Traffic Optimization Workshop 2015, October 8, 2015, Heidelberg

Arc Routing Problems: History, Applications and Perspectives

Ángel Corberán

Departament d'Estadística i Investigació Operativa

Universitat de València

Contents

- Introduction
- Applications
- Eulerian graphs and the Chinese postman problem
- The RPP, GRP and CARP
- Perspectives
  - Arc routing problems with profits
  - Arc routing problems with aesthetic constraints

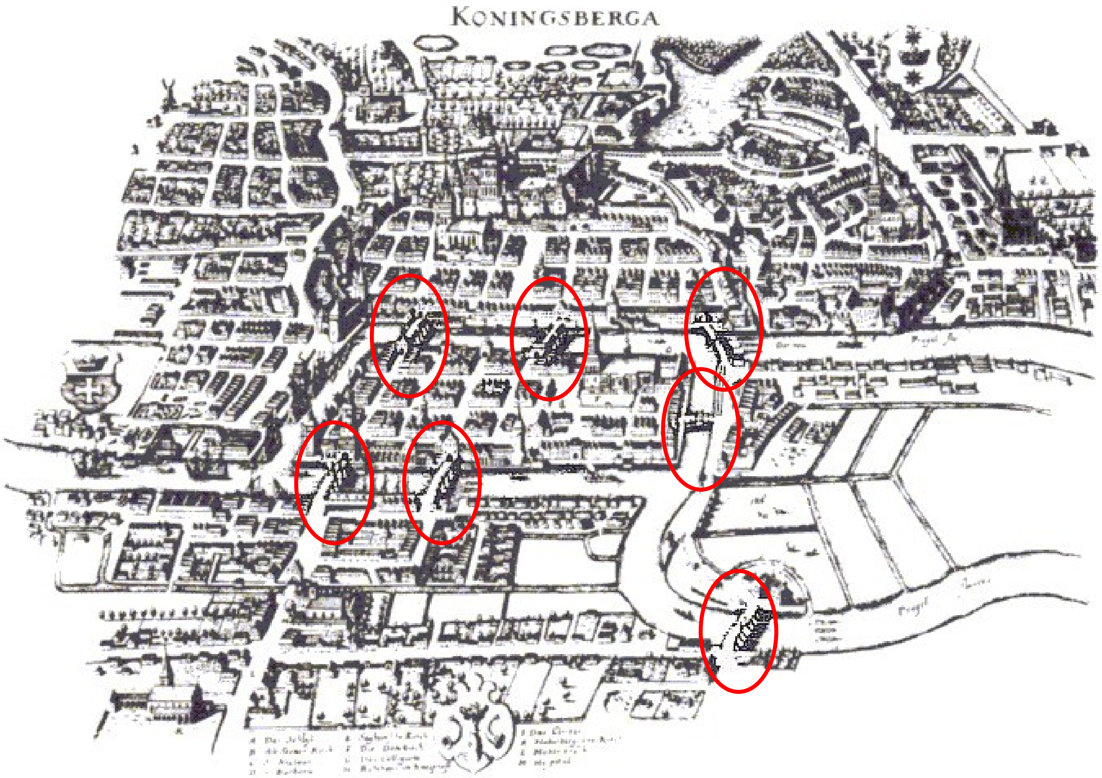
Königsberg's Bridge Problem

Königsberg is a city which was the capital of East Prussia but now is known as Kaliningrad (Russia).

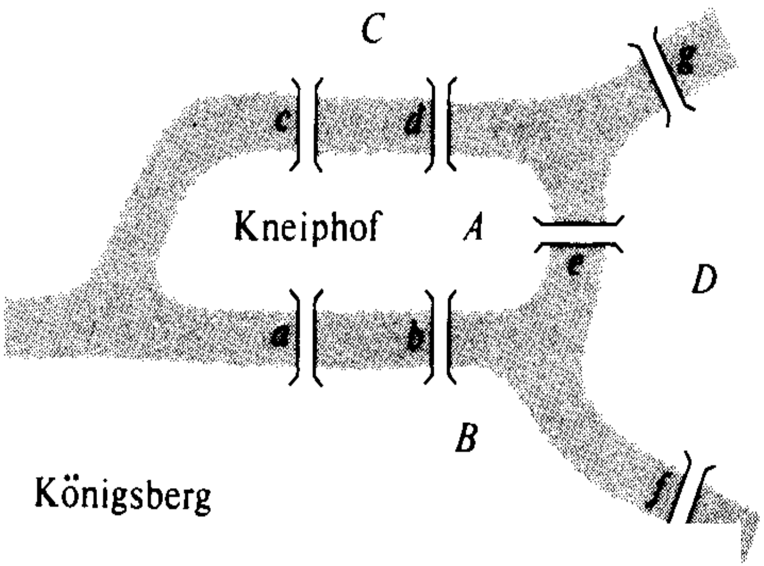
The city is built around the river Pregel where it breaks into 2 parts. An island named Kneiphof is in the middle of where the river splits. At the XVIII century, 7 bridges joined the 4 parts of the city.

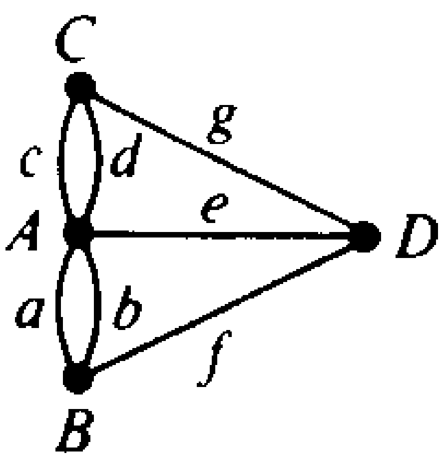
People tried to find a way to walk all seven bridges without crossing a bridge twice, but no one could find a way to do it.

Königsberg's Bridge Problem



Königsberg’s Bridge Problem

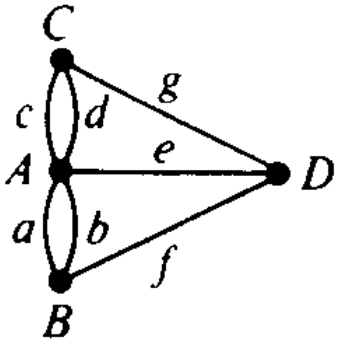


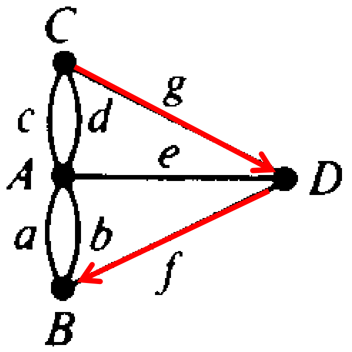


Königsberg's Bridge Problem

Euler pointed out that finding a route traversing every bridge exactly once is possible if and only if : "*when we traverse a bridge and arrive to a zone of the city, we should leave it by crossing another bridge*".

Each time a closed walk passes through a node, it will traverse two different edges incident with that node





If all the edges have to be traversed exactly once, then the number of edges incident with a node (degree) must be even

The Chinese Postman Problem

At the sixties, Meigu Guan, a mathematician at the Shandong Normal College, was encouraged (like many other scientists in China) to solve real-life problems during the Great Leap Forward movement (1958-1960), which attempted to transform the country from an agrarian to a modern economy.

"When the author was plotting a diagram for a postman's route, he discovered the following problem: A postman has to cover his assigned segment before returning to the post office. The problem is to find the shortest walking distance for the postman".

The Chinese Postman Problem

While the Königsberg's Bridge Problem raised only the problem about the existence of a tour and obtaining it ....

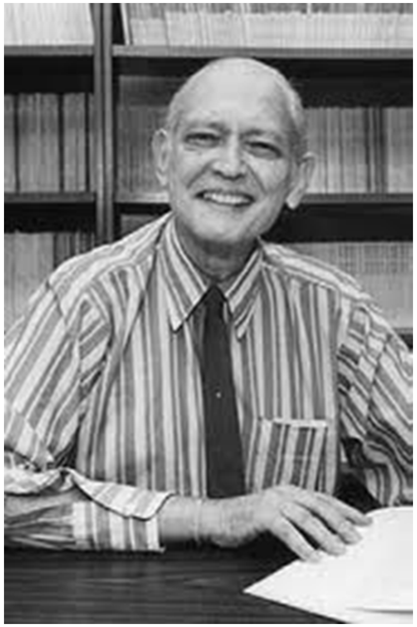
now the problem is dealing with situations in which probably there is not a Eulerian tour, but that need for a real solution.

If a graph does not have an Eulerian tour, a natural question is that of obtaining a minimum length tour traversing every edge in the graph at least once (Chinese Postman Problem, CPP)

The Chinese Postman Problem

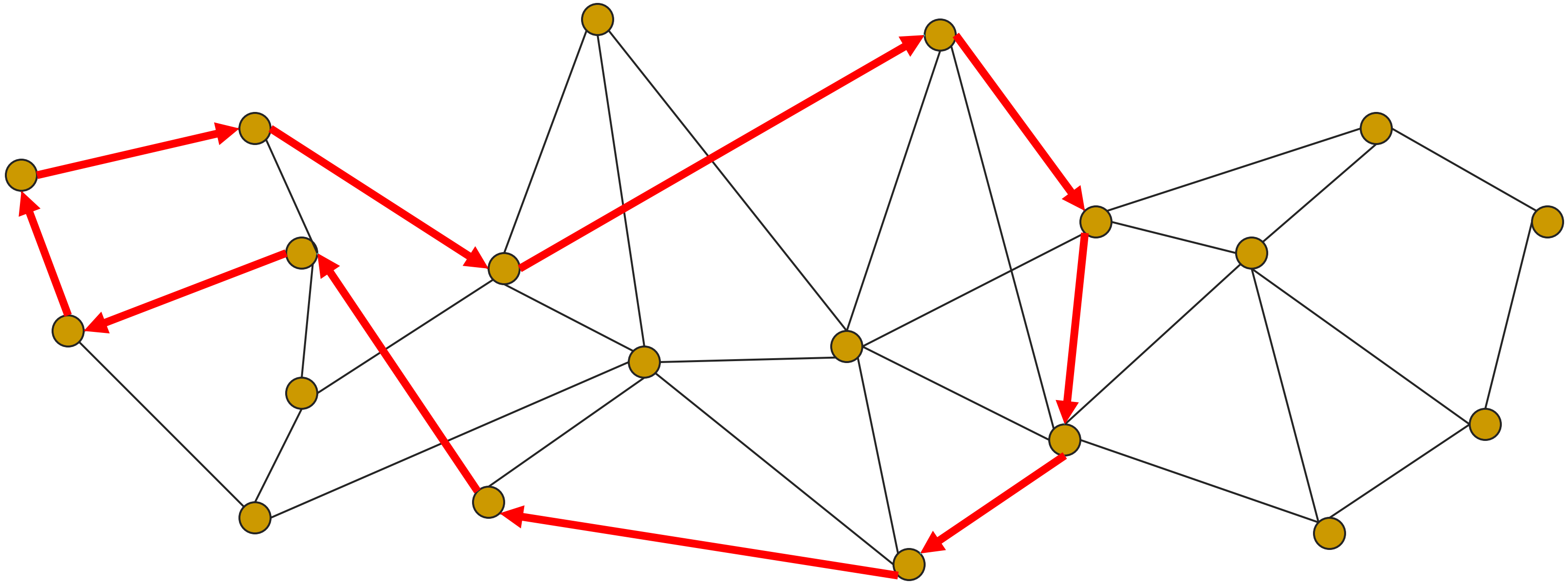
- Guan's article referred to optimizing a postman's route, was written by a Chinese author, and appeared in a Chinese maths journal.
- It seems that Alan Goldman mentioned it to Jack Edmonds when Edmonds was a member of Goldman's Operations Research group at the U.S. National Bureau of Standards.
- It is not know if Goldman suggested the name "Chinese Postman Problem" to Edmonds or whether it was Edmonds who coined that name.
- It seems that the name appeared for the first time in the title of an abstract by Edmonds for the 27th ORSA meeting (May 1965): "The Chinese Postman's Problem".

Pictures



Arc Routing Problems

Problems related to the traversal of some (or all) of the arcs of a transportation network



References

Surveys on Arc Routing Problems

- Assad and Golden (1995)
- Eiselt, Gendreau and Laporte (1995a,b)

Annotated Bibliography:

- C. and Prins (2010)

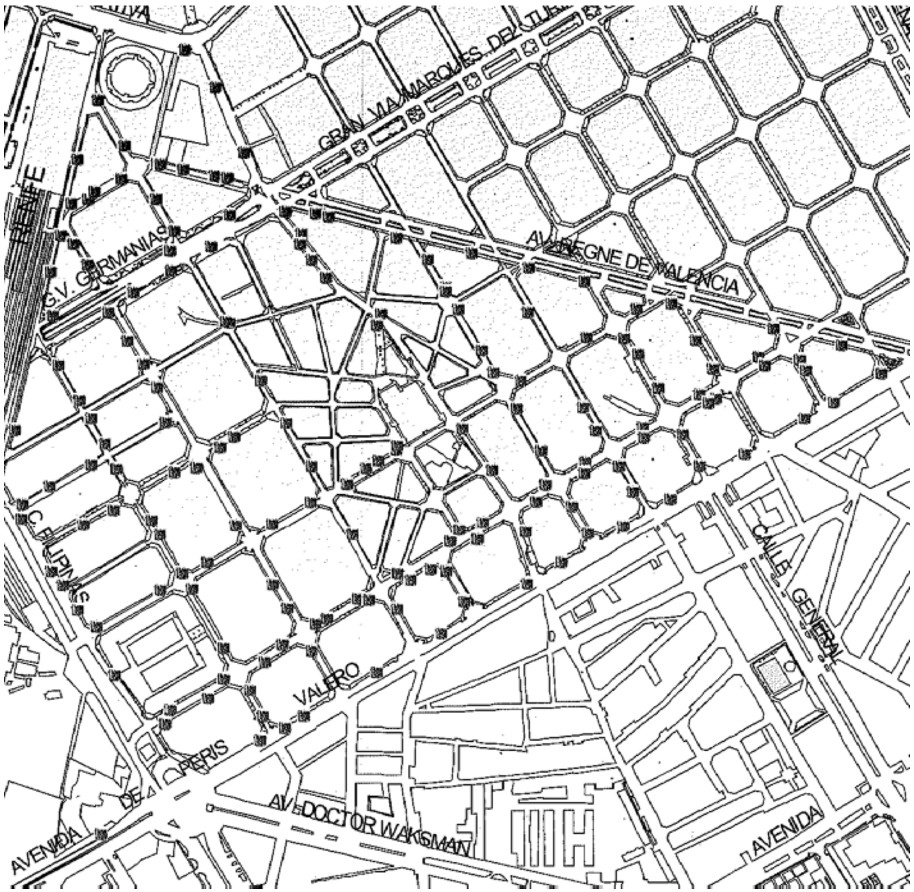
Books on ARPs:

- Dror, ed. (2000)
- C. and Laporte, eds. (2014)

Contents

- Introduction
- Applications
- Eulerian graphs and the Chinese postman problem
- The RPP, GRP and CARP
- Perspectives
  - Arc routing problems with profits
  - Arc routing problems with aesthetic constraints

Garbage collection



AJUNTAMENT DE VALENCIA

Barrio 2.1 - Russafa

Street cleaning and garbage collection

(2002) Valencia:

- Street cleaning (daily): 1028 workers, 11 trucks
- Street watering: 39 trucks
- Garbage collection: 10584 bins + 792 (glass) + 711 (cardboard) + 25 (plastic) + 30 (other) 101 trucks

- Budget 2007 : 130.107.449 Euros (18,23 %)

Street cleaning and garbage collection

First work: CLARK & GILLEAN (1975)

(1972-1974) Cleveland:

- Significant reductions in the garbage collection cost:

  from 1640 workers to 850 workers.
- Budget: from 14.8 million dollars in 1970

  to 8.8 million dollars in 1972

Snow and Ice control

Snow and Ice control

Highway 720 during a snow storm in Montrèal. The Montreal Gazette, 07/03/2011

(Synchronized Arc Routing for Snow Plowing Operations, Salazar-Aguilar, Langevin, Laporte, 2011)

Snow and Ice control

(1987-88) Indiana:

- Budget of the Highway Department for winter maintenance: 15 million dollars.
- 114000 miles (roads and highways)

  1500 workers

  1000 vehicles

HASLAM & WRIGHT (1991)

Snow and Ice control

“The importance of winter road maintenance is due to the magnitude of the expenditures associated to these operations, and to the indirect costs resulting from the loss of productivity and decreased mobility.
In the United States alone these operations consume over $2 billion yearly in direct costs.
In Japan and Europe snow removal expenditures are two to three times those of the United States”.

(Salazar-Aguilar, Langevin, Laporte, 2011)

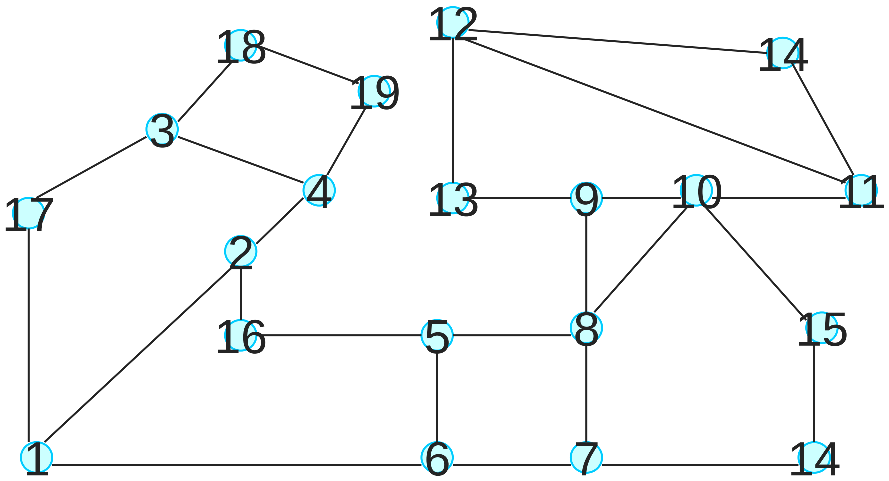
Snow and Ice control

“In Montrèal the average cost of a 20 cm snow storm in 2010 was $17 million Canadian dollars.
Each year, the city has to clear 6,550 km of sidewalks and 4,100 km of streets.
On average, there are 65 weather events calling for response every winter.
Snow clearings performed in four stages: salting, plowing, removal, and disposal.
Plowing operations begin as soon as there is an accumulation of 2.5 cm of snow on the ground and continue as long as the storm lasts, ending about eight hours after the snow stops falling”.

Pierce Point Minimization in Flame Cutting

Pierce Point Minimization and Optimal Torch Path Determination in Flame Cutting

Manber & Israni (1984) considered the problem of minimizing the number of piercing points required in the laser cutting process.

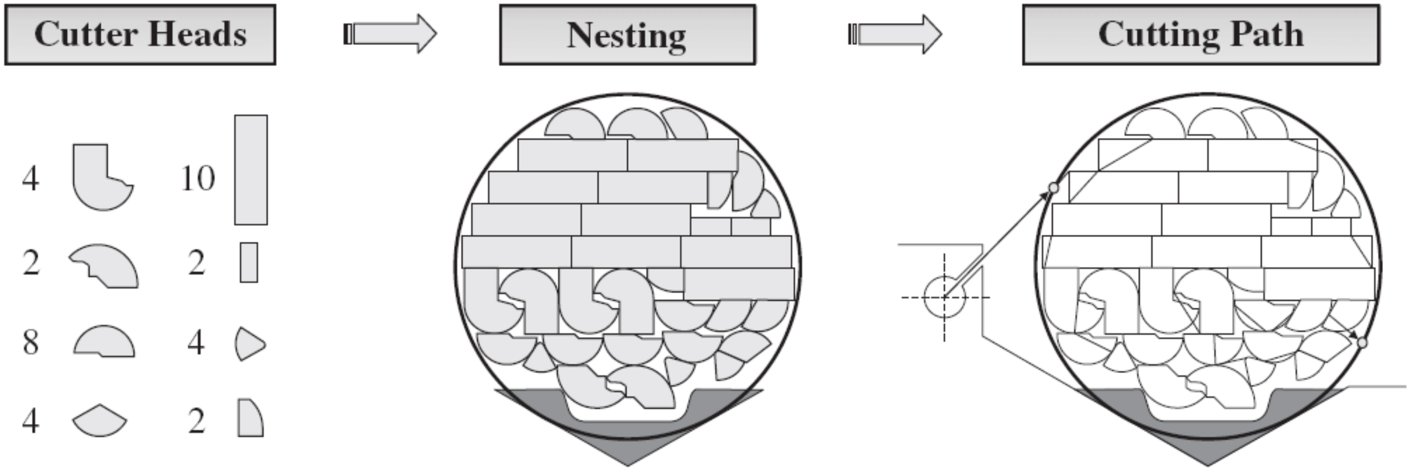


Cutting path determination problem

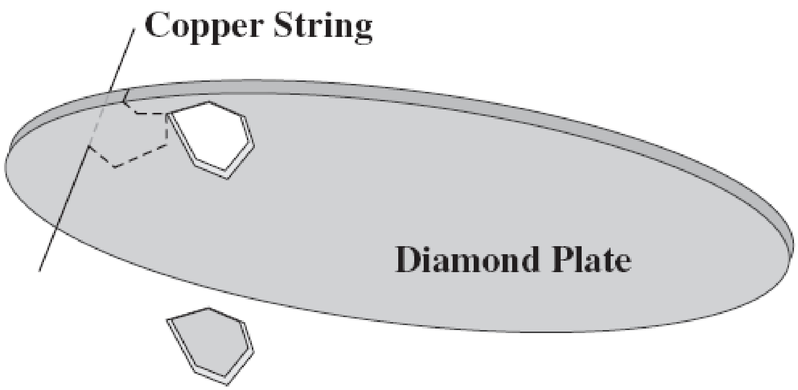
Moreira et al., 2007

A large company manufactures high precision tools for wood, plastic and composite materials. The production process includes the cutting of cutting heads which have to be cut off from expensive circular plates made of tungsten with a thin diamond layer.

The problem consists of finding an optimal cutting path for the cutting out of pieces.



Cutting path determination problem



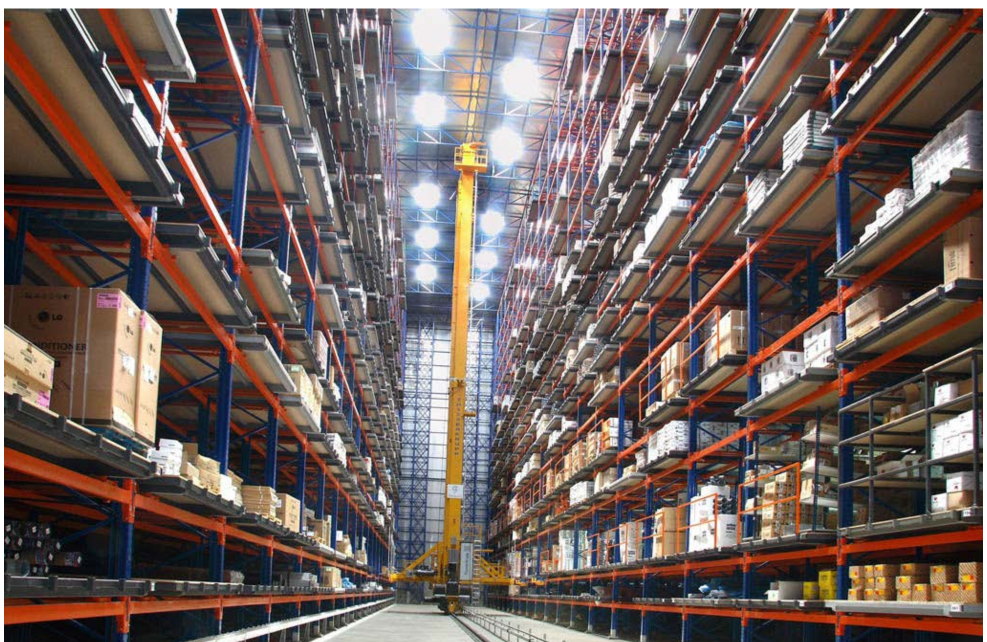
The cutting process is performed by an “electrified copper string”. Basically, the electrified string traverses the circular plate, cutting out the small pieces which fall off in a special container.

The plate is approx 10 cm wide in diameter, with a border waste of 0.5 mm. The copper string speed is constant 1.5 mm/min. A plate completely filled takes about 20 hours to be completely cut.

The Stacker Crane Problem

Frederickson, Hecht & Kim (1978)

A crane must start from an initial position, perform a set of movements, and return to the initial position. The objective is to schedule the movements of the crane so as to minimize the total cost.



Meter Reading

First work: STERN & DROR (1979)

Beersheva (Israel)

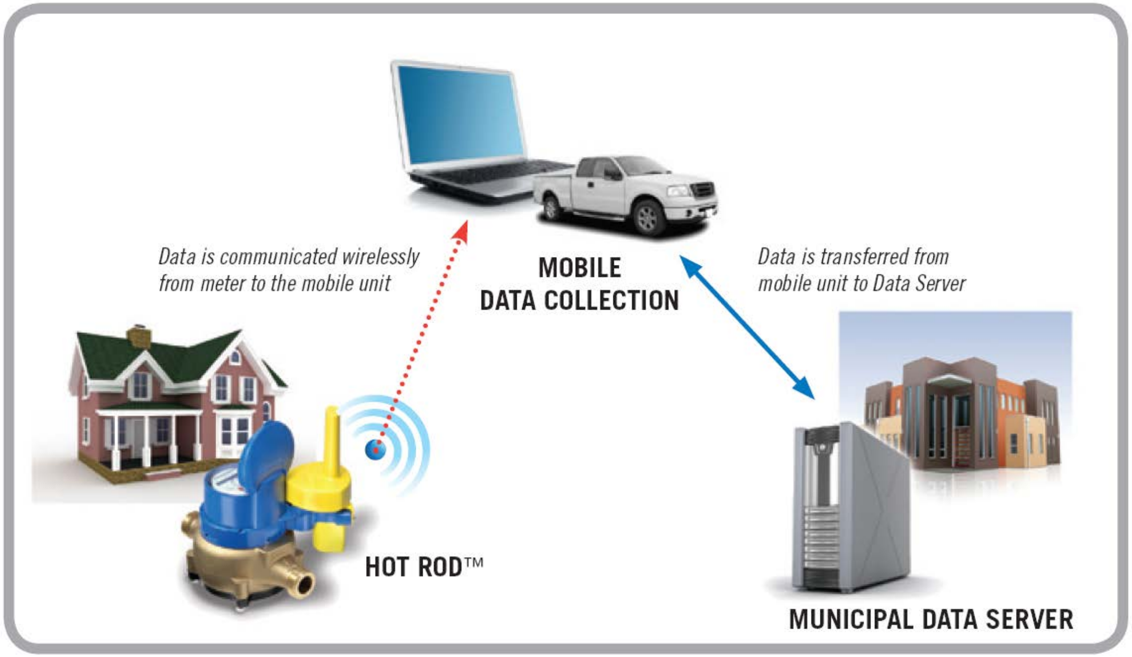
8 Zones (1 zone consists of 42 nodes and 62 edges)

- Important reduction: from 24 to 15 tours in 1 zone
- Estimated saving: 40% in 1 zone.

Meter Reading

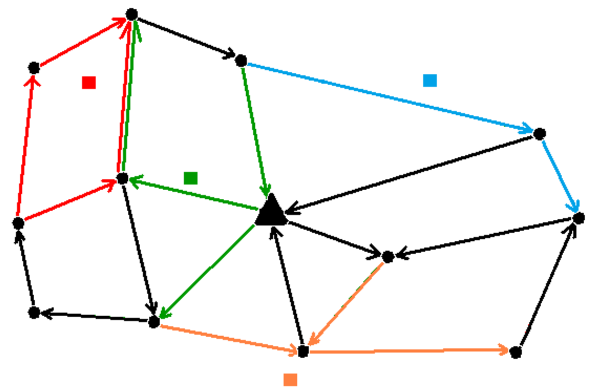
Each meter has a RFID (Radio Frequency IDentification) tag.

A RFID reader can read the data of each meter located closer than a given distance r .



Meter Reading

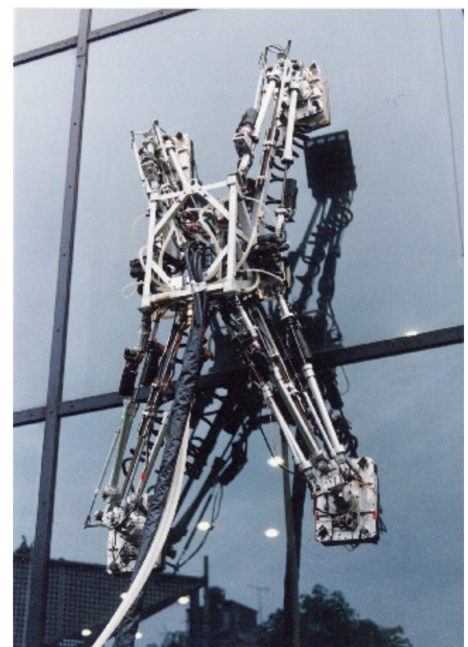
Nowadays, the service (meter reading) do not consist of traversing a given street, but a close-enough street to the customer



Each customer has associated a set of close-enough street segments. The goal is to traverse at least one of these streets for each customer, at minimum cost

Shuttleworth, Golden, Smith, and Wasil (2007)
Ha, Bostel, Langevin y Rousseau (2014)
Ávila, C., Plana & Sanchis (2015)

Inspection of 3D structures by teleoperated robots



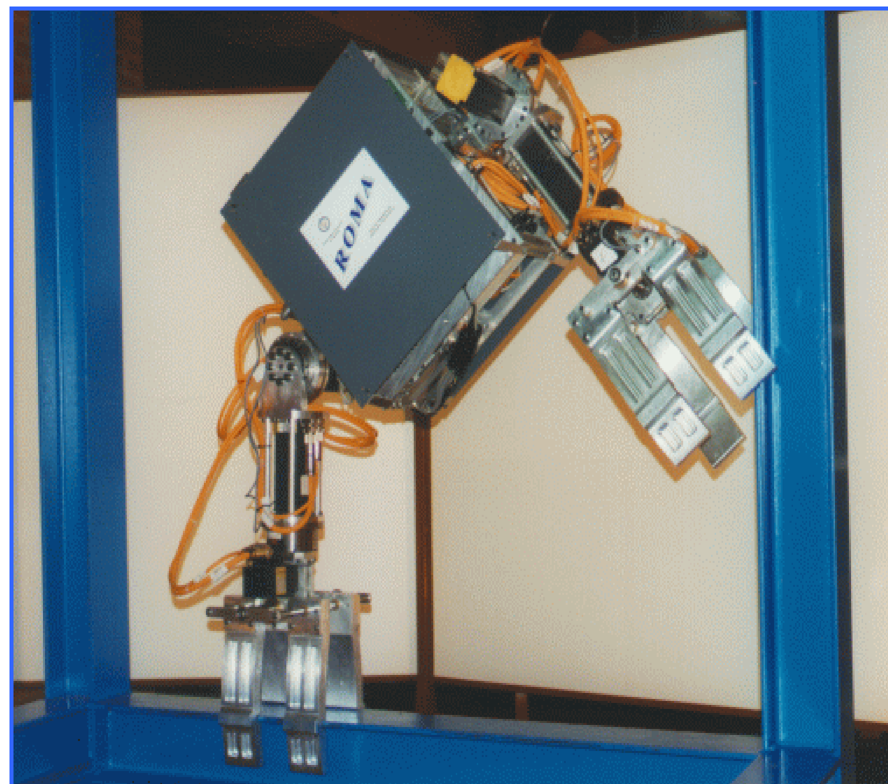
A climbing robot has to inspect a set of elements of a 3-D structure optimizing its energetical consumption

ROMA Robot

(RObot Multifuncional Autoportante)

Area de Ingeniería de Sistemas y Automática de la Univ. Carlos III

- Autonomy: 3 hours
- Weight: 75 Kg
- Intelligent control system (CPU, Ethernet via radio, TV camera, laser telemeter) on board



Modelling the problem

We want to find the optimal route for the robot:

- Minimizing its consumption
- Maximizing its autonomy

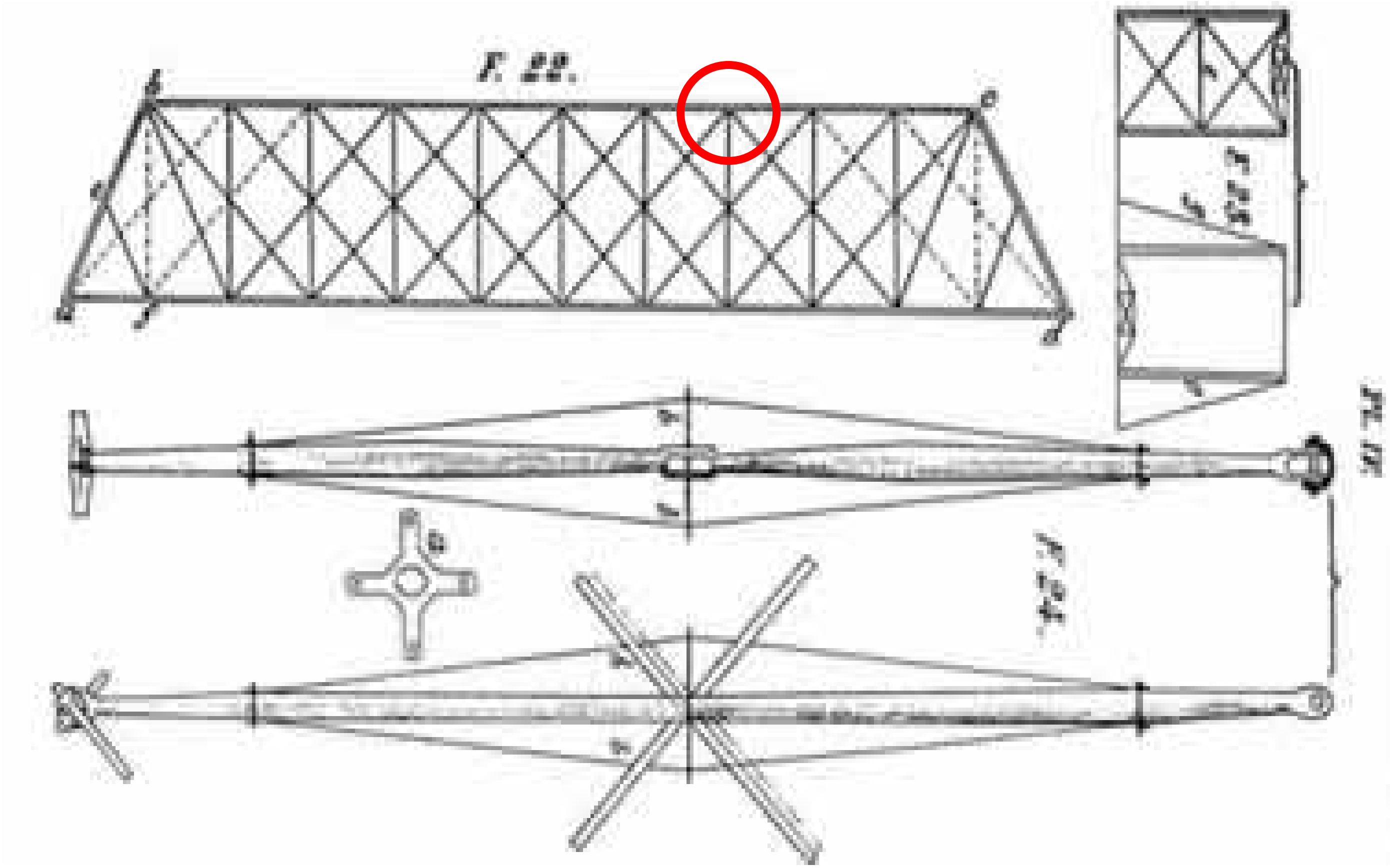
What is needed?

Information on the robot energy consumption:

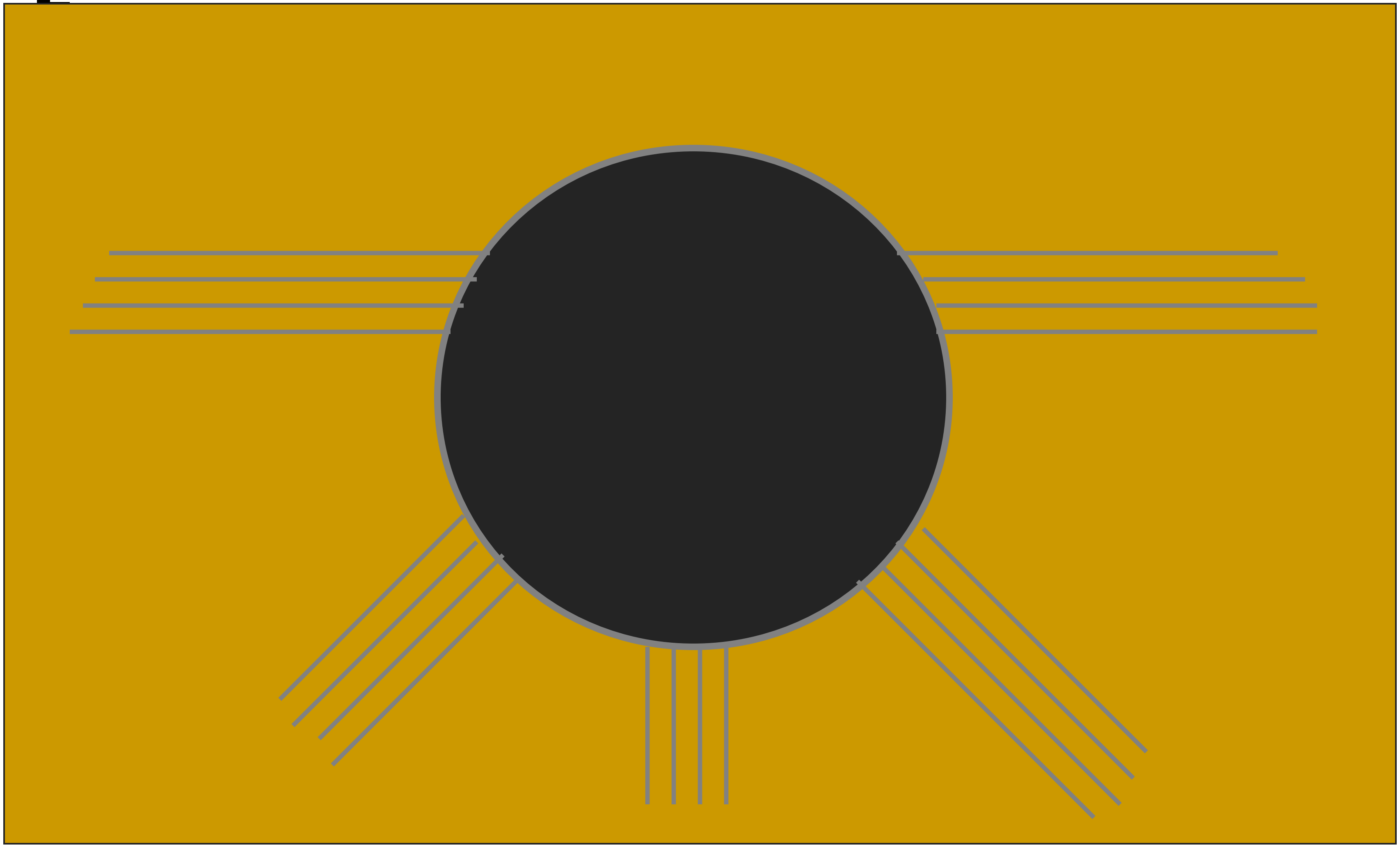
- Cost of traversing an element (asimmetry)
- Cost of traversing a junction (asimmetry)

Modelling the junctions

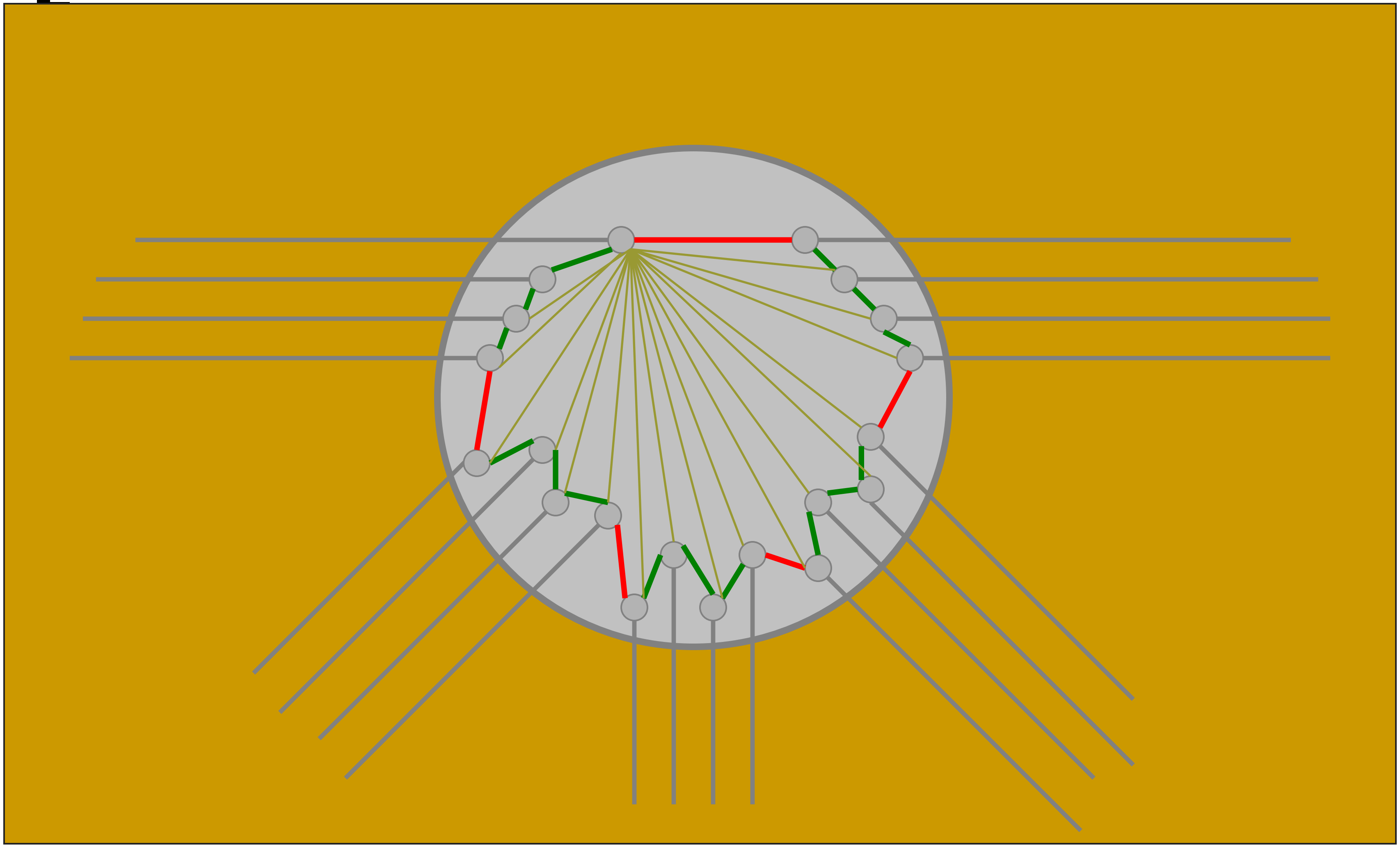
Modelling junctions



Modelling junctions



Modelling junctions

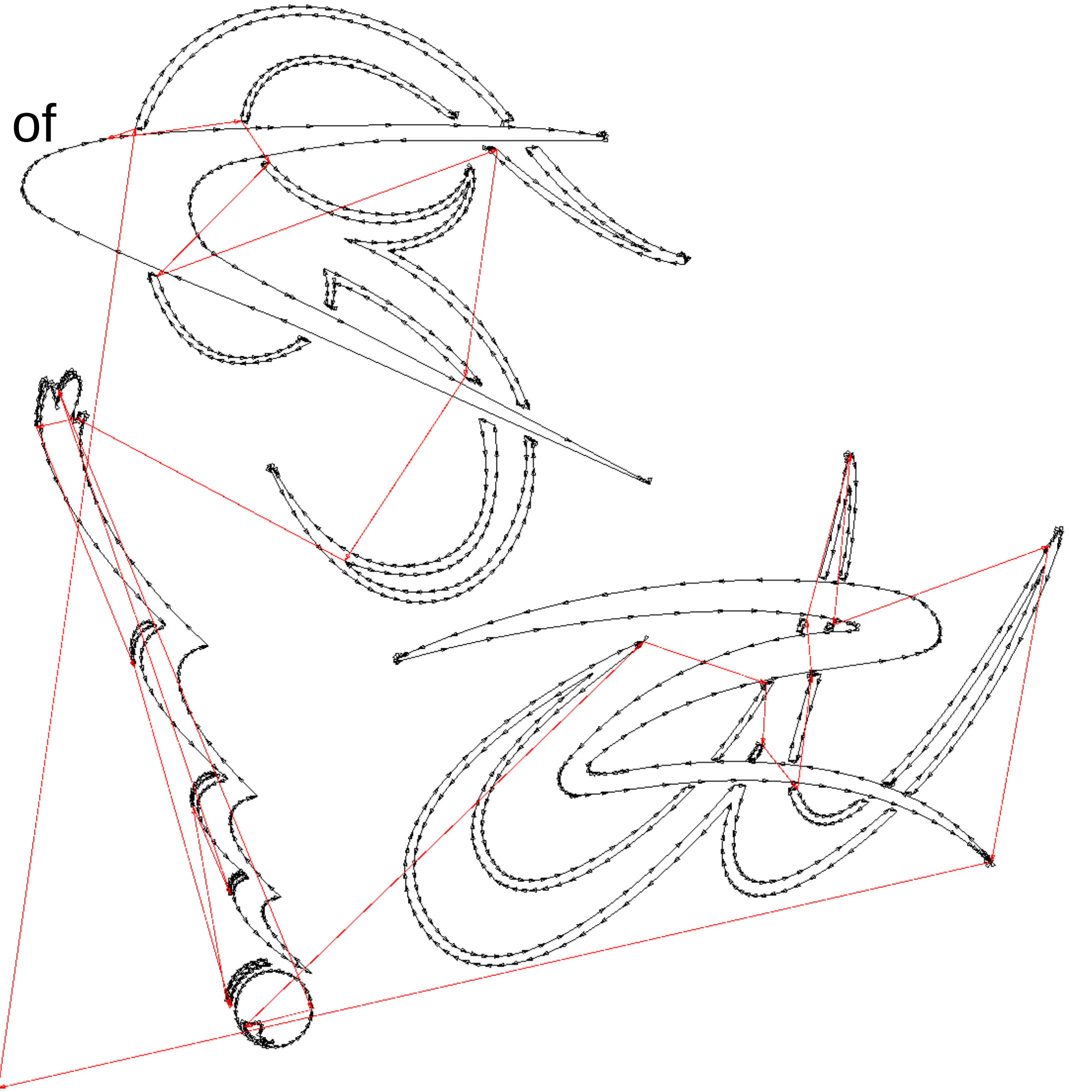


Cutting plotter

Sticker contour shapes

Cutting plotter

The design consists of a number of 'vectors' that need to be cut out with the knife down.



black arrows: edges to be traversed (cut out)

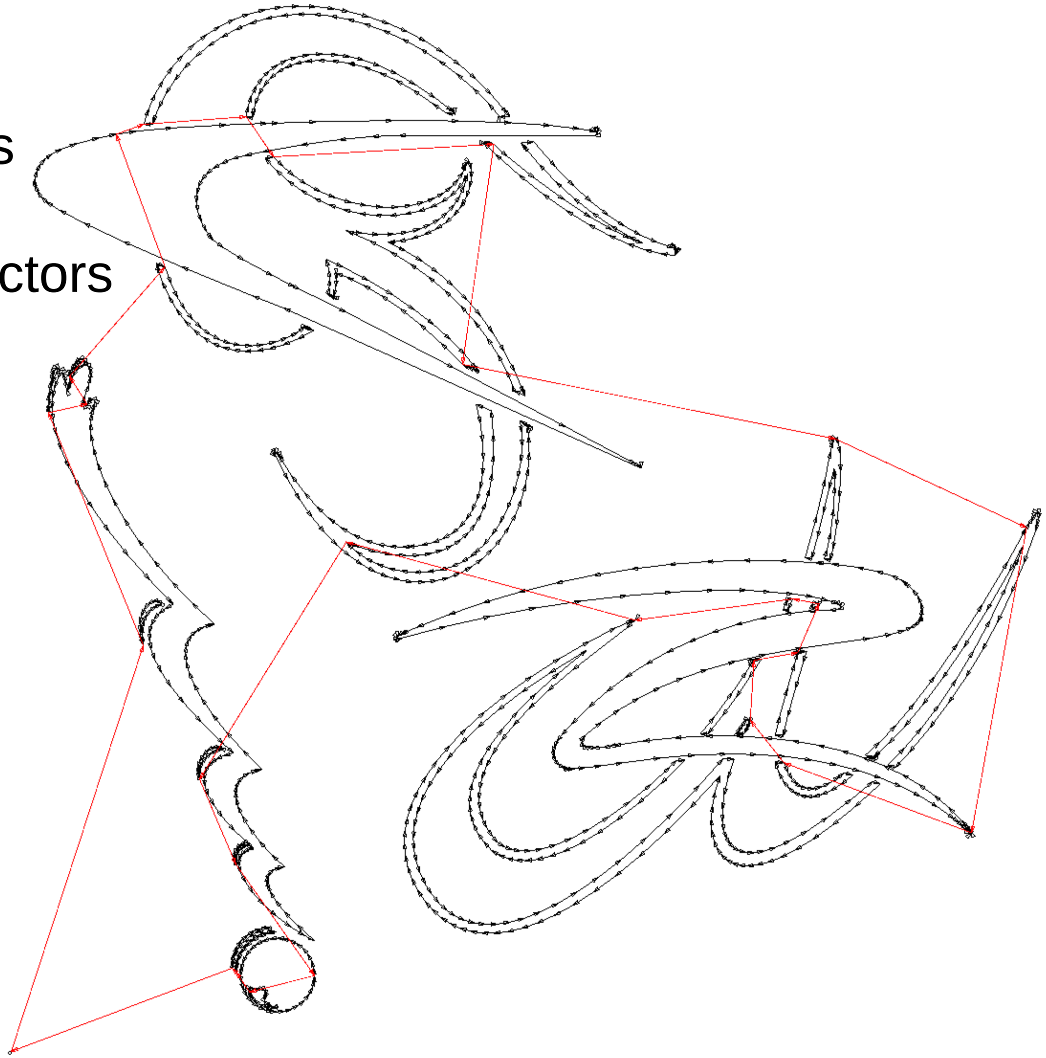
red arrows: non-required edges (knife-up moves)

Up time: 82564.29

Down time: 204545.60

Cutting plotter

For some material types, there is also a preferred or even obliged movement direction for these vectors (preference to pull the material instead of pushing it).
This is the 'windy' aspect.



Up time: 48520.55

Down time: 204545.60

Arc Routing Applications

Node aggregation

- Delivery of newspapers to subscribers,
- postal mail delivery,
- pickup of household waste, ...

In urban areas, there are often thousands of points to be serviced along a subset of street segments.

These problems can be formulated as arc routing problems with a drastic reduction of its size.

Contents

Eulerian graphs

A Eulerian tour is a closed walk (tour) that traverses each edge of the graph exactly once.
A Eulerian graph is one for which there is a Eulerian tour.

An undirected connected graph G=(V,E) is Eulerian if and only if all their vertices have even degree (even graph) (Euler 1736, Hierholzer 1873)

An undirected connected graph G=(V,E) is Eulerian if and only if it is the union of disjoint cycles.
(Veblen 1912)

Eulerian graphs

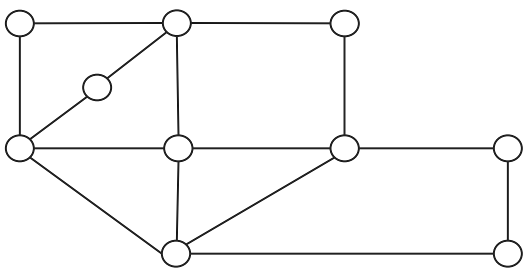
Hierholzer's algorithm for finding a Eulerian tour, $O(|E|)$

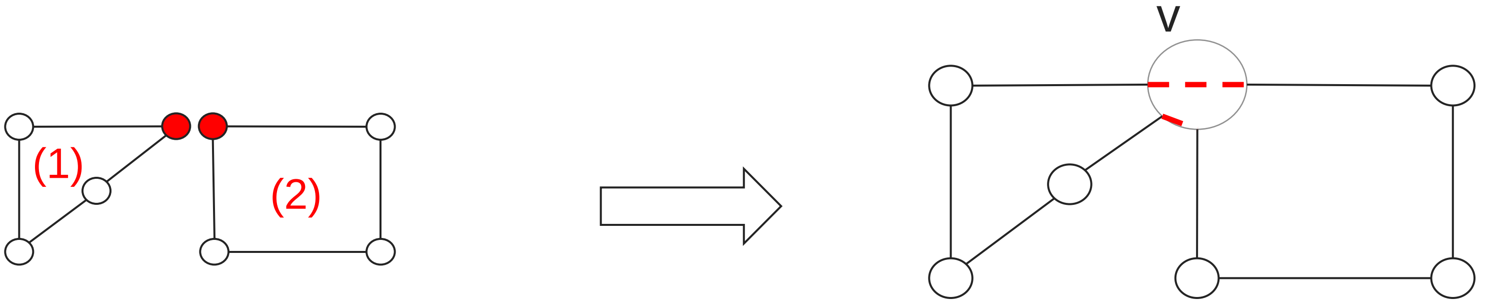
Step 1. Starting from an arbitrary node v, gradually traverse a cycle by following untraversed edges until returning to v.

Step 2. If all edges have been traversed, stop.

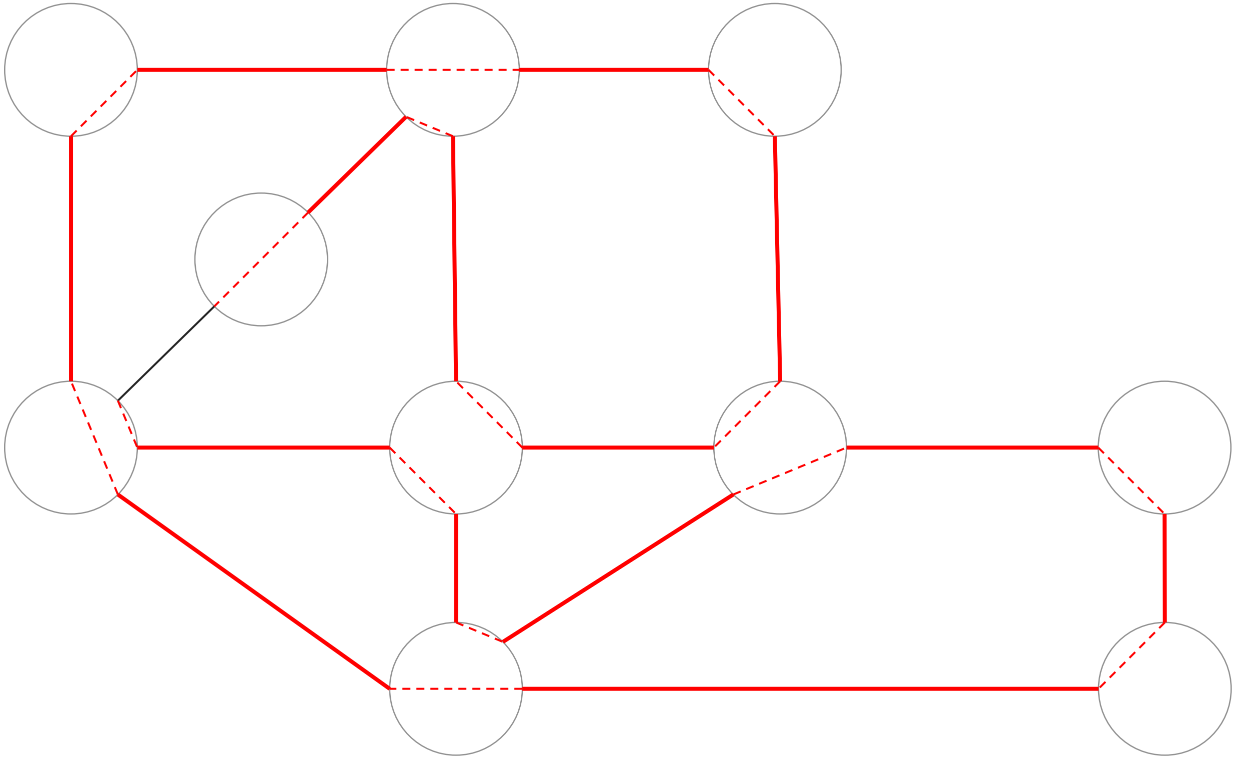
Step 3. Trace another cycle starting from an un-traversed edge incident to a node of the cycle. Merge the two cycles into one. Go to Step 2.

Traversing a Eulerian graph





Traversing a Eulerian graph



The Chinese Postman Problem

Guan, 1962

Let G=(V,E) be a connected undirected graph with costs $c_e \geq 0$ associated with its edges.

CPP: To find a minimum length tour traversing every edge at least once.

If G is Eulerian, the graph itself is the solution to the Chinese Postman Problem.

Otherwise, at least one of its edges will be traversed more than once. Therefore, we have the following equivalent augmentation problem:

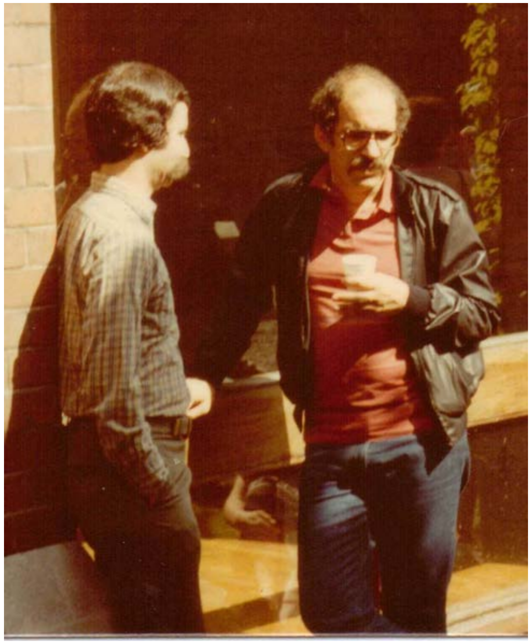
Find a set of edge copies with minimum total cost such that, when added to G, G becomes an even (Eulerian) graph.

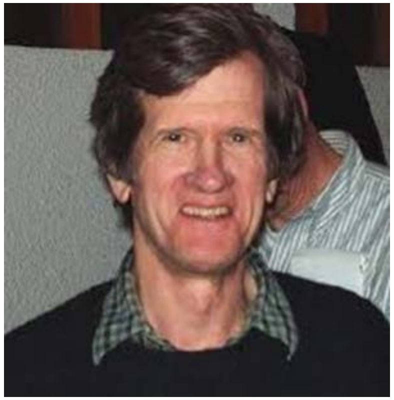
CPP: Resolution

Christofides, 1973
Edmonds and Johnson, 1973

1. Find the odd-degree vertices
2. Compute the shortest paths in G between every pair of odd nodes ($O(|V|^3)$)
3. Solve a Minimum Cost Matching Problem to match the odd nodes ($O(|V|^3)$)
4. For all pairs of matched odd nodes add to graph G a copy of the edges in the shortest paths $\rightarrow G'$.
5. Find a Eulerian tour in G'.

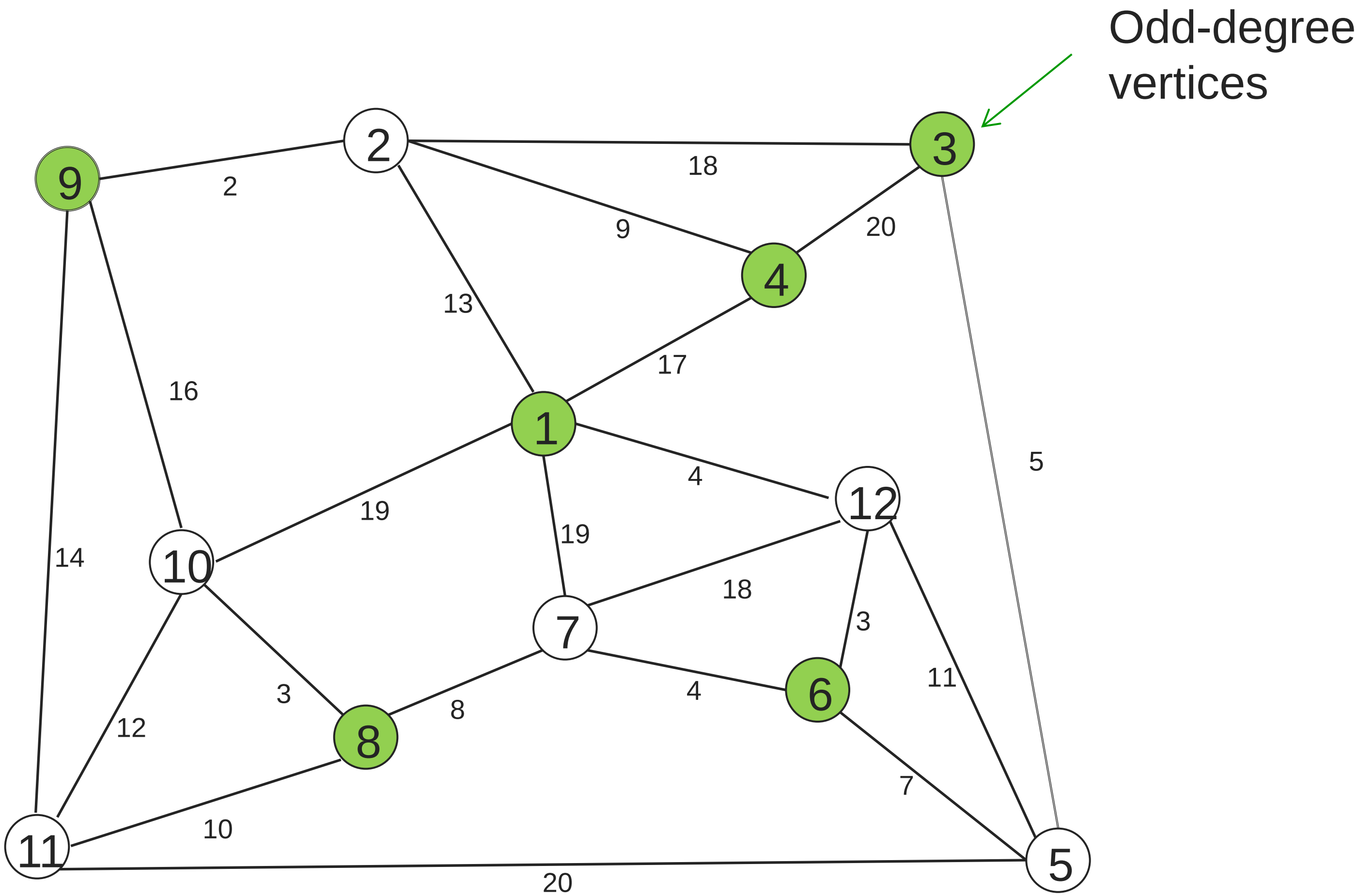
Pictures



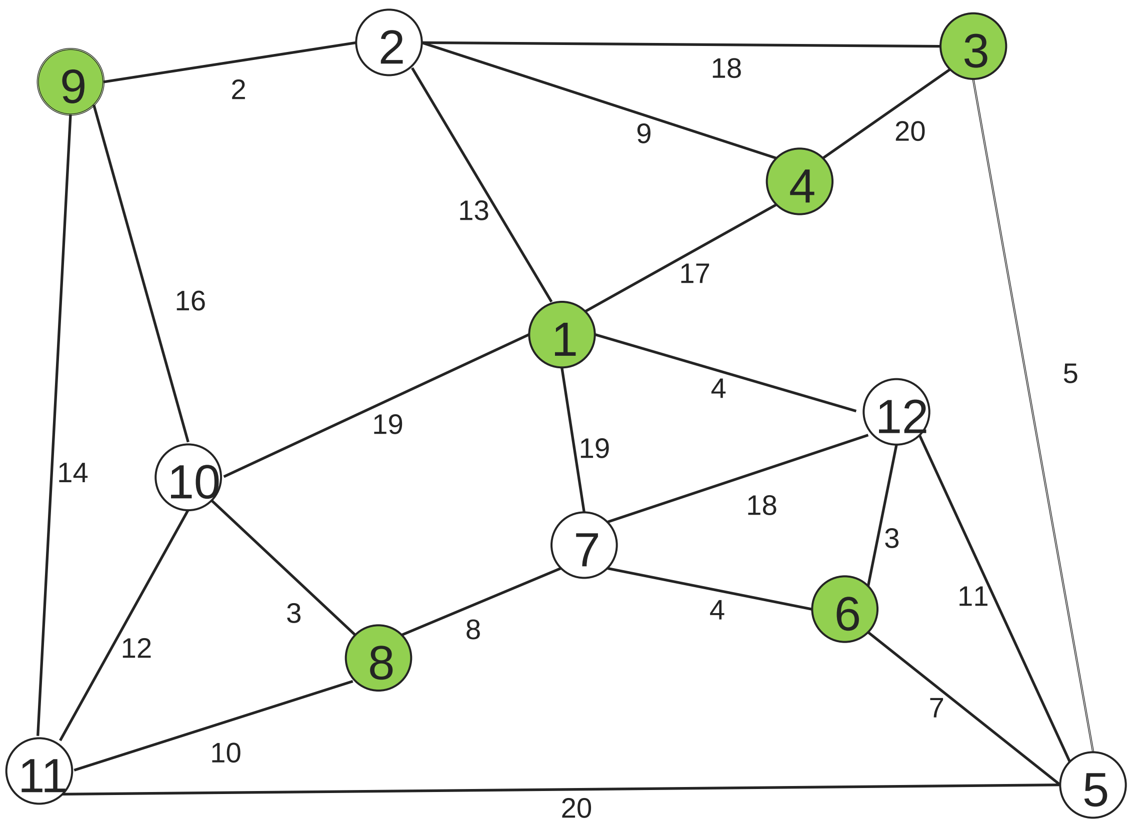


CPP: Resolution

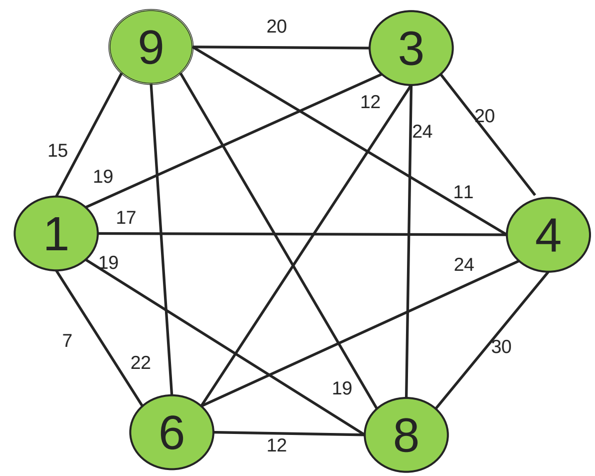
Graph G



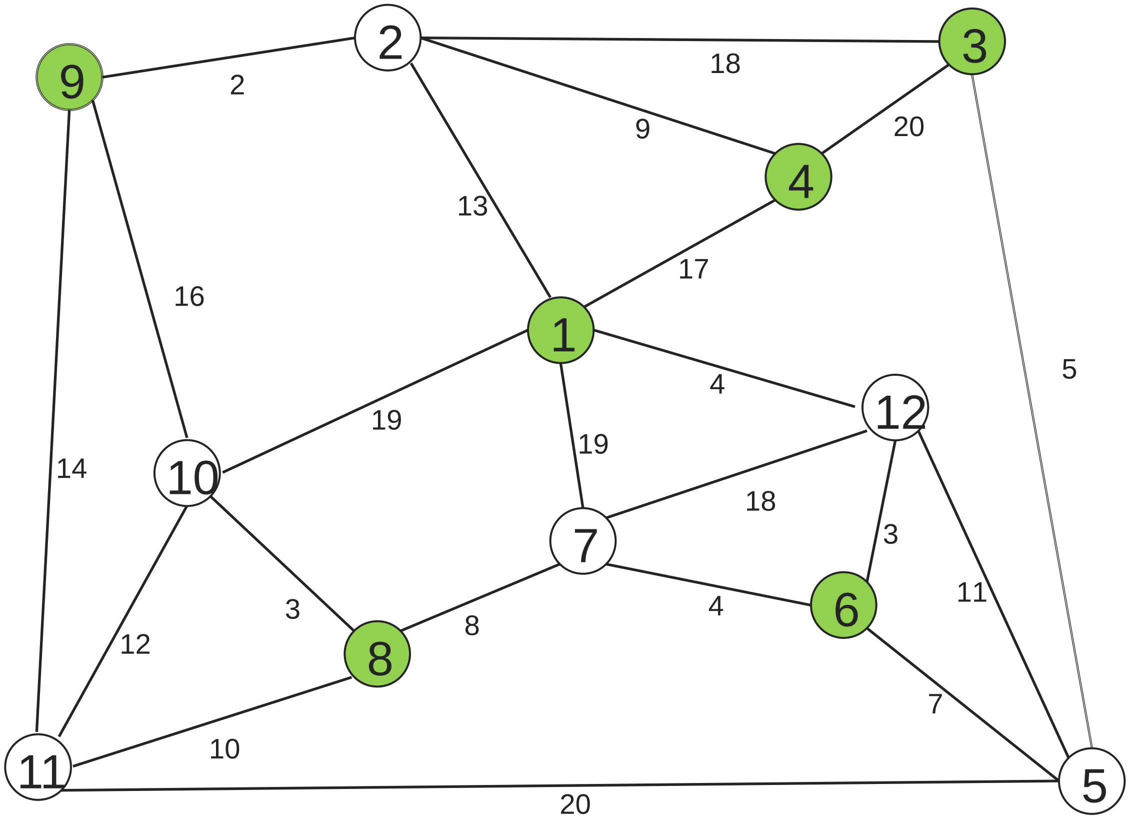
CPP: Resolution



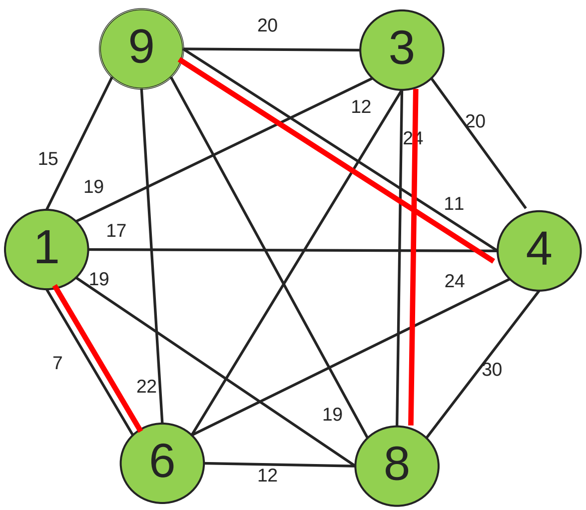
Shortest paths among odd-degree nodes



CPP: Resolution

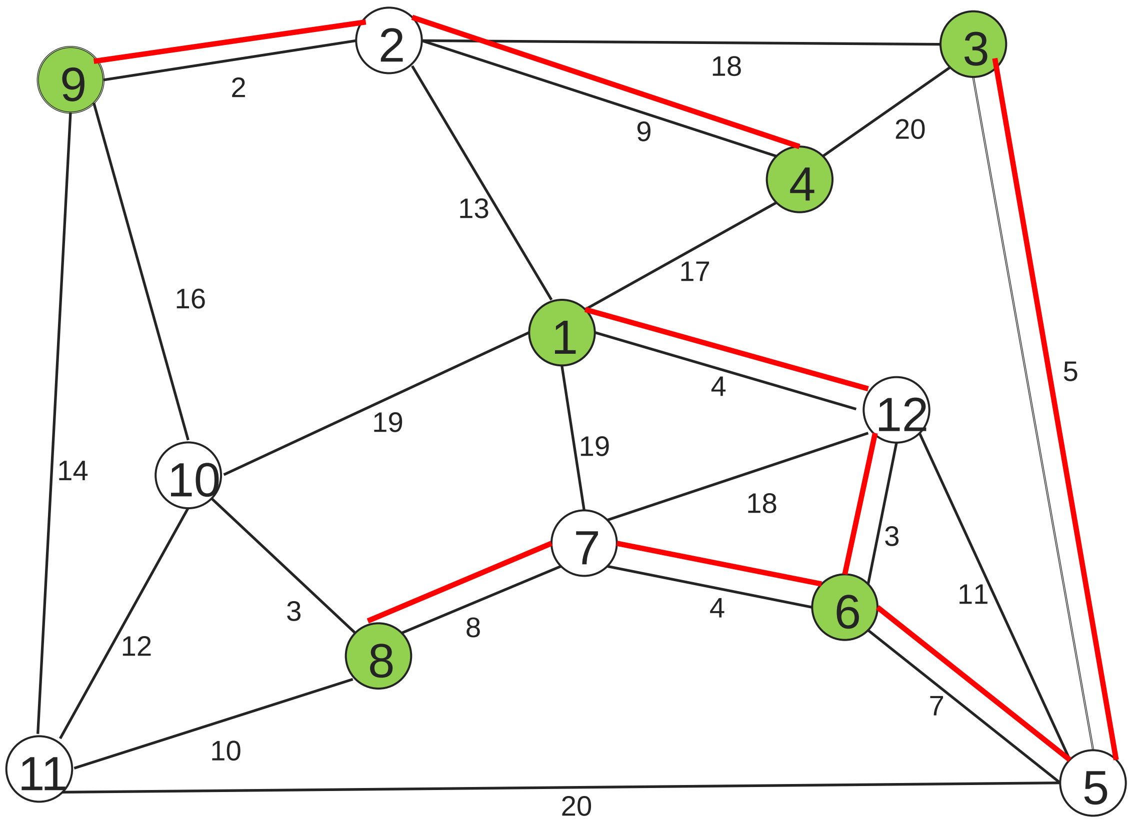


Minimum Cost Perfect Matching

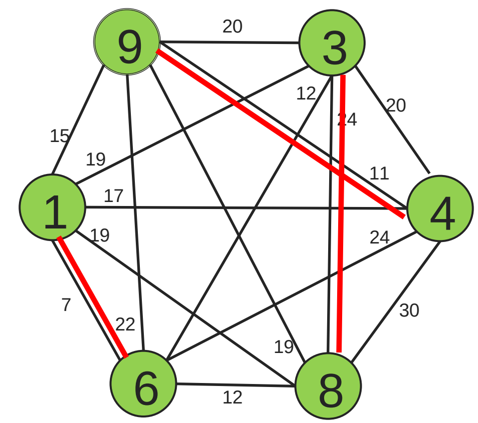


Cost = 7+24+11=42

CPP: Resolution



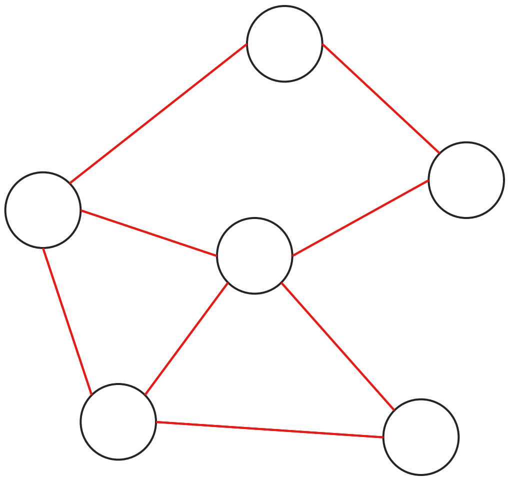
Duplicate shortest paths between odd nodes



Graph G'. It is Eulerian and corresponds to the optimal solution of the CPP on G.

CPP: Formulation

x_e=copies of e to be added to G in order to obtain a Eulerian graph.



CPP Formulation (Edmonds & Johnson, 1973) :

Minimize $\sum c_e x_e$

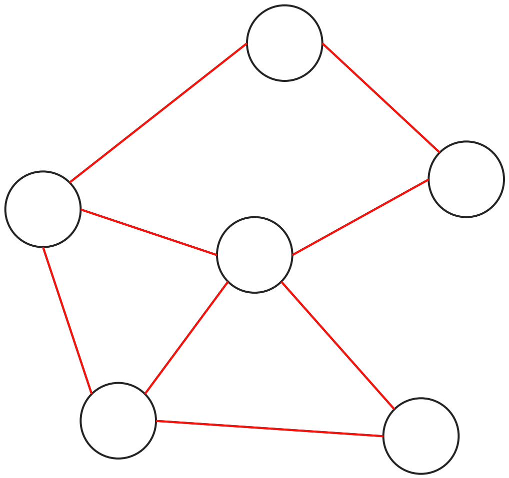
$x(\delta(v)) \equiv d(v) \pmod{2}, \forall v \in V$

$x_e \geq 0$ and integer, $\forall e \in E$

Parity
Non linear!!

$x(\delta(v)) \equiv d(v)$ is equivalent to $x(\delta(v)) + d(v) = 2 z_v$, $z_v \geq 1$ and integer

CPP: Formulation



x_e=copies of e to be added to G in order to obtain a Eulerian graph.

CPP Formulation (Edmonds & Johnson, 1973) :

Minimize $\sum c_e x_e$

exponential number !! → $x(\delta(S)) \geq 1, \; \forall S \subset V$ such that $|\delta(S)|$ is odd

$x_e \geq 0, \quad \forall e \in E$

Full polyhedral description

CPP: Odd cut inequalities

Parity is a fundamental issue in arc routing

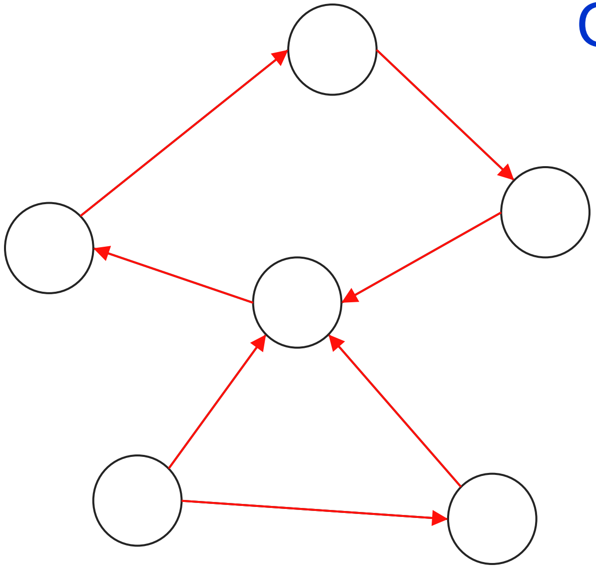
If an edge cutset contains an odd number of edges, at least one extra traversal will be needed

$x(\delta(S)) \geq 1, \forall S$ such that $|\delta(S)|$ is odd

Exact separation in polynomial time (Padberg and Rao, 1982)

Eulerian directed graphs

G=(V,A) strongly connected



The parity of the vertices is a necessary but not a sufficient condition for a directed graph to be Eulerian

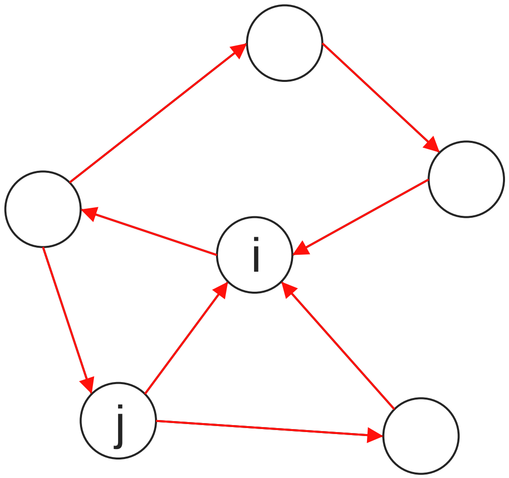
König (1936):

A strongly connected directed graph is Eulerian iff it is symmetric

(G is symmetric if $\forall i \in V$, # arcs entering at i = # arcs leaving i)

DCPP: Resolution

Liebling, 1970
Edmonds & Johnson, 1973



$d^+(i)=1$ $d^-(i)=3$ → supply(i)= $s_i = d^-(i) - d^+(i)$
$d^+(j)=2$ $d^-(j)=1$ → demand(j)= $t_j = d^+(j) - d^-(j)$

x_{ij}=copies of (i,j) to be added to G in order to obtain a Eulerian graph.

$$Min \sum_{i \in S, j \in T} c_{ij} x_{ij}$$

$$\sum_{i \in S} x_{ij} = t_j \quad \forall j \in T$$

$$\sum_{j \in T} x_{ij} = s_i \quad \forall i \in S$$

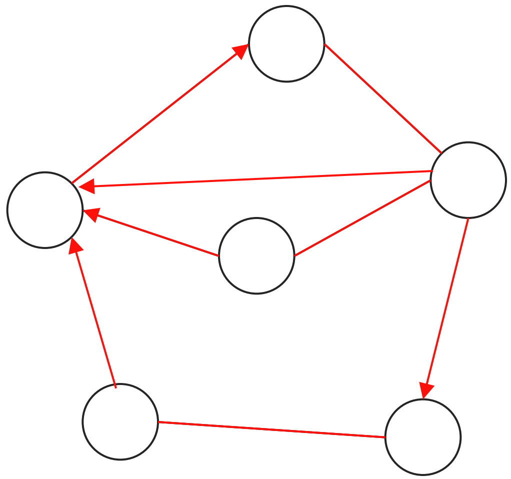
$$x_{ij} \geq 0$$

Polinomially solvable

Eulerian mixed graphs

G=(V,E,A) strongly connected

The parity of the vertices degree is again a necessary but not sufficient condition for a mixed graph to be Eulerian



Non Eulerian

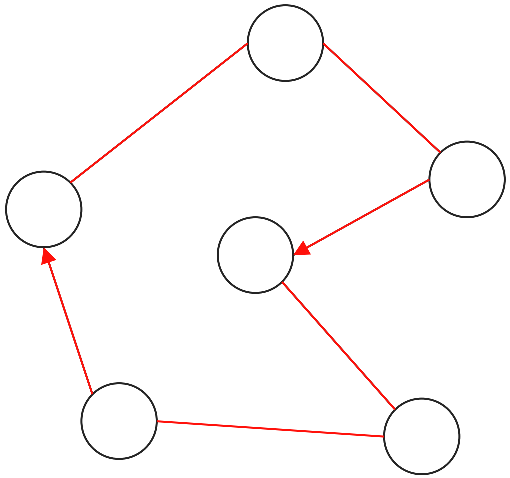
G=(V,E,A) is Eulerian if

G is even, and
G is symmetric

Are these conditions also necessary for G to be Eulerian ?

Eulerian mixed graphs

Obviously not, as the following figure shows:



Then, is there a necessary and sufficient condition for a mixed graph to be Eulerian ?

Eulerian mixed graphs

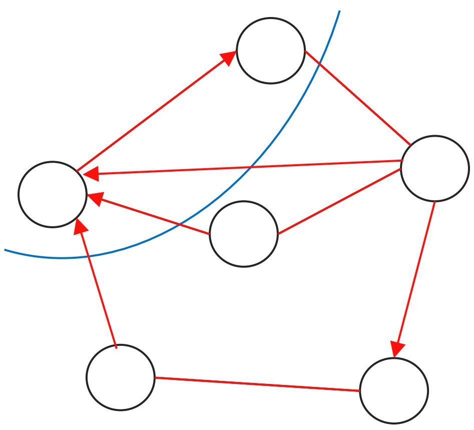
Ford and Fulkerson (1962)

G=(V,E,A) strongly connected is Eulerian iff

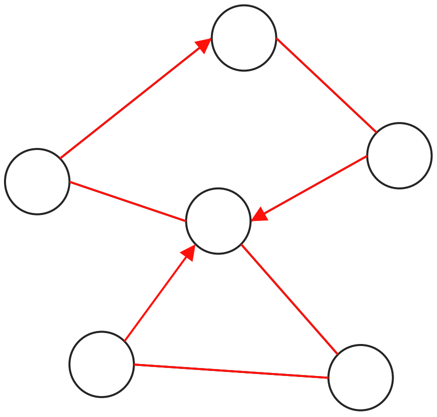
G is even, and

G is balanced, i.e. ∀S⊂V,

(arcs leaving S)-(arcs entering S) ≤ (edges between S and V\S)



Non balanced



Balanced

Pictures

Eulerian mixed graphs

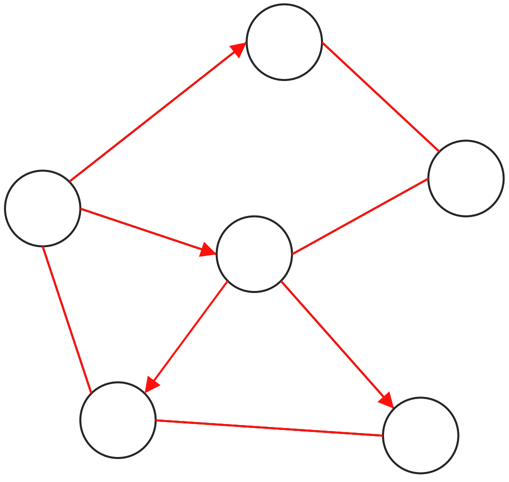
How can we check if a graph is balanced?

Nobert and Picard (1996) proposed a polynomial-time algorithm that finds a violated balanced inequality if it exists.

The Mixed Chinese Postman Problem (MCPP)

NP-hard (Papadimitriou,1976)

Polynomially solvable if G is even (Edmonds & Johnson,1973)



MCPP: Heuristic algorithms

The Edmonds and Johnson's exact algorithm for the case when *G* is even (called *Even MCPP Algorithm*) is the basis for two heuristics for the general case suggested by Edmonds & Johnson (1973) and developed and improved by Frederickson (1979):

Algorithm MIXED1 would be equivalent to first transforming *G* into an even graph and then applying the Even MCPP Algorithm.

MCPP: Heuristic algorithms

Algorithm MIXED2 can be considered as the reversed version of MIXED 1. It first solves a minimum cost flow problem in *G* to obtain a symmetric graph. Then, it solves the (undirected) CPP to finally obtain an even and symmetric graph.

MIXED1 and MIXED2, have a worst case ratio of 2, but the *Mixed Algorithm*, which consists of applying both heuristics and select the best tour obtained, has a worst case ratio of 5/3.

Raghavachary & Veerasamy (1998) proposed a modification to the Frederickson's Mixed Algorithm with a better worst case ratio of 3/2.

MCPP: Exact methods

Christofides, Benavent, Campos, C. & Mota (1984)

Branch & Bound based on Lagrangean relaxation

Nobert & Picard (1996)
C., Romero & Sanchis (2003)
C., Mejía & Sanchis (2005)

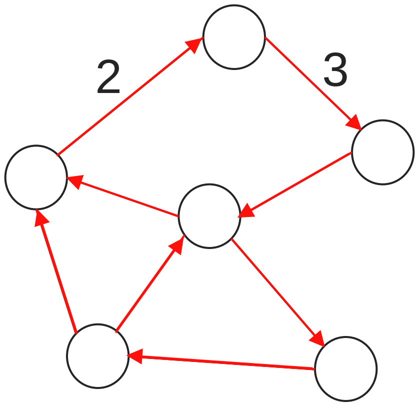
Branch & Cut based on an integer formulation

C., Plana, Oswald, Reinelt, Sanchis (2012)

Solve the MCPP as a special case of the Windy Postman Problem.

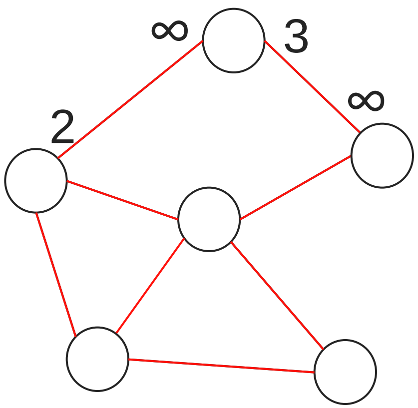
Branch & Cut capable of solving 17 out of 24 instances with |V|=3000, 1097≤|A| ≤6742 and 1992≤|E| ≤6799 in less than 15 minutes.

Routing problems on windy graphs



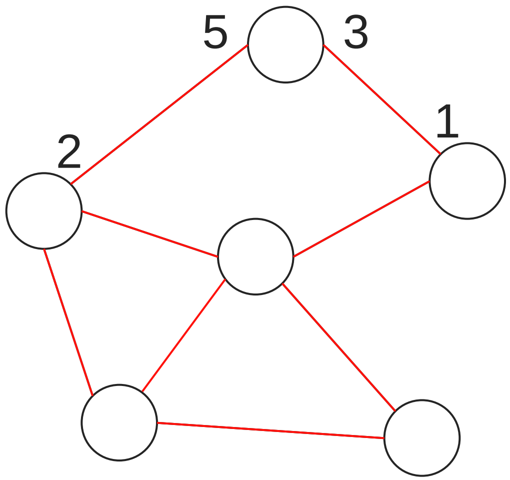
A “windy” graph is an undirected graph with asymmetric costs.

Undirected, directed and mixed graphs can be considered special cases of windy graphs.



Then, windy ARPs generalize the corresponding ARPs on undirected, directed and mixed graphs.

The Windy Postman Problem

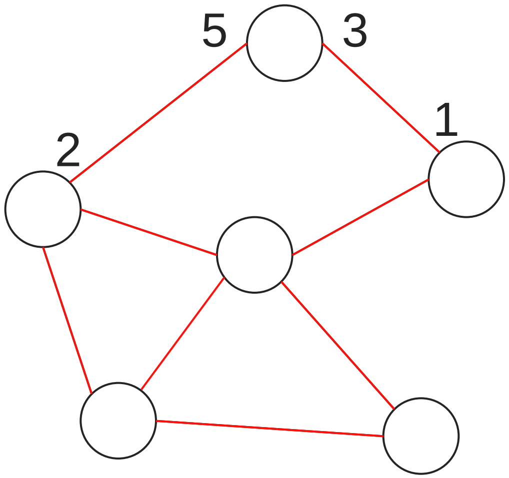


Minieka (1979)

(that the cost of traversing an edge is the same for either direction) “is hardly a good assumption when one direction might be uphill and the other downhill, when one direction might be with the wind and the other against the wind or when fares are different depending on direction”.

Given a windy graph G=(V, E), the WPP entails finding a minimum cost tour traversing all the edges in G at least once.

The Windy Postman Problem



WPP is NP-hard
(Brucker 1981 and Guan 1984)

Although some special cases can be solved in polynomial time:

-When the two orientations of every cycle C in G have the same cost (Guan 1984), and
-When G is even (Eulerian) (Win 1987)

The Windy Postman Problem

- Heuristic based on the solution of a minimum cost matching and then on a minimum cost flow problem (Win, 1989)

  Worst case ratio = 2

- Heuristic that interchanges the two steps above (Pearn & Li, 1994)

- LP-based heuristics (Win, 1987)

  Worst case ratio = 2

WPP formulation

Win (1987), Grötschel & Win (1992)

x_{ij} = # of times (i,j) is traversed from i to j

$$\text{Min } \sum_{(i,j)\in E}(c_{ij}x_{ij}+c_{ji}x_{ji})$$

$$x_{ij}+x_{ji} \geq 1,\ \forall(i,j)\in E \qquad (1)$$

$$\sum_{(i,j)\in\delta(i)}x_{ij} = \sum_{(i,j)\in\delta(i)}x_{ji},\ \forall i\in V \qquad (2)$$

$$x_{ij},\ x_{ji} \geq 0,\ \forall(i,j)\in E \qquad (3)$$

$$x_{ij},\ x_{ji} \text{ integer},\ \forall(i,j)\in E \qquad (4)$$

WPP exact algorithms

Win (1987), Grötschel & Win (1988):

Cutting-plane algorithm: solved 31/36 instances with $|V| \in (52,264)$ and $|E| \in (78,479)$

C., Plana, Sanchis (2006)

B&C

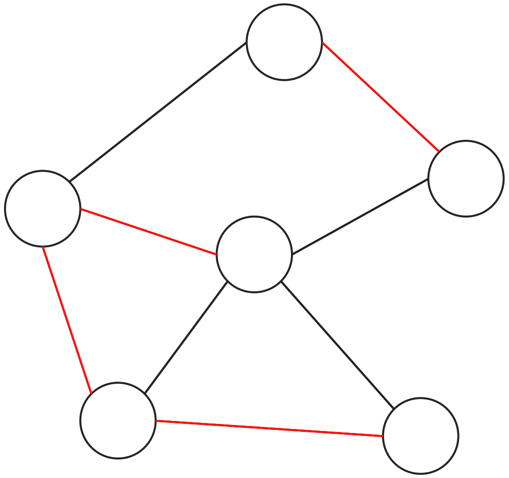
C., Oswald, Plana, Reinelt, Sanchis (2011):

B&C: solved 99/120 instances with $|V| \in (500,3000)$ and $|E| \in (813,9085)$

The Rural Postman Problem

Orloff (1974)

NP-hard (easy transformation from the TSP) Lenstra & Rinnooy Kan (1976)



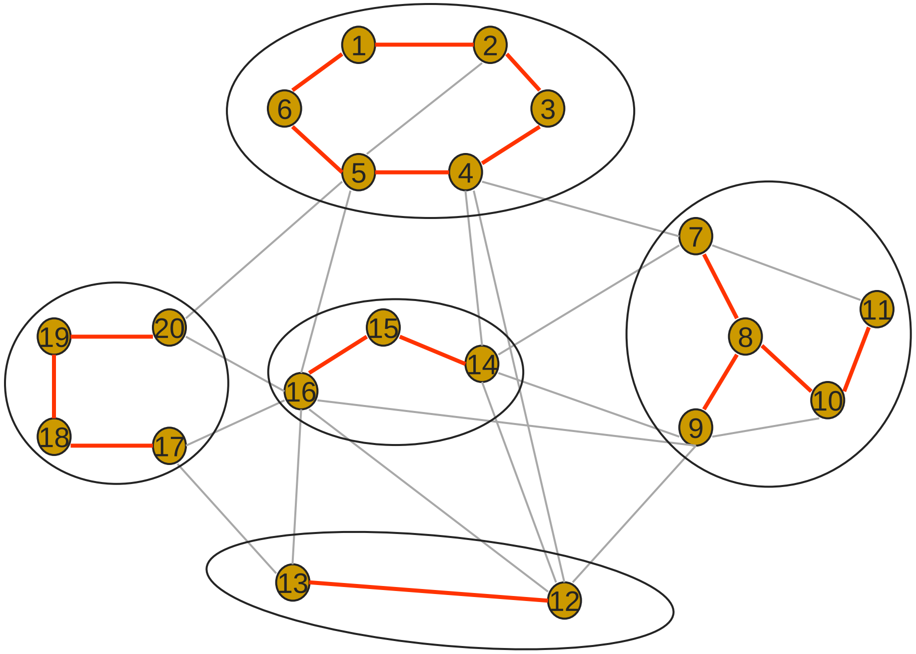
$G^R=(V,E_R)$ non connected

Polynomially solvable if G^R is connected.

Its difficulty increases with the number of R-sets.

The Rural Postman Problem

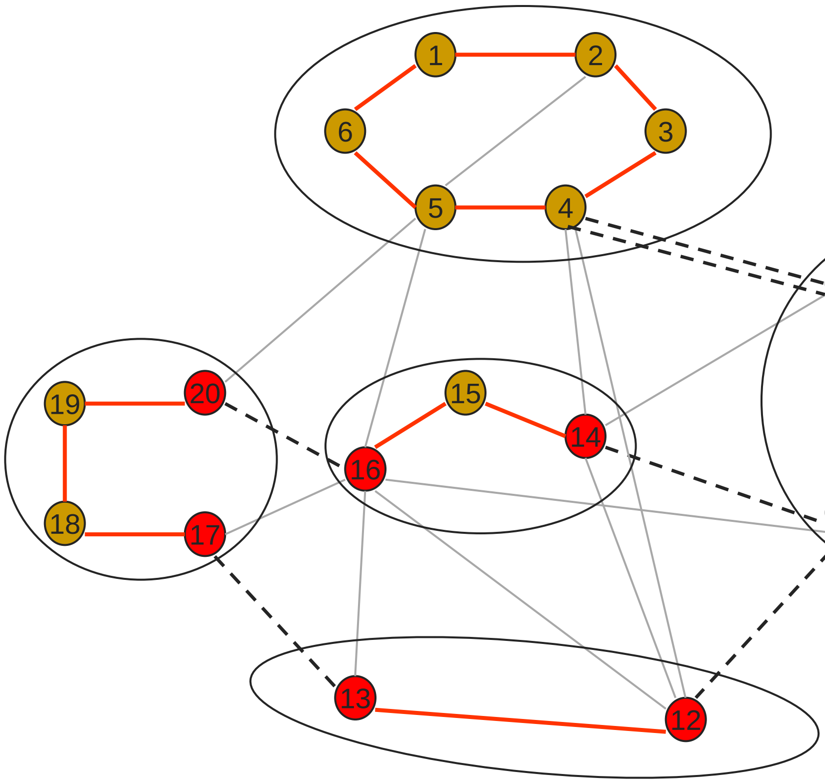
Equivalent augmentation problem



Add to G^R a set of edge copies with total minimum cost such that the resulting graph is connected and even.

The Rural Postman Problem

added edges



Feasible solution

RPP formulation

C. & Sanchis, 1994

x_e=copies of e to be added to G^R in order to obtain a Eulerian graph

$$\text{Minimize } \sum c_e x_e$$

$$x(\delta(S)) \geq 2, \quad \forall S \subset V,\ \delta_R(S)=\emptyset$$

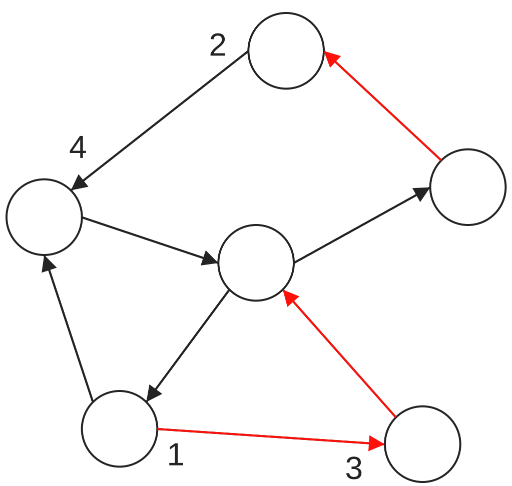
$$x(\delta(i)) \equiv |\ \delta_R(i)\ |\ (\text{mod. } 2), \quad \forall i \in V$$

$$x_e \geq 0 \text{ and integer} \quad \forall e \in E$$

where $\delta_R(S) = \delta(S) \cap E_R$

The General Routing Problem

Orloff (1974)



- Required links (arcs, edges)
- Required vertices

- On undirected graphs (GRP)
- On directed graphs (DGRP)
- On mixed graphs (MGRP)
- On “windy” graphs (WGRP)

Special Cases

Chinese Postman Problem (CPP)

- No required vertices ($V_R = \varnothing$)
- All links are required ($E_R = E$)

Rural Postman Problem (RPP)

- No required vertices ($V_R = \varnothing$)

Graphical TSP (GTSP)

- No required links ($E_R = \varnothing$)

GRP exact methods

C., Plana & Sanchis (2007)

Branch-and-cut algorithm for the WGRP (and special cases)

solves optimally

- WRPP instances with $|V| = 988, |E| \leq 3952$ and up to $150\ R-$components
- MRPP instances with $|V| \leq 999, |E| \leq 1969, |A| \leq$ 1984, and up to 188 R−components
- (undirected) RPP instances with $|V| \leq 1000, |E| \leq 3083,$ and $204\ R-$components
- DRPP instances with $|V| \leq 999\ |A| \leq 3139,$ and 213 R−components

The Capacitated Arc Routing Problem

Golden & Wong (1981)

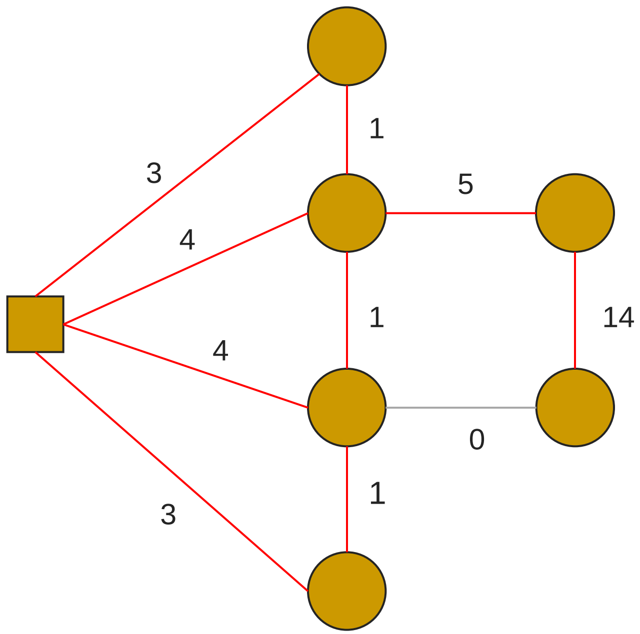
Let $G = (V, E)$ be an undirected and connected graph, with

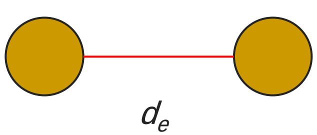
- E_R set of demand (required) edges,
- c_e cost of traversing edge $e \in E$,
- d_e demand of edge $e \in E_R$,
- K = fleet of vehicles with capacity Q
- depot ∈ V (node d)

The CARP consists of finding a minimum cost set of vehicle routes (tours) such that:

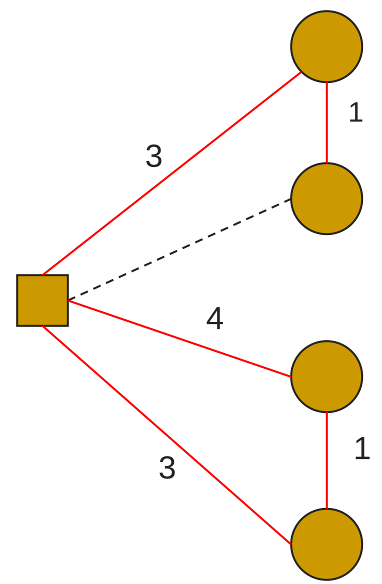
- each required edge is serviced by only one vehicle,
- each route starts and ends at the depot, and
- the demand serviced in each route does not exceed the vehicle capacity Q

The Capacitated Arc Routing Problem

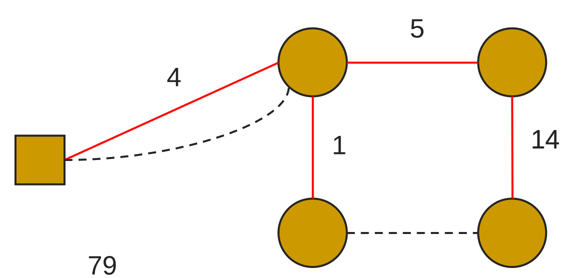




Capacity Q = 25



Route 1
Load = 12



Route 2
Load = 24

Heuristic methods for the CARP

Many heuristics and metaheuristics have been proposed for the CARP and its many variants.

Prins (2014), and

Muyldermans & Pang (2014)

are two excellent surveys on the topic.

Exact methods for the CARP

See Belenguer, Benavent & Irnich (2014)

- Branch-and-bound: Hirabayashi, Saruwatari & Nishida (1992)
- Transformation to node routing
  - Branch-and-cut: Baldacci & Maniezzo (2006)
  - Branch-and-price: Longo, Poggi de Aragao & Uchoa (2006)
  - Cut-and-column generation: Bartolini, Cordeau & Laporte (2011)
- Two-index formulation: Belenguer (1990), Belenguer & Benavent (1998)
- One-index formulation: Letchford (1997), Belenguer & Benavent (1998,2003)
- Branch-and-price: Bode & Irnich (2012), Martinelli, Pecin, Poggi de Aragao & Longo (2011)

Two-index formulation

Belenguer & Benavent (1998)

$$x_e^k = \begin{cases} 1 & \text{if edge } e \text{ is serviced by vehicle } k \\ 0 & \text{otherwise} \end{cases}$$

$$y_e^k = \text{ number of times vehicle } k \text{ deadheads through edge } e \in E$$

$$\text{Min} \sum_{k \in K} \sum_{e \in E} c_e{}^{serv} x_e^k + \sum_{k \in K} \sum_{e \in E} c_e y_e^k$$

Two-index formulation

$$\text{Min} \sum_{k \in K} \sum_{e \in E} c_e^{\,serv} x_e^k + \sum_{k \in K} \sum_{e \in E} c_e y_e^k$$

$$\sum_{k \in K} x_e^k = 1 \qquad \forall e \in E_R$$ assignment

$$\sum_{e \in E_R} d_e x_e^k \leq Q \qquad \forall k \in K$$ capacity

$$x^k\big(\delta_R(i)\big) + y^k\big(\delta(i)\big) \equiv 0 \ (mod.\,2), \qquad \forall i \in V, k \in K$$ parity

$$x^k\big(\delta_R(S)\big) + y^k\big(\delta(S)\big) \geq 2x_f^k \qquad \forall S \subseteq V \backslash \{d\}, f \in E_R(S), k \in K$$ connectivity

$$x_e^k \in \{0,1\},\ y_e^k \geq 0 \text{ and integer}$$

Two-index formulation

- The branch-and-cut based on this formulation was able to solve only small size instances.
- The lower bound obtained with the linear relaxation is very bad if aggregate constraints (R-odd cut and capacity) are not used.
- The formulation has a high degree of symmetry: the vehicle routes can be permuted leading to different integer solutions that are in fact identical. Many nodes of the branch-and-cut tree are identical.

One-index formulation

y_e = total number of times that edge $e \in E$ is deadheaded by all the vehicles

$$y_e = \sum_{k \in K} y_e^k$$

$$\text{Min} \quad C_T + \sum_{e \in E} c_e y_e$$

$$y(\delta(S)) \geq 2K(S) - |\delta_R(S)| \qquad \forall S \subseteq V \backslash \{d\}$$ Aggregate capacity

$$y(\delta(S)) \geq 1 \qquad \forall S \subseteq V \backslash \{d\}, \text{such that } |\delta_R(S)| \text{ is odd}$$ R-odd cut

$$y_e \geq 0 \text{ and integer}$$

$$C_T = \sum_{k \in K} \sum_{e \in E} c_e^{serv}$$

$K(S) = \left\lceil \frac{d(E_R(S) \cup \delta_R(S))}{Q} \right\rceil$ = lower bound on the number of vehicles needed for $(E_R(S) \cup \delta_R(S))$

One-index formulation

The one-index formulation allows non-feasible integer solutions

Given the optimal solution of the one-index formulation,

Is it possible to disaggregate the solution into $|K|$ routes?

NP-complete problem

Bin Packing Problem: BP(d1,d2,d3,d4,d5) ≤ 2 ?
capacity Q

One-index formulation

Cutting plane algorithm proposed by Belenguer & Benavent (2003)

Benchmark sets of instances

gdb # 23 $7 \leq |V| \leq 27, \quad 11 \leq |E| \leq 55$ (all edges required)

Golden, Dearmon & Baker (1983)

val # 34 $24 \leq |V| \leq 50, \quad 34 \leq |E| \leq 97$ (all edges required)

Benavent, Campos, C. & Mota (1992)

egl # 24 $77 \leq |V| \leq 140, \quad 98 \leq |E| \leq 190$

Li & Eglese (1996)

One-index formulation

Belenguer & Benavent (2003)

Can be used to prove the optimality of a heuristic solution or to provide a guarantee of its quality.

gdb	#optimality proofs	14/23,	average gap	0.14%
val	#optimality proofs	22/34,	average gap	0.41%
egl	#optimality proofs	0/24,	average gap	2.40%

Ahr (2004) and Martinelli, Poggi de Aragão & Subramaniam (2013) propose exact algorithms and dual ascent methods for separating capacity constraints that improve the lower bound obtained in some instances, but at a large computational effort.

Set-Covering formulations

The one-index formulation provides good lower bounds and is very fast, but no enumeration method has been implemented from it. It seems a very difficult task.

On the other hand the two-index formulation has the drawback of its high degree of symmetry, thus producing huge branch-and-cut trees.

The alternative is column generation based on set-partitioning or set-covering formulations

Set-Covering formulations

Ω set of all feasible CARP tours (routes for one vehicle)

$$a_{er} = \begin{cases} 1 & \text{if tour } r \in \Omega \text{ services edge } e \in E_R \\ 0 & \text{otherwise} \end{cases}$$

c_r total cost of tour $r \in \Omega$

$$\lambda_r = \begin{cases} 1 & \text{if tour } r \in \Omega \text{ is used in the solution} \\ 0 & \text{otherwise} \end{cases}$$

$$\text{Min} \sum_{r \in \Omega} c_r \lambda_r$$

$$\sum_{r \in \Omega} a_{er} \lambda_r \geq 1 \qquad \forall\, e \in E_R$$

$$\lambda_r \in \{0,1\} \;\; \forall r \in \Omega$$

Set-Covering formulations

Ω set of all feasible CARP tours (routes for one vehicle)

$$\text{Min} \sum_{r \in \Omega} c_r \lambda_r$$

$$\sum_{r \in \Omega} a_{er} \lambda_r \geq 1 \qquad \forall e \in E_R$$

$$\lambda_r \in \{0,1\} \ \forall r \in \Omega$$

The number of feasible CARP tours $|\Omega|$ is huge

The linear relaxation of SCF is solved by column generation: columns (tours) are dynamically generated as needed.

The integer program is solved by Branch-and-price

Cut-and-column generation

Gómez-Cabrero, Belenguer & Benavent (2005)

	Column-generation	Cut-and-column-generation	Cutting-plane generation *
gdb	4.92	0.07	0.13
val	7.21	0.39	0.66
egl	-	2.36	2.69

One of the drawbacks of the method is that the sparseness of the original graph is lost when solving the subproblem

Letchford & Oukil (2009) proposed a method to solve the subproblem that works on the original graph, thus avoiding this problem. Unfortunately they do not add cutting planes

Cut-first branch-and-price second

Bode & Irnich (2012)

They develop an exact method that works on the original sparse graph and integrates the cut-and-column generation into branch-and-price scheme

They add to the Set Covering model:

Variables $z_e \geq 0$, for all $e \in E$, that model deadheading cycles

Non-negative reduced costs are obtained

Adapt the labeling algorithm of Letchford & Oukil (2009) that works on the original graph

Cut-first branch-and-price second

Bode & Irnich (2012)

maximum CPU time: 4 hours

gdb : all 23 instances were optimally solved

val : all 34 instances solved

egl : 6 out of 24 instances optimally solved

Column generation on the GVRP

Bartolini, Cordeau & Laporte (2013)

The method by BCL, based on a transformation of the CARP into a Generalized Vehicle Routing Problem, shows slightly better results.

gdb : all 23 instances were optimally solved

val : 28 out of 34 instances solved

egl : 10 out of 24 instances optimally solved

Better lower bounds at the root node

Contents

- Introduction
- Applications
- Eulerian graphs and the Chinese postman problem
- The RPP, GRP and CARP
- Perspectives
  - Arc routing problems with profits
  - Arc routing problems with aesthetic constraints

Arc routing problems with profits

- Routing problems deal with the design of routes (for one or more vehicles).
- In most of these problems the objective is to service a given set of customers, with total minimum cost.
- In others, the objective is to select some customers with maximum profit from a set of potential customers and to service them.

Arc routing problems with profits

“Nowadays it is more and more frequent that demands for transportation services are posted on the web, usually in specific databases, and the carriers can pick up these demands and offer their service to some of these customers, possibly in the framework of an electronic auction. The carrier has to select within a set of potential customers those which are most convenient for him. In an electronic auction, the carrier will put a bid on these potential customers”.

(Archetti, Hertz and Speranza, 2005)

Arc routing problems with profits

In Feillet, Dejax & Gendreau (2005) these problems are called routing problems with profits and a classification is proposed:

- **Prize-collecting problems**: there is a lower bound on the total prize collected and the objective is to minimize the total cost.
- **Profitable problems**: the objective is to maximize the difference between the collected profits and the routing costs.
- **Orienteering problems**: there is an upper bound on the cost or length of the route and the collected profits are maximized (with multiple vehicles, they are called team orienteering problems.

Archetti and Speranza (2014) is an excellent survey of Arc Routing Problems with Profits.

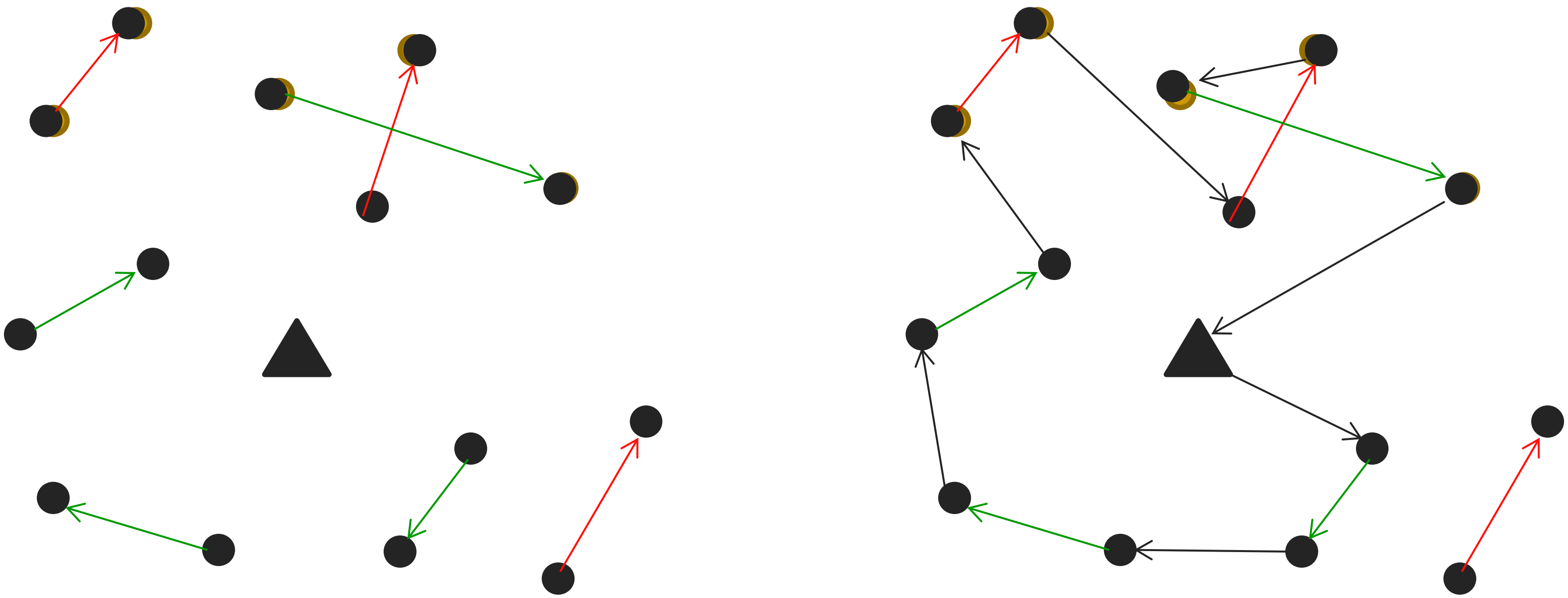
Problem	Proposed by	Studied by
Maximum Benefit CPP Special cases: Privatized RPP Prize-collecting RPP	Malandraki & Daskin (1993)	Pearn & Wang (2003) Pearn & Chiu (2004) Aráoz et al. (2006, 2009) C. et al. (2013)
Profitable DRPP Profitable WRPP Profitable Mixed CARP	Archetti et al. (2014) Schaeffer et al. (2014) Benavent et al. (2014)	Colombi and Mansini (2014) Ávila, C., Plana, Sanchis (2015)
Profitable Arc Tour problem	Feillet, Dejax, Gendreau (2005)	
Undirected CARP with profits	Archetti et al. (2010)	Zachariadis & Kiranoudis (2011)
Clustered Prize-collecting ARP Windy CPARP	Aráoz et al. (2009)	 C. et al. (2011)
Team orienteering ARP Orienteering ARP	Archetti et al. (2015a, b) Archetti et al. (2015c)	

Arc routing problems with profits

- In Archetti, C., Plana, Sanchis and Speranza (2015a, 2015b, and 2015c) the Team Orienteering ARP and the single vehicle version (the Orienteering ARP) are studied.
- The study was motivated by a real life application related to carriers making auctions on the web for transportation services.
- A transportation service is represented by an arc, and consists of reaching a node with an empty truck, filling the truck with load, traversing the arc and downloading the truck completely.
- The carrier has a set of regular customers which need to be served.
- The carrier has a vehicle or a fleet of vehicles with limited traveling time and looks for additional customers to fully use the traveling time of the vehicles.

The Orienteering ARP

Given a set of regular customers (green arcs) and given a set of potential customers (red arcs), we want to select a subset of potential customers with maximum profit that can also be serviced within the vehicle time limit.

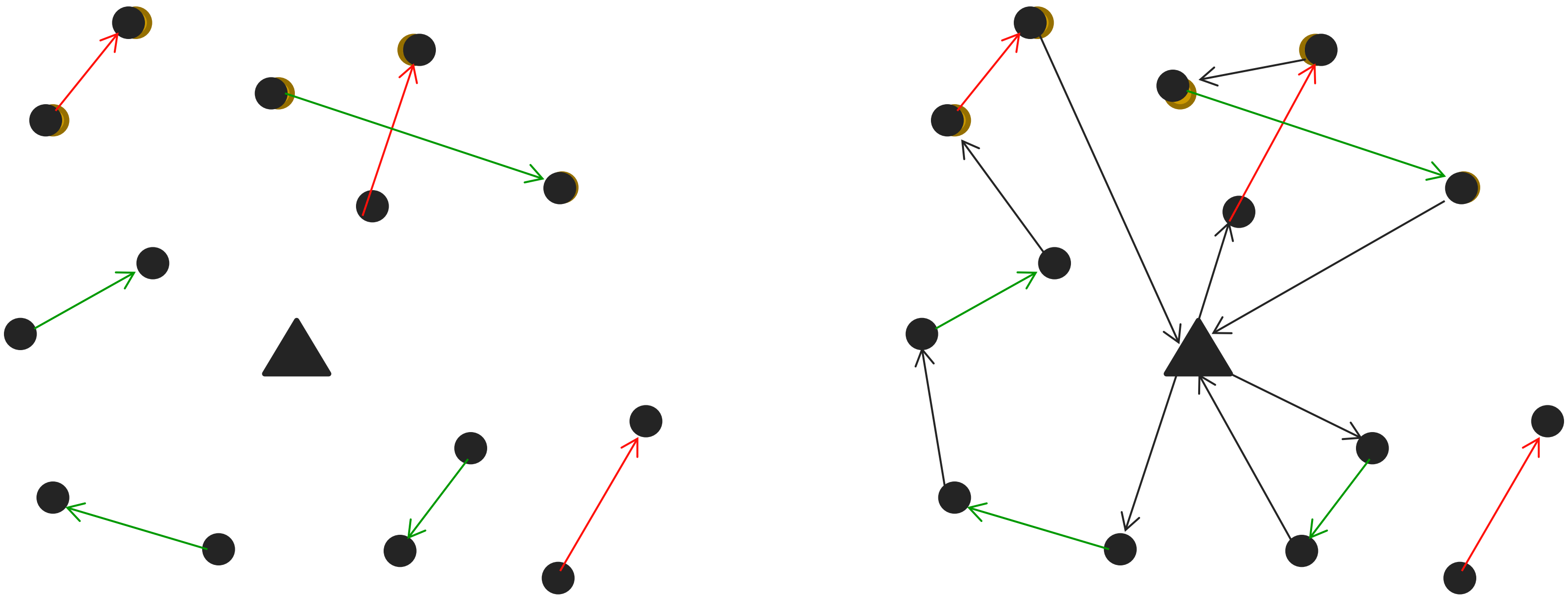


The Orienteering ARP

The Orienteering Arc Routing Problem, OARP, consists of finding a route starting and ending at the depot, such that

- its cost or time is no greater than a time limit Tmax,
- all the arcs associated with required customers are traversed at least once, and
- the sum of the profits of the traversed arcs associated with the potential customers is maximum.

The Team Orienteering ARP



The Team Orienteering ARP

The Team Orienteering Arc Routing Problem, TOARP, is defined as finding K routes starting and ending at the depot, such that

- each route is no greater than a time limit Tmax,
- all the arcs associated with required customers are traversed at least once, and
- the sum of the profits of the traversed arcs associated with the potential customers is maximum.

B&C for the OARP

- Run with a time limit of 1 hour.
- The instances have $1000 \leq |V| \leq 2000$ and $7000 \leq |A| \leq 14000$.
- 79 out of 80 instances with 1000 vertices and 7000 arcs were solved optimally.
- 76 out of 80 instances with 1500 vertices and 10500 arcs were solved optimally.
- 64 out of 80 instances with 2000 vertices and 14000 arcs were solved optimally.

B&C for the TOARP

- Run with a time limit of 1 hour.
- The instances have $11 \leq |V| \leq 100$, $42 \leq |A| \leq 846$ and $K=2,3,4$.
- 204 out of 207 instances with $K=2$ were solved to optimality.
- 188 out of 207 instances with $K=3$ were solved to optimality.
- 157 out of 207 instances with $K=4$ were solved to optimality.

Contents

- Introduction
- Applications
- Eulerian graphs and the Chinese postman problem
- The RPP, GRP and CARP
- Perspectives
  - Arc routing problems with profits
  - Arc routing problems with aesthetic constraints

ARPs with aesthetic constraints

Real world applications often require other constraints that must be added to the basic ARP models.

Examples of such situations arise when workloads need to be equitably distributed among the vehicles, or different vehicle routes have to be constrained to separated geographical regions.

Ghiani et al. (2014) summarize strategical and tactical issues involving these type of constraints in waste collection problems.

Mourgaya & Vanderbeck (2007) and Muyldermans et al. (2002) point out that too many intersections of the service areas of different vehicles can complicate the activities to be held in a region.

ARPs with aesthetic constraints

Kim, Kim & Sahoo (2006) and Poot, Kant & Wagelmans (2002) report that solutions with an excessive number of vehicle croosovers tend to be rejected by practitioners.

Kim et al. also remark that the overlapping of service areas is strongly related to the intersection of the vehicle routes. The number of intersections may decrease if each vehicle service area is concentrated in a geographical region.

How can we define “nice” regions (sets of arcs and/or edges)?

Besides being separated and workload balanced, their shape should have other “nice” characteristics, like connectivity, non-overlapping and “compacteness”.

ARPs with aesthetic constraints

Compactness is one of the most frequently mentioned characteristics, although not always is clearly defined.

Furthermore, the meaning of compactness slightly differs from author to author. It uses to be associated with:

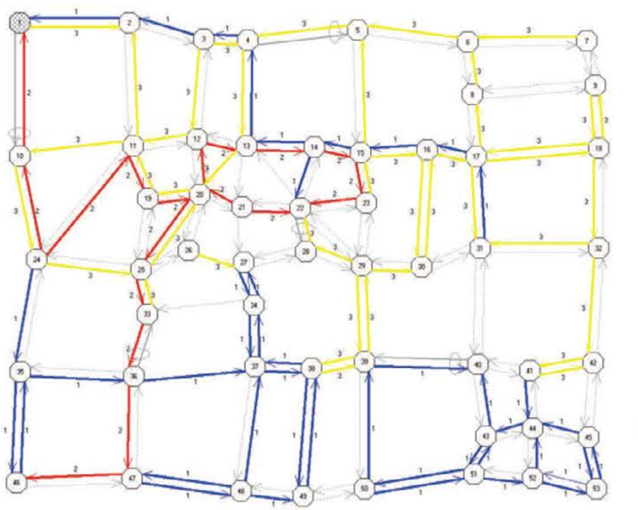
a) zones shapes as close as possible to circles, squares or rectangles,

b) geographically or visually compact zones, or

c) the proximity between the demand entities in the same zone.

ARPs with aesthetic constraints

Constantino, Gouveia, Mourao, Nunes (2015)

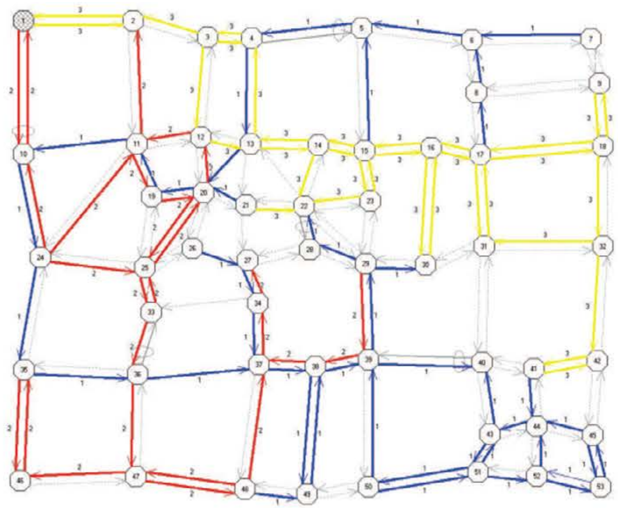


(a) MCARP

(a) Optimal MCARP solution: routes overlapping, not “nice” regions served by each route and disconnected sequence of required links serviced by each vehicle.

ARPs with aesthetic constraints

Constantino, Gouveia, Mourao, Nunes (2015)

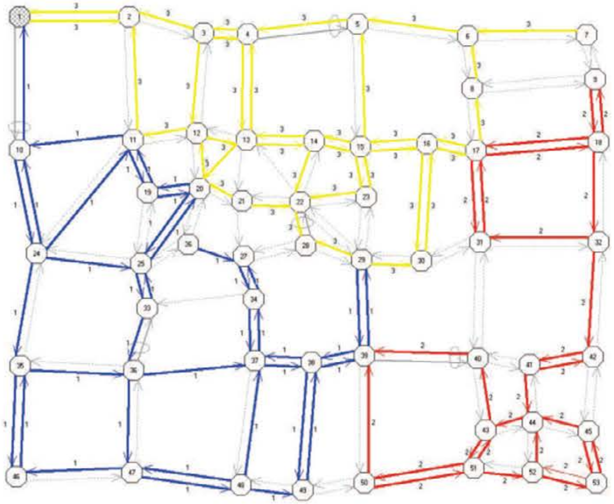


(b) Connectivity model

(b) Connectivity solution: Optimal MCARP solution after adding constraints forcing the required links in each route to define a connected subtgraph. It still shows routes that overlap and spread in the collection zone.

ARPs with aesthetic constraints

Constantino, Gouveia, Mourao, Nunes (2015)



(c) BCARP

(c) BCARP (bounded overlapping MCARP) solution: This model contains a constraint based on a measure of the non-overlapping of the routes (in terms of the number of nodes that are common to the required links serviced by the different routes)

ARPs with aesthetic constraints

- Constantino et al. (2014) define variables n_i^k = 1 if vertex i is an end node of a required link served in route k. The sum for all routes and all the vértices is used to measure the overlapping.

They solve an assignment problem of required links to routes that satisfies the capacity constraints and minimize the overlapping.

The optimal value n^* of the above problem is used to define an upper bound for the non-overlapping constraint:

$$\sum_i \sum_k n_i^k \leq n^*$$

that is added to the original MCARP problem to get the new BCARP model.

Conclusions

- The Chinese Postman, the Rural Postman and General Routing problems can be optimally solved for large instances in the undirected, directed, mixed and windy cases.
- Arc Routing problems with several vehicles, as the CARP, are much more difficult.
- There is no need for sophisticated heuristics for solving most ARPs with a single vehicle. However, they are needed for ARPs with several vehicles.

Conclusions

- New methods (and ideas) are needed to solve the CARP and other ARPs with several vehicles.
- Models for arc routing problems incorporating profits and/or aesthetic constraints like balanced workload and non-overlapping will be the subject of study in the next years.

Many thanks for your attention !!