Methods and Models for Combinatorial Optimization

Introduction

Luigi De Giovanni

Dipartimento di Matematica "Tullio Levi-Civita", Università di Padova

Contacts

Luigi De Giovanni

Dipartimento di Matematica, office 423

luigi@math.unipd.it

tel. 049 827 1349, **Zoom meeting**

office hours: Thursday, h 10:30 - 12:30 (or others, email the teacher)

Moodle of the course unit

https://stem.elearning.unipd.it/course/view.php?id=7086

*http://www.math.unipd.it/∼luigi/*courses/metmodoc.html

Course unit goals

- Introduction to advanced modelling and solution techniques for **combinatorial optimization problems** in decision supporting, where an optimal solution has to be determined among a number of alternatives that *combinatorially explodes*

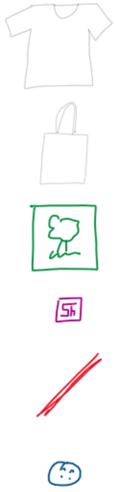
- The course aims at providing **mathematical and algorithmic tools** to solve optimization problems of practical interest, also with the use of the most popular **software packages or libraries**

- Ability to search for, find, understand, **adapt and implement state-of-the-art approaches to solve real-world combinatorial optimization problems**

Combinatorial Optimization: some examples

- Logistic and transportation network: optimal origin-destination paths, optimal pickup/delivery routes, line configuration, driver scheduling
- Production management: production and resource planning, job shop scheduling, optimal cutting patterns
- Machine learning: neural network configuration, optimal structure and weight of neural networks
- Data-driven decision making: cooling schedule based on massive simulation, air traffic management based on trajectory repositories
- Optimization on graphs and networks: coloring, cliques, quickest paths, multicommodity flows
- Telecommunication networks: telecom-facility location, virtual network configuration, optimal routing
- Social network analysis: community detection, influence maximization
- **.. and many others**

Combinatorial optimization problem: TOY example 1 "Young Money Makers"



10 T-shirts

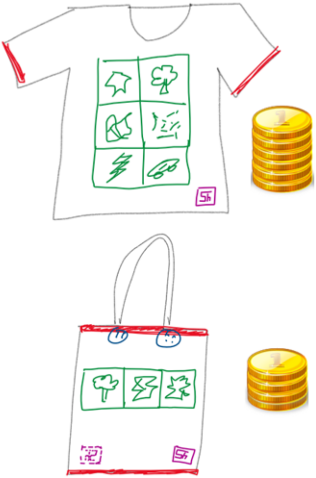
15 bags

32 drawings

24 labels

40 profiles

15 buttons



Goal:

Decisions: how many T, B?

The space of feasible combinations

- "Easy" to find a feasible solution
- "Easy" to find the optimal solution if all the feasible combinations can be explored
- but, *what if the number of product models and/or resources is large?*

How to manage the
combinatorial explosion of the size of the solution space
using a unifying approach?

↓

Methods and Models for Combinatorial Optimization

Combinatorial optimization problems: TOY example 2 "Farm 4.0"

A farmer owns 11 hectares of land where he can grow potatoes or tomatoes. Beyond the land, the available resources are: 70 kg of tomato seeds, 18 tons of potato tubers, 145 tons of fertilizer. The farmer knows that all his production can be sold with a profit of 6000 Euros per hectare of tomatoes and 7000 Euros per hectare of potatoes. Each hectare of tomatoes needs 7 kg seeds and 10 tons fertilizer. Each hectare of potatoes needs 3 tons tubers and 20 tons fertilizer. How does the farmer divide his land in order to gain as much as possible from the available resources?

Using a mathematical model: formulation

- Declare "what" is the solution, instead of stating "how" it is found
- What should we decide? **Decision variables**

  $x_T \geq 0,\ x_P \geq 0$

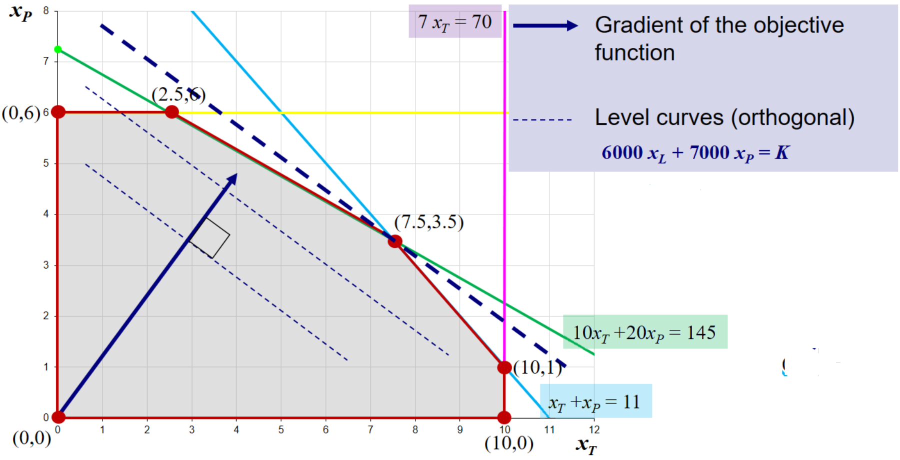
- What should be optimized? **Objective** as a function of the decision variables

  $\max 6000\, x_T + 7000\, x_P \qquad \text{(optimal total profit)}$

- What are the characteristics of the feasible combinations of values for the decisions variables? **Constraints** as mathematical relations among decision variables

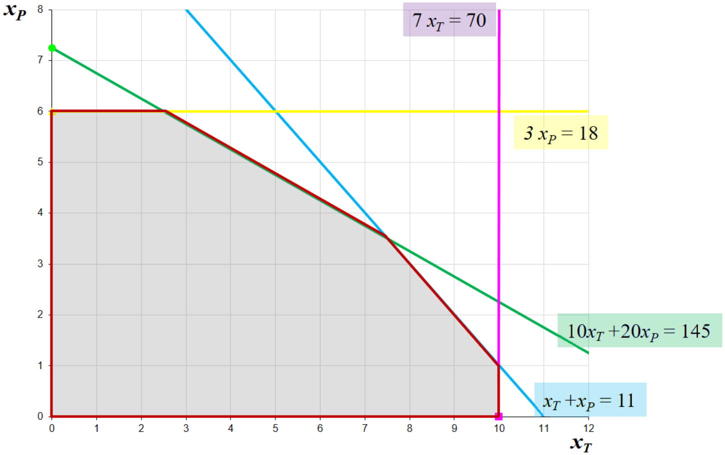
$$\begin{array}{rcrcrl} x_T & + & x_P & \leq & 11 & \text{(land)} \\ 7\,x_T & & & \leq & 70 & \text{(tomato seeds)} \\ & & 3\,x_P & \leq & 18 & \text{(potato tubers)} \\ 10\,x_T & + & 20\,x_P & \leq & 145 & \text{(fertilizer)} \end{array}$$

Using a mathematical model: solution!



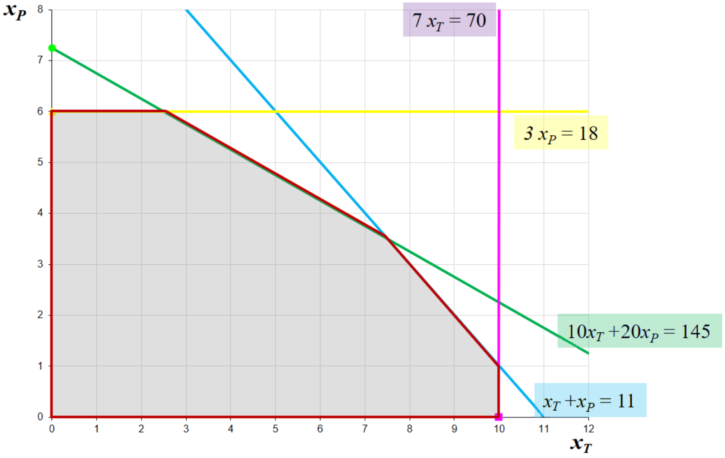
Linear relations: Linear Programming (LP) models

Example: integer variables - exact method



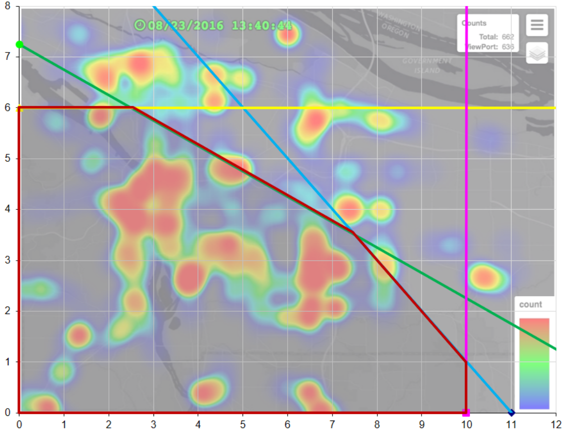
Cutting planes [improved geometry], branch-and-bound [implicit enumeration] (computational resources!)

Example: integer variables - heuristic method



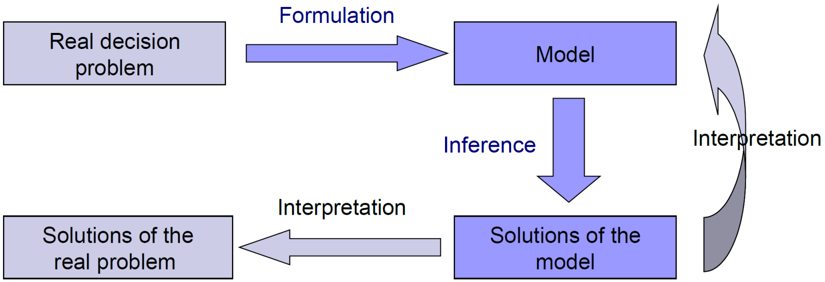
neighborhood search, evolutionary computation etc. to explore a "smart" subset of solutions (limited computational resources required)

Example: a more general combinatorial optimization problem



exact methods may be theoretically and computationally critical, heuristics still work

From decision problem to solution: the Operations Research approach



- **Formulation:** models (mathematical, graph, simulation, game theory), solution representation/perturbation, data driven ...
- **Inference:** quantitative methods, artificial intelligence, efficient algorithms

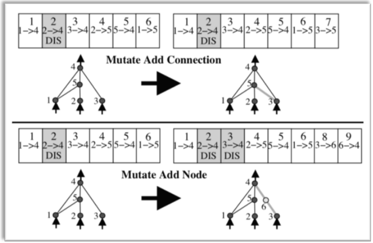
MeMoCO: Preliminary Programme (i)

- Review, advanced topics and application of Linear Programming and Duality
  - Linear Programming models, simplex method, basic notions of duality theory
  - Column generation technique for large size linear programming models
  - Examples: production planning optimization, network flows
- Advanced methods for Mixed **Integer** Linear Programming (MILP)
  - Branch & Bound and relaxation techniques
  - Alternative and strengthened formulations of MILP models
  - Cutting plane methods and Branch & Cut techniques
  - Examples: Travelling Salesman Problem, Facility Location, Set Covering etc.

MeMoCO: Preliminary Programme (ii)

- **Meta-heuristics** for Combinatorial Optimization
  - Neighbourhood search and variants
  - Genetic Algorithms
  - Introduction to hybrid methods data-driven optimization and Matheuristics
- Sample applications and case studies among:
  - Network Optimization: modelling optimization problems on graphs
  - Optimal routing in express freigth delivery
  - Data-driven optimization in Air Traffic Management
  - ..
- **Labs**
  - On-line optimization servers (e.g., NEOS)
  - Optimization software and Algebraic modelling languages (e.g. AMPL, **IBM-OPL**)
  - Optimization libraries (e.g. **IBM Cplex**, Coin-OR, Scip, Google OR-Tools, **python**, Matlab etc.)

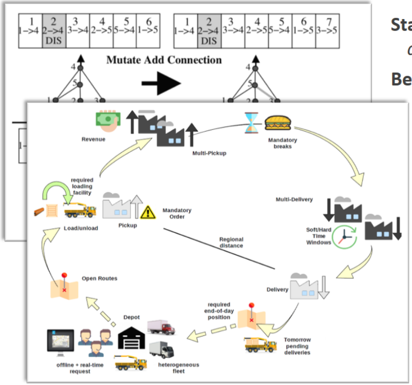
Sample case studies



Stanley et al. 2002, *Evolving Neural Networks through augmenting topologies*, Evol. Comp. Journal

Beccaro 2018, Tabu search approach, Master Thesis

Sample case studies



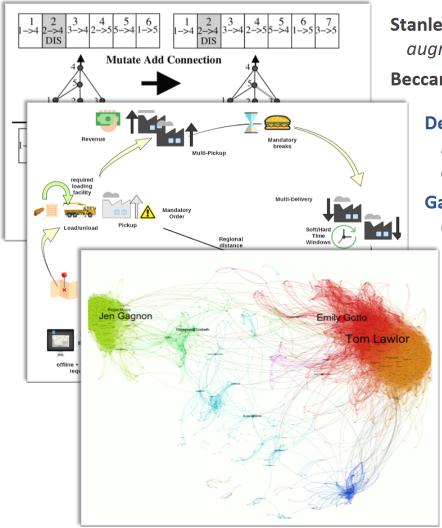
Stanley et al. 2002, *Evolving Neural Networks through augmenting topologies*, Evol. Comp. Journal

Beccaro 2018, Tabu search approach, Master Thesis

De Giovanni et al. 2019, *A two-level local search heuristic for pickup and delivery problems in express freight trucking*, Networks

Gastaldon 2019, Managing dynamic and stochastic demand with machine learning, PhD Thesis

Sample case studies



Stanley et al. 2002, *Evolving Neural Networks through augmenting topologies*, Evol. Comp. Journal

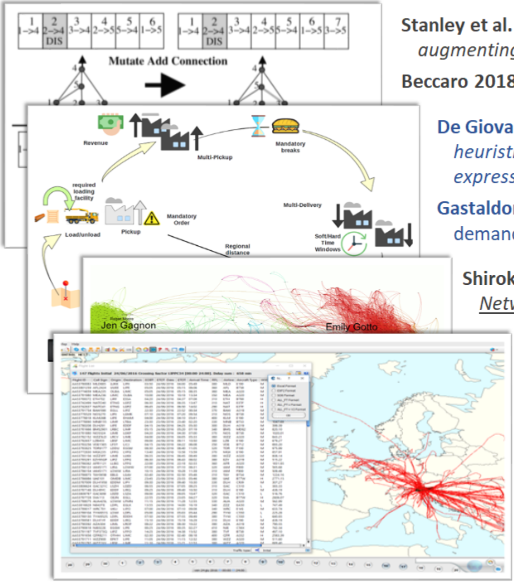
Beccaro 2018, Tabu search approach, Master Thesis

De Giovanni et al. 2019, *A two-level local search heuristic for pickup and delivery problems in express freight trucking*, Networks

Gastaldon 2019, Managing dynamic and stochastic demand with machine learning, PhD Thesis

Shirokikh et al. 2013, *Comb. Opt. Techniques for Network-Based Data Mining*, Handbook of C.O.

Sample case studies



Stanley et al. 2002, *Evolving Neural Networks through augmenting topologies*, Evol. Comp. Journal

Beccaro 2018, Tabu search approach, Master Thesis

De Giovanni et al. 2019, *A two-level local search heuristic for pickup and delivery problems in express freight trucking*, Networks

Gastaldon 2019, Managing dynamic and stochastic demand with machine learning, PhD Thesis

Shirokikh et al. 2013, *Comb. Opt. Techniques for Network-Based Data Mining*, Handbook of C.O.

Bertsimas et al. 2011, *An integer optimization approach to large-scale air traffic flow management*, Op.Res.

De Giovanni et al. 2022, *Data-driven optimization hybridizing machine learning and matheuristics for air traffic flow management with trajectory preferences*, arXiv:2211.06526

Peculiarities and relations to other course units[1]

- Integrated presentation of **diverse optimization techniques**
- Subjects presented with **specific emphasis**. Focus on:
  - **combinatorial** ("discrete", linear) optimization with deterministic settings
  - **engineering aspects**: design and implementation of models and algorithms suitable for real-world applications
  - **comparison and choice** between different approaches
- *State-of-the-art* presentation and insight into several **metaheuristic** techniques (e.g., local search, genetic algorithms etc.)
- Introduction to **hybrid approaches**:
  - mixing paradigms (hybrid metaheuristics)
  - metaheuristics exploiting or supporting exact methods (matheuristics)
  - including machine learning techniques (Data Science for Optimization)

[1] *e.g., "Operations Research", "Optimization", "Optimization for Data Science", "Stochastic Optimization"*

Practical info (i)

- 48 hours (36 lectures + 12 labs, **6 CFU**). First Semester
- **Teaching mode:** classroom or lab + recorded videos (+ streaming?)
- **Moodle**: lecture notes, papers, lab materials, recordings, notices **etc.** *https://stem.elearning.unipd.it/course/view.php?id=7086*
- **Schedule: Thursday and Friday, 2:30 – 4:30 pm**
  - room 2AB45 **or** LabP140 **or** LabTA : **always check!**
- **Learning activities**: Classes, Discussion about case studies, Labs (implementation of mathematical programming models and basic optimization algorithms).

Practical info (ii)

- **Textbooks and learning supports**
  - Lecture notes provided by the teacher + articles from scientific journals (available **before** the class: reading in advance is recommended!)
  - Optimization software packages available on line and in labs (or free student editions).

- **Examination method**
  - **Two lab exercises**: implementation of 1) a MILP model and 2) a metaheuristic (or alternative) algorithm, to be delivered some days before the oral examination (**no** due date during the classes).
    Mandatory [1-10 /30, minimum 5]
  - **Oral examination** on course unit contents.
    Mandatory [1-20 /30, minimum 10]
  - **Lab exercises + Oral examination $\geq$ 18**
  - Short project. Optional [+2 to +6 /30] (e.g., after the oral to improve the score if necessary): modeling and solving a specific problem, even suggested by you; implementing a component of an optimization method etc. (to be agreed with the teacher)