

DOI: 10.20535/1970.71(1).2026.361889

УДК 681.5

## КЛАСИФІКАЦІЯ РИЗИКІВ ТА ІНДИКАТОРІВ ВИЯВЛЕННЯ ВІДМОВ У СИСТЕМАХ НАВЧАННЯ ЗА ДЕМОНСТРАЦІЄЮ ДЛЯ ПРОМИСЛОВИХ РОБОТІВ-МАНІПУЛЯТОРІВ

Сенчук Ю. М., Матіко Ф. Д.

Національний університет «Львівська політехніка», Львів, Україна

E-mail: [yurii.m.senchuk@lpnu.ua](mailto:yurii.m.senchuk@lpnu.ua), [fedir.d.matiko@lpnu.ua](mailto:fedir.d.matiko@lpnu.ua)

У статті розглянуто процес формування програм руху промислових роботів-маніпуляторів у системах навчання за демонстрацією (Learning from Demonstration, LfD). Показано, що попри ефективність LfD-підходу для скорочення часу програмування та підвищення гнучкості роботизованих технологічних процесів, їх практичне впровадження ускладнюється багатоступеневим перетворенням демонстрацій оператора у виконувану програму керування. У межах цього процесу похибки можуть накопичуватися, залишатися невиявленими під час локальної перевірки та проявлятися на пізніх стадіях верифікації, генерації програмного коду або безпосереднього виконання руху маніпулятором. Це створює потребу в систематизації ризиків не лише для окремих алгоритмічних процедур, а й для всієї послідовності формування програми руху.

Авторами виділено 10 послідовних етапів формування програми руху – від постановки вимог і отримання демонстрацій до генерації програмного коду та допуску програми до виконання на реальному роботі-маніпуляторі. Зазначена послідовність охоплює роботу з демонстраційними даними, представлення траєкторії, кінематичні та часові перетворення, перевірку обмежень, симуляційну верифікацію і формування керуючої програми. Виконано аналіз та систематизовано джерела ризиків, потенційні відмови, індикатори їх виявлення та можливі форми пізнього прояву відхилень в межах кожного етапу. За результатами аналізу сформовано класифікацію характерних ризиків, індикаторів раннього виявлення та можливих пізніх проявів похибок і відмов. Розроблено матрицю ймовірних проявів відхилень (відмов) залежно від етапу їх виникнення. Вона ілюструє можливість міжетапного накопичення похибок під час розроблення програми робота та дає змогу простежити зв'язок між етапом виникнення відхилення і стадією, на якій його наслідки можуть стати помітними. Отримані результати можуть бути використані під час проектування LfD-систем, розроблення процедур офлайн-верифікації та засобів контролю надійності й безпечності роботизованих технологічних процесів.

**Ключові слова:** робот-маніпулятор, навчання за демонстрацією (LfD), похибка, прихована відмова, програма керування, оптимізація траєкторії, кінематичні та динамічні обмеження, класифікація ризиків і індикаторів відмов.

### Вступ

Зростання складності промислових технологічних процесів і вимог до гнучкості виробництва підвищує трудомісткість класичного програмування багатоосьових роботів-маніпуляторів. У цьому контексті методи навчання за демонстрацією (Learning from Demonstration, LfD) розглядаються як перспективний інструмент формування програм руху на основі показу оператора, що дає змогу скоротити час підготовки та знизити залежність від ручного програмування.

Водночас практичне застосування LfD у промислових умовах супроводжується інженерними ризиками, пов'язаними з багатоступеневим перетворенням демонстрацій у виконувану керуючу програму. Цей процес охоплює отримання демонстрацій, оброблення й представлення траєкторії, кінематичні перетворення, перевірку обмежень, генерацію програмного коду та його допуск до виконання на реальному роботі-маніпуляторі. У такому ланцюжку окремі похибки можуть накопи-

чуватися, залишатися непомітними на етапі виникнення та впливати на результати подальшої перевірки, генерації або виконання руху.

Актуальність дослідження зумовлена відсутністю систематизованих підходів до аналізу надійності повного технологічного ланцюга LfD-систем для промислових роботів. Більшість існуючих робіт зосереджена на окремих алгоритмічних аспектах, тоді як комплексний аналіз та класифікація ризиків і відмов на всіх етапах формування траєкторії, із врахуванням міжетапного накопичення відхилень і їх можливого пізнього прояву, потребують розвитку та доопрацювання.

### Аналіз останніх досліджень

У сучасних міжнародних дослідженнях проблематика навчання за демонстрацією для роботів-маніпуляторів розглядається переважно з позицій підвищення ефективності програмування та адаптивності робото-технічних систем. У працях A. Berekatain [1], W. Li [2] та K. Wang [3] підходи LfD

розглядаються як перспективний інженерний інструмент для промислових застосувань. Питання якості демонстрацій і впливу поведінки оператора на результати навчання висвітлюються у працях М. Bilal [4], А. Sena і М. Howard [5]. У роботах Calzada-Garcia et al. [6], Tamizi et al. [7], Yonezawa et al. [8] значну увагу також надано задачам інверсної кінематики, процедурам планування та оптимізації руху.

Разом із тим аналіз публікацій показує, що наявні дослідження переважно розглядають ризики локально – в межах окремих процедур (оброблення демонстрацій, апроксимації траєкторій, інверсної кінематики, часової параметризації або генерації коду) – без простеження механізмів їх накопичення та подальшого прояву у вигляді ризикованих відмов на наступних етапах.

**Метою роботи** є розроблення класифікації типових ризиків, потенційних відмов та індикаторів їх раннього виявлення на усіх етапах формування програм руху багатоосьових роботів-маніпуляторів у системах навчання за демонстрацією. Для досягнення поставленої мети у роботі вирішено такі завдання: виокремити основні етапи формування програми руху в LfD-системі; класифікувати характерні ризики, потенційні відмови та індикатори їх раннього виявлення для кожного етапу.

### Виклад основного матеріалу

Навчання за демонстрацією (LfD) у задачах промислової робототехніки передбачає перетворення демонстрацій оператора у програму руху робота-маніпулятора. На відміну від ручного програмування, такий підхід дає змогу зменшити трудомісткість підготовки програм і підвищити гнучкість роботизованих технологічних процесів [1].

Разом із цим, формування програми руху в LfD-системі є багатоступеневим процесом, що охоплює отримання демонстрацій, оброблення даних, представлення траєкторії, кінематичні перетворення, перевірку обмежень і генерацію керуючих команд. У межах цієї послідовності похибки, неточні припущення або некоректні параметри можуть накопичуватися та впливати на результати подальших етапів. Тому для аналізу ризиків доцільно розглядати не окремі алгоритмічні процедури, а повну послідовність формування програми руху робота-маніпулятора.

Авторами виконано аналіз процедури навчання за демонстрацією, а також представлення цієї процедури та її окремих етапів у джерелах [1], [2], [4 – 7], на основі чого сформовано узагальнену технологічну послідовність створення програми руху маніпулятора, представлену на рис. 1.



Рис. 1. Узагальнена послідовність етапів формування програми руху в системі навчання за демонстрацією для промислового робота-маніпулятора

Під час аналізу послідовності етапів формування програми руху в системі навчання за демонстрацією терміни «ризик», «відмова», «індикатор виявлення відмови» застосовано у такому трактуванні. Поняття **ризик** розглядається відповідно до підходу ISO 31000, у якому ризик пов'язується з впливом невизначеності на досягнення цілей [9]. У межах цієї статті такими цілями є коректність сформованої траєкторії, відтворюваність руху та безпечність подальшого виконання програми промисловим роботом-маніпулятором. Тому ризик у системі навчання за демонстрацією пов'язуємо з ймовірністю того, що сукупність умов, джерела невизначеності або відхилення на певному етапі формування програми руху негативно вплинуть на

досягнення зазначених цілей на цьому або наступних етапах.

Потенційна **відмова** у цій роботі трактується не як фізична несправність апаратного вузла, а як можливий некоректний, неповний або непридатний результат виконання окремого етапу формування програми руху. Йдеться про ситуацію, за якої етап не забезпечує очікуваного результату, необхідного для подальшого коректного формування, перевірки або виконання програми. Таке трактування узгоджується з підходами системного аналізу, де відмова пов'язується з невиконанням системою або її компонентом потрібної функції у встановлених межах [10], а також із загальною логікою поширення помилок і відмов у складних

технічних та програмних системах [11].

**Індикатором** виявлення відмови вважаємо контрольовану ознаку, показник або результат перевірки, що вказує на наявність некоректного результату окремого етапу або на підвищену ймовірність його виникнення до запуску програми на реальному роботі. Таке розуміння узгоджується з підходами до використання ключових індикаторів ризику, які застосовують для раннього виявлення ознак потенційних проблем до того, як вони призведуть до відмови або іншого небажаного наслідку [12].

**Можливий пізній прояв** у цій роботі розглядається як наслідок відхилення або потенційної відмови, що виникли на попередньому етапі, але можуть бути виявлені або суттєво вплинути на

результат на одному з подальших етапів формування, перевірки чи виконання програми руху.

У табл. 1 наведено класифікацію характерних ризиків, потенційних відмов, індикаторів їх виявлення та можливих форм пізнього прояву відхилень під час етапів формування програми руху в LfD-системі.

Запропонована класифікація має якісний характер і не передбачає кількісного оцінювання ймовірності відмов або розрахунку інтегрального показника ризику. Її призначення полягає у виявленні типових умов виникнення відхилень, визначенні ознак їх раннього виявлення для забезпечення процедур офлайн-верифікації програм руху до запуску на реальному робототехнічному обладнанні.

Таблиця 1. Класифікація характерних ризиків, потенційних відмов, індикаторів їх виявлення та можливих форм пізнього прояву відхилень

Етап	Характерні ризики та потенційні відмови	Індикатори виявлення	Можливий прояв відхилень на наступних етапах
<b>1. Постановка вимог</b>	<ul style="list-style-type: none"> <li>• Суперечливі або неповні вимоги;</li> <li>• відсутність пріоритетів між точністю, плавністю та часом циклу;</li> <li>• ігнорування обмежень робота</li> </ul>	<ul style="list-style-type: none"> <li>• Неузгодженість технічного завдання;</li> <li>• відсутність чітко заданих метрик;</li> <li>• неможливість ранжування критеріїв</li> </ul>	<ul style="list-style-type: none"> <li>• Неможливість отримати прийнятну траєкторію;</li> <li>• конфлікти під час оптимізації та приймання</li> </ul>
<b>2. Отримання демонстрацій</b>	<ul style="list-style-type: none"> <li>• Вимірювальний шум;</li> <li>• аномальні викиди;</li> <li>• пропуски даних;</li> <li>• нерівномірна дискретизація;</li> <li>• часові затримки;</li> <li>• точки поза робочою зоною робота</li> </ul>	<ul style="list-style-type: none"> <li>• Стрибки координат;</li> <li>• аномальні коливання швидкості;</li> <li>• систематичний вихід траєкторії за межі досяжності</li> </ul>	<ul style="list-style-type: none"> <li>• Помилки сегментації;</li> <li>• нестійкість апроксимації;</li> <li>• недосяжні конфігурації на етапі інверсної кінематики</li> </ul>
<b>3. Попередня обробка даних</b>	<ul style="list-style-type: none"> <li>• Надмірне згладжування;</li> <li>• втрата локальних особливостей руху;</li> <li>• спотворення швидкості та прискорення;</li> <li>• помилки визначення меж сегментів</li> </ul>	<ul style="list-style-type: none"> <li>• Необгрунтоване зменшення пікових прискорень або ривка;</li> <li>• різка зміна параметрів у точках сегментації</li> </ul>	<ul style="list-style-type: none"> <li>• Некоректне компактне представлення траєкторії;</li> <li>• помилки часової параметризації;</li> <li>• втрата відтворюваності руху</li> </ul>
<b>4. Компактне представлення траєкторії</b>	<ul style="list-style-type: none"> <li>• Груба апроксимація траєкторії;</li> <li>• втрата деталей у зонах високої кривизни;</li> <li>• локальні осциляції апроксимованої траєкторії;</li> <li>• неузгодженість похідних у вузлових точках;</li> <li>• некоректний спосіб опису орієнтації робочого органа;</li> <li>• розривність параметрів орієнтації за безперервного руху</li> </ul>	<ul style="list-style-type: none"> <li>• Перевищення допустимої просторової похибки;</li> <li>• розриви похідних у вузлових точках;</li> <li>• локальні осциляції траєкторії;</li> <li>• непропорційні зміни швидкості або прискорення;</li> <li>• стрибки параметрів орієнтації;</li> <li>• аномальні значення кутової швидкості</li> </ul>	<ul style="list-style-type: none"> <li>• некоректні розв'язки кінематики;</li> <li>• нестійка часова параметризація;</li> <li>• порушення плавності руху робочого органа;</li> <li>• перевищення допустимих швидкостей або прискорень після параметризації</li> </ul>

Продовження табл. 1

Етап	Характерні ризики та потенційні відмови	Індикатори виявлення	Можливий прояв відхилень на наступних етапах
<b>5. Інверсна кінематика</b>	<ul style="list-style-type: none"> <li>• Недосяжні положення;</li> <li>• стрибки між альтернативними розв'язками;</li> <li>• наближення до сингулярностей;</li> <li>• порушення допустимих меж суглобових координат</li> </ul>	<ul style="list-style-type: none"> <li>• Непропорційно великі зміни кутів суглобів;</li> <li>• зростання числа обумовленості матриці Якобіана</li> </ul>	<ul style="list-style-type: none"> <li>• Локальні піки швидкості та прискорення;</li> <li>• нестійка часова параметризація;</li> <li>• непридатність траєкторії до виконання</li> </ul>
<b>6. Перевірка обмежень і підбір часових параметрів</b>	<ul style="list-style-type: none"> <li>• Перевищення швидкості, прискорення або ривка;</li> <li>• неможливість вкластися в цикл;</li> <li>• концентрація навантажень у критичних ділянках</li> </ul>	<ul style="list-style-type: none"> <li>• Локальні піки прискорень або ривка;</li> <li>• суттєве зростання тривалості руху;</li> <li>• критичні сегменти з високим навантаженням</li> </ul>	<ul style="list-style-type: none"> <li>• Відхилення від виробничих вимог;</li> <li>• нестабільна оптимізація;</li> <li>• непроходження симуляційної перевірки;</li> <li>• відмова у допуску програми до виконання</li> </ul>
<b>7. Оптимізація траєкторії</b>	<ul style="list-style-type: none"> <li>• Надмірна орієнтація на мінімізацію часу;</li> <li>• втрата плавності;</li> <li>• зростання динамічних навантажень;</li> <li>• чутливість до параметрів</li> </ul>	<ul style="list-style-type: none"> <li>• Варіативність результатів при малих змінах налаштувань;</li> <li>• непропорційне зростання навантажень;</li> <li>• формально допустимі, але інженерно непридатні рішення</li> </ul>	<ul style="list-style-type: none"> <li>• Нестабільна поведінка при симуляції;</li> <li>• складність приймання;</li> <li>• зниження відтворюваності програми</li> </ul>
<b>8. Симуляція та верифікація траєкторії</b>	<ul style="list-style-type: none"> <li>• Невідповідність між моделлю та реальною системою;</li> <li>• ігнорування дискретної природи команд;</li> <li>• спрощене моделювання оточення</li> </ul>	<ul style="list-style-type: none"> <li>• Розбіжність між гладкою симульованою траєкторією та траєкторією після дискретизації;</li> <li>• відмінності в поведінці окремих сегментів</li> </ul>	<ul style="list-style-type: none"> <li>• розбіжності під час перевірки згенерованої програми;</li> <li>• неочікувана поведінка програми на контролері або реальному роботі</li> </ul>
<b>9. Генерація та перевірка керуючої програми</b>	<ul style="list-style-type: none"> <li>• Синтаксичні та семантичні помилки;</li> <li>• некоректне задання базової системи координат або системи координат інструмента;</li> <li>• помилки одиниць виміру;</li> <li>• перевищення обмежень контролера</li> </ul>	<ul style="list-style-type: none"> <li>• Неможливість компіляції або завантаження;</li> <li>• розбіжність між візуалізованою траєкторією та інтерпретованим рухом</li> </ul>	<ul style="list-style-type: none"> <li>• непроходження офлайн-перевірки;</li> <li>• некоректне виконання програми на обладнанні</li> </ul>
<b>10. Допуск програми до виконання</b>	<ul style="list-style-type: none"> <li>• Відсутність чітко визначених критеріїв допуску;</li> <li>• суб'єктивність експертної оцінки;</li> <li>• відсутність порогових умов недопуску програми;</li> <li>• неповне врахування результатів попередніх перевірок</li> </ul>	<ul style="list-style-type: none"> <li>• Різні експерти приймають протилежні рішення щодо однієї програми;</li> <li>• неможливість однозначного висновку про придатність</li> </ul>	<ul style="list-style-type: none"> <li>• Помилковий допуск небезпечної або непридатної програми до реального виконання</li> </ul>

Наведені в таблиці 1 результати показують, що для надійності LfD-систем важливими є не тільки виявлення локальних відхилень в межах окремих етапів, а також і врахування міжетапного

накопичення похибок. У цьому випадку відхилення, що виникають на ранніх стадіях роботи з демонстраційними даними, можуть залишатися прихованими під час локального контролю та виявля-

тися на подальших етапах формування, перевірки, генерації або допуску програми до виконання.

Кожен із виділених етапів виконує окрему функцію у процесі формування програми руху та має власний набір характерних ризиків. Розгляд цих етапів у межах єдиного технологічного ланцюжка дає змогу простежити, як локальні відхилення, що виникають під час роботи з демонстраційними даними, подальшого представлення траєкторії або кінематичних перетворень, можуть впливати на наступні стадії верифікації, генерації коду та приймання програми.

Початковий етап формування програми руху полягає у визначенні цілей руху, критеріїв якості та допустимих відхилень. У контексті впровадження LfD для промислових маніпуляторів саме на етапі постановки задачі визначають, що має бути продемонстровано, які характеристики руху є критичними та за якими ознаками надалі оцінюватиметься прийнятність отриманої програми [1]. Для промислових роботів такі вимоги пов'язані не тільки з геометричною точністю траєкторії, а й із плавністю, часом виконання та дотриманням кінематичних і динамічних обмежень, що є типовими критеріями задач планування руху [7]. Тому основні ризики цього етапу пов'язані з неузгодженістю вимог, конфліктом між окремими критеріями та відсутністю чітких метрик для подальшого порівняння варіантів траєкторії.

Формування вихідного масиву просторово-часових даних відбувається під час отримання демонстрацій руху оператора. Якість таких демонстрацій безпосередньо впливає на подальше навчання та відтворення руху, оскільки в LfD саме демонстраційні дані є основою для формування майбутньої траєкторії [1], [5]. У роботі [4] показано, що якість окремих демонстрацій може оцінюватися за кількісними ознаками руху, які відображають придатність демонстраційних даних для подальшого навчання та відтворення траєкторії. Це узгоджується з необхідністю раннього виявлення аномальних координатних стрибків, нерівномірностей швидкості, пропусків даних та інших ознак, які можуть знизити якість подальшої реконструкції руху.

Попередня обробка демонстраційних даних спрямована на зменшення шуму, узгодження часових рядів і підготовку даних до подальшого представлення траєкторії. У загальній процедурі LfD цей перехід від сирих демонстрацій до придатного для навчання або відтворення подання є одним із ключових етапів формування роботизованого руху [1], [2]. Основна небезпека полягає в тому, що надмірна фільтрація або некоректне виділення меж сегментів можуть змінити структуру початкового руху. У такому разі похибка виникає ще до побудови параметричного опису траєкторії, але її наслідки можуть проявитися на подальших етапах апроксимації, часової параметризації або перевірки

ки обмежень.

Компактне представлення траєкторії забезпечує перехід від дискретних демонстраційних даних до параметричної або іншої узагальненої моделі руху, придатної для подальших перетворень. Для систем LfD вибір способу представлення траєкторії є принциповим, оскільки саме це подання визначає, як очищені демонстраційні дані надалі використовуються для кінематичних перетворень, часової параметризації та генерації програми руху [2]. На цьому етапі виникає компроміс між точністю відтворення початкових даних і складністю моделі: надто грубе подання може втрачати важливі особливості руху, тоді як надмірно складне подання може погіршувати плавність або стійкість подальшого відтворення. Тому якість представлення впливає не тільки на геометричну точність, а й на коректність наступних кінематичних і часових перетворень.

Перехід від опису руху робочого органа до узагальнених координат суглобів виконується на етапі визначення інверсної кінематики. Для маніпуляторів ця задача може ускладнюватися наявністю кількох допустимих розв'язків, недосяжних положень, обмежень суглобів і наближенням до особливих конфігурацій [6]. Окремі підходи до інверсної кінематики також враховують ефективність руху суглобів, що підтверджує важливість вибору не тільки формально допустимого, а й інженерно придатного розв'язку [8]. Тому відхилення, які не є очевидними у просторі робочого органа, можуть призводити до різких або неприйнятних змін у суглобовому просторі.

Узгодження сформованої траєкторії з кінематичними та динамічними можливостями роботоманіпулятора здійснюється під час перевірки обмежень і підбору часових параметрів. У задачах планування руху промислових роботів ключовими є обмеження швидкості, прискорення, плавності, часу виконання та допустимості руху в заданому робочому середовищі [7]. Саме на цьому етапі стає зрозуміло, чи може геометрично коректна траєкторія бути реалізована з урахуванням фізичних і технологічних обмежень конкретного робота. Через це дана стадія є однією з основних зон виявлення відхилень, що були сформовані на попередніх етапах.

Подальша оптимізація траєкторії спрямована на поліпшення часу виконання, плавності та ефективності руху за умови дотримання заданих обмежень. У задачах планування руху промислових роботів оптимізація траєкторій зазвичай пов'язана з пошуком компромісу між продуктивністю, точністю, плавністю та обмеженнями системи [7]. Водночас надмірна орієнтація на один критерій, наприклад мінімізацію часу циклу, може призводити до рішень, які формально задовольняють окрему цільову функцію, але є небажаними з погляду динамічного навантаження, стабільності або

технологічної придатності.

Симуляційна перевірка призначена для оцінювання поведінки сформованої траєкторії без безпосереднього залучення реального обладнання. Під час офлайн-програмування віртуальне середовище використовується для формування, перевірки та подальшого перенесення роботизованих програм у реальну виробничу систему [13]. Однак результати такої перевірки залежать від точності математичної моделі робота, робочого середовища, систем координат і прийнятих припущень щодо виконання команд. Тому на цьому етапі можуть виявлятися розбіжності між розрахунковою траєкторією та траєкторією, яка є придатною для подальшого виконання промисловим контролером.

На етапі генерації та перевірки керуючої програми траєкторні дані перетворюються у послідовність команд, придатних для виконання конкретним промисловим контролером. У роботах з офлайн-програмування роботів-маніпуляторів цей процес розглядається як перетворення даних із віртуального подання у формат, придатний для реальної роботизованої системи, з урахуванням робочої зони робота, систем координат та специфіки виконання команд промисловим контролером [13] **Ошибка!** **Источник ссылки не найден.** Важливість цього етапу зумовлена тим, що навіть коректно сформована траєкторія може втратити придатність до виконання через помилки її перетворення у формат конкретного контролера або через некоректне перенесення параметрів руху.

Завершальним етапом є допуск програми до виконання на реальному роботі. На цьому етапі узагальнюються результати попередніх перевірок, оцінюється відповідність програми заданим обмеженням і приймається рішення щодо прийнятності залишкових ризиків. На відміну від попередніх

етапів, тут не формується нова траєкторна модель, а перевіряється готовність уже сформованої програми до безпечного використання на реальному робототехнічному обладнанні.

Отже, розгляд етапів формування програми руху показує, що критичними для LfD-систем є не лише локальні відхилення в межах окремих процедур, а й їх міжетапне накопичення. Відхилення, що виникають під час оброблення демонстраційних даних або представлення траєкторії, можуть не мати очевидного прояву на етапі виникнення, але впливати на результати подальших етапів.

Показовим прикладом міжетапного накопичення відхилень є описана в [14] ситуація, коли похибка виникає не під час виконання руху, а значно раніше – на етапі математичного опису орієнтації робочого органа. У роботі [14] демонстрація руху спочатку реєструється на рівні суглобових координат за допомогою давачів положення, після чого через пряму кінематику відновлюється траєкторія робочого органа. Автори показали, що в окремих ділянках траєкторії обраний спосіб опису орієнтації може спричинити стрибкоподібну зміну її параметрів, хоча сам рух при цьому залишається безперервним. Таке спотворення, що виникає на стадії представлення траєкторії, надалі впливає на обчислення параметрів руху та погіршує коректність його відтворення. Отже, локальна похибка раннього етапу може переходити у проблеми на наступних стадіях генерації та виконання програми руху, що ілюструє механізм накопичення відхилень у системах навчання за демонстрацією.

Авторами виконано аналіз міжетапного накопичення відхилень, за результатами якого розроблено матрицю ймовірних проявів відхилень (відмов) залежно від етапу їх виникнення (див. рис. 2).



Рис. 2. Матриця ймовірних проявів відхилень (відмов) залежно від етапу їх виникнення

Нумерація етапів у матриці відповідає послідовності етапів, що представлена у табл. 1. Темні комірочки позначають типові випадки, коли відхилення, що виникло на відповідному етапі, може проявитися через свої наслідки на цьому самому або одному з наступних етапів. Білі комірочки відповідають випадкам, для яких типовий прояв наслід-

ків відхилення не передбачається. Світло-сірі комірочки нижче діагоналі позначають хронологічно неможливі ситуації, коли прояв наслідків відхилення передував би етапу його виникнення.

#### Висновки

У статті виконано аналіз процедури навчання

за демонстрацією, на основі чого сформовано узагальнену послідовність створення програми руху робота-маніпулятора із 10 етапів, що дало можливість проаналізувати ризики виникнення відхилень та відмов на кожному з етапів.

Розроблено класифікацію ризиків, потенційних відмов та індикаторів їх виявлення в LfD-системах програмування промислових багатоосьових роботів-маніпуляторів із врахуванням поділу процедури навчання за демонстрацією на етапи. Показано, що важливим є врахування не лише локальних відхилень в межах окремих процедур, а й міжетапного накопичення похибок. Відхилення, що виникли на ранніх етапах розроблення LfD-програми, з демонстраційними даними або поданням траєкторії, можуть не мати очевидного прояву на етапі виникнення, але впливати на подальшу перевірку, генерацію та виконання програми руху.

Отримані класифікація ризиків, потенційних відмов та індикаторів їх виявлення, а також матриця ймовірних проявів відхилень (відмов) залежно від етапу їх виникнення можуть бути використані для розроблення процедур офлайн-верифікації, уточнення критеріїв прийнятності програм руху та для розроблення засобів раннього виявлення відмов у LfD-системах ще до залучення реального робототехнічного обладнання, задіяного в технологічних процесах.

Перспективи подальших досліджень пов'язані з розробленням кількісних метрик оцінювання ризиків на окремих етапах формування програми руху, побудовою інтегральних показників надійності LfD-систем, а також дослідженням вірогідності виникнення відмов і ступеня впливу окремих ризиків для конкретних типів роботів-маніпуляторів, контролерів і виробничих сценаріїв.

### Література

- [1] A. Berekatain, H. Habibi, H. Voos, "A Practical Roadmap to Learning from Demonstration for Robotic Manipulators in Manufacturing", *Robotics*, №13(7). 100, 2024.  
DOI: 10.3390/robotics13070100
- [2] W. Li, Y. Wang, Y. Liang, & D. T. Pham, "Learning from demonstration for autonomous generation of robotic trajectory: Status quo and forward-looking overview", *Adv. Eng. Informatics*, №2024, 62.  
DOI: 10.1016/j.aei.2024.102625.
- [3] K. Wang, Y. Fan, I. Sakuma, "Robot Grasp Planning: A Learning from Demonstration-Based Approach", *Sensors*, №24(2), 618, 2024.  
DOI: 10.3390/s24020618
- [4] M. Bilal, N. Lipovetzky, D. Oetomo, & W. Johal, "Beyond success: quantifying demonstration quality in learning from demonstration", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Abu Dhabi, United Arab Emirates, 2024, pp. 5120-5127.  
DOI: 10.1109/IROS58592.2024.10802187
- [5] A. Sena, M. Howard, "Quantifying teaching behavior in robot learning from demonstration", *The International Journal of Robotics Research*, №39(1), pp. 54-72, 2019.  
DOI: 10.1177/0278364919884623
- [6] A. Calzada-Garcia, J. G. Victores, F. J. Naranjo-Campos, & C. A. Balaguer, "Review on inverse kinematics, control and planning for robotic manipulators with and without obstacles via deep neural networks", *Algorithms*, №18(1), 23, 2025.  
DOI: 10.3390/a18010023
- [7] M. G. Tamizi, M. Yaghoubi, H. Najjaran, "A review of recent trend in motion planning of industrial robots", *Int J Intell Robot Appl.*, № 7, pp. 253-274, 2023. DOI: 10.1007/s41315-023-00274-2
- [8] A. Yonezawa, H. Yonezawa, I. Kajiwara, "Simple inverse kinematics computation considering joint motion efficiency", *IEEE Transactions on Cybernetics*, vol. 54, no. 9, pp. 4903-4914, 2024.  
DOI: <https://doi.org/10.1109/TCYB.2024.3372989>
- [9] ISO 31000:2018. Risk management – Guidelines. International Organization for Standardization, 2018.
- [10] ISO/IEC/IEEE 24765:2017. Systems and software engineering – Vocabulary. International Organization for Standardization / International Electrotechnical Commission / Institute of Electrical and Electronics Engineers, 2017.
- [11] A. Avizienis, J.-C. Laprie, B. Randell, C. Landwehr, "Basic Concepts and Taxonomy of Dependable and Secure Computing", *IEEE Transactions on Dependable and Secure Computing*, vol. 1, no. 1, pp. 11-33, 2004.  
DOI: 10.1109/TDSC.2004.2.
- [12] S. R. Boyd, J. A. Moolman, N. J. Nwosu, *Risk Reporting & Key Risk Indicators: A Case Study Analysis. Enterprise Risk Management Initiative*, North Carolina State University, 2016. 34 p. [Online]. Available: [https://erm.ncsu.edu/wp-content/uploads/sites/436/migrated-files/ERM\\_KRI\\_Case\\_Study\\_FINAL.pdf](https://erm.ncsu.edu/wp-content/uploads/sites/436/migrated-files/ERM_KRI_Case_Study_FINAL.pdf)
- [13] P. Neto, N. Mendes, "Direct off-line robot programming via a common CAD package", *Robotics and Autonomous Systems*, vol. 61, no. 8, pp. 896-910, 2013. DOI: 10.1016/j.robot.2013.02.005
- [14] C. Lauretti, C. Tamantini, H. Tomè, L. Zollo, "Robot Learning by Demonstration with Dynamic Parameterization of the Orientation: An Application to Agricultural Activities", *Robotics*, vol. 12, no. 6. Art. 166, 2023.  
DOI: 10.3390/robotics12060166

UDC 681.5

**Y. M. Senchuk, F. D. Matiko***Lviv Polytechnic National University, Lviv, Ukraine***CLASSIFICATION OF RISKS AND FAILURE DETECTION INDICATORS IN LEARNING FROM DEMONSTRATION SYSTEMS FOR INDUSTRIAL ROBOT MANIPULATORS**

The article considers the process of forming motion programs for industrial robot manipulators in Learning from Demonstration (LfD) systems. It is shown that, although LfD approaches can reduce programming time and increase the flexibility of robotic technological processes, their practical use is complicated by the multi-stage conversion of operator demonstrations into an executable control program. During this process, errors may accumulate. They may remain undetected during local checks and appear at later stages of verification, program code generation, or direct execution of the manipulator motion. This creates a need to systematize risks not only for separate algorithmic procedures, but also for the entire sequence of motion program formation.

The authors identified 10 consecutive stages of motion program formation, from defining requirements and obtaining demonstrations to program code generation and approval of the program for execution on a real robot manipulator. This sequence includes processing demonstration data, representing the trajectory, performing kinematic and time-related transformations, checking constraints, simulation-based verification, and forming the control program. The sources of risk, potential failures, indicators of their detection, and possible forms of late manifestation of deviations were analyzed and systematized for each stage. Based on this analysis, a classification of typical risks, early detection indicators, and possible late manifestations of errors and failures was formed. A matrix of possible manifestations of deviations and failures, depending on the stage of their occurrence was developed. This matrix illustrates the possibility of inter-stage accumulation of errors during the development of a robot program. It also makes it possible to trace the relationship between the stage at which a deviation occurs and the stage at which its consequences may become noticeable. The obtained results can be used in the design of LfD systems, the development of offline verification procedures, and the creation of tools for monitoring the reliability and safety of robotic technological processes.

**Keywords:** robot manipulator, learning from demonstration (LfD), error, latent failure, control program, trajectory optimization, kinematic and dynamic constraints, classification of risks and failure indicators.

*Надійшла до редакції  
18 березня 2026 року*

*Рецензовано  
13 квітня 2026 року*



© 2026 Copyright for this paper by its authors.  
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).