

# Дослідження Методів Мінімізації Алгебро-логічних Рівнянь

Денис Білозьоров  
кафедра Програмної Інженерії  
Харківський національний університет  
радіоелектроніки  
Харків, Україна  
riit.kharkiv.ua@gmail.com

Ігор Шубін  
кандидат технічних наук  
кафедра Програмної Інженерії  
Харківський національний університет  
радіоелектроніки  
Харків, Україна  
igor.shubin@nure.ua

## Research of Methods Minimizing of Algebraic-logical Equations

Denis Bilozorov  
Department of Software Engineering  
Kharkiv National University  
of Radio Electronics  
Kharkiv, Ukraine  
riit.kharkiv.ua@gmail.com

Igor Shubin  
PhD, associate professor  
Department of Software Engineering  
Kharkiv National University of Radio Electronics  
Kharkiv, Ukraine  
igor.shubin@nure.ua

**Анотація**—розроблено метод та модель навчання з використанням теорії штучного інтелекту, для моделювання процесів отримання знань обґрунтовано використання алгебри предикатів та предикатних операцій. Адаптивна організація навчального матеріалу описана в термінах алгебри скінченних предикатів та предикатних операцій, що дозволяє автоматизувати вирішення завдання створення навчальних програм з елементами штучного інтелекту

**Abstract**—It has been justified use algebra predicates and predicate operations. It has also been developed a general multilevel model of adaptive learning material, which was described in terms of predicate algebra operations that allows automating the solution of training programs with elements of artificial intelligence.

**Ключові слова**—адаптивна гіпермедіа, алгебра скінченних предикатів, навігаційне правило, мінімізація скінченних предикатів

**Keywords**—adaptive hypermedia, finite algebra predicates, navigation rule, minimization of finite predicates

### I. ВСТУП

Алгебра кінцевих предикатів и розроблені на її основі алгоритмічні й програмні рішення дозволяють вирішити

більшість із зазначених вище протиріч. Математичний апарат алгебри скінченних предикатів також можна використати на етапі поповнення словника новими поняттями та доповнення словарних статей існуючих понять новими визначеннями [1].

Лінгвістичне забезпечення автоматизованої гіпермедійної навчальної системи є важливим інструментом, який надає користувачу доступ до інформаційних ресурсів. Однією зі складових комплексного лінгвістичного забезпечення сучасної автоматизованої системи є лінгвістичне забезпечення процесів класифікації та систематизації документів, базою для розробки якого є інформаційно-пошукові тезауруси. Масив ключових слів такого тезауруса формується на основі концептуальної моделі певної предметної галузі. Ця модель може бути представлена електронним галузевим термінологічним словником [2].

З розвитком та поглибленням інформаційної бази даної області математики з'явилась проблема складності сприйняття матеріалу для студентів та зацікавлених. Метою роботи є розробка інформаційно – навчальної системи для вивчення матеріалу з алгебри скінченних предикатів та основних методів мінімізації бульових функцій.



## II. АЛГЕБРО-ЛОГІЧНІ РІВНЯННЯ

Наразі функціонування автоматизованих систем неможливо без організації великих за обсягом інформаційних банків. До складу інформації входить велика кількість неформалізованих даних, що представляють собою різні текстові повідомлення природною мовою для систем управління, це інформація про об'єкт управління, пов'язана з його структурою, функціонуванням і прийняттям рішень щодо стратегії управління. Сьогодні не виникає сумнівів, що обмін інформацією між користувачем і ЕОМ повинен проводитися на природній або хоча б близькій до природної мовою. У такому випадку машина повинна містити деякий набір знань, відповідний до нормативної граматики, і подібно людині успішно оперувати ним. Таким чином, необхідно створення процесора, який дозволить комп'ютеру сприймати природно-мовний запит і формувати відповідь. Тоді людині, яка працює з машиною, не знадобиться тривалих і серйозних занять з оволодіння мистецтвом програмування і вивчення специфіки ЕОМ.

Проблемам накопичення знань про природу мову, її устрій, принципи функціонування, вибір мови формального опису присвячено багато робіт. Успішне вирішення лінгвістичних проблем машинного перекладу, різного роду завдань по автоматичній обробці текстової інформації та мовного управління виробничими процесами, а також лінгвістичних проблем спілкування з ЕОМ обіцяє в майбутньому величезну економію сил і засобів [3].

Створення сучасних технічних засобів для автоматичної обробки мовних та фонетичних закономірностей вимагає розробки нових методів синтезу обчислювальних структур. Застосування таких способів дозволить на одному і тому ж пристрої одночасно вирішувати цілий ряд задач обробки та моделювання лінгвістичних завдань. Створювані для подібних систем алгоритми і програмні засоби повинні мати властивості регенерації та часткової регенерації вхідних і вихідних даних, бути надійними в експлуатації, володіти можливістю корекції в процесі обігу. Проблемам накопичення знань про природу мову, її устрій, принципи функціонування, вибір мови формального опису присвячено багато робіт. Успішне вирішення лінгвістичних проблем машинного перекладу, різного роду завдань по автоматичній обробці текстової інформації.

Однією з найважливіших проблем синтезу схем, пристроїв автоматики, вимірювальної та обчислювальної техніки, кібернетики є отримання найбільш простих (мінімальних) формул, що описують їх функціонування. При цьому проблема мінімізації є одним з основних етапів аналізу і синтезу схем автоматів, створення програмного забезпечення обчислювальних систем. Дане завдання, як показує науково-технічна література, їй присвячена, не вирішене в достатній мірі.

Особливу увагу, в першу чергу, слід зосередити на задачі мінімізації форм представлення багатозначних

логічних функцій, що грають важливу роль в проблемі синтезу дискретних автоматів [4]. У зв'язку з актуальністю проблеми мінімізації багатозначних перемикальних функцій з'явилися публікації, в яких робляться спроби узагальнити відомі методи та алгоритми знаходження мінімальних форм представлення булевих функцій на багатозначний випадок. Але складність подачі інформації в даних публікаціях призводить до потреби створення навчального інтерфейсу, орієнтованого на широке коло зацікавлених, заснований на простоті і доступності представленого матеріалу.

## III. ОСНОВНІ КОМПОНЕНТИ ДЛЯ ВИРІШЕННЯ ЗАДАЧ

Побудова гіпермедійного простору інтелектуальної системи потребує опису зв'язків та вузлів за допомогою апарату алгебри скінченних предикатів, при цьому основним критерієм оптимальності є кількість зв'язків між вузлами. Це відповідає завданню мінімізації площі відповідної характеристичної матриці, тобто добутку числа рядків на число стовпців даної матриці. Число певних елементів усередині матриці (елементів впізнання) прийом як додаткового критерію оптимізації. Таким чином, будемо будувати найкоротші ДНФ скінченних предикатів і одночасно враховувати вимоги мінімальності числа впізнань [5].

Іншими словами, будемо шукати мінімальне покриття одиничної області  $N_f$  скінченного предиката  $f$  максимальними інтервалами. Тому одинична область  $N_f$  подана у вигляді характеристичної матриці. У цьому випадку завдання мінімізації ДНФ скінченного предиката можна інтерпретувати як задачу максимального стиснення матриці, при якому деякі сукупності рядків цієї матриці, що утворюють інтервали гіперкуба  $M^n$ , замінюються відповідними слабо визначеними векторами. Якщо деяка точка  $u$  одиничної області  $N_f$  належить лише одному з максимальних інтервалів цієї множини, а саме – деякому інтервалу  $u$ , то будь-яке найкоротше покриття області  $N_f$  буде містити інтервал  $u$ . Будемо називати елемент  $A$  визначальним, а інтервал  $u$ , що однозначно ним визначається – обов'язковим [6].

Щоб встановити чи є деяка точка  $A$  визначальною, досить знайти всі сусідні з нею точки з області  $N_f$ , а потім побудувати мінімальний інтервал гіперкуба  $M^n$ , що їх містить, званий мінімальним поглинальним інтервалом. Якщо всі точки цього інтервалу належать до одиничної області, то цей інтервал є інтервалом області  $N_f$ , максимальним і обов'язковим, а елемент  $A$  є визначальним. Мінімальний поглинальний інтервал для заданої множини точок простору  $M^n$  отримують у такий спосіб. Будується вектор, компоненти якого приймають значення однойменних компонент координат точок з  $M^n$ , якщо ці значення збігаються, і приймають значення «-», якщо серед значень однойменних компонент координат точок зустрічаються різні літери з алфавіту. Якщо інтервал  $u$  містить багато елементів, то перевірка питання, чи міститься інтервал  $u$  в області  $N_f$ , зводиться до наступного.

З характеристичної матриці виділяється вектор, утворений перерахуванням рядків, сусідніх вектору  $u$  за



допомогою стовпців, в яких вектор  $u$  має значення «-», а потім виділяється перевірка мінору на вродженість (мінор назвемо виродженим, якщо він містить рядки, елементи яких представляють усі можливі комбінації з букв вихідного алфавіту). Якщо мінор вироджений, то інтервал  $u$  міститься в  $N_f$ .

Обчислення обов'язкових інтервалів прискорюється при малій кількості сусідів розглянутої точки. Якщо точка не має сусідів, то вона сама утворює обов'язковий інтервал. Якщо точка має двох сусідів, один з яких відрізняється від  $A$  за компонентою  $i$ , а інший за компонентою  $j$ , то обчислення зведуться до перевірки наявності в області  $N_f$  елемента, відмінного від  $A$  за двома компонентами  $i$  та  $j$ . Знайшовши в області  $N_f$  принаймні один обов'язковий інтервал, ми починаємо послідовно будувати елементи покриття одиничної області, яке за сприятливих умов дасть точне розв'язання нашої задачі.

Розглянемо поточну ситуацію, коли частина елементів одиничної області  $N_f$  вже покрита деякими інтервалами, а інша частина, що утворює поточну, тобто змінювану в процесі розв'язання множину  $N^*$ , складається з елементів одиничної області, які ще не належать жодному з вибраних інтервалів. Нехай деякий елемент  $C$  поточної множини  $N^*$  належить одразу декільком максимальним інтервалам області  $N_f$ . Розглянемо перетини цих інтервалів з множиною  $N^*$ , називаючи їх надалі проекціями інтервалів на множину  $N^*$ . Якщо серед цих проекцій знайдеться така, яка містить в ролі своїх підмножин інші, то можна вважати обґрунтованим вибір будь-якого відповідного їй максимального інтервалу (їх може виявитися декілька) як чергового елемента найкоротшого покриття. Цей інтервал забезпечує покриття всіх елементів з  $N^*$ , які тільки можна покрити інтервалами множини  $N_f$ , які покривають одночасно елемент  $C$ . Такий інтервал назвемо допустимим, відзначимо при цьому, що даний інтервал виявляється допустимим лише при даній послідовності попередніх виборів інтервалу покриття.

Тепер розглянемо спосіб знаходження визначальних елементів в проміжних ситуаціях. Нехай деякі два елементи одиничної області  $N_f$  несумісні, якщо поглинальний мінімальний інтервал міститься в  $N_f$ . Елемент  $C$  виявляється визначальним, якщо поглинає інтервал для всіх елементів множини  $N^*$ , сумісних з елементом  $C$ , міститься в області  $N_f$  [7].

Вибір у множині  $N^*$ , виконуваний під час пошуку сумісних з  $C$  елементів, прискорюється тим, що він обмежує розгляд елементів, що належать до мінімального поглинального інтервалу для всіх сусідів елемента  $C$ , як сумісні з  $C$  елементи можуть перебувати тільки в цьому інтервалі. Обидва мінімальних поглинальних інтервали як для сусідів елемента  $C$ , так і для сумісних з ним елементів будуються легко. Нарешті, вищенаведеним способом встановлюється, чи міститься останній з цих інтервалів одиничної області  $N_f$ . Позитивна відповідь на це питання доводить, що елемент  $C$  є визначальним. Допустимим інтервалом в цьому випадку виявляється будь-який мінімальний інтервал області  $N_f$ , що містить знайдений

мінімальний поглинальний інтервал для сумісних з  $C$  елементів. Зокрема, ці два інтервали можуть співпасти.

Проміжні інтервали суттєво відрізняються від початкового тим, що в початковій ситуації від визначального елемента  $C$  потрібно, щоб в одиничній області  $N_f$  містився мінімальний поглинальний інтервал для всіх сумісних з  $C$  елементів області  $N^*$ , в той час як у поточній ситуації ця вимога послаблена: там розглядаються мінімальні поглинальні інтервали тільки для тих елементів, сумісних з визначальним елементом, які належать до поточної множини  $N^*$ .

Ця відмінність визначає ланцюговий характер процесу побудови найкоротшого інтервального покриття одиничної області  $N_f$ . Включення в покриття чергового допустимого інтервалу супроводжується відповідним скороченням множини  $N^*$ , а це в свою чергу призводить до того, що деякі з решти елементів в  $N^*$  можуть стати визначальними, що дозволить включити в розв'язання нові допустимі інтервали, далі скоротити поточну множину  $N^*$  і т.д. У розглянутому способі конструювання найкоротшого інтервального покриття одиничної області  $N_f$  можуть зустрітися критичні ситуації, коли в поточній множині  $N^*$  не вдається знайти жодного визначального елемента.

У такому випадку для отримання точного розв'язання потрібно виконати перебір різних варіантів побудови покриття, що відповідають вибору різних максимальних інтервалів як чергового елемента покриття. Необхідно вжити заходів щодо скорочення цього перебору, застосовувати розгалужені процеси. Замість допустимого інтервалу, в описаній критичній ситуації вибираємо деяку сукупність інтервалів, тобто множину максимальних інтервалів, щодо якої відомо, що в ній міститься, принаймні, один з допустимих в поточній ситуації інтервалів, хоча і невідомо, який саме. При цьому поняття допустимого інтервалу вживається в широкому сенсі. Назвемо допустимою такою сукупність інтервалів одиничної області  $N_f$ , якщо існує деяке найкоротше інтервальне покриття області  $N_f$ , що її містить. Якщо сформована на поточний момент часу частина покриття, що конструюється є допустимою, то деякий максимальний інтервал буде допустимим в даному випадку, якщо, прийнявши його як черговий елемент покриття, ми отримаємо знову допустиму сукупність. Розгляд різних варіантів вибору чергового елемента покриття з достатньої сукупності утворює точку розгалуження процесу пошуку найкоротшого інтервального покриття. Число вихідних з цієї точки гілок дорівнює числу елементів у даній достатній сукупності.

Скорочення перебору зводиться в зв'язку з цим до знаходження в поточній ситуації деякої мінімальної достатньої сукупності. Наприклад, достатньою є сукупність всіх максимальних інтервалів, яким належить певний елемент  $C$  області  $N_f$ . Достатньою в поточній ситуації виявляється сукупність максимальних інтервалів, до яких належить певний елемент  $C$  множини  $N^*$  і деякі мають різні, що не містяться оди в одній проекціях на множині  $N^*$ . Число елементів в цих допустимих сумах, як



правило, менше, якщо у елемента  $C$ , що їх задає мало сусідів.

Можна зробити висновок, що для скорочення перебору під час пошуку оптимального розв'язання даної задачі: попередньо зважити всі елементи  $N_f$ , визначаючи його вагу як число сусідів, а наступні перебори в області  $N_f$  організувати в порядку не спадання цих ваг.

Даний метод породжує наближені методи, в яких не обов'язково розглядати всі варіанти, пов'язані з вибором різних елементів достатніх сукупностей. Наприклад, в кожній точці розгалуження може вибиратися один шлях, «кращий» за певним критерієм. Також може бути спрощено сам спосіб утворення точок розгалуження, в результаті чого задача знаходження мінімальної достатньої сукупності може вирішуватися також наближено або навіть не ставиться.

В описаному способі аналізу характеристичної матриці області  $N_f$  скінченного предиката  $f$ , що мінімізується необхідно розглядати велику кількість різних сполучень елементів у множині  $N_f$ . Наприклад, відшукуючи в області  $N_f$  сусідів, ми повинні перевірити всі пари елементів цієї множини. Цей перебір можна значно скоротити, задаючи розглянутий скінченний предикат  $f$  безпосередньо на гіперкубі  $M^n$ , елементи якого впорядковані і певним чином пронумеровані. Якщо в деякій вершині гіперкуба  $M^n$  скінченний предикат  $f$  приймає значення  $I$ , тобто якщо ця вершина належить до одиничної області  $N_f$ , то, відшукуючи його сусідів в області  $N_f$ , не обов'язково перебирати всі елементи цієї області і порівнювати їх з даними, а досить переглянути лише сусідні елементи гіперкуба  $M^n$  (їх число дорівнює  $n$ ) і вибрати з них ті, на яких предикат  $f$  приймає значення  $I$ . Таким чином, суцільний перебір елементів в характеристичній матриці, відповідної області  $N_f$ , замінюється впорядкованим вибором і аналізом елементів гіперкуба  $M^n$ . Такий спосіб назовемо скануванням.

Розглянемо метод мінімізації ДНФ скінченного предиката, що базується на виборі в одиничній області елементів в порядку зростання числа сусідів. Скінченний предикат, що мінімізується видається безпосередньо в табличній формі, де в лівій колонці перераховано всі елементи гіперкуба  $M^n$  – набори значень аргументів, впорядковано відповідно до позиційного  $k$ -го коду, де  $k$  – число букв алфавіту  $A$ , а праворуч значення на наборах скінченного предиката.

Оскільки послідовність наборів чітко впорядкована, тобто кожен набір однозначно визначено місцем, зайнятим у таблиці, то їх можна не показувати, що і робиться при поданні функції  $f$ , що мінімізується в пам'яті комп'ютера, коли набір ототожнюється з адресою, куди поміщається відповідне значення скінченного предиката  $f$ . Значення скінченного предиката на сусідніх елементах гіперкуба отримуються зміною значення тієї компоненти адреси, за якою відшукуються сусідні елементи. Чим менше сусідів в області  $N_f$  має певний елемент цієї множини, тим меншому, як правило, числу різних

максимальних інтервалів він належить. З огляду на це, описуваний метод починає побудову інтервального покриття множини  $N_f$  з пошуку інтервалів для покриття елементів з мінімальним числом сусідів – в цьому випадку конструювання покриття буде цілеспрямованим.

Першим етапом є побудова таблиці сусідів, в якій для кожного елемента характеристичної матриці покано, за якими змінними цей елемент має сусідів в одиничній області  $N_f$ . Цю таблицю сумісний з таблицею значень скінченного предиката  $f$ , заповнюючи прочерком відповідні позиції для елементів нульової області. Розгляд починаємо з елементів, які не мають сусідів; якщо таких не виявлено, то вибираємо елементи з одним сусідом. Знаходимо для цих елементів обов'язкові інтервали і далі проводимо всі операції, описані вище в алгоритмі.

#### IV. ВИСНОВКИ

В результаті проведеного дослідження порівняння інформаційно-початкової системи з існуючими аналогами був виділений ряд переваг реалізованого додатка перед схожими програмними рішеннями: досить висока швидкість на середніх апаратних конфігураціях; максимальна простота користування; інтуїтивність та зрозумілість інтерфейсу; безвартна робота з великими за об'ємом даними; можливість додавання нових навчальних тем; мобільність додатку.

Для подальшого покращення системи зроблені такі вдосконалення: посилення інформаційно – навчальної бази системи; підвищення рівня складності навчальної системи шляхом інтегрування тестуючого та ігрового навчального блоку; можливість роздрукування матеріалу лекцій за допомогою відповідних елементів інтерфейсу; вдосконалення презентативної здатності інтерфейсу через застосування нових елементів графічного відображення.

#### ЛІТЕРАТУРА REFERENCES

- [1] Bondarenko M (2007). Teoriya Intellekta / M.F. Bondarenko, Yu.P. Shabanov-Kushnarenko. Kharkiv.: «СМІТ», 2007. – 221 p.
- [2] Bondarenko M (2005). Algebra Predikatov i Predikatnykh Operacij / M. Bondarenko, Z. Dudar, Yu.P. Shabanov-Kushnarenko // Radioelectronics & Informatics. – Kharkiv: KhNURE, 2005. – № 1. – pp. 80–87
- [3] Tolkiehn G., Makariti A.(2015) MyWebRTC, a Free Do-It-Yourself Kit for Secure Real-Time Internet-Communication //Information Technologies in Information Business Conference (ITIB 2015) – Kharkiv, 2015. – pp 81-84.
- [4] Kuzikov B. Programme Zabezpechennya Dlya Pobudovy Individualnih Navchalnyh Traektorij / B.O.Kuzikov // Informatika, Matematika, Avtomatika : Materialy ta programma naukovo-tekhnichnoi konferencii, 2012, Sumy. –: SumDU , 2012. –pp. 54-55.
- [5] Shubin I., Karmanenko O., Umyarov K . The Methods of Adaptation in Computer-Based Training Systems //Information Technologies in Information Business Conference (ITIB 2015) – Kharkiv, 2015. – S 64-67
- [6] Shubin I., Kirichenko I. (2016) Informatsionnye Technologii Modelirovaniya Adaptivnyh System Obucheniya // Matherials of The 5Th Mezhdunarodnoj nauchno-tehnicheskoy konferencii «Informatsionnye sistemy i tehnologii ICT-2016», Kharkiv, 2016 – S. 244-245.
- [7] Shubin I., Kirichenko I. Modeli Intelktualnoy Podderzhki Navigacii v Computer Ttraining Systems / International Journal “Information Models and Analyses” Vol. 2/ 2013, Number 2. – Bulgaria – 2013 – pp. 194-199

