ПРОГРАММИРОВАНИЕ, 2004, № 3, с. 1–10

УДК 681.3.06

ИСПОЛЬЗОВАНИЕ СИМВОЛЬНОГО ПОДХОДА К РАЗЛОЖЕНИЮ БЕРНШТЕЙНА ПРИ АНАЛИЗЕ И ОПТИМИЗАЦИИ ПРОГРАММ

© 2004 г. **Ф. Клосс*, И. Ю. Чупаева****

**ICPS/LSIIT, Université Louis Pasteur, Strasbourg*
Pôle API, Bd Sébastien Brant, 67400 Illkirch – France
INRIA Rocquencourt, A3 Project
78153 Le Chesnay Cedex – France
***Механико-математический факультет МГУ им. М.В. Ломоносова*
119992 Москва, Воробьевы горы
E-mail: clauss@icps.u-strasbg.fr, irina@icps.u-strasbg.fr, ladyirina@shade.msu.ru

Поступила в редакцию

В последние десятилетия были разработаны математические пакеты для статического анализа программ. Несмотря на то, что программные пакеты широко используются, они имеют ряд ограничений. В частности, полиномы от многих переменных с целочисленными коэффициентами, часто встречающиеся в программах и при анализе систем, не поддерживаются в таких системах. Некоторые методы уже были предложены, но, к сожалению, могут быть применены только к некоторому подклассу такого типа выражений. В данной статье предлагается более общий подход к методу, основанному на разложении Бернштейна. Этот метод позволяет производить анализ целочисленных полиномов от многих переменных с меньшими вычислительными затратами.

1. ВВЕДЕНИЕ

Статический анализ программ привлекает большое внимание в последние десятилетия в связи с тем, что был разработан соответствующий математический аппарат. Например, с помощью многогранников можно моделировать вложенные циклы, целочисленные точки которых ассоциируются с итерациями цикла. Используя эту модель, можно применять для численного анализа и оптимизирующих преобразований специальные математические методы. Однако применение этой модели ограничено теми циклами, границы которых являются аффинными функциями внешних переменных цикла, и ссылки на массив также являются аффинными функциями переменных цикла.

Линейность обсуждаемых функций является одним из основных ограничений для формального анализа. Многие модели, в которых рассматриваются взаимодействия между программным обеспечением и аппаратной частью компьютера, содержат нелинейные выражения. Компиляторы, использующие улучшенные оптимизирующие преобразования, имеют такие же ограничения. В частности, полиномы от многих переменных возникают во многих ситуациях, например, при линеаризации индексных функций, при индуктивном распознавании переменных, и компиляторы не могут оптимизировать такие выражения.

В данной работе теория разложения Бернштейна обобщается на параметризованные полиномиальные выражения от многих переменных.

Разложение Бернштейна позволяет установить границу области значения полиномов на параллелограмме [1–3]. Численные приложения были предложены для решения системы строгих полиномиальных неравенств, использующего этот метод [4, 5]. Наш подход рассматривает параметризованные полиномы от многих переменных, заданные на области, ограниченной полиномами от параметров. Можно построить достаточные условия на параметры для решений строгих параметризованных неравенств.

Полиномы Бернштейна являются базисом множества полиномов, ограниченных по степени, как линейного пространства. Таким образом, любой полином максимальной мультистепени, не большей N, заданный на единичном кубе $[0,1]^n$, где n есть число переменных, может быть разложен по этому базису с коэффициентами, которые могут быть вычислены в явном виде. Такие явные формулы позволяют провести вычисления и в символьном виде, вызывая дополнительный интерес к использованию этой теории. Еще одним интересным следствием является то, что необходимые вычисления достаточно простые. Более того, применение этой модели позволяет автоматически и без больших вычислительных затрат решать сложные проблемы при анализе программ.

Маслов и Пью представили в работе [6] технику упрощения полиномиальных связей. Она базируется на разложении произвольной полиномиальной связи на пересечение аффинных связей и связей в некоторых специальных формах, которые в дальнейшем могут быть линеаризованы. Таким образом, данный подход не является общим и может работать только с такими полиномами, которые могут быть приведены к описанному виду.

В работе [7] Блюм и Эйгенманн предлагают проверить зависимость между итерациями в цикле, при наличии нелинейных выражений, путем проверки, не перекрывается ли диапазон таких выражений для последовательных итераций. Однако эта техника применима только для выражений, монотонно возрастающих или монотонно убывающих по переменной цикла.

Данная статья организована следующим образом. В разделе 2 даются некоторые обозначения и определяются полиномы, которые будут рассматриваться в дальнейшем. Основные сведения о численном разложении Бернштейна напоминаются в разделе 3. Наш символьный подход представлен в разделе 4, где мы имеем дело с параметризованными полиномами от многих переменных, заданных на параметризованном базовом множестве. Программный пакет, реализующий данный подход, описан в разделе 5. Некоторые примеры применений даны в разделе 6: нахождение зависимостей между ссылками с линеаризованными индексами и исключение неисполняемого кода в условных выражениях. Заключения даны в разделе 7.

2. РАССМАТРИВАЕМЫЕ ПОЛИНОМЫ И ОБОЗНАЧЕНИЯ

Пусть $x = (x_1, \ldots, x_{l_x})$ обозначает l_x-мерный набор рациональных переменных, и $u = (u_1, \ldots, u_{l_u})$ – l_u-мерный набор рациональных параметров. Рассмотрим множества

$$Q^{l_x} = \{\bar{x} = (x_1, \ldots, x_{l_x}) \mid x_i \in Q,\ 1 \le i \le l_x\}$$
$$Q^{l_u} = \{\bar{u} = (u_1, \ldots, u_{l_u}) \mid u_i \in Q,\ 1 \le i \le l_u\}.$$

Любой терм $x_1^{i_1} x_2^{i_2} \ldots x_{l_x}^{i_{l_x}}$ обозначается как $\bar{x}^{I_x}$, где $I_x = (i_1, \ldots, i_{l_x})$ – набор l_x неотрицательных целых значений. Аналогично, $\bar{u}^{I_u}$ обозначает любой терм $u_1^{i_i} \ldots u_{l_u}^{l_u}$ с $I_u = (i_1, \ldots, i_{l_u})$. I_x и I_u называются мультииндексом переменной x и переменной u соответственно. Нулевой вектор $(0, \ldots, 0)$ обозначается $\bar{0}$ в $N_0^{l_x}$ или в $N_0^{l_u}$.

Определение 1. Максимальной мультистепенью полинома $p(\bar{u})$ от переменных $u = (u_1, \ldots, u_{l_u})$ называется l_u-мерный набор $N_u = (n_{u,1}, \ldots, n_{u,l_u})$, состоящий из максимальных степеней по каждой переменной полинома.

Определение 2. Для двух наборов $I = (i_1, \ldots, i_l)$ и $J = (j_1, \ldots, j_l)$ из множества наборов Q^l неравенство $I \le J$ используется в том смысле, что для любого $0 \le k \le l$ выполняется свойство $i_k \le j_k$.

Пусть $Q[\bar{u}]$ – множество полиномов, задаваемых следующим образом:

$$Q[\bar{u}] = \{p(\bar{u}) \mid p(\bar{u}) = \sum_{\bar{0} \le I_u \le N_u} a_{I_u} \cdot \bar{u}^{I_u}\},$$

с коэффициентами $a_{I_u} \in Q$. Также рассматривается множество полиномов от переменных x с коэффициентами из множества $Q[\bar{u}]$ с максимальной мультистепенью: $N_x = (n_{x,1}, \ldots, n_{x,l_x})$

$$Q[\bar{u}][\bar{x}] = \{p(\bar{x}, \bar{u}) \mid p(\bar{x}, \bar{u}) = \\ = \sum_{\bar{0} \le I_x \le N_x} p_{I_x}(\bar{u}) \cdot \bar{x}^{I_x},\ p_{I_x}(\bar{u}) \in Q[\bar{u}]\}.$$

Любой полином $p(\bar{x}, \bar{u})$ рассматривается на области определения $D_x \times D_u \subset Q^{l_x+l_u}$, где $D_x \subset Q^{l_x}$ и $D_u \subset Q^{l_u}$. Наша цель – найти целочисленные решения неравенства

$$p(\bar{x}, \bar{u}) > 0, \text{ где } (\bar{x}, \bar{u}) \in D_x \times D_u \cap Z^{l_x+l_u}.$$

Решение представляется в виде множеств D_0 и D_1, где $D \equiv (D_x \times D_u) = (D_0 \sqcup D_1 \sqcup D_\epsilon)$, таких, что полином $p(\bar{x}, \bar{u}) > 0$ для любого $(\bar{x}, \bar{u}) \in D_0 \cap \cap Z^{l_x+l_u}$, $p(\bar{x}, \bar{u}) \leq 0$ для любого $(\bar{x}, \bar{u}) \in D_1 \cap \cap Z^{l_x+l_u}$ и D_ϵ обозначает множество, на котором знак $p(\bar{x}, \bar{u})$ остался неизвестным.

В данной статье рассматривается обобщение численного метода Бернштейна в применении к параметризованным полиномам из множества $Q[\bar{u}][\bar{x}]$.

Определение 3. Базовым множеством $D = [a_1, b_1] \times \ldots \times [a_l, b_l]$ в пространстве Q^l назовем такое множество, что a_i или b_i являются полиномами от параметров или переменных, a_{q+1} и b_{q+1} при $q < l$ могут зависеть от предыдущих q переменных:

$$\begin{aligned}
&D = [a_2 = pa_2(x_1, \bar{u}), b_2 = pb_2(x_1, \bar{u})] \times \\
&\times [a_3 = pa_3(x_1, x_2, \bar{u}), b_3 = pb_3(x_1, x_2, \bar{u})] \times \ldots \\
&\ldots \times [a_l = pa_l(x_1, x_2, \ldots, x_{l-1}, \bar{u}), \\
&b_l = pb_l(x_1, x_2, \ldots, x_{l-1}, \bar{u})].
\end{aligned}$$

В случае полиномов нулевой степени базовое множество является параллелограммом.

Сначала напомним некоторые факты из численного метода Бернштейна.

3. ЧИСЛЕННЫЙ МЕТОД БЕРНШТЕЙНА

Этот метод детально рассмотрен в работах [3–5].

Пусть

$$\begin{aligned}
&C_N^K = C_{n_1}^{k_1} \cdot \ldots \cdot C_{n_l}^{k_l}, \\
&\text{где } \bar{0} \leq K = (k_1, \ldots, k_l) \leq N = (n_1, \ldots, n_l), \\
&(K, N) \in N_0^l, \\
&C_n^k = \frac{n!}{k!(n-k)!}, \ 0 \leq k \leq n, \ (k, n) \in N_0;
\end{aligned}$$

$$\begin{aligned}
&I - J = (i_1 - j_1, \ldots, i_l - j_l), \\
&\text{где } I = (i_1, \ldots, i_l), \ J = (j_1, \ldots, j_l), \ I, J \in N_0^l; \\
&\alpha^I = \alpha_1^{i_1} \cdot \ldots \cdot \alpha_l^{i_l}, \\
&\text{где } \alpha = (\alpha_1, \ldots, \alpha_l) \in R^l.
\end{aligned}$$

Для полиномов с постоянными коэффициентами

$$p(\bar{x}) = \sum_{\bar{0} \leq I \leq N} a_I \cdot \bar{x}^I \in R[\bar{x}], \ \bar{x} \in D \subset R^l, \ a_I \in R,$$

и максимальной мультистепенью N полиномы Бернштейна и коэффициенты Бернштейна определяются следующим образом.

Полином Бернштейна с индексом i степени n на единичном отрезке $[0,1]$ определяется как

$$b_{n,i}^{[0,1]}(x) = C_n^i \cdot x^i (1-x)^{n-i}, \ 0 \leq i \leq n, \ x \in R.$$

В случае многих переменных полином Бернштейна с индексом I мультистепени $N = (n_1, \ldots, n_l)$ на единичном кубе $[0,1]^l$, где $I = (i_1, \ldots, i_l)$, определяется как

$$\begin{aligned}
&B_{N,I}^{[0,1]^l}(\bar{x}) = b_{n_1,i_1}^{[0,1]}(x_1) \cdot \ldots \cdot b_{n_l,i_l}^{[0,1]}(x_l), \ \bar{0} \leq I \leq N, \\
&\bar{x} = (x_1, \ldots, x_l) \in R^l.
\end{aligned}$$

Полиномы Бернштейна определены на всем l-мерном пространстве R^l и являются базисом пространства полиномов $p(\bar{x})$ степени не большей, чем N, как линейного пространства:

$$p(\bar{x}) = \sum_{\bar{0} \leq I \leq N} c_{I,p}^{[0,1]^l} B_{N,I}^{[0,1]^l} = \sum_{\bar{0} \leq I \leq N} a_I \cdot \bar{x}^I,$$

где коэффициенты $c_I^{[0,1]^l}$ суть коэффициенты Бернштейна полинома $p(\bar{x})$ на единичном кубе $[0,1]^l$. Они могут быть вычислены следующим образом:

$$c_{I,p}^{[0,1]^l} = \sum_{\bar{0} \leq J \leq I} \frac{C_I^J}{C_N^J} a_J, \quad \bar{0} \leq I \leq N.$$

Коэффициенты Бернштейна имеют некоторые интересные свойства (см., например, [5]).

Теорема. *Пусть $p(\bar{x})$ – полином максимальной мультистепени N на единичном кубе $[0,1]^l$. Тогда для любого $\bar{x} \in [0,1]^l$ справедливо неравенство*

$$\min_{\bar{0} \leq I \leq N} c_{I,p}^{[0,1]^l} \leq p(\bar{x}) \leq \max_{\bar{0} \leq I \leq N} c_{I,p}^{[0,1]^l}.$$

Полиномы на произвольном базовом множестве $D \in R^l$ могут быть рассмотрены без потери общности, любое непустое базовое множество может быть отображено в единичный куб $[0,1]^l$

с помощью линейного преобразования. Пусть ϕ – линейное преобразование произвольного базового множества D в $[0,1]^l$. Так как полином $p(\bar{x})$ рассматривается на базовом множестве D, то соответствующий полином на единичном кубе $\tilde{p}(\bar{y}) = p(\phi^{-1}(\bar{y}))$ при $\bar{y} \in [0,1]^l$, и коэффициенты Бернштейна полинома p на базовом множестве D равны коэффициентам Бернштейна полинома $\tilde{p}$ на единичном кубе $[0,1]^l$: $c^D_{I,p} = c^{[0,1]^l}_{I,\tilde{p}}$.

Отображение ϕ определяется следующим образом:

$$\begin{array}{lll} \phi: & \bar{x} = (x_1, \ldots, x_l) & \to \ \bar{y} = (y_1, \ldots, y_l) \\ & [a_1, b_1] \times \ldots \times [a_l, b_l] & \to \ [0,1]^l, \ a_i, b_i \in R, \end{array}$$

где $y_i = \dfrac{x_i - a_i}{b_i - a_i}$, $0 \le i \le l$. Таким образом, обратное отображение имеет следующий вид:

$$\begin{array}{lll} \phi^{-1}: & \bar{y} = (y_1, \ldots, y_l) & \to \ \bar{x} = (x_1, \ldots, x_l), \\ & [0,1]^l & \to \ [a_1, b_1] \times \ldots \times [a_l, b_l], \\ & a_i, b_i \in R, & \end{array}$$

где $x_i = \alpha_i \cdot y_i + \beta_i$, $\alpha_i = (b_i - a_i)$, $\beta_i = a_i$, $0 \le i \le l$, $0 \le i \le l$.

Полином $\tilde{p}(\bar{y})$ выглядит так:

$$\tilde{p}(\bar{y}) = \sum_{\bar{0} \le P \le N} \tilde{a}_P \bar{y}^P,$$

где $\tilde{a}_P = \alpha^P \cdot \sum\limits_{P \le K \le N} a_K \cdot C^P_K \cdot \beta^{K-P}$. Заметим, что максимальная мультистепень N одинакова для полиномов $p(\bar{x})$ и $\tilde{p}(\bar{y})$.

Для применения метода Бернштейна к полиному на произвольном параллелограмме D строится отображение ϕ, и метод Бернштейна применяется к соответствующему полиному $\tilde{p}(\bar{y})$ на единичном кубе $[0,1]^l$.

Используя вышеописанную теорему, можно построить алгоритм нахождения решений неравенства $p(\bar{x}) > 0$ при $x \in [0,1]^l$. Используя свойство коэффициентов Бернштейна полинома на параллелограмме D и равенство $c^D_{I,p} = c^{[0,1]^l}_{I,\tilde{p}}$, для всех $\bar{0} \le I \le N$ получаем:

- если $\min\limits_{\bar{0} \le I \le N} c^D_{I,p} > 0$, то $p(x) > 0$ для всех $x \in D$;
- если $\max\limits_{\bar{0} \le I \le N} c^D_{I,p} \le 0$, то $p(x) \le 0$ для всех $x \in D$.

Полный алгоритм может быть найден в [5].

4. СИМВОЛЬНЫЙ ПОДХОД

Рассмотрим полиномы $p(\bar{x}, \bar{u})$ с параметрическими коэффициентами на базовом множестве с параметрическими границами.

4.1. *Параметризованные полиномы от многих переменных*

Преобразование произвольного полинома

$$p(\bar{x}, \bar{u}) = \sum_{\bar{0} \le I_x \le N_x} p_{I_x}(\bar{u}) \cdot \bar{x}^{I_x}, \ \bar{x} \in D_x, \ \bar{u} \in D_u,$$

в полином

$$\tilde{p}(\bar{y}, \bar{u}) = \sum_{\bar{0} \le I_x \le N_x} \tilde{p}_{I_x}(\bar{u}) \cdot \bar{y}^{I_x}, \ \bar{y} \in [0,1]^{l_x}, \ \bar{u} \in D_u,$$

может быть найдено также и в случае параметрических коэффициентов $p_{I_x}(\bar{u})$: формулы выглядят так же, то есть

$$\tilde{p}_{I_x}(\bar{u}) = \alpha^{I_x} \cdot \sum_{I_x \le K_x \le N_x} p_{K_x}(\bar{u}) \cdot C^{I_x}_{K_x} \cdot \beta^{K_x - I_x},$$

где, как и выше, $D_x = [a_1, b_1] \times \ldots \times [a_{l_x}, b_{l_x}]$, $\alpha_i = (b_i - a_i)$, $\beta_i = a_i$, $0 \le i \le l_x$. Значит, параметрические коэффициенты Бернштейна могут быть вычислены следующим образом:

$$c^{[0,1]^l}_{I_x} = \sum_{\bar{0} \le J \le I_x} \frac{C^J_{I_x}}{C^J_{N_x}} \tilde{p}_J(\bar{u}), \ \bar{0} \le I_x \le N_x,$$

и неравенство

$$p(\bar{x}, \bar{u}) > 0, \ \bar{x} \in D_x, \ \bar{u} \in D_u,$$

сохраняется, если $\min\limits_{\bar{0} \le I_x \le N_x} c^{D_x}_{I_x} > 0$. Но так как коэффициенты Бернштейна также являются полиномами, зависящими от параметров $\bar{u}$, минимальный из этих коэффициентов трудно определить в общем случае. Но положительность минимального коэффициента Бернштейна эквивалентна положительности всех коэффициентов Бернштейна. Тогда можно рассмотреть следующую систему:

$$c^{D_x}_{I_x} > 0 \text{ для любого } \bar{0} \le I_x \le N_x.$$

Более того, если коэффициенты полинома $p_{I_x}(\bar{u})$ являются линейными функциями от параметров $\bar{u}$, коэффициенты Бернштейна полинома p на базовом множестве D_x также являются линейными функциями от тех же параметров, и последняя система суть система линейных неравенств. Такие системы могут быть решены с использованием средств работы с линейными функциями, такими, как, например, реализованные в библиотеке работы с многогранниками *PolyLib*.

Пример 1. Рассмотрим полином с максимальной мультистепенью $N_x = (1,1)$ $p(\bar{x}, \bar{u}) = (N+M)x_1x_2 + Nx_1 + Mx_2$, где $\bar{x} = (x_1, x_2)$, $\bar{u} = (N, M)$ и $\bar{x} \in D_x = [0,100] \times [0,100]$. Базовое множество D_x отображается на единичный куб $[0,1]^2$ при отображении ϕ, задаваемом выражением $\phi(\bar{x}) = \bar{y} = (y_1 = x_1/100, y_2 = x_2/100)$. Таким образом, получаем полином $\tilde{p}(\bar{y}, \bar{u}) = (10000N + 10000M)y_1y_2 + 100Ny_1 + 100My_2$, определенный при $\bar{y} \in [0,1]^2$. Коэффициенты Бернштейна вычисляются для всех мультииндексов $(0,0) \le I_x \le N_x = (1,1)$:

- для $I_x = (0,0)$: $c^{D_x}_{(0,0)} = \dfrac{C_0^0 C_0^0}{C_1^0 C_1^0} 0 = 0$;
- для $I_x = (0,1)$: $c^{D_x}_{(0,1)} = \dfrac{C_0^0 C_1^0}{C_1^0 C_1^0} 0 + \dfrac{C_0^0 C_1^1}{C_1^0 C_1^1} \times 100M = 100M$;
- для $I_x = (1,0)$: $c^{D_x}_{(1,0)} = \dfrac{C_0^0 C_1^0}{C_1^0 C_1^0} 0 + \dfrac{C_1^1 C_0^0}{C_1^1 C_1^0} \times 100N = 100N$;
- для $I_x = (1,1)$: $c^{D_x}_{(0,1)} = \dfrac{C_1^0 C_1^0}{C_1^0 C_1^0} 0 + \dfrac{C_1^0 C_1^1}{C_1^0 C_1^1} \times 100M + \dfrac{C_1^1 C_1^0}{C_1^1 C_1^0} 100N + \dfrac{C_1^1 C_1^1}{C_1^1 C_1^1}(10000N + 10000M) = 10100N + 10100M$.

Неравенство $p(\bar{x}, \bar{u}) \ge 0$ выполняется, когда все коэффициенты Бернштейна положительны. В данном примере это неравенство может быть легко приведено к условию $N \ge 0$ и $M \ge 0$.

4.2. *Параметрическое базовое множество*

Рассмотрим базовое множество $D_x = [a_1, b_1] \times \ldots \times [a_l, b_l]$ такое, что a_i и b_i являются полиномами от параметров и переменных, a_{q+1} и b_{q+1} при $q < l$ зависят от предыдущих q переменных:

$$\begin{aligned} &D_x = [a_2 = pa_2(x_1, \bar{u}), b_2 = pb_2(x_1, \bar{u})] \times \\ &\times [a_3 = pa_3(x_1, x_2, \bar{u}), b_3 = pb_3(x_1, x_2, \bar{u})] \times \ldots \\ &\ldots \times [a_l = pa_l(x_1, x_2, \ldots, x_{l-1}, \bar{u}), \\ &b_l = pb_l(x_1, x_2, \ldots, x_{l-1}, \bar{u})], \end{aligned}$$

где pa_i и pb_i – полиномы от некоторых переменных и параметров. В этом случае отображение ϕ при трансформации базового множества D_x в единичный куб определяется формулами от переменных и параметров: $y_i = \dfrac{x_i - a_i}{b_i - a_i}$, где a_i и b_i – в общем случае полиномы от переменных и параметров. Таким образом, ϕ определено тогда и только тогда, когда $b_i - a_i \neq 0$. Это означает, что $a_i < b_i$ при $a_i \le b_i$. То есть допустимое отображение на единичный куб может быть определено только для такого D_x, где параметры и переменные не принимают значения, удовлетворяющие равенству $b_i - a_i = 0$. Обозначим такое множество D_x^*:

$$\begin{aligned} &D_x^* = D_x - \{\bar{x} \in D_x \mid \exists\, 1 \le i \le l, \\ &x_i = Root\_of(a_i - b_i)\}. \end{aligned}$$

Когда a_i и b_i – аффинные функции от переменных и параметров, корни $(a_i - b_i)$ могут быть явно вычислены. В остальных случаях корни могут быть явно вычислены для полиномов максимальной степени 4. Выделяя точки, соответствующие корням полиномов, в множество особых точек, получаем вместо изначального базового множества объединение непересекающихся множеств, для каждого из которых можно построить преобразование в единичный куб и провести анализ полинома на нем по методу Бернштейна.

Специальный анализ полинома в особых точках, являющихся корнями полиномов $(a_i - b_i)$, проводится отдельно.

Пример 2. Рассмотрим полином $p(i,j) = (i^2 + i)/2 + j$, определенный на базовом множестве $D_x = [0, N] \times [0, i]$. Преобразование ϕ задается следующим образом: $\phi(i,j) = (i' = i/N, j' = j/i)$, так что оно не определено в точках $N = 0$ и $i = 0$. Таким образом, точка $(0,0)$ является особой, и базовое множество суть D_x: $D_x^* = D_x - \{(0,0)\} = [1, N] \times [0, i]$.

На базовом множестве D_x^* полином $p(i,j)$ трансформируется в полином $\tilde{p}(i', j') = (N^2 i'^2 + Ni')/2 + ij'$ при $(i', j') \in [0,1] \times [0,1]$.

Символьный подход используется для получения достаточных условий положительности или отрицательности любого параметризованного полинома от многих переменных. Мы приведем в разделе 6 некоторые примеры того, как этот метод позволяет автоматически и с небольшими вычислительными затратами решать более сложные проблемы анализа программ.

5. РЕАЛИЗАЦИЯ

Символьный подход, описанный выше, реализован для полиномов с рациональными коэффициентами. Это приложение использует библиотеку GNU-MP для арифметики произвольной точности.

Программа получает на вход форматированные данные из стандартного ввода и выдает результат в стандартный выход. В будущем планируется интегрировать наше приложение с библиотекой работы с полиномами *PolyLib*, чтобы иметь возможность напрямую решать системы линейных неравенств и расширить возможности библиотеки *Polylib*.

Основные шаги алгоритма являются следующими:

- ввести данные со стандартного входа или из файла: базовые множества D_x, D_u, полином $p(\bar{x}, \bar{u})$;
- преобразовать данные во внутренний формат и инициализировать задачу T;
- посчитать коэффициенты полинома $\tilde{p}(\bar{y}, \bar{u})$ на $[0,1]^{l_x} \times D_u$;
- посчитать коэффициенты Бернштейна полинома $\tilde{p}(\bar{y}, \bar{u})$ на единичном кубе;
- построить систему полиномиальных неравенств;
- вывести результат.

Покажем на примере, как работает программа.

Пример 3. Необходимо решить неравенство

$$P(\bar{x}, \bar{u}) = (5 + u_1 u_2^2 + 2u_1^2 u_2^3 u_3) + 2x_3 + \\ + \left(\frac{2}{3} + \frac{1}{2} u_1 u_2^2 u_3 + \frac{1}{3} u_1^2 u_2^3 u_3\right) x_2 > 0$$

на базовом множестве $D_x = [0,1] \times [0,1] \times [0,1]$ и $D_u = [0,1] \times [0,1] \times [0,1]$. На выходе получаем систему полиномиальных неравенств

```
+5+1*u1^1*u2^2*u3^0+2*u1^2*u2^3*u3^1>0
+7+1*u1^1*u2^2*u3^0+2*u1^2*u2^3*u3^1>0
+47/9+1*u1^1*u2^2*u3^0+1/6*u1^1*u2^2*u3^1
+19/9*u1^2*u2^3*u3^1>0
+65/9+1*u1^1*u2^2*u3^0+1/6*u1^1*u2^2*u3^1
+19/9*u1^2*u2^3*u3^1>0
+49/9+1*u1^1*u2^2*u3^0+1/3*u1^1*u2^2*u3^1
+20/9*u1^2*u2^3*u3^1>0
+67/9+1*u1^1*u2^2*u3^0+1/3*u1^1*u2^2*u3^1
+20/9*u1^2*u2^3*u3^1>0
+17/3+1*u1^1*u2^2*u3^0+1/2*u1^1*u2^2*u3^1
+7/3*u1^2*u2^3*u3^1>0
+23/3+1*u1^1*u2^2*u3^0+1/2*u1^1*u2^2*u3^1
+7/3*u1^2*u2^3*u3^1>0
+5+1*u1^1*u2^2*u3^0+2*u1^2*u2^3*u3^1>0
+7+1*u1^1*u2^2*u3^0+2*u1^2*u2^3*u3^1>0
+47/9+1*u1^1*u2^2*u3^0+1/6*u1^1*u2^2*u3^1
+19/9*u1^2*u2^3*u3^1>0
+65/9+1*u1^1*u2^2*u3^0+1/6*u1^1*u2^2*u3^1
+19/9*u1^2*u2^3*u3^1>0
+49/9+1*u1^1*u2^2*u3^0+1/3*u1^1*u2^2*u3^1
+20/9*u1^2*u2^3*u3^1>0
+67/9+1*u1^1*u2^2*u3^0+1/3*u1^1*u2^2*u3^1
+20/9*u1^2*u2^3*u3^1>0
+17/3+1*u1^1*u2^2*u3^0+1/2*u1^1*u2^2*u3^1
+7/3*u1^2*u2^3*u3^1>0
+23/3+1*u1^1*u2^2*u3^0+1/2*u1^1*u2^2*u3^1
+7/3*u1^2*u2^3*u3^1>0
+5+1*u1^1*u2^2*u3^0+2*u1^2*u2^3*u3^1>0
+7+1*u1^1*u2^2*u3^0+2*u1^2*u2^3*u3^1>0
+47/9+1*u1^1*u2^2*u3^0+1/6*u1^1*u2^2*u3^1
+19/9*u1^2*u2^3*u3^1>0
+65/9+1*u1^1*u2^2*u3^0+1/6*u1^1*u2^2*u3^1
+19/9*u1^2*u2^3*u3^1>0
+49/9+1*u1^1*u2^2*u3^0+1/3*u1^1*u2^2*u3^1
+20/9*u1^2*u2^3*u3^1>0
+67/9+1*u1^1*u2^2*u3^0+1/3*u1^1*u2^2*u3^1
+20/9*u1^2*u2^3*u3^1>0

+17/3+1*u1^1*u2^2*u3^0+1/2*u1^1*u2^2*u3^1
+7/3*u1^2*u2^3*u3^1>0
+23/3+1*u1^1*u2^2*u3^0+1/2*u1^1*u2^2*u3^1
+7/3*u1^2*u2^3*u3^1>0
on the u-BOX:[0,1]x[0,1]x[0,1]
```

Получена система из 24-х неравенств, зависящих только от u-переменных. Если эта система имеет решения, то при этих значениях параметров первоначальное неравенство выполняется на всем базовом множестве D_x.

6. ПРИЛОЖЕНИЯ

При тестировании на зависимость ссылок по линейным индексам обычно применяются различные математические методы. В том числе могут быть применены и методы, описанные выше. Рассмотрим некоторые примеры, данные в работе Маслова и Пью [6].

Пример 4. Первый пример, показанный на рис. 1, представляет собой тело цикла в программе BOAST, являющейся частью пакета RiCEPS. Тело цикла такого типа довольно часто встречается в программах.

```
do i=p,p+L-1
  do j=q,q+M-1
    do k=r,r+N-1
      A(M*N*i+N*j+k) = ...
      ... = A(M*N*i+N*j+k)
    enddo
  enddo
enddo
```

Рис. 1. Первый типичный пример тела цикла с линеаризованными ссылками.

Чтобы иметь возможность распараллелить или преобразовать произвольный цикл по переменным `i`, `j` и `k`, необходимо проверить, зависят ли данные второго выражения от первого в независимых ветках цикла. Компилятор не может определить эту зависимость в независимых ветках цикла, и существующая техника анализа зависимостей в таких полиномиальных выражениях не настолько универсальна, как та, что обсуждается далее.

Рассмотрим пару итераций (i_1, j_1, k_1) и $(i_1+\alpha, j_2, k_2)$. Если $MNi_1 + Nj_1 + k_1 < MN(i_1+\alpha) + Nj_2 + k_2$ для любого $\alpha \geq 1$, то $MNi_1+Nj_1 + k_1$ никогда не будет равным $MN(i_1+\alpha) + Nj_2 + k_2$. Значит, эти итерации независимы по переменной цикла i.

Используем наш метод для получения достаточных условий, когда $p(\bar{x}, \bar{u}) = MNi_1 + Nj_1+k_1 - (MN(i_1 + \alpha) + Nj_2 + k_2) = Nj_1 + k_1-Nj_2-k_2-MN\alpha < 0$ при $\bar{x} = (j_1, k_1, j_2, k_2)$ и $\bar{u} = (\alpha, N, M, p, L, q, r)$. Переменные рассматриваются на базовом множестве $D_x = [q, q + M - 1]\times[r, r+N-1]\times[q, q+M-1]\times[r, r+N-1]$. Отображение ϕ, которое трансформирует базовое множество D_x в единичный куб, определяется формулами $\bar{y} = \left(\frac{j_1 - q}{M-1}, \frac{k_1 - r}{N-1}, \frac{j_2 - q}{M-1}, \frac{k_2 - r}{N-1}\right)$, которые определены для любых значений переменных и для любых значений параметров строго больших единицы. Оно преобразует полином $p(\bar{x}, \bar{u})$ в $\tilde{p}(\bar{y}, \bar{u})$:

$$\begin{aligned}\tilde{p}(\bar{y}, \bar{u}) &= N((M-1)j_1' + q) + (N-1)k_1' + r-\\ &\quad -N((M-1)j_2' + q) - ((N-1)k_2' + r)-\\ &\quad -MN\alpha = (NM-N)j_1' + (N-1)k_1'-\\ &\quad -(NM-N)j_2' - (N-1)k_2' - MN\alpha,\end{aligned}$$

где $\bar{y} = (j_1', k_1', j_2', k_2') \in [0,1]^4$, $M > 1$ и $N > 1$. Используя нашу программу для вычисления коэффициентов Бернштейна полинома $p(\bar{y}, \bar{u})$, получим следующие достаточные условия отрицательности полинома:

$$p(\bar{x}, \bar{u}) < 0, \text{ если } \begin{cases} NM\alpha > 0, \\ NM\alpha + N - 1 > 0, \\ NM\alpha + NM - N > 0, \\ NM\alpha + NM - 1 > 0, \\ NM\alpha - N + 1 > 0, \\ NM\alpha + NM - 2N + 1 > 0, \\ NM\alpha - NM + N > 0, \\ NM\alpha - NM + 2N - 1 > 0, \\ NM\alpha - NM + 1 > 0. \end{cases}$$

Можно легко проверить, что эта система неравенств эквивалентна неравенству $\alpha \geq 1$, так как рассматриваются только целочисленные значения. Таким образом, необходимо доказать, что это выражение не зависит от переменной цикла i, по которой вычисления можно распараллелить.

Заметим, что, если значение α_{min} больше значения, полученного для α при решении системы неравенств, это значение может быть использовано для определения циклического распределения между α_{min} процессорами в целях выделения зависимых итераций на том же процессоре. Этот результат можно применить при раскрытии цикла α_{min} раз для распараллеливания.

Пример 5. Пример, показанный на рис. 2, является телом цикла доступа к одномерному массиву A, который является линеаризованной версией треугольной матрицы: ссылка на $A(i, j)$ вычисляется как $A(i*(i+1)/2+j)$. Аналогично предыдущему примеру можно проверить, есть

```
do i=0,N-1
  do j=0,i
    A(i*(i+1)/2+j) = ...
  enddo
  do j=0,i
    ... = A(i*(i+1)/2+j)
  enddo
enddo
```

Рис. 2. Второй пример тела цикла с линеаризованными ссылками на треугольную матрицу.

```
do i=2,N
  do j=1,i-1
    if 4*i*i+j >= 3*i*j+5*i+1
      ...
    else
      ...
    endif
  enddo
enddo
```

Рис. 3. Третий пример со сложным условным выражением, которое может быть исключено.

ли зависимость данных в разных итерациях цикла. В этом примере переменные используются и при задании базового множества.

Рассмотрим пару итераций (i_1, j_1) и $(i_1+\alpha, j_2)$. Необходимо доказать, что

$$i_1(i_1+1)/2 + j_1 < (i_1+\alpha)((i_1+\alpha)+1)/2 + j_2$$

для любого $\alpha \geq 1$. Можно найти достаточные условия для неравенства $p(\bar{x}, \bar{u}) < 0$, где $\bar{x} = (i_1, j_1, j_2)$, $\bar{u} = (\alpha, N)$ и $p(\bar{x}, \bar{u}) = -\alpha i_1 + j_1 - j_2 - \dfrac{\alpha^2+\alpha}{2}$. Переменные рассматриваются на базовом множестве $D_x = [0, N-1] \times [0, i_1] \times [0, i_1+\alpha]$. Преобразование ϕ для отображения базового множества D_x на единичный куб определяется как $\bar{y} = (\dfrac{i_1}{N-1}, \dfrac{j_1}{i_1}, \dfrac{j_2}{i_1+\alpha})$. Это преобразование не определено при $i_1 = 0$ и $N = 1$. Таким образом, точки с $(0,0,0)$ по $(0,0,\alpha)$ должны рассматриваться отдельно, и N должно быть строго больше, чем два: $D_x^* = [1, N-1] \times [0, i_1] \times [0, i_1+\alpha]$. Отображение ϕ переопределяем как $\bar{y} = \left(\dfrac{i_1-1}{N-2}, \dfrac{j_1}{i_1}, \dfrac{j_2}{i_1+\alpha}\right)$.

Отображение ϕ преобразует полином $p(\bar{x}, \bar{u})$ в $\tilde{p}(\bar{y}, \bar{u})$:

$$\begin{aligned}\tilde{p}(\bar{y}, \bar{u}) &= -\alpha((N-2)i_1'+1) + i_1 j_1' - \\ &\quad -(i_1+\alpha)j_2' - \frac{\alpha^2+\alpha}{2} = \\ &= -\alpha((N-2)i_1'+1) + ((N-2)i_1'+1)j_1' - \\ &\quad -(((N-2)i_1'+1)+\alpha)j_2' - \frac{\alpha^2+\alpha}{2} = \\ &= (N-2)i_1'j_1' - (N-2)i_1'j_2' - \alpha(N-2)i_1' + \\ &\quad + j_1' - (\alpha+1)j_2' - \frac{\alpha(\alpha+3)}{2},\end{aligned}$$

где $\bar{y} = (i_1', j_1', j_2') \in [0,1]^3$ и $N > 2$. Вычисляя коэффициенты Бернштейна полинома $p(\bar{y}, \bar{u})$, получаем следующую систему:

$$p(\bar{x}, \bar{u}) < 0, \text{ если } \begin{cases} \dfrac{\alpha(\alpha+3)}{2} > 0, \\ \dfrac{\alpha(\alpha+5)+2}{2} > 0, \\ \dfrac{\alpha(\alpha+3)-2}{2} > 0, \\ \dfrac{\alpha(\alpha+5)}{2} > 0, \\ \dfrac{\alpha(\alpha+2N-1)}{2} > 0, \\ \dfrac{\alpha(\alpha+2N+1)+2N-2}{2} > 0, \\ \dfrac{\alpha(\alpha+2N-1)-2N+2}{2} > 0, \\ \dfrac{\alpha(\alpha+2N+1)}{2} > 0. \end{cases}$$

Можно легко проверить, что все эти условия выполняются при $\alpha \geq 1$. В особой точке $i_1 = j_1 = 0$ полином $p(\bar{x}, \bar{u})$ отрицателен. Таким образом, выражения в этих итерациях не зависят от переменной цикла i.

Теперь рассмотрим пример, показывающий применимость нашего математического аппарата для удаления неисполняемого кода из условных операторов.

Пример 6. Рассмотрим тело цикла, приведенного на рис. 3.

Оно содержит сложное условное выражение, которое может зависеть от переменных цикла. Мы используем вышеописанные методы для проверки, является ли условие всегда верным или всегда ложным, чтобы исключить неисполняемые куски кода из программы.

Неравенство $4i^2 + j \geq 3ij + 5i + 1$ эквивалентно неравенству $p(\bar{x}, \bar{u}) = 4i^2 - 3ij - 5i + j - 1 \geq 0$. Получим $\bar{x} = (i, j)$, $\bar{u} = N$ и $D_x = [2, N] \times [1, i-1]$. Отображение ϕ задается как $\bar{y} = \left(\frac{i-2}{N-2}, \frac{j-1}{i-2}\right)$, оно определено только при N строго большем двух и для любых точек, кроме точки $(2, 1)$: $D_x^* = [3, N] \times [1, i-1]$. Значит, N должно быть строго больше, чем три, и ϕ переопределяется как $\bar{y} = \left(\frac{i-3}{N-3}, \frac{j-1}{i-2}\right)$. Вычислим $\tilde{p}(\bar{y}, \bar{u})$:

$$\begin{aligned}
\tilde{p}(\bar{y}, \bar{u}) &= 4((N-3)i'+3)^2 - 3((N-3)i'+3)\times \\
&\quad \times((i-2)j'+1) - 5((N-3)i'+3)+ \\
&\quad +((i-2)j'+1) - 1 = \\
&= 4((N-3)i'+3)^2 - 3((N-3)i'+3)\times \\
&\quad \times((((N-3)i'+3)-2)j'+1)- \\
&\quad -5((N-3)i'+3)+ \\
&\quad +(((N-3)i'+3)-2)j' = \\
&= (-24N+36+4N^2)i'^2 + (-27-3N^2+ \\
&\quad +18N)i'^2j' + (-48+16N)i'+ \\
&\quad +(-11N+33)i'j' - 8j' + 12,
\end{aligned}$$

где $\bar{y} = (i', j') \in [0, 1]^2$ и $N > 3$. Посчитаем коэффициенты Бернштейна полинома $\tilde{p}(\bar{y}, \bar{u})$ и получим систему неравенств, являющуюся достаточными условиями положительности полинома на базовом множестве:

$$p(\bar{x}, \bar{u}) > 0, \text{ если } \begin{cases} 12 > 0, \\ 4 > 0, \\ 8N - 12 > 0, \\ \dfrac{5N-7}{2} > 0, \\ 4N^2 - 8N > 0, \\ N^2 - N - 2 > 0. \end{cases}$$

Легко проверить, что все эти условия всегда выполняются при $N > 2$. Для особой точки $(2, 1)$ получаем, что $p(\bar{x}, \bar{u}) = 0$. Таким образом, полином $p(\bar{x}, \bar{u})$ всегда неотрицателен на D_x, и выражение в случае *else*-части кода может быть удалено.

7. ВЫВОДЫ

Было показано, что символьный подход к использованию разложения Бернштейна для анализа полиномов от многих переменных имеет несколько преимуществ:

- внутренние вычисления не требуют больших вычислительных затрат;
- данный подход работает с произвольными полиномами от многих переменных безо всяких ограничений;
- сложность полученной на выходе системы неравенств напрямую зависит от сложности параметризованных коэффициентов. Если эти коэффициенты являются линейными комбинациями параметров, то выходная система линейных неравенств может быть достаточно просто решена.

Однако при применении этого метода мы сталкиваемся с двумя основными проблемами:

- как результат работы программы, может быть получена довольно сложная система полиномиальных неравенств от многих переменных, из которой нельзя будет быстро сделать вывод о том, всегда ли выполняется наше изначальное неравенство или нет, и в каких случаях оно может выполняться;
- может возникнуть ситуация, когда нельзя заключить, является ли наш полином положительным или отрицательным на всем базовом множестве D_x, так как полученная система неравенств не будет иметь решений.

В первом случае предлагается еще раз применить этот метод к полиномиальной системе, полученной на выходе, путем разделения параметров на 2 типа переменных – переменные и параметры новой системы. Решения, найденные этим методом, также могут быть системой полиномиальных уравнений. После нескольких таких итераций решения всей системы должны быть найдены. Сейчас мы занимаемся исследованием этого подхода.

Во втором случае изначальное базовое множество D_x должно быть разделено на подмножества для того, чтобы найти решение на каждом из них. Гарлофф в работе [4] представил процедуру деления параллелограмма одним из возможных способов. Но так как мы рассматриваем параметризованные базовые множества, эта процедура не может быть применена в нашем случае.

При рассмотрении нелинейных выражений в программах применение приближенных методов становится все более и более распространенным при трассировке программ, эвристических оценках, грубых приближениях, динамическом анализе в реальном времени и т.д.

Мы выражаем благодарность Е.В. Панкратьеву за ценные замечания, сделанные в процессе подготовки статьи.

СПИСОК ЛИТЕРАТУРЫ

1. *Бернштейн С.Н.* Собрание сочинений. Т. 1. Академия наук СССР, 1952.
2. *Бернштейн С.Н.* Собрание сочинений. Т. 2. Академия наук СССР, 1954.
3. *Berchtold J., Bowyer A.* Robust arithmetic for multivariate Bernstein-form polynomials // Computer-aided Design. 2000. V. 32. P. 681–689.
4. *Garloff J.* Application of Bernstein Expansion to the Solution of Control Problems / Vehi J., Sainz M.A. (eds.) Proceedings of MISC'99 – Workshop on Applications of Interval Analysis to Systems and Control. University of Girona. Girona, Spain. P. 421–430.
5. *Garloff J., Graf B.* The Use of Symbolic Methods in Control System Analysis and Design. Solving Strict Polynomial Inequalities by Bernstein Expansion. London: Institution of Electrical Engineers (IEE), 1999. P. 339–352.
6. *Maslov V., Pugh W.* Simplifying Polynomial Constraints Over Integers to Make Dependence Analysis More Precise. Proc. CONPAR 94 – VAPP VI. Int. Conf. on Parallel and Vector Processing. 1994.
7. *Blume W., Eigenmann R.* Non-Linear and Symbolic Data Dependence Testing // IEEE Transactions on Parallel and Distributed Systems. 1998. V. 9. № 12. P. 1180–1194.
8. *Chatfield C.* Time-Series Forecasting. Chapman & Hall, 2000.
9. *Feautrier P.* The Data Parallel Programming Model. Automatic Parallelization in the Polytype Model // LNCS. 1996. V. 1132. P. 79–100.
10. *Ernst M.D., Cockrell J., Griswold W.G., Notkin D.* Dynamically Discovering Likely Program Invariants to Support Program Evolution. Proc. International Conference on Software Engineering. 1999. P. 213–224.
11. *Chilimbi T.M., Hill M.D., Larus J.R.* Cache-Conscious Structure Layout. Proc. SIGPLAN Conference on Programming Language Design and Implementation. 1999. P. 1–12.