

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЗАХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЗАХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Кваліфікаційна наукова  
праця на правах рукопису

**ЗАБЧУК ВАДИМ ВОЛОДИМИРОВИЧ**

УДК 004.942:519.6

**ДИСЕРТАЦІЯ**

МАТЕМАТИЧНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ МОДЕЛЮВАННЯ  
ПРОЦЕСІВ У БІОГАЗОВИХ УСТАНОВКАХ НА ОСНОВІ АНАЛІЗУ  
ІНТЕРВАЛЬНИХ ДАНИХ

Спеціальність 121 – Інженерія програмного забезпечення

Галузь знань 12 – Інформаційні технології

Подається на здобуття ступеня доктора філософії.

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

  
\_\_\_\_\_ В.В. Забчук

Науковий керівник: Пукас Андрій Васильович, доктор технічних наук, професор

Тернопіль – 2024

## АНОТАЦІЯ

*Забчук В. В.* Математичне та програмне забезпечення для моделювання процесів у біогазових установках на основі аналізу інтервальних даних. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття ступеня доктора філософії за спеціальністю 121 «Інженерія програмного забезпечення» – Західноукраїнський національний університет, Тернопіль, 2024.

Підготовка здійснювалась на кафедрі комп'ютерних наук Західноукраїнського національного університету Міністерства освіти і науки України.

Дисертаційна робота присвячена розв'язуванню актуального науково-технічного завдання забезпечення допустимих значень кислотності багатокомпонентного субстрату за рахунок розробки математичного та програмного забезпечення для моделювання динаміки кислотності субстрату у біогазових установках на основі аналізу інтервальних даних.

Виробництво біогазу у спосіб переробки органічних відходів життєдіяльності людини та відходів тваринництва й агропромислового комплексу є одним із пріоритетних напрямків розвитку альтернативної (зеленої) енергетики. Разом з тим, підвищення ефективності функціонування цього виробництва вимагає розробки нових підходів управління процесами в біогазових установках на основі застосування інформаційних технологій та методів математичного моделювання. Проблематика досліджень, наведених у праці, стосується застосування методів математичного моделювання з метою відображення взаємозв'язку між основною характеристикою процесу та чинниками, які на неї впливають, так і динаміки основної характеристики процесу, яка визначається кислотністю субстрату у біореакторі.

Для побудови математичних моделей обох видів запропоновано використати методи параметричної та структурної ідентифікації моделей статичних об'єктів та дискретних моделей динаміки об'єктів на основі аналізу інтервальних даних.

Перший тип моделей необхідний для опису взаємозв'язків між характеристикою процесу та чинниками, які на неї впливають, а другий – для відображення динаміки характеристики за допомогою різницевих та диференціальних рівнянь. Такі математичні моделі, хоча і характеризуються «гарантованими» прогностичними властивостями, проте вимагають застосування для їх ідентифікації складних оптимізаційних процедур з багатоекстремальними функціями мети та часто ці функції представлені в дискретному вигляді. Тому для розв'язування цих задач запропоновано використати цілісний універсальний підхід (однаковий для обох типів математичних моделей), який ґрунтується на метаевристичних алгоритмах оптимізації, які, своєю чергою, використовують механізми самоорганізації та самоадаптації в процесі пошуку оптимального чи квазіоптимального розв'язку. Серед метаевристичних алгоритмів обрано алгоритм, який симулює ройовий алгоритм штучної бджолоїної колонії.

Тому у дисертаційній роботі запропоновано математичне та програмне забезпечення для моделювання динаміки кислотності субстрату у біогазових установках на основі аналізу інтервальних даних з метою забезпечення допустимих значень кислотності багатокомпонентного субстрату.

У першому розділі розглядаються підходи до моделювання процесів у біогазових установках з точки зору функціонування енергетичної системи, а також з точки зору їх оптимізації. Відомі моделі, які описують процес виробництва біогазу, демонструють різний рівень деталізації. Моделювання процесів у біогазових установках може надати динамічну інформацію про процес анаеробного зброджування, наприклад, для прогнозування виходу біогазу або оптимізації процесу анаеробного зброджування.

Обґрунтовано використання сучасних інформаційних технологій як ефективних інструментів управління процесами у біогазових установках. Розглядаються сильні та слабкі сторони інтеграції детальних і глибоко досліджених моделей анаеробного процесу. Запропоновано використання математичного моделювання для відображення зв'язку між головною характеристикою процесу та

факторами, що на неї впливають, а також динаміки цієї характеристики, яка залежить від кислотності субстрату у біореакторі. Для створення таких математичних моделей запропоновано та обґрунтовано використання методів параметричної ідентифікації моделей статичних об'єктів та дискретних моделей динаміки об'єктів на основі аналізу інтервальних даних.

У завершальній частині розділу здійснено постановку задачі дослідження та деталізовано основні завдання.

У другому розділі дисертаційного дослідження розроблено математичне забезпечення для моделювання процесів у біогазових установках, розглянуто ідентифікацію математичних моделей для відображення функціонування біогазових установок при виробництві біогазу.

Для відображення взаємозв'язку між основною характеристикою процесу (показником кислотності субстрату) та чинниками, які на неї впливають, запропоновано використати нелінійне алгебричне рівняння, а для відображення динаміки основної характеристики процесу, яка визначається кислотністю субстрату у біореакторі, – запропоновано використати різницеве рівняння. Також це різницеве рівняння перетворене до диференціального рівняння.

Для розробки обох математичних моделей запропоновано та обґрунтовано використати методи аналізу інтервальних даних та метод параметричної ідентифікації цих моделей. Такі математичні моделі хоча і характеризуються «гарантованими» прогностичними властивостями, проте вимагають застосування для їх ідентифікації складних оптимізаційних процедур з багатоекстремальними функціями мети та часто ці функції представлені в дискретному вигляді. Тому, для розв'язування цих задач запропоновано використати цілісний універсальний підхід (однаковий для обох типів математичних моделей), який ґрунтується на метаевристичних алгоритмах оптимізації, які, своєю чергою, використовують механізми самоорганізації та самоадаптації в процесі пошуку оптимального чи квазіоптимального розв'язку. Серед метаевристичних алгоритмів запропоновано та обґрунтовано алгоритм, який симулює ройовий алгоритм штучної бджолоїної колонії.

Для пошуку структури моделі динаміки кислотності субстрату у біореакторі запропоновано двоетапний алгоритм. Спочатку, за допомогою більш простішої задачі ідентифікації моделі взаємозв'язку між основною характеристикою процесу та чинниками, які на неї впливають, отримано нелінійне алгебричне рівняння. Потім, на основі його аналізу, обрано необхідні структурні елементи для моделі динаміки кислотності субстрату у біореакторі.

Також для інтерпретації фізичних властивостей динаміки кислотності субстрату у біореакторі, запропоновано та обґрунтовано, на підставі отриманого різницевого рівняння, здійснити перехід до диференціального, в якому коефіцієнти мають фізичну інтерпретацію, на відміну від різницевого рівняння. В основі запропонованого методу використано формули розкладу значень шуканої функції у вузлах різницевої сітки в ряд Тейлора в околі базового вузла, а також різницевого представлення похідних в цьому ж околі. Також при цьому переході враховано представлення управляючих функцій.

У третьому розділі дисертаційної роботи розроблено відкриту, сервісно-орієнтовану архітектуру програмного забезпечення для моделювання процесів у біогазових установках, яка, на відміну від існуючих, інтегрує програмні компоненти: інтерпретації інтервальних дискретних моделей; ідентифікації цих моделей; розв'язування диференціальних рівнянь, що у сукупності уможливорює розробку програмних систем та комп'ютерних середовищ для управління процесами в біогазових установках.

Наведено основні аспекти реалізації програмного забезпечення для моделювання процесів у біогазових установках, особливістю якого є програмна інтерпретація інтервальних дискретних моделей та відкрита сервісно-орієнтована архітектура, що забезпечує їх інтеграцію в існуючі системи управління біогазовими установками та дозволяє в реальному часі управляти біохімічними процесами і тим самим підвищити ефективність функціонування біогазових установок. Приведено ряд діаграм, які ілюструють особливості програмної реалізації середовища для моделювання процесів у біогазових установках. У кінці розділу описано

особливості реалізації підсистеми зберігання та аналізу інформації.

У четвертому розділі дисертаційної роботи наведено вигляд графічного інтерфейсу. Описано особливості програмної реалізації системи, зокрема з використанням парадигми об'єктно-орієнтованого програмування на базі відкритої сервісно-орієнтованої архітектури з використанням технології Spring Framework на мові програмування Java, а також інтерпретатора Python. В якості Python інтерпретатора було обрано Jython, а в якості СУБД використовується MySQL.

Описано процеси інтеграційного тестування та тестування продуктивності середовища для моделювання процесів у біогазових установках. Здійснено опис тестування продуктивності системи та, відповідно, підтверджено ефективність використання середовища при моделюванні процесів у біогазових установках. Показано можливість подальшого розвитку проекту та його імплементації в діючі біогазові установки.

У кінці розділу здійснено опис графічного інтерфейсу програмного середовища для моделювання процесів у біогазових установках. Описано функціонування основних підсистем з точки зору взаємодії між собою, різними типами користувачів, а також можливість зовнішньої інтеграції в діючі біогазові установки.

Практичне значення отриманих результатів полягає у створенні програмного середовища для моделювання процесів у біогазових установках.

Ключові слова: програмне середовище, архітектура програмного забезпечення, математичне моделювання, методи аналізу інтервальних даних, інтервальні моделі, параметрична ідентифікація, алгоритм штучної бджолиної колонії, біоенергетика, біогазова установка, анаеробне зброджування.

## **ПЕРЕЛІК ОПУБЛІКОВАНИХ ПРАЦЬ ЗА ТЕМОЮ ДИСЕРТАЦІЇ**

*Наукові праці, в яких опубліковано основні наукові результати дисертації:*

1. Дивак М., Забчук В. Моделювання характеристик процесів у біогазових установках на основі аналізу інтервальних даних. *Вісник Хмельницького національного університету. Серія: Технічні науки.* 2024. Том 331. № 1. С.180-190.

(1,1 д.а. / 0,8 д.а.; особистий внесок: здобувачем запропоновано та обґрунтовано двоетапний метод ідентифікації математичної моделі динаміки показника кислотності субстрату у біореакторі на основі аналізу інтервальних даних).

DOI: <https://doi.org/10.31891/2307-5732-2024-331-28>

2. Дивак Т., Забчук В. Огляд і аналіз методів та засобів моделювання процесів у біогазових установках. *Наукові праці Донецького національного технічного університету. Серія: "Інформатика, кібернетика та обчислювальна техніка"*. 2024. №1 (38). С.57-70. (1,3 д.а. / 0,5 д.а.; особистий внесок: здобувачем здійснено порівняльний аналіз методів та засобів моделювання процесів у біогазових установках, виділено їх основні переваги та недоліки).

DOI: <https://doi.org/10.31474/1996-1588-2024-1-38-57-70>

3. Манжула В. І., Забчук В.В. Архітектура програмного забезпечення для моделювання процесів у біогазових установках. *Наукові праці Вінницького національного технічного університету*. 2024. Вип. 1. С. 1-10. (0,8 д.а. / 0,3 д.а.; особистий внесок: здобувачем реалізовано архітектуру програмного забезпечення для моделювання процесів у біогазових установках).

DOI: <https://doi.org/10.31649/2307-5376-2024-1-19-28>.

4. Manzhula V., Dyvak M. and Zabchuk V. The Improved Method for Identifying Parameters of Interval Nonlinear Models of Static Systems. *International Journal of Computing*. 2024. 23 (1). P. 19-25. (0,7 д.а. / 0,2 д.а.; особистий внесок: здобувачем отримано математичну модель динаміки показника кислотності субстрату у біореакторі у вигляді диференціального рівняння).

DOI: <https://doi.org/10.47839/ijc.23.1.3431>. (Scopus, Q4)

*Наукові праці, які засвідчують апробацію матеріалів дисертації:*

5. Забчук В. В. Аналіз методів побудови системи доповненої реальності для тестування практичних навиків. *Комп'ютерні інформаційні технології: матеріали школи-семінару молодих вчених і студентів СІТ'2019 (м. Тернопіль, 29 листопада 2019 р.)*. Тернопіль: ТНЕУ, 2019. С. 9. (0,1 д.а.)

6. Красносельська А.О., Веселов О.В., Забчук В.В. Інтелектуалізована система для оптимізації процесу формування документації програмного забезпечення. Комп'ютерні інформаційні технології: матеріали школи-семінару молодих вчених і студентів СІТ'2023. (Тернопіль, 30 листоп. 2023 р.). Тернопіль: ЗУНУ, 2023. С. 100-101. (0,1 д.а. / 0,03 д.а.; особистий внесок: здобувачем запропоновано компонентну структуру інтелектуалізованої системи для створення документації програмного забезпечення).

7. Dyvak M., Pukas A., Manzhula V., Kasatkina N., Komar M. and Zabchuk V. The Task of Parametric Identification the Interval Models with Nonlinear Parameters. *2022 12th International Conference on Advanced Computer Information Technologies (ACIT)*. (Spisska Kapitula, Slovakia, 26-28 Sept. 2022). P. 106-111. (0,6 д.а. / 0,1 д.а.; особистий внесок: здобувачем вдосконалено метод параметричної ідентифікації інтервальних моделей з нелінійними параметрами).

DOI: <https://doi.org/10.1109/ACIT54803.2022.9913166> (Scopus).

8. Pukas A., Simak A., Goncharuk-Cholach T., Konoplitska O., Zabchuk V. Intelligent Analyzing Module in the Academic Staff Performance Appraisal System. *CEUR-WS*. 2023. Vol. 3624. P. 423-429. (0,6 д.а. / 0,1 д.а.; особистий внесок: здобувачем запропоновано алгоритм інтеграції програмних модулів для реалізації методів інтелектуального аналізу).

[https://ceur-ws.org/Vol-3624/Short\\_2.pdf](https://ceur-ws.org/Vol-3624/Short_2.pdf) (Scopus).

9. Pukas A., Smal V., Zabchuk V. Software based on blockchain technology for consolidation the medical data about the patients examination. *CEUR-WS*. 2018. Vol. 2300. P. 170-174. (0,4 д.а. / 0,1 д.а.; особистий внесок: здобувачем описано процедуру використання технології блокчейн для консолідації даних).

<https://ceur-ws.org/Vol-2300/Paper41.pdf> (Scopus).



## ANNOTATION

*Zabchuk V. V.* Methods and software tools for modeling processes in biogas plants based on interval data analysis. – Scientific work on the rights of the manuscript.

Thesis for the degree of Doctor of Philosophy in the specialty 121 "Software Engineering" - West Ukrainian National University, Ternopil, 2024.

The research was carried out at the Department of Computer Science of the West Ukrainian National University of the Ministry of Education and Science of Ukraine.

The thesis is devoted to solving of the actual scientific and technical task of ensuring the tolerance values of acidity in a multicomponent substrate by developing mathematical tools and software for modeling the dynamics of substrate acidity in biogas plants based on the analysis of interval data.

Biogas production by way of processing of organic waste of human life and animal husbandry and agro-industrial complex is one of the priority areas of development of alternative (green) energy. At the same time, increasing the efficiency of the operation of this production requires the development of new approaches to process management in biogas plants based on the application of information technologies and mathematical modeling methods. The problems of the research presented in the thesis, relate to the application of mathematical modeling methods in order to reflect the relationship between the main characteristic of the process and the factors that affect it, as well as the dynamics of the main characteristic of the process, which is determined by the acidity of the substrate in the bioreactor.

To build mathematical models of both types, it is proposed to use the methods of parametric and structural identification of models of static objects and discrete models of dynamic objects based on the interval data analysis. The first type of models is needed to describe the relationships between the characteristics of the process and the factors affecting it, and the second is to show the dynamics of the characteristics using difference and differential equations. Although such mathematical models are characterized by "guaranteed" prognostic properties, they require the use of complex optimization

procedures with multi-extremal objective functions for their identification, and often these functions are presented in a discrete form. Therefore, to solve these problems, it is proposed to use an universal approach (the same for both types of mathematical models), which is based on metaheuristic optimization algorithms, which, in turn, use self-organization and self-adaptation mechanisms in the process of searching for an optimal or quasi-optimal solution. Among the metaheuristic algorithms, an algorithm that simulates the swarm algorithm of an artificial bee colony was chosen.

Therefore, the mathematical tools and software for modeling the dynamics of the acidity of the substrate in biogas plants based on the analysis of interval data in order to ensure the acceptable values of the acidity of the multicomponent substrate are proposed in this thesis.

In the first chapter, approaches to modeling are considered from the point of view of the energy system functioning, as well as from the point of view the processes optimization in biogas plants. Known models that describe the biogas production process show different levels of detail. Modeling processes in biogas plants can provide dynamic information about the anaerobic digestion process, for example, to predict biogas output or optimize the anaerobic digestion process.

The use of modern information technologies as effective process management tools in biogas plants is substantiated. The strengths and weaknesses of the integration of detailed and deeply researched models of the anaerobic process are considered, the use of mathematical modeling is proposed to describe the relationship between the main characteristic of the process and the factors affecting it, as well as the dynamics of this characteristic, which depends on the acidity of the substrate in the bioreactor. To create such mathematical models, the use of methods of parametric identification of models of static objects and discrete models of dynamic objects based on the analysis of interval data is proposed and substantiated.

In the final part of the chapter, the research problem is stated and the main tasks are detailed.

In the second chapter of the thesis the mathematical tools for modeling processes

in biogas plants was developed, the identification of mathematical models for displaying the functioning of biogas plants during biogas production was considered.

To display the relationship between the main characteristic of the process (substrate acidity indicator) and the factors affecting it, it is proposed to use a nonlinear algebraic equation, and to display the dynamics of the main characteristic of the process, which is determined by the acidity of the substrate in the bioreactor, it is proposed to use a difference equation. Also, this difference equation is transformed into a differential equation.

For the development of both mathematical models, it is proposed and substantiated to use the methods of interval data analysis and the method of parametric identification of these models. Although such mathematical models are characterized by "guaranteed" prognostic properties, they require the use of complex optimization procedures with multi-extremal objective functions for their identification, and often these functions are presented in a discrete form. Therefore, to solve these problems, it is proposed to use an universal approach (the same for both types of mathematical models), which is based on metaheuristic optimization algorithms, which, in turn, use self-organization and self-adaptation mechanisms in the process of finding an optimal or quasi-optimal solution. Among metaheuristic algorithms, an algorithm that simulates the swarm algorithm of an artificial bee colony is proposed and substantiated.

A two-stage algorithm is proposed to find the structure of the substrate acidity dynamics model in the bioreactor. First, with the help of a simpler problem of identifying the model of the relationship between the main characteristic of the process and the factors affecting it, a nonlinear algebraic equation was obtained. Then, based on its analysis, the necessary structural elements were selected for the model of the dynamics of substrate acidity in the bioreactor.

Also, in order to interpret the physical properties of the dynamics of the substrate acidity in the bioreactor, it is proposed and substantiated on the basis of the obtained difference equation to make a transition to the differential equation, in which the coefficients have a physical interpretation, unlike the difference equation. The proposed

method is based on formulas for the distribution of the values of the searching function at the nodes of the difference grid into a Taylor series in the neighborhood of the base node, as well as the difference representation of the derivatives in the same neighborhood. Also, during this transition, the presentation of control functions is taken into account.

In the third chapter of the thesis an open service-oriented software architecture for modeling processes in biogas plants is developed, which, unlike the existing ones, integrates software components: interpretation of interval discrete models; identification of these models; solving differential equations, which collectively enables the development of software systems and computer environments for managing the processes in biogas plants.

The main aspects of implementation the software for modeling processes in biogas plants are presented, the feature of which is the software interpretation of interval discrete models and an open service-oriented architecture, which ensures their integration into existing control systems of biogas plants and allows to control biochemical processes in real time and thereby increase efficiency of functioning of biogas plants. A number of diagrams illustrating the features of the software implementation of the environment for modeling processes in biogas plants are presented. At the end of the chapter, the features of the implementation of the information storage and analysis subsystem are described.

The fourth chapter of the dissertation gives an user interface. The features of the software implementation of the system are described, in particular, using the paradigm of object-oriented programming based on an open service-oriented architecture using the Spring Framework technology in the Java programming language, as well as the Python interpreter. Jython was chosen as the Python interpreter, and MySQL was used as the DBMS.

The processes of integration testing and environmental performance testing for modeling processes in biogas plants are described. The system performance testing was described and, accordingly, the efficiency of the environment using in the simulation of

processes in biogas plants was confirmed. The possibility of further development of the project and its implementation in operating biogas plants is shown.

At the end of the chapter, the graphical interface of the software environment for modeling processes in biogas plants was tested. The functioning of the main subsystems is described from the point of view of interaction between themselves, different types of users, as well as the possibility of external integration into operating biogas plants.

The practical significance of the obtained results lies in the creation of a software environment for modeling processes in biogas plants.

Keywords: software environment, software architecture, mathematical modeling, interval data analysis methods, interval models, parametric identification, artificial bee colony algorithm, bioenergy, biogas plant, anaerobic digestion.

## **LIST OF PUBLISHED PAPERS BY THE TOPIC OF THESIS**

### SCIENTIFIC PAPERS IN WHICH THE MAIN SCIENTIFIC RESULTS OF THE DISSERTATION WERE PUBLISHED:

1. Dyvak M., Zabchuk V. Modeling the characteristics of process in biogas plants based on interval data analysis. *Herald of Khmelnytskyi National University. Technical sciences*. 2024. Vol. 331. No. 1. P.180-190. (1.1 p.s. / 0.8 p.s.; personal contribution: a two-stage method of identification of the mathematical model of the dynamics of the substrate acidity indicator in the bioreactor based on the analysis of interval data was substantiated by the applicant).

DOI: <https://doi.org/10.31891/2307-5732-2024-331-28>

2. Dyvak T., Zabchuk V. Review and analysis of methods and tools for modeling processes in biogas plants. *Scientific works of the Donetsk National Technical University. Series: "Informatics, cybernetics and computer technology"*. 2024. No. 1 (38). P.57-70. (1.3 p.s. / 0.5 p.s.; personal contribution: the applicant made a comparative analysis of methods and means of modeling processes in biogas plants, their main advantages and disadvantages were selected).

DOI: <https://doi.org/10.31474/1996-1588-2024-1-38-57-70>

3. Manzhula V. I., Zabchuk V.V. Software architecture for modeling processes in biogas plants. *Scientific works of Vinnytsia National Technical University*. 2024. Issue 1. P. 1-10. (0.8 p.s. / 0.3 p.s.; personal contribution: the applicant implemented the software architecture for modeling processes in biogas plants).

DOI: <https://doi.org/10.31649/2307-5376-2024-1-19-28>.

4. Manzhula V., Dyvak M., Zabchuk V. Improved method of parameter identification of interval nonlinear models of static systems. *International Journal of Computing*. 2024. 23 (1). P. 19-25. (0.7 p.s. / p.s.; personal contribution: the applicant obtained a mathematical model of the dynamics of the indicator of substrate acidity in the bioreactor in the form of a differential equation).

DOI: <https://doi.org/10.47839/ijc.23.1.3431>. (Scopus, Q4)

#### SCIENTIFIC WORKS CERTIFYING THE APPROVAL OF THE DISSERTATION MATERIALS:

5. Zabchuk V. B. Analysis of methods of building an augmented reality system for testing practical skills. *Computer information technologies: materials of the school-seminar of young scientists and students of CIT'2019* (Ternopil, November 29, 2019). Ternopil: TNEU, 2019. P. 9. (0.1 p.s.).

6. Krasnoselska A.O., Veselov O.V., Zabchuk V.V. An intellectualized system for optimizing the process of creating software documentation. *Computer information technologies: materials of the school-seminar of young scientists and students CIT'2023*. (Ternopil, November 30, 2023). Ternopil: WUNU, 2023. P. 100-101. (0.1 p.s. / 0.03 p.s.; personal contribution: the applicant proposed a component structure of an intellectualized system for creating software documentation).

7. Dyvak M., Pukas A., Manzhula V., Kasatkina N., Komar M. and Zabchuk V. The problem of parametric identification of interval models with nonlinear parameters. *2022 12th International Conference on Advanced Computing Information Technologies*

(*ACIT*). (Spisska Kapitula, Slovakia, 26-28 Sept. 2022). P. 106-111. (0.6 p.s. / 0.1 p.s.; personal contribution: the applicant is proposed the method of parametric identification of interval models with nonlinear parameters).

DOI: <https://doi.org/10.1109/ACIT54803.2022.9913166> (Scopus).

8. Pukas A., Simak A., Honcharuk-Cholach T., Konoplytska O., Zabchuk V. Module of intellectual analysis in the system of evaluation of teachers' performance. *CEUR-WS*. 2023. Issue 3624. P. 423-429. (0.6 p.s. / 0.1 p.s.; personal contribution: the applicant offer an algorithm for integrating software modules for the implementation of intellectual analysis methods).

[https://ceur-ws.org/Vol-3624/Short\\_2.pdf](https://ceur-ws.org/Vol-3624/Short_2.pdf) (Scopus).

9. Pukas A., Smal V., Zabchuk V. Software based on blockchain technology for consolidation of medical data on patient examinations. *CEUR-WS*. 2018. Issue 2300. P. 170-174. (0.4 p.s. / 0.1 p.s.; personal contribution: the applicant described the procedure for using blockchain technology for data consolidation).

<https://ceur-ws.org/Vol-2300/Paper41.pdf> (Scopus).

## ЗМІСТ

АНОТАЦІЯ .....	2
ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	18
ВСТУП.....	19
РОЗДІЛ 1 АНАЛІЗ МЕТОДІВ ТА ПРОГРАМНИХ ЗАСОБІВ ДЛЯ МОДЕЛЮВАННЯ ПРОЦЕСІВ У БІОГАЗОВИХ УСТАНОВКАХ .....	28
1.1 Процеси анаеробного мікробіологічного бродіння в біогазових установках...	29
1.2 Аналіз засобів моніторингу та контролю процесів у біогазових установках...	33
1.3 Аналіз методів моделювання процесів у біогазових установках.....	37
1.4 Постановка задачі дослідження .....	51
Висновки до розділу 1 .....	53
РОЗДІЛ 2 МОДЕЛЮВАННЯ ХАРАКТЕРИСТИК ПРОЦЕСІВ У БІОГАЗОВИХ УСТАНОВКАХ НА ОСНОВІ АНАЛІЗУ ІНТЕРВАЛЬНИХ ДАНИХ .....	55
2.1 Постановка задач параметричної ідентифікації моделей характеристик процесів у біогазових установках.....	56
2.2 Обчислювальна схема методу параметричної ідентифікації моделей характеристик процесів у біогазових установках .....	63
2.3 Математична модель взаємозв'язку між показником рН ферментаційного середовища та чинниками, які на нього впливають .....	68
2.4 Математична модель динаміки показника кислотності рН ферментаційного середовища.....	72
Висновки до розділу 2 .....	79
РОЗДІЛ 3 АРХІТЕКТУРА ПРОГРАМНОЇ СИСТЕМИ ДЛЯ МОДЕЛЮВАННЯ ПРОЦЕСІВ У БІОГАЗОВИХ УСТАНОВКАХ .....	81
3.1 Обґрунтування використання інформаційних технології для забезпечення	



допустимих значень кислотності багатокomпонентного субстрату у біогазових установках .....	81
3.2 Архітектура програмної системи для моделювання процесів у біогазових установках .....	93
3.3 Підсистема аналізу та зберігання даних .....	107
Висновки до розділу 3 .....	119
РОЗДІЛ 4 ПРОГРАМНЕ СЕРЕДОВИЩЕ ДЛЯ МОДЕЛЮВАННЯ ПРОЦЕСІВ У БІОГАЗОВИХ УСТАНОВКАХ .....	120
4.1 Особливості програмної реалізації середовища для моделювання процесів у біогазових установках на основі відкритої сервісно-орієнтованої архітектури...	120
4.2 Тестування програмного середовища .....	127
4.3 Організація графічного інтерфейсу програмного середовища для моделювання процесів у біогазових установках .....	138
Висновки до розділу 4 .....	145
ВИСНОВКИ.....	147
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	150
ДОДАТОК А DDL БАЗИ ДАНИХ ДЛЯ МОДЕЛЮВАННЯ ПРОЦЕСІВ У БІОГАЗОВИХ УСТАНОВКАХ .....	164
ДОДАТОК Б ЛІСТИНГ КОДУ ОСНОВНИХ МОДУЛІВ ПРОГРАМИ .....	171
ДОДАТОК В ЛІСТИНГ КОДУ РЕАЛІЗАЦІЇ МАТЕМАТИЧНИХ МОДЕЛЕЙ..	177
ДОДАТОК Г СПИСОК ПУБЛІКАЦІЙ ЗА ТЕМОЮ ДИСЕРТАЦІЇ .....	183
ДОДАТОК Д АКТИ ВПРОВАДЖЕНЬ РЕЗУЛЬТАТІВ ДИСЕРТАЦІЙНОГО ДОСЛІДЖЕННЯ .....	185

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ABC	Artificial Bee Colony Algorithm
API	Application Programming Interface
DDL	Data Definition Language
DSS	Decision Support System
MVC	Model View Controller
АБК	алгоритм бджолоїної колонії
БГУ	біогазова установка
БД	база даних
ДДМ	дискретна динамічна модель
ІРР	інтервальне різницеve рівняння
ІДДМ	інтервальна дискретна динамічна модель
ІСНАР	інтервальна система нелінійних алгебричних рівнянь
ІМ	інтервальна модель
ПМБК	поведінкова модель бджолоїної колонії
ПЗ	програмне забезпечення
СУБД	система управління базами даних

## ВСТУП

**Актуальність теми дослідження.** Зростання усвідомлення наслідків забруднення довкілля та потреба в альтернативних джерелах енергії призвело до збільшення інтересу до використання біогазу як екологічно чистого палива. У зв'язку з постійним зменшенням запасів традиційних джерел енергії, таких як нафта та вугілля, біогаз стає привабливим джерелом альтернативної енергії [23, 24, 122, 125].

Ефективне моделювання процесів у біогазових установках допомагає розробити ефективні стратегії виробництва, що забезпечують стабільну поставку енергії, а також дозволяє оптимізувати процеси виробництва біогазу з метою мінімізації викидів шкідливих речовин та максимізації виходу енергії. Математичне моделювання забезпечує ефективну альтернативу для звичайних методів моніторингу процесів анаеробного бродіння у БГУ. Оскільки інструменти математичного моделювання широко використовуються в інших галузях промисловості, їх можна використовувати для проектування та оптимізації комплексних процесів у біогазових установках. Крім того, майбутні біогазові установки в Україні будуть працювати більш гнучко, ніж сьогодні, щодо обслуговування та виробництва біогазу, що потребуватиме покращеного моніторингу [23, 28, 112, 122, 125]. Інтеграція програмних систем із підсистемами математичного моделювання в моніторинг процесів в БГУ може сприяти ефективному прогностичному обслуговуванню, яке забезпечить гнучкість процесу виробництва біогазу.

Виробництво біогазу у спосіб переробки органічних відходів життєдіяльності людини та відходів тваринництва і агропромислового комплексу є одним із пріоритетних напрямків розвитку альтернативної (зеленої) енергетики. Разом з тим, підвищення ефективності функціонування цього виробництва вимагає розробки нових підходів управління процесами в біогазових установках на основі застосування інформаційних технологій та методів математичного моделювання.

У біогазових установках відбуваються складні процеси на різних етапах анаеробного бродіння. Відомі моделі анаеробного мікробіологічного бродіння, які були описані в роботах J. Monod, Н. Е.О. Powell, A.G. Hashimoto, R. Mitsdörffer, базуються на складних біохімічних перетвореннях та мають детермінований характер.

Альтернативним підходом до моделювання процесів у біогазових установках є використання дискретних динамічних моделей, які будуються на основі спостережень за характеристиками процесів. У роботах Matt E. Katherine D., Roderick I. Maskie, Н.Б. Голуб, описані математичні моделі, що враховують стохастичність процесу. Проте ці моделі є чутливими до зміни протікання виробництва біогазу на будь-якій стадії [119].

Проблематика досліджень, наведених у роботі, стосується застосування методів математичного моделювання з метою відображення взаємозв'язку між основною характеристикою процесу та чинниками, які на неї впливають, так і динаміки основної характеристики процесу, яка визначається кислотністю субстрату у біореакторі. Для побудови математичних моделей обох видів запропоновано використати методи параметричної та структурної ідентифікації моделей статичних об'єктів та дискретних моделей динаміки об'єктів на основі аналізу інтервальних даних (Дивак М.П., Стахів П.Г., Пукас А.В.) Перший тип моделей – необхідний для опису взаємозв'язків між характеристикою процесу та чинниками, які на неї впливають, а другий – для відображення динаміки характеристики за допомогою різницевого та диференціального рівнянь. Такі математичні моделі, хоча і характеризуються «гарантованими» прогностичними властивостями, проте вимагають застосування для їх ідентифікації складних оптимізаційних процедур з багатоекстремальними функціями мети та часто ці функції представлені в дискретному вигляді.

Тому для розв'язування цих задач запропоновано використати цілісний універсальний підхід (однаковий для обох типів математичних моделей), який ґрунтується на метаевристичних алгоритмах оптимізації, які, своєю чергою,

використовують механізми самоорганізації та самоадаптації в процесі пошуку оптимального чи квазіоптимального розв'язку. Серед метаевристичних алгоритмів обрано алгоритм, який симулює ройовий алгоритм штучної бджолоїної колонії. Проте при його реалізації для обох класів вищезазначених моделей виникають певні труднощі, які є предметом дослідження та вирішення у цій дисертаційній роботі. Саме тому є актуальною розробка нових підходів управління процесами в біогазових установках на основі застосування інформаційних технологій та методів математичного моделювання.

**Мета і завдання дослідження.** Забезпечення допустимих значень кислотності багатокомпонентного субстрату за рахунок розробки математичного та програмного забезпечення для моделювання динаміки кислотності субстрату у біогазових установках на основі аналізу інтервальних даних.

Для досягнення цієї мети необхідно виконати такі завдання:

- провести аналіз відомих методів та засобів для моделювання процесів у біогазових установках, виділити їх основні переваги та недоліки;
- розробити двоетапний метод ідентифікації математичної моделі динаміки показника кислотності субстрату у біореакторі на основі аналізу інтервальних даних;
- розробити математичну модель динаміки показника кислотності субстрату у біореакторі;
- вдосконалити сервісно-орієнтовану архітектуру програмного забезпечення для моделювання процесів у біогазових установках;
- розробити комп'ютерне середовище для моделювання процесів у біогазових установках з сервісно-орієнтованою архітектурою;
- провести апробацію розроблених методів та моделей, розробленого середовища для управління процесом налаштування біогазової установки на структуру сировини та темпів її подачі у біореактор.

*Об'єкт дослідження* – процеси анаеробного мікробіологічного бродіння у біогазових установках.

*Предмет дослідження* – математичне та програмне забезпечення для моделювання динаміки кислотності субстрату у біогазових установках на основі аналізу інтервальних даних.

**Методи дослідження.** Для розв'язання задач моделювання процесів у біогазових установках на основі інтервального підходу використано методи системного аналізу, теорії ідентифікації, математичного моделювання, інтервальної арифметики, методи оптимізації; для розв'язання задач реалізації програмного комплексу для моделювання процесів у біогазових установках – об'єктно-орієнтований підхід, функціональне та комплексне тестування.

**Наукова новизна отриманих результатів.** У межах дисертаційної роботи *вперше*:

- запропоновано та обґрунтовано двоетапний метод ідентифікації математичної моделі динаміки показника кислотності субстрату у біореакторі на основі аналізу інтервальних даних, який, на відміну від відомих, ґрунтується на розв'язуванні спочатку простішої задачі ідентифікації рівняння взаємозв'язку між основною характеристикою процесу та чинниками, які на неї впливають, і на цій основі, на другому етапі – ідентифікації математичної моделі динаміки цього показника у біореакторі, що у сукупності забезпечило спрощення побудови математичної моделі з гарантованими прогностичними властивостями;

- отримано математичну модель динаміки показника кислотності субстрату у біореакторі у вигляді диференціального рівняння, яке, на відміну від існуючих, налаштовано на параметри процесу для конкретної біогазової установки на підставі перетворення інтервальних різницевих рівнянь, що забезпечує гарантовані прогностичні властивості побудованої математичної моделі і уможливлює забезпечення допустимих значень кислотності багатокомпонентного субстрату в біогазових установках на основі управління подачею сировини;

*набули подальшого розвитку:*

- відкрита сервісно-орієнтована архітектура програмного забезпечення для моделювання процесів у біогазових установках, яка, на відміну від відомих

інтегрує програмні компоненти: інтерпретації інтервальних дискретних моделей; ідентифікації цих моделей; розв'язування диференціальних рівнянь, що у сукупності уможлиблює розробку програмних систем та комп'ютерних середовищ для управління процесами в біогазових установках.

- комп'ютерні середовища для моделювання процесів у біогазових установках, які, на відміну від існуючих, ґрунтуються на програмній інтерпретації інтервальних дискретних моделей та відкритій сервісно-орієнтованій архітектурі, що уможлиблює їх інтеграцію в існуючі системи управління біогазовими установками з метою забезпечення допустимих значень кислотності багатокomпонентного субстрату.

**Особистий внесок здобувача.** Усі результати, викладені в дисертаційній роботі, отримані автором самостійно. У друкованих працях, опублікованих у співавторстві, автору належать такі результати:

- [65] вдосконалено метод параметричної ідентифікації інтервальних моделей з нелінійними параметрами;
- [71] отримано математичну модель динаміки показника кислотності субстрату у біореакторі у вигляді диференціального рівняння;
- [85] запропоновано алгоритм інтеграції програмних модулів для реалізації методів інтелектуального аналізу;
- [86] описано процедуру використання технології блокчейн для консолідації даних;
- [116] запропоновано та обґрунтовано двоетапний метод ідентифікації математичної моделі динаміки показника кислотності субстрату у біореакторі на основі аналізу інтервальних даних;
- [119] здійснено порівняльний аналіз методів та засобів моделювання процесів у біогазових установках, виділено їх основні переваги та недоліки;
- [123] запропоновано компонентну структуру інтелектуалізованої системи для створення документації програмного забезпечення;

- [124] реалізовано архітектуру програмного забезпечення для моделювання процесів у біогазових установках.

Основні положення та результати дисертаційної роботи в наведених працях викладені в повному обсязі.

**Апробація результатів дисертації.** Основні положення і результати дисертаційної роботи презентовано на 5 конференціях, зокрема:

- 8th International Conference on “Advanced Computer Information Technologies” ACIT’2018 – Ceske Budejovice, Czech Republic – 2018;
- 12th International Conference on Advanced Computer Information Technologies, ACIT-2022, Spisska Kapitula, Slovakia, 26-28 Sept, 2022.
- X International Conference “Information Technology and Implementation”, IT&I (November 20 – November 21, 2023, Kyiv);
- Школа-семінар молодих вчених і студентів «Комп’ютерні інформаційні технології» СІТ’2019, Тернопіль;
- Школа-семінар молодих вчених і студентів СІТ’2023, Тернопіль, 30 листопада 2023 р.

**Публікації.** За результатами дисертаційних досліджень опубліковано 9 наукових праць (Додаток Г) загальним обсягом 63 сторінки, зокрема 4 статті у фахових наукових виданнях [2, 5, 6, 9], 1 з яких входить до міжнародної наукометричної бази Scopus та Web of Science та відповідно до класифікації SCImago Journal and Country Rank віднесено до квартилю Q4 [2], 5 публікацій у матеріалах конференцій [1, 3, 4, 7, 8], 3 з яких входять до міжнародної наукометричної бази Scopus [1, 3, 4].

**Структура та обсяг роботи.** Дисертаційна робота складається зі вступу, чотирьох розділів, висновків, списку використаних джерел із 125 найменувань та 5 додатків. Загальний обсяг роботи складає 188 сторінок друкарського тексту, з них 148 сторінок основного тексту. Робота містить 61 рисунок і 9 таблиць.



### **Зв'язок роботи з науковими програмами, планами, темами.**

Дисертаційна робота виконувалася в рамках пріоритетного напрямку «Інформаційні та комунікаційні технології» відповідно до закону України «Про пріоритетні напрями розвитку науки і техніки» та пов'язана з науковими дослідженнями, які виконувалися за держбюджетною тематикою кафедри комп'ютерних наук Західноукраїнського національного університету. Основні результати дисертаційного дослідження отримано в межах виконання таких тем:

- держбюджетного прикладного дослідження на тему: «Математичне та програмне забезпечення прототипу біогазової установки з підвищеною ефективністю функціонування» (державний реєстраційний номер 0124U000076), у якому автором запропоновано двоетапний метод ідентифікації математичної моделі динаміки показника кислотності субстрату у біореакторі на основі аналізу інтервальних даних, який, на відміну від відомих, ґрунтується на розв'язуванні спочатку простішої задачі ідентифікації рівняння взаємозв'язку між основною характеристикою процесу та чинниками, які на неї впливають, і на цій основі, на другому етапі – ідентифікації математичної моделі динаміки цього показника у біореакторі, що у сукупності забезпечило спрощення побудови математичної моделі з гарантованими прогностичними властивостями;

- держбюджетного дослідження на тему: «Математичне та комп'ютерне моделювання об'єктів з розподіленими параметрами на основі поєднання онтологічного та інтервального аналізу» (державний реєстраційний номер 0122U001497), у якому автором розроблено комп'ютерне середовище для моделювання процесів у біогазових установках, яке, на відміну від існуючих, ґрунтується на програмній інтерпретації інтервальних дискретних моделей та відкритій сервісно-орієнтованій архітектурі, що уможлиблює їх інтеграцію в існуючі системи управління біогазовими установками з метою забезпечення допустимих значень кислотності багатокомпонентного субстрату;

- госпдоговірної теми «Моделювання динаміки процесів у біогазових установках» (державний реєстраційний номер 0123U103785, замовник –

ТОВ «Захід Трейд Тернопіль»), у якій автором програмно інтерпретовано математичну модель динаміки показника кислотності субстрату у біореакторі у вигляді диференціального рівняння, яке, на відміну від існуючих, налаштовано на параметри процесу для конкретної біогазової установки на підставі перетворення інтервальних різницевих рівнянь, що забезпечує гарантовані прогностичні властивості побудованої математичної моделі і уможлиблює забезпечення допустимих значень кислотності багатокомпонентного субстрату в біогазових установках на основі управління подачею сировини.

Усі вищезгадані роботи виконувалися за безпосередньої участі автора.

**Практичне значення отриманих результатів** полягає у створенні програмної системи для моделювання процесів у біогазових установках. Архітектура програмного забезпечення включає необхідні модулі для введення початкових умов моделювання, виконання обчислень, візуалізації результатів та можливості внесення коректив у математичні моделі.

Теоретичні та прикладні результати дисертаційної роботи використано:

- у ТОВ «Теофіпольська енергетична компанія» для автоматичного налаштування параметрів біогазової установки в реальному часі на будь-якому етапі технологічного процесу виготовлення біогазу без залучення експертного середовища (акт про впровадження результатів дисертаційної роботи від 20 травня 2024 р.);
- в компанії «Eleks» (м. Тернопіль) в процесі реалізації програмно-технічних комплексів, які включають інтеграцію програмно-реалізованих засобів математичного моделювання (акт про впровадження результатів дисертаційної роботи від 2 травня 2024 р.);
- в НДЧ ЗУНУ при виконанні держбюджетних та госпдоговірних досліджень на тему:
  - «Математичне та програмне забезпечення прототипу біогазової установки з підвищеною ефективністю функціонування» (державний реєстраційний номер 0124U000076);

- «Математичне та комп'ютерне моделювання об'єктів з розподіленими параметрами на основі поєднання онтологічного та інтервального аналізу» (державний реєстраційний номер 0122U001497);

- «Моделювання динаміки процесів у біогазових установках» (державний реєстраційний номер 0123U103785, замовник – ТОВ «Захід Трейд Тернопіль») (акт про використання результатів дисертаційної роботи від 14 травня 2024 р.);

- в освітньому процесі Західноукраїнського національного університету на кафедрі комп'ютерних наук при викладанні дисциплін: «Конструювання програмного забезпечення», «Засоби програмування баз даних та знань», «Архітектура та проектування програмного забезпечення (акт про впровадження в освітній процес від 23 квітня 2024 р.)

У додатку Д до дисертації подано акти використання результатів дисертаційного дослідження.

## РОЗДІЛ 1

### АНАЛІЗ МЕТОДІВ ТА ПРОГРАМНИХ ЗАСОБІВ ДЛЯ МОДЕЛЮВАННЯ ПРОЦЕСІВ У БІОГАЗОВИХ УСТАНОВКАХ

Виробництво біогазу у спосіб переробки органічних відходів життєдіяльності людини та відходів тваринництва і агропромислового комплексу є одним із пріоритетних напрямків розвитку альтернативної (зеленої) енергетики. Разом з тим, підвищення ефективності функціонування цього виробництва вимагає розробки нових підходів управління процесами в біогазових установках на основі застосування інформаційних технологій та методів математичного моделювання.

У розділі розглядаються підходи до моделювання з точки зору функціонування енергетичної системи, а також з точки зору оптимізації процесів у біогазових установках. Відомі моделі, які описують процес виробництва біогазу демонструють різний рівень деталізації. Моделювання процесів у біогазових установках може надати динамічну інформацію про процес анаеробного зброджування, наприклад, для прогнозування виходу біогазу або оптимізації процесу анаеробного зброджування.

У розділі обґрунтовано використання сучасних інформаційних технологій як ефективних інструментів управління процесами у біогазових установках. Розглядаються сильні та слабкі сторони інтеграції детальних і глибоко досліджених моделей анаеробного процесу, запропоновано використання математичного моделювання для відображення зв'язку між головною характеристикою процесу та факторами, що на неї впливають, а також динамікою цієї характеристики, яка залежить від кислотності субстрату у біореакторі. Для створення таких математичних моделей запропоновано та обґрунтовано використання методів параметричної ідентифікації моделей статичних об'єктів та дискретних моделей динаміки об'єктів на основі аналізу інтервальних даних.

На основі проведеного аналізу предметної області дослідження здійснено постановку задачі дослідження та деталізовано основні завдання. Основні результати цього розділу опубліковані в працях [65, 71, 116, 119, 124].

## 1.1 Процеси анаеробного мікробіологічного бродіння в біогазових установках

В теперішній час споживання енергії в світі постійно зростає, і це також пов'язано з високим рівнем шкоди навколишньому середовищу внаслідок використання викопного палива. Ці процеси викликають занепокоєння у світлі дефіциту енергоресурсів і захисту навколишнього середовища. В останні кілька років виникла гостра потреба запровадження технологій відновлювальної енергетики для вирішення цих проблем [41, 43, 44, 119].

Особливе місце в цій системі займає біоенергетика, яка пов'язана з використанням біомаси для виробництва енергії. Використання біомаси (деревина, солома, сміття, а також біологічні залишки та відходи) для виробництва енергії може зменшити залежність від нестійких джерел енергії, таких як нафта та природний газ, та сприяти зменшенню викидів парникових газів [43, 44, 119].

Виробництво біогазу у спосіб переробки органічних відходів життєдіяльності людини та відходів тваринництва і агропромислового комплексу є одним із пріоритетних напрямків розвитку альтернативної (зеленої) енергетики.

Біогаз виробляється з відновлюваних джерел, таких як органічна біомаса, що робить його екологічно чистим джерелом енергії. Використання біогазу допомагає зменшити залежність від вуглеводневих палив та інших нестійких джерел енергії. Біогаз залишається актуальним та перспективним джерелом енергії, яке може відігравати важливу роль у майбутньому енергетичного сектору [36, 107, 108, 119].

Процеси виробництва біогазу є досить технологічно складними та вимагають додаткового вивчення структури сировини та залучення експертного середовища для їх налаштування в залежності від типу та структури сировини. Загальна схема функціонування типової біогазової установки деталізовано представлена на рисунку 1.1 [91, 95].

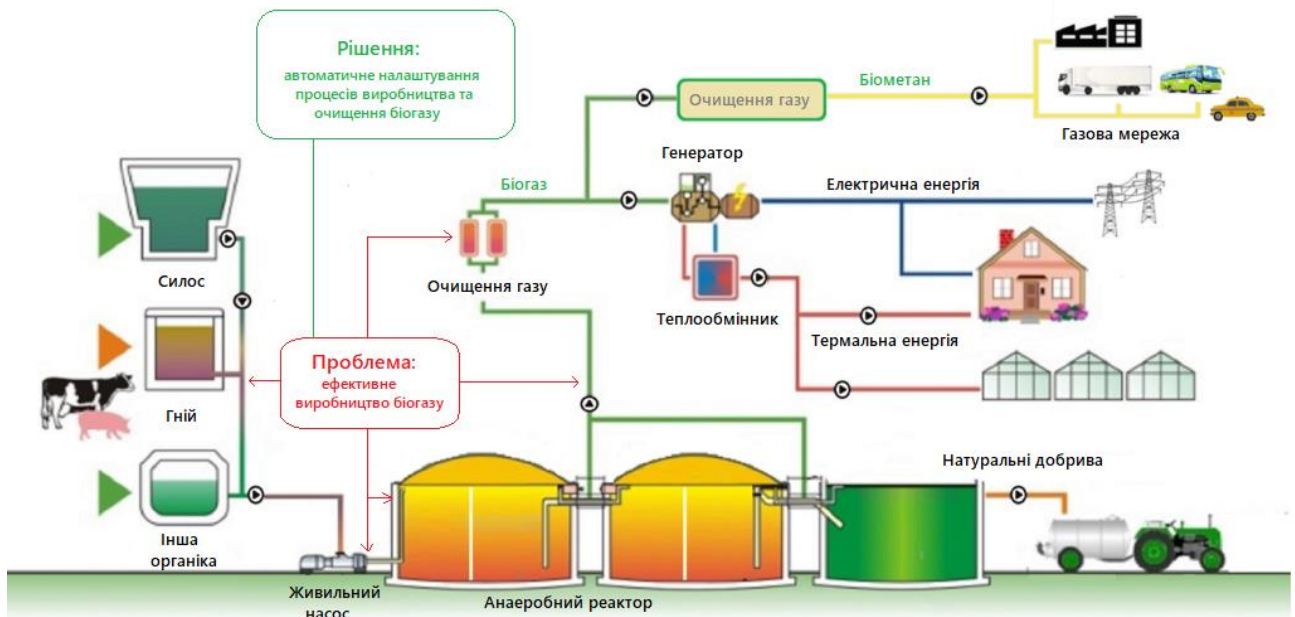


Рис. 1.1. Схема функціонування біогазової установки

Ефективність процесу анаеробного мікробіологічного бродіння залежить від різних параметрів, таких як об'єм завантаженої сировини ( $OLR$ ), конструкція біореактора, інгібітори та токсини ( $I$  та  $T$ ), летючі органічні сполуки, вміст органічних речовин, час гідравлічного утримання, леткі органічні кислоти, вологість сухої частини біосировини ( $HRT$ ) кислотність  $pH$  і температура ( $T$ ), (рисунок 1.2).

Склад вхідної сировини. Сюди відносять відходи сільськогосподарської діяльності (гній, жом, тощо), енергетичні культури, осад стічних вод, біовідходи, тверді міські відходи та промислові харчові відходи [22, 42, 90, 109-111, 119].

Конструкція реактора. Біогазова установка складається зі сховища для обробки, блоку зброджування, газосховищ, трубопроводів та газоперетворювальної установки.

Склад субстрату безпосередньо впливає на структуру компонентів у процесі анаеробного бродіння. Дослідження комбінації вхідних компонентів може суттєво впливати на ефективність функціонування біогазової установки [52, 53, 93, 94, 96].

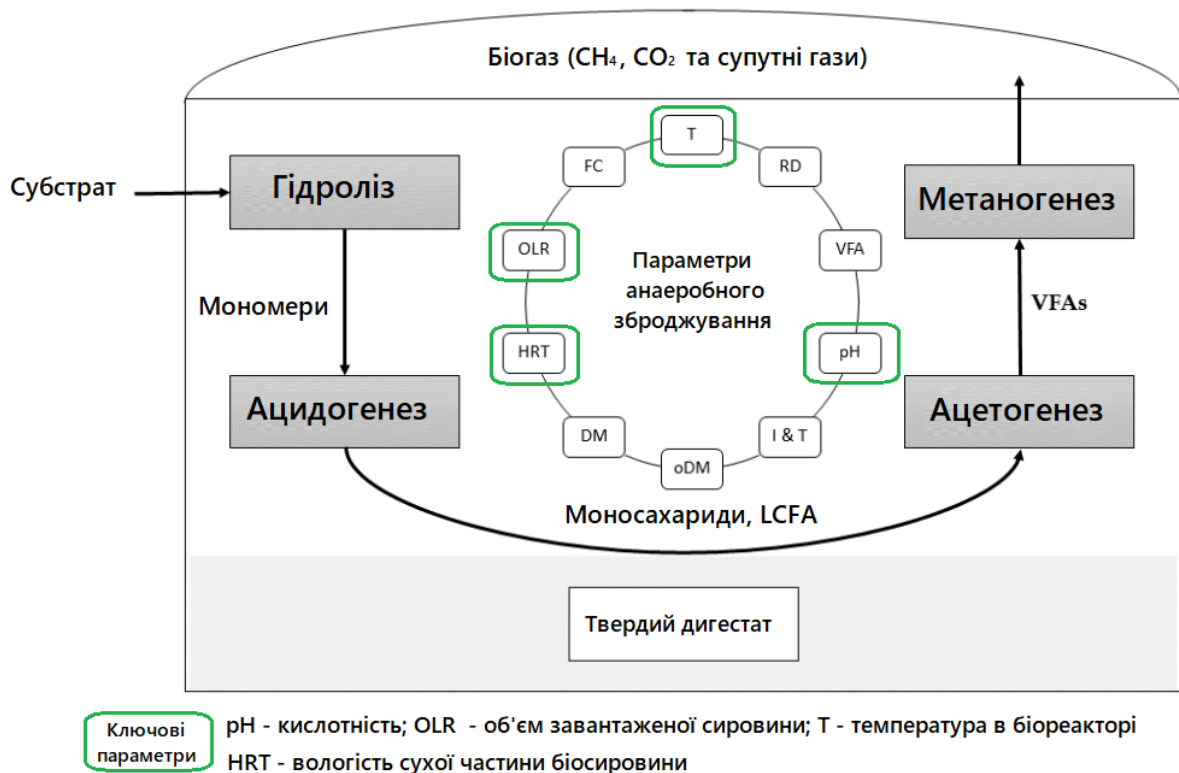


Рис. 1.2. Характеристики процесу анаеробного бродіння

Інгібітори/токсини: аміак, сульфід, іони легких металів (*Na, K, Mg, Ca* та *Al*), важкі метали (*Cr, Fe, Co, Cu, Zn, Cd* і *Ni*), органічні сполуки призводять до деструкції процесу в певних концентраціях [20].

Накопичення згаданих інгібіторів або токсинів може спричинити зниження виробництва біометану та, зрештою, призвести до збою процесу анаеробного бродіння [5, 8, 90].

Гідролітичні бактерії, які беруть участь у стадії гідролізу, можуть бути інгібовані надмірною кількістю інгібіторів та токсинів, парціальним тиском водню та гумінових кислот [9, 10, 17, 18]. Крім того, стадії ацетогенезу та метаногенезу в основному пригнічуються накопиченням аміаку. Загальна концентрація аміаку, яка перевищує допустиму норму, викликає повне гальмування процесу анаеробного бродіння [97]. Тим не менш, інгібування, викликане токсинами, не є незворотним процесом і при зміні параметрів управління процесом зброджування в біореакторі може бути відновлене [56, 106]. Інгібування аміаку в процесі анаеробного зброджування може бути спричинене іоном амонію.

Крім згаданих органічних сполуки, легкі і важкі метали можуть чинити гальмівну дію на процес бродіння. З іншого боку, легкі та важкі метали необхідні для метаболізму мікроорганізмів і для ферментативних реакцій.

Загальний вміст твердих речовин (TS) і летких твердих речовин (VS): вміст TS в субстраті визначає впроваджені в технологічний процес технології вологого та сухого зброджування [34].

Як правило, при вологому зброджуванні сировина містить 10–15% TS і збільшується до 24–40% TS при сухому зброджуванні. Леткі тверді речовини є одним з найважливіших параметрів субстрату, які використовуються для визначення питомої потужності виробництва біогазу [34]. Подібний вихід біогазу можна отримати в результаті процесів сухого та мокрого зброджування з відповідною кількістю живлення субстрату.

Швидкість органічного завантаження вказує на кількість летючих твердих речовин, що подаються до реактора за одиницю часу [34]. Підвищення цього показника призводить до підвищення ефективності всього процесу.

Гідравлічний час утримання визначає час перебування субстрату в системі, яка розраховується як ділення об'єму реактора на добову подачу субстрату. Більш тривалий час сприяє ефективній деградації. З іншого боку, маючи невеликий гідравлічний час, малі робочі об'єми підходять для виконання процесу бродіння.

Леткі жирні кислоти перетворюються в ацетат у процесі анаеробного бродіння. Особливо необхідно відзначити, що вони є суттєвими показниками для визначення ефективності етапу ацетогенезу. Їх концентрація вище норми призводить до пригнічення гідролізу.

*pH*(кислотність) різних мікроорганізмів, які беруть участь у процесі, потребують різних значень *pH* для їх оптимальної роботи. При його відхиленні від норми ефективність процесу виробництва біогазу знижується, а при досягненні критичних значень процесу метанового бродіння зупиняються через суттєве зменшення концентрації мезофільних бактерій [22].



Як правило, інгібування аміаку призводить до накопичення летких жирних кислот, які можна виявити через високі значення рН [7, 22].

Температура ( $T$ ). Процес анаеробного зброджування класифікується за трьома категоріями залежно від робочої температури: психрофільний ( $<25\text{ }^{\circ}\text{C}$ ), мезофільний ( $37\text{--}42\text{ }^{\circ}\text{C}$ ) і теплолюбний ( $50\text{--}60\text{ }^{\circ}\text{C}$ ). Температура безпосередньо впливає на мікробну динаміку і ефективність всього процесу [24, 60, 82]. Крім того, на активність ферментів впливає температура. Більшість ферментів стабільні в мезофільному діапазоні температур [24, 81]. З іншого боку, підвищення температури призводить до збільшення концентрації аміаку. Як бачимо з наведених структурних параметрів анаеробного бродіння, необхідно здійснювати чітке управління цим процесом, а вмiле поєднання необхідних комбiнацiй буде безпосередньо впливати на ефективність функціонування всієї біогазової установки.

## **1.2 Аналіз засобів моніторингу та контролю процесів у біогазових установках**

Як було зазначено в попередньому пункті, процес анаеробного бродіння складається з послідовних етапів: гідроліз, ацидогенез, ацетогенез та метаногенез. Ці етапи проходять одночасно, тому проблеми, що виникають на одному з етапів, можуть впливати на баланс усього процесу створення біогазу. Час, який витрачається на кожному етапі, залежить від особливостей субстрату, що подається в реактор. Гідроліз вважається обмежуючим етапом для складних органічних сполук, які не можуть бути легко гідролізовані, що може призвести до накопичення легко-гідролізованих сполук. З іншого боку, метаногенез, який є ключовим етапом процесу анаеробного бродіння, є обмежуючим етапом для субстратів, які легко розкладаються.

Дуже важливим засобом управління процесами у біогазових установках є онлайн-моніторинг та контроль [22, 119]. Збільшення кількості великомасштабних

установок підвищує попит на відповідний моніторинг і контроль цих систем. Системи моніторингу та контролю використовуються по-різному залежно від конкретної специфіки. За допомогою онлайн-моніторингу та контролю оптимізація процесу виробництва біогазу можлива за рахунок максимального використання виробничих потужностей та мінімізації втрат внаслідок відмов.

Для забезпечення стабільності та ефективності процесу необхідний постійний моніторинг і контроль. Сильно пригнічена популяція мікроорганізмів або повний збій усієї установки може мати серйозні фінансові наслідки для операторів установки анаеробного бродіння. За допомогою моніторингу процесу можна отримати загальну картину процесу виробництва біогазу, виявити майбутні нестабільності в анаеробних реакторах до того, як станеться збій, супроводжувати успішний запуск або повторний перезапуск всієї установки [22,106].

Частота живлення мікроорганізмів безпосередньо впливає на продуктивність процесу та середовище їхнього існування. Дослідження показали, що щоденне годування спричиняє більше коливання концентрації ацетату та рН (кислотності) порівняно з годуванням десять разів на день [22,106]. Крім того, зміна складу вихідної сировини викликає варіації в мікробіології процесу та їх продуктивності. Тому зміну субстратних сумішей слід проводити обережно, що дозволить адаптувати процес до нових умов.

Температура впливає на весь процес анаеробного бродіння шляхом впливу на швидкість деградації, вихід біогазу та стабільність всього процесу функціонування біогазової установки. Різні види мікроорганізмів, які беруть участь у процесі, мають свої специфічні оптимальні температури зростання. Для забезпечення інтенсивного контакту між мікроорганізмами та субстратом необхідний достатній час контакту, який може бути забезпечений відповідною стратегією змішування. Крім того, створенню шарів (плаваючих шарів і осаду) запобігає адекватне перемішування.

На рисунку 1.3 наведено схему, яка описує основні проблеми, які можуть виникати в процесі функціонування біогазової установки.

Процес виробництва біогазу є складним, тому сучасні біогазові установки потребують гнучкого, стабільного та недорогого онлайн-контролю всього процесу виробництва біогазу. Тому ефективний моніторинг та контролю в процесах анаеробного бродіння є дуже важливим та має прямий вплив на виробництво біогазу [22, 39, 42, 90].

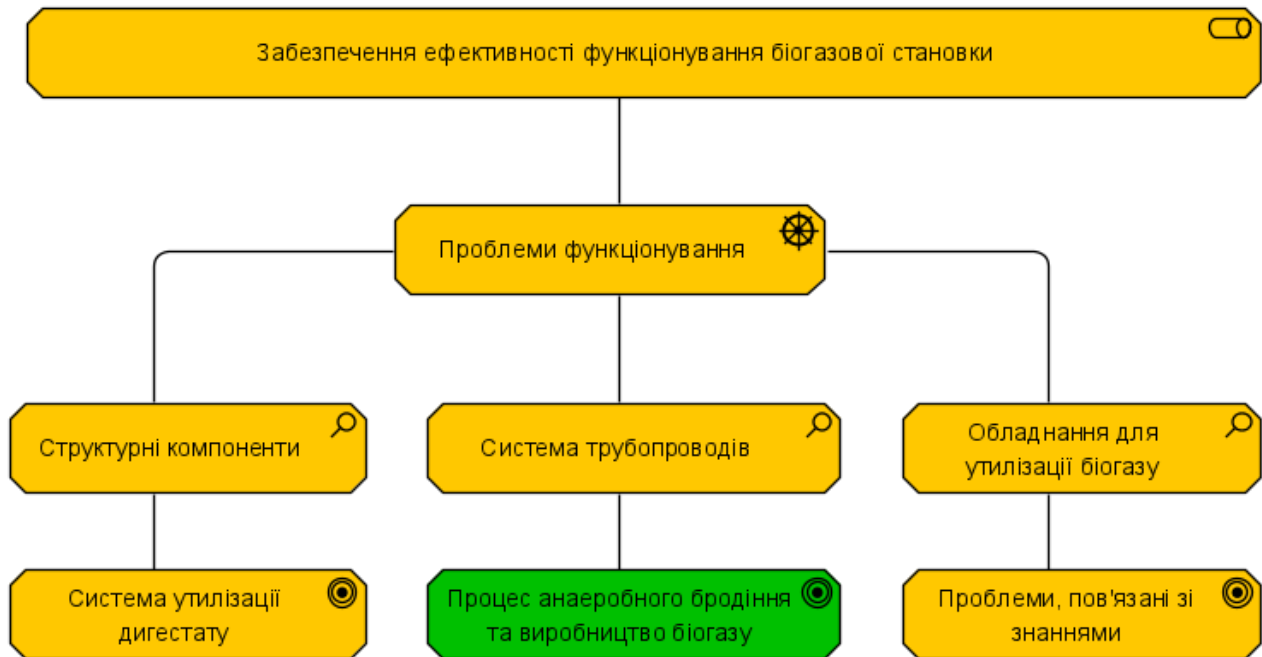


Рис. 1.3. Основні проблеми, які можуть виникати в процесі функціонування біогазової установки

Відсутність моніторингу та контролю на біогазових установках може призвести до збою процесу. Обидва інструменти, що використовуються на заводі, і параметри процесу повинні контролюватися в межах всього перебігу процесу [22]. Контроль вихідного матеріалу, якості біогазу, температури, тиску, насосів, змішувачі та кришки варочного котла необхідні для забезпечення реалізації та безперервності процесу.

Крім того, засоби візуального контролю можуть забезпечити швидке та економічне виявлення проблем, що в основному реалізується за допомогою міток [9, 22, 37].

З іншого боку, для біогазових установок існують різні стратегії автоматичного керування в залежності від складності.

Увімкнуті-вимкнуті — це проста стратегія управління, яка підходить для керування клапанами та насосами. Незважаючи на це, він не може забезпечити точний контроль і не має процесу прямого впливу на стабільність.

Адаптивне управління можна використовувати для управління нелінійними/динамічними системами, які забезпечують оцінку параметрів і прогнозування майбутніх збурень. Проте детально інформація та складні математичні розрахунки необхідні для моделі, яка може включати деяку невизначеність.

Нечітка логіка може бути реалізована в системах з кількома входами/виходами та нелінійних системах, але вона орієнтована на досвід оператора. Штучна нейронна мережа не вимагає моделі чи досвіду, але вимагається час для навчання і потрібні великі обсяги даних [78].

Загалом стратегії моніторингу процесу анаеробного бродіння можна класифікувати за трьома категоріями: мінімальний, стандартний і розширений моніторинг, як представлено на рисунку 1.4 [29].

Моніторинг процесу необхідний для визначення характеру процесу або для раннього виявлення проблем. Моніторинг різних робочих параметрів на біогазових установках може здійснюватися як онлайн, що передбачає вимірювання параметрів у режимі реального часу безпосередньо на біогазовій установці, так і автономний моніторинг за допомогою лабораторного аналізу. Онлайн моніторинг дозволяє отримувати дані безпосередньо під час функціонування установки, що дозволяє оперативно реагувати на зміни у параметрах. Автономний моніторинг через лабораторний аналіз надає можливість більш докладного та глибокого дослідження параметрів, але може бути менш оперативним у порівнянні з онлайн підходом. Обидва методи можуть бути корисними в залежності від потреб, завдань та можливостей дослідження на біогазових установках.

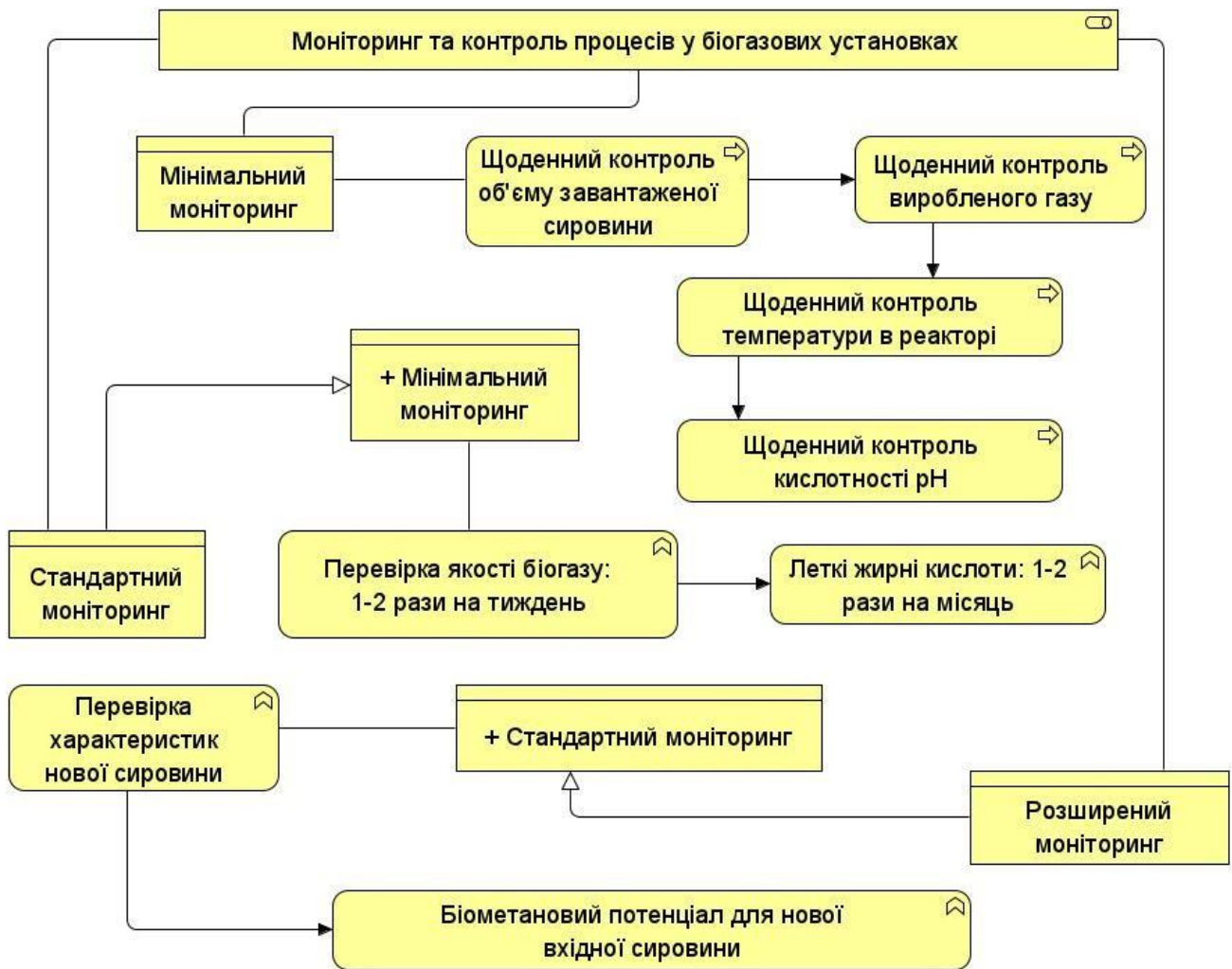


Рис. 1.4. Моніторинг процесів у біогазових установках

На додаток до цих двох технологій можна реалізувати поточний моніторинг для аналізу проб із датчиками в трубопроводі, що також відображає реальну ситуацію в реакторі [58].

### 1.3 Аналіз методів моделювання процесів у біогазових установках

Моделювання процесів у біогазових установках - це важлива та складна задача, що вимагає поєднання знань з різних предметних областей, зокрема, біології, хімії, інженерії та інших галузей науки. Біогазові установки використовують біологічний процес для перетворення органічних матеріалів, таких як стічні води, рослинні залишки та харчові відходи, у біогаз - суміш метану та вуглекислого газу.

На рисунку 1.5 представлено схему, яка описує основні аспекти, на які направлені процедури математичного моделювання процесів у біогазових установках. Розглянемо ці аспекти більш детально.

Моделювання кінетики біохімічних реакцій у біогазових установках є ключовим для розуміння та прогнозування процесів метаногенезу та анаеробного розкладу органічних сполук. Цей підхід базується на кінетичних моделях, таких як модель Monod або модель Michaelis-Menten [4, 22, 43], які враховують залежність швидкості реакції від концентрації субстрату та інгібіторів, а також вплив температури та кислотності рН на активність мікроорганізмів.

Моделі масообміну та теплообміну у біогазових установках базуються на рівняннях руху речовини та рівняннях теплопередачі. Вони враховують гідродинамічні умови у біореакторі, такі як температура, концентрація та розподіл речовини, а також теплові потоки, що виникають внаслідок біохімічних реакцій та теплообміну з навколишнім середовищем.

Розробка систем контролю та регулювання базується на математичних моделях динамічних процесів у біогазових установках. Ці моделі враховують динаміку системи, залежність вихідних параметрів від вхідних сигналів та алгоритми керування, спрямовані на забезпечення оптимальних умов для біохімічних процесів [45, 47].

Моделі газового потоку у біогазових установках враховують газодинамічні процеси, такі як продукція та відвід біогазу, його розподіл та зберігання. Вони також враховують вплив різноманітних факторів, таких як тиск, температура та вологість, на властивості газової суміші.

Для оцінки економічної ефективності біогазових установок розробляються моделі, які враховують витрати на будівництво та експлуатацію установки, доходи від продажу біогазу та інших продуктів, а також інші фактори, такі як ризики та рівень інвестицій. Ці аспекти моделювання є важливими для розробки та оптимізації біогазових установок, забезпечуючи їхню ефективність, стабільність та економічну вигоду.



Рис. 1.5. Основні аспекти математичного моделювання процесів у біогазових установках

Досі недостатньо знань про процес анаеробного бродіння та його реакції на зміни робочих параметрів, таких як сировина, температура та кислотність. Достатній моніторинг сам по собі не може допомогти впровадити підсистему прогнозування на біогазових установках. Отже, потрібно детально проаналізувати підходи до моделювання процесів анаеробного бродіння.

Першу модель для моделювання біогазових установок створив Andrews в 1969, а найскладнішою моделлю є модель анаеробного зброджування №1 (ADM1), яка виконує моделювання різних субстратів серед усіх моделей [8, 22, 43, 61, 62, 88, 89]. ADM1, створений Bastone та ін. (2002) є найбільш широко використовуваною моделлю в останні роки [22, 43, 88]. Характеристики методів моделювання процесів у біогазових установках наведено в таблиці 1.1.-1.4., та детально описано в [22, 119].

В таблиці 1.1. наведено характеристику моделей, які формують «перше покоління». Зазначені моделі характеризуються намаганням змоделювати основні мікробіологічні процеси.

Таблиця 1.1.

Методи математичного моделювання процесів у біогазових установках, перше покоління

Тип моделі	Модель	Опис моделі
Перше покоління	Andrews, (1969)	Ця модель показує, що моделювання кроку обмеження швидкості дає інформацію про весь процес. Бактеріальне пригнічення можна пояснити накопиченням кислоти.
	Andrews та Graef, (1970)	Проведено динамічне моделювання процесу ферментативного гідролізу складних органічних сполук
	Hill та Barth, (1977)	Ця модель була створена для забезпечення стабільності в процесі анаеробного бродіння відходів тваринництва. З урахуванням масових балансів між леткими речовинами, леткими кислотами, розчинною органікою, двома групами бактерій, катіонами, азотом і вуглекислим газом розраховано значення кислотності рН.
	Heyes та Hall, (1981)	Було розроблено динамічну модель для представлення інгібування воднем ацетогенезу та інгібування рН метаногенезу з використанням глюкози як субстрату.
	Hill, (1983)	Модель була розроблена для імітації стаціонарної продуктивності метану (якісної та кількісної) у процесі анаеробного бродіння відходів тваринництва.
	Mosey, (1983)	Чотири бактеріальні групи були визначені в моделі виробництва біогазу через AD глюкози. У моделі ацетогенез визначається як лімітуючий етап.

У таблиці 1.2. представлено моделі, що формують «друге покоління» і які



були направлені на вирішення проблем, які виникали в досліджуваних класах фізико-хімічних систем.

Таблиця 1.2.

Методи математичного моделювання процесів у біогазових установках, друге покоління

Тип моделі	Модель	Опис моделі
Друге покоління	Costello. (1991)	Реакторний процес, фізико-хімічна система та біологічний склад були використані в системі для створення математичної моделі. Крім того, у модель включено накопичення молочної кислоти, інгібування продукту та кислотність рН.
	Angelidaki (1993)	Модель була розроблена для моделювання анаеробної деградації складних органічних матеріалів з охопленням ферментативної гідролітичної стадії, чотирьох бактеріальних стадій і 12 хімічних сполук.
	Vavilin (1996)	Було розроблено модель для моделювання гідролітичної (лімітуючої) стадії анаеробного бродіння. Модель включає поверхневу колонізацію частинок гідролітичними бактеріями та поверхневу деградацію.
	Husain (1998)	Функції Моно використовувалися для визначення рівня смертності кислотогенів і метаногенів.

У таблиці 1.3. представлено моделі, що формують «третє покоління» і які були направлені на комплексне вирішення проблем анаеробного мікробіологічного бродіння.

Таблиця 1.3.

Методи математичного моделювання процесів у біогазових установках, третє покоління

Тип моделі	Модель	Опис моделі
Третє покоління	Bernard (2001)	Модель балансу маси була розроблена для визначення параметрів на стадіях ацидогенезу та метаногенезу процесу. Електрохімічна рівновага використовується для включення лужності в модель.
	Siegrist (2002)	Швидкість гідролізу, ацетотрофний метаногенез і деградація пропіонату були спеціальним фокусом створеної математичної моделі, яка моделювала динамічну поведінку як мезофільного, так і термофільного бродіння.
	Bastone (2002)	ADM1 включає як біохімічні, так і фізико-хімічні процеси. 26 змінних концентрації динамічного стану, 8 неявних алгебраїчних змінних і 32 змінних стану концентрації використовуються в цій узагальненій моделі.
	Zaher (2009)	Модель була створена для розуміння мікробної активності на основі наявності макроелементів ( <i>C, H, N, O, P</i> і <i>S</i> ) і термодинаміки ацидогенезу та метаногенезу.
	Rajendran (2014)	46 реакцій (для інгібування, швидкісткінетики, рН, аміаку, об'єму, швидкості завантаження та часу утримування) виконуються в моделі для прогнозування виробництва біогазу з будь-якого субстрату та за будь-яких робочих умов із використанням Aspen Plus.
	Arzate (2015)	Ця модель являє собою поєднання характеристики оцінки життєвого циклу і математичної моделі продуктивності процесу, яка може забезпечити зниження впливу процесів бродіння на середовище.

У таблиці 1.4. представлено моделі, що формують окремий клас моделей.

Серед зазначеного класу моделей необхідно виділити інтервальний підхід, в рамках якого і виконувалося дане дисертаційне дослідження.

Таблиця 1.4.

Методи математичного моделювання процесів у біогазових установках, інші моделі

Тип моделі	Модель	Опис моделі
Статистичні моделі	Barampouti (2005)	Модель була створена для прогнозування виробництва біогазу з дослідженням 17 параметрів з дворічних даних роботи станції очищення стічних вод.
	Nopharatana (2007)	Модель створено для моделювання біологічних реакцій у реакторі з твердих побутових відходів з розглядом їх у двох фракціях; розчинні і нерозчинні. У моделі реалізовано рівняння Контуа, Моно та Гомперца.
	Yusuf та Ify (2011)	Модель була створена для прогнозування максимального та кінцевого виходу біогазу та кінцевого виходу метану при спільному зброджуванні коров'ячого гною та водяного гіацинта на основі кінетичної моделі першого порядку.
	Syaichurrozi та Sumardino (2013)	Кінетична модель для визначення виробництва біогазу була розроблена з модифікуючим рівнянням Гомперца. Досліджено вплив співвідношення COD/N на кінетичну модель.
	Brule (2014)	Модель була створена для оптимізації аналізів сировини. Вона забезпечує контроль якості аналізів, інтерпретацію кінетики реакції та оцінку виходу метану.
Інтервальний підхід	Дивак, Гураль (2018)	Модель оцінювання щодобового виходу метану під час анаеробного мікробіологічного бродіння

Розглянемо детальніше деякі моделі з точки зору їх математичної

постановки. Для аналізу вибрано по одній моделі з кожного проаналізованого класу підходів до моделювання процесів у біогазових установках.

Рівняння Моно є одним із фундаментальних математичних виразів, які використовуються для опису кінетики росту мікроорганізмів у біореакторах та біологічних системах. Ця модель описує залежність швидкості росту мікроорганізмів від концентрації субстрату (поживних речовин) в середовищі. Рівняння Моно можна використовувати для прогнозування та оптимізації процесів біоперетворення, включаючи формування біогазу у біогазових установках [22, 42, 98, 119].

Рівняння Моно представляється у наступному вигляді:

$$\mu = \mu_{max} \frac{S}{K_s + S}, \quad (1.1)$$

де  $\mu$  - швидкість росту мікроорганізмів (часова зміна їхньої концентрації);  $\mu_{max}$  - максимальна швидкість росту мікроорганізмів (константа, яка визначається типом мікроорганізмів та умовами середовища);  $S$  - концентрація субстрату (поживних речовин);  $K_s$  - константа Моно, яка вказує на афінитет мікроорганізмів до субстрату.

Це рівняння відображає гіперболічну залежність швидкості росту від концентрації субстрату. Якщо концентрація субстрату  $S$  дуже велика, то  $\mu$  наближається до  $\mu_{max}$ , тобто мікроорганізми ростуть з максимальною швидкістю. Якщо ж концентрація субстрату низька, то швидкість росту пропорційно зменшується.

Це рівняння може бути використане для моделювання та прогнозування росту мікроорганізмів у біогазових установках, де вони використовуються для біологічного перетворення органічних відходів у біогаз.

Модель Кемпера-Метьюса є ще однією ключовою математичною моделлю, яка використовується для опису кінетики реакцій, зокрема для біохімічних процесів, які відбуваються у біогазових установках [22, 42, 89, 98, 119]. Ця модель

розглядає залежність швидкості реакції від концентрації субстрату, проте відрізняється від моделі Моно, оскільки враховує максимальну швидкість реакції ( $V_{max}$ ) та константу Міхаеліса-Ментен ( $K_M$ ).

Рівняння моделі Кемпера-Метьюса має наступний вигляд:

$$v = V_{max} \frac{[S]}{K_M + [S]}, \quad (1.2)$$

де  $v$  - швидкість реакції;  $V_{max}$  - максимальна швидкість реакції (константа, що залежить від кінцевої концентрації ферменту),  $[S]$  - концентрація субстрату (поживних речовин),  $K_M$  - константа Міхаеліса-Ментен, яка вказує на взаємозв'язок між швидкістю реакції та концентрацією субстрату.

Це рівняння також відображає гіперболічну залежність швидкості реакції від концентрації субстрату, але враховує максимальну можливу швидкість реакції ( $V_{max}$ ). Якщо концентрація субстрату  $[S]$  дуже велика, то  $v$  наближається до  $V_{max}$ . Якщо ж концентрація субстрату низька, то швидкість реакції пропорційно зменшується.

Модель Кемпера-Метьюса є корисним інструментом для аналізу та прогнозування біохімічних процесів, які відбуваються в біогазових установках, де мікроорганізми перетворюють органічні відходи на біогаз.

Модель реактора зі змінною швидкістю наповнення (Continuous Stirred-Tank Reactor, CSTR) є однією з базових моделей для аналізу та моделювання біохімічних процесів, таких як ті, які відбуваються у біогазових установках. У CSTR реакційна суміш стабільно перемішується, а нові реагенти додаються до реактора, а відпрацьовані продукти віддаляються, забезпечуючи стабільну температуру та склад реакційної суміші [89, 98, 119].

Математична модель CSTR може бути виражена за допомогою рівняння матеріального балансу для реакційної речовини у реакторі. Зазвичай це рівняння першого порядку, що враховує подачу субстрату, відбір продуктів та хімічні реакції в середовищі реактора. Рівняння матеріального балансу для CSTR може бути

записане у вигляді:

$$\frac{dC}{dt} = \frac{F}{V} (C_{in} - C) - kC , \quad (1.3)$$

де  $C$  - концентрація реакційної речовини у реакторі;  $C_{in}$  - вихідна концентрація реакційної речовини;  $F$  - об'ємний витік;  $V$  - об'єм реактора;  $k$  - кінетична константа реакції.

Це рівняння враховує різницю між концентрацією вхідної речовини і концентрацією у реакторі, а також кінетику реакції, що відбувається у реакторі. Модель CSTR використовується для аналізу та оптимізації багатьох хімічних та біологічних процесів, таких як розклад органічних сполук, ферментація та інші біохімічні реакції.

Важливо враховувати, що модель CSTR має свої обмеження, такі як необхідність стабільного змішування у реакторі та припущення про ідеальне розподілення концентрації у всьому об'ємі реактора. У деяких випадках можуть бути використані більш складні моделі, які враховують неідеальне змішування та інші фактори.

Модель теплопередачі описує перенос тепла від одного середовища до іншого. У біогазових установках, модель теплопередачі може бути використана для аналізу теплових процесів, таких як нагрівання субстрату, теплове регулювання, та інші теплові обміни, які відбуваються в реакторі.

Одним із основних рівнянь для моделювання теплопередачі є рівняння Фур'є:

$$q = -k \frac{dT}{dx} , \quad (1.4)$$

де  $q$  - тепловий потік (потік тепла на одиницю площі);  $k$  - коефіцієнт теплопровідності матеріалу;  $\frac{dT}{dx}$  - градієнт температури по відстані  $x$ .

Це рівняння показує, що тепловий потік через деяку поверхню залежить від

градієнту температури та коефіцієнта теплопровідності матеріалу [26, 27].

Однак, в біогазових установках можуть бути враховані інші аспекти теплопередачі, такі як конвекція та випромінювання тепла. У цьому випадку рівняння можуть бути більш складними і враховувати додаткові чинники.

Загалом, модель теплопередачі допомагає аналізувати теплові процеси в біогазових установках, що дозволяє оптимізувати енергоефективність та забезпечувати оптимальні умови для біологічних процесів.

Модель газового потоку у реакторі відображає те, як гази рухаються та переносяться усередині реактора. Ця модель важлива для розуміння гідродинамічних процесів та оптимізації роботи реактора, зокрема у біогазових установках.

Одним із ключових рівнянь для моделювання газового потоку є рівняння Нав'є-Стокса, яке описує рух газу у термодинамічно стабільній реакційній системі. Однак, через складність цього рівняння, часто використовують спрощені моделі, які враховують тільки основні аспекти газодинаміки, такі як розподіл тиску та швидкості [4, 119].

Одна з таких спрощених моделей - модель ідеального змішування (Ideal Mixing Model). Вона передбачає, що гази у реакторі повністю змішуються, і кожна точка у реакторі має однакову концентрацію газів та однаковий тиск. Ця модель дозволяє легше розраховувати газові потоки та проводити оцінку реакцій.

Інша поширена модель - модель плугового потоку (Plug Flow Model), яка передбачає, що гази рухаються в реакторі без змішування, а кожен газовий потік утворює плуг. Ця модель дозволяє краще аналізувати структуру та динаміку газового потоку в реакторі.

Ще один важливий аспект моделі газового потоку - розрахунок теплових ефектів. Оскільки біогазові реактори часто включають теплові процеси, такі як газифікація або термохімічні реакції, модель газового потоку також може включати розрахунок теплових втрат та теплового впливу на реакції.

Враховуючи ці аспекти, модель газового потоку у реакторі дозволяє краще

розуміти та оптимізувати роботу біогазових установок, забезпечуючи ефективну та стабільну роботу реакторів.

Розглянемо детальніше математичні співвідношення для цієї моделі, які можуть бути виражені наступним чином, матеріальний баланс:

$$\frac{dC}{dt} = \frac{F_{in}C_{in} - F_{out}C + R}{V}, \quad (1.5)$$

де  $C$  - концентрація газу в реакторі;  $t$  - час;  $F_{in}$  - об'ємний витік газу в реактор;  $C_{in}$  - вхідна концентрація газу;  $F_{out}$  - об'ємний витік газу з реактора;  $R$  - швидкість газової реакції (якщо є);  $V$  - об'єм реактора.

Рівняння збереження маси:

$$\frac{dM}{dt} = \frac{F_{in} - F_{out}}{V}, \quad (1.6)$$

де  $M$  - маса газу в реакторі.

Рівняння стану газу (ідеальний газ):

$$PV = nRT, \quad (1.7)$$

де  $P$  - тиск газу;  $V$  - об'єм реактора;  $n$  - кількість молекул газу;  $R$  - універсальна газова стала;  $T$  - температура газу.

Кінетичне рівняння (якщо враховується газова реакція):

$$R = kC, \quad (1.8)$$

де  $k$  - кінетична константа реакції.

Ці співвідношення дозволяють моделювати динаміку зміни концентрації газу, маси газу та інших параметрів у реакторі за плином часу за умови



рівномірного розподілу концентрації.

Проаналізовані моделі ґрунтуються на властивостях біохімічних перетворень і мають детерміністичний характер, що унеможлиблює їх застосування при розробці прототипу біогазової установки для автоматичного налаштування процесу виробництва та очищення біогазу в біогазовій установці (рисунок 1.6).



Рис.1.6. Переваги застосування методів аналізу інтервальних даних для моделюванні процесів у біогазових установках [119]

Адже ці моделі не можуть враховувати можливі відхилення на будь-якій стадії виробництва та очищення біогазу в технологічних параметрах цих процесів і не враховують типу та структури біосировини. Тому для вирішення даних проблем доцільно використати методи математичного моделювання на основі інтервального аналізу.

Методи моделювання на основі інтервального аналізу даних детально представлені в роботах [30-33, 114-118]. Інтервальний підхід до моделювання об'єктів має переваги перед стохастичним підходом, оскільки не потребує дослідження статистичних характеристик об'єкта, тим самим зменшуючи необхідність у великій кількості експериментів. Цей підхід особливо корисний для

аналізу складних об'єктів при обмеженій кількості даних.

У даному методі дані представляються у вигляді інтервалів можливих значень характеристик об'єкта [114, 115, 120]:

$$[z_{(i,j,h,k)}^-; z_{(i,j,h,k)}^+], i = 0, 1, 2, \dots, I, j = 0, 1, 2, \dots, J, h = 0, 1, 2, \dots, H, k = 0, 1, 2, \dots, K, (1.9)$$

де  $[z_{(i,j,h,k)}^-; z_{(i,j,h,k)}^+]$  - нижня та верхня межі інтервалів, які вказують на діапазон можливих значень вихідної характеристики об'єкта в певній точці з дискретно заданими просторовими координатами та  $k = 0, 1, 2, \dots, K$  часовими параметрами для динамічних об'єктів.

Також важливо ураховувати похибки вимірювань при встановленні нижніх і верхніх меж інтервалів:

$$z_{(i,j,h,k)}^- = z_{(i,j,h,k)} - z_{(i,j,h,k)} \cdot \varepsilon \text{ та } z_{(i,j,h,k)}^+ = z_{(i,j,h,k)} + z_{(i,j,h,k)} \cdot \varepsilon, (1.10)$$

де  $\varepsilon$  – величина, яка характеризує відносну похибку вимірювань.

Інтервальне представлення може бути корисним, коли похибка вимірювань суттєво перевищує інші види похибок і задає межі допуску відхилень від номінальних значень характеристик.

Для представлення моделі об'єкта необхідно визначити математичний об'єкт. У загальному випадку, модель обмежується дискретною лінійною моделлю, яка може бути виражена у наступній формі:

$$\vec{u}_{i,j,h,0}, \dots, \vec{u}_{i,j,h,k} \cdot \vec{g}, i = d, \dots, I, j = d, \dots, J, h = d, \dots, H, k = d, \dots, K \quad (1.11)$$

де  $\vec{f}^T(\bullet)$  - вектор базисних функцій використовується для перетворення значень модельованих характеристик об'єкта та вхідних змінних в дискретних точках простору та часу.

Процедура структурної ідентифікації розкриває дискретну модель, включаючи вектор базисних функцій, розмірність вхідних змінних та інші параметри. Параметрична ідентифікація полягає у визначенні параметрів моделі таким чином, щоб забезпечити відповідність моделі реальним значенням характеристик об'єкта, які отримані на основі експериментальних даних.

Цей підхід дозволяє враховувати велику кількість можливих значень характеристик об'єкта в межах встановлених інтервалів, що допомагає уникнути чи зменшити вплив невизначеності та похибок у дослідженнях. Отже, застосування методів інтервального аналізу даних може допомогти краще розуміти та інтерпретувати результати досліджень зазначеного класу об'єктів.

#### **1.4 Постановка задачі дослідження**

Характеристики процесу анаеробного бродіння пропонується встановити за результатами структурної та параметричної ідентифікації у вигляді інтервальних дискретних динамічних моделей [114-119]. Їх ідентифікацію доцільно проводити на основі даних вимірювань з обмеженими за амплітудою похибками, тобто представленими в інтервальному вигляді [116,117]. Саме завдяки використанню методів аналізу інтервальних даних отримаємо математичний опис процесів у біогазовій установці у вигляді дискретних динамічних моделей з гарантованою точністю. Використання зазначених моделей дасть можливість розробити прототип біогазової установки для автоматизованого адаптивного налаштування технологічних параметрів установки під визначені тип і структуру сировини, підвищення обсягів виробленого біогазу та дозволить налаштувати подальший процес очищення біогазу від сторонніх сполук для отримання високоякісного біометану, який відповідатиме європейським стандартам якості, що тим самим забезпечить підвищення ефективності функціонування біогазової установки.

Для розв'язування вказаної проблеми буде використано методи інтервального аналізу даних в контексті моделювання динамічних систем, де

існують похибки вимірювань, які обмежені за амплітудою, і враховуються технологічні допуски на параметри процесів виробництва та очищення біогазу.

На основі цього підходу мають бути розроблені методи ідентифікації інтервальних дискретних динамічних моделей, які забезпечать можливість прогнозування характеристик процесу анаеробного мікробіологічного бродіння у біогазових установках на різних стадіях (ацидогенез, ацетогенез, метаногенез) і дозволяють адаптивно налаштовувати параметри в залежності від структури і типу біосировини, що в свою чергу, дозволить забезпечити неперервну подачу субстрату у біореактор, щоб уникнути його закисання та відповідно підвищити кількість виробленого обсягу біогазу.

Звісно такий підхід вимагає розв'язування ряду нових наукових задач, включаючи створення нових методів ідентифікації таких моделей на основі аналізу інтервальних даних і побудову нових математичних моделей процесів у біогазових установках у вигляді інтервальних різницевих рівнянь та їх адаптації до використання в складі прототипу біогазової установки з автоматичним налаштування параметрів. Але, водночас, застосування такого підходу та розроблення нового математичного та програмного забезпечення прототипу установки для моделювання процесів у біогазовій установці дозволить в реальному часі управляти цими процесами і тим самим підвищити ефективність їх функціонування.

Виходячи із вищезазначеного, актуальною є задача забезпечення допустимих значень кислотності багатокomпонентного субстрату за рахунок розробки математичного та програмного забезпечення для моделювання динаміки кислотності субстрату у біогазових установках на основі аналізу інтервальних даних.

Для вирішення поставленої задачі у дисертаційному дослідженні поставлено наступні завдання:

- провести аналіз відомих методів та засобів для моделювання процесів у біогазових установках, виділити їх основні переваги та недоліки;

- розробити двоетапний метод ідентифікації математичної моделі динаміки показника кислотності субстрату у біореакторі на основі аналізу інтервальних даних;
- розробити математичну модель динаміки показника кислотності субстрату у біореакторі;
- вдосконалити сервісно-орієнтовану архітектуру програмного забезпечення для моделювання процесів у біогазових установках;
- розробити комп'ютерне середовище для моделювання процесів у біогазових установках з сервісно-орієнтованою архітектурою;
- провести апробацію розроблених методів та моделей, розробленого середовища для управління процесом налаштування біогазової установки на структуру сировини та темпів її подачі у біореактор.

## **Висновки до розділу 1**

1. Описано особливості розвитку джерел альтернативної енергетики, де віділено виробництво біогазу у спосіб переробки органічних відходів життєдіяльності людини та відходів тваринництва і агропромислового комплексу як один із пріоритетних напрямків розвитку альтернативної (зеленої) енергетики. Показано, підвищення ефективності функціонування цього виробництва вимагає розробки нових підходів управління процесами в біогазових установках на основі застосування інформаційних технологій та методів математичного моделювання.

2. Розглянуто підходи до моделювання з точки зору функціонування енергетичної системи, а також з точки зору оптимізації процесів у біогазових установках. Відомі моделі, які описують процес виробництва біогазу демонструють різний рівень деталізації. Моделювання процесів у біогазових установках може надати динамічну інформацію про процес анаеробного зброджування, наприклад, для прогнозування виходу біогазу або оптимізації процесу анаеробного зброджування.

3. Показано, що одною із визначальних характеристик процесу метанового бродіння в БГУ є показник рН ферментаційного середовища, тобто його кислотність. Запропоновано та обгрунтовано застосування методів математичного моделювання з метою відображення взаємозв'язку між основною характеристикою процесу та чинниками, які на неї впливають так і динаміки основної характеристики процесу, яка визначається кислотністю субстрату у біореакторі. Для побудови математичних моделей обох видів запропоновано використати методи параметричної ідентифікації моделей статичних об'єктів та дискретних моделей динаміки об'єктів на основі аналізу інтервальних даних.

## РОЗДІЛ 2

### МОДЕЛЮВАННЯ ХАРАКТЕРИСТИК ПРОЦЕСІВ У БІОГАЗОВИХ УСТАНОВКАХ НА ОСНОВІ АНАЛІЗУ ІНТЕРВАЛЬНИХ ДАНИХ

У попередньому розділі розглянуто особливості функціонування біогазових установок при виробництві біогазу та проведено огляд існуючих методів структурної і параметричної ідентифікації інтервальних моделей статичних об'єктів у формі алгебричних рівнянь та моделей у вигляді різницевих рівнянь, які відображають вихідні характеристики динамічних об'єктів.

У цьому розділі розглянуто ідентифікацію обох, вищезгаданих, типів математичних моделей для відображення функціонування біогазових установок при виробництві біогазу.

Як зазначалося, виробництво біогазу у спосіб переробки органічних відходів життєдіяльності людини та відходів тваринництва і агропромислового комплексу є одним із пріоритетних напрямків розвитку альтернативної (зеленої) енергетики. Для забезпечення підвищення ефективності функціонування цього виробництва необхідно розробляти нові підходи до управління процесами в біогазових установках на основі застосування методів математичного моделювання.

У цьому розділі, для відображення взаємозв'язку між основною характеристикою процесу (показником кислотності субстрату) та чинниками, які на неї впливають, запропоновано використати нелінійне алгебричне рівняння, а для відображення динаміки основної характеристики процесу, яка визначається кислотністю субстрату у біореакторі – запропоновано використати різницеве рівняння. Також це різницеве рівняння перетворене до диференціального рівняння.

Для розробки обох математичних моделей, запропоновано та обґрунтовано використати методи аналізу інтервальних даних та метод параметричної ідентифікації цих моделей. Такі математичні моделі, хоча і характеризуються «гарантованими» прогностичними властивостями, проте вимагають застосування для їх ідентифікації складних оптимізаційних процедур з багатоекстремальними

функціями мети та часто ці функції представлені в дискретному вигляді. Тому, для розв'язування цих задач запропоновано використати цілісний універсальний підхід (однаковий для обох типів математичних моделей), який ґрунтується на метаевристичних алгоритмах оптимізації, які, своєю чергою, використовують механізми самоорганізації та самоадаптації в процесі пошуку оптимального чи квазіоптимального розв'язку. Серед метаевристичних алгоритмів запропоновано та обґрунтовано алгоритм, який симулює ройовий алгоритм штучної бджолоїної колонії.

Для пошуку структури моделі динаміки кислотності субстрату у біореакторі запропоновано двоетапний алгоритм. Спочатку, за допомогою більш простішої задачі ідентифікації моделі взаємозв'язку між основною характеристикою процесу та чинниками, які на неї впливають, отримано нелінійне алгебричне рівняння. Потім, на основі його аналізу обрано необхідні структурні елементи для моделі динаміки кислотності субстрату у біореакторі. Також для інтерпретації фізичних властивостей динаміки кислотності субстрату у біореакторі, запропоновано та обґрунтовано на підставі отриманого різницевого рівняння здійснити перехід до диференціального, в якому коефіцієнти мають фізичну інтерпретацію, на відміну від різницевого рівняння. В основі запропонованого методу використано формули розкладу значень шуканої функції у вузлах різницевої сітки в ряд Тейлора в околі базового вузла, а також різницевого представлення похідних в цьому ж околі. Також при цьому переході враховано представлення управляючих функцій.

Результати цього розділу опубліковано автором у працях [65, 71, 85, 116, 124].

## **2.1 Постановка задач параметричної ідентифікації моделей характеристик процесів у біогазових установках**

Спочатку розглянемо постановку задачі побудови математичної моделі для відображення взаємозв'язку між основною характеристикою процесу, тобто



показником рН ферментаційного середовища та чинниками, які на неї впливають. В цьому випадку розглядаємо сам об'єкт як статичний, а вище зазначена залежність описується таким алгебричним рівнянням:

$$y(\vec{\beta}, \vec{X}) = f_1(\vec{\beta}, \vec{X}) + \dots + f_m(\vec{\beta}, \vec{X}), \quad (2.1)$$

де  $y(\vec{\beta}, \vec{X})$  - означає модельоване (обчислене) значення рН ферментаційного середовища;

$\vec{\beta}$  - вектор невідомих параметрів моделі;

$f_1(\vec{\beta}, \vec{X}), \dots, f_m(\vec{\beta}, \vec{X})$  - означає набір нелінійних базисних функцій, які, як бачимо, залежні від вектора вхідних змінних  $\vec{X}$  і від вектора параметрів  $\vec{\beta}$  моделі;

$\vec{X}$  - вектор вхідних змінних, які визначають структуру біомаси, що завантажують в біореактор, інші параметри процесу, такі як вологість сухої сировини та температуру в біореакторі.

Результати експерименту, які необхідні для ідентифікації параметрів нелінійної моделі (2.1) подаємо у такому вигляді [116]:

$$\vec{X}_i \rightarrow [y_i^-; y_i^+], \quad i = 1..N, \quad (2.2)$$

де  $[y_i^-; y_i^+]$  – нижня та верхня межі експериментально отриманих значень рН середовища для заданих  $i$ -тих умов проведення вимірювання, які визначаються вектором  $\vec{X}_i$ , для кожного з  $i = 1..N$  вимірювань.

В цьому випадку задача параметричної ідентифікації моделі у вигляді виразу (2.1) полягає в оцінюванні значень вектора  $\vec{\beta}$  параметрів. Наявність цих оцінок дає можливість отримати математичну модель з відображенням взаємозв'язку між основною характеристикою процесу, тобто показником рН ферментаційного середовища та чинниками, які на неї впливають у такому вигляді:

$$\hat{y}(\vec{\beta}, \vec{X}) = f_1(\vec{\beta}, \vec{X}) + \dots + f_m(\vec{\beta}, \vec{X}). \quad (2.3)$$

Для вищезазначеної задачі  $\hat{y}(\vec{\beta}, \vec{X})$  означає обчислене значення рН середовища ферментації для заданих умов та чинників.

Виходячи із вимог забезпечення заданої точності моделі, які використовують в аналізі інтервальних даних, можемо констатувати, що змодельовані значення рН середовища повинні належати числовим інтервалам, які отримані експериментально. Тобто при визначенні оцінок параметрів, для отриманої моделі мають виконуватися умови:

$$\hat{y}(\vec{\beta}, \vec{X}_i) \in [y_i^-; y_i^+], i = 1..N. \quad (2.4)$$

Звідси, підставляючи в умови (2.4) замість  $\hat{y}(\vec{\beta}, \vec{X}_i)$  вираз (2.3), отримаємо умови для оцінювання параметрів моделі  $\vec{\beta}$ :

$$y_i^- \leq f_1(\vec{\beta}, \vec{X}_i) + \dots + f_m(\vec{\beta}, \vec{X}_i) \leq y_i^+, i = 1..N. \quad (2.5)$$

Отримана система (2.5) є інтервальною системою нелінійних алгебраїчних рівнянь (ІСНАР) для невідомих оцінок вектора параметрів [116]. Набір рішень ІСНАР визначає множину векторів оцінок параметрів моделі.

Через високу (комбінаторну) обчислювальну складність розв'язання цієї ІСНАР на практиці обчислюють лише точкові оцінки параметрів. У цьому випадку для оцінки параметрів розв'язують оптимізаційну задачу у такому вигляді:

$$\delta(\vec{\beta}) \xrightarrow{\vec{\beta}, \alpha_i} \min, \vec{\beta} \in [\vec{\beta}^{low}; \vec{\beta}^{up}], \alpha_i \in [0,1], i = 1..N, \quad (2.6)$$

де  $\alpha_i$  – коефіцієнти лінійної комбінації для визначення точки в межах

експериментальних даних  $[y_i^-; y_i^+]$ , які додано для забезпечення гладкості цільової функції.

Цільова функція у цьому випадку має такий вигляд [71, 116]:

$$\begin{aligned} \delta(\vec{\beta}) &= \sum_{i=1}^N \left( \hat{y}_i(\vec{\beta}, \vec{X}_i) - P([y_i^-; y_i^+], \alpha_i) \right)^2 = \\ &= \sum_{i=1}^N \left( f_1(\vec{\beta}, \vec{X}_i) + \dots + f_m(\vec{\beta}, \vec{X}_i) - (\alpha_i \cdot y_i^- + (1 - \alpha_i) \cdot y_i^+) \right)^2. \end{aligned} \quad (2.7)$$

Як бачимо, такий підхід хоча і розширює простір змінних для пошуку вектора параметрів моделі, але разом з тим забезпечує гладкість функції мети в оптимізаційній задачі (2.6).

У випадках, коли рівняння (2.3) є нелінійним за параметрами, цільова функція (2.7) в оптимізаційній задачі (2.6) стає більш складнішою ніж квадратична, має велику кількість локальних мінімумів, то для розв'язування оптимізаційної задачі доцільно використовувати методи стохастичної оптимізації, еволюційні та метаевристичні алгоритми. Зокрема, в цьому випадку запропоновано використати алгоритм ройового інтелекту колонії медоносних бджіл [51, 71, 116].

Тепер розглянемо випадок, коли необхідно відобразити динаміку показника рН ферментаційного середовища.

Такі математичні моделі можна ефективно використовувати в управлінні процесами виробництва біогазу чи для дослідження фізичних властивостей процесу.

Для цього випадку, експериментальні дані подаємо у вигляді дискретних значень виміряного значення характеристики процесу:

$$\vec{X}_k \rightarrow [y_k^-; y_k^+], \quad k = 1..K, \quad (2.8)$$

де  $[y_k^-; y_k^+]$  – нижня та верхня межі експериментально отриманих значень рН

середовища для заданих умов проведення вимірювання в  $k$ -тий дискретний момент часу під дією набору значень керуючих чинників  $\vec{X}_k$  у ті ж самі дискретні моменти часу  $k = 1..K$ .

Враховуючи спосіб представлення результатів експерименту у вигляді інтервалів (2.8), задача параметричної ідентифікації математичної моделі показника рН ферментаційного середовища має множину рівнозначних розв'язків [2, 3, 21, 119].

Математичну модель динаміки показника рН ферментаційного середовища розглядаємо як дискретне рівняння у такому вигляді:

$$y_k(\vec{Y}, \vec{X}) = f_1(\vec{Y}_k, \vec{X}_k) * \beta_1 + f_2(\vec{Y}_k, \vec{X}_k) * \beta_2 + \dots + f_m(\vec{Y}_k, \vec{X}_k) * \beta_m, k = d, \dots, K, \quad (2.9)$$

де  $y_k(\vec{Y}, \vec{X})$  означає модельоване значення показника рН ферментаційного середовища;

$d$  – порядок різницевої схеми (2.9);

$\vec{\beta}$  - вектор невідомих параметрів моделі;

$f_1(\vec{Y}_k, \vec{X}_k), f_2(\vec{Y}_k, \vec{X}_k), \dots, f_m(\vec{Y}_k, \vec{X}_k)$  - означає набір базисних функцій, які, як бачимо залежні від вектора управляючих змінних  $\vec{X}$ , і від вектора  $\vec{Y}_k$ .

Вектор  $\vec{Y}_k$  у виразі (2.9) означає значення вихідної характеристики у попередні моменти часу, які впливають на динаміку процесу в наступні моменти часу і має такий вигляд:

$$\vec{Y}_k = (y_{k-d}, \dots, y_{k=d}, \dots, y_{k-1})^T. \quad (2.10)$$

Позначимо оцінки вектора параметрів за  $\vec{\hat{\beta}}$ . Тоді, математична модель характеристики об'єкта матиме вигляд інтервального різницевого рівняння:

$$\begin{aligned} [\hat{y}_k] = & f_1([\vec{Y}_k], \vec{X}_k) * \hat{\beta}_1 + f_2([\vec{Y}_k], \vec{X}_k) * \hat{\beta}_2 + \dots + \\ & + f_m([\vec{Y}_k], \vec{X}_k) * \hat{\beta}_m, k = d, \dots, K, \end{aligned} \quad (2.11)$$

де  $[\vec{Y}_k] = ([y_{k-d}, \dots, [y_{k=d}, \dots, [y_{k-1}])^T$  – інтервальний вектор з компонентами, які означають обчислені інтервальні оцінки показника рН ферментаційного середовища у попередні моменти часу.

За аналогією умов гарантованої точності, які розглянуто вище для моделі статичної системи, спираючись на гіпотезу, що

$$[\hat{y}_k^-; \hat{y}_k^+] \subset [y_k^-; y_k^+], \forall k = d, \dots, K, \quad (2.12)$$

отримуємо математичну задачу для обчислення оцінки  $\vec{\hat{\beta}}$  вектора параметрів моделі [55, 88, 119]:

$$\begin{cases} [\hat{y}_0^-; \hat{y}_0^+] \subset [y_0^-; y_0^+], \dots, [\hat{y}_{k=d-1}^-; \hat{y}_{k=d-1}^+] \subset [y_{k=d-1}^-; y_{k=d-1}^+] \\ y_k^- \leq f_1([\vec{Y}_k], \vec{X}_k) * \hat{\beta}_1 + \dots + f_m([\vec{Y}_k], \vec{X}_k) * \hat{\beta}_m \leq y_k^+; \\ k = d, \dots, K. \end{cases} \quad (2.13)$$

Отримана система (2.13) є подібною до інтервальної системи (2.5), тобто є ІСНАР для невідомих оцінок вектора параметрів. Проте, на відміну від ІСНАР (2.5) вона є складнішою, через наявність початкових умов у вигляді

$$[\hat{y}_0^-; \hat{y}_0^+] \subset [y_0^-; y_0^+], \dots, [\hat{y}_{k=d-1}^-; \hat{y}_{k=d-1}^+] \subset [y_{k=d-1}^-; y_{k=d-1}^+]. \quad (2.14)$$

Через високу (комбінаторну) обчислювальну складність розв'язування цієї ІСНАР, на практиці обчислюють лише точкові оцінки параметрів.

Тому замінімо ці умови на простіші, які дають можливість обчислювати

точкові оцінки як вектора параметрів моделі так і точкові оцінки показника рН ферментаційного середовища. Тобто замість умов (2.14) в ІСНАР (2.13) використаємо такі умови:

$$\hat{y}_0 \in [y_0^-; y_0^+], \dots, \hat{y}_{k=d-1} \in [y_{k=d-1}^-; y_{k=d-1}^+]. \quad (2.15)$$

Тоді ІСНАР (2.13) перепишемо в такому вигляді:

$$\left\{ \begin{array}{l} \hat{y}_0 \in [y_0^-; y_0^+], \dots, \hat{y}_{k=d-1} \in [y_{k=d-1}^-; y_{k=d-1}^+] \\ y_k^- \leq f_1(\vec{Y}_k, \vec{X}_k) * \hat{\beta}_1 + \dots + f_m(\vec{Y}_k, \vec{X}_k) * \hat{\beta}_m \leq y_k^+; \\ k = d, \dots, K. \end{array} \right. \quad (2.16)$$

Як бачимо, отримана ІСНАР не містить інтервальних оцінок показника рН ферментаційного середовища, отриманих на попередніх дискретах. В тому сенсі зазначена інтервальна система подібна до ІСНАР (2.5), яка є математичним формулюванням задачі ідентифікації моделі статичного об'єкта.

Тепер можемо перейти за аналогією з випадком ІСНАР (2.5) до формулювання оптимізаційної задачі у такому вигляді:

$$\delta(\vec{\beta}) \xrightarrow{\vec{\beta}, \alpha_k} \min, \vec{\beta} \in [\vec{\beta}^{low}; \vec{\beta}^{up}], \alpha_k \in [0, 1], k = d..K. \quad (2.17)$$

Як бачимо, задачу параметричної ідентифікації інтервальної моделі динаміки показника рН ферментаційного середовища, отримали у вигляді оптимізаційної задачі - подібної до задачі (2.6) параметричної ідентифікації математичної моделі для відображення взаємозв'язку між значенням показника рН ферментаційного середовища процесу та чинниками, які на нього впливають.

Разом з тим, функція мети  $\delta(\vec{\beta})$  в даному випадку має такий вигляд:

$$\begin{aligned} \delta(\vec{\hat{\beta}}) &= \sum_{k=d}^K (\hat{y}_k - P([y_k^-; y_k^+], \alpha_k))^2 = \\ &= \sum_{i=d}^K \left( f_1(\vec{Y}_k, \vec{X}_k) \cdot \beta_1 + \dots + f_m(\vec{Y}_k, \vec{X}_k) \cdot \beta_m - (\alpha_k \cdot y_k^- + (1 - \alpha_k) \cdot y_k^+) \right)^2, \end{aligned} \quad (2.18)$$

де,  $\hat{y}_k \in [y_k^-; y_k^+]$ ,  $\vec{Y}_k \in [\vec{Y}_k]$ .

Вище встановлені закономірності, дають підстави щоб для розв'язування обох оптимізаційних задач (2.6) та (2.17) використати спільний метод, який побудовано на поведінкових моделях бджолої колонії. Варто зауважити, що функція мети (2.18) формується рекурентно і тому є складнішою, з точки зору нелінійності у порівнянні із функцією мети (2.7) вищезгаданих оптимізаційних задач.

Варто також зауважити, що інтервальна модель (2.11) набуває такого вигляду:

$$\begin{aligned} \hat{y}_k &= f_1(\vec{Y}_k, \vec{X}_k) * \hat{\beta}_1 + f_2(\vec{Y}_k, \vec{X}_k) * \hat{\beta}_2 + \dots + \\ &+ f_m(\vec{Y}_k, \vec{X}_k) * \hat{\beta}_m, k = d, \dots, K. \end{aligned} \quad (2.19)$$

Тобто, як бачимо з виразу (2.19) оцінки динаміки показника рН ферментаційного середовища отримуємо у точковому, а не множинному вигляді. Разом з тим, вектор оцінок параметрів може відрізнятися для кожної математичної моделі. При цьому, варіація значень параметрів залежить від ширини інтервалів, отриманих інтервальних даних (2.8).

## **2.2 Обчислювальна схема методу параметричної ідентифікації моделей характеристик процесів у біогазових установках**

Для розв'язування отриманих у попередньому підрозділі оптимізаційних задач у вигляді (2.6) чи (2.17) обрано метаевристичний алгоритм, який ґрунтується

на поведінковій моделі бджолої колонії, і в якому закладено основні принципи ройового інтелекту [32, 63, 84]. Основні ідеї цього алгоритму полягають в принципах самоорганізації та самоадаптації колонії медоносних бджіл при пошуку джерел нектару [1-3, 6, 51, 105].

Розглянемо дану поведінкову модель в рамках розв'язування задачі (2.6) чи (2.17).

На *фазі ініціалізації* задаємо параметри алгоритму:  $S$  – чисельність усієї популяції бджіл,  $MCS$  – загальну кількість ітерацій,  $mcs = 1$  – номер поточної ітерації,  $LIMIT$  – число, яке визначає вичерпність джерела (для виходу з локальних мінімумів), а також початкові дані у вигляді (2.2) чи (2.8) і обираємо відповідно структури моделі у вигляді (2.1) чи (2.9). Також для задачі параметричної ідентифікації дискретної моделі динаміки показника рН ферментаційного середовища задаємо початкові умови:

$$\hat{y}_0 \in [y_0^-; y_0^+], \dots, \hat{y}_{k=d-1} \in [y_{k=d-1}^-; y_{k=d-1}^+]. \quad (2.20)$$

Тоді переходимо на фазу бджіл розвідників.

*Фаза бджіл розвідників.* Позначимо за  $\vec{\beta}_l$  вектори оцінок параметрів моделі та вектор коефіцієнтів  $\alpha_i \in [0,1], i = 1..N$ , які визначають на поточній ітерації можливі розв'язки задачі оптимізації (2.6) чи (2.17). Ці вектори є претендентами на мінімум функції мети (2.9). На цій фазі обчислюємо зазначені вектори  $\vec{\beta}_l$ , використовуючи таку формулу:

$$\beta_{jl} = \beta_{jl}^{low} + rand(0,1) \cdot (\beta_{jl}^{up} - \beta_{jl}^{low}), j = 1..m + N, l = 1..S. \quad (2.21)$$

Після цього, для усіх згенерованих за формулою (2.21) значень  $\vec{\beta}_l$  обчислюємо значення функції мети, щоб оцінити якість отриманих оцінок параметрів  $\hat{\beta}_j, j=1, \dots, m$  математичної моделі (2.19) та допоміжних коефіцієнтів



$\alpha_i \in [0,1], i = 1..N$  за формулою (2.18).

*Фаза робочих бджіл.* На цій фазі, додатково до обчислених оцінок на попередній фазі, обчислюємо вектори оцінок параметрів моделі та вектор коефіцієнтів  $\alpha_i \in [0,1], i = 1..N$ , які визначають на поточній ітерації можливі розв'язки задачі оптимізації (2.6) чи (2.17). При цьому, на цій фазі обчислюємо вектори  $\vec{\beta}_l$ , використовуючи таку формулу:

$$\beta_{jl}^{mcn} = \beta_{jl} + rand(-1,1) \cdot (\beta_{jl} - \beta_{jp}), j = 1..m + N, p \neq l = 1..S, \quad (2.22)$$

де,  $j = 1, \dots, m + N$  – випадково обраний індекс параметра;  $\vec{\beta}_p$  – випадково обраний вектор координат із  $p \neq l = 1..S$ .

Далі перевіряємо, чи обчислені за формулою (2.22) вектори  $\vec{\beta}_l$  задовольняють умовам

$$\vec{\beta} \in [\vec{\beta}^{low}; \vec{\beta}^{up}], \alpha_i \in [0,1], i = 1..N \quad (2.23)$$

у випадку ідентифікації моделі (2.3) (статична система) чи умовам

$$\vec{\beta} \in [\vec{\beta}^{low}; \vec{\beta}^{up}], \alpha_k \in [0,1], k = d..K \quad (2.24)$$

для випадку моделювання динаміки процесу (модель (2.19)).

У випадку, коли отримане значення параметрів моделей чи коефіцієнтів не задовольняють умовам (2.23) при побудові моделі статичної системи, чи умовам (2.24) - при побудові моделі динаміки процесу, то замість формули (2.22) використовуємо таку формулу:

$$\beta_{jl}^{mcn} = \beta_{jl} - rand(-1,1) \cdot (\beta_{jl} - \beta_{jp}), j = 1..m + N, p \neq l = 1..S. \quad (2.25)$$

В контексті оптимізаційної задачі, ця фаза означає обчислення нових оцінок для пошуку розв'язку з меншими (оптимальними) значеннями функції мети (2.9). З цією метою, обчислюємо якість оцінок параметрів математичної моделі за формулою (2.9) чи (2.18) для відповідних оптимізаційних задач і проводимо попарне порівняння існуючих і поточних значень оцінок параметрів:

$$\vec{\beta}_l = \left\{ \vec{\beta}_l, \text{if } \delta(\vec{\beta}_l) \leq \delta(\hat{\beta}_l^{mcn}) \right\}$$

or

$$\left\{ \hat{\beta}_l^{mcn}, \text{if } \delta(\vec{\beta}_l) > \delta(\hat{\beta}_l^{mcn}) \right\}. \quad (2.26)$$

Попарне порівняння дає можливість виділити  $S$  –точок в просторі пошуку розв'язків оптимізаційних задач, які забезпечують найменші значення відповідної функції мети.

*Фаза бджіл дослідників.* На цій фазі проводимо «поглиблене» вивчення точок в околі точок, де функція мети задач оптимізації (2.6) чи (2.17) має найменші значення, які встановлено на попередніх фазах. Для цього, для кожної точки, яку визначили у спосіб попарного порівняння за формулою (2.26), обчислюємо ймовірність  $P_l$  за такою формулою:

$$P_l = (1 - \delta([\hat{\beta}_l])) / \sum_{i=1}^S (1 - \delta([\hat{\beta}_i])). \quad (2.27)$$

Варто зауважити, що на кожній фазі результатом є  $l=1, \dots, S$  – отриманих точок в просторі розв'язків. Також, варто взяти до уваги, що  $\delta(\vec{\beta})$  у формулі (2.27) нормоване до 1.

В кінцевому результаті кількість точок для дослідження околу можливого мінімуму функції мети для кожної із обраних  $S$  – точок обчислюємо за формулою

$$R_l = \text{int}(P_l \cdot S), \quad (2.28)$$

де символ  $\text{int}(P_l \cdot S)$  - означає виділення цілого із результату обчислень у виразі.

Відповідно, у випадку якщо  $R_l > 0$ , тоді обчислюємо  $R_l$  нових сусідніх точок за формулами (2.22) чи (2.25). Для кожної точки із загальної кількості  $R_l$  обчислюємо якість оцінок параметрів математичної моделі за формулою (2.9) чи (2.18), для відповідних оптимізаційних задач. Далі проводимо групову перевірку отриманих та існуючих значень оцінок параметрів за формулою (2.26).

Наступний кроком, є пошук точки, яка забезпечує найменше значення функції мети (2.9) чи (2.18) на даній ітерації алгоритму.

Оптимізаційну задачу розв'язано, якщо хоча б для однієї із отриманих точок в просторі розв'язків, значення функції мети дорівнює нулю, тобто:  $\delta(\vec{\beta}_l) = 0$ . В іншому випадку, - збільшуємо лічильник *limit* для точки, у якій забезпечується найменше значення функції мети і навколо якої здійснювали пошук на одиницю і переходимо до формул, спочатку (2.26) - (2.27) – для обчислення та розподілу кількості точок для обстеження, а після цього до обчислення, власне, координат точок за формулами (2.22) чи (2.25) і подальших кроків реалізації цієї фази.

*Фаза бджіл розвідників.* Для уникнення зациклення на певних точках при пошуку мінімуму функції мети, використовуємо фазу бджіл розвідників. Ця фаза настає у випадку вичерпання джерела нектару. Тому для кожної точки в просторі розв'язків оптимізаційної задачі, згенерованих та відібраних на попередніх фазах, перевіряємо чи не перевищує її лічильник «вичерпаності» *limit* граничного значення LİMİT, яке встановлене як параметр алгоритму на початковій фазі ініціалізації. Точки, для яких перевищено значення LİMİT замінюємо новими, які генеруємо за допомогою формули (2.21). Після цього знову переходимо до фази робочих бджіл.

### 2.3 Математична модель взаємозв'язку між показником рН ферментаційного середовища та чинниками, які на нього впливають

Як було зазначено вище, для підвищення ефективності функціонування біогазової установки доцільно відобразити залежність між показником кислотності рН ферментаційного середовища та чинниками, які впливають на цю кислотність. Такими чинниками є об'єми завантаженої сировини у вигляді сухої та рідкої її частини, температура в біореакторі та вологість сухої частини біосировини.

У нашому випадку, для розробки математичної моделі рН середовища в біогазовій установці, використано результати вимірювань, які надано ТОВ «Геофіпільська енергетична компанія», відповідно до проекту «Моделювання динаміки процесів на біогазових установках», номер державної реєстрації 0123U103785, замовник ТОВ «Захід Трейд Тернопіль», від 12.09.2023 - 30.09.2024р., а також гранту МОН України «Математичний інструментарій і програмне забезпечення прототипу біогазової установки підвищеної ефективності» (січень 2024–грудень 2025, номер державної реєстрації 0124U000076), а дані експериментальних досліджень отримано в грудні місяці.

Результати вимірювань наведено в таблиці 2.1. При цьому, в зазначеному місяці для анаеробного бродіння в реактор періодично завантажувалася така біосировина: барда; жом з цукрових буряків; патока та сечівка. Відповідно впливовими чинниками в даному випадку обрано:

$x_1$  – об'єм в м<sup>3</sup> завантаженої у відповідний період (в поточній добі) барди;

$x_2$  – маса в 1000кг завантаженого у відповідний період (в поточній добі) жому з цукрових буряків;

$x_3$  – об'єм в м<sup>3</sup> кг завантаженої у відповідний період (в поточній добі) патоки;

$x_4$  – об'єм в м<sup>3</sup> завантаженої у відповідний період (в поточній добі) сечівки;

$x_5$  – вологість в %;

$x_6$  – температура в °С середовища бродіння.

Таблиця 2.1.

Результати експериментальних досліджень характеристик та параметрів процесу ферментації в БГУ за грудень місяць

Номер точки контролю	Об'єм завантаження барди, м3	Маса завантаження жому, 1000кг	Об'єм завантаження патоки, м3	Об'єм завантаження сечівки, м3	Вологість, %	Температура в біореакторі, °С	Нижня межа вимірного рН	Верхня межа вимірного рН
$i$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$y_i^-$	$y_i^+$
1	130	129	0	0	96	34	7.8447	8.0353
2	70	159	0	0	96.6	34.7	7.8842	8.0758
3	80	147	0	14.4	96.1	32.1	8.0127	8.2073
4	130	120	39.4	10	96.635	35.4	7.9732	8.1668
5	200	101	0	0	96.5	34.8	7.8546	8.0454
6	150	97.5	16.5	0	97	35.8	7.8941	8.0859
7	30	102	14.4	10	96.4	35.9	8.1312	8.3288
8	110	112.5	14.4	15	97	36	7.9633	8.1567
9	320	69	0	5	96.7	36.3	7.9336	8.1264
10	120	100	14.4	0	96.9	36.1	7.9633	8.1567
11	130	0	0	0	96.8	36.1	7.9139	8.1061
12	210	37.5	14.4	0	97.2	36	7.9435	8.1365
13	220	51	14.4	0	96.3	36.9	7.8447	8.0353
14	40	33	0	11	96.696	38	8.0028	8.1972
15	220	137.5	14.4	10	96.8	38.2	8.0522	8.2478

На початковій фазі, перед реалізацією методу ідентифікації інтервальних нелінійних моделей, виходячи із фізичних міркувань була обрана структура алгебричного рівняння, яка пов'язує рН середовища бродіння із обсягом та структурою сировини, а також із температурою середовища та вологістю сировини:

$$\hat{y}(\hat{\beta}, \vec{X}) = \beta_0 + \left( \frac{\beta_1 \cdot x_2}{1 - \beta_2 \cdot x_1 + \beta_3 \cdot x_3 + \beta_4 \cdot x_4} \right)^{\beta_9} + \beta_5 \cdot x_5^{\beta_6} + \beta_7 \cdot x_6^{\beta_8}. \quad (2.29)$$

У рівнянні (2.28) застосування виразу  $\left(\frac{\beta_1 \cdot x_2}{\beta_3 \cdot x_1 + \beta_2 \cdot x_3 + \beta_4 \cdot x_4}\right)^{\beta_9}$  обґрунтовано дотриманням відношення частки сухих та рідких компонент біосировини у біореакторі.

Варто відзначити, що вище наведений вираз відношення частки сухих та рідких компонент біосировини є ключовим для обох типів моделей – у вигляді алгебричного рівняння (2.29) так і для моделі, яка відображає динаміку процесу, що буде показано у наступному параграфі.

Також можемо відзначити, що співвідношення між часткою сухих та рідких складових, які завантажують у ферментатор є визначальним для підвищення ефективності функціонування біогазової установки.

В результаті застосування методу ідентифікації інтервальних нелінійних моделей на основі обчислювальної схеми ройового інтелекту отримано такий вектор оцінок  $\vec{\hat{\beta}}$  параметрів моделі:

$$\vec{\hat{\beta}}^T = (8.852, 35.613, 3369.812, -4807.859, 31.381, -9.061, -1.193, -13.971, -0.788, 0.6246).$$

Вимірювання рН середовища здійснювалося пристроєм із похибкою 1%. Тепер підставимо отримані результати ідентифікації у вираз (2.29):

$$\hat{y}(\vec{\hat{\beta}}, \vec{X}) = 8.852 + \left( \frac{35.613 \cdot x_2}{1 - 3369.812 \cdot x_1 - 4807.859 \cdot x_3 + 31.381 \cdot x_4} \right)^{0.6246} - 9.061 \cdot x_5^{-1.193} - 13.971 \cdot x_6^{-0.788}. \quad (2.30)$$

Отримана математична модель у вигляді виразу (2.30) описує залежність між рН середовища і обсягом та структурою сировини, а також із температурою та вологістю.

На рисунку 2.1 наведено результати зіставлення експериментальних даних та модельованих значень рН середовища субстрату в біореакторі.

Як бачимо, результати моделювання належать до вимірних інтервалів значень рН, що вказує на адекватність моделі.

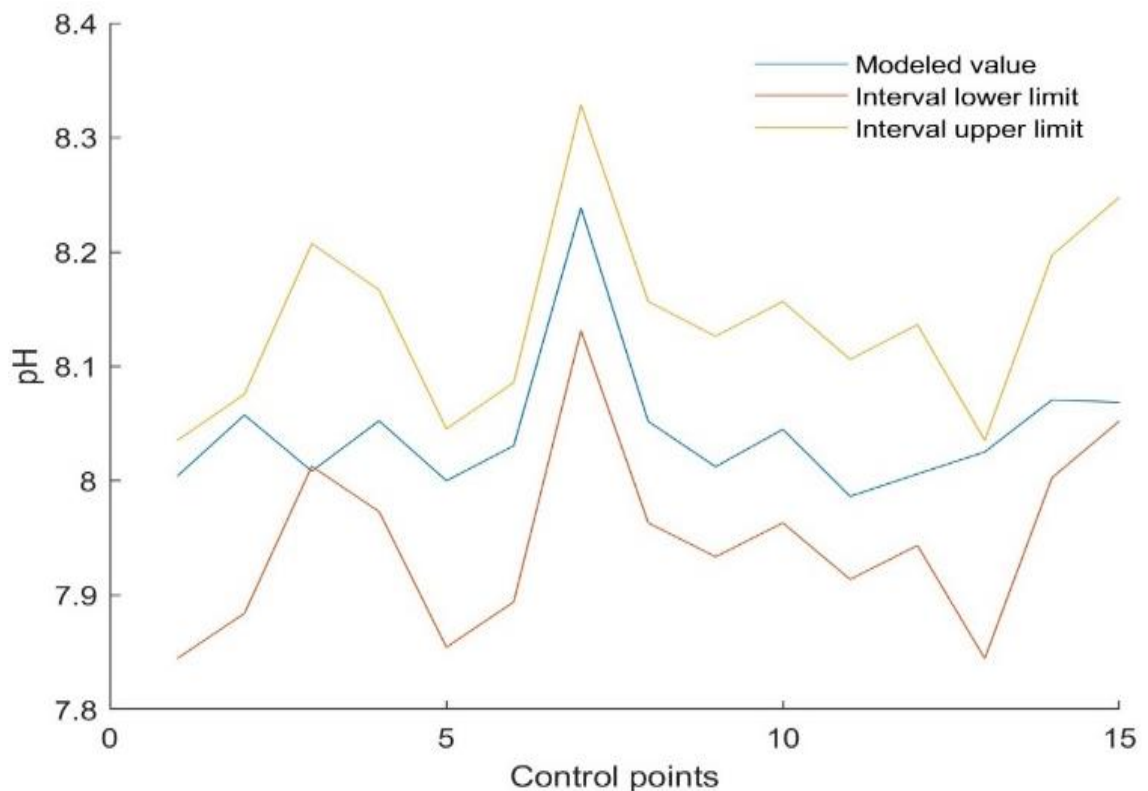


Рис. 2.1. Результати зіставлення експериментальних даних та модельованих значень рН субстрату

Для подальшого дослідження залежності рН від об'єму завантаження у відповідний період барди, маси завантаження жому з цукрових буряків, об'єму завантаженої патоки, об'єму завантаження сечівки, вологості та температури середовища бродіння необхідно проводити дослідження з метою визначення оптимальних значень чинників у кожен момент часу, який у цьому випадку вимірюється щодобово. Також для цих цілей необхідно побудувати модель динаміки показника кислотності рН ферментаційного середовища.

## 2.4 Математична модель динаміки показника кислотності рН ферментаційного середовища

Відображення динаміки процесу дає більш повну картину про можливості розробки механізму управління цим процесом. В цьому випадку відображається інерційність системи при зміні керуючих чинників. Як було зауважено вище, для побудови моделі динаміки можемо скористатися напрацюваннями, отриманими при побудові моделі (2.29), в якій є ключовий елемент, що задає відношення частки сухих та рідких компонент біосировини у біореакторі. Подібні ключові елементи необхідно внести в набір структурних елементів перед розв'язуванням задачі ідентифікації відповідної моделі динаміки. Як відомо [63-71, 84], перед розв'язуванням задачі структурної ідентифікації математичної моделі необхідно сформулювати набір структурних елементів, з яких формуватимуться моделі-претенденти.

Як і у попередньому випадку, для побудови моделі динаміки показника кислотності рН ферментаційного середовища скористаємося даними таблиці 1. Проте, для моделі динаміки використаємо ширину інтервалу у 3% від виміряного значення показника кислотності, беручи до уваги зміни кислотності в часі.

У цій таблиці наведено результати експериментальних досліджень характеристик та параметрів завантаження БГУ за грудень 2023 року. При цьому кожна дискрета означає період тривалістю дві доби. За початкові умови обираємо, орієнтуючись на факт максимального заданого порядку різницевого рівняння:

$$\hat{y}_0^- \in [y_0^-; y_0^+], \hat{y}_1^- \in [y_1^-; y_1^+], \hat{y}_2^- \in [y_2^-; y_2^+], \hat{y}_3^- \in [y_3^-; y_3^+]. \quad (2.31)$$

Варто зазначити, що ці умови змінюються залежно від обраного порядку різницевого рівняння конкретної моделі-претендента. Так, якщо структура моделі претендента відповідає різницевому рівнянню другого порядку, то початкові умови матимуть такий вигляд:



$$\hat{y}_0^- \in [y_0^-; y_0^+], \hat{y}_1^- \in [y_1^-; y_1^+].$$

Або у випадку моделі –претендента у вигляді різницевого рівняння третього порядку:

$$\hat{y}_0^- \in [y_0^-; y_0^+], \hat{y}_1^- \in [y_1^-; y_1^+], \hat{y}_2^- \in [y_2^-; y_2^+].$$

За результатами структурної ідентифікації на підставі алгоритму, наведеного в праці [63, 119] (і з урахуванням напрацювань, отриманими при побудові моделі (2.29)), для моделі динаміки показника кислотності рН ферментаційного середовища було визначено таку структуру різницевого рівняння:

$$\hat{y}_k = \hat{\beta}_0 + \hat{\beta}_1 \cdot \hat{y}_{k-1} + \hat{\beta}_2 \cdot \hat{y}_{k-2} + \hat{\beta}_4 \cdot \frac{x_{2,k-2}}{x_{1,k-2}}, \quad k = 4, \dots, K, \quad (2.32)$$

де  $\frac{x_{2,k-2}}{x_{1,k-2}}$  – відношення маси завантаженого жому з цукрових буряків до об'єму завантаженої барди на  $k - 2$  кроці.

Як бачимо, ключовим елементом в структурі отриманої моделі, як і у випадку моделі (2.29), є відношення маси сухої компоненти біосировини до об'єму рідкої компоненти.

Застосування вище описаного методу до ідентифікації інтервальної моделі динаміки показника кислотності рН ферментаційного середовища дало можливість обчислити оцінки параметрів математичної моделі (2.32) у вигляді:

$$\vec{\hat{\beta}}^T = (19,5659; -0,3154; -1,4121, 0,0462).$$

Тепер підставимо отримані результати ідентифікації у вираз (2.32):

$$\hat{y}_k = 19,5659 - 0,3154 \cdot \hat{y}_{k-1} - 1,4121 \cdot \hat{y}_{k-2} + 0,0462 \cdot \frac{x_{2,k-2}}{x_{1,k-2}}, k = 4, \dots, K. \quad (2.33)$$

Отримане різницеве рівняння у вигляді виразу (2.32) описує модель динаміки показника кислотності рН ферментаційного середовища.

На рисунку 2.2 наведено результати обчислень, наведені в інтерфейсному вікні розробленого програмного забезпечення.

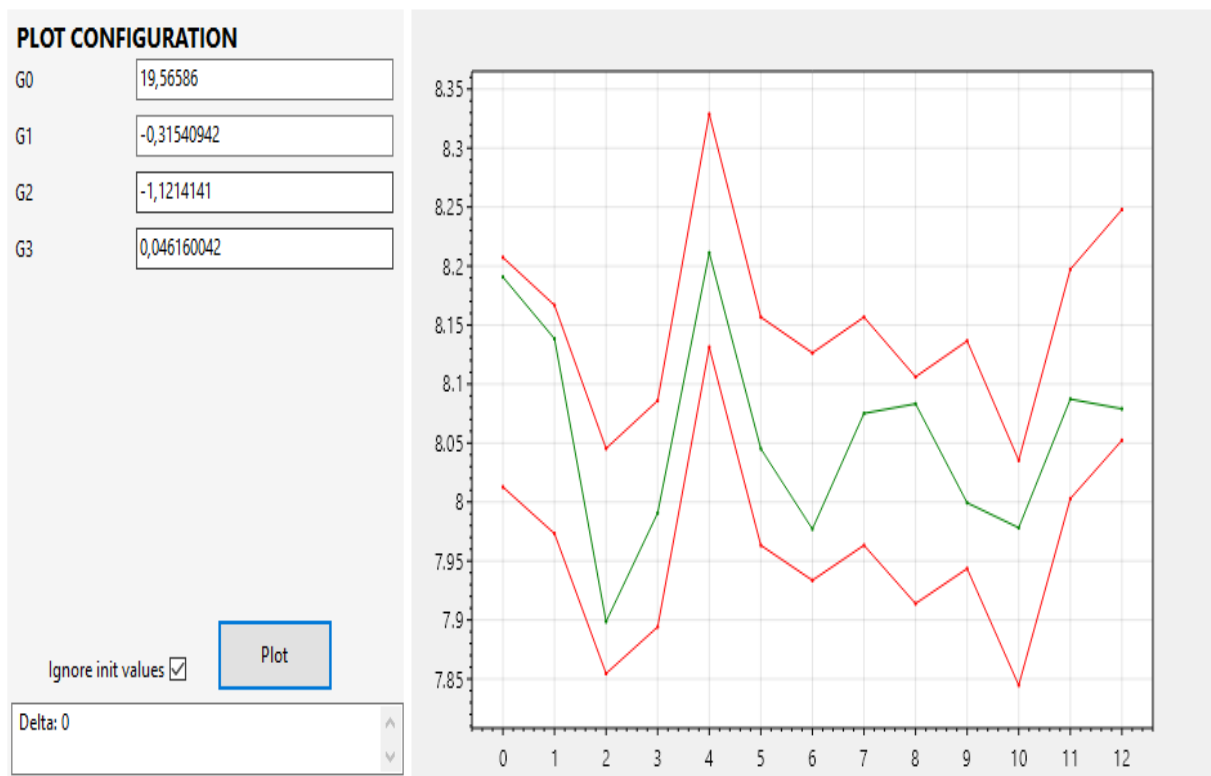


Рис. 2.2. Результати обчислень показника кислотності рН ферментаційного середовища в інтерфейсному вікні

На рисунку 2.3 наведено динаміку показника кислотності рН ферментаційного середовища. На цьому ж рисунку відображено результати вимірювань кислотності рН ферментаційного середовища в розрізі часу анаеробного бродіння з урахуванням похибок. Як бачимо, прогнозовані значення показника кислотності рН ферментаційного середовища у кожний дискретний момент часу належать до інтервалів, вимірних експериментально. Це означає, що

різницевого рівняння (2.32) адекватно описує зазначений процес із заданою точністю. Також із рисунку 2.2 можемо зробити висновок, що у цьому випадку забезпечується дотримання кислотності субстрату в БГУ в допустимих межах.

Проте, фізичні властивості процесу зручно досліджувати на підставі диференціальних рівнянь.

До того ж, для використання отриманого різницевого рівняння з метою моделювання різних режимів функціонування БГУ зручного його перетворити до диференціального рівняння. Для цього виконаємо ряд перетворень.

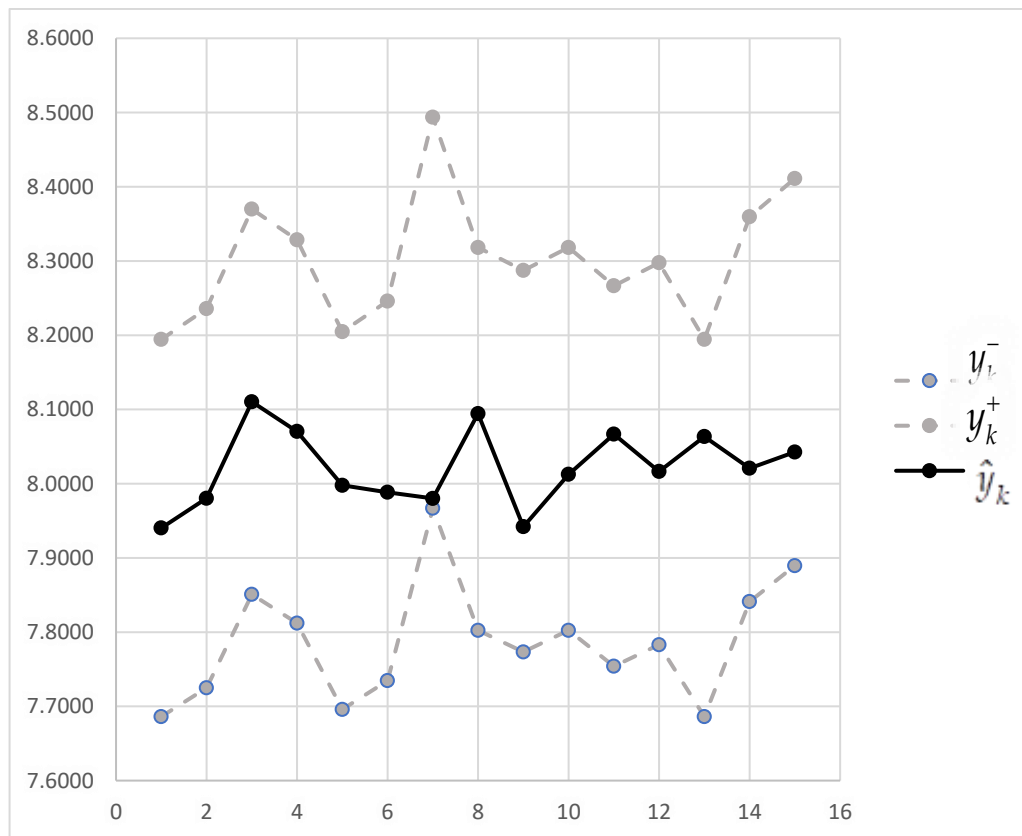


Рис. 2.3. Динаміка показника кислотності рН ферментаційного середовища

Оскільки окремі реалізації побудованої моделі є достатньо нерегулярними, для побудови диференціального рівняння використовуємо апроксимацію дискретних значень із зміщеними індексами застосуванням розкладу в ряд Тейлора до похідних першого порядку:

$$\hat{y}_{k-1} \approx \hat{y}_k - hy'_k, \quad (2.34)$$

$$\hat{y}_{k-2} \approx \hat{y}_k - 2hy'_k. \quad (2.35)$$

Апроксимуємо зміщення в індексах за допомогою розкладу в ряд Тейлора і застосуванням аргументів зі зміщенням. Для апроксимації дробово-раціональної функції використання розкладів у ряд Тейлора призводить до громіздких подань, тому змодельюємо зміщення в індексах її компонент за допомогою запізнюючих аргументів [99-101]. В результаті вираз

$$\frac{x_{2,k-2}}{x_{1,k-2}}$$

замінюємо таким виразом:

$$\frac{x_{2,k-2}}{x_{1,k-2}} \approx \frac{x_2(t_k-2h)}{x_1(t_k-2h)}. \quad (2.36)$$

Перепозначаємо момент часу  $t_k$  через  $t$  та використавши наведені вище наближення, отримуємо таке диференціальне рівняння:

$$y(t) = \hat{\beta}_0 + \hat{\beta}_1 \cdot y(t) - \hat{\beta}_1 \cdot hy'(t) + \hat{\beta}_2 \cdot y(t) - 2h\hat{\beta}_2 y'(t) + \hat{\beta}_3 \cdot \frac{x_2(t-2h)}{x_1(t-2h)}. \quad (2.37)$$

Звівши подібні члени, отримуємо остаточно диференціальне рівняння, наближений аналог до різницевого (2.33):

$$(\widehat{\beta}_1 + 2 \cdot \hat{\beta}_2)hy'(t) = (\hat{\beta}_1 + \hat{\beta}_2 - 1) \cdot y(t) + \hat{\beta}_0 + \hat{\beta}_3 \cdot \frac{x_2(t-2h)}{x_1(t-2h)}. \quad (2.38)$$

Перетворюємо диференціальне рівняння до форми, зручної для застосування чисельних методів

$$y'(t) = \frac{(\hat{\beta}_1 + \hat{\beta}_2 - 1)}{(\widehat{\beta}_1 + 2 \cdot \hat{\beta}_2)h} \cdot y(t) + \frac{\hat{\beta}_0}{(\widehat{\beta}_1 + 2 \cdot \hat{\beta}_2)h} + \frac{\hat{\beta}_3}{(\widehat{\beta}_1 + 2 \cdot \hat{\beta}_2)h} \cdot \frac{x_2(t-2h)}{x_1(t-2h)}. \quad (2.39)$$

Для розв'язання наведеного диференціального рівняння зручно застосувати модуль `odeint` із Python бібліотеки `scipy.integrate` [19, 35, 50, 54, 101-104] або інші програмно-інтерпретовані середовища [11-17, 38, 87]. Побудована модель у вигляді диференціального рівняння вимагає дослідження своєї адекватності. Результати здійснення такого роду аналізу наведені на рисунку 2.4 та в таблиці 2.2.

Таблиця 2.2

Результати моделювання динаміки показника кислотності рН ферментаційного середовища в біохімічному реакторі та їх зіставлення

$t$	$y_k^-$	$y_k^+$	$\bar{y}$	$x_1$	$x_2$	$\hat{y}_k$	$del$	$dy$	$d2y$	$y_{diff}$	$del_{equiv}$
1	7,6862	8,1944	7,94	130	129	7,94	0,00%	-0,04	0,09	7,940	0,00%
2	7,7249	8,2357	7,98	70	159	7,98	0,00%	-0,13	-0,17	7,980	0,00%
3	7,8508	8,3698	8,11	80	147	8,11	0,00%	0,04	-0,08	8,110	0,00%
4	7,8121	8,3285	8,07	130	120	8,07	0,00%	0,12	0,16	8,070	0,00%
5	7,6959	8,2047	7,95	200	101	7,99776	0,58%	-0,04	0,20	7,985	0,42%
6	7,7346	8,2460	7,99	150	97,5	7,98842	0,02%	-0,24	-0,41	7,953	0,45%
7	7,9669	8,4937	8,23	30	102	7,98013	3,04%	0,17	0,14	7,954	3,31%
8	7,8024	8,3182	8,06	110	112,5	8,09443	0,41%	0,03	0,06	7,958	1,61%
9	7,7733	8,2873	8,03	320	69	7,94205	1,07%	-0,03	-0,08	7,928	1,50%
10	7,8024	8,3182	8,06	120	100	8,01252	0,58%	0,05	0,08	7,907	1,73%
11	7,7540	8,2666	8,01	130	0	8,06671	0,69%	-0,03	-0,13	7,918	0,90%
12	7,7830	8,2976	8,04	210	37,5	8,01646	0,29%	0,10	0,26	7,936	1,04%
13	7,6862	8,1944	7,94	220	51	8,06340	1,50%	-0,16	-0,11	7,955	0,47%
14	7,8411	8,3595	8,10	40	33	8,02071	0,97%	-0,05		7,979	1,19%
15	7,8895	8,4111	8,15	220	137,5	8,04265	1,31%			8,002	1,80%
		MAX=	8,23							MAX=	3,31%

На рисунку 2.4 наведено результати спостережень за кислотністю середовища 3% інтервалу допустимих відхилень. На рисунку також подані значення модельованої характеристики, отримані за допомогою ідентифікованої на основі експериментальних інтервальних даних різницевої схеми. В подальшому, на основі отриманої різницевої схеми, згідно описаної методики побудовано диференціальне рівняння, яке є її наближеним аналогом. Також на цьому рисунку наведено чисельний розв'язок отриманого диференціального рівняння.

Аналіз графіків показує достатньо високу точність моделювання як за допомогою різницевої схеми (максимальна відносна похибка 3%), так і за допомогою диференціального рівняння (максимальна відносна похибка 3.3%). Співставні величини похибок моделей свідчать про адекватне подання за допомогою диференціального рівняння запропонованої різницевої схеми.

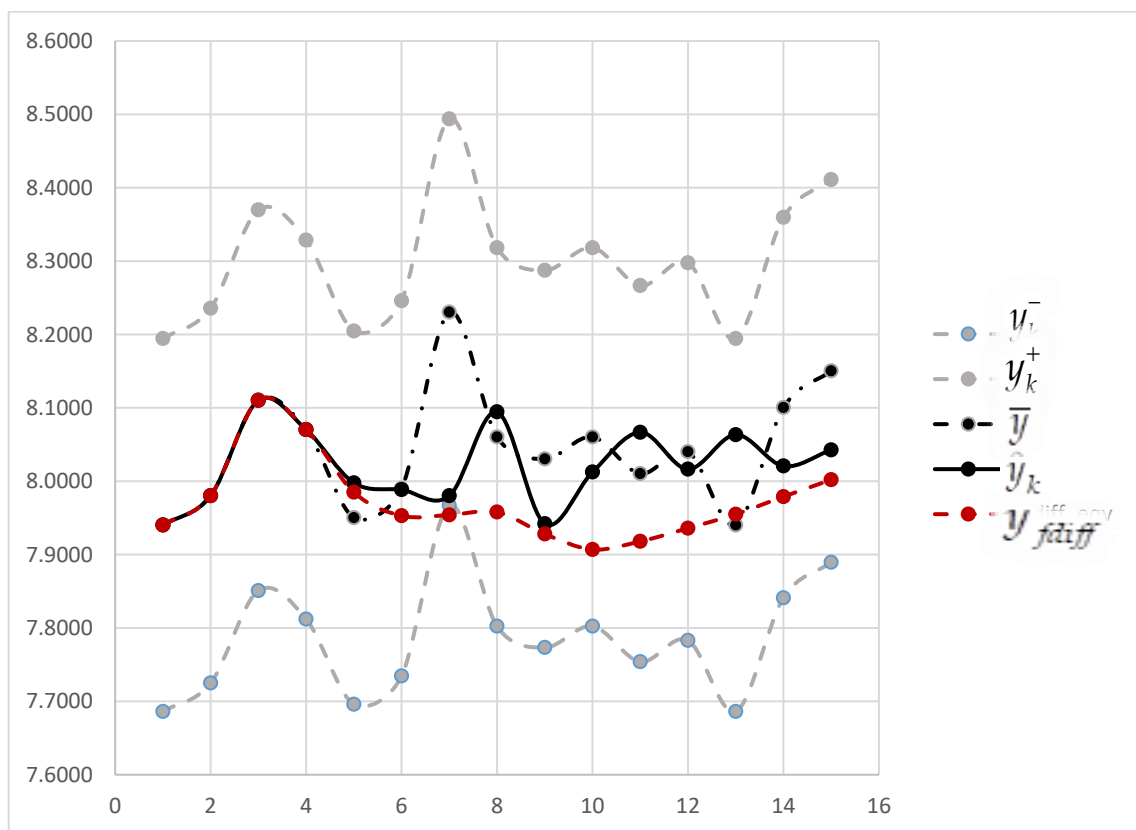


Рис. 2.4. Зіставлення результатів моделювання динаміки показника кислотності рН ферментаційного середовища в біохімічному реакторі

Візуально можна спостерігати краще наближення модельованого показника кислотності рН ферментаційного середовища за допомогою різницевого співвідношення (середня відносна похибка 0,7%) ніж за допомогою диференціального рівняння (середня відносна похибка 0,96%), хоча значення середніх відносних похибок доволі близькі. Натомість розв'язок диференціального рівняння має значно вищу гладкість.

## **Висновки до розділу 2**

1. Підвищення ефективності функціонування виробництва біогазу у спосіб переробки органічних відходів життєдіяльності людини та відходів тваринництва і агропромислового комплексу вимагає розробки нових підходів управління процесами в біогазових установках на основі застосування інформаційних технологій та методів математичного моделювання. Показано, що одною із визначальних характеристик процесу метанового бродіння в БГУ є показник рН ферментаційного середовища, тобто його кислотність. Запропоновано та обґрунтовано застосування методів математичного моделювання з метою відображення взаємозв'язку між основною характеристикою процесу та чинниками, які на неї впливають так і динаміки основної характеристики процесу, яка визначається кислотністю субстрату у біореакторі.

2. Для побудови математичних моделей обох видів запропоновано використати методи параметричної ідентифікації моделей статичних об'єктів та дискретних моделей динаміки об'єктів на основі аналізу інтервальних даних. Такі математичні моделі характеризуються «гарантованими» прогностичними властивостями. Для їх ідентифікації вперше сформульовано спільну оптимізаційну задачу і запропоновано та обґрунтовано метод її розв'язування, який ґрунтується на метаевристичних алгоритмах оптимізації і який симулює ройовий алгоритм штучної бджолиної колонії. Обидві отримані інтервальні математичні моделі можуть бути застосовані для управління процесами в біогазових установках.

3. Вперше, при ідентифікації математичної моделі динаміки показника кислотності субстрату у біореакторі запропоновано та обґрунтовано двоетапний підхід, який ґрунтується на розв'язуванні, спочатку простішої задачі ідентифікації математичної моделі взаємозв'язку між основною характеристикою процесу та чинниками, які на неї впливають з метою визначення значущих структурних елементів, і на цій основі, на другому етапі - генерування набору структурних елементів для структурної ідентифікації математичної моделі динаміки показника кислотності субстрату у біореакторі. Такий підхід забезпечує зниження обчислювальної складності реалізації методу ідентифікації математичної моделі динаміки показника кислотності субстрату у біореакторі.

4. Вперше при побудові математичної моделі динаміки показника кислотності субстрату у біореакторі запропоновано та обґрунтовано диференціальне рівняння, яке на відміну від існуючих підходів, отримано на підставі інтервальних різницевих рівнянь- результату ідентифікації моделі на підставі інтервальних даних. Запропонований підхід ґрунтується: на застосуванні формул розкладу значень шуканої функції у вузлах різницевої сітки в ряд Тейлора в околі базового вузла та різницевого представлення похідних в цьому ж околі; на застосуванні апроксимації управляючих функцій у спосіб моделювання зміщення в індексах її компонент за допомогою запізнюючих аргументів, значення яких отримано під час експерименту на певних дискретах. Такий підхід дав можливість поєднати переваги індуктивного та дедуктивного підходів для моделювання характеристик процесу в БГУ.

5. На тестових прикладах, які стосуються моделювання динаміки показника кислотності субстрату у біореакторі підтверджено ефективність та практична значущість методу. При побудові цієї інтервальної моделі (на основі отриманого диференціального рівняння) похибка розв'язку склала 3,3%, що співмірно із похибкою прогнозу, побудованого на підставі різницевої схеми. Також встановлено, що ця похибка залежить від точності апроксимації керуючої функції.



## **РОЗДІЛ 3**

### **АРХІТЕКТУРА ПРОГРАМНОЇ СИСТЕМИ ДЛЯ МОДЕЛЮВАННЯ ПРОЦЕСІВ У БІОГАЗОВИХ УСТАНОВКАХ**

У розділі описано особливості побудови архітектури програмного забезпечення для моделювання процесів у біогазових установках.

Представлено технологію створення програмної системи з використанням інтеграції підсистеми математичного моделювання та відповідної підсистеми збереження та накопичування результатів.

Описано особливості побудови архітектури програмного забезпечення для математичного моделювання процесів у біогазових установках. Наведено основні аспекти реалізації програмного забезпечення для моделювання процесів у біогазових установках, особливістю якого є програмна інтерпретація інтервальних дискретних моделей та відкрита сервісно-орієнтована архітектура, що забезпечує їх інтеграцію в існуючі системи управління біогазовими установками та дозволяє в реальному часі управляти біохімічними процесами і тим самим підвищити ефективність функціонування біогазових установок. Приведено ряд діаграм, які ілюструють особливості програмної реалізації середовища для моделювання процесів у біогазових установках. У кінці розділу описано особливості реалізації підсистеми зберігання та аналізу інформації.

Основні результати цього розділу опубліковані в працях [85, 86, 121, 123, 124].

#### **3.1 Обґрунтування використання інформаційних технологій для забезпечення допустимих значень кислотності багатокomпонентного субстрату у біогазових установках**

Для ефективного функціонування біогазових установок та підвищення обсягів виробленого біогазу необхідно вирішити дві задачі: забезпечення неперервного та максимального завантаження біосировини у біореактор та

налаштування параметрів функціонування БГУ залежно від типу сировини.

Обидві ці задачі можна вирішити за допомогою експертного середовища та неперервного контролю стану субстрату у БГУ, зокрема його температури та рівня рН. Якщо усунути зазначені контрольні заходи, то це призведе до швидкого закисання субстрату і відповідно до зниження темпів вироблення біогазу або ж навіть до повної зупинки цього процесу. Іншою проблемою при виробництві біогазу є наявність у його складі сполук сірки та вуглекислого газу, що суттєво знижує його якість. Зазначені проблеми суттєво знижують ефективність функціонування БГУ, збільшують термін їх окупності тим самим перешкоджають формуванню альтернативної енергетичної системи України із відновлювальних джерел енергії [124].

Вказані проблеми пропонується вирішити у спосіб розробки математичного та програмного забезпечення, як для моделювання процесів у БГУ, так і для управління ними, що у сукупності дає можливість створити прототип БГУ з підвищеною ефективністю функціонування.

Математичне моделювання відіграє ключову роль у забезпеченні ефективності управління процесами в біогазових установках. Це дозволяє аналізувати та прогнозувати різноманітні аспекти функціонування системи виробництва біогазу, такі як температурні умови, швидкість реакцій, витрати ресурсів і т.д. (рисунок 3.1).

Математичні моделі дозволяють оптимізувати процеси переробки органічних відходів, враховуючи різноманітні умови та параметри, що впливають на продуктивність біогазової установки. Вони допомагають у виборі оптимальних стратегій керування, максимізуючи виходи біогазу при мінімальних витратах та енергозатратах. Таким чином, математичне моделювання сприяє підвищенню загальної ефективності виробництва біогазу у біогазових установках.

Також необхідно відзначити основні проблеми, які виникають в процесі інтеграції програмних систем в процесі управління біогазовими установками, та виробити основні варіанти їх вирішення.

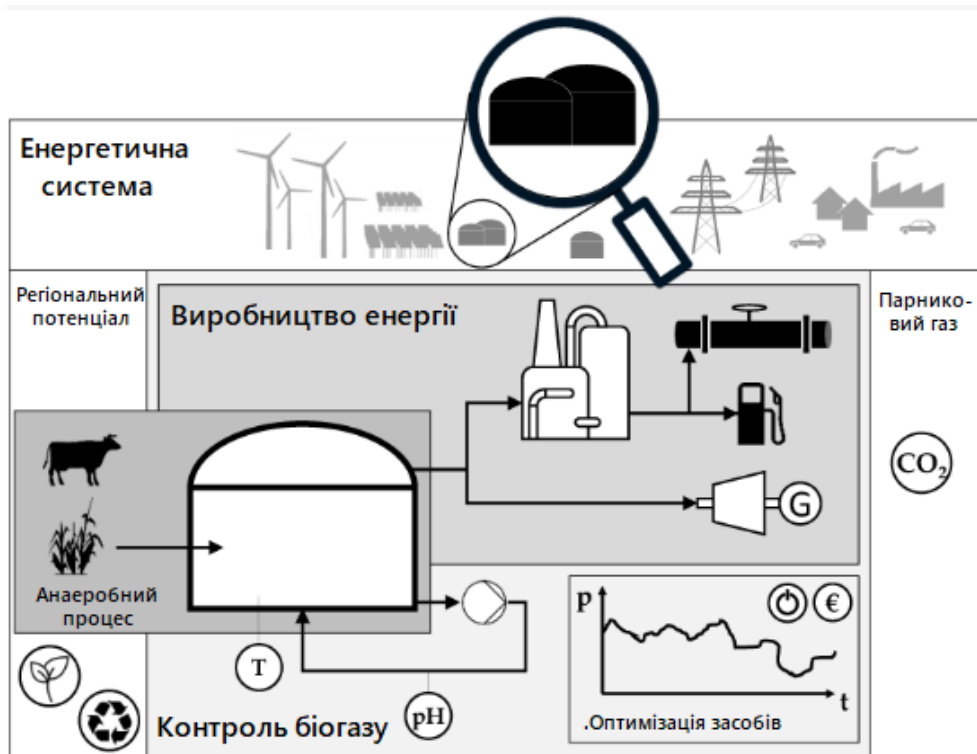


Рис. 3.1. Схема функціонування біогазової установки в обмеженнях енергетичної системи [43, 124]

Загальна схема із описом таких проблем та варіантів їх вирішення представлена на рисунку 3.2, яку розглянемо більш детально.

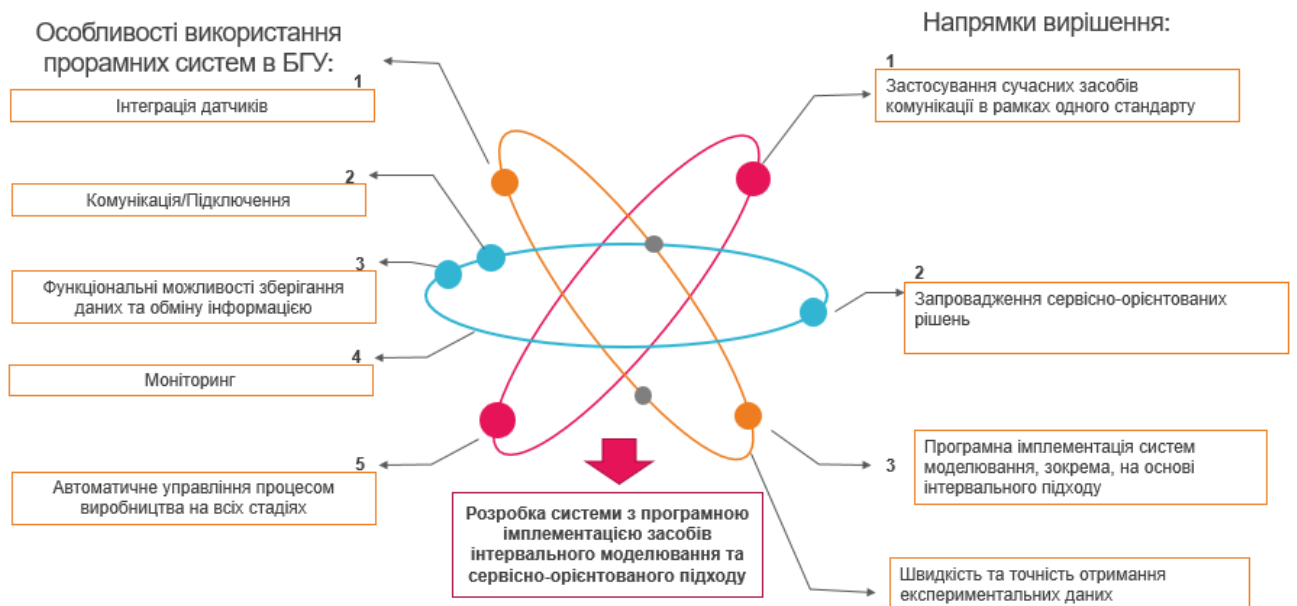


Рис. 3.2. Проблеми інтеграції програмних систем в процесі управління біогазовими установками та напрямки їх вирішення

Однією з ключових проблем є сумісність датчиків із програмним забезпеченням, яка вимагає, щоб використані датчики були сумісними з програмним забезпеченням для управління біогазовою установкою. Це може вимагати розробки або внесення змін у програмне забезпечення для забезпечення коректної роботи з відповідними датчиками. Правильне калібрування й налаштування датчиків є важливим етапом для забезпечення точності та надійності даних, які вони надають. Це може стати проблемою, особливо якщо датчики потребують індивідуального налаштування або вимагають спеціалізованої експертизи та залучення висококваліфікованих експертів. На рисунку 3.3. представлено проблеми використання датчиків в системі управління біогазовою установкою.

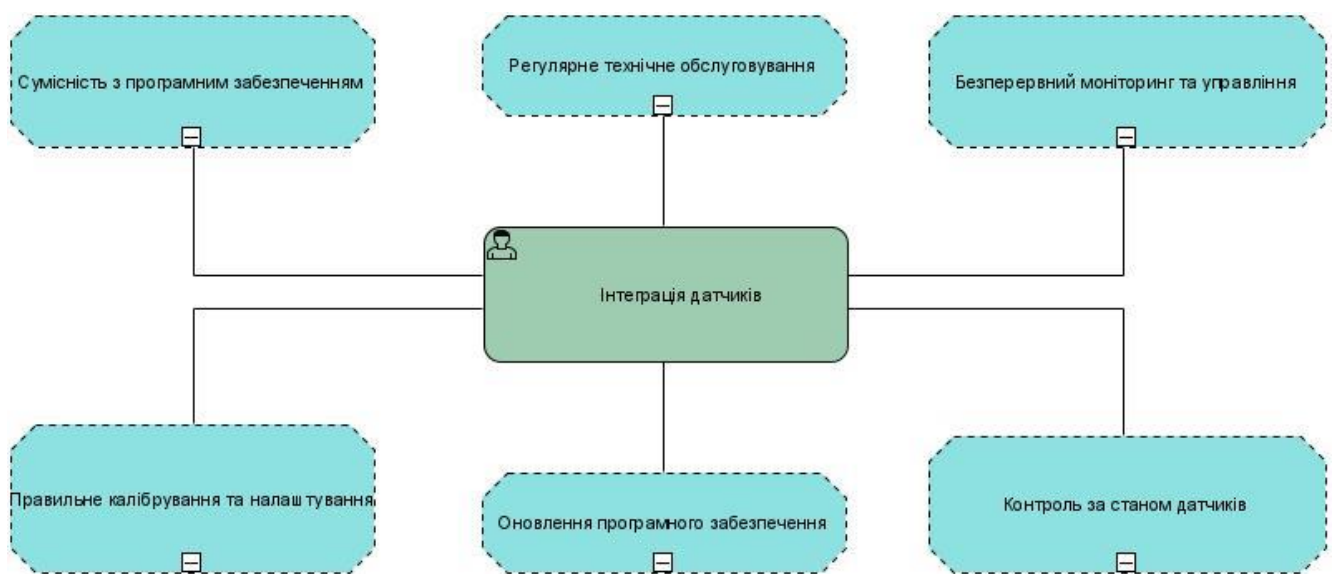


Рис. 3.3. Проблеми використання датчиків в системі управління біогазовою установкою

Справжнім викликом у цьому контексті є забезпечення безперервного моніторингу та управління процесами в реальному часі. Дані, що надходять від датчиків, потрібно ефективно обробляти та інтегрувати у програмні системи для подальшого аналізу та управління процесами в біогазовій установці. Це вимагає наявності потужної обчислювальної інфраструктури і можливості швидко

реагувати на будь-які зміни у процесах виробництва біогазу. Недоліки або затримки у збиранні та обробці даних можуть призвести до зниження ефективності та стабільності роботи установки.

Крім того, варто зазначити, що постійний контроль за станом датчиків і програмного забезпечення є критично важливим для попередження можливих збоїв і порушення роботи всієї системи. Регулярне технічне обслуговування і своєчасне оновлення програмного забезпечення здатні забезпечити високу надійність і продуктивність біогазової установки.

Ще однією важливою складовою є навчання персоналу для роботи з новими технологіями та програмними системами. Співробітники повинні бути достатньо кваліфікованими, щоб ефективно працювати з датчиками та програмним забезпеченням, а також швидко реагувати на будь-які відхилення в роботі системи.

Загалом, для досягнення успішної роботи біогазової установки необхідно забезпечити повну сумісність датчиків із програмним забезпеченням, належне калібрування та налаштування, ефективну обробку та інтеграцію даних, а також постійний контроль та навчання персоналу [85, 86]. Це комплексне завдання, яке вимагає злагодженої роботи багатьох елементів і наявності висококваліфікованих фахівців.

Окремим блоком варто виділити проблеми, які стосуються комунікації та підключення, які представлено на рисунку 3.4. Дуже часто виникають проблеми через несумісність протоколів зв'язку між програмними системами та обладнанням біогазових установок. Це вимагає ретельного вибору або розробки спеціального програмного забезпечення, яке б забезпечувало взаємодію між різними компонентами системи. У випадках існування різних протоколів зв'язку, може виникати необхідність у створенні проміжного програмного забезпечення або перетворення даних для забезпечення безперебійної взаємодії між системами.

Непередбачувані відмови мережі або втрати пакетних даних можуть значно вплинути на доступність та надійність комунікації між програмними системами та біогазовими установками. Це може призвести до затримок у передачі даних,

помилки у моніторингу та управлінні процесами, а також до зниження загальної ефективності роботи системи. Відсутність якісного зв'язку може стати серйозною перешкодою для ефективного управління та моніторингу процесів.

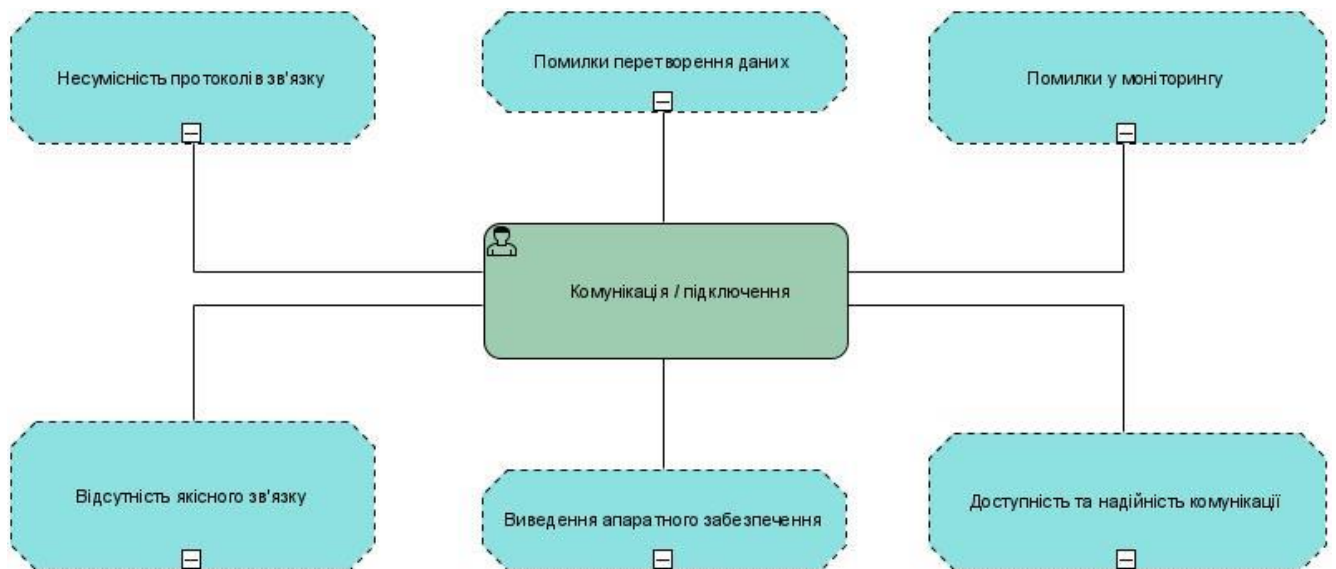


Рис. 3.4. Проблеми комунікації та підключення підсистем

Дані проблеми можуть потребувати реалізації резервних комунікаційних систем та впровадження заходів для забезпечення стійкості і надійності мережевої інфраструктури. Наприклад, використання дублюючих каналів передачі даних, встановлення додаткових засобів захисту від збоїв та втрат, а також розробка алгоритмів для автоматичного відновлення комунікацій після відмов. Це дозволить зменшити ризики, пов'язані з відмовами мережі, та забезпечити стабільну роботу біогазової установки навіть у разі виникнення непередбачуваних проблем.

Для вирішення комунікаційних проблем важливим є належне планування та проектування мережевої інфраструктури, а також тестування всіх можливих сценаріїв збоїв та їх впливу на систему. Це допоможе знизити ризики та забезпечити надійну взаємодію між усіма елементами біогазової установки.

Загалом, вирішення проблем, пов'язаних з комунікацією та підключенням, є важливим аспектом забезпечення ефективної та безперебійної роботи біогазових установок, що вимагатиме координації зусиль між розробниками програмного

забезпечення, постачальниками обладнання та інженерами.

З огляду на збільшення кількості кібератак на промислові системи, забезпечення захисту від несанкціонованого доступу до програмних систем управління біогазовими установками є критичним завданням, рисунок 3.5. Необхідно вживати заходів для забезпечення конфіденційності, цілісності та доступності даних.

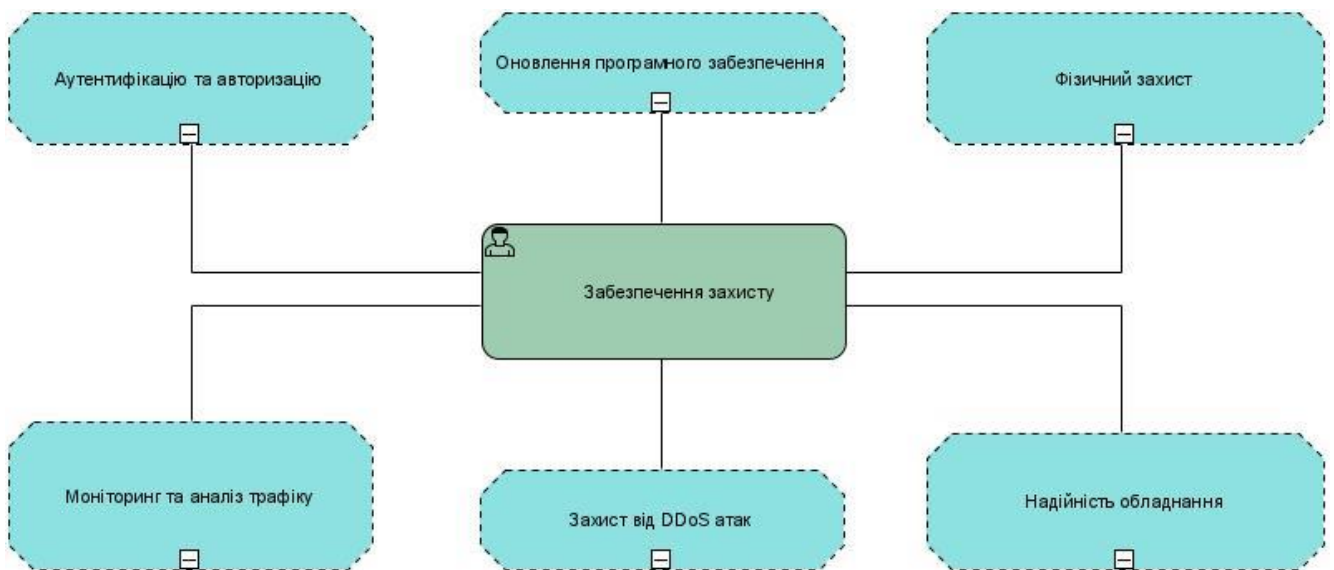


Рис. 3.5. Проблеми забезпечення захисту від несанкціонованого доступу

Для захисту промислових систем важливо впровадити комплексну стратегію кібербезпеки, яка включає:

- аутентифікацію та авторизацію користувачів, тобто забезпечення доступу тільки авторизованим користувачам за допомогою багаторівневої аутентифікації (наприклад, двофакторна аутентифікація);
- шифрування даних, яке включає використання сильних методів шифрування для захисту даних під час їх передачі та зберігання;
- моніторинг та аналіз трафіку, який включає постійний моніторинг мережевого трафіку та аналіз для виявлення підозрілої активності та можливих атак.
- регулярне оновлення програмного забезпечення для усунення відомих

вразливостей;

- захист від DDoS атак, що включає використання технологій для запобігання та мінімізації впливу розподілених атак на відмову в обслуговуванні (DDoS).

Складність інтеграції існуючих систем може бути досить великою через різноманітність технологій, що використовуються в різних частинах біогазових установок, рисунок 3.6. Ці технології можуть включати різні протоколи комунікації, інтерфейси та програмні компоненти, що ускладнює взаємодію між ними.

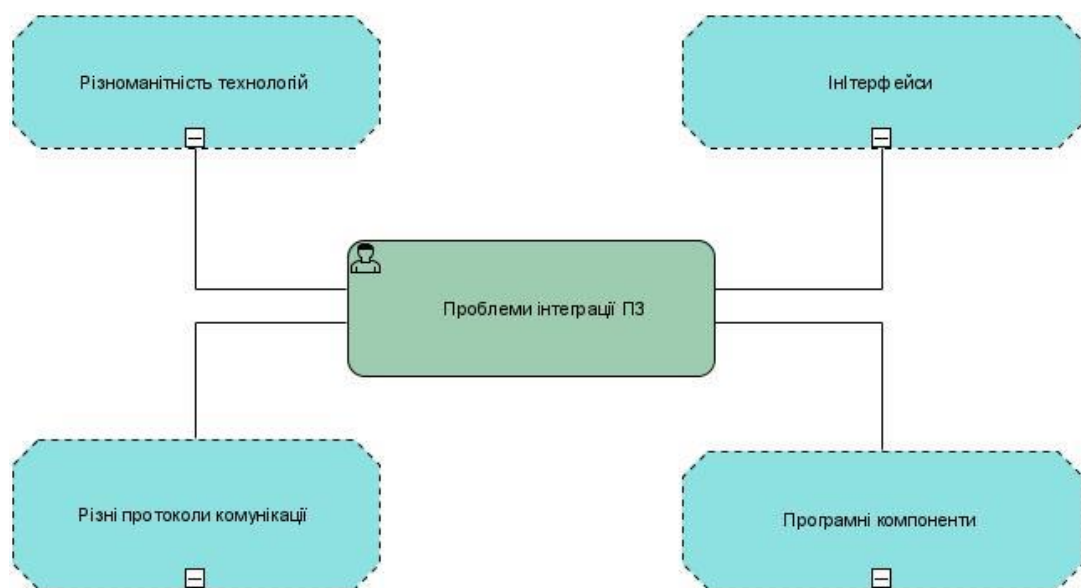


Рис. 3.6. Проблеми інтеграції існуючих систем в БГУ

Для подолання цих складнощів може виникнути необхідність розробки та впровадження стандартизованих протоколів комунікації та інтерфейсів. Ці стандарти дозволять забезпечити сумісність та взаємодію між різними компонентами системи, спрощуючи процес їх інтеграції та управління.

Основні кроки для покращення інтеграції систем:

- розробка стандартизованих протоколів, впровадження єдиних протоколів комунікації для взаємодії між різними компонентами біогазової установки;
- використання проміжного програмного забезпечення, тобто розробка



спеціалізованого програмного забезпечення для перетворення та передачі даних між системами, які використовують різні технології;

- уніфікація інтерфейсів, яка включає впровадження стандартизованих інтерфейсів для забезпечення взаємодії між різними компонентами та системами.

- тестування та перевірка, яке включає регулярне тестування системи на сумісність та функціональність для забезпечення надійної інтеграції всіх компонентів.

Враховуючи важливість безпеки та інтеграції, необхідно продумано підходити до проектування та впровадження програмних систем управління біогазовими установками.

Особливо необхідно відзначити основні аспекти, які пов'язані з автоматичним управлінням процесами виробництва біогазу на всіх стадіях, рисунок 3.7. Це включає в себе комплексний підхід до контролю та оптимізації кожного етапу процесу, забезпечуючи ефективність та стабільність роботи біогазової установки.

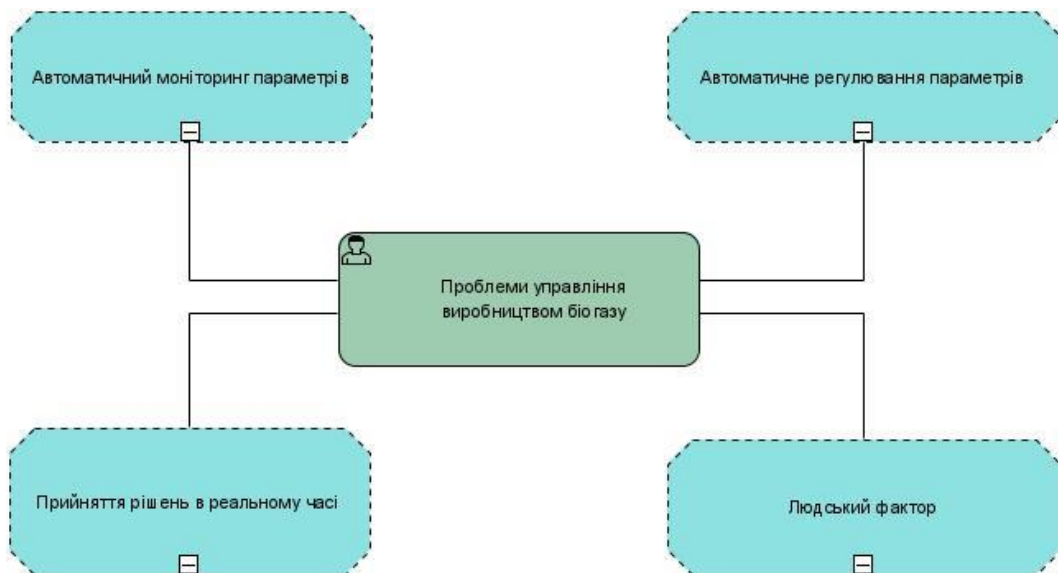


Рис. 3.7. Проблеми управління виробництвом біогазу на усіх стадіях

Автоматичний моніторинг параметрів. Програмні системи можуть використовуватися для автоматичного моніторингу різних параметрів виробництва

біогазу, таких як:

- температура: контроль температури є критичним для забезпечення оптимальних умов для анаеробного розщеплення органічної речовини;
- тиск: забезпечення стабільного тиску в системі дозволяє уникнути аварійних ситуацій та оптимізувати процеси газовиділення;
- кислотність (pH): підтримка відповідного рівня рН є важливою для підтримки активності мікроорганізмів, які виробляють біогаз;
- рівень метану та інших газів: моніторинг концентрацій газів для оцінки якості та кількості виробленого біогазу.
- вміст сухих речовин та інших хімічних показників: для оцінки ефективності процесу розщеплення біомаси.

Реальний час та прийняття рішень. Завдяки автоматичному моніторингу, операторам доступна інформація про стан процесу в реальному часі, що дозволяє:

- швидко реагувати на непротокольні ситуації, такі як відхилення параметрів від нормальних значень;
- приймати обґрунтовані рішення для запобігання аварійним ситуаціям або для корекції виробничого процесу.

Автоматичне регулювання параметрів. На основі отриманих даних, програмні системи можуть автоматично регулювати параметри процесу виробництва біогазу, включаючи:

- зміну режимів роботи обладнання: автоматичне налаштування роботи насосів, мішалок, нагрівальних елементів та іншого обладнання для підтримки оптимальних умов процесу;
- налаштування обсягу залученої сировини: регулювання подачі біомаси для забезпечення стабільного ходу процесу та уникнення перевантажень;
- додавання реагентів: автоматичне дозування хімічних реагентів, необхідних для стабілізації процесу та підтримки оптимального рівня рН або інших параметрів.

Переваги автоматизації. Автоматизація процесів виробництва біогазу має низку переваг, включаючи:

- підвищення ефективності: оптимізація параметрів процесу дозволяє максимізувати вихід біогазу при мінімальних витратах ресурсів;
- зниження ризиків: автоматичний моніторинг та регулювання параметрів допомагають уникнути аварійних ситуацій та знизити ризик пошкодження обладнання.
- зменшення людського фактору: скорочення необхідності постійного ручного втручання зменшує можливість помилок операторів та підвищує рівень безпеки;
- підвищення якості: автоматичне підтримання стабільних умов процесу забезпечує високоякісний вихідний продукт.

Застосування методів математичного моделювання у програмних системах відіграє важливу роль у прогнозуванні та оптимізації процесу виробництва біогазу, рисунок 3.8.

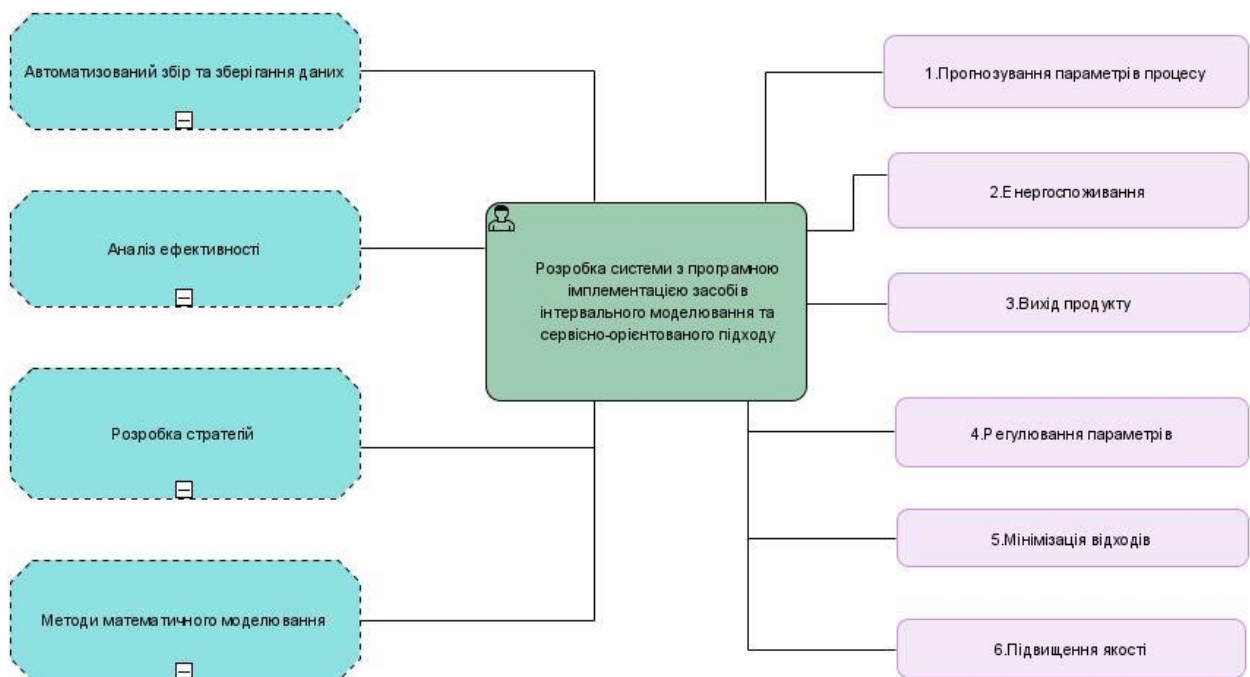


Рис. 3.8. Основні цілі розробки та використання системи з програмною імплементацією засобів інтервального моделювання та сервісно-орієнтованого підходу

Використання таких методів дозволяє досягти кількох значних цілей:

1. Прогнозування параметрів процесу: математичні моделі дозволяють прогнозувати поведінку системи під різними умовами, допомагаючи передбачити можливі результати та вплив змін у процесі.

2. Енергоспоживання: оптимізація параметрів для зменшення енергозатрат на всіх стадіях виробництва.

3. Вихід продукту: прогнозування обсягів виробництва біогазу на основі змінних параметрів і вхідних даних.

4. Регулювання параметрів: автоматичне коригування параметрів для досягнення максимального виходу біогазу та мінімальних витрат ресурсів.

5. Мінімізація відходів: оптимізація технологічного процесу для зменшення кількості відходів.

6. Підвищення якості: використання моделювання для покращення якості кінцевого продукту.

Автоматизований збір та зберігання даних. Програмні системи можуть автоматично збирати та зберігати великі обсяги даних про виробничий процес. Це включає:

1. Дані про параметри: збір інформації про температуру, тиск, кислотність, рівень метану, обсяги сировини тощо на всіх стадіях виробництва.

2. Історичні дані: збереження історичних даних для аналізу трендів та довгострокового планування.

3. Аналітика: накопичені дані можуть використовуватися для детального аналізу ефективності процесу та виявлення можливих шляхів покращення.

Аналіз ефективності, який включає:

1. Ідентифікацію слабких місць: аналіз даних допомагає виявляти етапи процесу, які можуть бути оптимізовані або покращені.

2. Проблемні області: виявлення потенційних проблем до їх виникнення, що дозволяє запобігти аварійним ситуаціям та оптимізувати роботу системи.

Розробка стратегій, яка включає:

1. Планування дій: на основі аналізу даних розробляються конкретні стратегії

для покращення роботи біогазової установки.

2. Постійне вдосконалення: використання отриманих результатів для постійного вдосконалення технологічного процесу та підвищення ефективності.

Переваги математичного моделювання. Застосування методів математичного моделювання у процесах виробництва біогазу має низку переваг:

1. Економія ресурсів: оптимізація використання енергії та матеріалів, що зменшує витрати.

2. Підвищення продуктивності: збільшення обсягів виробництва біогазу без додаткових витрат.

3. Поліпшення якості: забезпечення стабільної та високої якості кінцевого продукту.

4. Підвищення надійності: зменшення ризиків та забезпечення безперебійного функціонування біогазової установки.

Застосування методів математичного моделювання у програмних системах дозволяє проводити прогнозування та оптимізацію процесу виробництва біогазу. Це допомагає зменшити витрати енергії, збільшити обсяги виробництва та покращити якість кінцевого продукту. Програмні системи можуть автоматично збирати та зберігати дані про всі етапи виробництва біогазу з метою здійснення детального аналізу ефективності процесу, виявляти потенційні проблеми та розробляти стратегії для їх вирішення.

Далі у розділі розкриваються особливості проектування та реалізація програмного забезпечення для моделювання процесів у біогазових установках на основі методів аналізу інтервальних даних.

### **3.2 Архітектура програмної системи для моделювання процесів у біогазових установках**

Розглянемо основні аспекти реалізації програмної системи для моделювання процесів у біогазових установках. В основу такої системи закладено програмну

інтерпретацію інтервальних дискретних моделей та відкрита сервісно-орієнтована архітектура [119, 124].

Математичне моделювання на основа інтервального аналізу дозволяє враховувати не тільки точкові значення параметрів, а й їх діапазони зміни, що дозволяє більш точно прогнозувати та керувати процесами в біогазових установках для оптимізації виробничих процесів та підвищення ефективності. Такий підхід дозволяє враховувати невизначеність та варіабельність параметрів процесу, що може допомогти в управлінні ризиками та забезпеченні надійності роботи біогазових установок. Розробка системи з використанням сервісно-орієнтованого підходу дозволяє створювати легко масштабовані та гнучкі модулі, які можуть легко інтегруватися з існуючими технологіями та їх розширювати згідно з виникаючими потребами [124].

На рисунку 3.9 представлено узагальнену архітектуру програмного середовище для математичного моделювання процесів у біогазових установках, яка включає наступні підсистеми: підсистема моделювання процесів у біогазових установках на основі аналізу інтервальних даних та підсистема управління процесом виробництва біогазу.

Підсистема моделювання процесів у біогазових установках на основі аналізу інтервальних даних є критично важливою для забезпечення ефективного управління та оптимізації виробництва. Така підсистема включає кілька важливих компонентів, які забезпечують збір, обробку, моделювання та аналіз даних, а також інтерактивний інтерфейс користувача для управління та контролю процесів.

Підсистема інтервального моделювання включає наступні модулі:

1. Модуль обробки даних:

- збір даних: автоматичний збір інтервальних даних з датчиків та інших джерел;
- попередня обробка: фільтрація та очищення даних від шумів та некоректних записів;
- агрегація: узагальнення даних для різних часових інтервалів.

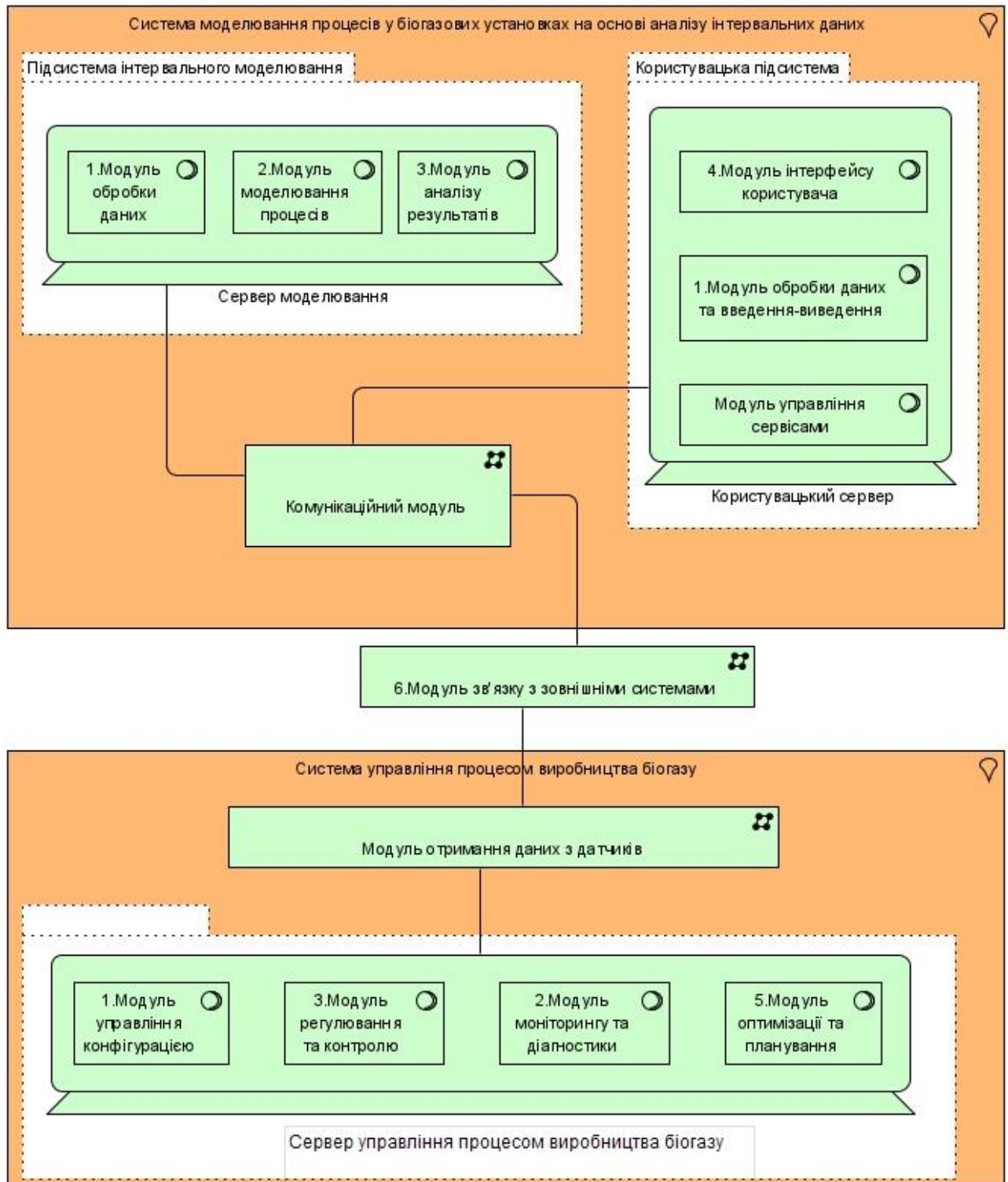


Рис. 3.9. Узагальнена архітектура програмного середовища для математичного моделювання процесів у біогазових установках

## 2. Модуль процесів моделювання:

- побудова моделей: використання інтервальних даних для створення

математичних моделей, що описують процеси виробництва біогазу;

- симуляція: проведення симуляцій для перевірки моделей під різними сценаріями роботи;
- калібрування: налаштування моделей для підвищення їх точності та відповідності реальним даним.

### 3. Модуль аналізу результатів:

- аналітика: аналіз результатів моделювання та симуляцій для виявлення тенденцій та закономірностей;
- візуалізація: побудова графіків, діаграм та інших візуальних засобів для представлення результатів аналізу;
- оцінка ефективності: порівняння результатів моделей з фактичними даними для оцінки точності та ефективності.

Користувацька підсистема включає наступні модулі:

#### 1. Модуль інтерфейсу користувача:

- інтерактивний інтерфейс: інтуїтивно зрозумілий інтерфейс для взаємодії з системою;
- налаштування: можливість налаштування параметрів моделювання та відображення результатів;
- сповіщення: повідомлення користувачів про критичні ситуації та результати аналізу.

#### 2. Модуль управління сервісами:

- управління роботою: контроль та керування роботою різних модулів підсистеми моделювання;
- планування: автоматичне планування обчислювальних задач для ефективного використання ресурсів;
- моніторинг: безперервний моніторинг стану системи та виконання завдань моделювання.

Інтеграція підсистеми інтервального моделювання з користувацькою підсистемою створює потужний інструмент для автоматизації та підвищення



ефективності виробництва біогазу.

Підсистема управління процесом виробництва біогазу включає набір модулів, які забезпечують повний цикл управління, від конфігурації і регулювання до моніторингу і оптимізації. Це дозволяє забезпечувати ефективне, безпечне і стабільне виробництво біогазу на всіх етапах. Розглянемо детальніше модулі управління процесом виробництва біогазу:

1. Модуль конфігурації обладнання:
  - ініціалізація обладнання: налаштування та запуск обладнання для роботи у визначеному режимі;
  - інтеграція компонентів: забезпечення сумісної роботи різних компонентів системи та їх налаштування;
2. Модуль конфігурації процесу виробництва біогазу:
  - налаштування параметрів: визначення оптимальних параметрів для роботи процесу, таких як температура, тиск, обсяг сировини тощо;
  - Налаштування рецептури та сценарії: зберігання і управління різними варіантами змішування та сценаріями для виробництва біогазу.
3. Модулі регулювання та контролю:
  - регулювання процесу: автоматичне регулювання використання PID-регуляторів або інших методів для підтримання стабільних умов процесу;
  - адаптивне управління: застосування алгоритмів для адаптації умов процесу в реальному часі на основі зворотного зв'язку від датчиків.
4. Модуль контролю параметрів:
  - неперервний моніторинг: постійний контроль параметрів процесу, таких як температура, тиск, рН і концентрація газів;
  - аварійні режими: виявлення і реагування на аварійні ситуації, включаючи автоматичне відключення або перехід до безпечних режимів;
5. Модулі моніторингу та діагностики:
  - моніторинг системи в реальному часі: відображення параметрів процесу в реальному часі для операторів.

- віддалений доступ: забезпечення можливості віддаленого моніторингу та управління системою;
  - виявлення несправностей: автоматичне виявлення несправностей в роботі обладнання та програмного забезпечення;
  - прогнозування відмов: використання історичних даних для прогнозування можливих відмов та попередження операторів.
6. Модулі оптимізації та планування включають:
- оптимізацію процесу: збір та аналіз даних для виявлення можливостей оптимізації виробничого процесу;
  - автоматична оптимізація: використання алгоритмів для автоматичної оптимізації параметрів процесу для зменшення енерговитрат та збільшення виходу біогазу.
  - планування виробництва: розробка графіків виробництва на основі прогнозів та потреб виробництва;
  - управління сировиною: оптимізація закупівель та використання сировини для забезпечення стабільної роботи системи.

Основними користувачами системи є (діаграма варіантів використання, яка представлена на рисунку 3.10): системний аналітик, диспетчер та адміністратор.

Системний аналітик відіграє ключову роль у забезпеченні ефективності та точності моделювання процесів у біогазових установках. Його робота включає аналіз і перетворення вимог користувачів у інструменти та функції, які можуть значно покращити виробничий процес. Цей користувач може здійснювати:

1. Аналіз вимог щодо моделювання процесів:
  - збір вимог від користувачів: взаємодія з операторами, інженерами та керівниками виробництва для збору й уточнення їх вимог та побажань щодо системи моделювання;
  - аналіз вимог: визначення критичних параметрів та умов, які повинні бути враховані в моделях для досягнення оптимальної продуктивності;
  - документування вимог: створення детальної документації по зібраним

вимогам, яка буде використовуватися для подальшої розробки й оновлення системи.

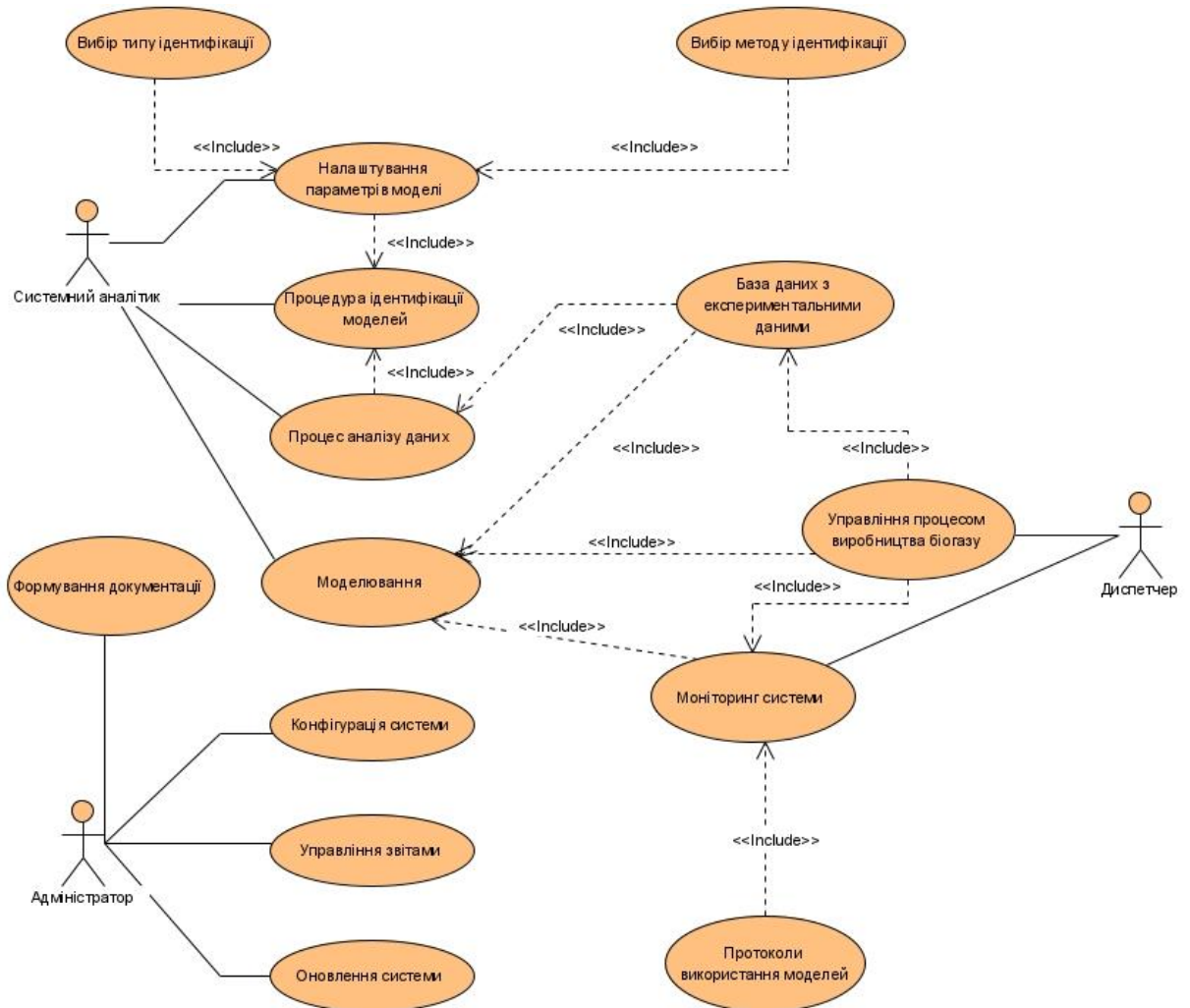


Рис. 3.10. Діаграма варіантів використання системи для математичного моделювання процесів у біогазових установках

2. Визначення потреб у нових функціях та можливостях системи моделювання:

- оцінка існуючих процесів та систем: аналіз функціональних можливостей поточних систем моделювання і виявлення їх недоліків та обмежень;
- визначення нових функцій: ідентифікація нових функцій, необхідних для покращення точності моделювання, автоматизації процесів та підвищення зручності користування;

- пропозиція щодо покращення: розробка пропозицій щодо покращення системи моделювання, включаючи впровадження нових модулів та інструментів.

### 3. Створення або оновлення моделей процесів:

- розробка нових моделей: створення нових математичних і комп'ютерних моделей для опису процесів виробництва біогазу на основі зібраних вимог та внутрішніх стандартів;

- оновлення існуючих моделей: регулярне поновлення і коригування існуючих моделей з відповідати змінам у виробництві та новим вимогам користувачів.

- калібрування та валідація: перевірка моделей на відповідність реальним даними, коригування та вдосконалення моделей для забезпечення їх точності та надійності.

### 4. Документування та комунікація:

- створення технічної документації: розробка детальної технічної документації для нових та оновлених моделей.

- комунікація з командою: надання необхідної інформації розробникам, тестувальникам та іншим членам команди для успішної реалізації та впровадження моделей в систему.

Диспетчер є однією з ключових фігур в управлінні роботою системи та забезпеченні безперебійного функціонування процесів у біогазових установках. Цей користувач може здійснювати моніторинг, контроль, реагування на відхилення та взаємодію з іншими користувачами системи.

#### 1. Моніторинг роботи системи:

- постійне спостереження за роботою системи у режимі реального часу;
- збір та аналіз даних з різних датчиків та модулів системи для оцінки поточного стану виробничих процесів;

- використання інтерфейсів для візуалізації ключових параметрів та показників роботи системи.

#### 2. Контроль поточного стану процесів та їх ефективності:

- оцінка ефективності роботи системи шляхом аналізу параметрів таких як температура, тиск, рівень кислотності та обсяг виробленого біогазу.
- відстеження продуктивності: контроль за виконанням запланованих виробничих показників та порівняння фактичних значень з цільовими;
- виявлення будь-яких відхилень від встановлених норм та стандартів виробництва.

### 3. Відстеження відхилень від стандартів та інструкцій:

- налаштування системи для автоматичного сповіщення про відхилення від стандартів або аварійні ситуації;
- заходи для виправлення відхилень вручну у випадках, коли автоматична система не може впоратися з проблемою;
- налагодження та оптимізація параметрів для повернення процесу у нормальний стан.

### 4. Реалізація взаємодії з іншими користувачами:

- координація з системним аналітиком: надання зворотного зв'язку системному аналітику про виявлені проблеми та пропозиції щодо внесення змін у моделі та параметри системи;
- спільна робота з оператором для забезпечення стабільної роботи системи та швидкого реагування на несправності;
- інформування керівництва про поточний стан системи, відхилення та заходи, які були вжиті для їх виправлення.

Системний адміністратор відіграє ключову роль у забезпеченні стабільної та ефективної роботи системи моделювання процесів у біогазових установках. Його обов'язки включають конфігурацію, налаштування, забезпечення безперебійної роботи інфраструктури, оновлення програмного забезпечення та підтримку користувачів.

### 1. Конфігурація та налаштування системи моделювання процесів:

- налаштування та конфігурація системи моделювання, включаючи встановлення програмного забезпечення та налаштування параметрів;

- забезпечення правильної конфігурації серверів, на яких працює система моделювання, включаючи налаштування безпеки, мережі та зберігання даних;

- конфігурація мережевих підключень і забезпечення стабільного зв'язку між компонентами системи;

2. Забезпечення безперебійної роботи серверів та інфраструктури системи:

- постійне відстеження стану серверів та інфраструктури для виявлення будь-яких збоїв або проблем;

- оптимізація використання серверних ресурсів для забезпечення максимальної продуктивності системи;

- налаштування та контроль за системами резервного копіювання для забезпечення збереження даних та можливості їх швидкого відновлення у разі збою;

- розробка і реалізація планів відновлення після аварій для швидкого повернення системи до нормального функціонування у разі інцидентів.

3. Відстеження змін в додатках та оновлення програмного забезпечення:

- відстеження змін у додатках та оновленнях від розробників програмного забезпечення;

- планування та впровадження оновлень програмного забезпечення для виправлення помилок, закриття вразливостей та покращення функціональності;

- попереднє тестування оновлень на тестових середовищах перед застосуванням їх у робочій системі для мінімізації ризику проблем під час впровадження.

Діаграма варіантів використання показує взаємодію користувачів з системою та функціональні можливості, які вони можуть використовувати для досягнення своїх цілей у контексті моделювання процесів у біогазових установках.

На рисунку 3.11 представлено узагальнену діаграму діяльності, яка описує основні процеси й рішення, які виникають на етапі експлуатації програмного

комплексу для моделювання процесів у біогазових установках. Діаграма включає три ключові компоненти: управління процесом виробництва біогазу, інтерфейс для моделювання та база даних експериментальних досліджень. Розглянемо їх більш детально.

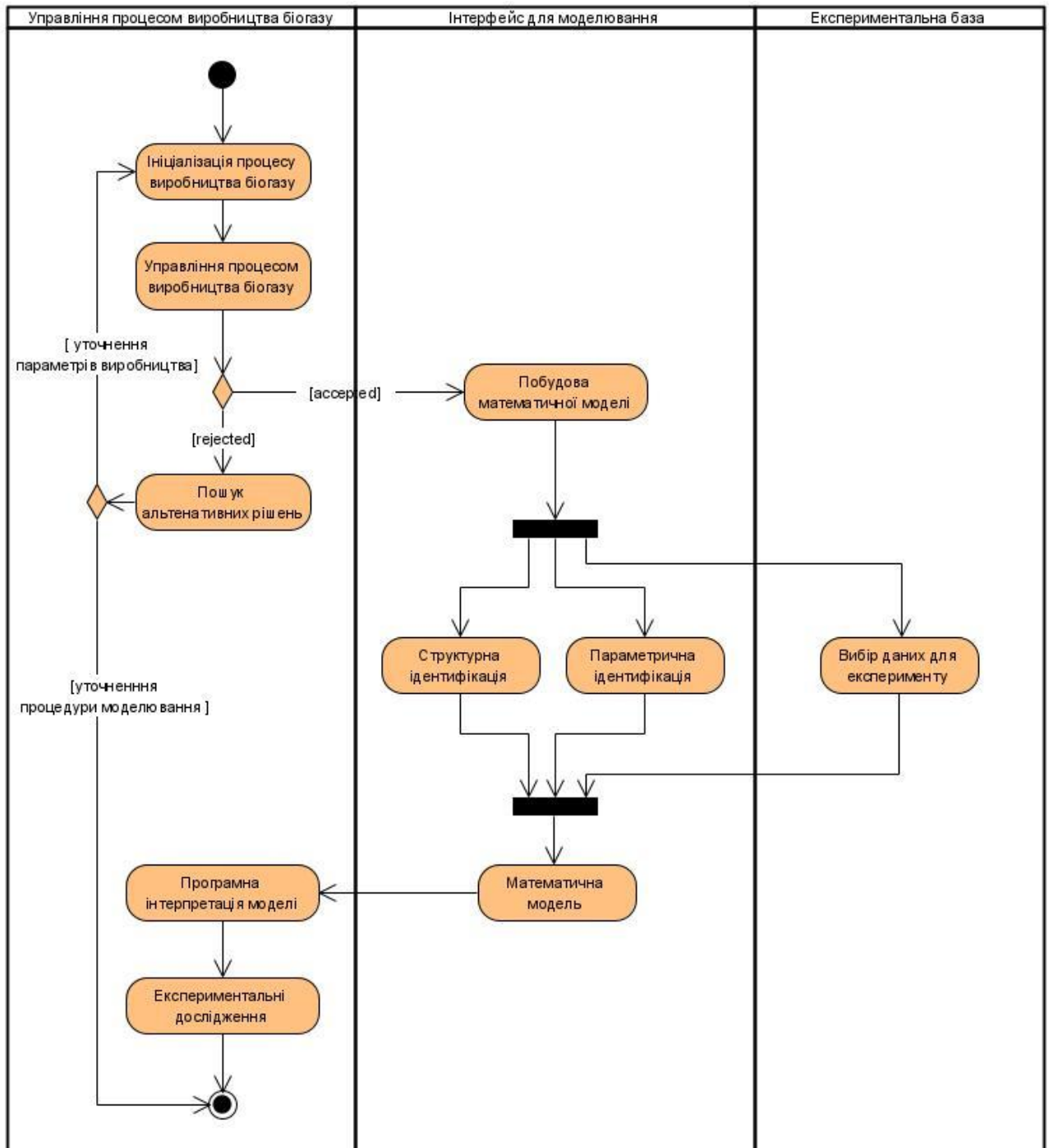


Рис. 3.11. Діаграма послідовності побудови інтервальної моделі

1. Управління процесом виробництва біогазу включає наступні основні завдання:

- моніторинг і контроль параметрів (збір даних про параметри процесу в режимі реального часу (температура, тиск, рН, обсяг сировини тощо);
- індикатори стану (відстеження ключових показників для забезпечення стабільної роботи);
- оцінка продуктивності (аналіз продуктивності виробництва на основі зібраних даних);
- виявлення відхилень ( ідентифікація відхилень від запланованих параметрів і стандартів);
- автоматичне регулювання (використання зворотного зв'язку для автоматичного корегування параметрів);
- застосування ручного втручання у випадках, коли автоматичні корекції неефективні.

2. Інтерфейс для моделювання включає наступні основні складові:

- побудова математичної моделі (структурна ідентифікація: визначення структури моделі, що відображає реальні процеси та параметрична ідентифікація: налаштування параметрів моделі для точного відображення процесів);
- програмна інтерпретація моделі (розробка програмного коду, який реалізує створену математичну модель);
- валідація (тестування програмної моделі для перевірки її відповідності реальним даним):
- симуляція сценаріїв (виконання експериментальних досліджень для перевірки різних сценаріїв роботи біогазової установки);
- аналіз результатів (оцінка результатів симуляцій для виявлення ефективності та надійності моделі).

3. База даних експериментальних досліджень включає реалізацію основних функцій:

- зберігання даних ( збереження даних, отриманих під час експериментальних



досліджень;

- архівування історичних даних для подальшого аналізу і побудови моделей;
- виявлення трендів і закономірностей на основі збережених даних;
- аналіз якості експериментальних результатів та їх відповідності моделі;
- використання даних для прийняття обґрунтованих рішень щодо корегування або оптимізації процесів;
- використання бази для вдосконалення математичних моделей.

На рисунку 3.12 представлена діаграма послідовності побудови інтервальної моделі опису процесів у біогазових установках. Ця діаграма відображає етапи створення моделі, інтеграцію даних та застосування моделі для прогнозування та аналізу.

Етапи побудови інтервальної моделі включають наступну послідовність кроків:

#### 1. Збір початкових даних:

- збір даних про вхідні параметри процесу, такі як обсяг сировини, температура, тиск, кислотність та інші показники;
- використання різних датчиків та систем для збирання інформації в реальному часі.

#### 2. Обробка та фільтрація даних:

- очищення даних від шумів та некоректних записів;
- видалення нерелевантних або помилкових даних для забезпечення точності моделювання.

#### 3. Аналіз інтервальних даних:

- узагальнення даних для різних варіантів інтервальних представлень;
- визначення мінімальних, максимальних та середніх значень для кожного інтервалу.

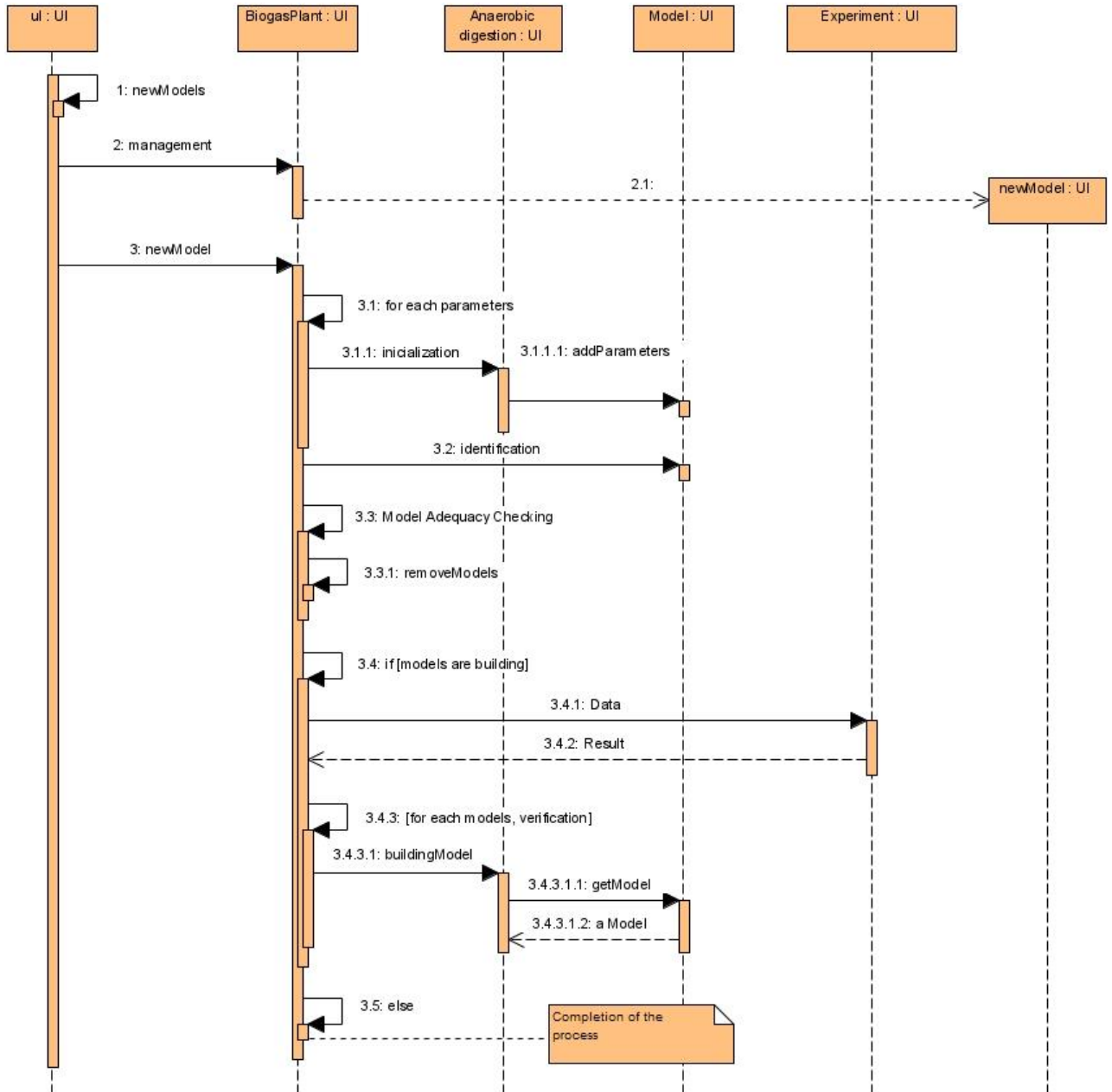


Рис. 3.12. Діаграма послідовності побудови інтервальної моделі

#### 4. Побудова інтервальної моделі:

- використання інтервальних даних для створення математичних моделей, які описують процеси у біогазовій установці;
- налаштування моделі на основі реальних даних для підвищення точності прогнозів.

#### 5. Валідація та тестування моделі:

- тестування моделі на історичних даних для перевірки її адекватності та

точності;

- внесення необхідних коригувань у модель на основі результатів тестування.

6. Програмна інтерпретація інтервальної моделі, яка дозволяє використовувати її для різних завдань, пов'язаних з управлінням та аналізом процесів у біогазових установках.

Програмну інтерпретацію отриманої інтервальної моделі можна використовувати для прогнозування та аналізу різних сценаріїв розвитку подій у біогазовій установці.

### **3.3 Підсистема аналізу та зберігання даних**

Успішна реалізація програмної системи для моделювання процесів у біогазових установках вимагає детального проектування та впровадження підсистеми аналізу та зберігання інформації. Для досягнення цієї мети було обрано систему управління базами даних MySQL. MySQL відомий своєю високою продуктивністю, завдяки оптимізованим алгоритмам обробки запитів, планувальникам і кешуванню, що дозволяє ефективно працювати з великими обсягами даних і високонавантаженими операціями.

Крім того, інтеграція з іншими продуктами робить СУБД MySQL дуже зручним у використанні. Він гармонійно поєднується з іншими технологіями, такими як Jython, Apache Tomcat Servlet, Java Virtual Machine, що спрощує процес розробки, розгортання та аналізу даних у межах розробки та використання системи моделювання процесів у біогазових установках. Система MySQL відзначається своєю надійністю і стабільністю завдяки вбудованим механізмам відновлення, резервного копіювання та відміни операцій. Це дозволяє запобігти можливим втратам даних або пошкодженню бази даних під час роботи з ними.

На рисунку 3.13 наведено ER діаграму бази даних, яка використовується в процесі моделювання процесів у біогазових установках.

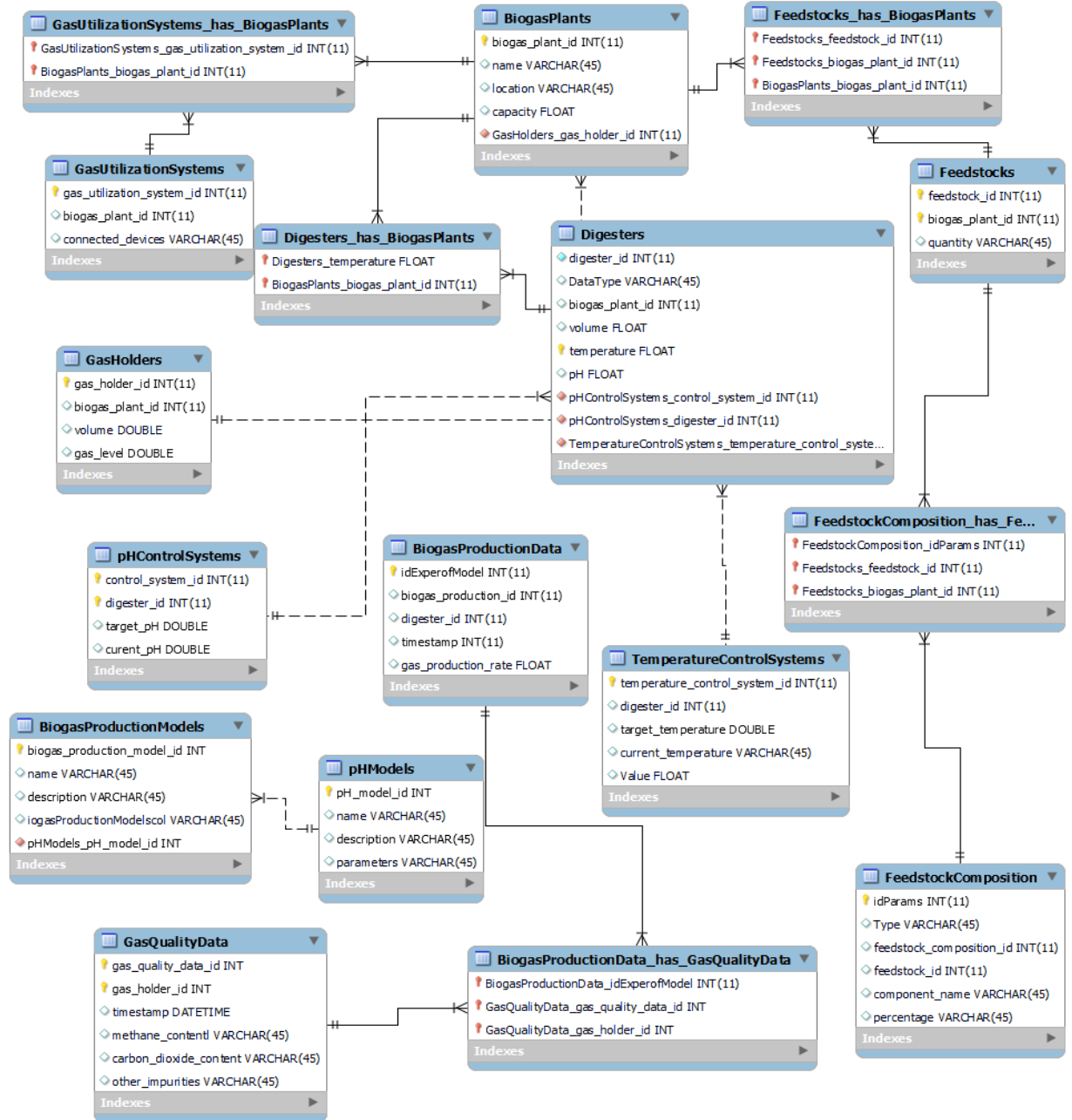


Рис. 3.13. ER-діаграма бази даних для моделювання процесів у біогазових установках

Для реалізації основної функціональності програмної системи було створено як ключові, так і додаткові відношення, що відображають структуру бази даних. Атрибути, які містяться в таблицях, відображають базові характеристики виділених сутностей, а їх типи даних та розмірність були обрані на основі фізичної інтерпретації цих характеристик. Це допомагає ефективно зберігати та обробляти інформацію, необхідну для роботи програмної системи.

Структура відношення “biogasplants” зображена на рисунку 3.14 і відображає інформацію про біогазову установку, її найменування, локацію та розміщення, вихідну потужність та деякі інші атрибути, які необхідні для збереження інформації про біогазову установку.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
•biogas_plant_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
◇•name	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
◇•location	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
◇•capacity	DOUBLE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.14. Структура відношення «biogasplants»

Для належного розуміння структури підсистеми аналізу та зберігання інформації є важливим мати відповідні зв'язки між сутностями предметної області та їх відображення в описах конкретної системи управління базами даних (СУБД). Для досягнення цієї мети використовується мова опису даних (DDL - Data Definition Language). DDL визначає метадані бази даних та використовується для створення, зміни та видалення структурних об'єктів бази даних. У Додатку А представлено DDL спроектованої бази даних для моделювання процесів у біогазових установках.

Таблиця "Digesters" (рисунок 3.15) моделюється переліком наступних атрибутів:

- digester\_id (PK): унікальний ідентифікатор;
- biogas\_plant\_id (FK): зовнішній ключ для зв'язку з біогазовою установкою;
- volume: об'єм;
- temperature: поточна температура;
- pressure: поточний тиск.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
•digester_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
•biogas_plant_id	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•volume	DOUBLE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•temperature	DOUBLE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•pressure	DOUBLE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.15. Структура відношення «Digesters»

Таблиця "Feedstocks" (рисунок 3.16) моделюється переліком наступних атрибутів:

- feedstock\_id (PK): унікальний ідентифікатор сировини;
- biogas\_plant\_id (FK): зовнішній ключ для пов'язання з установкою;
- type: тип сировини (наприклад, органічні відходи, сільськогосподарські залишки);
- quantity: кількість сировини.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
idFeedstocks	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
•biogas_plant_id	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•type	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•quantity	DOUBLE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.16. Структура відношення «Feedstocks»

Таблиця "GasHolders" (рисунок 3.17) моделюється переліком наступних атрибутів:

- gas\_holder\_id (PK): унікальний ідентифікатор резервуара для біогазу;
- biogas\_plant\_id (FK): зовнішній ключ для зв'язку з установкою;
- volume: об'єм резервуара;
- gas\_level: поточний рівень біогазу.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
•gas_holder_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
•biogas_plant_id	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•volume	DOUBLE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•gas_level	DOUBLE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.17. Структура відношення «GasHolders»

Таблиця "GasUtilizationSystems" (рисунок 3.18) моделюється переліком наступних атрибутів:

- gas\_utilization\_system\_id (PK): унікальний ідентифікатор системи використання біогазу;
- biogas\_plant\_id (FK): зовнішній ключ для зв'язку з установкою;
- connected\_devices: список підключених пристроїв.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
•gas_utilization_system_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
•biogas_plant_id	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•connected_devices	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.18. Структура відношення «GasUtilizationSystems»

Таблиця "TemperatureControlSystems" (рисунок 3.19) моделюється переліком наступних атрибутів:

- temperature\_control\_system\_id (PK): унікальний ідентифікатор системи контролю температури;
- digester\_id (FK): зовнішній ключ для зв'язку із змішувачем;

- target\_temperature: цільова температура;
- current\_temperature: поточна температура.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
temperature_control_syste...	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
digester_id	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
target_temperature	DOUBLE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
current_temperature	DOUBLE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.19. Структура відношення «TemperatureControlSystems»

Таблиця "PressureControlSystems" (рисунок 3.20) моделюється переліком наступних атрибутів:

- pressure\_control\_system\_id (PK): унікальний ідентифікатор системи контролю тиску;
- digester\_id (FK): зовнішній ключ для зв'язку із змішувачем;
- target\_pressure: цільовий тиск;
- current\_pressure: поточний тиск.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
pressure_control_system_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
digester_id	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
target_pressure	DOUBLE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
current_pressure	DOUBLE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.20. Структура відношення «PressureControlSystems»

Іншу група відношень у спроектованій базі даних використовуються для



збереження результатів, які ідентифікуються в процесі математичного моделювання процесів у біогазових установках. Розглянемо структуру цих відношень більш детально.

Таблиця "BiogasProductionData" (рисунок 3.21) моделюється переліком наступних атрибутів:

- biogas\_production\_id (PK): унікальний ідентифікатор даних виробництва біогазу;
- digester\_id (FK): зовнішній ключ для зв'язку із змішувачем;
- timestamp: час запису даних;
- gas\_production\_rate: швидкість виробництва біогазу.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
•biogas_production_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
◆digester_id	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
◆timestamp	DATETIME	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
◆gas_production_rate	DECIMAL(10,0)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.21. Структура відношення «BiogasProductionData»

Таблиця "TemperatureData" (рисунок 3.22) моделюється переліком наступних атрибутів:

- temperature\_data\_id (PK): унікальний ідентифікатор даних температури;
- digester\_id (FK): зовнішній ключ для зв'язку із змішувачем;
- timestamp: час запису даних;
- temperature: значення температури.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
•temperature_data_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
◇•digester_id	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
◇•timestamp	DATE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
◇•temperature	FLOAT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.22. Структура відношення «BiogasProductionData»

Таблиця "PressureData" (рисунок 3.23) моделюється переліком наступних атрибутів:

- pressure\_data\_id (PK): унікальний ідентифікатор даних тиску;
- digester\_id (FK): зовнішній ключ для зв'язку із змішувачем;
- timestamp: час запису даних;
- pressure: значення тиску.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
•pressure_data_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
◇•digester_id	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
◇•timestamp	DATE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
◇•pressure	DOUBLE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.23. Структура відношення «PressureData»

Таблиця "FeedstockComposition" (рисунок 3.24) моделюється переліком наступних атрибутів:

- feedstock\_composition\_id (PK): унікальний ідентифікатор складу сировини;
- feedstock\_id (FK): зовнішній ключ для зв'язку із сировиною;

- component\_name: назва компоненту сировини;
- percentage: відсоток складу компоненту.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
•feedstock_composition_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
•feedstock_id	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•component_name	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•percentage	DECIMAL(10,0)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.24. Структура відношення «FeedstockComposition»

Таблиця "GasQualityData" (рисунок 3.25) моделюється переліком наступних атрибутів:

- gas\_quality\_data\_id (PK): унікальний ідентифікатор даних якості біогазу;
- gas\_holder\_id (FK): зовнішній ключ для зв'язку з резервуаром біогазу;
- timestamp: час запису даних;
- methane\_content: вміст метану в біогазі;
- carbon\_dioxide\_content: вміст діоксиду вуглецю в біогазі;
- other\_impurities: інші домішки в біогазі.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
•gas_quality_data_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
•gas_holder_id	INT(11)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
•timestamp	DATE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•methane_content	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•carbon_dioxide_content	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•other_impurities	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.25. Структура відношення «GasQualityData»

Для здійснення процедури математичного моделювання, програмної інтерпретації математичних моделей, а також для зберігання моделей, які моделюють процеси в біогазових установках використовуються наступні відношення, які розглянемо більш детально.

Таблиця "BiogasProductionModels" (рисунок 3.26) моделюється переліком наступних атрибутів:

- biogas\_production\_model\_id (PK): унікальний ідентифікатор моделі виробництва біогазу;
- name: назва моделі;
- description: опис моделі;
- parameters: параметри моделі (наприклад, коефіцієнти моделі, константи).

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
•biogas_production_model_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
•name	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•description	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•parameters:	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.26. Структура відношення «BiogasProductionModels»

Таблиця "TemperatureModels" (рисунок 3.27) моделюється переліком наступних атрибутів:

- temperature\_model\_id (PK): унікальний ідентифікатор моделі температури;
- name: назва моделі;
- description: опис моделі;
- parameters: параметри моделі.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
•temperature_model_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
•name	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•description	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•parameters	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.27. Структура відношення «TemperatureModels»

Таблиця "PressureModels" (рисунок 3.28) моделюється переліком наступних атрибутів:

- pressure\_model\_id (PK): унікальний ідентифікатор моделі тиску;
- name: назва моделі;
- description: опис моделі;
- parameters: параметри моделі.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
•pressure_model_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
•name	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•description	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•parameters	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.28. Структура відношення «PressureModels»

Таблиця "FeedstockModels" (рисунок 3.29) моделюється переліком наступних атрибутів:

- feedstock\_model\_id (PK): унікальний ідентифікатор моделі сировини;
- name: назва моделі;
- description: опис моделі;

- parameters: параметри моделі.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
•feedstock_model_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
•name	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•description	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•parameters	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.29. Структура відношення «FeedstockModels»

Таблиця "GasQualityModels" (рисунок 3.30) моделюється переліком наступних атрибутів:

- gas\_quality\_model\_id (PK): унікальний ідентифікатор моделі якості біогазу;
- name: назва моделі;
- description: опис моделі;
- parameters: параметри моделі.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
•gas_quality_model_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
•name	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•description	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
•parameters	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 3.30. Структура відношення «GasQualityModels»

Інші відношення використовуються для забезпечення цілісності та повноти системи моделювання. В додатку Б представлено лістинг DDL відношень підсистеми зберігання інформації для системи математичного моделювання процесів у біогазових установках.

### Висновки до розділу 3

1. У даному розділі наведено детальний опис архітектури програмної системи, призначеної для моделювання процесів, які відбуваються в біогазових установках. Ця система складається з різних компонентів та модулів, які виконують певні завдання та взаємодіють з іншими частинами системи, забезпечуючи її коректну роботу та продуктивність. Архітектура програмної системи для моделювання біогазових установок є ключовим елементом у забезпеченні точності та надійності процесів моделювання, що допомагає вдосконалити управління біогазовими установками та оптимізувати їхню діяльність.

2. В межах реалізованої архітектури побудовано UML діаграми активності для визначення основних процесів, Use-case діаграм для опису варіантів використання системи, діаграми послідовності для узгодження послідовності операцій, діаграми діяльності для подальшої реалізації системи, діаграми комунікації для обміну даними між об'єктами, основну діаграму класів для класифікації об'єктів, діаграму компонентів системи для виділення компонентів та їх взаємодії, а також діаграми розгортання для опису фізичного розміщення компонентів. Цей підхід допоміг зрозуміти вимоги до системи для моделювання процесів у біогазових установках, сприяючи підвищенню ефективності розробки та використання математичних моделей у цій предметній області.

## РОЗДІЛ 4

### ПРОГРАМНЕ СЕРЕДОВИЩЕ ДЛЯ МОДЕЛЮВАННЯ ПРОЦЕСІВ У БІОГАЗОВИХ УСТАНОВКАХ

У розділі описано особливості програмної реалізації системи, зокрема з використанням парадигми об'єктно-орієнтованого програмування на базі відкритої сервісно-орієнтованої архітектури з використанням технології Spring framework на мові програмування Java, а також інтерпретатором Python. В якості Python інтерпретатора було обрано Jython, а в якості СУБД використовується MySQL.

Описано процеси інтеграційного тестування та тестування продуктивності середовища для моделювання процесів у біогазових установках. Здійснено опис тестування продуктивності системи та відповідно підтверджено ефективність використання середовища при моделюванні процесів у біогазових установках. Показано можливість подальшого розвитку проекту та його імплементації в діючі біогазові установки.

У кінці розділу здійснено опис графічного інтерфейсу програмного середовища для моделювання процесів у біогазових установках. Описано функціонування основних підсистем з точки зору взаємодії між собою, різними типами користувачів, а також можливість зовнішньої інтеграції в діючі біогазові установки.

Основні результати цього розділу опубліковані в працях [65, 85, 86, 123, 124].

#### **4.1 Особливості програмної реалізації середовища для моделювання процесів у біогазових установках на основі відкритої сервісно-орієнтованої архітектури**

Програмне середовище для моделювання процесів у біогазових установках розроблене на основі відкритої сервісно-орієнтованої архітектури з використанням технології Spring Framework на мові програмування Java [25, 73, 74, 92], а також



інтерпретатором Python. В якості Python інтерпретатора було обрано Jython [22, 113]. В якості СУБД використовується MySQL версії 5.7 [40, 77, 79]. Діаграма розгортання системи представлена на рисунку 4.1.

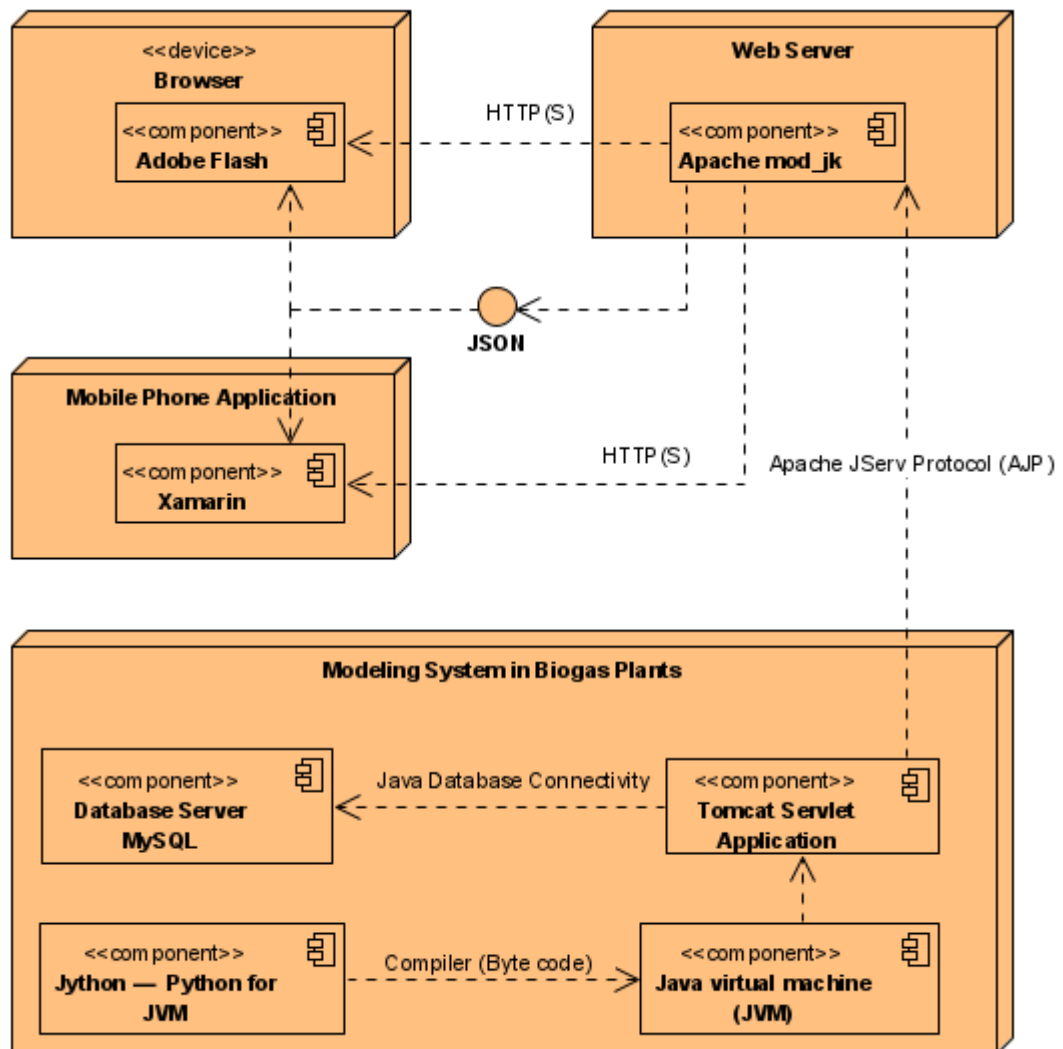


Рис. 4.1. Діаграма розгортання програмної системи для моделювання процесів у біогазових установках

Генерація інструментальних засобів для математичного моделювання здійснюється у формі окремого веб-додатку, а в якості веб-сервера, що написаний на Java, використовується Apache Tomcat. Також є можливість використання системи у формі мобільного додатку, реалізація здійснювалася з використанням Xamarin.

На рисунку 4.2. наведено діаграму класів програмної системи для

модельовання процесів у біогазових установках. Реалізація такої діаграми допомагає в структурованому проектуванні системи модельовання та полегшує комунікацію між розробниками, а це дуже важливий аспект в процесі реалізації сервісно-орієнтованої архітектури [121, 123, 124]. Розглянемо реалізацію спроектованих класів більш детально.

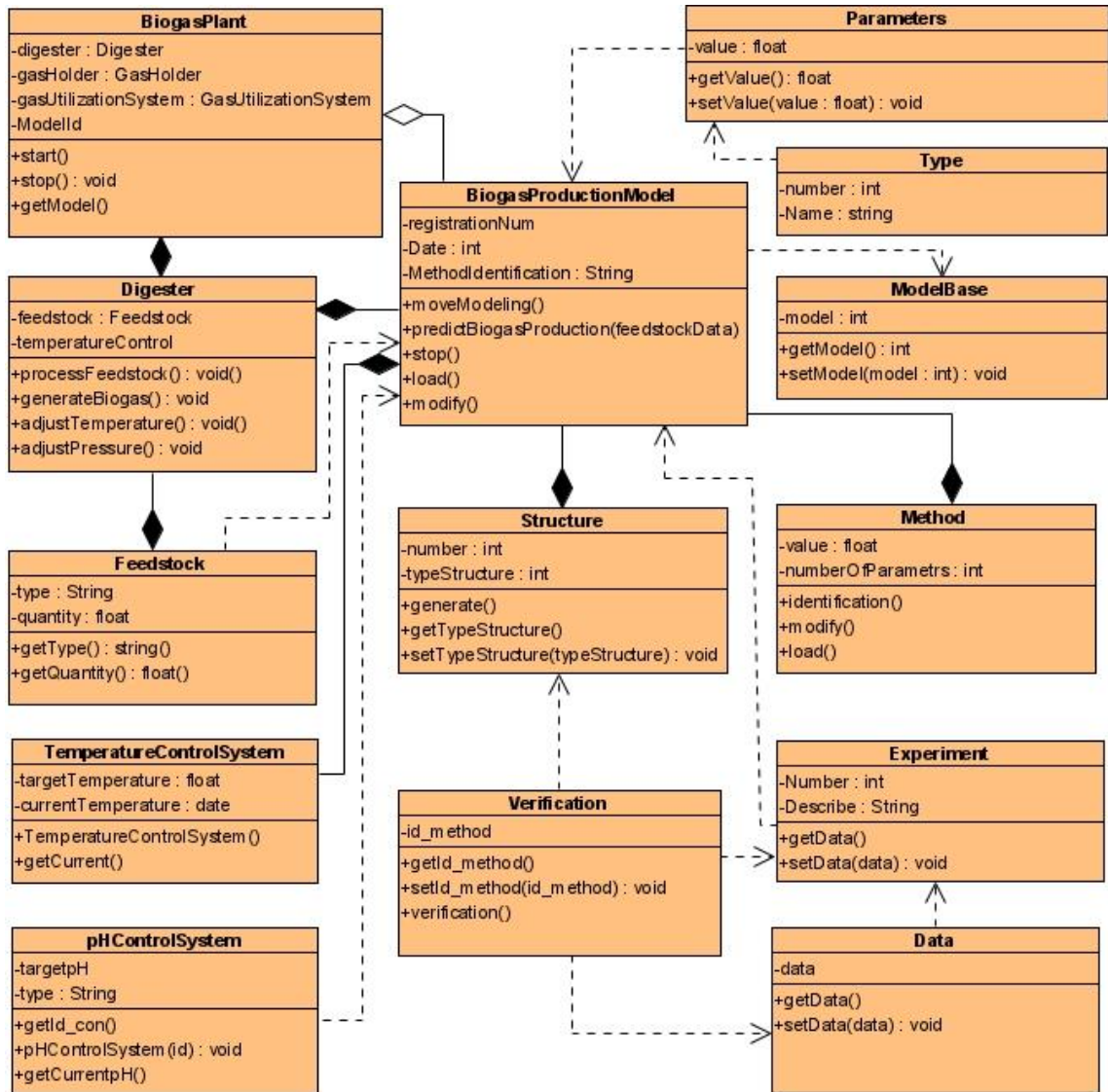


Рис. 4.2. Діаграма класів програмної системи для модельовання процесів у біогазових установках

BiogasPlant: клас, що описує біогазову установку та включає логіку відповідної інформаційної реалізації. На рисунку 4.3 представлено реалізацію цього класу в середовищі розробки NetBeans.

```

1 package biogas.plant;
2
3 public class BiogasPlant {
4
5     private final Digester digester;
6     private final GasHolder gasHolder;
7     private final GasUtilizationSystem gasUtilizationSystem;
8
9     public BiogasPlant(Digester digester, GasHolder gasHolder, GasUtilizationSystem gasUtilizationSystem)
10        {
11         this.digester = digester;
12         this.gasHolder = gasHolder;
13         this.gasUtilizationSystem = gasUtilizationSystem;
14     }
15
16     public void start() {
17         System.out.println("Starting the biogas plant...");
18         digester.start();
19         gasHolder.start();
20         gasUtilizationSystem.start();
21         System.out.println("Biogas plant started.");
22     }
23 }

```

Рис. 4.3. Реалізація класу BiogasPlant

Digester: клас, що представляє збірник для перетворення органічних матеріалів у біогаз. На рисунку 4.4. представлено реалізацію цього класу в середовищі розробки NetBeans.

```

1 package biogas.plant;
2
3 // Клас Digester (Реактор)
4 public class Digester {
5     private Feedstock feedstock;
6     private TemperatureControlSystem temperatureControlSystem;
7     private PressureControlSystem pressureControlSystem;
8
9     public Digester(Feedstock feedstock, TemperatureControlSystem temperatureControlSystem, PressureCont:
10        {
11         this.feedstock = feedstock;
12         this.temperatureControlSystem = temperatureControlSystem;
13         this.pressureControlSystem = pressureControlSystem;
14     }
15
16     public void processFeedstock() {
17         System.out.println("Processing feedstock of type: " + feedstock.getType() + " with quantity: " +
18         // Логіка для обробки сировини
19     }
20 }

```

Рис. 4.4. Реалізація класу Digester

Feedstock: клас, що описує сировину, яка використовується для виробництва біогазу (наприклад, органічні відходи). На рисунку 4.5. представлено реалізацію цього класу в середовищі розробки NetBeans.

```

1 package biogas.plant;
2 public class Feedstock {
3     private final String type; // Тип сировини, наприклад, органічні відходи
4     private final double quantity; // Кількість сировини у кілограмах
5     private final double moistureContent; // Вміст вологи у відсотках
6
7     public Feedstock(String type, double quantity, double moistureContent) {
8         this.type = type;
9         this.quantity = quantity;
10        this.moistureContent = moistureContent;
11    }
12
13    // Геттери та сеттери
14    public String getType() {
15        return type;
16    }
17
18 }

```

Рис. 4.5. Реалізація класу Feedstock

GasHolder: клас, що представляє контейнер для зберігання біогазу перед використанням. На рисунку 4.6. представлено реалізацію цього класу в середовищі розробки NetBeans.

```

1 package biogas.plant;
2 public class GasHolder {
3     private float volume; // Максимальний обсяг газу, що може бути збережений в газгольдері
4     private float currentGasLevel; // Поточний обсяг газу в газгольдері
5
6     // Конструктор для ініціалізації GasHolder
7     public GasHolder(float volume) {
8         this.volume = volume;
9         this.currentGasLevel = 0; // На початку резервуар порожній
10    }
11
12    // Метод для зберігання газу
13    public void storeGas(float volume) {
14        if (currentGasLevel + volume <= this.volume) {
15            currentGasLevel += volume;
16            System.out.println(volume + " м³ газу збережено. Поточний рівень газу: " + currentGasLevel + " ");
17        } else {
18            System.out.println("Недостатньо місця для зберігання " + volume + " м³ газу. Вільний обсяг: " +
19                (this.volume - currentGasLevel) + " м³");
20        }
21    }
22 }

```

Рис. 4.6. Реалізація класу GasHolder

GasUtilizationSystem: клас, що представляє систему використання біогазу (наприклад, генератор електроенергії або систему опалення). На рисунку 4.7. представлено реалізацію цього класу в середовищі розробки NetBeans.

```

1 public class GasUtilizationSystem {
2     private final List<String> connectedDevices; // Список підключених пристроїв
3     private float totalGasUtilized; // Загальний обсяг використаного газу
4
5     // Конструктор для ініціалізації GasUtilizationSystem
6     public GasUtilizationSystem() {
7         this.connectedDevices = new ArrayList<>(); // Ініціалізація порожнього списку підключених
8         this.totalGasUtilized = 0; // Початковий обсяг використаного газу
9     }
10
11    // Метод для підключення пристрою
12    public void connectDevice(String device) {
13        if (!connectedDevices.contains(device)) {
14            connectedDevices.add(device);
15            System.out.println("Пристрій " + device + " підключено.");
16        } else {
17            System.out.println("Пристрій " + device + " вже підключено.");
18        }
19    }

```

Рис. 4.7. Реалізація класу GasUtilizationSystem

TemperatureControlSystem: клас, що представляє систему контролю температури в збірнику для оптимізації процесу перетворення. На рисунку 4.8. представлено реалізацію цього класу в середовищі розробки NetBeans.

```

1 public class TemperatureControlSystem {
2     private float targetTemperature; // Цільова температура
3     private float currentTemperature; // Поточна температура
4
5     // Конструктор для ініціалізації TemperatureControlSystem
6     public TemperatureControlSystem(float initialTemperature, float targetTemperature) {
7         this.currentTemperature = initialTemperature;
8         this.targetTemperature = targetTemperature;
9     }
10
11    // Метод для встановлення цільової температури
12    public void setTargetTemperature(float targetTemperature) {
13        this.targetTemperature = targetTemperature;
14        System.out.println("Цільова температура встановлена на " + targetTemperature + " °C.");
15    }
16
17    // Метод для отримання поточної температури
18    public float getCurrentTemperature() {
19        return currentTemperature;
20    }

```

Рис. 4.8. Реалізація класу TemperatureControlSystem

PressureControlSystem: клас, що представляє систему контролю тиску в газових трубопроводах або резервуарах. На рисунку 4.9. представлено реалізацію цього класу в середовищі розробки NetBeans.

```

1 public class PressureControlSystem {
2     private float targetPressure; // Цільовий тиск
3     private float currentPressure; // Поточний тиск
4
5     // Конструктор для ініціалізації PressureControlSystem
6     public PressureControlSystem(float initialPressure, float targetPressure) {
7         this.currentPressure = initialPressure;
8         this.targetPressure = targetPressure;
9     }
10
11    // Метод для встановлення цільового тиску
12    public void setTargetPressure(float targetPressure) {
13        this.targetPressure = targetPressure;
14        System.out.println("Цільовий тиск встановлений на " + targetPressure + " бар.");
15    }
16
17    // Метод для отримання поточного тиску
18    public float getCurrentPressure() {
19        return currentPressure;
20    }

```

Рис. 4.9. Реалізація класу PressureControlSystem

BiogasQualityAnalyzer: клас, що представляє аналізатор якості біогазу. На рисунку 4.10. представлено реалізацію цього класу в середовищі розробки NetBeans.

```

1 public class BiogasQualityAnalyzer {
2     // Метод для аналізу якості газу
3     public QualityReport analyzeGasQuality(GasSample gasSample) {
4         float methaneContent = gasSample.getMethaneContent();
5         float carbonDioxideContent = gasSample.getCarbonDioxideContent();
6         boolean isQualityAcceptable = false;
7
8         // Простий критерій повинен бути оновлений на основі реальних стандартів якості біогазу
9         if (methaneContent > 50 && carbonDioxideContent < 50) {
10            isQualityAcceptable = true;
11        }
12
13        return new QualityReport(methaneContent, carbonDioxideContent, qualityAcceptable: isQualityAcceptable);
14    }
15
16    public static void main(String[] args) {
17        BiogasQualityAnalyzer analyzer = new BiogasQualityAnalyzer();
18        GasSample sample = new GasSample(пар: 65.0f, пар: 34.0f); // Вміст метану: 65%, Вміст вуглекислоти: 34%
19    }

```

Рис. 4.10. Реалізація класу BiogasQualityAnalyzer

BiogasProductionModel: клас, що представляє модель виробництва біогазу, яка враховує різноманітні фактори, такі як склад сировини, температура, тиск, кислотність тощо. На рисунку 4.11. представлено реалізацію цього класу в середовищі розробки NetBeans.

```

1 public class BiogasProductionModel {
2     // Метод для прогнозування виробництва біогазу
3     public BiogasProductionPrediction predictBiogasProduction(FeedstockData feedstockData)
4     {
5         float baseBiogasYield = 0.05f; // Базова кількість біогазу на кг сировини, у кубічних метрах
6         float quantity = feedstockData.getQuantity();
7         float moistureContent = feedstockData.getMoistureContent();
8
9         // формула для оцінки виробництва біогазу з урахуванням вмісту вологи
10        float effectiveQuantity = quantity * (1 - moistureContent / 100);
11        float estimatedBiogasVolume = baseBiogasYield * effectiveQuantity;
12
13        return new BiogasProductionPrediction(estimatedBiogasVolume);
14    }
15 }

```

Рис. 4.11. Реалізація класу BiogasProductionModel

Лістинг коду основних модулів системи для моделювання процесів у біогазових установках наведено у додатку Б, а у додатку В представлено фрагмент лістингу Python коду, який використовується для реалізації математичних моделей з використанням підходу на основі аналізу інтервальних даних.

## 4.2 Тестування програмного середовища

Тестування програмного забезпечення відіграє важливу роль у розробці будь-якого ПЗ, оскільки воно гарантує, що продукт функціонує правильно та відповідає всім вимогам та специфікаціям, передбаченим розробниками та замовниками. Перший і головний аспект тестування полягає у забезпеченні якості продукту. Це включає виявлення та усунення помилок, що можуть значно вплинути на функціональність та надійність програмного забезпечення. Тестування допомагає переконатись, що програма працює так, як очікується, і що вона не зазнає збоїв під час виконання ключових завдань.

Другий важливий аспект тестування програмного забезпечення полягає у забезпеченні задоволеності користувача. Якісне ПЗ, яке ефективно вирішує задачі користувача без помилок і збоїв, значно підвищує ймовірність того, що користувач буде задоволений продуктом.

Третій аспект тестування полягає у підтримці масштабування програмного продукту. Тестування дає змогу перевірити, як програмне забезпечення буде справлятися з зростанням кількості користувачів та обсягів даних. Це особливо важливо для веб-додатків та сервісів, які мають бути готові до великого навантаження і забезпечувати стабільність та швидкість роботи навіть за умови підвищеної активності. Регулярне тестування і аналіз результатів дають розробникам важливі відомості для усунення недоліків і вдосконалення функціоналу. Це створює основу для стабільного розвитку ПЗ, де продукт стає все більш надійним і ефективним.

Оскільки система для моделювання процесів у біогазових установках є складним програмним продуктом з великою кількістю інтеграцій та даних основну увагу було приділено наступним видам тестування:

1. Модульне тестування. Модульне тестування дозволяє легше знайти і виправити помилки в окремих модулях, забезпечуючи швидкий зворотній зв'язок під час розробки завдяки частому запуску тестів. Тести служать додатковою документацією, пояснюючи, як повинні працювати окремі частини системи. Це також дозволяє безпечно змінювати код під час рефакторингу, будучи впевненим, що існуючий функціонал не порушено.

2. API тестування. Тестування API має кілька важливих переваг. Воно легко автоматизується, що значно знижує обсяг ручної роботи. Тести виконуються швидше порівняно з тестами користувацького інтерфейсу (UI тестами). Оскільки API тести незалежні від стану та готовності UI, це дозволяє виявляти помилки на ранніх етапах розробки. Також API тестування забезпечує точність, перевіряючи логіку та поведінку додатку на рівні бізнес-логіки.

3. Тестування продуктивності. Перформанс тестування допомагає знайти та усунути проблеми з продуктивністю додатку. Це важливо для покращення користувацького досвіду, забезпечуючи стабільну роботу системи під навантаженням. Перформанс тестування дозволяє оцінити, як система



поводитиметься при збільшенні кількості користувачів або даних. Крім того, воно дозволяє тестувати додаток у реальних умовах використання.

Виходячи з особливостей архітектури системи, тестові випадки можна розробити для кожного модуля в рамках двох основних підсистем: підсистеми моделювання процесів у біогазових установках та підсистеми управління процесом виробництва біогазу. Нижче наведено приклади тестових випадків для різних модулів програмної системи для моделювання процесів у біогазових установках.

Модуль обробки даних:

1. Перевірка валідації вхідних даних:

- мета: переконатися, що модуль коректно валідує вхідні інтервальні дані (наприклад, формати дат, числові діапазони);
- вхідні дані: різні формати та типи даних;
- очікуваний результат: помилки валідації для некоректних даних та успішне прийняття коректних даних.

2. Перевірка обробки помилок:

- мета: забезпечити, що система адекватно обробляє помилки під час введення неправильних даних.
- вхідні дані: некоректні вхідні параметри.
- очікуваний результат: коректне виведення помилок та забезпечення стабільності системи.

Модуль процесів моделювання:

1. Перевірка точності моделювання:

- мета: перевірити, що модуль надає точні та відповідні результати моделювання на основі заданих інтервальних даних.
- вхідні дані: відомі інтервальні дані з очікуваними результатами.
- очікуваний результат: результати моделювання відповідають очікуванім.

2. Моделювання при крайніх значеннях:

- мета: переконатися, що модель стійка при обробці крайніх значень вхідних даних.

- вхідні дані: максимально високі та низькі значення параметрів.

- очікуваний результат: стабільна робота модулю без помилок виконання.

Модуль аналізу результатів:

#### 1. Валідність аналітичних висновків

- мета: перевірити, що аналіз результатів моделювання надає коректні та логічні висновки;

- вхідні дані: результати моделювання;

- очікуваний результат: аналіз відповідає логічним очікуванням та стандартам.

Модуль інтерфейсу користувача:

#### 1. Інтуїтивність користувацького інтерфейсу:

- мета: переконатися, що інтерфейс користувача є зрозумілим і легким для використання;

- вхідні дані: різні користувацькі сценарії;

- очікуваний результат: користувачі можуть легко навігувати та виконувати задачі без попереднього навчання.

Модуль управління конфігурацією:

#### 1. Гнучкість конфігурації:

- мета: перевірити, що модуль дозволяє користувачам гнучко налаштовувати параметри виробництва біогазу;

- вхідні дані: різні налаштування параметрів виробництва;

- очікуваний результат: система приймає налаштування та коректно застосовує їх у процесах виробництва.

Модуль регулювання та контролю:

#### 1. Реакція на зміну умов:

- мета: перевірити, що система швидко і коректно реагує на зміни умов виробництва.

- вхідні дані: штучні зміни у параметрах процесу (температура, тиск тощо).

- очікуваний результат: система адекватно адаптує регулювання відповідно до змін.

Таблиця 4.1

## Результати виконання модульного тестування

Назва модуля	К-сть тестів	Результат
Модуль обробки даних	8	Пройдено
Модуль процесів моделювання	16	Пройдено
Модуль аналізу результатів	13	Пройдено
Модуль інтерфейсу користувача	10	Пройдено
Модуль управління конфігурацією	8	Пройдено
Модуль регулювання та контролю	10	Пройдено

Оскільки розроблене програмне забезпечення працює на сервісно-орієнтованій архітектурі, то в роботі було проведено тестування API. Тестування API є важливою частиною розробки програмного забезпечення, оскільки це допомагає переконатися, що API веде себе як очікувалося під різними умовами. Нижче наведено групи тестів для перевірки програмної системи:

1. Позитивні тести. Ці тести перевіряють, чи відповідає API правильно в очікуваних і ідеальних умовах. Вони передбачають надсилання дійсних вхідних даних, щоб побачити, чи API повертає правильний вихід, дотримується специфікацій часу відповіді та зберігає послідовність між різними запитами.

2. Негативні тести. Негативне тестування є критично важливим для забезпечення надійного оброблення помилок та стійкості, навмисно надаючи неправильні, крайні або недійсні вхідні дані, щоб побачити, наскільки добре API може впоратися з такими випадками.

3. Тести безпеки. Тестування безпеки вивчає механізми API для захисту від зловмисних атак та забезпечення цілісності та конфіденційності даних.

4. Тести продуктивності. Ці тести оцінюють ефективність та масштабованість API за різних умов, забезпечуючи, що він може впоратися з очікуваними та піковими навантаженнями.

5. Оброблення помилок. Належне оброблення помилок є життєво важливим для зручності користування і надійності API.

В таблиці 4.2 наведено приклади тестових сценаріїв для кожної групи проведених тестів. Тестування API є важливою частиною розробки програмного забезпечення, оскільки це допомагає переконатися, що API веде себе як очікувалося під різними умовами. Нижче наведено групи тестів для перевірки програмної системи:

1. Позитивні тести. Ці тести перевіряють, чи відповідає API правильно в очікуваних і ідеальних умовах. Вони передбачають надсилання дійсних вхідних даних, щоб побачити, чи API повертає правильний вихід, дотримується специфікацій часу відповіді та зберігає послідовність між різними типами здійснених запитів.

2. Негативні тести. Негативне тестування є критично важливим для забезпечення надійного оброблення помилок та стійкості, навмисно надаючи неправильні, крайні або недійсні вхідні дані, щоб побачити, наскільки добре API може впоратися з такими випадками.

3. Тести безпеки. Тестування безпеки вивчає механізми API для захисту від зловмисних атак та забезпечення цілісності та конфіденційності даних.

4. Тести продуктивності. Ці тести оцінюють ефективність та масштабованість API за різних умов, забезпечуючи, що він може впоратися з очікуваними та піковими навантаженнями.

5. Оброблення помилок. Належне оброблення помилок є життєво важливим для зручності користування і надійності API.

Таблиця 4.2

## Групи тестових сценаріїв для перевірки API

Тестові категорії	Тестові сценарії
Позитивні тести	<p>Тестування функціональності: забезпечує, що кожна функція API виконується як очікується.</p> <p>Тестування валідації: перевіряє, що вхідні дані правильно приймаються, а вихідні дані точно генеруються: відправка запиту з правильними і очікуваними параметрами і перевірка, чи повертаються вірні відповіді перевірка правильності обробки різних типів даних ( числа, рядки, дати).</p>
Негативні тести	<p>Перевірка введення: тести, як API обробляє значення, що виходять за межі, неправильні формати даних і відсутні параметри.</p> <p>відправка запиту з невалідними параметрами або без обов'язкових параметрів, щоб перевірити, як API обробляє помилки перевірка повернення відповідних статусних кодів та повідомлень про помилки.</p> <p>Оброблення помилок: забезпечує, що API повертає відповідні повідомлення про помилки і коди, коли воно не може обробити запит, допомагаючи клієнтській програмі правильно обробляти помилки відправка запиту з параметрами на межі припустимих значень, наприклад максимальне та мінімальне значення числа, максимальна довжина рядка перевірка, чи правильно API обробляє межові випадки.</p>
Тести безпеки	<p>Ауθενфікація: тести для забезпечення того, що API правильно ідентифікує та автентифікує користувачів, відкидаючи спроби неавторизованого доступу.</p> <p>Авторизація: перевіряє, що API дозволяє доступ до функцій і даних відповідно до привілеїв користувача.</p> <p>Шифрування даних: Перевіряє, що чутливі дані шифруються під час передачі.</p>
Тести продуктивності	<p>Тестування навантаження: визначає, як API виконується під зростаючим навантаженням запитів, щоб ідентифікувати точку, в якій продуктивність API починає погіршуватися.</p> <p>Стрес-тестування: тести API понад нормальну робочу спроможність, щоб побачити, як він впорається з екстремальними умовами, що допомагає зрозуміти його точку зламу.</p>
Оброблення помилок	<p>Коди помилок: забезпечує, що API повертає правильні коди статусу HTTP та повідомлення про помилки, які точно відображають умови помилок.</p> <p>Послідовність: перевіряє, чи API послідовно обробляє помилки на різних кінцевих точках</p>

Визначення цих тестових випадків забезпечує всебічне охоплення всіх аспектів функціональності API. Це не тільки підвищує надійність API, але і забезпечує стабільний і передбачуваний інтерфейс.

Зважаючи на те, що програмна система для моделювання процесів у біогазових установках передбачає роботу з великим об'ємом даних то управління даними є одним з ключових елементів в процесі тестування API. Ось основні аспекти управління даними які були враховані для тестування API:

#### 1. Підготовка тестових даних.

Генерація даних: створення даних, що імітують реальні умови використання API. Це можуть бути дані, згенеровані автоматично за допомогою скриптів або вручну складені набори даних.

Використання існуючих даних: збір даних з реального середовища для тестування умов, які близькі до реального використання. Важливо забезпечити, що використання цих даних відповідає всім вимогам конфіденційності та безпеки.

Обробка даних: процес очищення та нормалізації даних для забезпечення їхньої консистентності та валідності перед використанням у тестах.

#### 2. Управління середовищем тестування.

Ізоляція даних: забезпечення, що тестові дані використовуються в ізольованому середовищі, щоб уникнути впливу на реальні дані або на інші тестові процеси.

Відтворення стану даних: можливість відтворення певного стану даних для повторення тестів або для тестування різних сценаріїв, які вимагають конкретного стану даних.

#### 3. Версіонування та відслідковування змін.

Версіонування даних: застосування систем версіонування для тестових даних, щоб забезпечити здатність відслідковувати зміни і відновлювати попередні стани.

Аудит та логування: запис змін у тестових даних та результатів тестування для аналізу та виявлення причин проблем у майбутньому.

На рисунку 4.12 показано приклад виконання запиту `get model` у програмі Postman. В даному випадку було протестовано швидкість виконання запиту (`response time is less than 200ms`) перевірку правильності статус-коду (`status`

code is 200), відповідність відповіді схемі (schema is valid) та наявність заголовка Content-Type у відповіді.

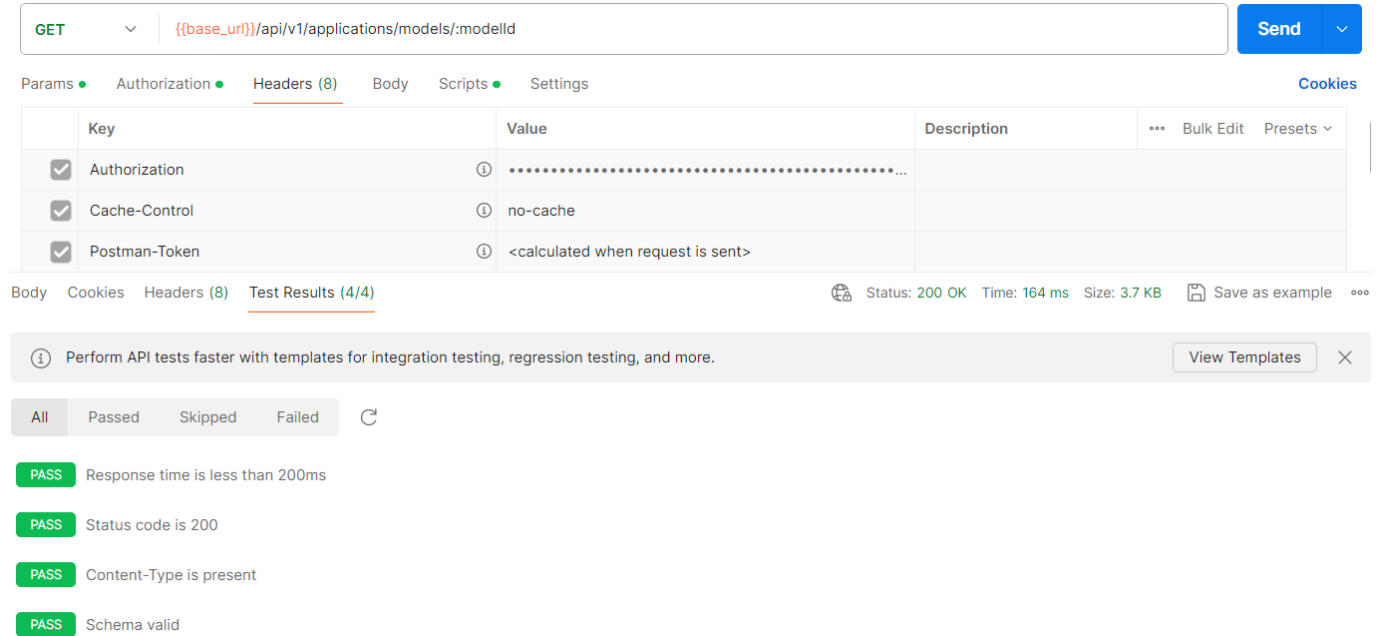


Рис. 4.12. Результат виконання тестів для запиту get model

Загалом було написано 719 тестів для 93 API запитів які були успішно пройдені. На рисунку 4.13. зображено звіт тестування API системи.

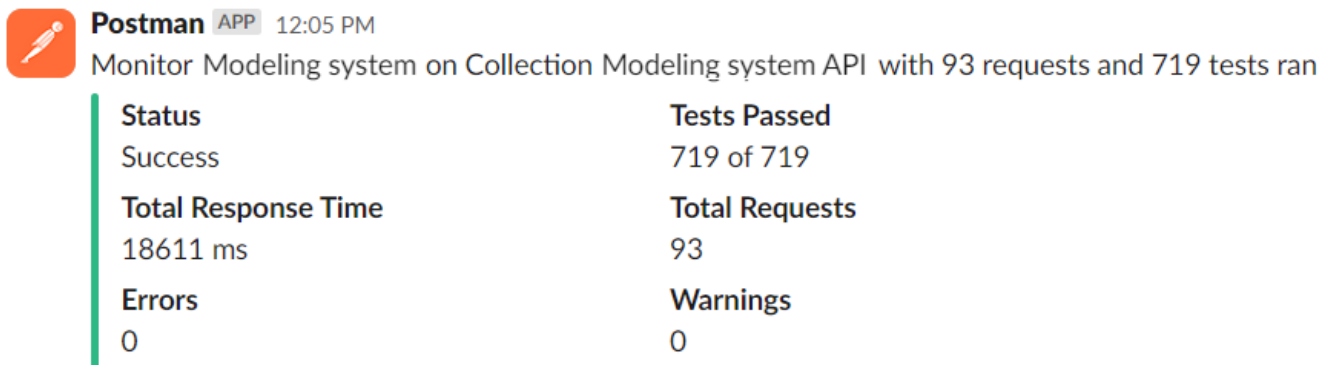


Рис. 4.13. Звіт тестування API системи

Тестування продуктивності є важливим для програмних систем і забезпечення їхньої оптимальної функціональності.

Тестування дозволяє виявити потенційні вузькі місця та слабкі точки в архітектурі системи, забезпечуючи її здатність обробляти очікуване навантаження запитів і одночасно виконувати різні активності. Імітуючи реальні сценарії використання, тестування продуктивності перевіряє час відгуку, використання ресурсів і масштабованість, що допомагає уникнути збоїв або уповільнень під час інтенсивного використання системи моделювання.

Оскільки програмна система для моделювання процесів у біогазових установках обробляє значні обсяги даних, тестування продуктивності допомагає точно визначити області для оптимізації та налаштування, підвищуючи загальну ефективність системи та знижуючи операційні ризики. Для тестування продуктивності системи було визначено наступні метрики:

- пропускна здатність
- час відповіді:
- 90 перцентиль;
- 95 перцентиль;
- 99 перцентиль;
- частота помилок.

Після чого розроблено наступні тестові сценарії:

- логін користувача;
- завантаження даних;
- побудова математичних моделей;
- регулювання даних;
- моделювання процесів.

Результати тестування продуктивності системи для моделювання процесів у біогазових установках представлено у таблиці 4.3.

Всі запити були виконані з нульовим рівнем помилок. Проте, запити для моделювання процесів мають найдовший час обробки – 2 сек. Це може вказувати на високу складність алгоритмів, що використовуються для обробки таких користувацьких запитів.



Таблиця 4.3

## Результати тестування продуктивності системи

Тег	Кількість запитів	90 Персентиль (мс)	95 персентиль (мс)	99 персентиль (мс)	Частота помилок %
Логін користувача	20	1843	1632	1503	0
Завантаження даних	100	1943	1843	1856	0
Побудова математичних моделей	150	2453	2278	2412	0
Регулювання даних	130	1887	1753	1723	0
Моделювання процесів	250	2013	1943	2135	0

Однак, враховуючи великий обсяг даних, така тривалість є допустимою. Тестування проводилося за допомогою інструменту Apache JMeter (рисунок 4.14), де були створені відповідні тестові скрипти.

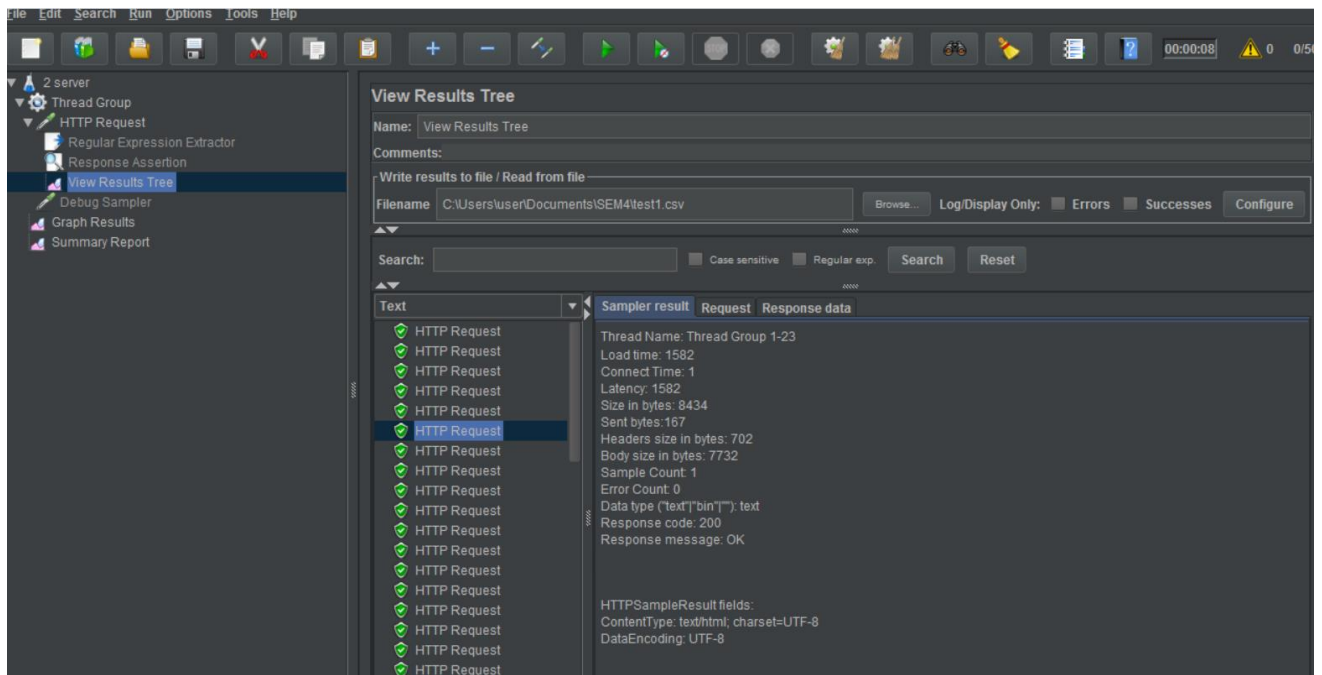


Рис. 4.14. Результат виконання тестів в програмі Apache JMeter

### 4.3 Організація графічного інтерфейсу програмного середовища для моделювання процесів у біогазових установках

Розглянемо детальніше реалізоване програмне середовище для моделювання процесів у біогазових установках, яке ґрунтується на програмній інтерпретації інтервальних дискретних моделей та відкритій сервісно-орієнтованій архітектурі. Дане середовище може інтегруватися в існуючі системи управління біогазовими установками з метою забезпечення допустимих значень кислотності багатокомпонентного субстрату.

На рисунку 4.15 представлено основні підсистеми, які реалізовано в рамках програмного середовища для моделювання процесів у біогазових установках.

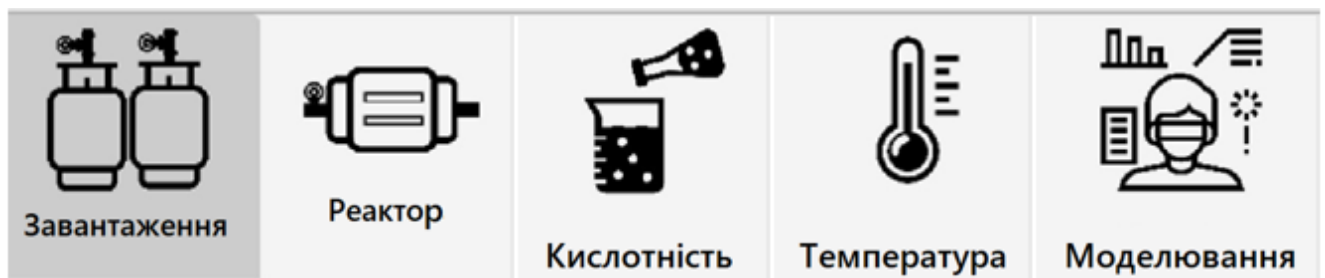


Рис. 4.15. Представлення основних підсистем

Підсистема «Завантаження» відіграє ключову роль на початковому етапі процесу виробництва біогазу. Основні характеристики та функції цієї підсистеми включають: вибір типу біомаси (барда; жом з цукрових буряків; патока та сечівка), введення основних показників на початок завантаження (кислотність, температура, вологість). Підсистема визначає вимоги до якості біомаси, які необхідно дотримуватися під час завантаження. Ці показники є важливими для забезпечення оптимальних умов для подальшого перетворення у біогаз. Підсистема забезпечує моніторинг та контроль за процесами завантаження біомаси, щоб забезпечити безперебійну роботу системи та запобігти можливим проблемам чи перешкодам під час цього етапу. На рисунку 4.16 представлено основну екранну форму, яка стосується управління підсистемою «Завантаження».

Моделювання    Оцінка параметрів    Звіти    Налаштування

Завантаження    Реактор    Кислотність    Температура    Моделювання

Тип біомаси

Жом    Зберегти    Відмінити

Об'єм біомаси

80    1000 Кг

Кислотність

80    рН

Температура

1.0    °C

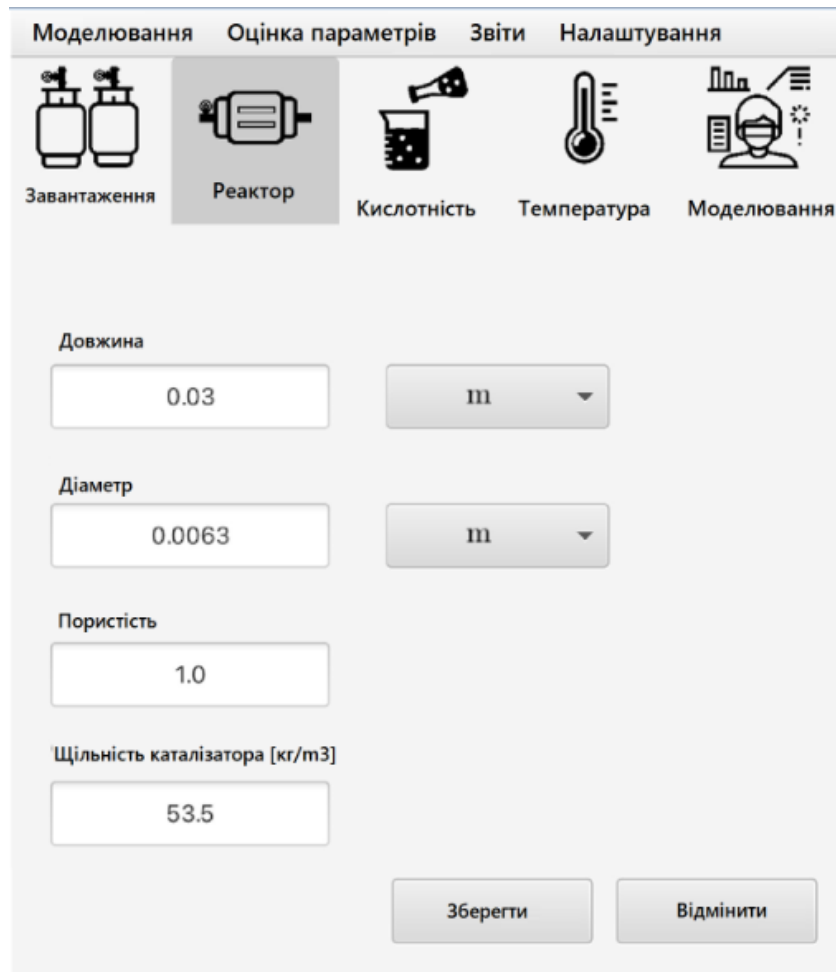
Вологість

1.0    кг/м3

Рис. 4.16. Екранна форма налаштування параметрів завантаження біомаси

На рисунку 4.17 представлено головну форму роботи системи при симуляції функціонування біогазової установки, яка стосується введення налаштувань системи відносно основних параметрів біореактора, таких як довжина, діаметр, пористість, щільність каталізатора.

Підсистема здійснює моніторинг та контроль за різними параметрами в реакторі, такими як рівень метану, інших газів, температура, рН тощо. Це дозволяє вчасно виявляти будь-які недоліки або зміни у процесі. Підсистема "Реактор" взаємодіє з іншими підсистемами, такими як "Завантаження", "Кислотність", "Температура", "Моделювання", для забезпечення взаємодії та синхронізації процесів усієї системи.



Моделювання    Оцінка параметрів    Звіти    Налаштування

Завантаження    Реактор    Кислотність    Температура    Моделювання

Довжина

0.03    m

Діаметр

0.0063    m

Пористість

1.0

Щільність каталізатора [кг/м<sup>3</sup>]

53.5

Зберегти    Відмінити

Рис. 4.17. Екранна форма налаштування параметрів біореактора

Підсистема "Кислотність" в системі моделювання процесів у біогазових установках відіграє важливу роль у забезпеченні оптимальних умов для ефективного проходження метанового бродіння та виробництва біогазу.

Оптимальне значення рН впливає на активність мікроорганізмів, які забезпечують процеси анаеробного бродіння. Підсистема допомагає забезпечити стабільність середовища у реакторі, що сприяє оптимальному функціонуванню мікроорганізмів, відповідальних за бродіння біомаси.

Важливою функцією підсистеми "Кислотність" є контроль буферного потенціалу середовища, який допомагає утримати стабільний рівень рН під час зміни умов реакцій.

Підсистема "Кислотність" взаємодіє з іншими компонентами системи, зокрема з підсистемою "Реактор", для забезпечення узгодженості основних

параметрів в реакторі та з підсистемою «Моделювання» для уточнення та побудови математичних моделей. На рисунку 4.18 представлено основну екранну форму, яка стосується управління підсистемою «Кислотність».

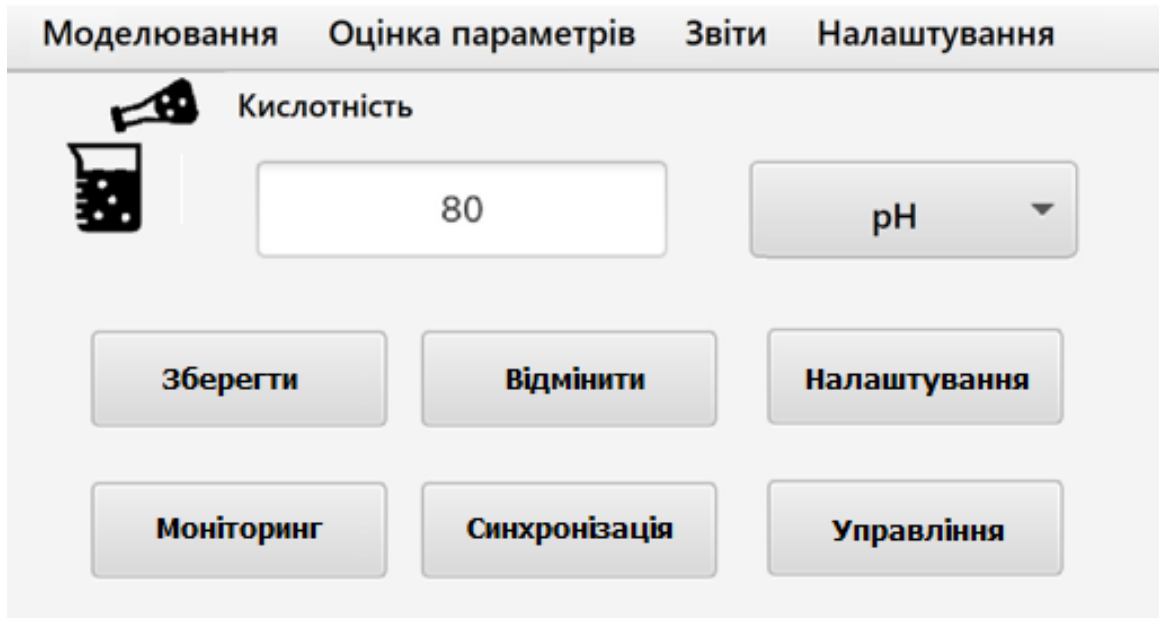


Рис. 4.18. Екранна форма налаштування параметрів кислотності

Підсистема "Температура" в системі моделювання процесів у біогазових установках відіграє значущу роль у забезпеченні оптимальних умов для виробництва біогазу шляхом метанового бродіння біомаси.

Оптимальна температура впливає на активність бактерій та швидкість бродіння, що забезпечує ефективне виробництво біогазу.

Підсистема також відповідає за контроль нагрівання чи охолодження середовища в реакторі, щоб забезпечити правильну температуру в умовах зміни зовнішніх параметрів.

Підсистема взаємодіє з іншими компонентами системи, зокрема з підсистемою "Реактор" та "Моделювання", для забезпечення правильних температурних умов у системі в цілому. На рисунку 4.19 представлено основну екранну форму, яка стосується управління підсистемою «Температура».

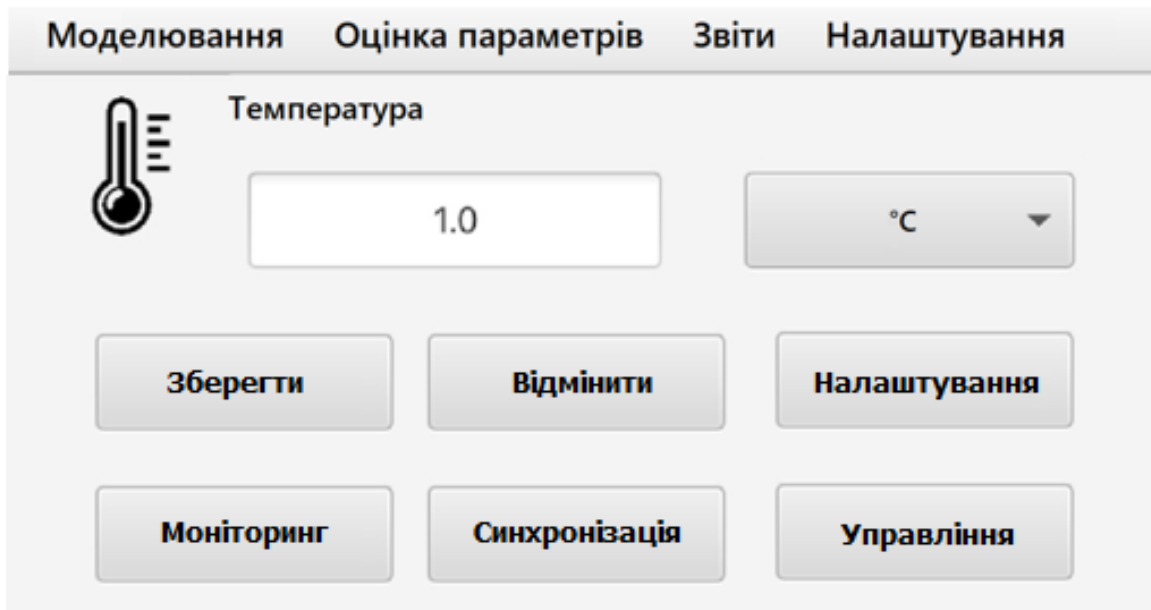


Рис. 4.19. Екранна форма налаштування параметрів температури

Особливе місце в даному середовищі належить підсистемі «Моделювання». Підсистема "Моделювання" у системі моделювання процесів у біогазових установках є ключовою складовою, яка відповідає за створення, управління та аналіз моделей, необхідних для імітації та оптимізації процесів у цій галузі.

Підсистема дозволяє користувачам розробляти математичні моделі, які відтворюють процеси, які відбуваються у біогазових установках. Підсистема дозволяє виконувати симуляції на основі розроблених моделей для аналізу поведінки системи у різних умовах і варіантах. Це допомагає приймати обґрунтовані рішення та оптимізувати ці процеси шляхом зміни параметрів у моделях, щоб досліджувати вплив різних чинників на результати аналізу. За допомогою цієї підсистеми можна виконувати оптимізацію процесів у біогазових установках шляхом зміни параметрів, умов та стратегій управління.

Підсистема "Моделювання" взаємодіє з іншими компонентами системи, такими як "Реактор", "Кислотність", "Температура", "Завантаження" для покращення точності та повноти модельованих процесів. На рисунку 4.20 представлено одну з ключових екранних форм в рамках підсистеми «Моделювання», яка стосується отримання даних для моделювання, самого

процесу побудови моделі та візуалізації отриманих результатів у формі відповідних графіків (рисунок 4.21).

5	7,6959	8,2047	7,95	200	101	7,99776	0,58%	-0,04	0,20	7,985	0,42%
6	7,7346	8,2460	7,99	150	97,5	7,98842	0,02%	-0,24	-0,41	7,953	0,45%
7	7,9669	8,4937	8,23	30	102	7,98013	3,04%	0,17	0,14	7,954	3,31%
8	7,8024	8,3182	8,06	110	112,5	8,09443	0,41%	0,03	0,06	7,958	1,61%
9	7,7733	8,2873	8,03	320	69	7,94205	1,07%	-0,03	-0,08	7,928	1,50%
10	7,8024	8,3182	8,06	120	100	8,01252	0,58%	0,05	0,08	7,907	1,73%
11	7,7540	8,2666	8,01	130	0	8,06671	0,69%	-0,03	-0,13	7,918	0,90%
12	7,7830	8,2976	8,04	210	37,5	8,01646	0,29%	0,10	0,26	7,936	1,04%
13	7,6862	8,1944	7,94	220	51	8,06340	1,50%	-0,16	-0,11	7,955	0,47%

Рис. 4.20. Екранна форма підсистеми «Моделювання»

Також у комп'ютерному середовищі реалізовану підсистему, яка використовується для формування звітів. Ця підсистема відповідає за створення звітів та аналітичну звітність на основі результатів моделювання процесів у біогазових установках.

Система автоматично генерує звіти, які містять результати моделювання процесів у біогазових установках. Ці звіти можуть включати дані про ефективність, витрати, параметри процесів та іншу важливу інформацію. Підсистема надає можливість аналізу та візуалізації отриманих результатів моделювання. Це допомагає користувачам отримувати необхідну інформацію та приймати відповідні управлінські рішення.

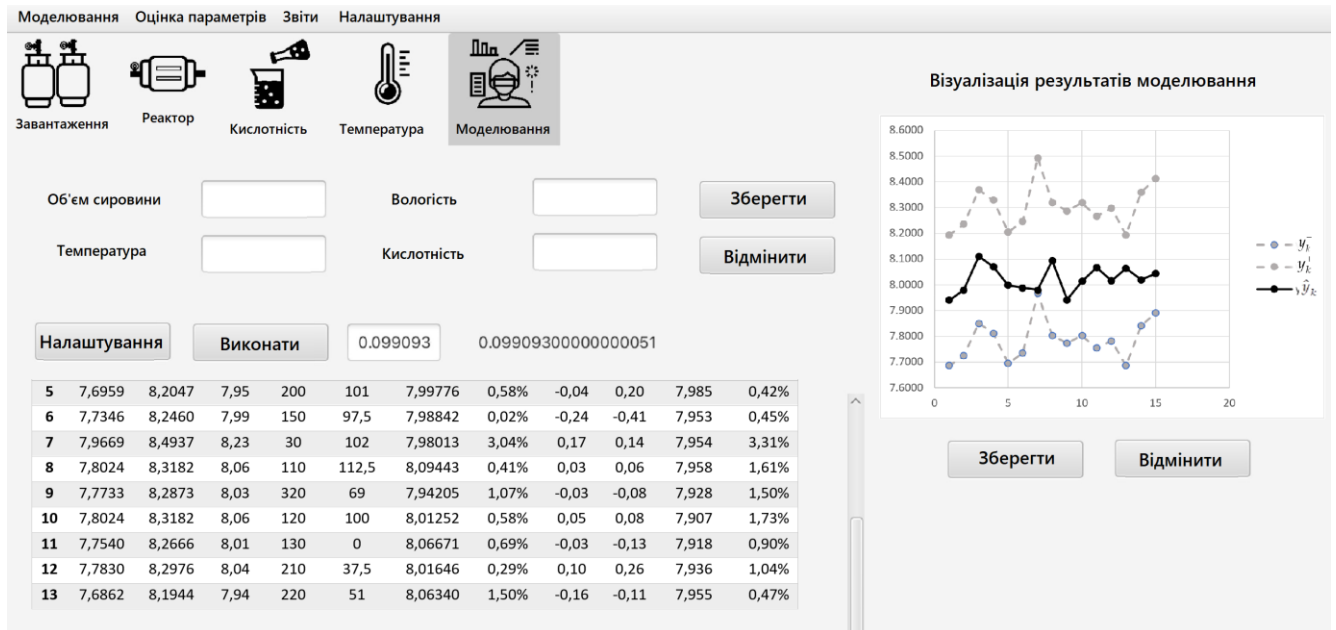


Рис. 4.21. Екранна форма відображення результатів моделювання

Підсистема може автоматично надсилати сповіщення про завершення генерації звітів та моніторити процеси моделювання для вчасного реагування на можливі аномалії. Підсистема "Звіти" може надавати користувачам доступ до згенерованих звітів через веб-інтерфейс або інші канали комунікації, сприяючи зручності та доступності інформації.

Також у середовищі реалізовано можливість отримання інформації на основі API – підсистема «Налаштування». Ця підсистема відповідає за налаштування та керування програмним інтерфейсом (API), який використовується для взаємодії з іншими програмами або сервісами.

Підсистема забезпечує можливості безпеки та автентифікації для взаємодії з API, наприклад, здійснює контроль доступу, шифрування даних, валідацію запитів тощо. Також дозволяє виконувати моніторинг та логування взаємодії з API для відстеження запитів, реакцій та виявлення можливих проблем.

Підсистема "Налаштування API" інтегрується з іншими компонентами системи, такими як "Звіти", "Реактор", "Кислотність", "Температура", "Завантаження", "Моделювання" для забезпечення ефективної роботи системи у цілому.



Програмне середовище для реалізації математичних моделей динаміки процесів у БГУ використано, зокрема для моделювання «динаміки концентрацій ацетату на стадії ацидогенезу», «динаміки відсотку малої субодиниці рРНК кислоутворюючих бактерій на стадії ацетогенезу», «динаміки відсотку малої субодиниці рРНК метаноутворюючих бактерій на стадії метаногенезу».

Аналіз результатів модельних розрахунків показує особливості врахування динаміки показника кислотності субстрату у біореакторі та дозволяє підтвердити ефективність використання методів аналізу інтервальних даних та відповідного розробленого програмного забезпечення. Ефективність запропонованих алгоритмів підтверджено результатами обчислювальних експериментів.

#### **Висновки до розділу 4**

1. Описано особливості програмної реалізації системи, зокрема з використанням парадигми об'єктно-орієнтованого програмування на базі відкритої сервісно-орієнтованої архітектури з використанням технології Spring Framework на мові програмування Java, а також інтерпретатором Python. В якості Python інтерпретатора було обрано Jython, а в якості СУБД використовується MySQL.

2. Під час реалізації програмного забезпечення здійснено інтеграційне тестування та тестування продуктивності середовища для моделювання процесів у біогазових установках. Сформовано відповідні тестові сценарії для перевірки коректності роботи API. За результатами перевірки, зовнішній інтерфейс успішно пройшов 719 із 719 тестів, що свідчить про його коректну роботу. Тестування продуктивності підтвердило ефективну роботу системи, оскільки всі тести були пройдені без помилок з показником 0%. Це свідчить про ефективність програми при моделюванні процесів у біогазових установках та вказує на можливість подальшого розвитку проекту та його імплементації в діючі біогазові установки.

3. Описано графічний інтерфейсу програмного середовища для моделювання процесів у біогазових. Описано функціонування основних підсистем з точки зору взаємодії між собою, різними типами користувачів, а також можливість зовнішньої інтеграції в діючі біогазові установки.

## ВИСНОВКИ

У дисертаційній роботі розв'язано науково-технічне завдання забезпечення допустимих значень кислотності багатокomпонентного субстрату за рахунок розробки математичного та програмного забезпечення для моделювання динаміки кислотності субстрату у біогазових установках на основі аналізу інтервальних даних. При цьому отримано такі наукові та практичні результати.

1. Проаналізовано особливості виробництва біогазу у спосіб переробки органічних відходів життєдіяльності людини та відходів тваринництва і агропромислового комплексу. Показано, що підвищення ефективності функціонування цього виробництва вимагає розробки нових підходів управління процесами в біогазових установках на основі застосування інформаційних технологій та методів математичного моделювання.

2. Розглянуто особливості моделювання процесів у біогазових установках може надати динамічну інформацію про процес анаеробного зброджування, наприклад, для прогнозування виходу біогазу або оптимізації процесу анаеробного зброджування. Показано, що одною із визначальних характеристик процесу метанового бродіння в БГУ є показник рН ферментаційного середовища. Запропоновано та обґрунтовано застосування методів математичного моделювання з метою відображення взаємозв'язку між основною характеристикою процесу та чинниками, які на неї впливають так і динаміки основної характеристики процесу, яка визначається кислотністю субстрату у біореакторі. Для побудови математичних моделей обох видів запропоновано використати методи параметричної ідентифікації моделей статичних об'єктів та дискретних моделей динаміки об'єктів на основі аналізу інтервальних даних.

3. Запропоновано та обґрунтовано новий двоетапний метод ідентифікації математичної моделі динаміки показника кислотності субстрату у біореакторі на основі аналізу інтервальних даних, який ґрунтується на розв'язуванні спочатку простішої задачі ідентифікації рівняння взаємозв'язку між основною

характеристикою процесу та чинниками, які на неї впливають, і на цій основі, на другому етапі - ідентифікації математичної моделі динаміки цього показника у біореакторі. Такий підхід забезпечив спрощення побудови математичної моделі з гарантованими прогностичними властивостями за рахунок спрощення процедури вибору структури моделі. Показано, що точність побудованої моделі знаходиться в межах точності отриманих інтервальних даних, які у тестовому прикладі було отримано з 1% похибкою.

4. За результатами застосування нового двоетапного методу ідентифікації математичної моделі отримано математичну модель динаміки показника кислотності субстрату у біореакторі у вигляді диференціального рівняння, яке, на відміну від існуючих, налаштовано на параметри процесу для конкретної біогазової установки на підставі перетворення інтервальних різницевих рівнянь, що забезпечує гарантовані прогностичні властивості побудованої математичної моделі в межах точності інтервальних даних і уможливорює забезпечення допустимих значень кислотності багатокomпонентного субстрату в біогазових установках на основі управління балансом сухої та рідкої частин сировини. При побудові цієї інтервальної моделі (на основі отриманого диференціального рівняння) похибка розв'язку склала 3,3%, що співмірно із похибкою прогнозу, побудованого на підставі різницевої схеми. Також встановлено, що ця похибка залежить від точності апроксимації керуючої функції.

5. У процесі розробки програмного комплексу запропоновано та обґрунтовано відкриту сервісно-орієнтовану архітектуру програмного забезпечення для моделювання процесів у біогазових установках, яка, на відміну від існуючих інтегрує програмні компоненти: інтерпретації інтервальних дискретних моделей; ідентифікації цих моделей; розв'язування диференціальних рівнянь. Такий підхід уможливив створення прототипу комп'ютерного середовища для управління процесами в біогазових установках, яке без додаткових налаштувань інтегрується в існуючі системи управління біогазовими установками

з метою забезпечення коридору допустимих значень кислотності багатокomпонентного субстрату.

6. Достовірність отриманих наукових результатів підтверджено апробацією розроблених методів, моделей та програмного комплексу на діючих біогазових установках, зокрема, на базі ТОВ «Теофіпольська енергетична компанія», а також при реалізації багатьох програмних проєктів у Тернопільській філії компанії «ЕЛЕКС», які включали інтеграцію програмно-реалізованих засобів математичного моделювання.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Abraham, A., Jatoth, R. K., Rajasekhar, A. (2012). Hybrid differential artificial bee colony algorithm. *J. Comput. Theor. Nanosci.*, № 9, с. 249–257. <https://doi.org/10.1166/jctn.2012.2019>
2. Akay B, Karaboga D. Algorithms Simulating Bee Swarm Intelligence; *Artificial Intelligence Review*; 31 (1), pp. 68-85, 2009.
3. Akay, B.; Karaboga, D.; Gorkemli, B.; Kaya, E. A survey on the artificial bee colony algorithm variants for binary, integer and mixed integer programming problems. *Appl. Soft Comput.* 2021, 106, 107351. [Online]. Available at: <https://doi.org/10.1016/j.asoc.2021.107351>
4. Albalawi, K.S.; Mishra, M.N.; Goswami, P. Analysis of the Multi-Dimensional Navier–Stokes Equation by Caputo Fractional Operator. *Fractal Fract.* 2022, 6, 743. [Online]. Available at: <https://doi.org/10.3390/fractalfract6120743>
5. Alrefai, A.M.; Alrefai, R.; Benyounis, K.Y.; Stokes, J. Biogas Produced by Anaerobic Digestion Process and Biodiesel from Date Seeds. *Energies* 2021, 14(16), 4851. [Online]. Available at: <https://doi.org/10.3390/en14164851>
6. Alshattawi, S., Afifi, L., Shatnawi, A. M., Barhoush, M. M. (2022). "Utilizing genetic algorithm and artificial bee colony algorithm to extend the WSN lifetime". *Int. J. Comput.*, 21(1), 25-31. <https://doi.org/10.47839/ijc.21.1.2514>.
7. Angelidaki, I.; Ahring, B.K. Thermophilic anaerobic digestion of livestock waste: The effect of ammonia. *Appl. Microbiol. Biotechnol.* 1993, 38, 560–564.
8. Arzate Salgado, J.A. Modeling and Simulation of Biogas Production Based on Anaerobic Digestion of Energy Crops and Manure. Berlin University of Technology: Berlin, Germany, 2019.
9. Azman, S.; Khadem, A.F.; Plugge, C.M.; Stams, A.J.M.; Bec, S.; Zeeman, G. Effect of humic acid on anaerobic digestion of cellulose and xylan in completely stirred tank reactors: Inhibitory effect, mitigation of the inhibition and the dynamics of the microbial communities. *Appl. Microbiol. Biotechnol.* 2017, 101, 889–901.

10. Azman, S.; Khadem, A.F.; van Lier, J.B.; Zeeman, G.; Plugge, C.M. Presence and role of anaerobic hydrolytic microbes in conversion of lignocellulosic biomass for biogas production. *Crit. Rev. Environ. Sci. Technol.* 2015, 45, 2523–2564.
11. Beck, A. (2014). *Introduction to nonlinear optimization: Theory, algorithms, and applications with MATLAB*. Society for Industrial and Applied Mathematics. <https://doi.org/10.1137/1.9781611973655>
12. Bezobrazov, S., Sachenko, A., Komar, M., Rubanau, V. (2016). "The methods of artificial intelligence for malicious applications detection in android OS". *Int. J. Comput.*, 15(3), 184-190. <https://doi.org/10.31891/1727-6209/2016/15/3-184-190>.
13. Billings, S. A. (2013). *Nonlinear System Identification: NARMAX Methods in the Time, Frequency, and Spatio-Temporal Domains*. Wiley. 688 p. <https://doi.org/10.1002/9781118535561>.
14. Bober, W. (2013). *Introduction to Numerical and Analytical Methods with MATLAB for Engineers and Scientists*. CRC Press. p. 517. ISBN 978-1-4665-7609-4. Retrieved October 15, 2020.
15. Bouhleb, M.A.; Hwang, J.T.; Bartoli, N.; Lafage, R.; Morlier, J.; Martins, J.R.R.A Python Surrogate Modeling Framework with Derivatives. *Adv. Eng. Softw.* 2019, 135, 102662.
16. Brilli, A., Liuzzi, G., Lucidi, S. (2022). An interior point method for nonlinear constrained derivative-free optimization. *Mathematics. Optimization and Control*. <https://arxiv.org/abs/2108.05157>
17. Callens, R., Moens, D., Faes, M. (2023). "Certified interval model updating using scenario optimization". In *Proceedings of the 5th ECCOMAS Thematic Conference on Uncertainty Quantification in Computational Sciences and Engineering*, pp. 408-418. <https://doi.org/10.7712/120223.10346.19855>.
18. Cazier, E.A.; Trably, E.; Steyer, J.-P.; Escudié, R. Biomass hydrolysis inhibition at high hydrogen partial pressure in solid-state anaerobic digestion. *Bioresour. Technol.* 2015, 190, 106–113.

19. Charles, T., Gwilliam, C. (2023). The Effect of Automated Error Message Feedback on Undergraduate Physics Students Learning Python: Reducing Anxiety and Building Confidence. *J. STEM Educ. Res.* 6, 326–357.
20. Chen, Y., Cheng, J.J., Creamer, K.S. (2008). Inhibition of anaerobic digestion process: A review. *Bioresour. Technol.* 99, 4044–4064.
21. Christou, I.T., Darrell, W.L., De Long, K., Martin, W. (2021). *Evolutionary Algorithms*. SpringerVerlag: New York, NY, USA.
22. Cinar, S., Cinar, S.O., Wiczorek, N., Sohoo, I., Kuchta, K. (2021). Integration of Artificial Intelligence into Biogas Plant Operation. *Processes* 9, 85. [Online]. Available at: <https://doi.org/10.3390/pr9010085>
23. Cinar, S.O., Qdais, N., Kuchta, K. (2019). Optimizing the performance of a large scale biogas plant by controlling stirring process: A case study. *Energy Conversion and Management* 2019, 198, 111931. Available at: <https://doi.org/10.1016/j.enconman.2019.111931>
24. Cinar, S.Ö., Kuchta, K. (2020). Evaluation of Temperature Changes in Anaerobic Digestion Process. *Detritus* 10, 11–Hans.
25. Cutting, V., Stephen, N. (2021). Comparative review of java and python. *Int. J. Res. Dev. Appl. Sci. Eng. (IJRDASE)* 21.
26. De Marchi, A. (2023). Proximal gradient methods beyond monotony. *Journal of Nonsmooth Analysis and Optimization*, vol. 4. <https://doi.org/10.46298/jnsao-2023-10290>
27. De Marchi, A., Themelis, A. (2022). An interior proximal gradient method for nonconvex optimization. *Mathematics, Computer Science*. <https://arxiv.org/abs/2208.00799>
28. Deng, L., Liu, Y., Wang, W. (2020). *Biogas Technology*. Springer: Singapore. pp. 109–156. ISBN 978-981-15-49-40-3.
29. Drogos, B. (2013). *Biogas Process Monitoring—Techniques and Recommendations*. Available online: [http://task37.ieabioenergy.com/files/daten-redaktion/download/publications/Workshops/2013\\_Switzerland/4\\_Biogas%20process%](http://task37.ieabioenergy.com/files/daten-redaktion/download/publications/Workshops/2013_Switzerland/4_Biogas%20process%20)



20monitoring\_techniques%20and%20recommendations\_Drosg.pdf (accessed on 7 2020).

30. Dyvak, M., Melnyk, A., Rot, A., Hernes, M., Pukas, A. (2022). "Ontology of Mathematical Modeling Based on Interval Data". *Complexity*, vol. 2022, Article ID 8062969, 19 pages. <https://doi.org/10.1155/2022/8062969>.

31. Dyvak, M., Papa, O., Melnyk, A., Pukas, A., Porplytsya, N., Rot, A. Interval model of the efficiency of the functioning of information web resources for services on ecological expertise (2020) *Mathematics*, 8 (12), art. no. 2116, pp. 1-12.

32. Dyvak, M., Porplytsya, N., Maslyak, Y., Shynkaryk, M. (2018). Method of Parametric Identification for Interval Discrete Dynamic Models and the Computational Scheme of Its Implementation. In: Shakhovska, N., Stepashko, V. (eds) *Advances in Intelligent Systems and Computing II. CSIT 2017. Advances in Intelligent Systems and Computing*, vol 689. Springer, Cham.

33. Dyvak, N.P., Manzhula, V.I. (2008). "Structural Identification of Interval Models of the Static Systems". *Journal of Automation and Information Sciences*, 40(4), 49-61.

34. FNR. (2010). *Guide to Biogas—From Production to Use*. Deutsche Gesellschaft für Internationale Zusammenarbeit: Eschborn, Germany.

35. Foramitti, J. (2021). AgentPy: A package for agent-based modeling in Python. *J. Open Source Softw.* 6, 3065.

36. Gas Market Report, Q1-2024, International Energy Agency. Available at: <https://www.iea.org/reports/gas-market-report-q1-2024>

37. Ghanavati, H. (2018). *Biogas Production Systems: Operation, Process Control, and Troubleshooting*. Springer: Berlin/Heidelberg, Germany. pp. 199–219.

38. Global Optimization Toolbox. [Online]. Available at: <https://www.mathworks.com/products/global-optimization.html>

39. Golub, N., Potapova, M., Karpenko, Y. (2019). "Mathematical Modeling of the Biogas Production Process from the Distillery Spent Wash on the First Stage". *Innov Biosyst Bioeng*, vol. 3, no. 2, pp. 96-104.

40. Gopi, K., Mazumder, D., Crawford, J., Gadd, P., Tadros, C.V., Atanacio, A., Saintilan, N., Sammut, J. (2023). Developing a MySQL Database for the Provenance of Black Tiger Prawns (*Penaeus monodon*). *Foods* 12(14), 2677. <https://doi.org/10.3390/foods12142677>
41. Guo, X., He, Y. (2022). Mathematical Modeling and Optimization of Platform Service Supply Chains: A Literature Review. *Mathematics*, 10, 4307. <https://doi.org/10.3390/math10224307>
42. Hanum, F., Yuan, L.C., Kamahara, H., Aziz, H.A., Atsuta, Y., Yamada, T., Daimon, H. (2019). Treatment of sewage sludge using anaerobic digestion in Malaysia: Current state and challenges. *Front. Energy Res.*, 7, 19.
43. Heiker, M., Kraume, M., Mertins, A., Wawer, T., Rosenberger, S. (2021). Biogas Plants in Renewable Energy Systems—A Systematic Review of Modeling Approaches of Biogas Production. *Appl. Sci.*, 11, 3361. <https://doi.org/10.3390/app11083361>
44. Heiker M., Mertins A., Rosenberger S., Wawer T. Competition in the conversion of the gas grid: Is the future of biogas biomethane or hydrogen?, *International Journal of Hydrogen Energy*, Volume 48, Issue 83, 2023, Pages 32469-32484, ISSN 0360-3199, <https://doi.org/10.1016/j.ijhydene.2023.04.270>.
45. Hemlata Karne, Aayush Bendre, Amartya Singh. (2023). Mathematical modelling of a pilot scale biogas plant. *Materials Today: Proceedings*. <https://doi.org/10.1016/j.matpr.2023.01.303>
46. Hernandez, M.J. (2018). *Database Design for Mere Mortals: A Hands-On Guide to Relational Database Design*. Addison-Wesley Professional. 763p.
47. Hülsemann, B., Baumgart, M., Lenz, L., Elviliana, Föllmer, M., Sailer, G., Dinkler, K., Oechsner, H. (2023). Coupled Biogas and Fiber Production from Agricultural Residues and Energy Crops with Steam Explosion Treatment. *Applied Biosciences* 2(2), 278-291. <https://doi.org/10.3390/applbiosci2020019>

48. Ivakhnenko, A., Ivakhnenko, G. (1995). The review of problems solvable by algorithms of the Group Method of Data Handling (GMDH). *Pattern Recognition and Image Analysis*, vol. 5, № 4, c. 527-535.
49. Ivakhnenko, A.G. (1966). Group Method of Data Handling - A Rival of the Method of Stochastic Approximation. *Soviet Automatic Control*, 13, 43-71.
50. Jaxa-Rozen, M., Kwakkel, J.H. (2018). PyNetLogo: Linking NetLogo with Python. *J. Artif. Soc. Soc. Simul.*, 21, 4.
51. Karaboğa, D. (2005). An Idea Based on Honey Bee Swarm For Numerical Optimization.
52. Karlsson, J. (2017). Modeling and simulation of existing biogas plants with SIMBA#Biogas.
53. Keshtkar, A., Abolhamd, G., Meyssami, B., Ghaforian, H. (2003). Modeling of Anaerobic Digestion of Complex Substrates. *Iranian Journal of Chemistry and Chemical Engineering*, 22, 61-74.
54. Kim, Y., Kim, J., Lee, W. (2023). Effect of Block-Based Python Programming Environment on Programming Learning. *Applied Sciences*, 13(19), 10898. <https://doi.org/10.3390/app131910898>
55. Komarysta, B., Dzhygyrey, I., Bendiuh, V., Yavorovska, O., Andreeva, A., Berezenko, K., Meshcheriakova, I., Vovk, O., Dokshyna, S., Maidanskyi, I. (2023). Optimizing biogas production using artificial neural network. *Eastern-European Journal of Enterprise Technologies*, 2, 53-64. <https://doi.org/10.15587/1729-4061.2023.276431>
56. Kougias, P.G., Angelidaki, I. (2018). Biogas and its opportunities—A review. *Front. Environ. Sci. Eng.*, 12, 14.
57. Kumar, A., Negi, G., Pant, S., Ram, M., Dimri, S. C. (2021). Availability-cost optimization of butter oil processing system by using nature inspired optimization algorithms. *Reliab. Theory Appl.*, SI 2, c. 188-200.
58. Lamb, J.J., Bernard, O., Sarker, S., Lien, K.M., Hjelme, D.R. (2019). Perspectives of optical colourimetric sensors for anaerobic digestion. *Renew. Sust. Energ. Rev.*, 111, 87–96.

59. Laskri, N., Nedjah, N. (2015). Comparative Study for Biogas Production from Different Wastes. *International Journal of Bio-Science and Bio-Technology*, 7, 39-46. <https://doi.org/10.14257/ijbsbt.2015.7.4.05>
60. Lin, Q., de Vrieze, J., Li, C., Li, J., Li, J., Yao, M., Hedenec, P., Li, H., Li, T., Rui, J., et al. (2017). Temperature regulates deterministic processes and the succession of microbial interactions in anaerobic digestion process. *Water Res.*, 123, 134–143.
61. Lin, T., Ma, S., Ye, Y., Zhang, S. (2021). "An ADMM-based interior-point method for large-scale linear programming". *Optimization Methods and Software*, 36(2-3), 389-424. <https://doi.org/10.1080/10556788.2020.1821200>.
62. Ljung, L. (1999). *System Identification: Theory for the User*. Prentice Hall, 608 p.
63. M. Dyvak, "Parameters Identification Method of Interval Discrete Dynamic Models of Air Pollution Based on Artificial Bee Colony Algorithm," 2020 10th International Conference on Advanced Computer Information Technologies (ACIT), Deggendorf, Germany, 2020, pp. 130-135. <https://doi.org/10.1109/ACIT49673.2020.9208972>.
64. M. Dyvak, A. Pukas, I. Oliynyk, A. Melnyk, "Selection the 'Saturated' Block from Interval System of Linear Algebraic Equations for Recurrent Laryngeal Nerve Identification," 2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP), Lviv, Ukraine, 2018, pp. 444-448. <https://doi.org/10.1109/DSMP.2018.8478528>.
65. M. Dyvak, A. Pukas, V. Manzhula, N. Kasatkina, M. Komar, V. Zabchuk, "The Task of Parametric Identification the Interval Models with Nonlinear Parameters," 2022 12th International Conference on Advanced Computer Information Technologies (ACIT), Spisska Kapitula, Slovakia, 26-28 Sept. 2022, pp. 106-111.
66. M. Dyvak, I. Darmorost, N. Porplytsya, I. Hural, "Structure Identification of Difference Equations with Interval Estimates of their Parameters," 2019 IEEE 15th International Conference on the Experience of Designing and Application of CAD

Systems (CADSM), Polyana, Ukraine, 2019, pp. 1-4.  
<https://doi.org/10.1109/CADSM.2019.8779308>.

67. M. Dyvak, O. Kozak, A. Pukas, "Interval model for identification of laryngeal nerves," *Przegląd Elektrotechniczny*, vol. 86, no. 1, pp. 139-140, 2010.

68. M. Dyvak, V. Manzhula, O. Kozak, "New Method Tolerance Estimation of the Parameters Set of Interval Model Based on Saturated Block of ISLAE," 2007 9th International Conference - The Experience of Designing and Applications of CAD Systems in Microelectronics, Lviv, Ukraine, 2007, pp. 376-379.  
<https://doi.org/10.1109/CADSM.2007.4297587>.

69. M. Dyvak, V. Manzhula, T. Dyvak, "Identification of parameters of interval nonlinear models of static systems using multidimensional optimization," *Computational Problems of Electrical Engineering*, vol. 12, issue 2, pp. 5-13, 2022.  
<https://doi.org/10.23939/jcpee2022.02.005>.

70. M. Dyvak, V. Manzhula, Yu. Trufanova, "Interval Non-linear Model of Information Signal Characteristics Distribution for Detection of Recurrent Laryngeal Nerve during Thyroid Surgery," *Proceedings of the 5th International Conference on Informatics & Data-Driven Medicine (IDDM-2022), CEUR Workshop Proceedings, 2022*, vol. 3302, pp. 99–107.

71. Manzhula, V., Dyvak, M., Zabchuk, V. (2024). "The Improved Method for Identifying Parameters of Interval Nonlinear Models of Static Systems". *International Journal of Computing*, 23(1), 19-25. <https://doi.org/10.47839/ijc.23.1.3431>.

72. Mao, Z., Han, Q., He, Y., Li, N., Li, C., Shan, Z., Han, S. (2023). "A Software Testing Workflow Analysis Tool Based on the ADCV Method". *Electronics*, 12(21), 4464. <https://doi.org/10.3390/electronics12214464>.

73. Masuoka, C. (2008). *Java Programming Using Voice Input: Adding Java Support to Voicecode*. University of Maryland at College Park: College Park, MD, USA.

74. Migallón, V., Penadés, J. (2023). "A Java Application for Teaching Graphs in Undergraduate Courses". *Appl. Sci.*, 13, 12945.  
<https://doi.org/10.3390/app132312945>.

75. Moore, R. E. (2014). *Reliability in Computing: The Role of Interval Methods in Scientific Computing*. Elsevier.
76. Moroz, O.G., Stepashko, V.S. (2016). "Combinatorial algorithm of MGUA with genetic search of the model of optimal complexity". In *Proceedings of the International Conference on Intellectual Systems for Decision Making and Problems of Computational Intelligence*, pp. 297-299.
77. MySQL. (2021). *MySQL Workbench 8.0*. Oracle Corporation: Austin, TX, USA.
78. Nguyen, D., Gadhamshetty, V., Nitayavardhana, S., Khanal, S.K. (2015). "Automatic process control in anaerobic digestion technology: A critical review". *Bioresour. Technol.*, 193, 513-522.
79. Nixon, R. (2014). *Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5*. 4th ed. O'Reilly Media. 812 p.
80. Nourisa, J., Zeller-Plumhoff, B., Willumeit-Römer, R. (2022). "CppyABM: An open-source agent-based modeling library to integrate C++ and Python". *Softw. Pract. Exp.*, 52, 1337-1351.
81. Önen, S., Kuchta, K. (2020). "Evaluation of temperature changes in anaerobic digestion process". *Detritus*, 10(6), 11-18.
82. Önen, S., Nsair, A., Kuchta, K. (2019). "Innovative operational strategies for biogas plant including temperature and stirring management". *Waste Manag. Res.*, 37, 237-246.
83. Poggio, D., Walker, M., Nimmo, W., Ma, L., Pourkashanian, M. (2016). *Modelling the anaerobic digestion of solid organic waste – Substrate characterisation method for ADM1 using a combined biochemical and kinetic parameter estimation approach*. *Waste Management* 53, 40-54.
84. Porplytsya, N., Dyvak, M., Spivak, I., Voytyuk, I. (2015). "Mathematical and algorithmic foundations for implementation of the method for structure identification of interval difference operator based on functioning of bee colony". In *The Experience of*

Designing and Application of CAD Systems in Microelectronics, Lviv, Ukraine, pp. 196-199. <https://doi.org/10.1109/CADSM.2015.7230834>.

85. Pukas, A., Simak, A., Goncharuk-Cholach, T., Konoplitska, O., Zabchuk, V. (2023). "Intelligent Analyzing Module in the Academic Staff Performance Appraisal System". CEUR-WS, Vol. 3624, pp. 423-429.

86. Pukas, A., Smal, V., Zabchuk, V. (2018). "Software based on blockchain technology for consolidation the medical data about the patients examination". CEUR-WS, Vol. 2300, pp. 170-174.

87. Quarteroni, Alfio; Saleri, Fausto (2006). *Scientific Computing with MATLAB and Octave*. Springer. ISBN 978-3-540-32612-0.

88. Ramachandran, A., Rustum, R., Adeloje, A. J. (2019). "Review of Anaerobic Digestion Modeling and Optimization Using Nature-Inspired Techniques". *Processes*, 7, 953.

89. Sari, T. (2022). "Best Operating Conditions for Biogas Production in Some Simple Anaerobic Digestion Models". *Processes*, 10, 258. <https://doi.org/10.3390/pr10020258>.

90. Scarlat, N., Dallemand, J.-F., Fahl, F. (2018). "Biogas: Developments and perspectives in Europe". *Renew. Energy*, 129, 457–472.

91. Scheme of the biogas plant. <https://biteco-energy.com/ua/help/1575899636/id39/>.

92. Schildt, H. (2016). *Java 8. Complete Guide*. Williams. 1376 p.

93. Shpilevskiy, V. V., Sapronov, Yu. A., Salashenko, T. I. (2013). "Analytical aspects of assessment of energy dependence in petroleum-deficit countries". *Problems of the econom*, No. 1, 14-21.

94. Slowik, A. (2020). *Swarm Intelligence Algorithms: Modification and Applications*, 1st ed. CRC Press: Boca Raton, FL, USA. <https://doi.org/10.1201/9780429422607>

95. Song, Y.-J., Oh, K.-S., Lee, B., Pak, D.-W., Cha, J.-H., Park, J.-G. (2021). "Characteristics of Biogas Production from Organic Wastes Mixed at Optimal Ratios in

an Anaerobic Co-Digestion Reactor". *Energies*, 14, 6812. <https://doi.org/10.3390/en14206812>.

96. Steiniger, B., Hupfauf, S., Insam, H., Schaum, C. (2023). "Exploring Anaerobic Digestion from Mesophilic to Thermophilic Temperatures—Operational and Microbial Aspects". *Fermentation*, 9(9), 798.

97. Sun, C., Cao, W., Banks, C. J., Heaven, S., Liu, R. (2016). "Biogas production from undiluted chicken manure and maize silage: A study of ammonia inhibition in high solids anaerobic digestion". *Bioresour. Technol.*, 218, 1215–1223.

98. Suo, L., Ren, J., Zhao, Z., Zhai, C. (2020). "Study on the Nonlinear Dynamics of the Continuous Stirred Tank Reactors". *Processes*, 8, 1436. <https://doi.org/10.3390/pr8111436>.

99. Svendsen, D. H., Martino, L., Camps-Valls, G. (2020). "Active Emulation of Computer Codes with Gaussian Processes—Application to Remote Sensing". *Pattern Recognit.*, 100, 107103.

100. Tolerances and Stopping Criteria [Online]. Available at: <https://se.mathworks.com/help/optim/ug/tolerances-and-stopping-criteria.html>

101. Vadan, Andrei-Mihai, and Liviu-Cristian Miclea. (2023). "Software Testing Techniques for Improving the Quality of Smart-Home IoT Systems". *Electronics*, 12(6), 1337. <https://doi.org/10.3390/electronics12061337>.

102. Vanderbei, R. (2014). *Linear Programming: Foundations and Extensions*. Springer US. <https://doi.org/10.1007/978-1-4614-7630-6>.

103. Wallace, R. M., Turchenko, V., Sheikhalishahi, M., Turchenko, I., Shults, V., Vazquez-Poletti, J. L., Grandinetti, L. (2013). "Applications of neural-based spot market prediction for cloud computing". In *Proceedings of the 2013 IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS)*, Berlin, Germany, pp. 710-716. <https://doi.org/10.1109/IDAACS.2013.6663017>.

104. Waltz, R. A., Morales, J. L., Nocedal, J., Orban, D. (2006). "An interior algorithm for nonlinear optimization that combines line search and trust region steps".



Mathematical Programming, 107(3), 391-408. <https://doi.org/10.1007/s10107-004-0560-5>.

105. Xiao, Ws., Li, Gx., Liu, C. et al. (2023). "A novel chaotic and neighborhood search-based artificial bee colony algorithm for solving optimization problems". *Sci Rep*, 13, 20496. <https://doi.org/10.1038/s41598-023-44770-8>.

106. Yenigün, O., Demirel, B. (2013). "Ammonia inhibition in anaerobic digestion: A review". *Process Biochem.*, 48, 901–911.

107. Yu, D., Liu, J., Sui, Q., Wei, Y. (2015). Biogas-pH automation control strategy for optimizing organic loading rate of anaerobic membrane bioreactor treating high COD wastewater. *Bioresource Technology* 203. <https://doi.org/10.1016/j.biortech.2015.12.010>

108. Yu, Dawei, Liu, Jibao, Sui, Qianwen, Wei, Yuansong. (2015). "Biogas-pH automation control strategy for optimizing organic loading rate of anaerobic membrane bioreactor treating high COD wastewater". *Bioresource Technology*, 203. <https://doi.org/10.1016/j.biortech.2015.12.010>.

109. Zepter, J.M., Gabderakhmanova, T., Andreasen, K.M., Boesgaard, K., Marinelli, M. (2020). "Biogas plant modelling for flexibility provision in the power system of Bornholm island". In *Proceedings of the 2020 55th International Universities Power Engineering Conference (UPEC)*, Turin, Italy, 1–4 September 2020, pp. 1–6.

110. Zepter, Jan Martin, Jan Engelhardt, Tatiana Gabderakhmanova, Mattia Marinelli. (2021). "Empirical Validation of a Biogas Plant Simulation Model and Analysis of Biogas Upgrading Potentials". *Energies*, 14(9), 2424. <https://doi.org/10.3390/en14092424>.

111. Zhang, C., Sun, Y., Cao, T., Wang, W., Huo, S., Liu, Z.-H. (2022). Influence of organic load on biogas production and response of microbial community in anaerobic digestion of food waste. *Int. J. Hydrogen Energy* 47(77), 32849-32860.

112. Гаценко К. В., Волошин М. Д. (2019). "Технологія отримання біогазу на основі харчових відходів". *Збірник наукових праць Дніпровського державного технічного університету. Технічні науки*, Вип. 1, 131-136.

113. Герасимов В. В., Матвеева Н. О. Розробка програмного забезпечення на платформі Java. Багатопоточне програмування і паралельні обчислення: навчальний посібник — Дніпро, 2020. — 174 с.

114. Дивак М. П. Задачі математичного моделювання статичних систем з інтервальними даними: монографія / М. П. Дивак. - Т. : Економ. думка ТНЕУ, 2011. - 215 с.

115. Дивак М. П. Прикладні задачі структурної та параметричної ідентифікації інтервальних моделей складних об'єктів [Електронний ресурс] : монографія / М. П. Дивак, А. В. Пукас, Н. П. Порплиця, А. М. Мельник. - Тернопіль : Університетська думка, 2021. - 212 с.

116. Дивак М., Забчук В. Моделювання характеристик процесів у біогазових установках на основі аналізу інтервальних даних / М. Дивак, В. Забчук/ Вісник Хмельницького національного університету. Серія: Технічні науки. Том 331 № 1 (2024), С.180-190. <https://doi.org/10.31891/2307-5732-2024-331-28>

117. Дивак М.П., Гураль І.В. Біохімічний аналіз процесів в біогазових установках та його застосування в задачі макромоделювання процесів виробництва біогазу. Вимірювальна та обчислювальна техніка в технологічних процесах. 2014. №2. С. 152–158.

118. Дивак М.П., Гураль І.В.. Інтервальне представлення динаміки анаеробного мікробіологічного бродіння в біогазових установках // Індуктивне моделювання складних систем: Зб. наук. пр. — К.: МННЦ ІТС НАН та МОН України, 2014. — Вип. 6. — С. 55-68.

119. Дивак Т., Забчук В. Огляд і аналіз методів та засобів моделювання процесів у біогазових установках. Наукові праці / Т. Дивак, В. Забчук/ Донецького національного технічного університету. Серія: “Інформатика, кібернетика та обчислювальна техніка”, №1 (38)2024, С.57-70. <https://doi.org/10.31474/1996-1588>

120. Дивак, М. П. Ідентифікація дискретних моделей систем з розподіленими параметрами на основі аналізу інтервальних даних [Текст] :

монографія / М. П. Дивак, Н. П. Порплиця, Т. М. Дивак. – Тернопіль: Економічна думка ТНЕУ, 2018. – 220 с.

121. Забчук, В. В. Аналіз методів побудови системи доповненої реальності для тестування практичних навиків [Текст] / В. В. Забчук // Комп'ютерні інформаційні технології : матеріали школи-семінару молодих вчених і студентів СІТ'2019 [м. Тернопіль, 29 листопада 2019 р.] / відп. за вип. М. П. Дивак. - Тернопіль : ТНЕУ, 2019. - С. 9.

122. Іванов В., Онищенко А., Засуха Л., Конкс Т., Кучер С. Біогазова установка для утилізації гною на свинокомплексі. Наука і техніка сьогодні. – 2022. - № 11 (11). – С. 298-306.

123. Красносельська А.О., Веселов О.В., Забчук В.В. (2023). "Інтелектуалізована система для оптимізації процесу формування документації програмного забезпечення". В Комп'ютерні інформаційні технології: матеріали школи-семінару молодих вчених і студентів СІТ'2023, Тернопіль, 30 листоп. 2023 р., відп. за вип. А. В. Пукас, 100-101.

124. Манжула В. І., Забчук В.В.. Архітектура програмного забезпечення для моделювання процесів у біогазових установках». Наукові праці Вінницького національного технічного університету, вип. 1 2024. <https://doi.org/10.31649/2307-5376-2024-1-19-28>.

125. Мельник А.М. Архітектура програмного забезпечення для математичного моделювання на основі інтервального та онтологічного підходу. Вісник Хмельницького національного університету. Технічні науки. 2022. №3. С. 265–273.

## ДОДАТОК А

### DDL БАЗИ ДАНИХ ДЛЯ МОДЕЛЮВАННЯ ПРОЦЕСІВ У БІОГАЗОВИХ УСТАНОВКАХ

```

-----
-- Schema biogas
-----
CREATE SCHEMA IF NOT EXISTS `biogas` DEFAULT CHARACTER SET
utf8mb4 ;
SHOW WARNINGS;
USE `biogas` ;

-----
-- Table `biogasplants`
-----
DROP TABLE IF EXISTS `biogasplants` ;

SHOW WARNINGS;
CREATE TABLE IF NOT EXISTS `biogasplants` (
  `•`  biogas_plant_id` INT(11) NOT NULL,
  `•`  name` VARCHAR(45) NULL DEFAULT NULL,
  `•`  location` VARCHAR(45) NULL DEFAULT NULL,
  `•`  capacity` DOUBLE NULL DEFAULT NULL,
  PRIMARY KEY (`•`  biogas_plant_id`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4;

SHOW WARNINGS;

-----
-- Table `biogasproductiondata`
-----
DROP TABLE IF EXISTS `biogasproductiondata` ;

SHOW WARNINGS;
CREATE TABLE IF NOT EXISTS `biogasproductiondata` (
  `•`  biogas_production_id` INT(11) NOT NULL,
  `•`  digester_id` INT(11) NULL DEFAULT NULL,
  `•`  timestamp` DATETIME NULL DEFAULT NULL,
  `•`  gas_production_rate` DECIMAL(10,0) NULL DEFAULT NULL,
  PRIMARY KEY (`•`  biogas_production_id`))
ENGINE = InnoDB

```

```

DEFAULT CHARACTER SET = utf8mb4;

SHOW WARNINGS;

-----
-- Table `biogasproductionmodels`
-----
DROP TABLE IF EXISTS `biogasproductionmodels` ;

SHOW WARNINGS;
CREATE TABLE IF NOT EXISTS `biogasproductionmodels` (
  `• biogas_production_model_id` INT(11) NOT NULL,
  `• name` VARCHAR(45) NULL DEFAULT NULL,
  `• description` VARCHAR(45) NULL DEFAULT NULL,
  `• parameters:` VARCHAR(45) NULL DEFAULT NULL,
  PRIMARY KEY (`• biogas_production_model_id`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4;

SHOW WARNINGS;

-----
-- Table `digesters`
-----
DROP TABLE IF EXISTS `digesters` ;

SHOW WARNINGS;
CREATE TABLE IF NOT EXISTS `digesters` (
  `• digester_id` INT(11) NOT NULL,
  `• biogas_plant_id` INT(11) NULL DEFAULT NULL,
  `• volume` DOUBLE NULL DEFAULT NULL,
  `• temperature` DOUBLE NULL DEFAULT NULL,
  `• pressure` DOUBLE NULL DEFAULT NULL,
  PRIMARY KEY (`• digester_id`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4;

SHOW WARNINGS;

-----
-- Table `feedstockcomposition`
-----
DROP TABLE IF EXISTS `feedstockcomposition` ;

```

```

SHOW WARNINGS;
CREATE TABLE IF NOT EXISTS `feedstockcomposition` (
  `• feedstock_composition_id` INT(11) NOT NULL,
  `• feedstock_id` INT(11) NULL DEFAULT NULL,
  `• component_name` VARCHAR(45) NULL DEFAULT NULL,
  `• percentage` DECIMAL(10,0) NULL DEFAULT NULL,
  PRIMARY KEY (`• feedstock_composition_id`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4;

```

```

SHOW WARNINGS;

```

```

-----
-- Table `feedstockmodels`
-----
DROP TABLE IF EXISTS `feedstockmodels` ;

```

```

SHOW WARNINGS;
CREATE TABLE IF NOT EXISTS `feedstockmodels` (
  `• feedstock_model_id` INT(11) NOT NULL,
  `• name` VARCHAR(45) NULL DEFAULT NULL,
  `• description` VARCHAR(45) NULL DEFAULT NULL,
  `• parameters` VARCHAR(45) NULL DEFAULT NULL,
  PRIMARY KEY (`• feedstock_model_id`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4;

```

```

SHOW WARNINGS;

```

```

-----
-- Table `feedstocks`
-----
DROP TABLE IF EXISTS `feedstocks` ;

```

```

SHOW WARNINGS;
CREATE TABLE IF NOT EXISTS `feedstocks` (
  `idFeedstocks` INT(11) NOT NULL,
  `• biogas_plant_id` INT(11) NULL DEFAULT NULL,
  `• type` VARCHAR(45) NULL DEFAULT NULL,
  `• quantity` DOUBLE NULL DEFAULT NULL,
  PRIMARY KEY (`idFeedstocks`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4;

```

```
SHOW WARNINGS;
```

```
-----  
-- Table `gasholders`  
-----
```

```
DROP TABLE IF EXISTS `gasholders` ;
```

```
SHOW WARNINGS;
```

```
CREATE TABLE IF NOT EXISTS `gasholders` (  
  `•` gas_holder_id` INT(11) NOT NULL,  
  `•` biogas_plant_id` INT(11) NULL DEFAULT NULL,  
  `•` volume` DOUBLE NULL DEFAULT NULL,  
  `•` gas_level` DOUBLE NULL DEFAULT NULL,  
  PRIMARY KEY (`•` gas_holder_id`))  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8mb4;
```

```
SHOW WARNINGS;
```

```
-----  
-- Table `gasqualitydata`  
-----
```

```
DROP TABLE IF EXISTS `gasqualitydata` ;
```

```
SHOW WARNINGS;
```

```
CREATE TABLE IF NOT EXISTS `gasqualitydata` (  
  `•` gas_quality_data_id` INT(11) NOT NULL,  
  `•` gas_holder_id` INT(11) NOT NULL,  
  `•` timestamp` DATE NULL DEFAULT NULL,  
  `•` methane_content` VARCHAR(45) NULL DEFAULT NULL,  
  `•` carbon_dioxide_content` VARCHAR(45) NULL DEFAULT NULL,  
  `•` other_impurities` VARCHAR(45) NULL DEFAULT NULL,  
  PRIMARY KEY (`•` gas_quality_data_id`))  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8mb4;
```

```
SHOW WARNINGS;
```

```
-----  
-- Table `gasqualitymodels`  
-----
```

```
DROP TABLE IF EXISTS `gasqualitymodels` ;
```

```
SHOW WARNINGS;
```

```

CREATE TABLE IF NOT EXISTS `gasqualitymodels` (
  `gas_quality_model_id` INT(11) NOT NULL,
  `name` VARCHAR(45) NULL DEFAULT NULL,
  `description` VARCHAR(45) NULL DEFAULT NULL,
  `parameters` VARCHAR(45) NULL DEFAULT NULL,
  PRIMARY KEY (`gas_quality_model_id`)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4;

SHOW WARNINGS;

-----
-- Table `gasutilizationsystems`
-----

DROP TABLE IF EXISTS `gasutilizationsystems` ;

SHOW WARNINGS;
CREATE TABLE IF NOT EXISTS `gasutilizationsystems` (
  `gas_utilization_system_id` INT(11) NOT NULL,
  `biogas_plant_id` INT(11) NULL DEFAULT NULL,
  `connected_devices` VARCHAR(45) NULL DEFAULT NULL,
  PRIMARY KEY (`gas_utilization_system_id`)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4;

SHOW WARNINGS;

-----
-- Table `pressurecontrolsystems`
-----

DROP TABLE IF EXISTS `pressurecontrolsystems` ;

SHOW WARNINGS;
CREATE TABLE IF NOT EXISTS `pressurecontrolsystems` (
  `pressure_control_system_id` INT(11) NOT NULL,
  `digester_id` INT(11) NULL DEFAULT NULL,
  `target_pressure` DOUBLE NULL DEFAULT NULL,
  `current_pressure` DOUBLE NULL DEFAULT NULL,
  PRIMARY KEY (`pressure_control_system_id`)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4;

SHOW WARNINGS;

```



```
-----
-- Table `pressuredata`
-----
```

```
DROP TABLE IF EXISTS `pressuredata` ;
```

```
SHOW WARNINGS;
```

```
CREATE TABLE IF NOT EXISTS `pressuredata` (
  `•` pressure_data_id` INT(11) NOT NULL,
  `•` digester_id` INT(11) NULL DEFAULT NULL,
  `•` timestamp` DATE NULL DEFAULT NULL,
  `•` pressure` DOUBLE NULL DEFAULT NULL,
  PRIMARY KEY (`•` pressure_data_id`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4;
```

```
SHOW WARNINGS;
```

```
-----
-- Table `pressuremodels`
-----
```

```
DROP TABLE IF EXISTS `pressuremodels` ;
```

```
SHOW WARNINGS;
```

```
CREATE TABLE IF NOT EXISTS `pressuremodels` (
  `•` pressure_model_id` INT(11) NOT NULL,
  `•` name` VARCHAR(45) NULL DEFAULT NULL,
  `•` description` VARCHAR(45) NULL DEFAULT NULL,
  `•` parameters` VARCHAR(45) NULL DEFAULT NULL,
  PRIMARY KEY (`•` pressure_model_id`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4;
```

```
SHOW WARNINGS;
```

```
-----
-- Table `temperaturecontrolsystems`
-----
```

```
DROP TABLE IF EXISTS `temperaturecontrolsystems` ;
```

```
SHOW WARNINGS;
```

```
CREATE TABLE IF NOT EXISTS `temperaturecontrolsystems` (
  `•` temperature_control_system_id` INT(11) NOT NULL,
  `•` digester_id` INT(11) NULL DEFAULT NULL,
  `•` target_temperature` DOUBLE NULL DEFAULT NULL,
```

```

`• current_temperature` DOUBLE NULL DEFAULT NULL,
PRIMARY KEY (`• temperature_control_system_id`)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4;

```

```
SHOW WARNINGS;
```

```

-----
-- Table `temperaturedata`
-----
DROP TABLE IF EXISTS `temperaturedata` ;

```

```

SHOW WARNINGS;
CREATE TABLE IF NOT EXISTS `temperaturedata` (
`• temperature_data_id` INT(11) NOT NULL,
`• digester_id` INT(11) NULL DEFAULT NULL,
`• timestamp` DATE NULL DEFAULT NULL,
`• temperature` FLOAT NULL DEFAULT NULL,
PRIMARY KEY (`• temperature_data_id`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4;

```

```
SHOW WARNINGS;
```

```

-----
-- Table `temperaturemodels`
-----
DROP TABLE IF EXISTS `temperaturemodels` ;

```

```

SHOW WARNINGS;
CREATE TABLE IF NOT EXISTS `temperaturemodels` (
`• temperature_model_id` INT(11) NOT NULL,
`• name` VARCHAR(45) NULL DEFAULT NULL,
`• description` VARCHAR(45) NULL DEFAULT NULL,
`• parameters` VARCHAR(45) NULL DEFAULT NULL,
PRIMARY KEY (`• temperature_model_id`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4;

```

```
SHOW WARNINGS;
```

```

SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;

```

## ДОДАТОК Б

### ЛІСТИНГ КОДУ ОСНОВНИХ МОДУЛІВ ПРОГРАМИ

#### 1. BiogasPlant

```

public class BiogasPlant {
    private Digester digester;
    private GasHolder gasHolder;
    private GasUtilizationSystem gasUtilizationSystem;

    public BiogasPlant(Digester digester, GasHolder gasHolder, GasUtilizationSystem
gasUtilizationSystem) {
        this.digester = digester;
        this.gasHolder = gasHolder;
        this.gasUtilizationSystem = gasUtilizationSystem;
    }

    public void start() {
        System.out.println("Starting the biogas plant...");
        digester.processFeedstock();
        digester.generateBiogas();
        System.out.println("Biogas plant started.");
    }

    public void stop() {
        System.out.println("Stopping the biogas plant...");
        // Дії, що виконуються при зупинці
        System.out.println("Biogas plant stopped.");
    }

    public void update() {
        System.out.println("Updating the biogas plant...");
        digester.adjustTemperature();
        digester.adjustPressure();
        System.out.println("Biogas plant updated.");
    }
}

```

#### 2. Digester

```

public class Digester {
    private Feedstock feedstock;
    private TemperatureControlSystem temperatureControlSystem;
    private PressureControlSystem pressureControlSystem;

    public Digester(Feedstock feedstock, TemperatureControlSystem temperatureControlSystem,
PressureControlSystem pressureControlSystem) {
        this.feedstock = feedstock;
        this.temperatureControlSystem = temperatureControlSystem;
        this.pressureControlSystem = pressureControlSystem;
    }

    public void processFeedstock() {
        System.out.println("Processing feedstock: " + feedstock.getType() + " (" +
feedstock.getQuantity() + " kg)");
        // Логіка для обробки сировини
    }
}

```

```

public void generateBiogas() {
    System.out.println("Generating biogas...");
    // Логіка для генерації біогазу
}

public void adjustTemperature() {
    temperatureControlSystem.regulateTemperature();
}

public void adjustPressure() {
    pressureControlSystem.regulatePressure();
}
}
3. Feedstock

```

```

public class Feedstock {
    private String type;
    private float quantity;

    public Feedstock(String type, float quantity) {
        this.type = type;
        this.quantity = quantity;
    }

    public String getType() {
        return type;
    }

    public float getQuantity() {
        return quantity;
    }
}
4. GasHolder

```

```

public class GasHolder {
    private float volume;
    private float currentGasLevel;

    public GasHolder(float volume) {
        this.volume = volume;
        this.currentGasLevel = 0.0f;
    }

    public void storeGas(float volume) {
        if (currentGasLevel + volume <= this.volume) {
            currentGasLevel += volume;
            System.out.println(volume + " m³ gas stored.");
        } else {
            System.out.println("Not enough capacity to store " + volume + " m³ gas.");
        }
    }

    public void releaseGas(float volume) {
        if (currentGasLevel >= volume) {
            currentGasLevel -= volume;
            System.out.println(volume + " m³ gas released.");
        } else {

```

```

        System.out.println("Not enough gas to release " + volume + " m³.");
    }
}

```

#### 5. GasUtilizationSystem

```

import java.util.ArrayList;
import java.util.List;

public class GasUtilizationSystem {
    private List<String> connectedDevices;

    public GasUtilizationSystem() {
        this.connectedDevices = new ArrayList<>();
    }

    public void connectDevice(String device) {
        if (!connectedDevices.contains(device)) {
            connectedDevices.add(device);
            System.out.println("Device " + device + " connected.");
        } else {
            System.out.println("Device " + device + " is already connected.");
        }
    }

    public void disconnectDevice(String device) {
        if (connectedDevices.contains(device)) {
            connectedDevices.remove(device);
            System.out.println("Device " + device + " disconnected.");
        } else {
            System.out.println("Device " + device + " is not connected.");
        }
    }

    public void utilizeGas(float volume) {
        System.out.println("Utilizing " + volume + " m³ gas.");
        // Логіка для використання газу
    }
}

```

#### 6. TemperatureControlSystem

```

public class TemperatureControlSystem {
    private float targetTemperature;
    private float currentTemperature;

    public TemperatureControlSystem(float initialTemperature, float targetTemperature) {
        this.currentTemperature = initialTemperature;
        this.targetTemperature = targetTemperature;
    }

    public void setTargetTemperature(float targetTemperature) {
        this.targetTemperature = targetTemperature;
        System.out.println("Target temperature set to " + targetTemperature + " °C.");
    }

    public float getCurrentTemperature() {
        return currentTemperature;
    }
}

```

```

public void regulateTemperature() {
    while (currentTemperature < targetTemperature) {
        currentTemperature += 0.1f;
    }
    while (currentTemperature > targetTemperature) {
        currentTemperature -= 0.1f;
    }
    System.out.println("Temperature regulated to " + currentTemperature + " °C.");
}
}
7. PressureControlSystem

```

```

public class PressureControlSystem {
    private float targetPressure;
    private float currentPressure;

    public PressureControlSystem(float initialPressure, float targetPressure) {
        this.currentPressure = initialPressure;
        this.targetPressure = targetPressure;
    }

    public void setTargetPressure(float targetPressure) {
        this.targetPressure = targetPressure;
        System.out.println("Target pressure set to " + targetPressure + " bar.");
    }

    public float getCurrentPressure() {
        return currentPressure;
    }

    public void regulatePressure() {
        while (currentPressure < targetPressure) {
            currentPressure += 0.1f;
        }
        while (currentPressure > targetPressure) {
            currentPressure -= 0.1f;
        }
        System.out.println("Pressure regulated to " + currentPressure + " bar.");
    }
}
8. BiogasQualityAnalyzer

```

```

public class BiogasQualityAnalyzer {
    public QualityReport analyzeGasQuality(GasSample gasSample) {
        float methaneContent = gasSample.getMethaneContent();
        float carbonDioxideContent = gasSample.getCarbonDioxideContent();
        boolean isQualityAcceptable = methaneContent > 50 && carbonDioxideContent < 50;

        return new QualityReport(methaneContent, carbonDioxideContent,
isQualityAcceptable);
    }
}

class GasSample {
    private float methaneContent;
    private float carbonDioxideContent;
}

```

```

public GasSample(float methaneContent, float carbonDioxideContent) {
    this.methaneContent = methaneContent;
    this.carbonDioxideContent = carbonDioxideContent;
}

public float getMethaneContent() {
    return methaneContent;
}

public float getCarbonDioxideContent() {
    return carbonDioxideContent;
}
}

class QualityReport {
    private float methaneContent;
    private float carbonDioxideContent;
    private boolean isQualityAcceptable;

    public QualityReport(float methaneContent, float carbonDioxideContent, boolean
isQualityAcceptable) {
        this.methaneContent = methaneContent;
        this.carbonDioxideContent = carbonDioxideContent;
        this.isQualityAcceptable = isQualityAcceptable;
    }

    @Override
    public String toString() {
        return "QualityReport{" +
            "methaneContent=" + methaneContent + "%" +
            ", carbonDioxideContent=" + carbonDioxideContent + "%" +
            ", isQualityAcceptable=" + isQualityAcceptable +
            '}' ;
    }
}
}
9. BiogasProductionModel

```

```

public class BiogasProductionModel {
    public BiogasProductionPrediction predictBiogasProduction(FeedstockData feedstockData)
{
        float baseBiogasYield = 0.05f;
        float quantity = feedstockData.getQuantity();
        float moistureContent = feedstockData.getMoistureContent();

        float effectiveQuantity = quantity * (1 - moistureContent / 100);
        float estimatedBiogasVolume = baseBiogasYield * effectiveQuantity;

        return new BiogasProductionPrediction(estimatedBiogasVolume);
    }

    public static void main(String[] args) {
        BiogasProductionModel model = new BiogasProductionModel();
        FeedstockData feedstockData = new FeedstockData("Organic Waste", 1000, 70);

        BiogasProductionPrediction prediction =
model.predictBiogasProduction(feedstockData);
        System.out.println(prediction);
    }
}

```

```
class FeedstockData {
    private String type;
    private float quantity;
    private float moistureContent;

    public FeedstockData(String type, float quantity, float moistureContent) {
        this.type = type;
        this.quantity = quantity;
        this.moistureContent = moistureContent;
    }

    public String getType() {
        return type;
    }

    public float getQuantity() {
        return quantity;
    }

    public float getMoistureContent() {
        return moistureContent;
    }
}

class BiogasProductionPrediction {
    private float estimatedBiogasVolume;

    public BiogasProductionPrediction(float estimatedBiogasVolume) {
        this.estimatedBiogasVolume = estimatedBiogasVolume;
    }

    public float getEstimatedBiogasVolume() {
        return estimatedBiogasVolume;
    }

    @Override
    public String toString() {
        return "BiogasProductionPrediction{" +
            "estimatedBiogasVolume=" + estimatedBiogasVolume + " m³" +
            '}';
    }
}
```



## ДОДАТОК В

### ЛІСТИНГ КОДУ РЕАЛІЗАЦІЇ МАТЕМАТИЧНИХ МОДЕЛЕЙ

```

import numpy as np

# Визначення цільової функції (тут ми використовуємо простий приклад з однією змінною)
def objective_function(x):
    return x**2 - 4*x + 4 # f(x) = x^2 - 4x + 4

# Генерація початкової популяції
def initialize_population(pop_size, dim, bounds):
    population = np.random.uniform(bounds[0], bounds[1], (pop_size, dim))
    return population

# Оновлення стану зайнятої бджоли
def update_employed_bee(population, fitness, bounds):
    new_population = np.copy(population)
    new_fitness = np.copy(fitness)
    for i in range(len(population)):
        phi = np.random.uniform(-1, 1, len(population[i]))
        partner = np.random.choice(list(range(i)) + list(range(i+1, len(population))))
        new_solution = population[i] + phi * (population[i] - population[partner])
        new_solution = np.clip(new_solution, bounds[0], bounds[1])
        new_solution_fitness = objective_function(new_solution)
        if new_solution_fitness < fitness[i]:
            new_population[i] = new_solution
            new_fitness[i] = new_solution_fitness
    return new_population, new_fitness

# Вибір найманих бджіл на основі ймовірності
def select_onlooker_bees(population, fitness, num_onlookers):
    fitness_prob = fitness / fitness.sum()
    selected_indices = np.random.choice(len(population), num_onlookers, p=fitness_prob)
    return population[selected_indices]

# Оновлення стану найманих бджіл
def update_onlooker_bee(population, fitness, selected_population, bounds):
    new_population = np.copy(population)
    new_fitness = np.copy(fitness)
    for sol in selected_population:
        phi = np.random.uniform(-1, 1, len(sol))
        partner = np.random.choice(len(population))
        new_solution = sol + phi * (sol - population[partner])
        new_solution = np.clip(new_solution, bounds[0], bounds[1])
        new_solution_fitness = objective_function(new_solution)
        if new_solution_fitness < fitness[np.where(population == sol)[0][0]]:
            new_population[np.where(population == sol)[0][0]] = new_solution
            new_fitness[np.where(population == sol)[0][0]] = new_solution_fitness
    return new_population, new_fitness

# Оновлення стану розвідувальних бджіл
def update_scout_bee(population, fitness, bounds, limit):
    for i in range(len(population)):
        if fitness[i] > np.mean(fitness):
            population[i] = np.random.uniform(bounds[0], bounds[1], len(population[i]))
            fitness[i] = objective_function(population[i])
    return population, fitness

```

```

# Алгоритм штучної бджолоїної колонії
def bee_colony_optimization(pop_size, dim, bounds, max_iter, limit):
    population = initialize_population(pop_size, dim, bounds)
    fitness = np.array([objective_function(ind) for ind in population])

    best_solution = population[np.argmin(fitness)]
    best_fitness = np.min(fitness)

    for iteration in range(max_iter):
        population, fitness = update_employed_bee(population, fitness, bounds)
        selected_population = select_onlooker_bees(population, fitness, pop_size // 2)
        population, fitness = update_onlooker_bee(population, fitness, selected_population,
bounds)
        population, fitness = update_scout_bee(population, fitness, bounds, limit)

        current_best = population[np.argmin(fitness)]
        current_best_fitness = np.min(fitness)

        if current_best_fitness < best_fitness:
            best_solution = current_best
            best_fitness = current_best_fitness

        print(f"Iteration {iteration+1}/{max_iter}, Best Fitness: {best_fitness}")

    return best_solution, best_fitness

# Параметри алгоритму
pop_size = 20
dim = 1
bounds = [-10, 10]
max_iter = 100
limit = 10

# Запуск алгоритму
best_solution, best_fitness = bee_colony_optimization(pop_size, dim, bounds, max_iter,
limit)
print(f"Best Solution: {best_solution}, Best Fitness: {best_fitness}")

import numpy as np

class Feedstock:
    def __init__(self, type, quantity):
        self.type = type
        self.quantity = quantity

    def getType(self):
        return self.type

    def getQuantity(self):
        return self.quantity

```

```
class GasHolder:
    def __init__(self, volume):
        self.volume = volume
        self.currentGasLevel = 0.0

    def storeGas(self, volume):
        if self.currentGasLevel + volume <= self.volume:
            self.currentGasLevel += volume
            print(f"{volume} m³ gas stored. Current level: {self.currentGasLevel} m³.")
        else:
            print(f"Not enough capacity to store {volume} m³ gas. Current level:
{self.currentGasLevel} m³.")

    def releaseGas(self, volume):
        if self.currentGasLevel >= volume:
            self.currentGasLevel -= volume
            print(f"{volume} m³ gas released. Current level: {self.currentGasLevel} m³.")
        else:
            print(f"Not enough gas to release {volume} m³. Current level:
{self.currentGasLevel} m³.")

class GasUtilizationSystem:
    def __init__(self):
        self.connectedDevices = []

    def connectDevice(self, device):
        if device not in self.connectedDevices:
            self.connectedDevices.append(device)
            print(f"Device {device} connected.")
        else:
            print(f"Device {device} is already connected.")

    def disconnectDevice(self, device):
        if device in self.connectedDevices:
            self.connectedDevices.remove(device)
            print(f"Device {device} disconnected.")
        else:
            print(f"Device {device} is not connected.")

    def utilizeGas(self, volume):
```

```
print(f"Utilizing {volume} m3 gas.")

class TemperatureControlSystem:
    def __init__(self, initialTemperature, targetTemperature):
        self.currentTemperature = initialTemperature
        self.targetTemperature = targetTemperature

    def setTargetTemperature(self, targetTemperature):
        self.targetTemperature = targetTemperature
        print(f"Target temperature set to {targetTemperature} °C.")

    def getCurrentTemperature(self):
        return self.currentTemperature

    def regulateTemperature(self):
        while self.currentTemperature < self.targetTemperature:
            self.currentTemperature += 0.1
        while self.currentTemperature > self.targetTemperature:
            self.currentTemperature -= 0.1
        print(f"Temperature regulated to {self.currentTemperature} °C.")

class PressureControlSystem:
    def __init__(self, initialPressure, targetPressure):
        self.currentPressure = initialPressure
        self.targetPressure = targetPressure

    def setTargetPressure(self, targetPressure):
        self.targetPressure = targetPressure
        print(f"Target pressure set to {targetPressure} bar.")

    def getCurrentPressure(self):
        return self.currentPressure

    def regulatePressure(self):
        while self.currentPressure < self.targetPressure:
            self.currentPressure += 0.1
        while self.currentPressure > self.targetPressure:
            self.currentPressure -= 0.1
        print(f"Pressure regulated to {self.currentPressure} bar.")
```

```

class BiogasQualityAnalyzer:
    def analyzeGasQuality(self, gasSample):
        methaneContent = gasSample.get('methaneContent', 0)
        carbonDioxideContent = gasSample.get('carbonDioxideContent', 0)
        isQualityAcceptable = methaneContent > 50 and carbonDioxideContent < 50
        return {
            'methaneContent': methaneContent,
            'carbonDioxideContent': carbonDioxideContent,
            'isQualityAcceptable': isQualityAcceptable,
        }

class Digester:
    def __init__(self, feedstock, temperatureControlSystem, pressureControlSystem):
        self.feedstock = feedstock
        self.temperatureControlSystem = temperatureControlSystem
        self.pressureControlSystem = pressureControlSystem

    def processFeedstock(self):
        print(f"Processing          feedstock:          {self.feedstock.getType()}
({self.feedstock.getQuantity()} kg)")
        # Simulating processing of feedstock

    def generateBiogas(self):
        print("Generating biogas...")
        # Simulating biogas generation
        return {
            'methaneContent': 60, # example values
            'carbonDioxideContent': 35
        }

    def adjustTemperature(self):
        self.temperatureControlSystem.regulateTemperature()

    def adjustPressure(self):
        self.pressureControlSystem.regulatePressure()

class BiogasPlant:
    def __init__(self, digester, gasHolder, gasUtilizationSystem):
        self.digester = digester
        self.gasHolder = gasHolder

```

```

        self.gasUtilizationSystem = gasUtilizationSystem

    def start(self):
        print("Starting the biogas plant...")
        self.digester.processFeedstock()
        biogas = self.digester.generateBiogas()
        gasVolume = self.digester.feedstock.getQuantity() * 0.02 # Example: 0.02 m³ of
biogas per kg feedstock
        self.gasHolder.storeGas(gasVolume)
        print("Biogas plant started.")

    def stop(self):
        print("Stopping the biogas plant...")
        print("Biogas plant stopped.")

    def update(self):
        print("Updating the biogas plant...")
        self.digester.adjustTemperature()
        self.digester.adjustPressure()
        print("Biogas plant updated.")

# Example usage:
if __name__ == "__main__":
    feedstock = Feedstock(type="Organic Waste", quantity=1000) # Example: 1000 kg of
organic waste
    tempControl = TemperatureControlSystem(initialTemperature=30, targetTemperature=37)
    pressureControl = PressureControlSystem(initialPressure=1, targetPressure=1.5)
    digester = Digester(feedstock, tempControl, pressureControl)
    gasHolder = GasHolder(volume=500) # Example: 500 m³ capacity of the gas holder
    gasUtilizationSystem = GasUtilizationSystem()

    biogasPlant = BiogasPlant(digester, gasHolder, gasUtilizationSystem)
    biogasPlant.start()
    biogasPlant.update()
    biogasPlant.stop()

```

**ДОДАТОК Г**  
**СПИСОК ПУБЛІКАЦІЙ ЗА ТЕМОЮ ДИСЕРТАЦІЇ**

1. Dyvak M., Pukas A., Manzhula V., Kasatkina N., Komar M. and Zabchuk V., "The Task of Parametric Identification the Interval Models with Nonlinear Parameters," 2022 12th International Conference on Advanced Computer Information Technologies (ACIT), Spisska Kapitula, Slovakia, 26-28 Sept. 2022, pp. 106-111.
2. Manzhula, Volodymyr, Mykola Dyvak, and Vadym Zabchuk. 2024. "The Improved Method for Identifying Parameters of Interval Nonlinear Models of Static Systems". International Journal of Computing 23 (1):19-25. <https://doi.org/10.47839/ijc.23.1.3431>.
3. Pukas, A., Simak, A., Goncharuk-Cholach, T., Konoplitska, O., Zabchuk, V. Intelligent Analyzing Module in the Academic Staff Performance Appraisal System. CEUR-WS. 2023. Vol. 3624. P. 423-429.
4. Pukas, A., Smal, V., Zabchuk, V. Software based on blockchain technology for consolidation the medical data about the patients examination. CEUR-WS. 2018. Vol. 2300. P. 170-174.
5. Дивак М., Забчук В. Моделювання характеристик процесів у біогазових установках на основі аналізу інтервальних даних / М. Дивак, В. Забчук/ Вісник Хмельницького національного університету. Серія: Технічні науки. Том 331 № 1 (2024), С.180-190. <https://doi.org/10.31891/2307-5732-2024-331-28>
6. Дивак Т., Забчук В. Огляд і аналіз методів та засобів моделювання процесів у біогазових установках. Наукові праці / Т. Дивак, В. Забчук/ Донецького національного технічного університету. Серія: "Інформатика, кібернетика та обчислювальна техніка", №1 (38)2024, С.57-70. [https:// doi: 10.31474/1996-1588](https://doi.org/10.31474/1996-1588)
7. Забчук, В. В. Аналіз методів побудови системи доповненої реальності для тестування практичних навиків [Текст] / В. В. Забчук // Комп'ютерні інформаційні технології : матеріали школи-семінару молодих вчених і студентів СІТ'2019 [м. Тернопіль, 29 листопада 2019 р.] / відп. за вип. М. П. Дивак. - Тернопіль : ТНЕУ, 2019. - С. 9.

8. Красносельська А.О., Веселов О.В., Забчук В.В. Інтелектуалізована система для оптимізації процесу формування документації програмного забезпечення. Комп'ютерні інформаційні технології: матеріали школи-семінару молодих вчених і студентів СІТ'2023. (Тернопіль, 30 листоп. 2023 р.). Тернопіль: ЗУНУ, 2023. С. 100-101.

9. Манжула В. І., Забчук В.В.. Архітектура програмного забезпечення для моделювання процесів у біогазових установках. Наукові праці Вінницького національного технічного університету, вип. 1 2024. <https://doi.org/10.31649/2307-5376-2024-1-19-28>.



**ДОДАТОК Д**  
**АКТИ ВПРОВАДЖЕНЬ РЕЗУЛЬТАТІВ ДИСЕРТАЦІЙНОГО ДОСЛІДЖЕННЯ**

ЗАТВЕРДЖУЮ

Виконавчий директор

ТОВ «Теофіпольська енергетична компанія»

 **Володимир МАРТИНЮК**

20 травня 2024 р.



АКТ

про впровадження результатів дисертаційної роботи  
 Забчука Вадима Володимировича

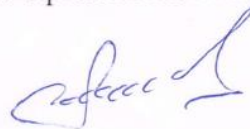
**“Математичне та програмне забезпечення для моделювання процесів у біогазових установках на основі аналізу інтервальних даних”**

Результати дисертаційної роботи аспіранта кафедри комп'ютерних наук Західноукраїнського національного університету Забчука Вадима Володимировича використано для підвищення ефективності функціонування біогазових установок, зокрема:

- методи математичного моделювання на основі інтервального підходу для відображення взаємозв'язку між основною характеристикою процесу мікробіологічного бродіння – кислотністю та чинниками, які на неї впливають, а також динаміки основної характеристики процесу, яка визначається кислотністю багатокомпонентного субстрату у біореакторі;
- програмне середовище для моделювання процесів у біогазових установках, яке побудовано на відкритій сервісно-орієнтованій архітектурі, що забезпечило його інтеграцію в існуючу систему управління біогазовими установками та дозволило в реальному часі управляти біохімічними процесами і тим самим підвищило ефективність їх функціонування.

Інженер-механік груповий

Технік з експлуатації та ремонту  
 устаткування



Резніченко Д.О.



Архіпович О.О.

ЗАТВЕРДЖУЮ



Керівник ТзОВ «ЕЛЕКС»  
Коновалов П.О.  
15 травня 2024 року

## АКТ

про впровадження результатів дисертаційної роботи  
Забчука Вадима Володимировича  
“Математичне та програмне забезпечення для моделювання процесів у  
біогазових установках на основі аналізу інтервальних даних”

Даний акт складений про те, що при реалізації багатьох програмних проєктів у Тернопільській філії компанії “ЕЛЕКС” використано результати дисертаційної роботи аспіранта кафедри комп’ютерних наук Західноукраїнського національного університету Забчука Вадима Володимировича.

Автором запропоновано:

- підхід до реалізації програмно-технічних комплексів, які включають інтеграцію програмно-реалізованих засобів математичного моделювання;
- відкриту сервісно-орієнтовану архітектуру програмного забезпечення для моделювання складних процесів, яка, на відміну від існуючих, інтегрує програмні компоненти: інтерпретації інтервальних дискретних моделей; ідентифікації цих моделей; розв’язування диференціальних рівнянь, що у сукупності уможливує розробку програмних систем та комп’ютерних середовищ для управління складними процесами.

Керівник тернопільської  
філії ТзОВ “ЕЛЕКС”

A handwritten signature in blue ink, appearing to read 'SK' followed by a stylized flourish.

Сергій КУТУЗОВ



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЗАХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
**НАУКОВО-ДОСЛІДНА ЧАСТИНА**

46009, Україна, м. Тернопіль, вул. Львівська, буд. 5А, тел. (0352) 51-75-52

№ 21/ 59 - 2024

«14» 05 2024 року

### ДОВІДКА

Видана **ЗАБЧУКУ** **Вадиму Володимировичу** про участь у науково-дослідних роботах, які виконує Західноукраїнський національний університет. Зокрема, у 2024 році на посадах молодшого наукового співробітника він бере участь у виконанні: фундаментального дослідження, що фінансується з державного бюджету «Математичне та комп'ютерне моделювання об'єктів з розподіленими параметрами на основі поєднання онтологічного та інтервального аналізу» (державний реєстраційний номер 0122U001497), держбюджетної наукової роботи молодих учених «Математичне та програмне забезпечення прототипу біогазової установки з підвищеною ефективністю функціонування» (державний реєстраційний номер 0124U000076) в лабораторії з проблем інформаційних технологій, а також госпдоговірної науково-дослідної роботи «Моделювання динаміки процесів у біогазових установках» (державний реєстраційний номер 0123U103785).

Накази ЗУНУ від 29.01.2024 р. №43; від 21.03.2024 р. № 116.

Заступник начальника  
науково-дослідної частини



Юрій МОСКАЛЬ



## МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

### ЗАХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

вул. Львівська, 11, м. Тернопіль, 46009; тел./факс +380 (352) 51-75-75;  
www.wunu.edu.ua; rektor@wunu.edu.ua; ідентифікаційний код за ЄДРПОУ 33680120



ЗАТВЕРДЖУЮ

Проректор з науково-педагогічної  
роботи Західноукраїнського  
національного університету  
к.е.н., доцент Віктор ОСТРОВЕРХОВ

23 квітня 2024 р.

АКТ

про впровадження в освітній процес Західноукраїнського національного  
університету результатів дисертаційної роботи

**Забчука Вадима Володимировича**

«Математичне та програмне забезпечення для моделювання процесів у біогазових  
установках на основі аналізу інтервальних даних»

Даний акт складений про те, що результати дисертаційної роботи Забчука Вадима Володимировича на тему: «Математичне та програмне забезпечення для моделювання процесів у біогазових установках на основі аналізу інтервальних даних» використані в освітньому процесі факультету комп'ютерних інформаційних технологій Західноукраїнського національного університету для студентів спеціальності «Інженерія програмного забезпечення».

При викладанні дисциплін «Моделювання систем», «Методи та програмні засоби теоретико-множинного інтервального підходу», «Інтервальний аналіз даних» розглядається двоетапний метод ідентифікації математичної моделі динаміки показника кислотності субстрату у біореакторі на основі аналізу інтервальних даних, який ґрунтується на розв'язуванні спочатку простішої задачі ідентифікації рівняння взаємозв'язку між основною характеристикою процесу та чинниками, які на неї впливають, і на цій основі, на другому етапі - ідентифікації математичної моделі динаміки цього показника у біореакторі. При викладанні дисципліни «Архітектура та проектування програмного забезпечення» розглядається комп'ютерне середовище для моделювання процесів у біогазових установках, які ґрунтуються на програмній інтерпретації інтервальних дискретних моделей та відкритій сервісно-орієнтованій архітектурі, що уможливило їх інтеграцію в існуючі системи управління біогазовими установками з метою забезпечення допустимих значень кислотності багатокомпонентного субстрату.

В.о. декана факультету комп'ютерних  
інформаційних технологій,  
к.т.н., доцент

Ігор ЯКИМЕНКО

Доцент кафедри  
комп'ютерних наук, к.т.н., доцент

Михайло ШПІНТАЛЬ