Final Project Report

Wanqi Yuan (wy21) & Jonathan Gan (wg18)

Tue., Apr. 30

# Introduction

## Problem Statement

The parking garage problem is currently structured to assume that demand is deterministic and constant. This is not representative of real world conditions as demand is highly dynamic and will change in response to various city conditions. For our final project, we want to better capture the variance in demand for parking, particularly in Houston. We will explore through research the relationship between price of parking and the demand of parking to implement into our get\_action function. We will mathematically represent this relationship through a equation derived from our research. We will be considering the case where we are taking yearly time steps to represent a garage that rents out spots on a yearly basis. At the end of each year, we will evaluate whether demand of that year given the elasticity of demand, exceeds capacity enough to justify building an additional level. We will analyze this case for multiple demand vs. price curve.

Demand is a quantifiable variable for many cases outside of parking that determine whether a given action will be taken. By more accurately modeling this relationship between how the price of *something* might change the demand of it, and therefore the construction that might take place as a result, we can apply this to other climate scenarios that would require sequential decision making. For example, the decision to expand a solar farm on a an annual basis dependent on the demand of the energy for the clients it serve.

## Selected Feature

Describe the feature you have selected to add to the existing decision-support tool. Discuss how this feature relates to the problem statement and its potential to improve climate risk assessment.

# Literature Review

Provide a brief overview of the theoretical background related to your chosen feature. Cite at least two relevant journal articles to support your approach (see [Quarto docs](https://quarto.org/docs/authoring/footnotes-and-citations.html) for help with citations). Explain how these articles contribute to the justification of your selected feature.

# Methodology

## Implementation

You should make your modifications in either the HouseElevation or ParkingGarage module. Detail the steps taken to implement the selected feature and integrate it into the decision-support tool. Include code snippets and explanations where necessary to clarify the implementation process.

## Validation

As we have seen in labs, mistakes are inevitable and can lead to misleading results. To minimize the risk of errors making their way into final results, it is essential to validate the implemented feature. Describe the validation techniques used to ensure the accuracy and reliability of your implemented feature. Discuss any challenges faced during the validation process and how they were addressed.

# Results

Present the results obtained from the enhanced decision-support tool. Use tables, figures, and visualizations to clearly communicate the outcomes. Provide sufficient detail to demonstrate how the implemented feature addresses the problem statement. Use the #| output: false and/or #| echo: false tags to hide code output and code cells in the final report except where showing the output (e.g.g, a plot) or the code (e.g., how you are sampling SOWs) adds value to the discussion. You may have multiple subsections of results, which you can create using ##.

# Conclusions

## Discussion

Analyze the implications of your results for climate risk management. Consider the context of the class themes and discuss how your findings contribute to the understanding of climate risk assessment. Identify any limitations of your approach and suggest potential improvements for future work.

## Conclusions

Summarize the key findings of your project and reiterate the significance of your implemented feature in addressing the problem statement. Discuss the broader implications of your work for climate risk management and the potential for further research in this area.

# References

using Plots  
using Distributions  
Plots.default(; margin=5Plots.mm)

function get\_profit(capacity, m, b, cost)  
  
 # Cost is cost per spot   
 price = div(capacity - b, m) # Floor Div for dollars  
 return (price \* capacity)-(cost \* capacity)  
end

get\_profit (generic function with 1 method)

function optimize(maxSpots, increment, start\_spots, m, cost\_per\_spot)  
 b = maxSpots  
 levels\_to\_add = div(maxSpots - start\_spots, increment)  
 optLevels = 0   
 optProfit = get\_profit(start\_spots, m, b, cost\_per\_spot)  
 curr\_levels = 0  
 while curr\_levels < levels\_to\_add  
 curr\_capacity = start\_spots + curr\_levels \* increment  
 profit = get\_profit(curr\_capacity, m, b, cost\_per\_spot)  
 if profit > optProfit  
 optProfit = profit  
 optLevels = curr\_levels  
 end  
 curr\_levels += 1  
 end  
 return optLevels  
end

optimize (generic function with 1 method)

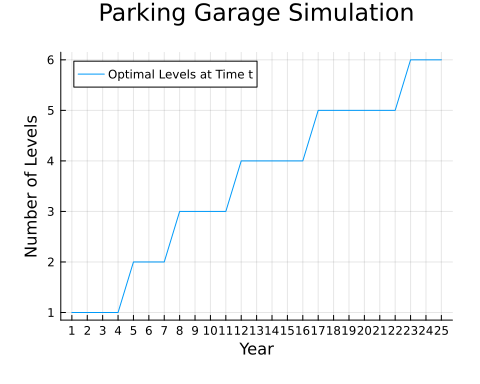
function simulation(timesteps, start\_spots, increment, demand\_growth\_rate, init\_b, m, inflation\_rate, cost\_per\_spot)  
 y\_levels = [] # LEVELS AT TIMETSTEP T  
 y\_curr\_profit = [] # PROFIT AT TIMESTEP T  
 y\_net\_profit = [] # NET PROFITS  
 net\_profit = 0  
 curr\_levels = 0  
 timestep = 0  
 curr\_spots = start\_spots  
 b = init\_b  
  
 while timestep < timesteps  
 levels\_to\_add = optimize(b, increment, curr\_spots, m, cost\_per\_spot)  
 if levels\_to\_add == 0  
 # println("Whoops")  
 end  
 curr\_spots += increment \* levels\_to\_add  
 curr\_levels += levels\_to\_add  
  
 curr\_profit = get\_profit(curr\_spots, m, b, cost\_per\_spot) / 1000  
 net\_profit += curr\_profit  
 push!(y\_curr\_profit, curr\_profit)  
 push!(y\_net\_profit, net\_profit)  
 push!(y\_levels, curr\_levels)  
  
 b \*= (1 + demand\_growth\_rate)  
 m /= (1 + inflation\_rate)  
  
 timestep += 1   
 end  
  
 return y\_levels  
end

simulation (generic function with 1 method)

INCREMENT = 100 Inflation Rate = 0.04 Construction Cost = 30000

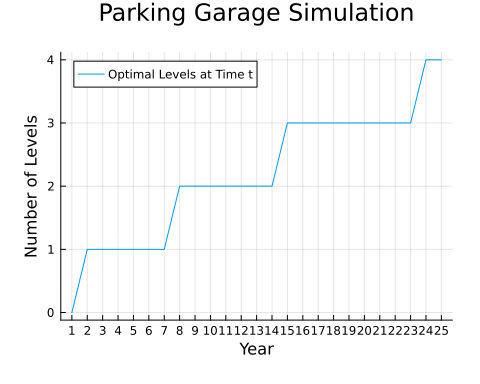
timesteps = 25   
start\_spots = 0  
increment = 100  
demand\_growth\_rate = rand(Normal(0.0045, 0.001))  
init\_b = 1500  
m = -0.046  
inflation\_rate = 0.04  
cost\_per\_spot = 30000  
y\_levels = simulation(timesteps, start\_spots, increment, demand\_growth\_rate, init\_b, m, inflation\_rate, cost\_per\_spot)  
display\_growth\_rate = round(demand\_growth\_rate, sigdigits=3)  
plot\_title = "Parking Garage Simulation"  
font\_size = 16  
println(y\_levels)  
plot(y\_levels, label="Optimal Levels at Time t", xlabel="Year", ylabel="Number of Levels",  
title=plot\_title, titlefontsize=font\_size, lw=1, gridlinewidth=1, xticks=0:1:25, yticks=0:1:100)  
# plot(y\_curr\_profit, label="Current Annual Profit at time t")  
# plot(y\_net\_profit, label="Net Profit at time t")  
# legend()  
#display()

Any[1, 1, 1, 1, 2, 2, 2, 3, 3, 3, 3, 4, 4, 4, 4, 4, 5, 5, 5, 5, 5, 5, 6, 6, 6]



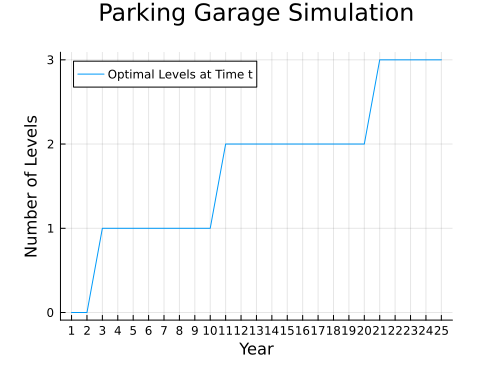
INCREMENT = 150 Inflation Rate = 0.04 Construction Cost = 30000

Any[0, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3, 3, 3, 4, 4]



INCREMENT = 200 Inflation Rate = 0.04 Construction Cost = 30000

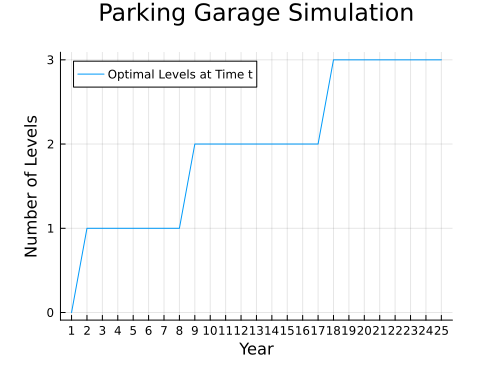
Any[0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3]



Increment = 150 INFLATION RATE = 0.03 Construction Cost = 30000

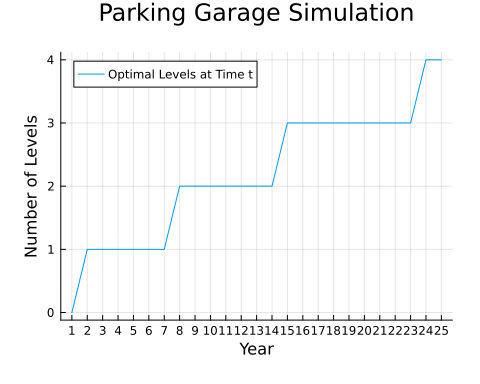
timesteps = 25   
start\_spots = 0  
increment = 150  
demand\_growth\_rate = rand(Normal(0.0045, 0.001))  
init\_b = 1500  
m = -0.046  
inflation\_rate = 0.03  
cost\_per\_spot = 30000  
y\_levels = simulation(timesteps, start\_spots, increment, demand\_growth\_rate, init\_b, m, inflation\_rate, cost\_per\_spot)  
display\_growth\_rate = round(demand\_growth\_rate, sigdigits=3)  
plot\_title = "Parking Garage Simulation"  
font\_size = 16  
println(y\_levels)  
plot(y\_levels, label="Optimal Levels at Time t", xlabel="Year", ylabel="Number of Levels",  
title=plot\_title, titlefontsize=font\_size, lw=1, gridlinewidth=1, xticks=0:1:25, yticks=0:1:100)  
# plot(y\_curr\_profit, label="Current Annual Profit at time t")  
# plot(y\_net\_profit, label="Net Profit at time t")  
# legend()  
#display()

Any[0, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3, 3]



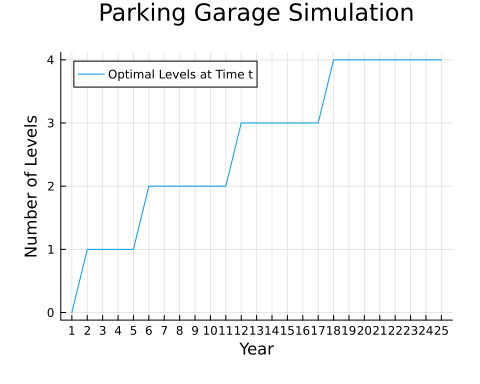
Increment = 150 INFLATION RATE = 0.04 Construction Cost = 30000

Any[0, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3, 3, 3, 4, 4]



Increment = 150 INFLATION RATE = 0.05 Construction Cost = 30000

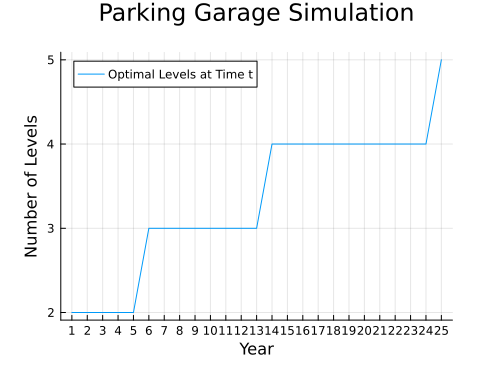
Any[0, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 4, 4, 4]



Increment = 150 Inflation Rate = 0.04 CONSTRUCTION COST = 20000

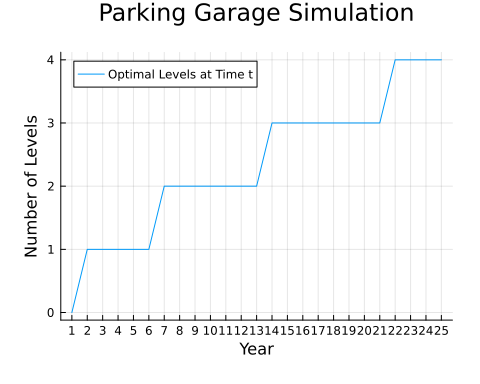
timesteps = 25   
start\_spots = 0  
increment = 150  
demand\_growth\_rate = rand(Normal(0.0045, 0.001))  
init\_b = 1500  
m = -0.046  
inflation\_rate = 0.04  
cost\_per\_spot = 20000  
y\_levels = simulation(timesteps, start\_spots, increment, demand\_growth\_rate, init\_b, m, inflation\_rate, cost\_per\_spot)  
display\_growth\_rate = round(demand\_growth\_rate, sigdigits=3)  
plot\_title = "Parking Garage Simulation"  
font\_size = 16  
println(y\_levels)  
plot(y\_levels, label="Optimal Levels at Time t", xlabel="Year", ylabel="Number of Levels",  
title=plot\_title, titlefontsize=font\_size, lw=1, gridlinewidth=1, xticks=0:1:25, yticks=0:1:100)  
# plot(y\_curr\_profit, label="Current Annual Profit at time t")  
# plot(y\_net\_profit, label="Net Profit at time t")  
# legend()  
#display()

Any[2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 5]



Increment = 150 Inflation Rate = 0.04 CONSTRUCTION COST = 30000

Any[0, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3, 3, 4, 4, 4, 4]



Increment = 150 Inflation Rate = 0.04 CONSTRUCTION COST = 40000

Any[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3]

