An Approach for the Determination of Drone Positions Set with Maximal Terrain Visibility

Andrii Dashkevych[1], Darya Vorontsova[1] and Sergii Rosokha[2]

[1]*National Technical University "Kharkiv Polytechnic Institute", Kharkiv 61002, Ukraine*

[2]*Private Institute for the Scientific Research of Fire Safety Ltd, Kharkiv 61017, Ukraine*

Abstract

In recent years, unmanned aerial vehicles (UAVs) have been found in many practical applications, one of the challenging problems is the estimation of the optimal path of the UAV to cover the target area. The visual inspection and monitoring with UAVs is the example of such applications and they have been attracted increasing attention in recent years. In our paper we present a novel approach to approximately estimate a set of viewpoints to reach maximal visibility of the polygonal model of the terrain in linear time. The proposed approach is based on a subdivision of a terrain model on a regular grid and on using ray casting technique to solve a terrain cover problem.

Keywords

unmanned aerial vehicles, visual inspection and monitoring, set of viewpoints, maximal visibility of the terrain, polygonal model, terrain cover problem

1. Introduction and Related Work

In recent years, unmanned aerial vehicles (UAVs or drones) have been found in many practical applications: farm monitoring, civil infrastructure inspection, military and emergence services. Drones are being actively introduced and are already being put into practice in emergency response services. The value of their use is primarily in saving time and resources. One of the challenging problems is the estimation of the optimal path of the UAV to cover the target area. The visual inspection with UAVs is the example of such application and it have attracted increasing attention in recent years. Since the 1980s researchers have proposed different approximation methods to find the solution [1, 2]. As a result of their work, an approach such as the "generate-test" approach became popular. Later, researchers proposed an "optimal scan zone" method [3]. More recently, a randomized sampling-based method was proposed [4]. The building inspection and surveillance applications have been studied by many researchers [5, 6]. It should be noted that an Unmanned Ground Vehicle (UGV) is also used to monitor the buildings with the viewpoint positions in 2D and 3D environment [5, 6]. But the methods used have their drawbacks, limitations and they are still not effective enough. Many scientists are trying to find the best solution. For example in the paper [7] had presented a novel two-step computational

method for finding near-optimal views to cover the surface of a target set of buildings using voxel dilation, medial objects, and random-key genetic algorithm. In [8] authors had devised a novel computation framework which consists of three components, i.e., optimized line-of-sight algorithm, R*-tree filter and MapReduce-based segmented computation. The proposed solution can support GIS systems to conduct efficient visibility analysis for massive observers. Today scientists present new algorithms and data structures that improve the previous results. In the work [9] authors have been studying the problem of computing the visibility polygon of any query point. As a special case of visibility problems, they also have been studying the ray-shooting problem of finding the first point on the polygon boundaries that is hit by any query ray. In others articles motivated by the inaccuracy of GPS devices in urban regions authors have been studying the problem of computing the visibility graph of an urban region. Given a scene of buildings, where a building is represented by the set of its walls, the vertices of the graph correspond to the buildings' walls, and there is an edge between two walls if and only if they are weakly visible to each other [10]. Also the algorithm for planning the self-correcting path based on visibility Graph method modification had been developed. The paper [11] presented the concept of self-correcting navigation based on planning the robot path in a known environment along walls of passed object so that it could be subsequently executed with only local control and without tracking the global position of the robot. Researchers are even considering the above issue using the visibility analysis in rural areas Bumiaji, Batu, where the characteristics of the rural mountains feared to be changed, threat natural environment or rural area, and then disrupt rural community activities [Integrating visibility analysis in rural spatial planning]. Visibility analysis using the view shed analysis tools in ArcGIS demonstrate the opportunities to combine methods of visual and spatial analysis that were used to obtain the ecological and aesthetic quality scores.

There are many other approaches considered in literature about UAV path planning such as [12, 13, 14, 15]. One part of them propose algorithm of optimal flight route in real world environment with no collision with environmental elements [16]. Some of them represent the aforementioned path or motion planning methods focus on obstacle avoidance issues [17]. There are articles, which address a path planning problem involving a small UAV with fuel constraints [18, 19, 20, 21, 22]. Also there is papers about maintained altitude constraints as well as the shape of the UAV formation [23]. In any cases it is important to find the shortest path for manoeuvres of UAV, since the power consumed during manoeuvres is tightly coupled with the length of the flight path. One of the way to solve this problem is to use evolutionary modeling. Evolutionary modeling is a field of research of artificial intelligence that combines methods of modeling evolutionary processes in the natural and artificial environment. It is divided into the following types: genetic algorithm; evolutionary and genetic programming; evolutionary strategies; differential evolution; particle swarm methods; ant algorithms and more. A genetic algorithm (GA) is based problem solving technique which treats possible solutions as biological entities and it produce new solutions from the best of the previous solution with some operator [24]. GA supports multi-purpose optimization, and it can be designed to work in parallel [25]. It uses various operators and selection methods to generate next-generation solutions.

2. Method Overview

2.1. Problem Statement

The main goal of our paper is to find a method to solve a modification of the set covering problem (SCP) in 3-dimensional space, which can be formulated as follows: given a polygonal model P of terrain and a set S of all possible viewpoints of the camera, we need to find a minimum subset $s \in S$ such that all viewpoints from s completely cover all the polygonal faces $F \in P$. The model $P = \{V, N, F\}$ is represented as the sets of vertices V, normals N and faces F of input polygons. The SCP is known as NP-complete problem thus it does not have a polynomial time solution and we need to use some heuristics to get an approximated solution. In our paper we present a novel approach to approximately estimate a subset s in linear time, which is based on a subdivision of a terrain model on a regular grid.

2.2. Candidate Viewpoints Generation

The first step is the generation of a full candidate viewpoint set S. At this stage we place a set of points which are uniformly cover a discrete unit hemisphere around of the center of each of the polygons $f_i \in F, i = 1..N$, where N – is a number of polygons in F (Fig.1). The orientation of a hemisphere is defined by a normal n_i of face f_i. Example of a polygonal model with normals and a corresponding generated set S of potentially visible points is provided in the Fig.2.

2.3. Spatial Hashing of Viewpoints

At the second step we approximate set S using the spatial hashing technique. In our approach we split S into regular grid with the rectangular cells of size c. The cell size is defined as follows:

$$c_x = \frac{R_x}{t}, \tag{1}$$

$$c_y = \frac{R_y}{t}, \tag{2}$$

where t is a subdivision parameter, which is equal for all of axis and:

$$R_x = x_{max} - x_{min}, \tag{3}$$

$$R_y = y_{max} - y_{min}, \tag{4}$$

are space dimensions in x, y direction.

Then we calculate the indices of each point $s \in S$:

$$i = \frac{x_s}{c_x}, \tag{5}$$

$$j = \frac{y_s}{c_y}, \tag{6}$$

Thus we can consider each cell as a point $p = (i, j)$ in 2-dimensional space. Then we can build spatial hash-table $H = \langle K, V \rangle$, where $K = hash(i, j)$ – is a list of spatial hashes of cells and V – is a list of indices of points that belong to the cell with hash H. The hashing function is described as follows:
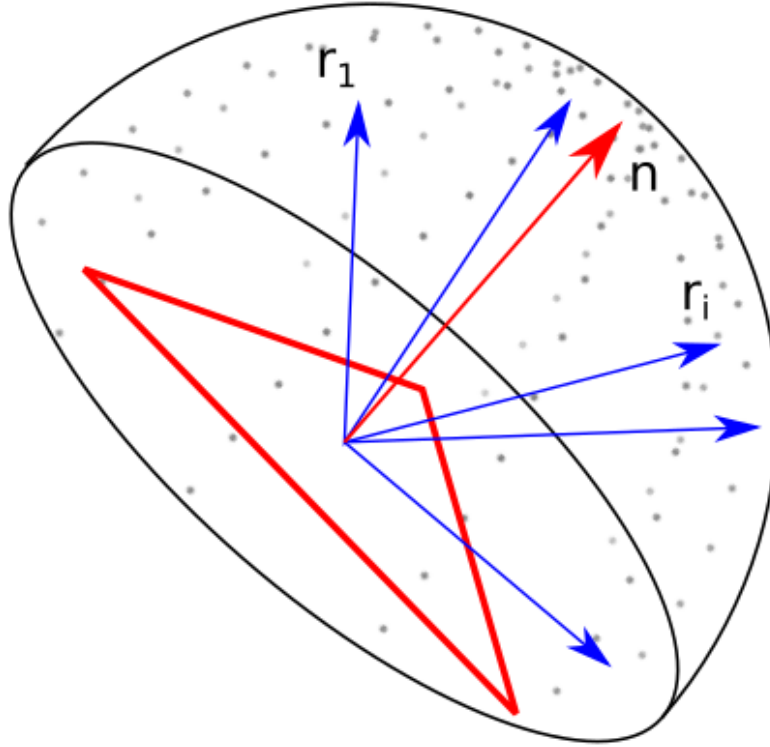
Figure 1: Discrete unit hemisphere.

- For each point p we take a concatenation of string representations of its components $K = str(i) + str(j)$, where $str(\cdot)$ is a function that converts integer value to its string representation.

After spatial hash-table is constructed, we can associate each cell with a weight – the number of triangles that are visible from the cell and, therefore, we can easily connect each region of terrain with a viewpoints. To provide fast mapping between the candidate viewpoint and the corresponding polygon we build a helper data structure A, which contains lists of indices of input polygons $f_i \in F$ and indices of each of the lists are pairwise corresponding to hash values in K.

2.4. Visibility Planning

In our approach we use a greedy algorithm to solve a problem of polygon covering, which can be formulated as follows:

$$\min \sum_{i=1}^{N} x_i, x_i \in \{0, 1\} \tag{7}$$

where $x_i = 1$ means that i-th polygon is visible and $x_i = 0$ means it is not visible from the cell viewpoint. As such viewpoint we choose a highest viewpoint in the current cell of the hash table.
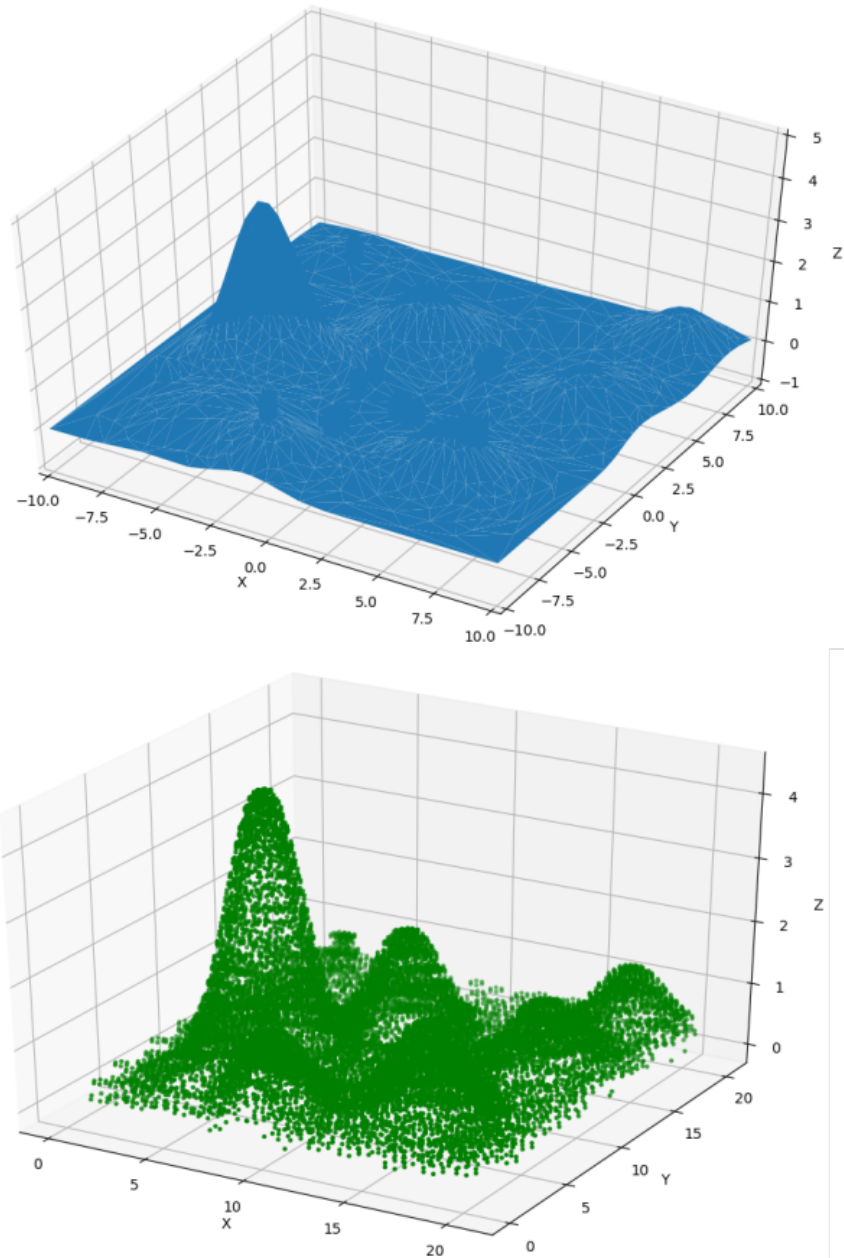
Figure 2: Example of polygonal model of terrain (top) and the generated candidate viewpoints (bottom).

To build a list of appropriate viewpoint which are covering maximal number of polygons we use the following algorithm:

1. Sort all cells in descending order according to their weights.
2. Choose the first/next cell with maximum weight and calculate the number of polygons covered by the corresponding cell viewpoint p and the total number of covered polygons

n.

3. Estimate the ratio of surface covering:

$$R = \frac{n}{N_P},$$ (8)

where N_P is a number of polygons in the model.

4. The viewpoint p is added into the list of viewpoints L.

5. If $R \geq R_t$ the algorithm stops, where $R_t \in (0, 1]$ is a predefined covering threshold. If this criterion is not reached the algorithm goes to step 2.

6. Return L.

3. Experiments

3.1. Experimental setup

For the quantitative evaluation of our approach we conducted a series of experiments. During experiments we evaluate the next parameters of the proposed method:

- grid resolution t;
- number of cell viewpoints n to reach the covering ratio $R_t = 0.9$;
- number of iterations I to reach R_t.

In the experiments we use the polygonal terrain model, which is shown in the Fig.1. The values of grid resolution was chosen from the sequence $t \in \{4, 5, 10, 16, 25, 32, 50, 64\}$

3.2. Results

The visualization of obtained hash table at different values of t and a corresponding weights is shown in the Fig.3. The dependence between the number of iterations I and the number of covered polygons is shown in the Fig.4. The dependence between the number of iterations I to cover $R_t = 0.9$ polygons and the grid resolution is shown in the Fig.5

4. Conclusion

The paper presents a novel approach to solve a problem of approximated estimation of a suitable set of viewpoints to cover the terrain. Our approach is based on a subdivision of a terrain model on a regular grid and on a filtering of a initial candidate viewpoint set with a spatial hashing technique and a greedy algorithm to solve a modification of a terrain cover problem. Our approach is aimed to a fast, non-iterative approximate generation of possible points of UAV motion path to cover the given landscape which is defined as a polygonal model.

The future work will consider the problem of generation of smooth, minimum-length trajectory of the UAV based on a generated viewpoint set.
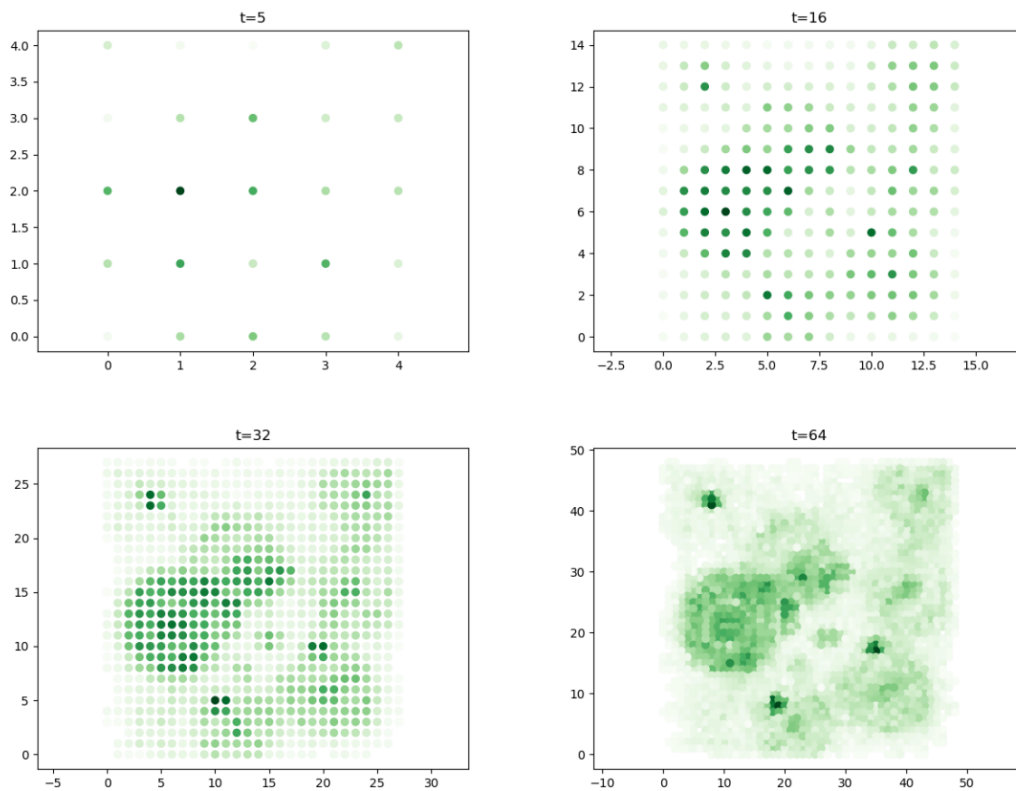
Figure 3: Visualization of hash table at different t values (darker cells contain more candidate viewpoints than the lighter ones).

References

[1] Tarabanis, K.A., Tsai, R.Y., Allen, P.K.: The MVP sensor planning system for robotic vision tasks. IEEE Transactions on Robotics and Automation. 11, 72–85 (1995).

[2] Scott, W., Roth, G., Rivest, J.-F.: View planning for automated three-dimensional object reconstruction and inspection. ACM Comput. Surv. 35, 64–96 (2003). https://doi.org/10.1145/641865.641868.

[3] Scott, W.R.: Model-based view planning. Machine Vision and Applications. 20, 47–69 (2007).

[4] Jing, W., Polden, J., Lin, W., Shimada, K.: Sampling-based view planning for 3D visual coverage task with Unmanned Aerial Vehicle. 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 1808–1815 (2016).

[5] Geng, L., Zhang, Y.F., Wang, J.J., Fuh, J.Y.H., Teo, S.H.: Mission planning of autonomous UAVs for urban surveillance with evolutionary algorithms. In: 2013 10th IEEE International Conference on Control and Automation (ICCA). pp. 828–833 (2013).

[6] Nilsson, U., Ogren, P., Thunberg, J.: Optimal positioning of surveillance UGVs. In: 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 2539–2544 (2008).

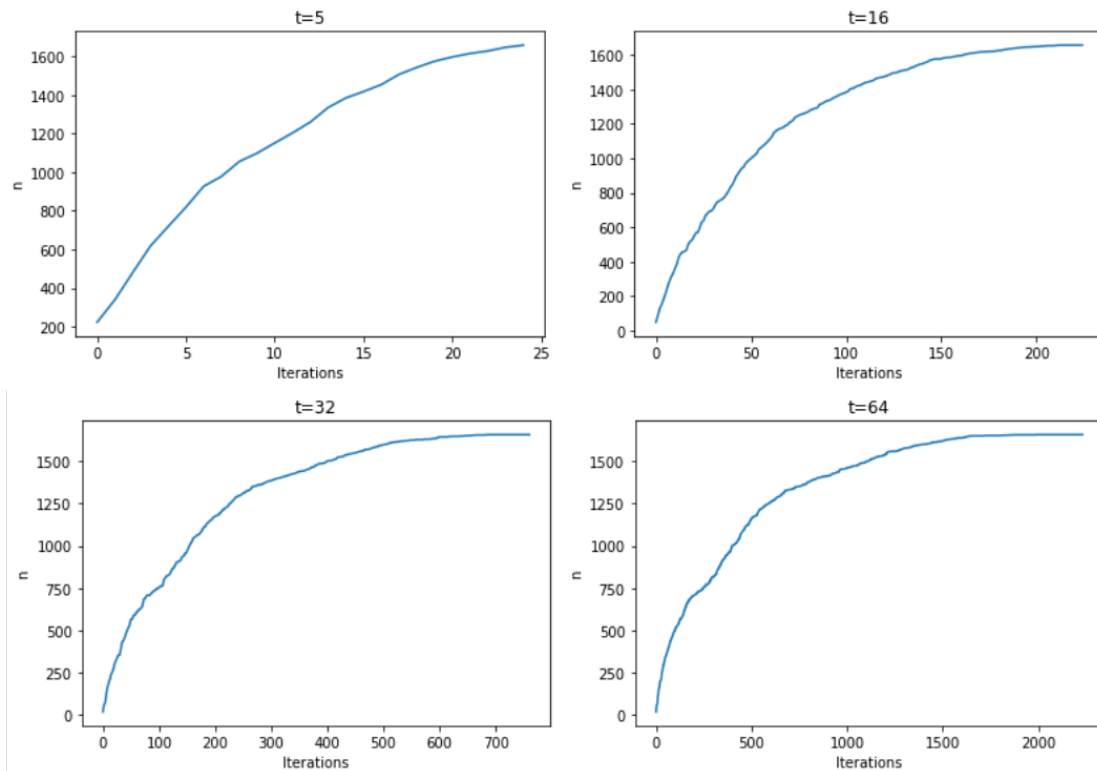[7] Jing, W., Shimada, K.: Model-based view planning for building inspection and

Figure 4: Number of iterations vs number of covered polygons.

surveillance using voxel dilation, Medial Objects, and Random-Key Genetic Algorithm. Journal of Computational Design and Engineering. 5, 337–347 (2018). https://doi.org/10.1016/j.jcde.2017.11.013.

[8] Wang, W., Tang, B., Fan, X., Mao, H., Yang, H., Zhu, M.: Efficient visibility analysis for massive observers. Procedia Computer Science. 111, 120–128 (2017). https://doi.org/10.1016/j.procs.2017.06.018.

[9] Chen, D.Z., Wang, H.: Visibility and ray shooting queries in polygonal domains. Computational Geometry. 48, 31–41 (2015). https://doi.org/10.1016/j.comgeo.2014.08.003.

[10] Carmi, P., Friedman, E., Katz, M.J.: Spiderman graph: Visibility in urban regions. Computational Geometry. 48, 251–259 (2015). https://doi.org/10.1016/j.comgeo.2014.10.004.

[11] Wołczowski, A.R.: Self-Correcting Trajectory Planning Using Modified Visibility Graph. IFAC Proceedings Volumes. 33, 611–616 (2000). https://doi.org/10.1016/S1474-6670(17)37998-3.

[12] Erdmann, M. and Lozano-Perez, T. (1987). On multiple moving objects. Algorithmica, 2:477–521.

[13] Latombe, J. C. (1990). Robot Motion Planning. Kluwer Academic Press.

[14] Fraichard, T. (1999). Trajectory planning in a dynamic workspace: A 'state-time space' approach. Advanced Robotics, 13(1):75–94.
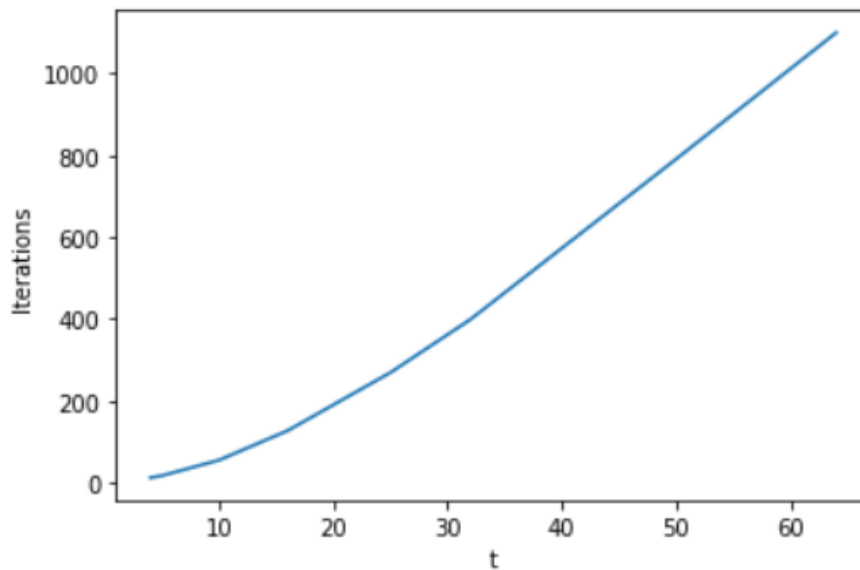
Figure 5: Number of iterations vs grid resolution.

[15] LaValle, S. M. and Kuffner, J. J. (1999). Randomized kinodynamic planning. In Proc. IEEE Int. Conf. on Robotics and Automation, pages 473–479, Detroit, MI.

[16] Hsu, D., Kindel, J. C., Latombe, J. C., and Rock, S. (2000). Randomized kinodynamic motion planning with moving obstacles. In Proc. Workshop on Algorithmic Foundation of Robotics, Hanover, NH.

[17] Goel, U., Varshney, S., Jain, A., Maheshwari, S., Shukla, A., 2018. Three Dimensional Path Planning for UAVs in Dynamic Environment using Glow-worm Swarm Optimization. Procedia Computer Science 133, 230–239. https://doi.org/10.1016/j.procs.2018.07.028

[18] Jun, M., D'Andrea, R., 2003. Path Planning for Unmanned Aerial Vehicles in Uncertain and Adversarial Environments, in: Butenko, S., Murphey, R., Pardalos, P.M. (Eds.), Cooperative Control: Models, Applications and Algorithms. Springer US, Boston, MA, pp. 95–110. https://doi.org/10.1007/978-1-4757-3758-5-6

[19] Sundar, K., Rathinam, S., 2014. Algorithms for Routing an Unmanned Aerial Vehicle in the Presence of Refueling Depots. IEEE Trans. Automat. Sci. Eng. 11, 287–294. https://doi.org/10.1109/TASE.2013.2279544

[20] Fu, Z., Yu, J., Xie, G., Chen, Y., Mao, Y., 2018. A Heuristic Evolutionary Algorithm of UAV Path Planning. Wireless Communications and Mobile Computing 2018, 1–11. https://doi.org/10.1155/2018/2851964

[21] Gao, X.-Z., Hou, Z.-X., Zhu, X.-F., Zhang, J.-T., Chen, X.-Q., 2013. The Shortest Path Planning for Manoeuvres of UAV. Acta Polytechnica Hungarica 10, 19.

[22] Wang, B., Bao, J., Zhang, L., Sheng, Q., 2018. UAV autonomous path optimization simulation based on radar tracking prediction. J Wireless Com Network 2018, 239. https://doi.org/10.1186/s13638-018-1260-9

[23] Torabbeigi, M., Lim, G.J., Kim, S.J., 2019. Drone Delivery Scheduling Optimiza-

tion Considering Payload-induced Battery Consumption Rates. J Intell Robot Syst. https://doi.org/10.1007/s10846-019-01034-w

[24] Hoang, V.T., Phung, M.D., Dinh, T.H., Ha, Q.P., 2018. Angle-Encoded Swarm Optimization for UAV Formation Path Planning, in: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Presented at the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, Madrid, pp. 5239–5244. https://doi.org/10.1109/IROS.2018.8593930

[25] Ozalp, N., Sahingoz, O.K., 2013. Optimal UAV path planning in a 3D threat environment by using parallel evolutionary algorithms, in: 2013 International Conference on Unmanned Aircraft Systems (ICUAS). Presented at the 2013 International Conference on Unmanned Aircraft Systems (ICUAS), IEEE, Atlanta, GA, pp. 308–317. https://doi.org/10.1109/ICUAS.2013.6564703