Optimal Location of Safety Landing Sites

Liding Xu
Claudia D'Ambrosio, Leo Liberti, Sonia Vanier

OptimiX, LIX, École polytechnique

liding.xu@polytechnique.edu

July, EURO 2021

Urban air mobility

- Urban air mobility (UAM) is driven by advancements in *battery, distributed electric propulsion, and autonomy technologies*.
- Electric vertical takeoff and landing (eVTOL) aircraft, are expected to be *safer, quieter, and less expensive to operate* than helicopters.
- We study the safety design of UAM networks: install safety landing sites (SLSs) for eVTOLs.

Outline

- Problem description.
- Mathematical models and formulations.
- Algorithms.
- Numerical experiments.
- Conclusion.

Background

- eVTOLs would exploit the vertical space i.e., to alleviate congestion on the ground.
- Safety is the primary consideration in network planning.

Space networks

- The 3d continuous sky is *discretized* into a 2d grid network.
- Vertiports are subset of nodes of the network.
- SLSs are located outside the grid, their covering areas are balls.
- **Demands** are transportation in eVTOLs among vertiports.
- **SLSs** allow eVTOLs to land in their covered ranges.

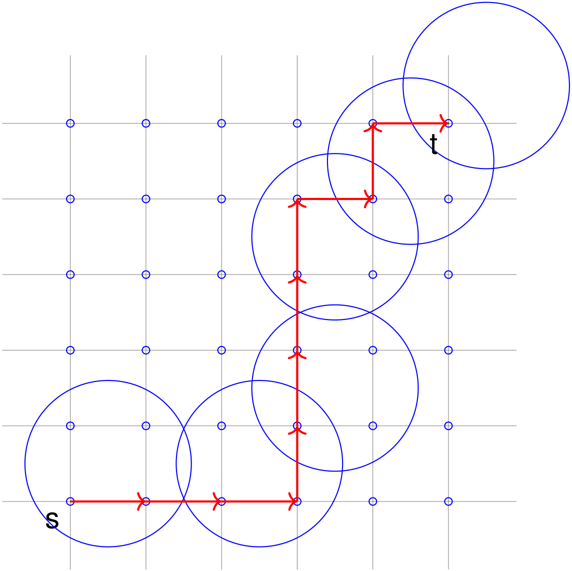

Figure: A path from vertiport s to vertiport t (in red)

Mathematical formulations

- Mathematical formulations are derived from multi-commodity flow (MCF) problem.
- Unsplittable MCF is known to be $\mathcal{NP}-$hard, but integer programming approach is efficient in practice.

Mathematical formulations

- Two representations, edge and path formulations.
- **Compact Edge formulation:** every node has flow conservation constraints and associated variables, and their size grows polynomially w.r.t. network size.
- **Path formulation:** edge variables are aggregated into path variables i.e. incident vectors. Exponential number of path variables.

SLS location problem

- **Decision variables** are the routing of eVTOLs and the selection of SLSs to install.
- **Objective:** the cost of eVTOL transportation.
- **Cover constraints:** every route is covered by SLSs.
- **Capacity constraints:** each edge can have a limited number of eVTOLs.
- **Budget constraints:** the number of installed SLSs is less than b.
- **Unsplittable constraints.**

Notations

- The network $G = (V, A, c, m)$ where V and A is the set of nodes and edges.
- c_{ij} : The cost of moving 1 eVTOL on edge, $(i, j) \in A$.
- m_{ij} : The capacity of eVTOLs on edge, $(i, j) \in A$.
- D demands, demand $h \in D$ requires transportation of a eVTOL from a source vertiport $s_h \in V$ to a destination vertiport $t_h \in V$.
- $\bar{\ell}$ is the number of available SLSs.
- A_ℓ is the set of edges covered by SLS $\ell \in \{1, \ldots, \bar{\ell}\}$.
- A_0 is the set of edges covered by all vertiports.

Edge formulation

$$\min_{x,y} \sum_{h\in D} \sum_{(i,j)\in A} c_{ij} x_{ij}^h$$

$$\sum_{(j,i)\in A} x_{ji}^h - \sum_{(i,j)\in A} x_{ij}^h = \begin{cases} -1 \text{ if } i = s^h \\ +1 \text{ if } i = t^h \\ 0 \text{ otherwise} \end{cases} \qquad \forall i \in N, h \in D$$

$$\sum_{h\in D} x_{ij}^h \le m_{ij} \qquad \forall (i,j) \in A,$$

$$x_{ij}^h \le \sum_{\ell=1,(i,j)\in A_\ell}^{\bar{\ell}} y_\ell \qquad \forall (i,j) \in A \setminus A_0, h \in D$$

$$\sum_{\ell=1}^{\bar{\ell}} y_\ell \le b$$

$$x_{ij} \in \{0,1\} \quad \forall (i,j) \in A, y_\ell \in \{0,1\} \quad \forall \ell = 1,\ldots,\bar{\ell}$$

Path formulation

$$\min \sum_{(i,j)\in A} c_{ij} \sum_{h=1}^{D} \sum_{p\in P^h,(i,j)\in p} x_p^h$$

$$\begin{array}{ll} \sum_{h=1}^{D} \sum_{p\in P^h,(i,j)\in p} x_p^h \leq m_{ij} & \forall (i,j) \in A, \\ \sum_{p\in P^h} x_p^h = 1 & \forall h = [1, D], \end{array}$$

Path formulation

$$\min \sum_{(i,j)\in A} c_{ij} \sum_{h=1}^{D} \sum_{p\in P^h,(i,j)\in p} x_p^h$$

$$\begin{array}{ll}
\sum_{h=1}^{D} \sum_{p\in P^h,(i,j)\in p} x_p^h \leq m_{ij} & \forall (i,j) \in A, \\
\sum_{p\in P^h} x_p^h = 1 & \forall h = [1, D], \\
\sum_{p\in P^h,(i,j)\in p} x_p^h \leq \sum_{\ell=1,(i,j)\in A_\ell}^{\ell} y_\ell & \forall h = [1, D], \forall (i,j) \in A \setminus A_0, \\
\sum_{\ell=1}^{\ell} y_\ell \leq b & \\
x_p^h \in \{0, 1\} & \forall h = [1, D], \forall p \in P^h, \\
y_\ell \in \{0, 1\} & \forall \ell = \{1, \ldots, \bar{\ell}\}
\end{array}$$

Its linear relaxation is solved by the column generation.

Algorithms

- Column generation is an efficient method for solving large scale linear programming.
- Column generation progressively solves restricted master problems (RMP).
- Efficient when column size is exponential to row size (LP relaxation of path formulation!).
- **Branch-and-bound:** implicitly enumerates solutions for combinatorical problems.
- **Branch-and-price:** embeds column generation into branch-and-bound.

Column generation

How does reduce cost pricing work?

Answer: Look at the dual problem of LP relaxation.

LP relaxation:

$$\min \sum_{(i,j)\in A} c_{ij} \sum_{h=1}^{D} \sum_{p\in P^h,(i,j)\in p} x_p^h$$

$$\begin{array}{ll}
\sum_{h=1}^{D} \sum_{p\in P^h,(i,j)\in p} x_p^h \leq m_{ij} & \forall (i,j) \in A, (\gamma_{ij} \geq 0) \\
\sum_{p\in P^h} x_p^h = 1 & \forall h = [1,D], (\mu_h \in R) \\
\sum_{p\in P^h:(i,j)\in p} x_p^h \leq \sum_{\ell=1:(i,j)\in A_\ell}^{\ell} y_\ell & \forall h = [1,D], \forall (i,j) \in A \setminus A_0, \\
 & (\eta_{ij}^h \geq 0) \\
\sum_{\ell=1}^{\ell} y_\ell \leq b & (\xi \geq 0) \\
x_p^h \in [0,\infty) & \forall h = [1,D], \forall p \in P^h, \\
y_\ell \in [0,\infty) & \forall \ell = \{1,\ldots,\bar{\ell}\}
\end{array}$$

Pricing problem

The dual is:

$$
\begin{array}{ll}
\max -\sum_{(i,j)\in A}\gamma_{ij}m_{ij} - \sum_{h=1}^{D}\mu_h - \xi\sum_{\ell=1}^{\ell} b & \\
\sum_{(i,j)\in p}(c_{ij}+\gamma_{ij}) + \mu_h + \sum_{(i,j)\in p:(i,j)\notin A_0}\eta_{ij}^h \geq 0, & \forall h=[1,D], \forall p \in P^h \\
-\sum_{h=1}^{D}\sum_{(i,j)\in A_\ell}\eta_{ij}^h + \xi \geq 0, & \forall \ell in\{1,\dots,\bar{\ell}\} \\
\gamma_{ij} \geq 0, & \forall (i,j)\in A \\
\mu_h \in R, & \forall h=[1,D] \\
\eta_{ij}^h \geq 0, & \forall h=[1,D], \\
 & \forall (i,j)\in A\setminus A_0 \\
\xi \geq 0. &
\end{array}
$$

- Reduced cost of $p \in P^h$:
  $RC(p) = \sum_{(i,j)\in p}(c_{ij}+\gamma_{ij}) + \mu_h + \sum_{(i,j)\in p:(i,j)\notin A_0}\eta_{ij}^h$.
- The column with the least reduced cost is found by a shortest path algorithm.

Numerical experiments: instances

Instance	Nodes	Edges	Demands	SLSs	vertiports
1	36	120	3	16	4
2	48	164	6	20	6
3	63	220	7	20	6
4	100	360	11	36	16
5	225	840	17	49	36
6	324	1224	20	64	49
7	400	1520	25	81	64
8	529	2024	25	100	81

Table: Instances

Numerical experiments: a visual example

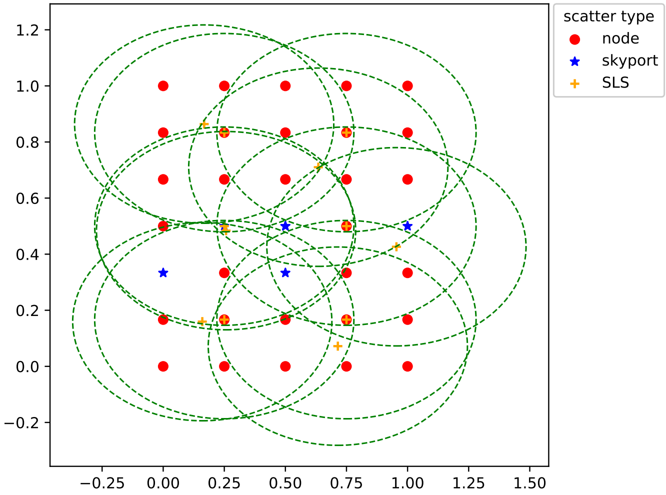

Figure: A template city

Numerical experiments: computational results

I	B	Edge formulation				Path formulation			
		$\bar{z}^*$	Gap(%)	t	Nodes	$\bar{z}^*$	Gap(%)	t	Nodes
1	5	175.98	0	0.02	1	175.98	0	0.53	4
2	5	355.92	0	0.05	1	355.92	0	0.2	23
3	5	591.19	0	4.74	1538	591.19	0	3600	128920
4	5	300.05	0	0.05	1	300.05	0	0.56	1
5	9	1512.13	0	22.83	1446	1512.18	0.31	3600	64666
6	20	2290.75	0	790.37	20861	-	-	3600	33192
7	25	3025.70	0.35	3600	30341	-	-	3600	10635
8	29	-	-	3600	20861	-	-	3600	10829

- Compact edge formulation is solved by Cplex 12.10.0 single thread mode.
- Path formulation is solved by Scip 7.0.1 with Cplex as a LP solver.
- Time limit is set to 3600 seconds.

Experiments: discussions

I	B	Edge formulation				Path formulation			
		$\overline{z}^*$	Gap(%)	t	Nodes	$\overline{z}^*$	Gap(%)	t	Nodes
1	5	175.98	0	0.02	1	175.98	0	0.53	4
2	5	355.92	0	0.05	1	355.92	0	0.2	23
3	5	591.19	0	4.74	1538	591.19	0	3600	128920
4	5	300.05	0	0.05	1	300.05	0	0.56	1
5	9	1512.13	0	22.83	1446	1512.18	0.31	3600	64666
6	20	2290.75	0	790.37	20861	-	-	3600	33192
7	25	3025.70	0.35	3600	30341	-	-	3600	10635
8	29	-	-	3600	20861	-	-	3600	10829

- Cplex indeed separates cuts to strengthen the edge formulation.
- The scip's branch and price deactives cut separation.
- SLS variable y makes the network design problem harder than the routing problem.

Conclusion

Summary

- We propose an model for SLS location problem.
- We propose 2 formulations for the model.
- We devise algorithms to solve 2 formulations.

Future work

- Combine with Bender decomposition.
- Improve the stability of column generation.
- Valid inequalities.