第 29 卷 第 10 期
2006 年 10 月
计 算 机 学 报
CHINESE JOURNAL OF COMPUTERS
Vol. 29 No. 10
Oct. 2006

一类半正定多项式的平方和分解及其表达式的自动生成

姚 勇 冯 勇
（中国科学院成都计算机应用研究所 成都 610041）

摘 要 建立了一个把半正定稀疏多项式表为多项式平方和的算法.这一算法依赖于 Hilbert 第 17 问题的一系列经典研究结果以及实闭域上量词消去的柱形代数剖分算法.该算法的机器实现为一类代数不等式可读性证明的自动生成提供了一种非常自然的途径.

关键词 平方和分解;柱形代数剖分;Gram 矩阵方法;Hilbert 17 问题
中图法分类号 TP301

The Sum of Squares Decomposition and Automated Production for a Class of Positive Semi-Definite Polynomials

YAO Yong FENG Yong
(*Chengdu Institute of Computer Applications, Chinese Academy of Sciences, Chengdu* 610041)

Abstract In this paper, an algorithm is presented for a positive semi-definite sparse polynomials to be decomposed into sum of squares. The algorithm is based on a series of research results of Hilbert 17th problem and Cylindrical Algebraic Decomposition. Its implementation on computer provides a nature approach for the automated production of readable proof for a class of algebraic inequalities.

Keywords sum of squares decomposition; cylindrical algebraic decomposition; Gram matrix method; Hilbert 17th problem

1 引 言

首先来考虑这样一个问题:如何判别二元六次型 $f(x,y)=x^6+y^6+2x^5y+4xy^5+5x^2y^4$ 在 $(x,y)\in\mathbb{R}^2$ 是半正定的? 即 $f(x,y)\geq 0$. 判别的方法有很多,比如多项式根的完全判别系统方法[1]以及基于胞腔分解的实闭域上量词消去方法[2]等,事实上,这两种方法都是一般性的判别方法,其形式非常漂亮,特别是后者,由于杨路及其合作者的工作,现已发展成为证明代数不等式的强有力工具,"Bottema"是杨路用 Maple 编程实现的不等式证明软件.但是这些方法从证明的可读性方面来讲则较弱(需要较多的专业基础知识).

如果我们能够直接将 $f(x,y)$ 表示为多项式的平方和的形式,那么判别 $f(x,y)$ 的半正定性将是十分显然的.下面是这个问题的大致处理过程:

假设 $f(x,y)$ 可以表为多项式的平方和,即 $\sum h_i^2$ 的形式,这里 h_i 是齐三次多项式;易见 h_i 是所有的单项三次型 x^3, x^2y, yx^2, y^3 的线性组合.

记向量 $\boldsymbol{X}^{\mathrm{T}}=(x^3, x^2y, yx^2, y^3)$,并设 $f(x,y)=$

收稿日期:2005-03-03;修改稿收到日期:2006-03-06.本课题得到国家"九七三"重点基础研究发展规划项目基金(2004CB318003)资助.
姚 勇,男,1974 年生,主要研究方向为计算机自动推理、智能软件技术.E-mail: yongyaoyong@yahoo.com.cn. **冯 勇**,男,1965 年生,教授,博士生导师,主要研究领域为大规模科学计算、符号数值混合计算技术.

$\boldsymbol{X}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{X}$，其中 $\boldsymbol{B}$ 是如下元素待定的对称矩阵

$$\boldsymbol{B}=\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{12} & a_{22} & a_{23} & a_{24} \\ a_{13} & a_{23} & a_{33} & a_{34} \\ a_{14} & a_{24} & a_{34} & a_{44} \end{bmatrix}.$$

将 $f(x,y)=\boldsymbol{X}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{X}$ 展开后与原多项式对照，得到线性方程组

$$\begin{cases} a_{11}-1=0 \\ 2a_{12}-2=0 \\ 2a_{13}+a_{22}=0 \\ 2a_{34}-4=0 \\ 2a_{24}-5+a_{33}=0 \\ a_{44}-1=0 \\ 2a_{23}-2a_{14}=0 \end{cases},$$

解出这个线性方程组得

$a_{11}=1$，$a_{12}=1$，$a_{22}=-2v$，$a_{34}=2$，$a_{33}=-2w+5$，$a_{44}=1$，$a_{14}=-s$，$a_{13}=v$，$a_{23}=s$，$a_{24}=w$，

其中，v,s,w 是自由变量. 回代入矩阵 $\boldsymbol{B}$，

$$\boldsymbol{B}=\begin{bmatrix} 1 & 1 & v & -s \\ 1 & -2v & s & w \\ v & s & -2w+5 & 2 \\ -s & w & 1 & 1 \end{bmatrix},$$

这个对称矩阵的特征多项式是

$$\begin{aligned} g=&Y^4+(-7+2w+2v)Y^3+ \\ &(6-w^2-4w+4vw-v^2-2s^2-14v)Y^2+ \\ &(5-2ws^2-2s^2v+6w^2+7s^2-8vw+2sv- \\ &2w^3+14v-2v^3-2sw+v^2)Y- \\ &1-2v+v^2w^2+10s^2v+4sw^2+s^4-2vws^2- \\ &6sw+2w-5w^5+3s^2+2sv+2w^3+8sv^2+2v^3. \end{aligned}$$

矩阵 $\boldsymbol{B}$ 是半正定的充要条件是它的特征多项式的所有根非负，根据笛卡尔符号法则（下面定理 3 的推论），则就是要满足以下不等式组

$$\begin{cases} -7+2w+2v\leq 0 \\ 6-w^2-4w+4vw-v^2-2s^2-14v\geq 0 \\ 5-2ws^2-2s^2v+6w^2+7s^2-8vw+2sv-2w^3+ \\ 14v-2v^3-2sw+v^2\leq 0 \\ -1-2v+v^2w^2+10s^2v+4sw^2+s^4-2vws^2- \\ 6sw+2w-5w^2+3s^2+2sv+2w^3+8sv^2+2v^3\geq 0 \end{cases} \tag{$*$}$$

利用胞腔分解算法（见文献[2,3]，实际运算时使用的是不完全胞腔分解算法 PCAD），可以得到不等式组（$*$）的一个特解：$\left\{s=\frac{4}{7},v=\frac{-4}{3},w=\frac{5}{11}\right\}$. 代回矩阵 $\boldsymbol{B}$ 成为

$$\boldsymbol{B}=\begin{bmatrix} 1 & 1 & \frac{-4}{3} & \frac{-4}{7} \\ 1 & \frac{8}{3} & \frac{4}{7} & \frac{5}{11} \\ \frac{-4}{3} & \frac{4}{7} & \frac{45}{11} & 2 \\ \frac{-4}{7} & \frac{5}{11} & 1 & 1 \end{bmatrix},$$

将 $\boldsymbol{B}$ 分解成一个矩阵和它的转置矩阵的乘积，即 $\boldsymbol{B}=\boldsymbol{V}^{\mathrm{T}}\boldsymbol{V}$（这种分解不唯一，方法很多，比如正交化方法，或选主元的 Cholesky 分解方法）.

$$\boldsymbol{V}=\begin{bmatrix} 1 & 1 & \frac{-4}{3} & \frac{-4}{7} \\ 0 & \frac{\sqrt{15}}{3} & \frac{8\sqrt{15}}{21} & \frac{79\sqrt{15}}{385} \\ 0 & 0 & \frac{\sqrt{7271}}{231} & \frac{106\sqrt{7271}}{50897} \\ 0 & 0 & 0 & \frac{\sqrt{670855510}}{254485} \end{bmatrix},$$

这样就得到 $f(x,y)=\boldsymbol{X}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{X}=(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{V}^{\mathrm{T}})(\boldsymbol{V}\boldsymbol{X})=(\boldsymbol{V}\boldsymbol{X})^{\mathrm{T}}(\boldsymbol{V}\boldsymbol{X})=\|(\boldsymbol{V}\boldsymbol{X})\|^2$，

$$\begin{aligned} f(x,y)=&\left(x^3+x^2y-\frac{4}{3}xy^2-\frac{4}{7}y^3\right)^2+ \\ &\left(\frac{\sqrt{15}}{3}x^2y+\frac{8\sqrt{15}}{21}xy^2+\frac{79\sqrt{15}}{285}y^3\right)^2+ \\ &\left(\frac{\sqrt{7271}}{231}xy^2+\frac{106\sqrt{7271}}{50897}y^3\right)^2+\frac{202982y^6}{19595345}. \end{aligned}$$

由于这个方法有几个地方不具有唯一性，所以稍微改变一下过程，就可以得到其它的表示方法. 比如，下面就是另一个表示

$$\begin{aligned} f(x,y)=&\left(y^3+2xy^2-\frac{1}{2}x^3\right)^2+ \\ &\left(xy^2+\frac{1}{2}x^2y-\frac{1}{6}x^3\right)^2+ \\ &\left(\frac{5\sqrt{3}}{6}x^2y+\frac{13\sqrt{3}}{30}x^3\right)^2+\frac{143x^6}{9}. \end{aligned}$$

一个自然的问题是：这个方法是否具有普遍性，即所有半正定的多项式是否都能表示成多项式的平方和. 对于这个问题，我们可以在下面这段历史中找到答案.

1888 年，年仅 26 岁的 Hilbert 证明了如下的定理[4]：n 元 m 次实系数半正定齐次多项式，在以下情况可以表为实系数齐次多项式的平方和（以下简记为 SOS），(i) $n\leq 2$，m 任意；(ii) n 任意，$m=2$；(iii) $n=3$，$m=4$. 对于其它的情况，Hilbert 指出：不一定能表为 SOS.

1893 年 Hilbert 又对三元齐次多项式做了进一步的研究，得到结论：半正定三元齐 m 次多项式可以表为两个齐次多项式平方和之商，并猜想这一结论对一般的半正定齐次多项式也是对的.

1900 年，Hilbert 在巴黎召开的第 2 届国际数学家大会上作了《23 个问题》的著名演讲，演讲中正式提出了他的第 17 问题，即 n 元 m 次实系数半正定多项式，能否表为若干个有理函数的平方和？

1927 年，Artin 证明了上述 Hilbert 的猜想，正面解决了第 17 问题[5].

1928 年，Polya 得到一个定理[6]：一个 n 元齐次多项式 $f(x_1,x_2,\cdots,x_n)$，如果是正定的，则存在某个充分大的自然数 r，使得齐次多项式

$$(x_1+x_2+\cdots+x_n)^r f(x_1,x_2,\cdots,x_n)$$

展开合并同类项后，全部单项式的系数都是正的.

1940 年，Habicht 利用 Polya 定理，构造性地给出了正定齐次多项式的有理函数平方和的有效表示[7]. 对一般的半正定情况，给出构造性算法还是一个十分困难的问题.

1967 年，Motzkin 寻找到了第一个齐次多项式，它可以表为有理函数的平方和，但不能表为 SOS. 这个多项式是 $Z^6+X^4Y^2+X^2Y^4-3X^2Y^2Z^2$.

1995 年，Choi，Lam 和 Reznick 合作给出了多项式的 Gram 矩阵表示方法[8]. 随后 Powers 和 Wormann 在 1998 年得到了一个将半正定多项式表为 SOS 的算法[9].

2002年，Prajna，Papachristodoulou，Seiler，Parrilo 4 人合作开发了一个程序 SOSTOOLS，在 Matlab 平台上实现了上面提到 Powers 和 Wormann 的算法. 这个程序使用了半正定规划[10]（SDP）方法. 其中半正定规划的作用相当于本文中柱形代数剖分算法的作用. 不同之处在于半正定规划是数值算法，所以效率颇高，但是也具有数值方法所共有的弱点，会出现误差；而柱形代数剖分是符号算法，优点是准确，弱点是效率不高. 两种方法互有补充.

一个非常详尽的关于 Hilbert 第 17 问题的发展报告可以参见文献[11].

随着时代的进步，高速电子计算机的出现以及近年来机器证明数学定理的巨大进展，Hilbert 第 17 问题的构造性解决问题，又吸引了众多的研究者. 由于代数不等式的机器可读证明一直是自动推理领域中一个比较困难的研究课题，而平方和表示则使得不等式机器证明的可读性成为显然，显示了这一方法的优越性. 另一方面，从上面平方和表示的历史，我们知道存在着不能表为多项式平方和的半正定多项式，这又展现出平方和表示方法的严峻性.

本文第 2 节将介绍多项式的 Gram 矩阵表示以及 Powers，Wormann 所给出的平方和算法，其中加入了我们对该算法的一点补充，即用实闭域上量词消去的柱形剖分算法寻找半代数系统的特解；第 3 节是本文的主要结果，建立了一个处理稀疏多项式幂半向量的新算法，并同时给出了用 Maple 语言编写的源程序；第 4 节给出了几个实例的处理过程.

2 多项式的 Gram 矩阵表示和 SOS 算法

我们用 $H_d(\mathbb{R}^n)$ 表示实数域 $\mathbb{R}$ 上所有 n 元 d 次齐次多项式组成的集合.

设 $\alpha=(\alpha_1,\alpha_2,\cdots,\alpha_n)\in\mathbb{N}_0^n$（$\mathbb{N}_0$ 是非负整数集合），记 $x^\alpha=x_1^{\alpha_1}x_2^{\alpha_2}\cdots x_n^{\alpha_n}$，$|\alpha|=\sum_{k=1}^{n}\alpha_k$，$c(\alpha)\in\mathbb{R}$. 于是 n 元 $2d$ 次齐次多项式可以写成

$$f(x_1,x_2,\cdots,x_n)=\sum_{|\alpha|=2d}c(\alpha)x^\alpha \tag{1}$$

简记半正定的齐次多项式为 PSD，可以表为多项式平方和的多项式为 SOS. 由前面叙述的历史已经看到有

$$SOS\subset PSD.$$

注意到半正定的齐次多项式其次数必然是偶数. 如果 $2d$ 次齐次多项式 f 是 SOS，即有

$$f=\sum_{j=1}^{m}h_j^2,\ \ h_j\in H_d(\mathbb{R}^n) \tag{2}$$

$h_j=\sum_{|\beta|=d}u_\beta^{(j)}x^\beta$，又设

$$\boldsymbol{V}_\beta=[u_\beta^{(1)},u_\beta^{(2)},\cdots,u_\beta^{(m)}]\in\mathbb{R}^m \tag{3}$$

于是

$$\begin{aligned}f&=\sum_{j=1}^{m}h_j^2=\sum_{j=1}^{m}\Big(\sum_{|\beta|=d}u_\beta^{(j)}x^\beta\Big)\Big(\sum_{|\beta'|=d}u_{\beta'}^{(j)}x^{\beta'}\Big)\\&=\sum_{\beta,\beta'}(\boldsymbol{V}_\beta\boldsymbol{V}_{\beta'}^{\mathrm{T}})x^{\beta+\beta'}\end{aligned} \tag{4}$$

比较式(1)和式(4)中多项式的系数得到

$$c(\alpha)=\sum_{\beta+\beta'=\alpha}(\boldsymbol{V}_\beta^{\mathrm{T}}\boldsymbol{V}_{\beta'}).$$

记 $\boldsymbol{B}_{\beta\beta'}=\boldsymbol{V}_\beta\boldsymbol{V}_{\beta'}^{\mathrm{T}}$，设 $\boldsymbol{B}$ 为矩阵$[\boldsymbol{B}_{\beta\beta'}]$，显见 $\boldsymbol{B}$ 是对称半正定的. 我们称矩阵 $\boldsymbol{B}$ 是多项式 f 的 Gram 矩阵.

反过来，如果对称矩阵 $\boldsymbol{B}$ 可以分解为 $\boldsymbol{B}=\boldsymbol{V}^{\mathrm{T}}\boldsymbol{V}$，$\boldsymbol{V}$ 是一个实矩阵. 并且满足 $c(\alpha)=\sum_{\beta+\beta'=\alpha}(\boldsymbol{B}_{\beta\beta'})$，那么我们就可以逆推上面步骤得到多项式 f 的平方和

表示. 这样我们就证明了下面的基本定理 1.

定理 1[9]. 齐次多项式 $f(x_1,x_2,\cdots,x_n)=\sum_{|\alpha|=2d}c(\alpha)x^\alpha\in H_{2d}(\mathbb{R}^n)$ 可以表为 SOS 的充要条件是，存在实对称半正定矩阵 $\boldsymbol{B}=[u_{ij}]$，满足 $f(x_1,x_2,\cdots,x_n)=\boldsymbol{X}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{X}$，$\boldsymbol{X}$ 是由变元 $x_1,x_2,\cdots,x_n$ 的所有可能出现的齐 d 次单项式组成的列向量.

定理 1 中的向量 $\boldsymbol{X}^{\mathrm{T}}$（或 $\boldsymbol{X}$）简称为多项式 f 的幂半向量，$f=\boldsymbol{X}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{X}$ 则称为多项式 f 的 Gram 矩阵表示.

于是我们看到，对一个给定的多项式 f，将它表示成 SOS 的关键在于求出它一个具体的 Gram 矩阵表示. 为此还需要进一步的结果作基础.

定理 2. 实对称矩阵 $\boldsymbol{B}=[u_{ij}]$ 是半正定的，等价于下列条件之一：

(1) 矩阵 $\boldsymbol{B}$ 的特征多项式的根都是非负的.

(2) 存在实矩阵 $\boldsymbol{V}$，满足 $\boldsymbol{B}=\boldsymbol{V}^{\mathrm{T}}\boldsymbol{V}$.

(3) 矩阵 $\boldsymbol{B}$ 的所有主子式都是非负的.

定理 2 的结论是熟知的，几乎每一本高等代数书上都有证明，可参看文献[13].

由著名的笛卡尔符号法则有如下结果.

定理 3[1]. 如果实系数多项式的根都是实的，则它的正根个数等于这个多项式的系数组的变号次数.

推论 1. 假如实系数多项式的 $g(Y)=Y^k+b_{k-1}Y^{k-1}+\cdots+b_0$ 的根都是实的，则它的所有根都是非负的充要条件是 $(-1)^{i+k}b_i\geq 0$，$i=0,1,\cdots,k-1$.

推论的必要性由根与系数关系的韦达定理推知. 充分性则由定理 3 保证.

作为另一工具的胞腔分解算法（或柱形代数剖分（CAD）算法），是 Collins 在 1975 年首次提出的（见文献[12]），经过后人的大量改进工作，已经得到很好的实现和应用. CAD 算法是解决实闭域上量词消去问题的完全算法，所以用来求式（*）这样的半代数系统的特解是很自然的思路. 实际编程求特解时，我们应用的是不完全柱形代数剖分 PCAD，这是已经实现的 Maple 程序[2]. 这方面现在已有很多好的参考方献，比如文献[3]，这里不再详谈.

有了上面的准备，现在可以叙述平方和表示的算法了.

算法 1（SOS）. 任给一齐次多项式 $f(x_1,x_2,\cdots,x_n)$，本算法将判断 f 是否能表为 SOS，并且在肯定回答时输出 SOS 表示式.

1. 计算 f 的次数 $de:=deg(f)$，如果 de 为奇数，则输出：f 不能表为 SOS，并停机；否则执行步 2.

2. 展开 $(x_1+x_2+\cdots+x_n)^{de/2}$，分离出每个单项式，形成列向量 $\boldsymbol{X}$.

3. 作 $f-\boldsymbol{X}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{X}=0$（$\boldsymbol{B}$ 为一待定对称矩阵），展开对照方程两边系数，得到一线性方程组. 解线性方程组求出对称矩阵 $\boldsymbol{B}$；#$\boldsymbol{B}$ 可能含有很多自由变量.

4. 计算矩阵 $\boldsymbol{B}$ 的特征多项式 $F(Y)$，并用定理 3 的推论，算出多项式 $F(Y)$ 所有根非负的条件，得一个半代数系统 K.

5. 用柱形代数分解算法（CAD），计算半代系统 K 的实解样本集 $D(K)$.

6. 如果 $D(K)$ 为空集，则输出：f 不能表为 SOS，并停机；否则执行下面步骤：

6.1 任取 $D(K)$ 中一样本作为特解，重写矩阵 $\boldsymbol{B}$，并将矩阵 $\boldsymbol{B}$ 分解为 $\boldsymbol{B}=\boldsymbol{V}^{\mathrm{T}}\boldsymbol{V}$ 的形式；

6.2 计算范数式 $\|(\boldsymbol{V}\boldsymbol{X})\|^2$，输出 SOS 表示式.

这个算法是综合了文献[8,11]的原始思想而得到的，这里也作了一点补充. 其正确性和终止性由前面的定理 1 到定理 3 的推论以及柱形代数剖分算法保证.

3　求稀疏多项式幂半向量的算法和源程序

算法 1 从效率上看，并不好，原因在于 Gram 矩阵 $\boldsymbol{B}$ 的阶增长太快，而算法 CAD 是一个完全精确的符号算法，这就导致了效率不高. 另一方面，我们大多时候遇到多项式的项并不是完全的，而是有许多项的系数为 0，这就是稀疏多项式，对这种情况自然希望将它的 Gram 矩阵的阶控制在较小的范围之内. 在这里我们将建立一种全新的纯代数的处理方法.

定理 4. 设齐 $2d$ 次多项式

$$f(x_1,x_2,\cdots,x_n)=\sum_{|\alpha|=2d}c(\alpha)x^\alpha\in H_{2d}(\mathbb{R}^n)$$

可以表示为 $\sum_{j=1}^{m}h_j^2$，$h_j\in H_d(\mathbb{R}^n)$，$g(x_1,x_2,\cdots,x_n)=\sum_{|\beta|=d}b(\beta)x^\beta$ 是一个带有文字的齐 d 次多项式，其中系数 $b(\beta)$ 是相互独立的文字或 0，使得通过相应的数值代换到 g 中文字，可获得每个多项式 h_j（$j=1,2,\cdots,m$）. 若多项式 $g^2-f=\sum_{|\delta|=2d}a(\delta)x^\delta$ 的某个系数 $a(\delta_r)$ 是某个文字 $b(\beta_s)$ 的平方，则必有

(1) $\delta_r=2\beta_s$；

(2) 每个 h_j 中项 x^{β_s} 前的系数为 0.

证明. 注意到

$$g^2-f=\Big(\sum_{|\beta|=d}b(\beta)x^\beta\Big)^2-\sum_{|\alpha|=2d}c(\alpha)x^\alpha=$$
$$\sum_{|\beta|=d}(b(\beta)x^\beta)^2+\sum_{\substack{|\beta|=d,|\beta'|=d\\ \beta\neq\beta'}}2(b(\beta)x^\beta)(b(\beta')x^{\beta'})-$$
$$\sum_{|\alpha|=2d}c(\alpha)x^\alpha=\sum_{|\delta|=2d}a(\delta)x^\delta.$$

因为 $b(\beta)$ 是未定的文字,第二个等号显示了合并展开式的时候,不会出现某些项系数被合并掉的情况. 如果某个系数 $a(\delta_r)$ 为某个文字 $b(\beta_s)$ 平方,则这一项只能由多项式 g 的项 $b(\beta_s)x^\beta$ 平方来提供,所以必有(1)成立. 这也就意味着 f 中 $x^{2\beta_s}$ 项的系数是 0.

另一方面,由于通过相应的数值代换到 g 中文字,可获得每个多项式 h_j,从而通过相同的代换可由 g^2-f 获得 $h_j^2-f(j=1,2,\cdots,m)$. 由条件知,$x^{2\beta_s}$ 在 h_j^2-f 中的系数等于 $b(\beta_s)^2$ 在代换后的值,即等于 d_j^2,这里 d_j 为 x^{β_s} 在 h_j 中的系数,$j=1,2,\cdots,m$. 由于 $(1-m)f=\sum_{j=1}^m h_j^2-mf=\sum_{j=1}^m(h_j^2-f)$,所以 $x^{2\beta_s}$ 在 $(1-m)f$ 中的系数恰好是 $\sum_{j=1}^m d_j^2$. 由前面讨论,f 中 $x^{2\beta_s}$ 项的系数是 0,即有 $\sum_{j=1}^m d_j^2=0$. 所以 $d_j=0$,$j=1,2,\cdots,m$. 从而叙述(2)也获证. 证毕.

现举例来说明定理 4 如何用于简化计算幂半向量 $\boldsymbol{X}^{\mathrm{T}}$ 的,如对多项式 $x^2y^2-2x^2yz+x^2z^2$,设一个系数未定的齐二次多项式 $g=(a_1x^2+a_2y^2+a_3z^2+a_4xy+a_5yz+a_6xz)$,做 g^2-f 并展开,可以见到项 x^4,y^4,z^4,y^2z^2 前的系数分别是 a_1^2,a_2^2,a_3^2,a_5^2,于是由定理 4,如果 $x^2y^2-2x^2yz+x^2z^2=\sum_{j=1}^m h_j^2$,$h_j\in H_2(\mathbb{R}^3)$,则必有 $h_j=axy+bxz$ 的形式. 故算出幂半向量 $\boldsymbol{X}^{\mathrm{T}}=(xy,xz)$. 而实际上显然有 $x^2y^2-2x^2yz+x^2z^2=(xy-xz)^2$. 和这个例完全相同的处理,我们总结出如下算法,它的正确性由定理 4 保证.

算法 2. 任给 $2d$ 次齐次多项式 $f(x_1,x_2,\cdots,x_n)$,本算法将输出 $f(x_1,x_2,\cdots,x_n)$ 的简化幂半向量 $\boldsymbol{X}^{\mathrm{T}}$.

1. 展开多项式 $(x_1+x_2+\cdots+x_n)^d$,分离出每个单项式,形成列向量 $\boldsymbol{X}$;

2. 计算向量 $\boldsymbol{X}$ 的维数 p,做系数待定的多项式 $g=\langle\boldsymbol{X}^{\mathrm{T}},\boldsymbol{S}\rangle$,$\boldsymbol{S}$ 是 p 维待定列向量. #$\langle\boldsymbol{X}^{\mathrm{T}},\boldsymbol{S}\rangle$ 表示向量的内积.

3. 令 $G_1:=g^2-f$, $G_2:=0$;

4. 循环执行下面步骤直到 $G_1-G_2=0$;

4.1 令 $G_2:=G_1$;

4.2 搜索 G_2 中每一项前的系数,如果有某一项前的系数仅是 $\boldsymbol{S}$ 中某个分量的平方,则挑出这一项和它的系数,记为 $\boldsymbol{X}_1$ 和 $\boldsymbol{S}_1$;否则令 $G_1:=G_2$;

4.3 令 $\boldsymbol{X}:=\boldsymbol{X}-\boldsymbol{X}_1$, $\boldsymbol{S}:=\boldsymbol{S}-\boldsymbol{S}_1$, $g=\langle\boldsymbol{X}^{\mathrm{T}},\boldsymbol{S}\rangle$, $G_1:=g^2-f$; #$\boldsymbol{X}-\boldsymbol{X}_1$ 表示在 $\boldsymbol{X}$ 中除去 $\boldsymbol{X}_1$ 得到的向量,其余类似表示式意义相同;

5. 输出 $\boldsymbol{X}^{\mathrm{T}}$,停机.

这个算法很容易由 Maple 程序来实现. 下面是 hafemi 源程序. 当我们求一个多项式 f 的幂半向量时,只需输入'hafemi(f,[ ]);','[ ]'中是多项式 f 变量的有序表.

```
hafemi := proc(f,var1)
local f1,list1,d1,n1,s1,exp1,t1,tt,t,se,W2,se1,m2,
  mt1,W,W1,lcoe,j1,j2,W5,W4,typ,p:
  with(linalg):
  f1 := expand(f): list1 := var1:
  d1 := degree(f1,list1)/2: n1 := nops(var1):
  s1 := sum(list1[i],i=1..n1):
  exp1 := expand(s1^d1):
    coeffs(exp1,list1,'t'):
    t1 := t:
    mt1 := convert(convert([t1],array),matrix):
  m2 := nops(exp1):
  W := array(1..m2,[]):
  W1 := evalm(W&* mt1)[1]:
  W2 := 0:
  while W1-W2<>0 do W2 := W1:
    W5 := expand(W2^2-f1):
    for j2 to n1-1 do
      W4 := subs(var1[j2]=var1[j2]* tt,W5):
      W5 := (subs(tt=1,W4)+
            subs(tt=-1,W4))/2: od:
      se := {}:
    lcoe := {coeffs(expand(W5),list1)}:
    for j1 to nops(lcoe) do
      typ := indets(lcoe[j1]):
      if nops(typ)=1 then se1 := solve(lcoe[j1]):
        if {se1}[1]=0 then
          se := {op(se),op(typ)=0}:
        else se := se:
        fi:
      else se := se:
      fi: od:
    W1 := subs(op(se),W2):
  od:
  coeffs(W1,list1,'p'):
  [p]:
end:
```

4　几个实例

下面用前文建立的算法来处理几个具体问题.

例 1.　证明多项式 $f(x,y,z)=z^6+x^4y^2+x^2y^4-3x^2y^2z^2$ 不是 SOS.

证明.　很容易由算术-几何平均不等式证明 $f(x,y,z)$ 是 PSD，Motzkin 在 1967 年证明了这个多项式不是 SOS. 现在我们用本文建立的算法来处理它. 这是一相当稀疏的多项式，利用算法 2 很快得到它的幂半向量为 $\boldsymbol{X}^{\mathrm{T}}=(z^3,x^2y,xy^2,xyz)$.

计算出 $f(x,y,z)$ 的 Gram 矩阵表示，所得结果是唯一的，即

$$z^6+x^4y^2+x^2y^4-3x^2y^2z^2=\boldsymbol{X}^{\mathrm{T}}\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&-3\end{bmatrix}\boldsymbol{X} \tag{6}$$

很明显式(6)中的矩阵不是半正定的. 也就是说不存在半正定的矩阵 $\boldsymbol{B}$ 使多项式 $f(x,y,z)$ 有表示式 $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{X}$，所以由定理 1 知这个多项式不是 SOS.

注记：$(z^2+x^2)(z^6+x^4y^2+x^2y^4-3x^2y^2z^2)=(z^4-x^2y^2)^2+(x^3y-z^2xy)^2+(xz^3-xy^2z)^2$.

例 2.　请将多项式 $f(x,y,z)=z^8y^2-xy^2z^7+4y^6x^4+\sqrt{2}x^2z^8$ 表示为多项式的平方和.

用算法 2 计算出 $\boldsymbol{X}^{\mathrm{T}}=(z^2y^2x,z^3yx,z^4y,z^4x,y^3x^2)$，接着执行算法 1 依次来得到

$$\boldsymbol{B}=\begin{bmatrix}1&0&0&-1&0\\0&2&\frac{-1}{2}&0&0\\0&\frac{-1}{2}&2&0&\frac{-1}{2}\\-1&0&0&\sqrt{2}&0\\0&0&\frac{-1}{2}&0&4\end{bmatrix},$$

$\boldsymbol{B}$ 可分解为 $\boldsymbol{B}=\boldsymbol{V}^{\mathrm{T}}\boldsymbol{V}$.

$$\boldsymbol{V}=\begin{bmatrix}1&0&0&-1&0\\0&\sqrt{2}&\frac{-\sqrt{2}}{4}&0&0\\0&0&\frac{\sqrt{14}}{4}&0&\frac{-\sqrt{14}}{7}\\0&0&0&\sqrt{\sqrt{2}-1}&0\\0&0&0&0&\frac{\sqrt{182}}{7}\end{bmatrix},$$

最后得到

$$\begin{aligned}f(x,y,z)&=-z^7xy^2+z^8y^2+\sqrt{2}z^8x^2+4x^4y^6\\&=(z^2xy^2-z^4x)^2+\left(\sqrt{2}z^3xy-\frac{\sqrt{2}}{4}z^4y\right)^2+\\&\quad\left(\frac{1}{4}\sqrt{14}z^4y-\frac{1}{7}\sqrt{14}y^3z^2\right)^2+\\&\quad(\sqrt{2}-1)z^8y^2+\frac{26x^6y^6}{7}.\end{aligned}$$

前面我们已经给出了算法 2 用 Mpale 编制的源程序，而 PCAD 也已经有现成的 Mpale 程序，基于上述两点，我们编制了 Maple 程序 sosdecomp，实现了平方和分解表达式的自动生成. 利用这一程序，我们发现了文献[9]的一处错误. 文献[9]在第 102 页中的例 2 谈到多项式

$$f(x,y,z)=x^4+2x^2y^2+x^3z+z^4$$

不是一个 SOS，而程序 sosdecomp 输出了如下结果：

$$f(x,y,z)=\left(xz+\frac{1}{2}x^2\right)^2+\left(\frac{\sqrt{3}x^2}{2}-\frac{\sqrt{3}z^2}{3}\right)^2+2x^2y^2+\frac{2z^4}{3},$$

经展开验证，这一表示式是正确的.

5　结束语

我们在第 1 节中已经提到，本文工作的主要目的是为不等式型定理机器可读证明的自动生成，提供一条可能的探索途径，这个途径涉及到著名的 Hilbert 17 问题，所以其难度也是相当大的. 就目前而言，这个领域还相当的不成熟，还有大量的工作需要做. 已经知道 Hilbert 17 问题的构造性是一个 NP 难度的问题，也许一个好的思路是对此问题的一些子类先进行研究，如人们比较感兴趣的、而且也比较重要的对称半正定多项式类，现在对这个子类已经有一些结果出现了.

参　考　文　献

1　Yang Lu, Zhang Jing-Zhong, Hou Xiao-Rong. Nonlinear Algebraic Equation System and Automated Theorem Proving. Shanghai: Shanghai Science's and Technology's Education Publishing House, 1996(in Chinese)
(杨　路，张景中，侯晓荣. 非线性代数方程组与定理机器证明. 上海：上海科技教育出版社，1996)

2　Yang Lu, Xia Shi-Hong. Automated proving for a class of constructive geometric inequalities. Chinese Journal of Computers, 2003, 26(7): 769～778(in Chinese)

(杨 路,夏时洪.一类构造性几何不等式的机器证明.计算机学报,2003,26(7):769～778)

3 Yang Lu. Computational real algebraic geometry. In: Wang Dong-Ming *et al*. eds.. Selected Lecture in Symbolic Computation. Beijing: Tsinghua University Press, 2003(in Chinese)
(杨 路.计算实代数几何.见:王东明等主编.符号计算选讲.北京:清华大学出版社,2003)

4 Hilbert D.. Uber die Darstellung definiter Formen als Summe von Formenquadraten. Math. Ann., 1888, 32: 342～350

5 Atin E.. Uber die Darstellung definiter Funktionen in Quadrate. Hamb. Abh., 1927, 5: 100～115

6 Polya G.. Uber positive Darstellung von Polynomen. Vierteljschr Natruforsch Ges. Zurich, 1928, 73: 141～145

7 Habicht W.. Uber die Zerlegung strikte definiter Formen in Quadrate. Comment. Math. Helv, 1940, 12: 317～322

8 Choi M. D., Lam T. Y., Reznick B.. Sum of squares of real polynomials. Proc. of the Symposia in Pure Mathematics, 1995, 58(2): 103～126

9 Powers V., Wormann T.. An algorithm for sums of squares of real polynomials. Journal of Pure and Applied Linear Algebra, 1998, 127: 99～104

10 Vandenberghe L., Boyd S.. Semidefinite programming. SIAM Review, 1996, 38(1): 49～95

11 Reznick B.. Some concrete aspects of Hilbert's 17th problem. Contemp Math., 2000, 253: 251～272

12 Collins G. E.. Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In: Lecture Notes in Computer Science 33, Springer-Verlag, 1975

13 Zhang X. K., Xu P. H.. Advance Algebra. 2nd Edition. Beijing: Tsinghua University Press, 2004(in Chinese)
(张贤科,许甫华编著.高等代数学.第2版.北京:清华大学出版社,2004)

14 Reznick B.. Extremal PSD forms with few terms. Duke Mathematical Journal, 1987, 45: 363～374

YAO Yong, born in 1974. His major research interests include automated theorem proving and symbolic computation.

FENG Yong, born in 1965, professor, Ph. D. supervisor. His major research interests include symbolic and numerical computation and its applications.

Background

This work is supported by the National Basic Research Program of China (973 Program) under grant No. 2004CB318003, which title is "The High Performance Algorithms in Real Geometry and Real Algebra". This paper presents a high efficient algorithm to decompose a sort of positive semi-definite sparse polynomials into sum of squares.