

**НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

Кваліфікаційна наукова
праця на правах рукопису

Долінце Богдан Ігорович

УДК 004.051:623.746-519(043.5)

**ДИСЕРТАЦІЯ
МЕТОДИ ТА ЗАСОБИ ПІДВИЩЕННЯ ТОЧНОСТІ ОБРОБКИ
ІНФОРМАЦІЇ В БОРТОВИХ ПІДСИСТЕМАХ БпЛА**

05.13.05 – Комп'ютерні системи та компоненти

Дисертація на здобуття наукового ступеня
кандидата технічних наук

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.



Б.І. Долінце

Науковий керівник:

Жуков Ігор Анатолійович

доктор технічних наук, професор,
Заслужений винахідник України

Київ – 2024

АНОТАЦІЯ

Долінце Б.І. Методи та засоби підвищення точності обробки інформації в бортових підсистемах БпЛА. – Кваліфікаційна наукова праця на правах рукопису.

Дисертаційна робота для здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.05 «Комп’ютерні системи та компоненти» – Національний авіаційний університет, Київ, 2024.

Дисертація зосереджена на розв’язанні важливого методологічного та науково-технічного завдання, що полягає у підвищенні ефективності обробки інформації в підсистемах позиціонування безпілотних літальних апаратів використовуючи методи злиття інформації від бортових та супутникових джерел інформації, а також методи корекції і оптимальної обробки інформації.

Для ефективного, точного та надійного визначення положення безпілотних літальних апаратів під час виконання завдань необхідно одночасно використовувати різні методи, засновані як на оптимальній обробці інформації в підсистемах, так і на можливості використання альтернативних джерел інформації, таких як LEO комунікаційні супутники. Такий підхід дозволить підвищити ефективність обробки інформації про позиціонування та дасть можливість знизити похибку таких даних.

Дисертаційна робота присвячена підвищенню ефективності систем позиціонування безпілотних літальних апаратів використовуючи методи злиття інформації від бортових та супутникових джерел інформації, а також методи корекції і оптимальної обробки інформації.

Аналіз літературних джерел показав, що відомі методи підвищенню ефективності систем позиціонування безпілотних літальних апаратів обмежені й мають низьку ефективність, а також потребують постійного зв’язку із глобальними системами супутникового позиціонування для підтримки їх роботи з бортовими засобами вимірювання. Це свідчить про те, що задача підвищення ефективності обробки даних систем позиціонування безпілотних літальних апаратах є актуальною.

Способи та методи підвищення точності обробки інформації в бортових підсистемах БпЛА описуються науковцями такими, як: В. Мелешко, В. Грабчак, О. Нестеренка, Х'ю Елкайм, М. Брайсон, С. Суккаріє, Когуре, Т. Хамфріс, М. Ельгамраві, М. Караїм та ін.

Незважаючи на значний прогрес і численні роботи, присвячені точності обробки інформації в бортових підсистемах БпЛА, варто відзначити, що сучасні завдання не можуть бути вирішені через існуючі методи та засоби позиціонування. Це робить розробку нових моделей та методів для підвищення точності позиціонування БпЛА особливо актуальною. Саме ці виклики визначають напрямок досліджень у даній дисертаційній роботі.

Складено перелік необхідних вимог та критеріїв для визначення необхідної точності та оперативності обробки інформації в бортових підсистемах БпЛА. Окреслено ключові напрямки удосконалення сучасних методів підвищення точності обробки інформації в бортових підсистемах БпЛА. Аналізуючи сучасні методи позиціонування, було виявлено потребу у розробці нових методів, які дозволяють використовувати як існуючі, так і новітні джерела інформації для покращення якості та підвищення точності обробки даних про позиціонування.

Виконано аналіз існуючих методів та сучасних моделей, які можуть бути використані при обробці інформації про позиціонування в бортових підсистемах БпЛА. Порівняно характеристики зазначених методів і визначено їх переваги, недоліки, сильні та слабкі сторони. Сформовано вимоги до обраних методів на основі збереження точності та швидкодії обробки інформації про позиціонування (без використання додаткових бортових обчислювальних засобів). Розглянуто методи компенсації та оптимальної фільтрації похибок вимірювань, що дозволяє отримувати робастні дані про положення та забезпечувати вищу стабільність роботи бортових підсистем. Фільтр Калмана є механізмом для оптимального злиття даних з кількох джерел. При поєднанні різних систем позиціонування можливе значне підвищення точності визначення параметрів позиціонування та руху БпЛА.

Описано основні недоліки та обмеження розглянутих систем позиціонування. Сформовано основні задачі проведення дослідження, визначені

шляхи підвищення точності обробки інформації про позиціонування в БпЛА. Визначено головні напрями дисертаційного дослідження та сформовано його основні завдання.

У роботі визначено методи підвищення точності позиціонування, що включають інтеграцію інерціальних та супутникових джерел інформації, обробку інформації в бортових підсистемах БпЛА з використанням інтеграції та оптимальної фільтрації, що використовує багатокomпонентний підхід обробки інформації з різних джерел даних позиціонування, включаючи телекомунікаційні супутники низької земної орбіти, глобальної системи позиціонування, бортові автономні інерціальні підсистеми та дискретного фільтра Калмана для оптимального злиття даних з кількох джерел для створення моделі робастної багатосупутникової системи з підвищеною точністю позиціонування в реальному часі. Досліджено можливості застосування моделі багатосупутникових системи позиціонування для вивчення особливостей роботи запропонованих методів.

Основні наукові результати роботи заключаються в тому, що: *уперше*: розроблено метод злиття інформації для багато-супутникової обробки інформації про позиціонування на базі методів оптимальної обробки інформації та прямої корекції помилок з використанням інформації від LEO-супутників, який дозволяє та дає можливість підвищити стійкість процесу обробки інформації та дозволяє підвищити точність визначення позиціонування; *удосконалено*: методику оцінювання інформації за одним вимірювальним каналом про позиціонування, який, на відміну від відомих підходів, дозволяє оцінювати значення параметрів шляхом злиття інформації від LEO-супутників та відомих бортових підсистем позиціонування БпЛА (GNSS та INS), які дозволяють в реальному часі обробляти надлишкову інформацію та отримувати точні дані про позиціонування (похибка позиціонування $\sim 3,12$ м, похибка визначення швидкості $\sim 0,2$ м/с); *дістали подальшого розвитку*: метод комплексного оброблення інформації в підсистемах позиціонування БпЛА, який на відміну від відомих відрізняється тим, що враховує оцінку цілісності даних в багатосупутниковій підсистемі позиціонування LeGNSS за допомогою

розробленого алгоритму оцінки цілісності отримуваної інформації та дає можливість здійснювати постійний контроль за даними, що обробляються в підсистемі позиціонування БпЛА (виходить на робочий режим роботи за 2–5с після ініціалізації); метод комплексної обробки інформації з оптимальною та адаптивною фільтрацією похибок позиціонування з використанням програмних алгоритмів, який на відміну від існуючих методів дає можливість визначення позиціонування за рахунок відфільтрування високочастотних похибок та адаптації до впливу постійних похибок, що дозволяє обмежити пікові відхилення похибок по точності позиціонування (підвищити точність на 9 %).

Проведено дослідження ефективності методу підвищення точності позиціонування шляхом використання багатокомпонентного підходу обробки інформації з різних джерел даних позиціонування, включаючи телекомунікаційні супутники низької земної орбіти, глобальної системи позиціонування, бортові автономні інерціальні підсистеми та дискретного фільтра Калмана для оптимального злиття даних в робастній багатосупутниковій системі з підвищеною точністю позиціонування (LeGNSS) в реальному часі. На основі аналізу отриманої інформації зроблений наступний висновок: підхід використання архітектури змішаних багато-супутникових систем позиціонування є перспективним та дозволяє досягти відчутних результатів підвищення точності позиціонування. Використовувати сигнали від супутників LEO, як доповнення або альтернативу для позиціонування, позиціонування та визначення часу, є доцільним оскільки вони знаходяться значно ближче до Землі порівняно з супутниками GNSS, що дозволяє підсистемі позиціонування швидше реагувати на переміщення БпЛА в просторі та частіше отримувати інформації про поточне положення і знизити загальну похибку системи. Метод оцінки цілісності системи менше навантажує бортову систему користувача, але частіше визначає підозрілу активність як погіршення якості інформації або її підміну. Результати порівняльного аналізу моделі LeGNSS демонструють, що запропоновані методи перевершують відомі моделі підвищення точності позиціонування в БпЛА, використані в порівняльному тесті та здатні виявити зовнішній вплив на джерела інформації.

Проведено порівняльний аналіз запропонованих в дослідженні рішень, котрий експериментально підтвердив, що дані методи підвищують ефективність обробки інформації про позиціонування в бортових підсистемах БпЛА.

Розроблені в дисертаційній роботі методи і моделі можуть бути використані для підвищення точності обробки інформації про позиціонування в бортових підсистемах БпЛА. Експериментальні дослідження надають підтвердження основних положень, що виносяться на захист дисертаційної роботи. Розроблені методи підвищення точності обробки інформації про позиціонування можуть бути застосовані для забезпечення роботи засобів позиціонування та керування БпЛА, а також відповідних систем автоматичного керування місіями дронів. Для захисту новизни запропонованих рішень подано документи на видачу патенту на корисну модель України (заявка № u202305401 від 13.11.2023).

Результати дисертаційної роботи використовуються: 1) для оцінки точності обробки інформації про позиціонування під час реалізації проєктів розробки автономних безпілотних системи в Громадській Спілці «Бізнес Інкубатор Груп. Україна» (акт впровадження від 06.10.2023); 2) для підвищення ефективності оцінки позиціонування і діагностики точності її обробки у Громадській Організації «Всеукраїнський Центр Реформ Транспортної Інфраструктури» (акт впровадження від 17.10.2023); 3) рекомендовані до застосування під час розроблення, виготовлення, випробування, забезпечення процесів комп'ютерного моделювання для сучасних космічних та безпілотних літальних апаратів в Асоціації «КОСМОС» (акт впровадження від 24.10.2023); 4) для підготовки авіаційних фахівців авіаційних підприємств щодо особливостей використання підходу побудови багато-супутникових систем позиціонування, які входять до Громадської Спілки «Українська Авіатранспортна Асоціація» (акт впровадження від 12.10.2023).

Ключові слова: LeGNSS, INS, супутники LEO, БпЛА, обробка інформації, багато-супутникова система позиціонування, точне позиціонування, управління безпілотними апаратами.

SUMMARY

Dolintse B.I. Methods and ways to improve the accuracy of information processing in the onboard subsystems of UAV. – Manuscript.

Dissertation for the degree of Candidate of Technical Sciences in specialty 05.13.05 “Computer systems and components”, – National Aviation University, Kyiv, 2024.

The dissertation solves the present methodological and scientific-technical problem of increasing the efficiency of unmanned aerial vehicle positioning systems using information fusion methods from onboard and satellite information sources and methods of correction and optimal information processing.

For efficient, accurate, and robust determination of positioning of the unmanned aerial vehicles during missions, it is necessary to implement and use methods based on both optimal information processing in subsystems and the possibility of using alternative sources of information, such as LEO communication satellites. This approach will increase the efficiency of positioning data processing and reduce errors.

An analysis of the published literature has demonstrated that known methods for improving the efficiency of unmanned aerial vehicle positioning systems could be more efficient and require consistent connectivity with global satellite positioning systems to support their work with onboard measurement devices. Increasing the data processing efficiency of UAV positioning systems is relevant.

Ways and methods to improve the accuracy of information processing in the onboard subsystems of UAVs are described by scientists such as V. Meleshko, V. Hrabchak, O. Nesterenko, Hugh Elkim, M. Bryson, S. Sukkarie, Kogure, T. Humphreys, M. Elgamrawi, M. Karaim, etc.

Despite significant progress and numerous studies on the accuracy of information processing in UAV onboard subsystems, it is worth noting that they have features that limit the effectiveness of modern positioning tools. Therefore, the issue of developing methods and models to improve the positioning accuracy for UAVs is becoming increasingly relevant. The tasks that arise in this case have determined the research direction in this dissertation.

Compiled a list with criteria and requirements for determining the accuracy and efficiency of information processing in UAVs and defined ways to improve modern methods of increasing the accuracy of information processing. While analysing methods of positioning, it was found that new techniques are needed to use both existing and new sources of information to improve and increase the positioning.

It analyses the current methods and modern models that can be used for processing positioning information in the onboard subsystems of UAVs. Characteristics of these methods are compared, and their advantages, disadvantages, strengths, and weaknesses are determined. Criteria for these methods are then developed, emphasizing the accuracy and efficiency of processing positioning (without additional onboard computing facilities). Methods of compensation and optimal filtering of measurement errors are considered, which allows for obtaining robust position data and ensuring higher stability of onboard subsystems. The Kalman filter is a mechanism for optimal data fusion from multiple sources. The fusion of different data sources makes it possible to improve the positioning accuracy.

The main disadvantages and limitations of the considered models of positioning systems are described. The main objectives of the study are formed, and ways to improve the accuracy of processing positioning information in UAVs are identified. The main directions of the dissertation are determined, and its main tasks are formed.

In this research, there have been identified methods for improving positioning accuracy, including the integration of inertial and satellite sources of information, processing of information in the onboard subsystems of UAVs using integration and optimal filtering, which uses a multicomponent approach to processing information from various sources of positioning data, including telecommunication satellites of low Earth orbit, global positioning system, onboard autonomous inertial subsystems and the discrete Kalman filter for optimal data fusion from several satellites. Using a model of a multisatellite system to simulate the operation of proposed methods are investigated.

The main scientific results are as follows: *for the first time proposed* a method of information fusion for multisatellite positioning information processing based on methods of optimal information processing and direct error correction using

information from LEO satellites, which allows and makes it possible to increase the stability of the information processing process and allows to increase the accuracy of positioning determination; *an improved* a method for evaluating information on one measurement channel about positioning, which, unlike known approaches, allows to estimate the values of parameters by merging information from LEO satellites and known onboard positioning subsystems of UAVs (GNSS and INS), which allow to process redundant information in real time and obtain accurate positioning data (positioning error ~ 3.12 m, speed error ~ 0.2 m/s); *have been further developed* a method of integrated information processing in the UAV positioning subsystems, which, unlike the known ones, differs in that it takes into account the assessment of data integrity in the LeGNSS multisatellite positioning subsystem using the developed algorithm for assessing the integrity of the received information and makes it possible to constantly monitor the data processed in the UAV positioning subsystem (reaches the operating mode in 2–5 seconds after initialization); *have been further developed* method of complex information processing with optimal and adaptive filtering of positioning errors using software algorithms, which, unlike existing methods, makes it possible to determine positioning by filtering out high-frequency errors and adapting to the influence of constant errors, which allows to limit peak deviations of errors in positioning accuracy (increase accuracy by 9%).

There was conducted research into the effectiveness of the method of improving positioning accuracy by using a multi-component approach to processing information from various sources of positioning data, including low Earth orbit telecommunication satellites, the global positioning system, onboard autonomous inertial subsystems, and a discrete Kalman filter for optimal data fusion in a robust multisatellite system with increased real-time positioning accuracy. Based on analysis of the collected information following conclusion was reached: the approach of using the architecture of mixed multisatellite positioning systems is promising and allows for achieving tangible results in improving positioning accuracy, using signals from LEO satellites as a supplement or alternative for positioning, navigation, and timing because they are much closer to the Earth compared to GNSS satellites, which allows the positioning subsystem to respond faster to UAV movement in space and receive information about

the current position more often and reduce the overall system error. The method of assessing system integrity puts less strain on the user's onboard system but more often identifies suspicious activity as deterioration of information quality or its substitution. The comparative LeGNSS model evaluation, as presented in this study, reveals that methodologies introduced in this dissertation enhance existing models in the accuracy of positioning UAVs, as confirmed in the comparative tests. These methods presented a superior capability in detecting external influences on information sources.

In addition, the solutions proposed in this study underwent a thorough comparative analysis, experimentally validating their effectiveness in improving the processing of positioning data in the onboard subsystems of UAVs.

Methods developed in the dissertation work can be used to improve the accuracy of positioning information processing in the onboard subsystems of UAVs. Experimental research confirms the main points made in the thesis. Developed methods for improving the accuracy of positioning information processing can be used to ensure the operation of UAV positioning and control for corresponding systems of automatic mission control. For the protection of system design, a model patent of Ukraine has been submitted (app. No.202305401, 13.11.2023).

The results of the dissertation are used: 1) to estimate the accuracy of positioning information processing during the implementation of projects for the development of autonomous unmanned systems in the Public Union “Business Incubator Group. Ukraine” (implementation act of 06.10.2023); 2) to increase efficiency of positioning assessment and diagnostics of its processing accuracy in the Public Organization “All-Ukrainian Center of Transport Infrastructure Reforms” (implementation act of 17.10.2023); 3) recommended to use for development, production, testing, and support of computer modeling processes of modern space and unmanned aerial vehicles by the Association “KOSMOS” (implementation act of 24.10.2023); 4) for professional training of aviation specialists at aviation enterprises on the features of using the multisatellite positioning system design approach, which are part of the Public Union “Ukrainian Air Transport Association” (implementation act of 12.10.2023).

Keywords: LeGNSS, INS, LEO satellites, UAV, information processing, multisatellite positioning system, precise positioning, autonomous vehicles control.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА

Праці, які відображають основні наукові результати дисертації:

1. Васильєв В.М., Рогожин В.О., Долінце Б.І. Схема комплексування інерціальної і супутникової навігаційних систем та її дослідження. *Проблеми створення, випробування, застосування та експлуатації складних інформаційних систем: збірник наукових праць*. Житомир: ЖВІ НАУ, 2013. Вип. 8. С. 20–25. (Входить до реєстру наукових видань України, Категорія Б; Входить до міжнародних наукометричних баз Copernicus, SemanticScholar, Ulrichsweb GSD, ResearchBib, GoogleScholar, BASE)
2. Васильєв В.М., Рогожин В.О., Долінце Б.І. Дослідження похибок інерціально-супутникової навігаційної системи. *Наука і техніка Повітряних сил Збройних сил України*. Харків: ХУПС, 2013. Вип. 3(12). С. 139–143. (Входить до реєстру наукових видань України, Категорія Б; Входить до міжнародних наукометричних баз GoogleScholar, DOAJ, WorldCat, BASE)
3. Vasyliiev V.M., Rogozhyn V.O., Dolintse B.I. Integration of inertial and satellite navigation systems using corrective circuits for UAV. *Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*. IEEE Xplore, 2015. P. 193–197. DOI: 10.1109/APUAVD.2015.7346597. (date of access: 27.12.2023; Входить до міжнародних наукометричних баз Scopus, IEEE Xplore)
4. Vasyliiev V., Rogozhyn V., Dolintse B. Accuracy analysis of inertial-satellite navigation system Integrated with using corrective circuits. *Electronics and Control Systems*. 2015. Вип. 4(46). P. 46–51. DOI: 10.18372/1990-5548.46.9930. (Входить до реєстру наукових видань України, Категорія Б; Входить до міжнародних наукометричних баз CrossRef, EBSCOhost, ResearchBib, SSM, Ulrichsweb GSD)
5. Vasyliiev V., Dolintse B. Integration of Inertial and Satellite Navigation Systems with using Corrective Circuits and Filtering. *Methods and Systems of Navigation and Motion Control (MSNMC)*. IEEE Xplore, 2016. P. 275–278. DOI:

10.1109/MSNMC.2016.7783160. (date of access: 27.12.2023; Входить до міжнародних наукометричних баз *Scopus*, IEEE Xplore)

6. Dolintse B., Balakin S. Risk assessment in computer networks inherent in critical infrastructures. *Проблеми інформатизації та управління*. 2022. Вип. 2(70). Р. 4–9. DOI: 10.18372/2073-4751.70.16840. (date of access: 27.12.2023; (Входить до реєстру наукових видань України, Категорія Б; Входить до міжнародних наукометричних баз Google Scholar, EBSCOhost, WorldCat, BASE, SSM, CAS)

7. Dolintse B. Architecture of integrated navigation systems with enhanced coordinate accuracy and fault detection. *Проблеми інформатизації та управління*. 2023. Вип. 2(74). Р. 31–37. DOI: 10.18372/2073-4751.74.17878. (date of access: 27.12.2023; Входить до реєстру наукових видань України, Категорія Б; Входить до міжнародних наукометричних баз Google Scholar, EBSCOhost, WorldCat, BASE, SSM, CAS)

8. Zhukov I., Dolintse B. Enhancing accuracy of information processing in onboard subsystems of UAVs. *Technology Audit and Production Reserves*. №5/2(73), 2023. Р. 6–10. DOI: 10.15587/2706-5448.2023.287700. (date of access: 27.12.2023; (Входить до реєстру наукових видань України, Категорія Б; Входить до міжнародних наукометричних баз Google Scholar, Copernicus, Ulrichsweb GSD, CrossRef, DOAJ, OAJI, CiteFactor, WorldCat, OAA, DRJI)

9. Zhukov I., Dolintse B., Balakin S. Improving the Accuracy of Air Navigation Systems for Unmanned Aerial Vehicles. *Dependable Systems, Services and Technologies (DESSERT)*. IEEE Xplore, 2023. Р. 1–7. DOI: 10.1109/DESSERT61349.2023.10416511. (date of access: 12.02.2024; Входить до міжнародних наукометричних баз *Scopus*, IEEE Xplore)

10. Dolintse B. Robust LeGNSS positioning subsystem for UAV with correction and optimal filtering. *Проблеми інформатизації та управління*. 2023. Вип. 4(76). Р. 10–16. DOI: 10.18372/2073-4751.76.18235. (date of access: 27.12.2023; Входить до реєстру наукових видань України, Категорія Б; Входить до міжнародних наукометричних баз Google Scholar, EBSCOhost, WorldCat, BASE, SSM, CAS)

Праці, які додатково відображають наукові результати дисертації:

11. Васильєв В.М., Рогожин В.О., Долінце Б.І. Моделювання та оптимізація параметрів інтегрованої інерціально-супутникової навігаційної системи з коригувальними ланцюгами. *Молодий вчений*. 2016. Вип. 7(34). С. 184–187. (Входить до міжнародних наукометричних баз CrossRef, GoogleScholar, OAJ, ResearchBib, OUCI, Index Copernicus)
12. Долінце Б.І. Вплив навігаційних похибок на прийняття рішень в системах попередження зіткнень літаків. *Політ. Сучасні проблеми науки: XII міжнар. науково-практ. конф. студентів та молодих учених, м. Київ, 4–5 квітня 2012 р.: тези доп.* Київ: НАУ, 2012. С. 30.
13. Долінце Б.І. Дослідження характеристик інерціально-супутникової навігаційної системи з коректорами в динамічних режимах роботи. *Політ. Сучасні проблеми науки: XIII міжнар. науково-практ. конф. студентів та молодих учених, м. Київ, 3–4 квітня 2013 р.: тези доп.* Київ: НАУ, 2013. С. 123.
14. Долінце Б.І. Сучасні методи та засоби обробки інформації в інтегрованих навігаційних системах літальних апаратів. *Політ. Сучасні проблеми науки: XV міжнар. науково-практ. конф. студентів та молодих учених, 8–9 квітня 2015 р.: тези доп.* Київ: НАУ, 2015. С. 25.
15. Dolintse B. Modern Trends and Issues of the Development and Improve the Accuracy of Navigation Systems for UAVs. В: *Safety in Aviation and Space Technologies: Proceeding of The Seventh World Congress “Aviation in the XXI-st Century”*, 19–21 вересня 2016, Kyiv. Vol.2. P. 59–62.
16. Zhukov I., Dolintse B., Balakin S. Using corrective circuits and filtering for integration of low-cost inertial and satellite navigation systems. *Safety in Aviation and Space Technologies: Proceeding of The Twenty Two World Congress “Aviation in the XXI-st Century”*, September 28–30, 2022, Kyiv. 2022. Vol.2.2. P. 8–12.
17. Долінце Б.І. Методи зменшення похибок інерціально-супутникової навігаційної системи. *Проблеми навігації і управління рухом: всеукр. наук.-практ.*

конф. молодих учених та студентів, 28–29 листопада 2012, Київ. Тези доп. НАУ. 2012. С. 45.

18. Васильєв В.М., Долінце Б.І. Інерціально-супутникові навігаційні системи для безпілотних літальних апаратів. *Сучасні проблеми авіакосмічних технологій та систем*: науково-практичний семінар, 17–21 червня 2013, м. Житомир. Тези доп. НАУ. 2013. С. 6.

19. Журавель Н.В., Долінце Б.І. Підвищення точності навігаційного сигналу за допомогою фільтру Калмана. *Збірник тез доповідей XIV Міжнародної науково-практичної конференції “Комп’ютерні системи та мережні технології” (CSNT-2023)*, Київ, 13–14 квітня 2023. Київ: НАУ. 2023. С. 92–93.

20. А. с. №122052 Україна. Комп’ютерна програма «Система адаптивної обробки інформації в багато-супутниковій системі позиціонування LeGNSS» / Б. І. Долінце. 2023.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	17
ВСТУП.....	18
РОЗДІЛ 1. АНАЛІЗ СТАНУ ПИТАННЯ І ПОСТАНОВКА ЗАВДАНЬ	
ДОСЛІДЖЕННЯ.....	26
1.1 Науково-технічні проблеми точності позиціонування в БпЛА	26
1.2 Огляд літератури	28
1.3 Призначення та будова сучасних БпЛА	35
1.4 Аналіз існуючих інструментальних засобів позиціонування	38
1.4.1 Особливості інерціальної системи позиціонування.....	42
1.4.2 Особливості роботи глобальних систем позиціонування.....	45
1.5 Основні методи підвищення точності обробки інформації в БпЛА.....	49
1.6 Концепція позиціонування з використанням низькоорбітальних спутникових сузір'їв	50
1.7 Перспективи та виклики розвитку систем позиціонування БпЛА	52
1.8 Висновки до першого розділу	54
РОЗДІЛ 2. МЕТОДИ ПІДВИЩЕННЯ ТОЧНОСТІ ТА ЕФЕКТИВНОСТІ	
ФУНКЦІОНУВАННЯ СИСТЕМ ПОЗИЦІОНУВАННЯ.....	56
2.1 Методи обробки інформації в системах позиціонування БпЛА.....	56
2.2 Метод компенсації при обробці інформації про позиціонування	58
2.3 Метод компенсації зі зворотнім зв'язком	61
2.4 Метод оптимальної фільтрації	62
2.5 Алгоритм для зменшення шуму та відхилень магнітометра.....	65
2.6 Алгоритми комплексування та корекції в INS/GNSS	67
2.7 Особливості обробки інформації в багатосупутникових системах	70
2.8 Розробка методів адаптивного позиціонування для різних умов експлуатації	75
2.9 Висновки до другого розділу	77

РОЗДІЛ 3. СИНТЕЗ ПІДСИСТЕМИ ПОЗИЦІОНУВАННЯ LeGNSS	79
3.1 Розробка інтегрованої підсистеми LeGNSS.....	79
3.2 Вимоги до робастності та надійності системи.....	81
3.3 Діагностика та компенсація збоїв у підсистемі LeGNSS	85
3.4 Дискретний фільтр Калмана та інтелектуальний алгоритм корекції ...	88
3.5 Використання псевдореальних даних для моделювання та валідації підсистеми	94
3.6 Висновки до третього розділу	95
РОЗДІЛ 4. ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ LeGNSS.....	97
4.1 Схема синтезованої системи в Matlab	97
4.2 Аналіз синтезованої схеми підсистеми позиціонування	99
4.3 Створення польотного завдання та сигналів керування БпЛА.....	102
4.4 Моделювання та дослідження роботи синтезованої схеми.....	121
4.5 Висновки до четвертого розділу	131
ВИСНОВКИ.....	132
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	136
ДОДАТОК А. Акти впровадження результатів роботи.....	161
ДОДАТОК Б. Свідоцтво про реєстрацію авторського права	165
ДОДАТОК В. Державна реєстрація винаходу.....	166
ДОДАТОК Г. Код програми моделювання роботи підсистеми позиціонування LeGNSS.....	167
ДОДАТОК Д. Список публікацій здобувача	212

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

EMA	- алгоритм експоненційного рухомого середнього
GNSS	- глобальна система супутникового позиціонування
GSM	- глобальна система мобільного зв'язку
INS	- інерціальна система позиціонування
ISN/GNSS	- інерціально-супутникова система позиціонування
LeGNSS	- інерціальна багатосупутникова підсистема позиціонування
LEO	- супутники на низькій навколоземній орбіті
MEO	- супутники на середній навколоземній орбіті
MEMS	- мікроелектромеханічна система
OFDM	- технологія ортогонального частотного мультиплексування
PNT	- позиціонування, навігація та часові вимірювання
RMS	- середньоквадратичне відхилення
БпЛА	- безпілотний літальний апарат
БПП	- бортова підсистема позиціонування
КМУ	- Кабінет Міністрів України
НАНУ	- Національна академія наук України
РЕБ	- радіо-електронна боротьба

ВСТУП

Актуальність теми. Сучасний світ зіткнувся із інтенсивним розвитком безпілотних систем та стрімким зростанням кількості їх експлуатантів. Сьогодні такі системи експлуатуються в комерційних, рекреаційних та безпекових цілях. Насичення повітряного простору безпілотними літальними апаратами (далі – БПЛА) висуває нові вимоги до точності позиціонування та якості їх систем обробки інформації, в тому числі в умовах складної міської забудови та повітряному просторі. Це змушує науковців шукати способи вдосконалення систем позиціонування та покращення технічних характеристик БПЛА для виконання складних завдань.

Зростання кількості джерел та обсягів даних, їх обробка в реальному часі вимагає вдосконалення існуючих систем та компонентів дронів. Сучасна наука зосереджена на розробці нових методів та засобів підвищення точності обробки інформації в бортових підсистемах БПЛА, що забезпечують швидкість, надійність та ефективність обробки інформації щодо позиціонування в просторі.

Вагомий внесок у вирішенні теоретичних і практичних задач обробки інформації про позиціонування внесли українські науковці В. Мелешко, В. Грабчак, О. Нестеренка, американські – Х'ю Елкайм, М. Брайсон, С. Суккаріє, Когуре, Т. Хамфріс, канадські – М. Ельгамраві, М. Караїм та ін.

Відомо, що інтеграція різних систем позиціонування може значно підвищити точність визначення параметрів руху. Проте, несумісність між системами та різна частота оновлення інформації може викликати додаткові похибки в обробці інформації.

Використання інерціально-супутникових методів в сучасних бортових підсистемах БПЛА вимагає глибокого розуміння їх взаємодії та інтеграції. Це стає особливо важливим, коли мова йде про обробку інформації для точного позиціонування безпілотних апаратів. В цьому контексті, виникає актуальне наукове завдання в галузі обробки інформації щодо позиціонування об'єкта в просторі: розробка методів та моделей для підвищення ефективності бортових

підсистем позиціонування БпЛА в умовах різних викликів.

Дана робота спрямована на вивчення та розробку нових підходів підвищення ефективності безпілотних систем, зокрема, на базі інноваційних технологій та компонентів. Основна задача дослідження полягає у створенні оптимальних алгоритмів обробки даних, які б дозволили забезпечити високу продуктивність та надійність роботи комп'ютерних систем.

Однією з ключових задач обробки інформації про позиціонування БпЛА при використанні множинних джерел даних є розробка комплексних підсистем позиціонування. Це досягається за допомогою методів компенсації або оптимальної фільтрації похибок вимірювань, що дозволяє отримувати дані про положення та забезпечувати вищу стабільність роботи бортових підсистем. Фільтр Калмана є механізмом для оптимального злиття даних з кількох джерел.

Дана робота пропонує підхід “багатосарової системи систем”, який обробляє інформацію з різних джерел, включаючи супутники низької земної орбіти (LEO), системи глобального позиціонування (PNT) та автономні інерціальні підсистеми користувача (INS). Даний комбінований підхід дозволяє побудувати модель робастної інерціальної багатосупутникової підсистеми позиціонування (LeGNSS) для БпЛА. Робастність обробки інформації про позиціонування характеризує здатність підсистеми адаптуватися, бути стійкою та забезпечувати надійну обробку інформації в умовах невизначеності, або впливу зовнішніх перешкод, мінімізувати вплив помилок у вхідних даних і гарантувати її безперебійну роботу.

Ефективність роботи запропонованого рішення оцінено з використанням математичного моделювання, на наближених до реальних вихідних даних та в динамічних умовах польоту БпЛА. Аналіз похибок показав, що запропонований підхід перевершує традиційні системи GNSS, особливо в складних умовах експлуатації.

Зв'язок роботи з науковими програмами, планами, темами. Дослідження та розробки, узагальнені у роботі, виконувались у рамках науково-дослідної роботи НДР №54-2022/09.01.04 (протокол Кафедри КСМ НАУ №13 від

07.09.2023 р.) «Технології створення високопродуктивних захищених комп'ютерних систем» (2023 р.). Слід зазначити, що вибраний напрямок досліджень має тісний зв'язок із науковим напрямом 1.2.11.2. «Методи обробки супутникових даних» затвердженим в Постанові Президії НАН України від 10.01.2024 р. № 8 «Про Основні наукові напрями та найважливіші проблеми фундаментальних досліджень у галузі природничих, технічних, суспільних і гуманітарних наук Національної академії наук України на 2024-2028 роки».

Мета і завдання дослідження. Метою роботи є підвищення точності обробки інформації щодо позиціонування в підсистемах, що використовуються в сучасних БпЛА, які працюють за рахунок більш точної оцінки похибок інформації отриманої від супутників в складних умовах експлуатації.

Для реалізації визначеної мети було поставлено такі **завдання**:

- аналіз методів та засобів обробки інформації про позиціонування, що використовуються в підсистемах БпЛА;
- обґрунтування вибору оптимальних методів та засобів підвищення точності обробки інформації про позиціонування в підсистемах БпЛА;
- розроблення методу злиття інформації в багатосупутникових підсистемах обробки інформації про позиціонування з використанням інформації від LEO-супутників;
- удосконалення методів оцінки сигналів похибок в компонентах підсистеми прийому та обробки даних;
- удосконалення методів визначення позиціонування з фільтрацією високочастотних похибок та адаптацією до постійних похибок, а також здійснення контролю за цілісністю даних, що надходять до підсистем БпЛА;
- удосконалення методики урахування похибок багатосупутникових систем, що виникають у складних умовах функціонування;
- розроблення методів та засобів комп'ютерного моделювання процесу обробки даних про позиціонування із підвищеними характеристиками точності;
- вдосконалення методики моделювання інерціальної та багатосупутникової системи позиціонування з оптимальною фільтрацією;

– аналіз ефективності застосування розроблених методів та засобів шляхом оцінки точності обробки даних про позиціонування БПЛА в складних умовах функціонування.

Об'єктом дослідження є процес обробки інформації про позиціонування в бортових підсистемах БПЛА.

Предметом дослідження виступають методи та засоби обробки інформації, спрямовані на підвищення точності та покращення якості інформації про позиціонування в бортових підсистемах БПЛА.

Методи дослідження. При проведенні досліджень було використано наступні методи проведення досліджень:

– при проведенні аналізу моделей підсистем позиціонування, що впливають на ефективність їх функціонування – методи спостереження та порівняння;

– при постановці наукового завдання та виборі методів його вирішення – методи порівняння, теорії ефективного інструментального забезпечення, математичного моделювання та теорії первинних і вторинних системних перетворень інформації;

– при розробленні методів та моделей підвищення ефективності підсистем позиціонування в умовах невизначеності – методи математичного аналізу і синтезу, теорії організації та вдосконалення систем обробки інформації, математичного моделювання, дискретної оптимізації, теорії статистичних оцінок та теорії статистичного моделювання.

Інформаційну базу дослідження становлять результати досліджень зарубіжних та вітчизняних науковців щодо оцінювання методів обробки інформації в підсистемах БПЛА та точності систем позиціонування.

Наукова новизна одержаних результатів полягає в наступному:

уперше:

– розроблено метод злиття інформації для багатосупутникової обробки інформації про позиціонування на базі методів оптимальної обробки інформації та прямої корекції помилок з використанням інформації від LEO-супутників,

який дозволяє підвищити стійкість процесу обробки інформації та точність визначення позиціонування;

удосконалено:

– методику оцінювання інформації за одним вимірювальним каналом про позиціонування, який, на відміну від відомих підходів, враховує оцінку значення параметрів шляхом злиття інформації від LEO-супутників та відомих бортових підсистем позиціонування БпЛА (GNSS та INS) та дозволяє в реальному часі обробляти надлишкову інформацію, а також отримувати точні дані про позиціонування (похибка позиціонування $\sim 3,12$ м, похибка визначення швидкості $\sim 0,2$ м/с);

набули подальшого розвитку:

– застосування методу комплексної обробки інформації з оптимальною та адаптивною фільтрацією похибок позиціонування з використанням програмних алгоритмів відфільтрування високочастотних похибок та адаптації до впливу постійних похибок, який на відміну від існуючих методів відрізняється тим, що враховує пікові відхилення похибок по точності позиціонування (підвищено точність на 9 %);

– застосування методу комплексного оброблення інформації в підсистемах позиціонування БпЛА, який на відміну від відомих відрізняється тим, що враховує оцінку цілісності даних в багатосупутниковій підсистемі позиціонування LeGNSS за допомогою розробленого алгоритму оцінки цілісності отримуваної інформації та дає можливість здійснювати постійний контроль за даними, що обробляються в підсистемі позиціонування БпЛА (перехід в робочий режим роботи за 2-5с після ініціалізації).

Практичне значення отриманих результатів полягає у наступному:

– Синтезовано математичну модель інтегрованої LeGNSS, яка підвищує стійкість процесу визначення позиціонування БпЛА та дає більш точні їх значення, з часом перехідного процесу менше 5 с;

– Синтезовано математичну модель комплексної обробки інформації в LeGNSS з оптимальною фільтрацією похибок вимірювань, що підвищує

робастність визначення параметрів позиціонування за рахунок відфільтрування високочастотних похибок та адаптації до постійних похибок, що дозволило обмежити пікові відхилення точності;

– Запропоновані схеми комплексування, які дозволяють в реальному часі обробляти надлишкову інформацію в БПЛА та отримувати точні дані про позиціонування з похибкою $\sim 3,12$ м, та про швидкість з похибкою $\sim 0,1-0,2$ м/с;

– Процес комплексного оброблення інформації в LeGNSS виходить на робочий режим роботи менше ніж за 1 с, що відповідає сучасним вимогам;

– Синтезовані схеми реалізовано у середовищі пакету математичного моделювання Matlab, що дає можливість проводити широкий спектр досліджень властивостей запропонованих схем та алгоритмів;

– Запропоновані схеми комплексного оброблення даних про положення дозволяють застосувати супутники низької земної орбіти, призначених для надання послуг швидкісного Інтернет зв'язку, через використання технологій мобільного супутникового швидкісного обміну даними, що робить їх оптимальними для забезпечення одночасного точного позиціонуванні та обміну інформацією без обмежень по відстані до пункту управління;

– Результати дисертаційного дослідження впроваджено: 1) для оцінки точності обробки інформації про позиціонування під час реалізації проєктів розробки автономних безпілотних системи в Громадській Спільноті «Бізнес Інкубатор Груп. Україна» (акт впровадження від 06.10.2023); 2) для підвищення ефективності оцінки позиціонування і діагностики точності її обробки у Громадській Організації «Всеукраїнський Центр Реформ Транспортної Інфраструктури» (акт впровадження від 17.10.2023); 3) рекомендовані до застосування під час розроблення, виготовлення, випробування, забезпечення процесів комп'ютерного моделювання для сучасних космічних та безпілотних літальних апаратів в Асоціації «КОСМОС» (акт впровадження від 24.10.2023); 4) для підготовки авіаційних фахівців авіаційних підприємств щодо особливостей використання підходу побудови багато-супутникових систем позиціонування, які входять до Громадської Спільноти «Українська Авіатранспортна Асоціація» (акт впровадження від 12.10.2023).

Отримані результати можуть бути використані під час проектування нових та модернізації існуючих систем позиціонування БпЛА різного призначення, а також для підвищення ефективності та безпеки їх експлуатації.

Особистий внесок здобувача. Теоретичні обґрунтування, практичні рекомендації, висновки та надані пропозиції в роботі отримані здобувачем у процесі дослідження самостійно. В роботах підготовлених у співавторстві, автору належить наступне: у роботі [1] було проведено визначення параметрів елементів для корекції комплексної інерціально-супутникової системи позиціонування із корекцією; у роботі [2] виконано синтез схеми комплексної інерціально-супутникової системи позиціонування з корекцією в середовищі Simulink, проведено дослідження похибок системи у динамічному режимі роботи; у роботі [3] описано типові похибки оцінки параметрів інтегрованої системи позиціонування; у роботі [4] проведено дослідження параметрів точності інтегрованої системи позиціонування, а також аналіз похибок системи в статичному та динамічному режимах роботи; у роботі [5] розглянуто схему інтеграції інерціальної та супутникової систем позиціонування з одночасним використанням корекції та оптимальної фільтрації; у роботі [6] виконано аналіз оцінки ризиків обробки недостовірних даних в бортових системах; у роботі [7] виконано аналіз архітектури інтегрованої системи позиціонування на основі GNSS з фільтрацією Калмана та алгоритмом виявленням несправностей; у роботі [8] розглянуто підхід «багатошарової системи систем», що виконує обробку інформації з різних джерел, включаючи супутники на низькій навколосемній орбіті, наземні системи позиціонування та бортові датчики; у роботі [9] проведено дослідження робастної підсистеми LeGNSS яка інтегрує дані супутників LEO з GNSS та бортовими інерціальними системами; у роботі [10] запропоновано метод комплексної обробки даних позиціонування БпЛА, що включає алгоритм компенсації похибок та оптимальну фільтрацію інформації.

Дисертаційна робота виконана в Національному авіаційному університеті на Кафедрі комп'ютерних систем та мереж Факультету комп'ютерних наук та технологій під науковим керівництвом доктора технічних наук, професора,

Заслуженого винахідника України, Жукова І.А.

Апробація результатів дисертації. Висновки, рекомендації та результати дослідження, отримані в дисертації, оприлюднені на наукових та науково-практичних конференціях, у тому числі міжнародних, всеукраїнських та за міжнародною участю: «Проблеми навігації і управління рухом» (Київ, 2012); «Політ. Сучасні проблеми науки» (Київ, 2012; 2013; 2015); «Сучасні проблеми авіакосмічних технологій та систем» (Житомир, 2013); «Aviation in the XXI Century. Aviation Security and Space Technology» (Київ, 2016; 2022); «Комп'ютерні системи та мережні технології» (Київ, 2023) та «Dependable Systems, Services and Technologies» (Афіни, 2023).

Отримано Свідоцтво про реєстрацію авторського права на комп'ютерну програму «Система адаптивної обробки інформації в багато-супутниковій системі позиціонування LeGNSS» (шифр «LeGNSS», державний реєстраційний номер №122052, від 18.12.2023). Подано заявку про державну реєстрацію винаходу – корисна модель «Спосіб підвищення точності позиціонування безпілотних літальних апаратів» (державний реєстраційний номер №u202305401, від 13.11.2023).

Публікації. Основні положення та результати дослідження викладено в 19 наукових працях, із них 7 статей опубліковано у виданнях, включених до переліку фахових категорії Б [1–2,4,6-8,10], 3 – в зарубіжних наукових виданнях, що входять до наукометричної бази Scopus [3,5,9]; 8 тез доповідей – у збірниках праць міжнародних науково-технічних конференцій різного рівня [11–19]. Отримано 1 авторське свідоцтво на комп'ютерну програму [20].

Структура та обсяг роботи. Дисертація логічно викладена та складається зі вступу, чотирьох основних розділів, висновків, списку використаних джерел та додатків. Повний обсяг дисертації 215 сторінок, із них 135 сторінок основного тексту. Робота включає 3 таблиці, 40 рисунків, 5 додатків. Список використаних джерел налічує 210 найменувань.

РОЗДІЛ 1

АНАЛІЗ СТАНУ ПИТАННЯ І ПОСТАНОВКА ЗАВДАНЬ ДОСЛІДЖЕННЯ

1.1 Науково-технічні проблеми точності позиціонування в БпЛА

У сучасному світі, де безпілотні літальні апарати (БпЛА) продовжують розширювати своє застосування, від виконання військових місій до використання у цивільних завданнях та наукових дослідженнях. Точність позиціонування в таких умовах стає критично важливою. Від неї залежать безпека польотів, ефективність виконання місій та надійність збору даних. Підвищення точності засобів позиціонування може сприяти розвитку нових застосувань БпЛА та забезпеченню їх безпечної та ефективної роботи.

Основні виклики для систем позиціонування включають вимоги до точності, залежність від зовнішніх сигналів, які можуть зазнавати зовнішніх перешкод, та високі вимоги до надійності обладнання. Точне позиціонування важливе для різноманітних застосувань БпЛА, включаючи аерофотозйомку, моніторинг навколишнього середовища, доставку вантажів та безпекові місії [21]. Точність впливає на безпеку, надійність та ефективність цих застосувань. Також важливими є питання від енергоспоживання та невеликих розмірів до функціональних компонентів підсистем позиціонування.

Атмосферні умови, такі як хмарність, дощ або пил при сильному вітрі, можуть впливати на точність позиціонування БпЛА [22–23]. Іншими джерелами перешкод сигналу супутників є високі будівлі або гірські масиви, лінії електропередач, масивні металеві інфраструктурні об'єкти, які також можуть вносити помилки в отриману інформацію системи позиціонування.

Ефективність обробки інформації про позиціонування характеризується наступними ключовими характеристиками:

– **Точність** [24] інформації є ключовим параметром, що визначає якість позиціонування. Вона вимірюється як близькість результатів вимірювань до

фактичного місцезнаходження об'єкта. Висока точність є критичною для багатьох застосувань, особливо у таких сферах, як авіація, геодезія, військові операції, та автономні системи.

– **Швидкість** [26–27] обробки інформації визначає, наскільки швидко система може обробити вхідну інформацію та надавати інформацію про позиціонування. Швидкість обробки є важливою для динамічних застосувань, де умови швидко змінюються, наприклад, при керуванні БПЛА у складних умовах або в автономних системах.

– **Цілісність** (надійність) [28–29] оцінює стабільність і точність системи позиціонування під час різних умов експлуатації. Вона стає особливо важливою у випадках, коли система піддається впливу зовнішніх перешкод (наприклад, електромагнітні завади, зміни атмосферного тиску). Підвищення цілісності досягається через застосування різних методів компенсації та фільтрації, а також через розробку алгоритмів, стійких до перешкод та помилок.

Якісні показники точності обробки інформації в підсистемах БПЛА охоплюють точність, швидкість обробки, та цілісність. Точність визначає, наскільки близько вимірювання відповідають фактичному положенню об'єкта, швидкість обробки вказує на спроможність системи швидко реагувати на зміни, а надійність відображає стабільність системи в різних умовах.

Ці вимоги стали рушійною силою до пошуку шляхів для об'єднання переваг і властивостей різних систем та засобів з метою отримати більш універсальні, точні і надійні системи позиціонування.

Проблеми у точності позиціонування включають обмеження поточних технологій, високу вартість точного обладнання, та виклики у інтеграції різних систем для забезпечення надійності та точності. Останні дослідження в області підвищення точності позиціонування включають використання оптичних датчиків, розробку нових алгоритмів для обробки даних та інтеграцію машинного навчання для покращення систем обробки інформації.

Розв'язання цих проблем передбачає не лише технічні нововведення, але й концептуальні зміни в підходах до позиціонування БПЛА. Наприклад, замість

традиційного підходу, що спирається на одиничні джерела інформації, вчені досліджують можливості створення гібридних складних систем, які комбінують супутникове позиціонування, інерційні датчики, та інші сенсори для досягнення вищої точності. Такий підхід дозволяє компенсувати слабкості однієї системи за допомогою переваг інших. Додатково, інтеграція машинного навчання дозволяє системам забезпечити самонавчання та адаптуватися до змінних умов експлуатації, значно підвищуючи ефективність і надійність позиціонування в складних та динамічних середовищах.

1.2 Огляд літератури

У сфері БПЛА активно досліджують та розширюють спроможності систем обробки даних під час польоту, що має вирішальне значення для підвищення точності та ефективності позиціонування в реальному часі для прикладного застосування дронів. Далі проведено огляд сучасних наукових статей та досліджень, які в сукупності проливають світло на останні досягнення і методи розвитку систем позиціонування для дронів, фокусуючись на оптимальній обробці інформації у реальному часі та високоточному позиціонуванні, включаючи інноваційні алгоритми, системну інтеграцію і практичне застосування, забезпечуючи комплексне розуміння поточного стану і майбутнього потенціалу розвитку підсистем позиціонування БПЛА.

У статті Аверіна, Боровицького та Микитенко розглядається зростаюча важливість БПЛА у різних секторах та супутні проблеми автоматизації їх польоту, зокрема під час зльоту та посадки. Автори підкреслюють критичну проблему: «Проблема полягає в забезпеченні високої точності позиціонування під час посадки та зльоту, що неможливо з одним лише GPS через його обмежену точність» [30]. Крім того, важкі камери зменшують корисне навантаження і збільшують вартість дрона. Це твердження підкреслює потребу точних системах позиціонування, крім традиційних GPS, які зазвичай пропонують точність в межах кількох метрів, чого недостатньо для певних місій.

У роботах Мелешко проведено поглиблений аналіз різних алгоритмів, що використовуються в INS системах, підкреслюючи їх різноманітність через відмінності в системах координат, рівняннях орієнтації, формах обчислення прискорення, методах інтегрування та типах використовуваних гіроскопічних датчиків. Ключовим моментом, який було зазначено, є те, що «алгоритми INS можуть суттєво відрізнятися залежно від типу використовуваних гіроскопічних вимірювань» [31], що підкреслює складність і варіативність розробки алгоритмів INS. У статті проведено детальний аналіз методів та технологій, що використовуються для підвищення точності та надійності безплатформних інерціальних систем. Окреслено роль сучасних методів цифрової обробки сигналів у поліпшенні характеристик таких систем. Розглядаються основні проблеми, пов'язані з мінімізацією помилок вимірювань, та методи їх подолання. Важливість точного калібрування та регулювання чутливих елементів системи підкреслюється як ключовий елемент у підвищенні точності та ефективності роботи безплатформних інерціальних систем. Також обговорюється обчислення вектору прискорення сили тяжіння для більш точного інтегрування основного значення акселерометрів.

У статтях В. Грабчака, П. Ванькевича та Ю. Іваника досліджено застосування методів варіаційного числення для опису руху «ідеального» об'єкта у вертикальній площині. Дане дослідження є модифікацією класичної задачі про брахістохрону, представляючи аналітичну залежність, що характеризує кінематичні характеристики руху «ідеальної» цілі у вертикальній площині [32]. Автори досліджують особливості шляху найменшого часу, надаючи новий погляд на оптимізацію траєкторії польоту. Підхід авторів пропонує розуміння оптимізації траєкторій ракети, враховуючи сили, що діють на об'єкт, і бажаний результат найкоротшого шляху.

А. Масієро та його колеги дослідили інтеграцію камери часу польоту з БпЛА для 3D-реконструкції в реальному часі в районах де сигнали GNSS недоступні. Вони підкреслюють що "запропонована система... забезпечує результати картографування з високою роздільною здатністю, з похибкою в

кілька сантиметрів" [33]. Топографічні моделі високої роздільності отримують за допомогою LiDAR та фотограмметрії. Вони використовуються для отримання точних моделей та більш доступних застосувань. Їх робота є важливим внеском у високоточне 3D картографування в реальному часі в складних умовах із використанням оптичного позиціонування.

Дільшад пропонує структуру для оцінки місцезнаходження БПЛА за допомогою обробки даних в реальному часі з візуальних датчиків адже "... визначення місцезнаходження є складним завданням, особливо в умовах обмежених можливостей GPS" [34]. Дане дослідження важливе своїм інноваційним підходом до визначення місцезнаходження БПЛА в районах, де відсутній GPS, що використовує легку нейронний мережу, яка обробляє дані візуального датчика в реальному часі. Модель використовується для виявлення об'єктів інтересу (наприклад, рекламного щита), після чого застосовується оптичне текстове розпізнавання (OCR) для видобутку контекстної інформації, яка подається до карти для визначення місцезнаходження БПЛА. У статті також наводиться набір даних для дослідження та порівняння різних методів виявлення просторових об'єктів та OCR, а також оптимізація моделі для використання на дронів.

Збір та обробка даних у реальному часі за допомогою системи БПЛА з підтримкою мобільних периферійних обчислень для збору і обробки даних в реальному часі, що була проведена науковцями Зенгоим і Таном, продемонструвала можливість інтеграції мобільних пристроїв в системи дронів. Автори зазначають що "... побудували нову модель швидкого збору та обробки даних для покращення характеристик обробки інформації у реальному часі" [35]. Активний та оперативний збір інформації має велике значення для розумних застосувань, таких як "розумна сітка". Для забезпечення оперативної обробки даних в реальному часі, авторами розроблено теоретичну модель швидкості збору даних. Результати симуляції свідчать про значні покращення продуктивності системи завдяки запропонованому дизайну та ефективності алгоритмів. Значення цієї роботи полягає в її внеску в оптимізацію ефективності

обробки даних в підсистемах БпЛА.

Зенг і Танг розширюють свої дослідження мобільних БпЛА з периферійними обчисленнями, акцентуючи увагу на обробці даних в реальному часі для різних місій БпЛА. Вони зазначають: "У цій статті досліджується парадигма збору даних, коли дрон динамічно збирає інформацію" [36]. Паралельно зі збором даних, БпЛА також аналізують навколишню інформацію в реальному часі. Для ефективною обробки даних в реальному часі, використовується мобільний розподіл обчислень на місці. У статті розглядається модель швидкості збору даних для забезпечення якості обробки в реальному часі. Автори розробили алгоритми для мінімізації споживання енергії та часу виконання місії. Ця стаття є ключовою в демонстрації адаптивності систем БпЛА до різноманітних потреб в обробці даних в реальному часі.

Високоточне позиціонування і обробка даних в режимі реального часу в роботах Ріке аналізують методи інтеграції кінематичного позиціонування в реальному часі в БпЛА. Вони стверджують: "Описаний підхід до позиціонування має потенціал для досягнення точності позиціонування 1–3 см" [37]. Високоточне позиціонування застосовується в сільському господарстві для виявлення дефектів та оцінки живлення культур. Орторектифікація аерознімків може бути автоматизованою, використовуючи дані позиціонування мікро-БпЛА. Мікро-БпЛА можуть використовувати RTK з супутниковими системами GNSS для покращення точності. Це дослідження важливе тим, що воно спрямоване на підвищення точності і надійності систем позиціонування БпЛА, особливо в чутливих до точності додатках.

Ріеса Сакіра обговорюють розгортання БпЛА в якості периферійних вузлів для обробки даних в режимі реального часу в промисловому інтернеті речей. У статті підкреслюється: "Результат показує, що використання машинного навчання... може збільшити кількість автономних систем, які можуть надавати послуги" [38]. Використання Deep Q-Network розглядається як метод оптимізації процесу обробки даних. Дослідження спрямоване на використання цих методів в сфері промислового Інтернету речей. Це дослідження має важливе значення

для демонстрації інтеграції БпЛА в промислових додатках з потребами в даних в реальному часі.

Відстеження повітряних цілей у реальному часі для систем протидії БпЛА з використанням OpenCV, що пропонують Мохамед Деведар та команда представляють систему відстеження безпілотників з використанням OpenCV засобів для відстеження цілей в реальному часі. Вони зазначають: "Точне відстеження цілей є важливою вимогою в системі захисту безпілотників при постановці завдань" [39]. Для оцінки ефективності системи використано систему комп'ютерного зору OpenCV для розробки системи відстеження БпЛА. У проведеному аналізі використано порівняльну оцінку та параметри точності та часу. Багато інтегровані рішення пропонують вищу стійкість систем позиціонування до RF-блокування, спуфінгу GPS, зниження впливу завдань GPS, тощо. Наведена стаття є важливим внеском у забезпеченні інструментів позиціонування в реальному часі та захисту БпЛА.

Адаптивне і робастне управління надвисокої роздільної здатності в реальному часі для інтелектуальних платформ спостереження, що розглядаються Юнгом і Кімом пропонують новий алгоритм для широкомасштабного спостереження за гаванями за допомогою дронів. Пропонується новий алгоритм для великомасштабного спостереження в глибоко адаптованих гаванях та логістичних платформах за допомогою безпілотників та алгоритму суперспостереження. Автори стверджують: "Цей алгоритм призначений для максимізації середньо часової продуктивності за стабільних умов польоту..." [40]. Алгоритм включає в себе планування між кількома дронами та багатьма системами розподілених обчислень на основі границь. Результати симуляцій підтверджують, що запропонований алгоритм досягає заданого рівня продуктивності. Дослідження є важливим для його застосування у підвищенні стабільності та продуктивності у завданнях спостереження при відсутності складних зовнішніх умов, що впливають на об'єкт.

Юфан Ляо та Д. Джуанг представляють систему на базі дрону для виявлення морського сміття в режимі реального часу, яка має на меті замінити ручні методи

моніторингу. В минулих дослідженнях було використано Інтернет речей і датчики для моніторингу, але вони не враховували потокове відео та аналіз зображень. Вони підкреслюють: "Було показано, що БПЛА здатний літати по задалегідь визначеному шляху і виявляти сміття в прибережних районах" [41]. Система, що була використана для виконання моніторингового завдання, дозволяє заощадити людський ресурс при спостереженні за поверхнею узбережжя. Стаття важлива своїм інноваційним підходом до екологічного моніторингу, що демонструє потенціал БПЛА в обробці даних в режимі реального часу для екологічних цілей.

В дослідженні методу високоточного позиціонування наземних цілей у режимі реального часу за допомогою безпілотного стереобачення та об'єднання просторової інформації Пінг Ванг з колегами пропонують метод високоточного позиціонування наземних цілей за допомогою камер стереобачення і злиття просторової інформації [42]. Позиційна точність включає багато факторів, таких як положення, стан польотної платформи, точність вказівки стабільної платформи та різні координатні перетворення. Розумна стратегія злиття даних гарантує стабільну точність позиціонування. У статті пропонується модель розташування на основі стереобачення та метод об'єднання просторової інформації. Модель інтегрує візуальну інформацію, супутникову позиційну інформацію та географічну інформацію, що значно підвищує точність позиціонування. запропонований метод виявляє значне покращення порівняно з іншими традиційними методами позиціонування на землі за допомогою стереобачення та об'єднання просторової інформації. У статті підкреслюється підвищення точності позиціонування і можливості роботи в реальному часі, що демонструє значний прогрес у побудові моніторингових систем на базі БПЛА.

Оцінка точності вимірювань кінематики в режимі реального часу для прямої гео-прив'язки в статті Деста Екасо та ін. оцінюють точність позиціонування RTK GNSS, зосереджуючись на їх застосуванні для прямої гео-прив'язки [43]. Завдяки тестовим польотам було встановлено, що трьохвимірна точність позиціонування для фотограмметричних продуктів коливається між 30 і 60 см. Геопросторова

інформація, зібрана за допомогою БпЛА, надає підтримку прийняттю рішень у багатьох областях та вимагає постійного розвитку технологій для отримання більш точних результатів. Матеріал також розглядає важливість компонентів системи БпЛА, включаючи GNSS приймач, INS та камеру, на якій базується якість фінального картографічного продукту. Дослідження дає підхід оцінки інформації про точність і потенційне застосування технології RTK в системах БпЛА.

Архітектура глибоких нейронних мереж DCNN-GA для позиціонування БпЛА в закритих приміщеннях запропонована в роботах П. Чхікара та команди представляє нову архітектуру глибокої нейронної мережі, оптимізовану за допомогою алгоритмів машинного навчання та генетичних алгоритмів, для автономного позиціонування в закритих приміщеннях. Традиційно GPS використовувався для позиціонування БпЛА, але в закритих приміщеннях ця система не працює. Автори підкреслили важливість подолання обмежень GPS в приміщенні, акцентуючи на тому що "через роботу в приміщенні система глобального позиціонування не може забезпечити точну локалізацію і позиціонування" [44]. Важливим завданням для автономного позиціонування БпЛА в закритих приміщеннях є уникнення перешкод та точна локалізація, оскільки GPS не працює в даних умовах. Це дослідження вирішує проблему позиціонування в приміщеннях, що є ключовим аспектом у розширенні застосування БпЛА за межі традиційних зовнішніх середовищ.

Розглянуті дослідження в сукупності підкреслили важливість ефективної обробки даних у реальному часі для підвищення точності та функціональності роботи підсистем позиціонування дронів. Від позиціонування в приміщеннях до високоточного позиціонування та інтеграції для вирішення широкого спектру інноваційних та практичних застосувань БпЛА. Фокус на подоланні обмежень стандартних підсистем GNSS, оптимізація ефективності обробки даних і адаптація до різних потреб у даних в режимі реального часу підкреслює постійний розвиток і потенціал систем БпЛА в різних середовищах і сферах застосування.

Крім того, інтеграція передових технологій, таких як глибокі нейронні мережі, нові супутникові сузір'я, мобільні периферійні обчислення і алгоритми машинного навчання, в системи дронів демонструє собою значний крок вперед у цій галузі. Ці досягнення не лише підвищують точність і ефективність застосування БпЛА, але й розширюють їх застосування в складних місцях, в тому числі в закритих приміщеннях, на територіях, де немає сигналів GNSS. Поєднання цих технологій прокладає шлях до більш надійних, універсальних і інтелектуальних підсистем позиціонування, здатних задовольнити зростаючі потреби різних секторів і зробити значний внесок у розвиток застосування технології точного позиціонування БпЛА.

1.3 Призначення та будова сучасних БпЛА

БпЛА стали незамінними помічниками у виконанні складних і ризикованих завдань. Вони не тільки дозволяють проводити моніторинг та збирати дані на відстані, але й передавати цю інформацію через телекомунікаційні та радіоканали, особливо у місцях, де присутність людини неможлива або небезпечна. Завдяки БпЛА можна ефективно контролювати стан різних об'єктів, їх роботу та потенційні загрози, навіть якщо ці об'єкти знаходяться далеко.

Для успішного використання безпілотників у таких завданнях важливо мати сучасні бортові системи та забезпечити точне дотримання запланованого маршруту польоту. Однією з переваг сучасних БпЛА є те, що вони часто ефективніші за пілотовані літаки. Це пояснюється відсутністю складних системах підтримки життєдіяльності пілота, що робить їх більш енергоефективними.

Сучасні БпЛА вражають своєю різноманітністю у формах, розмірах, конфігураціях та технічних характеристиках. Історично, БпЛА були простими дистанційно керованими літальними апаратами, але з часом все більше уваги приділяється розробці автономних систем.

Для забезпечення точного позиціонування та ефективного управління,

БпЛА оснащені різними технологічними підсистемами:

- Супутникова система глобального позиціонування (Galileo, GPS та інші);
- Системи радіо і телеметричної інформації;
- Системи командно-позиційного зв'язку;
- Система управління;
- Цифрові системи обміну інформацією;
- Обчислювальний процесор;
- Пристрої для зберігання інформації.

Загальна маса цих компонентів зазвичай не перевищує кількох кілограм, а також вони розраховані витримувати перевантаження і вібрації до 20 g, що робить їх ідеальними для різноманітних умов експлуатації.

Так, система радіо і телеметричної інформації відповідають за збір та передачу телеметричних даних, які включають висоту польоту, швидкість, стан батареї, та інші важливі параметри для керування БпЛА.

Система командно-позиційного зв'язку забезпечує двосторонній зв'язок між БпЛА та оператором. Вона дозволяє оператору відправляти команди та отримувати інформацію про стан і положення літального апарату в реальному часі, що є особливо важливим для місій з високими вимогами до реакції.

Система управління БпЛА включає програмне забезпечення та алгоритми, що дозволяють дрону полегшувати здійснення керування ним, або ж взагалі самостійно виконувати поставлені завдання. Це може включати автоматичне уникнення та обліт перешкод, стабілізацію польоту, автоматичне повернення на задану точку та інші функції керування БпЛА.

Бортові цифрові системи обміну інформацією відіграють ключову роль у забезпеченні обміну даними між різними компонентами БпЛА, такими як сенсори, камери, системи позиціонування та система управління. Вони дозволяють координувати роботу всіх підсистем дрона.

Обчислювальний процесор є центральним елементом автономної системи БпЛА. Він відповідає за обробку всіх даних, отриманих від сенсорів, підсистем позиціонування та інших джерел. Високопродуктивний обчислювальний

процесор необхідний для швидкого та ефективного виконання складних обчислень в динамічних умовах експлуатації.

Пристрій зберігання інформації вирішує завдання зберігання польотних та технічних даних, зібраних під час польоту. Це можуть бути фотографії, відеозаписи, телеметричні дані та інша важлива інформація для аналізу під час або після завершення місії.

Кожна з цих перелічених бортових підсистем відіграє важливу роль у забезпеченні функціональності та ефективності застосування БпЛА.

Сучасні БпЛА все частіше оснащуються передовими сенсорними системами, які підвищують їхні тактичні можливості. Ці системи включають камери високої роздільної здатності, інфрачервоні датчики та лазерно-вимірювальні технології LiDAR (Light Detection and Ranging). Ці сенсори дозволяють збирати детальні зображення та дані навіть у умовах низької видимості, що робить їх незамінними для завдань, таких як моніторинг навколишнього середовища, пошуково-рятувальні операції та збирання аграрних даних.

Системи руху сучасних БпЛА також значно удосконалилися. Електричні двигуни, що живляться від батареї, є поширеними в малих БпЛА, пропонуючи тиху роботу та низькі витрати на обслуговування, разом з тим, для більших систем розробляються гібридні силові установки, що поєднують електричні двигуни з традиційними двигунами внутрішнього згоряння, щоб розширити дальність польоту та вантажопідйомність.

БпЛА обробляють чутливі дані та працюють у різних середовищах, надійні системи зв'язку та безпеки даних є життєво важливими. Сучасні БпЛА використовують зашифровані канали передачі даних для забезпечення безпечної передачі інформації. Крім того, часто впроваджується резервування в системах зв'язку для підтримки контролю та потоку даних навіть у разі відмови однієї системи або в разі діє систем РЕБ.

Технологія застосування рою БпЛА, де група дронів працює разом у координований спосіб, є новим трендом розвитку застосування технологій.

Даний підхід дозволяє виконувати складні завдання більш ефективно та надійно. Рої дронів можуть одночасно охоплювати великі території, забезпечувати швидше збирання даних та адаптуватися до динамічних умов експлуатації.

Сучасні БПЛА розробляються з урахуванням нормативних вимог та сучасних стандартів безпеки польотів. Це включає алгоритми автоматичного повернення додому, системи уникнення зіткнень, та встановлення заборон і обмежень на польоти, щоб запобігти роботі в обмежених зонах. Крім того, важливим напрямком розвитку є можливістю модульної кастомізації, що дозволяє користувачам адаптувати системи під конкретні місії. Це може включати заміну сенсорних систем, підвісного корисного навантаження, зміну батарей або навіть заміну систем управління для підвищення функціональності БПЛА.

1.4 Аналіз існуючих інструментальних засобів позиціонування

Позиціонування БПЛА засноване на визначенні їх місцезнаходження та орієнтація в просторі. Для цього використовуються різноманітні технології, включаючи GPS, INS, лідари, радари, магнітні компаси, оптичні системи, тощо [45–49]. Кожна з цих технологій має свої переваги і обмеження, які наведено в таблиці 1.1, і часто вони використовуються у комбінації для підвищення точності. Ефективне позиціонування як правило вимагає інтеграції даних з цих різних джерел для забезпечення точності, надійності та стійкості системи позиціонування [50–53].

До сучасних БПЛА висуваються високі вимоги до глобальності, незалежності від гідрометеорологічних умов, підстилаючої поверхні, рельєфу рослинності, що оточує, забудови, часу доби й року, безперервності роботи, захищеності від завад, незалежності від висоти над поверхнею землі й інших умов руху об'єкту позиціонування та ін.

Таблиця 1.1

Порівняння різних систем позиціонування для БПЛА

Тип	Основні переваги	Основні недоліки	Застосування
GNSS	Висока точність, глобальне покриття, простота, розміри	Залежність від якості сигналу, проблеми у внутрішніх просторах	Велика дальність місій, загальне позиціонування
LEO	Висока точність, глобальне покриття	Залежність від сигналу супутника	Зв'язок із центром керування, загальне позиціонування
INS	Автономність, висока швидкість реакції, повнота параметрів (прискорення, швидкість, координати, кути положення, тощо)	Накопичення похибок з часом, потреба в періодичному калібруванні	Короткочасне позиціонування, позиціонування без корекції
Лідари	Висока точність у вимірюванні відстаней, висока роздільна здатність	Обмежена дальність, висока вартість	Топографічне картографування, точне землеробство
Радари	Здатність працювати в будь-яких погодних умовах, велика дальність дії	Менша точність порівняно з лідаром, потенційні проблеми з визначенням дрібних об'єктів, висока вартість	Пошуково-рятувальні операції, військові застосування

Тип	Основні переваги	Основні недоліки	Застосування
Магнітні компаси	Простота використання, низька вартість	Обмежена функціональність, вразливість до магнітних полів	Прості завдання позиціонування, основний орієнтир у просторі
Оптичні системи	Хороша точність у визначенні місцезнаходження, здатність до візуального відстеження	Потреба у великій кількості обчислювальних ресурсів, проблеми при низькій освітленості	Детальне картографування, промислові інспекції

Дані засоби отримання інформації складаються з різних компонентів, які включають сенсори (наприклад, GPS, інерційні мірники), процесори для обробки даних, та комунікаційне обладнання для передачі даних. Ці системи працюють, використовуючи сигнали супутників або інші засоби для визначення місцезнаходження, швидкості та напрямку руху БпЛА.

Серед них GNSS виокремлюється своєю універсальністю, тоді як інерційні системи відзначаються високою незалежністю. Оптичні системи забезпечують високу точність в обмежених умовах. Серед найбільш поширених технологій позиціонування для БпЛА виділяються GPS та INS. GPS забезпечує глобальне позиціонування за допомогою супутникових сигналів, тоді як INS використовується для внутрішнього позиціонування та керування польотом, базуючись на датчиках руху. Часто ці системи використовуються разом для отримання більш надійного та точного результату.

Точність GPS може варіюватися від декількох десятків метрів для загального каналу даних до кількох метрів для каналу підвищеної точності, залежно від умов експлуатації та обладнання. INS системи можуть забезпечувати високу точність позиціонування та вимірювання параметрів польоту на короткі

періоди, але швидко втрачають точність із часом. Комбінація обох систем дозволяє забезпечити більш стабільні та надійні дані.

З іншої сторони, відбувається зростання і ролі інтеграції альтернативних підсистем, таких як лідари, радары, магнітні компаси та оптичні системи, які є важливими в контексті підвищення точності позиціонування та керування. Вони доповнюють традиційні системи, такі як GNSS, LEO та INS, адже кожна з цих технологій має свої переваги та обмеження, залежно від застосування та умов експлуатації БпЛА.

Лідари використовуються для сканування навколишнього середовища за допомогою лазерних променів, забезпечуючи високу точність вимірювання відстаней і формування тривимірних карт місцевості. Це особливо необхідно для вирішення завдань обходу перешкод, топографічних вимірювань та аналізу місцевості, що робить лідар незамінним у геодезичних дослідженнях та при створенні точних карт рельєфу.

З іншого боку, радары використовують радіохвилі для визначення відстаней та виявлення об'єктів, що робить їх ефективними в умовах поганої видимості, таких як туман, дощ або нічний час. Вони також корисні для виявлення рухомих об'єктів та можуть використовуватися для систем уникнення зіткнень з іншими літальними апаратами.

Магнітні компаси хоча і є давньою та відносно простою технологією, є важливою для орієнтування БпЛА в просторі відносно магнітного полюса. Вони надають основну інформацію про напрямок, що є критично важливим для базового орієнтування, особливо в умовах, коли сигнали супутникових систем недоступні або нестабільні.

Оптичні системи, які використовують камери та спеціалізоване програмне забезпечення для визначення положення об'єктів, можуть забезпечувати важливу візуальну інформацію, яка може бути використана для уникнення перешкод, позиціонування або картографування.

Ці технології часто використовуються в комбінації з GNSS, LEO та INS для забезпечення кращого позиціонування та керування. GNSS, яка використовує

сигнали від супутників для визначення глобального положення, є основою для більшості систем позиціонування.

При порівнянні існуючих засобів позиціонування БпЛА важливо враховувати різні параметри, такі як точність, робочий дії, незалежність від зовнішніх умов та вартість. Майбутнє технологій позиціонування обіцяє ще більшу інтеграцію з штучним інтелектом та автономними системами. Розробляються нові методи, такі як оптичне позиціонування та більш ефективні алгоритми машинного навчання для обробки даних з різних джерел. Ці напрямки можуть суттєво змінити способи використання БпЛА в різноманітних сферах, від рятувальних операцій до доставки вантажів.

Сучасні технології позиціонування БпЛА постійно розвиваються, покращуючи точність та надійність. Аналіз ефективності цих систем показує, що інтеграція різних джерел даних, як-от супутникові, інерційні та оптичні системи, забезпечує найкращі результати, особливо в нестандартних умовах експлуатації.

1.4.1 Особливості інерціальної системи позиціонування

INS на основі MEMS датчиків є складною автоматизованою системою, яка використовується для точного вимірювання параметрів під час польоту, таких як положення, швидкість та прискорення об'єкта, на основі інерціальних розрахунків [54–57]. Мікроелектромеханічні системи, це маломасштабні системи, які складаються з мікромеханічних сенсорів, актуаторів та мікроелектронних схем. Ця система здатна обробляти отриману інерціальну інформацію та надавати її до інших підсистем БпЛА.

MEMS датчики забезпечують значні переваги у порівнянні з традиційними системами вимірювання координат через їх компактність, низьку вартість і прийнятну точність. Ці системи використовуються не тільки в БпЛА, але й у різних областях, включаючи автомобільну промисловість, мобільні пристрої та робототехніку. Сучасні технологічні досягнення в галузі MEMS систем дозволили значно зменшити розмір та вагу цих систем, роблячи їх ідеальними

для використання у легких і компактних БпЛА, де кожен грам ваги та кожен сантиметр простору мають критично важливе значення.

INS на основі MEMS датчиків є основою побудови сучасних дронів завдяки їх здатності до швидкої та точної обробки даних, зокрема в умовах, де GPS сигнал може бути недоступний або ненадійний. Це особливо актуально для польотів в урбанізованих зонах або в зонах з високим рівнем електромагнітних перешкод.

В INS використовується інерціальна система відліку, яка є нерухомою відносно точки старту, для точного вимірювання руху. Завдяки цьому, INS дозволяє визначати основні параметри руху різних об'єктів, таких як БпЛА, ракети, космічні кораблі, надводні та підводні судна, тощо.

Використання інерційної системи відліку забезпечує здатність переміщення БпЛА з високою точністю. Це стає можливим завдяки використанню високочутливих акселерометрів та гіроскопів, які вимірюють лінійні та кутові прискорення, дозволяючи точно визначати зміни положення і орієнтації об'єкта в просторі. Система працює на принципі моделювання поступального руху об'єкта, використовуючи зміни у прискоренні, швидкості та координатах, які фіксуються за допомогою вбудованих датчиків. Такі дрони підтримують високий рівень автономності та керованості, навіть у разі відсутності GPS-сигнали, під водою або в щільній міській забудові. Здатність INS системи забезпечувати безперервне відстеження положення, швидкості та орієнтації об'єкта робить її незамінною для місій, які вимагають високої точності та надійності.

Однією з ключових переваг INS перед іншими системами позиціонування є її автономність роботи, універсальність та висока стійкість до зовнішніх перешкод.

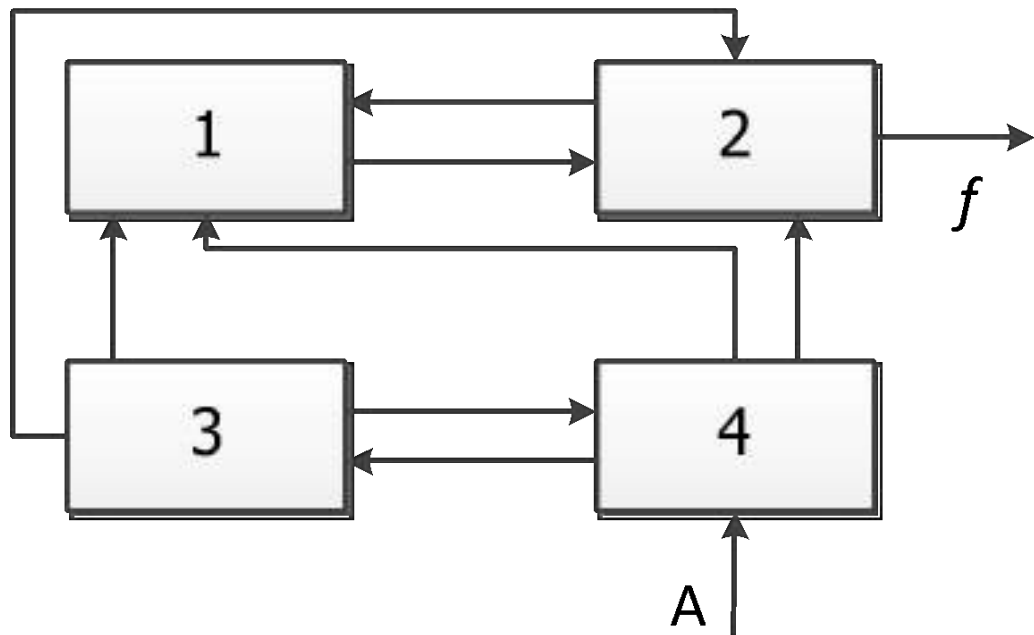


Рисунок 1.1. Структурна схема роботи системи INS.

На рис. 1.1. представлено структурну схему INS, що складається з наступних елементів:

- 1 – блок інерціальних вимірювачів (акселерометри та гіроскопи) для орієнтації та видачі даних;
- 2 – обчислювальний блок для інтеграції рівнянь руху та обчислення параметрів;
- 3 – блок часу для синхронізації системи;
- 4 – блок введення початкових даних для орієнтації акселерометрів;
- A – вхід для початкових даних;
- f – вихід для кінцевих даних про параметри руху.

Блок інерціальних вимірювачів складається з акселерометрів та гіроскопів. Акселерометри постійно вимірюють прискорення об'єкта, тоді як гіроскопи відстежують обертання та орієнтацію об'єкта в просторі. Ця інформація є критично важливою для визначення швидкості, напрямку руху та зміни положення об'єкта в часі.

Обчислювальний блок є цифровим мозком системи. Він інтегрує дані від інерціальних вимірювачів та використовує складні алгоритми обчислення

траєкторії руху, швидкості та орієнтації об'єкта в просторі. Обчислювальний блок виконує неперервну інтеграцію значення руху, що дозволяє визначити точне положення об'єкта в просторі протягом часу польоту, навіть при відсутності зовнішніх джерел інформації.

Для синхронізації роботи всіх компонентів INS та точності обчислень використовуються високоточний блок відліку часу. Він забезпечує синхронізацію між обчислювальним блоком та інерціальними вимірювачами, що є важливим для забезпечення точності даних.

Для ефективної роботи інерціальної системи перед її використанням необхідно ввести початкові дані. Для цього, перед початком роботи INS вводяться початкові дані про положення та орієнтацію в просторі, які допоможуть системі визначити її подальше положення та орієнтацію. Введення початкових даних включає координати точки старту, початковий азимут, висоту та інші параметри об'єкта. Блок введення початкових даних забезпечує початкове калібрування акселерометрів та гіроскопів, що є базовим для точності подальших вимірювань та розрахунків системи.

На виході система передає кінцеві дані про параметри руху об'єкта, які включають в себе позиціонування, швидкість та орієнтацію в просторі. Ці дані є результатом неперервної обробки інформації в INS, отриманої від інерціальних датчиків, та їх інтеграції обчислювальним блоком системи. Отримана інформація про позиціонування може використовуватися для різноманітних цілей, включаючи позиціонування, контроль руху, та інші автономні функції БпЛА. Дана інформація також може бути передана на зовнішні системи або бортові підсистеми для подальшого аналізу, інтегрування, зберігання або відображення.

1.4.2 Особливості роботи глобальних систем позиціонування

GNSS є складною електронно-технічною системою, що включає наземне та космічне обладнання яке дозволяє вирішити завдання визначення

позиціонування об'єкта. Дана система дозволяє визначати місцеположення, швидкість руху та точний час (PNT).

У сучасному світі GNSS представлена трьома основними глобальними системами позиціонування – GPS (США) [58–61], Galileo (Європейського Союзу) [60–65] та Beidou (Китаю) [66–69]. Дані системи є всепогодними та працюють завдяки мережі космічних супутників. Вони дозволяють визначати з високою точністю місцезнаходження, швидкість руху об'єктів та надають надійний часовий відлік часу на глобальному рівні, стаючи невід'ємною частиною сучасних технологічних пристроїв.

Кожна із наведених систем має свої унікальні характеристики та особливості роботи. Наприклад, GPS, як найдавніша та найбільш поширена система, відома своєю надійністю та рівномірним покриттям всіх континентів супутниками, а Galileo розроблялася як більш точна система з покращеними рішеннями для мінімізації затримки сигналу, особливо в урбанізованих місцевостях. Китайська Beidou розроблена щоб забезпечувати краще покриття в Азіатсько-Тихоокеанському регіоні, доповнюючи глобальне покриття GPS і Galileo. Комбінація різних систем в одному пристрої підвищує точність, надійність та доступність сигналів GNSS.

GNSS належать до класу багатопозиційних радіонавігаційних систем і використовуються для визначення просторового положення та вектора швидкості БпЛА [70]. В основі системи лежить мережа супутників, які служать як опорні точки для отримання інформації про позиціонування. Інформація про позиціонування БпЛА зазвичай визначається за допомогою вбудованого обладнання, яке дозволяє обчислювати не тільки положення в просторі, але й швидкість, а також визначати точний час в польоті.

Типова GNSS системи включає в себе такі основні компоненти (див. рис.1.2):

1. Орбітальне угруповання супутників, яке базується у космосі;
2. Контрольно-вимірювальні та керуючі комплекси, розміщені на землі;
3. Апаратура, яка використовується споживачами (БпЛА).

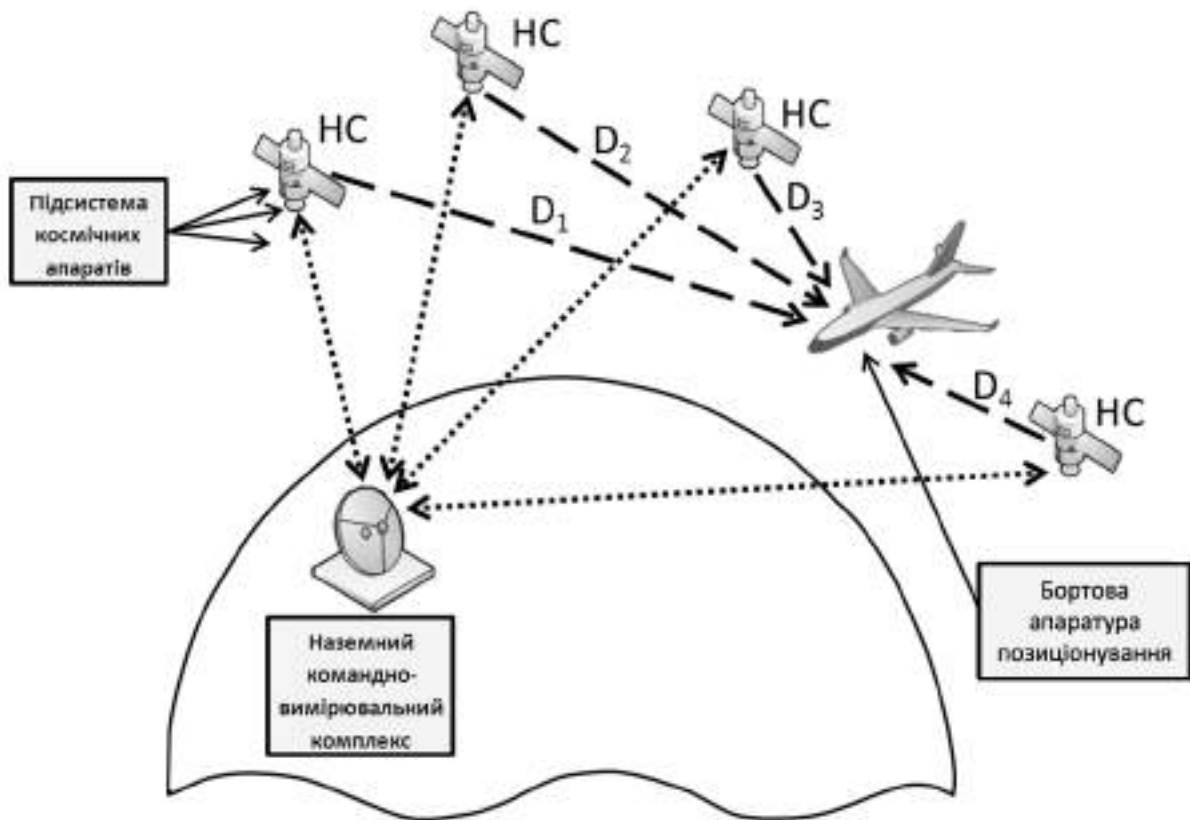


Рисунок 1.2. Загальна структурна схема роботи GNSS системи.

Орбітальне угруповання супутників розміщується в космосі на висоті близько 20 тис км від поверхні. даний компонент GNSS системи являє собою мережу супутників, які розміщені на зміщених орбітах навколо Землі. Кожен супутник оснащений високоточним атомним годинником та передавальними системами. Вони надсилають кодовані сигнали, що містять інформацію про час та їхнє місцезнаходження в просторі. Сигнали, випущені супутниками, використовуються для визначення точного місцезнаходження та часу на приймаючому пристрої. Орбітальне угруповання супутників розраховане таким чином, щоб у будь-який момент часу приймачу розміщеному на земній поверхні було доступно щонайменше декілька супутників, що забезпечує постійну високу точність та надійність роботи системи.

Важливою характеристикою GNSS супутників є їх здатність виконувати самокалібрування та корекцію похибок позиціонування ефемерид. Супутники постійно взаємодіють один з одним та з контрольно-вимірвальними станціями на Землі, що дозволяє точно визначати їхні положення в просторі та час,

забезпечуючи постійну стабільність роботи системи глобального позиціонування.

Контрольно-вимірювальні та керуючі станції розміщуються на землі та мають стаціонарні координати. Дані комплекси відіграють ключову роль у підтримці та керуванні супутниками та системою GNSS. Вони забезпечують збір даних від супутників, виконують їх обробку та аналіз. Ця інформація використовується для моніторингу стану орбітального угруповання, визначення точності їх сигналів та коригування будь-яких виявлених похибок супутників.

Крім збору та обробки даних комплекси також відповідають за керування супутниками. Вони надсилають керуючі та сервісні команди, проводять оновлення програмного забезпечення, забезпечуючи таким чином оптимальне функціонування системи. Таке керування супутниками може включати в себе коригування орбіт, налаштування орбітального обладнання, керування живленням, розгортання нового супутника або виведення старого із експлуатації. Ефективна робота наземних комплексів гарантує високу точність та надійність функціонування всього комплексу.

Кінцевим елементом GNSS систем є приймачі інформації кінцевих користувачів, які встановлюються на їх пристроях, таких як мобільні телефони або БпЛА. Ці приймачі розраховані на приймання сигналів від супутників, їх обробку та визначення з їх допомогою поточного місцезнаходження, швидкості та часу. Сучасні приймачі GNSS відрізняються високою точністю, компактними розмірами та енергоефективністю, що робить їх ідеальними для використання в бортових підсистемах позиціонування БпЛА. Сучасні приймачі GNSS часто оснащуються різними додатковими функціями, такими як автоматичне планування маршруту та управління польотом, що робить їх незамінними для застосування в БпЛА.

В цілому, сукупність трьох компонентів – орбітального угруповання супутників, контрольно-вимірювальних та керуючих комплексів на землі, та приймачів кінцевих користувачів – утворює багатофункціональну GNSS систему глобального позиціонування.

1.5 Основні методи підвищення точності обробки інформації в БпЛА

Проблеми у точності позиціонування включають обмеження поточних технологій, високу вартість точного обладнання, та виклики у інтеграції різних систем для забезпечення надійності та точності. Серед них – метод компенсації, який заснований на корекції сигналів з урахуванням відомих похибок та зовнішніх факторів. Метод фільтрації, зокрема застосування фільтра Калмана, дозволяє видаляти шум з вимірюваних даних, погіршуючи якість інформації про позиціонування. Оптимальна обробка інформації, включаючи методи як максимальної правдоподібності, спрямована на максимізацію точності на основі наявних даних.

В умовах високої швидкості переміщення БпЛА та впливу зовнішніх чинників, таких як атмосферні перешкоди чи електромагнітні завади, важливо вибирати методи обробки, які забезпечують надійність і точність. Метод компенсації дозволяє коригувати сигнали на основі заздалегідь відомих параметрів похибок, в той час як фільтрація ефективна для видалення випадкових шумів, які можуть значно змінюватися при швидкому переміщенні.

Існуючі підходи до підвищення точності обробки інформації в бортових підсистемах позиціонування БпЛА демонструють широкий спектр рішень, які використовують широкий спектр засобів протидії з перешкодам та завадам, такі як метод компенсації, найменших квадратів, фільтрації, оптимальної обробки інформації, метод максимальної правдоподібності та інші [71–73].

Інтеграція різних методів обробки може значно підвищити точність та надійність інформації про позиціонування. Наприклад, комбінування фільтрації з методами компенсації дозволяє збалансувати швидкість і точність обробки, адаптуючись до змін умов.

Серед найбільш перспективних на сьогоднішній день є різні схеми інтеграції INS та GNSS [74]. Вони включають вільно з'єднані, слабо пов'язані (Loosely Coupled Systems), тісно інтегровані (Tightly Coupled Systems) та глибоко інтегровані системи (Deep Integration) [75–76]. Вони також можуть поділятися за

типом зворотного зв'язку на відкриті та закриті системи. Зі збільшенням рівня інтеграції, як правило, зростає ефективність такої системи. Порівняння тісно інтегрованих та глибоко інтегрованих систем показує, що глибока інтеграція, хоча й вимагає розробки адекватних моделей для опису їх похибок, є більш ефективною [77].

Основна мета спільної обробки інформації в інтегрованих системах позиціонування полягає у зменшенні похибок при визначенні місцеположення та складових швидкості польоту БпЛА. Зниження похибок досягається за допомогою фільтрації, а точність забезпечується через якісну екстраполяцію.

Головним завданням постає визначення та аналіз оптимальних методів обробки інформації в підсистемах БпЛА, зокрема методів, які підвищують точність, швидкість та надійність інформації про позиціонування [78]. Дослідження сфокусовано на розробці алгоритму обробки даних, оцінку його ефективності в різних умовах експлуатації, та рішення для його впровадження в БпЛА.

При поєднанні різних джерел даних про позиціонування та використання декількох підходів обробки інформації про позиціонування при використанні методів компенсації та оптимальної фільтрації, можливо досягнути високої швидкості та надійності отримуваної інформації. Підвищена точність забезпечує надійне позиційне рішення, долаючи цифровий розрив, і має глибокі наслідки для секторів економіки, що залежать від точності позиціонування, включаючи авіацію, морські перевезення, наземні транспортні засоби та автономні дрони [79].

1.6 Концепція позиціонування з використанням низькоорбітальних супутникових сузір'їв

Сучасні дослідження в галузі позиціонування представляють нову глобальну систему позиціонування, яка використовує вимірювання доплерівського зсуву несучої з великих супутникових сузір'їв LEO, що є

альтернативою традиційним глобальним GNSS системам побудованих на основі псевдодальностей, таким як GPS, Galileo, Beidou, тощо. Даний підхід спрямований на подолання слабких сторін існуючих систем, таких як вразливість до підміни або глушіння сигналів через їх слабку силу і повільний час передачі, дороговартісний процес оновлення супутників в сузір'ях. Такі системи призначені для забезпечення 4-вимірного позиціонування, навігації та часу (PNT) шляхом пасивного прийому сигналів низхідної лінії зв'язку від цих великих LEO сузір'їв [80–82].

GNSS системи мають фундаментальне значення для сучасного позиціонування в просторі, але стикаються з такими проблемами, як багатопроменевість і перешкоди. Нещодавня поява великих супутникових LEO сузір'їв пропонує нові можливості для вирішення завдань позиціонування з такими перевагами, як вища потужність сигналу, краща доступність і сприятлива геометрія порівняно з супутниками на середній навколоземній орбіті (MEO), що використовуються в традиційних GNSS [83]. На відміну від традиційних GNSS, такі системи серед переваг мають вищу потужність, ширша смуга пропускання, швидка багатопроменева декореляція, а також потенціал для більш надійної автентифікації і нульовий час ефемерид, що обіцяє надати більшу точність і стійкість до перешкод і підробок сигналу [84–85].

Однак використання LEO супутників, особливо тих, що належать приватним операторам, таким як StarLink, пов'язане з певними проблемами, зокрема, не розкриття інформації про ефемериди супутників, синхронізацію годинників і специфікації структури сигналів [86–87]. В той же час, придатність сигналів StarLink для ширшого застосування в PNT і їхній потенціал, як повноцінної GNSS системи, залишаються відкритими питаннями, що залежать від специфіки сигналів що транслюються.

Даний метод не потребує попередніх знань про характеристики отриманого сигналу супутника, що є значною перевагою. Він включає в себе отримання та обробку частотного спектру прийнятого приймачем сигналу від групи супутників з урахуванням наявності високо динамічного каналу зв'язку між LEO

супутником і наземним клієнтом [88–89]. Фактично мова йде про так званий “сліпий” доплерівський детектор що аналізує та обробляє за заданим алгоритмом отриманий сигнал та оптимальний фільтр для очистки даних, як правило на основі фільтра Калмана [90–91]. Експериментальні результати показують, що цей метод дозволяє досягти високої точності відстеження доплерівського зсуву супутникових сигналів, що призводить до значного покращення точності позиціонування [92].

Такий підхід дозволяє використовувати доплерівський зсув несучої частоти, як єдину спостережувану величину. Такий підхід демонструє свою перспективність у точному вимірюванні доплерівського зсуву несучої за різними цифровими сигналами даних. Дана характеристика сигналу необхідна для адаптації сигналів StarLink, розроблених спочатку для широкосмугового доступу до Інтернету, для досягнення цілей PNT. Він відрізняється від роботи традиційного GNSS, який покладається на вимірювання псевдодальностей. Нова методологія потребує восьми або більше супутників для вимірювання доплерівських зсувів несучої для кожного передавача для точного визначення тривимірного положення, зсуву годинника приймача, швидкості та швидкості зсуву годинника приймача. Потреба в цих даних виникає тому, що всі ці невідомі наявні в будь-якій моделі доплерівського зсуву несучої частоти для отримання інформації про позиціонування [93]. Мета такої обробки інформації полягає в тому, щоб точно оцінити параметри сигналу на основі отриманих даних, та визначити символи, які застосовуються протягом захопленого часового інтервалу, а також ідентифікувати будь-які послідовності для синхронізації по часу в кадрі.

1.7 Перспективи та виклики розвитку систем позиціонування БпЛА

Незважаючи на значний технологічний прогрес, системи позиціонування БпЛА все ще стикаються з низкою серйозних викликів. Від недостатньої точності у складних умовах польоту, до проблем із забезпеченням безпеки та

конфіденційності, ці виклики вимагають глибокого аналізу та розробки нових рішень. Особливу увагу необхідно звертати на залежність БпЛА від зовнішніх сигналів GNSS [94], що створює ризики для стабільності роботи в умовах перешкод або збоїв сигналу.

Сучасні дослідження та розробки в області систем позиціонування зосереджуються на впровадженні новаторських технологій позиціонування, які можуть забезпечувати високу точність позиціонування без залежності від GNSS, використання штучного інтелекту для оптимізації польотних траєкторій, ці інновації відкривають нові можливості для підвищення надійності та ефективності БпЛА. Очікується, що наступні кроки у цій галузі будуть зосереджені на інтеграції з доповненою реальністю, підвищення автономності БпЛА та використанні більш складних алгоритмів обробки даних [95–97]. Це має забезпечити БпЛА здатність оперативно реагувати на зміни умов польоту та підтримувати високий рівень безпеки польотів.

Одним з основних сучасних викликів в питаннях позиціонування є зниження залежності від сигналів GNSS, що особливо актуально в умовах, де сигнали можуть бути заблоковані або підмінені. Для вирішення цього завдання розглядаються варіанти інтеграції альтернативних методів позиціонування, таких як візуальне позиціонування, використання систем астрономічного позиціонування та впровадження алгоритмів штучного інтелекту. Ці технології дозволяють БпЛА визначати своє місцезнаходження незалежно від зовнішніх сигналів, що забезпечує більшу стабільність та надійність у різних умовах експлуатації.

Іншим важливим аспектом є питання безпеки та конфіденційності. З огляду на все ширше використання БпЛА в цивільних та безпекових цілях, забезпечення захисту даних та запобігання несанкціонованому доступу є важливим. В цьому контексті важливою є розробка сучасних криптографічних методів та протоколів захищеного обміну даними. Також дуже актуальним є завдання виявлення та протидії спробам перехоплення інформації або спробам її підміни.

Алгоритми машинного навчання дозволяють БпЛА приймати рішення в реальному часі, адаптуватися до змін умов та навіть вчитися на даних з попередніх місій. Така автономія є важливою для виконання місій у віддалених або небезпечних районах, де ручне керування може бути обмеженим або неможливим.

Крім технічних аспектів, існує потреба у розробці міжнародних та державних регуляцій, що визначають параметри безпечного використання та експлуатації дронів. Це охоплює правила використання повітряного простору, вимоги до безпеки польотів та надійності обладнання, процедури обміну даними та відповідального управління польотами. Встановлення таких стандартів знизити ризики для громадськості та допоможе у створенні стабільного регуляторного середовища для подальшого розвитку та інновацій у даній сфері.

Збільшення частки комерційного використання БпЛА спонукатиме до розвитку нових нормативних та галузевих стандартів, покращення методів забезпечення безпеки польотів. Крім того, потрібно зосередити зусилля на створенні міцної, колаборативної взаємодії між розробниками, операторами БпЛА та регуляторними органами. Спільне обговорення та розробка галузевих стандартів та правил, які враховують інтереси усіх зацікавлених сторін, мають важливе значення для розвитку галузі. Це дозволить своєчасно реагувати на існуючі виклики та ефективно адаптуватися до швидко змінюваних умов та вимог сучасного світу.

1.8 Висновки до першого розділу

У першому розділі були проаналізовані та виявлені основні засоби та методи обробки інформації, що підвищують точність позиціонування БпЛА.

Значне зростання кількості БпЛА, які використовуються як у військових, так і в цивільних цілях, спонукає до необхідності їх систематичного покращення та підвищення їх характеристик. Завдяки сучасному науково-технічному прогресу, системи позиціонування використовують переваги механічних та

цифрових датчиків, які характеризуються надійністю та точністю.

Розглянуто різноманітні технології позиціонування, включаючи GPS, LEO, INS, лідари, радари та інші, та проаналізовано їх основні переваги та недоліки. Особлива увага приділена викликам, пов'язаним з точністю, залежністю від зовнішніх сигналів та вимогами до надійності обладнання.

Висвітлено ключові якісні показники точності, швидкості обробки та надійності, під час обробки інформації про позиціонування в БпЛА. Також розглянуто основні методи які використовуються для підвищення точності обробки інформації, включаючи методи компенсації, фільтрації та оптимальної обробки, тощо. Вказано на важливість комбінації різних підходів обробки інформації для досягнення оптимального результату. Основним напрямком підвищення точності позиціонування сучасних БпЛА є розробка інтегрованих інерціально-супутникових систем позиціонування.

Інтеграція різних методів та підходів у поєднанні із використанням сучасних засобів збору надлишкової інформації дозволяє значно підвищити точність та надійність позиціонування. Подальші дослідження та розробки у цій області відкривають широкі перспективи покращення функціональності та безпеки БпЛА.

РОЗДІЛ 2

МЕТОДИ ПІДВИЩЕННЯ ТОЧНОСТІ ТА ЕФЕКТИВНОСТІ ФУНКЦІОНУВАННЯ СИСТЕМ ПОЗИЦІОНУВАННЯ

2.1 Методи обробки інформації в системах позиціонування БпЛА

Переходячи від загального огляду сучасних систем позиціонування та їх застосування в БпЛА, ми підходимо до детального розгляду різних методів та підходів до обробки інформації. Ефективність використання інформації, що надходить до БпЛА, значно підвищується завдяки застосуванню різноманітних методів обробки цієї інформації. Серед ключових методів варто виділити оптимальну фільтрацію та оптимальне управління, які є фундаментальними для сучасних систем пілотажно-позиційного контролю під час польоту.

Сучасні системи позиціонування втілюють собою синтез високоточних інерціальних технологій та систем глобального супутникового позиціонування, що дозволяє досягти нових рівнів ефективності та точності. Ці підходи, як виявилось, є ключовими в забезпеченні точності, надійності та автономності сучасних БпЛА.

На рис.2.1 наведено класифікацію підходів до побудови інтегрованих систем позиціонування, що показала велику різноманітність існуючих методів та засобів [98], які можуть бути використані в побудові підсистем позиціонування. Існуючі схеми об'єднання GNSS та INS широко підтримується в різних застосуваннях, як перспективний та бюджетний підхід до підвищення точності та надійності, при розробці рішень точного позиціонування [99]. Такий підхід до обробки інформації про позиціонування, відомий як комплектування, та обробка надлишкової інформації, мають на меті підвищення точності систем позиціонування до необхідних рівнів. В сучасних умовах це є оптимальним рішенням для забезпечення високої точності позиціонування.



Рисунок 2.1. Загальна класифікація інтегрованих систем позиціонування.

Проте GNSS також має свої обмеження, особливо в умовах щільної міської забудови, де виникають проблеми з блокуванням сигналу та множинним шляхом розповсюдження сигналу. Для вирішення цих обмежень, розробники систем позиціонування інтегрують додаткові технології, такі як WiFi, Bluetooth, RFID, UWB (Ultra-Wideband) та інші. Наприклад, WiFi та Bluetooth можуть використовуватися для визначення позиції всередині будівель, де GNSS сигнали слабкі або відсутні. RFID та UWB технології можуть забезпечити високу точність позиціонування у специфічних умовах, наприклад, на складах та у виробничих цехах.

Як правило в більшості таких рішень розглядається використання методу злиття за допомогою фільтра Калмана для об'єднання інформації в системах із слабким зв'язком [100]. Крім класичних глобальних систем позиціонування, існуючі дослідження розглядають можливість використання геостаціонарних супутників, як джерело інформації для підвищення точності інформації про місцезнаходження об'єкта [101]. Крім того, інтегровані системи позиціонування відіграють важливу роль у розвитку розумних міст. Вони допомагають відстежувати рух транспорту, управляти міським освітленням, виконувати

моніторинг навколишнього середовища та надавати цінну інформацію для планування міського розвитку.

Існують навіть такі екзотичних рішення, як інтеграція систем GSM-орієнтування на поверхні землі [102], гідроакустичні системи підводного позиціонування у відкритому морі, які додатково підвищує надійність і стійкість роботи базових бортових підсистем позиціонування [103].

Такий підхід дозволяє зменшити вплив засобів постановки завдань у багаточастотних та багатосупутникових системах позиціонування [9]. Проте процес такої інтеграції часто є складним. Вирішення цих проблем має вирішальне значення для підтримання надійності і точності інформації про місцезнаходження. Розглянуто використання надлишкової інформації та удосконаленого методу обробки інформації для підвищення завадостійкості систем позиціонування [104].

Разом з застосуванням сучасних технологічних рішень, таких як розвиток 5G мереж, штучного інтелекту та машинного навчання, можливості для покращення та інновацій у сфері інтегрованих систем позиціонування збільшуються. Дані технології є багатообіцяючими та дозволяють зробити системи ще більш точними, надійними та доступними для ширшого кола застосувань.

2.2 Метод компенсації при обробці інформації про позиціонування

Метод компенсації став особливо популярним у процесі обробки пілотажної інформації [105–107], особливо в ситуаціях, коли один і той же параметр позиціонування фіксується кількома системами, що працюють на основі різних фізичних принципів. В таких випадках, алгоритм компенсації, який ефективно знижує помилки вимірювань, впроваджується згідно зі схемою, представленою на рис.2.2.

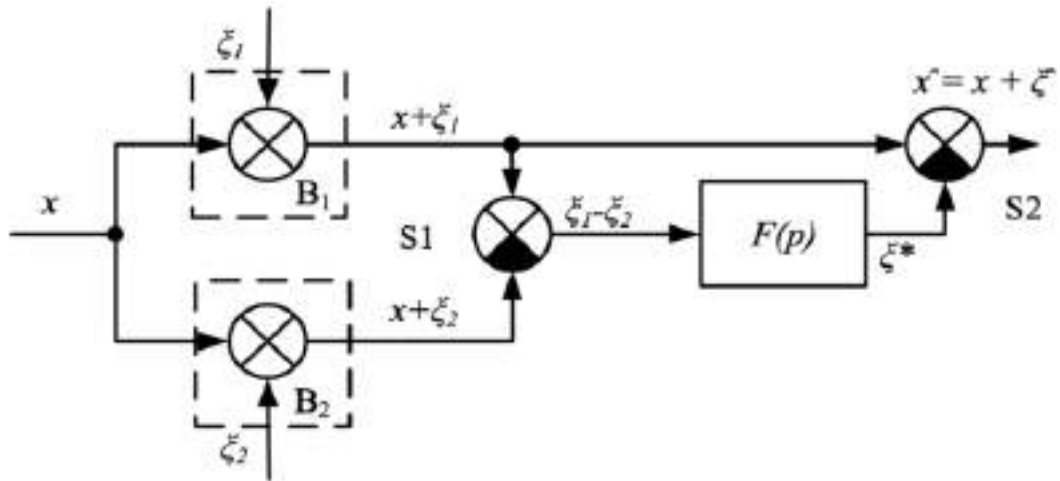


Рисунок 2.2. Схема компенсації із зовнішньою інформацією.

Сигнали суматорів B_1 і B_2 , отримують вимірювану величини $-x$, яка містить значення похибок ξ_1 і ξ_2 . Разом вони передаються на вхід суматора S_1 , що виконує віднімання B_1 , на виході якого формується інформація про оцінку похибки інформації:

$$e = (x + \xi_1) - (x + \xi_2) = \xi_1 - \xi_2. \quad (2.1)$$

Отримана інформація проходить через активний фільтр $F(p)$, яка подається на суматор S_2 де виконується віднімання значення оцінки похибки з сигналу першої (коректованої) системи.

Рівняння отриманої системи може бути записано у вигляді:

$$\hat{x} = x + \xi_1 - F(p)[\xi_1 - \xi_2], \quad (2.2)$$

або

$$\hat{x} = x + (1 - F(p))\xi_1 + F(p)\xi_2 = x + \hat{\xi}, \quad (2.3)$$

де $\hat{\xi} = (1 - F(p))\xi_1 + F(p)\xi_2$ – оцінка похибка системи позиціонування для моделі схеми з компенсацією.

В даній схемі фільтр $F(p)$ повинен пропускати з найменшим спотворенням похибку $\hat{\xi}_1$ і відрізати похибку $\hat{\xi}_2$, тоді загальна похибка системи буде

мінімальною. Тобто, отримана оцінка похибки $\hat{\xi}$ буде знижуватися залежно від різниць в характеристиках спектрів похибок $\hat{\xi}_1$ та $\hat{\xi}_2$.

При значній різниці в характеристиках похибок на виході блока $F(p)$ (див. рис.2.2) повністю буде відтворюватися завада $\hat{\xi}_1$, а на виході суматора 2 відбудеться віднімання $\hat{\xi}_2$ і відтвориться вимірне значення параметра x :

$$\hat{x} = x + \xi_1 - \xi_2 = x + \xi. \quad (2.4)$$

Фільтр пропускає частину завади $\hat{\xi}_1$ та не повністю знижує заваду $\hat{\xi}_2$, як наслідок цього вихідна інформація схеми крім потрібної інформації x буде містити похибку $\hat{\xi}$. Тут $F(p)$ має бути фільтром низьких частот, а фільтр $[1 - F(p)]$ буде фільтром високих частот.

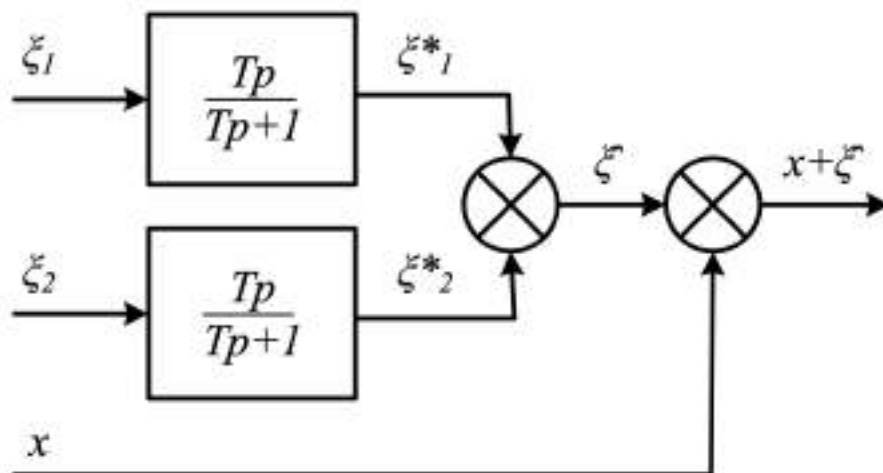


Рисунок 2.3. Схема компенсації з диференційованими ланками.

Якщо похибки ξ_1 і ξ_2 є стаціонарними і статично незалежними випадковими функціями часу зі спектральними густинами $S_{\xi_1}(\omega)$ і $S_{\xi_2}(\omega)$, то дисперсії похибок вихідних сигналів визначаються за формулою:

$$D(\xi_1) = \sigma_{\xi_1}^2 = \frac{1}{2\pi} \int_{-\infty}^{+\infty} S_{\xi_1}(\omega) d\omega, \quad (2.5)$$

$$D(\xi_2) = \sigma_{\xi_2}^2 = \frac{1}{2\pi} \int_{-\infty}^{+\infty} S_{\xi_2}(\omega) d\omega, \quad (2.6)$$

де S_{ξ_1} , S_{ξ_2} середньоквадратичні похибки вихідних даних.

Дисперсія системної похибки при реалізації методу компенсації наступна:

$$\sigma_{\xi_1}^2 = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \{ [1 - F(j\omega)]^2 S_{\xi_1}(\omega) + [F(j\omega)]^2 S_{\xi_2}(\omega) \} d\omega. \quad (2.7)$$

Виходячи із наведеного рівняння можна зробити висновок, що дисперсія похибок системи (рис.2.3), яка описується наведеною вище формулою, тим менша, чим більше відрізняються за частотою спектральні щільності похибок вхідних сигналів. В такому випадку, суміщення двох вимірів полягає у виборі такого значення зрізу фільтра $F(j\omega)$, щоб після злиття даних з диференційованими ланками відбувалася компенсація похибки [93–94], де оцінка отриманих даних буде рівна:

$$\hat{x} = x + \hat{\xi}. \quad (2.8)$$

2.3 Метод компенсації зі зворотнім зв'язком

Метод зі зворотнім зв'язком реалізується шляхом додавання ланцюга зворотного керуючого зв'язку, який створює двоконтурну систему із автоматичною корекцією інформації після обробки [110–111].

Схема такої компенсації може бути реалізована через використання зворотнього зв'язку зображеного на рис.2.4, а рівняння похибок у цьому випадку еквівалентні отриманим раніше для схеми компенсації без зворотного зв'язку (рис. 2.1). Рівняння похибок у цьому випадку еквівалентні отриманим раніше для схеми компенсації без зворотного зв'язку (рис. 2.3).

Запропоновану схему компенсації, можна записати у вигляді рівнянь:

$$\hat{x} = [(x + \xi_1) - \hat{x}]F(p) + x + \xi_2, \quad (2.9)$$

звідки маємо

$$\hat{x} = x + \frac{F(p)}{1+F(p)} \xi_1 + \frac{F(p)}{1+F(p)} \xi_2 = x + \hat{\xi}, \quad (2.10)$$

де,

$$\hat{\xi} = \frac{F(p)}{1+F(p)} \xi_1 + \frac{F(p)}{1+F(p)} \xi_2. \quad (2.11)$$

При $F(p) = Tp$ отримані рівняння похибок повністю співпадають.

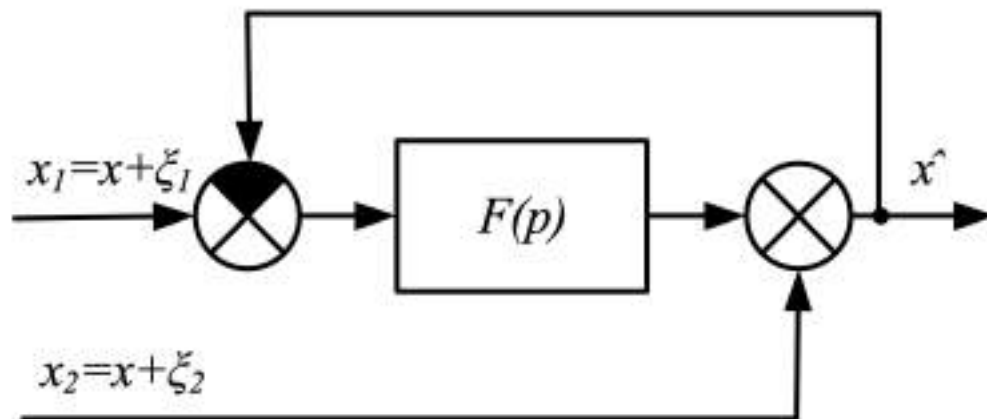


Рисунок 2.4. Схема компенсації зі зворотним зв'язком.

2.4 Метод оптимальної фільтрації

Метод оптимальної фільтрації – це метод статистичного машинного навчання, який використовується для видалення шуму з інформації [112–114]. Він працює, оцінюючи ймовірність того, що дані є корисними або шумом. Потім він використовує цю оцінку для видалення інформації, яка, ймовірно, є шумом.

Для досягнення максимальної ефективності в обробці інформації від джерел даних, ключовим є розробка оптимальної структури окремих приймачів даних та удосконалення алгоритмів спільної обробки інформації. Оптимальні методи лінійної фільтрації, які можуть бути застосовані в цьому контексті, включають:

- Фільтр Вінера-Колмогорова [115–116], призначений для стаціонарних стохастичних систем, з оцінкою ефективності за RMS;

– Лінійний фільтр Калмана-Бюсі [117], який є узагальненням для нестационарних стохастичних систем, з критерієм оцінки у вигляді мінімуму значення RMS;

– Дискретний фільтр Калмана [118], орієнтований на мінімізацію RMS;

– Метод найменших квадратів [119];

– Метод максимальної правдоподібності [120–121];

– Байєсівський підхід [122], заснований на апостеріорній ймовірності та ін.

Кожен з наведених методів оптимальної фільтрації та статистичного аналізу має свої особливості застосування, забезпечуючи ефективне рішення для різних задач обробки даних.

Для фільтрації шуму у стаціонарних стохастичних системах використовується фільтр Вінера-Колмогорова, де процеси та шуми мають постійні статистичні характеристики. Ефективність фільтрів, як правило, оцінюється за допомогою середньоквадратичної похибки (RMS), забезпечуючи оптимальне видалення шуму з інформації.

Для нестационарних стохастичних систем використовуються лінійний фільтр Калмана-Бюсі, який є адаптованим варіантом фільтра Вінера-Колмогорова, де характеристики процесу можуть змінюватися з часом. Фільтр Калмана-Бюсі мінімізує RMS похибку в динамічних системах, що дозволяє ефективно відстежувати стан системи зі зміною часу.

Для обробки даних, отриманих у дискретний момент часу, використовується дискретний фільтр Калмана спрямований на мінімізацію RMS похибки системи. Фільтр Калмана ефективний для систем, де наявного набору даних недостатньо або вони неповні.

Метод найменших квадратів є класичним методом що використовується для знаходження найкращого наближення отриманих значень інформації, мінімізуючи суму квадратів різниць між спостережуваними та оціненими значеннями позиціонування. Він широко застосовується у різних областях для регресійного аналізу інформації.

Метод максимальної правдоподібності застосовується для ідентифікації параметрів, які максимізують правдоподібність отриманих даних. Він часто використовується в статистичному аналізі інформації для оцінки параметрів моделі системи.

Байєсівський підхід використовує теорію апостеріорної ймовірності для оцінки невідомих параметрів системи, враховуючи попередні знання про позиціонування та поточні спостереження. Даний підхід включає оновлення моделі ймовірностей на основі нових даних системи, що надходять під час роботи системи, і є особливо ефективними в умовах коли отримана інформація має невизначеності.

Дані методи дозволяють оптимізувати обробку отриманої інформації про позиціонування БПЛА. Кожен з цих методів вносить свій внесок в обробку та аналіз інформації про позиціонування, забезпечуючи ефективні рішення для специфічних задач та умов.

Схема оптимальної фільтрації у випадку, коли один і той самий параметр вимірюється двома вимірювачами, має вигляд, показаний на рис.2.5.

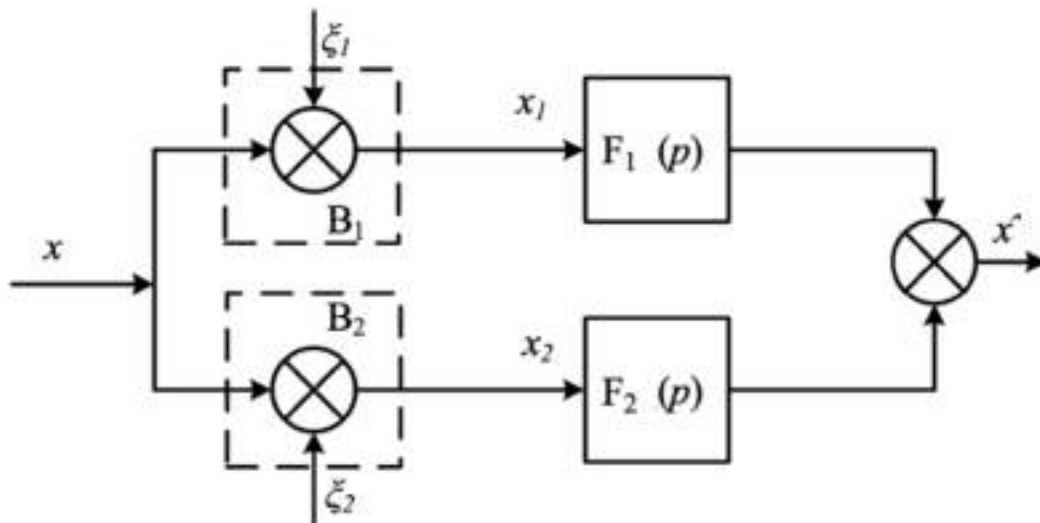


Рисунок 2.5. Схема оптимальної фільтрації.

Інформація на виході системи з оптимальною фільтрацією описується наступним рівнянням:

$$x = \hat{F}_1(p)(x + \xi_1) + \hat{F}_2(p)(x + \xi_2), \quad (2.12)$$

або

$$x = [\hat{F}_1(p) + \hat{F}_2(p)](x + \xi_2)x + \hat{F}_1(p)\xi_1 + \hat{F}_2(p)\xi_2. \quad (2.13)$$

Для того, щоб сама схема не додавала власні динамічні похибки [123–124], необхідно передбачити виконання наступної умови:

$$F_1(p) + F_2(p) = 1. \quad (2.14)$$

В описаному випадку вихідні дані будуть мати вигляд:

$$\hat{x} = x + [1 + \hat{F}_2(p)]\xi_1 + \hat{F}_2(p)\xi_2 = x + \hat{\xi}, \quad (2.15)$$

де

$$\hat{\xi} = [1 + F(p)]\xi_1 + F(p)\xi_2. \quad (2.16)$$

Таким чином, в умовах коли $F_2(p) = F(p)$ вираз (2.15) трансформується до виразу (2.3), що свідчить про еквівалентність запропонованих схем фільтрації та компенсації. Тоді, отримаємо схему що поєднує одночасно схеми компенсації і фільтрації [125–126], яка була зображена на рис.2.2.

2.5 Алгоритм для зменшення шуму та відхилень магнітометра

Магнітометри [127–129], які використовуються для визначення магнітного курсу, схильні до шумів та перешкод, які можуть виникати через зовнішні магнітні поля або внутрішні вібрації. Одним із дієвих підходів до зниження їх впливів є застосування експоненційного рухомого середнього (ЕМА) фільтрування [130–132].

ЕМА є типом згладжувального фільтра, який використовується для зменшення шуму в часових рядах даних. Його основна мета – вирівняти короткострокові коливання, зберігаючи при цьому довгострокові тенденції [133–135]. Це дозволяє генерувати більш точні та надійні вимірювання, які критично важливі для пілотування, особливо в динамічних умовах або вимогливих навколишніх середовищах.

Завдяки тому, що ЕМА приділяє більше уваги останнім спостереженням, система швидко адаптується до нових умов без втрати загальної інформативності трендів. Це забезпечує плавне відстеження змін курсу, що є необхідним для точного керування та планування траєкторії БПЛА або інших завдань, що використовують інформацію про курс. Логіку роботи цифрової ЕМА фільтрації, можна записати у вигляді рівняння:

$$C_t = \alpha Y C_t + (1 - \alpha) C_{t-1}, \quad (2.17)$$

де, C_t – значення курсу у момент часу (t), α – коефіцієнт згладжування, який знаходиться в діапазоні від 0 до 1, Y_t – спостережуване значення у момент часу (t), C_{t-1} – значення курсу в попередній часовий момент ($t-1$).

Алгоритм роботи ЕМА фільтру полягає в тому, що він надає більшу вагу останнім спостереженням у наборі даних. Його відмінність від простого ковзного середнього полягає у ваговому коефіцієнті α , який застосовується до кожного нового спостереження. Коефіцієнт α , який може варіюватися від 0 до 1, визначає ступінь впливу останніх спостережень системи на загальний результат. Високе значення α забезпечує вищу чутливість фільтра до останніх змін у отриманій інформації, тоді як низьке значення коефіцієнта α робить його менш чутливим до останніх спостережень, але більш згладженим і стабільним у тривалій перспективі [136–137].

В системах відстеження змін курсу та інших параметрів даний керування дозволяє забезпечуючи плавність та стабільність управління БПЛА. Це особливо важливо в динамічних умовах роботи, де раптові зміни напрямку руху можуть

викликати розбалансування системи керування. Застосування ЕМА в таких ситуаціях допомагає оптимізувати команди керування та знизити ризик відхилень, викликаних недосконалістю датчиків або зовнішніми перешкодами.

Використання ЕМА фільтрів допомагає підвищити робастність системи позиціонування, мінімізуючи вплив тимчасових аномалій або помилкових сигналів, що можуть виникнути від інших електронних пристроїв або металевих конструкцій.

2.6 Алгоритми комплексування та корекції в INS/GNSS

Інтеграція INS/GNSS є однією із найбільш поширених, ефективних та бюджетних рішень, та підходить для багатьох сценаріїв застосування БПЛА [138–140].

Використовуючи алгоритми комплексування та корекції в сучасних INS/GNSS, забезпечують точне визначення місцезнаходження БПЛА з точністю до кількох метрів. В основі рішення лежить спеціалізований метод обробки інформації, що дозволяє створювати інтегровані системи, які поєднують точність і функціональність з доступною вартістю, компактними розмірами та високою надійністю. Ці системи дозволяють автоматизувати управління БПЛА в основних режимах польоту, включаючи політ за заданим маршрутом, точне входження на задані координати, посадку, формування груп літаків, дотримання заданих порядків та ешелонів польоту, а також підвищення бойових можливостей при групових діях.

В інтегрованих INS/GNSS можливе використання бюджетних датчиків INS на основі MEMS чутливих елементів [141–143]. Такі інерціальні системи вирізняються малою вагою та компактними розмірами, але їх автономне використання ускладнюється через нестабільність робочих характеристик мікроелектромеханічних гіроскопів та акселерометрів, що призводить до швидкого накопичення похибок у отриманих даних позиціонування. Даний підхід дозволяє використовувати методи корекції зворотного зв'язку, щоб

протидіяти потенційному зниженню точності позиціонування протягом тривалого часу, що надає обчислювальні переваги та потенціал для забезпечення підвищеної точності позиціонування.

Використання інтегрованих INS/GNSS дозволяє в результаті мінімізувати або взагалі усунути недоліки кожної окремо взятої системи [144–145], а саме:

недоліки GNSS:

- сигнал GPS не може проходити через товсті перешкоди, тому він не може працювати в приміщеннях, під землею, під водою або під щільним дерев'яним навісом;
- має низьку точність в районах щільної забудови;
- точність GPS залежить від якості прийому сигналу, абсолютна мініатюризація неможлива, якщо необхідно зберігати хорошу точність позиціонування.

недоліки INS:

- необхідність введення початкових даних (початкового положення об'єкта), початкових значень швидкості, початкових координат), що необхідно для використання методу обчислення шляху;
- потрібна безперервність роботи (або після перерви в роботі необхідно знову вводити початкові дані).
- накопичення похибок із часом.

Використання GNSS-технологій для корекції INS підходить для багатьох рішень з низьким бюджетом та дозволяє використовувати відносно недорогі датчики, побудовані на мікроелектромеханічних та наномеханічних чутливих елементах [146–149].

Система інтеграції INS та GNSS є складним технологічним поєднанням двох технологій позиціонування, яка спрямоване на забезпечення отримання надійної та високоточної інформації про позиціонування. Архітектуру типової системи інтеграції INS та GNSS показано на рис.2.6.

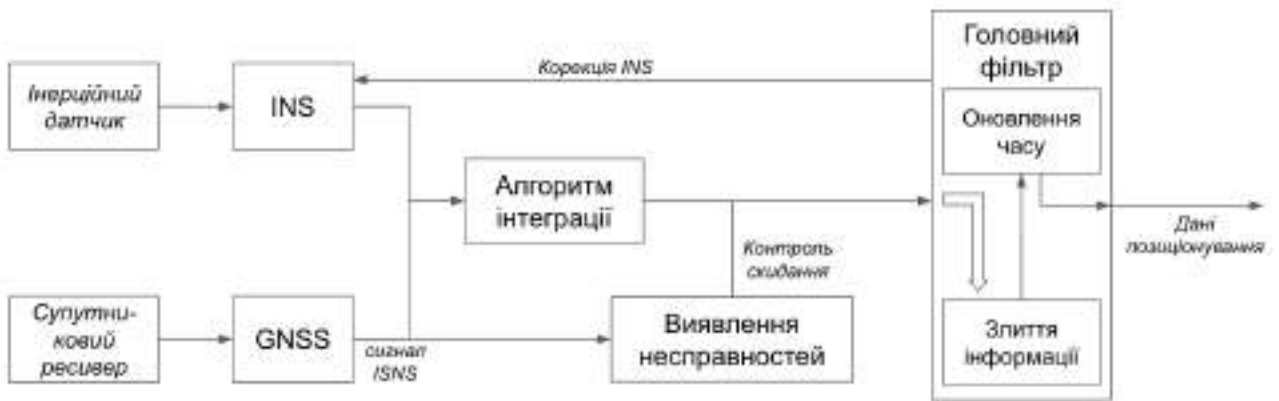


Рисунок 2.6. Архітектура універсальної системи інтеграції INS/GNSS.

В наведеній системі інерційні датчики починають процес відліку, фіксуючи фізичні рухи об'єкта. Надходження первинних даних від датчиків в INS ініціює їх початкову обробку, що полягає у перетворенні вимірювань у корисну інформацію про зміни станів об'єкта. В той же час, GNSS-приймачі здійснюють прийом сигналів від супутників, що знаходяться на орбіті, для визначення нового положення БПЛА в просторі.

Отримана інформація від обох систем подається на блок обчислення, що за допомогою алгоритму інтегрування виконує початкове злиття інформації, коригуючи та удосконалюючи вихідні дані INS за допомогою GNSS. Цей крок також включає в себе виявлення та усунення можливих несправностей та похибок в інформації.

Далі інформація подається на головний фільтр в якому відбувається оптимальна обробка та оптимізація інформації, часові відмітки оновлюються, шум і помилки фільтруються, інформація зливається для формування оцінки точної позиції об'єкта в просторі.

Після завершення процесу обробки інформації, на вихід системи подається остаточно оброблена інформація готова до використання в широкому спектрі завдань керування та позиціонування. Унікальність даної системи полягає у її здібності до адаптації в змінних умовах, що є незамінною для автономного керування БПЛА.

2.7 Особливості обробки інформації в багатосупутникових системах

Одним із найбільш перспективних сьогодні виглядає підхід використання архітектури багато-супутникових систем позиціонування [150–152].

Мета даного підходу полягає в тому, щоб використовувати сильні сторони окремих супутникових сузір'їв, одночасно компенсуючи їхні слабкі сторони. Супутникові служби позиціонування, що покладаються виключно на супутники високої та середньої навколоземної орбіти, є недостатніми для умов використання, що розглядаються в даній роботі. Тому важливим завданням є перехід до комплексного підходу побудови "багатошарової системи систем" [153]. Такий підхід дозволяє підвищити якість інформації з GNSS даними з LEO супутників (рис.2.7).

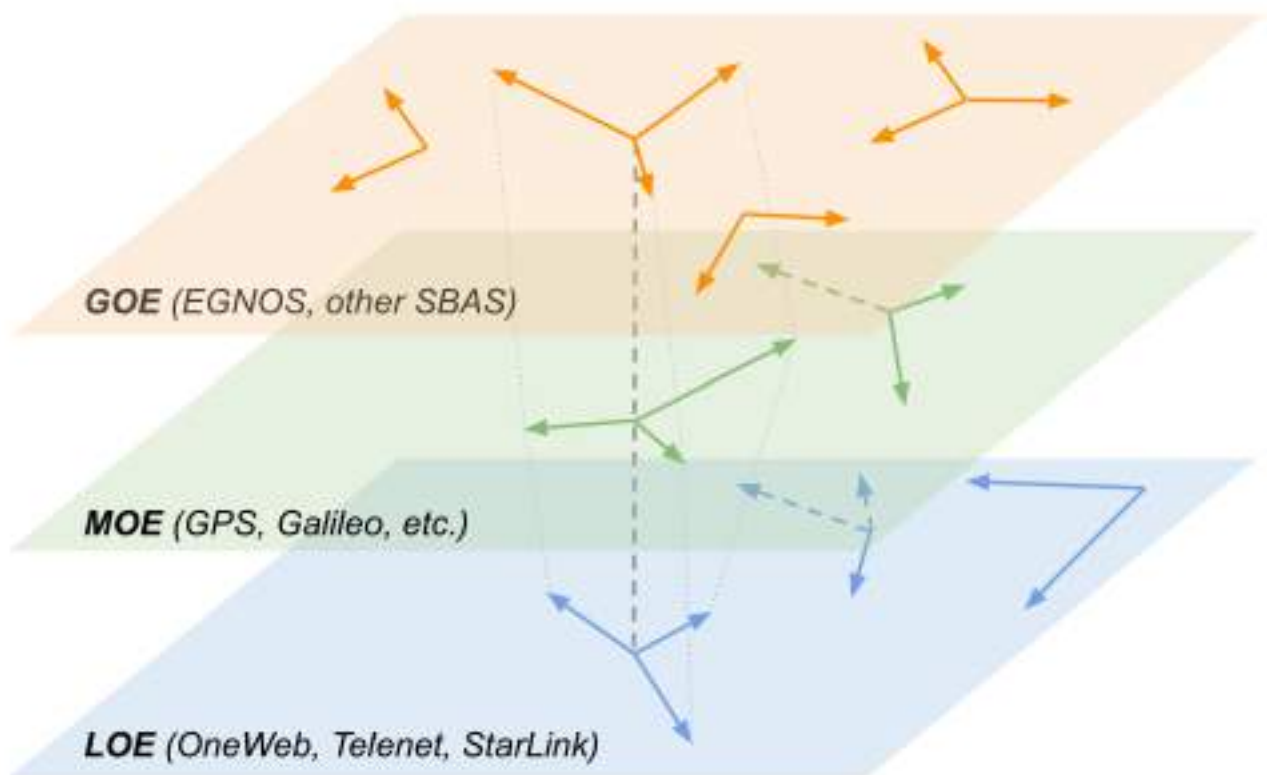


Рисунок 2.7. Ілюстрація методу "багатошарової системи систем" позиціонування з використанням супутників різного типу

На представленому рисунку представлена архітектуру багатошарової супутникової системи, що виконує інтегрування інформації з різних орбітальних

груп. Вона поєднує сузір'я супутників з різних орбіт – низької (LEO), середньої (MEO) та геостаціонарної (GEO).

На нижньому шарі (рівні) схеми розташовано LEO супутники, такі як OneWeb, Telenet, StarLink, які знаходяться ближче до Землі та забезпечують швидку та точну передачу даних завдяки коротшому затриманню сигналу. Вони становлять основу для швидкісного зв'язку БпЛА, особливо важливого в реальному часі.

Середній шар (рівень) представлений MEO супутниками, включаючи системи GPS, Galileo та інші. Ці супутники формують основу глобальних систем позиціонування. Вони забезпечують широке охоплення та довготривалу стабільність системи позиціонування.

На верхньому шарі (рівні) знаходяться GEO супутники, такі як EGNOS, SBAS та інші, які знаходяться на великій висоті, що дозволяє їм покривати великі території. Дані супутникові угруповання використовуються для підвищення точності, коригування похибок та надійності роботи системи на нижчих шарах.

Інтеграція даних між цими трьома шарами інформації дозволяє створити "багатошарову систему систем", яка максимізує потенціал кожного сузір'я, забезпечуючи високу точність позиціонування та надійність отриманої інформації. Це суттєво покращує якість даних для критично важливих застосувань, де вимагається найвища точність та низька затримка сигналів, наприклад, у керуванні БпЛА або під час виконання автономних місій в динамічних умовах зовнішнього середовища.

Оскільки сигнали GPS слабкі та сприйнятливі до перешкод, вони часто стають ненадійними джерелами інформації в певних місцях, наприклад у приміщеннях або в глибоких "міських каньйонах" [154]. Крім того, сигнали GNSS можна підробити, що створює серйозні ризики для безпеки в критично важливих для безпеки сферах застосування, таких як авіація та безпілотні системи [155–156].

Система з декількома сузір'ями працює краще, ніж система лише з GPS сузір'я, і перешкоди на діапазонах GNSS мають більш критичні наслідки для

точності позиціонування, ніж перешкоди діапазону низькоорбітальної системи (LEO). Крім того, існуючі дослідження впливу сигналів перешкод на приймачі демонструють переваги використання методів злиття кількох сузір'їв і врахування різних типів сигналів перешкод при розробці приймачів GNSS для критично важливих застосувань [157].

Угруповання супутників досягають майже постійного глобального покриття завдяки використанню унікальних характеристик динаміки переміщення LEO. Використовувати сигнали від супутників LEO, як доповнення або альтернативу для позиціонування, позиціонування та визначення часу, є доцільним оскільки вони знаходяться приблизно в 20 разів ближче до Землі порівняно з супутниками GNSS, які знаходяться на середній орбіті більше ніж на 20 000 кілометрів над Землею [158], що забезпечує швидшу передачу даних, менший вплив іоносферного викривлення даних і менший час затримки при їх передачі [159–161].

Водночас, під час аналізу та обробки отриманої інформації супутників необхідно враховувати, що супутникові платформи глобального позиціонування містять свої орбітальні похибки позиціонування. Так, орбітальні похибки трансляції ефемерид GNSS зазвичай не перевищують 10 м [162–163]. У разі використання недорогих приймачів систем позиціонування (наприклад, як у смартфонах), вони краще підходять для динамічних застосувань, та демонструють середній зсув між еталонною та вимірною траєкторіями від 1,23 до 5,98 м [164].

Використання інерціальної багато-супутникової підсистема позиціонування (LeGNSS) в бортових підсистемах БпЛА представляє собою значний крок уперед у точності та надійності позиціонування. Ця система, інтегруючи класичні GNSS сигнали з новітніми телекомунікаційними супутниковими угрупованнями на низьких навколосемних орбітах, забезпечує значно ширше покриття сигналами та вищу точність отриманої інформації про позиціонування. Завдяки більшій кількості супутників LeGNSS дозволяє інтегрувати із класичними GNSS (до 30

супутників) нові супутникові угруповання, такі як SpaceX (4487 супутників), OneWeb (720 супутників) і Telesat (117 супутників) [165–167].

Можливість використання телекомунікаційних LEO супутників для вирішення, у тому числі, задач позиціонування вперше було запропоновано в 2022 році, а вже в 2023 році було підтверджено практичну можливість декомпозиції сигналів даних супутників та вперше експериментально отримано інформацію про позиціонування приймача [168–170]. Дослідники виявили, що, проводячи спостереження за сигналами восьми LEO супутників протягом 10 хвилин, їхні алгоритми обробки інформації змогли досягти безпрецедентної точності визначення місцезнаходження нерухомого приймача на землі з просторовою похибкою приблизно 5,8 – 6,5 метра [171–172]. При цьому, для отримання даних про позиціонування не потрібна була яка небудь допомога операторів телекомунікаційних мереж для використання їх сигналів, достатньо зчитувати лише загальнодоступну інформація з ефіру, що транслюється супутниками до кінцевих користувачів послуг супутникового зв'язку, пов'язаної з частотою передачі супутників у низхідній лініях зв'язку та приблизною оцінкою розташування супутників [173–175].

Для розробки та аналізу характеристик запропонованої моделі LeGNSS, необхідно вдосконалити метод оцінки моделі похибок у компонентах підсистеми прийому та обробки даних, що враховує інтеграцію LEO супутників.

Похибка оцінки позиціонування від обраних систем може бути представлена як e_{GPS} , e_{LEO} та e_{INS} . Сукупна похибка e_{LeGNSS} є функцією окремих похибок кожного сузір'я:

$$e_{LeGNSS} = f(e_{GPS}, e_{LEO}, e_{INS}). \quad (2.18)$$

де f – це функція, яка залежить від способу злиття даних.

Якщо w_{GPS} , w_{LEO} та w_{INS} є ваговими коефіцієнтами для GPS, LEO та INS відповідно, то оптимізована сукупна похибка може бути представлена як:

$$e_{LeGNSS} = w_{GPS} e_{GPS} + w_{LEO} e_{LEO} + w_{INS} e_{INS} \quad (2.19)$$

де вагові коефіцієнти w визначаються через процес оптимізації моделі.

Вдосконалений методу оцінки похибок у компонентах підсистеми прийому та обробки даних передбачає кілька ключових етапів:

- 1) Спочатку необхідно ідентифікувати та виміряти похибки кожної із супутникових систем окремо, тобто e_{GPS} , e_{LEO} та e_{INS} . Це включає в себе збір даних з кожного джерела та оцінку їхньої точності і надійності.
- 2) Отримані похибки інтегруються за допомогою функції f , яка враховує не тільки величину окремих похибок, але й їх взаємозв'язок, кореляцію та потенційну компенсацію одна одної.
- 3) Через процес оптимізації визначаються вагові коефіцієнти w_{GPS} , w_{LEO} та w_{INS} . Дані коефіцієнти відображають внесок кожної системи в загальну точність моделі та дозволяють балансувати вплив кожної системи на кінцеву похибку e_{LeGNSS} .
- 4) Використовуючи отримані вагові коефіцієнти, формується оптимізована сукупна похибка e_{LeGNSS} , яка підсистема LeGNSS має мінімізувати для досягнення найкращої можливої моделі точності позиціонування.

Удосконалений метод оцінки похибок дозволяє не тільки з високою точністю визначити оцінки похибок моделі позиціонування, але й забезпечити адаптивність системи до змінних умов, підвищуючи таким чином загальну надійність та ефективність роботи підсистеми LeGNSS.

Використання статистичних методів, таких як фільтр Калмана, дозволяє оцінити стан системи в реальному часі та коригувати похибки [176–177]. Для цього фільтр Калмана здійснює серію вимірювань, спостережуваних протягом часу, що містять шум (наприклад, похибки вимірювань) та інші похибки, і надалі визначає оцінки для невідомих змінних, що дозволяють мінімізувати RMS похибку [178–181]. Важливо також аналізувати, як система реагує на різні зовнішні умови та помилки. Це включає в себе моделювання різних сценаріїв,

таких як втрата сигналу від одного з джерел, та оцінку впливу цих сценаріїв на загальну точність та надійність системи.

Формула 2.18 представляє залежності та взаємодії під час поєднання даних GPS, LEO та INS. Фактичні значення ваг, коефіцієнтів злиття та інших параметрів залежатимуть від конкретного застосування, якості вимірювань кожної системи, заданої точності та надійності комбінованих даних підсистеми.

Використання передових методів статистичного аналізу та алгоритмів фільтрації дозволить підвищити точність моделювання та, відповідно, ефективність самої системи LeGNSS. Цей підхід забезпечить глибше розуміння динаміки та поведінки системи в реальних умовах експлуатації, що є ключовим для подальшого вдосконалення та оптимізації системи позиціонування БпЛА.

Даний підхід побудови моделі робастної LeGNSS підсистеми позиціонування є оптимальним та дозволяє побудувати систему із надзвичайною стійкістю та точністю отримуваної інформації про позиціонування. Велика кількість супутників забезпечує кращу геометрію визначення координат, ніж GPS, і дозволяє використовувати дешевші приймачі отримуючи вищу точність інформації позиціонування.

2.8 Розробка методів адаптивного позиціонування для різних умов експлуатації

Зі зростанням складності завдань і розширенням географії використання дронів, зокрема безпілотних літальних систем, необхідність у точному та гнучкому позиціонуванні стає важливою. Сучасні системи позиціонування мають відповідати ряду вимог, включаючи високу точність, надійність та здатність до адаптації в різноманітних умовах експлуатації.

У контексті швидкого розвитку технологій і розширення діапазону завдань, що вирішуються за допомогою БпЛА, адаптивне позиціонування виходить на головний план, оскільки забезпечує необхідну гнучкість та ефективність роботи в складних умовах експлуатації та з зоні дії систем РЕБ [182–183].

Умови експлуатації можна класифікувати за кількома критеріями: від географічних особливостей місцевості до специфіки завдань, які виконуються. Кожен тип умови ставить перед системою позиціонування унікальні вимоги, чи то стосується стійкості до атмосферних перешкод, точності визначення координат в гірській місцевості, або надійності у міському середовищі [184–186]. Перешкоди, які виникають у певних середовищах, можуть істотно впливати на працездатність системи. Наприклад, у міських районах велика кількість перешкод може викликати проблеми з GNSS сигналами, тоді як у віддалених районах можуть виникати труднощі з підтримкою стабільного зв'язку.

Розробка алгоритмів адаптивного позиціонування передбачає створення таких рішень, які здатні ефективно адаптуватися до змінних умов експлуатації. Це вимагає від алгоритмів гнучкості та здатності до швидкого переналаштування в залежності від зміни зовнішніх чинників. Принципи розробки адаптивних алгоритмів залежать від необхідності забезпечення гнучкості, точності та швидкості відгуку. Важливими є такі аспекти, як модульність, що дозволяє легко інтегрувати нові компоненти або оновлення, та масштабованість, яка гарантує ефективну роботу системи при різних обсягах завдань та різній кількості цілей.

На практиці адаптивні алгоритми можуть включати системи, засновані на використанні штучного інтелекту для аналізу даних з декількох джерел, алгоритми фільтрації для мінімізації впливу помилок та шуму, та алгоритми, які оптимізують точність визначення місцеположення в залежності від специфіки середовища, та алгоритми що дають можливість системі приймати автономно рішення з допомогою раніше закладеного алгоритму.

Впровадження адаптивних методів у реальних умовах вимагає врахування багатьох факторів, включаючи енергоефективність, економічну доцільність, зручність використання та сумісність із існуючими системами [187–189]. У цивільних застосуваннях, таких як моніторинг навколишнього середовища чи сільськогосподарські дослідження, адаптивні методи дозволяють оптимізувати маршрути та забезпечити надійне збирання даних навіть у складних умовах. У військовому контексті, адаптивність системи позиціонування є ключовим

фактором для виконання складних операцій із мінімальним ризиком для обладнання та персоналу. Важливим аспектом є також забезпечення безпеки, особливо в умовах, де помилки у позиціонуванні можуть призвести до серйозних наслідків.

Адаптивні системи позиціонування стають все більш важливими у різних областях діяльності, відкриваючи нові горизонти для вдосконалення технологій застосування БпЛА. Успішне впровадження даних рішень вимагає не лише технічної майстерності, але й глибокого розуміння специфіки кожного сценарію використання, а також готовності до створення сучасних алгоритмів створених за допомогою машинного навчання.

2.9 Висновки до другого розділу

У другому розділі проведено аналіз ключових методів та засобів, що дозволяють підвищити ефективність та точність роботи підсистем позиціонування в контексті БпЛА.

За результатами проведеного аналізу визначено найбільш ефективним підходом є інтеграції надлишкових джерел даних про позиціонування, таких як інерціальні системи, глобальні супутникові системи позиціонування (GNSS) та інші технічні засоби та використання сучасних методів обробки інформації, зокрема, використання компенсації та оптимальної фільтрації для зменшення похибок і шуму в даних. Особливо ефективним для вирішення завдань підвищення точності позиціонування є використання фільтрів Калмана, які застосовуються для оптимальної інтеграції інформації з різних джерел.

Також було розглянуто застосування засобів для контролю цілісності роботи системи та виявлення втручання в отримувану інформацію. Такі засоби дозволяють ефективно коригувати відхилення та знижувати вплив зовнішніх чинників на точність визначення позиціонування БпЛА.

Важливим компонентом для побудови сучасних високоточних систем позиціонування є інтеграція багатосупутникових систем, які використовують

сигнали з різних сузір'їв, в тому числі дані від низькоорбітальних телекомунікаційних супутників. Це демонструє важливість розширення можливостей сучасних систем позиціонування шляхом включення додаткових джерел інформації.

Подальше вдосконалення та розвиток систем позиціонування відкривають нові перспективи для розробки більш ефективних та надійних систем, забезпечуючи високу якість та робастність роботи підсистем позиціонування для різноманітних застосувань в БПЛА.

РОЗДІЛ 3

СИНТЕЗ ПІДСИСТЕМИ ПОЗИЦІОНУВАННЯ LeGNSS

3.1 Розробка інтегрованої підсистеми LeGNSS

Підсистема LeGNSS є інноваційним рішенням поєднання INS та GNSS, що інтегрує інформацію від LEO систем. Мета цієї інтеграції полягає у створенні більш точної, надійної та ефективної системи позиціонування для БПЛА.

Інтеграція даних інерціальних та супутникових систем є ключовим аспектом роботи підсистеми LeGNSS. INS надає неперервні дані про положення та орієнтацію БПЛА, в той час як GNSS та LEO забезпечує дані про позиціонування. Інтеграція цих трьох систем дозволяє використати їх переваги для компенсації недоліків кожної, наприклад, висока частота даних INS та стабільна точність GNSS і LEO, забезпечуючи надійність та стабільність визначення позиціонування БПЛА.

Інтеграція з даними від LEO забезпечує додаткові переваги. Дані від телекомунікаційних супутників мають властивість забезпечувати більш часті оновлення даних та краще покриття у віддалених або складних для доступу регіонах, де робота GNSS може бути обмежена або подавлена. Це дозволяє підсистемі LeGNSS забезпечити високу точність позиціонування в різних умовах, в тому числі коли якась із систем може піддаватися впливу зовнішніх чинників.

Синтез інформації з різних джерел є важливою частиною розробки підсистеми LeGNSS. Це включає в себе алгоритми фільтрації та обробки інформації із використанням дискретного фільтра Калмана, для оптимізації точності та надійності отримуваної інформації. Ці алгоритми дозволяють об'єднати дані з INS, GNSS та LEO супутників, забезпечуючи точне та надійне позиціонування навіть у складних умовах експлуатації.

Основна ідея пропонованого рішення – подолати відому проблему дрейфу інерціальної системи за допомогою інтелектуального алгоритму виявлення збоїв

та набору надлишкових джерел інформації. Вирішенням проблеми є періодична корекція інерціальної системи позиціонування від інших джерел інформації. Під час корекції бортової інерціальної системи її вимірювання порівнюються з відповідними вимірюваннями систем корекції. Зазвичай порівнюють швидкість і координати. Використання методу оптимальної обробки інформації як додаткового інструменту фільтрації інформації сприяє збільшенню точності шляхом ефективного видалення високочастотних похибок в інформації, отриманій від супутникових джерел.

Враховуючи похибки визначення координат ΔS , швидкості ΔV і вертикалі β , можна скласти рівняння системи руху БПЛА та проаналізувати її. Характеристичне рівняння цієї системи має вигляд:

$$F(p) = p^3 + K_2 p^2 + \left(\frac{g}{R} + K_1\right) p + \left(\frac{g}{R} + K_2\right) = 0. \quad (3.1)$$

Аналіз характеристичного рівняння показує, що підбором коефіцієнтів K_1 і K_2 можна досягти стійкості перехідних процесів в схемі, а аналізуючи постійні значення похибок INS, можна зробити висновок, що в цьому випадку кут швидкості дрейфу не впливає на постійні значення похибок визначення швидкості польоту та координат БПЛА.

При об'єднанні даних з декількох систем (GPS, LEO, INS), залежності та взаємодії між цими системами можуть бути представлені за допомогою математичних формул. Інформація щодо позиціонування, отримане від кожної із систем, можна визначити як P_{GPS} , P_{LEO} та P_{INS} .

Об'єднана позиція P_{LeGNSS} (3.2) може бути середньозваженим значенням позиціонування від кожної системи:

$$P_{LeGNSS} = w_{GPS} \cdot P_{GPS} + w_{LEO} \cdot P_{LEO} + w_{INS} \cdot P_{INS}, \quad (3.2)$$

де w – ваговий коефіцієнт або рівень достовірності оцінки позиції для кожної із систем.

LEO супутникові системи покладається на технологію ортогонального частотного мультиплексування (OFDM) [190]. Дана технологія є ефективним методом кодування цифрових передач інформації, яка спочатку була розроблена Bell Labs у 1960-х роках, а зараз використовується в WiFi та 5G. Вона дозволяє упакувати найбільшу кількість бітів на секунду в задану пропускну здатність.

Системи що використовують технологію OFDM забезпечують високу щільність та надійність передачі даних, що є ключовим аспектом для БПЛА. Завдяки їх здатності до ускладнити втручання в інформацію, ці системи ідеально підходять для динамічних та ворожих умов експлуатації, в яких швидкість і точність передачі інформації мають вирішальне значення. Адже в таких умовах кожен біт інформації може мати вирішальне значення на кінцевий результат місії.

Крім того, використання OFDM у LEO супутникових системах дозволяє оптимізувати використання частотного спектру, що є критичним у контексті постійно зростаючого попиту на бездротовий зв'язок. Технологія дозволяє розділити високу частоту на кілька низьких частот, які передаються паралельно, зменшуючи інтерференцію та збільшуючи ефективність передачі. Це робить OFDM ідеальним рішенням для сучасних систем передачі інформації, які вимагають швидкої та надійної передачі великого обсягу інформації, особливо в умовах де звичайні методи можуть виявитися неефективними.

3.2 Вимоги до робастності та надійності системи

Робастність та надійність є критичними параметрами, що впливають на ефективність та безпеку використання системи. Важливо формувати чіткі вимоги до точності, стійкості системи до похибок, а також встановлювати граничні значення допустимих похибок.

Вимоги до точності мають враховувати різні сценарії використання та умови, в яких буде оперувати БпЛА. Це включає забезпечення високої точності позиціонування в різних географічних локаціях, при різних погодних умовах та в умовах змінної якості вхідної інформації GNSS.

Вимоги до робастності мають включати спроможність системи підтримувати точність позиціонування навіть при тимчасовій втраті сигналу GNSS, впливі електромагнітних завад або внутрішніх похибок системи.

Середньоквадратичне відхилення (RMS) є базовим аналітичним показником, що використовується для оцінки відхилення виміряного значення позиціонування від істинного. У контексті позиціонування БпЛА, оцінка RMS відіграє вирішальну роль, дозволяючи оцінити та проаналізувати просторові похибки, що мають системи позиціонування в тривимірному середовищі.

Для аналізу запропонованої моделі підсистеми позиціонування LeGNSS необхідно врахувати порівняння значень RMS похибок позиціонування ефемерид для обраних типів орбітальних супутників представлено в таблиці 3.1 [191].

Для аналізу точності сигналів бюджетних наземних приймачів, що відповідають рівню точності датчиків в смартфонах, було зібрано масив даних позиціонування та використано RMS похибки псевдодальностей для систем GPS, LEO, тощо [192–197], порівняння їх значення для різних діапазонів прийому сигналів представлено в таблиці 3.2.

Таблиця 3.1

Усереднені значення орбітальних RMS похибок ефемерид для супутників

Тип супутника	Поздовжня похибка (см)	Поперечна похибка(см)	Радіальна похибка (см)	3D похибка (см)	Похибка часу (нс)
BDS GEO	110.5	0.3	0.3	110.5	0.11
BDS IGSO	0.9	0.9	0.3	1.3	0.10
BDS MEO	0.7	0.6	0.3	1.0	0.09

GPS	0.6	0.2	0.1	0.7	0.07
LEO	1.3	0.6	1.2	1.9	0.20

Таблиця 3.2

Значення псевдодіапазону RMS похибки приймачів

Тип супутника	Min (см)	Макс (см)
BDS GEO	260	360
BDS IGSO	255	465
BDS MEO	276	696
GPS	277	573
LEO	580	2600

Для оцінки сумарної похибки системи позиціонування, яка включає в себе використання даних ефемерид орбітальних супутників та вимірювань псевдодіапазону з недорогих приймачів, необхідно використовувати квадратичне сумування індивідуальних похибок. Це виходить з припущення, що різні джерела похибок є некорельованими та мають нормальний розподіл:

$$RMS_{source} = \sqrt{(RMS^2_{\text{ефемериди}} + RMS^2_{\text{приймач}})} \quad (3.3)$$

де, RMS_{source} – загальна середньоквадратична похибка системи, отримана на основі розрахунку суми квадратів середньоквадратичних похибок ефемерид супутників та похибок приймачів.

Ця формула враховує, як похибки ефемерид супутників, так і похибки псевдодіапазонів приймачів, дозволяючи оцінити загальну похибку системи. Дана модель оцінки сумарної похибки дозволяє розробити ефективну та надійну систему позиціонування з урахуванням ключових факторів які впливають на похибку позиціонування.

Для інших типів супутників формула розрахунку RMS буде аналогічна до наведеної в формулі 3.3.

Встановлення граничних значень похибок є важливим для забезпечення безпеки та ефективності роботи проектованої схеми. Це включає в себе визначення максимально допустимих похибок в позиціонуванні та орієнтації БПЛА, що дозволяє гарантувати ефективність виконання завдань.

Для розробки ефективної та надійної підсистеми LeGNSS, важливо встановити чіткі вимоги до точності, робастності та граничних значень похибок. Для цілей розробки визначимо наступні очікувані вимоги:

вимоги до точності:

- **позиціонування:** RMS похибка тривимірного позиціонування (X, Y, Z координати) не повинна перевищувати 3,2 метра в 95% випадків;
- **швидкість:** RMS похибка оцінки абсолютної швидкості польоту не повинна перевищувати 1 м/с в 95% випадків;
- **курсорова орієнтація:** RMS похибка в орієнтації БПЛА не повинна перевищувати 1,5 градуса;
- **час реакції:** період реакції на зміну позиції повинен бути не більше 1 секунди.

вимоги до робастності:

- **стійкість:** система повинна підтримувати точність позиціонування при впливі зовнішніх завад, забезпечуючи роботу в різних умовах експлуатації;
- **адаптивність:** система має бути адаптивною до змін умов, таких як втрата сигналу, забезпечуючи надійність в різних географічних регіонах.
- **відновлення після збоїв:** система повинна мати механізм відновлення після тимчасових збоїв або перерв у роботі.

граничні значення похибок:

- **максимальна похибка позиціонування:** максимально допустима похибка в позиціонуванні не повинна перевищувати 4 метра;
- **максимальна похибка орієнтації:** максимальна похибка в орієнтації не повинна перевищувати 2 градуси.
- **максимальний час відновлення:** час відновлення системи після збою не повинен перевищувати 30 секунд.

Визначені вимоги формують основу для розробки надійної та точної підсистеми LeGNSS, яка може бути ефективно використана у широкому діапазоні умов та застосувань в сучасних БПЛА.

3.3 Діагностика та компенсація збоїв у підсистемі LeGNSS

Аналіз та оптимізація обробки інформації в підсистемі позиціонування є критично важливими для забезпечення її надійності та точності її роботи. Використання машинного навчання та передових методів аналізу інформації дозволяє ефективно аналізувати та обробляти дані, забезпечуючи стабільність роботи системи та вчасно реагувати на збої.

Збої у системі можуть бути викликані різними факторами, включаючи фізичні перешкоди, помилки у роботі приймача, збої в роботі бортової інформаційної системи, або навіть електромагнітні завади, які впливають на якість сигналу, призводячи до неточностей у визначенні положення БПЛА. Аналіз цих факторів вимагає комплексного підходу, з використанням статистичних методів та інтелектуальних алгоритмів обробки даних.

Типові методи виявлення збоїв можна звести до двох підходів. Одним з них є метод коригування для зважування спостережень псевдодальності, який буде статистичний ряд виявлення несправностей [198], обчислюючи різницю між отриманим позиційним рішенням і рішеннями окремих компонентів підсистеми. Іншим є метод спостереження псевдодальності, який збирає статистику для виявлення збоїв шляхом обчислення очікуваних значень підсистеми [199], але даних методом важко визначити та усунути збої.

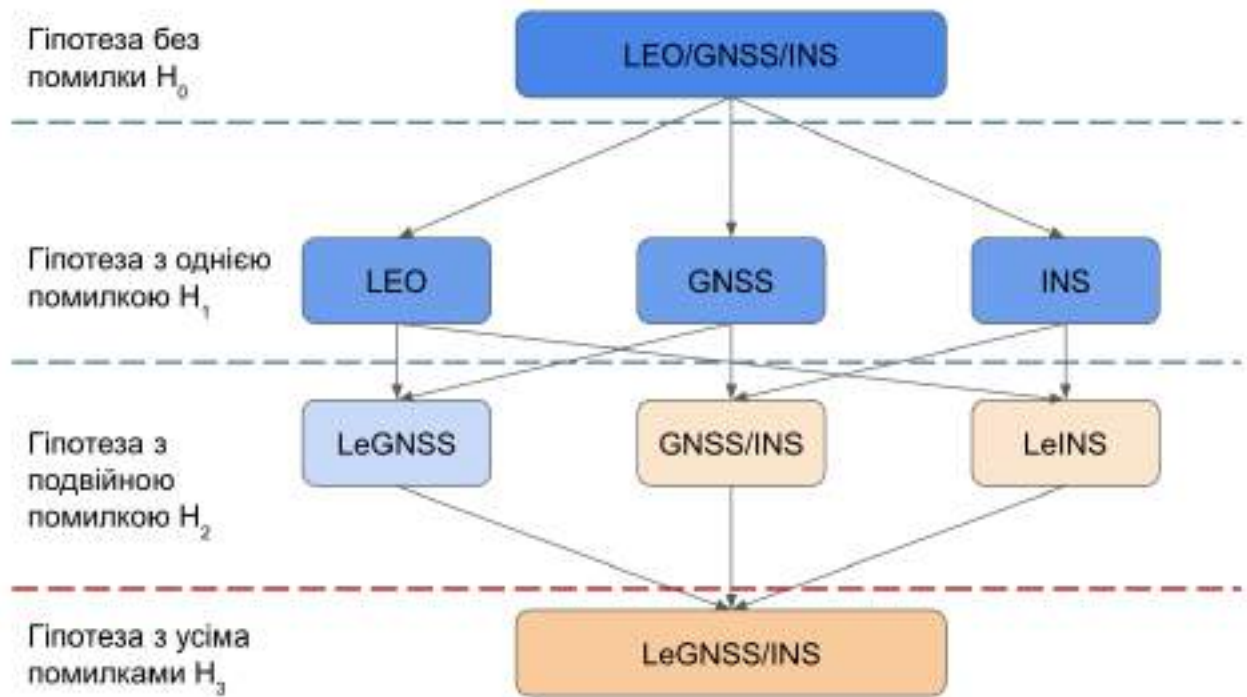


Рисунок 3.1. Типи можливих збоїв і помилок, які необхідно контролювати.

Дерево моніторингу ризиків цілісності інформації в підсистемі позиціонування БпЛА показано на рис. 3.1. Де жовтий колір позначає інформації, яку не потрібно контролювати на цілісність, а синій – інформації, що підлягає контролю. Тобто, збої другого рівня не повинні контролюватися. Таким чином, ми динамічно коригуємо підмножину несправностей, яку необхідно контролювати, відповідно до фактичної ситуації позиціонування з використанням багатьох джерел. Для реалізації такого контролю було запропоновано алгоритм внутрішнього контролю цілісності системи (рис. 3.2), що дозволяє виключати джерело некоректних даних із розрахунку на період поки надаються недостовірні дані.

Інтеграція декількох джерел інформації базується на використанні фільтра Калмана, що дозволяє синтезувати модель підсистеми. У цій моделі мінімально можлива одиниця несправності, ідентифікована на рівні підсистеми та її компонентів, визначається як окреме джерело даних про позиціонування.

Algorithm1: Multi-level autonomous integrity method**Input:** Results of INS, GNSS, LEO positioning sources information

Integrity risk of error and prior fault probability of each positioning source

Output: Integrity alarm

1. Establish subsystem level combination set:
INS/GNSS, INS/LEO, GNSS/LEO
2. **for** select combination set ($1 \leq i \leq 3$)
3. Calculate positioning data fault detection statistics $d_{s,i}$ and threshold T_s
4. **if** $d_{s,i} > T_s$
5. There is a fault in system and isolated the faulty positioning source F_{LeGNSS} .
6. **if** F_n is a non-redundant positioning source
7. Calculate w -detection statistics w_{source} and detection threshold δ_{source}
8. **if** $w_{source} < \delta_{source}$ for 5 consecutive times
9. Reintegrate F_n into the system positioning solution
10. **else**
11. Continue positioning data verification
12. **end if**
13. **else**
14. Calculate the number of subfilter J to be established.
15. **for** subfilter set ($1 \leq j < J$)
16. Summarize fault detection statistics and the score vector $\mathcal{S}_{all}(j)$.
17. **end for**
18. **if** \mathcal{S}_{all} has a unique zero value
19. Isolate the faulty data source and reintegrate F_{LeGNSS} , into the system.
20. Verify isolated faulty data source.
21. **else if** \mathcal{S}_{all} contains multiple zero values
22. Increase number of simultaneous faults and perform fault detection again.
23. **else**
24. Fault data source identification failed.
25. **end if**
26. **else**
27. The positioning data source F_{LeGNSS} is not fault.
28. **end if**
29. **end for**
30. Calculate system protection level ζ_{PL}
31. **if** $\zeta_{PL} > \zeta_{AL}$
32. LeGNSS data integrity alarm.
33. **else**
34. No alarm.
35. **end if**

Рисунок 3.2. Алгоритм оцінки цілісності багаточарової системи позиціонування.

3.4 Дискретний фільтр Калмана та інтелектуальний алгоритм корекції

Оптимізація процесу обробки даних полягає у виборі та налаштуванні алгоритмів, здатних ефективно виявляти та коригувати помилки. Це включає в себе використання фільтрів для згладжування даних, видалення аномалій та забезпечення консистентності даних. Важливою частиною є також визначення порогів чутливості алгоритмів, щоб збалансувати між чутливістю до помилок та уникненням хибних спрацювань.

Фільтр Калмана є одним з найбільш ефективних інструментів для обробки даних в системах позиціонування. У цьому розділі ми розглянемо розробку кастомного фільтра Калмана та інтелектуального алгоритму для виявлення збоїв, що значно підвищує точність та надійність підсистеми LeGNSS.

Дискретний фільтр Калмана використовується для оцінки станів системи на основі серії вимірювань, які містять шуми та інші випадкові похибки. Дискретний фільтр Калмана для системи LeGNSS оптимізований для обробки даних, які отримуються з різних джерел, таких як супутникові сигнали та інерціальні вимірювання.

Фільтр Калмана забезпечує механізм описаний рівнянням 3.3 для оптимального об'єднання інформація з кількох джерел. Рівняння визначення стану системи за наявності декількох джерел інформації має вигляд:

$$\widehat{P}_{k|k} = \widehat{P}_{k|k-1} + K_k \sum_i [z_{k,i} - H_i \widehat{P}_{k|k-1}], \quad (3.4)$$

де $z_{k,i}$ – вимірювання з i -ї системи, а H_i – модель спостереження для цієї системи. Крім того, інформація про швидкість і прискорення з кожної із систем також можуть бути об'єднані.

При об'єднанні інформації з декількох систем (3.5), об'єднана інформація про позиціонування може бути представлені як:

$$D_{LeGNSS} = \alpha \cdot D_{GPS} + \beta \cdot D_{LEO} + \gamma \cdot D_{INS}, \quad (3.5)$$

де α , β , γ – коефіцієнти злиття, які визначають внесок від кожної системи при об'єднанні інформації.

Принцип побудови оптимальної фільтрації у випадку динамічної лінійної системи описується системою рівнянь (3.6). Завдання полягає в тому, щоб знайти такі оцінки системи, при яких похибки будуть мінімальними:

$$\Delta e_{LeGNSS} = D(t) - \widehat{D_{LeGNSS}}(t). \quad (3.6)$$

Моделювання динаміки польоту вимагає використання рівнянь руху, зокрема Ньютона та Ейлера (3.7–3.8), дозволяє детально відтворити динаміку руху БПЛА. Для підвищення точності даних, отриманих від датчиків, застосовуються методи статистичної обробки, такі як фільтри Калмана, які ефективно зменшують шум та покращують якість інформації.

$$\hat{r} = \frac{F}{m} - \underline{g}, \quad (3.7)$$

$$\hat{\omega} = I^{-1}(T - \omega \times (I \omega)), \quad (3.8)$$

де r – прискорення центру мас, F – сила, що діє на БПЛА, m – його маса, g – прискорення вільного падіння, ω – кутова швидкість, T – момент сил, I – тензор інерції.

Що стосується похибок гіроскопа і акселерометра в INS, в цілях синтезу схеми буде прийматися, що вони мають експоненціальні кореляційні функції з великим часом кореляції. В такому разі вони описуються диференціальними рівняннями:

$$\hat{\omega}_{др} = -\frac{1}{T_{\omega}}\omega_{др} + \frac{1}{T_{\omega}}q_{\omega}; \quad (3.9)$$

$$\Delta\hat{a} = -\frac{1}{T_a}\Delta a + \frac{1}{T_a}q_a, \quad (3.10)$$

де q – випадковий процес типу «білий шум», T_a, T_{ω} – час кореляції.

Тому, для моделювання випадкових та детермінованих похибок датчиків MEMS, зокрема акселерометрів, гіроскопів та магнітометра, які використовуються в підсистемах БПЛА, було згенеровано модель похибок, яка включила систематичні (детерміновані) і випадкові похибки (3.11), які можуть виникнути через різні причини, такі як зміна температури, механічні напруження, електромагнітні перешкоди тощо.

$$\delta_{MEMS} = \delta_{bias}(t) + \delta_{scale} + \delta_{ortho} + \delta_{drift}(t) + \delta_{noise}(t), \quad (3.11)$$

де, $bias$ – похибка зсуву з часом, $scale$ – постійна величина, яка відображає відхилення в масштабі вихідних даних датчика, $ortho$ – постійна величина, що відображає похибку неортогональності осей датчика, $drift$ – швидкість зміни дрейфу датчика з часом, $noise$ – рівень шуму, або випадкові коливання вимірювань що не можуть бути передбачені або виправлені, (t) – функція часу, яка відображає швидкість зміни похибки з часом.

Щоб використовувати фільтр Калмана, потрібно сформулювати математичну модель процесу, який оцінюється, та модель вимірювань у відповідній формі [200].

$$X_t = A_t X_{t-1} + G(t)W(t). \quad (3.12)$$

Це рівняння описує, як поточний стан X_t системи що еволюціонує з попереднього стану X_{t-1} через матрицю переходу станів A_t , з додаванням шуму процесу $W(t)$, який моделюється через матрицю контролю $G(t)$.

Зв'язок з вектором оцінюваних станів встановлюється рівнянням:

$$Z_t = H(t)X_t + V_t, \quad (3.13)$$

де, Z_t – це вимірювання, $H(t)$ – матриця спостереження, яка перетворює стан X_t в вимірювання, V_t – шум вимірювання, а матриця спостережень в такій системі дорівнює:

$$H = [1000 \ 0100 \ 0010 \ 0001]. \quad (3.14)$$

Для того щоб отримати коефіцієнти корекції для фільтра Калмана, необхідно розв'язання рівняння Рикати [201], що описує оновлена коваріаційна матриця похибки оцінки:

$$P_t = A_t P_{t-1} A_t^T - P_{t-1} H_t^T R_t^{-1} H_t P_{t-1} + G Q G^T. \quad (3.15)$$

де вектори Q – коваріаційна матриця шуму, що характеризує вплив шуму на процеси, а R – оцінка невизначеності даних вимірювань, яка залежить від точності та надійності джерел інформації.

Коефіцієнти корекції фільтра Калмана, що використовуються для оновлення оцінки стану на основі нових вимірювань, розраховуються:

$$K_{\Phi t} = P_t H_t^T R_t^{-1}. \quad (3.16)$$

Наведені рівняння (3.12–3.16) дозволяють LeGNSS системі із фільтром Калмана ефективно інтегрувати нові вимірювання, оновлюючи оцінку стану системи та її невизначеність (або коваріацію), щоб мінімізувати загальну похибку оцінки позиціонування під час польоту БПЛА. Вони є фундаментальними для розуміння та реалізації дискретного фільтра Калмана та

описують основні механізми оновлення оцінки стану системи, прогнозування та корекції на основі вимірювань.

У контексті синтезу підсистеми LeGNSS із застосуванням фільтра Калмана, важливо відзначити, що рівняння коваріації може бути розв'язане автономно, окремо від реальних вимірювань Z_t . Це означає, що обчислення можна проводити незалежно від основного алгоритму фільтра, і навіть попередньо.

Для розв'язання рівнянь коваріації, першочергово потрібно провести оцінювання та встановлення значень ініціації P_0 на основі аналізу точності даних стосовно стану вектора X_t , а також початкових параметрів X_0 , що діють до активації фільтра. Зазвичай, це робиться шляхом задання головної діагоналі $p_{ij} < 0$ для вихідної дисперсії похибок оцінювання, припускаючи, що $p_{ij} = 0$ для $i \neq j$.

У подальшому, процес вирішення рівнянь коваріації досягає стабільного стану P_{robust} , що задовольняє наступному рівнянню:

$$P_{robust} H_t^T R_t^{-1} H_t P_{robust} = A P_{robust} + P_{robust} A^T + B R_x B^T. \quad (3.17)$$

де, P_{robust} – усталена коваріаційна матриця похибок оцінки, що відображає невизначеність в оцінці стану системи після того, як вона стабілізується через ряд ітерацій фільтра Калмана, B – матриця представляє вплив контролю або зовнішніх впливів на стан системи (наприклад, вплив вітрових збурень), використовується для врахування впливу зовнішніх збурень або контролю на сталу коваріаційну матрицю, R_x – коваріаційна матриця зовнішніх збурень або контролю, яка представляє невизначеність або помилки в цих зовнішніх впливах.

Завдання проведення оцінки параметрів схеми полягає у знаходженні оптимальних режимів фільтрації шуму в інформації, спектр якого визначений R_z , та значення корисної інформації, спектр якої визначений змінною R_x . Враховуючи динамічні характеристики описаного процесу, представлені матрицею коефіцієнтів A . Отримання стійкої і дискретної системи вимагає усунення випадкових збурень, тобто $R_x = 0$. В такому випадку завжди буде виконуватися рівність $p_{ij} < 0$ та $P_{robust} = 0$, а похибка оцінювання з часом

зменшується до нуля. Шуми вимірювачів при цьому не матимуть впливу на похибку оцінювання, оскільки вимірювачі відключаються від схеми фільтрації при $P = 0$ і, тоді рівняння для коефіцієнтів корекції буде $K_{\phi} = 0$ [202].

Під час розробки фільтрів обробки інформації важливо враховувати цей аспект, включаючи до збурень V_x не тільки фактичні фізичні збурення, такі як зсув вітру, але й потенційні неточності у визначенні моделей процесів та вимірювань керування, що виникають під час польоту БпЛА.

Рівняння 3.17 описує, як усталена коваріаційна матриця взаємодіє з динамікою системи, похибками вимірювань та зовнішніми збуреннями. Це дозволяє збалансувати фільтрацію похибок та корисного сигналу з урахуванням динамічних характеристик процесу та зовнішніх впливів.

В умовах високої динаміки та змінних умов експлуатації БпЛА, важливо мати засоби ефективної діагностики та корекції можливих збоїв системи. Методи машинного навчання забезпечують великі можливості для ідентифікації та діагностики збоїв в системі LeGNSS. Штучні нейронні мережі, підтримувані алгоритми глибокого навчання, можуть виявляти аномалії в даних, що є показниками потенційних збоїв. Ці методи навчаються на великих об'ємах даних, що дозволяє їм адаптуватися до різних умов роботи та виявляти збої на ранніх стадіях та усувати їх негативний вплив.

Машинне навчання забезпечує раннє виявлення та класифікацію збоїв, тоді як фільтр Калмана використовується для неперервної корекції та оновлення даних. Цей підхід дозволяє системі ефективно адаптуватися до різних умов роботи та підтримувати високий рівень точності та надійності.

Інтеграція дискретного фільтра Калмана з інтелектуальними алгоритмами виявлення збоїв створює потужну систему для бортової обробки даних БпЛА та побудови високонадійної підсистеми позиціонування. Фільтр Калмана забезпечує точну оцінку стану системи, в той час як алгоритми машинного навчання дозволяють виявляти та коригувати помилки, збої приймачів або аномалії в даних [203–205].

Алгоритмічний підхід для виявлення збоїв включає в себе використання штучного інтелекту та машинного навчання для аналізу вимірювань та ідентифікації відхилень. Ці алгоритми можуть виявляти неправильні вимірювання або збої в роботі системи, дозволяючи своєчасно реагувати та коригувати помилки. Практична реалізація алгоритму оцінки цілісності даних про позиціонування в підсистемі LeGNSS показана на рис. 3.3.



Рисунок 3.3. Оцінка цілісності інформації про позиціонування в підсистемі LeGNSS.

Наведені вище формули відображають залежності та взаємодію при об'єднанні даних GPS, LEO та INS з використанням оптимального злиття даних використовуючи дискретний фільтр Калмана. Фактичні значення ваг, коефіцієнтів злиття та інших параметрів можуть залежати від конкретного застосування, якості отриманої інформації від кожної із системи, а також від очікуваної точності, частоти та надійності об'єднаних даних.

3.5 Використання псевдореальних даних для моделювання та валідації підсистеми

Було обрано підхід використання псевдо реальних даних під час моделювання, для того, щоб збалансувати використання наявних даних та керованість симульованого середовища. Для підвищення реалістичності експериментів було використано пакет математичного моделювання Matlab. Цей інструмент допоміг згенерувати квазіреальні динамічні траєкторії [206–207], які

точно імітують реальні рухи БпЛА та поведінку супутників, дозволяючи при цьому контрольоване введення змінних, таких як зовнішні перешкоди і завади.

Під час моделювання динаміки польоту БпЛА, важливо врахувати аеродинамічні сили, зміни в густині повітря, та інші зовнішні фактори, такі як вітер та турбулентність, які впливають на рух апарату [208–210].

Аеродинамічний опір, який впливає на БпЛА під час польоту, може бути точно змодельованим за допомогою рівняння (3.18):

$$F_{drag} = -\frac{1}{2}\rho C_d A v^2, \quad (3.18)$$

де ρ – густина повітря, C_d – коефіцієнт опору, A – площа поперечного перерізу, та v – швидкість об'єкта. Дана рівняння дозволяє врахувати аеродинамічний опір повітря, для створення більш точної симуляції в різних умовах польоту. Крім того, це допомагає визначити оптимальні команди управління для БпЛА, щоб ефективно досягати заданих точок маршруту, враховуючи його фізичні характеристики та обмеження.

Сценарії моделювання відтворюють різноманітні аспекти польоту та позиціонування БпЛА, включно з плануванням маршрутів, реагуванням на управлінські сигнали, динамікою польоту та функціонуванням систем позиціонування. Методи оптимальної фільтрації, на кшталт фільтра Калмана, застосовуються для оцінки станів системи на основі зашумлених вимірювань. Ці методи базуються на теорії ймовірностей та статистичних моделях для корекції похибок вимірювань та прогнозування майбутніх станів системи.

3.6 Висновки до третього розділу

Існує багато систем і алгоритмів, що використовуються для підвищення точності даних про позиціонування, а більшість із них застосовують аналіз та використання надлишкової інформації, отриманої з різних джерел. Це дозволяє

підсистемам БПЛА надавати високоточну інформацію про позиціонування після відповідної корекції та фільтрації.

Прості системи корекції, які зазвичай використовують меншу обчислювальну потужність та вирізняються високою надійністю через свою простоту, можуть бути ефективними в певних випадках. Однак, фільтр Калмана, застосований у системі LeGNSS, є одним із найбільш перспективних та надійних підходів. Фільтр Калмана забезпечує високу точність позиціонування, але це супроводжується необхідністю в додаткових обчислювальних ресурсах та певними особливостями його застосування.

Використання фільтра Калмана в системі LeGNSS дозволяє інтегрувати дані з різноманітних джерел, ефективно фільтруючі та коригуючі дані систем позиціонування, що значно підвищує точність та надійність орієнтування БПЛА. Цей підхід відкриває нові можливості для використання БПЛА в різноманітних умовах, забезпечуючи точність та стабільність, необхідні для вирішення складних місій.

РОЗДІЛ 4

ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ LeGNSS

4.1 Схема синтезованої системи в Matlab

Для дослідження характеристик та параметрів роботи робастної LeGNSS підсистеми позиціонування БПЛА з корекцією та оптимальною фільтрацією похибок, в розділі 3 було складено схему синтезованої математичної моделі підсистеми LeGNSS, що представлена на рис. 4.1.

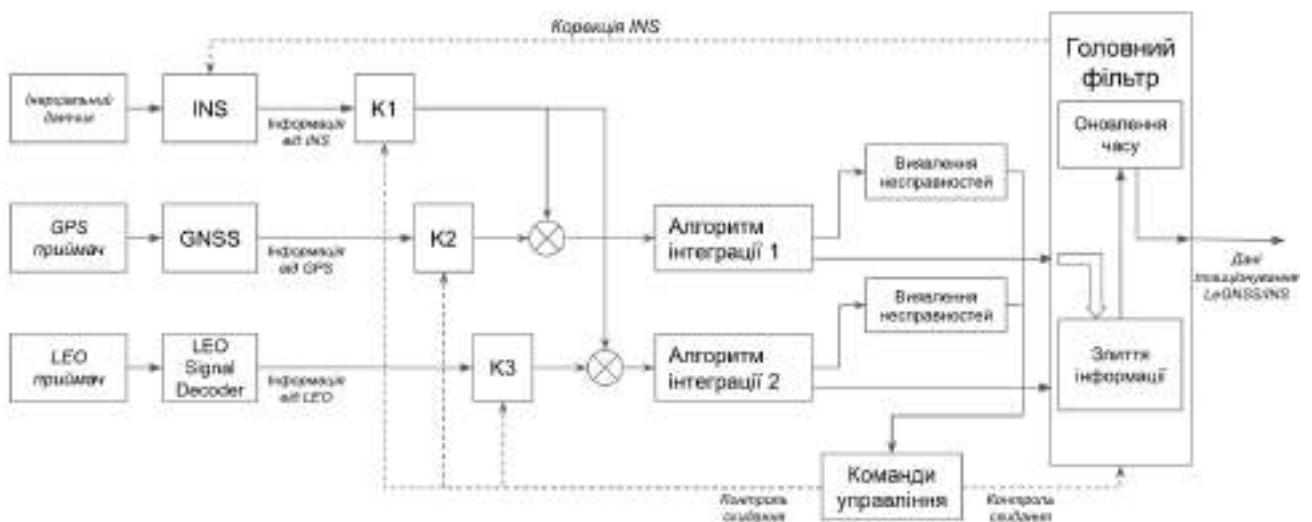


Рисунок 4.1. Схема комплексної підсистеми позиціонування LeGNSS.

Для проведення такого моделювання роботи комплексної підсистеми позиціонування LeGNSS було розроблено алгоритм (рис. 4.2), який відображає алгоритм для моделювання роботи бортових підсистем БПЛА. Він включає ініціалізацію даних, планування польоту, генерацію управлінських сигналів, та моделювання різних систем позиціонування. Кожен крок виконується за допомогою відповідного алгоритму, що забезпечує деталізоване відтворення процесів управління та позиціонування БПЛА.



Рисунок 4.2. Алгоритму моделювання інтегрованої багатошарової системи позиціонування LeGNSS в Matlab.

Даний алгоритм (рис.4.2) призначений для моделювання роботи інтегрованої системи позиціонування LeGNSS БпЛА. Він виконує послідовність кроків для імітації польоту БпЛА, використовуючи різні моделі позиціонування, такі як MEMS, GPS, системи LEO та LeGNSS.

На основі даного алгоритму було підготовлено код програми моделювання в Matlab (додаток Г). Він також містить функцію самодіагностики для перевірки цілісності інформації в системі (рис. 3.2), наявності усіх необхідних файлів перед початком моделювання та реалізує низку сценаріїв, що охоплюють підготовку даних, планування заданого маршруту та моделювання польоту БпЛА, а також моделювання роботи обраних підсистем позиціонування.

Для моделювання роботи системи в динамічних режимах польоту складено масиви даних про позиціонування БпЛА для відпрацювання місій, масиви даних середньоквадратичних значень похибок позиціонування для супутникових джерел інформації, масиви даних середньоквадратичних значень похибок позиціонування для інерціального джерела інформації, тощо.

Для наближення умов моделювання до природніх, було створено масиви псевдо реальних даних, які визначають зовнішній динамічний вплив рози вітрів на процес моделювання польоту БпЛА.

Для аналізу точності сигналів бюджетних приймачів інформації, що приблизно відповідають рівню точності позиційних приймачів в смартфонах,

було створено масив тестових даних позиціонування та використано середньоквадратичні похибки псевдодальностей для систем MEMS, GPS і LEO, виконано порівняння отриманих від них значення для різних етапів польоту під час виконання місії БпЛА.

Обробка інформації в бортових підсистемах БпЛА зосереджена на моделюванні динаміки польоту для точного відтворення поведінки апарату за допомогою аналізу впливу аеродинамічних та інерційних сил, що було розглянуто раніше в розділі 3.5. Перш за все, моделювання наближене до реальних умов, включає аеродинамічні та інерційні сил, що впливають на динаміку БпЛА під час польоту.

4.2 Аналіз синтезованої схеми підсистеми позиціонування

Розглянемо детально всі елементи схеми та послідовність роботи підсистеми позиціонування представленої на рис. 4.1.

На схемі зображено архітектуру підсистеми позиціонування LeGNSS. Вона демонструє інтеграцію різних джерел даних і методів обробки інформації для забезпечення точного позиціонування БпЛА. Детальний опис роботи системи:

1) *Вхідні дані*: Система отримує дані від трьох основних джерел: інерційної системи позиціонування (INS), системи глобального позиціонування (GNSS), та сигналів низькоорбітальних супутників (LEO), та інформацію аеродинамічні та інерційні параметри моделі БпЛА (рис.4.3);

2) *Корекція INS*: Дані від INS проходять через корекцію (позначено як $K1$), яка виправляє можливі помилки або невизначеності в інерційних вимірюваннях;

3) *Інтеграція даних*: Дані з GNSS і LEO об'єднуються з виправленими даними INS через алгоритми інтеграції (Алгоритм інтеграції 1 та Алгоритм інтеграції 2), кожен з яких використовує свої коефіцієнти ($K1$, $K2$ і $K3$). На виході алгоритму інтеграції першого порядку формується оптимальна оцінка похибок кожної системи $\hat{S}_{MEMS+GPS}$ та $\hat{S}_{LEO+INS}$, яка потім використовується для статистичної оцінки фактичного положення БпЛА, в результаті чого

підвищується точність визначення положення для кожного набору інтегрованих даних;

4) *Головний фільтр*: На вхід головного фільтру подається оцінка похибок інерціальної MEMS системи позиціонування, системи супутникового позиціонування та вимірні дані позиціонування від LEO супутникової мережі, інформація з яких була оброблена в алгоритму інтеграції першого порядку на вході в підсистему LeGNSS, що є результатом компенсації істинної інформації про положення БПЛА. Фільтр відповідає за видалення шумів, виправлення похибок та вироблення точної оцінки стану.

5) *Виявлення збоїв та несправностей*: В процесі роботи фільтра відбувається постійний контроль за збоями або можливими несправностями в підсистемі за допомогою алгоритму контролю цілісності, що був описаний в розділі 3.3. Цей алгоритм інтегрований у структуру системи та має на меті забезпечити усунення хибних або скомпрометованих даних, що використовуються для отримання інформації про позиціонування.

6) *Вихідні дані*: На виході алгоритму інтеграції другого порядку ми отримуємо статистичну оцінку положення БПЛА за вирахуванням раніше оцінених похибок позиціонування. В результаті такої роботи підсистеми LeGNSS на виході фільтра формуються оцінки щодо поточного позиціонування БПЛА, які можуть використовуватися для орієнтації в просторі, керування чи інших застосувань.

У контексті запропонованої підсистеми LeGNSS із використанням оптимальної фільтрації Калмана, роботу кожного кроку фільтра можна розділити на два основні етапи:

1) *Прогнозування*: На цьому етапі обчислюється прогнозований вектор стану підсистеми, виходячи з її попереднього стану. Це включає оцінку положення, швидкості та інших важливих параметрів стану LeGNSS на основі поточних даних та моделі динаміки системи. Прогнозування допомагає передбачити, де підсистеми буде знаходитися в наступний момент часу (рис.4.5), перед тим як будуть отримані нові вимірювання.

2) *Корекція*: В підсистему LeGNSS надходить інформація щодо поточних вимірювань, наприклад, від GPS та інших датчиків. Ці дані використовуються для уточнення прогнозованого значення вектора стану. За допомогою обчислення коефіцієнтів Калмана, система коригує прогнозований стан, зважаючи на нову інформацію, отриману від вхідних джерел інформації. Це дозволяє уточнити оцінку стану системи LeGNSS, зменшуючи похибку між прогнозованим та реальним станом.

```

% Пілотажні параметри БпЛА
max_horisontal_speed = 62/3.6; % максимальна швидкість зміни положення БпЛА (м/с) по осям X та Y
max_vertical_speed = 12; % максимальна швидкість набору висоти БпЛА (м/с) по осі Z, але зниження швидкості по осі Z відбувається зі швидкістю зниження що не перевищує значення g (прискорення вільного падіння) мінус значення аеродинамічного опору
accel_time = 14; % час (с) протягом якого БпЛА змінює швидкість від мінімальної до максимальної
max_tilt_angle = 25; % максимальний дозволений кут нахилу до горизонту (градусів)
max_yaw_rate = 25; % максимальна швидкість обертання БпЛА навколо осі Z (градусів/с)
max_roll_angle = 15; % стандартний кут крену під час польоту в горизонтальній площині (градусів)
max_pitch_roll_rate = 30; % максимальна швидкість обертання навколо осей Y та X (градусів/с)
current_azimuth = 0; % показання азимута напрямку орієнтування БпЛА в момент початку місії (градусів, азимут на північ)

% Параметри БпЛА
uav_m = 4.5; % маса квадрокоптера, кг
uav_J = diag([6.158e-2, 6.158e-2, 1.106e-1]); % Матриця інерції БпЛА
uav_Ct = 3.691e-5; % Коефіцієнт тяги пропелера
uav_Cd = 1.170e-1; % Коефіцієнт аеродинамічного опору
uav_rho = 1.225; % Густина повітря, кг/м^3
uav_A = 0.25; % Площа перерізу квадрокоптера, м^2 (приблизне значення)

% Параметри систем позиціонування БпЛА (RMS errors)
error_bounds = struct(); % Define RMS errors for different satellite types (min error (m), max error(m))
error_bounds.GEO = [2.6, 3.6];
error_bounds.MEO = [2.76, 6.96];
error_bounds.GPS = [2.77, 5.73];
error_bounds.LEO = [6.5, 9.5];

gps_update_rate = 1; % Гц, частота оновлення даних GPS
leo_update_rate = 25; % Гц, частота оновлення даних LEO

```

Рисунок 4.3. Фрагмент коду з масивом вихідних даних для проведення моделювання роботи БпЛА і граничних похибок його підсистем в Matlab.

Даний підхід дозволяє системі LeGNSS постійно оновлювати оцінку свого стану, реагуючи на нові дані та коригуючи будь-які помилки, виявлені за допомогою спостережень, що забезпечує високу точність та надійність роботи підсистеми позиціонування.

4.3 Створення польотного завдання та сигналів керування БпЛА

Використання інструментів Matlab для автоматизованого складання та візуалізації маршрутів польотів дозволяє детально планувати та оцінювати траєкторію БпЛА. Керування польотом здійснюється через алгоритм планування переміщення та управління переміщенням, який контролює швидкість, кути нахилу, курс та інші польотні параметри БпЛА, забезпечуючи точне слідування за заданим маршрутом.

Для генерації плану польоту було задано початкові умови (рис. 4.3), що включають координати початку та завершення місії, обмеження по мінімальній та максимальній висоті польоту над поверхнею, перелік завдань на політ, з 4 об'єктів навколо яких БпЛА має виконати круговий обліт.

```
% Початкова та кінцева точка польоту
flight_startPoint = [50.43935, 30.43105, 0]; % Початкова точка польоту (широта, довгота, висота в метрах)
flight_endPoint = [50.43970, 30.43072, 0]; % Кінцева точка польоту (широта, довгота, висота в метрах)
% Діапазон дозволеної висоти польоту
flight_altitude = [50, 150]; % Діапазон дозволеної мінімальної та максимальної висоти польоту по
маршруту над землею
% Об'єкти для аерофотозйомки
flight_objectives = [50.4495, 30.5253, 100; 50.4481, 30.5340, 140;
50.4393, 30.5553, 50; 50.4064, 30.4592, 200];
```

Рисунок 4.3. Фрагмент коду вихідних параметрів моделі БпЛА для створення плану польоту в Matlab.

Для генерації операціоналізованого польотного завдання було розроблено алгоритм автоматичного планування та складання операційного плану польоту

(рис. 4.4). Наведений фрагмент коду програми починається з ініціалізації початкової точки маршруту, після чого забезпечує автоматизований злет до заданої висоти, вибраної на основі визначених мінімальної та максимальної висот польоту. Програма обробляє кожну цільову точку на заданому маршруті, встановлюючи кругові обльоти навколо кожної з них та задаючи тривалість статичного зависання над цими точками.

```

% Ініціалізація маршруту польоту
uav_flight_path = flight_startPoint;
% Take off to desired altitude
desired_altitude = max(min(min_altitude, max_altitude), flight_startPoint(3));
uav_flight_path = [uav_flight_path; flight_startPoint(1), flight_startPoint(2), desired_altitude];
for i = 1:size(flight_objectives, 1)
    obj = flight_objectives(i, 1:2);
    altitude = flight_objectives(i, 3); % висота не менше min_altitude метрів
    if altitude < min_altitude
        altitude = min_altitude; % встановлення мінімальної висоти min_altitude метрів
    end
% Параметри для додавання кругових польотів навколо об'єктів
radius = 0.0003; % приблизно 30 метрів в географічних координатах
circle_points = 36; % кількість точок для кругового обльоту
% Додавання кругових польотів навколо об'єктів
for j = 1:circle_points
    angle = 2 * pi * j / circle_points;
    x = obj(1) + radius * cos(angle);
    y = obj(2) + radius * sin(angle);
    uav_flight_path = [uav_flight_path; x, y, altitude];
end
hover_time = 10 / 60; % час зависання в хвиликах (10 секунд = 10/60 хвилини)
desired_rate = 6; % так як ми хочемо зберігати точку кожні 10 секунд, то це буде 6 разів на хв
hover_points = round(hover_time * desired_rate); % визначаємо кількість точок для збереження
for j = 1:hover_points
    uav_flight_path = [uav_flight_path; uav_flight_path(end,1), uav_flight_path(end,2), altitude];
end
end
% Додавання кінцевої точки польоту (Landing to 0 altitude after reaching the final latitude and longitude)
uav_flight_path = [uav_flight_path; flight_endPoint(1), flight_endPoint(2), desired_altitude]; %вихід на
координату посадки
uav_flight_path = [uav_flight_path; flight_endPoint]; % посадка на визначену точку

```

Рисунок 4.4. Фрагмент коду для генерації масиву даних маршруту польоту.

Після завершення обльоту по вказаних точках, БПЛА виконує політ до наступних цілей, завжди дотримуючись безпечної висоти над рівнем поверхні. Після завершення місії відбувається приліт і посадка БПЛА на визначену кінцеву точку. Запропонована програма є оптимальною для завдань, що вимагають точного планування маршрутів польоту, таких як аерофотозйомка, моніторинг або картографування.

Для отримання карти фактичних висот поверхні по маршруту слідування БПЛА, щоб забезпечити дотримання висотних обмежень над поверхнею землі, було використано Google Maps Elevation API та скрипт, що отримує дані для кожної точки маршруту і операціоналізував план маршруту польоту по висоті над поверхнею у відповідності до заданого завдання (рис. 4.5–4.6).

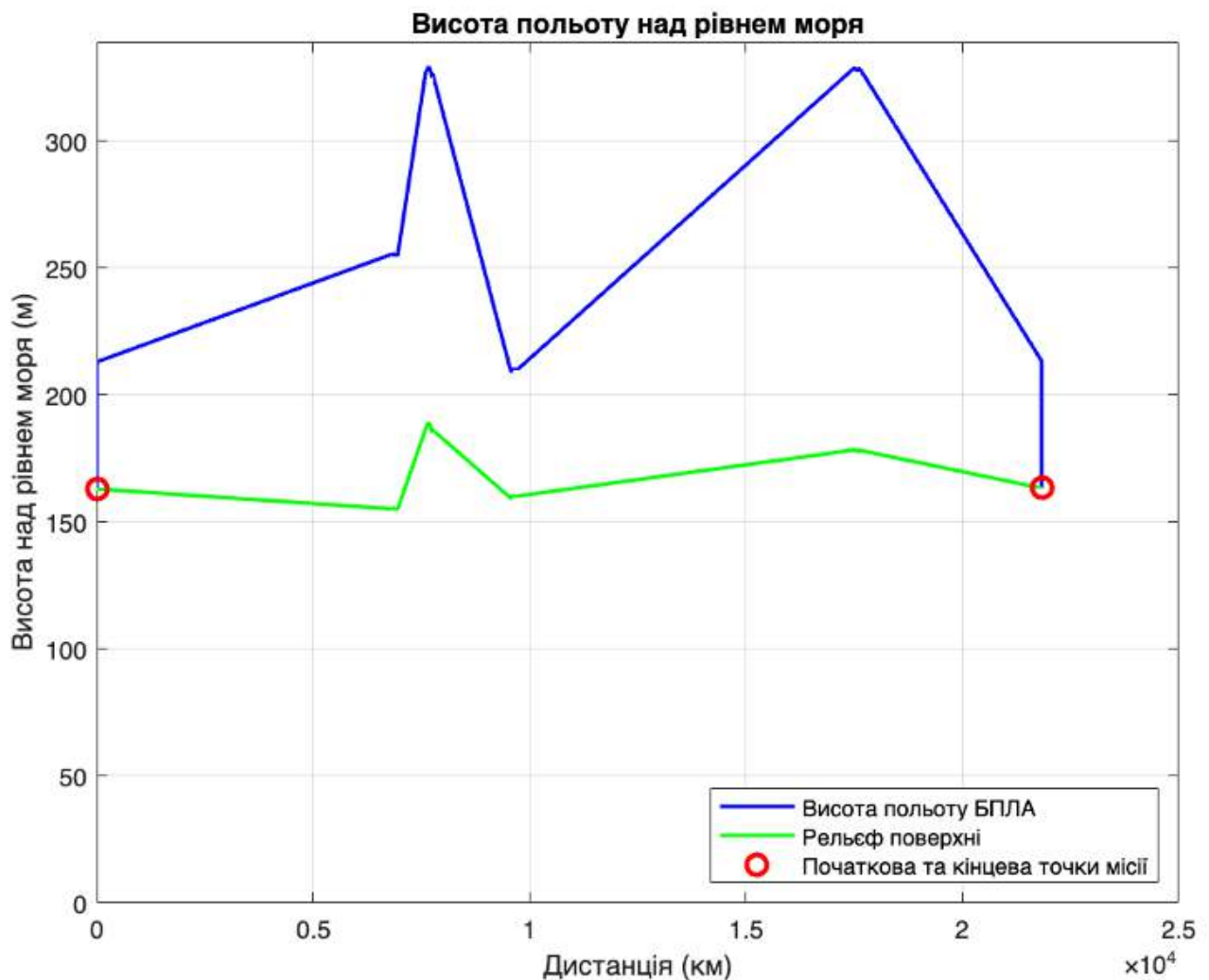


Рисунок 4.5. Визначення висоти польоту на місцевості.


```

%% Запит карти висот плану польотів до Google Maps Elevation API
% Файл: get_elevation_API.m
function [elevation, api_request_count] = get_elevation_API(latitude, longitude, api_key, existing_elevation,
api_request_count)
    fprintf('Створено запит до Google Maps Elevation API \n'); % Виведення сповіщення про відпрацювання
    % Якщо вже є дійсне значення висоти, повертаємо його
    if existing_elevation ~= 0
        elevation = existing_elevation;
        return;
    end
    % Створюємо URL для Google Maps Elevation API
    url = sprintf('https://maps.googleapis.com/maps/api/elevation/json?locations=%f,%f&key=%s', latitude,
longitude, api_key);

    % Робимо запит до веб-сервісу
    response = webread(url);
    % Перевіряємо статус відповіді
    if strcmp(response.status, 'OK')
        elevation = response.results(1).elevation;
        api_request_count = api_request_count + 1; % Збільшуємо лічильник запитів API
    else
        error('Не вдалося отримати висоту: %s', response.status);
    end
end

```

Рисунок 4.6. Фрагмент коду для отримання фактичних висот поверхні по маршруту польоту БпЛА.

Модель симуляції польоту БпЛА включає інтеграцію та обробку даних від декількох систем позиціонування, що забезпечує комплексний аналіз поведінки бортових підсистем в різних умовах. Використання псевдо реальних даних під час моделювання роботи інтегрованої системи позиціонування дозволило створити деталізовану 3D-модель польоту БпЛА, що відображає реалістичну поведінку з урахуванням системи корекції інформації про позиціонування нанесену на топографічну карту (рис. 4.7).

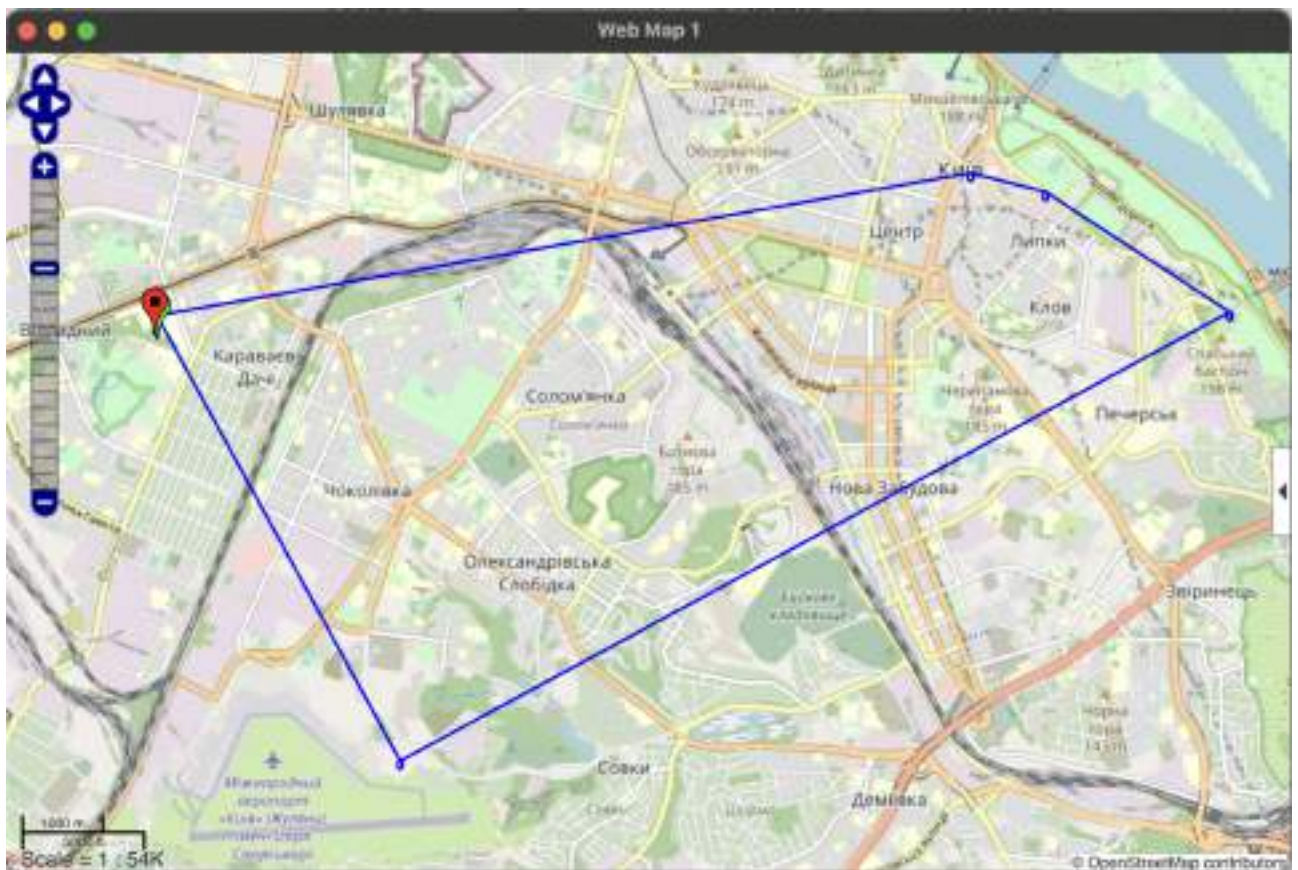


Рисунок 4.7. Генерація траєкторії польоту для виконання місії БПЛА.

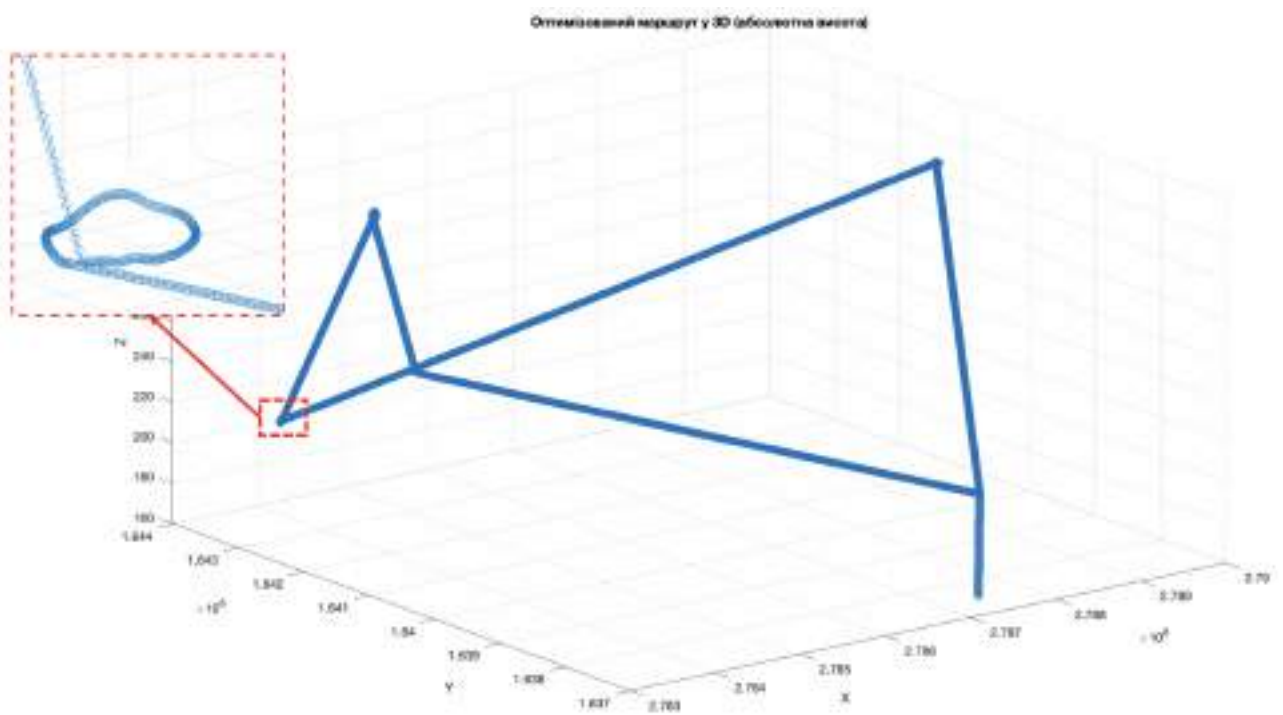


Рисунок 4.8. Траєкторія польоту БПЛА з операціоналізованим маршрутом.

Для врахування реальних умов польоту точок та проходження місії визначено допуск для проходження плану польоту – 2 м, для гарантування оптимальної динаміки польоту складено маршрут із сукупності кіл, центри яких проходять через точки маршруту із заданою відстанню не більше 1 м. Даний підхід гарантує оптимальне проходження всіх точок маршруту із заданою точністю та в оптимальних режимах польоту БпЛА (рис. 4.8).

Отриманий оптимізований та операціоналізований план польоту дозволяє сформувати масив керуючих сигналів для управління польотом БпЛА (рис. 4.10–4.11). Для цього отриманий операціоналізований план польоту обробляється алгоритмом генерації масиву управляючих сигналів, для забезпечення ефективного та оптимального управління БпЛА під час польоту в заданих умовах. Фрагмент код програми для генерації масиву управляючих сигналів наведено на рис. 4.9.

Шляхом інтеграції отриманих керуючих сигналів можна отримати фактичні значення зміни швидкості та прискорення БпЛА (рис. 4.12–4.13), які дозволяють керувати його переміщення в просторі під час виконання місії.

На кожній ітерації запропонований алгоритм обчислює вектор напрямку між поточною точкою і наступною. Цей вектор є різницею координат між двома послідовними точками маршруту. З цього вектора напрямку скрипт обчислює його величину, яка по суті є евклідовою відстанню між точками.

Отримана величина використовується і як міра відстані, і як знаменник при обчисленні кутових відхилень. Отримані кутові відхилення та прискорення відповідають кутам у 3D-просторі і дозволяють обчислити зміни складових елементів швидкості БпЛА в просторі.

```

% Визначення кількості точок у refined_path
num_points = size(refined_path, 1);

% Ітерація через усі точки маршруту для розрахунку дельта кутових відхилень і відстані між ними
for i = 1:num_points-1
    % Розрахунок вектору напрямку між поточною та наступною точками
    direction_vector = refined_path(i+1, :) - refined_path(i, :);
    % Розрахунок довжини вектору напрямку
    magnitude = norm(direction_vector);

    % Перевірка на нульову магнітуду, щоб уникнути ділення на нуль
    if magnitude == 0
        direction_vector_angles(i, :) = [0, 0, 0];
    else
        % Розрахунок дельта кутових відхилень
        alpha_angle = acos(direction_vector(1) / magnitude); %  $\Delta\alpha$ 
        beta_angle = acos(direction_vector(2) / magnitude); %  $\Delta\beta$ 
        gamma_angle = acos(direction_vector(3) / magnitude); %  $\Delta\gamma$ 
        % Заміна NaN на 0
        if isnan(alpha_angle)
            direction_vector_angles(i, 1) = 0;
        else
            direction_vector_angles(i, 1) = alpha_angle;
        end
        if isnan(beta_angle)
            direction_vector_angles(i, 2) = 0;
        else
            direction_vector_angles(i, 2) = beta_angle;
        end
        if isnan(gamma_angle)
            direction_vector_angles(i, 3) = 0;
        else
            direction_vector_angles(i, 3) = gamma_angle;
        end
    end
    % Розрахунок відстані між поточною та наступною точками
    distances(i) = magnitude;
end

```

Рисунок 4.9. Фрагмент коду для отримання керуючих сигналів зміни швидкості та прискорення БПЛА під час виконання місії.

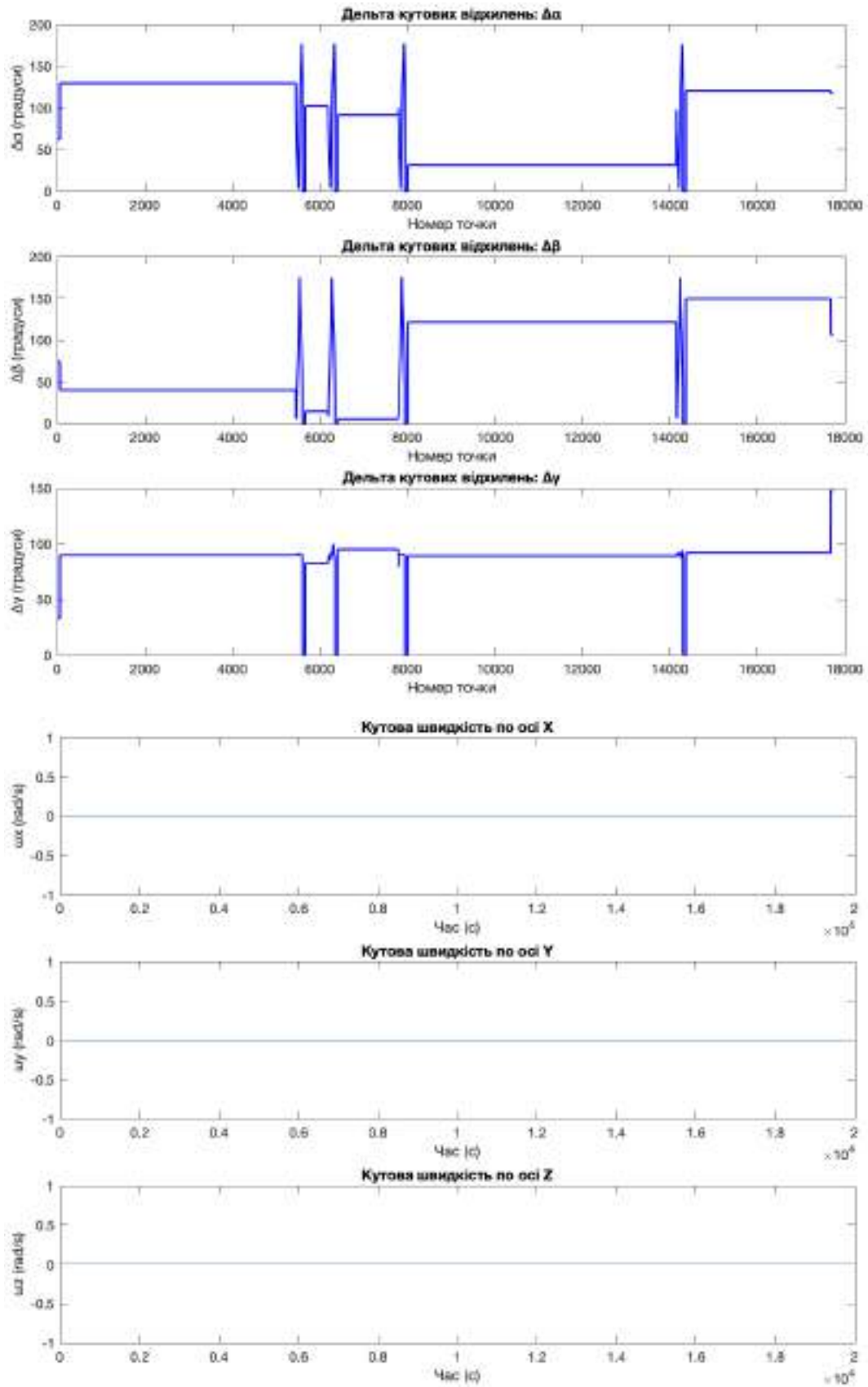


Рисунок 4.10. Дані щодо управляючих сигналів зміни курсу та кутових швидкостей БПЛА.

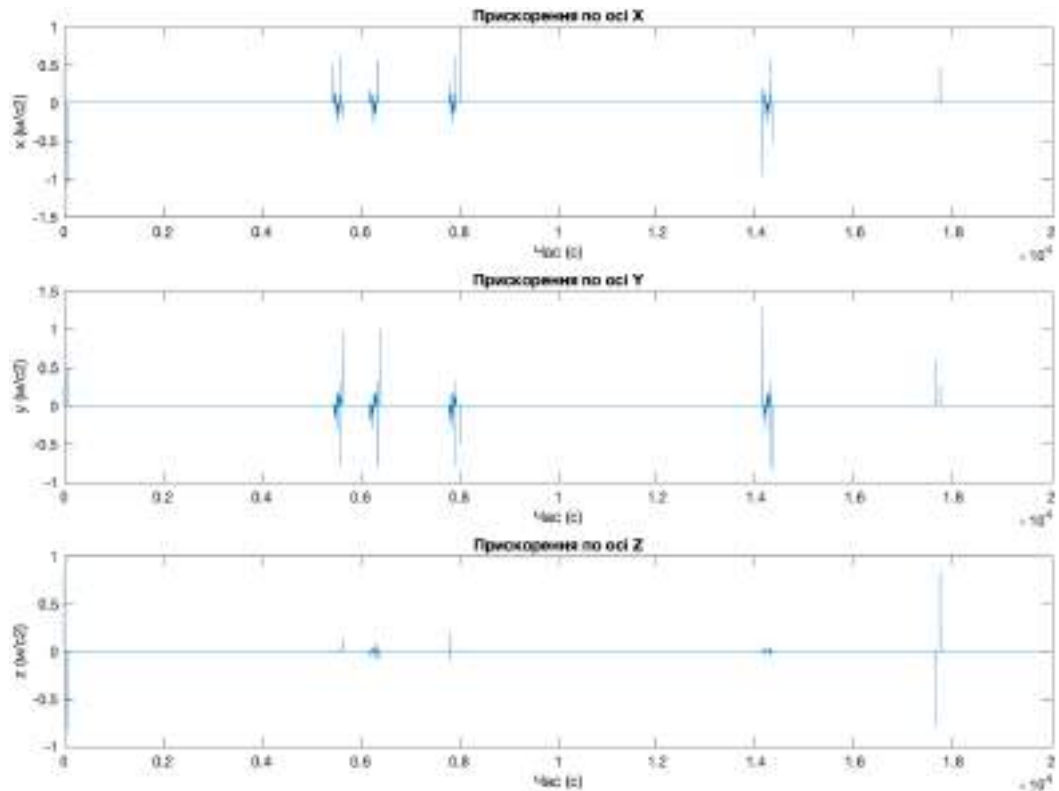


Рисунок 4.11. Дані щодо управляючого сигналу зміни прискорення БПЛА.

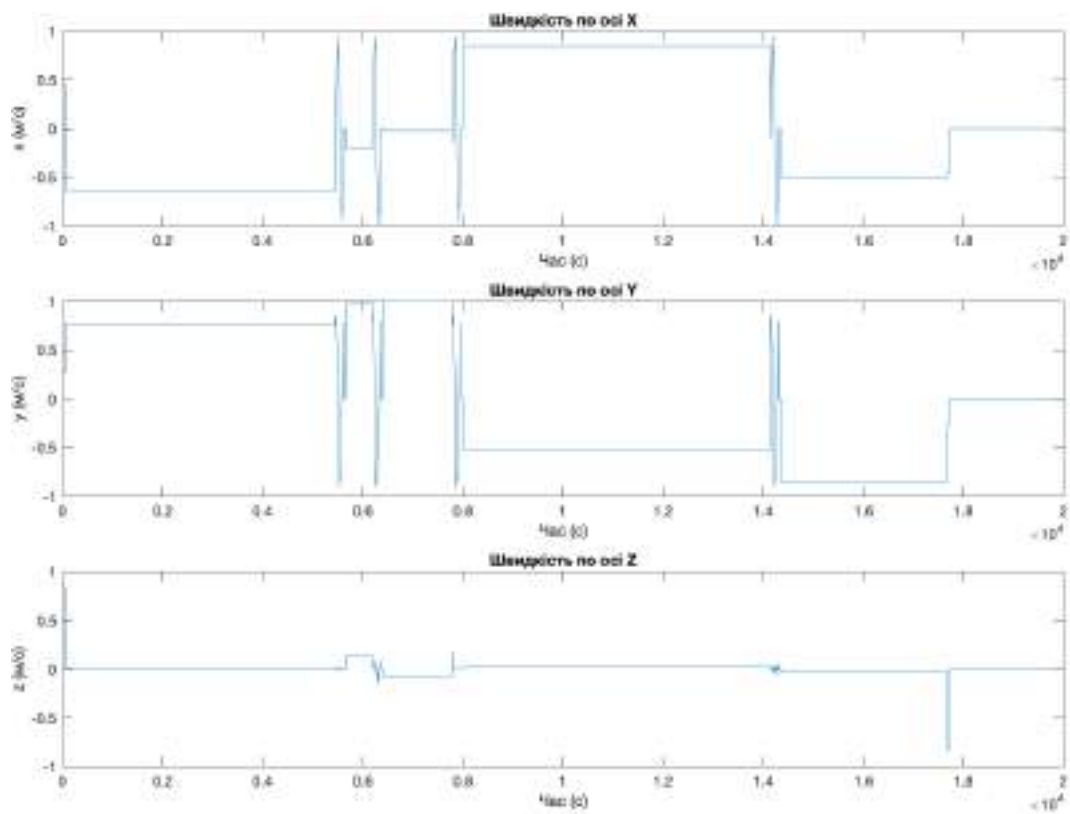


Рисунок 4.12. Фактичні значення зміни швидкості польоту.

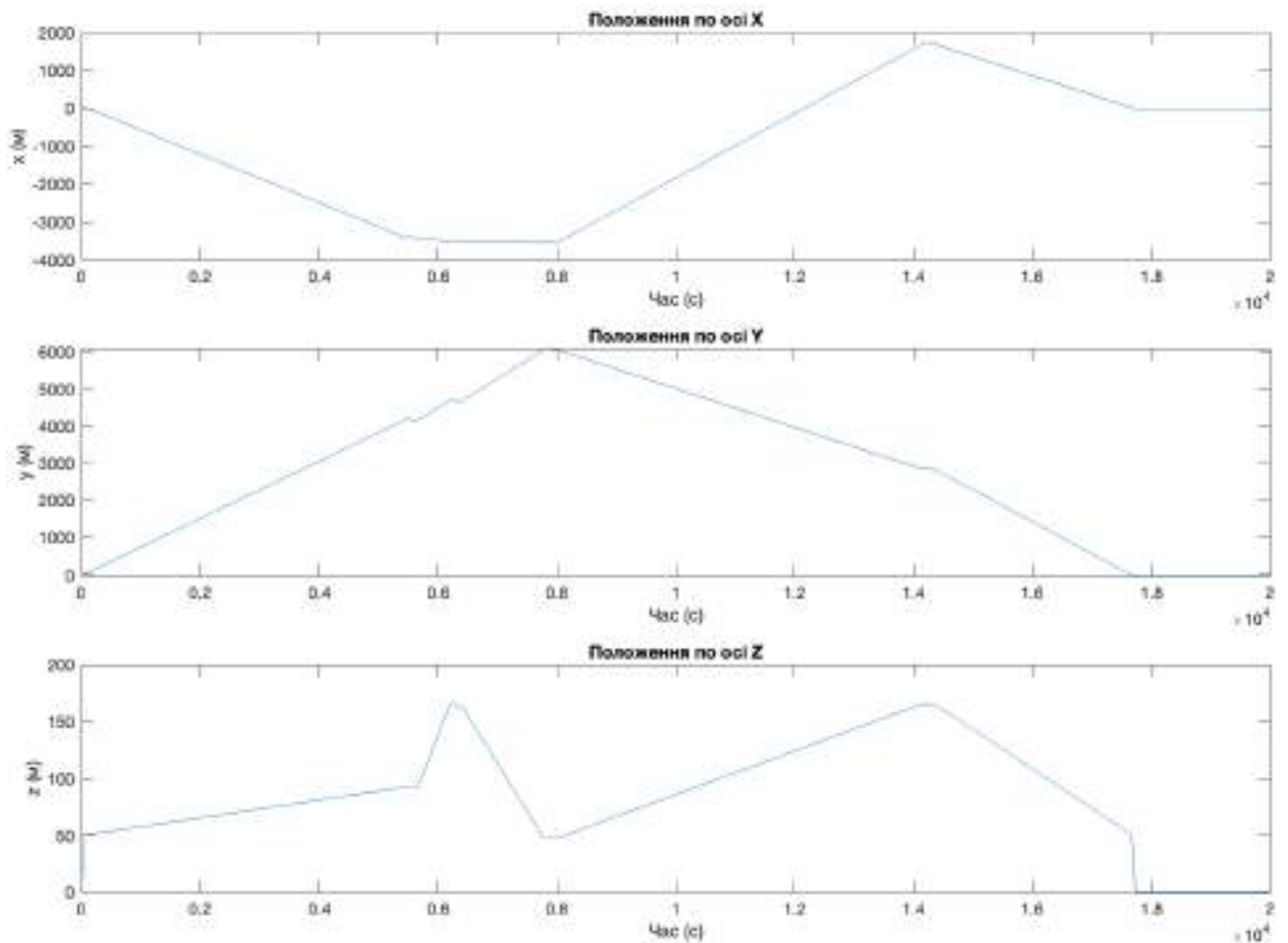


Рисунок 4.13. Фактичні значення зміни положення під час польоту.

Крім цього, в процесі моделювання додатково застосовано алгоритм калібрування датчиків БПЛА перед початком виконанням місії (рис. 4.14). Це дозволяє виконати корекцію систематичних похибок бортової підсистеми МЕМС, включаючи зсуви, масштабні коефіцієнти, неортогональність осей та дрейф, що забезпечує підвищення надійності та точності систем керування та позиціонування.

```

% Розрахунок постійної складової похибок (bias) та коефіцієнт дрейфу (drift rate) для кожної вісі
for i = 1:3
    p = polyfit(t_cal, mems_accelerometer_errors(i, :), 1);
    calib_mems_idx_accelerometer_drift_rate(i) = p(1); % коефіцієнт нахилу для акселерометра
    p = polyfit(t_cal, mems_gyroscope_errors(i, :), 1);
    calib_mems_idx_gyroscope_drift_rate(i) = p(1); % коефіцієнт нахилу для гіроскопа
end
%% Корекція прискорень
calibrated_accelerations = measured_accelerations ...
repmat(calib_mems_idx_accelerometer_bias, 1, size(measured_accelerations, 2)) ...
bsxfun(@times, calib_mems_idx_accelerometer_drift_rate', t);
%% Корекція гіроскопічних вимірювань (кутових швидкостей)
calibrated_gyro_measurements = measured_gyro_rates ...
repmat(calib_mems_idx_gyroscope_bias, 1, size(measured_gyro_rates, 2)) ...
bsxfun(@times, calib_mems_idx_gyroscope_drift_rate', t);
%% Інтегрування отриманих результатів для отримання фактичних значень вимірювання
calibrated_velocity(:,1) = uav_v_0;
calibrated_position(:,1) = uav_coord_0;
% Інтегрування каліброваних прискорень для отримання швидкості та переміщення
for i = 2:length(t)
    calibrated_velocity(:,i) = calibrated_velocity(:,i-1) + calibrated_accelerations(:,i) * dt;
    calibrated_position(:,i) = calibrated_position(:,i-1) + calibrated_velocity(:,i) * dt;
end
% Запис вимірних даних в визначений масив для передачі до обчислювача БПЛА
mems_data = calibrated_position;
mems_v = calibrated_velocity;
mems_a = calibrated_accelerations;
mems_giro = calibrated_gyro_measurements;

```

Рисунок 4.14. Алгоритм калібрування MEMS вимірювачів.

Наведений алгоритм реалізує підхід до підвищення точності отриманої інформації від датчиків MEMS, які мають важливе значення для роботи системи INS в умовах відсутності зовнішніх джерел інформації. Він оцінює базові похибки показань акселерометра і гіроскопа (зсув, дрейф) перед тим, як почати виконання місії та забезпечує подальше отримання уточнених даних для обчислення швидкості і положення БПЛА.

Алгоритм виконує початкове оцінювання усередненого зсуву і швидкості дрейфу для кожної осі акселерометра та гіроскопа. Це досягається за допомогою лінійного регресійного аналізу на основі зібраних даних протягом часу калібрування, де нахил кривої означає швидкість дрейфу. Процес початкового калібрування є критично необхідним для мінімізації проблеми довгострокової стабільності MEMS-датчиків, забезпечуючи подальшу обробку інформації в більш прогнозованій моделі INS системи.

Після калібрування алгоритм застосовує отримані оцінки похибок MEMS-датчиків для коригування подальших вимірювань датчиків системи INS. Для акселерометра він компенсує зміщення і дрейф з часом, отримуючи таким чином інформацію про прискорення. Аналогічна корекція дзеркально відображається для гіроскопічних даних, забезпечуючи уточнення кутових швидкостей і більш точне відображення справжнього руху БпЛА в просторі.

Основою алгоритму калібрування є етап інтеграції, де він перетворює відкалібровані дані прискорення в оцінки швидкості та положення БпЛА. Починаючи з наперед відомих початкових умов, він ітеративно обчислює ці величини, ефективно перераховуючи величину прискорення на інтегровану інформацію про швидкість та переміщення в просторі. Даний етап крок має вирішальне значення для отримання динамічної траєкторії переміщення БпЛА, даючи точне уявлення про його рух у просторі та позиціонування в заданий момент часу. Алгоритм не просто обробляє отримані дані, він дозволяє покращити розуміння БпЛА власної моделі руху, що робить його незамінним інструментом для вирішення завдань позиціонування.

Включення до підсистеми позиціонування БпЛА алгоритму базового калібрування датчиків системи MEMS до початку виконання місії та корекції оцінок систематичних складових похибок INS призвело до загального зниження рівня похибок (рис. 4.16–4.17).

Для підвищення точності даних курсового спрямування (рис. 4.19), що вимірюються бортовим магнітометром (магнітний компас), інформація на його виході додатково додано обробку інформації за допомогою цифрового фільтра

експоненціального ковзного середнього (ЕМА), фрагмент коду моделювання якого зображено на рис. 4.18. Алгоритм роботи даного фільтру був розглянутий раніше в розділі 2.5.

Наведений фрагмент коду ЕМА-фільтру демонструє цифрову реалізацію обробки інформації фільтром спеціально розробленим для згладжування даних магнітометра. Його основна мета – зменшити шум і флуктуації в необроблених показаннях магнітометра, які мають вирішальне значення для точного вимірювання курсу і напрямку в системах позиціонування БпЛА, зокрема, інформація про які може використовуватися для коригування бортових підсистем позиціонування.

Ключовим параметром синтезованого ЕМА-фільтру є коефіцієнт згладжування (α), який визначає ваговий коефіцієнт, що надається останнім вимірюванням порівняно з історичними даними. Менше значення α (в запропонованому фільтрі встановлено значення 0,01) підкреслює важливість історичної інформації про курс, що дозволяє отримати в результаті нормалізовану інформацію, але з потенційним запізненням у реакції на нові зміни інформації про курс. Довжина вимірювань N визначається на основі часового ряду t , що вказує на кількість точок даних, які потрібно обробити.

Ядром алгоритму фільтрації інформації магнітометра є цикл обчислення нормалізованого стану отриманої інформації. Це призводить до згладжування даних про зміну курсу БпЛА, де кожна наступне значення курсу знаходиться під впливом попередніх відфільтрованих даних, ефективно зменшуючи шум і раптові зміни інформації, присутні в необробленій інформації від магнітометра.

Отримана відфільтрована інформація є більш стабільним і надійним джерелом для цілей позиціонування та керування БпЛА, оскільки вона мінімізує вплив перехідних похибок або шуму в показаннях магнітометра. Інтегруючи такий алгоритм фільтрації, система позиціонування отримує більш точну і надійну інформацію про курс, що є критично важливим для точного маневрування і орієнтації в різних умовах застосування.

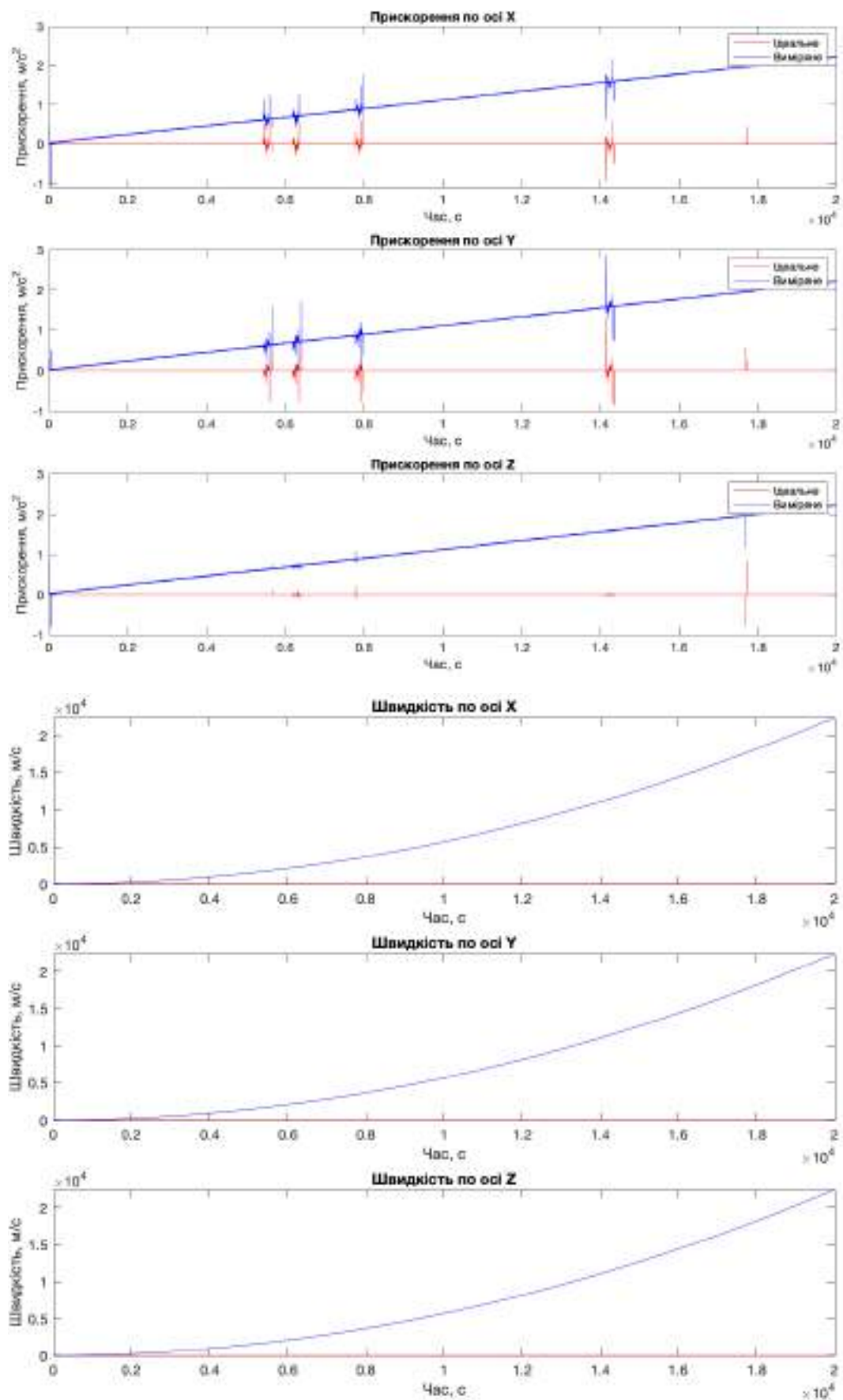


Рисунок 4.15. Інформація про вимірне прискорення та швидкість від датчиків MEMS.

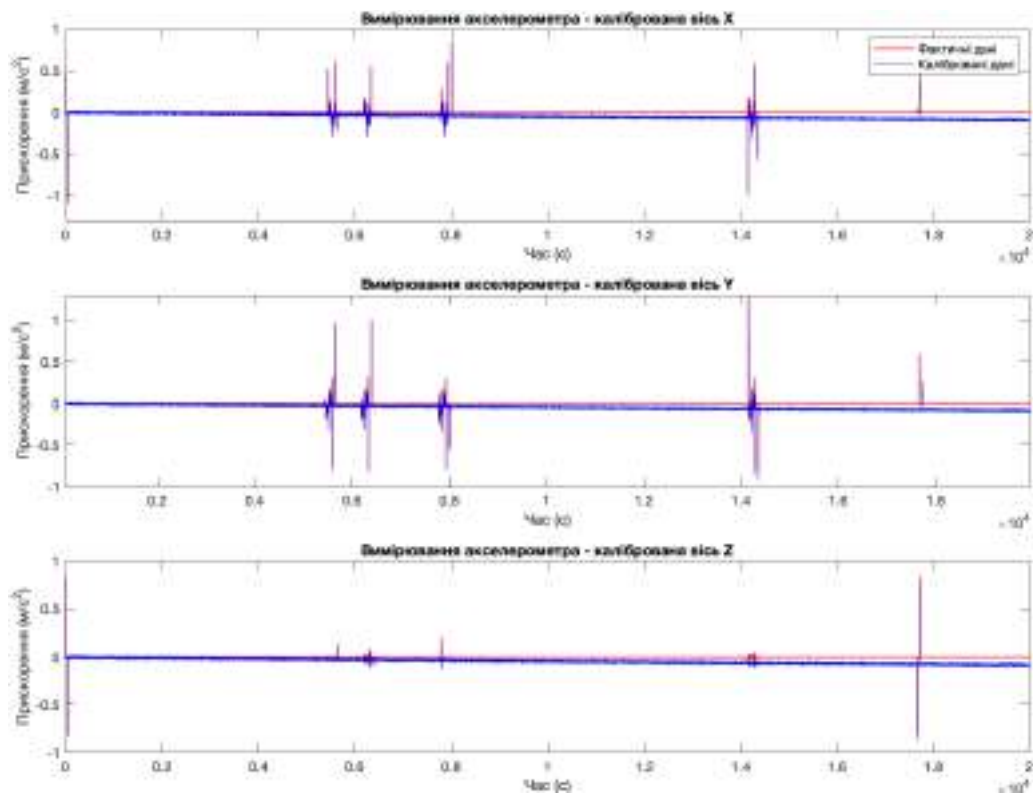


Рисунок 4.16. Інформація про значення прискорення від каліброваних MEMS акселерометрів.

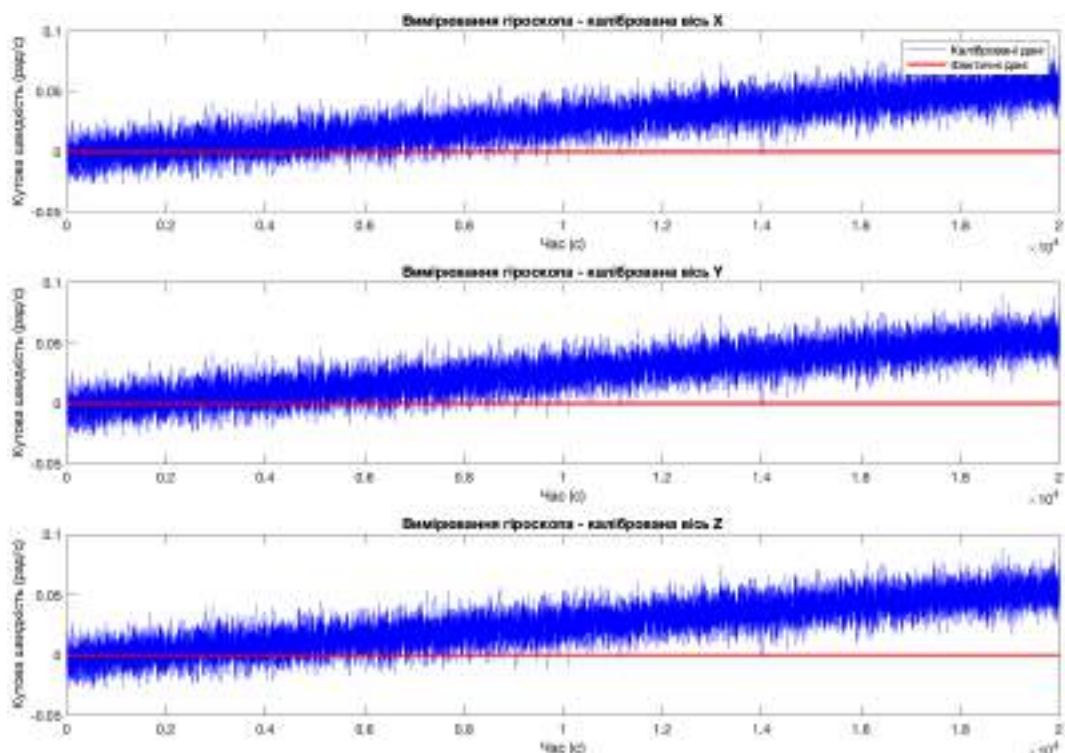


Рисунок 4.17. Інформація про виміряні значення кутових швидкостей каліброваного MEMS гіроскопа.

```

%% АЛГОРИТМ ФІЛЬТРАЦІЇ МАГНІТОМЕТРА
% Параметри фільтра ЕМА
alpha = 0.01; % Коефіцієнт згладжування для ЕМА
N = length(t); % Кількість вимірювань
% Ініціалізація фільтрованих даних
filtered_course = zeros(1, N);
% Використання ЕМА для фільтрації курсу
filtered_course(1) = normalized_course(1); % Початкове значення для ЕМА
for i = 2:N
    filtered_course(i) = alpha * normalized_course(i) + (1 - alpha) * filtered_course(i-1);
end
mag_data = filtered_course; % внесення інформації магнітометра для передачі до обчислювача БПЛА

```

Рисунок 4.18. Алгоритм фільтрації інформації магнітометра.

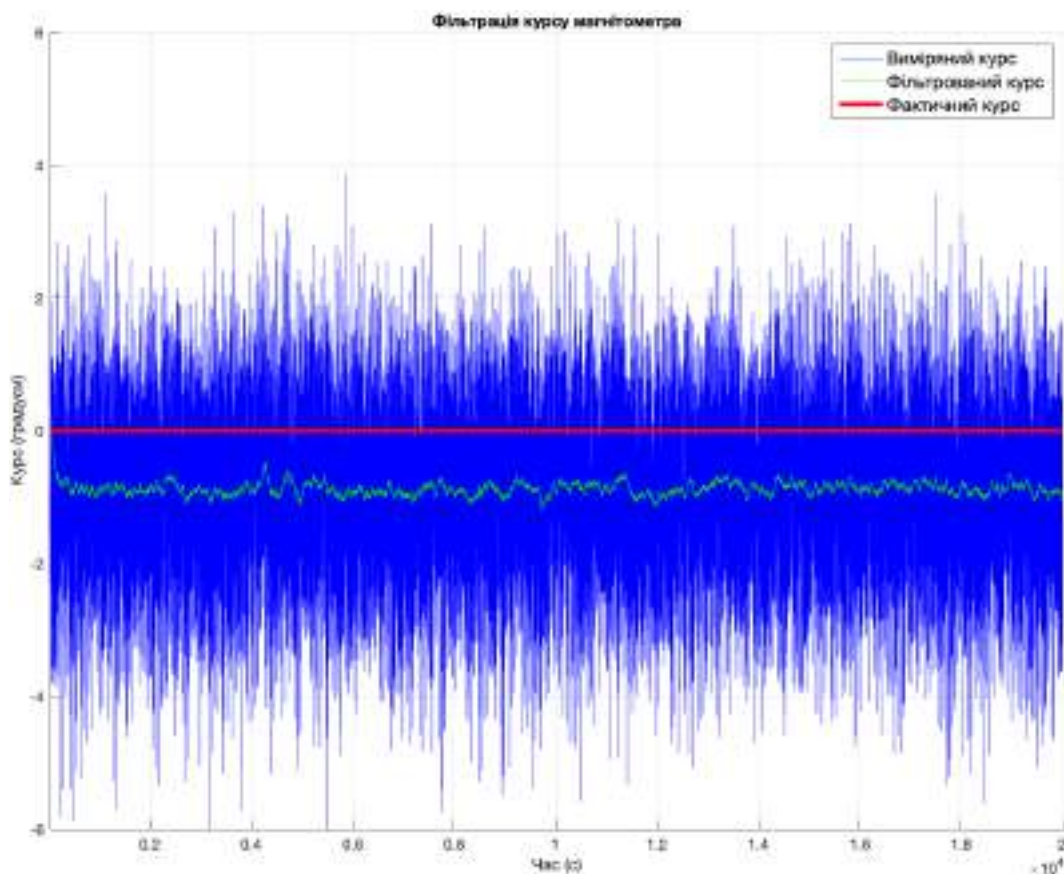


Рисунок 4.19. Інформація щодо вимірних значень курсового спрямування магнітометра (магнітного компаса) під час польоту.

Для оцінки ефективності запропонованої системи, необхідно порівняти її роботу із сучасними системами із оптимальною фільтрацією, що інтегрують бортову INS та GNSS. Для цього було синтезовано модель інтегрованої підсистеми позиціонування INS/GNSS з оптимальною обробкою даних в БПЛА (рис. 4.20).

Модель підсистеми INS/GNSS демонструє реалізацію фільтра Калмана, потужного інструменту, що використовується в системах обробки інформації і керування для оцінки стану динамічної системи за наявності невизначених і зашумлених даних. Основною метою запропонованого коду є моделювання роботи INS/GNSS під час оцінки інформації про положення і швидкості від наявних датчиків позиціонування БПЛА.

Ініціалізація фільтра Калмана задається вектором стану, що містить оцінене положення та згладжену швидкість. Розмірність цього вектора стану – n , визначається з довжини початкового вектора стану. Початкова коваріація похибки оцінок представлена матрицею тотожності P , що вказує на початкове припущення про визначеність оцінок стану. Матриця переходів між станами A призначена для моделювання фізики руху, інтегруючи часовий крок dt для переходу між станами. Матриця керування B відсутня, що вказує на відсутність зовнішнього керування в цій моделі.

Ключовим аспектом фільтра Калмана є його здатність обробляти шум як в процесі, так і при вимірюванні. Це моделюється за допомогою коваріаційної матриці шуму процесу Q і коваріаційної матриці шуму вимірювання R . Матриця шуму процесу Q має малі значення, що вказує на низьку невизначеність у динаміці системи, тоді як R кількісно визначає очікуваний шум у отриманій інформації.

Фільтр працює в циклі, оновлюючи оцінки отриманої інформації з кожним новим набором вимірювань. На кожній ітерації циклу по чергово проходять фази прогнозування та оновлення. На етапі прогнозування вектор стану і коваріація помилки прогножуються на наступний часовий інтервал. Цей прогнозований стан слугує початковою оцінкою перед поданням нової інформації в систему. Потім

обчислюється коефіцієнт посилення Калмана K , який має вирішальне значення для злиття прогнозу з фактичною інформацією для корекції оцінок стану.

На етапі оновлення фільтр коригує свою оцінку стану INS/GNSS (`isns_x_updated`) на основі різниці між фактичними вимірами (`gps_data`) і прогнозованими вимірами. Матриця коваріації помилок P також оновлюється, щоб відобразити зменшену невизначеність після оновлення оцінок інформації. Оновлені оцінки швидкості зберігаються окремо для подальшого використання.

Нарешті, алгоритм отримує оцінене положення (`isns_data`) і швидкість (`isns_v`) з оновленого вектора стану для кожного часового кроку. Це призводить до часового ряду оцінок положення і швидкості БпЛА, які є більш точними і менш зашумленими, ніж необроблена інформація. Така реалізація фільтра Калмана є типовою для систем позиціонування, де точна оцінка положення та швидкості на основі зашумлених даних датчиків є критично важливою.

Аналізуючи та оцінюючи характеристики перехідних процесів параметрів синтезованої підсистеми INS/GNSS, можна відзначити, що впровадження в систему контуру калібрування та позиційної корекції сприяє зниженню сталих похибок системи. Це стало особливо помітно після завершення базового процесу калібрування тривалістю приблизно 5 с значення RMS похибки системи по визначенню координати стало стабільним:

- по-перше, похибки в удосконаленій комплексній системі вже не накопичуються з часом, на відміну від ситуації з не скоригованою INS. Завдяки введеному контуру корекції, похибки системи стабілізувалися на ustalених рівнях, близьких до значення -0 , що дозволяє ефективно компенсувати ці похибки, враховуючи їх величину при синтезі системи;
- по-друге, система набула стійкості, що відображається у характері перехідних процесів. З часом криві значень параметрів стабілізуються, досягаючи ustalених рівнів, які для похибок системи є близькими до нуля.

```

%% Моделювання роботи фільтра Калмана
% Ініціалізація фільтра Калмана
isns_x_kalman = [isns_r_estimated(:, 1); isns_v_smoothed(:, 1)]; % початкова оцінка
n = length(isns_x_kalman); % розмірність стану
P = eye(n); % початкова матриця коваріації похибок оцінки
A = [eye(3), dt*eye(3); zeros(3, 3), eye(3)]; % матриця переходу стану
B = []; % матриця управління (немає зовнішнього входу)
C = [eye(3), zeros(3, 3)]; % матриця спостереження (ми спостерігаємо лише положення)
Q = diag([0.1, 0.1, 0.1, 0.01, 0.01, 0.01]); % коваріація шуму процесу
R = 0.1*eye(3); % коваріація шуму вимірювань
r_estimated(:, 1) = gps_data(:,1); % Ініціалізація МЕМС системи на старті, фіксація початкової координати
% Оптимальна фільтрація
isns_v_kalman = zeros(3, length(t));
for i = 2:length(t)
    % Прогноз
    isns_x_predicted = A * isns_x_kalman(:, i-1);
    P_predicted = A * P * A' + Q;
    % Обчислення коефіцієнта Калмана
    K = P_predicted * C' / (C * P_predicted * C' + R);
    % Оновлення оцінки на основі вимірювань
    isns_x_updated = isns_x_predicted + K * (gps_data(:, i) - C * isns_x_predicted);
    P = (eye(n) - K * C) * P_predicted;
    % Зберігання оновленої оцінки швидкості
    isns_v_kalman(:, i) = isns_x_updated(4:6);
    % Оновлення стану для наступної ітерації
    isns_x_kalman(:, i) = isns_x_updated;
end
isns_data = isns_x_kalman(1:3, :);
isns_v = isns_v_kalman;

```

Рисунок 4.20. Фрагмент коду моделювання роботи синтезованої INS/GNSS.

Отже, можна вважати, що головної мети включення корекції та калібрування було досягнуто. Сформована стабільна система з фіксованими параметрами, при цьому тривалість перехідних процесів лежить в межах 5–8 с, що відповідає заданим вимогам.

Для отримання більш точної системи коливання в кривих після її виходу на стабільний режим можна усунути постійні складові похибок використовуючи додаткову обробку інформації та фільтри.

4.4 Моделювання та дослідження роботи синтезованої схеми

Моделювання та дослідження синтезованої комплексної підсистеми LeGNSS з оптимальною фільтрацією було виконано у дискретній формі в середовищі Matlab (R2023b), використовуючи дискретний фільтр Калмана. Повний текст вихідного коду програми для моделювання підсистеми LeGNSS в Matlab наведено в додатку В.

На рис 4.21 показано фрагмент коду Matlab для моделювання роботи синтезованої підсистеми LeGNSS, що використовує методи оптимальної фільтрації. В основі системи лежить реалізація фільтра Калмана, популярного алгоритму в системах керування та позиціонування для оцінювання невідомих змінних на основі серії вимірювань, що спостерігаються протягом певного часу. Підсистема LeGNSS за допомогою цього коду демонструє свою здатність точно оцінювати як положення, так і швидкість БпЛА.

На початку модель ініціалізує параметри фільтра Калмана, де він встановлює стан системи з шістьма змінними – тривимірними координатами (x , y , z) і компонентами швидкості (V_x , V_y , V_z) в кожному з вимірів. Початковий стан системи дорівнює нулю, що вказує на початок відстеження, тоді як початкова коваріація визначається як матриця тотожності, що відображає початкову визначеність оцінок стану системи. Матриця спостережень розширюється, щоб врахувати три набори тривимірних вимірювань, які надходять від різних систем позиціонування, таких як GPS, INS/GNSS та низькоорбітальні LEO супутники (StarLink).

Можливості алгоритму до адаптації в змінних умовах розкриваються в ітеративному процесі, де кожен цикл складається з етапу прогнозування, за яким слідує етап оновлення системи. На етапі прогнозування наступний стан системи прогнозується на основі поточного стану і матриць переходів, інтегруючи коваріацію шуму процесу, щоб врахувати невизначеності, притаманні динаміці польоту БпЛА.

Згодом починається етап оновлення, на якому враховуються фактичні спостереження з трьох різних джерел даних (GPS, LEO і INS/GNSS). Коефіцієнти корекції підсилення Калмана, ключовий компонент цього етапу, він збалансовує оцінений стан з новими спостереженнями, уточнюючи оцінку стану отриманої інформації за останніми даними. Таке поєднання прогнозування і корекції триває протягом кожної ітерації, поступово підвищуючи точність оцінок положення і швидкості.

Перевага запропонованого коду полягає не лише в його математичній простоті, але й в тому, як він ефективно виконує злиття даних з різних джерел, щоб отримати більш надійну і точну оцінку положення і швидкості БпЛА. Результат роботи підсистеми LeGNSS є інформація про оцінку позиціонування та швидкості в заданий момент часу. Проведена симуляція алгоритму моделювання дає уявлення про потенціал запропонованого методу обробки інформації в LeGNSS підсистемі позиціонування.

Дослідження передбачає ідеалізовані умови для прийому супутникового сигналу, які не завжди відповідають реальним сценаріям. Поширення похибок системи INS було змодельовано лінійно, що могло не вловлювати всі нюанси реальних дрейфів INS. Через воєнний стан в Україні збір даних був ускладнений, у тому числі доступ до супутникових сигналів через можливі обмеження (StarLink, OneWeb тощо).

Порівняльне відображення результатів моделювання роботи різних підсистем позиціонування БпЛА показано на рис. 4.22, де показано результати обробки інформації про позиціонування в трьох підсистемах: оцінка позиціонування отримана після інтеграції окремих систем в GPS (1), оцінка отриманих даних позиціонування в INS/GNSS (2), оцінка отриманих даних позиціонування в LeGNSS (3) та фактичне значення положення БпЛА (4).

Аналіз результатів роботи синтезованої комплексної LeGNSS щодо обробки інформації про динаміку польоту представлено на рис. 4.23, де показано порівняння отриманих оцінок швидкості для підсистем INS/GNSS та LeGNSS.

```

%% Код для моделювання LeGNSS системи позиціонування з оптимальною фільтрацією
% Ініціалізація фільтра Калмана
legnss_numStates = 6; % Кількість станів системи: x, y, z, vx, vy, vz
legnss_initialState = zeros(legnss_numStates, 1); % Початковий стан
legnss_initialCovariance = eye(legnss_numStates); % Початкова коваріація
legnss_numObservations = 3 * 3; % Три виміри позиції від трьох систем
legnss_observationMatrix = eye(legnss_numObservations, legnss_numStates); % Розширена матриця
спостереження
legnss_stateTransitionMatrix = eye(legnss_numStates); % Матриця переходу станів
% Шуми спостереження для кожної системи
legnss_observationNoiseCovariance = blkdiag(eye(3) * 1, eye(3) * 1, eye(3) * 1); % Блочно-діагональна
матриця
legnss_processNoiseCovariance = eye(legnss_numStates) * 0.1; % Коваріація шуму процесу
% Ініціалізація змінних для збереження результатів
LeGNSS_data = zeros(3, length(gps_data)); % Зберігання оцінки позиції
LeGNSS_velocity = zeros(3, length(gps_data)); % Зберігання оцінки швидкості
% Цикл прогнозування та оновлення
for k = 1:length(gps_data)
    % Крок прогнозування
    legnss_predictedState = legnss_stateTransitionMatrix * legnss_initialState;
    legnss_predictedCovariance = legnss_stateTransitionMatrix * legnss_initialCovariance *
legnss_stateTransitionMatrix' + legnss_processNoiseCovariance;
    % Отримання спостереження
    legnss_observation = [gps_data(:, k); leo_data(:, k); isns_data(:, k)]; % Об'єднання спостережень
    % Крок оновлення
    legnss_KalmanGain = legnss_predictedCovariance * legnss_observationMatrix' / (legnss_observationMatrix
* legnss_predictedCovariance * legnss_observationMatrix' + legnss_observationNoiseCovariance);
    legnss_initialState = legnss_predictedState + legnss_KalmanGain * (legnss_observation -
legnss_observationMatrix * legnss_predictedState);
    legnss_initialCovariance = (eye(legnss_numStates) - legnss_KalmanGain * legnss_observationMatrix) *
legnss_predictedCovariance;
    % Збереження результатів
    LeGNSS_data(:, k) = legnss_initialState(1:3); % Позиція: x, y, z
    LeGNSS_velocity(:, k) = legnss_initialState(4:6); % Швидкість: vx, vy, vz
end

```

Рисунок 4.21. Фрагмент коду для моделювання роботи синтезованої комплексної підсистеми LeGNSS з оптимальною фільтрацією.

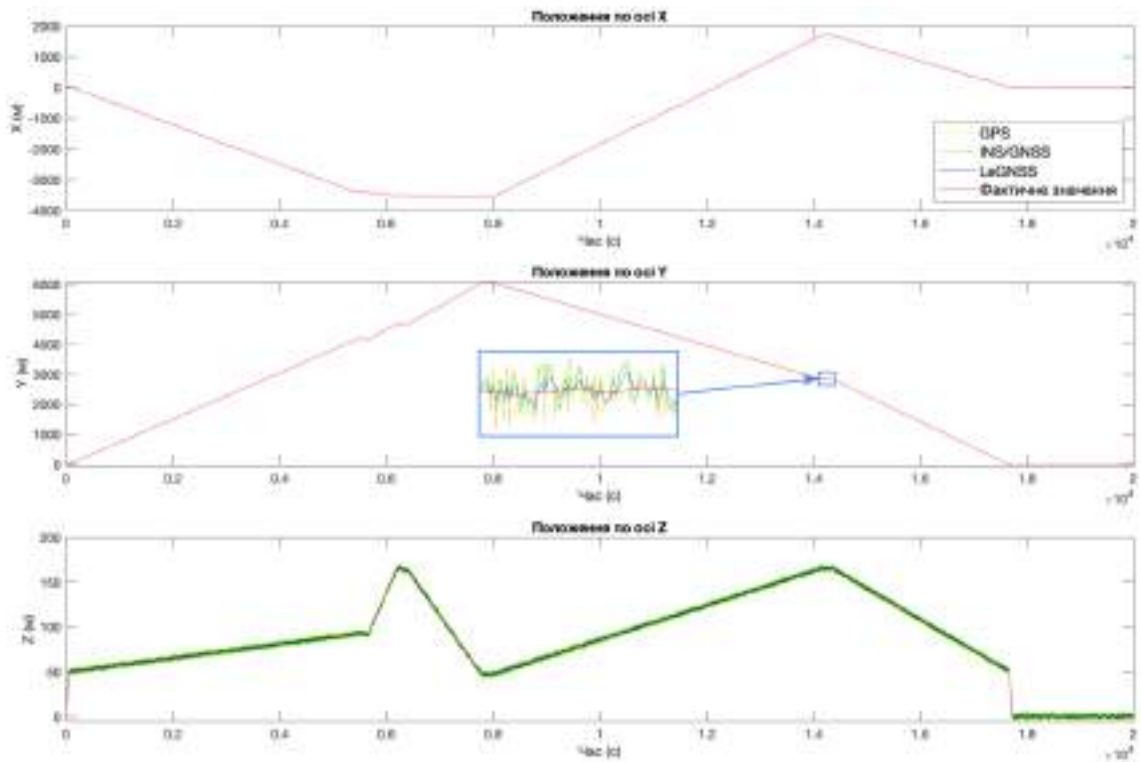


Рисунок 4.22. Оцінка даних про позиціонування БПЛА протягом польоту.

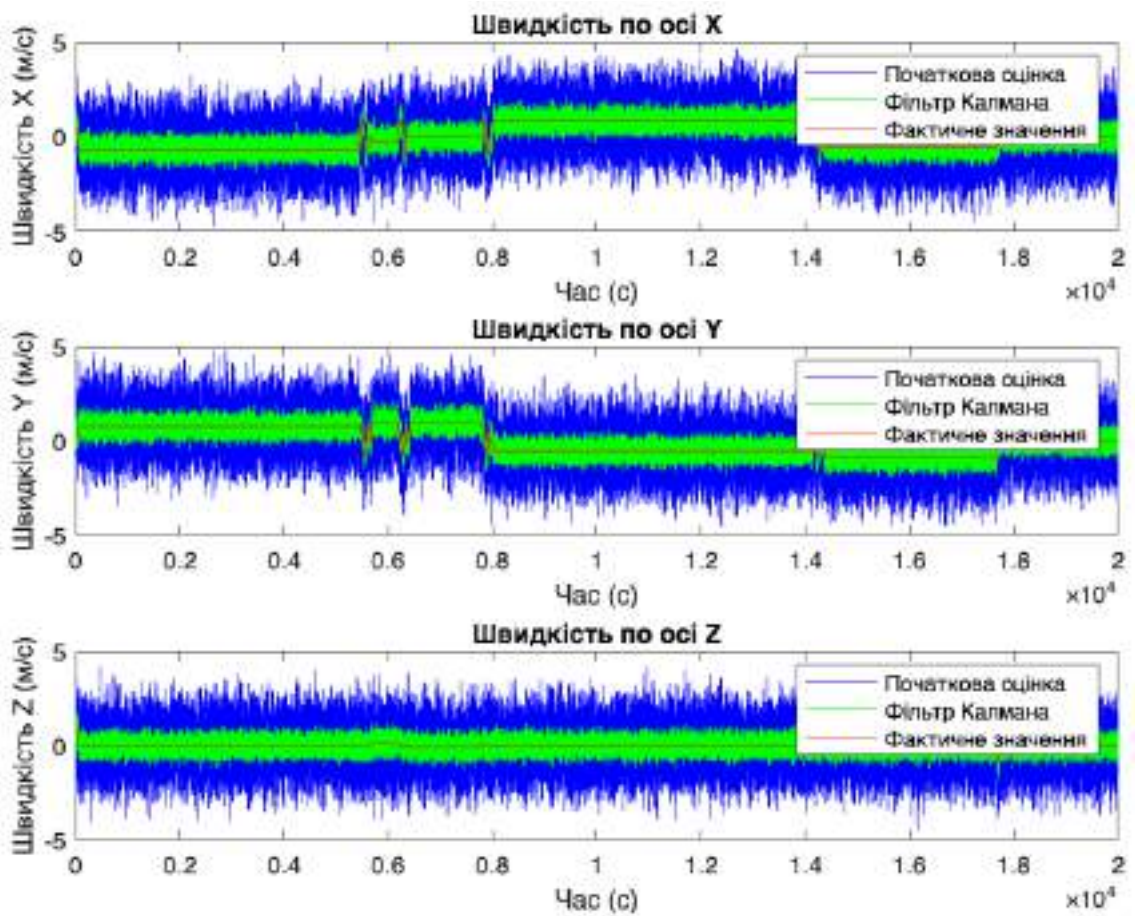


Рисунок 4.23. Збір даних про динаміку польоту під час моделювання.

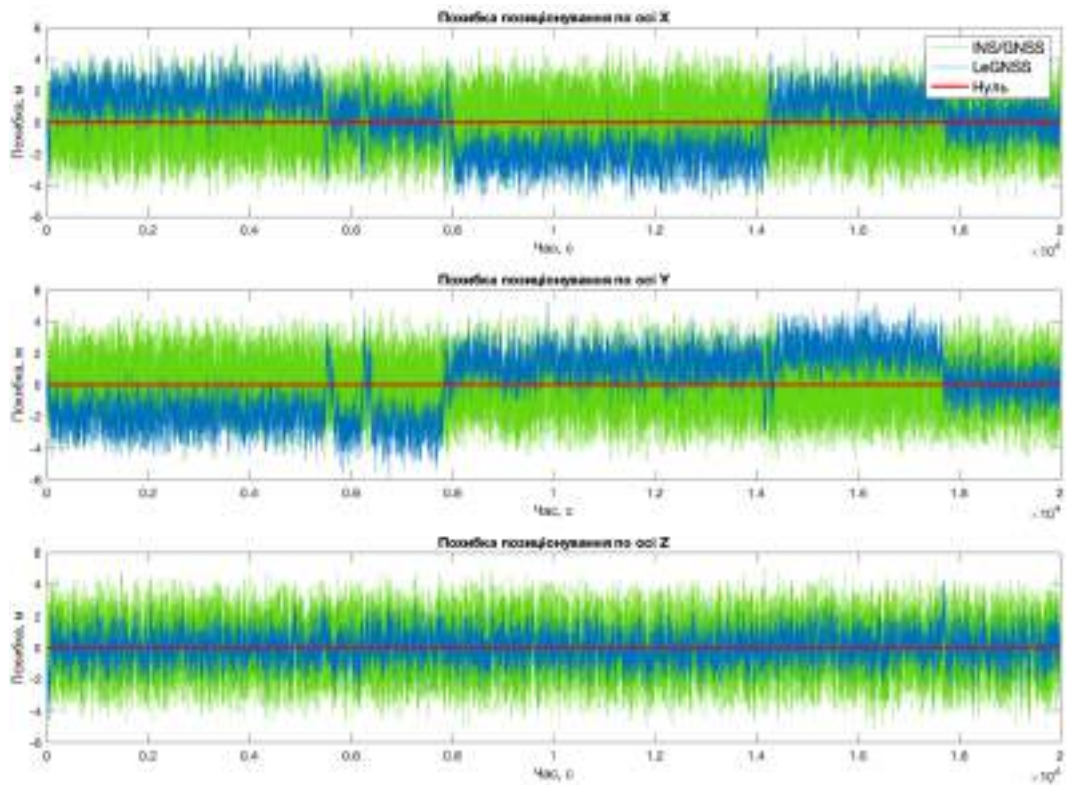


Рисунок 4.24. Порівняння похибок підсистем по позиціонуванню.

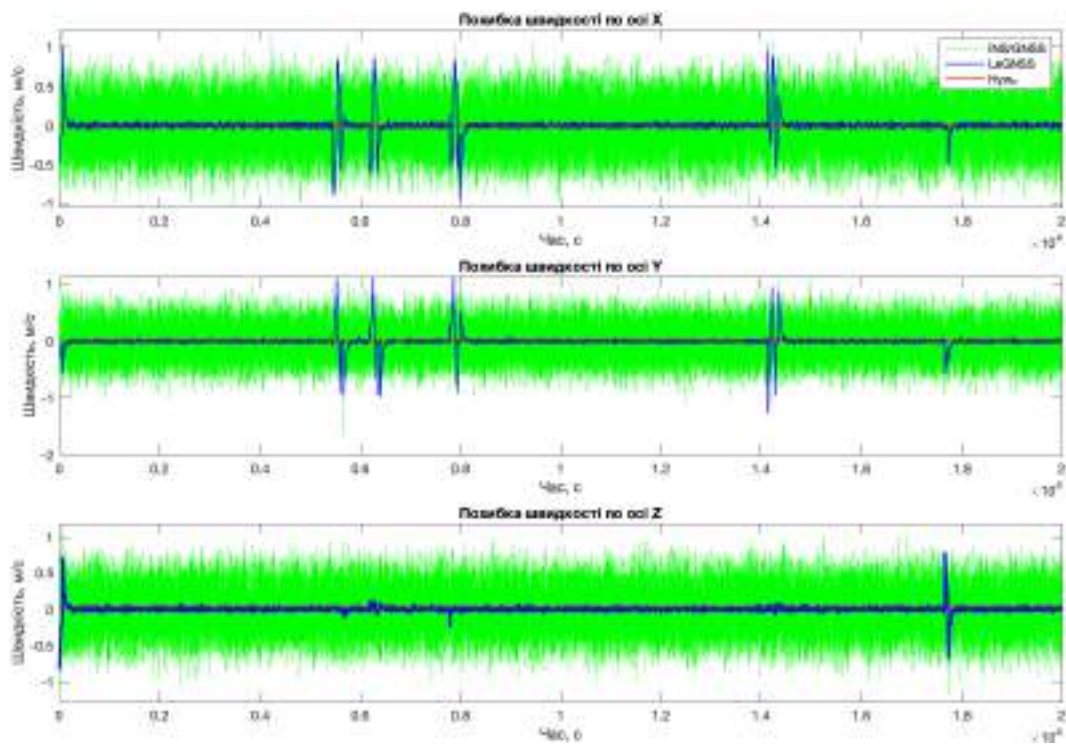


Рисунок 4.25. Порівняння похибок підсистем по швидкості.

На рис. 4.24–4.25 показано порівняльний аналіз похибок оцінювання підсистемами значень позиціонування та швидкості БПЛА для основних конкуруючих бортових підсистем позиціонування INS/GNSS та LeGNSS.

Наведені графіки показують, що включення в модель комплексної системи алгоритму оптимальної фільтрації та позиційної корекції дозволило досягти зменшення загальної похибки системи та зниженню рівня зашумленості отриманих оцінок про позиціонування для обох конкуруючих підсистем. Але додаткове введення інформації від LEO супутників дозволило суттєво підвищити точність та стабільність обробки даних в підсистемі і, як наслідок, кращі результати обробки інформації в динамічних умовах роботи.

На початку періоду моделювання похибка має незначний сплеск, проте не виходить за допустимі межі та залишається відносно низькою протягом всього періоду моделювання, що свідчить про надійну ініціалізацію системи. Протягом усього періоду можна спостерігати перехідні піки похибки. Ці піки можна пояснити значною зміною динаміки польоту під час проходження складних етапів маршруту, збоями сигналу або тимчасовими дрейфами INS. Незважаючи на випадкові піки, загальна тенденція кривої похибок є стабільною, що свідчить про стабільну роботу запропонованої системи LeGNSS/INS. Представлена на рис. 4.24 детальна оцінка похибки моделювання в запропонованій системі LeGNSS/INS демонструє усереднені похибки, які знаходяться в межах 3–3,5 метрів і слабо залежать від тимчасової втрати сигналу GNSS. Це демонструє високу ефективність обробки інформації про позиціонування підсистемою в реальному часі.

Для проведення фактичної оцінки рівня підвищення якості обробки даних про позиціонування в БПЛА необхідно розрахувати значення RMS кожної підсистеми. Так як більшість супутникових систем позиціонування мають хорошу точність тільки в двовимірному просторі, а БПЛА виконують місії в трьохвимірному просторі, необхідно зробити розрахунки трьохвимірної RMS.

Завданням аналізу RMS системи стає не лише кількісне визначення значень похибок, а й оптимізація вагових коефіцієнтів, що дозволяє мінімізувати

значення похибки та збільшити точність позиціонування за рахунок більш ефективного використання даних з різних супутникових систем. Оптимізована сукупна похибка 3D RMS, виведена на основі вагових коефіцієнтів, стає ключовим показником для оцінки якості запропонованої моделі системи позиціонування. Вона відображає інтегрований вплив похибок кожної із вісей координат, дозволяючи сформувати комплексну картину точності оцінки отриманої інформації про позиціонування в просторі.

Рівняння (4.1) визначає 3D RMS похибку, що дає нам оцінку розсіювання вимірювань відносно істинного значення в трьох вимірах:

$$RMS_{3D} = \sqrt{RMS_x^2 + RMS_y^2 + RMS_z^2}. \quad (4.1)$$

Рівняння (4.2) визначає абсолютне значення RMS системи, що дає нам загальну оцінку значення похибки протягом часу моделювання:

$$RMS_{total} = \sqrt{\frac{1}{n} \sum_{t=1}^n (RMS(t)_{3D}^2)}. \quad (4.2)$$

Для кількісної оцінки точності результату моделювання проведено багаторазове статистичне оцінювання з метою отримання дисперсії оцінки похибки комплексної LeGNSS підсистеми та порівняння отриманих даних із іншими підсистемами. На рис. 4.26 наведено графік зміни середньоквадратичного значення оцінки похибки позиціонування, отримані за період випробування, в підсистемах LeGNSS, GPS, ING/GNSS та LEO. При чому, підсистеми LeGNSS та ING/GNSS використовують оптимальну фільтрацію інформації (фільтру Калмана). А підсистема LeGNSS виконує обробку інформації отриманої шляхом оптимальної фільтрації даних з трьох підсистем БПЛА.

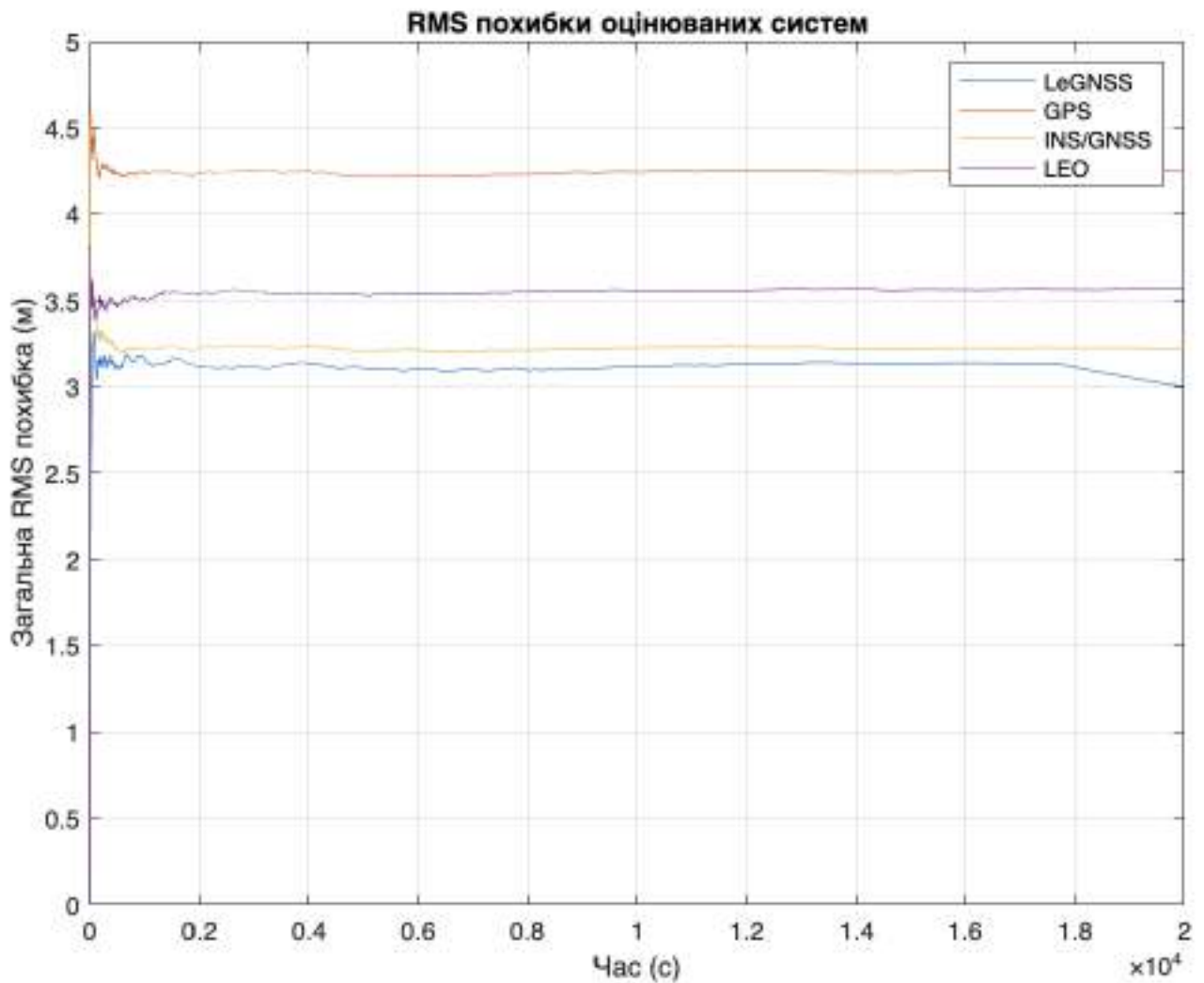


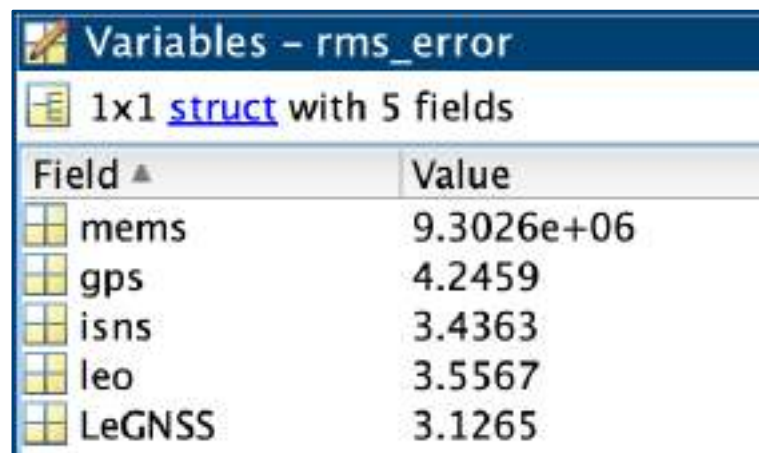
Рисунок 4.26. Аналіз зміни середньоквадратичних значень оцінки похибок для кожної із досліджуваних систем за результатами моделювання.

Проаналізувавши отримані графіки роботи підсистеми, можна відзначити, що підсистема LeGNSS демонструє високу ефективність, особливо коли мова йде про високі динамічні навантаження та швидкі зміни напрямку, що вимагають безперервної корекції траєкторії БПЛА. Це підтверджується більш низькими значеннями RMS для підсистеми LeGNSS порівняно з традиційними системами GPS та INS/GNSS, що свідчить про високу точність та стабільність під час бортової обробки інформації.

Хоча обидві інтегровані підсистеми (INS/GNSS та LeGNSS) показують високу точність та робастність, проте запропонована LeGNSS перевершує існуючі конфігурації, особливо в складних динамічних умовах виконання

автономних місій БПЛА. Таке покращення узгоджується з останніми публікаціями, що підкреслюють переваги підходу використання архітектури багато-супутникових систем позиціонування із сучасними низькоорбітальними LEO супутниками.

Отримані розрахунки статистичних середньоквадратичних значень оцінки похибок позиціонування для кожної із досліджуваних підсистем за результатами моделювання демонструють покращену точність LeGNSS підсистеми позиціонування БПЛА у порівнянні із інтегрованою INS/GNSS (ISNS) підвищеної точності на – 9,02%, та на 26,4% у порівнянні з бортовим GPS приймачем (рис.4.27).



The screenshot shows a window titled "Variables - rms_error" containing a table with 5 columns: "Field" and "Value". The table lists the following data:

Field	Value
mems	9.3026e+06
gps	4.2459
isns	3.4363
leo	3.5567
LeGNSS	3.1265

Рисунок 4.27. Середньоквадратичні значення оцінок похибки для кожної із досліджуваних систем за результатами статистичного моделювання.

Отримані результати роботи демонструють високу ефективність роботи комплексної LeGNSS підсистеми позиціонування БПЛА, утвореної поєднанням комунікаційної системи супутників на низькій орбіті LEO (StarLink), глобальної супутникової системи позиціонування GNSS (GPS) та бортової MEMS системи вимірювання. Внутрішня обробка інформації за допомогою алгоритмів оптимальної обробки даних та фільтрації в підсистемі LeGNSS із різних типів джерел дозволяє суттєво підвищити точність оцінки позиціонування та швидкості БПЛА. Це дозволяє реалізувати ефективну обробку інформації у

реальному часу, використовуючи невеликий об'єм обчислювальної потужності та оперативної пам'яті.

Однак, не можна обмежуватися тільки статистичним аналізом точності порівнюваних засобів. Практичне впровадження та експлуатація системи LeGNSS вимагає подальшого моніторингу та налагодження. Це передбачає оновлення алгоритмів бортової обробки інформації, які можуть адаптуватися до змінних умов середовища та вдосконалюватися відповідно до накопиченого досвіду використання БПЛА.

Майбутній розвиток підсистем позиціонування у сфері автономного керування повинні фокусуватися не тільки на збільшенні точності позиціонування та зменшенні похибок визначення координат, але й на підвищенні гнучкості та адаптивності системи до нових викликів. Це включає в себе розробку адаптивних алгоритмів, що можуть самостійно визначати ефективні маршрути переміщення, формувати оптимальні команди керування БПЛА, застосовувати комплексні стратегії фільтрації інформації на основі машинного навчання та штучного інтелекту. Системи самонавчання, здатні до самостійного визначення умов експлуатації дрона та реагування на непередбачені зміни таких умов, мають стати наступним кроком у розвитку підсистем позиціонування БПЛА.

Подальші дослідження можуть проаналізувати вплив більш складних факторів навколишнього середовища на продуктивність об'єднаної системи та додатково вивчити інтеграцію додаткових супутникових систем або датчиків для підвищення точності інформації про позиціонування. Подальше налаштування оптимальних параметрів фільтрації дозволяє досягти чудової плавності та точності отриманої позиційної інформації.

4.5 Висновки до четвертого розділу

Інтеграція до моделі новітньої комунікаційної системи LEO (StarLink) та Калманівського фільтра дозволяє суттєво знизити вплив шуму вимірювань і покращити точність оцінок позиціонування у порівнянні із раніше існуючими системами, що використовують бюджетні приймачі сигналів. Запропонована модель пропонує надійне, безперервне і точне рішення для позиціонування, використовуючи сильні сторони кожного компонента і забезпечуючи резервування.

Моделювання режимів роботи без корекції та із корекцією показало, що синтезована система являється високоефективною та більш точною, у порівнянні із кожною окремо взятою підсистемою. Графіки похибок системи отримані імітаційним моделюванням відповідають розрахунковим. Середній час виходу системи на робочі режими становить 5 с.

За результатами моделювання синтезована LeGNSS підсистеми позиціонування БПЛА демонструє покращену точність у порівнянні із інтегрованою INS/GNSS (ISNS) високої точності на $-9,02\%$, та на $26,4\%$ у порівнянні з бортовим GPS приймачем.

Висновки цього дослідження можуть бути застосовані в різних секторах, включаючи безпілотні авіаційні системи, цивільну авіацію, системи керування автономними дронами та сектор оборони. Покращена точність позиціонування може підвищити безпеку, точність та операційну ефективність розроблених систем.

У подальших дослідженнях можна проаналізувати вплив більш складних факторів навколишнього середовища на роботу LeGNSS, вивчити можливість розширення переліку супутникових систем або датчиків для підвищення точності інформації про позиціонування.

ВИСНОВКИ

У дисертаційній роботі вирішено актуальне науково-технічне завдання, яке дає можливість розв'язати важливу наукову проблему, що полягає в підвищенні ефективності оцінки інформації про позиціонування в бортових підсистемах БпЛА за рахунок використання надлишкових джерел інформації від декількох супутникових угруповань у поєднанні з бортовими засобами позиціонування та оптимальною фільтрацією інформації.

Наукові та практичні результати отримані в дослідженні дають підстави сформулювати відповідні висновки та рекомендації:

1. Проведено систематизацію основних дослідницьких підходів та виконано всесторонній аналіз сучасних принципів та особливостей функціонування засобів підвищення ефективності оцінки інформації про позиціонування, який дав підстави аргументувати доцільність і можливість створення методу оптимальної обробки інформації в бортових підсистемах БпЛА із використанням даних супутників LEO, GNSS та бортових інерціальних засобів позиціонування.

2. Проведений огляд сучасних засобів позиціонування, який дозволив сформулювати головні завдання проведення дослідження і слабкі сторони існуючих бортових рішень для визначення позиціонування БпЛА. Виявлено і систематизовано ключові недоліки існуючих бортових рішень, що лягло в основу розробки нових методів оптимальної фільтрації інформації. Охарактеризовано сучасні методи обробки інформації в бортових підсистемах БпЛА про позиціонування, котрі дозволяють виконувати надання точної інформації про позиціонування в реальному часі.

Додатково, для врахування реальних умов експлуатації БпЛА та зовнішніх факторів, що впливають на якість отриманої інформації про точність позиціонування, було виконано аналіз впливу атмосферних умов, інтерференції отриманого сигналу та іоносферних затримок. Це дозволило розробити методи компенсації цих ефектів за допомогою алгоритмів оптимальної фільтрації, значно підвищивши робастність системи в складних погодних умовах.

3. Обґрунтовано вибір методів обробки інформації для підвищення точності обробки інформації про позиціонування, що включають оптимальну фільтрацію Калмана та засоби контролю цілісності системи, які дозволяють забезпечити робастність і точність визначення положення БпЛА в просторі.

Запропонований метод із новітньою комунікаційною системою LEO-супутників (StarLink) та Калманівського фільтра дозволяє суттєво знизити вплив шуму вимірювань і покращити точність оцінок позиціонування у порівнянні із раніше існуючими системами, що використовують бюджетні приймачі сигналів. Запропонована модель пропонує надійне, безперервне і точне рішення позиціонування.

4. Розроблено вдосконалений метод оптимальної фільтрації в системах позиціонування, який базується на використанні надлишковості джерел інформації про позиціонування, через злиття даних GNSS та LEO супутникових сузір'їв. Розробка цього методу базується на новаторському підході до використання надлишкових джерел інформації, що є значним кроком уперед у технології позиціонування, тим самим підтверджуючи наукову новизну. На основі даного методу запропонована модель інтегрованої робастної LeGNSS підсистеми позиціонування, що здатна в реальному часі обробляти інформацію.

5. Удосконалено методику оцінювання похибок позиціонування підсистеми прийому та обробки даних, яка базується на злитті даних, що виходить за межі існуючих підходів та демонструє зниження впливу випадкових та детермінованих похибок датчиків супутникових даних та інерціальних приймачів інформації на результати оцінки положення БпЛА в просторі.

6. Запропоновано модель підсистеми для адаптивної обробки інформації в багато-супутниковій системі позиціонування з оптимальною фільтрацією для зниження впливу невизначеностей похибок системи в динамічних режимах роботи, що дозволило досягти рівня точності $\sim 3,12$ м. Адаптивна обробка інформації в багато-супутниковій системі позиціонування дозволила застосувати механізми контролю цілісності отримуваної інформації та алгоритми компенсації впливу скомпрометованих джерел інформації.

7. Удосконалено методику оцінки та компенсації похибок позиціонування в багато-супутникових системах, за рахунок використання сучасних алгоритмів обробки інформації та зниження впливу похибок на отримувані дані щодо позиціонування, що дозволило скоротити час виходу підсистеми позиціонування на робочий режим до 5–8 с.

8. Проведено експериментальні дослідження та виконано аналіз ефективності роботи синтезованої LeGNSS підсистеми, які підтверджують підвищення ефективності обробки даних в БпЛА та демонструють на 9,02% вищу точність отримуваної інформації про позиціонування, у порівнянні із бортовою інтегрованою INS/GNSS системою, та на 26,4% у порівнянні з бортовою GPS системою.

Кращі результати в динамічних умовах роботи демонструють інтегровані INS/GNSS підсистеми, разом з тим запропонована LeGNSS підсистема перевершує існуючі конфігурації, особливо в складних динамічних умовах виконання автономних місій БпЛА. Таке покращення узгоджується з останніми публікаціями, що підкреслюють переваги підходу використання архітектури багато-супутникових систем позиціонування із сучасними LEO супутниками.

Висновки цього дослідження можуть бути застосовані в різних секторах, включаючи безпілотні авіаційні системи, що дозволяє підвищити безпеку, точність та операційну ефективність існуючих систем.

Дослідження припускає ідеалізовані умови прийому супутникового сигналу, які не завжди можуть відповідати реальним сценаріям використання. Моделювання похибок системи MEMS може не враховувати всі деталі дрейфів та зсуву вимірювання конкретних датчиків. Через воєнний стан в Україні збір даних був ускладнений, включаючи доступ до супутникових сигналів через потенційні обмеження (StarLink, тощо). У подальших дослідженнях можна аналізувати вплив складних факторів середовища на роботу LeGNSS, вивчити можливість розширення переліку супутникових систем або датчиків для підвищення точності інформації про позиціонування.

У рамках дослідження в кооперації з партнерами Громадська спілка «Бізнес інкубатор груп. Україна» було також здійснено оновлення методичних рекомендацій для оцінки точності обробки інформації позиціонування для використання комплексних систем позиціонування в цивільній авіації та в рамках реалізації проекту розробки автономної безпілотної системи, що включають критерії вибору оптимальних систем позиціонування з урахуванням специфіки виконуваних місій та метеорологічних умов. Проведено оновлення методологічних рекомендації і технічного регламенту оцінки систем позиціонування для застосування в Всеукраїнському центрі реформ транспортної інфраструктури, що підтверджено актами впровадження.

Результати дисертаційного дослідження впроваджено у діяльності Громадська спілка «Бізнес інкубатор груп. Україна» (акт про впровадження від 06.10.2023) та Громадська спілка «Українська авіатранспортна асоціація» (акт про впровадження від 12.10.2023), Всеукраїнський центр реформ транспортної інфраструктури (акт про впровадження від 17.10.2023) та Українській асоціація високотехнологічних підприємств та організацій «КОСМОС» (акт про впровадження від 24.10.23). Отримано Свідоцтво про реєстрацію авторського права на комп'ютерну програму «Система адаптивної обробки інформації в багато-супутниковій системі позиціонування LeGNSS» (додаток Б; шифр «LeGNSS», державний реєстраційний номер №122052 від 18.12.2023) та подано заявку про державну реєстрацію винаходу (додаток В; державний реєстраційний номер №u202305401 від 13.11.2023).

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Васильєв В.М., Рогожин В.О., Долінце Б.І. Схема комплексування інерціальної і супутникової навігаційних систем та її дослідження. *Проблеми створення, випробування, застосування та експлуатації складних інформаційних систем: збірник наукових праць*. Житомир: ЖВІ НАУ, 2013. Вип. 8. С. 20-25.
2. Васильєв В.М., Рогожин В.О., Долінце Б.І. Дослідження похибок інерціально-супутникової навігаційної системи. *Наука і техніка Повітряних сил Збройних сил України*. Харків: ХУПС, 2013. Вип. 3(12). С. 139–143.
3. Vasyliiev V.M., Rogozhyn V.O., Dolintse B.I. Integration of inertial and satellite navigation systems using corrective circuits for UAV. *IEEE 3rd International Conference on Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD 2015)*, October 2015, Kyiv, Ukraine. IEEE Xplore. 2015. P. 193–197. DOI: 10.1109/APUAVD.2015.7346597. (date of access: 27.12.2023)
4. Vasyliiev V., Rogozhyn V., Dolintse B. Accuracy analysis of inertial-satellite navigation system Integrated with using corrective circuits. *Electronics and Control Systems*. 2015. Вип. 4(46). P. 46–51. DOI: 10.18372/1990-5548.46.9930.
5. Vasyliiev V., Dolintse B. Integration of Inertial and Satellite Navigation Systems with using Corrective Circuits and Filtering. *International Conference on Methods and Systems of Navigation and Motion Control (MSNMC)*, October 2016, Kyiv, Ukraine. IEEE Xplore. 2016. P. 275–278. DOI: 10.1109/MSNMC.2016.7783160. (date of access: 27.12.2023)
6. Dolintse B., Balakin S. Risk assessment in computer networks inherent in critical infrastructures. *Проблеми інформатизації та управління*. 2022. Вип. 2(70). P. 4–9. DOI: 10.18372/2073-4751.70.16840. (date of access: 27.12.2023)
7. Dolintse B. Architecture of integrated navigation systems with enhanced coordinate accuracy and fault detection. *Проблеми інформатизації та управління*. 2023. Вип. 2(74). P. 31–37. DOI: 10.18372/2073-4751.74.17878. (date of access: 27.12.2023)
8. Zhukov I., Dolintse B. Enhancing accuracy of information processing in onboard subsystems of UAVs. *Technology Audit and Production Reserves*. №5/2(73), 2023. P. 6–10. DOI: 10.15587/2706-5448.2023.287700. (date of access: 27.12.2023)

9. Zhukov I., Dolintse B., Balakin S. Improving the Accuracy of Air Navigation Systems for Unmanned Aerial Vehicles. *13th International Conference on Dependable Systems, Services and Technologies (DESSERT-2023)*, Athens, October 13–15, 2023. IEEE. 2023. P. 1–7. DOI: 10.1109/DESSERT61349.2023.10416511. (date of access: 27.12.2023)

10. Dolintse B. Robust LeGNSS positioning subsystem for UAV with correction and optimal filtering. *Проблеми інформатизації та управління*. 2023. Вип. 4(76). P. 10–16. DOI: 10.18372/2073-4751.76.18235. (date of access: 27.12.2023)

11. Васильєв В.М., Рогожин В.О., Долінце Б.І. Моделювання та оптимізація параметрів інтегрованої інерціально-супутникової навігаційної системи з коригувальними ланцюгами. *Молодий вчений*. 2016. Вип. 7 (34). С. 184–187.

12. Долінце Б.І. Вплив навігаційних похибок на прийняття рішень в системах попередження зіткнень літаків. *Політ. Сучасні проблеми науки: XII міжнар. науково-практ. конф. студентів та молодих учених, м. Київ, 4–5 квітня 2012 р.: тези доп.* Київ: НАУ, 2012. С. 30.

13. Долінце Б.І. Дослідження характеристик інерціально-супутникової навігаційної системи з коректорами в динамічних режимах роботи. *Політ. Сучасні проблеми науки: XIII міжнар. науково-практ. конф. студентів та молодих учених, м. Київ, 3–4 квітня 2013 р.: тези доп.* Київ: НАУ, 2013. С. 123.

14. Долінце Б.І. Сучасні методи та засоби обробки інформації в інтегрованих навігаційних системах літальних апаратів. *Політ. Сучасні проблеми науки: XV міжнар. науково-практ. конф. студентів та молодих учених, 8–9 квітня 2015 р.: тези доп.* Київ: НАУ, 2015. С. 25.

15. Dolintse B. Modern Trends and Issues of the Development and Improve the Accuracy of Navigation Systems for UAVs. В: *Safety in Aviation and Space Technologies: Proceeding of The Seventh World Congress “Aviation in the XXI-st Century”*, 19–21 вересня 2016, Kyiv. Vol.2. P. 59–62.

16. Zhukov I., Dolintse B., Balakin S. Using corrective circuits and filtering for integration of low-cost inertial and satellite navigation systems. *Safety in Aviation and Space Technologies: Proceeding of The Twenty Two World Congress “Aviation in the XXI-st Century”*, September 28–30, 2022, Kyiv. 2022. Vol.2.2. P. 8–12.

17. Долінце Б.І. Методи зменшення похибок інерціально-супутникової навігаційної системи. *Проблеми навігації і управління рухом: всеукр. наук.–*

практ. конф. молодих учених та студентів, 28–29 листопада 2012, Київ. Тези доп. НАУ. 2012. С. 45.

18.Васильєв В.М., Долінце Б.І. Інерціально-супутникові навігаційні системи для безпілотних літальних апаратів. *Сучасні проблеми авіакосмічних технологій та систем*: науково-практичний семінар, 17–21 червня 2013, м. Житомир. Тези доп. НАУ. 2013. С. 6.

19.Журавель Н.В., Долінце Б.І. Підвищення точності навігаційного сигналу за допомогою фільтру Калмана. *Збірник тез доповідей XIV Міжнародної науково-практичної конференції “Комп’ютерні системи та мережні технології” (CSNT-2023)*, Київ, 13–14 квітня 2023. Київ: НАУ. 2023. С. 92–93.

20.А. с. №122052 Україна. Комп’ютерна програма «Система адаптивної обробки інформації в багато-супутниковій системі позиціонування LeGNSS» / Б. І. Долінце. 2023.

21.Khelifi M., Butun I. Localization in Unprecedentedly Crowded Airspace for UAVs and SUAVs. *IEEE Access*. 2022. Vol. PP. P. 1–1. DOI: 10.1109/ACCESS.2022.3181377. (date of access: 27.12.2023)

22.Otto A., Agatz N., Campbell J., Golden B., Pesch E. Optimization approaches for civil applications of unmanned aerial vehicles (UAVs) or aerial drones: A survey. *Networks*. 2018. Vol. 72. P. 411–458. DOI: 10.1002/net.21818. (date of access: 27.12.2023)

23.Watson R. M., Sivaneri V., Gross J. N. Performance Characterization of Tightly-Coupled GNSS Precise Point Positioning Inertial Navigation within a Simulation Environment. *In AIAA Guidance, Navigation, and Control Conference*. American Institute of Aeronautics and Astronautics. 2016. DOI: 10.2514/6.2016-1869. (date of access: 27.12.2023)

24.Saponaro M., Tarantino E., Fratino U. Geometric Accuracy Evaluation of Geospatial Data Using Low-Cost Sensors on Small UAVs. *Computational Science and Its Applications*. Springer International Publishing. 2018. С. 364–374. DOI: 10.1007/978-3-319-95174-4_29. (date of access: 27.12.2023)

25.In H., Hatano H., Fujii M., Ito A., Watanabe Y. A study on reliable position estimation in kinematic positioning for single frequency GNSS receiver. *International*

Conference on ITS Telecommunications (ITST), 29–31 May 2017, Warsaw, Poland. IEEE. 2017. P. 1–7. DOI: 10.1109/ITST.2017.7972227. (date of access: 27.12.2023)

26.Chen B., Zhu Z., Zhang Y., Yue S., Yu B., Du H. Indoor UAV Positioning System Based on Rotational Vision and Cascade Controller. *Chinese Control Conference (CCC)*, 25–27 July 2022, Hefei, China. 2022. P. 3621–3626. DOI: 10.23919/CCC55666.2022.9902566. (date of access: 27.12.2023)

27.Ersoy S., Waqar T., Toptas E. Automatic decision making system with environmental and traffic data. *Mathematical Models in Engineering*. JVE International. Vol. 7, №2. 2021. P. 30–39. DOI: 10.21595/mme.2021.22020. (date of access: 27.12.2023)

28.Takács B., Siki Z., Markovits-Somogyi R. Extension of RTKLIB for the calculation and validation of protection levels. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS)*. 2017. Vol. XLII-4/W2. P. 161–166. DOI: 10.5194/ISPRS-ARCHIVES-XLII-4-W2-161-2017. (date of access: 27.12.2023)

29.Lo Presti L., Sabina S. Introduction and Book Objectives. *GNSS for Rail Transportation*. Springer International Publishing. 2018. P. 1–6. DOI: 10.1007/978-3-319-79084-8_1. (date of access: 27.12.2023)

30.Аверін Д., Боровицький В., Микитенко В. Системи позиціонування для дронів, які використовують цифрові камери. *Вісник Київського політехнічного інституту. Серія: Приладобудування*. Липень 2022. Вип. 63(1). С. 20–25. DOI: 10.20535/1970.63(1).2022.260632. (date of access: 27.12.2023)

31.Мелешко В.В., Нестеренко О.И. Безплатформні інерційні навігаційні системи: навч. посібник. Кропивницький: ПОЛІМЕД-Сервіс, 2011. 164 с.

32.Hrabchak V.I., Vankevych P.I., Ivanyuk Y.H. Застосування методів варіаційного числення до опису руху "ідеальної" ракети у вертикальній площині. *Військово-технічний збірник*. Травень 2014. Вип. 10. С. 24–27. DOI: 10.33577/2312-4458.10.2014.24-27. (date of access: 27.12.2023)

33.Masiero A., Sofia G., Tarolli P. Quick 3D with UAV and TOF camera for geomorphometric assessment. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS)*. Vol. XLIII-B1-2020. Copernicus

GmbH. 2020. P. 259–264. DOI: 10.5194/isprs-archives-XLIII-B1-2020-259-2020. (date of access: 27.12.2023)

34. Dilshad Naqqash, Ullah Amin, Kim Jaeho. LocateUAV: Unmanned Aerial Vehicle Location Estimation via Contextual Analysis in an IoT Environment. *Internet of Things Journal*. IEEE. 2023. Vol. 10, No. 5. P. 4021–4033.

35. Zeng Y., Tang J. Real-time Data Acquisition and Processing under Mobile Edge Computing-assisted UAV System. *GLOBECOM 2022 – IEEE Global Communications Conference*, December 04, 2022, Rio de Janeiro, Brazil. 2022. P. 5680–5685. DOI: 10.1109/GLOBECOM48099.2022.10001477. (date of access: 27.12.2023)

36. Zeng Y., Tang J. MEC-Assisted Real-Time Data Acquisition and Processing for UAV With General Missions. *IEEE Transactions on Vehicular Technology*. January 2023. Vol. 72, No. 1. P. 1058–1072. DOI: 10.1109/TVT.2022.3203704. (date of access: 27.12.2023)

37. Rieke M., Foerster T., Geipel J., Prinz T. High-precision positioning and real-time data processing of UAV systems. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS)*. Vol. XXXVIII-1/C22. 2011. P. 119–124. DOI: 10.5194/isprsarchives-XXXVIII-1-C22-119-2011. (date of access: 27.12.2023)

38. Sakir R.K.A., Ramli M.R., Lee J.-M., Kim D.-S. UAV-assisted Real-time Data Processing using Deep Q-Network for Industrial Internet of Things. *International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)*, Fukuoka, Japan. 2020. P. 208–211. DOI: 10.1109/ICAIIIC48513.2020.9065203. (date of access: 27.12.2023)

39. Dewedar M.A., Kamel A.M., Hassan M.A., Elhalwagy Y.Z. Real-Time Aerial Targets Tracking for Anti-UAV Systems Using OpenCV. *International Conference on Electrical Engineering (ICEENG)*, Cairo, Egypt. 2022. P. 1–4. DOI: 10.1109/ICEENG49683.2022.9781864. (date of access: 27.12.2023)

40. Jung S., Kim J. Adaptive and stabilized real-time super-resolution control for UAV-assisted smart harbor surveillance platforms. *Real-Time Image Proc.* 2021. Vol. 18. P. 1815–1825. DOI: 10.1007/s11554-021-01163-2. (date of access: 27.12.2023)

41. Liao Y.-H., Juang J.-G. Real-Time UAV Trash Monitoring System. *Applied Science*. 2022. Vol. 12. P. 1838. DOI: 10.3390/app12041838. (date of access: 27.12.2023)
42. Wang Ping, Luo Xianquan, Junwei Lv. Research on Real-time and High-precision Positioning Method of Ground Target through UAV Stereo Vision and Spatial Information Fusion. *Recent Advances in Electrical & Electronic Engineering*. 2023. Vol. 16(3). DOI: 10.2174/2352096515666220621100306. (date of access: 27.12.2023)
43. Desta Ekaso, Francesco Nex, Norman Kerle. Accuracy assessment of real-time kinematics (RTK) measurements on unmanned aerial vehicles (UAV) for direct georeferencing. *Geo-spatial Information Science*. 2020. Vol. 23:2. P. 165–181. DOI: 10.1080/10095020.2019.1710437. (date of access: 27.12.2023)
44. Chhikara P., Tekchandani R., Kumar N., Chamola V., Guizani M. DCNN-GA: A Deep Neural Net Architecture for Navigation of UAV in Indoor Environment. *IEEE Internet of Things Journal*. 2021. Vol. 8, № 6. P. 4448–4460. DOI: 10.1109/JIOT.2020.3027095. (date of access: 27.12.2023)
45. Rychlicki Mariusz, et al. Analysis of Accuracy and Reliability of Different Types of GPS Receivers. *Sensors*. 2019. Vol. 20, № 22. P. 6498. DOI: 10.3390/s20226498. (date of access: 27.12.2023)
46. Pai K.R., Marakala N. A review on Inertial navigational systems. 2016 *International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, Chennai, India. 2016. P. 1682–1686. DOI: 10.1109/ICEEOT.2016.7754971. (date of access: 27.12.2023)
47. Lanzagorta M., Uhlmann J., Venegas-Andraca S.E. Quantum sensing in the maritime environment. *OCEANS 2015, MTS/IEEE Washington*, Washington, DC, USA. 2015. P. 1–9. DOI: 10.23919/OCEANS.2015.7401973. (date of access: 27.12.2023)
48. Wandinger U. Introduction to Lidar. In: Weitkamp C. (eds) *Lidar. Springer Series in Optical Sciences*, vol 102. Springer, New York, NY. 2005. DOI: 10.1007/0-387-25101-4_1. (date of access: 27.12.2023)
49. Hu X., Olesen D., Knudsen P. Toward high-quality magnetic data survey using UAV: development of a magnetic-isolated vision-based positioning system. *GPS*

Solutions. 2021. Vol. 25, № 29. P. 1–17. DOI: 10.1007/s10291-020-01076-w. (date of access: 27.12.2023)

50.Arvai L., Homolya S. Filtering Methods for Indoor Magnetic Compass. *International Carpathian Control Conference (ICCC)*, High Tatras, Slovakia. 2020. P. 1–5. DOI: 10.1109/ICCC49264.2020.9257243. (date of access: 27.12.2023)

51.Sandmann André, et al. Baseband Linearity and Interference Effects in Intensity Modulated and Direct Detected Optical MIMO Systems. *Optical Fiber Technology*. 2020. Vol. 60. P. 102329. DOI: 10.1016/j.yofte.2020.102329. (date of access: 27.12.2023)

52.Ren Zhongju, et al. Advanced Study of Optical Imaging Systems for Virtual Reality Head-Mounted Displays. *Photonics*. 2023. Vol. 10, № 5. P. 555. DOI: 10.3390/photonics10050555. (date of access: 27.12.2023)

53.Gageik Nils, Strohmeier M., Montenegro S. An Autonomous UAV with an Optical Flow Sensor for Positioning and Navigation. *International Journal of Advanced Robotic Systems*. 2013. DOI: 10.5772/56813. (date of access: 27.12.2023)

54.Bao M. Introduction to MEMS Devices. B: Analysis and Design Principles of MEMS Devices. Elsevier. 2005. 32 p. DOI: 10.1016/B978-044451616-9/50002-3. (date of access: 27.12.2023)

55.Elnaggar A., El-Bassiouny A., Mosa G. Perturbation analysis of an electrostatic Micro-Electro-Mechanical System (MEMS) subjected to external and non-linear parametric excitations. *International Journal of Basic and Applied Sciences*. 2014. Vol. 3. P. 209–223. DOI: 10.14419/IJBAS.V3I3.2772. (date of access: 27.12.2023)

56.Finkbeiner S. MEMS for automotive and consumer electronics. *Proceedings of the European Solid-State Device Research Conference (ESSDERC)*, IEEE, Sep. 2013, Bucharest, Romania. 2013. P. 9–14. DOI: 10.1109/ESSDERC.2013.6818809. (date of access: 27.12.2023)

57.Maenaka K. MEMS inertial sensors and their applications. *International Conference on Networked Sensing Systems*, June 2008, Kanazawa, Japan. 2008. P. 71–73. DOI: 10.1109/INSS.2008.4610859. (date of access: 27.12.2023)

58.Engel P. The Global Positioning System: Signals, measurements, and performance. *International Journal of Wireless Information Networks*. 2010. Vol. 1. P. 83–105. DOI: 10.1007/BF02106512. (date of access: 27.12.2023)

59. Xu G., Xu Y. Coordinate and Time Systems. *GPS*. Springer, Berlin, Heidelberg. 2016. P. 17–36. DOI: 10.1007/978-3-662-50367-6_2. (date of access: 27.12.2023)
60. Agudelo G., Wang G., Liu Y., Bao Y., Turco M. J. GPS geodetic infrastructure for subsidence and fault monitoring. *Proceedings of the International Association of Hydrological Sciences*, 22 April 2020, Houston, Texas, USA. 2020. Vol. 382. P. 11–18. DOI: 10.5194/piahs-382-11-2020. (date of access: 27.12.2023)
61. O'Regan G. GPS. Book: The Innovation in Computing Companion. Springer. 2018. P. 137–141. DOI: 10.1007/978-3-030-02619-6_28. (date of access: 27.12.2023)
62. Lasisi L.S., Chris C.R. A Study of a Low-Cost Navigation Augmentation System: The United Kingdom's Immediate Answer to the Galileo Brexit Conundrum. *Recent Developments in Engineering Research*. Vol. 11. 2021. P. 41–59. DOI: 10.9734/bpi/rder/v11/6732D. (date of access: 27.12.2023)
63. Chronopoulos S. K., Koliopoulos C., Pappa A., Angelis C. T. Simulation of a Feasible Galileo System Operating in L1 and E5 Bands. Book: Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. Springer, Berlin, Heidelberg. 2010. P. 35–43. DOI: 10.1007/978-3-642-13618-4_3. (date of access: 27.12.2023)
64. Angelis C.T. Simulation analysis and performance of a feasible GNSS system with multi-beam antennas deployment operating in Galileo frequency bands. *International Journal of Microwave and Wireless Technologies*. 2012. Vol. 4(5). P. 537–543. DOI: 10.1017/S1759078712000529. (date of access: 27.12.2023)
65. Gehrt J., Breuer M., Konrad T., Abel D. A pseudolite position solution within a Galileo test environment for automated vehicle applications. *European Navigation Conference (ENC)*, May 2017, Lausanne, Switzerland. IEEE. P. 135–142. DOI: 10.1109/EURONAV.2017.7954202. (date of access: 27.12.2023)
66. Tang J., Gao X., Li Y. et al. Study of ionospheric responses over China during September 7–8, 2017 using GPS, Beidou (GEO), and Swarm satellite observations. *GPS Solut.* 2022. Vol. 26. P. 55. DOI: 10.1007/s10291-022-01244-0. (date of access: 27.12.2023)
67. Cheng J., Jiang C., Jia C., Ding J., Li H., Qi B. Evaluation of Ionospheric Delay Extraction Model Using Dual-Frequency Multisystem Observations. *IEEE Sensors Journal*. Vol. 23, №14. 2023. P. 16197-16209. (date of access: 27.12.2023)

68. Shi C., Zhao Q., Li M., Tang W., Hu Z., Lou Y., Zhang H., Niu X., Liu J. Precise orbit determination of Beidou Satellites with precise positioning. *Science China Earth Sciences*. 2012. Vol. 55. P. 1079–1086. DOI: 10.1007/s11430-012-4446-8. (date of access: 27.12.2023)
69. Yin Hua L., Xiao Hui L., Jun R., Hui Jun Z., Lin Y. The analysis of Differential Code Bias of BeiDou satellite navigation system. *IEEE International Frequency Control Symposium (FCS)*, Taipei, Taiwan. 2014. P. 1–5. DOI: 10.1109/FCS.2014.6859963. (date of access: 27.12.2023)
70. Hegarty C., Chatre E. Evolution of the Global Navigation Satellite System (GNSS). *Proceedings of the IEEE*. 2008. Vol. 96. P. 1902–1917. DOI: 10.1109/JPROC.2008.2006090. (date of access: 27.12.2023)
71. Bryson M., Sukkarieh S. UAV Localization Using Inertial Sensors and Satellite Positioning Systems. *Handbook of Unmanned Aerial Vehicles*. Springer, Dordrecht. 2015. P. 433–460. DOI: 10.1007/978-90-481-9707-1. (date of access: 27.12.2023)
72. Won J.-H., Pany T. Signal Processing. *Springer Handbook of Global Navigation Satellite Systems*. Springer. 2017. P. 401–442. DOI: 10.1007/978-3-319-42928-1_14. (date of access: 27.12.2023)
73. Elkaim G. H., Lie F. A. P., Gebre-Egziabher D. Principles of Guidance, Navigation, and Control of UAVs. *Handbook of Unmanned Aerial Vehicles*. Springer, Netherlands. 2014. P. 347–380. DOI: 10.1007/978-90-481-9707-1_56. (date of access: 27.12.2023)
74. Xu Yifan, et al. A Vehicle-Model-Aided Navigation Reconstruction Method for a Multicopter during a GPS Outage. *Electronics*. 2020. Vol. 10, № 5. P. 528. DOI: 10.3390/electronics10050528. (date of access: 27.12.2023)
75. Waegli A., Skaloud J. Optimization of two GPS/MEMS-IMU integration strategies with application to sports. *GPS Solutions*. 2009. Vol. 13. P. 315–326. DOI: 10.1007/S10291-009-0124-5. (date of access: 27.12.2023)
76. Strohmeier Michael, Montenegro Sergio. Coupled GPS/MEMS IMU Attitude Determination of Small UAVs with COTS. *Electronics*. 2017. Vol. 6, № 1. P. 15. DOI: 10.3390/electronics6010015. (date of access: 27.12.2023)
77. Kim Jeong Won, Hwang Dong-Hwan, Lee Sang Jeong. A Deeply Coupled GPS/INS Integrated Kalman Filter Design Using a Linearized Correlator Output.

IEEE/ION Position, Location, And Navigation Symposium, April 2006, Coronado, CA. 2006. P. 300–305. DOI: 10.1109/PLANS.2006.1650616. (date of access: 27.12.2023)

78. Dawson E., Rashed M., Abdelfatah W., Noureldin A. Radar-Based Multisensor Fusion for Uninterrupted Reliable Positioning in GNSS-Denied Environments. *IEEE Transactions on Intelligent Transportation Systems*. 2022. Vol. 23. P. 23384–23398. DOI: 10.1109/TITS.2022.3202139. (date of access: 27.12.2023)

79. Baldoni S., Battisti F., Brizzi M., Neri A. GNSS-Imaging Data Fusion for Integrity Enhancement in Autonomous Vehicles. *IEEE Transactions on Aerospace and Electronic Systems*. 2022. Vol. 58. P. 4690–4704. DOI: 10.1109/TAES.2022.3165771. (date of access: 27.12.2023)

80. Singh U. K., Shankar M. R. B., Ottersten B. Opportunistic localization using LEO signals. *Asilomar Conference on Signals, Systems, and Computers*, 31 October 2022, Pacific Grove, USA. 2022. P. 894–899. DOI: 10.1109/IEEECONF56349.2022.10051941. (date of access: 27.12.2023)

81. Prol F. et al. Position, Navigation, and Timing (PNT) Through Low Earth Orbit (LEO) Satellites: A Survey on Current Status, Challenges, and Opportunities. *IEEE Access*. 2022. Vol. 10. P. 83971–84002. DOI: 10.1109/access.2022.3194050. (date of access: 27.12.2023)

82. Ferre R., Praks J., Seco-Granados G., Lohan E. A Feasibility Study for Signal-in-Space Design for LEO-PNT Solutions With Miniaturized Satellites. *IEEE Journal on Miniaturization for Air and Space Systems*. 2022. Vol. 3. № 4. P. 171–183. DOI: 10.1109/JMASS.2022.3206023. (date of access: 27.12.2023)

83. Khalife J., Neinavaie M., Kassas Z. Navigation With Differential Carrier Phase Measurements From Megaconstellation LEO Satellites. *IEEE/ION Position, Location and Navigation Symposium (PLANS)*, Apr. 2020, Portland, USA. 2020. P. 1393–1404. DOI: 10.1109/PLANS46316.2020.9110199. (date of access: 27.12.2023)

84. Humphreys Todd E., Iannucci Peter A., Komodromos Zacharias M., Graff Andrew M. Signal Structure of the Starlink Ku-Band Downlink. *IEEE Transactions on Aerospace and Electronic Systems*. 2023. Vol. 59, № 5. P. 6016–6030. DOI: 10.1109/taes.2023.3268610. (date of access: 27.12.2023)

85. Han S., Lee W., Shin W., Kim J. Adaptive Beam Size Design for LEO Satellite Networks with Doppler Shift Compensation. *IEEE VTS Asia Pacific Wireless*

Communications Symposium (APWCS), 24 August 2022, Seoul, Republic of Korea. 2022. P. 26–30. DOI: 10.1109/APWCS55727.2022.9906490. (date of access: 27.12.2023)

86.Iannucci P., Humphreys T. Fused Low-Earth-Orbit GNSS. *arXiv: Signal Processing*. 2020. P. 1–19. DOI: 10.48550/ARXIV.2009.12334. (date of access: 27.12.2023)

87.Wunsche R., Krondorf M., Liang T.-J. Dummy Based Blind Beam Acquisition and Switching Strategy for DVB-S2X Applications in LEO Satellite Environments. *IEEE Conference on Antenna Measurements and Applications (CAMA)*, November 2023, Genoa, Italy. 2023. P. 221–226. DOI: 10.1109/CAMA57522.2023.10352768. (date of access: 27.12.2023)

88.Gomez-del-Hoyo P., Gronowski K., Samczyński P. The STARLINK-based passive radar: preliminary study and first illuminator signal measurements. 2022 23rd International Radar Symposium (IRS), 12–14 September 2022, Gdansk, Poland. 2022. P. 350–355. DOI: 10.23919/irs54158.2022.9905046. (date of access: 27.12.2023)

89.Neinavaie, Mohammad, Kassas, Zaher M. Unveiling Beamforming Strategies of Starlink LEO Satellites. *International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+)*, September 2022, Denver, Colorado. Proceedings. 2022. P. 2525–2531. DOI: 10.33012/2022.18580. (date of access: 27.12.2023)

90.Zehao L., Hanting Q., Chenjia W. Online Blind Extraction Algorithm For WSN Signal Based on UKF. *International Conference on Intelligent Systems Design and Engineering Applications (ISDEA)*, August 2015, Guiyang, China. IEEE. 2015. P. 542–545. DOI: 10.1109/ISDEA.2015.140. (date of access: 27.12.2023)

91.Zhang X., Wei W. Blind adaptive multiuser detection based on Kalman filtering. *IEEE Transactions on Signal Processing*. 2002. Vol. 50. №1. P. 87–95. DOI: 10.1109/78.972485. (date of access: 27.12.2023)

92.Kozhaya S.E., Kassas Z.M. Positioning with Starlink LEO Satellites: A Blind Doppler Spectral Approach. *Vehicular Technology Conference (VTC)*, June 2023, Florence, Italy. IEEE. 2023. P. 1–5. DOI: 10.1109/VTC2023-Spring57618.2023.10199264. (date of access: 27.12.2023)

93. Psiaki M.L. Navigation using carrier Doppler shift from a LEO constellation: TRANSIT on steroids. *Navigation*. 2021. Vol. 68. №3. P. 621–641. DOI: 10.1002/navi.438. (date of access: 27.12.2023)
94. Hunerbein K., Lange W. Timing and Navigation in UAVs: Synchronization of UAV Swarms and Testing GPS Error Effects on GNSS Reception. *Embedded Selforganising Systems*. 2018. Vol. 5. №1. P. 17–25. DOI: 10.14464/ess51265. (date of access: 27.12.2023)
95. Sabatini R., Moore T., Ramasamy S. Global navigation satellite systems performance analysis and augmentation strategies in aviation. *Progress in Aerospace Sciences*. Elsevier BV. 2017. Vol.95. P. 45–98. DOI: 10.1016/J.PAEROSCI.2017.10.002. (date of access: 27.12.2023)
96. Layh T., Gebre-Egziabher D. Design for graceful degradation and recovery from GNSS interruptions. *Aerospace and Electronic Systems Magazine*. IEEE. 2017. Vol. 32. P. 4–17. DOI: 10.1109/MAES.2017.160194. (date of access: 27.12.2023)
97. Warren M., Paton M., MacTavish K., Schoellig A. P., Barfoot T. D. Towards Visual Teach and Repeat for GPS-Denied Flight of a Fixed-Wing UAV. *Field and Service Robotics*. Springer International Publishing. 2017. P. 481–498. DOI: 10.1007/978-3-319-67361-5_31. (date of access: 27.12.2023)
98. Zhuang Y. et al. Multi-sensor integrated navigation/positioning systems using data fusion: From analytics-based to learning-based approaches. *Information Fusion*. Elsevier BV. Vol. 95. 2023. P. 62–90. DOI: 10.1016/j.inffus.2023.01.025. (date of access: 27.12.2023)
99. Wang D., Li M., Huang X., Zhang X. *Spacecraft Autonomous Navigation Technologies Based on Multi-Source Information Fusion*. Springer, Singapore. 2021. 361 p.
100. Wang G. et al. A GNSS/INS Integrated Navigation Algorithm Based on Kalman Filter. *IFAC-Papers Online*. Elsevier BV. 2017. Vol. 51, № 17. P. 232–237. DOI: 10.1016/j.ifacol.2018.08.151. (date of access: 27.12.2023)
101. Kogure S., Ganeshan A. S., Montenbruck O. Regional Systems. *Springer Handbook of Global Navigation Satellite Systems*. Springer International Publishing. 2017. P. 305–337. DOI: 10.1007/978-3-319-42928-1_11. (date of access: 27.12.2023)

- 102.Rizk H., Shokry A., Youssef M. Effectiveness of Data Augmentation in Cellular-based Localization Using Deep Learning. *Wireless Communications and Networking Conference (WCNC)*, April 2019, Marrakesh, Morocco. IEEE. 2019. P. 1–6. DOI: 10.1109/WCNC.2019.8886005. (date of access: 27.12.2023)
- 103.Chen R., Zhao L. Multi-level autonomous integrity monitoring method for multi-source PNT resilient fusion navigation. *Satellite Navigation*. Springer Science and Business Media. 2023. Vol. 4, № 1. P. 1–17. DOI: 10.1186/s43020-023-00111-8. (date of access: 27.12.2023)
- 104.Zhang P., Tu R., Zhang R., Gao Y., Cai H. Combining GPS, BeiDou, and Galileo Satellite Systems for Time and Frequency Transfer Based on Carrier Phase Observations. *Remote Sensing*. MDPI AG. 2018. Vol. 10, № 2(324). P. 1–13. DOI: 10.3390/rs10020324. (date of access: 27.12.2023)
- 105.Zhu Z., Guo Y., Ye W. A Real-time Gravity Compensation Method for a High-Precision Airborne Position and Orientation System based on a Gravity Map. *Journal of Navigation*. 2017. Vol. 71. P. 711–728. DOI: 10.1017/S0373463317000790. (date of access: 27.12.2023)
- 106.Xu K., Zhang G., Jia P., Hao X., Li D. Bias Compensation Model for Sensor Orientation Under Weak Conditions. *Geoscience and Remote Sensing Letters*. IEEE. 2021. Vol. 18. P. 172–176. DOI: 10.1109/LGRS.2020.2968503. (date of access: 27.12.2023)
- 107.Miyawaki K., Chiba K., Isobe H. Design of a positioning device using piezo element and investigation of a simple compensation method. *International Symposium on Micro-NanoMechatronics and Human Science (MHS)*. November 2015, Nagoya, Japan. IEEE. 2015. P. 1–7. DOI: 10.1109/MHS.2015.7438272. (date of access: 27.12.2023)
- 108.Weï S., Wang D., Xu L., Xu L. MEMS-based rotary strapdown inertial navigation system. *Measurement*. 2013. Vol. 46. P. 2585–2596. DOI: 10.1016/J.MEASUREMENT.2013.04.035. (date of access: 27.12.2023)
- 109.Mahmood K., Niki S., Nakahara Y., Lu X., Luque I., Mori K. Autonomous Real-Time Navigation for Service Level Agreement in Distributed Information Service System. *International Symposium on Autonomous Decentralized Systems (ISADS'07)*,

March 2007, Sedona, USA. IEEE. 2007. P. 231–238. DOI: 10.1109/ISADS.2007.24. (date of access: 27.12.2023)

110.Antsaklis P., Astolfi A. Linear State Feedback. Book: Encyclopedia of Systems and Control. Springer London. 2014. P. 1–6. DOI: 10.1007/978-1-4471-5102-9_196-1. (date of access: 27.12.2023)

111.Guo R., Li G. Modification for collection of master-slave synchronized chaotic systems. *Chaos Solitons & Fractals*. 2009. Vol. 40. P. 453–457. DOI: 10.1016/J.CHAOS.2007.07.085. (date of access: 27.12.2023)

112.Medrano-Cerda G. Filtering for linear systems involving time delays in the noise process. *Transactions on Automatic Control*. IEEE. 1983. Vol. 28. P. 801–803. DOI: 10.1109/TAC.1983.1103318. (date of access: 27.12.2023)

113.Alessandri A., Parisini T., Sanguineti M., Zoppoli R. Neural strategies for nonlinear optimal filtering. *International Conference on Systems Engineering*, September 1992, Kobe, Japan. IEEE. 1992. P. 44–49. DOI: 10.1109/ICSYSE.1992.236946. (date of access: 27.12.2023)

114.Attarian A., Batzel J. J., Matzuka B., Tran H. Application of the Unscented Kalman Filtering to Parameter Estimation. Book: Lecture Notes in Mathematics. Springer Berlin Heidelberg. 2013. P. 75–88. DOI: 10.1007/978-3-642-32882-4_4. (date of access: 27.12.2023)

115.Gomez V. Wiener–Kolmogorov Filtering and Smoothing. Book: Multivariate Time Series With Linear State Space Structure. Springer. 2016. C. 449–519. DOI: 10.1007/978-3-319-28599-3_7.

116.Gorev V., Gusev A., Korniienko V. Approximate solutions for the Kolmogorov-Wiener filter weight function for continuous fractional Gaussian noise. *Radio Electronics, Computer Science, Control*. 2021. Vol. 1. P. 29–35. DOI: 10.15588/1607-3274-2021-1-3. (date of access: 27.12.2023)

117.Bergemann K., Reich S. An ensemble Kalman-Bucy filter for continuous data assimilation. *Meteorologische Zeitschrift*. 2012. Vol. 21. №3. P. 213–219. DOI: 10.1127/0941-2948/2012/0307. (date of access: 27.12.2023)

118.Abolhasani M., Rahmani M. Robust Kalman filtering for discrete-time systems with stochastic uncertain time-varying parameters. *Electronics Letters*. 2017. Vol. 53. P. 146–148. DOI: 10.1049/EL.2016.2520. (date of access: 27.12.2023)

- 119.Fang T.H., Kim Y., Park S.G., Seo K., Park S.H. GPS and eLoran Integrated Navigation for Marine Applications Using Augmented Measurement Equation Based on Range Domain. *International Journal of Control, Automation and Systems*. Springer. 2020. Vol. 18, № 9. P. 2349–2359. DOI: 10.1007/s12555-019-0287-y. (date of access: 27.12.2023)
- 120.Wang F., Li H., Lu M. GNSS Spoofing Detection and Mitigation Based on Maximum Likelihood Estimation. *Sensors*. 2017. Vol. 17, № 7. P. 1532. DOI: 10.3390/s17071532. (date of access: 27.12.2023)
- 121.Bos M., Fernandes R., Williams S., Bastos L. Fast error analysis of continuous GNSS observations with missing data. *Journal of Geodesy*. 2013. Vol. 87. P. 351–360. DOI: 10.1007/s00190-012-0605-0. (date of access: 27.12.2023)
- 122.Katsouros V., Papavassiliou V., Emmanouilidis C. A Bayesian Approach for Maintenance Action Recommendation. 2020. Vol. 4. P. 034. DOI: 10.36001/IJPHM.2013.V4I2.2124. (date of access: 27.12.2023)
- 123.Poklukar D. Nonlinear filtering for jump-diffusions. *Journal of Computational and Applied Mathematics*. 2006. Vol. 197. P. 558–567. DOI: 10.1016/J.CAM.2005.11.014. (date of access: 27.12.2023)
- 124.Koenig D., Mammar S. Reduced order unknown input Kalman filter: application for vehicle lateral control. *American Control Conference*, 04–06 June 2003, Denver, USA. IEEE. 2003. Vol. 5. P. 4353–4358 vol.5. DOI: 10.1109/ACC.2003.1240523. (date of access: 27.12.2023)
- 125.Li C., Sun X. W. A Research of Reactive Compensation Scheme Based on the Hybrid Parallel Mode of SVG and PPF. *Applied Mechanics and Materials*. 2014. Vol. 556–562. C. 3627–3632. DOI: 10.4028/www.scientific.net/amm.556-562.3627. (date of access: 27.12.2023)
- 126.Peiris L., Bartl A., Bois J., Plummer A. Passivity control with adaptive feed-forward filtering for real-time hybrid tests. *IFAC Journal of Systems and Control*. Elsevier. 2020. Vol. 12. P. 100081. DOI: 10.1016/j.ifacsc.2020.100081. (date of access: 27.12.2023)
- 127.Cochrane C., Blacksberg J., Anders M., Lenahan P. Vectorized magnetometer for space applications using electrical readout of atomic scale defects in silicon carbide.

Scientific Reports. 2016. Vol. 6. №1. P. 37077. DOI: 10.1038/srep37077. (date of access: 27.12.2023)

128.Han K., Han H., Wang Z., Xu F. Extended Kalman Filter-Based Gyroscope-Aided Magnetometer Calibration for Consumer Electronic Devices. *IEEE Sensors Journal*. 2016. Vol. 17. P. 63–71. DOI: 10.1109/JSEN.2016.2624821. (date of access: 27.12.2023)

129.Madhu A. M., Aditi, Gopal R. Design and analysis of dual axis MEMS magnetometer. *AIP Conference*, 9–11 January 2017, Kerala, India. 2017. Vol. 1849. №1. DOI: 10.1063/1.4984169. (date of access: 27.12.2023)

130.Hoang M., Carratu M., Paciello V., Pietrosanto A. Noise Attenuation on IMU Measurement For Drone Balance by Sensor Fusion. *International Instrumentation and Measurement Technology Conference (I2MTC)*, 17 May 2021, Glasgow, United Kingdom. IEEE. 2021. P. 1–6. DOI: 10.1109/I2MTC50364.2021.9460041. (date of access: 27.12.2023)

131.Haynes D., Corns S., Venayagamoorthy G. K. An Exponential Moving Average Algorithm. *Congress on Evolutionary Computation*, June 2012, Brisbane, Australia. IEEE. 2012. P. 1–8. DOI: 10.1109/CEC.2012.6252962. (date of access: 27.12.2023)

132.Novotnak J., Szoke Z., Smelko M., Lipovsky P., Filko M., Kosuda M. Simple Filtering Algorithms for the Needs of Measuring UAV Parameters. *New Trends in Signal Processing (NTSP)*, 14–16 October 2020, Demanovska Dolina, Slovakia. IEEE. 2020. P. 1–5. DOI: 10.1109/NTSP49686.2020.9229529. (date of access: 27.12.2023)

133.Dagum E., Luati A. A Cascade Linear Filter to Reduce Revisions and False Turning Points for Real Time Trend-Cycle Estimation. *Econometric Reviews*. 2008. Vol.28. № 1–3. P. 40–59. DOI: 10.1080/07474930802387837. (date of access: 27.12.2023)

134.Mao W., Peng J., Zhou Q., Zhang Q., Su Y. Objective Analysis and Scale Separation of Precipitation Field Based on Barnes Wave Filtration Theory of Meteorological Stations. *Journal of Chengdu University Of Information Technology*. 2008. № 6. C. 668–672. (date of access: 27.12.2023)

135.Ertürk S. Real-Time Digital Image Stabilization Using Kalman Filters. *Real Time Imaging*. 2002. Vol. 8. P. 317–328. DOI: 10.1006/rtim.2001.0278. (date of access: 27.12.2023)

136. Kim D., Kim E., Ahn Y., Jeon M. Development of Improved EMC Power Line Filter in New Type. *Journal of electromagnetic engineering and science*. 2008. Vol. 8. P. 153–157. DOI: 10.5515/JKIEES.2008.8.4.153. (date of access: 27.12.2023)
137. Huang D., Leung H., EL-SHEIMY N. Expectation Maximization Based GPS/INS Integration for Land-Vehicle Navigation. *IEEE Transactions on Aerospace and Electronic Systems*. IEEE. 2007. T. 43. № 3. P. 1168–1177. DOI: 10.1109/taes.2007.4383607. (date of access: 27.12.2023)
138. Ebrahim A., Arafa I., Hendy H., Elhalwagy Y. Enhancement of Integrated Navigation System for High-Speed Flying Vehicles Navigation Applications. *Novel Intelligent and Leading Emerging Sciences Conference (NILES)*, 24 October 2020, Giza, Egypt. DOI: 10.1109/niles50944.2020.9257982. (date of access: 27.12.2023)
139. Wang G., Han Y., Chen J., Wang S., Zhang Z., Du N., Zheng Y. A GNSS/INS Integrated Navigation Algorithm Based on Kalman Filter. *IFAC-PapersOnLine*. 2018. Vol. 51. №17. P. 232–237. DOI: 10.1016/J.IFACOL.2018.08.151. (date of access: 27.12.2023)
140. Jiang W., Li Y., Rizos C. Improved decentralized multi-sensor navigation system for airborne applications. *GPS Solutions*. 2018. Vol. 22. P. 1–13. DOI: 10.1007/s10291-018-0743-9. (date of access: 27.12.2023)
141. Nourmohammadi H., Keighobadi J. Decentralized INS/GNSS System With MEMS-Grade Inertial Sensors Using QR-Factorized CKF. *IEEE Sensors Journal*. 2017. Vol. 17. P. 3278–3287. DOI: 10.1109/JSEN.2017.2693246. (date of access: 27.12.2023)
142. Davidson P., Piché R. A method for post-mission velocity and orientation estimation based on data fusion from MEMS-IMU and GNSS. *International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, November 2017, Daegu, Korea (South). IEEE. 2017. P. 576–580. DOI: 10.1109/MFI.2017.8170383. (date of access: 27.12.2023)
143. Xia G., Wang G., Chen X., Xue J. Low-cost MEMS-INS/GNSS integration using quaternion-based nonlinear filtering methods for USV. *OCEANS*, 10–13 April 2016, Shanghai, China. IEEE. 2016. P. 1–7. DOI: 10.1109/OCEANSAP.2016.7485595. (date of access: 27.12.2023)

- 144.Liu B., Zhang L., Li P., Li J., He Z. Positioning Error Characteristics of GPS/INS Integrated Navigation under Different Number of Visible Satellites. *International Conference on Digital Signal Processing*, 26 February 2021, New York, US. ACM. DOI: 10.1145/3458380.3458424. (date of access: 27.12.2023)
- 145.Shaker M., Ghaffari A., Maghsoodpour A., Khodayari A. GPS/INS Integration for Vehicle Navigation based on INS Error Analysis in Kalman Filtering. *International journal of automotive engineering*. 2017. Vol. 7. P. 2562–2570. DOI: 10.22068/IJAE.7.4.2562. (date of access: 27.12.2023)
- 146.Grejner-Brzezinska D., Toth C., Moore T., Raquet J., Miller M. M., Kealy A. Multisensor Navigation Systems: A Remedy for GNSS Vulnerabilities. *Proceedings of the IEEE*. 2016. Vol. 104. P. 1339–1353. DOI: 10.1109/JPROC.2016.2528538. (date of access: 27.12.2023)
- 147.Won S. M. et al. Multimodal Sensing with a Three-Dimensional Piezoresistive Structure. *ACS Nano*. 2019. Vol. 13. № 10. C. 10972–10979. DOI: 10.1021/acsnano.9b02030. (date of access: 27.12.2023)
- 148.Asua E., et al. Electronic Interface for Position Sensing Using Resonant Cavities. *IEEE International Workshop on Advances in Sensors and Interfaces (IWASI)*, 13–14 June 2013, Bari, Italy. 2013. C. 142–147. DOI: 10.1109/IWASI.2013.6576077. (date of access: 27.12.2023)
- 149.Wang R., Hou X., Liu F., Yu Y. GPS/INS Integrated Navigation for Quadrotor UAV Considering Lever Arm. *Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, 16–18 October 2020, Zhanjiang, China. 2020. P.132–136. DOI: 10.1109/YAC51587.2020.9337634. (date of access: 27.12.2023)
- 150.Li T., Zhang H., Gao Z., Chen Q., Niu X. High-Accuracy Positioning in Urban Environments Using Single-Frequency Multi-GNSS RTK/MEMS-IMU Integration. *Remote Sensing*. 2018. Vol. 10. № 2. P. 205. DOI: 10.3390/rs10020205. (date of access: 27.12.2023)
- 151.Reuper B., Becker M., Leinen S. Benefits of Multi-Constellation/Multi-Frequency GNSS in a Tightly Coupled GNSS/IMU/Odometry Integration Algorithm. *Sensors*. 2018. Vol. 18. № 9. P. 3052. DOI: 10.3390/s18093052. (date of access: 27.12.2023)

- 152.Jardak N., Adam R. Practical Use of Starlink Downlink Tones for Positioning. *Sensors*. 2023. Vol. 23. № 6. P. 3234. DOI: 10.3390/s23063234.
- 153.Xu X., Wang C., Jin Z. An analysis method for ISL of multilayer constellation. *Journal of Systems Engineering and Electronics*. 2022. Vol. 33. №4. P. 961–968. DOI: 10.23919/jsee.2022.000093. (date of access: 27.12.2023)
- 154.Groves P. Shadow Matching: A New GNSS Positioning Technique for Urban Canyons. *Journal of Navigation*. 2011. Vol. 64. P. 417–430. DOI: 10.1017/S0373463311000087. (date of access: 27.12.2023)
- 155.Osa J., Castello R., Radulovich J., Gillcrist B., Finocchiaro C. Distributed Positioning, Navigation and Timing (DPNT). *Position Location and Navigation Symposium (PLANS)*, Monterey, USA. 2004. C. 728–732. DOI: 10.1109/PLANS.2004.1309066. (date of access: 27.12.2023)
- 156.Madry S. Precision Navigation and Timing Systems. *Space Systems for Disaster Warning, Response, and Recovery*. Springer. New York. 2014. P. 91–98. DOI: 10.1007/978-1-4939-1513-2_6. (date of access: 27.12.2023)
- 157.Elghamrawy H., Karaim M., Tamazin M., Noureldin A. Experimental Evaluation of the Impact of Different Types of Jamming on Commercial GNSS Receivers. *Applied Sciences*. 2020. Vol.10(12), №4240. P. 1–15. DOI: 10.3390/app10124240. (date of access: 27.12.2023)
- 158.Peng Y., Scales W. Ionospheric Remote Sensing with GNSS. *Encyclopedia*. 2021. Vol.1. № 4. P.1246–1256. DOI: 10.3390/encyclopedia1040094. (date of access: 27.12.2023)
- 159.Rajanna M. R., Kantharaju C. H., Shivamurthy G. Satellite Networks Routing Protocol Issues and Challenges: A Survey. *International Journal of Innovative Research in Computer and Communication Engineering*. 2014. Vol. 2. №2. P.153–157.
- 160.Rabjerg J., Leyva-Mayorga I., Soret B., Popovski P. Exploiting topology awareness for routing in LEO satellite constellations. *IEEE Global Communications Conference (GLOBECOM)*, 07–11 December 2021, Madrid, Spain. IEEE. 2021. P. 1–6. DOI: 10.1109/GLOBECOM46510.2021.9685686. (date of access: 27.12.2023)
- 161.Wang M., Li J., Jiang Y., Di X. Stochastic performance analysis for LEO inter-satellite link based on finite-state Markov chain modeling. *International Conference*

on *Computer Science and Network Technology* (ICCSNT), 19–20 December 2015, Harbin, China. IEEE. 2015. P. 1230–1235. DOI: 10.1109/ICCSNT.2015.7490954.

162. Wang J., Zhou Z., Jiang W., Cai B., Shangguan W. A Multi-constellation Positioning Method Based on Optimal Stochastic Modelling. *China Satellite Navigation Conference (CSNC) 2020 Proceedings: Volume I*. Lecture Notes in Electrical Engineering. Vol. 650 / ed. Sun J., Yang C., Xie J. Singapore: Springer, 2020.

163. Montenbruck O., Steigenberger P., Hauschild A. Broadcast versus precise ephemerides: a multi-GNSS perspective. *GPS Solutions*. 2015. Vol. 19. P. 321–333. DOI: 10.1007/s10291-014-0390-8. (date of access: 27.12.2023)

164. Tomastik J., Chudá J., Tunak D., Chudy F., Kardos M. Advances in smartphone positioning in forests: dual-frequency receivers and raw GNSS data. *Forestry: International Journal of Forest Research*. 2021. Vol. 94, № 2. C. 292–310. DOI: 10.1093/forestry/cpaa032. (date of access: 27.12.2023)

165. Portillo I., Cameron B. G., Crawley E. F. A technical comparison of three low earth orbit satellite constellation systems to provide global broadband. *Acta Astronautica*. 2019. Vol. 159. P. 123–135. DOI: 10.1016/J.ACTAASTRO.2019.03.040. (date of access: 27.12.2023)

166. Pan L. et al. Satellite Availability and Point Positioning Accuracy Evaluation on a Global Scale for Integration of GPS, GLONASS, BeiDou and Galileo. *Advances in Space Research*. 2019. Vol. 63, № 9. P. 2696–2710. DOI: 10.1016/j.asr.2017.07.029. (date of access: 27.12.2023)

167. Pachler N., del Portillo I., Crawley E. F., Cameron B. G. An Updated Comparison of Four Low Earth Orbit Satellite Constellation Systems to Provide Global Broadband. *IEEE International Conference on Communications Workshops (ICC Workshops)*, 14–23 June 2021, Montreal, Canada. 2021. P. 1–7. DOI: 10.1109/ICCWorkshops50388.2021.9473799. (date of access: 27.12.2023)

168. Khalife J., Neinavaie M., Kassas Z. M. The First Carrier Phase Tracking and Positioning Results With Starlink LEO Satellite Signals. *Transactions on Aerospace and Electronic Systems*. IEEE. 2022. Vol. 58, № 2. P. 1487–1491. DOI: 10.1109/TAES.2021.3113880. (date of access: 27.12.2023)

169. Chi Y.-E., Park J., Park S.-O. Hybrid Multibeamforming Receiver With High-Precision Beam Steering for Low Earth Orbit Satellite Communication. *IEEE Transactions on Antennas and Propagation*. 2023. Vol. 71, № 7. P. 5695–5707. DOI: 10.1109/TAP.2023.3277195. (date of access: 27.12.2023)
170. Saroufim J., Hayek S. W., Kassas Z. M. Simultaneous LEO Satellite Tracking and Differential LEO-Aided IMU Navigation. *ION Position, Location and Navigation Symposium (PLANS)*, 24 April 2023, Monterey, USA. IEEE. 2023. P. 179–188. DOI: 10.1109/plans53410.2023.10140087. (date of access: 27.12.2023)
171. Kozhaya S., Kanj H., Kassas Z. M. Multi-Constellation Blind Beacon Estimation, Doppler Tracking, and Opportunistic Positioning with OneWeb, Starlink, Iridium NEXT, and Orbcomm LEO Satellites. *ION Position, Location and Navigation Symposium (PLANS)*, 24 April 2023, Monterey, USA. IEEE. 2023. P. 1184–1195. DOI: 10.1109/plans53410.2023.10139969. (date of access: 27.12.2023)
172. Neinavaie M., Kassas Z. M. Unveiling Starlink LEO Satellite OFDM-Like Signal Structure Enabling Precise Positioning. *IEEE Transactions on Aerospace and Electronic Systems*. IEEE. 2023. P. 1–4. DOI: 10.1109/taes.2023.3265951. (date of access: 27.12.2023)
173. Neinavaie M., Khalife J., Kassas Z. M. Exploiting Starlink Signals for Navigation: First Results. *International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+)*, 20–24 September 2021, St. Louis, Missouri, USA. 2021. P. 2766–2773. DOI: 10.33012/2021.18122. (date of access: 27.12.2023)
174. Neinavaie M., Khalife J., Kassas Z. M. Acquisition, Doppler Tracking, and Positioning With Starlink LEO Satellites: First Results. *IEEE Transactions on Aerospace and Electronic Systems*. 2022. Vol. 58, № 3. P. 2606–2610. DOI: 10.1109/TAES.2021.3127488.
175. Albano M., Hadzic S., Rodriguez J. Use of Negative Information in Positioning Algorithms. *Mobile Multimedia Communications*. Springer, Berlin, Heidelberg. 2012. P. 206–217. DOI: 10.1007/978-3-642-35155-6_17. (date of access: 27.12.2023)
176. Pulatova Z., Usmonov J. Selection of Optimal Methods in Decision Making to Properly Assess the Situation. *International Conference on Information Science and Communications Technologies (ICISCT)*, 04–06 November 2020, Tashkent,

Uzbekistan. IEEE. 2020. P.1–3. DOI: 10.1109/ICISCT50599.2020.9351471. (date of access: 27.12.2023)

177. Neumann I., Kutterer H. A Kalman filter extension for the analysis of imprecise time series. *European Signal Processing Conference (EUSIPCO)*, 03–07 September 2007, Poznan, Poland. IEEE. 2007. P. 1176–1180.

178. Venhovens P. J. T. H., Naab K. Vehicle Dynamics Estimation Using Kalman Filters. *Vehicle System Dynamics*. 1999. Vol. 32, № 2–3. P. 171–184. DOI: 10.1076/vesd.32.2.171.2088. (date of access: 27.12.2023)

179. Wang J. Test Statistics in Kalman Filtering. *Journal of Global Positioning Systems*. 2008. Vol. 7, № 1. P. 81. DOI: 10.5081/JGPS.7.1.81. (date of access: 27.12.2023)

180. Pereira P. Application of Kalman Filter. *Econometric Theory*. 1987. Vol. 3, №2. P. 306–309. DOI: 10.1017/S0266466600010392. (date of access: 27.12.2023)

181. Tew B. Application of Kalman Filter. *Econometric Theory*. 1985. Vol. 1. №1. P. 144–145. DOI: 10.1017/S0266466600011075. (date of access: 27.12.2023)

182. Bianculli A. Stepper motors: application and selection. *IEEE Spectrum*. 1970. Vol. 7. P. 25–29. DOI: 10.1109/MSPEC.1970.5213082. (date of access: 27.12.2023)

183. Brogliato B., Lozano-Leal R. Adaptive algorithm for force/position control of flexible joint manipulators with holonomic constraints. *IEEE Conference on Decision and Control*, 11–13 December 1991, Brighton, UK. IEEE. 1991. Vol. 1. P. 361–366. DOI: 10.1109/CDC.1991.261322. (date of access: 27.12.2023)

184. Gao H., Groves D. Environmental Context Detection for Adaptive Navigation using GNSS Measurements from a Smartphone. *Navigation: Journal of the Institute of Navigation*. 2018. Vol. 65. P. 99–116. DOI: 10.1002/NAVI.221. (date of access: 27.12.2023)

185. Gao H., Groves P. D. Context detection for advanced self-aware navigation using smartphone sensors. ArXiv preprint. 2020. Ver. 1. 21 p. DOI: 10.48550/arXiv.2005.07539. (Preprint. Royal Institute of Navigation: Brighton, UK; ArXiv.2005.07539; date of access: 27.12.2023)

186. Gao H., Groves P. Improving environment detection by behavior association for context-adaptive navigation. *Annual of Navigation*. 2020. Vol. 67. P. 43–60. DOI: 10.1002/navi.349. (date of access: 27.12.2023)

187.Huang M. C., Renau J., Torrellas J. Positional adaptation of processors: application to energy reduction. *Annual International Symposium on Computer Architecture*, 09–11 June 2003, San Diego, USA. IEEE. 2003. P. 157–168. DOI: 10.1109/ISCA.2003.1206997. (date of access: 27.12.2023).

188.Pernici B. Adaptive Services and Energy Efficiency. 2011. P. 165–166. DOI: 10.1007/978-3-642-24755-2_15.

Pernici B. Adaptive Services and Energy Efficiency. In: *Towards a Service-Based Internet. Lecture Notes in Computer Science*, Vol. 6994. Springer, Berlin, Heidelberg, 2011. P. 15. DOI: 10.1007/978-3-642-24755-2_15. (date of access: 27.12.2023)

189.Idziak P., Clarke S. An Analysis of Decision-Making Techniques in Dynamic, Self-Adaptive Systems. *International Conference on Self-Adaptive and Self-Organizing Systems Workshops*, 8–12 September 2014, London, United Kingdom. IEEE. 2014. P. 137–143. DOI: 10.1109/SASOW.2014.23. (date of access: 27.12.2023)

190.Zhang X., Wang S. Adaptive Sub-carrier Spacing OFDM Waveform in LEO Satellite Communication System. 2021 *International Conference on Communication Technology (ICCT)*, 13–16 October 2021, Tianjin, China. IEEE. 2021. P. 1372–1376. DOI: 10.1109/ICCT52962.2021.9657889. (date of access: 27.12.2023)

191.Elghamrawy H., Karaim M., Tamazin M., Noureldin A. Experimental Evaluation of the Impact of Different Types of Jamming informations on Commercial GNSS Receivers. *Applied Sciences*. 2020. Vol. 10, №12(4240). 15 p. DOI: 10.3390/app10124240. (date of access: 27.12.2023)

192.Yang Y., Liu L., Li J., Yang Y., Zhang T., Mao Y., Sun B., Ren X. Featured services and performance of BDS-3. *Science Bulletin*. 2021. Vol. 66, №20. P. 2135–2143. DOI: 10.1016/j.scib.2021.06.013. (date of access: 27.12.2023)

193.Zhu H., Xia L., Wu D., Xia J., Li Q. Study on Multi-GNSS Precise Point Positioning Performance with Adverse Effects of Satellite Signals on Android Smartphone. *Sensors*. Vol. 20, №22. 2020. P. 6447. DOI: 10.3390/s20226447. (date of access: 27.12.2023)

194.Jardak N., Adam R. Practical Use of Starlink Downlink Tones for Positioning. *Sensors*. Vol. 23, №6. 2023. P. 3234. DOI: 10.3390/s23063234. (date of access: 27.12.2023)

195. Neinavaie M., Kassas Z. M. Unveiling Starlink LEO Satellite OFDM-Like Signal Structure Enabling Precise Positioning. *IEEE Transactions on Aerospace and Electronic Systems*. 2023. P. 1–4. DOI: 10.1109/TAES.2023.3265951. (date of access: 27.12.2023)
196. Neinavaie M., Shadram Z., Kozhaya S., Kassas Z. M. First Results of Differential Doppler Positioning with Unknown Starlink Satellite Signals. *IEEE Aerospace Conference (AERO)*, 5–12 March 2022, Big Sky, USA. IEEE. 2022. P. 1–14. DOI: 10.1109/AERO53065.2022.9843493. (date of access: 27.12.2023)
197. Saroufim J., Hayek S., Kassas Z. M. Evaluation of Orbit Errors and Measurement Corrections in Differential Navigation with LEO Satellites. *International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+)*, 11–15 September 2023, Denver, Colorado. IEEE. 2023. P. 2823–2834. DOI: 10.33012/2023.19452. (date of access: 27.12.2023)
198. Meng Q., Hsu L. T. Integrity monitoring for all-source navigation enhanced by Kalman filter based solution separation. *IEEE Sensors Journal*. Vol. 216 №14. P. 15469–15484. DOI: 10.1109/JSEN.2020.3026081.
199. Wang E., Cai M., Pang T. A Simple and Effective GPS Receiver Autonomous Integrity Monitoring and Fault Isolation Approach. *International Conference on Control Engineering and Communication Technology*, 7–9 December 2012, Shenyang, China. IEEE. 2012. P. 657–660. DOI: 10.1109/ICCECT.2012.145. (date of access: 27.12.2023)
200. Koenig D., Mammari S. Reduced order unknown input Kalman filter: application for vehicle lateral control. *American Control Conference*, 4–6 June 2003, Denver, USA. IEEE. 2003. Vol. 5, P. 4353–4358. DOI: 10.1109/ACC.2003.1240523. (date of access: 27.12.2023)
201. Medrano-Cerda G. Filtering for linear systems involving time delays in the noise process. *IEEE Transactions on Automatic Control*. 1983. Vol. 28, №7. P. 801–803. DOI: 10.1109/TAC.1983.1103318. (date of access: 27.12.2023)
202. Філяшкін М.К., Рогожин В.О., Скрипець А.В., Лукінова Т.І. Інерціально-супутникові навігаційні системи: навчальний посібник. Київ: НАУ-друк, 2009. 272 с.

203. Gill D. Error Detection and Correction. *International Journal of Research*. 2014. Vol. 1, № 10. P. 1537–1545.
204. Bhattacharjee S., Rahim L. B. A., Aziz I. B. A. A multibit burst error detection and correction mechanism for application layer. *International Conference on Computer and Information Sciences (ICCOINS)*, 03–05 June 2014, Kuala Lumpur, Malaysia. IEEE. 2014. P. 1–6. DOI: 10.1109/ICCOINS.2014.6868380. (date of access: 27.12.2023)
205. Pervan B., Chan F. Detecting Global Positioning Satellite Orbit Errors Using Short-Baseline Carrier-Phase Measurements. *Journal of Guidance, Control, and Dynamics*. 2003. Vol. 26, No. 1. P. 122–131. DOI: 10.2514/2.5022. (date of access: 27.12.2023)
206. Xu W., Cao N. Research on the strategies for collision avoidance of multi-UAV with three dimensional formation in combination of consensus algorithm and uniform flow. *IET Control Theory & Applications*. 2023. Vol. 17, Issue 18. P. 2379–2401. DOI: 10.1049/cth2.12521. (date of access: 27.12.2023)
207. Zhou S., Zhao P., Jin J. Cascaded GPS/IMU Integrated Positioning System Based on Vehicle Kinematics Model. *Chinese Control And Decision Conference (CCDC)*, 22–24 August 2020, Hefei, China. IEEE. 2020. P. 1134–1139. DOI: 10.1109/CCDC49329.2020.9164392. (date of access: 27.12.2023)
208. Ngo T., Hoang D., Le T. Dynamics Modeling and Pitching Parameters Identification of a Novel Hybrid UAV. *Journal of Aeronautics, Astronautics and Aviation*. 2020. Vol. 52, №1. DOI: 10.6125/JoAAA.202003_52(1).03. (date of access: 27.12.2023)
209. Wong D. R., Ou Q., Sinclair M., Li Y. J., Chen X. Q., Marburg A. Unmanned Aerial Vehicle Flight Model Validation Using On-Board Sensing and Instrumentation. *International Conference on Mechatronics and Machine Vision in Practice*, 2–4 December 2008, Auckland, New Zealand. IEEE. 2008. P. 109–114. DOI: 10.1109/MMVIP.2008.4749516.
210. Belge E., Kaba H., Parlak A., Altan A., Hacıoglu R. Estimation of Small Unmanned Aerial Vehicle Lateral Dynamic Model with System Identification Approaches. *Balkan Journal of Electrical and Computer Engineering*. 2020. Vol. 8, №2. P. 121–126. DOI: 10.17694/bajece.654499.

ДОДАТОК А. Акти впровадження результатів роботи



Бізнес Інкубатор Груп Україна
65039, Україна, м. Одеса,
вул. Канітня 93, офіс 13
ЄДРПОУ: 42957658
E-mail: big.u.odessa@gmail.com
<https://linktr.ee/big.u>

№025/23/У від 06.10.2023

ДОВІДКА про впровадження результатів науково-дослідної роботи

У період 2022 - 2023 років Долінце Богдан Ігорович брав участь у науково-технічних, дослідних і технологічних роботах нашої організації як науковий консультант.

На підставі результатів науково-дослідної роботи, ГС "Бізнес інкубатор груп. Україна" підвищила якість підготовки проектів у сфері авіаційних технологій. Це сприяло зростанню ефективності стартапів, які супроводжує організація, розширенню експертизи для підтримки бізнес-проектів, зміцненню партнерських відносин, що сприяє обміну знань і досвіду між учасниками організації та освітніми установами.

Впровадження нових методів й підходів підвищення ефективності позиціонування безпілотних систем, зокрема на базі інноваційних технологій і компонентів, в сучасних бортових підсистемах БПЛА допомогло побудувати глибоке розуміння їх взаємодії та інтеграції. Опанування нових технологій та методів обробки інформації дозволило компанії розробляти та запускати нові продукти, що відповідають сучасним вимогам ринку. Це, своєю чергою, дозволяє залучати нових партнерів з аерокосмічної та авіаційної галузі, виробників і розробників безпілотних авіаційних комплексів, що сприяє росту та розвитку компетенцій організації.

В результаті цієї співпраці, ГС "Бізнес інкубатор груп. Україна" стала однією з небагатьох організацій в Україні, які активно сприяють розвитку національної галузі систем позиціонування та виробництва безпілотних авіаційних платформ.

Методика дослідження та результати комп'ютерного моделювання інерціальної та багатосупутникової навігаційних систем з оптимальною фільтрацією, запропоновані Долінце Богданом Ігоровичем, були успішно застосовані під час консультування партнерів нашої організації.

Голова


Сергій БАРСУКОВ





Адреса для листування: 02068, м. Київ, а/с 92, тел. (044)593 70 89, e-mail: yata.ko@gmail.com, код ЄДРПОУ 40707135

12 жовтня 2023 року № 04 -23-вд

АКТ

про впровадження результатів кандидатської дисертаційної роботи "Методи та засоби підвищення точності обробки інформації в бортових підсистемах БПЛА" Доліще Богдана Ігоровича

Теоретичні, науково-методичні та практичні результати дисертаційної роботи були впровадженні в практичну діяльність Громадської спілки "Українська авіатранспортна асоціація", а саме:

- науково обгрунтовані рекомендації та розроблений алгоритм підвищення точності позиціонування об'єктів в умовах щільної забудови із використанням оптимальних алгоритмів обробки даних, які б дозволили забезпечити високу продуктивність та надійність отримання інформації про позиціонування на території авіаційних підприємств.
- підготовки навчальних та методичних матеріалів ГС "YATA", які було поширено серед членів спілки, щодо переваг використання підходу побудови багатосупутникових систем позиціонування, що використовуються в умовах щільної забудови та роботах, які вимагають надзвичайно високої точності та робастності інформації, використовуючи модель LeGNSS/INS систем підвищеної точності позиціонування.

Представлені пропозиції, отримані на основі проведеного комплексного дисертаційного дослідження, мають необхідний теоретичний і методологічний рівень, практичну значимість, є надзвичайно актуальними та можуть бути використані для вдосконалення діяльності підприємств авіаційної галузі в Україні.

Виконавчий директор

М. Щербина



Українська асоціація
високотехнологічних підприємств та організацій «КОСМОС»

01133 м. Київ, вул. Амазова, 18/7, e-mail: a-kosmos@ukr.net, www.a-kosmos.com.ua, код ЄДРПОУ 42346137

Вих. №01/10 від 24.10.23

АКТ

про впровадження/використання результатів дисертаційної роботи

Долінце Богдана Ігоровича на тему:

«Методи та засоби підвищення точності обробки інформації в бортових підсистемах БПЛА»

Цей акт складено у тому, що результати дисертаційної роботи Долінце Б.І. використовуються в процесі підготовки освітніх курсів щодо основ аерокосмічної діяльності, лекцій для фахівців, дослідників, інженерів, управлінців, аспірантів, студентів, спрямованих на одержання базових знань з широкого кола проблем аерокосмічної діяльності.

Матеріали роботи також були рекомендовані до застосування учасниками асоціації в діяльності пов'язаній із розробленням, виготовленням, випробуванням, забезпечення процесів комп'ютерного моделювання для сучасних космічних та безпілотних літальних апаратів.

З повагою,

Голова Ради Асоціації

к.ф.-м.п.

В. Васильєв



ГРОМАДСЬКА ОРГАНІЗАЦІЯ
«ВСЕУКРАЇНСЬКИЙ ЦЕНТР РЕФОРМ
ТРАНСПОРТНОЇ ІНФРАСТРУКТУРИ»
NGO «UKRAINIAN CENTER FOR REFORM
OF TRANSPORT INFRASTRUCTURE»

ЄДРПОУ 39976115
Адреса для листування:
03118 м. Київ, вул. Казанська б.97

+380979773367
e-mail: crtiug@gmail.com

В/д №26-10 від 17.10.2023р.

АКТ

про впровадження та використання результатів кандидатської дисертації на тему
"Методи та засоби підвищення точності обробки інформації в бортових підсистемах БПЛА"
Долінце Богдана Ігоровича

Теоретичні положення, наукові та практичні результати отримані Долінце Б.І. у дисертаційній роботі, було використано в роботі Громадської організації «Всеукраїнський центр реформ транспортної інфраструктури» (далі ГО «ВЦРТІ») за такими напрямками:

- Розроблено методичні рекомендації щодо методів застосування алгоритмів обробки даних під час оцінки технічних пропозицій щодо підвищення точності систем позионування, оновлено технічний регламент оцінки точності позионування систем, що використовуються у складних оточуючих умовах та щільній міській забудові. Це дозволило ГО «ВЦРТІ» підвищити ефективність власних дослідницьких проєктів пов'язаних із підтримкою дослідження роботи систем позионування БПЛА та аналізом їх ефективності.
- Долінце Б.І., організував та провів два навчальні семінари для фахівців ГО «ВЦРТІ» щодо переваг використання багатосупутникових систем позионування на основі моделей LeGNSS/INS систем. В результаті було підвищено кваліфікацію фахівців організації.
- ГО «ВЦРТІ», використовує запропоновані методики для підготовки пропозицій щодо покращення систем безпілотних літальних апаратів із використанням запропонованого Долінце Б.І. підходу до комп'ютерного моделювання складних систем. Це дозволило врахувати вимоги до точності та завадостійкості сучасних систем позионування при розробці концепції нового стійкого БПЛА.

ГО «ВЦРТІ», висловлює подяку Долінце Б.І. за внесок в розвиток організації та відкрита до подальшої співпраці з метою впровадження інноваційних рішень. Результати співпраці сприяли підвищенню ефективності реалізації діючих проєктів ГО «ВЦРТІ», розширенню експертизи фахівців організації та врахуванню сучасних тенденцій розвитку БПЛА.

З повагою,

Голова ради ГО «ВЦРТІ»



В.В. Наумов

ДОДАТОК Б. Свідоцтво про реєстрацію авторського права

УКРАЇНА



СВІДОЦТВО

про реєстрацію авторського права на твір

№ 122052

Комп'ютерна програма «СИСТЕМА АДАПТИВНОЇ ОБРОБКИ ІНФОРМАЦІЇ В БАГАТО-СУПУТНИКОВІЙ СИСТЕМІ ПОЗИЦІОНУВАННЯ LeGNSS»

(назв твору)

Автор (співавтор) Додінець Богдан Ігорович

(прізвище, ім'я, по-батькові (за наявності), поштовий (за наявності))

Авторські майнові права належать повністю Додінець Богдан Ігорович, вул. Ореста Левницького, 9/21, кв. 42, м. Київ, Київська обл., 02166

(прізвище, ім'я, по-батькові (за наявності) фізичної особи / найменування юридичної особи, адреса)

Дата реєстрації 18 грудня 2023 р.

Директор Державної організації
«Український національний
офіс інтелектуальної власності
та інновацій»


Олена ОРЛЮК



ДОДАТОК В. Державна реєстрація винаходу

МІНЕКОНОМІКИ
НАЦІОНАЛЬНИЙ ОРГАН ІНТЕЛЕКТУАЛЬНОЇ ВЛАСНОСТІ
ДЕРЖАВНА ОРГАНІЗАЦІЯ
«УКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ ОФІС ІНТЕЛЕКТУАЛЬНОЇ ВЛАСНОСТІ ТА
ІННОВАЦІЙ»
(УКРНОІВІ)

вул. Галицька, буд. 1, м. Київ, 01601, тел. (044) 295-27-06
E-mail: office@ipo.gov.ua, код згідно з ЄДРНОУ 44672629

Розписка про одержання електронної заявки на корисну модель
Вх. № **195325** Дата та час одержання **13.11.2023 01:00:00**

Номер заявки	u202305401 (в подальшому обов'язково посилатись на цей номер)
Заявник	Долінець Богдан Ігорович
Назва корисної моделі	СПОСІБ ПІДВИЩЕННЯ ТОЧНОСТІ ПОЗИЦІОНУВАННЯ БЕЗПЛОТНИХ ЛІТАЛЬНИХ АПАРАТІВ
Адреса для листування	Долінець Богдан Ігорович; вул. Ореста Левицького, 9/21, кв. 42, м. Київ, Деснянський р-н, Київ обл., 02166
Представлені матеріали Заява про державну реєстрацію винаходу (корисної моделі) Опис корисної моделі (сторінок - 3) Формула корисної моделі (сторінок - 1) Реферат (сторінок - 1)	

Примітка: Одержано системою подання електронних заявок

Подана електронна заявка вважається оригіналом. Надіслання паперової копії заявки не вимагається!

Консультації з технічних питань роботи системи подання електронних заявок: (044) 494-05-57.

Сектор реєстрації документів за заявками на об'єкти промислової власності: (044) 494-05-95, (044) 494-05-94.

ДОДАТОК Г. Код програми моделювання роботи підсистеми позиціонування LeGNSS

```

%% Самодіагностика та моделювання LeGNSS в БПЛА
% Файл: main_script.mlx
% Цей файл містить основний код для запуску моделювання роботи системи LeGNSS для БПЛА.
%% Перевірка цілісності всіх функцій та файлів
% Список необхідних файлів та функцій
required_files = {
%Файли моделювання та функції
    'main_script_0.mlx',      % Основний код для моделювання роботи
    'InitialData_1.mlx',     % Вихідні дані для налаштування та проведення моделювання
    'flight_path_2.mlx',     % Скрипт для планування польоту
    'uav_flight_path_calculation_3.mlx', % Операціоналізація плану польоту
    'uav_flight_commands_4.mlx' % Розрахунок управляючих сигналів польоту
    'uav_dynamics_5.mlx',    % Динамічна модель UAV польоту UAV в часі
    'uav_mems_6.mlx',       % Модель MEMS системи позиціонування
    'gps_model_7.mlx',      % Модель глобальної системи позиціонування
    'isns_model_7.mlx',     % Модель ICHC з оптимальною фільтрацією
    'leo_model_7.mlx',      % Модель LEO системи позиціонування
    'LeGNSS_model_8.mlx',   %
%Файли збережених даних
    'api_key.mat',
    'coordinates.mat',
% додайте інші файли за потреби
};
AutoCheck(required_files);
%% Моделювання підсистеми LeGNSS-INS
clear;          % Видалити всі існуючі змінні з робочого середовища Matlab
clc;           % Очистити командне вікно
addpath('data'); % додати в пошук каталог із назвою "data" (знаходиться в поточному)
run('InitialData_1.mlx'); % Завантаження вихідних даних моделювання (крок 1)
run('flight_path_2.mlx'); % Створення базового плану польоту (крок 2)
run('uav_flight_path_calculation_3.mlx'); % Операціоналізація плану польоту та маршруту переміщення
БПЛА (крок 3)
run('uav_flight_commands_4.mlx'); % Створення масиву управляючих сигналів для виконання плану
польоту (крок 4)
run('uav_dynamics_5.mlx'); % Моделювання моделі динамічних режимів роботи БПЛА на маршруті
run('uav_mems_6.mlx'); % Моделювання бортової MEMS
run('gps_model_7.mlx'); % Моделювання GPS

```

```

run('isns_model_7.mlx');           % Моделювання бортової ІСНС з оптимальною фільтрацією
run('leo_model_7.mlx');           % Моделювання бортової LEO
run('LeGNSS_model_8.mlx');       % Моделювання бортової підсистеми LeGNSS

%% ФУНКЦІЯ САМОДІАГНОСТИКИ КОДУ
% Файл: AutoCheck.m
function AutoCheck(required_files)
    % Перевірка наявності кожного файлу
    missing_files = {};
    missing_mat_files = {};
    for i = 1:length(required_files)
        if exist(required_files{i}, 'file') == 0
            if exist(fullfile('data', required_files{i}), 'file') == 0 % Перевірка папки "data"
                if endsWith(required_files{i}, '.mat')
                    missing_mat_files{end+1} = required_files{i};
                else
                    missing_files{end+1} = required_files{i};
                end
            end
        end
    end
    fprintf('Кількість файлів для перевірки: %d\n', length(required_files));
    fprintf('Кількість знайдених файлів: %d\n', length(required_files) - length(missing_files) -
length(missing_mat_files));
    % Якщо які-небудь .mat файли відсутні, вивести повідомлення
    if ~isempty(missing_mat_files)
        missing_mat_files_str = strjoin(missing_mat_files, ', ');
        warning(['Відсутні наступні .mat файли: ' missing_mat_files_str '!']);
    end
    % Якщо які-небудь інші файли відсутні, вивести помилку та зупинити виконання
    if ~isempty(missing_files)
        missing_files_str = strjoin(missing_files, ', ');
        fprintf('Відсутні наступні файли: %s\n', missing_files_str);
        error(['Перевірте наявність цих файлів та спробуйте знову.']);
    end
end

%% Вихідні дані для налаштування та проведення моделювання
% Файл: 1_InitialData.m
%% Вихідні дані для налаштування та проведення моделювання
% Модель часу симуляції та крок

```



```

dt = 1; % крок часу в секундах
T = 20000; % час симуляції в секундах
t = 0:dt:T; % кількість вимірювань
% Параметри землі WGS-84
R_earth = 6378137.0; % (м) радіус Землі, більша піввісь
f_earth = 1/298.257223563; % сплюсненість Землі
e2_earth = 2*f_earth - f_earth^f_earth; % квадрат ексцентриситету Землі
g = [0; 0; 9.8]; % прискорення вільного падіння, м/с^2
% Початкова та кінцева точка польоту
flight_startPoint = [50.43935, 30.43105, 0]; % Початкова точка польоту (широта, довгота, висота в метрах)
flight_endPoint = [50.43970, 30.43072, 0]; % Кінцева точка польоту (широта, довгота, висота в метрах)
% Діапазон дозволеної висоти польоту
flight_altitude = [50, 150]; % Діапазон дозволеної мінімальної та максимальної висоти польоту по
маршруту над землею
% Об'єкти для аерофотозйомки
flight_objectives = [50.4495, 30.5253, 100; 50.4481, 30.5340, 140; 50.4393, 30.5553, 50; 50.4064, 30.4592,
200]; % Опис завдань або місії
time_objectives = 60; % (с) Час зависання біля заданих об'єктів в секундах
%Збереження даних про рух в часі
total_distance = 0; % Ініціалізація загальної дистанції
% Пілотажні параметри БпЛА
max_horisontal_speed = 62/3.6; % максимальна швидкість зміни положення БпЛА (м/с) по осям X та Y
max_vertical_speed = 12; % максимальна швидкість набору висоти БпЛА (м/с) по осі Z, але зниження
швидкості по осі Z відбувається зі швидкістю зниження що не перевищує значення g (прискорення вільного
падіння) мінус значення аеродинамічного опору
accel_time = 14; % час протягом якого БпЛА змінює швидкість від мінімальної до максимальної
(наприклад від мінус 17 до плюс 17 м/с по осі X)
max_tilt_angle = 25; % максимальний дозволений кут нахилу до горизонту (градусів)
max_yaw_rate = 25; % максимальна швидкість обертання бпла навколо осі Z (градусів/с)
max_roll_angle = 15; % стандартний кут крену під час польоту в горизонтальній площині (градусів)
max_pitch_roll_rate = 30; % максимальна швидкість обертання навколо осей Y та X (градусів/с)
current_azimuth = 0; % показання азимута напрямку орієнтування БпЛА в момент початку місії (градусів,
азимут на північ)
% Параметри БпЛА
uav_m = 4.5; % маса квадрокоптера, кг
uav_J = diag([6.158e-2, 6.158e-2, 1.106e-1]); % Матриця інерції БпЛА
uav_Ct = 3.691e-5; % Коефіцієнт тяги пропелера
uav_Cd = 1.170e-1; % Коефіцієнт аеродинамічного опору
uav_rho = 1.225; % Густина повітря, кг/м^3
uav_A = 0.25; % Площа перерізу квадрокоптера, м^2 (приблизне значення)
% Параметри систем позиціонування БпЛА (RMS errors)

```

```

error_bounds = struct();% Define RMS errors for different satellite types (min error (m), max error(m))
error_bounds.GEO = [2.6, 3.6];
error_bounds.IGSO = [2.55, 4.65];
error_bounds.MEO = [2.76, 6.96];
error_bounds.GPS = [2.77, 5.73];
error_bounds.LEO = [6.5, 9.5];
gps_update_rate = 1; % Гц, частота оновлення даних GPS
leo_update_rate = 25; % Гц, частота оновлення даних LEO

%% Скрипт для планування польоту
% Файл: flight_path.m
%% Створення плану польоту
% Алгоритм верифікації плану польоту та завантаження збережених результатів
% У цьому коді ми спершу завантажуюмо uav_flight_path, якщо він існує.
% Потім перевіряємо, чи всі flight_objectives присутні в цьому маршруті.
% Якщо вони присутні, код нічого не робить і продовжує виконання.
% Якщо вони відсутні, код виконує блок для створення нового маршруту польотів.
% Якщо файл не існує, код просто створює новий маршрут польотів.

%% Створення плану польотів
% Визначення мінімальної та максимальної висоти польоту
min_altitude = flight_altitude(1);
max_altitude = flight_altitude(2);

% Ініціалізація маршруту польоту
uav_flight_path = flight_startPoint;

% Take off to desired altitude
desired_altitude = max(min(min_altitude, max_altitude), flight_startPoint(3));
uav_flight_path = [uav_flight_path; flight_startPoint(1), flight_startPoint(2), desired_altitude];

for i = 1:size(flight_objectives, 1)
    obj = flight_objectives(i, 1:2);
    altitude = flight_objectives(i, 3); % висота не менше min_altitude метрів
    if altitude < min_altitude
        altitude = min_altitude; % встановлення мінімальної висоти min_altitude метрів
    end

% Параметри для додавання кругових польотів навколо об'єктів
radius = 0.0003; % приблизно 30 метрів в географічних координатах
circle_points = 36; % кількість точок для кругового польоту

```

```

% Додавання кругових польотів навколо об'єктів
for j = 1:circle_points
    angle = 2 * pi * j / circle_points;
    x = obj(1) + radius * cos(angle);
    y = obj(2) + radius * sin(angle);
    uav_flight_path = [uav_flight_path; x, y, altitude];
end

% Hover for 1 minute
hover_time = 10 / 60; % час зависання в хвиликах (10 секунд = 10/60 хвилини)
desired_rate = 6; % так як ми хочемо зберігати точку кожні 10 секунд, то це буде 6 разів на хвилину
hover_points = round(hover_time * desired_rate); % визначаємо кількість точок для збереження
for j = 1:hover_points
    uav_flight_path = [uav_flight_path; uav_flight_path(end,1), uav_flight_path(end,2), altitude];
end
end

% Додавання кінцевої точки польоту (Landing to 0 altitude after reaching the final latitude and longitude)
uav_flight_path = [uav_flight_path; flight_endPoint(1), flight_endPoint(2), desired_altitude]; %вихід на
координату посадки
uav_flight_path = [uav_flight_path; flight_endPoint]; % посадка на визначену точку

%% Отримання відміток висоти
if exist('data/coordinates.mat', 'file') % Перевірка на існування файлу coordinates
    load('data/coordinates.mat'); % Завантаження файлу coordinates
    fprintf('Файл завантажено успішно: coordinates.mat \n'); % Виведення сповіщення про завантаження
    api_request_count = 0; % Ініціалізація лічильника запитів до API
else
    coordinates = []; %створюємо пусту змінну
end
if isequal(size(uav_flight_path), size(coordinates)) %Порівняти розмірність двох матриць
    disp('Розмірність матриць однакова. Дані завантажено із сховища.');
```

```

else
    disp('Розмірність матриць різна. Дані отримано по Google API.');
```

```

% Якщо файлу немає, виконуємо запит по API
if exist('data/api_key.mat', 'file')
    load('data/api_key.mat'); %get Google maps API key
    fprintf('Файл Google maps API key завантажено успішно (api_key.mat) \n'); % Виведення сповіщення
про завантаження
end
end

```

```

% Перевірка чи має flight_path лише дві колонки, і якщо так, то додає третю колонку з нулями
if size(uav_flight_path, 2) == 2
    uav_flight_path(:, 3) = zeros(size(uav_flight_path, 1), 1);
end

% Перевірка на існування coordinates
if ~exist('coordinates', 'var') || size(coordinates, 1) ~= size(uav_flight_path, 1)
    coordinates = uav_flight_path(:, 1:2); % Копіюємо перші дві колонки з flight_path
    coordinates(:, 3) = 0; % Додаємо третю колонку з нулями
end

% Проходимося по координатам та отримуємо висоту
for i = 1:size(coordinates, 1)
    latitude = coordinates(i, 1);
    longitude = coordinates(i, 2);
    existing_elevation = coordinates(i, 3);
    [elevation, api_request_count] = get_elevation_API(latitude, longitude, api_key, existing_elevation,
api_request_count); % Запит до Google Maps Elevation API для отримання висоти на основі вказаних координат
(або завантаження з файлу)
    coordinates(i, 3) = elevation; % Оновлюємо значення висоти в матриці elevation
end

% Збереження результатів значень висоти в файл
save('data/coordinates.mat', 'coordinates');
fprintf('Нові значення висот збережено в файл: coordinates.mat \n'); % Виведення сповіщення про
завантаження
end

% Виведення кількості виконаних запитів
fprintf('Кількість виконаних запитів до API: %d\n', api_request_count);
fprintf('\n');

% Виводимо оновлену матрицю
disp(coordinates);

%% Коригування маршруту польоту (з урахуванням отриманої карти висот)
% Коригування маршруту польоту (з урахуванням отриманої карти висот)
% Визначення мінімальної та максимальної висоти польоту
min_altitude = flight_altitude(1);
max_altitude = flight_altitude(2);
flight_path_updated = uav_flight_path;
for i = 1:size(coordinates, 1)
    ground_elevation = coordinates(i, 3); % висота поверхні над рівнем моря
    current_altitude = uav_flight_path(i, 3); % поточна висота польоту БпЛА

```

```

% Для початкової та кінцевої точки
if i == 1 || i == size(coordinates, 1)
    flight_path_updated(i, 3) = ground_elevation;
    continue;
end

% Визначення нової висоти польоту
new_altitude = ground_elevation + current_altitude;

% Перевірка, чи нова висота відповідає вимогам flight_altitude
if new_altitude - ground_elevation < min_altitude
    new_altitude = ground_elevation + min_altitude;
elseif new_altitude - ground_elevation > max_altitude
    new_altitude = ground_elevation + max_altitude;
end

% Оновлення висоти польоту в маршруті
flight_path_updated(i, 3) = new_altitude;
end

fprintf('Оновлено план польоту із урахуванням карти висот місцевості (flight_path_updated.mat) \n'); %
Виведення сповіщення про завантаження

% Збереження результатів значень висоти в файл
save('data/flight_path_updated.mat', 'flight_path_updated');
fprintf('Оновлений план польоту збережено в файл: flight_path_updated.mat \n'); % Виведення сповіщення
про завантаження

%% Відображення оновленого маршруту польоту
% Просторова модель маршруту (3D)
figure;
hold on;
plot3(flight_path_updated(:,1), flight_path_updated(:,2), flight_path_updated(:,3), '-o', 'DisplayName',
'Updated Flight Path');
plot3(flight_path_updated(:,1), flight_path_updated(:,2), coordinates(:,3), '-r', 'DisplayName', 'Ground
Elevation');
xlabel('Latitude');
ylabel('Longitude');
zlabel('Altitude (m)');
title('Оновлений план польоту із урахуванням карти місцевості (3D)');
grid on;
legend;
view(3); % примусове відображення в 3D
hold off;

```

```

% Модель маршруту на поверхні (2D + карта)
figure; % створення нового вікна для графіка оновленого маршруту
geoplot(flight_path_updated(:,1), flight_path_updated(:,2), 'b', 'LineWidth', 1.5); % Червоним кольором
title('Оновлений план польоту із урахуванням карти місцевості (2D)');
legend('Скорегований маршрут БпЛА');
grid on;
%% Відображення даних на веб-карті OpenStreetMap (2D)
VisualizationOnMap(flight_path_updated, flight_startPoint, flight_endPoint); % вікно веб-карти з маршрутом
пльоту та початковою та кінцевою точками.
%% Розрахунок абсолютної дистанції руху БпЛА по маршруту
% Визначаємо кількість точок у нашому маршруті.
num_points = size(flight_path_updated, 1);
% Ініціалізуємо масив для зберігання накопичених дистанцій для кожної точки маршруту.
cumulative_distances = zeros(num_points, 1);
% Проходимося через всі точки маршруту, починаючи з другої точки.
for i = 2:num_points
    % Перетворюємо градуси у радіани.
    lat1 = deg2rad(flight_path_updated(i-1, 1));
    lon1 = deg2rad(flight_path_updated(i-1, 2));
    lat2 = deg2rad(flight_path_updated(i, 1));
    lon2 = deg2rad(flight_path_updated(i, 2));
    % Обчислюємо різницю у координатах між сусідніми точками.
    dlat = lat2 - lat1;
    dlon = lon2 - lon1;

    % Використовуємо формулу гаверсинусів для знаходження відстані між двома точками на поверхні
    кулі.
    a = sin(dlat/2)^2 + cos(lat1) * cos(lat2) * sin(dlon/2)^2;
    c = 2 * atan2(sqrt(a), sqrt(1-a));

    % Обчислюємо відстань між двома точками.
    distance = R_earth * c;

    % Додаємо обчислену відстань до загальної відстані.
    cumulative_distances(i) = cumulative_distances(i-1) + distance;
end
% Зараз у змінній cumulative_distances зберігається дистанція в кілометрах від початкової точки до
кожної іншої точки маршруту.
%% Відображення графіків різниці між початковим маршрутом та скорегованим
% Побудова графіків
figure;

```

```

% Графік зміни для координати Z
subplot(1, 1, 1); % 1 рядок, 2 стовпці, 1-й графік
plot(cumulative_distances, flight_path_updated(:,3), 'blue', 'LineWidth', 1.5); % різниця в зображенні з синім
кольором
hold on;
plot(cumulative_distances, coordinates(:,3), 'g', 'LineWidth', 1.5); % висота з зеленим кольором
plot(cumulative_distances(1), flight_path_updated(1,3), 'ro', 'MarkerSize', 8, 'LineWidth', 2); % Початкова
точка
plot(cumulative_distances(end), flight_path_updated(end,3), 'ro', 'MarkerSize', 8, 'LineWidth', 2); % Кінцева
точка
xlabel('Дистанція (км)');
ylabel('Висота над рівнем моря (м)');
title('Висота польоту над рівнем моря');
legend('Висота польоту БПЛА', 'Рельєф поверхні', 'Початкова та кінцева точки місії', 'Location',
'southeast');
ylim([0 max(flight_path_updated(:,3)) + 10]); % встановлення діапазону осі Y від 0 до максимального
значення + 10
grid on;
hold off; % відключення режиму утримання для наступних графіків
% Графік дельта для координати Z
figure;
subplot(1, 1, 1); % 1 рядок, 2 стовпці, 2-й графік
dif_flight_path = flight_path_updated - coordinates; % Висота польоту БПЛА над землею
plot(cumulative_distances, dif_flight_path(:,3), 'blue', 'LineWidth', 1.5);
hold on;
plot(cumulative_distances(1), dif_flight_path(1,3), 'ro', 'MarkerSize', 8, 'LineWidth', 2); % Початкова точка
plot(cumulative_distances(end), dif_flight_path(end,3), 'ro', 'MarkerSize', 8, 'LineWidth', 2); % Кінцева точка
xlabel('Дистанція (км)');
ylabel('Висота над поверхнею (м)');
title('Висота польоту БПЛА над землею');
legend('Висота польоту БПЛА', 'Початкова та кінцева точки місії', 'Location', 'northwest');
ylim([0 max(dif_flight_path(:,3)) + 10]); % встановлення діапазону осі Y від 0 до максимального значення
+ 10
grid on;
hold off; % відключення режиму утримання для наступних графіків
%% Перехід до спрощеної просторової трьохвимірної моделі (декартових координат)
% Константи для перетворення градусів в метри
LAT_TO_METERS = 111e3; % 111 км в метрах для широти
LON_TO_METERS = 111e3 * cos(mean(flight_path_updated(:,2)) * pi / 180); % довгота залежить від
широти

```

```

% Відніmemo координати початкової точки
ref_point = flight_path_updated(1, :);
% Переводимо географічні координати в декартові координати
flight_path_cartesian = [(flight_path_updated(:,2) - ref_point(2)) * LON_TO_METERS, ...
                        (flight_path_updated(:,1) - ref_point(1)) * LAT_TO_METERS, ...
                        flight_path_updated(:,3)];
% Побудова трьохмірного графіка
figure;
plot3(flight_path_cartesian(:,1), flight_path_cartesian(:,2), flight_path_cartesian(:,3), 'b-');
xlabel('X (метри)');
ylabel('Y (метри)');
zlabel('Висота над рівнем моря (м)');
title('Траскторія польоту БПЛА в декартових координатах (3D)');
grid on;

%% ФУНКЦІЇ ДЛЯ МОДЕЛЮВАННЯ ПЛАНУ ПОЛЬОТУ
%% Візуалізація карти польоту на OpenStreetMap
% Файл: VisualizationOnMap.m
%%
function VisualizationOnMap(flight_path_updated, flight_startPoint, flight_endPoint)
    % Відкриття вікна веб-карти OpenStreetMap
    wm = webmap('OpenStreetMap');

    % Зосередження карти на території Національного авіаційного університету із радіусом 10 км
    latlim = [flight_startPoint(1,1) - 0.09, flight_startPoint(1,1) + 0.09]; % приблизно 10 км в південному та
    північному напрямках
    lonlim = [flight_startPoint(1,2) - 0.15, flight_startPoint(1,2) + 0.15]; % приблизно 10 км на захід та схід
    wmlimits(latlim, lonlim);
    % Відображення маршруту польоту на карті
    wmline(flight_path_updated(:,1), flight_path_updated(:,2), 'Color', 'blue', 'Width', 2);
    % Додавання маркерів для початкової та кінцевої точок
    wmmarker(flight_startPoint(1,1), flight_startPoint(1,2), 'Color', 'g', 'FeatureName', 'Start Point');
    wmmarker(flight_endPoint(1,1), flight_endPoint(1,2), 'Color', 'r', 'FeatureName', 'End Point');
end

%% Запит карти висот плану польотів до Google Maps Elevation API
% Файл: get_elevation_API.m
function [elevation, api_request_count] = get_elevation_API(latitude, longitude, api_key, existing_elevation,
api_request_count)
    fprintf('Створено запит до Google Maps Elevation API \n'); % Виведення сповіщення про відпрацювання
    % Якщо вже є дійсне значення висоти, повертаємо його
    if existing_elevation ~= 0

```



```

    elevation = existing_elevation;
    return;
end

% Створюємо URL для Google Maps Elevation API
url = sprintf('https://maps.googleapis.com/maps/api/elevation/json?locations=%f,%f&key=%s', latitude,
longitude, api_key);

% Робимо запит до веб-сервісу
response = webread(url);

% Перевіряємо статус відповіді
if strcmp(response.status, 'OK')
    elevation = response.results(1).elevation;

    api_request_count = api_request_count + 1; % Збільшуємо лічильник запитів API
else
    error('Не вдалося отримати висоту: %s', response.status);
end
end

%% Створення 3D моделі переміщення БПЛА в декартових координатах
%% Розрахунок матриці маршруту з абсолютною відстанню не більше 1 метра
% Даний скрипт створює змінні які описують лінійний шлях переміщення БПЛА
% Файл: uav_flight_path_calculation.m
% Початкові параметри:
data = load('data/flight_path_updated.mat'); % Зчитування файлу з даними
geo_path = data.flight_path_updated; % Зберігаємо дані в geo_path
fprintf('Файл з даними зчитано: flight_path_updated.mat \n'); % Виведення сповіщення про завантаження
% Конвертація географічних координат у декартові
% Ініціалізація нової матриці маршруту
R_curv = zeros(size(geo_path(:, 3)));
R_curv = R_earth ./ sqrt(1 - e2_earth * sind(geo_path(:, 1)).^2); % радіус кривизни перпендикулярний до
поверхні в даній точці
% Конвертація географічних координат у декартові
path = zeros(size(geo_path));
path(:, 1) = (R_curv + geo_path(:, 3)) .* cosd(geo_path(:, 1)) .* cosd(geo_path(:, 2)); % x
path(:, 2) = (R_curv + geo_path(:, 3)) .* cosd(geo_path(:, 1)) .* sind(geo_path(:, 2)); % y
%path(:, 3) = (R_curv .* (1 - e2_earth) + geo_path(:, 3)) .* sind(geo_path(:, 1)); % z
path(:, 3) = geo_path(:, 3); % z балансування моделі до рівня моря

```

```

% Ініціалізація нової матриці маршруту
refined_path = [];
for i = 1:size(path, 1)-1
    current_point = path(i, :);
    next_point = path(i+1, :);

    % Якщо поточна точка співпадає з наступною, додайте точки зависання
    if isequal(current_point, next_point)
        for j = 1:time_objectives
            refined_path = [refined_path; current_point];
        end
    else
        % Обчислення відстані між точками
        distance = norm(next_point - current_point);
        % Визначення кількості додаткових точок
        num_extra_points = ceil(distance) - 1;
        % Додавання точок до refined_path
        for j = 0:num_extra_points
            new_point = current_point + (next_point - current_point) * (j / (num_extra_points + 1));
            refined_path = [refined_path; new_point];
        end
    end
end
% Додавання останньої точки
refined_path = [refined_path; path(end, :)];
fprintf('Маршрут було оптимізовано з відстанню між точками не більше 1 метра \n');
% Збереження результатів
save('data/refined_path.mat', 'refined_path');
fprintf('Файл з даними опраціоналізованого маршруту для БпЛА створено: refined_path.mat \n'); %
Виведення сповіщення про збереження

%% Візуалізація оптимізованого маршруту у 3D
% Файл: visualize_path.m
% Виведення 3D маршруту з кроком крок 1 метр
figure; % Створення нового вікна для графіка
plot3(refined_path(:, 1), refined_path(:, 2), (refined_path(:, 3)), '-o');
xlabel('X'); % Підпис осі X
ylabel('Y'); % Підпис осі Y
zlabel('Z'); % Підпис осі Z
title('Оптимізований маршрут у 3D (абсолютна висота)'); % Заголовок графіка
grid on; % Включення сітки

```

```

% Виведення 3D маршруту з кроком крок 1 метр
figure; % Створення нового вікна для графіка
plot3(refined_path(:, 1)-refined_path(1, 1), refined_path(:, 2)-refined_path(1, 2), refined_path(:, 3)-
refined_path(1,3), '-r');
xlabel('X, м'); % Підпис осі X
ylabel('Y, м'); % Підпис осі Y
zlabel('Висота'); % Підпис осі Z
title('Оптимізований маршрут у 3D (відносно точки вильоту)'); % Заголовок графіка
grid on; % Включення сітки
%% Зберігаємо відстані між кожною парою послідовних точок маршруту
% Ініціалізація масиву для зберігання відстаней між точками
num_points = size(refined_path, 1); % Кількість точок у refined_path
distances_between_points = zeros(num_points, 1);
% Ітерація через усі точки маршруту
for i = 1:num_points-1
    delta_x = refined_path(i+1, 1) - refined_path(i, 1);
    delta_y = refined_path(i+1, 2) - refined_path(i, 2);
    delta_z = refined_path(i+1, 3) - refined_path(i, 3);

    % Розрахунок відстані між поточною та наступною точками
    distance = sqrt(delta_x^2 + delta_y^2 + delta_z^2);
    distances_between_points(i) = distance;
end
num_segments = num_points - 1; % Кількість відрізків між точками
fprintf('Кількість вимірних відрізків маршруту: %d\n', num_segments);
% Збереження результатів
save('data/distances_between_points.mat', 'distances_between_points');
fprintf('Файл з даними про відстані між точками створено: distances_between_points.mat \n'); % Виведення
сповіщення
%% Код для розрахунку загальної відстані, яку подолає БпЛА:
uav_total_distance = 0; % Ініціалізація змінної для зберігання загальної відстані
% Ітерація через усі точки маршруту для розрахунку відстані між ними
for i = 1:size(refined_path, 1)-1
    delta_x = refined_path(i+1, 1) - refined_path(i, 1); % Розрахунок різниці координат X між поточною та
наступною точками
    delta_y = refined_path(i+1, 2) - refined_path(i, 2); % Розрахунок різниці координат Y
    delta_z = refined_path(i+1, 3) - refined_path(i, 3); % Розрахунок різниці координат Z

    % Розрахунок відстані між поточною та наступною точками
    distance = sqrt(delta_x^2 + delta_y^2 + delta_z^2); % Розрахунок відстані між поточною та наступною
точками

```

```

    uav_total_distance = uav_total_distance + distance; % Додавання відстані до загальної відстані
end
uav_total_distance_rounded = round(uav_total_distance, 2); % Округлення загальної відстані до 2 знаків
після коми
% Збереження результатів
save('data/uav_total_distance.mat', 'uav_total_distance');
fprintf('Файл з даними пройдений шлях БПЛА створено: uav_total_distance.mat \n'); % Виведення
сповіщення про збереження
fprintf('Загальна відстань, яку подолає БПЛА: %f метрів\n', uav_total_distance_rounded); % Виведення
розрахованої відстані
%% Ініціалізація початкової точки відліку
% Ініціалізуємо новий масив для збереження оновлених координат
refined_path_0 = zeros(size(refined_path));
% Віднімаємо першу точку від усіх інших точок
for i = 1:size(refined_path, 1)
    refined_path_0(i,:) = refined_path(i,:) - refined_path(1,:); % Тепер refined_path_0 містить оновлені
координати, де перша точка є (0, 0, 0)
end
%% Обчислення команд керування БПЛА під час місії
%% Код для розрахунку дельта кутових відхилень і відстані між точками маршруту БПЛА:
% Файл: uav_flight_commands.mlx
refined_path = refined_path_0; %(тест)
% Ініціалізація масивів для зберігання дельта кутових відхилень і відстаней
delta_angles = zeros(size(refined_path, 1) - 1, 3); % [ $\Delta\alpha$ ,  $\Delta\beta$ ,  $\Delta\gamma$ ]
distances = zeros(size(refined_path, 1) - 1, 1);
% Визначення кількості точок у refined_path
num_points = size(refined_path, 1);
% Ітерація через усі точки маршруту для розрахунку дельта кутових відхилень і відстані між ними
for i = 1:num_points-1
    % Розрахунок вектору напрямку між поточною та наступною точками
    direction_vector = refined_path(i+1, :) - refined_path(i, :);
    % Розрахунок довжини вектору напрямку
    magnitude = norm(direction_vector);
    % Перевірка на нульову магнітуду, щоб уникнути ділення на нуль
    if magnitude == 0
        direction_vector_angles(i, :) = [0, 0, 0];
    else
        % Розрахунок дельта кутових відхилень
        alpha_angle = acos(direction_vector(1) / magnitude); %  $\Delta\alpha$ 
        beta_angle = acos(direction_vector(2) / magnitude); %  $\Delta\beta$ 
        gamma_angle = acos(direction_vector(3) / magnitude); %  $\Delta\gamma$ 
    end
end

```

```

% Заміна NaN на 0
if isnan(alpha_angle)
    direction_vector_angles(i, 1) = 0;
else
    direction_vector_angles(i, 1) = alpha_angle;
end
if isnan(beta_angle)
    direction_vector_angles(i, 2) = 0;
else
    direction_vector_angles(i, 2) = beta_angle;
end
if isnan(gamma_angle)
    direction_vector_angles(i, 3) = 0;
else
    direction_vector_angles(i, 3) = gamma_angle;
end
end

% Розрахунок відстані між поточною та наступною точками
distances(i) = magnitude;
end

% Кутові відхилення у градусах
direction_vector_deg = rad2deg(direction_vector_angles);
% Графік дельта куткових відхилень
figure;
subplot(3, 1, 1);
plot(1:num_points-1, direction_vector_deg(:, 1), 'b', 'LineWidth', 1.5);
title('Дельта куткових відхилень:  $\Delta\alpha$ ');
xlabel('Номер точки');
ylabel('Delta (градуси)');
subplot(3, 1, 2);
plot(1:num_points-1, direction_vector_deg(:, 2), 'b', 'LineWidth', 1.5);
title('Дельта куткових відхилень:  $\Delta\beta$ ');
xlabel('Номер точки');
ylabel('Delta (градуси)');
subplot(3, 1, 3);
plot(1:num_points-1, direction_vector_deg(:, 3), 'b', 'LineWidth', 1.5);
title('Дельта куткових відхилень:  $\Delta\gamma$ ');
xlabel('Номер точки');
ylabel('Delta (градуси)');
% Графік відстаней між точками
figure;

```

```

plot(1:num_points-1, distances, '-o', 'LineWidth', 1.5);
title('Відстань між точками');
xlabel('Номер точки');
ylabel('Відстань (м)');

%% Код для розрахунку очікуваної динаміки БпЛА
%% Код для розрахунку динаміки БпЛА
% Файл: uav_dynamics.mlx
% Ініціалізація станів
uav_omega = [0; 0; 0]; % Кутова швидкість (припускаємо, що на початку вона дорівнює 0)
uav_tau = [0; 0; 0]; % Кутовий момент (момент, який прикладається до БпЛА)
uav_r = refined_path(1, :); % Встановлення початкової точки маршруту
uav_v = [0; 0; 0]; % Початкові значення швидкостей
target_idx = 1; % Індекс поточної цільової точки (мін, в метрах)
% Масиви для зберігання результатів симуляції
omega_history = zeros(3, length(t));
r_history = zeros(3, length(t));
v_history = zeros(3, length(t));
a_history = zeros(3, length(t));
%%
% Симуляція
for i = 1:length(t)
    % Визначення наступної цільової точки
    if norm(refined_path(target_idx, :) - uav_r) <= 1 && target_idx < size(refined_path, 1)
        target_idx = target_idx + 1;
    end
    target_point = refined_path(target_idx, :);

    % Керування
    desired_v = (target_point - uav_r) / dt;
    acceleration = (desired_v - uav_v) / dt;
    f_total = uav_m * (g + acceleration);
    % Розрахунок аеродинамічного опору
    F_drag = -0.5 * uav_rho * uav_A * uav_Cd * uav_v .* norm(uav_v);
    % Розрахунок підхідної від положення та швидкості
    uav_a = f_total/uav_m - g + F_drag/uav_m;
    v_dot = uav_a;

    % Розрахунок прохідної від кутової швидкості
    omega_dot = inv(uav_J) * (uav_tau - cross(uav_omega, uav_J * uav_omega));

```

```

% Оновлення станів
uav_omega = uav_omega + omega_dot * dt;
uav_v = uav_v + v_dot * dt;
uav_r = uav_r + uav_v * dt;

% Зберігаємо поточний стан
omega_history(:, i) = uav_omega;
a_history(:, i) = uav_a;
v_history(:, i) = uav_v;
r_history(:, i) = uav_r;
end

%% Графіки результатів динаміки польоту
% Кутова швидкість
figure;
subplot(3, 1, 1);
plot(t, omega_history(1, :));
title('Кутова швидкість по осі X');
xlabel('Час (с)');
ylabel('ωx (rad/s)');
subplot(3, 1, 2);
plot(t, omega_history(2, :));
title('Кутова швидкість по осі Y');
xlabel('Час (с)');
ylabel('ωy (rad/s)');
subplot(3, 1, 3);
plot(t, omega_history(3, :));
title('Кутова швидкість по осі Z');
xlabel('Час (с)');
ylabel('ωz (rad/s)');

% Зміна прискорення
figure;
subplot(3, 1, 1);
plot(t, a_history(1, :));
title('Прискорення по осі X');
xlabel('Час (с)');
ylabel('x (м/с²)');
subplot(3, 1, 2);
plot(t, a_history(2, :));
title('Прискорення по осі Y');
xlabel('Час (с)');

```

```

ylabel('y (м/с2)');
subplot(3, 1, 3);
plot(t, a_history(3, :));
title('Прискорення по осі Z');
xlabel('Час (с)');
ylabel('z (м/с2)');
% Зміна швидкості
figure;
subplot(3, 1, 1);
plot(t, v_history(1, :));
title('Швидкість по осі X');
xlabel('Час (с)');
ylabel('x (м/с)');
subplot(3, 1, 2);
plot(t, v_history(2, :));
title('Швидкість по осі Y');
xlabel('Час (с)');
ylabel('y (м/с)');
subplot(3, 1, 3);
plot(t, v_history(3, :));
title('Швидкість по осі Z');
xlabel('Час (с)');
ylabel('z (м/с)');
% Зміна координати
figure;
subplot(3, 1, 1);
plot(t, r_history(1, :));
title('Положення по осі X');
xlabel('Час (с)');
ylabel('x (м)');
subplot(3, 1, 2);
plot(t, r_history(2, :));
title('Положення по осі Y');
xlabel('Час (с)');
ylabel('y (м)');
subplot(3, 1, 3);
plot(t, r_history(3, :));
title('Положення по осі Z');
xlabel('Час (с)');
ylabel('z (м)');

```



```

%% Моделювання бортової МЕМС системи
% Файл: uav_mems_6.mlx
% Параметри похибок для акселерометра
accelerometer_params.bias = 0.01; % Похибка зсуву
accelerometer_params.bias_drift_rate = 1e-5; % Швидкість зсуву залежна від часу
accelerometer_params.scale_factor = 0.005; % Похибка масштабного коефіцієнта
accelerometer_params.non_orthogonality = 0.005; % Похибка неортогональності
accelerometer_params.drift_rate = 1e-4; % Швидкість дрейфу залежна від часу
accelerometer_params.noise_level = 0.01; % Рівень шуму
% Параметри похибок для гіроскопа
gyroscope_params.bias = 0.01; % Похибка зсуву
gyroscope_params.bias_drift_rate = 1e-5; % Швидкість зсуву залежна від часу
gyroscope_params.scale_factor = 0.005; % Похибка масштабного коефіцієнта
gyroscope_params.non_orthogonality = 0.005; % Похибка неортогональності
gyroscope_params.drift_rate = 1e-4; % Швидкість дрейфу залежна від часу
gyroscope_params.noise_level = 0.01; % Рівень шуму
% Ініціалізація мемс системи
calibration_time = 10; % Час калібрування у секундах
uav_v_0 = [0,0,0]; % Початкова швидкість БПЛА
uav_coord_0 = [0,0,0];
uav_azimuth = current_azimuth;

%% СТВОРЕННЯ МОДЕЛІ ВИМІРЯНИХ ЗНАЧЕНЬ МЕМС
% Створення похибок для акселерометра і гіроскопа
[accelerometer_errors, gyroscope_errors] = simulate_sensors_errors(t, accelerometer_params,
gyroscope_params);
% Ініціалізація ідеальних вимірювань
ideal_accelerations = a_history; % Ідеальні прискорення
ideal_angular_rates = omega_history; % Ідеальні кутові швидкості
% Додавання похибок до ідеальних вимірювань
measured_accelerations = ideal_accelerations + accelerometer_errors;
measured_gyro_rates = ideal_angular_rates + gyroscope_errors;
% Застосування похибок до швидкості та координат
% Встановлення початкових значень
measured_velocity(:,1) = uav_v_0;
measured_position(:,1) = uav_coord_0;
% Інтегрування прискорень для отримання швидкості та переміщення
for i = 2:length(t)
    measured_velocity(:,i) = measured_velocity(:,i-1) + measured_accelerations(:,i) * dt;
    measured_position(:,i) = measured_position(:,i-1) + measured_velocity(:,i) * dt;
end

```

```

% Відображення графіків прискорень
figure;
subplot(3,1,1);
plot(t, a_history(1,:), 'r', t, measured_accelerations(1,:), 'b');
title('Прискорення по осі X');
xlabel('Час, с');
ylabel('Прискорення, м/с^2');
legend('Ідеальне', 'Виміряне');
subplot(3,1,2);
plot(t, a_history(2,:), 'r', t, measured_accelerations(2,:), 'b');
title('Прискорення по осі Y');
xlabel('Час, с');
ylabel('Прискорення, м/с^2');
legend('Ідеальне', 'Виміряне');
subplot(3,1,3);
plot(t, a_history(3,:), 'r', t, measured_accelerations(3,:), 'b');
title('Прискорення по осі Z');
xlabel('Час, с');
ylabel('Прискорення, м/с^2');
legend('Ідеальне', 'Виміряне');
% Відображення графіків швидкостей
figure;
subplot(3,1,1);
plot(t, v_history(1,:), 'r', t, measured_velocity(1,:), 'b');
title('Швидкість по осі X');
xlabel('Час, с');
ylabel('Швидкість, м/с');
subplot(3,1,2);
plot(t, v_history(2,:), 'r', t, measured_velocity(2,:), 'b');
title('Швидкість по осі Y');
xlabel('Час, с');
ylabel('Швидкість, м/с');
subplot(3,1,3);
plot(t, v_history(3,:), 'r', t, measured_velocity(3,:), 'b');
title('Швидкість по осі Z');
xlabel('Час, с');
ylabel('Швидкість, м/с');
% Відображення графіків координат
figure;
subplot(3,1,1);
plot(t, r_history(1,:), 'r', t, measured_position(1,:), 'b');

```

```

title('Координата X');
xlabel('Час, с');
ylabel('Координата, м');
subplot(3,1,2);
plot(t, r_history(2,:),'r',t, measured_position(2,:), 'b');
title('Координата Y');
xlabel('Час, с');
ylabel('Координата, м');
subplot(3,1,3);
plot(t, r_history(3,:),'r',t, measured_position(3,:), 'b');
title('Координата Z');
xlabel('Час, с');
ylabel('Координата, м');

%% Алгоритм моделювання магнітометра
%% Параметри похибок для магнітометра
magnetometer_params.deviation_error = 0.01; % похибка відхилення
magnetometer_params.inertia_error = 0.02; % похибка інерції
%% Генерація похибок для магнітометра
magnetometer_errors = generate_magnetometer_errors(t, magnetometer_params);
%% Інтеграція кутових швидкостей для отримання кутів орієнтації
angles = cumsum(omega_history, 2) * dt; % Припускаємо, що omega_history – це матриця [3 x N] із
кутовими швидкостями по осях X, Y, Z
%% Перетворення кутів орієнтації на вектор магнітного поля
%% Припущення: магнітне поле Землі є однорідним і спрямованим на північ.
earth_magnetic_field = [1; 0; 0]; % Спрощене припущення: магнітне поле Землі спрямоване на північ
heading_angle = angles(1, :); % Курс БпЛА
mag_field_measured = earth_magnetic_field .* cos(heading_angle) + ...
    cross(earth_magnetic_field, [0; 0; 1]) .* sin(heading_angle) + ...
    magnetometer_errors; % Перетворення з урахуванням курсу та похибок
%% Розрахунок ідеального курсу з кутів орієнтації
ideal_course = mod(atan2(sin(angles(1, :)), cos(angles(1, :))), 2*pi);
%% Розрахунок виміряного курсу з даних магнітометра
measured_course = mod(atan2(mag_field_measured(2, :), mag_field_measured(1, :)), 2*pi);
%% Припустимо, що measured_course – це вектор з кутами в радіанах
measured_course_degrees = rad2deg(measured_course); % Перетворення в градуси
%% Унормування кутів для відображення від -180 до 180 градусів
normalized_course = mod(measured_course_degrees + 180, 360) – 180;
%% Відображення унормованих кутів
figure;
plot(t, normalized_course, 'b');

```

```

hold on; % Утримувати поточний графік, щоб додати наступну лінію
plot(t, rad2deg(ideal_course), 'r', 'LineWidth', 2); % Зробити червону лінію товстішою
hold off; % Відпустити графік
xlabel('Час (с)');
ylabel('Курс (градуси)');
title('Унормований курс');
legend('Ідеальний курс', 'Вимірний курс'); % Додати легенду для зручності
grid on; % Увімкнути сітку на графіку

%% СТВОРЕННЯ АЛГОРИТМУ ФІЛЬТРАЦІЇ МАГНІТОМЕТРА
% Параметри фільтра ЕМА
alpha = 0.01; % Коефіцієнт згладжування для ЕМА
N = length(t); % Кількість вимірювань
% Ініціалізація фільтрованих даних
filtered_course = zeros(1, N);
% Використання ЕМА для фільтрації курсу
filtered_course(1) = normalized_course(1); % Початкове значення для ЕМА
for i = 2:N
    filtered_course(i) = alpha * normalized_course(i) + (1 - alpha) * filtered_course(i-1);
end
mag_data = filtered_course; % внесення інформації магнітометра для передачі до обчислювача БпЛА
%% RMS системи
% Помилки курсу між ідеальним курсом та фільтрованим курсом
course_errors = rad2deg(ideal_course) - filtered_course;
% Розрахунок RMS для курсу
rms_course = sqrt(mean(course_errors.^2));
% Вивід значення RMS помилки курсу
fprintf('RMS похибка курсу: %f градусів\n', rms_course);
%% Графік системи
% Відображення унормованих та фільтрованих кутів курсу
figure;
hold on;
plot(t, normalized_course, 'b'); % Вимірний курс
plot(t, filtered_course, 'g'); % Фільтрований курс
plot(t, rad2deg(ideal_course), 'r', 'LineWidth', 3); % Ідеальний курс
hold off;
xlabel('Час (с)');
ylabel('Курс (градуси)');
title('Фільтрація курсу магнітометра');
legend('Вимірний курс', 'Фільтрований курс', 'Ідеальний курс');
grid on;

```

```

%% СТВОРЕННЯ АЛГОРИТМУ КАЛІБРУВАННЯ МЕМС
%% Ініціалізація параметрів для калібрування
calib_mems_idx_calibration_time = 60; % час калібрування в секундах
F_cal = 100; % Під час калібрування здійснюємо отримання даних із МЕМС гіроскопа із частотою 100
Гц
d_cal = 1/(dt*100);
t_cal = 0:d_cal:calib_mems_idx_calibration_time; % вектор часу для калібрування з кроком в 0.01 секунду
(100 Гц)
%% Симуляція похибок датчиків
[mems_accelerometer_errors, mems_gyroscope_errors] = simulate_sensors_errors_cal(t_cal,
accelerometer_params, gyroscope_params);
%% Розрахунок постійної складової похибок (bias)
calib_mems_idx_accelerometer_bias = mean(mems_accelerometer_errors, 2); % середнє значення похибок
акселерометра
calib_mems_idx_gyroscope_bias = mean(mems_gyroscope_errors, 2); % середнє значення похибок
гіроскопа
% Для кожної вісі окремо розраховуємо коефіцієнт дрейфу (drift rate)
for i = 1:3
    p = polyfit(t_cal, mems_accelerometer_errors(i, :), 1);
    calib_mems_idx_accelerometer_drift_rate(i) = p(1); % коефіцієнт нахилу для акселерометра
    p = polyfit(t_cal, mems_gyroscope_errors(i, :), 1);
    calib_mems_idx_gyroscope_drift_rate(i) = p(1); % коефіцієнт нахилу для гіроскопа
end
% Корекція прискорень measured_gyro_rates
%% Корекція прискорень
calibrated_accelerations = measured_accelerations ...
repmat(calib_mems_idx_accelerometer_bias, 1, size(measured_accelerations, 2)) ...
bsxfun(@times, calib_mems_idx_accelerometer_drift_rate', t);
%% Корекція гіроскопічних вимірювань (кутових швидкостей)
calibrated_gyro_measurements = measured_gyro_rates ...
repmat(calib_mems_idx_gyroscope_bias, 1, size(measured_gyro_rates, 2)) ...
bsxfun(@times, calib_mems_idx_gyroscope_drift_rate', t);
%% Інтегрування отриманих результатів для отримання фактичних значень вимірювання
(використовуючи відкалібровану систему)
% Застосування каліброваних даних для швидкості та координат
calibrated_velocity(:,1) = uav_v_0;
calibrated_position(:,1) = uav_coord_0;
% Інтегрування каліброваних прискорень для отримання швидкості та переміщення
for i = 2:length(t)
    calibrated_velocity(:,i) = calibrated_velocity(:,i-1) + calibrated_accelerations(:,i) * dt;
    calibrated_position(:,i) = calibrated_position(:,i-1) + calibrated_velocity(:,i) * dt;

```

```

end
% записуємо виміряні дані в визначений масив для передачі до обчислювача БпЛА
mems_data = calibrated_position;
mems_v = calibrated_velocity;
mems_a = calibrated_accelerations;
mems_giro = calibrated_gyro_measurements;
%% Аналіз результатів
%% RMS системи
% Помилки вимірювань
acceleration_errors = calibrated_accelerations - ideal_accelerations;
velocity_errors = calibrated_velocity - sqrt(sum(ideal_angular_rates.^2, 1)); % припускаючи, що
ideal_angular_rates містить ідеальні швидкості
angular_rate_errors = calibrated_gyro_measurements - ideal_angular_rates;
position_errors = calibrated_position - r_history; % припускаючи, що ідеальні координати містяться в
змінній ideal_position
% Розрахунок RMS для кожного типу даних
rms_acceleration = sqrt(mean(sum(acceleration_errors.^2, 1)));
rms_velocity = sqrt(mean(sum(velocity_errors.^2, 1)));
rms_angular_rate = sqrt(mean(sum(angular_rate_errors.^2, 1)));
rms_error.mems = sqrt(mean(sum(position_errors.^2, 1)));
% Вивід значень RMS
fprintf('RMS похибка прискорень: %f м/с^2\n', rms_acceleration);
fprintf('RMS похибка швидкостей: %f м/с\n', rms_velocity);
fprintf('RMS похибка кутових швидкостей: %f рад/с\n', rms_angular_rate);
fprintf('RMS похибка для координат: %f м\n', rms_error.mems);
%% Побудова графіків вимірювання в системі
% Побудова графіків для гіроскопа
figure;
subplot(3,1,1); % Використання субплота для гіроскопа
plot(t, omega_history(1,:), 'r', t, mems_giro(1,:), 'b', 'LineWidth', 2);
title('Вимірювання гіроскопа – калібрована вісь X');
xlabel('Час (с)');
ylabel('Кутова швидкість (рад/с)');
legend('Початкові дані', 'Калібровані дані');
subplot(3,1,2);
plot(t, omega_history(2,:), 'r', t, mems_giro(2,:), 'b', 'LineWidth', 2);
title('Вимірювання гіроскопа – калібрована вісь Y');
xlabel('Час (с)');
ylabel('Кутова швидкість (рад/с)');
subplot(3,1,3);
plot(t, omega_history(3,:), 'r', t, mems_giro(3,:), 'b', 'LineWidth', 2);

```

```

title('Вимірювання гіроскопа – калібрована вісь Z');
xlabel('Час (с)');
ylabel('Кутова швидкість (рад/с)');
% Побудова графіків для акселерометра
figure;
subplot(3,1,1); % Використання субплота для акселерометра
plot(t, a_history(1,:), 'r', t, calibrated_accelerations(1,:), 'b');
title('Вимірювання акселерометра – калібрована вісь X');
xlabel('Час (с)');
ylabel('Прискорення (м/с^2)');
legend('Початкові дані', 'Калібровані дані');
subplot(3,1,2);
plot(t, a_history(2,:), 'r', t, calibrated_accelerations(2,:), 'b');
title('Вимірювання акселерометра – калібрована вісь Y');
xlabel('Час (с)');
ylabel('Прискорення (м/с^2)');
subplot(3,1,3);
plot(t, a_history(3,:), 'r', t, calibrated_accelerations(3,:), 'b');
title('Вимірювання акселерометра – калібрована вісь Z');
xlabel('Час (с)');
ylabel('Прискорення (м/с^2)');
% Графіки каліброваних швидкостей
figure;
subplot(3,1,1);
plot(t, calibrated_velocity(1,:), 'b');
title('Калібрована швидкість – вісь X');
xlabel('Час (с)');
ylabel('Швидкість (м/с)');
subplot(3,1,2);
plot(t, calibrated_velocity(2,:), 'b');
title('Калібрована швидкість – вісь Y');
xlabel('Час (с)');
ylabel('Швидкість (м/с)');
subplot(3,1,3);
plot(t, calibrated_velocity(3,:), 'b');
title('Калібрована швидкість – вісь Z');
xlabel('Час (с)');
ylabel('Швидкість (м/с)');
% Графіки каліброваних координат
figure;
subplot(3,1,1);

```

```

plot(t, calibrated_position(1,:), 'b');
title('Калібровані координати – вісь X');
xlabel('Час (с)');
ylabel('Позиція (м)');
subplot(3,1,2);
plot(t, calibrated_position(2,:), 'b');
title('Калібровані координати – вісь Y');
xlabel('Час (с)');
ylabel('Позиція (м)');
subplot(3,1,3);
plot(t, calibrated_position(3,:), 'b');
title('Калібровані координати – вісь Z');
xlabel('Час (с)');
ylabel('Позиція (м)');

%% ФУНКЦІЇ ДЛЯ МОДЕЛЮВАННЯ ПОХИБОК МЕМС СИСТЕМИ
function [accelerometer_errors, gyroscope_errors] = simulate_sensors_errors(t, accelerometer_params,
gyroscope_params)
    % Функція для симуляції похибок акселерометрів та гіроскопів
    % Генерація похибок для акселерометра
    [accel_bias_error, accel_scale_factor_error, accel_non_orthogonality_error, ...
    accel_drift_error, accel_random_noise] = generate_deterministic_errors(t, accelerometer_params);
    % Генерація похибок для гіроскопа
    [gyro_bias_error, gyro_scale_factor_error, gyro_non_orthogonality_error, ...
    gyro_drift_error, gyro_random_noise] = generate_deterministic_errors(t, gyroscope_params);
    % Загальні похибки для акселерометра
    accelerometer_errors = repmat(accel_bias_error, 3, 1) + ...
        repmat(accel_scale_factor_error, 3, 1) + ...
        repmat(accel_non_orthogonality_error, 3, 1) + ...
        repmat(accel_drift_error, 3, 1) + ...
        repmat(accel_random_noise, 3, 1);
    % Загальні похибки для гіроскопа
    gyroscope_errors = repmat(gyro_bias_error, 3, 1) + ...
        repmat(gyro_scale_factor_error, 3, 1) + ...
        repmat(gyro_non_orthogonality_error, 3, 1) + ...
        repmat(gyro_drift_error, 3, 1) + ...
        repmat(gyro_random_noise, 3, 1);
end
function [bias_error, scale_factor_error, non_orthogonality_error, drift_error, random_noise] =
generate_deterministic_errors(t, params)

```



```

% Функція для генерації детермінованих похибок для гіроскопів та акселерометрів
% Похибка зсуву
bias_error = params.bias + params.bias_drift_rate .* t; % Постійна похибка зсуву плюс зсув залежний від
часу
% Похибка масштабного коефіцієнта
scale_factor_error = params.scale_factor; % Постійна похибка масштабного коефіцієнта
% Похибка неортогональності
non_orthogonality_error = params.non_orthogonality; % Похибка неортогональності осей датчика
% Похибка дрейфу
drift_error = params.drift_rate .* t; % Дрейф датчика залежний від часу
% Випадкові похибки
random_noise = params.noise_level .* randn(size(t)); % Генерація шуму як випадкова похибка
end
% Функція для генерації випадкових похибок магнітометра
function magnetometer_errors = generate_magnetometer_errors(t, params)
% Генерація випадкових похибок магнітометра
deviation_error = params.deviation_error * (randn(size(t)) - 0.5);
inertia_error = params.inertia_error * (randn(size(t)) - 0.5);
magnetometer_errors = deviation_error + inertia_error;
end

%% ФУНКЦІЇ ДЛЯ МОДЕЛЮВАННЯ ПОХИБОК МЕМС СИСТЕМИ (ПІД ЧАС КАЛІБРУВАННЯ)
function [accelerometer_errors, gyroscope_errors] = simulate_sensors_errors_cal(t_cal, accelerometer_params,
gyroscope_params)
% Функція для симуляції похибок акселерометрів та гіроскопів
% Генерація похибок для акселерометра
[accel_bias_error, accel_scale_factor_error, accel_non_orthogonality_error, ...
accel_drift_error, accel_random_noise] = generate_deterministic_errors_cal(t_cal, accelerometer_params);
% Генерація похибок для гіроскопа
[gyro_bias_error, gyro_scale_factor_error, gyro_non_orthogonality_error, ...
gyro_drift_error, gyro_random_noise] = generate_deterministic_errors_cal(t_cal, gyroscope_params);
% Загальні похибки для акселерометра
accelerometer_errors = repmat(accel_bias_error, 3, 1) + ...
repmat(accel_scale_factor_error, 3, 1) + ...
repmat(accel_non_orthogonality_error, 3, 1) + ...
repmat(accel_drift_error, 3, 1) + ...
repmat(accel_random_noise, 3, 1);
% Загальні похибки для гіроскопа
gyroscope_errors = repmat(gyro_bias_error, 3, 1) + ...
repmat(gyro_scale_factor_error, 3, 1) + ...
repmat(gyro_non_orthogonality_error, 3, 1) + ...

```

```

        repmat(gyro_drift_error, 3, 1) + ...
        repmat(gyro_random_noise, 3, 1);

end

function [bias_error, scale_factor_error, non_orthogonality_error, drift_error, random_noise] =
generate_deterministic_errors_cal(t_cal, params)
    % Функція для генерації детермінованих похибок для гіроскопів та акселерометрів
    % Похибка зсуву
    bias_error = params.bias + params.bias_drift_rate .* t_cal; % Постійна похибка зсуву плюс зсув залежний
від часу
    % Похибка масштабного коефіцієнта
    scale_factor_error = params.scale_factor; % Постійна похибка масштабного коефіцієнта
    % Похибка неортогональності
    non_orthogonality_error = params.non_orthogonality; % Похибка неортогональності осей датчика
    % Похибка дрейфу
    drift_error = params.drift_rate .* t_cal; % Дрейф датчика залежний від часу
    % Випадкові похибки
    random_noise = params.noise_level .* randn(size(t_cal)); % Генерація шуму як випадкова похибка
end

%% Модель бортової GNSS системи (GPS)
%% Код для розрахунку позиціонування GPS
% Файл: gps_model.mlx
% Ініціалізація системи
gps_data = zeros(size(r_history));
gps_data(:, 1) = r_history(:, 1); % додавання стартової координати
error_bounds_type = 'GPS'; % Виберіть тип супутника тут
min_error = error_bounds.(error_bounds_type)(1);
max_error = error_bounds.(error_bounds_type)(2);
for i = 2:length(t)
    if mod(i, round(1/dt)) == 0 % Update GPS every second
        % Generate random error for each coordinate
        error_xyz = (2*rand(3,1)-1) .* (min_error + (max_error - min_error) * rand(3,1));
        % Normalize the total error so it does not exceed max_error
        total_error = norm(error_xyz);
        if total_error > max_error
            error_xyz = error_xyz * (max_error / total_error);
        end
        gps_data(:, i) = r_history(:, i) + error_xyz;
    else
        if i > 1
            gps_data(:, i) = gps_data(:, i-1); % Use the previous value if not updating
        end
    end
end

```

```

    end
end
end
% Ініціалізація масивів для зберігання RMS похибок
rms_errors_x = zeros(1, length(t));
rms_errors_y = zeros(1, length(t));
rms_errors_z = zeros(1, length(t));
% Розрахунок RMS помилки для кожного моменту часу та для кожної складової координати
for i = 1:length(t)
    rms_errors_x(i) = sqrt(mean((r_history(1,1:i) - gps_data(1,1:i)).^2));
    rms_errors_y(i) = sqrt(mean((r_history(2,1:i) - gps_data(2,1:i)).^2));
    rms_errors_z(i) = sqrt(mean((r_history(3,1:i) - gps_data(3,1:i)).^2));
end
%% Модельовання похибок під час роботи GPS
%% Графік вимірених даних GPS по осям
figure;
% Графік для даних GPS по осі X
subplot(3, 1, 1);
plot(t(1, :), gps_data(1, :), 'b'); % Дані GPS синім кольором
hold on;
plot(t(1, :), r_history(1, :), 'r'); % Реальні дані червоним кольором
title('Дані GPS по осі X');
xlabel('Час (с)');
ylabel('X (м)');
hold off;
% Графік для даних GPS по осі Y
subplot(3, 1, 2);
plot(t(1, :), gps_data(2, :), 'g'); % Дані GPS зеленим кольором
hold on;
plot(t(1, :), r_history(2, :), 'r'); % Реальні дані червоним кольором
title('Дані GPS по осі Y');
xlabel('Час (с)');
ylabel('Y (м)');
hold off;
% Графік для даних GPS по осі Z
subplot(3, 1, 3);
plot(t(1, :), gps_data(3, :), 'b'); % Дані GPS синім кольором
hold on;
plot(t(1, :), r_history(3, :), 'r'); % Реальні дані червоним кольором
title('Дані GPS по осі Z');
xlabel('Час (с)');

```

```

ylabel('Z (м)');
hold off;
windowSize = 20; % Розмір вікна для згладжування
halfWindow = floor(windowSize / 3);
% Ініціалізація згладжених даних
smoothed_gps_data = zeros(size(gps_data));
% Застосування ковзного середнього (з урахуванням країв)
for i = 1:size(gps_data, 2)
    % Визначення індексів вікна
    window_indices = max(1, i-halfWindow):min(size(gps_data, 2), i+halfWindow);

    % Розрахунок середнього для вікна
    smoothed_gps_data(:, i) = mean(gps_data(:, window_indices), 2);
end
% Розрахунок відстаней між згладженими точками
diff_smoothed_gps = diff(smoothed_gps_data, 1, 2);
smoothed_gps_distances = sqrt(sum(diff_smoothed_gps.^2, 1));
% Розрахунок загального пройденого шляху зі згладжених даних
total_smoothed_gps_distances = cumsum(smoothed_gps_distances);
% Додавання нуля на початку для згладжених даних
total_smoothed_gps_distances = [0, total_smoothed_gps_distances];
% Візуалізація загального пройденого шляху відносно часу
figure;
plot(t, total_smoothed_gps_distances, 'b');
title('Загальний пройдений шлях відносно часу');
xlabel('Час (с)');
ylabel('Пройдений шлях (м)');
grid on;
%%
%% Побудова графіків похибок для кожної складової координати
figure;
subplot(3, 1, 1);
plot(t, (r_history(1,:) - gps_data(1,:)));
title('Зміна помилки по осі X в часі');
xlabel('Час (с)');
ylabel('похибка X (м)');
grid on;
subplot(3, 1, 2);
plot(t, r_history(2,:) - gps_data(2,:));
title('Зміна помилки по осі Y в часі');
xlabel('Час (с)');

```

```

ylabel('похибка Y (м)');
grid on;
subplot(3, 1, 3);
plot(t, r_history(3,:) - gps_data(3,:));
title('Зміна RMS помилки по осі Z в часі');
xlabel('Час (с)');
ylabel('похибка Z (м)');
grid on;
% Розрахунок загальної RMS помилки
rms_total = sqrt(rms_errors_x.^2 + rms_errors_y.^2 + rms_errors_z.^2);
% Побудова графіка загальної RMS помилки
figure;
plot(t, rms_total);
title('Зміна загальної RMS помилки в часі');
xlabel('Час (с)');
ylabel('Загальна RMS похибка (м)');
grid on;
%% Значення RMS 3D для GPS системи
%% Розрахунок RMS похибок для координат
gps_errors_position = zeros(3, length(t));
for i = 1:length(t)
    gps_errors_position(1, i) = sqrt(mean((r_history(1,1:i) - gps_data(1,1:i)).^2));
    gps_errors_position(2, i) = sqrt(mean((r_history(2,1:i) - gps_data(2,1:i)).^2));
    gps_errors_position(3, i) = sqrt(mean((r_history(3,1:i) - gps_data(3,1:i)).^2));
end
% Абсолютне RMS значення для 3D для MEMC системи
rms_error_gps = sqrt(mean((r_history(1,:) - gps_data(1,:)).^2 + ...
    (r_history(2,:) - gps_data(2,:)).^2 + ...
    (r_history(3,:) - gps_data(3,:)).^2));
fprintf('Абсолютне RMS значення для GPS система: %f м\n', rms_error_gps);

%% Модель бортової підсистеми ISNS з корекцією та фільтрацією
%% Моделювання системи ISNS для БПЛА з корекцією від ПЗІ та фільтрацією
% Файл: isns_model_7.xml
%% Моделювання системи ІНС
% Параметри для корекції
isns_K_position = 0.5; % Коефіцієнти корекції для позиції
isns_K_velocity = 0.5; % Коефіцієнти корекції для швидкості
% Ініціалізація масивів для зберігання вимірювань
isns_v_estimated = zeros(3, length(t));
isns_r_estimated = zeros(3, length(t));

```

```

% Встановлення початкової позиції
isns_r_estimated(:, 1) = mems_data(:, 1); % Ініціалізація системи на старті, фіксація початкової координати
місії

% Інтеграція прискорення для отримання швидкості та зміщення
for i = 2:length(t)
    % Інтеграція прискорення для отримання швидкості
    isns_v_estimated(:, i) = isns_v_estimated(:, i-1) + mems_a(:, i) * dt;

    % Інтеграція швидкості для отримання зміщення
    isns_r_estimated(:, i) = isns_r_estimated(:, i-1) + isns_v_estimated(:, i) * dt;

% Корекція на основі даних GPS з частотою 1 Гц
if mod(i, round(1/dt)) == 0
    isns_position_error = gps_data(:, i) - isns_r_estimated(:, i);
    isns_r_estimated(:, i) = isns_r_estimated(:, i) + isns_K_position * isns_position_error;
    isns_velocity_error = (gps_data(:, i) - gps_data(:, i-1))/dt - isns_v_estimated(:, i);
    isns_v_estimated(:, i) = isns_v_estimated(:, i) + isns_K_velocity * isns_velocity_error;
end
end

% Параметри фільтра експоненціального згладжування для швидкості
alpha = 0.05; % Коефіцієнт згладжування (0 < alpha < 1)
% Ініціалізація згладженої швидкості
isns_v_smoothed = zeros(size(isns_v_estimated));
% Початкове значення згладженої швидкості може бути задано як перше вимірювання
isns_v_smoothed(:, 1) = isns_v_estimated(:, 1);
% Застосування експоненційного згладжування
for k = 2:length(t)
    isns_v_smoothed(:, k) = alpha * isns_v_estimated(:, k) + (1 - alpha) * isns_v_smoothed(:, k - 1);
end

%% Візуалізація коригованих даних ІНС
% Графік коригованого положення
figure;
subplot(3, 1, 1);
plot(t, isns_r_estimated(1, :), 'b');
hold on;
plot(t, r_history(1, :), 'r');
title('Кориговане положення по осі X');
xlabel('Час (с)');
ylabel('X (м)');
legend('Кориговане значення', 'Фактичне значення');
hold off;

```

```

subplot(3, 1, 2);
plot(t, isns_r_estimated(2, :), 'g');
hold on;
plot(t, r_history(2, :), 'r');
title('Кориговане положення по осі Y');
xlabel('Час (с)');
ylabel('Y (м)');
legend('Кориговане значення', 'Фактичне значення');
hold off;
subplot(3, 1, 3);
plot(t, isns_r_estimated(3, :), 'b');
hold on;
plot(t, r_history(3, :), 'r');
title('Кориговане положення по осі Z');
xlabel('Час (с)');
ylabel('Z (м)');
legend('Кориговане значення', 'Фактичне значення');
hold off;
% Графік коригованої швидкості
figure;
subplot(3, 1, 1);
plot(t, isns_v_smoothed(1, :), 'b', t, v_history(1,:), 'r');
title('Коригована швидкість по осі X');
xlabel('Час (с)');
ylabel('Швидкість X (м/с)');
legend('Кориговане значення', 'Фактичне значення');
grid on;
subplot(3, 1, 2);
plot(t, isns_v_smoothed(2, :), 'g', t, v_history(2,:), 'r');
title('Коригована швидкість по осі Y');
xlabel('Час (с)');
ylabel('Швидкість Y (м/с)');
legend('Кориговане значення', 'Фактичне значення');
grid on;
subplot(3, 1, 3);
plot(t, isns_v_smoothed(3, :), 'b', t, v_history(3,:), 'r');
title('Коригована швидкість по осі Z');
xlabel('Час (с)');
ylabel('Швидкість Z (м/с)');
legend('Кориговане значення', 'Фактичне значення');
grid on;

```

```

%% Ініціалізація фільтра Калмана
% Стан: [положення; швидкість]
isns_x_kalman = [isns_r_estimated(:, 1); isns_v_smoothed(:, 1)]; % початкова оцінка
n = length(isns_x_kalman); % розмірність стану
P = eye(n); % початкова матриця коваріації похибок оцінки
A = [eye(3), dt*eye(3); zeros(3, 3), eye(3)]; % матриця переходу стану
B = []; % матриця управління (немає зовнішнього входу)
C = [eye(3), zeros(3, 3)]; % матриця спостереження (ми спостерігаємо лише положення)
Q = diag([0.1, 0.1, 0.1, 0.01, 0.01, 0.01]); % коваріація шуму процесу
R = 0.1*eye(3); % коваріація шуму вимірювань
% Встановлення базової позиції
r_estimated(:, 1) = gps_data(:,1); % Ініціалізація МЕМС системи на старті, фіксація початкової координати
% Оптимальна фільтрація
isns_v_kalman = zeros(3, length(t));
for i = 2:length(t)
    % Прогноз
    isns_x_predicted = A * isns_x_kalman(:, i-1);
    P_predicted = A * P * A' + Q;

    % Обчислення коефіцієнта Калмана
    K = P_predicted * C' / (C * P_predicted * C' + R);

    % Оновлення оцінки на основі вимірювань
    isns_x_updated = isns_x_predicted + K * (gps_data(:, i) - C * isns_x_predicted);
    P = (eye(n) - K * C) * P_predicted;

    % Зберігання оновленої оцінки швидкості
    isns_v_kalman(:, i) = isns_x_updated(4:6);

    % Оновлення стану для наступної ітерації
    isns_x_kalman(:, i) = isns_x_updated;
end
isns_data = isns_x_kalman(1:3, :);
isns_v = isns_v_kalman;
%% Візуалізація результатів
figure;
subplot(3, 1, 1);
plot(t, isns_v_estimated(1, :), 'b', t, isns_v_kalman(1, :), 'g', t, v_history(1,:), 'r');
title('Швидкість по осі X');
xlabel('Час (с)');
ylabel('Швидкість X (м/с)');

```



```

legend('Початкова оцінка', 'Фільтр Калмана', 'Фактичне значення');
subplot(3, 1, 2);
plot(t, isns_v_estimated(2, :), 'b', t, isns_v_kalman(2, :), 'g', t, v_history(2,:), 'r');
title('Швидкість по осі Y');
xlabel('Час (с)');
ylabel('Швидкість Y (м/с)');
legend('Початкова оцінка', 'Фільтр Калмана', 'Фактичне значення');
subplot(3, 1, 3);
plot(t, isns_v_estimated(3, :), 'b', t, isns_v_kalman(3, :), 'g', t, v_history(3,:), 'r');
title('Швидкість по осі Z');
xlabel('Час (с)');
ylabel('Швидкість Z (м/с)');
legend('Початкова оцінка', 'Фільтр Калмана', 'Фактичне значення');

%% Візуалізація результатів для положення
figure;
subplot(3, 1, 1);
plot(t, isns_r_estimated(1, :), 'g', t, isns_x_kalman(1, :), 'b', t, r_history(1, :), 'r');
title('Положення по осі X');
xlabel('Час (с)');
ylabel('X (м)');
legend('Початкова оцінка', 'Фільтр Калмана', 'Фактичне значення');
subplot(3, 1, 2);
plot(t, isns_r_estimated(2, :), 'g', t, isns_x_kalman(2, :), 'b', t, r_history(2, :), 'r');
title('Положення по осі Y');
xlabel('Час (с)');
ylabel('Y (м)');
legend('Початкова оцінка', 'Фільтр Калмана', 'Фактичне значення');
subplot(3, 1, 3);
plot(t, isns_r_estimated(3, :), 'g', t, isns_x_kalman(3, :), 'b', t, r_history(3, :), 'r');
title('Положення по осі Z');
xlabel('Час (с)');
ylabel('Z (м)');
legend('Початкова оцінка', 'Фільтр Калмана', 'Фактичне значення');

% Візуалізація тривимірного шляху виміряного засобами позиціонування БПЛА
figure;
plot3(isns_r_estimated(1, :), isns_r_estimated(2, :), isns_r_estimated(3, :), 'g', ...
      isns_x_kalman(1, :), isns_x_kalman(2, :), isns_x_kalman(3, :), 'b', ...
      r_history(1, :), r_history(2, :), r_history(3, :), 'r');
title('Тривимірний шлях');

```

```

xlabel('X (м)');
ylabel('Y (м)');
zlabel('Z (м)');
legend('Початкова оцінка', 'Фільтр Калмана', 'Фактичне значення');
grid on;

%% Обчислення RMS похибок позиціонування
isns_rms_errors_x = sqrt(mean((r_history(1,:) - isns_x_kalman(1,:)).^2));
isns_rms_errors_y = sqrt(mean((r_history(2,:) - isns_x_kalman(2,:)).^2));
isns_rms_errors_z = sqrt(mean((r_history(3,:) - isns_x_kalman(3,:)).^2));
%% Виведення розрахованих значень RMS похибок
fprintf('RMS похибка по координаті X: %f м\n', isns_rms_errors_x);
fprintf('RMS похибка по координаті Y: %f м\n', isns_rms_errors_y);
fprintf('RMS похибка по координаті Z: %f м\n', isns_rms_errors_z);
%% Розрахунок абсолютного значення RMS 3D для ISNS системи
%% Розрахунок загальної RMS помилки для ICHC системи за весь час
rms_error.isns = sqrt(mean(isns_rms_errors_x.^2 + isns_rms_errors_y.^2 + isns_rms_errors_z.^2));
%% Виведення результатів точності для ISNS системи
fprintf('Загальна RMS 3D похибка позиціонування ICHC системи: %f (м)\n', rms_error.isns);
%% Модель LEO системи позиціонування (StarLink)
%% Код для розрахунку позиціонування LEO
% Файл: leo_model.mlx
leo_t = 40 / 1000; % частота генерації сигналу позиціонування (в секундах)
leo_F = floor(dt / leo_t); % кількість оновлень даних LEO за одиницю часу dt
error_bounds_type = 'LEO'; % Виберіть тип супутника тут
min_error = error_bounds.(error_bounds_type)(1);
max_error = error_bounds.(error_bounds_type)(2);
%% Ініціалізація даних LEO
leo_data = zeros(size(r_history));
leo_data(:, 1) = r_history(:, 1); % додавання початкової координати
leo_F_errors = zeros(3, length(t) * leo_F); % Масив для зберігання похибок F_leo
error_index = 1;
for i = 2:length(t)
    leo_measurements = zeros(3, leo_F); % Масив для зберігання F_leo вимірювань
    for j = 1:leo_F
        % Генерація випадкової похибки для кожної координати
        leo_error = (2*rand(3,1)-1) .* (min_error + (max_error - min_error) * rand(3,1));
        % Нормалізація загальної похибки, щоб вона не перевищувала max_error
        leo_total_error = norm(leo_error);
        if leo_total_error > max_error
            leo_error = leo_error * (max_error / leo_total_error);

```

```

end
leo_measurements(:, j) = leo_error;

% Зберігання кожної вимірної похибки у масив
leo_F_errors(:, error_index) = leo_error; % Збереження конкретної похибки

% Збільшення індексу для наступного виміру
error_index = error_index + 1;
end

% Фільтрація вимірювань
filtered_error = mean(leo_measurements, 2); % Використання середнього значення похибок

% Додавання вимірної похибки до координати
leo_data(:, i) = r_history(:, i) + filtered_error;
end

% Розрахунок RMS помилки для кожного моменту часу та для кожної складової координати
for i = 1:length(t)
    leo_rms_errors_x(i) = sqrt(mean((r_history(1,1:i) - leo_data(1,1:i)).^2));
    leo_rms_errors_y(i) = sqrt(mean((r_history(2,1:i) - leo_data(2,1:i)).^2));
    leo_rms_errors_z(i) = sqrt(mean((r_history(3,1:i) - leo_data(3,1:i)).^2));
end

%% Графік вимірних даних leo по осям
% Створення нового вікна для графіків
figure;
% Графік для даних leo по осі X
subplot(3, 1, 1);
plot(t, leo_data(1, :), 'b'); % Дані LEO синім кольором
hold on;
plot(t, r_history(1, :), 'r'); % Реальні дані червоним кольором
title('Дані LEO по осі X');
xlabel('Час (с)');
ylabel('X (м)');
legend('LEO Data', 'Actual Data');
hold off;
% Графік для даних leo по осі Y
subplot(3, 1, 2);
plot(t, leo_data(2, :), 'b'); % Дані LEO синім кольором
hold on;

```

```

plot(t, r_history(2, :), 'r'); % Реальні дані червоним кольором
title('Дані LEO по осі Y');
xlabel('Час (с)');
ylabel('Y (м)');
legend('LEO Data', 'Actual Data');
hold off;
% Графік для даних leo по осі Z
subplot(3, 1, 3);
plot(t, leo_data(3, :), 'b'); % Дані LEO синім кольором
hold on;
plot(t, r_history(3, :), 'r'); % Реальні дані червоним кольором
title('Дані LEO по осі Z');
xlabel('Час (с)');
ylabel('Z (м)');
legend('LEO Data', 'Actual Data');
hold off;
windowSize = 20; % Розмір вікна для згладжування
halfWindow = floor(windowSize / 3);
% Ініціалізація згладжених даних
smoothed_leo_data = zeros(size(leo_data));
% Застосування ковзного середнього (з урахуванням країв)
for i = 1:size(leo_data, 2)
    % Визначення індексів вікна
    window_indices = max(1, i-halfWindow):min(size(leo_data, 2), i+halfWindow);
    % Розрахунок середнього для вікна
    smoothed_leo_data(:, i) = mean(leo_data(:, window_indices), 2);
end
% Розрахунок відстаней між згладженими точками
diff_smoothed_leo = diff(smoothed_leo_data, 1, 2);
smoothed_leo_distances = sqrt(sum(diff_smoothed_leo.^2, 1));
% Розрахунок загального пройденого шляху зі згладжених даних
total_smoothed_leo_distances = cumsum(smoothed_leo_distances);
% Додавання нуля на початку для згладжених даних
total_smoothed_leo_distances = [0, total_smoothed_leo_distances];
% Візуалізація загального пройденого шляху відносно часу
figure;
plot(t, total_smoothed_leo_distances, 'b');
title('Загальний пройдений шлях відносно часу');
xlabel('Час (с)');
ylabel('Пройдений шлях (м)');
grid on;

```

```

%% Побудова графіків похибок для кожної складової координати
figure;
subplot(3, 1, 1);
plot(t, (r_history(1,:) - leo_data(1,:)));
title('Зміна помилки по осі X в часі');
xlabel('Час (с)');
ylabel('похибка X (м)');
grid on;
subplot(3, 1, 2);
plot(t, r_history(2,:) - leo_data(2,:));
title('Зміна помилки по осі Y в часі');
xlabel('Час (с)');
ylabel('похибка Y (м)');
grid on;
subplot(3, 1, 3);
plot(t, r_history(3,:) - leo_data(3,:));
title('Зміна RMS помилки по осі Z в часі');
xlabel('Час (с)');
ylabel('похибка Z (м)');
grid on;
% Розрахунок загальної RMS помилки
leo_rms_total = sqrt(leo_rms_errors_x.^2 + leo_rms_errors_y.^2 + leo_rms_errors_z.^2);
% Побудова графіка загальної RMS помилки
figure;
plot(t, leo_rms_total);
title('Зміна загальної RMS помилки в часі');
xlabel('Час (с)');
ylabel('Загальна RMS похибка (м)');
grid on;
%% Значення RMS 3D для leo системи
leo_errors_position = zeros(3, length(t));
for i = 1:length(t)
    leo_errors_position(1, i) = sqrt(mean((r_history(1,1:i) - leo_data(1,1:i)).^2));
    leo_errors_position(2, i) = sqrt(mean((r_history(2,1:i) - leo_data(2,1:i)).^2));
    leo_errors_position(3, i) = sqrt(mean((r_history(3,1:i) - leo_data(3,1:i)).^2));
end
% Абсолютне RMS значення для 3D для MEMC системи
rms_error.leo = sqrt(mean((r_history(1,:) - leo_data(1,:)).^2 + ...
    (r_history(2,:) - leo_data(2,:)).^2 + ...
    (r_history(3,:) - leo_data(3,:)).^2));
fprintf('Абсолютне RMS значення для LEO система: %f м\n', rms_error.leo);

```

```

%% Модель бортової LeGNSS підсистеми позиціонування з оптимальною фільтрацією
%% Код для моделювання LeGNSS системи позиціонування з оптимальною фільтрацією
% Файл: LeGNSS_model_8.mlx
% Ініціалізація калманівського фільтра
legnss_numStates = 6; % Кількість станів системи: x, y, z, vx, vy, vz
legnss_initialState = zeros(legnss_numStates, 1); % Початковий стан
legnss_initialCovariance = eye(legnss_numStates); % Початкова коваріація
legnss_numObservations = 3 * 3; % Три виміри позиції від трьох систем
legnss_observationMatrix = eye(legnss_numObservations, legnss_numStates); % Розширена матриця
спостереження
legnss_stateTransitionMatrix = eye(legnss_numStates); % Матриця переходу станів
% Шуми спостереження для кожної системи
legnss_observationNoiseCovariance = blkdiag(eye(3) * 1, eye(3) * 1, eye(3) * 1); % Блочно-діагональна
матриця
legnss_processNoiseCovariance = eye(legnss_numStates) * 0.1; % Коваріація шуму процесу
% Ініціалізація змінних для збереження результатів
LeGNSS_data = zeros(3, length(gps_data)); % Зберігання оцінки позиції
LeGNSS_velocity = zeros(3, length(gps_data)); % Зберігання оцінки швидкості
% Цикл прогнозування та оновлення
for k = 1:length(gps_data)
    % Крок прогнозування
    legnss_predictedState = legnss_stateTransitionMatrix * legnss_initialState;
    legnss_predictedCovariance = legnss_stateTransitionMatrix * legnss_initialCovariance *
legnss_stateTransitionMatrix' + legnss_processNoiseCovariance;
    % Отримання спостереження
    legnss_observation = [gps_data(:, k); leo_data(:, k); isns_data(:, k)]; % Об'єднання спостережень
    % Крок оновлення
    legnss_KalmanGain = legnss_predictedCovariance * legnss_observationMatrix' / (legnss_observationMatrix
* legnss_predictedCovariance * legnss_observationMatrix' + legnss_observationNoiseCovariance);
    legnss_initialState = legnss_predictedState + legnss_KalmanGain * (legnss_observation -
legnss_observationMatrix * legnss_predictedState);
    legnss_initialCovariance = (eye(legnss_numStates) - legnss_KalmanGain * legnss_observationMatrix) *
legnss_predictedCovariance;
    % Збереження результатів
    LeGNSS_data(:, k) = legnss_initialState(1:3); % Позиція: x, y, z
    LeGNSS_velocity(:, k) = legnss_initialState(4:6); % Швидкість: vx, vy, vz
end

%% Графіки LeGNSS системи позиціонування
% Побудова графіків

```

```

% Графік зміни позиції
figure;
subplot(3, 1, 1);
plot(t, LeGNSS_data(1, :), 'b', t, r_history(1, :), 'r');
title('Зміна позиції по осі X');
xlabel('Час');
ylabel('Позиція, м');
legend('LeGNSS', 'Фактично');
subplot(3, 1, 2);
plot(t, LeGNSS_data(2, :), 'b', t, r_history(2, :), 'r');
title('Зміна позиції по осі Y');
xlabel('Час');
ylabel('Позиція, м');
legend('LeGNSS', 'Фактично');
subplot(3, 1, 3);
plot(t, LeGNSS_data(3, :), 'b', t, r_history(3, :), 'r');
title('Зміна позиції по осі Z');
xlabel('Час');
ylabel('Позиція, м');
legend('LeGNSS', 'Фактично');

%% Диференціальне обчислення швидкості БПЛА
%% Фільтрація даних швидкості БПЛА з використанням фільтру Калмана
% Розмірність стану для фільтра Калмана (позиція і швидкість)
kalmanStateSize = 6; % x, y, z, vx, vy, vz
% Ініціалізація стану та коваріації
kalmanState = zeros(kalmanStateSize, 1); % Початковий стан [x; y; z; vx; vy; vz]
kalmanCovariance = eye(kalmanStateSize); % Початкова коваріація
% Матриці для Калмана
A = [1 0 0 dt 0 0; % Матриця переходу станів
     0 1 0 0 dt 0;
     0 0 1 0 0 dt;
     0 0 0 1 0 0;
     0 0 0 0 1 0;
     0 0 0 0 0 1];
H = [0 0 0 1 0 0; % Матриця спостереження (тільки для швидкості)
     0 0 0 0 1 0;
     0 0 0 0 0 1];
Q = eye(kalmanStateSize) * 0.1; % Шум процесу
R = eye(3) * 0.5; % Шум спостереження (припускаємо більший шум для швидкості)
% Ініціалізація масивів для збереження результатів

```

```

filteredVelocity = zeros(3, length(t));
% Калманівський фільтр для оцінки швидкості
for k = 2:length(t)
    % Прогнозування наступного стану
    kalmanState = A * kalmanState;
    kalmanCovariance = A * kalmanCovariance * A' + Q;
    % Оновлення за допомогою вимірювання швидкості
    velocityMeasurement = (LeGNSS_data(:, k) - LeGNSS_data(:, k - 1)) / dt;
    z = velocityMeasurement - H * kalmanState; % Відхилення вимірювання від прогнозу
    S = H * kalmanCovariance * H' + R;
    K = kalmanCovariance * H' / S; % Коефіцієнт Калмана
    kalmanState = kalmanState + K * z; % Оновлення стану
    kalmanCovariance = (eye(kalmanStateSize) - K * H) * kalmanCovariance; % Оновлення коваріації
    % Оновлення стану з врахуванням прискорення
    accelerationMeasurement = calibrated_accelerations(:, k);
    kalmanState(4:6) = kalmanState(4:6) + accelerationMeasurement * dt; % vx = vx + ax*dt, vy = vy + ay*dt,
vz = vz + az*dt
    % Зберігання фільтрованої швидкості
    LeGNSS_filteredVelocity(:, k) = kalmanState(4:6);
end
% Параметри фільтра експоненціального згладжування
alpha = 0.03; % Коефіцієнт згладжування (0 < alpha < 1)
% Ініціалізація згладженої швидкості
LeGNSSsmoothedVelocity = zeros(size(isns_v));
% Початкове значення згладженої швидкості може бути задано як перше вимірювання
LeGNSSsmoothedVelocity(:, 1) = isns_v(:, 1);
% Застосування експоненційного згладжування
for k = 2:length(t)
    LeGNSSsmoothedVelocity(:, k) = alpha * isns_v(:, k) + (1 - alpha) * LeGNSSsmoothedVelocity(:, k - 1);
end

% Графік зміни швидкості
figure;
subplot(3, 1, 1);
plot(t, LeGNSSsmoothedVelocity(1, :), 'g', t, v_history(1,:), 'r');
title("Зміна швидкості по осі X");
xlabel("Час");
ylabel("Швидкість, м/с");
subplot(3, 1, 2);
plot(t, LeGNSSsmoothedVelocity(2, :), 'g', t, v_history(2,:), 'r');

```



```

title('Зміна швидкості по осі Y');
xlabel('Час');
ylabel('Швидкість, м/с');
subplot(3, 1, 3);
plot(t, LeGNSSsmoothedVelocity(3, :), 'g', t, v_history(3,:), 'r');
title('Зміна швидкості по осі Z');
xlabel('Час');
ylabel('Швидкість, м/с');

% Візуалізація тривимірного шляху визначеного підсистемою LeGNSS
figure;
plot3(LeGNSS_data(1, :), LeGNSS_data(2, :), LeGNSS_data(3, :) + coordinates(3,1), 'g', ...
      r_history(1, :), r_history(2, :), r_history(3, :) + coordinates(3,1), 'r');
title('Тривимірний шлях');
xlabel('X (м)');
ylabel('Y (м)');
zlabel('Z (м)');
legend('Оцінка GNSS', 'Фактичне значення');
grid on
%axis equal; % Ця команда забезпечує однаковий масштаб по всіх осях
% Встановлення масштабу для осей axis([Xmin Xmax Ymin Ymax Zmin Zmax])
xlim([-6000 2000])
ylim([-1000 6100])
zlim([coordinates(3,1) 500])
view([-244 20])

%% Розрахунок RMS похибок
%% Обчислення RMS похибок позиціонування
LeGNSS_rms_errors_x = sqrt(mean((r_history(1,:) - LeGNSS_data(1,:)).^2));
LeGNSS_rms_errors_y = sqrt(mean((r_history(2,:) - LeGNSS_data(2,:)).^2));
LeGNSS_rms_errors_z = sqrt(mean((r_history(3,:) - LeGNSS_data(3,:)).^2));
%% Виведення розрахованих значень RMS похибок
fprintf('RMS похибка по координаті X: %f м\n', LeGNSS_rms_errors_x);
fprintf('RMS похибка по координаті Y: %f м\n', LeGNSS_rms_errors_y);
fprintf('RMS похибка по координаті Z: %f м\n', LeGNSS_rms_errors_z);
%% Розрахунок абсолютного значення RMS 3D для ISNS системи
% Розрахунок загальної RMS помилки для ICHC системи за весь час
rms_error.LeGNSS = sqrt(mean(LeGNSS_rms_errors_x.^2 + LeGNSS_rms_errors_y.^2 +
LeGNSS_rms_errors_z.^2));
% Виведення результатів точності для ISNS системи
fprintf('Загальна RMS 3D похибка позиціонування ICHC системи: %f (м)\n', rms_error.LeGNSS);

```

```

%% Аналіз похибок систем позиціонування
figure;
subplot(3, 1, 1);
plot(t,isns_data(1, :)-r_history(1, :), 'g',t, LeGNSS_data(1, :)-r_history(1, :), 'b',t,r_history(1, :)-r_history(1,
:), 'r');
title('Похибка позиціонування по осі X');
xlabel('Час, с');
ylabel('Похибка, м');
legend('ISNS','LeGNSS','Нуль');
subplot(3, 1, 2);
plot(t,isns_data(2, :)-r_history(2, :), 'g',t, LeGNSS_data(2, :)-r_history(2, :), 'b',t,r_history(1, :)-r_history(1,
:), 'r');
title('Похибка позиціонування по осі Y');
xlabel('Час, с');
ylabel('Похибка, м');
legend('ISNS','LeGNSS','Нуль');
subplot(3, 1, 3);
plot(t,isns_data(3, :)-r_history(3, :), 'g',t, LeGNSS_data(3, :)-r_history(3, :), 'b',t,r_history(1, :)-r_history(1,
:), 'r');
title('Похибка позиціонування по осі Z');
xlabel('Час, с');
ylabel('Похибка, м');
legend('ISNS','LeGNSS','Нуль');

%% Похибки швидкості
figure;
subplot(3, 1, 1);
plot(t,isns_v_kalman(1, :)-v_history(1, :), 'g',t, LeGNSSsmoothedVelocity(1, :)-v_history(1, :), 'b',t,v_history(1,
:)-v_history(1, :), 'r');
title('Похибка швидкості по осі X');
xlabel('Час');
ylabel('Швидкість, м/с');
legend('ISNS','LeGNSS','Нуль');
subplot(3, 1, 2);
plot(t,isns_v_kalman(2, :)-v_history(2, :), 'g',t, LeGNSSsmoothedVelocity(2, :)-v_history(2, :), 'b',t,v_history(2,
:)-v_history(2, :), 'r');
title('Похибка швидкості по осі Y');
xlabel('Час');
ylabel('Швидкість, м/с');
legend('ISNS','LeGNSS','Нуль');

```

```

subplot(3, 1, 3);
plot(t, isns_v_kalman(3, :)-v_history(3, :), 'g', t, LeGNSSsmoothedVelocity(3, :)-v_history(3, :), 'b', t, v_history(3, :)-v_history(3, :), 'r');
title('Похибка швидкості по осі Z');
xlabel('Час');
ylabel('Швидкість, м/с');
legend('ISNS', 'LeGNSS', 'Нуль');

% Графік коригованого положення
figure;
subplot(3, 1, 1);
plot(t, isns_r_estimated(1, :), 'g');
hold on;
plot(t, LeGNSS_data(1, :), 'b');
plot(t, r_history(1, :), 'r');
title('Положення по осі X');
xlabel('Час (с)');
ylabel('X (м)');
legend('ISNS', 'LeGNSS', 'Фактичне значення');
hold off;
subplot(3, 1, 2);
plot(t, isns_r_estimated(2, :), 'g');
hold on;
plot(t, LeGNSS_data(2, :), 'b');
plot(t, r_history(2, :), 'r');
title('Положення по осі Y');
xlabel('Час (с)');
ylabel('Y (м)');
legend('ISNS', 'LeGNSS', 'Фактичне значення');
hold off;
subplot(3, 1, 3);
plot(t, isns_r_estimated(3, :), 'g');
hold on;
plot(t, LeGNSS_data(3, :), 'b');
plot(t, r_history(3, :), 'r');
title('Положення по осі Z');
xlabel('Час (с)');
ylabel('Z (м)');
legend('ISNS', 'LeGNSS', 'Фактичне значення');
hold off;

```

ДОДАТОК Д. Список публікацій здобувача**Праці, які відображають основні наукові результати дисертації:**

1. Васильєв В.М., Рогожин В.О., Долінце Б.І. Схема комплексування інерціальної і супутникової навігаційних систем та її дослідження. *Проблеми створення, випробування, застосування та експлуатації складних інформаційних систем: збірник наукових праць*. Житомир: ЖВІ НАУ, 2013. Вип. 8. С. 20–25. (Входить до реєстру наукових видань України, Категорія Б; Входить до міжнародних наукометричних баз Copernicus, SemanticScholar, Ulrichsweb GSD, ResearchBib, GoogleScholar, BASE)
2. Васильєв В.М., Рогожин В.О., Долінце Б.І. Дослідження похибок інерціально-супутникової навігаційної системи. *Наука і техніка Повітряних сил Збройних сил України*. Харків: ХУПС, 2013. Вип. 3(12). С. 139–143. (Входить до реєстру наукових видань України, Категорія Б; Входить до міжнародних наукометричних баз GoogleScholar, DOAJ, WorldCat, BASE)
3. Vasyliiev V.M., Rogozhyn V.O., Dolintse B.I. Integration of inertial and satellite navigation systems using corrective circuits for UAV. *Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*. IEEE Xplore, 2015. P. 193–197. DOI: 10.1109/APUAVD.2015.7346597. (date of access: 27.12.2023; Входить до міжнародних наукометричних баз Scopus, IEEE Xplore)
4. Vasyliiev V., Rogozhyn V., Dolintse B. Accuracy analysis of inertial-satellite navigation system Integrated with using corrective circuits. *Electronics and Control Systems*. 2015. Вип. 4(46). P. 46–51. DOI: 10.18372/1990-5548.46.9930. (Входить до реєстру наукових видань України, Категорія Б; Входить до міжнародних наукометричних баз CrossRef, EBSCOhost, ResearchBib, SSM, Ulrichsweb GSD)
5. Vasyliiev V., Dolintse B. Integration of Inertial and Satellite Navigation Systems with using Corrective Circuits and Filtering. *Methods and Systems of Navigation and Motion Control (MSNMC)*. IEEE Xplore, 2016. P. 275–278. DOI:

10.1109/MSNMC.2016.7783160. (date of access: 27.12.2023; Входить до міжнародних наукометричних баз *Scopus*, IEEE Xplore)

6. Dolintse B., Balakin S. Risk assessment in computer networks inherent in critical infrastructures. *Проблеми інформатизації та управління*. 2022. Вип. 2(70). Р. 4–9. DOI: 10.18372/2073-4751.70.16840. (date of access: 27.12.2023; (Входить до реєстру наукових видань України, Категорія Б; Входить до міжнародних наукометричних баз Google Scholar, EBSCOhost, WorldCat, BASE, SSM, CAS)

7. Dolintse B. Architecture of integrated navigation systems with enhanced coordinate accuracy and fault detection. *Проблеми інформатизації та управління*. 2023. Вип. 2(74). Р. 31–37. DOI: 10.18372/2073-4751.74.17878. (date of access: 27.12.2023; Входить до реєстру наукових видань України, Категорія Б; Входить до міжнародних наукометричних баз Google Scholar, EBSCOhost, WorldCat, BASE, SSM, CAS)

8. Zhukov I., Dolintse B. Enhancing accuracy of information processing in onboard subsystems of UAVs. *Technology Audit and Production Reserves*. №5/2(73), 2023. Р. 6–10. DOI: 10.15587/2706-5448.2023.287700. (date of access: 27.12.2023; (Входить до реєстру наукових видань України, Категорія Б; Входить до міжнародних наукометричних баз Google Scholar, Copernicus, Ulrichsweb GSD, CrossRef, DOAJ, OAJI, CiteFactor, WorldCat, OAA, DRJI)

9. Zhukov I., Dolintse B., Balakin S. Improving the Accuracy of Air Navigation Systems for Unmanned Aerial Vehicles. *Dependable Systems, Services and Technologies (DESSERT)*. IEEE Xplore, 2023. Р. 1–7. DOI: 10.1109/DESSERT61349.2023.10416511. (date of access: 12.02.2024; Входить до міжнародних наукометричних баз *Scopus*, IEEE Xplore)

10. Dolintse B. Robust LeGNSS positioning subsystem for UAV with correction and optimal filtering. *Проблеми інформатизації та управління*. 2023. Вип. 4 (76). Р. 10–16. DOI: 10.18372/2073-4751.76.18235. (date of access: 27.12.2023; Входить до реєстру наукових видань України, Категорія Б; Входить до міжнародних наукометричних баз Google Scholar, EBSCOhost, WorldCat, BASE, SSM, CAS)

Праці, які додатково відображають наукові результати дисертації:

11. Васильєв В.М., Рогожин В.О., Долінце Б.І. Моделювання та оптимізація параметрів інтегрованої інерціально-супутникової навігаційної системи з коригувальними ланцюгами. *Молодий вчений*. 2016. Вип. 7(34). С. 184–187. (Входить до міжнародних наукометричних баз CrossRef, GoogleScholar, OAJ, ResearchBib, OUCI, Index Copernicus)
12. Долінце Б.І. Вплив навігаційних похибок на прийняття рішень в системах попередження зіткнень літаків. *Політ. Сучасні проблеми науки: XII міжнар. науково-практ. конф. студентів та молодих учених, м. Київ, 4–5 квітня 2012 р.: тези доп.* Київ: НАУ, 2012. С. 30.
13. Долінце Б.І. Дослідження характеристик інерціально-супутникової навігаційної системи з коректорами в динамічних режимах роботи. *Політ. Сучасні проблеми науки: XIII міжнар. науково-практ. конф. студентів та молодих учених, м. Київ, 3–4 квітня 2013 р.: тези доп.* Київ: НАУ, 2013. С. 123.
14. Долінце Б.І. Сучасні методи та засоби обробки інформації в інтегрованих навігаційних системах літальних апаратів. *Політ. Сучасні проблеми науки: XV міжнар. науково-практ. конф. студентів та молодих учених, 8–9 квітня 2015 р.: тези доп.* Київ: НАУ, 2015. С. 25.
15. Dolintse B. Modern Trends and Issues of the Development and Improve the Accuracy of Navigation Systems for UAVs. В: *Safety in Aviation and Space Technologies: Proceeding of The Seventh World Congress “Aviation in the XXI-st Century”*, 19–21 вересня 2016, Kyiv. Vol.2. P. 59–62.
16. Zhukov I., Dolintse B., Balakin S. Using corrective circuits and filtering for integration of low-cost inertial and satellite navigation systems. *Safety in Aviation and Space Technologies: Proceeding of The Twenty Two World Congress “Aviation in the XXI-st Century”*, September 28–30, 2022, Kyiv. 2022. Vol.2.2. P. 8–12.
17. Долінце Б.І. Методи зменшення похибок інерціально-супутникової навігаційної системи. *Проблеми навігації і управління рухом: всеукр. наук.-практ.*

конф. молодих учених та студентів, 28–29 листопада 2012, Київ. Тези доп. НАУ. 2012. С. 45.

18. Васильєв В.М., Долінце Б.І. Інерціально-супутникові навігаційні системи для безпілотних літальних апаратів. *Сучасні проблеми авіакосмічних технологій та систем*: науково-практичний семінар, 17–21 червня 2013, м. Житомир. Тези доп. НАУ. 2013. С. 6.

19. Журавель Н.В., Долінце Б.І. Підвищення точності навігаційного сигналу за допомогою фільтру Калмана. *Збірник тез доповідей XIV Міжнародної науково-практичної конференції “Комп’ютерні системи та мережні технології” (CSNT-2023)*, Київ, 13–14 квітня 2023. Київ: НАУ. 2023. С. 92–93.

20. А. с. №122052 Україна. Комп’ютерна програма «Система адаптивної обробки інформації в багато-супутниковій системі позиціонування LeGNSS» / Б. І. Долінце. 2023.