Routing Problems with Loading Constraints

(with an introduction to Vehicle Routing)

Silvano Martello

DEI "Guglielmo Marconi", Università di Bologna

Workshop on Traffic Optimization, Heidelberg, October 2015

Summary

1. (Vertex) **Routing Problems**: A brief history and overview;

2. **Packing** (and Loading) **problems**:

   One-, Two-, and Three-dimensional packing;

3. A combination: **Routing problems with loading constraints**.

1. Routing problems

The origins: circuits on graphs

SOLUTION

D'UNE

QUESTION CURIEUSE QUI NE PAROIT SOUMISE À AUCUNE ANALYSE,

PAR M. EULER.

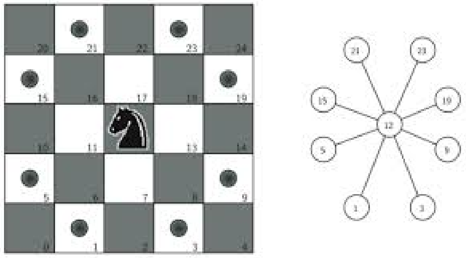
1.

Je me trouvai un jour dans une compagnie, où, à l'occaſion du jeu d'echecs quelqu'un propoſa cette queſtion: *de parcourir avec un cavalier toutes les caſes d'un échiquier, ſans parvenir jamais deux fois à la même, & en commençant par une caſe donnée.* On mettoit pour cette fin des jettons ſur toutes les 64 caſes de l'échiquier, à l'exception de celle où le Cavalier devoit commencer ſa route; & de chaque caſe où le Cavalier paſſoit conformément à ſa marche, on ôtoit le jetton, de ſorte qu'il s'agiſſoit d'enlever de cette façon ſucceſſivement tous les jettons. Il faloit donc éviter d'un côté, que le cavalier ne revint jamais à une caſe vuide, & d'un autre côté il faloit diriger en ſorte ſa courſe, qu'il parcourut enfin toutes les caſes.

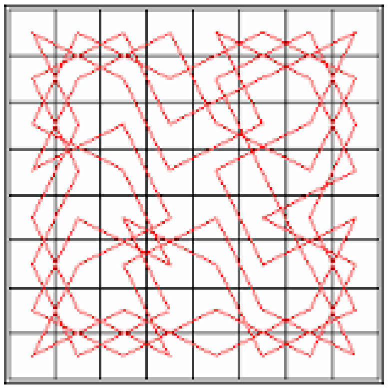
2. Ceux qui croyoient cette queſtion aſſez aiſée firent pluſieurs eſſais inutiles ſans pouvoir atteindre au but; après quoi celui qui avoit propoſé la queſtion, ayant commencé par une caſe donnée, a ſçu ſi bien diriger la route, qu'il a heureuſement enlevé tous les jettons. Cependant la multitude des caſes ne permettoit pas qu'on ait pû imprimer à la mémoire la route qu'il avoit ſuivie; & ce n'étoit qu'après pluſieurs eſſais, que j'ai enfin rencontré une telle route, qui ſatisfit à la queſtion; encore ne valoit-elle que pour une certaine caſe initiale. Je ne me ſouviens plus, ſi on lui a laiſſé la liberté de la choiſir lui-même; mais il a très poſitivement aſſuré qu'il étoit en état de l'éxécuter, quelle que ſoit la caſe où l'on voulut qu'il commençat.

3.

- **Leonard Euler** (1759): *Solution d'une question curieuse qui ne paroit soumise à aucune analyse*. **Knights tour problem**:
  - a knight is placed on an empty $n \times n$
    chessboard and must visit each square exactly once by only using valid chess moves of a knight:



  - by defining a graph in which the vertices correspond to the chessboard squares and the edges to the legal knight moves, a knight's tour corresponds to a path (or a cycle) that visits every vertex of the graph exactly once:

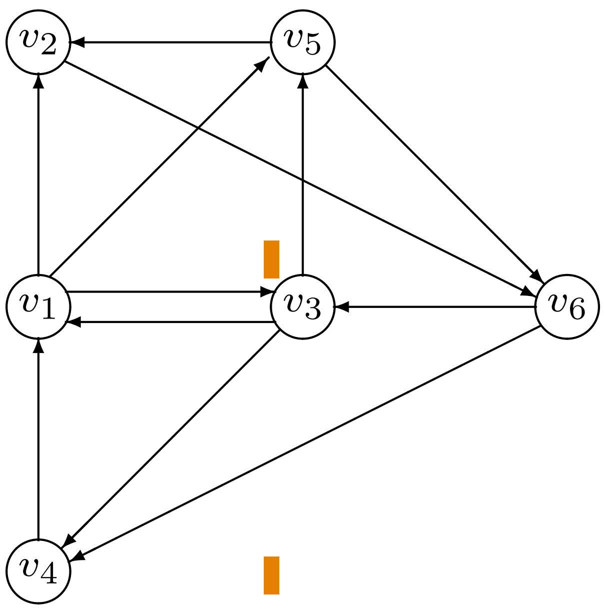


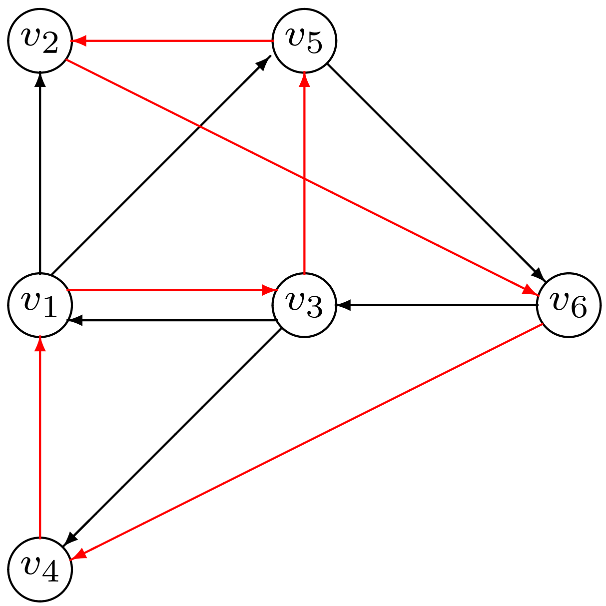
- **Sir William Rowan Hamilton**, Irish mathematician (1859): **Icosian Game**
  - played on a wooden planar representation of the edges of a dodecahedron (a graph), with holes at each of the twenty vertices:

  - The first player stuck five pegs in any consecutive vertices, and the second player was requested to stick the remaining fifteen pegs so as to complete the resulting path to a cycle visiting each vertex exactly once;
  - a circuit that passes through each vertex exactly once is called a **Hamiltonian circuit**;
  - sold for £25 to a Dublin toy manufacturer. (It seems that the sales were not satisfactory though.)

Hamiltonian circuits and the Traveling Salesman Problem

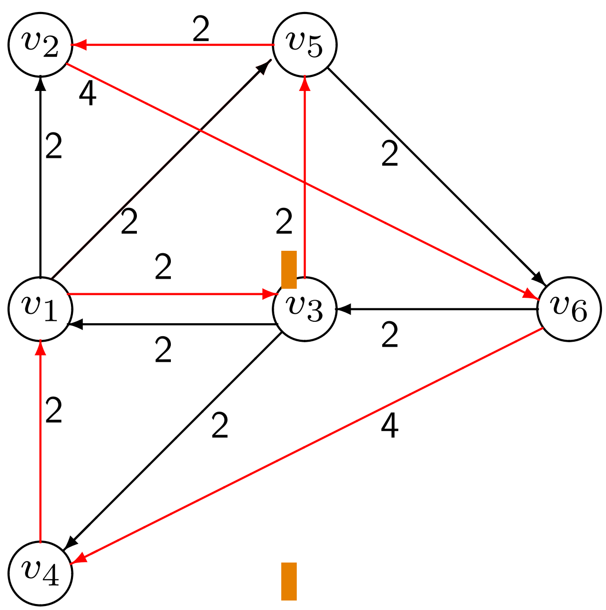
Given a directed graph $G = (V, A)$, $V = \{v_1, \ldots, v_n\}$, $A = \{(v_i, v_j) : v_i, v_j \in V\}$
or an undirected graph $G = (V, E)$, $E = \{(v_i, v_j) \equiv (v_j, v_i) : v_i, v_j \in V\}$
decide if the graph possesses a Hamiltonian circuit:



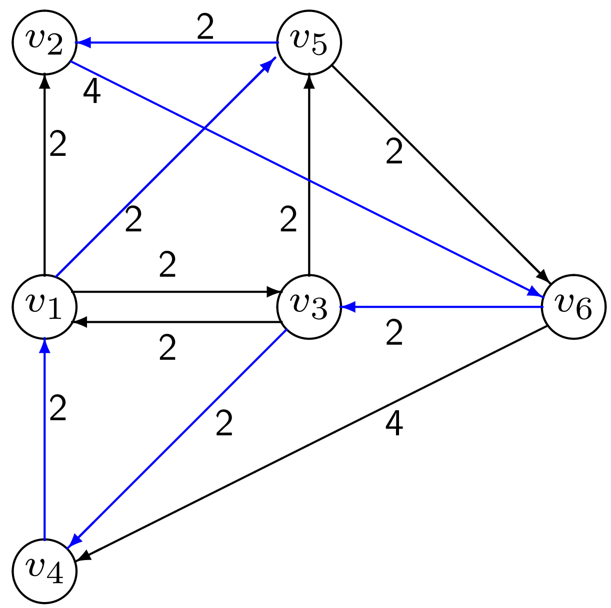


The problem can be solved by enumerating all permutations of the vertices and checking each of the for feasibility: This will take a time proportional to $(n-1)!$ (exponential, impractical) **but** the problem is $\mathcal{NP}$**-complete in the strong sense**: most likely it will never be possible to solve it in a time that grows polynomially with n.

If the graph has **weights** (costs, lengths, times, ...) associated with the arcs/edges, the **Traveling Salesman Problem (TSP)** is to find the **Hamiltonian circuit of minimum total weight**:



cost = 16



cost = 14

Book published in 1931 in German: *The Traveling Salesman Problem, how he should be and what he should do to be successful in his business. By a veteran traveling salesman.*

Generalization of the Hamiltonian circuit problem $\Longrightarrow$ $\mathcal{NP}$**-hard in the strong sense**.

Applications in freight transportation using a **single vehicle**. **BUT**

Real world applications use a **fleet of vehicles** based at **one or more depots**, and have **specific constraints**.

Vehicle Routing Problems

- Optimal delivering of goods, using a set of **vehicles**, based at one or more **depots**, through a **road network**.
- **Freight transportation**
  - takes 10% - 25% of the final cost for consumer goods;
  - use of optimization techniques $\Longrightarrow$ savings of 5% - 20% of the total transportation costs.
- General **Vehicle Routing Problem (VRP):** ***Find a set of routes, each assigned to a vehicle that starts and ends at its depot, so that a set of specific operational constraints is satisfied and the total transportation cost is minimized.***
- Generalization of the Traveling salesman problem $\Longrightarrow$ $\mathcal{NP}$**-hard in the strong sense**.
- Different models according to the considered constraints. Most models: **undirected graphs**

  Basic (simplest) VRP: **Capacitated Vehicle Routing Problem (CVRP)**:
  - vertex 0=depot; vertices i $(i = 1, \ldots, n)$=customers; edge costs $c_{ij} (i, j = 0, 1, \ldots, n)$;
  - customer i requests goods of total weight d_i $(i = 1, \ldots, n)$;
  - each customer must be visited by a single vehicle;
  - K vehicles having capacity D (or having capacities D_k $(k = 1, \ldots, K)$);
  - the total weight on each vehicle must be $\leq D$ (or on vehicle k must be $\leq D_k$ $\forall$ k).
  - minimize the total cost of the routes.

Vehicle Routing Problems (cont'd)

- **Distance-Constrained CVRP**: like CVRP, with an additional constraint:
  - t_{ij} = traveling time of edge (i, j) (either coinciding with c_{ij}, or not);
  - T_k = maximum total traveling time for vehicle k $(k = 1, \ldots, K)$.
- **VRP with Time Windows**: like CVRP, with an additional constraint:
  - t_{ij} = traveling time of edge (i, j);
  - for each customer i,
    - s_i = time needed to download goods;
    - $[a_i, b_i]$ = time window within which the delivery must start.
- **VRP with Backhauls**: like CVRP, with an additional constraint:
  - the customers are partitioned into two sets:
    - *linehaul customers* to whom goods are to be delivered;
    - *backhaul customers* whose goods need to be transported back to the depot;
  - each vehicle must visit all its linehaul customers before all its backhaul customers;
  - for each vehicle k,

    $\max\{$(total linehaul weight),(total backhaul weight)$\} \leq D$ (or $\leq D_k \ \forall\, k$).

Vehicle Routing Problems (cont'd)

- **VRP with Pickup and Delivery**: like CVRP, with an additional constraint:
  - for each customer i:
    - d_i = quantity of goods to be delivered to the customer;
    - p_i = quantity of goods to be picked up at the customer;
  - when the vehicle arrives, it first downloads then uploads;
  - for each vehicle k the total carried weight at any time must be $\leq D$ (or $\leq D_k \;\; \forall\, k$).
- **VRP with Loading Constraints**: like CVRP, with an additional constraint:
  - for each customer i:
    - list of the packages to be delivered, with their dimensions (height, width, length);
  - for each vehicle k: dimensions of the loading area (height, width, length);
  - the solution must provide, for each vehicle, the packing in the loading area.
- **Same constraints and variants studied for the TSP**
- **Combinations of the various constraints** (e.g., backhauls + time windows + ...)
- **Variants** (multiple depots, customers served by more vehicles, special compartments, ...)
- All **strongly $\mathcal{NP}$-hard**.

Bibliography

- **Hamiltonian circuits:**
  S. Martello. An enumerative algorithm for finding Hamiltonian circuits in a directed graph.
  ACM Transactions on Mathematical Software (1983) **freeware**

- **Traveling salesman problem:**
  E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, and D.B Shmoys, eds. *The traveling salesman problem*.
  John Wiley & Sons, Chichester (1985)

  G Carpaneto, M Dell'Amico, P Toth. Exact solution of large-scale, asymmetric traveling salesman problems
  ACM Transactions on Mathematical Software (1995) **freeware**

  C. D'Ambrosio, A. Lodi, S. Martello. Combinatorial traveling salesman problem algorithms.
  Wiley Encyclopedia of Operations Research and Management Science (2011)

- **Vehicle routing problems:**
  P.Toth and D.Vigo, eds. *Vehicle Routing: Problems, Methods, and Applications*
  SIAM (2014)

2. Packing problems

One-dimensional bin packing problem

A geometrical description:

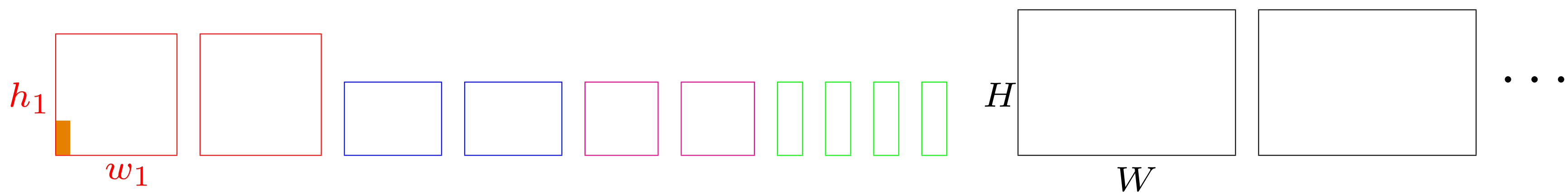
- given n **segments (items)** having width w_j $(j = 1, \ldots, n)$, and an unlimited number of identical **large segments (bins)** having width W,

pack all the segments, without overlapping, into the **minimum number of bins**:

- The problem is $\mathcal{NP}$**-hard in the strong sense**.
- **huge literature**

Two-dimensional bin packing problem (2BP)

- given n **rectangles** (**items**), having width w_j and height h_j $(j = 1, \dots, n)$,

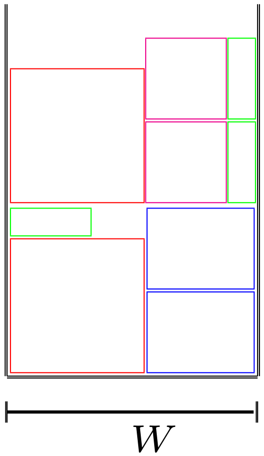


- and an unlimited number of large rectangles (**bins**), having width W and height H,
- **A. pack all the items**, without overlapping, into the **minimum number of bins**:

- **B. pack a subset of items**, without overl., in a **single bin by maximizing the packed area**.
- **Many variants** $\Leftarrow$ applications, cutting problems from standard stock pieces (wood, glass,...):
  - the items may/may not be **rotated**; by **90°**/by **any angle**;
  - **guillotine cutting** may/may not be imposed (items must be obtained through a sequence of edge-to-edge cuts parallel to the edges of the bin);
- Generalizations of the **One-Dimensional BP** $\Longrightarrow$ **$\mathcal{NP}$-hard in the strong sense**.

Related multi-dimensional packing problems

- **Two-Dimensional Strip Packing Problem (2SP)**:
  Pack a set of rectangular items into a strip of given width W and infinite height, by minimizing the overall height of the packing. [Applications: cutting rolls of cloth/paper/...]



- **Three-Dimensional Bin Packing Problem (3BP)**:
  Pack a set of rectangular boxes (**items**), of width w_j, height h_j and length l_j $(j = 1, \ldots, n)$, into the minimum number of three-dimensional rectangular containers of width W, height H, and length L. [Applications: container loading, foam cutting]

- **Three-Dimensional Strip Packing Problem (3SP)**:
  Pack a set of rectangular boxes into a strip of given width W, height H and infinite length, by minimizing the overall length of the packing. [Applications: pallet loading]

- **2BP, 2SP, 3BP and 3SP** are **strongly NP-hard**, and very **difficult** to solve in practice.

Bibliography

- **One-dimensional bin packing**
  S. Martello, P. Toth. *Knapsack Problems: Algorithms and Computer Implementations*, Ch. 6 John Wiley & Sons, Chichester-New York (1990) **freeware**

  E.G. Coffman, Jr, J. Csirik, G. Galambos, S. Martello, D. Vigo. Bin packing approximation algorithms: Survey and classification
  Handbook of Combinatorial Optimization, Springer (2013)

  M. Delorme, M. Iori, S, Martello. Bin Packing and Cutting Stock Problems: Mathematical Models and Exact Algorithms (to appear) **freeware**

- **Two-dimensional bin packing**
  A. Lodi, S. Martello, M. Monaci, D. Vigo. Two-dimensional bin packing problems.
  Paradigms of Combinatorial Optimization: Problems and New Approaches, ISTE and John Wiley & Sons (2010)

- **Three-dimensional bin packing**
  S. Martello, D. Pisinger, D. Vigo. The three-dimensional bin packing problem.
  Operations Research 48 (2000)

  S. Martello, D. Pisinger, D. Vigo, E. den Boef, J. Korst. Algorithms for General and Robot-Packable Variants of the Three-Dimensional Bin Packing Problem
  ACM Transactions on Mathematical Software 33 (2007) **freeware**

3. Routing problems with loading constraints

- Recall: **Capacitated Vehicle Routing Problem (CVRP):** find at most K routes of minimum total cost to deliver goods demanded by a set of clients i (each requiring goods of total weight d_i), for a fleet of K vehicles of limited capacity D, based at a central depot.

- **Applications to real world problems** limited by additional constraints:
  - **CVRP**: client demands $=$ total weight of the items to be delivered;
  - **Real-world**: demands $=$ sets of items with a weight and a shape $\Longrightarrow$ combination of CVRP with loading/packing problems.

- **2-Dimensional case:**
  - Transportation of rectangular-shaped items that cannot be stacked one on top of the other (big refrigerators, food trolleys, ...):
  - feasibility of packing on the truck bed;
  - feasibility of the loading and unloading operations.

- **3-Dimensional case:**
  - Transportation of rectangular-shaped boxes that can be stacked one on top of the other;
  - feasibility of box stacking ($\Longleftarrow$ fragility); constraints on the stability of the loading;
  - feasibility of the loading and unloading operations.

CVRP + 2-Dimensional packing

- Complete undirected **graph** $G = (V, E)$: $V = \{0\}$ (depot) $\bigcup$ $\{1, \ldots, n\}$ (clients); edge set $E = \{(i, j)\}$, with c_{ij} = cost of edge (i, j);

- K identical **vehicles**, each having
  - **weight capacity** D;
  - **rectangular loading surface** of width W and height H;

- demand of **client** i $(i = 1, \ldots, n)$:
  - m_i items of total weight d_i;
  - item $I_{i\ell}$ $(\ell = 1, \ldots, m_i)$ has width $w_{i\ell}$ and height $h_{i\ell}$;
  - the items must be orthogonally packed on the loading surface;

- each client must be served by a **single vehicle**;

- let $S(k) \subseteq \{1, \ldots, n\}$ be the set of clients served by vehicle k:
  - **Weight constraint**: total weight $\sum_{i \in S(k)} d_i \leq D$;
  - **Loading constraint**: there must be a feasible (non-overlapping) loading of all the transported items into the $W \times H$ loading area.

CVRP + 2-Dimensional packing (cont'd)

Objective:

- find a partition of the clients into at most K subsets and,
- $\forall$ subset, a route starting and ending at the depot such that
  - all client demands are satisfied;
  - the weight constraint is satisfied;
  - **the loading constraint is satisfied** (feasible packing on the loading area);
  - the total cost of the edges is a minimum.

Two variants:

- **Unrestricted**: no further constraint;
- **Sequential**: the loading of each vehicle must be such that

  when a client is visited, the items of its lot can be downloaded **through a sequence of straight movements (one per item) parallel to the H-edge** of the loading area.

(a) Dashed strip = forbidden area for clients visited after client i
Sequential (b) and non-sequential (c,d) packings

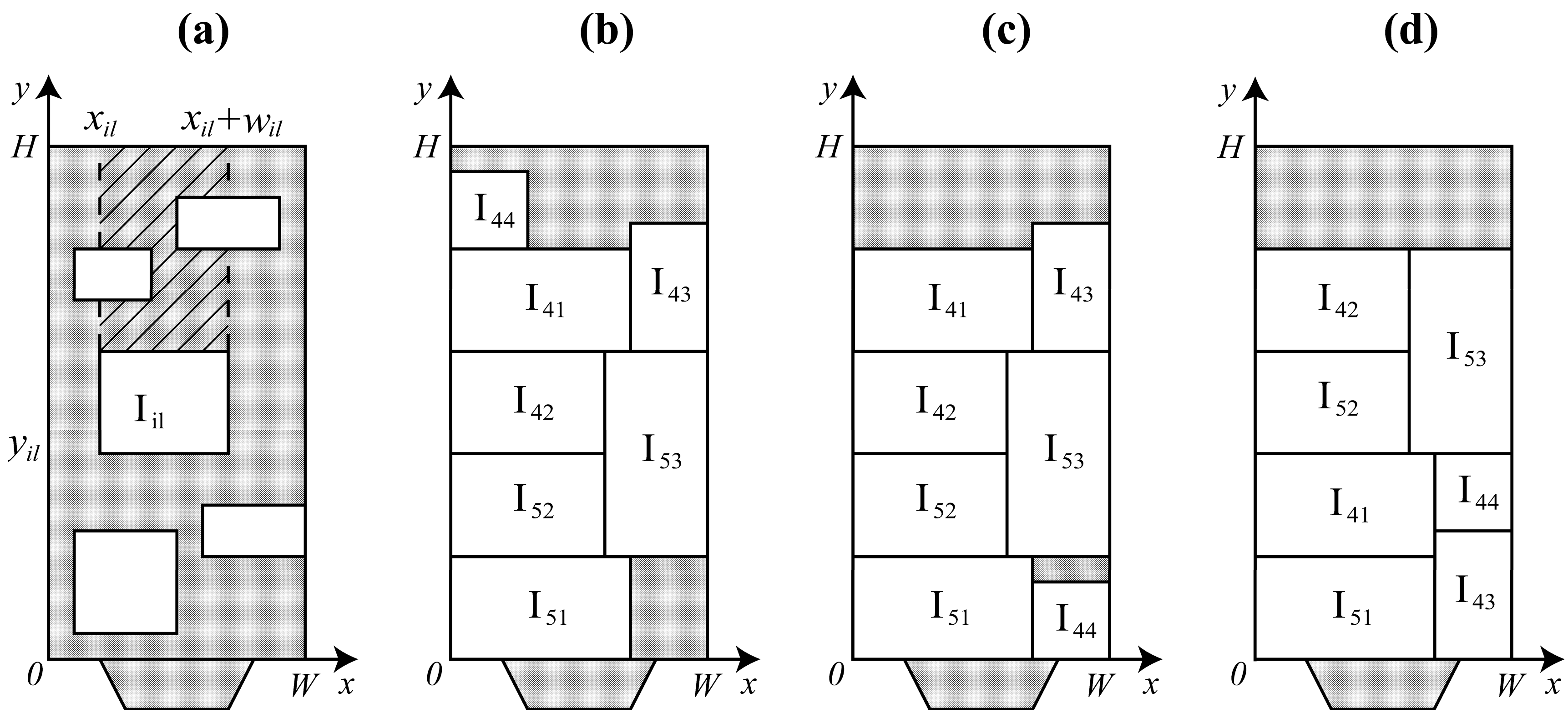
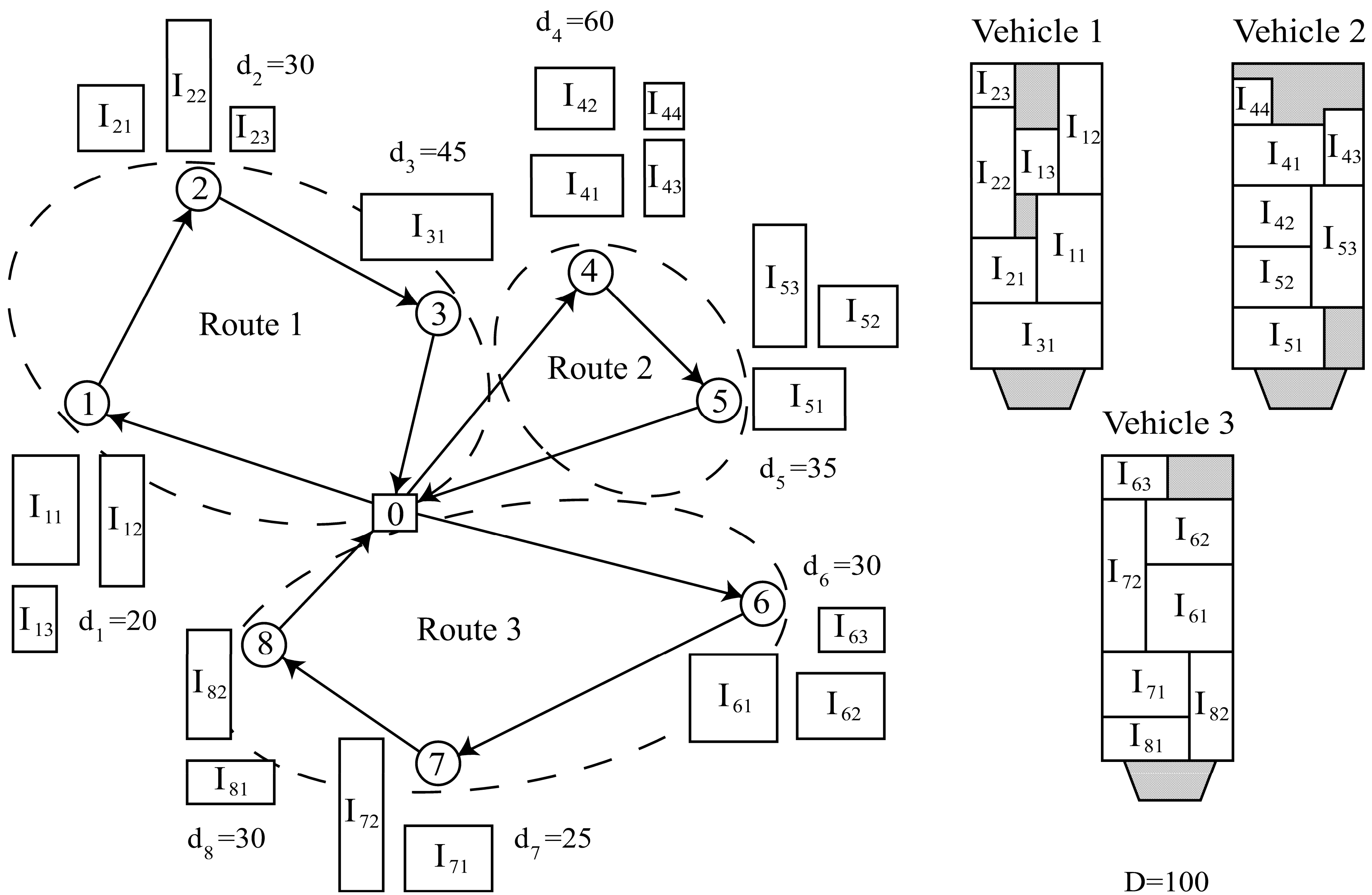


Figura 1: Client 4 visited before Client 5

An instance with 3 vehicles and 8 clients (D = 100)



CVRP + 2-Dimensional packing: Tabu search

- **Neighborhood**
  - derived from Taburoute (Gendreau, Hertz and Laporte, *Management Sci.*, 1994):
  - the algorithm can accept moves producing infeasible tours. Two infeasibilities:
    * *weight-infeasible*: total weight $> D$;
    * *load-infeasible*: height of the loading surface $> H$.
    * Infeasible moves are assigned a proportional *penalty*.
- **Feasibility check of the candidate tour:**
  - *weight-infeasiblity*: immediate;
  - *load-infeasiblity*: NP-hard problem $\Longrightarrow$ **heuristic algorithm** derived from heuristics for 2BP (Lodi, Martello and Vigo, *INFORMS J. Comp.*, 1999), and 2SP (Iori, Martello and Monaci, *Eur. J. Oper. Res.*, 2003).
- **Tabu search objective function (infeasibilities = penalties):**
  - solution s with $c(k)$ = total edge cost in route k:

$$Z(s) = \sum_{k=1}^{K} c(k) + \alpha q(s) + \beta h(s)$$

  - $q(s)$ = total weight excess;
  - $h(s)$ = total height excess in the infeasible loadings;
  - α and β = self-adjusting parameters.
- **Gendreau, Iori, Laporte, Martello, *Networks* (2010).**

CVRP + 2-Dimensional packing: Branch-and-Cut

ILP model:

- based on the classical two-index vehicle-flow formulation;
- only the sequential version has been addressed.

Infeasible routes:

- when an integer solution is found, every route is checked for weight and load feasibility;
- the checking procedure includes
  - greedy heuristics;
  - lower bounds;
  - branch-and-bound;
- $\forall$ infeasible route, a cut prohibiting it is added to the model;
- a pool of solved (proved feasible or infeasible) routes is stored.

Iori, Salazar González and Vigo, *Transportation Science* (2007).

CVRP + 2-Dimensional packing: Computational experiments

- **CVRP instances from the literature** with:
  - $15 \leq \#$ customers ≤ 255;
  - $3 \leq \#$ vehicles ≤ 27;
  - number and dimensions of items randomly generated according to five classes
- **Branch-and-Cut:**
  solves to optimality instances with up to 25 customers
- **Tabu Search:**
  TS (5' CPU limit) beats B&C (24 h CPU) for larger instances.
- Improve metaheuristic algorithms:
  Ant colony optimization: **Fuellerer, Doerner, Hartl, Iori, Comput Oper Res (2009)**
  Guided tabu search: **Zachariadis, Tarantilis, Kiranoudis, Eur J Oper Res (2009)**

CVRP + 3-Dimensional packing

Same constraints as the 2-dimensional case, but

- the items are **three-dimensional** boxes;
- the boxes can be rotated by 90° degrees on the horizontal plane;
- some items can be fragile;
- no non-fragile item be placed over a fragile one;
- when boxes are stacked, the supporting surface must be large enough to guarantee stability;
- the loading of each vehicle must be such that

  when a client is visited, the items of its lot can be downloaded

  without shifting the items requested by other clients.

A sequential three-dimensional vehicle loading

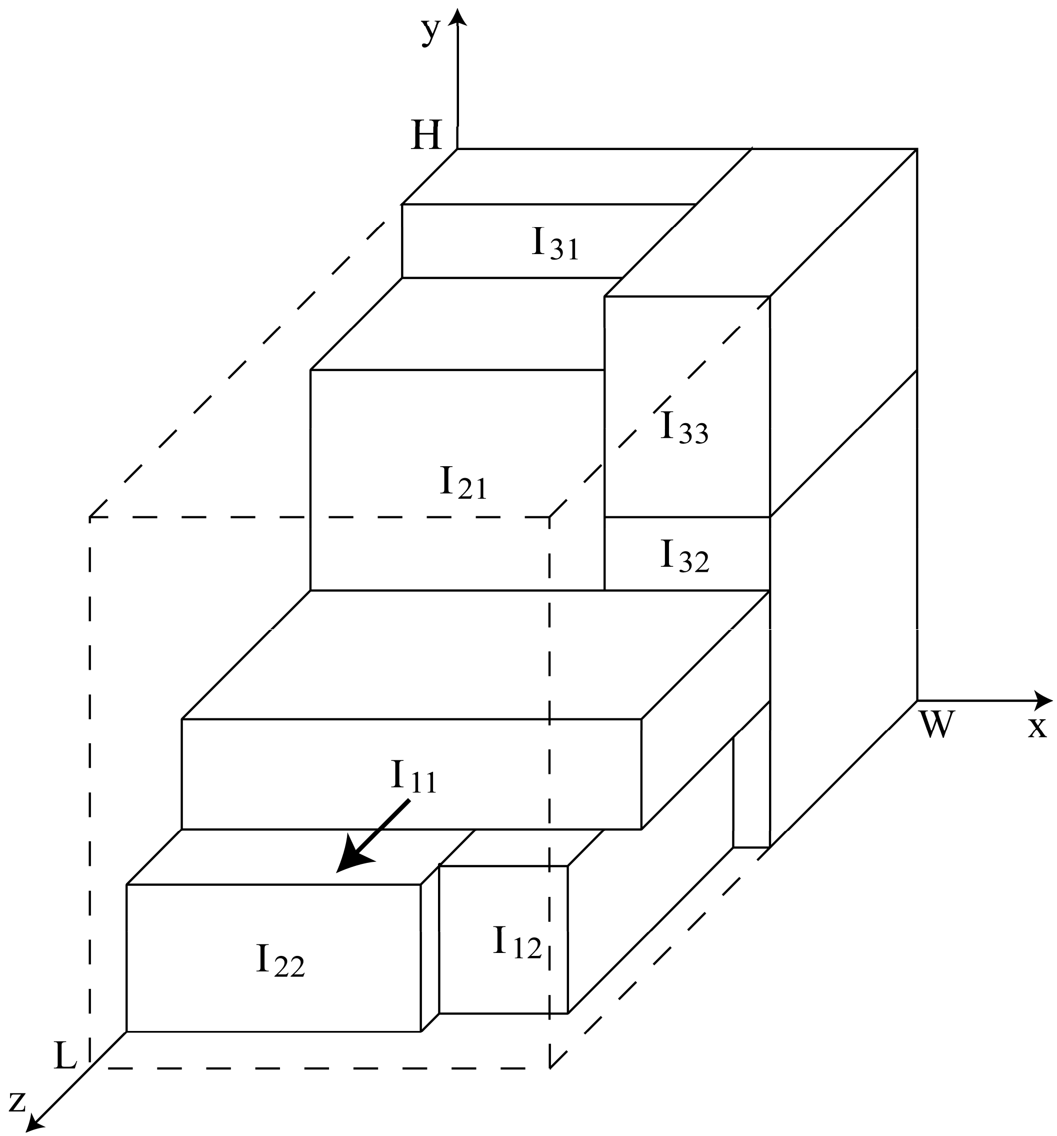
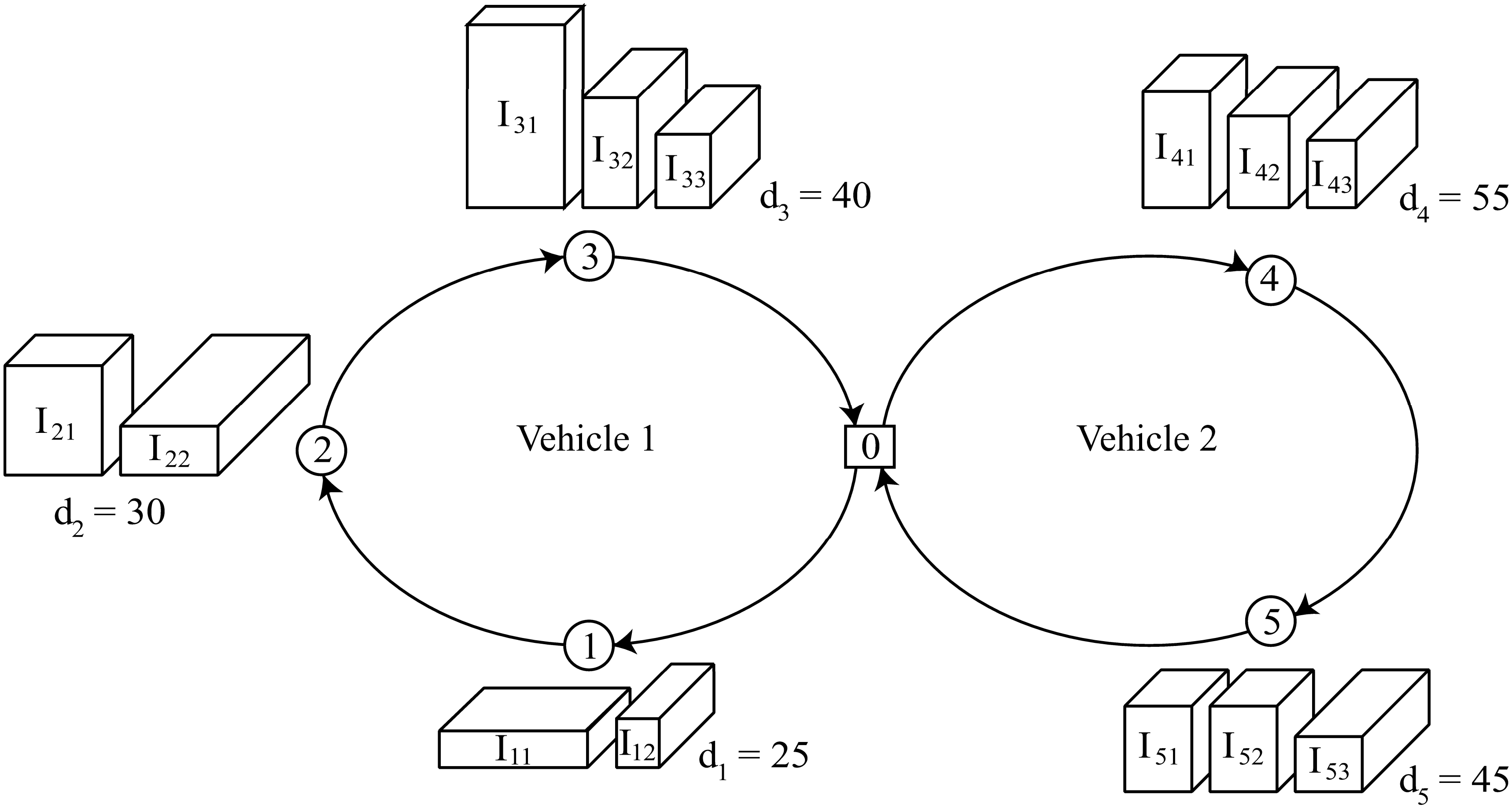


Figura 2: the vehicle is unloaded in the direction of the z axis

An instance with 2 vehicles and 5 clients (D = 100)



CVRP + 3-Dimensional packing: Tabu search

- **Inner Tabu search**
  - iteratively invoked by the main Tabu search;
  - subproblem solved:

    given an ordered set of clients to be visited in this order,

    can all the requested items be feasibly loaded into a single vehicle?
  - the subproblem is NP-hard;
  - solved as a three-dimensional strip packing problem;
  - **neighborhood:** modify the loading sequence and execute two greedy heuristics;
  - if no feasible solution is found, the algorithm returns a *load-infeasible* solution: loading length $\lambda > L$ (vehicle length);
  - if the total weight or the total volume exceeds the vehicle capacity, *dummy loading*: length $\lambda = 2L$;
  - the main Tabu search uses λ to evaluate the moves.
- **Outer Tabu search:** derived from the algorithm for CVRP + 2-Dimensional packing.
- **Gendreau, Iori, Laporte, Martello, Transportation Science (2006)**.
- Guided tabu search: **Fuellerer, Doerner, Hartl, Iori, Comput Oper Res (2009)**
- Ant colony optimization: **Fuellerer, Doerner, Hartl, Iori, Eur J Oper Res (2010)**

Other routing problems with loading constraints

- **Basis: Traveling Salesman Problem with Pickup and Delivery (TSPPD):**

  single vehicle must visit a set of customers, each associated with an origin location where some items must be picked up, and a destination location where such items must be delivered;

  find a shortest Hamiltonian cycle through all locations while ensuring that the pickup of any given request is performed before the corresponding delivery.

  - **TSPPD and LIFO loading:**
    pickups and deliveries must be performed in LIFO order (vehicles with a single access point);
  - **TSPPD and FIFO loading:**
    pickups and deliveries must be performed in FIFO order (AGVs that load items on one end and unload them at the other end);
- **CVRP + 2-dimensional loading + pickup and delivery constraints**;
- **3-dimensional container loading problems with multi-drop constraints** (special sequences);
- **CVRP with time windows and three-dimensional container loading**;
- **CVRP with pickup and delivery, delivery due dates and 3-dimensional loading**:
  auto-carrier transportation problem;
- **TSP with pickup and delivery and handling costs** (when the loading is not sequential);
- **...**

CVRP + 3-Dimensional packing: Computational experiments

Real-world instances:

- Italian company (furniture for bedrooms);
- fleet of private-owned vehicles paid per mileage;
- demands: three-dimensional rectangular items (to be assembled);
- identical vehicles (standard ISO containers);
- time windows neglected
- typical solutions: single day tours + multiple days tours.
- volumes between 1% and 4% of the vehicle volume;
- heights between 10% and 50% of the vehicle height.

Figure 3 Distribution of Clients in Italy (Instance F01)

Italy are handled through a fleet of privately owned

to those adopted in Tables 1 and 2, was mainly
to the increased difficulty associated with the th
dimensional loading. Indeed, the excess of length
computed as in §5.1, but we allowed $TS_{3L\text{-}SV}$ to
form up to 10 iterations (instead of 3). The first
entries give the instance name, the number of clie
(n), the number of items (M), the number of ve
cles (v), and the solution value (in kilometers) fou
by the initial heuristics (z_0). The three next pairs
entries give, for each time limit, the solution value
and the elapsed CPU time required to obtain
value (sec_z).

The initial heuristics find a feasible solution for
five instances, with an average solution value equa
7,091 km. The three runs decrease this value by 33.
35.3%, and 37.8%, respectively. For three instan
the values obtained after 10 hours remain unchan
after 24 hours. When used in an operational pl
ning situation, the preferred time limits are 10 ho
(overnight) or 1 hour (when relatively fast decisi
have to be taken).

5.3. Robustness and Parameters Setting

The parameters setting was performed on the 27
stances considered in §5.1. We start with the setting
the parameters needed to compute the score of a m
(see (1)). The algorithm proved to be quite robust w
respect to the starting values assigned to paramete

					1 hour CPU time		10 hours CPU time		24 hours CPU time	
Instance	n	M	K	Greedy	z	sec_z	z	sec_z	z	sec_z
F01	44	141	4	7711	3723	2839.4	3694	32133.9	3694	32133.9
F02	49	152	4	7167	4182	1993.8	4182	1993.8	3941	86046.8
F03	55	171	4	6111	3674	3478.5	3650	31776.5	3650	31776.5
F04	57	159	4	7059	4686	2520.5	4543	5049.7	4509	5995.1
F05	64	181	4	7408	7235	2366.3	6886	33917.9	6241	75441.1
Average				7091	4700	2639.7	4591	20974.3	4407	46278.7

M = total number of items to deliver.

Additional Bibliography

- **Surveys and special issues:**

  M. Iori, S. Martello. Routing problems with loading constraints. *TOP* (2010)

  M. Iori, S. Martello. An annotated bibliography of combined routing and loading problems. *Yugoslav Journal of Operations Research* (2013)

  T. Bektas, G. Laporte, and D. Vigo, eds. Special issue on Integrated Vehicle Routing Problems. *Computers & Operations Research* (2015)

  H. Pollaris, K. Braekers, A. Caris, G.K. Janssens, S. Limbourg. Vehicle routing problems with loading constraints: state-of-the-art and future directions. *OR Spektrum* (2015)

- **Recent results:**

  A. Bortfeldt. A hybrid algorithm for the capacitated vehicle routing problem with three-dimensional loading constraints. *Computers & Operations Research* (2012)

  A. Bortfeldt, J. Homberger. Packing first, routing second - a heuristic for the vehicle routing and loading problem. *Computers & Operations Research* (2013)

  A. Bortfeldt, T. Hahn, D. Männel, L. Mönch. Hybrid algorithms for the vehicle routing problem with clustered backhauls and 3D loading constraints. *European Journal of Operational Research* (2015)

Conclusions and open perspectives

Routing problems with loading constraints constitute a rich and lively research areas.

Many issues need to be explored or better studied. For example:

- Evaluation of classical vehicle routing situations (e.g., allowing split deliveries).
- Heterogeneous vehicle fleets.
- Addition of special loading requirements (e.g., issues related to the center of gravity of the load).
- Use of column generation techniques for effectively determining exact solutions.
- Study of different objective functions reflecting special practical needs, such as routes with similar lengths or loads.
- Integration of these models and algorithms with location issues.
- ...

Thank You for your attention