

Values of $G = (S, E, L, \phi, s_0, X)$ for simulation model of G/G/1

- State variable $s = (n)$, where n is the number of jobs present (in service and in queue). The state space is

$$S = \{(n), n = 0, 1, 2, 3, \ldots\}$$

- The state changes either by an arrival, or a service completion. These are the events. Denote α for arrival event. Denote β for service completion. The event space is

$$E = \{\alpha, \beta\}$$

- $L(s)$ is the list of active events in state s, called event list. These are the events that are currently scheduled to occur. $L(s)$ is listed below:

$$L(0) = \{\alpha\}$$

$$L(n) = \{\alpha, \beta\}, n = 1, 2, \ldots$$

State Transition from s to s'

State transition $\phi(s, e) = s'$ means when event e in state s occurs, the next state will be s'. Specifically,

$$\phi((n), \alpha) = (n + 1);$$

$$\phi((n), \beta) = (n - 1) \text{ for } n \geq 1.$$

The new state s' has an updated event list. For instance, $s = (0)$ with $L(s) = \{\alpha\}$ and event α occurs, then s' = (1) with $L(s') = \{\alpha, \beta\}$.

Which event $e \in L(s)$ will occur at state s?

Event remaining lifetime: which event will occur next?

- Denote the system state at event time t by s_t, and $s_t = s$ with event list $L(s)$.
  Which event in $\boldsymbol{L(s)}$ will occur next and when?

- Check the remaining lifetimes of all events in $L(s)$ at t.
  - If $e \in L(s)$ is a newly added event, then $C_t(e)$=$X(e)$, where $X(e)$ is a newly generated sample.
  - If $e \in L(s)$ is an old event added at time $\tau_e < t$, its remaining lifetime is $C_t(e)$=$X(e) - (t - \tau_e)$.

- Then the event to occur is determined by the minimum of the remaining lifetimes at t,
  $$e^* = \arg\min\{C_t(e): e \in L(s)\}$$
  And it will occur after $C_t(\mathrm{e}^*)$.

- Next we will show the complete event evolving scheme.

The Event Evolving Scheme

1. Suppose the system is empty at time $t = 0$, $s_0 = (0)$. Its associated event list is $L(s_0) = \{\alpha\}$, the event lifetime of α is $X(\alpha)$ (take a sample of interarrival time, assume $X^1(\alpha)$=τ_1), its remaining lifetime $C_t(\alpha)$=$X(\alpha)$.
   The next event is an arrival α, it occurs at τ_1.

2. State update at τ_1: $s_1 = (1)$, $L(s_1) = \{\alpha, \beta\}$.
   - Both events in $L(s_1)$ are new, generate samples for both new events (take a sample of interarrival time denoted by $X^1(\alpha)$, take a sample of service time denoted by $X^1(\beta)$)
   - Their remaining lifetimes are $C_{\tau_1}(\alpha)$=$X^1(\alpha)$and $C_{\tau_1}(\beta)$= $X^1(\beta)$
   - The next event can be an arrival or a departure, it occurs at τ_2 and τ_2= τ_1 + min($C_{\tau_1}(\alpha)$, $C_{\tau_1}(\beta)$) .
   - During (τ_1, τ_2) there is no state change. If $C_{\tau_1}(\alpha)$< $C_{\tau_1}(\beta)$, a new customer arrives at τ_2.

3. State update at τ_2: $s_2 = (2)$, $L(s_2) = \{\alpha, \beta\}$.
   - α is a new event, generate a sample $X^2(e)$ for new event as its remaining lifetime: $C_{\tau_2}(\alpha)$= $X^2(\alpha)$.
   - β is an old event, $C_{\tau_2}(\beta)$=$C_{\tau_1}(\beta) - (\tau_2 - \tau_1)$
   - The next event can be an arrival or a departure, it occurs at τ_3 and τ_3= τ_2 + min($C_{\tau_2}(\alpha)$, $C_{\tau_2}(\beta)$) and the associated event to occur is determined.

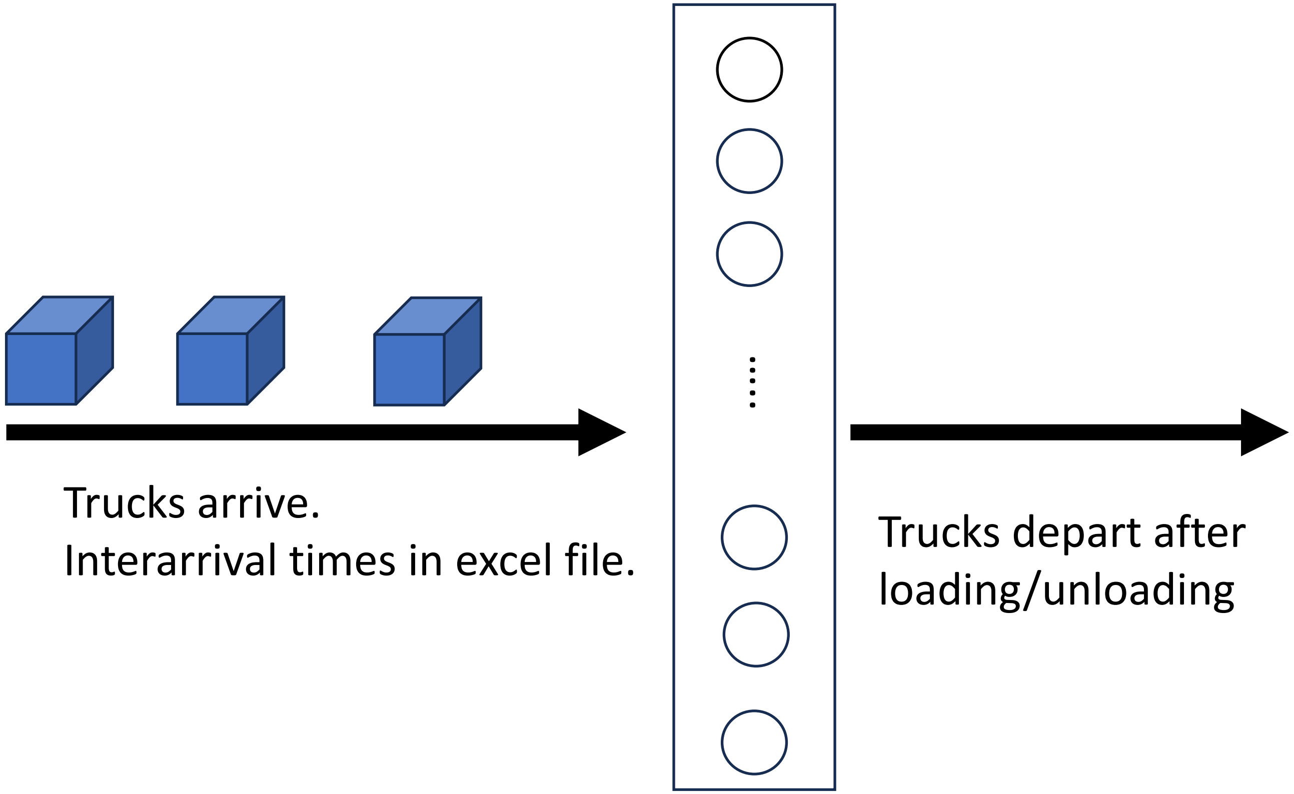
4. State update at τ_3, update state, eventlists.
   For new events generate samples, for old events, update their remaining lifetimes, and determine the next event and event occurance time.

The process continues until a stopping condition is met.

Python code

- See canvas

Project: KLM Warehouse



Given:
1 year data on interarrival times of trucks
service times at each dock

Objective:
Minimize the overall cost in the coming week, including:

average waiting cost;
labor cost;
outsourcing cost

30 docks available;
Each active dock costs 240 euros per day (labor cost);
At the end of each week, unserved trucks are outsourced with 20 euros per truck
Service times in excel file;

Project: KLM Warehouse

Tasks:

- analyze the data
- Create simulation model
- Evaluate the current policy
- Propose an improved policy and evaluate it