The Positive Impact of AI in Optimization Research

Michael R. Wagner

September 2026

Motivation

- Frontier AI is rapidly becoming capable at hard technical work — not just writing and code, but mathematical modeling and problem-solving
- Optimization research delivers real value, yet remains slow, expert-intensive, and team-dependent
- If AI can shoulder much of this work, the impact is large:
  - Faster and cheaper — weeks of analysis in hours
  - More accessible — research beyond large, well-resourced teams
- How far can it go — and what remains fundamentally human?

Amazon Project: Model & Computational Challenge

Core optimization model:

$$
\begin{aligned}
\min_{S \subseteq N} \quad & \max_{\substack{i \in S \\ j \in N \setminus S}} d_{ij} \\
\text{s.t.} \quad & \rho - \epsilon \leq \frac{1}{V} \sum_{i \in S} v_i \leq \rho + \epsilon
\end{aligned}
\quad \Leftrightarrow \quad
\begin{aligned}
\min_{t, \mathbf{x}} \quad & t \\
\text{s.t.} \quad & d_{ij} x_i (1 - x_j) \leq t, \;\; i, j \in N \\
& \rho - \epsilon \leq \frac{1}{V} \mathbf{v}' \mathbf{x} \leq \rho + \epsilon \\
& \mathbf{x} \in \{0, 1\}^n,
\end{aligned}
$$

- Original MIQCQP Formulation
  - $n = 500$: 2 hours to solve (Xpress)
  - $n = 1,000$: Timed out after 10 hours (Xpress)
  - Need to handle $n = 50,000$
- AI-Assisted Solution
  - Collaboration with `o1-preview` (when it was first released)
  - Key insight: optimal solution must occur at a discrete d_{ij} value
  - Led to efficient binary search algorithm

Amazon Project: Solution Methodology & Results

- Key Insight (AI-Generated)
  - Transform quadratic constraints to linear
  - $d_{ij} x_i (1 - x_j) \leq t \ \Leftrightarrow \ x_i \leq x_j$ for a fixed value of t and $d_{ij} > t$
- Binary Search Algorithm
  - Search over (unique sorted) discrete d_{ij} values in place of t
  - Each iteration: solve simple linear IP feasibility problem
- Additional (Human) Improvements
  - Decomposition techniques
  - Parallel processing
- Dramatic Computational Improvement
  - Before: $n = 1,000 \rightarrow 10+$ hours
  - After: $n = 5,000 \rightarrow 10$ minutes
  - **300x speedup** in computation
- Model was piloted and has now launched in Europe

AIs as Students in a Doctoral-Level Optimization Course

- Experiment: "Enroll" frontier AIs as students in my doctoral optimization course at the University of Washington
- Course covers LP, MIP, DP, convex, robust optimization, and more

Frontier AIs graded on 12 doctoral optimization problems.

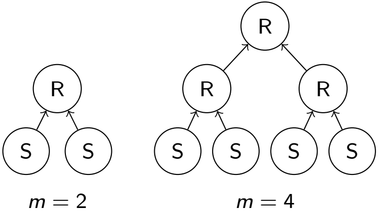
	T1	T2	T3	T4	T5	T6	T7	T8	O1	O2	O3	O4
Claude 3 Opus	×	✓	×	×	×	×	×	×	×	×	✓	×
DeepSeek-R1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	×
o1 pro	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×

T = textbook problems, O = original problems

- Single AIs already outperform most doctoral students
- Best single AI (o1 pro) scores 92%
- But they plateau on original research problems
  - Problem O4 stymied even o1 pro after many attempts!
  - Problem O4 is robust optimization, which has less training data
- Can we push further with multi-agent systems?

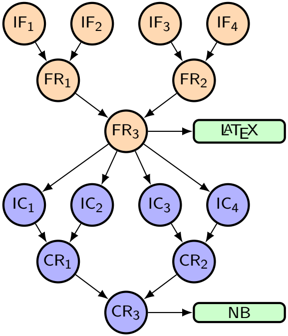
AI Teamwork: Collaborative Solving of Hard Problems

- AI model: o4 mini
- m parallel solvers $\rightarrow$ binary-tree reconciliation
- Problem O4 results (50 trials):
    - m=1: 28%
    - m=2: 42%
    - m=4: 56%
    - m=8: 68%
- Improvements statistically significant
- But diminishing marginal returns



Dynamic Programming Automator

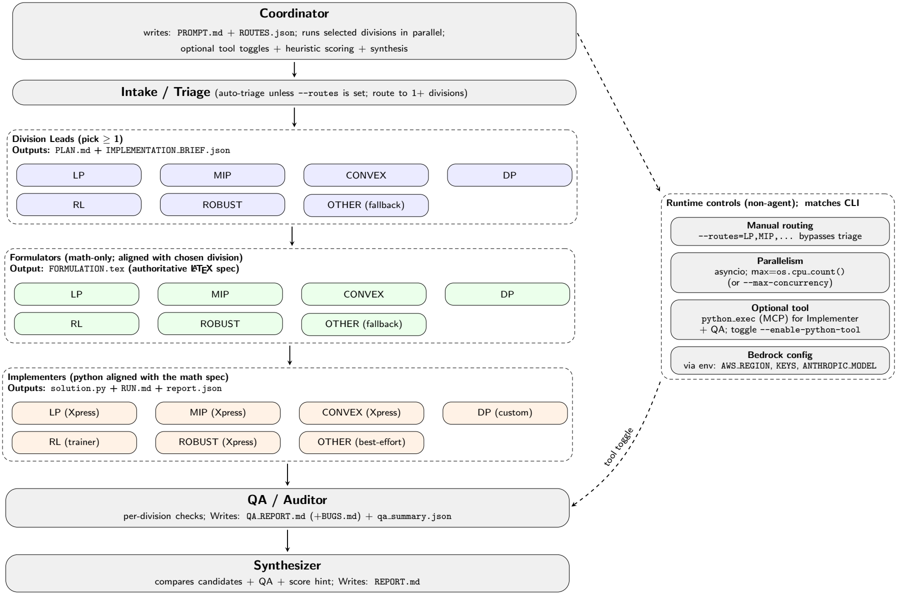
- AI model: o3
- Task: Natural-language $\rightarrow$ DP formulation + solver
- **Diamond #1** (Math): 4 formulators $\rightarrow$ 3 resolvers $\rightarrow$ LaTeX
- **Diamond #2** (Code): 4 coders $\rightarrow$ 3 resolvers $\rightarrow$ notebook
- Results (17 doctoral problems):
  - Single agent: 64.7%
  - Double-Diamond: 94.1%



Lessons from This Work

- What Worked
  - Hierarchical organization with clear roles
  - Parallel exploration followed by synthesis
  - Dramatic performance improvements over single agents
- Limitations
  - Fixed, predetermined structures
  - No adaptation to problem characteristics
  - Manual orchestration design — tedious!
  - No coding implementations apart from DP
- Key Question
  - Can we build adaptive organizations that route, scale, and select models based on the problem at hand?

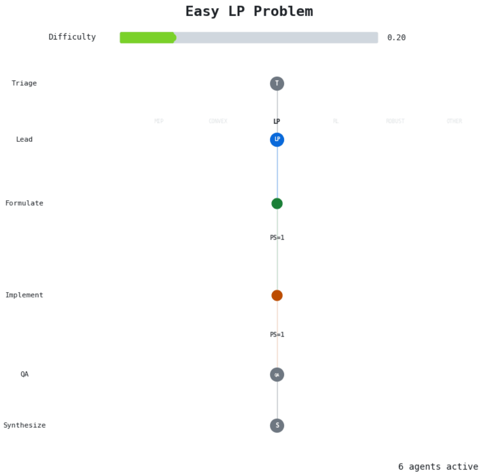
Organizational Chart



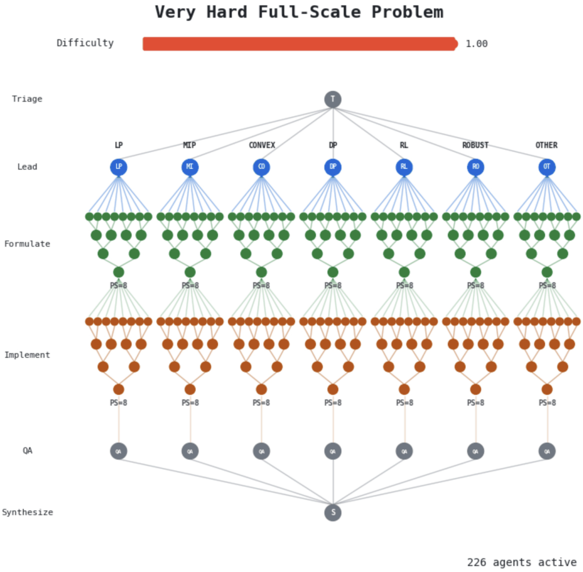
OrgAI: An Adaptive Organization of AI Agents

- Architecture: Claude Agent SDK + AWS Bedrock; 7 divisions (LP, MIP, CONVEX, DP, RL, ROBUST, OTHER). A Triage agent routes 1–7 in parallel; a Synthesizer agent picks the best.
- Four-stage pipeline per division: Division Lead $\rightarrow$ Formulator $\rightarrow$ Implementer $\rightarrow$ QA.
- Pyramid sub-structures: Formulator and Implementer expand into N parallel solvers $\rightarrow$ binary-tree resolvers.
- Adaptive allocation: Division-specific difficulty $d_{\text{eff}} \in [0, 1]$ drives pyramid size (PS 1$\rightarrow$8), model tier (Sonnet 4.5$\rightarrow$Opus 4.6), thinking budget/effort, and turn limits.
- Scales from 6 agents (easy, 1 division) to 226 (hard, 7 divisions); cost scales proportionally with difficulty.
- 70 pages of system prompts, written in collaboration with AI.

The Smallest/Weakest Organization



The Largest/Strongest Organization



Results: Curated Benchmarks + Hard-Problem Stress Test

- Curated benchmarks (120 problems, 6 types, Opus 4.6): v1 88.3% $\rightarrow$ v2 98.3% exact; 5/6 types 100% across both testbeds; RL gap fixed 30% $\rightarrow$ 90% (enriched prompts, PS=2)
- Stress test — all scaffolding removed: natural-language prose, no routing hints, no python tool, weaker model Sonnet 4.6; 7 hard problems from OpenAI GPT-5.2 Pro, verified by Opus 4.6
  - Triage: 5/7 correct primary classification from unstructured text; every problem routed to 2–3 divisions
  - Execution: 5/7 exact optimality (all polynomial-time problems + the ROBUST MILP exact; two NP-hard at 0.3% / 4.6% gap)
  - Synthesizer: 7/7 correct picks; end-to-end 29 min (all 7 in parallel); cost scales $\sim$proportionally with difficulty
- Notable example: a Job Shop Scheduling problem was solved by the MIP and OTHER divisions
  - The OTHER division used constraint programming, which greatly outperformed the MIP solution

OrgAI Replicates a Production Model

- Real Production Model: When I started at Amazon, I worked a few months on a stochastic dynamic program.
  - Reached production in the US and UK.
  - If you ordered Amazon Fresh or Whole Foods in the last 4 years, your order went through my model
- OrgAI: 3 complementary formulations from one prompt — DP, MIP, ROBUST
  - Replicated 90% of what I did in less than an hour, from a natural language prompt.
- Depth: proofs of convexity, monotonicity, edge-case catalog
- Synthesizer: recommended DP as best solution (matching the production decision)

Case Study: Auditing a Sustainability Model

- Review mode: point OrgAI at an existing model's artifacts (formulation, code, results); a Reviewer agent analyzes them, then the pipeline improves rather than reinvents
- The problem: a multi-objective sustainability model for a large delivery network
- Run twice (∼4 hours) — an audit for correctness, then a search for alternative formulations — delivering findings that would take weeks of manual analysis:
  - Audit: found 2 critical bugs — a weight-normalization collapse producing duplicate solutions, and an objective scale mismatch — plus 4 missing constraint families never implemented in code
  - Alternatives: produced a robust formulation for demand uncertainty, proved certain auxiliary variables unnecessary for correctness, and delivered a scenario sensitivity-analysis framework
  - Solver upgrade: introduced warm-starting, significantly reducing solve times

Some Takeaways and Questions

- AI is a **very capable** collaborator
- Claude Code will sometimes recommend I run OrgAI
- Research is changing **whether we want it to or not**
- AI can democratize research; domain experts without large teams can now pursue ideas that were previously out of reach
- If we no longer need junior researchers (as much), what happens to the pipeline that trains the next generation?
- Does advising students shift from a **research activity** to a **service activity**?
- What is the role of **industry scientists** when a senior researcher with AI can match the output of a team?
- Students learn by struggling; do AI shortcuts erode research intuition?
- A flood of AI-generated papers threatens to overwhelm peer review

People Are Still Needed

What AI cannot do (yet):

- **Choose** which problems are worth solving
- **Evaluate** novelty and significance
- **Exercise taste** — know what is elegant, surprising, or important
- **Teach** the next generation of researchers

Toward Sustainable Research in the Age of AI

Three principles:

1. **Embrace AI as infrastructure**
   Just as computation became infrastructure for the physical sciences, AI is becoming infrastructure for all research.
2. **Invest in verification, not just generation**
   Verification remains essential. Structure and QA matter more than raw capability.
3. **Redefine researcher roles around judgment**
   Producing research is automatable. Judgment, taste, and direction are not.

Research becomes more like directing an orchestra than playing every instrument.

Conclusion

- Collectives of AIs push past where single models plateau — OrgAI is adaptive, self-validating, and firm-like: 98.3% on curated benchmarks, 5/7 hard problems, and it replicates and audits real production models
- The positive impact is real and general: weeks of analysis in hours, one-off studies become living decision-support tools, and research is democratized beyond large teams
- The same shift is emerging across fields — yet people remain essential: the human role moves from *production* to *judgment* (choosing problems, taste, verification, teaching)
- Toward sustainable research: treat AI as infrastructure, invest in verification over raw generation, and organize — people and agents — around judgment

Thank you.

Any questions?