

Модель та алгоритми гранулярної фільтрації у задачі глобальної локалізації мобільного робота

Пантєєв Р. Л.^a, Бідюк П. І.^b

^a Київський інститут бізнесу та технологій, Україна

^b Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Україна

Анотація

Методи оцінювання параметрів і станів динамічних систем – актуальна задача, результати розв'язання якої знаходять своє застосування у різних галузях діяльності, включаючи дослідження процесів у технічних системах, космологічних та фізичних дослідженнях, медичних діагностичних системах, економіці, фінансах, біотехнологіях, екології та інших. Незважаючи на значні досягнення у цьому напрямі, дослідники багатьох країн світу продовжують пошуки нових методів оцінювання параметрів і станів досліджуваних об'єктів та удосконалення існуючих. Прикладом таких методів є цифрова та оптимальна фільтрація, які знайшли широке застосування у технічних системах ще у середині минулого століття, зокрема, у обробці фінансово-економічних даних, фізичних експериментах та інших інформаційних технологіях самого різного призначення.

Розглядається модель та алгоритми гранулярної фільтрації на практичному прикладі – варіанті задачі глобальної локалізації мобільного робота (global localization for mobile robots) або задачі про викраденого робота (hijacked robot problem). В загальному варіанті вона полягає у визначенні положення робота за даними з сенсора. Ця задача була в цілому розв'язана рядом імовірнісних методів в кінці 90-х-початку 2000-х років. Задача є важливою і знаходить застосування у мобільній робототехніці та промисловості. Схожими за суттю є задачі позиціонування підводних човнів, літальних апаратів, автомобілів тощо.

Також розглядається задача позиціонування робота. Нехай у темному лабіринті увімкнувся робот. Він має карту лабіринту та компас. У лабіринті в деяких точках встановлені позначені на карті станції, які можуть приймати і відбивати сигнал. Робот не знає, в якому місці лабіринту він знаходиться, але він може в кожний момент часу відправляти сигнал і з деякою похибкою дізнатись відстань до найближчої до нього станції. Робот починає блукати лабіринтом, роблячи кожний крок у новому випадково обраному напрямку, але його компас також дає деяку несистематичну похибку. На кожному кроці робот визначає відстань до найближчої станції. Мета – з'ясувати координати робота у лабіринті в системі відліку, введених на карті.

Ключові слова: оцінювання станів, динамічна система, цифрова фільтрація, оптимальна фільтрація, несистематична похибка, позиціонування робота.

Granular filtration model and algorithms in the mobile robot global localization problem

Panteyev R. L.^{a1}, Bidiuk P. I.^b

^a Kiev Institute of Business and Technology, Ukraine

^b National technical university of Ukraine “Igor Sikorsky Kyiv polytechnic institute”, Ukraine

Abstract

Methods for estimating the parameters and states of dynamic systems are an urgent task, the results of which are used in various fields, including processes in technical systems, cosmological and physical research, medical diagnostic systems, economics, finance, biotechnology, ecology and others. Despite significant scientific and practical advances in this area, researchers in many countries around the world continue to search for new methods for estimating the parameters and condition of the studied objects and improving existing ones. An example of such methods is digital and optimal filtering, which have been widely used in technical systems since the middle of the last century, in particular, in the processing of financial and economic data, physical experiments and other information technologies for various purposes. The model and algorithms of granular filtration are considered on a practical example - a variant of the problem of global localization of a mobile robot (global localization for mobile robots) or the problem of a hijacked robot problem. In the General embodiment, it is to determine the position of the robot according to the data from the sensor. This problem was generally solved by a number of probabilistic methods in the late 1990s and early 2000s. The task is important and finds application in mobile robotics and industry. The tasks of positioning submarines, aircraft, cars, etc. are essentially similar. The problem of robot positioning is also considered. Let the robot turn on in the dark maze. It has a maze map and a compass. In the labyrinth at some points there are stations marked on the map, which can receive and reflect the signal. The robot does not know where the maze is, but it can send a signal at any time and with some error know the distance to the nearest station. The robot begins to wander the maze, taking each step in a new randomly chosen direction, but his compass also gives some unsystematic error. At each step, the robot determines the distance to the nearest station. The goal is to find out the coordinates of the robot in the maze in the frame of reference entered on the map.

Keywords: state estimation, dynamic system, digital filtration, optimal filtration, non-systematic error, robot positioning.

¹ Corresponding author.

E-mail address: panteev.r@kibit.edu.ua

Вступ

Розглянуто задачу позиціонування робота. Нехай у темному лабіринті увімкнувся робот. Він має карту лабіринту та компас. У лабіринті в деяких точках встановлені позначені на карті станції, які можуть приймати і відбивати сигнал. Робот не знає, в якому місці лабіринту він знаходиться, але він може в кожний момент часу відправляти сигнал і з деякою похибкою дізнаватись відстань до найближчої до нього станції. Робот починає блукати лабіринтом, роблячи кожний крок у новому випадково обраному напрямку, але його компас також дає деяку несистематичну похибку. На кожному кроці робот визначає відстань до найближчої станції. Кінцева мета – з'ясувати координати робота у лабіринті в системі відліку, введеної на карті [1,2].

Наукова новизна статті полягає у тому, що в ній було розглянуто сучасні підходи до розв'язання задач лінійної та нелінійної фільтрації статистичних та експериментальних даних, що забезпечують отримання оптимальних оцінок станів досліджуваних об'єктів.

У створеній програмі для симуляції поставленої задачі глобальної локалізації мобільного робота відбувається гранулярна фільтрація з метою визначення стану робота обраним методом. Доступними є фільтр вибірки за значимістю з відсівом, допоміжний фільтр вибірки за значимістю з відсівом та регуляризований гранулярний фільтр. Базовий алгоритм послідовної вибірки за значимістю в чистому вигляді не використовується через проблему виродження вагових коефіцієнтів.

У статті також було проведено докладний аналіз методу гранулярної байєсівської фільтрації та подано приклад застосування методу гранулярної фільтрації і створення системи підтримки прийняття рішень для розв'язання задач вищезазначеного типу.

Мета статті - розв'язання наступних задач: розглянути деякі відомі підходи до розв'язання задач лінійної та нелінійної фільтрації статистичних / експериментальних даних, які забезпечують обчислення оптимальних оцінок станів досліджуваних об'єктів; виконати докладний аналіз методу гранулярної байєсівської фільтрації [3]; подати приклад застосування методу гранулярної фільтрації і створення системи підтримки прийняття рішень для розв'язання задач вказаного вище типу.

Виклад основного матеріалу

Переведемо вищезазначену задачу в модель у просторі станів. Невідомим станом робота, який необхідно оцінити за допомогою гранулярної фільтрації, є пара його координат на k -у кроці $(x(k), y(k))$. Вимірюваною змінною $z(k)$ є

відстань до найближчої станції на поточному кроці. Маємо таку систему рівнянь [4,5]:

$$\begin{pmatrix} x(k) \\ y(k) \end{pmatrix} = \begin{pmatrix} x(k-1) \\ y(k-1) \end{pmatrix} + L * \begin{pmatrix} \cos(\theta(k-1)) \\ \sin(\theta(k-1)) \end{pmatrix} \quad (1)$$

$$z(k) = \left\| \begin{pmatrix} x(k) \\ y(k) \end{pmatrix} - \begin{pmatrix} x^* \\ y^* \end{pmatrix} \right\| + v(k) \quad (2)$$

де L – довжина кроку робота;

$\theta(k) \sim U(\theta^0(k) - \Delta/2, \theta^0(k) + \Delta/2)$ – кут повороту робота в радіанах на k -му кроці. Він є рівномірно розподіленим, враховуючи похибку компаса, в інтервалі з центром $\theta^0(k)$ – випадково вибраним на цей крок значенням кута повороту, виходячи з показів компаса $\theta(k) \in [0; 2\pi]$;

(x^*, y^*) – координата найближчої до робота станції;

$v(k) \sim N(0, \sigma^2)$ – похибка вимірювання відстані до найближчої станції.

Застосування гранулярних фільтрів. Для застосування гранулярних фільтрів нам знадобляться розподіли

$$p(x(k), y(k) | x(k-1), y(k-1)),$$

$$p(z(k) | x(k), y(k)).$$

Очевидно, що

$$p(z(k) | x(k), y(k)) = N\left(\left\| \begin{pmatrix} x(k) \\ y(k) \end{pmatrix} - \begin{pmatrix} x^* \\ y^* \end{pmatrix} \right\|, \sigma^2\right) \quad (3)$$

Розглянемо розподіл

$$p(x(k), y(k) | x(k-1), y(k-1)).$$

З геометричних міркувань, якщо кут повороту робота розподілений рівномірно на відріжку [6], то координата $(x(k), y(k))$ після кроку буде належати відповідній дузі з центром у точці $(x(k-1), y(k-1))$ та радіусом L :

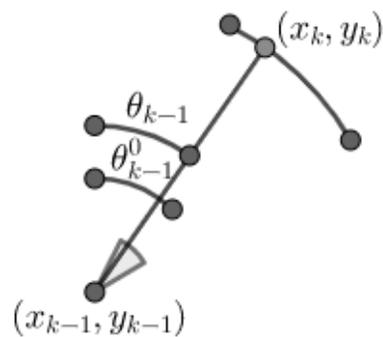


Рис. 1. Кінематичні співвідношення робота.

При цьому $(x(k), y(k))$ має рівномірний розподіл на дузі:

$$p(x(k), y(k) | x(k-1), y(k-1)) = \begin{cases} \frac{1}{2\pi L}, & (x(k), y(k)) \in \text{дузі} \\ 0, & \text{інакше} \end{cases}$$

Генерування стану $(x(k), y(k))$ з розподілу

$$p(x(k), y(k) | x(k-1), y(k-1))$$

можна виконувати безпосередньо, а саме згенерувати реалізацію випадкової величини $\theta(k-1)$, яка має рівномірний розподіл

$$U(\theta^0(k-1) - \Delta/2, \theta^0(k-1) + \Delta/2),$$

та обчислити $(x(k), y(k))$ з рівняння (1).

Оскільки попередньої інформації по положення робота немає, початкова координата $(x(1), y(1))$ є рівномірно розподіленою по вільній від перешкод області лабіринту.

У цій задачі гранули $\{x^i(k)\}_{i=1}^{N_S}$ по суті відповідають гіпотезам про координату робота. Задача полягає в знаходженні на кожному кроці розподілу ймовірностей над всіма можливими положеннями робота на карті.

У наданій програмі для симуляції поставленої задачі відбувається гранулярна фільтрація з метою визначення стану робота обраним методом. Доступними є фільтр вибірки за значимістю з відсівом (SIR), допоміжний фільтр вибірки за значимістю з відсівом (APF) та регуляризований гранулярний фільтр (RPF), описані вище. Базовий алгоритм послідовної вибірки за значимістю (SIS) в чистому вигляді не використовується через проблему виродження вагових коефіцієнтів [7]. Методи реалізовані відповідно до наведеного псевдокоду з наступними уточненнями:

- 1) При обчисленні розподілу $p(z(k) | x(k), y(k))$ за формулою для гранул, координати яких $(x(k), y(k))$ знаходяться поза лабіринтом, отримане значення штрафується – помножується на емпіричний коефіцієнт 0,5. Таким чином зменшується вага тих гранул, які завідомо не відповідають точному положенню робота;
- 2) У APF фільтрі в якості $\mu^i(k)$ використовується реалізація з перехідного розподілу стану робота:

$$\mu^i(k) \sim p(x(k), y(k) | x^i(k-1), y^i(k-1))$$
Тоді розподіл $p(z(k) | \mu^i(k))$ також має форму (24);
- 3) У RPF в якості розподілу, що пропонується $q(x(k), y(k) | x^i(k-1), y^i(k-1), z(k))$ використовується апіорний $p(x(k), y(k) | x^i(k-1), y^i(k-1))$;
- 4) У RPF в якості статистичного ядра використовується гаусівське ядро у двовимірному просторі, оскільки для нього зручно генерувати реалізації \mathcal{E} ;
- 5) У RPF порогове значення N_T емпірично обране рівним $0,2 * N_S$. (коефіцієнт можна змінити як константу в кодї програми).

Опис програми. Після запуску програми для симуляції поставленої задачі вимальовується поле лабіринту (рис. 2) відповідно до заданої

структури. Біле тло означає вільне місце, чорне – стіни, блакитні точки по кутах деяких квадратів зі стінами – станції.

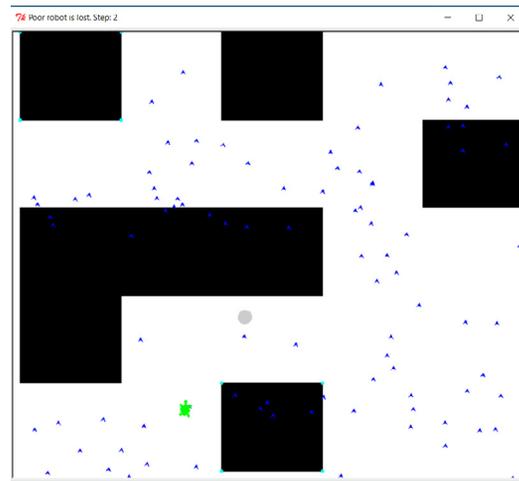


Рис. 2. Поле лабіринту.

Положення та напрямок руху робота відображає черепашка. Робот змінює напрямок руху, якщо врізається в перешкоду, виходить за межі лабіринту або занадто довго (10 кроків) рухається в одному напрямку.

Стрілки відображають поточні положення та напрямок руху гранул.

Круг позначає поточне математичне сподівання положення гранул, яке є оцінкою положення робота. Він змінює колір на зелений, коли 95% гранул знаходяться на відстані не більш ніж 1 від математичного сподівання.

У рядку заголовка вікна виводиться номер поточного кроку.

В результаті реалізації заданого методу гранулярної фільтрації (рис. 3) у випадку його збіжності оцінка положення робота з певного кроку стає близькою до його справжнього положення, більшість гранул скупчується навколо нього.

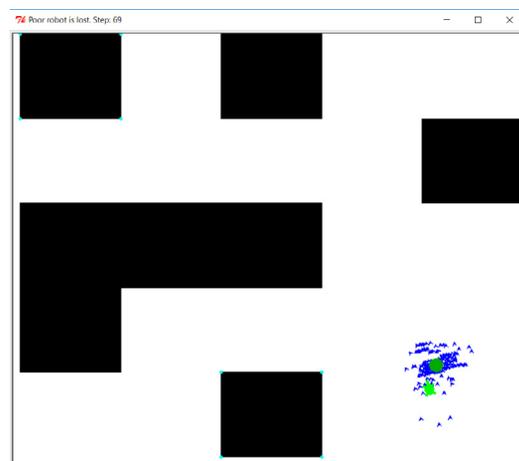


Рис. 3. Реалізація методу гранулярної фільтрації.

Для виконання лабораторної роботи надаються такі файли:

- main.py
- draw.py
- particle_filter.py

Швидкість робота (встановлено рівною 0.2) та кількість кроків (32), після якої гарантовано змінюється напрямок руху, задано як константи в `particle_filter.py`, за бажанням вони можуть бути змінені.

Вимоги для роботи програми. Програма для симуляції створена мовою програмування Python. Тому для її роботи необхідні Python 3 або Python 2.7 і такі бібліотеки, від яких залежить програмний код:

```
numpy, random, math, turtle, json,
time
```

Виконання дослідження

1. Запустити на виконання файл `main.py` та подивитися на роботу прикладу.

2. Створити новий файл або модифікувати `main.py`. Імпортувати наступні необхідні класи та функції з наданих модулів:

```
from particle_filter import Robot,
run_model from draw import Maze
```

3. Задати в коді або організувати введення з файлу власної мапи лабіринту у вигляді двовимірної прямокутної структури – списку списків, кортежу (tuple) кортежів абошо. Розмір лабіринту довільний, наприклад такий:

```
maze_map = ( ( 2, 0, 1, 0, 0 ),
              ( 0, 0, 0, 0, 1 ),
              ( 1, 1, 1, 0, 0 ),
              ( 1, 0, 0, 0, 0 ),
              ( 0, 0, 2, 0, 1 ) )
```

Введено такі позначення:

0 – порожній квадрат;

1 – квадрат зі стіною;

2 – квадрат зі стіною, по 4 кутах якого знаходяться станції.

4. Створити об'єкт класу `Maze`, задавши карту лабіринту:

```
world = Maze(maze_map)
```

5. Обрати параметри випадкових величин σ з діапазону [0.2;0.8] та Δ з діапазону [0.05;0.6]:

```
sigma = 0.3
delta = 0.5
```

6. Обрати для робота випадкове початкове вільне положення:

```
robot_position =
world.random_free_place()
```

7. Створити об'єкт класу `Robot`, задавши параметри випадкових величин у його моделі та початкове положення:

```
robot = Robot(sigma, delta,
robot_position)
```

8. Задати кількість гранул N_S (в залежності від розміру лабіринту) та кількість кроків оцінювання (бажано не менше 200):

```
particle_count = 1500
steps = 100
```

9. Запустити модель на виконання з необхідними параметрами. Цей запуск буде тестовим. Ініціалізовані в ньому випадкові початкові координати робота та гранул, а також дві послідовності випадкових кутів (отриманих з компаса та справжніх, враховуючи похибку), які задають напрямок руху робота, будуть збережені в json-файл для того, щоб в наступних кроках можна було порівняти результати трьох алгоритмів гранулярної фільтрації за однакових початкових умов та однакової поведінки досліджуваного об'єкта. В цьому запуску за замовчуванням використано `SIRParticleFilter`.

Значення параметра `save_parameters=True` спричинить створення файлу `"model_parameters.json"` після завершення роботи моделі:

```
run_model(robot, world, sigma,
delta, particle_count=
particle_count, steps=steps,
save_parameters=True)
```

Створений файл `"model_parameters.json"` має наступну структуру:

```
{
  "init_robot_pos": [
    50,
    50
  ],
  "init_particles_pos": [
    [
      20,
      20
    ],
    [
      30,
      30
    ],
    ],...
  ],
  "robot_actual_angles": [
    1.1,
    1.4,
    0.35,...
  ],
  "robot_compass_angles": [
    1.2,
    1.2,
    0.3,...
  ]
}
```

10. Організувати зчитування json-файлу `"model_parameters.json"` з початковими параметрами та параметрами поведінки об'єкта. Пропонується використовувати функцію `json.load` бібліотеки `json`.

11. Створити об'єкт класу `Robot`, додатково вказавши його початкове положення та список кутів:

```
robot = Robot(sigma, delta,
init_robot_pos,
actual_angles=robot_actual_angles,
compass_angles=robot_compass_angles)
```

12. Запустити модель на виконання для трьох алгоритмів гранулярної фільтрації: “SIR Particle Filter”, “Auxiliary Particle Filter”, “Regularized Particle Filter”. Оскільки вказуються початкові положення гранул, а також для робота вказаний список кутів, задавати кількість гранул та кроків не потрібно. Алгоритм фільтрації задається параметром *filtering_algorithm*. Наприклад:

```
result = run_model(robot, world,
sigma, delta, init_particles_pos=
init_particles_pos, filtering_
algorithm="SIRParticleFilter")
```

Результатом, який повертає функція *run_model*, є список елементів вигляду: ((3.5, 4.75), (2.1, 1.7)), де перша пара чисел є справжньою координатою робота на відповідному кроці, а друга пара – математичним сподіванням координати робота для обчисленого апостеріорного розподілу на цьому кроці.

13. Побудувати графіки зміни в часі відстані від справжньої координати об'єкта до її оцінки математичним сподіванням для кожного з трьох алгоритмів гранулярної фільтрації.

14. За бажанням дослідити вплив параметрів (σ , Δ , кількості гранул N_S , розміру лабіринту, кількості станцій, порогового значення N_T в RPF) на точність оцінювання.

Висновки

1. Випадкові збурення станів динамічних систем негативно впливають на якість оцінювання станів, тому ця задача потребує уваги дослідників, які працюють над зменшенням впливу випадкових збурень станів на значення (оцінки) змінних на виході систем. Шуми (похибки) вимірів і збурення станів враховуються у явному вигляді за допомогою математичних моделей динамічних систем у просторі станів, які набули широкого застосування особливо при розв'язуванні задач синтезу систем керування.

2. У задачах моделювання часових рядів у просторі станів основним поняттям є вектор стану, який містить всю необхідну для опису спостережуваної системи інформацію у конкретній постановці задачі. Вектор вимірів представляє зашумлені спостереження, пов'язані з вектором стану. Він може мати меншу розмірність від вектора стану внаслідок наявності не вимірюваних компонент.

3. При використанні ймовірнісних методів аналізу даних для формування ймовірнісного висновку щодо поточного стану динамічної системи необхідно будувати принаймні дві моделі. По-перше, модель, що описує зміну стану системи в часі (модель динаміки системи або

змінних її стану); по-друге, модель, що пов'язує зашумлені виміри компонент вектора стану із процесами наявних похибок (модель вимірів). Такі моделі у просторі станів мають бути доступними для дослідження та практичного застосування у ймовірнісній формі [7, 8].

4. Задачу лінійної та нелінійної фільтрації є формування (обчислення) ймовірнісного висновку щодо стану системи, виходячи з наявних вимірів. В межах Байєсівського підходу до аналізу даних це відбувається шляхом обчислення або апроксимації апостеріорного розподілу вектора стану за умови використання усіх наявних на момент обчислення вимірів та оцінок не вимірюваних компонент. Оскільки функція розподілу ймовірностей вимірів практично містить у собі всю доступну статистичну інформацію стосовно досліджуваного об'єкта, то її оцінювання є достатньо повним розв'язком задачі оцінювання стану, прогнозування його подальшого розвитку та підтримки прийняття рішень [8].

Список використаних джерел

1. B.D.O. Anderson, J.B. Moore, Optimal Filtering. – Englewood Cliffs, NJ: Prentice Hall, Inc., 1979. – 367 p.
2. П.І. Бідюк, В.Д. Романенко, О.Л. Тимошук – Аналіз часових рядів. – Київ: Політехніка, 2011. – 608 с.
3. S.M. Kay, Fundamentals of Statistical Signal Processing: Estimation Theory. – Upper Saddle River, NJ: Prentice Hall, 1993, 595 p.
4. C.K. Chui, G. Chen, Kalman Filtering with Real-Time Applications. – Berlin: Springer, 2009. – 239 p.
5. S. Haykin, Adaptive Filtering Theory. – Upper Saddle River NJ: Prentice Hall, 2007, 920 p.
6. М.З. Згуровский, В.Н. Подладчиков. Аналитические методы калмановской фильтрации. – Киев: Наукова думка, 1997. – 320 с.
7. S.J. Press, Subjective and Objective Bayesian Statistics. – Hoboken, NJ: John Wiley & Sons, Inc., 2003. – 558 p.
8. A. Pole, M. West, J. Harrison, Applied Bayesian Forecasting and Time Series Analysis. – Boca Raton, FL: Chapman & Hall/CRC, 2000. – 410 p.
9. D. Liu, Z. Wang. Recursive filtering for stochastic parameter systems with measurement quantizations and packet disorders. – Elsevier, Applied Mathematics and Computation, Vol. 398, 2021. DOI: <https://doi.org/10.1016/j.amc.2021.125960>.
10. L. Tan, C. Li. Output feedback leader-following consensus for nonlinear stochastic multiagent systems: The event-triggered method. - Elsevier, Applied Mathematics and Computation, Vol. 395, 2021. DOI: <https://doi.org/10.1016/j.amc.2020.125879>.
11. L.F. Chaparro, A. Akan. Signals and Systems Using MATLAB (Third edition). – Academic Press, 2019. DOI: <https://doi.org/10.1016/B978-0-12-814204-2.00023-5>.
12. V. Yilmaz. Automated ground filtering of LiDAR and UAS point clouds with metaheuristics. – Elsevier, Optics & Laser Technology, Vol. 138, 2021. DOI: <https://doi.org/10.1016/j.optlastec.2020.106890>.

13. B. Acciaio, J. Backhoff-Veraguas. Causal optimal transport and its links to enlargement of filtrations and continuous-time stochastic optimization. – Elsevier, *Stochastic Processes and their Applications*, Vol. 130, 2020. DOI: <https://doi.org/10.1016/j.spa.2019.08.009>.
14. L. Coutin, A. Lejay. Sensitivity of rough differential equations: An approach through the Omega lemma. – Elsevier, *Journal of Differential Equations*, Vol. 264, 2018. DOI: <https://doi.org/10.1016/j.jde.2017.11.031>.
15. Z. Weidong, Li. Guanhua. Positioning error compensation on two-dimensional manifold for robotic machining. – Elsevier, *Robotics and Computer-Integrated Manufacturing*, Vol. 59, 2019. DOI: <https://doi.org/10.1016/j.rcim.2019.05.013>.
16. A. Klimchik, A. Pashkevich. CAD-based approach for identification of elasto-static parameters of robotic manipulators. – Elsevier, *Finite Elements in Analysis and Design*, Vol. 75, 2013. DOI: <https://doi.org/10.1016/j.finel.2013.06.008>.
17. S.H.H. Zargarbashi, W. Khan. Posture optimization in robot-assisted machining operations. – Elsevier, *Mechanism and Machine Theory*, Vol. 51, 2012. DOI: <https://doi.org/10.1016/j.mechmachtheory.2011.11.017>.
18. S. Klecker, B. Hichri. Robotic trajectory tracking: Bio-inspired position and torque control. – Elsevier, *Procedia CIRP*, Vol. 88, 2020. DOI: <https://doi.org/10.1016/j.procir.2020.05.108>.
19. M.C. Pérez-Rubio, C. Losada-Gutiérrez, F. Espinosa. A realistic evaluation of indoor robot position tracking systems: The IPIN 2016 competition experience. – Elsevier, *Measurement*, Vol. 135, 2019. DOI: <https://doi.org/10.1016/j.measurement.2018.11.018>.
20. S. Sun, J. Zhao. Path planning for multiple mobile anchor nodes assisted localization in wireless sensor networks. – Elsevier, *Measurement*, Vol. 141, 2019. DOI: <https://doi.org/10.1016/j.measurement.2019.03.016>.
8. A. Pole, M. West, J. Harrison, *Applied Bayesian Forecasting and Time Series Analysis*. – Boca Raton, FL: Chapman & Hall/CRC, 2000. – 410 p.
9. D. Liu, Z. Wang. Recursive filtering for stochastic parameter systems with measurement quantizations and packet disorders. – Elsevier, *Applied Mathematics and Computation*, Vol. 398, 2021. DOI: <https://doi.org/10.1016/j.amc.2021.125960>.
10. L. Tan, C. Li. Output feedback leader-following consensus for nonlinear stochastic multiagent systems: The event-triggered method. – Elsevier, *Applied Mathematics and Computation*, Vol. 395, 2021. DOI: <https://doi.org/10.1016/j.amc.2020.125879>.
11. L.F. Chaparro, A. Akan. *Signals and Systems Using MATLAB (Third edition)*. – Academic Press, 2019. DOI: <https://doi.org/10.1016/B978-0-12-814204-2.00023-5>.
12. V. Yilmaz. Automated ground filtering of LiDAR and UAS point clouds with metaheuristics. – Elsevier, *Optics & Laser Technology*, Vol. 138, 2021. DOI: <https://doi.org/10.1016/j.optlastec.2020.106890>.
13. B. Acciaio, J. Backhoff-Veraguas. Causal optimal transport and its links to enlargement of filtrations and continuous-time stochastic optimization. – Elsevier, *Stochastic Processes and their Applications*, Vol. 130, 2020. DOI: <https://doi.org/10.1016/j.spa.2019.08.009>.
14. L. Coutin, A. Lejay. Sensitivity of rough differential equations: An approach through the Omega lemma. – Elsevier, *Journal of Differential Equations*, Vol. 264, 2018. DOI: <https://doi.org/10.1016/j.jde.2017.11.031>.
15. Z. Weidong, Li. Guanhua. Positioning error compensation on two-dimensional manifold for robotic machining. – Elsevier, *Robotics and Computer-Integrated Manufacturing*, Vol. 59, 2019. DOI: <https://doi.org/10.1016/j.rcim.2019.05.013>.
16. A. Klimchik, A. Pashkevich. CAD-based approach for identification of elasto-static parameters of robotic manipulators. – Elsevier, *Finite Elements in Analysis and Design*, Vol. 75, 2013. DOI: <https://doi.org/10.1016/j.finel.2013.06.008>.
17. S.H.H. Zargarbashi, W. Khan. Posture optimization in robot-assisted machining operations. – Elsevier, *Mechanism and Machine Theory*, Vol. 51, 2012. DOI: <https://doi.org/10.1016/j.mechmachtheory.2011.11.017>.
18. S. Klecker, B. Hichri. Robotic trajectory tracking: Bio-inspired position and torque control. – Elsevier, *Procedia CIRP*, Vol. 88, 2020. DOI: <https://doi.org/10.1016/j.procir.2020.05.108>.
19. M.C. Pérez-Rubio, C. Losada-Gutiérrez, F. Espinosa. A realistic evaluation of indoor robot position tracking systems: The IPIN 2016 competition experience. – Elsevier, *Measurement*, Vol. 135, 2019. DOI: <https://doi.org/10.1016/j.measurement.2018.11.018>.
20. S. Sun, J. Zhao. Path planning for multiple mobile anchor nodes assisted localization in wireless sensor networks. – Elsevier, *Measurement*, Vol. 141, 2019. DOI: <https://doi.org/10.1016/j.measurement.2019.03.016>.

References

1. B.D.O. Anderson, J.B. Moore, *Optimal Filtering*. – Englewood Cliffs, NJ: Prentice Hall, Inc., 1979. – 367 p.
2. P.I. Bidyuk, V.D. Romanenko, O.L. Tymoschuk - Analysis of time series. - Kyiv: Polytechnic, 2011. - 608 p. (in Russian)
3. S.M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. – Upper Saddle River, NJ: Prentice Hall, 1993, 595 p.
4. C.K. Chui, G. Chen, *Kalman Filtering with Real-Time Applications*. – Berlin: Springer, 2009. – 239 p.
5. S. Haykin, *Adaptive Filtering Theory*. – Upper Saddle River NJ: Prentice Hall, 2007, 920 p.
6. M.Z. Zgurovskiy, V.N. Podladchikov. Analytical methods of Kalman filtration. - Kiev: Naukova Dumka, 1997. - 320 p. (in Russian)
7. S.J. Press, *Subjective and Objective Bayesian Statistics*. – Hoboken, NJ: John Wiley & Sons, Inc., 2003. – 558 p.