

## **Тема 5. Логічне виведення. Типи логічного виведення. Приклади комп'ютерної обробки експертних знань.**

Методи логічного виведення. Пряме та зворотне логічне виведення. Нечітке логічне виведення.

Інтелектуальність програмної системи визначається її здатністю проводити логічні міркування. В даний час головна увага приділяється створенню та експлуатації систем чітких і нечітких логічних міркувань.

### **Системи чітких міркувань**

Традиційно в поданні знань виділяють формальні логічні моделі, засновані на класичному вирахованні предикатів 1-го порядку, коли предметна область або задача описується у вигляді набору аксіом. Реальне вираховання предикатів 1-го порядку в промислових експертних системах практично не використовується. Ця логічна модель застосовна в основному в дослідницьких "іграшкових" системах, тому що пред'являє дуже високі вимоги й обмеження до предметної області. У промислових же експертних системах використовуються різні її модифікації й розширення.

Найбільше поширення одержала продукційна модель подання знань. При її використанні база знань складається з набору правил, а програма, що управляє перебором правил, називається машиною висновку.

Машина висновку (інтерпретатор правил) — це програма, що імітує логічний висновок експерта, що користується даною продукційною базою знань для інтерпретації даних, що надійшли в систему. За звичай вона виконує дві функції:

- перегляд існуючих даних (фактів) з робочої пам'яті (бази даних) і правил з бази знань і додавання (у міру можливості) у робочу пам'ять нових фактів;

- визначення порядку перегляду й застосування правил. Цей механізм управляє процесом консультації, зберігаючи для користувача інформацію про отримані висновки, і запитує в нього інформацію, коли для спрацьовування чергового правила в робочій пам'яті виявляється недостатньо даних. У переважній більшості систем, заснованих на знаннях, механізм висновку являє собою невелику по об'єму програму й включає два компоненти - один реалізує безпосередньо висновок, інший - управляє цим процесом.

Дія компонента висновку засновано на застосуванні правила, названого *modus ponens*: "Якщо відомо, що щиро твердження А, і існує правило виду "ЯКЩО А, ТО В", тоді твердження В також істинно". Таким чином, правила спрацьовують, коли перебувають факти, що задовольняють їхній лівій частині: якщо щиро посилку, то повинне бути істинно й висновок. Компонент висновку повинен функціонувати навіть при недоліку інформації.

Отримане рішення може й не бути точним, однак система не повинна зупинятися через те, що відсутня яка-небудь частина вхідної інформації.

Керуючий компонент визначає порядок застосування правил і виконує чотири функції: 1.Зіставлення— зразок правила зіставляється з наявними фактами.

2.Вибір — якщо в конкретній ситуації можуть бути застосовані відразу кілька правил, то з них вибирається одне, найбільш підходяще за заданим критерієм (дозвіл конфлікту).

3.Спрацьовування — якщо зразок правила при зіставленні збігся з будь-якими фактами з робочої пам'яті, те правило спрацьовує.

4. Дія — робоча пам'ять піддається зміні шляхом додавання в неї висновку правила, що спрацьовало. Якщо в правій частині правила втримується вказівка на яку-небудь дію, то воно виконується (як, наприклад, у системах забезпечення безпеки інформації).

Інтерпретатор продукцій працює циклічно. У кожному циклі він переглядає всі правила, щоб виявити тих, посилки яких збігаються з відомими на даний момент фактами з робочої пам'яті. Після вибору правило спрацьовує, його висновок заноситься в робочу пам'ять, і потім цикл повторюється спочатку

Робота машини висновку залежить тільки від стану робочої пам'яті й від складу бази знань. На практиці звичайно враховується історія роботи, тобто поводження механізму висновку в попередніх циклах. Інформація про поводження механізму висновку запам'ятовується в пам'яті станів (мал. 2.6). Звичайно пам'ять станів містить протокол системи.

Існують системи з прямим і зворотним висновком.

У системах із прямим висновком по відомих фактах відшукується висновок, що із цих фактів треб. Якщо такий висновок вдається знайти, то воно заноситься в робочу пам'ять. Прямий висновок часто називають висновком, керованим даними, або висновком, керованим антецедентами.

У зворотному висновку потрібно підтвердити обрану мету за допомогою наявних правил і даних.

### **Методи пошуку в глибину й завширшки**

У системах, база знань яких нараховує сотні правил, бажаним є використання стратегії керування висновком, що дозволяє мінімізувати час пошуку рішення й тим самим підвищити ефективність висновку. До числа таких стратегій ставляться: пошук у глибину, пошук завширшки, розбивку на підзадачі й альфа-бетаалгоритм.

При пошуку в глибину в якості чергової підцілі вибирається та, котра відповідає наступний, більше детальному рівню опису задачі. Наприклад, що діагностує система,

зробивши на основі відомих симптомів припущення про наявність певного захворювання, буде продовжувати запитувати уточнювальні ознаки й симптоми цієї хвороби доти, поки повністю не спростує висунуту гіпотезу.

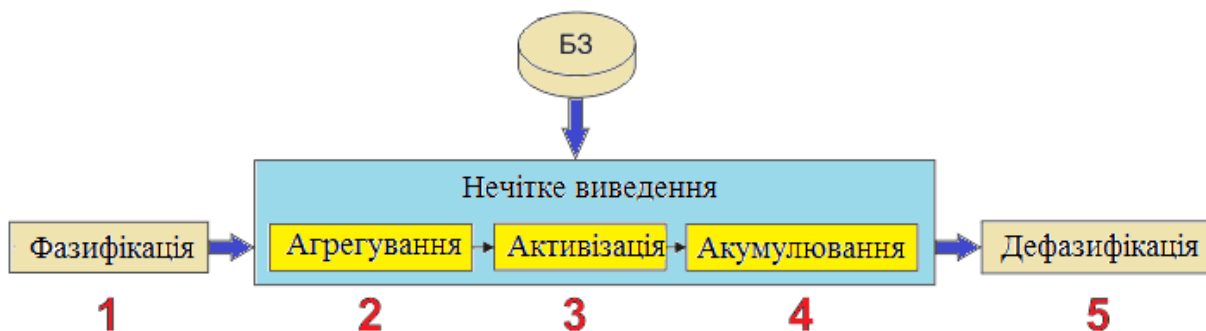
При пошуку завширшки, навпроти, система спочатку проаналізує всі симптоми, що перебувають на одному рівні простору станів, навіть якщо вони ставляться до різних захворювань, і лише потім перейде до симптомів наступного рівня детальності.

Розбивка на підзадачі має на увазі виділення підзадач, рішення яких розглядається як досягнення проміжних цілей на шляху до кінцевої мети. Прикладом, що підтверджує ефективність розбивки на підзадачі, є пошук несправностей у комп'ютері - спочатку виявляється підсистема, що відмовила (живлення, пам'ять і т.д.), що значно звужує простір пошуку. Якщо вдається правильно зрозуміти сутність задачі й оптимально розбити її на систему ієрархічно зв'язаних цілей-підцілей, то можна домогтися того, що шлях до її рішення в просторі пошуку буде мінімальний

Альфа-бета-алгоритм дозволяє зменшити простір станів шляхом видалення галузей, не перспективних для успішного пошуку. Тому проглядаються тільки ті вершини, у які можна потрапити в результаті наступного кроку, після чого безперспективні напрямки виключаються. Альфа-бета-алгоритм знайшов широке застосування в основному в системах, орієнтованих на різні ігри, наприклад, у шахових програмах.

### Системи нечітких міркувань

**Система нечіткого виведення** (англ. *fuzzy inference system*) – система логічного виведення, що базується на алгоритмі отримання нечітких висновків на основі нечітких передумов з використанням розглянутих основних понять нечіткої логіки. Процес нечіткого виведення поєднує в собі всі основні концепції теорії нечітких множин: функції належності, лінгвістичні змінні нечіткі логічні операції, методи нечіткої імплікації і нечітку композиції. Системи нечіткого виведення дозволяють вирішувати завдання автоматичного керування, класифікації даних, розпізнавання образів, прийняття рішень, машинного навчання та багато інших.



### **Формування бази знань у вигляді продукційних правил.**

База правил систем нечіткого виводу призначена для формального подання емпіричних знань або знань експертів в тій чи іншій проблемній області. У системах нечіткого виведення використовуються правила нечітких продукцій, в яких умови і висновки сформульовані в термінах нечітких лінгвістичних висловлювань.

Сукупність таких правил будемо далі називати базою правил нечітких продукцій. База правил нечітких продукцій є кінцевою множиною правил нечітких продукцій, узгоджених щодо використовуваних в них лінгвістичних змінних. Найбільш часто база правил представляється в формі структурованого тексту:

*ПРАВИЛО\_1: ЯКЩО «умова\_1, ТО висновок\_1» (F1)*

*ПРАВИЛО\_2: ЯКЩО «умова\_2, ТО висновок\_2» (F2)*

...

*ПРАВИЛО\_n: ЯКЩО «умова\_n, ТО висновок\_n» (Fn)*

де  $F_i$  – вагові коефіцієнти відповідних правил. Ці коефіцієнти можуть приймати значення з інтервалу  $[0, 1]$ . У разі, якщо ці вагові коефіцієнти відсутні, зручно прийняти, що їх значення дорівнює 1.

У системах нечіткого виведення лінгвістичні змінні, які використовуються в нечітких висловлюваннях передумови правил нечітких продукцій, часто називають вхідними лінгвістичними змінними, а змінні, які використовуються в нечітких висловлюваннях висновків правил нечітких продукцій, називають вихідними лінгвістичними змінними.

### **Фазифікація вхідних змінних**

В контексті нечіткої логіки під фазифікацією розуміється не тільки окремий етап виконання нечіткого виведення, а й власне процес або процедура знаходження значень функцій належності нечітких множин (термів) на основі звичайних (НЕ нечітких) вихідних даних. Фазифікацію ще називають введенням нечіткості.

Метою етапу фазифікації є встановлення відповідності між конкретним (зазвичай чисельним) значенням окремої вхідної змінної системи нечіткого виведення і значенням функції належності відповідного їй терма вхідної лінгвістичної змінної. Після завершення цього етапу для всіх вхідних змінних повинні бути визначені конкретні значення функцій належності по кожному з лінгвістичних термів, які використовуються в передумовах бази правил системи нечіткого виведення.

### ***Агрегування передумов в нечітких продукційних правилах.***

Агрегування (англ. *agggegation*) являє собою процедуру визначення ступеня істинності умов по кожному з правил системи нечіткого виведення. Формально процедура агрегування виконується наступним чином.

До початку цього етапу передбачаються відомими значення істинності всіх передумов системи нечіткого виведення, тобто множина значень . Далі розглядається кожна з умов правил системи нечіткого виведення. Якщо умова складається з декількох передумов, то визначається ступінь істинності складного висловлювання на основі відомих значень істинності передумов.

При цьому, для визначення результату нечіткої кон'юнкції або зв'язки «І» може бути використана одна з формул:

- **логічна кон'юнкція:**  $\mu (A \cap B) = \min\{ \mu (A), \mu (B)\}$ ,
- **алгебраїчний добуток:**  $\mu (A \cap B) = \mu (A) \cdot \mu (B)$ ,
- **граничний добуток:**  $\mu (A \cap B) = \max\{ (\mu (A) + \mu (B)) - 1, 0\}$ ,

де  $\mu (A)$ ,  $\mu (B)$  – функції належності нечітких висловлювань  $A$  та  $B$  відповідно.

Для визначення результату нечіткої диз'юнкції або зв'язки «АБО» може бути використана одна з формул:

- **логічна диз'юнкція:**  $\mu (A \cup B) = \max\{ \mu (A), \mu (B)\}$ ,
- **алгебраїчна сума:**  $\mu (A \cup B) = \mu (A) + \mu (B) - \mu (A) \cdot \mu (B)$ ,
- **гранична сума:**  $\mu (A \cup B) = \min\{ (\mu (A) + \mu (B)), 1\}$ ,

де  $\mu (A)$ ,  $\mu (B)$  – функції належності нечітких висловлювань  $A$  та  $B$  відповідно.

При цьому значення використовуються як аргументи відповідних логічних операцій. Тим самим знаходяться кількісні значення істинності всіх умов правил системи нечіткого виведення.

Етап агрегування вважається закінченим, коли будуть знайдені всі значення для кожного з правил, що входять в базу правил системи нечіткого виведення.

### ***Активізація або композиція висновків в нечітких продукційних правилах***

Активізація (англ. *activation*) в системах нечіткого виведення являє собою процедуру або процес знаходження ступеня істинності кожного з висновків правил нечітких продукцій. Формально процедура активізації виконується наступним чином.

До початку цього етапу передбачаються відомими значення істинності всіх передумов системи нечіткого виведення, тобто множина значень та значення вагових коефіцієнтів для кожного правила. Далі розглядається кожне з висновків правил системи нечіткого виведення. Якщо висновком правила є нечітке висловлювання, то ступінь його істинності дорівнює алгебраїчному добутку відповідного значення на ваговий коефіцієнт.

Таким чином, знаходяться всі значення ступенів істинності висновків для кожного з правил, що входять в дану базу правил системи нечіткого виведення.

### ***Акумуляція висновків нечітких продукційних правил***

Акумуляція або акумулявання (англ. *accumulation*) в системах нечіткого виведення являє собою процедуру або процес знаходження функції належності для кожної з вихідних лінгвістичних змінних множини.

Мета акумуляції полягає в тому, щоб об'єднати або акумулявати всі ступені істинності висновків для отримання функції належності кожної з вихідних змінних. Причина необхідності виконання цього етапу полягає в тому, що висновки, які відносяться до однієї і тієї ж вихідної лінгвістичної змінної, належать різним правилам системи нечіткого виведення.

### ***Дефазифікація вихідної змінної***

Дефазифікація в системах нечіткого виведення являє собою процедуру або процес знаходження звичайного (НЕ нечіткого) значення для кожної з вихідних лінгвістичних змінних. Мета дефазифікації полягає в тому, щоб, використовуючи результати акумуляції всіх вихідних лінгвістичних змінних, отримати звичайне кількісне значення (*crisp value*) кожної з вихідних змінних, яке може бути використане спеціальними пристроями, що не належать до системи нечіткого виведення.

### **Основні алгоритми нечіткого виведення**

Розглянуті вище етапи нечіткого виводу можуть бути реалізовані неоднозначним чином, оскільки включають в себе окремі параметри, які повинні бути фіксовані або специфіковані. Тим самим вибір конкретних варіантів параметрів кожного з етапів визначає алгоритм, який в повному обсязі реалізує нечіткий висновок в системах правил нечітких продукцій. До теперішнього часу запропоновано кілька алгоритмів нечіткого виведення