

УДК 629.7.05:004.4

doi: 10.32620/akt.2024.6.01

А. М. ГУМЕННИЙ<sup>1</sup>, П. П. МАЛАШТА<sup>1</sup>, С. В. ОЛІЙНИК<sup>1</sup>,  
О. Д. ПІДЛІСНИЙ<sup>2</sup>, В. В. АЛЕЙНІКОВ<sup>1</sup>

<sup>1</sup> Національний аерокосмічний університет ім. М. Є. Жуковського  
"Харківський авіаційний інститут", Харків, Україна

<sup>2</sup> Харківський національний університет Повітряних Сил  
імені Івана Кожедуба, Харків, Україна

## РОЗРОБКА НА LUA ПРОГРАМНОГО МОДУЛЯ ДЛЯ ВИЗНАЧЕННЯ ТОЧКИ СКИДАННЯ НЕКЕРОВАНОГО ВАНТАЖУ З БПЛА ЛІТАКОВОГО ТИПУ ІЗ СИСТЕМОЮ КЕРУВАННЯ ARDUPILOT

Доставка вантажів є одним з найбільш затребуваних напрямків застосування БПЛА літакового типу із системою керування Ardupilot. Автономні місії, які виконують такі БПЛА, пов'язані з необхідністю визначення точки скидання некерованого вантажу, що суттєво впливає на точність його доставки. Проте Ardupilot не комплектується програмним модулем визначення точки скидання некерованого вантажу, а таке додаткове програмне забезпечення рекомендується розробляти на Lua. Відомі програмні модулі для визначення точки скидання БПЛА, як правило, виконують циклічне розв'язання складної системи диференціальних рівнянь руху вантажу, для чого потрібні значні апаратні ресурси. Проте для висот до 400 м відносно поверхні землі можливо застосовувати спрощені аналітичні вирази руху вантажу при точності доставки  $\pm 5$  м для підвищення ефективності зазначеного програмного модуля. Отже, **об'єкт дослідження** – це процес визначення точки скидання некерованого вантажу з БПЛА літакового типу за допомогою Lua скрипта. **Предмет дослідження** – комп'ютерна модель руху некерованого вантажу на Lua, на підставі якої визначається точка скидання. **Метою роботи** є розробка на Lua програмного модуля для визначення точки скидання некерованого вантажу з БПЛА літакового типу із системою керування ArduPilot шляхом застосування спрощених рівнянь руху вантажу на висотах до 400 м. Отримано такі **результати**: розроблено загальний алгоритм роботи програмного модуля для визначення точки скидання некерованого вантажу, розроблено Lua скрипт для визначення точки скидання з висот до 40 м, розроблено Lua скрипт для визначення точки скидання з висот від 40 до 400 м відносно поверхні землі, надано Lua скрипт для визначення точки скидання з висот більше 400 м відносно поверхні землі, реалізовано інтерфейс програмного модуля для користувача. **Висновок**: на Lua розроблено програмний модуль для визначення точки скидання некерованого вантажу з БПЛА літакового типу із системою керування Ardupilot. Особливістю цього програмного забезпечення є застосування різних підходів для визначення точки скидання в інтервалах висот до 40 м, від 40 до 400 м і більше 400 м відносно поверхні землі. Це обумовлює більшу ефективність даного програмного модуля на малих висотах.

**Ключові слова**: безпілотний літальний апарат; БПЛА; скидання вантажу; система управління; алгоритм скидання; програмний модуль; точка скидання.

### 1. Вступ

#### 1.1. Мотивація дослідження

За останні роки суттєво збільшився спектр застосування безпілотних літальних апаратів (БПЛА) від розважальних заходів [1] до масштабних промислових рішень [2] та військових завдань [3]. Значна частина всіх місій, які здійснюють БПЛА, пов'язана з доставкою вантажів різного типу [3]: товари, продукти харчування, медикаменти і т. п. Виконання доставки за допомогою БПЛА літакового типу пов'язано

із необхідністю визначення точки скидання некерованого вантажу. При автономному польоті це здійснюється програмним модулем на борту БПЛА, який чисельним методом розв'язує систему диференціальних рівнянь, що описують рух некерованого вантажу [4, 5]. Проте, як показано в роботах [6, 7], на малих висотах падіння некерованого вантажу може бути описано спрощеними рівняннями при невеликій помилці доставки. Застосування зазначених рівнянь дозволить прискорити та спростити обчислення точки скидання програмним модулем, що важливо в умовах швидкої зміни положення БПЛА. До того ж, треба за-



значити, що сьогодні однією з найпопулярніших систем керування БПЛА з відкритим кодом є Ardupilot [8], яка підтримується низкою відповідного апаратного забезпечення. Для створення додаткових програмних модулів, функціонал яких відсутній в Ardupilot, рекомендується використовувати скриптову мову програмування Lua [8].

Все зазначене вище обумовлює актуальність розробки на Lua програмного модуля для визначення точки скидання некерованого вантажу з БПЛА літакового типу із системою керування Ardupilot.

## 1.2. Сучасний стан

Питання точного скидання було актуальним впродовж всього періоду розвитку та використання літальних апаратів для доставки вантажу парашутним, некерованим та керованим методом [9, 10]. Як свідчать роботи [11, 12], важливою складовою архітектури системи точного скидання вантажу з літального апарату літакового типу є модуль розрахунку точки скидання. Даний модуль, як правило, розв'язує систему диференціальних рівнянь руху вантажу (з парашутом або без нього) в повітряному просторі з використанням чисельних методів [13, 14]. Зазначений розв'язок виконується із застосуванням даних від системи датчиків БПЛА. В [5] для підвищення точності визначення точки скидання розрахунки виконуються циклічно із частотою у декілька герц. Проте, це підвищує навантаження на бортовий комп'ютер в рази. В роботі [13] додатково до вказаних розрахунків точки скидання було запропоновано застосування машинного зору для забезпечення високої точності доставки. Однак, це потребує завчасного маркування цілі та підвищує вартість обладнання БПЛА.

Загалом, підхід, заснований на розв'язанні системи диференціальних рівнянь, що описують рух некерованого вантажу, чисельними методами забезпечує високу точність визначення точки скидання, але потребує значних апаратних ресурсів.

У роботах [6, 7] показано, що при скиданні некерованого вантажу з висот до 40 м його рух можливо описати спрощеними аналітичними рівняннями при помилці доставки до  $\pm 5$  м. Не більша помилка доставки вантажу при скиданні з висот до 400 м прогнозується і в роботі [15]. З цього випливає потреба розробки програмного модуля розрахунку точки скидання вантажу з БПЛА літакового типу, в якому буде впроваджено результати з [6, 15] для підвищення ефективності даного програмного забезпечення.

## 1.3. Мета та підхід

Метою роботи є розробка на Lua програмного модуля для визначення точки скидання некерованого

вантаж з БПЛА літакового типу із системою керування ArduPilot шляхом застосування спрощених рівнянь руху вантажу на висотах до 400 м.

Методом дослідження є комп'ютерне моделювання на Lua руху некерованого вантажу для визначення його точки скидання в програмному середовищі Mission Planner від Ardupilot.

Комп'ютерна модель ґрунтується на розв'язанні диференціальних рівнянь руху некерованого вантажу та спрощених аналітичних рівняннях його руху на малих висотах.

При створенні комп'ютерної моделі руху некерованого вантажу на Lua використано параметри, їх позначення та величини, що наведено в таблиці 1. Вважалося, що вантаж має сферичну форму. Рух вантажу розглянуто відносно горизонтальної курсової системи координат (OXYZ) як у [15].

Для досягнення мети треба було вирішити наступні задачі:

- розробити загальний алгоритм роботи програмного модуля для визначення точки скидання некерованого вантажу (підрозділ 2.1);
- розробити Lua скрипт для визначення точки скидання некерованого вантажу з висот до 40 м відносно поверхні землі (підрозділ 2.2);
- розробити Lua скрипт для визначення точки скидання некерованого вантажу з висот від 40 до 400 м відносно поверхні землі (підрозділ 2.3);
- розробити Lua скрипт для визначення точки скидання некерованого вантажу з висот більше 400 м відносно поверхні землі (підрозділ 2.4);
- реалізувати інтерфейс для користувача (підрозділ 2.5);
- розглянути реалізацію розробленого програмного модуля для визначення точки скидання некерованого вантажу з висот 30, 300 та 1000 м (розділ 3);
- проаналізувати результати реалізації розробленого програмного модуля (розділ 4).

## 2. Результати дослідження процесу визначення точки скидання некерованого вантажу з БПЛА літакового типу

### 2.1. Загальний алгоритм роботи програмного модуля для визначення точки скидання некерованого вантажу

Реалізація загального алгоритму, блок-схема якого представлена на рис. 1, потребує введення даних від датчиків БПЛА: висота польоту відносно поверхні землі, повітряна швидкість, швидкість і напрямок вітру на висоті скидання, координати доставки, поточні координати БПЛА та інші. Треба зазначити,

Таблиця 1

Параметри, їх позначення та величини, що використовуються для створення комп'ютерної моделі руху некерованого вантажу на Суа

№	Позначення	Назва	Опис	Тип	Величина
1	m	маса вантажу	маса вантажу, що скидається з БПЛА, кг	числовий	10 кг
2	g	прискорення вільного падіння	прискорення, з яким вантаж рухається вниз під дією сили тяжіння, м/с <sup>2</sup>	числовий	9,81 м/с <sup>2</sup>
3	h0, dropHeight	початкова висота	висота скидання вантажу, м	числовий	10 – 10 <sup>3</sup> м
4	v0, initialVelocity	початкова повітряна швидкість	повітряна швидкість БПЛА у момент скидання вантажу, м/с	числовий	20 м/с
5	Angle, dropAngle	кут скидання	кут, під яким вантаж скидається відносно горизонту, кутові градуси (нульовий кут відповідає горизонтальному скиданню)	числовий	0°
6	wind_speed	швидкість вітру	швидкість вітру, який впливає на траєкторію вантажу після скидання, м/с	числовий	–
7	drag_coefficient, Cd	коефіцієнт опору повітря	безрозмірний коефіцієнт, що характеризує аеродинамічні властивості вантажу	числовий	0,47
8	Area, A	площа міделевого перерізу вантажу	площа поперечного перерізу вантажу, яка взаємодіє з повітряним потоком, м <sup>2</sup>	числовий	0,03 м <sup>2</sup>
9	air_density, rho	щільність повітря	щільність повітря, кг/м <sup>3</sup>	числовий	1,225 кг/м <sup>3</sup>

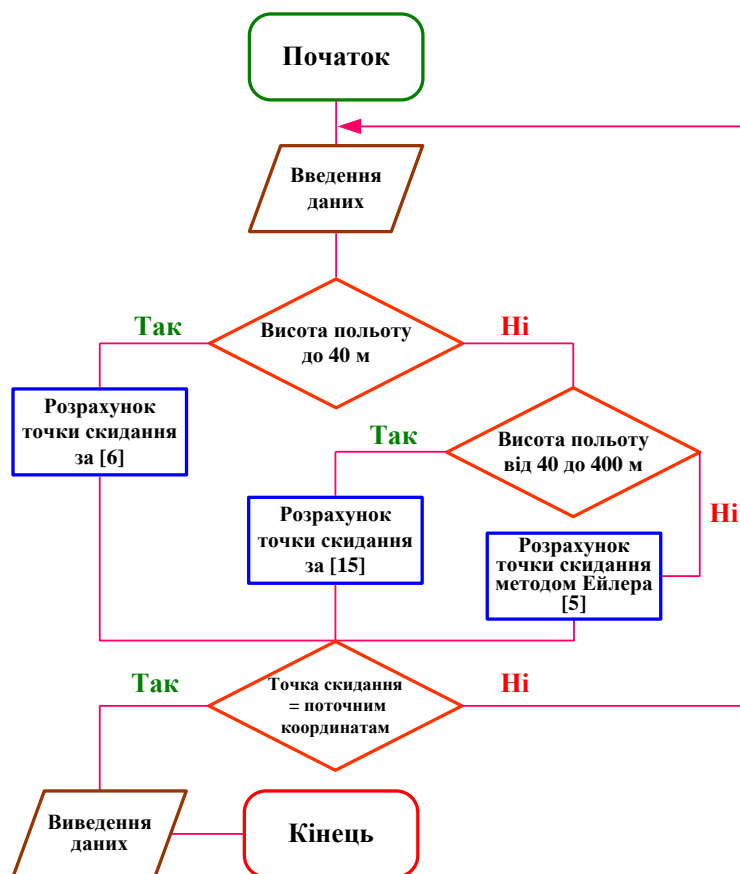


Рис. 1. Блок-схема загального алгоритму визначення точки скидання некерованого вантажу з БПЛА літакового типу

що висота скидання може задаватись у вигляді константи перед початком місії доставки. В цьому випадку автопілот виводить БПЛА на висоту скидання заздалегідь. Якщо ж висоту скидання не було задано перед початком місії доставки, то програмний модуль визначення точки скидання буде виконувати розрахунки на базі поточної висоти польоту. Подія скидання вантажу буде відбуватись, коли координати БПЛА будуть відповідати точці скидання.

Грунтуючись на отриманих даних (від датчиків та заданих перед початком місії) згідно рис. 1 відбувається перевірка висоти скидання. Залежно від висоти польоту застосовується один з трьох способів розрахунку точки скидання для висот до 40 м, від 40 до 400 м, та більше 400 м відносно поверхні землі.

### 2.2. Lua скрипт для визначення точки скидання некерованого вантажу з висот до 40 м відносно поверхні землі

Згідно робіт [8, 9] до висоти 40 м відносно поверхні землі рух вантажу можна описати рівняннями:

$$\begin{cases} x(t) = v_{0x} \cdot t, \\ y(t) = -\frac{gt^2}{2}, \\ z(t) = v_{0z} \cdot t, \\ t = \sqrt{\frac{2h}{g}}, \end{cases} \quad (1)$$

де,  $x$  – координата вантажу вздовж осі  $x$  в напрямку на ціль;  $y$  – координата вантажу в вертикальному напрямку відносно поверхні землі;  $z$  – координата вантажу вздовж осі  $z$  в боковому до курсу БПЛА напрямку;  $v_{0x}$  – повітряна швидкість вздовж осі  $x$  в момент скидання;  $v_{0z}$  – повітряна швидкість вздовж осі  $z$  в момент скидання;  $g$  – прискорення сили тяжіння;  $t$  – час руху;  $h$  – висота скидання вантажу. Видно, що дані рівняння не враховують силу опору повітря та наявність вітру.

Реалізує визначення точки скидання вантажу за рівняннями (1) функція програмного модуля на Lua, що представлена на рис. 2.

```

33
34 local function lowHeight()
35     print("low height")
36     local h = y0      -- висота
37     while h > 0 do
38         -- Оновлення часу
39         t0 = math.sqrt((2*h) / g)
40
41         displacement_x = Vx0 * t0
42         displacement_y = -1 * ((g * t0^2)/2)
43         displacement_z = Vz0 * t0
44
45         print(string.format("time: %0.1f  h: %4.1f  x(t): %6.3f  y(t): %7.3f  z(t):
46         %0.3f",
47                               t0, h, displacement_x, displacement_y, displacement_z))
48         -- Оновлення висоти
49         h = h - dh
50     end
end

```

Рис. 2. Функція на Lua для визначення точки скидання з висот до 40 м

### 2.3. Lua скрипт для визначення точки скидання некерованого вантажу з висот від 40 до 400 м відносно поверхні землі

Згідно роботі [15] для висот скидання від 40 до 400 м відносно поверхні землі рух вантажу можна описати рівняннями:

$$\begin{cases} v_{sx}(t) = \frac{v_{0x}}{1 + K_x v_{0x} t} + v_{wx}, \\ v_{sy}(t) = \left( -\sqrt{\frac{g}{K_y}} \right) \cdot \left( \frac{C - e^{-2\sqrt{gK_y} t}}{C + e^{-2\sqrt{gK_y} t}} \right) + v_{wy}, \\ v_{sz}(t) = \frac{v_{0z}}{1 + K_z v_{0z} t} + v_{wz} \end{cases} \quad (2)$$

та

$$\begin{cases} x(t) = \frac{1}{K_x} \cdot \ln(1 + K_x v_{0x} \cdot t) + v_{wx} t, \\ y(t) = v_{wy} t - \sqrt{\frac{g}{K_y}} \cdot t + \frac{1}{K_y} \cdot \ln\left( \frac{C+1}{C + e^{-2\sqrt{gK_y} t}} \right), \\ z(t) = \frac{1}{K_z} \cdot \ln(1 + K_z v_{0z} \cdot t) + v_{wz} t, \end{cases} \quad (3)$$

де  $C = \frac{\sqrt{\frac{g}{K_y}} - v_{0y}}{\sqrt{\frac{g}{K_y}} + v_{0y}}; \quad K_x = \frac{1}{2m} \cdot \xi_x \cdot \rho S_x;$

$$K_y = \frac{1}{2m} \cdot \xi_y \rho S_y; \quad K_z = \frac{1}{2m} \cdot \xi_z \rho S_z; \quad v_{sx}, \quad v_{sy},$$

$v_{sz}$  – шляхова швидкість вантажу вздовж відповідних осей координат;  $v_{wx}, v_{wy}, v_{wz}$  – швидкість вітру вздовж відповідних осей координат;  $v_{0x}, v_{0y}, v_{0z}$  – повітряна швидкість БПЛА в момент скидання;  $m$  – маса вантажу;  $\rho$  – густина повітря;  $\xi_x, \xi_y, \xi_z$  – коефіцієнт опору повітря вздовж відповідних осей координат;  $S_x, S_y, S_z$  – площа міделевого перерізу вантажу в напрямку відповідних осей координат.

```

51
52 local function mediumHeight()
53   print("medium height")
54   local Kx = (1 / (2 * mass)) * Cx * air_density * Sx
55   local Ky = (1 / (2 * mass)) * Cy * air_density * Sy
56   local Kz = (1 / (2 * mass)) * Cz * air_density * Sz
57
58   local sqrt_g_coefficient_y = math.sqrt(g / Ky)
59
60   local C = (sqrt_g_coefficient_y - Vy0) / (sqrt_g_coefficient_y + Vy0)
61   local extent_e = -2 * math.sqrt(g * Ky)
62
63   while t0 < 10.5 do
64
65     displacement_x = ((1 / Kx) * math.log(1 + Kx * Vx0 * t0)) + (additional_impact_x * t0)
66     displacement_y = (additional_impact_y * t0) - (sqrt_g_coefficient_y * t0) + (1 / Ky)
67                   * math.log((Cy + 1) / (Cy + math.exp(extent_e * t0)))
68     displacement_z = (1 / Kz) * math.log(1 + Kz * Vz0 * t0) + additional_impact_z * t0
69
70     print(string.format("time: %4.1f x(t): %7.3f y(t): %7.3f z(t): %6.3f",
71                       t0, displacement_x, displacement_y, displacement_z))
72     -- Оновлення часу
73     t0 = t0 + dt
74   end
75 --
76 end

```

Рис. 3. Функція на Lua для визначення точки скидання з висот від 40 до 400 м

## 2.4. Lua скрипт для визначення точки скидання некерованого вантажу з висот більше 400 м відносно поверхні землі

Використання чисельних методів (наприклад, методу Ейлера) для розрахунку положення вантажу у просторі є найбільш поширеним у наукових дослідженнях [4, 5].

Для розрахунку траєкторії за методом Ейлера необхідно розв'язати диференціальні рівняння для горизонтального та вертикального руху вантажу. Вважаємо, що  $x(t)$  та  $y(t)$  — координати вантажу у просторі,  $v_x(t)$  та  $v_y(t)$  — відповідно, горизонтальна та вертикальна швидкості:

$$\begin{cases} \frac{dx}{dt} = v_x(t), \\ \frac{dy}{dt} = v_y(t), \\ \frac{dv_x}{dt} = \frac{F_{\text{drag}_x}(t)}{m}, \\ \frac{dv_y}{dt} = \frac{F_{\text{drag}_y}(t) - mg}{m}, \end{cases} \quad (4)$$

Для визначення точки скидання з висот від 40 до 400 м на основі рівнянь (2) та (3) створено функцію на Lua, код якої представлено на рис. 3. Зазначимо, що рівняння (2) є додатковими і в прямому вигляді не використовуються у зазначеній функції. Проте питання визначення швидкості в момент падіння є важливим для місій доставки крихких вантажів, які повинні залишитися неушкодженими. Отже, рівняння (2) можуть бути використані у програмному модулі керування скиданням для визначення безпечної швидкості та висоти скидання крихких вантажів.

де  $F_{\text{drag}_x}, F_{\text{drag}_y}$  – сила опору повітря в напрямку відповідної осі координат;  $m$  – маса вантажу.

Метод Ейлера використовує кінцеві різниці для обчислення зміни положення і швидкості за невеликий проміжок часу  $\Delta t$ :

$$\begin{cases} x_{n+1} = x_n + \Delta t \cdot v_{x,n}, \\ y_{n+1} = y_n + \Delta t \cdot v_{y,n}, \\ v_{x,n+1} = v_{x,n} + \Delta t \cdot \frac{F_{\text{drag}_x,n}}{m}, \\ v_{y,n+1} = v_{y,n} + \Delta t \cdot \frac{F_{\text{drag}_y,n} - mg}{m}. \end{cases} \quad (5)$$

В (4) та (5) швидкість вантажа – це його шляхова швидкість з урахуванням впливу вітру. Якщо швидкість вітру характеризується експоненціальним профілем залежно від висоти, то, як показано у [15], це також можливо врахувати в рівняннях типу (5).

На рис. 4 наведено програмний блок на Lua на основі (4) та (5) для визначення точки скидання з висот більше 400 м.

Цей підхід дозволяє виконати симуляцію траєкторії вантажу з урахуванням основних сил та вітру, які впливають на його рух.

```

79 -- Функція для обчислення сили опору
80 local function calculateDrag(v)
81     return 0.5 * air_density * v^2 * Cx * Sx
82 end
83
84 local function highHeights()
85     print("high heights")
86     displacement_y = y0
87 -- Цикл для розрахунку траєкторії
88     for step = 1, steps do
89         -- Сили опору
90         local Fx_drag = calculateDrag(Vx0)
91         local Fy_drag = calculateDrag(Vy0)
92
93         -- Оновлення швидкостей
94         Vx0 = Vx0 - (Fx_drag / mass) * dt
95         Vy0 = Vy0 - (Fy_drag / mass + g) * dt
96
97         -- Оновлення положення
98         displacement_x = displacement_x + Vx0 * dt
99         displacement_y = displacement_y + Vy0 * dt
100
101         -- Зупинка, якщо вантаж досяг землі
102         if displacement_y <= 0 then
103             break
104         end
105         -- Вивід положення
106         print(string.format("Step: %2d, x: %5.2f, y: %6.2f",
107             step, displacement_x, displacement_y))
108     end
109 end
110

```

Рис. 4. Програмний код на Lua для визначення точки скидання з висот більше 400 м

## 2.5. Реалізація інтерфейсу для користувача

Інтерфейс потрібен для забезпечення можливості взаємодії користувача з програмним модулем визначення точки скидання вантажу. За його допомогою користувач може змінювати вхідні параметри та отримувати результати симуляції руху вантажу в зручному форматі. Зазначений інтерфейс має включати:

- введення вхідних параметрів, таких як вибір початкової швидкості, кута скидання, маси вантажу, коефіцієнтів опору, умов навколишнього середовища, тощо;
- перегляд і редагування графічних та чисельних результатів відображення траєкторії руху вантажу;
- збереження і завантаження конфігурацій, а саме зберігати набори параметрів та результатів для подальшого використання або аналізу.

Інтерфейс може бути реалізований як графічний або текстовий залежно від платформи та умов використання. На рис. 5 наведено приклад коду простого текстового інтерфейсу на Lua.

Залежно від потреб, інтерфейс користувача може бути вдосконалений:

- може бути забезпечений графічним інтерфейсом (GUI) з використанням бібліотеки для створення графічного інтерфейсу (наприклад, Love2D або Corona SDK), де користувач може взаємодіяти через кнопки, слайдери, графіки тощо;

- може бути забезпечений веб-інтерфейсом реалізованим з використанням HTML/CSS/JavaScript, що дозволить доступ до системи моделювання з будь-якого пристрою через браузер;

- може бути забезпечений функцією автоматизацією завдань з реалізацією додавання можливостей автоматизованого проведення серій розрахунків з різними параметрами та створення звітів.

Результати симуляції можуть бути представлені у вигляді текстових звітів, графіків, таблиць, або навіть 3D-візуалізацій. Це дозволить користувачу легко аналізувати дані та приймати рішення на основі отриманих даних.

Цей підхід забезпечить користувачам зручний доступ до інструментів симуляції траєкторії руху некерованого вантажу і підвищить ефективність використання програмного модуля визначення точки скидання в різних задачах.

## 3. Приклад реалізації

Виконання розробленого Lua скрипта було здійснено за допомогою програмного забезпечення Mission Planner від Ardupilot. Розрахунки часу падіння вантажу та горизонтального зміщення від точки скидання було реалізовано для висот 30, 300 та 1000 м (табл. 2) за умови ідентичності інших умов моделювання. Причому такі розрахунки для висот 30 та 300 м було виконано двома методами:

```

1 local mass = 10 -- маса вантажу, кг
2 local g = 9.81 -- прискорення вільного падіння
3 local air_density = 1.225 -- густина повітря, кг/м³
4 -- Початкові умови
5 local x0 = 0
6 local y0 = 445 -- м (початкове положення)
7
8 local Vx0 = 20 -- початкова швидкість по осі x
9 local Vy0 = 0 -- початкова швидкість по осі y
10 local Vz0 = 2 -- початкова швидкість по осі z
11
12 local Sx = 0.5 -- площа поперечного перерізу, м²
13 local Sy = 0.5 -- площа поперечного перерізу, м²
14 local Sz = 0.5 -- площа поперечного перерізу, м²
15
16 local Cx = 0.071 -- коефіцієнт лобового опору (приблизне значення для сфери)
17 local Cy = 0.071 -- коефіцієнт лобового опору (приблизне значення для сфери)
18 local Cz = 0.071 -- коефіцієнт лобового опору (приблизне значення для сфери)
19
20 local additional_impact_x = -15 -- додатковий вплив по осі x
21 local additional_impact_y = 0 -- додатковий вплив по осі y
22 local additional_impact_z = 0 -- додатковий вплив по осі z
23
24 local t0 = 0 -- початковий час
25 local dt = 0.5 -- Часовий крок в секундах
26 local dh = 1 -- крок переміщення
27 local steps = 200 -- кількість кроків
28
29
30 local displacement_x = 0
31 local displacement_y = 0
32 local displacement_z = 0

```

Рис. 5. Програмний код на Lua простого інтерфейсу користувача для модуля визначення точки скидання вантажу

Таблиця 2

Результат виконання Lua скрипта для визначення точки скидання вантажу

№ досліджу	Висота скидання 30 м		Висота скидання 300 м		Висота скидання 1000 м
	Середній час виконання Lua скрипта за спрощеними рівняннями (підрозділ 2.2), мс	Середній час виконання Lua скрипта за методом Ейлера (підрозділ 2.4), мс	Середній час виконання Lua скрипта за спрощеними рівняннями (підрозділ 2.3), мс	Середній час виконання Lua скрипта за методом Ейлера (підрозділ 2.4), мс	Середній час виконання Lua скрипта за методом Ейлера (підрозділ 2.4), мс
1	1,0	3,0	1,0	3,0	2,0
2	1,0	2,0	1,0	3,0	3,0
3	1,0	3,0	1,0	4,0	2,0
4	1,0	3,0	1,0	4,0	3,0
5	1,0	3,0	1,0	3,0	2,0
6	1,0	3,0	1,0	3,0	3,0
7	1,0	2,0	1,0	3,0	2,0
8	1,0	3,0	1,0	4,0	3,0
9	1,0	3,0	1,0	3,0	2,0
10	1,0	3,0	1,0	3,0	4,0
Усереднене значення часу виконання Lua скрипта, мс	1,0	2,8	1,0	3,3	2,6
Час падіння вантажу, с	2,5	2,0	7,8	6,5	-
Горизонтальне зміщення вантажу від точки скидання, м	49,50	37,92	156,40	113,11	-

за спрощеними виразами (підрозділи 2.2 і 2.3) та за методом Ейлера (підрозділ 2.4). Також було визначено час виконання Lua скрипта в середовищі Mission Planner. В даному програмному забезпеченні зазначений час вимірюється з точністю до 1 мс, що близько до часу виконання дослідженого Lua скрипта. Тому було виміряно час 25 послідовних виконань і потім визначено середнє значення роботи Lua скрипта. Такий дослід було проведено 10 разів з кожним дослідженим Lua скриптом. В результаті було отримано одне усереднене значення часу виконання кожного Lua скрипта. Як видно з таблиці 2, виконання Lua скрипта за спрощеними рівняннями швидше ніж розрахунки за методом Ейлера від 2,8 до 3,3 разів. До того ж, на різних висотах розрахунки за методом Ейлера тривають близько 3 мс. При цьому час падіння вантажу, отриманий за спрощеними рівняннями більше такого ж часу, отриманого методом Ейлера на 20 – 25%. Горизонтальне зміщення вантажу від точки скидання розраховане за спрощеними рівняннями більше за таке ж зміщення отримане методом Ейлера на 30 – 38 %.

#### 4. Обговорення

Результати з табл. 2 підтверджують зменшення приблизно у 3 рази часу розрахунку точки скидання вантажу при застосуванні Lua скрипта, який обчислює спрощені рівняння (підрозділи 2.2 і 2.3). За умови повітряної швидкості БПЛА близько 20 м/с це дасть можливість оновлювати дані щодо визначення точки скидання приблизно кожні 2 см шляху БПЛА, що менше у приблизно 3 рази порівняно з відповідними розрахунками за методом Ейлера. Це важливо для з точки зору збільшення точності визначення точки скидання вантажу згідно [5].

Той факт, що результати розрахунку точки скидання вантажу (див. табл. 2, час падіння вантажу та горизонтальне зміщення вантажу від точки скидання) за спрощеними рівняннями та методом Ейлера відрізняються до приблизно (30 – 40) % може бути обумовлений наступними причинами. Розмір кроку обчислень методом Ейлера визначає точність розрахунків та їх час. Отже вибір оптимального кроку обчислень для забезпечення найменшого часу розрахунків при максимальній їх точності є важливим, проте виходить за рамки цієї роботи. Також треба зазначити, що метод Ейлера характеризується методичною похибкою, яка пропорційна кроку обчислень в кінці інтервалу розрахунків і для результатів з табл. 2 може досягати 20 – 30 %. До того ж, розрахунки за спрощеними рівняннями з робіт [6, 15] мають методичну похибку, проте, як впливає з [6, 15], такий підхід все ж здатен забезпечити точність доставки вантажу  $\pm 5$  м.

Таким чином, результати табл. 2 свідчать про те, що ефективність розробленого на LUA програмного забезпечення для визначення точки скидання некерованого вантажу з висот до 400 м перевищує ефективність відомих аналогів [5] у приблизно 3 рази. При цьому оцінювана точність доставки вантажу відповідає інтервалу  $\pm 5$  м.

#### 5. Висновки

В роботі на Lua розроблено програмний модуль для визначення точки скидання некерованого вантажу з БПЛА літакового типу із системою керування ArduPilot. При цьому вперше застосовано різний підхід до розрахунку точки скидання на різних висотах відносно поверхні землі. В інтервалах малих висот даний модуль розраховує точку скидання за спрощеними аналітичними співвідношенням, а за висот більше 400 м – застосовується чисельне рішення диференціальних рівнянь руху вантажу. Це забезпечує більшу ефективність даного програмного забезпечення порівняно з [4, 5], а також забезпечує більший інтервал можливих висот скидання при помилці доставки до  $\pm 5$  м порівняно з роботами [6, 15].

Майбутні дослідження будуть присвячені експериментальному визначенню точності доставки вантажу за допомогою БПЛА літакового типу обладнаного розробленим на Lua програмним модулем для визначення точки скидання.

**Внесок авторів:** формулювання проблеми, мети та задач дослідження, обговорення, висновки – **А. М. Гуменний, С. В. Олійник**; аналіз літературних джерел – **О. Д. Підлісний**; розробка комп'ютерної моделі руху некерованого вантажу, приклад реалізації, написання рукопису роботи – **П. П. Малашга**; розробка загального алгоритму роботи програмного модуля, написання рукопису роботи – **В. В. Алейніков**.

#### Конфлікт інтересів

Автори заявляють, що немає конфлікту інтересів щодо матеріалів цієї публікації, фінансового, особистого, авторського чи іншого, який міг би вплинути на дослідження та його результати, представлені в цій статті.

#### Фінансування

Дослідження проводилося без фінансової підтримки.

#### Доступність даних

Рукопис не має пов'язаних даних.



### Використання засобів штучного інтелекту

Автори підтверджують, що не використовували технології штучного інтелекту при створенні представленої роботи.

Усі автори прочитали та погодились з опублікованою версією рукопису.

### Література

1. Alghamdi, Y. Architecture, classification, and applications of contemporary unmanned aerial vehicles [Text] / Y. Alghamdi, A. Munir, & H. Manh La // *IEEE Consumer Electronics Magazine*. – 2021. – Vol. 10, no. 6. – P. 9-20. DOI: 10.1109/MCE.2021.3063945.

2. Peksa, J. A review on the state of the art in copter drones and flight control systems [Text] / J. Peksa, & D. Mamchur // *Sensors*. – 2024. – Vol. 24, no. 11. – Article No. 3349. DOI: 10.3390/s24113349.

3. Hassanalian, M. Classifications, applications, and design challenges of drones: A review [Text] / M. Hassanalian, & A. Abdelkefi // *Progress in Aerospace sciences*. – 2017. – Vol. 91. – P. 99-131. DOI: 10.1016/j.paerosci.2017.04.003.

4. Beard, R. W. Small unmanned aircraft: Theory and practice [Text] / R. W. Beard, & T. W. McLain. – Princeton and Oxford: Princeton university press, 2012. – 320 p. – Available at: [https://github.com/byu-magicc/mavsim\\_public](https://github.com/byu-magicc/mavsim_public) (accessed: 01.08.2024).

5. Mathisen, S. H. Approach methods for autonomous precision aerial drop from a small unmanned aerial vehicle [Text] / S. H. Mathisen, V. Grindheim, & T. A. Johansen // *IFAC-PapersOnLine*. – 2017. – Vol. 50, no. 1. – P. 3566-3573. DOI: 10.1016/j.ifacol.2017.08.624.

6. Ivanovic, A. Parabolic airdrop trajectory planning for multicopter unmanned aerial vehicles [Text] / A. Ivanovic, & M. Orsag // *IEEE Access*. – 2022. – Vol. 10. – P. 36907-36923. DOI: 10.1109/ACCESS.2022.3164434.

7. Development of unmanned aerial vehicle (UAV) for dropping object accurately based on global positioning system [Text] / R. Mardiyanto, M. Pujiantara, H. Suryoatmojo, R. Dikairono, & A. N. Irfansyah // *International Seminar on Intelligent Technology and Its Applications (ISITIA)*. – IEEE, 2019. – P. 86-90. DOI: 10.1109/ISITIA.2019.8937269.

8. Ardupilot [Electronic resource]. – Available at: <https://ardupilot.org/> (accessed: 01.10.2024)

9. Civelek, B. A review on the precision guided airdrop systems [Electronic resource] / B. Civelek, & S. Kivrak // *International Journal of Latest Technology in Engineering, Management & Applied Science (IJLTEMAS)*. – 2019. – Vol. VIII, no. 1. – P. 13 – 17. – Available at:

<https://www.ijltemas.in/DigitalLibrary/Vol.8Issue1/13-17.pdf> (accessed: 01.08.2024).

10. Огляд і аналіз сучасних керованих систем точного повітряного десантування вантажів [Текст] / О. О. Вамболь, І. В. Калужинів, К. П. Баранов, А. В. Кондратьєв // *Космічна наука і технологія*. – 2023. – Т. 29, № 5. – С. 33-50. DOI: 10.15407/knit2023.05.033.

11. Wright, R. Precision Airdrop System [Text] / R. Wright, R. Benney, & J. McHugh // *Proceedings of the 18th AIAA Aerodynamic Decelerator Systems Technology Conference and Seminar, 23 - 26 May 2005 / AIAA*. – Munich, Germany, 2005. – Article No. 1644. DOI: 10.2514/6.2005-1644.

12. Automated air drop system for search and rescue applications utilizing unmanned aircraft systems [Text] / D. Boura, K. Strang, W. Semke, R. Schultz, & D. Hajicek // *Proceedings of the Infotech@Aerospace, 29 – 31 March 2011 / AIAA*. – St. Louis, Missouri, 2011. – Article No. 1528. DOI: 10.2514/6.2011-1528.

13. Autonomous ballistic airdrop of objects from a small fixed-wing unmanned aerial vehicle [Text] / S. G. Mathisen, F. S. Leira, H. H. Helgesen, K. Gryte, & T. A. Johansen // *Autonomous Robots*. – 2020. – Vol. 44. – P. 859-875. DOI: 10.1007/s10514-020-09902-3.

14. Development of a cargo airdrop modeling method for a tactical blended-wing-body UAV [Electronic resource] / D. Mitridis, N. Mathioudakis, P. Panagiotou, & K. Yakinthos // *Proceedings of the IOP Conference Series: Materials Science and Engineering*. – IOP Publishing, 2021. – Vol. 1024, no. 1. – Article No. 012044. DOI: 10.1088/1757-899X/1024/1/012044.

15. Construction of a ballistic model of the motion of uncontrolled cargo during its autonomous high precision drop from a fixed-wing unmanned aerial vehicle [Text] / A. Humennyi, S. Oleynick, P. Malashta, O. Pidlisnyi, & V. Aleinikov // *Eastern-European Journal of Enterprise Technologies*. – 2024. – Vol. 5, no. 7 (131). – P. 25-33. DOI: 10.15587/1729-4061.2024.313102.

### References

1. Alghamdi, Y., Munir, A., & Manh La, H. Architecture, classification, and applications of contemporary unmanned aerial vehicles. *IEEE Consumer Electronics Magazine*, 2020, vol. 10, no. 6, pp. 9-20. Available at: 10.1109/MCE.2021.3063945 (accessed: 01.10.2024).

2. Peksa, J., & Mamchur, D. A review on the state of the art in copter drones and flight control systems. *Sensors*, 2024, vol. 24, no. 11, article no. 3349. DOI: 10.3390/s24113349.

3. Hassanalian, M., & Abdelkefi, A. Classifications, applications, and design challenges of drones: A review. *Progress in Aerospace sciences*, 2017, vol. 91, pp. 99-131. DOI: 10.1016/j.paerosci.2017.04.003.

4. Beard, R. W., McLain, T. W. *Small unmanned aircraft: Theory and practice*. Princeton and Oxford, Princeton university press Publ., 2012. 320 p. Available at: [https://github.com/byu-magicc/mavsim\\_public](https://github.com/byu-magicc/mavsim_public) (accessed:01.08.2024).
5. Mathisen, S. H., Grindheim, V., & Johansen, T. A. Approach methods for autonomous precision aerial drop from a small unmanned aerial vehicle. *IFAC-PapersOnLine*, 2017, vol. 50, no. 1, pp. 3566-3573. DOI: 10.1016/j.ifacol.2017.08.624.
6. Ivanovic, A., & Orsag, M. Parabolic airdrop trajectory planning for multirotor unmanned aerial vehicles. *IEEE Access*, 2022, vol. 10, pp. 36907-36923. DOI: 10.1109/ACCESS.2022.3164434.
7. Mardiyanto, R., Pujiantara, M., Suryoatmojo, H., Dikairono, R., & Irfansyah, A. N. Development of unmanned aerial vehicle (UAV) for dropping object accurately based on global positioning system. *International Seminar on Intelligent Technology and Its Applications (ISITIA)*, 2019, pp. 86-90. DOI: 10.1109/ISITIA.2019.8937269.
8. *Ardupilot*. Available at: <https://ardupilot.org/> (accessed 01.10.2024).
9. Civelek, B., & Kivrak, S. A review on the precision guided airdrop systems. *International Journal of Latest Technology in Engineering, Management & Applied Science (IJLTEMAS)*, 2019, vol. VIII, no. I, pp. 13-17. Available at: <https://www.ijltemas.in/DigitalLibrary/Vol.8Issue1/13-17.pdf> (accessed:01.08.2024).
10. Vambol, O., Kaluzhynov, I., Barakhov, K., & Kondratiev, A. Ohlyad i analiz suchasnykh kerovanykh systemtochnoho povitryanoho desantuvannya vantazhiv [A review and analysis of existing guided precision airdrop systems]. *Cosmic Science and Technology*, 2023, vol. 29, no. 5, pp. 33-50. DOI: 10.15407/knit2023.05.033.
11. Wright, R., Benney, R., & McHugh, J. Precision Airdrop System. *Proceedings of the 18th AIAA Aerodynamic Decelerator Systems Technology Conference and Seminar*, Munich, Germany, AIAA, 2005, Article No. 1644. DOI: 10.2514/6.2005-1644.
12. Boura, D., Strang, K., Semke, W., Schultz, R., Hajicek, D. Automated air drop system for search and rescue applications utilizing unmanned aircraft systems. *Proceedings of the Infotech@Aerospace*, St. Louis, Missouri, AIAA, 2011, article no. 1528. DOI: 10.2514/6.2011-1528.
13. Mathisen, S. G., Leira, F. S., Helgesen, H. H., Gryte, K., & Johansen, T. A. Autonomous ballistic airdrop of objects from a small fixed-wing unmanned aerial vehicle. *Autonomous Robots*, 2020, vol. 44, pp. 859-875. DOI: 10.1007/s10514-020-09902-3.
14. Mitridis, D., Mathioudakis, N., Panagiotou, P., & Yakinthos, K. Development of a cargo airdrop modeling method for a tactical blended-wing-body UAV. *Proceedings of the IOP Conference Series: Materials Science and Engineering*, IOP Publishing, 2021, vol. 1024, no. 1, article no. 012044. DOI: 10.1088/1757-899X/1024/1/012044.
15. Humennyi, A., Oleynick, S., Malashta, P., Pidlisnyi, O., & Aleinikov, V. Construction of a ballistic model of the motion of uncontrolled cargo during its autonomous highprecision drop from a fixed-wing unmanned aerial vehicle. *Eastern-European Journal of Enterprise Technologies*, 2024, vol. 5, no. 7 (131), pp. 25-33. DOI: 10.15587/1729-4061.2024.313102.

Надійшла до редакції 06.08.2024, розглянута на редколегії 18.11.2024

#### DEVELOPMENT ON LUA OF A SOFTWARE MODULE FOR DETERMINING THE DROP POINT OF UNCONTROLLED CARGO FROM UAV OF AIRCRAFT TYPE WITH ARDUPILOT CONTROL SYSTEM

*Andrii Humennyi, Pavlo Malashta, Serhii Oliynick, Oleksandr Pidlisnyi, Vitalii Aleinikov*

Cargo delivery is one of the most popular areas of application of aircraft-type UAVs with the Ardupilot control system. Autonomous missions performed by such UAVs are associated with the need to determine the point of discharge of unguided cargo, which significantly affects the accuracy of its delivery. However, Ardupilot is not equipped with a software module for determining the point of discharge of unguided cargo, and such additional software is recommended to be developed in Lua. Known software modules for determining the point of discharge of UAVs, as a rule, perform a cyclic solution of a complex system of differential equations of cargo motion, which requires significant hardware resources. However, for heights up to 400 m relative to the ground surface, it is possible to use simplified analytical expressions of cargo motion with a delivery accuracy of  $\pm 5$  m to increase the efficiency of the specified software module. Therefore, the **object of research** is the process of determining the point of discharge of unguided cargo from an aircraft-type UAV using a Lua script. The **subject of research** is a computer model of the movement of unguided cargo in Lua, on the basis of which the point of discharge is determined. The **aim of the work** is to develop a software module in Lua to determine the point of dropping an unguided cargo from an aircraft-type UAV with the ArduPilot control system by using simplified equations of motion of the cargo at altitudes up to 400 m.

The **following results were obtained:** a general algorithm for the operation of the software module for determining the point of dropping an unguided cargo was developed, a Lua script was developed to determine the point of dropping from heights up to 40 m, a Lua script was developed to determine the point of dropping from heights from 40 to 400 m relative to the ground surface, a Lua script was provided to determine the point of dropping from heights above 400 m relative to the ground surface, and the user interface of the software module was implemented. **Conclusion:** a software module was developed in Lua to determine the point of dropping an unguided cargo from an aircraft-type UAV with the Ardupilot control system. A feature of this software is the use of different approaches to determining the point of dropping in the intervals of heights up to 40 m, from 40 to 400 m and more than 400 m relative to the ground surface. This makes this software module more effective at low altitudes.

**Keywords:** unmanned aerial vehicle; UAV; cargo drop; control system; drop algorithm; software module; drop point.

**Гуменний Андрій Михайлович** – канд. техн. наук, доц., проректор з НПР, Національний аерокосмічний університет ім. М. С. Жуковського «Харківський авіаційний інститут», Харків, Україна.

**Малашта Павло Петрович** – асп. каф. 501, Національний аерокосмічний університет ім. М.С. Жуковського «Харківський авіаційний інститут», Харків, Україна.

**Олійник Сергій Володимирович** – д-р техн. наук, доц. каф. 505, Національний аерокосмічний університет ім. М.С. Жуковського «Харківський авіаційний інститут», Харків, Україна.

**Підлісний Олександр Дмитрович** – наук. співроб. науково-дослідної лабораторії факультету автоматизованих систем управління та наземного забезпечення польотів авіації, Харківський національний університет Повітряних Сил імені Івана Кожедуба, Харків, Україна.

**Алейніков Віталій Валерійович** – асп. каф. 103, Національний аерокосмічний університет ім. М.С. Жуковського «Харківський авіаційний інститут», Харків, Україна.

**Andrii Humennyi** – Candidate of Technical Sciences, Associate Professor, Vice-Rector for Research and Teaching, National Aerospace University "Kharkiv Aviation Institute", Kharkiv, Ukraine, e-mail: a.gumennyu@khai.edu, ORCID: 0000-0003-1020-6304, Scopus Author ID: 57219051542.

**Pavlo Malashta** – PhD Student of Department of 501, National Aerospace University "Kharkiv Aviation Institute", Kharkiv, Ukraine, e-mail: p.p.malashta@khai.edu, ORCID: 0009-0000-2652-1507.

**Serhii Oliynick** – Doctor of Technical Sciences, Associate Professor of Department of 505, National Aerospace University "Kharkiv Aviation Institute", Kharkiv, Ukraine, e-mail: sergey.olynick@khai.edu, ORCID: 0000-0001-6073-8531, Scopus Author ID: 49864295800.

**Oleksandr Pidlisnyi** – Scientific Associate of Research Laboratory at the Faculty of Automated Control System and Ground Support of Aviation Flights, Ivan Kozhedub Kharkiv National Air Force University, Kharkiv, Ukraine, e-mail: apodlesny73@gmail.com, ORCID: 0000-0002-7805-8232.

**Vitalii Aleinikov** – PhD Student of Department of 103, National Aerospace University "Kharkiv Aviation Institute", Kharkiv, Ukraine, e-mail: aleinikov.v19@gmail.com, ORCID: 0009-0006-4109-4375.