JuMP Developers Workshop


Speakers

Chris Coey, MIT • Carleton Coffrin, LANL • Steven Diamond, Stanford • Joaquim Dias Garcia, PSR & PUC-Rio • Oscar Dowson, U. of Auckland • Joey Huchette, MIT • Jordan Jalving, UW-Madison • Benoît Legat, UCL • Miles Lubin, MIT • Yee Sian Ng, MIT • Jarrett Revels, MIT • Nestor Sepulveda, MIT • Bartolomeo Stellato, U. of Oxford • Juan Pablo Vielma, MIT

Sponsored by:

www.juliaopt.org/developersmeetup
Welcome and Thanks!

- Organizing committee:

  Jennifer Challis
  Miles Lubin
  Chris Coey
  Joey Huchette
Welcome and Thanks!

- Sponsor:

MIT MANAGEMENT
LATIN AMERICA OFFICE
Welcome and Thanks!

• Speakers (besides Chris, Joey and Miles):
  – Carleton Coffrin, Los Alamos National Lab
  – Steven Diamond, Stanford
  – Joaquim Dias Garcia, PSR-Inc. & PUC-Rio
  – Oscar Dowson, University of Auckland
  – Jordan Jalving, University of Wisconsin-Madison
  – Benoît Legat, Université Catholique de Louvain
  – Yee Sian Ng, MIT
  – Jarrett Revels, MIT
  – Nestor Sepulveda, MIT
  – Bartolomeo Stellato, University of Oxford
Welcome and Thanks!

- Audience and Julia/JuMP community:


(My) Quick Intro to

JuMP & JuliaOpt
20th Century Optimization Modelling Languages

- User-friendly algebraic modelling languages (AML):
  - GAMS
  - AMPL
  - PYOMO
  - CVX

  Standalone and Fast

  Based on General Language and Versatile, but Slow

- Commercial and Open-Source C/C++ frameworks:
  - COIN-OR
  - SCIP
  - GUROBI

  Fast and Versatile, but complicated (and some proprietary)
21st Century Optimization Modelling Language

Greedy Wish List:

• Modern, modular, easy to embed...
  – Within a simulation, interactive visualization, etc.
• Interact with solvers while they are running
• Easy to extend to specialized problem classes
JuliaOpt’s packages can be loosely grouped into two sets. The first set are standalone Julia packages:

### Overview of Packages

#### Getting Started

For optimization software.

- **What is Julia?**
- **What is JuliaOpt?**
- **Large Software Stack and Vibrant Community**

#### Julia:

1. You can find downloads and installation instructions for Julia on the Julia site.
2. Open-source solvers will automatically be downloaded and installed in your Julia environment.
3. Commercial solvers will need to be downloaded and installed separately.

#### Examples

- Controlling a rocket with JuMP
- Solving Sudoku puzzles with JuMP

#### Mailing list

- julia-opt (https://groups.google.com/forum/#!forum/julia-opt)

#### Code

- Github (https://github.com/JuliaOpt)

#### Workshop

- JuliaCon 2015 workshop (https://www.youtube.com/watch?v=nnL7yLMVu6c)

#### Workshops

- INFORMS Journal on Computing
- INFORMS Journal on Computing (http://dx.doi.org/10.1287/ijoc.2014.0623).
- Overview of Optimization Software (http://julialang.org/)

#### Documentation

- [PDF](http://arxiv.org/abs/1312.1431)

### Julia Optimization Packages

<table>
<thead>
<tr>
<th>MathProgBase.jl</th>
<th>Convex.jl</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cbc.jl</td>
<td>CPLEX.jl</td>
</tr>
<tr>
<td>ECOS.jl</td>
<td>Gurobi.jl</td>
</tr>
<tr>
<td>Ipopt.jl</td>
<td>Mosek.jl</td>
</tr>
<tr>
<td>NLopt.jl</td>
<td>SCS.jl</td>
</tr>
<tr>
<td>JuMP</td>
<td></td>
</tr>
</tbody>
</table>

- [Pkg.add](https://julialang.org) command from inside Julia
- [Pkg.update()](https://julialang.org) to get latest package info

**Diagram:**

- JuliaOpt logos
- Graph of Julia Optimization packages
- Picture of workshop participants
- Picture of workshop brochure
- Picture of workshop flyer

**Contact:**

- [julia-opt](https://groups.google.com/forum/#!forum/julia-opt)
- [IJulia.jl](https://github.com/JuliaLang/IJulia.jl)
- [Jupyter.org](https://jupyter.org)

**Resources:**

- [Jupyter.org](https://jupyter.org)
- [IJulia.jl](https://github.com/JuliaLang/IJulia.jl) notebooks, including:
  - Controlling a rocket with JuMP
  - Solving Sudoku puzzles with JuMP
  - Many more (https://github.com/JuliaOpt/juliaopt-notebooks)

---

**Image Credits:**

- JuliaCon 2015 workshop (https://www.youtube.com/watch?v=nnL7yLMVu6c)
- INFORMS Journal on Computing (http://dx.doi.org/10.1287/ijoc.2014.0623)
Sustained Growth and Some Milestones

- **Version 0.2**: Solver callbacks (December, 2013)
- **Version 0.5**: Nonlinear optimization (May, 2014)
- **Version 0.10**: Semidefinite optimization (August, 2015)
- **Version 0.12**: Rewrote nonlinear optimization (February, 2016)
- **Version 0.13**: Renamed everything from camelCase (April, 2016)
- **Version 0.15**: `sum{}` becomes `sum()` (December, 2016)

Hits to JuMP's documentation
JuliaOpt in University Courses
Serious Use of JuliaOpt

Optimal hybrid sequencing and assembly: Feasibility conditions for accurate genome reconstruction and cost minimization strategy

Chun-Chi Chen, Noushin Ghaafari, Xiaoning Qian, Byung-Jun Yoon

Simultaneous identification of specifically interacting paralogs and interprotein contacts by direct coupling analysis

Thomas Gueudré, Carlo Baldassi, Marco Zamparo, Martin Weigt, Andrea Pagnani

Inversion methods for fast-ion velocity-space tomography in fusion plasmas

Big 2's and Big 3's: Analyzing How a Team's Best Players Complement Each Other

Robert Ayer
MBA 2011
Massachusetts Institute of Technology
Cambridge, MA
Email: robertayer@gmail.com

Abstract

One of the most important aspects of team construction is identifying and acquiring the most talented and productive players on your team, the players on whom a team's fortunes most rely. Teams must decide which player-types, when combined, yield the best fit. As an example, suppose there is a team, whose current best player is a scoring, shoot-first point guard. Suppose this team is looking to bring in a top-flight free agent. What type of player should this team target? Should they bring in a defense-oriented big man? Should they acquire a multi-faceted, jack-of-all-trades wing? This paper aims to answer these questions. Analyzing player data and team season data from 1977, this paper first uses clustering techniques to group players into appropriate groups, then regression to determine the degree to which the composition of a team's top 2 and top 3 players affect that team's win total, while accounting for team quality and coaching ability. This paper shows that the composition of a team's top 2 and top 3 players is a strongly statistically significant factor in the success of a team, and shows which combinations yield over-performance, and which combinations yield underperformance, relative to the team's talent and coaching quality.

1 Introduction

There have been many instances in basketball where a team, with perhaps a collection of new acquisitions, underperforms relative to the perceived talent on the team. Most observers will intuitively conclude that this team, while talented, just doesn't fit well together. Conversely, there are also many instances where a team, with perhaps relatively modest top level talent, exceeds expectations. This team, most will conclude, is put together well, i.e., the players complement each other, they "fit." Similarly, at the player level, there are many cases in which a new player on a team, perhaps acquired through free agency, who, though talented, fails to live up to expectations. This can be attributed to many things: lack of effort, erosion of skills, poor scouting. However, many times a player's underperformance is attributed to a poor fit with the team. This is to say, that it is not enough for a player to have valuable basketball skills to fully reach his potential; the player must also be on a team which is constructed in a way that is complementary to those skills.

This paper aims to provide some insight into team construction and player fit by analyzing combinations of player types (specifically, 2 and 3 man combinations, known commonly as "Big 2's or Big 3's), and determining which combinations lead to increased wins, while accounting for talent level and coaching skill. Along the way, the research uncovered some interesting insights regarding coaching ability; while those insights will be addressed briefly, this is not a focus of the paper.

Picking Winners Using Integer Programming

David Scott Hunter, Juan Pablo Vielma, Tauhid Zaman
arXiv:1604.01455,
https://github.com/dscotthunter/Fantasy-Hockey-IP-Code
News
2016 ICS Prize for JuMP

“JuMP’s design leverages advanced features of the Julia language to offer distinctive functionality while achieving performance in instance creation often similar to commercial modeling tools.”

Congratulations Dr. Lubin!

http://youtu.be/Vifryc-Iqao

Mixed-integer convex optimization: outer approximation algorithms and modeling power

by

Miles Lubin

B.S., The University of Chicago (2011)
M.S., The University of Chicago (2011)

Submitted to the Sloan School of Management
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2017

© Massachusetts Institute of Technology 2017. All rights reserved.
Welcome Jarrett!

Jarrett Reveles
jrevels

Follow

Block or report user

MIT
Cambridge, MA
jarrettreveles@gmail.com

JuMP
“Sponsored Research Technical Staff”

Overview    Repositories 22    Stars 29    Followers 53    Following 21

Pinned repositories

JuliaDiff/ReverseDiff.jl
Reverse Mode Automatic Differentiation for Julia
Julia ⭐ 41 ⬇ 12

JuliaDiff/ForwardDiff.jl
Forward Mode Automatic Differentiation for Julia
Julia ⭐ 109 ⬇ 32

JuliaCI/BenchmarkTools.jl
A benchmarking framework for the Julia language
Julia ⭐ 59 ⬇ 8

JuliaCI/Nanosoldier.jl
A package for running JuliaCI services on MIT’s Nanosoldier cluster
Julia ⭐ 5 ⬇ 4

JuliaWeb/GitHub.jl
A Julia package for interfacing with GitHub
Julia ⭐ 47 ⬇ 25
Sustainability Plan for JuMP (and Julia)

Remain (Primarily) Open-Source
More JuMP in Latin America

MIT MANAGEMENT LATIN AMERICA OFFICE

Santiago in August

@J_P_Vielma