УДК 510.5

Ю. А. Василенко, Г. Е. Копча (Закарпатський держ. ун-т)

ОБЧИСЛЮВАЛЬНА СХЕМА В ЯКОСТІ АЛГЕБРАЇЧНОГО ОБ'ЄКТУ

There is the principally new approach the conception of scheme, analysis in the paper from the point of view general algebra. This approach, first, establishes a natural connection between algebra and theory, algorithms, and second, is of great practical importance, since it enables applications in computational mathematics, automata theory and other areas that use different schemes of transformation of information.

В даній роботі розглянуто принципово новий підхід до поняття схеми — з точки зору загальної алгебри. Такий підхід, по-перше, встановлює природній зв'язок між алгеброю та теорією алгоритмів, по-друге, має велике практичне значення, оскільки дає можливість його застосування в обчислювальній математиці, теорії автоматів та інших областях, де використовуються різні схеми перетворення інформації.

На даний час існує багато визначень схеми, які, в деякій мірі, повністю описують інтуїтивне поняття обчислювальної схеми. Але, з другого боку, в цих визначеннях є деякі обмеження, наприклад, суттєвим обмеженням являється те, що в цих схемах розглядаються конструктивні операції і предикати, які здійснюються над конструктивними об'єктами. Таким чином, ці визначення тісно пов'язані з ідеєю програми на ЕОМ і мають слабкий зв'язок з основними розділами сучасної алгебри.

Взагалі, до цього часу два великі розділи математики „Алгебра" і „Обчислювальна математика" розвивалися незалежними, паралельними шляхами. Кожен розділ оперував своїми власними математичними інструментами.

Але виявляється, що перетин цих шляхів можливий і дуже корисний.

Як показано в [1], поняття схеми може бути описане не лише з точки зору конструктивності (в єдності з ідеєю машинної програми), але й з точки зору теоретичної, „чистої" алгебри як універсальний та математично строгий об'єкт.

Такий підхід до поняття схеми, по-перше, встановлює природній зв'язок між алгеброю та теорією алгоритмів, по-друге, має велике практичне значення, оскільки дає можливість його застосування в теорії автоматів, обчислювальній математиці та інших областях, де використовуються різні схеми перетворення інформації [2,3].

Розглянемо наступний підхід по поняття схеми [1]. Цей підхід зручно проілюструвати на прикладах схем:

$$F^2(x,y) = f^2(\phi^2(x,y), f^2(x,y)), \tag{1}$$

$$r^1(x) = \begin{cases} f_1^1(x), & \text{якщо } p^1(x) = 0, \\ f_2^1(x), & \text{якщо } p^1(x) = 1, \end{cases} \tag{2}$$

де $f^2, \phi^2, f_1^1, f_2^1$ — функції, $p^1(x)$ — предикат. Схема (1) за довільними бінарними функціями $f^2(x,y)$ і $\phi^2(x,y)$ однозначно визначає деяку бінарну функцію $F^2(x,y)$. Схема (2) за довільними унарними функціями $f_1^1(x)$, $f_2^1(x)$ і унарним предикатом $p^1(x)$ однозначно визначає унарну функцію $r^1(x)$. Можна узагальнити поняття схеми для функції довільної арності [1]. Визначимо схему таким чином.

Схемою називається правило (оператор), яке кожному набору функцій $f_1^{n_1}$, $f_2^{n_2}$, $\ldots$, $f_k^{n_k}$ і предикатів $p_1^{s_1}$, $p_2^{s_2}$, $\ldots$, $p_m^{s_m}$ ставить у відповідність деяку функцію f^r (або предикат), де $n_1, \ldots, n_k, s_1 \ldots, s_m, r$ означають арності функцій та предикатів. Перевагами цього означення є універсальність, математична строгість, а також те, що це означення дозволяє встановити природній зв'язок між алгеброю і теорією алгоритмів та програмуванням. Дана теорія схем може бути успішно застосована в обчислювальній математиці, теорії автоматів та інших областях.

Крім описаного строгого визначення алгоритмічної схеми в роботі [1] ставляться такі проблеми:

1. Знаходження в класі K-многовидів (K — клас алгебр, який визначається деякою сукупністю схем) по можливості простої повної системи схем K_1, тобто такої системи K_1, що $[K_1] = K$. Наприклад, знаходження повної системи в класі Θ всіх алгоритмічних систем.

2. Алгебраїчний і категоричний опис K-многовидів.

3. Дано поняття K-многовиду (частинні випадки: напівгрупи і булеві алгебри).

4. Вивчені деякі загальні властивості схем.

5. Описані два нові K-многовиди, тобто знайдені повні системи тотожностей в цих многовидах.

В [1] розглянуто деякі важливі схеми: Z, Z_1, Z_2. Z_1-схемою називається Φ_1 і R_1-схеми, тобто відображення $L \to D$ і $L \to \{0,1\}$ (L – множина всіх комплексів $D_p = \bigcup\limits_{i=0}^{\infty} D_p^i$, D_p^i — множина всіх слів виду $p_j s$, де $p_j \in P$, $s \in D^i$, де P — деяка скінчена система символів, тобто алфавіт, D^i — множина всіх слів з D, довжина яких не перевищує i ($i = 0, 1, \ldots$), D — множина всіх слів з алфавіту F, що є деякою скінченою системою символів, яка відмінна від P).

В [1] визначені та вивчені алгоритмічні Z_1-схеми. Звичайно, що ці схеми не вичерпують всіх алгоритмічних схем. Для визначення класу Θ всіх алгоритмічних схем вводиться поняття Z_2-схеми. Z_2-схемами будемо називати Φ_2 і R_2-схеми, тобто, відповідно, частинні відображення виду $L \to D$ і $L \to \{0,1\}$. Оскільки повне відображення є частинним випадком часткового відображення, то кожна Z_1-схема являється Z_2-схемою. Z_2-схема a кожному (G,n,K)-набору $f_1, f_2, \ldots, f_n$; $p_1, p_2, \ldots, p_k (f_i \in \Phi'(G), p_j \in R'(G))(i = 1,2,\ldots,n; j = 1,2,\ldots,k)$ (де $\Phi'(G)$ — множина всіх всюдивизначених операцій на деякій множині G, $R'(G)$ – множина всіх всюдивизначених предикатів на цій множині) ставить у відповідність деякий елемент f_a із $Z(G)$ ($Z(G) = \Phi(G) \cup R(G)$, $\Phi(G)$ — множина всіх операцій на множині G, $R(G)$ – множина всіх предикатів на цій множині).

В [1] дається визначення фінітарної схеми і показується, що фінітарність є необхідною умовою для алгоритмічності Z_2-схеми.

Кожне регулярне відображення a представляє собою схему, яка кожному (G, n, K)-набору $f_1, f_2, \ldots, f_n$; p_1, p_2, $\ldots$, p_k ($f_i \in \Phi'(G)$, $p_j \in R'(G)$) ($i = 1$, 2, $\ldots$, n; $j = 1$, 2, $\ldots$, k) ставить у відповідність деякий елемент f_a із $Z(G)$.

Вводиться поняття Γ-схеми, їх зв'язок з Z_2-схемами.

Регулярні відображення виду $L^\infty \to D$, $L^\infty \to \{0, 1\}$ (L — множина всіх D_p-комплексів, $D_p = \bigcup\limits_{i=0}^{\infty} D_p^i$, D — множина всіх слів з алфавіту $F = f_1, \ldots, f_n$, D^i — множина всіх слів із D, довжина яких не перевищує i, D_p — множина всіх слів виду $p_j s$, $p_j \in P$, $s \in D$, $P = \{p_1, \ldots, p_k\}$, D_p^i — множина всіх слів виду $p_j s$, $p_j \in P$, $s \in D^i$, $i = 0, 1, \ldots$) називаються Γ-схемами.

Доводяться такі основні теореми [1].

Теорема 1. *Кожна фінітарна Z_2-схема представляється за допомогою деякої Γ-схеми a' і, навпаки, кожна Γ-схема a' представляє деяку фінітарну Z_2-схему a. Схеми a і a' Z'-еквівалентні* ($Z'(G) = \Phi'(G) \cup R'(G)$).

Теорема 2. *Дві Z_2-схеми Z'-еквівалентні тоді і тільки тоді, коли вони рівні.*

З цих двох теорем випливає така:

Теорема 3. *Дві Γ-схеми Z'-еквівалентні тоді і тільки тоді, коли вони представляють одну і ту ж фінітарну Z_2-схему a.*

Кожна алгоритмічна Z_2-схема має бути фінітарною Z_2-схемою. Тому клас алгоритмічних Z_2-схем визначається як підклас класу фінітарних Z_2-схем.

Γ-схема b називається алгоритмічною, якщо вона представляє собою частково-рекурсивне відображення виду $L^\infty \to D$ або $L^\infty \to \{0, 1\}$.

Z_2-схема a називається алгоритмічною, якщо вона представляється за допомогою деякої алгоритмічної Γ-схеми. Алгоритмічну Γ-схему b можна обчислювати за деяким алгоритмом A_b.

Таким чином, для кожної алгоритмічної Z_2-схеми a існує ефективна процедура, яка обчислює значення $f_a(x)$ (схеми a і b розглядаються на (G, n, K)-наборах із $Z'(G)$).

Розглядається поняття еквівалентності Z_2-схем. Доводиться теорема, що такі схеми будуть еквівалентними тоді і тільки тоді, коли вони рівні.

В [1] визначається поняття Z-схем. Ці схеми відрізняються від Z_1-схем і Z_2-схем тим, що зазначені схеми застосовні лише до таких (G, n, K)-наборів, в яких операції f_i і p_j всюди визначені, а Z-схеми застосовні до будь-яких (G, n, K)-наборів.

Γ-схеми можна застосовувати і до частково-визначених операцій, і предикатів.

Якщо порівнювати Z_2-схеми і Z-схеми, то можна помітити, що кожна Z_2-схема a' є обмеженням деякої Z-схеми на множині L. Тому кожну Z_2-схему можна представити як деяку Z-схему, що розглядається на (G, n, K)-наборах із $Z'(G)$. І навпаки, кожна Z-схема, що розглядається на (G, n, K)-наборах із $Z'(G)$ представляє собою деяку Z_2-схему.

Будь-який алгоритм обчислює деяку схему, а довільний N-алгоритм обчислює частково-рекурсивну функцію. Розглядаються алгоритмічні схеми, які мають вигляд програм. Ці схеми названі Π-схемами. Показано, що існують алгоритмічні схеми, які не еквівалентні жодній Π-схемі. Поняття алгоритмічної Z-схеми вважається уточненням інтуїтивного поняття алгоритмічної схеми.

Класи напівгруп та булевих алгебр являються частинними випадками K-многовидів, оскільки наприклад, схема $f_2(f_1(x))$ призводить до поняття напівгрупи, а схеми $p_1 \vee p_2$, $p_1 \wedge p_2$, $\overline{p}$ призводять до поняття булевої алгебри.

В [1] дані повні системи тотожностей для деяких нових K-многовидів, які названі предикатними алгебрами і предикатними напівгрупами відповідно. Повні системи тотожностей представляють собою повні системи еквівалентних перетворень в деяких класах схем.

Існує аналогія між класами алгебр і алгоритмічними схемами (стосовно операцій), з якої випливає такий висновок: будь-яка сукупність алгоритмічних схем повинна визначати певний клас алгебр. Тому досить цікавою проблемою є проблема вивчення класів алгебр, в основі яких лежать типові алгоритмічні схеми.

Розглянемо деякі алгебри та класи алгебр. K-многовид — це клас алгебр, який визначається певною сукупністю схем. Матимемо на увазі клас всіх алгебр виду $[M, K]$, де K — деяка множина схем, відносно якої система M замкнута, множина K фіксована, і будемо називати його K-многовидом.

В [1] одержано алгебраїчне описання K-многовидів виду $[M, K_1]$ і $[M, K_2]$, де $M \subset Z'(G)$ ($Z'(G) = \Phi'(G) \cup R'(G)$, $\Phi'(G)$ — множина всіх всюдивизначених операцій на деякій множині G, $R'(G)$ — множина всіх всюдивизначених предикатів на цій множині, K — деяка множина схем, $f_i \in \Phi'(G)$, $p_j \in R'(G)$), $K_1 = \{p(f_1, f_2)\}$ і $K_2 = \{f_1 f_2, p(f_1, f_2)\}$.

Алгебри із многовидів $[M, K_1]$ і $[M, K_2]$, відповідно, називатимемо алгебрами і предикатними напівгрупами.

При вивченні предикатних алгебр будемо розглядати замість відображень $f_i\colon G \to G$ більш загальні відображення $f_i\colon G \to B$, де G і B — довільні множини.

Нехай $\overline{H_0}$ — множина всіх відображень множини G в B. Будь-який одномісний предикат $p(x)$ над G задає деякий двохмісний оператор на множині $\overline{H_0}$, який визначається таким чином:

$$p(h_1, h_2) = \begin{cases} h_1(x), & \text{якщо } p(x) = 0 \text{ (хибність);} \\ h_2(x), & \text{якщо } p(x) = 1 \text{ (істина).} \end{cases} \tag{3}$$

Тут $h_1, h_2 \in \overline{H}_0, x \in G$. Очевидно, що $p(h_1, h_2) \in \overline{H}_0$. Оператор (3) називається предикатним оператором. Будемо вважати, що при записуванні предикатного оператора у виді $p(h_1, h_2)$ символ p означає відповідний предикат.

Система відображень $\overline{H} \subset \overline{H}_0$ замкнута відносно предиката p, якщо з h_1, $h_2 \in \overline{H}$ випливає $p(h_1, h_2) \in \overline{H}$. Система H замкнута відносно системи предикатів із $\overline{L}$, якщо H замкнута відносно всіх предикатів із $\overline{L}$. Сукупність предикатів $\overline{L}$ і замкнутої відносно $\overline{L}$ системи H утворює деяку алгебру. Цю алгебру назвемо предикатною алгеброю. Опишемо предикатну алгебру.

Розглянемо абстрактну алгебру, яка представляє собою сукупність множин H і заданої на H системи L бінарних операцій $p(h_1, h_2)$ (h_1, h_2, $p(h_1, h_2 \in H)$). Операції із L задовольняють наступним тотожностям:

$$\begin{array}{l} \text{а) } p(h, h) = h, \\ \text{б) } p(p(h_1, h_2), p(h_3, h_4)) = p(h_1, h_4), \\ \text{в) } p_1(p_2(h_1, h_2), p_2(h_3, h_4)) = p_2(p_1(h_1, h_3), p_1(h_2, h_4)), \end{array} \tag{4}$$

де $h, h_1, h_2, h_3, h_4 \in H$, $p, p_1, p_2 \in L$.

Таким чином визначену алгебру назвемо N-алгеброю. Очевидно, що всяка предикатна алгебра являється N-алгеброю.

В [1] доведена така важлива теорема.

Теорема 4. *Кожна вільна N-алгебра ізоморфна деякій предикатній алгебрі.*

Розглянемо вільну N-алгебру A, породжену деякою системою твірних N і системою операцій L. Вирази вільної N-алгебри A, елементи із N і L, відповідно, будемо називати формулами, H-символами і L-символами. Формули f_1 і f_2, які зображають один і той же об'єкт вільної N-алгебри A, називаються еквівалентними. Еквівалентність і нееквівалентність позначаються, відповідно, через $f_1 \approx f_2$ і $f_1 \not\approx f_2$. Формула f належить системі формул M ($f \in M$), якщо f еквівалентна одній із формул M. Через $f_1 = f_2$ позначимо співпадання формул f_1 і f_2. У вільній N-алгебрі A для кожного цілого числа $n \geq 0$ визначимо формули такого виду: $F(p_1, p_2, \dots, p_n; h_1, h_2, \dots, h_{2^n})$, де $h_1, h_2, \dots, h_{2^n} \in H$, $p_1, p_2, \dots, p_n \in L$, причому $p_i \neq p_j$ при $i \neq j$.

Формули $F(p_1, p_2, \dots, p_n; h_1, h_2, \dots, h_{2^n})$ визначаються індукцією по n. Для $n = 0$ покладаємо $F(0; h_1) = h_1$ (0 — символ пустого кортежу $p_1, p_2, \dots, p_n$). Якщо для n формули F визначені, то для $n + 1$ покладаємо

$$F(p_1, p_2, \dots, p_{n+1}; h_1, h_2, \dots, h_{2^{n+1}}) =$$
$$= p_1(F(p_2, p_3, \dots, p_{n+1}; h_1, h_2, \dots, h_{2^n}), F(p_2, p_3, \dots, p_{n+1}; h_{2^n+1}, h_{2^n+2}, \dots, h_{2^{n+1}})).$$

Таким чином, маємо

$$F(p_1; h_1, h_2) = p_1(h_1, h_2),$$

$$F(p_1, p_2; h_1, h_2, h_3, h_4) = p_1(p_2(h_1, h_2), p_2(h_3, h_4))$$

і т. д.

Наслідок 1. *Сукупність рівностей* (4) *представляє собою алгебраїчний опис класу всіх предикатних алгебр.*

В роботі [1] повністю описані і вивчені алгоритмічні схеми без циклів. В теорії цих схем важливе місце посідає один клас схем — вищезгадані Z_1-схеми. Z_1-схема є найбільш загальним поняттям оператора, який діє над наборами функцій і предикатів і який зберігає всюдивизначеність. Розглядаються елементарні Z_1-схеми, тобто такі схеми, які є звичайними алгоритмічними схемами без циклів.

Доведена така основна теорема.

Теорема 5. *Z_1-схема являється тоді і тільки тоді фінітарною, коли вона є елементарною.*

Дана теорема описує всі алгоритмічні схеми, які зберігають всюдивизначеність.

Показано, що всяка елементарна Z_1-схема представляється за допомогою виразу, що складається із операторів підстановки функцій в предикат, кон'юнкції, диз'юнкції, заперечення предикатів і оператора умовного обчислення.

З вищесказаного випливає важливий результат: всі алгоритмічні схеми, які зберігають всюдивизначеність, вичерпуються звичайними програмами, які складаються із вищевказаних операторів.

Таким чином, дана теорія схем може бути застосована в теорії програмування, теорії алгоритмів, обчислювальній математиці і інших областях, в яких застосовуються різні схеми перетворення інформації [2,3].

Можна сформулювати такі основні проблемні задачі, які випливають з результатів роботи [1]:

1. Дослідження різних класів схем, тобто їх класифікація, дослідження Z_1, Z_2, Z-схем і Π-схем, а також знаходження необхідних і достатніх умов належності схем до кожного з вищезгаданих класів схем.

2. Зображення алгоритмічних схем за допомогою алгоритмів, тобто вивчення і аналіз регулярних алгоритмів, їх використання для зображення схем, аналіз пам'яті, швидкодії та інших характеристик алгоритму, що зображує схему, та оптимізація алгоритмів, за допомогою яких зображуються схеми.

3. Повнота систем алгоритмічних схем відносно суперпозиції.

4. Еквівалентні перетворення і спрощення алгоритмічних схем (Вивчення різноманітних виразів, за допомогою яких зображуються алгоритмічні схеми, знаходження повних систем правил перетворення цих виразів та методів спрощення виразів і дослідження проблеми складності виразів).

5. Застосування теорії алгоритмічних схем в програмуванні, теорії цифрових автоматів, теорії аналого-цифрових перетворювачів, теорії розпізнавання образів (вивчення повноти алгоритмічних мов відносно представлення в них алгоритмічних схем, застосування теорії алгоритмічних схем в автоматизації програмування, вивчення проблеми синтезу різноманітних цифрових автоматів з точки зору алгоритмічних схем, вивчення проблеми синтезу оптимальних аналого-цифрових перетворювачів з позиції теорії алгоритмічних схем в теорії розпізнавання образів і теорії систем).

1. *Витенько И. В.* Схемы, алгоритмы и многообразия. – Ужгород: Уж. ун-т, 1970. – 75 с.
2. *Вітенько І. В.* Конструктивні операції. – Ужгород: Уж. ун-т, 1970. – 212 с.
3. *Вітенько І. В.* Математична логіка. – Ужгород: Уж. ун-т, 1971. – 214 с.

Одержано 10.01.2010