

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ**

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Система інтелектуалізації управління процесом 3D-друку на
основі застосування бази знань»

на здобуття освітнього ступеня магістра
зі спеціальності 124 Системний аналіз
освітньо-професійної програми «Інтелектуальні системи управління»

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

_____ Юрій БІЛИК
(підпис)

Виконав: здобувач вищої освіти групи САДМ-61

_____ Юрій БІЛИК

Керівник: _____ Ігор ПАТРАКЕЄВ
к.т.н., доцент

Рецензент: _____ Ім'я, ПРИЗВИЩЕ
*науковий ступінь,
вчене звання*

Київ 2024

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут інформаційних технологій**

Кафедра інформаційних систем та мереж
Ступінь вищої освіти магістр
Спеціальність 124 Системний аналіз
Освітньо-професійна програма Інтелектуальні системи управління

ЗАТВЕРДЖУЮ
Завідувач кафедру ІСТ
Каміла СТОРЧАК
«___» _____ 20__ р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Білик Юрій Мирославович
(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: Система інтелектуалізації управління процесом 3D-друку на основі застосування бази знань

керівник кваліфікаційної роботи: Ігор ПАТРАКЕСВ к.т.н..Доцент,
(Ім'я, ПРИЗВИЩЕ, науковий ступінь, вчене звання)

затверджені наказом Державного університету інформаційно-комунікаційних технологій
від «___» _____ 20__ р. № ____

2. Строк подання кваліфікаційної роботи «___» _____ 20__ р.

3. Вихідні дані до кваліфікаційної роботи: Методи створення продукційних баз знань, програмне забезпечення керування 3-Д друком, відомості про властивості матеріалів, програмне забезпечення для створення команд принтера, сервіси API

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Актуальність системи інтелектуалізації для 3D-друку
2. Загальна характеристика процесу 3D-друку
3. Порівняння різних підходів створення баз знань
4. Розробка структури бази знань
5. Розробка програмного забезпечення для впровадження системи інтелектуалізації

5. Перелік ілюстративного матеріалу: *презентація*

6. Дата видачі завдання «___» _____ 20__ р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Отримання завдання	01.05.2024	
2	Аналіз існуючих рішень та літературних джерел	04.05.2024 – 03.06.2024	
3	Розроблення плану роботи	01.09.2024 – 15.09.2024	
4	Переддипломна практика	30.09.2024 – 23.10.2024	
5	Розроблення першого та другого розділу	01.09.2024 – 28.09.2024	
6	Розроблення третього розділу	01.10.2024 – 12.12.2024	
7	Подання кваліфікаційної роботи на попередній захист	20.12.2024	
8	Подання кваліфікаційної роботи на захист	10.01.2025	

Здобувач вищої освіти

(підпис)

Юрій БЛИК
(Ім'я, ПРІЗВИЩЕ)

Керівник
кваліфікаційної роботи

(підпис)

Ігор ПАТРАКЕСВ
(Ім'я, ПРІЗВИЩЕ)

**ПОДАННЯ
ГОЛОВІ ЕКЗАМЕНАЦІЙНОЇ КОМІСІЇ
ЩОДО ЗАХИСТУ КВАЛІФІКАЦІЙНОЇ РОБОТИ
на здобуття освітнього ступеня магістра**

Каміла СТОРЧАК
(Ім'я, ПРІЗВИЩЕ)

ВІДГУК РЕЦЕНЗЕНТА
на кваліфікаційну роботу
на здобуття освітнього ступеня магістра

здобувача(ки) вищої освіти Білик Юрій Мирославович
(прізвище, ім'я, по батькові)

на тему «Система інтелектуалізації управління процесом 3D-друку на основі застосування бази знань»

Актуальність.

Актуальність теми обґрунтовано, вона полягає у необхідності створення інтелектуальних систем, які здатні адаптуватися до різноманітних умов виробництва та забезпечувати зручність і ефективність використання технологій 3D-друку навіть для користувачів з мінімальним досвідом у цій галузі. Таким чином, робота над розробкою системи інтелектуалізації управління процесом 3D-друку є важливим і перспективним напрямом, що відповідає сучасним тенденціям технологічного розвитку. Як зазначено в роботі, такий підхід значно спрощує сам процес промислового виробництва, та зменшує людський фактор, що однозначно корисно впливає на якість кінцевого результату, та його одноманітність.

Позитивні сторони.

1. У роботі розглянуто різні підходи до створення подібних систем
2. Розглянуто та застосовано різні методи створення моделей
3. Практична частина виконана з допомогою сучасних рішень

Недоліки.

1. Є доцільним заглиблене вивчення економічної доцільності
2. Вбачається необхідність у створення більш зручного інтерфейсу вводу даних і правил, якщо озиратися на їх значну кількість

Відзначені зауваження не впливають на загальну позитивну оцінку кваліфікаційної роботи магістерської

Висновок: *кваліфікаційна робота на здобуття ступеня магістра заслуговує оцінку " _____ ", а здобувач(ка) Юрій БІЛИК заслуговує присвоєння кваліфікації магістр з системного аналізу*

Рецензент:

науковий ступінь, вчене звання

підпис

Ім'я, ПРІЗВИЩЕ

РЕФЕРАТ

Текстова частина магістерської роботи 73с., 35рис., 4 формул, 47 джерел

Актуальність теми.

Розвиток технологій **адитивного виробництва** та **3D-друку** відіграє ключову роль у сучасній промисловості, медицині, будівництві та інших сферах. Завдяки своїй гнучкості, швидкості виробництва та можливості створювати складні геометричні форми, 3D-друк стає невід'ємною частиною виробничих процесів. Однак зростання вимог до якості продукції, оптимізації витрат на матеріали та підвищення ефективності процесів друку ставить перед інженерами та дослідниками нові виклики. Однією з **найактуальніших проблем** у 3D-друці, особливо при використанні технології **FDM** є наприклад **необхідність підтримок** для успішного друку моделей з ускладненою геометрією. Підтримки збільшують час друку, витрати на матеріали та потребують додаткової постобробки, що може негативно впливати на якість поверхні готової деталі. Крім того, неправильна орієнтація моделі на столі 3D-принтера призводить до зниження міцнісних якостей виробу через надмірну анізотропію матеріалу. Додаткової актуальності темі надає застосування **сучасних алгоритмічних підходів** та методів оптимізації, таких **метод "бази знань"**. Це дозволяє автоматизувати процес вибору комбінації різних параметрів моделі, знижуючи необхідність ручного втручання та підвищуючи ефективність роботи інженерів.

Об'єктом дослідження є процес оптимізації параметрів друку 3D-моделей на платформі 3D-принтера

Ціль роботи дослідження методів автоматизації процесу оптимізації параметрів 3D-моделей для 3D-друку, з використанням баз знань

Методи дослідження В даній роботі використовувався метод порівняльного аналізу результатів відомих досліджень, та аналіз отриманих результа

Результати дослідження та їх новизна полягає інтеграції засобів інтелектуалізації виробництва на основі баз знань в процес 3D-друку

Ключові слова : 3D-ДРУК, МОДЕЛІ БАЗ ЗНАНЬ, ПРОДУКЦІЙНІ МОДЕЛІ БАЗ ЗНАНЬ, ПАРАМЕТРИ, ІНТЕЛЕКТУАЛІЗАЦІЯ, *АДИТИВНЕ ВИРОБНИЦТВО, ДЕРЕВО РЫШЕНЬ*

ABSTRACT

The development of additive manufacturing technologies and 3D printing plays a key role in modern industry, medicine, construction, and other fields. Due to its flexibility, fast production speed, and the ability to create complex geometric shapes, 3D printing has become an integral part of manufacturing processes. However, the growing demands for product quality, material cost optimization, and improving printing process efficiency present new challenges for engineers and researchers. One of the most pressing issues in 3D printing, especially when using FDM technology, is, for example, the need for supports to successfully print models with complex geometries. Supports increase printing time, material costs, and require additional post-processing, which can negatively impact the surface quality of the finished part. Additionally, incorrect orientation of the model on the 3D printer bed leads to reduced mechanical properties of the product due to excessive anisotropy of the material. The relevance of the topic is further enhanced by the application of modern algorithmic approaches and optimization methods, such as the "knowledge base" method. This allows for automating the process of selecting combinations of different model parameters, reducing the need for manual intervention, and improving the efficiency of engineers' work.

The object of the study is the process of optimizing the printing parameters of 3D models on a 3D printer platform.

The goal of the work is to investigate methods of automating the process of optimizing 3D model parameters for 3D printing, using knowledge bases.

Research methods: This work utilized the method of comparative analysis of results from known studies and analysis of the obtained results.

The results of the research and their novelty lie in the integration of intelligent manufacturing tools based on knowledge bases into the 3D printing process.

Keywords: 3D PRINTING, KNOWLEDGE BASE MODELS, PRODUCTION KNOWLEDGE BASE MODELS, PARAMETERS, INTELLIGENTIZATION, ADDITIVE MANUFACTURING, DECISION TREE.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	11
ВСТУП.....	12
РОЗДІЛ 1 АНАЛІТИЧНИЙ ОГЛЯД ТЕХНОЛОГІЙ АДИТИВНОГО ВИРОБНИЦТВА ТА МЕТОДІВ ОПТИМІЗАЦІХ	13
1.1 Принцип друку методом спікання і плавлення матеріалів	13
1.2 Програмне забезпечення що задіяне в процесі 3D друку	17
1.3 Витратні матеріали адитивного друку	20
1.4 Механічні елементи принтера що впливають на підбір параметрів друку	27
1.5 Аналіз критеріїв оцінювання якості поліграфічної продукції.....	30
1.6 Огляд інформаційного середовища в сучасних системах 3D-друку	32
Розділ 2. АНАЛІЗ МЕТОДІВ ІНТЕЛЕКТУАЛІЗАЦІЇ УПРАВЛІННЯ.....	37
2.1 Аналіз принципів інтелектуалізації управління	37
2.2 Моделі представлення даних	39
2.3 Дослідження аналізу параметрів процесу друку	47
2.4 порівняння середовищ розробки МПЗ.....	51
2.5. Проектування продукційної моделі бази знань прямого виводу.....	59
РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО ЗАСТОСУНКУ	66
3.1 Обґрунтування вибору засобів керування принтера	66
3.2 Обґрунтування вибору інструменту розробки МПЗ	67
3.3 Програма визначення параметрів друку на основі продуктивної моделі бази знань	71
3.4 API для подання моделі з іншими програмними засобами	77
ПЕРЕЛІК ПОСИЛАНЬ	83
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ(ПРЕЗЕНТАЦІЯ)	89

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

МПЗ – Модель предствалення даних

FDM – fused deposition modeling

SLS – Selective laser sintering

БЗ – База даних

БД – База даних

CLIPS – Language Integrated Production System

KRL – knowledge representation language

ПЗ – Прграмне забезпечення

ВСТУП

3D-друк, або адитивне виробництво, став революційною технологією, що змінила підходи до виробництва у багатьох галузях, включаючи промисловість, медицину, архітектуру та багато інших. Завдяки своїй здатності створювати складні геометричні форми з мінімальними витратами матеріалу, 3D-друк відкриває нові можливості для розробки індивідуалізованих і високоточних виробів. Проте, процес 3D-друку часто супроводжується низкою проблем, таких як необхідність оптимізації параметрів друку для досягнення високої якості виробів, економії матеріалів і часу, а також зниження впливу людського фактору.

Одним з основних аспектів, що впливають на результат друку, є правильний вибір параметрів, таких як температура екструдера, температура столу, швидкість друку та товщина шару. Для досягнення оптимальних результатів необхідно враховувати безліч факторів, що змінюються в залежності від типу матеріалу, геометрії моделі та інших умов.

У цьому контексті використання методів автоматизації, зокрема моделей баз знань, дозволяє значно спростити процес налаштування параметрів друку. Такі системи можуть адаптувати параметри в реальному часі, враховуючи дані з датчиків принтера та параметри, надані слайсером. Це підвищує ефективність виробничих процесів, знижує ймовірність помилок та покращує загальну якість друку.

РОЗДІЛ 1 АНАЛІТИЧНИЙ ОГЛЯД ТЕХНОЛОГІЙ АДИТИВНОГО ВИРОБНИЦТВА ТА МЕТОДІВ ОПТИМІЗАЦІХ

1.1 Принцип друку методом спікання і плавлення матеріалів

Fused Deposition Modeling (FDM) або в перекладі – моделювання методом наплавлення – являється найбільш розповсюдженою технологією 3D-друку. FDM технологія у поліграфії набула великого розповсюдження з декількох причин. Перш за все, це одна з найперших запатентованих технологій пов'язаних з адитивним багатошаровим виробництвом і 3D друком в цілому. Також за принципом друку вона є найпростішою, тому не вимагає великих капіталовкладень

при впровадженні на поліграфічних підприємствах. Перевагою є і те, що на сьогодні існують готові комерційні рішення для практично в будь-якій цінovій категорії. Додрукарські операції, наприклад макетування поліграфічного виробу, не вимагають від спеціалістів видавничо-поліграфічної галузі особливих знань, так як програми 3D-моделювання працюють аналогічно до програмного забезпечення

щодо роботи із графічними файлами. Широке застосування у оперативній поліграфії також викликано і тим, що на ринку витратних матеріалів представлено широкий вибір марок пластиків для 3D-друку, що дозволяє обрати матеріал, який найкраще підходить вимогам до готової моделі.

Принцип друку методом спікання і плавлення матеріалів (FDM) подано на Рис. 1.1.

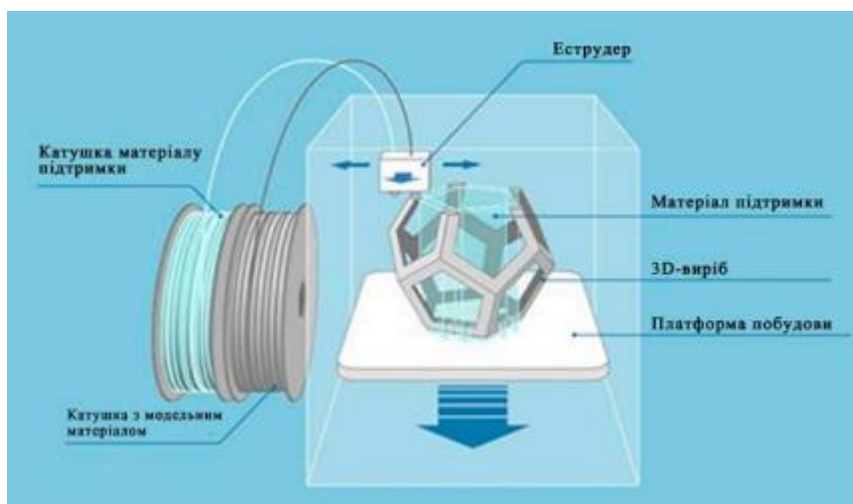


Рисунок 1.1 — Принцип друку методом спікання і плавлення матеріалів (FDM) [1]

Технологію друку поліграфічного виробу за методом спікання можна описати наступним чином: витратний матеріал подається з катушок в екструдер, де розігрівається до температури плавлення, далі за допомогою сопла з точним отвором малого діаметру пластик екструдується на платформу-столик при цьому сопло робить переміщення по площині виконуючи команди керуючого ПЗ,

після закінчення нанесення першого шару, екструдер піднімається вгору на величину товщини шару і процес повторюється [2].

Зазначимо, що FDM технології дозволяють друкувати практично будь-яким пластиком. Наприклад, на початку становлення цієї технології ентузіасти використовували пластик для мото-косарок, адже на той момент це був практично єдиний матеріал, що можна було придбати у вигляді гнучкого прутка з підходящим для використання у 3D-принтері діаметром. Зважаючи на вищесказане, варто узагальнити переваги та вказати на недоліки FDM технології при виготовленні поліграфічних виробів (Рис 1.2):

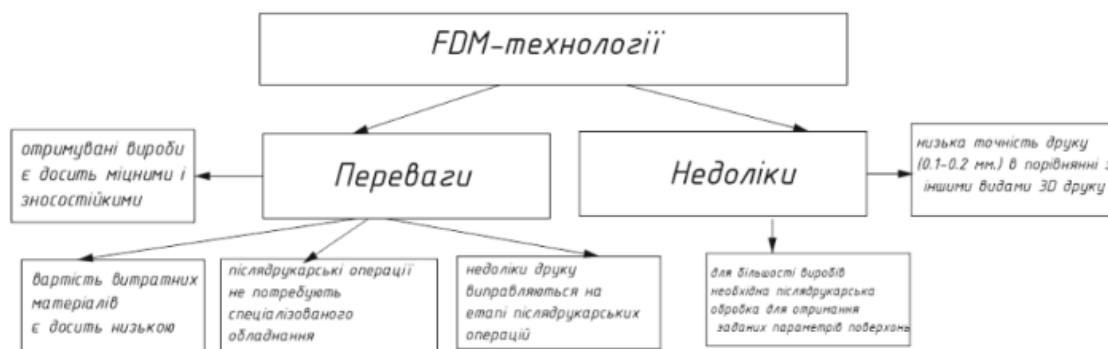


Рисунок 1.2 — Узагальнена схема переваг та недоліків FDM технології [54]

За технологією SLA друкарські операції здійснює за схемою, що подано Рис. 1.3.

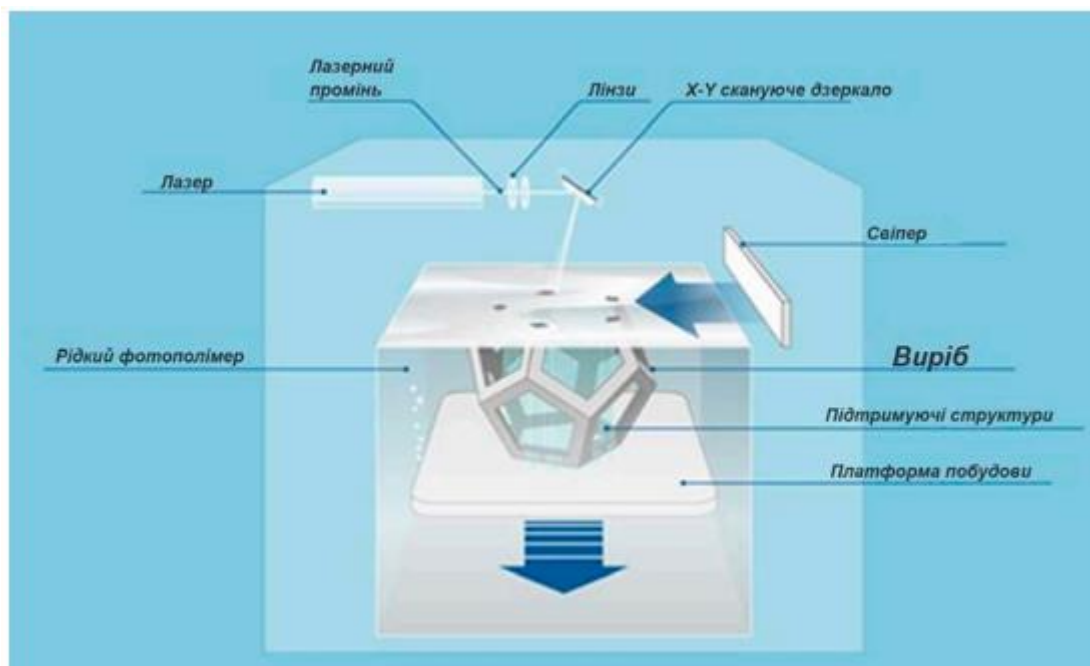


Рисунок 1.3 — Принцип виготовлення деталей за технологією SLA [10]

Суть операцій полягає в наступному: у ванну з рідким полімером вміщується платформа і занурюється на певну глибину, таким чином, щоб відстань від поверхні цієї платформи до поверхні рідкого полімеру становила задану товщину шару матеріалу. Далі лазер за допомогою системи наведення, що керується спеціалізованим ПЗ, взаємодіє із полімером викликаючи тим самим його

затвердіння в заданих точках поверхні. Після цього платформа зміщується на відстань одного шару вниз, свіпер розрівнює поверхню фото полімеру, прибираючи при цьому зайвий шлак, процес повторюється до тих пір поки модель не буде повністю завершена. Основні переваги та недоліки вищезазначеного методу наведено на Рис 1.4.



Рисунок 1.4 Переваги та недліки технології SLA

1.2 Програмне забезпечення що задіяне в процесі 3D друку

Програмне забезпечення макетування 3D-виробів із застосуванням адитивних технологій. Зважаючи на тенденції розвитку комп'ютерних графічних технологій, розуміння процесу створення тривимірного контенту в спеціалізованих системах є

важливим компонентом повноцінного професійного надання поліграфічних послуг. Програмне забезпечення (ПЗ), що використовується для 3D- друку можна поділити на чотири види (1.5)



1.5 – Класифікація програмного забезпечення для 3D- друку [3]

До першого виду програмного забезпечення щодо виготовлення або проектування 3D моделей, відносять конструкторські програми або програмні модулі, де проєктований об'єкт може створюється або з «нуля», або на сканері. Для отримання об'єкту скануванням обладнання має зовнішні датчики, що допомагають отримати 3D модель вже існуючого об'єкта.

Аналізуючи сучасне спеціалізоване програмне забезпечення, відзначимо

достатньо велику кількість програм для створення і редагування 3D графіки. Серед найбільш поширених пакетів для роботи з тривимірними моделями необхідно назвати продукти компанії AUTODESK (3DS MAX, Maya, Mudbox, AutoCAD). Схожі за своєю суттю, проте відмінні за структурою інтерфейсу та можливостями є програми CINEMA 4D, MODO, LightWave.

Програма Blender 3D є потужним інструментом, який дозволяє поліграфічним

підприємствам здійснювати якісне моделювання тривимірних комп'ютерних моделей у вигляді 3D-виробу із розкриттям текстуровання та анімуванням. Таким чином на додрукарському етапі по роботі із 3D-об'єктами втрати якості майбутніх виробів максимально виключено. Ще одним видом програмного забезпечення, що забезпечує етап додрукарських операцій є програми-слайсери. Саме завдяки цьому ПЗ відбувається перетворення високо дискретного, полігонального 3D-виробу на набір 2D шарів, у вигляді команд G-code, зрозумілих для 3D принтера. Порівняльна характеристика вищезазначеного програмного забезпечення за застосованою операційною системою, системними вимогами, кваліфікацією користувача, доступністю користування подано у таблицях 1.1- 1.2.

Назва	ОС	Вимоги кваліфікацій	Доступність
Cura	Windows, Mac, Linux	потребує	Безкоштовно
Slic3r	Windows, Mac, Linux	потребує	Безкоштовно
KISSkicer	Windows, Mac, Linux	потребує	Безкоштовно
XYZmaker	Windows, Mac, Linux	Не потребує	Безкоштовно

Таблиця 1.1 — Порівняльна характеристика 3D-слайсерів [4]

Назва	процесор	Роздільна здатність дисплея	Об'єм диску MB	Об'єм оперативної пам'яті	Гравфічна карта сумісна з OpenGL 4.1
Cura	Intel core i3	1920x1080	600	8	+
Slic3r	Intel core i3	1480x720	700	4	+
Orca	Intel core i5	1920x1080	680	8	+
XYZmaker	Intel core i3	1280x720	600	1	+

Таблиця 1.2 — Порівняльна характеристика 3D-слайсерів за системними вимогами [4]

Отже, програмне забезпечення XYZmaker Suite є потужним інструментарієм при роботі на поліграфічних підприємствах-початківців, що мають на меті здійснюють якісне виготовлення тривимірних комп'ютерних моделей у вигляді виробів. Таким чином на додрукарському етапі для роботи по поділу на відповідну кількість шарів за допомогою 3D-слайсеру обрано ПЗ – XYZmaker Suite (рис.1.6).

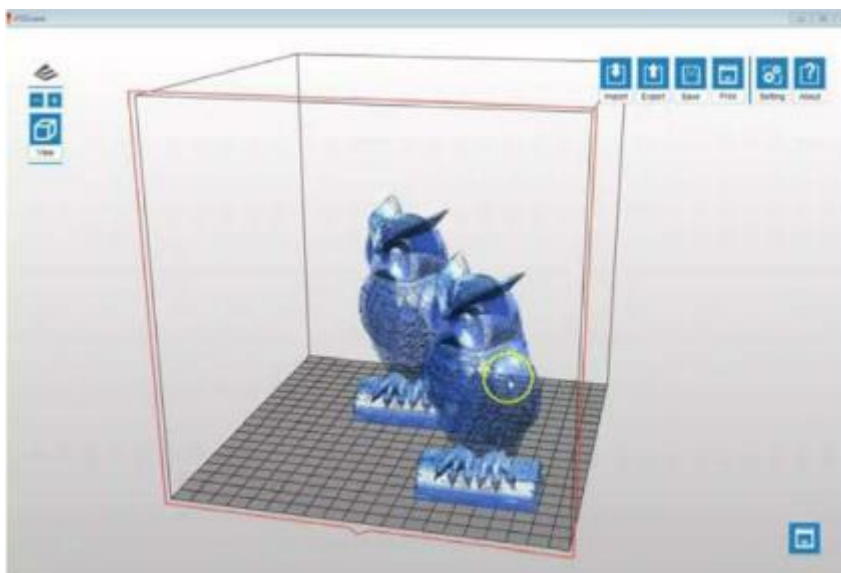


Рисунок 1.6 — Робоча зона 3D-слайсеру «XYZmaker Suite»

До третього виду ПЗ відноситься безпосередньо набір програмних компонентів, що керують друком: задають параметри температури, передають завчасно згенерований G-code на 3D-принтер, проводить калібровку принтера, тощо. При здійсненні друкарських операцій дослідних зразків 3D-виробів було використано ПЗ XYZmaker Suite, що містить інструменти, необхідні для встановлення температури (рис.1.8), передачі згенерованого G-code та проведення калібрування принтера.

Останнім, четвертим видом є набір ПЗ для мікроконтроллера, що безпосередньо отримує команди від слайсера і керує апаратними засобами принтера.

Перетворюючи G-code в сигнали, що посилаються на двигуни, екструдери, та приводи подачі матеріалу. Додатковою задачею такого ПЗ є виставлення екструдера в «нулі» і його калібровка відносно столу [5].

Програмне забезпечення XYZmaker Suite включає до свого складу і четвертий вид комплексу програм, що керують механізмами принтера.

1.3 Витратні матеріали адитивного друку

Витратні матеріали для здійснення адитивного друку прийнято класифікувати

за трьома категоріями (Рис.1.7).

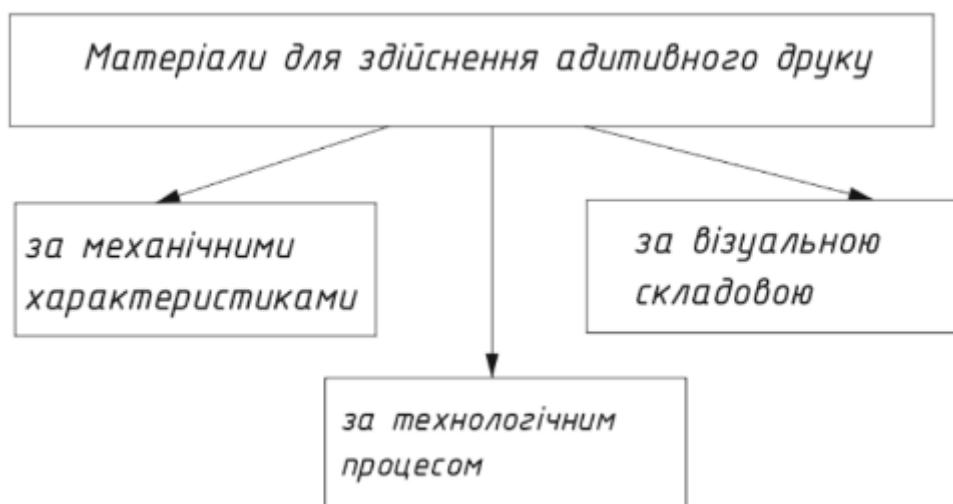


Рисунок 1.7 – Класифікація матеріалів адитивного друку [6]

Технологія FDM 3D-друку є найбільше розповсюдженою, тому вибір матеріалів саме для таких 3D-принтерів найширший. У світі сформувалась єдина класифікація матеріалів, якої дотримується більшість виробників. Залежно від спеціалізації друку, витратні матеріали для FDM друку поділяються на групи, кожна з яких призначена для визначеного типу виробів яким можуть знадобитися різні властивості матеріалу виготовлення (рис 1.8).

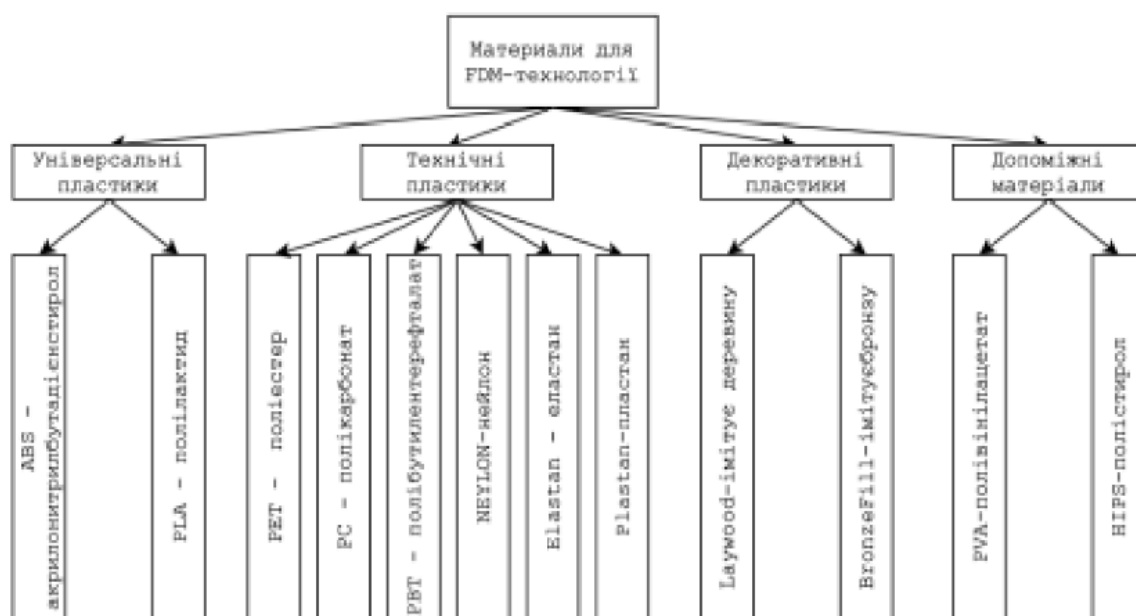


Рисунок 1.8– Схема класифікації витратних матеріалів FDM-технології [7]

Витратні матеріали технології друку FDM

Універсальні пластики загального призначення використовуються для створення виробів, що не зазнають значних механічних, термічних і інших видів навантажень.

Друк продукції буде здійснюватись саме витратними матеріалами для FDM-друку, тому проведемо порівняння властивостей матеріалів FDM технології, для подальшого вибору потрібного полімеру, від якого залежатиме якість отримання виробу [8].

Для зрозумілішої картини властивостей полімерів додамо деякі ключові критерії, від яких залежить вибір матеріалу (без урахування вартості та швидкості): – Параметр «Складність друку» має на увазі те, наскільки легко друкувати матеріал – максимальна швидкість друку, частота невдалих друкув, точність,

легкість подачі в принтер. Візуальна якість: наскільки добре виглядає готовий виріб. Параметр «Межа міцності» вказує на максимальне прикладання зусилля, яке виріб може витримати перед тим, як зламатися.

– Параметр «Міцність на розрив» вказує на максимальну довжина, на яку виріб розтягується перед розривом.

– Параметр «Опір удару» вказує на енергію необхідна для того, щоб зламати виріб різким ударом.

– Параметр «Адгезія шару» вказує наскільки висока адгезія між шарами матеріалу. Це з ізоотропією: що краще адгезія шару, то більш ізоотропним і цілістним в кінцевому результаті будуть якості що матиме виріб.

–Параметр «Вогневитривалість» вказує на максимальну температура, яку виріб витримує перед розм'якшенням та деформуванням.

PLA – поліактид – найпростіший полімер для друку, забезпечує гарну візуальну якість. Дуже жорсткий, досить міцний, але дуже крихкий.

Пластики типу ABS застосовуються замість PLA, у випадках, коли 3D-

виробам потрібно забезпечити стійкість до високої температури, а також більшу твердість. PET – поліетилентерефталат – трохи м'якіший полімер, який широко поширений і має цікаві додаткові властивості з кількома недоліками. Nylon – має високі механічні властивості і кращу ударостійкість. Однак адгезія шарів проблематична.

TPU — головним чином використовують для гнучких виробів, але його дуже високий опір удару може знайти незвичайне застосування.

PC – найміцніший матеріал з усіх, може бути цікавою альтернативою до ABS, оскільки їх властивості досить схожі.

Матеріал	Переваги	Недоліки
PLA	Біорозкладний	Низька вологостійкість
	Без запаху	Важко склеюється
	Можна обробляти шліфувальним папером та фарбувати акрилом	
	Стійкість до ультрафіолету	
ABS	Можлива пост обробка парами ацетону для надання глянцю	Чутливість до ультрафіолету
	Можна обробляти шліфувальним папером та фарбувати акрилом.	Запах під час друку
	Можна використовувати ацетон як клей	Потенційно шкідливі виділення
	Хороша зносостійкість	
PET	Можна використовувати з харчовими продуктами	Важче ніж PLA та ABS
	Висока вологостійкість	
	Висока хімічна стійкість	
	Піддається переробці	
Nylon	Хороша хімічна стійкість	Поглинає вологу
	Висока міцність	Потенційно шкідливі виділення
TPU	Хороша зносостійкість	Складність пост обробки
	Хороша стійкість до олії та мастила	Важко склеюється
PC	Піддається стерилізації	Чутливість до ультрафіолету
	Легкість пост обробки (шліфування)	

Рисунок 1.9 Порівняльна характеристика витратних матеріалів для FDM-друку [9]

Для узагальненого порівняння властивостей витратних матеріалів для FDM-друку створено графік, який представлено на Рис 1.10, де кожен матеріал був оцінений за вищезазначеними критеріями (Сис.1.10) за шкалою від 1 (низький) до 5 (високий) [1].



Рисунок 1.10– Розподіл оцінки матеріалів за критеріями [14]

Вибір правильного виду полімеру має вирішальне значення для отримання затребуваних замовником властивостей 3D-виробу, особливо, якщо виріб матимуть довготривале функціональне застосування.

Матеріали для SLA-друку мають найширший вибір [12].

SLA-матеріали виготовлені на основі термореактивних смол, різних за складом, кожна з яких розроблена з урахуванням характеристик звичайних пластиків. Найвища температурою експлуатації 238 °C.

На рисунку 1.11 зображено узагальнене схема класифікації термореактивних смол.



Рисунок 1.11 – Схема класифікації термореактивних смол [12]

Кожна класифікаційна назва використаної смоли у складі SLA-матеріалів відображає найбільш важливі характеристики матеріалу, його гнучкість, міцність,

візуальну якість, адгезивність шарів, вогневитривалість, тощо.

На принтерах SLA доступний друк з найширшим вибором біосумісних матеріалів для стоматології та медицини. На Рис 1.12 представлено іншу класифікацію витратних матеріалів, які

використовуються при технології SLA-друку, орієнтовану на подальшу сферу застосування виробів.



Рисунок 1.12 – Класифікація витратних матеріалів SLA-технології [4]

Найбільш широко застосованим матеріалом для технології друку SLS є поліамід (PA 12), ще відомий як нейлон 12, але він також може являти собою поліпропілен, алюмінідів, амід вуглецю, PEBA, PA 11, PEEK [13].

Ще викликано тим, що серед основних властивостей матеріалу визначено: механічну стійкість, біосумісність, гіпоалергенність, стійкість до ультрафіолетового впливу, жорсткість. Нейлон 12 підходить для SLS друку стандартних виробів, до характеристик яких не висувають суворих вимог. Нейлон 11 підходить для виробництва прототипів і деталей з підвищеним коефіцієнтом зносостійкості і пластичності. Вказані матеріали — однокомпонентні. SLS технологія 3D друку також передбачає використання порошків, які складаються з двох компонентів. Це може бути полімерна основа і додаток, який надає виробу більшої міцності, гнучкості та інших потрібних характеристик.

Також можна додавати до матеріалів інші добавки, такі як вуглецеві волокна,

скляні або алюмінієві волокна, тим самим покращуючи механічні характеристики деталей.

1.4 Механічні елементи принтера що впливають на підбір параметрів друку

Рама 3D-принтера є основною конструкцією, що забезпечує надійність та стійкість пристрою. Вона утримує всі ключові компоненти на своїх місцях і відповідає за стабільність руху під час друку.

Якісна рама зменшує вібрації та коливання, що можуть виникати під час роботи, забезпечуючи **точність нанесення шарів**. Якщо рама недостатньо жорстка або має дефекти, це може призвести до помилок у друці, таких як зміщення шарів або нерівні поверхні. Таким чином, надійна рама є основою для **прецизійного та якісного 3D-друку**. **Екструдер** – це ключовий компонент 3D-принтера, який відповідає за подачу та плавлення філаменту для формування друкованого об'єкта. Він складається з двох основних частин: **механізму подачі** (що подає нитку) та **гарячого кінця** (hotend), де відбувається нагрівання та екструзія матеріалу.[2]

Екструдер впливає на **швидкість друку та якість деталей**. Від його роботи залежить рівномірність подачі філаменту, що критично важливо для стабільного нанесення шарів. Якщо екструдер погано справляється з подачею або має засмічення, можуть виникати пропуски в шарах або нерівномірне плавлення матеріалу.

Існують **прямі екструдери** (знаходяться близько до сопла) і **боуден-екструдери** (розташовані окремо від гарячого кінця). Прямі забезпечують кращий контроль над подачею матеріалу[1], але додають ваги рухомих елементам. Боуден-екструдери дозволяють рухатися швидше через зниження навантаження на принтер, але можуть бути менш точними для гнучких філаментів.

Таким чином, **якісно налаштований екструдер** забезпечує рівномірну подачу матеріалу, що напряду впливає на **точність і надійність 3D-друку**.

Робочі осі принтера – це системи координат, по яких рухаються різні елементи 3D-принтера для створення об'єкта. Більшість принтерів працюють у трьох основних осях: X, Y та Z.

Ось X відповідає за горизонтальний рух (зазвичай вліво-вправо). По цій осі рухається або екструдер, або стіл, залежно від конструкції принтера.

Ось Y забезпечує рух вперед-назад, що дозволяє позиціонувати елемент друку в потрібному місці.

Ось Z відповідає за вертикальний рух (вгору-вниз), де відбувається зміна висоти шару в процесі друку.

Точність і стабільність руху по осях напряду впливають на якість друку. Будь-які люфти, неправильне налаштування або вібрації можуть призвести до зсуву шарів, неточності контурів і дефектів поверхні.

Охолодження у 3D-принтері відіграє важливу роль у якості друку, особливо при роботі з термопластичними матеріалами, такими як PLA або PETG. Система охолодження відповідає за стабільну температуру під час процесу, що впливає на формування шарів і загальну якість готового виробу.

Сопла в 3D-друку – це важлива частина 3D-принтера, яка відповідає за видавлювання розігрітого матеріалу (наприклад, пластику) під час друку. Сопло є кінцевим елементом **екструдера** (механізму подачі матеріалу) і виглядає як невеликий металевий наконечник з маленьким отвором.

Найбільш часто використовуваний розмір сопла – 0,4 мм. Це забезпечує хороший баланс технічних характеристик, оскільки діаметр сопла відіграє важливу роль у процесі 3D-друку. Великий діаметр сопла дозволить збільшити швидкість друку, а менший – забезпечить значно вищу точність,

але й час друку при цьому зростає. Швидкість також залежить від самої машини. У більшості роздрібних принтерів, таких як **Creality Ender 3 Pro** та **Ender 3 V2**, використовуються сопла діаметром **0,4 мм**. Однак можна замінити сопло на більший або менший розмір, щоб отримати іншу швидкість друку чи деталізацію. Важливо переконатися, що новий діаметр сумісний зі специфікаціями вашого принтера та може працювати з матеріалом нитки (філаментом), який ви плануєте використовувати.

Якщо ви хочете друкувати з висотою шару **0,6 мм або більше**, необхідно використовувати сопло діаметром **більшим за стандартний**, наприклад **0,6 мм**. Однак важливо пам'ятати, що максимальна ширина екструзії сопла не повинна перевищувати висоту шару. Під час вибору сопла для 3D-принтера важливо враховувати його **сумісність із матеріалом**, який ви плануєте використовувати. **Оптимальний розмір і тип сопла для 3D-принтера залежать від конкретних вимог проєкту друку**. Латунне сопло є найбільш популярним варіантом і зазвичай підходить для роботи з більшістю матеріалів. Деякі користувачі віддають перевагу соплам з **нержавіючої або загартованої сталі**, оскільки вони довговічніші та менш схильні до зношування. **Рубінові сопла** – це найдорожчий і найякісніший варіант, що забезпечує високу стійкість до зносу та підходить для друку екзотичними матеріалами, такими як **вуглеволокно** чи **металеві нитки**. При використанні таких спеціальних сопел слід враховувати, що вони не поглинають тепло так ефективно, як латунні. Тому температуру друку краще підвищити на **10–20 градусів за Цельсієм**.

Пам'ятайте, що результати використання сопел іншого розміру можуть відрізнятися залежно від характеристик принтера та матеріалу філаменту. Завжди рекомендується звертатися до **інструкції користувача** вашого принтера або до **вебсайту виробника** для отримання детальної інформації про **сумісність сопел** та рекомендації щодо висоти шару.

Якщо зосередити свою увагу лише на самому процесі створення продукту, відкинувши технічні моменти пов'язані з обслуговуванням принтера та його технічним станом, такі як натяг шківів, властивості моторів та інше. Ми побачимо що найвпливовішою частиною принтера є сопло через яке здійснюється видавлювання нагрітого матеріалу. Саме в ньому відбувається нагрів та придання форми лінії. Від інших елементів принтера потребується лише справна робота, в той час як властивості сопла можуть мати різні переваги та недоліки, а також вимоги до попередніх налаштувань. Подібне значення має тип екструдера, їх на відміну від сопла існує лише два види

1.5 Аналіз критеріїв оцінювання якості поліграфічної продукції

3D-друк зараз користується великою популярністю. 3D-друк можездійснюватися різними способами і з використанням різних матеріалів, але у будь-якому випадку в основі лежить принцип пошарового створення (вирощування) твердого об'єкту. Вимоги до якості 3D-друку з описаними стандартами (які, до речі, є на стадії розробки) зводяться в декілька технологічних груп (рис 1.13)



Рисунок 1.13 – Класифікація вимог до якості 3D-друку [14]

Ряд параметрів, що впливатимуть на якість отримання поліграфічного виробу

методом 3D-друку можна оцінити в абсолютних величинах (рис.1.14).

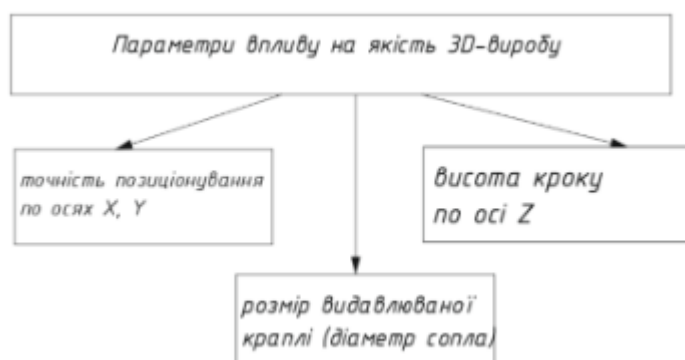


Рисунок 1.14 – Класифікація параметрів впливу на якість виробів [14]

Параметр точності позиціонування по осях X , Y вказує на те, на скільки вірно переміститься друкуюча голівка 3D-принтера відносно заданих комп'ютером координат. Точність позиціонування варіюється в межах близько 6-30 мікрон; Параметр висоти кроку по осі Z – це крок механічного приводу, який опускає платформу з моделлю. Орієнтовний показник – до 2,5 мікрона.

Проте це не означає, що висота шару при друці буде всього 2,5 мікрони. На висоту шару впливає цілий ряд фізичних обмежень, які не дозволяють створювати шари такого малого розміру. Слід враховувати також і розмір розплавленої в'язкої краплі матеріалу з екструдера.

Відповідно до чисельних експериментів FDM 3D-друку, оптимальний показник висоти шару складає – 200 мікрон. Саме така висота дозволяє краплі утворюватись досить великою, щоб склеїтися з попереднім шаром і надійно зафіксуватися на поверхні. Разом з тим цілком прийнятною є висота шару 100 мікрон, та це призводить до збільшення часу друку і роблять об'єкт більш крихким. Зазначимо, що описані в технічних характеристиках 100, 50 чи то 20 мікрон – це може бути не точність позиціонування, і не точність друку, а усього лише висота віддрукованого шару; – розмір краплі (діаметр сопла). Реальний розмір «точки» на створюваній 3D-моделей набагато більше залежить від діаметру отвору сопла екструдера, чим від висоти шару. Практика показує, що оптимальним параметром розміру сопла екструдера є показник у 400 - 500 мікрон. Звичайно, внаслідок зменшення розміру сопла, поліпшиться якість друку, але зовсім тонке сопло має і ряд

недоліків: швидке засмічення, часте зупинення друку, крихкість шарів виробу. Це пов'язано із тим, що нитка, яку видавлює принтер, неоднорідна і пластична, а в процесі нагріву і охолодження нитка міняє свою форму, плавиться і вигинається. Так як це фізичний недолік термопластмас та схожих з ними матеріалів, тому при застосуванні FDM технології на 3D-принтері важко отримати якість друку вищу за 200-500 мікрон. Зазвичай, точність 3D-друку варіюється межах 0,3-0,7 мм більш ніж достатньо.

1.6 Огляд інформаційного середовища в сучасних системах 3D-друку

Інформаційне середовище 3D-друку – це сукупність технологій, програмного забезпечення, обладнання, матеріалів, а також людських ресурсів і процедур, які взаємодіють[15] для реалізації процесу створення об'єктів методом пошарового друку. Вона охоплює всі етапи виробництва, починаючи від цифрового проєктування та підготовки моделі до фізичного друку, моніторингу процесу, постобробки виробу та контролю якості.

Рис.1.15

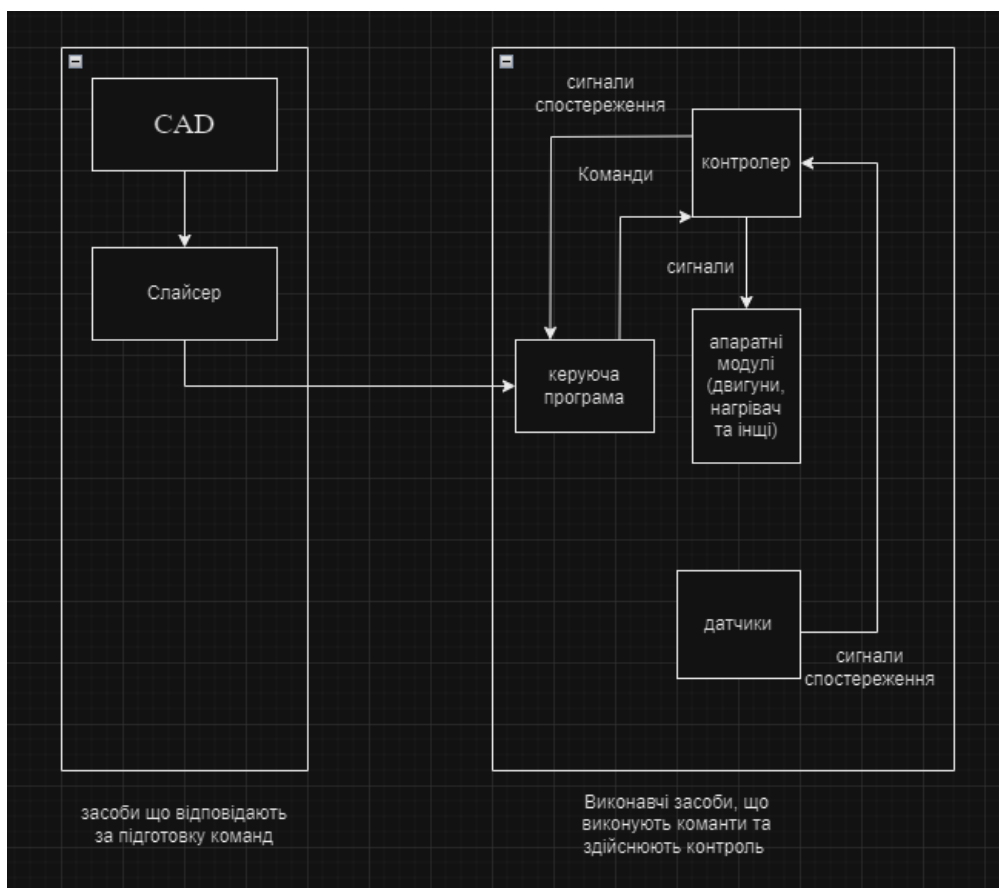


Рис.1.15 Схема інформаційного середовища 3D-друку

Перший етап – це **цифрове проєктування та моделювання**, де використовуються системи автоматизованого проєктування (CAD), такі як AutoCAD, SolidWorks, Fusion 360 чи Rhino, які дозволяють створити точну 3D-модель. Також використовуються програми для скульптингу, наприклад Blender чи ZBrush, а у випадку потреби – 3D-сканери для оцифрування реальних об'єктів. Наступним етапом є **підготовка до друку** з використанням слайсерів, таких як Cura, PrusaSlicer або Simplify3D, які перетворюють 3D-модель у G-коди, що є інструкціями для принтера.[16] На цьому етапі задаються параметри друку: товщина шару, швидкість, температура, заповнення моделі та наявність опорних структур.

Фізичне створення виробу відбувається на **обладнанні для 3D-друку**, яке включає різні типи принтерів: FDM/FFF для пластику, SLA/DLP для фотополімерів, SLS для порошкових матеріалів та DMLS для друку

металами. Обладнання працює під управлінням прошивок і контролерів, таких як Marlin, Klipper чи RepRapFirmware, що забезпечують точний рух механізмів та стабільний процес друку. Важливу роль відіграють **матеріали для друку**, які включають пластики (PLA, ABS, PETG), фотополімерні смоли, порошки з полімерів або металів, а також композитні матеріали з наповнювачами. Під час друку процес контролюється за допомогою сенсорів і моніторингових програм, наприклад OctoPrint чи Repetier-Server, які відстежують температуру, рух принтера та інші параметри. Після завершення друку об'єкт піддається **постобробці**, що включає шліфування, полірування, фарбування або термічну обробку для досягнення бажаного вигляду та властивостей. На завершальному етапі проводиться **контроль якості**, де використовуються програми для аналізу, такі як MeshLab або GOM Inspect, а також 3D-сканування для перевірки точності виробу. [16]

Таким чином, потік даних у процесі 3D-друку починається з проєктування моделі у CAD-системі, її нарізки на шари за допомогою слайсерів, передачі G-коду до контролерів принтера, виконання інструкцій обладнанням, моніторингу процесу, постобробки виробу та його перевірки на відповідність. У цій екосистемі беруть участь різні зацікавлені сторони: користувачі (інженери, дизайнери, архітектори), розробники програмного забезпечення, виробники обладнання, постачальники матеріалів та інтегратори, які налаштовують системи для конкретних завдань. Екосистема 3D-друку є багаторівневою та інтегрованою структурою, де кожен компонент взаємодіє для створення інноваційних рішень у виробництві, медицині, архітектурі, дизайні та інших сферах. [16]

Контролер 3D-принтера є ключовим компонентом в екосистемі 3D-друку, оскільки саме він відповідає за управління всім обладнанням під час процесу друку. Контролер є спеціалізованим мікроконтролером або платою, яка отримує інструкції у вигляді **G-коду** від слайсера, обробляє їх і перетворює на конкретні сигнали для механізмів принтера. Його основна функція –

координувати рухи крокових двигунів, які забезпечують переміщення екструдерів і робочої платформи по трьох осях (X, Y, Z). [16] Крім того, контролер керує температурою нагрівальних елементів, таких як **екструдер** і **робоча платформа**, підтримуючи їх у межах заданих параметрів для правильного розплавлення та осадження матеріалу.

Контролер також забезпечує обробку даних від **сенсорів**, які відстежують стан принтера, такі як датчики температури, оптичні датчики, кінцевики для визначення крайніх положень та інші елементи зворотного зв'язку. Завдяки цій інформації контролер може регулювати процес у реальному часі, забезпечуючи точність і стабільність роботи принтера. У випадку виявлення відхилень або помилок (наприклад, зміщення шарів чи перегріву), контролер може зупинити роботу обладнання або виконати коригування.

Контролери часто працюють під управлінням **прошивок**, таких як Marlin, Klipper, RepRapFirmware чи Duet, які визначають алгоритми управління принтером, підтримку різних датчиків, оптимізацію рухів і загальну ефективність друку. Вони також дозволяють користувачеві задавати та змінювати параметри процесу, такі як швидкість друку, прискорення, крокові налаштування, а також параметри калібрування для досягнення високої якості виробу. Деякі сучасні контролери підтримують **віддалене управління** через інтерфейси на основі Wi-Fi або Ethernet, що дозволяє користувачам контролювати процес друку в реальному часі за допомогою програм, таких як OctoPrint.

Таким чином, контролер 3D-принтера є «мозком» пристрою, який забезпечує узгоджену роботу всіх його компонентів, точне виконання інструкцій і стабільність процесу друку. Без контролера неможливо перетворити цифрову модель у фізичний об'єкт, оскільки він є центральним елементом, який пов'язує програмне забезпечення, механіку принтера та матеріали в єдину функціональну систему.

Висновки до першого розділу

У першому розділі дипломної роботи були розглянуті основні питання та проблеми, пов'язані з 3D-друком, а також можливості його оптимізації. Було проведено аналіз різних концепцій, зокрема підбір параметрів друку та аналіз якості виготовлених виробів. Встановлено, що існує значна кількість параметрів, які впливають на кінцевий результат, серед яких особливе значення мають різні види пластику з їх унікальними властивостями.

Основні проблеми включають велику кількість параметрів, які потрібно враховувати при налаштуванні процесу друку, відмінності у властивостях різних видів пластику, що вимагає ретельного підбору параметрів друку для кожного конкретного матеріалу, а також необхідність точного моніторингу та налаштування параметрів друку для забезпечення високої якості виробів.

Додатково було досліджено матеріали про предметну галузь, а також про місце інформаційних технологій у 3D-друці. Проаналізовано екосистему пристроїв та програмних комплексів, що задіяні в процесі.

Таким чином, розуміння цих аспектів є ключовим для розробки ефективних рішень у сфері 3D-друку та забезпечення високої якості продукції. Подальше дослідження спрямоване на розробку методів оптимізації параметрів друку та інтеграцію моделей бази знань для автоматичного керування процесом слайсингу.

Розділ 2. АНАЛІЗ МЕТОДІВ ІНТЕЛЕКТУАЛІЗАЦІЇ УПРАВЛІННЯ

2.1 Аналіз принципів інтелектуалізації управління

Інтелектуалізація технічного процесу виробництва – це процес впровадження сучасних технологій, які дозволяють автоматизувати і зробити виробництво більш ефективним, точним та гнучким. Це включає в себе використання різних інструментів, таких як штучний інтелект, системи машинного навчання, датчики та спеціалізоване програмне забезпечення, яке аналізує дані в режимі реального часу і приймає рішення на основі цих даних.[17] Наприклад, у виробництві 3D-друку система може самостійно коригувати параметри друку, такі як температура, швидкість екструзії, або навіть тип підтримки, враховуючи матеріали, які використовуються, і геометрію моделі. Це дозволяє уникнути людських помилок і значно підвищує точність і якість кінцевого продукту. Інтелектуалізація допомагає не тільки в автоматизації, але і в оптимізації всіх етапів виробничого процесу. Системи на базі штучного інтелекту можуть передбачити потенційні проблеми до їх виникнення, наприклад, поломки обладнання або дефекти в якості матеріалу, завдяки чому час простоїв значно зменшується. Вони також здатні швидко адаптувати виробничі процеси під нові умови, загальний принцип наведено в Рис. 2.1, наприклад, зміну партії матеріалу чи зміни в замовленнях, забезпечуючи таким чином безперервність роботи. Крім того, інтелектуалізація дозволяє значно знизити витрати на енергію та матеріали, оскільки система може оптимізувати витрати ресурсів в залежності від параметрів друку чи

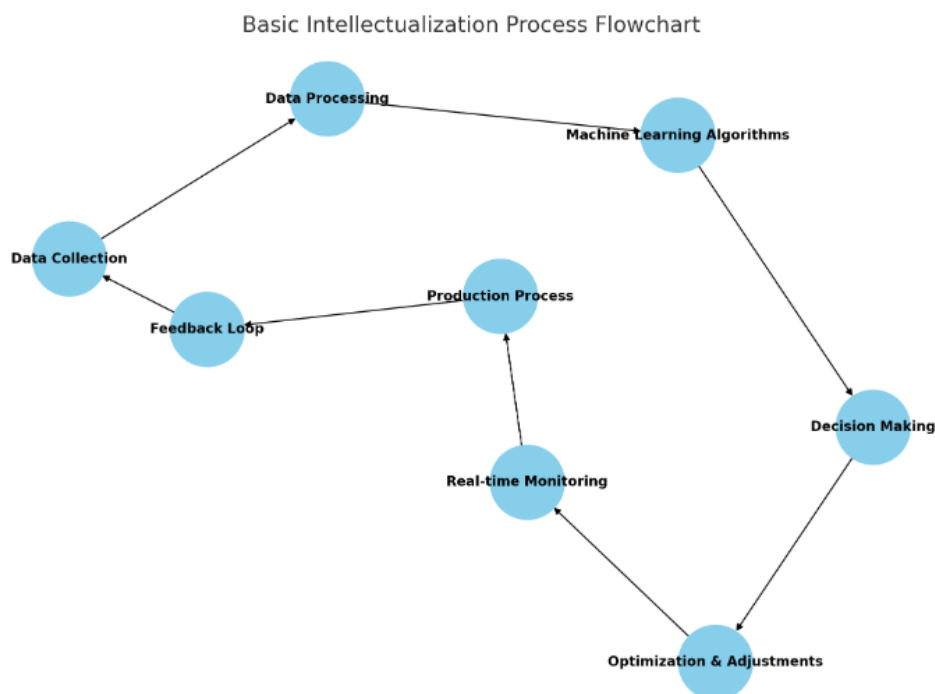


Рисунок 2.1 схема послідовності дій в роботі інтелектуалізованих системі

виробничих циклів.[18] Іншими словами, виробництво стає не тільки точним, а й економічно ефективним. Важливою частиною інтелектуалізації є також створення бази знань, яка зберігає досвід і налаштування, накопичені за час роботи, що дозволяє не тільки автоматизувати процес, а й підвищувати якість виробів із кожним новим циклом. Усе це в сукупності допомагає виробникам підвищити конкурентоспроможність, скоротити час на виробництво та знизити ймовірність виникнення браку.[18] Інтелектуалізація процесу виробництва – це шлях до більш гнучких, ефективних і стійких виробничих процесів, що в майбутньому стане стандартом у багатьох галузях. Це означає, що система може самостійно збирати і обробляти дані, приймати рішення без втручання людини, передбачати проблеми та коригувати виробничі параметри в режимі реального часу. Наприклад, у 3D-друці інтелектуальна система може оптимізувати налаштування друку залежно від матеріалу або зміни в умовах виробництва, що забезпечує високу точність і економію ресурсів. Інтелектуалізація не тільки підвищує ефективність, але й дозволяє

адаптувати процеси до нових умов і мінімізувати ризики.[20] Спрощений вигляд схеми інтелектуалізованої системи управління наведено на Рис 2.2

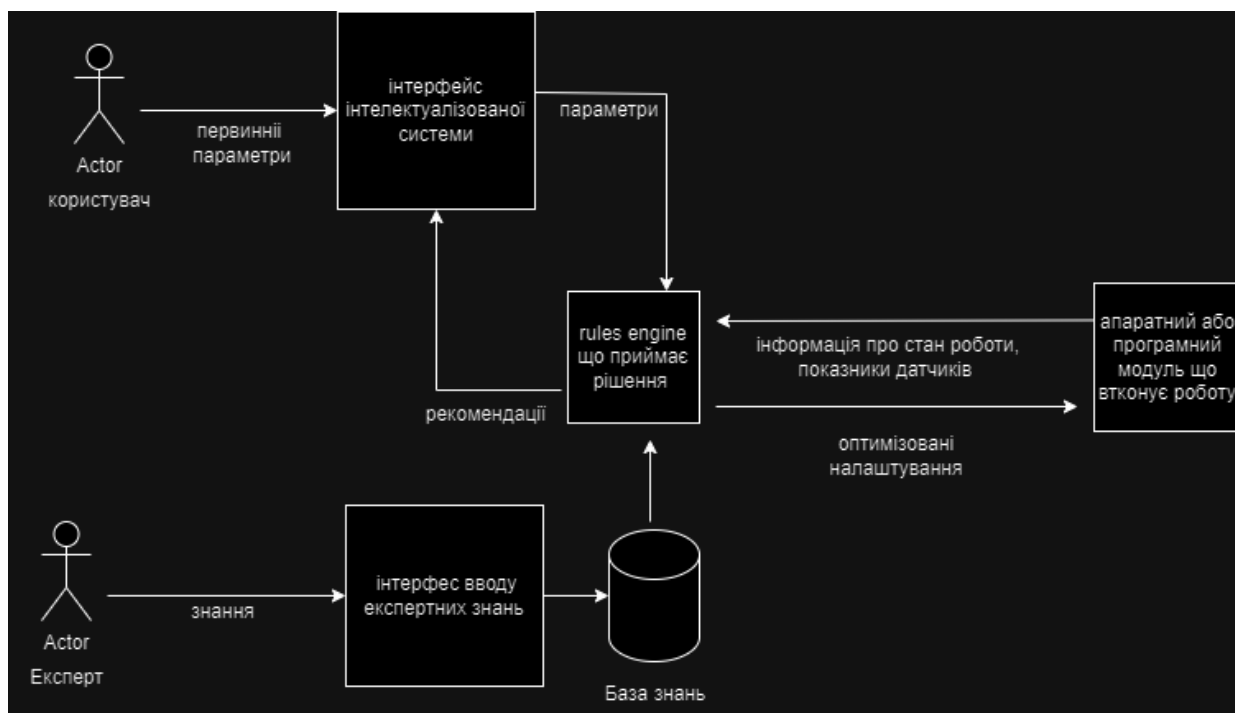


РИС.2.2 Спрощена модель інтелектуалізованого управління

2.2 Моделі представлення даних

Знання – це закономірності предметної області (принципи, закони, зв’язки), що набуті під час практичної діяльності та професійного досвіду, і в подальшому дозволяють фахівцям вирішувати задачі в цій області.

Під час обробки на ЕОМ знання, так само, як і дані трансформуються в кілька наступних етапів[21]:

1. Знання у пам’яті людини, як результат спостереження та мислення.
2. Знання, що перенесені на матеріальні носії (книги, підручники, довідники).
3. Сформоване поле знань – це умовний опис основних об’єктів предметної області, їх властивостей і закономірностей, що пов’язують ці об’єкти.
4. Знання, що описано і закладено у моделі представлення знань (продукційна модель, семантична сітка, фрейми).

5. Бази знань на машинних носіях інформації. Отже, знання – це добре структуровані дані про дані.

Існує багато способів визначення терміну «поняття», серед яких: Іntenсiонал поняття (гіпернім) – це визначення поняття через співвідношення з поняттями вищого рівня абстракції з вказанням специфічних властивостей. Іntenсiонали формулюють знання про об'єкти. 2 Екстенсiонал поняття (гіпонім) – це співвідношення поняття об'єкту з поняттями низького рівня абстракції або перелік факторів, які відносяться до об'єкту, що визначається. Для прикладу визначимо поняття «комп'ютер» Іntenсiонал: комп'ютер, це електронний обчислювальний пристрій, який можна купити за 300-1000 \$ і вирішувати за його допомогою широке коло завдань, і зокрема, спілкування в Інтернет. Екстенсiонал: комп'ютер – це IBM PC, Macintosh. Для збереження даних використовують Бази Даних. Вони мають великий об'єм і відносно невелику вартість зберігання інформації. Для збереження знань використовують Бази Знань. Тут, характерним є невеликий об'єм, але дорогі інформаційні масиви. База знань є основою для будь-якої інтелектуальної системи.[21]. Виникає питання у методах створення моделей БЗ їх прийнято поділяти на типи як вказано на Рис 2.3

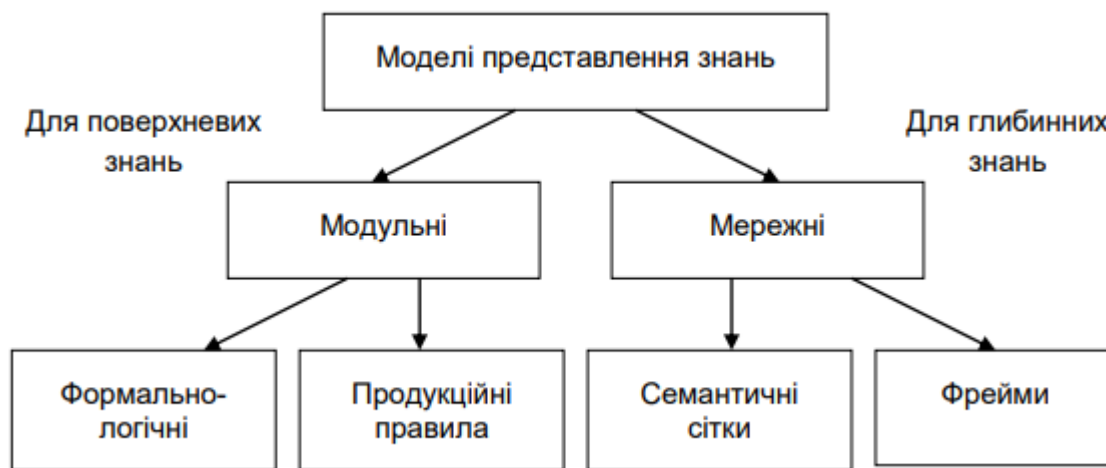


Рисунок 2.3 Схема класифікації моделей БЗ[21]

Логічна модель представлення

Логічна модель представлення знань використовується для відображення знань в системах, що ґрунтуються на першопорядковій логіці предикатів і логіці висловлень.

Формальна дедуктивна система, яка лежить в основі цієї моделі, може бути представлена як чотирикутник: PASB, де:

$$\phi = \langle B, S, A, P \rangle,$$

Основні переваги використання логіки предикатів для представлення знань полягають у тому, що потужний механізм виведення з чітко визначеними математичними властивостями може бути безпосередньо реалізований у програмному коді. Також можна контролювати логічну цілісність бази знань, перевіряючи її несуперечливість і повноту. Занесення фактів у таку базу легко реалізувати, що забезпечує компактність і простоту її структури. Однак основним недоліком логічної моделі є відсутність чітких принципів організації фактів у базі знань, через що великі бази важко піддаються аналізу та обробці.

Отже, логічні моделі доцільно застосовувати в галузях, де база знань має невеликий обсяг і відносно просту структуру.

Модель представлення знань на основі продукцій
Продукційна модель найбільш точно відображає процедурний характер знань. Вона описує знання через так звані правила продукції, які, по суті, є програмами у вигляді операторів типу "ЯКЩО <умова>, ТО <дія>".

Системи з базами знань, що ґрунтуються на цій моделі, називаються продукційними системами. Ці системи поділяються на два основних типи: з прямими та зворотними виведеннями.

У продукційних системах з зворотними виведеннями за допомогою правил будується дерево І/АБО, яке поєднує факти та висновки в єдину структуру. Оцінка цього дерева на основі наявних у базі даних фактів є процесом логічного виведення. Логічні виведення можуть бути прямими, зворотними або двонаправленими. У випадку прямого виведення початкові дані слугують відправною точкою, а процес оцінки припиняється на вузлах з запереченням. Заключення (якщо дерево не пройдене повністю) виступає як гіпотеза, що відповідає верхньому рівню дерева (кореню). Проте такий підхід має великий обсяг даних і часто включає оцінки, що не мають прямого відношення до висновку, що є надмірним. Перевага зворотного виведення полягає в тому, що оцінюються лише ті частини дерева, що безпосередньо стосуються висновку. Проте якщо неможливо підтвердити або спростувати твердження, створення дерева втрачає сенс. Двонаправлені виведення спочатку оцінюють невелику частину отриманих даних і формулюють гіпотезу, а потім запитують додаткові дані для підтвердження цієї гіпотези, що дозволяє створювати більш потужні та гнучкі системи.

Системи з прямими виведеннями мають найбільшу історію серед систем, що використовують знання, і вважаються основоположними. Вони складаються з трьох основних компонентів: бази правил, що включає набір продукцій (правил виведення), бази даних, яка містить безліч фактів, та інтерпретатора, що забезпечує логічне виведення на основі цих знань. База правил і база даних формують базу знань, а інтерпретатор виконує механізм логічного виведення. Виведення відбувається за циклом «розуміння – виконання», де кожен цикл виконання частини обраного правила оновлює базу даних. У результаті зміст бази даних трансформується з початкового стану до кінцевого, тобто досягається бажаний результат. Таким чином, для продукційних систем характерний простий цикл вибору та виконання правил. Однак через необхідність періодичного порівняння з шаблоном у базі

правил, зростання кількості правил значно знижує швидкість виведення. Тому такі системи не підходять для вирішення масштабних задач.

Сильні сторони:

Простота створення та розуміння окремих правил;

Легкість модифікації та розширення бази знань;

Простота механізму логічного виведення;

Природна модульність організації бази знань.

Слабкі сторони:

Нечіткість взаємозв'язків між правилами;

Складність оцінки загального стану знань;

Дуже низька ефективність обробки;

Відмінність від людської структури знань;

Відсутність гнучкості в логічному виведенні.

Семантична модель представлення знань

Основна ідея цього підходу до представлення знань полягає в тому, що предметна область розглядається як сукупність об'єктів і зв'язків між ними.

Семантична мережа, що представляє певну предметну область, є графом, де множині вершин відповідають об'єкти (поняття, події, процеси), а множині дуг – відношення на множині об'єктів. Вершини і дуги семантичної мережі позначаються іменами відповідних об'єктів та відносин предметної області.

Базовим функціональним елементом семантичної мережі є структура з двох компонентів: дуги та інцидентної їй пари вершин. Можна вважати, що кожен такий елемент представляє собою простий факт або висловлювання предметної області. Наприклад, висловлювання: «Сидоров вчиться в школі» можна представити у вигляді елемента семантичної мережі, зображеного на малюнку.

Сидоров → вчиться в → школа

Як видно з прикладу, дуга в мережі має напрямок, завдяки чому між поняттями в межах певного факту виражається відношення «суб'єкт/об'єкт». Очевидно, що кожна вершина графа може бути з'єднана з будь-якою кількістю інших вершин. Таким чином, формується мережа фактів.

Базову структуру семантичної мережі з точки зору логіки можна розглядати як еквівалент бінарного предиката. Тоді предикат відповідає напрямленій дузі, а два його аргументи – двом інцидентним цій дузі вершинам. Приклад на малюнку можна описати предикатом «вчитися (Сидоров, школа)».

У семантичних представленнях використовуються три основні типи об'єктів: поняття, події та властивості. Поняття зазвичай відповідають предметам або абстракціям. Під подією розуміють дії, які приводять до зміни ситуації. Властивості використовуються для уточнення або модифікації понять, подій чи інших властивостей.

При представленні знань семантичними мережами особливий практичний інтерес мають зв'язки таких видів: «є», «має», «являється». Зв'язок виду «являється» відображає належність до певного класу об'єктів. Зв'язки виду «має» і «є» вказують, відповідно, на те, що одне поняття є атрибутом іншого. Використовуючи ці відносини, можна досить легко представити складні сукупності фактів.

Перевагами моделей представлення знань за допомогою семантичних мереж є великі виразні можливості, наочність, близькість структури мережі до семантичної структури предметної області.

Основним недоліком таких моделей є слабо розвинена теоретична база логічного висновку, що призводить до посилення елемента довільності, який вводить розробник конкретної системи обробки знань на основі семантичної мережі. Тому процедури висновку в таких системах часто містять загрозу виникнення суперечностей, і, відповідно, процедури аналізу суперечливості в семантичних мережах потребують більше уваги, ніж, наприклад, у продукційних системах. Дуже часто цю функцію в семантичних моделях

покладають на людину. Таким чином, при досить великому обсязі знань рішення цієї задачі суттєво ускладнюється, що обмежує коло предметних областей, описуваних семантичними мережами, відносно невеликими проблемами.

Продукційна модель підходить для нашої системи більше за інші підходи, оскільки вона працює з простими командами, які можуть бути безпосередньо передані виконуючому пристрою на виробництві. У цьому випадку кожне правило продукції формулюється як умовно-результатна інструкція, що дозволяє чітко визначити, які дії потрібно виконати в залежності від наявних умов. Це дозволяє ефективно автоматизувати процеси управління, забезпечуючи швидке реагування на зміни в умовах виробництва та спрощує інтеграцію з різними пристроями, що використовуються для 3D-друку. Продукційна система може безпосередньо використовувати цю модель для прийняття рішень та контролю за виконанням завдань, забезпечуючи швидкість і точність операцій без потреби у складному програмуванні.

Фреймові моделі У 1975 році Марвін Мінські запропонував гіпотезу, згідно якої знання людини групуються у модулі – фрейми. Він розробив модель для позначення структури знань, що призначена для сприйняття просторових сцен. Фреймом тут називається структура даних, яка призначена для опису типових ситуацій або понять. В психології та філософії відоме поняття абстрактного образу. Наприклад, слово «кімната» викликає у людини образ кімнати: «житлове приміщення з чотирма стінами, підлогою, стелею, вікнами та дверима, площею 6-20 м² ». З цього опису нічого не можна прибрати (наприклад, прибравши вікна, отримаємо вже комору, а не кімнату), але в ньому є «дірки», або «слоти», - це незаповнені значення деяких атрибутів - кількість вікон, колір стін, висота стелі, покриття підлоги й ін. У цій теорії такий абстрактний образ називається фреймом. Визначення фрейму за Мінські Фреймом називається мінімально можливий опис певної сутності, такий, що подальше скорочення цього опису приводить до втрати цієї сутності.

Фреймом називається також і формалізована модель для відображення образу. В якості ідентифікатора фрейму слугує ім'я фрейму, яке має бути унікальним у всій фреймовій системі. Фрейм має певну внутрішню структуру, що складається з множини елементів (слотів), яким також привласнюються імена. За слотами слідують шпациї, в які поміщають дані, що представляють поточні значення слотів. Кожен слот у свою чергу представляється визначеною структурою даних. В значення слота підставляється конкретна інформація, що відноситься до об'єкта, який описується цим фреймом.

Машина виведення працює циклічно. В кожному циклі вона переглядає всі правила та існуючі на той момент факти. Рис 2.4

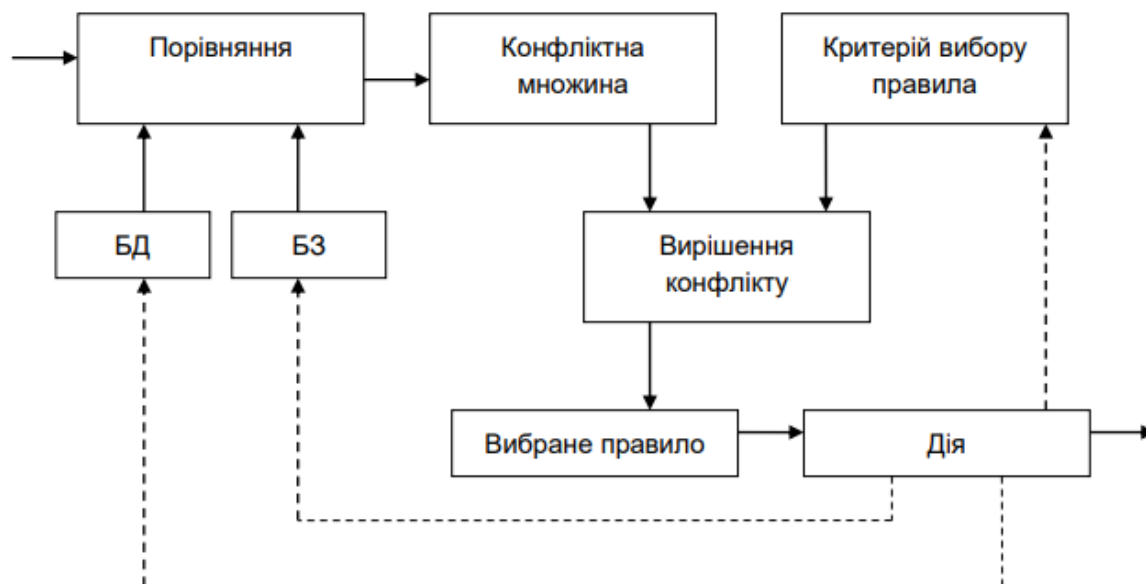


Рисунок 2.4 Цикл роботи машини виведення[21]

Факти з БД послідовно порівнюють з умовами правил для виявлення успішного порівняння. Сукупність вибраних правил складає конфліктну множину. Для вирішення конфлікту машина виведення має критерій, за яким вибирається лише одне правило. Факти, що утворюють висновок правила заносяться до БД або змінюють критерій вибору конфліктних правил. Якщо у висновку правила міститься кінцева дія, то вона виконується. Робота машини виведення залежить від стану БД та БЗ. На практиці враховується

поведінка машини виведення у попередніх циклах. Інформація про поведінку машини виведення (протокол системи) запам'ятовується у просторі станів.

2.3 Дослідження аналізу параметрів процесу друку

Ми вже знаємо що існує велика кількість параметрів які необхідно підтримувати для збереження рівня якості пробукції. Багато з них вже оптимізовані як раз за допомогою продуктових моделей, як наприклад підтримка температури, вказана температура зберігається в пам'яті, та постійно порівнюються з результатами датчиків, та корегується за потреби. Але існує велика кількість параметрів які вказуються в момент налаштування, та обраного матеріалу або сопла принтера.

Була проведена значна кількість досліджень різними інститутами, щоб проаналізувати вплив параметрів процесу на точність розмірів, шорсткість поверхні, час виготовлення та механічні властивості. Для цього оглядового дослідження використовувалися різні ключові слова для пошуку статей у наукових онлайн-базах даних (наприклад, Google Scholar, Elsevier, Springer тощо). Ці ключові слова включали FDM, моделювання методом плавленої депозиції, параметри процесу, оптимізація FDM та характеристики частин, виготовлених за допомогою FDM. У цьому дослідженні було включено відповідні опубліковані статті (журнали та матеріали конференцій) з 2005 по 2019 рік, але також були включені деякі важливі роботи з 2000 року. Всі статті, що були розглянуті, опубліковані лише англійською мовою. Під час початкового пошуку було зібрано понад 25 статей. Після первинного відбору за назвами та анотаціями було використано близько 10 статей як основний матеріал для цього оглядового дослідження. Окрім опублікованих статей, також використовувалися рекомендації виробників у вигляді посібників або веб-сторінок як джерела для властивостей філаментів або рекомендованих умов друку. Для попереднього підсумку проведеного огляду, на Рисунку 2.5

подано діаграму "риб'яча кістка" як візуальне зображення для ілюстрації впливу різних параметрів процесу на різні характеристики частин. Діаграма "риб'яча кістка" була розроблена на основі результатів різних існуючих досліджень. Зверніть увагу, що деякі параметри процесу перекриваються з характеристиками частин, оскільки один параметр процесу може впливати на кілька характеристик частини.

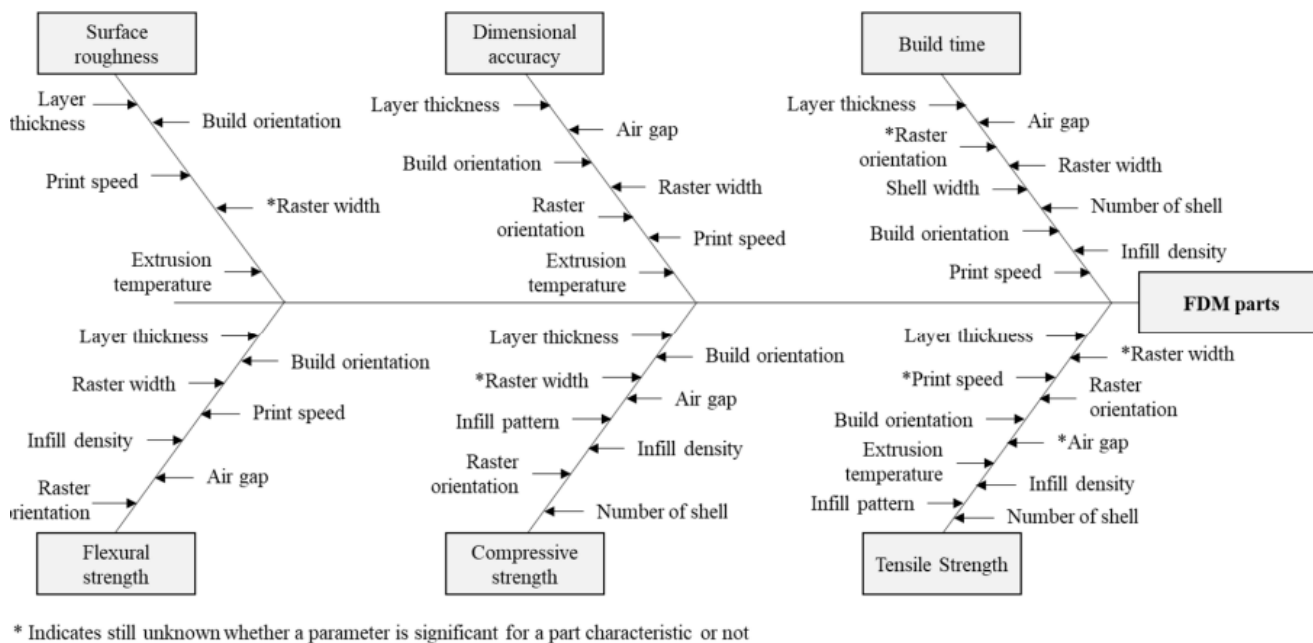


Рисунок 2.5. Діаграма "риб'яча кістка" для ілюстрації впливу параметрів процесу на характеристики частини.

У багатьох роботах використовувалися різні статистичні інструменти, такі як повний факторний дизайн [22,23], дробовий факторний дизайн [24,25] і централізований композитний дизайн з фасадно-центрованим методом (FCCCD) [30,33], щоб отримати максимальну інформацію з меншої кількості експериментів. Як інший підхід, оптимальні рівні параметрів для аналізу характеристик частин були визначені на основі експериментальних результатів. У деяких роботах, таких як роботи Мохамеда та ін. [27] і Чакон та ін. [25], були розроблені різні математичні моделі, що показують взаємозв'язок між параметрами процесу та якістю частини. Пень та ін. [34] і Райегані та ін. [35] працювали над оптимізацією математичних моделей

характеристик частин для визначення оптимальних рівнів параметрів процесу. Також були підходи, що розглядали характеристики кількох частин, які оптимізувалися одночасно. Для цього використовувалися різні евристичні алгоритми оптимізації, наприклад, оптимізація рою частинок з квантовою поведінкою (QPSO) [29], генетичний алгоритм із недомінованим сортуванням (NSGA-II) [31], диференціальна еволюція (DE) [35] та генетичний алгоритм (GA) [32]. У наступних підрозділах подано систематичний підсумок досліджень, що аналізують кожну характеристику якості частин, надрукованих за допомогою FDM.

Основні параметри для 3D-друку:

Швидкість друку (S) — швидкість екструзії матеріалу або руху друкуючої головки.

Температура екструдера (T) — температура, при якій матеріал розплавляється і екструдується через сопло.

Температура платформи (P) — температура друкованої платформи (столу), що допомагає запобігти деформаціям при охолодженні.

Висота шару (H) — товщина кожного шару, який наноситься принтером.

Тип матеріалу (M) — матеріал, що використовується для друку (PLA, ABS, PETG тощо).

Заповнення (F) — відсоток заповнення внутрішньої частини моделі (відповідає за щільність об'єкта).

Якість друку (Q) — кінцева якість об'єкта, що виводиться на основі комбінації параметрів.

Логічна формула для 3D-друку:

Як і в попередньому прикладі, побудуємо функцію, яка вказує на те, як параметри друку взаємопов'язані між собою:

$$Q = f(S, T, P, H, M, F) \quad (1.1)$$

Різні матеріали потребують різних температур для нормальної екструзії. Наприклад, PLA друкується при низькій температурі, а ABS потребує вищої температури. Якщо температура екструдера (T) занадто низька або занадто висока для конкретного матеріалу (M), це може призвести до поганої якості. Логічно це можна виразити так:

$$Q = (T_{min} \leq T \leq T_{max}) \quad (1.2)$$

Якщо швидкість друку (S) велика, це може знижувати якість друку, особливо якщо висота шару (H) мала, тому що матеріал не встигає рівномірно екструдуватися. Логічно це можна виразити так:

$$Q = (s \rightarrow H_{min}) \quad (1.3)$$

Об'єднана формула для якості 3D-друку, що враховує всі ці параметри, буде виглядати так:

$$Q = (S \rightarrow H_{min}) \cdot (T_{min} \leq T \leq T_{max}) \cdot (P_{min} \leq P \leq P_{max}) \cdot (F \geq F_{min}) \cdot (M \in \{PLA, ABS, PETG\}) \quad (1.4)$$

Окрім цього існує залежність від властивостей сопла, який вимагати збільшення температури, або типу екструдера, що впливає на обмеження швидкості друку.

Все це дозволяє зробити вивід що в процесі друку існує велика кількість взаємо залежних параметрів, які взаємо впливають один на одного, це створює

складнощі при вказанні налаштувань друку, та може бути оптимізовано за допомогою продуктової моделі баз знань.

Приклад:

Тип пластику і температурний діапазон:

ЯКЩО тип пластику = PLA, **ТО** діапазон температури екструдера 190–210°C.

ЯКЩО тип пластику = ABS, **ТО** діапазон температури екструдера на 230–250°C.

Ці правила дозволяють адаптувати температуру залежно від типу матеріалу для досягнення оптимальної адгезії та якості друку.

2.4 порівняння середовищ розробки МПЗ

Середовища розробки моделей представлення даних для створення експертних систем управління основаних на БЗ орієнтовані на обробку знань, прийняття рішень, логічний висновок і автоматизацію управлінських процесів. Ці системи використовують **інтелектуальне моделювання, машинне навчання, логічне програмування і бази знань** для підтримки прийняття рішень у різних галузях.[36] Таких середовищ існує досить багато. Та фактично відрізняються мовою програмування на якій засновані, та перймають їх особливості, так наприклад засновані на JAVA середовища ефективно інтегруються застосунками на цій же мові, в той час як мова що заснована на C, є більш гнучкою легко інтегрується з будь якими застосунками хоч і є більш простою. Розглянемо найпоширеніші середовища та їх особливості.

Звичайні мови програмування, такі як FORTRAN і C, розроблені та оптимізовані для процедурне маніпулювання даними (такими як числа та масиви). Однак люди часто вирішують складні проблеми проблеми з

використанням дуже абстрактних, символічних підходів, які не дуже підходять для реалізації в звичайні мови. Хоча абстрактна інформація може бути змодельована цими мовами, значна для перетворення інформації у формат, придатний для використання з процедурою, потрібні зусилля програмування парадигми програмування. Одним із результатів досліджень у галузі штучного інтелекту стала розробка техніки які дозволяють моделювати інформацію на вищих рівнях абстракції. Ці прийоми втілені в мови або інструменти, які дозволяють створювати програми, що дуже нагадують людську логіку і тому їх легше розробляти та підтримувати. Ці програми, які емулюють людину експертиза в чітко визначених проблемних областях, називаються експертними системами. Наявність експертної системи інструменти значно скоротили зусилля та витрати на розробку експертної системи. Інтегрована виробнича система мови C (CLIPS) [37] — це інструмент експертної системи, розроблений Відділ програмних технологій у Космічному центрі Джонсона NASA. Прототип CLIPS, версія 1.0. був розроблений навесні 1985 року в середовищі UNIX. Подальший розвиток CLIPS значно покращив його портативність, продуктивність і функціональність. Вийшов перший випуск CLIPS, версія 3.0 Липень 1986 р. Остання версія CLIPS, версія 6.0, була випущена в серпні 1993 р. Версія Також було розроблено CLIPS, повністю написаний на Ada, CLIPS/Ada. CLIPS наразі доступний для широкому загалу через Центр управління та інформації про комп'ютерне забезпечення[38]

Мова представлення кадрів **FRL**(Frame Representation Language) є експериментальною мовою написаний для вивчення використання фреймів як техніки представлення знань. Термін «рама», який використовується в FRL, був натхненний розробкою Мінського щодо рами теорія. FRL розширює традиційну схему представлення Property Lisp на дозволяючи властивості мати коментарі, значення за замовчуванням і обмеження, успадковувати відомості з абстрактних однотипних бланків і додаватися

процедури, ініційовані встановленням або видаленням значень, або якщо значення потрібне.[39]

KRL у формі, що проявляється в ресурсах знань

Онтологія OntoAgent є формальною моделлю світу, закодованою у чіткій мові представлення знань (KRL). Опис форми та змісту онтології, а також обґрунтування її створення, можна знайти у [44], розділ 7.1; аксіоматичне визначення представлено у розділі 7.1.6 цієї роботи; додаткові теоретичні та практичні питання обговорюються в [42], що готується до публікації. Тут ми наводимо основні моменти для загальної орієнтації.

Онтологія організована як ієрархічна колекція фреймів із множинним наслідуванням, яку очолюють концепти, позначені мовно-незалежними мітками. Наразі вона містить приблизно 9000 концептів, більшість з яких стосується загальної предметної області. Кількість концептів в онтології, що зараз становить близько 8000, значно менша за кількість слів чи фраз у будь-якій мові з кількох причин:

Синоніми та гіпоніми відображаються на один онтологічний концепт, а семантичні відтінки фіксуються у відповідних лексичних записах.

Багато лексичних одиниць описуються комбінацією концептів.

Лексичні одиниці, що представляють конкретну чи абстрактну точку на шкалі, вказують на єдину властивість, яка репрезентує цю шкалу.

Концепти розробляються з урахуванням мовної та культурної універсальності, тому ми зазвичай не вводимо концепти для вузьких понять, наприклад, таких як "відкликання купленого товару", оскільки малоімовірно, що всі мови чи культури мають цей концепт. Натомість значення таких слів описується композиційно у лексиконах тих мов, де вони використовуються. (Докладніше про розподіл між лексиконом та онтологією див. [43]).

Концепти поділяються на події, об'єкти та властивості. Властивості є примітивами, тобто їхнє значення ґрунтується на реальному світі без подальшої онтологічної декомпозиції. Виразність онтології підсилюється мультиваріантними значеннями властивостей, які реалізовані за допомогою фасетів. Фасети дозволяють онтології включати інформацію, таку як: "найтиповіші кольори автомобіля – білий, чорний, сріблястий та сірий; інші нормальні, але менш поширені кольори – червоний, синій, коричневий та жовтий; рідкісні кольори – золотий та фіолетовий".

Інвентар фасетів включає:

default – найбільш обмежений, дуже типовий піднабір значень;

sem – типові обмеження вибору;

relaxable-to – значення, які в принципі можливі, але не є типовими;

value – не обмеження, а фактичне, обов'язкове значення.

Приклад: обрані властивості з онтологічного фрейму для події DRUG-DEALING на Рис.2.6 демонструють використання фасетів:

DRUG-DEALING		
IS-A	value	CRIMINAL-ACTIVITY
AGENT	default	CRIMINAL, DRUG-CARTEL
	sem	HUMAN
	relaxable-to	SOCIAL-OBJECT
THEME	default	ILLEGAL-DRUG
INSTRUMENT	sem	MONEY
HAS-EVENT-AS-PART	sem	BUY, SELL
LOCATION	default	CITY
	sem	PLACE
	relaxable-to	PHYSICAL-OBJECT
...		

Рисунок 2.6 приклад коду для середовища KRL [41]

Об'єкти та події визначаються у середньому 16 властивостями кожен, багато з яких наслідуються, а не локально визначаються. Таким чином, значення об'єкта чи події визначається як набір триплетів: властивість-фасет-значення.

На відміну від більшості онтологій, онтологія OntoAgent включає складні події, також відомі як скрипти (пор. , що підтримують як симуляцію, так і логічне міркування про мову та світ.

Основні переваги написання онтології у KRL замість природної мови (NL) полягають у відсутності неоднозначності у KRL та можливості повторного використання цієї моделі для всіх природних мов. У перспективі, через двадцять років, онтологія OntoAgent повинна містити десятки тисяч добре описаних концептів, тисячі скриптів та логічні механізми, які можуть використовувати ці ресурси для підтримки роботи інтелектуальних агентів. Оскільки онтологія є мовно-незалежною, ця інфраструктура знань буде доступною для інтелектуальних агентів, які спілкуються будь-якою мовою, за умови наявності сумісного лексикону для відповідної мови.

Experta — це бібліотека для створення експертних систем на мові програмування Python. Вона надає інструменти для побудови систем, що опираються на базу знань і правила для автоматичного прийняття рішень на основі логічного виведення. Основною особливістю Experta є зручний синтаксис і легкість у використанні, що дозволяє створювати складні системи прийняття рішень з мінімальними зусиллями.

Бібліотека базується на концепції **правил і фактів**. Факти представляють собою знання про поточний стан задачі або ситуації, а правила описують логіку, за якою система робить висновки. Механізм логічного виведення у Experta працює на основі порівняння поточних фактів з умовами, визначеними в правилах, і виконує відповідні дії, коли умови виконуються. Цей підхід називається **продукційною моделлю**, оскільки правила задаються у вигляді "якщо-умова-то-дія".

Однією з ключових переваг Experta є **гнучкість і можливість розширення**. Система дозволяє легко додавати нові правила і факти, що робить її адаптивною для вирішення різних задач. Крім того, бібліотека підтримує

пріоритети правил, що дозволяє контролювати порядок їх виконання у разі конфліктів.

Серед інших переваг можна відзначити:

Простоту синтаксису: зрозумілий і декларативний підхід до опису правил та фактів.

Швидкість розробки: бібліотека дозволяє швидко створити прототип системи і поступово розширювати його.

Логічну прозорість: кожне правило працює незалежно, що дозволяє легко відслідковувати, як система прийшла до певного висновку.

Візуалізацію висновків: виведення результатів роботи у зрозумілому для користувача форматі.

Таким чином, *Experta* є ефективним інструментом для побудови експертних систем, оскільки поєднує у собі зручність розробки, прозорість логічного виведення та високу гнучкість. Вона ідеально підходить для задач, які потребують прийняття рішень на основі знань, виражених через правила, що особливо актуально для автоматизації процесів у 3D-друку. Приклад коду з використання бібліотеки на Рис 2.7

```

from experta import *

class SlicerExpertSystem(KnowledgeEngine):
    @Rule(Fact(material='PLA'))
    def pla_settings(self):
        self.declare(Fact(speed=60))
        self.declare(Fact(temperature=200))
        print("Рекомендовані параметри для PLA: швидкість 60 мм/с, температура 200°C.")

    @Rule(Fact(material='ABS'))
    def abs_settings(self):
        self.declare(Fact(speed=40))
        self.declare(Fact(temperature=230))
        print("Рекомендовані параметри для ABS: швидкість 40 мм/с, температура 230°C.")

def main():
    engine = SlicerExpertSystem()
    engine.reset()
    material_input = input("Введіть матеріал (PLA/ABS): ")
    engine.declare(Fact(material=material_input))
    engine.run()

if __name__ == "__main__":
    main()

```

Рисунок 2.7 Приклад коду для середовища **Experta**

Experta відрізняється від інших середовищ для розробки машин прийняття знань (МПЗ) своєю простотою, гнучкістю та інтеграцією з мовою програмування Python. На відміну від великих і складних систем, таких як **CLIPS**, **Jess** або інших інструментів для створення продукційних систем, **Experta** є легковаговою та мінімалістичною бібліотекою, яка орієнтована на швидку розробку та інтеграцію з існуючими Python-застосунками.

Однією з ключових особливостей Experta є використання **декларативного програмування**, що дозволяє описувати правила та факти у зрозумілій і компактній формі. На відміну від CLIPS або Jess, де програміст повинен писати код у спеціалізованих мовах, Experta використовує нативний Python-синтаксис, що робить її доступнішою для розробників, які вже працюють з

Python. Завдяки цьому значно зменшується кривина навчання, а процес розробки експертних систем стає швидшим і зручнішим.

Іншою важливою відмінністю є **інтегрованість у Python-екосистему**. Experta дозволяє безпосередньо використовувати численні бібліотеки Python, такі як NumPy, Pandas або інструменти для машинного навчання (наприклад, scikit-learn чи TensorFlow). Це дає можливість будувати гібридні системи, де логічне виведення поєднується з алгоритмами штучного інтелекту чи аналізом даних, що є значною перевагою над традиційними ізольованими інструментами для МПЗ.

На відміну від великих комерційних систем, **Experta є відкритим програмним забезпеченням** (open-source), що дозволяє розробникам адаптувати її під власні потреби та інтегрувати у будь-які проєкти без фінансових витрат. Це робить Experta популярною для наукових досліджень, навчання та швидкого прототипування систем на основі знань.

У технічному плані **Experta** підтримує **пряме виведення (forward chaining)** як основний механізм логічного аналізу, що забезпечує динамічну зміну фактів та ефективне прийняття рішень у реальному часі. При цьому можливість задавати **пріоритети правил** дозволяє ефективно вирішувати конфлікти та контролювати порядок виконання.

Таким чином, **Experta** вирізняється своєю простотою використання, інтеграцією з Python, гнучкістю та можливістю побудови гібридних систем, тоді як класичні середовища, такі як CLIPS чи Jess, вимагають використання окремих мов програмування і працюють більш ізольовано. Це робить Experta ідеальним вибором для розробників, які прагнуть створювати експертні системи швидко, ефективно та з мінімальними витратами на вивчення інструменту.

2.5. Проектування продукційної моделі бази знань прямого виводу

Продукційна модель знань представляє собою основу для проектування експертної системи. Її основні компоненти – це машина виводу, база правил та знань, також можна виділити робочу пам'ять. Продукційну модель було обрано через формальну задачу підбору стилю на основі тестування та подачі рекомендацій. Вона не виконує складних операцій, рахує сумарні ймовірності та дає рекомендації, тому дана модель представлення знань є найбільш вдалою. База знань поповнюється користувачем, відповідаючи на набір питань. Існують такі вхідні дані задачі: – 15 питань, для яких по 3–4 варіанти відповідей для кожного питання; – кожний варіант відповіді має значення ймовірності від 0 до 1; Вихідні дані задачі: – результати у вигляді рекомендації підбору стилю та візуального представлення результату та його динаміки у вигляді дерева рішень; – разом із результатами виводиться сумарна ймовірність для певного виду стилю, виводяться всі види, оскільки людина може обрати інший варіант з нижчою ймовірністю, що їй до вподоби. Питання та відповіді, що складають базу знань див. в додатку А. На основі цих даних створено БЗ продукційної моделі прямого виводу за допомогою простих конструкцій: If (умова) and (умова) or (умова) ... then (дія) If (умова) or (умова) and (умова) ... then (дія) Порядок виконання операцій залежить від порядку проведення логічної операції в конструкції.[45] База правил. Приклад правил Існує такий набір правил для моделі БЗ:

If (1 – 1), and (2 – 1), and (3 – 4), and (4 – 2), and (5 – 3), and (6 – 2), and (7 – 4), and (8 – 1), and (9 – 2), and (10 – 4), and (11 – 1), and (12 – 3), and (13 – 4), and (14 – 3), and (15 – 1), then (1 = 2,5), (2 = 0,5), (3 = 0,5), (4 = 0), (5 = 0,7), (6 = 0), (7 = 0,3), (8 = 2,5).

Примітка: в if перша цифра – номер питання, друга цифра – номер варіанта відповіді, у then перша цифра – номер рекомендації, друга цифра – значення ймовірності.

2.6. Метод дерева рішень Дерева рішень

Дерево має багато аналогій у реальному житті, і виявляється, що воно вплинуло на широку область машинного навчання, охоплюючи як класифікацію, так і регресію. При аналізі рішень дерево рішень може використовуватися для візуального та явного представлення рішень та прийняття рішень. Як випливає з назви, вона використовує деревоподібну модель рішень. Незважаючи на те, що інструмент інтелектуального аналізу даних часто використовується для вироблення стратегії досягнення певної мети. Рис. 2.5. Основні компоненти дерева рішень Представлення алгоритма у вигляді дерева Для цього треба розглянути дуже базовий приклад, який використовує набір даних для прогнозування виживання пасажирів чи ні. Нижче модель використовує 3 ознаки/атрибути/стовпці з набору даних, а саме стать, вік та супутні (кількість подружжя чи дітей разом).[45] Дерево рішень малюється догори дном, а корінь зверху. На зображенні зліва жирний текст чорним кольором представляє умову/внутрішній вузол, на основі 39 якого дерево розбивається на гілки/межі. Кінець гілки, яка більше не ділиться – це рішення/лист, в даному випадку чи загинув пасажир, чи вижив, як червоний та зелений текст відповідно.

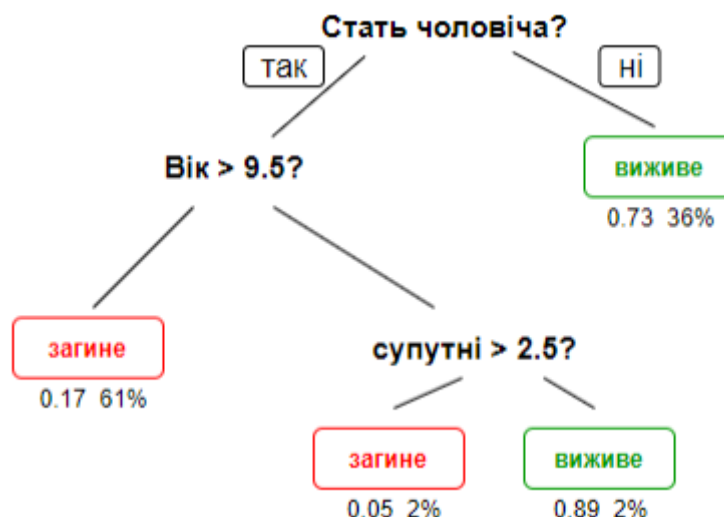


Рис. 2.8. Приклад дерева рішень

Хоча справжній набір даних матиме набагато більше можливостей, і це буде просто гілкою набагато більшого дерева, але не можна ігнорувати простоту цього алгоритму. Важливість функції зрозуміла, і відносини можна легко переглянути. Ця методологія більш відома як дерево прийняття рішень за даними, і вище дерево називається деревом класифікації, оскільки метою є класифікація пасажирів як вижилих або загиблих. Древа регресії представлені однаково, просто вони мають на увазі безперервні значення, такі як ціна будинку. Загалом алгоритми дерева рішень називаються CART або класифікаційними та регресійними деревами. “Вирощування” дерева передбачає прийняття рішення про те, які особливості вибрати і які умови використовувати для розбиття поряд із знанням часу зупинки. Оскільки дерево зазвичай росте у довільній манері, його необхідно буде обрізати, щоб воно виглядало так як потрібно. Саме тому існують різні загальні техніки, які використовуються для розбиття. Дерево рішень – один із керованих алгоритмів машинного навчання. Цей алгоритм може бути використаний для задач регресії та класифікації, але в основному використовується для задач класифікації. Дерево рішень існує за набором умов if - else, щоб візуалізувати дані та класифікувати їх відповідно до умов.



Рис. 2.9. Приклад дерева рішень

Метод дерева рішень – це метод аналізу величезних масивів даних. Слід також зазначити, що метод дерева рішень використовує ієрархічну структуру. Основні складові алгоритму дерева рішень Кореневий вузол такий атрибут використовують для розподілу даних на два або декілька наборів. Атрибут об'єкта в цьому вузлі обирається на основі методів вибору атрибутів. Гілка або піддерево Частина загального дерева рішень називається гілкою або піддеревом. Розбиття та Ділення вузла на два або більше підвузлів на основі умов if– else. Вузол прийняття рішень Після поділу підвузлів на подальші підвузли він називається вузлом прийняття рішень. Лист або термінальний вузол Кінець дерева рішень, де його не можна розділити на подальші підвузли. Обрізання Видалення підвузла з дерева називається обрізанням.

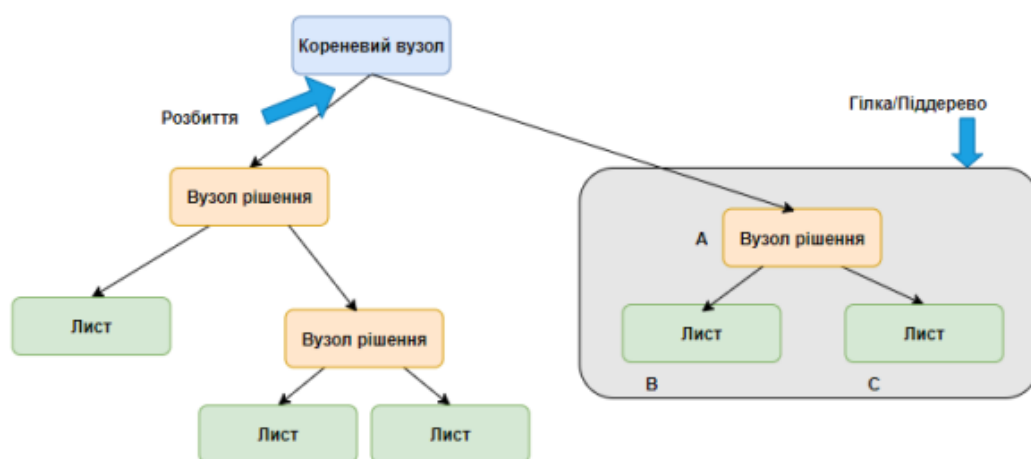


Рис. 2.10. Основні складові алгоритму дерева рішень

Робота дерева рішень Функція кореневого вузла обирається на основі результатів вимірювання вибору атрибутів (ASM). ASM повторюється доти, доки кінцевий вузол або кінцевий вузол не можна розділити на підвузли.



Рис. 2.11. Приклад генерації дерева рішень

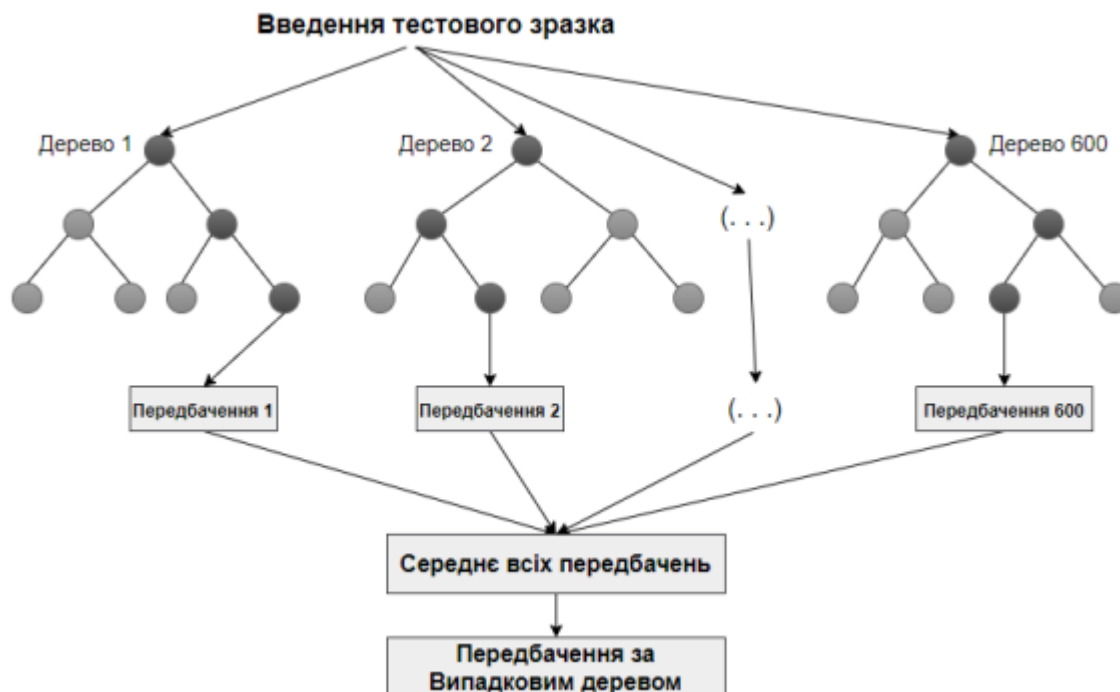


Рис. 2.12. Алгоритм побудови дерева рішень

Модель дерева рішень Модель дерева рішень представляє собою модель розрахунків, в якій алгоритмом є дерево рішень тобто послідовність розбиття об'єктів на основі порівняння. Просте дерево рішень Модель, в якій кожне рішення з'являється за допомогою порівняння двох чисел. Застосовується для сортування та пошуку. Лінійне дерево рішень Це просте дерево рішень, проте рішення з'являється за допомогою сукупності деяких значень вхідних даних та мають три вихідні гілки рішень. Використання методу дерева рішень: – зберігання інформації в необ'ємній і зручній формі, яка містить точні описи об'єктів; – застосування завдання класифікації – віднесення об'єктів до класів; – застосування регресії – визначення залежності змінної від незалежних вхідних змінних у випадку недостовірних значень змінної; – покращення якості та точності прийняття рішень; – графічне відображення та візуалізація послідовності рішень.

Висновки до другого розділу

У другому розділі було проведено детальний аналіз методів інтелектуалізації управління 3D-друком. Аналізувалися існуючі підходи, зокрема бази знань і моделі експертної оцінки, які можуть значно покращити процес друку. Використання баз знань дозволяє ефективно акумулювати та застосовувати накопичений досвід і знання, що призводить до кращого вибору параметрів друку та підвищення якості виробів. Моделі експертної оцінки забезпечують можливість більш точного налаштування процесу на основі експертного аналізу, що підвищує надійність і якість кінцевих продуктів.

Було також проаналізовано параметри друку, які підлягають інтелектуалізації. Це включає вибір матеріалів, налаштування температурних режимів, швидкість друку, товщину шарів та інші параметри, що безпосередньо впливають на якість і точність виробів.

Дослідження існуючих середовищ для створення моделей баз знань показало, що ці платформи забезпечують гнучкість і масштабованість у процесі розробки інтелектуальних систем управління 3D-друком. Вони дозволяють інтегрувати різні інструменти та технології для оптимізації процесу друку та забезпечення високої якості продукції.

У підсумку, використання інтелектуальних підходів, таких як бази знань і моделі експертної оцінки, значно покращує процес 3D-друку, підвищуючи його ефективність і якість. Подальші дослідження в цьому напрямку спрямовані на вдосконалення існуючих технологій і розробку нових методів для ще більшої оптимізації та автоматизації процесу.

РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО ЗАСТОСУНКУ

3.1 Обґрунтування вибору засобів керування принтера

Для реалізації поставленої задачі, де програма на основі експертної системи поєднується зі слайсером **OrcaSlicer** та контролером **Klipper**, розміщеним на сервері, вибір саме такої архітектури є найбільш доцільним та обґрунтованим завдяки її функціональності, гнучкості та ефективності управління 3D-друком.

OrcaSlicer є популярним інструментом для підготовки 3D-моделей до друку. Завдяки відкритій структурі, підтримці Python-скриптів і можливості конфігурування параметрів, OrcaSlicer дозволяє інтегрувати зовнішні програмні рішення. Це забезпечує можливість зчитування вхідних даних, таких як тип матеріалу, висота шару, температура чи швидкість друку, та передавати їх у створену програму на основі **Experta**. Саме ця експертна система буде виконувати інтелектуальний аналіз і оптимізацію параметрів друку залежно від заданих умов і передавати результати назад до слайсера. Завдяки такій інтеграції, весь процес підготовки до друку автоматизується, що зменшує необхідність ручного налаштування та знижує ризик помилок. Контролер **Klipper**, який розміщується на сервері, є ще однією важливою складовою цієї архітектури. Klipper відрізняється своєю високою продуктивністю, оскільки поєднує можливості мікроконтролера з обчислювальною потужністю сервера. Це дозволяє виконувати складні обчислення та керувати параметрами 3D-друку в реальному часі, забезпечуючи точність і стабільність роботи принтера. Серверна архітектура дозволяє програмі, створеній на Python, взаємодіяти з контролером Klipper для динамічного коригування параметрів друку, таких як температура, швидкість руху екструдера або охолодження, що є критично важливим для досягнення високої якості кінцевого виробу.

Таке поєднання OrcaSlicer, експертної системи на базі **Experta** та контролера Klipper дозволяє реалізувати **замкнутий цикл автоматизованого управління** процесом 3D-друку. Спочатку слайсер готує вихідні дані для експертної системи, яка на основі закладених правил і логічного виведення обирає оптимальні параметри друку. Далі ці параметри передаються до Klipper, який безпосередньо керує 3D-принтером у реальному часі. Така архітектура забезпечує гнучкість у налаштуванні, можливість швидкого реагування на зміни умов та адаптивне управління процесом друку.

Окрім того, використання **серверного рішення** з Klipper дозволяє розмістити всю логіку управління та обчислень на віддаленому сервері, що зменшує навантаження на сам принтер і підвищує продуктивність системи. Завдяки цьому користувач отримує надійне та високотехнологічне рішення для автоматизації 3D-друку, що поєднує зручність OrcaSlicer, інтелектуальність експертної системи та продуктивність контролера Klipper.

3.2 Обґрунтування вибору інструменту розробки МПЗ

Для реалізації завдання розробки експертної системи, що інтелектуалізує підбір параметрів слайсера у сфері 3D-друку, було обрано мову програмування **Python** та бібліотеку **Experta**. Обґрунтування вибору цих інструментів базується на їхніх особливостях, перевагах та відповідності поставленим вимогам задачі.

Python є однією з найпопулярніших мов програмування для наукових і технічних обчислень завдяки своїй простоті, читабельності та широкому набору бібліотек. Оскільки завдання потребує інтелектуалізації процесу на основі правил і знань, Python забезпечує оптимальне середовище для швидкого прототипування, гнучкої розробки та інтеграції з іншими інструментами. Окрім того, Python підтримує численні бібліотеки для аналізу

даних, машинного навчання та побудови інтерфейсів користувача, що дозволяє створити комплексне та інтегроване рішення.

Бібліотека **Experta** була обрана для реалізації ядра експертної системи, оскільки вона забезпечує зручний та ефективний спосіб побудови продукційних систем на основі правил. На відміну від інших інструментів, таких як CLIPS або Jess, Experta дозволяє використовувати нативний синтаксис Python, що значно спрощує процес розробки. За допомогою **Experta** можна чітко структурувати знання у вигляді **фактів і правил**, які формують логічний механізм виведення. Це особливо важливо для задачі підбору параметрів слайсера, оскільки вона вимагає врахування численних умов і залежностей, таких як тип матеріалу, висота шару, швидкість друку, температура тощо.

Використання **Experta** також обґрунтоване її підтримкою **прямого виведення (forward chaining)**, що дозволяє системі автоматично виводити нові факти на основі наявної бази знань. Це забезпечує динамічний підхід до прийняття рішень, оскільки результати системи можуть змінюватися в залежності від вхідних даних. Додатковою перевагою є можливість задавати **пріоритети правил**, що допомагає вирішувати конфлікти між правилами та контролювати їх виконання, забезпечуючи точність і надійність роботи системи.

Для зручності взаємодії з користувачем та введення вихідних даних було обрано інструменти для створення **графічного інтерфейсу** (наприклад, **Tkinter** або **PyQt**). Це дозволяє реалізувати інтуїтивно зрозумілий інтерфейс, де користувач може вводити параметри 3D-друку (наприклад, тип матеріалу чи вимоги до якості друку), а система автоматично підбере оптимальні значення для слайсера.

Таким чином, вибір **Python** у поєднанні з **Experta** є виправданим з огляду на швидкість розробки, доступність інструментів, гнучкість та можливість побудови експертних систем із прозорим механізмом виведення.

Використання графічного інтерфейсу додатково підвищує зручність застосування програми для кінцевого користувача. Це забезпечує ефективне рішення задачі підбору параметрів слайсера у сфері 3D-друку на основі експертної системи.

Experta є більш зручною для інтеграції в слайсер Orca завдяки своїй гнучкості, простоті використання та можливості працювати в середовищі Python, що є однією з найпопулярніших мов програмування для розробки інструментів у галузі 3D-друку. Слайсери, такі як OrcaSlicer, вже активно використовують Python-скрипти для розширення функціональності, автоматизації процесів і налаштування параметрів друку, що забезпечує ідеальну сумісність із бібліотекою Experta.

Однією з ключових причин є те, що Experta підтримує продукційну модель на основі правил і фактів, що дозволяє легко адаптувати систему до специфічних потреб Orca. У задачі підбору параметрів для слайсера важливо враховувати багато змінних, як-от тип матеріалу, температура, швидкість друку та товщина шару. Experta дозволяє структуровано описати ці залежності за допомогою правил у зрозумілому синтаксисі Python. Це дозволяє автоматизувати процес прийняття рішень, що особливо важливо для оптимізації параметрів друку та забезпечення високої якості кінцевого виробу.

Крім того, Experta є легковаговою бібліотекою, яка легко інтегрується в існуючий код або середовище OrcaSlicer без додаткових витрат на ресурси. Завдяки своїй відкритій архітектурі та можливості використання Python API Рис 3.1., Experta може працювати разом з іншими інструментами, які часто застосовуються у слайсерах, такими як NumPy для числових обчислень чи Pandas для обробки даних. Це дозволяє створити гібридну систему, у якій експертна логіка може поєднуватися з алгоритмами оптимізації чи машинного навчання для більш точного підбору параметрів.

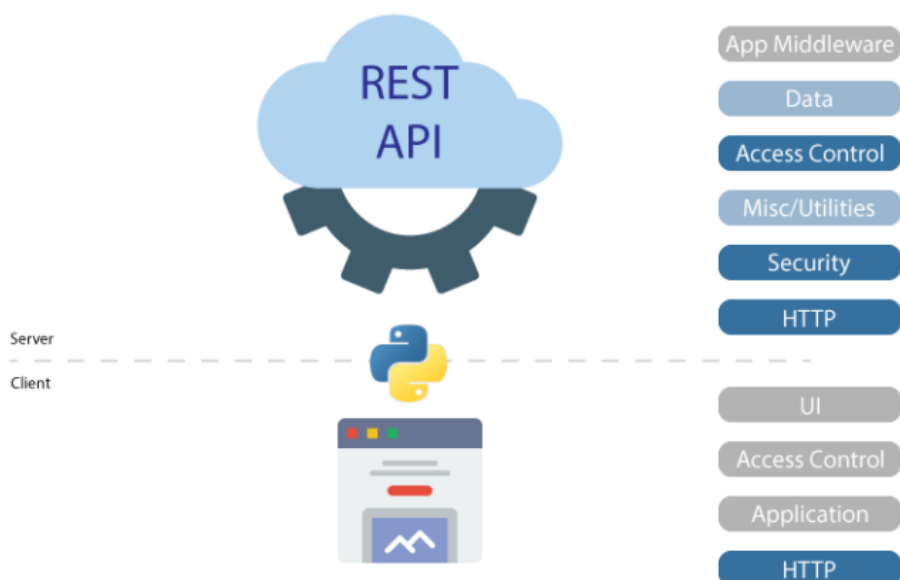


Рисунок 3.1. REST API backend[47]

Ще однією перевагою є швидке прототипування та гнучка адаптація системи під специфічні потреби користувачів OrcaSlicer. Якщо необхідно додати нові матеріали чи врахувати інші фактори, такі як геометрія моделі або швидкість охолодження, нові правила можна легко інтегрувати без суттєвих змін у структурі системи. Це дозволяє значно спростити подальший розвиток та обслуговування системи.

Таким чином, Experta є більш зручною для інтеграції в слайсер Orca завдяки своїй сумісності з Python, гнучкості у створенні систем на основі правил, можливості легко адаптуватися до нових умов і мінімальним вимогам до ресурсів. Це робить її ідеальним інструментом для автоматизації підбору параметрів друку та підвищення ефективності роботи слайсера.

3.3 Програма визначення параметрів друку на основі продуктивної моделі бази знань

У цьому прикладі ми задаємо правила для вибору оптимальних параметрів друку на основі введених користувачем значень, таких як тип матеріалу, товщина шару і швидкість друку.

Оголошення фактів: В програмі створено клас `PrintingParams`, який представляє факти для параметрів друку, таких як тип матеріалу, товщина шару та швидкість друку. Це дозволяє зберігати інформацію про поточний стан налаштувань друку.

```
from experta import *

# Оголошуємо факти для параметрів 3D-друку
class PrintingParams(Fact):
    """Факти для параметрів друку"""
    pass

class PrintingSystem(KnowledgeEngine):
    @DefFacts()
    def _initial_action(self):
        yield PrintingParams(material=None, layer_height=None, print_speed=None, extruder_temp=None, bed_temp=None)

    @Rule(PrintingParams(material="PLA", extruder_temp=MATCH.temp, bed_temp=MATCH.bed_temp))
    def set_temperature_pla(self, temp, bed_temp):
        print(f"Для PLA матеріалу:")
        print(f"Температура екструдера: {temp}°C")
        print(f"Температура столу: {bed_temp}°C")

    @Rule(PrintingParams(material="ABS", extruder_temp=MATCH.temp, bed_temp=MATCH.bed_temp))
    def set_temperature_abs(self, temp, bed_temp):
        print(f"Для ABS матеріалу:")
        print(f"Температура екструдера: {temp}°C")
        print(f"Температура столу: {bed_temp}°C")

    @Rule(PrintingParams(material="PETG", extruder_temp=MATCH.temp, bed_temp=MATCH.bed_temp))
    def set_temperature_petg(self, temp, bed_temp):
        print(f"Для PETG матеріалу:")
        print(f"Температура екструдера: {temp}°C")
        print(f"Температура столу: {bed_temp}°C")

    @Rule(PrintingParams(layer_height=0.1, print_speed=MATCH.speed))
    def set_slow_speed_for_thin_layer(self, speed):
        print(f"Для товщини шару 0.1 мм рекомендується швидкість: {speed} мм/г")

    @Rule(PrintingParams(layer_height=0.2, print_speed=MATCH.speed))
    def set_faster_speed_for_thicker_layer(self, speed):
        print(f"Для товщини шару 0.2 мм рекомендується швидкість: {speed} мм/г")
```

Рисунок 3.2. блок коду з встановленням параметрів та правил

Правила для налаштувань: Створені правила, які визначають параметри для різних типів матеріалів і товщини шару:

Якщо матеріал PLA, то встановлюються відповідні температури для екструдера та столу.

Якщо товщина шару 0.1 мм, рекомендується зменшити швидкість друку до 40 мм/с, щоб забезпечити високу якість.

Для товщини шару 0.2 мм швидкість друку збільшується до 60 мм/с.

Інтерактивність: Користувач вводить тип матеріалу та товщину шару, після чого система генерує рекомендації для оптимальних параметрів друку.

Запуск і виведення результатів: Після введення параметрів, система надає рекомендації щодо температури екструдера, температури столу і швидкості друку на основі введених значень.

```
@Rule(PrintingParams(material=MATCH.material, layer_height=MATCH.layer_height, extruder_temp=MATCH.temp, bed_temp=MATCH.bed_temp, print_speed=MATCH.speed))
def print_recommendations(self, material, layer_height, temp, bed_temp, speed):
    print(f"Рекомендації для матеріалу {material}:")
    print(f"Товщина шару: {layer_height} мм")
    print(f"Температура екструдера: {temp}°C")
    print(f"Температура столу: {bed_temp}°C")
    print(f"Швидкість друку: {speed} мм/хв")

# Ініціалізація та запуск системи
engine = PrintingSystem()

# Введення параметрів користувача
material = input("Виберіть матеріал (PLA/ABS/PETG): ").upper()
layer_height = float(input("Введіть товщину шару (0.1 або 0.2): "))
extruder_temp = int(input(f"Введіть температуру екструдера для {material}: "))
bed_temp = int(input(f"Введіть температуру столу для {material}: "))
print_speed = int(input("Введіть швидкість друку (мм/хв): "))

# Заявляємо факти в експертну систему
engine.reset() # очищаємо поточний стан
engine.declare(PrintingParams(material=material, layer_height=layer_height, extruder_temp=extruder_temp, bed_temp=bed_temp, print_speed=print_speed))

# Запускаємо виведення рекомендацій
engine.run()
```

Рисунок 3.3. Блок коду з обробкою та створенням рішення

Цей код можна розширювати, додаючи нові типи матеріалів, налаштування або правила для інших факторів, таких як тип моделі чи необхідність підтримки.

Щоб застосувати машинне навчання для коригування параметрів 3D-друку на основі даних з датчиків, можна використовувати підхід, де алгоритм машинного навчання навчається на історичних даних, щоб передбачати найкращі параметри для заданих умов. Кожен набір вхідних даних (наприклад, температура, швидкість друку, товщина шару) буде використовуватися для коригування параметрів друку в реальному часі в межах встановлених діапазонів, щоб досягти найкращих результатів. Зазначені діапазони для кожного параметра задаються експертом.

У цьому випадку можна використовувати **регресійні моделі машинного навчання**, такі як лінійна регресія, або більш складні моделі, наприклад, **дерева рішень** або **нейронні мережі** для прогнозування оптимальних значень параметрів.

```
import numpy as np
import pandas as pd
from sklearn.tree import DecisionTreeRegressor
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.tree import export_text

# Формат: Температура екструдера, Температура столу, Швидкість друку, Товщина шару, Якість друку (метрика, яку ми хочемо оптимізувати)

data = {
    'extruder_temp': [200, 210, 220, 230, 240, 250],
    'bed_temp': [60, 65, 70, 75, 80, 85],
    'print_speed': [40, 50, 60, 70, 80, 90],
    'layer_height': [0.1, 0.15, 0.2, 0.25, 0.3, 0.35],
    'print_quality': [95, 92, 90, 85, 80, 75] # Оцінка якості друку
}
```

Рисунок 3.4. Блок коду з заданими діапазонами параметрів

Деревовидна модель, або дерево рішень, є популярним інструментом машинного навчання, що використовується для класифікації та регресії. Вона працює за принципом поділу даних на підмножини на основі критеріїв, що найкраще розділяють різні класи або прогностичні значення. Структура дерева рішень подібна до дерева, де кожен вузол представляє умови перевірки (порівняння), а гілки ведуть до інших умов або до результатів (лістя), які дають фінальний прогноз.

У контексті 3D-друку, дерево рішень може бути використано для оптимізації параметрів друку, таких як температура екструдера, температура столу, швидкість друку та товщина шару, для досягнення найкращої якості друку. Використовуючи історичні дані, модель дерева рішень може навчитися, які комбінації параметрів дають найкращі результати.

Процес навчання дерева рішень полягає в тому, що алгоритм аналізує вхідні дані, шукаючи оптимальні точки розподілу (пороги), які мінімізують помилку у прогнозах. Наприклад, дерево рішень може визначити, що при температурі екструдера 230°C і температурі столу 70°C якість друку буде найвищою, і цей набір значень буде виведений як рекомендація.

Однією з основних переваг дерева рішень є його прозорість. Користувач може побачити, як модель приймає рішення, оскільки вона базується на простих умовах, які можна легко інтерпретувати. Це робить дерево рішень зручним для систем, де важлива зрозумілість і можливість корекції результатів.

Деревовидні моделі також добре працюють з категоріальними та числовими даними, що робить їх універсальними для різних типів вхідних даних, таких як температура, швидкість друку, товщина шару тощо. Крім того, дерева рішень є стійкими до шуму в даних і можуть добре адаптуватися до різних умов друку, що робить їх ефективним інструментом для автоматичного коригування параметрів 3D-друку в реальному часі.

У разі необхідності покращення результатів, дерево рішень можна адаптувати, використовуючи методи, такі як **підвищення (boosting)** або **вибірка з випадковими лісами (random forests)**, які дозволяють отримати більш точні прогнози.

Дерево рішень працює шляхом послідовного розбиття даних на підмножини на основі умов, що найкраще розділяють ці дані. Кожен розподіл або поділ створюється на основі певної ознаки, наприклад, температури екструдера або швидкості друку. Алгоритм аналізує кожну ознаку і визначає, яка з них найкраще відповідає за принципом "розділити і перемогти", тобто забезпечує найкраще розділення різних категорій або значень у даних. Кожен внутрішній вузол дерева представляє перевірку певної умови, наприклад, "Чи температура екструдера більше ніж 220°C?", а гілки дерева ведуть до подальших умов або до кінцевих результатів. Ці умови ділять дані на дві або більше частини, залежно від того, як вони впливають на кінцевий результат. Коли дерево рішень будується, алгоритм намагається мінімізувати помилку в прогнозах, тобто шукає найбільш інформативні умови, які дозволяють максимально точно прогнозувати результат. Кожен вузол дерева визначається так, щоб розділяти дані таким чином, щоб подальші прогнози

ставали все точнішими. Цей процес триває, поки не буде досягнуто певного рівня точності або доки дерево не стане занадто великим.

У контексті 3D-друку, алгоритм може почати з аналізу температури екструдера. Якщо вона перевищує певний поріг, дерево може рухатися далі і перевіряти наступний параметр, наприклад, температуру столу. Якщо температура екструдера знаходиться в певному діапазоні, дерево може передбачити певну якість друку. Врешті-решт, на основі кількох умов дерево приймає рішення про прогнозовану якість друку

Основною перевагою дерева рішень є його прозорість: кожен крок і рішення в ньому можна легко зрозуміти і інтерпретувати. Це дозволяє користувачам бачити, як і чому модель приймає певне рішення. Якщо, наприклад, параметри друку ведуть до поганої якості, дерево рішень чітко покаже, на яких саме умовах можна було б змінити параметри для покращення результату.

```
df = pd.DataFrame(data)

# Окремо виділяємо вхідні дані та цільову змінну (якість друку)
X = df[['extruder_temp', 'bed_temp', 'print_speed', 'layer_height']]
y = df['print_quality']

# Масштабування даних для покращення навчання
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# Розділяємо дані на тренувальний та тестовий набори
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2, random_state=42)

# Створюємо модель дерева рішень
model = DecisionTreeRegressor(random_state=42)

# Навчаємо модель
model.fit(X_train, y_train)

# Перевіряємо точність на тестових даних
accuracy = model.score(X_test, y_test)
print(f"Точність моделі: {accuracy:.2f}")

# Виводимо дерево рішень у текстовому вигляді
tree_rules = export_text(model, feature_names=['extruder_temp', 'bed_temp', 'print_speed', 'layer_height'])
print("\ndepends рішень:")
print(tree_rules)

# Прогнозування оптимальних параметрів для нових даних
def predict_print_quality(extruder_temp, bed_temp, print_speed, layer_height):
    input_data = np.array([[extruder_temp, bed_temp, print_speed, layer_height]])
    input_scaled = scaler.transform(input_data)
    predicted_quality = model.predict(input_scaled)
    return predicted_quality[0]

# Приклад використання: прогнозуємо якість друку для нових параметрів
new_extruder_temp = 220
new_bed_temp = 75
new_print_speed = 55
new_layer_height = 0.2
```

Рисунок 3.5. Блок коду з обробкою даних

Дерево рішень працює, аналізуючи вхідні параметри, такі як температура екструдера, температура столу, швидкість друку, товщина шару і навіть умови навколишнього середовища, і на основі цих даних робить прогнози щодо найкращих параметрів для поточної ситуації. Наприклад, якщо температура екструдера занадто висока або низька, дерево рішень може передбачити, що це вплине на якість друку, і скоригує параметри для забезпечення кращих результатів.

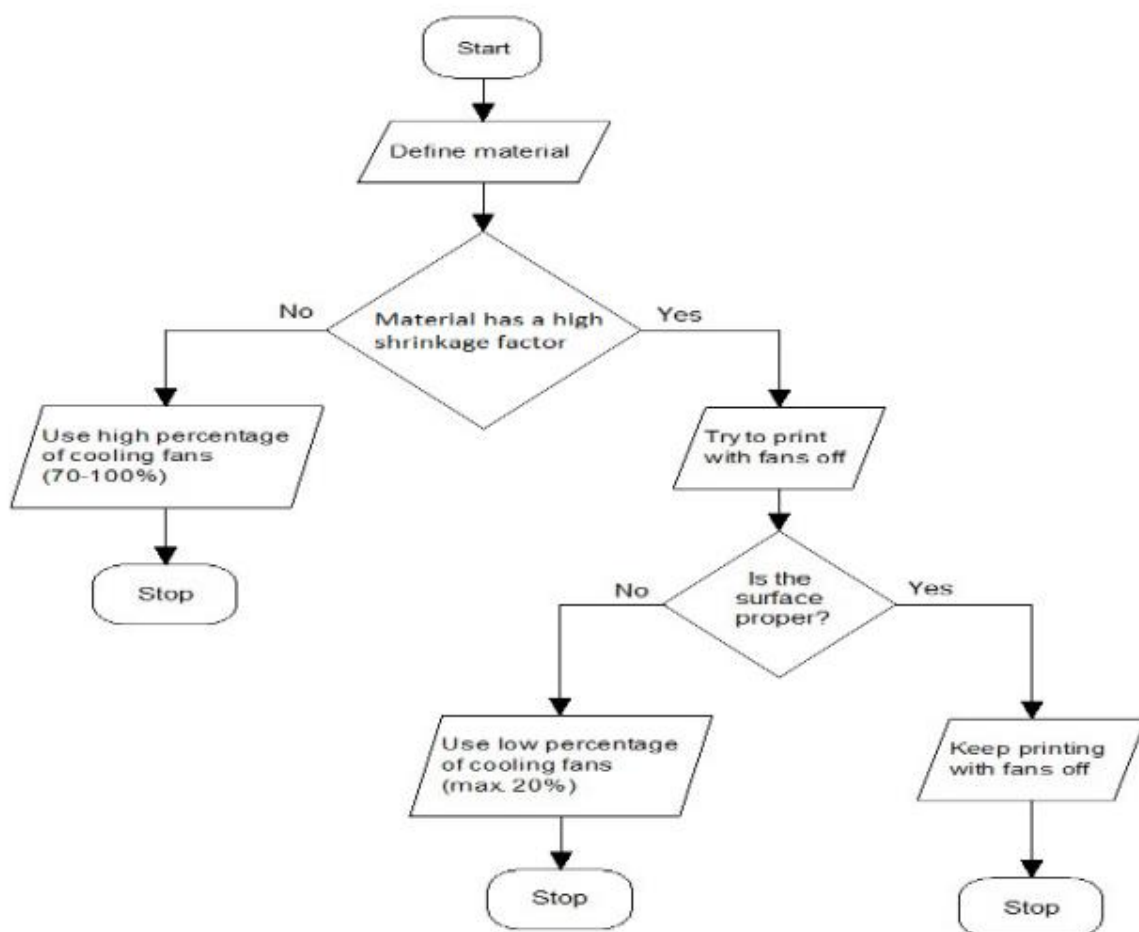


Рисунок 3.6. алгоритм вибору швидкості охолодження

Цей процес допомагає забезпечити високий рівень автоматизації, оскільки принтер може самостійно коригувати налаштування в реальному часі. Якщо, наприклад, датчики виявляють нестабільність температури або інші відхилення від оптимальних значень, дерево рішень приймає рішення щодо необхідних коригувань без втручання оператора. Це значно зменшує

ймовірність помилок через людський фактор і покращує точність і ефективність друку.

Дерево рішень також дозволяє оптимізувати процес друку, оскільки з часом, під час постійного збору даних і аналізу результатів, модель може бути переналаштована для кращого прогнозування. Наприклад, якщо модель спостерігає, що при певній температурі екструдера і швидкості друку якість друку часто погіршується, дерево рішень може визначити ці умови як ненадійні і порекомендувати зміни в налаштуваннях.

Цей підхід дозволяє значно зменшити час на налаштування принтера, підвищити його стабільність і знизити кількість дефектів, що виникають під час друку. Тому використання дерева рішень в процесі 3D-друку дозволяє не лише автоматизувати налаштування, але й забезпечити більш високий рівень контролю за якістю друку, що має вирішальне значення для досягнення оптимальних результатів на кожному етапі процесу.

3.4 API для подання моделі з іншими програмними засобами

Щоб створити API для Clipper та Orca Slicer, яке дозволить моделі бази знань керувати вибором параметрів слайсера та отримувати зворотний зв'язок, потрібно виконати кілька кроків. Ось короткий огляд процесу

Спершу потрібно визначити, які функції та ендпоінти будуть доступні в API.

Це можуть бути:

- Завантаження 3D-моделі
- Вибір параметрів слайсера
- Запуск процесу слайсингу
- Отримання статусу роботи
- Отримання результатів слайсингу
- Надсилання зворотного зв'язку від моделі

API може бути реалізований, на Python з використанням Flask або FastAPI.

Ці інструменти вибрані через їх здатність ефективно та надійно виконувати завдання, необхідні для створення та керування API для Clipper та Orca Slicer. FastAPI – це сучасний фреймворк для створення високопродуктивних API на Python, який забезпечує зручність використання та високу продуктивність. UploadFile дозволяє зручно обробляти файли, які завантажуються користувачами, що є важливим для завантаження 3D-моделей. Orca Slicer – це інструмент для перетворення 3D-моделі у формат, придатний для 3D-друку, що є ключовою функцією нашого API. Використання асинхронних функцій у FastAPI дозволяє паралельно обробляти запити, що підвищує ефективність та знижує затримки. JSON формат забезпечує зручність обміну даними між клієнтом та сервером, що є стандартом для веб-API. Ці інструменти разом забезпечують створення ефективного, продуктивного та зручного API, який здатен виконувати складні завдання слайсингу та керування параметрами зворотного зв'язку.

```

from fastapi import FastAPI, File, UploadFile
import orca_slicer
import clipper

app = FastAPI()

@app.post("/upload-model/")
async def upload_model(file: UploadFile = File(...)):
    with open(f"/path/to/models/{file.filename}", "wb") as f:
        f.write(file.file.read())
    return {"filename": file.filename}

@app.post("/set-params/")
async def set_params(params: dict):
    # Збереження параметрів для слайсингу
    return {"params_set": params}

@app.post("/start-slicing/")
async def start_slicing(params: dict):
    # Виклик функції слайсингу з параметрами
    sliced_file = orca_slicer.slice(params)
    return {"sliced_file": sliced_file}

@app.get("/get-status/")
async def get_status():
    # Отримання статусу процесу
    status = "running"
    return {"status": status}

@app.get("/get-results/")
async def get_results():
    # Повернення результатів слайсингу
    results = {"filename": "sliced_model.gcode"}
    return results

@app.post("/send-feedback/")
async def send_feedback(feedback: dict):
    # Отримання та обробка зворотного зв'язку
    return {"status": "feedback received"}

```

Рисунок 3.7. Код для створення API і отримання деяких параметрів

API дозволить завантажити 3D-модель, вибрати параметри слайсера, запустити процес слайсингу та отримати результат. Модель бази знань зможе контролювати параметри слайсера та надсилати зворотний зв'язок для оптимізації процесу. Інтеграція відбуватиметься через HTTP-запити між клієнтом і сервером. Додатково, принтер надсилатиме дані з датчиків, щоб забезпечити точне налаштування параметрів слайсингу та моніторинг процесу друку в режимі реального часу

Тому модель, отримавши параметри з слайсера, може коригувати їх відповідно до заданих експертом діапазонів та умов, що дозволяє оптимізувати друк для досягнення найкращих результатів.

Таким чином, модель є ключовим елементом, який інтегрує дані з різних джерел і здійснює динамічну корекцію параметрів, забезпечуючи оптимальну якість друку та адаптацію до змінюваних умов роботи принтера в реальному часі. Вона виступає як міст між фізичними датчиками та програмними налаштуваннями, здатна забезпечити більш точну та ефективну роботу 3D-принтера.

Висновки до третього розділу

У третьому розділі була розроблена програма з використанням експертної бази знань та деревовидних алгоритмів. Основна мета полягала у створенні інтелектуальної системи для оптимізації процесу 3D-друку. Вибір середовища розробки моделі бази знань на основі Experta був обґрунтований високою гнучкістю та широкими можливостями цього інструменту. Experta дозволяє ефективно управляти складними системами знань та забезпечує інтеграцію різноманітних алгоритмів, що робить його ідеальним вибором для нашого проекту.

Також було створено API для спілкування між моделлю та різними частинами процесу 3D-друку, такими як слайсер та прошивка Кліппер. Це забезпечило ефективну взаємодію між компонентами системи та дозволило оптимізувати параметри друку в режимі реального часу. API дозволив моделі бази знань автоматично налаштовувати параметри друку, забезпечуючи високоякісний результат та знижуючи потребу в ручному втручанні.

Крім того, програма має можливість коригувати параметри друку "на ходу", відповідно до заданих експертом значень. Це дозволяє оперативно реагувати на зміни в умовах друку та забезпечувати стабільно високий рівень якості кінцевих виробів.

Таким чином, розроблена програма на основі експертної бази знань та деревовидних алгоритмів у поєднанні з створенням ефективного API забезпечує значні переваги в процесі 3D-друку, підвищуючи його ефективність та якість. Подальші дослідження можуть бути спрямовані на розширення функціональних можливостей системи та її адаптацію до різних типів 3D-принтерів та матеріалів.

ПЕРЕЛІК ПОСИЛАНЬ

1. 3D PRINTING & ADDITIVE MANUFACTURING INTELLIGENCE: TCT Magazine . URL: www.tctmagazine.com/additive-manufacturing/university-of-sheffield-trials-3d-printed-unmanned-aircraft/)
2. Гречко О. М. Сучасні адитивні технології та 3D-друк. Огляд останніх досягнень в різних сферах людського життя // Вісник НТУ «ХПІ». Серія: Проблеми удосконалювання електричних машин і апаратів. Теорія і практика. – 2019. – №1. – С. 63-75. doi: 10.20998/2079-3944.2022.1.12.
3. Запровадження у навчальний матеріал шкільного курсу фізики поняття адитивних технологій. [Електронний ресурс]. Режим доступу: <https://phm.cuspu.edu.ua/ojs/index.php/NZ-PMFMTO/article/viewFile/1200/1179>
4. Пухальська, Г.В., & Порвін, І.Е. (2020). Лазерне маркування. Тези доповідей науковопрактичної конференції, Запоріжжя, 13–17 квітня 2020 р. –Запоріжжя: НУ «Запорізька політехніка», 2020. – С. 21-22.
5. Розвиток 3D-друку. [Електронний ресурс]. Режим доступу: <https://phm.cuspu.edu.ua/nauka/naukovo-populiarni-publikatsii/878-rozvytok-tekhnologii-3-d-druku.html>
6. 3D Printer Market Sales Will Exceed \$14.6 billion in 2019 [electronic resource]. — Access: <http://blogs.gartner.com/pete-basiliere/2015/09/29/3dprintermarket-sales-will-exceed-14-6-billion-in2019/>.
7. Андрощук Г. О. Адитивні технології: перспективи і проблеми 3D друку (І частина) / Г.О. Андрощук // Наука, технології, інновації. - 2017. - № 1. - С. 68-77.
8. Технологія друку [Електронний ресурс]. Режим доступу : URL:<http://ukrefs.com.ua/page,2,180744-Tehnologiya-pechat.html> – Назва з екрану.
9. Гречко О. М. Сучасні адитивні технології та 3D-друк. Огляд останніх

10. досягнень в різних сферах людського життя // Вісник НТУ «ХПІ». Серія: Проблеми удосконалювання електричних машин і апаратів. Теорія і практика. – 2019. – No1. – С. 63-75. doi: 10.20998/2079-3944.2022.1.12.
11. Андросчук Г.О. 3D-друк в епоху інноваційних технологій: проблеми регулювання / Г.О. Андросчук, Я.В. Копил // Інтелектуальна власність в Україні. —2016. — No 5. — С. 17–26.
12. 3D PRINTING & ADDITIVE MANUFACTURING INTELLIGENCE: TCT Magazine . URL: www.tctmagazine.com/additive-manufacturing/university-of-sheffieldtrials-3d-printedunmanned-aircraft/)
13. Використання адитивних технологій на сучасних підприємствах. Електронний ресурс]. Режим доступу: https://learn.ztu.edu.ua/pluginfile.php/9440/mod_resource/content/2/Лабораторна%20пробота%20No1.pdf
14. Методичні вказівки до курсового проекту з дисципліни «Інженерно-технічне забезпечення видавничо-поліграфічного виробництва. Модуль 2 –Управління проектами» для студентів освітньо-кваліфікаційного рівня «магістр» і«спеціаліст» для спеціальностей напрямку «Видавничо-поліграфічна справа» [Електронний ресурс] / НТУУ «КПІ» ; уклад. О. М.Величко, А. В. Шангін. –Електронні текстові дані (1 файл: 267 Кбайт). –Київ : НТУУ «КПІ», 2013. – 32 с. –Назва з екрана. — Ресурс доступу:<http://ela.kpi.ua/handle/123456789/2603>.
15. Danich, V., & Shevchenko, S. (2023). INFORMATION SPACE. REVIEW OF TRANSPORT ECONOMICS AND MANAGEMENT. <https://doi.org/10.15802/rtem2022/277626>.
16. Karakurt, I., & Lin, L. (2020). 3D printing technologies: techniques, materials, and post-processing. *Current opinion in chemical engineering*, 28, 134-143. <https://doi.org/10.1016/j.coche.2020.04.001>.

17. Simao, L., Carvalho, L., & Madeira, M. (2020). Intellectual structure of management innovation: bibliometric analysis. *Management Review Quarterly*, 71, 651-677. <https://doi.org/10.1007/S11301-020-00196-4>.
18. Alvino, F., Di Vaio, A., Hassan, R., & Palladino, R. (2020). Intellectual capital and sustainable development: a systematic literature review. *Journal of Intellectual Capital*, 22, 76-94. <https://doi.org/10.1108/jic-11-2019-0259>.
19. (2005). 5 – Expert Systems. , 367-377. <https://doi.org/10.1016/B978-012170960-0/50031-1>.
20. Klein, M. (2020). Expert Systems. *SAGE Research Methods Foundations*. [https://doi.org/10.1016/0950-7051\(90\)90067-r](https://doi.org/10.1016/0950-7051(90)90067-r).
21. ЛЕКЦІЯ 12 Представлення знань і вивід на знаннях Полтавської політехніки ст 1-16 URL: https://learn.ztu.edu.ua/pluginfile.php/273014/mod_resource/content/1/III_Л-12_Знання_1.pdf
22. Rayegani, F.; Onwubolu, G.C. Fused deposition modelling (FDM) process parameter prediction and optimization using group method for data handling (GMDH) and differential evolution (DE). *Int. J. Adv. Manuf. Technol.* 2014, 73, 509–519.
23. Cantrell, J.T.; Rohde, S.; Damiani, D.; Gurnani, R.; DiSandro, L.; Anton, J.; Young, A.; Jerez, A.; Steinbach, D.; Kroese, C. Experimental characterization of the mechanical properties of 3D-printed ABS and polycarbonate parts. *Rapid Prototyp. J.* 2017, 23, 811–824.
24. Kumar, G.P.; Regalla, S.P. Optimization of support material and build time in fused deposition modeling (FDM). *Appl. Mech. Mater.* 2012, 110–116, 2245–2251.
25. Montero, M.; Roundy, S.; Odell, D.; Ahn, S.-H.; Wright, P.K. Material characterization of fused deposition modeling (FDM) ABS by designed experiments. *Soc. Manuf. Eng.* 2001, 10, 1–21.

26. Chacón, J.; Caminero, M.; García-Plaza, E.; Núñez, P. Additive manufacturing of PLA structures using fused deposition modelling: Effect of process parameters on mechanical properties and their optimal selection. *Mater. Des.* 2017, 124, 143–157.
27. Mohamed, O.A.; Masood, S.H.; Bhowmik, J.L. Mathematical modeling and FDM process parameters optimization using response surface methodology based on Q-optimal design. *Appl. Math. Model.* 2016, 40, 10052–10073.
28. Byun, H.-S.; Lee, K.H. Determination of the optimal part orientation in layered manufacturing using a genetic algorithm. *Int. J. Prod. Res.* 2005, 43, 2709–2724.
29. Sood, A.K.; Ohdar, R.K.; Mahapatra, S.S. Experimental investigation and empirical modelling of FDM process for compressive strength improvement. *J. Adv. Res.* 2012, 3, 81–90.
30. Panda, S.K.; Padhee, S.; Anoop Kumar, S.; Mahapatra, S.S. Optimization of fused deposition modelling (FDM) process parameters using bacterial foraging technique. *Intell. Inf. Manag.* 2009, 1, 89–97.
31. Gurralla, P.K.; Regalla, S.P. Multi-objective optimisation of strength and volumetric shrinkage of FDM parts: a multi-objective optimization scheme is used to optimize the strength and volumetric shrinkage of FDM parts considering different process parameters. *Virtual Phys. Prototyp.* 2014, 9, 127–138.
32. Vasudevarao, B.; Natarajan, D.P.; Henderson, M.; Razdan, A. Sensitivity of RP surface finish to process parameter variation. In *Proceedings of the Solid Freeform Fabrication Proceedings*, Austin, TX, USA, 7–9 August 2000; pp. 251–258.
33. Zhang, Y.; Chou, K. A parametric study of part distortions in fused deposition modelling using three-dimensional finite element analysis. *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.* 2008, 222, 959–968.

34. Peng, A.; Xiao, X.; Yue, R. Process parameter optimization for fused deposition modeling using response surface methodology combined with fuzzy inference system. *Int. J. Adv. Manuf. Technol.* 2014, 73, 87–100.
35. Mohamed, O.A.; Masood, S.H.; Bhowmik, J.L.; Nikzad, M.; Azadmanjiri, J. Effect of process parameters on dynamic mechanical performance of fdm PC/ABS printed parts through design of experiment. *J. Mater. Eng. Perform.* 2016, 25, 2922–2935.
36. R A Burnashev et al 2019 *J. Phys.: Conf. Ser.* 1352 012008
37. CUPS Reference Manual, Version 6.0, NASA document JSC-25012, Houston, Texas, June 1993 22-50
38. Gary Riley Software Technology Branch NASA Johnson Space Center Mail Stop PT4 Houston, TX 77058
39. Інтернет ресурс спільноти розробників ПО URL : <https://github.com/lispm/FRL>
40. International Journal of Semantic Computing
41. Vol. 6, No. 1 (2012) 3!23°cWorld Scientific Publishing CompanyDOI: 10.1142/S1793351X12400016
42. M. McShane, S. Nirenburg and S. Beale, Meaning-Centric Language Processing, in preparation.
43. M. McShane, S. Nirenburg and S. Beale, An NLP lexicon as a largely language independent resource, *Machine Translation* 19(2) (2005) 139173.
44. S. Nirenburg and V. Raskin, *Ontological Semantics* (MIT Press, Cambridge, MA, 2004)
45. Практическое введение в технологию искусственного интеллекта и экспертных систем с иллюстрациями на Бэйсике / Р.Левин, Д.Дранг, Б.Эдельсон: Пер.с англ. – М.: Финансы и статистика, 1990 – 239 с.
46. Zakharkevich O.V., Pochuprin A.V.: Development of prototype of expert system for rapid change in production of women's outerwear, Easter– European

Journal of Enterprise Technologies 2/2(68), 2014, pp. 50– 55,
<https://doi.org/10.15587/1729-4061.2014.23327>.

47. Інтернет ресурс URL :<https://rapidapi.com/blog/best-python-api-frameworks/>

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ(ПРЕЗЕНТАЦІЯ)



Дипломна робота на тему:

«Система інтелектуалізації управління процесом 3D -друку на основі застосування бази знань



Виконав студент САДМ-61:
Білик Ю.М.
Науковий керівник:
Патракеєв І.М



Актуальність теми.

Розвиток технологій **адитивного виробництва** та **3D-друку** відіграє ключову роль у сучасній промисловості, медицині, будівництві та інших сферах. Завдяки своїй гнучкості, швидкості виробництва та можливості створювати складні геометричні форми, 3D -друк стає невід'ємною частиною виробничих процесів. Однак зростання вимог до якості продукції, оптимізації витрат на матеріали та підвищення ефективності процесів друку ставить перед інженерами та дослідниками нові виклики. ткової актуальності темі надає застосування **сучасних алгоритмічних підходів** та методів оптимізації, таких **метод "бази знань"**. Це дозволяє автоматизувати процес вибору комбінації різних параметрів моделі



Об'єктом дослідження є процес оптимізації параметрів друку 3D-моделей на платформі 3D-принтера

Ціль роботи дослідження методів автоматизації процесу оптимізації параметрів 3D-моделей для 3D-друку, з використанням баз знань

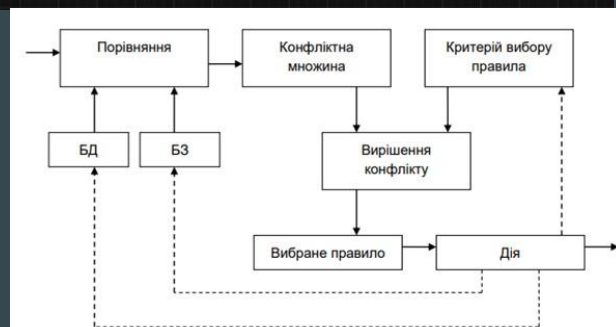
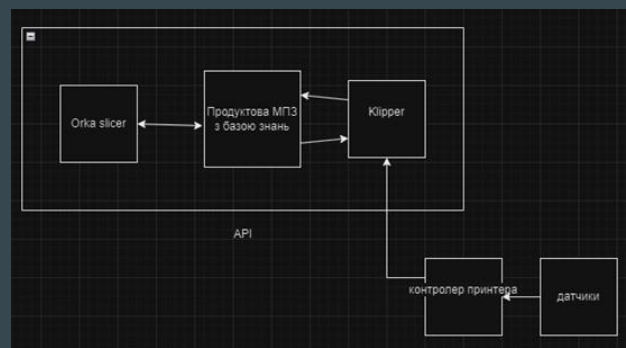
Методи дослідження В даній роботі використовувався метод порівняльного аналізу результатів відомих досліджень, та аналіз отриманих результатів

Результати дослідження та їх новизна полягає інтеграції засобів інтелектуалізації виробництва на основі баз знань в процес 3D-друку



Схема інтеграції 3D -друку

Машина виведення працює циклічно. В кожному циклі вона переглядає всі правила та існуючі на той момент факти. Факти з БД послідовно порівнюють з умовами правил для виявлення успішного порівняння. Сукупність вибраних правил складає конфліктну множину. Для вирішення конфлікту машина виведення має критерій, за яким вибирається лише одне правило. Факти, що утворюють висновок правила заносяться до БД або змінюють критерій вибору конфліктних правил. Якщо у висновку правила міститься кінцева дія, то вона виконується.





Переваги

- Можливість створення пресетів для різних цілей на основі знань експертів
- Оптимізація параметрів друку під час процесу друку в рамках заданих параметрів
- Запобігання помилкам пов'язаним з помилковими значеннями на етапі встановлення параметрів друку



Під час робіт були виконані такі завдання

Досліджено вплив параметрів на результат друку

Проаналізовано підходи для створення баз знань

Проаналізовано існуючі засоби створення моделей представлення даних

Запропоновано метод інтелектуалізації управління за допомогою поєднання бази знань та дерева рішень