

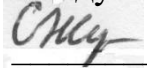
**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

**Радіотехнічний факультет
Кафедра радіотехнічних пристроїв**

«На правах рукопису»
УДК 621.396.621:004.896.72

До захисту допущено:

Завідувач кафедри

 Сергій ЖУК

« 12 » _____ грудня _____ 2022р.

Магістерська дисертація

на здобуття ступеня магістра

за освітньо-професійною програмою

«Радіотехнічні комп'ютеризовані системи»

зі спеціальності 172 «Телекомунікації та радіотехніка»

**на тему: «Алгоритм траєкторної фільтрації за даними системи
відеоспостереження і FMCW далекоміра»**

Виконав:

студент II курсу, групи РС-11мп

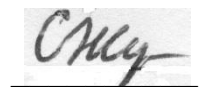
Сатиренко Максим Ігорович



Керівник:

д.т.н., проф.,

Жук Сергій Якович



Рецензент:

Доцент, проф. каф. РІ

Шарпан Олег Борисович

Засвідчую, що у цій магістерській
дисертації немає запозичень з праць
інших авторів без відповідних
посилань.

Студент



Київ – 2022 року

**Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»**

Радіотехнічний факультет

Кафедра радіотехнічних систем

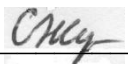
Рівень вищої освіти – другий (магістерський)

Спеціальність – 172 «Телекомунікації та радіотехніка»

Освітньо-професійна програма «Радіотехнічні комп'ютеризовані системи»

ЗАТВЕРДЖУЮ

Завідувач кафедри

 Сергій ЖУК

« 5 » вересня 2022р.

ЗАВДАННЯ

**на магістерську дисертацію
студенту Сатиренку Максиму Ігоровичу**

1. Тема дисертації: «Алгоритм траєкторної фільтрації БПЛА за даними системи відеоспостереження і FMCW далекоміра», науковий керівник дисертації Жук Сергій Якович, д.т.н., проф., затверджена наказом по університету від «09» листопада 2022 р. № 4137-с
2. Термін подання студентом дисертації 20 грудня 2022р.
3. Об'єкт дослідження: процес траєкторної фільтрації БПЛА за даними системи відеоспостереження та FMCW далекоміру
4. Вихідні дані: тактико-технічні характеристики та зображення типових малорозмірних БПЛА, характеристики оптичних систем відеоспостереження, характеристики FMCW далекоміру
5. Перелік завдань, які потрібно розробити: виконати огляд існуючих БПЛА; розглянути існуючі системи спостереження руху БПЛА; виконати аналіз систем відеоспостереження; виконати аналіз роботи FMCW далекоміру;

розробити алгоритм траєкторної фільтрації на основі системи з відеоспостереження та FMCW далекоміру; розглянути можливості практичного використання розробленого алгоритму в сучасних реаліях

6. Орієнтовний перелік графічного (ілюстративного) матеріалу: презентація у вигляді слайдів (PowerPoint)
7. Орієнтовний перелік публікацій: тези конференції
8. Дата видачі завдання 5 вересня 2022 року

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Обґрунтування актуальності обраної теми	25.10.2022	
2	Аналіз структури і принципів роботи системи відеоспостереження	28.10.2022	
3	Аналіз структури і принципів роботи FMCW далекоміру	3.11.2022	
4	Розрахунок математичної моделі вимірювання координат БПЛА за даними системи	9.11.2022	
5	Аналіз точносних характеристик вимірювання координат БПЛА за даними системи	15.11.2022	
6	Синтез алгоритмів траєкторної фільтрації БПЛА за даними системи відеоспостереження і FMCW далекоміру	24.11.2022	
7	Аналіз синтезованого квазіоптимального алгоритму траєкторної фільтрації БПЛА за даними системи відеоспостереження і FMCW далекоміру	1.12.2022	
8	Аналіз практичної реалізації розробленого алгоритму	10.12.2022	
9	Розробка стартап проекту	15.12.2022	

Студент
Науковий керівник

Максим САТИРЕНКО
Сергій ЖУК

РЕФЕРАТ

Магістерська дисертація на тему «Алгоритм траєкторної фільтрації БПЛА за даними системи відеоспостереження і FMCW далекоміра» складається з 84 сторінок, що включають в себе 8 розділів, 2 додатки, 20 ілюстрації, 19 таблиць, 17 бібліографічних найменувань за переліком джерел посилань.

В роботі розроблено алгоритм траєкторної фільтрації БПЛА за даними системи відеоспостереження та FMCW далекоміра для супроводження, та забезпечення оцінок положення БПЛА в просторі.

Метою дослідження являється підвищення ефективності локалізації БПЛА в просторі з використанням системи відеоспостереження та FMCW далекоміру, шляхом розробки алгоритму траєкторної фільтрації з розпізнаванням різних видів його руху.

Об’єкт дослідження – процес траєкторної фільтрації БПЛА за даними системи відеоспостереження та FMCW далекоміру.

Предмет дослідження – методи стеження та оцінки положення БПЛА за даними відеоспостережень та FMCW далекоміру.

Для виконання заданої задачі потрібно виконати такі завдання:

- провести аналіз існуючих методів стеження за БПЛА;
- синтезувати оптимальний алгоритм стеження та оцінювання положення БПЛА в просторі;
- оцінити ефективність розробленого алгоритму шляхом статистичного моделювання на ЕОМ;
- оцінити можливості практичної реалізації розробленого алгоритму.

Ключові слова: БПЛА, алгоритм, відеосупроводження, орієнтація в просторі, FMCW далекомір, траєкторна фільтрація, моделювання на ЕОМ.

ANNOTATION

The master's thesis on the topic "Algorithm of UAV trajectory filtering based on the data of the video surveillance system and FMCW rangefinder" consists of 84 pages, including 8 chapters, 2 appendices, 20 illustrations, 19 tables, 17 bibliographic names according to the list of reference sources.

The work developed an algorithm for UAV localization, which uses a video surveillance system and an FMCW range finder for tracking and providing estimates of the UAV's position in space.

The aim of the study is to increase the efficiency of UAV localization in space using a video surveillance system and FMCW rangefinder, by developing an algorithm aimed at specifying the position and recognizing various maneuvers of the target.

The object of research is the UAV trajectory filtering algorithm based on video surveillance system and FMCW rangefinder data.

The subject of research is methods of tracking and assessing the position of UAVs based on video surveillance and FMCW rangefinder data.

To complete the given task, we need to perform the following tasks:

- analyze existing UAV tracking methods;
- to synthesize the optimal algorithm for tracking and estimating the position of the UAV in space;
- evaluate the effectiveness of the developed algorithm by means of statistical modeling on a computer;
- evaluate the possibilities of practical implementation of the developed algorithm.

Keywords: UAV, adaptive algorithm, video tracking, orientation in space, FMCW radar, trajectory filtering.

ПЕРЕЛІК СКОРОЧЕНЬ

БПЛА – безпілотний літальний апарат;

FMCW – частотно-модульований безперервно-хвильовий;

ЕОМ –електронно-обчислювальна машина;

PTZ –функція панорамування, нахилу, та зуму;

F2F – міжкадрове співвідношення;

EKF – розширений фільтр Калмана;

СКВ – середньоквадратичне відхилення.

ЗМІСТ

ВСТУП	9
1 АКТУАЛЬНІСТЬ. ОГЛЯД ІСНУЮЧИХ РІШЕНЬ. РОЗРОБКА ТА АНАЛІЗ ТЕХНІЧОГО ЗАВДАННЯ.....	13
2 ОПИС ВІДЕОСПОСТЕРЕЖЕННЯ	22
2.1 Опис роботи системи.....	22
2.1.1 Реєстрація зображень	23
2.1.2 Одночасна оцінка орієнтації та фокусної відстані	25
3 ОПИС FMCW ДАЛЕКОМІРУ В СИСТЕМІ ТРАЄКТОРНОЇ ФІЛЬТРАЦІЇ РУХУ БПЛА.....	26
4 АНАЛІЗ ТОЧНОСИХ ХАРАКТЕРИСТИК ВИМІРЮВАННЯ КООРДИНАТ БПЛА ЗА ДАНИМИ СИСТЕМИ ВІДЕОСПОСТЕРЕЖЕННЯ І FMCW ДАЛЕКОМІРУ	32
5 СИНТЕЗ АЛГОРИТМІВ ТРАЄКТОРНОЇ ФІЛЬТРАЦІЇ БПЛА ЗА ДАНИМИ СИСТЕМИ ВІДЕОСПОСТЕРЕЖЕННЯ І FMCW ДАЛЕКОМІРУ.....	37
6 АНАЛІЗ СИНТЕЗОВАНОГО КВАЗІОПТИМАЛЬНОГО АЛГОРИТМУ ТРАЄКТОРНОЇ ФІЛЬТРАЦІЇ БПЛА ЗА ДАНИМИ СИСТЕМИ ВІДЕОСПОСТЕРЕЖЕННЯ І FMCW ДАЛЕКОМІРУ	44
7 АНАЛІЗ МОЖЛИВОСТЕЙ ПРАКТИЧНОЇ РЕАЛІЗАЦІЇ РОЗРОБЛЕНОГО КВАЗІОПТИМАЛЬНОГО АЛГОРИТМУ ТРАЄКТОРНОЇ ФІЛЬТРАЦІЇ	51
8 РОЗРОБКА СТАРТАП ПРОЕКТУ	55
8.1 Опис ідеї продукту.....	56
8.2 Технічний аудит ідеї проекту	57
8.3 Аналіз ринкових можливостей запуску стартап-проекту.....	58
8.4 Розроблення ринкової стратегії проекту	60

8.5 Розроблення маркетингової програми стартап-проекту	62
8.6 Висновки	64
ВИСНОВКИ.....	65
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	67
ДОДАТОК А.....	69
ДОДАТОК Б	81

ВСТУП

Магістерська робота на тему «**Алгоритм траєкторної фільтрації БПЛА за даними системи відеоспостереження і FMCW далекоміру**» складається з 8 розділів.

БПЛА зазвичай називають безпілотний літальний апарат, здатний літати і перебувати в повітрі без участі людини на борту, забезпечуючи більш економічно ефективні операції, ніж еквівалентні пілотовані системи, і виконуючи економічно ефективні критичні місії без ризику для людського життя [1].

Протягом останнього десятиліття БПЛА використовувалися в різних сферах, включаючи: виявлення та відстеження об'єктів, громадську безпеку, нагляд за дорожнім рухом, дослідження прихованих або небезпечних районів, навігацію в приміщеннях та на відкритому повітрі, зондування атмосфери, ліквідацію наслідків стихійних лих, охорону здоров'я, обмін даними, управління інфраструктурою, управління надзвичайними ситуаціями та кризовими ситуаціями, вантажоперевезення, моніторинг лісових пожеж та логістику.

Вони також отримали значну увагу в різних дисциплінах військових служб завдяки своїй підвищеній стабільності і витривалості в ряді операцій.

БПЛА відіграють невід'ємну роль у військових розвідувальних місіях. Деякі країни включили БПЛА до своїх оборонних стратегічних планів. Країни використовують ці літаючі робототехнічні машини для виявлення ворога, боротьби з браконьєрством, прикордонного контролю і морського моніторингу критично важливих морських шляхів. Недорогі, надійні та універсальні БПЛА сьогодні відіграють важливу роль у повітряному спостереженні, моніторингу та обстеженні будь-якої конкретної території з метою запобігання будь-якій протиправній діяльності. Наприклад, за допомогою безпілотників можна виявити будь-яку загрозу, і вони можуть бути використані для моніторингу будь-якої рухової активності в будь-якій

обмеженій зоні. БПЛА може надавати ці послуги, реагуючи на автоматичну тривогу з мінімальними ручними зусиллями.

У разі виникнення техногенних або природних катастроф, таких як терористичні акти, цунамі та повені, БПЛА можуть отримати доступ до місць, небезпечних для діяльності людей. Ці катастрофи можуть завдати серйозної шкоди телекомунікаційній інфраструктурі, транспортним, енергетичним та водопровідним комунікаціям. БПЛА можуть допомогти у зборі інформації, швидкому прийнятті рішень та орієнтуванні в завалах. Радари, датчики і високоякісні камери, вбудовані в БПЛА, можуть допомогти рятувальним командам виявити пошкодження і негайно розпочати відновлювальні роботи, а також відправити такі ресурси, як медичні аптечки першої допомоги і пілотовані вертольоти. БПЛА можуть допомогти у своєчасній оцінці масштабів катастрофи, оповіщенні про неї і пошуку ефективних контрзаходів. У випадку лісової пожежі рій дронів з вогнегасниками може контролювати, обстежувати та відстежувати будь-яку територію, не наражаючи на небезпеку людські життя. Таким чином, БПЛА можуть допомогти в режимі реального часу охопити велику територію без ризику для безпеки персоналу, який бере участь в операціях. Раннє попередження за допомогою БПЛА може допомогти врятувати людей і диких тварин, які перебувають у небезпеці.

З іншого боку БПЛА є однією з найбільших загроз, з якими стикаються учасники бойових дій. Вони можуть управлятися вручну, але сьогодні багато з них управляються віддаленим оператором або навіть автономно за допомогою бортових комп'ютерів. БПЛА є основною частиною безпілотних авіаційних систем (БАС), які також включають в себе оператора або комп'ютер і систему зв'язку, яка дозволяє цим двом частинам працювати разом.

БПЛА широко розрізняються за конфігурацією, розміром, дальністю польоту, вагою, типом двигуна і продуктивністю для того, щоб нести різноманітне корисне навантаження, в тому числі засоби зв'язку, навігаційне обладнання, датчики і камери.

Окремо зупинимось на малорозмірних БПЛА.

Малорозмірні БПЛА є досить проблемною ціллю, тому що розмір обмежує діапазони, на яких вони можуть бути виявлені як радарми, так і електрооптичними/інфрачервоними (ЕО/ІЧ) датчиками. Їх розмір також обмежує їх висоту і швидкість, оскільки вони зазвичай літають на висоті менше 125 метрів і зі швидкістю менше 50 метрів за секунду.

Розвиток технологій побудови малорозмірних безпілотних літальних апаратів (БПЛА) призвів до появи нового класу загроз, таких як: шпівонаж, тероризм, транспортування заборонених вантажів, ускладнення повітряного руху, пошкодження майна. Тому вирішення задачі нейтралізації загроз від використання малорозмірних БПЛА і створення відповідних систем захисту є надзвичайно актуальним.

Важливим класом систем виявлення малорозмірних БПЛА є системи відеоспостереження. Вони використовують камери високої роздільної здатності, а також спеціальне програмним забезпеченням. Такі камери автоматично фокусуються на цілях, якими є малорозмірні БПЛА, і у випадку їх розпізнавання за допомогою бази даних візуальних сигнатур, записують відео приближення БПЛА до об'єкту, та активують відповідну сигналізацію.

Але відеокамера вимірює лише кутові координати цілі і не дозволяє визначити її положення в просторі.

Поєднавши систему відеоспостереження та FMCW далекомір ми отримуємо комбіновану систему, що здатна виявляти, стежити, та точно локалізувати БПЛА в просторі.

В роботі синтезовано квазіоптимальний алгоритм траєкторної фільтрації БПЛА за даними системи відеоспостереження і FMCW далекоміру.

Метою дослідження являється підвищення ефективності локалізації БПЛА в просторі з використанням системи відеоспостереження та FMCW далекоміру, шляхом розробки алгоритму траєкторної фільтрації з розпізнаванням різних видів його руху.

Об’єкт дослідження – процес траєкторної фільтрації БПЛА за даними системи відеоспостереження та FMCW далекоміру.

Предмет дослідження – методи стеження та оцінки положення БПЛА за даними відеоспостережень та FMCW далекоміру.

Для виконання заданої задачі потрібно виконати такі завдання:

- провести аналіз існуючих методів стеження за БПЛА;
- синтезувати оптимальний алгоритм стеження та оцінки положення БПЛА в просторі;
- оцінити ефективність розробленого алгоритму шляхом статистичного моделювання на ЕОМ;
- оцінити можливості практичної реалізації розробленого алгоритму.

1 АКТУАЛЬНІСТЬ. ОГЛЯД ІСНУЮЧИХ РІШЕНЬ. РОЗРОБКА ТА АНАЛІЗ ТЕХНІЧОГО ЗАВДАННЯ

Безпілотні літальні апарати (БПЛА) блискавично входять у наше повсякденне життя, вирішуючи складні завдання, яких раніше не було. Найактивнішим сектором у розробці БПЛА є військова сфера, наукові дослідження, сільське господарство та рекреаційне використання. Сучасні інтелектуальні автономні технології, які використовуються в БПЛА, збирають інформаційні дані за формування пошуково-рятувальних, військових місій, операцій з протипожежної та медичної допомоги.

Варто згадати про зростаючу загрозу «безпілотника-камікадзе» (типу БПЛА, начиненого летальною вибухівкою та осколками) , яка стає широко доступною для створення та не демонструє практичних технологічних рішень для виявлення терористичних атак, особливо в густонаселених і густонаселених міських районах. Зазвичай такі БПЛА виготовляються вручну та використовують спеціалізоване обладнання дистанційного керування, яке не виробляється масово і його легко виявити в міських районах.

Нещодавня ситуація, заснована на загрозах БПЛА, показує, що БПЛА можуть виконувати руйнівні атаки на віддалену інфраструктуру та справляти великий негативний вплив на життя населення.

Загалом усі БПЛА потребують рушійної енергії для зльоту, польоту та посадки. Двигуни тяги використовуються для живлення будь-якого БПЛА для польоту, за винятком планерів і систем легших за повітря.

БПЛА планерної конструкції не потребують рушійної енергії, оскільки вони можуть виконувати свої задачі, використовуючи висхідні повітряні потоки, але їх потрібно гасити або буксирувати в небі. Основні типи силової установки та двигуна БПЛА:

– Двигуни, що працюють на хімічній або ядерній енергії (термічний двигун; двигун з ядерною реакцією; двигун внутрішнього згорання; двигун внутрішнього згорання).

– Двигуни з кінетичною енергією (двигуни з пружинним навантаженням)
– Електричні двигуни (електричний двигун, що працює від акумулятора; гібридна електродвигун на рідкому паливі й електростанції; гібридний електричний двигун від сонячної енергії та батареї; електричний двигун, що працює на водневих елементах). Електричні двигуни (електричний двигун, що працює від акумулятора; гібридна електродвигун на рідкому паливі й електростанції; гібридний електричний двигун від сонячної енергії та батареї; електричний двигун, що працює на водневих елементах).

Аналізуючи історію та перспективи розвитку конструкції малого БПЛА, очевидно, що для бойової роботи БПЛА обов'язковими є дві основні складові: джерело енергії та силова установка тяги. Багато інших компонентів роботи БПЛА стрімко розвиваються і швидко перетворюються на застарілі. Найпоширенішою тяговою силовою установкою малого БПЛА, що випускається серійно, є електродвигуни. БПЛА з електричним приводом завдяки високому ККД двигуна та можливостям щільного накопичення енергії можуть бути виготовлені в невеликих розмірах і вазі, що робить їх виявлення складним за допомогою звичайних методів виявлення БПЛА. БПЛА вже здатні виконувати повністю автономні бої, без необхідності радіозв'язку та GPS-навігації. Їх легко запрограмувати на подолання будь-якої активної радіочастотної системи виявлення радарів і виконання попередньо запрограмованої місії в режимі повного радіомовчання.

За різноманітністю конструкції існує чотири основні типи безпілотних літальних апаратів [2]:

- Мультикоптерні дрони являють собою літаючу платформу з безколекторними двигунами, з пропелерами. Вони мають можливості для вертикальних злетів або зависання над об'єктом і є найдешевшою та найдоступнішою версією БПЛА.
- БПЛА з нерухомим крилом реалізують структуру літака й ідеально підходять для далеких операцій.

- Однороторні дрони - безпілотні вертольоти набагато ефективніші, ніж багатороторні версії. Вони мають більш тривалий час польоту і можуть навіть приводитися в дію двигунами внутрішнього згоряння.

- Відомі також гібридні дрони.

За розмірами БПЛА поділяють на такі основні групи [3]:

Таблиця 1.1 – Класифікація БПЛА

Клас	Призначення
Мініатюрні БПЛА	Майже повністю невидимі у будь-якому спектрі та безшумні. Вони використовуються для розвідки в умовах, коли потрібна максимальна скритність.
Надмалі БПЛА	Забезпечують розвідувальною інформацією невеликі підрозділи, групи спеціального призначення та інші підрозділи, які автономно виконують бойові завдання. Вони залежать від злітної маси - 2-10 кг, корисного навантаження - 0,5-2 кг, часу польоту - до години.
Малі БПЛА	Ведуть розвідку, наводять вогневу зброю, регулюють вогонь, патрулюють територію, виконують ударні місії та борються з дронами противника. Такі дрони мають масу 10-50 кг, корисне навантаження 15-20 кг, радіус дії до 60 км, час польоту 4-5 годин.

У міру поширення дронів дедалі актуальнішим стає питання виявлення БПЛА в повітрі. Кожен метод [4] відрізняється надійністю, діапазоном, умовами роботи, точністю та багатьма іншими параметрами. Основними методами виявлення БПЛА, які розвиваються є:

- Аналіз сліду акустичного сигналу

Досить актуальним стає використання акустичних каналів для виявлення БПЛА. Акустичні сенсори дають змогу наземним засобам здійснювати пошук

і виявлення БПЛА в пасивному режимі, знижуючи таким чином імовірність визначення противником власних позицій. Тому модифікація наявних акустичних систем пошуку або створення нових систем може забезпечити надійний метод виявлення БПЛА.

Мікрофон або мікрофонна матриця отримує акустичний сигнал двигунів тяги БПЛА, який у реальному часі аналізується та порівнюється з відомою формою сигналу БПЛА. Якщо шаблони сигналів збігаються, БПЛА виявлено.

– Аналіз оптичного сліду

Для захоплення навколишнього зображення використовується одна відеокамера або група камер. Дані зображення обробляються в режимі реального часу для розпізнавання траєкторії бою.

PTZ-камери можуть використовуватися в поєднанні з іншими технологіями виявлення безпілотників для створення і передачі візуального запису інциденту з безпілотником.

Оптичне виявлення БПЛА дуже сильно залежить від чинників навколишнього середовища. Збільшення дальності виявлення досягається за рахунок звуження поля зору, зменшення зони огляду і збільшення часу пошуку.

– Теплова камера аналізу теплової сигнатури

Або матриця камер використовується для захоплення навколишнього зображення. Дані зображення обробляються в режимі реального часу для розпізнавання траєкторії бою.

– Аналіз радіолокаційної сигнатури БПЛА

Більшість безпілотників використовують радіочастотний (РЧ) спектр для зв'язку зі своїми операторами. Для забезпечення зв'язку між приймачем і передавачем вони об'єднуються в пару зі спеціальними RFID-чіпами, які не дають іншим пристроям, що працюють на тій же частоті, обігнати безпілотник.

Радіочастотне виявлення безпілотників використовує радіочастотні (РЧ) датчики, які пасивно прослуховують і контролюють частоти від 70 МГц до 6 ГГц для передачі лінії зв'язку між безпілотником і пілотом (приймачем), щоб

визначити місцезнаходження безпілотної, а в деяких випадках і місцезнаходження пілота.

Виявляється за наявністю їх радіолокаційної сигнатури, яка генерується, коли корпус БПЛА стикається з радіочастотними імпульсами, випромінюваними елементом виявлення. Сигнал, відбитий від БПЛА, обробляється в реальному часі для виявлення траєкторії польоту.

– Радіозв'язок

Дрони можуть бути виявлені шляхом прийому та аналізу радіосигналів: ліній зв'язку та управління, радіолокаційних висотомірів, постановників активних перешкод і радіолокаційних станцій. Однак цим методом можна встановити лише напрямок на БПЛА, причому точність визначення підвищується при збільшенні часу спостереження.

Деякі низькочастотні лінії зв'язку можуть бути виявлені на значних дистанціях. Випромінювання бортових радіолокаційних станцій (РЛС) і встановлення активних перешкод БПЛА можуть бути виявлені на ще більших дистанціях. Цей метод вимагає мінімального обладнання і дає змогу швидко визначити пеленг цілі при подальшій видачі цілевказівок на засоби оптичного або ІЧ спостереження. Пошук БПЛА за допомогою активних радіолокаційних станцій досить продуктивний, оскільки вони мають відносно великий імпульсний обсяг пошуку і значну дальність виявлення.

Істотним недоліком радіолокаційного методу є те, що більшість БПЛА виготовляють із композитних матеріалів, які досить погано відбивають електромагнітні хвилі. Радіохвилі проникають через поверхню апарата і лише частково відбиваються від неї.

Ще однією проблемою є те, що через невеликий розмір безпілотних літальних апаратів, з урахуванням обмежень у смузі пропускання і, що важливо, через високу вартість, радіолокаційні радарні спостереження не можуть використовуватися часто, особливо коли необхідно розгорнути мережу радарів, щоб гарантувати повне покриття контрольованої зони.

Структура та параметри акустичного сигналу, що випромінюється БПЛА, залежать від виду об'єкта, його форми, кількості двигунів, кількості несучих гвинтів і т.д. У свою чергу акустична локаційна станція (содар) повинна будуватися з урахуванням особливостей структури сигналу, що приймається.

Спектр радіочастот приймається та аналізується в режимі реального часу для модуляції, відео, телеметрії та декодування даних керування. Пакети відомих комунікаційних даних декодуються, і цінні дані витягуються.

– Комбіновані методи

Будь-який з кількох методів виявлення в комбінації використовується для підвищення точності, стійкості до відмов, стабільності, позиціонування та дальності виявлення БПЛА.

Переваги та недоліки існуючих методів [5] зведемо до таблиці 1.2.

Таблиця 1.2 — Переваги та недоліки

Метод	Переваги	Недоліки
Виявлення радіочастот	Велика дальність Може визначати місцезнаходження контролера та безпілотників	Сигнал управління повинен бути радіочастотним
За допомогою радару	Може виявляти безпілотники без радіосигналу Велика дальність	Важко виявити низьколітаючі безпілотники Проблеми з розрізненням від інших малих об'єктів

Акустичний	Може виявляти безпілотники без радіосигналу	Дуже обмежений діапазон Схильний до завад у шумному середовищі
Візуальний	Може виявляти безпілотники без радіосигналу Низька вартість	Більшість малих безпілотників виробляють дуже мало тепла Обмежений діапазон
Інфрачервоне випромінювання	Може виявляти безпілотники без радіосигналу	Проблеми з розрізненням в нічний час Обмежений діапазон

Радіолокатори безперервного (CW) і безперервного випромінювання з частотною модуляцією (FMCW) наразі являють собою найпривабливіше і найекономічніше рішення цієї проблеми.

Сигнал FMCW складається з лінійно модульованої безперервної радіохвилі, переданої в бажаному напрямку. Ці види сигналів відрізняються від CW сигналу, в якому робоча частота не змінюється під час передачі. У більшості випадків виявлення цілі спрямоване на виявлення наявності об'єктів, які не є постійними в спостережуваному сценарії, тобто об'єктів, які можуть динамічно з'являтися або зникати. Індикатор рухомої цілі - це режим роботи радара, що дає змогу системі розрізняти ціль як серед перешкод, так і серед нерухомих об'єктів у зоні спостереження. Ця властивість заснована на ефекті Доплера, оскільки нерухомі цілі не викликають будь-якого доплерівського зсуву частоти в спостережуваних сигналах.

Кожен з відомих методів має свої переваги, недоліки та обмеження, характеризується деякою областю можливостей, яка визначає безліч вимірюваних параметрів з відповідними точнісними характеристиками і просторово-часовим дозволом, діапазоном дальностей і т. д. Оскільки можливості різних методів відрізняються, спільне використання систем різного виду веде до підвищення ефективності їх функціонування.

Виявлення безпілотників, які не мають радіочастотної передачі, можна здійснити за допомогою недорогих датчиків камери на основі алгоритмів комп'ютерного зору. Здатність виявлення та класифікації найвища, коли ціль видно, а датчики камери мають перевагу, оскільки дають формальну візуальну перевірку того, що виявлений об'єкт є дроном, одночасно надаючи додаткову візуальну інформацію, таку як модель дрона, розміри та корисне навантаження, яке не можуть забезпечити інші системи виявлення дронів. Середня дальність виявлення, хороша локалізація, доступна ціна та легка інтерпретація людиною досягаються за допомогою технологій візуального виявлення безпілотників, які використовують зображення зон спостереження з камер.

В оптичному діапазоні було розроблено різноманітні методи для виявлення малогабаритних БПЛА:

- Активні (анагліфічний, лазерний)
- Пасивні (стереоскопічний, візуальний)

Анагліфічний метод може бути реалізовано на основі камер відеоспостереження, джерел освітлення, анагліфічних фільтрів, блоку попередньої обробки зображень, процесору аналізу та обробки зображень, каналу виведення інформації та блоку живлення.

Дані методи реалізуються на основі пристроїв, що забезпечують виявлення та визначення БПЛА в просторі.

Слід також брати до уваги необхідність кругового огляду в 360° , можливість паралельної роботи в оптичному, звуковому та радіолокаційному

діапазоні електромагнітних хвиль, та розміщення системи на рухомих об'єктах для подальшого прицілювання за необхідності.

Для наших цілей ми поєднаємо камеру відеоспостереження, яка буде виявляти та слідкувати за ціллю, з FMCW далекоміром, який буде точно визначати координати цілі в просторі.

Також слід зауважити, що враховуючи геометричні розміри БПЛА та використання сучасних засобів візуального захисту досліджуваного БПЛА, максимальна дальність його візуального виявлення становить близько 1 км.

2 ОПИС ВІДЕОПОСТЕРЕЖЕННЯ

В системах відеоспостереження використовують різні методи виявлення та супроводження цілей [6]. Також існують системи з декількома слідкуючими камерами.

В нашому випадку система відеоспостереження буде складатись з однієї слідкуючої PTZ камери, яка має огляд в 360 градусів.

Блок схема будови системи зображена на Рис. 2.1

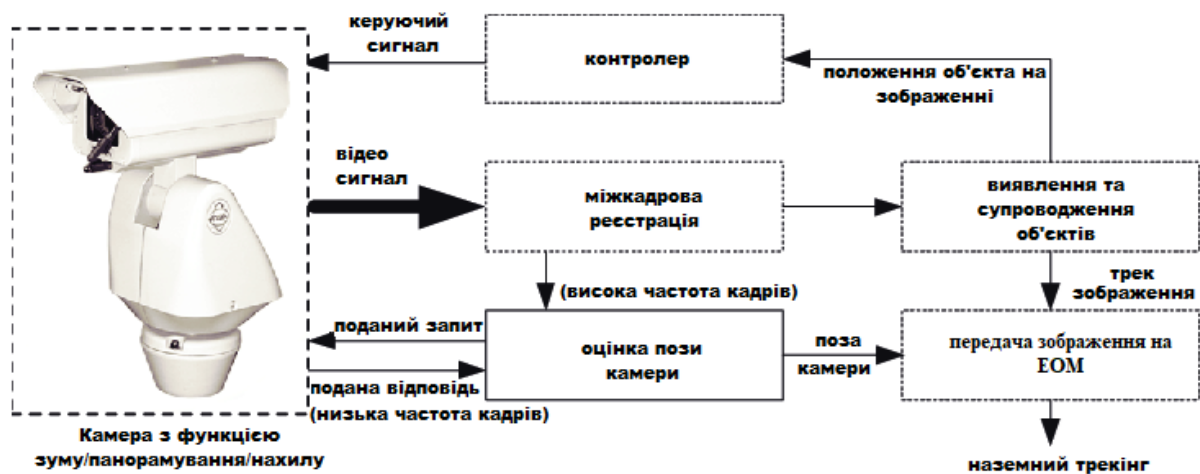


Рисунок 2.1— блок схема роботи системи відеоспостереження

Камера з функцією панорамування, нахилу і зуму (PTZ) є потужним інструментом, що забезпечує як широку загальну зону покриття, так і високу роздільну здатність зображення в межах одного приладу. Цього неможливо досягти за допомогою традиційної стаціонарної камери [7].

2.1 Опис роботи системи

У системі відеоспостереження та автоматичного відстеження об'єктів відеоспостереження та автоматичного стеження за об'єктами, наша камера може передавати зашумлені метадані PTZ-зображення з частотою не більше 1 Гц, а також отримувати зображення з частотою 15 Гц. З іншого боку, щоб

щоб цікавий об'єкт завжди знаходився в центрі зображення, система повинна посилати на камеру керуючі сигнали в кожному кадрі на основі поточного оціночного PTZ і прогнозованого положення і швидкості об'єкта.

Тому одним з наших завдань є оцінка параметрів значень PTZ на кожного кадру на основі метаданих та F2F-інформації.

Для цього, окрім використання низької частоти дискретизації метаданих, використовується ЕКФ, який використовує високочастотні F2F-відповідності для підвищення точності оцінювання [8].

Система призначена для виявлення і супроводу рухомих об'єктів і передачі їх траєкторій в наземній системі координат.

Кадри зображення, що надходять стабілізуються за допомогою F2F реєстрації. Реєстрація інформація та виміри положення камери надходять в модуль оцінки положення камери для оцінки PTZ-значення для кожного кадру зображення. Візуальне виявлення об'єктів і стеження за об'єктом здійснюються на стабілізованому відеосигналі. Отримані дані про положення об'єктів передаються в контролер для управління кутами панорамування і/або нахилу камери таким чином, щоб обраний об'єкт завжди був по центру зображення, а також для управління значенням масштабування, щоб об'єкт мав достатню деталізацію. Положення об'єктів на зображенні трансформуються в наземну систему координат за допомогою розрахункового значення PTZ. Наземні треки використовуються для візуалізації треків на плановій карті, а також для об'єднання з іншими PTZ-камерами (якщо вони наявні) та радіолокаційною системою [9].

2.1.1 Реєстрація зображень

За основу візьмемо камеру Pelco Esprit™ яка має такі характеристики:

- 23-кратний оптичний зум.
- Максимальна швидкість панорамування/нахилу 40/20 градусів за секунду.

- Відео знімається з роздільною здатністю 320 на 240 з швидкістю 15 кадрів в секунду.
- Метадані запитуються з частотою 1 Гц.

Зовнішній вигляд PTZ камери зображений на Рис. 2.2.



Рисунок 2.2 — PTZ-камера

PTZ-камера приблизно обертається вздовж центру проекції камери. Тому рух зображення можна описати перетворенням плоскої перспективи (тобто гомографією).

Проте афінне перетворення є достатнім для більшості ситуацій і є більш стійким, ніж гомографія завдяки меншій кількості параметрів. Ми обчислюємо афінне перетворення між двома послідовними зображеннями кадрів, використовуючи комбінацію методів, заснованих на особливостях, та прямого методу. Спочатку виділяються характерні точки (кути Харріса) на обох кадрах. Їх початкові відповідності встановлюються за допомогою міри SSD (сума квадратів різниць). Потім за допомогою RANSAC [10] обчислюється

афінне перетворення з початкових відповідностей яке містить значну кількість викидів. Це афінне перетворення далі локально уточнюється прямим методом [11].

На рисунку 3.2 показані деякі відповідні точки в двох послідовних кадрах.

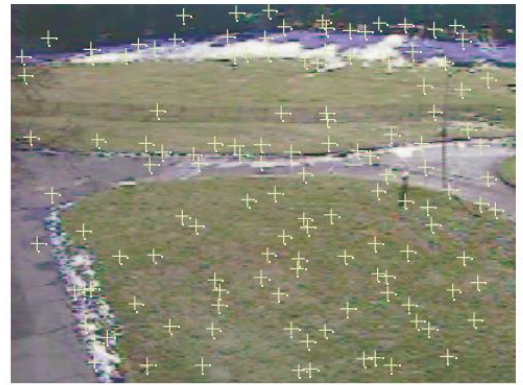
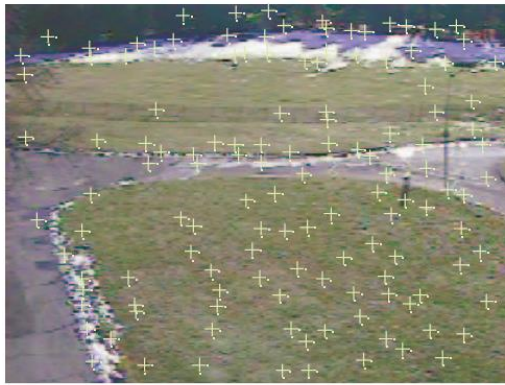


Рисунок 2.3 — деякі відповідні точки, отримані за допомогою зіставлення між двома послідовними кадрами

2.1.2 Одночасна оцінка орієнтації та фокусної відстані

Оскільки фільтр масштабування не залежить від фільтра РТ, його можна оцінити без інформації про панорамування та нахил. Однак стан РТ-фільтра не може бути оцінений без інформації про масштабування.

Отже, як показано на рисунку 3.4, пропонуємо алгоритм для одночасної оцінки параметрів РТЗ з метаданих та/або F2F вимірювань. Кожного разу, коли вимірювання надходить в поточний час, алгоритм спочатку оцінює стан масштабування на основі попередньої оціненої інформації про масштабування та вимірювання за допомогою алгоритму KF; потім, використовуючи оцінений поточний стан масштабування та поточне вимірювання, використовується ЕKF для оновлення поточного стану РТ. Обидва фільтри ініціалізуються метаданими [12].

3 ОПИС FMCW ДАЛЕКОМІРУ В СИСТЕМІ ТРАЄКТОРНОЇ ФІЛЬТРАЦІЇ РУХУ БПЛА

Лазерна далекометрія як метод визначення відстаней до об'єктів ґрунтується на зондуванні та обробці відбитого сигналу. Найпоширенішими є далекоміри, зондувальний сигнал яких піддається лінійній частотній модуляції (FMCW). Подібні ЛЧМ-далекоміри мають найкращі характеристики на малих і середніх відстанях [13].

Генерований опорний ЛЧМ-періодичний сигнал перетворюється на джерелі випромінювання на сигнал, що зондує об'єкт. Відбитий сигнал через час проходження сигналу до об'єкта і назад потрапляє на приймач. Змішувач частот переданого і прийнятого ЛЧМ-сигналів у послідовному з'єднанні з низькочастотним фільтром дає змогу отримати фрагмент періодичного сигналу, частота якого пропорційна відстані до об'єкта. Після дискретизації дальність до об'єкта алгоритмічно визначається на основі розрахунку частоти сигналу за набором його дискретних значень.

Структура FMCW-далекоміра показана на рис.3.1.

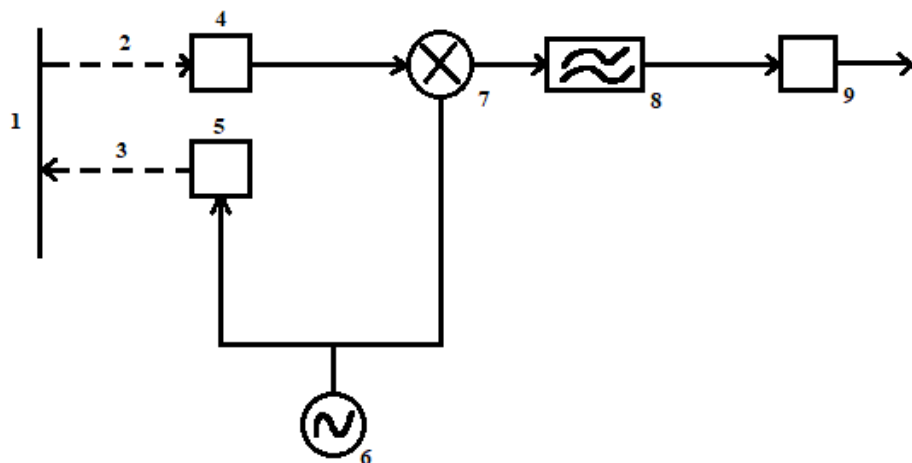


Рисунок 3.1 — Структура FMCW-далекоміра. 1 - об'єкт, 2 - відбитий сигнал, 3 - зондувальний сигнал, 4 - приймач, 5 - джерело, 6 - генератор сигналу, 7 - змішувач, 8 - фільтр низьких частот, 9 – обчислювач

Основою алгоритмів побудови траєкторної фільтрації являється модель руху цілі [14]. В якості базової бажано використовувати прямокутну систему координат, її основна перевага заключається в тому, що рівномірний прямолінійний рух цілі описується лінійними диференціальними або різницевиими рівняннями першого порядку, тоді як в інших системах координат рівняння руху нелінійні, що ускладнює обробку і може призвести до додаткових помилок [4].

Рух БПЛА з різними видами маневру в прямокутній системі координат можна описати стохастичною динамічною системою з випадковою структурою виду [5]:

$$\mathbf{u}_w(k) = \mathbf{F}_j \mathbf{u}_w(k-1) + \mathbf{G}_j \boldsymbol{\omega}_w(k), \quad j = \overline{1, M}, \quad (3.1)$$

яка враховує три основних види руху $M=3$: зависання $j=1$, майже рівномірний рух $j=2$, рух з маневром $j=3$. Вектор стану $\mathbf{u}_w^T(k) = (x_w(k), \dot{x}_w(k), \ddot{x}_w(k), y_w(k), \dot{y}_w(k), \ddot{y}_w(k), z_w(k), \dot{z}_w(k), \ddot{z}_w(k))$ включає координати положення, швидкості і прискорення по осях місцевої СК.

Матриці, що входять в модель руху (5.1) мають вид

$$\mathbf{F}_j(k, k-1) = \begin{bmatrix} \mathbf{F}_j^b & 0 & 0 \\ 0 & \mathbf{F}_j^b & 0 \\ 0 & 0 & \mathbf{F}_j^b \end{bmatrix}, \quad \mathbf{G}_j(k) = \begin{bmatrix} \mathbf{G}_j^b & 0 & 0 \\ 0 & \mathbf{G}_j^b & 0 \\ 0 & 0 & \mathbf{G}_j^b \end{bmatrix}, \quad (3.2)$$

де $\mathbf{F}_j^b, \mathbf{G}_j^b, j = \overline{1, 3}$ мають вид

$$\mathbf{F}_1^b = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{F}_2^b = \begin{bmatrix} 1 & T & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{F}_3^b = \begin{bmatrix} 1 & T & \frac{T^2}{2} \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{G}_1^b = \begin{bmatrix} a_1 \cdot T \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{G}_2^b = \begin{bmatrix} \frac{a_2 \cdot T^2}{2} \\ a_2 \cdot T \\ 0 \end{bmatrix}, \quad \mathbf{G}_3^b = \begin{bmatrix} \frac{a_3 \cdot T^3}{6} \\ \frac{a_3 \cdot T^2}{2} \\ a_3 \cdot T \end{bmatrix}; \quad (3.3)$$

a_{1w}, a_{2w}, a_{3w} — СКВ випадкових флуктуацій швидкості, прискорення і швидкості зміни прискорення БПЛА для першого, другого і третього типів руху, відповідно.

Для опису типу структури моделі руху БПЛА (3.1), що відповідає певному виду маневру, застосовується змінна перемикачання $a_j(k)$, $j = \overline{1, M}$, $M = 3$. Вона відноситься до класу кіл Маркова, з матрицею ймовірностей переходів $\Pi_{i,j}(k, k-1)$, $i, j = \overline{1, M}$ і початковими ймовірностями $w_i(0)$, $i = \overline{1, M}$, що дозволяє врахувати переходи між різними видами руху БПЛА в випадкові моменти часу.

Система відеоспостереження і FMCW далекомір вимірюють азимут $\beta_{\text{И}}(k)$, кут місця $\varepsilon_{\text{И}}(k)$ і радіальну дальність $r_{\text{И}}(k)$ до БПЛА та радіальну швидкість $\dot{r}_{\text{И}}(k)$ в сферичній системі координат. Враховуючи, що при побудові моделі руху цілі використано прямокутну систему координат, рівняння спостереження можна представити у вигляді:

$$\beta_{\text{И}}(k) = \arctg\left(\frac{y(k)}{x(k)}\right) + v_{\beta}(k); \quad (3.4)$$

$$\varepsilon_{\text{И}}(k) = \arctg\left(\frac{z(k)}{\sqrt{x^2(k) + y^2(k)}}\right) + v_{\varepsilon}(k); \quad (3.5)$$

$$r_{\text{И}}(k) = \sqrt{x^2(k) + y^2(k) + z^2(k)} + v_r(k); \quad (3.6)$$

$$\dot{r}_{\text{И}}(k) = \frac{x(k)\dot{x}(k) + y(k)\dot{y}(k) + z(k)\dot{z}(k)}{\sqrt{x^2(k) + y^2(k) + z^2(k)}} + v_{\dot{r}}(k) \quad (3.7)$$

де $v_\beta(k)$, $v_\varepsilon(k)$, $v_r(k)$, $v_{\dot{r}}(k)$ - некорельовані гаусівські помилки вимірювання азимуту, кута місця, радіальної дальності і радіальної швидкості БПЛА в полярній СК з нульовими математичними очікуваннями та дисперсіями складових, σ_β^2 , σ_ε^2 , σ_r^2 , $\sigma_{\dot{r}}^2$.

Вимірювані координати положення БПЛА в прямокутній і сферичній системах координат зв'язані нелінійними рівняннями

$$\begin{aligned}x_{\text{И}}(k) &= r_{\text{И}}(k) \cos \varepsilon_{\text{И}}(k) \cos \beta_{\text{И}}(k); \\y_{\text{И}}(k) &= r_{\text{И}}(k) \cos \varepsilon_{\text{И}}(k) \sin \beta_{\text{И}}(k); \\z_{\text{И}}(k) &= r_{\text{И}}(k) \sin \varepsilon_{\text{И}}(k),\end{aligned}\tag{3.8}$$

де $x_{\text{И}}(k)$, $y_{\text{И}}(k)$, $z_{\text{И}}(k)$ - вимірювані координати положення БПЛА в прямокутній системі координат.

Зв'язок між координатами положення БПЛА в сферичній і прямокутній системах координат показаний на Рис.3.2.

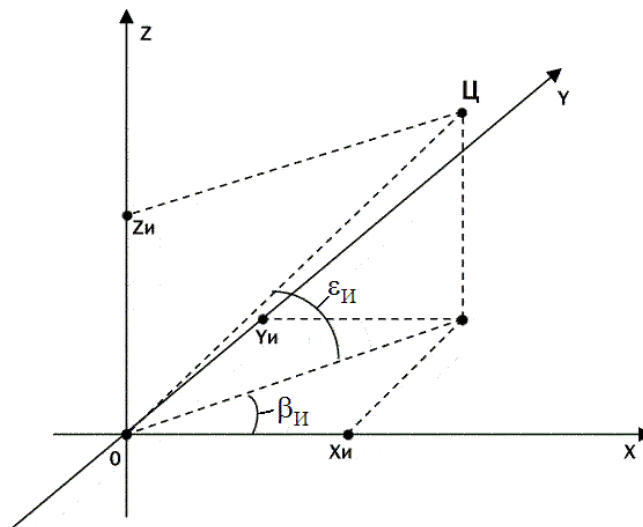


Рисунок 3.2 — зв'язок між координатами положення БПЛА в сферичній і прямокутній системах координат

Виконавши лінеаризацію виразів (4.8), рівняння спостереження системи відеоспостереження і далекоміра в прямокутній системі координат можна привести до вигляду [6]:

$$x_{\text{И}}(k) = x(k) + v_x(k);$$

$$\begin{aligned}
y_{\text{И}}(k) &= y(k) + v_y(k), \\
z_{\text{И}}(k) &= z(k) + v_z(k); \\
\dot{r}_{\text{И}}(k) &= \frac{x(k)\dot{x}(k) + y(k)\dot{y}(k) + z(k)\dot{z}(k)}{\sqrt{x^2(k) + y^2(k) + z^2(k)}} + v_{\dot{r}}(k),
\end{aligned} \tag{3.9}$$

де $v_x(k), v_y(k), v_z(k)$ - помилки вимірювання положення БПЛА в прямокутній СК.

У векторному вигляді рівняння спостереження можна представити у вигляді

$$u_{\text{И}}(k) = h(u(k)) + v(k), \tag{3.10}$$

де $u_{\text{И}}(k) = (x_{\text{И}}(k), y_{\text{И}}(k), z_{\text{И}}(k), \dot{r}_{\text{И}}(k))$ - вектор спостереження, що включає, виміряні координати положення БПЛА в прямокутній системі координат і радіальну швидкість; $h(u(k))$ - нелінійна векторна функція, значенням якої є вектор істинних координат положення у прямокутній СК та радіальної швидкості досліджуваної цілі $(x(k), y(k), z(k), \dot{r}(k))^T$; $v(k) = (v_x(k), v_y(k), v_z(k), v_{\dot{r}}(k))^T$ - вектор помилок вимірювання положення БПЛА у прямокутній СК та радіальній швидкості цілі з нульовим математичним очікуванням та кореляційною матрицею $R(k)$.

Кореляційна матриця помилок вимірювання системи відеоспостереження і далекоміра в прямокутній системі координат має вигляд

$$R(k) = \begin{bmatrix} \sigma_x^2(k) & \sigma_{xy}^2(k) & \sigma_{xz}^2(k) & 0 \\ \sigma_{xy}^2(k) & \sigma_y^2(k) & \sigma_{yz}^2(k) & 0 \\ \sigma_{xz}^2(k) & \sigma_{yz}^2(k) & \sigma_z^2(k) & 0 \\ 0 & 0 & 0 & \sigma_{\dot{r}}^2(k) \end{bmatrix}. \tag{3.11}$$

При цьому її елементи визначаються по формулах

$$\sigma_x^2(k) = \sigma_r^2 \cos^2 \varepsilon(k) \cos^2 \beta(k) + r^2(k) \sigma_\varepsilon^2 \cos^2 \beta(k) \sin^2 \varepsilon(k) + r^2(k) \sigma_\beta^2 \sin^2 \beta(k) \cos^2 \varepsilon(k); \quad (3.12)$$

$$\sigma_y^2(k) = \sigma_r^2 \cos^2 \varepsilon(k) \sin^2 \beta(k) + r^2(k) \sigma_\varepsilon^2 \sin^2 \beta(k) \sin^2 \varepsilon(k) + r^2(k) \sigma_\beta^2 \cos^2 \beta(k) \cos^2 \varepsilon(k); \quad (3.13)$$

$$\sigma_z^2(k) = \sigma_r^2 \sin^2 \varepsilon(k) + \sigma_\varepsilon^2 r^2(k) \cos^2 \varepsilon(k); \quad (3.14)$$

$$\sigma_{xy}^2(k) = 0.5 \sin 2\beta(k) (\cos^2 \varepsilon(k) \sigma_r^2 - r^2(k) \sigma_\beta^2 + \sin^2 \varepsilon(k) r^2(k) \sigma_\varepsilon^2); \quad (3.15)$$

$$\sigma_{xz}^2(k) = 0.5 \sin 2\varepsilon(k) \sin \beta(k) (\sigma_r^2 - r^2(k) \sigma_\varepsilon^2(k)); \quad (3.16)$$

$$\sigma_{yz}^2(k) = 0.5 \sin 2\varepsilon(k) \cos \beta(k) (\sigma_r^2 - r^2(k) \sigma_\varepsilon^2(k)), \quad (3.17)$$

де σ_r^2 , σ_β^2 , σ_ε^2 - дисперсії помилок вимірювання координат цілі в сферичній системі координат;

$r(k)$, $\beta(k)$, $\varepsilon(k)$ - координати цілі в сферичній системі координат.

Як випливає з виразів (3.12)...(3.17), компоненти кореляційної матриці помилок вимірювання в прямокутній системі координат залежать від поточного положення БПЛА. Тому вони повинні обчислюватися на кожному кроці спостережень.

4 АНАЛІЗ ТОЧНОСНИХ ХАРАКТЕРИСТИК ВИМІРЮВАННЯ КООРДИНАТ БПЛА ЗА ДАНИМИ СИСТЕМИ ВІДЕОСПОСТЕРЕЖЕННЯ І FMCW ДАЛЕКОМІРУ

Аналіз отриманих точносних характеристик системи відеоспостереження і FMCW далекоміру (3.12)...(3.17) виконаний роздільно в горизонтальній і вертикальній площинах. При визначенні точносних характеристик системи відеоспостереження і FMCW далекоміру на площині XOY кут місця ε вважався рівним нулю. Ціль рухається по колах з радіусами 1000м, 550м, 100м, в центрі яких знаходиться комплексна система. Помилка вимірювання системи $\sigma_r=15\text{м}$, $\sigma_\varepsilon=0.3$ град., $\sigma_\beta=0.3$ град. На рис 4.1,а,б суцільними лініями в полярній системі координат (R, β) показана залежність СКВ помилок визначення координат БПЛА по осях декартової системи координат X, Y від кута β , розраховані по формулах (3.12), (3.13). На рис.4.1,в суцільною лінією показана залежність модуля коефіцієнта взаємної кореляції $|\gamma_{xy}|$ помилок визначення координат цілі по осях декартової системи координат X, Y залежно від кута β , який визначається на основі виразу (3.15). При цьому дальність до БПЛА дорівнює 1000м.

Як впливає з отриманих результатів, СКВ помилок визначення координат цілі σ_x, σ_y по осях декартової системи координат X, Y і коефіцієнт взаємної кореляції γ_{xy} залежать від кута β . Основний внесок в помилки визначення координат цілі в декартовій системі координат на великих відстанях вносить помилка вимірювання азимута. Відповідно СКВ помилки σ_x досягає максимального значення в області кутів $\theta=[\pi/2, 3\pi/2]$, а СКВ помилки σ_y - в області кутів $\theta=[0, \pi]$. Із зменшенням дальності до цілі СКВ помилок визначення координат цілі σ_x, σ_y зменшується. На малих відстанях основний внесок в помилки вимірювання координат цілі в прямокутній системі координат вносить помилка вимірювання дальності.

Як випливає з рис.4.1,в, значення коефіцієнта кореляції носить нерівномірний характер. При цьому він досягає нуля при $\beta = [0, \pi/2, \pi, 3\pi/2]$. При інших відстанях до цілі характер зміни коефіцієнта взаємної кореляції γ_{xy} не змінюється. В цілому існує значна кореляція між помилками вимірювань в горизонтальній площині, що необхідно враховувати при розробці алгоритмів вторинної обробки радіолокаційної інформації.

Залежність СКВ помилок визначення координат БПЛА та залежність коефіцієнта взаємної кореляції по осях X, Y показана на рисунку 4.1.

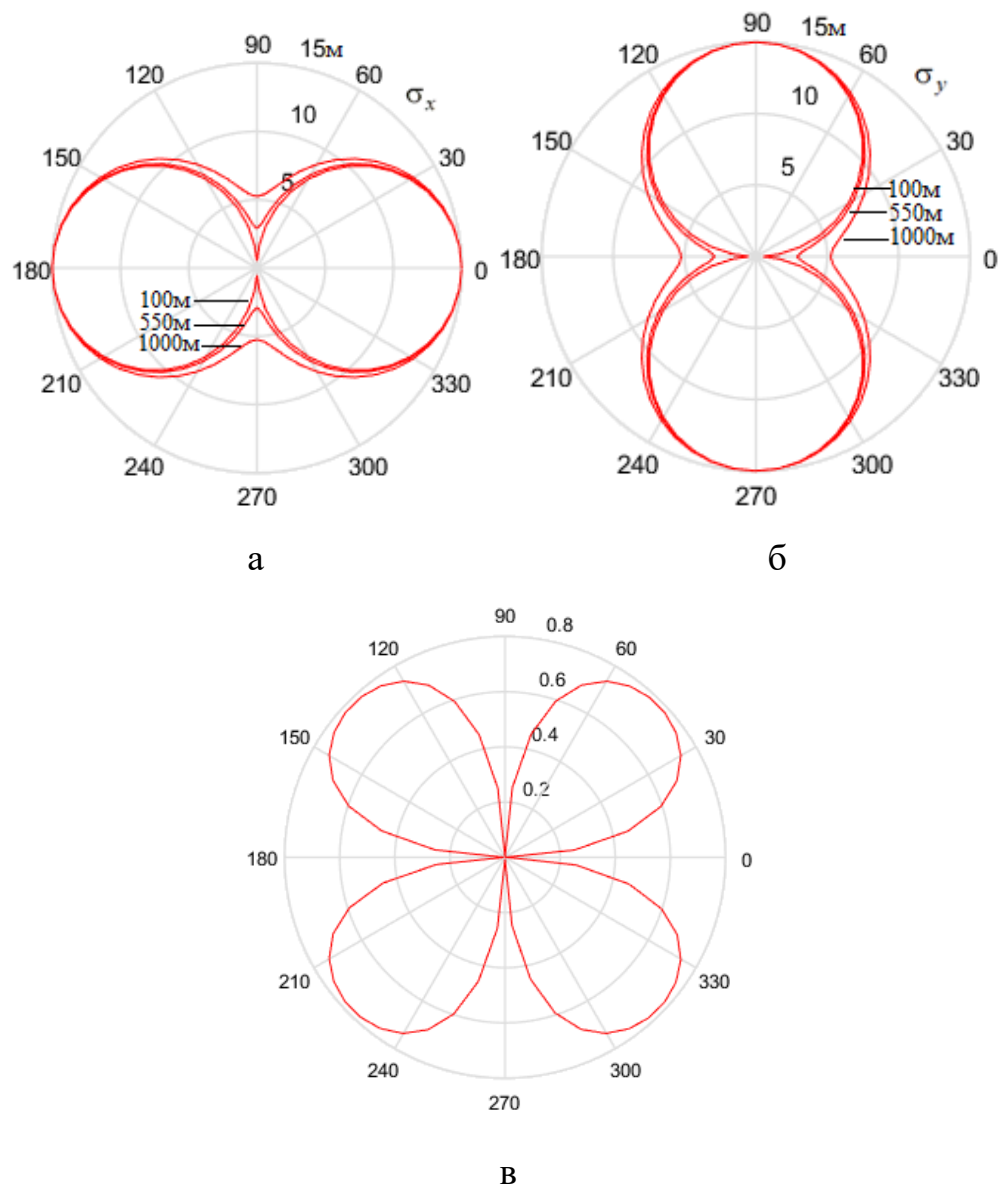


Рисунок 4.1 а,б, —залежність СКВ помилок визначення координат БПЛА по осях X, Y;

в — залежність коефіцієнта взаємної кореляції (X, Y)

На рис 4.2,а,б показано аналогічні залежності СКВ помилок визначення координат цілі σ_x, σ_y для системи 2 типу з помилками вимірювань $\sigma_r = 30\text{м}$, $\sigma_\varepsilon = 0.3$ град., $\sigma_\beta = 0.3$ град.

Як впливає з рис.4.2 характер зміни залежності зберігся. При цьому їх абсолютне значення збільшилося в два рази в порівнянні з аналогічною залежністю, приведеною на рис.4,1а,б. Залежність коефіцієнта взаємної кореляції γ_{xy} не змінюється і на рис. 4.2 не приводиться.

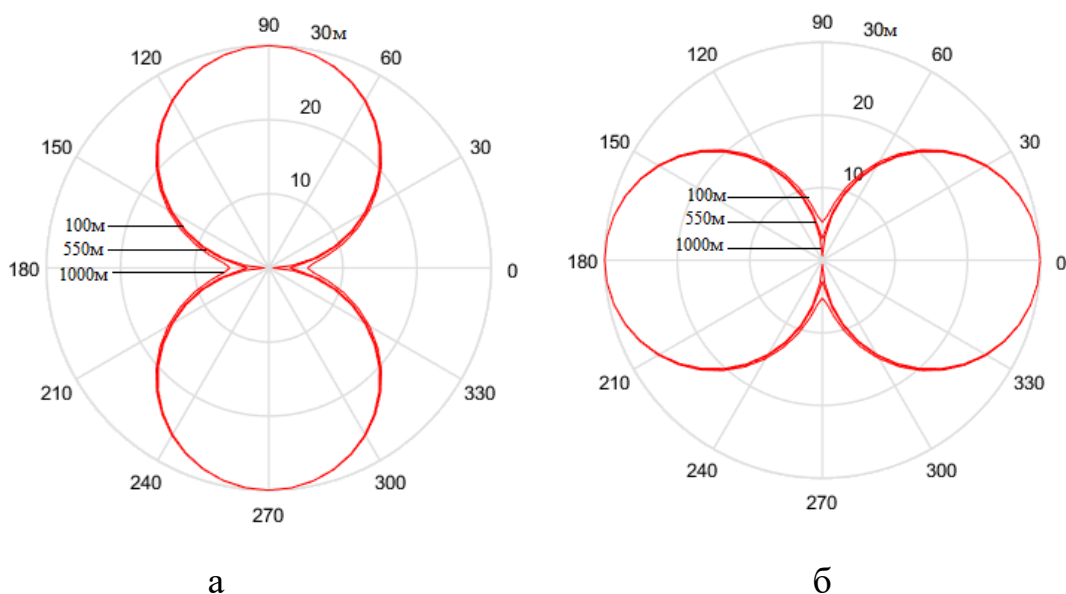


Рисунок 4.2 а,б, —залежність СКВ помилок визначення координат БПЛА по осях X, Y для системи іншого типу;

Аналіз точносних характеристик у вертикальній площині проведений аналогічно. Як впливає з виразу (3.8) дисперсія помилки вимірювання положення цілі по осі Z від азимута цілі β не залежить. Ціль рухається по дугах кіл у верхній півплощині з радіусами 1000м, 550м, 100м. На рис 4.3,а,б суцільними лініями в полярній системі координат (R, ε) показана залежність СКВ помилки визначення координат цілі по осі Z σ_z для системи типу 1 і типу 2. Як впливає з отриманих результатів, СКВ помилки визначення координат цілі по осі Z σ_z залежить від кута ε і дальності до цілі. Основний

внесок в помилку визначення координат цілі в декартовій системі координат на великих відстанях вносить помилка вимірювання кута місця ε .

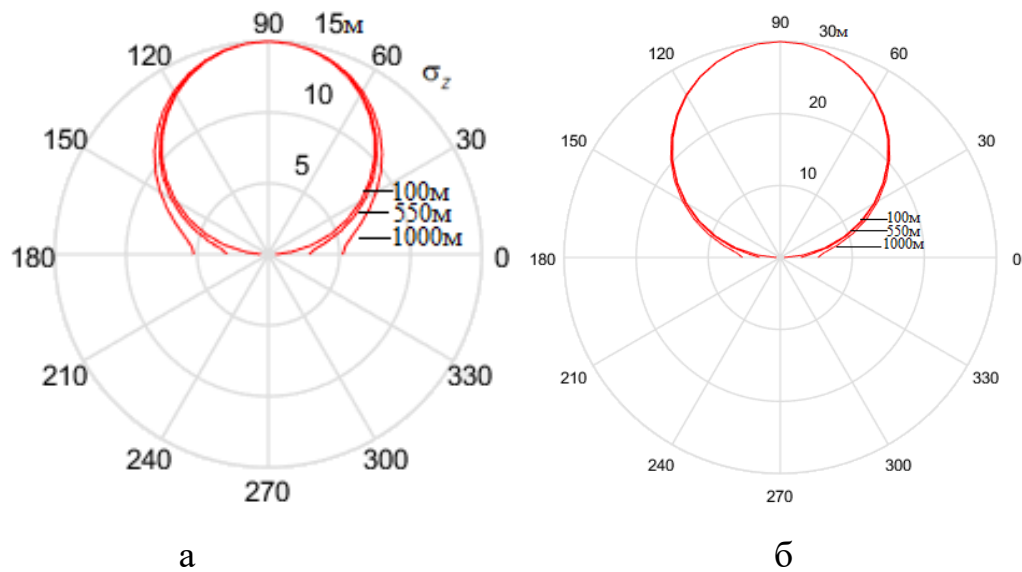


Рисунок 4.3 а,б, — залежність СКВ помилок визначення координат БПЛА по осі Z для різних систем

На рис.4.4,а,б суцільною лінією показана залежність модулів коефіцієнта взаємної кореляції $|\gamma_{xz}|$, $|\gamma_{yz}|$, помилок визначення координат цілі по осях декартової системи координат X,Z і Y,Z залежно від кута місця ε , які визначаються на основі виразів (3.10), (3.11). Дальність до цілі вважалася 250км. Азимут цілі дорівнює $\beta = \pi/3$.

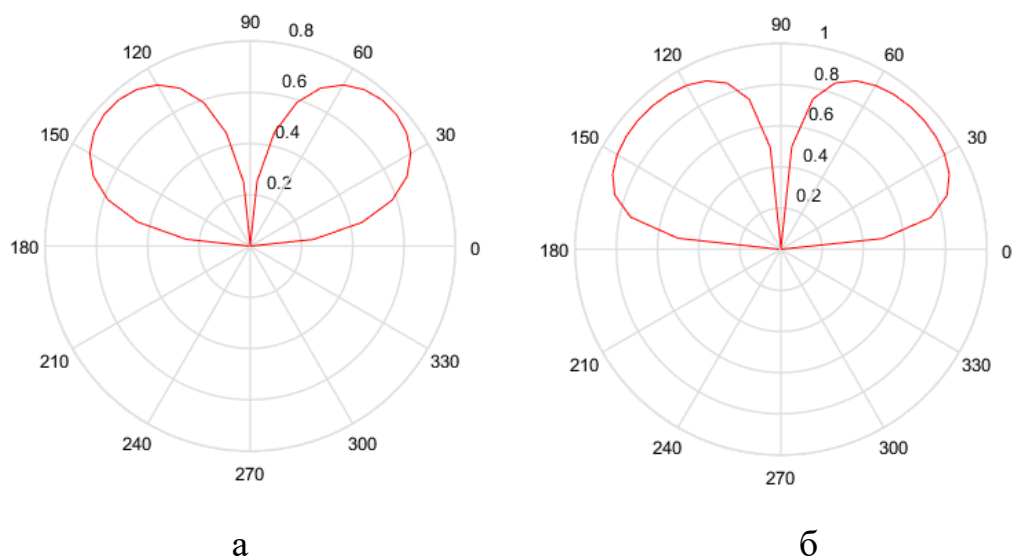


Рисунок 4.4 а,б, — залежність коефіцієнта взаємної кореляції (Z) для різних типів систем

Коефіцієнти взаємної кореляції γ_{xz} , γ_{yz} в цілому залежать від величини кута місця ε . Проте при невеликих значеннях ε вони приймають значення менше 0.1, що може бути використано при синтезі алгоритмів траєкторної фільтрації.

5 СИНТЕЗ АЛГОРИТМІВ ТРАЄКТОРНОЇ ФІЛЬТРАЦІЇ БПЛА ЗА ДА НИМИ СИСТЕМИ ВІДЕОСПОСТЕРЕЖЕННЯ І FMCW ДАЛЕКОМІРУ

У розглянутій постановці задачі, поряд з оцінкою невідомих параметрів руху об'єкта, має вирішуватись завдання розпізнавання виду його маневру. Результати розпізнавання маневру використовуються для формування оцінок вектора параметрів руху об'єкта. Тому синтезовані алгоритми ставляться до класу адаптивних. Найбільш повне рішення задачі фільтрації полягає у визначенні апостеріорної щільності ймовірності (ПВ) процесу, що фільтрується. На її основі можуть бути визначені оцінки невідомих параметрів будь-якої функції втрат. Вектор з безперервним значенням стану моделі руху цілі $u(k)$ та дискретний компонент $a_j(k)$, характеризує вигляд її руху, спільно утворюють змішаний марківський процес у дискретному часі. Дотримуючись методики синтезу, наведеної в монографії [15], можна показати, що апостеріорна ПВ змішаного процесу задовольняє системі рекурентних рівнянь

$$W_j^*(k) = \sum_{i=1}^M \Pi_{ij}(k, k-1) W_i(k-1); \quad (5.4)$$

$$W_j^*(u(k)) = \sum_{i=1}^M \Pi_{ij}(k, k-1) W_i(k-1) \times \\ \times \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Pi(u(k) / u(k-1), a_j(k)) W_i(u(k-1)) du(k-1) / W_j^*(k); \quad (5.5)$$

$$W_j(u(k)) = P(u_{\text{И}}(k) / u(k)) W_j^*(u(k)) / P(u_{\text{И}}(k) / a_j(k), U^{\text{И}}(k-1));, \quad (5.6)$$

$$W_j(k) = P(u_{\text{И}}(k) / a_j(k), U^{\text{И}}(k-1)) W_j^*(k) / P(u_{\text{И}}(k) / U^{\text{И}}(k-1)), \quad (5.7)$$

де $W_j^*(u(k)) = P(u(k) / a_j(k), U^I(k-1))$, $W_j(u(k)) = P(u(k) / a_j(k), U^I(k))$ – умовні екстрапольована та апостеріорна ПВ безперервного компонента $u(k)$ за умови $a_j(k)$;

$W_j^*(k) = P(a_j(k) / U^I(k-1))$, $W_j(k) = P(a_j(k) / U^I(k))$ – екстрапольована та апостеріорна ймовірності дискретного компонента $a_j(k)$;
 $U^I(k) = (u_I(k), \dots, u_I(1))$ – послідовність вимірів; $P(u_I(k) / u(k))$ – умовна ПВ, яка визначається за виразами (1); $P(u_I(k) / a_j(k), U^I(k-1))$ – умовна ПВ, яка визначається за допомогою формули:

$$P(u_I(k) / a_j(k), U^I(k-1)) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} P(u_I(k) / u(k)) W_j^*(u(k)) du(k); \quad (5.8)$$

$P(u_I(k) / U^I(k-1))$ – умовна ПВ, що обчислюється на основі співвідношення:

$$P(u_I(k) / U^I(k-1)) = \sum_{i=1}^M P(u_I(k) / a_j(k), U^I(k-1)) W_j^*(k). \quad (5.9)$$

Синтезований алгоритм (5.4)...(5.7) є нелінійним. Вирази (5.4), (5.7) використовуються для обчислення екстрапольованих $W_j^*(k)$ та апостеріорних $W_j(k)$ ймовірностей дискретних компонентів $a_j(k)$. Вирази (5.5), (5.6) використовуються для обчислення умовних екстрапольованих $W_j^*(u(k))$ та апостеріорних $W_j(u(k))$ ПВ безперервного компонента $u(k)$. Особливість синтезованого алгоритму полягає у нерозривній зв'язаності рівнянь фільтрації та екстраполяції дискретного та безперервного компонентів між собою. Алгоритм фільтрації (5.4)...(5.7) визначає структуру оптимального пристрою. Оптимальний фільтр містить канали і відноситься до класу пристроїв зі

зворотними зв'язками між каналами. Його структурна схема показана на рис. 5.1.

Кожен канал складається з блоку прогнозу БП, в якому обчислюються екстрапольовані розподіли $W_j^*(\mathbf{u}(k))$ і $W_j^*(k)$, і блоку фільтрації БФ, в якому обчислюються умовна апостеріорна щільність ймовірності $W_j(\mathbf{u}(k))$ і ненормована апостеріорна міра $W_j'(k) = P(u_{II}(k) / a_j(k), U^I(k-1))W_j^*(k)$.

Обчислені апостеріорні розподіли з виходів кожного каналу з затримкою на один такт надходять по лініях зворотного зв'язку в БП інших каналів оптимального пристрою, а також на вхід вирішального пристрою ВП, в якому визначається поточна спільна оцінка змішаного марківського процесу $\hat{\mathbf{\eta}}^B(k) = (\hat{\mathbf{u}}^B(k), \hat{\mathbf{a}}_i^B(k))$.

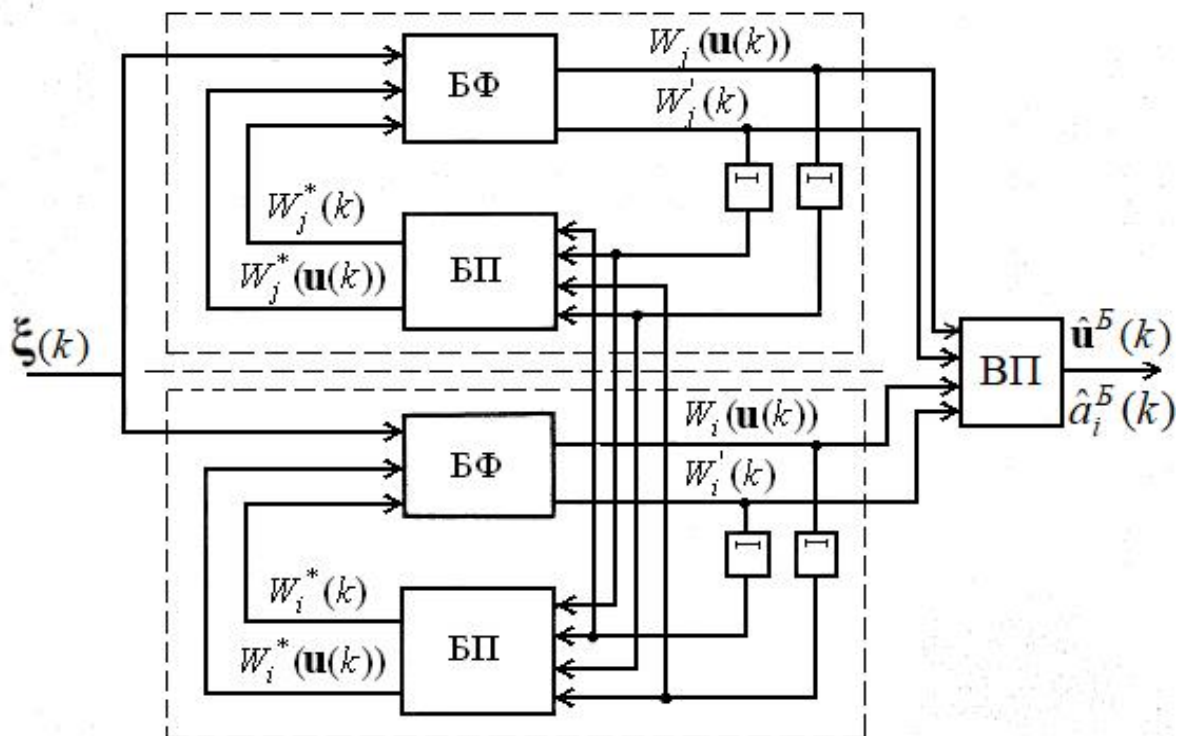


Рисунок 6.1 — структурна схема оптимальний пристрою (фільтру)

Зворотні зв'язку між каналами обумовлені марківською властивістю змінної перемикавання $a_j(k)$. Наявність зворотного зв'язку дозволяє виконувати

постійну корекцію каналів, налаштованих на помилкові значення змінної перемикачів $a_j(k)$ від "істинного" каналу, що запобігає їх розбіжності і забезпечує швидке розпізнавання зміни структури процесу $\mathbf{u}(k)$. У синтезованому оптимальному пристрої немає зростання числа каналів, що обумовлено марківською властивістю змішаного процесу.

На практиці реалізація синтезованого оптимального алгоритму спільної фільтрації змішаного марківського процесу викликає деякі складності. Вони обумовлені вимогою великих обсягів пам'яті і швидкодії обчислювача при виконанні чисельного інтегрування багатовимірних щільностей ймовірностей. Тому представляє значний інтерес синтез квазіоптимальних алгоритмів фільтрації і екстраполяції.

Квазіоптимальний адаптивний алгоритм можна отримати шляхом лінеаризації рівняння вимірювання радіальної швидкості, а також гаусівської апроксимації умовних екстрапольованих ПВ $W_j^*(u(k))$, $j = \overline{1, M}$ на кожному кроці k .

При цьому квазіоптимальний алгоритм адаптивної фільтрації параметрів руху маневруючого об'єкта у прямокутній системі координат має вигляд:

$$W_j^*(k) = \sum_{i=1}^M \Pi_{ij}(k, k-1) W_i(k-1); \quad (5.9)$$

$$u_j^*(k) = \sum_{i=1}^M \Pi_{ij}(k, k-1) W_i(k-1) F_j(k, k-1) \hat{u}_i(k-1) / W_j^*(k); \quad (5.10)$$

$$P_j^*(k) = \sum_{i=1}^M \left\{ F_j(k, k-1) \hat{P}_i(k-1) F_j^T(k, k-1) + G_j(k) G_j^T(k) + \right. \\ \left. + (F_j(k, k-1) \hat{u}_i(k-1) - u_j^*(k)) (F_j(k, k-1) \hat{u}_i(k-1) - u_j^*(k))^T \right\} \times \\ \times \Pi_{ij}^\Gamma(k, k-1) W_i(k-1) / W_j^*(k); \quad (5.11)$$

$$\hat{u}_j(k) = u_j^*(k) + K_j(k) (u^H(k) - h(u_j^*(k))); \quad (5.12)$$

$$K_j(k) = P_j^*(k) H_j^T(k) (H_j(k) P_j^*(k) H_j^T(k) + R(k))^{-1}; \quad (5.13)$$

$$\hat{P}_j(k) = P_j^*(k) - K_j(k)H_j(k)P_j^*(k) ; \quad (5.14)$$

$$W_j(k) = P(u^I(k) / a_j(k), U^I(k-1))W_j^*(k) / P(u^I(k) / U^I(k-1)), \quad (5.15)$$

де $u_j^*(k), \hat{u}_j(k), P_j^*(k), \hat{P}_j(k)$ - математичні очікування та кореляційні матриці умовних ПВ.

$W_j^*(u(k)), W_j(u(k))$; $K_j(k)$ - коефіцієнт посилення j -го каналу квазіоптимального пристрою; $H_j(k) = \nabla_{\mathbf{u}(k)} h(\mathbf{u}^*(k))$ - матриця Якобіану векторнозначної функції $h(\mathbf{u}^*(k))$, яка має вигляд:

$$H_j(k) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{\dot{x}_j^* r_j^* - \dot{r}_j^* x_j^*}{r_j^{*2}} & \frac{x_j^*}{r_j^*} & 0 & \frac{\dot{y}_j^* r_j^* - \dot{r}_j^* y_j^*}{r_j^{*2}} & \frac{y_j^*}{r_j^*} & 0 & \frac{\dot{z}_j^* r_j^* - \dot{r}_j^* z_j^*}{r_j^{*2}} & \frac{z_j^*}{r_j^*} & 0 \end{bmatrix}$$

Умовна ПВ $P(u^I(k) / a_j(k), U^I(k-1))$ - визначається за формулою:

$$P(u^I(k) / a_j(k), U^I(k-1)) = (2\pi)^{-1} \det(D_j(k))^{-1/2} \exp\{-0.5 \|u^I(k) - H u_j^*(k)\|_{D_j(k)}\},$$

де $D_j(k) = H_j(k)P_j^*(k)H_j^T(k) + R(k)$.

Квазіоптимальний адаптивний алгоритм часто розходиться на початковій ділянці його роботи, що зумовлено нелінійним характером рівняння вимірювання радіальної швидкості. Для усунення ефекту запропоновано збільшувати дисперсію помилки вимірювань радіальної швидкості на величину, пропорційну до дисперсії помилки екстраполяції радіальної швидкості $p_{\dot{r},k}^*$ по формулі:

$$R'_{4,4}(k) = R_{4,4}(k) + \gamma p_{\dot{r}}^*(k), \quad (5.16)$$

де γ - коефіцієнт, що регулює величину додаткової складової дисперсії помилки вимірювання, що вноситься; $p_{\dot{r},k}^*$ - дисперсія помилки прогнозу

радіальної швидкості, яка обчислюється за формулою:

$$p_{\dot{r},k}^* = \sum_{j=1}^6 (\sum_{i=1}^6 H_{4,i} P_{i,j}^*) H_{j,4}^T. \quad (5.17)$$

В роботі [5] показано, що доцільно вважати $\gamma = 0.8$.

Квазіоптимальний алгоритм (5.9)...(5.15) є нелінійним. На відміну від оптимального алгоритму при його реалізації обчислюються лише перший та другий моменти умовних апостеріорних розподілів. При цьому апостеріорна ПВ $W(u(k))$ при переході на наступний крок фільтрації апроксимується сумою M гаусівських щільностей. Синтезований квазіоптимальний фільтр також включає M каналів і відноситься до класу пристроїв із зворотними зв'язками між каналами. Він відрізняється від оптимального пристрою фільтрації тим, що в блоках БФ і БП квазіоптимального пристрою обчислюються тільки математичні очікування $\mathbf{u}_j^*(k)$, $\hat{\mathbf{u}}_j(k)$ і кореляційні матриці $\mathbf{P}_j^*(k)$, $\hat{\mathbf{P}}_j(k)$ умовних щільностей ймовірностей $W_j^*(\mathbf{u}(k))$ і $W_j(\mathbf{u}(k))$, $j = \overline{1, M}$. Структурна схема частини j -го каналу, що використовується для визначення оцінок безперервного компонента $\mathbf{u}(k)$ квазіоптимального пристрою фільтрації, показана на рис. 6.2.

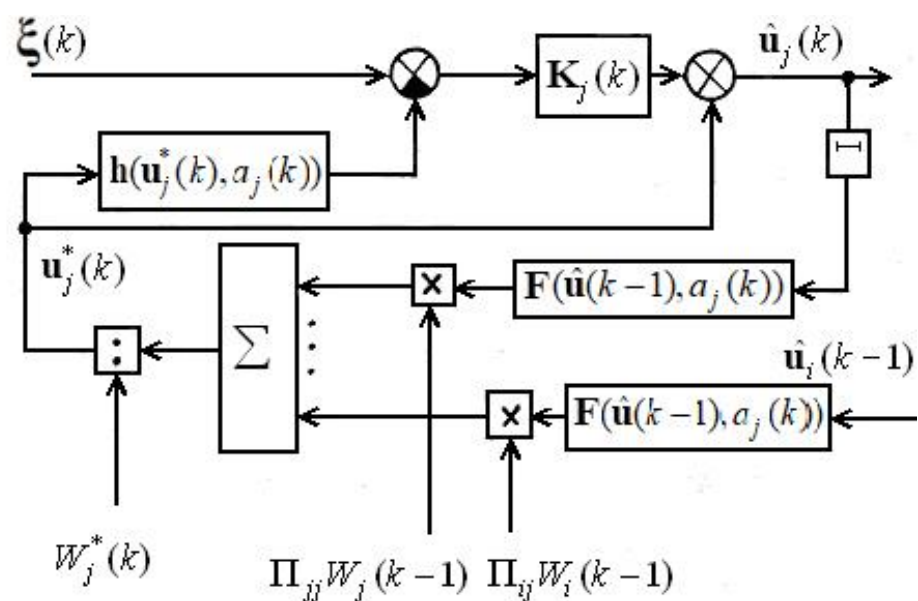


Рисунок 5.2 Структурна схема частини каналу

Початковий вектор оцінки та кореляційна матриця помилок оцінки, отримані на основі двох перших вимірів у моменти часу $k = 1$ і $k = 0$ описуються виразами (5.18), (5.19).

Для запуску квазіоптимального адаптивного алгоритму необхідно встановити початкові умови. За наявності двох вимірів у момент часу $k = 0$ і $k = 1$ вектори початкової оцінки однакові та мають вигляд:

$$\hat{u}_j^T(1) = (x_{И}(1); \frac{x_{И}(1) - x_{И}(0)}{T}; 0; y_{И}(1); \frac{y_{И}(1) - y_{И}(0)}{T}; 0; z_{И}(1); \frac{z_{И}(1) - z_{И}(0)}{T}; 0), j = \overline{1, M} \quad (6.18)$$

Кореляційні матриці помилок початкової оцінки різні та мають вигляд:

$$\hat{P}_j(1) = \begin{bmatrix} \sigma_x^2(1) & \frac{\sigma_x^2(1)}{T} & 0 & \sigma_{xy}^2(1) & \frac{\sigma_{xy}^2(1)}{T} & 0 & \sigma_{xz}^2(1) & \frac{\sigma_{xz}^2(1)}{T} & 0 \\ \frac{\sigma_x^2(1)}{T} & \frac{\sigma_x^2(1) + \sigma_x^2(0)}{T^2} + \sigma_{jax}^2 T^2 & 0 & \frac{\sigma_{xy}^2(1)}{T} & \frac{\sigma_{xy}^2(1) + \sigma_{xy}^2(0)}{T^2} & 0 & \frac{\sigma_{xz}^2(1)}{T} & \frac{\sigma_{xz}^2(1) + \sigma_{xz}^2(0)}{T^2} & 0 \\ 0 & 0 & \sigma_{jax}^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ \sigma_{xy}^2(1) & \frac{\sigma_{xy}^2(1)}{T} & 0 & \sigma_y^2(1) & \frac{\sigma_y^2(1)}{T} & 0 & \sigma_{yz}^2(1) & \frac{\sigma_{yz}^2(1)}{T} & 0 \\ \frac{\sigma_{xy}^2(1)}{T} & \frac{\sigma_{xy}^2(1) + \sigma_{xy}^2(0)}{T^2} & 0 & \frac{\sigma_y^2(1)}{T} & \frac{\sigma_y^2(1) + \sigma_y^2(0)}{T^2} + \sigma_{jay}^2 T^2 & 0 & \sigma_{yz}^2(1) & \frac{\sigma_{yz}^2(1) + \sigma_{yz}^2(0)}{T^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_{jay}^2 & 0 & 0 & 0 \\ \sigma_{xz}^2(1) & \frac{\sigma_{xz}^2(1)}{T} & 0 & \sigma_{yz}^2(1) & \sigma_{yz}^2(1) & 0 & \sigma_z^2(1) & \frac{\sigma_z^2(1)}{T} & 0 \\ \frac{\sigma_{xz}^2(1)}{T} & \frac{\sigma_{xz}^2(1) + \sigma_{xz}^2(0)}{T^2} & 0 & \frac{\sigma_{yz}^2(1)}{T} & \frac{\sigma_{yz}^2(1) + \sigma_{yz}^2(0)}{T^2} & 0 & \frac{\sigma_z^2(1)}{T} & \frac{\sigma_z^2(1) + \sigma_z^2(0)}{T^2} + \sigma_{jaz}^2 T^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sigma_{jaz}^2 \end{bmatrix}$$

$$j = \overline{1, M}, \quad (6.19)$$

6 АНАЛІЗ СИНТЕЗОВАНОГО КВАЗІОПТИМАЛЬНОГО АЛГОРИТМУ ТРАЄКТОРНОЇ ФІЛЬТРАЦІЇ БПЛА ЗА ДАНИМИ СИСТЕМИ ВІДЕОСПОСТЕРЕЖЕННЯ І FMCW ДАЛЕКОМІРУ

Аналіз точносних характеристик синтезованого алгоритму оцінювання параметрів руху БПЛА з розпізнаванням різних видів руху виконаний за допомогою статистичного моделювання на ЕОМ. Модель руху БПЛА описується стохастичною динамічною системою з випадковою структурою, яка враховує три основних види руху $M=3$: зависання, $j=1$; майже рівномірний рух, $j=2$; рух з маневром, $j=3$.

Формування траєкторій БПЛА здійснювалось на основі рівняння, яке враховує три основних види руху $M=3$. Траєкторія включає такти $-1 \leq k \leq 50$.

Такти $k=-1$, $k=0$ використовуються для формування початкових умов адаптивного алгоритму. На них моделюється рівномірний рух $j=2$ БПЛА.

Такти $1 \leq k \leq 50$ використовуються для оцінки ефективності роботи алгоритму.

Для наглядності аналізу ефективності роботи алгоритму кожна тестова траєкторія була розбита на інтервали, на яких моделювався відповідний тип руху:

- Перший $1 \leq k \leq 9$ – рівномірний рух $j=2$;
- Другий $10 \leq k \leq 15$ – маневр $j=3$;
- Третій $16 \leq k \leq 20$ – рівномірний рух $j=2$;
- Четвертий $21 \leq k \leq 25$ – маневр «скидання швидкості» $j=3$;
- П'ятий $26 \leq k \leq 33$ – зависання $j=1$;
- Шостий $34 \leq k \leq 39$ – маневр $j=3$;
- Сьомий $40 \leq k \leq 50$ – рівномірний рух $j=2$.

Початкові параметри руху БПЛА: $x_c(-1)=400$ м, $\dot{x}_w(-1)=-20$ м/с, $\ddot{x}_c(-1)=0$ м/с², $y_c(-1)=800$ м, $\dot{y}_c(-1)=-20$ м/с, $\ddot{y}_c(-1)=0$ м/с², $z_c(-1)=100$ м, $\dot{z}_c(-1)=0$ м/с, $\ddot{z}_c(-1)=0$ м/с². Темп формування траєкторії БПЛА $T = 1$ секунда.

На рис. 6.1 показано типову тестову траєкторію руху БПЛА в системі координат камери $X_c Y_c Z_c$.

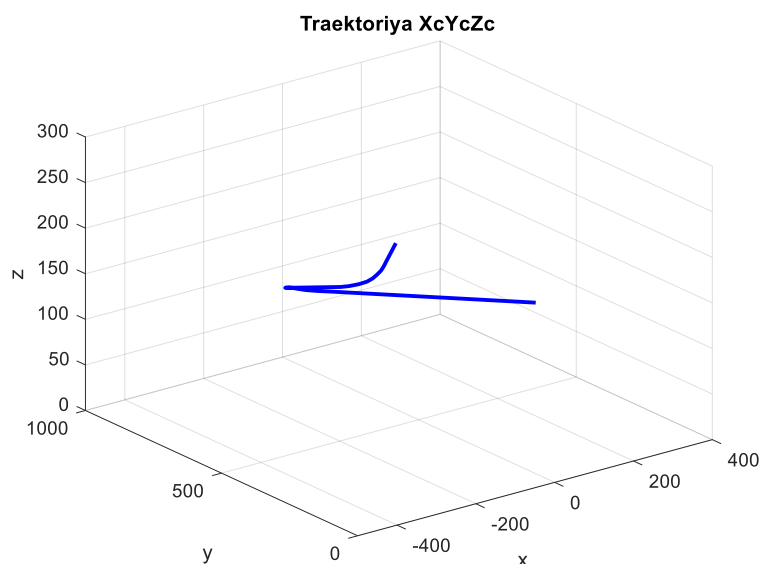


Рис. 6.1 – Траєкторія руху БПЛА в СК XYZ

На рис. 6.2 показано типову тестову траєкторію руху БПЛА на площині $X Y$.

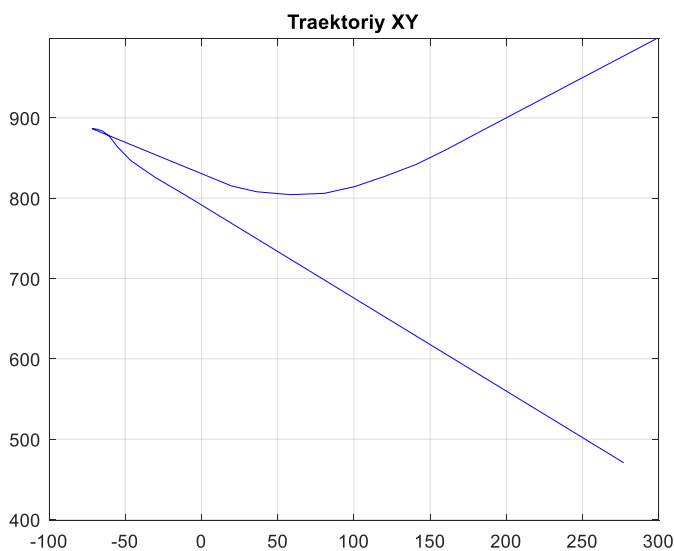


Рисунок 6.2 – Траєкторія руху БПЛА на площині XY

На рис. 6.3 показано залежності зміни параметрів руху БПЛА по осях XY .

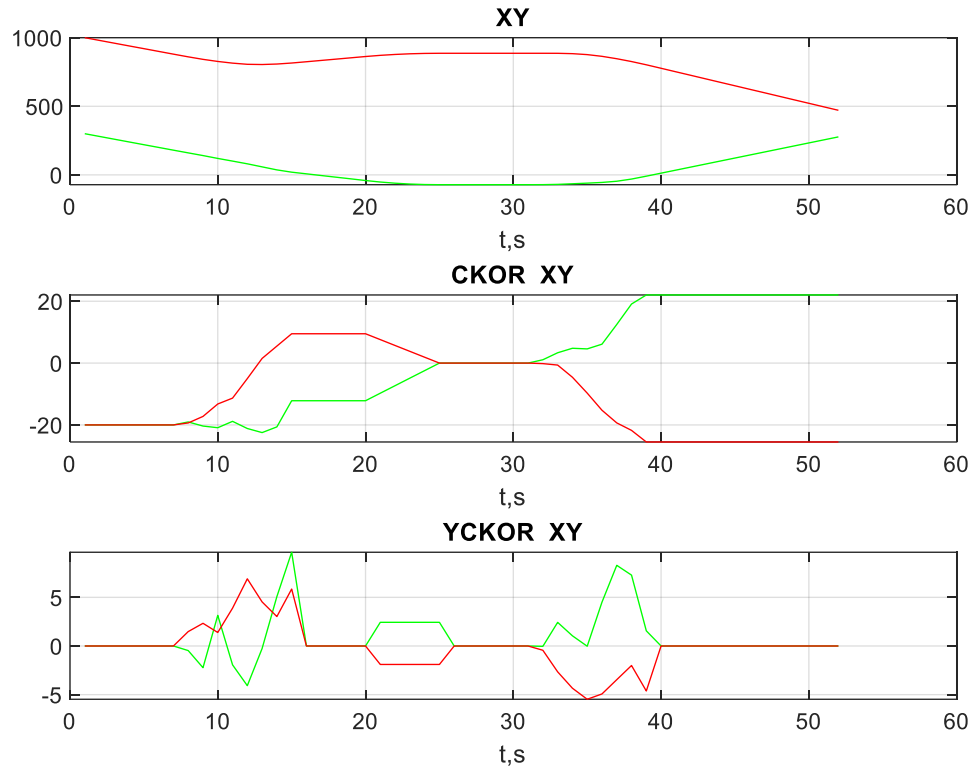


Рис. 6.3 – Зміни параметрів руху БПЛА по осях XY

При описі адаптивного алгоритму оцінювання СКВ шумів збудження для різних типів руху БПЛА встановлено: $a_1 = 1$ м/с; $a_2 = 1$ м/с²; $a_3 = 7$ м/с³.

Помилки вимірювання координат, системи встановлених складових, $\sigma_\beta = 0.1$ град., $\sigma_\varepsilon = 0.1$ град., $\sigma_r = 20$ м, $\sigma_{\dot{r}} = 10$ м/с. Кількість реалізацій метода Монте-Карло дорівнює $M=50$.

На рис.7.4 показані математичні очікування m_χ^*, m_μ^* (зелені криві) і СКВ $\hat{\sigma}_\chi, \hat{\sigma}_\mu$ (сині криві) помилок прогнозу положення об'єкту, котрі визначені методом Монте-Карло та СКВ помилок прогнозу $\sqrt{p_\chi^*}, \sqrt{p_\mu^*}$ (червоні криві) положення об'єкту, які обчислені фільтром.

Математичне очікування реальних похибки оцінювання добре корегується з СКВ похибок оцінювання, розрахованими фільтром, що свідчить про правильну роботу алгоритма.

На ділянках прямолінійного рівномірного руху БПЛА СКВ прогнозу його координат в 3.2 рази менше, ніж на ділянках руху з маневром. На ділянках зависання БПЛА додатково можна підвищити точність прогнозування його положення в 2 рази. Це дозволяє формувати строби супроводження адекватно в поточній ситуації.

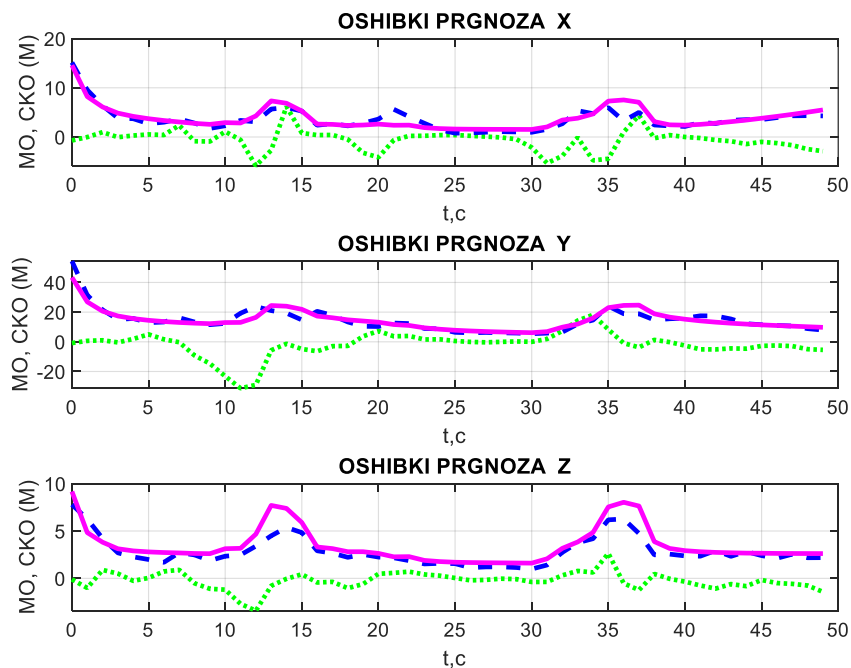


Рис. 6.4 – Графіки математичних очікувань і СКВ помилки помилок прогнозу положення БПЛА, розраховані адаптивним алгоритмом

На рис.6.5 показані математичні очікування $\hat{m}_x, \hat{m}_y$ (зелені криві) і $\hat{\sigma}_x, \hat{\sigma}_y$ (сині криві) помилок оцінки положення об'єкту, котрі визначені методом Монте-Карло та СКВ помилок оцінки $\sqrt{\hat{p}_x}, \sqrt{\hat{p}_y}$ (червоні криві) положення об'єкту, які обчислені фільтром. Фактичні та розраховані фільтром СКО помилок оцінок добре узгоджуються між собою, що свідчить про правильну роботу алгоритму.

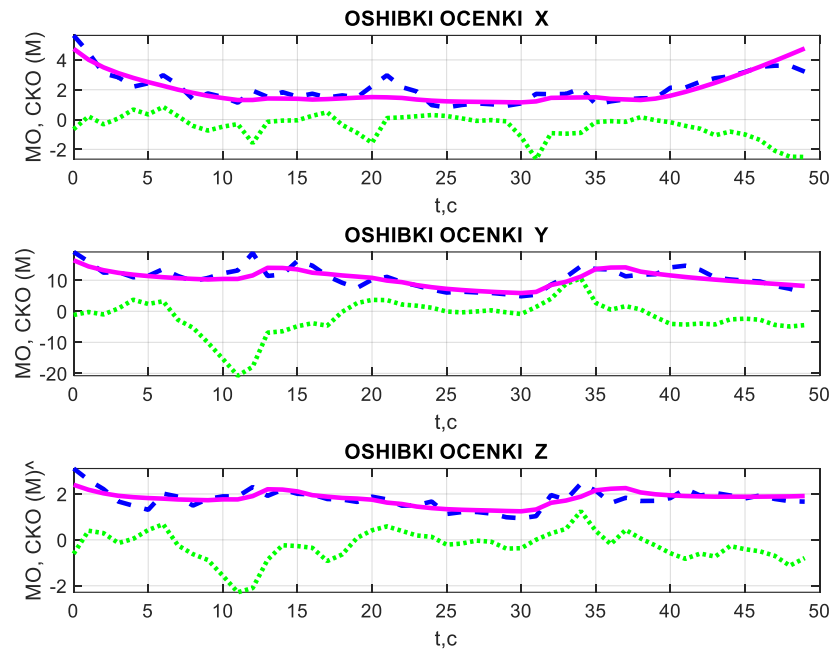


Рис. 6.5 – Графіки математичних очікувань і СКВ помилки помилок оцінок положення БПЛА, розраховані адаптивним алгоритмом

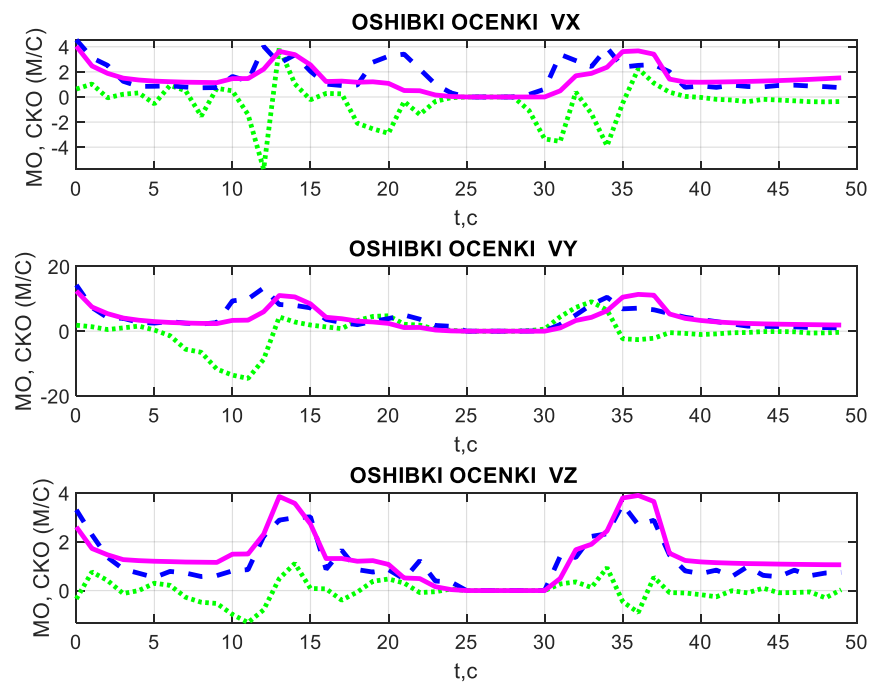


Рис. 6.6 – Графіки математичних очікувань і СКВ помилки помилок оцінок швидкості БПЛА, розраховані адаптивним алгоритмом

На рис. 6.7 показані ймовірності p_{m1} (зелена крива) p_{m2} (синя крива) , p_{m3} (червона крива) розпізнавання адаптивним алгоритмом різних видів руху БПЛА: зависання, майже рівномірний рух, рух з маневром. Адаптивний алгоритм забезпечує високу ймовірність розпізнавання різних видів руху БПЛА.

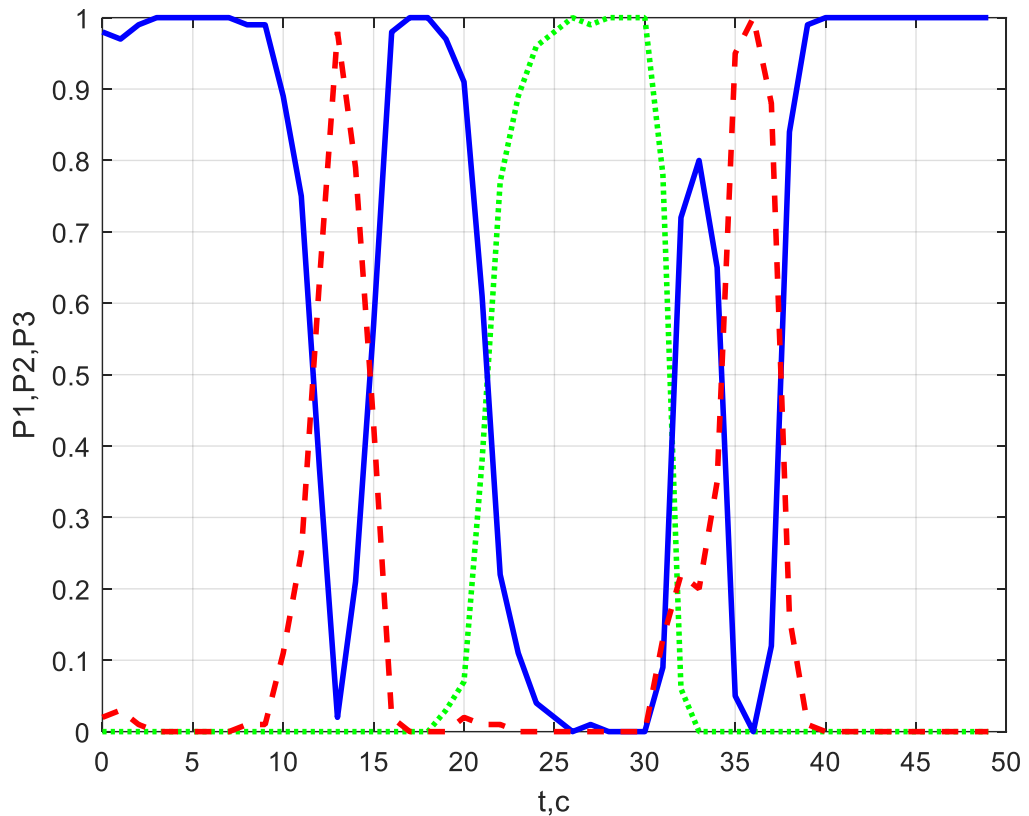


Рис. 6.7 – Ймовірності розпізнавання адаптивним алгоритмом різних видів руху БПЛА

На Рис. 6.8 Продемонстровано попадання вимірних координат положення БПЛА в строби супроводження по осях прямокутної СК.

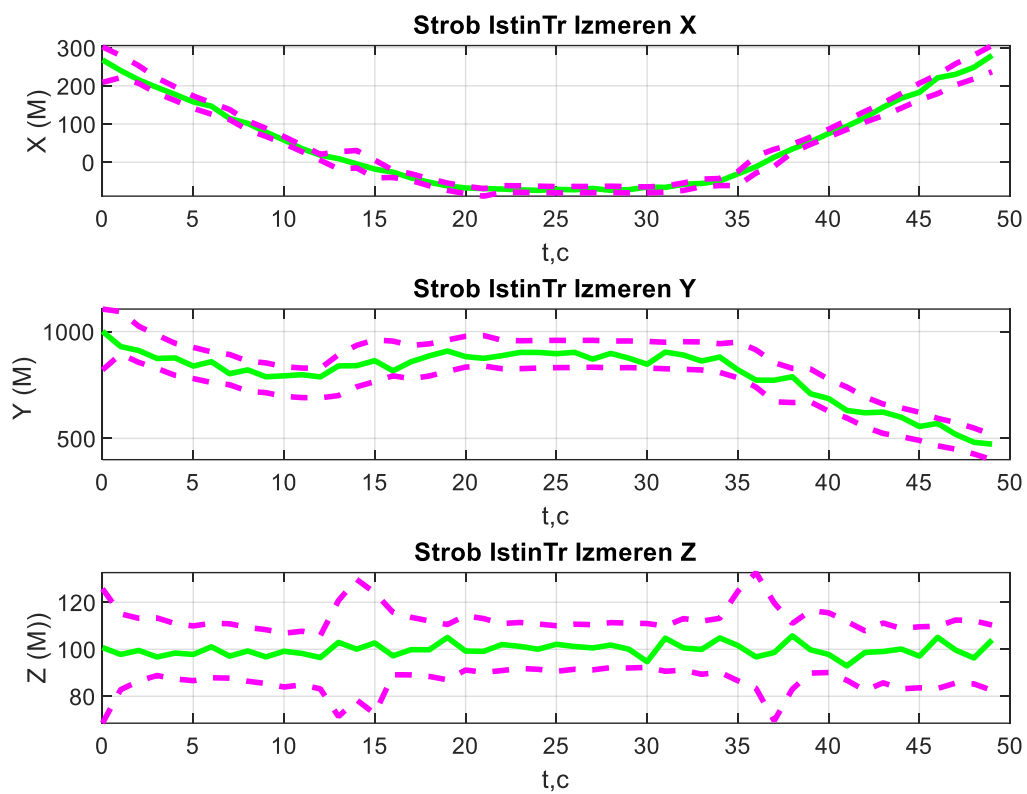


Рис. 6.8 – Продемонстровано попадання вимірних координат положення БПЛА в строби супроводження по осях прямокутної СК

7 АНАЛІЗ МОЖЛИВОСТЕЙ ПРАКТИЧНОЇ РЕАЛІЗАЦІЇ РОЗРОБЛЕНОГО КВАЗІОПТИМАЛЬНОГО АЛГОРИТМУ ТРАЄКТОРНОЇ ФІЛЬТРАЦІЇ

Основні характеристики для реалізації розробленого алгоритму визначаються на основі розміру пам'яті та обчислень виконаних в реальному часі. Час (швидкість) обчислень обчислювальної машини є основною характеристикою. Цей час визначається швидкістю ЕОМ та загальною кількістю операцій, необхідних під час обчислення.

Його більш точна оцінка вимагає знання системи команд конкретного комп'ютера, що реалізує цей фільтр. Однак при порівнянні різних способів організації обчислення задач фільтрації достатньою апроксимацією часу обчислення служить кількість необхідних операцій множення (ділення). При реалізації апаратними методами ці операції приблизно в 6...12 разів повільніше додавання, а при програмному (наприклад, в мікропроцесорних системах) — у 120 разів, тому на виконання цих операцій витрачається основний час.

За правилами лінійної алгебри, можливо дізнатись кількість операцій множення і додавання, необхідних для розрахунків оцінки фільтрації та комплексування на одному кроці.

Обсяг пам'яті, необхідний для збереження змінних, наведено у табл. 7.1.

Табл. 7.1 — Обсяг пам'яті для збереження змінних

Змінна	Розмірність	Необхідне число комірок ЗП
$\hat{u}(k)$	$n \times 1$	n
$F(k, k-1)$	$n \times n$	n^2
$K(k)$	$n \times s$	ns
$s(k)$	$s \times 1$	s
$H(k)$	$s \times n$	sn
$\hat{P}(k)$	$n \times n$	n^2
$R(k)$	$s \times s$	s^2
$P^*(k)$	$n \times n$	n^2
$G(k)$	n	n
$Q(k)$	$r \times r$	r^2
$u^*(k)$	$n \times 1$	n

Відомо, що обернена матриця визначається за стандартним алгоритмом Гауса, який потребує для своєї реалізації s^3 операцій множення й додавання, де $s \times s$ - розмірність матриці [16].

За таблицею 7.2 маємо:

Загальна кількість операцій множення на один крок фільтрації:

$$M_{\Phi}(n, s, r) = 3n^3 + 2n^2s + 2ns^2 + n^2r + nr^2 + 2ns + s^3 + n^2; \quad (6.1)$$

Загальна кількість операцій додавання на один крок фільтрації:

$$A_{\Phi}(n, s, r) = 3n^3 + 2n^2s + 2ns^2 + n^2r + nr^2 + s^3 - 2n^2 - nr; \quad (6.2)$$

Як видно з виразів вище, діапазон розрахунків пропорційний кубу розмірності системи.

Отримуємо:

$$M_{\Phi}(3, 1, 1) = 364,$$

$$A_{\Phi}(3, 1, 1) = 301.$$

Кількість комірок пам'яті необхідна для реалізації дорівнює 240.

Розрахунки приводяться при умови слідкування за однією ціллю. Всі суми операцій множення і додавання, за мінімальний темп надходження інформації у 5 тактів, і поточній обробці кадру буде близько 1000. Використаємо сигнальний процесор ADSP – TS 201 S, з тактовою частотою в 500 МГц, структурна схема котрого наведена на рис.7.1.

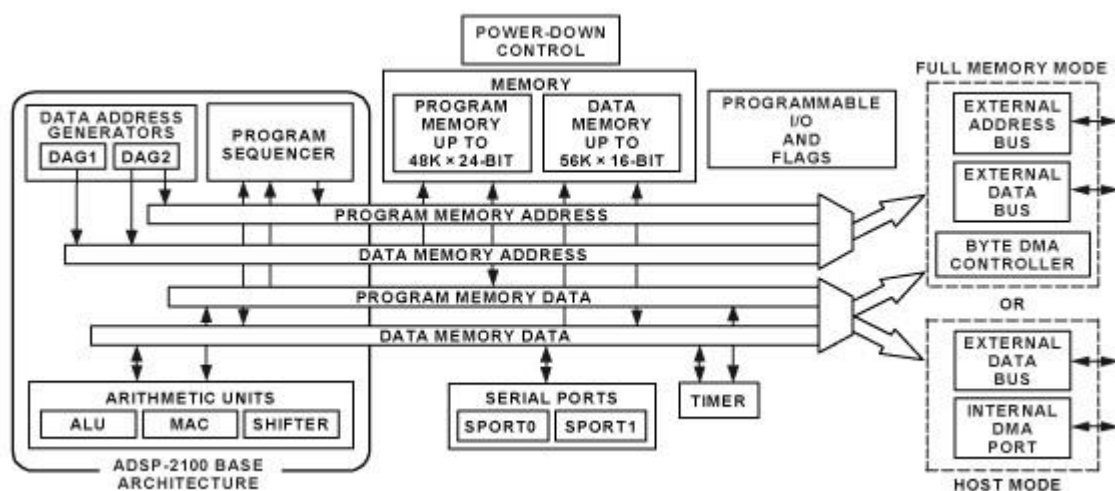


Рисунок 7.1 – Структурна схема процесору ADSP – TS 201 S

Даний сигнальний процесор має об'єм пам'яті 3 Мб, 4 швидкісних порта, по 600 Мбайт/сек, зовнішню шина 100 МГц та 12 каналів.

Табл. 7.2 — Кількість операції для розрахунку

Величина, що розраховується	Функціональна залежність	Поточний крок розрахунків	Кількість операцій множення	Загальна кількість операцій множення	Кількість операцій додавання	Загальна кількість операцій додавання
$\hat{u}(k k-1)$	$F(k, k-1)\hat{u}(k-1 k-1)$	$\hat{F}\hat{u}$	n^2	n^2	n^2-n	n^2-n
$P(k k-1)$	$F(k, k-1)P(k-1 k-1) \times$ $\times F^T(k, k-1) + G(k, k-1) \times$ $\times Q(k-1)G^T(k, k-1)$	P_{F^T}	n^3	$2n^3+n^2+n^2r$	n^3-n^2	$2n^3-2n^2+n^2+n^2r-nr$
		$F(P_{F^T})$	n^3		n^3-n^2	
		Q_G^T	m^2		m^2-nr	
		$G(Q_G^T)$	n^2r		n^2r-n^2	
$K(k)$	$P(k k-1)H^T(k) \times$ $\times [H(k)P(k k-1) \times$ $\times H^T(k) + R(k)]^{-1}$	$F^T P_{F^T} + G Q_G^T$	0	$n^2s+n^2s^2+s^3$	n^2	$n^2s+2ns^2-2ns+s^3$
		P_{H^T}	n^2s		n^2s-nr	
		$H(P_{H^T})^T$	ns^2		ns^2-s^2	
		$HP_{H^T} + R$	0		s^2	
$P(k k)$	$[1 - K(k)H(k)] \times$ $\times P(k k-1)$	$[HP_{H^T} + R]^{-1}$	s^3	ns^2-ns	s^3	$n^2s+2ns^2-2ns+s^3$
		$K(k)$	ns^2		ns^2-nr	
		KH	n^2s		n^2s-n^2	
		$1 - (KH)$	0		n^2	
$\hat{u}(k k)$	$\hat{u}(k k-1) + K(k)[s(k) -$ $- H(k)\hat{u}(k k-1)]$	$[1 - KH]P$	n^3	$2ns$	n^3-n^2	$2ns$
		$H\hat{u}$	ns		$ns-s$	
		$s - H\hat{u}$	0		s	
		$K[s - H\hat{u}]$	ns		$ns-n$	
		$\hat{u}(k k)$	0		n	
				$M(n, s, r)$		$A(n, s, r)$

8 РОЗРОБКА СТАРТАП ПРОЕКТУ

В цьому розділі розглядається опис ідеї стартап-проекту, з урахуванням різноманітних економічних факторів, та її реалізація [17].

Таблиця 8.1 – Загальний опис проекту

1. Повна назва проекту	Алгоритм траєкторної фільтрації за даними системи відеоспостереження і FMCW далекоміра
2. Автор проекту	Сатиренко М.І.
3. Анотація	В даній магістерській роботі було описано алгоритм траєкторної фільтрації за даними системи відеоспостереження і FMCW далекоміра. Метою даного розділу є реалізація продукту на світовому ринку, та подальший розвиток який зумовлений актуальністю та ефективними показниками роботи алгоритму.
4. Термін реалізації проекту	6
	<i>Тривалість проекту (місяці)</i>
5. Необхідні ресурси	Інтелектуальні: інженер – конструктор (15 000-20 000грн.), програміст (20 000-25 000грн.), менеджер (15 000 грн.). Матеріальні: ПК Фінансові: ~120 000 грн.
	<i>Перелік потрібних ресурсів</i>

6. Опис проблеми, котру вирішує проект	Вчасне виявлення, спостереження, та локалізація БПЛА в просторі є ключовим фактором для протидії. Загроза викликана дронами на сьогоднішній день є однією з нагальних проблем нашої держави. Розроблений алгоритм має на меті вирішити проблему траєкторної фільтрації руху БПЛА.
---	---

7. Головні цілі та завдання проекту	Метою дослідження є покращення ефективності розпізнавання маневрів малорозмірних БПЛА в просторі, шляхом поєднання відеоспостереження та FMCW далекоміру та розробки на базі даної системи ефективного адаптивного алгоритму траєкторної фільтрації, який може відрізняти різноманітні можливі маневри БПЛА, що є запорукою своєчасної подальшої протидії.
--	---

8.1 Опис ідеї продукту

Таблиця 8.2 – Опис ідеї проекту

Зміст ідеї	Напрямок застосування	Вигода для користувача
Код програми для траєкторної фільтрації маневруючого БПЛА	На підприємствах критичної інфраструктури	Ціна, якість, підвищення безпеки, компактність
	Для комерційних цілей	
	На військових об'єктах	
	На об'єктах, яким необхідний високий рівень захисту від проникнення	

Ідея продукту – розроблений алгоритм буде виявляти різноманітні маневри БПЛА, на базі системи відеоспостереження і FMCW далекоміру, що буде супроводжувати ціль та локалізувати її в просторі.

8.2 Технічний аудит ідеї проекту

Для розкриття основних переваг продукту потрібно привести технічний аудит технології, що описаний в наступних пунктах даного розділу.

Таблиця 8.3 – Переваги стартап проекту

№ п/п	Показники стану ринку(найменування)	Характеристика
1	Ціна готового товару (проект системи із ПЗ)	Низька в порівнянні з існуючими аналогами та прямими конкурентами
2	Динаміка ринку	Досить висока
3	Багатозадачність	Алгоритм має змогу обробити велику к-сть даних
4	Висока продуктивність	Має високі показники розпізнавання маневрів

Таблиця 8.4 – Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування
------------------	--	----------------------

1	Актуальна задача	Сьогодні використання дронів набуло неабиякої популярності та зростає надалі. Необхідність систем супроводу, траєкторної фільтрації та захисту для уникнення різноманітних форм збитків є очевидною.
2	Новизна	Покращення методів та підвищення загальної ефективності алгоритмів траєкторної фільтрації
3	Гнучкість	Додаткові оновлення програми або об'єднання з існуючими системами супроводу, захисту, що дає додаткові можливості для реалізації продукту

8.3 Аналіз ринкових можливостей запуску стартап-проекту

Обстеження можливостей ринку збуту, які можна використати під час просування проекту, та існуючих загроз для реалізації продукту дає змогу визначити напрямок розвитку проекту з урахуванням стану ринкового середовища, умов, необхідних потенційним клієнтам та існуючої продукції конкурентів.

Таблиця 8.5 – Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреба, що формує ринок	Цільові сегменти ринку	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
----------	--------------------------------	------------------------------	---	-----------------------------------

1	Точність в розпізнаванні траєкторії цілі	Сфера захисту, виробництва, установи з масовим скупченням людей, військові об'єкти	Ефективність, ціна, масове виробництво, швидкодія, якість	Можливість легкого використання, швидкого навчання, простота монтажу, швидкість у розробці товару
---	---	--	--	---

Таблиця 8.6 – Фактори можливості

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Вихід на європейський, азіатський, американський ринок	Показати алгоритм іноземним компаніям, у яких є необхідні умови та обладнання для розвитку проекту	Вдала реалізація алгоритму, що дозволить зменшити можливість втрати розрахунків траєкторії цілі та супроводу в цілому

Таблиця 8.7 – Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства
Вказати тип конкуренції – технократія	Існують інші запропоновані алгоритми	Впливає, збільшити відсоток розпізнавання маневрів
За рівнем конкурентної боротьби – глобальний	Конкуренція зі світовими виробниками	Розвивати алгоритм
3. За галузевою ознакою – внутрішньо галузева	Вузькоспеціалізований алгоритм	Доповнювати

		сумісність з різноманітними системами захисту
4. За характером конкурентних переваг – цінова		Пристосовувати ціну

8.4 Розроблення ринкової стратегії проекту

Розробка ринкової стратегії передбачає опис цільових груп можливих клієнтів, простоту входу в сегменти збуту, готовність споживачів прийняти продукцію та орієнтовний попит в їх межах. Метою являється вибір основного напрямку, який є головним та найбільш сумісним з товаром.

Таблиця 8.8 – Вибір цільових груп потенційних споживачів

№ п/ п	Опис профілю цільової групи потенційни х клієнтів	Готовність споживачів сприйняти продукт	Орієнтовни й попит в межах цільової групи	Інтенсивність конкуренції в сегменті	Простот а входу у сегмент
1	Військові об'єкти	Безпека та захист це основа стабільного розвитку економіки, готовність буде безпосередньою	Орієнтовани й попит буде помірним	Безпосередню конкуренцію представляють інші перевірені часом продукти, загалом конкуренція на середньому рівні.	помірна
2	Комерційні установи				
3	Охоронні організації				
Які цільові групи обрано: Військові.					

Таблиця 8.9 – Визначення базової стратегії розвитку

№ п/ п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспромо жні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
1	Підбір спеціалістів високого рівня в основній або суміжних сферах розробки	Покращення точності алгоритму, розширення його можливостей	Точність та простота у використанні	Диференціація

Таблиця 8.10 – Визначення базової стратегії конкурентної поведінки

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
1	Ні.	50/50	Лише іноваційні технології з подальшим їх розвитком	Стратегія самостійної розробки з елементами імітаційної стратегії

Таблиця 8.11 – Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	Точність вимірів	Диференціація	Ціна	Доступність
				Простота
2	Швидкодія			Дієвість

8.5 Розроблення маркетингової програми стартап-проекту

Таблиця 8.12 – Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Ефективність	Більша Ефективність завдяки оптимізованому алгоритму	Висока ефективність при відносно низькій ціні

Таблиця 8.13 – Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
1. Товар за задумом	Адаптивний алгоритм траєкторної фільтрації руху малих БПЛ.		
2. Товару реальному виконані	Властивості/характеристики	М/Н м	Вр/Тх/Тл/Е/ Ор
	1. Зручність	Нм.	Вр, Тх
	2. Якість	М	Тх, Тл
	3. Швидкість	М	Тх, Тл.
	Якість: оновлення ПЗ, тестування кожного оновлення.		
	Пакування: Пз.		
3. Товар з підкріпленням	До продажу: передплата.		
	Після продажу: гарантія, технічна підтримка.		
За рахунок чого потенційний товар буде захищено від копіювання: Встановлено рут-права на ПЗ, що заблокує доступ до звичайного користувача.			

Таблиця 8.14 – Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Фінансування оновленого функціоналу ПЗ	Повне налаштування та встановлення	Роздрібна торгівля	Продаж напряму клієнту

Таблиця 8.15 – Визначення цінових меж

№ п/п	Рівень цін на товари- замінники	Рівень цін на товари- аналоги	Рівень доходів цільової групи споживачі	Верхня та нижня межа встановлення ціни на товар/послугу
1	2-4 тис.доларів	2-4 тис.доларів	Фінансування приватними інвесторами	4 тис. доларів

8.6 Висновки

- Алгоритм має всі шанси зайняти місце не лише на внутрішньому, але і на світовому ринку.

- Розглянуто методи просування продукту з урахуванням реалій сьогодення.

- Запропоновано оптимальні стратегії для подальшого розвитку.

- Потреба в захисті є актуально як для державних установ так і для бізнесу. Попит з кожним роком буде лише зростати.

ВИСНОВКИ

В результаті проведених досліджень було вирішено практичну наукову задачу підвищення ефективності локалізації малих БПЛА в системі відеоспостереження та FMCW далекоміру шляхом синтезу квазіоптимального алгоритму траєкторної фільтрації.

Квазіоптимальний адаптивний алгоритм отримали шляхом лінеаризації рівняння вимірювання радіальної швидкості, а також гаусівської апроксимації умовних екстрапольованих ПВ.

Аналіз точносних характеристик синтезованого алгоритму оцінювання параметрів руху БПЛА з розпізнаванням різних видів руху виконаний за допомогою статистичного моделювання на ЕОМ. Модель руху БПЛА описується стохастичною динамічною системою з випадковою структурою, яка враховує три основних види руху: зависання, майже лінійний та рух з маневрами.

Квазіоптимальний алгоритм є нелінійним. На відміну від оптимального алгоритму при його реалізації обчислюються лише перший та другий моменти умовних апостеріорних розподілів.

Оптимальний фільтр містить канали і відноситься до класу пристроїв зі зворотними зв'язками між каналами. Синтезований квазіоптимальний фільтр також включає M каналів і відноситься до класу пристроїв із зворотними зв'язками між каналами. Він відрізняється від оптимального пристрою фільтрації тим, що в блоках БФ і БП квазіоптимального пристрою обчислюються тільки математичні очікування і кореляційні матриці умовних щільностей ймовірностей.

Для усунення ефекту розходження на початковій ділянці роботи алгоритму запропоновано збільшувати дисперсію помилки вимірювань

радіальної швидкості на величину, пропорційну до дисперсії помилки екстраполяції радіальної швидкості.

Фільтр адаптується до динаміки руху дрона, що дозволяє підвищити точність екстраполяції в областях майже рівномірного руху та зависання, або навпаки руху з маневрами.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Unmanned aerial vehicle. Wikipedia.org.
URL: https://en.wikipedia.org/wiki/Unmanned_aerial_vehicle.
2. G. Singhal, B. Bansod and L. Mathew, “Unmanned aerial vehicle classification applications and challenges: A review,” Preprints, 2018, 2018110601.
3. M. N. Bento, “Unmanned aerial vehicles: An overview,” Inside GNSS, vol. 3, no. 3, pp. 54–61, 2008
4. S. R. Ganti and Y. Kim, “Implementation of Detection and Tracking Mechanism for Small UAS,” 2016 Int’l. Conf. Unmanned Aircraft Systems, June 2016, pp. 1254–60.
5. E. Vattapparamban et al., “Drones for Smart Cities: Issues in Cybersecurity, Privacy, and Public Safety,” 2016 Int’l. Wire-less Commun. and Mobile Computing Conf., Sept. 2016, pp. 216–21.
6. Salhi A., Jammoussi, Y.A., “Object tracking system using Camshift, Meanshift and Kalman Filter”, World Academy of Science, Engineering and Technology.Vol:6, 2012.
7. C. Premebida, G. Monteiro, U. Nunes, et al., A lidar and vision-based approach for pedestrian and vehicle detection and tracking, 2007 IEEE intelligent transportation systems conference, 1044–1049(2007)
8. J. Bergen, P. Anandan, K.J. Hanna, R. Hingorani. Hierarchical model-based motion estimation. ECCV ’92: Proceedings of the Second European Conference on Computer Vision, Springer-Verlag, pp. 237-252, 1992.
9. J.-L. Dugelay and H. Sanson. Differential methods for identification of 2-d and 3-d motion models in image sequences. Signal processing: image communication, Vol 7, pp 105-127, 1995.
10. M. A. Fischler and R. C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Comm. of the ACM, Vol 24, pp 381-395, 1981.

11. E. Kim, J. Han and H. Kim. A kalman-filtering method for 3d camera motion estimation from image sequences. International Conference on Image Processing, Santa Barbara, CA, Vol 3, pp 630 - 633, 1997.
12. J. Park, N. Yagi, K. Enami, K. Aizawa and M. Hatori. Estimation of camera parameters from image sequence for model-based video coding. IEEE Trans. Circuits and Systems for Video Technology Vol 4(3), pp 288-296, 1994.
13. Huang Y., Brennan P. V. FMCW based MIMO imaging radar for maritimenavigation // Progress In Electromagnetics Research – 2011. – Vol. 115. – P. 327-342
14. Жук, С.Я. «Спільна фільтрація змішаних марківських процесів у дискретному часі». Извєсия вузів. Радіоелектроніка, Т. 31, №1, С. 33-40, 1988.
15. Жук С.Я., Товкач І.О. Методи адаптивного оцінювання параметрів руху безпілотного літального апарату на основі вимірювань сенсорної мережі. -К.: «Політехніка», КПІ ім. Ігоря Сікорського, 2019, 172 с.
16. Застосування цифрової обробки сигналів: Пер.з англ. / За ред. А.М.Рязанцева. - М.: Світ, 1980.-552с.
17. Методичні рекомендації до виконання розділу магістерських дисертацій для студентів інженерних спеціальностей / За заг. ред. О.А. Гавриша. – Київ : НТУУ «КПІ», 2016. – 28 с.

ДОДАТОК А

ПРОГРАМА КВАЗІОПТИМАЛЬНОГО АЛГОРИТМУ ТРАЄКТОРНОЇ ФІЛЬТРАЦІЇ

```
%АДАПТИВНЫЙ ФИЛЬТР БПЛА
clear all;
clc
MonteCarlo = 100;
%количество реализаций
длина реализации
%T = 5 ;
темп поступления данных
ks=3;% 3сигма

%СКО ошибки измерения в метрах
pr = 20;
pb =0.1*3.14/180;
pe =0.1*3.14/180;
pv =10; pv2=pv^2; %1!!!
rg=180/3.14;
N = 9;
%количество параметров
NumHip = 3; %количество гипотез

load TRAЕК
NN=length(xtr)
Length=NN-2;
T,
A1 = 1; A2 = 1; A3 = 7; %A=1;
%vA1 = A1/T; vA2 = A2/T; vA3 = A3/T;
%матрица перехода
F(:, :, 3) = [1 T T^2/2 0 0 0 0 0 0;
0 1 T 0 0 0 0 0 0;
0 0 1 0 0 0 0 0 0;
0 0 0 1 T T^2/2 0 0 0;
0 0 0 0 1 T 0 0 0;
0 0 0 0 0 1 0 0 0;
0 0 0 0 0 0 1 T T^2/2;
0 0 0 0 0 0 0 1 T;
0 0 0 0 0 0 0 0 1];
F(:, :, 2) = [1 T 0 0 0 0 0 0 0;
0 1 0 0 0 0 0 0 0;
0 0 0 0 0 0 0 0 0;
0 0 0 1 T 0 0 0 0;
0 0 0 0 1 0 0 0 0;
0 0 0 0 0 1 0 0 0;
0 0 0 0 0 0 1 T 0;
0 0 0 0 0 0 0 1 0;
0 0 0 0 0 0 0 0 1];
F(:, :, 1) = [1 0 0 0 0 0 0 0 0;
0 0 0 0 0 0 0 0 0;
0 0 0 0 0 0 0 0 0;
0 0 0 1 0 0 0 0 0;
0 0 0 0 0 0 0 0 0;
0 0 0 0 0 0 0 0 0;
0 0 0 0 0 0 1 0 0;
0 0 0 0 0 0 0 0 0;
0 0 0 0 0 0 0 0 0];
```

```

G(:,:,3) = [ A3*T^3/6  0  0;
              A3*T^2/2  0  0;
              A3*T      0  0;
              0  A3*T^3/6  0;
              0  A3*T^2/2  0;
              0  A3*T      0;
              0  0  A3*T^3/6;
              0  0  A3*T^2/2;
              0  0  A3*T ];
G(:,:,2) = [ A2*T^2/2  0  0;
              A2*T      0  0;
              0  0  0;
              0  A2*T^2/2  0;
              0  A2*T      0;
              0  0  0;
              0  0  A2*T^2/2;
              0  0  A2*T;
              0  0  0 ];
G(:,:,1) = [ A1*T      0  0;
              0  0  0;
              0  0  0;
              0  A1*T      0;
              0  0  0;
              0  0  0;
              0  0  A1*T;
              0  0  0;
              0  0  0 ];

for j = 1:NumHip
    Q(:, :, j) = G(:, :, j) * G(:, :, j)';
end

%начальное положение цели 250^2= 62 km

ppp=0.96;
PP=[ ppp          0.5*(1-ppp)      0.5*(1-ppp);
     0.5*(1-ppp)      ppp          0.5*(1-ppp);
     0.5*(1-ppp)      0.5*(1-ppp)      ppp ]; %матрица
вероятностей переходов

%матожидание и дисперсия оптимального положения
xom=zeros(1,Length);
xod=zeros(1,Length);
yom=zeros(1,Length);
yod=zeros(1,Length);
zom=zeros(1,Length);
zod=zeros(1,Length);
%матожидание и дисперсия предсказаного RBE
rzm=zeros(1,Length);
rzd=zeros(1,Length);
bzm=zeros(1,Length);
bzd=zeros(1,Length);
ezm=zeros(1,Length);
ezd=zeros(1,Length);
% wzm=zeros(1,Length);
% wzd=zeros(1,Length);
vrzm=zeros(1,Length);
vrzd=zeros(1,Length);
%матожидание и дисперсия предсказаного положения
xzm=zeros(1,Length);
xzd=zeros(1,Length);

```

```

yzm=zeros(1,Length);
yzd=zeros(1,Length);
zzm=zeros(1,Length);
zzd=zeros(1,Length);
%матожидание и дисперсия оптимальной скорости
Vxom=zeros(1,Length);
Vxod=zeros(1,Length);
Vyom=zeros(1,Length);
Vyod=zeros(1,Length);
Vzom=zeros(1,Length);
Vzod=zeros(1,Length);
%матожидание и дисперсия измеренного положения
xim=zeros(1,Length);
xid=zeros(1,Length);
yim=zeros(1,Length);
yid=zeros(1,Length);
zim=zeros(1,Length);
zid=zeros(1,Length);

xomK = zeros(1,Length);
yomK = zeros(1,Length);
zomK = zeros(1,Length);
xzmK = zeros(1,Length);
yzmK = zeros(1,Length);
zzmK = zeros(1,Length);
VxomK = zeros(1,Length);
VyomK = zeros(1,Length);
VzomK = zeros(1,Length);
pRxK = zeros(1,Length);
pRyK = zeros(1,Length);
pRzK = zeros(1,Length);

% opt = zeros(1,Length);
Pobn1 = zeros(1,Length);
Pobn2 = zeros(1,Length);
Pobn3 = zeros(1,Length);

% podx = zeros(1,Length);
% RealHip = zeros(1,Length);

for i = 1 : MonteCarlo;

    U0 = [xtr(1); vxtr(1); ytr(1); vytr(1); ztr(1); vztr(1)];
    %начальное положение и скорость

    r=sqrt(U0(1)^2+U0(3)^2+U0(5)^2);
    b=mod(atan2( U0(3),U0(1) ),2*pi);
    e=asin(U0(5)/r);

    ri=r+pr*randn; bi=b+pb*randn; ei=e+pe*randn;

    xil=ri*cos(ei)*cos(bi);
    yil=ri*cos(ei)*sin(bi);
    zil=ri*sin(ei);

    U= [xtr(2); vxtr(2); ytr(2); vytr(2);ztr(2); vztr(2)];

    r=sqrt(U(1)^2+U(3)^2+U(5)^2);
    b=mod(atan2( U(3),U(1) ),2*pi);
    e=asin(U(5)/r);

```

```

ri=r+pr*randn;    bi=b+pb*randn;    ei=e+pe*randn;
xi=ri*cos(ei)*cos(bi);
yi=ri*cos(ei)*sin(bi);
zi=ri*sin(ei);

Ry=pr^2*cos(ei)^2*sin(bi)^2+ri^2*pe^2*sin(bi)^2*sin(ei)^2+ri^2*pb^2*cos(bi)^2*cos(ei)^2;
Rx=pr^2*cos(ei)^2*cos(bi)^2+ri^2*pe^2*cos(bi)^2*sin(ei)^2+ri^2*pb^2*sin(bi)^2*cos(ei)^2;
Rz=pr^2*sin(ei)^2+ri^2*pe^2*cos(ei)^2;
Rxy=0.5*sin(2*bi)*( cos(ei)^2*pr^2-ri^2*pb^2*cos(ei)^2+ri^2*pe^2*sin(ei)^2 );
Rxz=0.5*sin(2*ei)*cos(bi)*( pr^2-ri^2*pe^2 );
Ryz=0.5*sin(2*ei)*sin(bi)*( pr^2-ri^2*pe^2 );
%na4al'nie usloviya

A = A3;
Po(1,1)=Rx; Po(1,2)=Rx/T; Po(1,3)=0; Po(1,4)=Rxy; Po(1,5)=Rxy/T; Po(1,6)=0; Po(1,7)=Rxz; Po(1,8)=Rxz/T; Po(1,9)=0;
Po(2,1)=Po(1,2);
Po(2,2)=2*Rx/T^2+A^2*T^2; Po(2,3)=0; Po(2,4)=Rxy/T; Po(2,5)=2*Rxy/T^2; Po(2,6)=0;
Po(2,7)=Rxz/T; Po(2,8)=2*Rxz/T^2; Po(2,9)=0;
Po(3,1)=0; Po(3,2)=0; Po(3,3)=A^2; Po(3,4)=0; Po(3,5)=0; Po(3,6)=0; Po(3,7)=0; Po(3,8)=0; Po(3,9)=0;
Po(4,1)=Po(1,4); Po(4,2)=Po(2,4); Po(4,3)=0; Po(4,4)=Ry; Po(4,5)=Ry/T; Po(4,6)=0; Po(4,7)=Ryz; Po(4,8)=Ryz/T; Po(4,9)=0;
Po(5,1)=Po(1,5); Po(5,2)=Po(2,5); Po(5,3)=0; Po(5,4)=Po(4,5); Po(5,5)=2*Ry/T^2+A^2*T^2; Po(5,6)=0; Po(5,7)=Ryz/T; Po(5,8)=2*Ryz/T^2; Po(5,9)=0;
Po(6,1)=0; Po(6,2)=0; Po(6,3)=0; Po(6,4)=0; Po(6,5)=0; Po(6,6)=A^2; Po(6,7)=0; Po(6,8)=0; Po(6,9)=0;
Po(7,1)=Po(1,7); Po(7,2)=Po(2,7); Po(7,3)=Po(3,7); Po(7,4)=Po(4,7); Po(7,5)=Po(5,7); Po(7,6)=Po(6,7); Po(7,7)=Rz; Po(7,8)=Rz/T; Po(7,9)=0;
Po(8,1)=Po(1,8); Po(8,2)=Po(2,8); Po(8,3)=Po(3,8); Po(8,4)=Po(4,8); Po(8,5)=Po(5,8); Po(8,6)=Po(6,8); Po(8,7)=Po(7,8); Po(8,8)=2*Rz/T^2+A^2*T^2; Po(8,9)=0;
Po(9,1)=0; Po(9,2)=0; Po(9,3)=0; Po(9,4)=0; Po(9,5)=0; Po(9,6)=0; Po(9,7)=0; Po(9,8)=0; Po(9,9)=A^2;

for j0 = 1:NumHip
%   if          (1 == j0) A = A3;
%   elseif      (2 == j0) A = A3;
%   elseif      (3 == j0) A = A3;
%   end

    x(:,j0) = [xi, (xi-xil)/T, 0, yi, (yi-yil)/T, 0, zi, (zi-zil)/T, 0]; %
    %начальное оптимальное значение параметров

%   Po(2,2)=2*Rx/T^2+A^2*T^2; Po(3,3)=A^2; Po(5,5)=2*Ry/T^2+A^2*T^2;
%   Po(6,6)=A^2; Po(8,8)=2*Rz/T^2+A^2*T^2; Po(9,9)=A^2;
%   P(:, :, j0) = zeros(N,N);
%   P(:, :, j0) = Po;
%   % начальная оптимальная матрица ошибок оценки
%   %Q(:, :, j0) = G(:, :, j0) * G(:, :, j0)';
%   W(j0) = 1 / NumHip;
end

for step = 1:Length
    U = [xtr(step+2); vxtr(step+2); ytr(step+2); vytr(step+2);
ztr(step+2); vztr(step+2)];

    xm(step)=U(1); ym(step)=U(3); zm(step)=U(5); % реальное значение
    %положения

```

```

%vm(step) = U(2); % реальное значение скорости

r=sqrt(U(1)^2+U(3)^2+U(5)^2);
b=mod(atan2(U(3),U(1)),2*pi);
e=asin(U(5)/r);
vr=(U(1)*U(2)+U(3)*U(4)+U(5)*U(6))/r; %2!!!

%rstep(step) = r;
ri=r+pr*randn; bi=b+pb*randn; ei=e+pe*randn;
vri=vr+pv*randn; % реальное + ошибка измерения * случ.велич 3!!!
xi=ri*cos(ei)*cos(bi);
yi=ri*cos(ei)*sin(bi); %4!!!
zi=ri*sin(ei);

y=[xi;yi;zi;vri]; %измеренное положение
imx(step)=xi; imy(step)=yi; imz(step)=zi;

[xim(step),xid(step)]=disper(i,xi-U(1),xim(step),xid(step));
[yim(step),yid(step)]=disper(i,yi-U(3),yim(step),yid(step));
[zim(step),zid(step)]=disper(i,zi-U(5),zim(step),zid(step));

Ry=pr^2*cos(ei)^2*sin(bi)^2+ri^2*pe^2*sin(bi)^2*sin(ei)^2+ri^2*pb^2*cos(bi)^2*cos(ei)^2;

Rx=pr^2*cos(ei)^2*cos(bi)^2+ri^2*pe^2*cos(bi)^2*sin(ei)^2+ri^2*pb^2*sin(bi)^2*cos(ei)^2;
Rz=pr^2*sin(ei)^2+ri^2*pe^2*cos(ei)^2;
Rxy=0.5*sin(2*bi)*[cos(ei)^2*pr^2-ri^2*pb^2*cos(ei)^2+ri^2*pe^2*sin(ei)^2];
Rxz=0.5*sin(2*ei)*cos(bi)*[pr^2-ri^2*pe^2];
Ryz=0.5*sin(2*ei)*sin(bi)*[pr^2-ri^2*pe^2];
pRx(step)=Rx; pRy(step)=Ry; pRz(step)=Rz;

for j0 = 1:NumHip
    R(:,j0)= [ Rx Rxz 0; %
               Rxy Ry Ryz 0;
               Rxx Ryx Rz 0;
               0 0 0 pv2 ]; %5!!!
end;

for j0 = 1:NumHip
    Wz(j0) = 0;
    for j1 = 1:NumHip
        Wz(j0) = Wz(j0) + PP(j1,j0) * W(j1); % экстраполированная
    end;
end;

% Wz, pause
for j0=1:NumHip
    xz(:,j0)=zeros(N,1);
    for j1=1:NumHip
        xz(:,j0) = xz(:,j0) + PP(j1,j0) * W(j1) * F(:,j0) * x(:,j1) /
    end;
end;

% xz,

```

```

for j0 = 1:NumHip
    Pz(:, :, j0) = zeros(N, N);
    for j1 = 1:NumHip
        Pz(:, :, j0) = Pz(:, :, j0) + ( F(:, :, j0)*P(:, :, j1)*F(:, :, j0)' +
Q(:, :, j0) + ( F(:, :, j0)*x(:, j1) - xz(:, j0) ) * ( F(:, :, j0)*x(:, j1) -
xz(:, j0) )' ) * PP(j1, j0) * W(j1) / Wz(j0);
    end;
end;
% Pz,

for j0=1:NumHip %6!!!
    H(:, :, j0) = zeros(N-5, N);    H(1, 1, j0)=1;    H(2, 4, j0)=1;    H(3, 7, j0)=1;

    rz(j0)=sqrt(xz(1, j0)^2+xz(4, j0)^2+xz(7, j0)^2);
    vrz(j0)=(xz(1, j0)*xz(2, j0) + xz(4, j0)*xz(5, j0) + xz(7, j0)*xz(8, j0)) / rz(j0);

    H(4, 1, j0) = (xz(2, j0)*rz(j0) - vrz(j0)*xz(1, j0)) / rz(j0)^2;
    H(4, 2, j0) = xz(1, j0) / rz(j0);
    H(4, 4, j0) = (xz(5, j0)*rz(j0) - vrz(j0)*xz(4, j0)) / rz(j0)^2;
    H(4, 5, j0) = xz(4, j0) / rz(j0);
    H(4, 7, j0) = (xz(8, j0)*rz(j0) - vrz(j0)*xz(7, j0)) / rz(j0)^2;
    H(4, 8, j0) = xz(7, j0) / rz(j0);

    xzi(:, j0) = [ xz(1, j0), xz(4, j0), xz(7, j0), vrz(j0) ];
end;

pp=0;
for j0=1:NumHip
    if (step > 0) %7!!!
        Rnew(:, :, j0) = H(:, :, j0) * Pz(:, :, j0) * H(:, :, j0)';
        R(4, 4, j0) = 0.8*Rnew(4, 4, j0) + R(4, 4, j0); %*exp(-1*step)
    end;

    K(:, :, j0) = Pz(:, :, j0) * H(:, :, j0)' / (H(:, :, j0) * Pz(:, :, j0) *
H(:, :, j0)' + R(:, :, j0) ); %
    x(:, j0) = xz(:, j0) + K(:, :, j0) * (y - xzi(:, j0));
    P(:, :, j0) = Pz(:, :, j0) - K(:, :, j0) * H(:, :, j0) * Pz(:, :, j0);
    D(:, :, j0) = H(:, :, j0) * Pz(:, :, j0) * H(:, :, j0)' + R(:, :, j0); %
    p(j0) = Wz(j0) * exp( -0.5*(y - xzi(:, j0))' / D(:, :, j0) * (y -
xzi(:, j0)) ) / ( 2 * pi *sqrt( det( D(:, :, j0) ) ) );
    pp = pp + p(j0); % сумма вероятностей
end;

suma_weroyat = 0;
for j0=1:NumHip
    W(j0) = p(j0) / pp;
    suma_weroyat = suma_weroyat + W(j0);
end;
% W, pause

% нахождение максимальной вероятности и оптимального индекса
opt_j = 1; maxW = W(1);
for j0=2:NumHip
    if (W(j0) > maxW)
        maxW = W(j0);
        opt_j = j0;
    end
end;
%opt(step) = opt_j ;

```

```

%if (opt_j ~= RealHip(step) ) Pem (step) = Pem (step) +1; end % если
индекс оптимальной матр не равняется индексу ист траектории, то подсчитать
сколько раз он не равняется на длине траектории
if (opt_j == 1 ) Pobn1 (step) = Pobn1 (step) +1; end
if (opt_j == 2 ) Pobn2 (step) = Pobn2 (step) +1; end
if (opt_j == 3 ) Pobn3 (step) = Pobn3 (step) +1; end

% оптимальные векторы и матрицы с гипотез
U_prog = xz(:,opt_j);
U_opt = x( :, opt_j);
P_opt = P( :, :, opt_j) ;
P_prog = Pz( :, :, opt_j) ;
pvr=Rnew(4,4,opt_j); pvrz(step)= pvr; %!!!!!!!!!!!!

%
a1=U_prog(1)/sqrt(U_prog(1)^2+U_prog(4)^2+U_prog(7)^2);a2=U_prog(4)/sqrt(U_prog(1)^2+U_prog(4)^2+U_prog(7)^2);a3=U_prog(7)/sqrt(U_prog(1)^2+U_prog(4)^2+U_prog(7)^2);
% prz1(step)=a1^2*P_prog(1,1)+a2^2*P_prog(4,4)+a3^2*P_prog(7,7);
%
prz2(step)=prz1(step)+2*a1*a2*P_prog(1,4)+2*a1*a3*P_prog(1,7)+2*a2*a3*P_prog(4,7);
%
%
pbz1(step)=1/((1+(U_prog(4)/U_prog(1))^2)*U_prog(1))^2*(P_prog(4,4)+P_prog(1,1)*(U_prog(4)/U_prog(1))^2);
% pbz2(step)=1/((1+(U_prog(4)/U_prog(1))^2)*U_prog(1))^2*(P_prog(4,4)-2*U_prog(4)*P_prog(1,4)/U_prog(1)+P_prog(1,1)*(U_prog(4)/U_prog(1))^2);
%
% rxy=sqrt(U_prog(1)^2+U_prog(4)^2); ec=1/(1+(U_prog(7)/rxy)^2);
ez=ec/rxy; ex=-ec*U_prog(7)*U_prog(1)/rxy^3;ey=-ec*U_prog(7)*U_prog(4)/rxy^3;
% pez1(step)=ex^2*P_prog(1,1)+ey^2*P_prog(4,4)+ez^2*P_prog(7,7);
%
pez2(step)=pez1(step)+2*ex*ey*P_prog(1,4)+2*ex*ez*P_prog(1,7)+2*ey*ez*P_prog(4,7);
%
% wz=sqrt(U_prog(2)^2+U_prog(5)^2); w1=U_prog(2)/wz; w2=U_prog(5)/wz;
% pwz1(step)=w1^2*P_prog(2,2)+w2^2*P_prog(5,5);
% pwz2(step)=pwz1(step)+2*w1*w2*P_prog(2,5);
%
%
[rzm(step),rzd(step)]=disper(i,sqrt(U_prog(1)^2+U_prog(4)^2+U_prog(7)^2)-r,rzm(step),rzd(step));
% bz=atan2(U_prog(4),U_prog(1)); db=bz-b; if abs(bz-b)>2 db=abs(bz)-abs(b);end
% %if abs(abs(bz(1))-abs(bmi(n)))>30 bz(1)=bmi(n);end
% [bzm(step),bzd(step)]=disper(i,db,bzm(step),bzd(step));
% mbz(step)=bz;
%
[ezm(step),ezd(step)]=disper(i,atan2(U_prog(7),sqrt(U_prog(1)^2+U_prog(4)^2))-e,ezm(step),ezd(step));
% [wzm(step),wzd(step)]=disper(i,wz-sqrt(U_opt(2)^2+U_opt(5)^2),wzm(step),wzd(step));

a1=U_opt(1)/sqrt(U_opt(1)^2+U_opt(4)^2+U_opt(7)^2);a2=U_opt(4)/sqrt(U_opt(1)^2+U_opt(4)^2+U_opt(7)^2);a3=U_opt(7)/sqrt(U_opt(1)^2+U_opt(4)^2+U_opt(7)^2);
prz1(step)=a1^2*P_opt(1,1)+a2^2*P_opt(4,4)+a3^2*P_opt(7,7);

prz2(step)=prz1(step)+2*a1*a2*P_opt(1,4)+2*a1*a3*P_opt(1,7)+2*a2*a3*P_opt(4,7);
);

```

```

pbz1(step)=1/((1+(U_opt(4)/U_opt(1))^2)*U_opt(1))^2*(P_opt(4,4)+P_opt(1,1)*(U_opt(4)/U_opt(1))^2);
pbz2(step)=1/((1+(U_opt(4)/U_opt(1))^2)*U_opt(1))^2*(P_opt(4,4)-2*U_opt(4)*P_opt(1,4)/U_opt(1)+P_opt(1,1)*(U_opt(4)/U_opt(1))^2);

rxy=sqrt(U_opt(1)^2+U_opt(4)^2); ec=1/(1+(U_opt(7)/rxy)^2); ez=ec/rxy;
ex=-ec*U_opt(7)*U_opt(1)/rxy^3; ey=-ec*U_opt(7)*U_opt(4)/rxy^3;
pez1(step)=ex^2*P_opt(1,1)+ey^2*P_opt(4,4)+ez^2*P_opt(7,7);

pez2(step)=pez1(step)+2*ex*ey*P_opt(1,4)+2*ex*ez*P_opt(1,7)+2*ey*ez*P_opt(4,7);

% wz=sqrt(U_opt(2)^2+U_opt(5)^2); w1=U_opt(2)/wz; w2=U_opt(5)/wz;
% pwz1(step)=w1^2*P_opt(2,2)+w2^2*P_opt(5,5);
% pwz2(step)=pwz1(step)+2*w1*w2*P_opt(2,5);

[rzm(step), rzd(step)] = disper(i, sqrt(U_opt(1)^2+U_opt(4)^2+U_opt(7)^2) - r, rzm(step), rzd(step));

[vrzm(step), vrzd(step)] = disper(i, (U_opt(1)*U_opt(2)+U_opt(4)*U_opt(5)+U_opt(7)*U_opt(8))/sqrt(U_opt(1)^2+U_opt(4)^2+U_opt(7)^2) - vr, vrzm(step), vrzd(step));
bz=atan2(U_opt(4), U_opt(1)); db=bz-b; if abs(bz-b)>2 db=abs(bz)-abs(b); end
%if abs(abs(bz(1))-abs(bmi(n)))>30 bz(1)=bmi(n); end
[bzm(step), bzd(step)] = disper(i, db, bzm(step), bzd(step));
mbz(step)=bz;

[ezm(step), ezd(step)] = disper(i, atan2(U_opt(7), sqrt(U_opt(1)^2+U_opt(4)^2)) - e, ezm(step), ezd(step));
%[wzm(step), wzd(step)] = disper(i, wz - sqrt(U(2)^2+U(4)^2), wzm(step), wzd(step));

% ВЫСТАВЛЕНИЕ СТРОБОВ
zmx(step) = U_prog(1); zmy(step) = U_prog(4);
zmz(step) = U_prog(7); sy = ks*sqrt(P_prog(4,4) + Ry);
sx = ks*sqrt(P_prog(1,1) + Rx);
sz = ks*sqrt(P_prog(7,7) + Rz);
sx1m(step) = U_prog(1) + sx; sy1m(step) = U_prog(4) + sy;
sz1m(step) = U_prog(7) + sz;
sx2m(step) = U_prog(1) - sx; sy2m(step) = U_prog(4) - sy;
sz2m(step) = U_prog(7) - sz;

omx(step) = U_opt(1); omy(step) = U_opt(4); omz(step) = U_opt(7);

podx(step) = P_opt(1,1); pody(step) = P_opt(4,4);
podz(step) = P_opt(7,7);
podVx(step) = P_opt(2,2); podVy(step) = P_opt(5,5);
podVz(step) = P_opt(8,8);
pzdx(step) = P_prog(1,1); pzdy(step) = P_prog(4,4);
pzdz(step) = P_prog(7,7);

[pRxK(step)] = mat_och(i, sqrt(Rx), pRxK(step));
[pRyK(step)] = mat_och(i, sqrt(Ry), pRyK(step));
[pRzK(step)] = mat_och(i, sqrt(Rz), pRzK(step));

[xom(step), xod(step)] = disper(i, U_opt(1) - U(1), xom(step), xod(step));
[xomK(step)] = mat_och(i, sqrt(P_opt(1,1)), xomK(step));
[yom(step), yod(step)] = disper(i, U_opt(4) - U(3), yom(step), yod(step));

```

```

[yomK(step)] = mat_och(i, sqrt(P_opt(4,4)) , yomK(step));
[zom(step),zod(step)] = disper(i, U_opt(7) - U(5),zom(step),zod(step));
[zomK(step)] = mat_och(i, sqrt(P_opt(7,7)),zomK(step));

[xzm(step),xzd(step)] = disper(i,U_prog(1)-U(1),xzm(step),xzd(step));
[xzmK(step)] = mat_och(i, sqrt(P_prog(1,1)) , xzmK(step));
[yzm(step),yzd(step)] = disper(i,U_prog(4)-U(3),yzm(step),yzd(step));
[yzmK(step)] = mat_och(i, sqrt(P_prog(4,4)) , yzmK(step));
[zzm(step),zzd(step)] = disper(i,U_prog(7)-U(5),zzm(step),zzd(step));
[zzmK(step)] = mat_och(i, sqrt(P_prog(7,7)) , zzmK(step));

[Vxom(step),Vxod(step)] = disper(i,U_opt(2)-U(2),Vxom(step),Vxod(step));
[VxomK(step)] = mat_och(i, sqrt(P_opt(2,2)) , VxomK(step));
[Vyom(step),Vyod(step)] = disper(i,U_opt(5)-U(4),Vyom(step),Vyod(step));
[VyomK(step)] = mat_och(i, sqrt(P_opt(5,5)) , VyomK(step));
[Vzom(step),Vzod(step)] = disper(i,U_opt(8)-U(6),Vzom(step),Vzod(step));
[VzomK(step)] = mat_och(i, sqrt(P_opt(8,8)) , VzomK(step));

% [R3od(step)] = disper3(i, U_opt(1)-U(1), U_opt(3)-U(3), U_opt(5)-U(5),
R3od(step)); % kak v stat'e
% [Vod(step)] = disper3(i, U_opt(2)-U(2), U_opt(4)-U(4), U_opt(6)-U(6),
Vod(step)); % kak v stat'e

end;
end

figure(21);
%subplot(224);
plot3(xm,ym,zm,'b','LineWidth',2), title('Traektoriya XcYcZc'),
xlim([-500,400]);zlim([0,300]);ylim([0,1000]);
xlabel('x'),ylabel('y'),zlabel('z');
grid on;

figure(1);
subplot(111);
plot(xm,ym,'b','LineWidth',2);
xlabel('X '),ylabel('Y '); grid on
%pause

T=1;
t=(1:Length)*T-T;
figure(2);
subplot(111);
plot(t,zm,'b','LineWidth',2),ylabel('Z ');
xlabel('t,c'),ylabel('Z (M)');grid on
%pause

figure(3);
subplot(311);
%plot(t,rzm,':g',t,sqrt(rzd),'b',t,sqrt(prz1),'r',t,sqrt(prz2),'m'),title('OS
HIBKI OCENKI R'), %'LineWidth',2,
plot(t,rzm,':g', t,sqrt(rzd),'--b',t,sqrt(prz2),'-m','LineWidth'
,2),title('OSHIBKI OCENKI R'),
xlabel('t,c'),ylabel('MO, CKO (M)');grid on
subplot(312);
%plot(t,bzm*rg,':g',t,sqrt(bzd)*rg,'b',t,sqrt(pbz1)*rg,'r',t,sqrt(pbz2)*rg,'m
'),title('OSHIBKI OCENKI B'),
plot(t,bzm*rg,':g',t,sqrt(bzd)*rg,'--b',t,sqrt(pbz2)*rg,'-m','LineWidth'
,2),title('OSHIBKI OCENKI B'),
xlabel('t,c'),ylabel('MO, CKO (Grad)');grid on
subplot(313);

```

```

%plot(t,ezm*rg,'g',t,sqrt(ezd)*rg,'b',t,sqrt(pez1)*rg,'r',t,sqrt(pez2)*rg,'m'
),title('OSHIBKI OCENKI E'),
plot(t,ezm*rg,':g',t,sqrt(ezd)*rg,'--b',t,sqrt(pez2)*rg,'-m','LineWidth'
,2),title('OSHIBKI OCENKI E'),
xlabel('t,c'),ylabel('MO, CKO (Grad)'); grid on
%pause

figure(4);
subplot(111);
%plot(t,rzm,':g',t,sqrt(rzd),'b',t,sqrt(prz1),'r',t,sqrt(prz2),'m'),title('OS
HIBKI OCENKI VR'), %'LineWidth',2,
plot(t,vrzm,':g',t,sqrt(vrzd),'--b',t,sqrt(pvrz),'-m','LineWidth'
,2),title('OSHIBKI OCENKI Rv'),
xlabel('t,c'),ylabel('MO, CKO (M)');grid on

figure(6);
subplot(311);
plot(t,imx,'-g',t,sx1m,'--m',t,sx2m,'--m','LineWidth',2),title('Strob
IstinTr Izmeren X'), %t,xm,'-k',
xlabel('t,c'),ylabel('X (M)'); grid on
subplot(312);
plot(t,imy,'-g',t,sylm,'--m',t,sy2m,'--m','LineWidth',2),title('Strob
IstinTr Izmeren Y'), %t,ym,'-k',
xlabel('t,c'),ylabel('Y (M)'); grid on
subplot(313);
plot(t,imz,'-g',t,sz1m,'--m',t,sz2m,'--m','LineWidth',2),title('Strob
IstinTr Izmeren Z'),%t,zm,'-k'
xlabel('t,c'),ylabel('Z (M)'); grid on
%pause

% figure;
% subplot(311);
% plot(t,xm,'-k',t,omx,'--k',t,zmx,'-.k',t,imx,':k'),title('IstinTr IzmerenTr
PrognozTr OcenkaTr X'),
% xlabel('t,c'),ylabel('X ');grid on
% subplot(312);
% plot(t,ym,'-k',t,omy,'--k',t,zmy,'-.k',t,imy,':k'),title('IstinTr IzmerenTr
PrognozTr OcenkaTr Y'),
% xlabel('t,c'),ylabel('Y ');grid on
% subplot(313);
% plot(t,zm,'-k',t,omz,'--k',t,zmz,'-.k',t,imz,':k'),title('IstinTr IzmerenTr
PrognozTr OcenkaTr Z'),
% xlabel('t,c'),ylabel('Z ');grid on
% pause
%
% figure;
% subplot(311);
% plot(t,xim,':k',t,sqrt(xid),'--k',t,pRxK,'-k'),title('OSHIBKI IZMERENIY
X'),
% xlabel('t,c'),ylabel('SKO');grid on
% subplot(312);
% plot(t,yim,':k',t,sqrt(yid),'--k',t,pRyK,'-k'),title('OSHIBKI IZMERENIY
Y'),
% xlabel('t,c'),ylabel('SKO'); grid on
% subplot(313);
% plot(t,zim,':k',t,sqrt(zid),'--k',t,pRzK,'-k'),title('OSHIBKI IZMERENIY
Z'),
% xlabel('t,c'),ylabel('SKO'); grid on
% pause
%
figure(7);

```

```

subplot(311);
plot(t,xzm,'g',t,sqrt(xzd),'--b',t,xzmK,'-m','LineWidth',2),title('OSIBKI
PRGNOZA X'),
xlabel('t,c'),ylabel('MO, CKO (M)');grid on
subplot(312);
plot(t,zym,'g',t,sqrt(yzd),'--b',t,zymK,'-m','LineWidth',2),title('OSIBKI
PRGNOZA Y'),
xlabel('t,c'),ylabel('MO, CKO (M)'); grid on
subplot(313);
plot(t,zzm,'g',t,sqrt(zzd),'--b',t,zzmK,'-m','LineWidth',2),title('OSIBKI
PRGNOZA Z'),
xlabel('t,c'),ylabel('MO, CKO (M)'); grid on
%pause
%
figure(8);
subplot(311);
plot(t,xom,'g',t,sqrt(xod),'--b',t,xomK,'-m','LineWidth',2),title('OSIBKI
OCENKI X'),
xlabel('t,c'),ylabel('MO, CKO (M)'); grid on
subplot(312);
plot(t,yom,'g',t,sqrt(yod),'--b',t,yomK,'-m','LineWidth',2),title('OSIBKI
OCENKI Y'),
xlabel('t,c'),ylabel('MO, CKO (M)'); grid on
subplot(313);
plot(t,zom,'g',t,sqrt(zod),'--b',t,zomK,'-m','LineWidth',2),title('OSIBKI
OCENKI Z'),
xlabel('t,c'),ylabel('MO, CKO (M)^'); grid on
%pause

% figure;
% subplot(311);
% plot(t,pRxK,'-k',t,xzmK,'--k',t,xomK,':k'),title('Teor SCO Izmeren Prognoz
Ocenka X'),
% xlabel('t,c'),ylabel('SKO '); grid on
% subplot(312);
% plot(t,pRyK,'-k',t,zymK,'--k',t,yomK,':k'),title('Teor SCO Izmeren Prognoz
Ocenka Y'),
% xlabel('t,c'),ylabel('SKO'); grid on
% subplot(313);
% plot(t,pRzK,'-k',t,zzmK,'--k',t,zomK,':k'),title('Teor SCO Izmeren Prognoz
Ocenka Z'),
% xlabel('t,c'),ylabel('SKO'); grid on
% pause

figure(9);
subplot(311);
plot(t,Vxom,'g',t,sqrt(Vxod),'--b',t,VxomK,'-m','LineWidth',
2),title('OSIBKI OCENKI VX'),
xlabel('t,c'),ylabel('MO, CKO (M/C) '); grid on
subplot(312);
plot(t,Vyom,'g',t,sqrt(Vyod),'--b',t,VyomK,'-m','LineWidth',
2),title('OSIBKI OCENKI VY '),
xlabel('t,c'),ylabel('MO, CKO (M/C)'); grid on
subplot(313);
plot(t,Vzom,'g',t,sqrt(Vzod),'--b',t,VzomK,'-m','LineWidth',
2),title('OSIBKI OCENKI VZ '),
xlabel('t,c'),ylabel('MO, CKO (M/C)'); grid on
%pause

figure(10);
subplot(111);

```

```

plot(t, Pobn1/MonteCarlo, ':g', t, Pobn2/MonteCarlo, '-b', t,
Pobn3/MonteCarlo, '--r','LineWidth',2 ),
xlabel('t,c'),ylabel('P1,P2,P3'); grid on
%pause

```

```

function [m1,d1] = disper(i,x,m,d)
b=x-m; b=b*b;
c=i-1;
if i>2,
d1=(d*(c-1)+b)/c;
else
d1=b+d;
end
m1=(m*c+x)/i;

```

```

function [m1] = mat_och(i,x,m)
c=i-1;
m1=(m*c+x)/i;

```

```

function [d1] = disper2(i,x1,x2,x3,d)
b1=x1; b1=b1*b1;
b2=x2; b2=b2*b2;
b3=x3; b3=b3*b3;
c=i-1;
if i>2,
d1=(d*(c-1)+b1+b2+b3)/c;
else
d1=b1+b2+b3+d;
end
% m1=(m*c+x)/i;

```

ДОДАТОК Б

ПРОГРАМА АНАЛІЗУ ТОЧНОСНИХ ХАРАКТЕРИСТИК КОМБІНОВАНОЇ СИСТЕМИ СПОСТЕРЕЖЕННЯ

Z:

```
N=60;
%pr=300;
%pb=0.166*3.14/180;
%pe=0.166*3.14/180;
pr=30;
% pr=15;
pb=0.3*3.14/180;
pe=0.3*3.14/180;

r=1000;
r1=550;
r2=100;

z=0;
hold off

ad=6.28/N;
aa=0;
% for i=0 : 0;

ri=r;
ri1=r1;
ri2=r2;
bi=pi/3;

for n=1 : N/2+1; %/2;
aa=(n-1)*ad;
alm(n)=aa;
ei=aa;
rrm(n)=ri;
rrm1(n)=ri1;
rrm2(n)=ri2;

Ry(n)=pr^2*cos(ei)^2*sin(bi)^2+ri^2*pe^2*sin(bi)^2*sin(ei)^2+ri^2*pb^2*cos(bi)^2*cos(ei)^2;
Rx(n)=pr^2*cos(ei)^2*cos(bi)^2+ri^2*pe^2*cos(bi)^2*sin(ei)^2+ri^2*pb^2*sin(bi)^2*cos(ei)^2;
Rz(n)=pr^2*sin(ei)^2+ri^2*pe^2*cos(ei)^2;
Rxy(n)=0.5*sin(2*bi)*[cos(ei)^2*pr^2-ri^2*pb^2+ri^2*pe^2*sin(ei)^2];
Ryz(n)=0.5*sin(2*ei)*sin(bi)*[pr^2-ri^2*pe^2];
Rxz(n)=0.5*sin(2*ei)*cos(bi)*[pr^2-ri^2*pe^2];
sigmz(n)=sqrt(Rz(n));
sigmyz(n)=abs(Ryz(n))/(sqrt(Ry(n))*sqrt(Rz(n)));
sigmxz(n)=abs(Rxz(n))/(sqrt(Rx(n))*sqrt(Rz(n)));

Ry(n)=pr^2*cos(ei)^2*sin(bi)^2+ri1^2*pe^2*sin(bi)^2*sin(ei)^2+ri1^2*pb^2*cos(bi)^2*cos(ei)^2;
```

```

Rx(n)=pr^2*cos(ei)^2*cos(bi)^2+ri1^2*pe^2*cos(bi)^2*sin(ei)^2+ri1^2*pb^2*sin(
bi)^2*cos(ei)^2;
Rz(n)=pr^2*sin(ei)^2+ri1^2*pe^2*cos(ei)^2;
Rxy(n)=0.5*sin(2*bi)*[cos(ei)^2*pr^2-ri1^2*pb^2+ri1^2*pe^2*sin(ei)^2];
Ryz(n)=0.5*sin(2*ei)*sin(bi)*[pr^2-ri1^2*pe^2];
Rxz(n)=0.5*sin(2*ei)*cos(bi)*[pr^2-ri1^2*pe^2];
sigmz1(n)=sqrt(Rz(n));
sigmyz1(n)=abs(Ryz(n))/(sqrt(Ry(n))*sqrt(Rz(n)));
sigmxz1(n)=abs(Rxz(n))/(sqrt(Rx(n))*sqrt(Rz(n)));

Ry(n)=pr^2*cos(ei)^2*sin(bi)^2+ri2^2*pe^2*sin(bi)^2*sin(ei)^2+ri2^2*pb^2*cos(
bi)^2*cos(ei)^2;
Rx(n)=pr^2*cos(ei)^2*cos(bi)^2+ri2^2*pe^2*cos(bi)^2*sin(ei)^2+ri2^2*pb^2*sin(
bi)^2*cos(ei)^2;
Rz(n)=pr^2*sin(ei)^2+ri2^2*pe^2*cos(ei)^2;
Rxy(n)=0.5*sin(2*bi)*[cos(ei)^2*pr^2-ri2^2*pb^2+ri2^2*pe^2*sin(ei)^2];
Ryz(n)=0.5*sin(2*ei)*sin(bi)*[pr^2-ri2^2*pe^2];
Rxz(n)=0.5*sin(2*ei)*cos(bi)*[pr^2-ri2^2*pe^2];
sigmz2(n)=sqrt(Rz(n));
sigmyz2(n)=abs(Ryz(n))/(sqrt(Ry(n))*sqrt(Rz(n)));
sigmxz2(n)=abs(Rxz(n))/(sqrt(Rx(n))*sqrt(Rz(n)));
end
% polar(alm,sigmz,'r');
% hold on
% polar(alm,sigmz1,'r');
% hold on
% polar(alm,sigmz2,'r');
% pause

polar(alm,sigmyz,'r');
hold on
polar(alm,sigmyz1,'r');
hold on
% polar(alm,sigmyz2,'r');
% pause

hold on
% end

```

XY:

```

N=60;
pr=30;
% pr=15;
pb=0.3*3.14/180;
pe=0.3*3.14/180;
%pr=600;
%pb=0.332*3.14/180;
%pe=0.332*3.14/180;

r=1000;
r1=550;
r2=100;
dr=450;
z=0;
hold off

ad=6.28/N;
aa=0;
% for i=0 : 2;

```

```

ri=r;
ril=r1;
ri2=r2;

ei=asin(z/ri);
eil=asin(z/ril);
ei2=asin(z/ri2);

for n=1 : N+1; %/2;
aa=(n-1)*ad;

bi=aa;
rrm(n)=ri;
rrm1(n)=ril;
rrm2(n)=ri2;
alm(n)=aa;

Ry(n)=pr^2*cos(ei)^2*sin(bi)^2+ri^2*pe^2*sin(bi)^2*sin(ei)^2+ri^2*pb^2*cos(bi)^2*cos(ei)^2;
Rx(n)=pr^2*cos(ei)^2*cos(bi)^2+ri^2*pe^2*cos(bi)^2*sin(ei)^2+ri^2*pb^2*sin(bi)^2*cos(ei)^2;
Rz(n)=pr^2*sin(ei)^2+ri^2*pe^2*cos(ei)^2;
Rxy(n)=0.5*sin(2*bi)*[cos(ei)^2*pr^2-ri^2*pb^2+ri^2*pe^2*sin(ei)^2];
Ryz(n)=0.5*sin(2*ei)*sin(bi)*[pr^2-ri^2*pe^2];
Rxz(n)=0.5*sin(2*ei)*cos(bi)*[pr^2-ri^2*pe^2];
sigm2x(n)=sqrt(Rx(n));
sigm2y(n)=sqrt(Ry(n));
sigm2xy(n)=abs(Rxy(n))/(sqrt(Rx(n))*sqrt(Ry(n)));

Ry1(n)=pr^2*cos(eil)^2*sin(bi)^2+r1l^2*pe^2*sin(bi)^2*sin(eil)^2+r1l^2*pb^2*cos(bi)^2*cos(eil)^2;
Rx1(n)=pr^2*cos(eil)^2*cos(bi)^2+r1l^2*pe^2*cos(bi)^2*sin(eil)^2+r1l^2*pb^2*sin(bi)^2*cos(eil)^2;
Rz1(n)=pr^2*sin(eil)^2+r1l^2*pe^2*cos(eil)^2;
Rxy1(n)=0.5*sin(2*bi)*[cos(eil)^2*pr^2-r1l^2*pb^2+r1l^2*pe^2*sin(eil)^2];
Ryz1(n)=0.5*sin(2*eil)*sin(bi)*[pr^2-r1l^2*pe^2];
Rxz1(n)=0.5*sin(2*eil)*cos(bi)*[pr^2-r1l^2*pe^2];
sigm2x1(n)=sqrt(Rx1(n));
sigm2y1(n)=sqrt(Ry1(n));
sigm2xy1(n)=abs(Rxy1(n))/(sqrt(Rx1(n))*sqrt(Ry1(n)));

Ry2(n)=pr^2*cos(ei2)^2*sin(bi)^2+ri2^2*pe^2*sin(bi)^2*sin(ei2)^2+ri2^2*pb^2*cos(bi)^2*cos(ei2)^2;
Rx2(n)=pr^2*cos(ei2)^2*cos(bi)^2+ri2^2*pe^2*cos(bi)^2*sin(ei2)^2+ri2^2*pb^2*sin(bi)^2*cos(ei2)^2;
Rz2(n)=pr^2*sin(ei2)^2+ri2^2*pe^2*cos(ei2)^2;
Rxy2(n)=0.5*sin(2*bi)*[cos(ei2)^2*pr^2-ri2^2*pb^2+ri2^2*pe^2*sin(ei2)^2];
Ryz2(n)=0.5*sin(2*ei2)*sin(bi)*[pr^2-ri2^2*pe^2];
Rxz2(n)=0.5*sin(2*ei2)*cos(bi)*[pr^2-ri2^2*pe^2];
sigm2x2(n)=sqrt(Rx2(n));
sigm2y2(n)=sqrt(Ry2(n));
sigm2xy2(n)=abs(Rxy2(n))/(sqrt(Rx2(n))*sqrt(Ry2(n)));
end
% polar(alm,rrm,'r');
% pause

polar(alm,sigm2x,'r');
hold on
polar(alm,sigm2x1,'r');

```

```
hold on
polar(a1m,sigm2x2,'r');
pause

% polar(a1m,sigm2y,'r');
% hold on
% polar(a1m,sigm2y1,'r');
% hold on
% polar(a1m,sigm2y2,'r');
% pause

% polar(a1m,sigm2xy,'r');
% pause
hold on
% end
pause
```