



ВІЗУАЛІЗАЦІЯ ТА ІНТЕРАКТИВНІ МУЛЬТИМЕДІЙНІ ТЕХНОЛОГІЇ

VISUALIZATION AND INTERACTIVE MULTIMEDIA TECHNOLOGIES

УДК 004.771:004.41

DOI: 10.31866/2617-796X.8.2.2025.347946

Олександр Ткаченко,

кандидат фізико-математичних наук, доцент,
доцент кафедри програмного забезпечення комп'ютерних систем,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»,
доцент кафедри інформаційних технологій,
Навчально-науковий інститут управління,
технологій та правових наук,
Національний транспортний університет,
Київ, Україна
aatokg@gmail.com
<https://orcid.org/0000-0001-6911-2770>

Микита Гребенюк,

магістрант кафедри інформаційних технологій,
Навчально-науковий інститут управління,
технологій та правових наук,
Національний транспортний університет,
Київ, Україна
4ekn14e@gmail.com
<http://orcid.org/0009-0009-9610-7518>

ДЕЯКІ АСПЕКТИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ВИЗНАЧЕННЯ КООРДИНАТ ПІД ЧАС АВТОНОМНОГО ПОЛЬОТУ БПЛА

На сьогодні розширюються сфери застосування безпілотних літальних апаратів, зокрема в логістиці, високоточній агрономії, обороні, моніторингових завданнях. Усе це актуалізує проблему забезпечення їхньої автономності та високої навігаційної точності. Аналіз стану сучасного ринку програмного забезпечення для автономного керування безпілотними літальними апаратами свідчить про недостатню кількість потрібних програмних рішень, що ускладнює вибір найбільш ефективного програмного комплексу.

Метою статті є дослідження сучасних програмних засобів, призначених для визначення координат та забезпечення навігації та локалізації безпілотних літальних апаратів в автономному режимі, функціональних і технічних можливостей цих апаратів.

Методами дослідження є методи порівняльного аналізу основних програмних рішень цієї предметної області (безпілотні літальні апарати). У статті розглянуто підходи до розроблення та функціонування програмного рішення для визначення координат під час автономного польоту таких апаратів на основі використання розширеного фільтра Калмана, який сприяє злиттю даних від інерціальних, візуальних і супутникових систем.

Новизною проведеного дослідження є аналіз сучасних систем підтримання ефективних автономних польотів безпілотних літальних апаратів, порівняльний аналіз новітніх Open Source платформ PX4 та ArduPilot з урахуванням оцінки їх інтеграційних можливостей для зовнішніх SLAM / VIO-модулів та розроблення гібридного багатосенсорного комплексу на базі Robot Operating System, що використовує фільтри Калмана для компенсації помилок інерціальних навігаційних систем даними візуальної одометрії та GNSS.

Висновки. У роботі виконано узагальнення фундаментальних методів визначення координат. Обґрунтовано необхідність Sensor Fusion для мінімізації накопичувальної похибки та забезпечення стійкості до відмов, що стало основою для подальшої програмної реалізації; проведено аналіз ринку програмного забезпечення для БПЛА; здійснено класифікацію сучасних програмних засобів за архітектурним принципом, виділивши дихотомію між відкритими платформами (ArduPilot, PX4) та комерційними екосистемами. Установлено, що відкриті платформи надають більш високу гнучкість, необхідну для імплементації нових алгоритмів SLAM, а комерційні рішення пропонують високий ступінь інтеграції та відповідності нормативним вимогам; для реалізації надійного визначення координат у закритих та GNSS відсутніх середовищах розроблено програмний комплекс на базі Robot Operating System (це забезпечило модульний підхід та легкість масштабування).

Ключові слова: безпілотний літальний апарат; автономний політ; визначення координат; програмне забезпечення для БПЛА; GNSS; фільтр Калмана.

Вступ. Збільшення кількості та видів безпілотних літальних апаратів (БПЛА), розширення сфер їх використання (від високоточної агрономії та моніторингу інфраструктури до логістики й оборонних завдань) – усе це робить актуальною проблемою розробку відповідного програмного забезпечення, яке обумовлює та підтримує не тільки їхню операційну стійкість, а й автономність функціонування та високу навігаційну точність.

Надійність відповідного програмного забезпечення та мінімізація ризиків втрати самого апарату напряму залежать від якості, швидкості обробки даних і робастності самого програмного забезпечення, яке дає змогу (Zhao et al., 2020; Dai et al., 2020):

- отримувати точно визначені координати об'єкта (наприклад, об'єкта спостереження чи об'єкта, до якого застосовуються відповідні дії з боку БПЛА);
- керувати траєкторією польоту БПЛА.

Незважаючи на велику кількість інновацій у сфері аеронавігаційних технологій, відповідного програмного забезпечення для автономного польоту БПЛА ще недостатньо для того, щоб забезпечити всі потреби користувачів та виробників таких апаратів.

На сьогодні є багато відкритих (Open Source) платформ (наприклад, таких як PX4 та ArduPilot (PX4 Autopilot User Guide, n.d.; What is ArduPilot, n.d.)), що мають суттєво різні концепції та підходи, зокрема, до:

- алгоритмів локалізації БПЛА;
- архітектури відповідного програмного забезпечення;

- вимог до сенсорних систем, які використовуються в БПЛА.

Така різниця програмних рішень значно ускладнює процес об'єктивної валідації та критичного відбору найбільш адаптивного програмного забезпечення для конкретних експлуатаційних умов і технічних завдань.

Тому проблема систематизованого критичного аналізу наявного програмного забезпечення БПЛА є актуальною та потребує свого вирішення для формування єдиного системного підходу до розроблення як безпосередньо самого програмного забезпечення БПЛА, так і супутніх систем і апаратного забезпечення (пов'язаного, наприклад, із використанням віртуальної реальності (VR) та доповненої реальності (AR)).

Мета й завдання дослідження. Метою є дослідження сучасних програмних засобів, призначених для визначення координат та забезпечення навігації та локалізації БПЛА в автономному режимі, функціональних і технічних можливостей таких апаратів.

Досягнення мети передбачає виконання таких завдань:

- визначити основні проблеми наявного програмного забезпечення, що використовується в сучасних БПЛА;
- провести аналіз алгоритмів визначення координат, що застосовуються в системах автономного керування БПЛА;
- провести класифікацію та систематизацію наявних програмних засобів, призначених для визначення координат та забезпечення навігації та локалізації БПЛА в автономному режимі;
- провести аналіз наявного програмного забезпечення, що підтримує функціонування БПЛА в автономному режимі, зокрема за критеріями точності локалізації, апаратного забезпечення та інтеграції з іншими програмними системами;
- визначити рекомендації щодо вибору оптимальних програмних рішень визначення координат і забезпечення навігації та локалізації БПЛА в автономному режимі.

Результати дослідження. Автономний політ БПЛА є процесом, що потребує надійного функціонування трьох взаємопов'язаних підсистем (відповідного програмного комплексу й апаратного забезпечення), які гарантують процеси:

- навігації;
- управління;
- планування місії польоту.

Основним завданням навігації є точне визначення просторового положення, орієнтації та швидкості об'єкта щодо обраної системи координат.

Точність визначення місцезнаходження БПЛА (його локалізації) є критично важливою, бо вона здійснює безпосередній вплив на здатність БПЛА:

- уникати помилок (під час руху та здійснення операцій щодо цілей польоту);
- виконувати складні маневри й успішно реалізовувати місії, особливо за умов підтримання польоту, що супроводжується небезпекою (наприклад, для самого апарату) чи високим рівнем ризиків, обумовлених несприятливими погодними чи іншими факторами.

Для опису положення БПЛА під час польоту використовують такі основні групи систем координат:

1. *Глобальні системи (World Frame)* (Zhao et al., 2020). Найбільш поширеною з таких систем на сьогодні є WGS-84 (Всесвітня геодезична система), яку використовують для супутникової навігації (наприклад, GPS, GNSS). WGS-84 забезпечує єдину, загальносвітову систему відліку, але її використання має певні обмеження, бо вона є ефективною лише на відкритих просторах.

2. *Локальні системи (Body Frame/Local Navigation Frame)* (Qin, Li and Shen, 2018). Такі системи жорстко зв'язані із самим апаратом, зокрема це може бути система його власних осей, або з обмеженою ділянкою місцевості. Локальні системи застосовуються для деталізованого контролю та моніторингу польоту і ґрунтуються на алгоритмах візуальної одометрії та SLAM (Cadena et al., 2016), де позиція визначається відносно оточення (об'єкта, який є ціллю польоту, самого БПЛА або точки старту), а не відносно супутникових даних.

Для забезпечення надійної автономії сучасні програмні засоби інтегрують декілька різних сенсорних технологій, кожна з яких має свої переваги та принципові обмеження (Sabzevari and Chatraei, 2021).

GNSS (*Global Navigation Satellite System*, Глобальні навігаційні супутникові системи) (Farrell and Barth, 1999), найбільш відомою з яких є GPS (*Global Positioning System*, Глобальна система позиціонування), залишаються основою глобальної навігації.

Принцип дії GPS полягає в тому, що система визначає координати БПЛА на основі трилатерації (обчислення відстані до мінімум чотирьох супутників через вимірювання різниці в часі отримання від них відповідних інформаційних сигналів). Перевагами GPS є її висока точність у відкритих просторах і глобальне покриття території (країни, регіону тощо). Крім очевидних переваг GPS має й певні недоліки та обмеження використання. Зокрема, система дуже залежить від зовнішнього сигналу, є вразливою до блокування, перешкод (*jamming*) (Zhao et al., 2020), спотворення (*spoofing*) (Dai et al., 2020) і втрати сигналу в так званих міських каньйонах або під густим лісовим покривом. Це обумовлює необхідність її компенсації іншими автономними методами.

3. *Інерціальні навігаційні системи* використовують внутрішні датчики (акселерометри та гіроскопи) для вимірювання прискорення і кутових швидкостей (Farrell and Barth, 1999). Принцип дії таких систем ґрунтується на тому, що на основі результатів відповідних вимірювань обчислюються зміни положення, швидкості та орієнтації БПЛА в просторі.

Перевагами інерціальних навігаційних систем є, зокрема, їх незалежність від зовнішніх сигналів (GPS, радіо) та висока частота оновлення даних, необхідних для позиціонування апарата. Основною проблемою інерціальних навігаційних систем є накопичення згодом похибок (так звана проблема дрейфу). Похибки вимірювання акселерометрів та гіроскопів інтегруються, що призводить до значного відхилення обчисленого положення БПЛА від істинного в довготривалих місіях. Це обумовлює рідке використання інерціальних навігаційних систем самостійно і потребує регулярної корекції за допомогою інших сенсорів (Grewal, Andrews and Bartone, 2020).

Методи сенсорної одометрії (візуальної та лазерної) (Cadena et al., 2016) базуються на використанні даних, отриманих із сенсорів оточення, для обчислення параметрів руху БПЛА (табл. 1) відносно його безпосереднього місцезорешування.

Кожен з визначених у табл. 1 методів є ефективним в умовах обмеженого або відсутнього GPS-сигналу (Dai et al., 2020) (наприклад, усередині приміщень чи лісах), що робить їх ключовими компонентами автономної навігації.

Таблиця 1

Порівняння методів обчислення руху БПЛА

Метод	Сенсор	Принцип дії	Недоліки
Візуальна одометрія (<i>Visual Odometry</i>) (Cadena et al., 2016),	Камери (моно / стерео / RGB-D)	Аналіз послідовних кадрів для обчислення зміни положення та орієнтації (<i>pose estimation</i>).	Високі обчислювальні вимоги, залежність від якості освітлення та текстури оточення.
LiDAR (<i>Light Detection and Ranging</i>) (Zhang and Singh, 2014),	Лазерний сканер	Використання лазерних імпульсів для вимірювання відстаней і створення високоточної 3D-хмари точок.	Висока вартість апаратного забезпечення та значна потреба в обчислювальних ресурсах для оброблення 3D-даних.

На рис. 1 продемонстровано деякі приклади роботи візуальної одометрії.

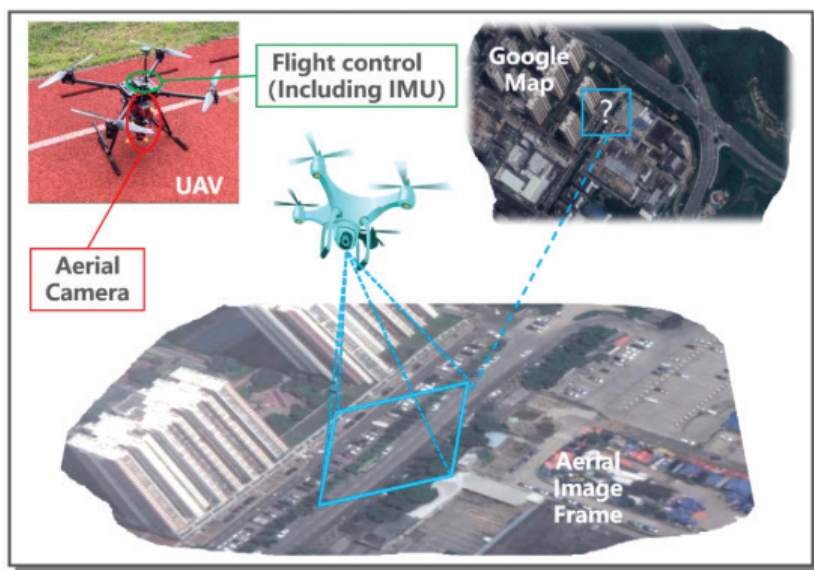


Рис. 1. Приклад роботи візуальної одометрії
Джерело: (Qin, Li and Shen, 2018)

Порівняльний аналіз наявних методів свідчить про те, що жоден метод визначення координат не може в повною мірою забезпечити необхідну надійність і стійкість до відмов під час здійснення автономного польоту БПЛА (Zhao et al., 2020).

Надійність можна досягнути за допомогою багатосенсорної інтеграції (*Sensor Fusion*), яка передбачає комбінування даних від різних типів сенсорів (GNSS, ІНС, VO, LiDAR тощо) для отримання єдиної, найбільш точної та стійкої оцінки стану БПЛА (Lucic et al., 2023). У процесі програмної реалізації багатосенсорної інтеграції та математичного моделювання (*Sensor Fusion*) важливу роль відіграють рекурсивні фільтри оцінки стану БПЛА (Grewal, Andrews and Bartone, 2020).

Фільтр Калмана (*Kalman Filter*) є оптимальним рекурсивним алгоритмом і слугує математичною моделлю для інтеграції навігаційних даних щодо позиціонування та руху БПЛА (Lin, 2008; Grewal, Andrews and Bartone, 2020). Процес оцінки стану БПЛА (при польоті, русі, місцерозташуванні) розбивається на:

- прогноз (*Prediction*): фільтр Калмана використовує модель динаміки БПЛА, яку можна отримати, наприклад, з даних інерціальних навігаційних систем, для прогнозування його наступного стану (з відповідними похибками);
- корекцію (*Update*): фільтр Калмана отримує нові результати вимірювання (наприклад, від GNSS або VO) і коригує свій прогноз, ураховуючи різницю між прогнозованим і вимірянним станом БПЛА та статистичну надійність кожного сенсора (Sabzevari and Chatraei, 2021).

Динамічні та сенсорні моделі БПЛА є нелінійними, тому на практиці частіше використовується розширений фільтр Калмана (*Extended Kalman Filter*) (Sabzevari and Chatraei, 2021). Цей фільтр є стандартом для інтеграції GNSS, інерціальних навігаційних систем ІНС та візуальної / лазерної одометрії, бо сприяє створенню єдиної моделі помилок і забезпечує високу точність навігації (Lucic et al., 2023).

SLAM (*Simultaneous Localization and Mapping*) є найбільш перспективним підходом для повноцінної автономії БПЛА й вирішує одночасно дві проблеми: визначення положення (*Localization*) та створення карти оточення (*Mapping*) (Mur-Artal, Montiel and Tardós, 2015).

Системи SLAM (наприклад, Visual-SLAM (Mur-Artal, Montiel and Tardós, 2015), LiDAR-SLAM (Zhang and Singh, 2014)) використовують сенсорні дані для створення моделі навколишнього середовища, одночасно оцінюючи власне положення БПЛА в цій моделі. Серед переваг SLAM можна відмітити, зокрема, такі:

- *універсальність* – здатність функціонувати в невідомих, динамічних чи закритих середовищах, де зовнішні сигнали (GPS) недоступні;
- *корекція за контуром* (*Loop Closure*): на відміну від одометрії, SLAM вміє розпізнавати вже відвідані місця та коригувати накопичену похибку, зменшуючи дрейф (Qin, Li and Shen, 2018).

Слід визнати, що, маючи вказані переваги, SLAM висуває високі вимоги до програмних і апаратних ресурсів БПЛА, зокрема такі, як:

- *обчислювальна інтенсивність*, що передбачає обробку в реальному часі великих обсягів сенсорних даних, яка потребує значних обчислювальних потужностей, що стає проблемою для компактних БПЛА;
- програмна реалізація таких систем, що часто базується на спеціалізованих фреймворках (наприклад, таких як Robot Operating System (ROS) (Macenski et al., 2022), який надає бібліотеки для роботи з сенсорами (наприклад, OpenCV для VO

та PCL для LiDAR)) та інтеграції алгоритмів, охоплюючи фільтри Калмана (Bai, Wen and Hsu, 2020; La Vallee, 2018).

Програмне забезпечення, що відповідає за автономну навігацію БПЛА, є важливим елементом програмного комплексу БПЛА, визначаючи функціональність, робастність та точність усього авіаційного комплексу (Zhao et al., 2020).

Програмні засоби для автономного польоту поділяються на декілька архітектурних рівнів, які взаємодіють через стандартизовані протоколи, (наприклад, MAVLink (Meier, Honegger and Pollefeys, 2015)). Серед цих рівнів слід виділити такі:

- прошивки автопілота (*Flight Stacks*) – низькорівневе програмне забезпечення, що функціонує на контролері польоту, реалізує алгоритми визначення координат (такі як фільтр Калмана, SLAM) та керує виконавчими механізмами (PX4 Autopilot User Guide, n.d.; What is ArduPilot, n.d.);

- програмні засоби локалізації (*SLAM / VO Modules*) – спеціалізовані бібліотеки (наприклад, VINS-Fusion (Qin, Li and Shen, 2018), ORB-SLAM (Mur-Artal, Montiel and Tardós, 2015)), які інтегруються з прошивкою автопілота для навігації при відсутності GNSS-сигналу (Lucic et al., 2023);

- наземні станції керування (*Ground Control Stations*) – високорівневе програмне забезпечення (наприклад, Mission Planner, QGroundControl), що використовується для планування місій, моніторингу телеметрії та діагностики БПЛА в польоті (QGroundControl Guide (Daily Builds), n.d.).

Відкриті платформи (*Open Source*) є основою сучасних інновацій у сфері БПЛА (Bergguist, 2022). Домінуючими на цьому сегменті є ArduPilot (What is ArduPilot, n.d.) та PX4 (PX4 Autopilot User Guide, n.d.).

ArduPilot – одна з найстаріших платформ, що характеризується сумісністю та підтримкою різноманітних апаратних платформ. ArduPilot характеризується, зокрема, тим, що:

- у навігаційному модулі ArduPilot використовує розширений фільтр Калмана завдяки своїй навігаційній бібліотеці (AP_NavEKF3), працюючи з трьома паралельними екземплярами фільтра (What is ArduPilot, n.d.; Bergguist, 2022);

- високим рівнем інтеграційної спроможності, бо ArduPilot підтримує інтеграцію зовнішніх модулів локалізації (наприклад, візуальної одометрії (Cadena et al., 2016) чи SLAM (Mur-Artal, Montiel and Tardós, 2015)) через протокол MAVLink (Meier, Honegger and Pollefeys, 2015), даючи змогу використовувати зовнішні обчислювальні одиниці на базі ROS (Macenski et al., 2022) та бібліотек OpenCV чи PCL (Zhang and Singh, 2014; La Vallee, 2018).

Платформа PX4 відома своєю модульною архітектурою та прецизійним контролем і надійністю (PX4 Autopilot User Guide, n.d.). Для PX4 характерним є:

- інтерфейс uORB – внутрішній комунікаційний механізм uORB, що забезпечує ефективний обмін даними між модулями, дозволяючи легко кастомізувати та інтегрувати нові сенсори або алгоритми визначення координат; це обумовлює швидку розробку специфічних навігаційних рішень, включаючи методи SLAM (Chen et al., 2024);

- більш висока точність утримання позиції порівняно з ArduPilot (Bergguist, 2022).

Комерційні програмні екосистеми (відповідні програмні рішення) пропонують високий ступінь інтеграції та надійність, але мають закритий код.

Екосистема DJI (SDK та API) є світовим лідером, що пропонує пропріетарні прошивки, надає розробникам доступ до апаратного та програмного забезпечення через свої SDK (*Software Development Kits*) (Bergguist, 2022). Для DJI характерним є:

- обмеження локалізації, коли базові функції визначення координат є закритими, однак SDK дає змогу розробникам отримувати дані із сенсорів і використовувати зовнішні обчислювальні платформи для запуску власних алгоритмів SLAM / VO (Lucic et al., 2023);

- наявність спеціалізованих рішень, бо комерційний ринок насичений пропріетарними GCS і системами планування польотів (наприклад, Agisoft Metashape), які інтегруються з відкритими автопілотами, але пропонують унікальні функції так званої постобробки та картографування.

Комерційні компанії (наприклад, такі як Auterion) будують свої програмні рішення на основі PX4 (PX4 Autopilot User Guide, n.d.). Перевагою такого підходу є те, що вони гарантують підвищену безпеку та відповідність нормативним вимогам (*Compliance momentum*), критично важливим для галузі, яку досліджують. Платформи підтримують детермінізм збірок і цифрові підписи параметрів, ставлячи основою сертифікації, але є недоступними в стандартних відкритих збірках.

Ефективність програмного забезпечення оцінюється за критеріями (Bergguist, 2022), представленими в табл. 2.

Таблиця 2

Порівняння архітектурної гнучкості та інтеграції

Критерій	Відкриті платформи (PX4 / ArduPilot)	Комерційні платформи (DJI SDK, Auterion)
Реалізація Sensor Fusion	Повна прозорість EKF-алгоритмів (Sabzevari and Chatraei, 2021). Легкість інтеграції користувацьких фільтрів (завдяки ROS (Macenski et al., 2022), PCL (Zhang and Singh, 2014), OpenCV (Cadena et al., 2016)).	Закриті, високооптимізовані алгоритми. Інтеграція кастомізованої локалізації відбувається через зовнішні SDK / API.
Модульність та адаптація	Висока. PX4 (uORB) пропонує кращу структурну модуляризацію (PX4 Autopilot User Guide, n.d.), що полегшує внесення змін до низькорівневого контролю.	Низька. Адаптація зводиться до використання готових функцій та налаштування через GCS (QGroundControl Guide (Daily Builds), n.d.).
Симуляція (SITL / HIL)	Висока цілісність симуляції. Активне використання Dual-SITL для регресійного контролю.	Симуляційні можливості обмежені або пропріетарні, орієнтовані на валідацію кінцевої місії.

Відкриті платформи (завдяки інтеграції з ROS-екосистемою) надають найбільшу кількість готових SLAM-рішень (наприклад, VINS-Fusion (Qin, Li and Shen, 2018)

або Open-VINS), які можуть досягати сантиметрової точності в приміщеннях, використовуючи візуальні дані (Lucic et al., 2023).

Боротьба з накопичувальною похибкою (так званим дрейфом) ІНС / одометрії є основним завданням. У відкритих системах це контролюється через перевірку цілісності симуляції та використання трьох паралельних фільтрів Калмана (ArduPilot) (What is ArduPilot, n.d.).

У табл. 3 наведено порівняння основних аналогів Open Source.

Таблиця 3

Основні аналоги Open Source

Платформа	Програмне забезпечення	Візуальна ілюстрація	Коментар
ArduPilot	Mission Planner (GCS) (QGroundControl Guide (Daily Builds), n.d.)	Рис. 2. Інтерфейс Mission Planner з вікном налаштування ядра	Демонструє високу функціональність наземної станції та доступ до налаштувань ядра
PX4	QGroundControl (GCS) (QGroundControl Guide (Daily Builds), n.d.)	Рис. 3. Інтерфейс QGroundControl з вікном налаштування Sensor Fusion	Наочно ілюструє модульний підхід PX4 та універсальність GCS

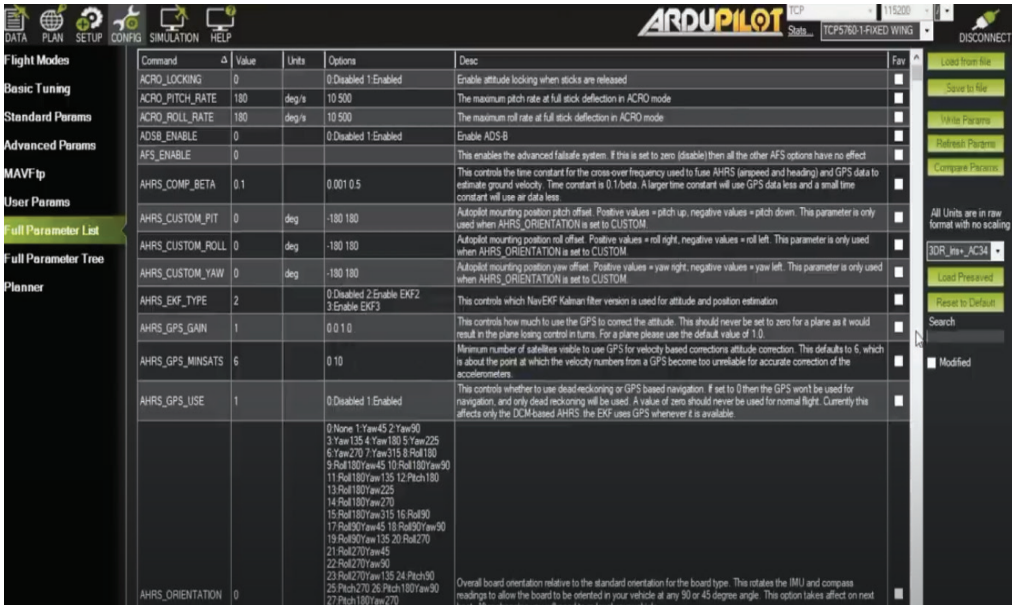


Рис. 2. Інтерфейс Mission Planner з вікном налаштування ядра
Джерело: авторська розробка

Вибір програмного забезпечення для визначення координат є компромісом між архітектурною гнучкістю та готовністю до експлуатації. PX4 та ArduPilot є оптимальними для наукових досліджень і розроблення нових алгоритмів локалізації (наприклад, SLAM-модулів) (Bergguist, 2022). Комерційні рішення надають вищий ступінь надійності та безпеки для сертифікованих операцій, але з обмеженою можливістю модифікації ядра навігації.

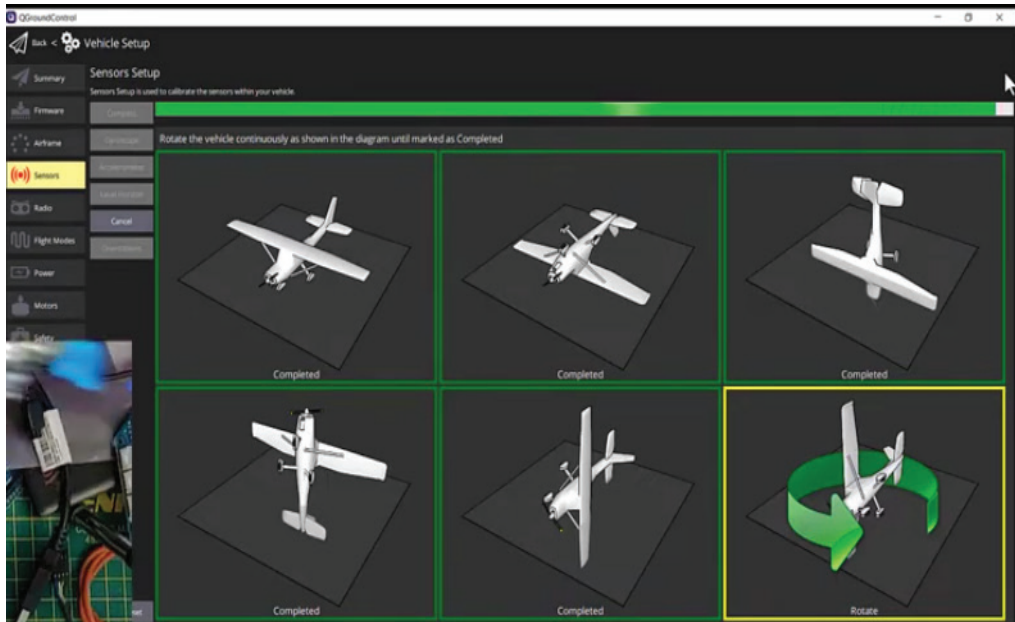


Рис. 3. Інтерфейс QGroundControl з вікном налаштування Sensor Fusion

Джерело: авторська розробка

Для перевірки ефективності багатосенсорної інтеграції використано архітектуру й імітаційну модель відповідного програмного комплексу, який був реалізований на основі платформи Robot Operating System (ROS) (Macenski et al., 2022), що виступає як інтеграційне середовище (фреймворк високого рівня), забезпечуючи асинхронну комунікацію між незалежними обчислювальними модулями, критичним для систем реального часу (Bai, Wen and Hsu, 2020).

На рис. 4 представлена архітектура розробленого комплексу, що демонструє функціональні (сенсорні, обчислювальні та вихідні) вузли й потоки даних.

На рис. 4 представлені такі вузли, як:

1. Сенсорні (*Input Nodes*) – незалежні модулі, що забезпечують первинну обробку даних. Для підтримки цих вузлів передбачено використання таких бібліотек:

- *gps_node* для обробки даних GNSS (з використанням спеціалізованих ROS-пакетів для GPS (Farrell and Barth, 1999; Lucic et al., 2023));

- *IMU_node* для обробки даних від інерціальної навігаційної системи, які важливі для фази прогнозу фільтра Калмана (Grewal, Andrews and Bartone, 2020);

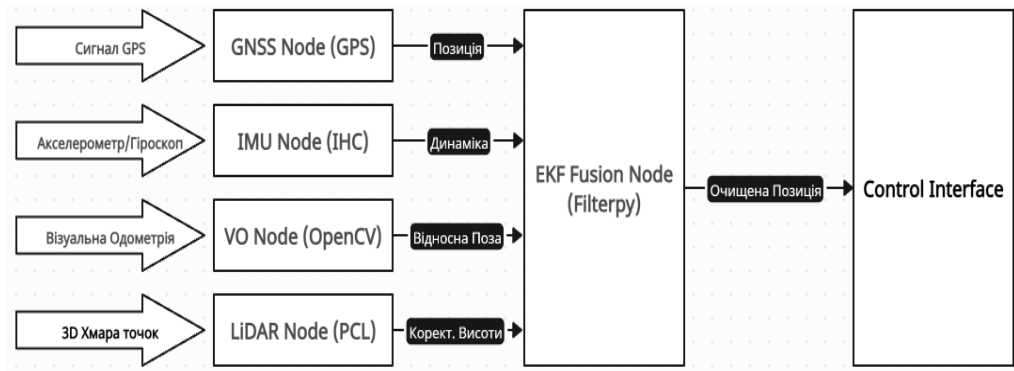


Рис. 4. Архітектура багатосенсорного програмного комплексу визначення координат (на базі інтеграції ROS-EKF)

Джерело: авторська розробка

- vo_node для реалізації візуальної одометрії (яка використовує бібліотеку OpenCV (Cadena et al., 2016) для виділення та відстеження ключових точок);
- lidar_node для обробки хмари точок LiDAR, яка використовує бібліотеку PCL (Point Cloud Library) (Zhang and Singh, 2014) для локального картографування.

2. Обчислювальний (*Fusion Node*) відповідає за інтеграцію асинхронних даних, використовуючи розширений фільтр Калмана (La Vallee, 2018).

3. Вихідний (*Control Node*) отримує фінальну оцінку стану і використовується для запису результатів або передачі даних до автопілота через MAVLink (Meier, Honegger and Pollefeys, 2015).

Для оцінки ефективності розробленого програмного комплексу проведено серію експериментів, спрямованих на порівняння точності визначення координат у режимах GNSS-навігації та гібридної навігації (табл. 3).

Критерієм оцінки була середньоквадратична похибка (Bednář et al., 2022).

Таблиця 3

Результати імітаційної валідації

Режим навігації	Середньоквадратична похибка позиції, м	Максимальне відхилення, м	Обчислювальне навантаження, %
GNSS	0.82	1.45	~15
Гібридна	0.11	0.25	~45

Отримані результати демонструють перевагу розробленого гібридного комплексу для прецизійного визначення координат при автономному польоті БПЛА.

Висновки. Проведене дослідження спрямоване на підвищення ефективності визначення координат при автономному польоті БПЛА за допомогою критичного аналізу ринкових рішень і розроблення робастного багатосенсорного програмного комплексу. У роботі досягнуто таких результатів:

- виконано узагальнення фундаментальних методів визначення координат;

- обґрунтовано необхідність Sensor Fusion для мінімізації накопичувальної похибки та забезпечення стійкості до відмов, що стало основою для подальшої програмної реалізації;
- проведено аналіз ринку програмного забезпечення для БПЛА;
- здійснено класифікацію сучасних програмних засобів за архітектурним принципом, виділивши дихотомію між відкритими платформами (ArduPilot, PX4) та комерційними екосистемами;
- встановлено, що відкриті платформи надають більш високу гнучкість, необхідну для імплементації нових алгоритмів SLAM, а комерційні рішення пропонують високий ступінь інтеграції та відповідності нормативним вимогам;
- для реалізації надійного визначення координат у закритих та GNSS відсутніх середовищах розроблено програмний комплекс на базі Robot Operating System, що забезпечило модульний підхід та легкість масштабування.

Результати дослідження підтверджують те, що багатосенсорна інтеграція на основі адаптованого фільтра Калмана є найбільш ефективним програмним рішенням для забезпечення прецизійного автономного польоту БПЛА, що має значну практичну цінність для розроблення навігаційних систем нового покоління.

REFERENCES

- Bai, X., Wen, W. and Hsu, L.-T., 2020. Robust Visual-Inertial Integrated Navigation System Aided by Online Sensor Model Adaption for Autonomous Ground Vehicles in Urban Areas. *Remote Sensing*, [e-journal] 12 (10), 1686. <https://doi.org/10.3390/rs12101686>
- Bednář, J., Petrлік, M., Vivaldini, K.C.T. and Saska, M., 2022. Deployment of Reliable Visual Inertial Odometry Approaches for Unmanned Aerial Vehicles in Real-world Environment. *IEEE International Conference on Unmanned Aircraft Systems (ICUAS)*, [e-journal] pp.168-176. <https://doi.org/10.1109/ICUAS54217.2022.9836067>
- Bergguist, C., 2022. ArduPilot VS PX4. Which is better? *Drone Dojo*, [online] 29 August. Available at: <<https://dojofordrones.com/ardupilot-vs-px4>> [Accessed 22 September 2025].
- Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I. and Leonard, J.J., 2016. Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. *IEEE Transactions on Robotics*, [e-journal] 32 (6) pp.1309-1332. <https://doi.org/10.48550/arXiv.1606.05830>
- Chen, R., Tan, J., Zhou, Y., Yang, Z., Nie, F. and Chen, T., 2024. Asymmetric low-rank double-level cooperation for scalable discrete cross-modal hashing. *Expert Systems with Applications*, [e-journal] 237, Part C, Article 121703. <https://doi.org/10.1016/j.eswa.2023.121703>
- Dai, Y., Gieseke, F., Oehmcke, S., Wu, Y. and Barnard, K., 2020. Attentional Feature Fusion. In: *IEEE Winter Conference on Applications of Computer Vision (WACV)*, Waikoloa, HI, USA, 03-08 January 2021. [online] Waikoloa: Institute of Electrical and Electronics Engineers, pp.3559-3568. Available at: <<https://arxiv.org/pdf/2009.14082>> [Accessed 22 September 2025].
- Farrell, J. and Barth, M., 1999. *The Global Positioning System & Inertial Navigation*. New York: McGraw-Hill Education.
- Grewal, M.S., Andrews, A.P. and Bartone, C.G., 2020. *Global Navigation Satellite Systems, Inertial Navigation, and Integration*. 4th ed. New York: John Wiley & Sons.

- La Vallee, S.M., 2018. FilterPy – Kalman filters and other optimal and non-optimal estimation filters in Python [online]. Available at: <<https://github.com/rlabbe/filterpy>> [Accessed 28 September 2025].
- Lin, C.-F., 2008. *Modern Navigation, Guidance, And Control Processing*. Pearson Technology Group.
- Lucic, M.C., Bouhamed, O., Ghazzai, H., Khanfor, A. and Massoud, Y., 2023. Leveraging UAVs to Enable Dynamic and Smart Aerial Infrastructure for ITS and Smart Cities: An Overview. *Drones*, [e-journal] 7 (2), 79. <https://doi.org/10.3390/drones7020079>
- Macenski, S., Foote, T., Gerkey, B., Lalancette, Ch. and Woodall, W., 2022. Robot Operating System 2: Design, Architecture, and Uses In The Wild. *Science Robotics*, [e-journal] 7 (66). <https://doi.org/10.48550/arXiv.2211.07752>
- Meier, L., Honegger, D. and Pollefeys, M., 2015. PX4: A node-based multithreaded open source robotics framework for deeply embedded platforms. In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*, Seattle, WA, USA, 26-30 May, 2015. Seattle: Institute of Electrical and Electronics Engineers, pp.6235-6240.
- Mur-Artal, R., Montiel, J.M.M. and Tardós, J.D., 2015. ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *IEEE Transactions on Robotics*, [e-journal] 31 (5), pp.1147-1163. <https://doi.org/10.1109/TRO.2015.2463671>
- PX4 Autopilot User Guide. n.d. [online] Available at: <<https://docs.px4.io/main/en/>> [Accessed 22 September 2025].
- QGroundControl Guide (Daily Builds), n.d. *QGC Guide*. [online] Available at: <<https://docs.qgroundcontrol.com/>> [Accessed 24 September 2025].
- Qin, T., Li, P. and Shen, S., 2018. VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator. *Ieee transactions on robotics*, [e-journal] 34 (4), pp.1004-1020. Available at: <<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8421746>> [Accessed 22 September 2025].
- Sabzevari, D. and Chatraei, A., 2021. INS/GPS Sensor Fusion based on Adaptive Fuzzy EKF with Sensitivity to Disturbances. *IET Radar, Sonar & Navigation*, [e-journal] 15 (11), pp.1535-1549. <https://doi.org/10.1049/rsn2.12144>
- What is ArduPilot? n.d. *ArduPilot*. [online] Available at: <<https://ardupilot.org/>> [Accessed 22 September 2025].
- Zhang, J. and Singh, S., 2014. LOAM: Lidar Odometry and Mapping in Real-Time. In: *Robotics: Science and Systems Conference*, Berkeley, [online] July. Available at: <https://www.ri.cmu.edu/pub_files/2014/7/Ji_LidarMapping_RSS2014_v8.pdf> [Accessed 24 September 2025].
- Zhao, Y., Yan, G., Qin, Y. and Fu, Q., 2020. Information Fusion Based on Complementary Filter for SINS/CNS/GPS Integrated Navigation System of Aerospace Plane. *Sensors*, [e-journal] 20 (24), 7193. <https://doi.org/10.3390/s20247193>

UDK 004.771:004.41

Oleksandr Tkachenko,

*PhD in Physics and Mathematics, Associate Professor,
Associate Professor at the Department of Computer Systems Software
National Technical University of Ukraine
“Igor Sikorsky Kyiv Polytechnic Institute”,
Associate Professor at the Department of Information Technologies
Educational and Scientific Institute of Management,
Technology and Legal Sciences,
National Transport University,
Kyiv, Ukraine
aatokg@gmail.com
<http://orcid.org/0000-0001-6911-2770>*

Mykyta Hrebeniuk,

*Master's Student at the Department of Information Technologies,
Educational and Scientific Institute of Management,
Technology and Legal Sciences
National Transport University,
Kyiv, Ukraine
4ekn14e@gmail.com
<http://orcid.org/0009-0009-9610-7518>*

SOME ASPECTS OF SOFTWARE FOR COORDINATE DETERMINATION DURING AUTONOMOUS UAV FLIGHT

Today, the scope of application of unmanned aerial vehicles is expanding, particularly in logistics, high-precision agronomy, defence, and monitoring tasks. All this highlights the problem of ensuring their autonomy and high navigation accuracy. An analysis of the current market for software for autonomous control of unmanned aerial vehicles reveals a shortage of necessary software solutions, which complicates the selection of the most effective software package.

The purpose of the article is to study modern software tools designed to determine the coordinates and ensure the navigation and localisation of uncrewed aerial vehicles in autonomous mode, as well as the functional and technical capabilities of these devices.

The research methodology includes a comparative analysis of the leading software solutions in this subject area (unmanned aerial vehicles). The article discusses approaches to developing and operating a software solution for determining coordinates during the autonomous flight of such aircraft, based on the use of an extended Kalman filter, which facilitates the fusion of data from inertial, visual, and satellite systems.

Conclusions. The paper summarises fundamental methods for determining coordinates. It substantiates the need for sensor fusion to minimise cumulative error and ensure fault tolerance, which became the basis for further software implementation. An analysis of the UAV software market is conducted, where modern software tools are classified according to architectural principles, highlighting the dichotomy between open platforms (such as ArduPilot and PX4) and commercial ecosystems. It was established that open platforms provide greater

flexibility, which is necessary for implementing new SLAM algorithms. At the same time, commercial solutions offer a high degree of integration and compliance with regulatory requirements. To implement reliable coordinate determination in closed environments and environments without GNSS, a software complex based on the Robot Operating System was developed (this ensured a modular approach and ease of scaling).

Keywords: unmanned aerial vehicle; autonomous flight; coordinate determination; UAV software; GNSS; Kalman filter.

Надійшла 03.10.2025

Прийнята 28.11.2025

Стаття була вперше опублікована онлайн 29.12.2025



This is an open access journal, and all published articles are licensed under a Creative Commons Attribution 4.0.