



Model Validation and Open Questions

Validating Optimization Problems via Generative AI

Connor Lawless, Stanford University

AAAI 2026 | January 20th, 2025

Can we leverage LLMs to help understand and validate optimization models?

Tutorial Gameplan

1. Background
2. Model Formulation

Coffee Break

3. Model Solving
4. **Model Validation**
 - a. Identifying Infeasible Subsystems
 - b. General Q&A
5. Open Questions



Diagnosing Infeasible Optimization Problems Using Large Language Models

INFOR 2025

Hao Chen, Gonzalo E. Constante-Flores, Can Li

Irreducible Infeasible Subset (IIS)

Many optimization problems are infeasible... but it's hard to diagnose why!

Irreducible Infeasible Subset (IIS) is a tool to characterizing infeasible models.

A subset of an infeasible optimization model is an IIS if it is:

(i) **Infeasible**: The subset has no feasible solutions

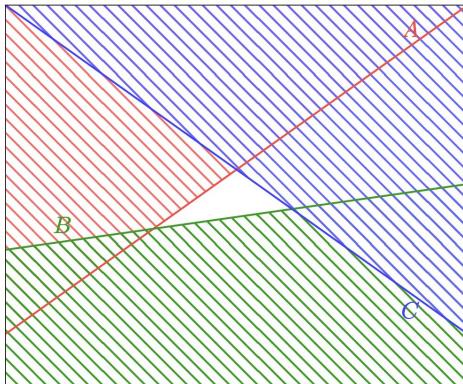
(ii) **Irreducible**: Any proper subset of the IIS is feasible.

Irreducible Infeasible Subset (IIS)

Many optimization problems are infeasible... but it's hard to diagnose why!

Irreducible Infeasible Subset (IIS) is a tool to characterizing infeasible models.

Example: Consider the following subset – is it an IIS?



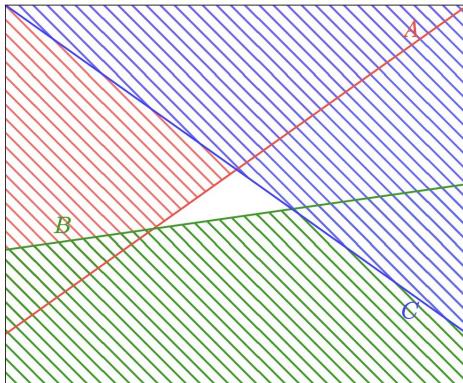
System of
3 Constraints

Irreducible Infeasible Subset (IIS)

Many optimization problems are infeasible... but it's hard to diagnose why!

Irreducible Infeasible Subset (IIS) is a tool to characterizing infeasible models.

Example: Consider the following subset – is it an IIS?



System of
3 Constraints

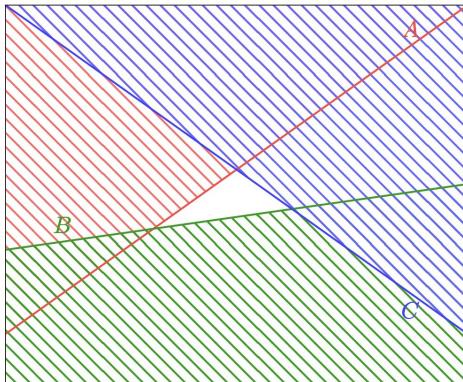
Infeasible? Yes!

Irreducible Infeasible Subset (IIS)

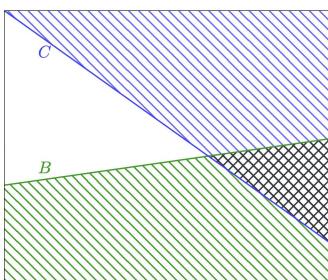
Many optimization problems are infeasible... but it's hard to diagnose why!

Irreducible Infeasible Subset (IIS) is a tool to characterizing infeasible models.

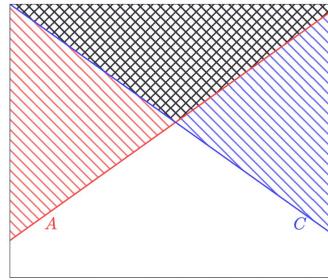
Example: Consider the following subset – is it an IIS?



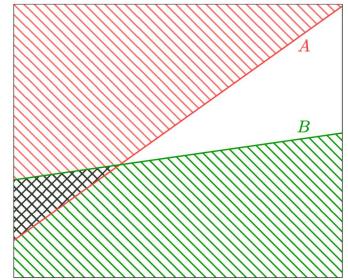
System of
3 Constraints



No A



No B



No C

Irreducible? Yes!

Key Challenge: Need optimization expertise to find and correct these infeasibilities!

OptiChat

Key Idea: Leverage LLMs as a natural language interface to correct infeasible models.



User

Specific Questions about Model

What is the most likely conflict among these constraints that you believe is causing the infeasibility?



OptiChat

The most likely conflict could be between ... **When demand is high, the model could be forcing high production levels to satisfy the commodity balance constraint, which in turn requires a large number of workers to meet such production levels.** However, ... to hire or fire enough workers is not flexible enough ... Therefore, I suggest ...



User

Additional Information about Problem

Our warehouse can request for additional goods via inventory redistribution. Which one do you think is more practical to change?



OptiChat

Adjusting the initial stock of goods ... thus **creating less pressure on your production team and workforce to meet high demands within tight deadlines** ... keep in mind it could also lead to increased storage and holding costs ...

OptiChat

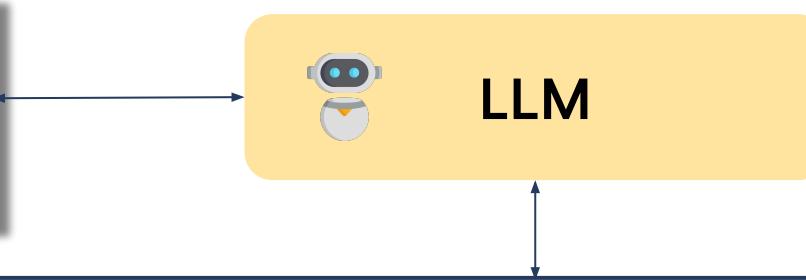
Key Idea: Leverage LLMs as a natural language interface to correct infeasible models.

User
Specific Questions about Model
What is the most likely conflict among these constraints that you believe is causing the infeasibility?

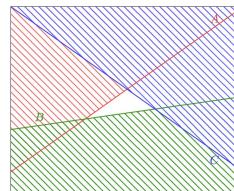
OptiChat
The most likely conflict could be between ... When demand is high, the model could be forcing high production levels to satisfy the commodity balance constraint, which in turn requires a large number of workers to meet such production levels. However, ... to hire or fire enough workers is not flexible enough ... Therefore, I suggest ...

User
Additional Information about Problem
Our warehouse can request for additional goods via inventory redistribution. Which one do you think is more practical to change?

OptiChat
Adjusting the initial stock of goods ... thus creating less pressure on your production team and workforce to meet high demands within tight deadlines ... keep in mind it could also lead to increased storage and holding costs ...



```
1 import json
2 import numpy as np
3 import gurobipy as gp
4
5 import os
6 with open('tmpData/sfXp01SzvKSMBeLzdp/data.json', 'r') as f:
7     data = json.load(f)
8
9
10
11 ServiceCost = data["cost"]
12 L = list(range(data["L"]))
13 Mousers = data["Mousers"]
14 C = data["C"]
15 HubCost = data["HubCost"]
16
17 # Define model
18 model = gp.Model('model')
19
20
21 # Define variables
22 HubPlaced = model.addVars(C, name='HubPlaced', vtype=gp.GRB.BINARY)
23 Serviced = model.addVars(len(C), len(L), name='Serviced', vtype=gp.GRB.BINARY)
24
25 # Define constraints
26
27 for i in C:
```



Pyomo Code
User Supplied

Identifying
IIS

$$\min_{\mathbf{x}, \delta \mathbf{A}^+, \delta \mathbf{A}^-, \delta \mathbf{b}^+} \sum_{(i,j) \in \mathcal{S}_A} (\delta \mathbf{A}_{ij}^+ + \delta \mathbf{A}_{ij}^-) + \sum_{i \in \mathcal{S}_b} \delta \mathbf{b}_i^+$$

subject to $(\mathbf{A} + \delta \mathbf{A}^+ - \delta \mathbf{A}^-)\mathbf{x} \leq \mathbf{b} + \delta \mathbf{b}^+$,

$$\mathbf{x} \in \mathbb{Z}^p \times \mathbb{R}^{n-p}, \quad \delta \mathbf{A}^+, \delta \mathbf{A}^-, \delta \mathbf{b}^+ \geq 0.$$

Resolving
Infeasibility

Experiment

Dataset: 63 infeasible problems adapted from the GAMS modelling library, PYOMO cookbook, and Resource task network model.

Adjusted to make infeasible by changing model parameters or added constraints.

Study Participants: 8 inexperienced and 7 expert optimization users.

Task: Troubleshoot an infeasible model with the help of OptiChat, evaluated via:

(i) Qualitative Description of User Experience

(ii) Accuracy: What fraction of the instances did they get a satisfactory answer

Experiment: Quantitative Results

Takeaway: Helps users across expertise levels, but easier to recover from errors if experienced users.

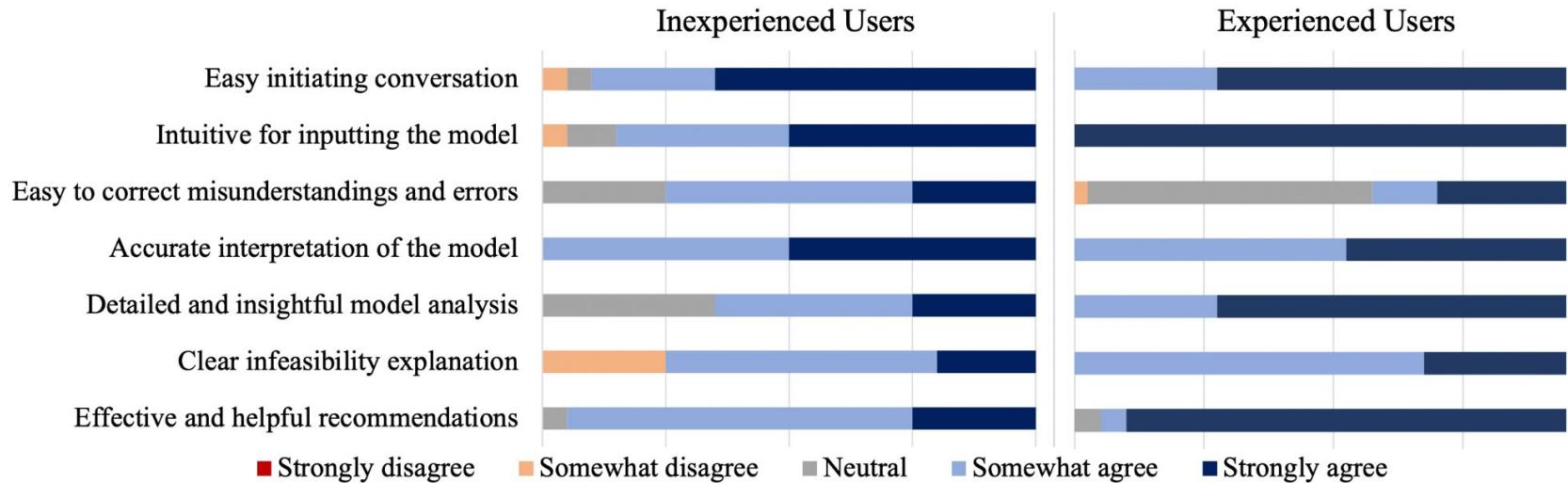
| Study group | Satisfactory answers | Troubleshooting success rate |
|---------------|----------------------|------------------------------|
| Inexperienced | 90.93% | 88% |
| Experienced | 87.20% | 96.77% |

Satisfactory Answers: Self-reported rate that chatbot's answers were considered satisfactory (no misinterpretation of the questions or inaccurate answers).

Troubleshooting: Success rate for trying to correct the assigned infeasible instance.

Experiment: Qualitative Results

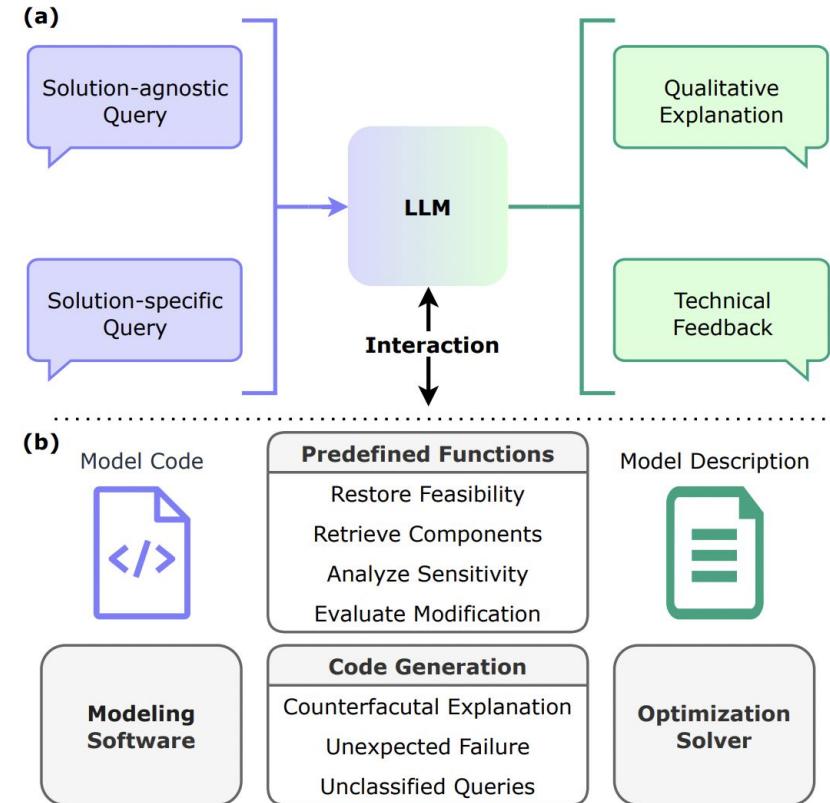
Takeaway: Very usable across all expertise levels!



It is very helpful that they can provide the literal explanation on what to improve while also providing some systematic understanding of the each problem formulation.

General Q&A

Same idea of using LLM as interface for dedicated tools underpins more general systems for validating optimization models.



OptiChat: Bridging Optimization Models and Practitioners with Large Language Models
Chen et al. IJDS (2025)

LLMs for Supply Chain Optimization

This model has also seen a lot of applied success (e.g., in Supply Chain Management)

👤 Show me the shipping plan.

🤖 Here is the plan. You can hover your mouse on each node or edge to see more details.

```
graph LR; S1 --> R1; S1 --> R2; S2 --> R1; S2 --> R2; S3 --> R2; R1 --> C1["20 L + 20 D"]; R1 --> C2["30 L + 20 D"]; R1 --> C3["30 L + 100 D"]; R2 --> C2["10 L"];
```

👤 Is it possible for Roastery 1 to be exclusively used by Cafe 2?

🤖 If we exclusively transport beans from Roastery 1 to Cafe 2, the cost would amount to \$2570, representing an approximate increase of $\frac{2570 - 2470}{2470} \approx 4\%$ compared to the current plan, which costs \$2470. Would you like to implement this change for future planning purposes?

Type your question here...

LLMs for Supply Chain
Optimization
Li, et al. (2023)

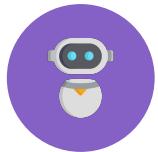


Takeaways



Interacting with a model is key to longevity!

Every model needs to deal with dynamic questions and updates.



LLMs can help domain experts understand a model

LLMs can help answer technical questions like the source of infeasibility



More to do!

Existing approaches do not help validate the *correctness* of a model.

Tutorial Gameplan

1. Background
2. Model Formulation

Coffee Break

3. Model Solving
4. Model Validation
- 5. Open Questions**

Revisiting Optimization Practice

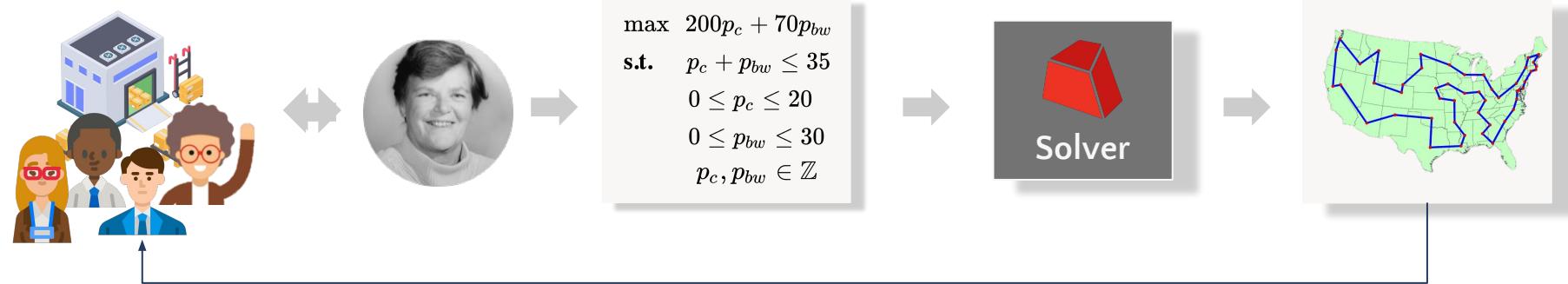
Traditionally, the OR community has focused on building really great algorithms.

$$\begin{aligned} \max \quad & 200p_c + 70p_{bw} \\ \text{s.t.} \quad & p_c + p_{bw} \leq 35 \\ & 0 \leq p_c \leq 20 \\ & 0 \leq p_{bw} \leq 30 \\ & p_c, p_{bw} \in \mathbb{Z} \end{aligned}$$



Revisiting Optimization Practice

.... But a lot of the tough work is actually figuring out the right problem to solve!



Problem Elicitation is Hard!

Initial problem descriptions are often vague, informal, or even contradictory!



“The whole process is like a flywheel... we’ll formulate, come up with a solution, then put it out there, gather feedback and come back to ‘what’s really your goal?’”



“The sheer number of stakeholders that we had to communicate with was especially challenging”



“99% of the time, something was not included in the model from the customer standpoint. They’ll say you can’t do that; and I said, well, you didn’t tell me that. And so then you have to go back and update the model.”

Interactive Problem Elicitation

Can we develop interactive systems that efficiently elicit underspecified problem descriptions, surface constraints, and navigate multi-stakeholder preferences?

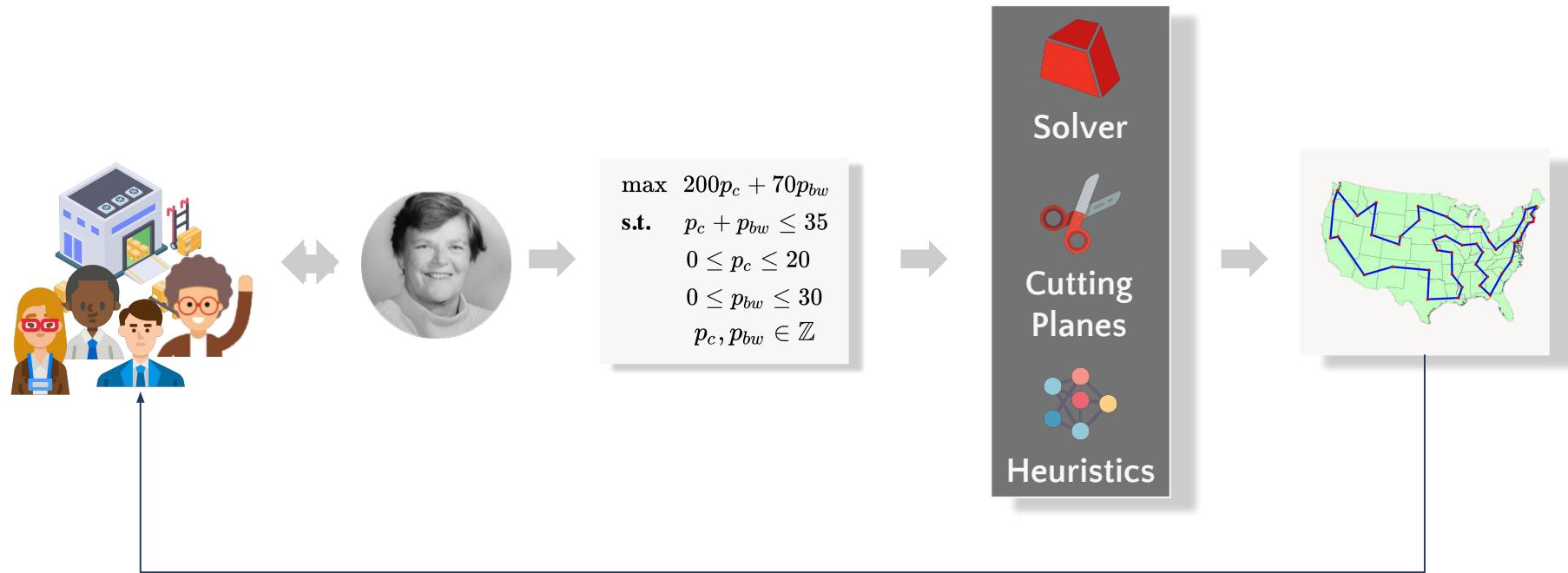
We need to decide where to open a set of facilities to service our customers ... **we want to achieve the goal of minimizing costs and maximizing coverage ...**



Here's a portfolio of solutions with different trade-offs between costs and coverage

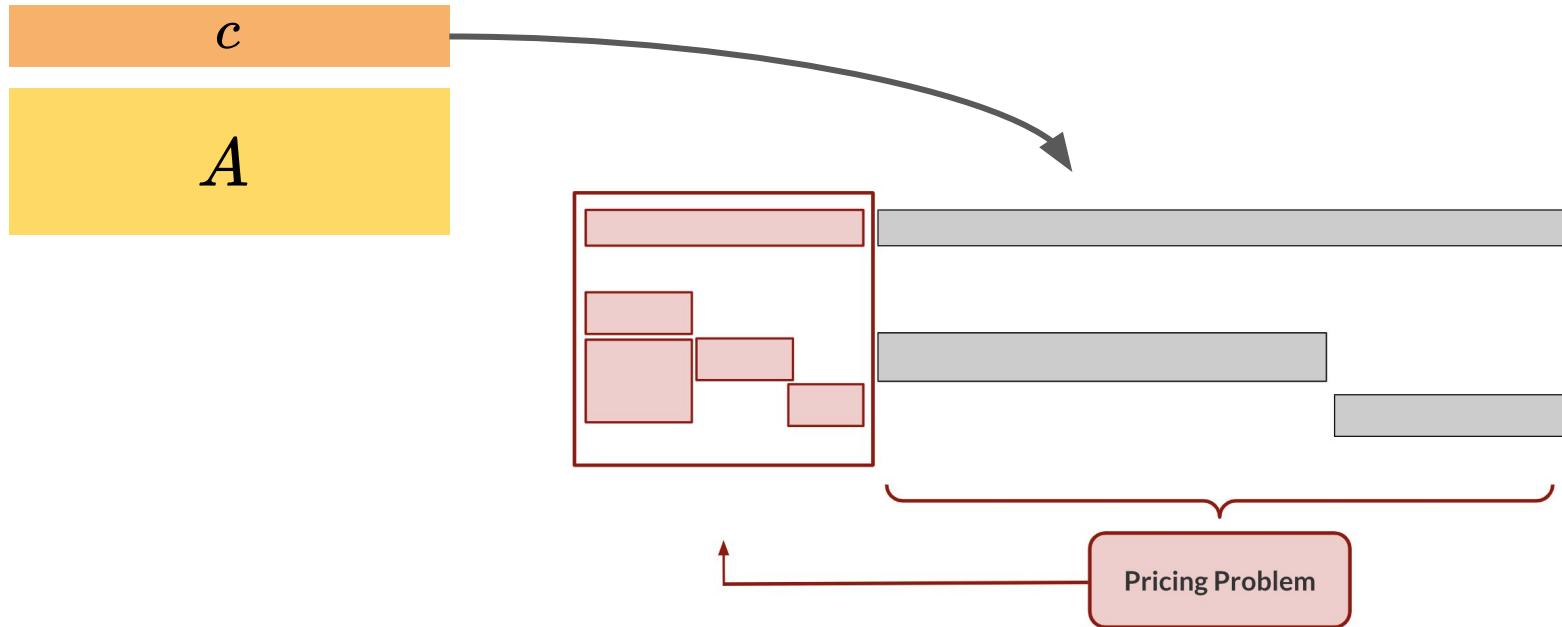
Revisiting Optimization Practice

.... and getting the right algorithmic building blocks to solve the problem efficiently!



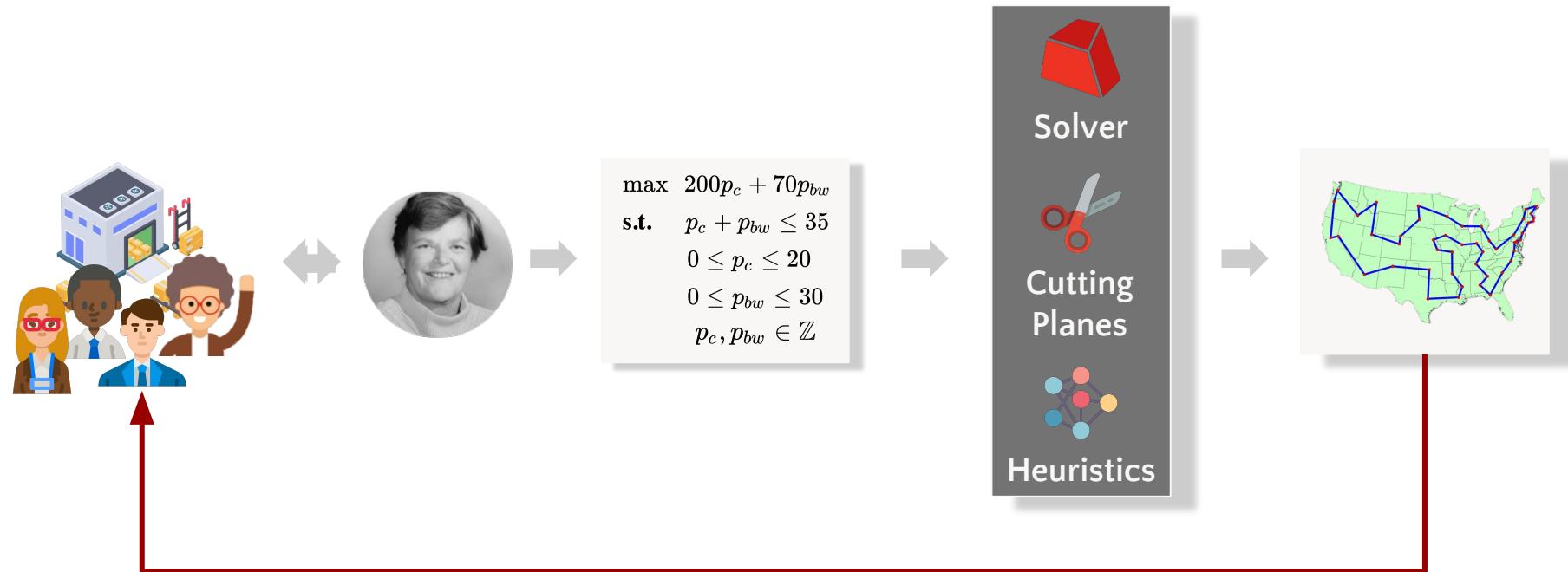
Automated Algorithm Design

Can ML be used to identify promising optimization structure to exploit, or find new cutting planes or primal heuristics to speed-up MILP solvers?



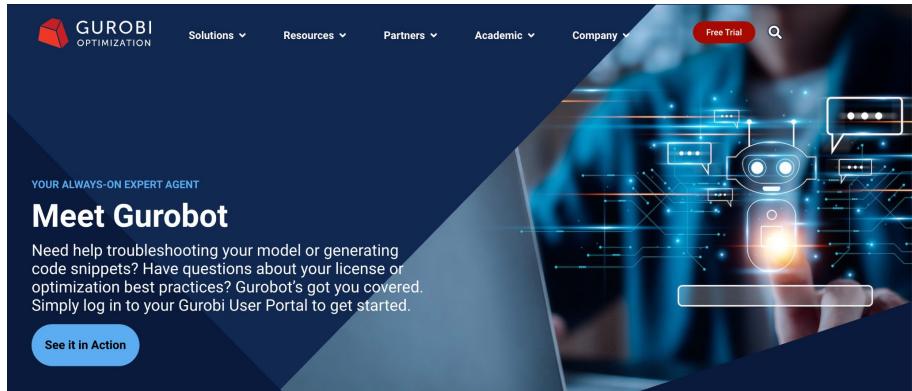
Revisiting Optimization Practice

.... and validating that the optimization problem actually solves the problem!



Auto-Formulation in the Wild

Industry interest in automated model formulation is growing, but can we trust it?



The screenshot shows the Gurobi Optimization website. At the top, there's a navigation bar with links for Solutions, Resources, Partners, Academic, and Company, along with a Free Trial button and a search icon. Below the navigation, a large banner features a blue-toned image of a person interacting with a futuristic-looking AI robot named "Gurobot". The robot has a screen-like face and is surrounded by glowing blue lines and data points. The text "YOUR ALWAYS-ON EXPERT AGENT" is above the robot, and "Meet Gurobot" is below it. A subtext explains: "Need help troubleshooting your model or generating code snippets? Have questions about your license or optimization best practices? Gurobot's got you covered. Simply log in to your Gurobi User Portal to get started." A blue button labeled "See it in Action" is at the bottom left.

Supply Chain Management

How Generative AI Improves Supply Chain Management

It can cut decision-making time from days to minutes and dramatically improve results. by Ishai Menache, Jeevan Pathuri, David Simchi-Levi and Tom Linton

From the Magazine (January–February 2025)

Building an AI Agent for Supply Chain Optimization with NVIDIA NIM and cuOpt



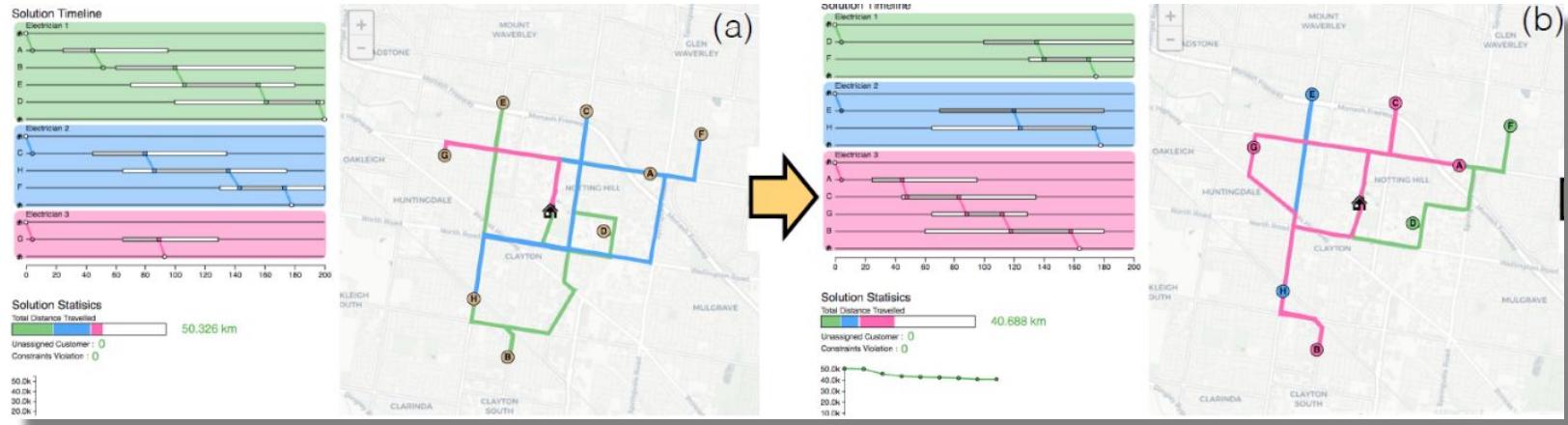
Jul 16, 2024

By Moon Chung, Hugo Linsenmaier and Victor Miara

 +27 Like  Discuss (0)

Trustworthy Optimization

Can we design systems that enable domain experts to accurately calibrate their trust in an optimization model?





lawlessc@stanford.edu
vitercik@stanford.edu



conlaw.github.io
vitercik.github.io



@LawlessOpt
@ellen-v

Thanks! Questions?

It's an exciting time for research at the intersection of ML and Optimization.

