Methods and Models for Combinatorial Optimization

Introduction

Luigi De Giovanni, Marco Di Summa

Dipartimento di Matematica, Università di Padova

Contacts

Luigi De Giovanni

Dipartimento di Matematica
office 427

tel. 049 827 1349

luigi@math.unipd.it

office hours: Thursday, h 10:30 - 12:30
(please, book via email)

Marco Di Summa

Dipartimento di Matematica
office 422

tel. 049 827 1348

disumma@math.unipd.it

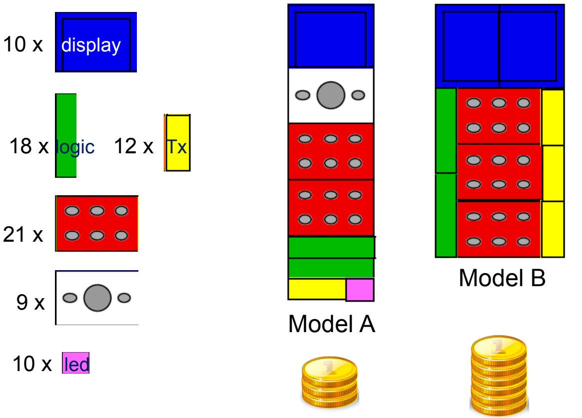
Course webpage

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

Course goals

- Introduction to advanced modelling and solution techniques for combinatorial optimization problems in decision supporting.
- 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.
- Find, understand, implement state-of-the-art approaches to solve real combinatorial optimization problems

Combinatorial optimization problem: example 1



Objective

Decision: How many A, 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?

- Quantum computing? Still "not operational..."
- Now and in the next future: **MeMoC(O)**

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?

- Quantum computing? Still "not operational..."
- Now and in the next future: **MeMoC(O)**

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?

- Quantum computing? Still "not operational..."
- Now and in the next future: **MeMoC(O)**

Combinatorial optimization problems: example 2

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 net gain of 3000 Euros per hectare of tomatoes and 5000 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

  $\max 3000\,x_T + 5000\,x_P$

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

$$\begin{array}{rcrcrl} x_T & + & x_P & \leq & 12 & \text{(land)} \\ 7\,x_T & & & \leq & 70 & \text{(tomato seeds)} \\ & & 3\,x_P & \leq & 18 & \text{(potato tubers)} \\ 10\,x_T & + & 20\,x_P & \leq & 160 & \text{(fertilizer)} \end{array}$$

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

  $\max 3000\, x_T + 5000\, x_P$

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

$$
\begin{array}{rcrcrl}
x_T & + & x_P & \leq & 12 & \text{(land)} \\
7\, x_T & & & \leq & 70 & \text{(tomato seeds)} \\
 & & 3\, x_P & \leq & 18 & \text{(potato tubers)} \\
10\, x_T & + & 20\, x_P & \leq & 160 & \text{(fertilizer)}
\end{array}
$$

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

  $\max 3000\, x_T + 5000\, x_P$

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

$$
\begin{array}{rcrcrl}
x_T & + & x_P & \leq & 12 & \text{(land)} \\
7\, x_T & & & \leq & 70 & \text{(tomato seeds)} \\
 & & 3\, x_P & \leq & 18 & \text{(potato tubers)} \\
10\, x_T & + & 20\, x_P & \leq & 160 & \text{(fertilizer)}
\end{array}
$$

Using a mathematical model: formulation

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

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

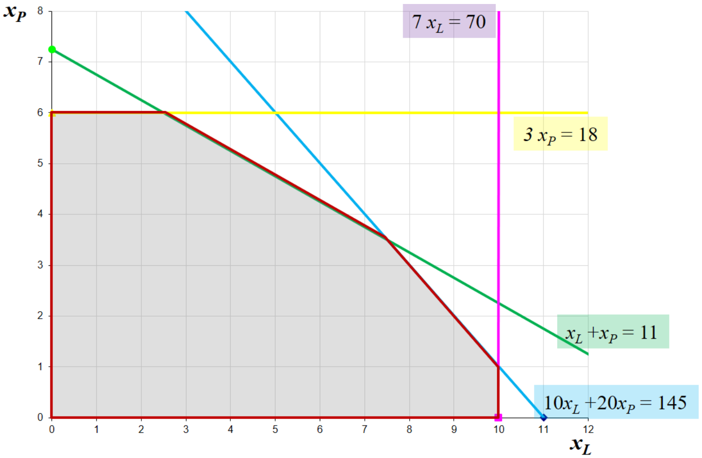
- What should be optimized? **Objective** as a function of the decision variables

  $\max 3000\, x_T + 5000\, x_P$

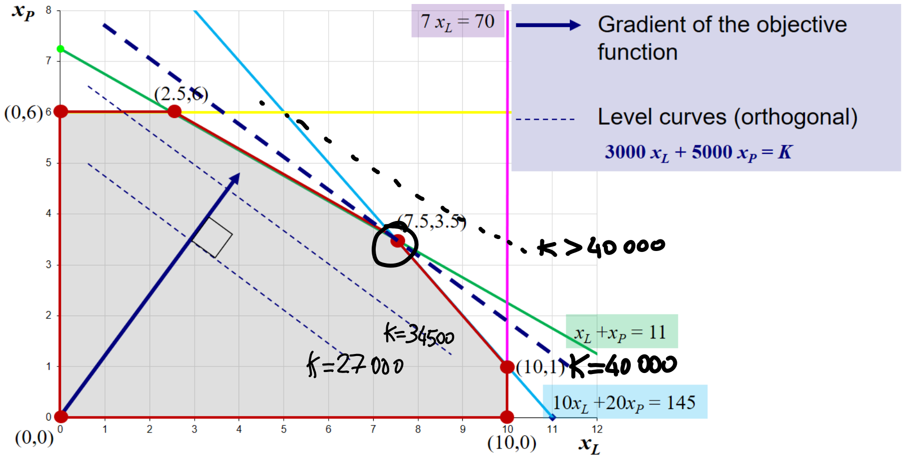
- What are the characteristics of the feasible combinations of values for the decisions variables? **Constraints** as mathematical relations among decision variables

$$\begin{array}{rcrcrl} x_T & + & x_P & \leq & 12 & \text{(land)} \\ 7\, x_T & & & \leq & 70 & \text{(tomato seeds)} \\ & & 3\, x_P & \leq & 18 & \text{(potato tubers)} \\ 10\, x_T & + & 20\, x_P & \leq & 160 & \text{(fertilizer)} \end{array}$$

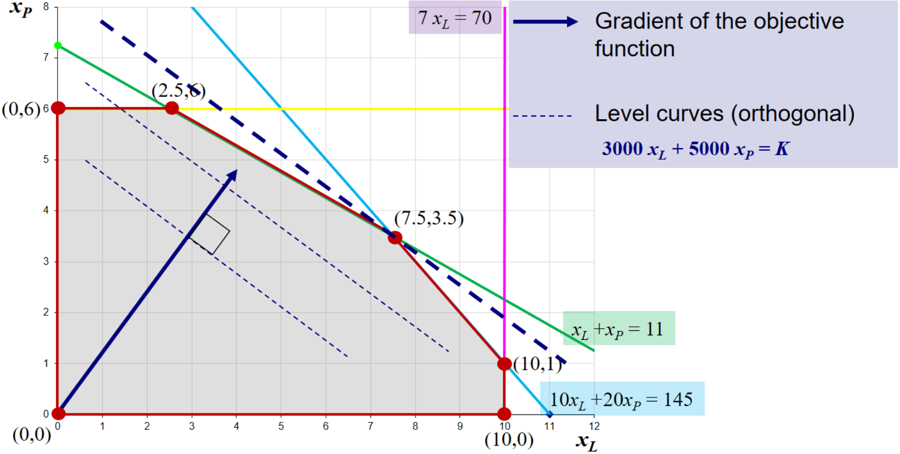
Using a mathematical model: solution!



Using a mathematical model: solution!

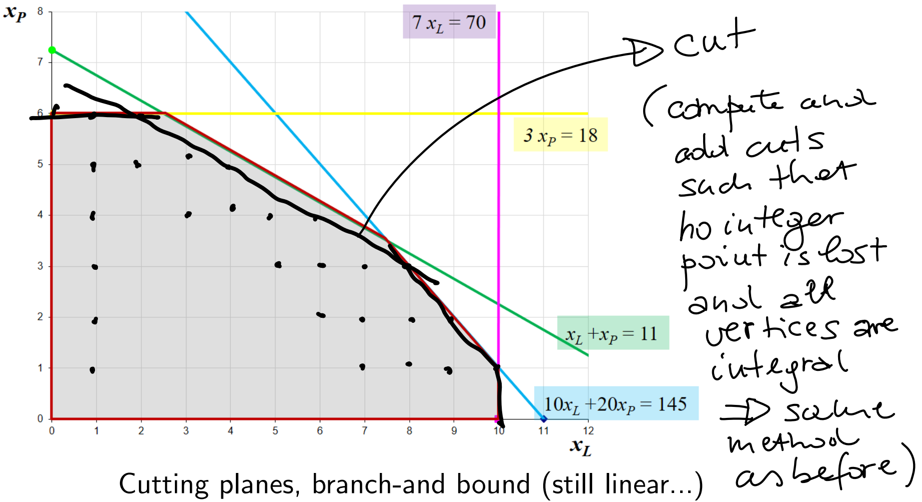


Using a mathematical model: solution!



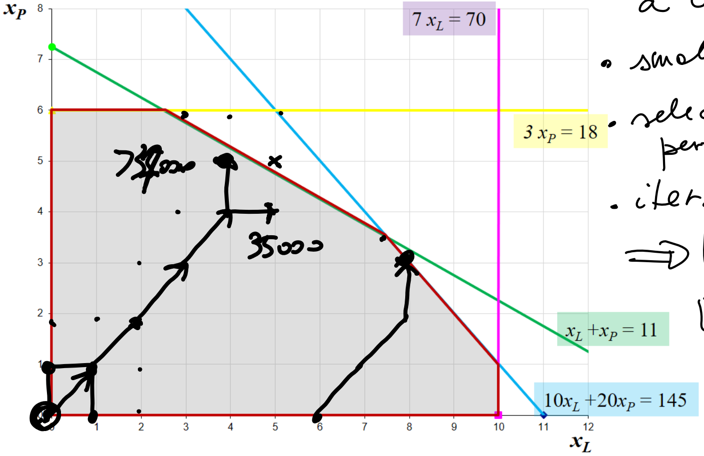
Linear relations: Linear Programming (LP) models

Example: integer variables - exact method



Cutting planes, branch-and bound (still linear...)

Example: integer variables - heuristic method



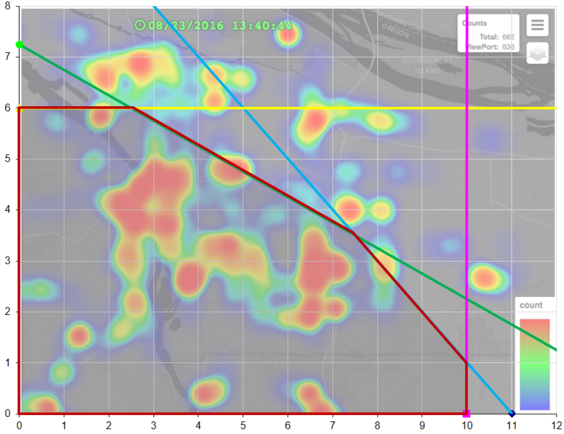
- start from a combination
- small perturbations
- select best perturbation
- iterate

⇒ local optimum

[start from different combinations ⇒ other (better) local optima]

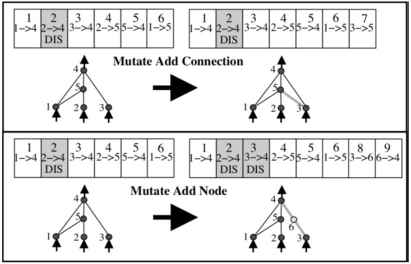
neighborhood search, evolutionary computation etc.

Example: a more general combinatorial optimization problem



exact methods (critical) heuristic (still works)

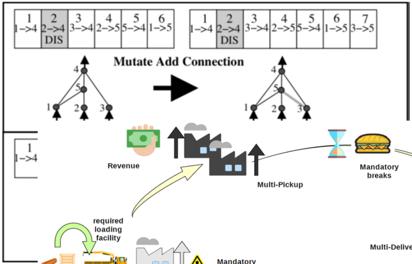
Example: applications

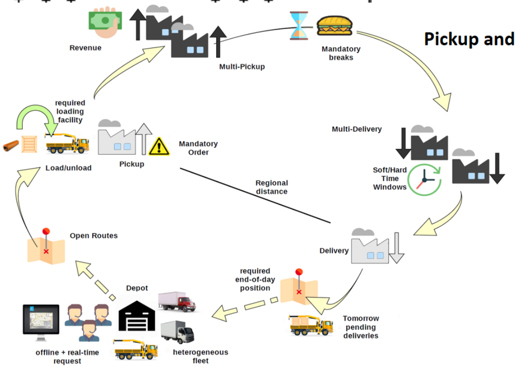


Stanley et al. 2002, Evolving Neural Networks through Augmenting Topologies, Evolutionary Computation Journal

Beccaro et al. 2018, Tabu searc approach, Master Thesis

Example: applications



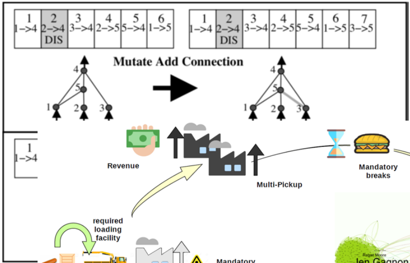


Stanley et al. 2002, Evolving Neural Networks through Augmenting Topologies, Evolutionary Computation Journal

Beccaro et al. 2018, Tabu searc approach, Master Thesis

Pickup and delivery routes in freight transportation

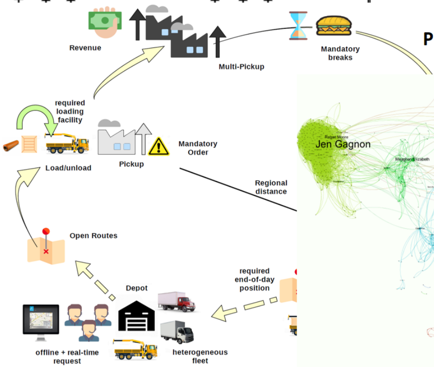
Example: applications

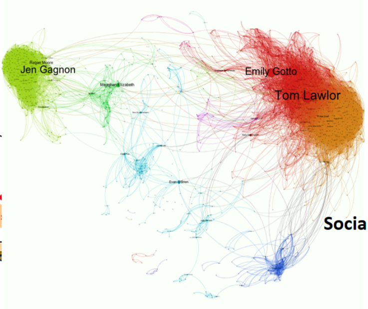


Stanley et al. 2002, Evolving Neural Networks through Augmenting Topologies, Evolutionary Computation Journal

Beccaro et al. 2018, Tabu searc approach, Master Thesis

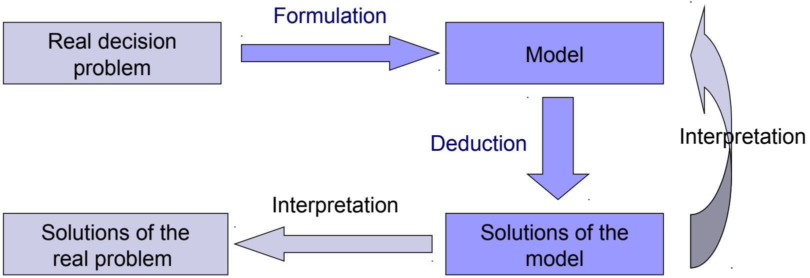
Pickup and delivery routes in freight transportation





Social Network Analysis

From decision problem to solution: the Operations Research approach



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

MeMoCO: Preliminary Programme

- Review, advanced topics and application of LP and Duality
  - LP models, simplex method, basic notions of duality theory
  - Column generation technique for large size linear programming models
  - Applications: production planning optimization, network flows
- Advanced methods for Mixed **Integer** Linear Programming (MILP)
  - Alternative formulations, Branch & Bound, Branch & Cut
  - Applications: TSP, Facility Location, Set Covering etc.
- **Meta-heuristics** for Combinatorial Optimization
  - Neighbourhood search and variants
  - Genetic Algorithms
- Relevant applications
  - Network Optimization: modelling optimization problems on graphs
  - Data driven optimization (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)

Practical info

- **Schedule**:
  - Thursday, Friday 8:30 - 10:30
  - room 1BC50 **or** LabTA (**always check**!)
- **Textbooks and course material**
  - Lecture notes provided by the teacher + articles from scientific journals (available **before** the class: read them!)
  - Optimization software packages available on line and in labs
  - *http://www.math.unipd.it/∼luigi/courses/metmodoc/metmodoc.html*
- **Examination methods**
  - **Two lab exercises**: implementation of 1) a MILP model and 2) a metaheuristic, to be delivered some days before the oral examination.
    Mandatory [1-10 /30, minimum 5]
  - **Oral examination** on course contents.
    Mandatory [1-20 /30, minimum 10].
  - Short project. Discretionary [+2 to +6 /30] (e.g., after the oral to improve the score if necessary)

Practical info

- **Schedule**:
  - Thursday, Friday 8:30 - 10:30
  - room 1BC50 **or** LabTA (**always check**!)
- **Textbooks and course material**
  - Lecture notes provided by the teacher + articles from scientific journals (available **before** the class: read them!)
  - Optimization software packages available on line and in labs
  - *http://www.math.unipd.it/∼luigi/courses/metmodoc/metmodoc.html*
- **Examination methods**
  - **Two lab exercises**: implementation of 1) a MILP model and 2) a metaheuristic, to be delivered some days before the oral examination. **Mandatory [1-10 /30, minimum 5]**
  - **Oral examination** on course contents. **Mandatory [1-20 /30, minimum 10]**.
  - Short project. Discretionary [+2 to +6 /30] (e.g., after the oral to improve the score if necessary)

Practical info

- **Schedule**:
  - Thursday, Friday 8:30 - 10:30
  - room 1BC50 **or** LabTA (**always check**!)
- **Textbooks and course material**
  - Lecture notes provided by the teacher + articles from scientific journals (available **before** the class: read them!)
  - Optimization software packages available on line and in labs
  - *http://www.math.unipd.it/~luigi/courses/metmodoc/metmodoc.html*
- **Examination methods**
  - **Two lab exercises**: implementation of 1) a MILP model and 2) a metaheuristic, to be delivered some days before the oral examination.
    Mandatory [1-10 /30, minimum 5] (NO FIXED DUE DATE)
  - **Oral examination** on course contents.
    Mandatory [1-20 /30, minimum 10].
  - Short project. Discretionary [+2 to +6 /30] (e.g., after the oral to improve the score if necessary)