



УДК 004.5

# ПРО КОНСТРУЮВАННЯ ЗНАННЯ-ОРІЄНТОВАНОГО ІНТЕРФЕЙСУ ПРОГРАМНОЇ СИСТЕМИ

## ON CONSTRUCTION OF KNOWLEDGE-ORIENTED INTERFACE OF A SOFTWARE SYSTEM

Степашко В.С.<sup>1</sup>, Савченко-Синякова Є.А.<sup>2</sup>, Головін Б.О.<sup>3</sup>  
Stepashko V.S.<sup>1</sup>, Savchenko-Syniakova Ye.A.<sup>2</sup>, Holovin B.O.<sup>3</sup>

<sup>123</sup>Інститут інформаційних технологій та систем НАН України, Пр. Акад. Глушкова, 40, м. Київ, Україна  
Institute of Information Technologies and Systems of the National Academy of Sciences of Ukraine, Kyiv, Ukraine  
ORCID: <sup>1</sup> <https://orcid.org/0000-0001-7882-3208>, <sup>2</sup> <https://orcid.org/0000-0003-4851-9664>,  
<sup>3</sup> <https://orcid.org/0009-0003-5895-6240>  
E-mail: <sup>1</sup> [stepashko@irtc.org.ua](mailto:stepashko@irtc.org.ua), <sup>2</sup> [savchenko\\_e@meta.ua](mailto:savchenko_e@meta.ua), <sup>3</sup> [b.holovin@gmail.com](mailto:b.holovin@gmail.com)

Copyright © 2026 by author and the journal “Automation of technological and business – processes”.  
This work is licensed under the Creative Commons Attribution International License (CC BY).  
<http://creativecommons.org/licenses/by/4.0>



DOI: 10.15673/atbp.v18i1.3435

**Анотація.** Виконано аналіз проблеми створення знання-орієнтованих систем, здатних формувати інтерактивні алгоритми розв'язання нечітко сформульованих задач, що характерне для інтелектуальних систем. Показано, що для цього необхідне розроблення моделей структуризації знань про заданий клас задач, методи їх розв'язання та досвід застосування цих методів. такої методології доцільною є гібридна архітектура, що поєднує представлення знань про процес поетапного розв'язання задач, -адаптивний інтерфейс з трьома режимами взаємодії з користувачем, компонент візуалізації для аналізу результатів на кожному етапі та систему збереження обраних шляхів розв'язання для накопичення досвіду. Така архітектура забезпечує баланс між автоматизацією рутинних операцій та можливістю експертного контролю на кожному етапі прийняття рішень. Застосування запропонованого підходу демонструється на задачі індуктивного моделювання складних об'єктів. Показано, що процес прийняття рішень у такій задачі є послідовністю декількох інформаційно залежних задач, розв'язання яких потребує засобів інтерфейсу користувача, що дозволяють ефективно і коректно застосовувати наявні методи. -Завдяки покроковому розв'язанню проблем мінімізуються вимоги до рівня підготовки користувача.

**Abstract.** This paper analyses the problem associated with creating knowledge-oriented systems capable of forming interactive algorithms for solving ill-defined problems, which is typical for intelligent systems. It is shown that this requires the development of knowledge structuring models of a given class of problems, methods for solving them, and experience in applying these methods. To implement such a methodology, a hybrid architecture is appropriate, combining the presentation of knowledge about the process of step-by-step problem solving, an adaptive interface with three modes of user interaction, a visualization component for analysing results at each stage and a system for saving selected solution paths to accumulate experience. This architecture provides a balance between the automation of routine operations and the possibility of expert control at each stage of decision-making. The application of the proposed approach is demonstrated on the problem of inductive modeling of complex objects. It is shown that the decision-making process in such a problem is a sequence of several information-dependent tasks, the solution of which requires user interface tools that allow for the effective and correct implementation of existing methods. Due to step-by-step problem solving, requirements for user training level are minimized.

**Ключові слова:** структуризація знань, процес прийняття рішень, інтерфейс користувача, рівень підготовленості, індуктивне моделювання.

**Keywords:** knowledge structuring, decision-making process, user interface, level of preparedness, inductive modeling.

### Вступ

Основною характеристикою інтелектуальних систем є те, що це передусім системи, засновані на знаннях, або знання-орієнтовані системи, які здатні розв'язувати нечітко сформульовані задач. Для створення таких систем необхідно накопичувати знання про цільовий клас задач, методи їх розв'язання та досвід застосування цих



методів. Це дає змогу напрацювати систематизовані конструктивні рекомендації щодо розроблення функціонального і діалогового забезпечення інтелектуальних систем, зокрема систем моделювання об'єктів зовнішнього світу.

Показано, що процес прийняття рішень у складній задачі є послідовністю декількох інформаційно залежних задач, розв'язання яких потребує розроблення знання-орієнтованого інтерфейсу для взаємодії з користувачем. Поетапним розв'язанням задач досягається мінімізація вимог до підготовленості користувача. Запропоновано принцип організації тривіневого інтелектуального інтерфейсу в системах розв'язання складних задач.

Запропонований підхід демонструється на задачі індуктивного моделювання складних об'єктів. Розроблено методологію конструювання інтерактивної системи підтримки рішень у процесі індуктивної побудови моделей, яка базується на сукупності знань про методи моделювання та досвід їх застосування. Це дозволяє користувачам різного рівня підготовки за підтримки такої системи поетапно будувати ефективні алгоритми моделювання із заданих підмножин їх конструктивних елементів.

### Мета і задачі дослідження

Метою даного дослідження є розроблення методики створення інтелектуальних інтерактивних засобів підтримки прийняття рішень у процесі розв'язання складних нечітко сформульованих задач. Застосування запропонованого підходу буде продемонстровано на задачах індуктивного моделювання складних об'єктів.

Завданнями даного дослідження є

- проаналізувати задачу побудови людино-машинного інтерфейсу та процесу його інтелектуалізації;
- проаналізувати сучасні підходи до організації людино-машинного інтерфейсу;
- визначити загальні вимоги до організації інтелектуального інтерфейсу;
- запропонувати підхід до структуризації знань про процес індуктивної побудови моделі;
- розглянути інтелектуальний інтерфейс як реалізацію знань про процес побудови моделі;
- навести особливості сучасних засобів програмної реалізації інтерфейсу користувача.

### Задача побудови людино-машинного інтерфейсу та його інтелектуалізація

Під *інтерфейсом* розуміють певні набори правил для обміну інформацією між людиною та будь-яким технічним пристроєм, наприклад, комп'ютером, які є обов'язковими для виконання обома сторонами [1]. Тому від побудови інтерфейсу буде залежати сама технологія спілкування людини з комп'ютером. Оскільки інтерфейс – це набір правил, а будь-які правила можна узагальнити, то перейшли до поняття «вид інтерфейсу», який є об'єднанням за схожістю способів взаємодії людини й комп'ютера [2].

Розрізняють такі види інтерфейсів:

*Командний інтерфейс*. Людина подає команди, комп'ютер їх виконує і видає результат.

*WIMP – інтерфейс* (Window – вікно, Image – образ, Menu – меню, Pointer – покажчик). Взаємодія з користувачем ведеться за допомогою графічних образів.

*SILK – інтерфейс* (Speech – мова, Image – образ, Language – мова, Knowledge – знання). Найбільш наближений до людської форми спілкування. Відбувається звичайна «розмова» людини і комп'ютера. Комп'ютер аналізує людську мову, обирає ключові фрази і виконує відповідні команди.

Інтелектуалізація такої взаємодії між людиною та комп'ютером передбачає додання до людино-машинних інтерфейсів елементів штучного інтелекту, що дозволяє імітувати процеси діяльності людського інтелекту, а це в свою чергу імітує людське мислення та здібності до навчання. Основною вимогою до технологій, побудованих за принципами інтелектуальної машино-людської взаємодії, є їх ефективність та функційність.

Якщо при побудові користувацьких інтерфейсів будуть використані методи глибинного навчання та обробки природної мови, це дозволить таким системам краще розуміти потреби та наміри користувачів, що зробить взаємодію більш природною.

Розроблення інтелектуальних інтерфейсів висуває певні вимоги до користувачів.

1. Для оператора вони мають бути стандартизовані й забезпечувати високу ефективність передачі даних. Дані мають бути масштабованими, щоб виконувати передачу великих обсягів даних та витримувати велике навантаження.
2. Для людини вони мають бути зручними для використання: адаптованими, вміти підлаштовуватись під потреби користувача і зовнішньо зрозумілими, щоб забезпечити зручність взаємодії.

Розглянемо детальніше сучасні підходи до організації та побудови людино-машинних інтерфейсів.

### Сучасні підходи до організації людино-машинного інтерфейсу

Серед сучасних підходів фундаментальною є концепція змішано-ініціативних інтерфейсів, де людина і комп'ютер виступають партнерами при розв'язанні задач. Основні принципи таких систем включають аналіз невизначеності щодо намірів користувача, використання діалогу для усунення цієї невизначеності, можливість коригування автоматичних рішень та мінімізацію критеріїв для прийняття рішень [3]. Сучасні дослідження розширюють підхід байссівською адаптацією, коли система моделює поведінку користувача і динамічно коригує



розподіл ініціативи на основі спостережуваних дій, що дозволяє покращити ефективність спільної роботи навіть за умови неповних знань обох сторін [4].

Адаптивні інтерфейси базуються на моделюванні характеристик користувача, контексту та предметної області. Розрізняють адаптацію, ініційовану системою, та адаптивність, ініційовану користувачем [5]. Новітні розробки демонструють адаптацію на основі когнітивного навантаження в реальному часі, коли система оцінює поточний когнітивний стан користувача і автоматично коригує деталізацію інформації та швидкість її передачі, що призводить до значного покращення продуктивності виконання задач [6]. У контексті моделювання складних об'єктів адаптивність може проявлятися через зміну рівня деталізації представлення проміжних результатів залежно від досвіду користувача з аналогічними задачами.

Дослідження взаємодії людини з системами штучного інтелекту дозволили сформулювати принципи, які включають чітке повідомлення можливостей системи, легке редагування та скасування некоректних дій, налаштування впевненості користувача у прийнятих рішеннях та їх пояснення [7]. Для систем моделювання це означає необхідність візуалізації результатів кожного етапу та пояснення отриманих рішень.

Сучасний підхід зосереджується на налаштуванні впевненості самого користувача, а не лише довіри до системи, через механізми типу "подумай навпаки", що веде до більш раціонального використання рекомендацій від засобів штучного інтелекту [9]. У системах моделювання це критично важливе, оскільки надмірна довіра може призвести до використання неадекватної моделі.

Аналіз показує, що більшість концепцій побудови інтерфейсу зосереджені на загальних принципах взаємодії, недостатньо враховуючи специфіку постановки і розв'язання конкретної прикладної задачі. Зокрема, класичні адаптивні інтерфейси прилаштовуються до поведінкових патернів користувача, але не до характеристик вхідних даних чи мети розв'язуваної задачі. Наприклад, для систем індуктивного моделювання необхідна адаптація, коли система враховує одночасно тип даних, мету моделювання, рівень кваліфікації користувача та обсяг апріорної інформації про об'єкт дослідження.

Таким чином, сучасні підходи до організації інтерфейсу базуються на принципах адаптивності, змішаної ініціативи та багаторівневої підтримки користувачів, що дозволяє створювати архітектуру інтелектуального діалогу для ефективного розв'язання складних задач.

### **Загальні вимоги до організації інтелектуального інтерфейсу**

Для побудови правил вибору рішень, на основі яких розробляється структура діалогового інтерфейсу на кожному етапі розв'язання задач, потрібен аналіз досвіду застосування методів їх розв'язання. Від рівня обґрунтованості таких правил та їх різноманіття щодо умов моделювання залежить інтелектуальний рівень діалогу і відповідно системи загалом. Крім того, структура діалогу в сучасній системі повинна враховувати також можливий рівень кваліфікації користувача. Для забезпечення підстроювання системи до користувача пропонується спосіб організації багаторівневого діалогу. При цьому прийнято, що передбачувану підготовленість можна віднести до одного з трьох основних рівнів:

- 1) *низький* – початкуючий користувач (проте знайомий зі своїм об'єктом);
- 2) *середній* – підготовлений користувач (знає об'єкт і принципи розв'язання задач);
- 3) *високий* – кваліфікований користувач (експерт, що знає і об'єкт, і методи розв'язання).

Вказаним рівням кваліфікації ставиться у відповідність три основних режими діалогу: 1) автоматичний, 2) інтерактивний, 3) меню-подібний, сутність яких така. Для кожного етапу вибору рішення складається повне меню можливих рішень (включно з відмовою від вибору). Далі для автоматичного режиму фіксується (за промовчанням) одне рішення, що на практиці приймається найчастіше; для інтерактивного режиму кілька рішень, які вибираються найчастіше (з досвіду), супроводжуються підказками і коментарями щодо умов їх переважного вибору; в третьому режимі пропонується вибирати рішення з повного меню без коментарів. Загальна схема такого поетапного діалогу подана на рисунку 2.

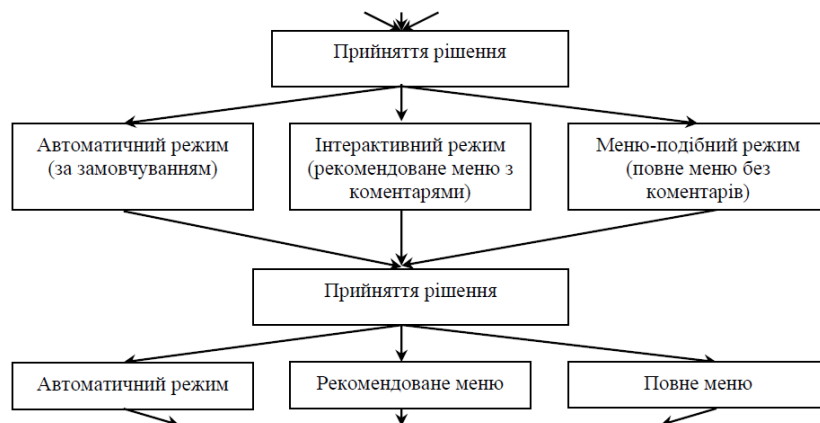
Оскільки такий принцип діалогу застосовується на кожному етапі розв'язання задач, при роботі системи можливий як один із трьох основних режимів, так і довільне їхнє сполучення, яке змінюється від одного етапу до іншого. Це дає ефект гнучкої адаптації системи до рівня кваліфікації користувача, тобто до його поінформованості про процес розв'язання, до знання ним об'єкта та до обсягу наявної в нього апріорної інформації на різних етапах прийняття рішень. Відповідні правила прийняття рішень мають розроблятися з метою максимально автоматизувати більшість рутинних операцій, щоб сприяти мінімізації непродуктивних витрат та інтенсифікації творчих можливостей дослідника (користувача).

Діалогова оболонка багато в чому визначає конструктивні особливості реалізації систем прийняття рішень, причому тут важливе поетапне розв'язання задач. При цьому досягається мінімізація вимог до підготовленості користувача щодо розв'язання задачі та поповнення цього мінімуму за сприяння програмної системи.

Зі сказаного видно, що структура поділу на етапи сприяє інтелектуалізації діалогу. Його застосування в «розумній» системі прийняття рішень є умовою того, щоб користувач, незалежно від рівня його підготовки, апріорних знань про об'єкт і початкові дані, на виході системи міг одержати і проаналізувати розв'язання своєї



конкретної задачі. Звичайно, чим більш підготовлений користувач і чим більше в нього інформації про об'єкт, тим більш адекватним буде результат.



**Рисунок 2 Структура тривірневого інтерфейсу**  
**Figure 2 Three-tier interface structure**

При цьому варто наголосити, що навіть користувач, що вперше зустрівся з задачею моделювання, за допомогою такої системи завжди може отримати хоча й просту, але все-таки задовільну модель за рахунок повністю автоматичного режиму роботи системи.

Таким чином, запропонований принцип організації інтелектуального інтерфейсу в системах розв'язання складних задач повинен мати такі основні властивості:

- реалізація знань людини-експерта, отриманих з аналізу досвіду застосування методів;
- інтерактивна підтримка всіх етапів;
- мінімізація вимог до рівня підготовки користувача;
- активне використання знань користувача;
- обов'язкове розв'язання задачі від початку до кінця;
- прийняття рішення системою в разі відмови користувача (процедура «не знаю»);
- здатність до навчання користувача в процесі роботи з системою.

Отже, кожна складна задача може бути розглянута як послідовність етапів вибору рішень. На кожному етапі існує деяка обмежена множина можливих рішень, що залежать від прийнятих раніше. Внаслідок цього кожне прийняте рішення звужує підмножини можливих рішень на всіх наступних етапах. Це дозволяє організувати діалог, інтелектуальність якого визначається реалізацією знань (теоретичних і набутих з досвіду) про процес і методи розв'язання задач та мінімізацією вимог до рівня підготовленості користувача.

### **Задача структуризації знань про процес індуктивної побудови моделі**

Для розроблення сучасних інформаційних технологій моделювання, прогнозування і пошуку закономірностей необхідно передусім виконати структуризацію знань в цій галузі. Ця задача допомагає напрацювати систематизовані рекомендації щодо розроблення функціонального і діалогового забезпечення інтелектуальних систем моделювання.

Відповідно до загальної методики [1], для структуризації знань в цій предметній галузі доцільно проаналізувати такі аспекти проблеми:

- 1) визначення основних етапів процесу розв'язання задачі моделювання;
- 2) виділення основних методів розв'язання такої задачі і порівняння їх функціональних особливостей;
- 3) порівняльний аналіз ефективності цих методів;
- 4) аналіз і узагальнення досвіду їх застосування з формуванням необхідних рекомендацій.

Розглянемо дещо докладніше вказані аспекти проблеми структуризації.

1. Основні етапи розв'язання задачі індуктивного моделювання визначимо за допомогою найпростішого її формулювання. Нехай є деяка множина  $F$  моделей, і якість кожної моделі  $f \in F$  характеризується значенням заданого критерію  $CR(f)$ . Тоді кращою (оптимальною) в сенсі цього критерію є модель, що відповідає мінімуму  $CR(f)$ .

В загальному випадку процес розв'язання такої задачі структурно-параметричної ідентифікації моделі за наявними даними включає передусім такі основні етапи:

- 1) Задавання вибірки даних (отриманої в результаті пасивного або активного експерименту), а також апріорної інформації;



- 2) Вибір або задавання класу базисних функцій та відповідне перетворення даних;
- 3) Генерація різних структур моделей в обраному класі;
- 4) Оцінювання параметрів структур, що генеруються, і формування множини  $F$ ;
- 5) Мінімізація заданого критерію  $CR(f)$  і вибір оптимальної моделі  $f^*$ .
- 6) Перевірка адекватності отриманої оптимальної моделі.
- 7) Прийняття рішення про завершення процесу.

Перелічені етапи описують довільний процес побудови моделей, причому залежно від апріорної інформації та мети моделювання ті чи інші етапи можуть бути відсутні. Наприклад, в окремому випадку задачі параметричної ідентифікації множина  $F$  складається з однієї моделі (задається єдина структура), тобто виключається етап 5 процесу. Очевидно, що сформульовані етапи повинні бути покладені в основу функціональної архітектури програмної системи структурної ідентифікації.

2. Нехай яким-небудь чином (за допомогою аналізу літератури або опитування експертів) відібрана деяка множина методів структурної ідентифікації. Кожний з них розв'язує вказану задачу, і відповідно може бути охарактеризований:

- 1) класом моделей (базисних функцій), для якого він призначений;
- 2) використовуваним в ньому генератором структур;
- 3) методом оцінювання параметрів;
- 4) критерієм якості моделей.

Вибір або задавання класу базисних функцій та відповідне перетворення даних залежать не від методу моделювання, а від апріорної інформації та мети моделювання. Способи аналізу адекватності моделей, що зводяться до перевірки правильності прийнятих апріорних припущень, носять достатньо загальний характер і в основні характеристики методу можуть не включатися.

Таким чином, кожен метод моделювання можна описати за допомогою чотирьох вказаних компонентів, з урахуванням яких легко можна порівняти функціональні особливості заданої множини методів.

З іншого боку, для відібраної сукупності методів, розчленувавши кожен з них на відповідні складові, можна утворити множини класів моделей, генераторів структур, методів оцінювання параметрів і критеріїв вибору моделей і аналізувати ці множини як самостійні наукові об'єкти. Зазначимо, що ці чотири множини мають той же сенс, що і множина можливих варіантів вибору на етапах 2-5 процесу моделювання, тобто вказане розчленування методів дає ключ до формування функціонального наповнення системи.

3. Порівняння ефективності різних методів моделювання зводиться до порівняння застосовуваних в них критеріїв вибору моделей. Це впливає з того, що для конкретного модельованого об'єкту при заданому обсязі апріорної інформації можна, як правило, вибрати найдоцільніший варіант класу моделей, генератора структур і методу оцінювання параметрів, і тоді ефективність досягнення бажаної мети моделювання визначається критерієм якості моделей.

4. Аналіз досвіду застосування алгоритмів необхідний для побудови правил вибору рішень на кожному етапі моделювання, на основі яких розробляється структура діалогу. Від рівня обґрунтованості таких правил та їх різноманіття по відношенню до умов моделювання залежить інтелектуальний рівень діалогу і відповідно системи загалом. Крім того, структура діалогу в сучасній системі повинна враховувати також можливий рівень кваліфікації користувача. Для забезпечення підстроювання системи до користувача можна застосувати описаний вище спосіб організації багаторівневого діалогу.

Отже, внаслідок розгляду викладеної методології структуризації знань виявлено, що ефективна в прикладному сенсі інтерактивна програмна система індуктивного моделювання повинна характеризуватися: достатньою загальністю в сенсі використання основних методів моделювання; адаптованістю до різних умов апріорної невизначеності; достатньою доступністю користувачам різної кваліфікації; обчислювальною ефективністю (швидкодією).

Зазначимо, що з погляду задач керування найбільш доцільно прагнути отримати внаслідок ідентифікації модель з найкращим прогностичними властивостями. У зв'язку з цим зі всіх підходів, наявних у структурній ідентифікації моделей, найбільший інтерес представляє підхід "перехресного підтвердження" (cross-validation). По своїй суті він безпосередньо спрямований на вибір моделей, що зменшують помилку прогнозування: в найпростішому випадку вибірка ділиться на дві частини, на одній з яких генерується множина моделей, а на іншій обчислюється помилка кожної моделі, і вибирається та, яка краще працює "в режимі прогнозування". Серед методів цього напрямку виділяється МГУА як метод автоматичного пошуку кращої моделі, що має більше різноманіття можливостей порівняно з іншими.

### **Інтелектуальний інтерфейс як реалізація знань про процес побудови моделі**

Успіх застосування індуктивних методів моделювання за даними вибірки спостережень визначається не тільки їх науковою обґрунтованістю, але і ефективною їх реалізацією у вигляді інтерактивних програмних засобів, інтелектуальна діалогова оболонка яких допомагає користувачу будь-якої кваліфікації успішно пройти всі етапи процесу моделювання – від аналізу даних до отримання моделі. Такі засоби повинні максимально



автоматизувати більшість рутинних операцій, щоб сприяти мінімізації непродуктивних витрат та інтенсифікації творчих можливостей дослідника (користувача).

Діалогова оболонка багато в чому визначає конструктивні особливості реалізації систем прийняття рішень, причому важливо забезпечити «покрокове» розв'язання задач. Під покроковістю розуміється мінімізація вимог до підготовленості користувача при ініціалізації розв'язання задачі та поповнення цього мінімуму за сприяння комп'ютерної системи. На кожному кроці діалогу аналізується коректність даних, що вводяться користувачем для продовження розв'язання задачі, та відбувається їх коригування у разі некоректності.

*Про поетапне розв'язання задач моделювання.* Досвідчений користувач може знати і вміти ефективно застосовувати певні алгоритми та критерії моделювання для задач зі своєї предметної галузі. У загальному ж випадку задача істотно складніша: у дослідника є певний набір апріорної інформації про досліджуваний процес (об'єкт), представлений у вигляді знань і припущень, є вибірка початкових даних спостережень за цим процесом (об'єктом) і певний набір невизначеностей. Йому необхідно побудувати модель такого процесу чи об'єкта, причому наперед не відомо, які алгоритми і критерії покажуть себе найбільш адекватними. Саме такий випадок розглядається як предмет для побудови інтелектуальної програмної системи на основі бази знань.

Вище наведено розбиття процесу моделювання на основні етапи, яким відповідають етапи послідовного прийняття рішень. На кожному з цих етапів приймається одне з низки можливих рішень (сформованих експертами, за літературою або на основі досвіду), після чого можна переходити на наступний етап. При цьому варто наголосити, що навіть користувач, що вперше зустрівся з задачею моделювання, за допомогою такої системи завжди може отримати можливо просту, але задовільну модель, при цьому реалізується повністю автоматичний режим роботи системи.

Як зазначалося в [10], при проектуванні інтелектуальної системи моделювання є потреба реалізувати такі ширші характеристики інтелектуального інтерфейсу користувача: інтерактивний режим користувач-система, який відстежує та підтримує всі етапи процесу моделювання; отримання та активне використання знань користувача під час виконання процесу; постійний моніторинг всіх рішень, прийнятих користувачем на будь-якому етапі; навчання користувача під час взаємодії з системою тощо.

### **Сучасні засоби програмної реалізації інтерфейсу користувача**

Фундаментальним принципом побудови інтерфейсу користувача є модульна архітектура з чітким розділенням відповідальності між компонентами [11]. Успішні системи використовують шарову структуру, яка містить модель даних, логіку обробки, представлення результатів та компонентів взаємодії. Такий підхід забезпечує можливість незалежної зміни окремих компонентів, спрощує розширення функціональності та полегшує інтеграцію нових алгоритмів. Еталонна архітектура систем наукових обчислень визначає рівні презентації, управління процесами, задач та операційний рівень [12]. Для систем моделювання це означає чітке відокремлення інтерфейсу від обчислювальних компонентів, що забезпечує гнучкість та можливість модернізації.

Сучасні платформи аналізу даних демонструють ефективність візуального програмування з модульною архітектурою на основі плагінів [13]. Користувачі будують аналітичні конвеєри з окремих блоків, кожен з яких інкапсулює певну функціональність та має стандартизований інтерфейс. Для систем індуктивного моделювання це може означати представлення кожного етапу процесу як окремого вузла, який включає один з компонентів: вибір базисних функцій, генерація структур, оцінювання параметрів, вибір моделі за критерієм. Проте такі системи не реалізують інтелектуальної підтримки вибору методів — користувач повинен самостійно знати, які блоки і в якій послідовності застосовувати, що обмежує їх застосування користувачами з недостатньою підготовкою.

Альтернативна парадигма представлена подвійним інтерфейсом: візуальний конструктор для швидкого прототипування поєднується зі скриптовим інтерфейсом для програмного доступу. Така гібридна архітектура забезпечує доступність для початківців через графічний інтерфейс, водночас надаючи експертам повний контроль через скрипти. Впливовою парадигмою стали обчислювальні ноутбуки, що поєднують у єдиному документі виконуваний код, результати обчислень та текстові пояснення [14]. Документо-орієнтований підхід забезпечує природний робочий процес послідовного виконання кроків аналізу з документуванням. Архітектура з розділенням інтерфейсу та обчислювального ядра дозволяє віддалене виконання та підтримку різних алгоритмів. Проте для реалізації тривірневого діалогу з адаптацією до користувача лінійної послідовності кроків недостатньо — необхідна деревоподібна структура з альтернативними шляхами на кожному етапі, де система рекомендує оптимальний шлях залежно від характеристик даних, а користувач може обирати інший варіант за наявності додаткової інформації.

Успішні системи реалізують інтерактивну візуалізацію, що дозволяє користувачам маніпулювати даними та їх представленням [15]. Основні операції включають специфікацію даних, вибір типу візуалізації, маніпулювання виглядом та керування процесом. Для систем моделювання важливою є архітектура з мережевим доступом, що включає серверну частину з базами даних та обчислювальними модулями, а також клієнтську частину з web-



інтерфейсом. Така архітектура забезпечує доступність через браузер без встановлення спеціалізованого програмного забезпечення, централізоване зберігання результатів та можливість спільної роботи.

Перспективним є онтологічний підхід до структурного проектування, коли формальні онтології використовуються для представлення знань предметної області, а структура інтерфейсу автоматично генерується на основі цих онтологій [16]. Це забезпечує узгодженість інтерфейсу з функціональністю системи, спрощує підтримку при додаванні нових методів та дозволяє реалізувати механізм інференції правил для автоматичного вибору рішень на кожному етапі. База знань містить не лише опис методів та їх параметрів, але й правила вибору методів на основі характеристик даних, мети моделювання та рівня користувача. Діалогова система запитує в онтології можливі варіанти рішень для поточного етапу, застосовує правила інференції для формування рекомендацій та динамічно будує відповідний інтерфейс взаємодії.

### **Обговорення отриманих результатів**

У статті виконано аналіз задачі створення знання-орієнтованих систем, здатних розв'язувати нечітко сформульовані задачі, що характерне для інтелектуальних систем. Показано, що для цього необхідне розроблення моделей структуризації знань про заданий клас задач, методи їх розв'язання та досвід застосування цих методів.

Процес прийняття рішень у складній задачі є послідовністю розв'язування декількох інформаційно залежних задач, що потребує засобів діалогу, які дозволяють ефективно і коректно застосовувати наявні методи, і, отже, тісно пов'язаний з проблемою розробки інтерфейсу з користувачем.

Аналіз досвіду застосування алгоритмів потрібен для побудови правил вибору рішень на кожному етапі розв'язування задачі, на основі яких розробляється структура діалогу. Від рівня обґрунтованості таких правил та їх різноманітності по відношенню до умов моделювання залежить інтелектуальний рівень діалогу і відповідно системи загалом. Крім того, структура діалогу в сучасній системі повинна враховувати також можливий рівень кваліфікації користувача з метою забезпечення належного підстроювання системи.

Для реалізації описаної методології доцільною є гібридна архітектура, що поєднує представлення знань про процес розв'язання, модульну систему алгоритмів з можливістю розширення, адаптивний інтерфейс з трьома режимами інтерактивної взаємодії, компонент візуалізації для аналізу результатів на кожному етапі та систему збереження обраних шляхів розв'язання для накопичення досвіду. Така архітектура забезпечує баланс між автоматизацією рутинних операцій та можливістю експертного контролю дій користувача на кожному етапі прийняття рішень.

Діалогова оболонка інтерфейсу багато в чому визначає конструктивні особливості реалізації систем прийняття рішень, причому ефективним є саме поетапне розв'язання задач. При цьому досягається мінімізація вимог до підготовленості користувача при розв'язанні задач та поповнення цього мінімуму за сприяння програмної системи. З цією метою запропоновано структуру поділу процесу розв'язання на етапи, що є одночасно структурою інтелектуального діалогу. Його застосування є умовою того, щоб користувач, незалежно від рівня його підготовки, апріорних знань про об'єкт і початкові дані, на виході системи міг одержати прийнятний результат розв'язання своєї конкретної задачі.

Застосування розробленого підходу демонструється на задачі індуктивного моделювання складних процесів і систем. Запропонована концепція побудови інтерактивної програмної системи моделювання базується на структуризації знань в області моделювання та їх реалізації з метою створення інтелектуальної діалогової системи. Вона допомагатиме досліднику в прийнятті обґрунтованих рішень про те, який алгоритм моделювання з множини можливих буде найбільш адекватним для конкретного процесу з урахуванням його характеру та цілей моделювання, а також у побудові відповідної моделі. Це дозволить максимально використовувати знання користувача і обсяг інформації, наявної у вибірці даних, контролюючи при цьому несуперечність прийнятих ним рішень і приймаючи рішення автоматично у разі, коли користувач відмовляється від вибору.

### **Висновки**

Виконаний аналіз проблеми розроблення людино-машинного інтерфейсу системи підтримки розв'язання складних задач дозволяє визначити, що доцільною для реалізації є гібридна архітектура, яка поєднує представлення знань про процес розв'язання, модульну систему алгоритмів з можливістю розширення, адаптивний інтерфейс з трьома режимами інтерактивної взаємодії, компонент візуалізації для аналізу результатів на кожному етапі та систему збереження обраних шляхів розв'язання для накопичення досвіду. Така архітектура забезпечить баланс між автоматизацією рутинних операцій та можливістю контролю експертами кожного етапу в процесі прийняття рішень. Показано, що процес прийняття рішень складної задачі є послідовністю декількох інформаційно залежних задач, розв'язання яких потребує засобів діалогу з користувачем. Завдяки покроковому розв'язанню задач мінімізуються вимоги до рівня підготовки користувача. Розроблений підхід демонструється на задачі підтримки рішень у процесі індуктивного моделювання складних об'єктів.



### Список літератури

1. Уткіна Г. А. Людино-машинний інтерфейс: навч. посібник. Київ: КЕІ ДВНЗ «КНЕУ імені Вадима Гетьмана», 2011. 162 с.
2. Доценко С. І. Людино-машинний інтерфейс: навч. посібник. Харків: УкрДУЗТ, 2022. 135 с.
3. Horvitz E. Principles of Mixed-Initiative User Interfaces. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '99). ACM, 1999. P. 159–166. <https://doi.org/10.1145/302979.303030>
4. Natarajan M., Xue C., van Waveren S., Feigh K.M., Gombolay M.C. Mixed-Initiative Human-Robot Teaming under Suboptimality with Online Bayesian Adaptation. Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024). IFAAMAS, 2024. P. 1454–1462. <https://doi.org/10.65109/TNQY2230>
5. Norcio A.F., Stanley J. Adaptive Human-Computer Interfaces: A Literature Survey and Perspective. IEEE Transactions on Systems, Man, and Cybernetics. 1989. Vol. 19, No. 2. P. 399–408. <https://doi.org/10.1109/21.31042>
6. Wen S., Ping S., Wang J., Zhu Y., Zhao T., Wang Y., Zhou M.X., Wang H. AdaptiveVoice: Cognitively Adaptive Voice Interface for Driving Assistance. Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems. ACM, 2024. Article 253. 18 p. DOI: 10.1145/3613904.3642876
7. Amershi S., Weld D., Vorvoreanu M., Fournery A., Nushi B., Collisson P., Suh J., Iqbal S., Bennett P.N., Inkpen K., Teevan J., Kikin-Gil R., Horvitz E. Guidelines for Human-AI Interaction. Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems. ACM, 2019. Article 3. 13 p. <https://doi.org/10.1145/3290605.3300233>
8. Ma S., Wang X., Lei Y., Wang X., Ming Y., Qu H. "Are You Really Sure?" Understanding the Effects of Human Self-Confidence Calibration in AI-Assisted Decision Making. Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems. ACM, 2024. Article 840. 20 p. <https://doi.org/10.1145/3613904.3642671>
9. Ma S., Chen Q., Wang X., Luo X., Liang H.-N., Qu H. Towards Human-AI Deliberation: Design and Evaluation of LLM-Empowered Deliberative AI for AI-Assisted Decision-Making. Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems. ACM, 2025. Article 261. 31 p. <https://doi.org/10.1145/3706598.3713423>
10. Stepashko V. On the Self-Organizing Induction-Based Intelligent Modeling / In: Advances in Intelligent Systems and Computing III / N. Shakhovska, M.O. Medykovsky, Editors. AISC book series, Volume 871. Cham: Springer, 2019. P. 433-448. [https://doi.org/10.1007/978-3-030-01069-0\\_31](https://doi.org/10.1007/978-3-030-01069-0_31)
11. Heer J., Agrawala M. Software Design Patterns for Information Visualization. IEEE Transactions on Visualization and Computer Graphics. 2006. Vol. 12, No. 5. P. 853–860. <https://doi.org/10.1109/TVCG.2006.178>
12. Lin C., Lu S., Fei X., et al. A Reference Architecture for Scientific Workflow Management Systems and the VIEW SOA Solution. IEEE Transactions on Services Computing. 2009. Vol. 2, No. 1. P. 79–92. <https://doi.org/10.1109/TSC.2009.4>
13. Berthold M.R., Cebron N., Dill F., et al. KNIME – The Konstanz Information Miner: Version 2.0 and Beyond. ACM SIGKDD Explorations Newsletter. 2009. Vol. 11, No. 1. P. 26–31. <https://doi.org/10.1145/1656274.1656280>
14. Kluyver T., Ragan-Kelley B., Pérez F., et al. Jupyter Notebooks — a Publishing Format for Reproducible Computational Workflows. Positioning and Power in Academic Publishing. 2016. P. 87–90. <https://doi.org/10.3233/978-1-61499-649-1-87>
15. Heer J., Shneiderman B. Interactive Dynamics for Visual Analysis. Communications of the ACM. 2012. Vol. 55, No. 4. P. 45–54. <https://doi.org/10.1145/2133806.2133821>
16. Denicola, A; Missikoff, M; Navigli, R (2009). A software engineering approach to ontology building (PDF). Information Systems. 34 (2): 258. <https://doi:10.1016/j.is.2008.07.002>

Отримана в редакції 25.02.2026. Прийнята до друку 09.03.2026. Розміщено в інтернеті 30 березня 2026.  
Received 25 February 2026. Approved 09 March 2026. Available in Internet 30 March 2026