

ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ МАТЕМАТИЧНОГО ЗАБЕЗПЕЧЕННЯ ПРОЦЕСІВ ЗГОРТАННЯ ТА РОЗГОРТАННЯ ФОРМУЛ АЛГОРИТМІВ

© Василюк А. С., Басюк Т. М., 2017

Описано означення процесів згортання формул алгебри алгоритмів. Синтезовано, мінімізовано, побудовано та досліджено математичне забезпечення процесів згортання та розгортання формул алгебри алгоритмів.

Ключові слова: унітерм, алгоритм, згортання, розгортання, математична модель.

This article is about determination of process of collapsing and expanding of algebra algorithms formulas. Synthesized, minimized, and built mathematical model of adaptation process and expanding and collapsing processes.

Key words: unitherm, collapsing, expanding, mathematical model.

Постановка проблеми

Існує відома алгебра алгоритмів [1, 2]. Ця теорія оперує конкретними знаками операцій, такими як секвентування, елімінування, паралелення та циклічні операції. Вони відображаються у вигляді спеціальних знаків, яких немає серед відомих математичних символів. З метою спрощення процесів набору і редагування формул алгоритмів необхідно створити структуру даних та математичне забезпечення редактора формул алгебри алгоритмів. Одним з важливих процесів у створенні й редагуванні формул алгоритмів є процеси згортання і розгортання формул алгебри алгоритмів.

Аналіз останніх досліджень та публікацій

У роботах [3–6] описуються підходи до розв’язання описаних задач, проте не виконано дослідження математичного забезпечення процесів згортання і розгортання формул у редакторі абстрактних алгоритмів.

Незважаючи на актуальність задачі, сьогодні накопичено порівняно невеликий досвід її розв’язання, який визначається передусім доволі новим напрямом досліджень.

Формулювання цілей статті

Для оптимальнішого і наочнішого генерування формул алгебри алгоритмів необхідно, щоб математичне забезпечення описувало процеси згортання і розгортання формул алгебри алгоритмів. Ці процеси забезпечать наочність та компактність формул алгебри алгоритмів, що дасть змогу краще простежити структуру формул алгебри алгоритмів і значно полегшить процеси редагування і генерування формул редактором абстрактних алгоритмів, оскільки жодна з наявних систем не оперує засобами згортання і розгортання формул алгебри алгоритмів. Метою цієї роботи є аналіз процесів згортання і розгортання та синтез і мінімізація математичного забезпечення цих процесів.

Процеси згортання та розгортання формул

Процес згортання формул алгоритмів

Математичне забезпечення повинно описувати можливість згортання та розгортання формул абстрактних алгоритмів, що зменшує візуально формулу абстрактного алгоритму.

Процес згортання формули абстрактного алгоритму має виконуватись так:

– вибрати формулу абстрактного алгоритму;

– вибрати на панелі інструментів інструмент “Згорнути чи розгорнути формулу абстрактного алгоритму”.

Система автоматично виконає згортання формули абстрактного алгоритму.

Розгортання формули абстрактного алгоритму має виконуватись за такою послідовністю:

- вибрати необхідну формулу абстрактного алгоритму (у вигляді унітерму);
- вибрати на панелі інструментів інструмент “Згорнути чи розгорнути формулу абстрактного алгоритму”.

На прикладі проілюструємо процес згортання формули абстрактного алгоритму. Нехай дано формулу абстрактного алгоритму:

$$\left(\begin{array}{l} \text{Вираз 1} \\ , \\ \text{Вираз 2 ; Вираз 4 ; } u - ? \\ , \\ \text{Вираз 3} \end{array} \right) F_6 \quad (1)$$

де F_6 – знак операції елімінування.

Після виконання згортання знака операції елімінування отримаємо таку формулу:

$$\left(\begin{array}{l} \text{Вираз 1} \\ , \\ \#_F_6 \end{array} \right) \quad (2)$$

де $\#_F_6$ – умовне позначення формули згорнутого елімінування.

Згортання – процес перетворення формули алгоритму (формула F_6 в формулі (1)), яка складається з елементарних унітермів, до її умовного позначення (наприклад, унітерм $\#_F_6$ в формулі (2)).

Для синтезу математичного забезпечення першим кроком необхідно виконати синтез секвенцій і елімінувань [1].

Синтез секвенцій. Абстрактний алгоритм згортання формули передбачає процеси, які подані на рис. 1 і зводяться до:

- присвоєння назви згорнутій формулі $P_1(T, f_1(c_1, N))$, де T – змінна, якою описується назва згорнутої формули, c_1 – константа ознаки згорнутості формули, N – змінна, якою описується назва формули до згортання, $f_1(c_1, N)$ – функція створення унітерму зі згорнутої формули;
- присвоєння інкрементованого значення кількості унітермів в алгоритмі $P_1(i, f_2(i))$, де i – змінна, якою описується поточна кількість унітермів у алгоритмі, $f_2(i)$ – функція інкременту кількості унітермів у алгоритмі;
- присвоєння змінній унітерму (i_i) значення назви згорнутої формули $P_1(i_b, T)$;
- присвоєння координаті X_1 згорнутої формули значення координати x_1 до згортання $P_1(X_1, x_1)$;
- присвоєння координаті Y_1 згорнутої формули значення координати y_1 до згортання $P_1(Y_1, y_1)$;
- присвоєння координаті X_2 згорнутої формули значення нової координати з урахуванням назви згорнутої формули $P_1(X_2, f_3(x_1, i_i))$, де $f_3(x_1, i_i)$ – функція обчислення нової координати;
- присвоєння координаті Y_2 згорнутої формули значення нової координати з урахуванням назви згорнутої формули $P_1(Y_2, f_4(y_1, i_i))$, де $f_4(y_1, i_i)$ – функція обчислення нової координати;
- присвоєння змінній H , яка описує висоту згорнутої формули, обчислюваної висоти згорнутої формули $P_1(H, f_5(Y_1, Y_2))$, де $f_5(Y_1, Y_2)$ – функція обчислення за модулем висоти згорнутої формули;
- присвоєння змінній W , яка описує довжину згорнутої формули, обчислюваної довжини згорнутої формули $P_1(W, f_5(X_1, X_2))$, де $f_5(X_1, X_2)$ – функція обчислення за модулем довжини згорнутої формули;

- присвоєння порядковому номеру згорнутої формули (i_n) значення поточної кількості унітермів у алгоритмі $P_1(i_n, i)$;
- присвоєння згорнутій формулі типу “унітерм” $P_1(t, c_2)$, де t – змінна, що описує тип формули, а c_2 – константа типу “унітерм”;
- присвоєння зв’язків для згорнутої формули $P_1(F_i, F_{id})$, де F_i – порядковий номер базової формули для згорнутої формули, а F_{id} – порядковий номер базової формули для вибраної формули до згортання.

Після синтезу всіх секвенцій та елімінувань отримаємо такий абстрактний алгоритм згортання формул:

$$\begin{aligned}
 & \left(\begin{array}{l} P_1(T, f_1(c_p, N_p)) ; P_1(i_t, T) \\ ; \\ P_1(i, f_2(i)) \end{array} \right) \\
 & ; \\
 & \left(\begin{array}{l} P_1(X_1, x_1) ; \left(P_1(X_2, f_3(X_1, i_t)) \right) \\ ; \\ P_1(Y_1, y_1) ; \left(P_1(X_2, f_4(Y_1, i_t)) \right) \end{array} \right) \\
 & ; \\
 & \left(P_1(H, f_5(Y_1, Y_2)) ; P_1(W, f_5(X_1, X_2)) \right) \\
 & ; \\
 & \left(P_1(i_n, i) ; P_1(t, c_2) ; P_1(F_i, F_{id}) \right) \\
 & ; \\
 & \underbrace{\mathcal{Z}_{j=i}}_{\left(\begin{array}{l} P_1(A_j, i_n) ; P_1(B_j, i_n) ; u_1^-? \\ ; \\ P_1(X_n^{id}, c_3) \\ ; \\ P_1(Y_n^{id}, c_3) \\ ; \\ c_{j=1} \end{array} \right)} ; *K ; u_2^-?
 \end{aligned} \tag{3}$$

Процес розгортання формул алгоритмів

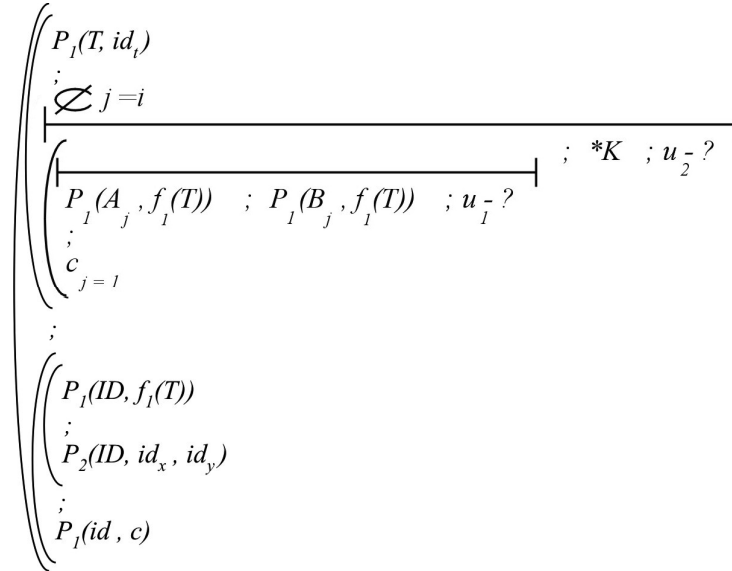
Розгортання – процес перетворення умовного позначення формули алгоритму (наприклад, унітерм #_F_6 у формулі (2)) до формули алгоритму (1).

Синтез секвенцій. Абстрактний алгоритм розгортання формули передбачає такі процеси:

- присвоєння назви згорнутої формули $P_1(T, id_t)$, де T – змінна, якій присвоюється назва згорнутої формули (id_t);
- присвоєння порядкового номера знайденої згорнутої формули $P_1(ID, f_i(T))$, де ID – порядковий номер знайденої згорнутої формули, $f_i(T)$ – функція пошуку порядкового номера формули за заданою назвою формули;

- переміщення розгорнутої формули в задану точку $P_1(ID, id_x, id_y)$, де id_x, id_y – нові координати розгорнутої формули;
- присвоєння нульової константи унітерму згорнутої формули $P_1(id, c)$, де id – порядковий номер згорнутої формули, c – нульова константа.

Після синтезу всіх секвенцій та елімінувань отримуємо такий абстрактний алгоритм розгортання формул абстрактних алгоритмів:



Математичне забезпечення процесів згортання та розгортання формул алгоритмів

Згортання формули алгоритму має виконуватись так:

- вибрати формулу алгоритму;
- вибрати на панелі інструментів інструмент “Згорнути чи розгорнути формулу абстрактного алгоритму”.

Система автоматично виконає згортання формули абстрактного алгоритму.

Розгортання формули алгоритму повинно виконуватись за такою послідовністю:

- вибрати необхідну формулу абстрактного алгоритму (у вигляді унітерму);
- вибрати на панелі інструментів інструмент “Згорнути чи розгорнути формулу абстрактного алгоритму”.

Математичне забезпечення процесу згортання формул алгоритмів

Побудуємо та дослідимо математичне забезпечення процесу згортання формул абстрактних алгоритмів. Математична модель абстрактного алгоритму будується заміною абстрактних унітермів предметними (конкретними) і заданням секвентних областей значень змінних та унітермів. Замінивши в абстрактному алгоритмі процесу згортання формул алгоритмів абстрактний унітерм предметним унітермом присвоєння $(P_1(T, f_1(c_1, N)))$ на $T=c_1+N$, $P_1(i, f_2(i))$ на $i=i+1$, $P_1(X_2, f_3(X_1, i_t))$ на $X_2=X_1+W_t$, де W_t – довжина назви згорнутої формули, $P_1(Y_2, f_4(Y_1, i_t))$ на $Y_2=Y_1+H_t$, де H_t – висота назви згорнутої формули, $P_1(H, f_5(Y_1, Y_2))$ на $H = |Y_1-Y_2|$ і т.д.), а абстрактний умовний унітерм u_1 предметним унітермом перевірки зв'язків з другою вкладеною формулою $(B_j=id)-?$ і абстрактний умовний унітерм u_2 предметним унітермом перевірки зв'язків з першою вкладеною формулою $(A_j=id)-?$ І, задавши секвентні області значень змінним, отримуємо модель абстрактного алгоритму процесу згортання формул. Дослідження опису моделлю процесу згортання формул виконано з використанням методу трансфінитної математичної індукції.

Модель, яка подана формулою, має такі змінні: $T, i, N_1, i_t, X_1, x_1, X_2, x_2, Y_1, y_1, Y_2, y_2, H, W, H_t, W_t, i_m, id, F_i, F_{id}, t, X_n^{id}, Y_n^{id}$. Як видно з формули, X_1, Y_1, X_2, Y_2, H, W обчислюються зі значень змінних $H_t, W_t, i, F_{id}, x_1, y_1, N_1$. Тому дослідження має бути виконане за цими змінними.

Нехай $H_t = a < p \hat{I} Q_3$, $W_t = b < q \hat{I} Q_3$, $x_t = d < r \hat{I} Q_1$, $y_t = e < s \hat{I} Q_2$, $i = m < u \hat{I} Q_4$, $F_{id} = o < z \hat{I} Q_5$, $N_t = l < g \hat{I} Q_6$ а формула описує процес згортання формул.

- Встановимо, чи моделлю описується процес згортання формули коли згорнута формула є другою формулою, вкладеною у базову ($B_j = id$), за назви згорнутої формули $N_t = l+1 = g$ та за значення змінної $x_t = d + 1 = r$.

$$\begin{array}{l}
 \left(T=c_1+N_t ; i=i+1 ; i_t=T \right) \\
 ; \\
 \left(\begin{array}{l} X_1=x_t \\ Y_1=y_t \end{array} ; \begin{array}{l} X_2=X_1+W_t \\ Y_2=Y_1+H_t \end{array} \right) \\
 ; \\
 \left(H=|Y_1-Y_2| ; W=|X_1-X_2| \right) \\
 ; \\
 \left(i_n=i ; t=c_2 \right) \\
 ; \\
 F_i=F_{id} \\
 ; \\
 \mathcal{Z}_{j=i} \\
 ; \\
 \left(\begin{array}{l} A_j=i_n ; \left(B_j=i_n ; * ; (B_j=id) - ? \right) ; (A_j=id) - ? \\ X_n^{id}=c_3 \\ Y_n^{id}=c_3 \end{array} \right) ; c_j
 \end{array}$$

У разі зміни значення координати x_t на 1 значення координати x_2 також змінюється на 1, в разі зміни порядкового номера згорнутої формули ($m=m+1$) на 1 значення порядкового номера другої вкладеної формули також змінюється на 1.

Математичне забезпечення процесу розгортання формул алгоритмів

Побудуємо та дослідимо модель процесу розгортання формул абстрактних алгоритмів.

Модель абстрактного алгоритму будується заміною абстрактних унітермів предметними (конкретними) і заданням секвентних областей значень змінних та унітермів.

Замінивши в абстрактному алгоритмі процесу розгортання формул алгоритмів абстрактний унітерм предметним унітермом присвоєння ($P_1(T, id)$ на $T=id$, $P_1(A_j, f_1(T))$ на $A_j=T_i$, де T_i – порядковий номер розгорнутої формули в результаті пошуку за назвою згорнутої формули, а абстрактний умовний унітерм u_1 предметним унітермом перевірки зв'язків з другою вкладеною формулою ($B_j=id$)-? і абстрактний умовний унітерм u_2 предметним унітермом перевірки зв'язків з першою вкладеною формулою ($A_j=id$)-? І, задавши секвентні області значень, змінним, отримаємо модель абстрактного алгоритму процесу розгортання формул.

Встановимо, чи моделлю описується процес розгортання формули коли згорнута формула є другою формулою, вкладеною у базову ($B_j = id$), за порядковим номером згорнутої формули $T_i = a+1 = p$. Отримаємо вираз:

$$\begin{pmatrix} T=id_i \\ ; \\ \mathcal{Z}_{j=i} \\ \left(\begin{array}{l} A_j=T_i ; \left| \begin{array}{l} B_j=T_i ; * ; (B_j=id) - ? \end{array} \right| ; (A_j=id) - ? \\ ; \\ c_j \end{array} \right) \\ ; \\ \left(\begin{array}{l} ID=T_i \\ ; \\ F(ID, id_x, id_y) \end{array} \right) \\ ; \\ id=c \end{pmatrix} \quad \begin{pmatrix} T = b \\ ; \\ B_j=a+1 \\ ; \\ ID=a+1 \\ ; \\ F(a+1, id_x, id_y) \\ ; \\ id=0 \end{pmatrix}$$

У разі зміни порядкового номера згорнутої формули ($a+1$) на 1 значення порядкового номера другої вкладеної формули також змінюється на 1 ($B_j = a + 1$).

На підставі трансфінитної математичної індукції стверджуємо, що формула описує розгортання формул алгоритмів для зміни змінних a . Аналогічно дослідження виконується для решти змінних.

Висновки

1. Виконано аналіз процесів згортання і розгортання формул алгебри алгоритмів.
2. Синтезовано абстрактні алгоритми процесів згортання і розгортання формул алгебри алгоритмів.
3. Синтезовано, мінімізовано і досліджено математичне забезпечення процесів згортання і розгортання формул алгебри алгоритмів.

1. Овсяк В., Бритковський В., Овсяк О., Овсяк Ю. Синтез і дослідження алгоритмів комп'ютерних систем. – Львів, 2004. – 276 с. 2. Овсяк В. АЛГОРИТМИ: методи побудови, оптимізації, дослідження вірогідності. – Львів: Світ, 2001. – 160 с. 3. Василюк А. С. Інтелектуальний аналіз структури даних математичного забезпечення редактора формул алгоритмів / А. С. Василюк // Вісн. Нац. ун-ту “Львівська політехніка” : Інформаційні системи та мережі. – 2015. – № 832 – С. 34–48. 4. Катренко А. В. Аналіз математичних моделей планування в мультипроектному середовищі / А. В. Катренко, А. С. Магац // Вісн. Нац. ун-ту “Львівська політехніка” : Інформаційні системи та мережі. – 2014. – № 783. – С. 443–450. 5. Овсяк В. Загальна модель редактора графічних унітернів / В. Овсяк, М. Козелко // Вісник ТНТУ. – 2013. – Т. 69. – № 1. – С. 183–192. 6. Василюк А. Абстрактний алгоритм редактора формул абстрактних алгоритмів “АбстрактАл” / А. Василюк // Комп'ютерні технології друкарства : зб. наук. пр. – Львів: УАД, 2006. – № 16. – С. 99–108.