# Computational Decision Making for Regular People

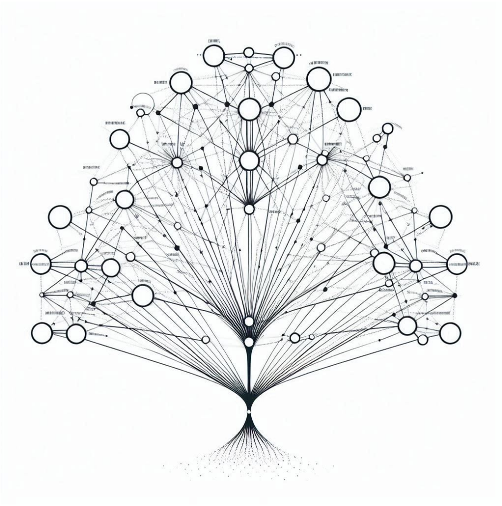
*04: Modeling Techniques*

November 5, 2024

**Nathan Davis Barrett**

# Today's Outline

1. Refresher on Math Modeling (Route Planning Problem)
2. Handling Multiple Sets At Once
3. Handling Multiple Time Periods (Talk about which set of parameters to use prior or latter)
4. Example: Multiple Time Periods
5. Handling Uncertainty
6. Example: Multiple Scenarios
7. Linearization
   - "Binary Activation ("Big M")
   - "If" statements
   - "Max" statements
   - "Min" statements



This image was created with the assistance of DALL·E 3

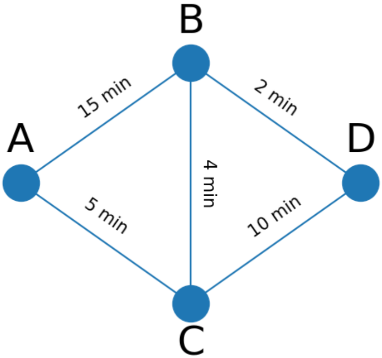
# A Note on Today's Class

Today we're going to cover a lot of material to set us up for next week when we have to tools to address real-life problems.

We'll cover a variety of individual "tools" that are somewhat disconnected from each other.

If you're feeling lost, that's okay. Go back and reference specific sections of these slides / jupyter notebooks to refresh your mind on how to use each tool if and when you need to use it.

# Refresher on Math Modeling




– – – FULL FORMULATION – – –

$$\min_{X_r, Y_p} \sum_{r \in \mathbf{R}} \delta_r X_r$$

$$s.t. \quad \sum_{r \in \mathbf{R}_p} X_r = 2Y_p \quad \forall p \in \mathbf{P}^{NON-TERM}$$

$$\sum_{r \in \mathbf{R}_p} X_r = Y_p \quad \forall p \in \mathbf{P}^{TERM}$$

$$Y_p = 1 \quad \forall p \in \mathbf{P}^{TERM}$$

$$X_r, Y_p \in \{0, 1\}$$

# Handling Multiple Sets

**Single Set:**

$$X_r \quad \forall r \in \mathbf{R}$$

$$Y_p = 1 \quad \forall p \in \mathbf{P}^{INIT}$$

**Multiple Sets:**

$$X_{r,t} \quad \forall r \in \mathbf{R}, t \in \mathbf{T}$$

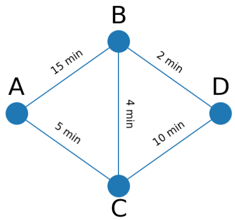
$$Y_{p,t} = 1 \quad \forall p \in \mathbf{P}^{INIT}, t \in \mathbf{T}$$

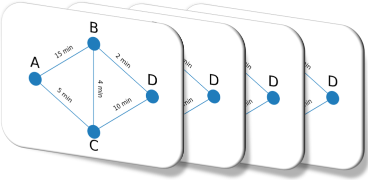
- Index each variable by multiple indices
- Iterate constraints and/or definitions over multiple sets
- An individual variable, constraint, or definition will be made for **every combination** of elements between the sets.
- In your code, you'll use the multiplication operator $(*)$ to create these combinations of sets:

$$\mathbf{R} = \{A, B\} \quad \mathbf{T} = \{0, 1, 2\} \quad \mathbf{R} * \mathbf{T} = \{(A, 0), (A, 1), (A, 2), (B, 0), (B, 1), (B, 2)\}$$

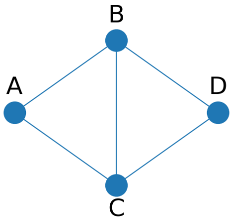
# Handling Multiple Time Periods

- Start by defining a new set (**T**) of each of the time periods you'd like to consider
- It's good to fill **T** with integers so that you can keep track of which time period is before/after which other time period.
- For each parameter, variable, and constraint that changes with time, expand it to also iterate over this new time set.
- Give special consideration/reformulation to the initial and final time point. **The constraints for these time points are likely to be different**. $(\mathbf{T} \rightarrow \mathbf{T}^{NON-TERM})$



# Example: Multiple Time Periods



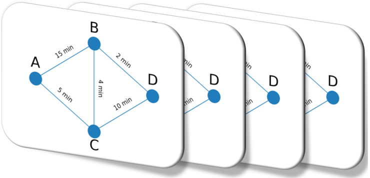

To keep things simple, let's assume that instead of time, $\delta_{r,t}$ how represents a toll amount.

| $r$ | $t = 0$ | $t = 1$ | $t = 2$ | $t = 3$ |
|---|---|---|---|---|
| $AB$ | \$15 | \$10 | \$8 | \$8 |
| $AC$ | \$5 | \$5 | \$9 | \$10 |
| $BC$ | \$4 | \$10 | \$13 | \$13 |
| $BD$ | \$2 | \$2 | \$6 | \$5 |
| $CD$ | \$10 | \$15 | \$4 | \$7 |

**Try on your own (on paper or your computer) to come up with how this problem should be re-formulated.**

(5 minutes)

# Example: Multiple Time Periods




$$\min_{X_{r,t}, Y_{r,t}} \sum_{r \in \mathbf{R}} \sum_{t \in \mathbf{T}} \delta_{r,t} X_{r,t}$$

$$s.t. \quad Y_{p,t+1} - Y_{p,t} = \sum_{r \in \mathbf{R}_p^+} X_{r,t} - \sum_{r \in \mathbf{R}_p^-} X_{r,t}$$

$$\forall p \in \mathbf{P}, t \in \mathbf{T} \neq 3$$

$$\sum_{r \in \mathbf{R}} X_{r,t} \leq 1 \quad \forall t \in \mathbf{T}$$

$$\sum_{p \in \mathbf{P}} Y_{p,t} = 1 \quad \forall t \in \mathbf{T}$$

$$Y_{A,0} = 1$$

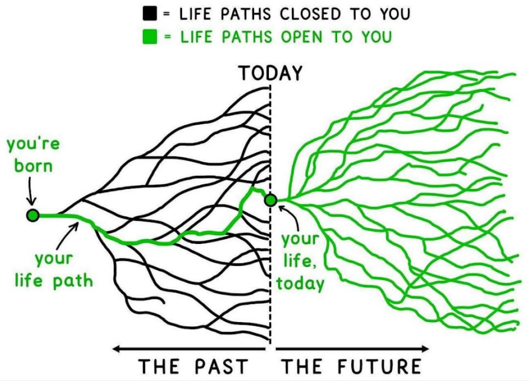
$$Y_{2,D} = 1$$

$$X_{r,t}, Y_{p,t} \in \{0, 1\}$$

*The code and solution for this problem can be found in the "Solutions.ipynb" notebook in this week's folder.*

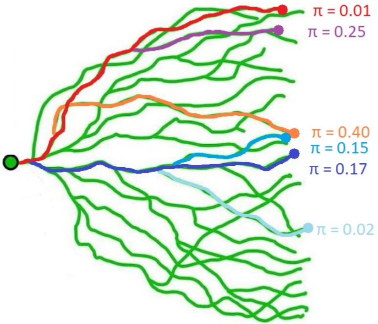
# Handling Uncertainty

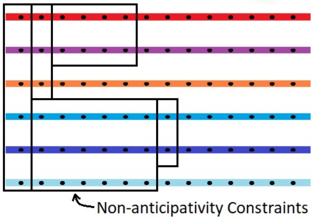
- This is an active field of research. If you're interested in learning more, the words to look up are "Stochastic Programming".
- Scenarios ($s \in \mathbf{S}$)
  - An enumeration of all possible outcomes
  - Each path should have an associated probability $\pi_s$
  - All probabilities should sum to 1 (100%)
- Coming up with these is one of the hardest parts
  - Virtually impossible to enumerate **all** outcomes.
  - Very hard to estimate the probability of each outcome.



# Handling Uncertainty

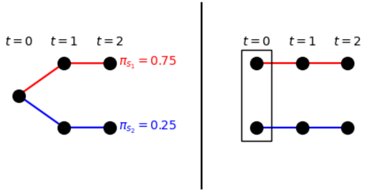
- Some variables are shared by all scenarios
  - Decisions that need to be made here and now that impact all scenarios moving forward.
- Some variables can be made in the future (and are thus specific to each scenario)
  - Each scenario get's a unique copy of that variable (e.g. iterate over $\mathbf{S}$)
  - But at certain points in time, certain scenarios are not yet distinguishable. In order to prevent the solution from "anticipating" future changes that are not yet clear, we need "non-anticipativity constraints"
  - For each box, select on representative scenario ($s_1$), all other scenarios ($s_2$) should equal that scenario
  - $X_{s_1,t} = X_{s_2,t} \quad \forall s_2 \in \mathbf{S}^{BOX}, t \in \mathbf{T}^{BOX}$



# Example: Handling Uncertainty

- Toll rates **right now** are known.
- Toll rates in 1 hour (the next time period) are unknown
  - Congested Scenario ($s_1$)
  - Clear roads Scenario ($s_2$)




$\delta_{r,t,s_1}$

| $r$ | $t=0$ | $t=1$ | $t=2$ | $t=3$ |
|---|---|---|---|---|
| $AB$ | \$5 | \$10 | \$12 | \$9 |
| $AC$ | \$30 | \$10 | \$12 | \$ 13 |
| $BC$ | \$4 | \$5 | \$6 | \$7 |
| $BD$ | \$2 | \$25 | \$50 | \$2 |
| $CD$ | \$10 | \$12 | \$13 | \$15 |

$\delta_{r,t,s_2}$

| $r$ | $t=0$ | $t=1$ | $t=2$ | $t=3$ |
|---|---|---|---|---|
| $AB$ | \$5 | \$8 | \$10 | \$7 |
| $AC$ | \$30 | \$5 | \$9 | \$10 |
| $BC$ | \$4 | \$10 | \$13 | \$13 |
| $BD$ | \$2 | \$2 | \$6 | \$6 |
| $CD$ | \$10 | \$15 | \$4 | \$10 |

| $s$ | $\pi_s$ |
|---|---|
| 1 | 0.75 |
| 2 | 0.25 |

**Try on your own (on paper or your computer) to come up with how this problem should be re-formulated.** (5 minutes)

# Example: Handling Uncertainty

$$\min_{X_{r,t,s},Y_{r,t,s}} \sum_{s\in\mathbf{S}} \pi_s \left( \sum_{r\in\mathbf{R}} \sum_{t\in\mathbf{T}} \delta_{r,t,s} X_{r,t,s} \right)$$

$$s.t. \quad Y_{p,t+1,s} - Y_{p,t,s} = \sum_{r\in\mathbf{R}_p^+} X_{r,t,s} - \sum_{r\in\mathbf{R}_p^-} X_{r,t,s} \quad \forall p \in \mathbf{P}, t \in \mathbf{T} \neq 3, s \in \mathbf{S}$$

$$\sum_{r\in\mathbf{R}} X_{r,t,s} \leq 1 \quad \forall t \in \mathbf{T}, s \in \mathbf{S}$$

$$\sum_{p\in\mathbf{P}} Y_{p,t,s} = 1 \quad \forall t \in \mathbf{T}, s \in \mathbf{S}$$

$$Y_{A,0,s} = 1 \quad \forall s \in \mathbf{S}$$

$$Y_{3,D,s} = 1 \quad \forall s \in \mathbf{S}$$

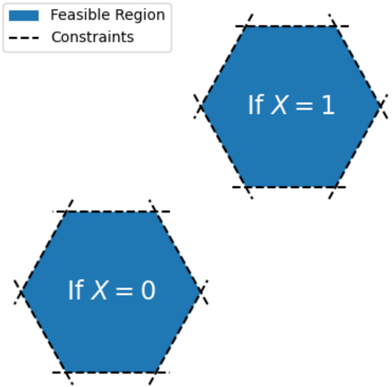
$$X_{r,0,s_1} = X_{r,0,s} \quad \forall r \in \mathbf{R}, s \in \mathbf{S} \neq s_1$$

$$Y_{p,0,s_1} = Y_{p,0,s} \quad \forall p \in \mathbf{P}, s \in \mathbf{S} \neq s_1$$

$$X_{r,t,s}, Y_{p,t,s} \in \{0,1\}$$

# Linearization

- Math Modeling only has basic algebra operators: $+, -, \times, \div, \sum, e^x, \log x, \sin x, etc.$
- Math Modeling does **not** directly support more advanced operators in the body of a model: *if*, *then*, *else*, $\min(a, b)$, $\max(a, b)$
- Remember, problems will solve MUCH faster if we can keep problems as linear as possible.
- We can apply some clever tricks to still model these behaviors.
- Thought exercise: How can we model the feasible region shown on the right?

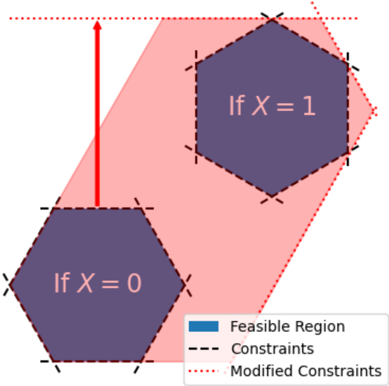


# "Big M" Constraints

- Main Idea: Include an extra term in any constraints that you want to de-activate when a binary variable is 1 (or 0). This extra term should make the constraint so loose that it'll never be binding.
- In general, you should just add a binary variable times some large parameter $\mathcal{M}$
- Example:

$$Y \leq 1$$

$$Y \leq 1 + \mathcal{M}X$$

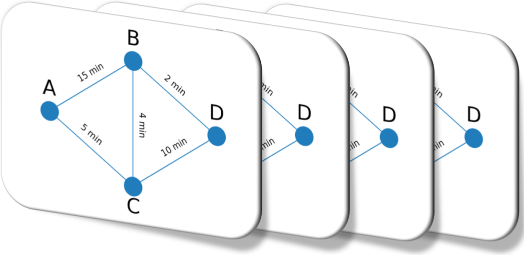



- NOTE OF CAUTION: Choosing a good $\mathcal{M}$ value is the name of the game here:
  - Too small: You'll be chopping off parts of the true feasible region. You're problem is likely to become infeasible.
  - Too big: The solver can get lost and can slow way down.
  - Too too big: Your computer can't handle huge numbers and (comparatively) tiny numbers at the same time. You're likely to encounter weird errors.

# Example: Big M Constraints

- Let's return to the Route Planning Problem (with Multiple Time Periods, but not Uncertainty)
- Let's say there was also an E-Z pass you could purchase for $12 that gave you a special discount on some, but not all, of the roads in the model.
- With the E-Z pass, here would be your $\delta^{EZ}_{r,t}$ values:

| $r$ | $t = 0$ | $t = 1$ | $t = 2$ | $t = 3$ |
|---|---|---|---|---|
| $AB$ | \$0 | \$0 | \$0 | \$0 |
| $AC$ | \$5 | \$5 | \$9 | \$10 |
| $BC$ | \$4 | \$10 | \$13 | \$13 |
| $BD$ | \$2 | \$2 | \$6 | \$5 |
| $CD$ | \$0 | \$0 | \$0 | \$0 |




- Should you buy the E-Z pass?
- How can you use Big M constraints to capture this decision?
  - Hint: Represent cost of traversing each path as it's own variable. Use two "if" statements and an intermediate variable to form this definition.

(Spend 5 minutes trying to figure it out)

# "If" Statements

- Normal "if" statements look like this:

$$A = \begin{cases} B & \textit{if } X \\ C & \textit{else} \end{cases}$$

- We can reformulate this like this:

$$A = A^B + A^C$$

$$A^B = \begin{cases} B & \textit{if } X \\ 0 & \textit{else} \end{cases} = BX$$

$$A^C = \begin{cases} C & \textit{if } (1-X) \\ 0 & \textit{else} \end{cases} = C(1-X)$$

- But remember! We want things to be linear!
- We can reformulate $A = BX$ using a series of Big M Constraints:

$$A \leq \beta^{MAX} X$$

$$A \leq B + \beta^{MIN}(X-1)$$

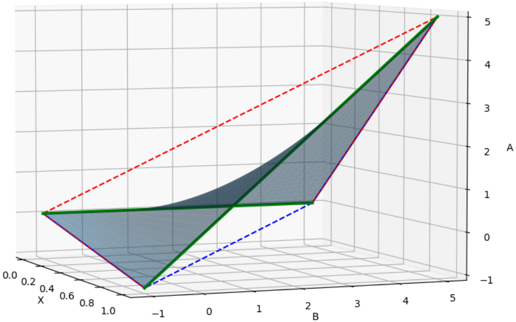
$$A \geq \beta^{MIN} X$$

$$A \geq B + \beta^{MAX}(X-1)$$

- Here, $\beta^{MAX}$ and $\beta^{MIN}$ are the maximum and minimum values that $B$ is capable of being. (Big Ms)
- If $C$ is not 0, repeat this process for C replacing $X$ with $1-X$.

# "If" Statement Caveats

Explanation of how to come up with the linearization of $A = BX$:




$$A \leq \beta^{MAX} X \qquad A \geq \beta^{MIN} X$$
$$A \leq B + \beta^{MIN}(X - 1) \qquad A \geq B + \beta^{MAX}(X - 1)$$

- Sometimes the objective you define indicates that $A$ will always be maximized or minimized.
- If you're certain that $A$ will ONLY ever be either maximized or minimized, you can drop two of the constraints.
  - Maximize $A$, drop "Greater Than" Constraints
  - Minimize $A$, drop "Less Than" Constraints
- Dropping unnecessary constraints means there are less things for the solver to consider: The solver will go faster.

# "Max" Statements

**IMPORTANT CONSIDERATION: Maximizing or Minimizing $A$**

If you will allways be minimizing $A = max(B, C)$,

- Model this as two individual inequalities (no binary variable needed!):

$$A \geq B$$

$$A \geq C$$

If you'll be maximizing $A = max(B, C)$ or you're not sure,

- You'll need to use a binary variable with and the Big M constraints shown on the right.

$$B - C \leq \mathcal{M}Y$$

$$C - B \leq \mathcal{M}(1 - Y)$$

$$A \geq B$$

$$A \geq C$$

$$A \leq B + \mathcal{M}(1 - Y)$$

$$A \leq C + \mathcal{M}Y$$

Here, $\mathcal{M}$ is the maximum possible difference between $B$ and $C(\max|B - C|)$ and $Y$ is a new binary variable.

# "Min" Statements

**IMPORTANT CONSIDERATION: Maximizing or Minimizing $A$**

If you will allways be maximizing $A = min(B, C)$,

- Model this as two individual inequalities (no binary variable needed!):

$$A \leq B$$

$$A \leq C$$

If you'll be minimizing $A = max(B, C)$ or you're not sure,

- You'll need to use a binary variable with and the Big M constraints shown on the right.

$$C - B \leq \mathcal{M}Y$$

$$B - C \leq \mathcal{M}(1 - Y)$$

$$A \leq B$$

$$A \leq C$$

$$A \geq B - \mathcal{M}(1 - Y)$$

$$A \geq C - \mathcal{M}Y$$

Here, $\mathcal{M}$ is the maximum possible difference between $B$ and $C(\max|B - C|)$ and $Y$ is a new binary variable.

# Next Class: Real life Problems

- Now that we have some powerful tools to capture real-world behaviors, we can jump into some real-world problems.
- I'd love to go over some problems you might be thinking of.
  - Do you have any requests?
- Problems I've already considered are:
  - How can I most effectively schedule tasks in my daily schedule?
  - How can I make a good choice about which insurance plan to select?
  - What's the most efficient way to transfer money between my bank accounts to pay my bills?