2021 IEEE International Conference on Robotics and Automation (ICRA 2021)
May 31 - June 4, 2021, Xi'an, China

Fast Light Show Design Platform for K-12 Children

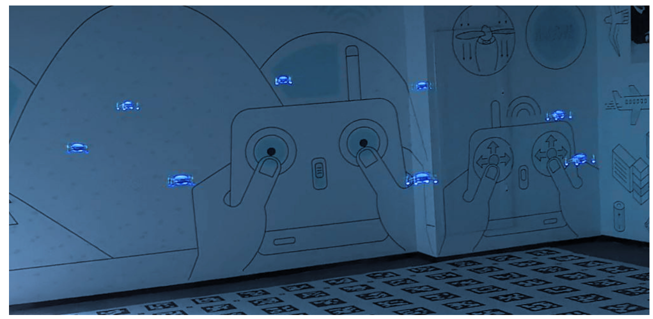
Pengda Mao[1], Yan Gao[1], Bo Wang[2], An Yan[2], Xiaoyu Chi[3], and Quan Quan[1]

***Abstract*— This paper aims to present a drone swarm light show design platform to support STEAM (science, technology, engineering, art and mathematics) education for K-12 children. With this platform, children can use this platform to design a drone swarm light show easily. To this end, the architecture of this platform contents three layers: UI layer, command layer, and physical layer. The UI layer has an easy-to-use interface for children. Children can feed parameters about the light show by clicking buttons and dragging sliders of four tracks. All actions designed for the swarm in the UI layer will be generated automatically in the form of the drone's desired trajectories through the command layer. The physical layer includes a router for communication and a drone swarm for the light show. Our experimental results demonstrate that this platform works efficiently and suits for being applied to real STEAM education.**

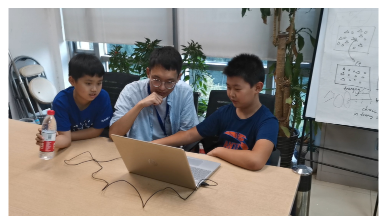
I. INTRODUCTION

In recent years, with more and more applications of robotics and automation in society, the demand for STEAM-related work continues increasing. Therefore, it is necessary to attract more children to engage in STEAM-related work in the future.

Educational robotics contribute a new way of STEAM education. The IEEE Robotics and Automation Society offers financial support for Technical Education Programs (RAS-TEP) [1] every year to make students closer to technology. The relationship between robotics and education has been studied before. Based on different roles of children when using educational robotics, educational robotics can be divided into four categories: social robots, robots kits, construction kits, and transparent educational robotics [2]. Among them, transparent educational robotics have become the future trend of educational robotics as they allow children to design and implement robotics independently. As a new type of educational robotics, educational drones can be used widely in children's STEAM education in the future as a new type of transparent educational robotics. Formbuena [3] proposed that drones can improve students' ability of spatial thinking which is a discipline of STEAM. Wu et al. [4] introduced an open-source tail-sitter micro aerial vehicle development platform: Phoenix drone for education and scientific research, which reduces the difficulty of getting started with beginners. Brand et al. [5] and Eller et al. [6] built low-cost drone platforms based on Raspberry Pi to teach undergraduates basic knowledge of drone control and state estimation. However, all these educational platforms are limited to college students and professional developers. Children cannot utilize these platforms to operate and design drones because of a lack of basic knowledge. The drone platform developed by Bermudez et al. [7] is specifically designed for the education of K-12 children. However it only stays in the simulation stage. The children have no interaction with drone directly. This is the initial motivation to develop a easy-to-use drone platform for K-12 children.



(a) Designed indoor swarm light show



(b) Children designing their light shows

(c) Fii F600 drone

Fig. 1. Education with indoor swarm light show.

Drone swarm light shows are attractive to most people, especially to children. Many large outdoor drone swarm light shows were recently demonstrated. Typically, Intel held a ceremony with 500 portable quadcopters in 2016[8]. Each Intel Shooting Star Mini drone is equipped with an LED payload that can produce over 4 billion color combinations. It is a big opportunity to make drones popular among children. However, the safety, expenses, and user-friendliness of drones are barriers to overcome for STEAM education. This motivates us to focus on the indoor drone light show, which is performed within a confined space. There have been some works on swarm platform and drone light show design. Soria et al. [9] built a drone swarm simulator based on MATLAB

This work was supported by Beijing Natural Science Foundation [grant number L182037].

[1]Pengda Mao, Yan Gao and Quan Quan are with School of Automation Science and Electrical Engineering, Beihang University, Beijing, 100191, P.R. China {maopengda, buaa_gaoyan, qq_buaa}@buaa.edu.cn

[2]Bo Wang and An Yan are with Goertek Robotics Co., Ltd., Shanghai, 201799, P.R. China {kalman.wang, leoyan.yan}@goertek.com

[3]Xiaoyu Chi is with Qingdao Research Institute of Beihang University, Qingdao, 266100, P.R. China terry.chi@goertek.com

for algorithm development and education, but this simulator lacked flight verification. Du et al. [10] realized fast and synchronized light show performances with 25 drones while lacking a simple interactive interface. DJI's RoboMaster TT [11] achieved indoor drone swarm light shows, but the trajectories for drones can only be designed one by one. Even the implementation of simple graph transform action has a high threshold and is dangerous, because of the possibility of exceeding the space. High Great's Fylo EDU [12] used a graphical interface to design the indoor drone swarm light show, but lack of various actions and graphics causing the performance to be compromised.

Motivated by these above, a drone swarm light show prototype design platform with various actions and graphs is introduced in this paper. Our drone light show platform lowers the age of education to K-12 children for the first time, which allows them to complete the entire process of design, simulation, and flight with some basic training, as shown in Fig. 1. This prototype platform is modular in structure. In terms of software, various functions are separated and packaged into standalone modules, which can be used as drag-and-drop modules of Scratch. In terms of hardware, as shown in Fig. 1(c), Goertek's Fii F600 drones [13] are used which are cheap, light and safe suitable for indoor use. With the proposed platform, numerous attractive drone shows can be designed easily. Children are able to select the graphics, set the light, choose music, and actions by clicking and dragging with our real experiment. Through using this platform to design drone light shows, children invited have showed their spatial thinking, art design, and mathematics ability. At the same time, the colorful light shows made them have strong interests. The contributions on challenges we have overcome are: (i) a layered platform architecture design for K-12 children to operate easily; (ii) basic swarm actions design used as elements to generate complex actions; (iii) graph transform, trajectory planning (collision avoidance and motion synchronization) and safety checking because of narrow space indoor realized automatically.

The rest of the paper is organized as follows. Section II describes the architecture of the platform. Section III explains the design of the UI layer. Section IV provides the design of the command layer. Section V describes the experimental results. Section VI draws the future work and conclusions.

II. PLATFORM ARCHITECTURE DESIGN

In this paper, the proposed platform consists of two parts: the software including a UI and algorithms behind, and the hardware including a router for sending data from PC to drones and real drones. It should be noted that we only focus on software design here. With generated command files in an appropriate format, many existing drone swarms can perform light shows. From another perspective, the architecture of the platform is stacked with three layers, as shown in Fig. 2: UI layer for children to use, command layer for generating command files, and the physical layer for actual flying. The functions of these three layers are in the following.

An important consideration for UI is that it must have concise logics and interfaces for K-12 students. A lot of graphical interfaces are designed in the UI to make students accept and operate easily. In the UI, students input drones parameters, space, and actions by clicking buttons and dragging blocks, which is able to avoid illegal inputs efficiently. There are four tracks to be designed, including a basic transform action track, a graph transform track, a music track, and a light track. The music track is filled with a chosen background music after the action track being designed. First, the number of drones and parameters of space should be set with default values. Secondly, after setting basic parameters, children should select or design a satisfying graph. Thirdly, children should select, and put action blocks into the action track. In blocks, rich parameters are set for children's purposes. In order to make the interaction friendly, action parameters, including time slots, have their default values and can be obtained by dragging a slider to prevent illegal inputs. After setting an action block, the simulation results will be previewed. Repeating the above process, children can complete the whole actions for the drone swarm light show. Finally, the light color is selected.

The higher-level command from the UI layer will be further decomposed into desired trajectories (3D position, 3D velocity, and timestamp) for all drones. Concretely, action superposition, target point matching, trajectory planning for collision avoidance and safety checking are performed behind the desired trajectory generation.

Through WiFi connection, the trajectory command files with timestamps are sent to each drone via a router. Drones with controllers track their corresponding trajectories to perform the light show.

III. UI LAYER DESIGN

As shown in Fig. 3, UI layer design aims to improve interaction on the basis of functional requirements. We will introduce UI layer design from two aspects: functionality and interactivity.

A. Functionality

In terms of functionality, the light show platform needs to set the basic parameters first. Then, design four tracks, namely graph transform action track, basic transform action track, music track, and light track, as shown in Fig. 4. Among them, the trajectories of drones to realize the graph transform action are automatically generated by the platform. As a result, only a graph needs to be chosen for the graph transform action track. The basic transform action track has rotation, translation, zooming, and sine wave actions which are used to be superposed. The start times and end times of the four actions are set independently. Lights in different colors are selected for each graph. So far, the music track is only allowed to be filled with one chosen background music for simplicity.

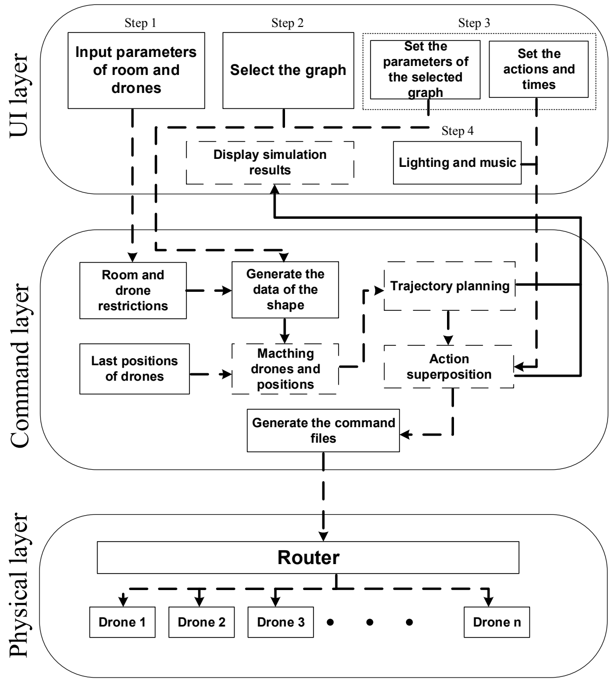


Fig. 2. Platform framework.

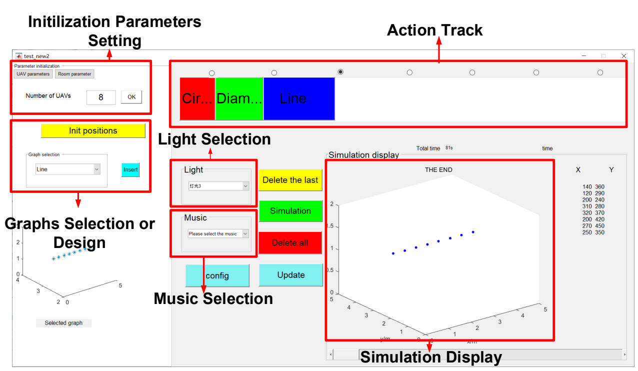


Fig. 3. Main interface of platform.

B. Interactivity

As is known to all, the keyboard input data is accurate but complex and error-prone for children. To improve interactivity, the keyboard input is reduced as much as possible. Instead, sliders and buttons are replaced to reduce input faults. As shown in Fig. 5, click radio buttons and drag sliders to select basic actions and set black axes which correspond to start times and end times. The graphs selected and simulation results will be displayed on the right to help children preview the designed drone swarm.

IV. COMMAND LAYER DESIGN

This section describes the design of the command layer as shown in Fig. 2, which consists of the basic transform action design and the graph transform action design.

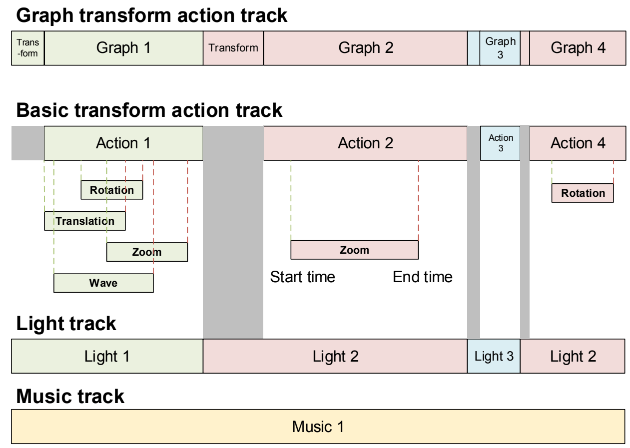


Fig. 4. Tracks design. Different colors of blocks in tracks correspond to different lights, except for music track. Every action in basic transform action track is the superposition of the basic transform actions including rotation, translation, zoom and wave.

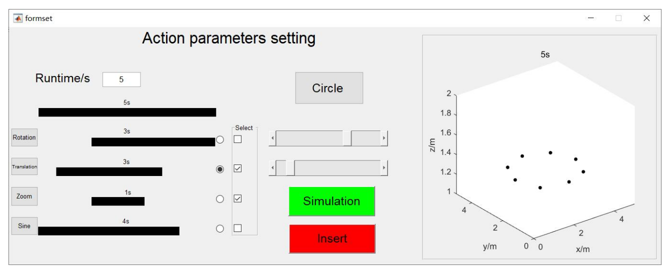


Fig. 5. Action parameter setting interface.

A. Basic transform action design

In order to make a light show more enjoyable, the drone swarm must have the ability to perform a variety of complex actions. However, the complexity of the action makes it difficult to design. In order to balance the complexity of the movement and the simplicity of the operation, we set several basic actions: rotation, translation, zooming and sine wave, as shown in Fig. 6.

1) Rotation: The rotation action needs to set the rotation center $\mathbf{r}_o \in \mathbb{R}^3$, the deflection angle of the rotation axis relative to the z-axis α, the time step t and the rotation angular velocity ω. Let the initial position coordinate of every drone in swarm be vector $\mathbf{p} \in \mathbb{R}^{\mathbf{3}}$. The rotation is described by means of a rotation matrix [14]. First, move the coordinate vector $\mathbf{p}$ to the origin and get the vector $\mathbf{p}_0$, which is shown

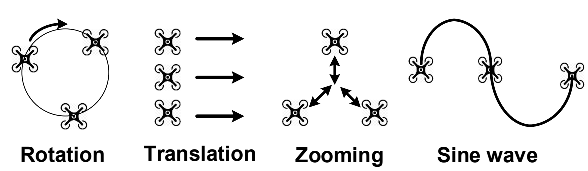


Fig. 6. Basic actions

as

$$\mathbf{p}_0 = \mathbf{p} - \mathbf{r}_o. \tag{1}$$

Then, two rotation operations should be carried out. The related formulas are list as follows:

$$\mathbf{p}_1 = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix}}_{\mathbf{B}} \mathbf{p}_0, \tag{2}$$

$$\mathbf{p}_2 = \underbrace{\begin{bmatrix} \cos(\omega t) & -\sin(\omega t) & 0 \\ \sin(\omega t) & \cos(\omega t) & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{A}} \mathbf{p}_1. \tag{3}$$

Combining the equations (1) (2) (3), the increment of coordinate vector $\mathbf{p}^r$ is obtained, which is expressed as follows:

$$\mathbf{p}^r = \mathbf{B}^{-1}\mathbf{p}_2 + \mathbf{r}_o - \mathbf{p} = (\mathbf{B}^{-1}\mathbf{A}\mathbf{B} - \mathbf{I})(\mathbf{p} - \mathbf{r}_o). \tag{4}$$

2) Translation: Let the translation speed of the drone swarm be $\mathbf{v}_t \in \mathbb{R}^3$. The increment of shifted coordinate vector $\mathbf{p}^t$ is shown as

$$\mathbf{p}^t = \mathbf{v}_t t. \tag{5}$$

3) Zoom: Let the zoom center be $\mathbf{r}_z \in \mathbb{R}^3$ and the zoom speed be $v_z \in \mathbb{R}$. Then the increment of coordinate vector $\mathbf{p}^z$ after one step zooming is expressed as

$$\mathbf{p}^z = \frac{\mathbf{p} - \mathbf{r}_z}{||\mathbf{p} - \mathbf{r}_z||} v_z t. \tag{6}$$

4) Sine wave: Let the vibration direction vector be $\mathbf{d}_s \in \mathbb{R}^3$, the amplitude be A_m and the vibration angular velocity be Ω. Then the increment of coordinate vector $\mathbf{p}^s$ is expressed as

$$\mathbf{p}^s = A_m \frac{\mathbf{d}_s}{||\mathbf{d}_s||} \sin(\Omega t). \tag{7}$$

A complicated actions $\mathbf{p}^*$ can be obtained through the vector synthesis of these basic movements above as follows:

$$\mathbf{p}^* = k_1(t)\mathbf{p}^r + k_2(t)\mathbf{p}^t + k_3(t)\mathbf{p}^z + k_4(t)\mathbf{p}^s + \mathbf{p}, \tag{8}$$

where $k_i(t) \in \{0, 1\}$ $(i = 1, 2, 3, 4)$ means whether the basic action is selected or not. If the time is between start time and end time of the basic action, the corresponding $k_i(t)$ will become 1, zero otherwise. As an exmaple, the Fig. 7 shows a complicated action for 8 drones.

B. Graph transform action design

The graph transform action design includes two parts: matching drones with target positions and trajectories planning. The essence of the matching problem can be viewed as a task assignment, so the auction algorithm [15] is introduced to match drones with target positions. For trajectory planning, the artificial potential field (APF) method is used to guide all drones to their desired positions.

1) Target assignment: We use auction algorithm for target assignment, which is described in **Algorithm 1**. The positions of drones compose a set $\boldsymbol{L}_{t1} = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, ..., \mathbf{x}_N\}$ and the positions of targets compose a set $\boldsymbol{L}_{t2} = \{\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, ..., \mathbf{y}_N\}$, where N is the number of drones and targets. The bidding value for the i-th drone to the j-th target P_{ij} is related to their Euclidian distance d_{ij}. For the i-th drone, the minimum distance from any target is depicted as d^i_{min}, while the second smallest value is d^i_{min2}. The coefficient θ is the constant value to provide the minimum bidding value.

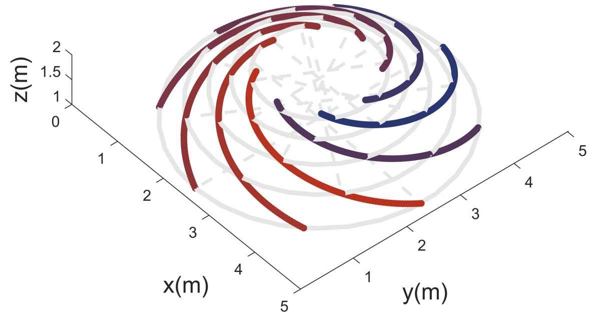


Fig. 7. An example of complicated actions which combine rotation, translation, and zoom actions for 8 drones. Different colors mark different drones' trajectories.

Algorithm 1: auction algorithm

Input: $\boldsymbol{L}_{t1}$, $\boldsymbol{L}_{t2}$
Output: $\boldsymbol{R}$
while $\boldsymbol{L}_{t1} \neq \oslash$ **do**
  for $\mathbf{x}_i$ *in* $\boldsymbol{L}_{t1}$ **do**
    for $\mathbf{y}_j$ *in* $\boldsymbol{L}_{t2}$ **do**
      $d_{ij} = ||\mathbf{x}_i - \mathbf{y}_j||_2$;
      if $d_{ij} < d^i_{min}$ **then**
        $d^i_{min} = d_{ij}$;
      end
      if $d_{ij} > d^i_{min}$ *and* $d_{ij} < d^i_{min2}$ **then**
        $d^i_{min2} = d_{ij}$;
      end
    end
    $P_{ij} = \frac{1}{d^i_{min}} - \frac{1}{d^i_{min2}} + \theta$;
    if $P^j_{before} + \alpha < P_{ij}$ **then**
      $P^j_{before} = P_{ij}$;
      $\boldsymbol{R} = \boldsymbol{R} \cup < i, j >$;
      $\boldsymbol{L}_{t1} = \boldsymbol{L}_{t1} \setminus \mathbf{x}_i$;
      $\boldsymbol{L}_{t2} = \boldsymbol{L}_{t2} \setminus \mathbf{y}_j$;
    end
  end
end

For the synchronization of the drone swarm, the maximum velocity $v_{m,i}$ should be set for each drone after matching completed. It is shown as

$$v_{m,i} = \frac{d_i}{d_{max}} v_{max}, \tag{9}$$

where v_{max} is the maximum speed of the drone. The distance d_{max} is the maximum flying distance among the drone swarm, and d_i is the distance between the i-th drone and its target. Problems of drone swarms synchronization can be further referred to [16], which is beyond the topic in this paper.

2) Trajectory planning: In APF, drones are labeled as positive charges and targets are labeled as negative charges. During the operation, the targets will attract their paired drones to get close to them, while no collisions will happen [17]. As shown in Fig. 8, the space around any drone is divided into three parts: safety area, avoidance area and irrelevant area.

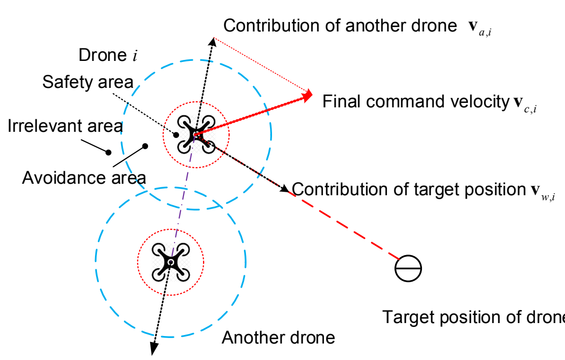


Fig. 8. The artificial potential field method.

After these areas are divided, how to generate the command velocity will be described. The command velocity $\mathbf{v}_{c,i}$ is affected by two parts: the contribution of the target $\mathbf{v}_{w,i}$ and other drones $\mathbf{v}_{a,i}$, which is shown as

$$\mathbf{v}_{c,i} = \mathbf{v}_{w,i} + \mathbf{v}_{a,i}. \tag{10}$$

When the drone position approaches the target position, the command velocity approaches zero.

C. Safety checking

Compared with the outdoor environment, the narrow space of the indoor environment makes it necessary to pay attention to the safety of the drone swarm. The trajectories of the drone swarm are defined as

$$\begin{aligned} &\mathbf{\Gamma}^i_{[t_0,t_1]} \triangleq \boldsymbol{P}_i(t) = [x_i(t), y_i(t), z_i(t)]^T, t \in [t_0, t_1], \\ &\mathbf{\Gamma}^i_{[t_0,t_1]}(x,y) \triangleq [x_i(t), y_i(t)]^T, t \in [t_0, t_1], \end{aligned} \tag{11}$$

where t_0, t_1 are the start time and end time.

The safety of the indoor flight is limited by the physical size of the indoor space $\boldsymbol{D} \subset \mathbb{R}^3$. The drone must not exceed the indoor height, width, and length. Otherwise, it will collide with walls or fly out of the allowed space. This restriction is described as

$$\mathbf{\Gamma}^i_{[t_0,t_1]} \subseteq \boldsymbol{D} \;\; i \in 1, 2, 3, \cdots N. \tag{12}$$

Similarly, the limitation of the drone speed $\boldsymbol{D}_v$ is shown as

$$\dot{\mathbf{\Gamma}}^i_{[t_0,t_1]} \subseteq \boldsymbol{D}_v. \tag{13}$$

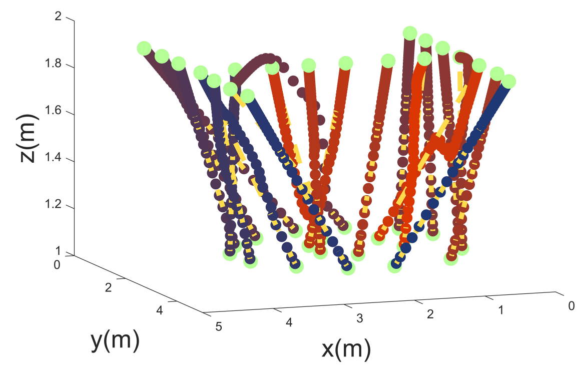


Fig. 9. An example of graph transform action. The green dots mark the initial and target positions of the drones, and the other colored dots mark the drones' trajectories. The yellow dotted lines mark the matching relationship between the initial positions and target positions.

In addition, the limitations on drones mean that drones should not get too close to each other, and should not have occlusion vertically for each drone in a stable state, because the drones' indoor localization often relies on the down-looking camera to the QR codes on the ground. Hence, this occlusion happens if

$$\begin{aligned} &\exists\, t_1, t_2, i, j, |t_1 - t_2| > \Delta t, \\ &\mathbf{\Gamma}^i_{[t_1,t_2]}(x,y) \cap \mathbf{\Gamma}^j_{[t_1,t_2]}(x,y) \neq \varnothing \end{aligned} \tag{14}$$

where Δt is the sampling time of the down-looking camera. After completing the action design, children can start the simulation. The platform will automatically carry out the safety checking. If the trajectory of the simulation does not meet (12) and (13) but (14), the platform will immediately stop and report errors.

Combine target assignment, trajectory planning, and safety checking, an example of graph transform action is shown in Fig. 9.

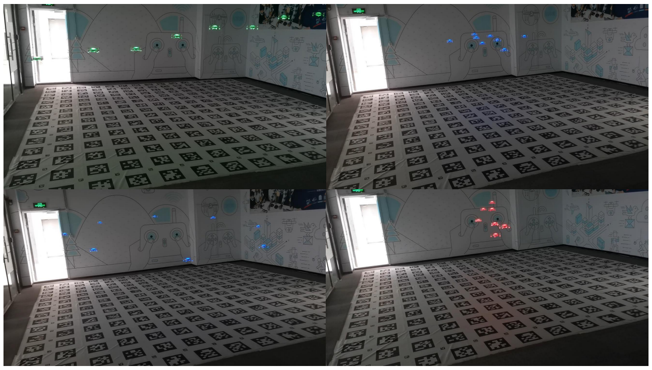
V. EXPERIMENTAL RESULTS

In this section, we will present the results of this platform in three parts: UI operation, flight results, and influences. To test this platform, we invited some children to use this platform. The hardware uses Fii F600 drones and an laptop with an Intel i7-8550U CPU and 8GB RAM. Video is available at `https://youtu.be/T3xj3GlrEis` or `https://j.mp/3qrZxqu`.

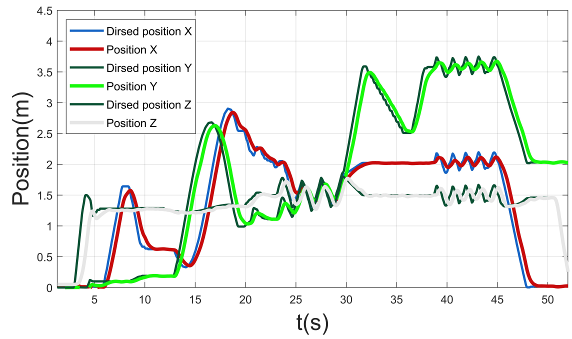
A. Flight result

After a demonstration operation and a practical operation under guidance, children can use the platform to independently complete the design of the light show. When the command files have generated, click the "Update" button, as shown in Fig. 3, the command files will be distributed to the drones through the router, and then the flight mission will be executed.

Choose the light show designed by one of the children as an example to analyze the performance of the drone swarm. According to the size of the drone, the safety radius of the drone r_m is set to 0.1 m, the avoidance radius r_a is 0.15 m,



(a) Four different graphs with colorful lights and actions.



(b) Desired and real trajectories for drone 1.

Fig. 10. An interesting drone swarm light show designed by children.

the time step is 500 ms, and the maximum velocity v_{max} is 3 m/s. The indoor space is a cuboid, with a length ranging from 0 to 5m, a width ranging from 0 to 5m, and a height ranging from 1 to 2m. Set the number of drones to 8. The actual flight is shown in Fig. 10(a). It can be seen that the lights change with different actions and graphs show a good effect. The following situation of the drones to the desired position is shown in Fig. 10(b). We recorded the actual flight trajectory of drone 1 among 8 drones. Compared with the desired trajectory of the drone 1, the actual trajectory shows that the drone follows the expected trajectory well, so that the drone swarm can achieve the simulation results and verify the effectiveness of the drone model we established before. The minimum distance among drones in the swarm at each moment is shown in Fig. 11. From the figure, it can be seen that the minimum distance in the whole process is not less than the safety radius of two drones, so there was no collision among drones. The actual flight results further show that the platform realizes the indoor drone light show designed by children.

B. *Educational effects on children*

To study the effects of the drone swarm light show platform on children, the children were interviewed before and after using, and the children's reactions when using the platform were also observed.

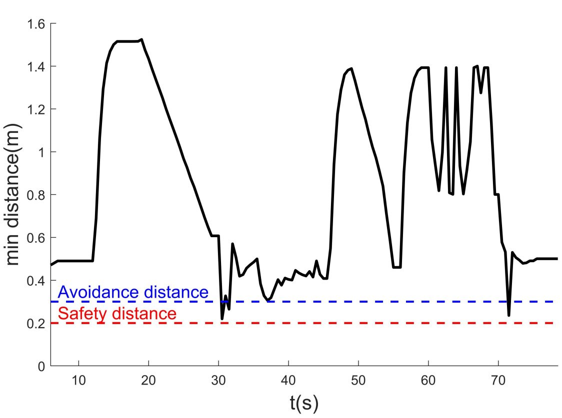


Fig. 11. The minimum distance among drones.

Before using this platform, the children claimed that they had little knowledge about drones in the school and did not know much about the drone swarm light show. Before children operating this prototype platform independently, half an hour of teaching was conducted through examples and guidance. When operating it, children showed great interest in actions and graphics design. They continued to conceive a variety of actions and graphics to observe the results through simulation. Because of the unconstrained thinking of children, the simulation of the drone swarm often triggered the safety checking. At this time, the children started to think about the reasons about the error including the positions of the drone swarm and the relationship between the movement and space, and adjusted the reasonable speed and direction of the movement. Then the final wonderful and shocking light show attracted children's interest. After the use, the children expressed their great interest in drones and drone light shows and really applied the mathematical knowledge learned in school. As for this platform, children thought the simulation display inside was very interesting and UI was easy to operate(getting started within 1 hour for them).

VI. CONCLUSIONS AND FUTURE WORK

The prototype of an interactive platform is built for STEAM education of children, with the purpose of enabling children to achieve drone swarm light show through user-friendly operations. The interactive platform changes the previous ways in that, as far as we know, all drone's trajectories have to be set one by one or be generated automatically without interaction for education. This new way greatly simplifies the process of setting, freeing students out of the complex operation. Along this way, the core tasks, namely the light show design and the cultivation of interest in STEAM, can be focused on. Compared with unmanned vehicles and robotic arm platforms, the proposed educational drone swarm light show has greater spatial flexibility of exercising children's three-dimensional spatial thinking ability. In the future, under the modularized framework, Scratch [18], a graphical programming language, can take the place of the current MATLAB GUI for other various education purpose.

REFERENCES

[1] Technical Education Program. Accessed October 3, 2020. [Online]. Available: https://www.ieee-ras.org/educational-resources-outreach/technical-education-programs

[2] M. Virnes, "Four Seasons of Educational Robotics," *The University of Eastern Finland: Kuopio, Finland*, 2014.

[3] A. Frombuena, "Unmanned Aerial Vehicles and Spatial Thinking: Boarding Education with Geotechnology and Drones," *IEEE Geoscience and Remote Sensing Magazine*, vol. 5, no. 3, pp. 8–18, 2017.

[4] Y. Wu, X. Du, R. Duivenvoorden, and J. Kelly, "The Phoenix Drone: An Open-source Dual-rotor Tail-sitter Platform for Research and Education," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 5330–5336.

[5] I. Brand, J. Roy, A. Ray, J. Oberlin, and S. Oberlix, "PiDrone: An Autonomous Educational Drone Using Raspberry Pi and Python," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 5697–5703.

[6] L. Eller, T. Guérin, B. Huang, G. Warren, S. Yang, J. Roy, and S. Tellex, "Advanced Autonomy on A Low-Cost Educational Drone Platform," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 1032–1039.

[7] A. Bermúdez, R. Casado, G. Fernández, M. Guijarro, and P. Olivas, "Drone Challenge: A Platform for Promoting Programming and Robotics Skills in K-12 Education," *International Journal of Advanced Robotic Systems*, vol. 16, no. 1, pp. 1–19, 2019.

[8] Shine Bright with Inter Drone Light Shows. Accessed February 11, 2020. [Online]. Available: https://www.intel.cn/content/www/cn/zh/technology-innovation/aerial-technology-light-show.html

[9] E. Soria, F. Schiano, and D. Floreano, "SwarmLab: A Matlab Drone Swarm Simulator," 2020. [Online]. Available: http://arxiv.org/abs/2005.02769

[10] X. Du, C. E. Luis, M. Vukosavljev, and A. P. Schoellig, "Fast and In Sync: Periodic Swarm Patterns for Quadrotors," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 9143–9149.

[11] RoboMaster TT. Accessed October 11, 2020. [Online]. Available: https://www.dji.com/cn/robomaster-tt/specs

[12] Fylo EDU. Accessed February 11, 2020. [Online]. Available: http://www.hg-fly.com/en/product.php?prodId=20

[13] F600 Educational Drone. Accessed February 11, 2020. [Online]. Available: http://www.fii-edu.com/Index.aspx

[14] Rotation Matrix. Accessed February 11, 2020. [Online]. Available: https://mathworld.wolfram.com/RotationMatrix.html

[15] D. P. Bertsekas, "The Auction Algorithm: A Distributed Relaxation Method for the Assignment Problem," *Annals of Operations Research*, vol. 14, no. 1, pp. 105–123, 1988.

[16] B. Zhu, Q. Zhang, and H. H. Liu, "Design and Experimental Evaluation of Robust Motion Synchronization Control for Multivehicle System without Velocity Measurements," *International Journal of Robust and Nonlinear Control*, vol. 28, no. 17, pp. 5437–5463, 2018.

[17] Q. Quan, *Introduction to Multicopter Design and Control*. Springer, 2017.

[18] Scratch. Accessed February 11, 2020. [Online]. Available: https://scratch.mit.edu/