An exact algorithm for robust flying-sidekick Traveling Salesman Problem

*Youngjoo Roh, Jaegwan Joo, and Chungmok Lee**

Hankuk University of Foreign Studies, Department of Industrial and Management Engineering

*chungmok@hufs.ac.kr

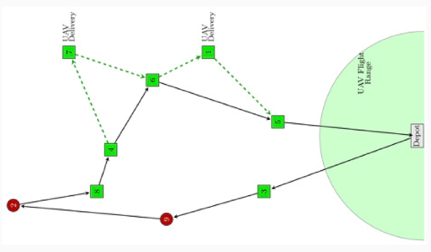
Hankuk University of Foreign Studies

1. Background

Drone delivery problems

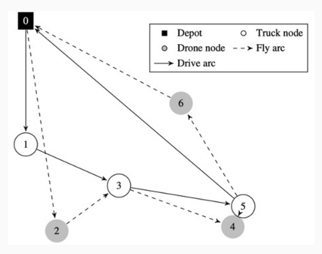
Flying sidekick TSP (FS-TSP)

Murray, Chase C, Amanda G Chu. 2015



TSP with Drone (TSP-D)

Agatz, Niels, Paul Bouman, Marie Schmidt. 2018



Previous Studies on the FSTSP and the TSP-D

Many papers focus on developing solution methods:

1. Simulated annealing heuristic (Ponza 2016)
2. Variable neighborhood search (Freitas and Penna 2020)
3. Dynamic programming (Bouman et al. 2018)
4. Branch-and-bound (Poikonen et al. 2019)
5. Branch-and-price (Roberti and Ruthmair 2019)

- **All of above did not consider uncertainties.**

Previous Studies on FSTSP w/ weather uncertainties

Unlike trucks, drones' operation depend on **weather conditions**

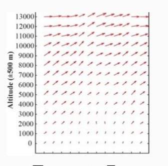
1. Drone routing problem considering the impact of wind on energy consumption (Radzki et al. 2019)
2. Divide the time period into several flying time slots, and the weather condition in each slot is known and analogous. (Thibbotuwawa et al. 2019)
3. Robustly optimize the schedules of the drone delivery systems to mitigate the risks of delivery delays due to uncertainty in wind conditions. (Chun Cheng et al. 2020)

- **All of above assume weather information is precisely given**

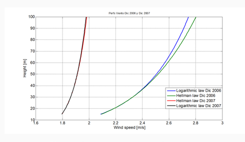
'Data' is still uncertain

The wind direction and speed **vary** with altitudes (often unpredictable)

- **Wind direction and speed on surface might be useless**



Turgut, E , Usanmaz, Ö. 2016

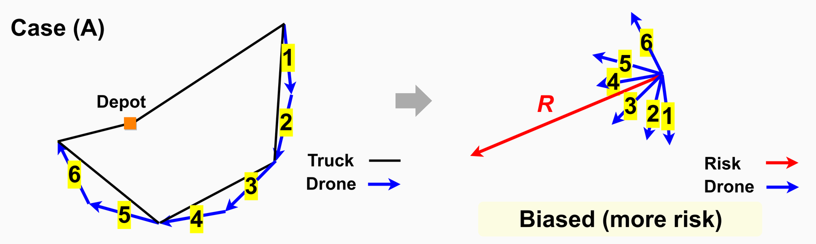


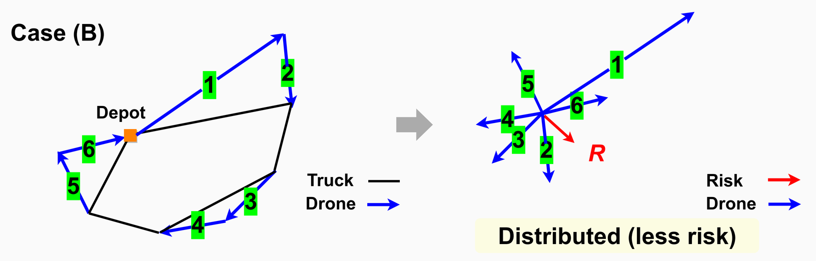
Ban˜uelos-Ruedas, F. et al. 2011

2. Idea

Robustness of the solutions

A collection of drone arcs would determine the overall robustness of the solutions





- **Case (B) is better because the directions and lengths of the arcs are more evenly distributed**

3. Define risk measure

Directional Statistics perspective

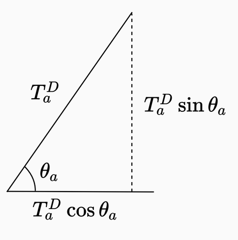
<u>Main idea:</u> **Directional statistics** is a branch of statistics dealing with observations that are directions

- Data of this type typically arise in:
  - meteorology (wind directions)
  - astronomy (directions of cosmic rays or stars)
  - ...

Reference: Christophe Ley, Thomas Verdebout - Modern Directional Statistics-Chapman and Hall/CRC (2017)

Consider a set of drone arcs A^D

- T^D_a : Flight time of the drone for arc $a \in A^D$
- θ_a : angle of arc $a \in A^D$ (polar coordinate)
- $X := \frac{1}{|A^D|}\sum_{a\in A^D} T^D_a \cos\theta_a$
- $Y := \frac{1}{|A^D|}\sum_{a\in A^D} T^D_a \sin\theta_a$
- $R := \sqrt{X^2+Y^2}$: Risk measure

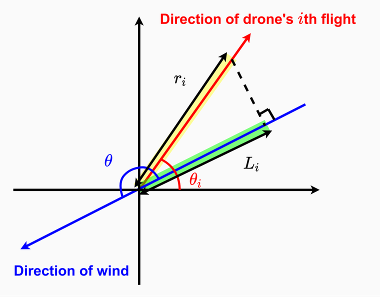


Resultant length R is a measure of angular dispersion, similar to the standard deviation

Robust optimizational perspective

<u>Main idea:</u> For a given wind direction θ

$$L = \sum_{i\in N} L_i = \sum_{i\in N} r_i \cos(\theta + \pi - \theta_i)$$



* L represents the average of drone arc vectors on the line for wind direction

- We consider a **directionally robust risk measure** :

$$\begin{aligned} R &= \max_{\theta\in[0,2\pi]} L \\ &= \max_{\theta\in[0,2\pi]} \sum_{i\in N} r_i \cos(\theta+\pi-\theta_i) \\ &= \sqrt{X^2+Y^2} \end{aligned}$$

⇒ Two perspectives agree on R can be a risk measure

Coherency

A 'good' risk measure might need to be **Coherent risk measure**

Examples of Coherent Risk Measures:

- Conditional VaR
- Tail VaR(with continuous distribution)
- Superhedging price
- ...

Axioms of Coherent Risk Measures:

- $P(L+a) = P(L) + a$: Translation invariance
- $P(\alpha L) = \alpha P(L)$: Positive homogeneity
- $P(L_1 + L_2) \le P(L_1) + P(L_2)$: Subadditivity
- $L_1 \le L_2, \forall\theta \in [0,\pi] \rightarrow P(L_1) \le P(L_2)$: Monotinicity

4. Problem Definition

Problem Definition

Data

- Depot
- Set of customers

Objective

- Minimize total travel time + risk measure

Constraints

- Vehicle must visit all customers
- Vehicle must start and end at depot
- Drone is launched from / collected by vehicle

Assumptions

- Truck is no faster than the drone
- Negligible service and drone charging times
- Drone can serve only one customer per operation
- All customer nodes can be visited by either the truck or the drone
- Both vehicles can wait while en route

Sets & Parameters

- s : $\{0\}$, which is the start node
- t : $\{n+1\}$, which is the end node
- N : $\{1,\dots,n\}$, Set of customers
- N_s : $\{0,1,\dots,n\}$, Set of customers
- N_t : $\{1,\dots,n,n+1\}$, Set of customers
- A : $\{(i,j) | i \in N_s, j \in N_t$ and $i \ne j$ and $(i,j) \ne (0,n+1)\}$, Set of arcs
- B : $\{(i,j,k) | i \in N_s, j \in N, k \in N_t$ and $i \ne j \ne k$ and $(i,k) \ne (0,n+1)\}$, Set of possible drone arcs
- T_{ij} : Travel time from i to j, where $(i,j) \in A$
- T^D_{ij} : Drone operation time from i to j, where $(i,j) \in A$
- T^D_{ijk} : Drone operation time from i to j to k, where $(i,j,k) \in B$
- C : Drone battery capacity
- θ_{ij} : Angle between $(i,j) \in A$ and x-axis
- α : Trade-off between total time and risk
- S : node subset
- $\delta^+(S) := \{(i,j) \in A \mid i \in S, j \notin S\}$

Decision Variables

- x_{ij} : 1 if the truck trave ls from i to j, and 0 otherwise
- y_{ijk} : 1 if the drone travels from i to j to k, and 0 otherwise
- t_i : earliest time for leaving i, where $i \in N_{st}$
- X : x-axis risk measure
- Y : y-axis risk measure
- R : risk measure

Mathematical Formulation (RFS-TSP)

$$\text{(P)} \quad \min \sum_{i\in N_t} t_i + \alpha R \qquad (1) \quad \text{Total travel time} \quad \text{Risk measure}$$

$$\text{s.t.} \quad \sum_{j\in N} x_{sj} = 1, \quad \sum_{i\in N} x_{it} = 1 \qquad (2)$$

$$\sum_{j\in N_t, i\ne j} x_{ij} = \sum_{j\in N_s, i\ne j} x_{ji}, \quad \forall i \in N \qquad (3)$$

$$\sum_{(i,j,k)\in B} y_{ijk} \le x_{ik}, \quad \forall (i,k) \in A \qquad (4)$$

$$\sum_{(i,j,k)\in B} T^D_{ijk} y_{ijk} \le C \qquad (5)$$

$$\sum_{(i,j)\in\delta^+(S)} x_{ij} \ge \sum_{(i,j)\in\delta^+(\{k\})} x_{ij}, \quad \forall k \in S, S \subseteq N_s, |S| \ge 2 \qquad (6) \quad \text{Sub-tour elimination}$$

$$\sum_{i\in N_s} T_{ij} x_{ij} \le t_j \quad \forall j \in N_t, \quad \sum_{i\in N_s} T^D_{ijk} y_{ijk} \le t_k, \quad \forall k \in N_t \qquad (7) \quad \text{Define arrival time of each node}$$

$$2\left(\sum_{(i,j,k)\in B} y_{ijk}\right) X = \sum_{(i,j,k)\in B} \left(T^D_{ij}\cos\theta_{ij} + T^D_{jk}\cos\theta_{jk}\right) y_{ijk} \qquad (8) \quad \text{Define x-axis risk measure}$$

$$2\left(\sum_{(i,j,k)\in B} y_{ijk}\right) Y = \sum_{(i,j,k)\in B} \left(T^D_{ij}\sin\theta_{ij} + T^D_{jk}\sin\theta_{jk}\right) y_{ijk} \qquad (9) \quad \text{Define y-axis risk measure}$$

$$R = \sqrt{X^2+Y^2} \qquad (10) \quad \text{Define Risk measure}$$

Constraints (7), (8), (9) make the problem non-linear

5. Linearization

Difficulties in the mathematical model (P)

1. Non-linear MIP
2. Too many constraints

To overcome the difficulties, we define (LBP^u_l) which has the properties below:

- Linear MIP
- Can provide the lower bound of **(P)**
- Delayed cut generation

Define linearized model (LBP^u_l)

1. Define decision variable p as the number of nodes served by drone. Decision variable p can be expressed as:

$$p = \sum_{(i,j,k)\in B} y_{ijk}$$

2. We define range of p with l and u so it can provide the **lower bound** of the MIP

$$l \le p \le u$$

3. We replace R with the norm equality $|X| + |Y| \le \sqrt{2}\sqrt{X^2+Y^2}$

(P) $R = \sqrt{X^2+Y^2}$ (LBP^u_l) $R = \frac{1}{\sqrt{2}}(X+Y)$

4. Using the range of p, linearize the constraint to provide lower bound of X

(P) $p = \sum_{(i,j,k)\in B} y_{ijk}$, $2pX = \sum_{(i,j,k)\in B}\left(T^D_{ij}\cos\theta_{ij} + T^D_{jk}\cos\theta_{jk}\right) y_{ijk}$

(LBP^u_l) $2uX \ge \sum_{(i,j,k)\in B}\left(T^D_{ij}\cos\theta_{ij} + T^D_{jk}\cos\theta_{jk}\right) y_{ijk}$, $2uX \ge -\sum_{(i,j,k)\in B}\left(T^D_{ij}\cos\theta_{ij} + T^D_{jk}\cos\theta_{jk}\right) y_{ijk}$

5. Using the range of p, linearize the constraint to provide lower bound of Y

(P) $p = \sum_{(i,j,k)\in B} y_{ijk}$, $2pY = \sum_{(i,j,k)\in B}\left(T^D_{ij}\sin\theta_{ij} + T^D_{jk}\sin\theta_{jk}\right) y_{ijk}$

(LBP^u_l) $2uY \ge \sum_{(i,j,k)\in B}\left(T^D_{ij}\sin\theta_{ij} + T^D_{jk}\sin\theta_{jk}\right) y_{ijk}$, $2uY \ge -\sum_{(i,j,k)\in B}\left(T^D_{ij}\sin\theta_{ij} + T^D_{jk}\sin\theta_{jk}\right) y_{ijk}$

Linearized model

Non-linear MIP (P)

$$\min \sum_{i\in N_t} t_i + \alpha R \qquad (1)$$

$$\text{s.t.} \quad (2)\cdots(7)$$

$$2\left(\sum_{(i,j,k)\in B} y_{ijk}\right) X = \sum_{(i,j,k)\in B}\left(T^D_{ij}\cos\theta_{ij} + T^D_{jk}\cos\theta_{jk}\right) y_{ijk} \qquad (8)$$

$$2\left(\sum_{(i,j,k)\in B} y_{ijk}\right) Y = \sum_{(i,j,k)\in B}\left(T^D_{ij}\sin\theta_{ij} + T^D_{jk}\sin\theta_{jk}\right) y_{ijk} \qquad (9)$$

$$R = \sqrt{X^2+Y^2} \qquad (10)$$

(LBP^u_l)

$$\min \sum_{i\in N_t} t_i + \alpha R \qquad (1)$$

$$\text{s.t.} \quad (2)\cdots(7)$$

$$2uX \ge \sum_{(i,j,k)\in B}\left(T^D_{ij}\cos\theta_{ij} + T^D_{jk}\cos\theta_{jk}\right) y_{ijk} \qquad (11)$$

$$2uX \ge -\sum_{(i,j,k)\in B}\left(T^D_{ij}\cos\theta_{ij} + T^D_{jk}\cos\theta_{jk}\right) y_{ijk} \qquad (12)$$

$$2uY \ge \sum_{(i,j,k)\in B}\left(T^D_{ij}\sin\theta_{ij} + T^D_{jk}\sin\theta_{jk}\right) y_{ijk} \qquad (13)$$

$$2uY \ge -\sum_{(i,j,k)\in B}\left(T^D_{ij}\sin\theta_{ij} + T^D_{jk}\sin\theta_{jk}\right) y_{ijk} \qquad (14)$$

$$l \le \sum_{(i,j,k)\in B} y_{ijk} \le u \qquad (15)$$

$$R = \frac{1}{\sqrt{2}}(X+Y) \qquad (16)$$

Observations

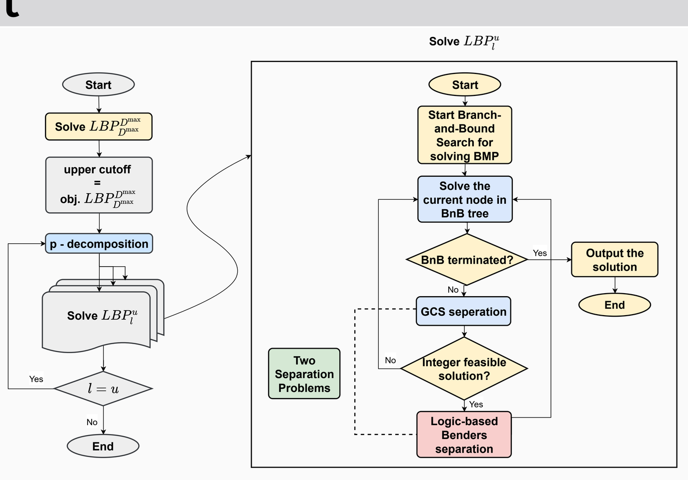
1. $Z_P \ge Z^*$
2. $Z_P \ge \underline{Z}_P \ge \underline{Z}^u_l$ if $u \ge P$ and $l \le P$

where

- Z^*: optimal value of **(P)**
- $\underline{Z}_P$: optimal value of **((P)** with $\sum_{(i,j,k)\in B} y_{ijk} = P$
- $\underline{Z}^u_l$: optimal value of **(LBP^u_l)**

6. Proposed Algorithms

Flowchart



Logic-based Benders SP for (LBP^u_l)

The Benders master problem is solved once, and **the Benders subproblem is solved to generate the Benders cuts whenever an integer feasible solution is found.**

If $\widehat{R} < \sqrt{\widehat{X}^2+\widehat{Y}^2}$, then solution $(\hat{y})$ produces an optimality cut defined as:

$$R \ge -\sqrt{\widehat{X}^2+\widehat{Y}^2}\left[\sum_{(i,j,k)\in B:\hat{y}_{ijk}=1}(1-y_{ijk}) + \sum_{(i,j,k)\in B:\hat{y}_{ijk}=0} y_{ijk}\right] + \sqrt{\widehat{X}^2+\widehat{Y}^2} \qquad (17)$$

- When $(y) = (\hat{y})$, then (17) reduces to $R \ge \sqrt{\widehat{X}^2+\widehat{Y}^2}$.
- In any other case, $\sum_{(i,j,k)\in B:\hat{y}_{ijk}=1}(1-y_{ijk}) + \sum_{(i,j,k)\in B:\hat{y}_{ijk}=0} y_{ijk} = m \ge 1$ and (17) reduces to $R \ge (1-m)\cdot\sqrt{\widehat{X}^2+\widehat{Y}^2}$

Sub-tour Elimination Constraints

The Benders master problem lacks the sub-tour elimination constraints A stronger linear relaxation bound can be obtained by using exponentially many constraints such as **generalized cut-set inequalities**

GCS, Taccari 2016

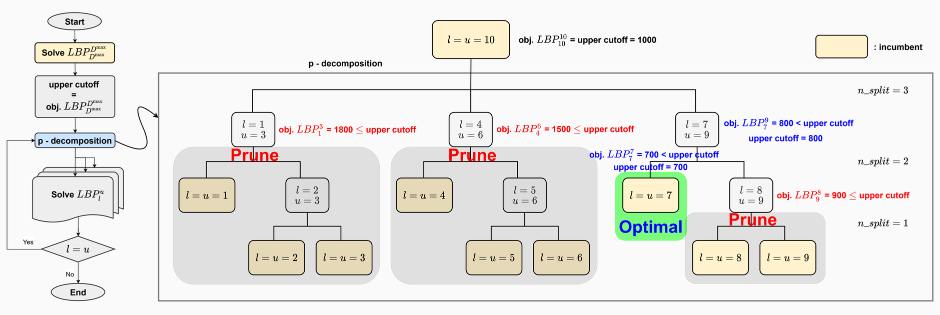
Algorithm 1 Separation of GCS for sub-tour elimination

```
1:  x* ← solution of the (BMP) at the current BnB node
2:  ε ← 0.8
3:  Construct graph G(N_st, A*), where A* := {(i,j) ∈ A | x*_ij > 0 or x*_ji > 0}
4:  𝒮 ← {S ⊆ N_s | S is a strongly connected component on G}     ▷ Depth-first search on G(N_st, A*)
5:  C ← ∅
6:  for S ∈ 𝒮 do
7:      for k ∈ S do
8:          v ← Σ_{(i,j)∈δ+({k})} x*_ij − Σ_{(i,j)∈δ+(S)} x*_ij
9:          if v ≥ ε then
10:             C ← C ∪ {(v, S, k)}
11:         end if
12:     end for
13: end for
14: return C
```

p - decomposition

Firstly, we define D^{max} satisfying battery capacity.

Then we **decompose p as l, u with the parameter** *n_split*.



p - decomposition is a branch-and-bound with LBP^u_l subproblem

7. Numerical Studies

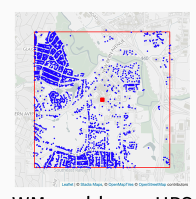
Test instances

Recently, the U.S. Federal Aviation Administration (FAA) issued certifications to some selected companies for drone delivery.

We selected the following two areas for **real-world** use cases:

- An area near CVS Pharmacy store in Cary, N.C., U.S.
- An area near WakeMed hospital campus in Raleigh, N.C., U.S.

CVS problems, UPS (2019a)



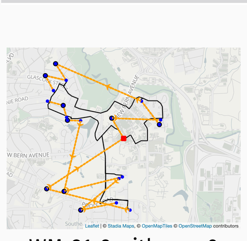
WM problems, UPS (2019b)

* Note that these are mostly residential areas without too many tall buildings.
* We assume that the drone is allowed to fly everywhere in the considered areas (i.e., no no-fly zones).

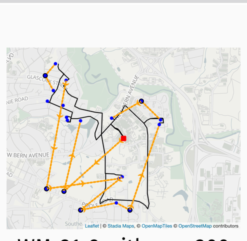
Experiments Environment

- Linux machine equipped with Intel i9-9900KS 5GHz CPU and 64GB RAM
- The algorithm was implemented by Python 3.8
- CPLEX 20.10 was used for solving the mathematical formulations

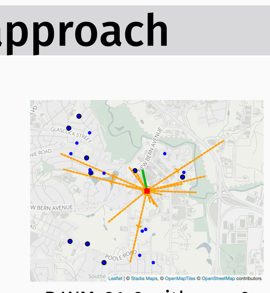
Deterministic vs Robust approach



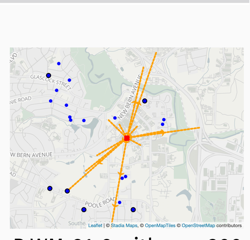
WM-21-9 with $\alpha = 0$



WM-21-9 with $\alpha = 300$



R WM-21-9 with $\alpha = 0$



R WM-21-9 with $\alpha = 300$

Computational Results: α=100

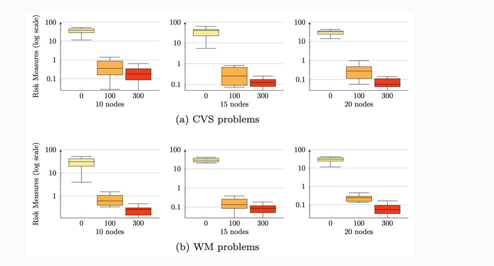
$\|N\|$	Problem	$\|D\|$	arrival	R	obj	time (sec)
11	CVS-11-1	5	1517	0.410	1534.66	107.49
11	CVS-11-2	5	1413	0.836	1454.04	84.16
11	CVS-11-3	4	1512	1.015	1595.61	57.03
11	CVS-11-4	5	1411	0.179	1512.50	79.73
11	CVS-11-5	5	1667	0.028	1684.92	91.34
16	CVS-16-1	8	1809	0.073	1816.32	1319.15
16	CVS-16-2	8	1589	0.086	1597.60	954.61
16	CVS-16-3	8	1716	0.096	1725.62	573.23
16	CVS-16-4	8	1372	0.253	1397.29	80.97
16	CVS-16-5	6	1740	0.628	1802.83	185.23
21	CVS-21-1	10	1985	0.078	1992.82	7897.51
21	CVS-21-2	9	1672	0.275	1699.47	714.27
21	CVS-21-3	10	1743	0.970	1839.95	1260.45
21	CVS-21-4	10	1851	0.611	1912.11	1361.54
21	CVS-21-5	10	1957	0.498	2006.77	2756.13

CVS problems computational results

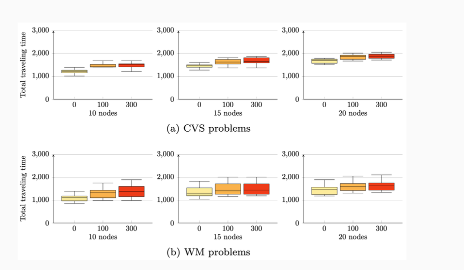
$\|N\|$	Problem	$\|D\|$	arrival	R	obj	time (sec)
11	WM-11-1	4	1452	0.614	1513.42	20.30
11	WM-11-2	4	1219	1.520	1371.00	19.70
11	WM-11-3	3	1414	1.380	1551.99	20.88
11	WM-11-4	4	985	0.465	1031.51	46.12
11	WM-11-5	5	1351	0.902	1441.22	201.86
16	WM-16-1	6	1759	0.183	1777.25	161.48
16	WM-16-2	6	1294	0.379	1331.87	5709.35
16	WM-16-3	6	2012	0.027	2014.68	412.72
16	WM-16-4	6	1162	0.322	1194.17	86.35
16	WM-16-5	7	1409	0.207	1429.65	1515.43
21	WM-21-1	9	1624	0.172	1641.16	3102.29
21	WM-21-2	9	1308	0.198	1327.83	2490.37
21	WM-21-3	9	1437	0.133	1450.28	2387.66
21	WM-21-4	10	1757	0.258	1782.83	3384.47
21	WM-21-5	8	1409	0.258	1434.84	18624.76

WM problems computational results

Correlation between risk measure and tradeoff value (α)



- The risk measure R decreases significantly as the tradeoff value (α) increases



- The total arrival time increases slightly as the tradeoff value (α) increases

8. Conclusion and Discussion

Contributions

- **Robust FSTSP** is introduced
- **An exact algorithm** for RFSTSP with nonlinear risk measure
- Present a nonlinear integer formulation
- Develop a **decomposition approach**
- Computational experiments with **real-life instances**

Possible extensions

- Time windows can be involved
- Multiple drones can be involved
- Robust algorithm incorporating given weather information