

ПОДАННЯ ТА ОБРОБЛЕННЯ ТЕХНОЛОГІЧНИХ ЗНАТЬ ПРО ПРОЦЕС БУРІННЯ НА ОСНОВІ ОБМЕЖЕНЬ

Р.Б. Вовк, В.Д. Мельник, Л.М. Гобир

ІФНТУНГ, 76019, м. Івано-Франківськ, вул. Карпатська, 15, тел. (03422) 42127,
e-mail: wolf@wolf.if.ua

При застосуванні сучасних інформаційних технологій в нафтогазовій предметній області особливе місце посідають інтелектуальні інформаційні системи і системи штучного інтелекту. До типових проблем, що можуть бути вирішені таким чином, можна віднести зокрема управління процесом буріння як складним нестационарним технологічним процесом; пошук оптимальних режимів буріння; діагностика перебігу технологічного процесу; виявлення та своєчасна ліквідація нештатних аварійних ситуацій в процесі буріння нафтових і газових свердловин.

Розглядаються методи роботи із знаннями про процес буріння на основі теорії представлення та задоволення обмежень. Здійснено дослідження алгоритмів і формалізмів представлення таких знань на основі обмежень, описано типи задач та структури знань, які ефективно опрацьовуються за їх допомогою.

Результатом дослідження є удосконалення алгоритмів подання знань з допомогою обмежень і реалізація на їх основі інтелектуального модуля, що здатний зберігати та опрацьовувати знання нафтогазової предметної області і може бути використаний як ядро системи інтелектуальної підтримки прийняття рішень в процесі буріння нафтових і газових свердловин.

Ключові слова: база знань, буріння свердловин, нештатні ситуації, технологічні обмеження, домени, правила, змінні, інтелектуальна система.

При применении современных информационных технологий в нефтегазовой предметной области особое место занимают интеллектуальные информационные системы и системы искусственного интеллекта. К типичным проблемам, которые могут быть решены таким образом, можно отнести в частности управление процессом бурения как сложным нестационарным технологическим процессом, поиск оптимальных режимов бурения; диагностика протекания технологического процесса; выявление и своевременная ликвидация нештатных аварийных ситуаций в процессе бурения скважин.

Рассматриваются методы работы со знаниями о процессе бурения на основе теории представлений и удовлетворения ограничений. Проведено исследование алгоритмов и формализмов представления таких знаний на основе ограничений, описаны типы задач и структуры знаний, которые эффективно разрабатываются с их помощью.

Результатом исследования является усовершенствование алгоритмов представления знаний с помощью ограничений и реализация на их основе интеллектуального модуля, который способен хранить и обрабатывать знания нефтегазовой предметной области и может быть использован как ядро системы интеллектуальной поддержки принятия решений в процессе бурения скважин.

Ключевые слова: база знаний, бурение скважин, нештатные ситуации, технологические ограничения, домены, правила, переменные, интеллектуальная система.

By the application of the modern information technologies in oil and gas subject area the special place is occupied by intelligent information systems and the systems of artificial intelligent. To the typical problems, that can be solved properly, the issue of running the drilling process belongs as complex non-steady technological process; the search for optimal drilling techniques; the diagnostic of technological processes progress; identification of emergencies and timely emergencies response during the drilling process in oil and gas wells. The drilling methods are considered which include data about the drilling process based on the theory of presentation and meeting constraints.

The research of relevant algorithms and formal descriptions has been carried out, data based on constraints have been proceeded, types of the tasks and structures of data have been described which are efficiently processed. The result of the research is the improvement of the algorithms for presenting information applying constraints and the implementation of intellectual module on their basis, which is capable of storing and processing information of oil and gas subject area and can be applied as the system center of the intellectual support for making decisions in drilling oil and gas wells.

Key words: data base, well drilling, non-standard situations, technological constraints, domains, rules, variables, intellectual system.

Представлення знань на основі обмежень [1-8] є одним з найновіших способів подання знань, що за останні кілька років швидко розвивається і стає все більш популярним. В його основі лежить концепція задоволення обмежень і програмування в обмеженнях (логічне або імперативне).

Даний спосіб подання знань є максимально декларативним порівняно з усіма попередніми і

визначається лише описом моделі предметної області, а не тільки алгоритмом її розв'язання відповідно. В свою чергу, модель подається у вигляді неупорядкованої множини відношень між технологічними параметрами в формі рівнянь, нерівностей та логічних виразів. Таку модель можна використовувати як у задачах прямого, так і зворотного пошуку, оскільки тут немає поділу параметрів на вхідні та вихідні.

Користувач вирішує, які з параметрів задані точно (детерміновані), які задані приблизно (з допомогою обмежень), а які є невизначеними. Загалом робота з системою обмежень полягає в чергуванні розрахунків і пошуку в просторі станів (коли знайдено деяке проміжне рішення).

Основним елементом таких систем на основі обмежень є бази знань. Проте, **недослідженим** залишається питання вибору способу подання знань, їх накопичення, збереження і опрацювання для отримання певного результату при вхідних даних заданих в формі обмежень.

Представлення знань на основі обмежень відіграє дуже важливу роль в інтелектуальних системах та системах інтелектуальної підтримки прийняття рішень в процесах нафтогазової предметної області, зокрема у процесі буріння нафтових і газових свердловин (НГС).

Тому **метою даної статі** є розроблення методів формалізації знань нафтогазової предметної області, зокрема знань про процес буріння НГС і пошуку відповідних способів їх подання в формі обмежень, при якому досягається максимально ефективний процес логічного висновку.

На формальному рівні задача задоволення обмежень формулюється у вигляді кортежу $(V; D; C)$, де $V = \{v_1, v_2, \dots, v_n\}$ – множина змінних (технологічних параметрів) системи; $D = \{D_1, D_2, \dots, D_n\}$ – множина доменів (областей визначення) кожної змінної; $C = \{c_1, c_2, \dots, c_n\}$ – множина обмежень, тобто відношень, заданих на підмножині змінних $\{v_{i_1}, \dots, v_{i_k}\}$ над їхніми доменами $\{D_{i_1} \times \dots \times D_{i_k}\} \supseteq c_i$.

Математичну модель бази знань нафтогазової предметної області на основі обмежень можна подати як кортеж:

$$M^{KB} = \langle V, D, O, C, R \rangle,$$

де V – набір змінних (технологічних параметрів), які можуть набувати значень, що належать певним множинам;

D – набір множин (областей допустимих значень змінних) або доменів;

O – операції, що застосовуються шляхом встановлення залежностей між змінними та перевірки здійснимості обмеження;

C – типи обмежень, що відображають різні види зв'язків між параметрами предметної області;

R – кінцева множина відношень між змінними та обмеженнями, що зв'язує інші складові і оновлюється у процесі постановки задачі та пошуку рішення.

В широкому розумінні обмеження можна розглядати як умову, що зв'язує один або декілька параметрів нафтогазової предметної області певними властивостями чи, відповідно, звужує область допустимих значень. Щоб зрозуміти можливості та правила роботи з базою знань на основі обмежень, необхідно розглянути основні поняття та означення логічного програ-

мування в обмеженнях. Формальна мова подання обмежень включає змінні, константи, атоми, обмеження, пропозиційні твердження, неявні рівності, факти, предикати, програми та запити.

Змінні зберігають певні значення протягом процесу пошуку рішення і подаються як множини $V = \{v_1, v_2, \dots, v_n\}$, де n – кількість змінних. Константи позначають значення, що не змінюються протягом часу виконання. Области допустимих значень змінних складають множину доменів $D = \{D_1, D_2, \dots, D_m\}$, де m – кількість доменів. Для довільної предметної області модель має задовольняти умові $n \geq m$ (немає сенсу вводити в модель домени, що не використовуються). Домен може бути як скінченною так і нескінченною множиною.

Атоми використовуються для введення позначень та функцій виклику процедур. Пропозиційні твердження – це спосіб опису цілі. Вони мають такий вигляд:

$$Goal : p \leftarrow c_1, c_2, \dots, c_k,$$

де p – результуючий атом;

c_i – інші атоми чи обмеження, від яких залежить p , $i = 1, 2, \dots, k$.

Обмеженням називають спеціальну форму задання залежностей між змінними та константами, що позначає не крок чи сукупність кроків для виконання, а певну властивість елемента знань. Завдяки таким властивостям обмеження містять декларативні, а не процедурні знання. Запишемо загальний вигляд обмежень:

$$C_j = C_j(v_1, v_2, \dots, v_n), j = 1..k,$$

де C_j – j -те обмеження системи (функція з операціями множини O);

v_1, \dots, v_n – змінні, між якими встановлюються залежності.

Функція залежності не обов'язково повинна включати всі змінні. Найчастіше обмеження накладається на два параметри (бінарна залежність) або взагалі на один (одноелементна область допустимих значень). Значенням функції може бути істина (якщо комбінація значень змінних задовольняють обмеженню) або хибна (у протилежному випадку).

Будь-яке твердження про предметну область, що не суперечить жодному з визначених у ній обмежень, вважається істинним. Для того, щоб задати параметри пошуку чи умови, у систему вводять конкретизуючі обмеження. Шукане рішення повинне перетворювати в істину весь набір рівностей. Для розв'язування систем обмежень можна застосовувати методи, аналогічні до відомих методів розв'язування систем математичних рівнянь. Проте інструменти роботи з обмеженнями до цього не зводяться, оскільки їх часто потрібно застосовувати для предметних областей, що не розглядаються класичною математикою. У зв'язку з цим математичний аспект пошуку знань, представлених через обмеження, не завжди є основним для формалізації.

База знань про процес буріння НГС на основі обмежень має специфічні характеристики і методи побудови висновку, що дає їй переваги для певних предметних областей. Крім цього, існують різні типи систем обмежень, тому для кожної конкретної задачі можна підібрати найбільш властиві.

Виділимо основні переваги концепції задоволення обмежень при поданні знань:

1. Зручний механізм задання множин (доменів).
2. Гнучка модель подання обмежень над змінними.
3. Ефективний механізм роботи з групами констант.
4. Широкі можливості у заданні типу обмеження.

Наведемо також деякі недоліки структур знань на основі обмежень:

1. Відсутність чіткого послідовного алгоритму роботи з обмеженнями, як і в інших формах подання знань.
2. Відсутність ефективного механізму реалізації обчислень та роботи з нескінченними доменами.
- 3 Слабка підтримка встановлення зв'язків між елементами домену.

Таким чином, концепція задоволення обмежень поєднує у собі переваги використання множин (доменів) і забезпечує гарантований результат при роботі із скінченими областями визначення, що ідеально підходить для подання декларативних знань.

Задача задоволення обмежень (*Constraint Satisfaction Problem – CSP*) визначається множиною змінних (технологічних параметрів) і множиною обмежень, де кожна змінна має непорожню область визначення можливих значень, а кожне обмеження включає деяку підмножину змінних і задає допустимі комбінації значень для цієї підмножини [4]. Стан задачі визначається шляхом присвоєння значень деяким або всім цим змінним. Присвоєння, яке не порушує жодних обмежень, називають сумісним (або допустимим) присвоєнням, а повним називають присвоєння, в якому беруть участь всі змінні.

Рішенням задачі може бути лише присвоєння, що задовольняє всім обмеженням. Інколи задачу зручно подати візуально у вигляді графа обмежень, вузли якого відповідають змінним задачі, а дуги – обмеженням. Існують також задачі з нетиповими цілями, наприклад, для деяких задач потрібно знайти рішення, яке б максимізувало цільову функцію.

Розглядаючи певну задачу у вигляді проблеми CSP, можна отримати важливі результати [2]. Так, представлення станів в технологічній проблемі відповідає деякому стандартному шаблону (тобто відображається у вигляді множини змінних з присвоєними значеннями), тому функцію визначення наступного стану і перевірку мети можна записати в універсальній формі, що може застосовуватись до всіх задач предметної області. Більш того, можуть бути розроблені ефективні, універсальні евристичні

функції, для створення яких не потрібні додаткові знання про конкретну проблемну область. Для спрощення процесу розв'язання може використовуватися сама структура графу обмежень, що дозволяє в деяких випадках домогтися експоненціального зменшення складності.

Будь-якій задачі з використанням обмежень, як і будь-якій стандартній задачі пошуку, може бути надане інкрементне формулювання наступним чином:

1) *Початковий стан*. Порожнє присвоєння {}, у якому жодній змінній не присвоєно жодного значення.

2) *Функція визначення наступного значення*. Значення може бути присвоєне будь-якій неініціалізованій змінній, за умови, що вона не конфліктуватиме з іншими змінними, значення яким були присвоєні раніше.

3) *Перевірка мети*. Поточне присвоєння є повним.

4) *Вартість шляху*. Розглядається постійна вартість (наприклад, 1) для кожного етапу.

Кожне рішення має бути повним присвоєнням, і тому повинне знаходитися на глибині n , якщо є n змінних. Крім того, дерево пошуку розповсюджується тільки на глибину n . З цих причин для вирішення завдань на основі обмежень широко застосовуються алгоритми пошуку в глибину. До того ж такі задачі характерні тим, що сам шлях, по якому досягається деяке рішення, не є важливим. Тому може також використовуватися формулювання з повним станом, в якому кожним станом є повне присвоєння, що задовольняє або не задовольняє накладеному обмеженню. На основі такого формулювання добре діють методи локального пошуку.

Задача на основі обмежень у простій формі характеризуються тим, що в ній використовуються змінні, які є дискретними і мають кінцеві області визначення. До такого класу відноситься задача «розфарбовування карти», задача «восьми королев». Так, якщо максимальний розмір області визначення будь-якої змінної в задачі є рівним d , то кількість можливих повних присвоєнь експоненціально залежить від кількості змінних, тобто може бути обчислена. До категорії задач на основі обмежень зі скінченною областю визначення відносяться також булеві задачі. Таким чином, у найгіршому випадку можна припускати, що час вирішення такої задачі зі скінченною областю визначення буде близький до експоненціального.

Дискретні змінні також можуть мати нескінченні області визначення, наприклад, множина всіх цілих чисел або множина всіх рядків. При вирішенні задач з нескінченними областями визначення більше немає можливості описувати обмеження, шляхом перерахування всіх допустимих комбінацій значень, або вирішувати задачу, перераховуючи всі можливі присвоєння, оскільки кількість подібних можливих присвоєнь є нескінченно великою. Замість цього повинна використовуватися спеціальна мова обмежень. Для лінійних обмежень з цілочисельними змінними (де кожна змінна подається

тільки в лінійній формі) існують спеціальні алгоритми пошуку рішень. Доведено, що не існує алгоритмів вирішення загальних нелінійних обмежень з цілочисельними змінними.

Задачі лінійного програмування можуть бути вирішені за час, який поліноміально залежить від кількості змінних. Крім того, проводилися дослідження задач з іншими типами обмежень і цільових функцій - задачі квадратичного програмування, конічного програмування другого порядку, тощо.

Окрім дослідження типів змінних, які можуть бути присутніми в задачах на основі обмежень, вивчають типи обмежень. Простим типом обмеження є унарне обмеження, яке обмежує значення єдиної змінної. Кожне унарне обмеження можна усунути, виконуючи попередню обробку області визначення відповідної змінної, щоб видалити будь-яке значення, що порушує це обмеження. Бінарне обмеження пов'язує між собою дві змінні. Бінарною задачею називається задача, в якій передбачені тільки бінарні обмеження; вона може бути подана у вигляді графа обмежень. В обмеженнях вищого порядку беруть участь три або більше змінних. Обмеження високого порядку можуть подаватися у вигляді гіперграфу обмежень. Проте деякі обмеження високого порядку можуть бути розбиті на декілька бінарних, шляхом введенням допоміжних змінних.

Всі обмеження, що розглядалися вище, були абсолютними обмеженнями, а їх порушення рівносильно тому, що якість потенційне рішення більше не розглядається. З іншого боку, в багатьох реальних задачах застосовуються обмеження з оцінками, що вказують на найбільш бажані, доцільні рішення. Отже, типи задач, що можуть бути вирішені за допомогою механізму побудови висновку на основі обмежень, досить різноманітні: пошук всіх можливих розв'язків, пошук оптимального рішення, розв'язування задач нечіткої логіки. Дані елементи ефективно використовуються для подання предметних областей різного роду і розроблення системи обмежень для роботи з цими областями та формулювання цілей.

Сучасні дослідження в області подання знань [6-7] спрямовані на розробку теоретичних засад та формалізмів концепції задоволення обмежень на основі баз знань, а також на розробку специфічних мов для зручного представлення обмежень різного роду над системами нелінійних рівнянь та нерівностей, де не існує формалізованих аналітичних методів розв'язання і застосування обмежень є найбільш перспективним при розроблянні інтелектуальних систем.

Реалізований програмний модуль подання знань на основі обмежень призначений для використання як його ядра інтелектуальної системи. У ньому було створено програмний інтерфейс для забезпечення таких можливостей:

а) побудови нафтогазової предметної області та внесення до неї змін;

б) формування конкретизуючих запитів користувача (оператора технологічного процесу);

в) повернення результату у зручній для обробки формі (надання експертної поради).

У створеній системі виділено такі особливості:

1) забезпечення можливості перенесення створеної бази з однієї ЕОМ на іншу і запам'ятовування отриманих результатів роботи для їх подальшого використання. Збереження всіх інформаційних елементів у бінарному файлі або їх перетворення у будь-яку іншу структуру (при наявності відповідного користувацького коду) забезпечує сумісність даного модуля з іншими програмами, які можуть використовуватись для прямого доступу до знань;

2) різноманітність типів знань, що підтримуються. У створену систему було включено механізми обробки знань у формі цілих чисел, чисел з плаваючою крапкою, дат, рядків, деревовидних структур тощо;

3) скінченність та зліченність діапазону можливих значень змінних. На даний момент було реалізовано підтримку лише скінченних областей визначення, що однак характерно майже для всіх систем роботи з обмеженнями для подання знань про технологічний процес;

4) динамічність та універсальність обмежень. Багато систем логічного висновку на основі обмежень передбачають окремо унарні, бінарні, тернарні обмеження тощо. У розробленій системі було вдосконалено таке подання і створено ряд універсальних обмежень, завдяки чому кожне обмеження працює з кількістю та типізацією аргументів, що відповідають встановленому інтервалу;

5) механізм висновку на основі обмежень, реалізований у модулі, що є алгоритмом зворотної перевірки (*back-tracking*). В ньому також використано систему задання пріоритетів обмеженням в залежності від їх типу і типу змінних, на які вони накладаються, що прискорило час перевірки обмежень та пошуку результатів і дало змогу уникнути зайвих операцій;

6) можливість задання списку типових задач для кожної предметної області, з яких при формуванні запиту користувача вибирається подібна і модифікується;

7) запам'ятовування останніх отриманих рішень і їх використання, у випадку, якщо запит користувача повторюється або є частковим по відношенню до вже виконаних.

Крім згаданих вище особливостей в оболонку інтелектуальної системи були включені інші сервісні елементи, щоб полегшити роботу з базою знань, зокрема підсистема перевірки виконуваних користувачем дій на їх доцільність і генерації зрозумілого повідомлення у випадку помилки з чітким вказуванням на її місце і причину.

Специфічною особливістю даного модуля є обчислення кількості часу, пам'яті і викликів обмежень, затрачених на процес пошуку рішення. Ця інформація зберігається для кожного рішення і може бути переглянута прикладною програмою. Завдяки цій можливості здійснювалось порівняння ефективності кількох різних

підходів задання предметної області за цими трьома основними показниками.

Деякі вдосконалення були запропоновані у механізмі виводу результатів. До них можна віднести групування результатів, налаштуваний вивід під час або після розрахунків, опис процесу пошуку рішення.

Для кращого розуміння механізму подання знань через обмеження, розглянемо типовий спосіб подання знань, що використовується найчастіше – на основі правил.

Основними елементами знань при описі виникнення нештатних ситуацій в процесі буріння НГС в базі знань на основі правил є:

Домен – набір значень певного типу, що використовуються для задання області визначення технологічних змінних, наприклад виду [9]:

$V^1 = \{ \text{порушення_режиму_циркуляції, розбурювання_глинистих_відкладень, підвищення_температури, відкладення_фільтраційної_кірки} \}$,

для процесу сальнікоутворення, що описується власною системою обмежень

$C^1 = \{ \text{негерметичність_бурової_колони, забруднення_стобура_свердловини, незадовільне_промивання, втрата_циркуляції, забруднення_приймальних_ємностей, накопичення_осаду, залипання_з_фільтраційною_кіркою, наявність_вимоїл, зниження_тиску, зниження_температури} \}$.

Тоді система доменів визначатиметься як:

$D^1 = \{ \text{періодичність_відриву_долота, припинення_циркуляції, глинисті_відкладення, розміри_каверн_і_жолоба, ступінчастість_стобура, прояви_затягування} \}$.

Константа – проіменоване значення, що використовується у фактах та правилах для певного параметра реального об'єкта або фізичної величини.

Факт – присвоєння певної властивості об'єкту або встановлення залежності між кількома об'єктами.

Правило – функція на основі однієї або кількох змінних при підстановці значень в яку, значення самої функції перетворюється на істину (що рівносильне додаванню нового факту з поточними значеннями змінних) або хибу (правило не використовується).

Кожне правило має ліву частину, де висувається гіпотеза, і праву частину, де висуваються умови, істинність яких свідчить про істинність гіпотези (для конкретних значень змінних). Ліва частина правила має такий же вигляд, як і тіло факту. Наприклад:

правило 1: якщо осьове зусилля на долото і бурову колону перевищує *граничні значення то* можливим є виникнення нештатної ситуації;

обмеження 1: граничні значення осьового зусилля осьового зусилля на долото і бурову колону визначають коректність процесу буріння;

правило 2: якщо спостерігається *значне збільшення* моменту на буровій колоні *і значне збільшення тиску бурового розчину і зменшення механічної швидкості буріння то, висока ймовірність* прихоплення бурової колони;

обмеження 2: прихоплення бурової колони визначається значним збільшення моменту на буровій колоні, тиску бурового розчину, зменшенням механічної швидкості буріння.

За допомогою фактів формуються декларативні знання, тоді як правила спрямовані на процедурну інформацію. При створенні бази знань спочатку визначаються всі константи (конкретні об'єкти, сталі величини) та домени. Далі на їх основі задаються правила, права частина яких може включати посилання на факти чи інші правила.

Під час пошуку рішення всі правила послідовно переглядаються з підстановкою на кожній ітерації поточних значень змінної. Якщо в результаті такої підстановки виявиться, що логічна формула у правій частині перетворилася в істину, то факт у лівій частині додається у базу і ітерація продовжується. Це виконується до тих пір, поки шукана ціль не буде знайдена або всі можливі комбінації змінних перевірені для кожного правила.

Недоліком в такому поданні є те, що неможливо одночасно виконати певну операцію з кількома об'єктами (вивести факт одразу для кількох об'єктів), оскільки вони не об'єднуються у множину однією командою. Тому, наприклад, для визначення певної властивості на множині об'єктів доводиться задавати цю властивість кожному об'єкту окремо через факти. Вивід будується так само окремо для кожного факту. Це призводить до того, що кількість фактів невпинно зростає.

Отже, представлення знань на основі фактів і правил має такий принцип: факти зберігають мінімальну кількість знань; всі інші знання виводяться з фактів на основі використання правил (розширення множини істинних тверджень).

Ключовими елементами подання знань у розробленому модулі є домени, константи та обмеження, що збігається із загальною концепцією задоволення обмежень.

В розробленому модулі використовуються характеристики обмежень, які можуть виражатись в лінгвістичній або числовій формі, а саме:

1) ваговий коефіцієнт обмеження, який визначає вагу кожного присвоєння змінним відповідних значень;

2) ймовірнісний коефіцієнт обмеження *срп* задає ймовірність задоволення (порушення) обмеження. Дана характеристика є статистичною. Ймовірнісні значення задоволення і по-

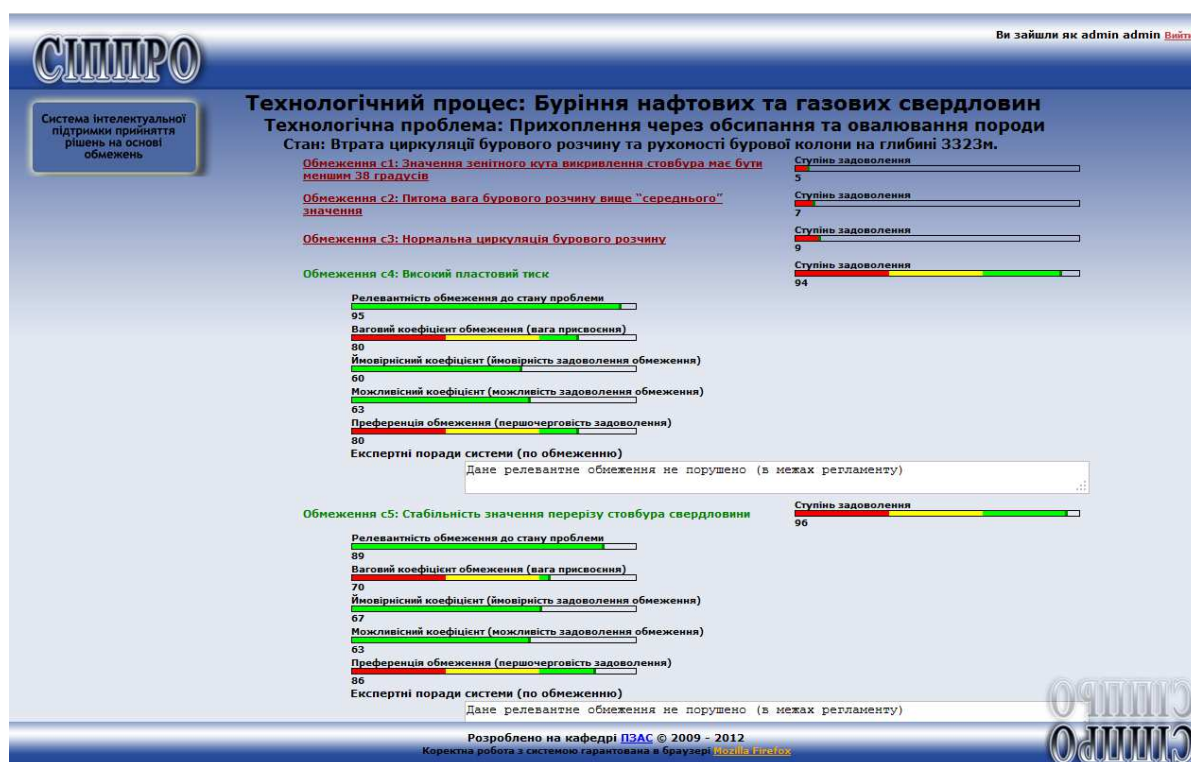


Рисунок 1 – Приклад експертної оцінки системи у випадку задоволених технологічних обмеженнях [10]

Таблиця 1 – Відмінності між продукційним підходом та концепцією CLP

Ознака	Факти і правила	Обмеження
Початкова інформація	Набір фактів	Набір доменів
Операційні елементи	Правила	Обмеження
Тип приведення початкового стану до шуканого	Розширення	Звуження
Умова істинності судження	Виведене або вже існуюче	Не заперечене

рушення обмеження зв'язуються формулою $cpr^{sat} = 1 - cpr^{viol}$;

3) можливісний коефіцієнт обмеження cps виражає можливість задоволення (порушення) обмеження. Можливісні коефіцієнти можна розглядати як різновид ймовірнісних коефіцієнтів, проте вони базуються не на статистичній оцінці, а на оцінці на основі знань. Можливісні коефіцієнти значення задоволення і порушення обмеження є взаємозв'язаними і виражаються формулою $cps^{sat} = 1 - cps^{viol}$;

4) преференцію обмеження pfc , що виражає важливість першочерговості задоволення (порушення) обмеження при вирішенні технологічної проблеми. Очевидною тезою, є те що степінь преференції обмеження повинен бути прямо пропорційним ступеню її релевантності, проте точний опис такої залежності визначатиметься вибраною стратегією;

5) ступінь релевантності rd обмеження стану технологічної проблеми;

6) лінгвістичне значення обмеження LV .

Залежно від вибору користувача характеристики 1-5 можуть задаватися у вигляді числових значень з діапазону [0-100], (тобто у вигляді процентної характеристики), або у вигляді лінгвістичного значення. Технологічно обидва види оцінок будуть переводитися в шкалу 0-1, що дозволяє уніфікувати процес виконання обчислень в механізмі висновку інтелектуальної системи.

Оскільки структура знань формується на як множина доменів, то підтримується ряд додаткових можливостей. Зокрема, завдяки тому, що існує можливість формувати домени будь-якого вмісту, можливо створити домен доменів, то у графовій інтерпретації має вигляд ієрархічної гілки. Таким чином, через вкладення доменів можливо створити будь-яку ієрархічну структуру. Дана властивість особливо зручна при поданні знань у формі дерев. Подібний принцип використовується для задання властивості над елементом або доменом. Властивість – розширений домен що включає всі елементи і дерева, у яких ця властивість присутня. Обме-

ження – це функція від кількох змінних, що повертає булеве значення. Якщо при конкретних значеннях змінних на певній ітерації обмеження повернуло істину, то перевіряються решта обмежень. У випадку коли всі обмеження задоволені, то конкретні значення змінних є шуканим результатом.

Отже, подання знань на основі обмежень має такий принцип: домени зберігають максимальну кількість знань; істинні знання отримуються з усієї множини шляхом відтинання хибних значень, що не задовольняють обмеженням (звуження множини істинних тверджень в формі експертних оцінок).

Щоб подати множину однорідних об'єктів на основі обмежень, достатньо помістити їх у спільний домен. Для того, щоб задати ієрархічну структуру або встановити залежності між знаннями, можна побудувати дерево через багаторазове вкладення доменів один в одного. Додатково можна визначити набір обмежень на цьому дереві. Для того, щоб задати властивість, необхідно створити домен і помістити в нього всі об'єкти та дерева об'єктів, що володіють цією властивістю. Інші елементи створюються з допомогою налаштовуваних доменів різного роду задаванням граничних меж в рамках продукційного підходу та підходу на основі обмежень (програмування в обмеженнях CLP).

В розробленій системі передбачено такі налаштування:

1) встановлення меж цілочисельного та дійсного інтервалу, які мають використовуватись користувачем, щоб уникнути помилки про перевищення допустимих значень;

2) оцінка системної похибки для дійсних чисел; дійсні числа вважаються рівними, якщо їх різниця за модулем не перевищує дане значення;

3) визначення ширини крок розбиття дійсного інтервалу при пошуку задовільного значення дійсної змінної;

4) введення змінної для визначення, того факту чи виводитиметься інформація про предметну область після завантаження;

5) введення змінної для визначення стратегії виведення результатів; якщо об'єднання результатів ввімкнене, то всі результати, які можуть бути об'єднані (мають одну відмінність), об'єднуються із заміною відмінних значень на множині з цими двома значеннями;

6) визначення кількості останніх результатів, які запам'ятовуються у базі знань. При отриманні нового розв'язку система перевіряє кількість розв'язків у базі знань. Якщо кількість рівна максимальному розміру, то старі розв'язки видаляються, а нові додаються.

7) оцінка модульності бази знань – найважливіше налаштування, що впливає на механізм перевірки обмежень. Якщо модульність не підтримується предметною областю, то всі рішення мають задовольняти всім обмеженням. Якщо ж підтримується, то перевіряються лише ті обмеження, які мають зв'язок до цілей або до змінних, пов'язаних з цілями іншими обмеженнями.

У розробленій інтелектуальній системі передбачено кілька різних типів обмежень, що накладаються на змінні. Всі вони мають спільний механізм перевірки умови задоволення. Кожне обмеження володіє такими властивостями: діапазон кількості аргументів, що підтримується; інформацію про допустимі типи аргументів; формула для перевірки задоволеності. При створенні обмеження перевіряються типи доменів аргументів і їх відповідність обмеженню. При перевірці обмеження вхідні змінні перевіряються на визначеність і відповідність їх значень типу обмеження.

Статусом обмеження (*constraint status*) є результат, що повертає обмеження після виконаної перевірки. На основі статусу визначаються подальші дії, які повинна виконати система для правильної та оптимальної роботи.

У розробленому модулі передбачено, що обмеження в результаті запиту може знаходитись у одному з чотирьох статусів:

- а) «Не властиве» (не включає цільових змінних або пов'язаних з ними);
- б) «Невизначено» (включає шукані змінні, але деякі змінні невизначені);
- в) «Незадоволене» (умова обмеження незадоволена);
- г) «Задоволене» (обмеження задоволене).

Проаналізуємо види обмежень, що використовуються у системі:

1) *Порівняльні обмеження* – виконують перевірку змінних на рівність, попарну відмінність та їх порівняння з певними величинами і т.п.;

2) *Синтаксичні обмеження* (для роботи з рядками). Включають перевірки рядків на наявність початку та кінця і вказаних входжень;

3) *Ієрархічні обмеження* (для роботи з деревами). Включають перевірку наявності спільної батьківської вершини, дочірніх вершин тощо.

4) *Універсальні обмеження*. Дані обмеження здатні обробляти формули з будь-якими операціями з реалізованої множини. Функціонування складається з кількох етапів: аналіз та перевірка формули, розбиття формули на елементи, впорядкування операцій за їх пріоритетність і виконання на основі транслітерації методом таблиць прецеденції, повернення отриманого результату виразу.

У розробленій системі передбачена можливість вибору алгоритму для пошуку рішення. Дана можливість реалізується шляхом визначення загального типу для розв'язувачів *Solver*, що як вхідне значення приймає поставлену ціль *Goal* (включає в себе перелік шуканих змінних, перелік використовуваних при цьому обмежень і граничне значення по кількості шуканих варіантів), і повертає об'єкт *Solution* (містить посилення на поставлену ціль).

Для побудови висновку у даній системі було використано розв'язувач, що реалізує алгоритм бек-трекінгу та використовує пошук у глибину. Даний алгоритм часто виявляється малоефективним при розв'язуванні задач, пов'язаних з обчисленнями на множині цілих

та дійсних чисел. Але при використанні для представлення знань, де числові параметри зустрічаються рідко і переважають категорійні домени, він дає приблизно таку ж ефективність.

Крім того, можливості цього алгоритму були розширені через використання пріоритизації обмежень, тобто механізму, при якому список обмежень сортується таким чином, що спочатку виконуються обмеження з меншою кількістю операцій. Даний засіб дає змогу підвищити ефективність пошуку рішення, але лише з наявності в системі обмежень з різними пріоритетами.

Пріоритет обмеження у реалізованому модулі залежить від таких чинників:

а) кількості операцій, передбачених для даного обмеження;

б) кількості змінних, що входять у дане обмеження;

в) розміру домену кожної змінної, що входять в дане обмеження.

Загальна формула розрахунку пріоритету обмеження має такий вигляд:

$$P = -k_c \prod_{i=1}^n sized(v_i),$$

де n – кількість індексованих змінних, що входять в обмеження; v_i – змінна з індексом i ;

$sized()$ – функція, що повертає розмір домену змінної;

k_c – коефіцієнт, який є специфічним для кожного типу обмеження (найчастіше рівний кількості операцій, яку виконує обмеження з мінімально допустимою для нього кількістю змінних, для підтвердження того, що значення змінних задовольняють обмеженню).

Наведемо алгоритм виводу. Вихідною точкою є відсортований за пріоритетністю масив (або список) обмежень; множина неініціалізованих змінних; порожній масив результатів, порожній стек змінних. Всі обмеження по чергово перевіряються на умову задоволення. Якщо обмеження задоволені, то результат зберігається у масиві (на місці неініціалізованих змінних ставляться їх домени, оскільки у цьому випадку всі значення змінної задовольнятимуть обмеження). Якщо обмеження включає у себе цільову змінну, але вона або інші змінні не активовані, перед перевіркою такого обмеження, неініціалізовані змінні ініціалізуються початковими значеннями своїх доменів і додаються до стеку. Від так відбувається повернення до початку списку обмежень, які вони перевіряються на істинність ще раз. Якщо ітерація добігає кінця, а у стеку залишаються змінні, то остання змінна інкрементується (отримує наступне значення з домену), а якщо вона вже отримала максимальне значення, то деініціалізується і видаляється зі стеку, а замість неї інкрементується наступна змінна і т.д. Ітерація (перевірка всіх обмежень) виконується знову для нових значень змінних.

В розробленому модулі набуття знань передбачено три умови завершення роботи алгоритму:

1) знаходження достатньої кількості розв'язків;

2) після перевірки всіх обмежень стек змінних порожній або перехід до наступного кроку неможливий (у випадку, коли всі змінні у ньому мають максимальні значення своїх доменів);

3) всі варіанти цільових змінних перевірені (повністю або частково).

Результатом роботи алгоритму є масив отриманих комбінацій змінних, що задовольняють всі накладені технологічні обмеження.

Таким чином, важливим є спосіб правильного вибору моделі подання для предметної області і відповідної типізації знань. Доцільним є також внесення різноманітних модифікацій у вибрану модель з метою її максимальної оптимізації як основи для побудови програмного ядра інтелектуальної системи підтримки прийняття рішень. Все це також дає змогу суттєво спростити роботу з базою знань.

Висновки

В пропонованому дослідженні введено основні поняття концепції задоволення обмежень і структур знань про процес буріння, з якими вони асоціюються. На основі отриманих результатів показано, що знання про процес буріння на основі обмежень не поступаються формою та наповненням способу подання знань на основі фактів та правил, що в даний час є більш поширеним. Більше того, методологія обмежень дозволяє значно спростити випадок опису кількох об'єктів шляхом об'єднання їх в один домен, що в класичних моделях подання знань вирішується шляхом перелічення всіх цих елементів з відповідним присвоєнням значень кожному елементу. Проведений аналіз між поданням, що базується на правилах, та поданням в формі обмежень показав, що елементи знань в них подібні, але підхід до утворення знань та процедура побудови логічного висновку відрізняються. Проте, було встановлено, що нафтогазова предметна область, представлена в обмеженнях, вимагає вибору оптимального алгоритму пошуку рішення, що здатен суттєво спростити механізм висновку і уникнути зайвих операцій та перевірок.

Отримані результати та порівняння дозволили сформулювати основні вимоги до програмного модуля подання знань та його внутрішньої структури.

Таким чином, подання знань про процес буріння нафтових і газових свердловин на основі обмежень у реалізованому модулі отримало такі переваги:

1) вдалося стисло подати досить велику базу знань про нештатні аварійні ситуації та способи їх усунення;

2) стало можливим формулювання постановок технологічних задач, без створення нових функціональних елементів;

3) створений модуль може бути застосований до інших задач нафтогазової предметної області.

Подальші дослідження даного напрямку будуть зосереджені на виконанні логічних та алгебраїчних розрахунків задач буріння нафтових і газових свердловин, що не вирішується представленими формальними методами.

Література

- 1 Tsang E.P. Foundations of Constraint Satisfaction. / E.P. Tsang // Academic Press. 1993. – ISBN 0-12-701610-4. – Режим доступу: <http://www.brasil.net/edward/FCS.html>.
- 2 Tsang E.P. Solution synthesis in the constraint satisfaction problem / E.P. Tsang, N.Foster // Technical Report CSM-142, DeptBibliography 401 of Computer Science, University of Essex, 1990. – 143 p.
- 3 Попов Э.В. Искусственный интеллект. Модели и методы. Справочник / Э.В. Попов. – М.: Радио и связь, 1990. – 303 с.
- 4 Задачи удовлетворения ограничений. – Режим доступу: <http://rriai.org.ru/zadachi-udovletvoreniya-ogranicheniy-7.html>.
- 5 Kumar V. Algorithms for constraint satisfaction problem / V.Kumar // Magazine. – 1992. – p. 32-44.
- 6 Xie H., Henderson P, Kernahan M. Modeling and solving engineering product configuration problems by constraint satisfaction/ International journal of Production research, 2005, - p. 44-69.
- 7 Shepherdson J. C. Negation as failure: A comparison of Clark's completed database and Reiter's closed world assumption / J. C. Shepherdson. // Logic Programming. – 1984.
- 8 Wu W., Hsiao M.S. Mining global constraints for improving bounded sequential equivalence checking / W. Wu, M.S. Hsiao // San Francisco. – CA, USA: IEEE DAC. – 2006.
- 9 Шавранський М.В. Система контролю для запобігання прихопленнь бурильної колони в процесі буріння: Дис. канд. техн. наук: 05.11.13 / Шавранський Михайло Васильович. - Івано-Франківськ, 2003. - 168 с.
- 10 Вовк Р.Б. Система інтелектуальної підтримки прийняття рішень при контролі технологічних параметрів / Р.Б. Вовк, В.І. Шекета, В.Д. Мельник // Методи і прилади контролю якості. – 2012. – № 29. – С. 119 – 129.

*Стаття надійшла до редакційної колегії
26.02.13*

*Рекомендована до друку
професором Тимківим Д.Ф.
(ІФНТУНГ, м. Івано-Франківськ)
професором Петришиним Л.Б.
(Прикарпатський національний університет
ім. В. Стефаника, м. Івано-Франківськ)*