Multi-Robot Waypoint Inspection Planning with Mixed Integer Linear Programming Project

Juan Carlos Cruz - ira406

ME 6033 Linear and Mixed Integer Optimization

May 2025

Introduction

- Based on previous work done in an outdoor concrete inspection multirobot framework human interface
- Precast concrete elements require efficient inspection methods after being transported to a site
- Multi-robot approach:
  - Aerial robots locate targets
  - Ground robots perform detailed inspections
- Wanted to extend idea to robot dispatching problem

Problem Description

- Two robot types with variable quantity: aerial and ground mobile robots
- Inspection targets as waypoints in a 2D plane (x, y)
- Depot location for each robot type. Must leave and return to this location.
- Sequential operation:
  - Aerial robots verify waypoint location first
  - Ground robots perform detailed inspection second
- Robot parameters (defined by robot type):
  - Fixed speeds (meters/minute)
  - Limited operation time (battery life time after leaving depot)
  - Required inspection time at waypoints

Literature Review

- Prior works demonstrate multi-robot planning applications
- Problem resembles multiple traveling salesman problem (mTSP)
- Traditional MILP for mTSP:
  - Routing variables
  - Subtour elimination constraints
- Initially tried this approach but solving time was too slow and larger problems became infeasible (likely implementation error) so used a simplification
- Route approximation: roundtrip distances from depot to waypoints estimate travel times

Mathematical Model - Overview

- Mixed Integer Linear Programming (MILP) formulation
- Objective: Maximize number of waypoints inspected
- Constraints consider:
  - Robot assignment to waypoints
  - Sequential operations (aerial robots first, then ground)
  - Operating time limitations from battery endurance
  - Travel speed and inspection time requirements
- Approximates routes using roundtrip distances from depot

Mathematical Model - Sets and Indices

- N: Set of all waypoints indexed by $i \in \{1, 2, \ldots, n\}$
- K: Set of aerial robots indexed by $k \in \{1, 2, \ldots, k_{\text{max}}\}$
- L: Set of ground robots indexed by $l \in \{1, 2, \ldots, l_{\text{max}}\}$
- d_A: Aerial robot depot
- d_G: Ground robot depot

Mathematical Model - Parameters

- p_i: Location of waypoint $i \in N$
- p_{d_A}: Location of aerial robot depot
- p_{d_G}: Location of ground robot depot
- $\mathrm{dist}(i, j)$: Euclidean distance between locations i and j
- s_A: Speed of aerial robots (m/min)
- s_G: Speed of ground robots (m/min)
- t_A^{insp}: Inspection time for aerial robots (min)
- t_G^{insp}: Inspection time for ground robots (min)
- $T_A^{\max}$: Maximum operation time for aerial robots (min)
- $T_G^{\max}$: Maximum operation time for ground robots (min)

Mathematical Model - Derived Parameters

- $t_{ij}^{A} = \frac{\text{dist}(i,j)}{s_A}$: Travel time for aerial robots from i to j (min)
- $t_{ij}^{G} = \frac{\text{dist}(i,j)}{s_G}$: Travel time for ground robots from i to j (min)
- M_A: Big-M value for aerial robot time constraints
  Calculated as $T_A^{\max} + \max(t_{ij}^{A}) + t_A^{\text{insp}}$
- M_G: Big-M value for ground robot time constraints
  Calculated as $T_G^{\max} + \max(t_{ij}^{G}) + t_G^{\text{insp}}$

Mathematical Model - Decision Variables

- $w_i^{a,k}$: Binary variable equals 1 if aerial robot k visits waypoint i
- $w_i^{g,l}$: Binary variable equals 1 if ground robot l visits waypoint i
- a_i^k: Time when aerial robot k completes inspection at waypoint i
- g_i^l: Time when ground robot l completes inspection at waypoint i
- use_k^a: Binary variable equals 1 if aerial robot k is used
- use_l^g: Binary variable equals 1 if ground robot l is used
- $z_i^{k,l}$: Binary variable equals 1 if ground robot l visits waypoint i after aerial robot k

Mathematical Model - Objective Function

- Goal: Maximize the number of waypoints visited by ground robots

$$\text{Maximize} \sum_{i \in N} \sum_{l \in L} w_i^{g,l} \tag{1}$$

- A waypoint is only considered completely inspected when a ground robot has visited it
- Aerial robot visits alone do not contribute to the objective

Mathematical Model - Assignment Constraints

- Each waypoint can be assigned to at most one aerial robot:

$$\sum_{k \in K} w_i^{a,k} \leq 1 \quad \forall i \in N \tag{2}$$

- Each waypoint can be assigned to at most one ground robot:

$$\sum_{l \in L} w_i^{g,l} \leq 1 \quad \forall i \in N \tag{3}$$

Mathematical Model - Robot Usage Constraints

- A robot is used if it visits at least one waypoint:

$$\sum_{i \in N} w_i^{a,k} \geq \text{use}_k^a \quad \forall k \in K \tag{4}$$

$$\sum_{i \in N} w_i^{g,l} \geq \text{use}_l^g \quad \forall l \in L \tag{5}$$

- A robot is used only if it visits at least one waypoint:

$$\sum_{i \in N} w_i^{a,k} \leq n \cdot \text{use}_k^a \quad \forall k \in K \tag{6}$$

$$\sum_{i \in N} w_i^{g,l} \leq n \cdot \text{use}_l^g \quad \forall l \in L \tag{7}$$

Mathematical Model - Precedence Constraints (1)

- Ground robots can only visit waypoints already visited by aerial robots:

$$w_i^{g,l} \leq \sum_{k \in K} w_i^{a,k} \quad \forall i \in N, \forall l \in L \tag{8}$$

- Ground robot's inspection must occur after aerial robot's inspection:

$$g_i^l \geq a_i^k - M_G \cdot (1 - z_i^{k,l}) - M_G \cdot (2 - \mathrm{use}_k^a - \mathrm{use}_l^g) \quad \forall i, k, l \tag{9}$$

Mathematical Model - Precedence Constraints (2)

Constraints on $z_i^{k,l}$ (linking variable for timing precedence):

$$z_i^{k,l} \leq w_i^{g,l} \quad \forall i \in N, \forall k \in K, \forall l \in L \tag{10}$$

$$z_i^{k,l} \leq w_i^{a,k} \quad \forall i \in N, \forall k \in K, \forall l \in L \tag{11}$$

$$z_i^{k,l} \leq \text{use}_k^a \quad \forall i \in N, \forall k \in K, \forall l \in L \tag{12}$$

$$z_i^{k,l} \leq \text{use}_l^g \quad \forall i \in N, \forall k \in K, \forall l \in L \tag{13}$$

$$z_i^{k,l} \geq w_i^{g,l} + w_i^{a,k} + \text{use}_k^a + \text{use}_l^g - 3 \quad \forall i, k, l \tag{14}$$

Mathematical Model - Time Constraints (1)

- Minimum inspection time at waypoints:

$$a_i^k \geq t_A^{\text{insp}} \cdot w_i^{a,k} \quad \forall i \in N, \forall k \in K \tag{15}$$

$$g_i^l \geq t_G^{\text{insp}} \cdot w_i^{g,l} \quad \forall i \in N, \forall l \in L \tag{16}$$

Mathematical Model - Time Constraints (2)

- Maximum operation time constraints:

$$a_i^k + t_{i,d_A}^A \cdot w_i^{a,k} \leq T_A^{\max} + M_A \cdot (1 - w_i^{a,k}) \tag{17}$$

$$\forall i \in N, \forall k \in K \tag{18}$$

$$g_i^l + t_{i,d_G}^G \cdot w_i^{g,l} \leq T_G^{\max} + M_G \cdot (1 - w_i^{g,l}) \tag{19}$$

$$\forall i \in N, \forall l \in L \tag{20}$$

The University of Texas at San Antonio

Mathematical Model - Route Length Constraints

- Total mission time estimation (route approximation):

$$\sum_{i\in N} w_i^{a,k} \cdot \left(2 \cdot t_{d_A,i}^{A}\right) + \sum_{i\in N} w_i^{a,k} \cdot t_A^{\mathrm{insp}} \tag{21}$$

$$\leq T_A^{\max} + M_A \cdot (1 - \mathrm{use}_k^a) \quad \forall k \in K \tag{22}$$

$$\sum_{i\in N} w_i^{g,l} \cdot \left(2 \cdot t_{d_G,i}^{G}\right) + \sum_{i\in N} w_i^{g,l} \cdot t_G^{\mathrm{insp}} \tag{23}$$

$$\leq T_G^{\max} + M_G \cdot (1 - \mathrm{use}_l^g) \quad \forall l \in L \tag{24}$$

Mathematical Model - Domain Constraints

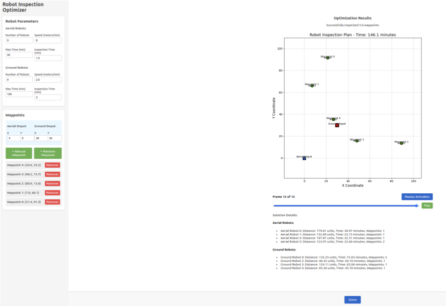
- Domain:

$$w_i^{a,k}, w_i^{g,l}, \mathrm{use}_k^a, \mathrm{use}_l^g, z_i^{k,l} \in \{0,1\} \qquad \forall i \in N, k \in K, l \in L \tag{21}$$

$$a_i^k, g_i^l \geq 0 \qquad \forall i \in N, k \in K, l \in L \tag{22}$$

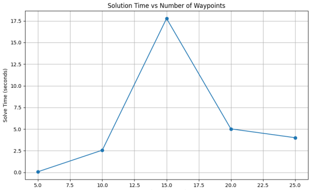
Solution Method

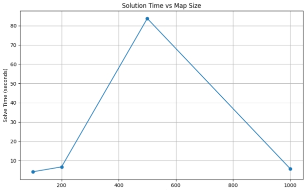
- Implemented in Python using PuLP library
- CBC solver from PuLP used for optimization
- Interactive browser-based GUI developed:
  - Parameter input for robot specifications
  - Waypoint location setting
  - Real-time solution visualization
- Code available at GitHub repository



Numerical Results - Computational Performance

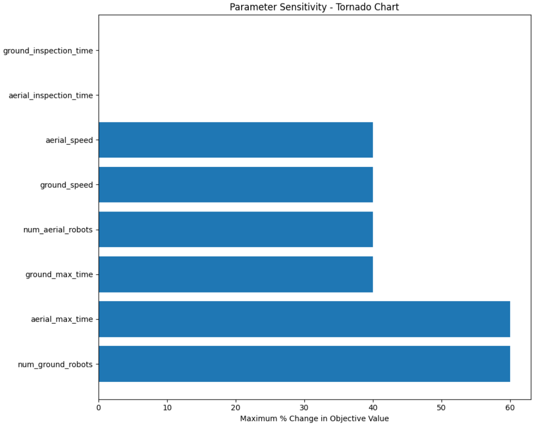
- Testing environment:
  - Intel i7, Python 3.12 Docker container
- Tested waypoint scaling (5 to 25 waypoints in 1000 m map) and map size scaling (100 to 1000 m with 20 random waypoints)
- Non-monotonic scaling behavior:
  - Solution time peaks at 15 waypoints then decreases
  - Computation time peaks at 500m map size



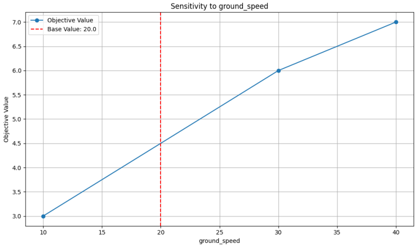


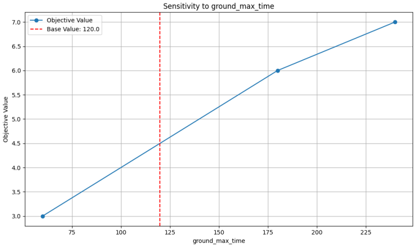
Numerical Results - Sensitivity Analysis

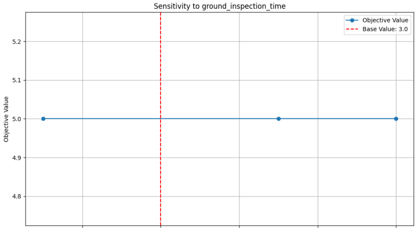
- Using fixed waypoints and map size we solve multiple times varying 8 params: robot speeds, operation times, inspection times, and fleet sizes.
- Most influential parameters:
  - Number of ground robots
  - Aerial robot maximum operation time
- Minimal impact: Inspection times

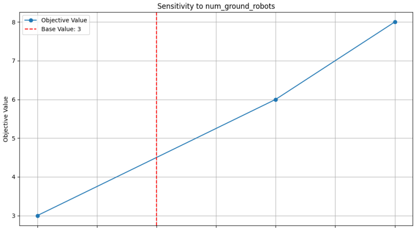


Numerical Results - Ground Robot Sensitivity Analysis

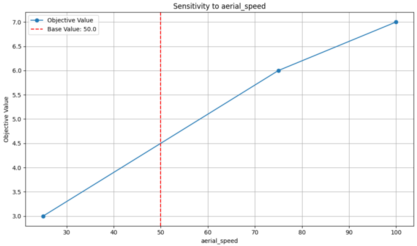


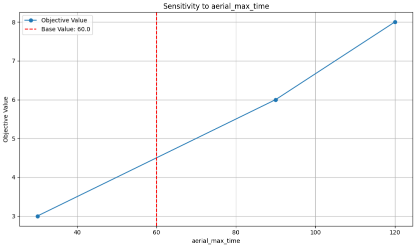


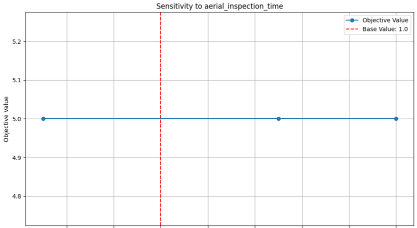


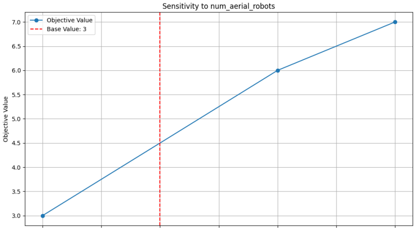


Numerical Results - Aerial Robot Sensitivity Analysis

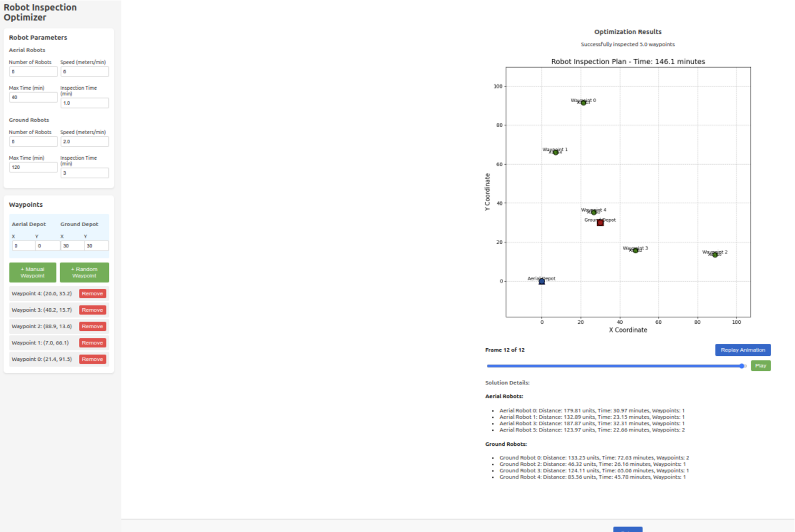








Demo Video



Robot Inspection Optimizer
Robot Parameters
Aerial Robots
Number of Robots
6
Speed (meters/min)
6
Max Time (min)
40
Inspection Time (min)
1.0
Ground Robots
Number of Robots
6
Speed (meters/min)
2.0
Max Time (min)
120
Inspection Time (min)
3
Waypoints
Aerial Depot
Ground Depot
X
Y
0
0
30
30
+ Manual Waypoint
+ Random Waypoint
Waypoint 4: (26.6, 35.2)
Remove
Waypoint 3: (48.2, 15.7)
Remove
Waypoint 2: (88.9, 13.6)
Remove
Waypoint 1: (7.0, 66.1)
Remove
Waypoint 0: (21.4, 91.5)
Remove
Optimization Results
Successfully inspected 5.0 waypoints
Robot Inspection Plan - Time: 146.1 minutes
Waypoint 0
Waypoint 1
Waypoint 4
Ground Depot
Waypoint 3
Waypoint 2
Aerial Depot
Y Coordinate
X Coordinate
0
20
40
60
80
100
Frame 12 of 12
Replay Animation
Play
Solution Details:
Aerial Robots:
Aerial Robot 0: Distance: 179.81 units, Time: 30.97 minutes, Waypoints: 1
Aerial Robot 1: Distance: 132.89 units, Time: 23.15 minutes, Waypoints: 1
Aerial Robot 3: Distance: 187.87 units, Time: 32.31 minutes, Waypoints: 1
Aerial Robot 5: Distance: 123.97 units, Time: 22.66 minutes, Waypoints: 2
Ground Robots:
Ground Robot 0: Distance: 133.25 units, Time: 72.63 minutes, Waypoints: 2
Ground Robot 2: Distance: 46.32 units, Time: 26.16 minutes, Waypoints: 1
Ground Robot 3: Distance: 124.11 units, Time: 65.06 minutes, Waypoints: 1
Ground Robot 4: Distance: 85.56 units, Time: 45.78 minutes, Waypoints: 1