УДК 681.3.06

І. В. Редько, Н. М. Снігур
Національний технічний університет України «КПІ»
проспект Перемоги, 37, 01037 Київ, Україна
тел. (044) 477-10-60

Алгебраїчна характеристика класу графових перетворювачів, які зберігають денотати

Розглянуто клас функцій, які зберігають денотати на множині графів. Визначено породжуючу множину алгебри функцій, які зберігають денотати на новому носії — графі, а також доведено її повноту.

***Ключові слова**: алгебра функцій і предикатів, які зберігають денотати, скінчений (орієнтований) граф, функція та предикат довільної арності, зберігаючі денотати, пошук породжуючих множин і базисів.*

Вступ

Проблема формального опису семантики програм виникла з появою програмування. Одним із науково обґрунтованих підходів у програмуванні є застосування алгебраїчних методів дослідження програм. В основі таких методів лежать програмні алгебри, тобто алгебри, носіями яких є спеціальні класи функцій, а операціями — композиції, що представляють собою абстарції від засобів синтезу програм. Зокрема, в термінах цих алгебр точно ставляться та вирішуються проблеми повноти в різних класах обчислюваних функцій. Ці проблеми займають одне з найважливіших місць у сучасному програмуванні.

Дана робота присвячена вивченню обчислюваних графових перетворювачів, що зберігають денотати. Вперше поняття функції, що зберігає денотати, було введено в [1] для опису семантики мов маніпуляції даними. В даній роботі вихідне визначення зберігання денотатів узагальнюється за допомогою введення параметра — так званої оцінки.

Обчислюваність вводиться згідно нумераційному підходу [1]. Вибір графових структур обумовлений їхньою важливістю та популярністю в теоретичному та прикладному програмуванні (див., наприклад, [3–11]). Зокрема, в роботах [3, 4] визначено та досліджено поняття обчислюваної функції над скінченими графами (комплексами), [5] присвячена вивченню граф-схем алгоритмів, [6–9] — ізоморфізму графів, [10] — питанням абстрактної обчислюваності в різних областях, [11] — графовим засобам специфікації програм. При цьому треба зазначити, що практично всі згадані дослідження акцентують увагу безпосередньо на графи, не розглядаючи задачі огляду класу обчислюваних функцій над графами (графових перетворювачів).

Інструментом розв'язання поставлених проблем виступають так звані примітивні програмні алгебри (ППА) [2]. Доцільність даного вибору обґрунтована результатами, отриманими, наприклад, в [12–24]. Основну увагу приділено пошуку породжуючої множини для такої ППА. Усі використані та невизначені в роботі поняття та позначення розуміються в сенсі [15].

Базові визначення, поняття, результати

Носій ППА складають n-арні функції і предикати для $n = 1,2,\ldots$ Сигнатуру ППА (позначатимемо її Ω) складають операції суперпозиції, розгалуження та $(n+1)$-арного циклування (визначення див. в [12]).

Під (скінченим) *орієнтованим графом* g розуміємо пару $< V, E >$, де V — деяка злічена непуста множина об'єктів, а E — бінарне відношення на V.

Стандартна інтерпретація цих об'єктів V та E в даному визначенні така: V — множина вершин графа, E — множина його дуг. У силу зліченості множини V не буде суттєвим обмеженням, якщо покласти $V = N$. При цьому на множині вершин вводиться цілком природна впорядкованість.

Домовимось надалі вершини графа позначати латинськими літерами u, v, w, а дуги — літерами e, p, s, можливо з індексами, $v_1,\ldots,e_1,\ldots$ У випадку необхідності явно вказати вершини дуги та її напрям, використовуватимемо запис $< v,u >$, розуміючи, що дуга направлена від v до u. Будемо називати *першою* вершиною графа вершину з найменшим номером. Граф будемо позначати через $g^r_{|E|}$, де r — номер першої вершини графа g, а $|E|$ — кількість його дуг, або скорочено просто g.

Вершини v та u дуги $e \equiv < v,u >$ будемо називати *суміжними*. При цьому дуга e вважається *додатно інцидентною* вершині u та *від'ємно інцидентною* вершині v. За аналогією домовимося вершину v графа $g \equiv < V, E >$ називати від'ємно (додатно) інцидентною (*від'ємною* (*додатною*)) до вершини w того ж графа, якщо $< w,v > \in E$ $(< v,w > \in E)$.

Множину всіх графів позначимо G. Під *функціями* далі розуміємо часткові функції з аргументами і значеннями із G, а під предикатами — часткові предикати на G.

Обчислюваність на G вводиться як нумераційна обчислюваність за допомогою арифметичної функції, що представляє дану функцію на G в деякій зафіксованій нумерації α_G множини G [19]. Існування такої нумерації випливає зі зліченості множини $V = N$.

Будь-яку частково-рекурсивну багатомісну функцію або будь-який частково-рекурсивний багатомісний предикат (чр-функція, чр-предикат) будемо називати також *графовим перетворювачем*.

Нехай $\beta : G \underset{на}{\rightarrow} 2^V$, де V — множина вершин і, відповідно, 2^V — множина всіх скінчених підмножин V.

Будемо казати, що *функція* f арності n β*-зберігає денотати*, якщо існує скінчена множина $V_f \subset V$ така, що для будь-якого $< x_1,\ldots,x_n > \in \mathrm{dom} f$ виконується

$$\beta(f(< x_1,\ldots,x_n >)) \subseteq \bigcup_{i=1}^{n} \beta(x_i) \cup V_f .$$

Через $A_{G,\beta}^{чр}$ позначимо ППА, носій якої складають графові перетворювачі на G, які β-зберігають денотати. *Породжуючу множину* алгебри $A_G^{чр}$ назвемо її повною системою (ПС), ПС ППА — I_m^n *базисом*, якщо будь-яка її підсистема, що отримується видаленням якого-небудь предиката або якої-небудь функції, відмінної від селекторної, вже не буде повною.

Примітивна програмна алгебра чр-функцій і чр-предикатів на множині скінчених орієнтованих графів

З вищезазначеного є цілком очевидним, що будь-який граф g можна ефективно представити певним вектором A_g, причому ефективність такого представлення прямо витікає з її побудови.

Нехай $g = \left\langle V = \{v_1,\ldots,v_n\}, E = \left\{\left\langle v_{i_1}, v_{i_2}\right\rangle, \left\langle v_{i_3}, v_{i_4}\right\rangle, \ldots\right\}\right\rangle$, $1 \le i_j \le n$, $j = 1,2,\ldots$.

У подальшому буде зручно перенумерувати також і дуги графа. Для цього домовимось про певну впорядкованість пар натуральних чисел[1]. Зауважимо, що порядок на множині N^2 можна вводити по-різному[2], проте в даному випадку більш зручним нам видається наступний:

$$\langle 0,0\rangle < \langle 0,1\rangle < \langle 0,2\rangle < \ldots < \langle 0,n\rangle < \ldots < \langle 1,0\rangle < \langle 1,1\rangle < \ldots < \langle 2,0\rangle < \langle 2,1\rangle < \ldots \qquad (1)$$

Тоді, дуги графа занумеруємо відповідно порядку (1), тобто першою дугою буде дуга, що відповідає найменшій парі в цьому порядку (найлівішій), а найбільший номер матиме дуга, що відповідає найбільшій парі (найправішій).

Покладемо $A_g = \left(v_{i_1} v_{i_2} v_{i_3} v_{i_4} \ldots 0 v_{k_1} 0 v_{k_2} \ldots\right)$, де v_{k_j}, $j = 1,2,\ldots$, — ізольовані вершини (якщо вони є). Зокрема, в такому представленні порожньому графу, очевидно відповідає порожній вектор Λ, а повністю незв'язаному графу, тобто графу $g \equiv < V, E >$, де $E = \mathrm{O}$, відповідає вектор $\left(0 v_1 0 v_2 0 \ldots 0 v_n\right)$. Таким чином, побудоване відображення $\varphi : \mathsf{G} \to \mathsf{N}^*$, де $\mathsf{N}^* = \bigcup_{i=0}^{\infty} \mathsf{N}^i$, $\mathsf{N}^0 = \{\Lambda\}$. Очевидно, φ — ін'єкція, але не сюр'єкція. Позначимо $\mathsf{V} := \varphi(\mathsf{G})$. Вочевидь, ця множина є рекурсивною в нумерації α_{G}.

[1] Такий крок є цілком природним, оскільки множина N^2, як відомо є ефективно зліченою.
[2] Наприклад, як в *канторівській нумерації* (див. [1, стр. 60]).

Нижче під L-функцією (L-предикатом) мається на увазі багатомісна частково операція (предикат) на множині L.

Означення 1. V-функцію $F(x_1,\ldots,x_n)$ назвемо векторним образом графого перетворювача $F(\varphi(g_1),\ldots,\varphi(g_n)) \cong \varphi F(g_1,\ldots,g_n)$ для всіх $g_1,\ldots,g_n$. Аналогічно V-предикат $P(x_1,\ldots,x_n)$ назвемо векторним образом граф-предиката $\mathsf{P}(\xi_1,\ldots,\xi_n)$, якщо $P(\varphi(g_1),\ldots,\varphi(g_n)) \cong \varphi P(g_1,\ldots,g_n)$ для всіх $g_1,\ldots,g_n$.

Лема 1. Векторний образ чр-графового перетворювача (чр-графового предиката) є чр-V-функція (чр-V-предикат).

Безпосередньо із рекурсивності множини V випливає наступна лема.

Лема 2. Будь-яка чр-V-функція є чр-N^*-функцією (тобто просто векторною чр-функцією). Для чр-V-предикатів аналогічно.

Наслідок 1. Векторний образ чр-графового перетворювача (чр-граф-предиката) є векторною чр-функцією (векторним чр-предикатом).

З метою моделювання векторних функцій графовими перетворювачами побудуємо також кодуюче відображення $\Phi : \mathsf{N}^* \to \mathsf{G}$. Для цього будь-якому вектору $A = (v_1 v_2 \ldots v_n)$ поставимо у відповідність граф $g = \langle V = \{v_1, v_2, \ldots, v_n\}$, $E = \{\langle v_1, v_2\rangle, \langle v_2, v_3\rangle, \ldots, \langle v_{n-1}, v_n\rangle\}\rangle$.

Означення 2. Графовий перетворювач $\mathsf{F}(\xi_1,\ldots,\xi_n)$ називається граф-моделлю векторного перетворювача $F(x_1,\ldots,x_n)$, якщо $\mathsf{F}(\Phi(v_1),\ldots,\Phi(v_n)) \cong \cong \Phi(F(v_1,\ldots,v_n))$ для всіх $v_1,\ldots,v_n$. Граф-модель предиката вводиться аналогічно.

Лема 3. Для будь-яких векторних чр-функцій та чр-предикатів існують їхні граф-моделі, які належать замиканню $[\sigma_{\mathsf{G}}]_\Omega$.

Нехай $\psi := \varphi \cdot \Phi$. Очевидно, що $\psi : \mathsf{G} \to \Phi(\mathsf{V})$ — бієкція. Через χ позначимо яке-небудь розширення відображення ψ^{-1}. Графові перетворювачі ψ та χ грають ролі кодуючої та декодуючої функцій відповідно.

Безпосередньо із наведених вище означень випливає лема.

Лема 4. Нехай $\mathsf{F}(\xi_1,\ldots,\xi_n)$ — чр-граф-функція, а $\mathsf{H}(\pi_1,\ldots,\pi_n)$ — граф-модель векторного образу функції $\mathsf{F}(\xi_1,\ldots,\xi_n)$. Тоді

$$\mathsf{F}(A_1,\ldots,A_n) = \chi(\mathsf{H}(\psi(A_1),\ldots,\psi(A_n)))$$

для всіх A_i, $i = \overline{1,n}$.

Аналогічно, нехай $\mathsf{P}(\xi_1,\ldots,\xi_m)$ — чр-граф-предикат, а $\mathsf{K}(\pi_1,\ldots,\pi_m)$ — граф-модель векторного образу цього предиката. Тоді

$$\mathsf{P}(A_1,\ldots,A_n) = \mathsf{K}(\psi(A_1),\ldots,\psi(A_n))$$

для всіх A_i, $i = \overline{1,n}$.

Розглянемо наступні функції на множині графів (графові перетворювачі).

1. C_0^{G} — константна функція: $C_0^{\mathsf{G}}(g) = g_0^1$.

2. $\hat{\cup}$ — об'єднання графів: $g_1 = \langle V_1, E_1\rangle$, $g_2 = \langle V_2, E_2\rangle$, $g_1 \hat{\cup} g_2 = \langle V_1 \cup V_2, E_1 \cup E_2\rangle$.

3. $\backslash$ — різниця графів: $g_1 = \langle V, E_1\rangle$, $g_2 = \langle V, E_2\rangle$, $g_1 \backslash g_2 = \langle V, E_1 \backslash E_2\rangle$.

4. E_e — виділення «першої» дуги, тобто дуги, номер якої стосовно введеної вище впорядкованості є найменшим. Формально, це означає, що якщо $g = \left\langle \{v_1, \ldots\}, \left\{\left\langle v_1, v_{i_1}\right\rangle, \ldots\right\}\right\rangle$, то $E_e(g) = \left\langle v_1, v_{i_1}\right\rangle$.

5. E_v — виділення «першої» вершини графа $g = \langle V = \{v_1, \ldots\}, E\rangle$: $E_v(g) = \left\langle \{v_1\}, \mathrm{O}\right\rangle$.

6. D_v — видалення «першої» вершини, якщо вона ізольована.

7. L — створення дуги. Часткова функція, що діє на множині з двох нульових графів $\{m\}$ та $\{n\}$ таким чином, що $L(\{m\},\{n\}) = \left\langle \{m, n\}, \{\langle m, n\rangle\}\right\rangle$.

8. D_e — видалення «першої» дуги графа $g = \langle V, E\rangle$ (див. визначення функції E_e)

9. $\overset{\circ}{D}_e$ — видалення всіх дуг графа $\overset{\circ}{D}_e(\pi) = \left(E_e(\pi) \neq \Delta\right) \underset{\pi}{*} \left\langle D_e(\pi)\right\rangle$.

10. $C_{\Delta_{\mathsf{G}}}$ — функція генерації порожнього графа: $C_{\Delta_{\mathsf{G}}}(g) = \Delta_{\mathsf{G}}$.

Покладемо

$$\sigma_G := \{C_0^G, \hat{\cup}, \backslash, E_e, E_v, D_v, L, =_G, I_m^n\}_{m=\overline{1,n}}^{n=1,2,\ldots}.$$

Можна безпосередньо перевірити, що всі ці функції є частково-рекурсивними.

Лема 5. Має місце

$$\psi, \chi \in [\sigma_{\mathsf{G}}]$$

Доведення.

1. ψ.

Покладемо

$$\mathsf{G}_1(\pi, \xi) = (E_e(\xi) \neq \Delta) \underset{\pi,\xi}{*} \left\langle \pi \cup E_e(\xi) \cup L\left(D_v\left(D_e\left(E_e(\xi)\right)\right), E_v\left(E_e\left(D_e(\xi)\right)\right)\right), D_e(\xi)\right\rangle.$$

Тоді $\mathsf{G}_1(\Delta, g)$ представляє собою граф-код за винятком ізольованих вершин. А отже, повний граф-код

$$\hat{g} = \psi(g) = G_1(\Delta, g) \cup \overset{\circ}{D}_e(g).$$

2. χ .

Розглянемо функцію

$$\mathsf{G}_2(\pi,\xi)=\left(E_e\left(\xi\right)\neq\Delta\right)\underset{\pi,\xi}{*}\left\langle\pi\cup E_e\left(\xi\right),D_e\left(D_e\left(\xi\right)\right)\right\rangle .$$

Тоді $g=\chi(\hat{g})=G_2\left(\Delta,\hat{g}\right)\cup\overset{\circ}{D}_e(\hat{g})$, що і треба було довести.

Теорема 1. σ_{G} є породжуючою множиною алгебри $A^{\text{чр}}_{\mathsf{G},\beta}$.

Доведення.

Твердження теореми випливає безпосередньо із доведених вище результатів.

Зауважимо, що видалити жодну із функцій σ_{G} неможливо. Справді:

1) $=_{\mathsf{G}}$ є єдиним предикатом множини;

2) лише функція C_0^G не зберігає множину графів $G\setminus\left\{g=\left\langle V,E\right\rangle|\,V=\right.$ $\left.=\left\{v_1,\ldots\right\},v_1=1\right\}$ (множина графів, у яких номер першої вершини більше 1);

3) лише функція $\hat{\cup}$ не зберігає множину $\left\{g=\left\langle V,E\right\rangle|\,\left|V\right|\leq 2\right\}$ (графи з кількістю вершин меншою або рівною 2);

4) лише функція \ не зберігає множину $G\setminus\left\{g=\left\langle V,\mathrm{O}\right\rangle|\,\left|V\right|\geq 2\right\}$ (усі графи, окрім тих, що є повністю незв'язними і мають більше однієї вершини);

5) лише E_e не зберігає $G\setminus\left(\left\{g=\left\langle\left\{1,2\right\},\left\{\left\langle 1,2\right\rangle\right\}\right\rangle\right\}\cup\left\{g=\left\langle\left\{v_1,\ldots\right\},E\right\rangle|\,v_1=2\right\}\cup\right.$ $\left.\cup\left\{g=\left\langle\left\{1,2\right\},\mathrm{O}\right\rangle\right\}\right)$ (усі графи, окрім таких: графа, що складається із дуги $\left\langle 1,2\right\rangle$; незв'язного графа, що складається із двох вершин з номерами 1 та 2; множини графів, для яких номер першої вершини дорівнює 2);

6) лише E_v не зберігає $G\setminus\left(\left\{g=\left\langle\left\{2\right\},\mathrm{O}\right\rangle\right\}\cup\left\{g=\left\langle\left\{1,2\right\},\mathrm{O}\right\rangle\right\}\cup\right.$ $\left.\cup\left\{g=\left\langle V,\left\{e_1,\ldots\right\}\right\rangle|\,e_1=\left\langle 1,2\right\rangle\right\}\right)$ (усі графи, окрім таких: графа з однієї вершини з номером 2; графа із двох вершин з номерами 1 та 2; множини графів, у яких першим ребром є $\left\langle 1,2\right\rangle$);

7) лише D_v не зберігає $\left\{g=\left\langle\left\{v_1,\ldots\right\},E\right\rangle|\,v_1=1\right\}\cup\Delta_G$ (множина графів, для яких номер першої вершини дорівнює 1, та порожній граф);

8) нарешті, лише L не зберігає множину $\left\{g=\left\langle V,\mathrm{O}\right\rangle\right\}\cup\Delta_G$ (множина усіх повністю незв'язних графів і порожній граф).

Це дає змогу припустити, що вказана система функцій σ_{G} є базисом ППА графових перетворювачів, що зберігають денотати.

Висновок

У статті було досліджено породжуючу множину для ППА класу функцій, які зберігають денотати над графами. Діаграми мов моделювання (зокрема, UML), сукупності взаємозв'язаних і впорядкованих структур Bpwin, Erwin можуть бути представлені графовими засобами специфікації програм. Завданням цієї роботи є здобуття опису класу обчислюваних граф-функцій, які зберігають денотати, та пошуку природного для області графових перетворювачів I_m^n-базису, що, безперечно, є актуальною задачею. Декілька наступних робіт будуть присвячені практичному застосуванню отриманих результатів.

1. *Басараб І.А.* Базы данных с логико-функциональной точки зрения / І.А. Басараб, В.Н. Редько // Программирование. — 1984. — № 2. — С. 53–67.
2. *Мальцев А.И.* Алгоритмы и рекурсивные функции / А.И. Мальцев. — М.: Наука, 1965. — 391 с.
3. *Колмогоров А.Н.* К определению алгоритма / А.Н. Колмогоров, В.А. Успенский // Успехи матем. наук. — 1958. — Т. 13, № 4. — С. 3–28.
4. *Asser G.* Berechenbare Graphenabbildungen / G. Asser // In: Kompliziertheit von Lern-Und Erkennungsprozessen. — Jena: Friedrich-Schiller-Universitat, 1975. — P. 7–17.
5. *Заславский И.Д.* Граф-схемы с памятью / И.Д. Заславський // Тр. мат. ин-та АН СССР. — 1964. — **72**. — С. 99−192.
6. *Babai L.* Isomorphism of Graphs with Bounded Eigenvalue Multiplicity / L. Babai, D. Grigoryev, D. Mount // Proc. 14th ACM Symp. on Theory of Comput. — STOC, 1982. — P. 310–324.
7. *Пролубников А.В.* Прямой алгоритм проверки изоморфизма графов / А.В. Пролубников // Сб. научн. тр. «Компьютерная оптика». под ред. акад. РАН Ю.И. Журавлева. Из-во Самарского государственного университета, 2007. — Вып. 27. — С. 123–128.
8. *Foggia P.* A Database of Graphs for Isomorphism and Sub-Graph Isomorphism Benchmarking / P. Foggia, C. Sansone, M. Vento // Proc. of the 3-rd IAPR TC-15 International Workshop on Graph-Based Representations. — Italy, 2001. — P. 157–168.
9. *Spence E.* The Stromgly Regular (40, 12, 2,4) Graphs // The Electronical Journal Of Combinatorics. — 2000. — Vol. 7(1).
10. *Ершов А.П.* Вычислимость в произвольных областях и базисах// В кн.: Семиотика и информатика. — Вып. 19. — М.: ВИНИТИ, 1979. — С. 3–58.
11. *Агафонов В.Н.* Спецификация программ: понятийные средства и их организация / В.Н. Агафонов. — Новосибирск: Наука, 1987. — 240 с.
12. *Буй Д.Б.* Примитивные программные алгебры целочисленных и словарных функций / Д.Б. Буй, В.Н. Редько // Докл. АН УССР. Сер. А. — 1984. — № 10. — С. 69−71.
13. *Буй Д.Б.* К теории программных алгебр / Д.Б. Буй, А.В. Мавлянов // Укр. мат. журн. — 1984. — **36**, № 6 — С. 761−764.
14. *Буй Д.Б.* Примитивные программные алгебры: автореф. дис. канд. физ.-мат. наук. — К., 1985. — 22 с.
15. *Буй Д.Б.* Примитивные программные алгебры / Д.Б. Буй, В.Н. Редько // Кибернетика. — 1984. — № 5. — С. 1−7.
16. *Буй Д.Б.* Примитивные программные алгебры / Д.Б. Буй, В.Н. Редько // Кибернетика. — 1985. — № 1. — С. 28−33.
17. *Буй Д.Б.* Неперервність в індуктивних множинах: основні поняття та допоміжні результати / Д.Б. Буй // Вісник Київського Університету. — К., 1998. — С. 142–148. — (Серія: Фіз.-мат. науки; вип. 1).

18. *Буй Д.Б.* Неперервність в індуктивних множинах: неперервність суперпозиції та суміжні результати / Д.Б. Буй // Вісник Київського Університету. — К., 1998. — С. 187–195. — (Серія: Фіз.-мат. науки; вип. 2).

19. *Буй Д.Б.* Неперервність в індуктивних множинах: неперервність рекурсії та суміжні результати / Д.Б. Буй // Вісник Київського Університету. — К., 1998. — С. 128–138. — (Серія: Фіз.-мат. науки; вип. 3).

20. *Буй Д.Б.* Композиційна семантика SQL-подібних мов: табличні структури даних, композиції, приклади / Буй Д.Б., Поляков С.А. // Вісник Київського Університету. — К., 1999. — С. 130–140. — (Серія: Фіз.-мат. науки; вип. 1).

21. *Ершов Ю.Л.* Теория нумераций / Ю.Л. Ершов. — М.: Наука, 1977. — 416 с.

22. *Голунков Ю.В.* О полноте операций в системах алгоритмических алгебр / Ю.В. Голунков // Алгоритмы и автоматы. — Казань: Из-во Казан. ун-та, 1978. — С. 11−53.

23. *Новиков Ф.А.* Дискретная математика для программистов / Ф.А. Новиков. — СПб: Питер, 2000. — 304 с.

24. *Мальцев А.И.* Конструктивные алгебры / А.И. Мальцев // Успехи мат. наук. — 1961. — **16**, № 3. — С. 3−60.

Надійшла до редакції 19.11.2010