软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2019,30(11):3243−3258 [doi: 10.13328/j.cnki.jos.005567]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

多项式循环程序的秩函数探测*

李 轶, 冯 勇

(中国科学院 重庆绿色智能技术研究院,重庆 401120)

通讯作者: 李轶, E-mail: zm_liyi@163.com

摘 要: 秩函数法是循环程序终止性分析的主流方法.针对一类多分支多项式循环程序,这类程序的秩函数计算问题被证明可归结为单形上正定多项式的探测问题,从而便于利用线性规划工具 Simplex 去计算这类程序的秩函数.不同于现有基于柱形代数分解的量词消去算法,该方法能够在可接受的时间内计算更为复杂的多项式秩函数.

关键词: 可信计算;多分支循环程序;终止性;秩函数

中图法分类号: TP301

中文引用格式: 李轶,冯勇.多项式循环程序的秩函数探测.软件学报,2019,30(11):3243−3258. http://www.jos.org.cn/1000-9825/5567.htm

英文引用格式: Li Y, Feng Y. Detection of ranking functions of polynomial loop programs. Ruan Jian Xue Bao/Journal of Software, 2019,30(11):3243−3258 (in Chinese). http://www.jos.org.cn/1000-9825/5567.htm

Detection of Ranking Functions of Polynomial Loop Programs

LI Yi, FENG Yong

(Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing 401120, China)

Abstract: Synthesizing ranking functions of loop programs is the dominant method for checking their termination. In this study, the synthesis of ranking functions of a class of loop program is reduced to the detection of positive polynomial on the standard simplex. This then enables the computation of the desired ranking functions by linear programming tool. Different from the existing methods based on cylindrical algebraic decomposition, the proposed method in the study can get more expressive polynomial ranking functions within an acceptable time.

Key words: trusted computing; multi-path loop program; termination; ranking function

嵌入式系统在人类生活中发挥着越来越大的作用,而作为嵌入式系统灵魂的嵌入式软件在其中所占有的比重也越来越大.因此,嵌入式软件的可靠性将变得更加重要.诸如航空、航天、军事、交通、医疗等关键应用领域都对嵌入式系统的可靠性和安全性要求非常高,任何错误的发生都可能带来灾难性后果.嵌入式系统具有 3 个重要属性,即可达性、终止性、不变式.可达性是指一个系统能否从一个给定状态到达另一个可接受状态.某些混成系统的可达性被证明是能用计算机代数工具来检验的.不变式是系统变量在循环迭代时永远保持的特性.终止性是研究系统中是否会发生死循环.不包括终止性分析的验证被称为程序的部分正确性证明.因此,程序的终止性分析是确保程序完全正确性的必要基础.

尽管程序的终止性问题早已被证明是不可判定的[1],但对其进行研究不仅具有理论意义,更具有实际意义.当前,国内外主要通过计算秩函数来进行循环终止性分析.秩函数是验证循环程序终止性的一条重要途径.对给

* 基金项目: 国家自然科学基金(61572024, 61103110, 11671377); 重庆市自然科学基金(cstc2019jcyj-msxmX0638)

Foundation item: National Natural Science Foundation of China (61572024, 61103110, 11671377); Natural Science Foundation of Chongqing (cstc2019jcyj-msxmX0638)

收稿时间: 2017-09-01; 修改时间: 2017-11-09, 2017-12-18; 采用时间: 2018-02-11; jos 在线出版时间: 2018-04-27

CNKI 网络优先出版: 2018-04-27 14:58:08, http://kns.cnki.net/kcms/detail/11.2560.TP.20180427.1457.007.html

定的一个程序,若能找到其秩函数,则表明该程序必然是终止的.例如在 2001 年,Colon 等人[2,3]就利用多面体理论计算线性程序的线性秩函数(ranking functions).2004 年,针对线性循环程序,Poldelski 等人[4]提出一个完备的线性秩函数生成方法,并开发出工具 RANKFINDER.他们的方法将线性程序的线性秩函数计算问题归结为线性规划(LP)问题.在 2005 年,针对线性程序,Manna 等人[5,6]提出了字典秩函数的概念,并利用 Farkas 引理呈现了线性字典秩函数的计算方法.最近,文献[7,8]研究了现有几种线性秩函数生成算法的时间复杂度.文献[8]中进一步分析了程序变量在整数集合上取值时,这类线性程序的线性秩函数合成方法,并研究了这类问题的复杂度.但即便是线性程序,其秩函数也未必是线性的.因此,针对非线性程序,Cousot[9]通过半正定规划方法 SDP 研究了多项式秩函数的计算问题.该方法能够找到高次数的秩函数,但由于 SDP 工具内部采用的是数值计算,因此由计算误差导致最终得到的函数未必是真正的秩函数.同时,这个方法也不能回答一个程序是否具有预定义形式的秩函数.

符号计算理论方法及其工具被逐渐应用于程序自动验证.例如,我国科学家陈应华等人和杨路等人将秩函数和不变式的计算归约为半代数系统的求解,并运用基于柱形代数分解的工具 DISCOVERER,提出了多项式不等式型不变式和非线性秩函数生成方法[2,10].不同于文献[9]中的方法,他们的方法能够精确回答循环程序是否有给定模板的秩函数或不变式.

除了秩函数法以外,还出现了其他方法去进行循环程序的终止性研究.如,文献[11,12]提出了试探性方法去探索给定循环程序的非终止性.众所周知,一般的程序终止性问题是不可判定的,但对某些类特殊的程序而言,人们总希望能证明其终止性问题是可判定的.由此,鉴于终止性问题本身的不可判定性所带来的实际困难,证明循环程序终止性的另一途径就是避开秩函数的合成,而采用数学方法严格证明某类或某些类循环程序的终止性是可判定的,并建立相对完备的判定算法.这首先需要对程序进行分类.当限定程序中的赋值语句和条件语句均为线性多项式时,2004 年,Tiwari 在文献[13]中首次证明了下列单重无分支线性循环程序在实域上的终止性是可判定的:

$$\textbf{while } Bx>0 \textbf{ do } \{x:=Ax\} \textbf{ endwhile},$$

这里,A,B 均是实矩阵.相似的结论在 2006 年被 Braveman[14]推广到整数环上.此外,为了避免 Jordan 标准型的浮点计算,文献[15]中提出了精确的符号计算方法对程序 P 进行终止性判定.既然一般形式的线性程序终止性是不可判定的[13],那么非线性多项式循环程序由于其更为复杂的动力行为使得其终止性分析将变得更加困难.一个程序被称为非线性的,是指循环中的赋值映射或循环条件中的约束是非线性表达式.2005 年,利用有限差分树理论,文献[16]提出了试探性算法对一类含多分支语句的多项式程序的终止性问题进行判定.2010 年,针对一类赋值为线性且循环条件由非线性多项式不等式构成的循环程序,文献[17]分析了其终止性问题,证明了当这类程序满足给定的 NZM 条件时,其终止性是可判定的.2013 年,文献[18]通过分析多项式映射 f 的发散区间讨论了一类多项式循环(迭代赋值型如 $x_i:=f_i(x_i)$)的终止性问题.针对一类循环条件为多项式等式,赋值语句为多项式的多分支循环程序,詹乃军等人在文献[19]中给出了其可终止或不可终止的充分条件.最近,针对这类多分支程序.他们进一步证明了各条路径形成的理想具有一致上界,从而证明了这类多分支程序的终止性是可判定的.记 $\mathbf{R}$ 为实数域,$\mathbf{Q}$ 为有理数域.$\mathbf{R}[x](\mathbf{Q}[x])$表示由关于 x 的实系数(有理系数)多项式构成的多项式环.

本文主要研究具有以下形式的多分支线性循环程序在实数域上的终止性问题.

$$\left.\begin{array}{l} P\ \textbf{while } Bx \geqslant b \textbf{ do} \\ \tau_1 : \boldsymbol{x} := F_1(\boldsymbol{x}); \\ \text{or} \\ \ldots \\ \text{or} \\ \tau_m : \boldsymbol{x} := F_m(\boldsymbol{x}); \end{array}\right\} \tag{1}$$

这里,$B\in\mathbf{Q}^{n\times n}$ 为一可逆矩阵,$b\in\mathbf{Q}^n$,$\boldsymbol{x}\in\mathbf{R}^n$,$F_i(\boldsymbol{x})=(f_{i1}(\boldsymbol{x}),\ldots,f_{in}(\boldsymbol{x}))^T$ 为关于 $\boldsymbol{x}$ 的 n 维多项式映射,$f_{ij}(\boldsymbol{x})\in\mathbf{Q}[\boldsymbol{x}]$,$i=1,\ldots,m$,$j=1,\ldots,n$.记 $\Omega=\{\boldsymbol{x}\in\mathbf{R}^n:B\boldsymbol{x}\geqslant b\}$.为方便,令 $P\triangleq P(\Omega,F_1,\ldots,F_m)$.在上述多分支循环程序中,我们忽略了每个分支的条件

判断语句,这为多分支循环的终止性研究带来了便利.这一处理方式并非新的,它早已存在于文献[19]中.

对任意给定的 v 个 n 维多项式映射 $g_1(\boldsymbol{x}),\ldots,g_v(\boldsymbol{x})$,令 $\oplus_{j=1}^{v} g_j(\boldsymbol{x}) = g_v(\boldsymbol{x})\circ\ldots\circ g_1(\boldsymbol{x})$ 个映射的复合.我们说多分支循环程序 P 在实域上是不可终止的,如果存在 $\boldsymbol{x}_0\in\mathbf{R}^n$ 和无穷下标列表 $\Lambda=[i_j]_{j=1}^{+\infty}(i_j\in\{1,2,\ldots,m\})$,使得对任意的非负整数 n,有:

$$\oplus_{j=0}^{n} F_{i_j}(\boldsymbol{x}_0) = F_{i_n}\circ\ldots\circ F_{i_2}\circ F_{i_1}\circ F_{i_0}(\boldsymbol{x}_0)\in\Omega.$$

特别地,记 $F_{i_0}=I$ 为恒等映射.这里,$F_{i_j}\in\{I,F_1,\ldots,F_m\}$,$j$=1,2,...如果那样的 $\boldsymbol{x}_0$ 不存在,则表明程序 P 是可终止的.目前,对程序 P,往往采用合成秩函数(ranking functions)的方法来进行终止性分析[2,3,5−9,20].也即当程序 P 中各个分支的赋值语句都是线性表达式时,基于 Farkas 引理,文献[3,5,6,20]中的方法能够被用于计算 P 的线性秩函数.一个线性循环程序可能存在秩函数,但其秩函数未必是线性的.更重要的是,上述基于Farkas引理的方法并不能计算非线性循环程序的线性(或非线性)秩函数.文献[2,9]中的方法相同点在于:事先都要设定一个带参数的秩函数模板,然后通过不同的计算方法去计算得到参数的一个值.具体地说,给定某多项式循环程序,文献[9]将秩函数计算问题转化为半正定规划(SDP)问题,从而利用半正定规划方法和工具去计算该循环程序的多项式秩函数.但由于 SDP 采用浮点算法会导致计算得到的秩函数有可能并不是真正的秩函数,因此当 SDP 求解器计算出一个参数向量的取值时,还需要最后验证其对应的函数是否为程序的秩函数.不同于文献[9]的方法,文献[2]的方法采用柱形代数分解(CAD)算法的量词消去方法去计算参数模板的参数空间.他们的方法是纯符号的精确方法,所有的计算都不会涉及到近似计算.但由于 CAD 算法在最糟糕情形下的复杂度被证明是双指数的,故当秩函数模板次数稍高(大于 2 次)或变元稍多(大于 3 个)时,计算时间往往不可接受.本文中,基于 Polya 定理,我们提出了一种新的纯符号方法去探测程序 P 的多项式秩函数.该方法将程序 P 的多项式秩函数计算问题归结为线性规划(LP)求解问题.尽管求解 LP 问题存在多项式时间算法,如椭球算法和内点算法,但类似于上述的 SDP 方法,这些算法由于采用浮点计算从而导致所得到的结果未必精确可信.因此本文中,我们将采用线性规划求解算法中的经典算法——单纯形算法,去计算得到可行域中的一个精确点.同时,采用单纯形算法,我们不仅能够探测含次数更高,变元更多的复杂多项式秩函数,而且保证了计算所得到的函数就是程序的秩函数,从而不必像文献[9]中的方法那样需要最后验证所求得的函数是否为秩函数.

考虑程序 P.令 $y=M^{-1}(x)=Bx-b,\Omega y=\{y:y\geqslant 0\}$.既然程序 P 中的矩阵 B 是可逆的,则 M^{-1} 为可逆映射,记 M^{-1} 的逆映射为 $M(y)=B^{-1}(y+b)$.因此,M,M^{-1} 互为 $\mathbf{R}^n\to\mathbf{R}^n$ 的可逆映射.显然有,M^{-1} 是Ω到Ω_y 上的双射.同样地,M 是 Ω_y 到Ω上的双射.故下面的定理表明 P 在实数域上的终止性问题等价于下列程序在实数域上的终止性问题.

$$\left.\begin{array}{l} U\ \textbf{while}\ y\geqslant 0\ \textbf{do}\\ \tau_1:\boldsymbol{y}:=M^{-1}\circ F_1\circ M(\boldsymbol{y});\\ \text{or}\\ \ldots\\ \text{or}\\ \tau_m:\boldsymbol{y}:=M^{-1}\circ F_m\circ M(\boldsymbol{y}); \end{array}\right\} \tag{2}$$

与程序 P 的不可终止定义类似,我们说程序 U 在实域上是不可终止的,如果存在 $\boldsymbol{y}_0\in\mathbf{R}^n$ 和无穷下标列表 $\wedge=[i_j]_{j=1}^{+\infty}(i_j\in\{1,2,\ldots,m\})$,使得对任意的非负整数 n,有:

$$\overset{n}{\underset{j=0}{\oplus}}(M^{-1}\circ F_{ij}\circ M(\boldsymbol{y}_0)) = (M^{-1}\circ F_{i_n}\circ M)\circ\ldots\circ(M^{-1}\circ F_{i_1}\circ M)\circ(M^{-1}\circ F_{i_0}\circ M)(\boldsymbol{y}_0)\in\Omega_{\boldsymbol{y}}.$$

这里,对任意的 $j=0,1,\ldots,M^{-1}\circ F_{i_j}\circ M\in\{M^{-1}\circ F_{i_0}\circ M=I,M^{-1}\circ F_1\circ M,\ldots,M^{-1}\circ F_m\circ M\}$.

定理 1. 记号同上.程序 P 在实数域上不可终止等价于程序 U 在实数域上不可终止.

证明:若程序 P 在实数域上不可终止,则必存在 $\boldsymbol{x}_0\in\mathbf{R}^n$ 和无穷下标列表 $\wedge=[i_j]_{j=1}^{+\infty}(i_j\in\{1,2,\ldots,m\})$,使得对任意的 $n\geqslant 0$,有:

$$\oplus_{j=0}^{n}F_{i_j}(\boldsymbol{x}_0)=F_{i_n}\circ\ldots\circ F_{i_2}\circ F_{i_1}\circ F_{i_0}(\boldsymbol{x}_0)\in\Omega \tag{3}$$

这里, $F_{i_j}\in\{F_1,\ldots,F_m\}, j=1,2,\ldots$, 且 $i_j\in\Lambda, j=1,2,\ldots$令 $\boldsymbol{y}_0=M^{-1}(\boldsymbol{x}_0)=B\boldsymbol{x}_0-b$,同时有 $\boldsymbol{x}_0=M(\boldsymbol{y}_0)$.因此对任意的非负整数 $n\geqslant 0$,有:

$$\left.\begin{aligned}\overset{n}{\underset{j=0}{\oplus}}(M^{-1}\circ F_{i_j}\circ M)(\boldsymbol{y}_0) &=(M^{-1}\circ F_{i_n}\circ M)\circ\ldots\circ(M^{-1}\circ F_{i_1}\circ M)\circ(M^{-1}\circ F_{i_0}\circ M)\\ &=M^{-1}\circ F_{i_n}\circ F_{i_{n-1}}\circ\ldots\circ F_{i_0}\circ M(\boldsymbol{y}_0)\\ &=M^{-1}\circ F_{i_n}\circ F_{i_{n-1}}\circ\ldots\circ F_{i_0}(\boldsymbol{x}_0)\\ &=M^{-1}\circ F_{i_j}(\boldsymbol{x}_0)\end{aligned}\right\} \tag{4}$$

这里, $i_j\in\Lambda, j=1,2,\ldots, F_{i_0}=I$. 根据公式(3)、公式(4)中 $\oplus_{j=0}^{n}F_{i_j}(\boldsymbol{x}_0)\in\Omega$, 既然 M^{-1} 是从 Ω 到 $\Omega_{\boldsymbol{y}}$ 上的双射,我们有:

$$M^{-1}\circ\oplus_{j=0}^{n}F_{i_j}(\boldsymbol{x}_0)\in\Omega\ .$$

也即对任意的非负整数 $n\geqslant 0, \oplus_{j=0}^{n}(M^{-1}\circ F_{i_j}\circ M)(\boldsymbol{y}_0)\in\Omega_{\boldsymbol{y}}$.因此,存在一个点 $\boldsymbol{y}_0$ 和一个无穷下标列表 $\Lambda=[i_j]_{j=1}^{+\infty}(i_j\in\{1,2,\ldots,m\})$,使得对任意 $n\geqslant 0$,我们有 $\oplus_{j=0}^{n}(M^{-1}\circ F_{i_j}\circ M)(\boldsymbol{y}_0)\in\Omega_{\boldsymbol{y}}$. 根据程序不可终止的定义得知,程序在 $\boldsymbol{y}_0$ 处不可终止.

反之,若程序 U 在实数域上不可终止,那么必存在一点 $\boldsymbol{y}_0$ 和一个无穷下标列表 $\Lambda=[i_j]_{j=1}^{+\infty}(i_j\in\{1,2,\ldots,m\})$,使得对任意的 $n\geqslant 0$,我们有 $\oplus_{j=0}^{n}(M^{-1}\circ F_{i_j}\circ M)(\boldsymbol{y}_0)\in\Omega_{\boldsymbol{y}}$. 令 $\boldsymbol{x}_0=M(\boldsymbol{y}_0)$.根据公式(4)和 M^{-1} 的可逆性,有:

$$M\circ\oplus_{j=0}^{n}(M^{-1}\circ F_{i_j}\circ M)(\boldsymbol{y}_0)=M\circ M^{-1}\circ\oplus_{j=0}^{n}F_{i_j}(\boldsymbol{x}_0)=\oplus_{j=0}^{n}F_{i_j}(\boldsymbol{x}_0).$$

既然 $\oplus_{j=0}^{n}(M^{-1}\circ F_{i_j}\circ M)(\boldsymbol{y}_0)\in\Omega_{\boldsymbol{y}}$, 根据 M 是从 $\Omega_{\boldsymbol{y}}$ 到 Ω 上的双射的性质可知, $\oplus_{j=0}^{n}F_{i_j}(\boldsymbol{x}_0)\in\Omega$. 既然这里的 n 是任意非负整数,故按其不可终止的定义,程序 U 在 $\boldsymbol{x}_0$ 上不可终止. □

根据定理 1 可知, P 的终止性问题总能等价归结为 U 的终止性问题.为方便分析,不失一般性,将原来 U 中的程序变元 $\boldsymbol{y}$ 替换为 $\boldsymbol{x}$, $M^{-1}\circ F_i\circ M(\boldsymbol{y})$ 替换为 $T_i(\boldsymbol{x})$,则程序 U 可被重写为以下形式.

$$\left.\begin{aligned}&U\ \textbf{while}\ \boldsymbol{x}\geqslant 0\ \textbf{do}\\ &\tau_1:\boldsymbol{x}:=T_1(\boldsymbol{x});\\ &\text{or}\\ &\ldots\\ &\text{or}\\ &\tau_m:\boldsymbol{x}:=T_m(\boldsymbol{x});\end{aligned}\right\} \tag{5}$$

这里, $\boldsymbol{x}=(x_1,x_2,\ldots,x_n)^T\in\mathbf{R}^n, T_i(\boldsymbol{x})=(t_{i1}(\boldsymbol{x}),\ldots,t_{in}(\boldsymbol{x}))^T$ 为关于 x 的 n 维多项式映射, $t_{ij}(\boldsymbol{x})\in\mathbf{Q}[\boldsymbol{x}], i=1,\ldots,m, j=1,\ldots,n$.记:

$$\Omega_x=\{\boldsymbol{x}\in\mathbf{R}^n: x_1\geqslant 0,\ldots,x_n\geqslant 0\}.$$

既然 P 等价于程序 U 的终止性,因此为判定 P 的终止性,我们可以先分析 U 的终止性.

下文中,我们将通过计算多项式秩函数的方法去判定公式(5)中的程序 U 的终止性.

1 预备知识

本节将介绍有关秩函数、Polya 定理的相关基本知识.

定义 1. 给定如公式(5)定义的循环程序 U.函数 $\rho(\boldsymbol{x})\in\mathbf{R}[\boldsymbol{x}]$ 被称为程序 U 的秩函数,如果存在一个正数 $c>0$, 使得 $\rho(\boldsymbol{x})$ 满足:

- (有界): $\forall\boldsymbol{x},(\boldsymbol{x}\in\Omega_x\Rightarrow\rho(\boldsymbol{x})\geqslant 0)$;
- (下降): $\overset{m}{\underset{i=1}{\wedge}}(\forall\boldsymbol{x}\forall\boldsymbol{x}',(\boldsymbol{x}\in\Omega_x\wedge\boldsymbol{x}'=T_i(\boldsymbol{x})\Rightarrow\rho(\boldsymbol{x})-\rho(\boldsymbol{x}')\geqslant c))$.

注:根据定义 1 及下面的定义 3,计算程序 U 的秩函数等价于在 Ω_x 上探测 $m+1$ 个半正定多项式:

$$\rho(\boldsymbol{x})-\rho(T_i(\boldsymbol{x}))-c.$$

众所周知,如果一个循环程序具有秩函数,那么这个程序必定是终止的.对程序 U,文献[3,5,6,20]中的方法均不能计算其秩函数.这是因为 U 中的表达式可以是非线性多项式表达式,而上述文中基于 Farkas 引理的方法仅能计算线性循环程序的线性秩函数.针对程序 U,目前有两种方法可以计算它的非线性多项式秩函数.

- 一种是由杨路等人提出的基于符号计算的多项式秩函数计算方法,他们的方法是纯符号的方法,并不涉及近似计算.这一方法的特点是能够完备判定给定的程序是否具有预设模板形式的秩函数,但该方法是基于复杂度较高的柱形代数分解算法的,故可用以计算含变元较少、次数较低的秩函数合成问题.
- 另一种是由 Cousot 提出的基于 SDP 的秩函数计算方法.利用 SDP 的求解是多项式时间复杂度,该方法能够探测到含有较高次数、较多变元的秩函数.但不足之处在于,由于 SDP 采用浮点计算,从而导致最终得到的秩函数可能是不精确的,故还需要进一步验证.

在下一节中,我们将给出一种新的多项式秩函数计算方法.我们的方法是基于下面的 Polya 定理.首先给出一些多项式相关的定义.

定义 2. 给定多元齐次多项式 $f(\boldsymbol{x})\in\mathbf{R}[\boldsymbol{x}]$.我们说 f 是关于 $\boldsymbol{x}$ 的齐次多项式,如果其中的所有非零系数项的次数都相同.

例如, $f(x_1,x_2)=2x_1^2-6x_1x_2$ 就是关于 x_1,x_2 的齐次多项式,其次数为 2.特别地,如果 f 是非齐次多项式,那么 f 中次数最高的单项式的次数被称为 f 的次数.例如,非齐次多项式 $f(x_1,x_2)=x_1x_2+3x_1-x_2+1$ 中,单项式(忽略系数)有 $x_1x_2,x_1,x_2,1$,其关于 x_1,x_2 的次数分别为 2,1,1,0,因此 f 的次数为 2=max{2,1,1,0}.令 $x^\alpha=x_1^{\alpha_1}...x_n^{\alpha_n}$, $d=|\alpha|=\sum_{i=1}^n\alpha_i$, 记 $c(\alpha)=\dfrac{d!}{a_1!...a_n!}$.根据上述记号,一个齐 d 次 n 元多项式能够被写成以下形式.

$$f(x)=\sum_{|\alpha|=d}a_\alpha x^\alpha=\sum_{|\alpha|=d}c(\alpha)b_\alpha x^\alpha \tag{6}$$

这里, $a_\alpha=c(\alpha)\cdot b_\alpha$,其中, $c(\alpha)$ 为 α 的函数.例如,考虑 2 元齐 2 次多项式:

$$f(x_1,x_2)=2x_1^2-6x_1x_2=2x^{(2,0)}-6x^{(1,1)}=c((2,0))\cdot b_{(2,0)}x^{(2,0)}+c((1,1))\cdot b_{(1,1)}x^{(1,1)},$$

其中, $c((2,0))=1,b_{(2,0)}=2,c((1,1))=2,b_{(1,1)}=-3$.

定义 3. 给定 n 元齐次(或非齐次)多项式 $f(x_1,\ldots,x_n)\in\mathbf{R}[\boldsymbol{x}]$ 和 $S\subseteq\mathbf{R}^n$. f 在 S 上是正定的,如果对任意的 $\boldsymbol{x}\in S\setminus\{0\}$,都有 $f(\boldsymbol{x})>0$;同时, f 在 S 上是半正定的,如果对任意的 $\boldsymbol{x}\in S$,有 $f(\boldsymbol{x})\geqslant 0$.

显然,若 f 在 S 上是正定的,那么 f 在 S 上必然是半正定的.记单形 $\Delta_n=\left\{(x_1,...,x_n):x_i\geqslant 0,\sum_{i=1}^n x_i=1\right\}$.

定理 2(Polya 定理). 给定 n 元齐次多项式 $f(\boldsymbol{x})\in\mathbf{R}[\boldsymbol{x}]$,如果 f 在 Δ_n 上是正定的,那么当 N 充分大时,表达式 $(x_1+\ldots+x_n)^N f(x_1,\ldots,x_n)$ 展开后的所有系数均为正.

例如,不难验证 $f(x_1,x_2)=x_1^2-x_1x_2+x_2^2$ 在 $\Delta_2=\{(x_1,x_2):x_1\geqslant 0,x_2\geqslant 0,x_1+x_2=1\}$ 上是正定的.这是因为 $f(x_1,x_2)=x_1^2-x_1x_2+x_2^2=\left(\dfrac{x_1^2}{2}+\dfrac{x_2^2}{2}\right)+\left(\dfrac{x_1}{\sqrt{2}}-\dfrac{x_2}{\sqrt{2}}\right)^2$.因为 $\dfrac{x_1^2}{2}+\dfrac{x_2^2}{2}$ 只能在原点 $O(0,0)$ 点处的值为 0,且 $o(0,0)\notin\Delta_2$,故对任意的 $(x_1,x_2)\in\Delta_2,\dfrac{x_1^2}{2}+\dfrac{x_2^2}{2}>0$.又因为 $\left(\dfrac{x_1}{\sqrt{2}}-\dfrac{x_2}{\sqrt{2}}\right)\geqslant 0$,因此 f 在 Δ_2 上正定.尽管 f 中的系数不是全部为正,但根据 Polya 定理,存在 N,使得 $(x_1+x_2)^N\cdot f$ 的系数全为正.为此,令 $N=1$,展开 $(x_1+x_2)^1\cdot f=x_1^3+x_2^3$,其系数均为正.

注:Polya 定理表明,如果给定的齐次多项式 f 在单形上是正定的,那么必然存在 N,使得 $\left(\sum_{i=1}^n x_i\right)^N f$ 展开后各项系数为正;反之,若存在 N,使得 $\left(\sum_{i=1}^n x_i\right)^N f$ 展开后各项系数为正,则 f 在单形是半正定的(正定多项式一定是半正定的).因此,Polya 定理给出了 f 在单形上是正定的必要条件,同时也给出了 f 是单形上的半正定多项式的充分条件.但 Polya 定理并没告诉我们 N 的界.在文献[21]中,Powers 等人构造了 Polya 定理中 N 的界,并证明了那样的界仅仅依赖于多项式 f 的次数、系数和 f 在单形上的最小值.

定理 3[21]. 给定在单形Δ_n上正定的 n 元齐 d 次多项式$f(\boldsymbol{x})$.如果

$$N>\frac{d(d-1)}{2}\frac{L}{\lambda}-d \tag{7}$$

那么,$(x_1+\ldots+x_n)^N f(x_1,\ldots,x_n)$的所有系数为正.其中,$L=L(f)=\max\{|b_\alpha|:|\alpha|=d\}$,$\lambda=\lambda(f)=\min\{f(\boldsymbol{x}):\boldsymbol{x}\in\Delta_n\}$.

注:关于定理 3 有几点说明.

(A) 若存在正整数 N,使得$(x_1+\ldots+x_n)^N f(x_1,\ldots,x_n)$的所有系数为正,那么对任意的非负整数 j,有$(x_1+\ldots+x_n)^{N+j} f(x_1,\ldots,x_n)$的所有系数为正.这是因为$(x_1+\ldots+x_n)$中各项前的系数均为正数,故$(x_1+\ldots+x_n)^j$ 中各项前的系数必为正数.所以当$(x_1+\ldots+x_n)^N f(x_1,\ldots,x_n)$的所有系数为正时,$(x_1+\ldots+x_n)^{N+j} f(x_1,\ldots,x_n)=(x_1+\ldots+x_n)^j(x_1+\ldots+x_n)^N f(x_1,\ldots,x_n)$的所有系数仍然为正.

(B) N的下界公式,即公式(7)中,N的下界值与 L 的值成正比而与最小值成反比.在下一节中,我们将通过对公式(7)中 L 或λ的值进行放缩来调整 N 的下界公式,并在新的下界公式下建立相应算法.

(C) 若存在 N,使得$(x_1+\ldots+x_n)^N f(x_1,\ldots,x_n)$的所有系数为正,那么 $f(x_1,\ldots,x_n)$至少是单形上的半正定多项式.这是因为所有变量取值于单形,故均为非负变量;同时,每项系数都为正.

2 程序 U 的秩函数计算

本节中,我们将程序 U 的秩函数计算问题归结为单形上的正定多项式的探测问题,并基于 Polya 和 Powers 等人的结果,将单形上正定多项式探测的问题归结为线性规划问题.

定义 1 中的秩函数定义是传统秩函数定义,呈现在许多文献中.在定义 1 中,秩函数定义的有界和下降两个条件是由有限个蕴涵式刻画,且蕴涵式后件为非严格不等式.但为了便于利用正定多项式的性质,我们将定义 1 中程序 U 的秩函数定义做了稍微修改,即将定义 1 中蕴涵式后件中的非严格不等式全部替换为严格不等式.

定义 4. 给定如公式(5)定义的循环程序 U,函数$\rho(\boldsymbol{x})$是程序 U 的秩函数,若存在一个正数 $c>0$,使得$\rho(\boldsymbol{x})$满足:

- (有界):$\forall\boldsymbol{x},(\boldsymbol{x}\in\Omega_x\Rightarrow\rho(\boldsymbol{x})>0)$;
- (下降): $\bigwedge_{i=1}^{m}(\forall x\forall x',(X\in\Omega_x\wedge x'=T_i(x)\Rightarrow\rho(x)-\rho(x')>c))$.

注:修改后的秩函数定义与传统秩函数定义是等价的.这可被简单证明如下.

首先,若$\rho(\boldsymbol{x})$是满足定义 1 中两个条件的秩函数,即有$\forall x,(x\in\Omega_x\Rightarrow\rho(x)\geqslant 0)$且 $\bigwedge_{i=1}^{m}(\forall x\forall x',x\in\Omega_x\wedge x'=T_i(x)\Rightarrow\rho(x)-\rho'(x)\geqslant c\wedge c>0)$,进而有$\forall\boldsymbol{x},(\boldsymbol{x}\in\Omega_x\Rightarrow\rho(\boldsymbol{x})\geqslant 0>-1)$且 $\bigwedge_{i=1}^{m}(\forall x\forall x',x\in\Omega_x\wedge x'=T_i(x)\Rightarrow\rho(x)-\rho(x')\geqslant c>c/2\wedge c>0)$.那么令 $\rho'(x)=\rho(x)+1,c'=\frac{c}{2}$,则有:存在一个正数 $c'>0$,使得$\rho'(x)$满足定义 4 中的有界和下降两个条件.反之,若$\rho(\boldsymbol{x})$是满足定义 4 中两个条件的秩函数,那么$\rho(\boldsymbol{x})$显然也满足定义 1 中的两个条件.因此,根据定义 4,计算 U 的秩函数等价于在Ω_x上探测 m+1 个正定多项式.既然定义 1 与定义 4 等价,故计算 U 的秩函数⇔在Ω_x上探测 m+1 个半正定多项式⇔在Ω_x上探测 m+1 个正定多项式.根据定义 4,要计算程序 U 的秩函数,仅需探寻函数$\rho(\boldsymbol{x})$,使其满足定义 4 中的有界和下降条件.

- (有界):

$$\forall\boldsymbol{x},(\boldsymbol{x}\in\Omega_x\Rightarrow\rho(\boldsymbol{x})>0) \tag{8}$$

- (下降):

$$\bigwedge_{i=1}^{m}(\forall x,(x\in\Omega_x\Rightarrow H_i(x)>c)) \tag{9}$$

这里,$H_i(\boldsymbol{x})=\rho(\boldsymbol{x})-\rho(T_i(\boldsymbol{x}))$,$c>0$.一般地,公式(8)和公式 9)中蕴含式后件的多项式表达式未必是关于 $\boldsymbol{x}$ 的齐次多项式,但我们总可以引入新的变元进行齐次化,并能保证齐次化后的蕴含式与原蕴含式是等价的.令 $\hat{\boldsymbol{x}}=(\boldsymbol{x},z)$, 记 $\Omega_{\hat{x}}=\{\hat{\boldsymbol{x}}\in\mathbf{R}^{n+1}:x_1\geqslant 0,\ldots,x_n\geqslant 0,z>0\}$,记 $0_{\hat{x}}$ 为 $\mathbf{R}^{n+1}$ 中的原点.显然, $0_{\hat{x}}\notin\Omega_{\hat{x}}$.

命题 1. 给定非齐次多项式 $G(\boldsymbol{x})\in\mathbf{R}[\boldsymbol{x}]$,记 G 的次数为 d,引进新变元 z,对 G 进行齐次化得到 $G_H(\boldsymbol{x},z)$.记 w_1=

$\forall \boldsymbol{x},(\boldsymbol{x}\in\Omega_{\boldsymbol{x}}\Rightarrow G(\boldsymbol{x})>0)$和 $w_2\triangleq\forall\hat{\boldsymbol{x}},(\hat{\boldsymbol{x}}\in\Omega_{\hat{\boldsymbol{x}}}\Rightarrow G_H(\hat{\boldsymbol{x}})>0)$,则有 $w_1\Leftrightarrow w_2$.

证明:$w_2\Rightarrow w_1$ 是显然成立的.这是因为当蕴涵式 w_2 成立时,令 w_2 中的 z=1 即可得到 w_1 也成立.下证 $w_1\Rightarrow w_2$.

若蕴涵式 w_1 成立,即$\forall \boldsymbol{x},(\boldsymbol{x}\in\Omega_{\boldsymbol{x}}\Rightarrow G(\boldsymbol{x})>0)$.又 $G_H(\hat{x})=G_H(x,z)=z^d\cdot G\left(\frac{x_1}{z},...,\frac{x_n}{z}\right)$,令 $t_1=\frac{x_1}{z},...,t_n=\frac{x_n}{z}$.若 $z>0$,则$(x_1,\ldots,x_n)\in\Omega_{\boldsymbol{x}}$ 等价于 $t_1\geqslant 0,\ldots,t_n\geqslant 0$.既然已知 w_1 成立,因此有 $t_1\geqslant 0\wedge\ldots\wedge t_n\geqslant 0\Rightarrow G(t_1,\ldots,t_n)>0$.又因为 $z>0$,故

$$t_1\geqslant 0\wedge\ldots\wedge t_n\geqslant 0\wedge z>0\Rightarrow z^d\cdot G(t_1,\ldots,t_n)>0 \tag{10}$$

将 $t_i=\frac{x_i}{z},i=1,...,n$ 代入到公式(10),则有:

$$\frac{x_1}{z}\geqslant 0\wedge...\wedge\frac{x_n}{z}\geqslant 0\wedge z>0\Rightarrow z^d\cdot G\left(\frac{x_1}{z},...,\frac{x_n}{z}\right)=G_H(x,z)>0 \tag{11}$$

又因为 $\left\{(x,z):\frac{x_1}{z}\geqslant 0\wedge...\wedge\frac{x_n}{z}\geqslant 0\wedge z>0\right\}=\{(x,z):x_1\geqslant 0\wedge...\wedge x_n\geqslant 0\wedge z>0\}$,将公式(11)中的前件替换为 $x_1\geqslant 0\wedge\ldots\wedge x_n\geqslant 0\wedge z>0$,即可得 w_2 成立. □

假设公式(8)和公式(9)中的多项式$\rho(\boldsymbol{x}),H_i(\boldsymbol{x})-c$ 的次数分别为 d_ρ,d_{H_i},然后引入新变元 z,对$\rho(\boldsymbol{x}),H_i(\boldsymbol{x})-c$,$i$=1,…,$m$ 进行齐次化.记齐次化所得的表达式分别为

$$\hat{\rho}(\hat{\boldsymbol{x}})=\hat{\rho}(\boldsymbol{x},z),M_i(\hat{\boldsymbol{x}})=M_i(\boldsymbol{x},z)=\hat{H}_i(\hat{\boldsymbol{x}})-c\cdot z^{d_{H_i}}.$$

显然,$\hat{\rho}(\hat{\boldsymbol{x}}),M_i(\hat{\boldsymbol{x}})$ 的次数分别为 d_ρ,d_{H_i},其中,$\hat{H}_i(\hat{\boldsymbol{x}})$ 是将 $H_i(\boldsymbol{x})$进行齐次化后所得到的表达式.根据命题 1,定义 4 中刻画秩函数的两个条件:公式(8)和公式(9)分别等价于以下公式.

$$\left.\begin{array}{l}\forall\hat{\boldsymbol{x}},(\hat{\boldsymbol{x}}\in\Omega_{\hat{\boldsymbol{x}}}\Rightarrow\hat{\rho}(\hat{\boldsymbol{x}})>0)\\ \mathop{\wedge}\limits_{i=1}^{m}(\forall\hat{\boldsymbol{x}},(\hat{\boldsymbol{x}}\in\Omega_{\hat{\boldsymbol{x}}}\Rightarrow M_i(\hat{\boldsymbol{x}})>0))\end{array}\right\} \tag{12}$$

令 $\bar{\Omega}_{\hat{\boldsymbol{x}}}=\{\hat{\boldsymbol{x}}=(\boldsymbol{x},z)\in\mathbf{R}^{n+1}:x_1\geqslant 0,...,x_n\geqslant 0,z\geqslant 0\}$,既然 $0_{\hat{x}}\notin\Omega_{\hat{\boldsymbol{x}}}$ 且 $\Omega_{\hat{\boldsymbol{x}}}\subseteq\bar{\Omega}_{\hat{\boldsymbol{x}}}$,显然有 $\Omega_{\hat{\boldsymbol{x}}}\subseteq\bar{\Omega}_{\hat{\boldsymbol{x}}}\setminus\{0_{\hat{x}}\}$.

令 $\Omega^o_{\hat{\boldsymbol{x}}}=\bar{\Omega}_{\hat{\boldsymbol{x}}}\setminus\{0_{\hat{x}}\}$,即 $\Omega^o_{\hat{\boldsymbol{x}}}$ 中不包含原点,因此,将公式(12)中的 $\Omega_{\hat{\boldsymbol{x}}}$ 替换为 $\Omega^o_{\hat{\boldsymbol{x}}}$,则得到:

$$\left.\begin{array}{l}\forall\hat{\boldsymbol{x}},(\hat{\boldsymbol{x}}\in\Omega^o_{\hat{\boldsymbol{x}}}\Rightarrow\hat{\rho}(\hat{\boldsymbol{x}})>0)\\ \mathop{\wedge}\limits_{i=1}^{m}(\forall\hat{\boldsymbol{x}},(\hat{\boldsymbol{x}}\in\Omega^o_{\hat{\boldsymbol{x}}}\Rightarrow M_i(\hat{\boldsymbol{x}})>0))\end{array}\right\} \tag{13}$$

既然 $\Omega_{\hat{\boldsymbol{x}}}\subseteq\Omega^o_{\hat{\boldsymbol{x}}}$,因此,若公式(13)成立,则公式(12)也成立.因此,计算满足公式(8)和公式(9)的函数$\rho(\boldsymbol{x})$可归结为判定公式(13)是否成立.

令$\Delta_{n+1}=\{(x_1,\ldots,x_n,z)\in\mathbf{R}^{n+1}:x_1\geqslant 0,\ldots,x_n\geqslant 0,z\geqslant 0,x_1+\ldots+x_n+z=1\}$.

显然,$0_{\hat{x}}\notin\Delta_{n+1}$ 且 $\Delta_{n+1}\subseteq\Omega_{\hat{\boldsymbol{x}}}$.因此,$\Delta_{n+1}\subseteq\Omega^o_{\hat{\boldsymbol{x}}}$.令

$$\left.\begin{array}{l}\forall\hat{\boldsymbol{x}},(\hat{\boldsymbol{x}}\in\Delta_{n+1}\Rightarrow\hat{\rho}(\hat{\boldsymbol{x}})>0)\\ \mathop{\wedge}\limits_{i=1}^{m}(\forall\hat{\boldsymbol{x}},(\hat{\boldsymbol{x}}\in\Delta_{n+1}\Rightarrow M_i(\hat{\boldsymbol{x}})>0))\end{array}\right\} \tag{14}$$

给定两个非零点 $\hat{x},\hat{y}\in\mathbf{R}^{n+1}$,我们记 $\hat{y}\sim\hat{x}$.如果存在正数λ>0,使得 $\hat{\boldsymbol{y}}=\lambda\cdot\hat{\boldsymbol{x}}$,令 $[\hat{\boldsymbol{x}}]=\{\hat{\boldsymbol{y}}\in\mathbf{R}^{n+1}:\hat{\boldsymbol{y}}\sim\hat{\boldsymbol{x}}\}$,即 $[\hat{\boldsymbol{x}}]$ 代表了一条从原点出发且通过点 $\hat{\boldsymbol{x}}$ 的射线,而 $\hat{\boldsymbol{x}}$ 被称为该射线的代表元.显然,射线上任一点均可被称为该射线的代表元,记 $\sum(\hat{\boldsymbol{x}})=x_1+...+x_n+z$. 下面的结论表明,公式(13)与公式(14)是等价的.

命题 2. 记号同上.公式(13)⇔公式(14).

证明:既然 $\Delta_{n+1}\subseteq\Omega^o_{\hat{\boldsymbol{x}}}$,那么公式(13)⇒公式(14)是平凡成立的.下面证公式(14)⇒公式(13)也成立.根据 $\Omega^o_{\hat{\boldsymbol{x}}}$ 的定义可知,$\Omega^o_{\hat{\boldsymbol{x}}}$ 是由从原点出发但不包含原点的射线构成,因此,给定 $\hat{\boldsymbol{x}}\in\Omega^o_{\hat{\boldsymbol{x}}}$,对任意的$\lambda$>0,都有: $\lambda\cdot\hat{\boldsymbol{x}}\in\Omega^o_{\hat{\boldsymbol{x}}}\cdot\Delta_{n+1}$ 是 $\Omega^o_{\hat{\boldsymbol{x}}}$ 与平面$(x_1+\ldots+x_n+z=1$ 的交集.下面证明 $\Omega^o_{\hat{\boldsymbol{x}}}$ 中的每一条射线均与 $x_1+\ldots+x_n+z=1$ 定义的平面相交.

任取一点 $\hat{\boldsymbol{x}}_0\in\Omega^o_{\hat{\boldsymbol{x}}}$,根据 $\Omega^o_{\hat{\boldsymbol{x}}}$ 的定义,$\hat{\boldsymbol{x}}_0\neq 0$.取 $\lambda_0=\frac{1}{\sum\hat{\boldsymbol{x}}_0}$.既然 $\Omega^o_{\hat{\boldsymbol{x}}}$ 中的任一非零点的所有分量均非负且不

全为0,故$\lambda_0>0$.因此,$\lambda_0\hat{\boldsymbol{x}}_0\in\Omega_{\hat{\boldsymbol{x}}}^o$.同时有$\sum(\lambda_0\hat{\boldsymbol{x}}_0)=\lambda_0\sum(\hat{\boldsymbol{x}}_0)=1$,故$\lambda_0\hat{\boldsymbol{x}}_0$落在$x_1+\ldots+x_n+z+\ldots=1$定义的平面上.由此可得,$\lambda_0\hat{\boldsymbol{x}}_0\in\Delta_{n+1}$即为交点.这就证明$\Omega_{\hat{\boldsymbol{x}}}^o$中的每一条射线均与$x_1+\ldots+x_n+z=1$定义的平面相交.其交点与从原点出发且过该交点的射线一一对应.因此,这样的交点实为与之对映的射线的代表元.为方便,称它们为标准代表元.因此,$\Omega_{\hat{\boldsymbol{x}}}^o$中的任何一点$\hat{\boldsymbol{x}}$都在$\Delta_{n+1}$上存在唯一一点$\frac{\hat{\boldsymbol{x}}}{\sum\hat{\boldsymbol{x}}}$与之对应.故,$\Omega_{\hat{\boldsymbol{x}}}^o$中所有射线对映的标准代表元构成了$\Delta_{n+1}$.给定齐$d$次多项式$U(\hat{\boldsymbol{x}})$,它具有如下简单性质:若对$\hat{\boldsymbol{x}}\in\Delta_{n+1}$有$U(\hat{\boldsymbol{x}})>0$,那么对任意$\lambda>0$,有$U(\lambda\hat{\boldsymbol{x}})=\lambda^dU(\hat{\boldsymbol{x}})>0$.也即若齐次多项式$U(\hat{\boldsymbol{x}})$在单形$\Delta_{n+1}$上某点的取值为正,那么该齐次多项式也必在过该点且从原点出发但不包含原点的射线上的任一点处取值为正.进而可得:如果齐次多项式$U(\hat{\boldsymbol{x}})$在单形Δ_{n+1}上的每一点处取值为正,则根据上述分析得知,$U(\hat{\boldsymbol{x}})$在$\Omega_{\hat{\boldsymbol{x}}}^o$的每一点处为正.因此,若公式(14)成立,即对$\forall\hat{\boldsymbol{x}},(\hat{\boldsymbol{x}}\in\Delta_{n+1}\Rightarrow\hat{\rho}(\hat{\boldsymbol{x}})>0)$且$\wedge_{i=1}^m(\forall\hat{\boldsymbol{x}},(\hat{\boldsymbol{x}}\in\Delta_{n+1}\Rightarrow M_i(\hat{\boldsymbol{x}})>0))$,那么因为$\rho(\hat{\boldsymbol{x}}),M_i(\hat{\boldsymbol{x}})$为齐次多项式且其次数分别为$d_\rho,d_{H_i}$,故有:

$$\forall\lambda\forall\hat{\boldsymbol{x}},(\lambda>0\wedge\hat{\boldsymbol{x}}\in\Delta_{n+1}\Rightarrow\lambda^{d_\rho}\hat{\rho}(\hat{\boldsymbol{x}})=\hat{\rho}(\lambda\hat{\boldsymbol{x}})>0)\text{且}\wedge_{i=0}^m(\forall\lambda\forall\hat{\boldsymbol{x}},(\lambda>0\wedge\hat{\boldsymbol{x}}\in\Delta_{n+1}\Rightarrow\lambda^{d_{M_i}}M_i(\hat{\boldsymbol{x}})=M_i(\lambda\hat{\boldsymbol{x}})>0).$$

又因为集合$\{\lambda\hat{\boldsymbol{x}}:\lambda>0\wedge\hat{\boldsymbol{x}}\in\Delta_{n+1}\}$正好就是$\Omega_{\hat{\boldsymbol{x}}}^o$,故公式(13)成立. □

注:若将公式(13)和公式(14)中的所有不等式均改为非严格不等式,则命题2仍然成立.

要判定公式(14)是否成立,等价于判定齐次多项式$\hat{\rho}(\hat{\boldsymbol{x}}),M_i(\hat{\boldsymbol{x}})$在单形$\Delta_{n+1}$上是否正定的.根据上述分析,为了计算程序$U$的秩函数,需要以下5个步骤.

S1. 给出秩函数模板——带参数的多项式表达式.不失一般性,可假设为$\rho(\boldsymbol{x})=\boldsymbol{a}^T\cdot\Gamma$,其中,$\boldsymbol{a}$为参系数向量,$\Gamma$是所有单项式构成的向量.如$\rho(\boldsymbol{x})=a_1x^2y+a_2x+a_3y+a_4$,则$\boldsymbol{a}=(a_1,\ldots,a_4)^T,\Gamma=(x^2y,x,y,1)^T$.

S2. 将$\rho(\boldsymbol{x})$代入到定义4中给出的秩函数条件中,从而构造公式(8)和公式(9).其中,可记公式(9)中的$H_i(\boldsymbol{x})=\rho(\boldsymbol{x})-\rho(T_i(\boldsymbol{x}))=\boldsymbol{b}_{\boldsymbol{a}}^T\cdot\Gamma'$.这里,$\boldsymbol{b}_{\boldsymbol{a}}$为一向量,其每个分量均是关于$\boldsymbol{a}$的齐次线性表达式;$\Gamma'$为单项式构成的向量.如,给定循环程序的第$i$个赋值语句$T_i(\boldsymbol{x})\triangleq(x:=3x-4y^2,y:=y)^T$,则$H_i(\boldsymbol{x})=\rho(\boldsymbol{x})-\rho(T_i(\boldsymbol{x}))=-8a_1x^2y-2a_2x+24a_1y^3x+4a_2y^2-16a_1y^5$.故,$\boldsymbol{a}=(-8a_1,-2a_2,24a_1,4a_2,-16a_1)^T,\Gamma'=(x^2y,x,y^3x,y^2,y^5)^T$.

S3. 对公式(8)和公式(9)中的$\rho(\boldsymbol{x}),H_i(\boldsymbol{x})-c$进行齐次化,得到:

$$\hat{\rho}(\hat{\boldsymbol{x}})=\boldsymbol{a}^T\cdot\Gamma_0,M_i(\hat{\boldsymbol{x}})=\hat{H}_i(\hat{\boldsymbol{x}})-c\cdot z^{d_{H_i}}=\boldsymbol{b}_{\boldsymbol{a}}^T\cdot\Gamma_0'-c\cdot z^{d_{H_i}}=(\boldsymbol{b}_{\boldsymbol{a}}^T,-c)\cdot(\Gamma_0'^T,z^{d_{H_i}})^T.$$

从而构造公式(12).这里,d_{H_i}为$H_i(\boldsymbol{x})$的次数,Γ_o,Γ_o'分别为Γ,Γ'齐次化后得到的向量.

S4. 根据上述分析,因为公式(13)⇒公式(12),所以要使得公式(12)成立,只需保证公式(13)成立即可.再根据命题2可知,公式(13)成立等价于公式(14)成立.进而将问题归结为保证公式(14)成立.

S5. 探测是否存在参系数$(\boldsymbol{a},\boldsymbol{b}_{\boldsymbol{a}},c)$的一组取值$(\boldsymbol{a}^*,\boldsymbol{b}_{\boldsymbol{a}}^*,c^*)$,使得$\hat{\rho}^*(\hat{\boldsymbol{x}})=\boldsymbol{a}^{*T}\cdot\tau_0,M_i^*(\hat{\boldsymbol{x}})=(\boldsymbol{b}_{\boldsymbol{a}}^{*T},-c^*)\cdot(\tau_0'^T,z^{d_{H_i}})^T$满足公式(14).若那样的一组取值存在,则表明程序U是终止的.

从上面5个步骤可以看出,计算程序U秩函数的问题被归约为探测单形上的正定多项式$\hat{\rho}(\hat{\boldsymbol{x}}),M_i(\hat{\boldsymbol{x}})$的问题.也即若单形上的正定多项式$\hat{\rho}(\hat{\boldsymbol{x}}),M_i(\hat{\boldsymbol{x}})$被探测到,则表明程序$U$具有秩函数,从而证明程序$U$是终止的.

根据下面的命题可知,在上述S5中,存在参系数$(\boldsymbol{a},\boldsymbol{b}_{\boldsymbol{a}},c)$的一组取值$(\boldsymbol{a}^*,\boldsymbol{b}_{\boldsymbol{a}}^*,c^*)$,使得$\hat{\rho}^*(\hat{\boldsymbol{x}})=\boldsymbol{a}^{*T}\cdot\Gamma_0,M_i^*(\hat{\boldsymbol{x}})=(\boldsymbol{b}_{\boldsymbol{a}}^{*T},-c^*)\cdot(\Gamma_0'^T,z^{d_{H_i}})^T$满足公式(14),等价于存在参系数$(\boldsymbol{a},\boldsymbol{b}_{\boldsymbol{a}},c)$在$[-1,1]^\ell$中的一组取值$(\boldsymbol{a}^{**},\boldsymbol{b}_{\boldsymbol{a}}^{**},c^{**})$,使得$\hat{\rho}^{**}(\hat{\boldsymbol{x}})=\boldsymbol{a}^{**T}\cdot\Gamma_0,M_i^{**}(\hat{\boldsymbol{x}})=(\boldsymbol{b}_{\boldsymbol{a}}^{**T},-c^{**})\cdot(\Gamma_0'^T,z^{d_{H_i}})^T$满足式(14).这里,$\ell=|(\boldsymbol{a},\boldsymbol{b}_{\boldsymbol{a}},c)|$为向量$(\boldsymbol{a},\boldsymbol{b}_{\boldsymbol{a}},c)$中分量的个数.$(\boldsymbol{a},\boldsymbol{b}_{\boldsymbol{a}},c)\in[-1,1]^\ell$表明,向量$(\boldsymbol{a},\boldsymbol{b}_{\boldsymbol{a}},c)$中每个分量都在区间$[-1,1]$中取值.

命题3. 记号同上.存在$(\boldsymbol{a}^*,\boldsymbol{b}_{\boldsymbol{a}}^*,c_*)\in R^\ell$,使得$\hat{\rho}^*(\hat{\boldsymbol{x}})=\boldsymbol{a}^{*T}\cdot\tau_0,M_i^*(\hat{\boldsymbol{x}})=(\boldsymbol{b}_{\boldsymbol{a}}^{*T},-c^*)\cdot(\tau_0'^T,z^{d_{H_i}})^T$满足公式(14),等价于存在$(\boldsymbol{a}^{**},\boldsymbol{b}_{\boldsymbol{a}}^{**},c^{**})\in[-1,1]^\ell$,使得$\hat{\rho}^{**}(\hat{\boldsymbol{x}})=\boldsymbol{a}^{**T}\cdot\tau_0,M_i^{**}(\hat{\boldsymbol{x}})=(\boldsymbol{b}_{\boldsymbol{a}}^{**T},-c^{**})\cdot(\tau_0'^T,z^{d_{H_i}})^T$满足公式(14).

证明:该证明非常简单."⇐"是平凡成立的,既然$[-1,1]^\ell\subset R^\ell$.故我们仅证"⇒".

若存在$\nu=(\boldsymbol{a}^*,\boldsymbol{b}_{\boldsymbol{a}}^*,c^*)\in\mathbf{R}^l$,使得$\hat{\rho}^*(\hat{\boldsymbol{x}}),M_i^*(\hat{\boldsymbol{x}})$满足公式(14)中的两个蕴涵式,那么我们可以令

$$v'=(\boldsymbol{a}^{**},\boldsymbol{b}_{\boldsymbol{a}}^{**},c^{**})=\left(\frac{\boldsymbol{a}^{*}}{abs(v)},\frac{\boldsymbol{b}_{\boldsymbol{a}}^{*}}{abs(v)},\frac{c^{*}}{abs(v)}\right).$$

这里,$abs(v)=\sum i\,|\,v_i\,|$ 为向量 v 的分量的绝对值之和.显然有,$v'\in[-1,1]^l$ 且 $\hat{\rho}^{**}(\hat{\boldsymbol{x}})=\frac{1}{abs(v)}\hat{\rho}^{*}(\hat{\boldsymbol{x}})=\frac{1}{abs(v)}\cdot\boldsymbol{a}^{*T}\cdot\tau_o,M_i^{**}(\hat{\boldsymbol{x}})=\frac{1}{abs(v)}M_i^{*}(\hat{\boldsymbol{x}})=\frac{1}{abs(v)}\cdot(\boldsymbol{b}_{\boldsymbol{a}}^{*T},-c^{*})\cdot(\tau_o'^{T},z^{d_{H_i}})^T$ 满足公式(14). □

根据定义 4、命题 1~命题 3,我们可以建立下面的结论.

定理 4. 给定程序 U,若存在 $(\boldsymbol{a}^{*},\boldsymbol{b}_{\boldsymbol{a}}^{*},c^{*})$ 的一组取值 $(\boldsymbol{a}^{*},\boldsymbol{b}_{\boldsymbol{a}}^{*},c^{*})\in[-1,1]^{\ell}$,使得 $\hat{\rho}^{*}(\hat{\boldsymbol{x}})=\boldsymbol{a}^{*T}\cdot\tau_0,M_i^{*}(\hat{\boldsymbol{x}})=(\boldsymbol{b}_{\boldsymbol{a}}^{*T},-c^{*})\cdot(\tau_0'^{T},z^{d_{H_i}})^T$ 满足公式(14),那么程序 U 必然终止.

在 S5 中,我们需要判断是否存在 $(\boldsymbol{a}^{*},\boldsymbol{b}_{\boldsymbol{a}}^{*},c^{*})$ 的一组取值 $(\boldsymbol{a}^{*},\boldsymbol{b}_{\boldsymbol{a}}^{*},c^{*})$,使得 $\hat{\rho}^{*}(\hat{\boldsymbol{x}}),M_i^{*}(\hat{\boldsymbol{x}})$ 满足公式(14),这等价于在单形 Δ_{n+1} 上探测是否存在正定多项式 $\hat{\rho}^{*}(\hat{\boldsymbol{x}}),M_i^{*}(\hat{\boldsymbol{x}})$.我们将利用上述的定理 2(Polya 定理)和定理 3 去探测单形上的正定多项式.定理 2 表明,若齐次多项式 f 在单形 Δ_n 上是正定的,则必然存在充分大的正整数 N,使得 $(x_1+\ldots+x_n)^N f(x_1,\ldots,x_n)$ 展开后的所有系数均为正.但 Polya 并没有给出 N 的上界.在定理 3 中,Powers 等人根据 f 的次数、系数以及 f 在单形上的最小值,构造了 N 的界.

$$N>\frac{d(d-1)}{2}\frac{L}{\lambda}-d \tag{15}$$

这里,$L=L(f)=\max\{|b_\alpha|:|\alpha|=d\},\lambda=\lambda(f)=\min\{f(\boldsymbol{x}):\boldsymbol{x}\in\Delta_n\}$.根据下面的结论,我们可以将公式(15)中的 L 替换为 1.

命题 4. 记号同上.给定齐 d 次 n 元多项式 $f(\boldsymbol{x})=\sum_{|\alpha|=d}a_\alpha\boldsymbol{x}^\alpha=\sum_{|\alpha|=d}c(\alpha)b_\alpha\boldsymbol{x}^\alpha$,这里,非负整数向量 $\alpha=(\alpha_1,\ldots,\alpha_n)\in\mathbf{N}^n$,则 $(\alpha_1+\ldots+\alpha_n)!\geqslant\alpha_1!\ldots\alpha_n!$.

证明:考虑 $n=2$.

$$(\alpha_1+\alpha_2)!=(\alpha_1-0+\alpha_2)(\alpha_1-1+\alpha_2)(\alpha_1-2+\alpha_2)\ldots(\alpha_1-(\alpha_1-1)+\alpha_2)\cdot(\alpha_2)!$$

显然,$(\alpha_1-0+\alpha_2)\geqslant\alpha_1,(\alpha_1-1+\alpha_2)\geqslant\alpha_1-1,(\alpha_1-2+\alpha_2)\geqslant\alpha_1-2,\ldots,(\alpha_1-(\alpha_1-1)+\alpha_2)\geqslant1$.故

$$(\alpha_1+\alpha_2)!\geqslant\alpha_1!\alpha_2! \tag{16}$$

考虑 $n=3$.

根据公式(16),有 $(\alpha_1+\alpha_2+\alpha_3)!=(\alpha_3+(\alpha_1+\alpha_2))!\geqslant(\alpha_3)!(\alpha_1+\alpha_2)!\geqslant\alpha_3!\alpha_1!\alpha_2!$.

以此类推,可得 $(\alpha_1+\ldots+\alpha_n)!\geqslant\alpha_1!\ldots\alpha_n!$. □

注:给定齐 d 次 n 元多项式 $f(\boldsymbol{x})=\sum_{|\alpha|=d}a_\alpha\boldsymbol{x}^\alpha=\sum_{|\alpha|=d}c(\alpha)b_\alpha\boldsymbol{x}^\alpha$,在公式(15)中,

$$L=L(f)=\max\left\{|b_\alpha|=\frac{|a_\alpha|}{c(\alpha)}:|\alpha|=d\right\}.$$

根据命题 4 可知,$c(\alpha)=\frac{d!}{\alpha_1!\ldots\alpha_n!}=\frac{(\alpha_1+\ldots+\alpha_n)!}{\alpha_1!\ldots\alpha_n!}\geqslant1$.假如 f 的所有系数 a_α 均在区间 $[-\eta,\eta]$ 中,那么 $\frac{|a_\alpha|}{c(\alpha)}\leqslant\eta$.

因此,倘若 f 的所有系数 a_α 均在区间 $[-\eta,\eta]$ 中,那么 $L=L(f)=\max\left\{|b_\alpha|=\frac{|a_\alpha|}{c(\alpha)}:|\alpha|=d\right\}\leqslant\eta$.根据上面的分析,若系数 a_α 在区间 $[-1,1]$,可以得到下列结果:

定理 5. 给定齐 d 次 n 元多项式 $f(\boldsymbol{x})=\sum_{|\alpha|=d}a_\alpha\boldsymbol{x}^\alpha$,且其所有系数 a_α 均在区间 $[-1,1]$ 中.如果 f 在 Δ_n 上是正定的,则当:

$$N>\frac{d(d-1)}{2}\frac{1}{\lambda}-d \tag{17}$$

有 $(x_1+\ldots+x_n)^N f(x_1,\ldots,x_n)$ 的所有系数为正.其中,$\lambda=\lambda(f)=\min\{f(\boldsymbol{x}):\boldsymbol{x}\in\Delta_n$.

根据定理 4 得知,可以在系数空间 $[-1,1]^{\ell}$ 中探测满足公式(14)的齐次多项式 $\hat{\rho}(\hat{\boldsymbol{x}})=\boldsymbol{a}^T\cdot\tau_0,M_i(\hat{\boldsymbol{x}})=(\boldsymbol{b}_{\boldsymbol{a}}^T,-c)\cdot(\tau_0'^{T},z^{d_{H_i}})^T$.这里,$(\boldsymbol{a},\boldsymbol{b}_{\boldsymbol{a}},c)\in[-1,1]^{\ell}$.同时,满足公式(14)的 $\hat{\rho}(\hat{\boldsymbol{x}}),M_i(\hat{\boldsymbol{x}})$ 恰好是单形 Δ_{n+1} 上的正定多项式.因此,既然

$\hat{\rho}(\hat{\boldsymbol{x}}),M_i(\hat{\boldsymbol{x}})$ 在单形Δ_{n+1} 上正定且其所有系数均在$[-1,1]^\ell$ 中取值,那么根据定理 5,必然存在一个 N(其值仅仅依赖于最小值λ),使得 $(x_1+...+x_n+z)^N\hat{\rho}(\hat{\boldsymbol{x}}),(x_1+...+x_n+z)^N M_i(\hat{\boldsymbol{x}})$ 的所有系数为正.因此,为了得到$(\boldsymbol{a},\boldsymbol{b_a},c)$的一组取值,我们可以从 $(x_1+...+x_n+z)^N\hat{\rho}(\hat{\boldsymbol{x}}),(x_1+...+x_n+z)^N M_i(\hat{\boldsymbol{x}})$ 中抽取出所有系数(它们均是关于$(\boldsymbol{a},\boldsymbol{b_a},c)$的齐次线性表达式),并令所有系数大于 0,且加上约束 $c>0\wedge(\boldsymbol{a},\boldsymbol{b_a},c)\in[-1,1]^\ell$,从而构造出关于$(\boldsymbol{a},\boldsymbol{b_a},c)$的不等式组 *Sys*.若 *Sys* 有实解,则它的任一组解 $(\boldsymbol{a}^*,\boldsymbol{b}_{\boldsymbol{a}}^*,c^*)$ 就正好给出了单形Δ_{n+1} 上的(半)正定多项式 $\hat{\rho}^*(\hat{\boldsymbol{x}}),M_i^*(\hat{\boldsymbol{x}})$ (这是因为根据定理 2 或定理 3 的注解可知,若存在 N,使得 $\left(\sum_{i=1}^n x_i\right)^N f$ 展开后各项系数为正,则f在单形是半正定的).下面的定理表明,若系统 *Sys* 有实解,则程序 U 可终止.

定理 6. 记号同上.给定程序 U,若系统 *Sys* 有实解,则程序 U 是终止的.

证明:不妨令 $(\boldsymbol{a}^*,\boldsymbol{b}_{\boldsymbol{a}}^*,c^*)$ 为系统 *Sys* 的一组解,则有$c^*>0\wedge(\boldsymbol{a}^*,\boldsymbol{b}_{\boldsymbol{a}}^*,c^*)\in[-1,1]^\ell$;且存在 N,使得$(x_1+...+x_n+z)^N\cdot\hat{\rho}^*(\hat{\boldsymbol{x}}),(x_1+...+x_n+z)^N M_i^*(\hat{\boldsymbol{x}})$ 的各项系数为正.这里, $M_i^*(\hat{\boldsymbol{x}})=(\boldsymbol{b}_{\boldsymbol{a}}^{*T},-c^*)\cdot(\Gamma_0'^T,z^{d_{H_i}})^T,\hat{\rho}^*(\hat{\boldsymbol{x}})=\boldsymbol{a}^{*T}\cdot\Gamma_0$. 因此,根据定理 2 或定理 3 的注解可知, $\hat{\rho}^*(\hat{\boldsymbol{x}}),M_i^*(\hat{\boldsymbol{x}})$ 是单形Δ_{n+1} 上的半正定多项式,即

$$\forall\hat{\boldsymbol{x}},(\hat{\boldsymbol{x}}\in\Delta_{n+1}\Rightarrow\hat{\rho}^*(\hat{\boldsymbol{x}})\geqslant 0)\text{且}\mathop{\wedge}_{i=1}^{m}(\forall\hat{\boldsymbol{x}},(\hat{\boldsymbol{x}}\in\Delta_{n+1}\Rightarrow M_i^*(\hat{\boldsymbol{x}})\geqslant 0)).$$

根据命题 2 及其注解,上式等价于

$$\forall\hat{\boldsymbol{x}},(\hat{\boldsymbol{x}}\in\Omega_{\hat{\boldsymbol{x}}}^o\Rightarrow\hat{\rho}^*(\hat{\boldsymbol{x}})\geqslant 0)\text{且}\mathop{\wedge}_{i=1}^{m}(\forall\hat{\boldsymbol{x}},(\hat{\boldsymbol{x}}\in\Omega_{\hat{\boldsymbol{x}}}^o\Rightarrow M_i^*(\hat{\boldsymbol{x}})\geqslant 0)).$$

也即 $\forall\hat{\boldsymbol{x}},(\hat{\boldsymbol{x}}\in\Omega_{\hat{\boldsymbol{x}}}^o\Rightarrow\hat{\rho}^*(\hat{\boldsymbol{x}})\geqslant 0)\text{且}\mathop{\wedge}_{i=1}^{m}(\forall\hat{\boldsymbol{x}},(\hat{\boldsymbol{x}}\in\Omega_{\hat{\boldsymbol{x}}}^o\Rightarrow M_i^*(\hat{\boldsymbol{x}})=M_i^*(\boldsymbol{x},z)=\hat{H}_i^*(\hat{\boldsymbol{x}})-c^*\cdot z^{d_{H_i}}\geqslant 0))$

在上式中令 z=1,得到:

$$\forall\boldsymbol{x},(\boldsymbol{x}\in\Omega_{\boldsymbol{x}}\Rightarrow\rho^*(\boldsymbol{x})\geqslant 0)\text{且}\mathop{\wedge}_{i=1}^{m}(\forall\boldsymbol{x},(\boldsymbol{x}\in\Omega_{\boldsymbol{x}}\Rightarrow\rho^*(\boldsymbol{x})-\rho^*(T_i(\boldsymbol{x}))-c^*\geqslant 0)).$$

根据定义 1 可知,满足上式的$\rho^*(\boldsymbol{x})$是程序 U 的秩函数.故程序 U 可终止. □

注:根据定理 6,若系统 *Sys* 有解,则能够得到一个最小值为预设定值λ^*的具有预定形式的秩函数;反之,若系统 *Sys* 无解,则表明在单形上没有最小值为预设定值λ^*且具有预定形式的正定多项式.

根据定理 4, $\hat{\rho}^*(\hat{\boldsymbol{x}})$ 的存在,表明了程序 U 是终止的.综上所述,N 的计算是关键的.在公式(17)中,N 的值仅依赖于多项式的次数 d 和齐次多项式在单形上的最小值λ.但由于 $\hat{\rho}(\hat{\boldsymbol{x}}),M_i(\hat{\boldsymbol{x}})$ 的所有系数均是关于参数$(\boldsymbol{a},\boldsymbol{b_a},c)$的线性表达式,故它们在单形$\Delta_{n+1}$ 上的最小值是关于$(\boldsymbol{a},\boldsymbol{b_a},c)$的函数,即 $\lambda_{\hat{\rho}(\hat{\boldsymbol{x}})}=\lambda(\boldsymbol{a},\boldsymbol{b_a},c),\lambda_{M_i(\hat{\boldsymbol{x}})}=\lambda(\boldsymbol{a},\boldsymbol{b_a},c)$.因此在实际的计算中,我们需事先给定 $\lambda_{\hat{\rho}(\hat{\boldsymbol{x}})},\lambda_{M_i(\hat{\boldsymbol{x}})}$ 的值,然后根据公式(17)计算得到 N 的值并抽取 $(x_1+...+x_n+z)^N\hat{\rho}(\hat{\boldsymbol{x}}),(x_1+...+x_n+z)^N M_i(\hat{\boldsymbol{x}})$ 的所有系数构造上述的不等式组 *Sys*.再根据定理 6,若 *Sys* 有实解,则获得一个满足预定模板形式的秩函数;但若该不等式组没有实解,则需要再次设定新的最小值 $\lambda_{\hat{\rho}(\hat{\boldsymbol{x}})},\lambda_{M_i(\hat{\boldsymbol{x}})}$,并使其值小于上一次设定的最小值(这是因为根据公式(17)可知,当最小值越小时,N 所需的下界值越大).因此,上述秩函数的探测过程是试探性(heuristic)的.当然,为了保证探测过程终止,可人为固定探测的深度 *depth*.根据上述过程,我们建立下列算法 1.

算法 1.

输入:一个程序 U,探测深度 *depth*, $\lambda_{\hat{\rho}(\hat{\boldsymbol{x}})}=\lambda_{M_i(\hat{\boldsymbol{x}})}=\lambda>0$, 变元个数 n,次数 d.

输出:一个具有预定形式的 n 元 d 次多项式秩函数.

Step 1: 设定多项式秩函数模板$\rho(\boldsymbol{x})=\boldsymbol{a}^T\cdot\Gamma$

Step 2: 构造齐次多项式 $\hat{\rho}(\hat{\boldsymbol{x}})=\boldsymbol{a}^T\cdot\Gamma_0,M_i(\hat{\boldsymbol{x}})=\hat{H}_i(\hat{\boldsymbol{x}})-c\cdot z^{d_{H_i}}$

Step 3: **For** i=1 to *depth*

Step 3.1: 根据 $\lambda_{\hat{\rho}(\hat{\boldsymbol{x}})},\lambda_{M_i(\hat{\boldsymbol{x}})}$ 的值以及公式(17)分别计算 $N_{\hat{\rho}},N_{M_i}$

Step 3.2: 抽取 $(x_1+...+x_n+z)^{N_\rho}\hat{\rho}(\hat{\boldsymbol{x}}),(x_1+...+x_n+z)^{N_{M_i}}M_i(\hat{\boldsymbol{x}})$ 的所有系数构造不等式组 *Sys*

Step 3.3: 用线性规划工具 Simplex 计算 *Sys*:若 *Sys* 有解,则输出多项式秩函数$\rho^*(\boldsymbol{x})$;否则,转 Step 3.4

Step 3.4: 令 $\lambda := \frac{\lambda}{2}; \lambda_{\hat{\rho}(\hat{\boldsymbol{x}})} := \lambda; \lambda_{M_i(\hat{\boldsymbol{x}})} := \lambda$ (串行赋值).

下面通过一个例子来阐述本文的方法.

例 1:考虑循环程序:

$$\left.\begin{array}{l} U_1\ \textbf{while}\ x \geqslant 0 \wedge y \geqslant 0\ \textbf{do} \\ \tau_1 : x := 1 - 3x - 4y - x^3; \\ y := -x - y; \\ \text{or} \\ \tau_2 : x := -y - 1; \\ y := -7x^2 + y; \end{array}\right\} \tag{18}$$

记 $T_1=(1-3x-4y-x^3,-x-y), T_2=(-y-1,-7x^2+y)$.

(a) 设定秩函数模板$\rho(\boldsymbol{x})=a_1x^3+a_2x^2y+a_3xy^2+a_4x^2+a_5xy+a_6y^2+a_7x+a_8y+a_9$,令

$$H_1(\boldsymbol{x})=\rho(\boldsymbol{x})-\rho(T_1(\boldsymbol{x})), H_2(\boldsymbol{x})=\rho(\boldsymbol{x})-\rho(T_2(\boldsymbol{x})).$$

(b) 对 $H_i(\boldsymbol{x})-c$ 进行齐次化,得到 $\hat{\rho}(\hat{\boldsymbol{x}}), M_1(\hat{\boldsymbol{x}}), M_2(\hat{\boldsymbol{x}})$,其中,

$$\hat{\rho}(x,y,z) = a_1x^3 + a_2x^2y + a_3xy^2 + a_4x^2z + a_5xyz + a_6y^2z + a_7xz^2 + a_8yz^2 + a_9z^3.$$

由于 $M_1(\hat{\boldsymbol{x}}), M_2(\hat{\boldsymbol{x}})$ 表达式过长,在此省略.

(c) 判定是否存在$(a_1,...,a_9)$的一组取值 $(a_1^*,...,a_9^*)$,使得 $\hat{\rho}(x,y,z)$ 和 $M_1(x,y,z), M_2(x,y,z)$在单形Δ_{2+1}上都同时正定.要计算得到那样的一组$(a_1,...,a_9)$的取值,根据定理 5,需首先计算出 N 的值.根据公式(17),N 依赖于 $\hat{\rho}, M_1, M_2$ 各自次数以及各自在Δ_3 上的最小值λ.但这里, $\hat{\rho}(x,y,z)$ 和 $M_1(x,y,z), M_2(x,y,z)$均是参系数齐次多项式,故其在单形上的最小值是随着参数取值的变动而变动的.因此,在 N 的实际计算中,我们需要事先固定λ的值,比如可令 $\lambda_{\hat{\rho}} = \lambda_{M_1} = \lambda_{M_2} = \lambda = \frac{1}{3}$,然后,既然该例中 $\hat{\rho}(x,y,z)$ 和$M_1(x,y,z), M_2(x,y,z)$的次数分别为3,9,5,那么根据公式(17),分别计算对应的 N 的取值范围为

$$N_{\hat{\rho}} > N\left(\hat{\rho}, \lambda = \frac{1}{3}\right) = 6,\ N_{M_1} > N\left(M_1, \lambda = \frac{1}{3}\right) = 99,\ N_{M_2} > N\left(M_2, \lambda = \frac{1}{3}\right) = 25.$$

故取 $N_{\hat{\rho}} = 7, N_{M_1} = 100, N_{M_2} = 26$ (当然,我们也可以令 $N_{\hat{\rho}} = N_{M_1} = N_{M_2} = 100$,即取公共界).根据定理 5,分别展开多项式 $(x+y+z)^{N_{\hat{\rho}}}\hat{\rho}(x,y,z), (x+y+z)^{N_{M_1}}M_1(x,y,z), (x+y+z)^{N_{M_2}}M_2(x,y,z)$, 合并同类项后,各自抽取出关于 x,y,z 项的所有系数(均为关于 $a_1,...,a_9,c$ 的线性表达式),然后再令所有系数大于 0,构造出关于 $a_1,...,a_9,c$ 的严格不等式组(为参系数的第 1 个约束系统),分别记为 $coe_{\hat{\rho}}, coe_{M_1}, coe_{M_2}$. 同时,注意定理 5 成立的前提条件是, $\hat{\rho}, M_1, M_2$ 关于 x,y,z 的所有系数都被限定在[−1,1]时,公式(17)才成立——即 N 的值才仅与其次数 d 和其在单形上的最小值λ相关.因此,再次分别从多项式 $\hat{\rho}, M_1, M_2$ 中提取其关于 x,y,z 的所有系数(均为关于 $a_1,...,a_9,c$ 的线性表达式),然后分别令每个系数≥−1 且≤1,由此构造出 $\hat{\rho}, M_1, M_2$ 关于 x,y,z 的系数的第 2 个约束系统,分别记为 $\Theta_{\hat{\rho}}, \Theta_{M_1}, \Theta_{M_2}$. 比如,在该例中,我们有:

$$\Theta_{\hat{\rho}} := a_1 \geqslant -1 \wedge a_1 \leqslant 1 \wedge a_2 \geqslant -1 \wedge a_2 \leqslant 1 \wedge a_3 \geqslant -1 \wedge a_3 \leqslant 1 \wedge a_4 \geqslant -1 \wedge a_4 \leqslant 1 \wedge a_5 \geqslant -1 \wedge a_5 \leqslant 1 \wedge$$
$$a_6 \geqslant -1 \wedge a_6 \leqslant 1 \wedge a_7 \geqslant -1 \wedge a_7 \leqslant 1 \wedge a_8 \geqslant -1 \wedge a_8 \leqslant 1 \wedge a_9 \geqslant -1 \wedge a_9 \leqslant 1.$$

同时,根据秩函数的定义 4 可知,$c>0$.最后,求解不等式组 $Sys := coe_{\hat{\rho}} \wedge coe_{M_1} \wedge coe_{M_2} \wedge c > 0 \wedge \Theta_{\hat{\rho}} \wedge \Theta_{M_1} \wedge \Theta_{M_2}$. 若 *Sys* 有解,则表明该程序具有预定形式的 3 次秩函数.根据定理 4 可知,该程序是终止的.但是因为 Maple 自带的线性规划工具 Simplex 仅能够求解非严格的不等式组——不等式组中的所有不等式都是非严格的,所以为了使用工具 Simplex,在实际计算中,我们让 $(x+y+z)^{N_{\hat{\rho}}}\hat{\rho}(x,y,z), (x+y+z)^{N_{M_1}}M_1(x,y,z), (x+y+z)^{N_{M_2}}M_2(x,y,z)$, 合并同类项后,各自抽取出关于 x,y,z 项的所有系数,然后再令所有系数≥某个正数δ.这相当于对 $coe_{\hat{\rho}}, coe_{M_1}$,

coe_{M_2} 中的所有不等式做了一个小小的扰动,即将>0 替换为≥δ(δ>0).记扰动后的系数集为 $\widetilde{coe}_{\hat{\rho}},\widetilde{coe}_{M_1},\widetilde{coe}_{M_2}$.同时,将 c>0 扰动为 $c\geqslant\delta$.显然,扰动后的系统 $\widetilde{Sys}:=\widetilde{coe}_{\hat{\rho}}\wedge\widetilde{coe}_{M_1}\wedge\widetilde{coe}_{M_2}\wedge c\geqslant\delta\wedge\Theta_{\hat{\rho}}\wedge\Theta_{M_1}\wedge\Theta_{M_2}$ 为非严格不等式组,如果它有解,则原系统 Sys 也必有解.为方便,我们让 $Sys,\widetilde{Sys}$ 不仅表示不等式系统,而且还表示不等式系统所对应的解集.在本例中,我们取 $\delta=\frac{1}{1000}$.利用 Maple 中的线性规划工具包 Simplex,求解不等式组 $\widetilde{Sys}$,得到:

$$\left\{\begin{array}{l}a_1=\frac{29}{7000},a_2=\frac{51}{12250},a_3=\frac{1}{980},a_4=-\frac{43}{7000},a_5=0,a_6=\frac{1}{1000},a_7=0,\\ a_8=\frac{955875294522846847882517}{154330444367487258828048000},a_9=\frac{9}{875},c=\frac{1}{1000}\end{array}\right\} \tag{19}$$

由于采用单纯形算法,故该点是满足 $\widetilde{Sys}$ 的一个精确点而非浮点向量.因此,我们可以得到程序 U_1 的一个 3 次秩函数 $\rho(x,y)=\frac{29x^3}{7000}+\frac{51x^2y}{12250}+\frac{xy^2}{980}-\frac{43x^2}{7000}+\frac{y^2}{1000}+\frac{955875294522846847882517}{154330444367487258828048000}y+\frac{9}{875}$.由于单纯形算法给出的是满足系统 $\widetilde{Sys}(\subseteq Sys)$ 的精确解而非数值解,故求得的函数必然是秩函数,因而不必再对其验证是否为秩函数.该秩函数的存在表明,该循环程序是终止的.

注:对于该程序,文献[3,5,6,20]中的方法已不能计算其预定形式的 3 次秩函数.因为这些方法仅能计算线性循环程序的线性秩函数,而本例中的循环是非线性的,且所要计算的秩函数也是 3 次的.此外,对该程序,我们也尝试利用一些基于量词消去技术[22]的工具,如 Redlog、RegularChains,去计算该程序的预定形式的 3 次秩函数(其中,RegularChains 集成了当前功能强大的实解分类工具 DISCOVERER[23],作者此前的诸多工作也是在该工具的支持下予以开展).这等价于利用这些量词消去工具从秩函数定义 4 中的两个蕴含公式(8)和公式(9)中消去变元 x,y 即可.但由于量词消去算法的双指数复杂度以及所处理的系统都是非线性的,从而使得这两个工具均无法计算得到该程序的 3 次秩函数.同时,我们也尝试用 Cousot 在文献[9]中的方法——半正定规划(SDP),并借助半正定规划求解器 YALMIP[24]计算得到一个函数:

$$42.87498608+4.1519x+25.9747y+1.5329x^3-0.8212y^2-8.7046x^2+3.9678xy-1.8358x^2y+1.8393xy^2.$$

但经过检验发现,该函数并不满足秩函数定义中的有界条件,因此它不是程序的秩函数.

例 2:考虑循环程序:

U_2 **while** $x\geqslant 0\wedge y\geqslant 0$ **do**
$\tau_1: x:=1+x-y;$
$y:=2x+y+3;$
or
$\tau_2: x:=x-y;$
$y:=y+2;$

通过本文方法,我们得到该循环的一个 3 次秩函数:

$$\rho(x,y)=\frac{56978573}{514034040000}x^3-\frac{10604929}{257017020000}x^2y+\frac{1155787}{79082160000}xy^2-\frac{16927733}{171344680000}xy+\frac{1}{52000}y^2+\frac{7408419}{26360720000}x-\frac{7}{13000}y+\frac{203}{26000}.$$

例 3:考虑循环程序:

$$\left.\begin{array}{l}U_3\ \textbf{while}\ x\geqslant 0\wedge y\geqslant 0\ \textbf{do}\\ \tau_1: x:=-x-3y+2;\\ y:=-4x^2+y-1;\\ \text{or}\\ \tau_2: x:=x-1;\\ y:=-y^2+x-3;\end{array}\right\} \tag{20}$$

通过本文方法,我们得到该循环的一个 3 次秩函数 $\rho(x,y)=\frac{1}{4600}x^3+\frac{63}{23000}y$.

基于 Polya 定理,算法 1 提供了一个试探性方法去探测程序 U 的多项式秩函数.下面我们将证明,给定齐 d 次 n 元多项式 $f(\boldsymbol{x})=\sum_{|\alpha|=d}a_\alpha\boldsymbol{x}^\alpha=\sum_{|\alpha|=d}c(\alpha)b_\alpha\boldsymbol{x}^\alpha$, 若

(A) 所有系数均为整数,即 $a_\alpha\in\mathbf{Z}$,

(B) 其系数的绝对值被界定,即存在正数 η,使得 $|a_\alpha|\leqslant\eta$,

则 Polya 定理中关于 N 的下界公式可以被改写为

$$N>\frac{d(d-1)}{2}\frac{\eta}{\sigma(n,d,\eta)}-d \tag{21}$$

显然,上述 N 的下界公式仅依赖于多项式 $f(\boldsymbol{x})$的次数、变元个数以及系数的绝对值上界.根据公式(21),我们将建立不同于算法 1 的新算法.在证明上述结论之前,我们首先引入文献[25]中建立的关于整系数正定多项式在单形上的最小值下界的一个重要结果.

定理 7[25]. 给定 d 次 n 元齐次整系数多项式 $f(\boldsymbol{x})=\sum_{|\alpha|=d}a_\alpha\boldsymbol{x}^\alpha=\sum_{|\alpha|=d}c(\alpha)b_\alpha\boldsymbol{x}^\alpha\in Z[\boldsymbol{x}]$,其系数均被正数 η 界定,即 $|a_\alpha|\leqslant\eta$.记 $\Delta_n=\{(x_1,...,x_n):x_i\geqslant 0,\sum_{i=1}^n x_i=1\}$ 为单形.如果 $f(\boldsymbol{x})$在单形 Δ_n 上是正定的,那么有:

$$\min_{\Delta_n}f(\boldsymbol{x})\geqslant(2\eta)^{-d^n}n^{-d^{n+1}-d}d^{-nd^n} \tag{22}$$

注:因本文仅考虑单形上的正定多项式,故定理 7 只引述了文献[25]中引理 3.3 中的前部分结论,其后部分所涉及到单形上的负定多项式的内容在此被舍去.

根据定理 2、定理 3 及其定理 7,我们建立下列定理 8.

定理 8. 给定 d 次 n 元齐次整系数多项式 $f(\boldsymbol{x})=\sum_{|\alpha|=d}a_\alpha\boldsymbol{x}^\alpha=\sum_{|\alpha|=d}c(\alpha)b_\alpha\boldsymbol{x}^\alpha\in\mathbf{Z}[\boldsymbol{x}]$,且其系数均被正数 η 界定,即 $|a_\alpha|\leqslant\eta$.如果 f 在 Δ_n 上是正定的,那么当

$$N>\frac{d(d-1)}{2}\frac{\eta}{\sigma(n,d,\eta)}-d \tag{23}$$

时,有 $(x_1+\ldots+x_n)^N f(x_1,\ldots,x_n)$的所有系数为正.这里, $\sigma(n,d,\eta)=(2\eta)^{-d^n}n^{-d^{n+1}-d}d^{-nd^n}$.

证明:首先,根据定理 3 可知,如果 n 元齐 d 次多项式 $f(\boldsymbol{x})$在单形上正定,那么存在正整数 N,当其满足:

$$N>\frac{d(d-1)}{2}\frac{L}{\lambda}-d \tag{24}$$

时,有 $(x_1+\ldots+x_n)^N f(x_1,\ldots,x_n)$的所有系数为正.其中,$L=L(f)=\max\{|b_\alpha|:|\alpha|=d\}$,$\lambda=\lambda(f)=\min\{f(\boldsymbol{x}):\boldsymbol{x}\in\Delta_n\}$.再根据命题 4 及其注解可知,倘若 f 的所有系数 a_α均在区间 $[-\eta,\eta]$中,那么,

$$L=L(f)=\max\left\{|b_\alpha|=\frac{|a_\alpha|}{c(\alpha)}:|\alpha|=d\right\}\leqslant\eta \tag{25}$$

同时,根据已知题设,既然 f 在单形上正定且其所有系数均为整数并均被正数 η 界定,那么由定理 7 可得,f 在单形 Δ_n 上的最小值:

$$\lambda_f=\min_{\Delta_n}f(\boldsymbol{x})\geqslant(2\eta)^{-d^n}n^{-d^{n+1}-d}d^{-nd^n} \tag{26}$$

根据公式(25)和公式(26)将公式(24)进行放缩,即得到公式(23). □

注:定理 8 中,公式(23)中 N 的下界公式仅仅依赖多项式的次数、变元个数以及系数绝对值上界.

给定带参数的多项式 $G(y_1,...,y_n)=\sum_{|\alpha|=d}g_\alpha(\boldsymbol{v})\boldsymbol{y}^\alpha$ (这里,$\boldsymbol{v}$ 为参数向量,$g_\alpha(\boldsymbol{v})$为参数 $\boldsymbol{v}$ 的齐次整系数线性函数),倘若我们限定其系数 $g_\alpha(\boldsymbol{v})$在某个区间取值,即 $-\eta\leqslant g_\alpha(\boldsymbol{v})\leqslant\eta$,那么根据定理 8,完成下列两个步骤便可以构造出使得多项式 G 在单形 Δ_n 上正定且其系数满足 $|g_\alpha(\boldsymbol{v})|\leqslant\eta$的参数 $\boldsymbol{v}$ 应满足的必要条件 $Sys_G^{\mathbf{z}}$ (与之前的 Sys 以示区别).

(i) 根据公式(23)计算 N 的值,记为 N_G;

(ii) 抽取 $(y_1+...+y_n)^{N_G}G(y_1,...,y_n)$ 的所有系数 $g_\alpha(\boldsymbol{v})$,分别令 $g_\alpha(\boldsymbol{v})>0$,且$|g_\alpha(\boldsymbol{v})|\leqslant\eta$,构造不等式组 $Sys_G^{\mathbf{z}}$.

这里,$g_\alpha(\boldsymbol{v})$为齐次整系数线性函数这一限定具有一般性.这是因为,若多项式 G 的系数均是有理数,那么总可以通过乘上所有有理数的分母将其系数变为整数.根据公式(5),程序 U 中的所有系数均为有理数,故参数模板多项式 $\hat{\rho}(\hat{\boldsymbol{x}})=\boldsymbol{a}^T\cdot\Gamma_0, M_i(\hat{\boldsymbol{x}})=\sum_{|\alpha|=d_{M_i}}g_\alpha(\boldsymbol{b_a},\boldsymbol{c})\hat{\boldsymbol{x}}^\alpha$ 的所有系数均为有理数.因此,可以将所有有理数系数的分母(若有负号,则将负号放到分子上)都乘在一起,记为β.则在公式(14)中的不等式两端同时乘上β并不会改变不等式的符号,故有 $\beta\cdot\hat{\rho}(\hat{\boldsymbol{x}}),\beta\cdot M_i(\hat{\boldsymbol{x}})$ 的系数均为整数.也即$\beta\cdot g_\alpha(\boldsymbol{b_a},\boldsymbol{c})$是关于参数 $\boldsymbol{b_a},\boldsymbol{c}$ 的齐次整系数线性函数.因此,若 $\hat{\rho}(\hat{\boldsymbol{x}}),M_i(\hat{\boldsymbol{x}})$ 含有有理系数,则可以乘上一个正数β,使得其所有系数为整数.根据公式(23),分别计算 $N_{\hat{\rho}},N_{M_i}$ 并抽取 $(x_1+...+x_n+z)^{N_{\hat{\rho}}}\hat{\rho}(\hat{\boldsymbol{x}}),(x_1+...+x_n+z)^{N_{M_i}}M_i(\hat{\boldsymbol{x}})$ 的所有系数分别构造不等式组 $Sys_{\hat{\rho}}^{\mathbf{z}},Sys_{M_i}^{\mathbf{z}}$. 令 $Sys_{\mathbf{z}}=Sys_{\hat{\rho}}^{\mathbf{z}}\wedge(\wedge_{i=1}^{m}Sys_{M_i}^{\mathbf{z}})$.既然所有参数均被设定为整数,故需要在整数环上求解系统 $Sys_{\mathbf{z}}$.类似定理 6,下面的定理 9 表明,若 $Sys_{\mathbb{Z}}$ 有整数解,则程序 U 是终止的.

定理 9. 记号同上.给定程序 U.若系统 $Sys_{\mathbf{z}}$ 有整数解,那么程序 U 有预定形式的且系数绝对值上界为η的秩函数.

证明:该证明完全类似于定理 6 的证明.在此省略. □

注:定理 9 表明,若系统 $Sys_{\mathbf{z}}$ 有整数解,则程序 U 有一个整系数多项式秩函数;反之,如果系统 $Sys_{\mathbf{z}}$ 没有整数解,则表明在单形上没有预定形式的且系数在$[-\eta,\eta]$的正定多项式.此时需要增大上界η的值继续构造新的系统并求解.因此,整个过程仍是试探性的.

算法 2.

输入:一个程序 U,变元个数 n,次数 d,系数绝对值上界η.

输出:一个具有预定形式的 n 元 d 次整系数多项式秩函数;不确定(unknown).

Step 1: 设定多项式秩函数模板$\rho(\boldsymbol{x})=\boldsymbol{a}^T\cdot\Gamma$

Step 2: 构造齐次多项式 $\hat{\rho}(\hat{\boldsymbol{x}})=\boldsymbol{a}^T\cdot\Gamma_0, M_i(\hat{\boldsymbol{x}})=\hat{H}_i(\hat{\boldsymbol{x}})-c\cdot z^{d_{H_i}}$

Step 3: 如果 $\hat{\rho}(\hat{\boldsymbol{x}}),M_i(\hat{\boldsymbol{x}})$ 含有有理系数,则将所有有理数系数的分母都乘在一起,记为β.令 $\hat{\rho}(\hat{\boldsymbol{x}}):=\beta\cdot\hat{\rho}(\hat{\boldsymbol{x}})$, $M_i(\hat{\boldsymbol{x}}):=\beta\cdot M_i(\hat{\boldsymbol{x}})$,转 Step 4;

Step 4: 根据公式(23)分别计算 $N_{\hat{\rho}},N_{M_i}$

Step 5: 抽取 $(x_1+...+x_n+z)^{N_{\hat{\rho}}}\hat{\rho}(\hat{\boldsymbol{x}}),(x_1+...+x_n+z)^{N_{M_i}}M_i(\hat{\boldsymbol{x}})$ 的所有系数构造分别构造不等式组:

$$Sys_{\hat{\rho}}^{\mathbf{z}},Sys_{M_i}^{\mathbf{z}}$$

Step 6: 利用整数线性规划算法计算 $Sys_{\mathbf{z}}=Sys_{\hat{\rho}}^{\mathbf{z}}\wedge(\wedge_{i=1}^{m}Sys_{M_i}^{\mathbf{z}})$ 中的一个整数解.若 $Sys_{\mathbf{z}}$ 有整数解,则输出整系数多项式秩函数$\rho^*(\boldsymbol{x})$;否则,输出 unknown

注:算法 2 中,我们仅是简单地使 $\hat{\rho}(\hat{\boldsymbol{x}}),M_i(\hat{\boldsymbol{x}})$ 的所有系数绝对值具有相同的上界η.实际上,可以对它们设置不同的系数绝对值上界.在判定 $Sys_{\mathbf{z}}$ 是否有整数解时,可以利用 Pugh 在文献[26]中提出的 Omega test 方法.

3 总 结

针对一类多项式循环程序,本文给出了一种新的方法去计算这类程序的多项式秩函数.该方法将这类程序的秩函数计算归结为单形上的正定多项式的探测问题;然后,利用 Polya 定理,将单形上的正定多项式探测问题归结为线性不等式约束系统的可行问题.从这一角度看,我们的方法是一个“线性化”的方法.而线性不等式系统的可行问题则可以利用(整数)线性规划工具进行求解.相对于现有诸如 Redlog、RegularChains 等基于柱形代数分解的量词消去工具,本文的算法 1 可以在可接受时间内进行复杂秩函数的计算.同时,也不同于基于 SDP 的方法,通过本文方法计算得到的函数是精确的秩函数,因此不必再次验证计算所得的函数是否为秩函数.

致谢 感谢匿名审稿人对本文工作提出的宝贵意见.同时,也感谢中国科学院成都计算机应用研究所的杨路先

生对本文的修改提供的好建议.

References:

[1] Cook B, Podelski A, Rybalchenko A. Proving program termination. Communications of the ACM, 2011,54(5):88−98.

[2] Chen YH, Xia BC, Yang L, Zhou CC. Discovering non-linear ranking functions by solving semi-algebraic systems. In: Proc. of the 4th Int'l Colloquium on Theoretical Aspects of Computing. Berlin, Heidelberg: Springer-Verlag, 2007. 34−49.

[3] Colo´n M, Sipma HB. Practical methods for proving program termination. In: Proc. of the Computer Aided Verification. Berlin, Heidelberg: Springer-Verlag, 2002. 227−240.

[4] Podelski A, Rybalchenko A. A complete method for the synthesis of linear ranking functions. In: Proc. of the 5th Int'l Conf. on Verification, Model Checking, and Abstract Interpretation. Berlin, Heidelberg: Springer-Verlag, 2004. 239−251.

[5] Bradley A, Manna Z, Sipma H. The polyranking principle. In: Caires L, Italiano G, Monteiro L, Palamidessi C, Yung M, eds. Proc. of the Automata, Languages and Programming. Berlin, Heidelberg: Springer-Verlag, 2005. 1349−1361.

[6] Bradley A, Manna Z, Sipma H. Linear ranking with reachability. In: Proc. of the Int'l Conf. on Computer Aided Verification. Berlin, Heidelberg: Springer-Verlag, 2005. 491−504.

[7] Bagnara R, Mesnard F, Pescetti A, Zaffanella E. A new look at the automatic synthesis of linear ranking functions. Information and Computation, 2012,215:47−67.

[8] Ben-Amram AM, Genaim S. On the linear ranking problem for integer linear-constraint loops. In: Proc. of the 40th Annual ACM SIGPLAN-SIGACT Symp. on Principles of Programming Languages. New York: ACM Press, 2013. 51−62.

[9] Cousot P. Proving program invariance and termination by parametric abstraction, langrangian relaxation and semidefinite programming. In: Proc. of the 6th Int'l Conf. on Verification, Model Checking, and Abstract Interpretation. Berlin, Heidelberg: Springer-Verlag, 2005. 1−24.

[10] Yang L, Zhou CC, Zhan NJ, Xia BC. Recent advances in program verification through computer algebra. Frontiers of Computer Science in China, 2010, 4(1):1−16.

[11] Chen HY, Cook B, Fuhs C, Nimkar K, *et al*. Proving nontermination via safety. In: Proc. of the 20th Int'l Conf. on Tools and Algorithms for the Construction and Analysis of Systems. Berlin, Heidelberg: Springer-Verlag, 2014. 156−171.

[12] Gupta A, Henzinger T, Majumdar R, *et al*. Proving non-termination. In: Proc. of the 35th Annual ACM SIGPLAN-SIGACT Symp. on Principles of Programming Languages. New York: ACM Press, 2008. 147−158.

[13] Tiwari A. Termination of linear programs. In: Proc. of the 16th Int'l Conf. on Computer Aided Verification. Berlin, Heidelberg: Springer-Verlag, 2004. 70−82.

[14] Braverman M. Termination of integer linear programs. In: Proc. of the 18th Int'l Conf. on Computer Aided Verification. Berlin, Heidelberg: Springer-Verlag, 2006. 372−385.

[15] Xia BC, Yang L, Zhan NJ, Zhang ZH. Symbolic decision procedure for termination of linear programs. Formal Aspects of Computing, 2009,23(2):171−190.

[16] Bradley A, Manna Z, Sipma H. Termination of polynomial programs. In: Proc. of the 6th Int'l Conf. on Verification, Model Checking, and Abstract Interpretation. Berlin, Heidelberg: Springer-Verlag, 2005. 113−129.

[17] Xia BC, zhang ZH. Termination of linear programs with nonlinear constraints. Journal of Symbolic Computation, 2010,45(11): 1234−1249.

[18] Babic D, Cook B, Hu AJ, *et al*. Proving termination of nonlinear command sequences. Formal Aspects of Computing, 2013,25(3): 389−403.

[19] Liu J, Xu M, Zhan NJ, Zhao HJ. Discovering non-terminating inputs for multi-path polynomial programs. Journal of Systems Science and Complexity, 2014,27(4):1284−1304.

[20] Colo´n M, Sipma HB. Synthesis of linear ranking functions. In: Proc. of the 7th Int'l Conf. on Tools and Algorithms for the Construction and Analysis of Systems. Berlin, Heidelberg: Springer-Verlag, 2001. 67−81.

[21] Powers V, Reznick B. A new bound for Polya's theorem with applications to polynomials positive on polyhedra. Journal of Pure and Applied Algebra, 2001,164(1-2):221−229.

[22] Collins GE. Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In: Proc. of the Automata Theory and Formal Languages. Berlin, Heidelberg: Springer-Verlag, 1975. 134−165.

[23] Yang L, Xia BC. Mechanical Inequality Proving and Automated Discovering. Beijing: Science Press, 2008 (in Chinese).

[24] Löfberg J. YALMIP: A MATLAB toolbox for rapid prototyping of optimization problems. Automatic Control Laboratory, Eidgenössische Technische Hochschule Zürich, 2004. http://control.ee.ethz.ch/joloef/yalmip.msql

[25] Hou XR, Shao JW. Bounds on the number of steps of WDS required for checking the positivity of integral forms. Applied Mathematics and Computation, 2011,217(2):9978−9984.

[26] Pugh W. The omega test: A fast and practical integer programming algorithm for dependence analysis. In: Martin JL, ed. Proc. of the Supercomputing. ACM Press, 1991. 4−13.

附中文参考文献:

[23] 杨路,夏壁灿.不等式机器证明与自动发现.北京:科学出版社,2008.

李轶(1980－),男,重庆人,博士,副研究员,CCF 专业会员,主要研究领域为程序验证,符号计算.

冯勇(1965－),男,博士,研究员,博士生导师,CCF 专业会员,主要研究领域为符号数值计算.