

**МІНІСТЕРСТВО ВНУТРІШНІХ СПРАВ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ВНУТРІШНІХ СПРАВ
КРЕМЕНЧУЦЬКИЙ ЛЬОТНИЙ КОЛЕДЖ**

Циклова комісія технічного обслуговування авіаційної техніки

ТЕКСТ ЛЕКЦІЇ

навчальної дисципліни

«Інформаційні системи контролю та діагностики газотурбінних двигунів»
вибіркових компонент
освітньо-професійної програми першого (бакалаврського) рівня вищої освіти

272 Авіаційний транспорт

Технічне обслуговування та ремонт повітряних суден і авіадвигунів

за темою №4. – Керування складними системами

Кременчук 2023

ЗАТВЕРДЖЕНО

Науково-методичною радою
Харківського національного
університету внутрішніх справ
Протокол від 30.08.2023 № 7

СХВАЛЕНО

Методичною радою Кременчуцького
льотного коледжу Харківського
національного університету внутрішніх
справ
Протокол від 28.08.2023 № 1

ПОГОДЖЕНО

Секцією Науково-методичної ради
ХНУВС з технічних дисциплін
Протокол від 29.08.2023 № 7

Розглянуто на засіданні циклової комісії технічного обслуговування авіаційної
техніки, протокол від 28.08.2023 № 1

Розробник:

*1. Викладач циклової комісії технічного обслуговування авіаційної техніки,
викладач-спеціаліст Самохліб Олександр Олександрович*

Рецензенти:

- 1. Завідувач кафедри технологій аеропортів Національного авіаційного
університету, д.т.н., професор Тамаргазін О.А.*
- 2. Викладач циклової комісії аеронавігації КЛК ХНУВС, к.т.н., с.н.с. Тягній
В.Г.*

План лекції

1. Інтелектуальні системи: Алгоритмічний та декларативний підходи до керування; Формалізація понять алгоритмічності та декларативності.
2. Квазіалгоритми.
3. Характеристика інтелектуальних систем з точки зору кібернетики: Означення інтелектуальної системи; Типова схема функціонування інтелектуальної системи.

Рекомендована література:

Основна

1. Нечипоренко О. М. Основи надійності літальних апаратів : навчальний посібник. Київ : НТУУ «КПІ», 2010. 240 с.

Допоміжна

2. Нестеренко О. В., Савенков О. І., Фаловський О. О. Інтелектуальні системи підтримки прийняття рішень : навчальний посібник. Київ : Національна академія управління, 2016. 188 с.
3. Вахнюк С.В. Технологія створення програмних та інтелектуальних систем: навчальний посібник. Суми : УАБС НБУ, 2011. 254 с.
4. Шаров С. В., Лубко Д. В., Осадчий В. В. Інтелектуальні інформаційні системи : навчальний посібник. Мелітополь: Вид-во МДПУ ім. Б. Хмельницького, 2015. 144 с.

Інформаційні ресурси в інтернеті

1. <https://openarchive.nure.ua/server/api/core/bitstreams/ced9dde2-bace-4fe0-8e5f-8a18fd4da046/content>

ТЕКСТ ЛЕКЦІЇ

4.1. Керування складними системами

4.1.1. Алгоритмічний та декларативний підходи до керування

На найбільш загальному рівні, можна виділити два підходи до керування складними системами та до програмування роботів і комп'ютерів, що могли б розв'язувати ті чи інші задачі. При програмуванні сучасних комп'ютерів в основному реалізований традиційний **алгоритмічний підхід**, який можна ще назвати імперативним. Цей підхід вимагає заздалегідь продумати та детально розписати, як треба вирішувати певну проблему. Написання програми вимагає задання чітких послідовностей інструкцій. Якщо таку послідовність вдається написати — комп'ютер зможе вирішувати дану задачу, якою б складною вона не була. Але, ясна річ, він виконувати інструкції, абсолютно не розуміючи їх змісту. Якщо зміняться умови, за яких виконується програма, комп'ютер може виявитися безпомічним.

Інший підхід — **декларативний**. Інтелектуальному виконавцеві (людині чи комп'ютерові) досить сказати, **що** треба робити, тобто лише сформулювати завдання, побудувавши всі взаємозв'язки між об'єктами у предметній області. **Як** цезавдання буде виконуватися — повинен визначити сам виконавець.

Запустити космічний корабель так, щоб він приземлився на Марсі, взяв зразки ґрунту та привіз їх назад — задача дуже складна, але вона піддається точній алгоритмізації. Математичні методи дозволяють точно розрахувати траєкторію, по якій повинен рухатися такий корабель.

Послати робота до магазину за пляшкою молока — задача, на декілька порядків більш складна. Не кажучи вже про сам процес спілкування з продавцем, робот повинен вирішити ряд не зовсім формалізованих підзадач. Заздалегідь розрахувати траєкторію руху неможливо, оскільки робот повинен уникнути зіткнень з людьми та автомобілями. Якщо магазин закритий на переоблік, він повинен знайти інший магазин. *Неможливо передбачити всі ситуації, які можуть виникнути, а відтак — алгоритмізувати розв'язок задачі.* Тому виконання такого завдання під силу лише інтелектуальній системі, яка вміє *орієнтуватися в зовнішньому світі, аналізувати поточні ситуації та коригувати, адаптувати свою поведінку на основі такого аналізу.*

4.1.2. Формалізація понять алгоритмічності та декларативності

Спробуємо формалізувати деякі з впроваджених раніше понять. Введемо такі позначення:

q — *первинні інструкції*, записані без будь-яких змін у тому вигляді, в якому їх сформулював автор; аналогічно можна визначити *первинний опис ситуації* як результат її безпосереднього сприйняття;

S — множина факторів, які визначають поточний стан виконавців; розглядатимемо S як об'єднання двох множин: S_1 та S_2 , де S_1 — множина *контрольованих факторів*, які відомі авторові процедури і на які він може мати вплив, S_2 — множина *неконтрольованих факторів*;

Z — знання, які має виконавець;

r_A — робочий алгоритм, який формується та реалізується виконавцем при

алгоритмічному підході. При цьому, як правило, автоматично формулюється і програма p_A , що відповідає цьому алгоритму;

r_D — робочий алгоритм, який формується та реалізується інтелектуальним виконавцем при декларативному підході. Згідно з фундаментальними тезами Тьюринга та Черча (все, що може бути виконано будь-яким виконавцем, може бути промодельоване на машині Тьюринга) сам факт виконання завдання свідчить про існування цього алгоритму. Щоправда, цей алгоритм може і не усвідомлюватися виконавцем, а тому не може бути явно сформульований у вигляді алгоритму чи програми (якщо ж програма, що відповідає алгоритму r_D , повинна бути згенерована, йдеться мова про *задачу синтезу програм*).

Тоді можна записати такі співвідношення:

$$\begin{aligned} r_A = \\ f(q, S_1), \\ r_D = g(q, S_1, S_2, Z) \end{aligned}$$

Принциповим є наступне. Алгоритм r_A формується на основі первинних інструкцій q однозначно. При цьому також можуть відбуватися зміна і поповнення первинних інструкцій, але цей процес має повністю контрольований і детермінований характер. Як приклад можна навести компіляцію програм, написаних мовами високого рівня. Тому автор процедури може бути впевнений у гарантованому результаті (якщо, звичайно, були дотримані певні формальні вимоги до первинних інструкцій і забезпечено належний стан виконавця).

На противагу цьому, за декларативного підходу такої впевненості немає. Інтелектуальний виконавець певним чином поповнює первинні інструкції, але їх авторів не завжди відомо, яким чином це поповнення відбувається. Тому не можна бути впевненим у результаті виконання інструкцій, і ця втрата гарантованості є неминучою платою за відмову від алгоритмічності. Отже, ми маємо справу з узагальненням поняття алгоритму, яке дістало назву "**квазіалгоритм**".

4.2. Квазіалгоритми

Алгоритмом називається чітка однозначна зрозуміла виконавцеві послідовність інструкцій, виконання яких обов'язково приводить до гарантованого результату за скінченний час. На відміну від цього, інструкції квазіалгоритму можуть бути не зовсім чіткими, і результат виконання квазіалгоритмічної процедури не обов'язково є гарантованим.

Можна виділити як мінімум чотири основні джерела квазіалгоритмічності:

1. **Дія випадкових чинників**, що не залежать від виконавця. Строго кажучи, зогляду на цей фактор квазіалгоритмом слід вважати будь-який, навіть найбільш формалізований, алгоритм. Алгоритм розрахований на роботу при певних умовах; якщо ці умови зміняться, алгоритм може не призвести до потрібного результату. *Якщо ж не вдається чітко окреслити межі застосування алгоритму або забезпечити виконання необхідних умов для його*

роботи, ми маємо справу зі справжнім квазіалгоритмом.

2. **Недостатнє врахування автором алгоритмічної процедури особливостей виконавця.** Це призводить до того, що виконавець неправильно розуміє, що від

нього вимагається. Процедура може бути в принципі алгоритмічною, за нормальних умов призводити до гарантованого результату, але цей результат може бути не передбачений автором і тому бути для нього несподіванкою.

3. **Нечіткість формулювань**, що фігурують в описі процедури. Розглянемо будь-який кулінарний рецепт, наприклад: “розтерти тісто, змішати з дрібно нарізаними яблуками, додати солі і перцю за смаком і смажити до появи рум’яної скоринки”. Це є типовим прикладом нечіткості формулювань, що призводить до невизначеностей. До якої межі слід розтирати тісто? Що означає “дрібно нарізані”? Що означає “за смаком” і т. д. Неінтелектуальна система, орієнтована на чисто алгоритмічне керування, просто не зрозуміє цього опису і тому не зможе його виконати. Інтелектуальна ж система, наприклад, людина, повинна спробувати поповнити і уточнити цей опис на основі наявних знань і досвіду. Сам по собі факт виконання завдання буде свідчити про те, що квазіалгоритмічний первинний опис процедури був зведений до деякого внутрішнього алгоритму, але навряд чи виконавець зможе чітко пояснити і розписати цей алгоритм. Крім того, ці внутрішні алгоритми для кожного виконавця будуть різними.

4. **"Свобода волі"**, яку можуть мати високоорганізовані, по-справжньому інтелектуальні системи. Ці системи можуть мати свої цілі, і, якщо дана процедура суперечить цим цілям, вони можуть відмовитися її виконувати або виконати не так, як це від них вимагається. “Свобода волі” полягає у тому, що інтелектуальна система самостійно приймає рішення про свою поведінку і в залежності від цих рішень може виконувати зовнішні розпорядження, не виконувати їх, або ж виконувати так, як вона вважає за потрібне.

4.3. Характеристика інтелектуальних систем з точки зору кібернетики

4.3.1. Означення інтелектуальної системи

Інтелектуальна система може припускати зовнішнє керування, але для неї характерною є **самокерованість**. Система має певну мету і прагне так **планувати свої дії, щоб досягати цієї мети**. Як вхідні стимули системи можна розглядати поточну ситуацію, що сприймається і аналізується системою. Результатом реакції системи стає зміна зовнішньої ситуації, і поведінка системи коригується в залежності від того, бажаною чи небажаною є ця зміна.

Людина має певну суму **знань про світ**, яка дозволяє їй орієнтуватися в життєвих ситуаціях та приймати правильні рішення. Крім того, людина вміє певним чином **використовувати ці знання**. Ці самі риси мають мати системи штучного інтелекту.

Також можна стверджувати, що **здатність до поповнення первинних знань, є однією із ключових рис інтелектуальних систем**. Ця властивість

інтелектуальних систем називається **здатністю до навчання**. Розрізняють *зовнішнє навчання та самонавчання*.

Підсумувавши усі сказане, можна стверджувати, що **інтелектуальною системою називається самокерована кібернетична система, яка має певну суму знань про світ і здатна на основі безпосереднього сприйняття і подальшого аналізу поточної ситуації до планування дій, спрямованих на досягнення мети, а також донавчання**.

4.3.2. Типова схема функціонування інтелектуальної системи

Функціонування інтелектуальної системи можна описати як постійне прийняття рішень на основі аналізу поточних ситуацій для досягнення певної мети. Схема функціонування інтелектуальної системи наведена на рис. 4.1.

Виокремлюють наступні *етапи функціонування інтелектуальних систем*:

1. Безпосереднє сприйняття зовнішньої ситуації.

Результатом є первинний опис ситуації.

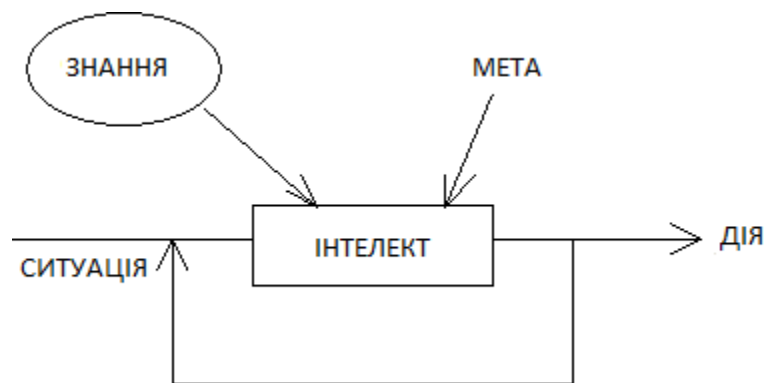


Рис. 4.1. Схема функціонування інтелектуальної системи.

2. **Зіставлення первинного опису зі знаннями системи і поповнення цього опису.** Результатом є формування вторинного опису ситуації в термінах знань системи. Цей процес можна розглядати як процес *розуміння ситуації* або як процес перекладу первинного опису на внутрішню мову системи. При цьому можуть змінюватися внутрішній стан системи та її знання.

Вторинний опис може бути не єдиним, і система має змогу вибирати між різними вторинними описами. Крім того, система в процесі роботи може переходити від одного вторинного опису до іншого. Якщо ми можемо формально задати форми внутрішнього представлення описів ситуацій та операції над ними, то сподіватимемося на певний автоматизований аналіз цих описів.

3. **Планування цілеспрямованих дій та прийняття рішень, аналіз можливих дій та їх наслідків і вибір тих дій, що найкраще узгоджуються з метою системи.** Таке рішення зазвичай формулюється певною внутрішньою мовою (свідомо або підсвідомо).

4. Зворотна інтерпретація прийнятого рішення, тобто формування робочого алгоритму для реагування системи.

5. Реалізація реакції системи; наслідками є зміни зовнішньої ситуації і внутрішнього стану системи і т. д.

Проте не слід вважати, що зазначені етапи є повністю відокремленими у тому розумінні, що наступний етап починається тільки після того, як повністю закінчиться попередній. Навпаки, для функціонування інтелектуальної системи характерним є взаємне проникнення цих етапів. Наприклад, ті чи інші рішення можуть прийматися вже на етапі безпосереднього сприйняття ситуації.