Multi-Objective Optimization

Lecture 8 - Management Science

Dr. Tobias Vlćek

Introduction

Client Briefing: EcoExpress Logistics

. . .

Operations Director's Dilemma:

"EU regulations demand 40% emission cuts, but we can't sacrifice profitability, service quality, or reliability!"

The Fleet Challenge

EcoExpress operates regional last-mile delivery across 3 cities

- EU Green Deal: 40% emission reduction by 2025
- Rising fuel costs (€2.1/L diesel)
- Amazon entering our market (speed pressure)
- Driver shortage (need automation-friendly vehicles)

. . .

Question: How do we transform our fleet while staying competitive?

Today's Learning Objectives

By the end of this lecture, you will be able to:

1. Explain why most decisions involve competing objectives
2. Identify and visualize Pareto optimal solutions
3. Apply normalization techniques to make objectives comparable
4. Implement apporaches to find trade-off solutions
5. Make decisions from a Pareto frontier

Quick Recap: Local Search

Last week we optimized routes for delivery:

- Started with greedy construction (e.g. Nearest Neighbor)
- Improved with local search (e.g. 2-opt)
- Considered time windows
- But: We only optimized distance

. . .

Question: What if we also care about emissions, cost, AND customer satisfaction?

The Problem

Single vs Multi-Objective

Single Objective

- “Minimize total distance”
- Clear winner. Easy, right!

. . .

Multiple Objectives

- “Minimize cost AND emissions AND maximize speed”
- No clear answer…

. . .

Question: Any idea how to approach this?

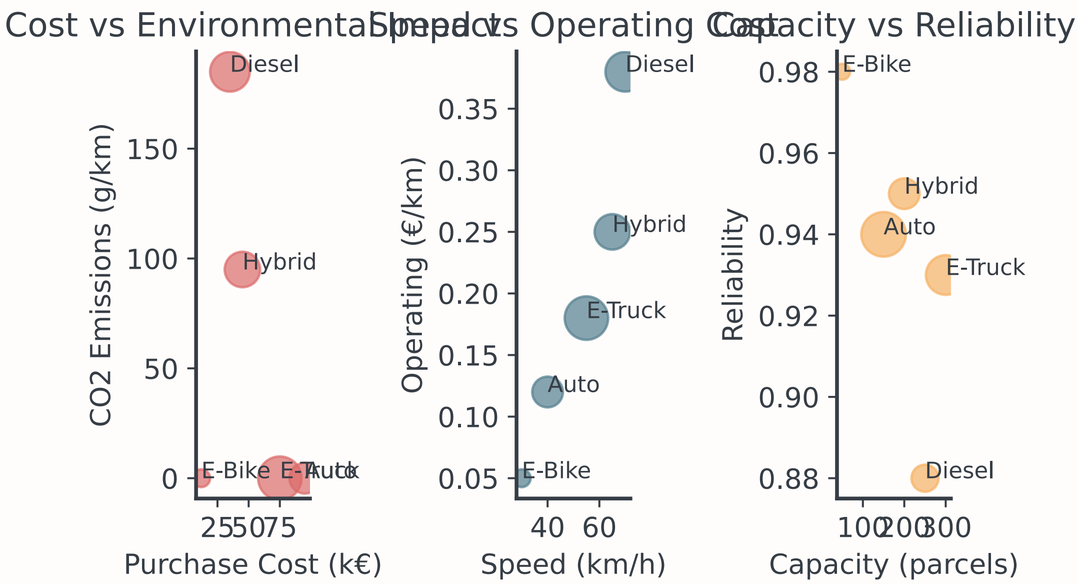
EcoExpress Vehicle Options

Type	Purchase Cost (€)	Operating (€/km)	CO2 (g/km)	Speed (km/h)	Capacity (parcels)	Reliability
E-Truck	75000	0.18	0	55	300	0.93
Hybrid	45000	0.25	95	65	200	0.95
Diesel	35000	0.38	185	70	250	0.88
E-Bike	12000	0.05	0	30	50	0.98
Auto	95000	0.12	0	40	150	0.94

. . .

Question: Which vehicle is “best” for EcoExpress?

Trade-offs Everywhere



. . .

> ! Important
>
> Every vehicle excels at something different!

Real Business Constraints

Beyond the numbers, consider:

- EU regulations: Carbon tax of €100/ton CO_2 starting 2025
- Competition: Amazon promises 2-hour delivery
- Labor market: Autonomous vehicles reduce driver dependency
- Urban zones: Zero-emission zones in city centers
- Peak times: Black Friday = 3x normal volume

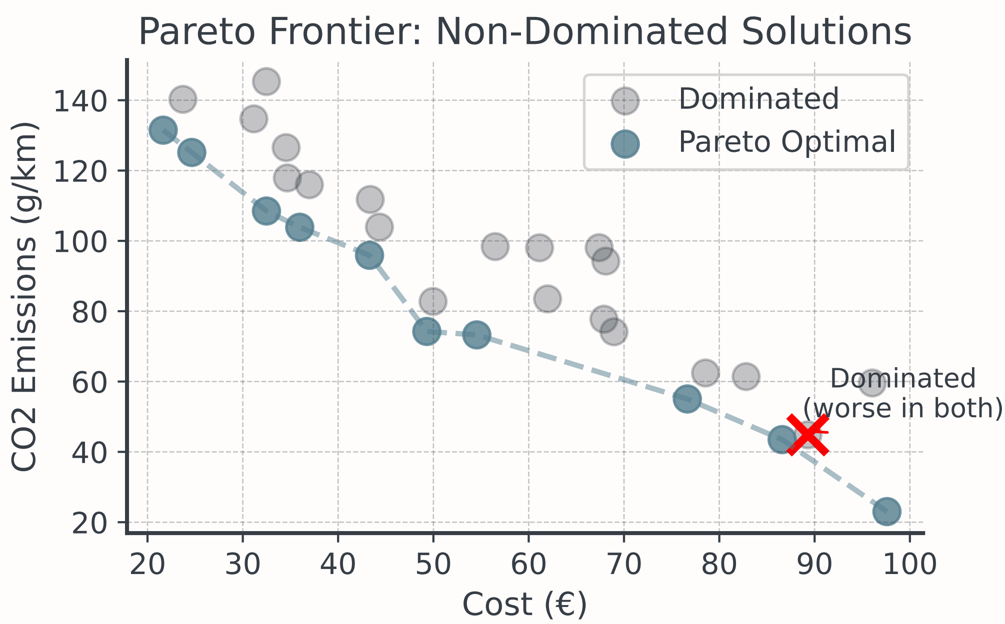
. . .

> ! Important
>
> There is no single “optimal” solution - only trade-offs

Pareto Optimality

Dominated Solutions

A solution is dominated if another solution is:



. . .

> ! Important
>
> Better in at least one objective and not worse in any objective!

The Pareto Frontier

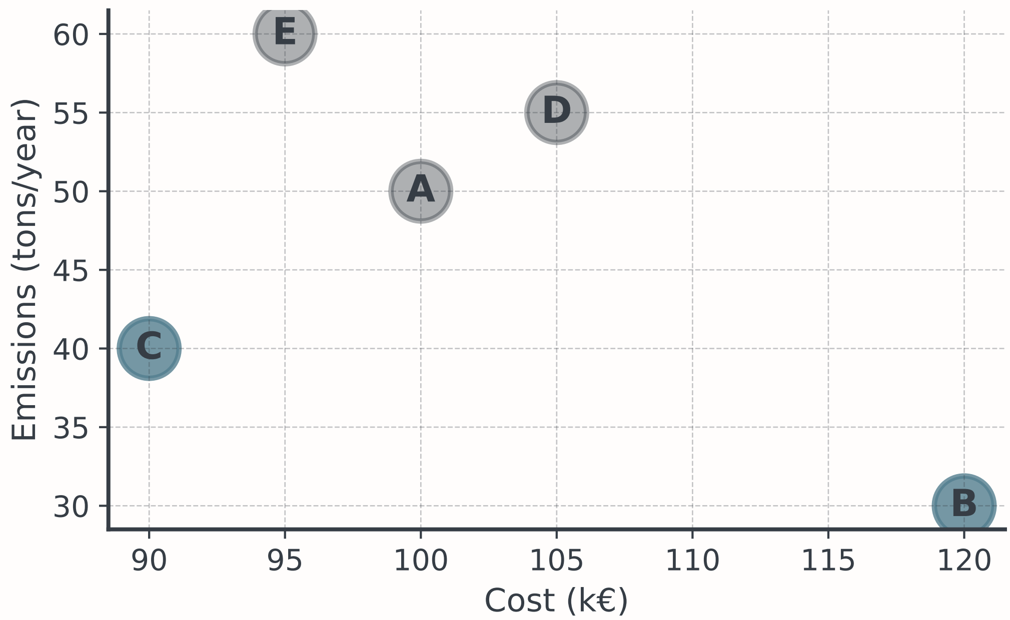
The Pareto frontier is the set of all non-dominated solutions

- No solution is objectively "better"
- Each represents a different trade-off
- Moving along frontier: gain in one objective, loss in another
- Decision makers choose based on preferences

. . .

Question Do you think you get the idea?

Find the Non-Dominated

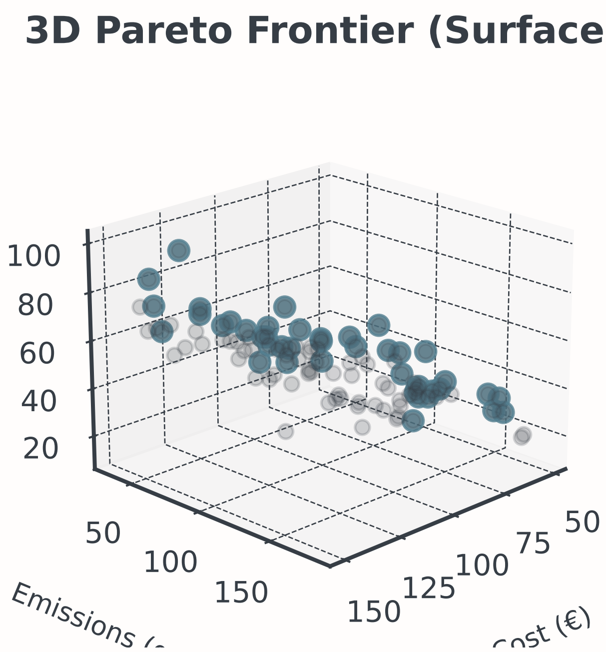


...

Question: Which fleets are non-dominated?

Three+ Objectives

With 3 objectives, the Pareto frontier becomes a surface:



...

! Important

Harder to visualize, but same principle applies!

Fleet Composition Problem

The Fleet Challenge

EcoExpress needs to replace their 80 diesel vans

- Must meet EU regulation: Average emissions ≤ 111 g CO_2/km
- Need capacity for 22,000 parcels/day
- Must balance cost vs. service quality
- 5 vehicle types available, each with trade-offs

. . .

Question: How do we choose the right mix?

Vehicle Options Recap

Type	Purchase Cost (€)	Operating (€/km)	CO2 (g/km)	Speed (km/h)	Capacity (parcels)	Reliability
E-Truck	75000	0.18	0	55	300	0.93
Hybrid	45000	0.25	95	65	200	0.95
Diesel	35000	0.38	185	70	250	0.88
E-Bike	12000	0.05	0	30	50	0.98
Auto	95000	0.12	0	40	150	0.94

. . .

Notice: No single vehicle is “best” at everything!

Fleet Composition Framework

This is a discrete selection problem, not continuous allocation

Decision Variables:

- Fleet: How many of each vehicle type? (discrete/integer)
- n_i = number of vehicles of type i (integers!)
- Example: $n_{\text{E-Truck}} = 20$, $n_{\text{Hybrid}} = 30$, etc.

Objective 1: Total Cost

Purchase cost + Operating cost over 3 years

$$\text{Total Cost} = \sum_i n_i \cdot (P_i + O_i \cdot d \cdot y)$$

- n_i = quantity of vehicle type i
- P_i = purchase cost of vehicle type i
- O_i = operating cost per km for type i
- d = daily distance × days per year
- y = years

Objective 2: Service Score

Composite measure of fleet performance

$$\text{Service Score} = 0.5 \cdot C_{\text{score}} + 0.3 \cdot R_{\text{score}} + 0.2 \cdot S_{\text{score}}$$

- $C_{\text{score}} = \min\left(1.0, \frac{\text{Total Capacity}}{22000}\right)$ (capacity adequacy)
- $R_{\text{score}} = \frac{\sum n_i \cdot r_i}{\sum n_i}$ (weighted avg. reliability)
- $S_{\text{score}} = \frac{\sum n_i \cdot s_i}{70 \cdot \sum n_i}$ (normalized speed)

. . .

> **ℹ Note**
>
> Service score captures multiple performance dimensions in one metric!

Hard Constraint: Emissions

EU regulation creates a feasibility boundary

$$\text{Average CO}_2 = \frac{\sum_i n_i \cdot e_i}{\sum_i n_i} \leq 111 \quad \text{g/km}$$

Where e_i = CO_2 emissions per km for vehicle type i

. . .

This eliminates some solutions:

- All diesel vans: 185 g/km > 111
- Mix with too many diesel: Still violates
- Zero-emission + some diesel: Might work

Data Source

Where Do These Numbers Come From?

. . .

Vehicle Specifications:

- Purchase costs: Manufacturer quotes, market research
- Operating costs: Fuel/electricity prices, maintenance records
- Capacity: Vehicle specs (cargo volume, weight limits)

- Reliability: Historical uptime data, manufacturer warranties
- EU Standards: WLTP certification for vehicles
- Electric vehicles: Grid carbon intensity (kWh → g CO_2)

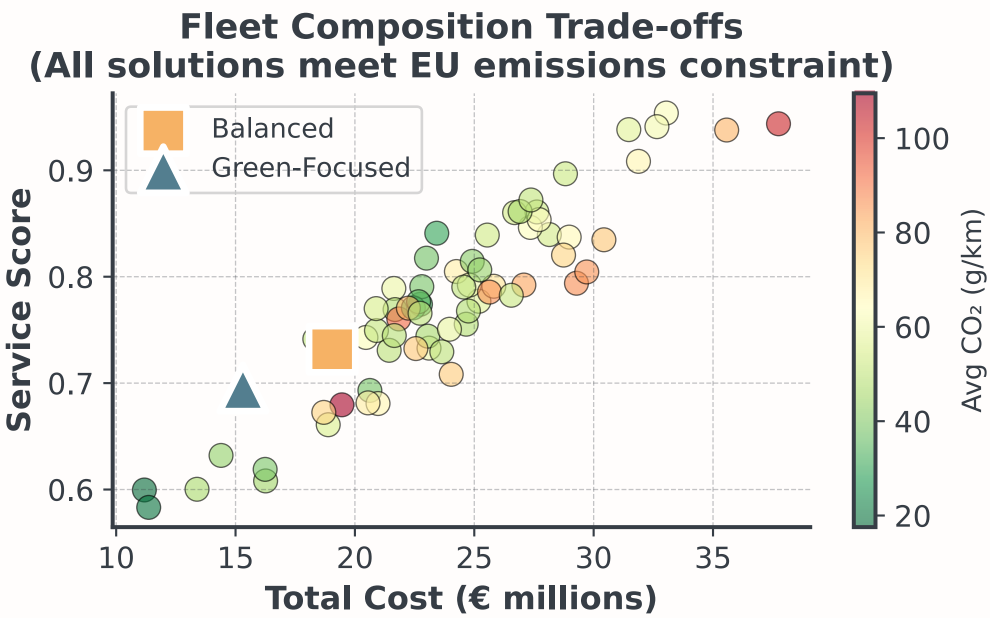
Example Fleet Comparison

```
Three Fleet Strategies:

         name    cost  service        co2  capacity  vehicles
 Cost-Focused 28.9996 0.809705 120.714286     15000        70
     Balanced 19.0478 0.731840  33.928571     13250        70
Green-Focused 15.3102 0.695373   0.000000     12750        75


Cost-Focused: ✗ VIOLATES (CO2: 120.7 g/km)
Balanced: ✓ Compliant (CO2: 33.9 g/km)
Green-Focused: ✓ Compliant (CO2: 0.0 g/km)
```

…

Question: Which strategy would you choose?

Visualizing Fleet Trade-offs




```
Generated 68 feasible fleet compositions
```

…

> ! Important
>
> Each point is a different fleet mix, all meeting emissions constraint!

Solution Approaches

Multi-Objective Optimization

You can use optimization solvers or heuristics!

. . .

With Optimization Solvers

- Weighted Sum Method
- ε-Constraint Method
- Goal Programming
- Optimal solutions
- Need mathematical model

With Heuristics

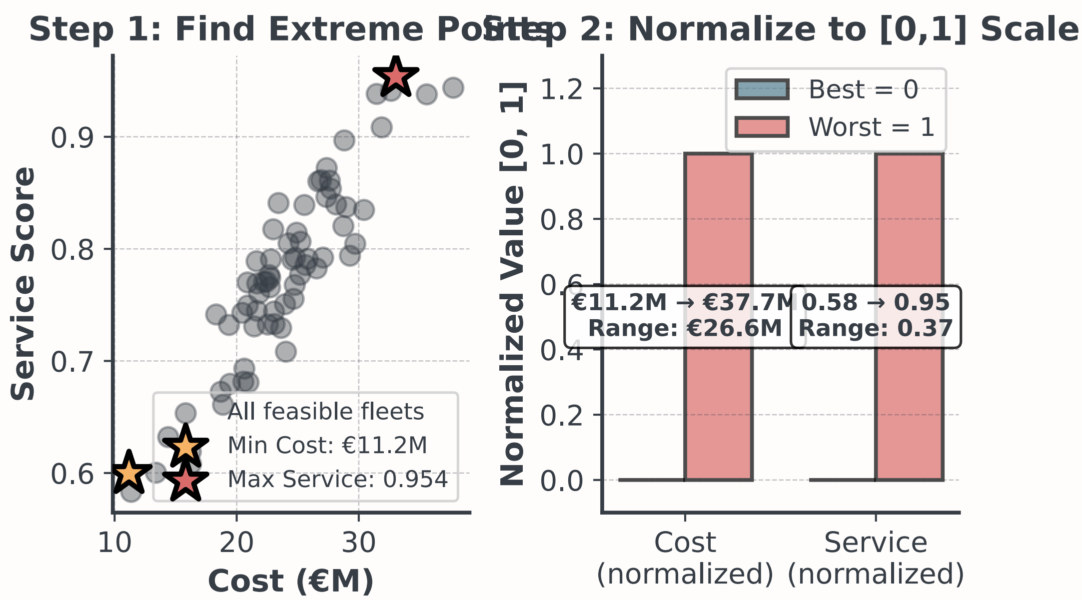
- Weighted Greedy Construction
- Multi-Objective Local Search
- Metaheuristics
- Good solutions, fast
- No optimality proof

. . .

> ! Important
>
> In this lecture we use heuristic approaches!

Foundation: Extreme Points

First step for BOTH approaches - find the boundaries:

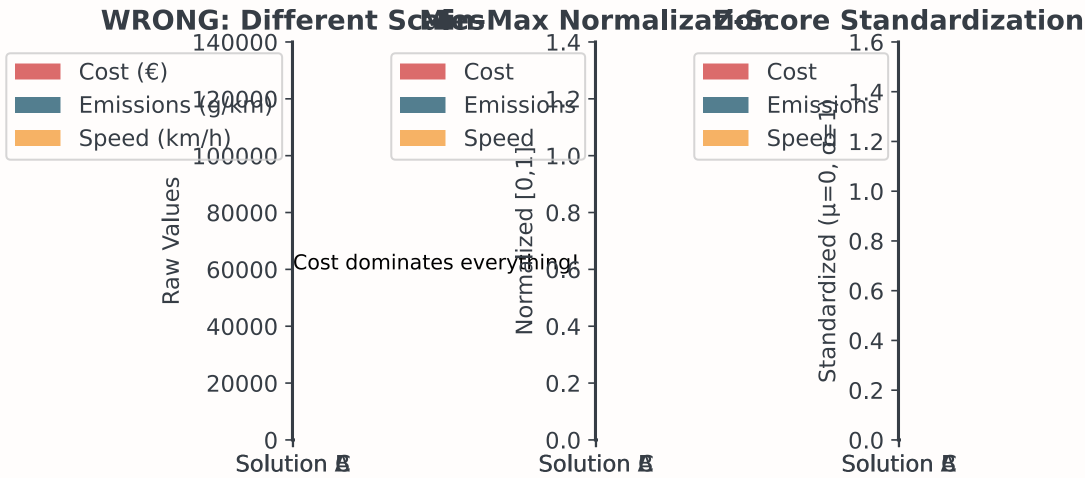


...

Question: Why is normalization essential?

Critical: Normalization

Without it, your analysis is meaningless



...

Question: Any intuition on how to do [0,1] normalization?

How to Normalize

The Normalization Formula for [0,1]

$$\text{Normalized}_i = \frac{x_i - x_{min}}{x_{max} - x_{min}}$$

. . .

In Python, this is rather simple!

. . .

```python
def normalize_objectives(data):
    return (data - data.min()) / (data.max() - data.min())

# Now weights actually mean something
weighted_score = w1 * normalize(cost) + w2 * normalize(emissions)
```

. . .

Tip

Easy, right?

Extreme Points

There are several reasons why extreme points matter:

1. Trade-off Space: Min/max values bound your Pareto frontier
2. Enable Proper Normalization: Need ranges for scaling to [0,1]
3. Feasibility: If single objectives not achievable, problem infeasible
4. Stakeholder: "Best cost is €50k, best emissions is 40kg"

. . .

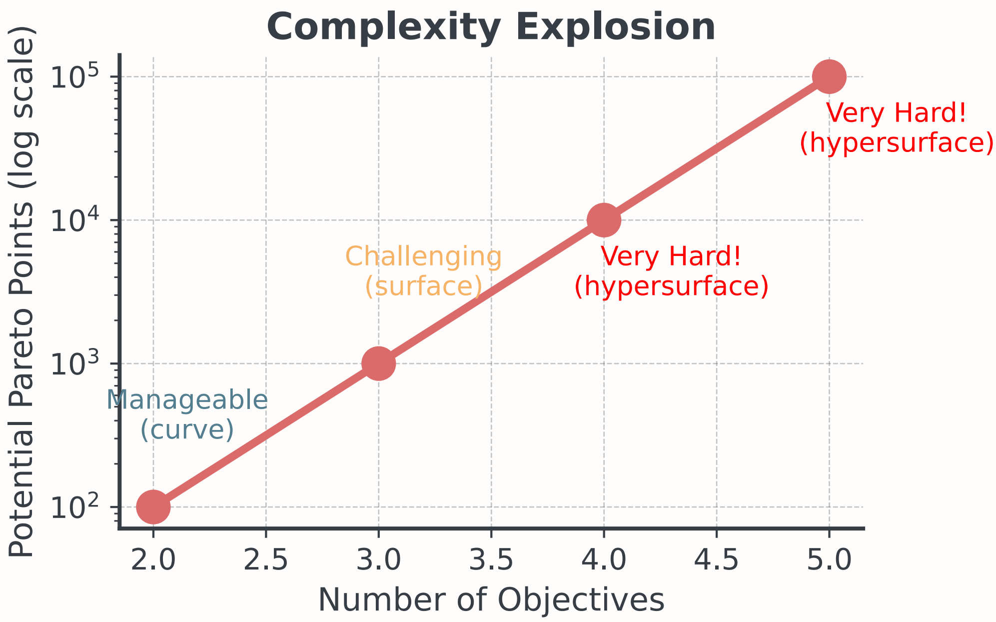
Implementation Pattern:

```python
def find_extreme_points(problem):
    # Solve for minimum cost (ignore emissions)
    min_cost_solution = minimize(cost_objective, constraints)
    # Solve for minimum emissions (ignore cost)
    min_emissions_solution = minimize(emissions_objective, constraints)
```

Computational Complexity

How hard does it get with more objectives?

. . .



…

> Tip
>
> Why? Because there are just way more potential solutions to check!

Solver-Based Methods

Quick overview - you won't implement these in assignments

1. Weighted Sum: Minimize $w_1 \times \text{cost} + w_2 \times \text{emissions}$
   - Simple, fast for convex problems
2. **ε**-Constraint: Minimize cost subject to emissions $\leq \varepsilon$
   - Systematically vary ε to find complete frontier
3. Goal Programming: Minimize deviations from targets
   - Set target for each objective, minimize weighted deviations

…

> Note
>
> For your fleet optimization: You'll use heuristic approaches instead!

Heuristic Approach

The Heuristic Strategy

For problems without mathematical models

…

1. Construction: Build initial solutions with weighted greedy
2. Improvement: Multi-objective local search
3. Selection: Filter dominated solutions to find Pareto frontier

. . .

> ! Important
>
> Key difference from solvers:
>
> - Solvers: Need mathematical model, guarantee optimality
> - Heuristics: Work with any evaluation function, find good solutions fast

Why Heuristics?

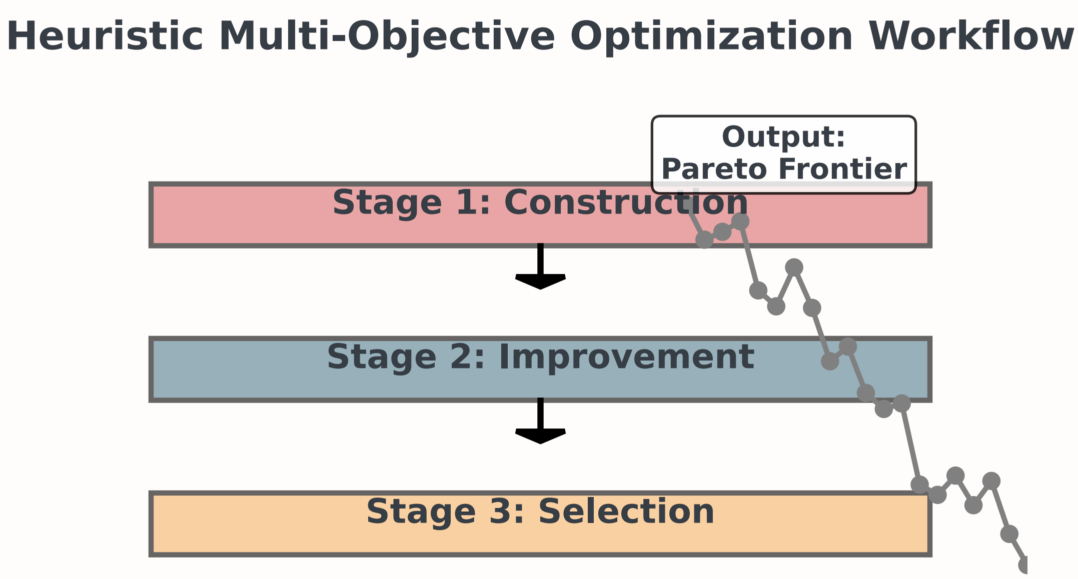
Depending on the problem:

- Combinatorial explosion
- Huge solution space even for one problem
- Evaluating one solution might thus take too long
- Need diverse Pareto frontier, not just one "optimal" solution
- Open Source Solvers too slow
- Commercial solvers too expensive

. . .

Question: How do we build good solutions without a solver?

The Three-Stage Heuristic Process



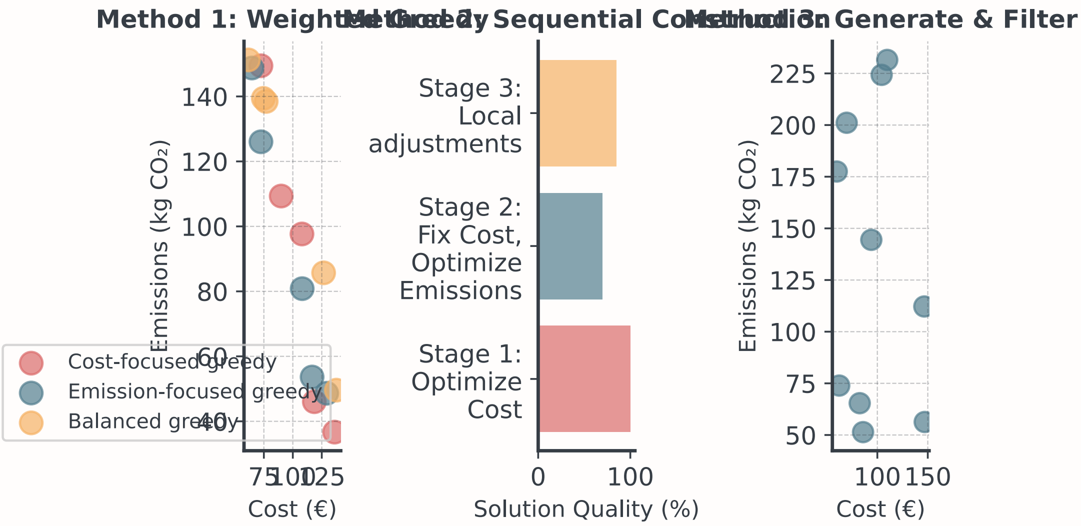
. . .

This is what you'll implement in your assignments!

Construction & Improvement

Construction Methods for MOO

How to build initial solutions when you have multiple objectives?



. . .

> **Note**
>
> Three choices (for starters). Let's check them out!

Weighted Greedy Construction

Making greedy choices on a weighted objective

1. Choose weight vector $w = (w_1, w_2)$
2. At each step, pick the choice that minimizes:

$$w_1 \cdot \text{cost}\ (x) + w_2 \cdot \text{emissions}\ (x)$$

3. Build complete solution greedily
4. Repeat with different weights to explore frontier

. . .

> **Tip**
>
> Different weights explore different trade-offs! Easy, right?

Sequential Greedy (Lexicographic)

Optimize one objective at a time, in priority order

1. Rank objectives by priority
   - E.g. cost (most important) and then emissions (tie-breaker)
2. At each step:
   - Find choices that minimize primary objective
   - If tie → use secondary objective
3. Build one working solution

. . .

> 💡 Tip
>
> We could also accept primary values within 10% of best so secondary has more influence!

Diverse Starting Pool

Generate many random solutions, keep the non-dominated ones

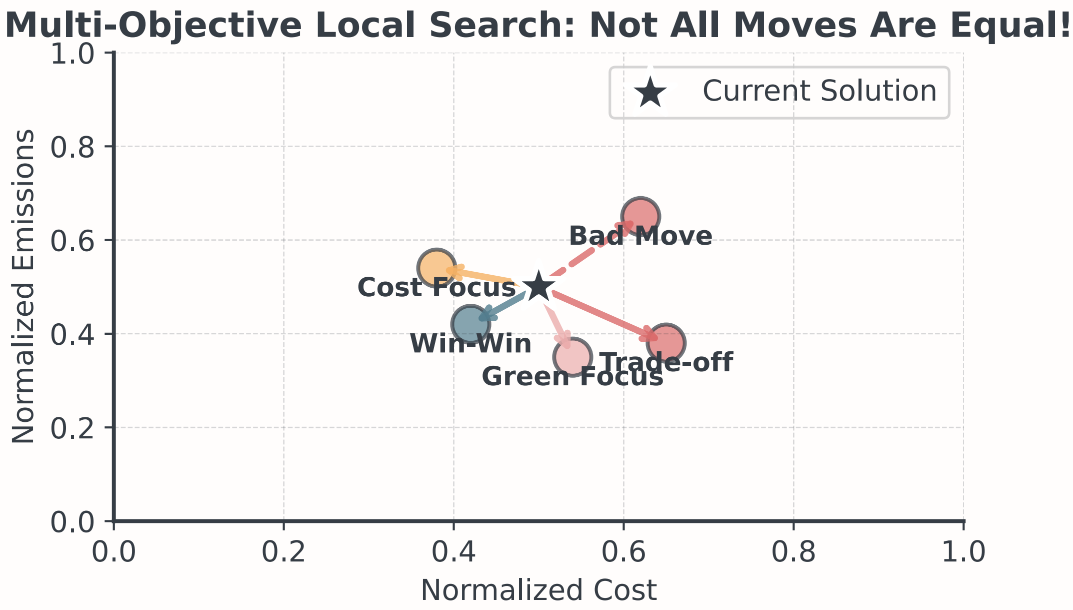
1. Generate N random solutions (e.g., N=100)
2. Evaluate all solutions on both objectives
3. Filter to keep only non-dominated solutions
4. Result: A diverse set of Pareto-optimal solutions

. . .

> 💡 Tip
>
> - Explores entire solution space
> - No bias toward specific weights
> - Great for warm-starting local search

Local Search for Multi-Objective

Special moves that improve multiple objectives:



. . .

Question: Which moves are acceptable?

MOO Local Search Rules

Accept a move if:

1. Dominance: New solution dominates current (win-win!)
2. Trade-off: Improves primary, acceptable loss in secondary
3. Probabilistic: Use temperature (like simulated annealing)

. . .

> ! Important
>
> Always keep all your objectives in mind when making decisions.

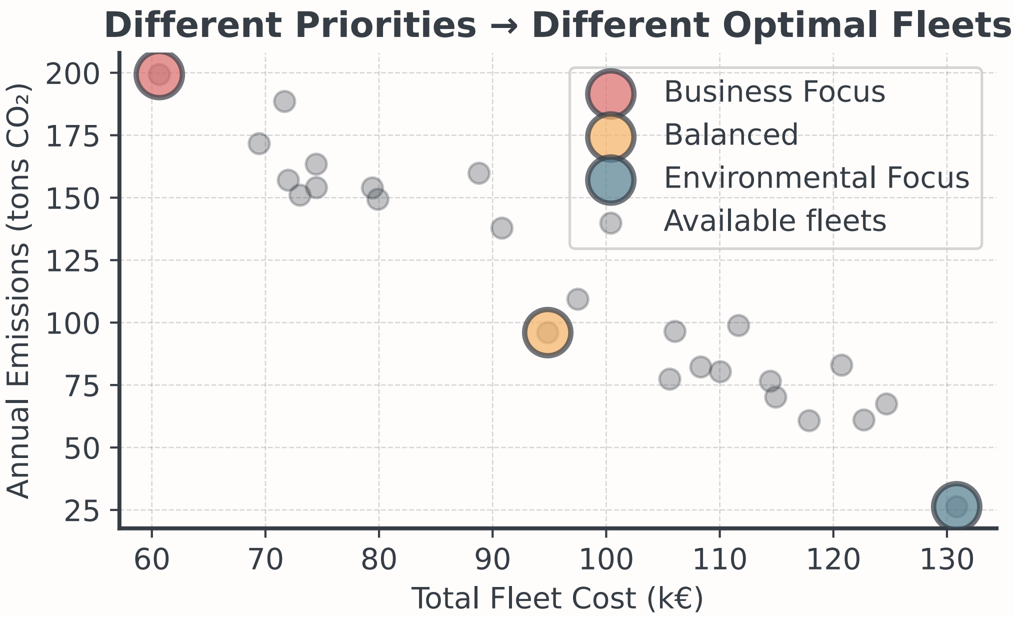
From Pareto Front to Decision

How to Choose!

1. The Knee Point: Find the "elbow" where improvement slows
2. Satisficing Levels: Set minimum acceptable thresholds
   - Cost must be < €100k (budget constraint)
   - Emissions must be < 100 kg (regulatory limit)
   - Service level must be > 90% (customer requirement)
3. Stakeholder Preferences: Let business priorities guide
   - Sustainability: Minimum emissions that meets constraints
   - Operations: Maximum service level within budget

Weighting has an Impact

The weights thus reflect your values!



. . .

> 💡 Tip
>
> Depending on your weight, the choice will vary.

Advanced

Speed vs Sustainability Dilemma

The Three-Way Trade-off in E-Commerce

1. Minimize Delivery Time (1-day/2-hour promise)
2. Minimize Cost (fuel, labor, fulfillment)
3. Minimize Environmental Impact (carbon footprint)

. . .

Faster delivery = More vehicles less full = Higher emissions

. . .

Question: What could retailers do?

Moving the Frontier

Instead of point on the frontier, move the entire frontier:

. . .

Question: Any idea of examples?

. . .

> Tip
>
> R&D can fundamentally change what's possible!

Briefing

Today

Hour 2: This Lecture

- Multi-objective
- Pareto optimality
- Weighted greedy
- Local search MOO

Hour 3: Notebook

- Bean Counter CEO
- Find Pareto frontier
- Apply weighted greedy
- Normalize objectives

Hour 4: Competition

- Fleet composition
- Vehicle selection
- Cost vs service
- Justify choice!

The Competition Challenge

EcoExpress Sustainable Fleet Design

. . .

1. Select optimal fleet mix (5 vehicle types)
2. Balance cost vs. service score
3. Meet EU emission constraint ($\leq$ 111 g CO_2/km)
4. Ensure sufficient capacity (22,000 parcels/day)

. . .

> ! Important
>
> Find the best trade-off for your business priorities!

Choosing Your MOO Approach

Different situations call for different methods:

Situation	Best	Why
Clear priorities	Sequential greedy	Fast, hierarchy
Exploring	Weighted greedy	Different solutions
Many solutions	Diverse pool	Builds frontier
Quick solution	Single weighted	One good compromise
Improve existing	Multi-objective local	Refines trade-offs

. . .

> Tip
>
> Competition? Generate diverse pool or weigted, then improve with local search.

Implementation Pitfalls to Avoid

Common bugs that cost you time:

1. Forgetting to Normalize
   - Always normalize to [0,1] first!
2. Optimizing Too Many Objectives
   - 2-3: Manageable, 4+: Exponentially harder
   - Combine related objectives or use constraints
3. Not Checking Solution Feasibility
   - Always verify constraints after optimization

Summary

Key Takeaways:

- Real decisions have multiple conflicting objectives
- Pareto frontier shows all rational trade-offs
- Normalization is essential for fair comparison
- Weights reflect values, make them explicit
- Visualization crucial for decision-making

Break!

Take 20 minutes, then we start the practice notebook

Next up: You'll become Bean Counter's expert

Then: The Sustainability competition

Bibliography