# Mixed-Integer Linear Programming Models for Multi-Robot Non-Adversarial Search

SC635 Course Project

Siddharth Saha(170100025) Gyandev Gupta(190100051)

(Under the guidance of Prof. Arpita Sinha)

April 2021

# Objective of the Project

## Motivation & Aim of the Project

- The aim of the project is to solve a Multi-Robot Efficient Search Path Planning (MESPP) problem introduced by G. Hollinger using Mixed Integer Linear Programming (MILP)
- Task- a team of robots is deployed in an environment represented as an undirected graph with the aim of capturing a moving non-adversarial target within a given deadline
- The problem is important because we need to catch it up in the given deadline it would have been given as to ensure that the target must always be located within a certain time in practical situations.
- Our models are the first to encompass multiple searchers, arbitrary capture ranges, and false negatives simultaneously
- The models are designed for computing optimal solutions and simulations were performed on it

# Terminology to get started

- **Non-adversarial** - It means that the target doesn't try to foil the capture team's plans and sticks to its pre-defined course of actions
- **MILP** - Mixed Integer Linear Programming (MILP) is an optimization problem where only some variables are constrained to be integers, while other variables can be non-integers. It deals with linear objective function subjected to linear constraints.
- **False Negatives** - Target was present within sensing range but the capture robot missed it
- **Arbitrary Capture Ranges**- How far the capture robot's sensing range can reach. Analogously, how close the capture team robots need to be to the moving target to declare the target as captured
- **Same Vertex Capture** - It means that the searcher and object must be on the same vertex to declare that target has been captured

# Related Work

- Target search kind of problems have been in study under research and game theory communities
- Researchers like T.Chung, G.Hollinger and many others provide an overview of how target search problems can be tackled from a robotic perspective
- Receding horizon method and sequential allocation MESPP
- A rigorous taxonomy with several classification dimensions. For example, the target can be static or dynamic ; adversarial , non-adversarial ; in case of known target's motion model, this can be a random walk or Markovian ;
- Environment - continuous, unbounded or bounded, or discrete and represented by a finite graph.

# Way to Proceed

- This problem version deals with a dynamic, non-adversarial target which moves in a graph-represented environment according to a known Markovian motion model
- Initially we prove that the MESPP problem is NP-hard even for two-dimensional grid environments with a static target and a single searcher
- We present the first set of Mixed-Integer Linear Programming (MILP) models, most general one for tackling the MESPP problem
- The proposed MILP models for the MESPP problem allow path enumeration to be performed in a much more efficient way by leveraging the sophisticated branching and pruning techniques of modern MILP solvers, yielding better computational performance and, as a consequence, longer planning horizons
- We tackle the MESPP problem with a receding horizon approach that can be implemented either in a centralized or in a distributed fashion.

# Setting up the problem

- Let G = (V, E) be an undirected, connected, and simple graph representing a known environment, with V = 1, 2, . . ., n
- Here we discretized the environment in a grids or constrained Delaunay triangulation
- $\delta(v)$ to denote the neighbors of v $\in$ V , while $\delta'(v)$ represents $\delta(v) \cup$ v. Let d(u, v) be the length of the shortest path between any two vertices u, v $\in$ V
- The object moves probabilistically in the graph, with motion encoded by a Markov chain specified by the stochastic matrix $\mathbf{M} \in \mathbb{R}^{n \times n}$ where $\mathbf{M}_{uv}$ represents the probability that the object will move from u to v between t and t + 1.

# NP-Hard

- Hardness of the MESPP problem by proving that its decision version, which we dub MESPP-D, is NP-hard
  **HAMILTONIAN PATH BETWEEN POINTS (2HP)**
- *Theorem* : MESPP-D is NP-hard even when the following conditions hold simultaneously:
  1) G is a grid graph,
  2) the target is stationary, and
  3) there is only one searcher with perfect sensing capabilities.

# Optimization Problem

- There are m searchers and $P$ denote the set of all possible joint path by all searchers
- $\mathbf{b}^{\pi}(t) = [b_c(t), b_1(t), ..., b_n(t)]$ represents the belief vector where $b_c(t)$, represents the probability that the searchers have located the object by time t
- $\pi^*$ it optimizes the path taken by all the searchers

$$\pi^* = \text{arg } max_{\pi \in P} \sum_{t=0}^{\tau} \gamma^t b_c(t)$$

where $\gamma \in (0, 1]$ is a discount factor

## Legal Path for Searchers

- $x_v^{s,t} = 1$ denotes presence of searcher **s** in vertex **v** at time t
- $y_{uv}^{s,t} = 1$ denotes the searcher **s** will move from **u** to **v** in interval [t,t+1]
- $\delta(v)$ to denote the neighbors of v $\in$ V and $\delta'(v)$ denotes $\delta(v) \cup \{v\}$

$$x_{v_0^s}^{s,0} = 1 = \sum_{j \in \delta'(v_0^s)} y_{v_0^s j}^{s,0} = \sum_{j \in V^{s,t}(\tau)} y_{jv_g}^{s,\tau}, \forall s \in S$$

$$x_v^{s,t} = \sum_{j \in \delta'(v) \cap V^{s,t}(t-1)} y_{jv}^{s,t-1} = \sum_{i \in \delta'(v)} y_{vi}^{s,t}, \forall (v, t < \tau) \in V^{s,t}$$

$$x_v^{s,\tau} = \sum_{j \in \delta'(v) \cap V^{s,t}(\tau-1)} y_{jv}^{s,\tau-1} = y_{vv_g}^{s,\tau}, \forall (v, t < \tau) \in V^{s,t}(\tau)$$

# Objects Motion

- $\beta_i^t$ - entry of the belief vectors at time t

$$\beta_i^t \in [0,1], \forall i \in V \cup \{c\}, t \in \{0\} \cup T$$

- $\alpha_v^t$ - result of the application of the motion model

$$\alpha_v^t \in [0,1], \forall v \in V, t \in T$$

$$\beta_i^0 = b_i(0), \forall i \in V \cup \{c\}$$
$$\alpha_v^t = \sum_{u \in V} \mathbf{M}_{uv} \beta_u^{t-1}, \forall v \in V, t \in T$$

# Capture Events in Same Vertex, Binary Detection

- $\psi_v^t$ a binary variable = 1 $\iff$ $\exists$ atleast one searcher located in v at t

$$\beta_v^t = \alpha_v^t(1 - \psi_v^t), \forall t \in T, v \in V$$

- The model is non linear so we linearize it with some valid assumptions they are

$$\beta_v^t \leq 1 - \psi_v^t$$

$$\beta_v^t \leq \alpha_v^t$$

$$\beta_v^t \geq \alpha_v^T - \psi_v^t$$

$$\psi_v^t \in \{0, 1\}$$

# Capture Events in Same Vertex, Binary Detection

- If all searchers are in vertex v at time t, $\psi_v^t$ = 1 then equality holds

$$\sum_{s \in S \; s.t \; v \in V^{s,t}(t)} x_v^{s,t} \leq m\psi_v^t, \forall v \in V, t \in T$$

- If however no searcher is in v equality holds

$$\psi_v^t \leq \sum_{s \in S \; s.t \; v \in V^{s,t}(t)} x_v^{s,t}, \forall v \in V, t \in T$$

- Probability of object getting caught of

$$\beta_c^t = 1 - \sum_{v \in V} \beta_v^t, \forall t \in T$$

# Capture Events Within Given Range, Binary Detection

- The searcher is located at u , to detect the object within the range of vertex v

$$\sum_{s \in S} \sum_{u \in V^{s,t}(t)\ s.t\ \mathbf{C}^{s,u}_{v0}=1} x^{s,t}_{v} \leq m\psi^{t}_{v}, \forall v \in V, t \in T$$

$$\psi^{t}_{v} \leq \sum_{s \in S} \sum_{u \in V^{s,t}(t)\ s.t\ \mathbf{C}^{s,u}_{v0}=1} x^{s,t}_{u}, \forall v \in V, t \in T$$

- Probability of object getting caught of

$$\beta^{t}_{c} = 1 - \sum_{v \in V} \beta^{t}_{v}, \forall t \in T$$

## Capture Events With False Negatives

- We need to account for the chance that the object is actually in that vertex, but has not been detected. We now modify the belief equation as

$$\beta_v^{s,t} = (1 - \zeta^s)\beta_v^{s-1,t}(1 - \psi_v^t) + \zeta^s \beta_v^{s-1,t}, \forall s \in S, t \in T, v \in V$$

$$\text{where } \beta_v^{0,t} = \alpha_v^t, \forall t \in T, v \in V$$

$$\beta_v^{s,t} \in [0,1], \forall t \in T, v \in V, s \in S$$

$$\psi_v^{s,t} \in \{0,1\}, \forall t \in T, v \in V, s \in S$$

- Define auxillary variable $\delta_v^{s,t} = \beta_v^{s-1,t}(1 - \psi_v^{s,t}) \in [0,1], \forall t \in T, v \in V, s \in S$

$$\delta_v^{s,t} \leq 1 - \psi_v^{s,t}$$

$$\delta_v^{s,t} \leq \beta_v^{s-1,t}$$

$$\delta_v^t \geq \beta_v^{s-1,t} - \psi_v^t$$

# Capture Events With False Negatives

- After taking into account above assumptions to linearize,

$$\beta_v^{s,t} = (1-\zeta^s)\beta_v^{s-1,t}(1-\psi_v^t) + \zeta^s \beta_v^{s-1,t}, \forall s \in S, t \in T, v \in V$$

$$\sum_{u \in V^{s,t}(t)\ s.t\ \mathbf{C}_{v0}^{s,u} > 0} x_v^{s,t} \leq \psi_v^t, \forall v \in V, t \in T$$

$$\psi_v^t \leq \sum_{s \in S} \sum_{u \in V^{s,t}(t)\ s.t\ \mathbf{C}_{v0}^{s,u} > 0} x_u^{s,t}, \forall v \in V, t \in T$$

- Probability that the object has been captured by time t

$$\beta_v^t = \beta_v^{m,t}, \forall t \in T$$

# Complete MILP Model

- It is optimized on the basis of constraints defined in the previous slides for respective case
- For same-vertex (SV) capture , no false negatives (FN):

$$\text{(SV-MILP) max } \sum_{t \in T} \gamma^t b_c(t)$$

- For arbitrary capture range, no false negatives:

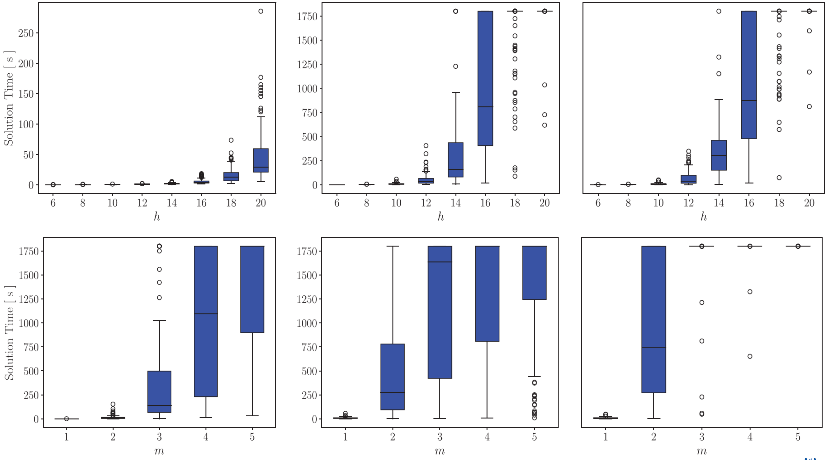
$$\text{(MV-MILP) max } \sum_{t \in T} \gamma^t b_c(t)$$

- Finally, the most general model, encompassing arbitrary capture ranges and false negatives:

$$\text{(FN -MV-MILP) max } \sum_{t \in T} \gamma^t b_c(t)$$

- Author's simulations show that it is possible to achieve 98% decrease in computational time relative to the previous state-of-the-art.


# Graphs



## Conclusion

- Leveraging the powerful techniques and tools used by modern solvers comes with a minor challenge: very rarely (three instances in total), the presolver might deal poorly with small probabilities and deem the problem infeasible
- This numerical issue is fixed either by turning the presolver off and increasing the solver timeout, or by keeping the searchers' in their current positions and re-planning on next time-step
- In author's simulations, the trade-off between expected mission time and required computational time is a challenging choice
- Future work will continue investigating MILP as a planning paradigm, towards the generalization of the presented models to handle heterogeneous teams of searchers (humans, ground and aerial vehicles), and the incorporation of connectivity constraints

# References


1. B. A. Asfora, J. Banfi and M. Campbell, "Mixed-Integer Linear Programming Models for Multi-Robot Non-Adversarial Search," in IEEE Robotics and Automation Letters, vol. 5, no. 4, pp. 6805-6812, Oct. 2020, doi: 10.1109/LRA.2020.3017473.