Attendance survey:

https://forms.gle/qQ8TrExBUsyxtxH88

Optimization with julia JuMP

MIT MANAGEMENT SLOAN SCHOOL

OPERATIONS RESEARCH CENTER

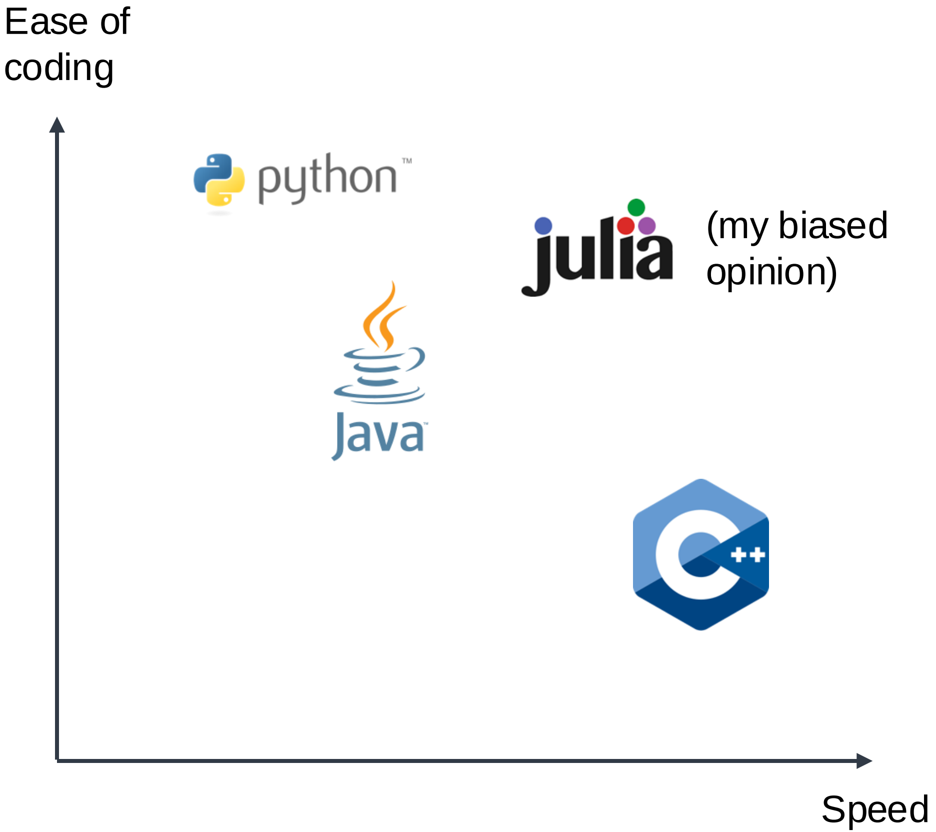
15.003 Analytics Tools

Sean Lo (seanlo@mit.edu)
Yu Ma (midsumer@mit.edu)

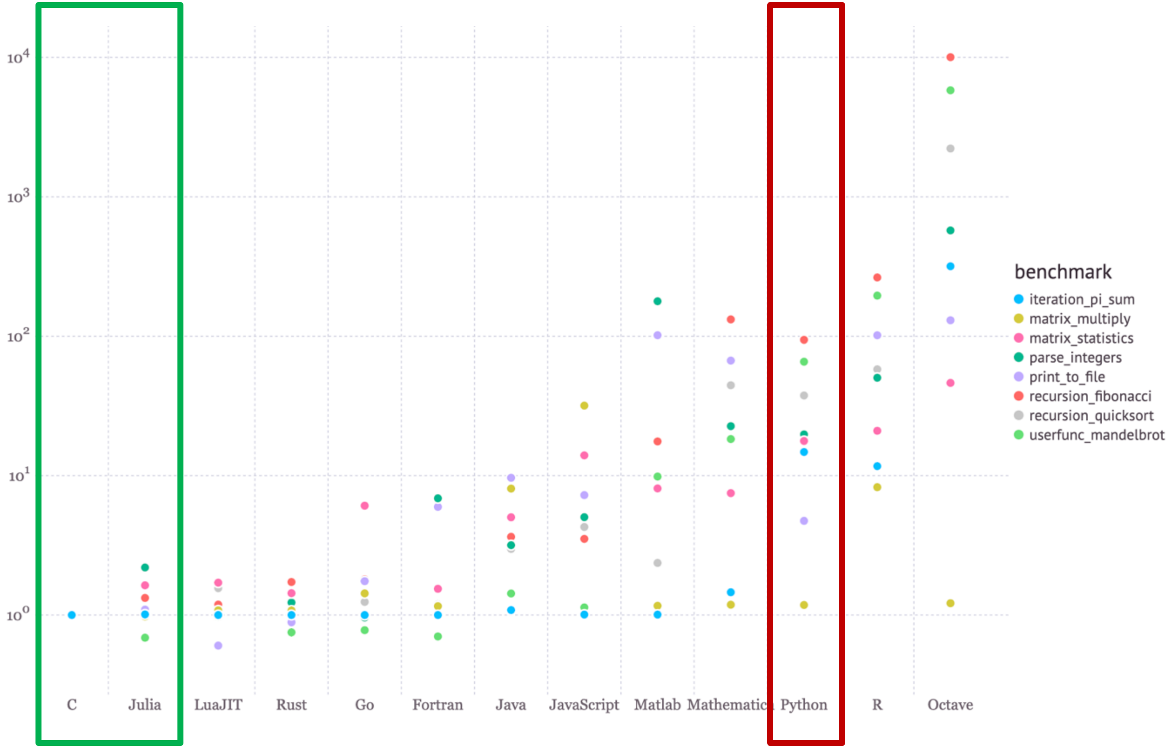
Introduction: What is Julia?

A **fast** and **easy-to-use** programming language, tailored for scientific computing:

- Developed in MIT in 2012
- “Python-like” syntax and “C-like” speed



Julia is fast



Julia is nearly as fast as C, while maintaining the flexibility of a high-level language!

What makes Julia special?

- **High-level** language: abstracts away e.g. memory usage
- **Dynamic** language: “reads like English”, can be used interactively, good for prototyping
- **High-performance**: vanilla Julia can be very fast (unlike Python)
- **Multiple dispatch**: “defines function behavior across combinations of argument types”
- Growing **package ecosystem**:
  - **Statistics**: StatsBase.jl and Statistics.jl
  - **Machine learning**: MLJ.jl, and Flux.jl for deep learning
  - **Data tools**: DataFrames.jl, CSV.jl, Arrow.jl and Spark.jl for big data
  - **Data visualization**: Plots.jl, Makie.jl
  - **Optimization**: JuMP.jl
  - **Differential equations**: DifferentialEquations.jl

Introduction: What is JuMP?

A modelling language built on Julia, “supporting packages for mathematical optimization”

- Unified **interface** for different solvers

Programming language:

Modelling language:

Optimization solvers:

Agenda

- Part 1: 9:00am – 10:20am: Introduction to Julia (80 mins)
- Break: 10:20am – 10:40am
- Part 2: 10:40am – 12:00pm: Introduction to Optimization (80 mins)

Part 1: Introduction to Julia

Break

Resume at 10:40

Attendance survey:

https://forms.gle/qQ8TrExBUsyxtxH88

Part 2: Introduction to JuMP

Introduction

- Overview of optimization
- How to formulate optimization problems
- Model implementation in Julia/JuMP
- Detour: introduction to Julia
- Model interpretation and takeaways

What is Optimization?

Optimization is a field of mathematics that involves finding the **minimum** or the **maximum** of a function under constraints.

- Most **problems** in real life where there is **uncertainty** can be modeled under an optimization framework.
- **Why**? When making decisions, we aim to achieve certain objectives while following some requirements.
- Optimization allows us to **formulate these problems mathematically** and take a **data-driven approach** to decision-making**.**

What is Optimization?

Optimization is a field of mathematics that involves finding the **minimum** or the **maximum** of a function under constraints.

- Most **problems** in real life where there is **uncertainty** can be modeled under an optimization framework.
- **Why**? When making decisions, we aim to achieve certain objectives while following some requirements.
- Optimization allows us to **formulate these problems mathematically** and take a **data-driven approach** to decision-making**.**

Examples:

"How should we schedule games to:

… maximize primetime viewership?

… minimize team travel distances?

… maximize rivalries at end of season?"

"Who should I include on my team roster to maximize team performance?"

Objectives, decisions, and constraints

Let’s break down one of the previous statements:

“Who should I include on my team roster to maximize team performance?”

Decision Variables	What are the decisions that we are trying to make?	For each candidate player, decide whether or not to include them on the team.

Objectives, decisions, and constraints

Let’s break down one of the previous statements:

“Who should I include on my team roster to maximize team performance?”

Decision Variables	What are the decisions that we are trying to make?	For each candidate player, decide whether or not to include them on the team.
Problem Objective	What objective do we want to maximize (or minimize), as a function of our decision variables?	Maximize team performance.

Objectives, decisions, and constraints

Let's break down one of the previous statements:

"Who should I include on my team roster to maximize team performance?"

Decision Variables	What are the decisions that we are trying to make?	For each candidate player, decide whether or not to include them on the team.
Problem Objective	What objective do we want to maximize (or minimize), as a function of our decision variables?	Maximize team performance.
Constraints	What are the logical constraints that must be obeyed in any proposed solution?	Team must obey size restriction and have a sufficient mix of positions.

Objectives, decisions, and constraints

Mixed-integer optimization (MIO) is a specific type of optimization that requires certain mathematical structure:

Decision Variables	What are the decisions that we are trying to make?	For each candidate player, decide whether or not to include them on the team.
Problem Objective	What objective do we want to maximize (or minimize), as a function of our decision variables?	
Constraints	What are the logical constraints that must be obeyed in any proposed solution?	

Objectives, decisions, and constraints

Mixed-integer optimization (MIO) is a specific type of optimization that requires certain mathematical structure:

Decision Variables	What are the decisions that we are trying to make?	Variables can be **binary** (0,1), **integer** (0,1,2,3,…), or **continuous** (decimals).
Problem Objective	What objective do we want to maximize (or minimize), as a function of our decision variables?	
Constraints	What are the logical constraints that must be obeyed in any proposed solution?	

Objectives, decisions, and constraints

Mixed-integer optimization (MIO) is a specific type of optimization that requires certain mathematical structure:

Decision Variables	What are the decisions that we are trying to make?	Variables can be **binary** (0,1), **integer** (0,1,2,3,…), or **continuous** (decimals).
Problem Objective	What objective do we want to maximize (or minimize), as a function of our decision variables?	Maximize team performance.
Constraints	What are the logical constraints that must be obeyed in any proposed solution?	

Objectives, decisions, and constraints

Mixed-integer optimization (MIO) is a specific type of optimization that requires certain mathematical structure:

Decision Variables	What are the decisions that we are trying to make?	Variables can be **binary** (0,1), **integer** (0,1,2,3,…), or **continuous** (decimals).
Problem Objective	What objective do we want to maximize (or minimize), as a function of our decision variables?	The objective must be a **linear function** of the variables. e.g. $\max x + y$ vs. $\max x * y$
Constraints	What are the logical constraints that must be obeyed in any proposed solution?	

Objectives, decisions, and constraints

Mixed-integer optimization (MIO) is a specific type of optimization that requires certain mathematical structure:

Decision Variables	What are the decisions that we are trying to make?	Variables can be **binary** (0,1), **integer** (0,1,2,3,…), or **continuous** (decimals).
Problem Objective	What objective do we want to maximize (or minimize), as a function of our decision variables?	The objective must be a **linear function** of the variables. e.g. $\max x + y$ vs. $\max x * y$
Constraints	What are the logical constraints that must be obeyed in any proposed solution?	Team must obey size restriction and have a sufficient mix of positions.

Objectives, decisions, and constraints

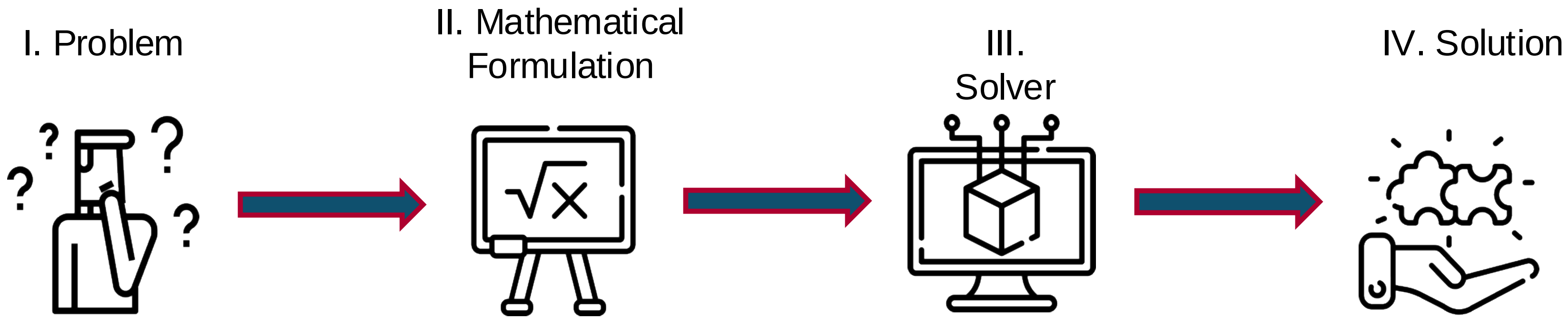
Mixed-integer optimization (MIO) is a specific type of optimization that requires certain mathematical structure:

Decision Variables	What are the decisions that we are trying to make?	Variables can be **binary** (0,1), **integer** (0,1,2,3,…), or **continuous** (decimals).
Problem Objective	What objective do we want to maximize (or minimize), as a function of our decision variables?	The objective must be a **linear function** of the variables. e.g. $\max x + y$ vs. $\max x * y$
Constraints	What are the logical constraints that must be obeyed in any proposed solution?	Constraints must be **linear functions** of the variables.

How do we solve these problems?

We use **solvers:**

- They are software packages;
- They incorporate one or more algorithms;
- They can find solutions to one or more types of problems.



Recent Speedups in MIO

- Many general purpose solvers are available:
  CPLEX, Gurobi, GLPK, COIN-OR
- In the past 30 years, the speed of integer optimization solvers has increased by a factor of **3 trillion times**!
- This doesn't include increasing speed of computers.

A problem that can be solved in 1 second today took 71000 years to solve 30 years ago!

Case Study: Constructing the USA Basketball Dream Team

Constructing an Olympic Team

USA Basketball must decide who to put on the 12-man roster for the 2021 Summer Olympics.

TOKYO 2020

We want to select a **high-performing team** that meets **Olympic regulations** and **desired constraints on team composition.**

We want to select a **high-performing team** that meets **Olympic regulations** and **desired constraints on team composition.**

How can we formalize this optimization problem as a mathematical model?

I. Defining the Decision Variables

Decision Variables	What are the decisions that we are trying to make?	For each candidate player, decide whether or not to include them on the team.

I. Defining the Decision Variables

Decision Variables	What are the decisions that we are trying to make?	For each candidate player, decide whether or not to include them on the team.

Suppose we have N=200 players who we are considering for our team. For each player:

$$\boldsymbol{x_i} = \begin{cases} 1 \text{ if player i is selected for the team} \\ 0 \text{ otherwise} \end{cases}$$

For example:

If $x_i = 1$ for Steph Curry, he has been selected for the team.

If $x_i = 0$ for Draymond Green, he has not been selected for the team.

II. Formalizing the Objective

Problem Objective	What objective do we want to maximize (or minimize), as a function of our decision variables?	Maximize team performance.

II. Formalizing the Objective

Problem Objective	What objective do we want to maximize (or minimize), as a function of our decision variables?	Maximize team performance.

Let s_i = performance rating for player i (e.g. player i's PER)

$$\max \sum_{i=1}^{200} s_i * x_i$$

III. Constructing the Constraints

Constraints	What are the logical constraints that must be obeyed in any proposed solution?	Team must <u>obey size restriction</u> and have a sufficient mix of positions.

III. Constructing the Constraints

Constraints	What are the logical constraints that must be obeyed in any proposed solution?	Team must obey size restriction and have a sufficient mix of positions.

Team Size: We must select a team of exactly 12 players.

$$\sum_{i=1}^{200} x_i = 12$$

III. Constructing the Constraints

Constraints	What are the logical constraints that must be obeyed in any proposed solution?	Team must obey size restriction and have a sufficient mix of positions.

For the 12-man roster, we want at least G guards, F forwards, and C centers.

Guards: Let g_i=1 if player i is a guard (else 0).

Forwards: Let f_i=1 if player i is a guard (else 0).

Centers: Let c_i=1 if player i is a guard (else 0).

III. Constructing the Constraints

Constraints	What are the logical constraints that must be obeyed in any proposed solution?	Team must obey size restriction and have a sufficient mix of positions.

For the 12-man roster, we want at least G guards, F forwards, and C centers.

Guards: Let g_i=1 if player i is a guard (else 0). $\sum_{i=1}^{200} g_i x_i \geq G$

Forwards: Let f_i=1 if player i is a guard (else 0). $\sum_{i=1}^{200} f_i x_i \geq F$

Centers: Let c_i=1 if player i is a guard (else 0). $\sum_{i=1}^{200} c_i x_i \geq C$

MIO offers a flexible decision framework

Incorporating new constraints:

- *NBA Team Diversity*: Ensure that at least 5 distinct teams are represented on the roster.
- *Injury Adjustments*: If a player becomes injured, we can force his decision variable to be 0 in the constraints.
- *Team capabilities*: Beyond DBPM, we could have added constraints on average value for other metrics. We could also constrain the minimum value, such as saying that we require players to have at least a DBPM of 0 to be considered for the team.

MIO offers a flexible decision framework

Incorporating new constraints:

- *NBA Team Diversity*: Ensure that at least 5 distinct teams are represented on the roster.
- *Injury Adjustments*: If a player becomes injured, we can force his decision variable to be 0 in the constraints.
- *Team capabilities*: Beyond DBPM, we could have added constraints on average value for other metrics. We could also constrain the minimum value, such as saying that we require players to have at least a DBPM of 0 to be considered for the team.

Considering other objectives:

- On a professional sports team, ownership wants to maximize performance while also controlling costs. We could add a budget term to the objective (to be balanced with performance).

MIO offers a flexible decision framework

Incorporating new constraints:

- *NBA Team Diversity*: Ensure that at least 5 distinct teams are represented on the roster.
- *Injury Adjustments*: If a player becomes injured, we can force his decision variable to be 0 in the constraints.
- *Team capabilities*: Beyond DBPM, we could have added constraints on average value for other metrics. We could also constrain the minimum value, such as saying that we require players to have at least a DBPM of 0 to be considered for the team.

Considering other objectives:

- On a professional sports team, ownership wants to maximize performance while also controlling costs. We could add a budget term to the objective (to be balanced with performance).

Capturing interactions between players:

- Certain people play better when in a pair than on their own, so we may want to add constraints that the players must always appear together (or conversely, cannot both be on the team).

Main Takeaways

- **Mixed-Integer Optimization (MIO)** is a powerful tool for modeling real-world decision problems and offering objective solutions.
- The biggest challenge of MIO is **defining and quantifying the problem**.
- Once a problem is formalized, Julia offers an **easy-to-use interface** that allows us to quickly solve large problems.
- MIO has a **flexible framework** that allows us to easily experiment with different objectives, constraints, and model parameters.

Further reading

- Resources for learning Julia (I recommend the Manual)
- Julia’s performance tips, workflow tips and style guide
- A comprehensive list of differences between Julia and Python (and other languages)
- A talk (35min) explaining what *multiple dispatch* is in Julia and why it works
- https://jump.dev/JuMP.jl/stable/installation/#Supported-solvers

Thank you!