

Ю. А. Ніцук, О. М. Семчак, І. В. Шаріпова

ШЛЯХИ ЗМЕНШЕННЯ ПОХИБОК РОЗРАХУНКІВ ЕОМ АВТОНОМНОГО РУХОМОГО ОБ'ЄКТА ДЛЯ АЛГОРИТМІВ SLAM НАВІГАЦІЇ

У статті розглянуто питання щодо проведення оцінювання складності алгоритмів EKF-SLAM та побудови карти місцевості відповідно до опорних точок з погляду його алгоритмічно-програмної реалізації. Це дає можливість визначати шляхи подальшого розвитку та адаптації відомих математичних співвідношень алгоритмів EKF-SLAM та DP-SLAM для зменшення похибок розрахунків координат бортовими ЕОМ автономного рухомого об'єкта для реалізації алгоритмів.

Оцінювання стану автономного мобільного пристрою здійснюють шляхом фільтрації частинок. Генерується безліч гіпотез, що є кінцевим числом, які передбачають місце розташування робота. Кожен значущий елемент карти, тобто орієнтир, у кожній частинці може бути оцінений із використанням розширених фільтрів Калмана, обумовлених позицією частинок робота. А коефіцієнт ваги частинок розраховується для визначення ймовірності попадання певної частинки в остаточний набір, який буде окреслювати не лише реальне місце розташування автономного рухомого об'єкта на карті, але й положення всіх виявлених орієнтирів.

Запропонований у роботі шлях модифікації відомих математичних співвідношень фільтрів Калмана щодо їх адаптації до особливостей алгоритмічної та програмної реалізації в бортових ЕОМ забезпечує економію пам'яті бортової ЕОМ і зменшення необхідного обчислювального ресурсу. Зауважено, що алгоритми реалізації SLAM навігації, змінені запропонованим шляхом, використовують меншу кількість частинок, ніж методи, що ґрунтуються тільки на частотному фільтрі. Помилка початкового обчислення координат орієнтирів зводиться до мінімуму і не накопичується з часом у математичному сенсі.

Ключові слова: автономний рухомий об'єкт; Simultaneous Localization And Mapping; показники якості; прогнозування параметрів; експертна оцінка; короткострокове прогнозування; фільтр Калмана.

Постановка проблеми в загальному вигляді. Точне знання положення робота є фундаментальною проблемою мобільної робототехніки. Адже саме з розв'язання задачі локалізації починається процес навігації. Однак на точність локалізації впливають випадкові й систематичні помилки в показаннях датчиків. Тому завдання синтезу високоточних алгоритмів обробки інформації датчиків мобільного автономного рухомого об'єкта для визначення його поточного положення в просторі є актуальним науково-прикладним завданням [1, 2].

Наявні системи інерціальної та супутникової навігації з об'єктивних причин не в повній мірі забезпечують необхідну точність визначення координат положення в просторі автономного рухомого об'єкта (АРО). У той же час розвиток технологій локальної навігації на основі візуального зворотного зв'язку в комплексі із застосуванням систем технічного поля зору дозволяє з високою точністю провести оцінювання не лише координат АРО, але й навколишніх об'єктів, розташованих у полі його зору [2, 3].

© Ю. А. Ніцук, О. М. Семчак, І. В. Шаріпова, 2020

Аналіз останніх досліджень і публікацій. Визначення свого місця розташування робот може виконувати як на підставі апріорно наявної карти простору (місцевості), так і на підставі своїх спостережень. В ідеальному випадку є можливість завантажити роботу карту навколишнього простору, однак на практиці це малоймовірно, тому постає актуальне завдання: навчити робота будувати карту місцевості й одночасно визначати своє положення та траєкторію руху. Галузь знань, що описує методи вирішення даного завдання, отримала назву SLAM (Simultaneous Localization And Mapping) [4].

Задача SLAM поділяється на кілька підзадач: обчислення поточного становища робота на основі даних із датчиків або на основі GPS; знаходження нових точок інтересу; асоціація нових і старих даних; зберігання карти місцевості.

У ході реалізації відомих алгоритмів проведення арифметичних операцій унаслідок округлень результату обчислень, обумовлених поданням двійкових чисел у форматі з плаваючою комою, уже при складанні систем рівнянь та обчисленні поліномів 4-го ступеня починають накопичуватися помилки комп'ютерних розрахунків, які призводять до виродження матриць і унеможливають отримання результату обчислень [6–8]. Ці помилки пов'язані з обмеженою довжиною розрядної сітки ЕОМ (що зазвичай дорівнює 32 або 64 біти). Для вирішення зазначеної проблеми запропоновано метод подання чисел як масивів [7, 8].

Для вивчення особливостей алгоритмічної та програмної реалізації становить інтерес аналіз математичних співвідношень реалізації варіантів фільтра Калмана, висвітлений у [9]. Саме в згаданій роботі проведено вивчення можливостей використання різноманітних варіантів відомих фільтрів Калмана для оцінювання стану апріорі відомої динамічної системи. За своїм змістом це завдання схоже із завданням навігації АРО.

Формулювання завдання дослідження. Основними функціональними можливостями, які повинен мати АРО, є вирішення різноманітних прикладних задач зі збору інформації та виконання складних технологічних операцій у різноманітному середовищі, на відкритій місцевості або в складних міських умовах. Одне з ключових завдань пов'язане з необхідністю точного визначення координат точки свого положення та побудовою карти місцевості. Воно може бути вирішене критичним аналізом математичних співвідношень відомих алгоритмів EKF-SLAM навігації щодо їх алгоритмічної та програмної реалізації [14].

Виклад основного матеріалу. Алгоритм Extended Kalman Filter SLAM (EKF-SLAM) – метод, який ґрунтується на розширеному фільтрі Калмана для розв'язання задачі SLAM [15–17].

Фільтр Калмана – ефективний рекурсивний фільтр, що оцінює вектор стану динамічної системи, використовуючи низку неповних і зашумлених вимірювань. У теорії статистичного оцінювання розширений фільтр Калмана (EKF) – це нелінійна версія фільтра Калмана, що лінеаризується на позначці поточного середнього значення і коваріації. У разі добре визначених моделей переходу розширений фільтр Калмана фактично було визнано стандартом у теорії оцінювання нелінійних станів, навігаційних систем і GPS.

Використання фільтрів Калмана має на меті дооцінку вектора стану апріорі відомої динамічної системи, тобто для розрахунку поточного стану системи необхідно знати поточний

вимір, а також попередній стан самого фільтра [9, 18]. Для випадку навігації АРО це означає розв'язання задачі визначення місцезнаходження, що залежить від двох змінних:

X_k – оцінка вектора поточного розташування АРО у момент часу k ;

P_k – коваріаційна матриця помилок (міра точності оцінювання поточного розташування АРО) у момент часу k .

У [9, 12] зазначено, що розширений фільтр Калмана (ЕКФ) дуже схожий на простий фільтр Калмана за винятком того, що він може бути використаний у нелінійних процесах. ЕКФ – це один з найбільш поширених методів розв'язання задачі SLAM. Він дозволяє не лише уточнювати оцінку положення АРО на карті, але й положення всіх виявлених орієнтирів. Зазвичай процес оцінки стану системи в контексті SLAM розбивають на три етапи:

- 1) оновлення оцінки стану системи на основі одометричних даних;
- 2) оновлення оцінки стану системи на основі повторно виявлених орієнтирів;
- 3) додавання нових орієнтирів у систему.

У розширеному фільтрі Калмана моделі переходу та спостереження не повинні бути обов'язково лінійними функціями стану, натомість вони можуть бути нелінійними диференційованими функціями.

Функція еволюції процесу / системи в разі використання фільтра Калмана в завданнях навігації може використовуватися для обчислення передбачуваного стану з попередньої оцінки, тобто для визначення поточного та подальшого місця розташування на наступному кроці. Аналогічно функція вимірювань, що зв'язує істинний вектор стану і вектор проведених вимірювань, може використовуватися для обчислення передбачуваного вимірювання з передбаченого стану.

Проте ці функції не можуть застосовуватися до коваріації безпосередньо, натомість обчислюється матриця часткових похідних (матриця Якобі).

На кожному такті обчислень поточного місця розташування матриця Якобі обчислюється для поточних передбачених станів системи, тобто поточного місцезнаходження АРО. Ці матриці можуть використовуватися в рівняннях фільтра Калмана. Процес, власне, лінеаризує нелінійну функцію навколо поточної оцінки місця розташування АРО.

Оскільки фільтр Калмана є різновидом рекурсивних фільтрів, то для обчислення оцінки стану системи на поточний такт роботи йому необхідна оцінка стану на попередньому й вимірювання на поточному тактах.

У [9–11] зауважено, що попри всю свою привабливість ЕКФ має низку недоліків, серед яких можна виокремити обмеження на кількість орієнтирів у системі. У [9] вказано, що ситуація обумовлена тим, що матриця P має розмірність $m \times m$, де m – кількість виявлених орієнтирів. На кожному етапі оновлення матриці P має бути оновлений кожен її елемент, у зв'язку з чим істотно підвищується складність алгоритму визначення поточного місцезнаходження. У [9, 11] зроблено висновок, що ЕКФ найбільше підходить до ситуації, коли середовище має не дуже велику кількість (декілька сотень) легко помітних орієнтирів.

Кожна ітерація фільтра Калмана ділиться на дві фази: екстраполяція (прогноз) і корекція. Зазвичай вони чергуються: екстраполяція проводиться за результатами корекції до наступного спостереження, а корекція – разом із доступними на наступному кроці спостереженнями [4, 5].

Під час екстраполяції фільтр отримує попередню оцінку стану системи на поточний крок за підсумковою оцінкою стану з попереднього кроку. Цю попередню оцінку також називають апріорною оцінкою стану, оскільки для її отримання не використовуються спостереження відповідного кроку.

Математичний опис оцінки стану динамічної системи на основі розширеного фільтра Калмана детально розглянуто в [9, 11].

Стан АРО в довільний момент часу описується за допомогою оцінки вектора його поточного місця розташування X_p і коваріаційної матриці P_p , що мають такий вигляд:

$$X_p = [x_p y_p \varphi_p]^T; \tag{1}$$

$$P_p = \begin{bmatrix} \sigma_{x_p x_p}^2 & \sigma_{x_p y_p}^2 & \sigma_{x_p \varphi_p}^2 \\ \sigma_{x_p y_p}^2 & \sigma_{y_p y_p}^2 & \sigma_{y_p \varphi_p}^2 \\ \sigma_{x_p \varphi_p}^2 & \sigma_{y_p \varphi_p}^2 & \sigma_{\varphi_p \varphi_p}^2 \end{bmatrix}, \tag{2}$$

де x_p – оцінка координати АРО за віссю абсцис;

y_p – оцінка координати АРО за віссю ординат;

φ_p – оцінка кутової орієнтації АРО. Для відомих рухомих об’єктів аналогом цих значень є значення напрямку поздовжньої осі машини.

Коефіцієнти коваріаційної матриці P_p відображають міру залежності числового значення координат АРО один від одного. Діагональні елементи описують середньоквадратичну похибку оцінки числового значення координати за відповідною віссю [4, 9, 11]. На початку роботи системи навігації для цих змінних у програмній реалізації мають бути присвоєні значення за замовчуванням, що характеризують ситуацію невідомого положення АРО відносно орієнтирів положення. Якою б точною не була інформаційно-вимірювальна система АРО, у [9] запропоновано спеціально задати значення помилки оцінки початкового положення, відмінне від нуля. Така пропозиція обумовлена математичним припущенням того, що за реалізації фільтра на бортовій ЕОМ нульові значення на діагональних елементах можуть призвести до помилки в ході обчислення зворотної матриці [9, 19].

Координати виявлених орієнтирів за умови, що вони є нерухомими об’єктами, описуються у вигляді вектора оцінки координат орієнтирів X_0 , і коваріаційної матриці P_0 , що мають такий вигляд:

$$X_0 = [x y_1 \dots x_n y_n]^T; \tag{3}$$

$$P_0 = \begin{bmatrix} \sigma_{(x_1 x_1)}^2 & \sigma_{(x_1 y_1)}^2 & \dots & \sigma_{(x_1 x_n)}^2 & \sigma_{(x_1 y_n)}^2 \\ \sigma_{(x_1 y_1)}^2 & \sigma_{(y_1 y_1)}^2 & \dots & \sigma_{(y_1 x_n)}^2 & \sigma_{(y_1 y_n)}^2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \sigma_{(x_1 x_n)}^2 & \sigma_{(y_1 x_n)}^2 & \dots & \sigma_{(x_p x_p)}^2 & \sigma_{(x_n y_n)}^2 \\ \sigma_{(x_1 y_n)}^2 & \sigma_{(y_1 y_n)}^2 & \dots & \sigma_{(x_n y_n)}^2 & \sigma_{(y_n y_n)}^2 \end{bmatrix}, \tag{4}$$

де n – кількість орієнтирів, виявлених АРО;

x_i – оцінка координати i -го орієнтира за віссю абсцис;

y_i – оцінка координати i -го орієнтира за віссю ординат.

Матриця P_0 за своєю суттю відображає міру залежності оцінки числового значення координат місцевих орієнтирів за віссю абсцис та віссю ординат один від одного.

Поточне місцезнаходження АРО визначається вектором X , що відображає оцінку координат АРО і орієнтирів на поточному кроці ітераційних розрахунків, а також коваріаційною матрицею P , що мають такий вигляд:

$$X = \begin{bmatrix} X_p^T & X_0^T \end{bmatrix}^T ; \quad (5)$$

$$P = \begin{bmatrix} P_p & P_{p0} \\ P_{p0}^T & P_0 \end{bmatrix}, \quad (6)$$

де P_{p0} – коваріаційна матриця розмірністю $3 \times n$, що відображає залежність між оцінкою координат АРО і оцінками положення орієнтирів.

На початку руху АРО в програмній реалізації фільтра передбачаються такі числові значення координат місця розташування за замовчуванням:

$$X = X_p = 0, P = P_p.$$

У коваріаційній матриці P_p значення діагональних елементів встановлюються відмінними від нуля [9, 19], це реалізує помилку оцінки початкових координат старту руху АРО [9].

Розвитком фільтра Калмана для випадку навігації АРО є алгоритм Distributed Particle SLAM (DP-SLAM) [16, 17]. Це один з підходів до вирішення завдань SLAM, який використовує числові значення змінних показників далекоміра і фільтр частинок для зберігання гіпотез про поточне значення координат АРО і конфігурації навколишнього середовища в сенсі значень координат орієнтирів.

Фільтр частинок (послідовний метод Монте-Карло) – рекурсивний алгоритм, що дозволяє на основі поточної карти обчислити найбільш імовірне положення робота за допомогою деякого набору (хмари) частинок, що займають простір станів.

Основна ідея методу фільтрації частинок полягає в поданні апостеріорного розподілу положення робота за допомогою кінцевого числа семплів, замість параметричного вигляду (наприклад, експоненційної функції в разі нормального розподілу). Таке уявлення є наближенням, але завдяки непараметричному вигляду дозволяє описати набагато складніші розподіли.

Під час руху робота відбувається збір даних з далекоміра та одометричних датчиків, а також побудова миттєвої карти простору з точок з імовірнісною характеристикою ваги (зважених частинок) [16, 17]. Чим більша вага частинки, тим вона значущіша. У міру переміщення є шанс зустріти ту ж частинку з деякою помилкою. З визначенням належності нової частинки до вже наявної її вага збільшується. Отже, повна карта складається з частинок з найбільшою вагою.

Модель простору станів складається з марковського процесу та масиву результатів процесу вимірів (поточних координат). Фільтр частинок зберігає зважену нормалізовану множину вибірки станів (поточних значень координат), нормалізує ваги для нової множини станів.

Хмара частинок, тобто множина координат, розрахованих відносно різних орієнтирів, характеризує невизначеність поточного місця розташування АРО. Чим більша кількість частинок-орієнтирів, тим більша ймовірність коректного визначення поточного місця розташування АРО. Одна частинка у фільтрі частинок містить положення і кутову орієнтацію робота. Ймовірність частинок розраховується на основі різниці реальних свідчень далекоміра і показань, які повинні були б бути в даній частинці [4, 12].

DP-SLAM полягає в тому, що у фільтрі підтримується безліч гіпотез про поточний стан робота [16, 17]. У початковий момент генерується випадковий набір гіпотез. У ході роботи алгоритму деякі з них будуть відсіюватися через невідповідність вторинним ознакам системи, які можуть бути виміряні більш точно. У результаті після завершення циклу сканування вибирається найбільш вірогідна з тих гіпотез, що залишилися у фільтрі, вона і буде шуканою картою, на якій до того ж буде відзначена траєкторія руху виконавця.

Карта зберігається у вигляді зв'язного графа ієрархічної системи [15], що називають деревом, у вузлах якого знаходяться частинки. Завдання зберігання частинок карти спрощується введенням так званих частинок «батьків» і «нащадків». Кожен вузол може мати тільки одного з «батьків» і кілька «нащадків», завдяки чому в ході асоціації даних не потрібно проходити граф повністю. Пошук проводиться лише за «батьківськими» частинками до «нащадків».

DP-mapping зберігає тільки одну копію сітки зайнятості (карти). Ідея реалізації дерева успадкування частинок досить проста. Його коренем є початкова частинка, з якої походять всі інші. Кожна частинка містить покажчик на свого «батька», має унікальний ідентифікатор (ID) і зберігає список осередків карти, які вона оновила.

На рис. 1 концептуально показано «родовід» частинок й оновлення карт, листям – «дерева предків», що вказують на карти. Кожна червона крапка в дереві родоходу є вибірковою позицією робота, а чорні лінії навколо червоної крапки – нові спостереження, пов'язані з поточною позицією робота. Сіра лінія показує частину карти, успадковану від попередньої частинки.

Варто звернути увагу на те, що всі карти в цьому прикладі узгоджуються з початковим виглядом. Дві ліві карти узгоджуються зі спостереженнями, зробленими лівим ребром кореня, у той час як дві праві карти – зі спостереженнями, зробленими правим ребром кореня. Зберігання та повторне копіювання цих розділів карти щоразу, коли частинка повторно відбирається, марна трата пам'яті й часу. Замість цього використовується єдина сітка зайнятості, яка зберігає дерево спостережень на кожному квадраті.

На рис 2 показано, як кожна частинка вставляє свої спостереження в глобальну сітку. Вони зберігаються у вигляді збалансованого дерева, індексованого за унікальним ідентифікатором, присвоєним кожній частинці.

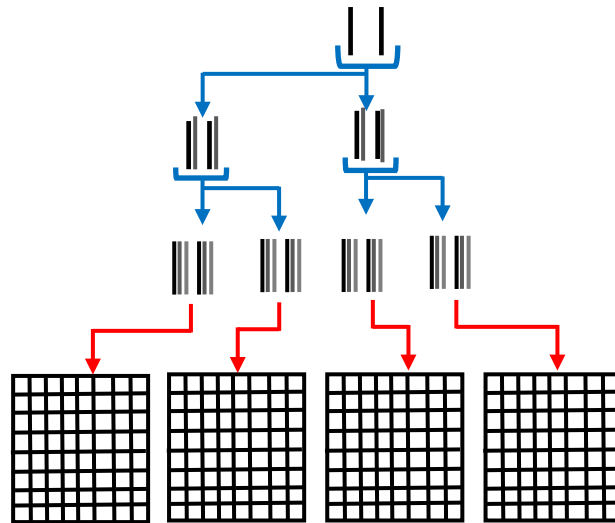


Рис. 1. Деревовидна система зберігання «гіпотетичних» карт

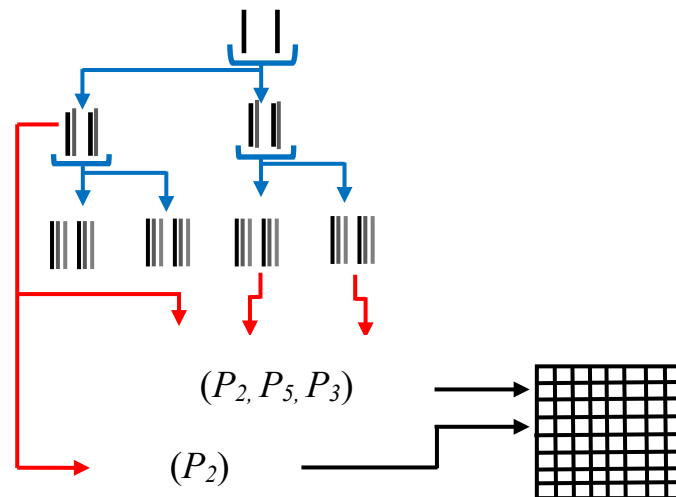


Рис. 2. Побудова карти за допомогою частинок

Звідси випливає, що перевагою алгоритму є низька складність:

$$O(ADlgP),$$

де A – кількість звернень частинки до карти в ході локалізації;

D – глибина вкладеності графа;

P – кількість частинок.

DP-SLAM має можливість швидко працювати з великими обсягами даних, але за умови $ADlgP > M$ знадобляться великі обчислювальні дані (де M – розмір карти).

Недоліком алгоритму є те, що для отримання коректного результату вимірювань необхідна велика кількість частинок.

На наступному кроці кожен орієнтир на карті оцінюється за допомогою розширеного фільтра Калмана. Усі EKF орієнтирів обумовлені шляхами робота, причому кожна частинка у фільтрі має власний набір EKF.

Якщо орієнтир виявлений уперше, то необхідно реалізувати такі кроки в алгоритмі програмної реалізації:

виконати ініціалізацію позиції орієнтира на основі вимірів і поточної позиції частинки;

знайти матрицю Якобі;

сформуванати ЕКФ для даного орієнтира;

в іншому разі:

отримати очікувані вимірювання і матрицю Якобі;

обчислити коваріацію вимірювань;

розрахувати посилення Калмана;

обчислити помилку між фактичним і прогнозованим наглядом;

оновити середнє значення й коваріацію ЕКФ.

У результаті виконання даного кроку алгоритму кожна частинка матиме N орієнтирів, поданих за допомогою ЕКФ.

Етап повторної вибірки є важливим аспектом, який чинить основний вплив на продуктивність фільтра частинок. Під час повторної вибірки частинки з низькою вагою зазвичай замінюються зразками з вищою вагою. З одного боку, необхідна повторна вибірка, оскільки використовується тільки кінцева кількість частинок. З іншого – етап повторної вибірки може видалити хороші зразки з набору, зумовлюючи виснаження частинок. Відповідно, важливо знайти критерій, коли виконувати етап повторної вибірки. У роботі [18] введено так зване «ефективне число» частинок, щоб оцінити, наскільки добре поточний набір частинок описує справжній апостеріор.

У запропонованому алгоритмі передбачено, що щоразу, коли вага частинки падає нижче заданого значення, то буде проводитися повторна вибірка. Цей підхід значно знижує ризик заміни хороших частинок, оскільки кількість операцій повторної вибірки скорочується, бо вони виконуються тільки за необхідності.

Повторна вибірка виконується відповідно до так званої «вибірки з низькою дисперсією», розглянутої в роботах [16–18].

Висновки. Отже, під час модифікації відомих математичних співвідношень фільтрів Калмана щодо їх адаптації до особливостей алгоритмічної та програмної реалізації в бортових ЕОМ забезпечується економія пам'яті апаратури та зменшення необхідного обчислювального ресурсу.

Найбільш доцільним шляхом зменшення похибок розрахунків координат бортовими ЕОМ автономного рухомого об'єкта для реалізації алгоритмів ЕКФ-SLAM навігації є розроблення алгоритму, що поєднує позитивні властивості розглянутих алгоритмів SLAM навігації.

Удосконалені алгоритми ЕКФ-SLAM навігації використовують меншу кількість частинок, ніж методи, що ґрунтуються тільки на частотному фільтрі. А помилка початкового обчислення координат орієнтирів зводиться до мінімуму і не накопичується з часом у математичному сенсі. З використанням методу подання чисел як масивів зводиться до мінімуму помилка початкового обчислення координат орієнтирів у сенсі подання чисел на рівні мікроархітектури програмних кодів базових арифметичних операцій.

Оцінювання поточного значення координат АРО в разі створення вдосконаленого алгоритму фільтрації досягається шляхом фільтрації частинок. Кожен елемент карти

в кожній частинці може бути оцінений з використанням розширених фільтрів Калмана, обумовлених позицією частинок робота. Коефіцієнт ваги частинок розраховується для визначення ймовірності попадання певної частинки в остаточний набір, який становитиме реальне місце розташування.

Запропонований шлях зменшення похибок розрахунків координат бортовими ЕОМ АРО для реалізації алгоритмів EKF-SLAM навігації використовує меншу кількість частинок зі зведенням до мінімуму ймовірності помилки поточного обчислення координат, що не накопичується з часом.

СПИСОК ЛІТЕРАТУРИ

1. Захаров А. А., Тужилкин А. Ю., Веденин А. С. Алгоритм определения положения и ориентации трехмерных объектов по видеоизображениям на основе вероятностного подхода // *Фундаментальные исследования*. 2014. № 11-8. С. 1683–1687.
2. Menache A. *Understanding motion capture for computer animation*. The Morgan Kaufmann Series In Computer Graphics. 2011. 254 p.
3. Tobon R. *The Mocap Book: A Practical Guide to the Art of Motion Capture*. Forisforce. 2010. 258 p.
4. SLAM – что это такое. URL: <https://icleborobot.by/slam-что-это-в-навигации.html> (дата обращения: 15.12.2018).
5. Nguyen V., Harati A., Siegwart R. Lightweight SLAM algorithm using orthogonal planes for indoor mobile robotics // *Intelligent Robots and Systems*. 2007. P. 658–663. DOI: <http://doi.org/10.1109/iros.2007.4399512>
6. Левченко А. А., Войтенков Р. М. Анализ предельных точностей вычислений в информационных системах с представлением чисел с плавающей запятой // *Збірка тез доп. 3-го наук.-техн. семінару “Перспективні шляхи розвитку інформаційних систем прицілювання та самонаведення високоточного озброєння РВіА”*. Львів : АСВ, 2012. С. 119.
7. Levchenko A. Arithmetic operation for binari numbers repressetated as arrays // *Modern engineering and innovative technologies / International periodic scientific journal*. Karlsruhe, Germany. 2019. № 9, Part 1. P. 51–59.
8. Левченко А. О., Войтенков Р. М. Метод подання чисел для програмних засобів гарантоздатних інформаційних технологій систем підтримки прийняття рішень для керування станом ОБТ // *Збірник тез доповідей 19-ї наук.-практ. конф. (“Проблеми створення, розвитку та застосування інформаційних систем спеціального призначення”*, 19 квітн. 2012, м. Житомир). Житомир : ЖВІ НАУ, 2012. С. 142–143.
9. Кучерский Р. В., Манько С. В. Алгоритмы локальной навигации и картографии для бортовой системы управления автономного мобильного робота // *Известия ЮФУ. Технические Науки*. 2012. № 3 (128). С. 13–22.
10. Semchak O., Levchenko A. Disadvantages of computer implementation of SLAM-methods of local navigation autonovus mobile objects // *SWorld journal : International periodic scientific journal*. Sofia, Bulgaria. 2019. № 2, Part 2. P. 108–115.
11. Aulinas J. The SLAM Problem: A Survey // *Proceedings of the 2008 Conference on Artificial Intelligence Research & Development*. 2008. P. 363–71. URL: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.163.6439> (last accessed: 15.07.2020).

12. Любкевич К. О., Гунченко Ю. О. Локалізація мобільного робота на місцевості. // Збірник матеріалів XIV Всеукр. конф. студентів і молодих науковців (“Інформатика, інформаційні системи та технології”). Одеса, 2017. С. 188.
13. Methods and technologies of monitoring of the position of a mobile object in space: Proceedings / Nechyporenko O. V. et. al. // *Kompiuterne modeliuвання ta optymizatsiia skladnykh system (KMOSS-2018)*. Dnipro : Balans-klub, 2018. P. 193–195.
14. Levchenko A. Features of implementation of information technology for provision of numerical value of parameters // *Modern engineering and innovative technologies // International periodic scientific journal*. Karlsruhe, Germany. 2019. № 10, Part 1. P. 36–42.
15. Robot Mapping And Ekf Slam. URL: <https://slideplayer.com/slide/5983202/> (last accessed: 10.01.2018).
16. Eliazar Austin I., Ronald Parr. DP-SLAM 2.0 // *ICRA '04: IEEE International Conference on Robotics and Automation*. 2004. Vol. 2. P. 1314–1320.
17. Austin Eliazarand, Ronald Parr. DP-SLAM. URL: <https://users.cs.duke.edu/~parr/dpslam/> – (last accessed: 10.02.2018).
18. Doucet A., Freitas N., Murphy K. P., Russell S. J. Rao-Blackwellised particle filtering for dynamic Bayesian net works // *Proc. of the 16th Conf.48 on Uncertainty in Artificial Intelligence*. San-Francisco: Morgan Kaufmann Publisers Inc, 2000. P. 176–183.
19. Левченко А. О., Войтенков Р. М. Витоки втрати працездатності систем діагностики ОБТ другого роду з представленням чисел з плаваючою комою // *Сб. науч. труд. Sword*. 2014. Вип. № 4 (37), Том 5. С. 27–35.

Подано 06.11.2020

REFERENCES

1. Zakharov, A. A., Tuzhylykyn, A. Yu., & Vedenyn, A. S. (2014). Alhorytm opredeleniya polozheniya y orientatsyy trekhmernykh ob"ektov po vydeoyzobrazheniyam na osnove veroiatnostnoho podkhoda [Algorithm for determining the position and orientation of three-dimensional objects from video images based on a probabilistic approach]. *Fundamentalnye yssledovaniya [Basic research]*, 11-8, 1683–1687 [in Russian].
2. Menache, A. (2011). *Understanding motion capture for computer animation*. The Morgan Kaufmann Series In Computer Graphics.
3. Tobon, R. (2010). *The Mocap Book: A Practical Guide to the Art of Motion Capture*. Forisforce.
4. SLAM – что это такое. (n.d.). Retrieved from <https://icleborobot.by/slam-что-это-в-навигации.html> [in Russian].
5. Nguyen V., Harati, A., & Siegwart, R. (2007). Lightweight SLAM algorithm using orthogonal planes for indoor mobile robotics. *Intelligent Robots and Systems*, 658–663. <http://doi.org/10.1109/iros.2007.4399512>
6. Levchenko, A. A., & Voitenkov, R. M. (2012). Analiz predelnykh tochnostei vychysleniy v ynfornatsyonnykh systemakh s predstavlenyem chysel s plavaiushchei zapiatoi [Analysis of the limiting accuracy of calculations in information systems with the representation of floating point numbers]. In *Zbirka tez dop. 3-ho nauk.-tekhn. seminaru [Collection of abstracts of papers 3rd scientific-technical seminar]*. (“Perspektyvni shliakhy rozvytku informatsiinykh system

prytsiliuvannia ta samonavedennia vysokotochnoho ozbroiennia RViA” ["Prospective ways of development of information systems for aiming and homing high-precision weapons"]). (p. 119). Lviv: NAA [in Russian].

7. Levchenko, A. (2019). Arithmetic operation for binari numbers representated as arrays. *Modern engineering and innovative technologies. International periodic scientific journal*, 9 (1), 51–59. Karlsruhe, Germany.

8. Levchenko, A. O., & Voitenkov, R. M. (2012). Metod podannia chysel dlia prohramnykh zasobiv harantozdatnykh informatsiinykh tekhnolohii system pidtrymky pryiniattia rishen dlia keruvannia stanom OVT [The method of representing numbers for software tools of guaranteeing information technologies of decision support systems for the management of weapons and military equipment]. In *Zbirnyk tez dopovidei 19-i nauk.-prakt. konf. [Collection of abstracts of the 19th scientific-practical conference]* ("Problemy stvorennia, rozvytku ta zastosuvannia informatsiinykh system spetsialnoho pryznachennia" ["Problems of creation, development and application of special purpose information systems"]). Zhytomyr, April 19, 2012. (pp. 142–143). Zhytomyr : ZhMI NAU [in Ukrainian].

9. Kucherskyi, R. V., & Manko, S. V. (2012). Alhorytmy lokalnoi navyhatsyy y kartohrafyy dlia bortovoi systemy upravleniia avtonomnoho mobylnoho robota [Local navigation and mapping algorithms for the on-board control system of an autonomous mobile robot]. *Yzvestyia YuFU. Tekhnicheskyye Nauky [Izvestia UFU. Technical science]*, 3 (128), 13–22 [in Russian].

10. Semchak, O., & Levchenko, A. (2019). Disadvantages of computer implementation of SLAM-methods of local navigation autonomus mobile objects. *SWorld journal. International periodic scientific journal*, 2 (2), 108–115. Sofia, Bulgaria.

11. Aulinas J. (2008). The SLAM Problem: A Survey. In *Proceedings of the 2008 Conference on Artificial Intelligence Research & Development*. (p. 363–371). Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.163.6439>

12. Liubkevych, K. O., & Hunchenko, Yu. O. (2017). Lokalizatsiia mobilnoho robota na mistsevosti [Localization of mobile robot in the field]. In *Zbirnyk materialiv XIV Vseukr. konf. studentiv i molodykh naukovtsiv [Collection of materials XIV All-Ukrainian. conf. students and young scientists]* ("Informatyka, informatsiini systemy ta tekhnolohii" ["Informatics, information systems and technologies"]). (p. 188). Odesa [in Ukrainian].

13. Nechyporenko, O. V. et. al. (2018). Methods and technologies of monitoring of the position of a mobile object in space. *Kompiuterne modeliuvannia ta optymizatsiia skladnykh system (KMOSS-2018) [Computer modeling and optimization of complex systems (CMOCS-2018)]*. (pp. 193–195). Dnipro: Balans-klub.

14. Levchenko, A. (2019). Features of implementation of information technology for provision of numerical value of parameters. *Modern engineering and innovative technologies. International periodic scientific journal*, 10 (1), 36–42. Karlsruhe, Germany.

15. Robot Mapping And Ekf Slam. (n.d.). Retrieved from <https://slideplayer.com/slide/5983202/>

16. Eliazar Austin, I., & Ronald Parr. (2004). DP-SLAM 2.0. *ICRA '04: IEEE International Conference on Robotics and Automation, Vol. 2*, 1314–1320.

17. Austin Eliazarand, Ronald Parr. (n.d.). DP-SLAM. Retrieved from <https://users.cs.duke.edu/~parr/dpslam>

18. Doucet, A., Freitas, N., Murphy, K. P., & Russell, S. J. (2000). Rao-Blackwellised particle filtering for dynamic Bayesian net works. In *Proc. of the 16th Conf.48 on Uncertainty in Artificial Intelligence*. (pp. 176–183). San-Francisco: Morgan Kaufmann Publisers Inc.

19. Levchenko, A. O., & Voitenkov, R. M. (2014). Vytoky vtraty pratsezdatnosti system diahnostryky OVT druhoho rodu z predstavleniam chysel z plavaiuchoiu komoiu [Sources of disability of weapons diagnostics systems and military equipment of the second kind with the representation of floating point numbers]. *Sb. nauch. trud. Sword [Collection of scientific papers Sword]*, Vol. 4 (37), Iss. 5, 27–35 [in Ukrainian].

Yu. A. Nitsuk, O. M. Semchak, I. V. Sharipova

WAYS OF DIMINISHING OF ERRORS OF CALCULATIONS OF COMPUTER OF AUTONOMOUS MOBILE OBJECT ARE FOR ALGORITHMS OF SLAM OF NAVIGATION

A question is in-process considered, in relation to the lead through of estimation of complication of algorithms of EKF-SLAM and construction of map of locality in accordance with supporting points, from point of his algorithmically programmatic realization. It enables to determine the ways of subsequent development and adaptation of the known mathematical correlations of algorithms of EKF-SLAM and DP-SLAM for diminishing of errors of calculations of co-ordinates airborne COMPUTERS of autonomous mobile object for realization of algorithms.

The estimation of the state of off-line mobile unit is arrived at by filtration of particles. The great number of hypotheses which are an eventual number is generated, which show by itself the predictable place of location of robot. Every meaningful element of map, that orienteer, in every particle can be appraised with the use of the extended filters of Kalmana, particles of robot conditioned position.

And the coefficient of weight of particles settles accounts for determination of probability of hit of certain part in a final set, which will present not only the real place of location of autonomous mobile object on a map but also position of found out all orienteers.

The way of modification of the known mathematical correlations of filters of Kalmana offered in-process from point of their adaptation to the features of algorithmic and programmatic realization in airborne COMPUTERS provides economy of memory of airborne COMPUTER and diminishing of necessary calculable resource

It is noticed that the algorithms of realization of SLAM of navigation are changed the offered way use less of particles, than methods, based only on a frequency filter. The error of initial calculation of co-ordinates of orienteer is taken to the minimum and does not accumulate in course of time in mathematical sense.

Keywords: *autonomous mobile object; Simultaneous Localization And Mapping; indexes of quality; prognostication of parameters; expert estimation; short-term prognostication; filter of Kalmana.*