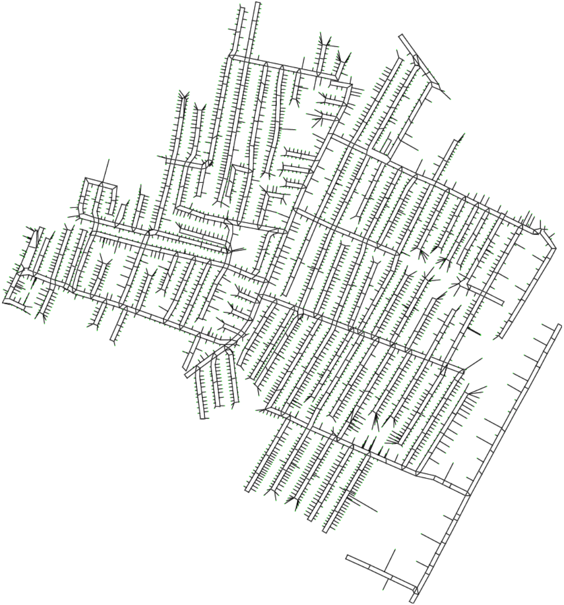
Optical distribution network planning

Alexander Maximenko

HUAWEI TECHNOLOGIES CO. LTD (Russia)

Moscow, 30.10.2020

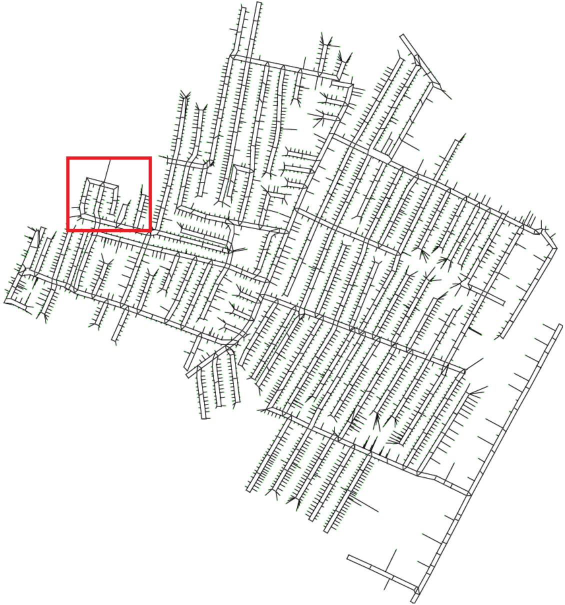
Example of a planar graph



Given:
a planar graph with
$\approx 10^4$ nodes,
$\approx 10^4$ edges.

Every edge is a channel where cable may be laid

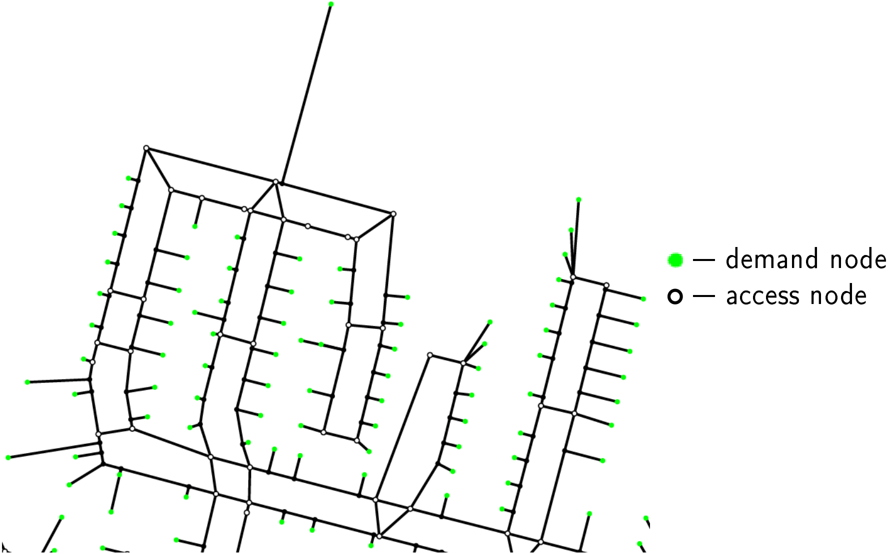
Example of a planar graph



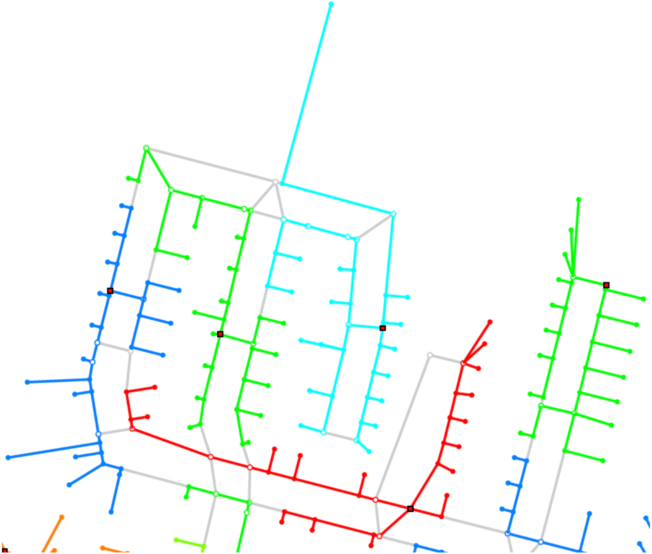
Given:
a planar graph with
$\approx 10^4$ nodes,
$\approx 10^4$ edges.

Every edge is a channel where cable may be laid

Fragment of a graph



Example: clustering with minimal number of clusters (FATs)



■ — Fiber Access Terminal (FAT)

FAT is connected to $\leq C_{max}$ demands

Clusters are disjoint

The problem formulation

Given

1. Set of nodes $V = D \cup U \cup W$,
   D — demand nodes, U — access nodes, W — auxiliary nodes.
2. Set of edges E and the length $\mathrm{len}(e) \in \mathbb{R}$ for every $e \in E$.
3. $\mathrm{C}_{\mathrm{max}} \in \mathbb{N}$ — the maximal capacity of an FAT (in demands).
4. $\mathrm{L}_{\mathrm{max}} \in \mathbb{R}_+$ — the maximal distance from an FAT to a demand.

The problem formulation

Given

1. Set of nodes $V = D \cup U \cup W$,
   D — demand nodes, U — access nodes, W — auxiliary nodes.
2. Set of edges E and the length $\mathrm{len}(e) \in \mathbb{R}$ for every $e \in E$.
3. $\mathrm{C}_{\mathrm{max}} \in \mathbb{N}$ — the maximal capacity of an FAT (in demands).
4. $\mathrm{L}_{\mathrm{max}} \in \mathbb{R}_{+}$ — the maximal distance from an FAT to a demand.

Definitions: distances and degrees

$\mathrm{path}(v_1, v_2) \subseteq E$ — the shortest path between v_1 and v_2, $v_1, v_2 \in V$.
$\mathrm{len}(v_1, v_2) := \sum_{e \in \mathrm{path}(v_1, v_2)} \mathrm{len}(e)$ — the distance between v_1 and v_2.
$\mathrm{Paths} := \{(d, u) \mid d \in D,\ u \in U,\ \mathrm{len}(d, u) \leq \mathrm{L}_{\mathrm{max}}\}$ — set of feasible paths. $\mathrm{deg}_S(v) := \#\{s \in S \mid v \in s\}$ — the degree of a node v in a set S.

The problem formulation

Given

1. Set of nodes $V = D \cup U \cup W$,
   D — demand nodes, U — access nodes, W — auxiliary nodes.
2. Set of edges E and the length $\text{len}(e) \in \mathbb{R}$ for every $e \in E$.
3. $\text{C}_{\text{max}} \in \mathbb{N}$ — the maximal capacity of an FAT (in demands).
4. $\text{L}_{\text{max}} \in \mathbb{R}_+$ — the maximal distance from an FAT to a demand.

Definitions: distances and degrees

$\text{path}(v_1, v_2) \subseteq E$ — the shortest path between v_1 and v_2, $v_1, v_2 \in V$.
$\text{len}(v_1, v_2) := \sum_{e \in \text{path}(v_1, v_2)} \text{len}(e)$ — the distance between v_1 and v_2.
$\text{Paths} := \{(d, u) \mid d \in D,\ u \in U,\ \text{len}(d, u) \leq \text{L}_{\text{max}}\}$ — set of feasible paths. $\text{deg}_S(v) := \#\{s \in S \mid v \in s\}$ — the degree of a node v in a set S.

Find $B \subseteq \text{Paths}$, such that

1. $\text{deg}_B(d) = 1,\ \forall d \in D$.
2. $\sum_{u \in U} x(u)$ is minimal, where $x(u) := \lceil \text{deg}_B(u) / \text{C}_{\text{max}} \rceil$.

Properties of the data and special requirements

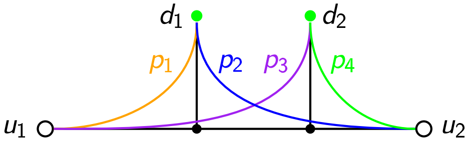
Properties

1. nodes ≈ 10000, demand nodes ≈ 5000, access nodes ≈ 2000.
2. nodes $<$ edges $< 1.2 \cdot$ nodes.
3. neighboring access nodes for a demand $\in [10, 30]$ (on average).
4. capacity $C_{max} \in \{8, 16, 32\}$.

Requirements

1. Clusters do not intersect (do not have common nodes).
2. Accuracy: no worse than 5% from optimum.
3. Do not use commercial software (e.g., CPLEX, Gurobi, etc.)
4. Time limit: 5 minutes.

Capacitated facility location problem

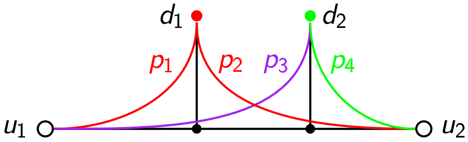


d_1, d_2 — demand nodes
u_1, u_2 — access nodes
p_1–p_4 — feasible paths

Variables:

$x(u) \in \mathbb{Z}_+$ is the number of FATs installed in $u \in U$;
$y(p) \in \{0, 1\}$ is the indicator of using path $p \in$ Paths.

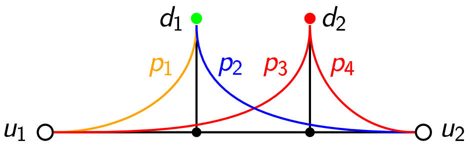
Capacitated facility location problem



d_1, d_2 — demand nodes
u_1, u_2 — access nodes
p_1–p_4 — feasible paths

Variables:
$x(u) \in \mathbb{Z}_+$ is the number of FATs installed in $u \in U$;
$y(p) \in \{0, 1\}$ is the indicator of using path $p \in$ Paths.
Constraints:

(1) $$y(p_1) + y(p_2) = 1,$$

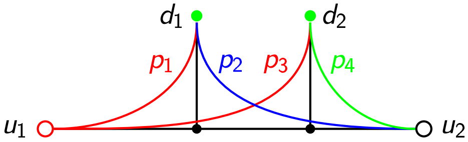


d_1, d_2 — demand nodes
u_1, u_2 — access nodes
p_1–p_4 — feasible paths

Variables:
$x(u) \in \mathbb{Z}_+$ is the number of FATs installed in $u \in U$;
$y(p) \in \{0, 1\}$ is the indicator of using path $p \in$ Paths.
Constraints:

$$y(p_1) + y(p_2) = 1, \quad (1)$$

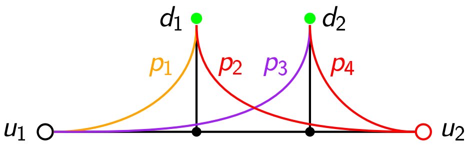
$$y(p_3) + y(p_4) = 1, \quad (2)$$



d_1, d_2 — demand nodes
u_1, u_2 — access nodes
p_1–p_4 — feasible paths

Variables:
$x(u) \in \mathbb{Z}_+$ is the number of FATs installed in $u \in U$;
$y(p) \in \{0, 1\}$ is the indicator of using path $p \in$ Paths.
Constraints:

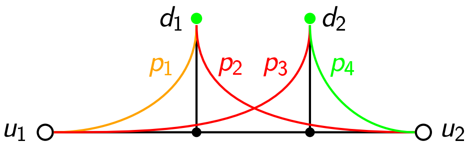
(1) $$y(p_1) + y(p_2) = 1,$$
(2) $$y(p_3) + y(p_4) = 1,$$
(3) $$y(p_1) + y(p_3) \leq \mathrm{C_{max}} \cdot x(u_1),$$



d_1, d_2 — demand nodes
u_1, u_2 — access nodes
p_1–p_4 — feasible paths

Variables:
$x(u) \in \mathbb{Z}_+$ is the number of FATs installed in $u \in U$;
$y(p) \in \{0, 1\}$ is the indicator of using path $p \in$ Paths.
Constraints:

(1) $$y(p_1) + y(p_2) = 1,$$
(2) $$y(p_3) + y(p_4) = 1,$$
(3) $$y(p_1) + y(p_3) \leq C_{max} \cdot x(u_1),$$
(4) $$y(p_2) + y(p_4) \leq C_{max} \cdot x(u_2),$$



d_1, d_2 — demand nodes
u_1, u_2 — access nodes
p_1–p_4 — feasible paths

Variables:
$x(u) \in \mathbb{Z}_+$ is the number of FATs installed in $u \in U$;
$y(p) \in \{0, 1\}$ is the indicator of using path $p \in$ Paths.
Constraints:

(1) $\quad y(p_1) + y(p_2) = 1,$

(2) $\quad y(p_3) + y(p_4) = 1,$

(3) $\quad y(p_1) + y(p_3) \leq \mathrm{C}_{\max} \cdot x(u_1),$

(4) $\quad y(p_2) + y(p_4) \leq \mathrm{C}_{\max} \cdot x(u_2),$

(5) $\quad y(p_2) + y(p_3) \leq 1.$

Capacitated Facility Location without intersections

D — demands, U — accesses, Paths — a set of feasible paths.
$x(u) \in \mathbb{Z}_+$ is the number of FATs installed in $u \in U$.
$y(p) \in \{0, 1\}$, for $p = (d, u) \in$ Paths.
N is the cost of FAT, $N > \mathrm{len}(p)$ for any $p \in$ Paths.

$$\text{minimize} \quad N \sum_{u \in U} x(u) + \sum_{(d,u) \in \text{Paths}} \mathrm{len}(d, u) y(d, u),$$

subject to

(1) $$\sum_{u \in U} y(d, u) = 1 \quad \forall d \in D,$$

(2) $$\sum_{d \in D} y(d, u) \leq x(u) \cdot \mathrm{C_{max}} \quad \forall u \in U,$$

(3) $$y(p_1) + y(p_2) \leq 1 \quad \forall \text{ intersected } \{p_1, p_2\} \in \text{Paths}.$$

Difficulties

The size of a problem

1. The number of 0/1-variables: $|\text{Paths}| \approx 10^5$.
2. The number of "intersection" inequalities $> 10^6$.

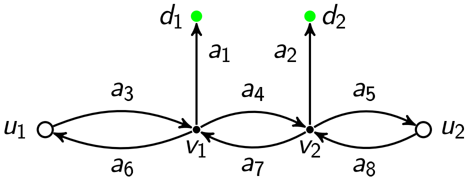
Requirements

1. Accuracy: no worse than 5% from optimum.
2. Do not use commercial software (e.g., CPLEX, Gurobi, etc.)
3. Time limit: 5 minutes.

Summary

The problem can't be solved by straightforward methods.

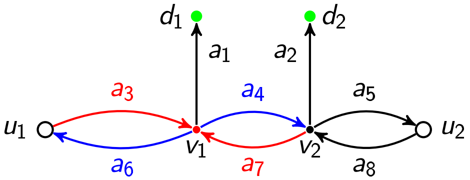
MILP model 2: flow in the source graph



d_1, d_2 — demand nodes
u_1, u_2 — access nodes
v_1, v_2 — auxiliary nodes
a_1–a_8 — arcs

Variables:

$x(u) \in \mathbb{Z}_+$ is the number of FATs installed in $u \in U$;
$y(a) \in \mathbb{R}_+$ is the value of a flow in arc $a \in A$.

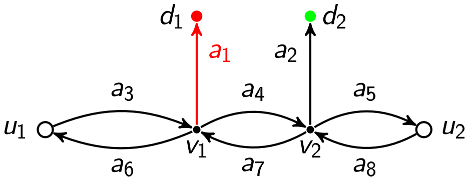


d_1, d_2 — demand nodes
u_1, u_2 — access nodes
v_1, v_2 — auxiliary nodes
a_1–a_8 — arcs

Variables:
$x(u) \in \mathbb{Z}_+$ is the number of FATs installed in $u \in U$;
$y(a) \in \mathbb{R}_+$ is the value of a flow in arc $a \in A$.
Constraints:

$$(1) \qquad \Delta(v_1) = (y(a_3) + y(a_7)) - (y(a_4) + y(a_6)) = 0,$$

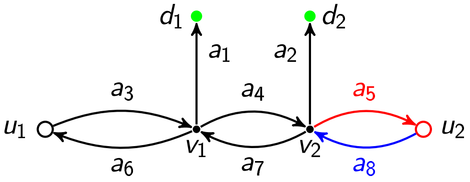
MILP model 2: flow in the source graph



d_1, d_2 — demand nodes
u_1, u_2 — access nodes
v_1, v_2 — auxiliary nodes
a_1–a_8 — arcs

Variables:
$x(u) \in \mathbb{Z}_+$ is the number of FATs installed in $u \in U$;
$y(a) \in \mathbb{R}_+$ is the value of a flow in arc $a \in A$.
Constraints:

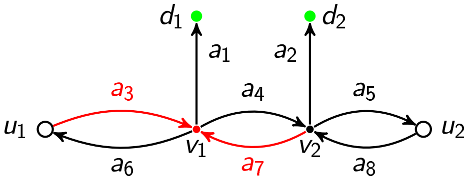
$$(1) \qquad \Delta(v_1) = (y(a_3) + y(a_7)) - (y(a_4) + y(a_6)) = 0,$$
$$(2) \qquad \Delta(d_1) = y(a_1) = 1,$$



d_1, d_2 — demand nodes
u_1, u_2 — access nodes
v_1, v_2 — auxiliary nodes
a_1–a_8 — arcs

Variables:
$x(u) \in \mathbb{Z}_+$ is the number of FATs installed in $u \in U$;
$y(a) \in \mathbb{R}_+$ is the value of a flow in arc $a \in A$.
Constraints:

(1) $$\Delta(v_1) = (y(a_3) + y(a_7)) - (y(a_4) + y(a_6)) = 0,$$
(2) $$\Delta(d_1) = y(a_1) = 1,$$
(3) $$-\Delta(u_2) = y(a_8) - y(a_5) \le x(u_2) \cdot \mathrm{C}_{\max},$$



d_1, d_2 — demand nodes
u_1, u_2 — access nodes
v_1, v_2 — auxiliary nodes
a_1–a_8 — arcs

Variables:
$x(u) \in \mathbb{Z}_+$ is the number of FATs installed in $u \in U$;
$y(a) \in \mathbb{R}_+$ is the value of a flow in arc $a \in A$.
Constraints:

$$\begin{aligned}
&(1) && \Delta(v_1) = (y(a_3) + y(a_7)) - (y(a_4) + y(a_6)) = 0, \\
&(2) && \Delta(d_1) = y(a_1) = 1, \\
&(3) && -\Delta(u_2) = y(a_8) - y(a_5) \le x(u_2) \cdot \mathrm{C}_{\max}, \\
&(4) && z(a_3) + z(a_7) \le 1,
\end{aligned}$$

$$z(a) \in \{0, 1\}, \quad \varepsilon \cdot y(a) \le z(a) \le y(a).$$

MILP model 2: flow in the source graph

$A := \{(v_1, v_2) \mid \{v_1, v_2\} \in E\}$ — set of arcs
$x(u) \in \mathbb{Z}_+$ is the number of FATs installed in $u \in U$.
$y(a) = y(v_1, v_2) \in \mathbb{R}_+$ is the amount of flow in arc $a = (v_1, v_2) \in A$.
$z(a) \in \{0, 1\}, \qquad \varepsilon \cdot y(a) \le z(a) \le y(a), \qquad a \in A.$
$\Delta(v) := \sum_{(s,v)\in A} y(s, v) - \sum_{(v,s)\in A} y(v, s)$ is the flow difference in v.

$$\text{minimize} \quad N \sum_{u\in U} x(u) + \sum_{a\in A} \text{len}(a) \cdot y(a),$$

subject to

(1) $\quad \Delta(w) = 0 \quad$ for every auxiliary node $w \in W$,

(2) $\quad \Delta(d) = 1 \quad$ for every demand $d \in D$,

(3) $\quad -\Delta(u) \le x(u) \cdot C_{\text{max}} \quad$ for every access node $u \in U$,

(4) $$\sum_{(v',v)\in A} z(v', v) + \varepsilon \cdot x(v) \le 1, \qquad \forall v \in V.$$

Pros and cons of the flow model

Pros: the numbers of variables and constraints are much smaller

	Flow model	Precise model
Variables	$\approx 30\,000$	$\approx 100\,000$
Constraints	$\approx 30\,000$	$> 1\,000\,000$

Cons: no distance constraints

The flow model has no a limit on the distance between demands and FATs. For adding this constraint, we may add variables-odometers to the model. But this will significantly increase the complexity of the model.

Results of experiments with the flow model

The size of model

The size: 26 000 variables, 26 000 constraints.

Half of variables and constraints are indicators:

$z(a) \in \{0, 1\}, \qquad \varepsilon \cdot y(a) \leq z(a) \leq y(a), \quad a \in A.$

Test results

1. CPLEX solves the flow model in 20 minutes with 11% gap.
2. Free solver CBC get 11% gap in 50 minutes.

Requirements

1. Accuracy: no worse than 5% from optimum.
2. Do not use commercial software (e.g., CPLEX, Gurobi, etc.)
3. Time limit: 5 minutes.

Idea 1: split the graph into several parts

Test results

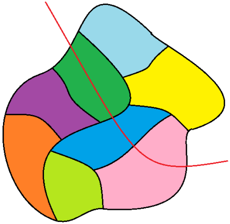
A subgraph of size 1000-2000 nodes (edges) is solved:

by CPLEX in 10-60 seconds;

by CBC in 10-60 minutes;

A disadvantage

The total solution quality deteriorates when we split the graph into several parts.
The smaller the part size,
the further a solution from the optimum.



Summary

The idea works, but the graph should be splitted carefully and the part size can't be small ($>$ 20 clusters in one part).

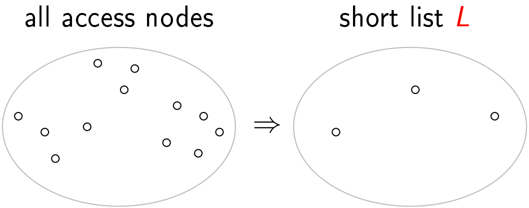
Idea 2: 2-stage approach

Prerequisite

The number of access nodes is much greater than the number of clusters in the solution.

2-stage approach

1. Choose a short list L of “good” access nodes (with a relaxation of the model or with some heuristic).
2. Run the precise model for L instead of the set of all access nodes.



Current results

Results

With using described and some other ideas, we get a good solution:

by CPLEX in 1 minute;

by CBC in 10 minutes;

To Do

- Tune CBC solver for this task.
- Develop an algorithm (heuristic) that does not use a MILP solver.
- Solve the task for a very large graph:
  100 000 demands, 300 000 accesses, cluster capacity $C_{max} = 1024$.

A similar task with a special requirement

In the previous task

Every demand is directly connected to a FAT.

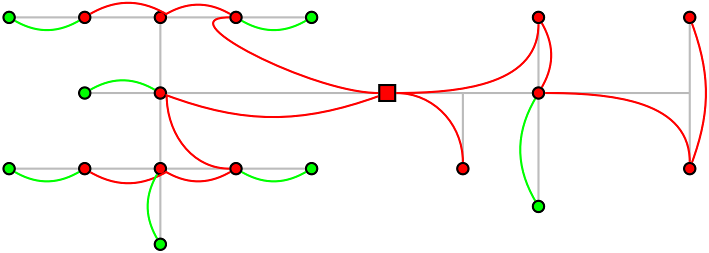
Thus, every cluster has a star-like structure.

In new task

Every cluster is a tree with at most 4 branches.

Every branch is a chain with at most 4 demand nodes

An additional demand may be connected to every node of a chain.



Thank You!