# GuideSwarm: A Drone Network Design to Assist Visually Impaired People

Görkem Sakarya(✉), Talip Tolga Sari, and Gökhan Seçinti

Istanbul Technical University, Istanbul, Turkey
{sakaryag,sarita,secinti}@itu.edu.tr

**Abstract.** Today, most of the metropolitan areas lack necessary infrastructure to guide visually impaired people. Even with an existing infrastructure, it is nearly impossible to ensure the safety of visually impaired people in dense/crowded urban areas. In this paper, we propose aerial swarm framework, namely GuideSwarm, cooperatively utilizing multiple drones to provide assistance to visually impaired people in crowded urban environments. In this manner, GuideSwarm first formally structures allocation and scheduling problems by defining different sets of assistance missions. Then, we form an optimization problem with the objective of minimizing average waiting time for the users and propose a heuristic method solving the defined optimization in $O(n^2)$ time complexity. Compared to the greedy approaches, our solution provides 19% less waiting time for users demanding assistance.

**Keywords:** UAV · Task scheduling · Visually impaired

## 1 Introduction

In World Report on Vision by World Health Organization [1], it is stated that there are 2.2 billion visually impaired people in the world and a significant number of these people suffer from either complete or nearly complete vision loss.

Visually impaired people have lots of troubles in their daily lives due to the lack of awareness of their surroundings. Streets in urban areas, crowded public transportation, shopping malls are only few of the examples, in where it is extremely inconvenient and stressful for people with such impairments to be. Such places/locations require solid infrastructure and dynamic control systems to guide these people through. Even today, most of the major cities still lack tactile pavements to help visually impaired people to follow sidewalks using white canes. In such places (where there is limited to none infrastructure), using guide dogs looks like a promising solution. Unfortunately, this is not a cost-effective solution due to high cost and effort in breeding, training and maintain these animals. In addition, they are prone to other people's reactions/interactions, where some people may naively try to pet them without realizing they are working animals and easily distract/disorient the people these animals are guiding.


I. M. Pires et al. (Eds.): GOODTECHS 2021, LNICST 401, pp. 139–152, 2021.
https://doi.org/10.1007/978-3-030-91421-9_11

In this paper, we propose to use Unmanned Aerial Vehicles (UAVs) to guide visually impaired people in large cities. In the system model we devised, multiple UAVs are located at different locations within the city, where they can be conveniently deployed for real-time assistance of visually impaired people. Inside a coverage area, our main purpose is providing assistance to user as soon as possible, where multiple users may request assistance simultaneously, which requires effective scheduling of multiple Unmanned Aerial Devices at different locations.

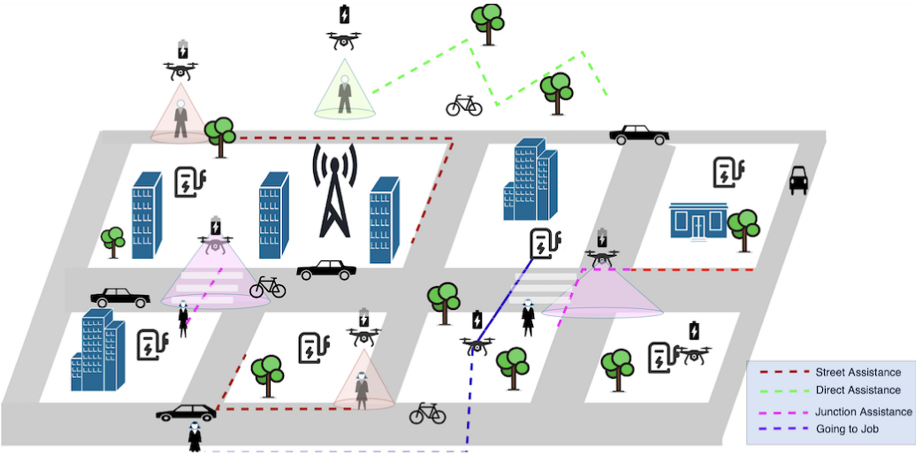


**Fig. 1.** System model

Figure 1 visualises the use cases we addressed in this paper. It shows the different types of assignments and the main principles of the system. All different assignment types are represented with dashed lines in different colors. The charge stations and representative current battery level are shown in the figure. Also the interchange between different assignment types are simulated. In order to solve the problem introduced above, we basically proposed a new scheduling algorithm in order to complete maximum number of tasks with minimum average waiting time for users. To this end, we formulate an optimization problem, with the constraints such as to satisfy power consumption and batter capacity limitation of UAVs as well as to assign suitable drone for awaiting tasks.

After we formulate the optimization problem, we employed several well-known scheduling algorithms, such as which are "First in First Out" and "Shortest Time First", in order to address the formulated allocation problem. Then, we devised a new heuristic algorithm in order to achieve a higher efficiency of obtaining smaller average waiting time than the other algorithms. The new heuristic gives better results for the optimization. The total of the time for going to task, doing the task and returning to the charging station for UAVs, either working or waiting, is used as a parameter. Also, while choosing the best UAV for the user, distance between UAVs' charging station, users' position and the target point are considered. Lastly, the UAV which have the minimum total time, is assigned to the user and this match enables us to have minimum average waiting time.

The main working principle is that, according to the optimization problem, the UAVs are being assigned to jobs. After this assignment, the assigned UAVs wait until the users are ready to work with UAV. After that the UAVs go to the users in coordination. When the user and the UAV is at the same location, they start to move together. While they are moving together, UAV will give some directives to the user in order to assist him/her. After they reach the target coordinates, the UAV returns to its own charging station. On the other hand, a UAV can be assigned to multiple jobs at different times. However, before starting to a new job, the drone must be charged to have enough charge level to meet the battery consumption of the next job. Also, UAVs have a safety threshold for being able to reach their charging stations before their charge finishes. When all of the waiting jobs are finished, all UAVs go to their stations and the program terminates.

This paper offers three main contributions as follows:

- We designed a system that provides assistance to visually impaired people. While providing this assistance, we have more than one UAV and multiple users in our scope.
- We formulate an optimization problem for scheduling the system according to the case that includes multiple task and multiple device.
- We implemented an algorithm that applies the optimization problem to the assistance system. We decreased the average waiting time of the user in the system by using this algorithm.

In Sect. 2, we examined some researches on both visually impaired assistance with different devices and applications of different types of task scheduling algorithms and in Sect. 3, we defined our optimization problem and explained the main structure of our project. In Sect. 4, we mentioned about our results and lastly in Sect. 5, we conclude our project briefly.

## 2 Related Work

There are various studies and products targeting visually impaired people, which recently include UAVs as well as other flourishing robotic technologies and equipment. These studies vary from, utilizing indoor robotic assistance for blind people [2], offering mobile robots to assist visually impaired people [11], and building drone-based navigation system for visually impaired person [5], to flying guide drones for runners [6]. Some of these studies rely on multiple agents to work in collaboration, where efficient task scheduling and allocation methods are required. There are various researches on Task Scheduling and one of them is, collaborative task allocation for multiple UAVs [8] and also some researchers work on control of multiple UAVs some of them are, algorithms and tests of controlling multiple UAVs [15], managing disaster with the help of UAVs [14]. Lastly, some researches with UAVs and/or task allocation which construct the base of our project are machine learning methods to communicate with UAVs [9] and task allocation for multiple robots by applying clustering [12]. The researches that work with reinforcement learning are using reinforcement learning for designing a project that schedule the UAV clusters [10] and applying multi-agent reinforcement learning techniques for dynamic task allocation [13].

Researchers have been working on different technological developments, and using some different devices for visually impaired people. Some of the researchers are using robots for assisting visually impaired people. Kulkarni et al. [2] developed an indoor robotic assistance for visually impaired people to follow the robot for avoiding obstacles and finding their ways. Also, in their research [3], Kulkarni et al. developed a mobile robot to assist visually impaired people more interactively using small robots. In an older research, Mori and Kotani [11] developed a mobile robot that assists visually impaired people by image processing and used some range sensors. As a different solution for visually impaired people, Simoes et al. [4] propose a wearable low-cost device for guiding visually impaired people in an indoor environment. They implemented an assistance system using a pair of glasses that contains cameras and sensors. Our Project also aims to apply different techniques and use UAVs for assisting visually impaired people and while doing this, we try to find more convenient ways to give them the opportunity to use this technology without need to hold something on hand or on them.

Similar to our project, there are some researches that are using Unmanned Aerial Devices to guide virtually impaired people. As one of these, Avila et al. [5] found a stable and reliable solution for the guidance system for visually impaired people. They developed a system by using a small drone to navigate virtually impaired people within an indoor area. Another UAV solution for blind people, Zayer et al. [6] proposed a UAV based assistance system for blind navigation. It provided a feasible system for UAV assistance by which visually impaired people could run by following the UAV. Unlike the other projects that project provided an outdoor solution. The main difference between our project and these researches is that we work on multiple UAV and multiple user domain. By applying the assistance in this way, we needed to deal with task allocation and coordination. In the paper by Trotta et al. [7] researchers have an approach to use multiple UAVs in coordination. They proposed a network architecture for being able to manage UAVs around a city. In this project we are also inspired from that approach to use multiple UAVs for visually impaired people's assistance within a city.

In addition to UAV assistance, our research focuses on the task allocation method for the UAVs. The projects that make task allocation vary according to their constraints and features. Some of these researches have been examined for having a background for our project. Fu et al. [8] studied an allocation problem to find the best allocation algorithm for their multi-UAV system and they presented a method. Like this research, there are lots of researches on multiple UAV collaboration and one of them is Nigam et al. [15] that tries to organize multiple UAVs and control them simultaneously. Erdelj et al. [14] also developed a system that works with multiple UAVs to help and coordinate on disasters. We also control and organize the UAVs on our simulation. While doing this, we implement some algorithms and also, we propose a new algorithm that increases the efficiency of the system.

Another research area is the application of some machine learning techniques on UAV control or task allocation. Bithas et al. [9] made a survey on different machine learning techniques to communicate with UAVs and while doing that they try different types of task allocation. Janati et al. [12] also stated that their proposed task allocation method increases the efficiency of task allocation on a large number of tasks and robots by applying the clustering method. Another technique to have a better task allocation is to use some reinforcement learning methods, as stated by Yang et al. [10]. They designed a Deep Q Learning method on UAV clusters to optimize the task schedule. Like this research, Noureddine et al. [13] proposed a method for dynamic task scheduling that is based on reinforcement learning and they improved the Deep Q learning method by adding some features for their purpose, and their method makes the cooperative task allocation. We planned to develop an agent based on reinforcement learning as a future work.

Briefly, our project combines some different methods from researches that work on the use of UAVs for the assistance of visually impaired and to achieve the necessary task allocation for this purpose. As a combination of all of these, we propose an optimization problem that is designed for multiple UAVs and multiple users and a new customized task allocation method to increase efficiency of the project.

## 3 System Model and Proposed Approach

Our project consists of two main phases to accomplish UAV assistance system for visually impaired people by implementing a task allocation. First, we construct an optimization problem to specify the main way to reach the purpose and to realize all of the limitations and constraints for this project. Following this, we propose a task allocation algorithm by using these constraints then, compared several different algorithms. As a result of these comparisons, we proposed an adapted task scheduling algorithm (Table 1).

**Table 1.** Symbol list

| Symbol | Refers |
|---|---|
| $\alpha$ | Unit discharge rate for street assistance |
| $\beta$ | Unit discharge rate for direct assistance |
| $\gamma$ | Unit discharge rate for junction assistance |
| $\theta$ | Charging rate |
| $\delta$ | Discharge rate when moving |
| $\zeta$ | Discharge rate when moving |
| $m$ | Total number of Jobs |
| $a_i$ | Assign Time of job $i$ |
| $t_i$ | Arrival time of user $i$ |
| $d_i$ | Duration of Job $i$ |
| $E_i$ | Whether there is street assistance for job $i$ |
| $D_i$ | Whether there is direct assistance for job $i$ |
| $J_i$ | Whether there is junction assistance for job $i$ |
| $R_{ij}$ | Whether drone $j$ returning from job $i$ |
| $G_{ij}$ | Whether drone $j$ going to job $i$ |
| $IDLE_i$ | Whether drone $i$ is idle |
| $C_j(t)$ | Charge level of $j$th drone at time $t$ |
| $Ch_j(t)$ | Whether drone $j$ is charging at time $t$ |
| $Q_i$ | Whether job $i$ is in the queue |
| $W_{ij}$ | Whether drone $j$ is assigned to job $i$ |
| MB | Maximum Battery Size |

### 3.1 Optimization Problem

The main objective is to minimize average waiting time which is the difference between the assign time and arrival time of user. The minimization function in order to achieve this objective is shown in below expression.

$$\min \frac{\sum_{i=0}^{m}(a_i - t_i)}{m} \tag{1}$$

$$\text{subject to: } E_i + D_i + J_i + Q_i \leq 1 \qquad \forall i \in Job \tag{2}$$

$$W_{ij} = 1 \implies$$
$$a_i = min\{t | C_j(t - g_{ij}) > (g_{ij} \times \delta + CNP_i + r_{ij} \times \delta + \zeta) \wedge (t - g_{ij} \geq t_i)\} \; \forall t \tag{3}$$

$$W_{kj} = 1 \;\wedge\; W_{lj} = 1 \implies$$
$$(a_k < a_l \;\wedge\; a_k + d_k + r_{kj} + ct_i < a_l - g_{lj}) \vee$$
$$(a_l < a_k \;\wedge\; a_l + d_l + r_{lj} + ct_i < a_k - g_{kj}) \tag{4}$$

$$Ch_j(t) = 1 \implies ((a_i - g_{ij} > t) \land (t > a_i + d_i + r_{ij})) \lor (a_i - g_{ij} < t) \tag{5}$$

Function (1) minimizes the average waiting time by using 3 variables, where $a_i$ defines the assign time of the job to drone, $t_i$ represents the arrival time of the user to the system and m stores the number of jobs. The minimization function meets the following constraints. The first constraint (2) implies that a job could be assigned to one of these variables at a time. For example, when a job is assigned to an street assistance it cannot be assigned to direct assistance. Another constraint (3) indicates that when $i^{th}$ job assigned to $j^{th}$ drone, the assign time of the $i^{th}$ job will be the minimum time that satisfies the condition is that the capacity of the $j^{th}$ drone at the starting time of the $i^{th}$ job will have higher capacity than the sum of going, processing and returning time and the safety threshold.

Lastly, constraint (4) is created to imply that if two jobs assigned to same drone, the time to work with the drone is not overlap. This expression creates the time to the drone for going to the user's location by subtracting this time from the assign time. For example, if $k^{th}$ job assigned before, the first assign time of $l^{th}$ cannot be earlier than the finish of the $k^{th}$ job. The last constraint 5 that we have created is for charging time of the drone. The charging time of the $j^{th}$ drone must be before a job assignment or after the assignment.

**Definitions.** The first part of the project is preparing an optimization problem and we developed our problem based on the features that we have decided. The main purpose of our project is to develop a UAV assistance for visually impaired people. Consequently, we defined Job and Resource variables. The Job stores the values for the users request as an array. While a user could have several job requests, one row of an array consists several values of a job which is a vector that specifies the path from the initial coordinate to target. It also has a variable which stores the assistance type and 3 variables that stores different times. These are duration, assigning time and arrival time of job. The assign time is defined with initial value -1 and when it is overwritten for assignment the assign time is assigned after the drone arrives to job location. The information about jobs stored in a two dimensional array that is called Jobs. The Jobs array consists all jobs with their information in 5 branches, where vector stores start and finish coordinates, type stores the type of job as Street, Direct or Junction assistance, $d_i$ stores the job duration, $a_i$ defines the assign time of the job to drone, $t_i$ represents the arrival time of the user to the system.

Type of the Job depends on user requirements and it has 3 different types which are Street, Junction and Direct assistance. The defined types correspond to the assistance of the UAVs. When the UAV assists the user while it is following a street in an environment, Junction assistance may contain multiple UAVs to help user to be able to pass the junctions. Direct assistance is the assistance method to users while they do not have a specific path like tactile paving. The features for assistance type are not considered for this project, they are only defined as a structure. Job types are defined as a boolean value that represents whether the user requests the corresponding type of Job.

On the other hand, the UAVs have some different features, definitions and status values. UAVs have 4 different statuses during the operations. These are Going to Job, Returning From the Job, Charging and IDLE. In the equation given below, these variables are defined with boolean type. The $G_{ij}$ is defined for the status that shows whether the drone is assigned to go to the user. If the value equals to 1, that means the $j^{th}$ drone assigned to $i^{th}$ job and drone started to go to the start position of the job from the base coordinate of itself. $R_{ij}$ is defined to specify that whether the drone assigned to return to base. If the value is equals to 1 the $j^{th}$ UAV completed $i^{th}$ job and returning from the target location of the job to the base location of itself. $Ch_j$ shows if it is equals to 1 the $j^{th}$ drone is charging. The last status is $IDLE_j$and which indicates that the UAV is ready to assign to a job. The structure of the Drone Status is given below.

$$UAV\_Status_{ij}(t) = \begin{cases} R_{ij} = 1, & j^{th}\ drone\ returning\ from\ i^{th} job \\ G_{ij} = 1, & j^{th}\ drone\ going\ to\ i^{th} job \\ IDLE_j = 1, & Drone's\ Status\ is\ IDLE \\ Ch_j(t) = 1, & Charging\ at\ time\ t \end{cases} \tag{6}$$

Also, there are some specifications for the UAVs which are Base Coordinates of the drones and Capacity of drones. These variables are stored and used for some constraints and functions. The Base coordinates stores a tuple that consists X and Y coordinates of the base station of corresponding drone into an array. Also, the capacity of drones stores the current battery level of the drone for tracking the capacity for planning and creating safety threshold. The $C_j(t)$ function gives a result, that shows the capacity level between 0 and 100, at time t.

Some of the definitions for the other variables are given below. The $Q_i$ is defined for show the status of a job. If the value equals to 1, that means the ith job is assigned to a UAV but it is waiting its starting time in a queue. $W_{ij}$ is a multi-dimensional list that stores the assignment of the user and job. When $i^{th}$ job assigned to $j^{th}$ job the corresponding value of $W_{ij}$ is assigned as 1. This variable is storing for tracking the assignments whether all jobs are finished or not.

$$Q_i = \begin{cases} 1, & i^{th}\ job\ is\ in\ queue \\ 0, & processing\ or\ not\ assigned \end{cases} \tag{7}$$

$$W_{ij} = \begin{cases} 1, & j^{th}\ drone\ assigned\ to\ i^{th}\ job \\ 0, & no\ relation \end{cases} \tag{8}$$

**Functions.** There are several functions to prepare the optimization function for implementation. These functions use the variables which are described before and store their values. The first function is calculating flying time from a location to another. As stated below, by dividing the path length to velocity of the $j^{th}$ drone the flying time is calculated. Going and returning times are calculated with this formula and task allocation will be done according to these variables.

$$FT(vector_i(start/end), BC_j) = \frac{Path}{V_j} \tag{9}$$

$$r_{ij} = FT(vector_i(end), BC_j) \tag{10}$$

$$g_{ij} = FT(vector_i(start), BC_j) \tag{11}$$

The consumption function below shows the total expended energy. As stated, before $\alpha$ refers the unit energy expended for street assistance, $\beta$ for direct assistance and $\gamma$ for junction assistance. And the consumption is calculated by multiplying these values with duration of job. At least one of the summing parts in the consumption function is not zero, because of the corresponding constraint that is given in the Eq. 7. The formula for consumption is given below.

$$CNP_i = \begin{cases} (\alpha \times d_i), & E_i = 1 \rightarrow\ street\ assistance\ to\ i^{th} job \\ (\beta \times d_i), & D_i = 1 \rightarrow\ direct\ assistance\ to\ i^{th} job \\ (\beta \times d_i), & J_i = 1 \rightarrow\ junction\ assistance\ to\ i^{th} job \end{cases} \tag{12}$$

Another function to our problem calculates the charging time after a job. The result of consumption function for $i^{th}$ job, divided by $\theta$ that stores the constant for charging amount within unit time.

$$ct_i = \frac{CNP_i}{\theta} \tag{13}$$

The last function calculates the capacity of drone at a time. Also in the function given below, the total consumption is calculated. This function have several conditions which calculate consumption during different states of drones. It calculates total_consumption in 6 cases, first and last cases occur when a drone has not an assigned job and it is ready to be assigned, there is no consumption. Second and fourth cases are specified for the consumption while going to job and returning from job. Third case also shows the consumption during the job and lastly fifth case shows the charging.

$$J_{ij}(t) = \begin{cases} 0\ ,\ job\ is\ not\ assigned & t < a_i - g_{ij} \\ UAV\_Status_{ij}(t) \times \delta\ ,\ going\ to\ user & a_i - g_{ij} \leq t < a_i \\ \dfrac{W_{ij} \times CNP_i}{d_i}\ , in\ job & a_i \leq t < a_i + d_i \\ UAV\_Status_{ij}(t) \times \delta\ ,\ returning & a_i + d_i \leq t < a_i + d_i + r_{ij} \\ -UAV\_Status_{ij}(t) \times \theta\ ,\ charging & t < a_i + d_i + r_{ij} + ct_i \\ 0\ ,\ IDLE & t \geq a_i + d_i + r_{ij} + ct_i \end{cases} \tag{14}$$

The last function calculates the capacity of drone at a time. In this function the charge level is calculated by subtracting the sum of total_consumption of all jobs from the maximum battery level.

$$C_j(t) = MB - \sum_{i=0}^{n} \sum_{s=0}^{t} J_{ij}(s) \tag{15}$$

### 3.2 Task Allocation Methods

Following the problem construction, we developed a task allocation algorithm to apply the optimization problem. This task scheduling algorithm makes the match between the job and drone, while doing this match it considers the main objectives of our optimization function and fits with the constraints. Although not all parts of the optimization problem are designed on the algorithm, it implements the main structure. It tries to decrease the average waiting time for randomized values. All of the values like, job number, job location, drone number and drone locations are selected randomly and the results are compared.

We researched lots of algorithms which could give better results on our project, and lastly, we decided to use Shortest Time and First Come First Serve scheduling algorithms. These algorithms are adapted to our optimization problem and coded in Python.

We implemented the First Come First Serve(FCFS) algorithm for our optimization problem and the main structure is built by selecting a drone for a job in a sequence while the time is being iterated. The sequence is determined according to the arrival time that is given at the $5^{th}$ column of the array for the corresponding job. Also, the shortest time algorithm is implemented and the Job array is sorted according to their duration and after the sort, it is iterated over time. The shortest time algorithm worked with a queue to take the sorted order at a time.

On the other hand, we proposed an algorithm that decreases the average waiting time of the jobs. This algorithm is developed by aiming to decrease average waiting time and for being able to do that it uses a priority queue. The priority queue is designed to return the id of job which will be finished earlier. The time includes both the drone's coming and returning time and the job duration. Because of that, the minimum time priority also avoided from selecting smaller jobs. Time is iterated in the algorithm to find assign time and it is checked with the arrival time of the job for avoiding to assign drones to jobs which are not arrived. However, this algorithm enables system to wait the jobs which could decrease average waiting time. Briefly, our method enables the system to select best job for a drone. We designed and tested this algorithm with different cases and the pseudo code of our algorithm is given below.

**Algorithm 1:** New Scheduling algorithm

**for all** $Jobs : Job_y$ **do**
  **for all** $Jobs : Job_x$ **do**
    **for** $Time : T$ **do**
      **if** $Job_x$ *is not assigned* **then**
        **if** $Drone(Y\%NumDrone)$ *available at* $(T, T + Work)$ **then**
          **assign** *max arrival time and* $T- >$ *temp_assign_time*
          **push** *temp_assign_time* $- >$ *Priority_Queue*
  **assign** *Priority_Queue*(*Min Finish Time*) $- >$ *Assign_Time*
  **assign** $Job_x$ *to* $Drone(Y\%NumDrone)$

## 4 Experimentation Environment and Experiment Design

We tested the implementation and examined the possible results in different cases. First Come First Serve algorithm gives bad results for the average waiting time because of the obscurity. Although, Shortest Time algorithm generally gives better results than FCFS, it gives too much waiting time for longer jobs. Also, the assignments could be inefficient because the distances between jobs and drones are not taken into account in these two algorithms. On the other hand, in lots of different cases our proposed heuristic gives better results than all of the other algorithms. In these cases, 4 drones are assigned to different number of users and according to the task scheduling method average waiting time is calculated and compared with each other A chart for different cases is given below (Fig. 2).

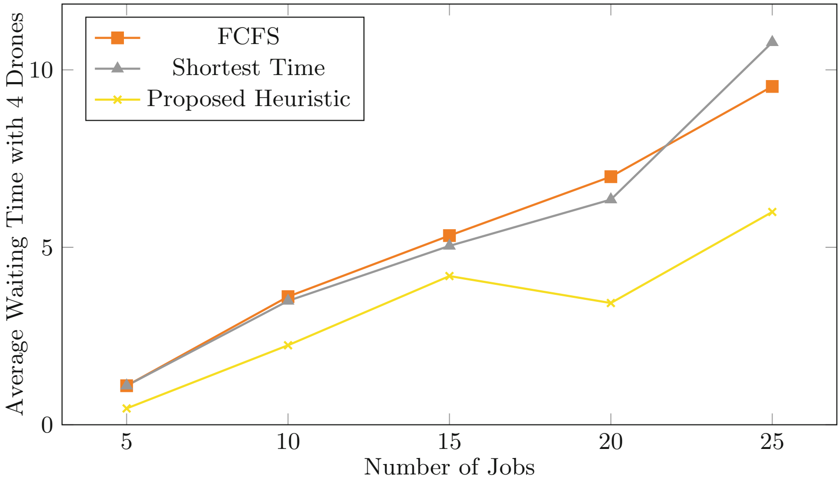


**Fig. 2.** Comparison of minimization functions.

For testing the program random values assigned to the variables of jobs and drones. This chart is created by using 4 drones and different number of jobs. The position of jobs and drones selected randomly. Also another graph that visualize the change of the average waiting time according to map size. While the map size increasing, the average waiting time of proposed heuristic increases less than the others. This test case created by using Poisson Distribution to arrival times and Gaussian Distribution to duration. While creating this chart 4 drones and 50 jobs are defined within the different maps (Fig. 3).

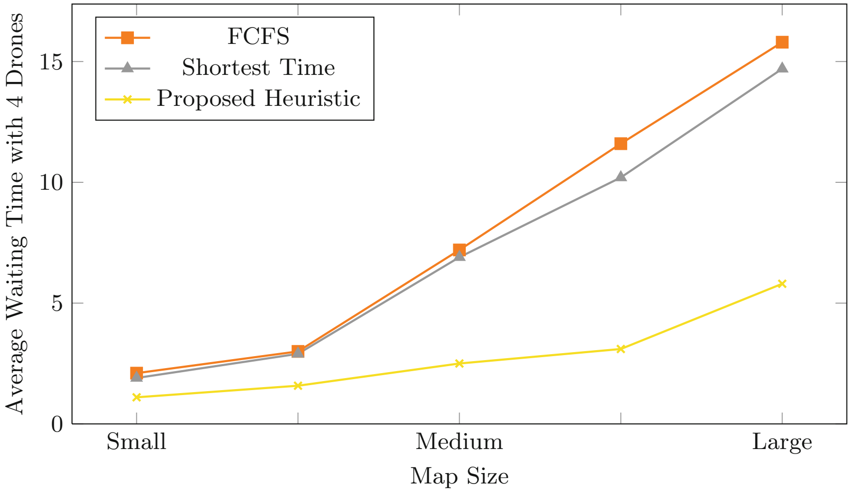


**Fig. 3.** Comparison of minimization functions on different map sizes

## 5 Conclusion and Future Work

Briefly, we designed a task allocation system for assisting the visually impaired people by cooperatively working multiple Unmanned Aerial Vehicles. The main purpose is making some sequence of operations to assist visually impaired people and this sequence is matching the distributed UAVs around the environment and the jobs which are created by the users. While doing this match, the task scheduling algorithms are being used to get better results. The results and the variables of task allocation algorithms depend on the optimization problem. Our optimization problem is constructed for minimizing the average waiting time of the jobs given the constraints. After comparing several task scheduling algorithms, a new algorithm has been proposed for the optimization problem. All of the experiments showed that using this algorithm decreases the average waiting time.

We determined some future works for the project. One of them is working on an agent that learns to schedule the jobs by using reinforcement learning and minimize the average waiting time. The other planned future work is calculating the energy consumption of the UAVs more accurate.

## References

1. World Health Organization. World report on vision. https://www.who.int/publications/i/item/world-report-on-vision
2. Kulkarni, A., Wang, A., Urbina, L., Steinfeld, A., Dias, B.: Robotic assistance in indoor navigation for people who are blind. In: 2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI), pp. 461–462. IEEE (2016)
3. Kulyukin, V., Gharpure, C., Nicholson, J., Osborne, G.: Robot-assisted wayfinding for the visually impaired in structured indoor environments. Auton. Robot. **21**(1), 29–41 (2006). https://doi.org/10.1007/s10514-006-7223-8
4. Simôes, W.C., De Lucena, V.F.: Blind user wearable audio assistance for indoor navigation based on visual markers and ultrasonic obstacle detection. In: 2016 IEEE International Conference on Consumer Electronics (ICCE), pp. 60–63. IEEE (2016)
5. Avila, M., Funk, M., Henze, N.: Dronenavigator: using drones for navigating visually impaired persons. In: Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility, pp. 327–328 (2015)
6. Al Zayer, M., Tregillus, S., Bhandari, J., Feil-Seifer, D., Folmer, E.: Exploring the use of a drone to guide blind runners. In: Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility, pp. 263–264 (2016)
7. Trotta, A., Andreagiovanni, F.D., Di Felice, M., Natalizio, E., Chowdhury, K.R.: When UAVs ride a bus: towards energy-efficient city-scale video surveillance. In: IEEE Infocom 2018-IEEE Conference on Computer Communications, pp. 1043–1051. IEEE (2018)
8. Fu, Z., Mao, Y., He, D., Yu, J., Xie, G.: Secure multi-UAV collaborative task allocation. IEEE Access **7**, 35579–35587 (2019)
9. Bithas, P.S., Michailidis, E.T., Nomikos, N., Vouyioukas, D., Kanatas, A.G.: A survey on machine-learning techniques for UAV-based communications. Sensors **19**(23), 5170 (2019)

10. Yang, J., You, X., Wu, G., Hassan, M.M., Almogren, A., Guna, J.: Application of reinforcement learning in UAV cluster task scheduling. Futur. Gener. Comput. Syst. **95**, 140–148 (2019)
11. Mori, H., Kotani, S.: Robotic travel aid for the blind: HARUNOBU-6. In: European Conference on Disability, Virtual Reality, and Assistive Technology (1998)
12. Janati, F., Abdollahi, F., Ghidary, S.S., Jannatifar, M., Baltes, J., Sadeghnejad, S.: Multi-robot task allocation using clustering method. In: Kim, J.-H., Karray, F., Jo, J., Sincak, P., Myung, H. (eds.) Robot Intelligence Technology and Applications 4. AISC, vol. 447, pp. 233–247. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-31293-4_19
13. Noureddine, D.B., Gharbi, A., Ahmed, S.B.: Multi-agent deep reinforcement learning for task allocation in dynamic environment. In: ICSOFT, Chicago, pp. 17–26 (2017)
14. Erdelj, M., Natalizio, E., Chowdhury, K.R., Akyildiz, I.F.: Help from the sky: leveraging UAVs for disaster management. IEEE Pervasive Comput. **16**(1), 24–32 (2017)
15. Nigam, N., Bieniawski, S., Kroo, I., Vian, J.: Control of multiple UAVs for persistent surveillance: algorithm and flight test results. IEEE Trans. Control Syst. Technol. **20**(5), 1236–1251 (2011)