Capacitated Vehicle Routing (CVRP)

Input:
- complete graph $G(V, A)$, where $V = \{0, \ldots, n\}$
- vertices $i = 1, \ldots, n$ are customers that must be visited
- vertex $i = 0$ is the single depot
- arc/edges have associated a cost $c_{ij}$ ($c_{ik} + c_{kj} \geq c_{ij}, \forall i, j \in V$)
- customers have associated a non-negative demand $d_i$
- a set of $K$ identical vehicles with capacity $C$ ($d_i \leq C$)

Task: Find collection of $K$ circuits with minimum cost, defined as the sum of the costs of the arcs of the circuits and such that:
- each circuit visit the depot vertex
- each customer vertex is visited by exactly one circuit; and
- the sum of the demands of the vertices visited by a circuit does not exceed the vehicle capacity $C$.

Lower bound to $K$: $K \geq K_{\text{min}}$ where $K_{\text{min}}$ is the number of bins in the associated Bin Packing Problem
CVRP

Construction Heuristics
- Nearest neighbors
- Savings heuristics (Clarke and Wright)
- Insertion heuristics
- Route-first cluster-second
- Cluster-first route-second
  - Sweep algorithm
  - Generalized assignment
  - Location based heuristic
  - Petal algorithm

Perturbative Search
- Solution representation: sets of integer sequences, one per route
- Neighborhoods structures:
  - intra-route: 2-opt, 3-opt
  - inter-routes: λ-interchange, relocate, exchange, CROSS, ejection chains, GENI

The Timetabling Activity
Assignment of events to a limited number of time periods and locations subject to constraints
Two categories of constraints:
Hard constraints $H = \{H_1, \ldots, H_n\}$: must be strictly satisfied, no violation is allowed
Soft constraints $\Sigma = \{S_1, \ldots, S_m\}$: their violation should be minimized

Typologies of timetabling
- Educational Timetabling,
  - Class model
  - Exams model
  - Courses model
- Employee Timetabling,
  - Crew Scheduling
  - Crew Rostering
- Transport Timetabling,
- Sports Timetabling,
- Communication Timetabling

Outline
1. Capacited Vehicle Routing
2. Timetabling
   Educational Timetabling
3. Optimization under Uncertainty
4. Multi-Objective Optimization
5. Conclusive Notes
An Example in Practice

University Course Timetabling (UCTP)
Find an assignment of lectures to periods and rooms which is feasible.

- Rooms are only used by one lecture at a time,
- Each lecture is assigned to a suitable room,
- No student has to attend more than one lecture at once;

and good

- Not more than two lectures in a row for a student,
- Unpopular periods avoided (last in a day),
- Students do not have one single lecture in a day.

Solution Approaches

- Typically solved in two phases: first a feasible solution satisfying all hard constraints is found and then the soft constraints are considered. Issue: exiting or not from feasibility?
- Handling the soft constraints gives rise to a multiobjective problem. Three approaches:
  - Combine objectives: combination of penalties into a single value by means of weights
  - Alternating objectives (lexicographic or multi-phased approaches): optimize one objective at a time while imposing constraints on the others.
  - Pareto-based: the whole Pareto-frontier of trade off solutions is determined
- The classroom assignment sub-problem can be solved efficiently if each period is considered independently.
- The model varies from institution to institution, hence heuristic methods are adjusted to the specific constraints and requirements and also their density.

Local Search Methods

- Basic components: Construction, Iterative Improvement, Metaheuristics
- Assemblage
- Tabu Search
- Simulated Annealing
- Iterated Local Search
- Iterated Greedy
- Ant Colony Optimization
- Evolutionary Algorithm

Set of candidate assignments

- Graph coloring model, though it does not comprise the assignment of rooms
- Matrix representation: it comprises the assignment of rooms
Initialization Function: Construction Heuristics

Graph Coloring DSATUR inspired
It consists of two main passages:

- Deciding the next lecture to schedule
- Deciding a place in the assignment matrix for the selected lecture

For example:

1. **Step 1.** Initialize the set $\hat{L}$ of all unscheduled lectures with $\hat{L} = L$.
2. **Step 2.** Pick the lecture $L_i \in \hat{L}$ with fewest feasible places (or minimal violations of $H$) in the assignment matrix.
3. **Step 3.** Let $\hat{X}$ be the set of all feasible positions for $L_i$ in the assignment matrix (or those positions in which the number of violations of constraints $H$ is minimal).
4. **Step 4.** Let $\bar{X} \subseteq \hat{X}$ be the subset of positions of $\hat{X}$ in which the weighted sum of the cost due to violations of the soft constraints $\Sigma$ is minimal.
5. **Step 5.** Choose a place for $L_i$ from $\bar{X}$. Update information.
6. **Step 6.** Remove $L_i$ from $\hat{L}$, and go to step 2 until $\hat{L}$ is not empty.

Outline

1. Capacitated Vehicle Routing
2. Timetabling
   - Educational Timetabling
3. Optimization under Uncertainty
4. Multi-Objective Optimization
5. Conclusive Notes

Neighborhood Structures for Iterative Improvement

- $N_1$: One Exchange
- $N_2$: Swap
- $N_3$: Period Swap
- $N_4$: Kempe Chain Interchange
Optimization Under Uncertainty

In many real life cases problem data might be uncertain.

Approaches (in decreasing order of information available):

▶ stochastic optimization
▶ dynamic optimization
▶ robust optimization
▶ online optimization

But other factors are relevant to determine the optimization approach.

Solutions in Problems under Uncertainty

▶ A priori solutions
▶ On line solutions
▶ Mixed strategy solutions (two-stage)

Two-stage solutions consists of:

▶ A first stage where some action is taken
▶ A second stage (made by recourse decisions) that compensates for any bad effects that might have been experienced as a result of the first-stage decision and the random outcome of events.

The optimal policy from such a model is a single first-stage policy and a collection of recourse decisions (a decision rule) defining which second-stage action should be taken in response to each random outcome.

Stochastic optimization

If probability distributions governing the data are known or can be estimated then it is possible to take advantage of this.

In stochastic optimization the goal is to find some policy that is feasible for all (or almost all) the possible data instances and maximizes the expectation of some function of the decisions and the random variables.

Solution approaches

▶ The expected value of the objective function can be computed mathematically.

\[ g(s) = E[f(\pi, s)] \]

where \( s \in S \) and \( \pi \) are the stochastic variables determining the possible scenarios.

Then no difference with a deterministic problem. Although the evaluation function may become computationally prohibitive.

▶ Alternatively, the expected value of the objective function can be estimated via sampling and simulation

\[ g(s) = \bar{f}(\pi, s) = \frac{1}{N} \sum_{l=1}^{N} f(\pi_l, s) \quad \text{(unbiased estimator)} \]
Robust Optimization

If only uncertainty sets rather than probability distributions are known. Unlike dynamic and stochastic programming this robust approach does not suffer from the curse of dimensionality.

The goal robust optimization is to find a solution which is feasible for all such data and optimal in some sense. For example, it optimizes for the worst-case scenario. Let the uncertain MP be given by

$$\min_{x} f(x; \pi) : x \in X(\pi)$$

where $$\Pi$$ is some set of scenarios (like parameter values). The robust optimization model is:

$$\min_{x} \{ \max_{\pi \in \Pi} f(x; \pi) \} : x \in X(\pi') \forall \pi' \in \Pi,$$

The policy $$x$$ is required to be feasible no matter what parameter value (scenario) occurs; hence, it is required to be in the intersection of all possible $$X(\Pi)$$. The inner maximization yields the worst possible objective value among all scenarios.

Multi-Objective Combinatorial Optimization Problems

A Multi-Objective Combinatorial Optimization Problem has a vector as objective function $$\vec{f} = (f_1, \ldots, f_p)$$. Example

Given a undirected graph $$G(V, E)$$ with weights $$\vec{d}(uv) \in \mathbb{R}^p$$ for each edge $$uv \in E$$. Find an Hamiltonian cycle $$H$$ such that

$$\vec{f}(H) = \sum_{uv \in H} \vec{d}(uv)$$

is “minimal”.

Elements of a Multiobjective Optimization Problem

- the set of feasible candidate solutions $$\mathcal{S}$$
- the objective function vector $$\vec{f} = (f_1, \ldots, f_p) : \mathcal{S} \rightarrow \mathbb{R}^p$$
- the objective space $$\mathbb{R}^p$$
- the ordered space $$(\mathbb{R}^P, \preceq)$$
- the model map $$\theta : \mathbb{R}^P \rightarrow \mathbb{R}^P$$

Outline

1. Capacitated Vehicle Routing
2. Timetabling
   - Educational Timetabling
3. Optimization under Uncertainty
4. Multi-Objective Optimization
5. Conclusive Notes
Classification

We distinguish four types of Multiobjective optimization problems according to the requirements:

▶ min-max optimality \((\theta : \mathbb{R}^p \rightarrow \mathbb{R})\)
▶ weighted sum optimality \((\theta : \mathbb{R}^p \rightarrow \mathbb{R})\)
▶ lexicographic optimality \((\theta : \mathbb{R}^p \rightarrow \mathbb{R})\)
▶ Pareto optimality \((\theta = I : \mathbb{R}^p \rightarrow \mathbb{R}^p)\)

Min-Max Optimality

\[
\min_{s \in S} \max_{i=1,...,p} f_i(s)
\]

Weighted Sum Optimality

\[
\min f_w(s) = \sum_{i=1}^{p} w_i f_i(s)
\]

Lexicographic optimality

<table>
<thead>
<tr>
<th>Name</th>
<th>Notation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>lexicographic order</td>
<td>(\vec{y}^1 &lt;_{\text{lex}} \vec{y}^2)</td>
<td>(\exists i \in {1, \ldots, p-1} \mid y^1_k = y^2_k) \forall k = 1, \ldots, i \text{ and } y^1_{i+1} &lt; y^2_{i+1})</td>
</tr>
</tbody>
</table>

A solution \(s \in S\) is lexicographic optimal if there is no \(s' \in S\) such that \(\vec{f}(s') <_{\text{lex}} \vec{f}(s)\).

\[
\text{lexmin}_{s \in S} \{f_1(s), \ldots, f_p(s)\}
\]

\(\Rightarrow\) Solve objective sequentially by decreasing order of priority and using the optimal solutions of higher priority objectives as constraints (goal programming).

Pareto Optimality

Needed some definitions on dominance relations

In Pareto sense, for two points (vectors) in \(\mathbb{R}^p\)

\(\vec{y}^1 \leq \vec{y}^2\) weakly dominates \(\vec{y}^1 \leq \vec{y}^2\) for all \(i = 1, 2, \ldots, p\)

\(\vec{y}^1 \parallel \vec{y}^2\) incomparable neither \(\vec{y}^1 \leq \vec{y}^2\) nor \(\vec{y}^2 \leq \vec{y}^1\)

Hence a set of solutions yields a set of mutually incomparable points (i.e., weakly non-dominated points)

A feasible solution \(s \in S\) is called efficient or Pareto global optimal if there is no other \(s' \in S\) such that \(\vec{f}(s') \leq \vec{f}(s)\).
Craft, Art or Science?

Science

Heuristics are pervasive in many areas: Complete Search Algorithms such as Constraint Programming, Integer Programming, but also, in Philosophy.

- which is the heuristic which works best in a given context?
- why is it so?

The final goal is not practical but intellectual. Iterated process:
- observe
- formulate hypothesis
- experiment
- build models and theories (evaluated on their explanatory and predictive power and conceptual simplicity). [A model is never identical with what it models, is a heuristic device to enable understanding of what it models.]

In other areas like Social Sciences, Behavioral Finance and Cognitive Psychology if they are understood they can be used to explain human behavior [Tversky and Kahneman].

Future Directions in Heuristics for Optimization

- Empirical analysis of heuristic algorithms behavior
  - Computational studies
  - Statistical methodologies
  - Comparison of different methods for configuration and tuning
  - Applications of extreme value theory
  - Comparison of development frameworks
- Problem characteristics, search space structure and metaheuristic behavior
- Theoretical results and foundations
  - probabilistic analysis
  - worst case analysis
  - convergence, stochastic processes
  - derandomization
- New heuristic methods
  - mainly hybridization with exact methods
- New applications
  - multiobjective optimization
  - stochastic problems
  - robust optimization