# Рецензія на симуляцію

# Створення моделі системи керування предиктивного контролю технологічного процесу збагачення залізної руди

## 1. Вступ

Сучасні технологічні процеси збагачення залізної руди потребують ефективних та надійних систем керування, здатних враховувати складну динаміку процесу, наявність шумів, збурень та невизначеностей. У даному дослідженні розглядається проблема керування технологічними процесами збагачення із застосуванням методу модельного предиктивного керування (MPC - Model Predictive Control).

Вибір MPC як підходу до керування обґрунтовується його здатністю враховувати обмеження, прогнозувати поведінку системи та оптимізувати керуючі дії на певному часовому горизонті. Порівняно з іншими контролерами, MPC забезпечує більш гнучке та адаптивне керування, що особливо важливо для складних технологічних процесів збагачення руди. MPC є оптимальним вибором для процесів збагачення руди з кількох фундаментальних причин. По-перше, на відміну від класичних ПІД-регуляторів, MPC використовує модель процесу для прогнозування його поведінки на горизонті передбачення, що дозволяє передбачати наслідки керуючих дій. По-друге, здатність MPC явно враховувати технологічні обмеження (як на керуючі впливи, так і на змінні стану) є критично важливою для процесів збагачення, де перевищення обмежень може призвести до зупинки виробництва або пошкодження обладнання. По-третє, MPC здатний ефективно працювати з багатовимірними системами з перехресними зв'язками між змінними, що характерно для збагачувальних процесів, де зміна одного параметра часто впливає на багато інших показників.

Основною метою дослідження є створення достовірної моделі системи керування, яка забезпечує максимальну відповідність між контролером та реальним процесом збагачення руди, з урахуванням всіх особливостей та викликів, притаманних даному технологічному процесу.

## 2. Загальна архітектура системи

### 2.1. Структура та модулі системи

Система представлена як мережа взаємодіючих модулів, кожен з яких відповідає за певний аспект функціонування системи керування. Основні файли та їх взаємозв'язки включають:

• data\_gen.py - генератор даних

• model.py - прогнозна модель процесу

• objectives.py - формулювання цільових функцій

• mpc.py - реалізація контролера

• sim.py - симуляційний цикл

• utils.py - допоміжні функції

• dashboard.py - візуалізація результатів

Модульна архітектура системи базується на принципах об'єктно-орієнтованого проектування та концепції розділення відповідальності (Separation of Concerns). Такий підхід обрано з метою забезпечення гнучкості, масштабованості та тестованості системи. Кожен модуль має чітко визначений інтерфейс та відповідальність, що дозволяє змінювати реалізацію окремого модуля без впливу на інші компоненти. Це особливо важливо для дослідницьких систем, де потрібно експериментувати з різними алгоритмами та підходами. Модульність також спрощує інтеграцію нових функцій та вдосконалень, дозволяючи швидко адаптувати систему до змінних вимог.

Загальна архітектура програмного комплексу включає такі основні компоненти:

• **Генератор даних (DataGenerator)** забезпечує отримання достовірних часових рядів із урахуванням масового балансу, шумів і аномалій.

• **Прогнозна модель (KernelModel)** реалізує налаштоване ядрове прогнозування динаміки на основі лагованих ознак.

• **Контролер (MPCController)** працює як MPC-серцевина з буфером історії, оптимізаційним модулем на базі CVXPY.

• **Модулі цільових функцій (objectives)** формулюють різні стратегії керування залежно від технологічної задачі.

• **Симуляційний цикл (sim)** поєднує всі компоненти у замкнений контур, записуючи результати для подальшого аналізу та візуалізації.

• **Детектор аномалій** розміщено на вході симулятора, що працює як перед навчанням моделі, так і в процесі обробки поточних даних. Це забезпечує ідентифікацію та обробку нетипових значень у режимі реального часу.

Порядок виконання модулів та розділення відповідальності між ними забезпечує гнучкість системи та можливість її адаптації до різних умов експлуатації.

## 3. Генератор даних та детекція аномалій

Генератор даних (DataGenerator) є ключовим модулем системи, який забезпечує створення реалістичних даних для навчання та тестування моделей. Основні функції генератора даних включають:

• Завантаження історичних даних (reference\_df з Parquet)

• Формування синтетичних даних з урахуванням фізики процесу

• Моделювання шуму, масового балансу та додавання аномалій

• Застосування різних стратегій додаткових похибок

• Створення лагованого датасету для прогнозування динаміки

• Генерація конфігурації аномалій для robust-тестування

### 3.1. Інтеграція детектора аномалій

На вході симулятора розміщено детектор аномалій, що працює як перед навчанням моделі, так і в процесі обробки поточних даних. Розробка стратегій виявлення та фільтрації аномалій показала еволюцію від класичних евристичних онлайн-детекторів для типових аномалій сигналів (spike, drift, drop, freeze) до більш інтелектуальних підходів. Агресивна фільтрація дрейфу призводила до конфлікту з задачами EKF і збіднення даних тренду, на які повинен реагувати контролер.

Сучасний підхід включає:

• Зміна логіки дрейф-детектора: замість видалення будь-яких трендів, рекомендовано діяти лише як «запобіжник» на явно нехарактерні для реального процесу дрейфи

• Перехід від ручного online-детектування до використання EKF інновації для виявлення аномалій

• Інтеграцію детектора SignalAnomalyDetector/MultiSignalDetector у online режимі з різними логіками фільтра для offline/online обробки

Генерація синтетичних даних базується на теорії стохастичних процесів та принципах фізичного моделювання. Використання історичних даних як основи для синтетичних забезпечує збереження характерних патернів та залежностей реального процесу, тоді як додавання контрольованого шуму та аномалій дозволяє тестувати стійкість алгоритмів керування в різних умовах.

Моделювання масового балансу є принциповим рішенням, яке базується на фундаментальному законі збереження маси. Це гарантує фізичну узгодженість даних, що критично важливо для промислових процесів збагачення, де порушення балансу мас може вказувати на помилки вимірювань або втрати в системі.

Створення лагованих датасетів обґрунтовується теорією динамічних систем, де поточний стан залежить від попередніх станів та входів. Це дозволяє враховувати темпоральні залежності при прогнозуванні, що суттєво підвищує точність моделей для процесів з інерційністю та затримками, характерними для збагачувальних процесів.

Реалістичне моделювання даних включає:

• Реалізацію масового балансу для фізичної узгодженості показників виходу (потік мас і елементного складу)

• Підготовку lagged-датасетів — розширюючи кожен момент часу на L попередніх кроків для тренування й роботи динамічних моделей

Ця гнучкість дозволяє створювати повні датасети для тренування моделей, тестування робастності контролера, а також для цільового дослідження особливостей виявлення та реакції на рідкісні події.

Для покращення реалістичності симуляції рекомендується впровадження фізичних моделей та вдосконалення динаміки з новими параметрами, що забезпечить більш точну імітацію реального технологічного процесу.

### **3.2. Архітектурна роль StatefulPlantMixin**

Ключовим архітектурним рішенням у структурі генератора даних є впровадження StatefulPlantMixin - спеціалізованого міксину, який забезпечує фундаментальне розділення між генерацією статичних навчальних датасетів та динамічною покроковою симуляцією "живого" технологічного процесу.

**Концептуальне значення StatefulPlantMixin:**

StatefulPlantMixin реалізує парадигму staneful simulation, де модель процесу зберігає внутрішній стан між викликами та забезпечує каузальність обробки даних. Це критично важливо для реалістичного моделювання процесів збагачення руди, де поточний стан системи залежить від попередньої історії входів та внутрішніх змінних стану.

**Ключові функції міксину:**

**1. Управління внутрішнім станом:** Міксин підтримує буфери для лагованих змінних, історію входів/виходів та внутрішні параметри динаміки (інерційні ланки, транспортні затримки).

**2. Каузальна обробка:** На відміну від batch-обробки статичних даних, StatefulPlantMixin забезпечує, що інформація про майбутні стани недоступна для поточних обчислень, що відповідає реальним умовам керування.

**3. Інтерфейс для MPC-інтеграції:** Міксин надає стандартизований інтерфейс step(), який дозволяє MPC контролеру взаємодіяти з процесом покроково, отримуючи реалістичну відповідь системи на керуючі впливи.

**4. Реалізація plant-model mismatch:** Через незалежне управління станом та параметрами, міксин дозволяє створювати контрольований структурний розрив між моделлю контролера та симульованим процесом.

**Практичне значення для досліджень MPC:**

Впровадження StatefulPlantMixin дозволяє досліджувати поведінку MPC контролерів в умовах, максимально наближених до реальних: з невизначеностями моделі, затримками, інерційністю та неможливістю "підглянути" майбутні стани системи. Це забезпечує науково обґрунтовану оцінку робастності та ефективності різних MPC-стратегій.

Архітектурне рішення щодо використання міксину також забезпечує гнучкість системи: один і той же генератор може працювати як у режимі створення статичних датасетів для навчання моделей, так і у режимі динамічної симуляції для тестування контролерів, просто змінюючи режим роботи через відповідні методи інтерфейсу.

## 4. Ядерна модель процесу (KernelModel)

Ядерна модель процесу є центральним елементом системи, що забезпечує прогнозування динаміки технологічного процесу на основі зібраних даних. Реалізація моделі включає:

• Застосування гібридного підходу з використанням таких методів, як:

• Zero-First (Z-F) компонента

• Лінійна регресія

• Гаусівські процеси (Gaussian Process)

• Support Vector Regression (SVR)

• Процес навчання, організований через fit/predict інтерфейс з підтримкою багатовихідних сценаріїв

• Автоматичний вибір гіперпараметрів, включаючи gamma/length параметри

• Підтримку різних ядер (Kernel Ridge Regression, Gaussian Process Regression, SVR)

### 4.1. Розширення спектру ядерних методів

Система підтримує не лише Kernel Ridge Regression та Gaussian Process Regression, а також модель Support Vector Regression (SVR). Включення SVR у арсенал методів розширює можливості системи для роботи з різними типами нелінійностей та шумів у даних. SVR особливо ефективний для задач з обмеженою кількістю навчальних даних та в умовах, коли потрібна висока узагальнююча здатність моделі.

Вибір ядерних методів для моделювання процесу збагачення руди має міцне наукове підґрунтя. На відміну від параметричних моделей (як ARIMA або лінійні регресії), ядерні методи належать до непараметричних, що дозволяє їм ефективно апроксимувати складні нелінійні залежності без необхідності специфікації конкретної функціональної форми моделі. Це особливо важливо для процесів збагачення руди, де точний вигляд залежностей між параметрами часто невідомий.

Ядерні методи базуються на теорії репродуктивних гільбертових просторів (RKHS), що дозволяє трансформувати нелінійну задачу в простір вищої розмірності, де вона стає лінійною (це так званий "kernel trick"). Таким чином, ядерна функція неявно визначає перетворення простору ознак у простір вищої розмірності, дозволяючи знаходити нелінійні залежності між вхідними та вихідними параметрами.

Вибір конкретних ядерних методів обумовлений їх властивостями:

• **Kernel Ridge Regression (KRR)** забезпечує гнучке моделювання з регуляризацією, що запобігає перенавчанню, а також має аналітичний розв'язок, що гарантує швидке обчислення.

• **Gaussian Process Regression (GPR)** надає не лише прогноз, але й оцінку невизначеності, що критично важливо для прийняття рішень в умовах шуму та збурень. GPR забезпечує оптимальне байєсівське оцінювання, враховуючи апріорні знання про процес.

• **Support Vector Regression (SVR)** використовує концепцію опорних векторів та ε-insensitive loss function, що робить його особливо стійким до викидів та ефективним для моделювання систем з розрідженими даними.

Ядрова модель реалізує динамічне навчання за допомогою методу підтримуючого навчання (KRR, GPR або SVR), що дозволяє ефективно прогнозувати поведінку системи на основі історичних даних.

Для подальшого вдосконалення моделі рекомендується розглянути:

• Впровадження мультививідного GP (Gaussian Process)

• Покращення масштабування для обробки великих обсягів даних

• Інтеграцію інтерсепта для підвищення точності моделі

• Перехід до RandomForest для "plant" з метою створення штучного структурного розриву

## 5. Формулювання цільової функції

Цільова функція в MPC визначає бажану поведінку системи та критерії оптимізації керуючих впливів. Реалізовано такі підходи до формування цільової функції:

• Інтерфейс Control-Objective, що забезпечує гнучку структуру для різних цільових функцій

• MaxIronMassObjective: проста лінійна ціль

• MaxIronMassTrackingObjective: розширене відстеження цільових значень із квадратичною, інтегральною та згладжуючою складовими

Формулювання цільової функції у MPC базується на теорії оптимального керування та методах варіаційного числення. Класична структура включає квадратичну складову відхилення від завдання та квадратичну складову зміни керування (регуляризацію), що математично виражається як:

J = Σ(y - r)^T Q (y - r) + Σ(Δu)^T R (Δu)

де y - прогнозовані виходи, r - задані значення (референс), Δu - зміни керуючих впливів, Q та R - вагові матриці.

Вибір такої структури обґрунтований кількома міркуваннями:

1. Квадратична форма забезпечує диференційованість та гладкість цільової функції, що спрощує оптимізацію.

2. Баланс між якістю відстеження та плавністю керування досягається через відповідні вагові матриці Q та R.

3. Штраф за зміну керування (R-матриця) не лише забезпечує плавність, але й робить систему менш чутливою до шумів вимірювання.

Для процесів збагачення руди особливо важливим є врахування масового балансу, тому включення відповідних членів у цільову функцію забезпечує фізичну узгодженість керування.

Параметризація ваг забезпечує гнучкий вибір пріоритетів різних компонентів цільової функції, що дозволяє налаштовувати поведінку контролера відповідно до конкретних вимог технологічного процесу.

Два підходи до формування функцій вартості дозволяють оптимізувати управління в MPC, здатні модифікуватись залежно від умов:

• Маса мембрани: цільова функція балансування мас

• Відстеження траєкторії: функція з урахуванням квадратичної похибки та регуляризації

Для покращення цільових функцій можливі такі розширення:

• Уніфікація API

• Докладна документація

• Додавання адаптивних штрафів

• Розширення цілей для врахування більш складних критеріїв оптимізації

## 6. Реалізація MPC контролера

MPC контролер (MPCController) інтегрує навчену модель та цільову функцію для оптимізації керуючих впливів. Основні особливості реалізації контролера:

• Структура класу MPCController забезпечує сувору інтеграцію навченої моделі та цільової функції

• Особливості роботи з лагованими входами/виходами для врахування динамічних ефектів

• Буферизована історія стану (lagged-інтерфейс) для врахування динамічних ефектів

• Формулювання CVXPY-виразів для задачі оптимізації: змінні, функції вартості, обмеження на амплітуду та швидкість зміни (delta\_u)

Реалізація MPC контролера базується на теорії рекурентного керування та оптимізації з ковзним горизонтом. Фундаментальний принцип MPC полягає у розв'язанні задачі оптимізації на кожному кроці керування, використовуючи актуальний стан системи та прогнозну модель.

Використання CVXPY для формулювання та розв'язання оптимізаційної задачі обґрунтовується теорією опуклої оптимізації. Для лінійних та квадратичних цільових функцій з лінійними обмеженнями гарантується знаходження глобального оптимуму за поліноміальний час, що критично важливо для систем реального часу.

### 6.1. Архітектурна роль CVXPY у системі оптимізації

Вибір CVXPY як основного інструменту для формулювання оптимізаційних задач у MPC контролері має глибоке практичне та методологічне обґрунтування. CVXPY забезпечує декларативний підхід до опису задач оптимізації, дозволяючи розробнику сформулювати проблему у природній математичній формі, не заглиблюючись у деталі чисельної реалізації.

Ключові переваги CVXPY для MPC-застосувань:

**1. Декларативність формулювання:** Цільова функція та обмеження описуються у вигляді, максимально наближеному до математичної нотації, що підвищує читабельність коду та зменшує ймовірність помилок при імплементації складних MPC-формулювань.

**2. Автоматична трансляція у стандартну форму:** CVXPY автоматично перетворює задачу у стандартну форму квадратичного програмування (QP), що є оптимальною для MPC-задач з квадратичними цільовими функціями та лінійними обмеженнями.

**3. Інтеграція з ефективними солверами:** Система автоматично обирає найбільш підходящий солвер (у нашому випадку OSQP - Operator Splitting Quadratic Program), який спеціально оптимізований для задач керування з розрідженими матрицями та потребами реального часу.

**4. Гнучкість архітектури:** Зміна структури цільової функції або додавання нових обмежень потребує лише модифікації високорівневого опису, не зачіпаючи низькорівневі деталі оптимізації.

OSQP, який використовується за замовчуванням, особливо підходить для MPC через свою здатність ефективно обробляти розріджені матриці (характерні для MPC через блочну структуру горизонту прогнозування) та забезпечувати швидку збіжність для задач з теплим стартом (warm start), що критично важливо для систем реального часу.

Буферизація історії стану (lagged-інтерфейс) випливає з теорії динамічних систем, де поточна поведінка визначається не лише поточним станом, але й попередньою історією. Це особливо актуально для процесів з транспортними затримками та інерційністю, характерними для збагачувальних процесів. Математично це відповідає використанню ARX-моделей (AutoRegressive with eXogenous inputs) для опису динаміки системи.

Обмеження на амплітуду та швидкість зміни керуючих впливів (delta\_u) мають як практичне, так і наукове обґрунтування. З практичної точки зору, вони запобігають різким змінам, які можуть пошкодити виконавчі механізми. З теоретичної точки зору, такі обмеження підвищують робастність системи керування, обмежуючи вплив невизначеностей моделі та шумів вимірювання.

Векторизовані вирази дозволяють оптимізувати панораму керування на часовому горизонті, а регуляризація забезпечує плавність сигналу й відповідність фізичним обмеженням. Високий пріоритет надається модульності: при зміні типу моделі чи цільової функції весь ланцюг не потребує перебудови.

Для подальшого вдосконалення контролера рекомендується впровадження:

• Нелінійного MPC (NMPC) з розробкою лінеаризації для KRR, GPR і SVR

• Удосконаленої оцінки збурень для покращення точності керування

• Адаптивного визначення ваг і штрафів у цільовій функції

• Методів trust region (гомотопія) для забезпечення стабільності оптимізації

## 7. Удосконалення системи керування

### 7.1. Offset-free MPC з розширеним станом

Для забезпечення відсутності постійного відхилення від цільових значень (offset-free) пропонується впровадження MPC з розширеним станом, що включає оцінку сталого збурення:

• Додається оцінка сталого збурення (d), яка компенсує довготривалий bias у виходах без необхідності інтегратора у цільовій функції

• Рекомендації щодо побудови R- та Q-матриць із урахуванням реального рівня шуму

• Методи роботи з аномаліями (відсікання або Huber-корекція)

• Налаштування швидкості спостерігача для оптимальної роботи системи

Концепція Offset-free MPC має глибоке наукове підґрунтя в теорії систем керування. Традиційний MPC без додаткових механізмів не гарантує нульової усталеної похибки в присутності постійних збурень або невідповідностей моделі, що критично для процесів збагачення, де точність підтримання заданих параметрів напряму впливає на якість продукції.

Рішення полягає у розширенні вектора стану додатковими компонентами, що представляють оцінки постійних збурень. Це базується на принципі внутрішньої моделі (Internal Model Principle), який стверджує, що для ефективної компенсації збурення, модель цього збурення має бути частиною регулятора.

Математично це реалізується через розширену динамічну систему:

y(k) = [C C\_d] [x(k)]

де d - вектор оцінок постійних збурень.

Використання фільтра Калмана для оцінки розширеного стану забезпечує оптимальне стохастичне оцінювання в умовах шумів процесу та вимірювань. Налаштування матриць коваріації шумів (Q та R у контексті фільтра Калмана) базується на теорії оптимальної фільтрації та має відповідати реальним характеристикам шумів у системі.

### 7.2. Фільтрація та обробка сигналів

Для підвищення якості керування рекомендується впровадження методів фільтрації та обробки сигналів:

• Інтеграція Kalman фільтра для оцінки стану системи

• Використання різних стратегій фільтрації вхідних сигналів

• Згладжування керуючих сигналів для забезпечення їх плавності

• Детекція та обробка аномалій у вхідних сигналах

Використання фільтра Калмана для оцінки стану системи базується на теорії оптимальної стохастичної фільтрації. Фільтр Калмана є оптимальним оцінювачем у сенсі мінімізації середньоквадратичної похибки для лінійних систем з гауссівськими шумами. Для нелінійних систем, характерних для процесів збагачення, використовується розширений фільтр Калмана (EKF), який базується на лінеаризації системи навколо поточної оцінки стану.

Математично EKF реалізує два основних етапи:

1. Прогноз стану: x̂(k|k-1) = f(x̂(k-1|k-1), u(k-1))

2. Корекція прогнозу на основі вимірювань: x̂(k|k) = x̂(k|k-1) + K(k)[y(k) - h(x̂(k|k-1))]

де K(k) - коефіцієнт підсилення Калмана, що оптимально балансує між довірою до моделі та до вимірювань.

Робоча парадигма реалістичної симуляції технологічного процесу включає послідовність: EKF-передбачення → MPC-розрахунок → дія на plant → оновлення вимірювань → EKF update. Алгоритм онлайн-перенавчання моделі передбачає збирання даних в ковзному вікні та контроль якості за інноваціями EKF.

Вибір стратégій фільтрації вхідних сигналів (медіанна фільтрація, експоненційне згладжування, тощо) обґрунтовується їх здатністю зменшувати вплив шумів на якість керування без внесення значних фазових затримок, які могли б дестабілізувати систему.

Детекція та обробка аномалій базується на статистичних методах виявлення викидів, таких як правило трьох сигм для гауссівських процесів або робастні методи, що менш чутливі до екстремальних значень. Це відповідає задачі перевірки статистичних гіпотез, де нульова гіпотеза відповідає нормальній поведінці системи.

### 7.3. М'які обмеження в MPC

Реалізація м'яких обмежень (soft constraints) в MPC дозволяє системі адаптуватися до екстремальних умов та уникати проблем з нереалізовністю оптимізаційної задачі:

• Впровадження м'яких обмежень з адаптивними штрафами

• Балансування між дотриманням обмежень та досягненням цільових значень

• Гнучка структура обмежень, що враховує пріоритети та фізичні можливості системи

Концепція м'яких обмежень (soft constraints) в MPC має потужне наукове підґрунтя в теорії оптимізації. Традиційні "жорсткі" обмеження можуть призвести до нереалізовності оптимізаційної задачі у випадках, коли зовнішні збурення або невідповідності моделі роблять неможливим їх дотримання. Це особливо критично для промислових процесів, де навіть тимчасова втрата керування може мати серйозні наслідки.

М'які обмеження реалізуються через введення додаткових змінних відхилення (slack variables) ε та додаткових штрафних членів у цільову функцію:

J = J\_original + ρ·||ε||

Де ρ - ваговий коефіцієнт, що визначає "вартість" порушення обмежень.

Це перетворює задачу з жорсткими обмеженнями:

На задачу з м'якими обмеженнями:

Адаптивні штрафи, де ρ може змінюватися в залежності від величини порушення або важливості обмеження, базуються на концепції точного штрафу (exact penalty) в теорії оптимізації. При достатньо великих значеннях ρ, оптимальне рішення з м'якими обмеженнями буде відповідати рішенню з жорсткими обмеженнями, якщо останнє існує.

Пріоритезація обмежень (різні значення ρ для різних обмежень) дозволяє відображати реальні виробничі пріоритети, де порушення деяких обмежень може бути більш критичним, ніж інших. Це відповідає багатокритеріальній оптимізації з лексикографічним впорядкуванням критеріїв.

## 8. Оцінка результатів та аналіз

Для оцінки якості роботи системи керування використовуються такі методи та метрики:

• Стандартизовані методи перевірки якості керування (RMSE, MAE, інші)

• Аналіз роботи контролерів у різних умовах та сценаріях

• Оцінка стійкості системи до збурень та аномалій

• Аналіз впливу шуму і зашумленості вхідних/вихідних сигналів

• Оцінка наявності стійкої похибки при різних підходах до керування

Вибір метрик для оцінки якості керування базується на теорії оцінювання систем та статистичному аналізі. RMSE (Root Mean Square Error) та MAE (Mean Absolute Error) є класичними метриками, що оцінюють середню величину помилки, але з різним ставленням до великих відхилень: RMSE більш чутливий до них через квадратичну природу.

Для забезпечення високої прозорості та ефективності симуляцій впроваджено детальну систему оцінювання стану EKF (розширений фільтр Калмана). Система дає змогу аналізувати RMSE, NEES, NIS, а також покриття довірчих інтервалів та надавати графіки діагностик. Це підвищує об'єктивність аналізу та дозволяє швидко виявляти як неправдиві покращення, так і небажані ефекти.

Для оцінки стохастичних властивостей системи використовуються NEES (Normalized Estimation Error Squared) та NIS (Normalized Innovation Squared), які дозволяють перевірити, чи є оцінка стану консистентною (чи коректно оцінюється коваріація помилки). Якщо система правильно змодельована, ці метрики повинні мати χ² розподіл з відповідною кількістю ступенів свободи.

Аналіз роботи контролерів у різних умовах (nominal case, worst case, різні типи збурень) базується на концепції робастності системи керування, що визначається як здатність системи зберігати стабільність та певний рівень продуктивності при наявності невизначеностей у моделі чи зовнішніх збуреннях.

Порівняння GPR та KRR на графіках дає глибший зміст: GPR забезпечує плавність і стратегічність (врахування довгострокового тренду), тоді як KRR — швидку, можливо надмірно реактивну відповідь на збурення та випадкові флуктуації. Перевага GPR у стабільному, «економному» керуванні (менше різких дій), у реальних умовах це зменшує знос обладнання та економить енергоресурси.

Аналіз поведінки керування для різних ядер та моделей (Kernel Ridge Regression, Gaussian Process Regression, SVR) дозволяє визначити оптимальні параметри системи та шляхи її подальшого вдосконалення. Особлива увага приділяється впливу агресивності моделі на стратегію керування та рекомендаціям щодо коригування target function у MPC за допомогою штрафів.

Дослідження стійкої похибки при різних підходах до керування спирається на теорему про кінцеве значення (Final Value Theorem) з теорії систем, яка дозволяє аналізувати усталену поведінку системи при типових вхідних сигналах.

## 9. Напрямки подальшого розвитку

Для подальшого розвитку системи керування рекомендується зосередитися на таких напрямках:

• Адаптація до зовнішніх збурень (perturbations) з використанням сучасних методів оцінки та компенсації

• Впровадження нових підходів у моніторингу і діагностиці для раннього виявлення аномалій

• Інтеграція багатовихідних моделей для комплексного керування процесом

• Розробка адаптивних стратегій для підтримання стабільності в розгорнутій системі

• Покращення швидкодії перенавчання моделей та обробки великих буферів даних

• Інтеграція сучасних методів адаптивної (онлайн) ідентифікації

• Впровадження Moving Horizon Estimation (MHE) для кращої оцінки невимірюваних станів

Адаптація до зовнішніх збурень спирається на теорію робастного та адаптивного керування. Робастне керування забезпечує стабільність при заданому діапазоні невизначеностей, тоді як адаптивне - дозволяє системі підлаштовуватися до змінних умов. Це реалізується через H∞-оптимізацію для робастного керування або через методи онлайн-ідентифікації для адаптивного.

Обґрунтування повільності GPR-моделей пов'язане з аналізом складності навчання. Рекомендації включають зменшення n\_restarts, зменшення розміру вікна, використання апроксимацій через Nyström, Ridge та потенційний перехід до GPflow/GPTorch.

Проблеми числової стабільності та шляхи уникнення патологічної лінеаризації GPR включають додавання стабілізуючих компонентів ядра, кліпінг градієнтів, періоди охолодження при перенавчанні. Лінеаризація складної моделі для QP потребує компромісу між точністю та швидкістю.

Впровадження Moving Horizon Estimation (MHE) базується на тих самих принципах оптимізації з ковзним горизонтом, що й MPC, але для задачі оцінки стану. На відміну від EKF, MHE може безпосередньо враховувати обмеження на стани та має кращі властивості збіжності для нелінійних систем.

Онлайн-ідентифікація та адаптивне навчання базуються на теорії машинного навчання для потокових даних (online learning), де модель постійно адаптується до нових даних без повного перенавчання. Це особливо важливо для промислових процесів, які можуть змінюватися з часом (наприклад, через зношення обладнання або зміну характеристик сировини).

Інтеграція багатовихідних моделей відповідає концепції багатовимірного керування (MIMO - Multi-Input Multi-Output), яке обґрунтоване в теорії систем керування та дозволяє враховувати взаємозв'язки між різними вихідними параметрами процесу.

## 10. Висновки

Розроблена система демонструє сучасний підхід до кінцево-коригованого MPC-застосування для технологічних ланцюгів збагачення залізної руди. Пріоритет зміщується до гнучкої, адаптивної й стійкої до збурень архітектури:

• Генератор даних моделює реальний світ із похибками та аномаліями

• Ядрова модель відтворює часову динаміку процесу з підтримкою KRR, GPR та SVR

• Контролер оптимізації враховує історію й обмеження

• Спостерігачі (offset-free augmentation) забезпечують нульовий steady-state error навіть в умовах структурної невизначеності

• Детектор аномалій забезпечує надійну роботу системи в умовах нештатних ситуацій

Розроблена система керування процесом збагачення залізної руди демонструє успішну інтеграцію класичних принципів теорії керування з сучасними методами машинного навчання та оптимізації. Ключова наукова цінність дослідження полягає у створенні гібридної системи, де статистичні методи (ядерні моделі) поєднуються з детермінованими (MPC), забезпечуючи як здатність навчатися з даних, так і гарантії щодо стабільності та продуктивності.

Симулятор був перетворений із "академічного" інструменту у більш науково-обґрунтований тест-б