МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Кваліфікаційна наукова
праця на правах рукопису

Долгоруков Сергій Олегович

УДК 004.9/.896]::[629.3.018.2:629.056.6-042.4](043.5)

ДИСЕРТАЦІЯ
АВТОМАТИЗОВАНЕ ПРОЕКТУВАННЯ ВИПРОБУВАЛЬНОГО СТЕНДУ НАВІГАЦІЙНОГО ОБЛАДНАННЯ

05.13.12 – Системи автоматизації проектувальних робіт

Подається на здобуття наукового ступеня кандидата технічних наук

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

Долгоруков С.О.

Науковий керівник Синєглазов Віктор Михайлович, доктор технічних наук, професор

Київ - 2021

АНОТАЦІЯ

Долгоруков С.О. Автоматизоване проектування випробувального стенду навігаційного обладнання. - Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.12 «Системи автоматизації проектувальних робіт». - Національний авіаційний університет, Київ, 2021.

Метою дисертаційного дослідження є підвищення ефективності випробувальних стендів (ВС), а так само його елементів, за рахунок побудови системи автоматизованого проектування (САПР) на основі використання інтелектуальних підходів, зокрема, навчання з підкріпленням.

Запропоновано багато-агентний підхід для автоматизації проектування ВС, який відрізняється від відомих тим, що використовує агентів-проектувальників з елементами штучного інтелекту (ШІ) - навчання з підкріпленням для вирішення завдання багатокритеріального прийняття рішень, в результаті забезпечуються покращення пошуку проектних рішень з точки зору використання попередніх знань. Підхід не тільки дозволяє легко включати нові функції нагород за необхідності, але і зважувати їх відносну важливість в залежності від потреб конкретного завдання проектування випробувального стенду. Розроблено метод побудови САПР ВС, який відрізняється від відомих тим, що забезпечує інтеграцію існуючих САПР в єдиний автономний комплекс, в якому людина-оператор визначає критерії та обмеження, в межах яких багато-агентній системі з підтримкою ШІ дозволяється управляти автоматизованими операціями, що призводить до економії часу проектування. Вперше запропоновано метод розв'язання задачі багатокритеріального прийняття рішень під час проектування, який при його імплементації використовує агенти з автономним навчанням, що самостійно збирають дані, генерують нові знання і використовують їх для коригування процесу прийняття рішень, що дозволяє безпосередньо оптимізувати цільові

параметри без необхідності визначати модель та форму апроксимації цих функцій.

Ключові слова: САПР, інженерне проектування, динамічні випробування, навігаційне обладнання, багато-агентні системи, оптимізація, обладнання для аерокосмічної галузі.

Список публікацій здобувача:

Статті у виданнях іноземних держав–членів ЄС:

1. S. O. Dolhorukov, “Navigation equipment test table computer-aided design artificial intelligence configuration optimization,” *Věda a perspektivy*, vol. 1, no. 1, pp. 214-227, May, 2021.

Статті у наукових фахових виданнях України:

2. V. M. Sineglazov and S. O. Dolgorukov, “Optimal choice of the technical means of rate, pitch and roll channels subsystems of navigation equipment simulation table,” *Electronics and control systems*, no. 2, pp. 33-39, 2014.
3. V. M. Sineglazov and S. O. Dolgorukov, “Identification of mechanical backlash of navigation equipment testing table,” *Electronics and control systems*, no. 3, pp. 104-108, 2014.
4. V. M. Sineglazov and S. O. Dolgorukov, “Complex criteria for navigation equipment test table optimal design problem,” *Electronics and control systems*, no. 1, pp. 87-92, 2015.
5. V. M. Sineglazov and S. O. Dolgorukov, “UAV equipment test table for preflight testing,” *Electronics and control systems*, no. 3, pp. 126-132, 2015.
6. V. M. Sineglazov and S. O. Dolgorukov, “Environmental Chamber for Navigation Equipment Test Table,” *Electronics and control systems*, no. 2, pp. 62-66, 2017.

Матеріали та тези наукових доповідей:

7. V. M. Sineglazov and S. O. Dolgorukov, “Computer-aided Design of Navigation Equipment Test Table based on Game Optimization,” *in Proc. IEEE 6th Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, (Kyiv, Ukraine), 2020, pp. 7-10, doi: 10.1109/MSNMC50359.2020.9255419.
8. V. M. Sineglazov and S. O. Dolgorukov, “Computer-Aided Design System of Navigation Equipment Test Table,” *in Proc. IEEE 5th Int. Conf. Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, (Kyiv, Ukraine), 2019, pp. 7-11, doi: 10.1109/APUAVD47061.2019.8943871.
9. V. M. Sineglazov and S. O. Dolgorukov, “Intelligent Integrated Cads of Navigation Equipment Testing System,” *in Proc. IEEE 5th Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, (Kyiv, Ukraine), 2018, pp. 7-10, doi: 10.1109/MSNMC.2018.8576314.
10. V. M. Sineglazov and S. O. Dolgorukov, “Dynamic testing of UAV navigation system,” *in Proc. IEEE 4th Int. Conf. Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, (Kyiv, Ukraine), 2017, pp. 211-214, doi: 10.1109/APUAVD.2017.8308812.
11. V. M. Sineglazov and S. O. Dolgorukov, “Multi objective optimization for UAV navigation equipment test table design,” *in Proc. IEEE 4th Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, (Kyiv, Ukraine), 2016, pp. 117-120, doi: 10.1109/MSNMC.2016.7783121.
12. V. M. Sineglazov and S. O. Dolgorukov, “Dynamic hardware-in-the-loop UAV ground testing system,” *in Proc. IEEE Int. Conf. Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, (Kyiv, Ukraine), 2015, pp. 91-94, doi: 10.1109/APUAVD.2015.7346569.
13. V. M. Sineglazov and S. O. Dolgorukov, “Test bench of UAV navigation equipment,” *in Proc. IEEE 3rd Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, Kyiv, Ukraine, 2014, pp. 108-111, doi: 10.1109/MSNMC.2014.6979743.

14. V. M. Sineglazov and S. O. Dolgorukov, “HardWare-In-The-Loop simulation table for UAV navigation complexes testing,” *in Proc. IEEE 2nd Int. Conf. Actual Problems of Unmanned Air Vehicles Developments Proceedings (APUAVD)*, (Kyiv, Ukraine), 2013, pp. 141-145, doi: 10.1109/APUAVD.2013.6705307.

ABSTRACT

Dolhorukov S.O. Computer-aided design of a navigation equipment test table. – Qualifying scientific work on the rights of the manuscript.

Ph.D. thesis in specialty 05.13.12 “Computer-aided design systems”. – National Aviation University, Kyiv, 2021.

The aim of the research is to improve the efficiency of navigation equipment test tables (NETT), as well as its elements, by building a computer-aided design system based on the use of intelligent approaches, in particular reinforcement learning.

A multi-agent approach for NETT design automation is proposed, which differs from the known ones in that it uses design agents with elements of artificial intelligence (AI) - reinforcement learning to solve a multi-criteria decision-making problem, resulting in improved design decision search in terms of using prior knowledge. The approach not only makes it easy to incorporate new features when needed, but also weighs their relative importance depending on the needs of the particular NETT design problem. A method for building an AI CAD system has been developed that differs from the known ones in that it integrates existing CAD systems into a single autonomous complex in which a human operator defines the criteria and constraints within which an AI-enabled multi-agent system is allowed to manage the automated operations, which leads to design time savings. For the first time, a method for solving the problem of multi-criteria design decision-making is proposed, which in its implementation uses agents with autonomous learning that independently collect data, generate new knowledge and use it to adjust the decision-making process, which allows direct optimization of target parameters without the need to define the model and form of approximation of these functions.

Keywords: CAD, engineering design, dynamic testing, navigation equipment, multi-agent systems, optimization, aerospace equipment.

ЗМІСТ

ВСТУП……10

РОЗДІЛ 1. ВИПРОБУВАЛЬНИЙ КОМПЛЕКС НАВІГАЦІЙНОГО ОБЛАДНАННЯ ЯК ОБ'ЄКТ АВТОМАТИЗОВАНОГО ПРОЕКТУВАННЯ……19

1.1. Навігаційне обладнання як об'єкт випробування…….19

1.2. Випробувальні комплекси навігаційного обладнання. Призначення. Класифікація. Виробники……..…30

1.3. Обґрунтування доцільності автоматизованого проектування випробувальних комплексів навігаційного обладнання……....37

1.4. Основні проблеми проектування випробувальних комплексів…41

1.5. Постановка завдання автоматизованого проектування випробувальних комплексів…….43

1.6. Застосування САПР для проектування випробувальних стендів…48

Висновки до розділу 1……...54

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ ДО РОЗДІЛУ 1……..55

РОЗДІЛ 2. РОЗРОБКА АВТОМАТИЗОВАНОЇ СИСТЕМИ ПРОЕКТУВАННЯ ВИПРОБУВАЛЬНИХ СТЕНДІВ……62

2.1. Автоматизоване проектування як засіб підвищення ефективності вирішення завдань проектування випробувальних стендів……..62

2.1.1. Загальні вимоги до конструкції випробувальних стендів……..66

2.1.2. Функціональне проектування випробувальних стендів….67

2.2. Постановка завдання мультидисциплінарного оптимального проектування випробувального стенду навігаційного обладнання……70

2.3. Огляд систем гібридного інтелектуального проектування……..74

2.4. Приклади діючих САПР. Програмний пакет SolidWorks……85

2.5. Багато-агентна система автоматизованого проектування випробувальних стендів навігаційного обладнання……103

Висновки до розділу 2……..…111

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ ДО РОЗДІЛУ 2……112

РОЗДІЛ 3. ІНТЕГРАЦІЯ КОМПЛЕКСУ ПРОГРАМНИХ БІБЛІОТЕК ТА САПР В АВТОМАТИЗОВАНИЙ ПРОЦЕС РОЗРОБКИ ВИПРОБУВАЛЬНИХ СТЕНДІВ ЗА ДОПОМОГОЮ ШТУЧНОГО ІНТЕЛЕКТУ…….118

3.1. Мета-евристичні методи багатокритеріальної оптимізації ……...118

3.2. Застосування інтелектуальних агентів для системи автоматизованого проектування випробувального стенда……122

3.3. Існуючі методології в області багато-агентного моделювання та алгоритми навчання з підкріпленням……136

3.4. Модель комплексу програмного середовища проектування. Інтеграція інтелектуальних агентів та САПР……...148

Висновки до розділу 3……..…158

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ ДО РОЗДІЛУ 3……....159

РОЗДІЛ 4. ОЦІНКА САПР НА ПРИКЛАДІ РЕАЛЬНОГО ЗАСТОСУВАННЯ ДЛЯ ПРОЕКТУВАННЯ ВИПРОБУВАЛЬНИХ СТЕНДІВ……....167

4.1. Приклад імплементації інтелектуального агента випробувального стенда……167

4.2. Приклад інтеграції САПР та моделі агента оптимізації параметрів механічної конструкції……180

4.3. Модернізація електричної частини випробувального стенду……186

4.4. Випробування дослідного зразка випробувального стенду навігаційного обладнання……...191

Висновки до розділу 4……195

ВИСНОВКИ…….196

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ ДО РОЗДІЛУ 4……199

Додаток 1. Технічні вимоги до випробувальних стендів……202

Додаток 2. Технічні характеристики випробувальних стендів……..205

Додаток 3. Акт про використання результатів дисертаційної роботи……...216

ДОДАТОК 4. СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ.....217

ВСТУП

Актуальність теми. На даний час велике значення має проблема підвищення якості навігаційного обладнання літальних апаратів, що досягається завдяки використанню випробувальних стендів. Один з підходів до підвищення продуктивності випробувальних стендів (ВС), до якого особливо прагнуть виробники, полягає в значному збільшенні динаміки руху ВС. В першу чергу це означає збільшення розгінної здатності, що дозволяє підвищити прискорення вздовж динамічної траєкторії, і лише в другу чергу - збільшення максимальної точності випробувань.

Збільшення максимального прискорення осі, а також досяжних швидкостей переміщення осей ВС може бути використано різними способами для підвищення продуктивності:

• Підвищення точності випробувань і можливостей ВС: Високоточна підтримка технологічно оптимальної швидкості випробувань уздовж запрограмованого просторового контуру за рахунок одночасного руху кількох осей вимагає їх постійного прискорення і уповільнення. Таким чином, збільшення прискорень осей є необхідною умовою для бажаного підвищення динамічної точності траєкторії і, отже, точності ВС, що виготовляється.

• Зниження виробничих витрат: Підвищена динаміка руху ВС дозволяє значно збільшити часові відрізки в послідовності випробувань з постійними, оптимальними технологічними умовами.

Однак для оцінки динамічних характеристик ВС можна використовувати не тільки дані про прогнозовані максимальних значеннях прискорень осей і швидкостей переміщення. Швидше, вони повинні розглядатися в поєднанні зі швидкістю зміни прискорення, яка може бути реалізована на ВС. Це пов'язано з тим, що дана змінна є вирішальною для того, які значення прискорення і швидкості руху можуть бути використані в реальній експлуатації ВС, а також на коротких ділянках прискорення.

Через фізичні причини вимога високої точності траєкторії при високодинамічних траверсних рухах являє собою класичний конфлікт цілей. Більш високі швидкості руху і, зокрема, більш високі прискорення осей, необхідні для цього, пов'язані з пропорційно вищими інерційними ефектами і потужностями приводів, що призводить до збільшення напруги і віброзбудження для елементів управління, приводів і рами. Таким чином, на додаток до статичної жорсткості і вібраційної поведінки конструкції ВС, динаміка руху пов'язаних осей ВС стає все більш важливою для потенціалу продуктивності високо динамічного ВС.

Основним завданням розробки є досягнення мінімально можливих рухомих мас в поєднанні з досить високою динамічною жорсткістю. Внаслідок цього зростають вимоги до відповідних навантажень конструкції механічної рами карданового підвісу і приводів ВС, які повинні бути реалізовані в процесі розробки.

Центральної мірою тут є інтегрована і синхронізована співпраця різних учасників виробничого процесу випробувальних стендів (механіка, електротехніка та електроніка, інформаційні технології) з самого початку розробки. Зростаюча технологічна складність і неоднорідність динамічних випробувальних стендів змушує приділяти більше уваги між-організаційному проектуванню в майбутніх процесах розробки такого обладнання.

Поведінку ВС повинно бути досліджено і оптимізовано не тільки на рівні окремих підсистем, на рівні ВС в цілому, а й на рівні взаємодії постачальників елементів всіх частин ВС - механіки, електротехніки та електроніки, інформаційних технологій.

Отже, при побудові даного виду систем вимагається застосування методів багатокритеріальної структурно-параметричної оптимізації з дотриманням набору технічних і економічних обмежень, що потребує більш детального вивчення, побудови моделей елементів ВС і розроблення системи автоматизованого проектування.

Зв'язок роботи з науковими програмами, планами, темами. Дисертаційна робота виконувалася у Національному авіаційному університеті за проведення науково-дослідних робіт з таких тем: «Ресурс» (реєстраційний номер 0101U006265), «Методологія автоматизованого проектування пілотажно-навігаційного комплексу гелікоптера» (реєстраційний номер 0102U000376), «Методики створення інтегрованої інформаційної системи літаків малої авіації на основі безплатформеної нанотехнологічної інерціальної навігаційної системи» (реєстраційний номер 0109U000689). «Комплексна робота зі створення дослідницького зразка безпілотної авіаційної системи «Україна» на базі двомоторного безпілотного повітряного судна» (реєстраційний номер 0101U006265).

Мета дослідження. Метою дисертаційного дослідження є підвищення ефективності проектування ВС, а так само його елементів, за рахунок побудови системи автоматизованого проектування на основі використання інтелектуальних підходів, зокрема, багатокритеріальних методів оптимізації.

Завдання дослідження. Основні завдання, які слід вирішити для досягнення мети роботи:

- визначити структуру системи автоматизованого проектування ВС;
- розробити інтелектуальний підхід для реалізації САПР на основі типової структури ВС, включаючи механіку, електротехніку та електроніку, програмне забезпечення;
- розробити метод проектування на базі навчання з підкріпленням для вирішення завдання багатокритеріального прийняття рішень
- розробити підсистему автоматизованого проектування інтелектуальних агентів проектування елементів ВС на базі штучного інтелекту (ШІ).
- розробити систему автоматизованого проектування ВС;

Об'єктом дослідження є система автоматизованого проектування ВС.

Предметом дослідження є методи, моделі і алгоритми, на основі яких відбувається розв'язується завдання побудови САПР ВС.

Методи дослідження базуються на положеннях:

- теорії оптимізації, дискретної математики і лінійної алгебри, що дозволило провести структурно-параметричний синтез;

- теорії мета-евристичних алгоритмів та навчання з підкріпленням, що дозволило провести пошук оптимальних проектних рішень в умовах невизначеності та обмежень;

- імітаційного та комп'ютерного моделювання, що дозволило підтвердити достовірність отриманих результатів, а так само виконати перевірку рішення під час вирішення завдання багатокритеріального прийняття рішень;

Як засіб розв'язання поставлених завдань використовувалося математичне та комп'ютерне моделювання. У дослідженнях використовувались програмні пакети Matlab, Simulink, а також мова програмування Python.

Наукова новизна одержаних результатів.

1. Запропоновано багато-агентний підхід для автоматизації проектування ВС, який відрізняється від відомих тим, що використовує агентів-проектувальників з елементами ШІ - навчання з підкріпленням для вирішення завдання багатокритеріального прийняття рішень, в результаті забезпечуються покращення пошуку проектних рішень з точки зору використання попередніх знань. Підхід не тільки дозволяє легко включати нові функції нагород за необхідності, але і зважувати їх відносну важливість в залежності від потреб конкретного завдання проектування випробувального стенду.

2. Вперше розроблена структура комплексу програмного середовища САПР, яка відрізняється від відомих тим, що поєднує взаємодію між ієрархічними рівнями інженерів-проектувальників та інтелектуальних агентів з власними моделями поведінки, що дозволяє скоротити час проектування та забезпечити взаємодію зацікавлених учасників процесу за межами організацій.

3. Розроблено метод побудови САПР ВС, який відрізняється від відомих тим, що забезпечує інтеграцію існуючих САПР в єдиний автономний комплекс, в якому людина-оператор визначає критерії та обмеження, в межах яких багато-агентній системі з підтримкою ШІ дозволяється управляти автоматизованими операціями, що призводить до економії часу проектування.

4. Вперше запропоновано метод розв'язання задачі багатокритеріального прийняття рішень під час проектування, який відрізняється від відомих тим, що при його реалізації використовуються агенти з автономним навчанням, що самостійно збирають дані, генерують нові знання і використовують їх для коригування процесу прийняття рішень, що дозволяє безпосередньо оптимізувати цільові параметри без необхідності визначати модель та форму апроксимації цих функцій.

Наукові результати і висновки, отримані у дисертації, підкріплено порівняльними дослідженнями результатів запропонованих методів з відомими.

Практичне значення одержаних результатів. На підставі розробленого математичного забезпечення:

1. Спроектована і побудована система випробування навігаційного обладнання, навчально-наукової лабораторії авіаційних комп’ютерно-інтегрованих комплексів, що дозволило перевірити ефективність системи автоматизованого проектування для побудови ВС;

2. На основі запрограмованої САПР розроблено, побудовано, налаштовано дослідну модель випробувального імітаційного динамічного трьох осевого стенду.

3. Результати дисертаційної роботи впроваджено у процес виробництва мікропроцесорних систем релейного захисту (МРЗС) державного підприємства «Виробниче об'єднання «Київприлад» у конструкторсько-технологічному відділі.

Особистий внесок здобувача. Всі результати, які відносяться до захисту, отримані автором особисто. Низку робіт надруковано у співавторстві.

У роботах [8,4] випробувальний стенд навігаційного обладнання розглядається як об'єкт автоматизованого проектування і визначаються основні завдання та проблеми проектування ВС. В [1] здобувачем виконана постановка задачі структурно-параметричного синтезу САПР ВС. Визначена типова структура ВС, включаючи механіку, електротехніку та електроніку і програмне забезпечення. Розроблено структуру системи автоматизованого проектування ВС з багато-агентною інтелектуальною підсистемою автоматизованого проектування. Стан технологій, методів і засобів розробки випробувальних стендів навігаційного обладнання розглянуто в [2,5,11,13,14]. Аналізується відмінність між загальними підходами та тими, які конкретно відносяться до розробки ВС. В роботах [7] та [9] обґрунтовано принцип автономного навчання агентів системи САПР проектування елементів ВС на базі ШІ та теорії ігор.

Апробація результатів дисертації. Основні теоретичні та методологічні положення та результати досліджень доповідалися і обговорювались на наступних міжнародних науково-технічних конференціях:

II міжнародна конференція IEEE International Conference Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD) (Kyiv, Ukraine, October 15-17, 2013); III міжнародна конференція IEEE International Conference on Methods and Systems of Navigation and Motion Control (MSNMC) (Kyiv, Ukraine, October 14-17, 2014); III міжнародна конференція IEEE International Conference Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD) (Kyiv, Ukraine, October 13-15, 2015); IV міжнародна конференція IEEE International Conference on Methods and Systems of Navigation and Motion Control (MSNMC) (Kyiv, Ukraine, October 18-20, 2016); IV міжнародна конференція IEEE International Conference Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD) (Kyiv, Ukraine, October 17-19, 2017); V міжнародна конференція IEEE International Conference on Methods and Systems of Navigation and Motion Control (MSNMC) (Kyiv, Ukraine, October 16-18, 2018); IV міжнародна конференція IEEE International Conference Actual Problems of

Unmanned Aerial Vehicles Developments (APUAVD) (Kyiv, Ukraine, October 22-24, 2019); VI міжнародна конференція IEEE International Conference on Methods and Systems of Navigation and Motion Control (MSNMC) (Kyiv, Ukraine, October 20-23, 2020).

Публікації. За результатами досліджень опубліковано 14 наукових праць, у тому числі 6 статей у наукових фахових виданнях (з них 1 стаття – у закордонному виданні ЕС та ОЕСР, 5 статей у виданнях України, які включені до міжнародних наукометричних баз), 8 тез доповідей у збірниках матеріалів конференцій (з них 8 включені до міжнародних наукометричних баз включаючи Scopus).

Структура та обсяг дисертації. Дисертаційна робота складається із анотації, вступу, чотирьох розділів, висновків, списків використаних джерел до кожного розділу, 3 додатків. Повний обсяг дисертації становить 218 сторінок, з яких 17 містять додатки. Загальний обсяг основної частини становить 200 сторінок друкованого тексту, включаючи 46 рисунків, 16 таблиць, 212 використаних джерел, які викладені на 24 сторінках.

Статті у виданнях іноземних держав–членів ЄС:

1. S. O. Dolhorukov, “Navigation equipment test table computer-aided design artificial intelligence configuration optimization,” *Věda a perspektivy*, vol. 1, no. 1, pp. 214-227, May, 2021.

Статті у наукових фахових виданнях України:

2. V. M. Sineglazov and S. O. Dolgorukov, “Optimal choice of the technical means of rate, pitch and roll channels subsystems of navigation equipment simulation table,” *Electronics and control systems*, no. 2, pp. 33-39, 2014.

3. V. M. Sineglazov and S. O. Dolgorukov, “Identification of mechanical backlash of navigation equipment testing table,” *Electronics and control systems*, no. 3, pp. 104-108, 2014.

4. V. M. Sineglazov and S. O. Dolgorukov, “Complex criteria for navigation equipment test table optimal design problem,” *Electronics and control systems*, no. 1, pp. 87-92, 2015.

5. V. M. Sineglazov and S. O. Dolgorukov, “UAV equipment test table for preflight testing,” *Electronics and control systems*, no. 3, pp. 126-132, 2015.

6. V. M. Sineglazov and S. O. Dolgorukov, “Environmental Chamber for Navigation Equipment Test Table,” *Electronics and control systems*, no. 2, pp. 62-66, 2017.

Матеріали та тези наукових доповідей:

7. V. M. Sineglazov and S. O. Dolgorukov, “Computer-aided Design of Navigation Equipment Test Table based on Game Optimization,” *in Proc. IEEE 6th Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, (Kyiv, Ukraine), 2020, pp. 7-10, doi: 10.1109/MSNMC50359.2020.9255419.

8. V. M. Sineglazov and S. O. Dolgorukov, “Computer-Aided Design System of Navigation Equipment Test Table,” *in Proc. IEEE 5th Int. Conf. Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, (Kyiv, Ukraine), 2019, pp. 7-11, doi: 10.1109/APUAVD47061.2019.8943871.

9. V. M. Sineglazov and S. O. Dolgorukov, “Intelligent Integrated Cads of Navigation Equipment Testing System,” *in Proc. IEEE 5th Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, (Kyiv, Ukraine), 2018, pp. 7-10, doi: 10.1109/MSNMC.2018.8576314.

10. V. M. Sineglazov and S. O. Dolgorukov, “Dynamic testing of UAV navigation system,” *in Proc. IEEE 4th Int. Conf. Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, (Kyiv, Ukraine), 2017, pp. 211-214, doi: 10.1109/APUAVD.2017.8308812.

11. V. M. Sineglazov and S. O. Dolgorukov, “Multi objective optimization for UAV navigation equipment test table design,” *in Proc. IEEE 4th Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, (Kyiv, Ukraine), 2016, pp. 117-120, doi: 10.1109/MSNMC.2016.7783121.

12. V. M. Sineglazov and S. O. Dolgorukov, “Dynamic hardware-in-the-loop UAV ground testing system,” *in Proc. IEEE Int. Conf. Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, (Kyiv, Ukraine), 2015, pp. 91-94, doi: 10.1109/APUAVD.2015.7346569.

13. V. M. Sineglazov and S. O. Dolgorukov, “Test bench of UAV navigation equipment,” *in Proc. IEEE 3rd Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, Kyiv, Ukraine, 2014, pp. 108-111, doi: 10.1109/MSNMC.2014.6979743.

14. V. M. Sineglazov and S. O. Dolgorukov, “HardWare-In-The-Loop simulation table for UAV navigation complexes testing,” *in Proc. IEEE 2nd Int. Conf. Actual Problems of Unmanned Air Vehicles Developments Proceedings (APUAVD)*, (Kyiv, Ukraine), 2013, pp. 141-145, doi: 10.1109/APUAVD.2013.6705307.

РОЗДІЛ 1.
ВИПРОБУВАЛЬНИЙ КОМПЛЕКС НАВІГАЦІЙНОГО ОБЛАДНАННЯ ЯК ОБ'ЄКТ АВТОМАТИЗОВАНОГО ПРОЕКТУВАННЯ

1.1. Навігаційне обладнання як об'єкт випробування

Навігаційне обладнання (НО) та гіроскопічні чутливі елементи (ЧЕ), що розробляються і виробляються для широкого кола застосувань, з метою забезпечення надзвичайно точної навігації та надійності, наприклад, на літальних апаратах (ЛА), безпілотних літальних апаратах (БПЛА), у керованих ракетах з малим часом польоту. Підвищити параметри точності та надійності навігаційного обладнання можливо за рахунок випробувань.

Незважаючи на те, що випробуванням ЧЕ і НО часто не приділяють належної уваги, вони мають велике значення із досить значними наслідками з точки зору вартості і продуктивності [1].

Середовище, в якому повинно працювати НО, сильно варіюється. Методи випробувань повинні відображати призначення НО, а також, середовище, в якому працюватимуть системи готового виробу.

Метою випробування ЧЕ є оцінка коефіцієнтів математичних моделей, щоб можна було передбачити роботу ЧЕ в конкретних умовах. Після визначення цих характеристик будь-які систематичні похибки можуть бути компенсовані, що підвищує його точність. Друга мета випробувань – це можливість калібрування вихідних сигналів ЧЕ і моделювання поведінки пристрою в різних ситуаціях і умовах.

Мета випробувань НО дещо відрізняється від цілей випробувань ЧЕ. Зазвичай, коли розробка НО досягає стадії лабораторних випробувань, характеристики ЧЕ компонентів вже добре відомі. Залежно від типу НО, може бути доцільно випробовувати в лабораторії або всю інерційну навігаційну систему, або тільки інерційний вимірювальний блок. Випробування проводяться з метою перевірки працездатності системи, для того щоб встановити, що вона працює так як передбачено на основі знань про характеристики ЧЕ, що входять до її складу. Іноді між цими компонентами

може виникати несприятлива взаємодія. Зазвичай такі випробування вимагають встановлення НО на випробувальному стенді (ВС). Потім НО згідно програми випробувань повертається на ряд точно відомих кутів, розташованих в різних орієнтаціях відносно локального вектора точки вимірювання. Потім домінуючі похибки ЧЕ можуть бути визначені на основі статичних вимірювань прискорення і швидкості повороту, отриманих при кожній орієнтації НО. Приклад такого підходу описаний в роботі [23].

Випробуваннями називають [18] експериментальне визначення кількісних та (або) якісних характеристик властивостей об'єкта випробувань як результату впливу на нього, під час його функціонування, під час моделювання об'єкта та (або) впливів.

Види випробувань НО та ЧЕ можна розділити на наступні категорії [2]:

1. Випробування характеристик і оціночні випробування НО, що знаходиться в стадії розробки. Перекваліфікаційні випробування для перевірки змін конструкції. Випробування для розробки або створення поліпшених методик випробувань і ВС.
2. Приймальні випробування, що вперше проводяться на серійних або відремонтованих ЧЕ, призначені для приймання в умовах, які, як очікується, будуть мати місце в передбачуваному реальному застосуванні.
3. Діагностичні випробування, що проводяться на НО, шо пройшло процедури приймальних випробувань, але згодом продемонструвало нетипові характеристики.
4. Калібрувальні випробування, що можуть проводитися під час приймальних випробувань та протягом всього терміну експлуатації ЧЕ на рівні НО, а також на рівні окремих ЧЕ.
5. Кваліфікаційні випробування, що проводяться на зразках, призначених для кваліфікації в якості задовільного виробу. Умови кваліфікаційних випробувань можуть бути більш жорсткими, ніж умови, що використовуються під час приймальних випробувань, оскільки метою кваліфікаційних випробувань є демонстрація проектного запасу

характеристик ЧЕ. ЧЕ, що пройшли кваліфікаційні випробування, як правило, не вважаються придатними для використання.

6. Випробування на надійність і випробування на ресурс, що проводяться для демонстрації рівнів надійності та середнього часу напрацювання НО до першої відмови. Прискорені випробування на надійність і довговічність можуть включати в себе умови, більш суворі, ніж умови, що зустрічаються при експлуатації.

Як правило, ЧЕ та НО проходять три види випробувань: кваліфікаційні, приймальні та випробування на надійність. Кваліфікаційні випробування є головними випробуваннями, які проводяться для ЧЕ та НО. Ці випробування зазвичай виконують перед виробництвом НО.

Випробування проводять за встановленими методиками згідно програми випробувань на ВС. Програма випробувань (ПВ) встановлює об'єкт та мету випробувань, види, послідовність і обсяг проведених експериментів, порядок, умови, місце і терміни проведення випробувань, забезпечення і звітність, а також відповідальність за забезпечення та проведення випробувань [18].

Завдання випробувань на стадії розробки, серійного виробництва та експлуатації, класифікація впливів та випробувань, програма випробувань, методика та способи випробувань детально описуються у [13].

Випробувальний стенд – засіб випробувань, що представляє собою технічний пристрій для відтворення умов випробувань НО, для встановлення НО в певних просторових положеннях, збору інформації про випробування й керування процесом випробувань згідно ПВ. Випробувальні стенди можуть випускатися серійно або створюватися в одиничному екземплярі.

Сучасні високоточні стенди - великогабаритне, енергоємне й дороге устаткування із системою безпосереднього керування двигуном, із забезпеченням стабільності кутової швидкості, яка задається, за рахунок великої інерційної маси платформи, використання повітряного підвісу її осі обертання й ртутних струмовідводів (для виключення сухого тертя між нерухомою основою й обертовою платформою) [22].

Для розробки теоретичних основ і правил конструювання ВС, призначених для випробування основних технічних характеристик широкого класу ЧЕ і НО, побудованого з їх застосуванням, з урахуванням тенденцій поліпшення технічних характеристик перспективного НО необхідно сформулювати основні вимоги до параметрів ВС. Це може бути виконано тільки на основі аналізу і оброблення інформації про досягнуті і очікувані значення основних характеристик гіроскопічних ЧЕ і НО.

Для отримання узагальненої оцінки характеристик точності гіроскопічних приладів з метою обґрунтування технічних вимог до ВС нижче аналізуються технічні характеристики кращих зразків ЧЕ та НО, що використовуються в авіаційних і ракетно-космічних системах інерційної навігації та орієнтації.

Досить докладно такий аналіз проведений в роботах [6, 7].

Вимірювачі кутової швидкості для безкарданних систем орієнтації і навігації включають: поплавкові датчики кутової швидкості (ДКШ), динамічно налаштовувані гіроскопи (ДНГ), волоконно-оптичні гіроскопи (ВОГ), лазерні гіроскопи (ЛГ), хвильові твердотільні гіроскопи (ХТГ), мікромеханічні гіроскопи (ММГ).

Поплавкові ДКШ за рівнем технічних характеристик можуть бути розділені на три основні групи: прецизійні, середньої точності і низькою точності, що використовуються в системах телеметрії [25-29].

Основною проблемою випробувань поплавкових ДКШ є завдання підтримки високостабільної кутової швидкості, постійної за величиною, навколо осі чутливості ДКШ у всьому робочому діапазоні кутових швидкостей для контролю коефіцієнта перетворення (статичної характеристики), а для телеметричних ДКШ - завдання навколо осі чутливості кутової швидкості, що змінюється за гармонійним законом, для контролю амплітудно-частотної характеристики та фазо-частотної характеристики - динамічних характеристик ДКШ.

ДНГ знайшли широке застосування в системах орієнтації і навігації для систем управління різними ЛА [6, 30-33, 7].

Фірма Litton розробила гіроскоп G-1200: для систем LN-30 (цивільні і військові літаки, ракети і крилаті ракети), для систем LN-35, LN-35M, LN-37A. Електромеханічні гіроскопи ДНГ застосовувалися в США з 1976 р, в той час, як в ті ж роки в СРСР застосовувалися в основному поплавкові ДКШ.

Крім фірми Litton розробкою ДНГ займаються фірми Zinger - Kirfortt, Kollins, Nortrop, Sazhem, Ferranty, Sperry, Sperry – Marcony.

В даний час ДНГ поступається за точністю прецизійним ДКШ. На думку ряду фахівців [6, 7], ДНГ для БІНС мають перспективу подальшого розвитку і застосування. Основною проблемою випробувань ДНГ є, як і для всіх вимірювачів кутових швидкостей, завдання стабільної, постійної за величиною кутової швидкості. Оскільки ДНГ має дві осі чутливості, то для забезпечення комплексності випробування доцільно, щоб ВС дозволяв при завданні кутових швидкостей встановлювати випробуваний ДНГ в різні кутові положення, тобто забезпечувати різну орієнтацію кожної осі чутливості щодо осі обертання стенду, при необертальній платформі це дозволяє контролювати всі складові дрейфу ДНГ.

ВОГ вважаються перспективними вимірювачами кутової швидкості для систем орієнтації і інерційної навігації [6, 7, 34-39].

У США та Німеччині розробка ВОГ ведеться з середини 70-х років XX століття. Найвідомішими фірмами є Sel, Teldix, IMAR, LITEF (Німеччина), Litton, Honeywell (США)

ЛГ з 1980 років набули широкого поширення в якості основних або допоміжних вимірювачів кутових швидкостей в навігаційних системах всіх видів ЛА [6, 40, 41, 7].

До переваг ЛГ відносяться простота конструкції, відсутність механічних обертових мас, малий час готовності, висока надійність. До недоліків ЛГ слід віднести істотний вплив вібрацій, температурних градієнтів і магнітних полів.

Лазерними гіроскопами займаються практично всі провідні аерокосмічні фірми США, Великобританії, Франції, Німеччини, Японії.

ХТГ у США виробляються фірмою DELCO. ХТГ працюють за умови інертності пружних хвиль [6, 35, 42-45]. Принцип дії ХТГ заснований на законі збереження моменту кількості руху матеріальним об'єктом. В якості такого об'єкта в даному випадку виступають матеріальні точки об'ємного резонатора, які вчиняють зворотно-поступальний рух в інерційному просторі завдяки згинальних коливань оболонки резонатора. Резонатор виконаний у вигляді півсфери, виготовленої з кварцового скла з золотим напиленням, в якому збуджена стояча хвиля другої моди згинальних коливань, тобто по периметру резонатора укладаються два повних періоди хвилі. При обертанні резонатора навколо осі його симетрії з деякою кутовою швидкістю на матеріальні точки, розташовані в області кучності коливань, будуть діяти сили Коріоліса в напрямку, перпендикулярному власному руху і вектору кутової швидкості, тобто по дотичній до краю резонатора. Ця сила прямо пропорційна швидкості власного руху і кутовій швидкості. В результаті будуть виникати коливання резонатора в ортогональній площині - в площині вузлів (при другій моді коливань ортогональна площина розташована під кутом 45 ° до площини початкових коливань). Зовні це буде проявлятися в зміщенні хвилі щодо резонатора (або корпусу, жорстко пов'язаного з ним) в сторону, протилежну зовнішньої кутової швидкості, зі швидкістю, пропорційної останній. З цього випливає, що кутове положення хвилі в системі координат, жорстко пов'язаної з корпусом, пропорційна інтегралу від проекції на вісь симетрії резонатора кутової швидкості руху корпуса в інерціальному просторі. Коливання резонатора порушуються системою силових електродів, вихідна інформація знімається за допомогою системи інформаційних електродів, кожен з яких утворює з поверхнею резонатора диференційний ємнісний датчик кута.

ММГ отримали розвиток в останні роки. Теорії і практиці розвитку ММГ присвячений цілий ряд робіт вчених [6, 35, 46-52]. ММГ мають вбудовану електроніку, що реалізує систему управління і обробки інформації. Вони

відрізняються малою масою, незначними габаритами і малим енергоспоживанням. В даний час ММГ мають низьку точність, неприйнятну для інерційних систем. Використання ММГ для вирішення навігаційних завдань можливо тільки при спільній роботі мікромеханічних гіроскопів і приймачів глобальних супутникових навігаційних систем.

Роторні вібраційні гіроскопи (РВГ) отримали розвиток в останні роки в зв'язку з необхідністю мініатюризації вимірювачів параметрів просторового руху об'єктів при збереженні достатньої точності. РВГ зручно використовувати на обертовому носії. Гіроскопи даного класу відрізняються простотою, високою надійністю і низькою вартістю. Дана особливість дозволяє відмовитися від роторного двигуна в приладі, що значно спрощує конструкцію гіроскопа, знижує його масу і енергоспоживання. РВГ є чутливий елемент в одноосьовому підвісі, площина якого перпендикулярна поздовжній осі обертання носія. При розворотах носія навколо осей, перпендикулярних осі його власного обертання, елемент гіроскопа здійснює гармонійні коливання в опорах підвісу з частотою обертання носія і амплітудою, яка пропорційна переносній кутовій швидкості [47] .

Стрижневі вібраційні гіроскопи (СВГ) призначені для роботи в умовах ударних впливів, що перевищують 20000 g. Розробка даного класу приладів отримала розвиток в останні роки в зв'язку з тим, що в сучасних системах випробування і управління в областях, де раніше застосування гіроскопічних технологій вважалося неможливим, все частіше і частіше потрібні вимірювачі кутової швидкості, здатні працювати в умовах високих перевантажень. СВГ затребувані у виробничій робототехніці, в областях техніки, де ударні операції є основними, як, наприклад, буріння свердловин, на високодинамічних об'єктах цивільного і військового призначення. Гіроскопічні ЧЕ дозволяють значно удосконалити системи управління об'єктів, що працюють в жорстких умовах ударних впливів, а тому до СВГ, крім точністних, пред'являються вимоги забезпечення працездатності в процесі і після закінчення ударних впливів. Вісь чутливості СВГ спрямована уздовж стрижня, який має високу

здатність навантаження. Амплітуда коливань стрижня по виходу осі пропорційна діючій кутовій швидкості. Гіроскопи даного класу відносяться до грубих вимірювачів кутової швидкості. Так як дія ударних навантажень в 20000 g і більше не може бути тривалою, то прилади даного класу повинні мати мінімальний час готовності [53].

ВС повинен мати технічні характеристики на порядок точніше, ніж у випробуваного приладу. Технічні вимоги до ВС, що необхідно забезпечити для випробування поплавкових ДКШ, ДНГ, ВОГ (прецизійних і середньої точності), ЛГ, ХТГ, ММГ, РВГ, СВГ наведені у Додатку 1.

Інтегруючі гіроскопи (ІГ) і одновісні гіростабілізатори (ОГС) на ІГ можуть бути використані для побудови блоків трьох і більше ОГС (БОГС), призначених для роботи в БІНС космічних апаратів [26-29, 54-65]. Перевагою ОГС є необмеженість кута розвороту навколо осі стабілізації. Однак такі системи не можна вважати перспективними з огляду на великі габарити і маси БОГС.

ВС, який міг би бути використаний для контролю кінематичної складової дрейфу ОГС, має мати 3 осі карданового підвісу, керовані таким чином, щоб імітувати рух космічного об'єкта по осях тангажу (Т), крену (К) і рискання (Р). При цьому рух по осі Т зазвичай являє собою обертання по орбіті з періодом 89 - 100 хв, на яке накладаються коливання. Рухи по осях К та Р носять коливальний характер. У загальному випадку частоти коливань по осях К, Р та Т різні.

Для випробувань ОГС, як чутливого елемента БОГС для безплатформових систем, існує проблема створення імітаційно-калібрувального ВС, що відтворює руху об'єкта управління.

При розробці сучасних приладів для БІНС, побудованих на вимірі кутової швидкості різних принципів дії, можливість проведення випробувань при установці приладу на ВС, що імітує рухи об'єкта, також має важливе значення.

Електростатичні вільні гіроскопи (ЕВГ) з порожнистим і суцільним ротором в даний час застосовуються в системах навігації - морських, авіаційних і космічних [6, 7, 8, 17]. При цьому ЕВГ, як правило, встановлюються на стабілізовану платформу, що забезпечує прийнятні умови, малі відхилення рухомої частини ЕВГ щодо нерухомого в просторі тіла - ротора ЕВГ, наслідком чого є мала величина і висока стабільність так званих «корпусних» моментів, які залежать від положення рухомої і нерухомої частин НО. Однак останнім часом ЕВГ стали застосовувати і в якості чутливих елементів БІНС.

До серйозних проблем, що обмежувались до теперішнього часу застосування ЕВГ в БІНС для космічних об'єктів, відносяться проблеми, пов'язані із забезпеченням вищевказаної малої величини дрейфу ЕВГ на рухомій платформі, що здійснює круговий рух по орбіті (навколо осі Т), з накладенням коливань по трьох осях Т, К і Р. у разі використання ЕВГ в безкарданній системі орієнтації на космічному об'єкті блок ЕВГ (БЕВГ) кріпиться безпосередньо на борту. Кути розвороту корпусу БЕВГ щодо нерухомого в просторі вектора кінетичного моменту ротора ЕВГ повністю відповідають кутам розвороту об'єкта в інерційному просторі.

Для експериментального дослідження дрейфу ЕВГ в умовах роботи ЕВГ на космічних апаратах доцільно створення імітаційно-калібрувального ВС, що дозволяє відтворювати в наземних умовах руху об'єкта по осях тангажу, крену та рискання.

Аналіз рівня технічних характеристик і проблем випробувань всіх вище перерахованих вимірювачів кутових швидкостей: ДКШ, ДНГ, ВОГ, ЛГ, ХТГ, ММГ, РВГ та СВГ, призначених для систем орієнтації і інерційної навігації, показує, що основна проблема, яка виникає при випробуваннях, загальна для всіх вимірювачів кутових швидкостей - це проблема створення ВС з однією віссю для задання постійної за величиною кутової швидкості навколо осі чутливості вимірювача.

З аналізу вимог до ВС видно, що необхідна розробка двох класів таких ВС; низької швидкості, з діапазоном кутових швидкостей 10 – 30 °/с і підвищеними вимогами до стабільності задання кутової швидкості і точності її виміру, і широкого діапазону швидкості, з діапазоном кутових швидкостей до 1000 °/с і менш жорсткими, але досить високими вимогами до стабільності задання кутової швидкості і точності її виміру. У табл. 1.1. наводяться узагальнені технічні вимоги до калібрувальних ВС, що вирішують завдання випробування всіх типів перспективних вимірювачів кутових швидкостей.

Таблиця 1.1. Узагальнені технічні вимоги до калібрувальних ВС

Технічні характеристики ВС	Для випробування вимірювачів кутових швидкостей		
	Низької швидкості		Широкого діапазону швидкості
	Прецизійних поплавкових ДНГ, ВОГ,ЛГ, ХТГ	Поплавкових ДКШ, ВОГ, середньої точності	Поплавкових ДКШ низької точності, широкодіапазонних ДНГ, ВОГ, ЛГ, ХТГ, ММГ
Число осей карданового підвісу ВС; Фіксовані кутові положення	1 2	1 -	1 2
Діапазон кутових швидкостей які задаються, °/с	10	30	≤1000
Мінімальна кутова швидкість, °/с	0,1	1	1
Дискретність кутової швидкості по осі підвісу ВС, кут. с./біт	0,04-0,08	0,62	0,2-0,6

Стабільність встановлюваної кутової швидкості, °/с	0,00001	0,00005-0,00001	0,001-0,005
Похибка вимірювання встановлюваної кутової швидкості, %	≤0,001	0,003	0,001
Діапазон фіксованих кутів розвороту навколо 2 і 3 осей карданового підвісу, град.	±180	±180	±180
Точність виставлення фіксованого кута розвороту, кут. с. Стабільність кутового положення 2 і 3 осей карданового підвісу, кут. с.	±30 ±1	±30 ±1	±30 ±1
Похибка задання синусоїдальних коливань За амплітудою, % Зв фазою, Гц	 0,1 0,1	 0,1 0,1	 0,1 0,1
Вантажопідйомність, кг	≤20		

Таким чином, можуть бути сформульовані наступні завдання створення ВС для випробування ЧЕ різного фізичного принципу дії та НО на їх основі для сучасних і перспективних навігаційних систем.

Необхідно створення:

1) одновісних калібрувальних ВС двох класів – низької швидкості і широкого діапазону швидкості, технічні характеристики яких відповідають табл. 1.1.

2) тривісних імітаційно-калібрувальних ВС для відтворення наземних умов рухів літальних апаратів навколо осей тангажу, крену та рискання.

1.2. Випробувальні комплекси навігаційного обладнання. Призначення. Класифікація. Виробники.

Для випробувань НО використовується спеціальний комплекс ВС, що знаходиться у випробувальній лабораторії. Випробувальна лабораторія повинна розташовуватися в максимально сейсмічно спокійному місці. Верхні поверхи будівель мають нестійке вібраційне середовище, ніж перший поверх, тому лабораторії для випробувань НО повинні розташовуватися на перших поверхах або в підвалі. Більш того, ВС в лабораторії повинні розташовуватися на опорах або майданчиках, ізольованих від фундаменту, щоб мінімізувати вплив при випробуванні високоточних ЧЕ. Залежно від місцевих геологічних умов масивний бетонний або кам'яний ізольований фундамент може спиратися на ґрунт або спускатися до корінних порід.

Найбільш розповсюдженим за використанням є наступний випробувальний комплекс НО:

- Оптичні ділильні головки.
- Випробувальні стенди:
    a. Одновісні ВС;
    b. Двовісні ВС;
    c. Тривісні ВС для випробування НО;
- Вібростенди та ударні стенди;
- Центрифуги;
- ВС імітації впливів середовища.

ЧЕ, наприклад акселерометр, встановлюється на *оптичну ділильну головку* (ОДГ) для проведення статичних випробувань (рис. 1.1.). Вісь чутливості (ВЧ) акселерометра зазвичай перпендикулярна горизонтальній осі обертання ділильної головки [2].

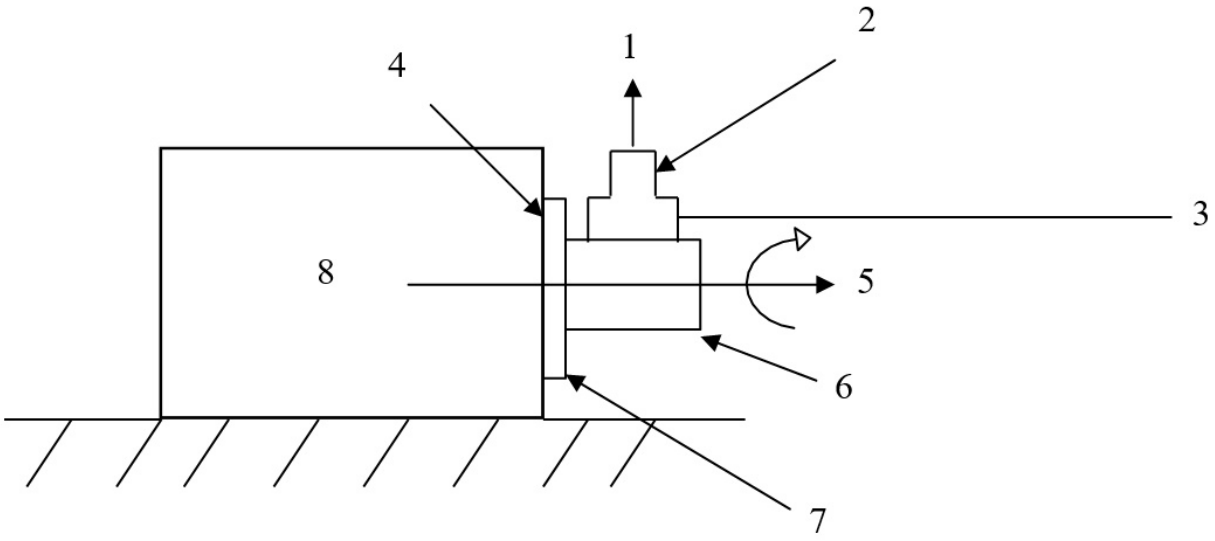


Рис. 1.1. Оптична ділильна головка для випробування акселерометра

1 - вісь чутливості акселерометра, 2 - акселерометр, 3 – кабельна лінія для збору інформації про випробування та для підключення систем живлення, 4 - шкала відліку кутів повороту, 5 - вісь обертання ОДГ, 6 – пристосування для встановлення ЧЕ, 7 – перехідна пластина для монтажу, 8 - електропривід повороту осі обертання ОДГ.

ОДГ дозволяє повертати акселерометр на різні кути навколо осі обертання з великою точністю. Точність кута повороту вказується в маркуванні ОДГ. Наприклад, ОДГ-10 означає, що ціна поділки кута повороту ОДГ становить 10 кутових секунд.

ОДГ можна дати команду управління на безперервне обертання для калібрування коефіцієнтів моделі чутливості кутової швидкості акселерометра і кутового прискорення, якщо ОДГ оснащена струмовідводами. Однак таке калібрування, краще проводити на одновісних випробувальних стендах і на рівні НО, а не на рівні ЧЕ.

Одновісний випробувальний стенд [2], зображений на рис. 1.2, має вертикально орієнтовану вісь обертання з кульковими або пневматичними підшипниками. ПВ такого ВС має певний діапазон кутових швидкостей та кутового прискорення відносно однієї осі для випробувань НО.

Одновісний випробувальний стенд (рис. 1.2) складається з: 1 - електропривід повороту осі обертання ВС, 2 - вісь чутливості гіроскопа, 3 – гіроскоп, 4 - пристосування для встановлення ЧЕ, 5 - шкала відліку кутів повороту та струмовідводи.

Орієнтація вертикальної осі ВС відносно напрямків схід-захід і північ-південь не має особливого значення, якщо одновісний ВС використовується тільки для вимірювань масштабного коефіцієнта гіроскопу з використанням повних обертів ВС.

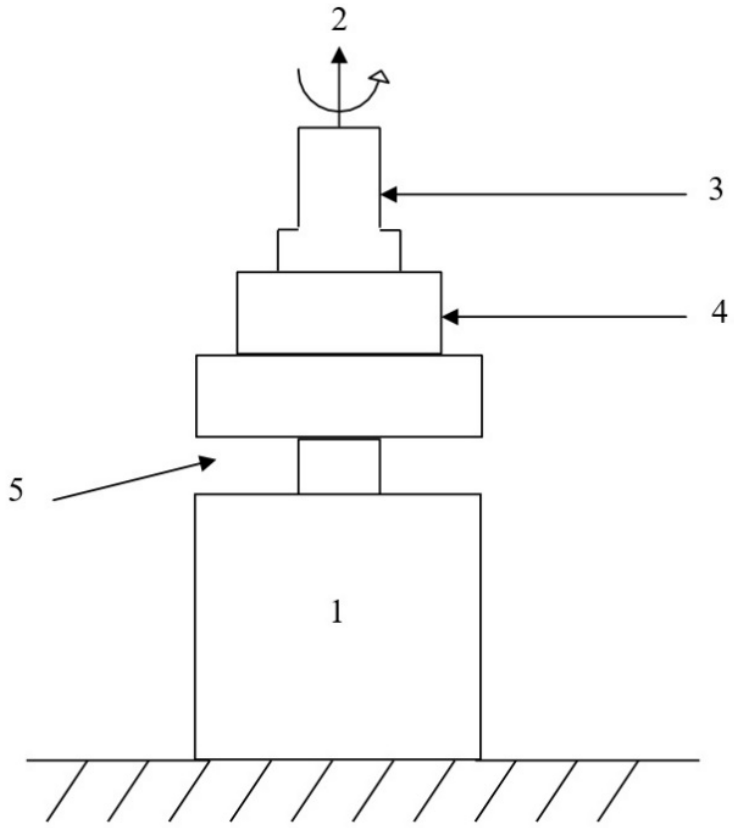


Рис. 1.2. Одновісний випробувальний стенд

Для вирівнювання осі ВС по вертикалі в межах заданих допусків можна використовувати прецизійні бульбашкові рівні, підкладаючи прокладки під основу ВС або використовуючи вирівнювальні домкрати. Реальні технічні дані розглянемо на прикладі ВС Acutronic та Ideal Aerosmith [5,19]. Характеристики ВС наведено в Додатку 2.

Двовісний випробувальний стенд [2] має горизонтальну цапфу, що обертається вручну, і внутрішню вісь з кульковими або пневматичними підшипниками, що має систему керування швидкості та положення (Рис. 1.3). В якості альтернативи, вісь цапфи може також мати систему керування швидкості та положення. містить горизонтальну вісь, що обертається автоматично або вручну та внутрішню вісь обертання з шарикопідшипниками або газодинамічними опорами. Лінії схід-захід і північ-південь можна визначити за допомогою теодоліту та за зірками. В якості альтернативи напрямок схід-захід можна визначити за допомогою гірокомпаса, в якому точна ВЧ гіроскопа почергово розташовується горизонтально на схід і захід.

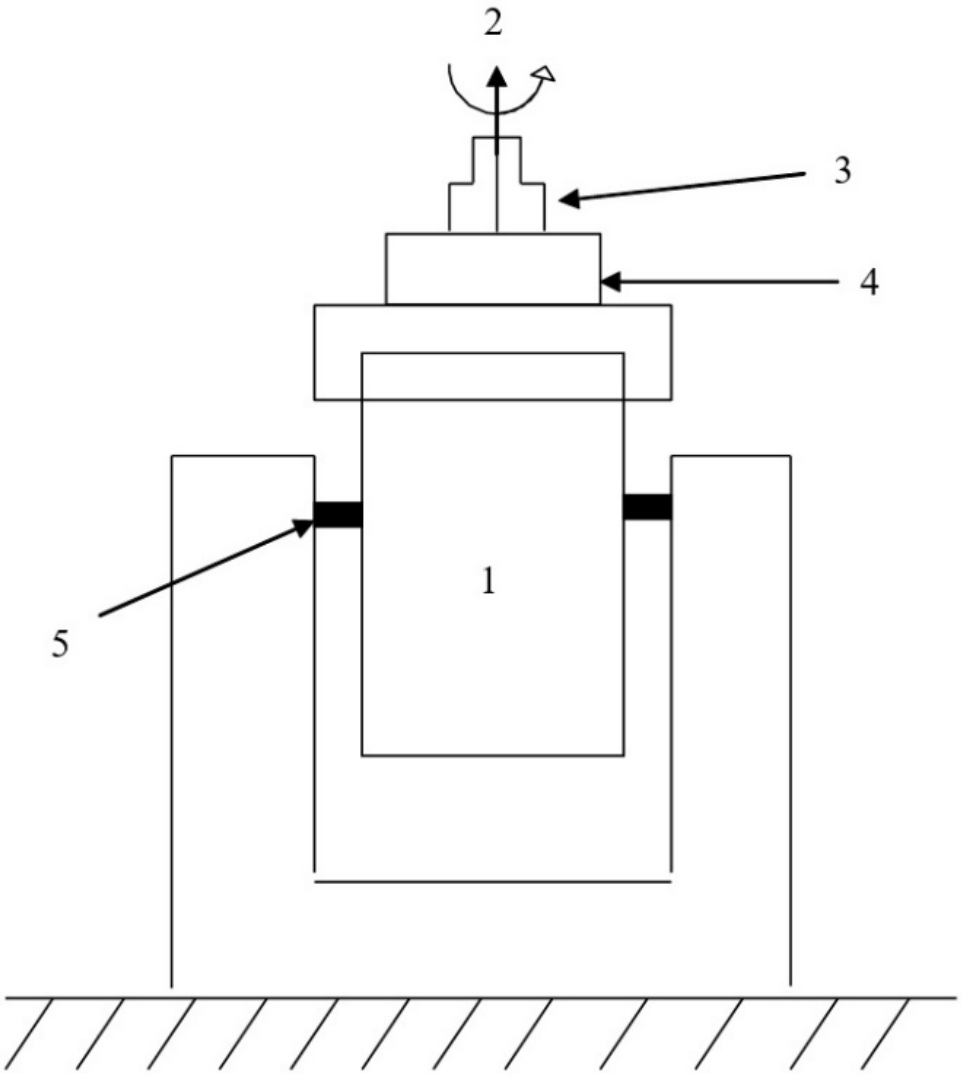


Рис. 1.3. Двовісний випробувальний стенд

ВС складається з: 1 - електропривід повороту осі обертання ВС, струмовідводи та шкала відліку кутів повороту, 2 - вісь чутливості гіроскопа, 3 – гіроскоп, 4 - пристосування для встановлення ЧЕ та перехідна пластина для монтажу, 5 – горизонтальна цапфа та шкала відліку кутів повороту.

При горизонтальному розташуванні осі цапфи зі сходу на захід внутрішня вісь кардана може повертатися паралельно і перпендикулярно осі обертання Землі. Кут між внутрішньою віссю кардана і горизонталлю, коли внутрішня вісь кардана паралельна вектору обертання Землі, є геодезичною широтою випробувального майданчика.

Характеристики ВС наведено в Додатку 2.

Тривісний випробувальний стенд [2] призначений для випробування НО, що містить три чи більше ЧЕ. Випробування НО може включати в себе розміщення його ВЧ під різними кутами відносно вектора сили тяжіння Землі і обертання НО навколо різних осей. Випробування НО часто може бути виконано на двовісному ВС, проте деякі траєкторії руху можуть потребувати використання тривісного ВС.

Тривісний ВС має три осі кардanового підвісу (рис. 1.4), яким можна дати команди керування повернути раму на задані кути або на обертання із заданою швидкістю. Можлива початкова орієнтація ВС з кутами повороту осей кардanового підвісу 0º, 0º, 0º: середня вісь орієнтована горизонтально та спрямована на схід-захід, а внутрішня вісь орієнтована горизонтально та спрямована на північ-південь. Послідовність обертання осей кардanового підвісу різних моделей ВС або їх розміщення може відрізнятись від зображеного на рис. 1.4.

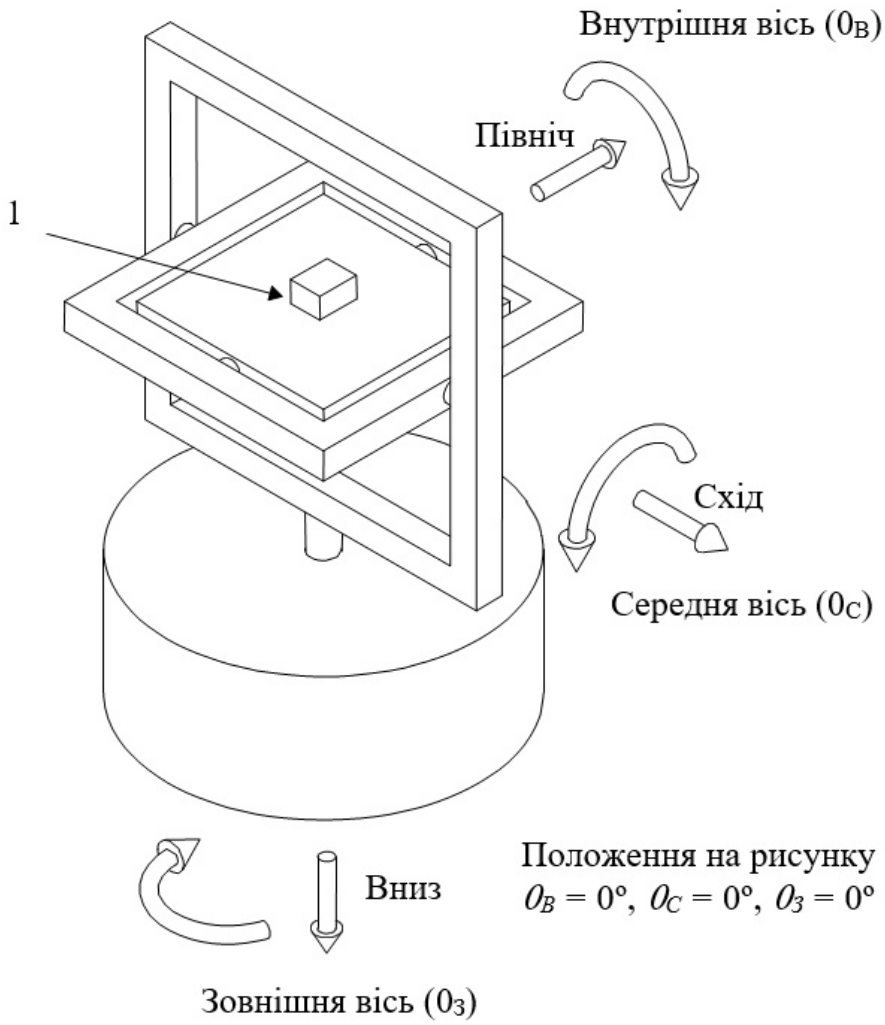


Рис. 1.4. Тривісний випробувальний стенд. 1 – НО.

Характеристики ВС наведено в Додатку 2.

НО встановлюється *на вібраційний або ударний ВС* для проведення різних лабораторних випробувань. Частотна розгортка вібростенда, наприклад, від 20 Гц до 2000 Гц, дозволяє знайти будь-які структурні резонанси в ЧЕ, які будуть виявлені за зсувами вихідного сигналу ЧЕ на частотах вібрації, де є резонанси. Зсуви і перехідні процеси аналізуються в вихідному сигналі ЧЕ під час випробувань збурень вібрації і ударів, що зустрічаються в реальних умовах експлуатації, також перевіряється спроможність ЧЕ компенсувати вібраційні перенавантаження.

Синусоїдальні коливання на різних амплітудах і вздовж різних осей ЧЕ використовуються для калібрування нелінійностей моделі ЧЕ.

Характеристики ВС наведено в Додатку 2.

НО або ЧЕ випробовується на *центрифугах* для визначення його чутливості до постійного прискорення високого значення. Центрифуга зазвичай обертається в горизонтальній площині навколо вертикальної осі обертання таким чином, що НО отримує прискорення $R\omega^2$ уздовж довжини центрифуги і локальне значення сили тяжіння в вертикальному напрямку вгору, де R - радіус обертання і ω - кутова швидкість обертання центрифуги (рис. 1.5).

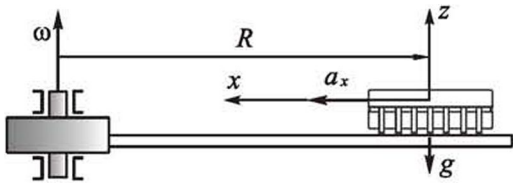


Рис. 1.5. Випробування НО на центрифугах

Якщо кутова швидкість центрифуги перевищує можливості ЧЕ за кутовою швидкістю, то або ЧЕ повинен бути розташований перпендикулярно осі обертання центрифуги, або ЧЕ повинен бути встановлений на столі, що обертається в протилежному напрямі на кінці важеля центрифуги. Віброфуги призначені для комплексного випробування, що складається з вібрації і постійного прискорення. Це поєднання центрифуги і вібростенда.

Характеристики ВС наведено в Додатку 2.

ВС імітації впливів середовища дозволяють проводити випробування в умовах впливу температури, тиску, вологості, сонячної радіації, піску і пилу, соляного туману, грибка і т.д. [20, 70]

Характеристики ВС наведено в Додатку 2.

Вимірювання магнітної чутливості НО включають в себе розміщення котушки Гельмгольця навколо обладнання, що випробовується. Кільця Гельмгольця мають циліндричну симетрію для створення рівномірного магнітного поля всередині циліндра. Структурна підтримка для котушок забезпечується дерев'яним або іншим подібним немагнітним каркасом.

Коли НО не знаходиться в котушці, струм пропускається через котушку, і результуюча сила магнітної індукції всередині котушки вимірюється за допомогою вимірювача Гауса (або магнітометра для низьких магнітних полів). Потім НО встановлюється в котушку, і потрібне магнітне поле створюється шляхом пропускання відповідного струму через котушку [15, 16].

Один з видів випробувань який в даний час дуже популярний серед розробників ЛА та БПЛА, називається *напівнатурними випробуваннями*. Така форма випробувань часто використовується на етапі розробки, щоб, наприклад, встановити точну оцінку характеристик БПЛА [66]. Іноді ЧЕ, але зазвичай НО, встановлюється на випробувальний стенд. Як правило це трьох осевий випробувальний стенд, як показано на рис. 1.4, який може точно відтворювати кутові рухи, які БПЛА буде мати під час експлуатації, наприклад, під час горизонтального польоту, що часто обмежується кутовими обертаннями навколо відповідних осей. Вихідні сигнали від випробовуваного НО підключаються через відповідний інтерфейс до ПК керування випробуваннями, що моделює роботу БПЛА. Ця математична модель в ПК також генерує сигнали, які керують випробувальним стендом, тому випробування в реальному часі, використовуючи реальне обладнання, може бути налаштоване таким чином, щоб забезпечити реалістичні оцінки ефективності для складних систем, що працюють в різних режимах польоту [68].

1.3. Обґрунтування доцільності автоматизованого проектування випробувальних комплексів навігаційного обладнання

Процес інженерного проектування зазвичай передбачає розробку технічного завдання об’єкту, яке задовольняє функціональним специфікаціям, що відповідає певним вимогам до робочих параметрів та ресурсним обмеженням, завдання має реалізовуватися в цільовій технології, що задовольняє одному або декільком критеріям технологічності: надійності,

безпеки і, останнім часом, обмежень, пов'язаних з захистом навколишнього середовища. Поширене тлумачення інженерного дизайну часто сприймається як перетворення потреби - вираженої в абстрактній концепції з точки зору загальної функціональності - в продукт, що задовольняє цю потребу, і його процес, що включає відображення певної функції на реалізовану фізичну структуру - спроектований продукт.

Процес інженерного проектування є комплексним і насамперед вимагає від інженера-проектувальника проявити ініціативу і винахідливість, а також використовувати широкий спектр навичок і досвіду для досягнення рішення. У міждисциплінарному середовищі проектування проектувальник також часто повинен бути професіоналом загального профілю, орієнтуючись на можливості використання широкого спектру технологій. Зростаюче число випадків розробки міждисциплінарних продуктів не тільки усунуло багато з традиційних обмежень, пов'язаних з проектуванням, а й надало проектувальнику набагато ширшу свободу вибору найкращого рішення конкретної проектної задачі.

В даний час велика частина інженерного проектування здійснюється окремими особами або групами осіб, які пройшли підготовку за однією інженерною дисципліною з обмеженими знаннями або досвідом інших. Ці люди "прив'язуються" до традиційних і знайомих технологій, які можуть привести до менш ніж оптимального рішення їх проектної задачі. Існує два способи міждисциплінарного проектування виробу: (1) навчання всіх проектувальників, щоб вони володіли загальними знаннями багатьох технологій, і (2) спільне розміщення фахівців з різних технологій з метою заохочення комунікації і співпраці в проектній групі. Однак, у зв'язку з прогресом, досягнутим в області автоматизованого проектування, і впровадженням таких засобів штучного інтелекту (ШІ), як функціональне мислення в традиційні інструменти комп'ютерного проектування для обробки обмежень і планування, неважко побачити, як може виникнути третя альтернатива вищевказаним підходам - ***розробка автоматизованої***

комп'ютерної системи для підтримки та керівництва проектувальником через сферу доступних технологічних можливостей, за допомогою відповідних інструментів для виконання обробки проектних даних на належному рівні для побудови проектного рішення, розробки концепції, вибору і узгодження технології, виконання моделі, оптимізації та візуалізації.

Сучасне промислове середовище характеризується зростаючими вимогами до гнучкості виробів, швидкої і надійної поставки замовлень, а також зменшеним життєвим циклом продукції. Зіткнувшись з цими вимогами, компанії по всьому світу досліджують, оцінюють і модернізують свою діяльність з метою ***усунення непродуктивної роботи, раціоналізації основних процесів і підвищення продуктивності*** за рахунок використання засобів автоматизації і інформаційних технологій.

Ефективне управління інформацією є ключовим компонентом цієї діяльності. В майбутньому компанії не зможуть вижити тільки завдяки чудовому логістичному управлінню фізичним матеріальним потоком, що проходить через різні види діяльності; вони також повинні перевершувати логістичне управління різними видами управління інформаційними потоками. Ключовим завданням логістичного управління інформацією є повторне використання наявної інформації: в ідеалі вся створена або вивчена інформація повинна бути доступна для подальшого використання правильним, корисним і своєчасним чином.

Термін служби виробу скорочується; продукт, який раніше залишався на ринку протягом десяти років, тепер повинен бути замінений після закінчення п'яти років або навіть менше. У той же час збільшується кількість варіантів продукції ; все більше і більше клієнтів вимагають виробів, розроблених з урахуванням їх конкретних потреб. Продукція також більш складна, ніж раніше: підвищені вимоги до функціональності, якості, довговічності та екологічної безпеки вимагають більш складних і інтегрованих конструкцій виробів, в яких потрібно кілька видів інженерної експертизи. І останнє, але не

менш важливе: клієнти вимагають, щоб поставка продукції відбувалася як можна швидше.

Таким чином ми приходимо до висновку, що ***традиційні комп'ютерні технології не підходять для вирішення проблеми великих наукомістких комплексних багатоетапних проектів***. Такі проекти мають спільні риси: (i) вони є великими; (ii) вони тривають довгий час; (iii) вони вимагають великої кількості співпрацюючих людей; (iv) вони вимагають специфічних знань; (v) вони спираються на попередні рішення.

Для того щоб вирішувати завдання в таких проектах, потрібно розробити кращі умови проектування для спрощення завдань проектувальників.

Проектування ***випробувальних стендів*** передбачає співпрацю ряду фахівців. Це робиться в кілька етапів. Перший етап носить інноваційний характер і полягає в визначенні базового скелета випробувального стенда. Інша робота носить в основному рутинний характер. Дійсно, весь процес спрямований на пошук компромісу між різними проектними рішеннями: наприклад, чим менше точне кутове положення потрібне, тим більше проектних можливостей є, в цьому випадку ціна випробувального стенда буде низькою.

Проектування випробувального стенда рівнозначно пошуку якогось вихідного рішення і перегляду його знову і знову з метою поліпшення різних критеріїв, встановлених різними групами проектувальників.

На етапі ескізного проекту, протягом декількох місяців, необхідно розглянути ряд можливих проектних рішень (до 50), щоб вибрати 2 або 3 варіанти, які приведуть до детального технічного проектування, яке, в свою чергу, може тривати до десяти років.

Середовище проектування прототипів випробувальних стендів зазвичай інтегрує ряд незалежних програмних продуктів, кожен з яких алгоритмічно вирішує різне завдання (оцінка моделі помилки, прогнозування ціни компонентів , компоновка, внутрішні та зовнішні конструкції). Різні програмні модулі для створення, аналізу і обробки ряду альтернативних

конструкцій на різних рівнях деталізації надають всім задіяним фахівцям (фахівцю, головному інженеру, інженеру-електрику, економісту і т.д.) можливість додати свої особисті погляди, пов'язані з відповідними даними.

Попереднє проектування тривісного прецизійного випробувального стенда [10] *зайняло близько 6 місяців*, в результаті чого було складено більше *12 окремих технічних звітів*.

Останні досягнення в області штучного інтелекту і автоматизації виробництва відкривають нові можливості. Таким чином**,** *сучасні засоби САПР повинні бути інтегровані в процес проектування випробувального стенду.*

1.4. Основні проблеми проектування випробувальних комплексів

Випробувальні стенди навігаційного обладнання зазвичай складаються з трьох частин:

1) виконавчої або імітаційної, в яку входять об'єкт випробування – навігаційне обладнання і системи, що забезпечують вплив різних експлуатаційних факторів – термокамера, ***система приводів, кардановий підвіс та опорні підшипники*** ;

2) інформаційно-керуючої, в яку входять системи управління та вимірювальні системи, що інформують про параметри об'єкта випробування і стенду – ***система кодування положення***, ***контрольно-вимірювальні прилади та система управління***, ***передача сигналу в обладнання для випробувань***;

3) систем енергоживлення.

Проектування випробувальних стендів - складний процес, що включає в себе розробку технічного завдання, проектних пропозицій, ескізного і технічного проектів, розробку функціональних і принципових схем, розрахунок елементів і систем стенду, вибір конфігурації стенда, що

забезпечує задані проектні параметри, і остаточну розробку робочої документації на стенд.

На рис. 1.6 показана блок-схема для випробувального комплексу навігаційного обладнання Національного Авіаційного Університету [67, 69].

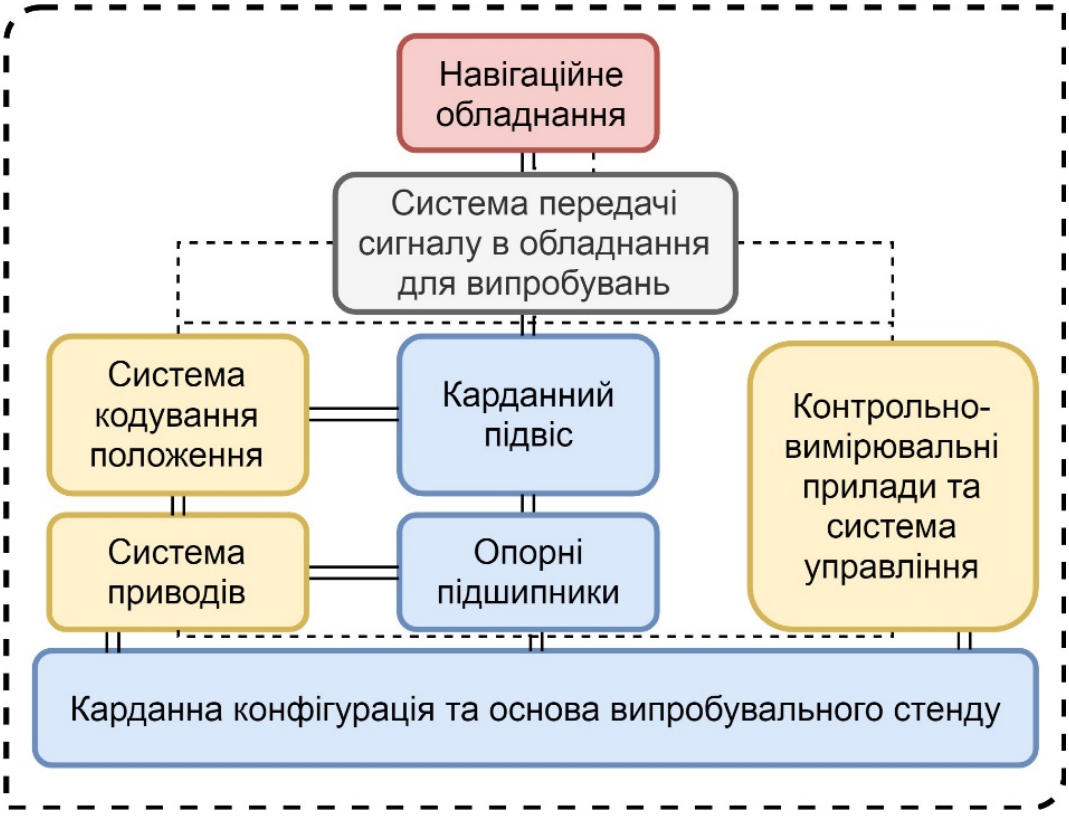


Рис. 1.6. Блок-схема випробувального стенду

Існують різні підходи до проектування випробувальних стендів. При індивідуальному підході вирішуються в основному окремі завдання створення стенду, які виникають на тому чи іншому підприємстві в зв'язку з необхідністю контролю певних параметрів навігаційного обладнання або його елемента.

При комплексному системному підході питання проектування стендів вирішуються виходячи з комплексних завдань забезпечення заданого рівня якості навігаційного обладнання. У цьому випадку передбачається створення системи взаємопов'язаних стендів, необхідних для відпрацювання, випробування і контролю систем створюваного навігаційного обладнання на всіх стадіях його життєвого циклу: проектування, виробництва і експлуатації.

До ***головних проблем в області проектування випробувальних стендів*** можна віднести:

• Створення на базі теорії фізичного моделювання та теорії подібності імітаторів експлуатаційних навантажень, що ***максимально наближають умови наземних випробувань виробів до експлуатаційних***. Вирішення цієї проблеми спрямоване на підвищення ефективності експериментального відпрацювання.

• ***Підвищення точності відтворення*** експлуатаційних факторів і ***точності вимірювань*** контрольованих параметрів випробуваного навігаційного обладнання. Вирішення цієї проблеми спрямоване на підвищення точності бортових систем БПЛА, яка є одним з основних критеріїв, що визначають їх якість.

• ***Автоматизація проектування стендів*** на базі конструкторського досвіду і математичного моделювання з використанням ПК для вибору оптимальних компонентів стендів. ***Вирішення цієї проблеми спрямоване на підвищення якості проектів і скорочення часу виготовлення стенду***.

1.5. Постановка завдання автоматизованого проектування випробувальних комплексів

Завдання проектування стенду полягає в тому, щоб в будь-яких умовах застосування виявити властивості випробувального стенду, які забезпечували б його максимальну ефективність.

Формалізацію задачі оптимізації проектних параметрів випробувального стенду при методах проектування за допомогою САПР можна представити у вигляді:

$$x = \begin{matrix} \arg \ \ extr\,\{C, M, A, T, R\} \\ x \in \Omega, \ \ y \end{matrix}$$

x = opt, C = min; M = max; A = min; T = max; R = max,

де x – проектні параметри; $\{C, M, A, T, R\}$ критерії оптимізації – вартість; маса випробуваного обладнання; граничні відхилення параметрів від заданих; довговічність; надійність, та відповідні обмеження критеріїв $y = \{y_{Cmin} \leq y_C \leq y_{Cmax}, y_{Mmin} \leq y_M \leq y_{Mmax}, y_{Amin} \leq y_A \leq y_{Amax}, y_{Tmin} \leq y_T \leq y_{Tmax}, y_{Rmin} \leq y_R \leq y_{Rmax}\}$,

та $\Omega = \{x|x_i \geq 0, x|x_{imin} \leq x_i \leq x_{imax}, i \in [1, n]\}$, x – вектор проектних параметрів та обмежень, представлений набором n елементів.

Системний підхід до проблем проектування отримав розвиток в організації проектування і науково обґрунтованому виробленні рішень, а також в їх взаємодії з метою досягнення максимального ефекту. ***Організація проектування полягає в створенні моделі процесу проектування, заснованої на концепції управління***. Проектні процедури поділяються на процедури аналізу і синтезу (Рис. 1.7). Синтез полягає в створенні опису випробувального стенду, а аналіз - у визначенні властивостей і дослідженні працездатності стенду за його описом, тобто під час синтезу створюються, а при аналізі оцінюються проекти випробувальних стендів.

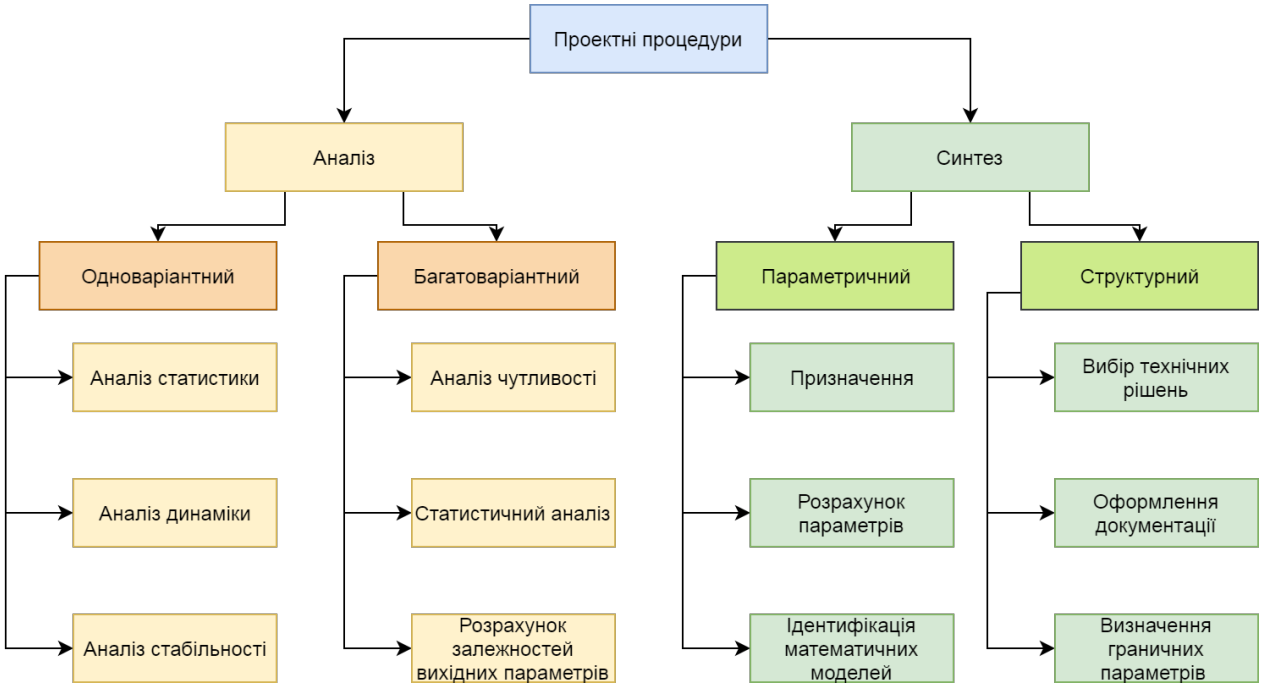


Рис. 1.7. Блок-схема проектних процедур

Процедури аналізу можуть бути одноваріантні і багатоваріантні. Одноваріантний аналіз передбачає встановлення значень внутрішніх і зовнішніх параметрів і визначення значень вихідних параметрів об'єкта. Завдання аналізу з одним варіантом зводиться до одноразового рішення рівнянь, що складають математичну модель. Багатоваріантний аналіз полягає в дослідженні властивостей ВС в певних межах простору його внутрішніх параметрів та вимагає багаторазового вирішення систем рівнянь.

Процедури синтезу - параметричні та структурні. Метою структурного синтезу є визначення структури ВС та переліку типів елементів, що становлять ВС, і способу зв'язку елементів підсистем між собою в складі ВС. Метою параметричного синтезу є визначення числових значень параметрів підсистем ВС при заданих значеннях структури і умови працездатності. На рис. 1.8 представлена послідовність етапів проектування.

Проектування ВС починається з синтезу початкового варіанту його структури.

Для оцінки цього варіанту створюється математична модель, динамічно подібна модель, фізична модель, стенд. Після вибору вихідних значень параметрів підсистем виконується аналіз варіантів. Потім виконується оцінка, що полягає в перевірці виконання працездатності ВС. Якщо результат перевірки незадовільний, то вибирається один з можливих шляхів поліпшення проекту. Найчастіше змінюють числові значення параметрів елементів. Сукупність процедур модифікації параметрів, аналіз і оцінка результатів аналізу становить *параметричний синтез*. Якщо шляхом параметричного синтезу не досягаються необхідні умови працездатності, то модифікують структуру ВС. Створюється новий варіант структури, для якого повторюються кроки побудови моделі і параметричного синтезу. У виняткових випадках коригується ТЗ, що обумовлює ітераційний характер процесу проектування.

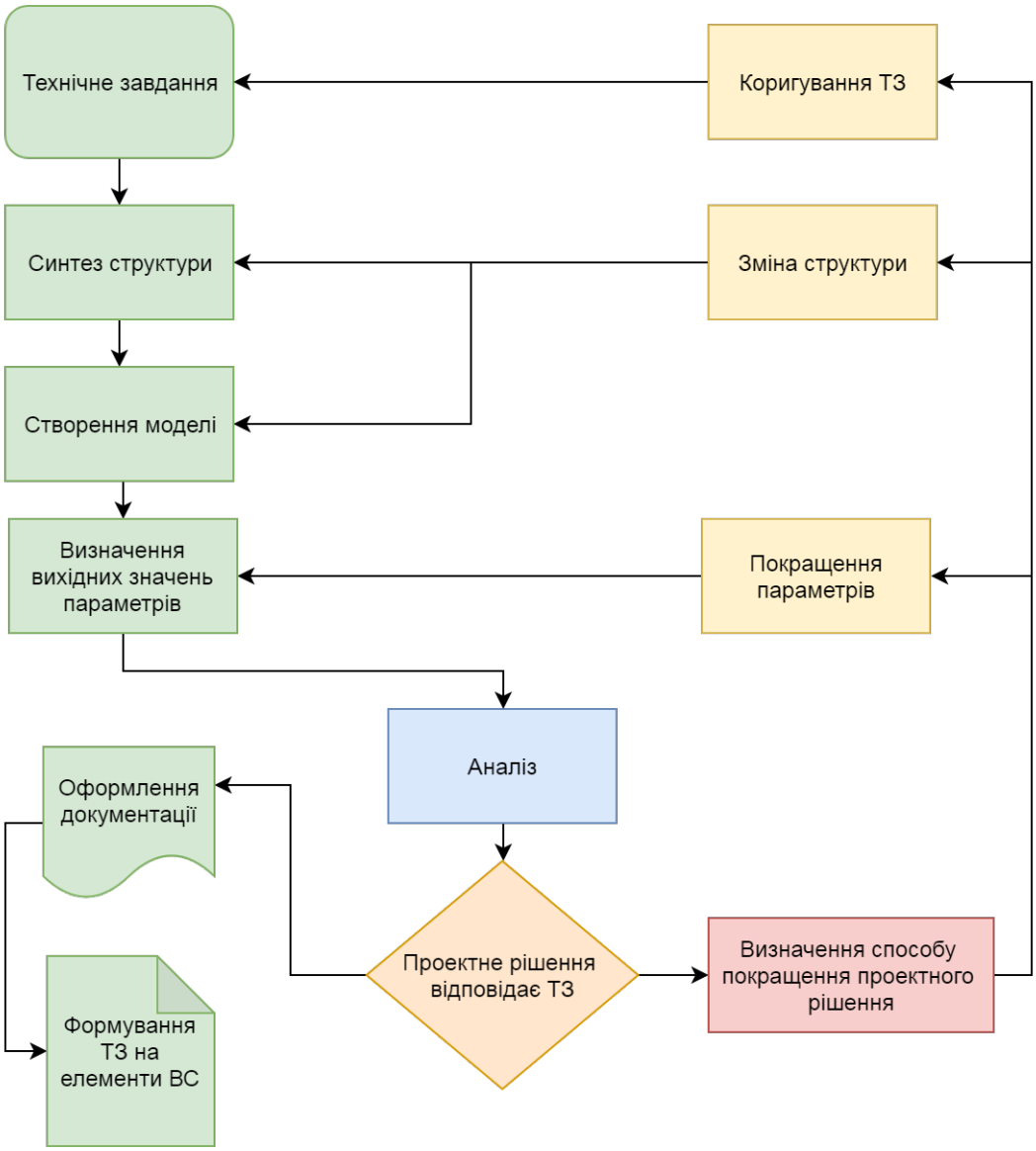


Рис. 1.8. Схема процесу проектування випробувального стенда

Приклад ТЗ для початку процесу проектування випробувального стенда наведено далі.

Стенд повинен виконувати завдання випробувань навігаційного обладнання:

1. Стенд повинен бути розрахований на випробування обладнання вагою P, яке займає обсяг V і вимагає для підключення N електричних з'єднань.

2. Стенд повинен забезпечувати максимальну точність випробувань, тобто його загальна сумарна похибка по всіх осях не повинна перевищувати A кутових секунд вимог випробування обладнання для якого він проектується.

3. Стенд повинен обертати обладнання, що випробовується навколо взаємно перпендикулярних осей *Z*. Кількість осей *Z* задається завданням випробувань обладнання, для якого він проектується. Відлік осей карданового підвісу стенду йде із зовнішнього осі до внутрішньої, зовнішня вісь лежить в горизонтальній площині.

Вихідні параметри проектування:

P = 50 кг;

V= 1м^3 (1м x 1м x 1м);

N=4;

A=100;

Z=3.

Різні підсистеми стенду, що використовуються для оцінки та аналізу під час проектування визначені в Таблиці 1.2.

Таблиця 1.2. Проектні рішення окремих підсистем завдання проектування стенду

Підсистема	Проектні рішення	Діапазон значень параметрів, кількість варіантів	Критерії для оцінки та аналізу
Карданна конфігурація і матеріали виготовлення стенду	Матеріал основи	Граніт, залізобетон	Стабільність у часі, вартість виробництва
	Матеріал карданового підвісу	Алюміній, композит	Термічний опір, вимоги до обертового моменту, вартість виробництва, час виходу на ринок, найнижчий торсійний резонанс
	Конфігурація карданового підвісу	Прямокутна, сферична	Жорсткість перерізу, стабільність розмірів, складність інтерфейсу корисного навантаження, складність досягнення вимог та допусків точності під час виробництва, вартість
Опорні підшипники	Тип підшипників	Механічні шарикопідшипники, гідростатичні масляні підшипники, гідростатичні	Хитавиця (биття), тертя, ортогональність осей, жорсткість, надійність, простота

		повітряні підшипники, магнітні підшипники	зовнішньої системи підтримки, вартість
Система кодування положення	Тип датчика кутового положення	Інерційні датчики, синусно-косинусний диференційний трансформатор (СКДТ), оптичні екодери, індуктосин	Перешкодостійкість, Точність, Короткострокова стабільність, похибка, Вартість, Складність.
Система приводів	Тип приводу	Синхронний двигун з постійними магнітами: електродвигуни із явно вираженими полюсами; електродвигуни із неявно вираженими полюсами; синхронний двигун з поверхневою установкою постійних магнітів; синхронний двигун із вбудованими магнітами; з розподіленою обмоткою; з зосередженої обмоткою.	пульсації обертового моменту, синхронний момент, безперервний обертовий момент, простота системи керування, вартість
Система передачі сигналу в обладнання для випробувань	Тип передачі сигналу	Провідний обмежений, провідний необмежений, безпровідний	Перешкодостійкість, Вартість, Складність.

Показані підсистеми, з якими пов'язана кожен проектний параметр, вибір проектного рішення, кількість доступних параметрів для проектного рішення та критерії для оцінки та аналізу.

1.6. Застосування САПР для проектування випробувальних стендів

Компанія Carco Electronics (Acutronic) є найбільшим в світі виробником випробувальних динамічних стендів ракетних систем та випробувальних стендів навігаційного обладнання. Основною спеціалізацією компанії

залишаються прецизійні системи управління випробувальних стендів з серво-керуванням. ACUTRONIC постачає динамічні випробувальні стенди, системи імітації динамічних навантажень, стенди випробувань систем з інерційним наведенням та стенди напівнатурних динамічних випробувань.

Нижче наведено приклад застосування САПР для проектування та виготовлення системи динамічного випробування обладнання, здатну відтворювати збурення, які виникають у типовій системі наведення ракети в польоті [71]. Використання інструментів MathWorks для розробки прототипу контролера системи динамічного випробування обладнання, генерації коду і моделювання контролера з реальною механічною системою, дозволило знизити витрати на розробку на 30%, а час проектування скоротився більш ніж на два місяці. Крім того система управління була оптимізована відповідно до специфікацій та стандартів.

Ракети-перехоплювачі схильні до дуже високочастотних динамічних рухів, таких як вібрація. Це може привести до помилок прямої видимості датчика в замкнутому контурі наведення ракети. Для вирішення цієї проблеми інженери використовують імітатори руху в польоті для відтворення збурень. Однак найсучасніші випробувальні системи мають смугу пропускання всього 60-100 Гц, що призводить до ненадійних результатів випробувань.

Компанія CARCO Electronics використовувала продукти MathWorks для розробки і виробництва випробувальної системи з шістьма ступенями свободи, яка може працювати на частоті 1000 Гц. Продукти MathWorks заощадили CARCO час і кошти, дозволивши змоделювати поведінку контролера і механічної системи разом, а потім оптимізувати систему управління до створення високовартісного прототипу.

Рис. 1.9. Система динамічного випробування обладнання CARCO з шістьма ступенями свободи

Завдання на проектування. Компанія CARCO розробляла систему динамічного випробування обладнання, яка могла б точно відтворювати низькоамплітудну, високочастотну динаміку до 1000 Гц. Можливість створення мехатронної системи, що володіє необхідними властивостями, була під питанням. Розробка традиційної платформи Стюарта (гексапод), яка могла б працювати на частоті 1000 Гц, представляла певні труднощі. Наприклад, механічна конструкція повинна була бути жорсткою і при цьому мати невелику вагу, а виконавчі механізми і система управління повинні були миттєво реагувати на команди керування.

Тому їм необхідно було подолати дві основні перешкоди: розробити механічну систему, досить гнучку, щоб відповідати їх структурним вимогам, і створити контролер системи управління, здатний досягти частоти відгуку 1000 Гц. Щоб задовольнити високочастотні вимоги, CARCO необхідно було оптимізувати компоненти механічної системи і контролера разом.

Самі по собі ці основні критерії можуть бути оптимізовані. Однак в інтегрованій системі вони являють собою складну інженерну проблему.

Використані САПР системи. CARCO використовувала інструменти MathWorks для розробки високочастотного випробувального стенду, який

складається з трьох анкерних блоків, плити, на якій кріпиться корисне навантаження, шести нижніх і трьох верхніх шарнірів і шести електромагнітних лінійних приводів.

Процес розробки включав п'ять етапів [71] (Рис. 1.10):

1. Моделювання механічної системи за допомогою MATLAB і Simulink
2. Розробка системи управління в Simulink і оптимізація динаміки механічної системи разом з контролером системи управління,
3. Автоматична генерація коду за допомогою Simulink Coder,
4. Симуляція моделі системи управління з реальною системою динамічних випробувань обладнання за допомогою Simulink Real-Time,
5. Верифікація моделі механічної системи за допомогою Simscape Multibody.

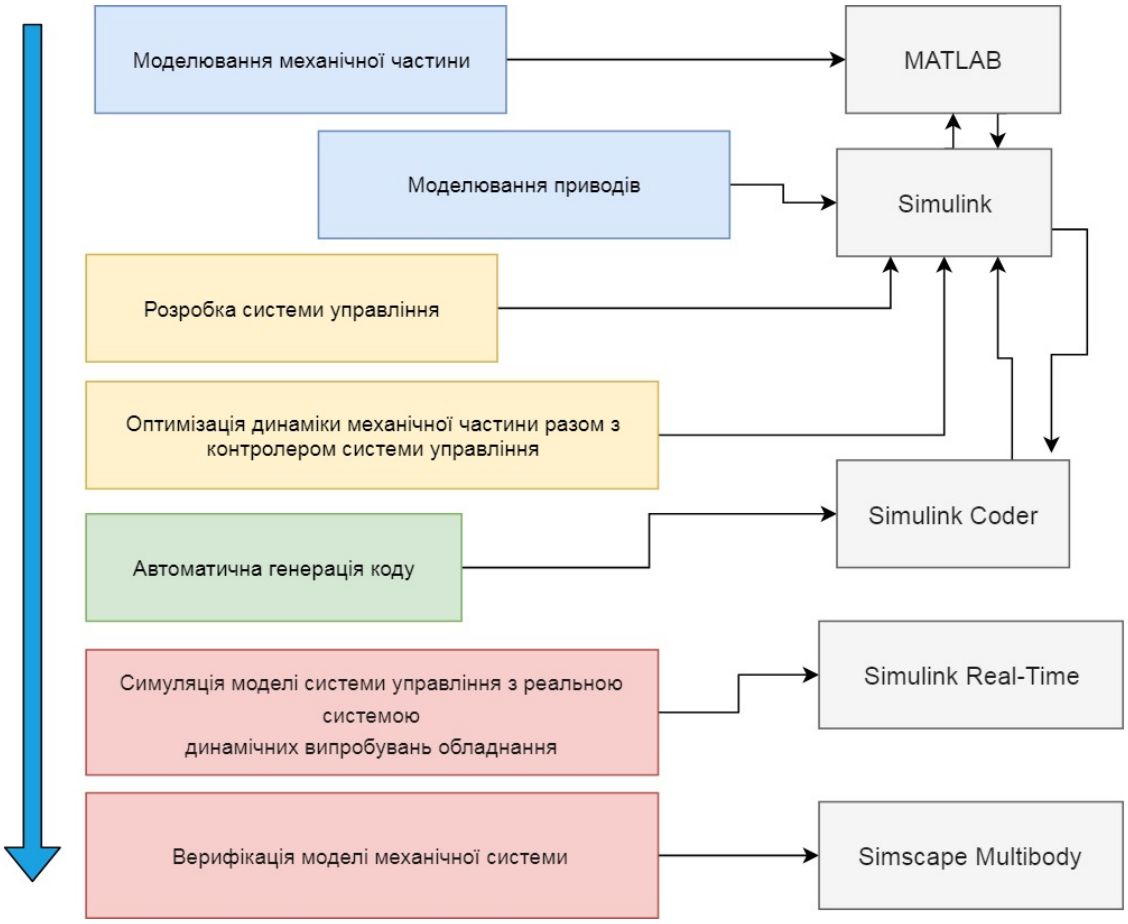


Рис. 1.10. Процес розробки випробувального стенда компанії CARCO

Для механічної системи CARCO розробила новий набір спеціалізованих механічних приводів та змоделювала ці приводи в Simulink.

Для контролера було розроблено багатоосьову, динамічно розв'язану систему управління, щоб мінімізувати вплив безлічі власних значень у пов'язаної кінематики платформи Стюарта. Вони реалізували алгоритми управління за допомогою Simulink і MATLAB і визначили архітектуру системи управління за допомогою блоків Simulink. Кожна підсистема визначалась С-функцією, що викликається і відповідні їй вхідні і вихідні змінні [71].

Потім використовувався Simulink Coder для генерації ANSI C коду з розробленого закону управління на блок-схемі Simulink.

На наступному етапі використовувався Simulink Real-Time для запуску алгоритму управління у виробничому середовищі і виконання його в реальному часі на PC-сумісному обладнанні. Потім було створене фізичне обладнання і протестована комбінована система, підключивши обладнання до контролера за допомогою Simulink Real-Time.

Конструкцію було верифіковано за допомогою Simscape Multibody. CARCO виявила, що створення складної моделі за допомогою Simscape Multibody було простим і скоротило час розробки більш ніж на два місяці [71].

Після цього контролер системи управління пройшов інтеграцію і остаточне тестування.

Результати застосування САПР:

- Зниження витрат на розробку на 30%. Після дворічного дослідження впливу продуктів MathWorks на витрати і продуктивність в компанії CARCO, Інститут програмної інженерії Університету Карнегі-Меллона визначив, що CARCO скоротила витрати на розробку на 30%, використовуючи Simulink Coder для автоматичної генерації коду.
- Час проектування скоротився більш ніж на два місяці. Побудова і перевірка вихідної моделі зайняли близько трьох місяців. За допомогою Simscape Multibody модель була завершена протягом двох тижнів.

- Система управління стенду була оптимізована відповідно до специфікацій. Використовуючи Simulink, було знайдено максимальну динамічну ефективність випробувального стенду, а потім, ґрунтуючись на цій інформації, було визначено критерії ефективності реальних приводів. Це дозволило оптимізувати конструкції випробувального стенду до кроку генерації коду [71].

Висновки до розділу 1

Проведено аналіз випробувального комплексу навігаційного обладнання як об’єкта проектування. Встановлено, що проектування ВС – це результат компромісів проектних рішень між людьми, які в цілому не розуміють один одного і переслідують різні цілі і завдання.

Встановлено, що автоматизоване проектування ВС стосується не тільки технічного боку проблеми і не обмежується етапами проектування на всьому життєвому циклі виробу. Воно пов'язане з питаннями більш ефективного управління загальними знаннями в цілому, а також з впровадженням проектної діяльності в глобальний виробничий процес. Тобто насправді питання має полягати в тому, як проектування випробувальних комплексів може спиратися на організоване управління знаннями, а не в тому, як організувати знання для вдосконалення проектування. Таким чином, проблема полягає в інженерії процесів проектування, а не в самому проектуванні.

Визначено, що на питання "Чи можна побудувати САПР-систему, що включає в себе знання про проектування та виріб для поліпшення процесу проектування та для поширення цих знань серед усіх людей, які мають справу з проектованим випробувальним комплексом протягом всього його життєвого циклу?" ми можемо відповісти, що традиційні інформаційні технології не здаються досить придатними для того, щоб це допустити. З іншого боку, нові підходи та САПР, розроблені в області автоматизованого проектування, мабуть, мають можливість вирішити частину питання.

Здійснено формалізовану постановку завдання проектування випробувального стенду за допомогою САПР із визначенням завдань, виконання яких дозволить розробити ВС та побудувати автоматизовану систему проектування ВС.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ ДО РОЗДІЛУ 1

1. D. Titterton and J. Weston, *Strapdown Inertial Navigation Technology*, 2nd ed. London, UK: The Institution of Engineering and Technology, 2005.

2. *IEEE Recommended Practice for Inertial Sensor Test Equipment, Instrumentation, Data Acquisition, and Analysis*, IEEE 1554-2005, 2005.

3. В. Д. Малинский, В. Х. Бегларян, Л. Г. Дубицкий, *Испытания аппаратуры и средств измерений на воздействия внешних факторов: справочник*. М.: Машиностроение, 1993.

4. Р. И. Адгамов, М.М. Берхеев, И. А Заляев, *Автоматизированные испытания в авиастроении*. М.: Машиностроение, 1989.

5. “Simulation & Test”. Acutronic. [Електронний ресурс]. Режим доступу: https://www.acutronic.com/simulation-test (дата звернення 24.05.2020).

6. О. Н. Анучин и Г. И. Емельянцев, *Интегрированные системы ориентации и навигации для морских подвижных объектов*. 2-е изд. СПб: Электроприбор, 1999.

7. В. Г. Пешехонов, “Современное состояние и перспективы развития гироскопических систем,“ *Гироскопия и навигация*, № 1(72), С. 3-16, 2011.

8. А. П. Буравлев, Б. Е. Ландау, С. Л. Левин, “О модели дрейфа ЭСГ для БИНС,” *Судостроительная промышленность*, № 30, С. 43-48, 1992.

9. И. Б. Челпанов, М. И. Евстифеев, А. В. Кочетков, “Акустические испытания микромеханических гироскопов,” *Интернет-журнал «Науковедение»*, №5(18), С.1-8, 2013, номер статьи в журнале: 106ТВН513.

10. L. A. Demore, R. A. Peterson, L. B. Conley; H. Havliscek, N. P. Andrianos, “Design study for a high-accuracy three-axis test table,” *Journal of Guidance, Control, and Dynamics*, vol. 10, no. 1, pp. 104-114, 1987.

11. “Испытательный Центр”. НПО «Наука» [Електронний ресурс]. Режим доступу: https://npo-nauka.ru/technologies/testing_center/ (дата звернення 15.04.2020).

12. В. В. Аврутов и Н. И. Бурау, *Надежность и диагностика приборов и систем. Учебн. пос.* К.: НТУУ «КПИ», 2014.

13. В. В. Аврутов, І. В. Аврутова, В.М. Попов*, Випробування приладів і систем. Види випробувань та сучасне обладнання. Навч. посіб*. К.:, НТУУ «КПІ», 2009.

14. О.П. Глудкин, *Методы и устройства испытаний РЭС и ЭВС. Учеб. для вузов*. М.: Высш. шк., 1991

15. “Helmholtz Coil System”. NanoSat Lab [Електронний ресурс]. Режим доступу: https://nanosatlab.upc.edu/en/facilities-folder/helmholtz-coil-system (дата звернення 17.04.2020).

16. “Expert Calibration And Measuring Service”. Laboratorio Elettrofisico [Електронний ресурс]. Режим доступу: https://www.laboratorio. elettrofisico.com/measuring-service/ (дата звернення 17.04.2020).

17. Б. Е. Ландау, “Электростатический гироскоп со сплошным ротором,” *Гироскопия и навигация*, № 1(1), С. 6-12, 1993.

18. *Испытания и контроль качества продукции.* ГОСТ 16504-81, 2011.

19. “Motion Simulation Solutions”. Ideal Aerosmith [Електронний ресурс]. Режим доступу: http://ideal-acrosmith.com (дата звернення 17.04.2020).

20. “Expert In Environmental Simulation”. Climats [Електронний ресурс]. Режим доступу: https://www.climats-tec.com/en/ (дата звернення 17.04.2020).

21. И. Б. Челпанов, М. И. Евстифеев, А. В. Кочетков, “Роторное стендовое оборудование для испытаний микромеханических приборов,” *Интернет-журнал «Науковедение»*, №5, С.1-11, 2013, номер статьи в журнале: 109TBH513.

22. Д. В. Калихман, *Прецизионные управляемые стенды для динамических испытаний гироскопических приборов*. СПб.: ГНЦ ЦНИИ «Электроприбор», 2008.

23. D. K. Joos and U. K. Krogmann, “Estimation of strapdown sensor parameters for inertial system error-compensation,” *in AGARD Conf. Proc.*, Mar. 1981, pp. 10.1-10.25.

24. Л. Г. Доросинский, О. М. Зверева, *CALS-технологии*. М.: LAP Lambert Academic Publishing, 2014.

25. С. Ф. Коновалов, Е. А. Никитин, Л. М. Селиванова, *Гироскопические системы. Проектирование гироскопических систем*. М.: Высшая школа, 1980.

26. Д.С. Пельпор, *Гироскопические приборы и автопилоты*. М.: Машиностроение, 1964.

27. Д. С. Пельпор, *Гироскопические системы*, в 3 ч. М.: Высшая школа, 1986-1988.

28. Д. С. Пельпор, И. А. Михалев, В. А. Бауман, В. А. Матвеев, В. В. Ягодкин, С. Ф. Коновалов, *Гироскопические системы. Гироскопические приборы и системы*. М.: Высшая школа, 1988.

29. Д. С. Пельпор, Ю. А. Осокин, Е. Р. Рахтеенко, *Гироскопические приборы систем ориентации и стабилизации*. М.: Машиностроение, 1977.

30. Л. И. Брозгуль, *Динамически настраиваемые гироскопы. Модели погрешностей для систем навигации*. М: Машиностроение, 1989.

31. Л. И. Брозгуль, Е. Л. Смирнов, *Вибрационные гироскопы*. М.: Машиностроение, 1970.

32. Л. З. Новиков, М. Ю. Шаталов, *Механика динамически настраиваемых гироскопов*. М.:Наука, 1985.

33. Д. С. Пельпор, В. А. Матвеев, В. Д. Арсеньев, *Динамически настраиваемые гироскопы. Теория и конструкция*. - М.: Машиностроение, 1988.

34. В. И. Бусурин, Ю. Р. Носов, *Волоконно-оптические датчики: Физические основы, вопросы расчёта и применения*. М.; Энергоатомиздат, 1990.

35. В. Э. Джашитов, В. М. Панкратов, *Датчики, приборы и системы авиакосмического и морского приборостроения в условиях тепловых воздействий*. СПб: ГНЦ ЦНИИ «Электроприбор», 2005.

36. Ю. А. Лукомский, В. Г. Пешехонов, Д. А. Скороходов, *Навигация и управление движением судов*. СПб: Элмор, 2002.

37. Т. Окоси, К. Окамото, М. Оцу и др., *Волоконно-оптические датчики*, пер. с яп. Г. Н. Горбунова. Л.: Энергоатомиздат, 1990.

38. П. К. Плотников, А. В. Михеев, В. Е. Прилуцкий, Ю. Н. Коркишко, В. А. Федоров, “Волоконно-оптический гирокомпас на основе бесплатформенной инерциальной навигационной системы ориентации и навигации,” *XI СПб межд. конф. по интегрированным навигационным системам*, 2004, С. 140-141.

39. А. Г. Шереметьев, *Волоконный оптический гироскоп*. М.: Радио и связь, 1987.

40. С. И. Бычков, Д. П. Лукьянов, А. И. Бакаляр, *Лазерный гироскоп*. М.: Сов. радио, 1975.

41. А. В. Молчанов, Д. А. Морозов, И. В. Осетров, М. В. Чиркин, “Эксплуатационная надежность кольцевых лазеров, используемых в инерциальных навигационных системах,” *XI СПб межд. конф. по интегрированным навигационным системам*, 2004, С. 147-149.

42. В. Ф. Журавлёв, Д. М. Климов, *Волновой твердотельный гироскоп*. М.: Наука, 1985.

43. В. Ф. Журавлёв, Е. А. Измайлов, “Анализ условий, порождающих дрейф волнового твердотельного гироскопа,” *VIII СПб межд. конф. по интегрированным навигационным системам*, 2001. С. 67-74.

44. Б. С. Лунин, “Проблемы и пути развития конструкции волновых твердотельных гироскопов,” *XII СПб межд. конф. по интегрированным навигационным системам*, 2005, С. 220-224.

45. В. А. Матвеев, В. И. Липатников, А. В. Алехин, *Проектирование волнового твердотельного гироскопа: учеб. пособие*. М.: Изд. МГТУ им. Н.Э. Баумана, 1997.

46. В. А. Бурцев, К. А. Злотников, А. М. Лестев и др., “Особенности комплексирования объемной микромеханики и БИС в измерительных системах,” *X СПб межд. конф. по интегрированным навигационным системам*, 2003, С. 217-225.

47. С. Ф. Коновалов, А. В. Кулешов, Н. А. Носов, В. П. Подчезерцев, В. В. Фатеев, Е. Н. Фролов, К. Б. Квон, С. В. Нам, “Вибрационные датчики угловой скорости,” *X СПб межд. конф. по интегрированным навигационным системам*, 2003, С. 261-271.

48. А. П. Мезенцев, В. М. Ачильдиев, В. Н. Дрофа, “Состояние и перспективы создания микромеханических вибрационных гироскопов и акселерометров для бесплатформенных инерциальных навигационных систем,” *X СПб межд. конф. по интегрированным навигационным системам*, 2003, С. 176-177.

49. В. Г. Пешехонов, Л. П. Несенюк, С. Г. Кучерков, М. И. Евстифеев, Я. А. Некрасов, С. Ренар, П. Пфлюгер, К. Пизелла, Ж. Колле, “Результаты разработки микромеханического гироскопа,” *XII СПб межд. конф. по интегрированным навигационным системам*, 2005, С. 268-274.

50. И. В. Попова, А. А. Лестев, Е. М. Семенов и др., “Микромеханические датчики и системы, практические результаты и перспективы развития,” *XII СПб межд. конф. по интегрированным навигационным системам*, 2005, С. 262-265.

51. И. В. Попова, А. М. Лестев, А. А. Семенов, Е. Н. Пятышев, М. С. Лурье, В. А. Иванов, А. А. Шабров, “Микромеханические датчики и системы, практические результаты и перспективы развития,” *XII СПб межд. конф. по интегрированным навигационным системам*, 2005, С. 262-267.

52. В. Я. Распопов, *Микромеханические приборы*. Тула: Гриф и К., 2004.

53. С. Ф. Коновалов, А. В. Кулешов, В. П. Подчезерцев, В. В.Фатеев, О. А. Кеун, “Разработка вибрационного гироскопа, предназначенного для эксплуатации в условиях экстремальных ударных воздействий,” *XI СПб межд. конф. по интегрированным навигационным системам*, 2004, С. 200-207.

54. В. Д. Андреев, *Теория инерциальной навигации. Автономные системы*. М.: Наука, 1966.

55. В. Н. Бранец, И. П. Шмыглевский, *Введение в теорию бесплатформенных инерциальных навигационных систем*. М.: Наука, 1992.

56. И. Б. Вайсгант и др., *Инерциальные навигационные системы морских объектов*. Л.: Судостроение,1989.

57. *Гироскопы. Производство и исследование*, (под ред. Г.А. Сломянского). М.: Машиностроение, 1969.

58. В. П. Данилин, *Гироскопические приборы*. М.: Высшая школа, 1965.

59. Л. И. Каргу, *Гироскопические приборы и системы*. Л.: Судостроение, 1988.

60. Д. П. Лукьянов, Л. А. Северов, Е. Л. Смирнов, А. В. Тиль, *Тенденция совершенствования гироскопов и гиростабилизированных платформ*. Т.30. Л.: Изв. вузов СССР. Приборостроение. 1987.

61. Б. И. Назаров, *Гироскопические устройства*. М.: Миноборонысы СССР, 1970.

62. Е. А. Никитин, А. А. Балашова, *Проектирование дифференцирующих и интегрирующих гироскопов и акселерометров*. М.: Машиностроение, 1969.

63. Е. А. Никитин, С. А. Шестов, В. А. Матвеев, *Гироскопические системы. Элементы гироскопических приборов*. М.: Высш. шк., 1988.

64. А. В. Репников, Т. П. Сачков, А. И. Черноморский, *Гироскопические системы*. М.: Машиностроение, 1983.

65. Я. Н. Ройтенберг, *Гироскопы*. М.: Наука, 1966.

66. V. M. Sineglazov and S. O. Dolgorukov, “Dynamic testing of UAV navigation system,” *in Proc. IEEE 4th Int. Conf. Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, (Kyiv, Ukraine), 2017, pp. 211-214, doi: 10.1109/APUAVD.2017.8308812.

67. V. M. Sineglazov and S. O. Dolgorukov, “Multi objective optimization for UAV navigation equipment test table design,” *in Proc. IEEE 4th Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, (Kyiv, Ukraine), 2016, pp. 117-120, doi: 10.1109/MSNMC.2016.7783121.

68. V. M. Sineglazov and S. O. Dolgorukov, “Dynamic hardware-in-the-loop UAV ground testing system,” *in Proc. IEEE Int. Conf. Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, (Kyiv, Ukraine), 2015, pp. 91-94, doi: 10.1109/APUAVD.2015.7346569.

69. V. M. Sineglazov and S. O. Dolgorukov, “Complex criteria for navigation equipment test table optimal design problem,” *Electronics and control systems*, no. 1, pp. 87-92, 2015.

70. V. M. Sineglazov and S. O. Dolgorukov, “Environmental Chamber for Navigation Equipment Test Table,” *Electronics and control systems*, no. 2, pp. 62-66, 2017.

71. “CARCO Electronics Reduces Development Costs by 30%”. MATLAB & Simulink. [Електронний ресурс]. Режим доступу: https://www.mathworks.com/company/user_stories/carco-electronics-reduces-development-costs-by-30.html (дата звернення 18.05.2020).

РОЗДІЛ 2.
РОЗРОБКА АВТОМАТИЗОВАНОЇ СИСТЕМИ ПРОЕКТУВАННЯ ВИПРОБУВАЛЬНИХ СТЕНДІВ

2.1. Автоматизоване проектування як засіб підвищення ефективності вирішення завдань проектування випробувальних стендів

Великі проекти інженерного проектування вимагають співпраці багатопрофільних проектних груп, які використовують складні і специфічні інженерні інструменти, такі як комерційні інструменти САПР, інженерні бази даних, системи, засновані на знаннях, системи моделювання або інші обчислювальні інструменти спеціального призначення. Окремі проектні групи працюють паралельно і незалежно, часто протягом досить тривалого часу, з інструментами, розташованими на різних майданчиках. Таким чином, в будь-який час окремі члени можуть працювати над різними версіями проекту, розглядаючи його з різних точок зору, як наприклад, електроніка, виробництво, планування, на різних рівнях деталізації. Для координації проектної діяльності цих груп і забезпечення взаємодії між інженерними інструментами необхідно розробити ефективне розподілене інтелектуальне середовище проектування. Таке середовище має не тільки автоматизувати окремі завдання, як наприклад, традиційні засоби автоматизованого проектування, а й допомагати окремим членам групи обмінюватися інформацією, проектними даними та знаннями і координувати свої дії при пошуку оптимального технічного рішення.

В попередньому розділі ми стверджували, що, хоча методи планування штучного інтелекту (ШІ) можуть бути потенційно корисні в багатьох областях виробництва, цей потенціал не може бути реалізований без розробки більш конкретних і надійних підходів до питань, що важливі для інженерів-розробників. Ми також стверджували, що якщо поглянути на питання, важливі для інженерів-технологів, то дослідники ШІ зможуть виявити принципи, придатні для планування з допомогою ШІ в інших суміжних областях. У даній роботі зроблена спроба вирішити обидві ці задачі в області планування

виробництва, яка значно відрізняється від задачі, описаної в [1,2]: проектування і виробництво складних електромеханічних пристроїв. Основна увага приділяється використанню методів ШІ для планування процесів при розробці і виробництві випробувальних комплексів навігаційного обладнання.

На Рис. 2.1 проілюстровано цикл проектування і виробництва випробувальних стендів навігаційного обладнання (ВСНО), який є високо міждисциплінарним за своєю природою. Конструктори випробувальних стендів навігаційного обладнання розробляють детальне завдання на проектування; конструктори механічної частини розробляють пристрої для опору ударним і вібраційних навантаженням, розробляють вузли обертання, системи термокамер для підведення тепла чи холоду до корпусу випробуваного пристрою, тощо.

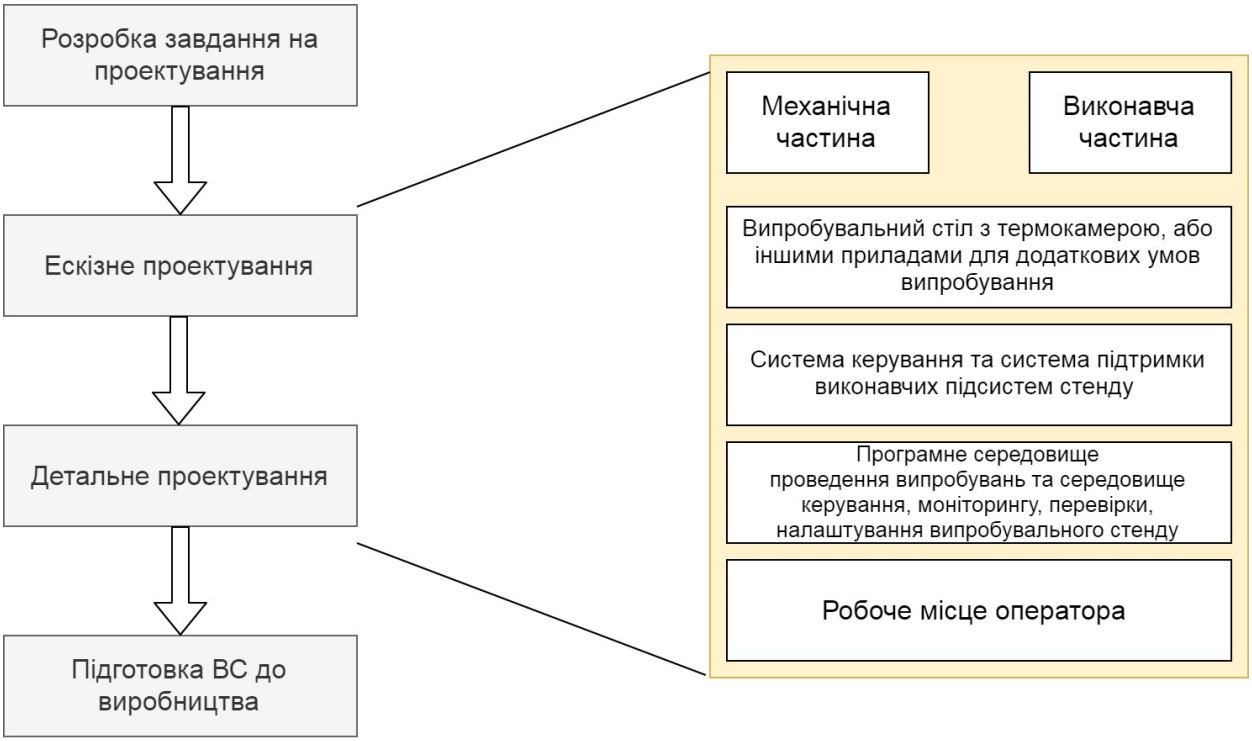


Рис. 2.1. Цикл проектування і виробництва випробувальних стендів навігаційного обладнання

Процедури розробки ВС, що використовуються на практиці в машинобудівній промисловості, можуть бути описані відповідно до загальної моделі процедур стандарту VDI 2206 [46,49]. При розробці ВС складна

функціональна структура ВС приводить до виникнення безлічі завдань, що вимагають компетенцій з різних дисциплін. Як правило завдання механічного проектування, електричного проектування і проектування системи управління вирішуються відділами, які розділені за організацією процесів в дуже послідовному процесі розробки. З метою скорочення часу розробки, використовується процес одночасної і інтегрованої обробки всіх завдань і етапів з використанням методів спільної розробки. Розповсюджене використання стандартних компонентів, що поставляються ззовні, або повних функціональних вузлів, таких як поворотні столи чи системи керування приводами.

Процес проектування ВС складається с наступних етапів:

1. Розробка завдання на проектування:

- Визначення цільової вартості, експлуатаційних характеристик, робочих умов, завдань випробувань, ступені автоматизації.

2. Ескізне проектування

- Вибір типу динамічних випробувань

- Визначення типу ВС та кількості осей

- Вибір типу і кількості пристосувань для встановлення навігаційного обладнання на ВС, пристосувань для підключення до системи обробки даних і необхідного додаткового обладнання

- Призначення рухів окремих компонентів ВС

- Визначення просторового положення осей переміщення

- Вибір датчиків зворотного зв’язку і приводів

3. Детальне проектування

- Вибір, визначення розмірів і просторового положення, що визначають конструкцію компонентів ВС: такі компоненти, як датчики, приводи, елементи для встановлення навігаційного обладнання

- З'єднання, що визначають конструкцію компонентів через компоненти карданного підвісу. Це компоненти, що залежать від форми елементів стенду.

- Вибір і визначення розмірів відповідних конструкцій рам карданного підвісу, функціонально-обумовлених отворів і детальних геометричних форм для компонентів карданного підвісу та несучої конструкції

4. Підготовка ВС до виробництва

- Геометрична деталізація у спеціалізованих САПР

- Доповнення виробничих специфікацій ВС

- Детальне визначення точної структури підсистем ВС

Для багатьох існуючих нині випробувальних комплексів рішення, що приймаються в процесі проектування виробу, будуть визначати більшу частину собівартості виготовлення випробувального комплексу. Це породило філософію інтегрованого проектування продукту і технологічного процесу, яка намагається враховувати виробничі особливості в процесі проектування продукту. Однак при проектуванні випробувального комплексу це вимагає координації дій у великий міждисциплінарній групі розробників. У великих організаціях це може бути важким завданням [2,3].

Через складність мультидисциплінарних проблем проектування виникає потреба спроектувати процес проектування. Іншими словами, необхідна фаза мета-проектування, щоб лінгвістично визначити та структурувати процес проектування та продукт, що створюється [50]. Існує багато підходів до формулювання та вирішення складних проблем проектування, включаючи різні одно- та багаторівневі підходи. Процес розробки виробів мехатронних систем може здійснюватися багато в чому згідно загальної методології проектування, описаної в VDI 2206. Проте не існує "канонізованої оптимальної форми процесу проектування, якій проектувальник може слідувати в фіксованому графіку" [49]. У цьому розділі розглянуто мультидисциплінарне оптимальне проектування (МОП) з теоретико-ігрової точки зору, проектування на основі інтелектуальних агентів та досліджуються шляхи вирішення завдань МОП ВС.

2.1.1. Загальні вимоги до конструкції випробувальних стендів

Ефективність ВС визначається характеристиками кількості випробувань, виробничих витрат і точності випробувань. У динамічних ВС динамічна поведінка системи має вирішальне значення для досягнення ефективності ВС. В процесі випробувань конструкція ВС, приводи і система управління піддаються постійним навантаженням і вібраційним збуренням через вагові навантаження, дисбаланс, статичні і динамічні технологічні сили, а також сили маси і тертя при прискоренні осей стенда. Вся конструкція ВС являє собою коливальну багатомасову пружинно-демпферну систему, механічний резонанс якої обмежує динаміку приводу і досяжну точність управління. Відхилення в управлінні приводами осей стенда, а також деформації конструкції ВС через прикладені статичні навантаження призводять до помилок відпрацювання траєкторії руху в процесі випробувань. Динамічні навантаження призводять конструкцію ВС і приводи до вібрації. Якщо динамічну поведінка ВС несприятлива, вібраційні ефекти призводять до погіршення якості випробувань і підвищеного зносу ВС, що змушує знижувати динаміку і швидкість обертання осей ВС.

Це призводить до наступних основних вимог до розробки ВС з високою точністю траєкторії при високих швидкостях динамічних випробувань [52]:

• Механічні конструкції з високою статичної та динамічної жорсткістю при малих рухливих масах.

• Динамічні приводи осей стенда з високою здатністю прискорень і високою швидкістю переміщення.

• Контроль стану осей стенду з високою точністю та достовірністю отримання інформації.

• Швидке управління з короткими циклами інтерполяції, а також управління рухом з малими ривками з безперервною прогресією швидкостей і прискорень осей ВС.

2.1.2. Функціональне проектування випробувальних стендів

Структуру трьох-осевого випробувального динамічного стенду наведено на Рис. 2.2. Випробувальний стенд складається з: механічної частини, виконавчої частини , випробувального столу, системи керування та системи підтримки виконавчих підсистем стенду, робочого місця оператора, програмного середовища.

Стенд представляє собою керований кардановий підвіс, що містить блок управління приводними двигунами, керуючий пристрій системи управління та персональний комп'ютер (ПК). Керуючий пристрій на базі ПЛІС приймає сигнали датчиків на осях підвісу стенду та виробляє алгоритм управління двигунами

Випробувальний стенд складається з:

- ККП - керований карданний підвіс;
- Дα, Дβ, Дγ - двигуни осей тангажу, крену та курсу;
- ДКα, ДКβ, ДКγ – датчики кутового положення осей тангажу, крену та курсу; П - платформа випробувального стенда;
- НО – навігаційне обладнання;
- КП - керуючий пристрій на базі ПЛІС;
- АГ, АФ, АУ - алгоритми генерування, формування та управління;
- БУД – блок управління двигунами,
- ПК – комп'ютер для обробки, зберігання та документування результатів випробувань та керування експериментом.

Керуючий пристрій на базі ПЛІС має структуру, описану нижче:

- ПЛІС з кількістю програмованих логічних схем не менше 8000;
- Цифро-аналоговий інтерфейс, що містить АЦП і ЦАП з програмованою частотою дискретизації (розрядність не менше 16) для прийому сигналів з ланцюгів зворотного зв'язку БУД та датчиків кутів та для видачі сигналу на вхід суматора;

- паралельний портовий інтерфейс (розрядність не менше 16) для передачі дискретної інформації з датчиків кута, а також інтерфейс шини ПК і внутрішні шини зв'язку між окремими вузлами модуля керування.

Крім того, передбачається установка на кільце карданного підвісу термокамери з випробуваним приладом. ПК через КП формує закони зміни кутів тангажу α, крену β, курсу γ в часі, відповідно до експлуатаційних умов.

Цими законами можуть бути:

- синусоїдальна хитавиця з різними по осях тангажу, крену та курсу амплітудами, частотами та фазовими зміщеннями;
- випадкові процеси типу «білого шуму» із заданою матрицею спектральної густини;
- комбінований закон (для осі тангажу), що імітує кругове обертання із заданою кутовою швидкістю з накладенням коливань, гармонійних або випадкових.

У схемі керування забезпечується формування керуючих впливів на систему БУД, яка формує керуючі впливи для двигунів по всіх осях. Значення кутів α, β, γ виміряні відповідними датчиками кута ДКα, ДКβ, ДКγ, передаються через цифровий інтерфейс (RS-485) в КП на базі ПЛІС, який здійснює порівняння заданих значень кутів і вимірюваних на кожному такті управління. Сигнали різниці надходять на КП і далі на БУД для коригування відтворених законів руху по осях підвісу випробувального стенду. Інформація з навігаційного обладнання заводиться в ПК.

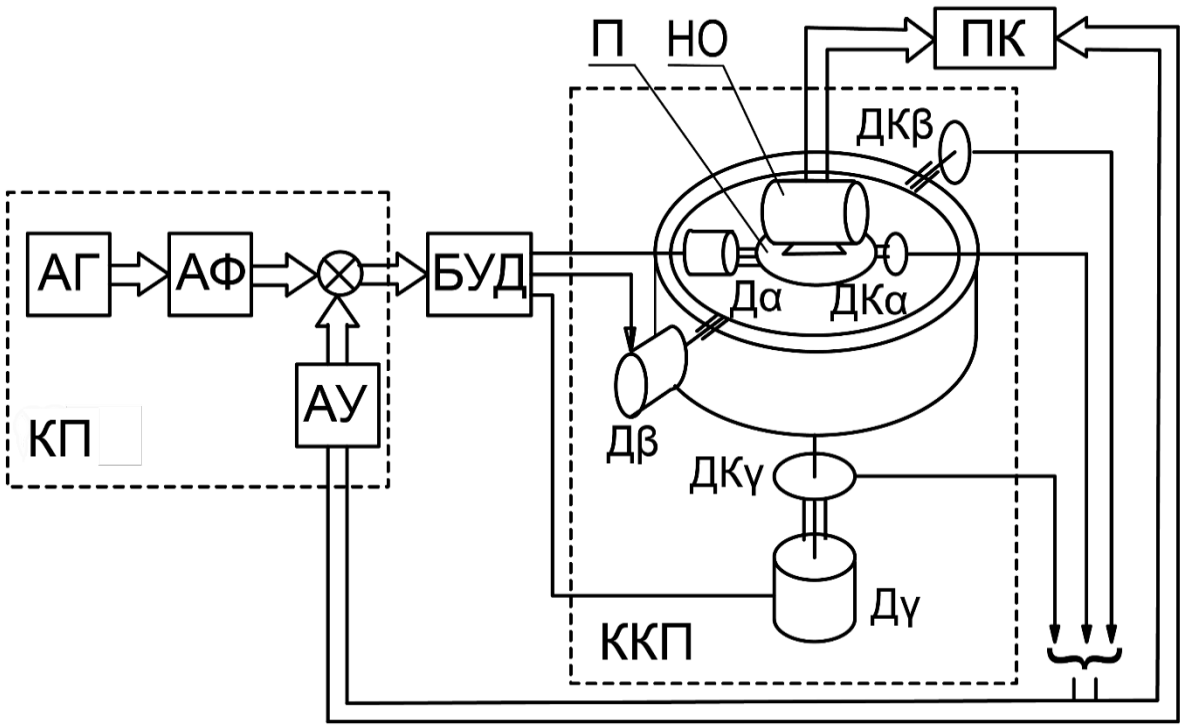


Рис 2.2. Функціональні компоненти ВС навігаційного обладнання

Стенд може бути віднесений до тривісних імітаційно-калібрувальних випробувальних стендів. Він може бути використаний для вирішення завдань калібрування вимірювачів кутових швидкостей за динамічними характеристиками та для випробувань всіх типів гіроскопічних приладів в умовах, що імітують роботу на платформі з хитавицею. Використання випробувального стенду в якості калібрувальної платформи для атестації датчиків кутових швидкостей за статичними характеристикам можливе тільки для датчиків середньої точності та грубих датчиків. Керуючий пристрій на базі ПЛІС (КП) генерує незалежні білі шуми, з яких блоком АФ за допомогою алгоритму формування генерується векторний випадковий процес із заданою матрицею спектральних густин. Блок АУ формує алгоритм управління двигуном, тобто стенд має замкнуту систему управління за кутовим положенням карданового підвісу. Кути за трьома осями карданового підвісу вимірюються датчиками ДКα, ДКβ, ДКγ. Випробувальний стенд, таким чином, імітує хитавицю по осях тангажу, крену та курсу. Завдання контурів

позиційного регулювання полягає в тому, щоб якомога точніше і без затримки відстежувати переміщення окремих осей карданового підвісу відповідно до заданої траєкторії динамічної хитавиці. Приводи по осях тангажу, крену та курсу використовуються для виконання необхідних рухів динамічних випробувань і для створення необхідних зусиль переміщення карданового підвісу. Елементи рами і основна опора разом утворюють механічну несучу конструкцію ВС. Її функція полягає в передачі механічних сил у обладнання, що випробовується. Для того, щоб в точці встановлення навігаційного обладнання зі статичними і динамічними силами навантаження, що виникають в результаті роботи системи, було якомога менше зсувів і відносних вібрацій, необхідна висока ступінь жорсткості і достатня амортизація конструкції ВС і приводів.

2.2. Постановка завдання мультидисциплінарного оптимального проектування випробувального стенду навігаційного обладнання

Розглянемо окремі системи випробувального стенду більш детальніше (Рис. 2.3.).

1. Механічна частини складається з карданового підвісу та інших опорних конструкцій стенду.

2. Виконавча частина складається з електричних приводів, системи кодування положення, система передачі сигналу в обладнання для випробувань. Загалом з опорними підшипниками це утворює підсистему механічної частини, що має приводити в рух кардановий підвіс разом с іншим обладнанням.

3. Випробувальний стіл з термокамерою, або іншими приладами для додаткових умов випробування. Випробувальний стіл має бути спроектований для можливості встановлення та підключення до нього навігаційного обладнання та мати всі необхідні механічні, електричні та інші відповідні вузли.

Рис. 2.3. Випробувальний стенд як складна мехатронна система

4. Система керування та система підтримки виконавчих підсистем стенду (електричних, гідравлічних, пневматичних, тощо).

5. Робоче місце оператора для проведення випробувань

6. Програмне середовище проведення випробувань та середовище керування, моніторингу, перевірки, налаштування випробувального стенду.

Середовище має забезпечувати всі вимоги програми випробувань навігаційного обладнання та мати можливість калібрування та перевірки придатності випробувального стенду та відповідність стенду специфікаціям виробника — програма самодіагностики.

Завдання проектування на першому етапі починається з верхнього рівня підсистем та прямує на рівні підсистем, що підпорядковані за ієрархією.

В цьому дослідженні основна увага сконцентрована на двох верхніх рівнях р.1 та р.2. Інші підсистеми можуть бути спроектовані раніше ніж р.1 та р.2, однак такі обмеження накладають суттєві проблеми та унеможливлюють проектування оптимальних підсистем р.1 та р.2. Основні обмеження та параметри проектування р.3 використовуються як обмеження при проектуванні р.1 та р.2.

Рівень 4 проектується з використанням обмежень р.1 - р.3. Проектування оптимальної системи керування складає окреме комплексне завдання досліджень та виходить за межі цієї роботи.

Рівні проектування 5, 6 та інші підсистеми нижчого рівня мають проектуватись тільки з використанням параметрів обмежень р.1 – р.4. Оптимальне проектування програмного середовища для комплексу випробувального стенду навігаційного обладнання потребує окремих засобів, методології та інструментів дослідження, що не висвітлюються в даній роботі.

Загальне завдання мультидисциплінарного оптимального проектування випробувального стенду навігаційного обладнання показане на рис. 2.4. Позначення термінів на рис. 2.4 розкривається нижче.

(D_1) - виконавча частина (електроніка) складається з електричних приводів, системи кодування положення, система передачі сигналу в обладнання для випробувань. (D_2) - механічна частина складається з карданового підвісу та інших опорних конструкцій стенду (D_3). Загалом з опорними підшипниками (D_4) це утворює підсистему механічної частини, що має приводити в рух кардановий підвіс разом с іншим обладнанням. (D_5) Випробувальний стіл з термокамерою, або іншими приладами для додаткових умов випробування. Випробувальний стіл має бути спроектований для можливості встановлення та підключення до нього навігаційного обладнання та мати всі необхідні механічні, електричні та інші відповідні вузли. (D_6) Система керування та система підтримки виконавчих підсистем стенду (електричних, гідравлічних, пневматичних, тощо). (D_7) Робоче місце оператора для проведення випробувань. (D_8) Програмне середовище

проведення випробувань та середовище керування, моніторингу, перевірки, налаштування випробувального стенду. Середовище має забезпечувати всі вимоги програми випробувань навігаційного обладнання та мати можливість калібрування та перевірки придатності випробувального стенду та відповідність стенду специфікаціям виробника — програма самодіагностики.

Вищезазначений перелік не є вичерпним, його можна продовжити до декількох десятків різних дисциплін D_N, підсистем нижчого рівня N, що складають комплекс випробувального стенду навігаційного обладнання.

Дисципліна D_2 сама є системою нижчого рівня, тому передбачається, що всі функції проектування і зв'язані змінні обчислюються в рамках її дисциплін від *3* до *N*. Таким чином, в математичному формулювання завдання проектування ВС немає f_2та g_2. Однак є системні змінні проектування ВС (*M, A*), пов'язані з підсистемою механічної частини, x_2. У постановці завдання нижче дисципліни з *4* по *N* пов'язані тільки з дисципліною *3*, а не один з одним і не з дисципліною 1. Функції проектування: у*ij* містить ті функції, що обчислені в дисципліні *i*, які потрібні в дисципліні *j*; y^{31}*, y^{43}*: зв'язані змінні; *x*: змінні проектування ВС навігаційного обладнання; x_1, x_2, x_N: змінні проектування дисциплін D_1-D_N; g_1, g_2, g_N: функції обмеження проектування ВС; f_1,f_2, f_N - цільові функції проектування ВС, d_i– функції розбіжності.

Завдання мультидисциплінарного оптимального проектування випробувального стенду навігаційного обладнання полягає в наступному:

Знайти: f, x, x_1, y_{31}*

Мінімізувати: f

Задовольняють: $g_1, < 0, \ f_1 - f < 0,$

$d_2 < 0$

Завдання оптимізації механічної частини ВС (Дисципліна D_2):

Знайти: $x_2 > x_3 > y_{43}$*,…,y_{N3}*,d_2

Мінімізувати: d_2

Задовольнити: $g_3 < d_2, f_3 - f < d_2,$

$y_{31} - y_{31}$* $< d_2, y_{31}$* $- y_{31} < d_2$

$d_3<d_2, ...,$ $d_N<d_2$

Завдання оптимізації дисципліни i, де $i \in [4, N]$:

Знайти: $x_i, d_i,$

Мінімізувати: d_i

Задовольнити: $g_i<di,$ $f_i\text{-}f<d_i,$

$y_{i3}\text{-}y_{i3}^*<d_i,$ $y_{i3}^*\text{-}y_{i3}<d_i$

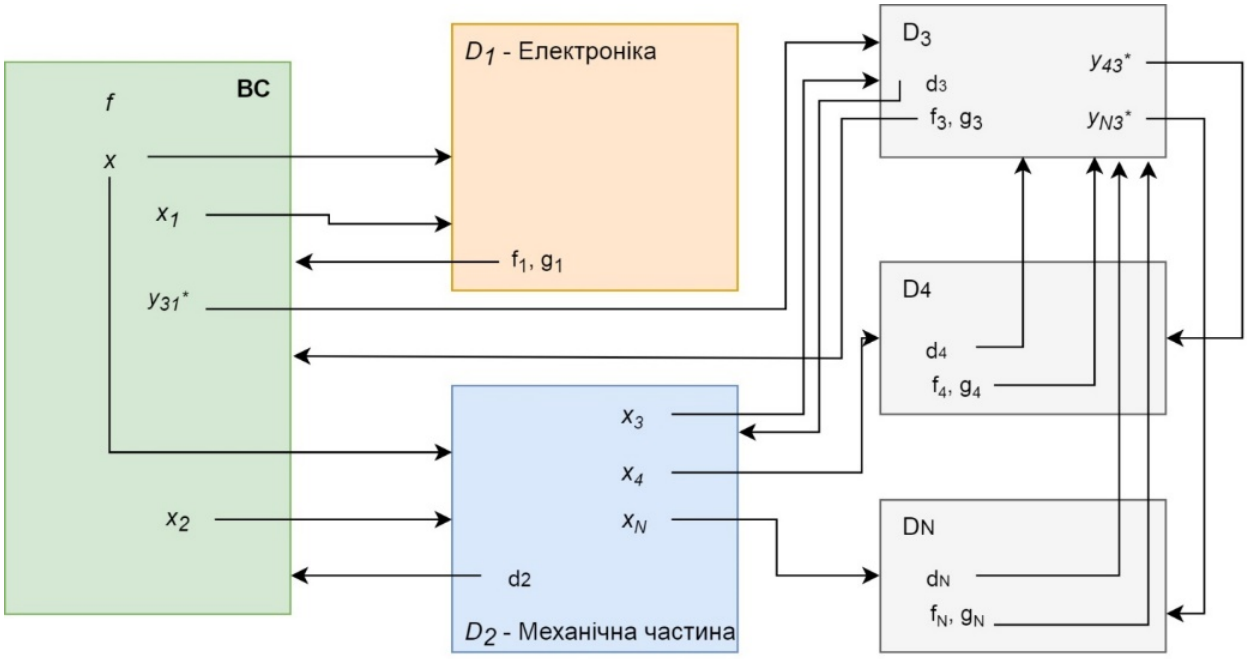


Рис. 2.4. Багаторівнева модель проектування ВС

2.3. Огляд систем гібридного інтелектуального проектування

Цей розділ містить огляд гібридних систем інтелектуального проектування. Гібридні інтелектуальні системи інтегрують системи, що базуються на знаннях, нечіткій логіці, нейронній мережі, генетичних алгоритмах та індивідуальних методах. Їх ефективність була доведена в широкому колі реальних складних проблем, включаючи планування, представлення та аргументацію, інтелектуальний інтерфейс, гібридні та інтегровані системи (САПР, бази даних) тощо. Гібридні інтелектуальні системи швидко зростають у значенні та видимості в інженерному проекті

[17]. Акман, тен Хаген і Вееркамп [18], а також Акман, тен Хаген і Томіяма [19] запропонували бажані функціональні можливості інтелектуальних гібридних САПР та уніфіковану структуру інтелектуального, інтегрованого та інтерактивного САПР (IIICAD) для опису та застосування знання проектування. Проте ніяких деталей щодо розробки та застосування систем IIICAD не надано. Заревар і Гоулдінг [20] описали реалізацію гібридного дизайну САПР на основі нейронних мереж для механічних передач енергії (конструкція коробки передач). Гібридне інтелектуальне середовище складалося з нечіткої експертної системи та нейронної мережі зворотного поширення. Наукова робота є корисним посібником для застосування гібридного інтелекту до проблеми проектування. Бахрамі та Даглі [21] повідомляють про застосування нечіткої асоціативної пам'яті в концептуальному проекті шляхом відображення нечітких функціональних вимог до чіткого проектного рішення. Їх робота встановила основний фундамент інтелектуальної системи САПР, інструменту, який може допомогти проектувальникам на всіх етапах процесу проектування. Відносно мало роботи було зроблено для отримання чіткого проектного рішення на основі вимог нечіткого введення та простоти його практичного застосування, що робить систему привабливим інструментом у фазі створення ідеї процесу проектування. Пауелл, Сколнік та Тонг [22] запропонували уніфікований підхід до оптимізації інженерного проекту на основі їх опитування щодо існуючих методів оптимізації, таких як експертні системи, чисельна оптимізація та генетичні алгоритми, що використовує комбінацію експертних систем, чисельну оптимізацію та генетичні алгоритми. Цей "уніфікований" підхід капіталізує індивідуальні переваги кожної окремої методики оптимізації, але компенсує їх недоліки. Це дозволяє користувачеві зосередитися на проблемі проекту, не турбуючись про вибір та точність алгоритмів оптимізації. Бахрамі Лінч та Даглі [23] далі розглянули застосування гібридних нейронних мереж у проектуванні та виробництві. Система включає три парадигми нейронної мережі - нечітка асоціативна

пам'ять, зворотне поширення та адаптивна резонансна теорія - для досягнення інтеграції проектного продукту-процесу. Підхід нечіткої асоціативної пам'яті використовується для автоматизації відображення маркетингових характеристик у попередньо розроблених структурах. Теорія адаптивного резонансу використовується для ідентифікації об'єкта. Для калібрування камери використовується нейронна мережа зворотного поширення, а інша мережа зворотного поширення використовується для керування швидкістю робочої руки на основі розміру деталі. Це унікальне застосування нейронних мереж у дизайні та виробництві може бути розширено для більш складних інструментів при паралельному проектуванні. Впроваджені концепції посилюють необхідність застосування одночасних інженерних методологій та специфічних технологій на основі штучного інтелекту. Су та Вакелам [4,5,6] застосовують інтегровану інтелектуальну систему, що успішно підходить для інженерного проектування та інтеграції виробництва, яка включає в себе досвід проектування та виробництва та інтеграцію відповідних технологій САПР та ШІ у загальний процес проектування та виробництва. Гібридний інтелектуальний системний підхід був розроблений для інтеграції різноманітних заходів, пов'язаних з проектуванням та виробництвом, включаючи специфікацію проекту продукту, концептуальний проект, деталізацію, планування процесу, оцінка вартості і виробництво виробу.

Він поєднує в собі систему на основі правил (СНП), штучні нейронні мережі (ШНМ), генетичні алгоритми (ГА), мультимедіа та пакети САПР в єдине середовище. СНП та ШНМ охоплюють досвід розробки та виробництва, а інші завдання обробляються відповідними програмними пакетами САПР. ГА та Гіпермедіа використовуються для оптимізації проекту та забезпечують ефективні засоби для користувацьких інтерфейсів та передачі даних. СНП пов’язаний з іншими і працює як координатор, що контролює весь процес. Чен та Очея [25] запропонували платформу програмних обчислень для паралельного проектування розвитку продукту та оцінки, а також розроблена паралельна система оцінки проекту (CONDENSE), яка може допомогти в

оцінці можливих рішень та альтернатив на ранній стадії етапу проектування. Під час етапу пошуку застосовується якісна оцінка аспектів для принципів рішення та їх комбінацій, щоб допомогти визначити технічні характеристики проекту, а також застосовується кількісна оцінка аспектів для надання інформації про продуктивність, збірність, технологічність та витрати, щоб полегшити вибір проекту. Переваги CONDENSE включають пришвидшення проектування та розробки, покращення якості проекту та полегшення вибору проекту. На основі гібридної інтеграції нейронних мереж та нечіткої експертної системи (наприклад, нейро-нечіткої гібридної схеми), Жа і Лім [26,27] розробили платформу для програмних обчислень для проектування та моделювання системи машина-людина. Комплексна система машина-людина описується параметрами людини та машини в рамках всеосяжної моделі, в якій процедури та алгоритми проектування машино-машинобудівної системи, економічної та ергономічної оцінки та оптимізації введені в інтегровану схему САПР та платформу програмних обчислень. При поєднанні індивідуальних нейронних мереж та нечітких логічних прийомів нейро-нечітка гібридна програмно-обчислювальна схема реалізує нечіткий блок якщо-то для розробки, оцінки та оптимізації машин-систем за допомогою обумовленої нечіткої архітектури нейронної мережі. Для навчання та тестування цілі моделюються та виконуються на самостійній автономній лабораторній робочій станції з гнучкою системою вимірювання та аналізу руху. Підготовлена нечітка мережа нейронних мереж здатна прогнозувати пози позицій операторів і спільні кути руху, пов'язані з різними конфігураціями робочих станцій. Розроблена система забезпечує уніфіковану, інтелектуальну обчислювальну структуру для проектування та моделювання стану машини-людини.

В останні роки віртуальне проектування продукту - це поширена технологія, яка дозволяє інженерам візуалізувати багатовимірні властивості нових продуктів на етапі їх проектування без фактичного створення прототипу [24, 28]. Нещодавня доповідь Національного наукового фонду США вказує на

те, що віртуальне інтелектуальне проектування та кібер-інфраструктура відіграють ключову роль у підтримці та формуванні майбутніх предикативних систем і процесів реалізації продуктів. У віртуальному проектуванні продукту, машина практично зібрана; її продуктивність моделюється за допомогою симуляційного пакета та тестується новими змінними входами. Це дозволяє виробникам зменшувати або обходити декілька фізичних етапів прототипу та досягати значно більшого рівня ефективності проектування продукту, заощаджуючи час та вартість з циклу розробки. Ключовим питанням дизайну та розробки віртуального продукту є точне співставлення результатів моделювання з умовами введення, а потім - для оцінки та прогнозування поведінки та продуктивності машини в нових умовах.

Є багато індивідуальних інтелектуальних систем і підходів, таких як експертна система, нечітка логіка, нейронна мережа та генетичні алгоритми, які можуть виконувати деякі проектні завдання [24,28]. Проте через певні обчислювальні властивості індивідуальних технік інтелектуальної системи, гібридні інтелектуальні рішення необхідні для вирішення складних задач проектування, які можуть включати в себе одну або декілька з вищевказаних індивідуальних інтелектуальних прийомів. Концепція програмних обчислень, представлена Сарідакісом та Дентсорасом [29], служить для висвітлення виникнення обчислювальних методологій, в яких акцентом є використання толерантності до неточності та невизначеності для досягнення стійкості, надійності та низької вартості рішення. На цьому етапі основними компонентами програмного обчислення є нечітка логіка, нейро-обчислювальні, еволюційні обчислення та ймовірнісні обчислення, з останніми підгрупами мереж вірування, хаотичними системами та частинами теорії навчання. Програмні обчислювальні методи полегшують використання нечіткої логіки, нейрокомп'ютера, еволюційного обчислення та імовірнісного обчислення в комбінації, що призводить до концепції гібридних інтелектуальних систем. Такі системи швидко зростають у значенні та видимості. У книзі Жа [30], яка базується на гібридній інтеграції нейронних

мереж та експертної системи на основі знань, пропонується конструкція програмних обчислень для проектування. Основним завданням цієї роботи є розробка та впровадження гібридної інтелектуальної системи для інженерного проектування, яка спрямована на подолання обмежень традиційних методів проектування. Її мета полягає не в тому, щоб замінити звичайні методи, а їх доповнювати та доповнювати, щоб можна було належним чином розглянути набагато ширший діапазон проектних ситуацій.

Нещодавно було багато спроб вирішити ці проблеми, застосовуючи нейронну мережу та нечіткі методи експертних систем на основі правил [31]. Можливості експертних систем, основаних на правилах (нечіткі), за своєю суттю добре підходять для вирішення проблем проектування. Головним недоліком, однак, є те, що програміст повинен визначити функції, що лежать в основі багатозначної або рейтингової оптимізації можливостей. Крім того, правила експертного типу використовують комплексну мовну систему, яка може мати вбудовані упередження, вбудовані цілі та приховані інформаційні структури, які можуть призвести до помилок [31]. Нейромережі використовують математичні зв'язки та відображення для розробки та оптимізації систем людини та машини. Вони здатні приймати статистичні рішення, даючи неповну та непевну інформацію про проект, і можуть бути розроблені для адаптації до вимог проектувальника. На відміну від експертних систем на базі правил (нечітка), вони оцінюють всі обмеження конфлікту проекту одночасно, моделюють або вивчають базу знань за допомогою технологій чорної коробки. Вони не використовують правила у формальному значенні, тому час проектування може сильно зменшитися, ніж моделювання на основі правил. Сильні сторони нейронних мереж залежать від того, що вони не потребують апріорних припущень моделей та їх здатності виводити складні, нелінійні основні відносини. З точки зору статистиків, нейронні мережі є по суті статистичними пристроями для здійснення індуктивного висновку і аналогічні непараметричним моделям нелінійної регресії. Проте існуючі нейронні схеми використовують дві чи більше окремі нейронні мережі

щоб виконати деякі проектні завдання відповідно, і треба тренувати їх окремо. Це довго і дорого, іноді дуже складно. Для подолання виявлених недоліків або труднощів необхідні додаткові зусилля для розробки більш загальних топологій нейронних моделей, алгоритмів навчання та апроксимаційних теорій, з тим щоб ці моделі були застосовні в системі моделювання, проектування та управління складними системами. Тому для підтримки проекту потрібен новий вид гібридних нейронних мереж. Слід також визнати, що експертні системи на базі правил набагато легше перевіряють помилки, ніж ансамбль безперервних рівнянь у нейронних мережах [20]. Сьогодні в інтелектуальній спільноті проектувальників зростає потреба в тому, що для складних дизайнерських завдань потрібні гібридні рішення.

Наступне покоління розробки інтелектуальних систем буде інтегрувати обчислювальні парадигми експертних систем або систем з базами знань, штучних нейронних систем, нечіткої логіки [31], управління невизначеностями та генетичні алгоритми [32], та інші методи такі як ієрархічний коеволюційний підхід [33]. Цей підхід є відображенням неефективності чисто символічних, числових чи розподілених представлень та методологій для роботи зі складними даними та даними з випадковою помилкою, що є частиною більшості проблем, що вирішуються (Рис. 2.5.).

Дві методології розробки, були отримані від систем базованих на знаннях, що дозволяють поєднувати в собі декілька «інтелектуальних» методів та злиття даних, - це ієрархічний підхід та підхід на основі концепції дошки оголошень. Ієрархічний підхід до проектування інтелектуальних систем передбачає використання різних рівнів або декількох рівнів. Цілі високого рівня розкладаються або поділяються на більш елементарні або простіші задачі або цілі. Цей поділ продовжує існувати до тих пір, поки завдання не стануть легкодоступними. Інформація від різних модулів нижчого шару доступна для більш високих рівнів зазвичай за допомогою механізму зв'язку, який також дозволяє системі встановити петлі зворотного зв'язку;

інформація на нижчих шарах може впливати на більш високі рівні, що, в свою чергу, змінює інформацію, що надходить на нижчі рівні.

Цей підхід має легкодоступну і описану структуру управління; контроль, як правило, зверху внизу

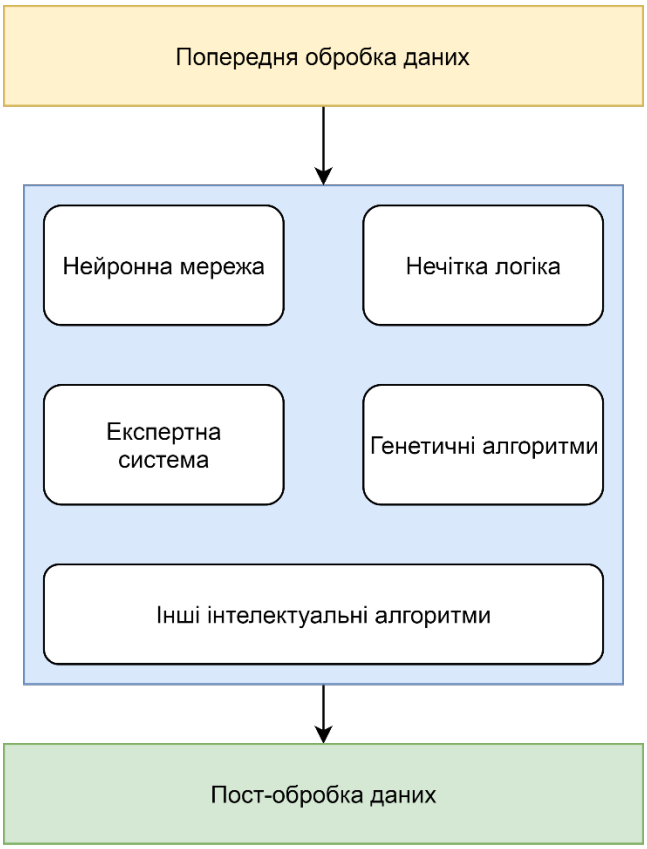


Рис 2.5. Гібридна інтелектуальна система

Це робить системи порівняно простими. На жаль, зворотний зв'язок, який є можливим у цьому прикладі, складніше описати та контролювати. Зворотній зв'язок важко реалізувати, якщо система призначена для багатьох рівнів. Часто це змушує інженерів обмежувати кількість шарів і зазвичай ці шари є зв’язаними. Це може призвести до недостатнього використання інформації та перешкоджає вирішенню проблем системи як цілісного об’єкту.

Створення наступного покоління інтелектуальних систем буде інтегрувати обчислювальні парадигми експертних або систем заснованих на знаннях та штучних нейронних мереж, а також інших методологій, таких як нечіткої логіки та генетичних алгоритмів. Це є відображенням неефективності

системного підходу фізичної символьної системи, класичної інформаційної системи, де поняттям даються локальні уявлення, і ці символи маніпулюють ланцюгами висновків або правилами, зазвичай, під централізованим контролем. Реальні проблеми, що існують у середовищі обчислень і швидкому переналагоджувальному виробництві, використовують складні та неточні дані та цілі, а класичні системи ШІ, як правило, мають обмежений простір, протиріччя у роботі з задачами, а також мало можливостей навчання.

Метою проектування сучасних інтелектуальних систем є визначення, проектування та впровадження систем, що мають високий рівень машинного інтелекту, щоб ці системи могли самостійно вивчати і адаптуватися в непевних або частково відомих середовищах. Використовуючи розподілені уявлення, ШНМ отримують свої висновки, застосовуючи роботи та інструменти, що модернізуються. Цю організацію, як правило, можна розглядати, як ієрархію. Найбільш поширеною діяльністю в системі є планування. Дії відбираються, ресурси заплановані, а цілі, які мають бути досягнуті, пов'язані з виробництвом, якістю та економікою. Все це потрібно здійснити у динамічному середовищі, і, в ідеалі, інтелектуальна система повинна реагувати на зміни в режимі реального часу.

Оптимізація багатопрофільного проектування - метод оптимізації , який використовується в інженерних задачах з декількома підсистемами. Досягнення в області обчислювальних потужностей підвищили популярність даного виду методології оптимізації. У спробі об'єднати системне розкладання і оптимізацію з'явилося кілька методів , таких як спільна оптимізація (CO) і дворівневий синтез інтегрованої системи (BLISS). [36]

Обидві методології є потужними і ефективними для вирішення проблем з чітко визначеними дисциплінарними межами, великою кількістю загальних змінних і обчислень, а також мінімальною міждисциплінарною взаємодією. Однак, з іншого боку, це призводить до обчислень з високою розмірністю , що вимагає високої обчислювальної потужності.

BLISS працює аналогічно CO, метою якого є оптимізація розподілених інженерних систем, створених досвідченими командами, що одночасно працюють над проектною завданням. Методологія відрізняється від CO тим, що для оптимізації на рівні підсистем використовуються вагові коефіцієнти переваг. Проектні рішення в BLISS розділені на три групи. У першій групі змінні оптимізуються на локальному рівні і знаходяться тільки в кожної з підсистем. Друга група містить змінні, які виводяться однієї підсистемою і використовуються як вхідні дані іншій підсистеми. Нарешті, третя група включає в себе змінні проектування системного рівня, які є загальними як мінімум для двох підсистем .

Розкладання проблеми на більш дрібні завдання робить загальну проблему більш керованою , але в результаті оптимізація на системному рівні є більш складною задачею. Об'єднання різної кількості підсистем в одну непроста задача. З ростом складності системи зростає і складність моделі, необхідної для завершення оптимізації системного рівня .

Мульти-агентна координація є ключовим напрямком досліджень в області агентно-орієнтованих підходів до автоматизації [7, 37].

Однією з найбільших проблем при такому підході є децентралізація контролю та, зокрема, питання про те, як стимулювати окремих агентів, щоб вони працювали разом для досягнення мети системи. Основна проблема полягає в тому, що розробник системи повинен вирішити дві основні проблеми призначення винагороди: структурні і тимчасові винагороди. Перший агент, який повинен отримати винагороду (або штраф) за роботу системи, а другий агент - за ключову дію на ключовому тимчасовому кроці, що відповідає за рішення задачі [28,37].

Питання про нагородження було ретельно вивчене в ході навчання агентів для вирішення завдання проектування квадрокоптера [28]. Проблема структурного нагородження також привернула увагу і була вирішена з використанням двох широких підходів: формування зворотного зв'язку і організаційних структур. Формування зворотного зв'язку направлено на

формування мети системи таким чином, щоб дії агентів, які оптимізують локальні цілі, приводили до бажаних результатів на системному рівні [38]. Організаційні структури розподіляють самих агентів на ролі, що забезпечують узгоджену поведінку [39].

Одна з специфічних областей вивчення проблеми присвоєння кредиту зосереджена на забезпеченні відповідності цілей агентів завданням системи (тобто, що добре для агента, добре для системи), а також на те, щоб цілі системи були чутливі до дій агентів [28]. Надання агентам цілей, які відповідають цим двом властивостям, призводить до вирішення проблеми, в якій неявно враховуються ключові взаємодії між агентами. Успіх підходу, заснованого на відмінності цілей при розробці відповідних завдань для тренування агентів, дозволяє припустити, що даний метод можна застосовувати при проектуванні, де існує проблема структурного нагородження при розробці окремих компонентів .

Простий генетичний алгоритм (ГА) зазвичай містить три основних оператора: оператор генерації рішення, оператор мутації і оператор вибору. Вони використовуються на початковому наборі можливих рішень для генерації нових рішень і збереження рішень, які показують покращення результату. Однак, ГА потребує налаштування, щоб мати можливість вирішувати великі завдання багато-агентного пошуку для розподіленої оптимізації. Однією з таких модифікацій є коеволюція, коли кілька груп населення розвиваються одночасно для розробки політики агентського взаємодії .

В результаті агенти розвиваються для того, щоб добре співпрацювати з широким колом інших агентів, а не для того, щоб розвиватися, формуючи оптимальні партнерські відносини. Таким чином, крім спроб зменшити складність процесу навчання, дослідження в області коеволюції спрямовані на досягнення оптимальної , а не стабільної політики [33, 37].

В останні роки в області навчання з підкріпленням (НП) [40] були досягнуті значні результати. Це призвело до різкого збільшення числа вдалих

застосувань і методів. У недавній роботі вивчалися не тільки сценарії з одним агентом , але і сценарії навчання за участю кількох агентів. Деякі результати свідчать про успіхи в складних областях навчання за участю кількох агентів, проте існує ряд проблем, які необхідно вирішити.

Однією з основних проблем, що стоять перед НП і традиційним машинним навчанням в цілому, є необхідність розробки якісних функцій для навчання.

Глибоке машинне навчання (ГМН) забезпечує ефективне представлення процесу навчання, що дозволяє автоматично визначати властивості системи. В останні роки ГМН успішно застосовується в різних областях, таких як комп'ютерний зір та обробка природної мови [41,42]. Одним з ключових аспектів глибокого навчання є використання нейронних мереж (ШНМ), які можуть знаходити компактні визначення в багатовимірних даних [42].

У процесі глибокого машинного навчання з підкріпленням (ГМНП) [43,44], глибокі нейронні мережі навчаються знаходити оптимальну політику і цільову функцію. Таким чином, глибока ШНМ, яка виступає в якості апроксиматора функції, забезпечує потужне узагальнення. Одним з ключових переваг ГМНП є те, що воно дозволяє НП масштабуватися для вирішення проблем з великим виміром станів і просторів дії. Однак більшість існуючих успішних впроваджень ГМНП до цих пір працювали в візуальних областях (як наприклад, ігри Atari [45]), і багато ще належить зробити, щоб адаптувати ці рішення до реально існуючих прикладних завдань, таким як розробка САПР, що є метою цієї роботи.

2.4. Приклади діючих САПР. Програмний пакет SolidWorks

Огляд літератури дозволяє зробити висновок, що етап еволюції САПР, на якому підвищення ефективності проектування досягалося виключно за рахунок властивостей самих систем проектування, закінчився.

CALS-технології мають на увазі створення єдиного інформаційного простору для всіх етапів життєвого циклу ВС НО [13,14,15].

Ця ідея зародилася і була успішно реалізована в військово-промисловому комплексі США і отримала назву CALS (Computer Aided Logistics Support) – автоматизоване проектування логістичних процесів.

Після того як концепція CALS-технологій стала успішно застосовуватися і для проектування виробів цивільного призначення, цей термін стали розшифровувати інакше: Continuous Acquisition and Lifecycle Support - безперервний супровід та підтримка життєвого циклу виробів. У вітчизняній термінології аналогом поняття CALS є поняття «інформаційна підтримка виробництва» (ІПВ).

Вираз Continuous Acquisition означає безперервність інформаційної взаємодії не тільки між проектувальниками, а й зворотний зв'язок із замовником для формалізації його вимог, формування самого замовлення, процесу постачання, тощо. Для забезпечення максимальної ефективності, а також скорочення витрат ресурсів процес взаємодії замовника і постачальника повинен бути безперервним протягом тривалого терміну.

Вираз Life Cycle Support передбачає системність підходу до інформаційної підтримки всіх етапів життєвого циклу виробу.

Оскільки витрати на підтримку складного наукомісткого виробу в працездатному стані часто рівні або навіть помітно перевищують витрати на його придбання, значне скорочення вартості експлуатації виробу забезпечується інвестиціями у створення системи підтримки його життєвого циклу.

CALS-технології забезпечують комплексну комп'ютеризацію всіх сфер промислового виробництва, уніфікацію і стандартизацію специфікацій виробів на всіх етапах їх життєвого циклу. Головні специфікації представлені у вигляді проектної, технологічної, виробничої, маркетингової, експлуатаційної документації.

Показники ефективності впровадження CALS-технологій в промисловості США в цілому складають:

- пряме зменшення витрат на проектування від 10 до 30%;
- скорочення часу розробки виробів в 1,5-2 рази;
- скорочення часу виведення нових виробів на ринок від 25 до 75%;
- зменшення частки дефектів і обсягу конструктивних змін від 23 до 73%;
- зменшення витрат на підготовку проектної документації до 40%;
- зменшення витрат на розробку робочої документації до 30%.

Основний сенс концепції CALS полягає в підвищенні конкурентоспроможності випуску продукції за рахунок ефективного управління інформаційними ресурсами на протязі всього життєвого циклу. Звісно, це призводить до посилення вимог щодо ступеня автоматизації виробничого процесу та необхідності організовувати високорівневу інформаційну взаємодію всіх її учасників [13,14,15].

Фактично саме доступність і однозначність тлумачення отриманих даних є наріжним каменем, на якому стоять CALS-технології. Однак на різних етапах життєвого циклу можуть використовуватися різні САПР, формат файлів конструкторської документації у яких може не збігатися.

Тому для передачі даних використовують універсальні формати, які підтримуються будь-якими сучасними САПР. Наприклад, для обміну графічними даними використовують стандарти IGES і STEP [14].

Формат IGES є стандартом на передачу і обмін графічних даних між різними CALS-системами.

Формат STEP - це міжнародний стандарт ISO 10303, розроблений для полегшення зберігання і обміну всіх типів інформації, яка причетна до проектованого виробу. Передбачається, що з часом відбудеться поступова заміна IGES на STEP.

Сучасні САПР, що забезпечують проектування складного продукту, мають в своєму складі велику кількість різних модулів. При використанні CALS-технологій виникають проблеми, пов'язані з побудовою загальних баз даних та взаємодії САПР, з організацією спільного використання даних при роботі групи проектувальників [13,14,15].

Для вирішення цих проблем були розроблені спеціальні засоби управління проектними даними - системи PDM, які або входять до складу модулів САПР, або мають самостійне значення і можуть функціонувати окремо.

Кожен з учасників життєвого циклу виробу повинен отримувати вільний доступ до розділів всіх необхідних йому даних з єдиного інформаційного простору. Слід звернути увагу, що «єдність» інформаційного простору не означає фізичне розташування всіх даних на одному комп'ютері.

Навпаки, одними з найбільш ефективних CALS-систем вважаються ті, в яких реалізований розподілений принцип зберігання даних. Прикладом такого розподіленого зберігання інформації можуть служити «хмарні» сховища даних.

Хмарне сховище даних - модель онлайн-сховища інформації, в якому дані зберігаються на численних, розподілених у комп'ютерній мережі серверах, що надаються в користування проектувальникам.

Робоча документація проектувальника та дані зберігаються на віртуальних серверах. Однак фізично такі сервери можуть розташовуватися дуже віддалено один від одного географічно, аж до розташування в різних країнах.

Хмарні шлюзи - технологія, яка може бути використана для більш зручного представлення доступу інженера-проектувальника до хмари. За допомогою відповідного програмного забезпечення інформаційний простір хмари може відображатися для клієнта як локальний диск на його персональному комп'ютері. У цьому випадку взаємодія з хмарою стає максимально простою, а при наявності швидкого зв'язку з хмарою інженер

може не помічати, що працює з даними, що фізично зберігаються дуже далеко від нього.

Використання хмарних технологій для реалізації концепції CALS-проектування розширює можливості проектувальників, однак поряд з цим накладає дуже жорсткі рамки на пропускну здатність каналу підключення до Інтернету.

Підвищення ефективності створення і використання складного наукоємного продукту на основі CALS-технологій досягається вирішенням наступних завдань.

По-перше, покращується якість виробів за рахунок більш повного врахування наявної інформації під час проектування та прийняття проектних рішень. Обґрунтованість рішень, що прийняті в автоматизованій системі управління виробництвом (АСУВ), буде вище, якщо особи, які приймають такі рішення, мають оперативний доступ не тільки до бази даних АСУВ, але і до баз даних інших автоматизованих систем. Такий підхід дозволяє оптимізувати календарні плани робіт, зміст заявок, тощо. Під оперативним доступом слід розуміти не просто можливість зчитування інформації з баз даних, а й легкість їх правильної інтерпретації, тобто узгодженість з синтаксису і семантиці з протоколами, прийнятими в АСУВ. Те ж саме відноситься і до інших систем. Наприклад, технологічні підсистеми САПР повинні приймати і правильно обробляти дані, що надходять від підсистем автоматизованого проектування, досить складно, оскільки проектувальники працюють з різними автоматизованими системами.

По-друге, зменшуються витрати на проектування і розробку технічних виробів. Застосування CALS-технологій дозволяє значно скоротити час виконання проектних робіт, завдяки повторному використанню раніше виконаних компонентів і пристроїв, багатьох складових частин обладнання, машин і систем, які проектувались раніше. Вони є доступними будь-якому користувачеві CALS-технологій. Така доступність забезпечується

узгодженістю форматів файлів, способів і керівництв в різних частинах загальної інтегрованої системи.

Для організації взаємодії користувача з САПР розроблені і використовуються геометричні моделі проектованого обладнання. Геометричні моделі випробувального стенду навігаційного обладнання (ВС НО) - це моделі, що відображають геометричні властивості проектованого обладнання. Якщо мова йде тільки про розробку комплекту конструкторської документації (складальних і робочих креслень), то проектувальник оперує двовимірними геометричними моделями. Але в САПР більш високого рівня можлива робота як з двовимірними, так і з тривимірними моделями проектованого обладнання. Як приклад можна привести широко поширені програмні пакети SolidWorks, T-FLEX і КОМПАС-3D [8,9,10].

Серед тривимірних геометричних моделей розрізняють каркасні (дротяні), поверхневі та об'ємні (твердотільні).

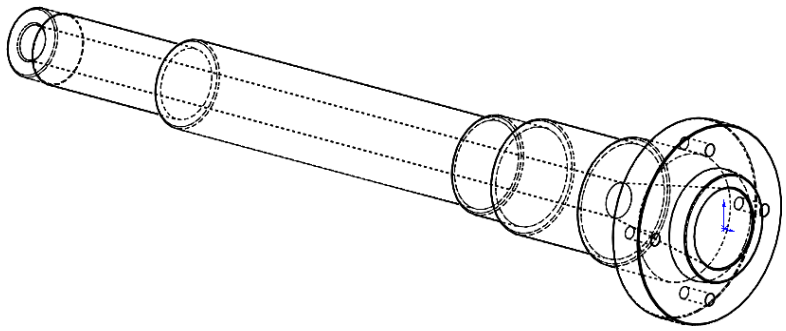
Каркасна геометрична модель являє форму деталі у вигляді якоїсь сукупності ліній, що лежать на поверхнях (ребрах) деталі. Для кожної лінії, що утворює модель, відомі координати характерних точок і вказана їх інцидентність ребер або поверхнях. Інцидентність - поняття, яке використовується для вираження геометричного розташування ребра і вершини геометричної моделі проектованого ВС. Так, якщо *v1, v2* - вершини, а *e = (v1, v2)* - з'єднуюче їх ребро, тоді вершини *v1, v2* і ребро *e* інцидентні. Дві вершини (так само, як і два ребра) інцидентними бути не можуть.

Працювати з каркасної моделлю на подальших кроках процесу проектування незручно, тому каркасні моделі використовують нечасто. Однак іноді розробники САПР спеціально передбачають примусовий перехід до каркасних моделей, щоб підкреслити тип режиму проектування, в якому працює користувач. Так, наприклад, при роботі в програмному пакеті SolidWorks при низхідному проектуванні (тобто коли створюється складальний вузол, під геометрію якого підлаштовуються деталі) деталь, з якої

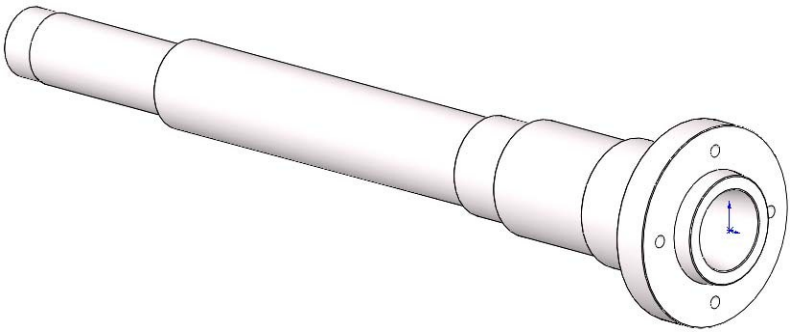
працює користувач, відображається твердотільною, а всі інші елементи збірки - каркасними.

Об'ємні геометричні моделі відрізняються від поверхневих тим, що в них в явній формі містяться відомості про належність елементів внутрішньому чи зовнішньому (по відношенню до проектованої деталі) простору.

На рис. 2.6 наведені приклади каркасної (рис. 2.6, а) і твердотільної (рис. 2.6, б) моделі вала ВС, виконаної в програмі SolidWorks (зовні поверхнева модель вала не відрізнялася б від твердотільної, тому відображати її на рис. 2.6 немає сенсу).



а)



б)

Рис. 2.6. Каркасна (а) і твердотільна (б) модель вала ВС

Розглянуті моделі відображають тіла з замкнутими контурами. Деякі САПР допускають оперування тілами з незамкненими контурами. Такий підхід використовується, наприклад, при моделюванні тіл, що стосуються один одного в якійсь одній точці або вздовж прямої. Моделі тіл з незамкнутим об'ємом використовуються в процесі проектування на проміжних етапах, коли

необхідно працювати одночасно з тривимірними і двовимірними моделями без товщини стінок конструкції.

Найважливішим поняттям, пов'язаним з геометричним моделюванням в САПР, є параметризація.

Різні проектувальники САПР можуть абсолютно по-різному інтерпретувати поняття параметризації. Так, за матеріалами [11, 12, 13] під параметричних моделюванням можуть розуміти:

1) збереження історії створення моделі з можливістю зміни вихідних даних;

2) визначення геометричних залежностей між елементами креслення і моделі;

3) визначення змінних параметрів моделі і креслення, з можливістю їх подальшого зміни.

Розміри, що визначають конкретний геометричний об'єкт і які становлять варійовані параметри моделі проектованого виробу, можуть бути двох видів: іменованими (ідентифікованими) і не іменованими. Іменований розмір має деякий символьний або символічно-цифровий ідентифікатор, синтаксис якого дозволяє звертатися до нього через редактор параметризації САПР. Оскільки такий параметризований розмір пов'язаний з відповідною змінною моделі проектованого об’єкта, його часто називають змінною моделі (або просто змінної).

Привласнюючи цим змінним різні числові значення, можна істотно змінювати геометричну форму виробу, що моделюється, розміри та інші його властивості, що залежать від форми і розмірів. Таким чином, основна, відмінність параметризованого геометричного елемента від звичайного, полягає у великих можливостях модифікації створеного параметризованого елемента і скорочення часу, що витрачається на його модифікацію.

У сучасних САПР механізм параметризації дозволяє управляти положенням деталі в складальної одиниці за рахунок зміни параметрів прив'язок. У свою чергу, це дає можливість ставити в одній моделі збірки різні

положення деталей, які вони приймають в процесі функціонування, тобто моделювати руху деталей при роботі виробу.

Основною особливістю параметричної технології зміни положення деталей, в порівнянні з простою зміною параметрів прив'язок, є те, що в процесі роботи складального вузла високого рівня можуть взаємопов'язано змінювати своє положення деталі з різних складальних одиниць нижчого рівня.

Використання параметричних змінних підвищує продуктивність праці інженера-конструктора. Якщо розміри, визначаючі просторове положення деталей, пов'язані з відповідними змінними, то міняти положення деталей можна, просто варіюючи цей розмір, причому змінюватися він буде відразу у всіх моделях збірок.

Розрізняють декілька основних різновидів параметризації. Таблична параметризація передбачає створення таблиці параметрів типових деталей. В цьому випадку створення нової модифікації проектованої деталі проводиться шляхом вибору з таблиці типу розмірів, на підставі якої САПР автоматично генерує геометричну модель. Таблична параметризація, хоча і дозволяє істотно спростити створення бібліотек стандартних елементів, але успішно застосовується лише при проектуванні нескладного виробу.

Ієрархічна параметризація базується на історії всіх геометричних перетворень, що проводяться з деталлю. Її суть полягає у тому, що в ході побудови моделі вся послідовність проектних операцій відображається в окремому вікні у вигляді дерева проектування. У дереві перераховані всі існуючі в моделі допоміжні елементи, матеріали і виконані операції над деталлю.

Стосовно до проектування складальних вузлів у САПР зберігаються всі відомості про ієрархію окремих елементів, наприклад, у вигляді послідовності складання → підзбірки → деталі.

Варіаційна (розмірна) параметризація заснована на побудованні ескізів з наступним накладенням на об'єкти ескізу різних параметричних зв'язків, а

також заданням користувачем обмежень. Ці обмеження являють собою систему рівнянь, що однозначно визначають взаємозв'язок між параметрами. Геометрична параметризація має на увазі, що геометрія кожного параметричного об'єкта, що проектується перераховується в залежності від положення і параметрів батьківських (основних) об'єктів. У цьому випадку модель ВС НО складається з елементів побудови і елементів зображення. Елементи побудови задають параметричні зв'язки і можуть бути виконані як конструкторські лінії. Елементи зображення поділяються на лінії зображення (ними обводятся конструкторські лінії) і елементи оформлення (розміри, штрихування і т.д.).

Відомі три підходи до створення параметризованих геометричних моделей проектованого ВС:

1. Параметричне конструювання з повним і неповним набором зв'язків.

2. Використання асоціативної геометрії.

3. Об'єктно-орієнтоване моделювання з використанням базових операцій додавання (видалення) матеріалу.

Стосовно до параметричного конструювання прийнято розрізняти два його різновиди:

1. З жорстким (повним) набором зв'язків.

2. З неповним набором зв'язків (м'яка).

САПР SolidWorks призначена для автоматизації робіт з проектування та виробництва виробів будь-якого ступеня складності і призначення на етапах підготовки технологічної документації. Спеціалізовані підсистеми САПР вирішують завдання на етапах виробництва та проектування.

В основі SolidWorks лежить ядро Parasolid. Основний акцент робиться на зручність роботи і звичний для користувача інтерфейс, що повністю відповідає стандартам Windows, а також локалізацію системи. Русифікована версія SolidWorks являє собою програмний комплекс САПР, що дозволяє проектувати вироби будь-якої складності без обмеження кількості елементів

збірки і випускати креслення, специфікації та інші документи відповідно до вимог ЕСКД.

Програмний пакет SolidWorks [8] дозволяє вирішувати такі завдання на етапі конструкторської підготовки проектної документації (КППД):

1. Тривимірне проектування виробів високого ступеня складності з урахуванням специфіки виготовлення (базові конфігурації SolidWorks).

2. Створення конструкторської документації в суворій відповідності з ДСТУ (базові конфігурації SolidWorks, DraftSight, SWR-Специфікація).

3. Дизайн (базові конфігурації SolidWorks).

4. Реверсивний інжиніринг (SolidWorks Premium).

5. Проектування комунікацій (електроджгутів і об'ємний монтаж-SolidWorks Premium, SWR-Електрика; трубопроводи - Solid Works Premium, SolidWorks Routing).

6. Інженерний аналіз на міцність, теплопередачу, частотний аналіз, лінійне і нелінійне наближення - SolidWorks Simulation; динаміка механізмів - SolidWorks Motion; газо та гідродинаміка - SolidWorks Flow Simulation; оптика та світлотехніка - Optis Works.

7. Аналіз розмірних ланцюгів (SolidWorks Premium).

8. Управління даними і процесами на етапі КППД (SWE-PDM).

На етапі технологічної підготовки проектної документації (ТППД) САПР SolidWorks дозволяє вирішувати наступні завдання:

1. Аналіз конструкцій виробу (базові конфігурації SolidWorks, DFMXpress / DFMProfessional).

2. Аналіз процесів виготовлення (лиття пластмас - SimpoeWorks, аналіз процесів штампування - BlankWorks та FastForm).

3. Розробка технологічних процесів за ЕСКД, включаючи матеріальне нормування (SWR-Технологія).

4. Проектування засобів технологічного оснащення (базові конфігурації SolidWorks, MoldWorks, Electrode Works, Logopress і ін.).

5. Розробка керуючих програм для верстатів з числовим програмним управлінням - фрезерна, токарна, токарно-фрезерна і електроерозійна обробка - CAMWorks; лазерне, плазмове і гідроабразивне різання, вирубні штампи - cncKad; координатно-вимірювальні машини - CMMWorks.

6. Управління даними і процесами на етапі ТППД (SWE-PDM).

Як приклад можливостей програмного комплексу на рис. 2.7 показаний фрагмент тривимірної моделі збірки вала ВС НО, виконаної в рамках цієї роботи. При розробці моделі використовувалася технологія висхідного проектування.

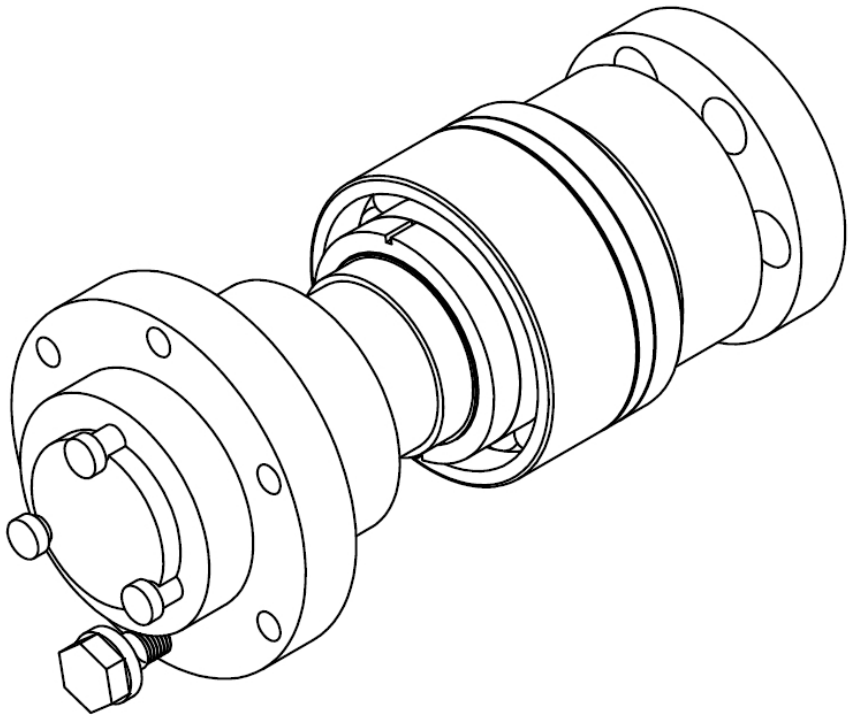


Рис. 2.7. Тривимірна модель збірки вала ВСНО, виконана в програмі SolidWorks

Система управління проектними даними SolidWorks Enterprise PDM в складі програмного комплексу САПР SolidWorks дозволяє формувати єдиний інформаційний простір підприємства, забезпечуючи одночасну розробку виробу і технологій виготовлення, управління робочою документацією, повторне використання даних, автоматизацію проектних процесів, підготовку даних для САПР управління ресурсами підприємства та багато іншого.

В процесі розробки виробу в програмному пакеті Solid Works забезпечуються:

1. Розробка декількох різновидів виробу.

2. Повноцінне відновлення з історії розвитку виробу на будь-який етап розробки.

3. Запозичення даних між проектами.

4. Сповіщення учасників процесу підготовки виробництва про зміну компонентів проекту.

5. Кероване оновлення пов'язаних документів.

У сховищі даних SWE-PDM користувач може працювати з розділами:

1. Оперативного доступу з розташованими в розробці документами;

2. Затвердженої документації із затвердженими документами та оригіналами;

3. Нормативно-довідкової документації;

4. Стандартних та покупних виробів;

5. Бібліотек будь-якого виду.

Особливу зручність при роботі з програмою SolidWorks для користувача представляє використання бібліотечного модуля ToolBox, в якому розташовані параметризовані моделі різних стандартизованих об'єктів.

Програма SWE-PDM автоматично, вручну або комбіновано створює структуру виробу, веде історію її зміни, синхронізує зміни документів, працює з документами через структуру виробу і навпаки. Можлива робота з документами, зі структурою виробів або в змішаному режимі.

Модуль SWR-Специфікації створює конструкторські специфікації за ЕСКД і може:

• Працювати з моделями SolidWorks або структурою виробу SWE PDM;

• Обмінюватися даними з моделями та кресленнями SolidWorks;

• Випускати групові специфікації за ДСТУ.

Програмний модуль SWE-PDM працює з документами будь-яких типів, електронними та паперовими, з файлами будь-яких форматів. Має можливість

повноцінного перегляду файлів більше 250 форматів в робочому вікні SWE-PDM без установки додаткового ПЗ. Можливе використання цифрового підпису, контролю унікальності імен файлів і значень атрибутів документів.

Програмний комплекс SolidWorks дозволяє працювати над одним проектом з територіально рознесених майданчиків. На кожному майданчику організовується своє сховище оперативного доступу, на кожному сервері зберігаються метадані. Сервери спілкуються через мережу Інтернет і синхронізуються автоматично за розкладом або по вимогам. При тимчасовому відключенні від головного сервера можлива автономна робота з раніше отриманими і знову створюваними файлами. Можлива повноцінна робота через Інтернет за допомогою WEB-доступу до виділеної частини сховища. Підтримуються VPN-канали та шифровані протоколи (SSL, HTTPS).

Система SWE-PDM включає наступні модулі:

- SWE-PDM CAD Editor - робоче місце конструктора, робота з САПР-системами SolidWorks, AutoCAD, Inventor, Pro / Engineer і Wildfire, Solid Edge і з файлами будь-яких типів;
- SWE-PDM Contributor - робоче місце користувача, робота з різними документами, включаючи файли AutoCAD. Дозволяє переглядати САПР-файли;
- SWE-PDM Contributor / WEB - робоче місце WEB-доступу. Дозволяє працювати з будь-якими документами, включаючи САПР-файли, і брати участь в процесах Workflow;
- SWE-PDM Viewer - робоче місце користувачів з доступом до документів тільки для читання і участю в процесах Workflow;
- SWE-PDM / ECAD - модуль отримання даних з САПР призначених для проектування електроніки;
- DraftSight - редактор файлів у форматі DWG, створених в AutoCAD або інших системах. Підтримує файли будь-яких версій формату DWG;

- eDrawings Professional - засіб перегляду файлів ряду САПР-систем (SolidWorks, AutoCAD, Pro / Engineer) і анотування САПР-документів. Поставляється в складі комплексу SWE-PDM.

Дуже важливою особливістю програми SolidWorks є можливість додавання модулів, розроблених незалежно за допомогою так званого інтерфейсу SolidWorks API.

Програмний засіб SolidWorks API - це інтерфейс прикладного програмування, що дозволяє розробляти профільні програми на платформі САПР SolidWorks. Цей інтерфейс постачається в складі базової конфігурації SolidWorks і, крім API-функцій, містить динамічні бібліотеки типів і констант.

Програма SolidWorks містить сотні API-функцій, які можна викликати з програм Microsoft Visual Basic, VBA (Microsoft Excel, Word, Access і т.д.), Microsoft Visual C, C ++, .NET або з файлів-макросів SolidWorks. API-функції забезпечують прямий доступ до функціональних можливостей пакета SolidWorks.

Використання інтерфейсу API стає особливо актуальним при вирішенні завдань інтеграції різнорідних додатків на рівні єдиної інформаційної системи САПР ВС НО.

Можливості програмного пакета SolidWorks не обмежуються однією лише підготовкою конструкторської документації. Використання програмного модуля SolidWorks Simulation дозволяє виконувати найрізноманітніші розрахунки на міцність досліджуваних конструкцій.

Модуль SolidWorks Simulation заснований на методі кінцевих елементів і існує в трьох конфігураціях: SolidWorks Simulation; SolidWorks Simulation Professional; SolidWorks Simulation Premium.

Основні функціональні особливості модуля Simulation для різних конфігурацій показані в табл. 2.1. [8].

Таблиця 2.1. Можливості САПР Solidworks

Функції	SolidWorks Simulation	SolidWorks Simulation Professional	SolidWorks Simulation Premium
Розрахунок збірок	√	√	√
Розрахунок зварних конструкцій	√	√	√
Прогнозування втомного руйнування	–	√	√
Параметрична оптимізація	–	√	√
Розрахунок резонансних частот	–	√	√
Розрахунок навантажень втрати стійкості	–	√	√
Тепловий розрахунок без урахування руху середовища	–	√	√
Імітації падіння	–	√	√
Розрахунок трубопроводів під тиском	–	√	√
Лінійна динаміка: модальний аналіз, аналіз випадкових коливань, гармонійний аналіз	–	–	√
Фізична і геометрична нелінійність	–	–	√
Нелінійна динаміка	–	–	√

Відповідно до інформації виробника САПР SolidWorks [8] в програмному модулі SolidWorks Simulation допускаються кілька типів аналізу:

- лінійна статика;
- гіпотези малих і великих переміщень з автоматичною реалізацією приросту навантажень;
- облік зміни жорсткості тонкостінних об'єктів при дії навантажень в площині.

Які аналізовані об'єкти можуть використовуватись:

- тіла;
- поверхні;
- структурні елементи зварних конструкцій;
- довільні їх комбінації в деталях і збірках;

- віртуальний пружний півпростір.

Контактні умови між об'єктами моделювання можуть задаватися як:

- посадка з натягом;
- вхід і вихід з контакту з ковзанням;
- вхід і вихід з контакту з пружним півпростором;
- реалізація контактів у варіантах вузол до вузла, вузол до поверхні, поверхня до поверхні;
- різноманітні варіанти зв'язування тіл, оболонок і балок в межах як деталі, так і збірки;
- умовно обраний проміжок між тілами і оболонками;
- обраний проміжок між оболонками, побудованими на базі об'єктів з листового металу;
- облік тертя в контактах з ковзанням.

Моделюватися можуть матеріали:

- ізотропні пружні;
- ортогонально і циліндрично-анізотропні пружні;
- віртуальні абсолютно жорсткі об'єкти.

Можлива генерація наступних типів сіток кінцевих елементів при створенні моделі:

- сітка просторових скінченних елементів - тетраедрів з лінійним і параболічних полем переміщень;
- сітка кінцевих елементів оболонок - криволінійні трикутники з лінійним і параболічних полем переміщень;
- сітка кінцевих елементів стрижнів, балок, а також балок з довільною комбінацією моментних та безмоментних варіантів зв'язку у вузлах;
- довільна комбінація в одній моделі тіл, оболонок, балок з різними варіантами взаємодії;

- адаптивні сітки з автоматичним ущільненням або підвищенням порядку базової функції.

Результатами розрахунку можуть бути:

- переміщення, деформації, напруження в довільній ортогональній або циліндричній системах координат;
- сили реакції у всіх деталях в довільному напрямку;
- контактні напруги і контактні сили;
- зусилля в віртуальних болтах і штифтах;
- запас міцності за різними критеріями міцності для тіл, оболонок і з'єднувачів у вигляді болтів і штифтів;
- ізолінії і ізоповерхні, значення в точці, інтегральні величини на геометричних об'єктах Solidworks, епюри функції вздовж крайок;
- збереження деформованого вигляду тіл та оболонок як моделей, обмежених тріангуляцією;
- формування звітів, які можливо налаштувати;
- діаграма з рекомендацією щодо оптимізації матеріалу.

2.5. Багато-агентна система автоматизованого проектування випробувальних стендів навігаційного обладнання

В процесі розробки ВС його складна функціональна структура призводить до виникнення безлічі завдань, що вимагають застосування різних інструментів САПР. Для вирішення специфічних вимог до проектування ВС повинні бути розроблені додаткові програмні середовища. Це досягається застосуванням багато-агентної САПР ВС, де агенти (А) або інтелектуальні агенти (ІА) відіграють вирішальну роль, виконуючи роль посередника між людиною і інструментами САПР загального призначення [53].

Згідно стандарту VDI 2206 (Методологія проектування мехатронних систем) описана структура загальної мехатронної моделі ВС. Вона складається з агентів, призначених для роботи з САПР загального призначення для проектування механічних конструкцій, електроніки та системи управління:

(A_1) – агент виконавчої частини (електроніки), яка складається з електричних приводів, системи кодування положення, системи передачі сигналу в обладнання для випробувань.

(A_2) – агент механічної частини, що складається з карданового підвісу та інших опорних конструкцій стенду – агент (A_3). Разом з агентом опорних підшипників (A_4) це утворює підсистему САПР механічної частини, що разом с іншим обладнанням підлягають спільному проектуванню.

(A_5) агент для випробувального стола, термокамери, або іншими приладами для додаткових умов випробування. Конструкція випробувального столу повинна забезпечувати можливість установки і підключення навігаційного обладнання і повинна мати всі необхідні механічні, електричні та інші відповідні сполучні вузли. (A_6) Агент системи управління і підтримки підсистем енергозабезпечення випробувального столу (електричної, гідравлічної, пневматичної і т.д.). (A_7) Агент з проектування робочого місця оператора. (A_8) Агент з розробки програмного забезпечення середовища

проведення випробувань та середовища керування, моніторингу, перевірки, налаштування випробувального стенду. Середовище має забезпечувати всі вимоги програми випробувань навігаційного обладнання та мати можливість калібрування та перевірки придатності випробувального стенду та відповідність стенду специфікаціям виробника — програма самодіагностики.

Наведений вище список не є вичерпним, він може бути розширений до декількох десятків різних агентів A_N, загальною кількістю N, що складають комплекс САПР ВС. Основний елемент системи САПР ВС називається агентом.

Проектування на першому етапі починається з верхнього рівня агентів та прямує на рівні підсистем, що підпорядковані за ієрархією.

В цьому дослідженні основна увага сконцентрована на двох верхніх рівнях агентів (1) та (2). Інші підсистеми можуть бути спроектовані раніше ніж (1) та (2), однак такі обмеження накладають суттєві проблеми та унеможливлюють проектування оптимальних підсистем (1) та (2). Основні обмеження та параметри проектування (2) використовуються як обмеження при проектуванні (3) та (4). Рівень (5) проектується з використанням обмежень (2) – (4).

Проектування оптимальної системи керування складає окреме комплексне завдання досліджень та виходить за межі цієї роботи. Рівні проектування (6), (7), (8) та інші підсистеми нижчого рівня мають проектуватись тільки з використанням параметрів обмежень (2). Оптимальне проектування програмного середовища для комплексу випробувального стенду навігаційного обладнання потребує окремих засобів, методології та інструментів дослідження, що не висвітлюються в даній роботі.

САПР проектування випробувальних стендів навігаційного обладнання (Рис.2.8) складається з набору агентів і загальних інструментів проектування САПР (ІП). Будемо називати ІП - підсистемами проектування.

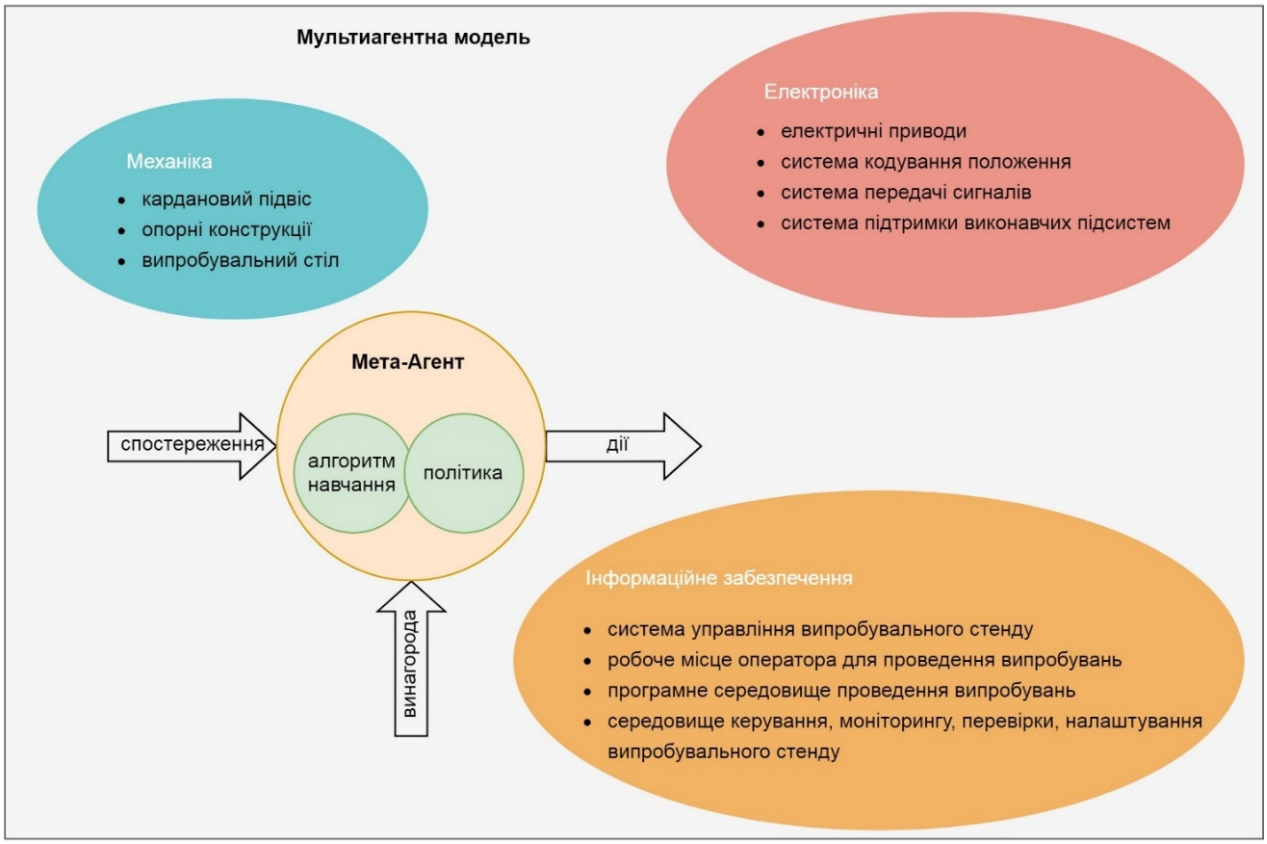


Рис. 2.8. Модель багатоагентної системи проектування ВС

Агенти перетворюють інформацію для підсистем проектування ВС, отримують завдання від мета-агента і відправляють назад результати. Кожен агент управляє одним або декількома підсистемами проектування, але контроль ІП не може бути розподілений між кількома агентами.

Агент або інтелектуальний агент (ІА) - це автономна сутність, вміщена в середовище проектування, який обробляє інформацію, отриману через сприйняття. Агент може керувати ним за допомогою дій і діяти відповідно до його станів для досягнення своїх цілей, які визначаються винагородою. що здатна контролювати його і діяти відповідно до його стану для досягнення поставлених завдань. Агент обробляє отриману інформацію за допомогою навчання, досліджує рішення відповідно до правил своєї політики і вибирає краще рішення, співпрацюючи з іншими агентами в середовищі проектування ВС.

Агенти взаємодіють через мета-агента. Мета-агент - виконує певні підзавдання, які агенти можуть виконувати для інших агентів. Дії мета-агента – це вибір між низкою проектних рішень з урахуванням ряду мір переваг або властивостей, це процес визначення найкращих значень або комбінації змінних проектних рішень, таким чином, щоб спроектована система була коректною щодо обмежень, а ефективність системи максимізована щодо численних, можливо, суперечливих цілей. Мета-агент приймає евристичні рішення – це поєднання попереднього вибору та компромісного рішення. Процес прийняття евристичного рішення відрізняється тим, що має елементи ШІ.

Таким чином САПР ВС складається з:

• *M*, мета-агент,

• *A*, набір агентів,

• *G*, набір підсистем проектування *ІП*.

• *T* - набір завдань, які виконує мета-агент. $T = \{T_1, ..., T_n\}$, де *n* - кількість різних завдань в системі САПР.

• *ST*, набір підзавдань, які виконують агенти. $ST = \{ST_1,..,ST_m\}$, де *m* - кількість різних підзавдань в системі САПР.

• *D* показує залежність між агентами та підсистемами проектування *ІП*.

• *S* - це архітектура системи. Це набір залежностей між підсистемами проектування.

• *R* - це вимоги до вихідної інформації для кожної підсистеми проектування.

Позначимо САПР символом $\mathrm{AI_{CAD}}$. Таким чином САПР:

$$\mathrm{AI_{CAD}} = \langle M_T(p_s, a_s), A_{ST}, G, D, S_R \rangle,$$

де $G = \{G_1, ..., G_p\}$, *p* – кількість підсистем проектування в системі САПР. Стан підсистем проектування *G* визначається p_s. Якщо $p_s = 0$, підсистема проектування не може виконувати підзавдання. Якщо $p_s = 1$, підсистема проектування працює, вона може виконувати підзавдання, якщо має достатньо вихідної інформації;

$A = \{A_1,...,A_a\}$, a - кількість агентів в системі САПР. Активний стан агента A визначається змінною a_s. Якщо $a_s = 0$, агент не може виконувати завдання. Якщо $a_s = 1$, агент працює та може виконувати завдання.

Основним елементом для визначення ІА є його логіка, яка може бути математично визначена як функція.

Функція агента визначається як S → A, а сприйняття як S → P,

де $S = \{s_1, s_2, ..., s_t\}$ - інформація з середовища проектування, $A = \{a_1, a_2, ..., a_n\}$ - дії агента, таким чином щоб:

$$IA = \langle S, A, R, P, \rho_0 \rangle,$$

де S це набір всіх дійсних станів,

A це набір всіх дійсних дій,

$R: S \times A \times S \to \mathbb{R}$ це функція винагороди, де $r_t = R(s_t, a_t, s_{t+1})$,

$P: S \times A \to \mathcal{P}(S)$ є функцією ймовірності переходу, де $P(s'|s, a)$ ймовірність переходу в стан s' якщо почати перехід із стану s та виконати дію a, і ρ_0 це розподіл початкового стану.

Рассел і Норвіг [51] визначили п'ять основних класів інтелектуальних агентів, які втілюють принципи, що лежать в основі майже всіх інтелектуальних систем, і засновані на ступенях їх сприйняття, інтелекту і можливостей. До цих класів відносяться:

(1) прості агенти, які отримують команди і реагують безпосередньо на навколишнє середовище відповідно до заздалегідь визначених правил;

(2) агенти, засновані на моделях, які інтерпретують отриману інформацію для визначення своїх дій;

(3) агенти, засновані на цілях, які є розширенням агентів, заснованих на моделях, що здатні визначати цілі під час прийняття рішень;

(4) агенти, засновані на корисності, які можуть взаємодіяти з іншими агентами, координувати потік інформації і вирішувати ситуації, коли цілі знаходяться в конфлікті;

(5) агенти, що навчаються.

Далі надається порівняння теорії ігор та процесу проектування ВС (Таблиця 2.2.).

Таблиця 2.2. Елементи теорії ігор для проектування ВС

ТЕОРІЯ ІГОР	ПРОЕКТУВАННЯ ВИПРОБУВАЛЬНОГО СТЕНДУ
ІТЕРАЦІЙНА ГРА	Процес проектування ВС
ГРАВЕЦЬ	Команда інженерів-проектувальників та процедури аналізу та синтезу САПР
ЗАВДАННЯ ГРАВЦЯ	Багатодисциплінарна модель проектування ВС
ВИНАГОРОДА (ВИТРАТИ)	Критерії оптимізації

Теорія ігор - це математична основа, яка моделює і аналізує ситуації, в яких взаємодіють кілька сутностей, які приймають рішення [54]. Використовуючи стратегічні рішення, кожен учасник гри намагається досягти своїх цілей.

Гра - це кортеж $\langle N, (A_i)_{i \in N}, (u_i)_{i \in N} \rangle$, де:

N - безліч гравців-агентів,

A_i - безліч дій гравця *i*, де A: $=\{ a \parallel a = (a_i)_{i \in N}, a_i \in A_i, \forall_i \in N\}$,

u_i: $A \rightarrow R$ - функція винагороди агента *i*, яка може бути прибутком (максимізація) або витратами (мінімізація).

Існує кілька типів ігор та відповідних математичних формулювань. Два основні типи в теорії ігор - це теорія некооперативної і кооперативних ігор. У кооперативних іграх гравці вимагають оптимальних узгоджених дій або розумних правил розподілу винагород, які роблять коаліції стабільними, а гравцям дозволяється спілкуватися і отримувати окремі винагороди, щоб діяти як єдине ціле, покращуючи своє становище в грі. Теорія некооперативної ігор може бути використана для аналізу стратегій процесів прийняття рішень незалежними гравцями, що мають конфліктні завдання. У цих іграх, беручи до уваги дії інших учасників, кожен гравець оптимізує свою функцію корисності автоматично без координації стратегічного вибору кожного гравця. Рівновага

Неша є одним з найбільш важливих понять рішення в теорії ігор. У рівновазі Неша жоден гравець не може збільшити свої винагороди, змінивши в односторонньому порядку свою стратегію, враховуючи, що дії інших гравців фіксовані. Таким чином, досягається взаємні оптимальні дії всіх гравців. Рівновага Неша в некооперативної іграх - це оптимальне рішення, коли немає відносин лідер-послідовник, і кожен гравець конкурує з іншими, щоб максимізувати свою функцію корисності. У кооперативних іграх завдання досягаються, коли набір стратегій може максимізувати винагороду кожного гравця.

Ця робота розширює інтеграцію теорії ігор у багатоагентну систему [54]. У схемі мультиагентного оптимального проектування якщо використовується однорівневий підхід, гра є спільною, оскільки агентні завдання поєднуються в одну однорівневу проблему. Якщо використовується багаторівневий підхід, тоді агенти повинні бути визначені статусом у грі-проектуванні. Можливі види ігор та взаємодії визначаються мета-агентом:

• кооперативна: кожна команда ІА співпрацює та представляє інформацію інших команд.

• некооперативна: кожна команда ІА повинна робити припущення щодо інших команд,

• лідер: ІА вирішує спочатку, що домінує в процесі проектування, припускаючи, що послідовники поводяться раціонально,

• послідовник: ІА або чекає на іншого ІА, або домінує інший.

У системі САПР використовується мета-агент, що використовує дві моделі, обидві - глибокі нейронні мережі, одна з яких називається «Актор», а інша – «Критик». Завдання моделі Критика - навчитися оцінювати, чи призвела дія, що виконав Актор, до покращення нагороди з навколишнього середовища чи ні, і дати цю інформацію Актору. Критик передає Актору дійсне число, яке вказує на оцінку дії, здійсненої в попередньому стані s_{s-1}. Порівнюючи цей рейтинг, отриманий від Критика, Актор може порівняти

свою поточну політику з новою політикою і вирішити, як він хоче поліпшити себе, щоб зробити кращі дії.

Метою навчання з підкріплення є навчання агента виконувати завдання в умовах невизначеності. Агент отримує спостереження та винагороду від оточення та надсилає дії до оточення. Нагорода - це міра того, наскільки успішною є дія щодо виконання цілі завдання.

Агент містить два компоненти: політику та алгоритм навчання. Політика - це відображення, яке вибирає дії на основі спостережень із навколишнього середовища. Як правило, політика - це апроксиматор функцій із налаштовуваннями параметрів, наприклад, глибока нейронна мережа. Алгоритм навчання постійно оновлює параметри політики на основі дій, спостережень та винагород. Метою алгоритму навчання є пошук оптимальної політики, яка максимізує очікувану сукупну довгострокову винагороду, отриману під час виконання завдання.

Висновки до розділу 2

В цьому розділі було визначено структуру системи автоматизованого проектування ВС. Для цього загальні вимоги до конструкції випробувальних стендів було сформульовано у вигляді завдання мультидисциплінарного оптимального проектування випробувального стенду навігаційного обладнання. Дисципліни в загальному значенні було перетворено на багато-агентну систему автоматизованого проектування випробувальних стендів навігаційного обладнання.

Було розроблено інтелектуальний підхід для реалізації САПР на основі типової структури ВС, включаючи механіку, електротехніку та електроніку, програмне забезпечення; спираючись на існуючий досвід в побудові систем гібридного інтелектуального проектування.

Було розроблено метод проектування на базі навчання з підкріпленням для вирішення завдання багатокритеріального прийняття рішень.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ ДО РОЗДІЛУ 2

1. D. Nau, S. Gupta, and W. Regli, “AI Planning Versus Manufacturing-Operation Planning: A Case Study,” *Proc. of IJCAI-95 Conf.*, 1995.

2. S.J.J. Smith, K. Hebbar, D.S. Nau, I. Minis, “Integrating Electrical and Mechanical Design and Process Planning,” in: *Knowledge Intensive CAD,* M. Mäntylä, S. Finger, T. Tomiyama (eds). Boston, MA: Springer, 1997, pp 269-288.

3. P. O'Grady, R. E. Young, A. Greef & L. Smith, “An advice system for concurrent engineering”, *International Journal of Computer Integrated Manufacturing*, vol. 4, no. 2, pp. 63-70, 1991, doi: 10.1080/ 09511929108944481.

4. D. Su and M. Wakelam, “Evolutionary optimization within an intelligent hybrid system for design integration,” *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 13, no. 5, pp. 351-363, 1999, doi:10.1017/S0890060499135054.

5. D. Su and M. Wakelam, “Intelligent hybrid system for integration in design and manufacture,” *Journal of Material Processing Technology*, vol. 76, no. 1, pp. 23-28, 1998.

6. D. Su, M. Wakelam, K. Jambunathan, “Integration of a knowledge-based system, artificial neural networks and multimedia for gear design,” *Journal of Materials Processing Technology*, vol. 107, pp. 53-59, 2000, doi: 10.1016/S0924-0136(00)00716-0.

7. *Multiagent Systems*, (edited by Gerhard Weiss), 2nd edition. The MIT Press, 2013

8. “Solutions to Get From Concept to Manufacturing”. SOLIDWORKS. [Електронний ресурс]. Режим доступу: https://www.solidworks.com/solutions (дата звернення 19.05.2020).

9. “Топ Системы”. T-FLEX PLM. [Електронний ресурс]. Режим доступу: https://www.tflex.ru (дата звернення 19.05.2020).

10. “САПР КОМПАС”. КОМПАС-3D. [Електронний ресурс]. Режим доступу: https://kompas.ru (дата звернення 19.05.2020).

11. Л. Н. Андреев, Д. Е. Бортяков, С. В. Мещеряков, *Системы автоматизированного проектирования*. СПб.: Изд-во СПбГТУ, 2002.

12. В. Н. Бельков, В. Л. Ланшаков, *Автоматизированное проектирование технических систем, уч. пос*. М.: Академия естествознания, 2009.

13. П. Ю. Волков, *Системы автоматизированного проектирования технологического оборудования машиностроительных производств*. Пенза: ПГУ, 2013.

14. С. И. Пестрецов, *CALS-технологии в машиностроении: основы работы в CAD/CAE-системах*. Тамбов: Изд-во ГОУ ВПО ТГТУ, 2010.

15. Л. Г. Доросинский, О. М. Зверева, *CALS-технологии*. М.: LAP Lambert Academic Publishing, 2014.

16. Р.А. Литвинов, С.В. Лукинских, В.А. Шилов, “Проектирование рабочих клетей прокатных станов в SolidWorks,” *САПР и графика*, №8, с. 87-90, 2009.

17. X. F. Zha, “Artificial Intelligence and Integrated Intelligent Systems in Product Design and Development,” in *Intelligent Knowledge-Based Systems*, C.T. Leondes (eds). Boston, MA: Springer, 2005, pp. 1067-1123.

18. V. Akman, P. J. V. ten Hagen and P. J. Veerkamp (Eds.). *Intelligent CAD systems II: implementational issues*. Springer-Verlag Berlin Heidelberg, 1989.

19. C. H. Dagli (Ed.), *Intelligent systems in design and manufacturing*, ASME Press, 1994.

20. H. Zarefar, and J. H. Goulding, “Neural Networks in Design of Products: A Case Study”, in *Intelligent Design and Manufacturing*, (ed. A. Kusiak), John Wiley, 1992, pp. 179–201.

21. A. Bahrami and C.H. Dagli, “From fuzzy input requirements to crisp design,” *International journal of Advanced manufacturing Technology*, vol. 8, no. 1, pp. 52–60, 1993, doi: 10.1007/BF01756637.

22. D. C. Powell, M. M. Skolnick and S. Tong, “EnGENEous Domain IndependentMachine Learning for Design Optimization,” *in Proc. of the 3rd Int. Conf. on Genetic Algorithms (ICGA)*, pp. 151-159, 1989.

23. A. Bahrami, M. Lynch and C.H. Dagli, “Integrating Product and Process Design Through Intelligent Systems”, in *Intelligent Systems in Design and Manufacturing*, (eds. C.H. Dagli and A. Kusiak), ASME Press, 1994, pp. 237-259.

24. J. Kisielnicki, *Virtual Technologies: Concepts, Methodologies, Tools, and Applications,* (3 Vol.). Poland, Warsaw: Information Science Reference, 2008.

25. C. Chen, L. G. Occea, “Soft computing for concurrent product design evaluation under a blackboard environment,” *3rd Int. Conf. on Engineering Design and Automation (EDA 99)*, (Vancouver, Canada), 1999.

26. X. Zha, “Soft computing framework for intelligent human-machine system design, simulation and optimization,” *Soft Comput*., vol. 7, pp.184-198, 2003, doi: 10.1007/s00500-002-0196-4.

27. X. F. Zha, S. Y. E. Lim, W. F. Lu, “A Knowledge Intensive Multi-Agent System For Cooperative/Collaborative Assembly Modeling And Process Planning,” *Journal of Integrated Design and Process Science*, vol.7, no.1, pp. 99-122, 2003.

28. D. Hulse, K. Tumer, C. Hoyle, I. Tumer, “Modeling multidisciplinary design with multiagent learning,” *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 33, no. 1, pp. 85-99, 2019.

29. K.M. Saridakis, A.J. Dentsoras, “Soft computing in engineering design – A review”, *Advanced Engineering Informatics*, vol. 22, no. 2, pp. 202-221, 2008.

30. X. F Zha, *Artificial intelligence and integrated intelligent information systems : emerging technologies and applications*. Hershey, PA: IGI Global, 2007

31. G. Castellano, C. Castiello, A.M. Fanelli, L. Jain, “Evolutionary Neuro-Fuzzy Systems and Applications,” in *Advances in Evolutionary Computing for System Design. Studies in Computational Intelligence*, L.C. Jain, V. Palade, D. Srinivasan (eds), vol 66. Springer Berlin Heidelberg, 2007

32. S. Gunawan, A. Farhang-Mehr, S. Azarm, “Multi-level Multi-objective Genetic Algorithm Using Entropy to Preserve Diversity,” in *Evolutionary Multi-Criterion Optimization*, C. M. Fonseca, P. J. Fleming, E. Zitzler, L. Thiele, K. Deb (eds), vol 2632. Springer Berlin Heidelberg, 2003.

33. Y. Jin and W. Li, “Design Concept Generation: A Hierarchical Coevolutionary Approach,” *J. Mech. Des*, vol. 129, no. 10, pp. 1012-1022, 2007, doi: 10.1115/1.2757190.

34. H. P. Nii, “The Blackboard Model of Problem Solving and the Evolution of Blackboard Architectures,” *AI Magazine*, vol. 7, no. 2, pp. 38-53, 1986, doi: 10.1609/aimag.v7i2.537.

35. L. D. Erman, F. H.-Roth, V. R. Lesser, D. R. Reddy, “The Hearsay-II Speech-Understanding System: Integrating Knowledge to Resolve Uncertainty,” B. L. Webber, N. J. Nilsson (eds.), *in Readings in Artificial Intelligence*, Morgan Kaufmann, 1981, pp. 349-389.

36. Joaquim R. R. A. Martins; Andrew B. Lambe, “Multidisciplinary Design Optimization: A Survey of Architectures,” *AIAA Journal*, vol. 51, no. 9, pp. 2049-2075, 2013.

37. N. Soria, M. K. Colby, I. Y. Tumer, C. Hoyle and K. Tumer, “Design of Complex Engineered System Using Multiagent Coordination,” *ASME J. Comput. Inf. Sci. Eng.*, vol. 18, no. 1, 2017.

38. P. Stone and M. Veloso, “Multiagent Systems: A Survey From a Machine Learning Perspective,” *Auton. Robots*, vol. 8, no. 3, pp. 345–383, 2000.

39. S. V. Albrechta and P. Stone, “Autonomous agents modelling other agents: A comprehensive survey and open problems,” *Artificial Intelligence*, vol. 258, pp. 66-95, 2018.

40. L. Busoniu, R. Babuska and B. De Schutter, “A Comprehensive Survey of Multiagent Reinforcement Learning,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 2, p. 156-172., 2008, doi: 10.1109/TSMCC.2007.913919.

41. Y. LeCun, Y. Bengio, G. Hinton, “Deep learning,” *Nature*, vol. 521, no 7553, pp. 436-444, 2015.

42. J. Schmidhuber, “Deep learning in neural networks: An overview,” *Neural Networks*, vol 61, pp 85-117, 2015.

43. K. Arulkumaran, M. P. Deisenroth, M. Brundage and A. A. Bharath, “Deep Reinforcement Learning: A Brief Survey,” *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26-38, 2017, doi: 10.1109/MSP.2017.2743240.

44. T. T. Nguyen, N. D. Nguyen and S. Nahavandi, “Deep Reinforcement Learning for Multiagent Systems: A Review of Challenges, Solutions, and Applications,” *IEEE Transactions on Cybernetics*, vol. 50, no. 9, pp. 3826-3839, Sept. 2020, doi: 10.1109/TCYB.2020.2977374.

45. V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, M. Riedmiller. “Playing Atari with Deep Reinforcement Learning,” *NIPS Deep Learning Workshop*, 2013.

46. I. Graessler, and J. Hentze, “The new V-Model of VDI 2206 and its validation,” *Automatisierungstechnik*, vol. 68, no. 5, pp. 312-324, 2020, doi:10.1515/auto-2020-0015.

47. *Smart manufacturing - Reference architecture model industry 4.0 (RAMI4.0)*, IEC PAS 63088:2017, 2017.

48. “Alignment Report for Reference Architectural Model for Industrie 4.0. Intelligent Manufacturing System Architecture.” [Електронний ресурс]. Режим доступу: https://www.plattformi40.de/PI40/Redaktion/DE/Downloads/Publikation/hm-2018-manufactoring.html (дата звернення 24.03.2021).

49. *Design Methodology For Mechatronic Systems*, VDI 2206:2004, Verlag des Vereins Deutscher Ingenieure, 2004.

50. K. Lewis, “An Algorithm for Integrated Subsystem Embodiment and System Synthesis,” Georgia Institute of Technology, Atlanta, Georgia, USA, NASA Contractor Rep. 201732, 1997.

51. S. Russell, P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed. New Jersey: Upper Saddle River, 2010.

52. M. Weißenberger, “Optimierung der Bewegungsdynamik von Werkzeugmaschinen im rechnergestützten Entwicklungsprozess,” Zugl.: München, Techn. Univ., Diss., 2001.

53. S. O. Dolhorukov, “Navigation equipment test table computer-aided design artificial intelligence configuration optimization,” *Věda a perspektivy*, vol. 1, no. 1, pp. 214-227, May, 2021

54. V. M. Sineglazov and S. O. Dolgorukov, “Computer-aided Design of Navigation Equipment Test Table based on Game Optimization,” *in Proc. IEEE 6th Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, (Kyiv, Ukraine), 2020, pp. 7-10, doi: 10.1109/MSNMC50359.2020.9255419.

РОЗДІЛ 3.
ІНТЕГРАЦІЯ КОМПЛЕКСУ ПРОГРАМНИХ БІБЛІОТЕК ТА САПР В АВТОМАТИЗОВАНИЙ ПРОЦЕС РОЗРОБКИ ВИПРОБУВАЛЬНИХ СТЕНДІВ ЗА ДОПОМОГОЮ ШТУЧНОГО ІНТЕЛЕКТУ

3.1. Мета-евристичні методи багатокритеріальної оптимізації

За останні кілька десятиліть для цілей оптимізації було розроблено багато різних метаевристичних алгоритмів, кожен з яких має певні переваги та недоліки завдяки численним застосуванням у різних сферах оптимізації. Такими алгоритмами є оптимізація складних завдань з обмеженнями математичного та інженерного проектування з використанням оптимізації на основі хаотичних ігор [1], в яких методологія конфігурації фракталів гри в хаос розглядаються поряд з проблемами самоподібності фракталів. метаевристичний алгоритм для вирішення завдань інженерної оптимізації заснований на людській діяльності в живій черзі [2]. Далі в цьому параграфі розглянемо детально метаевристичний алгоритм оптимізації заснованій на поведінці орлів.

Це новий метод оптимізації на основі популяції, який називається Aquila Optimizer (AO) [3], який натхненний поведінкою орлів (Aquila) в природі під час процесу лову здобичі. Отже, процедури оптимізації запропонованого алгоритму AO представлені чотирма методами; вибір простору пошуку високим злетом з вертикальним нахилом, дослідження в розбіжному просторі пошуку контурним польотом з коротким нападом ковзання, експлуатація в просторі пошуку сходяться низьким польотом з атакою повільного спуску та нахили при ходьбі та схопленні здобичі Для перевірки здатності нового оптимізатора знаходити оптимальне рішення для різних задач оптимізації проведено набір експериментальних серій. Наприклад, під час першого експерименту AO застосовується для пошуку розв’язку відомих 23 функцій. Друга та третя експериментальні серії мають на меті оцінити ефективність AO,

щоб знайти рішення для більш складних задач, таких як тридцять тестових функцій CEC 2017 та десять тестових функцій CEC 2019, відповідно. Нарешті, використовується набір із семи реальних інженерних проблем. З експериментальних результатів АО, які порівняно з відомими мета-евристичними методами, спостерігається перевага розробленого алгоритму АО.

АО це популяційний метод, правило оптимізації починається з популяції рішень-кандидатів (X), як представлено в рівнянні (3.1), яка генерується стохастично між верхньою межею (UB) і нижньою межею (LB) даного завдання. На кожній ітерації в оптимального рішення приблизно визначається найкраще на даний момент рішення.

$$X_{ij} = r_1 \times (UB_j - LB_j) + LB_j, i = 1,2, \dots \dots, Nj = 1,2, \dots, Dim \quad (3.1)$$

де X позначає безліч поточних рішень-кандидатів, які генеруються випадковим чином, X_i позначає значення (позиції) рішень, N - загальна кількість рішень-кандидатів (популяція), Dim позначає розмірність завдання, LB_j позначає j-ю нижню межу, а UB_j - j-ю верхню межу даного завдання.

У першому методі (X_1) орел розпізнає область здобичі і вибирає кращу область полювання, злітаючи вгору з вертикальною стійкою. Тут АО широко досліджує простір пошуку з високого злету, щоб визначити область простору пошуку, де знаходиться здобич. Математично ця поведінка представлена в рівнянні (3.2).

$$X_1(t+1) = X_{best}(t) \times \left(\frac{1-t}{T}\right) + (X_M(t) - X_{best}(t) * rand), \quad (3.2)$$

де, $X_1(t + 1)$ - рішення наступної ітерації t, яке генерується першим методом пошуку (X_1). $X_{best}(t)$ - найкраще рішення, отримане до t-ї ітерації, яке відображає приблизне місце здобичі. Це рівняння (1 використовується для управління розширеним пошуком (розвідкою) через кількість ітерацій. $X_M(t)$ позначає середнє значення місцеположення поточних рішень, пов'язаних на t-й ітерації, $rand$ - випадкова величина між 0 і 1. t і T - поточна ітерація і максимальне число ітерацій, відповідно.

$$X_M(t) = \frac{1}{N}\sum_{i=1}^{N} X(t), \forall j = 1,2,\dots,Dim \quad (3.3)$$

де *Dim* - розмірність задачі, а *N* - кількість рішень-кандидатів (розмір популяції).

У другому методі (X_2), коли область здобичі знайдена з високого польоту, орел кружляє над метою здобичі, готується підлетіти до землі, а потім атакує. Цей метод називається контурним польотом з короткою глісадною атакою. Тут АО вузько досліджує обрану ділянку цільової здобичі, готуючись до атаки. Математично це поведінка представлено в рівнянні (3.4).

$$X_2(t+1) = X_{best}(t) \times Levy(D) + X_R(t) + (y - x) * rand, \quad (3.4)$$

де $X_2(t + 1)$ - рішення наступної ітерації *t*, яке генерується другим методом пошуку (X_2). *D* - простір розмірності, а *Levy(D)* - функція розподілу *Levy*, яка обчислюється за допомогою рівняння (3.5). $X_R(t)$ - випадкове рішення, взяте в діапазоні [1 *N*] на першій ітерації.

$$Levy(D) = s \times \frac{u \times \sigma}{|\upsilon|^{\frac{1}{\beta}}}, \sigma = \left(\frac{\Gamma(1+\beta) \times sin(\frac{\pi\beta}{2})}{\Gamma(\frac{1+\beta}{2}) \times \beta \times 2^{(\frac{\beta-1}{2})}}\right) \quad (3.5)$$

де *s* - постійна величина, фіксована на *0,01*, u і υ - випадкові числа від *0* до *1*.

$$y = r \times cos(\theta), x = r \times sin(\theta) \quad (3.6)$$

де *p* - постійна величина, фіксована на *1,5*. У рівнянні (3.6) для надання форми спіралі в пошуку використовуються *y* і *x*, які розраховуються наступним чином.

$$r = r_1 + U \times D_1, \theta = -\omega \times D_1 + \theta_1, \theta 1 = \frac{3 \times \pi}{2} \quad (3.7)$$

r_i приймає значення від 1 до 20 для фіксованої кількості циклів пошуку, а *U* - мале значення, фіксоване на 0,00565. D_i - цілі числа від 1 до довжини простору пошуку (*Dim*), а *w* - невелике значення, фіксоване на 0,005.

У третьому методі (X_3), коли область здобичі точно визначена, і орел готовий до посадки і атаки, орел спускається вертикально з попередньою атакою, щоб виявити реакцію здобичі. Цей метод названий низьким польотом з повільною атакою спуску. Тут АО використовує обрану область мети, щоб

наблизитися до здобичі і атакувати. Математично ця поведінка представлено в рівнянні (3.8).

$$X_3(t+1) = (X_{best}(t) - X_M(t)) \times \alpha - rand + ((UB - LB) \times rand + LB) \times \delta, \quad (3.8)$$

де $X_3(t + 1)$ - рішення наступної ітерації t, яке генерується третім методом пошуку (X_3). $X_{best}(t)$ означає приблизне місце розташування здобичі до *i*-ї ітерації (найкраще отримане рішення), а $X_M(t)$ означає середнє значення поточного рішення на *t*-й ітерації, яке розраховується за допомогою рівняння (3.3). *rand* - випадкова величина між 0 і 1. *a* і 5 - параметри налаштування експлуатації, фіксовані в даній роботі на малому значенні (0,1). *LB* означає нижню межу, а *UB* - верхню межу даного завдання.

У четвертому способі (X_4), коли орел наблизився до здобичі, він атакує здобич над землею відповідно до його стохастичних рухів. Цей метод називається "ходити і хапати здобич". Тут, і нарешті, АО атакує здобич в останньому кроці. Математично ця поведінка представлена в рівнянні (3.9).

$$X_4(t+1) = QF \times X_{best}(t) - (G_1 \times X(t) \times rand) - G_2 \times Levy(D) + rand \times G_1, \quad (3.9)$$

де $X4(t + 1)$ - рішення наступної ітерації *t*, яке генерується четвертим методом пошуку (X_3). QF позначає функцію якості, яка використовується для врівноваження стратегій пошуку, яка розраховується за допомогою рівняння (3.10). G_1 позначає різні рухи орла, які використовуються для відстеження здобичі під час втечі, яке генерується за допомогою рівняння (3.11). G_2 являє собою значення від 2 до 0, які позначають нахил польоту орла, використовуваного для стеження за здобиччю під час втечі від першого місця розташування (*1*) до останнього місця розташування (*t*), яке генерується за допомогою рівняння (3.12). $X(t)$ - поточне рішення на *t*-й ітерації.

$$QF(t) = t^{\frac{2 \times rand - 1}{(1-T)^2}} \quad (3.10)$$

$$G_1 = 2 \times rand - 1 \quad (3.11)$$

$$G_2 = 2 \times \left(1 - \frac{t}{T}\right) \quad (3.12)$$

QF(t) - значення функції якості на *t*-й ітерації, а *rand* - випадкова величина між 0 і 1. *t* і *T* - поточна ітерація і максимальне число ітерацій, відповідно. *Levy (D)* - функція розподілу польоту, розрахована за допомогою рівняння (3.5).

В алгоритмі АО оптимізація починає процедури поліпшення рішення з генерації випадкового визначеного набору рішень-кандидатів, званого популяцією. Через траєкторію повторення, пошукові стратегії АО досліджують розумні позиції близького до оптимального рішення або найкращого отриманого рішення. Кожне рішення оновлює свої позиції відповідно до найкращих рішень, отриманих в результаті процесів оптимізації АО. Щоб підкреслити рівновагу між стратегіями пошуку АО (тобто розвідка та експлуатація), існують чотири стратегії пошуку для розвідки і експлуатації (тобто розширена розвідка, звужена розвідка, розширена експлуатація і звужена експлуатація). Нарешті, процес пошуку ОА завершується при досягненні критерію закінчення.

3.2. Застосування інтелектуальних агентів для системи автоматизованого проектування випробувального стенда

Модель середовища для прийняття рішень агента (СПРА) це, $\langle S, A, R, P, \rho_0 \rangle$ (Рис. 3.1), де

S це набір всіх дійсних станів,

A це набір всіх дійсних дій,

$R: S \times A \times S \to \mathbb{R}$ це функція винагороди, де $r_t = R(s_t, a_t, s_{t+1})$,

$P: S \times A \to \mathcal{P}(S)$ є функцією ймовірності переходу, де $P(s'|s, a)$ ймовірність переходу в стан s' якщо почати перехід із стану s та виконати дію a,

і ρ_0 це розподіл початкового стану.

СПРА означає, що система підпорядковується правилу, що переходи залежать тільки від останнього виконаного стану і дії, а не від попередньої історії дій, станів, тощо.

Рис. 3.1. Модель взаємодії агента і середовища проектування

Головними елементами навчання з підкріпленням (НП) є агент і середовище. Навколишнє середовище - це світ, в якому знаходиться агент і з яким він взаємодіє. На кожному етапі взаємодії агент бачить (можливо, часткове) спостереження за станом світу, а потім вирішує, яку дію зробити. Навколишнє середовище змінюється, коли агент діє на нього, але може також змінюватися саме по собі.

Агент також отримує сигнал нагороди з навколишнього середовища, число, яке говорить йому, наскільки добрий або поганий поточний стан середовища. Метою агента є максимізація його сукупної винагороди. Методи та алгоритми НП - це способи, за допомогою яких агент може вивчитись певній поведінці для досягнення своєї мети.

Стан *s* це повний опис стану світу. Не може бути ніякої інформації про світ, яка була б прихована від стану. Спостереження *o* - це частковий опис стану, в якому може бути пропущена деяка інформація.

Коли агент може спостерігати за повним станом навколишнього середовища, ми говоримо, що навколишнє середовище повністю спостерігається (обсервується). Коли агент може отримати тільки часткове спостереження, ми говоримо, що навколишнє середовище спостерігається частково.

Різні середовища дозволяють виконувати різні дії. Набір всіх допустимих дій в даному середовищі часто називається простором дій. Деякі середовища, мають дискретні простори дій, в яких агенту доступне тільки кінцеве число ходів. В інших середовищах агент має безперервний простір дій. У безперервних просторах дій дії являють собою вектори з реальними значеннями.

Ця різниця визначає застосування різних методів в глибинному НП. Деякі сімейства алгоритмів можуть бути безпосередньо застосовані тільки в одному випадку і повинні бути істотно перероблені для іншого.

Політика - це, по суті, інтелект агента, правило, яке використовується агентом для прийняття рішення про те, які дії слід зробити. Воно може бути детермінованим, і в цьому випадку зазвичай позначається як μ:

$$a_t = \mu(s_t),$$

або воно може бути стохастичним, і в цьому випадку зазвичай позначається як π:

$$a_t \sim \pi(\cdot\,|s_t).$$

У глибинному НП ми маємо справу з параметризованими політиками: політиками, результати яких є обчислюються функціями, залежними від набору параметрів (наприклад, ваги і зміщення нейронної мережі), які ми можемо налаштувати для зміни поведінки за допомогою деякого оптимізаційного алгоритму.

Параметри такої політики позначаються θ або ϕ:

$$a_t = \mu_\theta(s_t)$$

$$a_t \sim \pi_\theta(\cdot\,|s_t).$$

Два найбільш поширених види стохастичної політики в глибинному НП - категорична політика та діагональна політика Гауса.

Категоричні політики можуть використовуватися в просторах дискретних дій, в той час як діагональні політики Гауса використовуються в просторах безперервних дій.

Два ключових обчислення центрально важливі для використання і навчання стохастичної політики:

- вибірка дій з політики,
- та обчислення функції логарифмічної правдоподібності конкретних дій, $\log \pi_\theta(a|s)$.

Категорична політика схожа на класифікатор дискретних дій. Нейронна мережа будується для категоричної політики так само, як будується класифікатор: на вході знаходиться спостереження, далі за ним знаходиться кілька шарів (можливо, конволюційних або щільно пов'язаних, в залежності від типу входу), а потім є один останній лінійний шар, який дає логіт для кожної дії, за яким слідує функція softmax для перетворення логіту в ймовірності [4].

Вибірка дій з політики. Маючи ймовірності кожної дії, такі фреймворки, як PyTorch[5] і Tensorflow[4], мають вбудовані інструменти для вибірки.

Позначимо останній шар ймовірностей як $P_\theta(s)$. Це вектор, в якому стільки ж елементів, скільки і дій, тому ми можемо розглядати дії як елементи вектора. Функція логарифмічної правдоподібності для дії a може бути отримана як:

$$\log \pi_\theta(a|s) = \log[P_\theta(s)]_a.$$

Багатовимірний розподіл Гауса або багатовимірний нормальний розподіл описується вектором середніх значень, μ, та матрицею коваріацій, Σ. Діагональний розподіл Гауса є особливим випадком, коли матриця коваріацій має тільки записи по діагоналі. Таким чином її можна записати вектором.

Діагональна політика Гауса завжди має нейронну мережу, яка має на вході спостереження, а на виході середні значення дій, $\mu_\theta(s)$. Існує два різні типові способи запису матриці коваріацій.

Перший спосіб: існує один вектор логарифму стандартних відхилень, $\log\sigma$, що не є функцією стану: $\log\sigma$ це автономні параметри.

Другий спосіб: існує нейронна мережа, яка зіставляє стани із стандартними відхиленнями, $\log\sigma_\theta(s)$. Вона може мати спільні шари із мережею середніх значень дій.

В обох випадках обчислюються логарифми стандартних відхилень. Це пов'язано з тим, що логарифми стандартних відхилень можуть вільно приймати будь-які значення в$(-\infty,\infty)$, в той час як стандартні відхилення не можуть бути від'ємними.

Маючи середні значення дії $\mu_\theta(s)$ та стандартне відхилення $\sigma_\theta(s)$, і вектор z шуму від сферичного гаусового $(z\sim\mathcal{N}(0,I))$, вибірка дії може бути обчислена як:

$$a=\mu_\theta(s)+\sigma_\theta(s)\odot z,$$

де $\odot$ позначає по елементний добуток двох векторів.

Функція логарифмічної правдоподібності k –вимірної дії a, для діагонального розподілу Гауса з середнім значенням $\mu=\mu_\theta(s)$ та стандартним відхиленням $\sigma=\sigma_\theta(s)$, має вигляд:

$$\log\pi_\theta(a|s)=-\frac{1}{2}\left(\sum_{i=1}^{k}\left(\frac{(a_i-\mu_i)^2}{\sigma_i^2}+2\log\sigma_i\right)+k\log 2\pi\right).$$

Траєкторія τ це послідовність станів і дій в середовищі,

$$\tau=(s_0,a_0,s_1,a_1,\ldots).$$

Найперший стан середовища, s_0, випадковим чином вибирається з розподілу стартового стану, що позначається як ρ_0:

$$s_0\sim\rho_0(\cdot).$$

Переходи станів, що відбуваються з середовищем між станом в часі t, s_t, та станом в часі $t+1$, s_{t+1}, регулюються природними законами середовища, і залежать тільки від крайньої дії, a_t. Вони можуть бути або детермінованими,

$$s_{t+1} = f(s_t, a_t)$$

або стохастичними,

$$s_{t+1} \sim P(\cdot\,|s_t, a_t).$$

Дії виконуються агентом відповідно до його політики. Траєкторії також часто називають епізодами або програмами.

Функція винагороди (здобутку) R має вирішальне значення в НП. Вона залежить від поточного стану середовища, дії яка щойно відбулась, та наступного стану середовища:

$$r_t = R(s_t, a_t, s_{t+1}).$$

Часто вона спрощується до функції залежності від поточного стану , $r_t = R(s_t)$, або пари стану-дії $r_t = R(s_t, a_t)$.

Метою агента є максимізація деякого поняття сукупної винагороди по траєкторії, що також називається епізодами. Всі такі випадки позначимо $R(\tau)$.

Одним з видів здобутку є недисконтований здобуток, який являє собою лише суму винагород, отриманих в фіксованому вікні кроків:

$$R(\tau) = \sum_{t=0}^{T} r_t.$$

Інший вид здобутку - сума всіх винагород, отриманих агентом за весь час, але зі знижкою в залежності від того, наскільки давно в часі вони були отримані. Це формулювання винагороди включає коефіцієнт дисконтування $\gamma \in (0,1)$:

$$R(\tau) = \sum_{t=0}^{\infty} \gamma^t r_t.$$

Одним з видів здобутку (винагороди) для вибору оптимальної політики агентів системи САПР може бути модель помилки орієнтації випробувального стенда, рис. 3.2. [6].

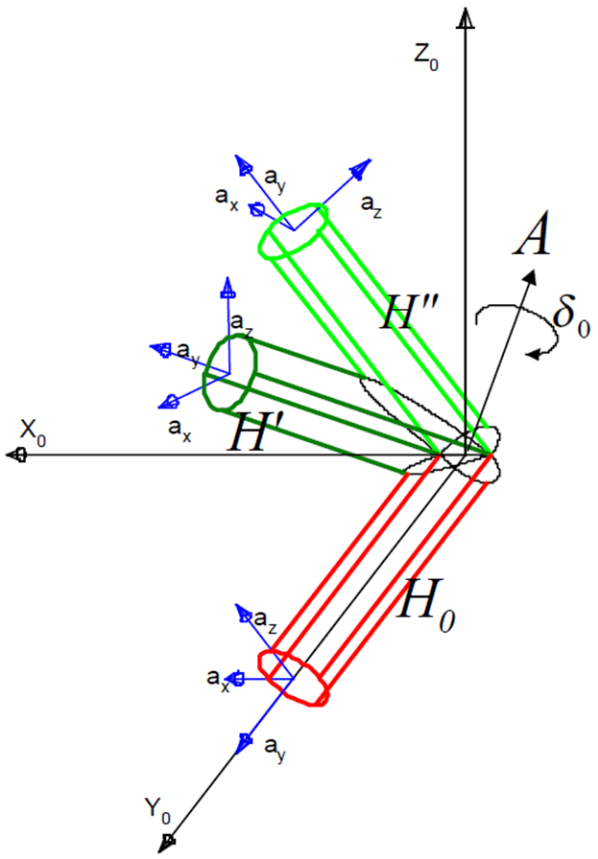


Рис. 3.2. Модель похибки орієнтації випробувального стенда

Як і при будь-якій зміні орієнтації тіла, теорема Ейлера стверджує, що ми завжди можемо знайти вектор A та обертання δ_O, що примусить тіло H'' співпадати з H'. Ми визначаємо кут δ_O, який переміщує тіло навігаційного обладнання, розташоване в орієнтації H'' в орієнтацію H', як похибку орієнтації випробувального стенда. Квадрат величини похибки орієнтації вказано як:

$$\delta_O^2 = \frac{1}{2} Tr\{\delta R^T \delta R\}$$

де δR це похибка матриці обертання (косинуси напрямку), тобто

$$\delta R \equiv R(\Theta) - R(\Theta, \Xi),$$

де $R(\Theta)$ це відповідна матриця косинусів напрямку. Окремі похибки випробувального стенду (ВС) представлені у вигляді набору похибок $\Xi = (K, W, E)$. Набір похибок, зображений на рис. 3.3, є функцією фізичних перекосів (неортогональності), похибок підшипників і похибок датчика

положення сервоприводу випробувального стенду навігаційного обладнання. Ці похибки мають певний геометричний зв'язок з осями випробувального стенду.

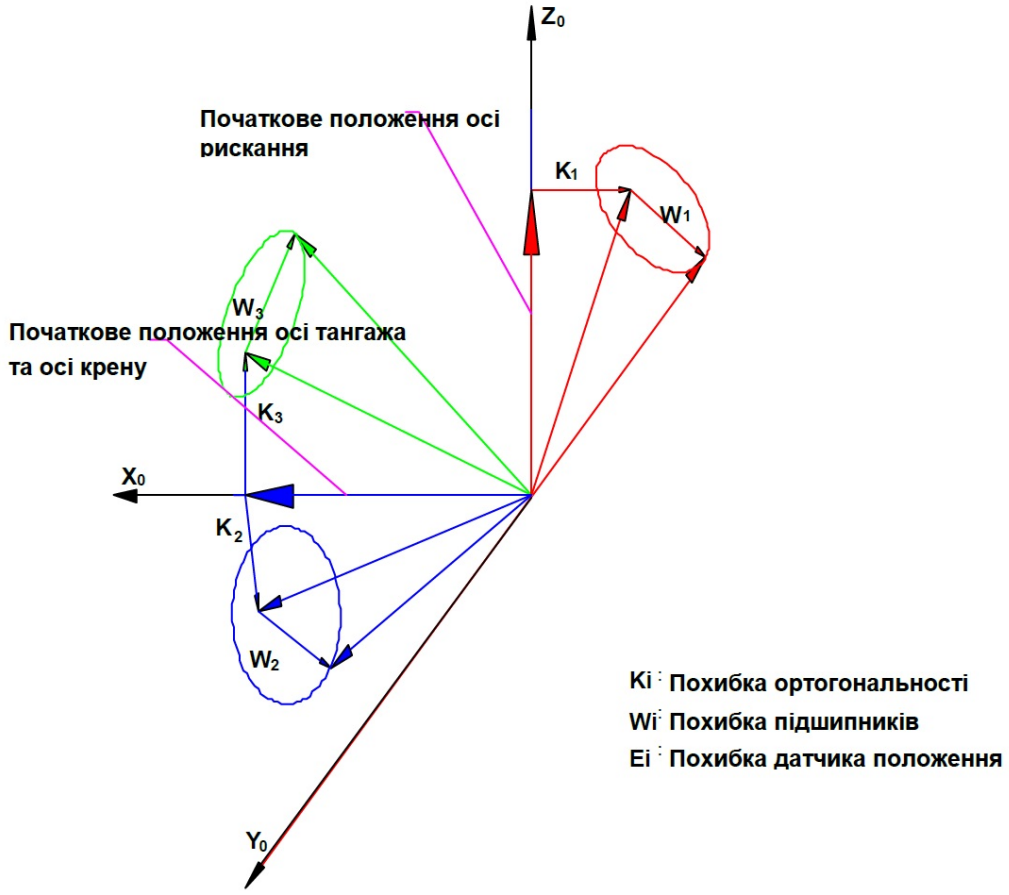


Рис. 3.3. Окремі похибки ВС, що показують їх взаємозв'язок між собою та з набором початкових положень осей ВС

Максимальна похибка орієнтації випробувального стенда:

$$\delta_{Omax} = \max_{\Theta,\Xi}\{\delta_O\}$$

$$\delta_{Omax} = \frac{1}{\sqrt{2}}\sqrt{Tr\{\delta R^T \delta R\}}\Big|_{max}$$

Для обчислення моделі похибки розроблена програмна модель на мові програмування Pyhon, щоб забезпечити її використання у системі проектування випробувального стенду (Рис. 3.4.).

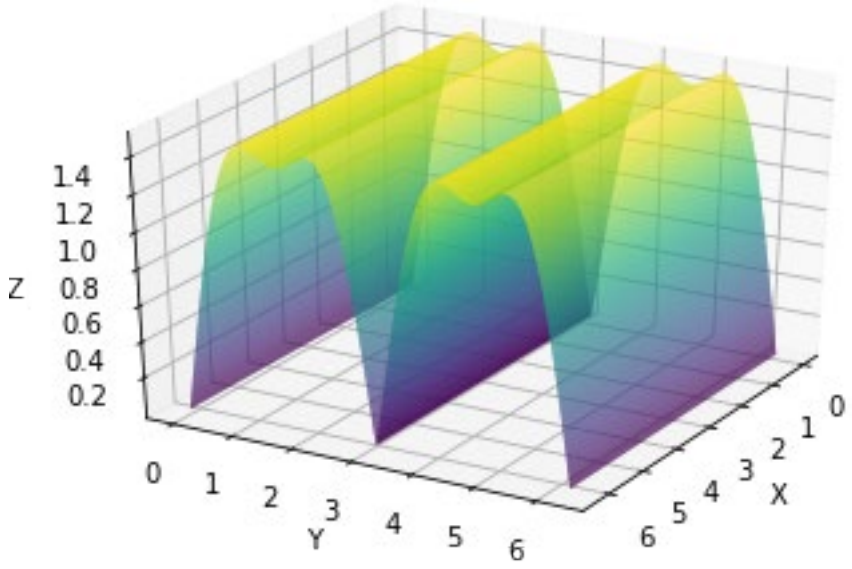


Рис. 3.4. Модель похибки для відхилення ортогональності 1 кутова сек.

Завданням ***навчання агента*** (НА) є побудова політики, яка максимізує очікуваний здобуток. Покладемо, що і переходи станів середовища, і політика є стохастичними. У цьому випадку ймовірність траєкторії з кроків T:

$$P(\tau|\pi) = \rho_0(s_0) \prod_{t=0}^{T-1} P(s_{t+1}|s_t, a_t)\pi(a_t|s_t).$$

Тоді очікуваний здобуток, що позначається $J(\pi)$:

$$J(\pi) = \int_{\tau} P(\tau|\pi)R(\tau) = \underset{\tau\sim\pi}{\mathrm{E}}[R(\tau)].$$

Головне завдання оптимізації НА може бути виражене наступним чином:

$$\pi^* = \arg\max_{\pi} J(\pi),$$

де $\pi^* -$ оптимальна політика.

Значення пари стану-дія – очікуваний здобуток, якщо починати в цьому стані або парі стан-дія, а потім надалі виконувати дії відповідно до певної політики визначається функцією значень здобутку (ФЗЗ).

Використовуються чотири основні ФЗЗ:

1. ФЗЗ за політикою, $V^{\pi}(s)$, яка дає очікуваний здобуток, якщо починати в стані s і потім завжди діяти відповідно до політики π:

$$V^{\pi}(s) = \underset{\tau \sim \pi}{\mathrm{E}}[R(\tau)|s_0 = s]$$

2. ФЗЗ за політикою та дією, $Q^{\pi}(s,a)$, яка дає очікуваний здобуток, якщо починати в стані s, виконати довільну дію a (що може не виходити з політики), і потім завжди діяти відповідно до політики π:

$$Q^{\pi}(s,a) = \underset{\tau \sim \pi}{\mathrm{E}}[R(\tau)|s_0 = s, a_0 = a]$$

3. Оптимальна ФЗЗ, $V^{*}(s)$, яка дає очікуваний здобуток, якщо починати в стані s і завжди діяти відповідно до оптимальної політики в середовищі:

$$V^{*}(s) = \underset{\pi}{max} \underset{\tau \sim \pi}{\mathrm{E}}[R(\tau)|s_0 = s]$$

4. Оптимальна ФЗЗ за дією, $Q^{*}(s,a)$, яка дає очікуваний здобуток, якщо починати в стані s, виконати довільну дію a, і потім завжди діяти відповідно до оптимальної політики в середовищі:

$$Q^{*}(s,a) = \underset{\pi}{max} \underset{\tau \sim \pi}{\mathrm{E}}[R(\tau)|s_0 = s, a_0 = a]$$

Також виникає два ключових зв'язки між функцією ФЗЗ та ФЗЗ за дією:

$$V^{\pi}(s) = \underset{a \sim \pi}{\mathrm{E}}[Q^{\pi}(s,a)],$$

та

$$V^{*}(s) = \underset{a}{max} Q^{*}(s,a).$$

Крім того існує важливий зв'язок між оптимальної ФЗЗ за дією $Q^{*}(s,a)$ та дією, що вибрана оптимальної політикою. За визначенням, $Q^{*}(s,a)$ дає очікуваний здобуток, якщо починати в стані s, виконати довільну дію a, і потім завжди діяти відповідно до оптимальної політики.

Оптимальна політика в s вибере дію, яка максимізує очікуваний здобуток від початку в стані s. В результаті, якщо у нас є Q^{*}, ми можемо безпосередньо отримати оптимальне дію, $a^{*}(s)$, через

$$a^*(s) = \arg\max_a Q^*(s,a).$$

Може бути кілька дій, які максимізують $Q^*(s,a)$. В цьому випадку всі вони оптимальні, і оптимальна політика може випадково вибрати будь-яку з них. Але завжди є оптимальна політика, яка детерміновано вибирає дію.

Всі чотири ФЗЗ підкоряються спеціальним рівнянням самопогодження, що називаються рівняннями Беллмана. Основна ідея рівнянь Беллмана полягає в наступному.

Здобуток відправної точки (стану) - це винагорода, що очікується в цій точці, плюс здобуток в будь якій наступній точці.

Рівняння Беллмана для ФФЗ за політикою наступні:

$$V^\pi(s) = \mathop{\mathrm{E}}_{\substack{a\sim\pi \\ s'\sim P}}[r(s,a) + \gamma V^\pi(s')],$$

$$Q^\pi(s,a) = \mathop{\mathrm{E}}_{s'\sim P}[r(s,a) + \gamma \mathop{\mathrm{E}}_{a'\sim P}[Q^\pi(s',a')]]$$

де $s' \sim P$ скорочене найменування $s' \sim P(\cdot\,|s,a)$, що вказує, що наступний стан s' відбирається з правил переходу середовища; $a \sim \pi$ скорочене найменування $a \sim \pi(\cdot\,|s)$; иа $a' \sim \pi$ скорочене найменування $a' \sim \pi(\cdot\,|s')$.

Рівняння Беллмана для оптимальної ФФЗ наступні:

$$V^*(s) = \max_a \mathop{\mathrm{E}}_{s'\sim P}[r(s,a) + \gamma V^*(s')],$$

$$Q^*(s,a) = \mathop{\mathrm{E}}_{s'\sim P}[r(s,a) + \gamma \max_{a'}[Q^*(s',a')]].$$

Вирішальною відмінністю між рівняннями Беллмана для ФФЗ за політикою та оптимальної ФФЗ є відсутність або наявність функції max над діями. Її включення відображає той факт, що кожний раз, коли агент вибирає свою дію, для того щоб діяти оптимально, він повинен вибирати ту дію, яка призводить до максимального значення здобутку.

Іноді нам не потрібно визначати, наскільки дія є правильною в абсолютному значенні цього слова, а тільки наскільки вона краща за інші в середньому. Тобто, ми хочемо знати відносну перевагу цієї дії. Для цього використовується функція переваги дії (ФПД).

ФПД $A^{\pi}(s,a)$ відповідно до політики π характеризує наскільки краще зробити конкретну дію a в стані s, ніж випадково обрану дію згідно з $\pi(\cdot\,|s)$, за умови, що надалі всі дії будуть виконуватись відповідно до π. ФПД визначається як

$$A^{\pi}(s,a) = Q^{\pi}(s,a) - V^{\pi}(s).$$

Функція переваги має вирішальне значення для методів навчання таких як проксимальний алгоритм оптимізації політики (PPO) [34].

$$\theta_{k+1} = \underset{\theta}{\operatorname{argmax}} \underset{s,a\sim\pi_{\theta_k}}{\mathrm{E}} [L(s,a,\theta_k,\theta)],$$

Навчання виконується максимізуючи цільову функцію агента *L*.

$$L(s,a,\theta_k,\theta) = \min\left(\frac{\pi_{\theta}(a|s)}{\pi_{\theta_k}(a|s)} A^{\pi_{\theta_k}}(s,a),\quad g(\epsilon, A^{\pi_{\theta_k}}(s,a))\right),$$

де

$$g(\epsilon, A) = \begin{cases} (1+\epsilon)A & ; A \geq 0 \\ (1-\epsilon)A & ; A < 0. \end{cases}$$

де ϵ - це гіперпараметр, який приблизно визначає, наскільки далеко нова політика може відхилятися від старої.

Алгоритм навчання агента – це проксимальний алгоритм оптимізації політики (PPO). Він передбачає збір невеликої партії досвіду взаємодії з навколишнім середовищем і використання цієї партії для поновлення політики прийняття рішень. Як тільки політика оновлюється за допомогою цієї партії, досвід відкидається, і збирається нова партія з новою оновленої політикою.

Ключовим внеском PPO є забезпечення того, що нове оновлення політики не надто сильно змінює її в порівнянні з попередньою політикою. Це призводить до меншої дисперсії в навчанні ціною деякого зміщення, але забезпечує більш плавне навчання, а також гарантує, що агент не потрапить у безкінечний цикл навчання через помилково обрані дії.

Алгоритм PPO представлений далі:

1. Вхідні дані: початкові параметри політики θ_0, початкові параметри функції ФПД ϕ_0
2. Визначити параметри: *кроків навчання* (наприклад: 10000), та кількість оновлень на кожній ітерації (наприклад: 10).
3. Виконувати цикл *кроків навчання:*
    a. Зібрати набір траєкторій (дії, стани, нагороди) $\mathcal{D}_k = \{\tau_i\}$ використовуючи стару політику $\pi_k = \pi(\theta_k)$ в середовищі.
    b. Розрахувати поточну винагороду $\hat{R}_t$.
    c. Обчислити функцію переваг, $\hat{A}_t$ на основі поточної функції ФЗЗ V_{ϕ_k}.
    d. Виконувати цикл *оновлень на кожній ітерації*
        i. Визначити поточне значення $\pi_\theta(a_t|s_t)$ шляхом виконання дій з новою політикою π_θ із станом і діями з попередніх кроків.
        ii. Визначити відношення $r(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_k}(a_t|s_t)}$.
        iii. Розрахувати $r(\theta)$* A_k.
        iv. Оновлення політики шляхом максимізації мети за допомогою стохастичного градієнтного сходження викостовуючи Адам:

$$\theta_{k+1} = \underset{\theta}{\operatorname{argmax}} \frac{1}{|\mathcal{D}_k|T} \sum_{\tau \in \mathcal{D}_k} \sum_{t=0}^{T} \min\left(\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_k}(a_t|s_t)} A^{\pi_{\theta_k}}(s_t, a_t), \quad g(\epsilon, A^{\pi_{\theta_k}}(s_t, a_t)) \right),$$

Апроксимуйте функцію ФПД шляхом регресії за середньоквадратичною помилкою за допомогою алгоритму градієнтного спуску:

$$\phi_{k+1} = \underset{\phi}{\operatorname{argmin}} \frac{1}{|\mathcal{D}_k|T} \sum_{\tau \in \mathcal{D}_k} \sum_{t=0}^{T} \left(V_\phi(s_t) - \hat{R}_t \right)^2,$$

4. Стоп.

Вирішення завдань, пов'язаних з багато-агентним навчанням, включають в себе комплекс заходів політики - по одній на кожного агента, - які визначають, як кожен з них повинен діяти, з тим щоб забезпечити максимальну загальну винагороду. У разі проблем скінчених інтервалів можна використовувати дерева політики. Кожне дерево політик представляє собою дерево рішень, де кожен вузол позначений дією, а гілки позначені спостереженнями. Починаючи з кореневого вузла, на кожному кроці кожен агент виконує дію поточного вузла, а потім гілки до підпорядкованих гілок на основі отриманого спостереження.

Однак розмір дерев політик зростає в геометричній прогресії з інтервалом завданнях, що ускладнює введення повноцінних дерев політик у великих завданнях. А коли завдання має нескінченний інтервал, дерева політик більше не можна використовувати для подання рішень. Загальний підхід в цьому випадку полягає в тому, щоб кожен агент підсумовував те, що він знає, використовуючи кінцеву пам'ять, і представляв політики за допомогою контролерів (автоматів) кінцевого стану. Кожен стан контролера являє собою проміжний внутрішній стан пам'яті агента. Починаючи з початкового стану контролера, в кожному стані агент вибирає дію на основі свого внутрішнього стану, а потім на основі отриманого спостереження відгалужується до нового внутрішньому стану. Як вибір дії, так і переходи контролера можуть бути детермінованими або стохастичними; більш високе значення можна отримати за допомогою стохастичних відповідностей, але завдання оптимізації буде складніше.

Такі рішення присвоюють кожному агенту i, локальну політику δ_i, яка представляє собою відповідність з локальної історії спостережень або внутрішньої пам'яті стану на дії. Спільна політика - це кортеж локальних політик, по одній для кожного агента.

$$\delta = \langle \delta_1, \ldots, \delta_n \rangle,$$

Для завдань з скінченними інтервалами з кількістю кроків T значення спільної політики δ з початковим станом s_0:

$$\mathrm{V}^{\delta}(s_0) = E\left[\sum_{t=0}^{T-1} R(\overrightarrow{a_t}, s_t)|s_0, \delta\right]$$

Для завдання нескінченного інтервалу з початковим станом s_0 і коефіцієнтом дисконтування $\gamma \in [0\ ,\ 1)$, значення спільної політики δ становить

$$\mathrm{V}^{\delta}(s_0) = E\left[\sum_{t=0}^{\infty} \gamma^t R(\overrightarrow{a_t}, s_t)|s_0, \delta\right]$$

Вирішення завдань багато-агентного навчання та створення відповідних алгоритмів розглядається в роботах [57-66].

3.3. Існуючі методології в області багато-агентного моделювання та алгоритми навчання з підкріпленням

Теоретичне планування рішень направлено на вибір дій в умовах невизначеності шляхом максимізації очікуваного значення деякого показника ефективності, що називається корисністю. Планування в цьому випадку явно обумовлює невизначеність результатів дій і стану області визначення, спрямоване на оптимізацію корисності, а не на доведене досягнення певних цілей. Наприклад, теоретична проектна пропозиція для дослідницького випробувального стенда навігаційного обладнання може забезпечити максимальну наукову віддачу, вимірювану кількістю або значенням отриманих даних щодо характеристик підсистем, в умовах невизначеності щодо темпів вдосконалення стенда та його ефективності. Теоретичне планування рішень в багато-агентній системі оптимізує одночасно завдання підсистем та координаційні рішення. Значення, пов'язане з кожною дією, засноване на його вплив на область визначення, інформації, яку він передає іншим агентам, і інформації, яку він отримує з області визначення або від інших агентів. Таким чином, єдиний процес планування оптимізує всеосяжну

цінність, оптимізуючи тим самим як дії в області визначення, так і координацію.

Стандартної основою для вирішення проблеми планування в умовах невизначеності є модель середовища для прийняття рішень агента (СПРА)[47]. Модель являє собою область визначення, яка використовує набір станів. Вона призначена для одного агента, що приймає рішення, дії якого призводять до стохастичних переходів до нових станів і винагороді, яка може залежати від дії і результату. Процес прийняття рішень агентом, що частково спостерігається (POMDP) є узагальненням базової моделі, яка враховує недосконалі спостереження. У POMDP агент, який приймає рішення, отримує часткову інформацію про стан середовища після виконання кожної дії. У цьому випадку агент може підтримувати стан переконання (розподіл ймовірностей за станами області визначення) і повинен діяти, не знаючи точного стану середовища. Одне з ключових спостережень, що дозволило вирішити POMDP з одним агентом, полягає в тому, що будь-яка POMDP може розглядатися як POMDP з станами переконання - СПРА, станами області визначення якої є розподіл ймовірностей за станам реального середовища. На жаль, це не так у випадку з кількома агентами , що істотно ускладнює планування.

Більш загальна проблема планування виникає, коли два або більше агента повинні координувати свої дії. Планування в цьому випадку стає особливо складним, коли кожен агент отримує різні спостереження і має різне часткове уявлення про загальну ситуацію. Такі завдання децентралізованого контролю, маю таку особливість, що всі агенти контролюють один процес у співпраці, але кожен з них повинен діяти децентралізовано, ґрунтуючись на своїх власних спостереженнях.

Розповсюдження POMDP для вирішення мультиагентних завдань пропонуються і широко вивчаються з кінця 1990-х років. [48,49,50]

Децентралізований частково спостережуваний процес прийняття рішень (DEC-POMDP) одна с таких моделей для мультиагентних завдань. На

кожному кроці кожен агент робить дію, спільний набір дій викликає стохастичну зміну стану середовища, і на основі дій і їх результатів генерується винагорода. Потім кожен агент отримує власне приватне спостереження, і цикл повторюється. Метою є максимізація кумулятивної (дисконтованої) винагороди за деякий кінцевий горизонт T або за нескінченний горизонт. Модель включає в себе тільки одну функцію винагороди, що вказує на те, що агенти спільно працюють над досягненням однієї мети. Модель DEC-POMDP еквівалентна моделі POMDP з одним агентом. Зв'язок між агентами може бути змодельований як явно, так і неявно. Неявний зв'язок виникає щоразу, коли дії одного агента впливають на спостереження іншого агента. Явний зв'язок - обмін повідомленнями між агентами - може бути представлений на шляхом перетворення повідомлення в компонент кожного спостереження. Кожну дію в цьому випадку можна розділити на дві частини: дія в області визначення, яка впливає на стан середовища, і дія зв'язку, яке впливає на повідомлення, що одержуються іншими агентами.

Модель, еквівалентна DEC-POMDP, так звана проблема рішення мультиагентних команд (MTDP), була введена в 2002 році [48]. Інша споріднена модель, що явно представляє переконання про інших агентів, так звана інтерактивна POMDP (I-POMDP), була введена в 2005 році [49].

Таким чином, ми концентруємося на кооперативних системах, в яких кожен агент хоче максимізувати спільну глобальну функцію винагороди. На відміну від цього, некооперативні мультиагентні системи, такі як стохастичні ігри, що частково спостерігаються (POSG), дозволяють кожному агенту мати свою власну приватну функцію винагороди. Методи рішення для некооперативної мультиагентних систем дуже різні і часто ґрунтуються на методиках алгоритмічної теорії ігор [51] і проектуванні обчислювальних систем [52].

Одним з важливих аспектів, що відрізняють моделі одну від іншої, є моделювання спілкування агентів. Деякі концепції прямо моделюють дії

агентів в області комунікації, а інші поділяють їх на загальні набори дій. Кожен підхід має свої переваги і недоліки в залежності від напрямку аналізу. Другий аспект, який відрізняє ці моделі, полягає в тому, чи використовують вони непряме або явне представлення переконань агентів. Огляд моделей наведено на рис 3.5.

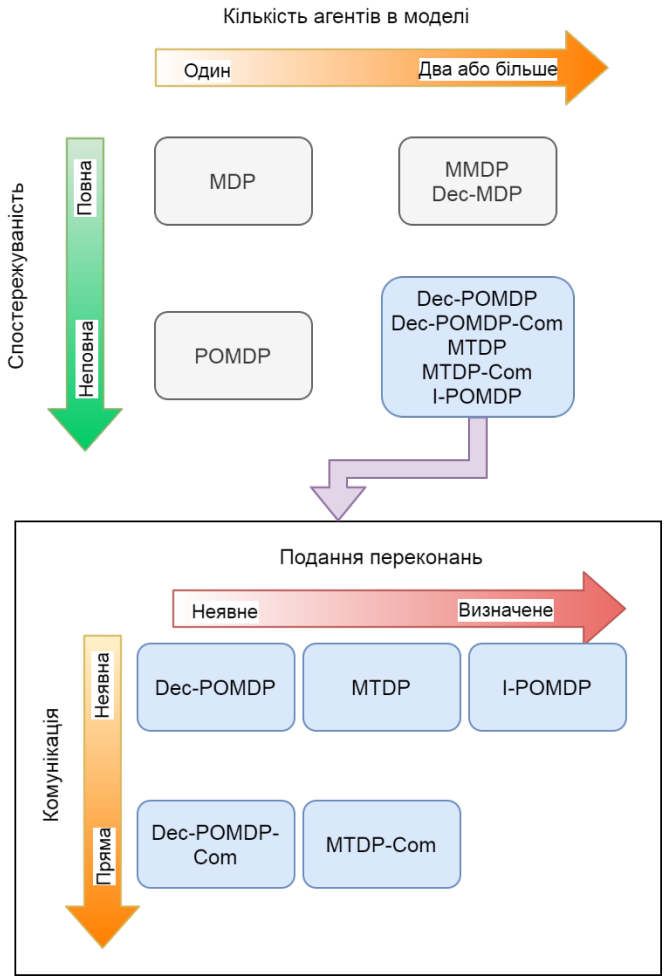


Рис. 3.5. Порівняння різних моделей багато-агентних систем

Модель I-POMDP більш експресивна, ніж модель DEC-POMDP, тому що I-POMDP представляє індивідуальну точку зору агента як на навколишнє середовище, так і на інших агентів, таким чином, також дозволяючи відмовитися від співпраці. Для кооперативних відносин в рамках DEC-POMDP повинні бути вирішене завдання, пов'язане з декількома I-POMDP, де у всіх агентів одна і та ж функція винагороди. Крім того, моделі інших агентів повинні включати інформацію про те, що всі агенти співпрацюють з метою

обчислити адекватне переконання. На практиці точні переконання не можуть бути обчислені, коли вони нескінченно вкладені. Таким чином, модель I-POMDP має такі недоліки [53]:

1. На відміну від моделі DEC-POMDP, модель I-POMDP не має відповідний оптимальний алгоритм. Нескінченна вкладеність переконань призводить до необчислювальних функцій агентів. Завдяки самостійним зіставленням, скінченно вкладені I-POMDP можуть дати тільки обмежені оптимальні рішення.

2. Точне рішення завдання скінченно вкладених I-POMDP також є складним завданням, так як кількість можливих обчислюваних моделей нескінченна. Таким чином, на кожному рівні скінченно вкладеної POMDP агент *i* мав би розглядати нескінченно велику кількість моделей агента *j*, обчислювати їх значення і множити на відповідні ймовірності. Аналіз складності ґрунтується на додатковому припущенні про те, що кількість моделей, що розглядаються на кожному рівні, обмежене кінцевим числом, що є ще одним обмежуючим фактором. Обмежень на похибку результуючої апроксимації не встановлено. Хоча в деяких випадках можна компактно представити переконання відносно нескінченної множини моделей, загальної методики для цього не існує.

3. Навіть з обома наближеннями, тобто з кінцевим вкладенням і обмеженою кількістю моделей, Схоже, що I-POMDP мають подвійну експонентну найгіршу часову складність, і таким чином її ймовірно, так само складно буде вирішити, як і DEC-POMDP.

Незважаючи на ці недоліки, модель I-POMDP являє собою важливу альтернативу DEC-POMDP. Незважаючи на відсутність оптимального алгоритму, це не є великим недоліком, так як оптимальність може бути встановлена тільки для дуже маленьких екземплярів інших описаних нами моделей.

Завдання планування і координації в мультиагентних системах з великою кількістю агентів, що виконують загальне завдання, що працюють в

невизначених і частково спостережуваних середовищах, ускладнює складність обчислювальних задач. Дослідження спрямовані на використання структури в задачах, які дозволяють декомпозицію загального завдання на підзавдання [7].

Мультиагентне планування вивчалося з моменту заснування області розподіленого ШІ. До числа найбільш ранніх робіт в цій області відноситься робота Джорджаффа [8, 9], який розробив деякі з найбільш ранніх методик вирішення конфліктів мультиагентних взаємодій, і Коркілла [10], який розробив розподілену версію планувальника NOAH, створеного Sacerdoti [11]. Коркілл і його колеги, особливо Лессер, були піонерами у використанні організаційних технік для управління множинами агентів [12]. Декер і Лессер узагальнили методи координації агентів у своїй роботі над GPGP, а також для представлення складних завдань мультиагентних мереж у своїй роботі над TAEMS [13].

Планування груп агентів досліджувалося не тільки Тамбе [14, 15], а й Гроше і Краусом [16], грунтуючись на концепціях Коена і Левеска [17]. Планування багатьох агентів, пов'язане з тимчасовими обмеженнями, також має багато посилань в літературі (наприклад, [18, 19]). До інших робіт з координації завдань, які в значній мірі формуються агентами окремо, відноситься робота Тоніно і ін. [20].

До інших методів, що сформулювали завдання багатоагентного планування в теоретичному плані, відносяться ті, які вирішують проблеми, де агенти взаємодіють за допомогою призначення і перепризначення ресурсів [21, 22], і де агенти взаємодіють, змінюючи загальний стан структурованими способами, які дозволяють один одному [23, 24].

Динамічне програмування на основі пам'яті (MBDP) [25] в останні роки було значно покращено за рахунок впровадження різних методів для скорочення числа спостережень, що розглядаються в рамках цієї політики, і використання ефективних методів обрізки. Останнім часом були впроваджені точкові методи, що дозволяють впоратися з NP-складністю операції

резервного копіювання [26]. Цей алгоритм використовує останні досягнення в літературі по зваженим CSP, щоб запропонувати схему полі-тимчасової апроксимації, яка може обробляти набагато більшу кількість точок переконання (MaxTrees). Інший метод - динамічне програмування на основі проб (TBDP) [27] - поєднує в собі основні переваги DP-JESP і MBDP, що дозволяє уникнути дорогих операцій резервного копіювання, дозволяючи вирішувати проблеми з набагато більшими просторами станів.

Локальність агентського взаємодії - той факт, що кожен агент взаємодіє з невеликою кількістю сусідніх агентів - виявився вирішальним при розробці алгоритмів DEC-POMDP, здатних працювати з десятками агентів. Для охоплення структурованих взаємодій і розробки ранніх алгоритмів, здатних використовувати такі структури, були впроваджені спеціалізовані моделі, такі як мережеві розподілені POMDP (ND-POMDP) [28]. Останнім часом було показано, що алгоритм динамічного програмування на основі обмежень (CBDP) забезпечує велике прискорення завдяки своїй лінійної складності в кількості агентів [29]. Розроблено також алгоритми вирішення слабозв’язанних завдань з нескінченними інтервалами. Одним з перспективних та практичних напрямків є трансформація завдання оптимізації політики в задачу максимізації ймовірності та їх поєднання в динамічних байесовських мережах [30]. На основі цього переформулювання завдання був використаний алгоритм “очікування-максимізації” (EM) для обчислення політики за допомогою простої парадигми передачі повідомлень, керованої графом взаємодії агентів.

Далі ми розглянемо найбільш фундаментальні варіанти вибору алгоритмів глибокого навчання з підкріпленням, компроміси в цьому виборі та кілька відомих сучасних алгоритмів в контексті щодо цього вибору. Невичерпний перелік алгоритмів НП показано на рис. 3.6.

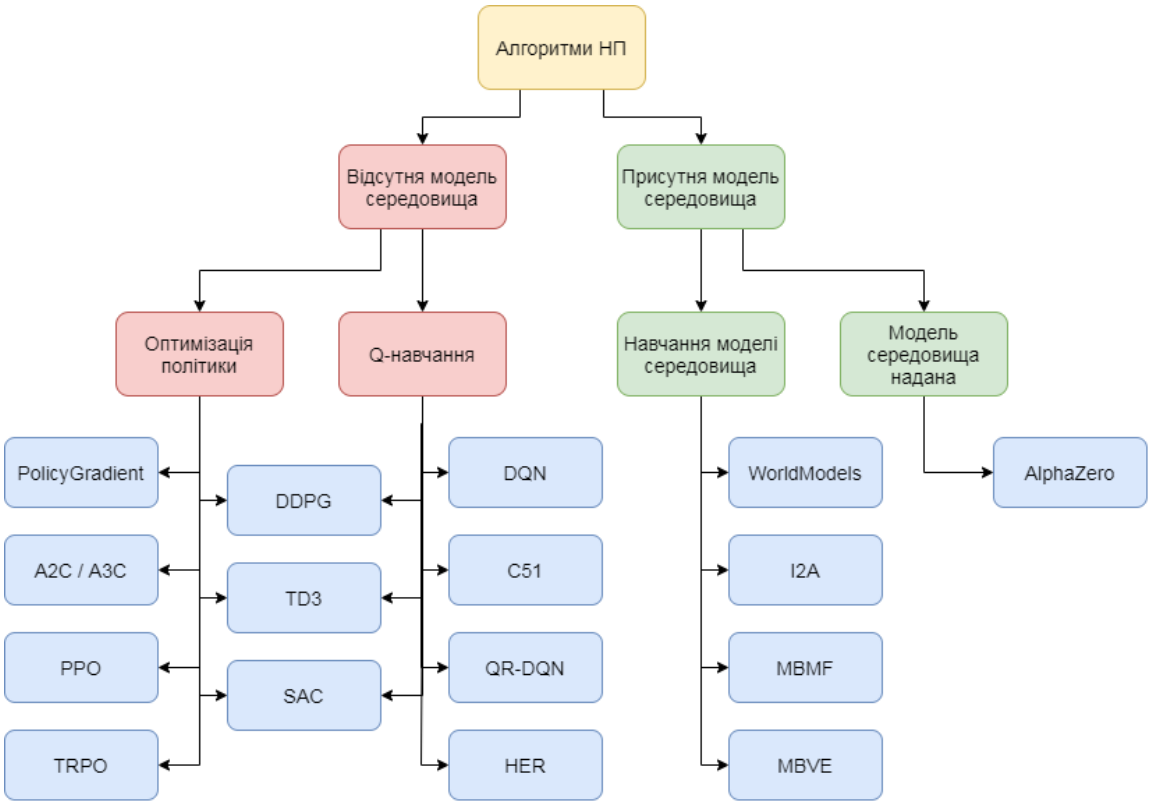


Рис. 3.6. Найбільш відомі алгоритми глибокого НП

Одним з найбільш важливих моментів розгалуження в алгоритмах НП (рис. 3.6) є питання про те, чи має агент доступ (або вивчає) модель середовища. Під моделлю середовища ми розуміємо функцію, що визначає переходи станів та винагороди.

Основною перевагою наявності моделі середовища є те, що вона дозволяє агенту планувати, прораховуючи заздалегідь, що станеться з цілою низкою можливих варіантів дій, і чітко вибирати між своїми варіантами дій. Потім агенти можуть перетворити результати завчасного планування в вивчену політику. Особливо відомим прикладом такого підходу є AlphaZero [31]. Коли це працює, це може привести до значного підвищення ефективності вибірки дій в порівнянні з методами, які не мають моделі.

Основний недолік полягає в тому, що агент, як правило, не має доступу до повної моделі навколишнього середовища. Якщо агент хоче використовувати модель в цьому випадку, він повинен вивчити модель виключно з досвіду, що створює кілька проблем. Найбільша проблема полягає

в тому, що агент може використовувати модель упереджено, в результаті чого агент, який добре працює по відношенню до вивченої моделі, може поводитись неоптимально або неприпустимо в реальному середовищі. Вивчення моделі в основному важке завдання, тому навіть інтенсивні зусилля з великими витратами часу і обчислень, можуть не виправдатись.

Алгоритми, що використовують модель, називаються методами, що мають модель середовища, а інші, тими, де відсутня модель середовища. У той час як методи, де відсутня модель середовища, не мають потенційного підвищення ефективності вибірки дій в результаті використання моделі, вони, як правило, легше впроваджуються і налаштовуються. На момент написання методи, де відсутня модель середовища більш популярні і були більш широко розроблені і випробувані, ніж методи, що мають модель середовища.

Інша важлива точка розгалуження в алгоритмах НП - це питання про те, чому навчається агент. Перелік включає наступне:

- політики - стохастичні або детерміновані,
- функції ФФЗ за дією (Q-функції),
- ФФЗ,
- моделі середовища.

Існує два основних підходи до побудови і навчання агентів в алгоритмах НП, де відсутня модель середовища:

1. Оптимізація політики. Методи в цьому сімействі будують політику, явно виражену як $\pi_\theta(a|s)$. Вони оптимізують параметри θ безпосередньо шляхом градієнтного підйому на цільовій функції $J(\pi_\theta)$, або опосередковано, шляхом максимізації локальних апроксимацій $J(\pi_\theta)$. Ця оптимізація майже завжди виконується за політикою, що означає, що кожне оновлення стану використовує тільки дані, зібрані в процесі роботи відповідно до останньої версії політики. Оптимізація політики також зазвичай включає в себе вивчення апроксиматора $V_\phi(s)$ для ФФЗ за політикою $V^\pi(s)$, який використовується для того, щоб з'ясувати, як оновлювати політику.

Відомими методами оптимізації політики є:

- A2C / A3C [32,33], який виконує градієнтний підйом, щоб безпосередню максимізувати результат,
- та PPO [34], який опосередковано максимізує результат, максимізуючи сурогатну цільову функцію, яка дає консервативну оцінку того, на скільки $J(\pi_\theta)$ зміниться в результаті оновлення стану.

2. Q-навчання. Методи в цьому сімействі вивчають апроксиматор $Q_\theta(s,a)$ для оптимальної ФФЗ за дією, $Q^*(s,a)$. Зазвичай вони використовують цільову функцію, засновану на рівнянні Беллмана. Ця оптимізація майже завжди виконується поза політикою, що означає, що кожне оновлення стану може використовувати дані, зібрані в будь-який момент під час навчання, незалежно від того, як агент обирав вивчення середовища, в той час коли дані були отримані. Відповідна політика отримується через зв'язок між Q^* та π^*: дії, що виконуються агентом з Q-навчанням записані нижче:

$$a(s) = \arg\max_a Q_\theta(s,a).$$

Відомі методи Q-навчання включають:

- DQN [35], класичний метод, що започаткував напрямок вивчення глибинного НП,
- та C51 [36], варіант, що вивчає розподіл здобутку, очікування якого Q^*.

Основною перевагою методів оптимізації політики є те, що вони безпосередньо оптимізують те, що потрібно. Це, як правило, робить їх стабільними і надійними. Методи Q-навчання, навпаки, тільки побічно оптимізують агентів, шляхом тренування Q_θ, щоб задовольнити рівняння самопогодження. Існує безліч режимів невдач для такого навчання, тому воно вважається менш стабільним [37]. Але переваги методів Q-навчання полягають в тому, що вони значно ефективніші при роботі з вибірками, так

як можуть повторно використовувати дані більш ефективно, ніж методи оптимізації політики.

3. Гібридні методи оптимізації політики та Q-навчання

Оптимізація політики та Q-навчання не є несумісними, а в деяких обставинах, навіть, еквівалентними, та існує цілий ряд алгоритмів, що знаходяться між цими крайнощами. Такі гібридні алгоритми, здатні ретельно шукати компроміс між сильними і слабкими сторонами оптимізації політики та Q-навчання. Найбільш поширені методи включають в себе:

- DDPG [38], алгоритм, який одночасно навчає детерміністичну політику та Q-функцію, використовуючи кожну для поліпшення іншої,
- та SAC [39], варіант, який використовує стохастичні політики, регуляризацію ентропії і кілька інших прийомів для стабілізації навчання і отримання більш високих результатів, ніж DDPG за стандартними контрольними показниками.

На відміну від алгоритмів, що не мають моделі середовища, алгоритми НП, що мають модель середовища важко структурувати тому, що існує безліч різноманітних способів використання моделей. У кожному разі модель може бути або надана, або вивчена з середовища.

Найголовніший підхід ніколи не представляє політику в явному вигляді, а замість цього використовує чисті методи планування, такі як передбачуваний контроль моделі (ПКМ) для вибору дій. В ПКМ [40] кожен раз, коли агент спостерігає за середовищем, він обчислює план, що оптимальний по відношенню до моделі, в якому план описує всі дії, які необхідно зробити протягом деякого фіксованого проміжку часу після теперішнього часу. Майбутні винагороди за горизонтом дій можуть бути враховані алгоритмом планування за допомогою функції ФФЗ. Потім агент виконує першу дію плану, а від інших негайно відмовляється. Кожний раз при підготовці до взаємодії з середовищем він розраховує новий план, щоб уникнути використання дії з плану з більш коротким горизонтом планування.

- В роботі MBMF [41] досліджується ПКМ з вивченими моделями середовища на деяких стандартних еталонних завданнях для глибинного НП.

Пряме доповнення до чистого планування передбачає використання і вивчення чіткого представлення політики $\pi_\theta(a|s)$. Агент використовує алгоритм планування, наприклад, Monte Carlo Tree Search [42], в моделі, генеруючи кандидатські дії для плану за вибіркою з поточної політики. Алгоритм планування виробляє дію, яка краща, ніж та, що могла б бути надана політикою, тому він є "експертом" по відношенню до політики. Політика потім оновлюється, щоб отримати дію, що більш схожа на результат роботи алгоритму планування.

- Алгоритм ExIt [43] використовує цей підхід для навчання глибинних нейронних мереж для гри в Hex.
- AlphaZero [31] ще один приклад такого підходу.

Можна використовувати генерування додаткових даних для алгоритмів НП, без моделі середовища. Можна використовувати такі алгоритми для навчання політики або Q-функції, але збільшувати реальний досвід при оновленні стану агента вигаданим, або використовувати тільки фіктивний досвід при оновленні стану агента.

- MBVE [44] є прикладом доповнення реального досвіду вигаданими.
- WorldModels [45] є прикладом використання тільки вигаданого досвіду для навчання агента, що автори називають "тренуванням уві сні".

Інший підхід вбудовує процедуру планування безпосередньо в політику в якості підпрограмної основи, таким чином що повні плани стають побічною інформацією для політики - в той же час навчаючи результуючу політику будь-яким стандартним алгоритмом, що не використовує модель середовища. Ключова концепція полягає в тому, що в такому вигляді політика може навчитися вибирати, як і коли використовувати плани. Це робить

упередженість агента відносно моделі меншою проблемою, тому що якщо модель погано підходить для планування в деяких станах, політика може просто навчитися ігнорувати це.

- I2A [46] є прикладом того, як агенти наділені таким стилем навчання.

3.4. Модель комплексу програмного середовища проектування. Інтеграція інтелектуальних агентів та САПР

Для вирішення завдання оптимального проектування випробувального стенду потрібно забезпечити інтеграцію комплексу програмних бібліотек та САПР в автоматизований процес розробки ВС за допомогою штучного інтелекту [67]. Структура багато-агентної моделі проектування ВС з використанням інтегрованих систем САПР показана на (Рис 3.7.).

Етапи оптимізації та проектування розподіляються на декілька кроків, відповідно до моделі проектування визначеної у попередньому розділі. Визначається підтип технічного рішення для застосування в підсистемі ВС. Результат розв'язання задачі - обраний клас технічного рішення. Після цього вирішується завдання оптимізації параметрів підсистем випробувального стенду. Результат - визначені параметри підсистем.

В якості прикладу розроблено агента для вирішення завдання оптимального вибору комплексу технічних засобів для проектування карданного підвісу випробувального стенду. Завдання вибору комплексу технічних засобів сформулювано як завдання багатокритеріального прийняття рішень.

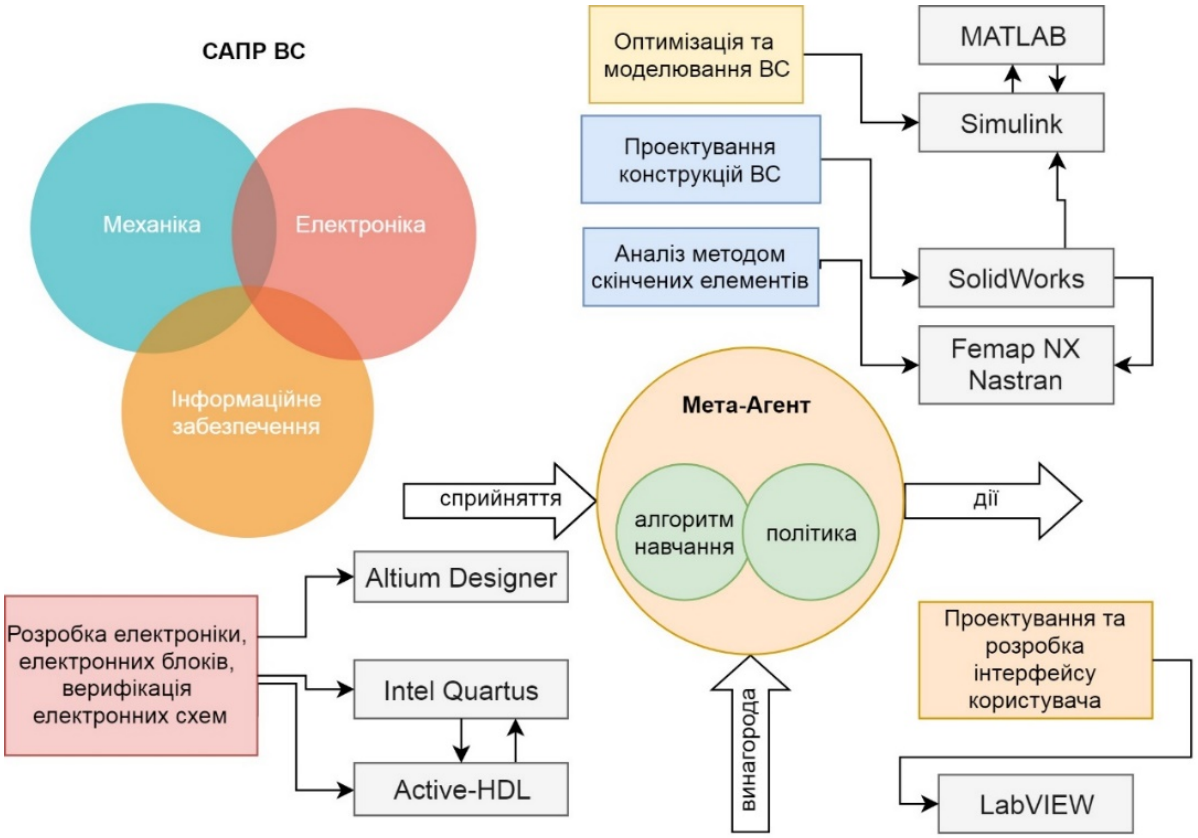


Рис. 3.7. Інтеграція існуючих САПР в єдиний автономний комплекс

Основні критерії для вибору матеріалів та форми карданного підвісу є термовитривалість, вимоги до обертового моменту, низький обертальний резонанс, поперечна жорсткість, просторова динамічна стабільність конструкції, складність побудови інтерфейсу для під'єднання навігаційного обладнання, складність забезпечення вимог до відхилень параметрів конструкції та стандартів якості необхідних на виробництві, вартість виробництва, тривалість циклу виробництва .

Таблиця 3.1. Основні критерії для вибору матеріалів та форми карданного підвісу

Матеріали	термо витривалість	вимоги до обертового моменту	вартість виробництва	Тривалість циклу виробництва	низький обертальний резонанс
	max	min	min	min	max

алюміній	1	5	1	1	1
композитні матеріали	1	1	10	5	2.5

Форма	поперечна жорсткість	просторова динамічна стабільність конструкції	складність побудови інтерфейсу для під'єднання навігаційного обладнання	складність забезпечення вимог до відхилень параметрів конструкції та стандартів якості необхідних на виробництві	вартість виробництва
	max	max	min	min	min
Прямокутна	менша	менша	менша	менша	менша
Сферична	більша	більша	більша	більша	більша

Параметри наведені у відносних величинах, за необхідності та наявності можливе використання точних параметрів від виробника або експериментальних даних.

В попередньому випадку всі критерії вважались рівнозначними. Для завдання вибору комплексу технічних засобів підсистеми опорних підшипників додано параметр ваги критерію.

Таблиця 3.2. Основні критерії для вибору підсистеми опорних підшипників

	Биття	Тертя	Ортогональність осей	Жорсткість	Надійність	складність системи живлення, забезпечення та підтримки	вартість виробництва
Вага, %	15	10	20	10	10	10	25
	min	min	min	max	max	min	min
Механічні кулькові підшипники	4	4	4	4	4	1	1
Гідростатичні гідравлічні підшипники	2	3	3	3	1	4	5
Гідростатичні пневматичні підшипники	2	2	3	2	3	2	3
магнітні підшипники	1	1	1	1	2	3	10

Для системи кодування положення вибір технічних засобів складається з інерційних датчиків, синусно-косинусного обертального трансформатора (СКОТ), оптичного енкодера та індуктосина. Застосування інерційних датчиків як системи кодування положення запропоновано дослідниками як протилежний підхід до класичного. У світовій практиці розглядаються в більшості випадків оптичні енкодери, індуктосини та СКОТ.

Таблиця 3.3. Основні критерії для вибору підсистеми кодування положення

	стійкість до шумів	точність	короткострокова стабільність показників	вартість виробництва	складність інтеграції
	max	min	min	min	min
інерційні датчики	1	4	4	4	4
СКОТ	3	3	3	2	3
оптичні енкодери	2	2	1	1	1
індуктосин	4	1	2	3	2

Таблицю 3.1. завдання вибору матеріалу для прийняття рішення далі перетворюємо у завдання навчання агента. Для цього перетворюємо матеріал кардонового підвісу у дії агента, а критерії кодуємо станами середовища навчання.

На рис. 3.8 показаний граф середовища для прийняття рішень агента вибору матеріалу кардонового підвісу.

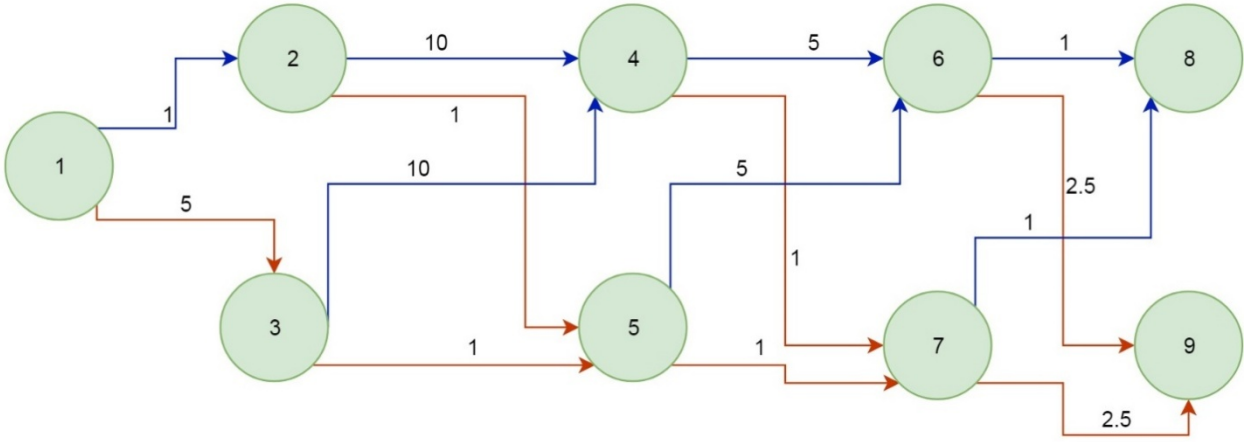


Рис. 3.8. Середовище навчання з підкріпленням для інтелектуального агента

На рис. 3.8. кожне коло являє собою стан. У кожному стані приймається рішення про вибір матеріалу карданного підвісу: синя стрілка - алюміній,

оранжева стрілка - композитні матеріали. Агент починає із стану 1. В процесі переходу між станами агент отримує винагороду, яка дорівнює значенню на кожному переході в графі. Значення винагороди це критерії багатокритеріального завдання мінімізації та максимізації із таблиці наведеної вище. Метою навчання є отримання максимальної сумарної винагороди.

Для оптимізації всього випробувального стенду загалом, модель нагороди багатоагентного середовища повинна використовувати багатотільне моделювання випробувального стенду у спеціалізованих САПР. (Рис. 3.9.)

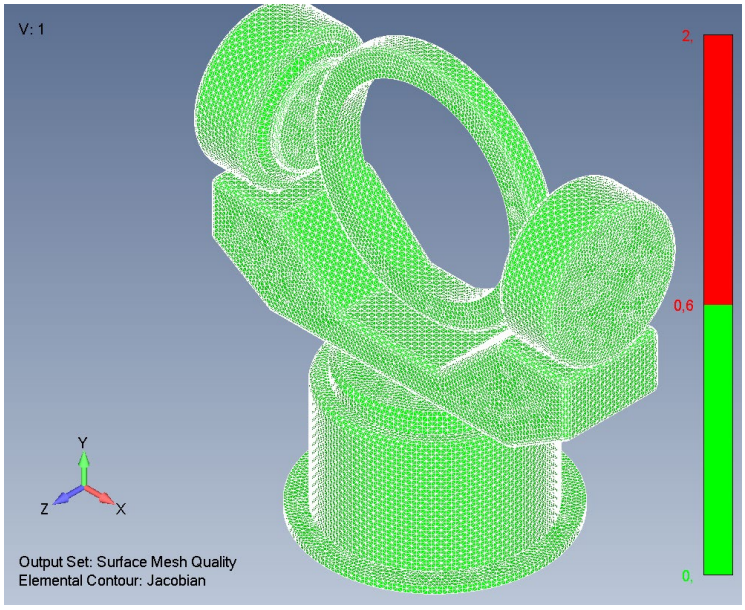


Рис. 3.9. Підготовка моделі випробувального стенду для багатотільного моделювання у САПР

Навчання у багато-агентному середовищі, дозволяє включити обмеження завдання на проектування ВС у вигляді функції нагороди. СПРА проектування стенду складається з чотирьох ключових елементів:

1. Стани кодують інформацію про часткове проектне рішення, включаючи параметри електричних підсистем (моменти, прискорення, механічні коефіцієнти, частота вибірки інформації про кутове положення), характеристики механічних частин (маса, ортогональність, матеріал), характеристики системи управління.

2. Дії - це всі можливі параметри проектних рішень підсистем, які можуть бути визначені в поточний момент без порушення будь-яких жорстких обмежень на завдання проектування ВС.

3. Переходи між станами визначають розподіл ймовірностей за такими станами, з огляду на стан і дію.

4. Винагорода дорівнює 0 для всіх дій, крім останнього дії, де винагорода є зваженою сумою моделі статичної похибки, динамічної похибки та моделі механічної конфігурації ВС. Розробка правильної функції винагороди є одним з найбільш важливих рішень. Функція винагороди повинна швидко обчислюватись через те що навчання агента часто потребує сотні тисяч ітерацій для досягнення результату. Функція винагороди повинна максимально корелювати з істинною метою завдання проектування ВС. Врахування обмежень – це отримання агентом великої негативної винагородою щоразу за проектні параметри що виходять за межі проектного завдання.

В моделі винагороди для вирішення завдання координації пропонується використати:

- Модель статичної похибки ВС НО [6];
- Модель динамічної похибки ВС НО [54];
- Багатотільну модель механічної конфігурації ВС НО [55].

Агенти мають свій простір спостереження обмежений параметрами підсистеми за яку вони відповідають. Простір дій дозволяє обрати або змінити компонент, сигналізувати, що немає варіантів, недостатньо параметрів, або запросити погодження у спорідненого агента [68].

Винагорода кожного агента має включати найважливіші з точки зону завдання на проектування обмеження, такі як точність, вартість, швидкість, що притаманні конкретній підсистемі. Наприклад, для вибору системи кодування

положення вибір за точністю може не дати бажаного результату, тому як потрібно враховувати швидкість та погоджувати її з іншим агентом. Інколи важливо обрати найдешевші за вартістю компоненти, а потім замінити деякі частини на більш дорогі. Для цього система має сигналізувати, що виникла проблема у виборі, і необхідно або розширити бюджет пошуку, або зменшити точність чи інший параметр підсистеми. Така інтерактивністю дозволяє більш гнучку співпрацю с людиною оператором, який має приймати кінцеві рішення.

Навчання агента, що моделюється нейронною мережею, відбувається через повторювані епізоди послідовності станів, дій і винагород. Агент вчиться робити дії, максимізує сукупну винагороду. Для навчання доцільно застосовувати алгоритм (PPO) для поновлення параметрів мережі політики, з огляду на кумулятивну винагороду за кожне обране проектне рішення ВС.

В початковому стані s_0 ми маємо порожній масив даних проектних параметрів ВС. На кожному кроці оптимізується один проектний параметр, і кінцевий стан, s_{Final}, відповідає повністю оптимізовану багатокритеріальну задачу ВС. Таким чином, T дорівнює загальній кількості проектних параметрів в завданні проектування ВС. На кожному кроці часі t агент починає виконувати дію в стані s_t, виконує дію (a_t), переходить в новий стан (s_{t+1}) і отримує винагороду (r_t) від середовища проектування. s_{t+1} - це наступний стан, який включає оновлене уявлення агента, що містить інформацію про новий оптимізований проектний параметр ВС.

Модель дій агентів представляє собою простір дій $m \times n$, що є розподілом підсистем m на кількість їх проектних рішень n.

Незважаючи на відмінності між алгоритмами оптимізації в популяційних метаевристичних методах, процес оптимізації ділиться на дві основні фази: розвідка та диверсифікація, експлуатація та інтенсифікація. Це потрібно для широкого охоплення простору пошуку шляхом використання різних пошукових рішень використовуваного алгоритму для уникнення проблем пошуку.

Для покращення алгоритму оптимізації пропонується поєднання мета-евристичного алгоритму полювання орла (АО) та навчання з підкріпленням.

Чотири фази пошуку рішення алгоритму АО - це рівняння (3.2), формула (3.4), формула (3.8) та формула (3.9), перетворюються на дії агента. Крім того агент може налаштовувати додаткові параметри чотирьох дій для досягнення найкращих результатів повністю моделюючи таким чином поведінку орла. Як результат, призведе до вибору найкращих оптимальних рішень параметрів ВС під час дослідження величезного простору можливих параметрів.

Пропонований АО-Агентний алгоритм
1: Вхідні дані рішення-кандидати (X) та кількість епізодів навчання (e_{tmax}).
2: Ініціювати стани, дії та параметри агента. $$a_t = (a_{0,t}, \dots, a_{X,t})$$ $$s_t = (q_{0,t}, q_{1,t}, \dots, q_{X,t}, d_{t-1}, \dots, d_{t-\tau})$$ де $$X_t = (X_{1,t}, \dots, X_{X,t})$$
5: Виконувати цикл $e_t <= e_{tmax}$
6: Розрахувати винагороду в поточному стані. $$R(\tau) = \sum_{t=0}^{T} r_t$$
7: Знайти (X_{best}).
8: Оновити політику навчання агента використовуючи оператори АО в формулах (3.3)–(3.11).
9: t=t+1.
10: завершити виконання
11: Зберегти параметр Xbest.

Як завдання для майбутніх досліджень, дії агента можуть бути будь-яким природничим алгоритмом - завдання полягає в систематизації всіх метаевристичних способів дослідження та звуження простору пошуку.

Через повторювані епізоди (послідовності станів, дій і винагород) нейронна мережа поведінки агента вчиться робити дії, які максимізують сукупну винагороду.

Пошук архітектури нейронної мережі яка може з найбільшою точністю передбачувати нагороду для оптимізації випробувального стенду є окремим складним завданням. Потім ця архітектура кодує мережу поведінки агента та забезпечує оптимальну координацію дій.

Висновки до розділу 3

У третьому розділі розроблена підсистема автоматизованого проектування інтелектуальних агентів проектування елементів ВС на базі штучного інтелекту.

Використовуючи теорії мета-евристичних алгоритмів та навчання з підкріпленням, на базі існуючою методології в області багато-агентного моделювання та алгоритмів навчання з підкріпленням була створена модель комплексу програмного середовища проектування ВС.

Метод побудови САПР ВС забезпечує інтеграцію існуючих САПР в єдиний автономний комплекс, в якому людина-оператор визначає критерії та обмеження, в межах яких багато-агентній системі з підтримкою ШІ дозволяється управляти автоматизованими операціями. Агенти-проектувальники з елементами ШІ САПР ВС використовують алгоритми навчання з підкріпленням для вирішення завдання багатокритеріального прийняття рішень, в результаті забезпечуються покращення простору пошуку проектних рішень з точки зору використання попередніх знань.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ ДО РОЗДІЛУ 3

1. S. Talatahari, M. Azizi, “Chaos Game Optimization: a novel metaheuristic algorithm,” *Artif. Intell. Rev.,* vol. 54, pp. 917–1004, 2021, doi: 10.1007/s10462-020-09867-w.

2. Z. Jinhao, X. Mi , G. Liang , P. Quanke, “Queuing search algorithm: A novel metaheuristic algorithm for solving engineering optimization problems,” *Applied Mathematical Modelling*, 2018, doi: 10.1016/j.apm.2018.06.036.

3. L. Abualigah, D. Yousri, M. A. Elaziz , A. A. Ewees, M. A. A. Al-qaness,
A. H. Gandomi, “Aquila Optimizer: A novel meta-heuristic optimization Algorithm,” *Computers & Industrial Engineering*, 2021, doi: 10.1016/j.cie.2021.107250.

4. “TensorFlow”. Open source platform for machine learning. [Електронний ресурс]. Режим доступу: https://www.tensorflow.org (дата звернення 19.03.2021).

5. “PyTorch”. An open source machine learning framework. [Електронний ресурс]. Режим доступу: https://pytorch.org (дата звернення 19.03.2021).

6. N. Andrianos, C. Diez, L. DeMore, “Flight table orientation error transparency for hardware-in-the-loop facilities,” *in Proc. of SPIE - Aerospace/Defense Sensing, Simulation, and Controls*, vol. 4366, 2001, doi: 10.1117/12.438076.

7. G. Weiss et al., *Multiagent Systems*. 2nd ed. MIT Press, 2013.

8. M. Georgeff, “Communication and interaction in multi-agent planning,” *in Proc. of the 3rd National Conf. on Artificial Intelligence (AAAI-83)*, pp. 125–129, 1983.

9. M. P. Georgeff, “A theory of action for multiagent planning,” *in Proc. AAAI-84*, pp. 121–125, 1984.

10. D. Corkill, “Hierarchical planning in a distributed problem-solving environment,” *in Proc. of the 7th Int. Joint Conf. on Artif. Intell*., pp. 168–175, 1979.

11. E. Sacerdoti, *A Structure for Plans and Behavior*. Elsevier North-Holland Inc.,1977.

12. D. Corkill and V. Lesser, "The use of meta-level control for coordination in a distributed problem solving network*," in Proc. of the 8th Int. Joint Conf. on Artificial Intelligence*, pp. 748–756, 1983.

13. V. Lesser, K. Decker, T. Wagner, N. Carver, A. Garvey, B. Horling, D. Neiman, R. Podorozhny, M. Nagendra Prasad, A. Raja, R. Vincent, P. Xuan, and X.Q. Zhang, "Evolution of the GPGP/TAEMS domain-independent coordination framework" *Autonomous Agents and Multi-Agent Systems*, vol. 9, no. 1, pp. 87–143, 2004.

14. D. V. Pynadath and M. Tambe, "An automated teamwork infrastructure for heterogeneous software agents and humans," *Autonomous Agents and Multi-Agent Systems*, vol. 7, pp. 71–100, 2003.

15. M. Tambe "Agent architectures for flexible, practical teamwork," *in Proc. of AAAI/IAAI'97*, pp. 22–28, 1997.

16. B. J. Grosz and S. Kraus, "Collaborative plans for complex group action," *Artificial Intelligence*, vol. 86, no. 2, pp. 269–357, 1996.

17. P. R. Cohen and H. J. Levesque, "Intention is choice with commitment," *Artificial Intelligence*, vol. 42, no. 2-3, pp. 213–261, 1990.

18. A. Beynier and A.-I. Mouaddib, "A polynomial algorithm for decentralized Markov decision processes with temporal constraints," *in Proc. of the 4th Int. Joint Conf. on Autonomous Agents and Multiagent Systems (AAMAS-05)*, pp. 963–969, 2005.

19. I. Tsamardinos, M. E. Pollack, and J. F. Horty, "Merging plans with quantitative temporal constraints, temporally extended actions, and conditional branches," *in Proc. of the Conf. on AI Planning Systems*, pp. 264–272, 2000.

20. H. Tonino, A. Bos, M. de Weerdt, and C. Witteveen, "Plan coordination by revision in collective agent based systems," *Artificial Intelligence*, vol. 142, no. 2, pp. 121–145, 2002.

21. D. A. Dolgov and E. H. Durfee, “Resource allocation among agents with MDP-induced preferences,” *Journal of Artificial Intelligence Research*, vol. 27, pp. 505–549, 2006.

22. J. Wu and E. H. Durfee, “Resource-driven mission-phasing techniques for constrained agents in stochastic environments,” *Journal of Artificial Intelligence Research*, vol. 38, pp. 415–473, 2010.

23. R. Becker, V. Lesser and S. Zilberstein, “Decentralized Markov decision processes with event-driven interactions,” *in Proc. of the Third Int. Joint Conf. on Autonomous Agents and Multi-Agent Systems*, vol. 1, pp. 302–309, 2004.

24. S. J. Witwicki and E. H. Durfee, “Influence-based policy abstraction for weakly-coupled Dec-POMDPs,” *in Proc. of the 20th Int.Conf. on Automated Planning and Scheduling (ICAPS 2010)*, pp. 185–192, 2010.

25. S. Seuken and S. Zilberstein, “Memory-bounded dynamic programming for DEC-POMDPs,” *in Proc. of the Twentieth Int. Joint Conf. on Artificial Intelligence (IJCAI-07)*, pp. 2009–2015, 2007.

26. A. Kumar and S. Zilberstein, “Point-based backup for decentralized POMDPs: Complexity and new algorithms,” *in Proc. of the Ninth Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS-10)*, pp. 1315–1322, 2010.

27. F. Wu, S.Zilberstein and X. Chen, “Trial-based dynamic programming for multi-agent planning,” *in Proc. of the Twenty-Fourth Conf. on Artif. Intell. (AAAI-10)*, pp. 908–914, 2010.

28. R. Nair, P. Varakantham, M. Tambe and M. Yokoo, “Networked distributed POMDPs: A synthesis of distributed constraint optimization and POMDPs,” *in Proc. of the 20th National Conf. on Artif. Intell. (AAAI-05)*, pp. 133–139, 2005.

29. A. Kumar and S. Zilberstein, “Constraint-based dynamic programming for decentralized POMDPs with structured interactions,” *in Proc. of the 8h Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS-09)*, pages 561–568, 2009.

30. A. Kumar and S. Zilberstein, “Message-passing algorithms for large structured decentralized POMDPs,” *in Proc. of the Tenth Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS-11)*, pp. 1087–1088, 2011.

31. D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan and D. Hassabis. *Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm*. London: Google Deepmind, 2017.

32. V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. P. Lillicrap, T. Harley, D. Silver and K. Kavukcuoglu, “Asynchronous Methods for Deep Reinforcement Learning,” *in Proc. of the 33rd Int. Conf. on Machine Learning*, vol. 48, 2016, pp. 1928–1937.

33. L. Espeholt, H. Soyer, R. Munos, K. Simonyan, V. Mnih, T. Ward, Y. Doron, V. Firoiu, T. Harley, I. Dunning, S. Legg and K. Kavukcuoglu, “IMPALA: Scalable Distributed Deep-RL with Importance Weighted Actor-Learner Architectures,” *in Proc. of the 35th Int. Conf. on Machine Learning*, 2018.

34. J. Schulman, F. Wolski, P.Dhariwal, A. Radford and O. Klimov, “Proximal Policy Optimization Algorithms,” OpenAI, 2017, [Електронний ресурс]. Режим доступу: https://arxiv.org/pdf/1707.06347.pdf (дата звернення 12.03.2021).

35. V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, M. Riedmiller, “Playing Atari with Deep Reinforcement Learning,” *in NIPS Deep Learning Workshop*, 2013.

36. M. G. Bellemare, W. Dabney and R. Munos, “A Distributional Perspective on Reinforcement Learning,” *in Proc. Of Int. Conf. on Machine Learning*, pp. 449-458, 2017.

37. J. N. Tsitsiklis and B. Van Roy, “An analysis of temporal-difference learning with function approximation,” *in IEEE Trans. on Automatic Control*, vol. 42, no. 5, pp. 674-690, 1997, doi: 10.1109/9.580874.

38. T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, “Continuous control with deep reinforcement learning,” *in Proc. of 4th Int. Conf. on Learning Representations*, 2016.

39. T. Haarnoja, A. Zhou, P. Abbeel and S. Levine, “Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor,” *in Proc. of the 35th Int. Conf. on Machine Learning*, 2018.

40. R. Kamyar; E. Taheri, “Aircraft Optimal Terrain/Threat-Based Trajectory Planning and Control.” *Journal of Guidance, Control and Dynamics*, vol. 37, no. 2, pp. 466–483, 2014.

41. A. Nagabandi, G. Kahn, R. S. Fearing and S. Levine, “Neural Network Dynamics for Model-Based Deep Reinforcement Learning with Model-Free Fine-Tuning,” 2017, [Електронний ресурс]. Режим доступу: https://arxiv.org/pdf/1708.02596.pdf (дата звернення 12.03.2021).

42. D. Silver, A. Huang, C. Maddison, et al., “Mastering the game of Go with deep neural networks and tree search,” *Nature*, vol. 529, pp. 484–489, 2016, doi: 10.1038/nature16961.

43. T. Anthony, Z. Tian and D. Barber, “Thinking Fast and Slow with Deep Learning and Tree Search,” *in Proc. of 31st Conf. on Neural Information Processing Systems (NIPS 2017)*, 2017.

44. V. Feinberg, A. Wan, I. Stoica, M. I. Jordan, J. E. Gonzalez and S. Levine, “Model-Based Value Estimation for Efficient Model-Free Reinforcement Learning,” *in Proc. of the 35th Int. Conf. on Machine Learning*, 2018.

45. D. Ha and J. Schmidhuber, “World Models,” 2018, [Електронний ресурс]. Режим доступу: https://arxiv.org/pdf/1803.10122.pdf (дата звернення 12.03.2021).

46. T. Weber, S. Racanière, D. P. Reichert, L. Buesing, A. Guez, D. J. Rezende, A. P. Badia, O. Vinyals, N. Heess, Y. Li, R. Pascanu, P. Battaglia, D. Hassabis, D. Silver, and D. Wierstra, “Imagination-Augmented Agents for Deep Reinforcement Learning,“ *in Proc. of 31st Conf. on Neural Information Processing Systems (NIPS 2017)*, 2017.

47. M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. New York, NY: John Wiley & Sons, Inc, 1994.

48. D. V. Pynadath and M. Tambe, "The communicative multiagent team decision problem: Analyzing teamwork theories and models," *Journal of Artificial Intelligence Research*, vol. 16, pp. 389–423, 2002.

49. P. J. Gmytrasiewicz and P. Doshi, "A framework for sequential planning in multiagent settings," *Journal of Artificial Intelligence Research*, vol. 24, pp. 49–79, 2005.

50. S. Seuken, S.Zilberstein, "Formal models and algorithms for decentralized decision making under uncertainty," *Auton. Agents and Multi-Agent Syst.,* vol. 17, no. 2, pp. 190–250, 2008.

51. S. Arora, P. Doshi, and B. Banerjee, "A Framework and Method for Online Inverse Reinforcement Learning," *Auton. Agents and Multi-Agent Syst*., vol. 35, no. 4, 2021.

52. S. Kambhampati, M. Cutkosky, M. Tenenbaum and S. H. Lee, "Combining specialized reasoners and general purpose planners: A case study," *in Proc. of the 9th National Conf. on Artif. Intell. (AAAI-91)*, pp. 199–205, 1991.

53. R. Cogill, M. Rotkowitz, B. van Roy, S. Lall, "An approximate dynamic programming approach to decentralized control of stochastic systems," *in Proc. of the Allerton Conf. on Comm., Control, and Computing*, pp. 1040–1049, 2004.

54. L. A. DeMore, R. A. Peterson, L. B. Conley, H. Havliscek and N. P. Andrianos, "Design study for a high-accuracy three-axis test table," *Journal of Guidance, Control, and Dynamics*, vol. 10, no. 1, 1987, doi: 10.2514/3.20187.

55. F. N. Barnes, "Stable Member Equations of Motion for a Three-Axis Gyro Stabilized Platform," *in IEEE Trans. on Aerospace and Electronic Systems*, vol. AES-7, no. 5, pp. 830-842, 1971, doi: 10.1109/TAES.1971.310323.

56. D. S. Bernstein, S. Zilberstein, N. Immerman, "The complexity of decentralized control of Markov decision processes," *in Proc. of the 16th Conf. on Uncertainty in Artificial Intelligence (UAI)*, pp. 32–37, 2000.

57. D. Szer, F. Charpillet, S. Zilberstein, “MAA*: A heuristic search algorithm for solving decentralized POMDPs,” *in Proc. of the Twenty-First Conf. on Uncertainty in Artificial Intelligence (UAI)*, pp. 576–583, 2005.

58. R. Nair, D. Pynadath, M. Yokoo, M. Tambe, S. Marsella, “Taming decentralized POMDPs: Towards efficient policy computation for multiagent settings,” *in Proc. of the 18th Int. Joint Conf. on Artif. Intell. (IJCAI)*, pp. 705–711, 2003.

59. R. Emery-Montemerlo, G. Gordon, J. Schneider, S. Thrun, “Approximate solutions for partially observable stochastic games with common payoffs,” in *Proc. of the Third Int. Joint Conf. on Auton. Agents and Multi-Agent Systems (AAMAS)*, pp. 136–143, 2004.

60. S. Seuken, S. Zilberstein, “Memory-bounded dynamic programming for DEC-POMDPs,” *in Proc. of the Twentieth Int. Joint Conf. on Artif. Intell. (IJCAI)*, pp. 2009–2015, 2007.

61. S. Seuken, S. Zilberstein, “Improved memory-bounded dynamic programming for decentralized POMDPs,” *in Proc. of the 23rd Conf. on Uncertainty in Artif. Intell. (UAI)*, 2007.

62. C. V. Goldman, S. Zilberstein, “Goal-oriented DEC-MDPs with direct communication”, Dept. of Comp.Science, Univ. of Massachusetts, Amherst, MA, USA, Tech. Report 04-44, 2004.

63. D. S. Bernstein, “Complexity analysis and optimal algorithms for decentralized decision making,” Ph.D. dissertation, Dept. of Comp. Science, Univ. of Massachusetts, Amherst, MA, USA, 2005.

64. D. S. Bernstein, E. A. Hansen, S. Zilberstein, “Bounded policy iteration for decentralized POMDPs,” *in Proc. of the 19th Int. Joint Conf. on Artif. Intell. (IJCAI)*, pp. 1287–1292, 2005.

65. D. Szer, F. Charpillet, “An optimal best-first search algorithm for solving infinite horizon DEC-POMDPs,” *in Proc. of the 16th European Conf. on Machine Learning (ECML)*, pp. 389–399, 2005.

66. C. Amato, D. Bernstein, S. Zilberstein, “Optimizing memory-bounded controllers for decentralized POMDP,” *in Proc. of the 23rd Conf. on Uncertainty in Artif. Intell. (UAI)*, 2007.

67. V. M. Sineglazov and S. O. Dolgorukov, “Computer-Aided Design System of Navigation Equipment Test Table,” *in Proc. IEEE 5th Int. Conf. Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, (Kyiv, Ukraine), 2019, pp. 7-11, doi: 10.1109/APUAVD47061.2019.8943871.

68. V. M. Sineglazov and S. O. Dolgorukov, “Intelligent Integrated Cads of Navigation Equipment Testing System,” *in Proc. IEEE 5th Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, (Kyiv, Ukraine), 2018, pp. 7-10, doi: 10.1109/MSNMC.2018.8576314.

РОЗДІЛ 4.
ОЦІНКА САПР НА ПРИКЛАДІ РЕАЛЬНОГО ЗАСТОСУВАННЯ ДЛЯ ПРОЕКТУВАННЯ ВИПРОБУВАЛЬНИХ СТЕНДІВ

4.1. Приклад імплементації інтелектуального агента випробувального стенда

Для навчання з підкріпленням використовується Matlab Reinforcement Learning Toolbox [1]. Це програмне середовище дозволяє використовувати функції та блоки для навчання агентів за допомогою алгоритмів навчання з підкріпленням. Агенти після навчання можуть приймати рішення у складних автономних системах. Reinforcement Learning Toolbox дозволяє взаємодіяти з середовищем, що складається з моделей MatLab чи Simulink. Крім того зручні інструменти, дозволяють оцінювати алгоритми, проводити налаштування параметрів алгоритмів, відслідковувати хід навчання агентів. MatLab є одним з найбільших середовищ для побудови систем САПР, дозволяє інтегрувати максимальну кількість обчислювальних модулів та взаємодіяти з іншими САПР. Для підвищення ефективності роботи системи можлива інтеграція з хмарними обчисленнями та додатковими серверами (Parallel Computing Toolbox та MATLAB Parallel Server). Підтримується імпорт агентів з середовищ глибинного навчання Tensor Flow, Keras та PyTorch (DeepLearningToolbox). Більш того підтримується автоматична генерація програмного коду на мовах програмування C, C++ для розгортання політик агентів на мікроконтролерах, мікропроцесорах та графічних системах обчислення.

Далі розглянемо імплементацію агента, що приймає рішення про вибір матеріалу випробувального стенду у середовищі MatLab. На рисунку 4.1 показана модель середовища навчання агента для моделювання процесу прийняття рішень розглянутого у попередньому розділі.

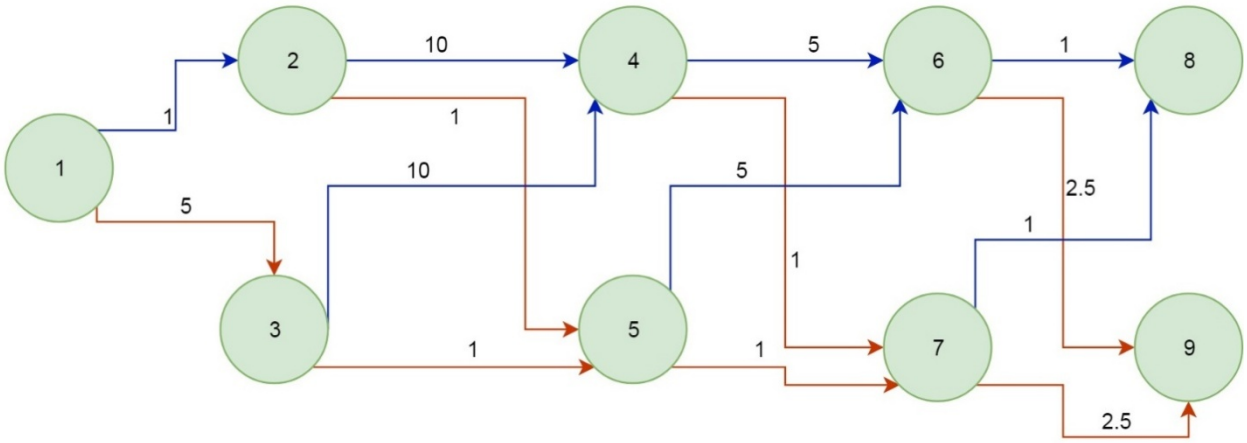


Рис. 4.1. Модель середовища навчання агента для визначення матеріала виготовлення ВС.

Спочатку потрібно створити модель середовища навчання агента з дев'ятьма станами та двома діями - вибір матеріалу алюміній або композитні матеріали.

MDP = createMDP(9,["aluminium";"composite"]);

Для моделювання переходів між станами на рисунку 4.1 потрібно задати значення матриць переходу станів та матриці нагород.

Перші дві стрічки визначають перехід із стану 1 в стан 2 виконуючи дію 1 - вибір матеріалу ("aluminium") та визначають отримання нагороди +1 за цей перехід стану.

Наступні дві стрічки визначають перехід із стану 1 в стан 3 виконуючи дію 2 вибір матеріалу ("composite") та отримання нагороди +5 за цей перехід стану.

```
MDP.T(1,2,1) = 1;
MDP.R(1,2,1) = 1;
MDP.T(1,3,2) = 1;
MDP.R(1,3,2) = 5;
```

Аналогічним чином визначаються переходи станів та нагороди для інших семистанів.

```
% Стан 2
```

```
MDP.T(2,4,1) = 1;
MDP.R(2,4,1) = 10;
MDP.T(2,5,2) = 1;
MDP.R(2,5,2) = 1;
% Стан 3
MDP.T(3,4,1) = 1;
MDP.R(3,4,1) = 10;
MDP.T(3,5,2) = 1;
MDP.R(3,5,2) = 1;
% Стан 4
MDP.T(4,6,1) = 1;
MDP.R(4,6,1) = 5;
MDP.T(4,7,2) = 1;
MDP.R(4,7,2) = 1;
% Стан 5
MDP.T(5,6,1) = 1;
MDP.R(5,6,1) = 5;
MDP.T(5,7,2) = 1;
MDP.R(5,7,2) = 1;
% Стан 6
MDP.T(6,8,1) = 1;
MDP.R(6,8,1) = 1;
MDP.T(6,9,2) = 1;
MDP.R(6,9,2) = 2.5;
% Стан 7
MDP.T(7,8,1) = 1;
MDP.R(7,8,1) = 1;
MDP.T(7,9,2) = 1;
MDP.R(7,9,2) = 2.5;
% Стан 8
```

```
MDP.T(8,8,1) = 1;
MDP.R(8,8,1) = 0;
MDP.T(8,8,2) = 1;
MDP.R(8,8,2) = 0;
% Стан 9
MDP.T(9,9,1) = 1;
MDP.R(9,9,1) = 0;
MDP.T(9,9,2) = 1;
MDP.R(9,9,2) = 0;
```

Далі визначаємо стани 8 та 9 як кінцеві стани моделі середовища.

```
MDP.TerminalStates = ["s8";"s9"];
```

Потім створюємо середовище навчання агента вибору матеріалу випробувального стенду для навчання агентів з підкріпленням.

```
env = rlMDPEnv(MDP);
```

Вказуємо, що початковий стан для агента завжди є станом 1, за допомогою функції скидання, яка повертає початковий стан агента. Ця функція викликається на початку кожного епізоду навчання та симуляції.

```
env.ResetFcn = @() 1;
```

Фіксуємо генерацію випадкового генератора чисел для повторюваності процесу навчання.

```
rng(0)
```

Для створення агента спочатку потрібно створити таблицю спостережень та дій агента під час взаємодії з середовищем навчання. Встановлюємо швидкість навчання агента 1.

```
obsInfo = getObservationInfo(env);
actInfo = getActionInfo(env);
qTable = rlTable(obsInfo, actInfo);
qRepresentation = rlQValueRepresentation(qTable, obsInfo, actInfo);
qRepresentation.Options.LearnRate = 1;
```

Після цього ініціалізуємо агента використовуючи вище створені таблиці та визначаємо процес дослідження середовища за допомогою алгоритма Epsilon-Greedy [7].

```
agentOpts = rlQAgentOptions;
agentOpts.DiscountFactor = 1;
agentOpts.EpsilonGreedyExploration.Epsilon = 0.9;
agentOpts.EpsilonGreedyExploration.EpsilonDecay = 0.01;
qAgent = rlQAgent(qRepresentation,agentOpts);
```

Для тренування агента визначаємо наступні параметри:

Тривалість тренування складає 200 епізодів, кожен епізод триває 20 кроків. Зупинка тренування відбувається, коли агент отримує середню загальну нагородубільшу ніж 22,5 за 30 епізодів навчання.

```
trainOpts = rlTrainingOptions;
trainOpts.MaxStepsPerEpisode = 20;
trainOpts.MaxEpisodes = 200;
trainOpts.StopTrainingCriteria = "AverageReward";
trainOpts.StopTrainingValue = 22.5;
trainOpts.ScoreAveragingWindowLength = 30;
```

Для тренування агента викликаємо функцію train.

```
trainingStats = train(qAgent,env,trainOpts);
```

На рисунку 4.2, показаний результат тренування агента.

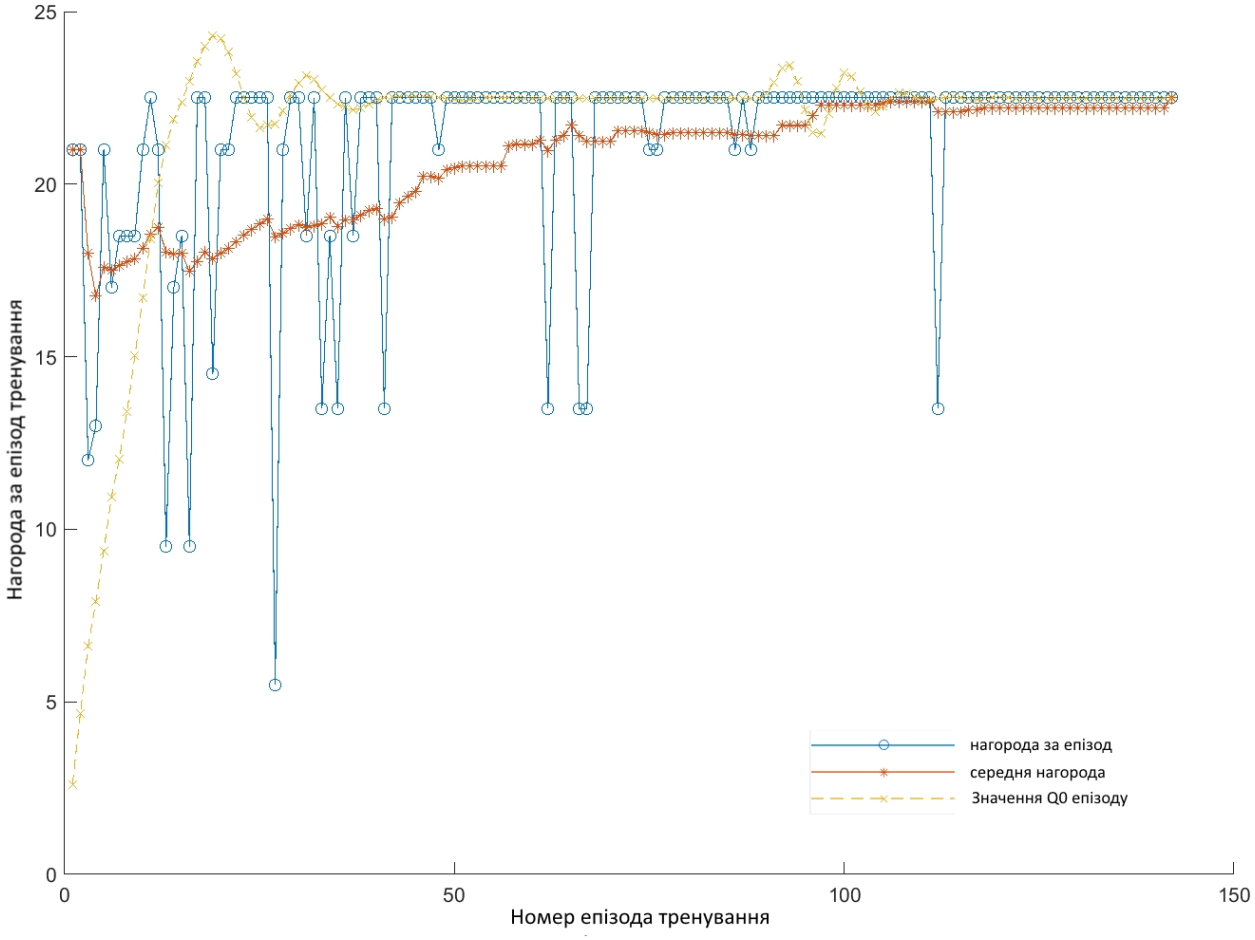


Рис.4.2. Тренування агента вибору матеріала стенда

В таблиці 4.1 показані додаткові параметри тренування та результати тренування агента.

Таблиця 4.1. Параметри тренування агента

Інформація про епізоди	
Кількість епізодів навчання	142
Нагорода за епізод	22.5
Кількість кроків епізоду	4
Значення Q0 епізоду	22.4904
Загальна кількість кроків	568
Середнє значення результатів	

Середнє значення винагороди	22.5
Середня кількість кроків	4
Довжина вікна для усереднення	30
Варіанти навчання	
Апаратний ресурс	ЦП
Швидкість навчання	1
Максимальна кількість епізодів	200
Максимум кроків за епізод	20
Кінцеві результати	
Навчання зупинено	Значення середньої винагороди
Навчання зупинилося із значенням	22.5
Витрачений час	47,021 сек

Для валідації результатів тренування агента виконаємо симуляцію роботи агента за допомогою функції sim у тренувальному середовищі. В результаті симуляції, агент успішно знаходить оптимальні рішення, що дозволяють отримати загальну винагороду 22.5.

```
Data = sim(qAgent,env);
cumulativeReward = sum(Data.Reward)
```

Розглянувши таблицю дій, станів та винагород агента можна оцінити як прийняття рішень стосовно вибору матеріалу виготовлення стенду в процесі вирішення багатокритеріального завдання впливає на винагороду агента у кожному стані процесу прийняття рішень.

```
QTable = getLearnableParameters(getCritic(qAgent));
table = QTable{1}
```

Таблиця 4.2. Таблиця дій, станів та винагород агента

	Алюміній	Композит
Стан 1	16.5358651663305	22.4903673536263
Стан 2	17.1697573172722	9.74858189499054
Стан 3	17.4949573918056	1.47084966447384
Стан 4	7.49836822409867	4.89440194137921
Стан 5	7.02834132847593	1.53320262405713
Стан 6	2.95192006031222	2.49965004691496
Стан 7	0.917466002588707	3.84734717262843
Стан 8	0	0
Стан 9	0	0

Обчисливши середню винагороду агента виконання дій з вибором матеріалу можемо оцінити, що середня винагорода за вибір матеріалу алюміній вища ніж інше рішення.

Average_reward = mean(table, 1)

Таблиця 4.3. Середня винагорода за прийняття рішення про вибір матеріалу

Алюміній	Композит
6.87310037389800	5.16493341089671

На рис. 4.3 показано як змінюється винагорода в процесі переходу станів агента вибору матеріалу кардонового підвісу.

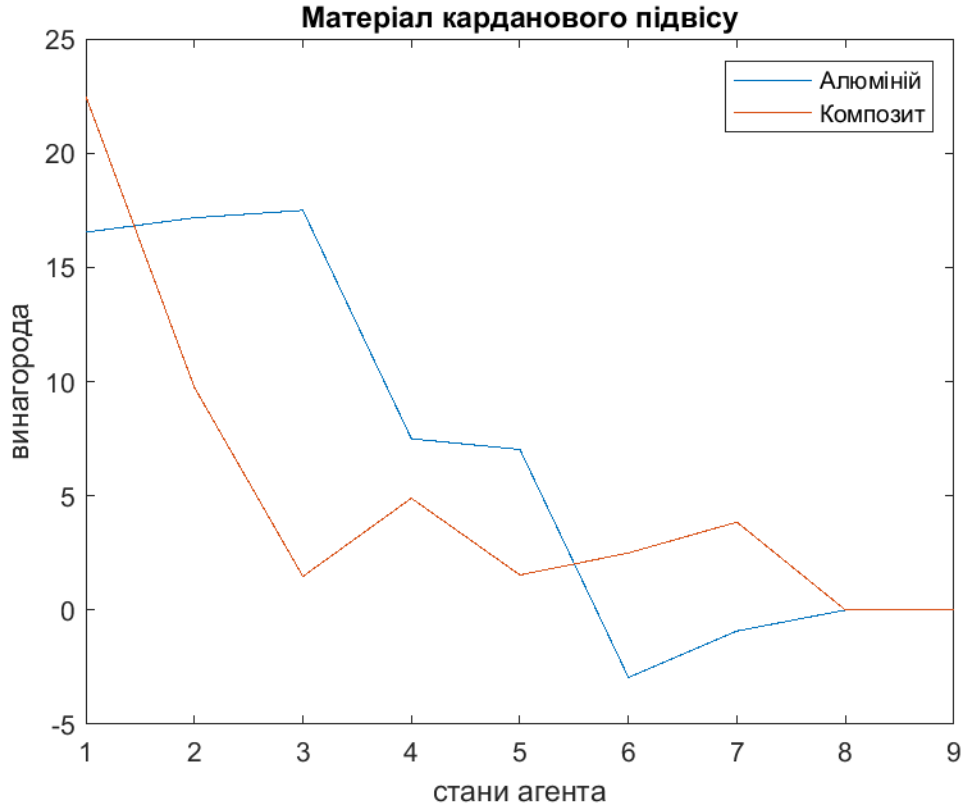


Рис.4.3. Винагорода агента в залежності від станів

Таким чином розраховується всі інші необхідні підсистеми в яких потрібно визначити оптимальне технічне рішення для наступного кроку оптимізації параметрів випробувального стенду.

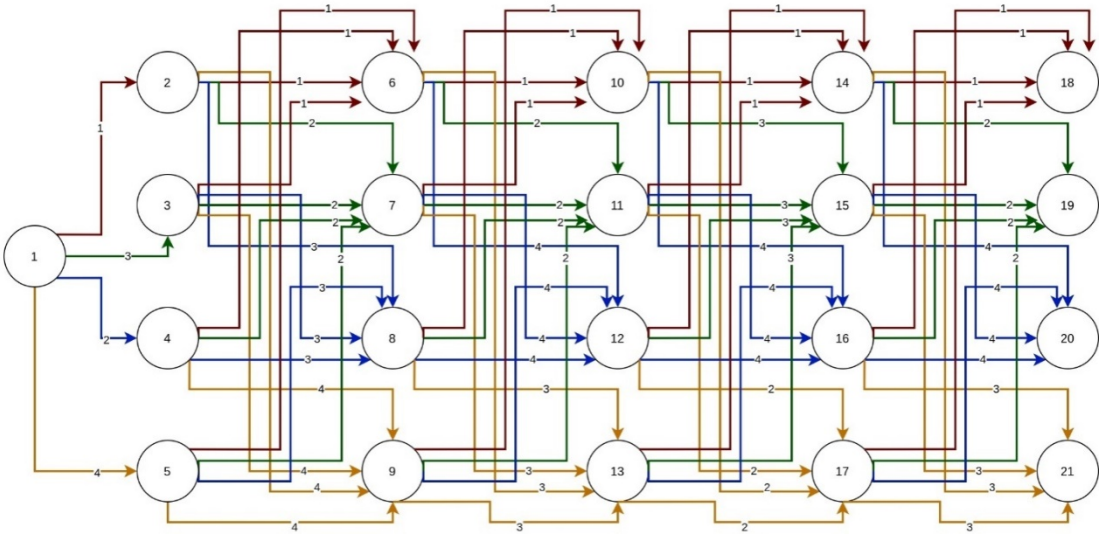


Рис. 4.4. Модель середовища агента для системи кодування положення ВС.

Результати порівняно з широко відомими методами Topsis (TS) [2] та Promethee (PM) [3].

Таблиця 4.4. Результати розробленого методу навчання агента

інерційні датчики	**СКОТ**	**оптичні енкодери**	**індуктосин**
0,534855010307733	1,55541045541538	4,24292579799570	4,59179246151047

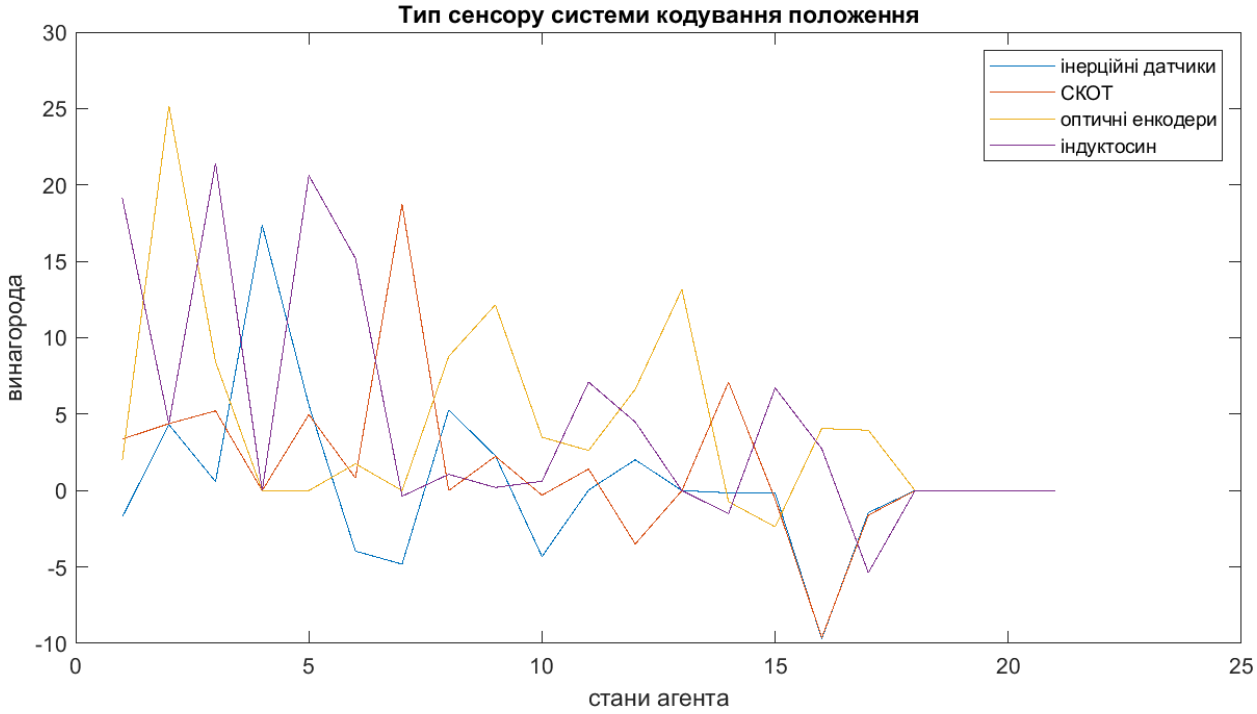


Рис. 4.5 Залежність винагороди від станів агента

Таблиця 4.5. Результати методу Topsis

	інерційні датчики	**СКОТ**	**оптичні енкодери**	**індуктосин**
D+	0,244948974	0,13662601	0,081649658	0,089442719
D-	0	0,121106014	0,206559112	0,18973666
CC	0	0.469891215330286	0.716699605900981	0.679622758982959

Таблиця 4.6. Результати методу Promethee

	Phi	**Phi+**	**Phi-**
оптичні енкодери	0,6000	0,8000	0,2000
індуктосин	0,4667	0,7333	0,2667
СКОТ	-0,0667	0,4667	0,5333
інерційні датчики	-1,0000	0,0000	1,0000

Таблиця 4. 7. Зведені результати порівнюваних методів

	інерційні датчики	**СКОТ**	**оптичні енкодери**	**індуктосин**
MA	4	3	2	1
TS	4	3	1	2
PM	4	3	1	2

Всі три методи однаково визначили 3 та 4 місце. Однак 1 та 2 місце має розбіжності. Якщо поглянути на різницю методу Topsis, то вона дуже незначна між 1 та 2 місцем – 0.71 та 0.67, проте оптичним енкодерам віддана перевага. Результати методу Promethee також віддають незначну перевагу оптичним датчикам, та як видно з графічного представлення результатів при однакових вагових коефіцієнтах всіх критеріїв, оптичні датчики та індуктосини знаходяться по різних сторонах від комбінованого узагальненого критерію (червона лінія).

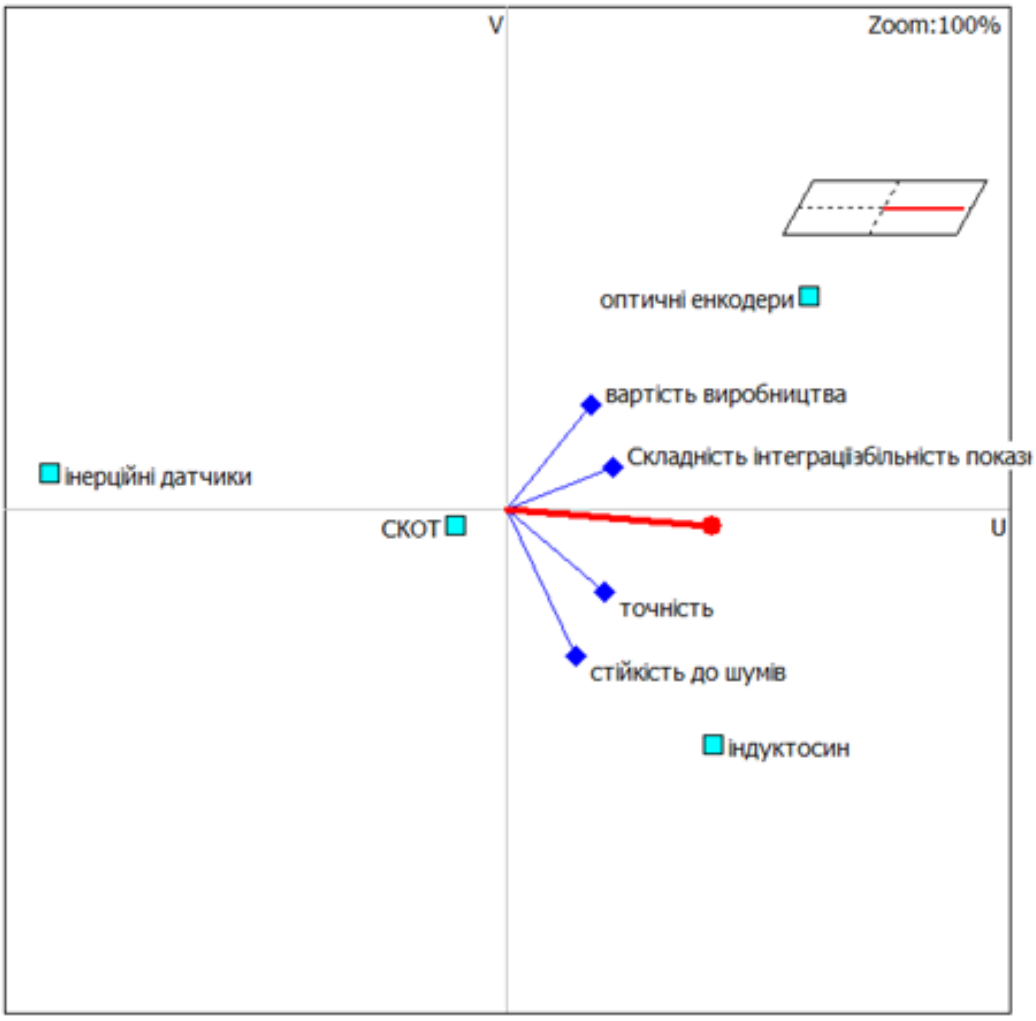


Рис. 4.6. Результати методу Promethee

Метод навчання агента на противагу визначає правильно найкраще рішення для датчика системи кодування положення, що має підтвердження у реальних проектах ВС. Маючи деякі недоліки індуктосини мають найвищу точність та стійкість до шумів, що в прецизійних стендах насамперед грає визначну роль.

Наступна підсистема, що досліджувалась була система опорних підшипників.

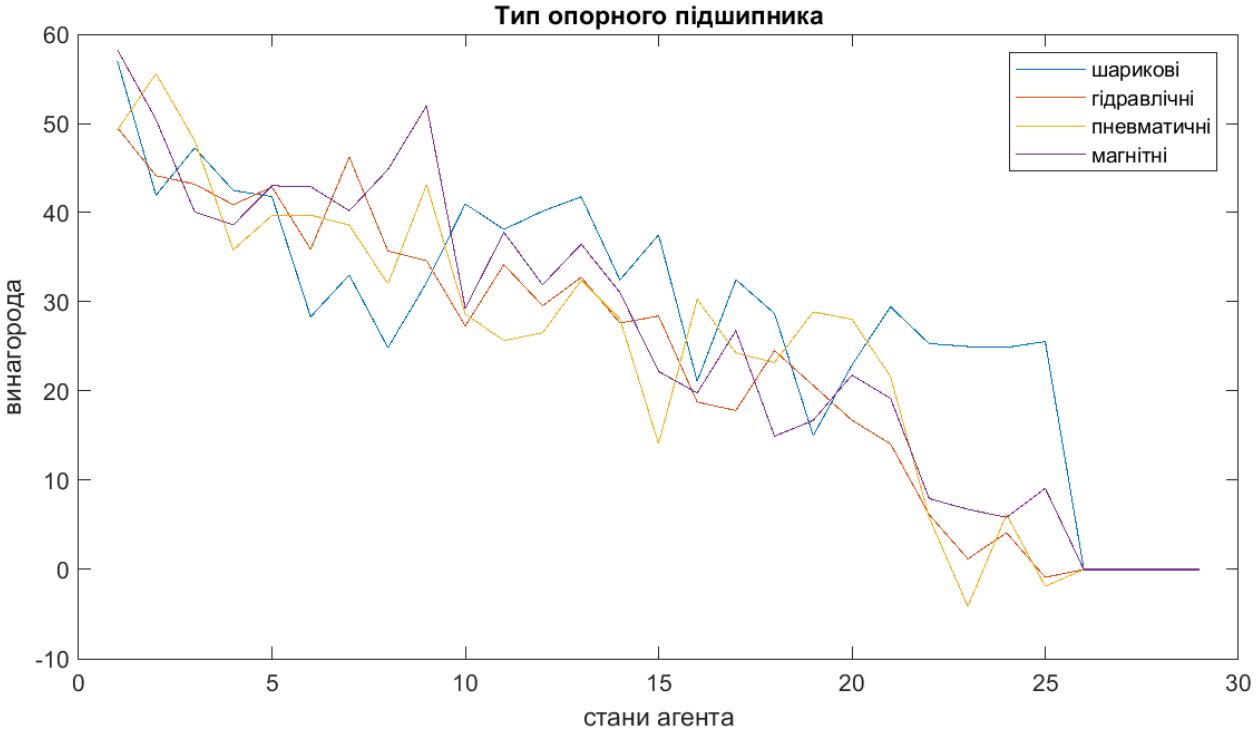


Рис. 4.7. Залежність винагороди від станів агента

Порівняння також включає методи Topsis [2] та Promethee [3]. Зведені результати показані в таблиці 4.8 а,б.

Таблиця 4.8а. Результати методів метод навчання агента, Topsis та Promethee

	шарикові	гідравлічні	пневматичні	магнітні
MA	28.6318	23.3137	24.1282	25.7919
TS	0,7679	0,5357	0,73804341	0,2363
PM	0,1000	-0,2833	0,1500	0,0333

Таблиця 4.8б. Результати прийняття рішень

	шарикові	гідравлічні	пневматичні	магнітні
MA	1	4	3	2
TS	1	3	2	4
PM	2	4	1	3

Постановка завдання включала вагові коефіцієнти, тому робота методів відрізняється від попередньої підсистеми. У ВС застосовуються всі типи, однак у постановці завдання був відданих максимальних коефіцієнт (0.25)

критерію вартості і метод метод навчання агента та TS правильно визначив 1 місце. Щодо інших проектних рішень, то результати всіх методів різняться. Останнє місце в узагальненій таблиці займають гідравлічні рішення. Це найдорожче рішення з точки зору проектування та інтеграції. Пневматичні та магнітні рішення у реальних проектах [4,5,6] часто співставляються; та магнітні опорні підшипники, з точки зору інтеграції, через непотрібність інтеграції системи стисненого повітря та більшої жорсткості переважають над пневматичними.

Таким чином валідація та верифікація результатів розробленого методу навчання агента для задачі багатокритеріального прийняття рішень під час проектування підсистем ВС показала задовільні результати та максимально близька до рішень реальних інженерів-проектувальників.

4.2. Приклад інтеграції САПР та моделі агента оптимізації параметрів механічної конструкції

Існує напрямок розробки випробувальних стендів, що призначені для випробування навігаційного обладнання, який базується на побудові замкнених систем управління двигуном. До цього напряму належать роботи КП СПБ «Арсенал» та Національного Авіаційного Університету, які виготовили тривісний динамічний стенд для випробувань навігаційного обладнання [8]. КП СПБ «Арсенал» виготовили кардановий підвіс та основу стенда, в НАУ на базі навчальної лабораторії випробування навігаційного обладнання кафедри авіаційних комп'ютерно-інтегрованих комплексів протягом роботи над дисертацією було відпрацьовано принципи та можливості реалізації технологічних вузлів випробувальних стендів навігаційного обладнання. Виготовлено відповідні електронні блоки та успішно проходять випробування дослідного зразка. Система управління випробувального комплексу виконана на базі новітніх напівпровідників та програмованих логічних інтегральних схем (ПЛІС).Випробувальний стенд

призначений для застосування у авіації, оборонній та аерокосмічних сферах, для виробництва БПЛА. Є можливість проведення випробувань не тільки окремих елементів чи систем навігаційного обладнання, а всього БПЛА у зборі. Також можливе завдання режимів роботи випробувального комплексу, що відповідають реальним умовам польоту та отримання інформації протягом довгого часу, що відповідає польоту на велику відстань. В порівнянні з найкращими світовими аналогами Acutronic 357 серія (Швейцарія) [6] чи iXBlue EVO 30 (Франція) виробництво такого типу випробувальних стендів коштує значно менше.

Рис. 4.8. Випробувальний стенд кафедри АКІК НАУ

Технічні характеристики випробувального стенда показані у таблиці 4.9.

Таблиця 4.9 Характеристики випробувального стенда НАУ

Параметр	Характеристики
Обертова платформа: - діаметр, мм - сфера обертання, мм - регулювання положення відносно центру перехрещення осей, мм	 1000 1100 +63...-38
Кути повороту, град. та середні радіуси, мм - Курс (Зовнішня вісь) - Крен (Середня вісь) - Тангаж (Внутрішня вісь)	 ±180 (360), 700 ±30 (60), 600 ±180 (360), 530
Діапазон переміщення основи стенда на домкратах (за висотою), мм	±15
Маса, кг - поворотної частини "курсу" - поворотної частини "крену" - поворотної частини "тангажу" - загальна стенда Моменти рушання, Н·м - поворотної частини "курсу" - поворотної частини "крену" - поворотної частини "тангажу"	 232 133 77 432 66.7 37.3 35.8
Роздільна здатність, град. - керування трьома каналами Відображення на ПК, град.	0.35 0.00549

- канал курсу - канал крену і тангажу	0.1
Точність за кутами, град. - канал крену і тангажу - канал курсу	 ±1 ±1.5
Полоса пропускання -3дб, Гц - канал тангажу - канал курсу і крену	 2 1
Кутова швидкість, град/сек - канал тангажу - канал крену - канал курсу	 700 160 150
Точність за кутовою швидкістю, % - канал курсу - канал тангажу та крену	 ±10 ±5
Габаритні розміри стенда, мм	1840x1715x1400
Максимальна маса навігаційного обладнання, що випробовується, кг	100
Можливе встановлення термокамери для розширеного температурного режиму випробувань від -80 до +225 град. С.	так

Для відпрацювання дослідного зразка на базі основи, що була виготовлена КП СПБ «Арсенал» за допомогою САПР Solidworks було спроектовано кріплення датчика кута для каналу курса (рис. 4.9) та новий вузол валу (рис. 4.10) .

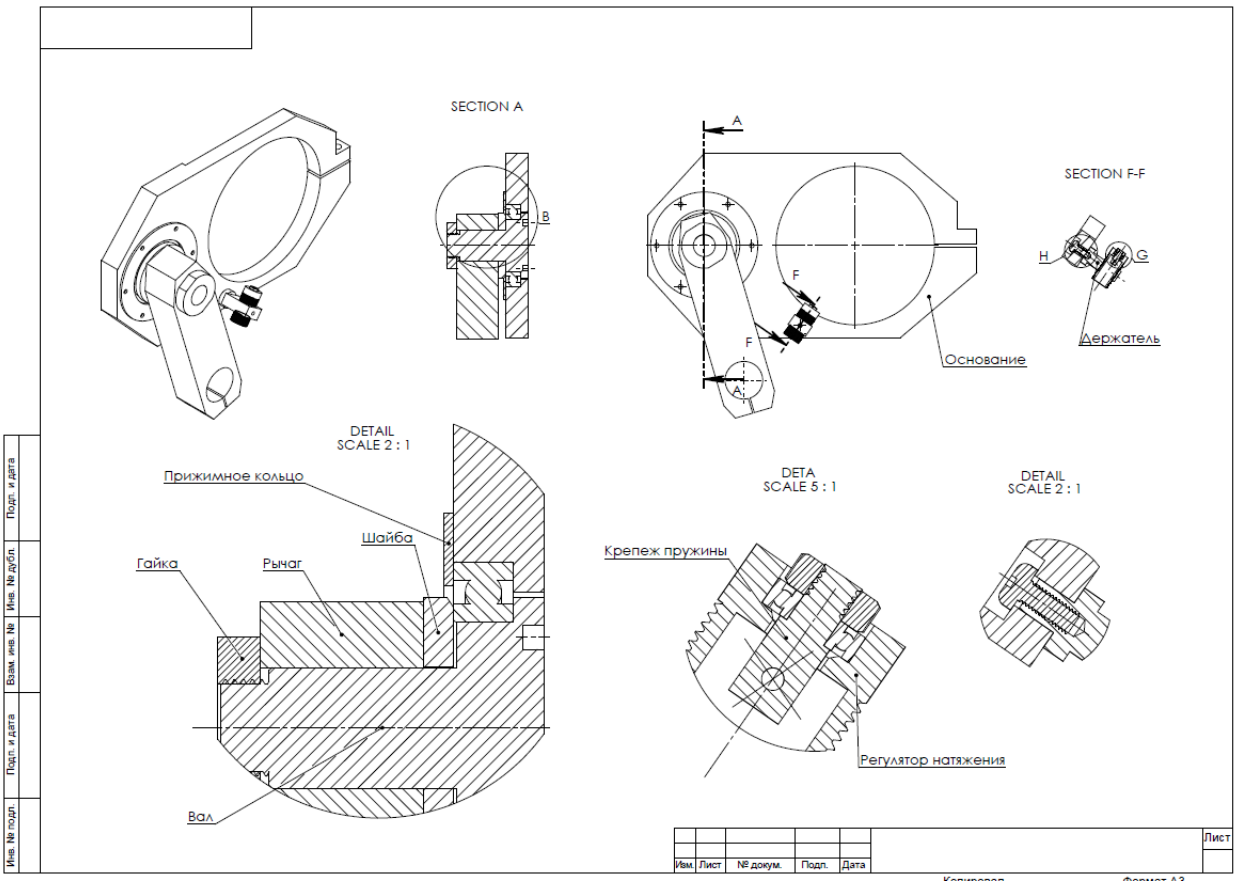


Рис. 4.9. Пакет креслень для вузла кріплення датчика кута для каналу курса

Новий вал та допоміжні елементи вузла були виготовлені на виробничих потужностях ДП «ДержККБ «Луч» (рис. 4.10) та успішно змонтовані на випробувальний стенд. Це дозволило забезпечили надійну та плавну роботу двигуна каналу курсу та вирішило проблему значного механічного люфту в цьому вузлі, що досягав більше 5 градусів [16, 17].

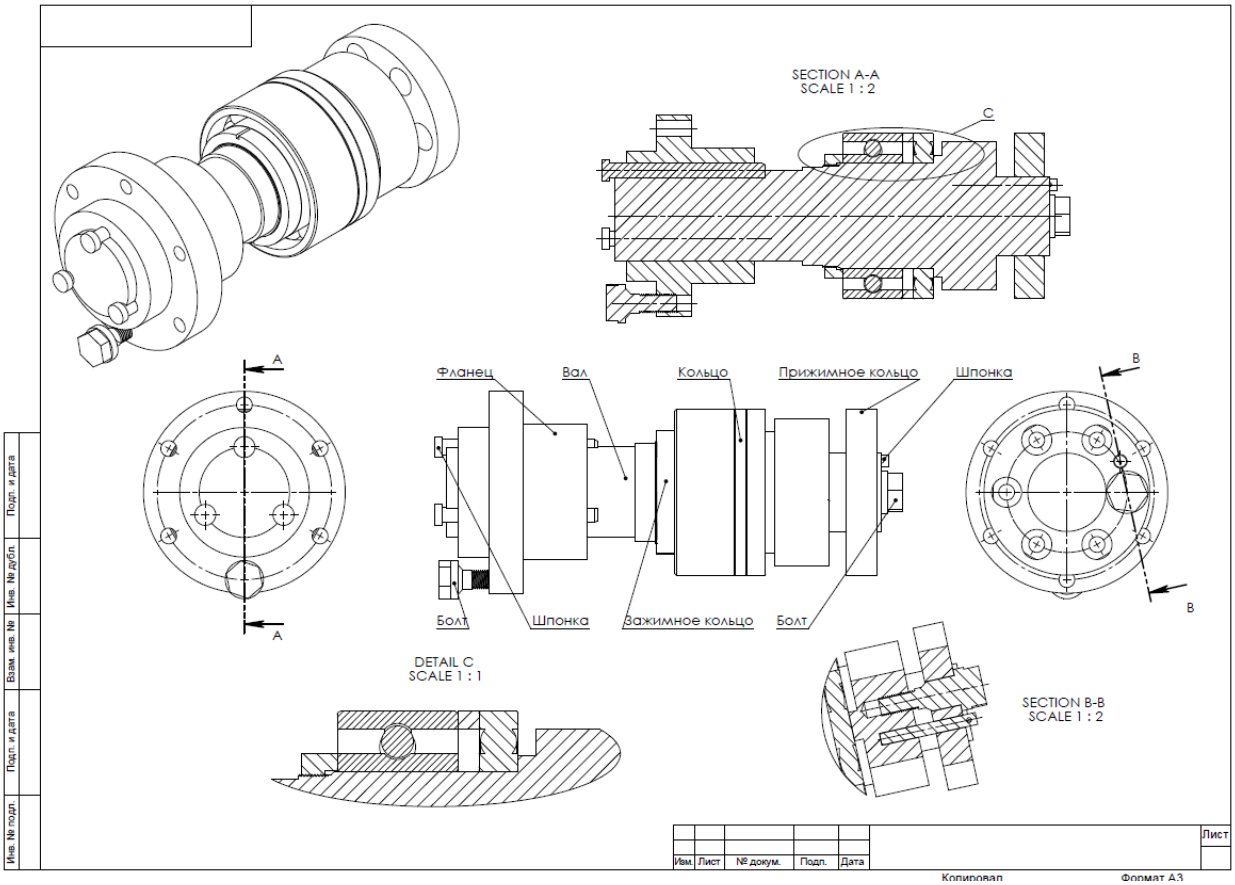


Рис. 4.10. Пакет креслень вузла кріплення двигуна та обертової рами каналу курсу

На рисунку показаний результат віртуальний симуляції конструкції збірки вала в програмі САПР Nastran до якої за допомогою інтерфейсу OLE (ObjectLinking & Embedding) через мову програмування Python підключений агент розрахунку моделі похибки випробувального стенду.

Механізм взаємодії побудований таким чином що віртуальна модель випробовується за допомогою методів вбудованих САПР Nastran, а через інтерфейс OLE до САПР підключається обчислювальний модуль моделі статичної похибки випробувального стенду. Результати обчислень показані в нижній частині інтерфейсу САПР на рисунку 4.11.

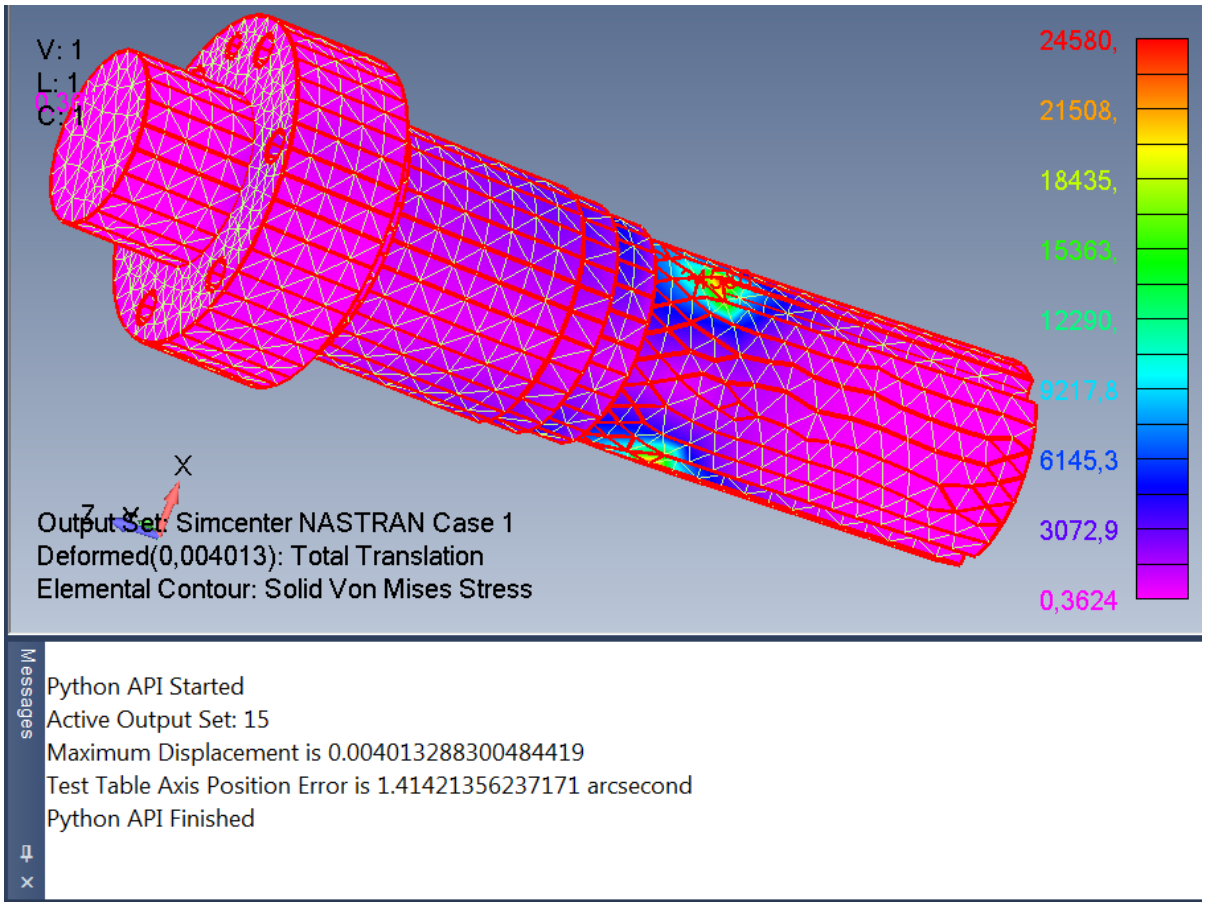


Рис. 4.11. Інтеграція САПР та середовища програмування Python

4.3. Модернізація електричної частини випробувального стенду

Електронна підсистема ВС складається з наступних компонентів:

- збір, передача і попередня обробка даних;
- малопотужна та потужна керуюча електроніка;
- робоча станція з монітором;
- призначений для користувача інтерфейс і програмне забезпечення, що працює в реальному часі;
- збір і передача даних для навігаційного обладнання , що випробовується;
- система електропостачання.

У процесі створення випробувального комплексу для проведення динамічних випробувань навігаційного обладнання БПЛА були вирішені наступні завдання [18, 19]:

1. **Збір, передача і попередня обробка даних**. Завдання вирішене за допомогою ПЛІС та шляхом реалізації промислового інтерфейсу RS-485 з використанням екранованої крученої пари та передавачів, які забезпечують високу швидкість передачі даних поза межами стандартних специфікацій RS-485 / 422, а також володіють завадостійкістю за відповідними стандартами [14, 15].

Електроніка проектувалася за допомогою САПР Altium Designer [9]. Зокрема було спроектовано та виготовлено програматор для ПЛІС Altera, рис. 4.12.

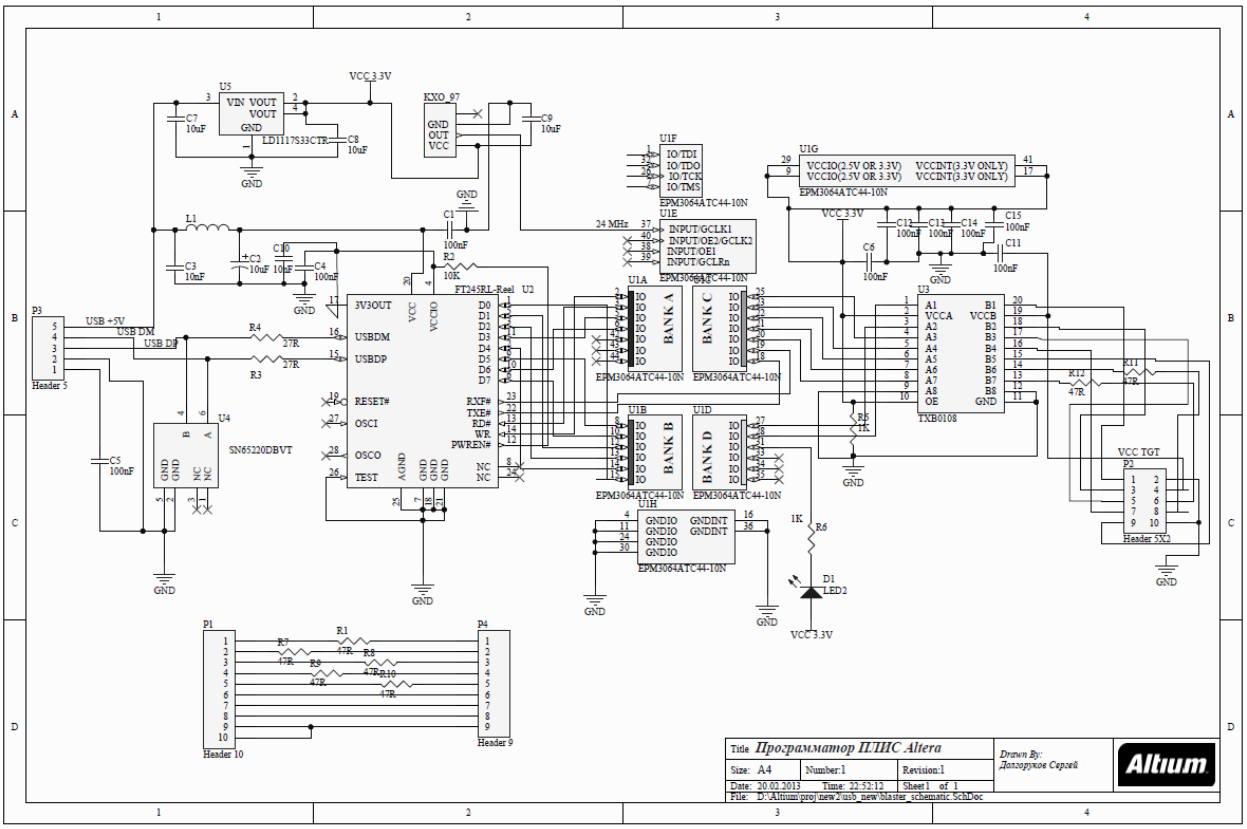


Рис. 4.12. Принципова схема програматора ПЛІС розроблена в САПР Altium

2. **Управління двигуном постійного струму** - це область реального часу, яка була областю використання мікроконтролерів протягом досить

тривалого періоду часу. Нові показники енергоефективності на додаток до нових вимог безпеки підвищують складність проектування автоматизованих систем управління. Така складність нових алгоритмів виводить вимоги до продуктивності і енергоефективності на новий рівень. Для задоволення цих нових вимог необхідно використовувати мікроконтролер або мікропроцесор з більш високою тактовою частотою. Але для досягнення цих більш високих тактових частот постачальники мікроконтролерів і мікропроцесорів зазвичай збільшують кількість кроків конвеєра, додають інтелектуальні алгоритми розгалуження, навіть окремі кеші команд і даних, жертвуючи при цьому детермінізмом в процесі обробки. При перериванні роботи цих мікропроцесорів час затримки може збільшуватися до десятків або сотень тактових періодів. Після цього система управління більше не працює в реальному часі. Ці зміни в алгоритмі управління мають істотні наслідки для роботи системи і можуть мати вирішальне значення в деяких застосунках. Єдиний спосіб досягти необхідного детермінізму при екстремально високій швидкості обчислень полягає в зміні критичних алгоритмів системи за рахунок апаратного прискорення.

Для попередньої обробки даних в реальному часі і відпрацювання алгоритму управління використовується ПЛІС середнього класу Altera з САПР Quartus II, всі модулі були написані на мові VHDL та перевірені тестовим стендом в САПР Active-HDL рис. 4.13 [10].

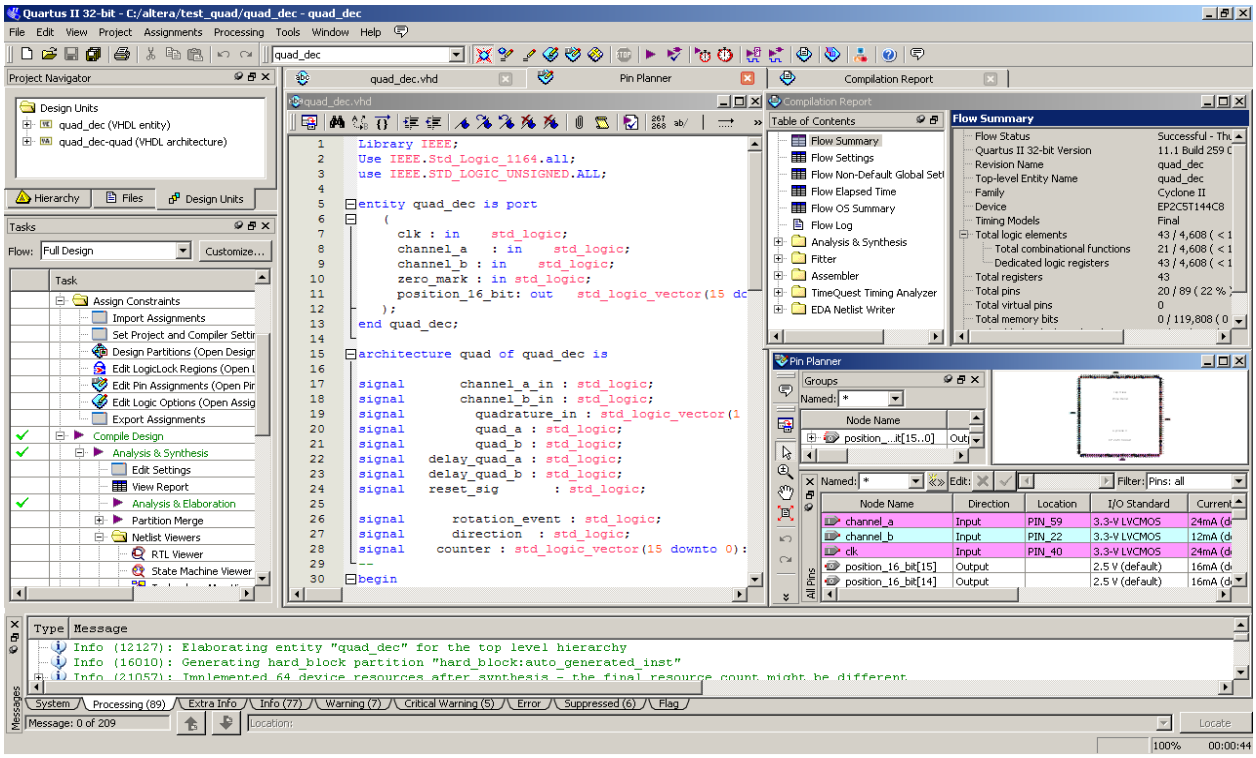


Рис. 4.13. Інтерфейс САПР Quartus під час розробки блоків обробки інформації з датчика кута

САПР Active-HDL [10] надає різні методики для симулювання проекту написаною мовою VHDL. Для досягнення оптимального результату [11] був використаний найбільш підходящий тип симулятора для кожної стадії проектування. На початку циклу проектування використовувалися інтерактивні симулятори для швидкого створення імітаційних входів і миттєвого перегляду проектних реакцій. У міру розробки поліпшеного плану тестування проектного рішення симулятори були перетворені в тестовий стенд VHDL за допомогою майстра тестових стендів. Потім були додані додаткові функції тестування. В кінці були додані деякі сценарії моделювання для автоматизації процесу верифікації та виклику тестового стенду в пакетному режимі поряд з іншими завданнями проектування, такими як синтез, впровадження і порівняння результатів різних етапів проектування. На рис. 4.14 показано результат верифікації роботи інтерфейсу декодування інформації з датчика кута.

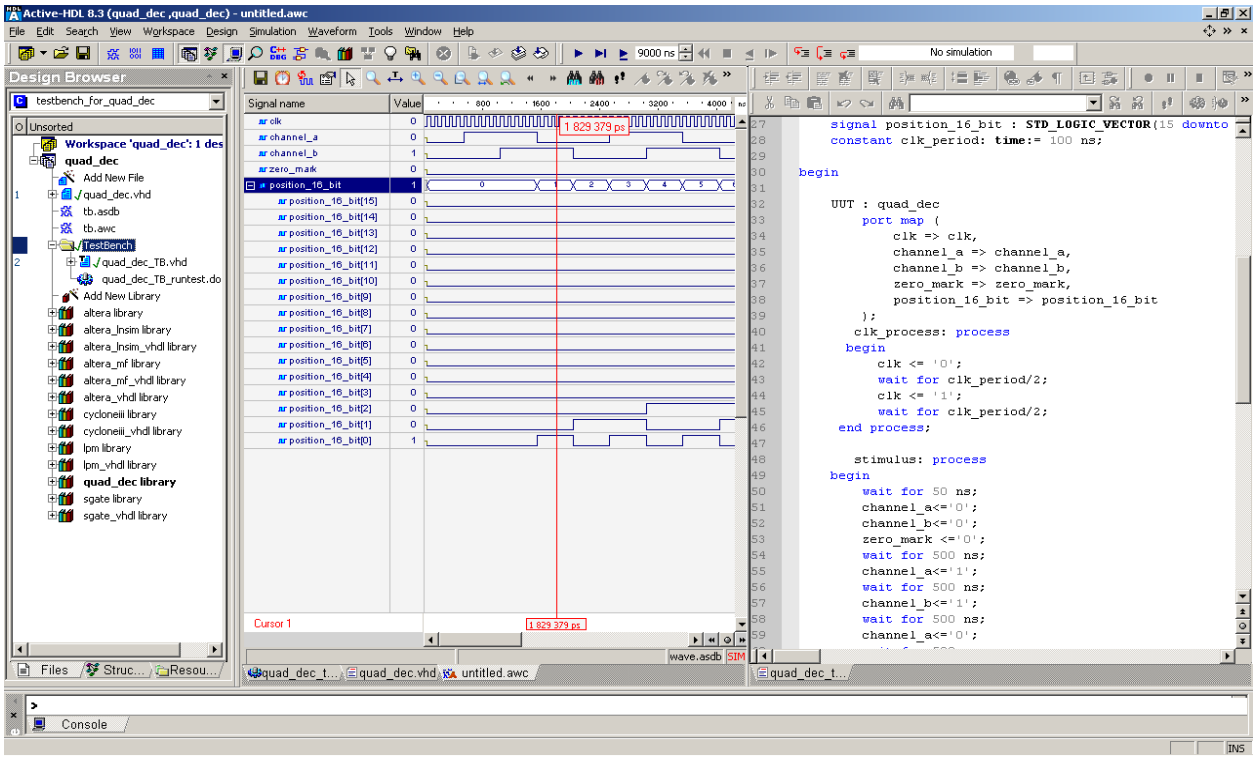


Рис. 4.14. САПР Active-HDL під час верифікації модуля ПЛІС датчика кута

3. **Інтерфейс користувача та програмне забезпечення, що працює в реальному часі.** Завдання було виконане за допомогою САПР LabVIEW [12] (Laboratory Virtual Instrumentation Engineering Workbench). Це середовище розробки і платформа для виконання програм, створених на графічній мові програмування «G» фірми National Instruments (США). Lab VIEW широко використовується в системах збору та обробки даних, а також для управління технічними об'єктами і технологічними процесами. За допомогою LabVIEW та пристрою вводу-виводу інформації через шину PCI було створено робоче місце оператора випробувального комплексу, рис. 4.15.

Рис. 4.15. Інтерфейс користувача та робоче місце оператора випробувального стенду. Зліва від ПК розташовані блоки керування випробувальним стендом.

4.4. Випробування дослідного зразка випробувального стенду навігаційного обладнання

LIS3LV02DL [13]- це тривісний цифровий лінійний акселерометр, який включає чутливий елемент та інтерфейс, здатний приймати інформацію від чутливого елемента та подавати вимірювані сигнали прискорення у зовнішні блоки керування через послідовний інтерфейс I2C / SPI. Сенсорний елемент, здатний визначати прискорення, виготовляється за допомогою спеціалізованого процесу, розробленого STMicroelectronics [13] для виробництва інерційних датчиків з кремнію.

LIS3LV02DL має повну шкалу, що вибирається користувачем, ± 2 g, ± 6 g, і здатний вимірювати прискорення з частотою до 640 Гц для всіх осей. Пропускна здатність пристрою може бути обрана відповідно до вимог застосування.

LIS3LV02DL належить до сімейства продуктів, придатних для різних застосувань:

- виявлення вільного падіння,
- функції, що активуються рухом у портативних терміналах,
- інерційна навігація,
- пристрої для ігор та віртуальної реальності,
- моніторинг та компенсація вібрації.

Для проведення натурного експерименту з випробування навігаційного обладнання даний акселерометр був закріплений на рухомій платформі випробувального стенда та під'єднаний до ПК через інтерфейс USB. Випробування проводились на трьох різних за амплітудою та фазою видах хитавиці.

Перший відрізок (рис. 4.16) показує дані з акселерометра під час хитавиці 1g, середній відрізок фіксує хитавицю понад 4g. Тут ми можемо побачити аномалію за віссю Y акселерометра на кінцевому проміжку середнього відрізку. Показники прискорень осі Y вирівнюються відносно нуля на третьому відрізку, проте вісь Z так і зберігає незначний дрейф. Дрейф в таких типах акселерометрів є відомими недоліком, що значно звужує сферу їх застосування.

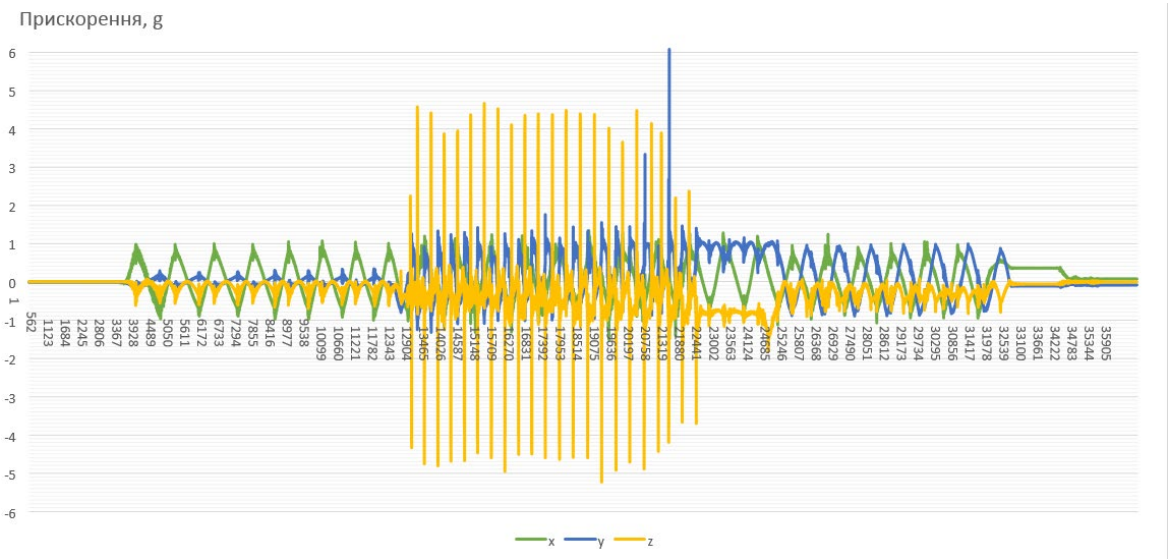


Рис. 4.16. Дані випробування акселерометра MEMS з частотою вибірки 140 Гц

Робоче місце оператора ПК та процес випробування показаний на рис. 4.17.

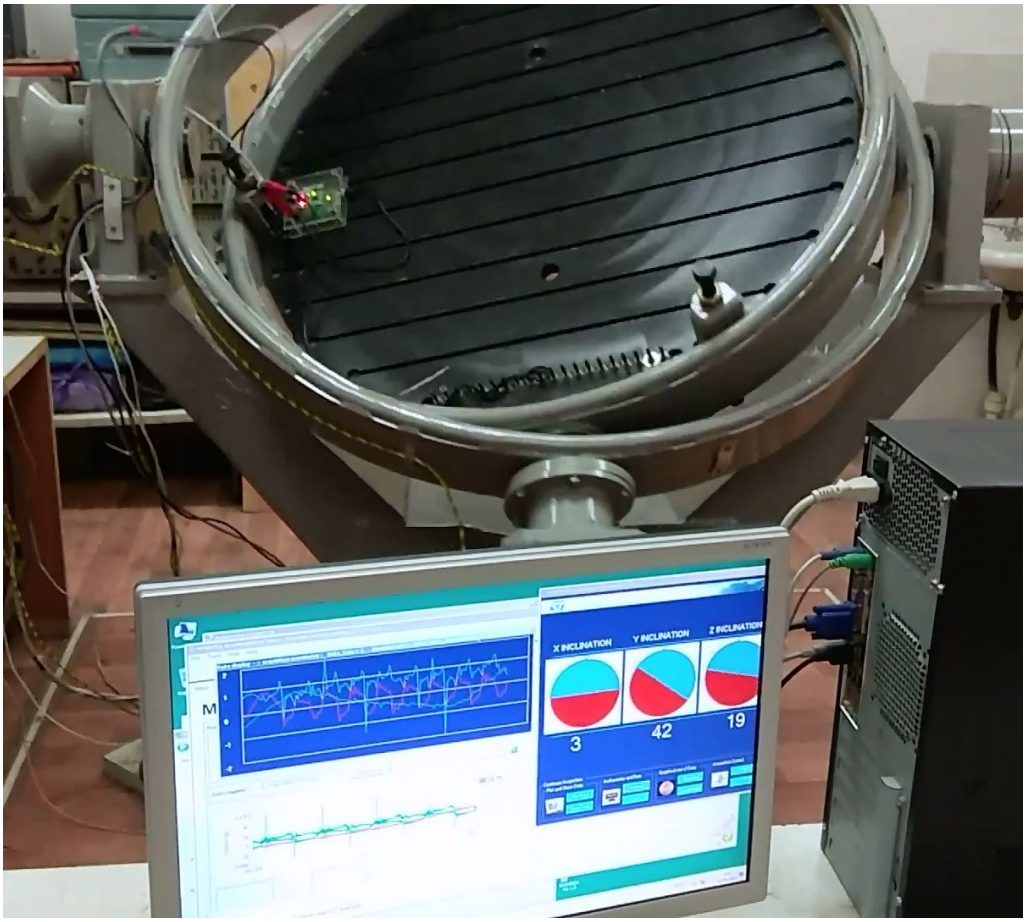


Рис. 4.17. Випробування МЕМС акселерометра LIS3LV02DL

Таким чином підтверджено придатність дослідного зразку випробувального стенда навігаційного обладнання виконувати завдання, що висуваються до тривісних імітаційно-калібрувальних рухомих платформ.

Під час розробки та створення ВС НАУ було використано інструменти САПР для розробки систем випробувального стенду, який складається з трьох осей карданового підвісу.

Процес розробки включав п'ять етапів (Рис. 4.18):

1. Оптимізація та моделювання ВС за допомогою САПР MATLAB і Simulink.

2. Проектування конструкцій ВС за допомогою САПР SolidWorks,

3. Аналіз методом скінчених елементів за допомогою САПР Femap NX Nastran,

4. Розробка електроніки, електронних блоків, верифікація електронних схем за допомогою САПР Altium Designer, САПР Intel Quartus, САПР Active-HDL.

5. Проектування та розробка інтерфейсу користувача за допомогою САПР LabVIEW.

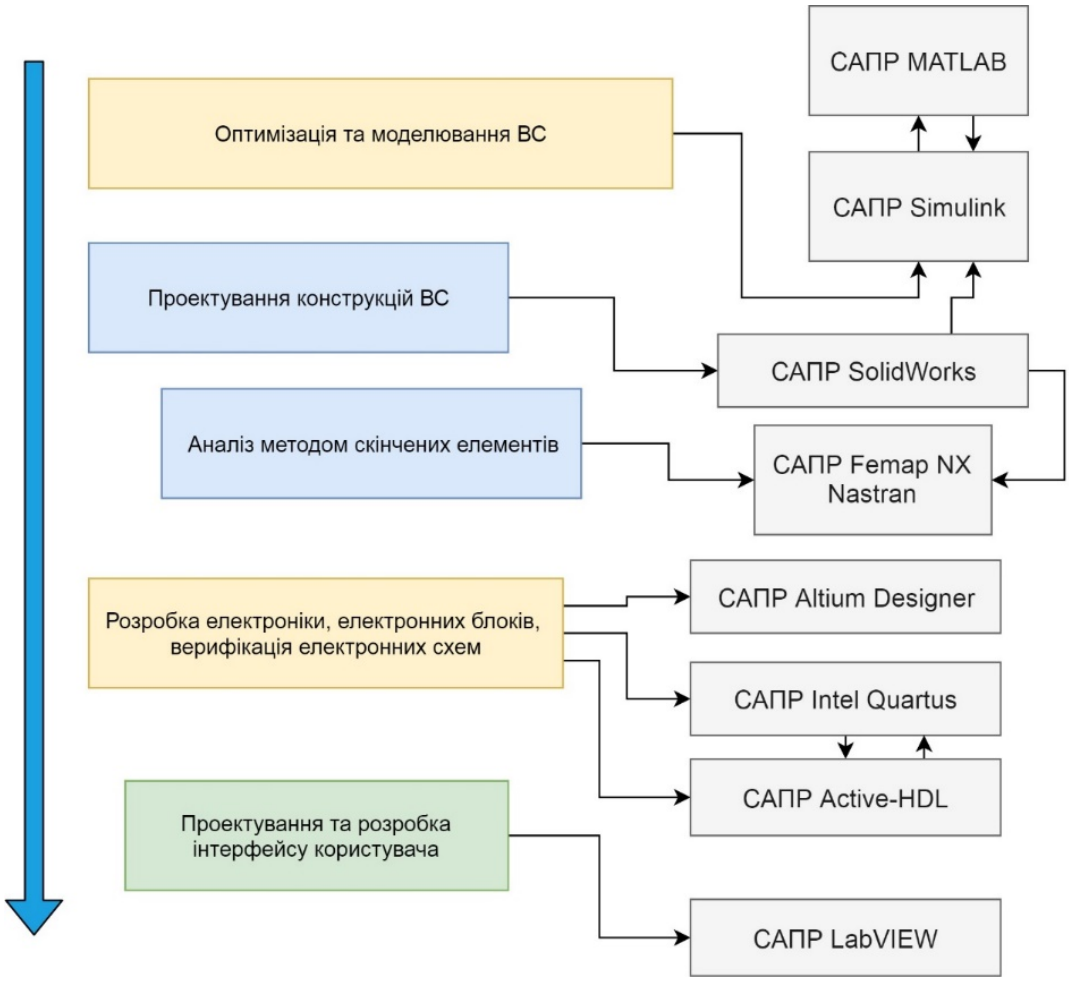


Рис. 4.18. Процес розробки випробувального стенда НАУ

Застосування САПР для проектування випробувального стенду знизило витрати на розробку, а час проектування скоротився більш ніж на два роки.

Висновки до розділу 4

Проведено моделювання агентів для проектування ВС в середовищі Matlab та Simulink, а саме: агента вибору матеріалів карданного підвісу, агента системи опорних підшипників, агента системи кодування положення. Результати розробленого методу навчання агента для задачі багатокритеріального прийняття рішень під час проектування підсистем ВС показали задовільні результати та максимально близькі до рішень реальних інженерів-проектувальників.

Проведено моделювання інтеграції САПР Femap Nastran та агента механічної збірки відповідального за розрахунок моделі похибки випробувального стенду в середовищі програмування Python, що дозволило заощадити час проектування збірки вала опорної частини ВС.

Здійснено реалізацію підстем ВС на основі запрограмованої САПР для створення дослідної моделі випробувального імітаційного динамічного трьох осевого стенду.

Спроектована і побудована система випробування навігаційного обладнання, навчально-наукової лабораторії авіаційних комп’ютерно-інтегрованих комплексів НАУ, що дозволило перевірити ефективність системи автоматизованого проектування для побудови ВС.

ВИСНОВКИ

У дисертаційній роботі розглянуто актуальні проблеми проектування випробувальних стендів навігаційного обладнання, які були теоретично та експериментально досліджені.

Представлений у дисертації підхід до проектування випробувальних стендів полягає в тому, щоб розкласти загальне завдання проектування ВС на комплекс різних процедур мультиагентної системи. Процес розробки випробувальних стендів, сумісний з вимогами четвертої промислової революції вимагає зміни як методів, так і систематичної процедури в процесі розробки – це напрямок розвитку промисловості проектування випробувальних стендів на найближчі десятиліття.

Завдання мультиагентної системи полягає в тому, щоб зібрати разом відповідні САПР-інструменти для формування середовища розробки і підтримати взаємодію інструментів САПР за допомогою моделей, системних інтерфейсів, процесів та технології методів проектування ВС. Розроблена система автоматизованого проектування - це симулятор підсистем ВС, віртуальний конструктор з можливостями оптимізувати граничні параметри між взаємопов’язаними підсистемами ВС для наступного аналізу.

Головні переваги, що забезпечує застосування багато-агентної системи проектування:

1. Зв'язки між підсистемами - агенти враховують всі конфлікти цілей.

2. Можна додавати і прибирати агентів з системи проектування за необхідністю.

3. Архітектура моделі політики вирішує завдання координації агентів.

4. Модель агента максимально гнучка для адаптації під будь-які завдання проектування.

5. Імітаційне середовище з даними для навчання агентів – це розроблене ядро системи проектування, яку потім за необхідність можна розширювати.

6. Мережа проектування ВС може перетинати кордони компаній та виробничих організацій.

7. Учасники процесу проектування ВС взаємодіють між ієрархічними рівнями.

8. Всі учасники процесу проектування ВС можуть обмінюватись інформацією один з одним.

9. Система проектування має рівень автономії прийняття рішень. Під час проектування ВС, оператор визначає граничні параметри багатоагентної системи, в межах якої системою агентів з ШІ, дозволено контролювати автоматизовані операції. Багатоагентна система самостійно контролює своє середовище в межах заданих системних меж. Якщо агент використовує закономірність, отриману на етапі навчання, він може напівавтомно виконати відповідну вивчену дію. Наприклад, він самостійно активує зупинку процесу проектування, коли виявляє неправильне визначення правильності виконання проектної процедури в процесі проектування ВС навігаційного обладнання.

10. Автономне навчання - автоматичне отримання даних та навчання. Автоматизована багатоагента система проектування ВС самостійно збирає дані, генерує нові знання та використовує їх для корекції свого процесу прийняття рішень (вироблення правил) у заданих системних обмеженнях. Адаптація правил відбувається повністю автоматизовано без якого-небудь втручання, але лише в рамках раніше заданих обмежень.

Основні досягнуті результати:

1. Запропоновано багато-агентний підхід для автоматизації проектування ВС, який використовує агентів-проектувальників з елементами ШІ - навчання з підкріпленням для вирішення завдання багатокритеріального прийняття рішень, що дає в результаті покращення пошуку проектних рішень з точки зору використання попередніх знань. Підхід не тільки дозволяє легко включати нові функції нагород за необхідності, але і зважувати їх відносну важливість в залежності від потреб конкретного завдання проектування випробувального стенду.

2. Вперше розроблена структура комплексу програмного середовища САПР, в якій поєднана взаємодія між ієрархічними рівнями інженерів-

проектувальників та інтелектуальних агентів з власними моделями поведінки, що дозволяє скоротити час проектування та забезпечити взаємодію зацікавлених учасників процесу за межами організацій.

3. Розроблено метод побудови САПР ВС, що забезпечує інтеграцію існуючих САПР в єдиний автономний комплекс, в якому людина-оператор визначає критерії та обмеження, в межах яких багато-агентній системі з підтримкою ШІ дозволяється управляти автоматизованими операціями, що призводить до економії часу проектування.

4. Вперше запропоновано метод розв'язання задачі багатокритеріального прийняття рішень під час проектування, який при його імплементації використовує агенти з автономним навчанням, що самостійно збирають дані, генерують нові знання і використовують їх для коригування процесу прийняття рішень, що дозволяє безпосередньо оптимізувати цільові параметри без необхідності визначати модель та форму апроксимації цих функцій.

5. На основі запрограмованої САПР розроблено, побудовано, налаштовано випробувальний стенд навігаційного обладнання з наступними характеристиками: габаритні розміри стенда - 1840x1715x1400 мм; максимальна маса навігаційного обладнання, що випробовується - 100 кг; діаметр обертової платформи - 1000 мм; кути повороту та середні радіуси - зовнішня вісь ±180 (360) град., 700 мм; середня вісь ±30 (60) град., 600 мм; внутрішня вісь ±180 (360) град., 530 мм; точність за кутами, крену і тангажу ±1 град., канал курсу ±1.5 град.; кутова швидкість каналу тангажу - 700 град/сек, каналу крену - 160 град/сек, каналу курсу - 150 град/сек.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ ДО РОЗДІЛУ 4

1. “Reinforcement Learning Toolbox Documentation”. MathWorks. [Електронний ресурс]. Режим доступу: https://www.mathworks.com/help/reinforcement-learning/ (дата звернення 29.03.2021).

2. J. Papathanasiou, N. Ploskas, “TOPSIS,” in *Multiple Criteria Decision Aid*. *Springer Optimization and Its Applications*, vol. 136, pp. 1-30. Springer International Publishing, 2018.

3. J. Papathanasiou, N. Ploskas, “PROMETHEE,” in *Multiple Criteria Decision Aid. Springer Optimization and Its Applications*, vol. 136, pp. 57-89. Springer International Publishing, 2018.

4. L. A. Demore, R. A. Peterson, L. B. Conley; H. Havliscek, N. P. Andrianos, “Design study for a high-accuracy three-axis test table,” *Journal of Guidance, Control, and Dynamics*, vol. 10, no. 1, pp. 104-114, 1987.

5. “One-, Two- And Three-Axis Rate And Positioning Tables”. Ideal Aerosmith. [Електронний ресурс]. Режим доступу: https://www.ideal-aerosmith.com/products/1-2-or-3-axis-rate-and-positioning-tables/ (дата звернення 29.03.2021).

6. “Simulation & Test Rate Tables”. Acutronic. [Електронний ресурс]. Режим доступу: https://www.acutronic.com/simulation-test (дата звернення 29.03.2021).

7. M. Tokic, “Adaptive ε-Greedy Exploration in Reinforcement Learning Based on Value Differences,” in *Advances in Artificial Intelligence*, R. Dillmann, J. Beyerer, U. D. Hanebeck, T. Schultz (eds), Lec. No. in Comp. Sc., vol. 6359. Springer Berlin Heidelberg, 2010, doi: 10.1007/978-3-642-16111-7_23.

8. V. M. Sineglazov and S. O. Dolgorukov, “HardWare-In-The-Loop simulation table for UAV navigation complexes testing*,” in Proc. IEEE 2nd Int. Conf. Actual Problems of Unmanned Air Vehicles Developments Proceedings (APUAVD)*, (Kyiv, Ukraine), 2013, pp. 141-145, doi.: 10.1109/APUAVD.2013.6705307.

9. “PCB Design Software and Tools”. Altium. [Електронний ресурс]. Режим доступу: https://www.altium.com (дата звернення 29.03.2021).

10. “Active-HDL. FPGA Simulation”. Aldec. [Електронний ресурс]. Режим доступу: https://www.aldec.com/en/products/fpga_simulation/active_hdl_student (дата звернення 29.03.2021).

11. “How to Simulate Designs in Active-HDL”. Application Notes, Documentation, Aldec. [Електронний ресурс]. Режим доступу: https://www.aldec.com/en/support/resources/documentation/articles/1060 (дата звернення 29.03.2021).

12. “LabVIEW”. National Instruments. [Електронний ресурс]. Режим доступу: https://www.ni.com/ru-ru/shop/labview.html (дата звернення 29.03.2021).

13. “3-axes digital output linear accelerometer”. STMicroelectronics. [Електронний ресурс]. Режим доступу: https://www.st.com/en/mems-and-sensors/lis3lv02dl.html (дата звернення 29.03.2021).

14. C. Kinnaird, “RS-485 for Digital Motor Control Applications,” Texas Instruments Incorported, Application Report SLLA143, 2003.

15. F. Alicke, M. Feulner, F. Dehmelt, A.Verma and G. Becke, “Comparing Bus Solutions,” Texas Instruments Incorported, Application Report SLLA067C, 2017.

16. V. M. Sineglazov and S. O. Dolgorukov, “Optimal choice of the technical means of rate, pitch and roll channels subsystems of navigation equipment simulation table,” *Electronics and control systems*, no. 2, pp. 33-39, 2014.

17. V. M. Sineglazov and S. O. Dolgorukov, “Identification of mechanical backlash of navigation equipment testing table,” *Electronics and control systems*, no. 3, pp. 104-108, 2014.

18. V. M. Sineglazov and S. O. Dolgorukov, “UAV equipment test table for preflight testing,” *Electronics and control systems*, no. 3, pp. 126-132, 2015.

19. V. M. Sineglazov and S. O. Dolgorukov, “Test bench of UAV navigation equipment,” *in Proc. IEEE 3rd Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, Kyiv, Ukraine, 2014, pp. 108-111, doi: 10.1109/MSNMC.2014.6979743.

Додаток 1. Технічні вимоги до випробувальних стендів

Таблиця 1. Технічні характеристики ВС для випробування поплавкових ДКШ та ВТГ, ММГ, РВГ, СВГ

Технічні характеристики ВС	Для випробування		
	Поплавкових прецизійних ДКШ, ВТГ	Поплавкових ДКШ середньої точності	Поплавкових ДКШ низької точності, ММГ, РВГ, СВГ
Число осей карданового підвісу ВС	1	1	1
Діапазон встановлюваних кутових швидкостей, °/с	≤10	<30-50	А: <50 Б: ± 1000
Мінімальна кутова швидкість, що задається, °/с	±0,1	±1	1% від максимальної величини
Дискретність кутової інформації по осі підвісу ВС, кут. с/біт	0,04-0,08	0,62	-
Стабільність кутової швидкості, що задається, ° / с	0,00001	0,00005-0,00001	0,001-0,005
Похибка вимірювання кутової швидкості, що задається, %	≤0,001	≤0,003	А: ≤0,03 Б: ≤0,2
Похибка задання гармонічних коливань: за амплітудою, % за фазою, Гц	-	-	 0,1 0,1
Вантажопідйомність, кг	≤20	≤20	≤20

Таблиця 2. Технічні характеристики ВС для випробування ДНГ

Технічні характеристики ВС	Для випробування	
	Сучасних прецизійних ДНГ	Перспективних широкодіапазонних ДНГ
Число осей кардanового підвісу навколо яких задається обертання, фіксовані кутові положення	1 2	1 2
Діапазон встановлюваних кутових швидкостей, °/с	10	≤1000
Мінімальна кутова швидкість, що задається, °/с	0,1	1
Дискретність кутової інформації по осі підвісу ВС, кут. с/біт	0,04-0,08	2-6
Стабільність кутової швидкості, що задається, ° / с	0,0005-0,00001	0,005-0,01
Похибка вимірювання кутової швидкості, що задається, %	0,001	0,005
Діапазон фіксованих кутів розвороту навколо 2 і 3 осей кардanового підвісу, град.	±180	
Точність виставлення фіксованого кута розвороту, кут. с.	±30	
Стабільність кутового положення 2 і 3 осей кардanового підвісу, кут. с.	±5	
Вантажопідйомність, кг	20	

Таблиця 3. Технічні характеристики ВС для випробування широкодіапазонних ВОГ

Назва параметра	Значення параметра
1. Число осей карданового підвісу	1
2. Діапазон кутових швидкостей, які задаються, °/с	≤1000
3. Мінімальна кутова швидкість, яка задається, °/с	1
4. Дискретність кутової інформації по осі підвісу ВС, кут. с/біт	2-6
5. Стабільність кутової швидкості, що задається, °/с	0,005-0,001
6. Похибка вимірювання кутової швидкості, яка задається, %	0,005-0,01

Таблиця 4. Технічні характеристики ВС для випробування ЛГ

Назва параметра	Значення параметра
1. Число осей карданового підвісу	1
2. Діапазон кутових швидкостей, які задаються, °/с	≤1000
3. Мінімальна кутова швидкість, яка задається, °/с	1
4. Дискретність кутової інформації по осі підвісу ВС, кут. с/біт	0,2-0,6
5. Стабільність кутової швидкості, що задається, °/с	0,001-0,005
6. Похибка вимірювання кутової швидкості, яка задається, %	0,001

Додаток 2. Технічні характеристики ВС

Серія DC-ACI 125

Рис. 1. Одновісний поворотний стенд DC-ACI 125

Габарити :

висота, мм -905;

діаметр стола, мм -500;

ширина, мм — 500;

Виріб, що випробується: вага ,кг - не більше 454;

Таблиця 1. Технічні дані одновісних поворотних стендів

Технічні характеристики	DC-1125-020	DC-1125-060	AC-1125-100	AC-1125-350
Роздільна здатність (дискретність) кута повороту, град	0.00001	0.00001	0.00001	0.00001
Кутова швидкість, град/сек.	+/-1000	+/-600	+/-1200	+/-1200
Роздільна здатність кутової швидкості, град/сек	0.00001	0.00001	0.00001	0.00001
Стабільність кутової швидкості,% більше 360° більше 10°	0.0001 0.001 0.01	0.0001 0.005 0.05	0.0001 0.005 0.05	0.0001 0.005 0.05
Момент, Нм	22	60	100	350
Прискорення, град/сек^2	2500	6800	12000	45000
Смуга пропускання, Гц	50	60	60	60
Керуючий пристрій	ACT3001 -PA	ACT3001 -PA	ACT3000	ACT3000

Серія AC-DC130

Рис. 2. Одновісний поворотний стенд AC-DC130

Габарити :

висота загальна, мм -1702;

висота від підлоги до стола, мм

без термокамери -991;

з термокамерою -1158;

діаметр стола, мм -445

Виріб, що випробується:

вага ,кг - не більше 227;

висота максимальна, мм - 445;

діаметр максимальний, мм - 305- 559;

струмовідводи – 30 ліній.

Таблиця 2. Технічні дані серії AC-DC130

Технічні характеристики	AC-DC130-10	AC-DC130-20	AC-DC130-80	AC-DC130-150
Роздільна здатність	0.0000	0.0000	0.0000	0.00001

(дискретність) кута повороту, град				
Кутова швидкість, град/сек.	±10000	±1000	±1000	±1000
Роздільна здатність кутової швидкості, град/сек	0.00001	0.00001	0.00001	0.00001
Стабільність кутової швидкості,% - більше 360°	0.0001	0.0001	0.0001	0.0001
Момент, Нм	22	60	100	350
Прискорення, град/сек^2	700	5750	14500	25000

Термокамера (опція):

діапазон температури, °С : -55+85;

стабільність °С : +/- 1.0;

зміна температури, °С/хв.:

LN_2 або *CO_2 :* 5 (охолодження та нагрів);

механічний рефрижератор (опція) : 1.5 (охолодження); 5 (нагрів)

Серія DC-AC2256

Серія 2256 розроблена для високоточних випробувань інерційних чутливих елементів та НО. Стенд дозволяє одночасно випробувати декілька модулів НО, що значно зменшує собівартість випробувань або калібрування.

Габарити :

висота загальна, мм -1925;

ширина, мм — 1825;

основа, мм - 950х1275;

стіл , мм - 500; матеріал - анодований алюміній з отворами М6

Рис. 3. Двовісний поворотний стенд DC-AC2256

Виріб, що випробується:

вага номінальна (максимальна) ,кг - 40 (120);

висота максимальна, мм - 445;

- діаметр максимальний, мм — 305-559;

- струмопідводи - 60 ліній : 20А , 400VAC; 2А,150VDC

Таблиця 3. Порівняння технічних даних типів DC та AC

Технічні характеристики	DC2256		AC2256	
	Внутрішня вісь	Зовнішня вісь	Внутрішня вісь	Зовнішня вісь
Роздільна здатність (дискретність) кута повороту, град	0.00001	0.00001	0.00001	0.00001
Кутова швидкість, град/сек	±1000	±200	±1200	±600
Роздільна здатність кутової швидкості, град/сек	0.00001	0.00001	0.00001	0.00001
Стабільність кутової швидкості,% - більше 360°	0.0001	0.0001	0.0001	0.0001
Прискорення, град/сек^2	1200	400	3000	700
Тип двигунів	DC	DC	AC	AC

Серія АС2267-ТС

Серія АС2267-ТС містить досить об’ємну термокамеру і також розроблена для випробувань інерційних чутливих елементів та НО. Стенд дозволяє одночасно випробувати кілька модулів НО.

Рис. 4. Двовісний поворотний стенд АС2267-ТС

Габарити :

висота загальна, мм -1915;

ширина, мм - 2420;

основа, мм - 900х1870;

стіл , мм - 660; матеріал - анодований алюміній з отворами М6

Виріб, що випробується:

вага номінальна (максимальна), кг-40(120);

висота максимальна, мм — 500;

діаметр максимальний, мм - 600;

струмовідводи : 20А , 400VAC;

2А,150VDC

Термокамера :

діапазон температури, °С : -50+90;

стабільність ,°С : ±1.0;

зміна температури, °С/хв. :

LN_2 або CO_2 : 5 (охолодження та нагрів);

механічний рефрижератор : 1.5 (охолодження); 5 (нагрів)

Таблиця 4. Технічні дані АС2267-ТС

Технічні характеристики	АС2267-ТС	
	Внутрішня вісь	Зовнішня вісь
Роздільна здатність (дискретність) кута повороту, град	0.00001	0.00001
Кутова швидкість, град/сек.	±1000	±600
Роздільна здатність кутової швидкості, град/сек.	0.00001	0.00001
Стабільність кутової швидкості,% - більше 360°	0.0001	0.0001
Прискорення, град/сек^2	3000	200
Тип двигунів	АС	АС

Серія АС3350

Рис. 5. Тривісний поворотний стенд АС3350

Серія АС3350 розроблена для випробувань систем інерційної навігації, сенсорів руху та інших інерційних компонентів.

Габарити :

висота загальна, мм -2280;

ширина, мм - 2000;

Виріб, що випробується:

вага номінальна ,кг - 40 ;

висота максимальна, мм - 500;

діаметр максимальний, мм - 500;- струмовідводи : 20А , 400VAC; 2А,150VDC

Таблиця 5. Технічні характеристики АС3350-08

Технічні характеристики АС3350-08	Внутрішня вісь (крену)	Середня вісь (тангажу)	Зовнішня вісь (рискання)
Кутова швидкість, град/сек	±300	±300	±100
Роздільна здатність кутової швидкості, град/сек	0.00001	0.00001	0.00001
Стабільність кутової швидкості - більше 360°	0.001	0.001	0.001
Прискорення, град/сек^2	2500	350	150
Момент, Нм	150	250	750
Моменти інерції, кг/м^2	3	35	250

Серія CARCO 824DS

Серія CARCO 824DS розроблена для випробувань , калібрування та дослідів акселерометрів та різних інерційних приладів.

Рис. 6. Центрифуга CARCO 824DS

Габарити :

висота загальна, мм -1270;

ширина, мм - 2160;

радіус стола, мм - 762

Виріб, що випробується:

вага номінальна, кг - 9 ;

габарити, мм - 508 х 254 х 254;

струмовідводи : 30 ліній по 2А кожна

Таблиця 6. Технічні характеристики CARCO 824 DS

Технічні характеристики	CARCO 824 DS	
	DS-1	DS-2
Роздільна здатність (дискретність) кута повороту, град	0.00001	0.00001
Кутова швидкість, град/сек	1300	1700
Роздільна здатність кутової швидкості, град/сек	0.00001	0.00001
Стабільність кутової швидкості,% - більше 360°	0.0001	0.0001
Час досягнення 40 g , сек	35	13
Діапазон зміни температури, град С	-55+85	-55+85
Стабільність, град С	±1,0	±1,0

Рис. 7. Вібростенд серії ВЭДС

Таблиця 7. Технічні характеристики вібростендів

Найменування	ВЭДС-100	ВЭДС-200	ВЭДС-400	ВЭДС-1500
Максимальна збуджуюча сила, Н	1000	2000	4000	15000
Робочий діапазон частот, Гц номінальний	20-2500	20-2500	20-1500	20-1500
розширений	5-5000	5-5000	5-3500	5-2500
Максимальне прискорення, м/с^2	400			430
Максимальне переміщення стола, мм	±4,5			±6
Максимальна маса навантаження, кг	22	45	0	300
Споживана потужність від мережі (380/220 В, 50 Гц), кВт	5	7	9	40

Рис. 8. Віброфуга V67-4H

Габарити :

висота загальна, мм -4016;

радіус обертання, мм - 4650

Виріб, що випробується:

вага (max) , кг - 120 ;

габарити, мм - 600 х 600 х 600;

струмопідводи : 10 ліній по 1А

Технічні характеристики:

- рівень прискорення (номінальний радіус) — від 5 до 80 g;

номінальний радіус -4 м;

кутова швидкість - 30-135 об/хв.;

час готовності - 8 хв.;

сила вібратора - 50000 Н;

діапазон частот - 10-2000 Гц

Рис. 9. Камера EXCAL класу холод/тепло.

Таблиця 8. Технічні характеристики камер EXCAL

Об’єм, літри	Розміри, мм	Мін. температура, °С	Макс, температура, °С
140	550x500x500	від -40 до -70	+180
220	550x500x750	від -30 до -80	+180
540	750 x 800 x 900	від -30 до -85	+180
770	900 x 950 x 900	від -40 до -90	+180
1400	1150x1100x1100	від -35 до -85	+180

Рис. 10. Камера вологості EXCAL класу холод/тепло/вологість модельного ряду ТМН

Камери EXCAL містять персональний комп’ютер промислового виконання з

15” сенсорним екраном, USB портом, інтерфейсом RS232 та програмним забезпеченням Spiral 3, яке працює з ОС Windows.

Камера вологості:

температурний вплив: від - 60°С до + 150°С;

вплив відносної вологості навколишнього середовища: до від 30 до 90%.

Камери солоного туману, температурного й барометричного тиску, впливу пилом:

- вплив солоного туману при відносній вологості 90 - 95% та концентрації 2-3 г/куб.м;
- барометричний вплив від 100 кПа до 1 кПа;
- температурний вплив: від - 60°С до + 120°С;
- вплив пилу при температурі від + 20°С до + 55°С;
- і швидкості потоку від 1 м/сек до 16 м/сек.

Рис. 11. Камери солоного туману, температурного й барометрического тиску, впливу пилом

Додаток 3. Акт про використання результатів дисертаційної роботи

ЗАТВЕРДЖУЮ

Перший заступник генерального директора-

Головний інженер

_____Білий В.М.

«14» квітня 2021р

АКТ

Про практичне впровадження результатів дисертаційного дослідження Долгорукова Сергія Олеговича на тему: «Автоматизоване проектування випробувального стенду навігаційного обладнання», поданого на здобуття наукового ступеня кандидата наук за спеціальністю

05.13.12 – системи автоматизації проектувальних робіт

1. Комісія у складі:

Голова: Білий В.М.- перший заступник генерального директора-головний інженер

Члени комісії: Подольський В.П.- начальник конструкторсько-технологічного відділу,

Махно Ю.П. – головний технолог.

розглянула матеріали впровадження результатів зазначеної дисертаційної роботи у процес виробництва.

2. Комісія прийшла до наступного висновку:

Результати наукового дослідження «Автоматизоване проектування випробувального стенду навігаційного обладнання» Долгорукова С.О. впроваджуються у практичну діяльність ДП ВО «Київприлад». Впровадження програмного середовища автоматизованого проектування, яке використовує агентів-проектувальників з елементами штучного інтелекту для вирішення завдання багатокритеріального прийняття рішень з використанням результатів проектування попередніх виробів дозволяє скоротити час проектування нових моделей пристроїв та обладнання.

Голова комісії

_____ В.М. Білий

Члени комісії:

_____ В.П. Подольський

_____ Ю.П. Махно

Додаток 4. СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Статті у виданнях іноземних держав–членів ЄС:

1. S. O. Dolhorukov, “Navigation equipment test table computer-aided design artificial intelligence configuration optimization,” *Věda a perspektivy*, vol. 1, no. 1, pp. 214-227, May, 2021.

Статті у наукових фахових виданнях України:

2. V. M. Sineglazov and S. O. Dolgorukov, “Optimal choice of the technical means of rate, pitch and roll channels subsystems of navigation equipment simulation table,” *Electronics and control systems*, no. 2, pp. 33-39, 2014.
3. V. M. Sineglazov and S. O. Dolgorukov, “Identification of mechanical backlash of navigation equipment testing table,” *Electronics and control systems*, no. 3, pp. 104-108, 2014.
4. V. M. Sineglazov and S. O. Dolgorukov, “Complex criteria for navigation equipment test table optimal design problem,” *Electronics and control systems*, no. 1, pp. 87-92, 2015.
5. V. M. Sineglazov and S. O. Dolgorukov, “UAV equipment test table for preflight testing,” *Electronics and control systems*, no. 3, pp. 126-132, 2015.
6. V. M. Sineglazov and S. O. Dolgorukov, “Environmental Chamber for Navigation Equipment Test Table,” *Electronics and control systems*, no. 2, pp. 62-66, 2017.

Матеріали та тези наукових доповідей:

7. V. M. Sineglazov and S. O. Dolgorukov, “Computer-aided Design of Navigation Equipment Test Table based on Game Optimization,” *in Proc. IEEE 6th Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, (Kyiv, Ukraine), 2020, pp. 7-10, doi: 10.1109/MSNMC50359.2020.9255419.

8. V. M. Sineglazov and S. O. Dolgorukov, “Computer-Aided Design System of Navigation Equipment Test Table,” *in Proc. IEEE 5th Int. Conf. Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, (Kyiv, Ukraine), 2019, pp. 7-11, doi: 10.1109/APUAVD47061.2019.8943871.

9. V. M. Sineglazov and S. O. Dolgorukov, “Intelligent Integrated Cads of Navigation Equipment Testing System,” *in Proc. IEEE 5th Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, (Kyiv, Ukraine), 2018, pp. 7-10, doi: 10.1109/MSNMC.2018.8576314.

10. V. M. Sineglazov and S. O. Dolgorukov, “Dynamic testing of UAV navigation system,” *in Proc. IEEE 4th Int. Conf. Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, (Kyiv, Ukraine), 2017, pp. 211-214, doi: 10.1109/APUAVD.2017.8308812.

11. V. M. Sineglazov and S. O. Dolgorukov, “Multi objective optimization for UAV navigation equipment test table design,” *in Proc. IEEE 4th Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, (Kyiv, Ukraine), 2016, pp. 117-120, doi: 10.1109/MSNMC.2016.7783121.

12. V. M. Sineglazov and S. O. Dolgorukov, “Dynamic hardware-in-the-loop UAV ground testing system,” *in Proc. IEEE Int. Conf. Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, (Kyiv, Ukraine), 2015, pp. 91-94, doi: 10.1109/APUAVD.2015.7346569.

13. V. M. Sineglazov and S. O. Dolgorukov, “Test bench of UAV navigation equipment,” *in Proc. IEEE 3rd Int. Conf. on Methods and Systems of Navigation and Motion Control (MSNMC)*, Kyiv, Ukraine, 2014, pp. 108-111, doi: 10.1109/MSNMC.2014.6979743.

14. V. M. Sineglazov and S. O. Dolgorukov, “HardWare-In-The-Loop simulation table for UAV navigation complexes testing,” *in Proc. IEEE 2nd Int. Conf. Actual Problems of Unmanned Air Vehicles Developments Proceedings (APUAVD)*, (Kyiv, Ukraine), 2013, pp. 141-145, doi: 10.1109/APUAVD.2013.6705307.