

### **АЛГОРИТМИ БЕЗКОЛІЗІЙНОГО ПЕРЕМІЩЕННЯ СХВАТІВ ПРОМИСЛОВИХ РОБОТІВ**

На даному етапі розвитку робототехніки важливе місце посідають задачі оптимізації різних показників, які є необхідним при проектуванні/синтезі роботизованих механоскладальних технологій (РМСТ), що реалізуються в гнучких виробничих комірках (ГВК). Важливе місце серед них посідають траєкторні задачі, розв'язування яких неможливе без планування рухів промислових роботів (ПР) в ГВК та вибір оптимальної з них за певним критерієм.

Для реалізації планувальних задач контролююча система ПР повинна бути здатна будувати квазіоптимальну траєкторію, в ідеалі - оптимальну за попередньо прийнятим критерієм та вільну від зіткнень з іншими елементами ГВК (так звану безколізійну траєкторію). Побудова безперешкодного, близького до оптимального шляху необхідна для мінімізації часових та енергетичних затрат за рахунок бажаного зменшення відстані та збільшення швидкості маніпуляційної системи та схвата (Сх) ПР.

Задача пошуку найкоротшого шляху, який оминає перешкоди, в заданому просторі станів розглядається досить давно і включає багато алгоритмів їх вирішення, найпопулярнішими з яких є: KPIECE (Kinodynamic Motion Planning by Interior-Exterior Cell Exploration), Bi-directional KPIECE, Lazy Bi-directional KPIECE, PRM (probabilistic roadmap method), SBL (A Single-Query Bi-Directional Probabilistic Roadmap Planner with Lazy Collision Checking), RRT (Rapidly exploring random tree), RRT Connect, Lazy RRT та EST (Expansive Space Trees).

Змістовний аналіз визначених алгоритмів вказує наступне. Дані алгоритми досліджують простір станів, починаючи з початкової конфігурації  $q_{init}$  і до поки не досягнуть кінцевої конфігурації  $q_{goal}$ . Дослідження простору проходить шляхом розповсюдження дерев, що розростаються від початкової точки до кінцевої, оминаючи при цьому наявні перешкоди. В деяких із наведених вище алгоритмів одночасно будується два дерева  $T_{init}$  та  $T_{goal}$  (в прямому напрямку, від початкового стану  $q_{init}$ , та у зворотному напрямку, від цілі  $q_{goal}$ ), зупиняючись після того, як два дерева з'єднуються. Перевірка на колізії (зіткнення, перетини) в кожному із наведених алгоритмів проходить на різних стадіях їх роботи, але спільним є те, що у випадку виявлення перешкоди, сегмент, що її містить, видаляється, після чого процес пошуку продовжується.

За результатами роботи вказаних алгоритмів будуються безперешкодні (безколізійні) траєкторії. Для вибору оптимальної траєкторії використовується алгоритм вирізання кутів. Його стислий зміст зводиться до наступного. На траєкторії вибирається три суміжних вузла. Якщо переміщення по прямій із крайнього лівого вузла в крайній правий вузол можливо, то середній вузол відкидається. Процес повторюється ітеративно, доки не будуть видалені все можливі вузла. Тому для побудови траєкторії, близької до оптимальної, бажано використовувати одразу декілька алгоритмів, співставляти результати та оптимізувати отримані рішення.

Крім того, наведені вище алгоритми використовують повний перебір всіх можливих варіантів переміщення Сх ПР, а доцільність їх використання визначається тим, що найкоротший шлях стає відомим ще до початку маршруту.

Автоматизоване підтвердження працездатності аналізованих алгоритмів реалізовано в створеному програмному продукті (ПП). Даний ПП написаний на мові програмування C++ з використанням графічної бібліотеки OpenGL (Open Graphics Library) та QT, що являє собою крос-платформовий інструментарій розробки програмного забезпечення та включає в себе всі основні класи, які потрібні при розробці елементів графічного інтерфейсу. Також застосовується бібліотека OMPL (Open Motion Planning Library) для побудови самих траєкторій та бібліотека Assimp (Open Asset Import Library) для роботи з 3-D моделями.

Результатом роботи є параметри згладжуваності та шляху, який проходить технологічний роботизований комплект (ТРК) за РМСТ. Очевидною є необхідність вибору оптимального алгоритму із згаданих вище, що за своєю змістовною сутністю є задачею дискретної оптимізації на множині дискретних значень показників функціонування (згладжуваність, шлях) кожного із аналізованих алгоритмів. Вказане на сьогодні автоматизовано реалізовується в Житомирському державному технологічному університеті.