Академія наук України
Інститут кібернетики імені В. М. Глушкова

На правах рукопису

УДК 007::159.955

РОМАНЕНКО Ігор Борисович

МАШИННІ МЕТОДИ ПОБУДОВИ ВИВЕДЕННЯ У ДЕЯКИХ НЕКЛАСИЧНИХ ЧИСЛЕННЯХ

05.25.05 — інформаційні системи та процеси

Автореферат дисертації на здобуття вченого ступеня кандидата технічних наук

Київ 1993

Робота виконана в Інституті кібернетики імені В. М. Глушкова АН України.

Науковий керівник: доктор технічних наук, професор
КОВАЛЬ В. Н.

Офіційні опоненти: доктор технічних наук, професор
ЛІТВІНОВ В. В.,
кандидат фізико-математичних наук
ДЕГТЯРЬОВ А. І.

Провідна установа: Київський політехнічний інститут.

Захист відбудеться «29» червня 1993 р. о 14⁰⁰ год. на засіданні спеціалізованої вченої ради К 016.45.05 при Інституті кібернетики імені В. М. Глушкова АН України за адресою:

252207 Київ 207, проспект Академіка Глушкова, 40.

З дисертацією можна ознайомитись у науково-технічному архіві інституту.

Автореферат розісланий «20» травня 1993 р.

Учений секретар
спеціалізованої ради РЕВЕНКО В. Л.

ЗАГАЛЬНА ХАРАКТЕРИСТИКА РОБОТИ

Актуальність теми. Останнім часом відмічається швидке зростання кількості застосувань математичної логіки в інформатиці. Як приклади можна назвати логічне програмування, японський проект ЕОМ п'ятого покоління, продукційні експертні системи, засоби для верифікації й аналізу програм та ін. В інформаційних системах досягнення математичної логіки можуть використовуватися для формалізації знань експертів, для аналізу складних запитів, при побудові наслідків, що спираються на інформацію, яка зберігається у інформаційній системі, а також при розробці інтерфейсу людина/ЕОМ, що реалізує обмежену підмножину природної мови. При цьому активно використовуються як класична логіка першого порядку, так і некласичні логіки. В ряді випадків використання некласичних логік має істотні переваги у порівнянні з використанням класичної логіки, оскільки у некласичних логіках, як правило, використовуються спеціальні механізми побудови виведення, моделювання яких за допомогою стандартних засобів може призвести до істотного зростання об'єму роботи. Останнє, по суті, може позбавити нас можливості розв'язувати реальні задачі за припустимий час, навіть з використанням обчислювальної техніки.

Один з широко розповсюджених підходів до розв'язання задач з використанням логічних засобів полягає у зведенні вихідної постановки задачі до задачі пошуку виведення деякої формули з множини формул. Наприклад, часто задача про побудову об'єкта з заданими властивостями може бути зведена до пошуку конструктивного доведення існування такого об'єкта. Таким чином, проблема автоматизації пошуку виведення у некласичних логіках є вельми актуальною.

Мета роботи. Вельми розповсюдженим методом автоматичного пошуку виведення для класичної логіки є метод резолюцій. До його переваг звичайно відносять: відсутність обмежень на клас розглядуваних формул, можливість ефективної реалізації, особливо на паралельних архітектурах, наявність істотних наробок як у галузі теоретичних досліджень властивостей і характеристик відповідних числень, так і в галузі побудови алгоритмів їх підтримки. Тому метою цієї роботи була побудова числень резолюційного типу для двох некласичних логік, а саме, для ДСМ-логіки і так званої ситуаційної логіки. Зрозуміло, що сам факт побудови числень не мав би цінності без доведення їх повноти і демонстрації практичної застосовності. Останнє потребувало створення програмної системи, що реалізує запропоновані алгоритми пошуку виведення.

Загальна методика досліджень полягає в аналізі відповідних логік з метою вирізнення виводимих об'єктів (аналогів літер та диз'юнктів для класичного випадку) з подальшою побудовою правила резолюцій. Доведення повноти відповідає широко розповсюдженій схемі, що використовує поняття семантичного дерева. В дослідженні застосовувалися методи математичної логіки та теорії графів. При розробці програм, що обговорюються у третьому розділі, використовувались сучасні технології програмування.

Наукова новизна. У роботі отримано такі нові результати:

побудовано числення резолюційного типу для ДСМ-логіки і доведено його коректність та повноту;

побудовано числення резолюційного типу для ситуаційної логіки і доведено його коректність та повноту;

введено та обгрунтовано поширені правила поглинання та факторизації для побудованих числень;

розроблено алгоритми підтримки побудованих числень і експериментально підтверджено їх практичну застосовність.

Практична цінність. Результати, які отримано у роботі, можуть використовуватися при створенні інтелектуальних інформаційних систем, наприклад експертних систем нового типу. Резолюційне числення для ДСМ-логіки дозволяє реалізувати змішане вірогідне і правдоподібне виведення в межах єдиної моделі формалізації міркувань.

Резолюційне числення для ситуаційної логіки дозволяє автоматизувати процес побудови міркувань у межах моделі, що передбачає формалізацію знань експерта у вигляді сукупності ситуацій із заданими відношеннями між ними.

Запропоновані модифікації правил поглинання та факторизації можуть істотно підвищити ефективність процесу побудови виведення.

Реалізація результатів досліджень. На базі результатів, отриманих у роботі, автор створив програми побудови резолюційного виведення як для ДСМ-логіки, так і для ситуаційної логіки. Опису основних алгоритмів цих програм і експериментів з ними присвячено третій розділ роботи. Програма побудови резолюційного виведення для ДСМ-логіки входить як складова частина до експертної системи з аналізу протипухлинної активності фармакологічних препаратів, яку розроблено у ВІНІТІ РАН.

Публікації і апробація роботи. Основні результати дисертації опубліковано в роботах [1-7]. Вони доповідались на семінарах товариства Курта Гьоделя (Відень), механіко-математичного факультету МДУ й Інституту кібернетики ім. В.М.Глушкова АН України. Програмна система демонструвалась на виставці Всесоюзної конференції з штучного інтелекту (Мінськ, 1990) і на міжнародній конференції з логічного програмування та автоматизації міркувань (Санкт-Петербург, 1992).

Структура роботи. Дисертація складається з вступу, трьох розділів, закінчення, списку літератури і додатку.

ЗМІСТ РОБОТИ

У вступі розглядається історія питання, наводиться огляд літератури.

Перший розділ роботи присвячено методу резолюцій для ДСМ-логіки. ДСМ-метод було розроблено у ВІНІТІ АН Росії групою під керівництвом В.К.Фінна. Головна мета розробки - формалізація міркувань, що містять в собі висування й оцінку гіпотез, які базуються на деякій множині експериментальних фактів вигляду "об'єкт X має множину властивостей Y". Тоді гіпотези можуть змістовно розумітися як твердження вигляду "підоб'єкт X' є причиною наявності (відсутності) множини властивостей Y'". Якщо маємо сукупність гіпотез, то можемо прогнозувати (довизначати) властивості об'єктів, базуючись на наявності або відсутності у них підоб'єктів.

Далі можна породжувати нові гіпотези, якщо розглядати вихідні і довизначені об'єкти і намагатися довизначити нові об'єкти. Кажуть, що ДСМ-вивід закінчено, якщо на черговій ітерації не довизначається жоден об'єкт.

Зрозуміло, що як твердження про наявність/відсутність властивостей у об'єктів, так і твердження, що відповідають гіпотезам, мусять мати вирізнені логічні статуси. Цього можна досягти, змінивши спосіб оцінки предикатних символів. А саме: нехай Λ - непорожня скінченна множина, елементи якої будемо звати істинносними значеннями, $\mathbb{N}$ - множина невід'ємних цілих чисел. Тоді значенням оцінки θ на атомі P(...) (де P - предикатний символ) буде пара $\langle\lambda,n\rangle$, де $\lambda \in \Lambda$ - логічний статус ствердження, а $n \in \mathbb{N}$ - номер ітерації алгоритму, на якій ствердження отримало цей статус. Предикатні символи, що оцінюються таким чином, будемо звати "кольоровими" (оскільки λ можна вважати "кольором", а n - його насиченістю). Поряд з ними до формул можуть входити і звичайні ("чорно-білі") предикати або формули, що оцінюються тільки в істину або хибність в класичному їх розумінні.

Змістовна трактовка може бути, наприклад, такою. Розглянемо $\Lambda = \{-1, 0, 1, \tau\}$, причому -1 відповідає "експериментальній хибності", 0 - "суперечності", 1 - "експериментальній істині" і τ - "невизначеності". Тоді значення оцінки, що дорівнює $\langle -1,0\rangle$ на твердженні вигляду $X \Rightarrow_1 Y$ (де $\Rightarrow_1$ - кольоровий предикатний символ, який розглядається як "мати властивість") означає, що на нульовій ітерації (тобто до того, як застосовувався ДСМ-метод) твердження про те, що об'єкт X мав множину властивостей Y мало статус "експериментальна хибність". Іншими словами, серед вихідних експериментальних фактів був факт про те, що об'єкт X не має множини властивостей Y. Значення оцінки, що дорівнює $\langle 1,1\rangle$, на твердженні того ж вигляду означає, що на першій ітерації було отримано прогноз про те, що об'єкт X повинен мати множину властивостей Y. Нарешті, значення оцінки $\langle\tau,2\rangle$ свідчить, що на другій ітерації не вдалося спрогнозувати наявність/відсутність множини властивостей Y у об'єкта X. Інший кольоровий предикатний символ - $\Rightarrow_2$ ("бути причиною") використовується для запису гіпотез про причини наявності/

відсутності властивостей. Наприклад, твердження вигляду $X' \Rightarrow_2 Y'$ оцінюється у <-1,1>, якщо на першій ітерації було породжено гіпотезу, що підоб'єкт X' є причиною відсутності множини властивостей Y'.

Звичайно, мову ДСМ-логіки розширюють двома операторами, а саме: $J_{\langle\lambda,n\rangle}$ і $J_{(\lambda,n)}$, які застосовуються лише до кольорових атомів. Значенням оцінки на формулі $J_{\langle\lambda,n\rangle}P(\ldots)$ буде істина тоді і тільки тоді, коли значення цієї ж оцінки на атомі $P(\ldots)$ дорівнює $\langle\lambda,n\rangle$. В усіх інших випадках значенням оцінки на формулі $J_{\langle\lambda,n\rangle}P(\ldots)$ є хибність. Значенням оцінки на формулі $J_{(\lambda,n)}P(\ldots)$ буде істина тоді і тільки тоді, коли значення цієї ж оцінки на атомі $P(\ldots)$ дорівнює $\langle\lambda,k\rangle$ і $k \leqslant n$. В усіх інших випадках значенням оцінки на формулі $J_{(\lambda,n)}P(\ldots)$ є хибність. Таким чином, формула, яка містить J-оператор на верхньому рівні, є чорно-білою. Наявність J-операторів дозволяє записати деякі цікаві твердження, наприклад:

$$\forall n(\ (J_{\langle-1,n\rangle}X\Rightarrow_1 Y \vee J_{\langle 0,n\rangle}X\Rightarrow_1 Y \vee J_{\langle 1,n\rangle}X\Rightarrow_1 Y) \Rightarrow \neg J_{\langle\tau,n+1\rangle}X\Rightarrow_1 Y)$$

означає, що об'єкт не може "втратити" своїх властивостей.

Наведений вище формалізм може застосовуватися для побудови так званого правдоподібного виведення. Наведемо як приклад один з найпростіших його варіантів. Нехай об'єктам відповідають множини, а розглядувані множини властивостей одноелементні (останнє, по суті, не є обмеженням). Процес побудови виведення складається з послідовності ітерацій, кожну з яких розіб'ємо на дві фази. У першій фазі (породження гіпотез) для кожної властивості у розглядаються всі твердження вигляду $J_{\langle 1,k\rangle}(X \Rightarrow_1 y)$, де $k < n$, n - номер ітерації. Для цих тверджень будуються всі непорожні максимальні перерізи X' множин у лівій частині $\Rightarrow_1$, і до сукупності тверджень додаються твердження $J_{\langle 1,n\rangle}(X' \Rightarrow_2 y)$ (позитивні гіпотези). Аналогічно по множині тверджень вигляду $J_{\langle-1,k\rangle}(X \Rightarrow_1 y)$ будуються негативні гіпотези (твердження вигляду $J_{\langle-1,n\rangle}(X' \Rightarrow_2 y)$). Далі, якщо існують позитивна і негативна гіпотези з однаковими лівою та правою частинами, то обидві ці гіпотези викреслюються, а до сукупності тверджень додається твердження вигляду $J_{\langle 0,n\rangle}(X' \Rightarrow_2 y)$

(суперечлива гіпотеза).

У другій фазі (довизначення фактів) для кожного твердження вигляду $J_{(\tau,n-1)}(X \Rightarrow_1 y)$ шукаються позитивні (або негативні) гіпотези $J_{\langle 1,n\rangle}(X' \Rightarrow_2 y)$ (відповідно $J_{\langle -1,n\rangle}(X' \Rightarrow_2 y)$), такі, що $X' \subseteq X$. Якщо знайдено лише позитивні (відповідно негативні) гіпотези, то твердження $J_{(\tau,n-1)}(X \Rightarrow_1 y)$ замінюється на $J_{\langle 1,n\rangle}(X \Rightarrow_1 y)$ (відповідно $J_{\langle -1,n\rangle}(X \Rightarrow_1 y)$). Якщо знайдено гіпотези обох знаків, або відповідні гіпотези не існують взагалі, то твердження $J_{(\tau,n-1)}(X \Rightarrow_1 y)$ замінюється на $J_{(\tau,n)}(X \Rightarrow_1 y)$.

Поряд з тільки що розглянутим правдоподібним виведенням для розв'язання багатьох задач може бути необхідним і використання так званого вірогідного виведення, наприклад, для перевірки, чи суперечить отримана сукупність гіпотез деяким заданим аксіомам. Прикладом може бути така аксіома:

$$\begin{aligned}\forall n\, \forall x\, \forall y\, (\, J_{\langle 1,0\rangle}(x \Rightarrow_1 y) \Rightarrow \neg(\, \exists z\, \exists w\, (\, & (z \subset w)\ \&\ \neg(z = \varnothing)\ \& \\ & \&\ (w \subseteq y)\ \&\ \neg(w = \varnothing)\ \& \\ & \&\ (\, J_{\langle -1,n\rangle}(z \Rightarrow_2 w) \vee \\ & \quad J_{\langle 0,n\rangle}(z \Rightarrow_2 w)\,)\,)\,)\,)\end{aligned}$$

Змістовно вона означає, що для кожного об'єкта, що має деяку множину властивостей, не повинно існувати гіпотез про те, що деяка частина цього об'єкта несе відповідальність за відсутність деякої підмножини множини властивостей, які мав вихідний об'єкт.

Як зазначалося вище, для розв'язання схожих задач звичайно використовується метод резолюцій. Побудові його аналогу для ДСМ-логіки присвячено перший розділ. Загальновідомо, що об'єктами виведення в резолюційному численні для класичної логіки є диз'юнкти, що розглядаються як множини літер. Аналогами літер для ДСМ-логіки можуть бути вирази вигляду $\llcorner\lambda,n\lrcorner P(\ldots)$ та $\ulcorner\lambda,n\urcorner P(\ldots)$, де $P(\ldots)$ - кольоровий атом (основа літери), $\lambda \in \Lambda$, $n \in \mathbb{N}$. Вирази $\llcorner\lambda,n\lrcorner$ та $\ulcorner\lambda,n\urcorner$ звичайно звуть знаками літер. Кажуть, що оцінка θ справджує літеру $\llcorner\lambda,n\lrcorner P(\ldots)$, якщо її значення на $P(\ldots)$ дорівнює $\langle\lambda,k\rangle$, причому $k \geqslant n$. Відповідно, оцінка θ справджує літеру $\ulcorner\lambda,n\urcorner P(\ldots)$, якщо її значення на $P(\ldots)$ дорівнює $\langle\lambda,k\rangle$, причому $k \leqslant n$. Літери

L_I і L_2 звуть контрарними, якщо жодна оцінка не справджує їх одночасно. Зауважимо, що контрарні літери повинні мати уніфікуємі (в звичайному розумінні) атоми. Диз'юнкти для ДСМ-логіки визначаються як множини літер, що розуміються як диз'юнкція літер. Диз'юнкт, що складається з порожньої множини літер, зветься порожнім диз'юнктом. Можна довести, що довільна ДСМ-формула може бути зведена до еквівалентної множини диз'юнктів, яка розуміється як кон'юнкція диз'юнктів.

Бінарною резольвентою диз'юнктів D_1 та D_2, таких, що $D_1 = \{L_1\} \cup D_1'$ і $D_2 = \{L_2\} \cup D_2'$, L_I контрарне до L_2 і найбільш загальний уніфікатор основ L_I і L_2 дорівнює σ, назвемо диз'юнкт $(D_1' \cup D_2')\sigma$. Фактором (склейкою) диз'юнкту $D = \{L_1\} \cup \{L_2\} \cup \ldots \cup \{L_k\} \cup D'$, $k \geq 1$, такого, що $L_I, \ldots, L_k$ мають однакові знаки і уніфікуємі з найбільш загальним уніфікатором σ, назвемо диз'юнкт $(\{L_1\} \cup D')\sigma$. Нарешті, резольвентою диз'юнктів D_I та D_2 $(R(D_I, D_2))$ назвемо їх бінарну резольвенту, або бінарну резольвенту їх факторів.

За таких означень власне правило резолюцій має такий же вигляд, як і в класичному випадку:

Правило резолюцій. Нехай Δ - множина диз'юнктів, $D_I \in \Delta$, $D_2 \in \Delta$ і існує резольвента $R(D_1, D_2)$. Тоді дозволяється додати $R(D_1, D_2)$ до Δ. Якщо на якійсь ітерації цього процесу ми отримаємо диз'юнкт D, то будемо казати, що D виводиться з Δ.

Істотною особливістю методу резолюцій для ДСМ-логіки є наявність потенційно нескінченної множини літер з однією і тією ж основою, але різними знаками. Для того, щоб зрозуміти, як це може вплинути на метод резолюцій, розглянемо простий приклад. Нехай P - пропозиціональна змінна, а (нескінченна) множина Δ складається з таких диз'юнктів: $\{\lfloor\lambda, 0\rfloor P\}$, $\{\lfloor\lambda, 1\rfloor P\}$, $\{\lfloor\lambda, 2\rfloor P\}$, ... Певна річ, довільна пара диз'юнктів з Δ не має контрарних літер, однак сама множина Δ не справджується жодною оцінкою. Тому обмежимося так званими "скінченно різноманітними" множинами диз'юнктів, де кількість входжень кожного предикатного символу до множини скінченна.

Головним результатом першого розділу є така теорема, що стверджує повноту методу резолюцій:

Теорема. Скінченно різноманітна множина диз'юнктів суперечлива тоді і тільки тоді, коли з неї можна вивести порожній диз'юнкт.

Доведення цієї теореми провадиться за допомогою семантичних дерев, тобто дерев, у яких кожне з ребер помічене основним атомом з оцінкою $\langle\lambda,n\rangle$. Легко бачити, що це дерево є нескінченно розгалуженим. Проте для кожної скінченно різноманітної множини диз'юнктів виявляється можливим побудувати аналогічне (у деякому розумінні) скінченно розгалужене дерево.

Далі в першому розділі розглядаються факторизація (склейка) та поглинання диз'юнктів. Раніше ми вимагали, щоб при породженні фактору знаки літер збігалися. Однак, не порушуючи повноти методу, можна поширити факторизацію і на деякі випадки, коли знаки склеюваних літер не збігаються. Оскільки фактор містить менше літер, ніж диз'юнкт, з якого його було отримано, поширення факторизації може призвести до підвищення ефективності процесу побудови виводу.

Слід особливо відмітити важливість правила поглинання з точки зору практичного застосування методу резолюцій. Дійсно, загальновідомо, що для класичної логіки спроба обійтися без нього призводить до катастрофічного засмічення пошукового простору "зайвими" диз'юнктами. Для ДСМ-логіки ситуація потенційно ще гірша, бо різних літер з даною основою може бути не дві, як в класичній логіці, а, загалом, довільна кількість. Таким чином, наявність правила поглинання, яке дозволяє видаляти з пошукового простору "зайві" диз'юнкти, істотно підвищує практичну цінність методу.

Решту першого розділу присвячено розгляду формул, які мають вигляд

$$\forall n\ \forall m\ (\ L_1(n)\ \varphi\ L_2(m)\ ),$$

де $L_1(n)$ та $L_2(m)$ - літери, які містять n і m в знаках, а φ - або кон'юнкція, або диз'юнкція. Для більшості таких формул наводяться еквівалентні формули, до яких не входять n і m. Важливість розгляду пов'язана, з одного боку, з тим, що формули, які мають вищезазначений вигляд, часто зустрічаються в множині диз'юнктів, а з іншого боку, - з тим, що застосування правила резолюцій до диз'юнктів, деякі літери яких містять змінні в знаках, пов'язане з додатковими накладними витратами. Тому можливість зменшити кількість літер зі змінними в знаках призводить до підвищення ефективності побудови виводу.

Другий розділ присвячено розгляду так званої ситуаційної логіки. Ця логіка, розроблена К.П.Вершиніним, призначена для формалізації міркувань експерта у важкоформалізуємих предметних галузях. Головною одиницею знань, вмінь та навиків експерту при цьому вважається ситуація, що розглядається як сукупність тверджень, понять, імперативних процедур, прикладів розв'язання задач та прецедентів (прикладів застосування процедур до розв'язання задач). Прикладом такої ситуації може бути проектування житлового будинку. Певна річ, твердження, які є істинними в одній ситуації, можуть бути хибними в іншій (не можна проектувати мікрорайон точнісінько так, як проектується один будинок). Крім того, в ситуації можуть бути здійснені дії, які, можливо, переводять її в іншу ситуацію. На множині ситуацій задаються два відношення: "бути більш загальною" та "бути підпорядкованою". Ситуація S є більш загальною ніж ситуація S′, якщо кожне твердження, істинне в S, є істинним і в S′. Ситуація S є підпорядкованою ситуації S′, якщо кожна дія, що змінює S (тобто переводить її в іншу ситуацію), змінює і S′. Таким чином, ми одержуємо ситуаційну структуру, що становить множину ситуацій з заданими на них відношеннями "бути більш загальною" та "бути підпорядкованою". В роботі Вершиніна та Гергея (1990) наводиться логічний формалізм SSL, який дозволяє, зокрема, виразити властивості ситуацій і ситуаційних структур.

Обмеження SSL на випадок, коли ситуаційна структура складається з однієї ситуації розглядається в [6]. Таке обмеження (SSL_0) може бути використане для опису ситуацій "зсередини". Головною його відмінністю від класичної логіки є запровадження третього істинносного значення, що розглядається як "невизначеність" ("незнання"). Введення третього істинносного значення пов'язане з необхідністю розгляду питання про істинність тверджень, які містять поняття, невідомі в даній ситуації. Отже, оцінка предикатного символу в SSL_0 може приймати одне з трьох значень: 0 (хибність), Λ (невизначеність) і 1 (істина). Ці значення вважаються упорядкованими таким чином: $0 < \Lambda < 1$.

Мова SSL_0 відрізняється від мови класичної логіки предикатів наявністю нової зв'язки: ~ (інволюція). Кон'юнкція та диз'юнкція розглядаються, відповідно, як мінімум та максимум в розумінні вказаного вище порядку. Нижче наведено таблиці істинності для заперечень та імплікації:

A	¬A	~A
0	1	1
Λ	0	Λ
1	0	0

A	0	0	0	Λ	Λ	Λ	1	1	1
B	0	Λ	1	0	Λ	1	0	Λ	1
A⇒B	1	1	1	0	1	1	0	Λ	1

Квантори загальності та існування розглядаються, відповідно, як нескінченні кон'юнкція та диз'юнкція.

Першу частину другого розділу присвячено розгляду таких понять, як логічний наслідок, справджуваність та суперечність відносно SSL_0. Виявляється, що логічний наслідок можна визначити одним з чотирьох шляхів, а саме:

$$\Gamma \vDash_1 F \Leftrightarrow \forall\theta\left(\min_{F'\in\Gamma} \theta(F') = v \text{ и } \theta(F) = v' \Rightarrow v \leqslant v' \right)$$
$$\Gamma \vDash_2 F \Leftrightarrow \forall\theta\left(\forall F'\in\Gamma \ (\theta(F') = 1) \Rightarrow \theta(F) = 1 \right)$$
$$\Gamma \vDash_3 F \Leftrightarrow \forall\theta\left(\forall F'\in\Gamma \ (\theta(F') \neq 0) \Rightarrow \theta(F) \neq 0 \right)$$
$$\Gamma \vDash_4 F \Leftrightarrow \forall\theta \left(\forall F'\in\Gamma \ (\theta(F')\neq 1) \vee \theta(F)\neq 0 \right)$$

де Γ - множина формул; F та F' формули; v та v' -логічні значення; θ - оцінка. Варто зауважити, що $\Gamma \vDash_2 F$ і $\Gamma \vDash_3 F$, тоді і тільки тоді, коли $\Gamma \vDash_I F$, а також що $\Gamma \vDash_2 F$ або $\Gamma \vDash_3 F$, тягне за собою $\Gamma \vDash_4 F$.

Далі, у відповідності до кожного з означень логічного наслідку ми можемо визначити справджуваність і суперечність формули.

Означення I. Оцінка θ справджує формулу F в розумінні $\vDash_I$ ($\vDash_2$), якщо $\theta(F) = 1$. Оцінка θ справджує формулу F в розумінні $\vDash_3$ ($\vDash_4$), якщо $\theta(F) \in \{u,1\}$.

Таким чином, виявляється, що поняття справджуваності формули залежить від поняття логічного наслідку, тому надалі ми будемо казати, що "формула справджується в розумінні $\models_1$" ($\models_2$, $\models_3$, $\models_4$). Можливі різні означення суперечливості, а саме:

Означення 2. Суперечливою зветься множина формул, яка не справджується жодною оцінкою.

Означення 2′. Суперечливою зветься множина формул, з якої логічно випливає формула, значення довільної оцінки на якій дорівнює 0.

Можна довести, що у випадку $\models_2$ та $\models_3$ ці означення суперечливості "рівнооб'ємні", тобто формула є суперечливою у відповідності з одним з цих означень тоді і тільки тоді, коли вона суперечлива у відповідності з іншим. Для $\models_1$ та $\models_4$ ця властивість не виконується, тому в решті другого розділу розглядаються лише $\models_2$ та $\models_3$.

Як відомо, для методу резолюцій істотною є зводимість питання, чи є формула логічним наслідком множини формул до питання про суперечливість множини формул. Така зводимість існує лише для $\models_2$ та $\models_3$.

Теорема. $\Gamma \models_2 F$ тоді і тільки тоді, коли $\Gamma \cup \{\neg\neg F\}$ суперечливе в розумінні $\models_2$.

Теорема. $\Gamma \models_3 F$ тоді і тільки тоді, коли $\Gamma \cup \{\neg F\}$ суперечливе в розумінні $\models_3$.

Тепер перейдемо до визначення літер та диз'юнктів. Легко переконатися, що в SSL_0 існує шість різних (в розумінні таблиць істинності) комбінацій заперечень, включаючи і відсутність заперечення. Якщо задано формулу F, то з використанням цих комбінацій можна побудувати такі шість формул:

$$F,\ {\sim}F,\ \neg F,\ \neg{\sim}F,\ \neg\neg F,\ \neg\neg{\sim}F$$

Таким чином, ми визначаємо літеру як атом, якому передує одна з указаних комбінацій заперечень. Диз'юнкт визначається як множина літер.

Легко довести, що формула ${\sim}F$ справджується в розумінні $\models_2$ тоді і тільки тоді, коли справджується (в розумінні $\models_2$) формула

$\neg F$. Можна також довести, що формула F справджується в розумінні $\models_2$ тоді і тільки тоді, коли справджується формула $\neg\sim F$. Тому достатньо обмежитися розгляд м чотирьох типів літер, а саме:

$$P(\ldots),\ \sim P(\ldots),\ \neg\neg P(\ldots),\ \neg\neg\sim P(\ldots)$$

Аналогічно, якщо розглядати справджуваність і суперечність у розумінні $\models_3$, достатньо розглянути такі чотири типи літер:

$$P(\ldots),\ \sim P(\ldots),\ \neg P(\ldots),\ \neg\sim P(\ldots)$$

Як і в класичному випадку, контрарними звуться такі літери, які не справджуються одночасно жодною оцінкою. Тоді ми можемо зформулювати два правила резолюцій (для $\models_2$ і $\models_3$ відповідно). Відповідні формулювання збігаються з наведеними вище для ДСМ-логіки.

Головним результатом другого розділу є така

Теорема. Множина диз'юнктів суперечлива в розумінні $\models_2$ ($\models_3$) тоді і тільки тоді, коли з неї за допомогою правила резолюцій можна вивести порожній диз'юнкт. Доведення цієї теореми провадиться за допомогою семантичних дерев.

В заключній частині другого розділу розглядаються поширення понять факторизації та поглинання для SSL_0. Як і для ДСМ-логіки, вдається відмовитися від вимоги про повний збіг знаків склеюваних літер.

У третьому розділі розглядається комплекс програм, що реалізують процес побудови виведення для ДСМ-логіки та SSL_0. Ці програми було реалізовано на ПЕОМ IBM/PC під керуванням операційної системи MS DOS. Докладно описано алгоритми та структури данних, які застосовано в цих програмах.

В програмах реалізовано деякі цікаві можливості. Так, в них існують засоби для роботи з рівністю, а саме переписування термів. Поряд з цим в програмі для ДСМ-логіки реалізовано правило парамодуляції. Переписування термів може використовуватись не лише для моделювання рівності, а й для обчислень над абстрактними термами. Наприклад, два таких правила реалізують конкатенацію списків:

```
concat([X,L],L1) -> [X,concat(L,L1)]
concat(nil,L1) -> L1
```

У програмі для ДСМ-логіки переписування термів використовується для реалізації операцій над множинами.

У програмі для SSL_0, у порівнянні з програмою для ДСМ-логіки, наявні додаткові особливості, що поширюють її можливості:

1. Гнучкий синтаксис, який дозволяє використовувати префіксну, інфіксну та міксфіксну форми запису.
2. Можливість роботи з багатосортними сигнатурами, на сортах яких задано частковий порядок. (Відповідно, реалізовано багатосортні варіанти таких алгоритмів як уніфікація та порівняння із зразком).
3. Дозволяється описувати функціональні символи як асоціативно-комутативні. Тоді уніфікація та порівняння з зразком будуть виконуватися по модулю асоціативності-комутативності, що не потребує явного виписування відповідних аксіом.
4. Реалізовано загальновживані сорти (типи даних): цілі, символи, послідовності символів. Константи цих сортів зберігаються у машинному вигляді, а не у вигляді абстрактних термів, що істотно економить пам'ять. При уніфікації (порівнянні із зразком) дозволяється зіставляти константу одного з цих сортів з абстрактним термом цього сорту.
5. Користувач може написати свої власні програми на процедурній мові програмування для виконання операцій над константами сортів, які розглядаються у п. 4. Наприклад, можна написати програми, що реалізують цілочисельну арифметику.

Слід також зауважити, що в обох програмах є цілий ряд параметрів, які дозволяють користувачу змінювати їх роботу. Наприклад, можна задати максимальну кількість літер в диз'юнкті - диз'юнкти з більшою кількістю літер будуть відкидатися.

Автором було проведено чимало експериментів з обома програмами. В дисертації докладно розглянуто як відповідні задачі, так і результати, що було отримано при їх розв'язанні. Відповідні протоколи наведено у додатках.

Заключну частину третього розділу присвячено інтерфейсу користувача. Розглядається інтегроване оточення LWB (Logician's Workbench) [7], куди входять ряд програм доведення теорем для класичної логіки та вищезгадані програми. Головною метою розробки LWB було створення комфортного оточення для користувача. Імена файлів з даними, конкретний тип програми для доведення теорем та її параметри можуть задаватися шляхом вибору пунктів з меню, а відповідна інформація демонструється у вікнах (або друкується на друкуючому пристрої).

З іншого боку, програми можуть працювати і автономно, а також в складі інших програмних систем. В останньому випадку може видатися корисним вимкнути вивід інформації на екран, щоб зробити роботу програми "прихованою".

Список літератури

1. Комплекс программ для экспериментов с выводами резолюционного типа / А.В.Брановицкий, К.П.Вершинин, Д.М.Райко, И.Б.Романенко, А.А.Сахно // Тез. докл. II Всесоюз. конф. по прикладной логике, 7-9 июня 1988 г. - Новосибирск, 1988. - С. 25-26.

2. Вершинин К.П., Романенко И.Б. Программа построения вывода для ДСМ-логики // II Всесоюз. конф. "Искусственный интеллект-90". - Минск, 1990. - С. 160-161.

3. Вершинин К.П., Романенко И.Б. Распространение метода резолюций на некоторые недвузначные логики // Семиотика и информатика. - 1990. - Вып. 31. - С. 104-123

4. Вершинин К.П., Райко Д.М., Романенко И.Б. Комплекс средств анализа экспертных систем // Выставка программных средств. - Киев: Коммерческий центр Института кибернетики им. В.М.Глушкова АН Украины, 1990. - ч. 1. - С. 15

5. Rajko D., Vershinin K., Romanenko I. Algebraic Program Interpreter APREX2 // Lect. Notes Comput. Sci. - 1991. - 480. - P. 547-548.

6. Vershinin K.P., Romanenko I.B. One more logic with uncertainty and resolution principle for it // Lect. Notes Comput. Sci. - 1992. - 607. - P. 663-667

7. Romanenko I.B. Logician's Workbench // Lect. Notes in Artificial Intelligence. - 1992. - 624. - P. 499-500.

ПІдп. до друк. 23.02.93. Формат 60x84/16. ПапІр друк. № 2.
Офс. друк. Ум. друк. арк. 0,93. Ум.фарбо-вІдб. І,05.
Обл.-вид. арк. І.0. Тираж І00 прим. Зам. 522.

РедакцІйно-видавничий вІддІл з полІграфІчною дІльницею
Інституту кІбернетики ІменІ В.М.Глушкова АН УкраІни
252207 КиІв 207, проспект АкадемІка Глушкова, 40

[illegible]