УДК 519.714

Глибовець М. М., Іващенко С. А.

ПРО ОДИН ПІДХІД ДО РОЗВ'ЯЗКУ ЗАДАЧІ МІНІМІЗАЦІЇ БУЛЕВИХ ФУНКЦІЙ

Розроблено метод визначення нижньої межі часової і об'ємної складності задачі. Доводиться неможливість існування поліноміального алгоритму розв'язання задачі мінімізації булевих функцій. Викладено алгоритм розв'язання задачі з фіксованою часовою складністю.

Вступ

Задача мінімізації булевих функцій (МБФ) є однією з найвідоміших і в той же час найважливіших проблем теорії алгоритмів, що має широке застосування на практиці. Проте основною проблемою задачі МБФ є значна часова складність її розв'язання. Це змушує дослідників знову і знову шукати нові алгоритми або вдосконалювати старі.

Алгоритми мінімізації булевих функцій

Для постановки задачі МБФ розглянемо деякі прийняті в науці терміни.

Елементарною кон'юнкцією (ЕК) називають логічне множення булевих змінних в деяких степенях $K = x_{i_1}^{\sigma_1} \dots x_{i_r}^{\sigma_r}$, в якому всі змінні різні. Число r називають рангом кон'юнкції. У випадку, коли $r = 0$ кон'юнкцію називають порожньою і вважають, що вона дорівнює 1.

Диз'юнктивною нормальною формою (ДНФ) називають диз'юнкцію $D = K_1 \vee K_2 \vee \dots \vee K_m$ елементарних кон'юнкцій K_j, в якій всі K_j — різні. Число m називають *довжиною ДНФ*. Коли $m = 0$, ДНФ називають порожньою і вважають, що вона дорівнює 0.

Конституентою одиниці називають ЕК, до якої входять всі змінні (з запереченнями або без них).

Досконалою диз'юнктивною нормальною формою (ДДНФ) називають таку ДНФ, всі ЕК якої є конституентами одиниці.

Відомо, що будь-яка функція алгебри логіки f: $X \times X \times \dots \times X \to Y$, $X = \{0,1\}$, $Y = \{0,1\}$, відмінна від 0, може бути представлена у вигляді досконалої диз'юнктивної нормальної форми [1].

$$f(x_1,\dots,x_n) = \bigvee_{\substack{(\alpha_1,\dots,\alpha_n) \\ f(x_1,\dots,x_n)}} x_1^{\alpha_1} \dots x_n^{\alpha_n}, \text{ де } x^{\alpha} = \begin{cases} x, \alpha = \\ \bar{x}, \alpha = \end{cases}$$

Проте відомо, що ДДНФ можна спростити, так що отримаємо формулу, яка також реалізує відображення f, але має меншу кількість символів.

Домовимося надалі розглядати методи заданих функцій булевої алгебри мінімальними формулами в класі диз'юнктивних нормальних форм.

Внаслідок того, що булева функція може бути представлена не тільки у вигляді ДНФ, то маємо можливість вибору найзручнішої її реалізації. Для вибору потрібного критерію зручності вводять індекс простоти L(D), що характеризує "складність" ДНФ. Для функціоналу L(D) вимагається виконання наступних аксіом:

I. Аксіома невід'ємності. Для будь-якої ДНФ D $L(D) \geq 0$.

II. Аксіома монотонності (відносно множення). Нехай $D = D' \vee x_i^{\sigma_i} K'$. Тоді $L(D) \geq L(D' \vee K')$.

III. Аксіома опуклості (відносно додавання). Нехай ДНФ D розбита в пряму суму ДНФ D_1 і D_2, тобто $D = D_1 \vee D_2$ і D_1, D_2 не мають спільних членів. Тоді $L(D) \geq L(D_1) + L(D_2)$.

IV. Аксіома інваріантності (відносно ізоморфізму). Нехай ДНФ D′ отримана з ДНФ D шляхом перейменування змінних (без ототожнень). Тоді $L(D') = L(D)$ [2].

Зустрічаються такі індекси простоти ДНФ:

1. $L_Б(D)$ — кількість букв змінних, які зустрічаються в записі ДНФ D.
2. $L_К(D)$ — кількість ЕК, що входять до складу D.
3. $L_З(D)$ — кількість заперечень змінних, які зустрічаються в записі D.

Найбільшу цінність з точки зору дослідника для задачі МБФ становлять індекси $L_Б$ і $L_К$.

Означення 1. *Мінімальною ДНФ* функції $f(x_1, \ldots, x_n)$ називають ДНФ, що реалізує $f(x_1, \ldots, x_n)$ і має найменший індекс простоти $L_Б(D)$ з усіх ДНФ, що реалізують $f(x_1, \ldots, x_n)$.

Означення 2. *Найкоротшою ДНФ* функції $f(x_1, \ldots, x_n)$ називають ДНФ, що реалізує $f(x_1, \ldots, x_n)$ і має найменший індекс простоти $L_К(D)$ з усіх ДНФ, що реалізують $f(x_1, \ldots, x_n)$.

В даній статті ми скоротимо задачу МБФ до знаходження мінімальної ДНФ для заданої функції.

Розв'язок задачі мінімізації булевих функцій

Існує тривіальний алгоритм побудови мінімальної (найкоротшої) ДНФ для будь-якої функції алгебри логіки $f(x_1, \ldots, x_n)$. Всі ДНФ, складені з елементів $x_1, \ldots, x_n, \bar{x}_1, \ldots, \bar{x}_n$, впорядковані за кількістю літер і по порядку для кожної ДНФ D перевіряють рівність $D = f(x_1, \ldots, x_n)$. Перша ж ДНФ, для якої виконується рівність, очевидно, є мінімальною ДНФ для функції $f(x_1, \ldots, x_n)$.

Теорема 1. Кількість різних ДНФ, сформованих із змінних $x_1, \ldots, x_n$, дорівнює 2^{3^n}.

Доведення. Кількість різних ЕК, складених із змінних $x_1, \ldots, x_n$, дорівнює 3^n. Дійсно, кожна з n змінних або не входить в ЕК, або входить в неї з запереченням, або входить в неї без заперечення. Очевидно, що кількість різних ДНФ, складених із змінних $x_1, \ldots, x_n$, дорівнює кількості всіх можливих підмножин в множині з 3^n різних елементів, тобто 2^{3^n}.

Теорема доведена.

Оскільки для реалізації описаного вище алгоритму треба перебрати всі 2^{3^n} об'єктів, то зрозуміло, що даний алгоритм практично незастосовний.

Існують дві групи методів, які покращують ефективність алгоритму синтезу мінімальної ДНФ. До першої групи належать методи, в яких з множини всіх ЕК певним нескладним за часом способом видаляються кон'юнкції, які не можуть входити до множини мінімальних ДНФ. Це призводить до скорочення потужності множини ДНФ, з якої здійснюється перебір. До другої групи належать методи, пов'язані з більш економним перебором ДНФ із утвореної множини ДНФ [3]. Однак, незважаючи на досить велику кількість підходів до розв'язання, задача все ще не розв'язана. Тому необхідне більш детальне вивчення задачі з метою виявлення меж ефективної часової та об'ємної складності.

Авторський підхід до проблеми

У ході розв'язку задачі пошуку мінімальної ДНФ отримуємо суттєву проблему, сформульовану наступною теоремою.

Теорема 2. Не існує алгоритму побудови мінімальної ДНФ з поліноміальною об'ємною складністю.

Доведення. Насамперед слід зазначити такий факт: оскільки кількість ЕК досягає 2^n (n — арність булевої функції), то можна було б стверджувати, виходячи з досвіду роботи з обчислювальними системами, що будь-який алгоритм, призначений для розв'язання даної задачі, так чи інакше буде обробляти числа довжини, не меншої 2^n, якщо він буде призначений для обробки всіх варіантів булевих функцій. Проте теоретично можна припустити існування деякої невідомої до цього часу форми задання булевих функцій, для якої не буде важливою їх об'ємна складність. У такому разі нам слід довести, що задання вихідної ДНФ не може бути поліноміальним.

Розглянемо найпростіший варіант. Нехай на виході маємо послідовність символів, що представляють окремі змінні. Кожен символ може бути одним із $2n + 1$ (тобто змінна або її заперечення, або "пустий символ" — відсутня змінна). Якщо не враховувати обмеження, які накладаються правилами побудови ДНФ, можна вважати, що з такої вибірки змінних ми можемо побудувати $(2n + 1)^{nk}$ варіантів вихідної послідовності довжини n^k (тобто поліноміальної довжини). Хоча насправді таких варіантів значно менше, проте навіть при такому підході ми отримуємо кількість варіантів вихідної послідовності символів меншу, ніж кількість можливих мінімізованих ДНФ, яка дорівнює кількості можливих булевих функцій арності $n - 2^{2^n}$. Це видно з наступного співвідношення:

$$(2n + 1)^{nk} \ ? \ 2^{2^n},$$

$$n^k \log_2(2n + 1) \ ? \ 2^n, \ \Rightarrow \ \forall\ k = \mathrm{const},$$

$$k \in N, \ \exists\ \varepsilon > 0, \ \forall\ n > \varepsilon\ (2n + 1)^{nk} < 2^{2^n}.$$

Це означає, що ми не можемо побудувати алгоритм з поліноміальною об'ємною складністю, бо об'єм мінімальної ДНФ, яка утвориться в результаті будь-якого алгоритму, обов'язково стане неполіноміальним для булевих певних функцій.

Теорема доведена.

Продовжимо наші міркування і спробуємо з'ясувати межі довжини мінімізованого булевого виразу. Повернімося до виведеної раніше формули: $(2n + 1)^{nk} \ ? \ 2^{2^n}$, і спробуємо замість поліноміальної об'ємної складності поставити деяку невідому складність α:

$$(2n + 1)^{\alpha} = 2^{2^n}.$$

Звідси отримуємо:

$$\alpha \log_2(2n + 1) = 2^n \Rightarrow \alpha = 2^n / \log_2(2n + 1) \Rightarrow$$

$$\alpha = (2/\sqrt[n]{\log_2(2n+1)}{}^{n}, \text{ отже, при } n \to \infty, \alpha \to 2^n.$$

Отже, об'ємна складність будь-якого алгоритму розв'язання задачі МБФ не може бути меншою $O(2^n)$.

Зауважимо, що цей висновок був зроблений за умови, що ми можемо поставити будь-яку змінну на будь-яке місце, проте насправді ДНФ будується зі значними обмеженнями на це правило, тобто в реальній ситуації можливих варіантів мінімізованих ДНФ буде менше, ніж у нашому припущенні, і число α насправді буде дещо більшим (хоча, не виключено, що лише в межах $k\alpha$, де k — константа).

З другого боку, ми знаємо, що всі булеві функції можуть бути задані у вигляді ДДНФ, а її максимальна довжина може становити не більше $O(n * 2^n)$.

Таким чином, ми отримали верхню і нижню межі теоретичної максимальної довжини МДНФ.

Теорема 3. Не існує алгоритму побудови мінімальної ДНФ з часовою складністю, меншою $O(2^n)$.

Доведення. Нехай маємо деякий алгоритм А, реалізований на детермінованій машині Тюрінга, що розв'язує задачу МБФ. В результаті виконання алгоритму на стрічці МТ може залишитися МДНФ експоненційної довжини відносно кількості змінних в ній. Якщо в ході виконання алгоритму для будь-якої функції вказівник МТ може пройти лише по поліноміальній ділянці запису МДНФ на стрічці, тоді, як видно з наведених вище міркувань, ми не зможемо реалізувати алгоритм А для всіх булевих функцій з кількістю змінних n. А оскільки вказівник мусить пройти принаймні один раз по всій довжині запису результуючої МДНФ, то МТ здійснить також експоненційну кількість кроків виконання алгоритму. Звідси випливає, що часова складність алгоритму також не може бути меншою $O(2^n)$.

Теорема доведена.

Теорема дає можливість нам зробити ще один висновок відносно загальної теорії алгоритмів: дана задача не належить до класу NP-повних задач. Для того, щоб вона до цього класу належала, вона насамперед повинна мати поліноміальний алгоритм розв'язання на недетермінованій машині Тюрінга (НМТ). Проте, якщо вихідний простір даних задачі буде в будь-якому зображенні більший, ніж поліном від розміру простору вхідних даних, то, очевидно, що такого алгоритму не існує, бо навіть для генерації випадкового простору вихідних даних НМТ повинна здійснити неполіноміальну кількість кроків.

Отже, хоч, як видно з усього сказаного вище, для розв'язку даної задачі було запропоновано величезну кількість алгоритмів, всі вони один від одного не дуже відрізняються, тому що всі мають експоненційну часову складність (або й вищу), а саме таку складність ми можемо вважати оптимальною при розв'язанні даної задачі.

В даній роботі реалізуємо алгоритм, що теж має експоненційну часову складність і пояснимо в чому його ефективність.

Алгоритм мінімізації булевих функцій

На вході алгоритму маємо ДДНФ D булевої функції $f(x_1, x_2, x_3, \ldots, x_n)$.

Крок 1. Переведемо ДДНФ D функції $f(x_1, \ldots, x_n)$ в множину бінарних векторів V; кожній координаті α_{ij} вектора K_i $(\alpha_{i1}, \alpha_{i2}, \ldots, \alpha_{in})$ буде відповідати знак (заперечення чи не заперечення) змінної x_j в i-й ЕК ДДНФ, якщо знак заперечення — відповідно, координата 0, якщо ні — 1.

Крок 2. Якщо елементарних кон'юнкцій в D виявилося 2^n — значить функція $f(x_1, \ldots, x_n)$ є константою істини, і ми переходимо на кінець алгоритму, інакше якщо в D немає жодної ЕК, то функція $f(x_1, \ldots, x_n)$ є константою похибки і ми теж переходимо на кінець алгоритму, інакше — продовжуємо.

Крок 3. Починаючи з фіксованої кількості змінних (m), що дорівнює 1, перебираємо всі можливі варіанти ЕК з кількістю змінних, що дорівнює m $K_{e1}(a_1, a_2, \ldots, a_m)$, і для кожного варіанта підраховуємо кількість (v) бінарних векторів K_i ($\alpha_{i1}, \alpha_{i2}, \dots \alpha_{in}$) в множині V, для яких виконується наступна умова (P): для кожної змінної a_k з утвореної ЕК, якщо координата α_{ij} вектора K_i (α_{i1}, $\alpha_{i2}, \dots \alpha_{in}$), що відповідає змінній a_k, дорівнює 1, то і дана змінна a_k знаходиться в $K_{el}(a_1, a_2, \dots, a_m)$ без заперечення, і навпаки, якщо 0 — то змінна a_k знаходиться в $K_{el}(a_1, a_2, \dots, a_m)$ під запереченням. Якщо v дорівнює максимально можливій кількості векторів, що можуть відповідати умові P ($2^{(n-m)}$), то всі вектори, що відповідають умові, переводимо в певну додаткову множину векторів S з множини V, а кон'юнкцію $K_{el}(a_1, a_2, \dots, a_m)$ додаємо до результуючої множини елементарних кон'юнкцій R.

Крок 3а. Якщо в множині S є вже вектори, то повторюємо процедуру, описану як крок 3 при незмінному m для об'єднання множин V та S, проте відбір векторів K_i відбувається тільки тоді, коли умові P відповідає принаймні один вектор з множини V.

Крок 4. Повторюємо крок 3, збільшуючи щоразу фіксовану кількість змінних, поки вона менша або дорівнює n.

Крок 5. Здійснюємо диз'юнкцію всіх ЕК з множини R, яка і є результуючою МДНФ.

Крок 6. Кінець алгоритму.

Теорема 4. Даний алгоритм коректний і має часову складність $O(n * 6^n)$ і об'ємну складність $O(n * 2^n)$.

Доведення. Даний алгоритм належить до класу перебірних алгоритмів, проте перебір здійснюється в площині можливих ЕК. Перебираються всі можливі ЕК і для кожної з них перевіряється, чи може вона входити до складу ДНФ, яка відповідає заданій функції $f(x_1, \dots, x_n)$. Принцип перевірки є коректним, бо справді, якщо кількість ЕК в ДДНФ, в яких змінні збігаються зі змінними у вибраній елементарній кон'юнкції K, є максимально можливою, то всі інші змінні (яких немає в K) не відіграють жодної ролі для булевої функції. Наприклад:

$$a\,\bar{b}c\,\bar{d} \vee abc\,\bar{d} \vee \bar{a}bc\,\bar{d} \vee \bar{a}\,\bar{b}c\,\bar{d} =$$

$$= c\,\bar{d}(c\,\bar{d} \vee c\,\bar{d} \vee c\,\bar{d} \vee c\,\bar{d}) = c\,\bar{d} \wedge \text{true} = c\,\bar{d}.$$

Оскільки порядок виконання побудовано від ЕК з найменшою кількістю змінних до ЕК з найбільшою кількістю змінних, то таким чином на кожному кроці вибираються найкоротші ЕК і залишається довести, що алгоритм не вибирає зайві комбінації змінних, тобто такі ЕК, які можна утворити диз'юнкцією певної кількості вже вибраних елементарних кон'юнкцій.

Такий вибір зайвих ЕК відтинається специфічним порядком виконання операцій на кроках 3, 3а. Множина S зберігає вже "використані" ЕК, а множина V — ті, що ще треба використати. Механізм на кроці 3 (що вибирає всі повністю сформовані з невикористаних ЕК підмножини для перевірки), а після того механізм на кроці 3а (який не дозволяє вибирати з усіх ЕК такі підмножини, що в кожну з таких підмножин не входить хоча б одна ще не використана ЕК) не дозволяють вибрати таку підмножину ЕК, в якій всі ЕК були б уже вибрані.

Три принципи:

— вибір усіх можливих ЕК, які можуть входити до ДНФ функції $f(x_1, \dots, x_n)$,

— відсіювання всіх ЕК з більшою кількістю змінних і вибір тільки ЕК з найменш можливою кількістю змінних,

— виключення тих ЕК, які можна побудувати диз'юнкцією інших ЕК, призводять до створення на виході ДНФ найменшої довжини, що і вимагається умовою задачі.

Часова складність алгоритму зводиться до добутку можливої кількості ЕК в заданій ДДНФ (яка обмежена числом 2^n), кількості всіх можливих ЕК для n змінних (H) та довжини ЕК в ДДНФ.

Кількість всіх можливих ЕК для n змінних дорівнює 3^n, що було доведено вище. Довжина ЕК дорівнює n. Отже, загальна часова складність алгоритму буде $O(n * 2^n * 3^n) = O(n * 6^n)$.

Вся робота алгоритму обмежена обсягом пам'яті, що утримує множину ЕК. Найбільша можлива кількість ЕК дорівнює 2^n, а найбільша довжина кожної ЕК може дорівнювати кількості змінних n. Отже, об'ємна складність алгоритму дорівнює $O(n * 2^n)$.

Теорема доведена.

Даний алгоритм є важливим скоріше з теоретичної точки зору, тому не виключено, що для практичного застосування він може бути вдосконалений.

Висновки

1. Будь-який алгоритм розв'язання задачі мінімізації булевих функцій теоретично може мати експоненційну об'ємну та часову складність, не нижчу $O(2^n)$.
2. Дана задача не належить до класу NP-повних задач.
3. Існує коректний алгоритм розв'язання цієї задачі з часовою складністю $O(n * 6^n)$ і об'ємною складністю $O(n * 2^n)$, запропонований в даній роботі. Переваги алгоритму: його часова складність наближена до теоретично можливої і є строго обмеженою, в той час, як інші відомі алгоритми теоретично можуть мати час виконання до $O(2^{2^n})$; простота реалізації. Недолік його в тому, що він може бути вдосконалений і мати (можливо, в окремих випадках) меншу часову складність.
4. Застосування запропонованого алгоритму найбільш ефективне в комплексних алгоритмах. Оскільки даний алгоритм має гарантовану верхню

межу часової складності, його можна використовувати паралельно з іншими алгоритмами, що дозволяють знаходити МДНФ в багатьох випадках швидше, ніж цей алгоритм. Якщо часова складність іншого евристичного алгоритму досягне часової складності даного алгоритму, то система, що розв'язує задачу, може застосувати для пошуку МДНФ даний алгоритм. Таким чином, час розв'язання задачі МБФ ніколи не перевищить 0(« * 6").

1. Дискретная математика и математические вопросы кибернетики / Под общ. ред. Яблонского С. В. и др. — М., 1974.— Т. 1.—С. 22—23.
2. *Горбатов В. А.* Основы дискретной математики.— М: Высш. школа, 1986 [check pages].
3. *Яблонский С. В.* Функциональные построения в А-значной логике // Труды МИАН СССР.— М., Изд-во АН СССР, 1958.—Т. 51.—С. 5—142.

Glibovets M. M., Ivashchenko S. A.

ABOUT A CERTAIN APPROACH TO MINIMIZATION BOOLEAN FUNCTIONS PROBLEM SOLVING

The method of examining lower limit of time and size complexity of the problem is developed. Also impossibility of developing polynomial time algorithms for minimization boolean functions problem solving is proved. The algorithm of the problem solving with fixed time complexity is proposed.