engineering proceedings

MDPI

*Proceeding Paper*

# Computer Simulation of Anti-Drone System [†]

**Nikita Bykov [1] and Vadim Fedulov [2,*]**

[1] Russian University of Transport (MIIT), Obraztsova st., 9, Building 9, Moscow 127994, Russia; bykovnv@bk.ru
[2] Bauman Moscow State Technical University (BMSTU), 2-nd Baumanskaya, 5, Moscow 105005, Russia
[*] Correspondence: vadimfedulov.bmstu@gmail.com; Tel.: +7-977-654-1722

[†] Presented at the 15th International Conference "Intelligent Systems" (INTELS'22), Moscow, Russia, 14–16 December 2022.

**Abstract:** In this article, we present the results of an anti–drone system simulation. The system is designed to counter mini unmanned aerial vehicles. A radar system with one or several antennas and an elimination system with one or more countermeasures are included in the system. The drones are destroyed by kinetic weapons. In the developed computer model, it is possible to simulate a raid of several drones against several countermeasures in an environment without obstacles. The computer model-specific feature is a discrete-event approach that provides higher calculating performance compared with the "soft time" method.

**Keywords:** anti-drone system; drones; UAV; simulation; modelling; radar; detection; elimination; countermeasures; discrete-event approach

check for updates

**Citation:** Bykov, N.; Fedulov, V. Computer Simulation of Anti-Drone System. *Eng. Proc.* **2023**, *33*, 24. https://doi.org/10.3390/engproc2023033024

Academic Editors: Askhat Diveev, Ivan Zelinka, Arutun Avetisyan and Alexander Ilin

Published: 14 June 2023

## 1. Introduction

One of the most famous modern urban planning concepts is the concept of the Smart City. The main goal of the Smart City is effective city management and ensuring a high quality of life for citizens [1]. Urban specialists talk about the need to create a wide and advanced Internet of Things (IoT) infrastructure to implement this concept. It is expected that unmanned vehicles (cars, buses, trains, etc.) will be one of the most important parts of the IoT. For example, IoT will make it possible to improve city traffic management and to use unmanned public transport [2]. Moreover, small unmanned aerial vehicles (UAVs) may operate as a part of services for the delivery of various commodities [3]. Mini-UAVs are also widely used in modern military conflicts for aerial reconnaissance, guidance and fire adjustment of artillery [4].

However, there are a number of questions regarding the safety of using drones as part of the modern city system [5,6]. Potential security threats include drone hacking and cyber attacks on or using drones, which in turn can lead to attacks on critical urban infrastructure. Threats also exist in the field of combat application of mini-drones. One of the most effective ways to prevent illegal actions of drones is the physical destruction of drones. Such destruction methods in an urban environment should be carried out in a manner that is safe for people.

The relevance of our work can be attributed to the lack of studies on the effectiveness of complex anti-drone systems. There are few works on related topics. For example, the authors of the article [7] propose a method for determining the probability of hitting a drone using an assault rifle. They take the target geometry into account. A defence system based on the employment of cooperative interceptor drones is discussed in the study [8]. The interceptor drones operate in cooperation and can destroy the target through various countermeasures. In particular, in the article [9], the authors propose applying a group of interacting drones to physically neutralize the target using a usual net. The net blocks the target propellers. Nonetheless, there are no papers in which a complex system with countermeasure and detecting and localizing subsystems is analysed.

We present the computer model of the mini unmanned aerial vehicle (UAV) countermeasures system (the anti–drone system) acting in 3D space. The model takes into account subsystems used to detect, localise, and eliminate targets. It allows us to solve a *direct task*. The direct task of a counteracting the drones' raid involves obtaining the counteraction results and accumulating statistics for given initial and boundary conditions.

## 2. Model Description

The anti-drone system (ADS) consists of

- The countermeasure subsystem with one or several countermeasures.
- The radar subsystem with one or more active antennas and one detector that are detect and localize targets.

The counteraction process is considered in empty 3D space. There are no obstacles, terrain folds or anything else that can hide observable objects in this space.

The computer model is developed using the Python 3 programming language. Its SimPy package [10] is applied for the discrete-event approach realization.

### *2.1. Simulation Process Description*

The general scheme of the simulation process is shown in Figure 1.

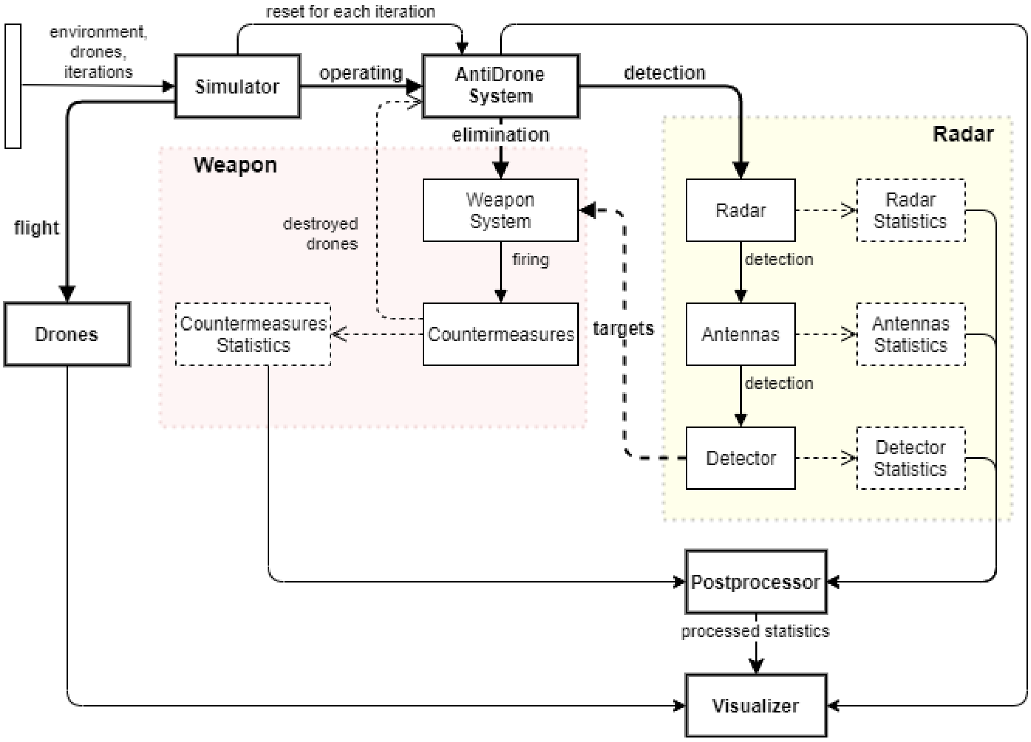

**Figure 1.** The general scheme of counteraction process.

Before modelling, instances of drones (`drones`), antennas (`Antennas`), detector (`Detector`), countermeasures (`Countermeasures`), weapon (`Weapon`) and radar (`Radar`) subsystems are initialized using the parameters given by the researcher. The weapon and radar subsystems instances form the anti–drone system model (`AntiDroneSystem`). This model contains the necessary fields and methods for starting and linking simulation processes. In addition, at this step, an instance of the environment (`environment`) is created to implement the discrete-event approach.

Initialized objects are passed to the simulator (`Simulator`) input. The simulator starts two parallel processes:

- The flight process (`flight`) of every drone.
- The ADS' operating process (`operating`).

The `AntiDroneSystem` instance starts two parallel processes, too:

- The countermeasures `elimination` processes.
- The radar detecting and localizing process (`detection`).

The `elimination` process begins the `firing` process of every counterweapon. The radar's `detection` process initializes the `detection` process of every antenna. The antennas, in turn, start the detector process `detection` when there is a desired signal in the detector's input. The detector tries to detect targets and, in case of successful detection, it puts the target into a shared list of targets (`targets`).

The dynamic list (`targets`) of detected targets shared between countermeasures system and radar is used by the targets distributor of the countermeasures system. The targets distributor activates the `firing` process of the selected countermeasure. When there are no targets in the `targets` list, the `firing` processes are inactive.

The antenna, detector, drones, weapon and other instances accumulate statistics. The simulation runs a given number of times (`iterations`). Before every run, states of all objects are reset and saved in files.

At the simulation end, accumulated statistics are passed in the postprocessor (`Postprocessor`) input. The postprocessor processes the data. The results are introduced to the visualization module (`Visualizer`).

### *2.2. The Drone Model*

The model of an unmanned aerial vehicle is an entity that has geometric shape and flies along a trajectory. A trajectory is the third-ordered Bezier curve. The UAV's 3D geometry consists of parallelepiped that is the central body and four ellipsoids that are engines. The drone model is described in more detail in the article [11].

### *2.3. The Gun Model*

The kinetic countermeasure is the same as in [11]. Additionally, the weapon model is described in more detail in those paper.

However, the weapon model presented in [11] has been improved; namely, it can interact with the radar subsystem. If the UAV is not detected by radars then the countermeasures have nothing to aim at. In cases in which the drone is detected, we make the assumption that the weapon subsystem has its own target tracking system; for example, an optoelectronic. Due to this system, the countermeasures are able to track the target and calculate the aiming point.

The criterion for destroying a target is at least one hit in its projection in the *picture plane*. The picture plane is perpendicular to the line of sight "weapon–target center of mass" and contains the UAV center of mass. The drone projection on the picture plane is calculated for each shot.

### *2.4. The Radar Model*

The radar includes one or more antennas and one detector. The major function of the antenna is to determine the direction and distance to the target which flew into the radar area.

Coordinates from the antenna output are directed to the detector input. The purpose of the detector function is to establish the fact of detection and initiate the target track. It allows us to approximate the drone trajectory using the radar for the initial aiming of countermeasures. It requires $N_{obs}$ detection in a row to initiate the target track. The target tracking is a necessary condition to input the target trajectory parameters into the weapon subsystem. If the target is not detected for $N_{lost}$ times in a row then there is a mistrack. In this case, the algorithm of detection starts over [12,13].

The output voltage of the detector depends on many factors, including those of a random nature, as well as on the specific implementation of the radar data processing path [12,13]. Therefore, it was decided to simplify the detector model as follows.

The detector is characterized by probabilities of false alarm $p_{FA}$ and correct detection $p_{CD}$. The random value of the output voltage $U_D$ is generated according to the normal distribution law $U_D \sim \mathcal{N}(\mu_{U_D}, \sigma^2_{U_D})$ with the mean voltage value $\mu_{U_D}$ and the standard deviation $\sigma_{U_D}$. If $U_D > U_T$, where $U_T$ is the threshold, then, taking into account the probability $p_{CD}$, a decision is made to detect the object. The target coordinates and velocity are measured. Otherwise, the target is not detected.

There is always a thermal noise in the detector input. This interfering signal negatively affects the accuracy of the detector. We describe this thermal noise as a white Gaussian noise. When there is no useful signal in the input, the output voltage value is distributed according to the normal law $U_{noise} \sim \mathcal{N}(0, \sigma^2_{U_{noise}})$ with the standard deviation $\sigma_{U_{noise}}$ [13].

The threshold voltage value $U_T$ is determined according to the Neuman–Pirson criterion and based on a given level of false alarm probability $p_{FA}$. As a rule, $p_{FA}$ is of order $10^{-2} \ldots 10^{-8}$ and $p_{CD}$ is taken to be 0.9 [12]. The values of standard deviations $\sigma_{U_D}$ and $\sigma_{U_{noise}}$ are derived experimentally or through computer simulation of the radar operating [13].

The antenna instance is initialized with distance range $[R_{min}; R_{max}]$, azimuth angle range $[\varphi_0; \varphi_1]$, elevation angle range $[\theta_0; \theta_1]$, rotation speed $\omega_a$, etc. The $R_{max}$ value depends on the radar cross-section (RCS) value of the drone. The antenna can operate in sector mode or a full overview (Figure 2). There are $R_{min} = 0$ and $\theta_0 = 0$ on the Figure 2.

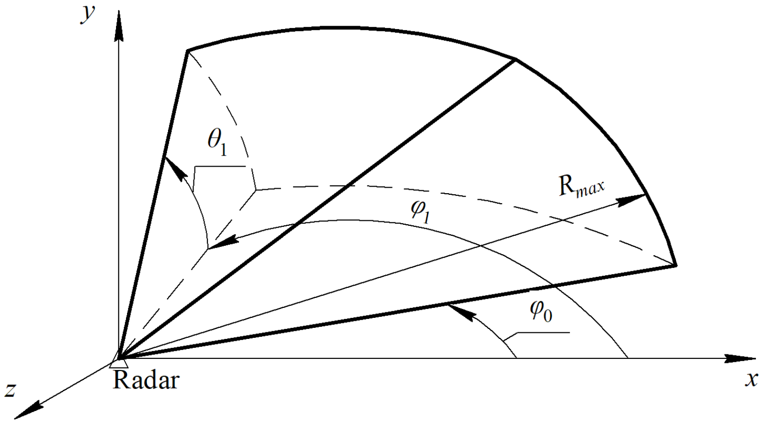

**Figure 2.** The antenna sector area.

## 3. Simulation Results

Flights of several (up to five) drones have been simulated. UAVs were eliminated by one or more countermeasures. In each run, the coordinates of the guns changed in the range ±500 m along the $Ox$ and $Oz$ axes; the coordinate along the $Oy$ axis was constant and equalled 0. Each drone spawned at a random point in space with coordinates from the following ranges: $x_0 = -1000$ m, $y_0 \subset [250; 750]$ m and $z_0 \subset [-1000; 1000]$ m. For each drone, the end point of the trajectory was randomly generated with the next set of coordinates: $x_1 = 1000$ m, $y_1 \subset [250; 750]$ m and $z_1 \subset [-1000; 1000]$ m. The parameters of drones and countermeasures are presented in the Table 1. Table 2 contains the coordinates and dimensions of the drone components. All drones are the same. Parameters of the radar antenna are shown in the Table 3. A detection system with one antenna was modelled. When a target was hit, the target distributor assigned the closest tracked drones to the newly freed guns.

**Table 1.** The drone parameters.

| Parameter Name | Distribution Law | Distribution Parameters |
|---|---|---|
| Initial speed | Normal | $\mathcal{N}(\mu = 21, \sigma = 3)$ m/s |
| Angle between initial velocity and $Ox$ axis | Uniform | $[-30°; 30°]$ |
| Initial drone angle of attack | Uniform | $[-10°; 10°]$ |
| Final speed | Normal | $\mathcal{N}(\mu = 21, \sigma = 3)$ m/s |
| Angle between final velocity and $Ox$ axis | Uniform | $[-30°; 30°]$ |
| Final drone angle of attack | Uniform | $[-10°; 10°]$ |

**Table 2.** The drone geometry.

| Airframe Part | Local Coordinates, mm | Axis Dimensions, mm |
|---|---|---|
| Central body (parallelepiped) | $(0; 0; 0)$ | $(400; 400; 250)$ |
| 1st engine(ellipsoid) | $(450; 300; 450)$ | $(600; 50; 600)$ |
| 2nd engine(ellipsoid) | $(-450; 300; 450)$ | $(600; 50; 600)$ |
| 3rd engine(ellipsoid) | $(450; 300; -450)$ | $(600; 50; 600)$ |
| 4th engine(ellipsoid) | $(-450; 300; -450)$ | $(600; 50; 600)$ |

**Table 3.** The radar parameters.

| Parameter Name | Unit Measure | Value |
|---|---|---|
| Position | m | $(0; 0; 0)$ |
| Rotation speed | rad/s | $\pi/4$ |
| Range | m | 0 to 600 |
| Azimuth angle range | — | 360° |
| Elevation angle range | — | $[0; 90°]$ |
| Detector mean output voltage | V | 12 |
| Detector output voltage standard deviation | V | 3 |
| Detector noise voltage standard deviation | V | 1 |
| Correct detection probability | — | 0.9 |
| False alarm probability | — | $10^{-7}$ |

The simulation results are shown in Figures 3–5. The number of iterations is 200,000.

The destroyed drones fraction shown in Figure 3 is the result of modelling the ADS without a radar subsystem. In this case, the one drone raid against an anti-drone system with one countermeasure was simulated. The countermeasure parameters are listed in Table 4. There are also accuracy variables in this table:

- $\sigma_{aim}$ is a standard deviation of the aiming point at a distance of 1 m to the target;
- $\sigma_b$ is a standard deviation of the mean aiming point of the burst at a distance of 1 m;
- $\sigma_s$ is a standard deviation of each individual hit point at a distance of 1 m.

These parameters are used to generate random points of hits in a picture plane of a target, as described in [11]. Random points are distributed according to the normal law.

**Table 4.** The countermeasure parameters.

| Parameter Name | Unit Measure | Value |
|---|---|---|
| Fire rate | shots/min | 800 |
| Burst length | — | 10 |
| Shots capacity | — | 120 |
| Aiming accuracy $\sigma_{aim}$ | — | $(10^{-3}; 10^{-3})$ |
| Burst shots accuracy $\sigma_b$ | — | $(1.6 \times 10^{-3}; 1.2 \times 10^{-3})$ |
| Individual shot accuracy $\sigma_s$ | — | $(1.5 \times 10^{-3}; 1.5 \times 10^{-3})$ |

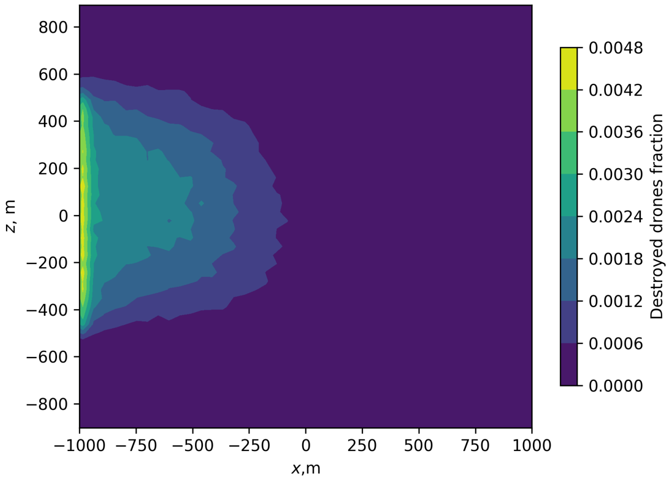

**Figure 3.** Spatial probability density of drones destroyed by one weapon without radar.

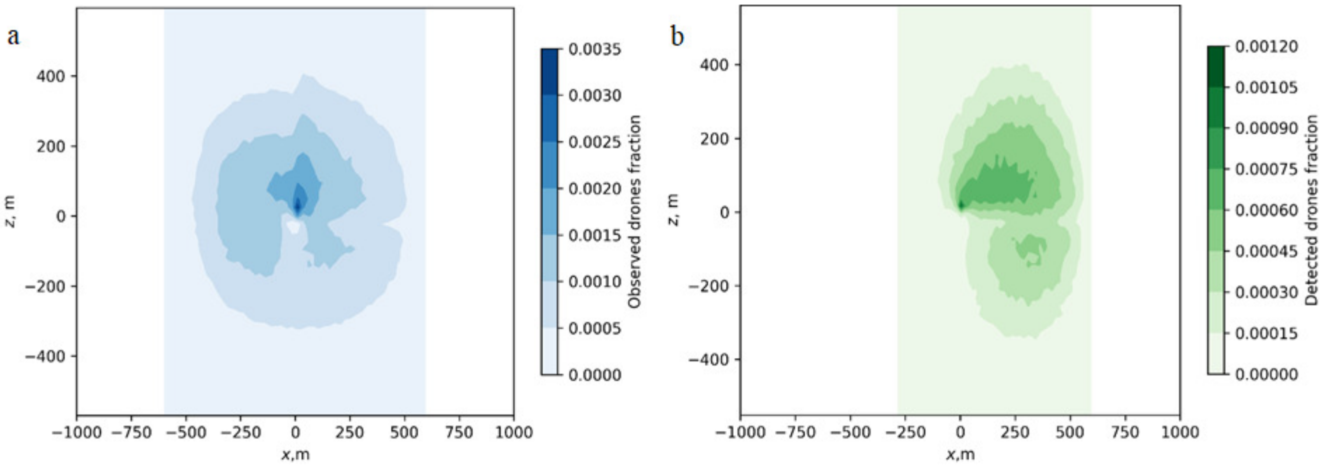

**Figure 4.** Spatial probability density of (**a**) observed and (**b**) detected drones.

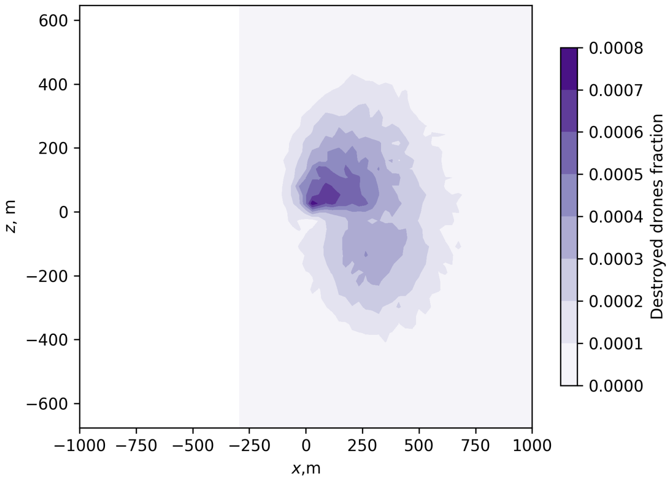

**Figure 5.** Spatial probability density of destroyed drones.

The contour graphs of observed and detected drones quantity are presented in Figure 4a,b. The *Oxz* plane corresponds to the top view. It can be seen from the figures that the graph in Figure 4b is shifted relative to Figure 4a in the direction of the drones' flight. This is due to the need to detect the target $N_{obs}$ times in a row to capture its trajectory. In

Figure 5, the percentage of destroyed targets is shown. This shows that countermeasures are more effective against closer targets.

From Figures 3 and 5, it can be seen the inclusion of a radar in the counter-drone system significantly affects the position of target destruction zones in space.

The computer model also stores the simulation statistics; for example, the consumption of ammunition and the number of drones that flew to the end point [11], etc. In the future, this information can be used to calculate the efficiency criterion for anti–drone system. In addition, our computer model allows us to compare anti-drone systems that have radar subsystems with different accuracy.

## 4. Conclusions

The direct task of simulating the process of counteracting UAV raids can be solved employing the developed computer model. The discrete-event approach provides greater calculating performance and better scalability of the computer model.

The computer model of the counter-drone system will be improved. Obstacles and terrain folds will be added and intelligent agents of drone and detection and destruction subsystems control systems will be trained to more effectively control them. This will allow the application of machine learning or artificial neural networks and conduct research on an complex countermeasure system with intelligent agents. The intelligent agent of the countermeasure system will take into account the consequences of the UAV fall.

It will be possible to solve the inverse problem of modelling; namely, the need to create a more efficient anti-drone system structure.

**Author Contributions:** Supervision, N.B.; software, V.F.; writing—original draft preparation, V.F., writing—review and editing, N.B. All authors have read and agreed to the published version of the manuscript.

**Funding:** The work was supported by RFBR grant 19–29–06090 mk.

**Data Availability Statement:** Data sharing not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Nomenclature

| | |
|---|---|
| $\mathcal{N}$ | Normal distribution |
| $\mu$ | Mean value |
| $\sigma$ | Standard deviation |

## References

1. Roldan, J.J.; Garcia-Aunon, P.; Pena-Tapia, E.; Barrientos, A. SwarmCity Project: Can an Aerial Swarm Monitor Traffic in a Smart City? In Proceedings of the 2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), Kyoto, Japan, 11–15 March 2019; pp. 862–867.
2. Alsamhi, S.H.; Ma, O.; Ansari, M.S.; Almalki, F.A. Survey on Collaborative Smart Drones and Internet of Things for Improving Smartness of Smart Cities. *IEEE Access* **2019**, *7*, 128125–128152. [CrossRef]
3. Abualigah, L.; Diabat, A.; Sumari, P.; Gandomi, A.H. Applications, Deployments, and Integration of Internet of Drones (IoD): A Review. *IEEE Sens. J.* **2021**, *21*, 25532–25546. [CrossRef]
4. Calcara, A.; Gilli, A.; Gilli, M.; Marchetti, R.; Zaccagnini, I. Why Drones Have Not Revolutionized War: The Enduring Hider-Finder Competition in Air Warfare. *Int. Secur.* **2022**, *46*, 130–171. [CrossRef]
5. Laufs, J.; Borrion, H.; Bradford, B. Security and the smart city: A systematic review. *Sustain. Cities Soc.* **2020**, *55*, 102023. [CrossRef]
6. Ilgi, G.S.; Ever, Y.K. Critical analysis of security and privacy challenges for the Internet of drones: A survey. In *Drones in Smart-Cities*; Elsevier: Amsterdam, The Netherlands, 2020; pp. 204–214.
7. Racek, F.; Balaz, T.; Krejci, J.; Prochazka, S.; Macko, M. Tracking, aiming, and hitting the UAV with ordinary assault rifle. In Proceedings of the SPIE 10441, Counterterrorism, Crime Fighting, Forensics, and Surveillance Technologies, Warsaw, Poland, 11–14 September 2017; pp. 112–122.
8. Castrillo, V.U.; Manco, A.; Pascarella, D.; Gigante, G. A Review of Counter-UAS Technologies for Cooperative Defensive Teams of Drones. *Drones* **2022**, *6*, 65–101. [CrossRef]

9. Rothe, J.; Strohmeier, M.; Montenegro, S. A concept for catching drones with a net carried by cooperative UAVs. In Proceedings of the IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), Würzburg, Germany, 2–4 September 2019; pp. 126–132.
10. SimPy. PyPI 2020. Available online: https://pypi.org/project/simpy/ (accessed on 30 August 2022).
11. Tovarnov, M.S.; Bykov, N.V.; Vlasova, N.S.; Fedulov, V.A.; Pozharsky, A.A. Computer simulation of the physical neutralization of drones in a Smart City. *J. Phys. Conf. Ser.* **2022**, *2308*, 012003. [CrossRef]
12. Bakulev, P.A. *Radar Systems. Textbook for High Schools*; Radiotechnics: Moscow, Russia, 2004.
13. Verba, V.S.; Gavrilov, K.Y.; Ilchuk, A.R.; Tatarsky, B.G.; Filatov, A.A. *Radiolocation for Everyone*; Technosphere: Moscow, Russia, 2020.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.