

СТВОРЕННЯ АВТОМАТИЗОВАНИХ СИСТЕМ ЗА ДОПОМОГОЮ СИМУЛЯТОРА РОБОТОТЕХНІКИ

Ануфрієв В.В.

Науковий керівник – д.ф.-м.н., проф. Бондаренко І.М.

Харківський національний університет радіоелектроніки, каф. МЕЕПІ,
м. Харків, Україна

e-mail: valentyn.anufriiev@nure.ua

Robots are important in any field of work, as they can perform monotonous and hazardous work for humans, thereby increasing productivity and safety in the workplace. To develop them, robotics simulators are used to develop and test algorithms by simulating scenarios in a virtual environment, reducing the costs and risks associated with real-world testing. The paper of this article is to consider the development process and advantages of using robotic simulators in the development of automated systems.

Використання симуляторів має велике значення для розробки бюджетних роботизованих систем, оскільки дозволяє тестувати та вдосконалювати їх у віртуальному середовищі, зменшуючи потребу в дорогих фізичних прототипах [1].

Мета даної роботи – розглянути процес та переваги використання симуляторів для розробки автоматизованих систем.

Webots як і більшість симуляторів робототехніки складається з декількох компонентних блоків, які працюють разом, щоб імітувати поведінку роботів у віртуальному середовищі [2]. Хоча конкретна структура може відрізнятися залежно від симулятора, загальні компоненти включають в себе наступне:

- фізичний рушій (фізична взаємодія між роботами та модельованим оточенням);
- графічний рендеринг (генерація візуального представлення симульованого середовища та роботів);
- моделі роботів: (геометрія, кінематика та динаміка симульованих роботів);
- моделі середовища (віртуальний світ, в якому працюють роботи);
- сенсорні моделі: (імітація поведінки сенсорів, таких як камери, LIDAR тощо);
- інтерфейс керування (взаємодія користувача з симулятором, наприклад, відправка команд роботам, зміна параметрів симуляції та перегляд результатів симуляції);
- двигун симуляції (організація взаємодії між різними компонентами симулятора);
- API та бібліотеки (розширення функціоналу симулятора).

Побудова моделі робота ґрунтується на поєднанні двох підходів розробки програмного забезпечення [3]:

- на основі застосування існуючої компонентної бази;
- на основі симуляції умов навколишнього середовища.

Підхід з використанням існуючої компонентної бази зосереджується на розробці програмних систем з використанням готових програмних компонентів, які можна використовувати повторно. Він спрямований на підвищення продуктивності, зниження витрат на розробку та підвищення якості програмного забезпечення за рахунок використання існуючих компонентів.

Симуляція умов навколишнього середовища зосереджується на автоматизації тестування взаємодії у компонентах програмного забезпечення роботів за допомогою симуляції. Цей підхід автоматично генерує тестові умови, застосовуючи методи на основі специфікацій та враховуючи залежні від симуляції параметри. Під час тестування контролюються тестові параметри, які впливають на поведінку компонента (параметри, залежні від симуляції, вхідні/вихідні параметри необхідного інтерфейсу тощо). Основною перевагою цього методу є те, що він дозволяє виявити помилки, спричинені взаємодією між компонентом та модельованим середовищем.

Використання цих підходів в одному застосунку дозволяє розробляти повністю функціональне та протестоване програмне забезпечення для роботизованих систем з мінімальними витратами часу та ресурсів.

Процес реалізації роботизованої системи за допомогою Webots складається з 3 етапів:

1. Побудова фізичної моделі.
2. Розробка програмного забезпечення.
3. Симуляція робота.

На першому етапі проводиться реалізація фізичної моделі робота, яка може створена за допомогою бібліотек базових геометричних фігур або імпортована з систем автоматизованого проєктування, таких як SolidWorks, AutoCAD, 3Ds Max тощо. На цьому етапі також проводиться розміщення датчиків, налаштування рухомих елементів та фізична взаємодія з навколишнім середовищем.

Другий етап – програмування поведінки робота. Для цього існують різні інструменти програмування. Вони включають високорівневі мови програмування (C/C++, Java та Python) та пакет для числового програмування Matlab. Програма, що керує роботом, як правило, являє собою нескінченний цикл, який поділяється на три частини:

1. Зчитування значень, вимірних датчиками робота.
2. Обчислення наступних дій робота.
3. Надсилання команд актюаторам.

Найпростішими є перша та третя частини. Найскладнішою є друга частина, оскільки саме тут проводиться обчислення отриманих даних та надсилання команд.

На третьому етапі проводиться симуляція робота. Вона дозволяє перевірити коректність побудованої фізичної моделі та розробленого алгоритму поведінки. При запуску симуляції, в режимі реального часу почне відображатися змодельований робот, що виконує розроблену програму. Також є можливість виведення значень, отримані датчиками, результати обробки програми тощо.

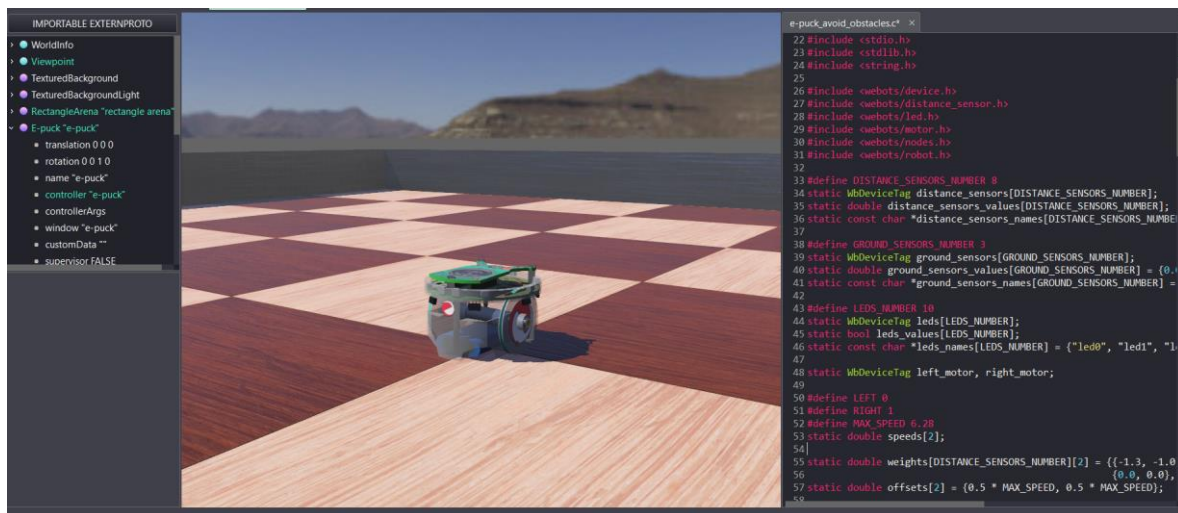


Рисунок 1 – Фізична модель робота та його програмне забезпечення в змодельованому віртуальному середовищі

Після симуляції, коли розроблене програмне забезпечення можна експортувати з віртуального середовища до фізичного робота для використання в реальних умовах.

Таким чином, симулятори робототехніки є ефективним засобом для розробки програмного забезпечення, що дозволяють реалізувати функціональну автоматизовану систему та проаналізувати її в умовах, наближених до реальних, економлячи кошти та час на розробку.

Список використаних джерел:

1. Створення бюджетного робота-асистента для медичних закладів / В. Ануфрієв та ін. Grail of Science. 2023. № 31. С. 197–201.
2. Kumar K., Reel P. S. Analysis of contemporary robotics simulators. 2019 International Conference on Emerging Trends in Electrical and Computer Technology (ICETECT 2019), м. Nagercoil, India, 23–24 берез. 2019 р. 2019.
3. SITAT: Simulation-based Interface Testing Automation Tool for Robot Software Component / J.-S. Kang та ін. Journal of Institute of Control, Robotics and Systems. 2020. Т. 16, № 6. С. 608–616.